

Industrial Automation Headquarters

Taiwan: Delta Electronics, Inc.
Taoyuan Technology Center
No.18, Xinglong Rd., Taoyuan District,
Taoyuan City 33068, Taiwan
TEL: +886-3-362-6301 / FAX: +886-3-371-6301

Asia

China: Delta Electronics (Shanghai) Co., Ltd.
No.182 Minyu Rd., Pudong Shanghai, P.R.C.
Post code : 201209
TEL: +86-21-6872-3988 / FAX: +86-21-6872-3996
Customer Service: 400-820-9595

Japan: Delta Electronics (Japan), Inc.
Industrial Automation Sales Department
2-1-14 Shibadaimon, Minato-ku
Tokyo, Japan 105-0012
TEL: +81-3-5733-1155 / FAX: +81-3-5733-1255

Korea: Delta Electronics (Korea), Inc.
1511, 219, Gasan Digital 1-Ro., Geumcheon-gu,
Seoul, 08501 South Korea
TEL: +82-2-515-5305 / FAX: +82-2-515-5302

Singapore: Delta Energy Systems (Singapore) Pte Ltd.
4 Kaki Bukit Avenue 1, #05-04, Singapore 417939
TEL: +65-6747-5155 / FAX: +65-6744-9228

India: Delta Electronics (India) Pvt. Ltd.
Plot No.43, Sector 35, HSIIDC Gurgaon,
PIN 122001, Haryana, India
TEL: +91-124-4874900 / FAX: +91-124-4874945

Thailand: Delta Electronics (Thailand) PCL.
909 Soi 9, Moo 4, Bangpoo Industrial Estate (E.P.Z),
Pattana 1 Rd., T.Phraksa, A.Muang,
Samutprakarn 10280, Thailand
TEL: +66-2709-2800 / FAX: +66-2709-2827

Australia: Delta Electronics (Australia) Pty Ltd.
Unit 2, Building A, 18-24 Ricketts Road,
Mount Waverley, Victoria 3149 Australia
Mail: IA.au@deltaww.com
TEL: +61-1300-335-823 / +61-3-9543-3720

Americas

USA: Delta Electronics (Americas) Ltd.
5101 Davis Drive, Research Triangle Park, NC 27709, U.S.A.
TEL: +1-919-767-3813 / FAX: +1-919-767-3969

Brazil: Delta Electronics Brazil Ltd.
Estrada Velha Rio-São Paulo, 5300 Eugênio de
Melo - São José dos Campos CEP: 12247-004 - SP - Brazil
TEL: +55-12-3932-2300 / FAX: +55-12-3932-237

Mexico: Delta Electronics International Mexico S.A. de C.V.
Gustavo Baz No. 309 Edificio E PB 103
Colonia La Loma, CP 54060
Tlalnepantla, Estado de México
TEL: +52-55-3603-9200



Дистриб'ютор в Україні

Україна: ТОВ "Системи реального часу - Україна"
www.delta-electronics.com.ua
вул. Святослава Хороброго, 29-А, 49001, м.Дніпро
Пошта: sales@rts.ua
ТЕЛ : +38 0562 392223 / +38 068 2392223

EMEA

EMEA Headquarters: Delta Electronics (Netherlands) B.V.
Sales: Sales.IA.EMEA@deltaww.com
Marketing: Marketing.IA.EMEA@deltaww.com
Technical Support: iatechnicalsupport@deltaww.com
Customer Support: Customer-Support@deltaww.com
Service: Service.IA.emea@deltaww.com
TEL: +31(0)40 800 3900

BENELUX: Delta Electronics (Netherlands) B.V.
Automotive Campus 260, 5708 JZ Helmond, The Netherlands
Mail: Sales.IA.Benelux@deltaww.com
TEL: +31(0)40 800 3900

DACH: Delta Electronics (Netherlands) B.V.
Coesterweg 45, D-59494 Soest, Germany
Mail: Sales.IA.DACH@deltaww.com
TEL: +49(0)2921 987 0

France: Delta Electronics (France) S.A.
Zl du bois Challand 2, 15 rue des Pyrénées,
Lisses, 91090 Evry Cedex, France
Mail: Sales.IA.FR@deltaww.com
TEL: +33(0)1 69 77 82 60

Iberia: Delta Electronics Solutions (Spain) S.L.U
Ctra. De Villaverde a Vallecas, 265 1º Dcha Ed.
Hormigueras - P.I. de Vallecas 28031 Madrid
TEL: +34(0)91 223 74 20
Carrer Llacuna 166, 08018 Barcelona, Spain
Mail: Sales.IA.Iberia@deltaww.com

Italy: Delta Electronics (Italy) S.r.l.
Via Meda 2-22060 Novedrate(CO)
Piazza Grazioli 18 00186 Roma Italy
Mail: Sales.IA.Italy@deltaww.com
TEL: +39 039 8900365

Turkey: Delta Greentech Elektronik San. Ltd. Sti.(Turkey)
Şerifali Mah. Hendem Cad. Kule Sok. No:16-A
34775 Ümraniye - İstanbul
Mail: Sales.IA.Turkey@deltaww.com
TEL: + 90 216 499 9910

MEA: Eltek Dubai (Eltek MEA DMCC)
OFFICE 2504, 25th Floor, Saba Tower 1,
Jumeirah Lakes Towers, Dubai, UAE
Mail: Sales.IA.MEA@deltaww.com
TEL: +971(0)4 2690148

Керівництво з програмування серії AS



Оцифрована автоматизація для світу, що змінюється

Керівництво з програмування серії AS

Керівництво з програмування серії AS

Версія	Ревізія	Дата
1 ^{ст}	Першу версію було опубліковано.	2016 / 11 / 30
2-й	1. Оновлену інформацію в розділі 2. 1. 1, 2. 2. 7, 2. 2. 8, 2. 2. 14, 2. 2. 16. 2. Додано нові інструкції в розділі 3. 3. Оновлено та додано нові інструкції в розділі 6. 4. Оновлену інформацію в Розділі 7.	2018 / 05 / 07
3-й	1. Оновлена інформація про SM у розділі 2. 1. 1. 2. Оновлена інформація SM, включаючи SM 784-815, SM896-927, SM 1440-1447 і додані SM 478-479, SM 498-499, SM 518-519, SM 538-539, SM 558-559, SM 578 - 579, SM 1581 - 1608, SM1611 - 1618, SM 1621 - 1628, SM 1683 - 1684 і SM 1691 - 1698 SM інсекти іони 2. 2. 7. 3. Оновлено таблицю часу оновлення SM і додано новий час оновлення SM у таблиці в розділі 2. 2. 8. 4. Оновлені SR 182 - 183, SR 217 - 218, SR 424, SR 825 - 893, SR 1336 і SR 1376, додані до 2. 2. 14. 5. Додано 187-188 SR, 478-479 SR, 498-499 SR, 518-519 SR, 538-539 SR , 558-559 SR , 578-579 SR , 640-651 SR , 751-758 SR , і SR 761 - 768 додатки і на 2. 2. 14. 6. Оновлені описи в пункті 4 Комунікаційні функції, пункті 14 Крива і пункті 18 Прапорці та реєстри щодо обміну даними. Додано пункт 15 Функція компенсації підтвердження в розділі 2. 2. 16. 7. До списків у Розділі 3 додано нові інструкції. 8. Додано нові інструкції в розділі 6: API 0222, API 0601, API 0711, A PI 1012, API 1013, A PI 1226, API 1402, API1403, API 1404, A PI 1405, API 1406, A PI 1407, API 08, API 1409, API 1410, A PI 1415, A PI 1416, API 1818, API 1819, API 1820, A PI 1906, API 2122, A PI 2123, API 2211, API 2212, API 2808, A PI 2809, API 2810 і A PI 2811. 9. Оновлені інструкції в Розділі 6: API 0222, A PI 0600, API 0702, A PI 0708, API 0709, A PI 0710, API 1004, API 1601, API 1607, A PI 1808, API 1812, A PI 1814, API 1815, API1816, API 1817, A PI 2200, API 2200 - 2203, API 2208, A PI2210, API 2300, A PI 2301, API 2302, A PI 2303, API 2401, API 2500, API 2700, A PI 2701, API 2702, A PI 2703, API 2704, API 2705, API 2706, A PI 2707, API 2708, A PI 2709, API 2710, API 2711, API 2712, A PI 2713, API 2714, A PI 2715, API 2716, API 2717, API 2718, A PI2721, API 2723, A PI 2800, API 2801, API 2802, API 2803, A PI 2804, API 2805, A PI 2806, API 2807 і A PI 2808. 10. Оновлені рішення для помилок, включаючи коди 7203, 8105, 8106 і 8107.	2018 / 05 / 22
4 ^{тис}	1. Виправлення вручну на пристрої, що використовується в SR 182, 183, 185, 210, 213, 421, 422, 423 і 424.	2018 / 06 / 12
5 ^{тис}	1. Додано описи модулів CP U серії AS 300/AS 200	2019/ 5/ 10

Версія	Ревізія	Дата
	у розділі 1. 1. 2 Опис моделі. 2. Додано й оновлено SM і SR у розділі 2: SM 230 - SM 261, SM 585 - SM 587, SM 1110, SM 1685, SR 652 - SR 655, SR 791 - SR 806, SR 811 - SR 818, SR 823 - SR 824, SR 1015, SR 1520 - SR 1540, SR 1559 - SR 1592, SM76 - SM 91, SM 168 - SM171, SM 455, SM 1006 - SM 1031, SM 1184 - SM 1199, SM 1216 - SM 1231, SM 1248 - SM 1263, SM 1271 - SM 1299, SM 1424 - SM 1439, SM 1681, SM 1709 - SM 1919, SR 166 - 179 SR, 182 - 183 SR, 187 SR - 188 SR, 215 - 216 SR, 1000 SR - 1014 SR і 1435 SR - 1511 SR. 3. Додано та оновлено інструкції в розділі 3: API 0602, API 1417, A PI 2304, API 2724, A PI 2812 - A PI 2816, API0711, API 1110, A PI 1111, A PI 1402, A PI 1407, API 1408 і API 1416. 4. Додано відомості про використання реєстрів індексів в інструкціях щодо помилок у розділі 4.4. 5. Оновлені інструкції в розділі 5: PED, APE D, OPED, NE D, AN ED, ONE D, P LS і P LF. 6. Оновлені інструкції в розділі 6: API 0602, A PI 1417, API 2304, A PI 2724, API 2812 - API 2816, API 0018 - API 0023, API 0708 - API 0711, API 1004 - API 1007, API 1110, API 1111, API 1226, A PI 1402 - API 1410, A PI 1415 - A PI 1416, API 1816, API 2200, A PI 2201, API 2203, A PI 2208, API 2210 - API 2212, API 2300 - API 2302, A PI 2704, A PI 2706 - A PI 2709, API 2712 - API 2716, A PI 2719, API 2800, A PI 2803, API 2805 і A PI 2807. 7. Оновлена екстрена інформація в розділі 7. 3. 1 і 7. 3. 2. Додано коди помилок 2030 і 2031.	
6 тис	1. Виправлення вручну в описах кодів функцій K 5566 і K 5577 SR 36 і SR 412 і SR 415 у Розділ 2. 2. Оновлено описи в SM 1375 ~SM 1447 у розділі 2. 3. Оновлено описи в SR 1000~SR 1006 у розділі 2. 4. Оновлено опис у цьому коді 16#00xx у розділі 2. 5. Додано стандартну частоту введення в інструкцію D CN T (API 1004) у розділі 6. 6. Оновлено допустимий діапазон введення для S₁ в інструкції PWD (API 1009) у розділі 6. 7. Оновлено таблиці функцій позначок вирівнювання в інструкції DD RVM (API 2709) у розділі 6. 8. Додано приклад використання символів переривання та прапорів завершення в інструкції TPO (A PI 2718) у розділі 6.	2019/ 6/ 21

7 тис	1. Оновлений опис AS300 , AS 200 , AS02 , 04P U-A в Розділі 1 1. 1. 2. 2. Додано нову інформацію про продукт для AS 100, AS 02H C - A, AS 04 SIL-A, AS-FPF N 02 і AS-FO PC 02 у розділі 1. 3. Оновлене співвідношення між дією PLC і типом пристрою в розділі 2. 1. 3. Додано оновлену інформацію AS 100 або мат SM 1375 ~ SM 1477 і додано SM 1448 ~ 1455 , SM 1686 у Розділ 2. 2. 7 Спеціальні допоміжні реле (SM) і розд. 2. 2. 14 Спеціальні реєстри даних (SR). Додано інструкцію CN T щодо інформації або підрахунку в розділі 2. 2. 11 Лічильники. Додано DC NT Інструкція в розділі 2. 2. 12 32-розрядних лічильників (HC). Додана часто відображена інформація про реєстри Розділ і на 2. 2. 13 . Додано нові спеціальні реєстри даних (SR): SR 42, S R48, SR 54, SR 60, S R66, S R72, SR 78, SR 84, S R90, SR 96, SR 102, SR 108, SR 114, SR 120, SR 126, SR 132, SR 138 , SR 144 , SR 150 , SR 156 , SR 405 , SR 406 і видалить е SR 1540 у розділі і на 2. 2. 14 . Додано # 2 коди помилок або кодів SM27 / SR 27, оновлена форма №22 у F-мітках SM 230 ~ 261, оновлено # 7 Помилка PLC або інформація про журнал, додано # 24 Прапорці автоматичного налаштування для імпульсного виходу в розділі 2. 2 . 16. 4. Додано та оновлено інструкції в розділі 3: API 1014, API 1015, A PI 1227, API 1418 - API 1423, API 1821, API 2817, API 2818, A PI 1102 - API 1108, API 1114, API 1115, API 2806, API 1415. 5. Додано як розділ, що пояснює, які інструкції слід використовувати в Ladder Programming Language у розділі 4. 3. 6. Усі інструкції внутрішнього посібника HE підтримують тип даних U INT. Видалить UI NT зі списку підтримки в Розділі 6. Оновлені інструкції в Розділі 6: API 0202, API 0216, A PI 0501, API 0600, A PI 0708, API 0709, API 0710, API 0801, A PI 1002, API 1003, A PI 1004, API 1005, API 1006, API 1007, A PI 1009, API 1102 - API 1108, API 1114, API 1115, API 1210, A PI 1212, API 1213, A PI 1226, API 1401, API 1402, API 1404, A PI 1405, API 1407, A PI 1408, API 1415, API 1416, API 1417, A PI 1501 - API 1504, A PI 1808, API 2106, API 2116, API 2117, A PI 2119, A PI 2200, A PI 2204, API 2208, API 2209, API 2212, A PI 2301, API 2704, A PI 2706, API 2708, API 2709, API 2718, A PI 2800, API 2803, A PI 2806, API 2812 - API 2815. Додано нові інструкції в розділі 6: API 1014, API 1015, API 1227, A PI 1418 - API 1423, A PI 1426, API 1427, API 1821, API 2817 та API 2818. 7. Додано пояснення щодо позиції кроку в розділі 7. 1. 4. 8. Оновлено коди помилок 2001–2027 у розділі 7. 2. 8 та 7. 2. 9. Оновлено індикатори ПОМИЛОК, які блимають щоразу 0,2 секунди для модулів AIO та AS 02 LC у розділі 7.3.1. Додано коди помилок 16#1500 і 16#1505 для або AS 00 S CM у розділі I на 7. 3. 4.	2020 / 12 / 08
-------	---	----------------

8 тис	1. Додано нову інформацію про продукт для AS-PS 03 C, AS-AT XB та оновлену інформацію про продукт для AS-FE N02, AS-FP FN 02, AS-FOPC 02 у розділі 1. 2. Додано й оновлено SM і SR у розділі 2: SM27, SM120 - 151, SM360 - 391, SM603, SM 1160 - 1161, SM 1700 - 1705 SR , 27 SR , 29 - 31 SR , 1016 - 1019 SR , 1600 - 1605 SR , SR 1700 , SR 1702 , SR 1704 , SR 1706 , SR 1708 , SR 1710 . Видалено наступні SM і SR з розділу 2: SM 468, SM488, SM 508, SM 528, SM548, SM 568, SR 604-609. Додано та оновлено наступну інформацію в розділі 2. 2. 16: 227 SR nf orm at i on, back k las hc om pensation inf or mat i on, Ether net IP related f lags, SM 230 - 261 , aut o-прапорці скидання або імпульсного виходу, прапорці автоматичного скидання для зв'язку Delt a CAN, як використовувати DP LS Y на високій швидкості порівняння на перерві (I 2 xx) і як використовувати компенсацію зворотного удару в CSFO. 2. Додано та оновлено інструкції в розділі 3: API 1229, API 1424, A PI 1425, API 1426, A PI 1427 та API 1228. 3. Оновлені інструкції в розділі 6: API 0501, A PI 0708, API 0709, A PI1004, API 1228, A PI 1400, API 1401, API 1402, API 1403, A PI1404, API 1405, A PI 1406, API 1407, API 1408, API 1409, A PI 1410, API 1417, A PI 1418, API 1419, API 1420, API 1421, A PI 1422, API 1423, A PI 2300, API 2301, API 2302, API 2304, A PI 2703, API 2706, A PI 2710, API 2712, API2714, API 2810, A PI 2817 та API 2818. Додано нові інструкції в розділі 6: API 1229, PI 1424, A PI1425, API 1426 і API 1427. 5. Додано коди помилок 16#19 B 0 - 16 #19 FC у розділі 7. 2. 9. Додано та оновлено інформацію для AS 02/ 04P U, AS02 HC, AS 04 SIL, A S01 DN ET і AS00 SC M інсекцій 7. 3 та 7. 4.	2021/06/24
9 тис	1. Додано нові інструкції API 1016 - A PI 1018, API 1430, API 2214, A PI 2215, API 2216, A PI2305, API 2820, API2821, API 2822, A PI 2823 та оновлені інструкції AP I 1226, API 1228 у розділі 3. 2. Додано інструкцію CA N RS для відкритих інструкцій із спілкування CAN під час використання програмування сходової ланки мова в розділі і на 4. 3 . 3. Оновлено підтримувану версію I SPS часто для інструкцій PLF у Розділі 5. 4. Додано нові інструкції API 1016 - A PI 1018, API 1430, API 2214, A PI 2215, API 2216, A PI 2305, API 2214, API 2215, API 2216, A PI 2305, API 2820, A PI 2821, API 2822, API 2823 та оновив наступні інструкції API 0222, API 0600, API 1100, A PI 1101, A PI 1226, A PI 1228, API 1421, API 1422, API 1424, A PI 1425, API 1516, A PI 1819, API 1821, API 1901, API 2704, A PI 2717, API 2718, A PI 2719, API 2720, API 2721,	2022/08/01
	API 2724, A PI 2800, API 2801, A PI 2802, API 2803, API 2804, API 2805, A PI 2806, API 2807, A PI 2809, API 2810, API 2811, API 2812, A PI 2817 та API 2818 у розділі 6. 5. Додано нові помилки або коди 1 6 #000 D, 16# 0020, 16#0021, 16 #0028 , 16# 0040 - 16#0047 , 16 #1000 - 16#10 FF, 16# 140C та оновлені коди помилок або кодів 16 #1900 - 16# 191 C у розділі і на 7. 2 . 2.	

Керівництво з програмування серії AS

Зміст

Розділ 1 Вступ

1.1 Огляд	1-2
1.1.1 Відповідні посібники	1-2
1.1.2 Опис моделі	1-2
1.2 Програмне забезпечення	1-10
1.2.1 Редактор програм	1-10
1.2.2 Організаційні підрозділи програми та завдання	1-12

Розділ 2 Пристрої

2.1 Знайомство з пристроями	2-2
2.1.1 Таблиця пристроїв.....	2-2
2.1.2 Базова структура сховища вводу-виводу	2-3
2.1.3 Зв'язок між дією ПЛК і типом пристрою	2-3
2.1.4 Зафіксовані зони в діапазоні пристрою	2-4
2.2. Функції пристрою	2-5
2.2.1 Значення та константи	2-5
2.2.2 Числа з плаваючою комою	2-7
2.2.3 Рядки	2-8
2.2.4 Вхідні реле (X)	2-9
2.2.5 Вихідні реле (Y)	2-10
2.2.6 Допоміжні реле (M)	2-10
2.2.7 Спеціальні допоміжні реле (SM).....	2-11
2.2.8 Час оновлення для спеціальних допоміжних реле.....	2-71
2.2.9 Реле (S).....	2-79
2.2.10 Таймери (T)	2-79
2.2.11 Лічильники	2-81
2.2.12 32-розрядні лічильники (HC)	2-83
2.2.13 Регістри даних (D)	2-85
2.2.14 Спеціальні регістри даних (SR)	2-86
2.2.15 Умови оновлення регістрів спеціальних даних	2-133
2.2.16 Додаткові зауваження щодо спеціальних допоміжних реле та спеціальних даних	
Регістри	2- 137
2.2.17 Індексні регістри (E)	2-171
2.2.18 Регістри файлів (FR).....	2-171

Розділ 3 Таблиці інструкцій

3.1 Типи інструкцій	3-2
3.1.1 Основні інструкції	3-2
3.1.2 Прикладні інструкції	3-2
3.2 Розуміння таблиць інструкцій	3-3
3.2.1 Основні інструкції	3-3
3.2.2 Застосовувані інструкції (відсортовані за числами)	3-4
3.2.3 Застосовувані інструкції (відсортовані за алфавітом).....	3-5
3.2.4 Таблиці пристроїв	3-6
3.3 Списки основних інструкцій	3-8
3.4 Списки прикладних інструкцій	3-11
3.4.1 Застосовувані інструкції (відсортовані за номерами API)	3-11
3.4.2 Прикладні вказівки (відсортовані за алфавітом)	3-48

Розділ 4 Структура інструкції

4.1 Прикладні інструкції - опис API	4-2
4.2 Опис використання операнда	4-4
4.3 Обмеження щодо використання інструкцій	4-7
4.4 Індексні реєстри	4-9
4.5 Реєстри показчиків	4-12
4.6 Реєстри показчиків таймерів	4-14
4.7 Реєстри показчиків для 16-розрядних лічильників	4-16
4.8 Реєстри показчиків для 32-розрядних лічильників	4-18
4.9 Регістрація файлів	4-20

Розділ 5 Основні інструкції

5.1 Перелік основних інструкцій	5-2
5.2 Основні інструкції	5-3

Розділ 6 Прикладні інструкції

6.1 Інструкції з порівняння	6-4
6.1.1 Список інструкцій для порівняння	6-4
6.1.2 Пояснення інструкцій порівняння.....	6-7
6.2 Арифметичні вказівки	6-46
6.2.1 Список арифметичних інструкцій	6-46
6.2.2 Пояснення арифметичних інструкцій	6-47
6.3 Інструкції з перетворення даних	6-78
6.3.1 Список інструкцій щодо перетворення даних.....	6-78
6.3.2 Пояснення вказівок щодо перетворення даних	6- 79
6.4 Інструкції з передачі даних	6-122
6.4.1 Перелік інструкцій щодо передачі даних.....	6-122
6.4.2 Пояснення інструкцій щодо передачі даних.....	6-123
6.5 Інструкції зі стрибків	6-150
6.5.1 Список інструкцій зі стрибків	6-150
6.5.2 Пояснення інструкцій зі стрибка.....	6-151
6.6 Інструкції щодо виконання програми	6-159
6.6.1 Перелік інструкцій щодо виконання програми	6-159
6.6.2 Пояснення інструкцій щодо виконання програми	6-160
6.7 Інструкції з оновлення ІО	6-172
6.7.1 Список інструкцій щодо оновлення ІО	6-172
6.7.2 Пояснення вказівок щодо оновлення ІО	6-173
6.8 Різні інструкції	6-181
6.8.1 Список зручних інструкцій	6-181
6.8.2 Пояснення зручних інструкцій.....	6-182
6.9 Логічні інструкції	6-235
6.9.1 Список логічних інструкцій	6-235
6.9.2 Пояснення логічних інструкцій	6-236
6.10 Інструкції щодо обертання	6-257
6.10.1 Список інструкцій щодо обертання	6-257
6.10.2 Пояснення інструкцій щодо обертання	6-258
6.11 Інструкції щодо таймера та лічильника	6-269
6.11.1 Перелік інструкцій таймера та лічильника.....	6-269
6.11.2 Пояснення інструкцій таймера та лічильника	6-270
6.12 Інструкції щодо зміни	6-317
6.12.1 Список інструкцій щодо зміни	6-317

6.12.2 Пояснення вказівок зміни.....	6-318
6.13 Інструкції з обробки даних	6-359
6.13.1 Список інструкцій з обробки даних	6-359
6.13.2 Пояснення інструкцій з обробки даних	6-360
6.14 Інструкції зі створення структури	6-425
6.14.1 Список інструкцій зі створення структури.....	6-425
6.14.2 Пояснення інструкцій зі створення структури	6-426
6.15 Інструкції модуля	6-434
6.15.1 Перелік інструкцій модуля	6-434
6.15.2 Пояснення інструкцій модуля	6-435
6.16 Числа з плаваючою комою Інструкції	6-560
6.16.1 Список інструкцій для чисел з плаваючою комою	6-560
6.16.2 Пояснення інструкцій для чисел з плаваючою комою	6-561
6.17 Інструкції щодо годинника реального часу	6-596
6.17.1 Список інструкцій для годинника реального часу.....	6-596
6.17.2 Пояснення вказівок годинника реального часу	6- 597
6.18 Вказівки периферійних пристроїв	6-628
6.18.1 Перелік периферійних інструкцій	6-628
6.18.2 Пояснення периферійних інструкцій	6-629
6.19 Інструкції щодо зв'язку	6-645
6.19.1 Перелік інструкцій щодо зв'язку.....	6-645
6.19.2 Пояснення інструкцій зв'язку	6-646
6.19.3 Описи пов'язаних із зв'язком прапорів і регістрів	6-733
6.20 Інші інструкції	6-737
6.20.1 Перелік інших інструкцій	6-737
6.20.2 Пояснення до інших інструкцій	6-738
6.21 Інструкції з обробки рядка	6-750
6.21.1 Список інструкцій з обробки рядків	6-750
6.21.2 Пояснення інструкцій з обробки рядків	6-751
6.22 Інструкції Ethernet	6-814
6.22.1 Список інструкцій Ethernet	6-814
6.22.2 Пояснення інструкцій Ethernet	6-815
6.23 Інструкції щодо карти пам'яті	6-878
6.23.1 Перелік інструкцій щодо карти пам'яті.....	6-878
6.23.2 Пояснення інструкцій щодо картки пам'яті	6 -879
6.24 Інструкції з керування завданнями	6-904
6.24.1 Перелік інструкцій із керування завданнями	6-904

6.24.2 Пояснення інструкцій з керування завданнями	6-905
6.25 Інструкції SFC	6-909
6.25.1 Перелік інструкцій SFC	6-909
6.25.2 Пояснення інструкцій SFC	6-910
6.26 Інструкції щодо високошвидкісного виведення	6-917
6.26.1 Список інструкцій високошвидкісного виводу.....	6-917
6.26.2 Пояснення інструкцій високошвидкісного виведення	6-918
6.27 Інструкції зі зв'язку Delta CANopen	6-1032
1. Перелік інструкцій із зв'язку Delta CANopen	6-1032
2. Пояснення інструкцій зі зв'язку Delta CANopen	6-1033
3. Поширені запитання в Delta Special CANopen Communication and Troubleshooting	6-1116
 Розділ 7 Усунення несправностей	
7.1 Усунення несправностей	7-3
7.1.1 Основні дії з усунення несправностей	7-3
7.1.2 Очистити стани помилок	7-3
7.1.3 Усунення несправностей SOP	7-4
7.1.4 Системний журнал і позиції кроку помилки	7-5
7.2 Усунення несправностей для модулів процесора	7-7
7.2.1 Світлодіодні індикатори ERROR	7-7
2. Світлодіодні індикатори ERROR блимають кожні 0,5 секунди	7-7
3. Світлодіодні індикатори ERROR швидко блимають кожні 0,2 секунди	7-9
4. Світлодіодні індикатори ERROR Повільно блимають кожні 3 секунди та світяться протягом 1 секунда	7-9
7.2.5 BAT. Світлодіодні індикатори LOW	7-10
6. BAT. LOW Світлодіодні індикатори блимають кожні 0,5 секунди	7-10
7. Світлодіодні індикатори RUN і ERROR блимають одночасно 0,5 секунди	7-11
8. Індикатори RUN і LED блимають один за одним 0,5 секунди.	7-11
7.2.9 Інші помилки (без світлодіодних індикаторів).....	7-11
7.3 Усунення несправностей для інших модулів вводу/виводу.....	7-19
1. Усунення несправностей для аналогових модулів (AD/DA/XA) і температури Модулі (RTD/TC)	7-19
1. Пошук і усунення несправностей для модуля позиціонування AS02/04PU	7-20
2. Пошук і усунення несправностей модуля високошвидкісного лічильника AS02HC ...	7-20
3. Пошук і усунення несправностей модуля тензодатчика AS02LC	7-21
4. Усунення несправностей для модуля AS04SIL IO-Link як зв'язкуМодуль.....	7-22

6.	Усунення несправностей для модуля AS00SCM як модуля зв'язку.....	7-25
7.	Усунення несправностей для модуля AS00SCM як віддаленого модуля	7-26
7.3.8	Усунення несправностей для модулів AS01DNET	7-27

7.4 Коды помилок і світлодіодні індикатори для модулів 7-28

1.	Коды помилок і світлодіодні індикатори для модулів центрального процесора	7-28
2.	Коды помилок і світлодіодні індикатори для аналогових/температурних модулів...	7-35
3.	Коды помилок і світлодіодні індикатори для модуля позиціонування AS02 / 04PU..	7-36
4.	Коды помилок і світлодіодні індикатори для модуля високошвидкісного лічильника AS02HC	7-36
5.	Коды помилок та світлодіодні індикатори для модуля тензодатчика AS02LC.....	7-37
6.	Коды помилок і світлодіодні індикатори для модуля AS04SIL IO-Link as комунікаційний модуль	7-37
7.	Коды помилок і світлодіодні індикатори для модуля AS00SCM як комунікаційного модуля.....	7 -38
8.	Коды помилок і світлодіодні індикатори для модуля AS00SCM як віддаленого модуля.....	7 -38
9.	Коды помилок і світлодіодні індикатори для модуля AS01DNET (Режим Master/Slave)	7-39
10.	Коды помилок і світлодіодні індикатори для модуля AS01DNET (режим RTU)	7-40

Розділ 1 Вступ

	Зміст
1.1 Огляд	1-2
1.1.1 Супутні посібники	1-2
1.1.2 Опис моделі	1-2
1.2 Програмне забезпечення	1-10
1.2.1 Редактор програм	1-10
1.2.2 Організаційні підрозділи програми та завдання	1-12

1.1 Огляд

У цьому посібнику описано, як програмувати програмовані логічні контролери серії AS, основні і прикладні інструкції. Для роботи з програмним забезпеченням вам потрібно ISPSoft, DIADesigner або DIADesigner-AX. Якщо ви використовуєте ЦП ПЛК серії AS, зверніться до посібника користувача ISPSoft або посібника DIADesigner для отримання додаткової інформації.

1.1.1 Пов'язані посібники

Відповідні посібники для програмованих логічних контролерів серії AS наведено нижче.

- Швидкий старт серії AS

Це допоможе вам розпочати роботу з системою, перш ніж читати інші пов'язані посібники.

- Керівництво з програмування серії AS (цей посібник)

Це познайомить вас із програмуванням програмованих логічних контролерів серії AS, основними інструкціями та прикладними інструкціями.

- Посібник користувача ISPSoft / DIADesigner

Це знайомство з програмним забезпеченням ISPSoft / DIADesigner, яке ви використовуєте для програмування програмованих логічних контролерів серії AS. Він описує мови програмування (схеми сходів, списки інструкцій, послідовні функціональні діаграми, функціональні блок-схеми та структуровані тексти), концепцію POU та концепцію завдань.

- Керівництво з обладнання та експлуатації серії AS

Тут описано електричні характеристики, зовнішній вигляд, розміри, функції центрального процесора, пристрої, таблиці модулів, усунення несправностей і так далі.

- Посібник модуля серії AS

Це вводить використання спеціальних модулів введення/виведення; наприклад: мережеві модулі, аналогові модулі введення/виведення, модулі вимірювання температури та інші.

1.1.2 Опис моделі

Класифікація	Назва моделі	опис
Модуль живлення	AS-PS02	Вхід: 100-240 В змінного струму, 50/60 Гц Вихід: 24 В постійного струму/2 А, 48 Вт (для внутрішнього використання ПЛК)
	AS-PS02A	Вхід: 100-240 В змінного струму, 50/60 Гц Вихід: 24 В постійного струму/1,5 А, 36 Вт (для внутрішнього використання ПЛК) Вихід: 24 В постійного струму/0,5 А, 12 Вт (для зовнішнього використання)
Адаптер модуля живлення	AS-ATXB	Використовується із зовнішніми джерелами живлення для стійок ПЛК, які потребують понад 2 А. Для полегшення проводки цей адаптер дозволяє проводити електропроводку в нижній частині модуля, а не на лівій стороні модуля.
Модуль ЦП серії AS300	AS332P-A	Модуль ЦП, вхід живлення 24 В постійного струму, вихід PNP, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, 2 функціональні карти (опціонально), підтримка 32 входів/виходів (16DI+16DO) і до 1024 введення/виведення. Ємність програми: 128 тис. кроків, клемні блоки високої щільності
	AS332T-A	Модуль процесора, вхід живлення 24 В постійного струму, вихід NPN, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, 2 функціональні карти (опціонально), підтримка 32 входів/виходів (16DI+16DO) і до 1024 введення/виведення. Ємність програми: 128 тис. кроків, клемні блоки високої щільності

Класифікація	Назва моделі	опис
	AS324MT-A	Модуль ЦП, вхід живлення 24 В постійного струму, диференціальний вихід NPN, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, 2 функціональні карти (опціонально), підтримка 24 входів/виходів (12DI+12DO) і до 1016 введення/виведення. Ємність програми: 128 тис. кроків, клемні блоки високої щільності
	AS320P-B	Модуль ЦП, вхід живлення 24 В постійного струму, вихід PNP, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, 2 функціональні карти (опціонально), підтримка 20 входів/виходів (8DI+12DO) і до 1024 введення/виведення. Місткість програми: 128K кроків, знімні клемники
	AS320T-B	Модуль ЦП, вхід живлення 24 В постійного струму, вихід NPN, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, 2 функціональні карти (опціонально), підтримка 20 входів/виходів (8DI+12DO) і до 1024 введення/виведення. Місткість програми: 128K кроків, знімні клемники
	AS300N-A	Модуль центрального процесора, вхідна напруга 24 В постійного струму, без входів/виходів, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, 2 функціональні карти (додатково) і підтримує до 1024 входів/виходів. Місткість програми: 128 тис. кроків
Модуль ЦП серії AS200	AS228P-A	Модуль ЦП, вхід живлення 24 В постійного струму, вихід PNP, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 28 входів/виходів (16DI+12DO) і до 1024 входів/виходів . Місткість програми: 64K кроків, знімні клеми
	AS228T-A	Модуль ЦП, вхід живлення 24 В постійного струму, вихід NPN, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 28 входів/виходів (16DI+12DO) і до 1024 входів/виходів . Місткість програми: 64K кроків, знімні клеми
	AS228R-A	Модуль процесора, вхід живлення 24 В постійного струму, релейний вихід, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 28 входів/виходів (16DI+12DO) і до 1024 входів/виходів . Місткість програми: 64K кроків, знімні клеми
	AS218PX-A	Модуль ЦП, вхід живлення 24 В постійного струму, вихід PNP, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 18 входів/виходів (8DI+6DO+2AI+2AO) і до 1024 введення/виведення. Місткість програми: 64K кроків, знімні клеми
	AS218TX-A	Модуль ЦП, вхід живлення 24 В постійного струму, вихід NPN, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 18 входів/виходів (8DI+6DO+2AI+2AO) і до 1024 введення/виведення. Місткість програми: 64K кроків, знімні клеми
	AS218RX-A	Модуль ЦП, вхід живлення 24 В постійного струму, релейний вихід, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 18 входів/виходів (8DI+6DO+2AI+2AO) і до 1024 введення/виведення. Місткість програми: 64K кроків, знімні клеми
Модуль ЦП серії AS100	AS132P-A	Модуль ЦП, вхідна напруга 100-240 В змінного струму, вихід PNP, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 32 входів/виходів (16DI+16DO) і до 1024 введення/виведення. Ємність програми: 64 тис. кроків, знімні клемники
	AS132T-A	Модуль ЦП, 100-240 В змінного струму, вихід NPN, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, CAN

Класифікація	Назва моделі	опис
		комунікаційний порт, що підтримує 32 входи/виходи (16DI+16DO) і до 1024 входів/виходів. Місткість програми: 64К кроків, знімні клеми
	AS132R-A	Модуль ЦП, вхідна напруга 100-240 В змінного струму, релейний вихід, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 32 входів/виходів (16DI+16DO) і до 1024 введення/виведення. Місткість програми: 64К кроків, знімні клеми
	AS148P-A	Модуль ЦП, вхідна напруга 100-240 В змінного струму, вихід PNP, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 48 входів/виходів (24DI+24DO) і до 1024 введення/виведення. Ємність програми: 64 тис. кроків, знімні клемники
	AS148T-A	Модуль ЦП, вхідна напруга 100-240 В змінного струму, вихід NPN, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 48 входів/виходів (24DI+24DO) і до 1024 введення/виведення. Місткість програми: 64К кроків, знімні клеми
	AS148R-A	Модуль ЦП, вхідна напруга 100-240 В змінного струму, релейний вихід, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 48 входів/виходів (24DI+24DO) і до 1024 введення/виведення. Місткість програми: 64К кроків, знімні клеми
	AS164P-A	Модуль процесора, вхідна напруга 100-240 В змінного струму, вихід PNP, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 64 входів/виходів (32DI+32DO) і до 1024 введення/виведення. Ємність програми: 64 тис. кроків, знімні клемники
	AS164T-A	Модуль ЦП, вхідна напруга 100-240 В змінного струму, вихід NPN, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 64 входів/виходів (32DI+32DO) і до 1024 введення/виведення. Місткість програми: 64К кроків, знімні клеми
	AS164R-A	Модуль ЦП, вхідна напруга 100-240 В змінного струму, релейний вихід, 1 порт Ethernet, 2 порти RS-485, 1 порт USB, 1 інтерфейс Micro SD, порт зв'язку CAN, підтримка 64 входів/виходів (32DI+32DO) і до 1024 введення/виведення. Місткість програми: 64К кроків, знімні клеми
Цифровий модуль введення/виведення	AS08AM10N-A	24 В постійного струму 5 мА 8 входів Клемна колодка з пружинним затискачем
	AS08AN01P-A	5 - 30 В постійного струму 0.5А/вихід, 4А/COM 8 виходів Пошук вихідних даних Клемна колодка з пружинним затискачем
	AS08AN01R-A	240 В змінного струму/24 В постійного струму 2А/вихід, 8А/COM 8 виходів реле Клемна колодка з пружинним затискачем
	AS08AN01T-A	5 - 30 В постійного струму 0.5А/вихід, 4А/COM 8 виходів

Класифікація	Назва моделі	опис
		Тонуть вихід Клемна колодка з пружинним затискачем
	AS16AM10N-A	24 В постійного струму 5 мА 16 входів Клемна колодка з пружинним затискачем
	AS16AN01P-A	5 - 30 В постійного струму 0.5А/вихід, 4А/COM 16 виходів Пошук вихідних даних Клемна колодка з пружинним затискачем
	AS16AN01R-A	240 В змінного струму/24 В постійного струму 2А/вихід, 8А/COM 16 виходів реле Клемна колодка з пружинним затискачем
	AS16AN01T-A	5 - 30 В постійного струму 0.5А/вихід, 4А/COM 16 виходів Тонуть вихід Клемна колодка з пружинним затискачем
	AS16AP11P-A	24 В постійного струму 5 мА 8 входів 5 - 30 В постійного струму 0.5А/вихід, 4А/COM 8 виходів Пошук вихідних даних Клемна колодка з пружинним затискачем
	AS16AP11R-A	24 В постійного струму 5 мА 8 входів 240VAC/24VDC 2А/вихід, 8А/COM 8 виходів реле Клемна колодка з пружинним затискачем
	AS16AP11T-A	24 В постійного струму 5 мА 8 входів 5 - 30 В постійного струму 0.5А/вихід, 4А/COM 8 виходів Тонуть вихід Клемна колодка з пружинним затискачем
	AS32AM10N-A	24 В постійного струму 3,2 мА

1

Класифікація	Назва моделі	опис
		32 входи роз'єм MIL
	AS32AN02T-A	5 - 30 В постійного струму 0,1 А/вихід, 3,2 А/COM 32 виходи Вихідний роз'єм MIL
	AS64AM10N-A	24 В постійного струму 3,2 мА 64 входи роз'єм MIL
	AS64AN02T-A	5 - 30 В постійного струму 0,1 А/вихід, 3,2 А/COM 64 виходи Вихідний роз'єм MIL
Модуль аналогового введення/ви ведення	AS04AD-A	4-канальний модуль аналогового введення Апаратна роздільна здатність: 16 біт 0–10 В, 0/1–5 В, від -5 до +5 В, від -10 до +10 В, 0/4–20 мА, -20–+20 мА Час перетворення: 2 мс/канал
	AS08AD-B	8-канальний модуль аналогового введення Апаратна роздільна здатність: 16 біт від 0 до +10 В, 0/1–5 В, від -5 В до +5 В, від -10 В до +10 В Час перетворення: 2 мс/канал
	AS08AD-C	8-канальний модуль аналогового введення Апаратна роздільна здатність: 16 біт 0/4–20 мА, -20 мА–+20 мА Час перетворення: 2 мс/канал
	AS04DA-A	4-канальний модуль аналогового виведення Апаратна роздільна здатність: 12 біт Від -10 до +10 В, 0–20 мА, 4–20 мА Час перетворення: 2 мс/канал
	AS06XA-A	4-канальний аналоговий вхід Апаратна роздільна здатність: 16 біт 0–10 В, 0/1–5 В, від -5 до +5 В, від -10 до +10 В, 0/4–20 мА, від -20 до +20 мА Час перетворення: 2 мс/канал 2- канальний аналоговий вихід Апаратна роздільна здатність: 12 біт Від -10 до +10 В, 0–20 мА, 4–20 мА Час перетворення: 2 мс/канал
	AS02ADH-A	2-канальний модуль аналогового введення Апаратна роздільна здатність: 16 біт 0–10 В, 0/1–5 В, від -5 до +5 В, від -10 до +10 В, 0/4–20 мА, від -20 до +20 мА Час високошвидкісного перетворення: 2 мкс Повна ізоляція (аналогові канали ізолювані один від одного.) Увімкнення/вимкнення функції реєстрації: 2000 даних на канал, пікові записи Фільтрація: фільтр низьких частот, смуговий фільтр

Класифікація	Назва моделі	опис
Модуль вимірювання температури	AS04RTD-A	4-канальний, 2-/3-провідний RTD Тип датчика: Pt100 / Ni100 / Pt1000 / Ni1000 / JPt100 / LG-Ni1000 / Cu50 / Cu100 / 0-300 Ом / 0-3000 Ом вхідний опір Роздільна здатність: 0,1°C/0,1°F (16 біт) Час перетворення: 200 мс/канал
	AS06RTD-A	6-канальний, 2-/3-провідний RTD Тип датчика: Pt100 / Ni100 / Pt1000 / Ni1000 / JPt100 / LG-Ni1000 / Cu50 / Cu100 / 0-300 Ом / 0-3000 Ом вхідний опір, Роздільна здатність: 0,1°C/0,1°F (16 біт) Час перетворення: 200 мс/канал
	AS04TC-A	4-канальна термopара Тип датчика: J, K, R, S, T, E, N, B і від -100 до +100 мВ Роздільна здатність: 0,1°C/0,1°F (24 біти) Час перетворення: 200 мс/канал
	AS08TC-A	8-канальна термopара Тип датчика: J, K, R, S, T, E, N, B і від -100 до +100 мВ Роздільна здатність: 0,1°C/0,1°F (24 біти) Час перетворення: 200 мс/канал
Модуль датчика навантаження	AS02LC-A	2-канальний, 4-провідний/6-провідний датчик навантаження Власні значення датчика навантаження: 1, 2, 4, 6, 20, 40, 80 мВ/В Найвища точність 1/10000 при 50 мс часу перетворення Роздільна здатність АЦП: 4 біти Час перетворення: 2,5–400 мс (дев'ять варіантів на вибір)
Модуль позиціонування	AS02PU-A	2-осьовий контроль руху, 5-24 В постійного струму, 1 диференціальний вхід (фаза A/B/Z), максимальна пропускна здатність апаратного входу при 200 кГц 24 В постійного струму, 5 мА, 5 входів, максимальна пропускна здатність апаратного введення при 1 кГц 5 В постійного струму, 2 осі, 4 диференціальних виходи, високошвидкісна вихідна частота 200 кГц
	AS04PU-A	Контроль руху по 4 осях 24 В постійного струму, 5 мА, 6 виходів, максимальна апаратна вхідна смуга пропускання на 1 кГц 5-30 В постійного струму, 0,1 А, 4 осі, 8 виходів NPN, максимальна частота високої швидкості при 100 кГц
Модуль лічильника	AS02HC-A	2-канальні швидкодіючі лічильники Методи введення для 2-канального: імпульсний вхід (макс. при 200 кГц) і вхід через комунікаційний інтерфейс SSI (макс. при 1,25 МГц) Збільшення / зменшення вхідного кодера 4-точковий високошвидкісний вихід з відкритим колектором, 5-30 В постійного струму, 0,1 А, робота з високошвидкісним диференціальним виходом
Мережевий модуль	AS00SCM-A	Модуль послідовного зв'язку, 2 порти зв'язку для комунікаційних карт, підтримка протоколів MODBUS
	AS01DNET-A	Комунікаційний порт DeviceNet, що працює як головний або підлеглий
	AS04SIL-A	Модуль IO-Link, вбудований з 4 комунікаційними портами IO-Link
Модуль віддаленого введення/виведення	AS00SCM-A +	Для функціональних карт AS-FCOPM

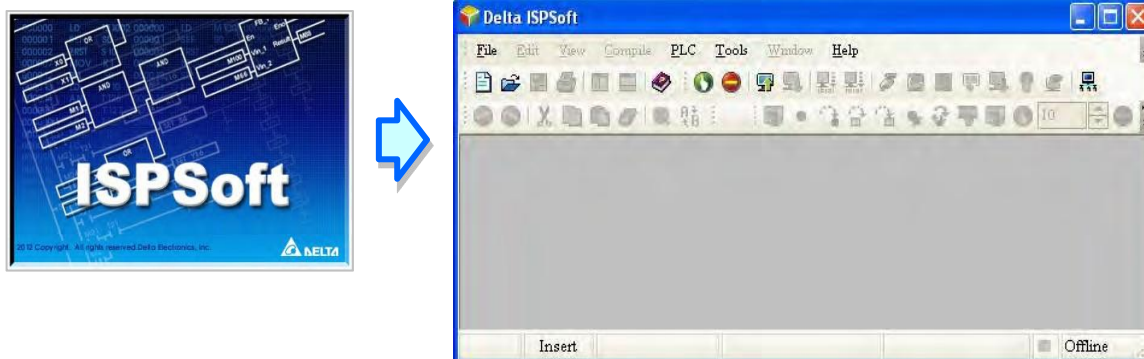
Класифікація	Назва моделі	опис
	AS-FCOPM	
	AS00SCM-A + AS-FEN02	Для функціональних карт AS-FEN02
	AS01DNET-A (RTU)	Віддалений підлеглий пристрій вводу-виводу DeviceNet, його права сторона підключається до модулів розширення серії AS, включаючи цифрові модулі, аналогові модулі, модулі температури тощо.
Функціональні карти	AS-F232	Послідовний порт зв'язку, RS232, працює як головний або підлеглий
	AS-F422	Послідовний порт зв'язку, RS422, працює як головний або підлеглий
	AS-F485	Послідовний порт зв'язку, RS485, працює як головний або підлеглий
	AS-FCOPM	Комунікаційний порт CANopen, що підтримує віддалені модулі серії DS301, AS і сервосистеми Delta
	AS-F2AD	2-канальний аналоговий вхід 0-10 В (12 біт), 4-20 мА (11 біт) Час перетворення: 3 мс/канал
	AS-F2DA	2-канальний аналоговий вихід 0-10 В, 4-20 мА (12 біт) Час перетворення: 2 мс/канал
	AS-FEN02	Доступно лише для процесора ПЛК серії AS300, 2 портів Ethernet, підтримки функції комутатора, Modbus TCP, адаптера EtherNet/IP і пульта дистанційного керування серії AS
	AS-FPFN02	Доступно лише для ЦП ПЛК серії AS300, 2 порти Ethernet, підтримка функції комутатора, пристрій PROFINET (підлеглий)
	AS-FOPC02	Доступно лише для ЦП ПЛК серії AS300, 2 порти Ethernet, підтримка функції комутатора, сервер OPC-UA, Modbus TCP
	AS-FFTP01	Доступно лише для ЦП ПЛК серії AS300, 1 порт Ethernet, підтримка FTP-сервера, OPC UA-сервера, клієнта MQTT, веб-сервера, журналу даних
	AS-FECAT	Доступно лише для ЦП ПЛК серії AS300, 2 порти Ethernet, підтримка обміну даними (до ввімкнення EtherCAT Master), сервер MODBUS TCP (1 з'єднання), 16-осьове керування позиціонуванням EtherCAT Master (доступно лише для приводу Delta)
Кабель для програмування	UC-PRG015-01A (1,5M)	Використовується для підключення між ПЛК і ПК через міні-порт USB, використовується для AS332T-A, AS332P-A і AS324MT-A
	UC-PRG030-01A (3M)	Використовуйте для підключення між ПЛК і ПК за допомогою міні-порту USB, використовуйте для AS332T-A, AS332P-A і AS324MT-A
	UC-PRG030-20A (3M)	Використовуйте для підключення між ПЛК і ПК за допомогою порту RJ45, використовуйте для AS332T-A, AS332P-A, AS324MT-A, AS-FEN02, AS-FPFN02 і AS-FECAT.
Подовжувач вводу/виводу	UC-ET010-24B (1M) UC-ET020-24B (2M) UC-ET030-24B (3M)	Роз'єм MIL, 40Pin ↔ 40-контактний, екранований, використовуйте для AS32AM10N-A, AS32AN02T-A, AS64AM10N-A та AS64AN02T-A
	UC-ET010-24D (1M) UC-ET020-24D (2M) UC-ET030-24D (3M)	Роз'єм MIL, 40Pin ↔ 2x 20-контактний, екранований, використовуйте для AS332T-A, AS332P-A, AS324MT-A, AS32AM10N-A, AS32AN02T-A, AS64AM10N-A та AS64AN02T-A
зовнішній	UB-10-ID16A	16 входів/виходів, 20-контактний роз'єм MIL, використовувати для AS332T-A, AS332P-

Класифікація	Назва моделі	опис
термінальний модуль		A, AS324MT-A, AS32AM10N-A, AS32AN02T-A, AS64AM10N-A та AS64AN02T-A
	UB-10-ID32A	32 входи, 40-контактний роз'єм MIL, використовуйте для AS32AM10N-A та AS64AM10N-A
	UB-10-IO32D	Клемний блок (пружинний затискач/роз'єм MIL), роз'єм MIL до 40-контактного клемного блоку з пружинним затискачем, використовуйте для AS332T-A, AS332P-A, AS324MT-A, AS32AM10N-A, AS32AN02T-A
	UB-10-OP16A	16 релейних виходів, 20-контактний роз'єм MIL, NPN, використовувати для AS332T-A, AS32AN02T-A і AS64AN02T-A
	UB-10-OP16Б	16 релейних виходів, 20-контактний роз'єм MIL, PNP, використовувати для AS332P-A
	UB-10-OT32A	32 транзисторних виходи, 40-контактний роз'єм MIL, NPN, використовувати для AS32AN02T-A і AS64AN02T-A
Мережеві кабелі	UC-CMC003-01A (0,3M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-CMC005-01A (0,5M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-CMC010-01A (1M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-CMC015-01A (1,5M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-CMC020-01A (2M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-CMC030-01A (3M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-CMC050-01A (5M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-CMC100-01A (10M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-CMC200-01A (20M)	Кабель зв'язку CANopen, використовуйте для серії AS-FCOPM
	UC-EMC003-02A (0,3M)	Кабель зв'язку Ethernet, використовуйте для модулів ЦП серії AS, функціональних карт AS-FEN02, AS-FPFN02 і AS-FECAT.
	UC-EMC005-02A (0,5M)	Кабель зв'язку Ethernet, використовуйте для модулів ЦП серії AS, функціональних карт AS-FEN02, AS-FPFN02 і AS-FECAT.
	UC-EMC010-02A (1M)	Кабель зв'язку Ethernet, використовуйте для модулів ЦП серії AS, функціональних карт AS-FEN02, AS-FPFN02 і AS-FECAT.
	UC-EMC020-02A (2M)	Кабель зв'язку Ethernet, використовуйте для модулів ЦП серії AS, функціональних карт AS-FEN02, AS-FPFN02 і AS-FECAT.
	UC-EMC050-02A (5M)	Кабель зв'язку Ethernet, використовуйте для модулів ЦП серії AS, функціональних карт AS-FEN02, AS-FPFN02 і AS-FECAT.
	UC-EMC100-02A (10M)	Кабель зв'язку Ethernet, використовуйте для модулів ЦП серії AS, функціональних карт AS-FEN02, AS-FPFN02 і AS-FECAT.
	UC-EMC200-02A (20M)	Кабель зв'язку Ethernet, використовуйте для модулів ЦП серії AS, функціональних карт AS-FEN02, AS-FPFN02 і AS-FECAT.

1.2 програмне забезпечення

1.2.1 Редактор програм

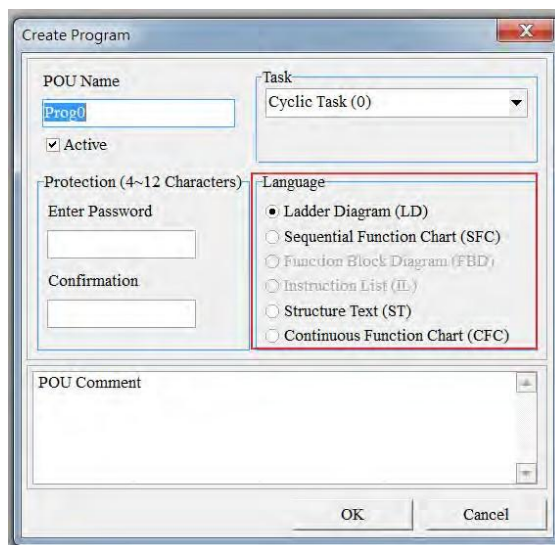
У розділі описано програмний редактор ISPSoft.



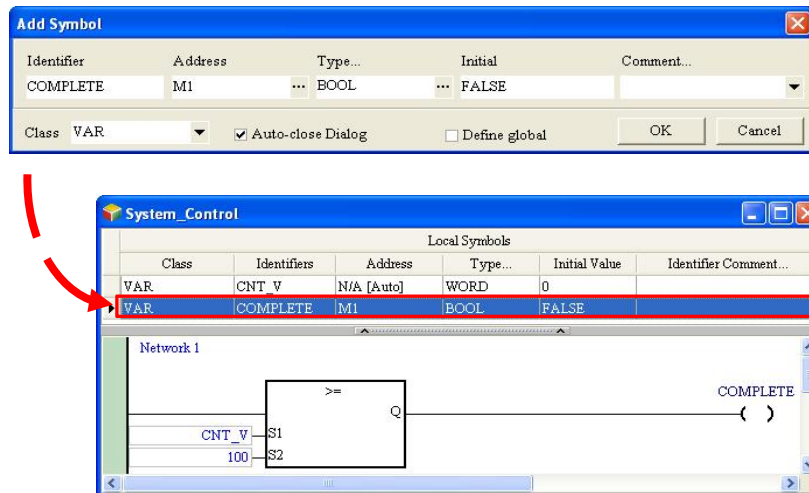
- Підтримуються такі мови програмування: структурний текст (ST), сходова діаграма (LD), послідовна функціональна діаграма (SFC), безперервна функціональна діаграма (CFC) і мова C.

ПРИМІТКА. Програмування CFC доступне лише в ISPSoft версії 3.01 або вище.

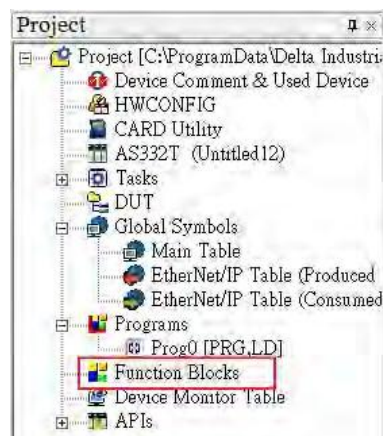
ПРИМІТКА. Мова C доступна лише в ISPSoft версії 3.08 або вище.



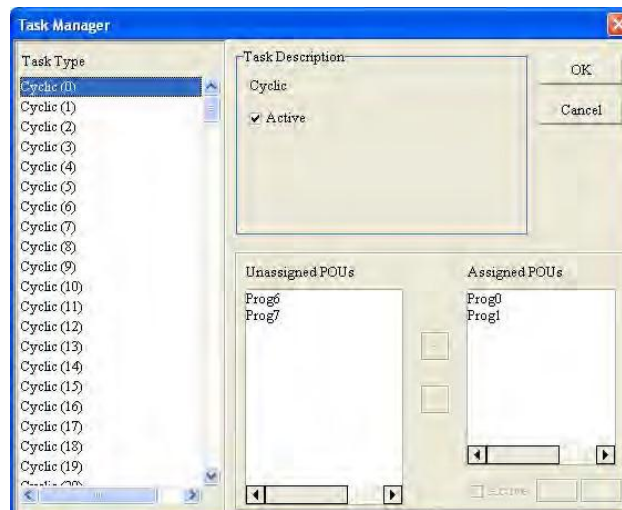
- Визначені користувачем змінні дозволяють визначити змінну для заміни імені пристрою ПЛК. Це покращує читабельність програми та економить час при зверненні до пристрою.



- Структура Program Organization Unit (POU) поділяє основну програму на кілька програмних одиниць, а також замінює традиційні підпрограми функціями та функціональними блоками. Це робить структуру програми модульною та легшою в управлінні.



- Завдання керують порядком виконання програм. Завдання допомагають керувати розробкою великомасштабної програми.



1.2.2 Організаційні підрозділи програми та завдання

1

Організаційні підрозділи програми (POU) є основними елементами, які складають програму PLC. На відміну від традиційної програми ПЛК, структура програми, запроваджена стандартом IEC 61131-3, дозволяє розділити велику програму на кілька малих блоків. Ці невеликі одиниці називаються POU. POU можна класифікувати на три типи.

1. Програма (PROG): Програма POU є основною програмою в ПЛК. Ви можете визначити виконання цього POU як циклічне сканування або переривання, а також упорядкувати порядок сканування в списку завдань для програмних POU.
2. Функціональний блок (FB): POU функціонального блоку (FB) схожий на підпрограму. Інструкції у функціональному блоці виконуються після того, як програма POU викликає функціональний блок із відповідними параметрами.
3. Функція (FC): функція (FC) POU схожа на макроінструкцію. Тобто ви можете записати багато операційних інструкцій або функцій у POU функціонального типу, а потім використовувати в POU програми або функціонального блоку POU.

Завдання - це функції, які керують порядком виконання програми або відповідно до певних умов переривання.

Завдання надає кожному POU програми певне завдання виконання та вказує порядок виконання для POU програми або спосіб їх увімкнення.

Зазвичай лише деякі програмні POU у проекті беруть участь у фактичному виконанні. Завдання контролює, чи виконувати програму POU чи ні, і як її виконувати. Якщо POU типу програми не призначено в завданні, POU програми зберігається як звичайний вихідний код разом із проектом замість компіляції в код виконання для ПЛК. Крім того, завданням потрібно призначити лише програмний POU. POU функціональних блоків або POU функцій автоматично викликаються програмою POU. Є три види завдань.

1. Циклічне завдання: POU програми, призначені циклічним завданням, скануються циклічно та виконуються ОК.
2. Завдання тимчасового переривання: якщо досягнуто часу переривання, усі POU програми, призначені завданню тимчасового переривання, виконуються по порядку.
3. Завдання умовного переривання: умовні переривання можна розділити на кілька типів, наприклад зовнішні переривання та переривання введення/виведення. Ви повинні переконатися, що ПЛК підтримує переривання, перш ніж використовувати умовні переривання в проекті. Якщо ви призначаєте POU програми для завдання умовного переривання, POU програми схоже на підпрограму переривання. Коли умова переривання задовольняється (наприклад, спрацьовує контакт зовнішнього переривання), тоді всі POU програми, призначені завданню умовного переривання, виконуються по порядку.

Розділ 2 Пристрої

Зміст

2.1 Знайомство з пристроями	2-2
2.1.1 Таблиця пристроїв	2-2
2.1.2 Базова структура сховища вводу-виводу	2-3
2.1.3 Зв'язок між дією ПЛК і типом пристрою	2-3
2.1.4 Зафіксовані області в діапазоні пристрою	2-4
2.2. Функції пристрою	2-5
2.2.1 Значення та константи	2-5
2.2.2 Числа з плаваючою комою	2-7
2.2.3 Рядки	2-8
2.2.4 Вхідні реле (X)	2-9
2.2.5 Вихідні реле (Y)	2-10
2.2.6 Допоміжні реле (M)	2-10
2.2.7 Спеціальні допоміжні реле (SM).....	2-11
2.2.8 Час оновлення для спеціальних допоміжних реле	2- 71
2.2.9 Реле (S)	2-79
2.2.10 Таймери (T)	2-79
2.2.11 Лічильники	2-81
2.2.12 32-розрядні лічильники (HC)	2-83
2.2.13 Регістри даних (D)	2-85
2.2.14 Спеціальні регістри даних (SR)	2-86
2.2.15 Умови оновлення регістрів спеціальних даних	2-133
2.2.16 Додаткові зауваження щодо спеціальних допоміжних реле та спеціальних даних Регістри	2-137
2.2.17 Індексні регістри (E)	2-171
2.2.18 Регістри файлів (FR)	2-171

2.1 Знайомство з пристроями

У цьому розділі описано значення та рядки, які обробляються ПЛК. Він також описує функції пристроїв, включаючи вхідні, вихідні та допоміжні реле, а також таймери, лічильники та регістри даних. ПЛК імітує зовнішні пристрої у внутрішній пам'яті ПЛК, тому слово «пристрій» є загальною назвою, яка стосується всіх місць внутрішньої пам'яті ПЛК. Пристроєм може бути бітовий пристрій або пристрій слова. Бітові пристрої імітують котушки, контакти та прапорці, тоді як словесні пристрої імітують регістри.

2.1.1 Таблиця пристроїв

Тип	Ім'я пристрою		Кількість пристроїв	Діапазон
Розрядний пристрій	Вхідне реле	X	1024	X0,0–X63,15
	Вихідне реле	Y	1024	Y0,0–Y63,15
	Регістр даних	D	48 000	D0.0–D29999.15
		W	48 000	W0.0–W29999.15 * 4
	Допоміжне реле	M	8192	M0–M8191
	Спеціальне допоміжне реле	SM	2048	SM0–SM4095
	реле	S	2048	S0–S2047
	Таймер	T	512	T0–T511
	Лічильник	C	512	C0–C511
	32-розрядний лічильник	HC	256	HC0–HC255
Пристрій Word	Вхідне реле	X	64	X0–X63
	Вихідне реле	Y	64	Y0–Y63
	Регістр даних	D	30000	D0–D29999
		W	30000	W0–W29999 * 4
	Спеціальне допоміжне реле	SR	2048	SR0–SR2047
	Регістр файлів	FR	65536	FR0–FR65535
	Таймер	T	512	T0–T511
	Лічильник	C	512	C0–C511
	32-розрядний лічильник	HC	256 (512 слів)	HC0–HC255
	Індексний реєстр	E	10	E0–E9
			5	E10–E14 * 4
Постійний* 1	Десяткова система	K	16 біт: від -32768 до 32767 32 біти: від -2147483648 до 2147483647	
Постійний* 2	Шістнадцяткова система	16#	16 біт: 16#0–16#FFFF 32 біти: 16#0–16#FFFFFFFF	
	Число з плаваючою комою одинарної точності	F	32 біти: $\pm 1,17549435 \cdot 10^{-38}$ до $\pm 3,40282347 \cdot 10^{+38}$	
Рядок* 3	Рядок	«\$»	1–31 символ	

*1: Константи позначені літерою K у списках пристроїв у розділах 5 і 6 Посібника з програмування серії AS.

Наприклад, якщо в посібнику з програмування AS з'являється «K50», в ISPSoft введіть лише число 50.

*2: Числа з плаваючою комою позначаються F/DF у списках пристроїв у розділах 5 і 6 Посібника з програмування серії AS, але в ISPSoft вони представлені десятковими комами. Наприклад, для числа з плаваючою комою F500 введіть 500.0 в ISPSoft.

*3: Рядки позначаються \$ у Розділі 5 і Розділі 6 Посібника з програмування серії AS, але в ISPSOft вони представлені лапками (« »). Наприклад, для рядка 1234 введіть «1234» у ISPSOft.

*4: Використовується лише для редагування в ISPSOft.

2.1.2 Базова структура сховища введення/виведення

пристрій	функція	Доступ по бітах	Доступ за словами	Змінити ISPSOft	Примусове ввімкнення/вимкнення біта
X	Вхідне реле	OK	OK	OK	OK
Y	Вихідне реле	OK	OK	OK	OK
M	Допоміжне реле	OK	-	OK	-
SM	Спеціальне допоміжне реле	OK	-	OK	-
S	реле	OK	-	OK	-
T	Таймер	OK	OK	OK	-
C	Лічильник	OK	OK	OK	-
HC	32-розрядний лічильник	OK	OK	OK	-
D	Регістр даних	OK	OK	OK	OK
SR	Регістр спеціальних даних	-	OK	OK	-
FR	Регістр файлів	-	OK * 1	-	-
E	Індексний реєстр	-	OK	OK	-

*1: Використовуйте інструкцію для написання FR.

2.1.3 Зв'язок між дією ПЛК і типом пристрою

Тип пристрою Дія ПЛК			Не закривається область		Закрита зона	
			Пристрій Y	Інші пристрої	Регістр файлів	Інші пристрої
Живлення: OFF. → УВІМК			Очищено	Очищено	Зберігається	Зберігається
Відновити налаштування за замовч.			Очищено	Очищено	Очищено	Очищено
STOP ↓ RUN * 1	Незамкнену область очищається.		Очищено	Очищено	Зберігається	Зберігається
	Стан незамкненої області зберігається.		Зберігається	Зберігається	Зберігається	Зберігається
RUN ↓ СТОП * 1	Стан пристрою Y очищено.	SM203 вимкнено. (Стан інших пристроїв зберігається.)	Очищено	Зберігається	Зберігається	Зберігається
	Стан пристрою Y зберігається.	SM203 вимкнено. (Стан інших пристроїв зберігається.)	Зберігається	Зберігається	Зберігається	Зберігається
	Стан пристрою Y очищено.	SM203 увімкнено. * 3 (Стан інших пристроїв очищається.)	Очищено	Очищено	Зберігається	Зберігається
	Стан пристрою Y зберігається.	SM203 увімкнено. (Стан інших пристроїв очищається.)	Зберігається	Очищено	Зберігається	Зберігається
SM204 увімкнено. * 2			Очищено	Очищено	Зберігається	Зберігається

Дія ПЛК (Усі незаблоковані області очищені.)	Тип пристрою		Зона без фіксатора		Закрита зона	
	Пристрій Y	Інші пристрої	Регістр файлів	Інші пристрої		
SM205 увімкнено.* 2 (Усі заблоковані області очищено.)	Зберігається	Зберігається	Зберігається	Очищено		

*1: Щоб дізнатися більше про налаштування станів, перегляньте HWCONFIG в ISPSOft. За замовч. для PLC STOP->RUN є «очистити незафіксовану область». За замовч. для PLC RUN->STOP є «очистити стан пристрою Y».

*2: ПЛК очищає зони, коли SM перемикається з OFF. на УВІМК. Після очищення зон ПЛК знову вимикає SM.

*3: Коли SM203 увімкнено, він очищає незафіксовану область один раз, коли PLC RUN->STOP. Ця функція має працювати з процесором AS PLC із програмною програмою версії 1.08.30 або новішої.

2.1.4 Зафіксовані області в діапазоні пристрою

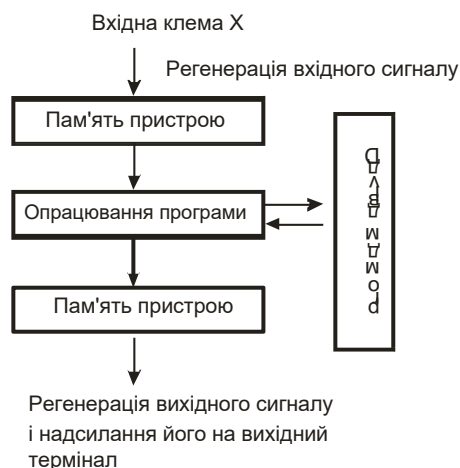
пристрій	функція	Діапазон приладу	Закрита зона
X	Вхідне реле	X0–X63	Всі прилади беззапірні.
Y	Вихідне реле	Y0–Y63	Всі прилади беззапірні.
M* 1	Допоміжне реле	M0–M8191	Стандартний діапазон: M6000–M8191.
SM	Спеціальне допоміжне реле	SM0–SM2047	Деякі пристрої заблоковані, і їх неможливо змінити. Для отримання додаткової інформації зверніться до списку спеціальних допоміжних реле.
S* 1	реле	S0–S1023	Стандартний діапазон – S512–S1023
T	Таймер	T0–T511	Всі прилади беззапірні.
C* 1	Лічильник	C0–C511	Діапазон за замовч. C448–C511
HC* 1	32-розрядний лічильник	HC0–HC255	Стандартний діапазон: HC128–HC255
D* 1	Регістр даних	D0–D29999	Стандартний діапазон: D20000–D23999
		W0–W29999	* 2
FR	Регістр файлів	FR0–FR65535	Всі прилади закриваються.
SR	Регістр спеціальних даних	SR0–SR2047	Деякі заблоковані, і їх неможливо змінити. Для отримання додаткової інформації зверніться до списку спеціальних реєстрів даних.
E	Індексний реєстр	E0–E9	Всі прилади беззапірні.
		E10–E14	* 2

*1: Щоб отримати додаткові відомості про налаштування заблокованої області, перегляньте HWCONFIG в ISPSOft. Налаштування фіксованої області означає, що інші області розглядаються як незафіксовані області. Діапазон зафіксованих областей не може перевищувати діапазон пристрою. Наприклад, встановлення M600–M7000 як зафіксованих областей робить зони M0–M5999 і M7001–M8191 незафіксованими.

*2: Використовується лише для редагування в ISPSOft.

2.2. Функції пристрою

Наступна блок-схема показує процедуру обробки програми в ПЛК.



- Регенерація вхідного сигналу
 1. Перед виконанням програми стан зовнішнього вхідного сигналу зчитується в область пам'яті для вхідного сигналу.
 2. Коли програма виконується, стан в області пам'яті для вхідного сигналу не змінюється, навіть якщо вхідний сигнал змінюється з UBIMK. на OFF. або з OFF. на UBIMK. Вхідний сигнал не оновлюється, доки не почнеться наступне сканування.
- Опрацювання програми

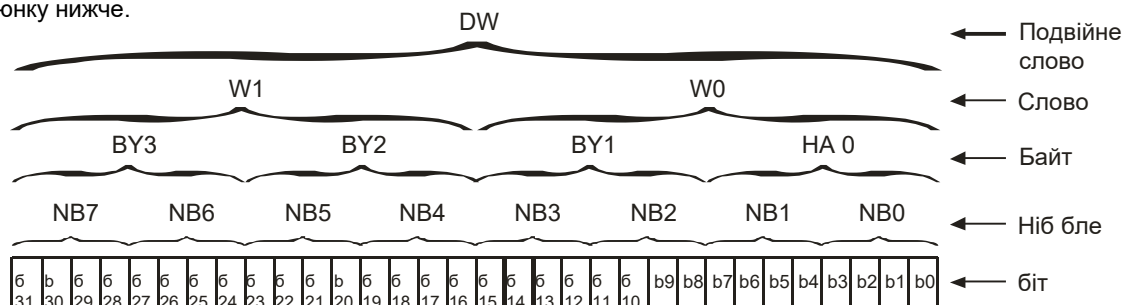
Після оновлення вхідного сигналу інструкції в програмі виконуються ОК від початкової адреси програми. Результати зберігаються в пам'яті приладу.
- Відновлення стану виходу

Після виконання інструкції END стан у пам'яті пристрою надсилається на вказаний вихідний термінал.

2.2.1 Значення та

Ім'я	опис
біт	Біт є основною одиницею в двійковій системі. Його стан дорівнює 1 або 0.
Відкушувати	Ніббл складається з чотирьох послідовних бітів (наприклад b3 – b0). Ніббли можуть представляти 0–9 у десятичній системі чи 0 – F у шістнадцятковій системі.
Байт	Байт складається з двох послідовних бітів (8 біт, b7 – b0). Байти можуть представляти 00 – FF у шістнадцятковій системі.
Слово	Слово складається з двох послідовних байтів (16 біт, b15 – b0). Слова можуть представляти 0000 – FFFF у шістнадцятковій системі.
Подвійне слово	Подвійне слово складається з двох послідовних слів (тобто 32 біти, b31 – b0). Подвійні слова представляють 00000000–FFFFFFFF у шістнадцятковій системі числення.

Зв'язок між бітами, байтами, словами та подвійними словами в двійковій системі показано на малюнку нижче.



ПЛК використовує чотири типи значень для виконання операції відповідно до різних цілей керування.

1. Двійкове число (BIN)

ПЛК використовує двійкову систему для роботи зі значеннями.

2. Десяткове число (DEC)

ПЛК використовує десяткові числа для:

- Значення налаштування таймера (Т) або значення налаштування лічильника (С/НС); наприклад, TMR C0 50 (**постійна К**).
- номер пристрою; наприклад, M 10 і T 30 (номер пристрою)
- Число до або після коми; наприклад, X0.0, Y0.11 і D10.0 (номер пристрою).
- **Константа К** , яка використовується як операнд у прикладній інструкції. Наприклад, MOV 123 D0 (**константа К**) .

3. Двійкове десяткове число (BCD)

Десяткове значення, представлене півбайтом або чотирма бітами, так що шістнадцять послідовних бітів представляють чотиризначне десяткове значення.

4. Шістнадцяткове число (HEX)

ПЛК використовує шістнадцяткові числа для:

- **Константа 16#** , яка використовується як операнд у прикладній інструкції; наприклад, MOV 16#1A2B D0 (шістнадцяткова константа).

У наступній таблиці наведено відповідні значення.

Двійкове число (BIN)	Десяткове число (DEC)	Двійковий десятковий код (BCD)	Шістнадцяткове число (HEX)
Внутрішнє виконання ПЛК	Константа К , номер пристрою	Інструкція щодо BCD	Константа 16#, номер пристрою
0000	0	0000	0
0001	1	0001	1
0010	2	0010	2
0011	3	0011	3
0100	4	0100	4
0101	5	0101	5
0110	6	0110	6
0111	7	0111	7
1000	8	1000	8
1001	9	1001	9
1010	10	-	A
1011	11	-	B
1100	12	-	C
1101	13	-	D
1110	14	-	E
1111	15	-	F
10000	16	0001 0000	10
10001	17	0001 0001	11

приклад 2:

-23 представлено числом з плаваючою комою одинарної точності.

Для перетворення -23,0 у число з плаваючою комою використовуються ті самі дії, що й для перетворення 23,0 у число з плаваючою комою, за винятком того, що знаковий біт дорівнює 1.

$1\ 100\ 0001\ 1\ 011\ 1000\ 0000\ 0000\ 0000\ 0000_2 = C1B80000_{16}$

2.2.2.2 Десяткові числа з плаваючою комою

- Числа з плаваючою комою одинарної точності та числа з плаваючою комою подвійної точності можна перетворити на десяткові числа з плаваючою комою, щоб люди могли їх прочитати. Однак внутрішньо ПЛК використовує числа з плаваючою комою одинарної точності та числа з плаваючою комою подвійної точності.
- 32-розрядне десяткове число з плаваючою комою представлено двома послідовними регістрами. Константа зберігається в першому регістрі з меншим номером, а експонента зберігається в регістрі з більшим номером. Візьмемо, наприклад, (D1, D0).

Десяткове число з плаваючою комою = [Константа D0] * $10^{[\text{Показник D1}]}$

Основне число D0 = ± 1000 до ± 9999

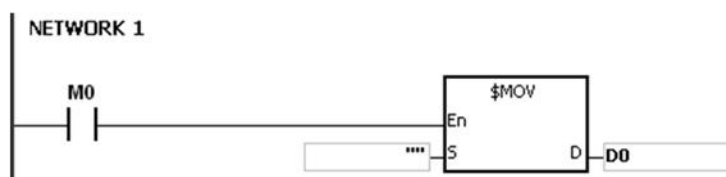
Експонента D1 = -41 до +35

Базове число 100 не існує в D0, оскільки 100 представлено як 1000×10^{-1} . 32-розрядні десяткові числа з плаваючою комою варіюються від $\pm 1175 \times 10^{-41}$ до $\pm 402 \times 10^{+35}$.

2.2.3 Рядки

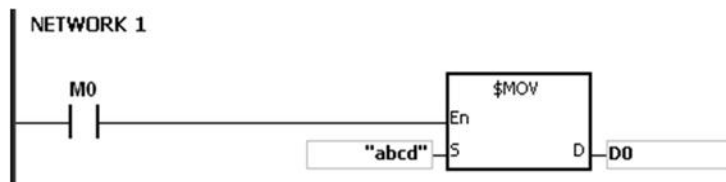
ПЛК може обробляти рядки, що складаються з кодів ASCII (*1). Повний рядок починається символом початку та закінчується символом закінчення (код NULL). Рядки можуть мати максимальну довжину 31 символ, і ISPSOft автоматично додає кінцевий символ (16#00).

1. Жоден рядок (код NULL) не переміщується.



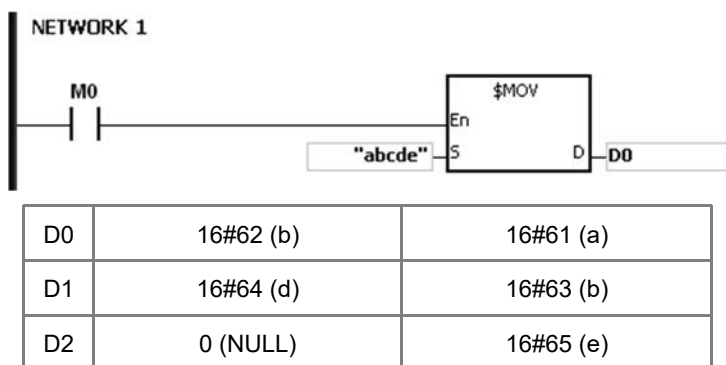
D0=0 (NULL)

2. Рядок містить парну кількість символів.



D0	16#62 (b)	16#61 (a)
D1	16#64 (d)	16#63 (b)
D2	0 (NULL)	

3. Рядок містить непарну кількість символів.



*1: Таблиця кодів ASCII

Hex	0	1	2	3	4	5	6	7	8	9	A	Б	C	D	E	F
ASCII	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒
Hex	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
ASCII	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒	☒
Hex	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
ASCII	SP	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
Hex	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
ASCII	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
Hex	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
ASCII	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Hex	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
ASCII	P	Q	R	S	T	U	V	W	X	Y	Z	☒	☒	☒	☒	☒
Hex	60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F
ASCII	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
Hex	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F
ASCII	p	q	r	s	t	u	v	w	x	y	z	{		}	-	☒

Примітка: ☒ представляє невидимий персонаж. Не використовуйте його в рядках.

2.2.4 Вхідні реле (X)

● Функція введення

Вхід підключається до пристрою введення (зовнішні пристрої, такі як кнопкові перемикачі, поворотні перемикачі та цифрові перемикачі), і ПЛК зчитує вхідний сигнал. Ви можете використовувати вхідний контакт А або контакт В кілька разів у програмі, і стан ON/OFF входу змінюється залежно від стану ON/OFF пристрою введення.

● Введіть число (десятькове число)

Для ПЛК вхідні номери починаються з X0.0. Кількість входів залежить від кількості входів на цифрових модулях вводу/виводу. Входи пронумеровані відповідно до порядку, в якому цифрові модулі вводу/виводу підключені до модуля CPU. Максимальна кількість входів для ПЛК становить 8192, а діапазон номерів входів – від X0.0 до X511.15.

● Тип введення

Вхідні дані поділяються на два типи.

1. Регенеровані входи: ПЛК зчитує стан регенованого входу перед виконанням програми; наприклад, LD X0.0.
2. Прямий вхід: стан прямого введення зчитується ПЛК під час виконання інструкцій; наприклад, LD DX0.0.

2.2.5 Вихідні реле (Y)

● Вихідна функція

Вихід надсилає сигнал ON/OFF для керування навантаженням, підключеним до виходу, таким як зовнішня сигнальна лампа, цифровий дисплей або електромагнітний клапан. Існує чотири типи виходів. Це реле, транзистори (NPN і PNP) і TRIAC (тиристори). Ви можете використовувати вихідний контакт А або контакт В кілька разів у програмі. Використовуйте вихід Y лише один раз у програмі; інакше, відповідно до функції програмного сканування ПЛК, стан виходу залежить від схеми, підключеної до останнього виходу Y у програмі.

● Вихідне число (десятькове число)

Для ПЛК вихідні номери починаються з Y0.0. Кількість виходів залежить від кількості виходів на цифрових модулях вводу/виводу. Виходи пронумеровані відповідно до порядку, в якому цифрові модулі вводу/виводу підключені до ПЛК. Максимальна кількість виходів на ПЛК становить 1024, а діапазон – від Y0.0 до Y63.15.

Вихід, який не використовується як пристрій виведення, можна використовувати як загальний пристрій.

● Типи виходу

Виходи класифікуються на два типи.

1. Відновлений вихід: стан відновленого виводу не записується, доки програма не виконає інструкцію END відповідно до станів виходів; наприклад, OUT Y0.0.
2. Прямий вихід: стан прямого виходу записується ПЛК під час виконання інструкцій відповідно до станів виходів; наприклад, OUT DY0.0.

2.2.6 Допоміжні реле (M)

Допоміжне реле має контакт А і контакт В. Його можна використовувати кілька разів у програмі. Ви можете об'єднати контури керування за допомогою допоміжного реле, але ви не можете керувати зовнішнім навантаженням за допомогою допоміжного реле. Ви можете використовувати допоміжні реле одним із двох наведених нижче способів.

- | | |
|---------------------------------|---|
| 1. Для загального користування: | У загальному випадку, якщо збій електроенергії виникає під час роботи ПЛК, допоміжне реле скидається у стан OFF. Коли живлення відновлюється, допоміжне реле залишається у стані OFF. |
| 2. Для замкнутого використання: | У замкнутому режимі, ff збій електроенергії, коли ПЛК працює, стан допоміжного реле зберігається. Коли живлення відновлюється, стан реле залишається таким же, як і до збою живлення. |

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM79	Очікування на отримання відповіді через функціональну картку 2	○	-	OFF	OFF	—	N	R	OFF
SM80	Прийом через функціональну картку 1 завершено.	○	-	OFF	OFF	—	N	R/W	OFF
SM81	Прийом через функціональну картку 2 завершено.	○	-	OFF	OFF	—	N	R/W	OFF
SM82	Під час отримання даних через функціональну картку 1 під час використання інструкції MODRW або інструкції RS виникає помилка.	○	-	OFF	OFF	—	N	R	OFF
SM83	Під час отримання даних через функціональну картку 2 під час використання інструкції MODRW або інструкції RS виникає помилка.	○	-	OFF	OFF	—	N	R	OFF
SM84	Немає даних через функціональну картку 1 після визначеного періоду часу.	○	-	OFF	OFF	—	N	R/W	OFF
SM85	Немає даних через функціональну картку 2 після визначеного періоду часу.	○	-	OFF	OFF	—	N	R/W	OFF
SM86	Вибір, зроблений функціональною платою 1 між 8-бітним режимом обробки та 16-бітним режимом обробки ON: 8-бітний режим обробки OFF.: 16-бітний режим обробки	○	-	OFF	—	—	N	R/W	OFF
SM87	Вибір, зроблений функціональною платою 2 між 8-бітним режимом обробки та 16-бітним режимом обробки ON: 8-бітний режим обробки OFF.: 16-бітний режим обробки	○	-	OFF	—	—	N	R/W	OFF
SM90	Змінюється протокол зв'язку функціональної карти 1	○	-	OFF	—	—	N	R/W	OFF
SM91	Змінюється протокол зв'язку функціональної карти 2	○	-	OFF	—	—	N	R/W	OFF
SM94	Зміна керування світлодіодним освітленням у COM1	○	○	—	—	—	H	R/W	OFF
SM95	Зміна керування світлодіодним освітленням у COM2	○	○	—	—	—	H	R/W	OFF
*SM96	Дані надсилаються через COM1.	○	○	OFF	OFF	—	N	R/W	OFF
*SM97	Дані надсилаються через COM2.	○	○	OFF	OFF	—	N	R/W	OFF
*SM98	Очікування на отримання відповіді через COM1	○	○	OFF	OFF	—	N	R	OFF
*SM99	Очікування відповіді через COM2	○	○	OFF	OFF	—	N	R	OFF
*SM100	Прийом через COM1 завершено.	○	○	OFF	OFF	—	N	R/W	OFF
*SM101	Прийом через COM2 завершено.	○	○	OFF	OFF	—	N	R/W	OFF
*SM102	Під час отримання даних через COM1 під час використання інструкції MODRW або інструкції RS виникає помилка.	○	○	OFF	OFF	—	N	R/W	OFF
*SM103	Під час отримання даних через COM2 під час використання інструкції MODRW або інструкції RS виникає помилка.	○	○	OFF	OFF	—	N	R/W	OFF
*SM104	Немає даних через COM1 через визначений період часу.	○	○	OFF	OFF	—	N	R/W	OFF
*SM105	Немає даних через COM2 через визначений період часу.	○	○	OFF	OFF	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SM106	Вибір, зроблений COM1 між 8-бітним режимом обробки та 16-бітним режимом обробки ON: 8-бітний режим обробки OFF.: 16-бітний режим обробки	○	○	OFF	–	–	N	R/W	OFF
*SM107	Вибір, зроблений COM2 між 8-бітним режимом обробки та 16-бітним режимом обробки ON: 8-бітний режим обробки OFF.: 16-бітний режим обробки	○	○	OFF	–	–	N	R/W	OFF
*SM120	I200 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM121	I201 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM122	I202 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM123	I203 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM124	I210 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM125	I211 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM126	I212 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM127	I213 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM128	I220 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM129	I221 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM130	I222 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM131	I223 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM132	I230 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM133	I231 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM134	I232 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM135	I233 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM136	I240 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM137	I241 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SM138	I242 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM139	I243 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM140	I250 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM141	I251 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM142	I252 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM143	I253 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM144	I260 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM145	I261 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM146	I262 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM147	I263 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM148	I264 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM149	I265 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM150	I266 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
*SM151	I267 високошвидкісне переривання порівняння для DPLSY для тимчасового збереження вихідних параметрів	○	○	OFF	OFF	–	N	R/W	OFF
SM166	VR0 запущено (працює з SR166)	○	○	OFF	–	–	N	R/W	OFF
SM167	VR1 запущено (працює з SR167)	○	○	OFF	–	–	N	R/W	OFF
SM168	З'єднання для функціональної карти 1 встановлено.	○	-	–	–	–	N	R	OFF
SM169	Функціональна карта 1 працює.	○	-	OFF	–	–	N	R	OFF
SM170	З'єднання для функціональної карти 2 встановлено.	○	-	–	–	–	N	R	OFF
SM171	Функціональна карта 12 працює.	○	-	OFF	–	–	N	R	OFF
SM172	Функціональна карта AS-FFTP01 готова.	○	-	OFF	–	–	N	R	OFF
*SM203	Коли PLC RUN->STOP очищає незаблоковану область один раз.	○	○	OFF	–	–	N	R/W	OFF
*SM204	Усі незамкнені області очищено.	○	○	OFF	OFF	OFF	N	R/W	OFF
*SM205	Усі замкнені області очищено.	○	○	OFF	OFF	OFF	N	R/W	OFF
SM206	Весь вихід блокується	○	○	OFF	–	–	N	R/W	OFF
*SM209	Змінюється протокол зв'язку COM1	○	○	OFF	OFF	OFF	N	R/W	OFF
*SM210	Вибір, зроблений COM1 між режимом ASCII і режимом RTU	○	○	–	–	–	H	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
	ON: режим RTU OFF.: режим ASCII								
*SM211	Змінюється протокол зв'язку COM1	○	○	OFF	OFF	OFF	N	R/W	OFF
*SM212	Вибір, зроблений COM2 між режимом ASCII і режимом RTU ON: режим RTU OFF: режим ASCII	○	○	–	–	–	H	R/W	OFF
SM215	Робочий стан ПЛК	○	○	OFF	ON	OFF	N	R/W	OFF
SM217	Немає відповіді від RTC	○	○	OFF	–	–	N	R	OFF
SM218	Помилка: несправність годинника реального часу	○	○	OFF	–	–	N	R	OFF
SM219	Помилка: заряд батареї годинника реального часу низький	○	○	OFF	–	–	N	R	OFF
*SM220	Калібрування годинника реального часу в межах ± 30 секунд	○	○	OFF	OFF	–	N	R/W	OFF
SM221	Увімкнути літній час (DST)	○	○	–	–	–	H	R	OFF
SM222	Відкалібруйте системний годинник за допомогою NTP (ON: успішне калібрування; OFF: невдале калібрування або без калібрування)	○	○	OFF	–	–	N	R	OFF
*SM230	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 1-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM231	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 2-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM232	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 3-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM233	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 4-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM234	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 5-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM235	Вибравши «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 6-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM236	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 7-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM237	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 8-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM238	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 9-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM239	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 10-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SM240	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 11- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM241	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 12- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM242	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 13- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM243	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 14- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM244	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 15- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM245	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 16- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM246	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 17- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM247	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 18- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM248	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 19- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM249	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 20- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM250	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 21- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM251	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 22- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM252	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 23- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM253	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 24- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM254	Якщо вибрати «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 25- й модуль праворуч від модуля ЦП.	○	○	—	—	—	Y	R/W	OFF
*SM255	При виборі «Manual + Flags» (Вручну + прапорці) у «I/O allocation setting when Power On» (Налаштування розподілу вводу/виводу під час увімкнення живлення) використовуйте цей прапорець, щоб активувати	○	○	—	—	—	Y	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
	26-й модуль праворуч від модуля CPU.								
*SM256	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 27-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM257	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 28-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM258	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 29-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM259	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 30-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM260	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу при включенні», використовуйте цей прапорець, щоб активувати 31-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
*SM261	Вибираючи «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу під час увімкнення», використовуйте цей прапорець, щоб активувати 32-й модуль праворуч від модуля ЦП.	○	○	–	–	–	Y	R/W	OFF
SM270	Зміна напрямку введення для прапора MPG 1 (X0.0/X0.1)	○	○	OFF	OFF	–	N	R/W	OFF
SM271	Зміна напрямку введення для прапора MPG 2 (X0.2/X0.3)	○	○	OFF	OFF	–	N	R/W	OFF
SM272	Зміна напрямку введення для прапора MPG 3 (X0.4/X0.5)	○	○	OFF	OFF	–	N	R/W	OFF
SM273	Зміна напрямку введення для прапора MPG 4 (X0.6/X0.7)	○	○	OFF	OFF	–	N	R/W	OFF
SM274	Зміна напрямку введення для прапора MPG 5 (X0.8/X0.9)	○	○	OFF	OFF	–	N	R/W	OFF
SM275	Зміна напрямку введення для прапора MPG 6 (X0.10/X0.11)	○	○	OFF	OFF	–	N	R/W	OFF
SM281	Зміна напрямку введення для прапора високошвидкісного лічильника 1	○	○	OFF	OFF	–	N	R/W	OFF
SM282	Зміна напрямку введення для прапора високошвидкісного лічильника 2	○	○	OFF	OFF	–	N	R/W	OFF
SM283	Зміна напрямку введення для прапора високошвидкісного лічильника 3	○	○	OFF	OFF	–	N	R/W	OFF
SM284	Зміна напрямку введення для прапора високошвидкісного лічильника 4	○	○	OFF	OFF	–	N	R/W	OFF
SM285	Зміна напрямку введення для високошвидкісного лічильника 5 прапор	○	○	OFF	OFF	–	N	R/W	OFF
SM286	Зміна напрямку введення для прапора високошвидкісного лічильника 6	○	○	OFF	OFF	–	N	R/W	OFF
SM287	Зміна напрямку введення для прапора високошвидкісного лічильника 7	○	○	OFF	OFF	–	N	R/W	OFF
SM288	Зміна напрямку введення для високошвидкісного лічильника	○	○	OFF	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↕ ON	STOP ↕ RUN	RUN ↕ STOP	Засув	Атрибут	За замовч.
	8 прапор								
SM291	Очищення вхідної точки для прапора швидкісного лічильника 1	○	○	OFF	OFF	–	N	R/W	OFF
SM292	Очистка точки введення для прапора швидкісного лічильника 2	○	○	OFF	OFF	–	N	R/W	OFF
SM293	Очищення прапора точки введення швидкодіючого лічильника 3	○	○	OFF	OFF	–	N	R/W	OFF
SM294	Очистка точки введення для прапора швидкісного лічильника 4	○	○	OFF	OFF	–	N	R/W	OFF
SM295	Скидання прапора точки введення швидкодіючого лічильника 5	○	○	OFF	OFF	–	N	R/W	OFF
SM296	Очистка точки введення для прапора швидкісного лічильника 6	○	○	OFF	OFF	–	N	R/W	OFF
SM300	Налаштування режиму підрахунку для HC200. HC200 веде зворотний відлік, коли SM300 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM301	Налаштування режиму підрахунку для HC201. HC201 веде зворотний відлік, коли SM301 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM302	Налаштування режиму підрахунку для HC202. HC202 веде зворотний відлік, коли SM302 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM303	Налаштування режиму підрахунку для HC203. HC203 веде зворотний відлік, коли SM303 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM304	Налаштування режиму підрахунку для HC204. HC204 веде зворотний відлік, коли SM304 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM305	Налаштування режиму підрахунку для HC205. HC205 веде зворотний відлік, коли SM305 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM306	Налаштування режиму підрахунку для HC206. HC206 веде зворотний відлік, коли SM306 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM307	Налаштування режиму підрахунку для HC207. HC207 веде зворотний відлік, коли SM307 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM308	Налаштування режиму підрахунку для HC208. HC208 веде зворотний відлік, коли SM308 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM309	Налаштування режиму підрахунку для HC209. HC209 веде зворотний відлік, коли SM309 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM310	Налаштування режиму підрахунку для HC210. HC210 веде зворотний відлік, коли SM310 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM311	Налаштування режиму підрахунку для HC211. HC211 веде зворотний відлік, коли SM311 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM312	Налаштування режиму підрахунку для HC212. HC212 веде зворотний відлік, коли SM312 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM313	Налаштування режиму підрахунку для HC213. HC213 веде зворотний відлік, коли SM313 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM314	Налаштування режиму підрахунку для HC214. HC214 веде зворотний відлік, коли SM314 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM315	Налаштування режиму підрахунку для HC215. HC215 веде зворотний відлік, коли SM315 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM316	Налаштування режиму підрахунку для HC216. HC216 веде зворотний відлік, коли SM316 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
SM317	Налаштування режиму підрахунку для HC217. HC217 веде зворотний відлік, коли SM317 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM318	Налаштування режиму підрахунку для HC218. HC218 веде зворотний відлік, коли SM318 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM319	Налаштування режиму підрахунку для HC219. HC219 веде зворотний відлік, коли SM319 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM320	Налаштування режиму підрахунку для HC220. HC220 веде зворотний відлік, коли SM320 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM321	Налаштування режиму підрахунку для HC221. HC221 веде зворотний відлік, коли SM321 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM322	Налаштування режиму підрахунку для HC222. HC222 веде зворотний відлік, коли SM322 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM323	Налаштування режиму підрахунку для HC223. HC223 веде зворотний відлік, коли SM323 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM332	Налаштування режиму підрахунку для HC232. HC232 веде зворотний відлік, коли SM332 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM333	Налаштування режиму підрахунку для HC233. HC233 веде зворотний відлік, коли SM333 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM334	Налаштування режиму підрахунку для HC234. HC234 веде зворотний відлік, коли SM334 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM335	Налаштування режиму підрахунку для HC235. HC235 веде зворотний відлік, коли SM335 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM336	Налаштування режиму підрахунку для HC236. HC236 веде зворотний відлік, коли SM336 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM337	Налаштування режиму підрахунку для HC237. HC237 веде зворотний відлік, коли SM337 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM338	Налаштування режиму підрахунку для HC238. HC238 веде зворотний відлік, коли SM338 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM339	Налаштування режиму підрахунку для HC239. HC239 веде зворотний відлік, коли SM339 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM340	Налаштування режиму підрахунку для HC240. HC240 веде зворотний відлік, коли SM340 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM341	Налаштування режиму підрахунку для HC241. HC241 веде зворотний відлік, коли SM341 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM342	Налаштування режиму підрахунку для HC242. HC242 веде зворотний відлік, коли SM342 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM343	Налаштування режиму підрахунку для HC243. HC243 веде зворотний відлік, коли SM343 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM344	Налаштування режиму підрахунку для HC244. HC244 веде зворотний відлік, коли SM344 увімкнено.	○	○	OFF	OFF	–	N	R/W	OFF
SM345	Налаштування режиму підрахунку для HC245. HC245 веде зворотний відлік, коли SM345 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM346	Налаштування режиму підрахунку для HC246.	○	○	OFF	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
	HC246 веде зворотний відлік, коли SM346 увімкнено.								
SM347	Налаштування режиму підрахунку для HC247. HC247 веде зворотний відлік, коли SM347 увімкнено.	○	○	OFF	—	—	N	R	OFF
SM348	Налаштування режиму підрахунку для HC248. HC248 веде зворотний відлік, коли SM348 увімкнено.	○	○	OFF	OFF	—	N	R/W	OFF
SM349	Налаштування режиму підрахунку для HC249. HC249 веде зворотний відлік, коли SM349 увімкнено.	○	○	OFF	—	—	N	R	OFF
SM350	Налаштування режиму підрахунку для HC250. HC250 веде зворотний відлік, коли SM350 увімкнено.	○	○	OFF	OFF	—	N	R/W	OFF
SM351	Налаштування режиму підрахунку для HC251. HC251 веде зворотний відлік, коли SM351 увімкнено.	○	○	OFF	—	—	N	R	OFF
SM352	Налаштування режиму підрахунку для HC252. HC252 веде зворотний відлік, коли SM352 увімкнено.	○	○	OFF	OFF	—	N	R/W	OFF
SM353	Налаштування режиму підрахунку для HC253. HC253 веде зворотний відлік, коли SM353 увімкнено.	○	○	OFF	OFF	—	N	R/W	OFF
*SM360	I200 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM361	I201 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM362	I202 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM363	I203 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM364	I210 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM365	I211 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM366	I212 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM367	I213 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM368	I220 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM369	I221 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM370	I222 високошвидкісне переривання порівняння для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM371	I223 високошвидкісне переривання порівняння для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM372	I230 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF
*SM373	I231 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SM374	I232 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM375	I233 високошвидкісне переривання порівняння для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM376	I240 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM377	I241 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM378	I242 високошвидкісне переривання порівняння для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM379	I243 високошвидкісне переривання порівняння для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM380	I250 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM381	I251 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM382	I252 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM383	I253 високошвидкісне переривання порівняння для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM384	Високошвидкісне порівняльне переривання I260 для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM385	I261 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM386	I262 високошвидкісне переривання порівняння для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM387	I263 високошвидкісне переривання порівняння для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM388	I264 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM389	I265 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM390	I266 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM391	I267 високошвидкісне порівняльне переривання для DPLSY для активації виходу	○	○	OFF	OFF	–	N	R/W	OFF
*SM400	Прапор завжди увімкнено (нормально відкритий), коли ЦП працює.	○	○	OFF	ON	OFF	N	R	OFF
*SM401	Прапорець завжди OFF (нормально закритий), коли ЦП працює.	○	○	OFF	OFF	ON	N	R	OFF
*SM402	Прапорець увімкнено лише під час першого сканування.	○	○	OFF	ON	OFF	N	R	OFF
*SM403	Прапор OFF лише під час першого сканування.	○	○	OFF	OFF	ON	N	R	OFF
*SM404	Тактовий імпульс тривалістю 10 мілісекунд, під час якого імпульс увімкнено протягом 5 мілісекунд, а потім вимкнено протягом 5	○	○	OFF	–	–	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
	мілісекунд								
*SM405	Тактовий імпульс тривалістю 100 мілісекунд, під час якого імпульс увімкнено протягом 50 мілісекунд, а потім вимкнено протягом 50 мілісекунд	○	○	OFF	—	—	N	R	OFF
*SM406	Тактовий імпульс тривалістю 200 мілісекунд, під час якого імпульс увімкнено на 100 мілісекунд, а потім вимкнено на 100 мілісекунд	○	○	OFF	—	—	N	R	OFF
*SM407	Односекундний тактовий імпульс, протягом якого імпульс увімкнено протягом 500 мілісекунд, а потім вимкнено протягом 500 мілісекунд	○	○	OFF	—	—	N	R	OFF
*SM450	Карта пам'яті є ON: карта пам'яті є. OFF: карта пам'яті відсутня.	○	○	—	—	—	N	R	OFF
*SM452	Виконується доступ до даних на карті пам'яті. ON: здійснюється доступ до даних на карті пам'яті. OFF.: доступ до даних на карті пам'яті недоступний.	○	○	OFF	—	—	N	R	OFF
*SM453	Під час роботи карти пам'яті виникає помилка. ON: сталася помилка. OFF.: немає помилок.	○	○	OFF	—	—	N	R	OFF
SM454	Увімкнення/вимкнення реєстратора даних. ON: увімкнуті OFF.: вимкнено	○	○	OFF	—	—	N	R/W	OFF
SM455	Наразі реєстратор відбирає проби. ON: кількість записів досягла ліміту вибірки	○	○	OFF	—	—	N	R	OFF
*SM456	Виконання реєстратора даних і карти пам'яті. ON: виконання за значеннями в SR902.	○	○	OFF	—	—	N	R/W	OFF
SM457	Стан параметрів вибірки в реєстраторі даних увімкнено: параметр вибірки встановлено.	○	○	—	—	—	N	R	OFF
SM460	Виведення Y0.0/вісь 1 (Y0.0/Y0.1).	○	○	OFF	OFF	—	N	R	OFF
SM461	Виведення Y0.0/вісь 1 (Y0.0/Y0.1) завершено.	○	○	OFF	OFF	—	N	R/W	OFF
SM462	Змінити вихідний напрямок осі 1 (Y0.1)	○	○	OFF	OFF	—	N	R/W	OFF
*SM463	Припинення виведення Y0.0/вісь 1 (Y0.0/Y0.1)	○	○	OFF	OFF	—	N	R/W	OFF
SM464	Увімкнення апаратного позитивного обмеження на осі 1 (Y0.0/Y0.1)	○	○	—	—	—	Y	R/W	OFF
SM465	Сигналізація: перевищення позитивного ліміту на осі 1 (Y0.0/Y0.1) або під час повернення на нуль і досягнуто позитивного ліміту.	○	○	OFF	OFF	—	N	R/W	OFF
SM466	Увімкнення апаратного негативного обмеження на осі 1 (Y0.0/Y0.1)	○	○	—	—	—	Y	R/W	OFF
SM467	Сигнал тривоги: перевищення від'ємного ліміту на осі 1 (Y0.0/Y0.1) або під час повернення на нуль і досягнення від'ємного ліміту.	○	○	OFF	OFF	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM469	Увімкнення підйому/зниження фіксованого нахилу на осі 1 (Y0.0/Y0.1)	○	○	OFF	OFF	–	N	R/W	OFF
SM470	Автоматичне скидання після завершення виведення Y0.0/вісь 1 (Y0.0/Y0.1).	○	○	OFF	OFF	–	N	R/W	OFF
SM471	Виконання переривання I500, коли імпульсний вихід закінчується на осі 1 (Y0.0/Y0.1)	○	○	OFF	OFF	–	N	R/W	OFF
SM472	Виведення Y0.1.	○	○	OFF	OFF	–	N	R	OFF
SM473	Виведення Y0.1 завершено.	○	○	OFF	OFF	–	N	R/W	OFF
*SM474	Зупинка виведення Y0.1.	○	○	OFF	OFF	–	N	R/W	OFF
SM475	Автоматичне скидання після завершення виведення Y0.1.	○	○	OFF	OFF	–	N	R/W	OFF
*SM476	Вивід негайно припиняється, коли інструкцію вимкнено або зупиняється для Y0.0/осі 1 (Y0.0/Y0.1)	○	○	OFF	OFF	–	N	R/W	OFF
*SM477	Вивід негайно припиняється, коли вказівка вимкнена, або зупиняється для Y0.1	○	○	OFF	OFF	–	N	R/W	OFF
SM478	Змініть цільову позицію під час виведення Y0.0/вісь 1 (Y0.0/Y0.1)	○	○	OFF	OFF	–	N	R/W	OFF
SM479	Змініть цільову позицію під час виведення Y0.1	○	○	OFF	OFF	–	N	R/W	OFF
SM480	Виведення Y0.2/вісь 2 (Y0.2/Y0.3).	○	○	OFF	OFF	–	N	R	OFF
SM481	Y0.2/вісь 2 (Y0.2/Y0.3) вихід завершено.	○	○	OFF	OFF	–	N	R/W	OFF
SM482	Зміна напрямку вихідного сигналу осі 2 (Y0.3)	○	○	OFF	OFF	–	N	R/W	OFF
*SM483	Припинення виведення Y0.2/вісь 2 (Y0.2/Y0.3)	○	○	OFF	OFF	–	N	R/W	OFF
SM484	Увімкнення апаратного позитивного обмеження на осі 2 (Y0.2/Y0.3)	○	○	–	–	–	Y	R/W	OFF
SM485	Сигналізація: перевищення позитивного ліміту на осі 2 (Y0.2/Y0.3) або під час повернення на нуль і досягнуто позитивного ліміту.	○	○	OFF	OFF	–	N	R/W	OFF
SM486	Увімкнення апаратного негативного обмеження на осі 2 (Y0.2/Y0.3)	○	○	–	–	–	Y	R/W	OFF
SM487	Сигналізація: перевищення від'ємного ліміту на осі 2 (Y0.2/Y0.3) або під час виконання повернення на нуль і досягнення від'ємного ліміту.	○	○	OFF	OFF	–	N	R/W	OFF
SM489	Увімкнення підйому/зниження фіксованого нахилу на осі 2 (Y0.2/Y0.3).	○	○	OFF	OFF	–	N	R/W	OFF
SM490	Автоматичне скидання після завершення виведення Y0.2/вісь 2 (Y0.2/Y0.3).	○	○	OFF	OFF	–	N	R/W	OFF
SM491	Виконання переривання I501, коли імпульсний вихід закінчується на осі 2 (Y0.2/Y0.3).	○	○	OFF	OFF	–	N	R/W	OFF
SM492	Виведення Y0.3.	○	○	OFF	OFF	–	N	R	OFF
SM493	Виведення Y0.3 завершено.	○	○	OFF	OFF	–	N	R/W	OFF
*SM494	Зупинка виведення Y0.3.	○	○	OFF	OFF	–	N	R/W	OFF
SM495	Автоматичне скидання після завершення виведення Y0.3.	○	○	OFF	OFF	–	N	R/W	OFF
*SM496	Вивід негайно припиняється, коли інструкцію вимкнено, або зупиняється для Y0.2/осі 2 (Y0.2/Y0.3).	○	○	OFF	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM497	Вивід негайно припиняється, коли інструкція вимкнена, або припиняється для Y0.3.	○	○	OFF	OFF	–	N	R/W	OFF
SM498	Змініть цільову позицію під час виведення Y0.2/вісь 2 (Y0.2/Y0.3)	○	○	OFF	OFF	–	N	R/W	OFF
SM499	Змініть цільову позицію під час виведення Y0.3	○	○	OFF	OFF	–	N	R/W	OFF
SM500	Виведення Y0.4/вісь 3 (Y0.4/Y0.5).	○	○	OFF	OFF	–	N	R	OFF
SM501	Виведення Y0.4/вісь 3 (Y0.4/Y0.5) завершено.	○	○	OFF	OFF	–	N	R/W	OFF
SM502	Зміна напрямку вихідного сигналу осі 3 (Y0.5).	○	○	OFF	OFF	–	N	R/W	OFF
*SM503	Припинення виведення Y0.4/вісь 3 (Y0.4/Y0.5).	○	○	OFF	OFF	–	N	R/W	OFF
SM504	Увімкнення апаратного позитивного обмеження на осі 3 (Y0.4/Y0.5)	○	○	–	–	–	Y	R/W	OFF
SM505	Сигналізація: перевищення позитивної межі на осі 3 (Y0.4/Y0.5) або під час повернення на нуль і досягнуто позитивної межі.	○	○	OFF	OFF	–	N	R/W	OFF
SM506	Увімкнення апаратного негативного обмеження на осі 3 (Y0.4/Y0.5)	○	○	–	–	–	Y	R/W	OFF
SM507	Сигнал тривоги: перевищення негативного ліміту на осі 3 (Y0.4/Y0.5) або під час виконання повернення на нуль і негативне ліміт досягнуто.	○	○	OFF	OFF	–	N	R/W	OFF
SM509	Увімкнення підвищення/зниження фіксованого нахилу на осі 3 (Y0.4/Y0.5)	○	○	OFF	OFF	–	N	R/W	OFF
SM510	Автоматичне скидання після завершення виведення Y0.4/вісь 3 (Y0.4/Y0.5).	○	○	OFF	OFF	–	N	R/W	OFF
SM511	Виконання переривання I502, коли вихід імпульсу закінчується для осі 3 (Y0.4/Y0.5).	○	○	OFF	OFF	–	N	R/W	OFF
SM512	Виведення Y0.5.	○	○	OFF	OFF	–	N	R	OFF
SM513	Виведення Y0.5 завершено.	○	○	OFF	OFF	–	N	R/W	OFF
*SM514	Припинення виведення Y0.5.	○	○	OFF	OFF	–	N	R/W	OFF
SM515	Автоматичне скидання після завершення виведення Y0.5.	○	○	OFF	OFF	–	N	R/W	OFF
*SM516	Вивід негайно припиняється, коли інструкцію вимкнено або зупиняється на Y0.4/осі 3 (Y0.4/Y0.5).	○	○	OFF	OFF	–	N	R/W	OFF
*SM517	Вивід негайно припиняється, коли вказівку вимкнено, або зупиняється на Y0.5.	○	○	OFF	OFF	–	N	R/W	OFF
SM518	Змініть цільову позицію під час виведення Y0.4/вісь 3 (Y0.4/Y0.5)	○	○	OFF	OFF	–	N	R/W	OFF
SM519	Змініть цільову позицію під час виведення Y0.5	○	○	OFF	OFF	–	N	R/W	OFF
SM520	Виведення Y0.6/вісь 4 (Y0.6/Y0.7).	○	○	OFF	OFF	–	N	R	OFF
SM521	Y0.6/вісь 4 (Y0.6/Y0.7) вихід завершено.	○	○	OFF	OFF	–	N	R/W	OFF
SM522	Зміна напрямку вихідного сигналу осі 4 (Y0.7) .	○	○	OFF	OFF	–	N	R/W	OFF
*SM523	Припинення виведення Y0.6/вісь 4 (Y0.6/Y0.7).	○	○	OFF	OFF	–	N	R/W	OFF
SM524	Увімкнення апаратного позитивного обмеження на осі 4 (Y0.6/Y0.7).	○	○	–	–	–	Y	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM525	Сигналізація: перевищення позитивної межі на осі 4 (Y0.6/Y0.7) або під час повернення на нуль і досягнуто позитивної межі.	○	○	OFF	OFF	–	N	R/W	OFF
SM526	Увімкнення апаратного негативного обмеження на осі 4 (Y0.6/Y0.7).	○	○	–	–	–	Y	R/W	OFF
SM527	Сигналізація: перевищення негативного ліміту на осі 4 (Y0.6/Y0.7) або під час виконання повернення на нуль і негативне ліміт досягнуто.	○	○	OFF	OFF	–	N	R/W	OFF
SM529	Увімкнення фіксованого нахилу вгору/вниз на осі 4 (Y0.6/Y0.7).	○	○	OFF	OFF	–	N	R/W	OFF
SM530	Автоматичне скидання після завершення виведення Y0.6/вісь 4 (Y0.6/Y0.7).	○	○	OFF	OFF	–	N	R/W	OFF
SM531	Виконання переривання I503, коли імпульсний вихід закінчується на осі 4 (Y0.6/Y0.7).	○	○	OFF	OFF	–	N	R/W	OFF
SM532	Виведення Y0.7.	○	○	OFF	OFF	–	N	R	OFF
SM533	Виведення Y0.7 завершено.	○	○	OFF	OFF	–	N	R/W	OFF
*SM534	Зупинка виведення Y0.7.	○	○	OFF	OFF	–	N	R/W	OFF
SM535	Автоматичне скидання після завершення виведення Y0.7.	○	○	OFF	OFF	–	N	R/W	OFF
*SM536	Вивід негайно припиняється, коли інструкцію вимкнено, або зупиняється для Y0.6/осі 4 (Y0.6/Y0.7).	○	○	OFF	OFF	–	N	R/W	OFF
*SM537	Вивід негайно припиняється, коли інструкція вимкнена, або зупиняється для Y0.7.	○	○	OFF	OFF	–	N	R/W	OFF
SM538	Змініть цільову позицію під час виведення Y0.6/вісь 4 (Y0.6/Y0.7)	○	○	OFF	OFF	–	N	R/W	OFF
SM539	Змініть цільову позицію під час виведення Y0.7	○	○	OFF	OFF	–	N	R/W	OFF
SM540	Виведення Y0.8/вісь 5 (Y0.8/Y0.9).	○	○	OFF	OFF	–	N	R	OFF
SM541	Виведення Y0.8/вісь 5 (Y0.8/Y0.9) завершено.	○	○	OFF	OFF	–	N	R/W	OFF
SM542	Зміна напрямку вихідного сигналу осі 5 (Y0.9).	○	○	OFF	OFF	–	N	R/W	OFF
*SM543	Припинення виведення Y0.8/осі 5 (Y0.8/Y0.9).	○	○	OFF	OFF	–	N	R/W	OFF
SM544	Увімкнення апаратного позитивного обмеження на осі 5 (Y0.8/Y0.9).	○	○	–	–	–	Y	R/W	OFF
SM545	Сигналізація: перевищення позитивної межі на осі 5 (Y0.8/Y0.9) або під час повернення на нуль і досягнуто позитивної межі.	○	○	OFF	OFF	–	N	R/W	OFF
SM546	Увімкнення апаратного негативного обмеження на осі 5 (Y0.8/Y0.9).	○	○	–	–	–	Y	R/W	OFF
SM547	Сигналізація: перевищення негативної межі на осі 5 (Y0.8/Y0.9) або під час повернення на нуль і досягнуто позитивної межі.	○	○	OFF	OFF	–	N	R/W	OFF
SM549	Увімкнення фіксованого нахилу вгору/вниз на осі 5 (Y0.8/Y0.9).	○	○	OFF	OFF	–	N	R/W	OFF
SM550	Автоматичне скидання, коли Y0.8/вісь 5 (вихід Y0.8/Y0.9) завершено.	○	○	OFF	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SM551	Виконання переривання I504, коли імпульсний вихід закінчується на осі 5 (Y0.8/Y0.9).	○	○	OFF	OFF	–	N	R/W	OFF
SM552	Виведення Y0.9.	○	○	OFF	OFF	–	N	R	OFF
SM553	Виведення Y0.9 завершено.	○	○	OFF	OFF	–	N	R/W	OFF
*SM554	Припинення виведення Y0.9.	○	○	OFF	OFF	–	N	R/W	OFF
SM555	Автоматичне скидання після завершення виведення Y0.9.	○	○	OFF	OFF	–	N	R/W	OFF
*SM556	Вивід негайно припиняється, коли інструкцію вимкнено, або зупиняється для Y0.8/осі 5 (Y0.8/Y0.9).	○	○	OFF	OFF	–	N	R/W	OFF
*SM557	Вивід негайно припиняється, коли інструкція вимкнена, або припиняється для Y0.9.	○	○	OFF	OFF	–	N	R/W	OFF
SM558	Змініть цільову позицію під час виведення Y0.8/вісь 5 (Y0.8/Y0.9)	○	○	OFF	OFF	–	N	R/W	OFF
SM559	Змініть цільову позицію під час виведення Y0.9	○	○	OFF	OFF	–	N	R/W	OFF
SM560	Виведення Y0.10/вісь 6 (Y0.10/Y0.11).	○	○	OFF	OFF	–	N	R	OFF
SM561	Виведення Y0.10/вісь 6 (Y0.10/Y0.11) завершено.	○	○	OFF	OFF	–	N	R/W	OFF
SM562	Зміна напрямку вихідного сигналу осі 6 (Y0.11).	○	○	OFF	OFF	–	N	R/W	OFF
*SM563	Припинення виведення Y0.10/вісь 6 (Y0.10/Y0.11).	○	○	OFF	OFF	–	N	R/W	OFF
SM564	Увімкнення апаратного позитивного обмеження на осі 6 (Y0.10/Y0.11).	○	○	–	–	–	Y	R/W	OFF
SM565	Сигналізація: перевищення позитивної межі на осі 6 (Y0.10/Y0.11) або під час повернення на нуль і досягнуто позитивної межі.	○	○	OFF	OFF	–	N	R/W	OFF
SM566	Увімкнення апаратного негативного обмеження на осі 6 (Y0.10/Y0.11).	○	○	–	–	–	Y	R/W	OFF
SM567	Сигнал тривоги: перевищення негативного ліміту на осі 6 (Y0.10/Y0.11) або під час виконання повернення на нуль і негативний ліміт досягнуто.	○	○	OFF	OFF	–	N	R/W	OFF
SM569	Увімкнення фіксованого нахилу вгору/вниз на осі 6 (Y0.10/Y0.11).	○	○	OFF	OFF	–	N	R/W	OFF
SM570	Автоматичне скидання після завершення виведення Y0.10/вісь 6 (Y0.10/Y0.11).	○	○	OFF	OFF	–	N	R/W	OFF
SM571	Виконання переривання I505, коли імпульсний вихід закінчується на осі 6 (Y0.10/Y0.11).	○	○	OFF	OFF	–	N	R/W	OFF
SM572	Виведення Y0.11.	○	○	OFF	OFF	–	N	R	OFF
SM573	Виведення Y0.11 завершено.	○	○	OFF	OFF	–	N	R/W	OFF
*SM574	Припинення виведення Y0.11.	○	○	OFF	OFF	–	N	R/W	OFF
SM575	Автоматичне скидання після завершення виведення Y0.11.	○	○	OFF	OFF	–	N	R/W	OFF
*SM576	Вивід негайно припиняється, коли інструкція вимкнена, або зупиняється для Y0.10/осі 6 (Y0.10/Y0.11).	○	○	OFF	OFF	–	N	R/W	OFF
*SM577	Вивід негайно припиняється, коли інструкцію вимикається, або зупиняється для Y0.11.	○	○	OFF	OFF	–	N	R/W	OFF
SM578	Змініть цільову позицію під час виведення Y0.10/вісь 6 (Y0.10/Y0.11)	○	○	OFF	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM579	Змініть цільову позицію під час виведення Y0.11	○	○	OFF	OFF	–	N	R/W	OFF
SM580	Усі виходи негайно припиняються, коли інструкція вимкнена або припиняється.	○	○	OFF	OFF	–	N	R/W	OFF
SM581	Ви можете змінити позначку часу прискорення/уповільнення під час виконання інструкції JOG.	○	○	OFF	OFF	–	N	R/W	OFF
SM585	Під час виконання DTPWS / DTPWL встановіть прапор SM585 на ON, якщо вам потрібно змінити час для прискорення або уповільнення. А потім змініть час прискорення та уповільнення, встановлений у SR, щоб змінити час прискорення в русі від точки до точки по одній осі та багатосегментному русі по одній осі або час уповільнення в русі від точки до точки по одній осі.	○	○	OFF	OFF	–	N	R/W	OFF
*SM586	Використовується для розрахунку інтервалу частоти прискорення та уповільнення (для осі 1-6)	○	○	OFF	OFF	–	N	R/W	OFF
*SM587	Коли інструкція TPO використовує таблицю планування позиції для керування виводом, якщо вивід зупинено під час виконання, вісь має посилатися на відповідні прапорці (наприклад, SM476, SM496--SM576), щоб зупинити дію.	○	○	OFF	OFF	–	N	R/W	OFF
SM591	Встановіть цей прапор на ON під час виконання інструкції TPO, якщо вам потрібно зупинити поточний вивід і почати новий для осі 1, ПЛК діятиме відповідно до налаштування в SR1291, щоб вивести наступний сегмент або певну кількість сегментів. .	○	○	OFF	OFF	–	N	R/W	OFF
SM592	Встановіть цей прапор на ON під час виконання інструкції TPO, якщо вам потрібно зупинити поточний вивід і почати новий для осі 2, ПЛК діятиме відповідно до налаштування в SR1292, щоб вивести наступний сегмент або певну кількість сегментів. .	○	○	OFF	OFF	–	N	R/W	OFF
SM593	Встановіть цей прапор на ON під час виконання інструкції TPO, якщо вам потрібно зупинити поточний вивід і почати новий для осі 3, ПЛК діятиме відповідно до налаштування в SR1293, щоб вивести наступний сегмент або певну кількість сегментів. .	○	○	OFF	OFF	–	N	R/W	OFF
SM594	Встановіть цей прапор на ON під час виконання інструкції TPO, якщо вам потрібно зупинити поточний вивід і почати новий для осі 4, ПЛК діятиме відповідно до налаштування в SR1294, щоб вивести наступний сегмент або певну кількість сегментів. .	○	○	OFF	OFF	–	N	R/W	OFF
SM595	Встановіть цей прапор на ON під час виконання інструкції TPO, якщо вам потрібно зупинити поточний вивід і почати новий для осі 5, ПЛК діятиме відповідно до налаштування в SR1295, щоб вивести наступний сегмент або певну кількість сегментів. .	○	○	OFF	OFF	–	N	R/W	OFF
SM596	Встановіть цей прапор на ON під час виконання інструкції TPO, якщо вам потрібно зупинити поточний вихід і почати новий для осі 6, ПЛК діятиме	○	○	OFF	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Зсув	Атрибут	За замовч.
	відповідно до налаштування в SR1296, щоб вивести наступний сегмент або певну кількість сегментів.								
SM600	Нульовий прапор	○	○	OFF	–	–	N	R	OFF
SM601	Позичити прапор	○	○	OFF	–	–	N	R	OFF
SM602	Носіть прапор	○	○	OFF	–	–	N	R	OFF
SM603	Повідомлення про помилку для того, щоб кількість завдань, які потрібно виконати, перевищила кількість завдань, якими робоча станція може керувати для інструкції XCMP.	○	○	OFF	OFF	–	N	R/W	OFF
SM604	Встановлення режиму роботи інструкції SORT. ON: у порядку спадання OFF.: у порядку зростання	○	○	OFF	–	–	N	R/W	OFF
SM605	Позначення режиму роботи інструкції SMOV.	○	○	OFF	–	–	N	R/W	OFF
SM606	8-бітний або 16-бітний режим роботи. ON: 8-біт OFF.: 16-біт	○	○	OFF	–	–	N	R/W	OFF
SM607	Прапор порівняння матриць. ON: Порівняння еквівалентних значень. OFF.: Порівняння різних значень.	○	○	OFF	–	–	N	R/W	OFF
SM608	Порівняння матриць завершено. Коли порівнюються останні біти, SM608 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM609	Коли SM609 увімкнено, порівняння починається з біта 0.	○	○	OFF	–	–	N	R	OFF
SM610	Прапор пошуку бітів матриці. Коли матричний пошук знаходить відповідні біти або завершує пошук, пошук негайно припиняється, і SM610 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM611	Прапор помилки покажчика матриці. Коли значення покажчика перевищує діапазон порівняння, SM611 увімкнено.	○	○	OFF	–	–	N	R	OFF
SM612	Прапор збільшення покажчика матриці. Поточне значення покажчика збільшується на одиницю.	○	○	OFF	–	–	N	R/W	OFF
SM613	Прапор очищення покажчика матриці. Поточне значення вказівника скидається на нуль.	○	○	OFF	–	–	N	R/W	OFF
SM614	Прапор переносу для повороту/зсуву/виведення матриці.	○	○	OFF	–	–	N	R	OFF
SM615	Позичити позначку для зсуву/виведення матриці.	○	○	OFF	–	–	N	R/W	OFF
SM616	Позначка напрямку обертання/зсуву матриці. OFF.: біти зсуваються вліво. ON: біти зсуваються праворуч.	○	○	OFF	–	–	N	R/W	OFF
SM617	Підраховуються біти зі значенням 0 або 1.	○	○	OFF	–	–	N	R/W	OFF
SM618	ON, коли результат підрахунку матриці дорівнює 0.	○	○	OFF	–	–	N	R/W	OFF
SM619	ON, коли виконується інструкція EI.	○	○	OFF	OFF	–	N	R	OFF
SM620	Якщо результатом порівняння за допомогою інструкції CMPT# є те, що всі пристрої увімкнено, SM620 увімкнено.	○	○	OFF	–	–	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM621	Налаштування режиму підрахунку для HC0. HC0 веде зворотній відлік, коли SM621 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM622	Налаштування режиму підрахунку для HC. HC1 веде зворотний відлік, коли SM622 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM623	Налаштування режиму підрахунку для HC2. HC2 веде зворотний відлік, коли SM623 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM624	Налаштування режиму підрахунку для HC3. HC3 веде зворотний відлік, коли SM624 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM625	Налаштування режиму підрахунку для HC4. HC4 веде зворотний відлік, коли SM625 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM626	Налаштування режиму підрахунку для HC5. HC5 веде зворотний відлік, коли SM626 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM627	Налаштування режиму підрахунку для HC6. HC6 веде зворотний відлік, коли SM627 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM628	Налаштування режиму підрахунку для HC7. HC7 веде зворотний відлік, коли SM628 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM629	Налаштування режиму підрахунку для HC8. HC8 веде зворотний відлік, коли SM629 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM630	Налаштування режиму підрахунку для HC9. HC9 веде зворотний відлік, коли SM630 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM631	Налаштування режиму підрахунку для HC10. HC10 веде зворотний відлік, коли SM631 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM632	Налаштування режиму підрахунку для HC11. HC11 веде зворотний відлік, коли SM632 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM633	Налаштування режиму підрахунку для HC12. HC12 веде зворотний відлік, коли SM633 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM634	Налаштування режиму підрахунку для HC13. HC13 веде зворотний відлік, коли SM634 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM635	Налаштування режиму підрахунку для HC14. HC14 веде зворотний відлік, коли SM635 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM636	Налаштування режиму підрахунку для HC15. HC15 веде зворотний відлік, коли SM636 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM637	Налаштування режиму підрахунку для HC16. HC16 веде зворотний відлік, коли SM637 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM638	Налаштування режиму підрахунку для HC17. HC17 веде зворотний відлік, коли SM638 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM639	Налаштування режиму підрахунку для HC18. HC18 веде зворотний відлік, коли SM639 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM640	Налаштування режиму підрахунку для HC19. HC19 веде зворотний відлік, коли SM640 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM641	Налаштування режиму підрахунку для HC20. HC20 веде зворотний відлік, коли SM641 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM642	Налаштування режиму підрахунку для HC21.	○	○	OFF	—	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
	HC21 веде зворотний відлік, коли SM642 увімкнено.								
SM643	Налаштування режиму підрахунку для HC22. HC22 веде зворотний відлік, коли SM643 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM644	Налаштування режиму підрахунку для HC23. HC23 веде зворотний відлік, коли SM644 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM645	Налаштування режиму підрахунку для HC24. HC24 веде зворотний відлік, коли SM645 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM646	Налаштування режиму підрахунку для HC25. HC25 веде зворотний відлік, коли SM646 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM647	Налаштування режиму підрахунку для HC26. HC26 веде зворотний відлік, коли SM647 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM648	Налаштування режиму підрахунку для HC27. HC27 веде зворотний відлік, коли SM648 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM649	Налаштування режиму підрахунку для HC28. HC28 веде зворотний відлік, коли SM649 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM650	Налаштування режиму підрахунку для HC29. HC29 веде зворотний відлік, коли SM650 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM651	Налаштування режиму підрахунку для HC30. HC30 веде зворотний відлік, коли SM651 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM652	Налаштування режиму підрахунку для HC31. HC31 веде зворотний відлік, коли SM652 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM653	Налаштування режиму підрахунку для HC32. HC32 веде зворотний відлік, коли SM653 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM654	Налаштування режиму підрахунку для HC33. HC33 веде зворотний відлік, коли SM654 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM655	Налаштування режиму підрахунку для HC34. HC34 веде зворотний відлік, коли SM655 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM656	Налаштування режиму підрахунку для HC35. HC35 веде зворотний відлік, коли SM656 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM657	Налаштування режиму підрахунку для HC36. HC36 веде зворотний відлік, коли SM657 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM658	Налаштування режиму підрахунку для HC37. HC37 веде зворотний відлік, коли SM658 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM659	Налаштування режиму підрахунку для HC38. HC38 веде зворотний відлік, коли SM659 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM660	Налаштування режиму підрахунку для HC39. HC39 веде зворотний відлік, коли SM660 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM661	Налаштування режиму підрахунку для HC40. HC40 веде зворотний відлік, коли SM661 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM662	Налаштування режиму підрахунку для HC41. HC41 веде зворотний відлік, коли SM662 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM663	Налаштування режиму підрахунку для HC42. HC42 веде зворотний відлік, коли SM663 увімкнено.	○	○	OFF	—	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM664	Налаштування режиму підрахунку для HC43. HC43 веде зворотний відлік, коли SM664 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM665	Налаштування режиму підрахунку для HC44. HC44 веде зворотний відлік, коли SM665 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM666	Налаштування режиму підрахунку для HC45. HC45 веде зворотний відлік, коли SM666 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM667	Налаштування режиму підрахунку для HC46. HC46 веде зворотний відлік, коли SM667 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM668	Налаштування режиму підрахунку для HC47. HC47 веде зворотний відлік, коли SM668 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM669	Налаштування режиму підрахунку для HC48. HC48 веде зворотний відлік, коли SM669 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM670	Налаштування режиму підрахунку для HC49. HC49 веде зворотний відлік, коли SM670 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM671	Налаштування режиму підрахунку для HC50. HC50 веде зворотний відлік, коли SM671 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM672	Налаштування режиму підрахунку для HC51. HC51 веде зворотний відлік, коли SM672 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM673	Налаштування режиму підрахунку для HC52. HC52 веде зворотний відлік, коли SM673 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM674	Налаштування режиму підрахунку для HC53. HC53 веде зворотний відлік, коли SM674 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM675	Налаштування режиму підрахунку для HC54. HC54 веде зворотний відлік, коли SM675 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM676	Налаштування режиму підрахунку для HC55. HC55 веде зворотний відлік, коли SM676 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM677	Налаштування режиму підрахунку для HC56. HC56 веде зворотний відлік, коли SM677 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM678	Налаштування режиму підрахунку для HC57. HC57 веде зворотний відлік, коли SM678 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM679	Налаштування режиму підрахунку для HC58. HC58 веде зворотний відлік, коли SM679 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM680	Налаштування режиму підрахунку для HC59. HC59 веде зворотний відлік, коли SM680 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM681	Налаштування режиму підрахунку для HC60. HC60 веде зворотний відлік, коли SM681 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM682	Налаштування режиму підрахунку для HC61. HC61 веде зворотний відлік, коли SM682 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM683	Налаштування режиму підрахунку для HC62. HC62 веде зворотний відлік, коли SM683 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM684	Налаштування режиму підрахунку для HC63. HC63 веде зворотний відлік, коли SM684 увімкнено.	○	○	OFF	—	—	N	R/W	OFF
SM685	Інструкція DSCLP використовує число з плаваючою комою	○	○	OFF	—	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
	операції.								
SM686	Режим інструкцій RAMP	○	○	OFF	–	–	N	R/W	OFF
SM687	Виконання інструкції RAMP завершено.	○	○	OFF	–	–	N	R/W	OFF
SM688	Виконання інструкції INCD завершено.	○	○	OFF	–	–	N	R/W	OFF
SM690	Режим керування рядком.	○	○	OFF	–	–	N	R/W	OFF
SM691	Режим введення інструкцій НКУ 16-бітний. ON: вхід є шістнадцятковим введенням OFF: A–F є функціональними клавішами.	○	○	OFF	–	–	N	R/W	OFF
SM692	Після завершення виконання інструкції НКУ SM692 увімкнено для циклу сканування.	○	○	OFF	–	–	N	R/W	OFF
SM693	Після завершення виконання інструкції SEGL SM693 увімкнено для циклу сканування.	○	○	OFF	–	–	N	R/W	OFF
SM694	Після завершення виконання інструкції DSW SM694 увімкнено для циклу сканування.	○	○	OFF	–	–	N	R/W	OFF
SM695	Радіан/градус. ON: градуси OFF.: радіани	○	○	OFF	–	–	N	R/W	OFF
SM749	Виникає помилка при ініціалізації обміну даними через COM1.	○	○	OFF	–	–	N	R/W	OFF
*SM750	Обмін даними через COM1 підтримує ISPSOft.	○	○	OFF	–	–	H	R/W	OFF
*SM752	Розпочато підключення 1 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM753	Розпочато підключення 2 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM754	Розпочато підключення 3 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM755	Розпочато підключення 4 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM756	Розпочато з'єднання 5 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM757	Розпочато підключення 6 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM758	Розпочато підключення 7 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM759	Розпочато підключення 8 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM760	Розпочато підключення 9 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM761	Розпочато підключення 10 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM762	Розпочато підключення 11 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF
*SM763	Розпочато підключення 12 для обміну даними через COM1	○	○	OFF	–	–	H	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM764	Розпочато підключення 13 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM765	Розпочато підключення 14 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM766	Розпочато підключення 15 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM767	Розпочато підключення 16 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM768	Розпочато підключення 17 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM769	Розпочато підключення 18 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM770	Розпочато підключення 19 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM771	Розпочато підключення 20 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM772	Розпочато підключення 21 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM773	Розпочато підключення 22 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM774	Розпочато підключення 23 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM775	Розпочато підключення 24 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM776	Розпочато підключення 25 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM777	Розпочато підключення 26 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM778	Розпочато підключення 27 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM779	Розпочато підключення 28 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM780	Розпочато з'єднання 29 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM781	Розпочато підключення 30 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM782	Розпочато підключення 31 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM783	Розпочато підключення 32 для обміну даними через COM1	○	○	OFF	—	—	H	R/W	OFF
*SM784	Успішне з'єднання для обміну даними 1 через COM1	○	○	OFF	—	—	N	R	OFF
*SM785	Успішне з'єднання для обміну даними 2 через COM1	○	○	OFF	—	—	N	R	OFF
*SM786	Успішне з'єднання для обміну даними 3 через COM1	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM787	Успішне з'єднання для обміну даними 4 через COM1	○	○	OFF	—	—	N	R	OFF
*SM788	Успішне з'єднання для обміну даними 5 через COM1	○	○	OFF	—	—	N	R	OFF
*SM789	Успішне з'єднання для обміну даними 6 через COM1	○	○	OFF	—	—	N	R	OFF
*SM790	Успішне з'єднання для обміну даними 7 через COM1	○	○	OFF	—	—	N	R	OFF
*SM791	Успішне з'єднання для обміну даними 8 через COM1	○	○	OFF	—	—	N	R	OFF
*SM792	Успішне з'єднання для обміну даними 9 через COM1	○	○	OFF	—	—	N	R	OFF
*SM793	Успішне з'єднання для обміну даними 10 через COM1	○	○	OFF	—	—	N	R	OFF
*SM794	Успішне з'єднання для обміну даними 11 через COM1	○	○	OFF	—	—	N	R	OFF
*SM795	Успішне з'єднання для обміну даними 12 через COM1	○	○	OFF	—	—	N	R	OFF
*SM796	Успішне з'єднання для обміну даними 13 через COM1	○	○	OFF	—	—	N	R	OFF
*SM797	Успішне з'єднання для обміну даними 14 через COM1	○	○	OFF	—	—	N	R	OFF
*SM798	Успішне з'єднання для обміну даними 15 через COM1	○	○	OFF	—	—	N	R	OFF
*SM799	Успішне з'єднання для обміну даними 16 через COM1	○	○	OFF	—	—	N	R	OFF
*SM800	Успішне з'єднання для обміну даними 17 через COM1	○	○	OFF	—	—	N	R	OFF
*SM801	Успішне з'єднання для обміну даними 18 через COM1	○	○	OFF	—	—	N	R	OFF
*SM802	Успішне з'єднання для обміну даними 19 через COM1	○	○	OFF	—	—	N	R	OFF
*SM803	Успішне з'єднання для обміну даними 20 через COM1	○	○	OFF	—	—	N	R	OFF
*SM804	Успішне з'єднання для обміну даними 21 через COM1	○	○	OFF	—	—	N	R	OFF
*SM805	Успішне з'єднання для обміну даними 22 через COM1	○	○	OFF	—	—	N	R	OFF
*SM806	Успішне з'єднання для обміну даними 23 через COM1	○	○	OFF	—	—	N	R	OFF
*SM807	Успішне з'єднання для обміну даними 24 через COM1	○	○	OFF	—	—	N	R	OFF
*SM808	Успішне з'єднання для обміну даними 25 через COM1	○	○	OFF	—	—	N	R	OFF
*SM809	Успішне з'єднання для обміну даними 26 через COM1	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
*SM810	Успішне з'єднання для обміну даними 27 через COM1	○	○	OFF	—	—	N	R	OFF
*SM811	Успішне з'єднання для обміну даними 28 через COM1	○	○	OFF	—	—	N	R	OFF
*SM812	Успішне з'єднання для обміну даними 29 через COM1	○	○	OFF	—	—	N	R	OFF
*SM813	Успішне з'єднання для обміну даними 30 через COM1	○	○	OFF	—	—	N	R	OFF
*SM814	Успішне з'єднання для обміну даними 31 через COM1	○	○	OFF	—	—	N	R	OFF
*SM815	Успішне з'єднання для обміну даними 32 через COM1	○	○	OFF	—	—	N	R	OFF
*SM816	Помилка підключення обміну даними 1 через COM1	○	○	OFF	—	—	N	R	OFF
*SM817	Помилка підключення обміну даними 2 через COM1	○	○	OFF	—	—	N	R	OFF
*SM818	Помилка підключення обміну даними 3 через COM1	○	○	OFF	—	—	N	R	OFF
*SM819	Помилка підключення обміну даними 4 через COM1	○	○	OFF	—	—	N	R	OFF
*SM820	Помилка підключення обміну даними 5 через COM1	○	○	OFF	—	—	N	R	OFF
*SM821	Помилка підключення обміну даними 6 через COM1	○	○	OFF	—	—	N	R	OFF
*SM822	Помилка підключення обміну даними 7 через COM1	○	○	OFF	—	—	N	R	OFF
*SM823	Помилка підключення обміну даними 8 через COM1	○	○	OFF	—	—	N	R	OFF
*SM824	Помилка підключення обміну даними 9 через COM1	○	○	OFF	—	—	N	R	OFF
*SM825	Помилка підключення обміну даними 10 через COM1	○	○	OFF	—	—	N	R	OFF
*SM826	Помилка підключення обміну даними 11 через COM1	○	○	OFF	—	—	N	R	OFF
*SM827	Помилка підключення обміну даними 12 через COM1	○	○	OFF	—	—	N	R	OFF
*SM828	Помилка підключення обміну даними 13 через COM1	○	○	OFF	—	—	N	R	OFF
*SM829	Помилка підключення обміну даними 14 через COM1	○	○	OFF	—	—	N	R	OFF
*SM830	Помилка підключення обміну даними 15 через COM1	○	○	OFF	—	—	N	R	OFF
*SM831	Помилка підключення обміну даними 16 через COM1	○	○	OFF	—	—	N	R	OFF
*SM832	Помилка підключення обміну даними 17 через COM1	○	○	OFF	—	—	N	R	OFF
*SM833	Помилка підключення обміну даними 18 через COM1	○	○	OFF	—	—	N	R	OFF
*SM834	Помилка підключення обміну даними 19 через COM1	○	○	OFF	—	—	N	R	OFF
*SM835	Помилка підключення обміну даними 20 через COM1	○	○	OFF	—	—	N	R	OFF
*SM836	Помилка підключення обміну даними 21 через COM1	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF	STOP	RUN	Засув	Атрибут	За замовч.
				↕ ON	↕ RUN	↕ STOP			
*SM837	Помилка підключення обміну даними 22 через COM1	○	○	OFF	—	—	N	R	OFF
*SM838	Помилка підключення обміну даними 23 через COM1	○	○	OFF	—	—	N	R	OFF
*SM839	Помилка підключення обміну даними 24 через COM1	○	○	OFF	—	—	N	R	OFF
*SM840	Помилка підключення обміну даними 25 через COM1	○	○	OFF	—	—	N	R	OFF
*SM841	Помилка підключення обміну даними 26 через COM1	○	○	OFF	—	—	N	R	OFF
*SM842	Помилка підключення обміну даними 27 через COM1	○	○	OFF	—	—	N	R	OFF
*SM843	Помилка підключення обміну даними 28 через COM1	○	○	OFF	—	—	N	R	OFF
*SM844	Помилка підключення обміну даними 29 через COM1	○	○	OFF	—	—	N	R	OFF
*SM845	Помилка підключення обміну даними 30 через COM1	○	○	OFF	—	—	N	R	OFF
*SM846	Помилка підключення обміну даними 31 через COM1	○	○	OFF	—	—	N	R	OFF
*SM847	Помилка підключення обміну даними 32 через COM1	○	○	OFF	—	—	N	R	OFF
SM861	Помилка ініціалізації обміну даними через COM2.	○	○	OFF	—	—	N	R/W	OFF
*SM862	Обмін даними через COM2 підтримує ISPSOft.	○	○	OFF	—	OFF	H	R/W	OFF
*SM864	Розпочато підключення 1 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM865	Розпочато підключення 2 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM866	Розпочато підключення 3 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM867	Розпочато підключення 4 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM868	Розпочато з'єднання 5 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM869	Розпочато підключення 6 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM870	Розпочато підключення 7 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM871	Розпочато підключення 8 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM872	Розпочато підключення 9 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM873	Розпочато підключення 10 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM874	З'єднання 11 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
	почався								
*SM875	Розпочато підключення 12 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM876	Розпочато підключення 13 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM877	Розпочато підключення 14 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM878	Розпочато підключення 15 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM879	Розпочато підключення 16 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM880	Розпочато підключення 17 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM881	Розпочато підключення 18 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM882	Розпочато підключення 19 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM883	Розпочато підключення 20 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM884	Розпочато підключення 21 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM885	Розпочато підключення 22 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM886	Розпочато підключення 23 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM887	Розпочато підключення 24 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM888	Розпочато підключення 25 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM889	Розпочато підключення 26 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM890	Розпочато підключення 27 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM891	Розпочато підключення 28 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM892	Розпочато підключення 29 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM893	Розпочато підключення 30 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM894	Розпочато підключення 31 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM895	Розпочато підключення 32 для обміну даними через COM2	○	○	OFF	—	—	H	R/W	OFF
*SM896	Успішне з'єднання для обміну даними 1 через COM2	○	○	OFF	—	—	N	R	OFF
*SM897	Успішне підключення для обміну даними 2 через	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
	COM2								
*SM898	Успішне з'єднання для обміну даними 3 через COM2	○	○	OFF	—	—	N	R	OFF
*SM899	Успішне з'єднання для обміну даними 4 через COM2	○	○	OFF	—	—	N	R	OFF
*SM900	Успішне з'єднання для обміну даними 5 через COM2	○	○	OFF	—	—	N	R	OFF
*SM901	Успішне з'єднання для обміну даними 6 через COM2	○	○	OFF	—	—	N	R	OFF
*SM902	Успішне з'єднання для обміну даними 7 через COM2	○	○	OFF	—	—	N	R	OFF
*SM903	Успішне з'єднання для обміну даними 8 через COM2	○	○	OFF	—	—	N	R	OFF
*SM904	Успішне з'єднання для обміну даними 9 через COM2	○	○	OFF	—	—	N	R	OFF
*SM905	Успішне з'єднання для обміну даними 10 через COM2	○	○	OFF	—	—	N	R	OFF
*SM906	Успішне з'єднання для обміну даними 11 через COM2	○	○	OFF	—	—	N	R	OFF
*SM907	Успішне з'єднання для обміну даними 12 через COM2	○	○	OFF	—	—	N	R	OFF
*SM908	Успішне з'єднання для обміну даними 13 через COM2	○	○	OFF	—	—	N	R	OFF
*SM909	Успішне з'єднання для обміну даними 14 через COM2	○	○	OFF	—	—	N	R	OFF
*SM910	Успішне з'єднання для обміну даними 15 через COM2	○	○	OFF	—	—	N	R	OFF
*SM911	Успішне з'єднання для обміну даними 16 через COM2	○	○	OFF	—	—	N	R	OFF
*SM912	Успішне з'єднання для обміну даними 17 через COM2	○	○	OFF	—	—	N	R	OFF
*SM913	Успішне з'єднання для обміну даними 18 через COM2	○	○	OFF	—	—	N	R	OFF
*SM914	Успішне з'єднання для обміну даними 19 через COM2	○	○	OFF	—	—	N	R	OFF
*SM915	Успішне з'єднання для обміну даними 20 через COM2	○	○	OFF	—	—	N	R	OFF
*SM916	Успішне з'єднання для обміну даними 21 через COM2	○	○	OFF	—	—	N	R	OFF
*SM917	Успішне з'єднання для обміну даними 22 через COM2	○	○	OFF	—	—	N	R	OFF
*SM918	Успішне з'єднання для обміну даними 23 через COM2	○	○	OFF	—	—	N	R	OFF
*SM919	Успішне з'єднання для обміну даними 24 через COM2	○	○	OFF	—	—	N	R	OFF
*SM920	Успішне з'єднання для обміну даними через 25	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
	COM2								
*SM921	Успішне з'єднання для обміну даними 26 через COM2	○	○	OFF	—	—	N	R	OFF
*SM922	Успішне з'єднання для обміну даними 27 через COM2	○	○	OFF	—	—	N	R	OFF
*SM923	Успішне з'єднання для обміну даними 28 через COM2	○	○	OFF	—	—	N	R	OFF
*SM924	Успішне з'єднання для обміну даними 29 через COM2	○	○	OFF	—	—	N	R	OFF
*SM925	Успішне з'єднання для обміну даними 30 через COM2	○	○	OFF	—	—	N	R	OFF
*SM926	Успішне з'єднання для обміну даними 31 через COM2	○	○	OFF	—	—	N	R	OFF
*SM927	Успішне з'єднання для обміну даними 32 через COM2	○	○	OFF	—	—	N	R	OFF
*SM928	Помилка підключення обміну даними 1 через COM2	○	○	OFF	—	—	N	R	OFF
*SM929	Помилка підключення обміну даними 2 через COM2	○	○	OFF	—	—	N	R	OFF
*SM930	Помилка підключення обміну даними 3 через COM2	○	○	OFF	—	—	N	R	OFF
*SM931	Помилка підключення обміну даними 4 через COM2	○	○	OFF	—	—	N	R	OFF
*SM932	Помилка підключення обміну даними 5 через COM2	○	○	OFF	—	—	N	R	OFF
*SM933	Помилка підключення обміну даними 6 через COM2	○	○	OFF	—	—	N	R	OFF
*SM934	Помилка підключення обміну даними 7 через COM2	○	○	OFF	—	—	N	R	OFF
*SM935	Помилка підключення обміну даними 8 через COM2	○	○	OFF	—	—	N	R	OFF
*SM936	Помилка підключення обміну даними 9 через COM2	○	○	OFF	—	—	N	R	OFF
*SM937	Помилка підключення обміну даними 10 через COM2	○	○	OFF	—	—	N	R	OFF
*SM938	Помилка підключення обміну даними 11 через COM2	○	○	OFF	—	—	N	R	OFF
*SM939	Помилка підключення обміну даними 12 через COM2	○	○	OFF	—	—	N	R	OFF
*SM940	Помилка підключення обміну даними 13 через COM2	○	○	OFF	—	—	N	R	OFF
*SM941	Помилка підключення обміну даними 14 через COM2	○	○	OFF	—	—	N	R	OFF
*SM942	Помилка підключення обміну даними 15 через COM2	○	○	OFF	—	—	N	R	OFF
*SM943	Помилка підключення обміну даними 16 через COM2	○	○	OFF	—	—	N	R	OFF
*SM944	Помилка підключення обміну даними 17 через COM2	○	○	OFF	—	—	N	R	OFF
*SM945	Помилка підключення обміну даними 18 через COM2	○	○	OFF	—	—	N	R	OFF
*SM946	Помилка підключення обміну даними 19 через COM2	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM947	Помилка підключення обміну даними 20 через COM2	○	○	OFF	—	—	N	R	OFF
*SM948	Помилка підключення обміну даними 21 через COM2	○	○	OFF	—	—	N	R	OFF
*SM949	Помилка підключення обміну даними 22 через COM2	○	○	OFF	—	—	N	R	OFF
*SM950	Помилка підключення обміну даними 23 через COM2	○	○	OFF	—	—	N	R	OFF
*SM951	Помилка підключення обміну даними 24 через COM2	○	○	OFF	—	—	N	R	OFF
*SM952	Помилка підключення обміну даними 25 через COM2	○	○	OFF	—	—	N	R	OFF
*SM953	Помилка підключення обміну даними 26 через COM2	○	○	OFF	—	—	N	R	OFF
*SM954	Помилка підключення обміну даними 27 через COM2	○	○	OFF	—	—	N	R	OFF
*SM955	Помилка підключення обміну даними 28 через COM2	○	○	OFF	—	—	N	R	OFF
*SM956	Помилка підключення обміну даними 29 через COM2	○	○	OFF	—	—	N	R	OFF
*SM957	Помилка підключення обміну даними 30 через COM2	○	○	OFF	—	—	N	R	OFF
*SM958	Помилка підключення обміну даними 31 через COM2	○	○	OFF	—	—	N	R	OFF
*SM959	Помилка підключення обміну даними 32 через COM2	○	○	OFF	—	—	N	R	OFF
SM976	Статус підключення віддаленого модуля AS №1 (УВІМК.: підключено; OFF.: відключено або не підключено)	○	○	OFF	—	—	N	R	OFF
SM977	Статус підключення віддаленого модуля AS №2	○	○	OFF	—	—	N	R	OFF
SM978	Статус підключення віддаленого модуля AS №3	○	○	OFF	—	—	N	R	OFF
SM979	Статус підключення віддаленого модуля AS №4	○	○	OFF	—	—	N	R	OFF
SM980	Статус підключення віддаленого модуля AS №5	○	○	OFF	—	—	N	R	OFF
SM981	Стан підключення віддаленого модуля AS №6	○	○	OFF	—	—	N	R	OFF
SM982	Статус підключення віддаленого модуля AS №7	○	○	OFF	—	—	N	R	OFF
SM983	Статус підключення віддаленого модуля AS №8	○	○	OFF	—	—	N	R	OFF
SM984	Статус підключення віддаленого модуля AS №9	○	○	OFF	—	—	N	R	OFF
SM985	Статус підключення віддаленого модуля AS №10	○	○	OFF	—	—	N	R	OFF
SM986	Статус підключення віддаленого модуля AS №11	○	○	OFF	—	—	N	R	OFF
SM987	Статус підключення віддаленого модуля AS №12	○	○	OFF	—	—	N	R	OFF
SM988	Статус підключення віддаленого модуля AS №13	○	○	OFF	—	—	N	R	OFF
SM989	Статус підключення віддаленого модуля AS №14	○	○	OFF	—	—	N	R	OFF
SM990	Статус підключення віддаленого модуля AS №15	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM1000	Прапор налаштування Ethernet; ON: дані в SR1000–SR1006 записуються у флеш-пам'ять.	○	○	OFF	–	–	N	R/W	OFF
SM1001	Стан підключення Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1006	Обмін даними через AS-FEN02 підтримує ISPSOft.	○	–	OFF	–	–	N	R/W	OFF
*SM1008	Розпочато підключення 1 для обміну даними через AS-FEN02	○	–	OFF	–	–	N	R/W	OFF
*SM1009	Розпочато підключення 2 для обміну даними через AS-FEN02	○	–	OFF	–	–	N	R/W	OFF
*SM1010	Розпочато підключення 3 для обміну даними через AS-FEN02	○	–	OFF	–	–	N	R/W	OFF
*SM1011	Розпочато підключення 4 для обміну даними через AS-FEN02	○	–	OFF	–	–	N	R/W	OFF
*SM1012	Розпочато підключення 5 для обміну даними через AS-FEN02	○	–	OFF	–	–	N	R/W	OFF
*SM1013	Розпочато підключення 6 для обміну даними через AS-FEN02	○	–	OFF	–	–	N	R/W	OFF
*SM1014	Розпочато підключення 7 для обміну даними через AS-FEN02	○	–	OFF	–	–	N	R/W	OFF
*SM1015	Розпочато підключення 8 для обміну даними через AS-FEN02	○	–	OFF	–	–	N	R/W	OFF
*SM1016	Успішне з'єднання для обміну даними 1 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1017	Успішне з'єднання для обміну даними 2 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1018	Успішне з'єднання для обміну даними 3 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1019	Успішне з'єднання для обміну даними 4 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1020	Успішне з'єднання для обміну даними 5 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1021	Успішне з'єднання для обміну даними 6 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1022	Успішне з'єднання для обміну даними 7 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1023	Успішне з'єднання для обміну даними 8 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1024	Помилка підключення обміну даними 1 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1025	Помилка підключення обміну даними 2 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1026	Помилка підключення обміну даними 3 через AS-FEN02	○	–	OFF	–	–	N	R	OFF
*SM1027	Помилка підключення обміну даними 4 через AS-FEN02	○	–	OFF	–	–	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM1028	Помилка підключення обміну даними 5 через AS-FEN02	○	—	OFF	—	—	N	R	OFF
*SM1029	Помилка підключення обміну даними 6 через AS-FEN02	○	—	OFF	—	—	N	R	OFF
*SM1030	Помилка підключення обміну даними 7 через AS-FEN02	○	—	OFF	—	—	N	R	OFF
*SM1031	Помилка підключення обміну даними 8 через AS-FEN02	○	—	OFF	—	—	N	R	OFF
*SM1090	TCP-з'єднання зайняте.	○	○	OFF	—	—	N	R	OFF
*SM1091	UDP-з'єднання зайняте.	○	○	OFF	—	—	N	R	OFF
SM1092	Неправильний порт сервера Modbus TCP (доступний для FW V1.10.20 або новішої версії)	○	○	OFF	—	—	N	R/W	OFF
SM1100	Помилка підключення Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1106	Базове керування Ethernet: помилка підключення	○	○	OFF	—	—	N	R	OFF
*SM1107	Базове керування Ethernet: помилка базового налаштування	○	○	OFF	—	—	N	R	OFF
*SM1109	Базове керування сокетом TCP/UDP: локальний порт уже використовується.	○	○	OFF	—	—	N	R	OFF
SM1111	Прапор обміну даними EtherNet/IP	○	○	OFF	—	—	N	R	OFF
*SM1113	Помилка служби електронної пошти	○	○	OFF	—	—	N	R	OFF
*SM1116	Перемикач 1 для електронної пошти.	○	○	OFF	—	—	N	R	OFF
*SM1117	Тригер 1 для електронної пошти	○	○	OFF	—	—	N	R	OFF
*SM1119	ON: спрацьовує тригер 1 і електронний лист успішно надіслано.	○	○	OFF	—	—	N	R	OFF
*SM1120	ON: активується тригер 1, але електронний лист неможливо надіслати через помилку вмісту електронного листа.	○	○	OFF	—	—	N	R	OFF
*SM1122	ON: спрацьовує тригер 1 і час очікування відповіді сервера SMTP.	○	○	OFF	—	—	N	R	OFF
*SM1123	ON: спрацьовує тригер 1 і виникає помилка відповіді сервера SMTP.	○	○	OFF	—	—	N	R	OFF
*SM1124	ON: спрацьовує тригер 1, і розмір вкладеного файлу перевищує обмеження.	○	○	OFF	—	—	N	R	OFF
*SM1125	Тригер 1 спрацьовує, і вкладення не знайдено.	○	○	OFF	—	—	N	R	OFF
*SM1126	Тригер 2 для електронної пошти.	○	○	OFF	—	—	N	R	OFF
*SM1127	Тригер 2 для електронного листа	○	○	OFF	—	—	N	R	OFF
*SM1129	ON: запускається тригер 2, і електронний лист успішно надіслано.	○	○	OFF	—	—	N	R	OFF
*SM1130	ON: активується тригер 2, але електронний лист не може бути надісланий через помилку вмісту електронного листа.	○	○	OFF	—	—	N	R	OFF
*SM1132	ON: спрацьовує тригер 2 і час очікування відповіді сервера SMTP.	○	○	OFF	—	—	N	R	OFF
*SM1133	ON: спрацьовує тригер 2 і виникає помилка відповіді сервера SMTP.	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
*SM1134	ON: спрацьовує тригер 2, і розмір вложеного файлу перевищує обмеження.	○	○	OFF	—	—	N	R	OFF
*SM1135	ON: Тригер 2 спрацьовує, і вкладення не знайдено.	○	○	OFF	—	—	N	R	OFF
*SM1136	Тригер 3 для електронної пошти.	○	○	OFF	—	—	N	R	OFF
*SM1137	Тригер 3 для електронного листа	○	○	OFF	—	—	N	R	OFF
*SM1139	ON: спрацьовує тригер 3 і електронний лист успішно надіслано.	○	○	OFF	—	—	N	R	OFF
*SM1140	ON: Тригер 3 спрацьовує, але електронний лист неможливо надіслати через помилку вмісту електронного листа.	○	○	OFF	—	—	N	R	OFF
*SM1142	ON: спрацьовує тригер 3 і час очікування відповіді сервера SMTP.	○	○	OFF	—	—	N	R	OFF
*SM1143	ON: спрацьовує тригер 3 і виникає помилка відповіді сервера SMTP.	○	○	OFF	—	—	N	R	OFF
*SM1144	ON: спрацьовує тригер 3, і розмір вложеного файлу перевищує обмеження.	○	○	OFF	—	—	N	R	OFF
*SM1145	ON: Тригер 3 спрацьовує, і вкладення не знайдено.	○	○	OFF	—	—	N	R	OFF
*SM1146	Тригер 4 для електронної пошти.	○	○	OFF	—	—	N	R	OFF
*SM1147	Тригер 4 для електронного листа	○	○	OFF	—	—	N	R	OFF
*SM1149	ON: спрацьовує тригер 4 і електронний лист успішно надіслано.	○	○	OFF	—	—	N	R	OFF
*SM1150	ON: активується тригер 4, але електронний лист неможливо надіслати через помилку вмісту електронного листа.	○	○	OFF	—	—	N	R	OFF
*SM1152	ON: спрацьовує тригер 4 і час очікування відповіді сервера SMTP.	○	○	OFF	—	—	N	R	OFF
*SM1153	ON: спрацьовує тригер 4 і виникає помилка відповіді сервера SMTP.	○	○	OFF	—	—	N	R	OFF
*SM1154	ON: спрацьовує тригер 4, і розмір вложеного файлу перевищує обмеження.	○	○	OFF	—	—	N	R	OFF
*SM1155	ON: Тригер 4 спрацьовує, і вкладення не знайдено.	○	○	OFF	—	—	N	R	OFF
SM1160	Процес відновлення параметрів накопичувача завершено. OFF.: коли значення в SR1600 змінюється з 1 на 0, цей прапорець скидається на OFF. ON: коли процес відновлення параметрів завершено та без помилок, цей прапорець увімкнено.	○	○	OFF	—	—	N	R	OFF
SM1161	Сталася помилка під час процесу відновлення параметрів накопичувача. OFF.: коли значення в SR1600 змінюється з 1 на 0, цей прапорець скидається на OFF. ON: якщо під час процесу відновлення параметра сталася помилка, цей прапорець увімкнено.	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF	STOP	RUN	Засув	Атрибут	За замовч.
				⇕ ON	⇕ RUN	⇕ STOP			
*SM1166	Помилка обміну даними через Ethernet	○	○	—	—	—	N	R	OFF
*SM1167	Розпочато обмін даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1168	Розпочато підключення 1 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1169	Розпочато підключення 2 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1170	Розпочато підключення 3 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1171	Розпочато підключення 4 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1172	Розпочато підключення 5 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1173	Розпочато підключення 6 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1174	Розпочато підключення 7 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1175	Розпочато підключення 8 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1176	Розпочато підключення 9 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1177	Розпочато підключення 10 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1178	Розпочато підключення 11 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1179	Розпочато підключення 12 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1180	Розпочато підключення 13 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1181	Розпочато підключення 14 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1182	Розпочато підключення 15 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1183	Розпочато підключення 16 для обміну даними через Ethernet	○	○	OFF	—	—	H	R/W	OFF
*SM1184	Розпочато з'єднання 17 для обміну даними через Ethernet	○	—	OFF	—	—	H	R/W	OFF
*SM1185	Розпочато підключення 18 для обміну даними через Ethernet	○	—	OFF	—	—	H	R/W	OFF
*SM1186	Розпочато підключення 19 для обміну даними через Ethernet	○	—	OFF	—	—	H	R/W	OFF
*SM1187	Розпочато підключення 20 для обміну даними через Ethernet	○	—	OFF	—	—	H	R/W	OFF
*SM1188	Розпочато з'єднання 21 для обміну даними через Ethernet	○	—	OFF	—	—	H	R/W	OFF
*SM1189	Розпочато з'єднання 22 для обміну даними через Ethernet	○	—	OFF	—	—	H	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM1190	Розпочато з'єднання 23 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1191	Розпочато з'єднання 24 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1192	Розпочато з'єднання 25 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1193	Розпочато з'єднання 26 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1194	Розпочато з'єднання 27 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1195	Розпочато підключення 28 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1196	Розпочато з'єднання 29 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1197	Розпочато підключення 30 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1198	Розпочато з'єднання 31 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1199	Розпочато підключення 32 для обміну даними через Ethernet	○	–	OFF	–	–	H	R/W	OFF
*SM1200	Успішне з'єднання для обміну даними 1 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1201	Успішне з'єднання для обміну даними 2 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1202	Успішне з'єднання для обміну даними 3 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1203	Успішне з'єднання для обміну даними 4 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1204	Успішне з'єднання для обміну даними 5 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1205	Успішне з'єднання для обміну даними 6 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1206	Успішне з'єднання для обміну даними 7 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1207	Успішне з'єднання для обміну даними 8 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1208	Успішне з'єднання для обміну даними 9 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1209	Успішне з'єднання для обміну даними 10 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1210	Успішне з'єднання для обміну даними 11 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1211	Успішне з'єднання для обміну даними 12 через Ethernet	○	○	OFF	–	–	N	R	OFF
*SM1212	Успішне з'єднання для обміну даними 13 через Ethernet	○	○	OFF	–	–	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM1213	Успішне з'єднання для обміну даними 14 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1214	Успішне з'єднання для обміну даними 15 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1215	Успішне з'єднання для обміну даними 16 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1216	Успішне з'єднання для обміну даними 17 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1217	Успішне з'єднання для обміну даними 18 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1218	Успішне з'єднання для обміну даними 19 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1219	Успішне з'єднання для обміну даними 20 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1220	Успішне з'єднання для обміну даними 21 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1221	Успішне з'єднання для обміну даними 22 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1222	Успішне з'єднання для обміну даними 23 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1223	Успішне з'єднання для обміну даними 24 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1224	Успішне з'єднання для обміну даними 25 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1225	Успішне з'єднання для обміну даними 26 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1226	Успішне з'єднання для обміну даними 27 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1227	Успішне з'єднання для обміну даними 28 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1228	Успішне з'єднання для обміну даними 29 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1229	Успішне з'єднання для обміну даними 30 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1230	Успішне з'єднання для обміну даними 31 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1231	Успішне з'єднання для обміну даними 32 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1232	Помилка підключення обміну даними 1 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1233	Помилка підключення обміну даними 2 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1234	Помилка в з'єднанні обміну даними 3 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1235	Помилка підключення обміну даними 4 через Ethernet	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM1236	Помилка підключення обміну даними 5 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1237	Помилка підключення обміну даними 6 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1238	Помилка підключення обміну даними 7 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1239	Помилка підключення обміну даними 8 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1240	Помилка підключення обміну даними 9 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1241	Помилка підключення обміну даними 10 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1242	Помилка підключення обміну даними 11 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1243	Помилка підключення обміну даними 12 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1244	Помилка підключення обміну даними 13 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1245	Помилка підключення обміну даними 14 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1246	Помилка підключення обміну даними 15 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1247	Помилка підключення обміну даними 16 через Ethernet	○	○	OFF	—	—	N	R	OFF
*SM1248	Помилка підключення обміну даними 17 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1249	Помилка підключення обміну даними 18 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1250	Помилка підключення обміну даними 19 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1251	Помилка підключення обміну даними 20 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1252	Помилка підключення обміну даними 21 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1253	Помилка підключення обміну даними 22 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1254	Помилка підключення обміну даними 23 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1255	Помилка підключення обміну даними 24 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1256	Помилка підключення обміну даними 25 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1257	Помилка підключення обміну даними 26 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1258	Помилка підключення обміну даними 27 через Ethernet	○	—	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM1259	Помилка підключення обміну даними 28 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1260	Помилка підключення обміну даними 29 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1261	Помилка підключення обміну даними 30 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1262	Помилка підключення обміну даними 31 через Ethernet	○	—	OFF	—	—	N	R	OFF
*SM1263	Помилка підключення обміну даними 32 через Ethernet	○	—	OFF	—	—	N	R	OFF
SM1269	Помилка конфігурації сокетів	○	○	OFF	—	—	N	R/W	OFF
SM1270	Успішне підключення TCP-сокета 1	○	○	OFF	—	—	N	R	OFF
SM1271	Отримано дані TCP-сокета 1	○	○	OFF	—	—	N	R	OFF
SM1272	Дані TCP-сокета 1 надіслано	○	○	OFF	—	—	N	R	OFF
SM1273	Запуск з'єднання TCP-сокета 1	○	○	OFF	—	—	N	R	OFF
SM1274	Закриття з'єднання TCP-сокета 1	○	○	ON	—	—	Y	R	ON
SM1275	Надсилаються дані TCP-сокета 1	○	○	OFF	—	—	N	R	OFF
SM1277	TCP сокет 1 — прапор помилки	○	○	OFF	—	—	N	R	OFF
SM1278	Успішне підключення TCP-сокета 2	○	○	OFF	—	—	N	R	OFF
SM1279	Отримано дані TCP-сокета 2	○	○	OFF	—	—	N	R	OFF
SM1280	Дані TCP-сокета 2 надіслано	○	○	OFF	—	—	N	R	OFF
SM1281	Запускається з'єднання TCP-сокета 2	○	○	OFF	—	—	N	R	OFF
SM1282	Закриття з'єднання TCP-сокета 2	○	○	ON	—	—	Y	R	ON
SM1283	Надсилаються дані TCP-сокета 2	○	○	OFF	—	—	N	R	OFF
SM1285	TCP-сокет 2 — позначка помилки	○	○	OFF	—	—	N	R	OFF
SM1286	Успішне підключення TCP-сокета 3	○	○	OFF	—	—	N	R	OFF
SM1287	Отримано дані TCP-сокета 3	○	○	OFF	—	—	N	R	OFF
SM1288	Дані TCP-сокета 3 надіслано	○	○	OFF	—	—	N	R	OFF
SM1289	Запускається з'єднання TCP-сокета 3	○	○	OFF	—	—	N	R	OFF
SM1290	Закриття з'єднання TCP-сокета 3	○	○	ON	—	—	Y	R	ON
SM1291	Надсилаються дані TCP-сокета 3.	○	○	OFF	—	—	N	R	OFF
SM1293	TCP-сокет 3—прапор помилки	○	○	OFF	—	—	N	R	OFF
SM1294	Успішне підключення TCP-сокета 4	○	○	OFF	—	—	N	R	OFF
SM1295	Отримано дані TCP-сокета 4	○	○	OFF	—	—	N	R	OFF
SM1296	Дані TCP-сокета 4 надіслано	○	○	OFF	—	—	N	R	OFF
SM1297	Запускається підключення TCP-сокета 4	○	○	OFF	—	—	N	R	OFF
SM1298	Закриття з'єднання TCP-сокета 4	○	○	ON	—	—	Y	R	ON
SM1299	Надсилаються дані TCP-сокета 4.	○	○	OFF	—	—	N	R	OFF
SM1301	TCP-сокет 4—прапор помилки	○	○	OFF	—	—	N	R	OFF
SM1312	Стан підключення 1 RTU-EN01 (ON: підключено; OFF: відключено чи ні)	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
	підключений)								
SM1313	Статус підключення 2 RTU-EN01	○	○	OFF	—	—	N	R	OFF
SM1314	Стан підключення 3 RTU-EN01	○	○	OFF	—	—	N	R	OFF
SM1315	Стан підключення 4 RTU-EN01	○	○	OFF	—	—	N	R	OFF
SM1334	Розпочато з'єднання UDP-сокета 1	○	○	OFF	—	—	N	R	OFF
SM1335	Отримання даних UDP-сокета 1	○	○	OFF	—	—	N	R	OFF
SM1336	Надсилання даних через сокет 1 UDP	○	○	OFF	—	—	N	R	OFF
SM1337	Підключення UDP-сокета 1 закрито	○	○	ON	—	—	Y	R	ON
SM1338	UDP-сокет 1 — позначка помилки	○	○	OFF	—	—	N	R	OFF
SM1339	Розпочато підключення через сокет 2 UDP	○	○	OFF	—	—	N	R	OFF
SM1340	Отримання даних через UDP-сокет 2	○	○	OFF	—	—	N	R	OFF
SM1341	Надсилання даних через сокет 2 UDP	○	○	OFF	—	—	N	R	OFF
SM1342	З'єднання UDP-сокета 2 закрито	○	○	ON	—	—	Y	R	ON
SM1343	UDP-сокет 2—позначка помилки	○	○	OFF	—	—	N	R	OFF
SM1344	Розпочато підключення через сокет 3 UDP	○	○	OFF	—	—	N	R	OFF
SM1345	Отримання даних через сокет 3 UDP	○	○	OFF	—	—	N	R	OFF
SM1346	Надсилання даних через сокет 3 UDP	○	○	OFF	—	—	N	R	OFF
SM1347	З'єднання UDP-сокета 3 закрито	○	○	ON	—	—	Y	R	ON
SM1348	UDP-сокет 3—позначка помилки	○	○	OFF	—	—	N	R	OFF
SM1349	Розпочато підключення через сокет 4 UDP	○	○	OFF	—	—	N	R	OFF
SM1350	Отримання даних через сокет UDP 4	○	○	OFF	—	—	N	R	OFF
SM1351	Надсилання даних через сокет 4 UDP	○	○	OFF	—	—	N	R	OFF
SM1352	З'єднання UDP-сокета 4 закрито	○	○	ON	—	—	Y	R	ON
SM1353	UDP-сокет 4—прапор помилки	○	○	OFF	—	—	N	R	OFF
*SM1375	Розпочато обмін даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1376	Розпочато підключення 1 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1377	Розпочато підключення 2 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1378	Розпочато підключення 3 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1379	Розпочато підключення 4 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1380	Розпочато підключення 5 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1381	Розпочато підключення 6 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1382	Розпочато підключення 7 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1383	Розпочато підключення 8 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SM1384	Розпочато підключення 9 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1385	Розпочато підключення 10 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1386	Розпочато з'єднання 11 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1387	Розпочато підключення 12 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1388	Розпочато підключення 13 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1389	Розпочато з'єднання 14 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1390	Розпочато з'єднання 15 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1391	Розпочато підключення 16 для обміну даними через EtherNet/IP (сканер).	○	○	OFF	—	—	H	R/W	OFF
*SM1392	Розпочато з'єднання 17 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1393	Розпочато підключення 18 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1394	Розпочато підключення 19 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1395	Розпочато підключення 20 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1396	Розпочато з'єднання 21 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1397	Розпочато підключення 22 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1398	Розпочато з'єднання 23 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1399	Розпочато з'єднання 24 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1400	Розпочато підключення 25 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1401	Розпочато підключення 26 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1402	Розпочато з'єднання 27 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1403	Розпочато підключення 28 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1404	Розпочато підключення 29 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1405	Розпочато підключення 30 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1406	Розпочато підключення 31 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
*SM1407	Розпочато підключення 32 для обміну даними через EtherNet/IP (сканер).	○	—	OFF	—	—	H	R/W	OFF
*SM1408	Помилка підключення до обміну даними 1 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1409	Помилка підключення до обміну даними 2 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1410	Помилка підключення до обміну даними 3 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1411	Помилка підключення обміну даними 4 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1412	Помилка підключення обміну даними 5 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1413	Помилка підключення обміну даними 6 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1414	Помилка підключення обміну даними 7 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1415	Помилка підключення до обміну даними 8 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1416	Помилка підключення обміну даними 9 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1417	Помилка підключення обміну даними 10 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1418	Помилка підключення обміну даними 11 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1419	Помилка підключення обміну даними 12 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1420	Помилка підключення обміну даними 13 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1421	Помилка підключення обміну даними 14 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1422	Помилка підключення обміну даними 15 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1423	Помилка підключення обміну даними 16 через EtherNet/IP (сканер)	○	○	OFF	—	—	N	R	OFF
*SM1424	Помилка підключення обміну даними 17 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1425	Помилка підключення обміну даними 18 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1426	Помилка підключення обміну даними 19 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1427	Помилка підключення до обміну даними 20 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1428	Помилка підключення обміну даними 21 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1429	Помилка підключення обміну даними 22 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SM1430	Помилка підключення обміну даними 23 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1431	Помилка підключення обміну даними 24 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1432	Помилка підключення обміну даними 25 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1433	Помилка підключення обміну даними 26 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1434	Помилка підключення обміну даними 27 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1435	Помилка підключення до обміну даними 28 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1436	Помилка підключення обміну даними 29 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1437	Помилка підключення до обміну даними 30 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1438	Помилка підключення обміну даними 31 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1439	Помилка підключення до обміну даними 32 через EtherNet/IP (сканер)	○	—	OFF	—	—	N	R	OFF
*SM1440	Помилка в підключенні I/O 1 через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1441	Помилка в підключенні I/O 2 через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1442	Помилка в підключенні I/O 3 через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1443	Помилка підключення введення/виведення 4 через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1444	Помилка підключення вводу/виводу 5 через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1445	Помилка в підключенні I/O 6 через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1446	Помилка підключення введення/виведення 7 через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1447	Помилка підключення вводу-виводу 8 через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1448	З'єднання I/O 1 встановлюється через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1449	Підключення I/O 2 встановлюється через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1450	З'єднання I/O 3 встановлюється через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1451	З'єднання введення/виведення 4 встановлюється через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF	STOP	RUN	Засув	Атрибут	За замовч.
				↕ ON	↕ RUN	↕ STOP			
*SM1452	З'єднання введення/виведення 5 встановлюється через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1453	З'єднання I/O 6 встановлюється через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1454	Підключення I/O 7 встановлюється через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
*SM1455	З'єднання введення/виведення 8 встановлюється через EtherNet/IP (адаптер)	○	○	OFF	—	—	N	R	OFF
SM1581	Автоматичне скидання керування через код зв'язку CAN 1 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання керування через ідентифікатор зв'язку ECAT 1 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1582	Автоматичне скидання керування через код зв'язку CAN 2 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання керування через ідентифікатор зв'язку ECAT 2 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1583	Автоматичне скидання керування через код зв'язку CAN 3 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання керування через зв'язок ECAT ID 3 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1584	Автоматичне скидання керування через код зв'язку CAN 4 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання керування через зв'язок ECAT ID 4 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1585	Автоматичне скидання керування через код зв'язку CAN 5 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання контролю над зв'язком ECAT ID 5 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1586	Автоматичне скидання керування через код зв'язку CAN 6 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання керування через зв'язок ECAT ID 6 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1587	Автоматичне скидання керування через ідентифікатор зв'язку CAN 7 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання керування через зв'язок ECAT ID 7 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1588	Автоматичне скидання керування через код зв'язку CAN 8 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання керування через ідентифікатор зв'язку ECAT 8 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1601	Автоматичне скидання керування через код зв'язку CAN 21 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Автоматичне скидання керування через зв'язок ECAT ID 9 сервоінвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM1602	Автоматичне скидання керування через код зв'язку CAN 22 сервоприводу Delta	○	○	OFF	OFF	–	N	R/W	OFF
	Автоматичне скидання керування через зв'язок ECAT ID 10 сервоінвертора Delta	○	–	OFF	OFF	–	N	R/W	OFF
SM1603	Автоматичне скидання керування через код зв'язку CAN 23 сервоприводу Delta	○	○	OFF	OFF	–	N	R/W	OFF
	Автоматичне скидання керування через ідентифікатор зв'язку ECAT 11 сервоінвертора Delta	○	–	OFF	OFF	–	N	R/W	OFF
SM1604	Автоматичне скидання керування через код зв'язку CAN 24 сервоприводу Delta	○	○	OFF	OFF	–	N	R/W	OFF
	Автоматичне скидання керування через ідентифікатор зв'язку ECAT 12 сервоінвертора Delta	○	–	OFF	OFF	–	N	R/W	OFF
SM1605	Автоматичне скидання керування через код зв'язку CAN 25 сервоприводу Delta	○	○	OFF	OFF	–	N	R/W	OFF
	Автоматичне скидання керування через зв'язок ECAT ID 13 сервоінвертора Delta	○	–	OFF	OFF	–	N	R/W	OFF
SM1606	Автоматичне скидання керування через код зв'язку CAN 26 сервоприводу Delta	○	○	OFF	OFF	–	N	R/W	OFF
	Автоматичне скидання керування через ідентифікатор зв'язку ECAT 14 сервоінвертора Delta	○	–	OFF	OFF	–	N	R/W	OFF
SM1607	Автоматичне скидання керування через код зв'язку CAN 27 сервоприводу Delta	○	○	OFF	OFF	–	N	R/W	OFF
	Автоматичне скидання керування через ідентифікатор зв'язку ECAT 15 сервоінвертора Delta	○	–	OFF	OFF	–	N	R/W	OFF
SM1608	Автоматичне скидання керування через код зв'язку CAN 28 сервоприводу Delta.	○	○	OFF	OFF	–	N	R/W	OFF
	Автоматичне скидання керування через ідентифікатор зв'язку ECAT 16 сервоінвертора Delta	○	–	OFF	OFF	–	N	R/W	OFF
SM1611	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 21 інвертора Delta	○	○	OFF	OFF	–	N	R	OFF
	З'єднання втрачено на зв'язку ECAT ID 9 сервоінвертора Delta	○	–	OFF	OFF	–	N	R	OFF
SM1612	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 22 інвертора Delta	○	○	OFF	OFF	–	N	R	OFF
	З'єднання втрачено на зв'язку ECAT ID 10 сервоінвертора Delta	○	–	OFF	OFF	–	N	R	OFF
SM1613	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 23 інвертора Delta	○	○	OFF	OFF	–	N	R	OFF
	Втрачено з'єднання з ідентифікатором зв'язку ECAT 11 сервоінвертора Delta	○	–	OFF	OFF	–	N	R	OFF
SM1614	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 24 інвертора Delta	○	○	OFF	OFF	–	N	R	OFF
	Втрачено з'єднання з ідентифікатором зв'язку ECAT 12 сервоінвертора Delta	○	–	OFF	OFF	–	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
SM1615	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 25 інвертора Delta	○	○	OFF	OFF	–	N	R	OFF
	Втрачено з'єднання з ідентифікатором зв'язку ECAT 13 серво/інвертора Delta	○	–	OFF	OFF	–	N	R	OFF
SM1616	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 26 інвертора Delta	○	○	OFF	OFF	–	N	R	OFF
	Втрачено з'єднання з ідентифікатором зв'язку ECAT 14 серво/інвертора Delta	○	–	OFF	OFF	–	N	R	OFF
SM1617	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 27 інвертора Delta	○	○	OFF	OFF	–	N	R	OFF
	Втрачено з'єднання з ідентифікатором зв'язку ECAT 15 серво/інвертора Delta	○	–	OFF	OFF	–	N	R	OFF
SM1618	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 28 інвертора Delta	○	○	OFF	OFF	–	N	R	OFF
	Втрачено з'єднання з ідентифікатором зв'язку ECAT 16 серво/інвертора Delta	○	–	OFF	OFF	–	N	R	OFF
SM1621	Ідентифікатор зв'язку CAN 21 інвертора Delta запускається.	○	○	OFF	OFF	–	N	R	OFF
	Ідентифікатор зв'язку ECAT 9 інвертора серво/інвертора Delta запускається.	○	–	OFF	OFF	–	N	R	OFF
SM1622	Ідентифікатор зв'язку CAN 22 інвертора Delta запускається.	○	○	OFF	OFF	–	N	R	OFF
	Ідентифікатор зв'язку ECAT 10 інвертора серво/інвертора Delta запускається.	○	–	OFF	OFF	–	N	R	OFF
SM1623	Ідентифікатор зв'язку CAN 23 інвертора Delta запускається.	○	○	OFF	OFF	–	N	R	OFF
	Ідентифікатор зв'язку ECAT 11 інвертора серво/інвертора Delta запускається.	○	–	OFF	OFF	–	N	R	OFF
SM1624	Ідентифікатор зв'язку CAN 24 інвертора Delta запускається.	○	○	OFF	OFF	–	N	R	OFF
	Ідентифікатор зв'язку ECAT 12 інвертора серво/інвертора Delta запускається.	○	–	OFF	OFF	–	N	R	OFF
SM1625	Ідентифікатор зв'язку CAN 25 інвертора Delta запускається.	○	○	OFF	OFF	–	N	R	OFF
	Ідентифікатор зв'язку ECAT 13 інвертора серво/інвертора Delta запускається.	○	–	OFF	OFF	–	N	R	OFF
SM1626	Ідентифікатор зв'язку CAN 26 інвертора Delta запускається.	○	○	OFF	OFF	–	N	R	OFF
	Ідентифікатор зв'язку ECAT 14 інвертора серво/інвертора Delta запускається.	○	–	OFF	OFF	–	N	R	OFF
SM1627	Ідентифікатор зв'язку CAN 27 інвертора Delta запускається.	○	○	OFF	OFF	–	N	R	OFF
	Ідентифікатор зв'язку ECAT 15 інвертора серво/інвертора Delta запускається.	○	–	OFF	OFF	–	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF	STOP	RUN	Засув	Атрибут	За замовч.
				↕ ON	↕ RUN	↕ STOP			
SM1628	Ідентифікатор зв'язку CAN 28 інвертора Delta запускається.	○	○	OFF	OFF	—	N	R	OFF
	Ідентифікатор зв'язку ECAT 16 інвертора серво/інвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1631	Позиціонування завершено за ідентифікатором зв'язку CAN 1 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Позиціонування завершено за ідентифікатором зв'язку ECAT 1 сервоприводу Delta	○	—	OFF	OFF	—	N	R	OFF
SM1632	Позиціонування завершено за ідентифікатором зв'язку CAN 2 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Позиціонування завершено за ідентифікатором зв'язку ECAT 2 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1633	Позиціонування завершено за ідентифікатором зв'язку CAN 3 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Позиціонування завершено за ідентифікатором зв'язку ECAT 3 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1634	Позиціонування завершено за ідентифікатором зв'язку CAN 4 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Позиціонування завершено за ідентифікатором зв'язку ECAT 4 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1635	Позиціонування завершено за ідентифікатором зв'язку CAN 5 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Позиціонування завершено за ідентифікатором зв'язку ECAT 5 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1636	Позиціонування завершено за ідентифікатором зв'язку CAN 5 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Позиціонування завершено за ідентифікатором зв'язку ECAT 6 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1637	Позиціонування завершено за ідентифікатором зв'язку CAN 7 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Позиціонування завершено за ідентифікатором зв'язку ECAT 7 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1638	Позиціонування завершено за ідентифікатором зв'язку CAN 8 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Позиціонування завершено за ідентифікатором зв'язку ECAT 8 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1641	Зв'язок припиняється на ідентифікаторі зв'язку CAN 1 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Зв'язок припиняється на ECAT зв'язку ID 1 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1642	Зв'язок припиняється на ідентифікаторі зв'язку CAN 2 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Зв'язок припиняється на ECAT зв'язку ID 2 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1643	Зв'язок припиняється на ідентифікаторі зв'язку CAN 3 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
	Зв'язок припиняється на ECAT зв'язку ID 3 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1644	Зв'язок припиняється за ідентифікатором зв'язку CAN 4 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Зв'язок припиняється на зв'язку ECAT ID 4 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1645	Зв'язок припиняється на ідентифікаторі зв'язку CAN 5 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Зв'язок припиняється на зв'язку ECAT ID 5 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1646	Зв'язок припиняється за ідентифікатором зв'язку CAN 6 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Зв'язок припиняється на зв'язку ECAT ID 6 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1647	Зв'язок припиняється за ідентифікатором зв'язку CAN 7 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Зв'язок припиняється на зв'язку ECAT ID 7 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1648	Зв'язок припиняється за ідентифікатором зв'язку CAN 8 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Зв'язок припиняється на зв'язку ECAT ID 8 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1651	Ідентифікатор зв'язку CAN 1 сервоприводу Delta запускається.	○	○	OFF	OFF	—	N	R	OFF
	Ідентифікатор зв'язку ECAT 1 серво/інвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1652	Ідентифікатор зв'язку CAN 2 сервоприводу Delta запускається.	○	○	OFF	OFF	—	N	R	OFF
	Ідентифікатор зв'язку ECAT 2 серво/інвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1653	Ідентифікатор зв'язку CAN 3 сервоприводу Delta запускається.	○	○	OFF	OFF	—	N	R	OFF
	Ідентифікатор зв'язку ECAT 3 серво/інвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1654	Ідентифікатор зв'язку CAN 4 сервоприводу Delta запускається.	○	○	OFF	OFF	—	N	R	OFF
	Ідентифікатор зв'язку ECAT 4 серво/інвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1655	Ідентифікатор зв'язку CAN 5 сервоприводу Delta запускається.	○	○	OFF	OFF	—	N	R	OFF
	Ідентифікатор зв'язку ECAT 5 серво/інвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1656	Ідентифікатор зв'язку CAN 6 сервоприводу Delta запускається.	○	○	OFF	OFF	—	N	R	OFF
	Ідентифікатор зв'язку ECAT 6 інвертора серво/інвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1657	Ідентифікатор зв'язку CAN 7 сервоприводу Delta запускається.	○	○	OFF	OFF	—	N	R	OFF
	Ідентифікатор зв'язку ECAT 7 серво/інвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1658	Ідентифікатор зв'язку CAN 8 сервоприводу Delta запускається.	○	○	OFF	OFF	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
2	Ідентифікатор зв'язку ECAT 8 сервоінвертора Delta запускається.	○	—	OFF	OFF	—	N	R	OFF
SM1661	Функція переміщення вперед і назад увімкнена на CAN зв'язку ID 1 сервоприводу Delta.	○	○	OFF	OFF	—	N	R/W	OFF
	Функція переміщення вперед і назад увімкнена на ECAT зв'язку ID 1 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1662	Функція переміщення вперед і назад увімкнена на CAN зв'язку ID 2 сервоприводу Delta.	○	○	OFF	OFF	—	N	R/W	OFF
	Функція переміщення вперед і назад увімкнена на ECAT зв'язку ID 2 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1663	Функція переміщення вперед і назад увімкнена на CAN зв'язку ID 3 сервоприводу Delta.	○	○	OFF	OFF	—	N	R/W	OFF
	Функція переміщення вперед і назад увімкнена на ECAT зв'язку ID 3 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1664	Функція переміщення вперед і назад увімкнена на CAN зв'язку ID 4 сервоприводу Delta.	○	○	OFF	OFF	—	N	R/W	OFF
	Функція переміщення вперед і назад увімкнена на ECAT зв'язку ID 4 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1665	Функція переміщення вперед і назад увімкнена на CAN зв'язку ID 5 сервоприводу Delta.	○	○	OFF	OFF	—	N	R/W	OFF
	Функція переміщення вперед і назад увімкнена на комунікаційному ідентифікаторі ECAT 5 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1666	Функція переміщення вперед і назад увімкнена на CAN зв'язку ID 6 сервоприводу Delta.	○	○	OFF	OFF	—	N	R/W	OFF
	Функція переміщення вперед і назад увімкнена на комунікаційному ідентифікаторі ECAT 6 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1667	Функція переміщення вперед і назад увімкнена на CAN зв'язку ID 7 сервоприводу Delta.	○	○	OFF	OFF	—	N	R/W	OFF
	Функція переміщення вперед і назад увімкнена на комунікаційному ідентифікаторі ECAT 7 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1668	Функція переміщення вперед і назад увімкнена на CAN зв'язку ID 8 сервоприводу Delta.	○	○	OFF	OFF	—	N	R/W	OFF
	Функція переміщення вперед і назад увімкнена на ECAT зв'язку ID 8 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1671	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку CAN 1 сервоприводу Delta.	○	○	OFF	OFF	—	N	R	OFF
	Прапорець індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 1 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1672	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку CAN 2 сервоприводу Delta.	○	○	OFF	OFF	—	N	R	OFF
	Прапор індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 2 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1673	Прапорець індикації напрямку руху назад/вперед на ідентифікаторі зв'язку CAN 3 сервоприводу Delta.	○	○	OFF	OFF	—	N	R	OFF
	Прапорець індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 3 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SM1674	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку CAN 4 сервоприводу Delta.	○	○	OFF	OFF	—	N	R	OFF
	Прапорець індикації напрямку руху назад/вперед на ідентифікаторі зв'язку ECAT 4 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1675	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку CAN 5 сервоприводу Delta.	○	○	OFF	OFF	—	N	R	OFF
	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку ECAT 5 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1676	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку CAN 6 сервоприводу Delta.	○	○	OFF	OFF	—	N	R	OFF
	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку ECAT 6 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1677	Прапорець індикації напрямку руху назад/вперед на ідентифікаторі зв'язку CAN 7 сервоприводу Delta.	○	○	OFF	OFF	—	N	R	OFF
	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку ECAT 7 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1678	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку CAN 8 сервоприводу Delta.	○	○	OFF	OFF	—	N	R	OFF
	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку ECAT 8 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1681	Інструкція INITC/CASD прапор завершення зв'язку CAN сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Прапор завершення інструкції CASD для зв'язку ECAT між серво/інвертором Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1682	Прапор помилки зв'язку CAN для серво/інвертора Delta	○	○	OFF	OFF	—	N	R/W	OFF
	Прапор помилки зв'язку ECAT для серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1683	Інструкція INITC/CASD прапор завершення зв'язку CAN інвертора Delta	○	○	OFF	OFF	—	N	R/W	OFF
SM1684	Щоб встановити дію для зв'язку CAN сервоприводу/інвертора Delta, коли з'єднання втрачено: OFF.: припиняє всі зв'язки. УБІМК.: припиняє лише зв'язок із втраченим з'єднанням	○	○	OFF	OFF	—	N	R/W	OFF
	Щоб встановити дію для зв'язку ECAT серво/інвертора Delta, коли з'єднання втрачено: OFF.: припиняє всі зв'язки. УБІМК.: припиняє лише зв'язок із втраченим з'єднанням	○	—	OFF	OFF	—	N	R/W	OFF
SM1685	Для зчитування самостійно визначених параметрів Delta сервоприводу P0-12	○	○	OFF	OFF	—	N	R/W	OFF
SM1686	OFF.: ініціалізація послідовної кількості модулів. УБІМК.: ініціалізація певного модуля	○	○	OFF	OFF	—	N	R/W	OFF
SM1691	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 1 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
	З'єднання втрачено на зв'язку ECAT ID 1 серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1692	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 2 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	З'єднання втрачено на зв'язку ECAT ID 2 серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1693	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 3 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	З'єднання втрачено на зв'язку ECAT ID 3 серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1694	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 4 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	З'єднання втрачено на зв'язку ECAT ID 4 серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1695	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 5 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	З'єднання втрачено на зв'язку ECAT ID 5 серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1696	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 6 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	З'єднання втрачено на зв'язку ECAT ID 6 серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1697	Помилка серцевого ритму на зв'язку CAN ID 7 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	З'єднання втрачено на зв'язку ECAT ID 7 серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1698	Помилка серцевого ритму на ідентифікаторі зв'язку CAN 8 сервоприводу Delta	○	○	OFF	OFF	—	N	R/W	OFF
	З'єднання втрачено на зв'язку ECAT ID 8 серво/інвертора Delta	○	—	OFF	OFF	—	N	R/W	OFF
*SM1700	Прапор для активації компенсації люфту на Y0.0/Y0.1 для інструкції CSFO	○	○	OFF	OFF	—	N	R/W	OFF
*SM1701	Позначка для активації компенсації люфту на Y0.2/Y0.3 для інструкції CSFO	○	○	OFF	OFF	—	N	R/W	OFF
*SM1702	Позначка для активації компенсації люфту на Y0.4/Y0.5 для інструкції CSFO	○	○	OFF	OFF	—	N	R/W	OFF
*SM1703	Позначка для активації компенсації люфту на Y0.6/Y0.7 для інструкції CSFO	○	○	OFF	OFF	—	N	R/W	OFF
*SM1704	Прапор для активації компенсації люфту на Y0.8/Y0.9 для інструкції CSFO	○	○	OFF	OFF	—	N	R/W	OFF
*SM1705	Позначка для активації компенсації люфту на Y0.10/Y0.11 для інструкції CSFO	○	○	OFF	OFF	—	N	R/W	OFF
* SM1709	Помилка ініціалізації обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ НА	СТОП ↓ БІГУ	БІГ ↓ СТОП	Засув	Атрибут	За замовч.
* SM1710	Обмін даними через функціональну картку 1 увімкнено ISPSOft	○	–	OFF	–	OFF.	N	R/W	OFF
* SM1712	З'єднання 1 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1713	Connection 2 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1714	Connection 3 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1715	Connection 4 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1716	Connection 5 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1717	Connection 6 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1718	Connection 7 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1719	Connection 8 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1720	Connection 9 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1721	Connection 10 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1722	Connection 11 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1723	З'єднання 12 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1724	Connection 13 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1725	Connection 14 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1726	Connection 15 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1727	З'єднання 16 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1728	Connection 17 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1729	З'єднання 18 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1730	Connection 19 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1731	Connection 20 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1732	Connection 21 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF
* SM1733	Connection 22 для обміну даними через функцію Розпочато картку 1	○	–	OFF	–	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
* SM1734	Розпочато підключення 23 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1735	Розпочато підключення 24 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1736	Розпочато підключення 25 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1737	Розпочато з'єднання 26 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1738	Розпочато підключення 27 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1739	Розпочато підключення 28 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1740	Розпочато підключення 29 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1741	Розпочато з'єднання 30 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1742	Розпочато підключення 31 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1743	Розпочато з'єднання 32 для обміну даними через функціональну картку 1	○	—	OFF	—	—	N	R/W	OFF
* SM1744	Успішне з'єднання для обміну даними 1 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1745	Успішне з'єднання для обміну даними 2 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1746	Успішне з'єднання для обміну даними 3 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1747	Успішне з'єднання для обміну даними 4 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1748	Успішне з'єднання для обміну даними 5 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1749	Успішне з'єднання для обміну даними 6 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1750	Успішне з'єднання для обміну даними 7 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1751	Успішне з'єднання для обміну даними 8 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1752	Успішне з'єднання для обміну даними 9 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1753	Успішне з'єднання для обміну даними 10 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1754	Успішне з'єднання для обміну даними 11 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1755	Успішне з'єднання для обміну даними 12 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1756	Успішне з'єднання для обміну даними 13 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ НА	СТОП ↓ БІГУ	БІГ ↓ СТОП	Засув	Атрибут	За замовч.
* SM1757	Успішне з'єднання для обміну даними через 14 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1758	Успішне з'єднання для обміну даними через 15 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1759	Успішне з'єднання для обміну даними через 16 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1760	Успішне з'єднання для обміну даними через 17 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1761	Успішне з'єднання для обміну даними через 18 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1762	Успішне з'єднання для обміну даними через 19 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1763	Успішне з'єднання для обміну даними через 20 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1764	Успішне з'єднання для обміну даними через 21 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1765	Успішне з'єднання для обміну даними через 22 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1766	Успішне з'єднання для обміну даними через 23 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1767	Успішне з'єднання для обміну даними через 24 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1768	Успішне з'єднання для обміну даними через 25 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1769	Успішне з'єднання для обміну даними через 26 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1770	Успішне з'єднання для обміну даними через 27 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1771	Успішне з'єднання для обміну даними через 28 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1772	Успішне з'єднання для обміну даними через 29 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1773	Успішне з'єднання для обміну даними через 30 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1774	Успішне з'єднання для обміну даними через 31 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1775	Успішне з'єднання для обміну даними через 32 Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1776	Помилка підключення до обміну даними 1 через Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1777	Помилка підключення обміну даними 2 через Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1778	Помилка підключення до обміну даними 3 через Функціональна картка 1	○	—	OFF	—	—	N	R	OFF
* SM1779	Помилка підключення обміну даними 4 через Функціональна картка 1	○	—	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
* SM1780	Помилка підключення обміну даними 5 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1781	Помилка підключення обміну даними 6 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1782	Помилка підключення обміну даними 7 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1783	Помилка підключення обміну даними 8 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1784	Помилка підключення обміну даними 9 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1785	Помилка підключення обміну даними 10 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1786	Помилка підключення обміну даними 11 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1787	Помилка підключення обміну даними 12 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1788	Помилка підключення обміну даними 13 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1789	Помилка підключення обміну даними 14 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1790	Помилка підключення обміну даними 15 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1791	Помилка підключення обміну даними 16 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1792	Помилка підключення обміну даними 17 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1793	Помилка підключення обміну даними 18 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1794	Помилка підключення обміну даними 19 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1795	Помилка підключення обміну даними 20 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1796	Помилка підключення обміну даними 21 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1797	Помилка підключення обміну даними 22 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1798	Помилка підключення обміну даними 23 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1799	Помилка підключення обміну даними 24 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1800	Помилка підключення обміну даними 25 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1801	Помилка підключення обміну даними 26 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF
* SM1802	Помилка підключення обміну даними 27 через функціональну картку 1	○	—	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ НА	СТОП ↓ БІГУ	БІГ ↓ СТОП	Засув	Атрибут	За замовч.
* SM1803	Помилка підключення до обміну даними 28 через Функціональна картка 1	○	–	OFF	–	–	N	R	OFF
* SM1804	Помилка підключення до обміну даними 29 через Функціональна картка 1	○	–	OFF	–	–	N	R	OFF
* SM1805	Помилка підключення до обміну даними 30 через Функціональна картка 1	○	–	OFF	–	–	N	R	OFF
* SM1806	Помилка підключення до обміну даними 31 через Функціональна картка 1	○	–	OFF	–	–	N	R	OFF
* SM1807	Помилка підключення до обміну даними 32 через Функціональна картка 1	○	–	OFF	–	–	N	R	OFF
* SM1821	Помилка під час ініціалізації обміну даними через функціональну картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1822	Обмін даними через функціональну картку 2 увімкнено ISPSOft	○	–	OFF	–	OFF.	N	R/W	OFF
* SM1824	З'єднання 1 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1825	Connection 2 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1826	Connection 3 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1827	Connection 4 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1828	Connection 5 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1829	Connection 6 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1830	Connection 7 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1831	Connection 8 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1832	Connection 9 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1833	Connection 10 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1834	Connection 11 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1835	Connection 12 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1836	Connection 13 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1837	Connection 14 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1838	Connection 15 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF
* SM1839	З'єднання 16 для обміну даними через функцію Розпочато картку 2	○	–	OFF	–	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
* SM1840	Розпочато підключення 17 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1841	Розпочато підключення 18 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1842	Розпочато підключення 19 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1843	Розпочато підключення 20 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1844	Розпочато підключення 21 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1845	Розпочато підключення 22 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1846	Розпочато підключення 23 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1847	Розпочато підключення 24 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1848	Розпочато підключення 25 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1849	Розпочато підключення 26 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1850	Розпочато підключення 27 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1851	Розпочато підключення 28 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1852	Розпочато з'єднання 29 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1853	Розпочато з'єднання 30 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1854	Розпочато підключення 31 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1855	Розпочато з'єднання 32 для обміну даними через функціональну картку 2	○	—	OFF	—	—	N	R/W	OFF
* SM1856	Успішне з'єднання для обміну даними 1 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1857	Успішне з'єднання для обміну даними 2 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1858	Успішне з'єднання для обміну даними 3 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1859	Успішне з'єднання для обміну даними 4 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1860	Успішне з'єднання для обміну даними 5 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1861	Успішне з'єднання для обміну даними 6 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1862	Успішне з'єднання для обміну даними 7 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ НА	СТОП ↓ БІГУ	БІГ ↓ СТОП	Засув	Атрибут	За замовч.
* SM1863	Успішне з'єднання для обміну даними через 8 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1864	Успішне з'єднання для обміну даними через 9 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1865	Успішне з'єднання для обміну даними через 10 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1866	Успішне з'єднання для обміну даними через 11 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1867	Успішне з'єднання для обміну даними через 12 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1868	Успішне з'єднання для обміну даними через 13 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1869	Успішне з'єднання для обміну даними через 14 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1870	Успішне з'єднання для обміну даними через 15 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1871	Успішне з'єднання для обміну даними через 16 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1872	Успішне з'єднання для обміну даними через 17 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1873	Успішне з'єднання для обміну даними через 18 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1874	Успішне з'єднання для обміну даними через 19 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1875	Успішне з'єднання для обміну даними через 20 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1876	Успішне з'єднання для обміну даними через 21 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1877	Успішне з'єднання для обміну даними через 22 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1878	Успішне з'єднання для обміну даними через 23 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1879	Успішне з'єднання для обміну даними через 24 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1880	Успішне з'єднання для обміну даними через 25 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1881	Успішне з'єднання для обміну даними через 26 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1882	Успішне з'єднання для обміну даними через 27 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1883	Успішне з'єднання для обміну даними через 28 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1884	Успішне з'єднання для обміну даними через 29 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF
* SM1885	Успішне з'єднання для обміну даними через 30 Функціональна картка 2	○	—	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF	STOP	RUN	Засув	Атрибут	За замовч.
				↓ ON	↓ RUN	↓ STOP			
* SM1886	Успішне з'єднання для обміну даними 31 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1887	Успішне з'єднання для обміну даними 32 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1888	Помилка підключення обміну даними 1 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1889	Помилка підключення обміну даними 2 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1890	Помилка підключення обміну даними 3 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1891	Помилка підключення обміну даними 4 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1892	Помилка підключення обміну даними 5 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1893	Помилка підключення обміну даними 6 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1894	Помилка підключення обміну даними 7 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1895	Помилка підключення обміну даними 8 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1896	Помилка підключення обміну даними 9 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1897	Помилка підключення обміну даними 10 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1898	Помилка підключення обміну даними 11 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1899	Помилка підключення обміну даними 12 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1900	Помилка підключення обміну даними 13 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1901	Помилка підключення обміну даними 14 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1902	Помилка підключення обміну даними 15 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1903	Помилка підключення обміну даними 16 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1904	Помилка підключення обміну даними 17 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1905	Помилка підключення обміну даними 18 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1906	Помилка підключення обміну даними 19 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1907	Помилка підключення обміну даними 20 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF
* SM1908	Помилка підключення обміну даними 21 через функціональну картку 2	○	—	OFF	—	—	N	R	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ НА	СТОП ↓ БІГУ	БІГ ↓ СТОП	Засув	Атрибут	За замовч.
* SM1909	Помилка підключення до обміну даними 22 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1910	Помилка підключення до обміну даними 23 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1911	Помилка підключення до обміну даними 24 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1912	Помилка підключення до обміну даними 25 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1913	Помилка підключення до обміну даними 26 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1914	Помилка підключення до обміну даними 27 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1915	Помилка підключення до обміну даними 28 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1916	Помилка підключення до обміну даними 29 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1917	Помилка підключення обміну даними 30 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1918	Помилка підключення до обміну даними 31 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
* SM1919	Помилка підключення до обміну даними 32 через Функціональна картка 2	○	–	OFF	–	–	N	R	OFF
SM1921	Позиціонування завершено за ідентифікатором зв'язку ECAT 9 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1922	Позиціонування завершено за ідентифікатором зв'язку ECAT 10 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1923	Позиціонування завершено за ідентифікатором зв'язку ECAT 11 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1924	Позиціонування завершено за ідентифікатором зв'язку ECAT 12 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1925	Позиціонування завершено за ідентифікатором зв'язку ECAT 13 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1926	Позиціонування завершено за ідентифікатором зв'язку ECAT 14 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1927	Позиціонування завершено за ідентифікатором зв'язку ECAT 15 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1928	Позиціонування завершено за ідентифікатором зв'язку ECAT 16 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1931	Зв'язок припиняється на зв'язку ECAT ID 9 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1932	Зв'язок припиняється на ECAT зв'язку ID 10 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1933	Зв'язок припиняється на ECAT зв'язку ID 11 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF
SM1934	Зв'язок припиняється на ECAT зв'язку ID 12 сервоприводу Delta	○	–	OFF.	OFF	–	N	R/W	OFF

SM	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SM1935	Зв'язок припиняється на ECAT зв'язку ID 13 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1936	Зв'язок припиняється на ECAT зв'язку ID 14 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1937	Зв'язок припиняється на ECAT зв'язку ID 15 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1938	Зв'язок припиняється на ECAT зв'язку ID 16 сервоприводу Delta	○	—	OFF	OFF	—	N	R/W	OFF
SM1941	Функція переміщення вперед і назад увімкнена на комунікаційному ідентифікаторі ECAT 9 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1942	Функція переміщення вперед і назад увімкнена на ECAT зв'язку ID 10 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1943	Функція переміщення вперед і назад увімкнена за ідентифікатором зв'язку ECAT 11 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1944	Функція переміщення вперед і назад увімкнена на ECAT зв'язку ID 12 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1945	Функція переміщення вперед і назад увімкнена на ECAT зв'язку ID 13 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1946	Функція переміщення туди-сюди ввімкнена за ідентифікатором зв'язку ECAT 14 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1947	Функція переміщення вперед і назад увімкнена на комунікаційному ідентифікаторі ECAT 15 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1948	Функція переміщення вперед і назад увімкнена за ідентифікатором зв'язку ECAT 16 сервоприводу Delta.	○	—	OFF	OFF	—	N	R/W	OFF
SM1951	Прапорець індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 9 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1952	Прапорець індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 10 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1953	Прапор індикації напрямку руху назад/вперед на ідентифікаторі зв'язку ECAT 11 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1954	Прапорець індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 12 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1955	Прапорець індикації напрямку руху назад/вперед на ідентифікаторі зв'язку ECAT 13 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1956	Прапорець індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 14 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1957	Прапорець індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 15 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF
SM1958	Прапор індикації напрямку руху назад/ вперед на ідентифікаторі зв'язку ECAT 16 сервоприводу Delta.	○	—	OFF	OFF	—	N	R	OFF

*1: для елементів зі знаком * зверніться до розділу 2.2.1.6 для додаткових зауважень щодо спеціальних допоміжних реле та спеціальних регістрів даних.

*2: Система працює відповідно до параметрів, встановлених у HWCONFIG. Коли живлення SM змінюється з OFF. на УВІМК., стан -, а зафіксована область позначається як N.

*3: Функціональна карта тут означає AS-F232, AS-F422 і AS-F485.

2.2.8 Час оновлення для спеціальних допоміжних реле

Спеціальне допоміжне реле	Час оновлення
SM0–SM1	Система автоматично встановлює прапорець на ON і скидає його на OFF. ON: помилка операції.
SM5	Система автоматично встановлює прапорець на ON і скидає його на OFF. ON: виникає помилка, коли програма записується в ПЛК.
SM6	Під час увімкнення живлення система перевіряє, чи не було втрачено дані в заблокованій області. ON: дані в зафіксованій області втрачено. Ви скидаєте його на OFF.
SM7	ON: живлення (24 В) недостатнє. Ви скидаєте його на OFF.
SM8	Система автоматично встановлює SM8 на ON і скидає його на OFF. ON: виникла помилка сторожового таймера.
SM9	Система автоматично встановлює SM9 на ON і скидає його на OFF. ON: виникла системна помилка.
SM10	Система автоматично встановлює SM10 на ON і скидає його на OFF. ON: виникла помилка шини введення/виведення.
SM22–SM24	Ви встановлюєте прапорець на ON, і система автоматично скидає його на OFF. ON: журнал очищено.
SM25–SM26	ON: користувачі редагують за допомогою ISPSOft. OFF: користувачі вийшли з ISPSOft.
SM27	Оновить значення у вбудованому AD для AS-F2AD. Система скидає його на OFF.
SM28	Система перевіряє, чи є щось несправне. ON: щось не так. Ви скидаєте його на OFF.
SM29	Система автоматично встановлює прапорець на ON. Ви скидаєте його на OFF.
SM30	ON: у віддаленому модулі сталася помилка. Система скидає його на OFF.
SM34	ON: введено неправильний пароль. Система скидає його на OFF.
SM36	ON: система зберігає дані на карту пам'яті. Після завершення збереження система автоматично скидає його до OFF. Ви встановлюєте його на ON, щоб увімкнути збереження.
SM38	Система автоматично встановлює прапорець на ON. Ви скидаєте його на OFF.
SM76–SM77	ON: система виконує комунікаційне завдання. Після завершення комунікаційного завдання система автоматично скидає його до OFF. Ви встановлюєте значення ON, щоб увімкнути виконання.
SM78–SM79	ON: зв'язок триває. Після завершення зв'язку система автоматично скидає його на OFF.
SM80–SM81	ON: прийом завершено. Ви скидаєте його на OFF.
SM82–SM83	ON: у відповіді виникає помилка. Ви скидаєте його на OFF.
SM84–SM85	ON: відбувається тайм-аут. Ви скидаєте його на OFF.
SM86–SM87	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: 8-бітний режим OFF.: 16-бітний режим
SM90–SM91	Ви встановлюєте прапор на ON. Після зміни протоколу зв'язку система скидає його на OFF.
SM94–SM95	Після ввімкнення живлення прапорець увімкнено або вимкнено відповідно до налаштувань у HWCONFIG. Ви можете змінити це налаштування.
SM96–SM97	Ви встановлюєте прапор на ON. Після надсилання даних система автоматично скидає прапор на OFF.
SM98–SM99	ON: спілкування триває. Після завершення зв'язку система автоматично скидає його на OFF.
SM100–SM101	Система автоматично встановлює прапорець на ON, а ви скидаєте його на OFF. ON: команда отримана.
SM102–SM103	Система автоматично встановлює прапорець на ON, а ви скидаєте його на OFF.

Спеціальне допоміжне реле	Час оновлення
	ON: отримана команда неправильна.
SM104–SM105	Система автоматично встановлює прапорець на ON, а ви скидаєте його на OFF. ON: час очікування прийому.
SM106–SM107	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: 8-бітний режим OFF.: 16-бітний режим
SM120–SM151	Ви встановлюєте прапорець на ON, і під час виконання інструкції DPLSY система скидає його на OFF.
SM166–SM167	Ви встановлюєте прапорець на ON і скидаєте його на OFF.
SM168–SM171	Система автоматично встановлює прапорець на ON і скидає його на OFF.
SM172	Система автоматично встановлює прапорець на ON і скидає його на OFF.
SM203	Ви встановлюєте прапорець на ON і після PLC RUN->STOP очищає незафіксовану область. Система автоматично скидає прапор на OFF.
SM204–SM205	Ви встановлюєте прапорець на ON, і система автоматично скидає його на OFF. ON: очистити незафіксовані/зафіксовані області.
SM206	Ви встановлюєте SM206 на ON і скидаєте його на OFF. ON: заборонити весь вихід.
SM209	Ви встановлюєте SM209 на ON, і система автоматично скидає його на OFF. ON: протокол зв'язку COM1 змінюється.
SM210	Ви встановлюєте SM210 на ON і скидаєте його на OFF для COM1. ON: режим RTU OFF.: режим ASCII
SM211	Ви встановлюєте SM211 на ON, і система автоматично скидає його на OFF. ON: протокол зв'язку COM2 змінюється.
SM212	Ви встановлюєте SM210 на ON і скидаєте його на OFF для COM2. ON: режим RTU OFF.: режим ASCII
SM215	Ви встановлюєте SM215 на ON і скидаєте його на OFF. ON: ПЛК працює. OFF.: ПЛК зупиняється.
SM217	Система автоматично встановлює прапорець на ON.
SM218	Під час увімкнення система перевіряє годинник реального часу. ON: помилка годинника реального часу Ви скидаєте його на OFF.
SM219	Система контролює заряд батареї годинника реального часу. ON: потужність годинника реального часу низька. Система скидає його на OFF.
SM220	Ви встановлюєте SM220 на ON і скидаєте його на OFF. ON: калібрування годинника реального часу в межах ± 30 секунд
SM221	Прапор оновлюється відповідно до налаштувань у HWCONFIG або під час виконання інструкції DST (API 1607). ON: інструкція DST виконується.
SM222	Система автоматично встановлює прапорець на ON і скидає його на OFF.
SM230–SM261	Прапор оновлюється відповідно до налаштувань у HWCONFIG; ви можете змінити це налаштування.
SM270–SM275	Прапорець увімкнено, коли виконується інструкція CSFO. ON: увімкнути зміну напрямку введення OFF.: вимкнути зміну напрямку введення
SM281–SM288	Ви встановлюєте прапорець на ON і скидаєте його на OFF.

Спеціальне допоміжне реле	Час оновлення
	ON: зворотний відлік OFF.: підрахунок
SM291–SM296	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: увімкнути очищення точок введення OFF.: вимкнути очищення точок введення
SM300	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM301–SM303	Система встановлює прапор на ON і скидає його на OFF. ON: зворотний відлік OFF.: підрахунок
SM304	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM305–SM307	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM308	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM309–SM311	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM312	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM313–SM315	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM316	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM317–SM319	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM320	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM321–SM323	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM332	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM333–SM335	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM336	Ви встановлюєте прапорець на ON і скидаєте його на OFF.

Спеціальне допоміжне реле	Час оновлення
	ON: зворотний відлік OFF.: підрахунок
SM337–SM339	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM340	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM341	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM342	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM343	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM344	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM345	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM346	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM347	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM348	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM349	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM350	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM351	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM352–SM353	Ви встановлюєте прапорець на ON і скидаєте його на OFF. ON: зворотний відлік OFF.: підрахунок
SM360-SM391	Система встановлює прапорець на ON або OFF.
SM400–SM403	Система автоматично встановлює прапорець на ON і скидає його на OFF. Прапор оновлюється кожного циклу сканування.
SM404	Система автоматично встановлює прапорець на ON і скидає його на OFF.

Спеціальне допоміжне реле	Час оновлення
	SM404 оновлюється кожні 5 мілісекунд.
SM405	Система автоматично встановлює SM405 на ON і скидає його на OFF. SM405 оновлюється кожні 50 мілісекунд.
SM406	Система автоматично встановлює SM406 на ON і скидає його на OFF. SM406 оновлюється кожні 100 мілісекунд.
SM407	Система автоматично встановлює SM407 на ON і скидає його на OFF. SM407 оновлюється кожні 0,5 секунди.
SM450	Система автоматично вмикає SM450 і скидає на OFF. ON: карта пам'яті вставлена в ПЛК. OFF: картку пам'яті вилучено з ПЛК.
SM452–SM453	Система встановлює прапорець на ON або OFF.
SM454	Ви встановлюєте прапорець на ON або OFF.
SM455	Система встановлює прапорець на ON або OFF.
SM456	Ви встановлюєте прапорець на ON, щоб зберегти. Після завершення збереження система скидає його до OFF.
SM457	Система встановлює прапорець на ON або OFF.
SM460	Система встановлює прапорець на ON або OFF.
SM461	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM462–SM464	Ви встановлюєте прапорець на ON або OFF.
SM465	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM466	Ви встановлюєте прапорець на ON або OFF.
SM467	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM469–SM471	Ви встановлюєте прапорець на ON або OFF.
SM472	Система встановлює прапорець на ON або OFF.
SM473	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM474–SM477	Ви встановлюєте прапорець на ON або OFF.
SM478–SM479	Ви встановлюєте прапорець на ON або OFF, а система встановлює прапорець на OFF.
SM480	Система встановлює прапорець на ON або OFF.
SM481	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM482–SM484	Ви встановлюєте прапорець на ON або OFF.
SM485	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM486	Ви встановлюєте прапорець на ON або OFF.
SM487	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM489–SM491	Ви встановлюєте прапорець на ON або OFF.
SM492	Система встановлює прапорець на ON або OFF.
SM493	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM494–SM497	Ви встановлюєте прапорець на ON або OFF.
SM498–SM499	Ви встановлюєте прапорець на ON або OFF, а система встановлює прапорець на OFF.
SM500	Система встановлює прапорець на ON або OFF.
SM501	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM502–SM504	Ви встановлюєте прапорець на ON або OFF.
SM505	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM506	Ви встановлюєте прапорець на ON або OFF.
SM507	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM509–SM511	Ви встановлюєте прапорець на ON або OFF.
SM512	Система встановлює прапорець на ON або OFF.
SM513	Система встановлює прапорець на ON, а ви скидаєте його на OFF.

Спеціальне допоміжне реле	Час оновлення
SM514–SM517	Ви встановлюєте прапорець на ON або OFF.
SM518–SM519	Ви встановлюєте прапорець на ON або OFF, а система встановлює прапорець на OFF.
SM520	Система встановлює прапорець на ON або OFF.
SM521	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM522–SM524	Ви встановлюєте прапорець на ON або OFF.
SM525	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM526	Ви встановлюєте прапорець на ON або OFF.
SM527	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM529–SM531	Ви встановлюєте прапорець на ON або OFF.
SM532	Система встановлює прапорець на ON або OFF.
SM533	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM534–SM537	Ви встановлюєте прапорець на ON або OFF.
SM538–SM539	Ви встановлюєте прапорець на ON або OFF, а система встановлює прапорець на OFF.
SM540	Система встановлює прапорець на ON або OFF.
SM541	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM542–SM544	Ви встановлюєте прапорець на ON або OFF.
SM545	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM546	Ви встановлюєте прапорець на ON або OFF.
SM547	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM549–SM551	Ви встановлюєте прапорець на ON або OFF.
SM552	Система встановлює прапорець на ON або OFF.
SM553	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM554–SM557	Ви встановлюєте прапорець на ON або OFF.
SM558–SM559	Ви встановлюєте прапорець на ON або OFF, а система встановлює прапорець на OFF.
SM560	Система встановлює прапорець на ON або OFF.
SM561	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM562–SM564	Ви встановлюєте прапорець на ON або OFF.
SM565	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM566	Ви встановлюєте прапорець на ON або OFF.
SM567	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM569–SM569	Ви встановлюєте прапорець на ON або OFF.
SM572	Система встановлює прапорець на ON або OFF.
SM573	Система встановлює прапорець на ON, а ви скидаєте його на OFF.
SM574	Ви встановлюєте прапорець на ON або OFF.
SM578–SM579	Ви встановлюєте прапорець на ON або OFF, а система встановлює прапорець на OFF.
SM580	Ви встановлюєте прапорець на ON, і система скидає його на OFF. ON: вимкнути високошвидкісний вихід
SM581	Ви встановлюєте прапорець на ON і скидаєте його на OFF.
SM585	Ви встановлюєте прапорець на ON або OFF, а система встановлює прапорець на OFF.
SM586–SM587	Ви встановлюєте прапорець на ON або OFF.
CM591–CM596	Ви встановлюєте прапорець на ON, а система скидає його на OFF або ви скидаєте його на OFF.
SM600–SM602	Система автоматично встановлює прапорець на ON і скидає його на OFF. Прапор оновлюється, коли інструкція виконується.
SM603	Виконати інструкцію XCMP.
SM604	Ви встановлюєте SM604 на ON і скидаєте його на OFF. ON: сортувати ОК спадання

Спеціальне допоміжне реле	Час оновлення
	OFF.: сортувати ОК зростання
SM605	Ви встановлюєте SM605 на ON і скидаєте його на OFF.
SM606	Ви встановлюєте SM606 на ON і скидаєте його на OFF. ON: 8-бітний режим OFF.: 16-бітний режим
SM607	Ви встановлюєте прапорець на ON або OFF.
SM608	Прапор оновлюється, коли інструкція виконується.
SM609	Ви встановлюєте прапорець на ON або OFF.
SM610–SM611	Прапор оновлюється, коли інструкція виконується.
SM612–SM613	Ви встановлюєте прапорець на ON або OFF.
SM614	Прапор оновлюється, коли інструкція виконується.
SM615–SM617	Ви встановлюєте прапорець на ON або OFF.
SM618	Прапор оновлюється, коли інструкція виконується.
SM619	Прапор оновлюється, коли виконується інструкція EI або DI.
SM620	Прапор оновлюється, коли виконується інструкція CMPT.
SM621–SM686	Ви встановлюєте прапорець на ON або OFF.
SM687	Прапор оновлюється, коли виконується інструкція RAMP.
SM688	Прапор оновлюється, коли виконується інструкція INCD.
SM690–SM691	Ви встановлюєте прапорець на ON або OFF.
SM692	Прапор оновлюється, коли виконується інструкція HKY.
SM693	Прапор оновлюється, коли виконується інструкція SEGL.
SM694	Прапор оновлюється, коли виконується інструкція DSW.
SM695	Ви встановлюєте прапорець на ON або OFF.
SM749	Система оновлюється під час увімкнення після завантаження параметрів обміну даними.
SM750–SM783	Після завантаження параметрів обміну даними ви встановлюєте прапорець на ON або OFF.
SM784–SM847	Прапорець увімкнено, коли система оновлюється.
SM861	Система оновлюється під час увімкнення після завантаження параметрів обміну даними.
SM862–SM895	Після завантаження параметрів обміну даними ви встановлюєте прапорець у положення ON або OFF.
SM896–SM959	Прапор ON, коли система оновлюється автоматично.
SM976–SM990	Прапор ON, коли система оновлюється автоматично.
SM1000	Ви встановлюєте прапорець на ON, а після збереження система встановлює прапорець на OFF.
SM1006	Ви встановлюєте прапорець на ON або OFF.
SM1008–SM1015	Ви встановлюєте прапорець на ON або OFF.
SM1016–SM1031	Прапор ON, коли система оновлюється автоматично.
SM1001	ON: з'єднання Ethernet активне. OFF: з'єднання Ethernet не активне.
SM1006	Після завантаження параметрів обміну даними ви встановлюєте прапорець у положення ON або OFF.
CM1008–CM1015	Після завантаження параметрів обміну даними ви встановлюєте прапорець у положення ON або OFF.
CM1016–CM1031	Прапор ON, коли система оновлюється автоматично.
SM1090	ON: з'єднання TCP зайняте.
SM1091	ON: з'єднання UDP зайняте.
SM1092	Оновлення системи.
SM1100	Прапор оновлюється, коли виконується API 2200-API 2210 або підключається мережевий кабель.
SM1106	ON: ініціалізація PDU не вдається.
SM1107	ON: IP-адреса, адреса маски мережі та адреса шлюзу встановлені неправильно.
SM1109	ON: функцію розетки увімкнено, і використовується той самий порт.

Спеціальне допоміжне реле	Час оновлення
SM1111	Ви встановлюєте прапорець на ON або OFF.
SM1113	ON: виникла помилка сервера.
SM1116	ON: тригер параметра PLC увімкнено.
SM1117	ON: спрацьовує тригер параметра PLC.
SM1119	ON: тригер увімкнено, і останній лист успішно надіслано.
SM1120	ON: тригер увімкнено, а останній лист надіслано з помилкою.
SM1122–SM1123	ON: тригер увімкнено та час очікування відповіді сервера SMTP.
SM1124	ON: тригер увімкнено, а розмір вкладеного файлу перевищує обмеження.
SM1125	ON: тригер увімкнено, і вкладення не знайдено.
SM1126–SM1127	ON: тригер параметра PLC увімкнено.
SM1129	ON: тригер увімкнено, і останній лист успішно надіслано.
SM1130	ON: тригер увімкнено, а останній лист надіслано з помилкою.
SM1132	ON: тригер увімкнено та час очікування відповіді сервера SMTP.
SM1133	ON: тригер увімкнено та виникає помилка відповіді сервера SMTP.
SM1134	ON: тригер увімкнено, а розмір вкладеного файлу перевищує обмеження.
SM1135	ON: тригер увімкнено, і вкладення не знайдено.
SM1136	ON: тригер для параметра PLC увімкнено.
SM1137	ON: запускається тригер для параметра PLC.
SM1139	ON: тригер увімкнено, і останній лист успішно надіслано.
SM1140	ON: тригер увімкнено, а останній лист надіслано з помилкою.
SM1142	ON: тригер увімкнено та час очікування відповіді сервера SMTP.
SM1143	ON: тригер увімкнено та виникає помилка відповіді сервера SMTP.
SM1144	ON: тригер увімкнено, а розмір вкладеного файлу перевищує обмеження.
SM1145	ON: тригер увімкнено, і вкладення не знайдено.
SM1146	ON: тригер для параметра PLC увімкнено.
SM1147	ON: запускається тригер для параметра PLC.
SM1149	ON: тригер увімкнено, і останній лист успішно надіслано.
SM1150	ON: тригер увімкнено, а останній лист надіслано з помилкою.
SM1152	ON: тригер увімкнено та час очікування відповіді сервера SMTP.
SM1153	ON: тригер увімкнено та виникає помилка відповіді сервера SMTP.
SM1154	ON: тригер увімкнено, а розмір вкладеного файлу перевищує обмеження.
SM1155	ON: тригер увімкнено, і вкладення не знайдено.
SM1160-SM1161	Відновлення параметрів на пристроях Delta
SM1166	Після завантаження параметрів обміну даними система оновлюється.
SM1167–SM1199	Після завантаження параметрів обміну даними ви встановлюєте прапорець на ON або OFF.
SM1200–SM1263	ON: коли система оновлюється.
SM1269	ON: виникла помилка конфігурації сокета.
SM1270–SM1353	Прапор оновлюється, коли виконується функція сокета.
SM1312–SM1315	Прапор ON, коли система оновлюється автоматично.
SM1375–SM1407	Після завантаження параметрів обміну даними через EtherNet/IP ви встановлюєте прапорець на ON або OFF.
SM1408–SM1439	ON: виникла помилка в обміні даними через EtherNet/IP.
SM1440–SM1447	ON: виник тайм-аут в адаптері підключення вводу-виводу через EtherNet/IP.
SM1448~SM1455	ON: з'єднання вводу-виводу встановлюється через EtherNet/IP (адаптер).

Спеціальне допоміжне реле	Час оновлення
	OFF: з'єднання вводу-виводу НЕ встановлено через EtherNet/IP (адаптер) або зв'язок розірвано.
SM1581–SM1588	Ви встановлюєте прапорець на ON або OFF.
SM1601–SM1608	Ви встановлюєте прапорець на ON або OFF.
SM1611–SM1618	Система встановлює прапорець на ON або OFF.
SM1621–SM1628	Система встановлює прапорець на ON або OFF.
SM1631–SM1638	Система встановлює прапорець на ON, а ви встановлюєте його на OFF.
SM1641–SM1648	Ви встановлюєте прапорець на ON або OFF.
SM1651–SM1658	Система встановлює прапорець на ON або OFF.
SM1661–SM1668	Ви встановлюєте прапорець на ON або OFF.
SM1671–SM1682	Система встановлює прапорець на ON або OFF.
SM1683	Система встановлює прапорець на ON або OFF.
CM1684–CM1685	Ви встановлюєте прапорець на ON або OFF.
SM1691–SM1698	Система встановлює прапорець на ON або OFF.
SM1700–SM1705	Система встановлює прапорець на ON або OFF.
SM1709–SM1710	Ви встановлюєте прапорець на ON або OFF.
SM1712–SM1743	Ви встановлюєте прапорець на ON або OFF.
SM1744–SM1807	Система встановлює прапорець на ON або OFF.
SM1821–SM1822	Ви встановлюєте прапорець на ON або OFF.
SM1824–SM1855	Ви встановлюєте прапорець на ON або OFF.
SM1856–SM1919	Система встановлює прапорець на ON або OFF.
SM1921–SM1928	Система встановлює прапорець на ON або OFF.
SM1931–SM1938	Ви встановлюєте прапорець на ON або OFF.
SM1941–SM1948	Ви встановлюєте прапорець на ON або OFF.
SM1951–SM1958	Система встановлює прапорець на ON або OFF.

2.2.9 Реле (S)

Ви можете легко використовувати реле в промисловій автоматизації для встановлення процедури. Це найпростіший пристрій у програмуванні послідовної функціональної схеми (SFC).

Існує 2048 реле (S0 – S2047). Кожне крокове реле схоже на вихідне реле, оскільки воно має вихідну котушку, контакт А та контакт В. Ви можете використовувати реле кілька разів у програмі, але реле не може безпосередньо керувати зовнішнім навантаженням. Крім того, ви можете використовувати реле як загальне допоміжне реле, якщо воно не використовується в послідовній функціональній схемі.

2.2.10 Таймери (T)

У цьому розділі описано таймери, доступні в ISPSOft. Додаткову інформацію про таймери див. у посібнику користувача ISPSOft.

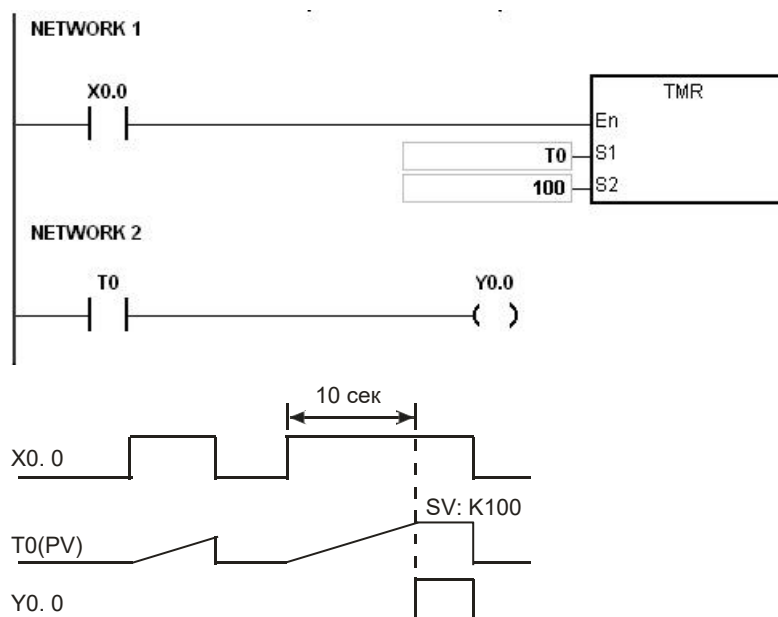
- Таймер на 100 мілісекунд: таймер, визначений інструкцією TMR, приймає 100 мілісекунд як одиницю часу.
- Таймер на 1 мілісекунду: таймер, визначений інструкцією TMRH, приймає 1 мілісекунду як одиницю часу.
- Накопичувальні таймери ST0–ST511. Якщо ви хочете використовувати функцію моніторингу пристрою, ці таймери можуть контролювати T0–T511.
- Якщо ви повторно використовуєте той самий таймер у програмі, зокрема в різних інструкціях TMR і TMRH, значення налаштування таймера є тим, яке таймер відповідає першому.
- Якщо ви повторно використовуєте той самий таймер у програмі, таймер ВИМКНЕТЬСЯ, коли один із умовних контактів OFF.

- Якщо ви використовуєте той самий таймер у програмі, що й таймер для виключного використання підпрограмою, і накопичувальний таймер у програмі, він OFF, коли один із умовних контактів OFF.
- Коли таймер перемикається з UBIMK. на OFF., а умовний контакт увімкнено, таймер скидається та починає відлік.
- Коли виконується інструкція TMR, указана котушка таймера вмикається, і таймер починає відлік. Коли значення таймера збігається зі значенням налаштування таймера (значення таймера $\geq$ значення налаштування), стан контакту ON.

А. Таймери загального призначення

Коли інструкція TMR виконується, таймер загального призначення починає відлік. Коли значення таймера збігається зі значенням налаштування таймера, вихідна котушка вмикається.

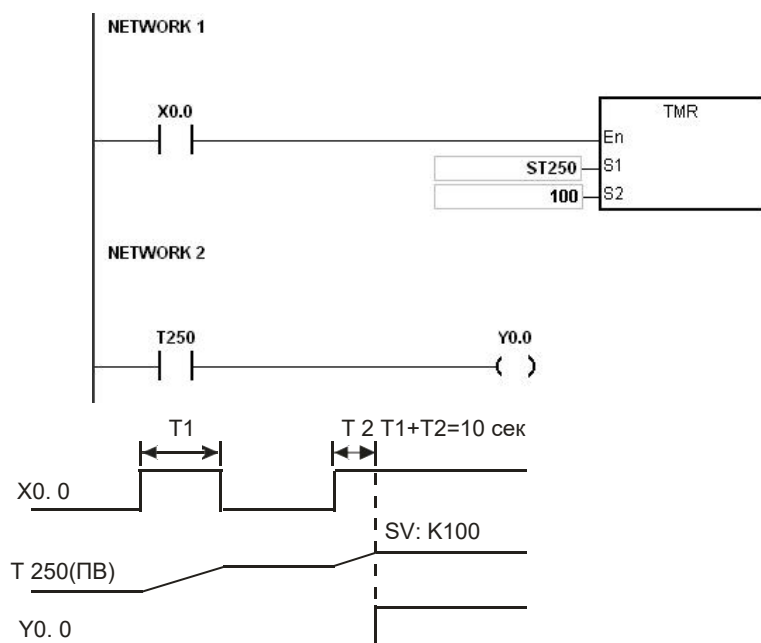
- Коли X0.0=ON і таймер приймає 100 мс як одиницю часу, вихідна котушка T0 є ВКЛ, коли значення таймера = значення налаштування таймера 100.
- Коли X0.0=OFF або живлення вимкнено, значення таймера дорівнює 0, а вихідна котушка T0 вимкнена.



В. Накопичувальні таймери

Коли інструкція TMR виконується, накопичувальний таймер починає відлік. Коли значення таймера збігається зі значенням налаштування таймера, вихідна котушка вмикається. Поки ви додаєте літеру S перед літерою T, таймер стає накопичувальним. Коли умовний контакт OFF, значення накопичувального таймера не скидається. Коли умовний контакт увімкнено, накопичувальний таймер відраховує поточне значення.

- Коли X0.0=ON і таймер T250 приймає 100 мс як одиницю часу, вихідна котушка T250 є ВКЛ, коли значення таймера = значення налаштування таймера 100.
- Коли X0.0=OFF або живлення вимкнено, накопичувальний таймер ST250 припиняє відлік, а значення таймера залишається незмінним. Коли X0.0=ON, значення таймера є накопичувальним значенням. Коли накопичене значення = значення налаштування таймера 100, вихідна котушка T250 увімкнена.



С. Таймери, що використовуються у функціональних блоках

T412–T511 — це таймери, які можна використовувати у функціональному блоці або в перериваннях.

Коли виконується інструкція TMR або END, таймер у функціональному блоці починає відлік. Коли значення таймера збігається зі значенням налаштування таймера, вихідна котушка вмикається.

Якщо ви використовуєте таймер загального призначення у функціональному блоці або перериванні, і функція або переривання не виконується, таймер не може правильно розраховувати.

2.2.11 Лічильники

● Характеристики 16-розрядного лічильника

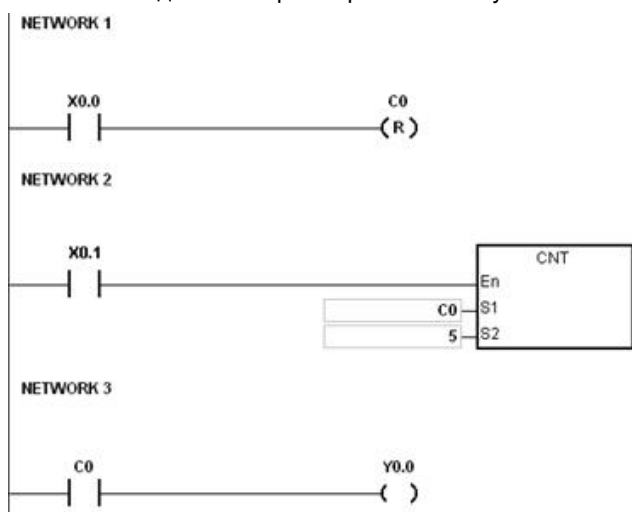
Пункт	16-розрядний лічильник
Тип	Загальний тип
Номер	C0–C511
Напрямок	Підраховуючи
Значення налаштування	0–32,767
Вказівка значення установки лічильника	Значення налаштування може бути константою або значенням у регістрі даних.
Зміна поточного значення	Лічильник припиняє підрахунок, коли значення лічильника збігається зі значенням налаштування лічильника.
Вихідний контакт	Контакт увімкнено, коли значення лічильника відповідає встановленому значенню лічильника.
Скинути	Коли виконується інструкція RST, поточне значення скидається до нуля, а контакт скидається у положення OFF.
Дія контакту	Після завершення сканування контакт починає діяти.

● Функція лічильника

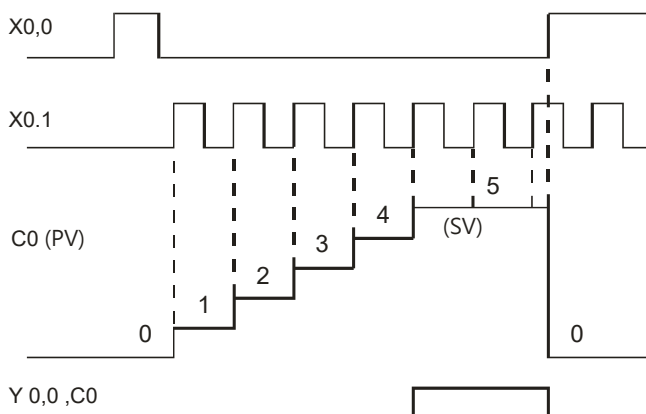
Кожного разу, коли вхід перемикається з OFF. на УВІМК., значення лічильника збігається з значенням вихідної котушки. Як значення налаштування лічильника можна використовувати десяткову константу або значення в регістрі даних.

16-бітний лічильник:

1. Діапазон налаштувань: 0–32 767. Значення параметрів 0 і 1 означають те саме, що вихідний контакт увімкнено, коли лічильник веде підрахунок вперше.
2. Для лічильника загального призначення поточне значення лічильника стирається, коли втрачається живлення. Якщо лічильник фіксується, поточне значення лічильника та стан контакту до втрати живлення зберігаються. Фіксований лічильник відраховується від поточного значення, коли живлення відновлюється.
3. Якщо ви використовуєте інструкцію MOV або ISPSOft для передачі значення, що перевищує значення налаштування лічильника, до регістра поточного значення C0, контакт лічильника C0 увімкнено, і поточне значення стає таким самим, як значення налаштування лічильника наступного разу X0. 1 перемикається з OFF. на UVIMK.
4. Ви можете використовувати константу або значення в регістрі даних як значення налаштування лічильника.
5. Значення налаштування лічильника може бути позитивним або негативним. Якщо лічильник починає відлік від 32 767, наступним значенням буде 0.
6. Використовуйте тут лічильники з інструкцією CNT. Приклад, використаний тут, є поширеним. Зверніться до інструкції API 1003 CNT для більш розширених застосувань.



7. Коли X0.0=ON, інструкція RST виконується, поточне значення C0 скидається на нуль, а вихідний контакт лічильника C0 стає FF.
8. Коли X0.1 змінюється з OFF на ON, значення лічильника збільшується на одиницю.
9. Коли значення лічильника C0 досягає значення налаштування лічильника 5, контакт лічильника C0 вмикається (поточне значення C0 = значення налаштування лічильника = 5). Після цього тригер з X0.1 не приймається C0, а поточне значення C0 залишається на 5.



2.2.12 32-розрядні лічильники (НС)

● Характеристики 32-розрядного лічильника

Пункт	32-розрядний лічильник		
Тип	Лічильник вгору/вниз	Лічильник вгору	Швидкісний лічильник
Номер	НС0–НС63	НС64–НС199	НС200–НС255
Напрямок	Рахує вгору/вниз	Підраховує	Рахує вгору/вниз
Значення налаштування	від -2 147 483 648 до +2 147 483 647		
Специфікація значення установки лічильника	Значення налаштування лічильника може бути або константою, або значенням, що займає два регістри даних (32-біт).		
Зміна поточного значення	Лічильник продовжує підрахунок навіть після того, як значення лічильника збігається зі значенням налаштування лічильника.		
Вихідний контакт	Контакт увімкнено, коли значення додаткового лічильника відповідає значенню налаштування лічильника. Контакт скидається у положення OFF, коли значення лічильника віднімання збігається зі значенням налаштування лічильника.		
Скинути	Коли виконується інструкція RST, поточне значення скидається до нуля, а контакт скидається до OFF.		
Дія контакту	Після завершення сканування інструкцій DCNT контакт починає діяти.		

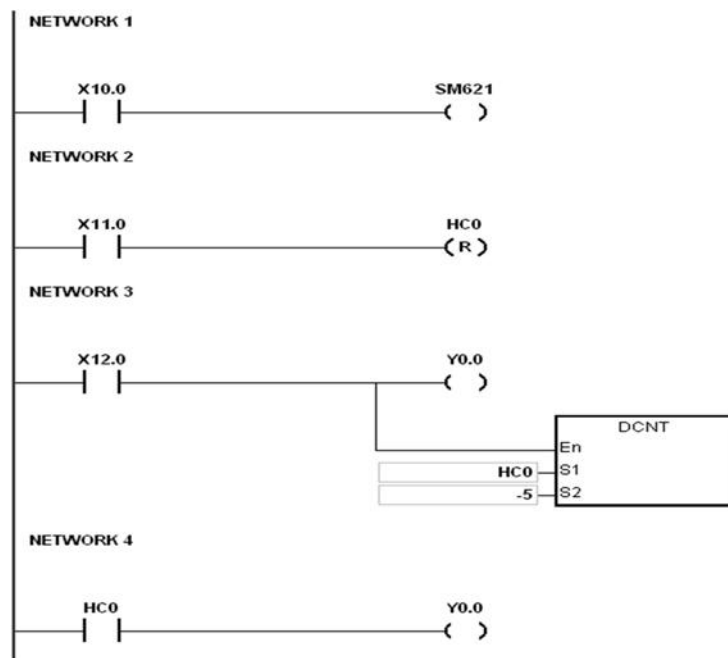
● 32-розрядний лічильник додавання/віднімання загального призначення

- Різниця між 32-розрядними лічильниками додавання загального призначення та 32-розрядними лічильниками віднімання загального призначення залежить від станів спеціальних допоміжних реле SM621–SM684. Наприклад, лічильник НС0 є лічильником додавання, коли SM621 вимкнено, тоді як НС0 є лічильником віднімання, коли SM621 увімкнено.
- Ви можете використовувати константу або значення в регістрах даних як значення налаштування лічильника, і це значення налаштування може бути позитивним або негативним. Якщо ви використовуєте значення в регістрах даних як значення налаштування лічильника, це значення налаштування займає два послідовні регістри.
- Для лічильника загального призначення поточне значення лічильника стирається при втраті живлення. Якщо лічильник фіксується, поточне значення лічильника та стан контакту до втрати живлення зберігається. Зафіксований лічильник відраховується від поточного значення, коли живлення відновлюється.
- Якщо лічильник відраховує від 2 147 483 647, наступне прирістне значення буде -2 147 483 648. Якщо лічильник веде зворотній відлік від -2 147 483 648, наступним приростом буде 2 147 483 647.
- Використовуйте тут високошвидкісні лічильники з інструкцією DCNT. Приклад, використаний тут, є поширеним. Зверніться до інструкції API 1004 DCNT для більш розширених застосувань.

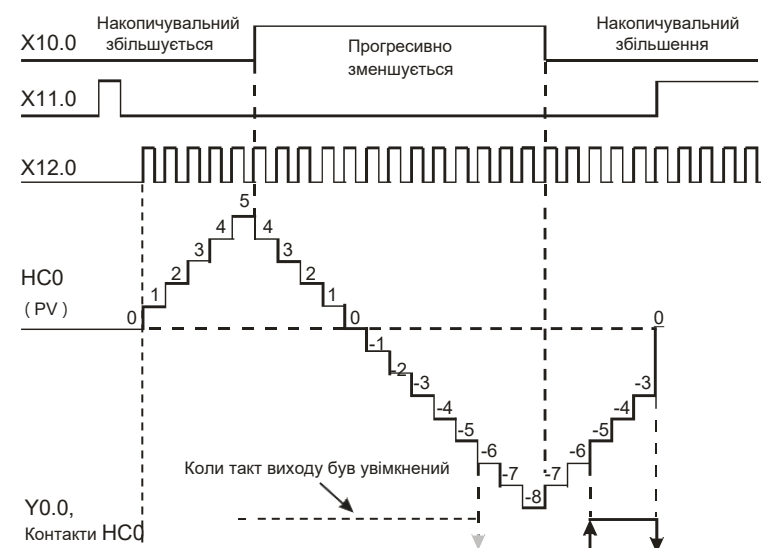
● 32-розрядний високошвидкісний лічильник додавання/віднімання

Зверніться до опису інструкцій API1004 DCNT у Посібнику з програмування серії AS для отримання додаткової інформації.

приклад:



1. X10.0 управляє SM621 для визначення напрямку підрахунку (вгору/вниз) для HC0.
2. Коли X11.0 змінюється з OFF на ON, інструкція RST виконується, і PV в HC0 очищається до 0, а його контакт вимкнено.
3. Коли X12.0 змінюється з OFF на ON, PV для HC0 підраховуватиметься вгору (плюс 1) або зворотний відлік (мінус 1).
4. Коли PV в HC0 змінюється від -6 до -5, контакт HC0 змінюється з OFF. на УВІМК. Коли PV в HC0 змінюється від -5 до -6, контакт HC0 змінюється з ON на OFF.



2.2.13 Регістри даних (D)

Регістр даних зберігає 16-бітні дані. Старший біт представляє або позитивний, або негативний знак, а значення, які можуть зберігати регістри даних, коливаються від -32 768 до +32 767.

Два 16-розрядних регістра можна об'єднати в 32-розрядний регістр; наприклад, (D+1, D), у якому молодший числовий регістр представляє молодші 16 бітів. Старший біт представляє або позитивний, або негативний знак, а значення, які можуть зберігати регістри даних, коливаються від -2 147 483 648 до +2 147 483 647.

- Чотири 16-розрядних регістра можна об'єднати в 64-розрядний регістр; наприклад, (D+3, D+2, D+1, D), в якому молодший числовий регістр представляє молодші 16 бітів. Старший біт представляє або позитивний, або негативний знак, а значення, які можуть зберігати регістри даних, коливаються від -9 223 372 036 854 776 до +9,223,372,036,854,775,807.
- Ви також можете використовувати регістри даних для оновлення значень у регістрах керування в модулях, відмінних від модулів цифрового введення/виведення. Зверніться до посібника користувача ISPSOft для отримання додаткової інформації про оновлення значень у керуючих регістрах.

Існує три типи реєстрів.

- Регістри загального призначення: коли ПЛК переходить у режим RUN або від'єднується, значення в регістрі обнулюється. Якщо ви хочете зберегти дані, коли ПЛК змінює режим RUN, зверніться до посібника користувача ISPSOft для отримання додаткової інформації. Зверніть увагу, що значення все ще обнулюється, коли ПЛК від'єднано.
- Фіксований регістр: якщо ПЛК відключено, дані в регістрі блокування не очищаються. Іншими словами, значення до відключення зберігається. Якщо ви хочете очистити дані в зафіксованій області, ви можете скористатися інструкціями RST або ZRST.
- Часто відображені регістри: якщо програма ПЛК містить модуль, після кожного циклу сканування ПЛК виконує відображення даних із модулем. І зіставлені дані зберігаються між D26000~D29999. Додаткову інформацію про автоматично призначені адреси пристроїв див. у Розділі 8 Посібника з експлуатації AS. Ви також можете перевірити призначені адреси пристрою під час налаштування в HWCONFIG ISPSOft.

2.2.14 Регістри спеціальних даних (SR)

Кожен спеціальний регістр даних має власне визначення та специфічну функцію. Стан системи та повідомлення про помилки зберігаються в спеціальних регістрах даних. Ви також можете використовувати спеціальні регістри даних для моніторингу статусів системи. Регістри спеціальних даних та їхні функції перераховані в таблиці нижче.

- Для номерів SR, позначених «*», зверніться до додаткових приміток у розділі 2.2.1.6 про спеціальні допоміжні реле/спеціальні регістри даних.
- «R» у стовпці атрибутів вказує на те, що спеціальний регістр даних може читати дані; «R/W» у стовпці атрибутів вказує на те, що він може читати та записувати дані.
- Знак «—» означає, що статус реєстру спеціальних даних не змінюється.
- «#» означає, що система налаштована відповідно до стану ПЛК, і ви можете прочитати налаштування. Для отримання додаткової інформації зверніться до відповідного посібника.
- Буква «Y» у колонці «заблокований» означає, що він зафіксований, «N» означає, що він не зафіксований; і «H» означає, що він відповідає налаштуванням у HWCONFIG.

Під час виконання ви можете редагувати програми в ПЛК, але налаштування в HWCONFIG не змінюються.

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
SR0	Помилка роботи ПЛК/операнда	○	○	0	0	-	N	R	0
SR1	Адреса помилки операції (32-біт)	○	○	0	0	-	N	R	0
SR2		○	○	0	0	-	N	R	0
SR4	Помилка перевірки граматики	○	○	0	0	-	N	R	0
SR5	Адреса помилки перевірки граматики (32-розрядна)	○	○	0	0	-	N	R	0
SR6		○	○	0	0	-	N	R	0
*SR8	Адреса кроку, на якій увімкнено сторожовий таймер (32-розрядний)	○	○	0	-	-	N	R	0
SR9		○	○	0	-	-	N	R	0
SM27	AS-F2AD: позначки помилок для вбудованої AD	○	○	0	-	-	N	R/W	0
SR28	Номер останнього виведення, коли інструкція високошвидкісного виведення використовується багаторазово	○	○	-1	-1	-1	N	R	-1
SR29	Точка виведення, де сталася остання помилка під час виконання інструкції CSFO.	○	○	-1	-1	-1	N	R	-1
SR30	Запишіть номер кроку, на якому частота виводилася повторно з останньої високошвидкісної точки виведення	○	○	0	0	-	N	R	0
SR31		○	○	0	0	-	N	R	0
SR32	Адреса останньої інструкції, яка перевищила дозволений діапазон	○	○	-1	-1	-	N	R	-1
SR33		○	○	-1	-1	-	N	R	-1
*SR36	Система зберігає дані на карту пам'яті. Ця функція працює з SM36	○	○	0	-	-	N	R/W	0
SR38	Позначка помилки для AS-FEN02, FPFN02, FOPC02 і FFTP01.	○	-	0	-	-	N	R/W	0
*SR40	Кількість журналів помилок	○	○	-	-	-	Y	R	0
*SR41	Показчик журналу помилок	○	○	-	-	-	Y	R	0
*SR42	Журнал помилок 1: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR43	Журнал помилок 1: ID модуля	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR44	Журнал помилок 1: код помилки	○	○	-	-	-	Y	R	0
*SR45	Журнал помилок 1: рік і місяць	○	○	-	-	-	Y	R	0
*SR46	Журнал помилок 1: день і година	○	○	-	-	-	Y	R	0
*SR47	Журнал помилок 1: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR48	Журнал помилок 2: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR49	Журнал помилок 2: ID модуля	○	○	-	-	-	Y	R	0
*SR50	Журнал помилок 2: код помилки	○	○	-	-	-	Y	R	0
*SR51	Журнал помилок 2: рік і місяць	○	○	-	-	-	Y	R	0
*SR52	Журнал помилок 2: день і година	○	○	-	-	-	Y	R	0
*SR53	Журнал помилок 2: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR54	Журнал помилок 3: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR55	Журнал помилок 3: ID модуля	○	○	-	-	-	Y	R	0
*SR56	Журнал помилок 3: код помилки	○	○	-	-	-	Y	R	0
*SR57	Журнал помилок 3: рік і місяць	○	○	-	-	-	Y	R	0
*SR58	Журнал помилок 3: день і година	○	○	-	-	-	Y	R	0
*SR59	Журнал помилок 3: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR60	Журнал помилок 4: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR61	Журнал помилок 4: ID модуля	○	○	-	-	-	Y	R	0
*SR62	Журнал помилок 4: код помилки	○	○	-	-	-	Y	R	0
*SR63	Журнал помилок 4: рік і місяць	○	○	-	-	-	Y	R	0
*SR64	Журнал помилок 4: день і година	○	○	-	-	-	Y	R	0
*SR65	Журнал помилок 4: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR66	Журнал помилок 5: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR67	Журнал помилок 5: ID модуля	○	○	-	-	-	Y	R	0
*SR68	Журнал помилок 5: код помилки	○	○	-	-	-	Y	R	0
*SR69	Журнал помилок 5: рік і місяць	○	○	-	-	-	Y	R	0
*SR70	Журнал помилок 5: день і година	○	○	-	-	-	Y	R	0
*SR71	Журнал помилок 5: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR72	Журнал помилок 6: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR73	Журнал помилок 6: ID модуля	○	○	-	-	-	Y	R	0
*SR74	Журнал помилок 6: код помилки	○	○	-	-	-	Y	R	0
*SR75	Журнал помилок 6: рік і місяць	○	○	-	-	-	Y	R	0
*SR76	Журнал помилок 6: день і година	○	○	-	-	-	Y	R	0
*SR77	Журнал помилок 6: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR78	Журнал помилок 7: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR79	Журнал помилок 7: ID модуля	○	○	-	-	-	Y	R	0
*SR80	Журнал помилок 7: код помилки	○	○	-	-	-	Y	R	0
*SR81	Журнал помилок 7: рік і місяць	○	○	-	-	-	Y	R	0
*SR82	Журнал помилок 7: день і година	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR83	Журнал помилок 7: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR84	Журнал помилок 8: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR85	Журнал помилок 8: ID модуля	○	○	-	-	-	Y	R	0
*SR86	Журнал помилок 8: код помилки	○	○	-	-	-	Y	R	0
*SR87	Журнал помилок 8: рік і місяць	○	○	-	-	-	Y	R	0
*SR88	Журнал помилок 8: день і година	○	○	-	-	-	Y	R	0
*SR89	Журнал помилок 8: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR90	Журнал помилок 9: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR91	Журнал помилок 9: ID модуля	○	○	-	-	-	Y	R	0
*SR92	Журнал помилок 9: код помилки	○	○	-	-	-	Y	R	0
*SR93	Журнал помилок 9: рік і місяць	○	○	-	-	-	Y	R	0
*SR94	Журнал помилок 9: день і година	○	○	-	-	-	Y	R	0
*SR95	Журнал помилок 9: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR96	Журнал помилок 10: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR97	Журнал помилок 10: ID модуля	○	○	-	-	-	Y	R	0
*SR98	Журнал помилок 10: код помилки	○	○	-	-	-	Y	R	0
*SR99	Журнал помилок 10: рік і місяць	○	○	-	-	-	Y	R	0
*100 SR	Журнал помилок 10: день і година	○	○	-	-	-	Y	R	0
*SR101	Журнал помилок 10: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR102	Журнал помилок 11: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR103	Журнал помилок 11: ID модуля	○	○	-	-	-	Y	R	0
*SR104	Журнал помилок 11: код помилки	○	○	-	-	-	Y	R	0
*SR105	Журнал помилок 11: рік і місяць	○	○	-	-	-	Y	R	0
*SR106	Журнал помилок 11: день і година	○	○	-	-	-	Y	R	0
*SR107	Журнал помилок 11: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR108	Журнал помилок 12: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR109	Журнал помилок 12: ID модуля	○	○	-	-	-	Y	R	0
*SR110	Журнал помилок 12: код помилки	○	○	-	-	-	Y	R	0
*SR111	Журнал помилок 12: рік і місяць	○	○	-	-	-	Y	R	0
*SR112	Журнал помилок 12: день і година	○	○	-	-	-	Y	R	0
*SR113	Журнал помилок 12: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR114	Журнал помилок 13: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR115	Журнал помилок 13: ID модуля	○	○	-	-	-	Y	R	0
*SR116	Журнал помилок 13: код помилки	○	○	-	-	-	Y	R	0
*SR117	Журнал помилок 13: рік і місяць	○	○	-	-	-	Y	R	0
*SR118	Журнал помилок 13: день і година	○	○	-	-	-	Y	R	0
*SR119	Журнал помилок 13: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR120	Журнал помилок 14: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR121	Журнал помилок 14: ID модуля	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR122	Журнал помилок 14: код помилки	○	○	-	-	-	Y	R	0
*SR123	Журнал помилок 14: рік і місяць	○	○	-	-	-	Y	R	0
*SR124	Журнал помилок 14: день і година	○	○	-	-	-	Y	R	0
*SR125	Журнал помилок 14: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR126	Журнал помилок 15: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR127	Журнал помилок 15: ID модуля	○	○	-	-	-	Y	R	0
*SR128	Журнал помилок 15: код помилки	○	○	-	-	-	Y	R	0
*SR129	Журнал помилок 15: рік і місяць	○	○	-	-	-	Y	R	0
*SR130	Журнал помилок 15: день і година	○	○	-	-	-	Y	R	0
*SR131	Журнал помилок 15: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR132	Журнал помилок 16: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR133	Журнал помилок 16: ID модуля	○	○	-	-	-	Y	R	0
*SR134	Журнал помилок 16: код помилки	○	○	-	-	-	Y	R	0
*SR135	Журнал помилок 16: рік і місяць	○	○	-	-	-	Y	R	0
*SR136	Журнал помилок 16: день і година	○	○	-	-	-	Y	R	0
*SR137	Журнал помилок 16: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR138	Журнал помилок 17: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR139	Журнал помилок 17: ID модуля	○	○	-	-	-	Y	R	0
*SR140	Журнал помилок 17: код помилки	○	○	-	-	-	Y	R	0
*SR141	Журнал помилок 17: рік і місяць	○	○	-	-	-	Y	R	0
SR142	Журнал помилок 17: день і година	○	○	-	-	-	Y	R	0
*SR143	Журнал помилок 17: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR144	Журнал помилок 18: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR145	Журнал помилок 18: ID модуля	○	○	-	-	-	Y	R	0
*SR146	Журнал помилок 18: код помилки	○	○	-	-	-	Y	R	0
*SR147	Журнал помилок 18: рік і місяць	○	○	-	-	-	Y	R	0
*SR148	Журнал помилок 18: день і година	○	○	-	-	-	Y	R	0
*SR149	Журнал помилок 18: хвилина і секунда	○	○	-	-	-	Y	R	0
*150 SR	Журнал помилок 19: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR151	Журнал помилок 19: ID модуля	○	○	-	-	-	Y	R	0
*SR152	Журнал помилок 19: код помилки	○	○	-	-	-	Y	R	0
*SR153	Журнал помилок 19: рік і місяць	○	○	-	-	-	Y	R	0
*SR154	Журнал помилок 19: день і година	○	○	-	-	-	Y	R	0
*SR155	Журнал помилок 19: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR156	Журнал помилок 20: ЦП або віддалений номер	○	○	-	-	-	Y	R	0
*SR157	Журнал помилок 20: ID модуля	○	○	-	-	-	Y	R	0
*SR158	Журнал помилок 20: код помилки	○	○	-	-	-	Y	R	0
*SR159	Журнал помилок 20: рік і місяць	○	○	-	-	-	Y	R	0
*SR160	Журнал помилок 20: день і година	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR161	Журнал помилок 20: хвилина і секунда	○	○	-	-	-	Y	R	0
SR162	Тривалість часу, протягом якого ПЛК увімкнено (одиниця: хвилини) (32-розрядний)	○	○	-	-	-	Y	R	-
SR163									
SR166	Значення VR0 (працює з SM166)	○	-	0	-	-	N	R	0
SR167	Значення VR1 (працює з SM167)	○	-	0	-	-	N	R	0
*SR168	Серія AS300: вхідне значення в каналі 1 функціональної карти 1 F2AD Серія AS218: вхідне значення в канал AD 1	○	○	0	-	-	N	R	0
*SR169	Серія AS300: вхідне значення в каналі 2 функціональної карти 1 F2AD Серія AS218: вхідне значення в канал AD 2	○	○	0	-	-	N	R	0
*SR170	Вхідне значення в канал 1 функціональної карти 2 F2AD	○	-	0	-	-	N	R	0
*SR171	Вхідне значення в канал 2 функціональної карти 2 F2AD	○	-	0	-	-	N	R	0
*SR172	Серія AS300: Вихідне значення в каналі 1 функціональної карти 1 F2DA Серія AS218: вхідне значення в DA канал 1	○	○	0	-	0	N	R/W	0
*SR173	Серія AS300: Вихідне значення в каналі 2 функціональної карти 1 F2DA Серія AS218: Вихідне значення в DA каналі 2	○	○	0	-	0	N	R/W	0
*SR174	Вихідне значення в каналі 1 функціональної карти 2 F2DA	○	-	0	-	0	N	R/W	0
*SR175	Вихідне значення в каналі 2 функціональної карти 2 F2DA	○	-	0	-	0	N	R/W	0
SR176	Ідентифікатор зв'язку (COM11) карти послідовного зв'язку 1	○	-	-	-	-	N	R	1
SR177	Код протоколу зв'язку (COM11) карти послідовного зв'язку 1	○	-	-	-	-	N	R	0x24
SR178	Ідентифікатор зв'язку (COM12) карти послідовного зв'язку 1	○	-	-	-	-	N	R	1
SR179	Код протоколу зв'язку (COM12) карти послідовного зв'язку 1	○	-	-	-	-	N	R	0x24
SR180	Останній код попередження про помилку	○	○	0	-	-	N	R	0
SR182	Тривалість тайм-ауту комунікаційної карти 1 (COM11) (одиниця: мілісекунди). 0: немає тайм-ауту	○	-	-	-	-	N	R/W	200
SR183	Тривалість тайм-ауту комунікаційної карти 2 (COM12) (одиниця: мілісекунди). 0: немає тайм-ауту	○	-	-	-	-	N	R/W	200
SR185	Час циклу зв'язку для всіх віддалених модулів (одиниця: мілісекунди)	○	○	0	-	-	N	R	0
SR190	Множення частоти високошвидкісного лічильника групи 1 (за замовч.: 1-кратна частота)	○	○	1	-	-	N	R/W	1
SR191	Множення частоти високошвидкісного лічильника групи 2 (за замовч.: 1-кратна частота)	○	○	1	-	-	N	R/W	1

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SR192	Множення частоти високошвидкісного лічильника групи 3 (за замовч.: однократна частота)	○	○	1	-	-	N	R/W	1
SR193	Множення частоти високошвидкісного лічильника групи 4 (за замовч.: 1-кратна частота)	○	○	1	-	-	N	R/W	1
SR194	Множення частоти високошвидкісного лічильника групи 5 (за замовч.: 1-кратна частота)	○	○	1	-	-	N	R/W	1
SR195	Множення частоти високошвидкісного лічильника групи 6 (за замовч.: 1-кратна частота)	○	○	1	-	-	N	R/W	1
SR196	Множення частоти високошвидкісного лічильника групи 7 (за замовч.: однократна частота)	○	○	1	-	-	N	R/W	1
SR197	Множення частоти високошвидкісного лічильника групи 8 (за замовч.: 1-кратна частота)	○	○	1	-	-	N	R/W	1
SR198	Пі (π), число з плаваючою комою (32-розрядне)	○	○	16# 0FDB	16# 0FDB	16# 0FDB	N	R	16# 0FDB
SR199				16# 4049	16# 4049	16# 4049	N		16# 4049
*SR201	Адреса зв'язку COM1	○	○	-	-	-	H	R/W	1
*SR202	Адреса зв'язку COM2	○	○	-	-	-	H	R/W	1
*SR209	Протокол зв'язку COM1	○	○	-	-	-	H	R/W	16# 0024
*SR210	Тайм-аут зв'язку COM1 (одиниця виміру: мілісекунда) 0: час очікування відсутній	○	○	-	-	-	H	R/W	0
*SR212	Протокол зв'язку COM2	○	○	-	-	-	H	R/W	16# 0024
*SR213	Час очікування зв'язку COM2 (одиниця виміру: мілісекунди) 0: час очікування відсутній	○	○	-	-	-	H	R/W	0
*SR215	Назва функціональної картки 1	○	-	-	-	-	N	R	0
*SR216	Назва функціональної картки 2	○	-	-	-	-	N	R	0
SR217	Значення швидкості передачі даних COM1 (одиниця: 100 біт/с)	○	○	96	-	-	H	R/W	96
SR218	Значення швидкості передачі даних COM2 (одиниця: 100 біт/с)	○	○	96	-	-	H	R/W	96
*SR220	Значення року годинника реального часу (RTC): 00–99 (BCD)	○	○	-	-	-	Y	R	0
*SR221	Місячне значення годинника реального часу (RTC): 01–12 (BCD)	○	○	-	-	-	Y	R	1
*SR222	Годинник реального часу (RTC) значення дня: 1–31 (BCD)	○	○	-	-	-	Y	R	1
*SR223	Годинник реального часу (RTC): 00–23 (BCD)	○	○	-	-	-	Y	R	0
*SR224	Хвилини годинника реального часу (RTC): 00–59 (BCD)	○	○	-	-	-	Y	R	0
*SR225	Друге значення годинника реального часу (RTC): 00–59 (BCD)	○	○	-	-	-	Y	R	0
*SR226	Тижневе значення годинника реального часу (RTC): 1–7 (BCD)	○	○	-	-	-	Y	R	1
*SR227	Кількість журналів завантажень (максимум 20)	○	○	-	-	-	Y	R	0
*SR228	Завантажити показчик журналу	○	○	-	-	-	Y	R	0
*SR229	Завантажити журнал 1: номер дії	○	○	-	-	-	Y	R	0
*SR230	Завантажити журнал 1: рік і місяць	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR231	Завантажити журнал 1: день і година	○	○	-	-	-	Y	R	0
*SR232	Завантажити журнал 1: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR233	Завантажити журнал 2: номер дії	○	○	-	-	-	Y	R	0
*SR234	Завантажити журнал 2: рік і місяць	○	○	-	-	-	Y	R	0
*SR235	Завантажити журнал 2: день і година	○	○	-	-	-	Y	R	0
*SR236	Завантажити журнал 2: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR237	Завантажити журнал 3: номер дії	○	○	-	-	-	Y	R	0
*SR238	Завантажити журнал 3: рік і місяць	○	○	-	-	-	Y	R	0
*SR239	Завантажити журнал 3: день і година	○	○	-	-	-	Y	R	0
*SR240	Завантажити журнал 3: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR241	Завантажити журнал 4: номер дії	○	○	-	-	-	Y	R	0
*SR242	Завантажити журнал 4: рік і місяць	○	○	-	-	-	Y	R	0
*SR243	Завантажити журнал 4: день і година	○	○	-	-	-	Y	R	0
*SR244	Завантажити журнал 4: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR245	Завантажити журнал 5: номер дії	○	○	-	-	-	Y	R	0
*SR246	Завантажити журнал 5: рік і місяць	○	○	-	-	-	Y	R	0
*SR247	Завантажити журнал 5: день і година	○	○	-	-	-	Y	R	0
*SR248	Завантажити журнал 5: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR249	Завантажити журнал 6: номер дії	○	○	-	-	-	Y	R	0
*250 SR	Завантажити журнал 6: рік і місяць	○	○	-	-	-	Y	R	0
*SR251	Завантажити журнал 6: день і година	○	○	-	-	-	Y	R	0
*SR252	Завантажити журнал 6: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR253	Завантажити журнал 7: номер дії	○	○	-	-	-	Y	R	0
*SR254	Завантажити журнал 7: рік і місяць	○	○	-	-	-	Y	R	0
*SR255	Завантажити журнал 7: день і година	○	○	-	-	-	Y	R	0
*SR256	Завантажити журнал 7: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR257	Завантажити журнал 8: номер дії	○	○	-	-	-	Y	R	0
*SR258	Завантажити журнал 8: рік і місяць	○	○	-	-	-	Y	R	0
*SR259	Завантажити журнал 8: день і година	○	○	-	-	-	Y	R	0
*SR260	Завантажити журнал 8: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR261	Завантажити журнал 9: номер дії	○	○	-	-	-	Y	R	0
*SR262	Завантажити журнал 9: рік і місяць	○	○	-	-	-	Y	R	0
*SR263	Завантажити журнал 9: день і година	○	○	-	-	-	Y	R	0
*SR264	Завантажити журнал 9: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR265	Завантажити журнал 10: номер дії	○	○	-	-	-	Y	R	0
*SR266	Завантажити журнал 10: рік і місяць	○	○	-	-	-	Y	R	0
*SR267	Завантажити журнал 10: день і година	○	○	-	-	-	Y	R	0
*SR268	Завантажити журнал 10: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR269	Завантажити журнал 11: номер дії	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR270	Завантажити журнал 11: рік і місяць	○	○	-	-	-	Y	R	0
*SR271	Завантажити журнал 11: день і година	○	○	-	-	-	Y	R	0
*SR272	Завантажити журнал 11: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR273	Завантажити журнал 12: номер дії	○	○	-	-	-	Y	R	0
*SR274	Завантажити журнал 12: рік і місяць	○	○	-	-	-	Y	R	0
*SR275	Завантажити журнал 12: день і година	○	○	-	-	-	Y	R	0
*SR276	Завантажити журнал 12: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR277	Завантажити журнал 13: номер дії	○	○	-	-	-	Y	R	0
*SR278	Завантажити журнал 13: рік і місяць	○	○	-	-	-	Y	R	0
*SR279	Завантажити журнал 13: день і година	○	○	-	-	-	Y	R	0
*SR280	Завантажити журнал 13: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR281	Завантажити журнал 14: номер дії	○	○	-	-	-	Y	R	0
*SR282	Завантажити журнал 14: рік і місяць	○	○	-	-	-	Y	R	0
*SR283	Завантажити журнал 14: день і година	○	○	-	-	-	Y	R	0
*SR284	Завантажити журнал 14: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR285	Завантажити журнал 15: номер дії	○	○	-	-	-	Y	R	0
*SR286	Завантажити журнал 15: рік і місяць	○	○	-	-	-	Y	R	0
*SR287	Завантажити журнал 15: день і година	○	○	-	-	-	Y	R	0
*SR288	Завантажити журнал 15: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR289	Завантажити журнал 16: номер дії	○	○	-	-	-	Y	R	0
*SR290	Завантажити журнал 16: рік і місяць	○	○	-	-	-	Y	R	0
*SR291	Завантажити журнал 16: день і година	○	○	-	-	-	Y	R	0
*SR292	Завантажити журнал 16: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR293	Завантажити журнал 17: номер дії	○	○	-	-	-	Y	R	0
*SR294	Завантажити журнал 17: рік і місяць	○	○	-	-	-	Y	R	0
*SR295	Завантажити журнал 17: день і година	○	○	-	-	-	Y	R	0
*SR296	Завантажити журнал 17: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR297	Завантажити журнал 18: номер дії	○	○	-	-	-	Y	R	0
*SR298	Завантажити журнал 18: рік і місяць	○	○	-	-	-	Y	R	0
*SR299	Завантажити журнал 18: день і година	○	○	-	-	-	Y	R	0
*300 SR	Завантажити журнал 18: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR301	Завантажити журнал 19: номер дії	○	○	-	-	-	Y	R	0
*SR302	Завантажити журнал 19: рік і місяць	○	○	-	-	-	Y	R	0
*SR303	Завантажити журнал 19: день і година	○	○	-	-	-	Y	R	0
*SR304	Завантажити журнал 19: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR305	Завантажити журнал 20: номер дії	○	○	-	-	-	Y	R	0
*SR306	Завантажити журнал 20: рік і місяць	○	○	-	-	-	Y	R	0
*SR307	Завантажити журнал 20: день і година	○	○	-	-	-	Y	R	0
*SR308	Завантажити журнал 20: хвилина і секунда	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR309	Кількість журналів змін стану ПЛК (максимум 20)	○	○	-	-	-	Y	R	0
*SR310	Показчик журналу змін статусу ПЛК	○	○	-	-	-	Y	R	0
*SR311	Журнал змін статусу ПЛК 1: номер дії	○	○	-	-	-	Y	R	0
*SR312	Журнал змін стану ПЛК 1: рік і місяць	○	○	-	-	-	Y	R	0
*SR313	Журнал змін стану ПЛК 1: день і година	○	○	-	-	-	Y	R	0
*SR314	Журнал змін стану ПЛК 1: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR315	Журнал змін статусу ПЛК 2: номер дії	○	○	-	-	-	Y	R	0
*SR316	Журнал змін стану ПЛК 2: рік і місяць	○	○	-	-	-	Y	R	0
*SR317	Журнал змін стану ПЛК 2: день і година	○	○	-	-	-	Y	R	0
*SR318	Журнал змін стану ПЛК 2: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR319	Журнал змін статусу ПЛК 3: номер дії	○	○	-	-	-	Y	R	0
*SR320	Журнал змін стану ПЛК 3: рік і місяць	○	○	-	-	-	Y	R	0
*SR321	Журнал змін стану ПЛК 3: день і година	○	○	-	-	-	Y	R	0
*SR322	Журнал змін статусу ПЛК 3: хвилина та секунда	○	○	-	-	-	Y	R	0
*SR323	Журнал змін статусу ПЛК 4: номер дії	○	○	-	-	-	Y	R	0
*SR324	Журнал змін стану ПЛК 4: рік і місяць	○	○	-	-	-	Y	R	0
*SR325	Журнал змін стану ПЛК 4: день і година	○	○	-	-	-	Y	R	0
*SR326	Журнал змін статусу ПЛК 4: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR327	Журнал змін статусу ПЛК 5: номер дії	○	○	-	-	-	Y	R	0
*SR328	Журнал змін стану ПЛК 5: рік і місяць	○	○	-	-	-	Y	R	0
*SR329	Журнал змін статусу ПЛК 5: день і година	○	○	-	-	-	Y	R	0
*SR330	Журнал змін статусу ПЛК 5: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR331	Журнал змін статусу ПЛК 6: номер дії	○	○	-	-	-	Y	R	0
*SR332	Журнал змін стану ПЛК 6: рік і місяць	○	○	-	-	-	Y	R	0
*SR333	Журнал змін статусу ПЛК 6: день і година	○	○	-	-	-	Y	R	0
*SR334	Журнал змін статусу ПЛК 6: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR335	Журнал змін статусу ПЛК 7: номер дії	○	○	-	-	-	Y	R	0
*SR336	Журнал змін стану ПЛК 7: рік і місяць	○	○	-	-	-	Y	R	0
*SR337	Журнал змін статусу ПЛК 7: день і година	○	○	-	-	-	Y	R	0
*SR338	Журнал змін статусу ПЛК 7: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR339	Журнал змін статусу ПЛК 8: номер дії	○	○	-	-	-	Y	R	0
*SR340	Журнал змін стану ПЛК 8: рік і місяць	○	○	-	-	-	Y	R	0
*SR341	Журнал змін стану ПЛК 8: день і година	○	○	-	-	-	Y	R	0
*SR342	Журнал змін статусу ПЛК 8: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR343	Журнал змін статусу ПЛК 9: номер дії	○	○	-	-	-	Y	R	0
*SR344	Журнал змін стану ПЛК 9: рік і місяць	○	○	-	-	-	Y	R	0
*SR345	Журнал змін статусу ПЛК 9: день і година	○	○	-	-	-	Y	R	0
*SR346	Журнал змін статусу ПЛК 9: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR347	Журнал змін статусу ПЛК 10: номер дії	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
*SR348	Журнал змін стану ПЛК 10: рік і місяць	○	○	-	-	-	Y	R	0
*SR349	Журнал змін стану ПЛК 10: день і година	○	○	-	-	-	Y	R	0
*SR350	Журнал змін стану ПЛК 10: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR351	Журнал змін статусу ПЛК 11: номер дії	○	○	-	-	-	Y	R	0
*SR352	Журнал змін стану ПЛК 11: рік і місяць	○	○	-	-	-	Y	R	0
*SR353	Журнал змін стану ПЛК 11: день і година	○	○	-	-	-	Y	R	0
*SR354	Журнал змін статусу ПЛК 11: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR355	Журнал змін статусу ПЛК 12: номер дії	○	○	-	-	-	Y	R	0
*SR356	Журнал змін стану ПЛК 12: рік і місяць	○	○	-	-	-	Y	R	0
*SR357	Журнал змін стану ПЛК 12: день і година	○	○	-	-	-	Y	R	0
*SR358	Журнал змін стану ПЛК 12: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR359	Журнал змін статусу ПЛК 13: номер дії	○	○	-	-	-	Y	R	0
*SR360	Журнал змін стану ПЛК 13: рік і місяць	○	○	-	-	-	Y	R	0
*SR361	Журнал змін стану ПЛК 13: день і година	○	○	-	-	-	Y	R	0
*SR362	Журнал змін статусу ПЛК 13: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR363	Журнал змін статусу ПЛК 14: номер дії	○	○	-	-	-	Y	R	0
*SR364	Журнал змін стану ПЛК 14: рік і місяць	○	○	-	-	-	Y	R	0
*SR365	Журнал змін стану ПЛК 14: день і година	○	○	-	-	-	Y	R	0
*SR366	Журнал змін статусу ПЛК 14: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR367	Журнал змін статусу ПЛК 15: номер дії	○	○	-	-	-	Y	R	0
*SR368	Журнал змін стану ПЛК 15: рік і місяць	○	○	-	-	-	Y	R	0
*SR369	Журнал змін стану ПЛК 15: день і година	○	○	-	-	-	Y	R	0
*SR370	Журнал змін стану ПЛК 15: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR371	Журнал змін статусу ПЛК 16: номер дії	○	○	-	-	-	Y	R	0
*SR372	Журнал змін стану ПЛК 16: рік і місяць	○	○	-	-	-	Y	R	0
*SR373	Журнал змін стану ПЛК 16: день і година	○	○	-	-	-	Y	R	0
*SR374	Журнал змін статусу ПЛК 16: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR375	Журнал змін статусу ПЛК 17: номер дії	○	○	-	-	-	Y	R	0
*SR376	Журнал змін стану ПЛК 17: рік і місяць	○	○	-	-	-	Y	R	0
*SR377	Журнал змін стану ПЛК 17: день і година	○	○	-	-	-	Y	R	0
*SR378	Журнал змін статусу ПЛК 17: хвилина та секунда	○	○	-	-	-	Y	R	0
*SR379	Журнал змін статусу ПЛК 18: номер дії	○	○	-	-	-	Y	R	0
*SR380	Журнал змін стану ПЛК 18: рік і місяць	○	○	-	-	-	Y	R	0
*SR381	Журнал змін стану ПЛК 18: день і година	○	○	-	-	-	Y	R	0
*SR382	Журнал змін статусу ПЛК 18: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR383	Журнал змін статусу ПЛК 19: номер дії	○	○	-	-	-	Y	R	0
*SR384	Журнал змін стану ПЛК 19: рік і місяць	○	○	-	-	-	Y	R	0
*SR385	Журнал змін стану ПЛК 19: день і година	○	○	-	-	-	Y	R	0
*SR386	Журнал змін статусу ПЛК 19: хвилина і секунда	○	○	-	-	-	Y	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
*SR387	Журнал змін статусу ПЛК 20: номер дії	○	○	-	-	-	Y	R	0
*SR388	Журнал змін стану ПЛК 20: рік і місяць	○	○	-	-	-	Y	R	0
*SR389	Журнал змін стану ПЛК 20: день і година	○	○	-	-	-	Y	R	0
*SR390	Журнал змін статусу ПЛК 20: хвилина і секунда	○	○	-	-	-	Y	R	0
*SR391	Значення року годинника реального часу (RTC): 00–99 (BCD)	○	○	-	-	-	Y	R	0
*SR392	Місячне значення годинника реального часу (RTC): 01–12 (BCD)	○	○	-	-	-	Y	R	1
*SR393	Годинник реального часу (RTC) значення дня: 1–31 (BCD)	○	○	-	-	-	Y	R	1
*SR394	Годинник реального часу (RTC): 00–23 (BCD)	○	○	-	-	-	Y	R	0
*SR395	Хвилини годинника реального часу (RTC): 00–59 (BCD)	○	○	-	-	-	Y	R	0
*SR396	Друге значення годинника реального часу (RTC): 00–59 (BCD)	○	○	-	-	-	Y	R	0
*SR397	Тижневе значення годинника реального часу (RTC): 1–7 (BCD)	○	○	-	-	-	Y	R	1
SR403	Функціональна карта 1 версії мікропрограми (наприклад, 10670 = V1.06.70; 0 = немає мікропрограми)	○	○	-	-	-	N	R	0
SR404	Версія мікропрограми карти Funciton 2 (наприклад, 10670 = V1.06.70; 0 = прошивка відсутня)	○	○	-	-	-	N	R	0
SR405	Версія прошивки головного процесора ПЛК (наприклад, 10670 = V1.06.70; 0 = прошивка відсутня)	○	○	-	-	-	Y	R	-
SR406	Версія мікропрограми субпроцесора ПЛК (наприклад, 0 = немає субпроцесора)	○	○	-	-	-	Y	R	-
SR407	Коли ПЛК працює, значення в SR407 збільшується на одиницю щосекунди. SR407 рахує від 0 до 32767, а потім від -32768 до 0.	○	○	0	0	-	N	R/W	0
SR408	Коли ПЛК працює, значення в SR408 збільшується на одиницю кожного циклу сканування. SR408 рахує від 0 до 32767, а потім від -32768 до 0.	○	○	0	0	-	N	R/W	0
SR411	Поточний час сканування зберігається в SR411 і SR412 (одиниця вимірювання 100 мікросекунд).	○	○	0	-	-	N	R	0
SR412	Мілісекунди зберігаються в SR411 (діапазон 0–65535). Мікросекунди зберігаються в SR412 (діапазон 0–900). Наприклад, якщо SR411=12 і SR412=300, то поточний час сканування становить 12,3 мілісекунди.	○	○	0	-	-	N	R	0
SR413	Максимальний час сканування зберігається в SR413 і SR414 (одиниця вимірювання 100 мікросекунд). Значення мілісекунди зберігається в SR413.	○	○	0	-	-	N	R	0
SR414									
SR415	Мінімальний час сканування зберігається в SR415 і SR416 (одиниця вимірювання 100 мікросекунд). Значення мілісекунди зберігається в SR415.	○	○	0	-	-	N	R	0
SR416									
SR421	Тривалість переривання таймера I601 (одиниця: мілісекунди). За замовч. 0, тобто система використовує налаштування в HWCONFIG.	○	○	0	0	-	N	R/W	0
SR422	Тривалість переривання таймера I602 (одиниця: мілісекунди). За замовч. 0, тобто система використовує налаштування в HWCONFIG.	○	○	0	0	-	N	R/W	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
SR423	Тривалість переривання таймера I603 (одиниця: мілісекунди). За замовч. 0, тобто система використовує налаштування в HWCONFIG.	○	○	0	0	-	N	R/W	0
SR424	Тривалість переривання таймера I604 (одиниця виміру: 0,1 мілісекунди). За замовч. 0, тобто система використовує налаштування в HWCONFIG.	○	○	0	0	-	N	R/W	0
SR440	MAC-адреса	○	○	-	-	-	Y	R	-
SR441	(Приклад: 12:34:56:78:9A:BC => SR440=16#1234,	○	○	-	-	-	Y	R	-
SR442	SR441=16#5678, SR442=16#9ABC) (Оновлюється один раз, коли ПЛК подається живлення)	○	○	-	-	-	Y	R	-
SR443	Серійний номер ПЛК	○	○	-	-	-	Y	R	-
SR444	(Оновлюється один раз, коли ПЛК подається живлення)	○	○	-	-	-	Y	R	-
SR445	ПРИКЛАД : AS324MTAW15500012	○	○	-	-	-	Y	R	-
SR446	AS → SR443 = 16#5341	○	○	-	-	-	Y	R	-
SR447	32 → SR444 = 16#3233	○	○	-	-	-	Y	R	-
SR448	4M → SR445 = 16#4D34	○	○	-	-	-	Y	R	-
SR449	TA → SR446 = 16#4154	○	○	-	-	-	Y	R	-
SR450	W1 → SR447 = 16#3157	○	○	-	-	-	Y	R	-
SR451	55 → SR448 = 16#3535	○	○	-	-	-	Y	R	-
	00 → SR449 = 16#3030								
	01 → SR450 = 16#3130								
	2 → SR451 = 16#0032 (Оновлюється один раз, коли ПЛК подається живлення)	○	○	-	-	-	Y	R	-
*SR453	Якщо під час роботи карти пам'яті виникає помилка, код помилки записується.	○	○	-	-	-	Y	R	0
SR460	Позиція Y0.0/осі 1 (Y0.0/Y0.1) (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR461	Режим виведення осі 1 (Y0.0/Y0.1).	○	○	-	-	-	Y	R/W	0
SR462	Початкова/кінцева частота осі 1 (Y0.0/Y0.1).	○	○	-	-	-	Y	R/W	200
SR463	Час прискорення осі 1 (Y0.0/Y0.1).	○	○	-	-	-	Y	R/W	200
SR464	Час уповільнення осі 1 (Y0.0/Y0.1).	○	○	-	-	-	Y	R/W	200
SR465	Частота JOG по осі 1 (Y0.0/Y0.1).	○	○	-	-	-	Y	R/W	200
SR466	Номер осі 1 (Y0.0/Y0.1) у поточній таблиці планування позиції	○	○	0	0	-	N	R	0
SR467	Значення чисельника осі 1 (Y0.0/Y0.1), передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR468	Знаменник осі 1 (Y0.0/Y0.1), переданий із машинного блоку	○	○	-	-	-	H	R/W	0
SR469	Положення машинного блоку на осі 1 (Y0.0/Y0.1) (значення з плаваючою комою одинарної точності)	○	○	-	-	-	Y	R	0
SR470	Цільова частота осі 1 (Y0.0/Y0.1) для фіксованого нахилу	○	○	-	-	-	Y	R	0
SR471									
SR472									

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SR473									
SR474	Положення Y0.1 (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR475									
SR476	Y0.1 початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR477	Y0.1 час прискорення/уповільнення	○	○	-	-	-	Y	R/W	200
SR478	Імпульс компенсації зазору по осі 1 (Y0.0).	○	○	-	-	-	Y	R/W	0
SR479	Імпульс компенсації люфту Y0.1	○	○	-	-	-	Y	R/W	0
SR480	Позиція Y0.2/осі 2 (Y0.2/Y0.3) (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR481									
SR482	Режим виведення Y0.2/вісь 2 (Y0.2/Y0.3).	○	○	-	-	-	Y	R/W	0
SR483	Y0.2/вісь 2 (Y0.2/Y0.3) початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR484	Y0.2/вісь 2 (Y0.2/Y0.3) час прискорення	○	○	-	-	-	Y	R/W	200
SR485	Y0.2/вісь 2 (Y0.2/Y0.3) час уповільнення	○	○	-	-	-	Y	R/W	200
SR486	Y0.2/вісь 2 (Y0.2/Y0.3) Частота JOG	○	○	-	-	-	Y	R/W	200
SR487	Номер Y0.2/осі 2 (Y0.2/Y0.3) у таблиці планування поточної позиції	○	○	0	0	-	N	R	0
SR488	Y0.2/вісь 2 (Y0.2/Y0.3) значення чисельника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR489	Y0.2/вісь 2 (Y0.2/Y0.3) значення знаменника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR490	Y0.2/осі 2 (Y0.2/Y0.3) положення блоку верстата (значення з плаваючою комою одинарної точності)	○	○	-	-	-	Y	R	0
SR491									
SR492	Y0.2/вісь 2 (Y0.2/Y0.3) цільова частота для фіксованого нахилу	○	○	-	-	-	Y	R	0
SR493									
SR494	Позиція Y0.3 (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR495									
SR496	Y0.3 початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR497	Y0.3 час прискорення/уповільнення	○	○	-	-	-	Y	R/W	200
SR498	Імпульс компенсації зазору по осі 2 (Y0.2).	○	○	-	-	-	Y	R/W	0
SR499	Імпульс компенсації люфту Y0.3	○	○	-	-	-	Y	R/W	0
SR500	Позиція Y0.4/осі 3 (Y0.4/Y0.5) (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR501									
SR502	Режим виведення Y0.4/вісь 3 (Y0.4/Y0.5).	○	○	-	-	-	Y	R/W	0
SR503	Y0.4/вісь 3 (Y0.4/Y0.5) початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR504	Y0.4/вісь 3 (Y0.4/Y0.5) час прискорення	○	○	-	-	-	Y	R/W	200
SR505	Y0.4/вісь 3 (Y0.4/Y0.5) час уповільнення	○	○	-	-	-	Y	R/W	200
SR506	Y0.4/вісь 3 (Y0.4/Y0.5) частота JOG	○	○	-	-	-	Y	R/W	200
SR507	Номер Y0.4/осі 3 (Y0.4/Y0.5) у таблиці планування поточної позиції	○	○	0	0	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SR508	Y0.4/вісь 3 (Y0.4/Y0.5) значення чисельника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR509	Y0.4/вісь 3 (Y0.4/Y0.5) значення знаменника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR510	Y0.4/осі 3 (Y0.4/Y0.5) положення блоку верстата (значення з плаваючою комою одинарної точності)	○	○	-	-	-	Y	R	0
SR511									
SR512	Y0.4/вісь 3 (Y0.4/Y0.5) цільова частота для фіксованого нахилу	○	○	-	-	-	Y	R	0
SR513									
SR514	Положення Y0.5 (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR515									
SR516	Y0.5 початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR517	Y0.5 час прискорення/уповільнення	○	○	-	-	-	Y	R/W	200
SR518	Імпульс компенсації зазору по осі 3 (Y0.4).	○	○	-	-	-	Y	R/W	0
SR519	Імпульс компенсації люфту Y0.5	○	○	-	-	-	Y	R/W	0
SR520	Позиція Y0.6/осі 4 (Y0.6/Y0.7) (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR521									
SR522	Режим виведення Y0.6/вісь 4 (Y0.6/Y0.7).	○	○	-	-	-	Y	R/W	0
SR523	Y0.6/вісь 4 (Y0.6/Y0.7) початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR524	Y0.6/вісь 4 (Y0.6/Y0.7) час прискорення	○	○	-	-	-	Y	R/W	200
SR525	Y0.6/вісь 4 (Y0.6/Y0.7) час уповільнення	○	○	-	-	-	Y	R/W	200
SR526	Y0.6/вісь 4 (Y0.6/Y0.7) Частота JOG	○	○	-	-	-	Y	R/W	200
SR527	Номер Y0.6/осі 4 (Y0.6/Y0.7) у таблиці планування поточної позиції	○	○	0	0	-	N	R	0
SR528	Y0.6/вісь 4 (Y0.6/Y0.7) значення чисельника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR529	Y0.6/вісь 4 (Y0.6/Y0.7) значення знаменника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR530	Y0.6/осі 4 (Y0.6/Y0.7) положення машинного блоку (значення з плаваючою комою одинарної точності)	○	○	-	-	-	Y	R	0
SR531									
SR532	Y0.6/вісь 4 (Y0.6/Y0.7) цільова частота для фіксованого нахилу	○	○	-	-	-	Y	R	0
SR533									
SR534	Позиція Y0.7 (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR535									
SR536	Y0.7 початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR537	Y0.7 час прискорення/уповільнення	○	○	-	-	-	Y	R/W	200
SR538	Імпульс компенсації зазору по осі 4 (Y0.6).	○	○	-	-	-	Y	R/W	0
SR539	Імпульс компенсації люфту Y0.7	○	○	-	-	-	Y	R/W	0
SR540	Y0.8/вісь 5 (Y0.8/Y0.9) положення де (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR541									
SR542	Режим виведення осі 5 (Y0.8/Y0.9).	○	○	-	-	-	Y	R/W	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SR543	Y0.8/вісь 5 (Y0.8/Y0.9) початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR544	Y0.8/вісь 5 (Y0.8/Y0.9) час прискорення	○	○	-	-	-	Y	R/W	200
SR545	Y0.8/вісь 5 (Y0.8/Y0.9) час уповільнення	○	○	-	-	-	Y	R/W	200
SR546	Y0.8/вісь 5 (Y0.8/Y0.9) частота JOG	○	○	-	-	-	Y	R/W	200
SR547	Номер Y0.8/осі 5 (Y0.8/Y0.9) у таблиці планування поточної позиції	○	○	0	0	-	N	R	0
SR548	Y0.8/вісь 5 (Y0.8/Y0.9) значення чисельника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR549	Y0.8/вісь 5 (Y0.8/Y0.9) значення знаменника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR550	Y0.8/осі 5 (Y0.8/Y0.9) положення блоку машини	○	○	-	-	-	Y	R	0
SR551	(значення з плаваючою комою одинарної точності)								
SR552	Y0.8/вісь 5 (Y0.8/Y0.9) цільова частота для фіксованого нахилу	○	○	-	-	-	Y	R	0
SR553									
SR554	Позиція Y0.9 (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR555									
SR556	Y0.9 початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR557	Y0.9 час прискорення/уповільнення	○	○	-	-	-	Y	R/W	200
SR558	Імпульс компенсації зазору по осі 5 (Y0.8).	○	○	-	-	-	Y	R/W	0
SR559	Імпульс компенсації люфту Y0.9	○	○	-	-	-	Y	R/W	0
SR560	Позиція Y0.10/осі 6 (Y0.10/Y0.11) (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR561									
SR562	Режим виведення Y0.10/вісь 6 (Y0.10/Y0.11).	○	○	-	-	-	Y	R/W	0
SR563	Y0.10/вісь 6 (Y0.10/Y0.11) початкова/кінцева частота	○	○	-	-	-	Y	R/W	200
SR564	Y0.10/вісь 6 (Y0.10/Y0.11) час прискорення	○	○	-	-	-	Y	R/W	200
SR565	Y0.10/вісь 6 (Y0.10/Y0.11) час уповільнення	○	○	-	-	-	Y	R/W	200
SR566	Y0.10/вісь 6 (Y0.10/Y0.11) частота JOG	○	○	-	-	-	Y	R/W	200
SR567	Номер Y0.10/осі 6 (Y0.10/Y0.11) у таблиці планування поточної позиції	○	○	0	0	-	N	R	0
SR568	Y0.10/вісь 6 (Y0.10/Y0.11) значення чисельника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR569	Y0.10/вісь 6 (Y0.10/Y0.11) значення знаменника, передане з машинного блоку	○	○	-	-	-	H	R/W	0
SR570	Y0.10/вісь 6 (Y0.10/Y0.11) положення блоку	○	○	-	-	-	Y	R	0
SR571	верстата (значення з плаваючою комою одинарної точності)								
SR572	Y0.10/вісь 6 (Y0.10/Y0.11) цільова частота для фіксованого нахилу	○	○	-	-	-	Y	R	0
SR573									
SR574	Позиція Y0.11 (одиниця: кількість імпульсів)	○	○	-	-	-	Y	R/W	0
SR575									
SR576	Y0.11 початкова/кінцева частота	○	○	-	-	-	Y	R/W	200

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SR577	Y0.11 час прискорення/уповільнення	○	○	-	-	-	Y	R/W	200
SR578	Імпульс компенсації зазору по осі 6 (Y0.10).	○	○	-	-	-	Y	R/W	0
SR579	Імпульс компенсації люфту Y0.11	○	○	-	-	-	Y	R/W	0
*SR580	Позитивний ліміт осі 1 (Y0.0/Y0.1) у ISPSoft (одиниця:	○	○	-	-	-	H	R/W	0
*SR581	кількість імпульсів)								
*SR582	Негативний ліміт осі 1 (Y0.0/Y0.1) в ISPSoft (одиниця:	○	○	-	-	-	H	R/W	0
*SR583	кількість імпульсів)								
*SR584	Позитивне обмеження осі 2 (Y0.2/Y0.3) в ISPSoft	○	○	-	-	-	H	R/W	0
*SR585	(одиниця: кількість імпульсів)								
*SR586	Негативне обмеження осі 2 (Y0.2/Y0.3) в ISPSoft	○	○	-	-	-	H	R/W	0
*SR587	(одиниця: кількість імпульсів)								
*SR588	Позитивне обмеження осі 3 (Y0.4/Y0.5) в ISPSoft	○	○	-	-	-	H	R/W	0
*SR589	(одиниця: кількість імпульсів)								
*SR590	Негативне обмеження осі 3 (Y0.4/Y0.5) в ISPSoft	○	○	-	-	-	H	R/W	0
*SR591	(одиниця: кількість імпульсів)								
*SR592	Позитивний ліміт осі 4 (Y0.6/Y0.7) в ISPSoft (одиниця:	○	○	-	-	-	H	R/W	0
*SR593	кількість імпульсів)								
*SR594	Негативне обмеження осі 4 (Y0.6/Y0.7) в ISPSoft	○	○	-	-	-	H	R/W	0
*SR595	(одиниця: кількість імпульсів)								
*SR596	Позитивний ліміт осі 5 (Y0.8/Y0.9) в ISPSoft (одиниця:	○	○	-	-	-	H	R/W	0
*SR597	кількість імпульсів)								
*SR598	Негативний ліміт осі 5 (Y0.8/Y0.9) в ISPSoft (одиниця:	○	○	-	-	-	H	R/W	0
*SR599	кількість імпульсів)								
*600 SR	Позитивний ліміт осі 6 (Y0.10/Y0.11) в ISPSoft	○	○	-	-	-	H	R/W	0
*SR601	(одиниця: кількість імпульсів)								
*SR602	Негативний ліміт осі 6 (Y0.10/Y0.11) в ISPSoft	○	○	-	-	-	H	R/W	0
*SR603	(одиниця: кількість імпульсів)								
SR610	Поточна вихідна швидкість осі 1 (Y0.0/Y0.1) (одиниці: Гц)	○	○	0	0	0	N	R	0
SR611									
SR612	вісь 2 (Y0.2/Y0.3) поточна вихідна швидкість (одиниці: Гц)	○	○	0	0	0	N	R	0
SR613									
SR614	Поточна вихідна швидкість осі 3 (Y0.4/Y0.5) (одиниці: Гц)	○	○	0	0	0	N	R	0
SR615									
SR616	Поточна вихідна швидкість осі 4 (Y0.6/Y0.7) (одиниці: Гц)	○	○	0	0	0	N	R	0
SR617									
SR618	Поточна вихідна швидкість осі 5 (Y0.8/Y0.9) (одиниці: Гц)	○	○	0	0	0	N	R	0
SR619									
SR620	Поточна вихідна швидкість осі 6 (Y0.10/Y0.11) (одиниці: Гц)	○	○	0	0	0	N	R	0
SR621									

SR	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SR623	Умова переривання зовнішнього входу: точки входу X0.0–X0.15 спрацьовують за спадаючим фронтом	○	○	FFFF	FFFF	-	N	R	FFFF
SR624	Умова зовнішнього переривання: точки входу X0.0–X0.15 запускаються по наростаючому фронту	○	○	FFFF	FFFF	-	N	R	FFFF
SR625	Стан високошвидкісного переривання порівняння I200–I233	○	○	FFFF	FFFF	-	N	R	FFFF
SR626	Стан високошвидкісного переривання порівняння I240–I253	○	○	FFFF	FFFF	-	N	R	FFFF
SR627	Умова високошвидкісного переривання порівняння I260–I267	○	○	FFFF	FFFF	-	N	R	FFFF
SR628	Стан переривань зв'язку I300–I307	○	○	FFFF	FFFF	-	N	R	FFFF
SR629	Стан вихідних переривань I500–I505	○	○	FFFF	FFFF	-	N	R	FFFF
SR630	Стан вихідних переривань I510–I519	○	○	FFFF	FFFF	-	N	R	FFFF
SR632	Стан переривання таймера I601–I604	○	○	FFFF	FFFF	-	N	R	FFFF
SR633	Стан модуля розширення I400–I415	○	○	FFFF	FFFF	-	N	R	FFFF
SR634	Стан модуля розширення перериває I416–I431	○	○	FFFF	FFFF	-	N	R	FFFF
SR640	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.0	○	○	0	-	-	N	R/W	0
SR641	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.1	○	○	0	-	-	N	R/W	0
SR642	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.2	○	○	0	-	-	N	R/W	0
SR643	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.3	○	○	0	-	-	N	R/W	0
SR644	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.4	○	○	0	-	-	N	R/W	0
SR645	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.5	○	○	0	-	-	N	R/W	0
SR646	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.6	○	○	0	-	-	N	R/W	0
SR647	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.7	○	○	0	-	-	N	R/W	0
SR648	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.8	○	○	0	-	-	N	R/W	0
SR649	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.9	○	○	0	-	-	N	R/W	0
SR650	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.10	○	○	0	-	-	N	R/W	0
SR651	Встановіть час виведення на 0-20 мс раніше (одиниця: 1 мс), щоб працювати з вихідним імпульсом Y0.11	○	○	0	-	-	N	R/W	0
*SR652	Номер імпульсу в прискоренні з останнього виведення позиції (32-розрядний)	○	○	0	0	-	N	R	0
*SR653		○	○	0	0	-	N	R	0
*SR654	Номер імпульсу при уповільненні від останнього положення	○	○	0	0	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR655	вихід (32-розрядний)	○	○	0	0	-	N	R	0
SR656	Код скасування SDO для зв'язку ECAT між серво/інвертором Delta (32-розрядний)	○	-	0	0	-	N	R/W	0
SR658	Ідентифікатор зв'язку CAN серво/інвертора Delta, де сталася помилка	○	○	0	-	-	N	R/W	0
	Ідентифікатор зв'язку ECAT серво/інвертора Delta, де сталася помилка	○	-	0	-	-	N	R/W	0
SR659	Код помилки зв'язку CAN серво/інвертора Delta	○	○	0	-	-	N	R/W	0
	Код помилки зв'язку ECAT серво/інвертора Delta	○	-	0	-	-	N	R/W	0
SR661	Команда PR на зв'язку CAN ID 1 сервоприводу Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 1 сервоприводу Delta								
	Поточний стан зв'язку ECAT ID 1 інвертора Delta	○	-	0	-	-	N	R	0
SR662	Команда PR на ідентифікатор зв'язку CAN 2 сервоприводу Delta	○	○	0	-	-	N	R	0
	Команда PR на ECAT зв'язку ID 2 сервоприводу Delta								
	Поточний стан ECAT зв'язку ID 2 інвертора Delta	○	-	0	-	-	N	R	0
SR663	Команда PR на зв'язку CAN ID 3 сервоприводу Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 3 сервоприводу Delta								
	Поточний стан зв'язку ECAT ID 3 інвертора Delta	○	-	0	-	-	N	R	0
SR664	Команда PR на зв'язку CAN ID 4 сервоприводу Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 4 сервоприводу Delta								
	Поточний стан зв'язку ECAT ID 4 інвертора Delta	○	-	0	-	-	N	R	0
SR665	Команда PR на зв'язку CAN ID 5 сервоприводу Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 5 сервоприводу Delta								
	Поточний стан зв'язку ECAT ID 1 інвертора Delta	○	-	0	-	-	N	R	0
SR666	Команда PR на зв'язку CAN ID 6 сервоприводу Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 6 сервоприводу Delta								
	Поточний стан зв'язку ECAT ID 6 інвертора Delta	○	-	0	-	-	N	R	0
SR667	Команда PR на зв'язку CAN ID 7 сервоприводу Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 7 сервоприводу Delta								
	Поточний стан зв'язку ECAT ID 7 інвертора Delta	○	-	0	-	-	N	R	0
SR668	Команда PR на зв'язку CAN ID 8 сервоприводу Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 8 сервоприводу Delta								
	Поточний стан зв'язку ECAT ID 8 інвертора Delta	○	-	0	-	-	N	R	0
SR671	Код аварійного сигналу для ідентифікатора зв'язку CAN 1 сервоприводу Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для ідентифікатора зв'язку ECAT 1 сервоприводу Delta	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SR672	Код аварійного сигналу для ідентифікатора зв'язку CAN 2 сервоприводу Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 2 сервоприводу Delta	○	-	0	-	-	N	R	0
SR673	Код аварійного сигналу для ідентифікатора зв'язку CAN 3 сервоприводу Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 3 сервоприводу Delta	○	-	0	-	-	N	R	0
SR674	Код аварійного сигналу для зв'язку CAN ID 4 сервоприводу Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 4 сервоприводу Delta	○	-	0	-	-	N	R	0
SR675	Код аварійного сигналу для зв'язку CAN ID 5 сервоприводу Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 5 сервоприводу Delta	○	-	0	-	-	N	R	0
SR676	Код аварійного сигналу для зв'язку CAN ID 6 сервоприводу Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 6 сервоприводу Delta	○	-	0	-	-	N	R	0
SR677	Код аварійного сигналу для зв'язку CAN ID 7 сервоприводу Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 7 сервоприводу Delta	○	-	0	-	-	N	R	0
SR678	Код аварійного сигналу для зв'язку CAN ID 8 сервоприводу Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 8 сервоприводу Delta	○	-	0	-	-	N	R	0
SR681	Поточний стан DO ідентифікатора зв'язку CAN 1 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DO ідентифікатора зв'язку ECAT 1 сервоприводу Delta	○	-	0	-	-	N	R	0
SR682	Поточний стан DO ідентифікатора зв'язку CAN 2 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DO ідентифікатора зв'язку ECAT 2 сервоприводу Delta	○	-	0	-	-	N	R	0
SR683	Поточний стан DO ідентифікатора зв'язку CAN 3 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DO ідентифікатора зв'язку ECAT 3 сервоприводу Delta	○	-	0	-	-	N	R	0
SR684	Поточний стан DO зв'язку CAN ID 4 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DO ідентифікатора зв'язку ECAT 4 сервоприводу Delta	○	-	0	-	-	N	R	0
SR685	Поточний стан DO зв'язку CAN ID 5 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DO ідентифікатора зв'язку ECAT 5 сервоприводу Delta	○	-	0	-	-	N	R	0
SR686	Поточний стан DO зв'язку CAN ID 6 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DO ідентифікатора зв'язку ECAT 6 сервоприводу Delta	○	-	0	-	-	N	R	0
SR687	Поточний стан DO зв'язку CAN ID 7 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DO ідентифікатора зв'язку ECAT 7 сервоприводу Delta	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SR688	Поточний стан DO зв'язку CAN ID 8 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DO ідентифікатора зв'язку ECAT 8 сервоприводу Delta	○	-	0	-	-	N	R	0
SR691	Поточна позиція ідентифікатора зв'язку CAN 1 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR692	Поточна позиція ідентифікатора зв'язку ECAT 1 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR693	Поточна позиція ідентифікатора зв'язку CAN 2 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR694	Поточна позиція зв'язку ECAT ID 2 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR695	Поточна позиція зв'язку CAN ID 3 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR696	Поточна позиція зв'язку ECAT ID 3 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR697	Поточна позиція зв'язку CAN ID 4 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR698	Поточна позиція зв'язку ECAT ID 4 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR699	Поточна позиція зв'язку CAN ID 5 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR700	Поточна позиція зв'язку ECAT ID 5 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR701	Поточна позиція зв'язку CAN ID 6 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR702	Поточна позиція зв'язку ECAT ID 6 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR703	Поточна позиція зв'язку CAN ID 7 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR704	Поточна позиція зв'язку ECAT ID 7 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR705	Поточна позиція зв'язку CAN ID 8 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR706	Поточна позиція зв'язку ECAT ID 8 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR711	Цільове положення ідентифікатора зв'язку CAN 1 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR712	Цільове положення ідентифікатора зв'язку ECAT 1 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR713	Цільове положення ідентифікатора зв'язку CAN 2 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR714	Цільове положення ідентифікатора зв'язку ECAT 2 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
SR715	Цільове положення ідентифікатора зв'язку CAN 3 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR716	Цільове положення ідентифікатора зв'язку ECAT 3 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR717	Цільове положення ідентифікатора зв'язку CAN 4 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR718	Цільове положення ідентифікатора зв'язку ECAT 4 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR719	Цільове положення ідентифікатора зв'язку CAN 5 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR720	Цільове положення ідентифікатора зв'язку ECAT 5 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR721	Цільове положення ідентифікатора зв'язку CAN 6 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR722	Цільове положення ідентифікатора зв'язку ECAT 6 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR723	Цільове положення ідентифікатора зв'язку CAN 7 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR724	Цільове положення ідентифікатора зв'язку ECAT 7 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR725	Цільове положення ідентифікатора зв'язку CAN 8 сервоприводу Delta (32-розрядний)	○	○	0	-	-	N	R	0
SR726	Цільове положення ідентифікатора зв'язку ECAT 8 сервоприводу Delta (32-розрядний)	○	-	0	-	-	N	R	0
SR731	Поточний стан DI зв'язку CAN ID 1 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 1 сервоприводу Delta	○	-	0	-	-	N	R	0
SR732	Поточний стан DI зв'язку CAN ID 2 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 2 сервоприводу Delta	○	-	0	-	-	N	R	0
SR733	Поточний стан DI зв'язку CAN ID 3 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 3 сервоприводу Delta	○	-	0	-	-	N	R	0
SR734	Поточний стан DI зв'язку CAN ID 4 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 4 сервоприводу Delta	○	-	0	-	-	N	R	0
SR735	Поточний стан DI зв'язку CAN ID 5 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 5 сервоприводу Delta	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
SR736	Поточний стан DI зв'язку CAN ID 6 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 6 сервоприводу Delta	○	-	0	-	-	N	R	0
SR737	Поточний стан DI зв'язку CAN ID 7 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 7 сервоприводу Delta	○	-	0	-	-	N	R	0
SR738	Поточний стан DI зв'язку CAN ID 8 сервоприводу Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 8 сервоприводу Delta	○	-	0	-	-	N	R	0
SR741	Поточний крутний момент зв'язку CAN ID 1 сервоприводу Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 1 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR742	Поточний крутний момент зв'язку CAN ID 2 сервоприводу Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 2 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR743	Поточний крутний момент зв'язку CAN ID 3 сервоприводу Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 2 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR744	Поточний крутний момент зв'язку CAN ID 4 сервоприводу Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 4 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR745	Поточний крутний момент зв'язку CAN ID 5 сервоприводу Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 5 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR746	Поточний крутний момент зв'язку CAN ID 6 сервоприводу Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 6 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR747	Поточний крутний момент зв'язку CAN ID 7 сервоприводу Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 7 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR748	Поточний крутний момент зв'язку CAN ID 8 сервоприводу Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 8 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
SR751	Поточний стан зв'язку CAN ID 21 інвертора Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 9 сервоприводу Delta Поточний стан зв'язку ECAT ID 9 інвертора Delta	○	-	0	-	-	N	R	0
SR752	Поточний стан зв'язку CAN ID 22 інвертора Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 10 сервоприводу Delta Поточний стан зв'язку ECAT ID 10 інвертора Delta	○	-	0	-	-	N	R	0
SR753	Поточний стан зв'язку CAN ID 23 інвертора Delta	○	○	0	-	-	N	R	0
	Команда PR на комунікаційному ідентифікаторі ECAT 11 сервоприводу Delta Поточний стан зв'язку ECAT ID 11 інвертора Delta	○	-	0	-	-	N	R	0
SR754	Поточний стан зв'язку CAN ID 24 інвертора Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 12 сервоприводу Delta Поточний стан зв'язку ECAT ID 12 інвертора Delta	○	-	0	-	-	N	R	0
SR755	Поточний стан зв'язку CAN ID 25 інвертора Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 13 сервоприводу Delta Поточний стан зв'язку ECAT ID 13 інвертора Delta	○	-	0	-	-	N	R	0
SR756	Поточний стан зв'язку CAN ID 26 інвертора Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 14 сервоприводу Delta Поточний стан зв'язку ECAT ID 14 інвертора Delta	○	-	0	-	-	N	R	0
SR757	Поточний стан зв'язку CAN ID 27 інвертора Delta	○	○	0	-	-	N	R	0
	Команда PR на зв'язку ECAT ID 15 сервоприводу Delta Поточний стан зв'язку ECAT ID 15 інвертора Delta	○	-	0	-	-	N	R	0
SR758	Поточний стан зв'язку CAN ID 28 інвертора Delta	○	○	0	-	-	N	R	0
	Команда PR на комунікаційному ідентифікаторі ECAT 16 сервоприводу Delta	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
	Поточний стан зв'язку ECAT ID 16 інвертора Delta								
SR761	Поточна швидкість двигуна зв'язку CAN ID 21 інвертора Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для ідентифікатора зв'язку ECAT 9 сервоприводу Delta	○	-	0	-	-	N	R	0
SR762	Поточна швидкість двигуна зв'язку CAN ID 22 інвертора Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 10 сервоприводу Delta	○	-	0	-	-	N	R	0
SR763	Поточна швидкість двигуна зв'язку CAN ID 23 інвертора Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для ідентифікатора зв'язку ECAT 11 сервоприводу Delta	○	-	0	-	-	N	R	0
SR764	Поточна швидкість двигуна зв'язку CAN ID 24 інвертора Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для ідентифікатора зв'язку ECAT 12 сервоприводу Delta	○	-	0	-	-	N	R	0
SR765	Поточна швидкість двигуна зв'язку CAN ID 25 інвертора Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для ідентифікатора зв'язку ECAT 13 сервоприводу Delta	○	-	0	-	-	N	R	0
SR766	Поточна швидкість двигуна зв'язку CAN ID 26 інвертора Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для ідентифікатора зв'язку ECAT 14 сервоприводу Delta	○	-	0	-	-	N	R	0
SR767	Поточна швидкість двигуна зв'язку CAN ID 27 інвертора Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для ідентифікатора зв'язку ECAT 15 сервоприводу Delta	○	-	0	-	-	N	R	0
SR768	Поточна швидкість двигуна зв'язку CAN ID 28 інвертора Delta	○	○	0	-	-	N	R	0
	Код аварійного сигналу для зв'язку ECAT ID 16 сервоприводу Delta	○	-	0	-	-	N	R	0
SR771	Поточний крутний момент зв'язку CAN ID 21 інвертора Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 9 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR772	Поточний крутний момент зв'язку CAN ID 22 інвертора Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 10 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR773	Поточний крутний момент зв'язку CAN ID 23 інвертора Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 11 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR774	Поточний крутний момент зв'язку CAN ID 24 інвертора Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 12 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR775	Поточний крутний момент зв'язку CAN ID 25 інвертора Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 13 Delta	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
	серво (одиниця: 0,1%)								
SR776	Поточний крутний момент зв'язку CAN ID 26 інвертора Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 14 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR777	Поточний крутний момент зв'язку CAN ID 27 інвертора Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 15 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR778	Поточний крутний момент зв'язку CAN ID 28 інвертора Delta (одиниця: 0,1%)	○	○	0	-	-	N	R	0
	Поточний крутний момент зв'язку ECAT ID 16 сервоприводу Delta (одиниця: 0,1%)	○	-	0	-	-	N	R	0
SR781	Поточний стан DI зв'язку CAN ID 21 інвертора Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 9 сервоприводу Delta	○	-	0	-	-	N	R	0
SR782	Поточний стан DI зв'язку CAN ID 22 інвертора Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 10 сервоприводу Delta	○	-	0	-	-	N	R	0
SR783	Поточний стан DI зв'язку CAN ID 23 інвертора Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 11 сервоприводу Delta	○	-	0	-	-	N	R	0
SR784	Поточний стан DI зв'язку CAN ID 24 інвертора Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 12 сервоприводу Delta	○	-	0	-	-	N	R	0
SR785	Поточний стан DI зв'язку CAN ID 25 інвертора Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 13 сервоприводу Delta	○	-	0	-	-	N	R	0
SR786	Поточний стан DI зв'язку CAN ID 26 інвертора Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 14 сервоприводу Delta	○	-	0	-	-	N	R	0
SR787	Поточний стан DI зв'язку CAN ID 27 інвертора Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 15 сервоприводу Delta	○	-	0	-	-	N	R	0
SR788	Поточний стан DI зв'язку CAN ID 28 інвертора Delta	○	○	0	-	-	N	R	0
	Поточний стан DI зв'язку ECAT ID 16 сервоприводу Delta	○	-	0	-	-	N	R	0
SR791 SR792	Прочитати самовизначене значення з ідентифікатора зв'язку CAN 1 Delta sevo (32-розрядний)	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
	Прочитайте самовизначене значення з ідентифікатора зв'язку ECAT 1 Delta sevo (32-розрядний)	○	-	0	-	-	N	R	0
SR793	Прочитати самовизначене значення з ідентифікатора зв'язку CAN 2 Delta sevo (32-розрядний)	○	○	0	-	-	N	R	0
SR794	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 2 Delta sevo (32-розрядний)	○	-	0	-	-	N	R	0
SR795	Прочитати самовизначене значення з ідентифікатора зв'язку CAN 3 Delta sevo (32-розрядний)	○	○	0	-	-	N	R	0
SR796	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 3 Delta sevo (32-розрядний)	○	-	0	-	-	N	R	0
SR797	Прочитайте самовизначене значення з ідентифікатора зв'язку CAN 4 Delta sevo (32-розрядний)	○	○	0	-	-	N	R	0
SR798	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 4 Delta sevo (32-розрядний)	○	-	0	-	-	N	R	0
SR799	Прочитати самовизначене значення з ідентифікатора зв'язку CAN 5 Delta sevo (32-розрядний)	○	○	0	-	-	N	R	0
SR800	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 5 Delta sevo (32-розрядний)	○	-	0	-	-	N	R	0
SR801	Прочитати самовизначене значення з ідентифікатора зв'язку CAN 6 Delta sevo (32-розрядний)	○	○	0	-	-	N	R	0
SR802	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 6 Delta sevo (32-розрядний)	○	-	0	-	-	N	R	0
SR803	Прочитати самовизначене значення з ідентифікатора зв'язку CAN 7 Delta sevo (32-розрядний)	○	○	0	-	-	N	R	0
SR804	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 7 Delta sevo (32-розрядний)	○	-	0	-	-	N	R	0
SR805	Прочитати самовизначене значення з ідентифікатора зв'язку CAN 8 Delta sevo (32-розрядний)	○	○	0	-	-	N	R	0
SR806	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 8 Delta sevo (32-розрядний)	○	-	0	-	-	N	R	0
SR811	Прийнятий діапазон для ідентифікатора зв'язку CAN 1 сервоприводу Delta для визначення позиціонування завершено. (Якщо значення налаштування дорівнює 0, воно вказує на поточну позицію має збігатися з цільовою позицією, щоб визначити, що позиціонування завершено. Якщо значення налаштування дорівнює 5, це вказує на те, що відстань між поточною позицією та цільовою позицією може становити від +5 до -5, щоб ПЛК міг встановити позначку завершення.)	○	○	0	-	-	N	R/W	0
	Прийнятий діапазон для ідентифікатора зв'язку ECAT 1 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	-	-	N	R/W	0
SR812	Прийнятий діапазон для ідентифікатора зв'язку CAN 2 сервоприводу Delta для визначення позиціонування завершено.	○	○	0	-	-	N	R/W	0
	Прийнятий діапазон для ідентифікатора зв'язку ECAT 2 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	-	-	N	R/W	0

SR	функція	Серія AS300	Серія AS100/200	OFF ON	STOP RUN	RUN STOP	Засув	Атрибут	За замовч.
SR813	Прийнятий діапазон для ідентифікатора зв'язку CAN 3 сервоприводу Delta для визначення позиціонування завершено.	○	○	0	-	-	N	R/W	0
	Прийнятий діапазон для ідентифікатора зв'язку ECAT 3 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	-	-	N	R/W	0
SR814	Прийнятий діапазон для ідентифікатора зв'язку CAN 4 сервоприводу Delta для визначення позиціонування завершено.	○	○	0	-	-	N	R/W	0
	Прийнятий діапазон для ідентифікатора зв'язку ECAT 4 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	-	-	N	R/W	0
SR815	Прийнятий діапазон для ідентифікатора зв'язку CAN 5 сервоприводу Delta для визначення позиціонування завершено.	○	○	0	-	-	N	R/W	0
	Прийнятий діапазон для ідентифікатора зв'язку ECAT 5 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	-	-	N	R/W	0
SR816	Прийнятий діапазон для ідентифікатора зв'язку CAN 6 сервоприводу Delta для визначення позиціонування завершено.	○	○	0	-	-	N	R/W	0
	Прийнятий діапазон для ідентифікатора зв'язку ECAT 6 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	-	-	N	R/W	0
SR817	Прийнятий діапазон для ідентифікатора зв'язку CAN 7 сервоприводу Delta для визначення позиціонування завершено.	○	○	0	-	-	N	R/W	0
	Прийнятий діапазон для ідентифікатора зв'язку ECAT 7 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	-	-	N	R/W	0
SR818	Прийнятий діапазон для ідентифікатора зв'язку CAN 8 сервоприводу Delta для визначення позиціонування завершено.	○	○	0	-	-	N	R/W	0
	Прийнятий діапазон для ідентифікатора зв'язку ECAT 8 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	-	-	N	R/W	0
SR820	Код стану головного/підлеглого зв'язку CANopen DS301	○	○	-	-	-	Y	R	0
SR821	Код версії CANopen DS301	○	○	-	-	-	Y	R	-
SR822	Швидкість передачі даних CANopen (одиниця вимірювання: 1 кбіт/с)	○	○	-	-	-	H	R	125
*SR823	Кількість апаратних помилок отримання під час зв'язку CANopen (макс. 255)	○	○	0	-	-	N	R	0
*SR824	Кількість апаратних помилок надсилання під час зв'язку CANopen (макс. 255)	○	○	0	-	-	N	R	0
SR825	Код для головного стану зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR826	Стан підлеглого ID 1–16 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR827	Стан підлеглого ID 17–32 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR828	Стан підлеглого ID 33–48 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR829	Стан підлеглого ID 49–64 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR830	Стан підлеглого ID 1 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR831	Стан підпорядкованого ID 2 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1

SR	функція	Серія AS300	Серія AS100/200	OFF	STOP	RUN	Засув	Атрибут	За замовч.
				⇕ ON	⇕ RUN	⇕ STOP			
SR832	Стан підлеглого ID 3 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR833	Стан підлеглого ID 4 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR834	Стан підпорядкованого ID 5 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR835	Стан підпорядкованого ID 6 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR836	Стан підлеглого ID 7 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR837	Стан підлеглого ID 8 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR838	Стан підпорядкованого ідентифікатора 9 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR839	Стан підпорядкованого ID 10 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR840	Стан підпорядкованого ідентифікатора 11 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR841	Стан підлеглого ID 12 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR842	Стан підпорядкованого ID 13 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR843	Стан підлеглого ID 14 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR844	Стан підлеглого ID 15 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR845	Стан підпорядкованого ідентифікатора 16 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR846	Стан підпорядкованого ID 17 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR847	Стан підлеглого ID 18 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR848	Стан підлеглого ID 19 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR849	Стан підлеглого ID 20 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR850	Стан підпорядкованого ID 21 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR851	Стан підпорядкованого ідентифікатора 22 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR852	Стан підлеглого ID 23 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR853	Стан підлеглого ID 24 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR854	Стан підпорядкованого ідентифікатора 25 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR855	Стан підлеглого ID 26 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR856	Стан підлеглого ідентифікатора 27 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR857	Стан підлеглого ідентифікатора 28 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR858	Стан підпорядкованого ідентифікатора 29 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR859	Стан підлеглого ID 30 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR860	Стан підпорядкованого ідентифікатора 31 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR861	Стан підпорядкованого ідентифікатора 32 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR862	Стан підпорядкованого ідентифікатора 33 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR863	Стан підпорядкованого ідентифікатора 34 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR864	Стан підпорядкованого ідентифікатора 35 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR865	Стан підпорядкованого ідентифікатора 36 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR866	Стан підлеглого ID 37 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR867	Стан підпорядкованого ідентифікатора 38 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR868	Стан підпорядкованого ідентифікатора 39 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR869	Стан підлеглого ID 40 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1
SR870	Стан підпорядкованого ідентифікатора 41 у зв'язку CANopen DS301	o	o	-1	-	-	N	R	-1

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
SR871	Стан підпорядкованого ідентифікатора 42 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR872	Стан підлеглого ID 43 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR873	Стан підпорядкованого ідентифікатора 44 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR874	Стан підлеглого ID 45 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR875	Стан підпорядкованого ідентифікатора 46 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR876	Стан підпорядкованого ідентифікатора 47 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR877	Стан підлеглого ID 48 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR878	Стан підпорядкованого ідентифікатора 49 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR879	Стан підпорядкованого ідентифікатора 50 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR880	Стан підпорядкованого ідентифікатора 51 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR881	Стан підпорядкованого ідентифікатора 52 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR882	Стан підпорядкованого ідентифікатора 53 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR883	Стан підпорядкованого ідентифікатора 54 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR884	Стан підпорядкованого ідентифікатора 55 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR885	Стан підпорядкованого ідентифікатора 56 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR886	Стан підпорядкованого ідентифікатора 57 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR887	Стан підпорядкованого ідентифікатора 58 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR888	Стан підпорядкованого ідентифікатора 59 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR889	Стан підпорядкованого ідентифікатора 60 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR890	Стан підпорядкованого ідентифікатора 61 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR891	Стан підпорядкованого ідентифікатора 62 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR892	Стан підпорядкованого ідентифікатора 63 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR893	Стан підпорядкованого ідентифікатора 64 у зв'язку CANopen DS301	○	○	-1	-	-	N	R	-1
SR900	Кількість зразків у реєстраторі даних (32-розрядний)	○	○	0	-	-	N	R	0
SR901		○	○						
SR902	Код для виконання реєстратора та карти пам'яті (працює з SM456).	○	○	0	-	-	N	R/W	0
*1000 SR	IP-адреса Ethernet (32-розрядна)	○	○	-	-	-	H	R/W	0
*SR1001		○	○						
*SR1002	Адреса маски мережі Ethernet (32-розрядна)	○	○	-	-	-	H	R/W	0
*SR1003		○	○						
*SR1004	Адреса шлюзу Ethernet (32-розрядна)	○	○	-	-	-	H	R/W	0
*SR1005		○	○						
*SR1006	Тривалість TCP-з'єднання (сек.)	○	○	-	-	-	H	R/W	30
*SR1007	Швидкість передачі Ethernet	○	○	0	-	-	N	R	0
*SR1009	Кількість TCP-з'єднань	○	○	0	-	-	N	R	0
*SR1010	Конкретний час повторного надсилання через з'єднання TCP (мс)	○	○	-	-	-	N	R/W	20
*SR1011	Номер підключення сервера Modbus/TCP	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇕ ON	STOP ⇕ RUN	RUN ⇕ STOP	Засув	Атрибут	За замовч.
*SR1012	Номер підключення клієнта Modbus/TCP	○	○	0	-	-	N	R	0
*SR1013	Номер підключення адаптера EtherNet/IP	○	○	0	-	-	N	R	0
*SR1014	Номер підключення сканера EtherNet/IP	○	○	0	-	-	N	R	0
*SR1015	Тайм-аут отримання пакета сервера Modbus/TCP (мс)	○	○	10	-	-	N	R/W	10
*SR1016	Час затримки для надсилання даних після встановлення з'єднання клієнта Modbus/TCP. (мс) (0~1000)	○	○	0	-	-	N	R/W	0
*SR1017	Максимальна кількість пакетів Ethernet, які можна обробити за один цикл сканування.	○	○	-	-	-	Y	R/W	80
*SR1018	Кількість пакетів Ethernet, оброблених в останньому циклі сканування.	○	○	0	-	-	N	R	0
*SR1019	Накопичена кількість пакетів Ethernet, які відкидаються після заповнення буферної зони.	○	○	0	-	-	N	R/W	0
*SR1020	Стан з'єднання EtherNet/IP 1	○		0	-	-	N	R	0
*SR1021	Стан з'єднання EtherNet/IP 2	○	○	0	-	-	N	R	0
*SR1022	Стан з'єднання EtherNet/IP 3	○	○	0	-	-	N	R	0
*SR1023	Стан з'єднання EtherNet/IP 4	○		0	-	-	N	R	0
*SR1024	Стан з'єднання EtherNet/IP 5	○	○	0	-	-	N	R	0
*SR1025	Стан з'єднання EtherNet/IP 6	○		0	-	-	N	R	0
*SR1026	Стан з'єднання EtherNet/IP 7	○	○	0	-	-	N	R	0
*SR1027	Стан з'єднання EtherNet/IP 8	○		0	-	-	N	R	0
*SR1028	Стан з'єднання EtherNet/IP 9	○	○	0	-	-	N	R	0
*SR1029	Стан з'єднання EtherNet/IP 10	○	○	0	-	-	N	R	0
*SR1030	Стан з'єднання EtherNet/IP 11	○	○	0	-	-	N	R	0
*SR1031	Стан з'єднання EtherNet/IP 12	○	○	0	-	-	N	R	0
*SR1032	Стан з'єднання EtherNet/IP 13	○	○	0	-	-	N	R	0
*SR1033	Стан з'єднання EtherNet/IP 14	○	○	0	-	-	N	R	0
*SR1034	Стан з'єднання EtherNet/IP 15	○	○	0	-	-	N	R	0
*SR1035	Стан з'єднання EtherNet/IP 16	○	○	0	-	-	N	R	0
*SR1036	Стан з'єднання EtherNet/IP 17	○	○	0	-	-	N	R	0
*SR1037	Стан з'єднання EtherNet/IP 18	○	○	0	-	-	N	R	0
*SR1038	Стан з'єднання EtherNet/IP 19	○	○	0	-	-	N	R	0
*SR1039	Стан з'єднання EtherNet/IP 20	○	○	0	-	-	N	R	0
*SR1040	Стан з'єднання EtherNet/IP 21	○	○	0	-	-	N	R	0
*SR1041	Стан з'єднання EtherNet/IP 22	○	○	0	-	-	N	R	0
*SR1042	Стан з'єднання EtherNet/IP 23	○	○	0	-	-	N	R	0
*SR1043	Стан з'єднання EtherNet/IP 24	○	○	0	-	-	N	R	0
*SR1044	Стан з'єднання EtherNet/IP 25	○	○	0	-	-	N	R	0
*SR1045	Стан з'єднання EtherNet/IP 26	○	○	0	-	-	N	R	0
*SR1046	Стан з'єднання EtherNet/IP 27	○	○	0	-	-	N	R	0
*SR1047	Стан з'єднання EtherNet/IP 28	○	○	0	-	-	N	R	0
*SR1048	Стан з'єднання EtherNet/IP 29	○	○	0	-	-	N	R	0
*SR1049	Стан з'єднання EtherNet/IP 30	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1050	Стан з'єднання EtherNet/IP 31	○	○	0	-	-	N	R	0
*SR1051	Стан з'єднання EtherNet/IP 32	○	○	0	-	-	N	R	0
SR1052	Час оновлення для з'єднання EtherNet/IP 1 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1053	Час оновлення для з'єднання EtherNet/IP 2 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1054	Час оновлення для з'єднання EtherNet/IP 3 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1055	Час оновлення для з'єднання EtherNet/IP 4 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1056	Час оновлення для з'єднання EtherNet/IP 5 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1057	Час оновлення для з'єднання EtherNet/IP 6 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1058	Час оновлення для з'єднання EtherNet/IP 7 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1059	Час оновлення для з'єднання EtherNet/IP 8 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1060	Час оновлення для з'єднання EtherNet/IP 9 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1061	Час оновлення для з'єднання EtherNet/IP 10 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1062	Час оновлення для з'єднання EtherNet/IP 11 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1063	Час оновлення для з'єднання EtherNet/IP 12 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1064	Час оновлення для з'єднання EtherNet/IP 13 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1065	Час оновлення для з'єднання EtherNet/IP 14 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1066	Час оновлення для з'єднання EtherNet/IP 15 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1067	Час оновлення для з'єднання EtherNet/IP 16 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1068	Час оновлення для з'єднання EtherNet/IP 17 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1069	Час оновлення для з'єднання EtherNet/IP 18 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1070	Час оновлення для з'єднання EtherNet/IP 19 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1071	Час оновлення для з'єднання EtherNet/IP 20 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1072	Час оновлення для з'єднання EtherNet/IP 21 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1073	Час оновлення для з'єднання EtherNet/IP 22 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1074	Час оновлення для з'єднання EtherNet/IP 23 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1075	Час оновлення для з'єднання EtherNet/IP 24 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1076	Час оновлення для з'єднання EtherNet/IP 25 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1077	Час оновлення для з'єднання EtherNet/IP 26 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1078	Час оновлення для з'єднання EtherNet/IP 27 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1079	Час оновлення для з'єднання EtherNet/IP 28 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1080	Час оновлення для з'єднання EtherNet/IP 29 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1081	Час оновлення для з'єднання EtherNet/IP 30 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1082	Час оновлення для з'єднання EtherNet/IP 31 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1083	Час оновлення для з'єднання EtherNet/IP 32 (одиниця: мс)	○	○	0	-	-	N	R	0
SR1092	Порт сервера Modbus TCP (ЦП ПЛК як сервер); доступно для FW V1.10.20 або новішої версії.	○	○	502	-	-	N	R/W	502
SR1093	Порт клієнта Modbus TCP (процесор ПЛК як клієнт); доступно для FW V1.10.20 або новішої версії.	○	○	502	-	-	N	R/W	502
SR1100	Значення лічильника вхідних пакетів (старший байт)	○	○	0	-	-	N	R	0
SR1101	Значення лічильника вхідних пакетів (молодший байт)		○						

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
SR1102	Значення лічильника вхідних октетів (старший байт)	○	○	0	-	-	N	R	0
SR1103	Значення лічильника вхідних октетів (молодший байт)	○	○	0	-	-	N	R	0
SR1104	Значення вихідного лічильника пакетів (старший байт)	○	○	0	-	-	N	R	0
SR1105	Значення вихідного лічильника пакетів (молодший байт)	○	○	0	-	-	N	R	0
SR1106	Значення вихідного лічильника октетів (старший байт)	○	○	0	-	-	N	R	0
SR1107	Значення вихідного лічильника октетів (молодший байт)	○	○	0	-	-	N	R	0
SR1110	Код для зміни параметра NTP (NTP має бути ввімкнено та введено код команди зміни 16#1234. Щойно ПЛК завершить зміну, код зміни буде автоматично видалено.	○	○	0	0	-	N	R/W	0
SR1111	IP-байт сервера NTP (старший байт)	○	○	0	0	-	N	R/W	0
SR1112	IP-байт сервера NTP (молодший байт)	○	○	0	0	-	N	R/W	0
SR1113	Час оновлення циклу NTP (одиниця: хвилина)	○	○	0	0	-	N	R/W	0
SR1114	Часовий пояс NTP (одиниця: година)	○	○	0	0	-	N	R/W	0
SR1116	Лічильник електронної пошти	○	○	0	-	-	N	R	0
SR1117	Лічильник помилок електронної пошти	○	○	0	-	-	N	R	0
*SR1120	Фактичний час підключення для обміну даними через підключення Ethernet 1 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1121	Фактичний час підключення для обміну даними через підключення Ethernet 2 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1122	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 3 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1123	Фактичний час підключення для обміну даними через підключення Ethernet 4 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1124	Фактичний час підключення для обміну даними через підключення Ethernet 5 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1125	Фактичний час підключення для обміну даними через підключення Ethernet 6 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1126	Фактичний час підключення для обміну даними через підключення Ethernet 7 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1127	Фактичний час підключення для обміну даними через підключення Ethernet 8 (одиниця: час)	○		0	-	-	N	R	0
*SR1128	Фактичний час підключення для обміну даними через підключення Ethernet 9 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1129	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 10 (одиниця: час)	○		0	-	-	N	R	0
*SR1130	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 11 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1131	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 12 (одиниця: час)	○		0	-	-	N	R	0
*SR1132	Фактичний час підключення для обміну даними через підключення Ethernet 13 (одиниця: час)	○		0	-	-	N	R	0
*SR1133	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 14 (одиниця: час)	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1134	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 15 (одиниця: час)	○		0	-	-	N	R	0
*SR1135	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 16 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1136	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 17 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1137	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 18 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1138	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 19 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1139	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 20 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1140	Фактичний час підключення для обміну даними через підключення Ethernet 21 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1141	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 22 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1142	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 23 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1143	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 24 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1144	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 25 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1145	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 26 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1146	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 27 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1147	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 28 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1148	Фактичний час підключення для обміну даними через підключення Ethernet 29 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1149	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 30 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1150	Фактичний час підключення для обміну даними через підключення Ethernet 31 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1151	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 32 (одиниця: час)	○	○	0	-	-	N	R	0
*SR1152	Код помилки обміну даними через з'єднання Ethernet 1	○	○	0	-	-	N	R	0
*SR1153	Код помилки обміну даними через з'єднання Ethernet 2	○	○	0	-	-	N	R	0
*SR1154	Код помилки обміну даними через з'єднання Ethernet 3	○	○	0	-	-	N	R	0
*SR1155	Код помилки обміну даними через з'єднання Ethernet 4	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1156	Код помилки обміну даними через з'єднання Ethernet 5	○	○	0	-	-	N	R	0
*SR1157	Код помилки обміну даними через з'єднання Ethernet 6	○	○	0	-	-	N	R	0
*SR1158	Код помилки обміну даними через з'єднання Ethernet 7	○	○	0	-	-	N	R	0
*SR1159	Код помилки обміну даними через з'єднання Ethernet 8	○	○	0	-	-	N	R	0
*SR1160	Код помилки обміну даними через з'єднання Ethernet 9	○	○	0	-	-	N	R	0
*SR1161	Код помилки обміну даними через з'єднання Ethernet 10	○	○	0	-	-	N	R	0
*SR1162	Код помилки обміну даними через з'єднання Ethernet 11	○	○	0	-	-	N	R	0
*SR1163	Код помилки обміну даними через з'єднання Ethernet 12	○	○	0	-	-	N	R	0
*SR1164	Код помилки обміну даними через з'єднання Ethernet 13	○	○	0	-	-	N	R	0
*SR1165	Код помилки обміну даними через з'єднання Ethernet 14	○	○	0	-	-	N	R	0
*SR1166	Код помилки обміну даними через з'єднання Ethernet 15	○	○	0	-	-	N	R	0
*SR1167	Код помилки обміну даними через з'єднання Ethernet 16	○	○	0	-	-	N	R	0
*SR1168	Код помилки обміну даними через з'єднання Ethernet 17	○	○	0	-	-	N	R	0
*SR1169	Код помилки обміну даними через з'єднання Ethernet 18	○	○	0	-	-	N	R	0
*SR1170	Код помилки обміну даними через з'єднання Ethernet 19	○	○	0	-	-	N	R	0
*SR1171	Код помилки обміну даними через з'єднання Ethernet 20	○	○	0	-	-	N	R	0
*SR1172	Код помилки обміну даними через з'єднання Ethernet 21	○	○	0	-	-	N	R	0
*SR1173	Код помилки обміну даними через з'єднання Ethernet 22	○	○	0	-	-	N	R	0
*SR1174	Код помилки обміну даними через з'єднання Ethernet 23	○	○	0	-	-	N	R	0
*SR1175	Код помилки обміну даними через з'єднання Ethernet 24	○	○	0	-	-	N	R	0
*SR1176	Код помилки обміну даними через з'єднання Ethernet 25	○	○	0	-	-	N	R	0
*SR1177	Код помилки обміну даними через з'єднання Ethernet 26	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1178	Код помилки обміну даними через з'єднання Ethernet 27	○	○	0	-	-	N	R	0
*SR1179	Код помилки обміну даними через з'єднання Ethernet 28	○	○	0	-	-	N	R	0
*SR1180	Код помилки обміну даними через Ethernet-з'єднання 29	○	○	0	-	-	N	R	0
*SR1181	Код помилки обміну даними через з'єднання Ethernet 30	○	○	0	-	-	N	R	0
*SR1182	Код помилки обміну даними через з'єднання Ethernet 31	○	○	0	-	-	N	R	0
*SR1183	Код помилки обміну даними через з'єднання Ethernet 32	○	○	0	-	-	N	R	0
SR1191	Поточний стан DO ідентифікатора зв'язку ECAT 9 сервоприводу Delta	○	-	0	0	-	N	R	0
SR1192	Поточний стан DO ідентифікатора зв'язку ECAT 10 сервоприводу Delta	○	-	0	0	-	N	R	0
SR1193	Поточний стан DO ідентифікатора зв'язку ECAT 11 сервоприводу Delta	○	-	0	0	-	N	R	0
SR1194	Поточний стан DO ідентифікатора зв'язку ECAT 12 сервоприводу Delta	○	-	0	0	-	N	R	0
SR1195	Поточний стан DO ідентифікатора зв'язку ECAT 13 сервоприводу Delta	○	-	0	0	-	N	R	0
SR1196	Поточний стан DO ідентифікатора зв'язку ECAT 14 сервоприводу Delta	○	-	0	0	-	N	R	0
SR1197	Поточний стан DO ідентифікатора зв'язку ECAT 15 сервоприводу Delta	○	-	0	0	-	N	R	0
SR1198	Поточний стан DO ідентифікатора зв'язку ECAT 16 сервоприводу Delta	○	-	0	0	-	N	R	0
SR1201	Поточне положення ідентифікатора зв'язку ECAT 9 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1202	Поточна позиція зв'язку ECAT ID 10 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1203	Поточна позиція ідентифікатора зв'язку ECAT 11 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1204	Поточна позиція зв'язку ECAT ID 12 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1205	Поточне положення ідентифікатора зв'язку ECAT 13 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1206	Поточна позиція зв'язку ECAT ID 14 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1207	Поточна позиція зв'язку ECAT ID 15 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1208	Поточне положення ідентифікатора зв'язку ECAT 13 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1209	Поточна позиція зв'язку ECAT ID 14 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1210	Поточна позиція зв'язку ECAT ID 15 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1211	Поточна позиція зв'язку ECAT ID 16 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1212	Поточна позиція зв'язку ECAT ID 17 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1213	Поточна позиція зв'язку ECAT ID 18 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1214	Поточна позиція зв'язку ECAT ID 19 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ON	STOP RUN	RUN STOP	Засув	Атрибут	За замовч.
SR1215	Поточна позиція зв'язку ECAT ID 16 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1221	Цільове положення ідентифікатора зв'язку ECAT 9 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1222	Цільове положення ідентифікатора зв'язку ECAT 10 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1223	Цільове положення ідентифікатора зв'язку ECAT 11 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1224	Цільове положення ідентифікатора зв'язку ECAT 12 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1225	Цільове положення ідентифікатора зв'язку ECAT 13 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1226	Цільове положення ідентифікатора зв'язку ECAT 14 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1227	Цільове положення ідентифікатора зв'язку ECAT 15 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1228	Цільове положення ідентифікатора зв'язку ECAT 16 сервоприводу Delta (32-розрядний)	○	-	0	0	-	N	R	0
SR1229	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 9 Delta sevo (32-розрядний)	○	-	0	0	-	N	R	0
SR1230	Прочитайте самовизначене значення з ідентифікатора зв'язку ECAT 10 Delta sevo (32-розрядний)	○	-	0	0	-	N	R	0
SR1231	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 11 Delta sevo (32-розрядний)	○	-	0	0	-	N	R	0
SR1232	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 12 Delta sevo (32-розрядний)	○	-	0	0	-	N	R	0
SR1233	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 13 Delta sevo (32-розрядний)	○	-	0	0	-	N	R	0
SR1234	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 14 Delta sevo (32-розрядний)	○	-	0	0	-	N	R	0
SR1235	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 15 Delta sevo (32-розрядний)	○	-	0	0	-	N	R	0
SR1236	Прочитати самовизначене значення з ідентифікатора зв'язку ECAT 16 Delta sevo (32-розрядний)	○	-	0	0	-	N	R	0
SR1241	Прийнятий діапазон для ідентифікатора зв'язку ECAT 9 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	0	-	N	R/W	0
SR1242	Прийнятий діапазон для ідентифікатора зв'язку ECAT 10 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	0	-	N	R/W	0
SR1243	Прийнятий діапазон для ідентифікатора зв'язку ECAT 11 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	0	-	N	R/W	0
SR1244	Прийнятий діапазон для ідентифікатора зв'язку ECAT 12 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	0	-	N	R/W	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SR1265	Прийнятний діапазон для ідентифікатора зв'язку ECAT 13 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	0	-	N	R/W	0
SR1266	Прийнятний діапазон для ідентифікатора зв'язку ECAT 14 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	0	-	N	R/W	0
SR1267	Прийнятний діапазон для ідентифікатора зв'язку ECAT 15 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	0	-	N	R/W	0
SR1268	Прийнятний діапазон для ідентифікатора зв'язку ECAT 16 сервоприводу Delta для визначення позиціонування завершено.	○	-	0	0	-	N	R/W	0
SR1291	Вісь 1 буде виводити наступний сегмент або певний номер сегмента під час виконання інструкції TPO; це має працювати з SM591.	○	○	0	0	-	N	R/W	0
SR1292	Вісь 2 буде виводити наступний сегмент або певний номер сегмента під час виконання інструкції TPO; це має працювати з SM592.	○	○	0	0	-	N	R/W	0
SR1293	Вісь 3 буде виводити наступний сегмент або певний номер сегмента під час виконання інструкції TPO; це має працювати з SM593.	○	○	0	0	-	N	R/W	0
SR1294	Вісь 4 буде виводити наступний сегмент або певний номер сегмента під час виконання інструкції TPO; це має працювати з SM594.	○	○	0	0	-	N	R/W	0
SR1295	Вісь 5 буде виводити наступний сегмент або певний номер сегмента під час виконання інструкції TPO; це має працювати з SM595.	○	○	0	0	-	N	R/W	0
SR1296	Вісь 6 буде виводити наступний сегмент або певний номер сегмента під час виконання інструкції TPO; це має працювати з SM596.	○	○	0	0	-	N	R/W	0
*SR1312	Код зв'язку для підключення RTU-EN01 1	○	○	0	-	-	N	R	0
*SR1313	Код зв'язку для з'єднання RTU-EN01 2	○	○	0	-	-	N	R	0
*SR1314	Код зв'язку для з'єднання RTU-EN01 3	○	○	0	-	-	N	R	0
*SR1315	Код зв'язку для з'єднання RTU-EN01 4	○	○	0	-	-	N	R	0
SR1318	Лічильник входів розетки	○	○	0	-	-	N	R	0
SR1319	Лічильник виходу розетки	○	○	0	-	-	N	R	0
SR1320	Лічильник помилок сокета	○	○	0	-	-	N	R	0
*SR1335	Фактичний час циклу підключення 1–32 для обміну даними через COM1; одиниця: мс	○	○	0	-	-	N	R	0
*SR1336	Номер підключення, яке в даний момент здійснює циклічний обмін даними через COM1	○	○	0	-	-	N	R	0
*SR1340	Код помилки обміну даними через з'єднання COM1 1	○	○	0	-	-	N	R	0
*SR1341	Код помилки обміну даними через з'єднання COM1 2	○	○	0	-	-	N	R	0
*SR1342	Код помилки обміну даними через з'єднання COM1 3	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
*SR1343	Код помилки обміну даними через з'єднання COM1 4	○	○	0	-	-	N	R	0
*SR1344	Код помилки обміну даними через з'єднання COM1 5	○	○	0	-	-	N	R	0
*SR1345	Код помилки обміну даними через з'єднання COM1 6	○	○	0	-	-	N	R	0
*SR1346	Код помилки обміну даними через з'єднання COM1 7	○	○	0	-	-	N	R	0
*SR1347	Код помилки обміну даними через з'єднання COM1 8	○	○	0	-	-	N	R	0
*SR1348	Код помилки обміну даними через з'єднання COM1 9	○	○	0	-	-	N	R	0
*SR1349	Код помилки обміну даними через з'єднання COM1 10	○	○	0	-	-	N	R	0
*SR1350	Код помилки обміну даними через з'єднання COM1 11	○	○	0	-	-	N	R	0
*SR1351	Код помилки обміну даними через з'єднання COM1 12	○	○	0	-	-	N	R	0
*SR1352	Код помилки обміну даними через з'єднання COM1 13	○	○	0	-	-	N	R	0
*SR1353	Код помилки обміну даними через з'єднання COM1 14	○	○	0	-	-	N	R	0
*SR1354	Код помилки обміну даними через з'єднання COM1 15	○	○	0	-	-	N	R	0
*SR1355	Код помилки обміну даними через з'єднання COM1 16	○	○	0	-	-	N	R	0
*SR1356	Код помилки обміну даними через з'єднання COM1 17	○	○	0	-	-	N	R	0
*SR1357	Код помилки обміну даними через з'єднання COM1 18	○	○	0	-	-	N	R	0
*SR1358	Код помилки обміну даними через з'єднання COM1 19	○	○	0	-	-	N	R	0
*SR1359	Код помилки обміну даними через з'єднання COM1 20	○	○	0	-	-	N	R	0
*SR1360	Код помилки обміну даними через з'єднання COM1 21	○	○	0	-	-	N	R	0
*SR1361	Код помилки обміну даними через з'єднання COM1 22	○	○	0	-	-	N	R	0
*SR1362	Код помилки обміну даними через з'єднання COM1 23	○	○	0	-	-	N	R	0
*SR1363	Код помилки обміну даними через з'єднання COM1 24	○	○	0	-	-	N	R	0
*SR1364	Код помилки обміну даними через з'єднання COM1 25	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1365	Код помилки обміну даними через з'єднання COM1 26	○	○	0	-	-	N	R	0
*SR1366	Код помилки обміну даними через з'єднання COM1 27	○	○	0	-	-	N	R	0
*SR1367	Код помилки обміну даними через з'єднання COM1 28	○	○	0	-	-	N	R	0
*SR1368	Код помилки обміну даними через з'єднання COM1 29	○	○	0	-	-	N	R	0
*SR1369	Код помилки обміну даними через з'єднання COM1 30	○	○	0	-	-	N	R	0
*SR1370	Код помилки обміну даними через з'єднання COM1 31	○	○	0	-	-	N	R	0
*SR1371	Код помилки обміну даними через з'єднання COM1 32	○	○	0	-	-	N	R	0
*SR1375	Фактичний час циклу підключення 1–32 для обміну даними через COM2; одиниця: мс	○	○	0	-	-	N	R	0
*SR1376	Номер підключення, яке в даний момент здійснює циклічний обмін даними через COM2	○	○	0	-	-	N	R	0
*SR1380	Код помилки обміну даними через з'єднання COM2 1	○	○	0	-	-	N	R	0
SR1381	Код помилки обміну даними через з'єднання COM2 2	○	○	0	-	-	N	R	0
SR1382	Код помилки обміну даними через з'єднання COM2 3	○	○	0	-	-	N	R	0
SR1383	Код помилки обміну даними через з'єднання COM2 4	○	○	0	-	-	N	R	0
SR1384	Код помилки обміну даними через з'єднання COM2 5	○	○	0	-	-	N	R	0
SR1385	Код помилки обміну даними через з'єднання COM2 6	○	○	0	-	-	N	R	0
SR1386	Код помилки обміну даними через з'єднання COM2 7	○	○	0	-	-	N	R	0
SR1387	Код помилки обміну даними через з'єднання COM2 8	○	○	0	-	-	N	R	0
SR1388	Код помилки обміну даними через з'єднання COM2 9	○	○	0	-	-	N	R	0
SR1389	Код помилки обміну даними через з'єднання COM2 10	○	○	0	-	-	N	R	0
SR1390	Код помилки обміну даними через з'єднання COM2 11	○	○	0	-	-	N	R	0
SR1391	Код помилки обміну даними через з'єднання COM2 12	○	○	0	-	-	N	R	0
SR1392	Код помилки обміну даними через з'єднання COM2 13	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
SR1393	Код помилки обміну даними через з'єднання COM2 14	○	○	0	-	-	N	R	0
SR1394	Код помилки обміну даними через з'єднання COM2 15	○	○	0	-	-	N	R	0
SR1395	Код помилки обміну даними через з'єднання COM2 16	○	○	0	-	-	N	R	0
SR1396	Код помилки обміну даними через з'єднання COM2 17	○	○	0	-	-	N	R	0
SR1397	Код помилки обміну даними через з'єднання COM2 18	○	○	0	-	-	N	R	0
SR1398	Код помилки обміну даними через з'єднання COM2 19	○	○	0	-	-	N	R	0
SR1399	Код помилки обміну даними через з'єднання COM2 20	○	○	0	-	-	N	R	0
1400 SR	Код помилки обміну даними через з'єднання COM2 21	○	○	0	-	-	N	R	0
SR1401	Код помилки обміну даними через з'єднання COM2 22	○	○	0	-	-	N	R	0
SR1402	Код помилки обміну даними через з'єднання COM2 23	○	○	0	-	-	N	R	0
SR1403	Код помилки обміну даними через з'єднання COM2 24	○	○	0	-	-	N	R	0
SR1404	Код помилки обміну даними через з'єднання COM2 25	○	○	0	-	-	N	R	0
SR1405	Код помилки обміну даними через з'єднання COM2 26	○	○	0	-	-	N	R	0
SR1406	Код помилки обміну даними через з'єднання COM2 27	○	○	0	-	-	N	R	0
SR1407	Код помилки обміну даними через з'єднання COM2 28	○	○	0	-	-	N	R	0
SR1408	Код помилки обміну даними через з'єднання COM2 29	○	○	0	-	-	N	R	0
SR1409	Код помилки обміну даними через з'єднання COM2 30	○	○	0	-	-	N	R	0
SR1410	Код помилки обміну даними через з'єднання COM2 31	○	○	0	-	-	N	R	0
SR1411	Код помилки обміну даними через з'єднання COM2 32	○	○	0	-	-	N	R	0
SR1430	Потім номер підключення OPC UA Server	○	-	0	-	-	N	R	0
SR1431	Номер підключення OPC UA Client	○	-	0	-	-	N	R	0
*SR1435	Фактичний час циклу для обміну даними через функціональну картку 1; одиниця: мс	○	-	0	-	-	N	R	0
*SR1436	Номер підключення, яке в даний момент виконує циклічний обмін даними через функціональну картку 1	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1440	Помилка підключення обміну даними 1 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1441	Помилка підключення обміну даними 2 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1442	Помилка підключення обміну даними 3 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1443	Помилка підключення обміну даними 4 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1444	Помилка підключення обміну даними 5 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1445	Помилка підключення обміну даними 6 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1446	Помилка підключення обміну даними 7 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1447	Помилка підключення обміну даними 8 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1448	Помилка підключення обміну даними 9 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1449	Помилка підключення обміну даними 10 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1450	Помилка підключення обміну даними 11 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1451	Помилка підключення обміну даними 12 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1452	Помилка підключення обміну даними 13 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1453	Помилка підключення обміну даними 14 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1454	Помилка підключення обміну даними 15 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1455	Помилка підключення обміну даними 16 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1456	Помилка підключення обміну даними 17 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1457	Помилка підключення обміну даними 18 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1458	Помилка підключення обміну даними 19 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1459	Помилка підключення обміну даними 20 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1460	Помилка підключення обміну даними 21 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1461	Помилка підключення обміну даними 22 через функціональну картку 1	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1462	Помилка підключення обміну даними 23 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1463	Помилка підключення обміну даними 24 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1464	Помилка підключення обміну даними 25 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1465	Помилка підключення обміну даними 26 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1466	Помилка підключення обміну даними 27 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1467	Помилка підключення обміну даними 28 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1468	Помилка підключення обміну даними 29 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1469	Помилка підключення обміну даними 30 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1470	Помилка підключення обміну даними 31 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1471	Помилка підключення обміну даними 32 через функціональну картку 1	○	-	0	-	-	N	R	0
*SR1475	Фактичний час циклу для обміну даними через функціональну картку 2; одиниця: мс	○	-	0	-	-	N	R	0
*SR1476	Номер підключення, яке в даний момент виконує циклічний обмін даними через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1480	Помилка підключення обміну даними 1 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1481	Помилка підключення обміну даними 2 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1482	Помилка підключення обміну даними 3 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1483	Помилка підключення обміну даними 4 через функціональну картку 2	○	-	0	-	-	N	R	0
SR1484	Помилка підключення обміну даними 5 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1485	Помилка підключення обміну даними 6 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1486	Помилка підключення обміну даними 7 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1487	Помилка підключення обміну даними 8 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1488	Помилка підключення обміну даними 9 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1489	Помилка підключення обміну даними 10 через функціональну картку 2	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1490	Помилка підключення обміну даними 11 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1491	Помилка підключення обміну даними 12 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1492	Помилка підключення обміну даними 13 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1493	Помилка підключення обміну даними 14 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1494	Помилка підключення обміну даними 15 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1495	Помилка підключення обміну даними 16 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1496	Помилка підключення обміну даними 17 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1497	Помилка підключення обміну даними 18 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1498	Помилка підключення обміну даними 19 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1499	Помилка підключення обміну даними 20 через функціональну картку 2	○	-	0	-	-	N	R	0
*1500 SR	Помилка підключення обміну даними 21 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1501	Помилка підключення обміну даними 22 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1502	Помилка підключення обміну даними 23 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1503	Помилка підключення обміну даними 24 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1504	Помилка підключення обміну даними 25 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1505	Помилка підключення обміну даними 26 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1506	Помилка підключення обміну даними 27 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1507	Помилка підключення обміну даними 28 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1508	Помилка підключення обміну даними 29 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1509	Помилка підключення обміну даними 30 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1510	Помилка підключення обміну даними 31 через функціональну картку 2	○	-	0	-	-	N	R	0
*SR1511	Помилка підключення обміну даними 32 через функціональну картку 2	○	-	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1520	Фактичний час зв'язку для з'єднання обміну даними 1 через AS-FEN02; одиниця: мс	○	-	0	-	-	N	R	0
*SR1521	Фактичний час зв'язку для з'єднання обміну даними 2 через AS-FEN02; одиниця: мс	○	-	0	-	-	N	R	0
*SR1522	Фактичний час зв'язку для з'єднання обміну даними 3 через AS-FEN02; одиниця: мс	○	-	0	-	-	N	R	0
*SR1523	Фактичний час зв'язку для з'єднання обміну даними 4 через AS-FEN02; одиниця: мс	○	-	0	-	-	N	R	0
*SR1524	Фактичний час зв'язку для з'єднання обміну даними 5 через AS-FEN02; одиниця: мс	○	-	0	-	-	N	R	0
*SR1525	Фактичний час зв'язку для з'єднання обміну даними 6 через AS-FEN02; одиниця: мс	○	-	0	-	-	N	R	0
*SR1526	Фактичний час зв'язку для з'єднання обміну даними 7 через AS-FEN02; одиниця: мс	○	-	0	-	-	N	R	0
*SR1527	Фактичний час зв'язку для з'єднання обміну даними 8 через AS-FEN02; одиниця: мс	○	-	0	-	-	N	R	0
*SR1528	Помилка підключення обміну даними 1 через AS-FEN02	○	-	0	-	-	N	R	0
*SR1529	Помилка підключення обміну даними 2 через AS-FEN02	○	-	0	-	-	N	R	0
*SR1530	Помилка підключення обміну даними 3 через AS-FEN02	○	-	0	-	-	N	R	0
*SR1531	Помилка підключення обміну даними 4 через AS-FEN02	○	-	0	-	-	N	R	0
*SR1532	Помилка підключення обміну даними 5 через AS-FEN02	○	-	0	-	-	N	R	0
*SR1533	Помилка підключення обміну даними 6 через AS-FEN02	○	-	0	-	-	N	R	0
*SR1534	Помилка підключення обміну даними 7 через AS-FEN02	○	-	0	-	-	N	R	0
*SR1535	Помилка підключення обміну даними 8 через AS-FEN02	○	-	0	-	-	N	R	0
*SR1536	Номер підключення TCP AS-FEN02 / AS-FECAT	○	-	0	-	-	N	R	0
*SR1537	Кількість підключень до сервера Modbus/TCP через AS-FEN02 / AS-FECAT	○	-	0	-	-	N	R	0
*SR1538	Кількість підключень клієнта Modbus/TCP через AS-FEN02	○	-	0	-	-	N	R	0
*SR1539	Кількість підключень адаптера EtherNet/IP через AS-FEN02	○	-	0	-	-	N	R	0
*SR1559	Кількість віддалених модулів, підключених до модуля ЦП (усі праві модулі не включені)	○	○	0	-	-	N	R	0
*SR1560	Кількість модулів правої сторони, підключених до модуля CPU	○	○	0	-	-	N	R	0
*SR1561	Код моделі 1 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1562	Код моделі 2 - го правого модуля, підключеного до модуля ЦП або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1563	Код моделі 3 - го правого модуля, підключеного до модуля ЦП або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1564	Код моделі 4 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1565	Код моделі 5 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1566	Код моделі 6 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1567	Код моделі 7 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1568	Код моделі 8 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1569	Код моделі 9 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1570	Код моделі 10 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1571	Код моделі 11 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1572	Код моделі 12 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1573	Код моделі 13 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1574	Код моделі 14 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1575	Код моделі 15 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1576	Код моделі 16 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1577	Код моделі 17 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1578	Код моделі 18 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1579	Код моделі 19 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1580	Код моделі 20 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1581	Код моделі 21 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1582	Код моделі 22 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1583	Код моделі 23 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1584	Код моделі 24 - го правого модуля, підключеного до модуля CPU або виносного модуля	○	○	0	-	-	N	R	0
*SR1585	Код моделі 25 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1586	Код моделі 26 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1587	Код моделі 27 - го правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0

SR	функція	Серія AS300	Серія AS100/200	OFF ↓ ON	STOP ↓ RUN	RUN ↓ STOP	Засув	Атрибут	За замовч.
*SR1588	Код моделі 28 - 1 ^{го} правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1589	Код моделі 29 - 1 ^{го} правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1590	Код моделі 30 - 1 ^{го} правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1591	Код моделі 31 - 1 ^{го} правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
*SR1592	Код моделі 32 - 1 ^{го} правого модуля, підключеного до модуля CPU або віддаленого модуля	○	○	0	-	-	N	R	0
1600 SR	Активувати відновлення параметрів приводу 0: Вимкнути (за замовч.) 1: Коли ПЛК увімкнено або переведено з режиму STOP на RUN, параметри будуть збережені. Після завершення процесу відновлення параметрів стан буде скинуто до 0.	○	○	-	-	-	Y	R/W	0
SR1601	Призначте номер цільового приводу для відновлення параметрів (SR1601.0-SR1601.15 відповідає номеру цільового диска 1-16 0: Вимкнути (за замовч.) ON: призначений цільовий привод виконує відновлення параметрів	○	○	-	-	-	Y	R/W	0
SR1602	Призначте номер цільового приводу для відновлення параметрів (Зарезервовано SR1602.0-SR1602.15 для цільового диска номер 17-32 для майбутнього використання.)	○	○	-	-	-	Y	R/W	0
SR1603	Загальна кількість накопичувачів, у яких виникли помилки під час відновлення параметрів. 0: помилки не сталось 1-16: Загальна кількість накопичувачів, які зіткнулися з помилками під час відновлення параметрів.	○	○	0	-	-	N	R	0
SR1604	Номер останнього диска, на якому виникли помилки під час відновлення параметрів. 0-255: остання цифра IP-адреси останнього накопичувача, на якому виникли помилки під час відновлення параметрів. (Працює, лише коли SM1161 увімкнено.)	○	○	0	-	-	N	R	0
SR1605	Код помилки останньої помилки накопичувача під час відновлення параметрів.	○	○	0	-	-	N	R	0
*1700 SR	Позначка для визначення частоти для компенсації люфтів на Y0.0/Y0.1 для інструкції CSFO (Гц)	○	○	0	0	-	N	R/W	0
*SR1702	Прапорець для визначення частоти для компенсації люфтів на Y0.2/Y0.3 для інструкції CSFO (Гц)	○	○	0	0	-	N	R/W	0
*SR1704	Позначка для визначення частоти для компенсації люфтів на Y0,4/Y0,5 для інструкції CSFO (Гц)	○	○	0	0	-	N	R/W	0
*SR1706	Позначка для визначення частоти для компенсації люфтів на Y0,6/Y0,7 для інструкції CSFO (Гц)	○	○	0	0	-	N	R/W	0

SR	функція	Серія AS300	Серія AS100/200	OFF ⇓ ON	STOP ⇓ RUN	RUN ⇓ STOP	Засув	Атрибут	За замовч.
*SR1708	Позначка для визначення частоти для компенсації люфтів на Y0.8/Y0.9 для інструкції CSFO (Гц)	○	○	0	0	-	N	R/W	0
*SR1710	Прапор для визначення частоти для компенсації люфтів на Y0.10/Y0.11 для інструкції CSFO (Гц)	○	○	0	0	-	N	R/W	0

Примітка 1: для SR* зверніться до таблиці SM/SR для отримання додаткової інформації

Примітка 2: комунікаційна карта в таблиці відноситься до AS-F232, AS-F422 і AS-F485.

2.2.15 Умови оновлення реєстрів спеціальних даних

Регістр спеціальних даних	Час оновити
SR0–SR2	Оновлення реєстру при виникненні помилки виконання програми.
SR4–SR6	Оновити реєстр, якщо є помилка перевірки граматики
SR8–SR9	Оновити реєстр, якщо виникає помилка сторожового таймера.
SR27	Оновити значення у вбудованому AD для AS-F2AD. Система скидає його на OFF.
SR28-SR31	Оновити реєстр, якщо вихідні дані багаторазово використовуються більш ніж однією інструкцією високошвидкісного виведення.
SR32	Оновити реєстр один раз, коли є помилка. -1: помилки не сталося.
SR36	Регістр оновлюється вами. Ви встановлюєте прапорець SM36 на ON, і система зберігає дані на карту пам'яті. Після завершення збереження система автоматично скидає його до OFF
SR38	Регістр оновлюється системою. Ви скидаєте його на OFF.
SR40–SR161	Оновити реєстр, якщо є помилка.
CP162–CP163	Після того, як виготовлений ПЛК залишає завод, кожна хвилина живлення ПЛК підраховується, а реєстр оновлюється щохвилини.
CP166–CP171	Регістр оновлюється системою.
CP172–CP175	Регістр оновлено вами.
CP176–CP179	Оновити реєстр відповідно до параметрів у HWCONFIG.
SR180	Оновлюйте реєстр, коли ПЛК увімкнено та вимкнено.
CP182–CP183	Оновити реєстр відповідно до параметрів у HWCONFIG, коли ПЛК увімкнено. Згодом ви можете змінити налаштування.
SR185	Оновлювати реєстр для часу циклу після сканування циклу кожного разу, коли активується віддалений модуль
CP190–CP197	Регістр оновлено вами.
CP198–CP199	Після циклічного сканування, коли ПЛК увімкнено.
SR201–SR213	Регістр оновлюється відповідно до налаштувань у HWCONFIG, коли ПЛК увімкнено. Згодом ви можете змінити налаштування.
SR215–SR216	Регістр оновлюється системою.
SR217–SR218	Регістр оновлюється відповідно до налаштувань у HWCONFIG під час увімкнення. Згодом ви можете змінити налаштування.
SR220–SR226	Оновлюйте реєстр кожного циклу сканування.
SR227–SR308	Оновити реєстр, коли програму буде завантажено на ПЛК.
SR309–SR390	Оновлювати реєстр, коли змінюється статус ПЛК.
SR391–SR397	Оновлюйте реєстр кожного циклу сканування.
SR403-SR404	Регістр оновлюється системою.
SR407	Оновлювати реєстр кожну секунду.
SR408–SR416	Оновлювати реєстр кожного разу, коли виконується інструкція END.
SR421–SR424	Регістр оновлено вами.
SR440–SR443	Оновити реєстр, коли ПЛК увімкнено.
SR444–SR451	Оновити реєстр, коли ПЛК увімкнено.
SR453	Оновити реєстр, якщо є помилка.
SR460	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR462–SR466	Регістр оновлено вами.
SR467	Регістр оновлюється системою.
SR468–SR469	Коли ПЛК подається живлення, реєстр оновлюється відповідно до таблиці планування позицій. Згодом ви можете змінити налаштування.

Регістр спеціальних даних	Час оновити
SR470	Регістр оновлюється системою.
SR472	Регістр вас оновлено.
SR474	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR476–SR477	Регістр оновлено вами.
SR480	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR482–SR486	Регістр оновлено вами.
SR487	Регістр оновлюється системою.
SR488–SR489	Оновіть реєстр, коли ПЛК увімкнено, відповідно до таблиці планування позицій. Згодом ви можете змінити налаштування.
SR490	Регістр оновлюється системою.
SR492	Регістр оновлено вами.
SR494	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR496–SR497	Регістр оновлено вами.
SR500	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR502–SR506	Регістр оновлено вами.
SR507	Регістр оновлюється системою.
SR508–SR509	Оновіть реєстр, коли ПЛК увімкнено, відповідно до таблиці планування позицій. Згодом ви можете змінити налаштування.
SR510	Регістр оновлюється системою.
SR512	Регістр оновлено вами.
SR514	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR516–SR517	Регістр оновлено вами.
SR520	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR522–SR526	Регістр оновлено вами.
SR527	Регістр оновлюється системою.
SR528–SR529	Оновіть реєстр, коли ПЛК увімкнено, відповідно до таблиці планування позицій. Згодом ви можете змінити налаштування.
SR530	Регістр оновлюється системою.
SR532	Регістр оновлено вами.
SR534	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете відредагувати налаштування.
SR536–SR537	Регістр оновлюється користувачем.
SR540	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR542–SR546	Регістр оновлюється користувачем.
SR547	Регістр оновлюється системою.
SR548–SR549	Оновіть реєстр, коли ПЛК подається живлення відповідно до таблиці планування позицій. Згодом ви можете змінити налаштування.
SR550	Регістр оновлюється системою.
SR552	Регістр оновлено вами.
SR554	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення, а програма виконується.

Регістр спеціальних даних	Час оновити
	відскановані. Якщо інструкція не виконується, можна редагувати налаштування.
SR556–SR557	Регістр оновлено вами.
SR560	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, можна редагувати налаштування.
SR562–SR566	Регістр оновлено вами.
SR567	Регістр оновлюється системою.
SR568–SR569	Оновіть реєстр, коли ПЛК увімкнено, відповідно до таблиці планування позицій. Згодом ви можете змінити налаштування.
SR570	Регістр оновлюється системою.
SR572	Регістр оновлено вами.
SR574	Оновлювати реєстр кожного разу, коли виконується інструкція високошвидкісного виведення та сканується програма. Якщо інструкція не виконується, ви можете редагувати налаштування.
SR576–SR579	Регістр оновлено вами.
SR580–SR603	Оновіть реєстр відповідно до параметрів у HWCONFIG, коли ПЛК увімкнено. Згодом ви можете змінити налаштування.
SR610–SR621	Оновлювати реєстр кожного разу, коли виконується інструкція виведення.
SR623–SR634	Оновлювати реєстр кожного разу, коли виконується інструкція EIX або DIX. ON: переривання включено OFF.: переривання вимкнено
SR640–SR651	Регістр оновлено вами.
SR652–SR655	Регістр оновлюється системою.
SR656–SR657	Регістр оновлюється системою. Ви скидаєте його на OFF.
SR658–SR726	Регістр оновлюється системою.
SR731–SR738	Регістр оновлюється системою.
SR741–SR748	Регістр оновлюється системою.
SR751–SR768	Регістр оновлено вами.
SR771–SR778	Регістр оновлюється системою.
SR781–SR788	Регістр оновлюється системою.
SR791–SR806	Регістр оновлюється системою.
SR811–SR818	Регістр оновлено вами.
SR820	Оновіть реєстр відповідно до налаштувань CANopen Builder.
SR821	Оновіть реєстр, коли оновиться прошивка.
SR822	Оновіть реєстр відповідно до параметрів у HWCONFIG.
SR823–SR893	Регістр оновлюється системою.
SR900–SR901	Регістр оновлюється системою.
SR902	Регістр оновлено вами.
SR1000–SR1006	1. Регістр оновлено вами. 2. Регістр оновлюється після ввімкнення живлення або після завантаження HWCONFIG.
SR1007	Регістр оновлюється системою.
SR1009	Регістр оновлюється системою.
SR1010	Регістр оновлено вами.
SR1011–SR1014	Регістр оновлюється системою.
SR1015	Регістр оновлено вами.
SR1016	Регістр оновлено вами.
SR1017	Регістр оновлено вами.
SR1018	Оновлювати реєстр кожен цикл сканування системою.

Регістр спеціальних даних	Час оновити
SR1019	1. Оновлювати реєстр кожен цикл сканування системою. 2. Регістр оновлено вами.
SR1020–SR1107	1. Оновіть реєстр, коли з'єднання буде встановлено. 2. Оновлюйте реєстр кожного циклу сканування.
SR1092–SR1093	Регістр оновлено вами.
SR1110	Регістр оновлено вами. Система скидає його на OFF.
CP1111–CP1114	Регістр оновлено вами.
CP1116–CP1117	Оновіть реєстр, коли програму буде завантажено на ПЛК.
SR1120–SR1183	Оновіть реєстр після завершення зв'язку.
SR1191–SR1198	Регістр оновлюється системою.
SR1201–SR1216	Регістр оновлюється системою.
SR1221–SR1236	Регістр оновлюється системою.
SR1241–SR1256	Регістр оновлюється системою.
SR1261–SR1268	Регістр оновлено вами.
SR1291–SR1296	Регістр оновлено вами. Система скидає його на OFF.
SR1312–SR1315	Оновлювати реєстр під час спілкування
SR1318–SR1320	Оновіть реєстр, коли параметр завантажувється в ПЛК або коли ПЛК подається живлення.
SR1335–SR1336	Оновлювати реєстр кожен цикл сканування після ввімкнення функції обміну даними.
SR1340–SR1371	Оновіть реєстр, якщо є помилка.
SR1375–SR1376	Оновлювати реєстр кожен цикл сканування після ввімкнення функції обміну даними.
SR1380–SR1411	Оновіть реєстр, якщо є помилка.
SR1430–SR1431	Регістр оновлюється системою.
SR1435–SR1436	Оновлювати реєстр під час спілкування
SR1440–SR1471	Оновити реєстр, якщо під час зв'язку виникає помилка
SR1475–SR1476	Оновлювати реєстр під час спілкування
SR1480–SR1511	Оновити реєстр, якщо під час зв'язку виникає помилка
SR1520–SR1535	Оновіть реєстр після завершення зв'язку через AS-FEN02.
SR1536–SR1539	Регістр оновлюється системою.
SR1559–SR1592	Регістр оновлюється системою, коли система подається живлення, від'єднується від віддаленого модуля або відновлюється з'єднання віддаленого модуля.
SR1600–SR1602	Регістр оновлено вами.
SR1603–SR1605	Виникла помилка на накопичувачі під час відновлення параметрів; Регістр оновлюється системою.
SR1700, SR1702, SR1704, SR1706, SR1708, SR1710	Регістр оновлено вами.

2.2.16 Додаткові зауваження щодо спеціальних допоміжних реле та спеціальних регістрів даних

1. Таймер очікування сканування

● SM8/SR8

Коли час очікування сканування відбувається під час виконання програми, світлодіодний індикатор помилки на ПЛК починає світитися безперервно, а SM8 світиться.

SR8 містить адресу кроку, на якому сторожовий таймер змінюється на ON.

2. Очищення сигнальної лампи

● SM22

Встановлення SM22 на ON очищає журнал помилок і попереджувальну лампу.

3. Модуль ЦП і функціональна карта AD/DA

● Аналоговий вхід

AS-F2AD (застосовується до ПЛК серії AS300)		
Пункт	Вхід напруги	Поточний вхід
Вхід аналогового сигналу	Постійний струм 0–10 В	Постійний струм 4–20 мА
роздільна здатність	12-бітний	11-бітний
Діапазон цифрового перетворення	0 - 4000	0 - 2000
Номер SR	SR168 (Канал 1 карти AS-F2AD 1) SR169 (Канал 2 карти AS-F2AD 1) SR170 (Канал 1 карти AS-F2AD 2) SR171 (Канал 2 карти AS-F2AD 2)	

AS-F2DA (застосовується до ПЛК серії AS300)		
Пункт	Вихідна напруга	Поточний вихід
Вихід аналогового сигналу	Постійний струм 0–10 В	Постійний струм 4–20 мА
роздільна здатність	12-бітний	11-бітний
Діапазон цифрового перетворення	0 - 4000	0 - 2000
Номер SR	SR172 (Канал 1 карти AS-F2AD 1) SR173 (Канал 2 карти AS-F2AD 1) SR174 (Канал 1 карти AS-F2AD 2) SR175 (Канал 2 карти AS-F2AD 2)	

Вбудована функція AD AS218			
Пункт	Вхід напруги	Поточний вхід	
Вихід аналогового сигналу	Постійний струм -10 В ~ +10 В	Постійний струм - 20 мА ~ 20 мА	Постійний струм 4 мА ~ 20 мА* 1
роздільна здатність	12-бітний	11-бітний	11-бітний
Діапазон цифрового перетворення	-2000 ~ 2000	-1000 ~ 1000	0 ~ 1000
Номер SR	SR168 (CH1) SR169 (CH2)		

*1: Це доступно лише для AS218 із FW V1.06.20 та ISPSOFT V3.08 або новішої версії.

● Код помилок аналогового входу в SM27/SR27

Коли значення перетворення на аналоговому вході перевищує допустимий діапазон або виникає будь-яка втрата з'єднання, прапор попередження SM27 буде ON, а код помилки аналогового входу буде збережено в SR27. Якщо для параметра Input Warning у HWCONFIG встановлено значення Enable, процесори серії AS300 і AS200 із програмним забезпеченням версії 1.10.00 або новішої підтримують цю функцію. Дивіться наступні зображення налаштувань для довідки.

Серія AS300: AS-F2AD встановлено у функціональній карті 1/2

Hardware Configuration						
General		Data Exchange				
- AS332T-A		Function Card 1 Setting				
+ System settings		Name	Setting Value	Unit	Default	Minimum Maximum
COM1 Port Setting		Delay time to sending	0	ms	0	0 3000
COM2 Port Setting		Received Data Timeout	200	ms	200	0 3000
Ethernet Port Basic Setting		F2AD Analog Input mode	0~10V		0~10V	- -
+ Ethernet Port Advanced Setting		F2DA Analog Output mode	0~10V		0~10V	- -
Function Card 1 Setting		F2AD Sampling Time	3	ms	3	3 15
+ Function Card 2 Setting		F2AD Average Times	10		10	1 15
+ Delta Device Parameter Restore S...		F2AD Analog Input Warning	Enable		Disable	- -

Серія AS200: вбудоване налаштування AD/DA

Hardware Configuration						
General		Data Exchange				
- AS218TX-A		Built-in AD/DA Setting				
+ System settings		Name	Setting Value	Unit	Default	Minimum Maximum
COM1 Port Setting		AD channel 1 input mode	Disable		Disable	- -
COM2 Port Setting		AD channel 2 input mode	Disable		Disable	- -
Ethernet Port Basic Setting		DA channel 1 output mode	Voltage(-10 ~ +1...		Voltage(-1...	- -
+ Ethernet Port Advanced Setting		DA channel 2 output mode	Voltage(-10 ~ +1...		Voltage(-1...	- -
Built-in AD/DA Setting		AD Sampling Time	3	ms	3	3 15
Built-in CAN communication		AD Average Times	10		10	1 15
+ Delta Device Parameter Restor...		AD Analog Input Warning	Enable		Disable	- -

Код помилки	опис	Рішення
16#0040	Значення, яке потрібно перетворити з вбудованого вхідного каналу AD 1 або AS-F2AD (функціональна карта 1), перевищує допустимий діапазон.	Перевірте фактичний вхідний сигнал
16#0041	Значення, яке потрібно перетворити з вбудованого вхідного каналу AD 2 або AS-F2AD (функціональна карта 1), перевищує допустимий діапазон.	Перевірте фактичний вхідний сигнал
16#0042	Значення, яке потрібно перетворити з вхідного каналу 1 AS-F2AD (функціональна карта 2), перевищує допустимий діапазон.	Перевірте фактичний вхідний сигнал
16#0043	Значення, яке потрібно перетворити з вхідного каналу 2 AS-F2AD (функціональна карта 2), перевищує допустимий діапазон.	Перевірте фактичний вхідний сигнал
16#0044	Підключення до поточного входу в режимі 4 мА ~ 20 мА від вбудованого вхідного каналу AD 1 або AS-F2AD (функціональна карта 1) втрачається.	Перевірте підключення
16#0045	Підключення до поточного входу в режимі 4 мА ~ 20 мА від вбудованого вхідного каналу AD 2 або AS-F2AD (функціональна карта 1) втрачається.	Перевірте підключення
16#0046	З'єднання з поточним входом у режимі 4мА~20мА від вхідного каналу 1 AS-F2AD (функціональна карта 2) втрачається.	Перевірте підключення
16#0047	З'єднання з поточним входом у режимі 4мА~20мА від вхідного каналу 2 AS-F2AD (функціональна карта 2) втрачається.	Перевірте підключення

Функціональна карта: AS-F2AD

Режим аналогового входу	Відповідний діапазон значень	Зауваження
Напруга: 0В ~ 10В	0 ~ 4000	1. Значення > 4047 вважається поза діапазоном 2. Немає виявлення втрати з'єднання
Струм 4мА ~ 20мА	0 ~ 2000	1. Значення < -24 або > 2023 вважається поза діапазоном 2. Значення < -36 розглядається як втрачене з'єднання.

Вбудований аналоговий вхід AS218:

Режим аналогового входу	Відповідний діапазон значень	Зауваження
Напруга: -10В ~ 10В	-2000 ~ 2000	1. Значення < -2024 або > 2023 вважається поза діапазоном 2. Немає виявлення втрати з'єднання
Струм: -20мА ~ 20мА	-1000 ~ 1000	1. Значення < -1012 або > 1011 вважається поза діапазоном 2. Немає виявлення втрати з'єднання
Струм: 4 мА ~ 20 мА	0 ~ 1000	1. Значення < -12 або > 1011 вважається поза діапазоном 2. Значення < -36 розглядається як втрачене з'єднання.

● Аналоговий вихід

AS-F2DA (застосовується до ПЛК серії AS300)		
Пункт	Вхід напруги	Поточний вхід
Вхід аналогового сигналу	Постійний струм 0–10 В	Постійний струм 4–20 мА
роздільна здатність	12-бітний	11-бітний
Діапазон цифрового перетворення	0 - 4000	0 - 4000
Номер SR	SR172 (Канал 1 карти AS-F2DA 1) SR173 (Канал 2 карти AS-F2DA 1) SR174 (Канал 1 карти AS-F2DA 2) SR175 (Канал 2 карти AS-F2DA 2)	

AS218 вбудована функція DA		
Пункт	Вихідна напруга	Поточний вихід
Вихід аналогового сигналу	Постійний струм -10 В ~ +10 В	Постійний струм -0 мА ~ 20 мА
роздільна здатність	12-бітний	12-бітний
Діапазон цифрового перетворення	-2000 ~ 2000	0 ~ 4000
Номер SR	SR172 (CH1) SR173 (CH2)	

4. Годинник реального часу

● SM220, SR220–SR226 і SR391–SR397

SM220: відкалібруйте годинник реального часу в межах ± 30 секунд

Коли SM220 змінюється з OFF. на UBIMK., система калібрує годинник реального часу.

Якщо друге значення годинника реального часу знаходиться в діапазоні 0–29, значення хвилин фіксується, а значення секунд обнулюється.

Якщо значення секунди на годиннику реального часу знаходиться в діапазоні 30–59, значення хвилин збільшується на одиницю, а значення секунд обнулюється.

У цій таблиці наведено відповідні функції та значення SR220–SR226 і SR391–SR397.

пристрій		функція	Значення
Двійкова десяткова система числення	Десяткова система		
SR220	SR391	рік	00–99 (н.е.)
SR221	SR392	місяць	1–12
SR222	SR393	День	1–31
SR223	SR394	годину	0–23
SR224	SR395	хвилина	0–59
SR225	SR396	секунда	0–59
SR226	SR397	тиждень	1–7

SR391–SR397 відповідають SR220–SR226. Різниця між SR220–SR226 і SR391–SR397 полягає в тому, що перший використовує двійкову десяткову систему, тоді як другий використовує десяткову систему. Наприклад, грудень представлений як 12 у SR392, тоді як він представлений як 1100 у двійковому десятковому коді. Додаткову інформацію про годинник реального часу див. у Розділі 6.

5. Функції спілкування

- SM96–SM107, SM209–SM212, SR201–SR202 і SR209–SR216

SR215 і SR216 записують коди функціональних карт. У наступній таблиці наведено функції, представлені кодами інтерфейсу.

Код	0	1	2	3	4	5	6
функція	Без картки	RS232	RS422	RS485	F2AD	F2DA	FCOPM
Код	7	8	9	10	11	7	8
функція	FEN02	FPFN02	FOPC02	FFTP01	FECAT	FEN02	FPFN02

Функціональні карти, зокрема FCOPM, FEN02, FPFN02, FOPC02, FFTP01, FECAT, підтримуються лише як функціональна карта 2.

Якщо протоколом порту зв'язку для ПЛК є RS485, RS232 або RS422, тоді SR209 записує формат зв'язку порту COM1 на ПЛК, а SR212 записує формат зв'язку порту COM2 на ПЛК. У наведеній нижче таблиці наведено налаштування для протоколів зв'язку. Зверніться до розділу 6 для отримання додаткової інформації щодо інструкцій щодо зв'язку.

b0	Довжина даних			7 (значення=0)		8 (значення=1)	
b1 b2	Біти парності			00	:	Жодного	
				01	:	Непарні біти парності	
				10	:	Парні біти парності	
b3	Стоп-біти			1 біт (значення=0)		2 біти (значення=1)	
b4 b5 b6 b7	0001	(H 1)	:	4800			
	0010	(H 2)	:	9600			
	0011	(H 3)	:	19200			
	0100	(H 4)	:	38400			
	0101	(H 5)	:	57600			
	0110	(H 6)	:	115200			

	0111	(H 7)	:	230400	
	1000	(H 8)	:	500000	
	1001	(H 9)	:	921000	
	1010	(16#A)	:	Невизначено	
	1011	(16#B)	:	Невизначено	
	1100	(16#C)	:	Невизначено	
	1101	(16#D)	:	Невизначено	
	1110	(16#E)	:	Невизначено	
	1111	(16#F)	:	Визначається користувачем *1	
b8-b15	Не визначено (зарезервовано)				

1: Зверніться до Параметри HWCONFIG в ISPSOft для визначеної користувачем швидкості
 2: Зверніться до передачі даних. іон 6.19.3 для використання комунікаційних прапорів і
 розділу реєстрів.

навчання де **вміст заст**

SM203/SM2 04/SM205

пристрій номер	Пристрій, який очищено
SM203/ SM204 Усі незамкнені області є очищено.	Зони без фіксації у вхідних реле, вихідних реле, крокових реле та допоміжних реле очищаються. Незафіксовані області в таймерах, лічильниках і 32-розрядних лічильниках очищаються. Незафіксовані області в реєстрах даних та індексних реєстрах очищаються. Сторожовий таймер не працює протягом цього періоду часу.
SM205 Усі замкнені області очищено.	Заблоковані області в таймерах, лічильниках і 32-розрядних лічильниках очищаються. Зафіксовані допоміжні реле очищені. Зафіксовані реєстри даних очищаються. Сторожовий таймер не працює протягом цього періоду часу.

Зверніться до Розділу 2.1.4 для отримання додаткової інформації про зони блокування в діапазоні пристрою.

7. Журнал помилок ПЛК

● SR40-SR161

SR40: Максимальна кількість журналів помилок, які зберігаються в SR40, становить 20. Якщо кількість журналів помилок перевищує максимальне число 20, буде збережено лише 20 з них. Кожен журнал помилок займає 6 реєстрів.

SR41: показник журналу помилок вказує на останній журнал помилок. Коли виникає помилка, значення показника журналу помилок збільшується на одиницю. Діапазон значень показника 0–19. Наприклад, показник журналу помилок вказує на четвертий журнал помилок, коли значення в SR41 дорівнює 3. Якщо значення в SR41 дорівнює 20, це означає, що кількість журналів помилок перевищує 20 і зараз у циклі.

SR42~SR161: Час виникнення помилки та місце, де сталася помилка, записуються в SR42–SR161. У стовпці Isot записується ЦП (0) і номер віддаленого модуля (1~15). У наступній таблиці перераховані відповідні функції цих реєстрів даних.

Номер	Слот	ID модуля	Код помилки	Час виникнення помилки					
				рік	місяць	День	годину	хвилина	секунда
1	SR42 Молодший байт	SR43	SR44	SR45 Старший байт	SR45 Низький байт	SR46 Старший байт	SR46 Низький байт	SR47 Старший байт	SR47 Низький байт

Номер	Слот	ID модуля	Код помилки	Час виникнення помилки					
				рік	місяць	День	годину	хвилина	секунда
2	SR48 Молодший байт	SR49	SR50	SR51 Старший байт	SR51 Низький байт	SR52 Старший байт	SR52 Низький байт	SR53 Старший байт	SR53 Низький байт
3	SR54 Молодший байт	SR55	SR56	SR57 Старший байт	SR57 Низький байт	SR58 Старший байт	SR58 Низький байт	SR59 Старший байт	SR59 Низький байт
4	SR60 Молодший байт	SR61	SR62	SR63 Старший байт	SR63 Низький байт	SR64 Старший байт	SR64 Низький байт	SR65 Старший байт	SR65 Низький байт
5	SR66 Молодший байт	SR67	SR68	SR69 Старший байт	SR69 Низький байт	SR70 Старший байт	SR70 Низький байт	SR71 Старший байт	SR71 Низький байт
6	SR72 Молодший байт	SR73	SR74	SR75 Старший байт	SR75 Низький байт	SR76 Старший байт	SR76 Низький байт	SR77 Старший байт	SR77 Низький байт
7	SR78 Молодший байт	SR79	SR80	SR81 Старший байт	SR81 Низький байт	SR82 Старший байт	SR82 Низький байт	SR83 Старший байт	SR83 Низький байт
8	SR84 Молодший байт	SR85	SR86	SR87 Старший байт	SR87 Низький байт	SR88 Старший байт	SR88 Низький байт	SR89 Старший байт	SR89 Низький байт
9	SR90 Молодший байт	SR91	SR92	SR93 Старший байт	SR93 Низький байт	SR94 Старший байт	SR94 Низький байт	SR95 Старший байт	SR95 Низький байт
10	SR96 Молодший байт	SR97	SR98	SR99 Старший байт	SR99 Низький байт	SR100 Старший байт	SR100 Низький байт	SR101 Старший байт	SR101 Низький байт
11	SR102 Молодший байт	SR103	SR104	SR105 Старший байт	SR105 Молодший байт	SR106 Старший байт	SR106 Низький байт	SR107 Старший байт	SR107 Низький байт
12	SR108 Молодший байт	SR109	SR110	SR111 Старший байт	SR111 Низький байт	SR112 Старший байт	SR112 Низький байт	SR113 Старший байт	SR113 Низький байт
13	SR114 Молодший байт	SR115	SR116	SR117 Старший байт	SR117 Молодший байт	SR118 Старший байт	SR118 Низький байт	SR119 Старший байт	SR119 Низький байт
14	SR120 Молодший байт	SR121	SR122	SR123 Старший байт	SR123 Молодший байт	SR124 Старший байт	SR124 Низький байт	SR125 Старший байт	SR125 Низький байт
15	SR126 Молодший байт	SR127	SR128	SR129 Старший байт	SR129 Молодший байт	SR130 Старший байт	SR130 Низький байт	SR131 Старший байт	SR131 Низький байт
16	SR132 Молодший байт	SR133	SR134	SR135 Старший байт	SR135 Молодший байт	SR136 Старший байт	SR136 Низький байт	SR137 Старший байт	SR137 Низький байт
17	SR138 Молодший байт	SR139	SR140	SR141 Старший байт	SR141 Молодший байт	SR142 Старший байт	SR142 Низький байт	SR143 Старший байт	SR143 Низький байт
18	SR144 Молодший байт	SR145	SR146	SR147 Старший байт	SR147 Молодший байт	SR148 Старший байт	SR148 Низький байт	SR149 Старший байт	SR149 Низький байт
19	SR150 Молодший байт	SR151	SR152	SR153 Старший байт	SR153 Молодший байт	SR154 Старший байт	SR154 Низький байт	SR155 Старший байт	SR155 Низький байт
20	SR156 Молодший байт	SR157	SR158	SR159 Старший байт	SR159 Молодший байт	SR160 Старший байт	SR160 Низький байт	SR161 Старший байт	SR161 Низький байт

8. Журнал завантажень ПЛК

● SR227-SR308

SR227: Максимальна кількість журналів завантажень, які зберігаються в SR227, становить 20. Кожен журнал завантажень займає 4 регістри. Дії завантаження, які записуються, пронумеровані, як показано в наступній таблиці.

Число (шістнадцяткове)	Дія завантаження
0x0001	Завантаження програми
0x0002	Завантаження налаштувань ПЛК
0x0003	Завантаження таблиці модулів введення/виведення
0x000A	Операційні коди ПЛК (ISP)
0x000B	Параметри ПЛК (HWCONFIG)
0x000C	Конфігурації модуля (HWCONFIG)
0x000D	Операційні коди та програми ПЛК (ISP)
0x000E	Операційні коди ПЛК (онлайн редагування)
0x000F	Операційні коди та програми ПЛК (онлайн редагування)
0x0010	Операційні коди та програми ПЛК (відновлення з SD карти)
0x0011	Конфігурації модуля та параметри ПЛК (HWCONFIG)
0x0012	Конфігурації модулів і таблиця обміну даними (HWCONFIG)
0x0013	Конфігурації модулів і таблиця обміну даними (HWCONFIG)
0x0014	Конфігурації модуля (EIP Builder)
0x0015	Конфігурації модуля, параметри ПЛК і таблиця обміну даними (відновлення з SD-карти)

Примітка: ISPSooft V3.13 або пізнішої версії та ЦП із FW V1.10 або пізнішої версії підтримують 0x000A – 0x0015.

SR228: показник журналу завантажень вказує на останній журнал завантажень. Коли виконується дія завантаження, значення вказівника журналу завантажень збільшується на одиницю. Діапазон значень показника 0–19. Наприклад, показник журналу завантажень вказує на четвертий журнал завантажень, коли значення в SR228 дорівнює 3.

Час завантаження та номери дій записуються в SR229–SR30. У наступній таблиці перераховані відповідні функції цих регістрів даних.

Немає.	Номер дії	* Час, коли відбувається завантаження					
		рік	місяць	День	годину	хвилина	секунда
1	SR229	SR230 Старший байт	SR230 Молодший байт	SR231 Старший байт	SR231 Молодший байт	SR232 Старший байт	SR232 Молодший байт
2	SR233	SR234 Старший байт	SR234 Молодший байт	SR235 Старший байт	SR235 Молодший байт	SR236 Старший байт	SR236 Молодший байт
3	SR237	SR238 Старший байт	SR238 Молодший байт	SR239 Старший байт	SR239 Молодший байт	SR240 Старший байт	SR240 Молодший байт
4	SR241	SR242 Старший байт	SR242 Молодший байт	SR243 Старший байт	SR243 Молодший байт	SR244 Старший байт	SR244 Молодший байт
5	SR245	SR246 Старший байт	SR246 Молодший байт	SR247 Старший байт	SR247 Молодший байт	SR248 Старший байт	SR248 Молодший байт
6	SR249	SR250 Старший байт	SR250 Молодший байт	SR251 Старший байт	SR251 Молодший байт	SR252 Старший байт	SR252 Молодший байт

Немас.	Номер дії	* Час, коли відбувається завантаження					
		рік	місяць	День	годину	хвилина	секунда
7	SR253	SR254 Старший байт	SR254 Молодший байт	SR255 Старший байт	SR255 Молодший байт	SR256 Старший байт	SR256 Молодший байт
8	SR257	SR258 Старший байт	SR258 Молодший байт	SR259 Старший байт	SR259 Молодший байт	SR260 Старший байт	SR260 Молодший байт
9	SR261	SR262 Старший байт	SR262 Молодший байт	SR263 Старший байт	SR263 Молодший байт	SR264 Старший байт	SR264 Молодший байт
10	SR265	SR266 Старший байт	SR266 Молодший байт	SR267 Старший байт	SR267 Молодший байт	SR268 Старший байт	SR268 Молодший байт
11	SR269	SR270 Старший байт	SR270 Молодший байт	SR271 Старший байт	SR271 Молодший байт	SR272 Старший байт	SR272 Молодший байт
12	SR273	SR274 Старший байт	SR274 Молодший байт	SR275 Старший байт	SR275 Молодший байт	SR276 Старший байт	SR276 Молодший байт
13	SR277	SR278 Старший байт	SR278 Молодший байт	SR279 Старший байт	SR279 Молодший байт	SR280 Старший байт	SR280 Молодший байт
14	SR281	SR282 Старший байт	SR282 Молодший байт	SR283 Старший байт	SR283 Молодший байт	SR284 Старший байт	SR284 Молодший байт
15	SR285	SR286 Старший байт	SR286 Молодший байт	SR287 Старший байт	SR287 Молодший байт	SR288 Старший байт	SR288 Молодший байт
16	SR289	SR290 Старший байт	SR290 Молодший байт	SR291 Старший байт	SR291 Молодший байт	SR292 Старший байт	SR292 Молодший байт
17	SR293	SR294 Старший байт	SR294 Молодший байт	SR295 Старший байт	SR295 Молодший байт	SR296 Старший байт	SR296 Молодший байт
18	SR297	SR298 Старший байт	SR298 Молодший байт	SR299 Старший байт	SR299 Молодший байт	SR300 Старший байт	SR300 Молодший байт
19	SR301	SR302 Старший байт	SR302 Молодший байт	SR303 Старший байт	SR303 Молодший байт	SR304 Старший байт	SR304 Молодший байт
20	SR305	SR306 Старший байт	SR306 Молодший байт	SR307 Старший байт	SR307 Молодший байт	SR308 Старший байт	SR308 Молодший байт

* Формат часу дії завантаження: дані зберігаються як десяткове число з двійковим кодом. У таблиці нижче наведено діапазон значень.

функція	Значення
рік	00–99 (н.е.)
місяць	01–12
День	01–31
годину	00–23
хвилина	00–59
секунда	00–59

9. Журнал змін стану ПЛК

● SR309–SR390

SR309: максимальна кількість журналів змін стану ПЛК, які зберігаються в SR309, становить 20. Кожен журнал змін стану ПЛК займає 4 регістри. Записані дії зміни статусу ПЛК пронумеровані, як показано в наступній таблиці.

Зміна статусу ПЛК	Номер
ПЛК увімкнено.	1
ПЛК відключено.	2
ПЛК починає працювати.	3
ПЛК зупиняється.	4
Налаштування за замовч. у ПЛК	5
Аварійна зупинка	7
Напишіть RTC	8

SR310: Показчик журналу змін стану ПЛК вказує на останній журнал змін стану ПЛК. Коли статус ПЛК змінюється, значення вказівника журналу змін статусу ПЛК збільшується на одиницю. Діапазон значень показчика 0–19. Наприклад, показчик журналу змін статусу ПЛК вказує на четвертий журнал змін статусу ПЛК, коли значення в SR310 дорівнює 3.

Час, коли відбуваються дії щодо зміни статусу ПЛК, записується в SR311–SR390. У поточній таблиці перераховані відповідні функції цих регістрів даних.

Номер	Номер дії	*Час зміни статусу ПЛК					
		рік	місяць	День	годину	хвилина	секунда
1	SR311	SR312 Старший байт	SR312 Молодший байт	SR313 Старший байт	SR313 Молодший байт	SR314 Старший байт	SR314 Молодший байт
2	SR315	SR316 Старший байт	SR316 Молодший байт	SR317 Старший байт	SR317 Молодший байт	SR318 Старший байт	SR318 Молодший байт
3	SR319	SR320 Старший байт	SR320 Молодший байт	SR321 Старший байт	SR321 Молодший байт	SR322 Старший байт	SR322 Молодший байт
4	SR323	SR324 Старший байт	SR324 Молодший байт	SR325 Старший байт	SR325 Молодший байт	SR326 Старший байт	SR326 Молодший байт
5	SR327	SR328 Старший байт	SR328 Молодший байт	SR329 Старший байт	SR329 Молодший байт	SR330 Старший байт	SR330 Молодший байт
6	SR331	SR332 Старший байт	SR332 Молодший байт	SR333 Старший байт	SR333 Молодший байт	SR334 Старший байт	SR334 Молодший байт
7	SR335	SR336 Старший байт	SR336 Молодший байт	SR337 Старший байт	SR337 Молодший байт	SR338 Старший байт	SR338 Молодший байт
8	SR339	SR340 Старший байт	SR340 Молодший байт	SR341 Старший байт	SR341 Молодший байт	SR342 Старший байт	SR342 Молодший байт
9	SR343	SR344 Старший байт	SR344 Молодший байт	SR345 Старший байт	SR345 Молодший байт	SR346 Старший байт	SR346 Молодший байт
10	SR347	SR348 Старший байт	SR348 Молодший байт	SR349 Старший байт	SR349 Молодший байт	SR350 Старший байт	SR350 Молодший байт

Номер	Номер дії	*Час зміни статусу ПЛК					
		рік	місяць	День	годину	хвилина	секунда
11	SR351	SR352 Старший байт	SR352 Молодший байт	SR353 Старший байт	SR353 Молодший байт	SR354 Старший байт	SR354 Молодший байт
12	SR355	SR356 Старший байт	SR356 Молодший байт	SR357 Старший байт	SR357 Молодший байт	SR358 Старший байт	SR358 Молодший байт
13	SR359	SR360 Старший байт	SR360 Молодший байт	SR361 Старший байт	SR361 Молодший байт	SR362 Старший байт	SR362 Молодший байт
14	SR363	SR364 Старший байт	SR364 Молодший байт	SR365 Старший байт	SR365 Молодший байт	SR366 Старший байт	SR366 Молодший байт
15	SR367	SR368 Старший байт	SR368 Молодший байт	SR369 Старший байт	SR369 Молодший байт	SR370 Старший байт	SR370 Молодший байт
16	SR371	SR372 Старший байт	SR372 Молодший байт	SR373 Старший байт	SR373 Молодший байт	SR374 Старший байт	SR374 Молодший байт
17	SR375	SR376 Старший байт	SR376 Молодший байт	SR377 Старший байт	SR377 Молодший байт	SR378 Старший байт	SR378 Молодший байт
18	SR379	SR380 Старший байт	SR380 Молодший байт	SR381 Старший байт	SR381 Молодший байт	SR382 Старший байт	SR382 Молодший байт
19	SR383	SR384 Старший байт	SR384 Молодший байт	SR385 Старший байт	SR385 Молодший байт	SR386 Старший байт	SR386 Молодший байт
20	SR387	SR388 Старший байт	SR388 Молодший байт	SR389 Старший байт	SR389 Молодший байт	SR390 Старший байт	SR390 Молодший байт

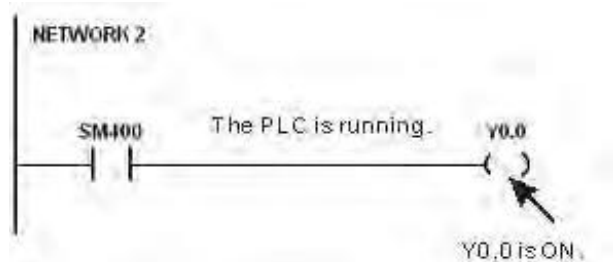
*Формат часу зміни статусу ПЛК: дані зберігаються як десяткове двійкове кодування. У таблиці нижче наведено діапазон значень.

функція	Значення
рік	00–99 (н.е.)
місяць	01–12
День	01–31
годину	00–23
хвилина	00–59
секунда	00–59

10. Прапор роботи ПЛК

● SM400-SM403

SM400: прапорець завжди ввімкнено, коли ЦП працює.

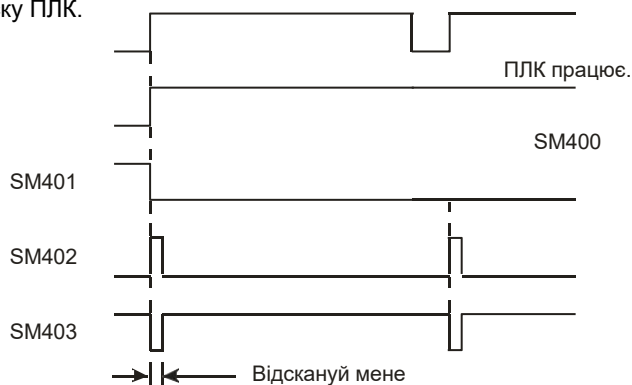


SM401: прапорець завжди OFF, коли ЦП працює.

SM402: прапорець увімкнено лише під час першого сканування. Ширина імпульсу дорівнює одному часу сканування. Ви можете використовувати цей контакт для

налаштування початкового значення.

SM403: прапорець OFF лише під час першого сканування. Тобто негативний імпульс генерується в момент запуску ПЛК.



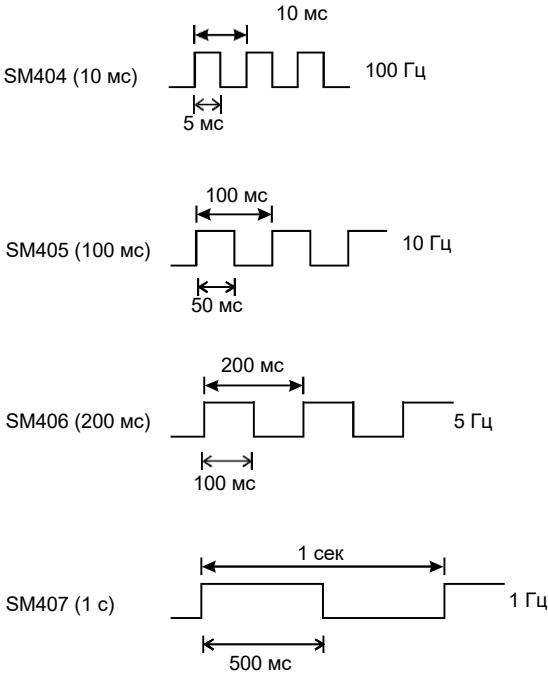
11. Початковий тактовий імпульс

- SM404 · SM405 · SM406 · SM407

ПЛК забезпечує сім типів тактових імпульсів. Коли ПЛК увімкнено, сім типів тактових імпульсів діють автоматично.

пристрій	функція
SM404	Тактовий імпульс тривалістю 10 мілісекунд, під час якого імпульс увімкнено протягом 5 мілісекунд, а потім вимкнено на 5 мілісекунд
SM405	Тактовий імпульс тривалістю 100 мілісекунд, під час якого імпульс увімкнено протягом 50 мілісекунд, а потім вимкнено на 50 мілісекунд
SM406	Тактовий імпульс тривалістю 200 мілісекунд, під час якого імпульс увімкнено на 100 мілісекунд, а потім вимкнено на 100 мілісекунд
SM407	Односекундний тактовий імпульс, під час якого імпульс увімкнено протягом 500 мілісекунд, а потім вимкнено на 500 мілісекунд

Тактові імпульси проілюстровано на наступних графіках.



12. Прапорці, пов'язані з картою пам'яті

- SM36, SM450-SM453, SM456-SR36, SR453, SR902

Ви використовуєте карту пам'яті для резервного копіювання даних у ПЛК. Зверніться до розділу 6, щоб отримати інструкції щодо карти пам'яті.

пристрій	функція
SM36	Увімкніть збереження даних на карту пам'яті. Коли ON, ПЛК працює відповідно до значення в SR36.
SM450	Карта пам'яті є. ON: карта пам'яті є. OFF: карта пам'яті відсутня.
SM451	Карта пам'яті захищена від запису.
SM452	Виконується доступ до даних на карті пам'яті. ON: виконується доступ до даних на карті пам'яті. OFF.: доступ до даних на карті пам'яті недоступний.
SM453	Помилка під час роботи карти пам'яті. ON: сталася помилка. OFF.: немає помилок.
SM456	Виконання реєстратора даних і карти пам'яті. ON: виконання за значеннями в SR902.
SR453	У разі виникнення помилки на карті пам'яті тут зберігаються коди помилок. A. 16#005D: карта пам'яті не встановлена B. 16#005E: виникає помилка під час ініціалізації карти пам'яті C. 16#005F: шлях неправильний або файл не існує D. 16#0060: неможливо створити папку за замовч. E. 16#0061: недостатньо місця в пам'яті F. 16#0062: карта пам'яті захищена від запису G. 16#0063: виникає помилка, коли дані записуються у файл

пристрій	функція
	H. 16#0064: неможливо прочитати дані з карти пам'яті I. 16#0065: файл призначений лише для читання J. 16#0066: під час відновлення карти пам'яті ПЛК знаходиться в стані RUN K. 16#0067: під час відновлення карти пам'яті ідентифікатор ПЛК не збігається під час відновлення карти пам'яті L. 16#0068: при відновленні карти пам'яті пароль не збігається M. 16#0069: під час відновлення карти пам'яті шлях до файлу та ім'я файлу містять ім'я модуля ЦП, але воно не збігається з фактичним модулем ЦП
SR902	Код для виконання реєстратора та карти пам'яті (працює з SM456); наприклад, H5AA5: запишіть дані вибірки з реєстратора даних на карту пам'яті.
SR36	Використовуйте наступні коди функцій для резервного копіювання або відновлення даних. Увімкніть SM36, щоб коди функцій працювали. A. K1122 (резервне копіювання) ^{№1}, вказує на резервне копіювання програм ПЛК на карту пам'яті. (Ідентифікатор ПЛК і пароль включені) Виконайте це резервне копіювання, коли ПЛК перебуває в стані RUN/Stop. Назва файлу резервної копії за замовч.: SDCard\PLC CARD\AS300\SysDup\user_program.dup B. K2211 (відновлення) ^{№2}, вказує на відновлення програм ПЛК із карти пам'яті. (Ідентифікатор ПЛК і пароль включені) Виконайте відновлення, коли ПЛК перебуває в стані зупинки. Назва файлу відновлення за замовч.: SDCard\PLC CARD\AS300\SysDup\user_program.dup C. K3344 (резервне копіювання) ^{№1}, це вказує на резервне копіювання зафіксованих значень (D, S, M, C, HC) і зафіксованого діапазону на карту пам'яті. Виконайте це резервне копіювання, коли ПЛК перебуває в стані RUN/Stop. Назва файлу резервної копії за замовч.: SDCard\PLC CARD\AS300\SysDup\retained_data.dup D. K4433 (відновлення) ^{№2}, вказує на відновлення зафіксованих значень (D, S, M, C, HC) і зафіксованого діапазону з карти пам'яті. Виконуйте відновлення, коли ПЛК перебуває в стані зупинки. Назва файлу відновлення за замовч.: SDCard\PLC CARD\AS300\SysDup\retained_data.dup E. K5566 (резервне копіювання) ^{№1}, вказує на резервне копіювання програм і параметрів ПЛК на карту пам'яті. (статус і значення зберігаючих пристроїв НЕ включено). Виконайте це резервне копіювання, коли ПЛК перебуває в стані зупинки. F. K5577 (резервне копіювання) ^{№1}, вказує на резервне копіювання програм і параметрів ПЛК на карту пам'яті. (включаються статус і значення зберігаючих пристроїв). Виконайте це резервне копіювання, коли ПЛК перебуває в стані зупинки. *Функції E та F такі самі, як і функція резервного копіювання в ISPSOFT Card Utility. Назва файлу за замовч.: SDCard\PLC CARD\AS300\SysDup\AS300_save_backup.dup G. K6755 (відновлення) ^{№2}, вказує на відновлення всіх даних ПЛК із карти пам'яті (файли резервної копії з коду функції E або F). Виконуйте відновлення, коли ПЛК перебуває в стані зупинки. Назва файлу за замовч.: SDCard\PLC CARD\AS300\SysDup\AS300_save_backup.dup Примітка: якщо вам потрібно відновити після увімкнення живлення, вам потрібно змінити назву файлу резервної копії та зберегти файл на SDCard \PLC CARD\AS300\SysDup\AS300 BACKUP.dup перед виконанням відновлення з увімкненого живлення. H. K5588 (резервне копіювання) ^{№1}, це вказує на резервне копіювання всіх реєстрів файлів FR з ПЛК на карту пам'яті. Виконайте це резервне копіювання, коли ПЛК перебуває в стані зупинки. Назва файлу за замовч.: SDCard\PLC CARD\AS300\SysDup\AS300_FR_backup.dup I. K8855 (відновлення) ^{№2}, він вказує на відновлення всіх реєстрів файлів FR з карти пам'яті на ПЛК. Виконуйте відновлення, коли ПЛК перебуває в стані зупинки. Ім'я файлу має збігатися з ім'ям файлу в H. (Назва файлу за замовч.: SDCard\PLC CARD\AS300\SysDup\AS300_FR_backup.dup) #1 : під час резервного копіювання, якщо є файл із такою ж назвою, система перезаписує наявний файл. #2 : під час відновлення, якщо шлях до файлу та ім'я файлу не збігаються з файлом резервної копії, система не виконує відновлення. ПРИМІТКА 1: імена файлів за замовч., наведені вище, використовують як приклад AS300. Якщо ПЛК, який ви використовуєте, НЕ AS300, ви можете змінити ім'я файлу. Якщо ім'я файлу резервної копії неправильне, система не виконує відновлення та надсилає коди помилок.

ПРИМІТКА 2. Коды функцій A, B, C і D, показані в SR36, доступні для мікропрограми V1.04.30 або новішої версії.

Функціональні коды E, F і G у SR36 доступні для мікропрограми версії 1.04.90 або новішої. Функціональні коды H та I у SR36 доступні для мікропрограми версії 1.05.30 або новішої.

ПРИМІТКА 3. Визначення вищезгаданої програми ПЛК, параметрів і пристроїв, які можна зберегти:

Програма ПЛК включає коды виконання програмування ПЛК, коды програмування, пароль проекту та ідентифікатор ПЛК.

Параметри включають конфігурації HWCONFIG, таблицю відображення даних і таблицю планування позиції.

Зафіксована область означає параметри, і всі значення, встановлені в цій області, можна зберегти.

● Під час відновлення програми ПЛК із карти пам'яті система перевіряє файл резервної копії, ідентифікатор ПЛК та пароль.

- ◆ Перевірка ідентифікатора ПЛК: якщо для файлу резервної копії встановлено ідентифікатор ПЛК, це запобігає відновленню програми ПЛК на неправильному ПЛК.

	Модуль CPU з ідентифікатором PLC	Модуль CPU без ідентифікатора PLC
Файл резервної копії з ідентифікатором ПЛК	Ідентифікатор ПЛК збігся: перевірка пароля Ідентифікатор ПЛК не збігається: не відновлюється	Не відновлюється
Файл резервної копії без ідентифікатора ПЛК	Не відновлюється	Перевірка пароля

- ◆ Перевірка пароля: якщо для файлу резервної копії встановлено пароль, це забезпечує безпеку програми ПЛК і запобігає відновленню програми ПЛК на неправильному ПЛК.

	Модуль CPU з паролем	Модуль CPU без пароля
Файл резервної копії з паролем	Пароль збігся: відновлюється Пароль не збігається: не відновлюється	Відновлення, і ви також можете встановити модуль ЦП з паролем
Файл резервної копії без пароля	Не відновлюється	Відновлення

● SR36 зберігає лише 2 журнали :

- Число 1234 означає, що ПЛК зберігає журнал помилок (SR40–SR161) на карті пам'яті.
- Число 3456 означає, що ПЛК зберігає журнал помилок (SR40–SR161) і журнал змін стану ПЛК (SR309–SR390) на карті пам'яті.

13. Виконується інструкція високошвидкісного виведення. Вивід негайно припиняється, коли інструкція вимкнена або зупиняється.

- SM587: під час використання таблиці планування позиції для керування виходом вісь має посилатися на наступні позначки, щоб дія зупинилася. Ця функція доступна для мікропрограми версії 1.06.00 або новішої.

- SM476, SM477, SM496, SM497, SM516, SM517, SM536, SM537, SM556, SM557, SM576, SM577:

OFF. (за замовч.): зупинка уповільнення.

УВІМК.: негайна зупинка

- SM463, SM474, SM483, SM494, SM503, SM514, SM523, SM534, SM543, SM554, SM563, SM574 : використовуйте ці прапорці для призупинення виведення.

Коли прапор змінюється з OFF. на УВІМК., це означає припинення виведення. Це працює з цими прапорами; зверніться до розділу вище для отримання додаткових дій щодо зупинки.

Коли прапор змінюється з ON на OFF, це означає виконання решти виходів.

- Прапорець для осі виводу позиції 1-6 (парна кількість точок виведення високошвидкісного імпульсу) у зоні збільшення/зниження доступний для мікропрограми версії 1.06.00 або новішої.

● SM586

OFF. (за замовч.): використовуйте 250 кроків високої інтенсивності для перемикавання на різні швидкості в зоні підвищення/зниження, і частота помилок часу перемикавання становить приблизно 10%.

ON : використовуйте високоінтенсивні 25 кроків для перемикавання на різні швидкості в зоні підвищення/зниження, і частота помилок часу перемикавання становить близько 1%.

14. Вихідний ліміт контролю позиції в ISPSoft

● SR580-SR603

Обмеження позитивного виходу: установіть обмеження в ISPSoft. Коли вихідна позиція перевищує позитивну межу, вихід негайно припиняється.

Обмеження негативного виводу: установіть обмеження в ISPSoft. Коли вихідна позиція менша за негативне обмеження, вихід негайно припиняється.

Якщо як позитивний, так і негативний вихідні межі дорівнюють 0, функція вимкнена. Ця функція працює з інструкціями виведення. Система перевіряє обмеження, встановлене в ISPSoft, лише під час виконання інструкції. Таким чином, система не зупиняється миттєво, навіть якщо вона перевищує вихідний ліміт. Якщо потрібна негайна зупинка, рекомендується використовувати зовнішній вхід як спосіб перевірки обмеження.

15. Функція компенсації люфту

● SR478, SR479, SR498, SR499, SR518, SR519, SR538, SR539, SR558, SR559, SR578, SR579

Для серії AS можна встановити до 12 високошвидкісних виходів (Y0.0-Y0.11). Кожен вихід працює з відповідним SR, щоб користувачі могли встановити номер виходу для компенсації люфту. Діапазон налаштувань 0-32767. Якщо значення параметра

≤ 0 , ця функція вимкнена.

- Ця функція доступна для мікропрограми версії 1.02.30 і новішої. Інструкції виводу, які підтримують виведення осі непарних номерів і є спрямованим виведенням за замовч., це JOG, DZRN, DPLSV, DDRVI, DDRVA, DPPMR, DPPMA, DCICR, DCICA, DCICCR, DCICCA, DCCMR, DCCMA, DPPGB і TPO. Для вихідних сигналів осі парних номерів ви можете використовувати такі інструкції JOG, DPLSV, DDRVI та DDRVA. Для мікропрограми версії 1.10 або новішої також підтримується інструкція CSFO.

- Якщо використовуються вищезазначені підтримувані інструкції та інструкція зупиняється одразу після зміни напрямку виведення, якщо за цієї умови ввімкнено функцію компенсації люфту, ПЛК має запустити компенсовані інструкції перед справжньою зупинкою. Щоб не було некомпенсованих люфтів, коли буде включений наступний вихід.

16. Прапорці для показу номеру вихідного сигналу поточного виходу позиціонування в області збільшення/зниження.

(доступно для мікропрограми версії 1.06.00 або новішої)

● SR652-653 (32-розрядне значення)

Вихідне число в області наростання: накопичене вихідне число, підраховане від початкової точки прискорення до цільової точки швидкості (не включено).

● SR654-655 (32-розрядне значення)

Вихідне число в області спаду: накопичене вихідне число, підраховане від цільової точки швидкості (не включено) до вихідної точки зупинки.

17. Прапорці для відображення якості зв'язку CAN (доступно для прошивки V1.06.00 або новішої версії)

● SR823

Лічильник помилок прийому апаратного забезпечення для зв'язку CAN: коли номер помилки перевищує 127, це означає, що помилка апаратного забезпечення надто часта, вам потрібно перевірити кабелі зв'язку або відрегулювати швидкість передачі або змінити точки вибірки вручну.

● SR824

Лічильник апаратних помилок надсилання для зв'язку CAN: коли номер помилки перевищує 127, це вказує на те, що помилки апаратного забезпечення надто часті, вам потрібно перевірити кабелі зв'язку або налаштувати швидкість передачі чи змінити точки вибірки вручну.

18. Прапори, пов'язані з Ethernet IP

- SM1000, SR1000~SR1015 і (SR1016, SR1536~SR1539 доступні для мікропрограми версії 1.06.00 або новішої)

SM/SR	функція	Дія
SM1000	Прапор налаштування Ethernet.	ON: значення в SR1000–SR1006 записуються у флеш-пам'ять. Після запису значень ПЛК скидає його на OFF. ПРИМІТКА 1: не встановлюйте прапорець у положення ON постійно, щоб уникнути пошкодження флеш-пам'яті. ПРИМІТКА 2: перед записом значень у флеш-пам'ять ПЛК має бути в стані STOP.
SR1000	IP-адреса Ethernet (32-розрядна)	Наприклад: 192.168.1.5, SR1000 — 16#C0A8, а SR1001 — 0105.
SR1001		
SR1002	Адреса маски мережі Ethernet (32-розрядна)	Наприклад: 255.255.255.0, SR1002 — 16#FFFF, а SR1003 — FF00
SR1003		
SR1004	Адреса шлюзу Ethernet (32-розрядна)	Наприклад: 192.168.1.1, SR1004 — 16#C0A8, а SR1005 — 0101
SR1005		
SR1006	Тривалість з'єднання TCP	Одиниця: секунда; за замовч.: 30 секунд; діапазон налаштування: 1-32000; якщо значення налаштування перевищує діапазон налаштування, коли Якщо значення виходить за межі діапазону, система розглядає значення як мінімум 1 або як максимум 32000. Одиниця: Мбіт/с Кількість усіх TCP-з'єднань
SR1007	Поточна швидкість Ethernet	
SR1009	Номер поточного TCP-з'єднання	
SR1010	Час перезапустити зв'язок TCP	Одиниця: мс; за замовч.: 20 мс; діапазон налаштування: 0-32000; коли значення виходить за межі діапазону, система розглядає значення як мінімальне значення 0 або як максимальне 32000. Якщо значення налаштування дорівнює 0, це означає, що система автоматично перезапускає зв'язок.
SR1011	Номер підключення сервера Modbus/TCP	Система оновлюється автоматично.
SR1012	Номер підключення клієнта Modbus/TCP	Система оновлюється автоматично.
SR1013	Номер підключення адаптера EtherNet/IP	Система оновлюється автоматично.
SR1014	Номер підключення сканера EtherNet/IP	Система оновлюється автоматично.
SR1015	Тайм-аут отримання пакета сервера Modbus/TCP	Одиниця: мс; за замовч.: 10 мс; діапазон налаштування: 0-32000; коли значення перевищує діапазон, система розглядає значення як мінімум 0 або максимум 32000
SR1016	Час затримки для надсилання даних після встановлення з'єднання клієнта Modbus/TCP. (мс) (0~1000)	Одиниця: мс; за замовч.: 10 мс; діапазон налаштування: 0-1000; коли значення перевищує діапазон, система розглядає значення як мінімальне 0 або максимальне 1000
SR1536	Поточний AS-FEN02 / AS-FECAT TCP номер підключення	Кількість усіх TCP-з'єднань AS-FEN02
SR1537	AS-FEN02 / AS-FECAT Modbus/TCP Обслуговує номер підключення	Система оновлюється автоматично.

SM/SR	функція	Дія
SR1538	AS-FEN02 Номер підключення клієнта Modbus/TCP	Система оновлюється автоматично.
SR1539	Номер підключення адаптера EtherNet/IP AS-FEN02	Система оновлюється автоматично.

- Пов'язана з якістю зв'язку Ethernet, доступна для мікропрограми AS CPU версії 1.10.00 або новішої.

SR	функція	Дія
SR1017	Максимальна кількість пакетів Ethernet, які можна обробити за один цикл сканування.	За замовч.: 80 пакетів, діапазон налаштувань: 10-1000; коли значення перевищує діапазон, система розглядає значення як мінімум 10 або як максимум 1000. Регістр оновлено вами.
SR1018	Кількість пакетів Ethernet, оброблених в останньому циклі сканування.	Оновлювати реєстр кожен цикл сканування системою.
SR1019	Накопичена кількість пакетів Ethernet, які відкидаються після заповнення буферної зони.	1. Оновлювати реєстр кожен цикл сканування системою. 2. Ви можете очистити реєстр до 0.

приклад:

1. Визначте якість зв'язку Ethernet Крок 1: Перевірте, чи значення в SR1019 дорівнює 0.

SR1019 = 0	Хороша якість зв'язку
SR1019 > 0	Під час зв'язку сталася помилка, перейдіть до кроку 2

Крок 2. Перевірте поточні значення в SR1017 і SR1018

Результат SR1018/SR1019 < 0,8	Якщо робоче навантаження зв'язку становить менше 80%, якість зв'язку хороша; встановіть значення в SR1019 на 0. Якщо помилка виникає час від часу, попросіть адміністратора мережі перевірити, чи є будь-який пристрій, який може викликати помилку зв'язку в мережі.
Результат SR1018/SR1019 становить ≥ 0,8	Якщо робоче навантаження зв'язку більше або дорівнює 80%, зв'язок має високе навантаження. Якщо значення в SR1019 НЕ 0, це означає, що зв'язок може бути на обмін даними або оновити

2. Оновіть ємність мережі

- ◆ Коли ви підвищуєте значення в SR1017, значення в SR1019 також має бути підвищено. Але якщо значення в SR1019 залишається незмінним, але результат SR1018/SR1017 все ще високий (більше 0,8), це означає, що робоче навантаження зв'язку все ще досить повне. Пропонується підвищити значення в SR1017, і коефіцієнт має бути нижчим за 0,8. Вам потрібно очистити значення в SR1019, щоб перевірити, чи воно працює щоразу, коли ви змінюєте значення в SR1017.

SR1017		150
SR1018		110
SR1019		0

- ◆ Коли мережевий трафік інтенсивний, якщо ви підвищуєте значення в SR1017, кожне збільшення значення на 10 збільшує час сканування приблизно на 2,2 мс. (тестове середовище: немає модуля розширення, немає RTU або таблиці обміну даними.) Тому після підвищення значення в SR1017 вам потрібно звернути увагу на сканування

час. Якщо мережевий трафік не надто великий, значення в SR1017 не сильно вплине на час сканування. Однак, якщо мережа ненормальна, ЦП використовує більше ресурсів для обробки пакетів, і це збільшить час сканування. Коли ви встановлюєте значення в SR1017, вам потрібно враховувати робоче навантаження мережі та час сканування.

- ◆ Якщо пакети не можуть бути оброблені протягом допустимого часу сканування, ви можете розглянути наступне, щоб зменшити навантаження на мережу.
 - Зменште кількість мережевих з'єднань із центральним процесором, наприклад, зменшіть кількість обміну даними через Modbus TCP або EtherNet/IP.
 - Знизьте частоту оновлення в обміні даними, наприклад, зменшіть цикл оновлення для обміну даними через Modbus TCP або збільште значення RPI для обміну даними через EtherNet/IP.

● SM1090, SM1091, SM1106-SM1109

SM	функція	Дія
SM1090	TCP-з'єднання зайняте.	ON: Тайм-аут з'єднання TCP
SM1091	UDP-з'єднання зайняте.	ON: тайм-аут з'єднання UDP
SM1110	AS-FEN02 Ethernet - Помилка налаштування фільтра	Коли програма завантажується в ПЛК
SM1106	Помилка підключення Ethernet	ON: не вдається ініціалізувати PHY. OFF.: ініціалізація PHY успішна.
SM1107	Помилка базового налаштування Ethernet	ON: помилка основного налаштування OFF: виправте базове налаштування
SM1109	Локальний порт сокета TCP/UDP уже використовується.	ON: використовується той самий порт

Коди помилок, відповідні світлодіодні індикатори та інші способи усунення несправностей див. у розділі 12 Посібника з експлуатації серії AS300.

19. Налаштування електронної пошти

● SM1113, SM1116-SM1155

Якщо надсилання електронної пошти не вдається, прапор помилки служби електронної пошти SM1113 увімкнено.

У наведеній нижче таблиці наведено тригери для надсилання електронної пошти та відповідні позначки (SM1116–SM1155).

функція \ Пункт	Тригер 1	Тригер 2	Тригер 3	Тригер 4
Служба електронної пошти	SM1116	SM1126	SM1136	SM1146
	ON: увімкнено, OFF: вимкнено			
Надсилання електронною поштою	SM1117	SM1127	SM1137	SM1147
	ON: надсилання електронного листа зараз, ВИМКНЕНО: повідомлення надіслано			
Листи успішно надіслано	SM1119	SM1129	SM1139	SM1149
	ON: електронний лист надіслано успішно			
Помилка надсилання електронної пошти 1	SM1120	SM1130	SM1140	SM1150
	Неможливо надіслати електронний лист через помилку вмісту електронного листа			
Час очікування відповіді SMTP	SM1122	SM1132	SM1142	SM1152

функція \ Пункт	Тригер 1	Тригер 2	Тригер 3	Тригер 4
	Після надсилання електронного листа час очікування відповіді SMTP-сервера.			
Помилка відповіді сервера SMTP	SM1123	SM1133	SM1143	SM1153
	Після надсилання електронного листа помилка відповіді сервера SMTP.			
Помилка надсилання електронної пошти 2 з	SM1124	SM1134	SM1144	SM1154
	Після надсилання електронного листа розмір вкладення перевищує ліміт.			
Помилка надсилання електронного листа 3	SM1125	SM1135	SM1145	SM1155
	Після надсилання електронного листа вкладення не знайдено, SM1125 увімкнено.			

20. Прапори та реєстри щодо обміну даними

- Прапори для обміну даними через з'єднання COM1

SM	Тип	функція
SM750	R/W	Обмін даними через COM1 підтримує ISPSOft.
SM752–SM783	R/W	Розпочато з'єднання 1–32 через COM1 для обміну даними.
SM784–SM815	R	Дані отримані через з'єднання 1–32 COM 1 для обміну даними.
SM816–SM847	R	Помилка підключення COM1 1–32 для обміну даними.

- Прапорці для обміну даними через з'єднання COM2

SM	Тип	функція
SM862	R/W	Обмін даними через COM2 підтримує ISPSOft.
SM864–SM895	R/W	Розпочато з'єднання 1–32 через COM2 для обміну даними.
SM896–SM927	R	Дані, отримані через з'єднання COM2 1–32 для обміну даними.
SM928–SM959	R	Помилка підключення COM2 1–32 для обміну даними

- Реєстри для обміну даними через з'єднання COM1 і COM2

SR	функція
SR1335	Фактичний час циклу підключення 1–32 для обміну даними через COM1
SR1336	Номер підключення, яке в даний момент здійснює циклічний обмін даними через COM1
SR1340–SR1371	Код помилки обміну даними через з'єднання COM1 1–32
SR1375	Фактичний час циклу підключення 1–32 для обміну даними через COM2
SR1376	Номер підключення, яке в даний момент виконує циклічний обмін даними через COM2
SR1380–SR1411	Код помилки обміну даними через з'єднання COM2 1–32

Коди помилок 1–7 є стандартними кодами помилок відповідей протоколу Modbus. Код помилки 9 означає час очікування.

- Прапори для обміну даними через Function Card 1 і Function Card 2

SM	Тип	Функціональна картка 1
----	-----	------------------------

SM1710	R/W	Обмін даними через функціональну картку 1, увімкнену ISPSoft
SM1712 ~ SM1743	R/W	Розпочато з'єднання 1-32 для обміну даними через функціональну картку 1
SM1744 ~ SM1775	R	Успішне з'єднання для обміну даними 1-32 через функціональну картку 1
SM1776 ~ SM1807	R	Помилка підключення обміну даними 1-32 через функціональну картку 1
SM	Тип	Функціональна картка 2
SM1822	R/W	Обмін даними через функціональну картку 1, увімкнену ISPSoft
SM1824 ~ SM1855	R/W	Розпочато з'єднання 1-32 для обміну даними через функціональну картку 1
SM1856 ~ SM1887	R	Успішне з'єднання для обміну даними 1-32 через функціональну картку 1
SM1888 ~ SM1919	R	Помилка підключення обміну даними 1-32 через функціональну картку 1

- Регістри для обміну даними через Funciton Card 1 і Funciton Card 2 (доступно для прошивки V1.06.00 або пізнішої версії)

SR	функція
SR1435	Фактичний час циклу для обміну даними через функціональну картку 1
SR1436	Номер підключення, яке в даний момент виконує циклічний обмін даними через функціональну картку 1
SR1440 ~ SR1471	Помилка підключення обміну даними 1-32 через функціональну картку 1 Коди помилок 1-7 пов'язані з проблемами Modbus; зверніться до інструкції від підлеглих для отримання додаткової інформації. Код помилки 9 вказує на час очікування зв'язку.
SR1475	Фактичний час циклу для обміну даними через функціональну картку 2
SR1476	Номер підключення, яке в даний момент виконує циклічний обмін даними через функціональну картку 2
SR1480 ~ SR1511	Помилка підключення обміну даними 1-32 через функціональну картку 2 Коди помилок 1-7 пов'язані з проблемами Modbus; зверніться до інструкції від підлеглих для отримання додаткової інформації. Код помилки 9 вказує на час очікування зв'язку.

- Прапори для обміну даними через з'єднання Ethernet

SM	Тип	функція
SM1167	R/W	Обмін даними через порт Ethernet, включений ISPSoft.
SM1168–SM1199	R/W	Розпочато підключення 1–32 через порт Ethernet для обміну даними.
SM1200–SM1231	R	Дані, отримані через підключення порту Ethernet 1–32 для обміну даними.
SM1232–SM1263	R	Помилка підключення порту Ethernet 1–32 для обміну даними

- Регістри для обміну даними через з'єднання Ethernet

SR	функція
SR1120–SR1151	Фактичний час з'єднання для обміну даними через з'єднання Ethernet 1–32
SR1152–SR1183	Код помилки обміну даними через з'єднання Ethernet 1–32

- Прапори для обміну даними через Ethernet-з'єднання в ЦП ПЛК серії AS (доступно для мікропрограми версії 1.10.20 або новішої)

SM	Тип	функція
SM1092	R/W	Неправильний порт сервера Modbus TCP

SR	Funciton
SR1092	Порт сервера Modbus TCP (ЦП ПЛК як сервер)
SR1093	Порт клієнта Modbus TCP (ЦП ПЛК як клієнт)

Деталі операції:

- Коли ЦП ПЛК діє як сервер Modbus TCP
 - За замовч. протокол використовує порт 502 як комунікаційний порт у зв'язку Modbus TCP.
 - Але якщо порт 502 недоступний, ви можете використовувати SR1092, щоб налаштувати порт для зв'язку.
 - Налаштування тут не можна зберегти. Коли ви вимикаєте та вмикаєте ЦП ПЛК, за замовч. порт зв'язку знову буде портом 502. Для налаштування використовуйте програму PLC.
 - Якщо номер порту, встановлений у SR1092, уже зайнятий іншими функціями, такими як сокет або веб. Налаштування тут буде недійсним. І SM1092 увімкнеться, вказуючи на неправильний порт сервера Modbus TCP.
- Коли ЦП ПЛК діє як клієнт Modbus TCP
 - За замовч. протокол використовує порт 502 як комунікаційний порт у зв'язку Modbus TCP.
 - Але якщо порт 502 недоступний, ви можете використовувати SR1093, щоб налаштувати порт для зв'язку.
 - Налаштування тут не можна зберегти. Коли ви вимикаєте та вмикаєте ЦП ПЛК, за замовч. порт зв'язку знову буде портом 502. Для налаштування використовуйте програму PLC.
 - Значення налаштування тут лише для комунікаційного порту клієнта Modbus TCP. Комунікаційні порти ЦП ПЛК НЕ будуть вплинути.

- Прапори для обміну даними через з'єднання Ethernet в AS-FEN02 (доступно для прошивки V1.06.00 або пізнішої версії)

SM	Тип	функція
SM1006	R/W	Обмін даними через AS-FEN02 підтримує ISPSOft.
SM1008 ~ SM1015	R/W	Розпочато з'єднання 1–8 через AS-FEN02 для обміну даними.
SM1016 ~ SM1023	R	Успішне з'єднання для обміну даними 1-8 через AS-FEN0 2
SM1024 ~ SM1031	R	Помилка з'єднання AS-FEN02 1–8 для обміну даними

- Регістри для обміну даними через з'єднання Ethernet в AS-FEN02 (доступно для прошивки V1.06.00 або пізнішої версії)

SR	Funciton
SR1520 ~ SR1527	Фактичний час зв'язку для з'єднання обміну даними 1-8 через AS-FEN02
SR1528 ~ SR1535	Помилка підключення обміну даними 1-8 через AS-FEN02

- Коди помилок для підключень Ethernet

Код помилки	опис
16#00XX	Помилка відповіді віддаленого модуля Коди 1~7 є стандартними кодами помилок протоколу Modbus TCP
16#F000	З'єднання Ethernet не встановлено
16#F001	Час очікування відповіді віддаленого модуля
16#F003	Тайм-аут з'єднання TCP
16#F007	Помилка відповіді
16#F009	У віддаленому модулі втрачено з'єднання

- Список станів SM/SR при підключенні до RTU-EN01 через порт Ethernet

SM / SR		опис
SM1312 - SM1315		Прапори стану зв'язку з'єднання RTU-EN01 ID 1-4
SR1312 - SR1315		Коди стану зв'язку з'єднання RTU-EN01 ID 1-4
SR	стан CM	опис
0	OFF	Підключення закрито
1	ON	Вдале підключення
2	OFF	Час очікування зв'язку
3	OFF	З'єднання закрито примусово
4	OFF	Відповідь RTU-EN01: помилка у вмісті
5	OFF	Відповідь RTU-EN01: помилка
6	OFF	Мережа не підключена або підключення не вдалося
стан CM		RX & RCR Читання даних (область відображення введення) RY & RCR Запис даних (область відображення виводу)
OFF → ON		Очистити до 0
ON → OFF		Без змін

Примітка. Перед встановленням з'єднання рекомендується НЕ використовувати дані (RX/RY/RCR Read/RCR Write) в області відображення.

- Прапори зв'язку Ethernet/IP доступні для програмного забезпечення PLC версії 1.06 або новішої.

SM	Attr.	Опис обміну даними Ethernet/IP
SM1375	R/W	Розпочато обмін даними через EtherNet/IP (сканер).
SM1376 - SM1407	R/W	Розпочато підключення 1-32 для обміну даними через EtherNet/IP (сканер).
SM1408 - SM1439	R	Помилка підключення обміну даними 1-32 через EtherNet/IP (сканер)
SM1440 - SM1447	R	Помилка підключення введення/виведення 1-8 через EtherNet/IP (адаптер)
SM1448 – SM1455 *1	R	Підключення I/O 1-8 встановлюється через EtherNet/IP (адаптер)

*1: доступно лише для PLC FW V1.06.50 або новішої версії

- Комунікаційні реєстри Ethernet/IP доступні для програмного забезпечення PLC версії 1.06 або новішої.

SR	Attr.	Опис обміну даними Ethernet/IP
SR1020 - SR1051	R	Код статусу підключення 1-32 для обміну даними через EtherNet/IP (сканер)

- Нижче наведено коди помилок зв'язку Ethernet/IP. Для отримання додаткової інформації зверніться до розділу 9 посібника з експлуатації серії AS.

Код помилки	опис	Рішення
16#0100	З'єднання введення/виведення дублюються	1. Перевірте, чи система створила з'єднання введення/виведення. 2. Змініть тип підключення на Лише прослуховування.
16#0106	Конфлікти прав власності сканера на підключення вводу-виводу	1. Перевірте власника сканера. 2. Переналаштуйте недійсний сканер. 3. Змініть підключення на багатоадресне.
16#0110	Неправильна конфігурація адаптера	1. Перевірте стан підключення введення/виведення. 2. Знову активуйте підключення введення/виведення.
16#0111	RPI адаптера не підтримується	Перевірте RPI для адаптера.
16#0113	Недостатньо з'єднань введення/виведення	1. Перевірте, чи не перевищує обмеження кількість підключень 2. Зменшіть кількість підключень продукту.
16#0119	Не встановлено лише з'єднання без прослуховування.	1. Перевірте, чи система створила з'єднання введення/виведення. 2. Перевірте стан підключення вводу/виводу сканера.

16#0127	Неправильний вхідний розмір адаптера	Перевірте номер модуля та файл налаштування продукту, щоб перевірити, чи вони збігаються.
16#0128	Неправильний вихідний розмір адаптера	Перевірте вихідний розмір у параметрах підключення.
16#012D	Спожитий TAG не існує.	Перевірте, чи правильно налаштовано параметри в тегу Consumed.
16#012E	Виготовлений TAG не існує.	Перевірте, чи правильно встановлено параметри в тегу Produced.
16#0204	Тайм-аут з'єднання вводу-виводу	Немає відповіді від адаптера; перевірте, чи живлення та підключення до мережі для адаптера працюють належним чином.
16#0302	Пропускна здатність мережі НЕ доступна для даних	1. Перевірте ліміт з'єднання введення/виведення між сканером і адаптером. 2. Збільште значення RPI або зменшіть кількість підключень.
16#0315	Неправильний екземпляр введення/виведення адаптера	Перевірте номер модуля та файл налаштування продукту, щоб перевірити, чи вони збігаються.
16#0203	Тайм-аут зв'язку з'єднання введення/виведення	Немає відповіді від адаптера; перевірте, чи живлення та підключення до мережі для адаптера працюють належним чином.

21. Прапори SM230-SM261 призначені для активації модулів при виборі «Вручну + прапорці» в «Налаштуванні розподілу вводу/виводу при включенні».

Вибираючи «Вручну + прапорці (тільки модуль вводу-виводу модуля ЦП)» у «Налаштуванні розподілу вводу-виводу під час увімкнення», використовуйте прапорці SM230-SM261, щоб активувати 1-й – 32-й модуль праворуч від модуля ЦП. (доступно для FW V1.06.00 або новішої версії).

Вибираючи «Вручну + прапорці (модуль вводу/виводу ЦП і віддалений модуль)», використовуйте прапорці SM230-SM261, щоб віддалено активувати 1–32 модулі праворуч від модуля ЦП (доступно для FW V1.08.50 або новішої версії). і ISPSOFT V3.12 або пізнішої версії підтримує цю функцію.)

Для додатків з меншою кількістю модулів ви можете використовувати ту саму програму ПЛК і один раз завантажити таблицю розподілу вводу-виводу для різних додатків вводу-виводу. Застосовуючи цей режим, слід звернути увагу на дві речі:

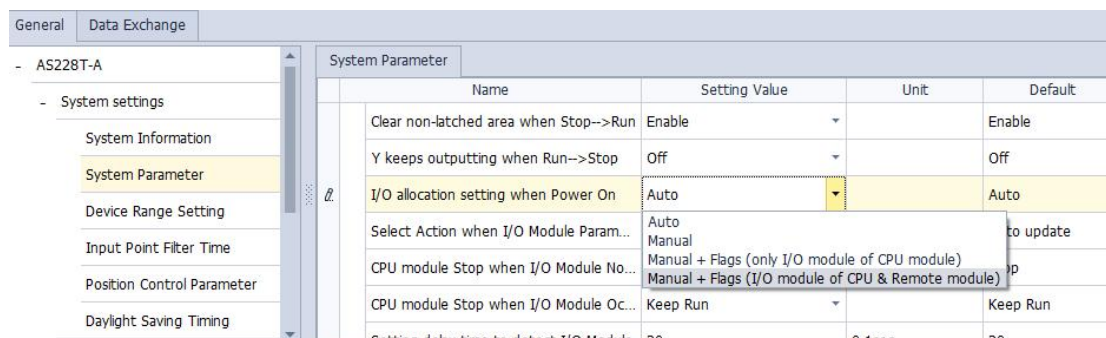
А. Вам потрібно заздалегідь використовувати більше планування розподілу вводу/виводу та зберігати це планування в ПЛК серії AS.

В. Ви можете просто використовувати спеціальні прапорці, щоб позначити, який модуль вводу/виводу НЕ використовуватиметься відповідно до фактичного розташування модуля вводу/виводу. Але ви не можете змінити вихідний порядок розташування модулів.

Додайте новий вибір «Вручну + прапорці» для опції налаштування «Налаштування розподілу модулів вводу/виводу під час увімкнення живлення». Виберіть цю опцію, її потрібно використовувати з прапорцями SM230-SM261. Для додатків із меншою кількістю модулів ви можете просто використовувати спеціальні прапорці, щоб позначити, який модуль вводу/виводу НЕ використовуватиметься відповідно до фактичного розташування модуля вводу/виводу. У цьому режимі ви можете використовувати ту саму програму ПЛК і один раз завантажити таблицю розподілу вводу/виводу для різних додатків вводу/виводу.

приклад:

Крок 1: Виберіть «Manual + Flags (I/O module of CPU & Remote module)» для опції налаштування «I/O module allocation setting when Power ON» у HWCONFIG.



Крок 2: Створіть весь розподіл модулів і завантажте таблицю розподілу вводу/виводу в HWCONFIG.

Крок 3: Якщо 16AP, 04DA, 08AM (правий модуль) і 16AP, а також 06X6 (віддалений модуль) не потрібні в програмі, ви можете встановити прапорці SM231 (вказує на 16AP), SM233 (вказує на 04DA), SM235 (включаючи 08:00), SM237 (вказує на віддалений 16AP) і SM239 (вказує на віддалений 06X6) у положення OFF., щоб відповідати фактичному розміщенню модуля введення/виведення. Вам не потрібно змінювати їх номер замовлення в програмі PLC. Після вимкнення та повторного увімкнення живлення фактичне розташування виглядає як показано нижче. Номер IO у програмі ПЛК змінювати НЕ потрібн



Фактичне розміщення модуля введення/виведення



Номери замовлення, назви моделей і відповідні номери пристроїв у повному списку розподілу вводів/виводів. Праві модулі:

Номер замовлення	0	1	2	3	4	5	6	7
Модель	AS332	16AP	16AP	04 AD	04DA	06XA	08AM	08AN
Відповідний пристрій починається з		X1.0/ Y1.0	X2.0/ Y2.0	D28040	D28060	D28080	X3.0	Y3.0

Віддалені модулі:

Номер замовлення	0	1	2	3	4
Модель	SCM	16AP	06XA	06XA	16AP
Відповідний пристрій починається з	D29000	X4.0/Y4.0	D59040	D29060	X5.0/Y5.0

Номер замовлення та відповідні номери пристроїв залишаються незмінними.

Праві модулі:

Номер замовлення	0	1	3	5	7
Модель	AS332	16AP	04 AD	06XA	08AN
Відповідний пристрій починається з		X1.0/Y1.0	D28040	D28080	Y3.0

Віддалені модулі:

Номер замовлення	0	2	4
Модель	SCM	06XA	16AP
Відповідний пристрій починається з	D29000	D59040	X5.0/Y5.0

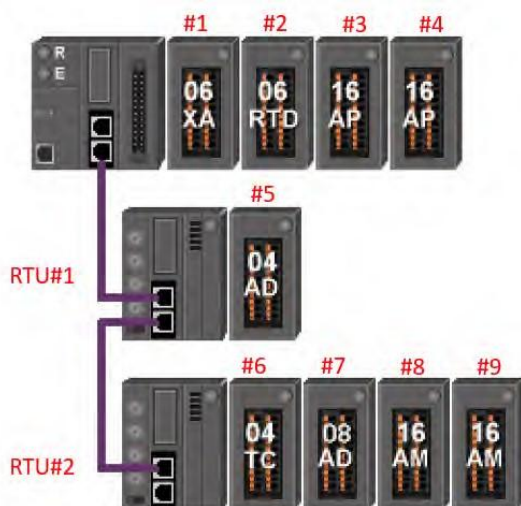
Примітка. Правило номера замовлення також застосовується, коли ви використовуєте FROM/TO або інші інструкції, пов'язані з модулем.

22. Кількість модулів і коди пристроїв

SR	опис
SR1559	Кількість віддалених модулів, підключених до модуля ЦП (усі праві модулі не включені)
SR1560	Кількість модулів правої сторони, підключених до модуля CPU
SR1561~SR1592	Код моделі 1-го – 32-го правого модуля, підключеного до модуля CPU або віддаленого модуля

Модулі DIO					
Модель	Код пристрою	Модель	Код пристрою	Модель	Код пристрою
AS08AM10N-A	16#0201	AS08AN01T-A	16#0206	AS08AN01P-A	16#0207
AS08AN01R-A	16#0208	AS16AM10N-A	16#0210	AS16AP11T-A	16#0212
AS16AP11P-A	16#0213	AS16AP11R-A	16#0214	AS16AN01T-A	16#0216
AS16AN01P-A	16#0217	AS16AN01R-A	16#0218	AS32AM10N-A	16#0220
AS32AN02T-A	16#0226	AS64AM10N-A	16#0230	AS64AN02T-A	16#0236
Модулі AIO (температура, тензодатчик)					
Модель	Код пристрою	Модель	Код пристрою	Модель	Код пристрою
AS04AD-A	16#0301	AS08AD-B	16#0302	AS08AD-C	16#0303
AS02ADH-A	16#0304	AS04DA-A	16#0305	AS06XA-A	16#0309
AS04RTD-A	16#0310	AS06RTD-A	16#0311	AS04TC-A	16#0315
AS08TC-A	16#0316	AS02LC-A	16#0320		
NIO модулі (зв'язок)					
Модель	Код пристрою	Модель	Код пристрою	Модель	Код пристрою
AS00SCM-A	16#0401	AS01DNET-A	16#0402		
Модулі PIO (позиціонування та лічильник)					
Модель	Код пристрою	Модель	Код пристрою	Модель	Код пристрою
AS02PU-A	16#0501	AS04PU-A	16#0502	AS02HC-A	16#0505

приклад:



Після ввімкнення розподіл модулів у наведеному вище прикладі виглядає так, як показано нижче.

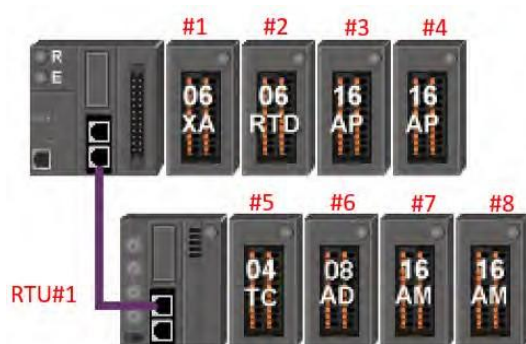
SR1559 = 2; SR1560 = 4

Номер замовлення підраховується модулями правої сторони, підключеними до модуля ЦП, а потім підраховується за номером RTU.

Номер СР (замовлення)	SR1651 (№1)	SR1652 (№2)	SR1653 (№3)	SR1654 (№4)	SR1655 (№5)
Код пристрою	16#0309	16#0311	16#0212	16#0212	16#0301
Номер СР (замовлення)	SR1656 (№6)	SR1657 (№7)	SR1658 (№8)	SR1659 (№9)	
Код пристрою	16#0315	16#0302	16#0210	16#0210	

Якщо RTU відключено, після повторного підключення система оновлює SR.

SR1559 = 1; SR1560 = 4



23. Автоматичне скидання прапорів для імпульсного виходу

- Під час виконання інструкцій імпульсного виведення, щоб уникнути керування одним і тим же пристроєм імпульсного виведення, багаторазово приймаються різні типи або різні інструкції виведення. ПЛК автоматично скидає право керувати імпульсною інструкцією до першої виконаної інструкції, доки призначена інструкція не буде повністю зупинена. Через унікальну характеристику сканування ПЛК не може виявити, чи була зупинена імпульсна інструкція, коли ви використовуєте інструкцію імпульсного виведення на мові ST, підпрограму переривання або одноразовий функціональний блок. Таким чином, право керувати імпульсними інструкціями зберігається, і будь-які поточні інструкції не можуть бути ні виконані, ні виведені нормально.
- Щоб звільнити право виконувати інші команди імпульсного виведення, ви можете використовувати прапори автоматичного скидання для ПЛК, щоб перевірити, чи завершено вихід.

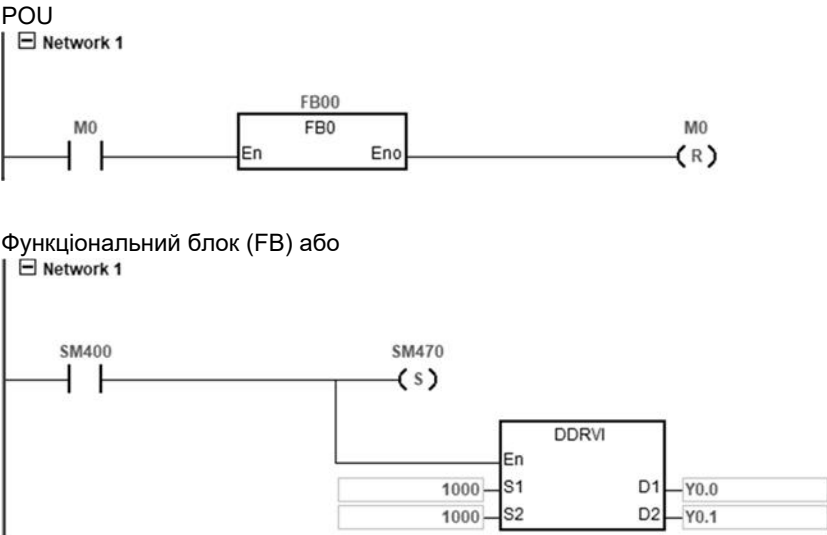
Примітка: ви можете встановити прапорці автоматичного скидання на ON, і після завершення виведення прапорець автоматично скинеться на OFF.

Парна вісь для виведення	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
Автоматичне скидання після завершення виведення.	SM470	SM490	SM510	SM530	SM550	SM570
Непарна вісь для виведення	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Автоматичне скидання після завершення виведення.	SM475	SM495	SM515	SM535	SM555	SM575

Список інструкцій імпульсного виходу:

Інструкція	Навіть числова вісь	Непарний числова вісь	авто- прапор скидання	Автоматичне скидання часу	Зауваження
PWM/DPWM	B	B	Так	Прапор автоскидання ввімкнено	- Немає необхідності встановлювати кількість імпульсів для виведення. - Пропонується використовувати в додатках, що інструкція зупиняється, і інструкція не буде скануватися знову. (наприклад, програмування мовою ST, переривання та одноразовий функціональний блок).
JOG/DJOG	B	B			
DPLSV	B	B			
DPPGB	B				
CSFO	B		Так	Прапор автоматичного скидання ввімкнено, і імпульсний вихід завершено.	- Необхідно налаштувати кількість імпульсів для виведення. - Спрямований вихід інструкції з позитивними/негативними межами - Більше двох осі, кожна вісь вимагає прапора автоматичного скидання; одна вісь, один прапор автоматичного скидання.
PLSR	B	B			
PLSY	B	B			
DDRVi	B	B			
DDRVA	B	B			
DDRVM	B				
DPPMR	B				
DPPMA	B				
DCICR	B				
DCICA	B				
DCICCR	B				
DCICCA	B				
DCCMR	B				
DCCMA	B				
DZRN	B		Так FW: V1.12.00 або пізніше	Прапор автоматичного скидання ввімкнено, і імпульсний вихід завершено.	- Інструкції будуть скануватися та контролюватись; немає потреби в прапорці автоматичного скидання. - НЕ використовуйте його в одноразовому функціональному блоці. - Ви повинні використовувати прапор автоматичного скидання разом з інструкцією при виконанні на мові програмування ST.
DZRN2	B	B			

Приклад 1 (використовується у функціональних блоках або перериваннях)



Пояснення

- i. Коли M0 увімкнено, Y0.0 виводить 1000 імпульсів на 1 кГц і встановлює SM470 на Y0.0.
- ii. Коли виведення Y0.0 завершено, ПЛК звільняє право на повторну готовність Y0.0.
- iii. Коли M0 знову переходить з OFF. на УВІМК., вихід Y0.0 може початися знову.

Приклад 2 (використовується в мові програмування ST)

```

0001 IF M0 THEN
0002     DDRVI(10000, 1000, Y0.0, Y0.1);
0003     M1 := TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
0008

```

Пояснення

- i. Коли M0 увімкнено, Y0.0 видає 1000 імпульсів із частотою 1 кГц, і встановить M1 у положення ON для запису Y0.0.
- ii. Коли M0 вимкнено, після завершення виведення Y0.0 SM461 увімкнено. ЯКЩО M1 увімкнено, він запускає прапор автоматичного скидання Y0.0 (SM470=ON), і ПЛК звільняє право для Y0.0 знову бути готовим.
- iii. Коли M0 знову переходить з OFF. на УВІМК., вихід Y0.0 може початися знову.

24. Прапори автоматичної зупинки для зв'язку Delta CAN

- Під час виконання інструкцій CAN, щоб уникнути контролю над одним і тим самим ідентифікатором сервоприводу/інвертора, багаторазово беруться команди CAN різного типу або різні запущені. ПЛК призначає право керувати імпульсною інструкцією першій виконаній інструкції до повної зупинки призначеної інструкції. Через унікальну характеристику сканування ПЛК не може виявити, чи зупинено інструкцію CAN, коли ви використовуєте інструкцію CAN на мові ST, підпрограму переривання або одноразовий функціональний блок. Таким чином, право керувати інструкціями CAN зберігається, і будь-які поточні інструкції не можуть бути виконані або нормально керувати сервоприводом/інвертором.
- Щоб звільнити право виконувати інші вихідні інструкції CAN, ви можете використовувати прапори автоматичного скидання для ПЛК, щоб перевірити, чи завершено зв'язок між Delta Servo або Delta Moto. Примітка: ви можете встановити прапорці автоматичного скидання на ON, і після завершення зв'язку CAN прапорець автоматично скинеться на OFF.

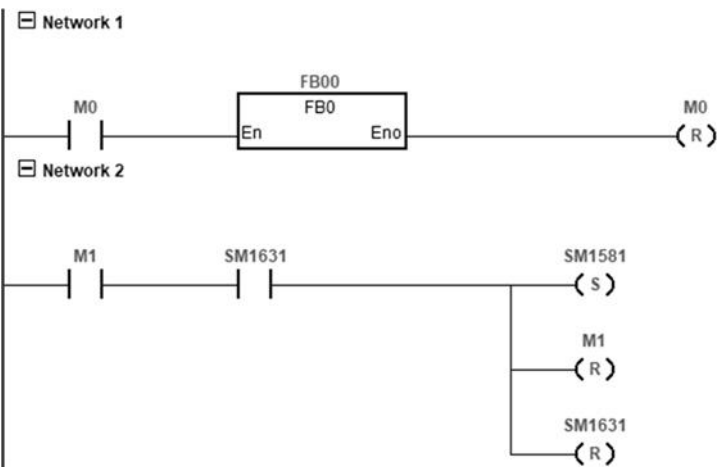
ID сервостанції Delta	1	2	3	4	5	6	7	8
Автоматичне скидання права на виконання інших інструкцій CAN.	SM1581	SM1582	SM1583	SM1584	SM1585	SM1586	SM1587	SM1588
Delta Moto Station ID	21	22	23	24	25	26	27	28
Автоматичне скидання права на виконання інших інструкцій CAN.	SM1601	SM1602	SM1603	SM1604	SM1605	SM1606	SM1607	SM1608

● Список інструкцій із зв'язку CAN:

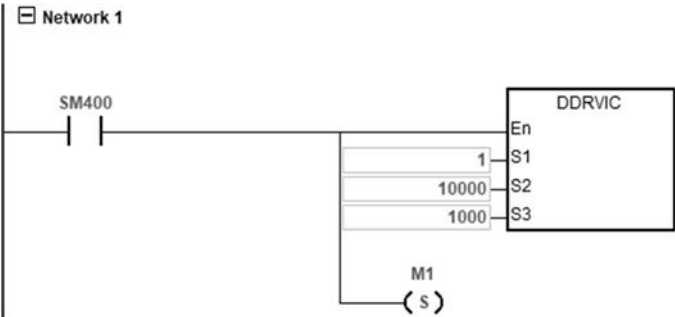
Інструкція	Delta Servo	Дельта інвертор	авто-прапор скидання	Автоматичне скидання часу	Зауваження
INIC	B	B	Так	Прапор автоскидання ввімкнено	
ASDON	B	B			
CASD	B	B			
DDRVIC	B				
DDRVAC	B				
DPLSVC	B	B			
ZRNC/DZRNC	B				
RSTD	B	B			
DCSFOC	B				
DTQC	B	B			
DTQLC	B	B			
ZRNM	B		N/A		Інструкції будуть скануватися та контролюватись; немає потреби в прапорці автоматичного скидання.
EMER	B	B			
COPRW	B	B			
COPWL/DCOPWL	B	B			
ECAMD	B				
DECAMS	B				
ECAMA	B				
ECAMC	B				

Приклад 1 (використовується у функціональних блоках)

POU:



FB:



Пояснення

- i. Коли M0 увімкнено, сервопривод №1 працює до відносного положення 10000 PUU зі швидкістю 100,0 об/хв. І встановить для M1 значення ON, щоб записати, що право зв'язку з Servo#1 використовується.
- ii. Коли SM1631 увімкнено (позиціонування завершено), а M1 увімкнено, переведить SM1581 у положення ON, щоб ПЛК звільнив право зв'язку для сервоприводу №1.
- iii. Коли M0 знову переходить із OFF. на УВІМК., сервопривод №1 знову готовий до позиціонування.

Приклад 2 (використовується в мові програмування ST)

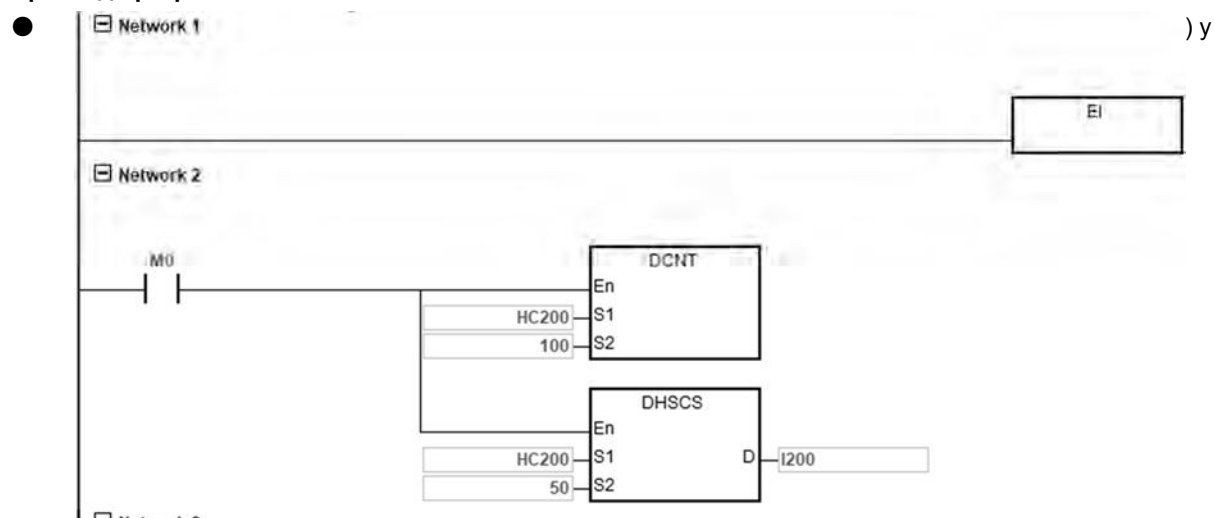
```

0001 IF M0 THEN
0002     DDRVIC(1, 10000, 1000);
0003     M1 := TRUE;
0004 ELSIF M1 AND SM1631 THEN
0005     SM1581 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
0008

```

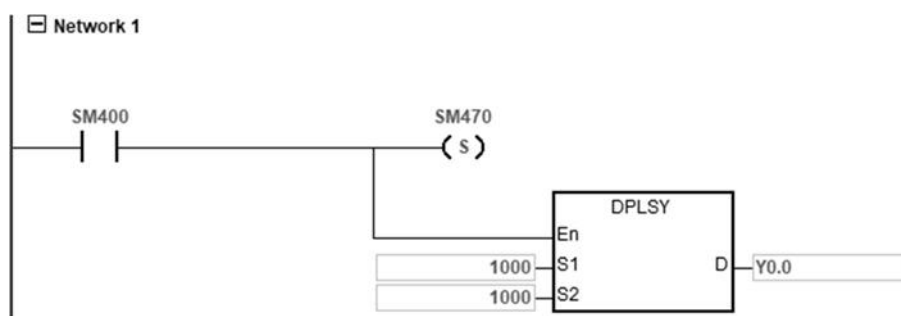
Пояснення

- Коли M0 увімкнено, сервопривод №1 працює до відносного положення 10000 PUU зі швидкістю 100,0 об/хв. І встановить для M1 значення ON, щоб записати, що право зв'язку з Servo#1 використовується.
- Коли M0 вимкнено, SM1631 увімкнено (позиціонування завершено) і M1 увімкнено, переведить SM1581 у положення ON, щоб ПЛК звільнив право зв'язку для сервоприводу №1.
- Коли M0 знову переходить із OFF. на УВІМК., сервопривод №1 знову готовий до позиціонування.

25. Використовуйте DPLSY у високошвидкісному перериванні порівняння (I2xx), доступному для FW V1.08.70 або новішої версії.
Приклад програми PLC:

Примітка: коли M0 увімкнено, запускається високошвидкісний лічильник HC200. Коли підраховане значення в S2 досягає 50, запускається високошвидкісне переривання порівняння I200.

- Записати інструкцію DPLSY у високошвидкісному перериванні порівняння I200.



Примітка: автоматичне скидання керування під час виведення 1000 імпульсів на частоті 1 кГц від Y0.0. є повними.

● Процедура

Є чотири процедури AD, коли вхідне підраховане значення змінюється 49 -> 50 для виведення імпульсів через Y0.0.

- A. Введене підраховане значення змінюється з 49 на 50.
- B. Спрацьовує переривання високошвидкісного порівняння. Програма PLC переривається, і виконується переривання високошвидкісного порівняння.
- C. Виконати інструкцію DPLSY, яка записана в перериванні високошвидкісного порівняння.
- D. Імпульси виводяться з Y0.0.

Оскільки на переривання впливає час його ініціювання та поточні виконання ПЛК, такі як оновлення точок вводу-виводу, впровадження зв'язку тощо, час виведення імпульсів може бути значно затриманим.

Щоб покращити час виконання, додано нові прапорці SM.

● SM120-SM151: Для попереднього збереження параметрів для виведення DPLSY

Встановіть SM у цій категорії на ON, щоб зберегти параметри DPLSY, які потрібно вивести, включаючи номер вихідної точки, цільову частоту та кількість імпульсів у флеш-пам'яті ПЛК. Коли спрацьовує високошвидкісне переривання порівняння і будь-який із SM360-391 встановлено на ON, вихід може бути розпочато негайно. Після використання будь-якого з SM120-151 для нього встановлюється значення OFF. Перед повторним використанням його потрібно увімкнути.

● SM360-SM391: щоб увімкнути запуск DPLSY у перериванні для вихідних імпульсів

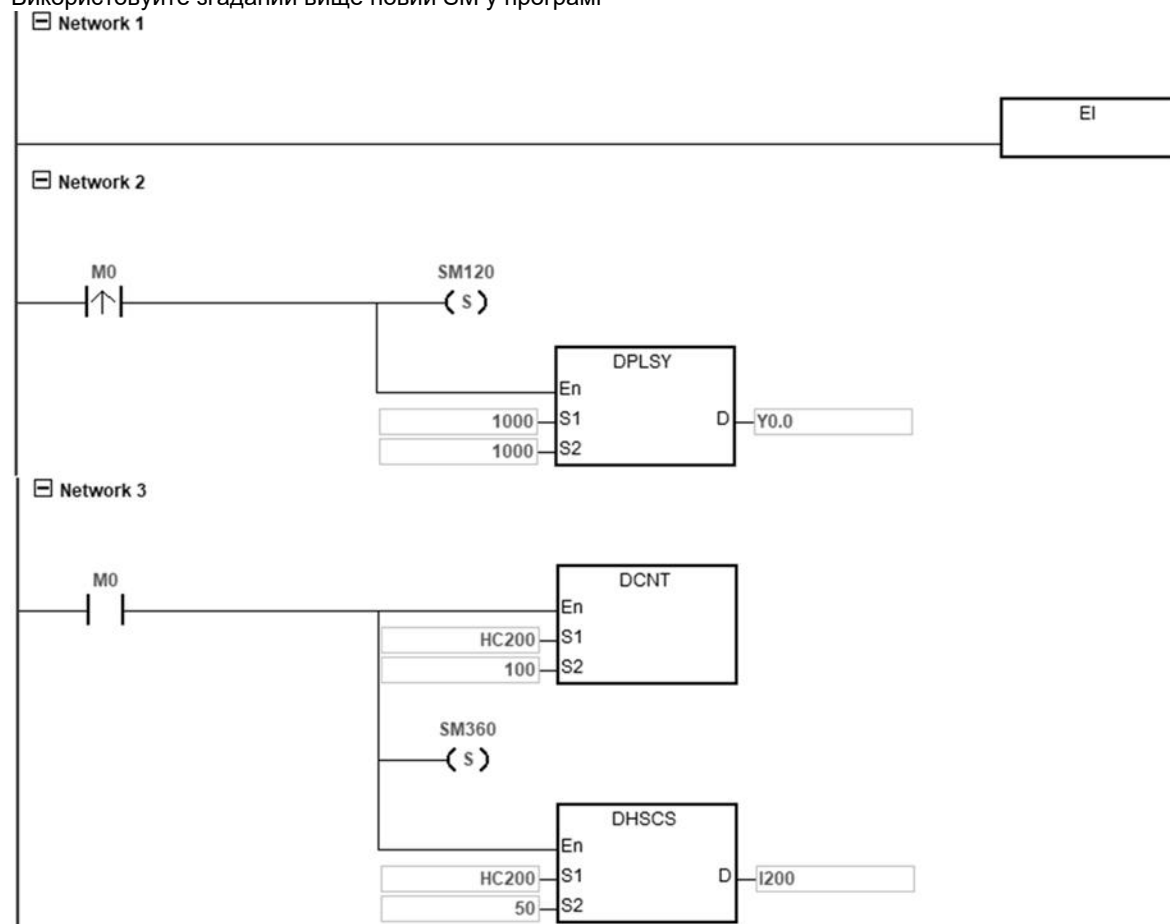
Коли спрацьовує високошвидкісне переривання порівняння і будь-який із SM360-391 встановлено на ON, вихід може бути розпочато негайно, щоб вивести будь-який із збережених параметрів DPLSY SM120-151. Автоматичне скидання керування після завершення виведення.

Нижче наведено відповідні номери високошвидкісних переривань порівняння для щойно доданих SM.

Номер переривання високошвидкісного порівняння	Для попереднього збереження параметрів для виведення DPLSY	Щоб увімкнути запуск DPLSY у перериванні для вихідних імпульсів
I200	SM120	SM360
I201	SM121	SM361
I202	SM122	SM362
I203	SM123	SM363
I210	SM124	SM364
I211	SM125	SM365
I212	SM126	SM366
I213	SM127	SM367
I220	SM128	SM368
I221	SM129	SM369
I222	SM130	SM370
I223	SM131	SM371
I230	SM132	SM372
I231	SM133	SM373
I232	SM134	SM374
I233	SM135	SM375
I240	SM136	SM376
I241	SM137	SM377

I242	SM138	SM378
I243	SM139	SM379
I250	SM140	SM380
I251	SM141	SM381
I252	SM142	SM382
I253	SM143	SM383
I260	SM144	SM384
I261	SM145	SM385
I262	SM146	SM386
I263	SM147	SM387
I264	SM148	SM388
I265	SM149	SM389
I266	SM150	SM390
I267	SM151	SM391

Використовуйте згаданий вище новий SM у програмі



Пояснення

Встановіть M0 на ON, щоб увімкнути SM120, щоб зберегти параметри DPLSY, які потрібно вивести (виведення 1000 імпульсів на частоті 1 кГц) у флеш-пам'ять ПЛК.

Увімкніть високошвидкісний лічильник HC200 і встановіть SM360. Коли підраховане значення в S2 ^{досягає 50,} запускається високошвидкісне переривання порівняння I200. у момент, коли спрацює переривання, Y0.0 починає видавати 1000 імпульсів на частоті 1 кГц.

● Переривання високошвидкісного
☐ Network 1



2

Пояснення

Оскільки виконання високошвидкісного виведення було завершено в POU, немає необхідності писати будь-яку програму в перериванні.

26. Використовуйте компенсацію люфту в CSFO (доступно для FW V1.08.30 або новішої версії)

ПЛК використовує формулу пропозиційного перетворення, щоб перетворити частоту та число імпульсів для входів у те, що підходить для виходів. Від введення до виводу цей процес займає не лише час, але й може спричинити проблеми з люфтом. Коли кількість імпульсів, що проходять через вхід, збільшується, збільшується і люфт.

Для точної роботи використовуйте наступний SM/SR, щоб компенсувати люфти, спричинені тривалою роботою CSFO.

Вихідна вісь, визначена CSFO	Позначте, щоб активувати компенсацію люфту	Збільште частоту для компенсації люфтів (Гц)
Y0,0 / Y0,1	SM1700	1700 SR
Y0,2 / Y0,3	SM1701	SR1702
Y0,4 / Y0,5	SM1702	SR1704
Y0,6 / Y0,7	SM1703	SR1706
Y0,8 / Y0,9	SM1704	SR1708
Y0,10 / Y0,11	SM1705	SR1710

● **SM1700~1705: прапор для активації компенсації люфту**

ON: Щоб збільшити фактичну вихідну частоту та кількість вихідних імпульсів, щоб зменшити відстань між положення входу і виходу. Компенсація припиняється автоматично, коли положення входу та виходу збігаються. Ви також можете встановити цей прапорець у положення OFF, якщо позиції входу та виводу не повинні бути однаковими.

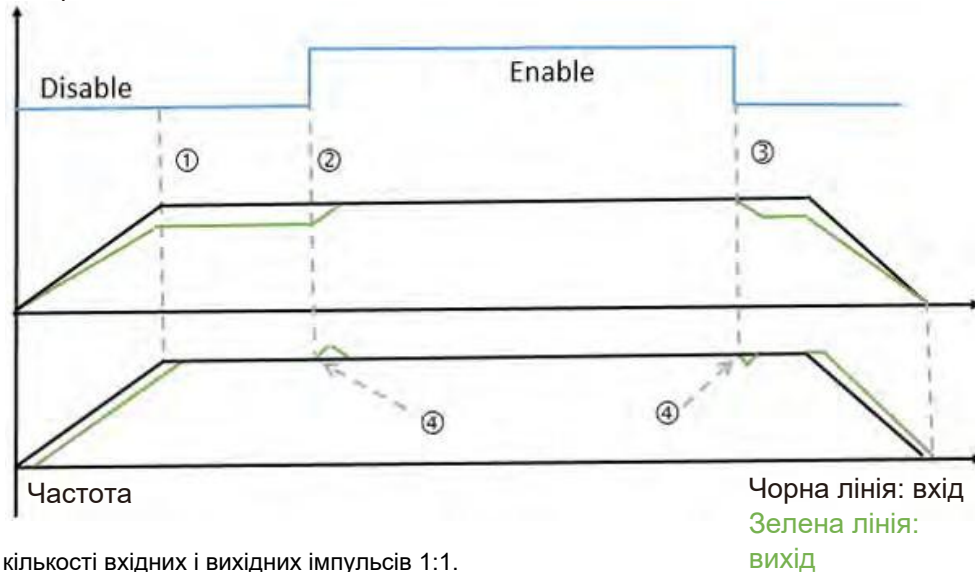
OFF.: Для зменшення фактичної вихідної частоти та кількості вихідних імпульсів для припинення компенсації до тих пір, поки вихідний сигнал не відстане до початкового положення із затримкою.

ПРИМІТКА. Не забудьте закрити прапорець, щоб дезактивувати компенсацію, перш ніж зупинити виконання CSFO. Якщо ні, ви можете виявити, що компенсований вихід продовжує працювати навіть після припинення виконання CSFO.

● **SR1700~1710: Збільште частоту, щоб компенсувати люфти**

Ви можете визначити частоту в діапазоні від 50 Гц до 10 КГц (одиниця: 50 Гц) для компенсації люфтів. Спочатку рекомендується спробувати більш низьку частоту. Інакше різкі та швидкі рухи можуть вплинути на роботу механізму. Компенсація припиняється автоматично, коли положення входу та виходу збігаються. Одиниця частоти 50 Гц; таким чином, якщо використовується 80 Гц, вона розглядатиметься як 50 Гц, а 120 Гц розглядатиметься як 100 Гц.

Ілюстрація використання компенсації люфту в CSFO.
Кількість імпульсів



Співвідношення кількості вхідних і вихідних імпульсів 1:1.

Часи відображаються як ①, ②, ③, ④. Дивіться пояснення нижче.

- ① : Оскільки кількість імпульсів, що проходять через вхід, велика, а вхідна частота висока, вихід не може наздогнати його, відстань між вхід і вихід зростає все більше і більше. Компенсація люфту ще не почалася.
- ② : увімкніть прапорець, щоб активувати компенсацію люфту. Збільште кількість вихідних імпульсів відповідно до частоти входу та компенсації, поки положення входу та виходу не стануть однаковими.
- ③ : Вимкніть прапорець, щоб дезактивувати компенсацію люфту. Зменшіть фактичну вихідну частоту та кількість вихідних імпульсів, щоб припинити компенсацію, доки вихід не відстане до початкового положення із затримкою.
- ④ : Коли прапорець для активації компенсації люфту ввімкнено або вимкнено, ПЛК обчислює та збільшує або зменшує вихідну частоту для покращення плавності компенсації.

Наведений тут приклад показує, що якщо поточна вихідна частота становить 1 кГц, а компенсована частота – 100 Гц, фактична вихідна частота має становити приблизно 1,1 кГц.

2.2.17 Індексні регістри (E)

Індексний регістр - це 16-розрядний регістр даних. Він подібний до загального реєстру тим, що дані з нього можна читати та записувати в нього; однак він в основному використовується як індексний регістр. Діапазон індексних регістрів E0 – E9. Зверніться до розділу 4.4 у

Посібник з програмування серії AS300 для отримання додаткової інформації про використання індексних регістрів.

2.2.18 Регістри файлів (FR)

- ПЛК серії AS300 надає вам файлові регістри для зберігання більшої кількості параметрів.
- Ви можете редагувати, завантажувати та звантажувати параметри в реєстрах файлів через ISPSOft.
- Ви можете читати значення в регістрах файлів під час роботи ПЛК. Зверніться до інструкції MEMW (API 2303) у Посібнику з програмування серії AS300 для отримання додаткової інформації про те, як записувати в регістри файлів.

ПАМ'ЯТКА

Розділ 3 Таблиці інструкцій

Зміст

3.1 Типи інструкцій	3-2
3.1.1 Основні інструкції	3-2
3.1.2 Прикладні інструкції	3-2
3.2 Розуміння таблиць інструкцій	3-3
3.2.1 Основні інструкції	3-3
3.2.2 Застосовувані інструкції (відсортовані за номерами)	3-4
3.2.3 Застосовувані вказівки (відсортовані за алфавітом)	3-5
3.2.4 Таблиці пристроїв	3-6
3.3 Списки основних інструкцій	3-8
3.4 Списки прикладних інструкцій	3-11
3.4.1 Застосовувані інструкції (відсортовані за номерами API).....	3-11
3.4.2 Застосовувані інструкції (відсортовані за алфавітом)	3-48

3.1 Види інструкцій

Інструкції, що використовуються в ПЛК серії AS300, включають основні і прикладні інструкції.

3.1.1 Основні інструкції

Класифікація	опис
Контактні інструкції	Такі інструкції, як завантаження контакту, послідовне з'єднання контакту та паралельне з'єднання контакту
Вихідні інструкції	Розрядність виведення пристрою; імпульсний вихід
Інструкція головного контролю	Налаштування та скидання головного керування
Контактні інструкції щодо виявлення висхідного/спадаючого фронту	Запуск інструкцій, які завантажують контакт, з'єднують контакти послідовно та з'єднують контакти паралельно
Інструкції диференціального виходу по наростаючому/спадаючому фронту	Розрядний вихід диференціального пристрою
Інші інструкції	Інші інструкції

3.1.2 Прикладні інструкції

API	Класифікація	опис
0000-0083	Інструкція з порівняння	Такі порівняння, як =, <>, >, >=, <, <=
0100-0118	Арифметичні вказівки	Використання двійкових чисел або десяткових чисел у двійковому коді для додавання, віднімання, множення або ділення
0200-0217	Інструкції з перетворення даних	Перетворення десяткового числа з двійковим кодом у двійкове число та перетворення двійкового числа на десяткове число з двійковим кодом
0300-0310	Інструкції з передачі даних	Передайте вказані дані
0400-0402	Інструкції зі стрибка	Контроль переходить до іншої частини програми
0500-0504	Інструкція по виконанню програми	Включення або відключення переривання
0600-0602	Інструкції з оновлення вводу/виводу	Оновлення введення-виведення
0700-0711	Різні інструкції	Такі вказівки, як ті, що стосуються лічильників, таймерів режиму навчання та спеціальних таймерів
0800-0817	Логічні інструкції	Логічні операції, такі як логічне додавання та логічне множення
0900-0904	Інструкції з ротації	Обертання/зміщення вказаних даних
1000-1018	Основні інструкції	Інструкції таймера та інструкції лічильника
1100-1115	Інструкції по зміні	Зсув зазначених даних
1200-1229	Інструкції з обробки даних	16-бітна обробка даних, наприклад декодування та кодування
1300-1302	Інструкція по створенню конструкції	Вкладені цикли
1400-1430 роки	Інструкції модуля	Читання даних із певного модуля та запис даних у певний модуль
1500-1517 роки	Інструкції з числами з плаваючою комою	Операції з числами з плаваючою комою
1600-1608 роки	Інструкції годинника реального часу	Читання і письмо, додавання, віднімання та порівняння часу
1700-1704 роки	Периферійні інструкції	Точки введення/виведення, підключені до периферійного пристрою
1806-1821 роки	Інструкції щодо спілкування	Контроль периферійного зв'язку
1900-1906 роки	Інші інструкції	Сторожовий таймер, таймер програмної затримки, ширина імпульсу та індексні регістри

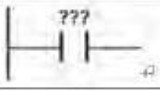
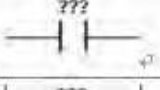
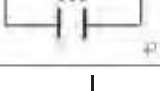
API	Класифікація	опис
2100-2119	Інструкції з обробки рядка	Перетворення між двійковими чи двійково закодованими десятковими числами та кодами ASCII; перетворення між двійковими числами та рядками; перетворення між числами з плаваючою комою та рядками; обробка рядків
2200-2216	Інструкції по Ethernet	Управління обміном даними Ethernet
2300-2305	Карта пам'яті / Інструкції щодо реєстрації файлів	Зчитування даних з карти пам'яті / реєстру файлів і запис даних на карту пам'яті / реєстр файлів
2400-2401	Інструкції з контролю завдань	Контрольні завдання в програмі
2500-2502	Інструкції щодо послідовних функціональних діаграм (SFC).	Керування інструкціями SFC
2700-2724	Інструкції щодо швидкого виведення	Інструкції з високошвидкісного виведення та контролю положення
2800-2823	Спеціальні інструкції щодо зв'язку Delta CANopen / EtherCAT	Інструкції зі зв'язку CANopen / EtherCAT спеціально для пристроїв Delta

3.2 Розуміння таблиць інструкцій

У цьому розділі описано формат таблиці, який використовується в цьому розділі та в решті цього посібника для опису кожної інструкції. Формат різний в залежності від типу навчання: базове або прикладне.

3.2.1 Основні інструкції

У цьому розділі описано формат таблиці, який використовується в цьому розділі для опису основних інструкцій у розділі 3.3.

Instruction code	Symbol	Function	Operand
LD		Loading the contact A/Connecting the contact A in series/Connecting the contact A in parallel	DX X Y M S T C HC D L SM PR
AND			
OR			
↓	↓	↓	↓
①	②	③	④

- опис:
- ① : назва інструкції
 - ② : символ інструкції на драбиновій діаграмі в ISPSoft
 - ③ : функція інструкції
 - ④ : Операнди, які підтримуються інструкцією

3.2.2 Застосовувані інструкції (відсортовані за номерами)

У цьому розділі описано формат таблиці, який у цій главі використовується для опису прикладних інструкцій (відсортованих за номером API) у Розділі 3.4.1.

API	Instruction code		Pulse Instruction	Symbol		Function
	16-bit	32-bit				
0000	LD=	DLD=	—			Comparing the contact types ON: S1 = S2 OFF: S1≠S2
0001	LD<>	DLD<>	—			Comparing the contact types ON: S1≠S2 OFF: S1 = S2
0002	LD>	DLD>	—			Comparing the contact types ON: S1 > S2 OFF: S1≤S2

↓

↓

↓

↓

↓

↓

①

②

③

④

⑤

⑥

API	Instruction code		Pulse Instruction	Symbol		Function
	32-bit	64-bit				
0018	FLD=	DFLD=	—			Comparing the floating-point number contact types ON: S1 = S2 OFF: S1≠S2

↓

⑦

опис:

- ① : номер застосованої інструкції
- ② : назва інструкції
- ③ : Якщо 16-бітну інструкцію можна використовувати як 32-бітну інструкцію, додайте D перед 16-бітною інструкцією, щоб сформувати 32-бітну інструкцію.
- ④ : ✓ вказує на те, що ви можете використовувати інструкцію як імпульсну інструкцію, тоді як — вказує на те, що її не можна використовувати як пульсова інструкція. Для інструкцій щодо пульсу додайте літеру P позаду інструкції.
- ⑤ : символ інструкції на драбиновій діаграмі в ISPSoft
- ⑥ : функція інструкції
- ⑦ : для інструкцій з плаваючою комою одинарної точності (32-розрядні) в інструкції з'являється F.

3.2.3 Застосовувані інструкції (відсортовані за алфавітом)

У цьому розділі описано формат таблиці, який у цій главі використовується для опису прикладних інструкцій (відсортованих за алфавітом) у Розділі 3.4.2.

Classification	API	Instruction code		Pulse instruction	Function
		16-bit	32-bit		
E	0501	EI	—	—	Enabling the interrupt
	2208	EIPRW	—	—	Reading and writing via Ethernet/IP connection
	0503	EIX	—	—	Disabling the specific interrupt
	1203	ENCO	—	✓	Encoder
	1905	EPOP	—	✓	Reading the data into the index registers
	1904	EPUSH	—	✓	Storing the contents of the index registers

↓

↓

↓

↓

↓

↓

①

②

③

④

⑤

⑥

опис:

① : Ініціал назви інструкції

② : номер застосованої інструкції

③ ~ ④ : назви інструкцій

Якщо 16-бітну інструкцію можна використовувати як 32-бітну інструкцію, додайте D перед 16-бітною інструкцією, щоб сформувати 32-бітну інструкцію.

⑤ : ✓ вказує на те, що ви можете використовувати інструкцію як імпульсну інструкцію, тоді як — вказує на те, що її не можна використовувати як імпульсну інструкцію. Для імпульсної інструкції додайте P у кінці інструкції.

⑥ : функція інструкції

3.2.4 Таблиці пристроїв

У цьому розділі описано формат таблиці, який використовується в решті цього посібника для опису кожної інструкції.

①

②

③

④

API	Instruction code				Operand								Function					
0100	D		+	P	$S_1 \cdot S_2 \cdot D$								Addition of binary numbers					

⑤

Device	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		○	○	○	○		
S ₂	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

⑥

Data type	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●		●	●	●				●	●	
S ₂		●	●		●	●	●				●	●	
D		●	●		●	●	●				●	●	

Pulse instruction	16-bit instruction	32-bit instruction
AS	AS	AS

⑧

Symbol:

+		+P	
En	D	En	D
S1		S1	
S2		S2	

S₁ : Augend

S₂ : Addend

D : Sum

опис:

① : номер застосованої інструкції

② : назва інструкції

Якщо 16-бітну інструкцію можна використовувати як 32-бітну інструкцію, додайте D перед 16-бітною інструкцією, щоб сформувати 32-бітну інструкцію.

③ : операнд

④ : функція інструкцій

⑤ : пристрої, які підтримуються операндом

1. Десяткові форми позначені «K», але ви вводите їх безпосередньо в ISPSOft. Наприклад, в ISPSOft введіть десяткове число 30.
2. Шістнадцяткові форми позначаються 16#. Наприклад, десяткове число 30 представлено 16#1E у шістнадцятковій системі.
3. Числа з плаваючою комою позначаються «F» або «DF», але в ISPSOft вони представлені десятковими комами. Наприклад, число з плаваючою комою F500 представлено як 500.0 в ISPSOft.
4. Рядки позначаються «\$», але в ISPSOft вони представлені лапками (« »). Наприклад, рядок 1234 представлений як «1234» в ISPSOft.

5. ○: порожнє коло

Вказує на те, що пристрій не можна змінити за допомогою індексного регістру.

6. ●: суцільне коло

Вказує, що пристрій можна змінювати за допомогою індексного регістру.

⑥ : одиниця операнда

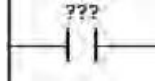
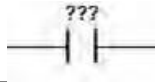
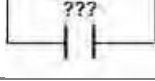
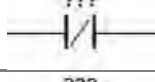
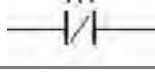
⑦ : Формат інструкції

Вказує, чи можна використовувати інструкцію як імпульсну, 16-бітну або 32-бітну інструкцію.

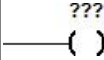
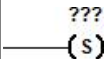
⑧ : Символ інструкції на схемі в ISPSOft.

3.3 Перелік основних інструкцій

● Контактні інструкції

Код інструкції	символ	функція	Операнд
LD		Навантажувальний контакт А/Послідовне підключення контакту А/Паралельне підключення контакту А	DX, X, Y, M, SM, S, T, C, HC, D
AND			
OR			
LDI		Навантажувальний контакт В/Послідовне підключення контакту В/Послідовне підключення контакту В	DX, X, Y, M, SM, S, T, C, HC, D
ANI			
ORI			

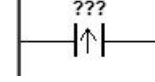
● Вихідні інструкції

Код інструкції	символ	функція	Умова виконання	Операнд
OUT		Приведення в рух котушки		DY, Y, M, SM, S, T, C, HC, D
SET		Тримайте пристрій увімкненим		DY, Y, M, SM, S, T, C, HC, D

● Керівні інструкції майстра

Код інструкції	символ	функція	Операнд
MC		Налаштування головного керування	N
MCR		Скидання головного керування	N

● Контактні інструкції щодо виявлення висхідного/спадаючого фронту

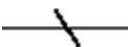
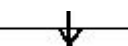
Код інструкції	символ	функція	Операнд	Код інструкції
LDP		Запуск виявлення переднього фронту/Connecti		DX, X, Y, M, SM, S, T, C, HC, D

Код інструкції	символ	функція	Операнд	Код інструкції
ANDP		послідовне підключення детектора наростаючого фронту/паралельне підключення детектора наростаючого фронту		X, Y, M, SM, S, T, C, HC, D
ORP				
PED				
APED				
OPED				
LDF		Запуск детекції спадаючого фронту/Послідовне підключення детектора спадаючого фронту/Паралельне підключення детектора спадаючого фронту		DX, X, Y, M, SM, S, T, C, HC, D
ANDF				
ORF				
NED				
ANED				
ONED				

● Інструкції диференціального виходу по наростаючому/спадаючому фронту

Код інструкції	символ	функція	Умова виконання	Операнд
PLS		Диференціальний вихід по висхідному фронту		Y, M, SM, S
PLF		Диференціальний вихід на спаді		Y, M, SM, S

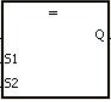
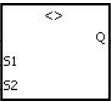
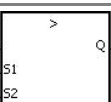
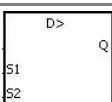
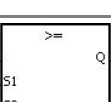
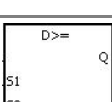
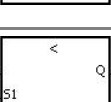
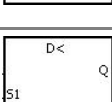
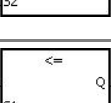
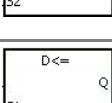
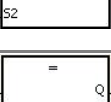
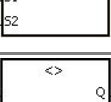
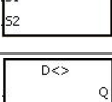
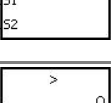
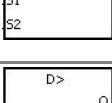
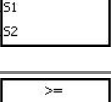
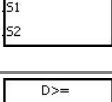
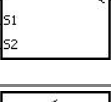
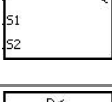
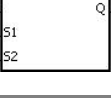
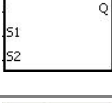
● Інші інструкції

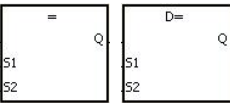
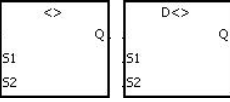
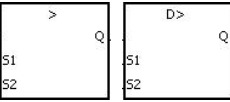
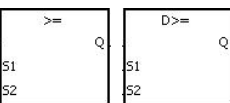
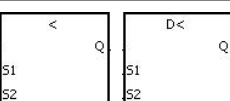
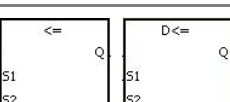
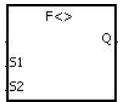
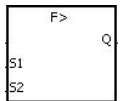
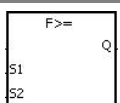
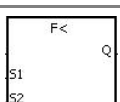
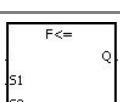
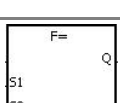
Код інструкції	символ	функція	Операнд
INV		Інвертування результату логічної операції	—
NP		Ланцюг спрацьовує по висхідному фронту.	—
PN		У ланцюзі запускається спадаючий фронт.	—
FB_NP		Ланцюг спрацьовує по висхідному фронту.	Y, M, S, D
FB_PN		У ланцюзі запускається спадаючий фронт.	Y, M, S, D

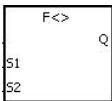
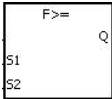
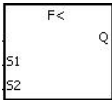
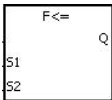
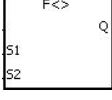
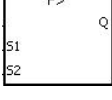
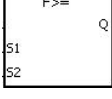
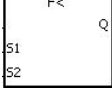
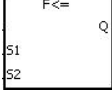
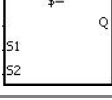
3.4 Переліки прикладних інструкцій

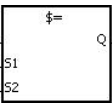
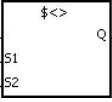
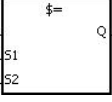
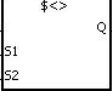
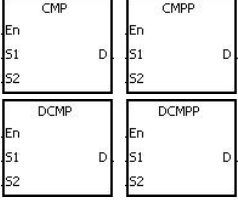
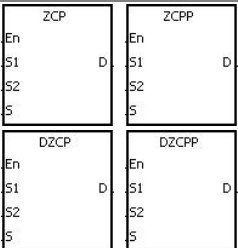
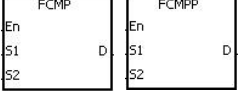
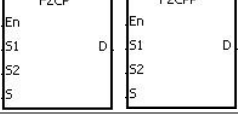
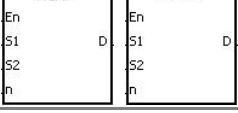
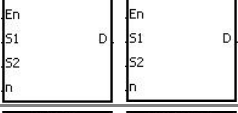
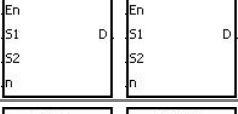
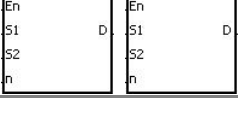
3.4.1 Застосовувані інструкції (відсортовано за номером API)

● Інструкція з порівняння

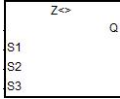
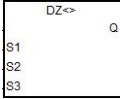
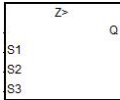
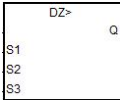
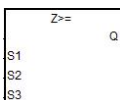
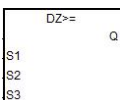
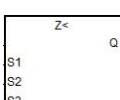
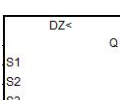
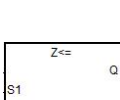
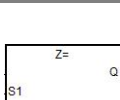
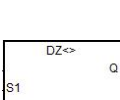
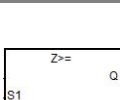
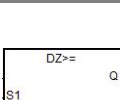
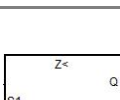
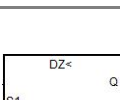
API	Код інструкції		Інструкція по пульсу	символ		функція
	16-бітний	32-розрядний				
0000	LD=	DLD=	—			Порівняння значень ON: $S_1 = S_2$ OFF.: $S_1 \neq S_2$
0001	LD<>	DLD<>	—			Порівняння значень ON: $S_1 \neq S_2$ OFF.: $S_1 = S_2$
0002	LD>	DLD>	—			Порівняння значень ON: $S_1 > S_2$ OFF.: $S_1 \leq S_2$
0003	LD>=	DLD>=	—			Порівняння значень ON: $S_1 \geq S_2$ OFF.: $S_1 < S_2$
0004	LD<	DLD<	—			Порівняння значень ON: $S_1 < S_2$ OFF.: $S_1 \geq S_2$
0005	LD<=	DLD<=	—			Порівняння значень ON: $S_1 \leq S_2$ OFF.: $S_1 > S_2$
0006	AND=	DAND=	—			Порівняння значень ON: $S_1 = S_2$ OFF.: $S_1 \neq S_2$
0007	AND<>	DAND<>	—			Порівняння значень ON: $S_1 \neq S_2$ OFF.: $S_1 = S_2$
0008	AND>	DAND>	—			Порівняння значень ON: $S_1 > S_2$ OFF.: $S_1 \leq S_2$
0009	AND>=	DAND>=	—			Порівняння значень ON: $S_1 \geq S_2$ OFF.: $S_1 < S_2$
0010	AND<	DAND<	—			Порівняння значень ON: $S_1 < S_2$ OFF.: $S_1 \geq S_2$
0011	AND<=	DAND<=	—			Порівняння значень ON: $S_1 \leq S_2$ OFF.: $S_1 > S_2$

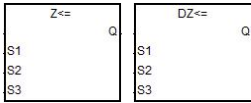
API	Код інструкції		Інструкція по пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
0012	OR=	DOR=	—		Порівняння значень ON: $S_1 = S_2$ OFF.: $S_1 \neq S_2$
0013	OR<>	DOR<>	—		Порівняння значень ON: $S_1 \neq S_2$ OFF.: $S_1 = S_2$
0014	OR>	DOR>	—		Порівняння значень ON: $S_1 > S_2$ OFF.: $S_1 \leq S_2$
0015	OR>=	DOR>=	—		Порівняння значень ON: $S_1 \geq S_2$ OFF.: $S_1 < S_2$
0016	OR<	DOR<	—		Порівняння значень ON: $S_1 < S_2$ OFF.: $S_1 \geq S_2$
0017	OR<=	DOR<=	—		Порівняння значень ON: $S_1 \leq S_2$ OFF.: $S_1 > S_2$
0018	—	FLD=	—		Порівняння чисел з плаваючою комою ON: $S_1 = S_2$ OFF.: $S_1 \neq S_2$
0019	—	FLD<>	—		Порівняння чисел з плаваючою комою ON: $S_1 \neq S_2$ OFF.: $S_1 = S_2$
0020	—	FLD>	—		Порівняння чисел з плаваючою комою ON: $S_1 > S_2$ OFF.: $S_1 \leq S_2$
0021	—	FLD>=	—		Порівняння чисел з плаваючою комою ON: $S_1 \geq S_2$ OFF.: $S_1 < S_2$
0022	—	FLD<	—		Порівняння чисел з плаваючою комою ON: $S_1 < S_2$ OFF.: $S_1 \geq S_2$
0023	—	FLD<=	—		Порівняння чисел з плаваючою комою ON: $S_1 \leq S_2$ OFF.: $S_1 > S_2$
0024	—	FAND=	—		Порівняння чисел з плаваючою комою ON: $S_1 = S_2$ OFF.: $S_1 \neq S_2$

API	Код інструкції		Інструкція по пульсу	символ	функція
	16-бітний	32-розрядний			
0025	—	FAND<>	—		Порівняння чисел з плаваючою комою ON: $S_1 \neq S_2$ OFF.: $S_1 = S_2$
0026	—	FAND>	—		Порівняння чисел з плаваючою комою ON: $S_1 > S_2$ OFF.: $S_1 \leq S_2$
0027	—	FAND>=	—		Порівняння чисел з плаваючою комою ON: $S_1 \geq S_2$ OFF.: $S_1 < S_2$
0028	—	FAND<	—		Порівняння чисел з плаваючою комою ON: $S_1 < S_2$ OFF.: $S_1 \geq S_2$
0029	—	FAND<=	—		Порівняння чисел з плаваючою комою ON: $S_1 \leq S_2$ OFF.: $S_1 > S_2$
0030	—	FOR=	—		Порівняння чисел з плаваючою комою ON: $S_1 = S_2$ OFF.: $S_1 \neq S_2$
0031	—	FOR<>	—		Порівняння чисел з плаваючою комою ON: $S_1 \neq S_2$ OFF.: $S_1 = S_2$
0032	—	FOR>	—		Порівняння чисел з плаваючою комою ON: $S_1 > S_2$ OFF.: $S_1 \leq S_2$
0033	—	FOR>=	—		Порівняння чисел з плаваючою комою ON: $S_1 \geq S_2$ OFF.: $S_1 < S_2$
0034	—	FOR<	—		Порівняння чисел з плаваючою комою ON: $S_1 < S_2$ OFF.: $S_1 \geq S_2$
0035	—	FOR<=	—		Порівняння чисел з плаваючою комою ON: $S_1 \leq S_2$ OFF.: $S_1 > S_2$
0036	LD\$=	—	—		Порівняння рядків ON: $S_1 = S_2$ ON: $S_1 \neq S_2$
0037	LD\$<>	—	—		Порівняння рядків ON: $S_1 \neq S_2$ OFF.: $S_1 = S_2$

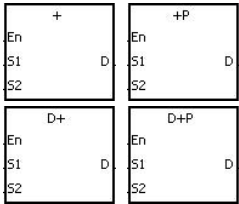
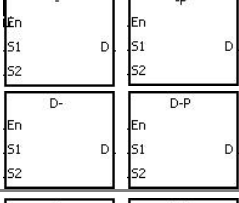
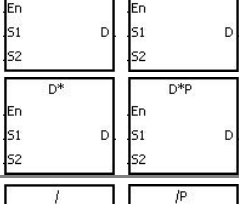
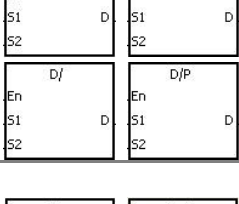
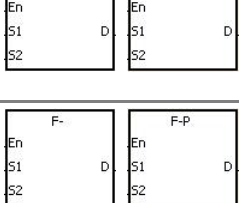
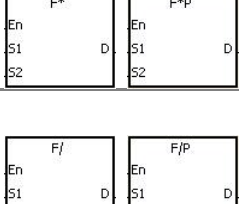
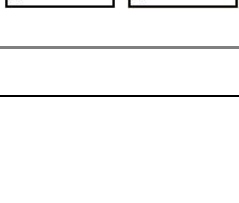
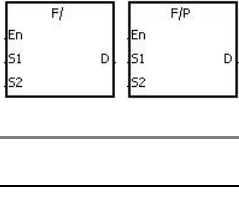
API	Код інструкції		Інструкція по пульсу	символ	функція
	16-бітний	32-розрядний			
0042	AND\$=	—	—		Порівняння рядків ON: $S_1 = S_2$ OFF: $S_1 \neq S_2$
0043	AND\$<>	—	—		Порівняння рядків ON: $S_1 \neq S_2$ OFF: $S_1 = S_2$
0048	OR \$=	—	—		Порівняння рядків ON: $S_1 = S_2$ OFF: $S_1 \neq S_2$
0049	OR\$<>	—	—		Порівняння рядків ON: $S_1 \neq S_2$ OFF: $S_1 = S_2$
0054	CMP	DCMP	✓		Порівняння значень
0055	ZCP	DZCP	✓		Порівняння зон
0056	—	FCMP	✓		Порівняння чисел з плаваючою комою
0057	—	FZCP	✓		Порівняння зон із плаваючою комою
0058	MCMP	—	✓		Матричне порівняння
0059	CMPT=	—	✓		Порівняння таблиць ПО: =
0060	CMPT<>	—	✓		Порівняння таблиць ПО: ≠
0061	CMPT>	—	✓		Порівняння таблиць ПО: >

API	Код інструкції		Інструкція по пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
0062	CMPT>=	—	✓		Порівняння таблиць ON: > =
0063	CMPT<	—	✓		Порівняння таблиць ПО: <
0064	CMPT<=	—	✓		Порівняння таблиць ON: < =
0065	CHKADR	—	—		Перевірка адреси типу контакту покажчика регістру
0066	LDZ=	DLDZ=	—		Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 = S_3 $ OFF: $ S_1 - S_2 \neq S_3 $
0067	LDZ<>	DLDZ<>	—		Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \neq S_3 $ OFF: $ S_1 - S_2 = S_3 $
0068	LDZ>	DLDZ>	—		Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 > S_3 $ OFF: $ S_1 - S_2 \leq S_3 $
0069	LDZ>=	DLDZ>=	—		Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \geq S_3 $ OFF: $ S_1 - S_2 < S_3 $
0070	LDZ<	DLDZ<	—		Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 < S_3 $ OFF: $ S_1 - S_2 \geq S_3 $
0071	LDZ<=	DLDZ<=	—		Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \leq S_3 $ OFF: $ S_1 - S_2 > S_3 $
0072	ANDZ=	DANDZ=	—		Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 = S_3 $ OFF: $ S_1 - S_2 \neq S_3 $

API	Код інструкції		Інструкція по пульсу	символ	функція
	16-бітний	32-розрядний			
0073	ANDZ<>	DANDZ<>	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \neq S_3 $ OFF: $ S_1 - S_2 = S_3 $
0074	ANDZ>	DANDZ>	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 > S_3 $ OFF: $ S_1 - S_2 \leq S_3 $
0075	ANDZ>=	DANDZ>=	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \geq S_3 $ OFF: $ S_1 - S_2 < S_3 $
0076	ANDZ<	DANDZ<	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 < S_3 $ OFF: $ S_1 - S_2 \geq S_3 $
0077	ANDZ<=	DANDZ<=	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \leq S_3 $ OFF: $ S_1 - S_2 > S_3 $
0078	ORZ=	DORZ=	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 = S_3 $ OFF: $ S_1 - S_2 \neq S_3 $
0079	ORZ<>	DORZ<>	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \neq S_3 $ OFF: $ S_1 - S_2 = S_3 $
0080	OP3>	DORZ>	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 > S_3 $ OFF: $ S_1 - S_2 \leq S_3 $
0081	OP3>=	DORZ>=	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \geq S_3 $ OFF: $ S_1 - S_2 < S_3 $
0082	OP3<	DORZ<	—	 	Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 < S_3 $ OFF: $ S_1 - S_2 \geq S_3 $

API	Код інструкції		Інструкція по пульсу	символ	функція
	16-бітний	32-розрядний			
0083	ORZ<=	DORZ<=	—		Порівняння абсолютного результату контактного типу ON: $ S_1 - S_2 \leq S_3 $ OFF: $ S_1 - S_2 > S_3 $

● Арифметичні вказівки

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0100	+	D+	✓		Додавання двійкових чисел $S_1 + S_2 = D$
0101	-	D-	✓		Віднімання двійкових чисел $S_1 - S_2 = D$
0102	*	D*	✓		Множення двійкових чисел $S_1 * S_2 = D$
0103	/	D/	✓		Ділення двійкових чисел $S_1 / S_2 = D$
0104	—	F+	✓		Додавання чисел з плаваючою комою $S_1 + S_2 = D$
0105	—	F-	✓		Віднімання чисел з плаваючою комою $S_1 - S_2 = D$
0106	—	F*	✓		Множення чисел з плаваючою комою $S_1 * S_2 = D$
0107	—	F/	✓		Ділення чисел з плаваючою комою $S_1 / S_2 = D$

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0112	BK+	DBK+	✓	En S1 S2 л En S1 S2 л En S1 S2 л En S1 S2 л	Додавання двійкових чисел у блоки
0113	BK-	DBK-	✓	En S1 S2 л En S1 S2 л En S1 S2 л En S1 S2 л	Віднімання двійкових чисел блоками
0114	\$+	—	✓	En S1 S2 En S1 S2	Зв'язування рядків
0115	INC	DINC	✓	En D En D	Додавання одиниці до двійкового числа
0116	DEC	DDEC	✓	En D En D	Віднімання одиниці з двійкового числа
0117	MUL16	MUL32	✓	En S1 S2 En S1 S2	Множення двійкових чисел для 16-розрядних множення двійкових чисел для 32-розрядних
0118	DIV16	DIV32	✓	En S1 S2 En S1 S2	Ділення двійкових чисел для 16-бітних Ділення двійкових чисел для 32-бітних

● Інструкції з перетворення даних

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0200	BCD	DBCD	✓	En S En S	Перетворення двійкового числа в двійкове десятикове число
0201	BIN	DBIN	✓	En S En S	Перетворення десятикового числа з двійковим кодом у двійкове

API	Код інструкції		Інструкція пульсу	СИМВОЛ		функція
	16-бітний	32-розрядний				
0202	FLT	DFLT	✓	En S D En S D	En S D En S D	Перетворення двійкового цілого числа в двійкове число з плаваючою комою
0204	INT	DINT	✓	En S D En S D	En S D En S D	Перетворення 32-розрядного числа з плаваючою комою в двійкове ціле число
0206	MMOV	—	✓	En S D	En S D	Перетворення 16-бітного значення на 32-бітове
0207	RMOV	—	✓	En S D	En S D	Перетворення 32-бітного значення на 16-бітне
0208	GRY	DGRY	✓	En S D En S D	En S D En S D	Перетворення двійкового числа в код Грея
0209	GBIN	DGBIN	✓	En S D En S D	En S D En S D	Перетворення коду Грея в двійкове число
0210	NEG	DNEG	✓	En S D En S D	En S D En S D	Двоє доповнення до числа
0211	—	FNEG	✓	En S D	En S D	Зміна знака 32-розрядного числа з плаваючою комою
0212	—	FBCD	✓	En S D	En S D	Перетворення двійкового числа з плаваючою комою в десяткове число з плаваючою комою
0213	—	FBIN	✓	En S D	En S D	Перетворення десяткового числа з плаваючою комою у двійкове число з плаваючою комою
0214	BKBCD	—	✓	En S D	En S D	Перетворення двійкових чисел у блоках у двійкові десяткові числа у блоках
0215	BKBIN	—	✓	En S D	En S D	Перетворення двійкових чисел у блоках у двійкові десяткові числа у блоках
0216	SCAL	DSCAL	✓	En S1 S2 S3 D En S1 S2 S3 D	En S1 S2 S3 D En S1 S2 S3 D	Пошук масштабованого значення (точка-нахил)

API	Код інструкції		Інструкція пульсу	символ		функція
	16- бітний	32- розрядний				
0217	SCLP	DSCLP	✓	SCLP En S1 S2 D SCLPP En S1 S2 D DSCLP En S1 S2 D DSCLPP En S1 S2 D	Знаходження масштабованого значення (два бали)	
0222	SCLM	DSCLM	✓	SCLM En S1 S2 S3 S4 D SCLMP En S1 S2 S3 S4 D DSCLM En S1 S2 S3 S4 D DSCLMP En S1 S2 S3 S4 D	Багатоточкова операція співвідношення площ	

● Інструкції з передачі даних

API	Код інструкції		Інструкція пульсу	символ		функція
	16- бітний	32- розрядний				
0300	MOV	DMOV	✓	MOV En S D MOVDP En S D DMOV En S D DMOVDP En S D	Передача даних S: Джерело даних D: призначення даних	
0302	\$MOV	—	✓	\$MOV En S D \$MOVDP En S D	Перенесення рядка	
0303	CML	DCML	✓	CML En S D CMLDP En S D DCML En S D DCMLDP En S D	Інвертування даних	
0304	BMOV	DBMOV	✓	BMOV En S D n BMOVDP En S D n DBMOV En S D n DBMOVDP En S D n	Передача всіх даних	
0305	NMOV	DNMOV	✓	NMOV En S D n NMOVDP En S D n DNMOV En S D n DNMOVDP En S D n	Передача даних на кілька пристроїв	
0306	XCH	DXCH	✓	XCH En S1 S2 XCHDP En S1 S2 DXCH En S1 S2 DXCHDP En S1 S2	Обмін даними	

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0307	BXCH	—	✓		Обмін усіма даними
0308	SWAP	DSWAP	✓	 	Обмін старшого байта на молодший
0309	SMOV	—	✓		Перенесення цифр
0310	MOVB	—	✓		Передача кількох бітів

● Інструкції зі стрибка

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0400	CJ	—	✓		Умовний стрибок
0401	JMP	—	—		Безумовний стрибок
0402	GOEND	—	—		Перейти до END

● Інструкція по виконанню програми

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0500	DI	—	—		Відключення переривання
0501	EI	—	—		Увімкнення переривання
0503	EIX	—	—		Відключення конкретного переривання
0504	DIX	—	—		Включення певного переривання

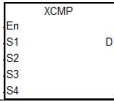
● Інструкції з оновлення вводу/виводу

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0600	REF	—	✓		Оновлення введення-виведення

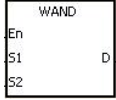
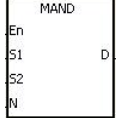
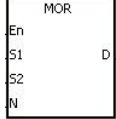
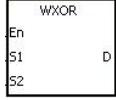
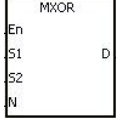
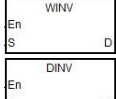
API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0601	—	DHSRF	✓	DHSRF En S S1 DHSRFP En S S1	Оновлення значень швидкісного порівняння
0602	REFF	—	✓	REFF En Xno Length Filter REFFP En Xno Length Filter	Оновлення часу фільтрації введення/виведення

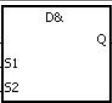
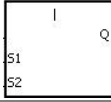
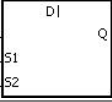
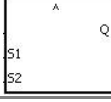
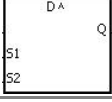
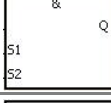
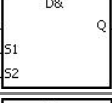
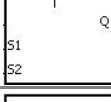
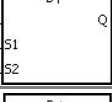
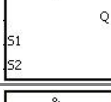
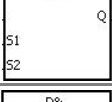
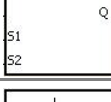
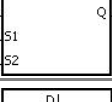
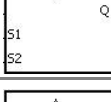
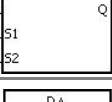
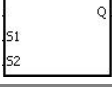
● Різні інструкції

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
0700	ALT	—	✓	ALT En D ALTP En D	Чергування між ON і OFF
0701	TTMR	—	—	TTMR n D	Таймер режиму навчання
0702	STMR	—	—	STMR S m D	Спеціальний таймер
0703	RAMP	DRAMP	—	RAMP S1 S2 N D DRAMP En S1 S2 n D	Сигнал циклічної рамп
0704	MTR	—	—	MTR En S D1 N D2	Матричний вхід
0705	ABSD	DABSD	—	ABSD S1 S2 N D DABSD En S1 S2 N D	Абсолютний барабанний секвенсор
0706	INCD	—	—	INCD S1 S2 N D	Інкрементний барабанний секвенсор
0708	—	DPIDE	—	DPIDE En PID_RUN SV PV PID_MODE PID_MAN MOUT_AUTO CYCLE Kc_Kp Ti_Ki Td_Kd Tf PID_EQ PID_DE PID_DIR ERR_DBW MV_MAX MV_MIN MOUT BIAS LMV MV	ПІД-алгоритм

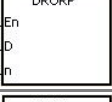
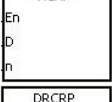
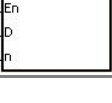
API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
0709	XCMP	—	—		Налаштування для порівняння вхідних даних кількох робочих станцій
0710	YOUT	—	—		Порівняння результатів кількох робочих станцій
0711	—	DSUNRS	✓		Налаштування часу сходу та заходу сонця

● Логічні інструкції

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
0800	WAND	DAND	✓		Логічна операція AND
0801	MAND	—	✓		Матрична операція AND
0802	WOR	DOR	✓		Логічна операція OR
0803	MOR	—	✓		Матрична операція OR
0804	WXOR	DXOR	✓		Логічна операція виключаючого OR
0805	MXOR	—	✓		Матрична операція виняткового OR
0808	WINV	DINV	✓		Логічна зворотна операція INV

API	Код інструкції		Інструкція пульсу	СИМВОЛ		функція
	16- бітний	32- розрядний				
0809	LD&	DLD&	—			$S_1 \& S_2$
0810	LD	DLD	—			$S_1 S_2$
0811	LD^	DLD^	—			$S_1 \wedge S_2$
0812	AND&	DAND&	—			$S_1 \& S_2$
0813	AND	DAND	—			$S_1 S_2$
0814	AND^	DAND^	—			$S_1 \wedge S_2$
0815	OR&	DOR&	—			$S_1 \& S_2$
0816	OR	DOR	—			$S_1 S_2$
0817	OR^	DOR^	—			$S_1 \wedge S_2$

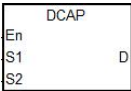
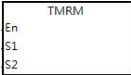
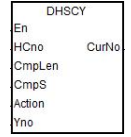
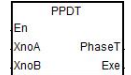
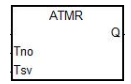
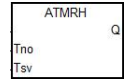
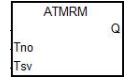
● Інструкції з ротації

API	Код інструкції		Інструкція пульсу	СИМВОЛ		функція
	16- бітний	32- розрядний				
0900	ROR	DROR	✓			Обертання бітів у групі вправо
						
0901	RCR	DRCR	✓			Обертання бітів у групі вправо з прапором перенесення
						

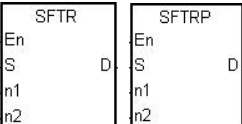
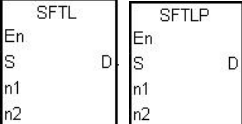
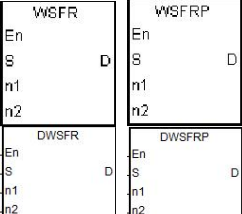
API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
0902	ROL	DROL	✓		Обертання бітів у групі вліво
0903	RCL	DRCL	✓		Обертання бітів у групі вліво з прапором перенесення
0904	MBR	—	✓		Обертання бітів вправо або вліво в матриці

● Таймер і лічильник інструкції

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1000	RST	DRST	—		Скидання контакту на OFF або очищення значення в реєстрі
1001	TMR	—	—		16-бітний таймер (одиниця: 100 мс)
1002	TMRH	—	—		16-бітний таймер (одиниця: 1 мс)
1003	CNT	—	—		16-розрядний лічильник
1004	—	DCNT	—		32-розрядний лічильник (включаючи використання високошвидкісних лічильників)
1005	—	DHSCS	—		Налаштування високошвидкісного порівняння
1006	—	DHSCR	—		Скидання високошвидкісного порівняння
1007	—	DHSZ	—		Швидке порівняння зон введення
1008	—	DSPD	—		Виявлення швидкості
1009	PWD	—	—		Виявлення ширини імпульсу

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1010	—	DCAP	—		Захоплення високошвидкісного значення підрахунку в зовнішньому входному перериванні
1011	TMRM	—	—		16-бітний таймер (одиниця: 10 мс)
1012	IETS	—	✓		Початок вимірювання часу виконання інструкції
1013	IETE	—	✓		Кінець вимірювання часу виконання інструкції
1014	—	DHSCY	—		Циклічний підрахунок, порівняння та виведення
1015	PPDT	—	—		Виявлення різниці в часі між двома фазами
1016	ATMR	—	—		16-бітний контактний таймер (одиниця: 100 мс)
1017	ATMRH	—	—		16-бітний контактний таймер (одиниця: 1 мс)
1018	ATMRM	—	—		16-бітний контактний таймер (одиниця: 10 мс)

● Інструкції по зміні

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1100	SFTR	—	✓		Зсув станів пристроїв вправо
1101	SFTL	—	✓		Зсув станів пристроїв вліво
1102	WSFR	DWSFR	✓		Зсув даних у слові пристрої вправо

API	Код інструкції		Інструкція пульсу	СИМВОЛ		функція
	16- бітний	32- розрядний				
1103	WSFL	DWSFL	✓	WSFL En S n1 n2 D DWSFL En S n1 n2 D	WSFLP En S n1 n2 D DWSFLP En S n1 n2 D	Зсув даних у слові пристрої вліво
1104	SFWR	DSFWR	✓	SFWR En S n D DSFWR En S n D	SFWRP En S n D DSFWRP En S n D	Зміщення даних і запис їх у слово пристрій
1105	SFRD	DSFRD	✓	SFRD En S n D DSFRD En S n D	SFRDP En S n D DSFRDP En S n D	Зсув даних і зчитування їх із пристрою слова
1106	SFPO	DSFPO	✓	SFPO En S D DSFPO En S D	SFPOP En S D DSFPOP En S D	Читання останніх даних зі списку даних
1107	SFDEL	DSFDEL	✓	SFDEL En S n D DSFDEL En S n D	SFDELP En S n D DSFDELP En S n D	Видалення даних зі списку даних
1108	SFINS	DSFINS	✓	SFINS En S n D DSFINS En S n D	SFINSP En S n D DSFINSP En S n D	Вставлення даних у список даних
1109	MBS	—	✓	MBS En S n D	MBS P En S n D	Зсув бітів матриці
1110	SFR	DSFR	✓	SFR En n D DSFR En n D	SFRP En n D DSFRP En n D	Зсув значень розрядів у регістрах на n розрядів вправо
1111	SFL	DSFL	✓	SFL En n D DSFL En n D	SFLP En n D DSFLP En n D	Зсув значень розрядів у регістрах на n розрядів вліво
1112	BSFR	—	✓	BSFR En n D	BSFRP En n D	Зсув станів n-розрядних пристроїв на один біт вправо

API	Код інструкції		Інструкція пульсу	символ		функція
	16-бітний	32-розрядний				
1113	BSFL	—	✓	BSFL En D	BSFLP En D	Зсув станів n -розрядних пристроїв на один біт вліво
1114	NSFR	DNSFR	✓	NSFR En D	NSFRP En D	Зсув n регістрів вправо
				DNSFR En D	DNSFRP En D	
1115	NSFL	DNSFL	✓	NSFL En D	NSFLP En D	Зсув n регістрів вліво
				DNSFL En D	DNSFLP En D	

● Інструкції з обробки даних

API	Код інструкції		Інструкція пульсу	символ		функція
	16-бітний	32-розрядний				
1200	SER	DSER	✓	SER En S1 S2 IN D	DSER En S1 S2 IN D	Пошук даних
				SERP En S1 S2 IN D	DSERP En S1 S2 IN D	
1201	SUM	DSUM	✓	SUM En S D	SUMP En S D	Кількість бітів, стани яких увімкнено
				DSUM En S D	DSUMP En S D	
1202	DECO	—	✓	DECO En S n D	DECOP En S n D	Декодер
1203	ENCO	—	✓	ENCO En S n D	ENCOP En S n D	Кодувальник
1204	SEGD	—	✓	SEGD En S D	SEGDP En S D	Семисегментное декодування
1205	SORT	DSORT	✓	SORT En S m1 m2 n D	DSORT En S m1 m2 n D	Сортування даних
				SORTP En S m1 m2 n D	DSORTP En S m1 m2 n D	
1206	ZRST	—	✓	ZRST En D1 D2	ZRSTP En D1 D2	Скидання зони

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1207	BON	DBON	✓	BON En S n D DBON En S n D	Перевірка стану долота
1208	MEAN	DSEAN	✓	MEAN En S n D DMEAN En S n D	Середній
1209	CCD	—	✓	CCD En S n D CCDP En S n D	Перевірка суми
1210	ABS	DABS	✓	ABS En S n D DABS En S n D	Абсолютне значення
1211	MINV	—	✓	MINV En S n D MINVP En S n D	Інвертування бітів матриці
1212	MBRD	—	✓	MBRD En S n D MBRDP En S n D	Читання біта матриці
1213	MBWR	—	✓	MBWR En S n D MBWRP En S n D	Запис розрядності матриці
1214	MBC	—	✓	MBC En S n D MBCP En S n D	Підрахунок бітів зі значенням 0 або 1
1215	DIS	—	✓	DIS En S n D DISP En S n D	Роз'єднання 16-бітних даних
1216	UNI	—	✓	UNI En S n D UNIP En S n D	Об'єднання 16-бітних даних
1217	WSUM	DWSUM	✓	WSUM En S n D DWSUM En S n D	Отримання суми

API	Код інструкції		Інструкція пульсу	СИМВОЛ		функція
	16-бітний	32-розрядний				
1221	LIMIT	DLIMIT	✓	LIMIT En S1 D S2 S3 DLIMIT En S1 D S2 S3	Обмеження значення в межах	
1222	BAND	DBAND	✓	BAND En S1 D S2 S3 DBAND En S1 D S2 S3	Контроль зони нечутливості	
1223	ZONE	DZONE	✓	ZONE En S1 D S2 S3 DZONE En S1 D S2 S3	Контроль зони	
1224	—	FMEAN	✓	FMEAN En S n D FMEANP En S n D	Середнє значення чисел з плаваючою комою	
1225	—	FSUM	✓	FSUM En S n D FSUMP En S n D	Сума чисел з плаваючою комою	
1226	DTM	—	✓	DTM En S m D n DTMP En S m D n	Передача та переміщення даних	
1228	ZSET	—	✓	ZSET En SBit EBit ZSETP En SBit EBit	Налаштування зони	
1229	—	FSORT	✓	FSORT En src dst group column ref FSORTP En src dst group column ref	Сортування даних у форматі з плаваючою комою	

● Інструкція по створенню конструкції

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1300	FOR	—	—	FOR S	Початок вкладеного циклу
1301	NEXT	—	—	NEXT	Кінець вкладеного циклу
1302	BREAK	—	—	BREAK En D P	Завершення циклу FOR-NEXT

● Інструкції модуля

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1400	FROM	DFROM	✓	FROM FROMP DFROM DFROMP	Зчитування даних з контрольного регістру в спеціальному модулі
1401	TO	DTO	✓	TO TOP DTO DTOP	Запис даних у контрольний регістр у спеціальному модулі
1402	—	DPUCONF	✓	DPUCONF DFUONFP	Налаштування вихідних параметрів керування модулем ПУ
1403	PUSTAT	—	—	PUSTAT	Читання вихідного стану модуля PU
1404	—	DPUPLS	—	DPUPLS	Імпульсний вихід модуля PU (без прискорення)
1405	—	DPUDRI	—	DPUDRI	Вихід відносного положення модуля PU (з прискоренням і уповільненням)

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1406	—	DPUDRA	—	DPUDRA En Module Down Axis Error ATarPosi ErrCode TarSpeed	Вихід абсолютної адресації модуля PU (з прискоренням і уповільненням)
1407	—	DPUZRN	—	DPUZRN En Module Done Axis Error Mode ErrCode TarSpeed JogSpeed	Наведення модуля ПУ
1408	—	DPUJOG	—	DPUJOG En Module Busy Axis Error JogSpeed ErrCode	Руховий вихід модуля PU
1409	—	DPUMPG	—	DPUMPG En Module OPulse Axis OSpeed InMode Error InPulse ErrCode InSpeed Rate	Вихід MPG модуля PU
1410	—	DPUCNT	—	DPUCNT En Module InPulse InMode InSpeed Period Error ZeroS ErrCode	Функція високошвидкісного лічильника модуля PU
1415	—	DLCCAL	—	DLCCAL En Group CPoint Module Done ChNo Adone Trigger Error TPoint ErrCode TWeight	Калібрування каналу модуля LC

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1416	—	DLCWEI	—	DLCWEI En Group Weight Module Status ChNo Error Stable ErrCode ZeroS TareS TareW	Зчитування значення ваги через модуль LC
1417	—	DMPID	—	DMPID En PV CHDUP PV MODULE PV CH PV UPDATE Error PID RUN PV PID MODE PID MAN MODE AUTO AUTO DDW HOLD TLK TLK Kd IT PID EG PID DE HOLD PV CHD DDW ALPHA ZETA MOUT HABS CYCLE	Алгоритм PID для модуля RTD/TC
1418	—	DHCCNT	—	DHCCNT En Module CurCnt ChNo ST Update MT Action AStat Value RefCnt Dir CntStat Error ErrCode	Підраховано значення лічильника модуля HC
1419	—	DHCCAP	—	DHCCAP En Module Capt1 ChNo Cmpl1 TrgSel Cmpl2 Error ErrCode	Зафіксовано значення лічильника модуля HC
1420	HCDO	—	—	HCDO En Module Dostate Update Error Dodata ErrCode	Цифровий вихід модуля HC
1421	—	DHCCMP	—	DHCCMP En Module Match1 ChNo Match2 Update Error Comp1 ErrCode Action1 Yno1 Comp2 Action2 Yno2	Значення лічильника модуля HC у порівнянні для виведення

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1422	—	DHCCMPT	—	DHCCMPT En Module CurNo ChNo Error Update ErrCode CmpLen CompS ActionS YnoS Inos	Значення лічильника модуля НС у груповому порівнянні для виведення
1423	—	DHCMEAS	—	DHCMEAS En Module Freq ChNo RPM Update Error Cnt/Rev ErrCode Smpl Avg	Модуль НС вимірювання частоти та обертів
1424	ADLOG	DADLOG	—	ADLOG En Group Datalog Module CurNo ChNo Cmplt Mode Error Period ErrCode Points Postrig DADLOG En Group Datalog Module CurNo ChNo Cmplt Mode Error Period ErrCode Points Postrig	Запис даних модуля аналогового введення
1425	ADPEAK	DADPEAK	—	ADPEAK En Group MAX Module MIN ChNo Error ErrCode DADPEAK En Group MAX Module MIN ChNo Error ErrCode	Запис пікових даних модуля аналогового введення
1426	IOLINKR	—	—	IOLINKR En Group Done Module Error Port ErrCode Index ReadLen SubIndex DataRead DataType	Зчитайте параметр із пристрою O-Link

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1427	IOLINKW	—	—		Запишіть параметр у пристрій IO-Link
1430	MLPF	—	—		Цифровий фільтр низьких частот для аналогових вхідних сигналів

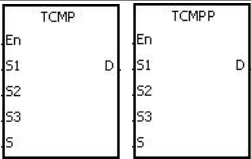
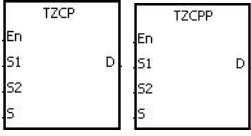
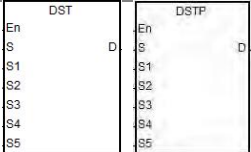
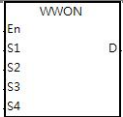
● Інструкції з числами з плаваючою комою

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
1500	—	FSIN	✓		Синус числа з плаваючою комою
1501	—	FCOS	✓		Косинус числа з плаваючою комою
1502	—	FTAN	✓		Тангенс числа з плаваючою комою
1503	—	FASIN	✓		Арксинус числа з плаваючою комою
1504	—	FACOS	✓		Аркосинус числа з плаваючою комою
1505	—	FATAN	✓		Арктангенс числа з плаваючою комою
1506	—	FSINH	✓		Гіперболічний синус числа з плаваючою комою
1507	—	FCOSH	✓		Гіперболічний косинус числа з плаваючою комою
1508	—	FTANH	✓		Гіперболічний тангенс числа з плаваючою комою

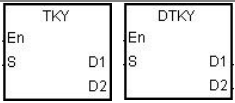
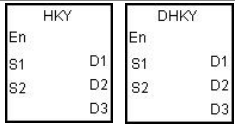
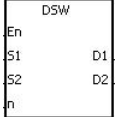
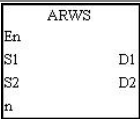
API	Код інструкції		Інструкція пульсу	СИМВОЛ		функція
	16-бітний	32-розрядний				
1509	—	FRAD	✓			Перетворення градусів у радіани
1510	—	FDEG	✓			Перетворення радіан в градуси
1511	SQR	DSQR	✓	 	 	Квадратний корінь з двійкового числа
1512	—	FSQR	✓			Квадратний корінь із числа з плаваючою комою
1513	—	FEXP	✓			Експонента числа з плаваючою комою
1514	—	FLOG	✓			Логарифм числа з плаваючою комою
1515	—	FLN	✓			Натуральний логарифм двійкового числа з плаваючою комою
1516	—	FPOW	✓			Піднесення числа з плаваючою комою до степеня
1517	RAND	—	✓			Випадкове число

● Інструкції годинника реального часу

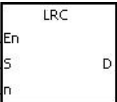
API	Код інструкції		Інструкція пульсу	СИМВОЛ		функція
	16-бітний	32-розрядний				
1600	TRD	—	✓			Читання часу
1601	TWR	—	✓			Написання часу
1602	T+	—	✓			Додавання часу
1603	T-	—	✓			Віднімання часу
1604	HOUR	—	—			Лічильник часу роботи

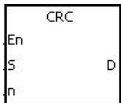
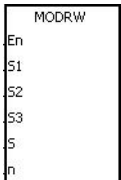
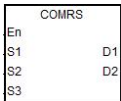
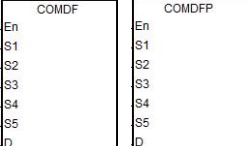
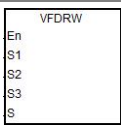
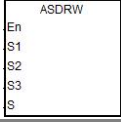
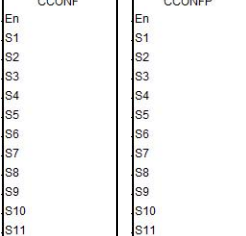
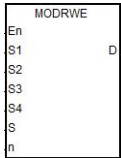
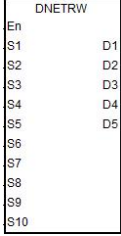
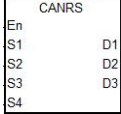
API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
1605	TCMP	—	✓		Порівняння часу
1606	TZCP	—	✓		Порівняння часових поясів
1607	TZCP	—	✓		Літній час
1608	WWON	—	—		Встановлення тижневого робочого часу

● Периферійні інструкції

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
1700	TKY	DTKY	—		Десятиклавішна клавіатура
1701	HKY	DHKY	—		Шістнадцятиклавішна клавіатура
1702	DSW	—	—		DIP-перемикач
1703	ARWS	—	—		Клавіші зі стрілками
1704	SEGL	—	—		Семисегментний дисплей із засувками

● Інструкції щодо спілкування

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
1806	LRC	—	—		Поздовжня перевірка парності

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16- бітний	32- розрядний			
1807	CRC	—	—		Циклічна перевірка надмірності
1808	MODRW	—	—		Читання та запис даних MODBUS
1812	COMRS	—	—		Надсилання та отримання комунікаційних даних
1813	COMDF	—	✓		Встановлення формату зв'язку для послідовного порту зв'язку
1814	VFDRW	—	—		Інструкція по послідовному зв'язку, виключно для приводу змінного струму Delta
1815	ASDRW	—	—		Інструкція по послідовному зв'язку, виключно для сервоприводу Delta
1816	CCONF	—	✓		Налаштування параметрів у таблиці обміну даними для комунікаційного порту
1817	MODRWE	—	—		Читання та запис даних Modbus без використання будь-яких прапорів
1818	DNETRW	—	—		Читання та запис даних зв'язку DeviceNet
1819	CANRS	—	—		Надсилання та отримання користувацького зв'язку CAN

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
1820	DMVSH	—	—		Увімкнення виявлення та зв'язку Delta DMV
1821	DESO	—	—		Виконайте призначений номер зв'язку таблиці обміну даними один раз

● Інші інструкції

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
1900	WDT	—	✓		Сторожовий таймер
1901	DELAY	—	✓		Затримка виконання програми
1902	GPWM	—	—		Загальна широтно-імпульсна модуляція
1904	EPUSH	—	✓		Зберігання вмісту індексних регістрів
1905	EPOP	—	✓		Зчитування даних в індексні регістри
1906	INFO	—	✓		Читання системних даних

● Інструкції з обробки рядка

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
2100	BINDA	DBINDA	✓		Перетворення десяткового числа зі знаком у код ASCII
2101	BINHA	DBINHA	✓		Перетворення двійкового шістнадцяткового числа в шістнадцятковий код ASCII
2102	BCDDA	DBCDDA	✓		Перетворення двійкового десяткового числа в код ASCII

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
2103	DABIN	DDABIN	✓	DABIN En S DABINP En S DDABIN En S DDABINP En S	Перетворення десяткового коду ASCII зі знаком у десяткове двійкове число зі знаком
2104	HABIN	DABIN	✓	HABIN En S HABINP En S DHABIN En S DHABINP En S	Перетворення шістнадцяткового коду ASCII у шістнадцяткове двійкове число
2105	DABCD	DDABCD	✓	DABCD En S DABCDP En S DDABCD En S DDABCDP En S	Перетворення ASCII-коду в двійкове десяткове число
2106	\$LEN	—	✓	\$LEN En S \$LENP En S	Обчислення довжини рядка
2109	\$FSTR	—	✓	\$FSTR En S1 S2 \$FSTRP En S1 S2	Перетворення числа з плаваючою комою в рядок
2110	\$FVAL	—	✓	\$FVAL En S \$FVALP En S	Перетворення рядка в число з плаваючою комою
2111	\$RIGHT	—	✓	\$RIGHT En S n \$RIGHTP En S n	Отримання символів із рядка починається справа.
2112	\$LEFT	—	✓	\$LEFT En S n \$LEFTP En S n	Отримання символів із рядка починається зліва.
2113	\$MIDR	—	✓	\$MIDR En S1 S2 \$MIDRP En S1 S2	Отримання частини рядка
2115	\$SER	—	✓	\$SER En S1 S2 N \$SERP En S1 S2 N	Пошук рядка
2116	\$RPLC	—	✓	\$RPLC En S1 S2 S3 S4 \$RPLCP En S1 S2 S3 S4	Заміна символів у рядку
2117	\$DEL	—	✓	\$DEL En S1 S2 S3 \$DELP En S1 S2 S3	Видалення символів у рядку
2118	\$CLR	—	✓	\$CLR En S \$CLRP En S	Очищення рядка
2119	\$INS	—	✓	\$INS En S1 S2 S3 \$INSP En S1 S2 S3	Вставка рядка
2122	SPLIT	—	✓	SPLIT En S1 S2 S3 S4 SPLITP En S1 S2 S3 S4	Розбиття струни
2123	MERGE	—	✓	MERGE En S1 S2 S3 MERGEP En S1 S2 S3	Об'єднання рядка

● Інструкції по Ethernet

API	Код інструкції		Інструкція пульсу	символ		функція
	16- бітний	32- розрядний				
2200	SOPEN	—	✓	SOPEN En S1 S2 S3 SOPENP En S1 S2 S3	Відкриття розетки	
2201	SSEND	—	✓	SSEND En S1 S2 SSENDP En S1 S2	Передача даних через сокет	
2203	SCLOSE	—	✓	SCLOSE En S1 S2 SCLOSEP En S1 S2	Закриття розетки	
2204	MSEND	—	✓	MSEND En S1 S2 S3 D MSENDP En S1 S2 S3 D	Надсилання електронного листа	
2206	INTOA	—	✓	INTOA En S D INTOAP En S D	Перетворення IP-адреси цілого типу в IP-адресу рядкового типу	
2207	IATON	—	✓	IATON En S D IATONP En S D	Перетворення IP-адреси рядкового типу в IP-адресу цілого типу	
2208	EIPRW	—	—	EIPRW En S1 S2 S3 S4 S5 S6 S7 n S D1 D2	Читання та запис даних EtherNet/IP	
2209	SCONF	—	✓	SCONF En S1 S2 S3 S4 S5 S6 S7 S8 S9 S10 SCONFP En S1 S2 S3 S4 S5 S6 S7 S8 S9 S10	Налаштування параметрів сокетів TCP/UDP	
2210	MCONF	—	✓	MCONF En S1 S2 S3 S4 S5 S6 S7 S8 S9 S10 S11 S12 MCONFP En S1 S2 S3 S4 S5 S6 S7 S8 S9 S10 S11 S12	Читання/запис даних Modbus TCP	
2211	EMCONF1	—		EMCONF1 En Server Port LMail Sub Verify User Passw EMCONF1P En Server Port LMail Sub Verify User Passw	Налаштування значень параметрів сервера електронної пошти	

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
2212	EMCONF2	—	✓	EMCONF2 En Index Mail EMCONF2P En Index Mail	Встановлення електронної адреси
2214	—	MQTT_ Connected	—	MQTT_Connect En Execute Status SID BrokenN- ClientID CleanSes- WillTopic WillMes- WillRet- WillQos WillFlag KeepAli-	Встановлення підключення MQTT
2215	MQTT_ Publish	—	—	MQTT_Publish En Execute Done SID Busy BrokenN- Error Topic ErrCode Message Size Qos Retain Dup	Публікація повідомлення MQTT
2216	MQTT_ Subscribe	—	—	MQTT_Subscribe En Execute Status SID Message BrokenN- RecvTop- Topic Qos	Підписка на тему MQTT

● Карта пам'яті / Інструкції щодо

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
2300	MWRIT	—	✓	MWRIT En C S S1 S2 S3 S4 MWRITP En C S S1 S2 S3 S4	Запис даних з ПЛК на карту пам'яті

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
2301	MREAD	—	✓	MREAD En C S S1 S2 S3 D MREADP En C S S1 S2 S3 D	Зчитування даних з карти пам'яті в ПЛК
2302	MTWRIT	—	✓	MTWRIT En C S S1 S2 S3 MTWRITP En C S S1 S2 S3	Записування рядка на карту пам'яті
2303	MEMW	—	✓	MEMW En S D n MEMWP En S D n	Запис даних у файловий реєстр
2304	MDEL	—	✓	MDEL En ctrl fname MDELP En ctrl fname	Видалення файлів на карті пам'яті
2305	Record_trigger	—	—	Record_Trigger En Execute SID Tb_num W_mode Done Busy Error ErrCode	Запис і запис даних у файл

● Інструкції з контролю завдань

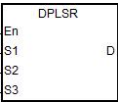
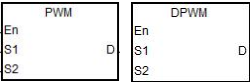
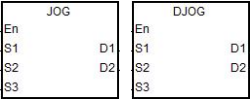
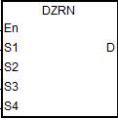
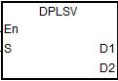
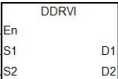
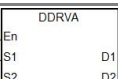
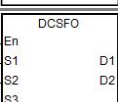
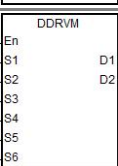
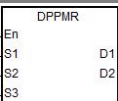
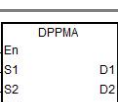
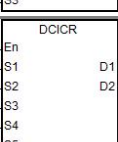
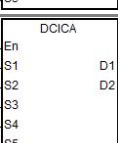
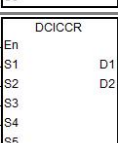
API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
2400	TKON	—	✓	TKON En S TKONP En S	Включення циклічного завдання
2401	TKOFF	—	✓	TKOFF En S TKOFFP En S	Відключення циклічного завдання

● Інструкції щодо послідовних функціональних діаграм (SFC).

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
2500	SFCRUN	—	—	SFCRUN En S1 S2 S3	Увімкнення SFC
2501	SFCPSE	—	—	SFCPSE En S1 S2	Призупинення SFC
2502	SFCSTP	—	—	SFCSTP En S	Зупинка SFC

● Інструкції щодо швидкого виведення

API	Код інструкції		Інструкція пульсу	символ	функція
	16-бітний	32-розрядний			
2700	—	DPLSY	—	DPLSY En S1 S2 D	Високошвидкісний імпульсний вихід (без процесу підвищення/зниження)

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
2701	–	DPLSR	–		Високошвидкісний імпульсний вихід (з процесом підвищення/зниження)
2702	PWM	DPWM	–		Широтно-імпульсна модуляція
2703	JOG	DJOG	–		Вихід JOG
2704	–	DZRN	–		Нульова віддача
2705	–	DPLSV	–		Регульований імпульсний вихід
2706	–	DDRVI	–		Контроль відносного положення
2707	–	DDRVA	–		Абсолютний контроль позиції
2708	CSFO	–	–		Швидкість улову та пропорційний вихід
2709	–	DDRVM	–		Розташування вирівнювання позначок
2710	–	DPPMR	–		2-осьовий синхронізований рух від точки до точки з відносною координатою
2711	–	DPPMA	–		2-осьовий абсолютний координатний синхронізований рух від точки до точки
2712	–	DCICR	–		2-осьова дугова інтерполяція відносного положення за годинниковою стрілкою
2713	–	DCICA	–		2-осьова дугова інтерполяція за годинниковою стрілкою в абсолютному положенні
2714	–	DCICCR	–		2-осьова дугова інтерполяція відносного положення проти годинникової стрілки

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
2715	—	DCICCA	—		2-осьова дугова інтерполяція проти годинникової стрілки
2716	—	DCCMR	—		Креслення кола відносного положення
2717	—	DCCMA	—		Малювання кола в абсолютному положенні
2718	TPO	—	—		Таблиця планування позицій контролює вихід
2719	—	DTPWS	✓		Налаштування вихідних параметрів однієї осі в таблиці планування положення
2720	—	DTPWL	✓		Встановлення параметрів лінійної інтерполяції в таблиці планування позиції
2721	—	DPTWC	✓		Налаштування параметрів дугової інтерполяції в таблиці планування положення
2723	—	DPPGB	—		Від точки до точки ходити вперед і назад
2724	—	DZRN2	—		Нульове повернення 2 (можна встановити напрямок виведення)

● Інструкції зі зв'язку Delta Special CANopen / EtherCAT

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
2800	INIC	—	—		Ініціалізація сервоприводу для зв'язку CANopen
2801	ASDON	—	—		Servo-ON і Servo-OFF
2802	CASD	—	—		Встановлення часу розгону та часу уповільнення для сервоприводу
2803	—	DDRVIC	—		Контроль відносного положення сервоприводу

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16-бітний	32-розрядний			
2804	—	DDRVAC	—	DDRVAC En S1 S2 S3	Абсолютний контроль положення сервоприводу
2805	—	DPLSVC	—	DPLSVC En S1 S2	Серворегулювання швидкості
2806	ZRNC	DZRNC	—	ZRNC En S1 S2 S3 DZRNC En S1 S2 S3	Сервонаведення
2807	COPRW	—	—	COPRW En S1 S2 S3 S4 S5 D	Читання та запис даних зв'язку CANopen
2808	COPWL	DCOPWL	—	COPWL En S1 S2 S3 D DCOPWL En S1 S2 S3 D	Запис кількох значень параметрів CANopen
2809	RSTD	—	—	RSTD En Node Para Ok Err	Надсилання команди Reset або NMT
2810	ZRNM	—	—	ZRNM En Node Mode Ok Err	Налаштування режиму наведення для сервоприводу Delta
2811	EMER	—	—	EMER En Node Dest Len Ok Err	Читання екстреного повідомлення
2812	—	DCSFOC	—	DCSFOC En Xno InCnt InSpd Gear Axis OutCnt OutSpd	Керування функцією відстеження сервоприводу через зв'язок
2813	ECAMD	—	—	ECAMD En Slave Source Len CAAddr Done Err ErrCode	Завантаження діаграми E-CAM на Slave
2814	—	DECAMS	—	DECAMS En Slave Smode CAAddr CLen Rpuu Cmode WbPuu OutT CaPuu CwPuu Orate Done Err ErrCode	Налаштування та планування параметрів E-CAM
2815	ECAMA	—	—	ECAMA En Slave CAMEn CAMSt MIPuu SOPuu Err ErrCode	Активація управління E-CAM і відображення стану
2816	ECAMC	—	—	ECAMC En Slave CAAddr CLen Done Err ErrCode	Зміна початкової адреси для діаграми E-CAM і довжини діаграми E-CAM у сервоприводі

API	Код інструкції		Інструкція пульсу	СИМВОЛ	функція
	16- бітний	32- розрядний			
2817	—	DTQC	—	DTQC En Node CurTQ TarTQ Reach Lspeed Err	Контроль крутного моменту
2818	—	DTQLC	—	DTQLC En Node CurTQ TarSpd Reach LimitTQ Err	Швидкість регулювання крутного моменту
2820	INITEC	—	—	INITEC En SlaveNo Done Stype ErrCo ErrCode	Ініціалізація підлеглих пристроїв для зв'язку EtherCAT
2821	ECATRW	—	—	ECATRW En Slave Done RWCo ErrCo Index ErrCode SubInd- Length Data	Читання та запис даних EtherCAT SDO
2822	MOTST	—	—	MOTST En Slave Position Reset State Done ErrCo ErrCode	Читання стану руху підлеглого EtherCAT
2823	ECATST	—	—	ECATST En Slave State Done ErrCo ErrCode	Читання стану підлеглого EtherCAT

3.4.2 Застосовувані інструкції (відсортовані за алфавітом)

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
символ	0114	\$+	–	✓	З'єднання двох рядків
	2118	\$CLR	–	✓	Очищення рядка
	2117	\$DEL	–	✓	Видалення символів у рядку
	2109	\$FSTR	–	✓	Перетворення числа з плаваючою комою в рядок
	2110	\$FVAL	–	✓	Перетворення рядка в число з плаваючою комою
	2119	\$INS	–	✓	Вставка рядка
	2112	\$LEFT	–	✓	Отримання символів у рядку починається зліва.
	2106	\$LEN	–	✓	Обчислення довжини рядка
	2113	\$MIDR	–	✓	Отримання частини рядка
	0302	\$MOV	–	✓	Перенесення рядка
	2111	\$RIGHT	–	✓	Отримання символів у рядку починається справа.
	2116	\$RPLC	–	✓	Заміна символів у рядку
	2115	\$SER	–	✓	Пошук рядка
	0102	*	D*	✓	Множення двійкових чисел
	0103	/	D/	✓	Ділення двійкових чисел
	0100	+	D+	✓	Додавання двійкових чисел
A	1210	ABS	DABS	✓	Абсолютне значення
	0705	ABSD	DABSD	–	Абсолютний барабанний секвенсор
	1424	ADLOG	DADLOG	–	Запис даних модуля аналогового введення.
	1425	ADPEAK	DADPEAK	–	Запис пікових даних модуля аналогового введення.
	0700	ALT	–	✓	Чергування між ON і OFF
	0043	AND\$<>	–	–	S1≠S2
	0042	AND\$=	–	–	S1 = S2
	0812	AND&	DAND&	–	S1&S2
	0814	AND^	DAND^	–	S1^S2
	0813	AND	DAND	–	S1 S2
	0010	AND<	DAND<	–	S1 < S2
	0011	AND<=	DAND<=	–	S1 <= S2
	0007	AND<>	DAND<>	–	S1≠S2
	0006	AND=	DAND=	–	S1 = S2
	0008	AND>	DAND>	–	S1 > S2
	0009	AND>=	DAND>=	–	S1 >= S2

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
	0076	ANDZ<	DANDZ<	–	S1-S2 < S3
	0077	ANDZ<=	DANDZ<=	–	S1-S2 <= S3
	0073	ANDZ<>	DANDZ<>	–	S1-S2 ≠ S3
	0072	ANDZ=	DANDZ=	–	S1-S2 = S3
	0074	ANDZ>	DANDZ>	–	S1-S2 > S3
	0075	ANDZ>=	DANDZ>=	–	S1-S2 >= S3
	1703	ARWS	–	–	Введення клавіш зі стрілками
	2801	ASDON	–	–	Servo ON і servo OFF
	1815	ASDRW	–	–	Інструкція по послідовному зв'язку виключно для сервоприводу Delta
	1016	ATMR	–	–	16-бітний контактний таймер (одиниця: 100 мс)
	1017	ATMRH	–	–	16-бітний контактний таймер (одиниця: 1 мс)
	1018	ATMRM	–	–	16-бітний контактний таймер (одиниця: 10 мс)
B	1222	BAND	DBAND	✓	Контроль зони нечутливості
	0200	BCD	DBCD	✓	Перетворення двійкового числа в двійкове десятичне число
	2102	BCDDA	DBCDDA	✓	Перетворення двійкового десятичного числа в код ASCII
	0201	BIN	DBIN	✓	Перетворення десятичного числа з двійковим кодом у двійкове
	2100	BINDA	DBINDA	✓	Перетворення підписаного десятичного числа в код ASCII
	2101	BINHA	DBINHA	✓	Перетворення двійкового шістнадцяткового числа в шістнадцятковий код ASCII
	0113	BK-	DBK-	✓	Віднімання двійкових чисел блоками
	0112	BK+	DBK+	✓	Додавання двійкових чисел у блоки
	0214	BKBCD	–	✓	Перетворення двійкових чисел у блоках у двійкові десятичні числа в блоках
	0215	BKBIN	–	✓	Перетворення двійкових чисел у блоках у двійкові десятичні числа в блоках
	0304	BMOV	DBMOV	✓	Передача всіх даних
	1207	BON	DBON	✓	Перевірка стану біта
	1302	BREAK	–	–	Завершення циклу FOR-NEXT
	1113	BSFL	–	✓	Зсув станів n-розрядних пристроїв на один біт вліво
	1112	BSFR	–	✓	Зсув станів n-розрядних пристроїв на один біт вправо
	0307	BXCH	–	✓	Обмін усіма даними
C	1819	CANRS	–	–	Надсилання та отримання користувацького зв'язку CAN
	2802	CASD	–	–	Встановлення часу розгону та часу уповільнення для сервоприводу

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
	1209	CCD	–	✓	Контрольна сума
	1816	CCONF	–	✓	Налаштування параметрів у таблиці обміну даними комунікаційного порту
	0065	CHKADR	–	–	Перевірка адреси регістра покажчика
	0400	CJ	–	✓	Умовний стрибок
	0303	CML	DCML	✓	Інвертування даних
	0054	CMP	DCMP	✓	Порівняння значень
	0063	CMPT<	–	✓	Порівняння таблиць ПО: <
	0064	CMPT<=	–	✓	Порівняння таблиць ON: <=
	0060	CMPT<>	–	✓	Порівняння таблиць ПО: ≠
	0059	CMPT=	–	✓	Порівняння таблиць ПО: =
	0061	CMPT>	–	✓	Порівняння таблиць ПО: >
	0062	CMPT>=	–	✓	Порівняння таблиць ON: >=
	1003	CNT	–	–	16-розрядний лічильник
	1813	COMDF	–	✓	Встановлення формату зв'язку для послідовного порту зв'язку
	1812	COMRS	–	–	Надсилання та отримання комунікаційних даних
	2807	COPRW	–	–	Читання та запис даних зв'язку CANopen
	2808	COPWL	DCOPWL	–	Запис кількох значень параметрів CANopen
	1807	CRC	–	–	Циклічна перевірка надмірності
	2708	CSFO	–	–	Швидкість улову та пропорційний вихід
D	0101	–	D-	✓	Віднімання двійкових чисел $S_1 - S_2 = D$
	1010	–	DCAP	–	Захоплення високошвидкісного значення підрахунку в зовнішньому вхідному перериванні
	2717	–	DCCMA	–	Малювання кола в абсолютному положенні
	2716	–	DCCMR	–	Креслення кола відносного положення
	2713	–	DCICA	–	2-осьова дугова інтерполяція за годинниковою стрілкою в абсолютному положенні
	2715	–	DCICCA	–	2-осьова дугова інтерполяція проти годинникової стрілки
	2714	–	DCICCR	–	2-осьова дугова інтерполяція відносного положення проти годинникової стрілки
	2712	–	DCICR	–	2-осьова дугова інтерполяція відносного положення за годинниковою стрілкою

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
	1004	–	DCNT	–	32-розрядний лічильник
	2812	–	DCSFOC	–	Керування функцією відстеження сервоприводу через зв'язок
	2707	–	DDRVA	–	Абсолютний контроль позиції
	2804	–	DDRAC	–	Абсолютний контроль положення сервоприводу
	2706	–	DDRVI	–	Контроль відносного положення
	2803	–	DDRVIC	–	Контроль відносного положення сервоприводу
	2709	–	DDRVM	–	Розташування вирівнювання позначок
	2814	–	DECAMS	–	Налаштування та планування параметрів E-CAM
	1419	–	DHCCAP	–	Зафіксовано значення лічильника модуля HC
	1421	–	DHCCMP	–	Значення лічильника модуля HC у порівнянні для виведення
	1422	–	DHCCMPT	–	Значення лічильника модуля HC у груповому порівнянні для виведення
	1418	–	DHCCNT	–	Підраховано значення лічильника модуля HC
	1423	–	DHCMEAS	–	Модуль HC вимірювання частоти та обертів
	1006	–	DHSCR	–	Скидання високошвидкісного порівняння
	1005	–	DHSCS	–	Налаштування високошвидкісного порівняння
	1014	–	DHSCY	–	Циклічний підрахунок, порівняння та виведення
	0601	–	DHSRF	✓	Оновлення значень швидкісного порівняння
	1007	–	DHSZ	–	Швидке порівняння зон введення
	1415	–	DLCCAL	–	Калібрування каналу модуля LC
	1417	–	DMPID	–	Алгоритм PID для модуля RTD/TC
	0708	–	DPIDE	–	ПІД-алгоритм
	2701	–	DPLSR	–	Високошвидкісний імпульсний вихід (з процесом підвищення/зниження)
	2705	–	DPLSV	–	Регульований імпульсний вихід
	2805	–	DPLSVC	–	Серворегулювання швидкості
	2700	–	DPLSY	–	Високошвидкісний імпульсний вихід (без процесу підвищення/зниження)
	2723	–	DPPGB	–	Від точки до точки ходити вперед і назад
	2711	–	DPPMA	–	2-осьовий синхронізований рух від точки до точки з абсолютною координатою
	2710	–	DPPMR	–	2-осьовий синхронізований рух від точки до точки з відносною координатою
	2721	–	DPTWC	✓	Налаштування параметрів дугової інтерполяції в таблиці планування положення
	1410	–	DPUCNT	–	Функція високошвидкісного лічильника модуля PU
	1402	–	DPUCONF	✓	Налаштування вихідних контрольних параметрів ПУ

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
					модуль
	1406	–	DPUDRA	–	Вихід абсолютної адресації модуля PU (з прискоренням і уповільненням)
	1405	–	DPUDRI	–	Вихід відносного положення модуля PU (з прискоренням і уповільненням)
	1408	–	DPUJOG	–	Руховий вихід модуля PU
	1409	–	DPUMPG	–	Вихід MPG модуля PU
	1404	–	DPUPLS	–	Імпульсний вихід модуля PU (без прискорення)
	1407	–	DPUZRN	–	Наведення модуля ПУ
	1008	–	DSPD	–	Виявлення швидкості
	0711	–	DSUNRS	–	Налаштування часу сходу та заходу сонця
	2720	–	DTPWL	✓	Встановлення параметрів лінійної інтерполяції в таблиці планування позиції
	2719	–	DTPWS	✓	Налаштування вихідних параметрів однієї осі в таблиці планування положення
	2817	–	DTQC	–	Контроль крутного моменту
	2818	–	DTQLC	–	Швидкість регулювання крутного моменту
	2704	–	DZRN	–	Нульова віддача
	2724	–	DZRN2	–	Нульове повернення 2 (напрямок виходу можна встановити)
	2105	DABCD	DDABCD	✓	Перетворення ASCII-коду в двійкове десяткове число
	2103	DABIN	DDABIN	✓	Перетворення десяткового коду ASCII зі знаком у десяткове двійкове число зі знаком
	0116	DEC	DDEC	✓	Віднімання одиниці з двійкового числа
	1202	DECO	–	✓	Декодер
	1901	DELAY	–	✓	Затримка виконання програми
	1821	DESO	–	–	Виконайте призначений номер зв'язку таблиці обміну даними один раз
	0500	DI	–	–	Відключення переривання
	1215	DIS	–	✓	Роз'єднання 16-бітних даних
	0118	DIV16	DIV32	✓	Ділення двійкових чисел на 16-біт. Ділення двійкових чисел на 32-біт.
	0504	DIX	–	–	Включення певного переривання
	1820	DMVSH	–	–	Увімкнення виявлення та зв'язку Delta DMV
	1818	DNETRW	–	–	Читання та запис даних зв'язку DeviceNet
	1607	TZCP	–	✓	Літній час
	1702	DSW	–	–	DIP-перемикач
	1226	–	DTM	✓	Передача та переміщення даних
E	2815	ECAMA	–	–	Активація керування та відображення E-CAM

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
					статус
	2816	ECAMC	–	–	Зміна початкової адреси для діаграми E-CAM і довжини діаграми E-CAM у сервоприводі
	2813	ECAMD	–	–	Завантаження діаграми E-CAM на Slave
	2821	ECATRW	–	–	Читання та запис даних EtherCAT SDO
	2823	ECATST	–	–	Читання стану підлеглого EtherCAT
	0501	EI	–	–	Увімкнення переривання
	2208	EIPRW	–	–	Читання та запис через з'єднання EtherNet/IP
	0503	EIX	–	–	Відключення конкретного переривання
	2211	EMCONF1	–	✓	Налаштування значень параметрів сервера електронної пошти
	2212	EMCONF2	–	✓	Встановлення електронної адреси
	2811	EMER	–	–	Читання екстреного повідомлення
	1203	ENCO	–	✓	Кодувальник
	1905	EPOP	–	✓	Зчитування даних в індексні реєстри
	1904	EPUSH	–	✓	Зберігання вмісту індексних реєстрів
	0105	–	F-	✓	Віднімання чисел з плаваючою комою $S_1 - S_2 = D$
F	0106	–	F*	✓	Множення чисел з плаваючою комою $S_1 * S_2 = D$
	0107	–	F/	✓	Ділення чисел з плаваючою комою $S_1 / S_2 = D$
	0104	–	F+	✓	Додавання чисел з плаваючою комою $S_1 + S_2 = D$
	1504	–	FACOS	✓	Аркусинус числа з плаваючою комою
	0028	–	FAND<	–	$S_1 < S_2$
	0029	–	FAND<=	–	$S_1 \leq S_2$
	0025	–	FAND<>	–	$S_1 \neq S_2$
	0024	–	FAND=	–	$S_1 = S_2$
	0026	–	FAND>	–	$S_1 > S_2$
	0027	–	FAND>=	–	$S_1 \geq S_2$
	1503	–	FASIN	✓	Арксинус числа з плаваючою комою
	1505	–	FATAN	✓	Арктангенс числа з плаваючою комою
	0212	–	FBCD	✓	Перетворення двійкового числа з плаваючою комою в десяткове число з плаваючою комою
	0213	–	FBIN	✓	Перетворення десяткового числа з плаваючою комою у двійкове число з плаваючою комою
	0056	–	FCMP	✓	Порівняння чисел з плаваючою комою
	1501	–	FCOS	✓	Косинус числа з плаваючою комою

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
	1507	–	FCOSH	✓	Гіперболічний косинус числа з плаваючою комою
	1510	–	FDEG	✓	Перетворення радіанів у градуси
	1513	–	FEXP	✓	Показник степеня числа з плаваючою комою
	0022	–	FLD<	–	$S1 < S2$
	0023	–	FLD<=	–	$S1 \leq S2$
	0019	–	FLD<>	–	$S1 \neq S2$
	0018	–	FLD=	–	$S1 = S2$
	0020	–	FLD>	–	$S1 > S2$
	0021	–	FLD>=	–	$S1 \geq S2$
	1515	–	FLN	✓	Натуральний логарифм двійкового числа з плаваючою комою
	1514	–	FLOG	✓	Логарифм числа з плаваючою комою
	1224	–	FMEAN	✓	Середнє значення чисел з плаваючою комою
	0211	–	FNEG	✓	Зміна знака числа з плаваючою комою
	0034	–	FOR<	–	$S1 < S2$
	0035	–	FOR<=	–	$S1 \leq S2$
	0031	–	FOR<>	–	$S1 \neq S2$
	0030	–	FOR=	–	$S1 = S2$
	0032	–	FOR>	–	$S1 > S2$
	0033	–	FOR>=	–	$S1 \geq S2$
	1516	–	FPOW	✓	Піднесення числа з плаваючою комою до степеня
	1509	–	FRAD	✓	Перетворення градусів у радіани
	1500	–	FSIN	✓	Синус числа з плаваючою комою
	1506	–	FSINH	✓	Гіперболічний синус числа з плаваючою комою
	1229	–	FSORT	✓	Сортування даних у форматі з плаваючою комою
	1512	–	FSQR	✓	Квадратний корінь із числа з плаваючою комою
	1225	–	FSUM	✓	Сума чисел з плаваючою комою
	1502	–	FTAN	✓	Тангенс числа з плаваючою комою
	1508	–	FTANH	✓	Гіперболічний тангенс числа з плаваючою комою
	0057	–	FZCP	✓	Порівняння зон із плаваючою комою
	0202	FLT	DFLT	✓	Перетворення двійкового цілого числа в двійкове число з плаваючою комою
	1300	FOR	–	–	Початок вкладеного циклу
	1400	FROM	DFROM	✓	Читання даних з керуючого регістру в модулі розширення
	0209	GBIN	DGBIN	✓	Перетворення коду Грея в двійковий

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
					номер
G	0402	GOEND	–	–	Перехід до END
	1902	GPWM	–	–	Загальна широтно-імпульсна модуляція
	0208	GRY	DGRY	✓	Перетворення двійкового числа в код Грея
	2104	HABIN	DHABIN	✓	Перетворення шістнадцяткового коду ASCII у шістнадцяткове двійкове число
H	1420	HCDO	–	–	Цифровий вихід модуля HC
	1701	HKY	DHKY	–	Введення шістнадцяткової клавіші
	1604	HOUR	–	–	Лічильник часу роботи
	2207	IATON	–	✓	Перетворення IP-адреси рядкового типу в IP-адресу цілого типу
I	1012	IETS	–	✓	Початок вимірювання часу виконання інструкції
	1013	IETE	–	✓	Кінець вимірювання часу виконання інструкції
	0115	INC	DINC	✓	Додавання одиниці до двійкового числа
	0706	INCD	–	–	Інкрементний барабанний секвенсор
	1906	INFO	–	✓	Читання системних даних
	2800	INIC	–	–	Ініціалізація сервоприводу для зв'язку CANopen
	2820	INITEC	–	–	Ініціалізація підлеглих пристроїв для зв'язку EtherCAT
	0204	INT	DINT	✓	Перетворення 32-розрядного числа з плаваючою комою в двійкове ціле число
	2206	INTOA	–	✓	Перетворення IP-адреси цілого типу в IP-адресу рядкового типу
	1426	IOLINKR	–	–	Зчитайте параметр із пристрою O-Link
	1427	IOLINKW	–	–	Запишіть параметр у пристрій IO-Link
	0401	JMP	–	–	Безумовний стрибок
J	2703	JOG	DJOG	–	Вихід JOG
	1416	LCWEI	–	–	Зчитування значення ваги через модуль LC
L	0037	LD\$<>	–	–	S1≠S2
	0036	LD\$=	–	–	S1 = S2
	0809	LD&	DLD&	–	S1&S2
	0811	LD^	DLD^	–	S1^S2
	0810	LD	DLD	–	S1 S2
	0004	LD<	DLD<	–	S1 < S2
	0005	LD<=	DLD<=	–	S1 <= S2
	0001	LD<>	DLD<>	–	S1≠S2

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
	0000	LD=	DLD=	–	S1 = S2
	0002	LD>	DLD>	–	S1 > S2
	0003	LD>=	DLD>=	–	S1 >= S2
	0070	LDZ<	DLDZ<	–	S1-S2 < S3
	0071	LDZ<=	DLDZ<=	–	S1-S2 <= S3
	0067	LDZ<>	DLDZ<>	–	S1-S2 ≠ S3
	0066	LDZ=	DLDZ=	–	S1-S2 = S3
	0068	LDZ>	DLDZ>	–	S1-S2 > S3
	0069	LDZ>=	DLDZ>=	–	S1-S2 >= S3
	1221	LIMIT	DLIMIT		Обмеження значення в межах
	1806	LRC	–	–	Поздовжня перевірка парності
M	2214	–	MQTT_Connect	–	Встановлення підключення MQTT
	0801	MAND	–	✓	Матрична операція I
	2304	MDEL	–	✓	Видалення файлів на карті пам'яті
	2123	MERGE	–	✓	Об'єднання рядка
	1214	MBC	–	✓	Підрахунок бітів зі значенням 0 або 1
	0904	MBR	–	✓	Обертання біт матриці
	1212	MBRD	–	✓	Читання біта матриці
	1109	MBS	–	✓	Зсув бітів матриці
	1213	MBWR	–	✓	Запис розрядності матриці
	0058	MCMP	–	✓	Матричне порівняння
	2210	MCONF	–	✓	Читання/запис даних Modbus TCP
	1208	MEAN	DSEAN	✓	Середній
	2303	MEMW	–	✓	Запис даних у файловий реєстр
	1211	MINV	–	✓	Інвертування бітів матриці
	1430	MLPF	–	–	Цифровий фільтр низьких частот для аналогових вхідних сигналів
	0206	MMOV	–	✓	Перетворення 16-бітного значення на 32-бітове
	1808	MODRW	–	–	Читання/запис даних MODBUS
	1817	MODRWE	–	–	Читання та запис даних Modbus без використання будь-яких прапорів
	0803	MOR	–	✓	Матрична операція OR
	2822	MOTST	–	–	Читання стану руху підлеглого EtherCAT
	0300	MOV	DMOV	✓	Передача даних
	0310	MOVB	–	✓	Передача кількох бітів
	2215	MQTT_Publish	–	–	Публікація повідомлення MQTT

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
	2216	MQTT_Subscribe	–	–	Підписка на тему MQTT
	2301	MREAD	–	✓	Зчитування даних з карти пам'яті в ПЛК
	2204	MSEND	–	✓	Надсилення електронного листа
	0704	MTR	–	–	Матричний вхід
	2302	MTWRIT	–	✓	Записування рядка на карту пам'яті
	0117	MUL16	MUL32	✓	Множення двійкових чисел для 16-біт/32-біт
	2300	MWRIT	–	✓	Запис даних з ПЛК на карту пам'яті
	0805	MXOR	–	✓	Матрична операція виняткового OR
	0210	NEG	DNEG	✓	Доповнення двох
N	1301	NEXT	–	–	Кінець вкладеного циклу
	0305	NMOV	DNMOV	✓	Передача даних на кілька пристроїв
	1115	NSFL	DNSFL	✓	Зсув n регістрів вліво
	1114	NSFR	DNSFR	✓	Зсув n регістрів вправо
	0049	OR\$<>	–	–	S1≠S2
O	0048	OR \$=	–	–	S1 = S2
	0815	OR&	DOR&	–	S1&S2
	0817	OR^	DOR^	–	S1^S2
	0816	OR	DOR	–	S1 S2
	0016	OR<	DOR<	–	S1 < S2
	0017	OR<=	DOR<=	–	S1<=S2
	0013	OR<>	DOR<>	–	S1 ≠ S2
	0012	OR=	DOR=	–	S1 = S2
	0014	OR>	DOR>	–	S1 > S2
	0015	OR>=	DOR>=	–	S1>=S2
	0082	ORZ<	DORZ<	–	S1-S2 < S3
	0083	ORZ<=	DORZ<=	–	S1-S2 <= S3
	0079	ORZ<>	DORZ<>	–	S1-S2 ≠ S3
	0078	ORZ=	DORZ=	–	S1-S2 = S3
	0080	ORZ>	DORZ>	–	S1-S2 > S3
	0081	ORZ>=	DORZ>=	–	S1-S2 >= S3
	1015	PPDT	–	–	Виявлення різниці в часі між двома фазами
P	1403	PUSTAT	–	–	Читання вихідного стану модуля PU
	1009	PWD	–	–	Виявлення ширини імпульсу
	2702	PWM	DPWM	–	Широтно-імпульсна модуляція

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
R	0703	RAMP	DRAMP	–	Сигнал рампи
	1517	RAND	–	✓	Випадкове число
	0903	RCL	DRCL	✓	Обертання вліво з прапорцем для перенесення
	0901	RCR	DRCR	✓	Обертання вправо з прапорцем для перенесення
	2305	Record_trigger	–	–	Запис і запис даних у файл
	0600	REF	–	✓	Оновлення введення-виведення
	0602	REFF	–	✓	Оновлення часу фільтрації введення/виведення
	0207	RMOV	–	✓	Перетворення 32-бітного значення на 16-бітне
	0902	ROL	DROL	✓	Обертання вліво
	0900	ROR	DROR	✓	Обертання вправо
	1000	RST	DRST	–	Скидання контакту або очищення реєстру
	2809	RSTD	–	–	Надсилання команди Reset або NMT
	0216	SCAL	DSCAL	✓	Операція масштабного значення
	0222	SCLM	DSCLM	✓	Багатоточкова операція співвідношення площ
S	2203	SCLOSE	–	✓	Закриття розетки
	0217	SCLP	DSCLP	✓	Тип параметра операції значення масштабу
	2209	SCONF	–	✓	Налаштування параметрів сокетів TCP/UDP
	1204	SEGD	–	✓	Семисегментное декодування
	1704	SEGL	–	–	Семисегментний дисплей із засувками
	1200	SER	DSER	✓	Пошук даних
	2501	SFCPSE	–	–	Призупинення SFC
	2500	SFCRUN	–	–	Увімкнення SFC
	2502	SFCSTP	–	–	Зупинка SFC
	1107	SFDEL	DSFDEL	✓	Видалення даних зі списку даних
	1108	SFINS	DSFINS	✓	Вставлення даних у список даних
	1111	SFL	DSFL	✓	Зсув значень розрядів у регістрах на n розрядів вліво
	1106	SFPO	DSFPO	✓	Читання останніх даних зі списку даних
	1110	SFR	DSFR	✓	Зсув значень розрядів у регістрах на n розрядів вправо
	1105	SFRD	DSFRD	✓	Зсув даних і читування їх із пристрою слова
	1101	SFTL	–	✓	Зсув станів пристроїв вліво
	1100	SFTR	–	✓	Зсув станів пристроїв вправо
	1104	SFWR	DSFWR	✓	Зміщення даних і запис їх у слово пристрій

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
	0309	SMOV	–	✓	Перенесення цифр
	2122	SPLIT	–	✓	Розбиття струни
	2200	SOPEN	–	✓	Розкриття розетки
	1205	SORT	DSORT	✓	Сортування даних
	1511	SQR	DSQR	✓	Квадратний корінь з двійкового числа
	2201	SEND	–	✓	Надсилання даних через сокет
	0702	STMR	–	–	Спеціальний таймер
	1201	SUM	DSUM	✓	Кількість бітів, стани яких увімкнено
	0308	SWAP	DSWAP	✓	Замінити старший байт на молодший
	1603	T-	–	✓	Віднімання часу
T	1602	T+	–	✓	Додавання часу
	1605	TCMP	–	✓	Порівняння часу
	2401	TKOFF	–	✓	Відключення циклічного завдання
	2400	TKON	–	✓	Включення циклічного завдання
	1700	TKY	DTKY	–	Введення десяти клавіш
	1001	TMR	–	–	16-бітний таймер (одиниця: 100 мс)
	1002	TMRH	–	–	16-бітний таймер (одиниця: 1 мс)
	1011	TMRM	–	–	16-бітний таймер (одиниця: 10 мс)
	1401	TO	DTO	✓	Запис даних у контрольний регістр у спеціальному модулі
	2718	TPO	–	–	Таблиця планування позицій контролює вихід
	1600	TRD	–	✓	Читання часу
	0701	TTMR	–	–	Таймер режиму навчання
	1601	TWR	–	✓	Написання часу
	1606	TZCP	–	✓	Порівняння часових поясів
	1216	UNI	–	✓	Об'єднання 16-бітних даних
U	1814	VFDRW	–	–	Інструкція по послідовному зв'язку виключно для приводу змінного струму Delta
V	0800	WAND	DAND	✓	Логічна операція I
W	1608	WON	–	–	Встановлення тижневого робочого часу
	1900	WDT	–	✓	Сторожовий таймер
	0808	WINV	DINV	✓	Логічна зворотна операція INV
	0802	WOR	DOR	✓	Логічна операція OR
	1103	WSFL	DWSFL	✓	Зсув даних у слові пристрої вліво
	1102	WSFR	DWSFR	✓	Зсув даних у слові пристрої вправо
	1217	WSUM	DWSUM	✓	Отримання суми

Класифікація	API	Код інструкції		Інструкція пульсу	функція
		16-бітний	32-розрядний		
	0804	WXOR	DXOR	✓	Логічна операція виключаючого OR
	0306	XCH	DXCH	✓	Обмін даними
X	0709	XCMP	–	–	Налаштування для порівняння вхідних даних кількох робочих станцій
	0710	YOUT	–	–	Порівняння результатів кількох робочих станцій
Y	0055	ZCP	DZCP	✓	Порівняння зон
Z	1223	ZONE	DZONE	✓	Контроль зони
	2806	ZRNC	DZRNC	–	Сервонаведення
	2810	ZRNM	–	–	Налаштування режиму наведення для сервоприводу Delta
	1206	ZRST	–	✓	Скидання зони
	1228	ZSET	–	✓	Налаштування зони

Розділ 4 Структура інструкції

Зміст

4.1 Прикладні інструкції - опис API	4-2
4.2 Опис використання операнда	4-4
4.3 Обмеження щодо використання інструкцій ..	4-7
4.4 Індексні реєстри	4-9
4.5 Реєстри показчиків	4-12
4.6 Реєстри показчиків таймерів	4-14
4.7 Реєстри показчиків для 16-розрядних лічильників	4-16
4.8 Реєстри показчиків для 32-розрядних лічильників	4-18
4.9 Реєстр файлів	4-20

4.1 Прикладні інструкції - Опис API

У цьому розділі описано, як цей посібник документує кожну інструкцію API. Кожна інструкція має власний код інструкції та номер API. Інструкції поділено на розділи на основі пов'язаних функцій інструкцій, так що всі арифметичні інструкції знаходяться в одному розділі, а всі інструкції порівняння – в іншому. У наступному прикладі використовується інструкція MOV. Номер API інструкції в таблиці – 0300, код інструкції – MOV, а функція передає дані.

API	Код інструкції			Операнд							функція						
0300	D	MOV	P	S, D							Передача даних						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		○	○	○	○		○
D		●			●	●	●	●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●				●		●		●	●	
D		●	●				●		●		●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

Символ:

MOV	MOV P
En	En
S	S
D	D
DMOV	DMOV P
En	En
S	S
D	D

S : Джерело даних
D : Призначення даних

1. Пристрої, які використовує інструкція, перераховані в стовпці операнда. **S**, **D**, **n** і **m** використовуються як операнди відповідно до їхніх функцій. Якщо використовується більше ніж один операнд, і ці операнди мають однакову функцію, вони розрізняються нижніми індексами; наприклад, **S**₁ і **S**₂.
2. Якщо ви можете використовувати 16-бітну інструкцію як 32-бітну інструкцію, буква **D** додається до коду 16-бітної інструкції, щоб сформувати 32-бітну форму інструкції. Якщо ви можете використовувати інструкцію як імпульсну інструкцію, до коду інструкції додається літера **P**. Наприклад, «D***P», де «***» є кодом інструкції, вказує на 32-бітну імпульсну інструкцію.
3. **F** в області операнда вказує на число з плаваючою комою одинарної точності (32-розрядне).
4. Суцільне коло ● вказує на те, що пристрій можна змінювати за допомогою індексного регістра, а порожнє коло ○ вказує на те, що пристрій не можна змінювати за допомогою індексного регістру. Наприклад, регістр даних, позначений операндом **S**, може бути змінений регістром індексу.
5. Відповідна модель вказана в таблиці. Ви можете перевірити, чи можна використовувати інструкцію як імпульсну інструкцію, 16-бітну інструкцію, 32-бітну інструкцію або 64-бітну інструкцію відповідно до інформації в таблиці.
6. Якщо ви хочете використовувати інструкцію у функціональному блоці, а пристрої виводу, введення та даних підтримуються в операндах, ви повинні використовувати регістри покажчиків. AS вказує на використання таймера, 16-бітного лічильника та 32-бітного

лічильник, який підтримується в операндах, ви повинні використовувати регістр показника таймера, 16-бітний регістр показника лічильника та 32-бітний регістр показника лічильника. Зверніться до розділів 4.4–4.7 для отримання додаткової інформації або до розділу 7.2.4 у посібнику ISPSOft.

7. Символи, що представляють інструкцію MOV в ISPSOft:

MOV, MOVP, DMOV і DMOVP є кодами інструкцій для цієї інструкції

En : увімкнути

S : Джерело даних (придатний формат операнда — слово/подвійне слово.)

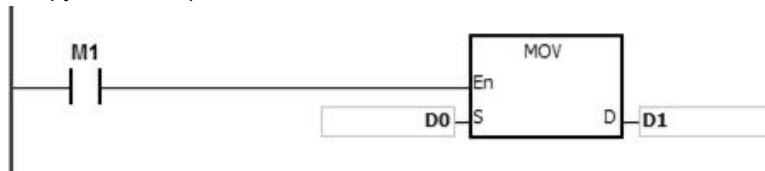
D : адресат даних (придатним форматом операнда є слово/подвійне слово.)

Прикладний інструкційний склад

Деякі прикладні інструкції складаються з кодів інструкцій. Наприклад, інструкції EI, DI та WDT; однак більшість прикладних інструкцій складаються з кодів інструкцій і кількох операндів.

Кожна застосована інструкція має власний номер API та код інструкції. Наприклад, код інструкції API 0300 є інструкцією MOV (передача даних). Ви можете ввести прикладну інструкцію трьома способами.

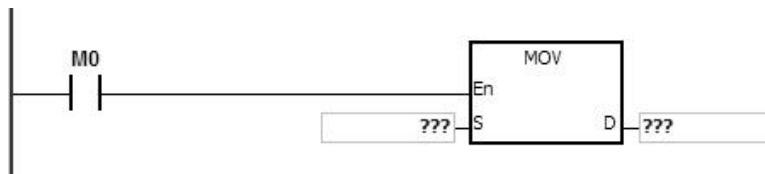
Введіть інструкцію безпосередньо: ви можете ввести інструкцію в ISPSOft. Для інструкції MOV введіть назву інструкції та операнди, щоб позначити «MOV D0



Введіть інструкцію шляхом перетягування: ви можете перетягнути інструкцію MOV з **API** в ISPSOft до редактора схованої діаграми.

Введіть інструкцію з панелі інструментів: ви можете клацнути **Вибір API/FB** на панелі інструментів у ISPSOft, а потім клацнути **API**.

Нарешті клацніть інструкцію MOV у **Передачі даних**.



S	Вихідний операнд Якщо є більше ніж один операнд-джерело, операнди-джерела позначаються індексом (наприклад, S 1 , S 2).
D	Операнд призначення Якщо існує більше одного операнда призначення, операнди призначення представлені індексом (наприклад, D 1 , D 2).
Якщо операнд може бути тільки константою K/H або значенням регістру, він представлений m , m₁ , m₂ , n , n₁ або n₂ .	

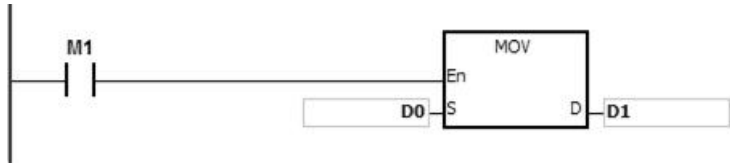
Довжина операнда (6-бітні, 32-бітні чи числові інструкції з плаваючою комою):

16-розрядні або 32-розрядні інструкції

Значення операндів в інструкціях поділяються на 16-бітні значення та 32-бітові значення. Для обробки даних різничевої довжини інструкції поділяються на 16-бітні та 32-бітові. Щоб відрізнити 32-бітну інструкцію від 16-бітної форми, перед кодом 16-бітної інструкції (16-біт MOV і 32-біт DMOV) додається D.

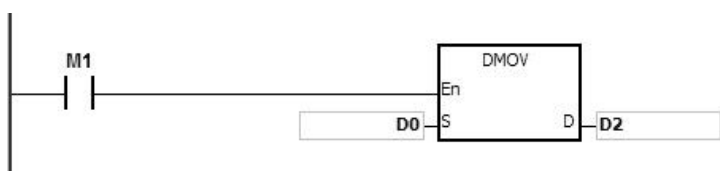
Інструкція числа з плаваючою комою

16-розрядна інструкція MOV



Коли M1 увімкнено, дані з D0 передаються на D1.

32-розрядна інструкція DMOV

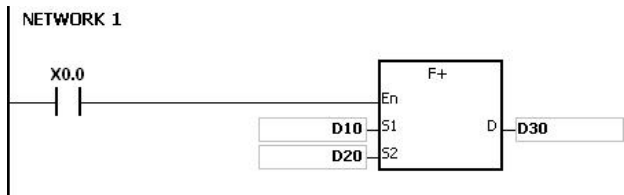


Коли M1 увімкнено, дані з (D1, D0) передаються в (D3, D2).

Інструкції з числами з плаваючою комою

Інструкції чисел з плаваючою комою підтримують 32-розрядні інструкції чисел з плаваючою комою, які відповідають точні інструкції чисел з плаваючою комою. Зверніться до глави 2 для отримання додаткової інформації про числа з плаваючою комою.

32-розрядна інструкція F+ з числом з плаваючою комою одинарної точності



Коли X0.0 увімкнено, дані в (D11, D10) і (D21, D20) перенесено в (D31, D30).

Безперервне виконання та імпульсне виконання команд

1. Виконання інструкцій можна розділити на безперервне та імпульсне. Ви можете скоротити цикл сканування за допомогою імпульсних інструкцій, тому що, коли інструкція не виконується, для виконання програми потрібно менше часу.
2. Імпульсна функція дозволяє відповідній інструкції увімкнути керуючий вхід, що запускається по наростаючому фронту. Інструкція ввімкнена для одного циклу сканування.
3. Якщо керуючий вхід залишається УВІМКНЕНИМ, а відповідна інструкція не виконується, для виконання інструкції керуючий вхід потрібно знову перемкнути з OFF. на ON.
4. Нижче показано різницю між імпульсними та безперервними інструкціями:

Імпульсне виконання	Коли M1 перемикається з OFF на ON, інструкція MOVP виконується один раз. Інструкція не виконується знову в циклі сканування. Тому її називають імпульсною інструкцією.
Безперервне виконання	Щоразу, коли M1 увімкнено під час циклу сканування, інструкція MOV виконується один раз. Тому інструкцію називають безперервною.

Коли умовний контакт M1 вимкнено, жодна інструкція не виконується, і значення в операнді призначення D не змінюється.

4.2 Опис використання операнда

У серії AS є 2 типи операндів: визначені користувачем і визначені системою.

Визначені користувачем операнди

- Вхідні реле: X0.0–X63.15 або X0–X63
- Вихідні реле: Y0.0–Y63.15 або Y0–Y63
- Внутрішні реле: M0–M8191
- Крокові реле: S0–S2047
- Таймери: T0–T511
- 16-розрядні лічильники: C0–C511
- 32-розрядні лічильники: HC0–HC255
- Регістри даних: D0–D29999 або D0.0–D29999.15
- Регістри файлів: FR0–FR65535
- Спеціальні допоміжні прапори: SM0–SM2047
- Регістри спеціальних даних: SR0–SR2047
- Індексні регістри: E0–E9
- Константи: десяткові константи позначаються K, а шістнадцяткові – 16#.
- Рядки: "\$"
- Числа з плаваючою комою: числа з плаваючою комою одинарної точності позначаються F.
- Довжина даних в одному регістрі зазвичай становить 16 біт. Якщо ви хочете зберігати 32-розрядні дані в регістрі, призначте два послідовні регістри для даних.
- Якщо операнд у 32-розрядній інструкції використовує D0, він займає 32-розрядний регістр даних, що складається з (D1, D0). D1 представляє старші 16 бітів, а D0 представляє молодші 16 бітів. Це ж правило стосується таймера і 16-бітного лічильника.
- Коли ви використовуєте 32-розрядний лічильник HC як регістр даних, він може використовуватися лише операндом у 32-розрядній інструкції. Ви можете використовувати тільки індексні регістри в 16-розрядних інструкціях.

Для отримання додаткової інформації зверніться до розділу 2 Пристрої.

Визначені системою операнди

- Система призначає змінні для оголошення, наприклад BOOL, WORD, INT тощо: U0–U16387 та W0– W29999.
- Щоб розпочати або зупинити завдання, використовуйте інструкції TK0–TK31.

У наведеній нижче таблиці наведено символи змінних типів показників, підтримувані пристрої та використання.

Тип показника	Використання	
Загальний показник (Показник)	Діапазон приладу	PR0–PR15 · PR0.0–PR15.15
	Максимальна кількість	Використовуйте до 16 показників у кожному функціональному блоці
	Можна призначити	Змінні символи типів WORD/DWORD/LWORD/INT/DINT/LINT або регістри даних, вхідні реле або вихідні релейні пристрої (наприклад, X0, Y0 тощо)
Показник для таймера (T_POINTER)	Діапазон приладу	TR0–TR7
	Максимальна кількість	Використовуйте до 8 показників у кожному функціональному блоці
	Можна призначити	Змінні символи таймерного типу або пристроїв таймерного типу
Показник для лічильника (C_POINTER)	Діапазон приладу	CR0–CR7
	Максимальна кількість	Використовуйте до 8 показників у кожному функціональному блоці
	Можна призначити	Змінні умовні позначення приладів лічильного типу або лічильного типу
Показник для високошвидкісного лічильника (HC_POINTER)	Діапазон приладу	HCR0–HCR7
	Максимальна кількість	Використовуйте до 8 показників у кожному функціональному блоці
	Можна призначити	Змінні символи 32-розрядного лічильника або пристрої 32-розрядного лічильника

4.3 Обмеження щодо використання інструкцій

- Ви можете використовувати такі інструкції лише у функціональних блоках: API0065 CHKADR, FB_NP, FB_PN, NED, ANED, ONED, PED, APED, OPED
- Ви не можете використовувати таку інструкцію в завданнях переривання: GOEND
- Ви не можете використовувати функціональні блоки: LDP, ANDP, ORP, LDF, ANDF, ORF, PLS, PLF, NP, PN, MC/MCR, GOEND і всі імпульсні інструкції в прикладних інструкціях.

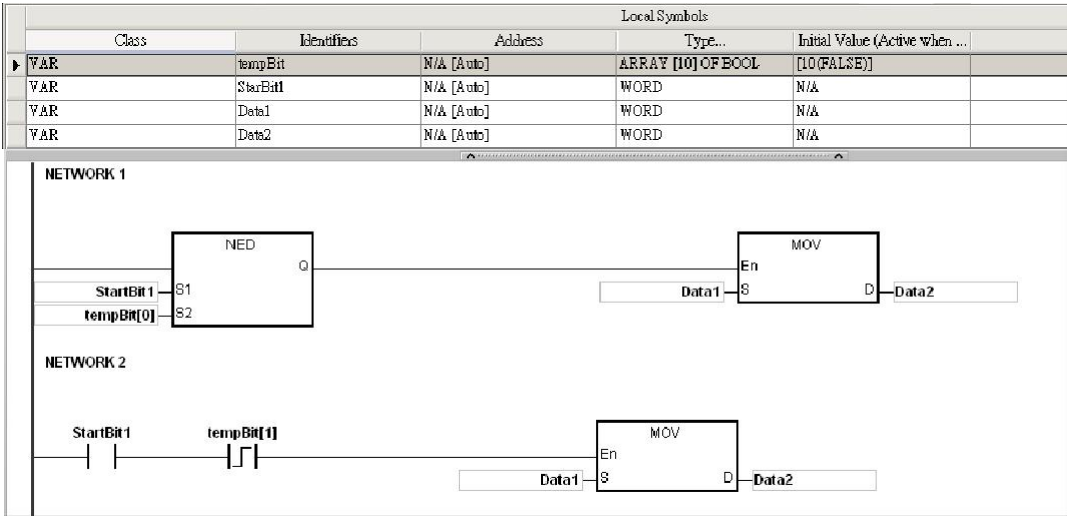
Якщо ви бажаєте використати деякі з інструкцій, згаданих вище, ви можете використовувати заміну інструкцій у наступній таблиці.

Інструкція, яку не можна використовувати у функціональному блоці	Замінити інструкцію у функціональному блоці
LDP/ANDP/ORP	PED/aped/oped
LDF/ANDF/ORF	NED/ANED/ONED
PLS	-
PLF	-
NP	FB_NP
PN	FB_PN
MC	-
MCR	-
Усі імпульсні інструкції в прикладних командах	*1

*1: Імпульсні інструкції не можна використовувати у функціональних блоках. Якщо ви хочете отримати функцію імпульсної інструкції у функціональному блоці, зверніться до наступного прикладу.

приклад:

1. Спочатку оголошите 10-розрядні змінні tempBit[10] для використання в системі.
2. Коли StartBit1 перемикається з OFF на ON, метод 1 (мережа 1) і метод 2 (мережа 2) можуть виконати інструкцію MOV лише один раз; ви можете вибрати, який з них використовувати.
3. Ви не можете використовувати змінну tempBit у системі більше одного разу.



● Використовуйте наступні інструкції з мовою програмування Ladder

Для наступних інструкцій вам потрібно розпочати виконання інструкції, коли ПЛК перемикається з STOP на RUN. І коли наступні інструкції перестають виконуватися, ПЛК також має припинити програмування. Рекомендується використовувати наступні інструкції з мовою програмування Ladder. НЕ використовуйте мову програмування, яка виконує лише один раз, наприклад, функціональні блоки, які можна викликати лише один раз. Або не використовуйте мову програмування, яка припиняє сканування після припинення виконання інструкції, наприклад, мови програмування ST і SFC. Якщо функціональний блок може бути викликаний різними програмами кілька разів, він також не підходить для наступних інструкцій.

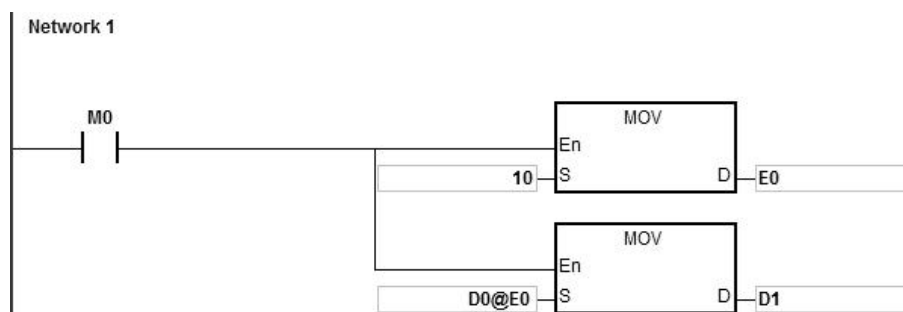
Тип інструкції	Інструкції
Інструкції з таймера та лічильника	CNT, DCNT, DHSCS, DHSCR, DHSZ, DHSCY
Інструкції з периферійних пристроїв	TKY, DTKY, HKY, DHKY, DSW, ARWS
Різна інструкція	RAMP, DRAMP, MTR, INCD
Інструкції модуля	DPUCONF, PUSTAT
Інструкції зі зв'язку CANopen	CANRS, COPRW, COPWL, DCOPWL, ZRNM, ECAMD, ECAMS, ECAMA, ECAMC
Інструкції з реєстрації карти пам'яті/файлу	MEMW
Інструкції щодо спілкування	DNETRW
Інші інструкції	INFO

4.4 Індексні регістри

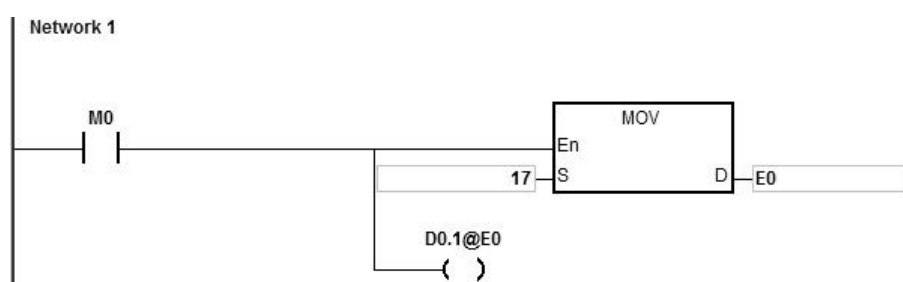
Індексні регістри — це 16-розрядні регістри даних. Вони подібні до загального реєстру тим, що з них можна зчитувати дані та записувати в них дані. Однак вони в основному використовуються як індексні регістри. Діапазон індексних регістрів E0–E9. Не рекомендується використовувати індексні регістри для глобальних змінних; їх можна використовувати лише для часткових змінних і для тимчасових.

Індексні регістри використовуються наступним чином.

1. Використання імені реєстру для зміни пристрою:



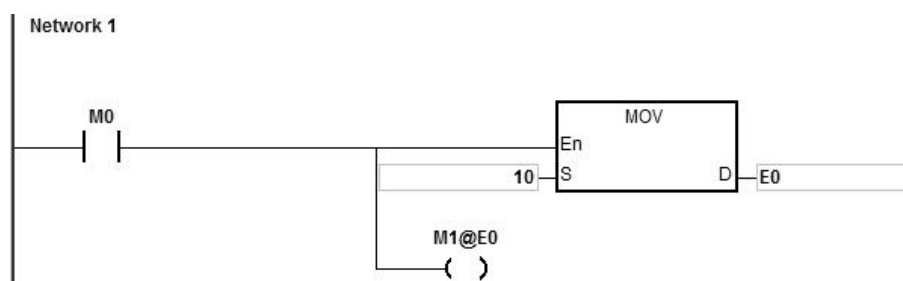
Коли M0 увімкнено, E0=10, E1=17, D1@E0=D (1+10)=D11, D11 увімкнено.



ПРИМІТКА 1. Серія AS підтримує використання імені реєстру для зміни пристрою; наприклад, D0.1@E0, але не підтримує дворівневу модифікацію, наприклад, D0@E1.1@E0.

ПРИМІТКА 2: Коли E0=17, D0.1@E0=D0.(1+17)=D1.2, а D1.2 увімкнено. Розрядна частина 1@E0=(1+17)=18. Однак максимальне число бітів дорівнює 15. Оскільки $m=18/16=1$, а залишок дорівнює 2, результатом останньої модифікації є D (0+1).2=D1.2. D1.2 увімкнено.

Коли M0 увімкнено, E0=10 і M1@E0=M (1+10)=M11. M11 увімкнено.



2. Спочатку оголошення змінних, а потім модифікація пристрою:

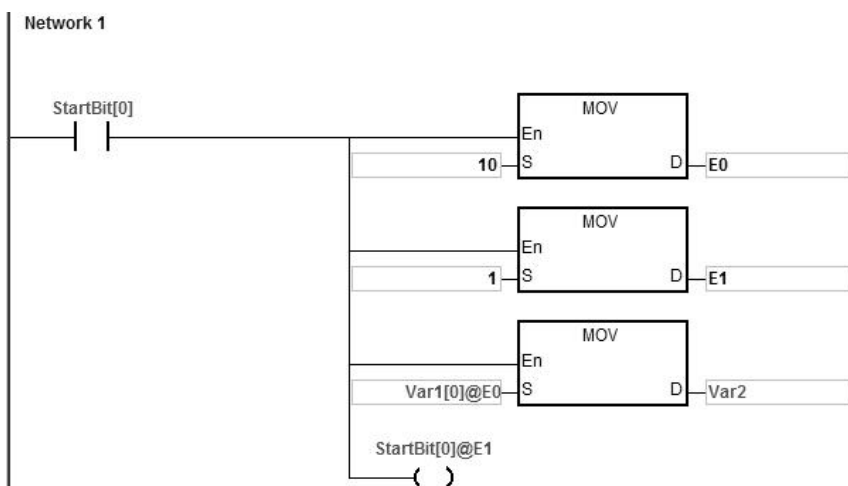
- Оголошіть три змінні StartBit, Var1 і Var2 в ISPSOft.

Тип StartBit — логічний масив, а його розмір — 2 біти. Діапазон від StartBit[0] до StartBit[1]. Тип Var1 - це масив слів, його розмір становить 11 слів. Діапазон від Var1[0] до Var1[10].

Тип Var2 — це слово, а його розмір — одне слово.

Local Symbols					
Class	Identifiers	Address	Type...	Initial Value	Identifier Comment...
VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	N/A	
VAR	Var1	N/A [Auto]	ARRAY [11] OF WORD	N/A	
VAR	Var2	N/A [Auto]	WORD	N/A	

- Коли StartBit[0] увімкнено, E0=10, E1=1, Var1[0]@E0=Var1[10], Var2=Var1[10] і StartBit[0]@E1=StartBit[1]. StartBit[1] увімкнено.

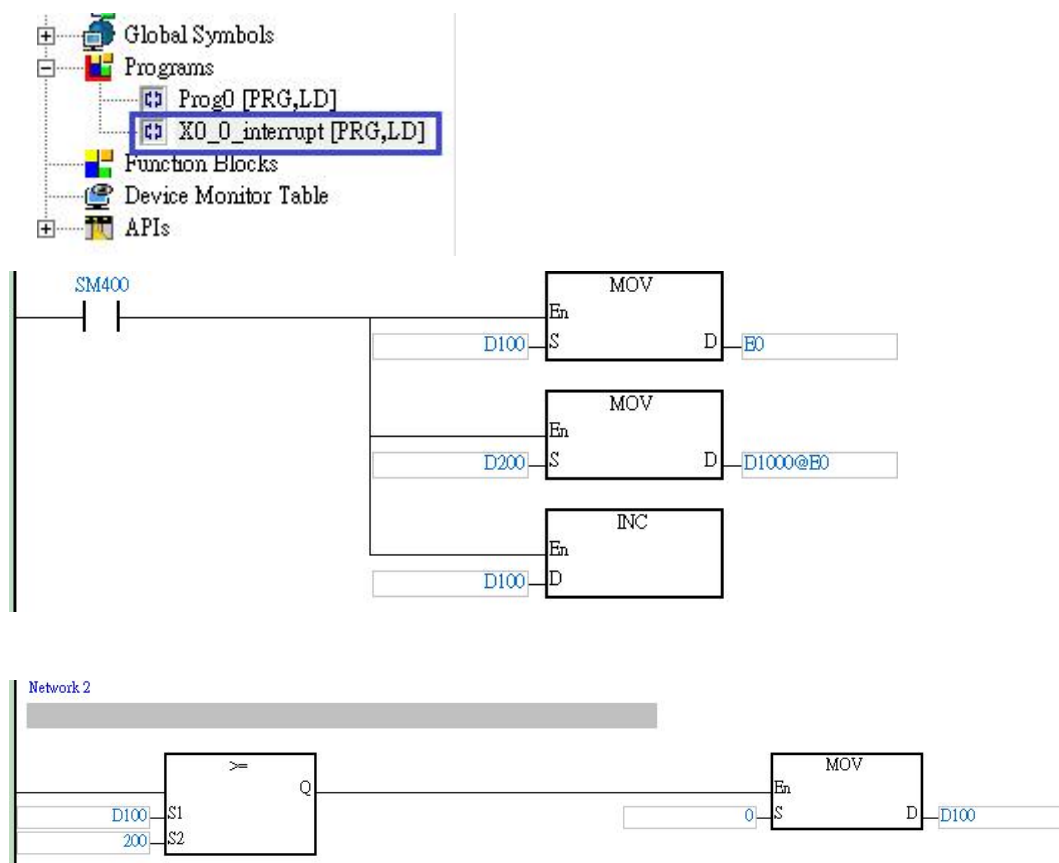


Додаткове зауваження: коли ви оголошуєте змінні в ISPSOft і змінні додаються до вмісту регістрів, щоб сформувати адреси фактичних даних, ви повинні записати адреси, щоб запобігти неправильному виконанню програми.

3. Використання індексних регістрів в інструкції переривання:

- Індексні регістри E0~E9 в основній програмі (наприклад, POU, SFC і FB) відрізняються від індексних регістрів E0~E9 у програмі переривання. Вони є незалежними групами індексних регістрів і працюють окремо.
- Ви можете використовувати ISPSOft для моніторингу значень індексних регістрів у головній програмі.
- Застосування: якщо ви використовуєте E0 в основній програмі та в програмі переривання. Коли виникає переривання, перед виконанням програми переривання система зберігає значення E0 в основній програмі. Після виконання програми переривання значення E0 в основній програмі буде збережено. Таким чином, програма переривання не вплине на значення E0 в основній програмі і результат виконання основної програми може залишитися незмінним.
- Зверніться до наступного прикладу, якщо вам потрібно контролювати значення індексних регістрів E0~E9 у програмі переривання, або якщо основна програма також використовує значення індексних регістрів E0~E9 у програмі переривання. (Примітка: НЕ рекомендується використовувати регістри індексів як глобальні змінні. Якщо вам потрібно використовувати їх як глобальні змінні, вам потрібно оголосити або визначити інші пристрої для використання їх як глобальних змінних.)

- Приклад: Оголосити D100 як змінну індексного регістра E0. І цей приклад використовує інструкцію MOVE для демонстрації. Щоразу, коли зовнішнє переривання виникає на X0.0, значення в D200 накопичує входження в значення в D1000 ~ D1199.



4.5 Регістри показчиків

- ISPSoft підтримує функціональні блоки. Коли тип оголошення змінної VAR_IN_OUT, а тип даних POINTER, змінна є регістром показчика. Значення в регістрі показчика може посилатися безпосередньо на значення, що зберігається в пристрої X, Y або D; а регістр показчика може вказувати на адресу, пов'язану зі змінною, автоматично встановленою в ISPSoft.
- Ви можете оголосити 16 регістрів показчиків у кожному функціональному блоці. Діапазон PR0–PR15 або PR0.0–PR15.15.

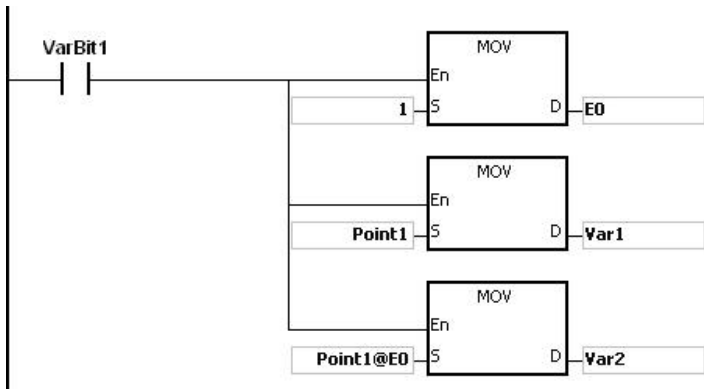
приклад:

Створіть організаційний підрозділ програми (POU) в ISPSoft.

Створіть функціональний блок під назвою FB0.



Створіть програму у функціональному блоці FB0.



Оголошуємо змінну у функціональному блоці FB0.

Виберіть VAR_IN_OUT як тип оголошення, Point1 як ідентифікатор, POINTER як тип даних. Змінна є регістром показчика.

Local Symbols						
	Class	Identifiers	Address	Type...	Initial Value	Identifier Comment...
	VAR	VarBit1	N/A [Auto]	BOOL	FALSE	
	VAR	Var1	N/A [Auto]	WORD	0	
	VAR	Var2	N/A [Auto]	WORD	0	
▶	VAR_IN_OUT	Point1	N/A [Auto]	POINTER	N/A	

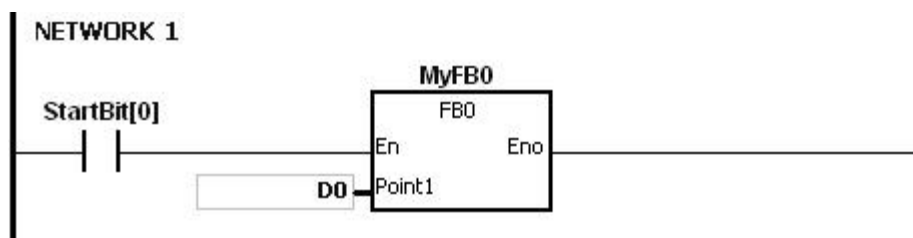
Оголошіть змінну в організаційному підрозділі програми (POU).

Local Symbols						
	Class	Identifiers	Address	Type...	Initial Value	Identifier Comment...
	VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	N/A	
	VAR	CVar1	N/A [Auto]	ARRAY [2] OF WORD	N/A	
▶	VAR	MyFB0	N/A [Auto]	FB0	N/A	

Виклик функціонального блоку FB0 в організаційному блоці програми (POU).

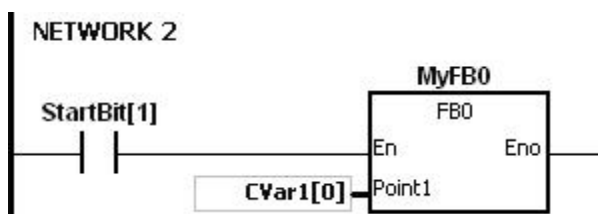
Програма в організаційному підрозділі програми (POU) працює, як показано нижче.

Мережа 1: коли StartBit[0] увімкнено, адреса D0 передається до точки 1 у FB0.



Коли VarBit1 у FB0 увімкнено, $E0=1$, $Var1=D0$, $Point1@E0=D(0+1)=D1$ і $Var2=D1$.

Мережа 2: коли StartBit[1] увімкнено, адреса CVar1[0] передається в Point1 у FB0.



$Var2=CVar1[1]$ • Коли VarBit1 у FB0 увімкнено, $E0=1$, $Var1=CVar1[0]$,
 $Point1@E0=CVar1(0+1)=Cvar1[1]$ і $Var2=CVar1[1]$.

4.6 Регістри показчиків таймерів

- ISPSoft підтримує функціональні блоки. Якщо ви хочете використовувати таймер у функціональному блоці, ви повинні оголосити регістр показчика таймера у функціональному блоці. Адреса таймера передається в регістр показчика таймера при виклику функціонального блоку.
- Коли тип оголошення змінної - VAR_IN_OUT, а тип даних - T_POINTER, змінна є регістром показчика таймера. Значення в регістрі показчика таймера може посилатися безпосередньо на значення, що зберігається в пристрої T, або в змінній, яка є таймером в ISPSoft.
- Ви можете оголосити до 8 регістрів показчиків таймерів у кожному функціональному блоці. Діапазон TR0–TR7.
- Якщо ви хочете використовувати інструкцію у функціональному блоці, а таймер підтримується операндами, ви повинні використовувати регістр показчика таймера.

Приклад: використання таймера у функціональному блоці.

1. Створіть організаційний підрозділ програми (POU) в ISPSoft.
2. Створіть функціональний блок під назвою FB0.

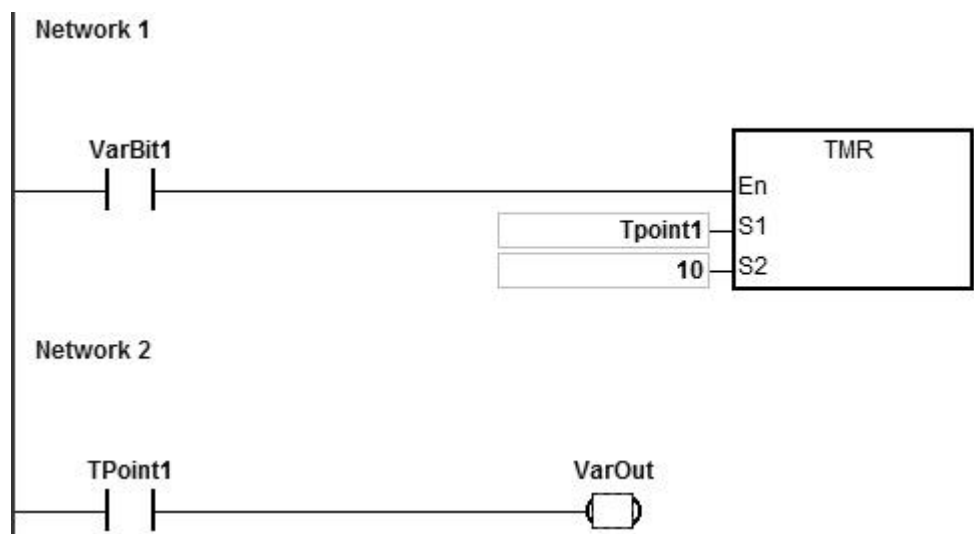


3. Оголосити змінну у функціональному блоці FB0.

Виберіть VAR_IN_OUT як тип оголошення, TPoint1 як ідентифікатор і T_POINTER як тип даних. Змінна є регістром показчика таймера.

Local Symbols						
	Class	Identifiers	Address	Type...	Initial Value	Identifier Comment...
	VAR	VarBit1	N/A [Auto]	BOOL	FALSE	
	VAR_IN_OUT	TPoint1	N/A [Auto]	T_POINTER	N/A	
▶	VAR	VarOut	N/A [Auto]	BOOL	FALSE	

4. Програма у функціональному блоці FB0 виглядає так, як показано нижче.



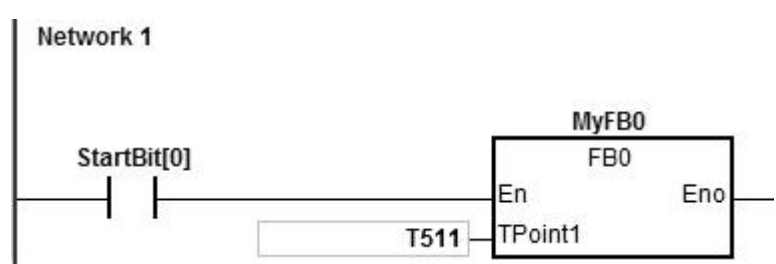
5. Оголошіть змінну в організаційному підрозділі програми (POU).

Тип даних CVar1 має бути TIMER.

	Local Symbols					
	Class	Identifiers	Address	Type...	Initial Value	Identifier Comment...
	VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	[2(FALSE)]	
	VAR	CVar1	T0	TIMER	N/A	
►	VAR	MyFB0	N/A [Auto]	FB0	N/A	

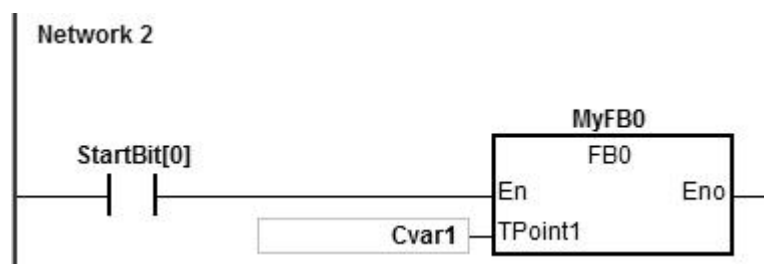
- 6. Виклик функціонального блоку FB0 в організаційному блоці програми (POU).
- 7. Програма в організаційному підрозділі програми (POU) працює, як показано нижче.

Мережа 1: коли StartBit[0] увімкнено, адреса T511 передається до TPoint1 у FB0.



Коли VarBit1 у FB0 увімкнено, інструкція TMR виконується, і TPoint1 (T511) починає підрахунок. Коли значення TPoint1 збігається з налаштуванням TPoint1, VarOut увімкнено.

Мережа 2: коли StartBit[1] увімкнено, адреса CVar1[0] передається до TPoint1 у FB0.



Коли VarBit1 у FB0 увімкнено, інструкція TMR виконується, і TPoint (CVar1) починає підраховувати. Коли значення TPoint1 збігається з налаштуванням TPoint1, VarOut увімкнено.

4.7 Регістри показчиків для 16-розрядних лічильників

- ISPSoft підтримує функціональні блоки. Якщо ви хочете використовувати 16-розрядний лічильник у функціональному блоці, ви повинні оголосити 16-розрядний регістр показчика лічильника у функціональному блоці. Адреса 16-розрядного лічильника передається в регістр показчика 16-розрядного лічильника при виклику функціонального блоку.
- Коли тип оголошення змінної - VAR_IN_OUT, а тип даних - C_POINTER, змінна є 16-бітним регістром показчика лічильника. Значення в 16-бітному регістрі показчика лічильника може посилатися безпосередньо на значення, збережене в пристрої T, або в змінній, яка є лічильником в ISPSoft.
- Ви можете оголосити до восьми 16-розрядних регістрів показчика лічильника в кожному функціональному блоці. Діапазон CR0–CR7.
- Якщо ви хочете використовувати інструкцію у функціональному блоці, і лічильник підтримується операндами, ви повинні використовувати 16-розрядний регістр показчика лічильника.

Приклад: використання 16-бітного лічильника у функціональному блоці.

1. Створіть організаційний підрозділ програми (POU) в ISPSoft.
2. Створіть функціональний блок під назвою FB0.



3. Оголошити змінну у функціональному блоці FB0.

Виберіть VAR_IN_OUT як тип оголошення, CPoint1 як ідентифікатор, C_POINTER як тип даних. Змінна є 16-розрядним регістром показчика лічильника.

Local Symbols					
	Class	Identifiers	Address	Type...	Initial Value
	VAR	VarBit1	N/A [Auto]	BOOL	FALSE
▶	VAR_IN_OUT	CPoint1	N/A [Auto]	C_POINTER	N/A

4. Програма у функціональному блоці FB0 виглядає так, як показано нижче:



5. Оголошіть змінну в організаційному підрозділі програми (POU).

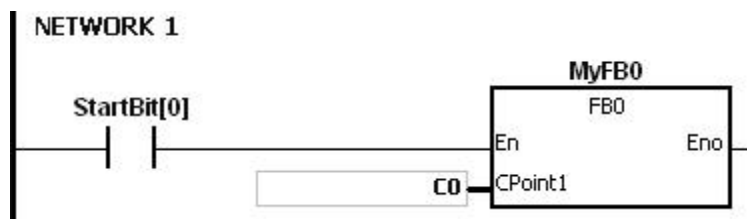
Тип даних CVar1 має бути COUNTER.

Local Symbols					
	Class	Identifiers	Address	Type...	Initial Value
	VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	[2(FALSE)]
	VAR	CVar1	C1	COUNTER	N/A
▶	VAR	MyFB0	N/A [Auto]	FB0	N/A

6. Викличте функціональний блок FB0 в організаційному блоці програми (POU).

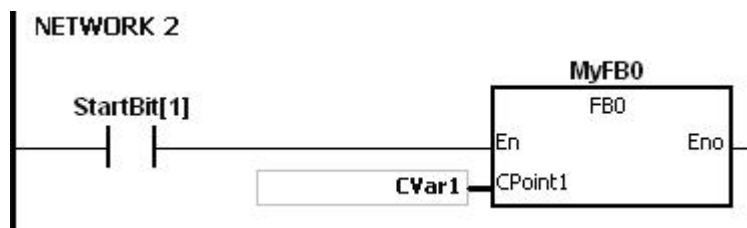
7. Програма в організаційному підрозділі програми (POU) працює, як показано нижче.

Мережа 1: коли StartBit[0] увімкнено, адреса C0 передається до CPoint1 у FB0.



Коли VarBit1 у FB0 увімкнено, CPoint1 (C0) увімкнено.

Мережа 2: коли StartBit[1] увімкнено, адреса CVar1 передається до CPoint1 у FB0.



Коли VarBit1 у FB0 увімкнено, CPoint1 (CVar1) увімкнено.

4.8 Регістри показчиків для 32-розрядних лічильників

- ISPSoft підтримує функціональні блоки. Якщо ви хочете використовувати 32-розрядний лічильник у функціональному блоці, ви повинні оголосити 32-розрядний регістр показчика лічильника у функціональному блоці. Адреса 32-розрядного лічильника передається в регістр показчика 32-розрядного лічильника при виклику функціонального блоку.
- Коли тип оголошення змінної - VAR_IN_OUT, а тип даних - HC_POINTER, змінна є 32-розрядним регістром показчика лічильника. Значення в 32-розрядному регістрі показчика лічильника може посилатися безпосередньо на значення, що зберігається в пристрої HC або в змінній, яка є лічильником в ISPSoft.
- У кожному функціональному блоці можна оголосити до восьми 32-розрядних регістрів показчика лічильника. Діапазон HCR0–HCR7.
- Якщо ви хочете використовувати інструкцію у функціональному блоці, а 32-розрядний лічильник підтримується операндами, ви повинні використовувати 32-розрядний регістр показчика лічильника.

Приклад: використання 32-розрядного лічильника у функціональному блоці.

1. Створіть функціональний блок під назвою FB0.



2. Оголошуємо змінну у функціональному блоці FB0.

Виберіть VAR_IN_OUT як тип оголошення, HCPoint1 як ідентифікатор, HC_POINTER як тип даних. Змінна є 32-розрядним регістром показчика лічильника.

Local Symbols					
	Class	Identifiers	Address	Type...	Initial Value
	VAR	VarBit1	N/A [Auto]	BOOL	FALSE
▶	VAR_IN_OUT	HCPoint1	N/A [Auto]	HC_POINTER	N/A

3. Програма у функціональному блоці FB0 виглядає наступним чином:



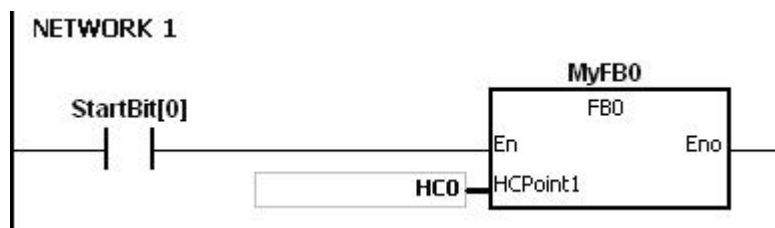
4. Оголошіть змінну в організаційному підрозділі програми (POU).

Тип даних CVar1 має бути COUNTER, і ви повинні заповнити стовпець адреси дійсною адресою 32-розрядного лічильника.

Local Symbols					
	Class	Identifiers	Address	Type...	Initial Value
	VAR	StartBit	N/A [Auto]	ARRAY [2] OF BOOL	[2(FALSE)]
	VAR	CVar1	HC1	COUNTER	N/A
▶	VAR	MyFB0	N/A [Auto]	FB0	N/A

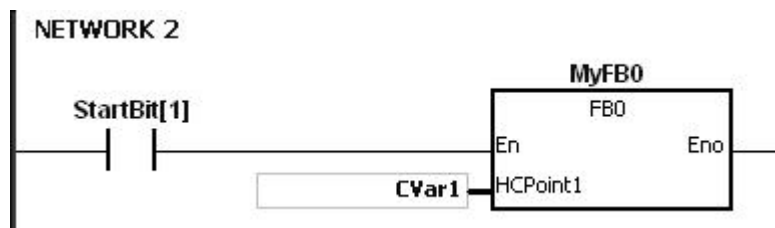
5. Виклик функціонального блоку FB0 в організаційному блоці програми (POU).
6. Програма в організаційному підрозділі програми (POU) працює наступним чином:

Мережа 1: коли StartBit[0] увімкнено, адреса HC0 передається до HCPoint1 у FB0.



Коли VarBit1 у FB0 увімкнено, HCPPoint1 (HC0) увімкнено.

Мережа: коли StartBit[1] увімкнено, адреса CVar1 передається до HCPPoint1 у FB0.



Коли VarBit1 у FB0 увімкнено, HCPPoint1 (CVar1) увімкнено.

4.9 Реєстр файлів

- ПЛК серії AS забезпечує файлові реєстри (FR) для зберігання більшої кількості параметрів.
- Ви можете редагувати, завантажувати та завантажувати параметри в реєстрах файлів за допомогою ISPSOft.
- Значення в реєстрах файлів можна читати під час роботи ПЛК. Зверніться до інструкції MEMW (API 2303) у Посібнику з програмування серії AS300 для отримання додаткової інформації про те, як читати та записувати параметри в реєстри файлів.

Розділ 5 Основні інструкції

Зміст

5.1 Перелік основних інструкцій	5-2
5.2 Основні інструкції	5-3

5.1 Перелік основних інструкцій

У наведеній нижче таблиці наведено основні інструкції, описані в цьому розділі.

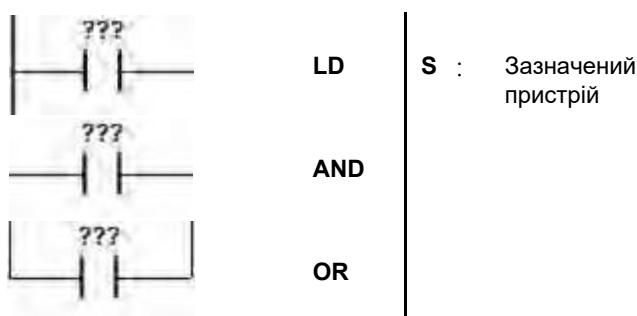
Код інструкції	функція	Операнд	Операція час (мкс)
LD/AND/OR	Навантажувальний контакт A/Послідовне підключення контакту A/Паралельне підключення контакту A	DX, X, Y, M, SM, S, T, C, HC, D	0,025
LDI/ANI/ORI	Навантажувальний контакт B/Послідовне підключення контакту B/Послідовне підключення контакту B	DX, X, Y, M, SM, S, T, C, HC, D	0,03
OUT	Приведення в рух котушки	DY, Y, M, SM, S, T, C, HC, D	0,04
SET	Тримайте пристрій увімкненим	DY, Y, M, SM, S, T, C, HC, D	0,04
MC/MCR	Налаштування/скидання головного керування	N	0,24
LDP/ANDP/ORP	Запуск виявлення наростаючого фронту /Послідовне підключення детектора наростаючого фронту /Підключення детектора наростаючого фронту паралельно	DX, X, Y, M, SM, S, T, C, HC, D	0,22
LDF/ANDF/ORF	Запуск детекції спадаючого фронту/Послідовне підключення детектора спадаючого фронту/Паралельне підключення детектора спадаючого фронту	DX, X, Y, M, SM, S, T, C, HC, D	0,22
PED/APED/OPED	Запуск виявлення наростаючого фронту/Послідовне підключення детектора наростаючого фронту/Підключення детектора наростаючого фронту паралельно	X, Y, M, SM, S, T, C, HC, D	0,22
NED/ANED/ONED	Запуск детекції спадаючого фронту/Послідовне підключення детектора спадаючого фронту/Паралельне підключення детектора спадаючого фронту	X, Y, M, SM, S, T, C, HC, D	0,22
PLS	Вихід по висхідному краю	Y, M, SM, S, D	0,22
PLF	Вихід із спадаючим краєм	Y, M, SM, S, D	0,22
INV	Інвертування результату логічної операції	—	0,22
NP	Схема запускається по висхідному фронту.	—	0,24
PN	Ланцюг спрацьовує по краю.	—	0,24
FB_NP	Схема запускається по висхідному фронту.	Y, M, S, D	0,24
FB_PN	Ланцюг спрацьовує по краю.	Y, M, S, D	0,24

5.2 Основні інструкції

Код інструкції		Операнд		функція							
LD/AND/OR		S		Навантажувальний контакт A/Послідовне підключення контакту A/Паралельне підключення контакту A							
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D
S	●		●	●	●	O	●	●	●	●	●

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

СИМВОЛ

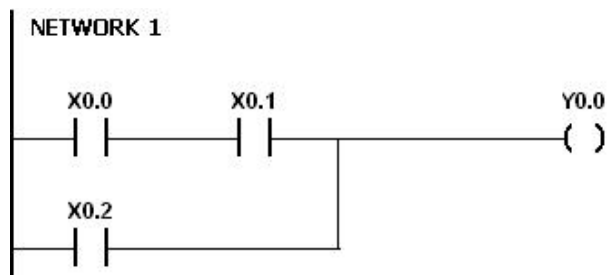


Пояснення

- LD застосовується до контакту A, який починається від основної лінії, або контакту A, який є початком контактної кола. Використовуйте його для збереження поточного вмісту та збереження стану контакту в накопичувальному регістрі.
- AND з'єднує контакт A послідовно. Він зчитує стан контакту, який вказано як з'єднаний послідовно, і виконує операцію AND з результатом попередньої логічної операції. Він зберігає кінцевий результат у накопичувальному регістрі.
- OR з'єднує контакт A паралельно. Він зчитує стан контакту, який вказано як паралельно підключений, і виконує операцію OR з результатом попередньої логічної операції. Він зберігає кінцевий результат у накопичувальному регістрі.

приклад

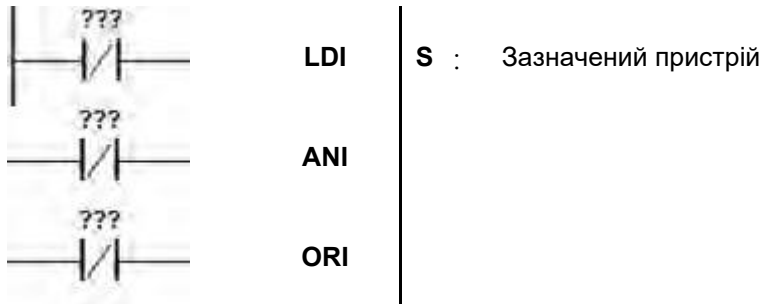
- Контакт A X0.0 навантажувальний, контакт A X0.1 з'єднаний послідовно, контакт A X0.2 з'єднаний паралельно, і котушка Y0.0 приводиться в дію.
- Коли X0.0 і X0.1 увімкнені, або коли X0.2 увімкнено, Y0.0 увімкнено.



Код інструкції		Операнд			функція						
LDI/ANI/ORI		S			Навантажувальний контакт В/Послідовне підключення контакту В/Послідовне підключення контакту В						
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D
S	●		●	●	●	O	●	●	●	●	●

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

СИМВОЛ

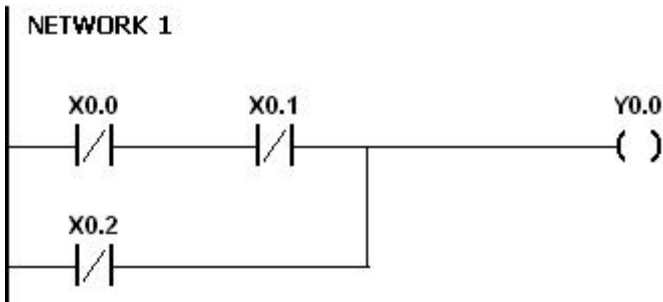


Пояснення

- LDI застосовується до контакту В, який починається від основної лінії, або контакту В, який є початком контактної кола. Використовуйте його для збереження поточного вмісту та збереження стану контакту в накопичувальному регістрі.
- ANI з'єднує контакт В послідовно. Він зчитує стан контакту, який вказано як з'єднаний послідовно, і виконує операцію AND з результатом попередньої логічної операції. Він зберігає кінцевий результат у накопичувальному регістрі.
- ORI з'єднує контакт В паралельно. Він зчитує стан контакту, який вказано як паралельно підключений, і виконує операцію OR з результатом попередньої логічної операції. Він зберігає кінцевий результат у накопичувальному регістрі.

приклад

- Контакт В X0.0 навантажений, контакт В X0.1 з'єднаний послідовно, контакт В X0.2 з'єднаний паралельно, і котушка Y0.0 приводиться в дію.
- Коли X0.0 і X0.1 увімкнені, або коли X0.2 увімкнено, Y0.0 увімкнено.



Код інструкції	Операнд							функція			
OUT	D							Приведення в рух котушки			

пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D
D		●		●	●	O	●				●

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

СИМВОЛ

???
— ()

D : Зазначений пристрій

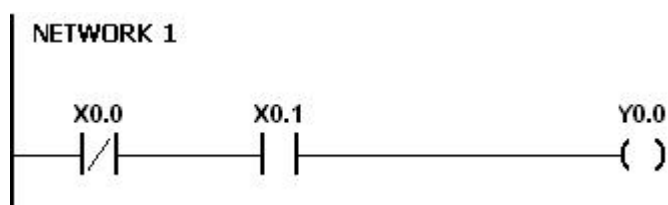
Пояснення

- Результат логічної операції перед застосуванням інструкції OUT виводиться на вказаний пристрій.
- У наступній таблиці описано дію контакту котушки.

Результат операції	OUT		
	Котушка	контакт	
		Контакт А (нормально відкритий)	Контакт В (нормально замкнутий)
помилковий	OFF	OFF	ON
правда	ON	ON	OFF

приклад

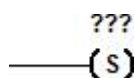
- Контакт В X0.0 навантажується, контакт А X0.1 підключається послідовно, і котушка Y0.0 приводиться в дію.
- Коли X0.0 вимкнено, а X0.1 увімкнено, Y0.0 увімкнено.



Код інструкції		Операнд						функція				
SET		D						Тримайте пристрій увімкненим				
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D	
D		●		●	●	O	●				●	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

СИМВОЛ



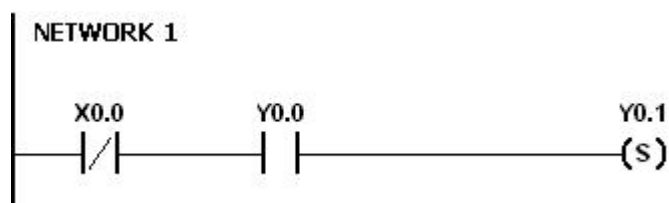
D : Зазначений пристрій

Пояснення

Коли виконується інструкція SET, вказаний пристрій встановлюється на ON. Не має значення, якщо інструкція SET все ще виконується, вказаний пристрій залишається увімкненим. Ви можете вимкнути вказаний пристрій за допомогою інструкції RST.

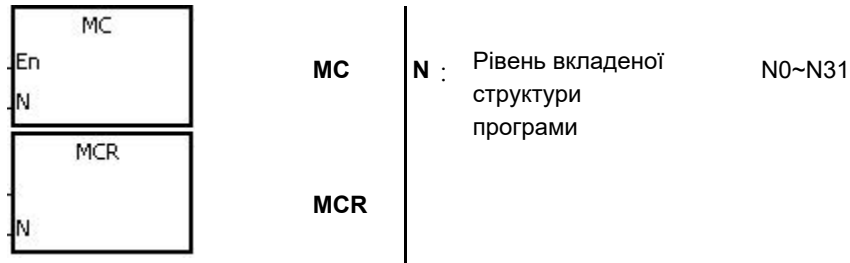
приклад

- Контакт B X0.0 завантажений, контакт A Y0.0 з'єднаний послідовно, а Y0.1 залишається увімкненим.
- Коли X0.0 вимкнено, а Y0.0 увімкнено, Y0.1 увімкнено. Навіть якщо результат операції змінюється, Y0.1 все одно залишається УВІМКНЕНИМ.



Код інструкції	Операнд	функція
MC/MCR	N	Налаштування/скидання головного керування

СИМВОЛ



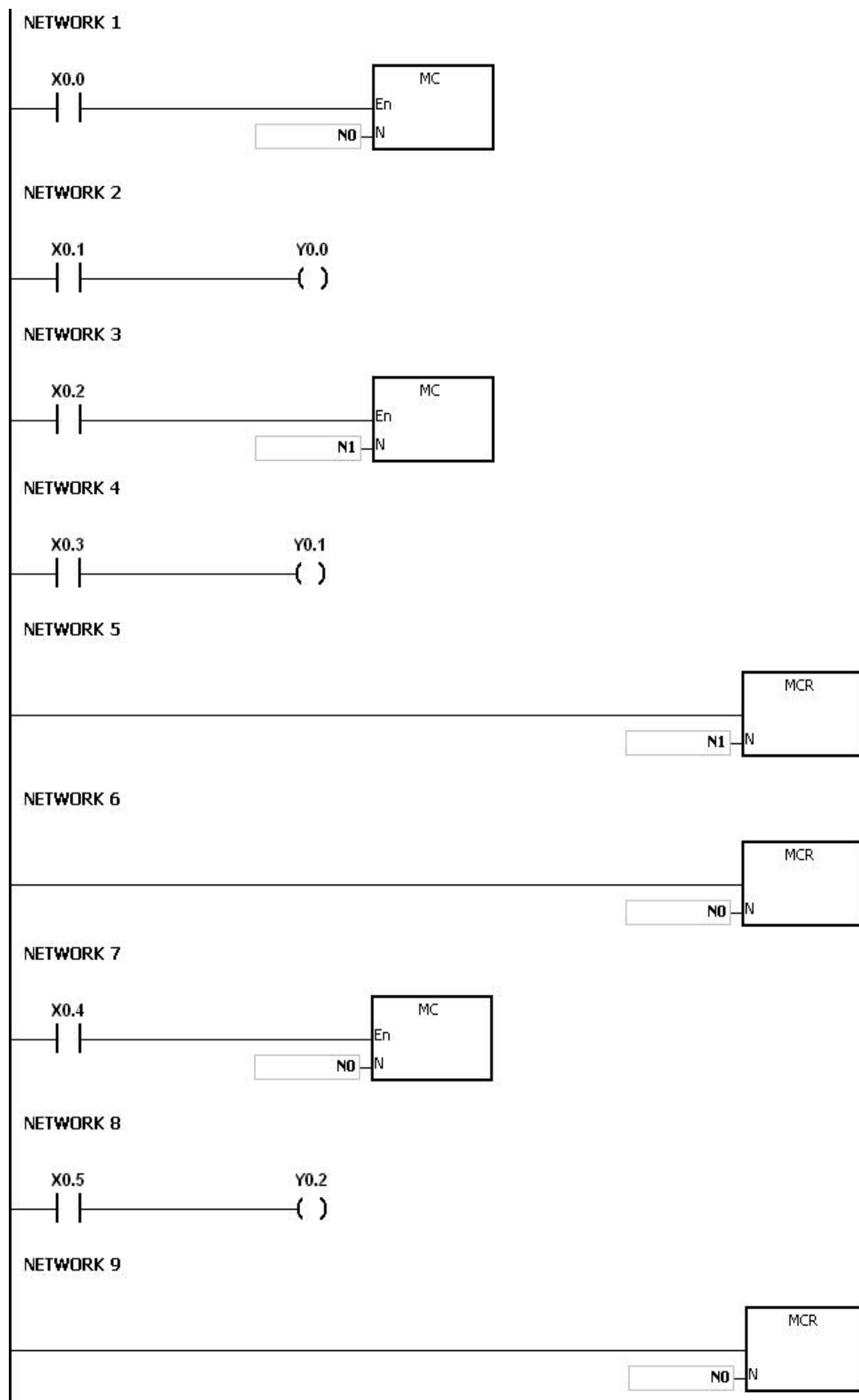
Пояснення

1. MC встановлює основний контроль. Коли виконується інструкція MC, інструкції між MC і MCR виконуються як зазвичай. Коли інструкція MC вимкнена, дії інструкцій між MC і MCR описані в наступній таблиці.

Тип інструкції	опис
Таймер загального призначення	Значення таймера скидається на нуль. Котушка і контакт вимкнені.
Таймер у функціональному блоці	Значення таймера скидається на нуль. Котушка і контакт вимкнені.
Накопичувальний таймер	Котушка вимкнена. Значення таймера та стан контакту залишаються незмінними.
Лічильник	Котушка вимкнена. Значення таймера та стан контакту залишаються незмінними.
Котушки з приводом OUT	Усі котушки вимкнено.
Пристрої, керовані SET і RST	Стани пристроїв залишаються такими ж.
Прикладна інструкція	Усі застосовані інструкції не виконуються. Цикл FOR/NEXT все ще повторюється N разів, але дії інструкцій усередині циклу FOR/NEXT відповідають діям інструкцій між MC і MR.

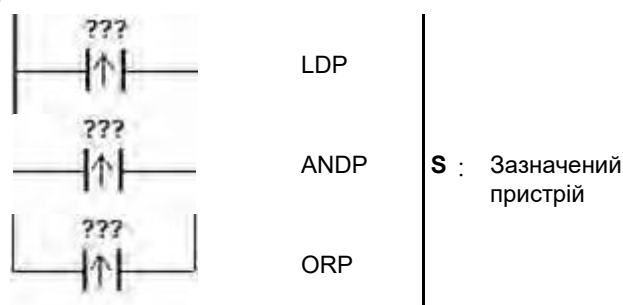
2. MCR скидає головне керування та розміщується в кінці головної програми керування. Перед MCR не повинно бути жодних контактних інструкцій.
3. MC/MCR підтримує структуру вкладки програми. Існує не більше 32 рівнів вкладених програмних структур (N0~N31). Зверніться до прикладу нижче.

приклад



Код інструкції		Операнд		функція							
LDP/ANDP/ORP		S		Запуск виявлення наростаючого фронту /Послідовне підключення детектора наростаючого фронту /Підключення детектора наростаючого фронту паралельно							
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D
S	●		●	●	●	O	●	●	●	●	●

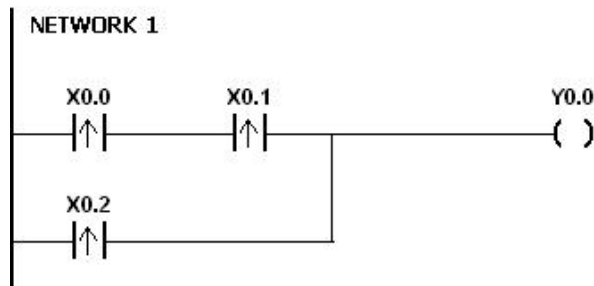
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

СИМВОЛ**Пояснення**

1. LDP зберігає поточний вміст і зберігає виявлення переднього фронту контакту в накопичувальному регістрі.
2. ANDP підключає виявлення переднього фронту контакту послідовно.
3. ORP підключає детектування переднього фронту контакту паралельно.
4. Система повинна сканувати LDP/ANDP/ORP, щоб отримати стан пристрою. Зміни в стані пристрою не виявляються, доки LDP/ANDP/ORP не буде скановано наступного разу.
5. Використовуйте відповідні інструкції PED, APED і OPED у підпрограмах.

приклад

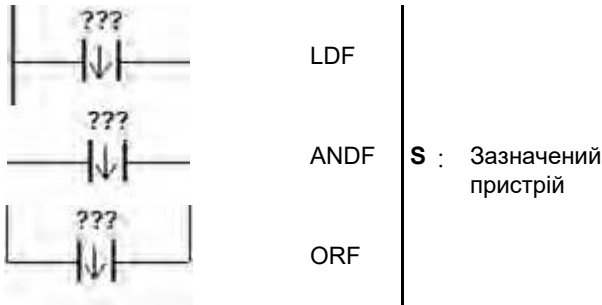
1. Виявлення наростаючого фронту X0.0 починається, виявлення наростаючого фронту X0.1 підключається послідовно, виявлення наростаючого фронту X0.2 підключається паралельно, і котушка Y0.0 приводиться в дію.
2. Коли як X0.0, так і X0.1 перемикаються з OFF. на ON., або коли X0.2 перемикається з OFF. на ON., Y0.0 увімкнено для циклу сканування.



Код інструкції		Операнд			функція						
LDF/ANDF/ORF		S			Запуск детекції спадаючого фронту/Послідовне підключення детектора спадаючого фронту/Паралельне підключення детектора спадаючого фронту						
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D
S	●		●	●	●	O	●	●	●	●	●

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

СИМВОЛ

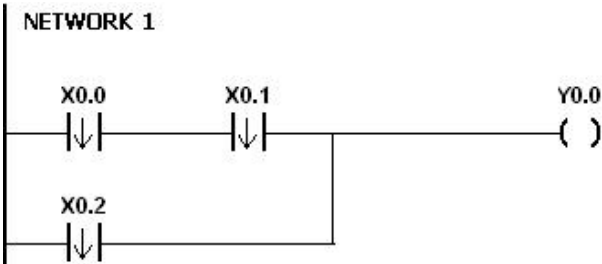


Пояснення

1. LDF зберігає поточний вміст і зберігає виявлення падіння контакту в накопичувальному регістрі.
2. ANDF підключає детекцію спадаючого фронту контакту послідовно.
3. ORF паралельно підключає виявлення спадаючого фронту контакту.
4. Система повинна сканувати LDF/ANDF/ORF, щоб отримати стан пристрою. Зміни в стані пристрою не виявляються, доки LDF/ANDF/ORF не буде скановано наступного разу.
5. Використовуйте відповідні інструкції NED, ANED і ONED у підпрограмах.

приклад

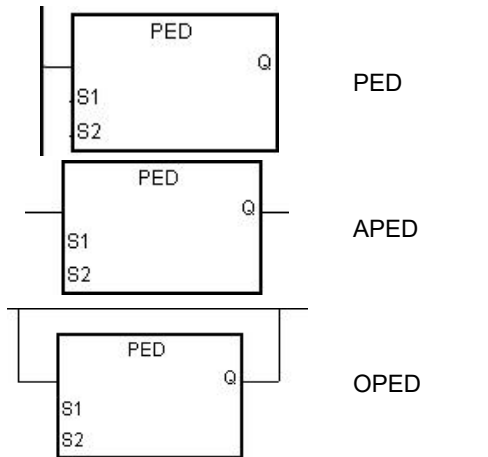
1. Виявлення спадного фронту X0.0 починається, виявлення спадального фронту X0.1 підключається послідовно, виявлення спадального фронту X0.2 підключається паралельно, і котушка Y0.0 приводиться в дію.
2. Коли і X0.0, і X0.1 перемикаються з OFF. на ON., або коли X0.2 перемикається з OFF. на ON., Y0.0 увімкнено для циклу сканування.



Код інструкції		Операнд		функція							
PED/APED/OPED		S_1, S_2		Запуск виявлення наростаючого фронту/Послідовне підключення детектора наростаючого фронту/Підключення детектора наростаючого фронту паралельно							
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D
S_1			●	●	●	○	●	●	●	●	●
S_2				●	●		●				●

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2	●												

СИМВОЛ

 S_1 : Зазначений пристрій S_2 : Для внутрішнього застосування Q : Вивести стан результату операції

Пояснення

1. PED/APED/OPED відповідають LDP/ANDP/ORP. Єдина відмінність між PED/APED/OPED і LDP/ANDP/ORP полягає в тому, що ви повинні вказати бітовий пристрій S_2 , у якому зберігатиметься попередній стан контакту під час виконання PED/APED/OPED. Не використовуйте пристрій S_2 повторно в програмі. Інакше з'являється неправильний результат виконання.

2. APED підключає виявлення переднього фронту контакту послідовно.

3. OPED паралельно підключає виявлення переднього фронту контакту.

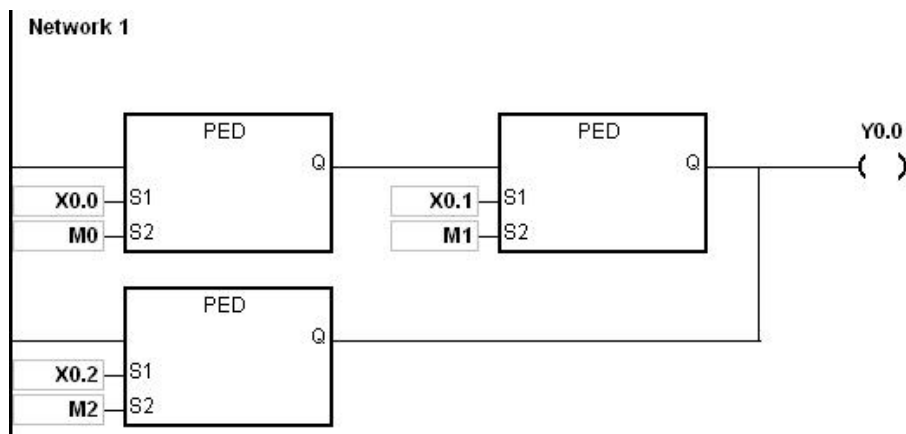
4. Система повинна сканувати PED/APED/OPED, щоб отримати стан пристрою. Зміни в стані пристрою не виявляються, доки PED/APED/OPED не буде скановано наступного разу

5. Ви можете використовувати PED/APED/OPED лише у функціональних блоках.

6. Після виконання інструкції автоматично виводиться стан результату операції. Для цього не потрібно використовувати пристрій введення.

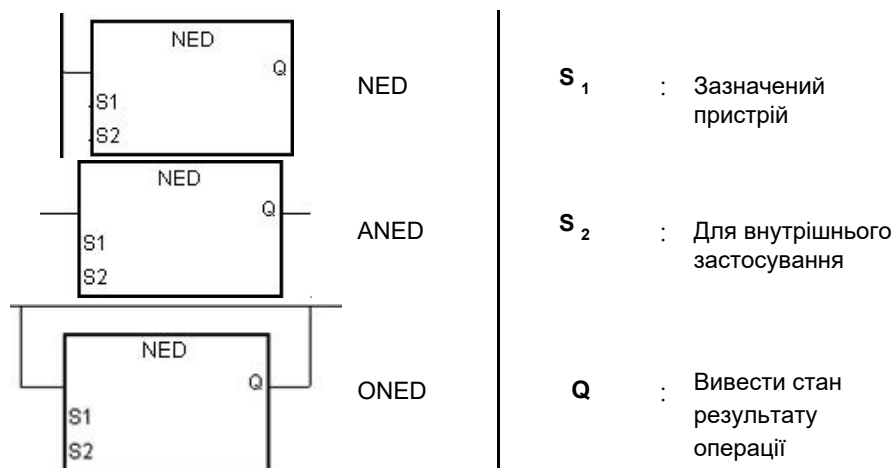
приклад

1. Виявлення наростаючого фронту X0.0 починається, виявлення наростаючого фронту X0.1 підключається послідовно, виявлення наростаючого фронту X0.2 підключається паралельно, і котушка Y0.0 приводиться в дію.
2. Коли і X0.0, і X0.1 перемикаються з OFF. на ON., або коли X0.2 перемикається з OFF. на ON., Y0.0 увімкнено для циклу сканування.



Код інструкції		Операнд					функція						
NED/ANED/ONED		S_1, S_2					Запуск детекції спадаючого фронту/Послідовне підключення детектора спадаючого фронту/Паралельне підключення детектора спадаючого фронту						
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D		
S_1			●	●	●	○	●	●	●	●	●		
S_2				●	●		●				●		
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2	●												

СИМВОЛ

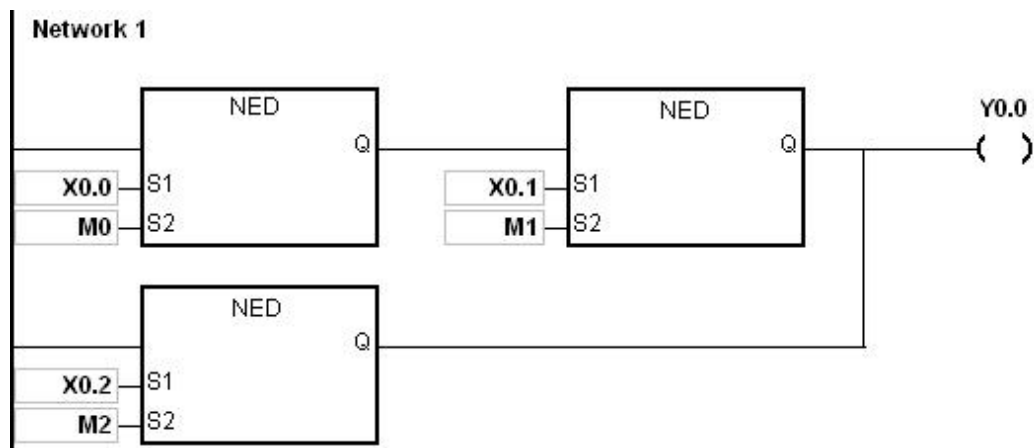


Пояснення

1. NED/ANED/ONED відповідають LDF/ANDF/ORF. Єдина відмінність між NED/ANED/ONED і LDF/ANDF/ORF полягає в тому, що ви повинні вказати бітовий пристрій S_2 , у якому зберігатиметься попередній стан контакту під час виконання NED/ANED/ONED. Не використовуйте пристрій S_2 повторно в програмі. Інакше з'являється неправильний результат виконання.
2. ANED підключає детекцію спадаючого фронту контакту послідовно.
3. ONED паралельно підключає виявлення спадаючого фронту контакту.
4. Система повинна сканувати NED/ANED/ONED, щоб отримати стан пристрою. Зміни в стані пристрою не виявляються, доки NED/ANED/ONED не буде скановано наступного разу
5. NED/ANED/ONED можна використовувати лише у функціональних блоках.
6. Після виконання інструкції автоматично виводиться стан результату операції. Для цього не потрібно використовувати пристрій введення.

приклад

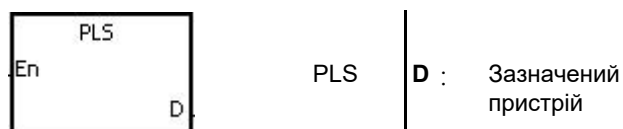
1. Виявлення спадаючого фронту X0.0 починається, детектування спадаючого фронту X0.1 підключається послідовно, детектування спадаючого фронту X0.2 підключається паралельно, і котушка Y0.0 приводиться в дію.
2. Коли і X0.0, і X0.1 перемикаються з OFF. на ON., або коли X0.2 перемикається з OFF. на ON., Y0.0 увімкнено для циклу сканування.



Код інструкції		Операнд						функція			
PLS		D						Вихід по висхідному краю			
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D
D				●	●	O	●				●

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

СИМВОЛ

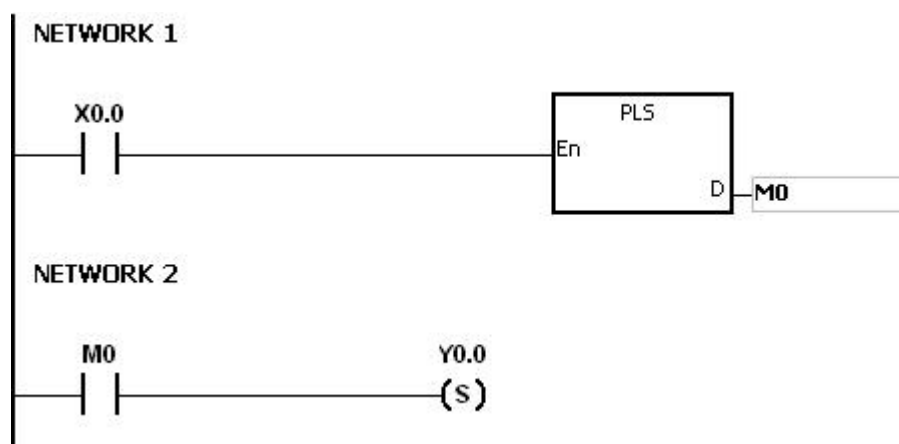


Пояснення

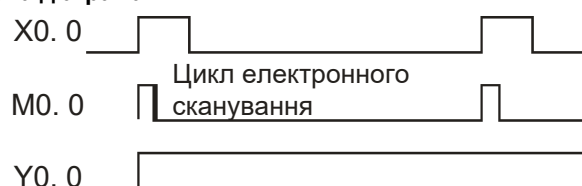
1. Коли умовний контакт перемикається з OFF. на ON., виконується інструкція PLS, і пристрій D надсилає імпульс для циклу сканування.
2. Не використовуйте інструкцію PLS у функціональних блоках.
3. ISPSOft V3.06.01 підтримує тип даних D і змінну типу BOOL для оголошення.

приклад

Коли X0.0 увімкнено, M0 увімкнено на час імпульсу. Коли M0 увімкнено, Y0.0 увімкнено.



Часова діаграма



Код інструкції		Операнд						функція				
PLF		D						Вихід із спадаючим краєм				
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D	
D				●	●	O	●				●	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

СИМВОЛ



Пояснення

- 5
1.

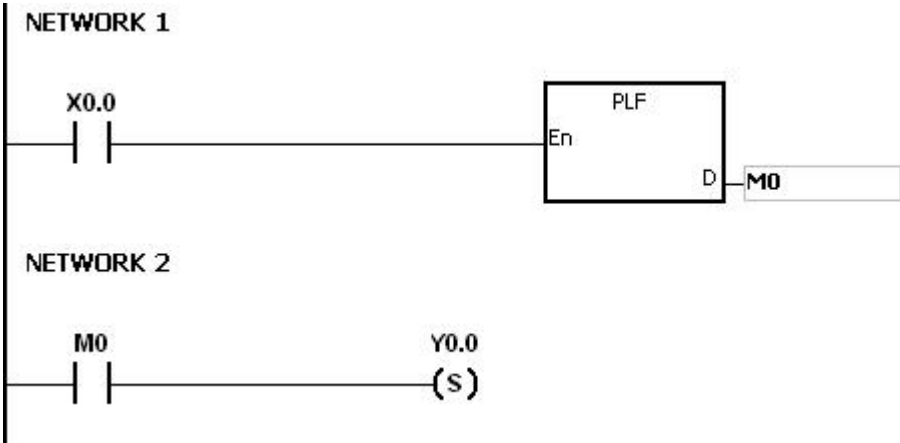
Коли умовний контакт перемикається з ON на OFF, виконується інструкція PLF, і пристрій D надсилає імпульс для циклу сканування.
2.

Не використовуйте інструкцію PLS у функціональних блоках.
3.

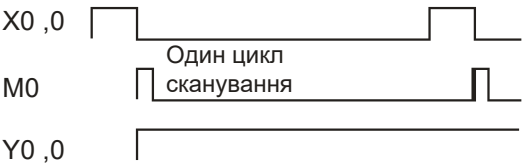
ISPSoft V3.17 підтримує тип даних D і змінну типу BOOL для оголошення.

приклад

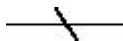
Коли X0.0 увімкнено, M0 увімкнено на час імпульсу. Коли M0 увімкнено, Y0.0 увімкнено.



Графік часу



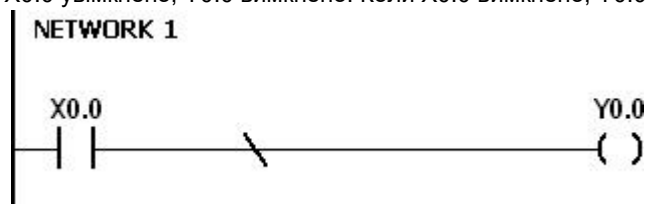
Код інструкції	Операнд	функція
INV	-	Інвертування результату логічної операції

СИМВОЛ**Пояснення**

Результат логічної операції, що передувє інструкції INV, інвертується, а результат інверсії зберігається в накопичувальному регістрі.

приклад

Коли X0.0 увімкнено, Y0.0 вимкнено. Коли X0.0 вимкнено, Y0.0 увімкнено.



Код інструкції	Операнд	функція
NP	-	Спрацьовування схеми по наростаючому фронту.

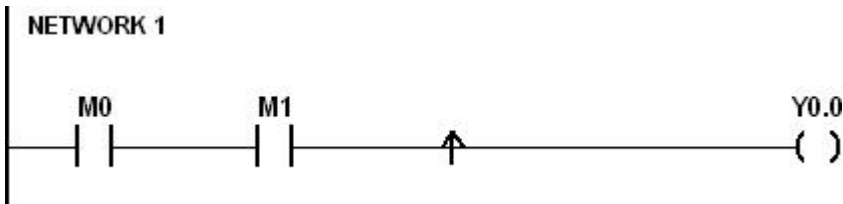
СИМВОЛ



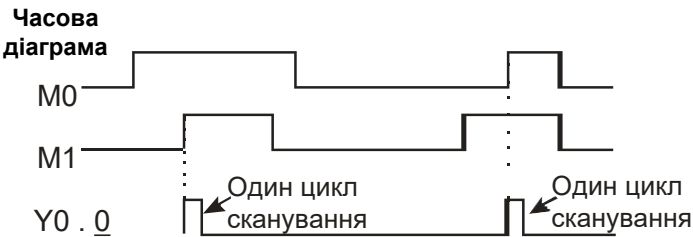
Пояснення

- 1. Коли значення в накопичувальному регістрі змінюється з 0 на 1, інструкція NP зберігає значення 1 в накопичувальному регістрі протягом циклу сканування. Після завершення другого циклу сканування значення в накопичувальному регістрі змінюється на 0.
- 2. Використовуйте інструкцію FB_NP у функціональних блоках.

приклад



Інструкція	Операція
LD M0	Контакт A M0 завантажений.
AND M1	Контакт A M1 з'єднаний послідовно.
NP	Схема запускається по висхідному фронту.
OUT Y0.0	Котушка Y0.0 приводиться в рух.



Код інструкції	Операнд	функція
PN	-	Спрацьовування схеми по спадаючому фронту

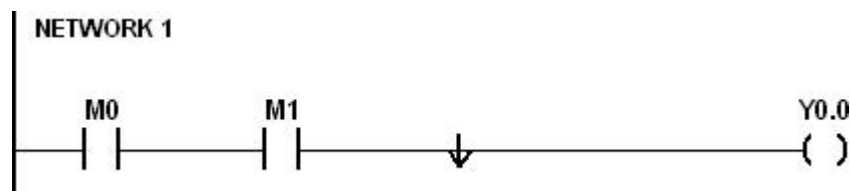
СИМВОЛ



Пояснення

- Коли значення в накопичувальному регістрі змінюється з 1 на 0, інструкція PN зберігає значення 1 в накопичувальному регістрі протягом циклу сканування. Після завершення другого циклу сканування значення в накопичувальному регістрі змінюється на 0.
- Використовуйте інструкцію FB_ PN у функціональних блоках.

приклад



Інструкція

Операція

LD M0

Контакт A M0 завантажений.

AND M1

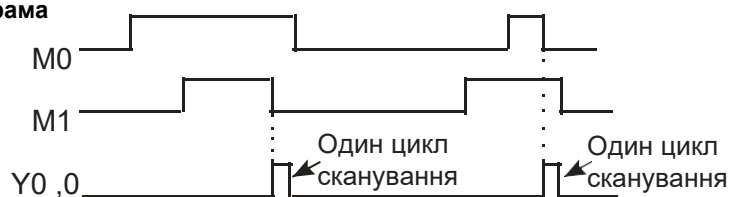
Контакт A M1 з'єднаний послідовно.

PN

Ланцюг спрацьовує по краю.

OUT Y0.0

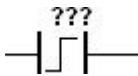
Котушка Y0.0 приводиться в рух.

Часова
діаграма

Код інструкції		Операнд					функція					
FB_NP		S					Спрацьовування схеми по наростаючому фронту.					
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D	
S				●	●		●				●	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

СИМВОЛ

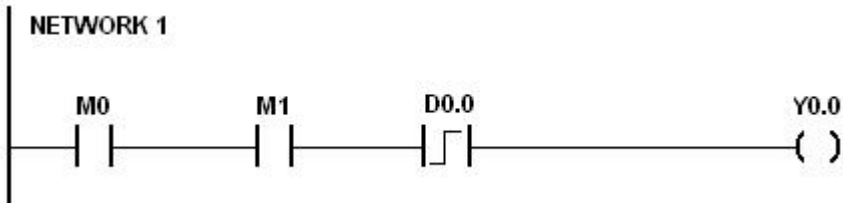


S : Для
внутрішнього
застосування

Пояснення

1. Коли значення в накопичувальному регістрі змінюється з 0 на 1, інструкція FB_NP зберігає значення 1 в накопичувальному регістрі протягом циклу сканування. Після завершення другого циклу сканування значення в накопичувальному регістрі змінюється на 0.
2. Попередній стан контакту зберігається в розрядному пристрої S . Не використовуйте S повторно в програмі. Інакше з'являється неправильний результат виконання.
3. Використовуйте FB_NP лише у функціональних блоках.

приклад



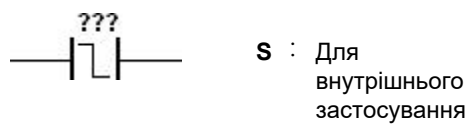
Інструкція		Операція
LD	M0	Контакт A M0 завантажений.
AND	M1	Контакт A M1 з'єднаний послідовно.
FB_NP	D0.0	Схема запускається по висхідному фронту.
OUT	Y0.0	Котушка Y0.0 приводиться в рух.



Код інструкції		Операнд					функція					
FB_PN		S					Спрацьовування схеми по спадаючому фронту.					
пристрій	DX	DY	X	Y	M	SM	S	T	C	HC	D	
S				●	●		●				●	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												

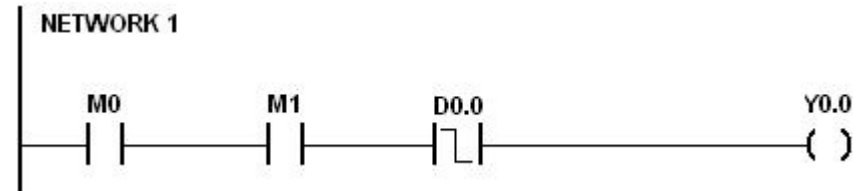
СИМВОЛ



Пояснення

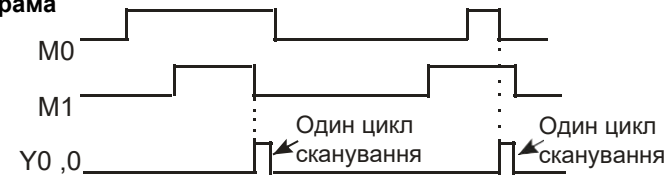
1. Коли значення в накопичувальному регістрі змінюється з 1 на 0, інструкція FB_PN зберігає значення 1 в накопичувальному регістрі протягом циклу сканування. Після завершення другого циклу сканування значення в накопичувальному регістрі змінюється на 0.
2. Попередній стан контакту зберігається в розрядному пристрої **S** . Не використовуйте **S** повторно в програмі. Інакше з'являється неправильний результат виконання.
3. Використовуйте FB_PN лише у функціональних блоках.

приклад



Інструкція		Операція
LD	M0	Контакт A M0 завантажений.
AND	M1	Контакт A M1 з'єднаний послідовно.
FB_PN	D0.0	Ланцюг спрацьовує по краю.
OUT	Y0.0	Котушка Y0.0 приводиться в рух.

Часова діаграма



ПАМ'ЯТКА

Розділ 6 Прикладні інструкції

Зміст

6.1 Інструкції з порівняння	6-4
6.1.1 Список інструкцій для порівняння	6-4
6.1.2 Пояснення інструкцій порівняння.....	6-7
6.2 Арифметичні інструкції	6-46
6.2.1 Список арифметичних інструкцій	6-46
6.2.2 Пояснення арифметичних інструкцій.....	6-47
6.3 Інструкції з перетворення даних	6-78
6.3.1 Перелік інструкцій щодо перетворення даних	6-78
6.3.2 Пояснення вказівок щодо перетворення даних	6-79
6.4 Інструкції з передачі даних	6-122
6.4.1 Перелік інструкцій щодо передачі даних.....	6-122
6.4.2 Пояснення інструкцій щодо передачі даних	6-123
6.5 Інструкції зі стрибків	6-150
6.5.1 Список інструкцій зі стрибків	6-150
6.5.2 Пояснення інструкцій зі стрибка.....	6-151
6.6 Інструкції щодо виконання програми	6-159
6.6.1 Перелік інструкцій щодо виконання програми	6-159
6.6.2 Пояснення інструкцій щодо виконання програми	6-160
6.7 Інструкції з оновлення ІО	6-172
6.7.1 Список інструкцій щодо оновлення ІО	6-172
6.7.2 Пояснення вказівок щодо оновлення ІО	6-173
6.8 Різні інструкції	6-181
6.8.1 Список допоміжних інструкцій.....	6-181
6.8.2 Пояснення зручних інструкцій.....	6-182
6.9 Логічні інструкції	6-235
6.9.1 Список логічних інструкцій	6-235
6.9.2 Пояснення логічних інструкцій.....	6-236
6.10 Інструкції щодо обертання	6-257
6.10.1 Список інструкцій щодо обертання	6-257
6.10.2 Пояснення інструкцій щодо обертання.....	6-258

6.11 Інструкції щодо таймера та лічильника	6-269
6.11.1 Перелік інструкцій таймера та лічильника	6-269
6.11.2 Пояснення інструкцій таймера та лічильника	6-270
6.12 Інструкції щодо зміни	6-317
6.12.1 Перелік інструкцій щодо зміни	6-317
6.12.2 Пояснення вказівок зміни.....	6-318
6.13 Інструкції з обробки даних	6-359
6.13.1 Список інструкцій з обробки даних	6-359
6.13.2 Пояснення інструкцій з обробки даних	6-360
6.14 Інструкції зі створення структури	6-425
6.14.1 Список інструкцій зі створення структури.....	6-425
6.14.2 Пояснення інструкцій зі створення структури	6-426
6.15 Інструкції модуля	6-434
6.15.1 Перелік інструкцій модуля	6-434
6.15.2 Пояснення інструкцій модуля	6-435
6.16 Числа з плаваючою комою Інструкції	6-560
6.16.1 Список інструкцій для чисел з плаваючою комою	6-560
6.16.2 Пояснення інструкцій для чисел з плаваючою комою	6-561
6.17 Інструкції щодо годинника реального часу	6-596
6.17.1 Список інструкцій для годинника реального часу.....	6-596
6.17.2 Пояснення вказівок годинника реального часу	6- 597
6.18 Вказівки периферійних пристроїв	6-628
6.18.1 Перелік периферійних інструкцій	6-628
6.18.2 Пояснення периферійних інструкцій	6-629
6.19 Інструкції щодо зв'язку	6-645
6.19.1 Перелік інструкцій щодо зв'язку.....	6-645
6.19.2 Пояснення інструкцій зв'язку	6-646
6.19.3 Описи пов'язаних із зв'язком прапорів і регістрів	6-733
6.20 Інші інструкції	6-737
6.20.1 Перелік інших інструкцій	6-737
6.20.2 Пояснення до інших інструкцій	6-738
6.21 Інструкції з обробки рядка	6-750
6.21.1 Список інструкцій з обробки рядків	6-750
6.21.2 Пояснення інструкцій з обробки рядків	6-751
6.22 Інструкції Ethernet	6-814
6.22.1 Список інструкцій Ethernet	6-814
6.22.2 Пояснення інструкцій Ethernet	6-815

6.23 Інструкції щодо карти пам'яті	6-878
6.23.1 Перелік інструкцій щодо карти пам'яті	6-878
6.23.2 Пояснення інструкцій щодо карти пам'яті.....	6-879
6.24 Інструкції з керування завданнями	6-904
6.24.1 Перелік інструкцій із керування завданнями	6-904
6.24.2 Пояснення інструкцій з керування завданнями	6-905
6.25 Інструкції SFC	6-909
6.25.1 Перелік інструкцій SFC	6-909
6.25.2 Пояснення інструкцій SFC.....	6-910
6.26 Інструкції щодо високошвидкісного виведення	6-917
6.26.1 Список інструкцій високошвидкісного виводу	6-917
6.26.2 Пояснення інструкцій високошвидкісного виведення	6-918
6.27 Інструкції зі зв'язку Delta CANopen	6-1032
6.27.1 Перелік інструкцій із зв'язку Delta CANopen	6-1032
6.27.2 Пояснення інструкцій зі зв'язку Delta CANopen	6-1033
6.27.3 Поширені запитання в Delta Special CANopen Communication and Troubleshooting	6-1116

6.1 Інструкція порівняння

6.1.1 Список інструкцій для порівняння

У наведеній нижче таблиці наведено інструкції з порівняння, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
0000	LD=	DLD=	–	$S1 = S2$
0001	LD<>	DLD<>	–	$S1 \neq S2$
0002	LD>	DLD>	–	$S1 > S2$
0003	LD>=	DLD>=	–	$S1 \geq S2$
0004	LD<	DLD<	–	$S1 < S2$
0005	LD<=	DLD<=	–	$S1 \leq S2$
0006	AND=	DAND=	–	$S1 = S2$
0007	AND<>	DAND<>	–	$S1 \neq S2$
0008	AND>	DAND>	–	$S1 > S2$
0009	AND>=	DAND>=	–	$S1 \geq S2$
0010	AND<	DAND<	–	$S1 < S2$
0011	AND<=	DAND<=	–	$S1 \leq S2$
0012	OR=	DOR=	–	$S1 = S2$
0013	OR<>	DOR<>	–	$S1 \neq S2$
0014	OR>	DOR>	–	$S1 > S2$
0015	OR>=	DOR>=	–	$S1 \geq S2$
0016	OR<	DOR<	–	$S1 < S2$
0017	OR<=	DOR<=	–	$S1 \leq S2$
0018	–	FLD=	–	$S1 = S2$
0019	–	FLD<>	–	$S1 \neq S2$
0020	–	FLD>	–	$S1 > S2$
0021	–	FLD>=	–	$S1 \geq S2$
0022	–	FLD<	–	$S1 < S2$
0023	–	FLD<=	–	$S1 \leq S2$
0024	–	FAND=	–	$S1 = S2$
0025	–	FAND<>	–	$S1 \neq S2$
0026	–	FAND>	–	$S1 > S2$
0027	–	FAND>=	–	$S1 \geq S2$
0028	–	FAND<	–	$S1 < S2$
0029	–	FAND<=	–	$S1 \leq S2$
0030	–	FOR=	–	$S1 = S2$
0031	–	FOR<>	–	$S1 \neq S2$

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
0032	–	FOR>	–	$S1 > S2$
0033	–	FOR>=	–	$S1 \geq S2$
0034	–	FOR<	–	$S1 < S2$
0035	–	FOR<=	–	$S1 \leq S2$
0036	LD\$=	–	–	$S1 = S2$
0037	LD\$<>	–	–	$S1 \neq S2$
0042	AND\$=	–	–	$S1 = S2$
0043	AND\$<>	–	–	$S1 \neq S2$
0048	OR\$=	–	–	$S1 = S2$
0049	OR\$<>	–	–	$S1 \neq S2$
0054	CMP	DCMP	✓	Порівняння значень
0055	ZCP	DZCP	✓	Порівняння зон
0056	–	FCMP	✓	Порівняння чисел з плаваючою комою
0057	–	FZCP	✓	Порівняння зон із плаваючою комою
0058	MCMP	–	✓	Матричне порівняння
0059	CMPT=	–	✓	Порівняння таблиць ПО: =
0060	CMPT<>	–	✓	Порівняння таблиць НА: <>
0061	CMPT>	–	✓	Порівняння таблиць ПО: >
0062	CMPT>=	–	✓	Порівняння таблиць ON: $\geq$
0063	CMPT<	–	✓	Порівняння таблиць ПО: <
0064	CMPT<=	–	✓	Порівняння таблиць ON: $\leq$
0065	CHKADR	–	–	Перевірка адреси в реєстрі показчиків
0066	LDZ=	DLDZ=	–	$ S1-S2 = S3 $
0067	LDZ<>	DLDZ<>	–	$ S1-S2 \neq S3 $
0068	LDZ>	DLDZ>	–	$ S1-S2 > S3 $
0069	LDZ>=	DLDZ>=	–	$ S1-S2 \geq S3 $
0070	LDZ<	DLDZ<	–	$ S1-S2 < S3 $
0071	LDZ<=	DLDZ<=	–	$ S1-S2 \leq S3 $
0072	ANDZ=	DANDZ=	–	$ S1-S2 = S3 $
0073	ANDZ<>	DANDZ<>	–	$ S1-S2 \neq S3 $
0074	ANDZ>	DANDZ>	–	$ S1-S2 > S3 $
0075	ANDZ>=	DANDZ>=	–	$ S1-S2 \geq S3 $

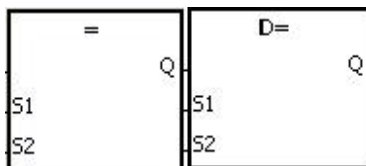
API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32- розрядний		
0076	ANDZ<	DANDZ<	–	$ S1-S2 < S3 $
0077	ANDZ<=	DANDZ<=	–	$ S1-S2 \leq S3 $
0078	ORZ=	DORZ=	–	$ S1-S2 = S3 $
0079	ORZ<>	DORZ<>	–	$ S1-S2 \neq S3 $
0080	ORZ>	DORZ>	–	$ S1-S2 > S3 $
0081	ORZ>=	DORZ>=	–	$ S1-S2 \geq S3 $
0082	ORZ<	DORZ<	–	$ S1-S2 < S3 $
0083	ORZ<=	DORZ<=	–	$ S1-S2 \leq S3 $

6.1.2 Пояснення інструкцій порівняння

API		Код інструкції			Операнд							функція					
0000-0005		D	LD ※			S ₁ , S ₂							Порівняння значень LD ※				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●	●	●	●		○	○	○	○			
S ₂	●	●			●	●	●	●	●		○	○	○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

СИМВОЛ



Наприклад, LD= і DLD=

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

 S_1 : Джерело даних 1 S_2 : Джерело даних 1

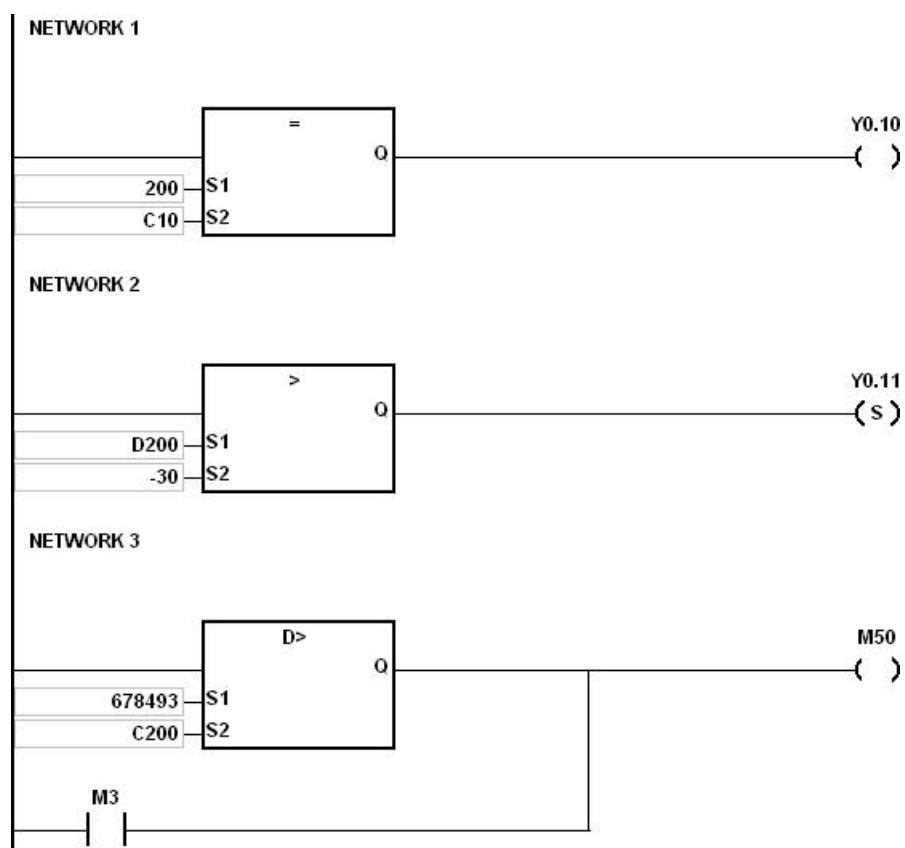
Пояснення

- Ці інструкції порівнюють значення в S_1 і S_2 . Візьмемо, наприклад, інструкцію LD=. Коли порівняння результат полягає в тому, що значення в S_1 дорівнює значенню в S_2 , умова безперервності інструкції виконується. Коли результатом порівняння є те, що значення в S_1 не дорівнює значенню в S_2 , умова розриву інструкції виконується.
- Тільки 32-бітна інструкція може використовувати 32-бітний лічильник, але не пристрій E.

номер API	16-бітна інструкція	32-розрядна інструкція	Умова безперервності	Умова розриву
0000	LD =	D LD =	$S_1 = S_2$	$S_1 \neq S_2$
0001	LD < >	D LD < >	$S_1 \neq S_2$	$S_1 = S_2$
0002	LD >	D LD >	$S_1 > S_2$	$S_1 \leq S_2$
0003	LD > =	D LD > =	$S_1 \geq S_2$	$S_1 < S_2$
0004	LD <	D LD <	$S_1 < S_2$	$S_1 \geq S_2$
0005	LD < =	D LD < =	$S_1 \leq S_2$	$S_1 > S_2$

приклад

- Коли значення в C10 дорівнює 200, Y0.10 увімкнено.
- Коли значення в D200 перевищує -30, Y0.11 залишається УВІМКНЕНИМ.
- Якщо значення в (C201, C200) менше 678 493 або коли M3 увімкнено, M50 увімкнено.

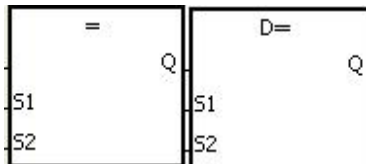


API		Код інструкції			Операнд							функція					
0006-0011		D	AND ※			S ₁ , S ₂							Порівняння значень AND ※				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●	●	●	●		○	○	○	○			
S ₂	●	●			●	●	●	●	●		○	○	○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ



Наприклад, AND= і DAND=

 S_1 : Джерело даних 1 S_2 : Джерело даних 2

Пояснення

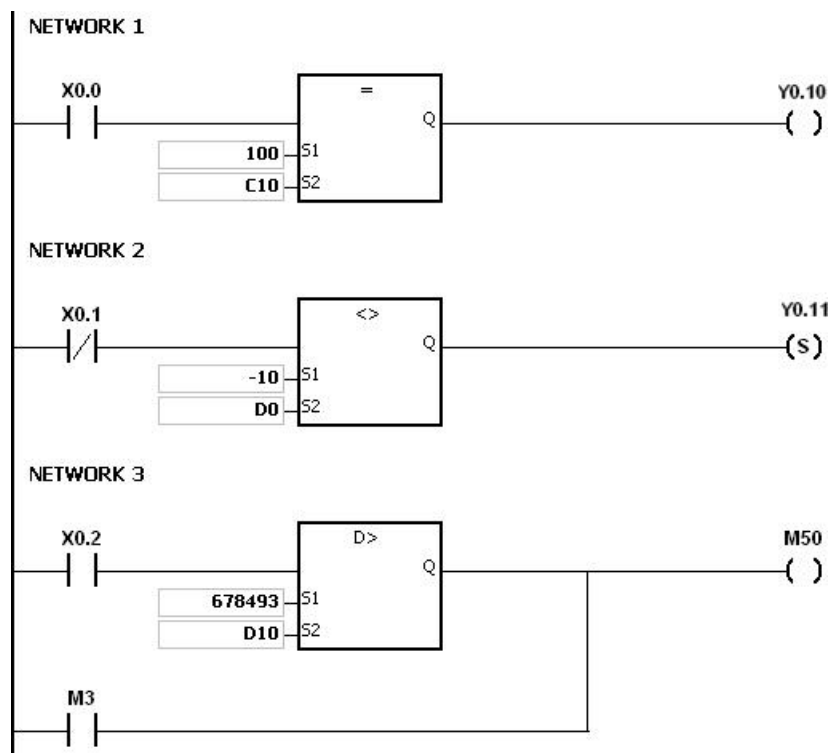
1. Ці інструкції порівнюють значення в S_1 і S_2 . Візьмемо, наприклад, інструкцію AND=. Коли результатом порівняння є те, що значення в S_1 дорівнює значенню в S_2 , умова безперервності інструкції виконується. Коли результатом порівняння є те, що значення в S_1 не дорівнює значенню в S_2 , умова розриву інструкції виконується.

2. Тільки 32-бітна інструкція може використовувати 32-бітний лічильник, але не пристрій E.

номер API	16-бітна інструкція	32-розрядна інструкція	Умова безперервності	Умова розриву
0006	AND =	D AND =	$S_1 = S_2$	$S_1 \neq S_2$
0007	AND < >	D AND < >	$S_1 \neq S_2$	$S_1 = S_2$
0008	AND >	D AND >	$S_1 > S_2$	$S_1 \leq S_2$
0009	AND > =	D AND > =	$S_1 \geq S_2$	$S_1 < S_2$
0010	AND <	D AND <	$S_1 < S_2$	$S_1 \geq S_2$
0011	AND < =	D AND < =	$S_1 \leq S_2$	$S_1 > S_2$

приклад

- Коли X0.0 увімкнено, а поточне значення в C10 дорівнює 100, Y0.10 увімкнено.
- Коли X0.1 вимкнено, а значення в D0 не дорівнює -10, Y0.11 залишається увімкненим.
- Коли X0.2 увімкнено, а значення в (D11, D10) менше 678 493, або коли M3 увімкнено, M50 увімкнено.

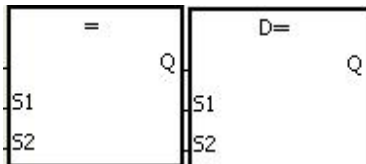


API	Код інструкції				Операнд							функція					
0012-0017	D	OR ※			S_1, S_2							Порівняння значень OR ※					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●	●	●	●		O	O	O	O			
S_2	●	●			●	●	●	●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ

 S_1 : Джерело даних 1 S_2 : Джерело даних 2

Наприклад, OR= і DOR=

Пояснення

1. Ці інструкції порівнюють значення в S_1 і S_2 . Візьмемо, наприклад, інструкцію OR=. Коли результатом порівняння є те, що значення в S_1 дорівнює значенню в S_2 , умова безперервності інструкції виконується. Коли результатом порівняння є те, що значення в S_1 не дорівнює значенню в S_2 , умова розриву інструкції виконується.

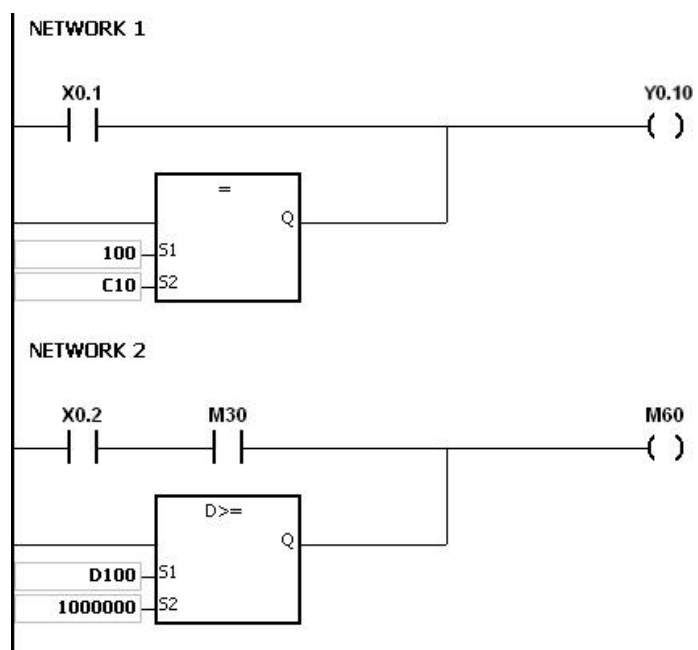
2. Тільки 32-бітна інструкція може використовувати 32-бітний лічильник, але не пристрій E.

номер API	16-бітна інструкція	32-розрядна інструкція	Умова безперервності	Умова розриву
0012	OR =	D OR =	$S_1 = S_2$	$S_1 \neq S_2$
0013	OR < >	D OR < >	$S_1 \neq S_2$	$S_1 = S_2$
0014	OR >	D OR >	$S_1 > S_2$	$S_1 \leq S_2$
0015	OR > =	D OR > =	$S_1 \geq S_2$	$S_1 < S_2$
0016	OR <	D OR <	$S_1 < S_2$	$S_1 \geq S_2$
0017	OR < =	D OR < =	$S_1 \leq S_2$	$S_1 > S_2$

приклад

1. Коли X0.1 увімкнено або коли поточне значення в C10 дорівнює 100, Y0.10 увімкнено.

2. Якщо як X0.2, так і M30 увімкнені, або коли значення в (D101, D100) більше або дорівнює 1000 000, M60 увімкнено.



API	Код інструкції				Операнд							функція				
0018-0023		FLD ※			S ₁ , S ₂							Порівняння чисел з плаваючою комою LD ※				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		O					O
S ₂	●	●			●	●	●	●	●		O					O
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING			
S ₁									●							
S ₂									●							
СИМВОЛ	Інструкція по пульсу							16-бітна інструкція				32-розрядна інструкція				
	-							-				AS				



Наприклад, FLD=

 S_1 : Джерело даних 1 S_2 : Джерело даних 2

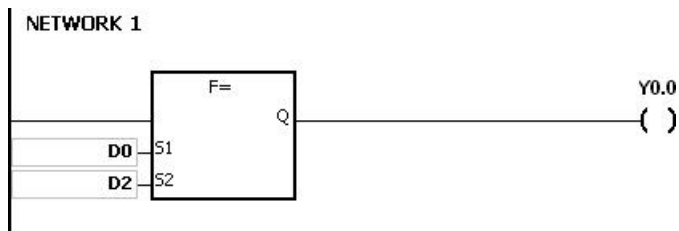
Пояснення

- Ці інструкції порівнюють 32-бітове число з плаваючою комою одинарної точності
- Порівняйте значення в S_1 і S_2 . Візьмемо, наприклад, інструкцію FLD=. Коли результатом порівняння є те, що значення в S_1 дорівнює значенню в S_2 , умова безперервності інструкції виконується. Коли результатом порівняння є те, що значення в S_1 не дорівнює значенню в S_2 , умова розриву інструкції виконується. Щоб отримати додаткові відомості щодо інструкції FLD=, зверніться до пункту 2 додаткових зауважень нижче.

номер API	32-розрядна інструкція	Умова безперервності	Умова розриву
0018	FLD =	$S_1 = S_2$	$S_1 \neq S_2$
0019	FLD < >	$S_1 \neq S_2$	$S_1 = S_2$
0020	FLD >	$S_1 > S_2$	$S_1 \leq S_2$
0021	FLD > =	$S_1 \geq S_2$	$S_1 < S_2$
0022	FLD <	$S_1 < S_2$	$S_1 \geq S_2$
0023	FLD < =	$S_1 \leq S_2$	$S_1 > S_2$

приклад

для прикладу інструкцію $FLD =$. Коли значення в D0 дорівнює значенню в D2, Y0.0 увімкнено.



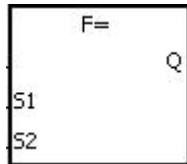
Додаткові зауваження

1. Якщо значення в S_1 або S_2 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, контакт вимкнено, SM увімкнено, а код помилки в SR0 — 16#2013.
2. Якщо значення з плаваючою комою в S_1 або S_2 отримано із зовнішнього введення або від ISPSOft, швидше за все, може виникнути незначна різниця значень і результат інструкції «FLD=» може бути ненадійним. Використовуйте інструкції «FLD<» і «FAND>» або «FLD>» і «FAND<», якщо значення з плаваючою комою в S_1 або S_2 надходить із зовнішнього введення або від ISPSOft.

API		Код інструкції				Операнд						функція						
0024-0029			FAND ※				S ₁ , S ₂						Порівняння чисел з плаваючою комою AND ※					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S ₁	●	●			●	●	●	●	●		○					○		
S ₂	●	●			●	●	●	●	●		○					○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				

СИМВОЛ



Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

 S_1 : Джерело даних 1 S_2 : Джерело даних 2

Наприклад, FAND= і DFAND=

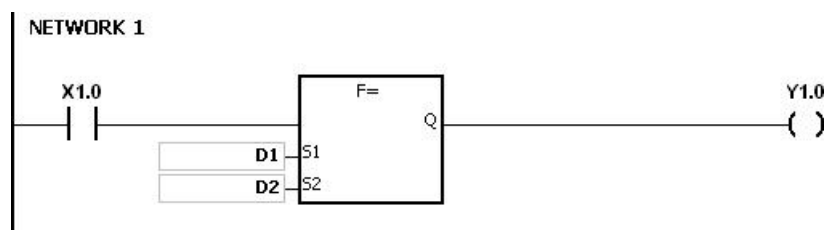
Пояснення

- Ці інструкції порівнюють 32-розрядні числа з плаваючою комою одинарної точності
- Порівняйте значення в S_1 і S_2 . Візьмемо, наприклад, інструкцію FAND=. Коли результатом порівняння є те, що значення в S_1 дорівнює значенню в S_2 , умова безперервності інструкції виконується. Коли результатом порівняння є те, що значення в S_1 не дорівнює значенню в S_2 , умова розриву інструкції виконується.

номер API	32-розрядна інструкція	Умова безперервності	Умова розриву
0024	FAND =	$S_1 = S_2$	$S_1 \neq S_2$
0025	FAND < >	$S_1 \neq S_2$	$S_1 = S_2$
0026	FAND >	$S_1 > S_2$	$S_1 \leq S_2$
0027	FAND > =	$S_1 \geq S_2$	$S_1 < S_2$
0028	FAND <	$S_1 < S_2$	$S_1 \geq S_2$
0029	FAND < =	$S_1 \leq S_2$	$S_1 > S_2$

приклад

наприклад, інструкцію FAND = . Коли X1.0 увімкнено, а значення в D1 дорівнює значенню в D2, Y1.0 увімкнено.



Додаткові зауваження

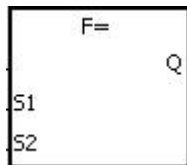
1. Якщо значення в S_1 або S_2 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, контакт вимкнено, SM увімкнено, а код помилки в SR0 — 16#2013.

API		Код інструкції				Операнд						функція					
0030-0035			FOR ※			S ₁ , S ₂						Порівняння чисел з плаваючою комою OR ※					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●	●	●	●		○					○	
S ₂	●	●			●	●	●	●	●		○					○	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ

 S_1 : Джерело даних1 S_2 : Джерело даних2

Взявши, наприклад, FOR= і DFOR=

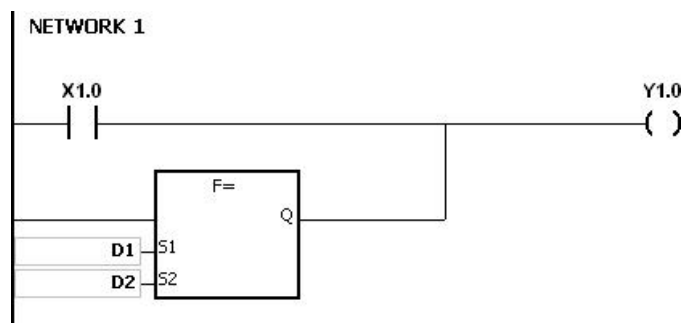
Пояснення

- Ці інструкції порівнюють 32-розрядні числа з плаваючою комою одинарної точності.
- Порівняйте значення в S_1 і S_2 . Візьміть, наприклад, інструкцію FOR=. Коли результатом порівняння є те, що значення в S_1 дорівнює значенню в S_2 , умова безперервності інструкції виконується. Коли результатом порівняння є те, що значення в S_1 не дорівнює значенню в S_2 , умова розриву інструкції виконується.

номер API	32-розрядна інструкція	Умова безперервності	Умова розриву
0030	FOR =	$S_1 = S_2$	$S_1 \neq S_2$
0031	HA < >	$S_1 \neq S_2$	$S_1 = S_2$
0032	FOR >	$S_1 > S_2$	$S_1 \leq S_2$
0033	FOR > =	$S_1 \geq S_2$	$S_1 < S_2$
0034	FOR <	$S_1 < S_2$	$S_1 \geq S_2$
0035	FOR < =	$S_1 \leq S_2$	$S_1 > S_2$

приклад

Коли X1.0 увімкнено або коли значення в D1 дорівнює значенню в D2, Y1.0 увімкнено.



Додаткові зауваження

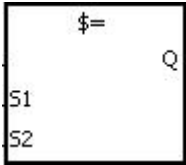
1. Якщо значення в S_1 або S_2 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, контакт вимкнено, SM увімкнено, а код помилки в SR0 — 16#2013.

API		Код інструкції				Операнд						функція					
0036-0037			LD\$ ※			S ₁ , S ₂						Порівняння рядків LD\$ ※					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
	S ₁	●	●			●	●		●	●					○		
	S ₂	●	●			●	●		●	●					○		

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁													●
S ₂													●

СИМВОЛ	Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
	-	AS	-

СИМВОЛ



S₁ : Джерело даних1

S₂ : Джерело даних2

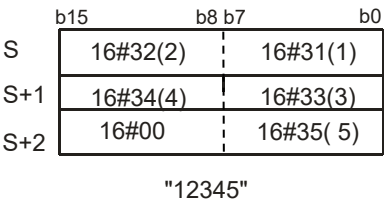
Взявши, наприклад, LD\$=

Пояснення

- 1. Ці інструкції порівнюють рядки в S₁ і S₂.
- 2. S₁ і S₂ можуть містити рядки до 256 символів (16#00 символ кінця включено).
- 3. Візьмемо, наприклад, інструкцію LD\$=. Коли результатом порівняння є те, що значення в S₁ дорівнює значенню в S₂, умова безперервності інструкції виконується. Коли результатом порівняння є те, що значення в S₁ не дорівнює значенню в S₂, умова розриву інструкції виконується.

номер API	16-бітна інструкція	Умова безперервності	Умова розриву
0036	LD\$ =	S ₁ = S ₂	S ₁ ≠ S ₂
0037	LD\$ < >	S ₁ ≠ S ₂	S ₁ = S ₂

- 4. Тільки якщо дані в S – S + n (n вказує на n- й пристрій, до 256 символів у кожному рядку) включають 16#00, дані можна порівнювати повними рядками. Наприклад:



5. Якщо два рядки однакові, відповідні результати порівняння інструкцій наведено нижче. Наприклад:

б 15 б8 б7 б0 б 15 б8 б7 б0

S1

S1 +1

S1 +2

16#42(B)	:	16#41(A)
16#44(D)	:	16#43(C)
16#00	:	16#45(E)

"ABCDE"

Знак порівняння S2

S2 +1

S2 +2

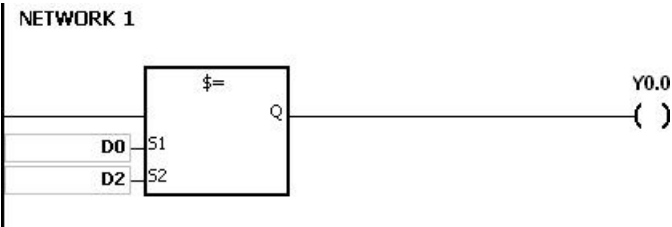
16#42(B)	:	16#41(A)
16#44(D)	:	16#43(C)
16#00	:	16#45(E)

"ABCDE"

Символ порівняння	Результат операції порівняння
\$ =	Безперервність
\$ < >	Розривність

приклад

Коли рядок, що починається з даних у D0-16#00, дорівнює рядку, що починається з D2-16#00, Y0.0 увімкнено.



Додаткові зауваження

- 6
1.

Якщо рядок містить більше 256 символів або рядок не закінчується на 16#00, інструкція не виконується, SM увімкнено, а код помилки в SR0 — 16#200E.
2.

Під час порівняння рядка рядок закінчується, коли знайдено символ кінця 16#00. Символ 16#00 визначає довжину рядка.

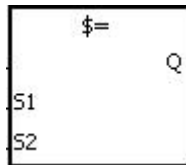
API	Код інструкції			Операнд							функція						
0042-0043		AND\$ ※		S_1, S_2							Порівняння рядків AND\$ ※						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●						O	
S_2	●	●			●	●		●	●						O	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													●
S_2													●

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

 S_1 : Джерело даних1 S_2 : Джерело даних2

Наприклад, AND\$=

Пояснення

- Ці інструкції порівнюють рядки в S_1 і S_2 .
- S_1 і S_2 можуть містити рядок довжиною до 256 символів (16#00 включає символ кінця).
- Візьмемо, наприклад, інструкцію AND\$=. Коли результатом порівняння є те, що значення в S_1 дорівнює значенню в S_2 , виконується умова безперервності інструкції. Коли результатом порівняння є те, що значення в S_1 не дорівнює значенню в S_2 , умова розриву інструкції виконується.

номер API	16-бітна інструкція	Умова безперервності	Умова розриву
0042	AND \$ =	$S_1 = S_2$	$S_1 \neq S_2$
0043	AND \$ < >	$S_1 \neq S_2$	$S_1 = S_2$

Лише коли дані в $S - S + n$ (n вказує на n-й пристрій, до 256 символів у кожному рядку) містять 16#00, інструкція може порівняти дані як повні рядки. Наприклад:

	b 15	b8	b7	b0
S	16#32(2)		16#31(1)	
S+1	16#34(4)		16#33(3)	
S+2	16#00		16#35(5)	

"12345"

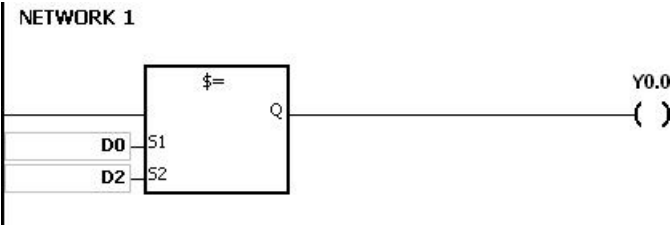
4. Якщо два рядки однакові, відповідні результати операції порівняння інструкцій наведено нижче.
Наприклад:



Символ порівняння	Результат операції порівняння
\$ =	Безперервність
\$ < >	Розривність

приклад

Коли рядок, що починається з даних у D0-16#00, дорівнює рядку, що починається з D2-16#00, Y0.0 увімкнено.



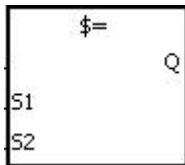
Додаткові зауваження

- 1. Якщо рядок містить більше 256 символів або рядок не закінчується на 16#00, інструкція не виконується, SM увімкнено, а код помилки в SR0 — 16#200E.
- 2. Під час порівняння рядка рядок закінчується, коли знайдено символ кінця 16#00. Символ 16#00 визначає довжину рядка.

API		Код інструкції			Операнд						функція					
0048-0049			OR \$ ※		S ₁ , S ₂						Порівняння рядків OR\$ ※					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●						○	
S ₂	●	●			●	●		●	●						○	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													●
S_2													●

СИМВОЛ



Наприклад, OR\$=

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

 S_1 : Джерело даних1 S_2 : Джерело даних2

Пояснення

- Ці інструкції порівнюють рядки в S_1 і S_2 .
- S_1 і S_2 можуть містити до 256 символів (16#00 символ кінця включено).
- Візьмемо, наприклад, інструкцію OR\$=. Коли результатом порівняння є те, що значення в S_1 дорівнює значенню в S_2 , умова безперервності інструкції виконана. Коли результатом порівняння є те, що значення в S_1 не дорівнює значенню в S_2 , умова розриву інструкції виконується.

номер API	16-бітна інструкція	Умова безперервності	Умова розриву
0048	OR \$ =	$S_1 = S_2$	$S_1 \neq S_2$
0049	OR < > \$	$S_1 \neq S_2$	$S_1 = S_2$

Лише коли дані в $S - S + n$ (n вказує на n-й пристрій, до 256 символів у кожному рядку) містять 16#00, інструкція може порівняти дані як повні рядки. Наприклад:

	b 15	b8 b7	b0
S	16#32(2)	⋮	16#31(1)
S+1	16#34(4)	⋮	16#33(3)
S+2	16#00	⋮	16#35(5)

"12345"

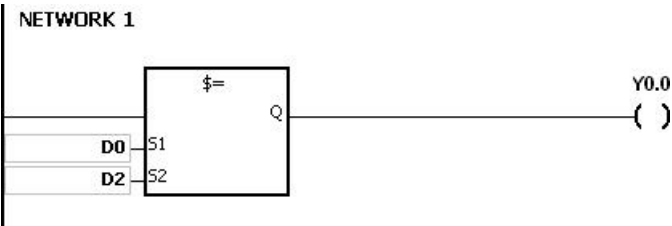
4. Якщо два рядки однакові, відповідні результати операції порівняння інструкцій наведено нижче.
Наприклад:



Символ порівняння	Результат операції порівняння
\$ =	Безперервність
\$ < >	Розривність

приклад

Коли рядок, що починається з даних у D0-16#00, дорівнює рядку, що починається з D2-16#00, Y0.0 увімкнено.



6

Додаткові зауваження

1. Якщо рядок містить більше 256 символів або рядок не закінчується на 16#00, інструкція не виконується, SM увімкнено, а код помилки в SR0 — 16#200E.
2. Під час порівняння рядка рядок закінчується, коли знайдено символ кінця 16#00. Символ 16#00 визначає довжину рядка.

API	Код інструкції				Операнд						функція					
0054	D	CMP		P	S ₁ , S ₂ , D						Порівняння значень					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		○	○	○	○		
S ₂	●	●			●	●	●	●	●		○	○	○	○		
D		●	●	●				●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D	●												

СИМВОЛ

CMP		CMPP	
En		En	
S1	D	S1	D
S2		S2	

DCMP		DCMPP	
En		En	
S1	D	S1	D
S2		S2	

- S1** : Порівняльна величина1
- S2** : Порівняльна величина2
- D** : Результат порівняння

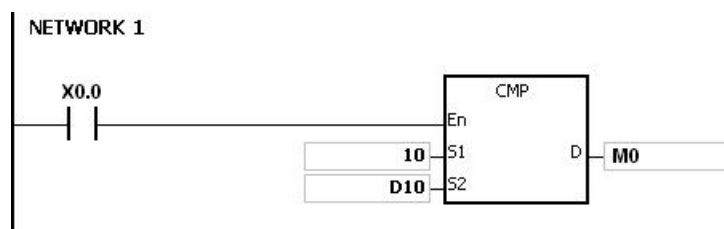
Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

Пояснення

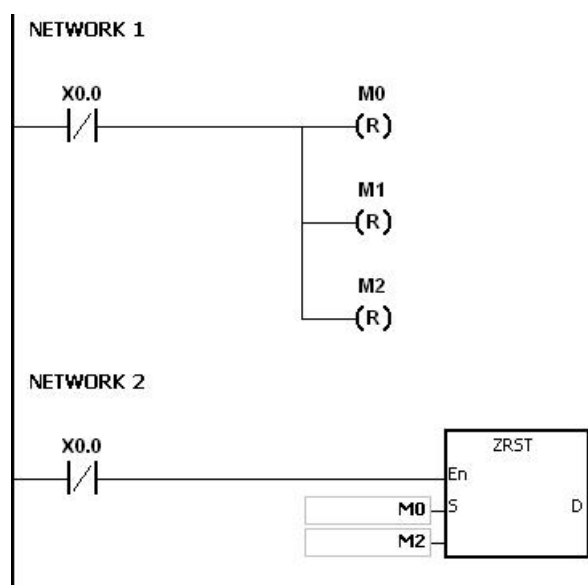
- Ці інструкції порівнюють одиничні десяткові числа в S_1 і S_2 і зберігають результати порівняння в D .
- Операнд D займає 3 послідовні пристрої. Результати порівняння зберігаються в D , $D+1$ і $D+2$. Якщо значення в S_1 більше, ніж значення в S_2 , D увімкнено. Якщо значення в S_1 дорівнює значенню в S_2 , $D+1$ увімкнено. Якщо значення в S_1 менше значення в S_2 , $D+2$ увімкнено.
- Тільки інструкції DCMP і DCMPP можуть використовувати 32-бітний лічильник, але не пристрій E.

приклад

- Якщо операндом D є M0, результати порівняння зберігаються в M0, M1 і M2, як показано нижче.
- Коли X0.0 увімкнено, виконується інструкція CMP. M0, M1 або M2 увімкнено. Коли X0.0 вимкнено, виконання інструкції CMP припиняється, а стан M0, стан M1 і стан M2 залишаються незмінними.



3. Якщо потрібно очистити результат порівняння, використовуйте інструкцію RST або ZRST.



Додаткові зауваження

1. Якщо ви оголошуєте операнд **D** в ISPSOft, типом даних буде ARRAY [3] BOOL.
2. Якщо **D +2** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції			Операнд							функція						
0055	D	ZCP	P	S_1, S_2, S, D							Порівняння зон						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		O	O	O	O		
S_2	●	●			●	●	●	●	●		O	O	O	O		
S	●	●			●	●	●	●	●		O	O	O	O		
D		●	●	●				●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
S		●	●			●	●				●	●	
D	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

ZCP	ZCPP
En	En
S1	S1
S2	S2
S	S
D	D

DZCP	DZCPP
En	En
S1	S1
S2	S2
S	S
D	D

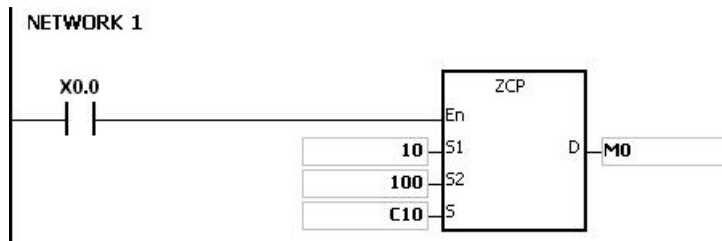
- S1** : Мінімальне значення зони порівняння
- S2** : Максимальне значення зони порівняння
- S** : Значення порівняння
- D** : Результат порівняння

Пояснення

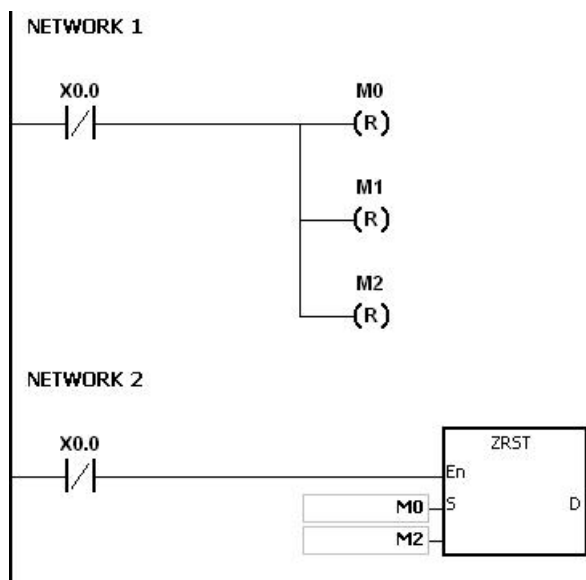
- Ці інструкції порівнюють десяткові числа зі знаком у S і S_1 , а також порівнюють десяткові числа зі знаком у S і S_2 , і зберігає результати порівняння в D .
- Значення в S_1 має бути менше, ніж у S_2 . Якщо значення в S_1 більше, ніж у S_2 , S_1 є максимальним/мінімальним значенням під час виконання інструкції ZCP.
- Операнд D займає три послідовних пристроїв. Результати порівняння зберігаються в D , $D+1$ і $D+2$. Якщо значення в S_1 менше значення в S , D увімкнено. Якщо значення в S знаходиться між значеннями в S_1 і S_2 , $D+1$ увімкнено. Якщо значення в S більше, ніж значення в S_2 , $D+2$ увімкнено.
- Тільки інструкції DZCP і DZCPP можуть використовувати 32-розрядний лічильник, але не пристрій E.

приклад

1. Якщо операндом **D** є M0, результати порівняння зберігаються в M0, M1 і M2, як показано нижче.
2. Коли X0.0 увімкнено, виконується інструкція ZCP. M0, M1 або M2 увімкнено. Коли X0.0 вимкнено, інструкція ZCP не виконується, а стан M0, стан M1 і стан M2 залишаються незмінними.



3. Якщо потрібно очистити результат порівняння, використовуйте інструкцію RST або ZRST.



Додаткові зауваження

1. Якщо ви оголошуєте операнд **D** в ISPSOft, типом даних буде ARRAY [3] BOOL.
2. Якщо **D +2** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API		Код інструкції			Операнд						функція							
0056			FCMP		P	S ₁ , S ₂ , D						Порівняння чисел з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S ₁	●	●			●	●	●	●	●		O					O		
S ₂	●	●			●	●	●	●	●		O					O		
D		●	●	●				●		O								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁									●				
S ₂									●				
D	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

FCMP	FCMPP
En	En
S1	S1
S2	S2
D	D

S₁ :Значення порівняння з плаваючою комою1

S₂ :Значення порівняння з плаваючою комою2

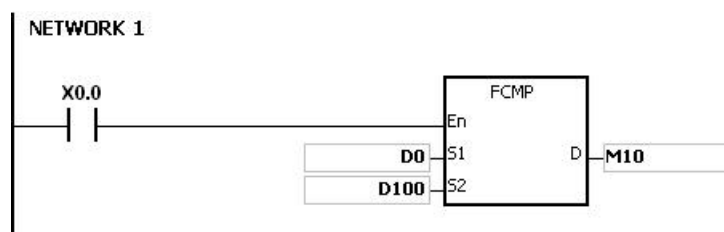
D :Результат порівняння

Пояснення

- Ця інструкція порівнює числа з плаваючою комою в S_1 і S_2 і зберігає результати порівняння (> · = · <) у D .
- Операнд D займає три послідовні пристрої. Результати порівняння зберігаються в D , D +1 і D +2. Якщо значення в S_1 більше, ніж значення в S_2 , D увімкнено. Якщо значення в S_1 дорівнює значенню в S_2 , D +1 увімкнено. Якщо значення в S_1 менше значення в S_2 , D +2 увімкнено.

приклад

- Якщо операндом D є M10, результати порівняння зберігаються в M10, M11 і M12, як показано нижче.
- Коли X0.0 увімкнено, виконується інструкція FCMP. M10, M11 або M12 увімкнено. Коли X0.0 вимкнено, інструкція FCMP не виконується, а стан M10, стан M11 і стан M12 залишаються незмінними.
- Якщо ви хочете отримати результат порівняння $\geq$, $\leq$ або $\neq$, з'єднайте M10–M12 послідовно або паралельно.
- Якщо ви хочете очистити результат порівняння, використовуйте інструкцію RST або ZRST.



Додаткові зауваження

1. Якщо значення в S_1 або S_2 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, контакт вимкнено, SM увімкнено, а код помилки в SR0 — 16#2013.
2. Якщо ви оголошуєте операнд **D** в ISPSoft, типом даних буде ARRAY [3] BOOL.
3. Якщо $D + 2$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд						функція							
0057		FZCP	P	S_1, S_2, S, D						Порівняння зон із плаваючою комою							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		○					○
S_2	●	●			●	●	●	●	●		○					○
S	●	●			●	●	●	●	●		○					○
D		●	●	●				●		○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
S									●				
D	●												

СИМВОЛ

FZCP		FZCPP	
En		En	
S1	D	S1	D
S2		S2	
S		S	

- S_1 : Мінімальне значення зони порівняння
- S_2 : Максимальне значення зони порівняння
- S : Значення порівняння
- D : Результат порівняння

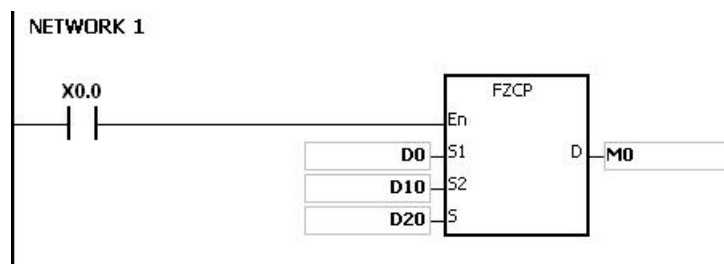
Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

Пояснення

- Ця інструкція порівнює числа з плаваючою комою в S і S_1 , а також порівнює числа з плаваючою комою в S з числами в S_2 , а потім зберігає результати в D .
- Значення в S_1 має бути менше, ніж у S_2 . Якщо значення в S_1 більше, ніж значення в S_2 , S_1 є максимальним/мінімальним значенням під час виконання інструкції FZCP.
- Операнд D займає три послідовних пристрої. Результати порівняння зберігаються в D , $D+1$ і $D+2$. Якщо значення в S_1 більше, ніж значення в S , D увімкнено. Якщо значення в S знаходиться між значенням в S_1 і значенням в S_2 , $D+1$ увімкнено. Якщо порівнюване значення в S_2 менше, ніж значення в S , $D+2$ увімкнено.

приклад

- Якщо операндом D є M0, результати порівняння зберігаються в M0, M1 і M2.
- Коли X0.0 увімкнено, виконується інструкція FZCP. M0, M1 або M2 увімкнено. Коли X0.0 вимкнено, інструкція FZCP не виконується, а стан M0, стан M1 і стан M2 залишаються незмінними.
- Якщо ви хочете очистити результат порівняння, використовуйте інструкцію RST або ZRST.



Додаткові зауваження

1. Якщо значення в S_1 , S_2 або S перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, контакт вимкнено, SM увімкнено, а код помилки в SR0 – 16#2013.
2. Якщо ви оголошуєте операнд D в ISPSOft, типом даних буде ARRAY [3] BOOL.
3. Якщо $D + 2$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API		Код інструкції			Операнд						функція						
0058			MCMP		P	S ₁ , S ₂ , n, D						Матричне порівняння					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●		●	●		○						
S ₂	●	●			●	●		●	●		○						
n	●	●			●	●		●	●		○		○	○			
D		●			●	●		●			○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
n		●				●							
D		●				●							

СИМВОЛ

MCMP		MCMPD	
En		En	
S1	D	S1	D
S2		S2	
n		n	

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-
S_1	: Матричний вихідний пристрій 1	
S_2	: Матричний вихідний пристрій 2	
n	: Довжина масиву	
D	: покажчик	

Пояснення

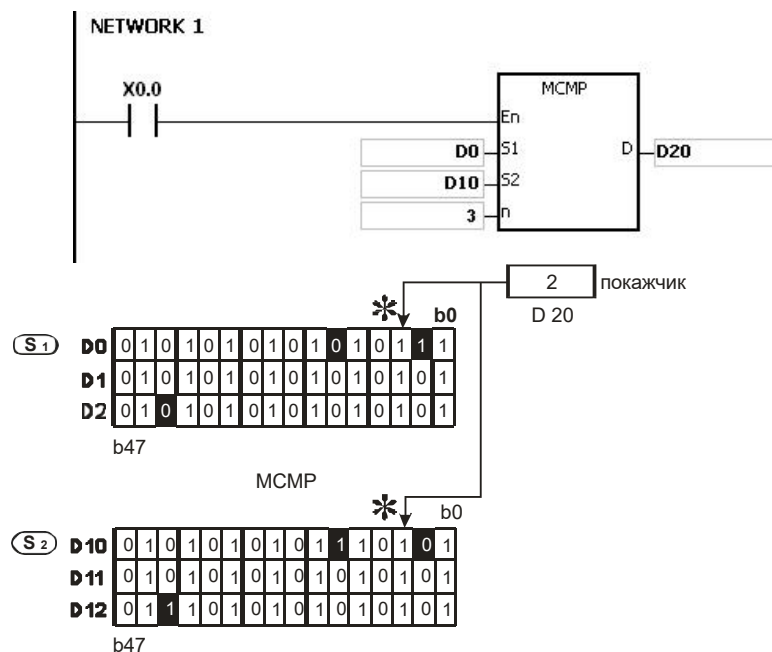
- Ця інструкція шукає біти з різними станами, починаючи з бітів, указаних шляхом додавання одиниці до поточного значення в D . Після знаходження бітів з різними станами інструкція зберігає номер біта в D , і порівняння завершується.
- Операнд n має бути між 1–256.
- Коли SM607 увімкнено, еквівалентні значення порівнюються. Коли SM607 вимкнено, різні значення порівнюються. Коли знайдено відповідні біти, порівняння негайно припиняється, і SM610 увімкнено. Коли порівнюються останні біти, SM608 увімкнено, а номер біта зберігається в D . Порівняння починається з 0-го біта в наступному циклі сканування, і SM609 увімкнено. Коли значення в D перевищує діапазон, SM611 увімкнено.
- Коли виконується інструкція MCMP, вам потрібен 16-бітний регістр, щоб вказати певний біт серед 16 n бітів у матриці для операції. Регістр називається покажчиком і вказується вами. Значення в регістрі знаходиться між 0–16 n -1 і відповідає біту від b0 до b16 n -1. Під час операції вам заборонено змінювати значення вказівника, якщо це впливає на пошук відповідних бітів. Якщо значення вказівника перевищує діапазон, SM611 увімкнено, і інструкція MCMP не виконується.
- Якщо SM608 і SM610 виникають одночасно, вони ввімкнені одночасно.

приклад

1. Коли X0.0 перемикається з OFF. на ON., SM609 вимкнено. Пошук бітів з різними станами (SM607 вимкнено) починається з бітів, визначених додаванням одиниці до поточного значення вказівника.

2. Припустимо, поточне значення в D20 дорівнює 2. Коли X0.0 чотири рази перемикається з OFF. на ON., ви отримуєте наступні результати виконання.

- Значення в D20 дорівнює 5, SM610 увімкнено, а SM608 вимкнено.
- Значення в D20 дорівнює 45, SM610 увімкнено, а SM608 вимкнено.
- Значення в D20 дорівнює 47, SM610 вимкнено, а SM608 увімкнено.
- Значення в D20 дорівнює 1, SM610 увімкнено, а SM608 вимкнено.



Додаткові зауваження

1. Код помилки роботи: якщо пристрої $S_1 + n - 1$ і $S_2 + n - 1$ перевищують діапазон, інструкція MCMP не виконується, SM увімкнено, а код помилки в SR0 – 16#2003. Якщо значення в операнді n не між 1 і 256, інструкція MCMP не виконується, SM увімкнено, а код помилки в SR0 – 16#200B.

2. Прапори операцій:

- | | |
|--------|---|
| | Прапор порівняння матриць. |
| SM607: | ON: порівняння еквівалентних значень OFF: порівняння різних значень |
| SM608: | Порівняння матриць завершується. Коли порівнюються останні біти, SM608 увімкнено. |
| SM609: | ON: порівняння починається з біта 0. |
| SM610: | Прапор пошуку бітів матриці. Коли знайдено відповідні біти, порівняння негайно припиняється, і SM610 увімкнено. |
| SM611: | Прапор помилки покажчика матриці. Коли значення покажчика перевищує діапазон порівняння, SM611 увімкнено. |

API		Код інструкції			Операнд						функція						
0059-0064			CMPT ※		P	S ₁ , S ₂ , n, D						Порівняння таблиць					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●		●	●		○		○	○			
S ₂	●	●			●	●		●	●		○						
n	●	●			●	●		●	●		○	○	○	○			
D		●	●	●				●		●							

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
n		●				●							
D	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

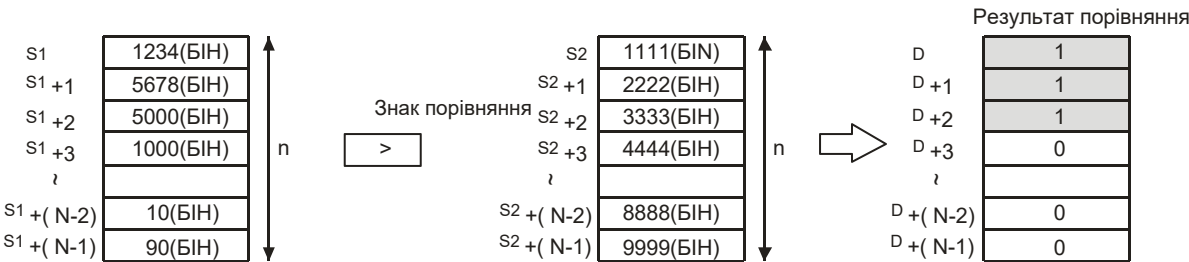
СИМВОЛ

CMPT=	CMPT=P
En	En
S1	S1
S2	S2
n	n

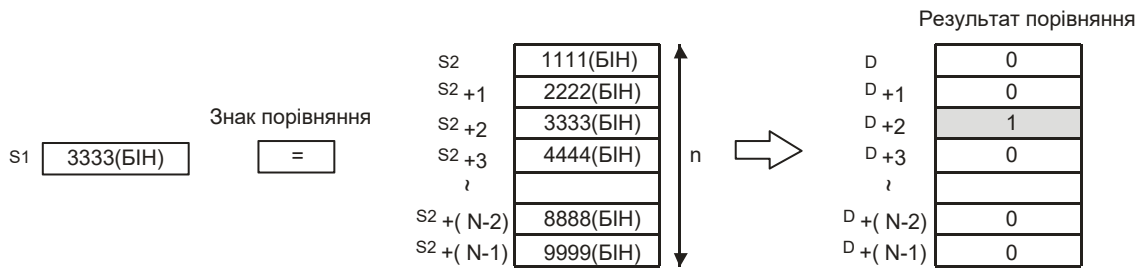
- S_1 : Пристрій-джерело 1
- S_2 : Пристрій-джерело 2
- n : Довжина даних
- D : Результат порівняння

Пояснення

- Ця інструкція порівнює n десяткових чисел зі знаком у пристроях, що починаються з S_1 , з такими в пристроях, що починаються з S_2 , а потім зберігає результати порівняння в D .
- Операнд n має бути між 1–256.
- Значення, яке записується в операнд D , є однобітовим.
- Коли результати порівняння за допомогою інструкції CMPT# показують, що всі пристрої ввімкнено, SM620 увімкнено. В іншому випадку SM620 OFF.
- Якщо операнд S_1 є пристроєм, порівняння відбувається, як показано нижче.



6. Якщо операнд S_1 є константою від -32768 до 32767, порівняння відбувається, як показано нижче.

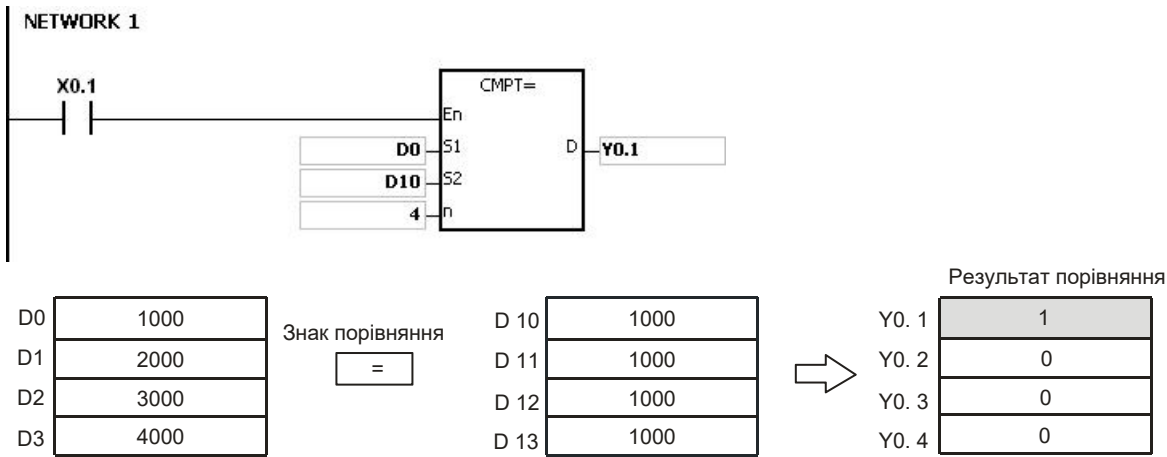


7. Відповідні результати порівняння інструкцій наведено в наступній таблиці.

номер API	16-бітна інструкція	Результат операції порівняння	
		ON	OFF
0059	CMPT =	$S_1 = S_2$	$S_1 \neq S_2$
0060	CMPT < >	$S_1 \neq S_2$	$S_1 = S_2$
0061	CMPT >	$S_1 > S_2$	$S_1 \leq S_2$
0062	CMPT > =	$S_1 \geq S_2$	$S_1 < S_2$
0063	CMPT <	$S_1 < S_2$	$S_1 \geq S_2$
0064	CMPT < =	$S_1 \leq S_2$	$S_1 > S_2$

приклад

Дані в D0–D3 порівнюються з даними в D10–D13. Якщо результатом порівняння є те, що дані в D0–D3 такі ж, як у D10–D13, Y0.1–Y0.4 увімкнено.



Додаткові зауваження

1. Якщо значення в операнді n знаходиться не між 1–256, інструкція не виконується, SM увімкнено, а код помилки в SR0 – 16#200B.
2. Якщо кількість пристроїв, визначена $S_1 - S_1 + n$, $S_2 - S_2 + n$ або D , недостатня, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції				Операнд				функція							
0065		CHKADR			S , n , D				Перевірка адрес у регістрі показчиків							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S																
n	●	●			●	●		●	●		○	○	○	○		
D		●	●	●				●		○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													
n		●				●							
D	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- S** : Регістр показників
- n** : Кількість пристроїв
- D** : Перевірте результат

Пояснення

- Ця інструкція перевіряє, чи значення в **S** і (значення в **S**)+ **n** -1 перевищує діапазон пристрою. Якщо результат полягає в тому, що значення в **S** і (значення в **S**)+ **n** -1 не перевищують діапазон пристрою, пристрій **D** увімкнено. В іншому випадку він OFF.
- S** підтримує регістри показників D, T, C, HC (POINTER/T_POINTER/C_POINTER/HC_POINTER).
- Операнд **n** має бути між 1–1024.
- Інструкцію CHKADR можна використовувати лише у функціональному блоці. Використовуйте CHKADR на початковому етапі розробки програми або коли ви не впевнені, що діапазон пристрою буде перевищено. Після написання та налагодження програми ви можете видалити цю інструкцію.

приклад

1. Створіть програму (Prog0) і функціональний блок (FB0) в ISPSOft.



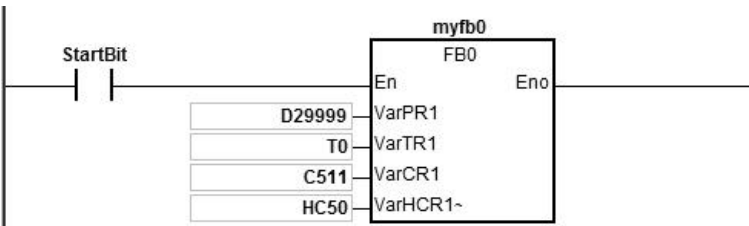
Оголосити дві змінні в програмі.

Local Symbols				
Declaration Type	Identifiers	Address	Type...	Initial Value
VAR	myfb0	N/A [Auto]	FB0	N/A
VAR	StartBit	N/A [Auto]	BOOL	FALSE

2. Оголосить VarPR1, VarTR1, VarCR1 і VarHCR1 у функціональному блоці та призначте їм типи даних POINTER, T_POINTER, C_POINTER і HC_POINTER відповідно.

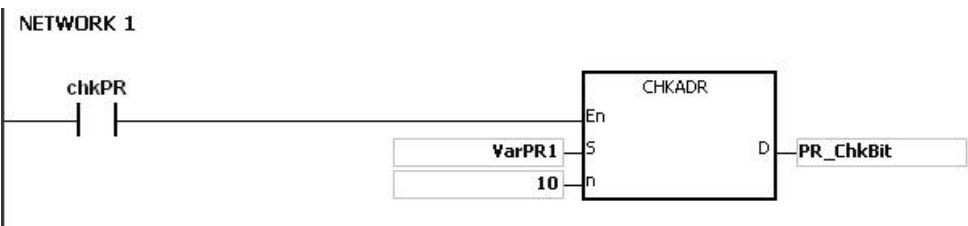
Local Symbols				
Declaration Type	Identifiers	Address	Type...	Initial Value
VAR_IN_OUT	VarPR1	N/A [Auto]	POINTER	N/A
VAR_IN_OUT	VarTR1	N/A [Auto]	T_POINTER	N/A
VAR_IN_OUT	VarCR1	N/A [Auto]	C_POINTER	N/A
VAR_IN_OUT	VarHCR1	N/A [Auto]	HC_POINTER	N/A
VAR	PR_ChkBit	N/A [Auto]	BOOL	FALSE
VAR	TR_ChkBit	N/A [Auto]	BOOL	FALSE
VAR	CR_ChkBit	N/A [Auto]	BOOL	FALSE
VAR	HCR_ChkBit	N/A [Auto]	BOOL	FALSE
VAR	chkPR	N/A [Auto]	BOOL	N/A
VAR	chkTR	N/A [Auto]	BOOL	N/A
VAR	chkCR	N/A [Auto]	BOOL	N/A
VAR	chkHCR	N/A [Auto]	BOOL	N/A

3. Викличте функціональний блок FB0 у програмі та призначте D29999, T0, C511 та HC50 для VarPR1, VarTR1, VarCR1 та VarHCR1 у FB0 відповідно.

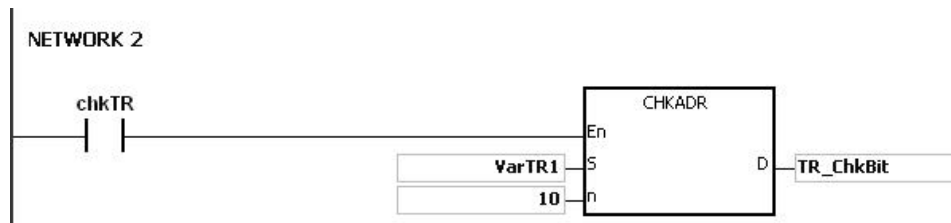


4. Використовуйте інструкцію CHKADR, щоб перевірити, чи VarPR1, VarTR1, VarCR1 і VarHCR1 перевищують діапазон.

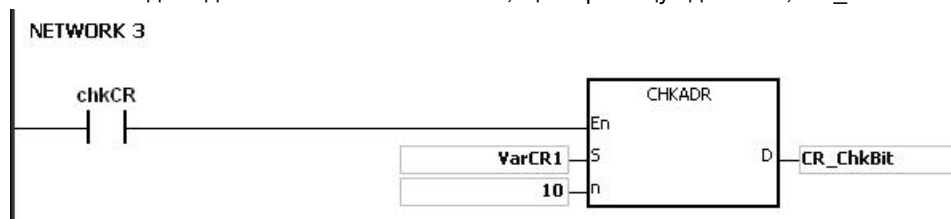
5. Коли chkPR увімкнено, фактичним пристроєм, представленим VarPR1, є D29999. Оскільки дозволений діапазон пристроїв становить від D0 до D29999, а D29999+10-1=D30008 перевищує діапазон, PR_ChkBit вимкнено.



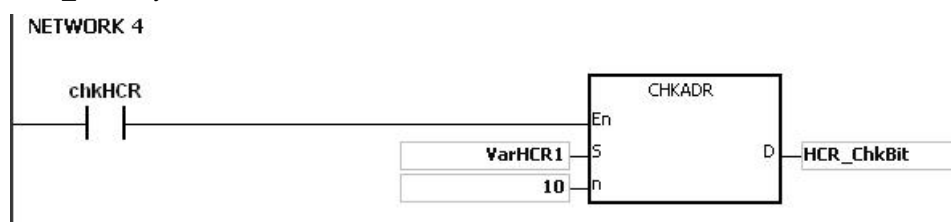
6. Коли chkTR увімкнено, фактичним пристроєм, представленим VarTR1, є T0. Оскільки дозволений діапазон пристроїв становить від T0 до T511 і $T0+10-1=T9$, що не перевищує діапазон, TR_ChkBit увімкнено.



7. Коли chkCR увімкнено, фактичний пристрій представлено C511. Оскільки дозволений діапазон пристроїв становить від C0 до C511 і $C511+10-1=C520$, що перевищує діапазон, CR_ChkBit OFF.



8. Коли chkHCR увімкнено, фактичним пристроєм, представленим HC50, є VarHCR1. Оскільки допустимий діапазон блокувань становить від HC0 до HC255 і $HC50+10-1=HC59$, що не перевищує діапазон, HCR_ChkBit увімкнено.



Додаткові зауваження

1. Якщо значення (фактична адреса пристрою) у **S** перевищує діапазон пристрою, інструкція CHKADR не виконується, SM увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо значення в операнді **n** не знаходиться між 1–1024, інструкція CHKADR не виконується, SM увімкнено, а код помилки в SR0 — 16#200B.

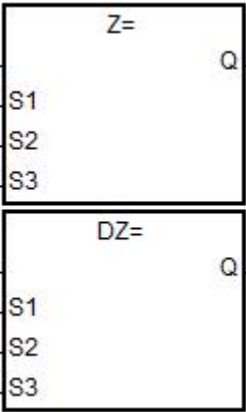
API	Код інструкції			Операнд								функція					
0066-0071	D	LDZ ※		S_1, S_2, S_3								Порівняння абсолютних значень типу контакту LDZ ※					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
S_3	●	●			●	●	●	●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
S_3		●	●			●	●				●	●	

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ



- S_1 : Джерело даних1
- S_2 : Джерело даних2
- S_3 : Результат порівняння

Наприклад, LDZ= і DLDZ=

Пояснення

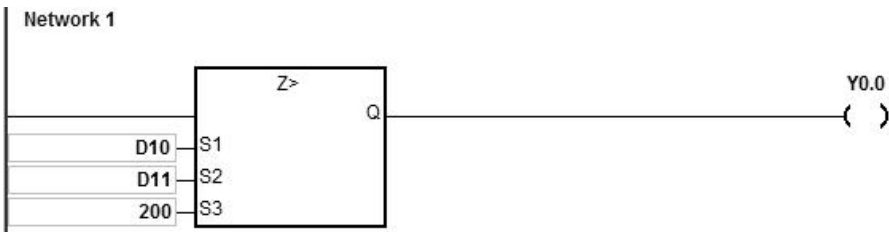
- Ці інструкції порівнюють абсолютне значення різниці між S_1 і S_2 з абсолютним значенням S_3 . Візьмемо, наприклад, інструкцію LDZ=. Якщо результатом порівняння є те, що абсолютне значення різниці між S_1 і S_2 дорівнює абсолютному значенню S_3 , умова безперервності інструкції виконується. Якщо результатом порівняння є те, що абсолютне значення різниці між S_1 і S_2 не дорівнює абсолютному значенню S_3 , умова розриву інструкції виконується.

2. Тільки 32-розрядна інструкція може використовувати 32-розрядний пристрій НС, але не пристрій Е.

номер API	16-бітна інструкція	32-розрядна інструкція	Умова безперервності	Умова розриву
0066	LDZ =	D LDZ =	$ S_1 - S_2 = S_3 $	$ S_1 - S_2 \neq S_3 $
0067	LDZ < >	D LDZ < >	$ S_1 - S_2 \neq S_3 $	$ S_1 - S_2 = S_3 $
0068	LDZ >	D LDZ >	$ S_1 - S_2 > S_3 $	$ S_1 - S_2 \leq S_3 $
0069	LDZ > =	D LDZ > =	$ S_1 - S_2 \geq S_3 $	$ S_1 - S_2 < S_3 $
0070	LDZ <	D LDZ <	$ S_1 - S_2 < S_3 $	$ S_1 - S_2 \geq S_3 $
0071	LDZ < =	D LDZ < =	$ S_1 - S_2 \leq S_3 $	$ S_1 - S_2 > S_3 $

приклад

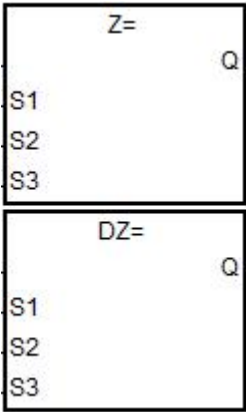
1. Коли абсолютна різниця D10 і D11 перевищує 200, Y0.0 увімкнено. Хоча абсолютна різниця менше 200, Y0.0 вимкнено.



API	Код інструкції				Операнд								функція				
0072-0077	D	ANDZ ※			S_1, S_2, S_3								Порівняння абсолютних значень типу контакту ANDZ ※				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●	●	●	●		○	○	○	○			
S_2	●	●			●	●	●	●	●		○	○	○	○			
S_3	●	●			●	●	●	●	●		○	○	○	○			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1		●	●			●	●				●	●					
S_2		●	●			●	●				●	●					
S_3		●	●			●	●				●	●					

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ



Наприклад, ANDZ= і DANDZ=

- S_1 : Джерело даних1
 S_2 : Джерело даних2
 S_3 : Результат порівняння

Пояснення

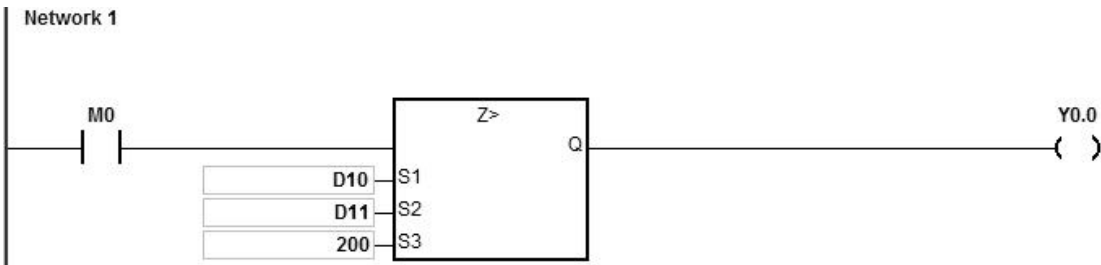
1. Ці інструкції порівнюють абсолютне значення різниці між S_1 і S_2 з абсолютним значенням S_3 . Візьмемо, наприклад, інструкцію ANDZ=. Якщо результатом порівняння є те, що абсолютне значення різниці між S_1 і S_2 дорівнює абсолютному значенню S_3 , умова безперервності інструкції виконується. Якщо результатом порівняння є те, що абсолютне значення різниці між S_1 і S_2 не дорівнює абсолютному значенню S_3 , умова розриву інструкції виконується.

2. Тільки 32-розрядна інструкція може використовувати 32-розрядний пристрій НС, але не пристрій Е.

номер API	16-бітна інструкція	32-розрядна інструкція	Умова безперервності	Умова розриву
0072	ANDZ =	D ANDZ =	$ S_1 - S_2 = S_3 $	$ S_1 - S_2 \neq S_3 $
0073	ANDZ < >	D ANDZ < >	$ S_1 - S_2 \neq S_3 $	$ S_1 - S_2 = S_3 $
0074	ANDZ >	D ANDZ >	$ S_1 - S_2 > S_3 $	$ S_1 - S_2 \leq S_3 $
0075	ANDZ > =	D ANDZ > =	$ S_1 - S_2 \geq S_3 $	$ S_1 - S_2 < S_3 $
0076	ANDZ <	D ANDZ <	$ S_1 - S_2 < S_3 $	$ S_1 - S_2 \geq S_3 $
0077	ANDZ < =	D ANDZ < =	$ S_1 - S_2 \leq S_3 $	$ S_1 - S_2 > S_3 $

приклад

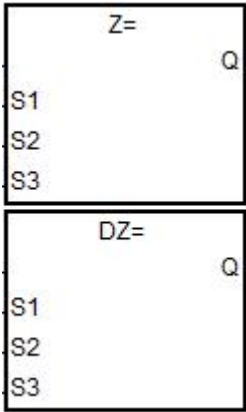
1. Коли M0 увімкнено, а абсолютна різниця D10 і D11 перевищує 200, Y0.0 увімкнено. Хоча абсолютна різниця менше 200, Y0.0 вимкнено.



API	Код інструкції				Операнд							функція					
0078-0083	D	ORZ ※				S_1, S_2, S_3							Порівняння абсолютних значень типу контакту ORZ ※				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●	●	●	●		○	○	○	○			
S_2	●	●			●	●	●	●	●		○	○	○	○			
S_3	●	●			●	●	●	●	●		○	○	○	○			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1		●	●			●	●				●	●					
S_2		●	●			●	●				●	●					
S_3		●	●			●	●				●	●					

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ



- S_1 : Джерело даних1
 S_2 : Джерело даних2
 S_3 : Результат порівняння

Наприклад, ORZ= і DORZ=

Пояснення

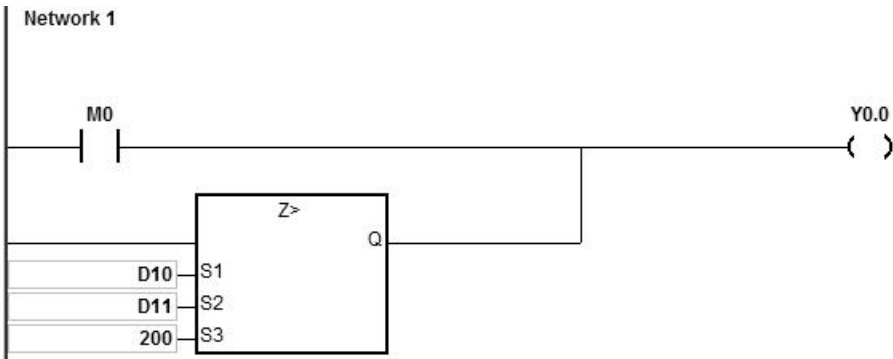
1. Ці інструкції порівнюють абсолютне значення різниці між S_1 і S_2 з абсолютним значенням S_3 . Візьмемо, наприклад, інструкцію ORZ=. Якщо результатом порівняння є те, що абсолютне значення різниці між S_1 і S_2 дорівнює абсолютному значенню S_3 , умова безперервності інструкції виконується. Якщо результатом порівняння є те, що абсолютне значення різниці між S_1 і S_2 не дорівнює абсолютному значенню S_3 , умова розриву інструкції виконується.

2. Тільки 32-розрядна інструкція може використовувати 32-розрядний пристрій HC, але не пристрій E.

номер API	16-бітна інструкція	32-розрядна інструкція	Умова безперервності	Умова розриву
0078	ORZ =	D ORZ =	$ S_1 - S_2 = S_3 $	$ S_1 - S_2 \neq S_3 $
0079	ORZ < >	D ORZ < >	$ S_1 - S_2 \neq S_3 $	$ S_1 - S_2 = S_3 $
0080	ORZ >	D ORZ >	$ S_1 - S_2 > S_3 $	$ S_1 - S_2 \leq S_3 $
0081	ORZ > =	D ORZ > =	$ S_1 - S_2 \geq S_3 $	$ S_1 - S_2 < S_3 $
0082	ORZ <	D ORZ <	$ S_1 - S_2 < S_3 $	$ S_1 - S_2 \geq S_3 $
0083	ORZ < =	D ORZ < =	$ S_1 - S_2 \leq S_3 $	$ S_1 - S_2 > S_3 $

приклад

1. Коли M0 увімкнено, а абсолютна різниця D10 і D11 перевищує 200, Y0.0 увімкнено. Хоча абсолютна різниця менше 200, Y0.0 вимкнено.



6.2 Арифметичні інструкції

6.2.1 Список арифметичних інструкцій

У наведеній нижче таблиці наведено арифметичні інструкції, описані в цьому розділі.

API	Код інструкції		Пульт інструкція	функція
	16-бітний	32-розрядний		
0100	+	D+	✓	Додавання двійкових чисел
0101	-	D-	✓	Віднімання двійкових чисел
0102	*	D*	✓	Множення двійкових чисел
0103	/	D/	✓	Ділення двійкових чисел
0104	—	F+	✓	Додавання чисел з плаваючою комою
0105	—	F-	✓	Віднімання чисел з плаваючою комою
0106	—	F*	✓	Множення чисел з плаваючою комою
0107	—	F/	✓	Ділення чисел з плаваючою комою
0112	BK+	DBK+	✓	Додавання двійкових чисел у блоки
0113	BK-	DBK-	✓	Віднімання двійкових чисел блоками
0114	\$+	—	✓	Зв'язування рядків
0115	INC	DINC	✓	Додавання одиниці до двійкового числа
0116	DEC	DDEC	✓	Віднімання одиниці з двійкового числа
0117	MUL16	MUL32	✓	MUL16: множення двійкових чисел для 16-розрядних інструкцій MUL32: множення двійкових чисел для 32-розрядних інструкцій
0118	DIV16	DIV32	✓	DIV16: Ділення двійкових чисел для 16-розрядних інструкцій DIV32: Ділення двійкових чисел для 32-розрядних інструкцій

6.2.2 Пояснення арифметичних інструкцій

API	Код інструкції			Операнд							функція						
0100	D	+	P	S_1, S_2, D							Додавання двійкових чисел						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		O	O	O	O		
S_2	●	●			●	●	●	●	●		O	O	O	O		
D		●			●	●	●	●			O	O				

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

+	+P
En	En
S1 D	S1 D
S2	S2

S_1 : Augend
 S_2 : Додавання
D : сума

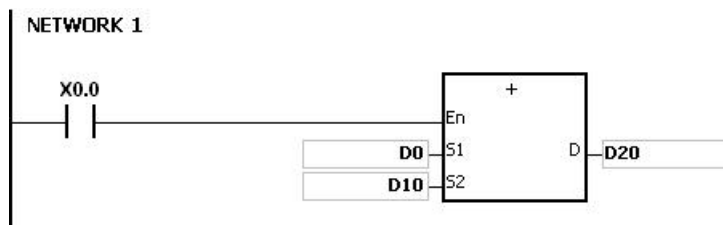
D+	D+P
En	En
S1 D	S1 D
S2	S2

Пояснення

- Ця інструкція додає двійкове значення в S_2 до двійкового значення в S_1 і зберігає суму в D.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.
- Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор) і SM602 (переносний прапор)
- Коли результат операції дорівнює нулю, SM600 увімкнено. В іншому випадку він OFF.
- Для 16-бітних двійкових значень, коли результат операції перевищує діапазон 16-бітних двійкових значень, SM602 увімкнено. В іншому випадку він OFF.
- Для 32-розрядних двійкових значень, коли результат операції перевищує діапазон 32-розрядних двійкових значень, SM602 увімкнено. В іншому випадку він OFF.

Приклад 1

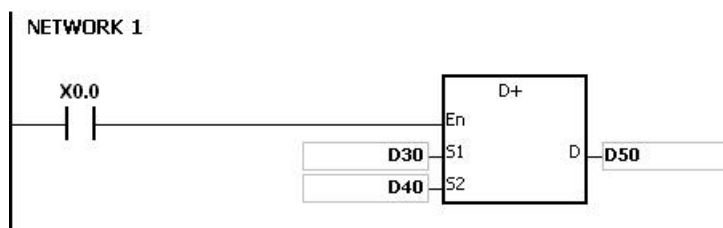
Додавання 16-розрядних двійкових значень: коли X0.0 увімкнено, інструкція додає додаток у D10 до augend у D0 та зберігає суму в D20.



- Коли значення в D0 і D10 дорівнюють 100 і 10 відповідно, D0 плюс D10 дорівнює 110, а 110 зберігається в D20.
- Коли значення в D0 і D10 є 16#7FFF і 16#1 відповідно, D0 плюс D10 дорівнює 16#8000, а 16#8000 зберігається в D20.
- Коли значення в D0 і D10 дорівнюють 16#FFFF і 16#1 відповідно, D0 плюс D10 дорівнює 16#10000. Оскільки результат операції перевищує діапазон 16-бітних двійкових значень, SM602 увімкнено, а значення, що зберігається в D20, дорівнює 16#0. Оскільки результатом операції є 16#0, SM600 увімкнено.

Приклад 2

Додавання 32-розрядних двійкових значень: коли X0.0 увімкнено, інструкція додає додаток у (D41, D40) до аугенда в (D31, D30), і зберігає суму в (D51, D50). Дані в D30, D40 і D50 є нижчими 16-бітними даними, тоді як дані в D31, D41 і D51 є вищими 16-бітними даними.



- Коли значення в (D31, D30) і (D41, D40) становлять 11111111 і 44444444 відповідно, (D31, D30) плюс (D41, D40) дорівнює 55555555, а 55555555 зберігається в (D51, D50).
- Коли значення в (D31, D30) і (D41, D40) дорівнюють 16#80000000 і 16#FFFFFFFF відповідно, (D31, D30) плюс (D41, D40) дорівнює 16#17FFFFFFF. Оскільки результат операції перевищує діапазон 32-розрядних двійкових значень, SM602 увімкнено, а значення, що зберігається в (D51, D50), дорівнює 16#7FFFFFFF.

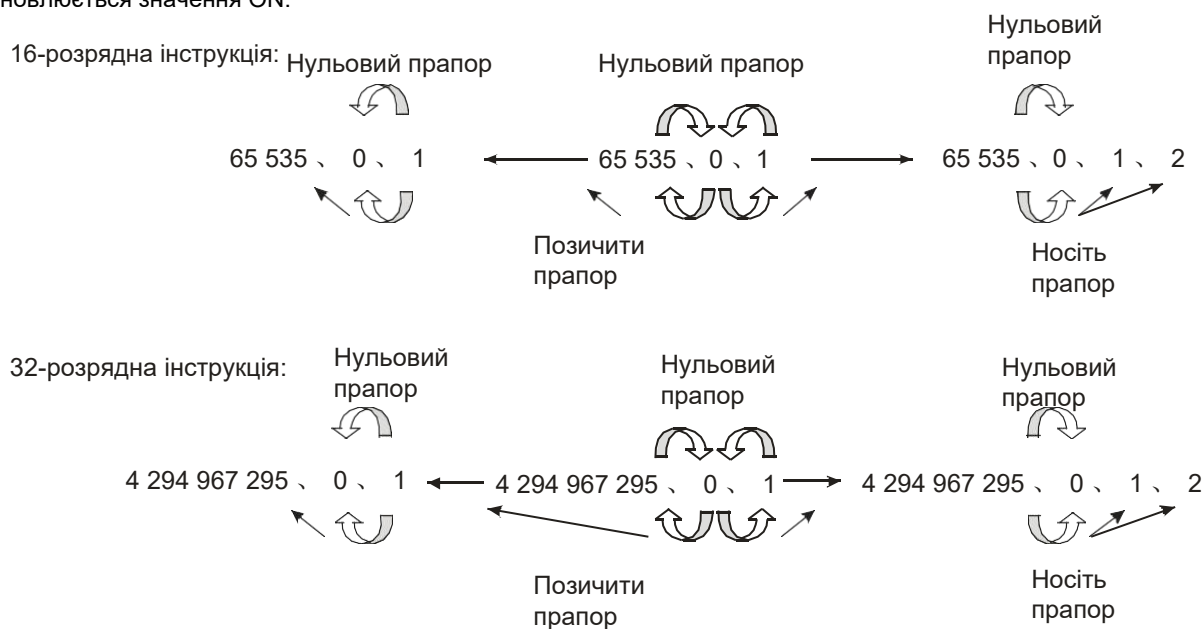
Прапори

Для 16-бітних двійкових значень:

1. Якщо результат операції дорівнює нулю, SM600 увімкнено.
2. Якщо результат операції перевищує 65 535, SM602 увімкнено.

Для 32-розрядних значень:

1. Якщо результат операції дорівнює нулю, SM600 встановлено на ON.
2. Якщо результат операції перевищує 4 294 967 295, для SM602 встановлюється значення ON.



API	Код інструкції			Операнд						функція						
0101	D	-	P	S_1, S_2, D						Віднімання двійкових чисел						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

-	-P
En	En
S1	S1
S2	S2
D-	D-P
En	En
S1	S1
S2	S2

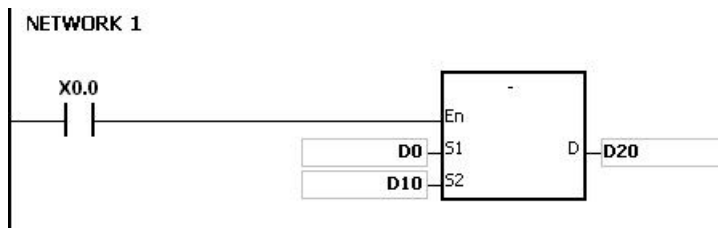
S_1 : Minuend
 S_2 : Від'ємник
D : Різниця

Пояснення

- Ця інструкція віднімає двійкове значення в S_2 із двійкового значення в S_1 і зберігає різницю в D.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.
- Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор) і SM602 (переносний прапор)
- Коли результат операції дорівнює нулю, SM600 увімкнено. В іншому випадку він OFF.
- Коли запозичення відбувається під час арифметики, SM601 увімкнено. В іншому випадку він OFF.

Приклад 1

Віднімання 16-розрядних двійкових значень: коли X0.0 увімкнено, інструкція віднімає від'ємник у D10 від меншого в D0 та зберігає різницю в D20.

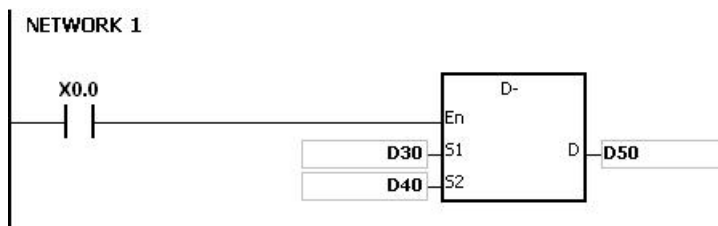


- Коли значення в D0 і D10 дорівнюють 100 і 10 відповідно, D0 мінус D10 залишає 90, а 90 зберігається в D20.
- Коли значення в D0 і D10 становлять 16#8000 і 16#1 відповідно, D0 мінус D10 залишає 16#7FFF, а 16#7FFF зберігається в D20.
- Коли значення в D0 і D10 дорівнюють 16#1 і 16#2 відповідно, D0 мінус D10 залишає 16#FFFF. Оскільки під час операції відбувається запозичення, SM601 увімкнено, а значення, що зберігається в D20, дорівнює 16#FFFF.
- Коли значення в D0 і D10 дорівнюють 16#0 і 16#FFFF відповідно, D0 мінус D10 залишає 16#F0001. Оскільки під час операції відбувається запозичення, SM601 увімкнено, а значення, що зберігається в D20, дорівнює 16#1.

Приклад 2

Додавання 32-розрядних двійкових значень: коли X0.0 увімкнено, інструкція віднімає від'ємник у (D41, D40) від меншого у

(D31, D30) і зберігає суму в (D51, D50). Дані в D30, D40 і D50 є нижчими 16-бітними даними, тоді як дані в D31, D41 і D51 є вищими 16-бітними даними.



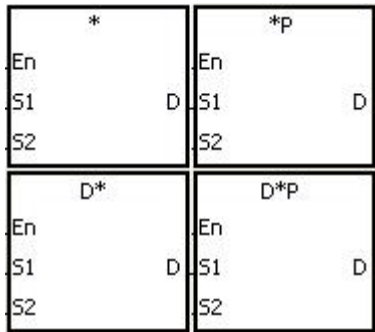
- Коли значення в (D31, D30) і (D41, D40) становлять 55555555 і 11111111 відповідно, (D31, D30) мінус (D41, D40) залишає 44444444, а 44444444 зберігається в (D51, D50).
- Коли значення в (D31, D30) і (D41, D40) дорівнюють 16#80000000 і 16#FFFFFFFF відповідно, (D31, D30) мінус (D41, D40) залишає 16#F80000001. Оскільки під час операції відбувається запозичення, SM601 увімкнено, а значення, що зберігається в (D51, D50), дорівнює 16#80000001.

API	Код інструкції				Операнд						функція						
0102	D	*		P	S ₁ , S ₂ , D						Множення двійкових чисел						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●	●	●	●		○	○	○	○			
S ₂	●	●			●	●	●	●	●		○	○	○	○			
D		●			●	●	●	●			○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●				●	●	
S ₂		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

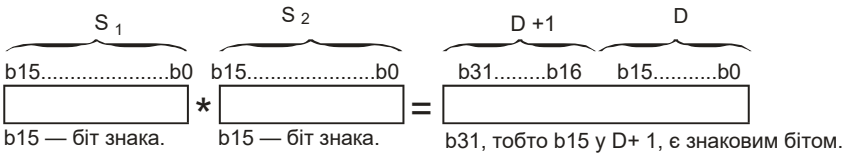
Символ:



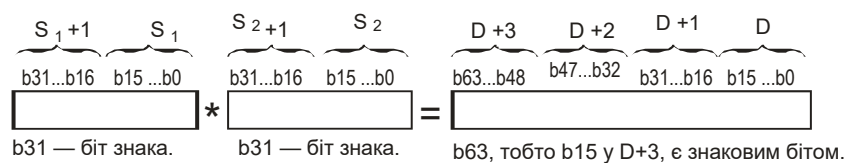
S₁ : Множене
S₂ : Множник
D : Продукт

Пояснення

- 1. Ця інструкція множить двійкове значення зі знаком у S₁ на двійкове значення зі знаком у S₂ і зберігає добуток у D.
- 2. Тільки інструкція D* може використовувати 32-розрядний лічильник.
- 3. Перемноження 16-бітних двійкових значень:



4. Перемноження 32-розрядних двійкових значень:



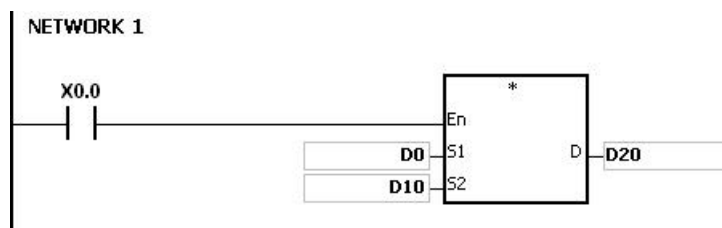
Добуток є 64-бітним значенням і зберігається в регістрі (D+3, D+2, D+1, D0), який складається з 64 бітів.

Коли знаковий біт b63 дорівнює 0, добуток є додатним значенням. Коли знаковий біт b63 дорівнює 1, добуток є від'ємним значенням.

приклад

Інструкція множить 16-бітне значення в D0 на 16-бітне значення в D10 і зберігає 32-бітний добуток у (D21, D20).

Дані в D21 є старшими 16-бітними даними, тоді як дані в D20 є нижчими 16-бітними даними. Чи є результат позитивним чи негативним значенням, залежить від стану старшого біта b31. Коли b31 вимкнено, результат є позитивним значенням. Коли b31 увімкнено, результатом є від'ємне значення.



$$D0 \times D10 = (D21, D20)$$

$$16\text{-бітне значення} \times 16\text{-бітне значення} = 32\text{-бітне значення}$$

API	Код інструкції			Операнд							функція					
0103	D	/	P	S_1, S_2, D							Ділення двійкових чисел					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

$/$	$/P$
En	En
S1 D	S1 D
S2	S2

D/	D/P
En	En
S1 D	S1 D
S2	S2

S_1 : Дивіденд
 S_2 : Дільник
 D : коефіцієнт;
залишок

Пояснення

- Ця інструкція ділить двійкове значення зі знаком у S_1 на двійкове значення зі знаком у S_2 і зберігає приватне та залишок у D .
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник.
- Коли знаковий біт дорівнює 0, значення є додатним. Коли знаковий біт дорівнює 1, значення є від'ємним.
- Ділення 16-бітних значень:

S_1

S_2

b15.....b0

b15.....b0

$/$

$=$

Коефіцієнт

D

Залишок

$D+1$

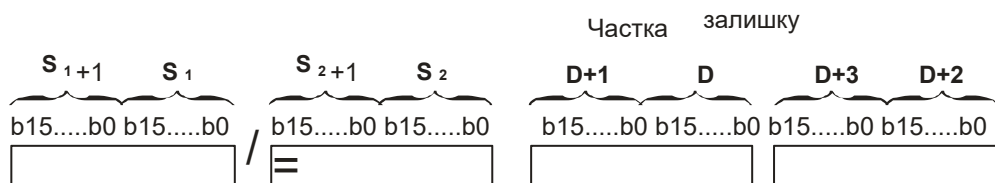
b15.....b0

b15.....b0

Операнд D займає два послідовних пристрої. Частка зберігається в D , а залишок зберігається в $D + 1$.

6 - 54

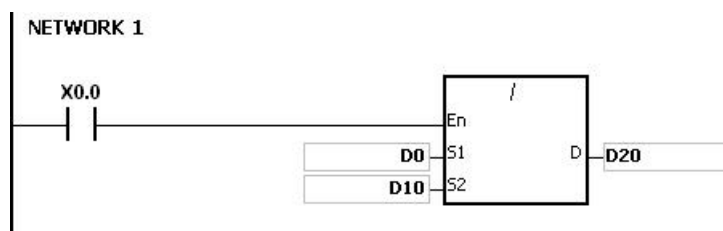
5. Ділення 32-розрядних значень:



Операнд **D** займає два пристрої. Частка зберігається в (**D +1, D**) , а залишок зберігається в (**D +3, D +2**).

приклад

Коли X0.0 увімкнено, інструкція ділить ділене в D0 на дільник у D10 і зберігає частку в D20, а залишок зберігає в D21. Чи буде результат додатним чи від'ємним значенням, залежить від стану старшого біта.



Додаткові зауваження

1. Якщо пристрій виходить за межі діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо дільник дорівнює 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2012.
3. Якщо ви оголосите операнд **D** , який використовується під час виконання 16-бітної інструкції в ISPSOft, тип даних буде ARRAY [2] від WORD/INT.
4. Якщо ви оголосите операнд **D** , який використовується під час виконання 32-розрядної інструкції в ISPSOft, тип даних буде ARRAY [2] DWORD/DINT.

API	Код інструкції			Операнд							функція						
0104		F+	P	S ₁ , S ₂ , D							Додавання чисел з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●	●	●	●							O	
S ₂	●	●			●	●	●	●	●							O	
D		●			●	●	●	●									
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S ₁									●								
S ₂									●								
D									●								
							Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція				
							AS			-			AS				

СИМВОЛ

F+		F+P	
En		En	
S1	D	S1	D
S2		S2	

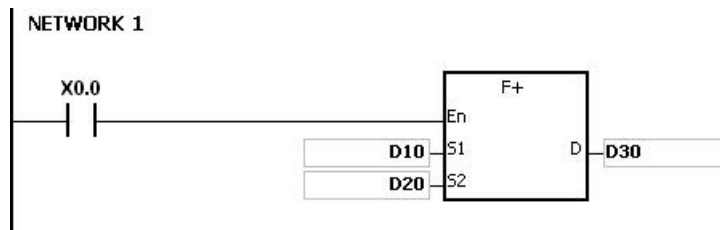
S_1 : Augend
 S_2 : Додавання
D : сума

Пояснення

1. Ця інструкція додає 32-розрядні числа з плаваючою комою одинарної точності в S_2 і S_1 і зберігає суму в D .
2. Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор) і SM602 (переносний прапор)
- Коли результат операції дорівнює нулю, SM600 увімкнено. В іншому випадку він OFF.
 - Коли абсолютне значення результату операції менше значення, яке може бути представлене мінімальним числом з плаваючою комою, значення в D дорівнює 16#FF800000, а SM601 увімкнено.
 - Коли абсолютне значення результату операції перевищує значення, яке може бути представлене максимальним числом з плаваючою комою, значення в D дорівнює 16#7F800000, а SM602 увімкнено.

приклад

Додавання чисел з плаваючою комою одинарної точності: коли X0.0 увімкнено, інструкція додає додаток 16#4046B852 у (D21, D20) до авгенда 16#3FB9999A у (D11, D10) і зберігає суму 16#4091C28F в (D31, D30). 16#4046B852, 16#3FB9999A і 16#4091C28F представляють числа з плаваючою комою 3,105, 1,450 і 4,555 відповідно.

**Додаткове зауваження**

Якщо значення в S_1 або значення в S_2 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції			Операнд						функція							
0105		F-	P	S_1, S_2, D						Віднімання чисел з плаваючою комою							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●	●	●	●								O
S_2	●	●			●	●	●	●	●								O
D		●			●	●	●	●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

F-	F-P
En	En
S1	S1
S2	S2

- S_1 : Мінусенд
- S_2 : Від'ємник
- D : Різниця

6

Пояснення

1. Ця інструкція віднімає 32-розрядне число з плаваючою комою одинарної точності в S_2 із 32-розрядного числа одинарної точності
- Числа з плаваючою комою нумерують у S_1 , а різницю зберігають у D.
2. Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор) і SM602 (переносний прапор)
- Коли результат операції дорівнює нулю, SM600 увімкнено.
 - Коли абсолютне значення результату операції менше значення, яке може бути представлено мінімальним числом з плаваючою комою, значення в D дорівнює 16#FF800000, а SM601 увімкнено.
 - Коли абсолютне значення результату операції перевищує значення, яке може бути представлено максимальним числом з плаваючою комою, значення в D дорівнює 16#7F800000, а SM602 увімкнено.

S₁+1

S₁

b31.....b16 b15.....b0

-

S₂+1

S₂

b31.....b16 b15.....b0

=

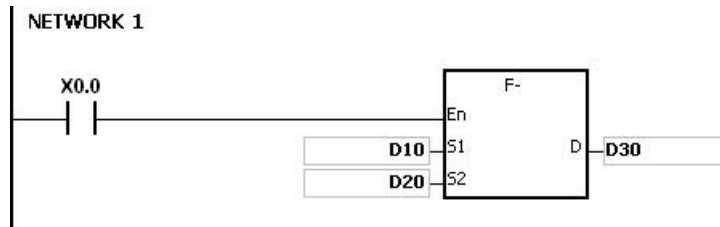
D+1

D

b31.....b16 b15.....b0

приклад

Віднімання 32-розрядних чисел з плаваючою комою одинарної точності: коли X0.0 увімкнено, інструкція віднімає від'ємник у (D21, D20) від меншого у (D21, D20) і зберігає різницю в (D31, D30) .

**Додаткові зауваження**

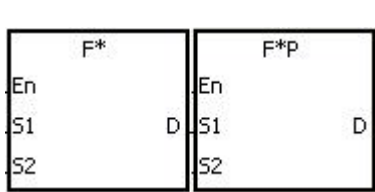
Якщо значення в S_1 або значення в S_2 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API		Код інструкції			Операнд							функція						
0106			F*	P	S ₁ , S ₂ , D							Множення чисел з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S ₁	●	●			●	●	●	●	●							O		
S ₂	●	●			●	●	●	●	●							O		
D		●			●	●	●	●										

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁									●				
S ₂									●				
D									●				

Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
AS		-	AS

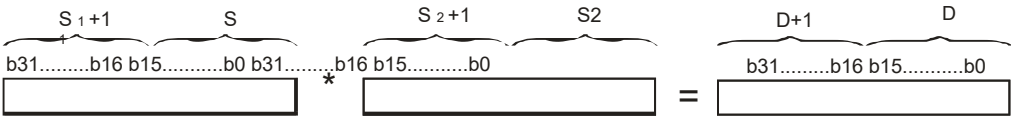
СИМВОЛ



- S₁ : Множене
- S₂ : Множник
- D : Продукт

Пояснення

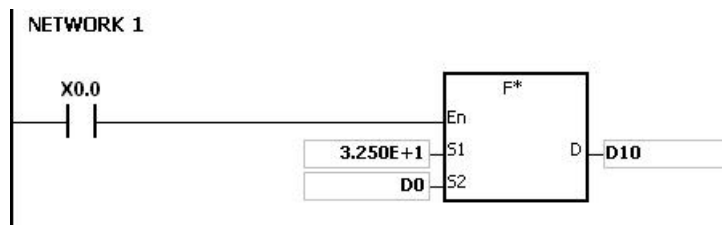
1. Ця інструкція множить 32-розрядне число з плаваючою комою одинарної точності в S₁ на 32-розрядне число з плаваючою комою одинарної точності.
номер точки в S₂ і зберігає добуток в D .
2. Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор) і SM602 (переносний прапор)
 - Коли результат операції дорівнює нулю, SM600 увімкнено.
 - Коли абсолютне значення результату операції менше значення, яке може бути представлено мінімальним числом з плаваючою комою, значення в D дорівнює 16#FF800000, а SM601 увімкнено.
 - Коли абсолютне значення результату операції перевищує значення, яке може бути представлено максимальним числом з плаваючою комою, значення в D дорівнює 16#7F800000, а SM602 увімкнено.



приклад

Множення 32-розрядних чисел з плаваючою комою одинарної точності: коли X0.0 увімкнено, інструкція множить множене

32,5 множитимом у (D1, D0) і зберігає добуток у (D11, D10).

**Додаткові зауваження**

Якщо значення в S_1 або значення в S_2 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції			Операнд	функція
0107		F/	P	S_1, S_2, D	Ділення чисел з плаваючою комою

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●							○
S ₂	●	●			●	●	●	●	●							○
D		●			●	●	●	●								

Тип данных	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
s_1									●				
s_2									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

F/		F/P	
En		En	
S1	D	S1	D
S2		S2	

S₁ : Дивіденд

S₂ : ДІЛЬНИК

D : Коефіцієнт

Пояснения

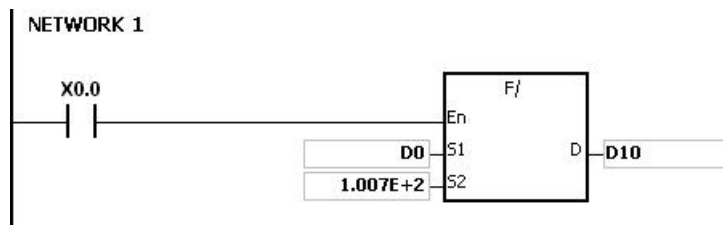
1. Ця інструкція ділить 32-бітове число з плаваючою комою одинарної точності в S_1 на 32-бітове число одинарної точності
- число з плаваючою комою в S_2 і зберігає приватне в D .
2. Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор) і SM602 (переносний прапор)

- Коли результат операції дорівнює нулю, SM600 увімкнено.
- Коли абсолютне значення результату операції менше значення, яке може бути представлено мінімальним числом з плаваючою комою, значення в **D** дорівнює 16#FF800000, а SM601 увімкнено.
- Коли абсолютне значення результату операції перевищує значення, яке може бути представлено максимальним числом з плаваючою комою, значення в **D** дорівнює 16#7F800000, а SM602 увімкнено.

$$\frac{\overbrace{b_{31} \dots b_{16}}^{S_1+1} \overbrace{b_{15} \dots b_0}^{S_1}}{\overbrace{b_{31} \dots b_{16}}^{S_2+1} \overbrace{b_{15} \dots b_0}^{S_2}} = \overbrace{b_{31} \dots b_{16}}^{D+1} \overbrace{b_{15} \dots b_0}^D$$

приклад

Ділення 32-розрядних чисел з плаваючою комою одинарної точності: коли X0.0 увімкнено, інструкція ділить ділене в (D1, D0) на дільник 100,7 і зберігає частку в (D11, D10).

**Додаткові зауваження**

1. Якщо дільник дорівнює 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2012.
2. Якщо значення в S_1 або значення в S_2 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції			Операнд						функція						
0112	D	BK+	P	S_1, S_2, n, D						Додавання двійкових значень у блоках						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●							
S_2	●	●			●	●	●	●	●				○	○		
n	●	●			●	●	●	●	●				○	○		
D		●			●	●	●	●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●					●	●	
S_2		●				●					●	●	
n		●				●					●	●	
D		●				●					●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

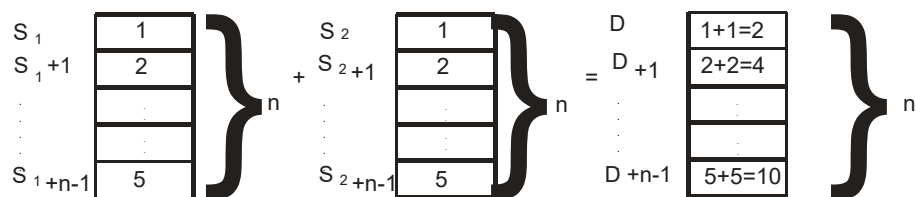
BK+	BK+P
En	En
S1	S1
S2	S2
n	n
DBK+	DBK+P
En	En
S1	S1
S2	S2
n	n

S_1 : Augend
 S_2 : Додавання
 n : Довжина даних
 D : сума

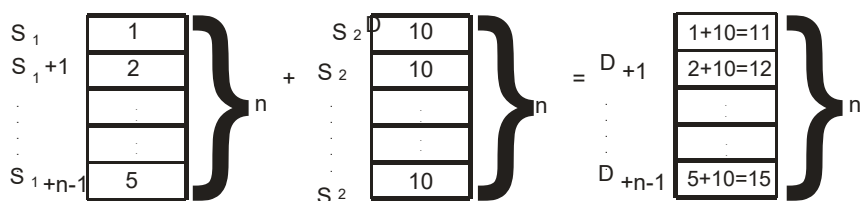
Пояснення

- Ця інструкція додає n частин даних у пристроях, починаючи з S_2 , до тих, що знаходяться в пристроях, починаючи з S_1 .
Аугенди а доданки є двійковими значеннями, а інструкція зберігає суми в D .
- Операнд n має бути між 1–256.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник.
- Коли результат операції дорівнює нулю, SM600 увімкнено.
- Для 16-розрядних інструкцій, коли результат операції менше ніж –32 768, SM601 увімкнено.
- Для 16-розрядних інструкцій, коли результат операції перевищує 32 767, SM602 увімкнено.
- Для 32-розрядних інструкцій, коли результат операції менший за –21 474 836 488, SM601 увімкнено.
- Для 32-розрядних інструкцій, коли результат операції перевищує 2 147 483 647, SM602 увімкнено.

9. Приклад 16-бітної інструкції: коли операнд S_2 є пристроєм (а не константою чи шістнадцятковим значенням)

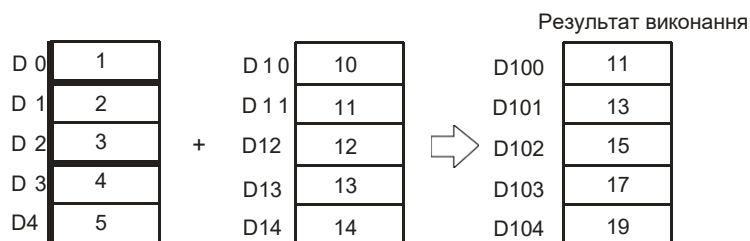
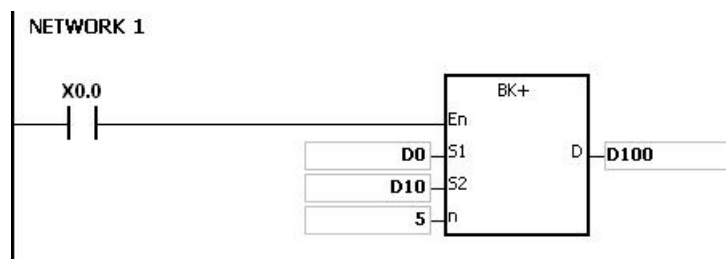


10. Приклад 16-бітної інструкції: коли операнд S_2 є константою або шістнадцятковим значенням



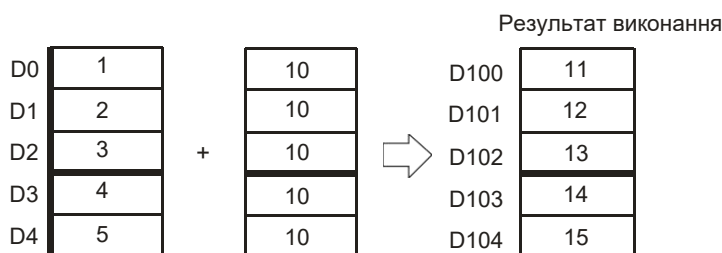
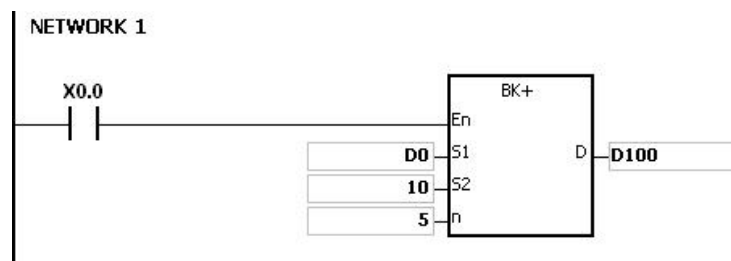
Приклад 1

Коли X0.0 увімкнено, інструкція додає двійкові значення в D10–D14 до двійкових значень у D0–D4 і зберігає суми.



Приклад 2

Коли X0.0 увімкнено, інструкція додає додавач 10 до двійкових значень у D0–D4 і зберігає суми в D100–D104.



Додаткові зауваження

- Для 16-розрядних інструкцій, якщо пристрої $S_1 - S_1 + n - 1$, $S_2 - S_2 + n - 1$ або $D - D + n - 1$ перевищують діапазон пристроїв, інструкція не виконується, SM виконується ON, а код помилки в SR0 – 16#2003.
- Для 32-розрядних інструкцій, якщо пристрої $S_1 - S_1 + 2 * n - 1$, $S_2 - S_2 + 2 * n - 1$ або $D - D + 2 * n - 1$ перевищують діапазон пристрою, інструкція не виконується, SM увімкнено, а код помилки в SR0 — 16#2003.
- Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
- Для 16-розрядних інструкцій, якщо $S_1 - S_1 + n - 1$ перекриваються $D - D + n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.
- Для 32-розрядних інструкцій, якщо $S_1 - S_1 + 2 * n - 1$ перекриваються $D - D + 2 * n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.
- Для 16-розрядних інструкцій, якщо $S_2 - S_2 + n - 1$ перекриваються $D - D + n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.
- Для 32-розрядних інструкцій, якщо $S_2 - S_2 + 2 * n - 1$ перекриваються $D - D + 2 * n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.

API	Код інструкції			Операнд							функція						
0113	D	BK-	P	S_1, S_2, n, D							Віднімання двійкових значень у блоках						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●							
S_2	●	●			●	●	●	●	●				O	O		
n	●	●			●	●	●	●	●				O	O		
D		●			●	●	●	●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●					●	●	
S_2		●				●					●	●	
n		●				●					●	●	
D		●				●					●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

BK-	BK-P
En	En
S1	S1
S2	S2
n	n
DBK-	DBK-P
En	En
S1	S1
S2	S2
n	n

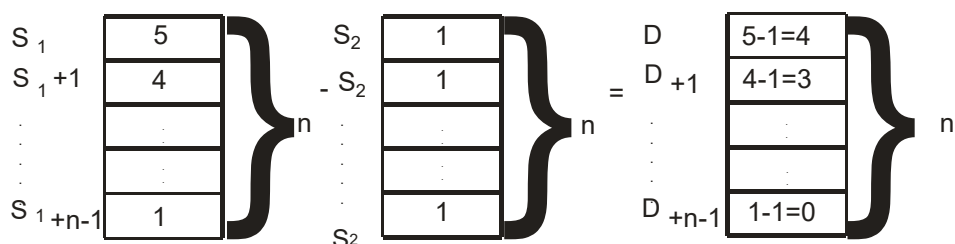
S_1 : Мінуєнд
 S_2 : Від'ємник
 n : Довжина даних
 D : Різниця

Пояснення

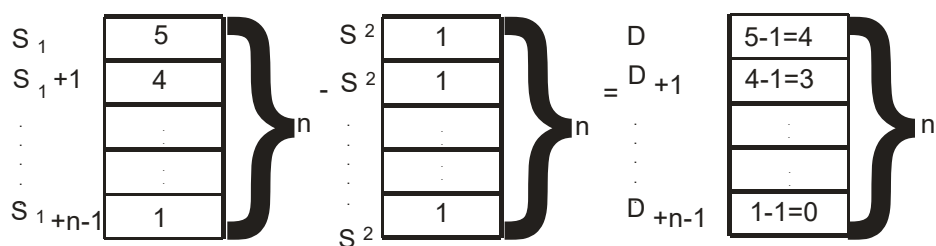
- Ця інструкція віднімає n фрагментів даних у пристроях, починаючи з S_2 , із даних у пристроях, починаючи з S_1 .
від'ємні та від'ємник є двійковими значеннями, а інструкція зберігає різницю в D .
- Операнд n має бути між 1–256.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник.
- Коли результат операції дорівнює нулю, SM600 увімкнено.
- Для 16-розрядних інструкцій, коли результат операції менше ніж –32 768, SM601 увімкнено.
- Для 16-розрядних інструкцій, коли результат операції перевищує 32 767, SM602 увімкнено.
- Для 32-розрядних інструкцій, коли результат операції менший за –2 147 483 648, SM601 увімкнено.

8. Для 32-розрядних інструкцій, коли результат операції перевищує 2 147 483 647, SM602 увімкнено.

9. Приклад 16-бітної інструкції: коли операнд S_2 є пристроєм (а не константою чи шістнадцятковим значенням)

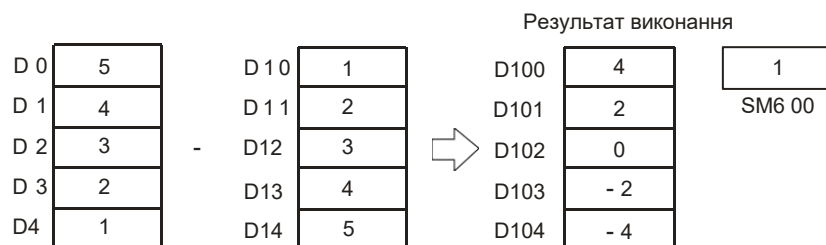
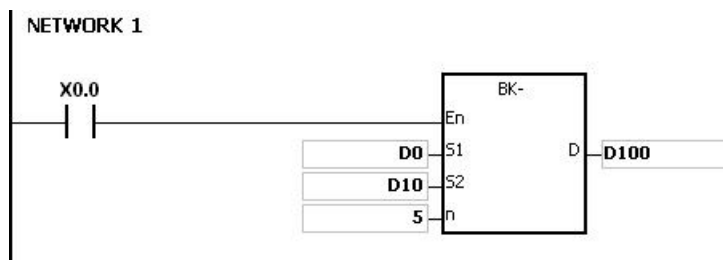


10. Приклад 16-бітної інструкції: коли операнд S_2 є константою або шістнадцятковим значенням



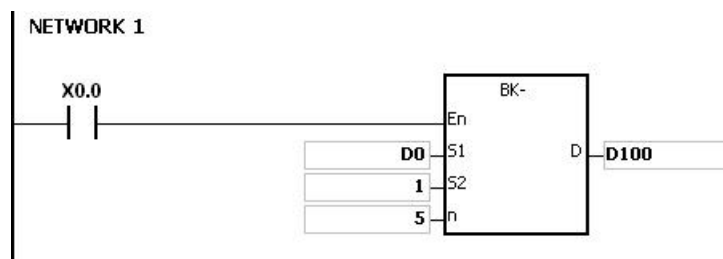
Приклад 1

Коли X0.0 увімкнено, інструкція віднімає двійкові значення в D10–D14 із двійкових значень у D0–D4 та зберігає різниці в D100–D104.



Приклад 2

Коли X0.0 увімкнено, інструкція віднімає від'ємник 1 із двійкових значень у D0–D4 та зберігає різниці в D100–D104.



Результат виконання

D0	10	-	➡	D100	9
D1	9			D101	8
D2	8			D102	7
D3	7			D103	6
D4	6			D104	5

Додаткові зауваження

- Для 16-розрядних інструкцій, якщо пристрої $S_1 - S_1 + n - 1$, $S_2 - S_2 + n - 1$ або $D - D + n - 1$ перевищують діапазон пристроїв, інструкція не виконується, SM виконується ON, а код помилки в SR0 – 16#2003.
- Для 32-розрядних інструкцій, якщо пристрої $S_1 - S_1 + 2 * n - 1$, $S_2 - S_2 + 2 * n - 1$ або $D - D + 2 * n - 1$ перевищують діапазон пристрою, інструкція не виконується, SM увімкнено, а код помилки в SR0 – 16#2003.
- Якщо n не знаходиться в межах 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
- Для 16-розрядних інструкцій, якщо $S_1 - S_1 + n - 1$ перекриваються $D - D + n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.
- Для 32-розрядних інструкцій, якщо $S_1 - S_1 + 2 * n - 1$ перекриваються $D - D + 2 * n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.
- Для 16-розрядних інструкцій, якщо $S_2 - S_2 + n - 1$ перекриваються $D - D + n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.
- Для 32-розрядних інструкцій, якщо $S_2 - S_2 + 2 * n - 1$ перекриваються $D - D + 2 * n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.

API		Код інструкції			Операнд						функція					
0114			\$+	P	S ₁ , S ₂ , D						Зв'язування рядків					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●						O	
S ₂	●	●			●	●		●	●						O	
D		●			●	●		●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁													●
S ₂													●
D													●

Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
AS		AS	-

СИМВОЛ

$\$+$	$\$+P$
En	En
S_1	S_1
S_2	S_2

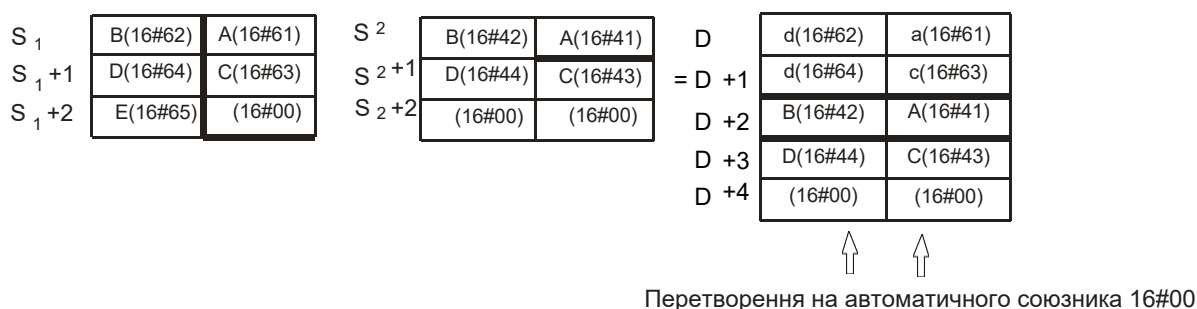
- S_1 : Рядок 1
- S_2 : Рядок 2
- D : Пристрій, в якому зберігається рядок

Пояснення

1. Ця інструкція пов'язує рядок, що починається з даних у пристрої, визначеному S_1 (за винятком 16#00), і рядок починаючи з даних у пристрої, визначеному S_2 (крім 16#00), і зберігає результат у D. Крім того, інструкція додає код 16#00 до кінця пов'язаного рядка в D. Коли інструкція не виконується, дані в D не змінюються.
2. Якщо S_1 , S_2 або D не є рядком (\$), вміст джерела даних може містити до 256 символів (включаючи закінчення код 16#00).
3. У разі успіху рядок у S_1 і рядок у S_2 зв'язуються та зберігаються в D, як показано нижче.

S_1	B(16#62)	A(16#61)	S_2	B(16#42)	A(16#41)	D	B(16#62)	A(16#61)
S_1+1	D(16#64)	C(16#63)	S_2+1	D(16#44)	C(16#43)	$D+1$	D(16#64)	C(16#63)
S_1+2	(16#00)	E(16#65)	S_2+2	(16#00)	(16#00)	$D+2$	A(16#41)	E(16#65)
						$D+3$	C(16#43)	B(16#42)
						$D+4$	(16#00)	D(16#44)

↑ Автоматичний перехід на 16#00

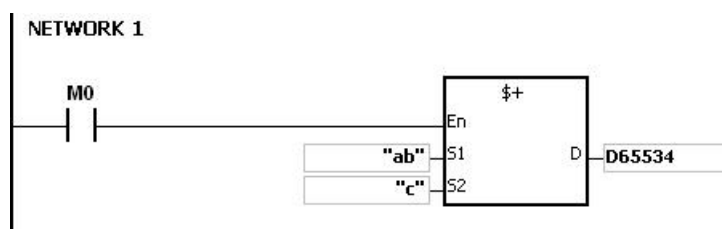


4. Якщо S_1 , S_2 або D не є рядком (\$), кінцевий код 16#00 додається в кінець даних, які переміщуються.
5. Якщо S_1 або S_2 не є рядком, тоді, коли інструкція виконується і перший символ є кодом 16#00, 16#00 все ще пов'язується та переміщується.
6. Рядок «abcde» у S_1 показаний нижче.

S_1	b(16#62)	a(16#61)
$S_1 + 1$	d(16#64)	c(16#63)
$S_1 + 2$	(16# 00)	e(16#65)

приклад

Припустимо, що S_1 — рядок «ab», а S_2 — рядок «с». Після ввімкнення умовного контакту M0 дані в D65534 становлять 16#6261, а дані в D65535 — 16#0063.

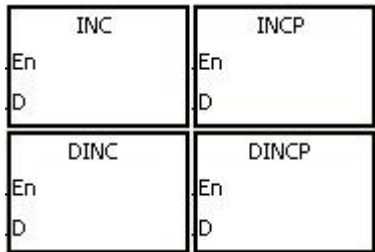


Додаткові зауваження

1. Якщо S_1 або S_2 є рядком, можна перемістити щонайбільше 31 символ.
2. Якщо D недостатньо для вмісту рядка, що складається з рядків у S_1 і S_2 , інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо рядок $S_1 + S_2$ містить більше 256 символів (включно з кінцевим кодом 16#00), інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
4. Якщо S_1 або S_2 перекриває D, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200C.
5. Якщо рядок у S_1 або S_2 не закінчується на 16#00, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200E.

API		Код інструкції			Операнд						функція						
0115		D	INC		P	D						Додавання одиниці до двійкового числа					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			O	O					
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
D		●	●			●	●				●	●					
						Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція							
						AS		AS		AS							

СИМВОЛ



D : Пристрій призначення

Пояснення

- 6
- Ця інструкція додає 1 до значення в D .
 - Тільки інструкція DINC може використовувати 32-розрядний лічильник.
 - Для 16-розрядної операції 32 767 плюс 1 дорівнює -32 768. Для 32-розрядної операції 2 147 483 647 плюс 1 дорівнює 2 147 483 648.

приклад

Коли X0.0 перемикається з OFF. на ON., значення в D0 збільшується на одиницю.

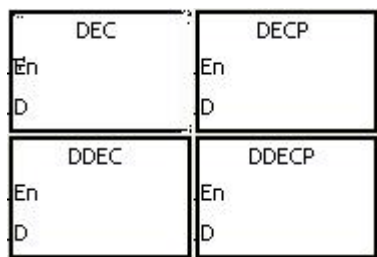


API	Код інструкції			Операнд							функція						
0116	D	DEC	P	D							Віднімання одиниці з двійкового числа						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



D : Пристрій призначення

Пояснення

- 1. Ця інструкція віднімає 1 від значення в D .
- 2. Тільки інструкція DDEC може використовувати 32-розрядний лічильник.
- 3. Для 16-розрядної операції -32 768 мінус 1 залишає 32 767. Для 32-розрядної операції -2 147 483 648 мінус 1 залишає 2 147 483 647.

приклад

Коли X0.0 перемикається з OFF. на ON., значення в D0 зменшується на одиницю.



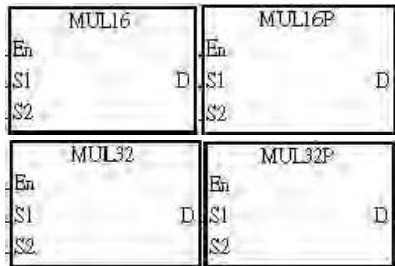
API	Код інструкції			Операнд				функція								
0117		MUL16	P	S_1, S_2, D				Множення 16-розрядних двійкових чисел Множення 32-розрядних двійкових чисел								

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		O	O	O	O		
S_2	●	●			●	●	●	●	●		O	O	O	O		
D		●			●	●	●	●			O	O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

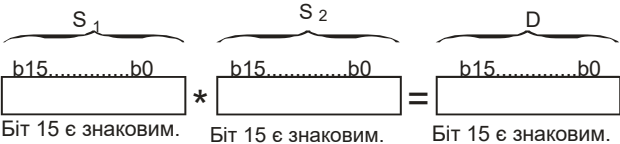
СИМВОЛ



S_1 : Множене
 S_2 : Множник
D : Продукт

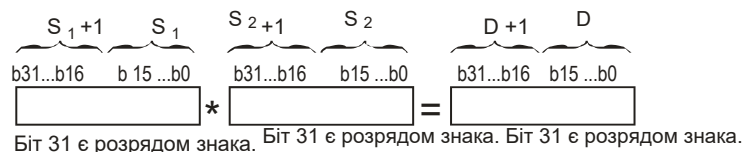
Пояснення

- Ця інструкція множить двійкове значення зі знаком у S_1 на двійкове значення зі знаком у S_2 і зберігає добуток у D .
- Тільки інструкція MUL32 може використовувати пристрій HC.
- 16-розрядне двійкове множення:



Добуток є 16-бітне значення, що зберігається в D , який є 16-бітним регістром. Якщо b15 у D дорівнює 0, добуток, що зберігається в D, є позитивним значенням. Якщо b15 у D дорівнює 1, продукт, що зберігається в D , є від'ємним значенням.

4. 32-розрядне двійкове множення:

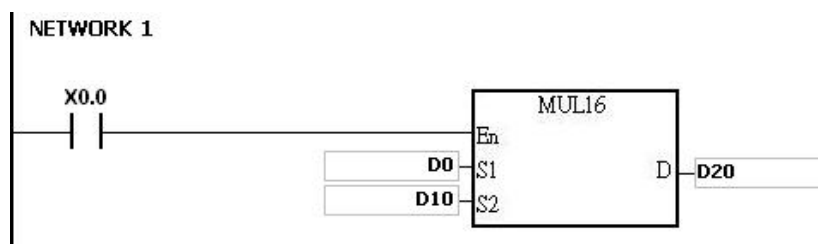


Добуток є 32-бітне значення, яке зберігається в (D , $D + 1$), який є 32-бітним регістром. Якщо b_{31} у D дорівнює 0, добуток, що зберігається в (D , $D + 1$), є таким

позитивне значення. Якщо b_{31} у D дорівнює 1, добуток, що зберігається в (D , $D + 1$), є від'ємним значенням.

приклад

Інструкція множить 16-бітне значення в $D0$ на 16-бітне значення в $D10$ і зберігає добуток у $D20$. Знак добутку (позитивний чи негативний) залежить від крайнього лівого біта (біт 15) у $D20$. Якщо біт 15 у $D20$ дорівнює 0, добуток, що зберігається в $D20$, є додатним значенням. Якщо біт 15 у $D20$ дорівнює 1, добуток, що зберігається в $D20$, є від'ємним значенням.



$$D0 \times D10 = D20$$

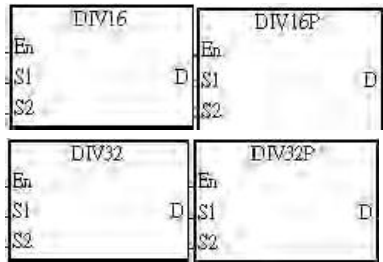
16-бітне значення $\times$ 16-бітне значення = 16-бітне значення

Додаткові зауваження

1. Якщо добуток 16-розрядного множення не є доступним 16-розрядним значенням зі знаком і перевищує максимальне 16-розрядне позитивне число K32767 або менше мінімального від'ємного числа K-32768, прапор переносу SM602 увімкнено, і записуються лише молодші 16 бітів.
2. Якщо вам потрібен повний результат 16-бітного множення (32-бітне значення), використовуйте інструкцію $*/P$ (API 0102). Зверніться до пояснення до $*$ інструкції (API 0102) для отримання додаткової інформації.
3. Якщо добуток 32-розрядного множення не є доступним 32-розрядним значенням зі знаком і є більшим за максимальне 32-розрядне додатне число K2147483647 або менше за мінімальне від'ємне число K-2147483648, прапор переносу SM602 увімкнено, і записуються лише молодші 32 біти.
4. Якщо вам потрібен повний результат 32-розрядного множення (64-розрядне значення), використовуйте API 0102 $D*/D*P$. Зверніться до пояснення до $*$ інструкції (API 0102) для отримання додаткової інформації.

API		Код інструкції			Операнд						функція					
0118			DIV16 DIV32	P	S ₁ , S ₂ , D						Ділення 16-розрядних двійкових чисел Ділення 32- розрядних двійкових чисел					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		O	O	●	●		
S ₂	●	●			●	●	●	●	●		O	O	●	●		
D		●			●	●	●	●			O	O				
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING			
S ₁		●	●			●	●				●	●				
S ₂		●	●			●	●				●	●				
D		●	●			●	●				●	●				
						Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція						
						AS		AS		AS						

СИМВОЛ



S_1 : Дивіденд
 S_2 : Дільник
D : коефіцієнт; залишок

Пояснення

- Ця інструкція ділить двійкове значення зі знаком у S_1 на двійкове значення зі знаком у S_2 і зберігає частку в D.
- Тільки 32-розрядна інструкція може використовувати пристрій HC.
- Знаковий біт=0 (додатне число); знаковий біт =1 (негативне число)
- 16-розрядне двійкове ділення:

S_1

b15.....b0

/

S_2

b15.....b0

=

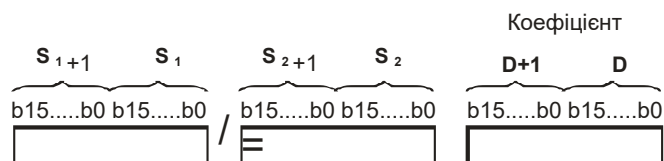
Коефіцієнт

D

b15.....b0

Частка зберігається в D.

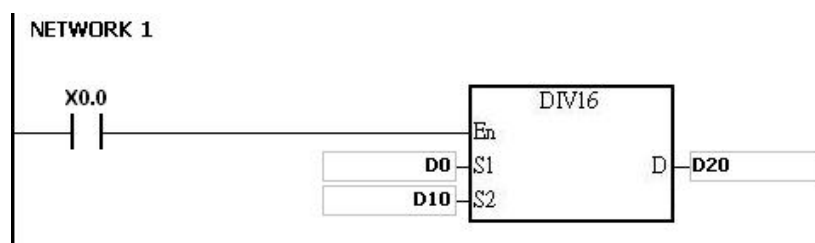
5. 32-розрядне двійкове ділення:



D займає два послідовних пристрої. Частка зберігається в (**D +1**, **D**).

приклад

Коли X0.0 увімкнено, інструкція ділить ділене в D0 на дільник у D10 і зберігає частку D20. Чи є частка додатним чи від'ємним значенням, залежить від крайнього лівого біта в D20.



Додаткові зауваження

1. Якщо пристрій недоступний, інструкція не виконується, SM0 буде увімкнено, а код помилки, збережений у SR0, 16 № 2003.
2. Якщо дільник дорівнює 0, інструкція не виконується, SM0 буде увімкнено, а код помилки, що зберігається в SR0, — 16#2012.
3. Якщо ви хочете зберегти залишок, використовуйте інструкцію «/» (Ділення двійкових значень). Для отримання додаткової інформації зверніться до пояснення до інструкції «/» (API 0103).

6.3 Інструкції з перетворення даних

6.3.1 Список інструкцій з перетворення даних

У наведеній нижче таблиці наведено інструкції з перетворення даних, описані в цьому розділі.

API	Код інструкції		Пульт інструкція	функція
	16-бітний	32-розрядний		
0200	BCD	DBCD	✓	Перетворення двійкового числа в двійкове десятичне число
0201	BIN	DBIN	✓	Перетворення десятичного числа з двійковим кодом у двійкове
0202	FLT	DFLT	✓	Перетворення двійкового цілого числа в двійкове число з плаваючою комою
0204	INT	DINT	✓	Перетворення 32-розрядного числа з плаваючою комою в двійкове ціле число
0206	MMOV	–	✓	Перетворення 16-бітного значення на 32-бітове
0207	RMOV	–	✓	Перетворення 32-бітного значення на 16-бітне
0208	GRY	DGRY	✓	Перетворення двійкового числа в код Грея
0209	GBIN	DGBIN	✓	Перетворення коду Грея в двійкове число
0210	NEG	DNEG	✓	Доповнення двох
0211	–	FNEG	✓	Зміна знака 32-розрядного числа з плаваючою комою
0212	–	FBCD	✓	Перетворення двійкового числа з плаваючою комою в десятичне число з плаваючою комою
0213	–	FBIN	✓	Перетворення десятичного числа з плаваючою комою в двійкове число з плаваючою комою
0214	BKBCD	–	✓	Перетворення двійкових чисел у блоках у двійково-кодовані десятичні числа в блоках
0215	BKBIN	–	✓	Перетворення двійкових чисел у блоках у двійково-кодовані десятичні числа в блоках
0216	SCAL	DSCAL	✓	Пошук масштабованого значення (точка-нахил)
0217	SCLP	DSCLP	✓	Знаходження масштабованого значення (два бали)
0222	SCLM	DSCLM	✓	Багатоточкова операція співвідношення перетинів

6.3.2 Пояснення інструкцій щодо перетворення даних

API	Код інструкції			Операнд								функція					
0200	D	BCD	P	S, D								Перетворення двійкового числа в двійкове десяткове число					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O	O					
D		●			●	●	●	●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

BCD	BCDP
En	En
S	S
D	D

DBCD	DBCDP
En	En
S	S
D	D

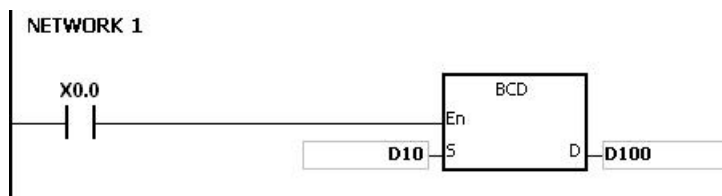
- S** : Вихідний пристрій
- D** : Результат перетворення

Пояснення

- Ця інструкція перетворює двійкове значення в **S** у двійкове десяткове значення та зберігає результат перетворення в **D**.
- Тільки інструкція DBCD може використовувати 32-бітний лічильник, але не пристрій E.
- Чотири основні арифметичні операції в ПЛК, інструкція INC і інструкція DEC працюють з двійковими числами. Щоб відобразити десяткове значення на дисплеї, використовуйте інструкцію BCD для перетворення двійкового значення в двійкове десяткове значення

приклад

1. Коли X0.0 увімкнено, інструкція перетворює двійкове значення в D10 у десяткове значення двійкового коду та зберігає результат перетворення в D100.



2. Якщо D10=16#04D2=1234, результатом перетворення є D100=16#1234.

Додаткові зауваження

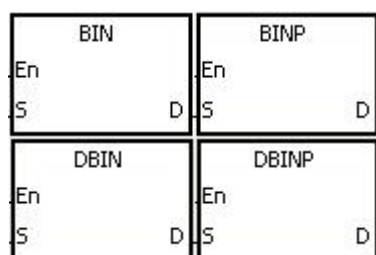
1. Якщо результат перетворення перевищує діапазон 0–9999, інструкція BCD не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200D. Двійкове десяткове значення представлено шістнадцятковим значенням, але одна з цифр не знаходиться між 0–9.
2. Якщо результат перетворення перевищує діапазон 0–99 999 999, інструкція DBCD не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200D. Двійкове десяткове значення представлено шістнадцятковим значенням, але одна з цифр не знаходиться між 0–9.

API	Код інструкції			Операнд						функція						
0201	D	BIN	P	S, D						Перетворення десяткового числа з двійковим кодом у двійкове						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O	O				
D		●			●	●	●	●			O	O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



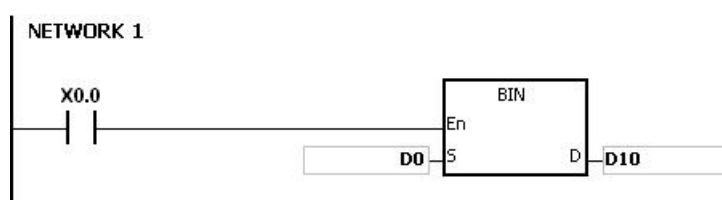
S : Вихідний пристрій
D : Результат перетворення

Пояснення

1. Ця інструкція перетворює двійкове десяткове значення в **S** на двійкове значення та зберігає результат перетворення в **D**.
2. 16-бітне двійкове десяткове значення в **S** має бути між 0–9999, а 32-бітне двійкове десяткове значення у **S** має бути між 0–99 999 999.
3. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.
4. Система автоматично перетворює константи та шістнадцяткові значення у двійкові. Тому вам не потрібно використовувати інструкцію для цього перетворення.

приклад

Коли X0.0 увімкнено, інструкція перетворює двійкове десяткове значення в D0 у двійкове значення та зберігає результат перетворення в D10.

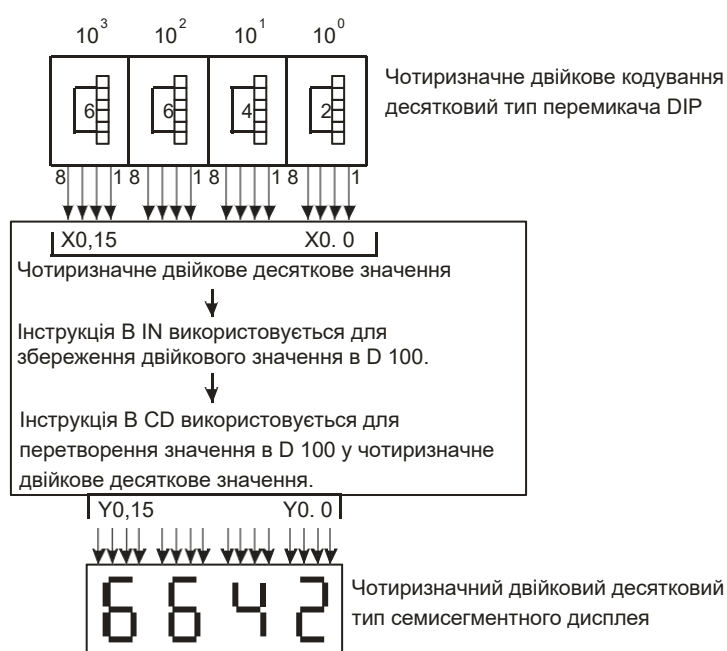
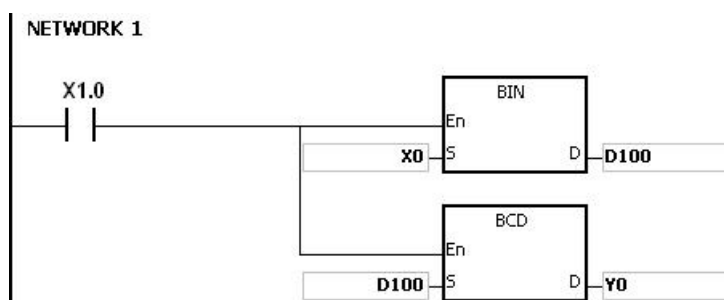


Додаткові зауваження

1. Якщо значення в **S** не є десятковим значенням у двійковому кодуванні, виникає помилка операції, SM0 увімкнено, а код помилки в SR0 – 16#200D. Двійкове десяткове значення представлено шістнадцятковим значенням, але одна з цифр не знаходиться між 0–9.

2. Застосування інструкцій BCD та BIN:

- Перш ніж значення двійкового десяткового типу DIP-перемикача буде зчитано в ПЛК, використовуйте інструкцію BIN для перетворення даних у двійкове значення та збереження результату перетворення в ПЛК.
- Якщо ви хочете відобразити дані, що зберігаються всередині ПЛК, на семисегментному дисплеї десяткового типу з двійковим кодуванням, використовуйте інструкцію BCD для перетворення даних у двійкове десяткове значення перед тим, як дані буде надіслано на семисегментний дисплей. дисплей.
- Коли X1.0 увімкнено, інструкція BIN перетворює двійкове десяткове значення в X0.0–X0.15 у двійкове значення та зберігає результат перетворення в D100. Згодом інструкція BCD перетворює двійкове значення в D100 у двійкове десяткове значення та зберігає результат перетворення в Y0.0–Y0.15.



API	Код інструкції			Операнд					функція							
0202	D	FLT	P	S, D					Перетворення двійкового цілого числа в двійкове число з плаваючою комою							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O	O				
D		●					●	●			O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

FLT	FLTP
En	En
S	S
D	D

DFLT	DFLTP
En	En
S	S
D	D

S : Вихідний пристрій

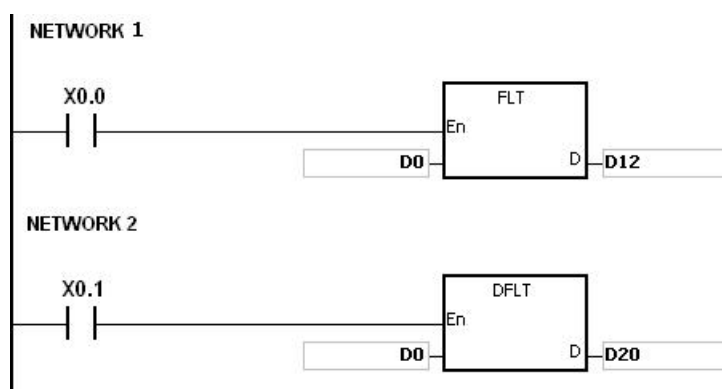
D : Результат перетворення

Пояснення

- Ця інструкція перетворює двійкове ціле число в **S** на число з плаваючою комою одинарної точності та зберігає результат перетворення в **D**.
- Операнд **S**, який використовується в інструкції FLT, не може бути 32-бітним лічильником, а не пристроєм E.
- Пристрій-джерело **S**, що використовується в інструкції FLT, займає один регістр, а **D**, що використовується в FLT, займає два регістри.
- Пристрій-джерело **S**, що використовується в інструкції DFLT, займає два регістри, а **D**, що використовується в DFLT, також займає два регістри.
 - Коли абсолютне значення результату перетворення перевищує значення, яке може бути представлено максимальним числом з плаваючою комою, SM602 увімкнено, а максимальне число з плаваючою комою зберігається в **D**.
 - Коли абсолютне значення результату перетворення менше значення, яке може бути представлено мінімальним числом з плаваючою комою, SM601 увімкнено, а мінімальне число з плаваючою комою зберігається в **D**.
 - Коли результат перетворення дорівнює нулю, SM600 увімкнено.

Приклад 1

1. Коли X0.0 увімкнено, інструкція перетворює двійкове ціле число в D0 на число з плаваючою комою одинарної точності та зберігає результат перетворення в (D13, D12).
2. Коли X0.1 увімкнено, інструкція перетворює двійкове ціле число в (D1, D0) у число з плаваючою комою одинарної точності та зберігає результат перетворення в (D21, D20).
3. Припустимо, значення в D0 дорівнює 10. Коли X0.0 увімкнено, інструкція перетворює 10 у число з плаваючою комою одинарної точності 16#41200000, а потім зберігає 16#41200000 у 32-розрядному регістрі (D13, D12).
4. Припустимо, що значення в 32-розрядному регістрі (D1, D0) дорівнює 100 000. Коли X0.1 увімкнено, інструкція перетворює 100 000 у число з плаваючою комою одинарної точності 16#47C35000 і зберігає 16#47C35000 у 32-розрядному регістрі (D21, D20).

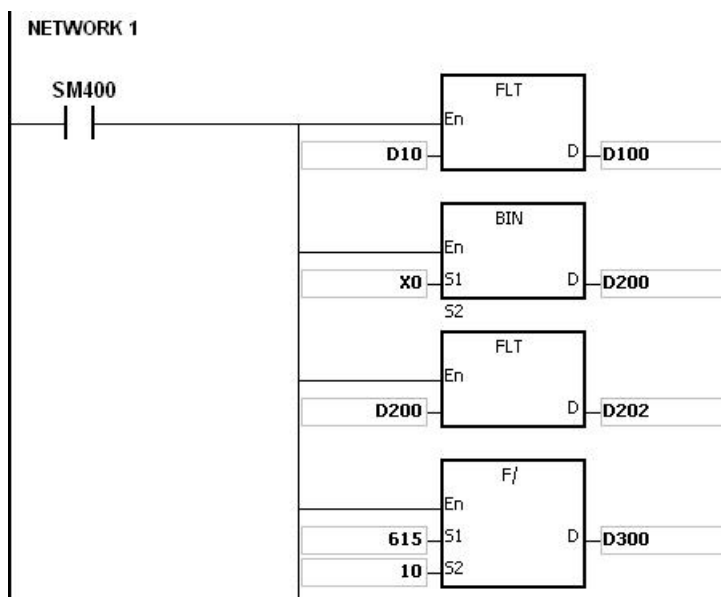
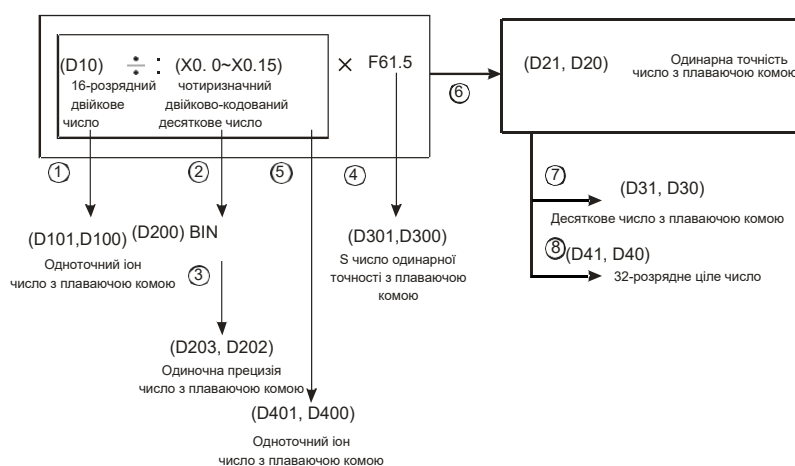


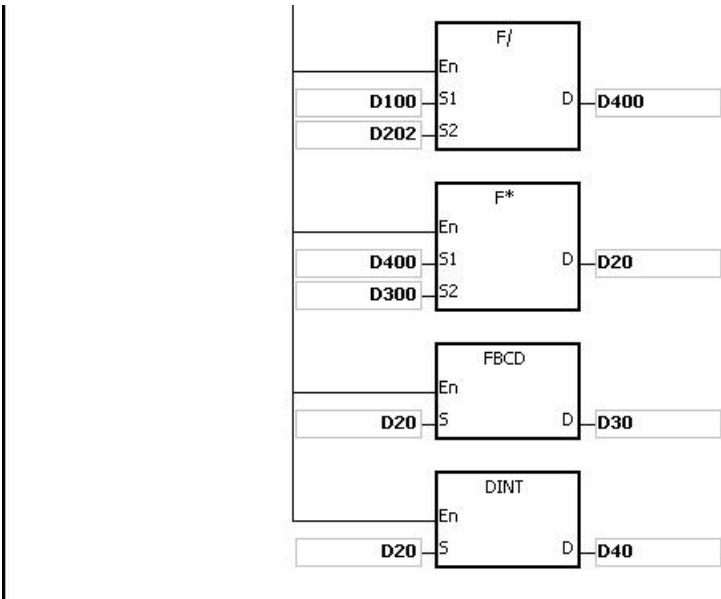
Приклад 2

Ви можете скористатися прикладними інструкціями для виконання наступного розрахунку.

- Перетворіть двійкове ціле число в D10 на число з плаваючою комою одинарної точності та збережіть результат перетворення в (D101, D100).
- Перетворіть двійкове десятичне значення X0.0–X0.15 у двійкове значення та збережіть результат перетворення в D200.
- Перетворіть двійкове ціле число в D200 на число з плаваючою комою одинарної точності та збережіть результат перетворення в (D203, D202).
- Розділіть константу 615 на константу 10 і збережіть приватне, яке є числом з плаваючою комою одинарної точності, у (D301, D300).
- Розділіть число з плаваючою комою одинарної точності в (D101, D100) на число з плаваючою комою одинарної точності в (D203, D202) і збережіть частку, яка є числом з плаваючою комою одинарної точності, у (D401, D400).

- Помножьте число з плаваючою комою одинарної точності в (D401, D400) на число з плаваючою комою одинарної точності
номер у (D301, D300) і збережіть добуток, який є числом з плаваючою комою одинарної точності, у (D21, D20).
- Перетворіть число з плаваючою комою одинарної точності в (D21, D20) на десяткове число з плаваючою комою та збережіть результат перетворення в (D31, D30).
- Перетворіть число з плаваючою комою одинарної точності в (D21, D20) на двійкове ціле число та збережіть результат перетворення в (D41, D40).





API	Код інструкції			Операнд	функція
0204	D	INT	P	S, D	Перетворення 32-розрядного числа з плаваючою комою в двійкове ціле число

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O					
D		●			●	●	●	●			O	O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

INT	INTP
En	En
S	S
D	D
DINT	DINTP
En	En
S	S
D	D

S : Вихідний пристрій

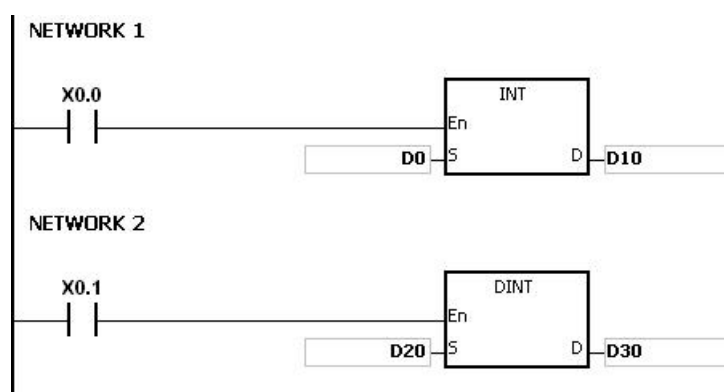
D : Результат перетворення

Пояснення

- Ця інструкція перетворює число з плаваючою комою одинарної точності в **S** у двійкове ціле число, а потім округляє двійкове число з плаваючою комою до найближчої цілої цифри стає двійковим цілим числом, а потім інструкція зберігає двійкове ціле число в **D**.
- Пристрій-джерело **S**, що використовується в інструкції INT, займає два регістри, а **D**, що використовується в INT, займає один регістр.
- Пристрій джерела **S**, який використовується в інструкції DINT, займає два регістри, а **D**, який використовується в DINT, також займає два регістри.
- Операнд **D**, який використовується в інструкції INT, не може бути 32-бітним лічильником, але не пристроєм E.
- Інструкція INT протилежна інструкції FLT.
- Коли результат перетворення дорівнює нулю, SM600 увімкнено.
- Під час перетворення, якщо число з плаваючою комою округляється до найближчої цілої цифри, SM601 увімкнено.
- Коли результат перетворення перевищує діапазон, SM602 увімкнено.
- Для інструкцій INT/IINTP діапазон результату перетворення становить від -32 768 до 32 767.
- Для інструкцій DINT/DINTP діапазон результатів перетворення становить від -2 147 483 648 до 2 147 483 647.

приклад

1. Коли X0.0 увімкнено, інструкція перетворює число з плаваючою комою одинарної точності в (D1, D0) у двійкове ціле число та зберігає результат перетворення в D10. Ця інструкція округлює двійкове число з плаваючою комою до найближчої цілої цифри.
2. Коли X0.1 увімкнено, інструкція перетворює число з плаваючою комою одинарної точності в (D21, D20) у двійкове ціле число та зберігає результат перетворення в (D31, D30). Ця інструкція округлює двійкове число з плаваючою комою до найближчої цілої цифри.



Додаткові зауваження

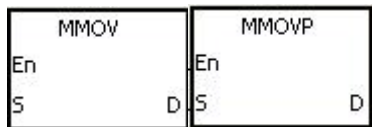
Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконано, SM0 увімкнено, а код помилки в SR0 – 16#2013.

API	Код інструкції				Операнд				функція							
0206		MMOV	P		S, D				Перетворення 16-бітного значення на 32-бітове							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●		○	○	○	○		
D		●			●	●	●	●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●	●				●	●	
D			●				●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



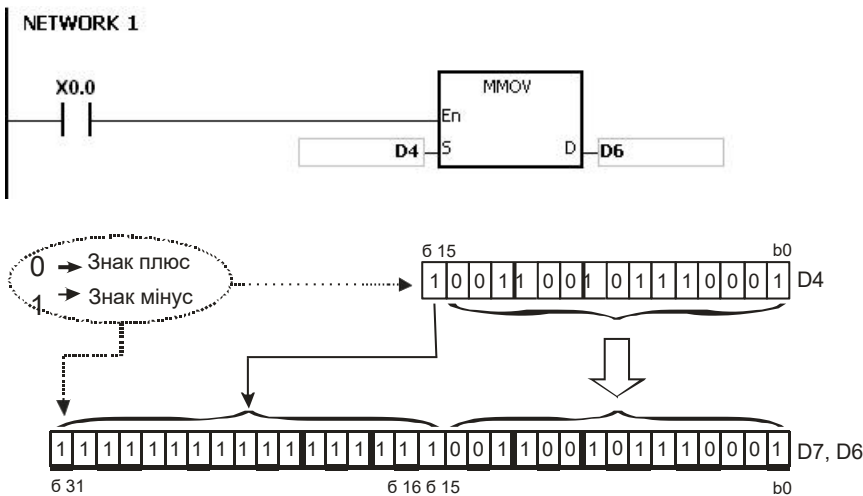
- S : Вихідний пристрій
- D : Результат перетворення

Пояснення

Ця інструкція копіює дані з 16-розрядного пристрою S на 32-розрядний пристрій D і копіює знаковий біт із S у D.

приклад

Коли X0.0 увімкнено, інструкція копіює значення b15 у D4 у b15–b31 у (D7, D6), копіює значення b0–b14 у відповідні біти в (D7, D6) і ігнорує біти b15–b30. Дані в (D7, D6) є від’ємним значенням (те саме, що й джерело).

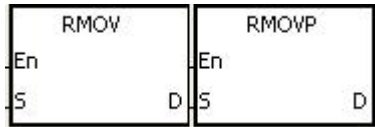


API	Код інструкції				Операнд							функція					
0207		RMOV	P		S, D							Перетворення 32-бітного значення на 16-бітне					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O		O	O			
D		●			●	●		●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S			●				●						
D		●				●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



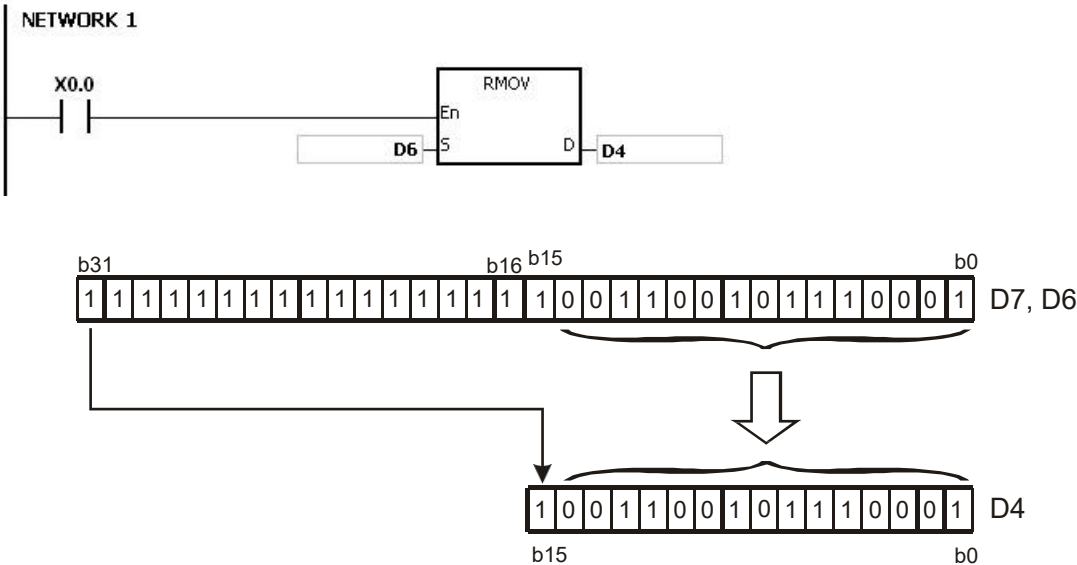
- S : Вихідний пристрій
- D : Результат перетворення

Пояснення

Ця інструкція копіює дані з 32-розрядного пристрою S на 16-розрядний пристрій D і копіює знаковий біт з b31 в b15, тому що значення в D4 від’ємне (те саме, що b31).

приклад

Коли X0.0 увімкнено, інструкція копіює значення b31 у b15 у D4, копіює значення b0–b14 у відповідні біти в D4 та ігнорує біти b15–b30. Дані в (D7, D6) є від’ємним значенням (те саме, що й джерело).



API	Код інструкції			Операнд						функція							
0208	D	GRY	P	S, D						Перетворення двійкового числа в код Грея							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		○	○	○	○			
D		●			●	●	●	●			○	○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



S : Вихідний пристрій

D : Результат перетворення

Пояснення

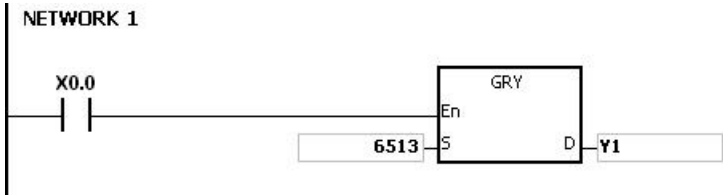
- Ця інструкція перетворює двійкове значення в пристрої, визначеному **S** , у код Грея та зберігає перетворення призначають до пристрою, зазначеного **D** .
- Тільки інструкція DGRY може використовувати 32-лічильник, але не пристрій E.
- Значення в **S** має бути в межах доступного діапазону.

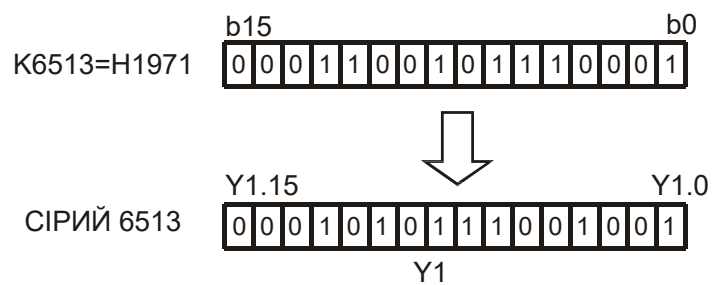
Значення в **S** у 16-бітній інструкції має бути між 0–32 767.

Значення **S** у 32-розрядній інструкції має бути між 0–2 147 483 647.

приклад

Коли X0.0 увімкнено, інструкція перетворює константу 6513 на код Грея та зберігає результат перетворення в Y1.0–Y1.15.





Додаткові зауваження

Якщо значення в **S** менше 0, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд						функція							
0209	D	GBIN	P	S, D						Перетворення коду Грея в двійкове число							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O	O	O	O		
D		●			●	●	●	●			O	O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

GBIN	GBINP
En	En
S	D
DGBIN	DGBINP
En	En
S	D

S : Вихідний пристрій

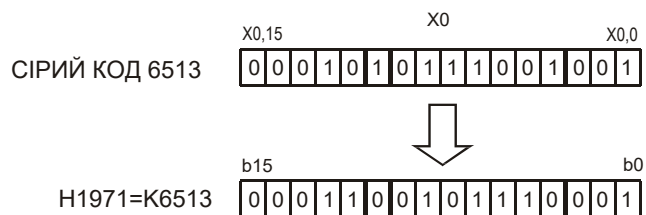
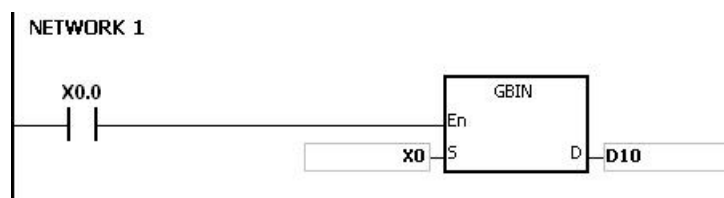
D : Результат перетворення

Пояснення

- Ця інструкція перетворює код Грея в пристрої, визначеному **S**, у двійкове значення та зберігає результат перетворення в пристрої, визначеному **D**.
- Використовуйте цю інструкцію для перетворення коду Грея в датчик абсолютного положення, підключений до входу терміналу ПЛК у двійкове значення. Результат перетворення зберігається у вказаному регістрі.
- Тільки інструкція DGBIN може використовувати 32-лічильник, але не пристрій E.
- Значення в приладі **D** має бути в межах доступного діапазону.
Значення в пристрої **D** у 16-бітній інструкції має бути між 0–32 767.
Значення пристрою **D** у 32-розрядній інструкції має бути між 0–2 147 483 647.

приклад

Коли X0.0 увімкнено, інструкція перетворює код Грея в датчику абсолютного положення, який підключено до входів X0.0–X0.15, у двійкове значення та зберігає результат перетворення в D10.



Додаткові зауваження

Якщо значення в **S** менше 0, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд							функція						
0210	D	NEG	P	D							Знаходження доповнення двох						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



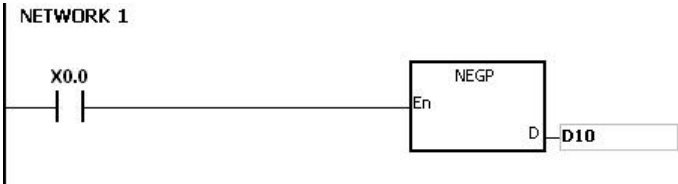
D : Пристрій, в якому два комплемент зберігається

Пояснення

- 1. Ця інструкція перетворює від’ємне двійкове значення на комплімент двох.
- 2. Тільки інструкція DNEG може використовувати 32-розрядний лічильник.
- 3. Як правило, ви використовуєте імпульсні інструкції NEGP та DNEGP.

Приклад 1

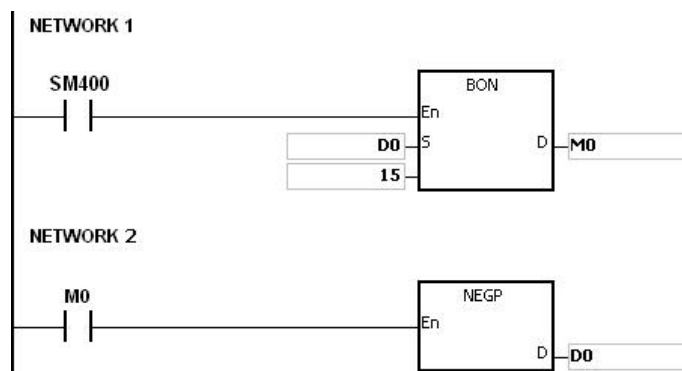
Коли X0.0 перемикається з OFF на ON, ця інструкція інвертує всі біти в D0 (0 стає 1, а 1 стає 0), і 1 додається до результату, а потім зберігає кінцеве значення у вихідному регістрі D10.



Приклад 2

Знаходження двох компліментів від'ємного значення:

1. Коли значення 15-го біта в D0 дорівнює 1, M0 увімкнено, а значення в D0 є від'ємним значенням.
2. Коли M0 увімкнено, інструкція NEG знаходить доповнення до двох від'ємних значень у D0 (відповідне додатне значення).

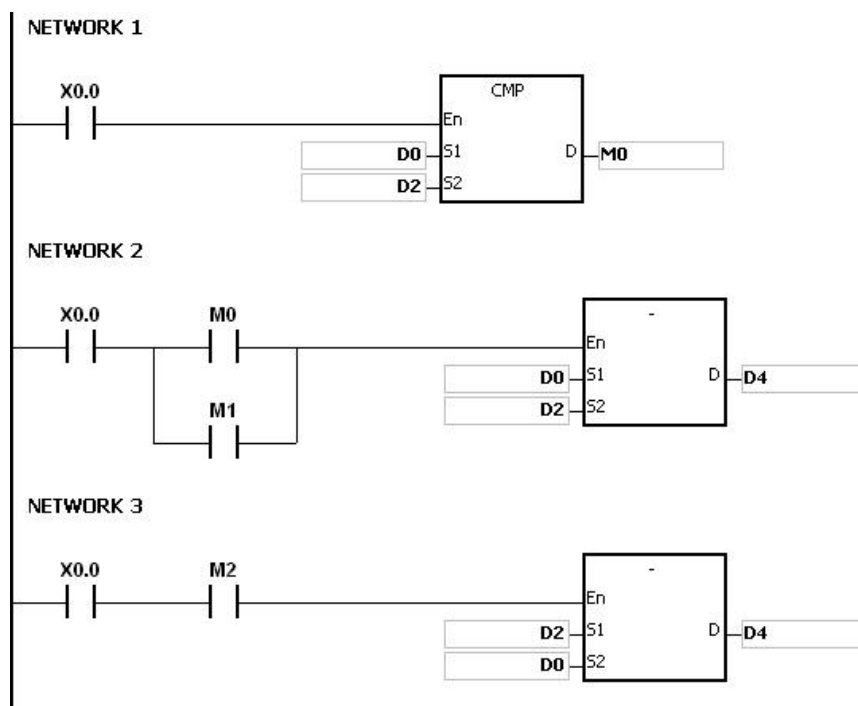


Приклад 3

Знаходження компліменту різниці між двома значеннями:

Коли X0.0 увімкнено,

1. Якщо значення в D0 більше, ніж у D2, M0 увімкнено.
2. Якщо значення в D0 дорівнює значенню в D2, M1 увімкнено.
3. Якщо значення в D0 менше, ніж у D2, M2 увімкнено.
4. Значення в D4 є додатним значенням.



Додаткові зауваження

Двійкове представлення значення та його абсолютне значення:

1. Чи є дані позитивним чи негативним значенням, залежить від значення старшого біта в регістрі. Якщо старший біт у регістрі дорівнює 0, дані є позитивним значенням. Якщо старший біт дорівнює 1, дані є від'ємним значенням.
2. Ви можете перетворити від'ємне значення в його абсолютне значення за допомогою інструкції NEG.

(D0)=2

0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(D0)=1

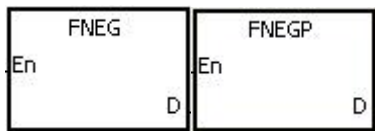
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(D0)=0

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

API		Код інструкції			Операнд						функція						
0211			FNEG	P	D						Зміна знака 32-розрядного числа з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			O						
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
D									●								
						Інструкція пульсу				16-бітна інструкція				32-розрядна інструкція			
						AS				-				AS			

СИМВОЛ



D : Пристрій, в якому знак величини є зворотним

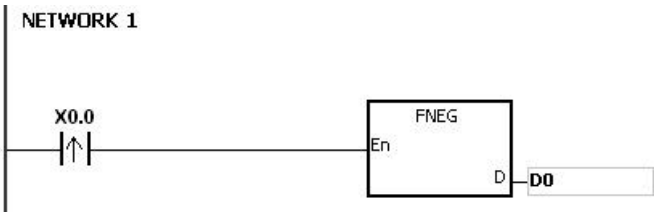
Пояснення

Ця інструкція змінює знак числа з плаваючою комою одинарної точності в D .

приклад

Перед виконанням інструкції значення в (D1, D0) є від'ємним значенням 16#AE0F9000. Коли X0.0 перемикається з OFF на ON, інструкція змінює знак числа з плаваючою комою одинарної точності в (D1, D0). Іншими словами, після виконання інструкції значення в (D1, D0) є позитивним значенням 16#2E0F9000.

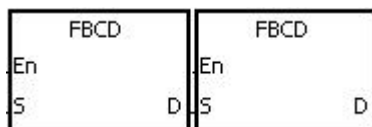
Перед виконанням інструкції значення в (D1, D0) є позитивним значенням 16#2E0F9000. Коли X0.0 перемикається з OFF на ON, інструкція змінює знак числа з плаваючою комою одинарної точності в (D1, D0). Іншими словами, після виконання інструкції значення в (D1, D0) є від'ємним значенням 16#AE0F9000.



API	Код інструкції			Операнд					функція							
0212		FBCD	P	S, D					Перетворення двійкового числа з плаваючою комою в десяткове число з плаваючою комою							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O					O
D		●			●	●		●			O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				
					Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція		
					AS			-			AS		

СИМВОЛ



S : Джерело даних

D : Результат перетворення

Пояснення

1. Ця інструкція перетворює число з плаваючою комою одинарної точності в регістрі, визначеному S , у десяткове число з плаваючою комою та зберігає результат перетворення в регістрі, визначеному D .

2. Операція з плаваючою комою в ПЛК базується на числах з плаваючою комою одинарної точності. Використовуйте FBCD інструкція для перетворення числа з плаваючою комою одинарної точності в десяткове число з плаваючою комою.

3. Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор) і SM602 (переносний прапор)

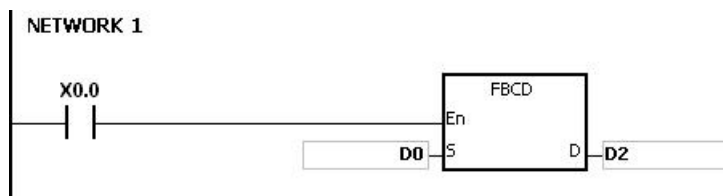
Якщо абсолютне значення результату перетворення перевищує значення, яке може бути представлено максимальним числом з плаваючою комою, SM602 увімкнено.

Якщо абсолютне значення результату перетворення менше значення, яке може бути представлено мінімальним числом з плаваючою комою, SM601 увімкнено.

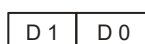
Коли результат перетворення дорівнює нулю, SM600 увімкнено.

приклад

Коли X0.0 увімкнено, інструкція перетворює число з плаваючою комою одинарної точності в (D1, D0) у десяткове число з плаваючою комою та зберігає результат перетворення в (D3, D2).



Двійкове число з плаваючою комою



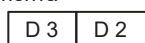
Реальне число: 23 біти; Експонента: 8 біт ts; знак: 1 біт



Експонента

Реальне число

Десяткове число з плаваючою комою



Математична форма

⇒ [D 2] x 10^[D 3]

Додаткові зауваження

Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції				Операнд				функція							
0213		FBIN	P		S, D				Перетворення десяткового числа з плаваючою комою у двійкове число з плаваючою комою							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●		●					
D		●			●	●	●	●			●					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

FBIN	FBINP
En	En
S	S
D	D

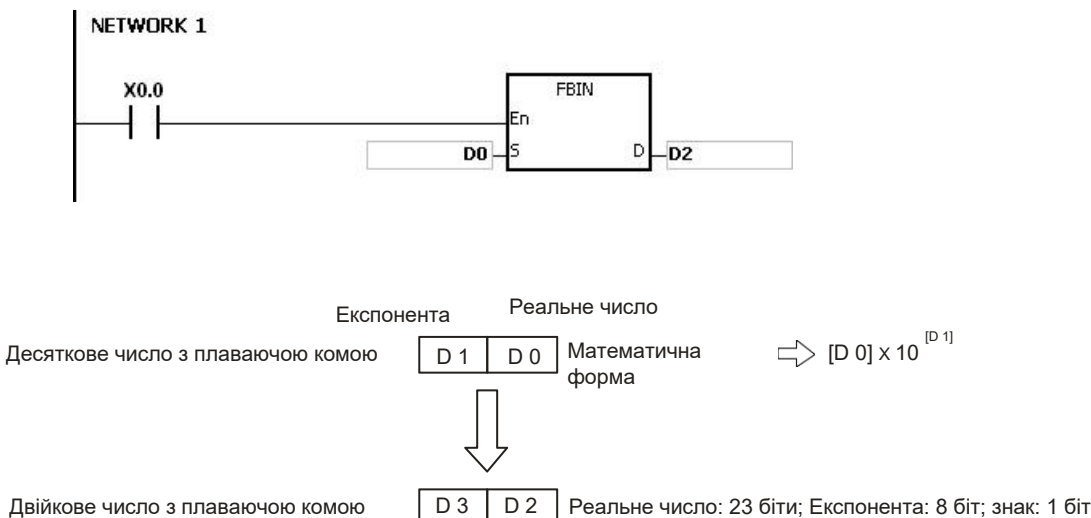
S : Джерело даних**D** : Результат перетворення

Пояснення

- Ця інструкція перетворює десяткове число з плаваючою комою в регістрі, заданому **S**, на одинарну точність число з плаваючою комою та зберігає результат перетворення в регістрі, визначеному **D**.
- Припустимо, що значення в **S** дорівнює 1234, а значення в **S + 1** дорівнює 3. Інструкція перетворює значення в **S** в $1,234 \times 10^{-6}$.
- Значення в **D** має бути числом з плаваючою комою одинарної точності, а значення в **S** і **S + 1** представляють десяткове дійсне число та десятковий експонент відповідно.
- Використовуйте інструкцію FBIN, щоб перетворити десяткове число з плаваючою комою в число з плаваючою комою одинарної точності.
- Дійсна кількість десяткових чисел з плаваючою комою становить від -9 999 до +9 999, показники десяткових чисел з плаваючою комою - від -41 до +35. Практичний діапазон десяткових чисел з плаваючою комою в ПЛК знаходиться між $\pm 1175 \times 10^{-41}$ та $\pm 3402 \times 10^{+35}$. Коли результат операції дорівнює нулю, SM600 увімкнено.

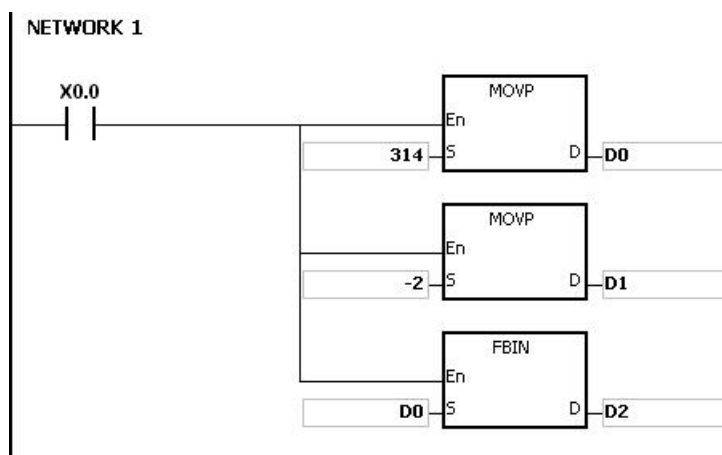
Приклад 1

1. Коли X0.0 увімкнено, інструкція перетворює десяткове число з плаваючою комою в регістрі (D1, D0) у число з плаваючою комою одинарної точності та зберігає результат перетворення в (D3, D2).



Приклад 2

1. Перед виконанням операції з плаваючою комою використовуйте інструкцію FLT, щоб перетворити двійкове ціле число в число з плаваючою комою одинарної точності. Перед перетворенням переконайтеся, що значення, яке потрібно перетворити, є двійковим цілим числом. Ви можете використовувати інструкцію FBIN, щоб перетворити число з плаваючою комою в число з плаваючою комою одинарної точності.
2. Коли X0.0 увімкнено, K314 і K-2 переміщуються до D0 і D1 відповідно, а потім FBIN об'єднує їх у десяткове число з плаваючою комою ($3,14=314 \times 10^{-2}$).



Додаткові зауваження

Якщо дійсна частина десяткового числа з плаваючою комою в **S** не знаходиться в межах від -9999 до +9999, або якщо експонента десяткового числа з плаваючою комою в **S** не знаходиться в межах від -41 до +35, інструкція не виконується. , SM0 увімкнено, а код помилки в SR0 – 16#2013.

API	Код інструкції				Операнд								функція				
0214		BKBCD	P		S, n, D								Перетворення двійкових чисел у блоках у двійкові десяткові числа в блоках				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●	●								
n	●	●			●	●		●	●				O	O			
D		●			●	●		●									
Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S		●				●											
n		●				●											
D		●				●											
				Інструкція пульсу				16-бітна інструкція				32-розрядна інструкція					
				AS				AS				-					

СИМВОЛ

BKBCD

En

S

n

BKBCDP

En

S

n

S

: Джерело даних

n

: Довжина даних

D

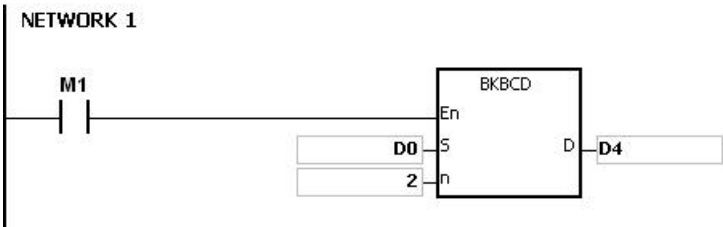
: Результат перетворення

Пояснення

1. Інструкція перетворює **n** фрагментів даних (двійкових значень), починаючи з **S** , у двійкові десяткові значення, і зберігає результати перетворення в **D** .
2. Операнд **n** має бути між 1–256.

приклад

Коли M1 увімкнено, інструкція перетворює двійкові значення в D0 та D1 у двійкові десяткові значення та зберігає результати перетворення в D4 та D5.



Додаткові зауваження

1. Якщо n менше 1 або більше 256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
2. Якщо пристрої, визначені $S + n - 1$ і $D + n - 1$, перевищують діапазон можливих пристроїв, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо результат перетворення не між 0–9999, інструкція не виконується, а код помилки в SR0 — 16#200D.
Двійкове десяткове значення представлено шістнадцятковим числом, але одна з цифр не знаходиться між 0–9.
4. Якщо $S - S + n - 1$ перекриває $D - D + n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200C.

API	Код інструкції			Операнд					функція							
0215		BKBIN	P	S, n, D					Перетворення двійкових чисел у блоках у двійкові десяткові числа в блоках							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●							
n	●	●			●	●		●	●				O	O		
D		●			●	●		●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



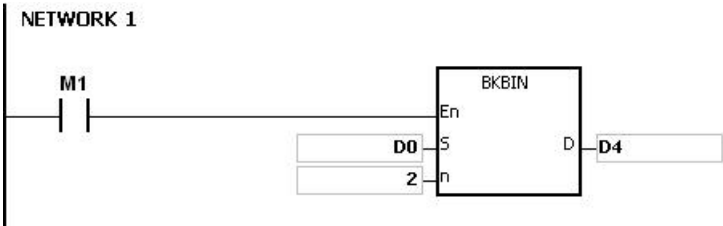
- S : Джерело даних
- n : Довжина даних
- D : Результат перетворення

Пояснення

- Інструкція перетворює n частин даних (двійкові десяткові значення), починаючи з S , у двійкові значення, і зберігає результати перетворення в D .
- Двійкове десяткове значення в S має бути між 0–9999.
- Операнд n має бути від 1 до 256.

приклад

Коли M1 увімкнено, інструкція перетворює десяткові значення двійкового коду в D0 та D1 у двійкові значення та зберігає результати перетворення в D4 та D5.



Додаткові зауваження

1. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
2. Якщо пристрої, визначені $S + n - 1$ і $D + n - 1$, перевищують діапазон можливих пристроїв, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Якщо дані в S не є десятковим у двійковому коді, інструкція не виконується, а код помилки в SR0 – 16#200D. Двійкове десяткове значення представлено шістнадцятковим числом, але одна з цифр не знаходиться між 0–9.
4. Якщо $S - S + n - 1$ перекриває $D - D + n - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200C.

API	Код інструкції			Операнд						функція							
0216		SCAL	P	S_1, S_2, S_3, D						Пошук масштабованого значення (точка-нахил)							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2	●	●			●	●		●	●		○	○	○	○		
S_3	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
S_3		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

SCAL	SCALP
En	En
S1 D	S1 D
S2	S2
S3	S3
DSCAL	DSCALP
En	En
S1 D	S1 D
S2	S2
S3	S3

 S_1 : Джерело даних S_2 : Схил S_3 : Зсув

D : Пристрій призначення

Пояснення

1. лінійне значення для числа в S_1 , використовуючи нахил лінії в S_2 і зміщення в S_3

для визначення коефіцієнта масштабування та збереження результату в D.

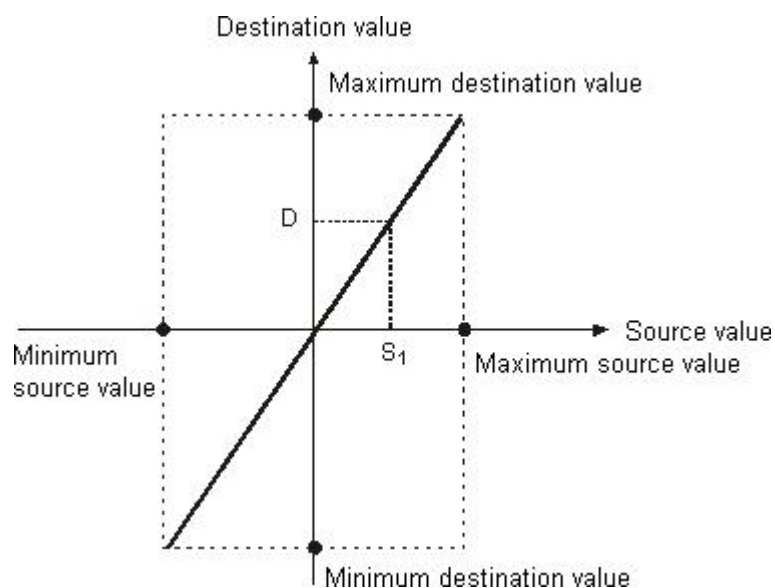
2. Операційне рівняння в інструкції таке: $D = (S_1 \times S_2) \div 1000 + S_3$

3. Щоб отримати значення в S_2 і S_3 , спочатку скористайтесь рівнянням нахилу та рівнянням зсуву, наведеним нижче, а потім округліть результати до найближчої цілої цифри. Введіть остаточні 16-бітні значення в S_2 і S_3 .

Рівняння нахилу: $S_2 = [(\text{Максимальне значення призначення} - \text{Мінімальне значення призначення}) \div (\text{Максимальне значення джерела} - \text{Мінімальне значення джерела})] \times 1000$

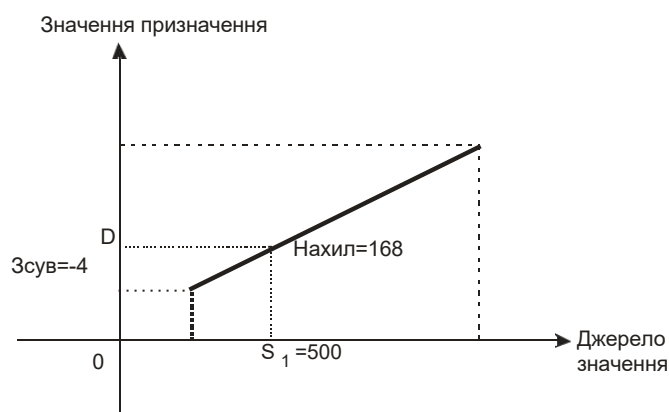
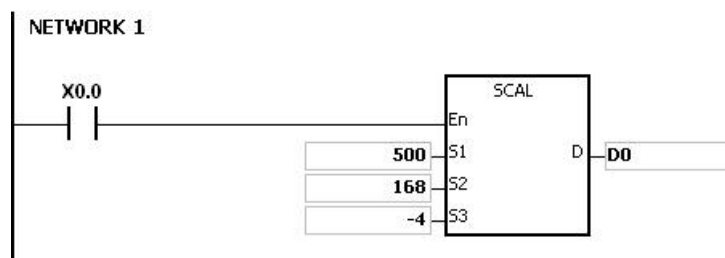
Рівняння зсуву: $S_3 = \text{Мінімальне значення призначення} - \text{Мінімальне значення джерела} \times S_2 \div 1000$

Вихідна крива показана нижче:



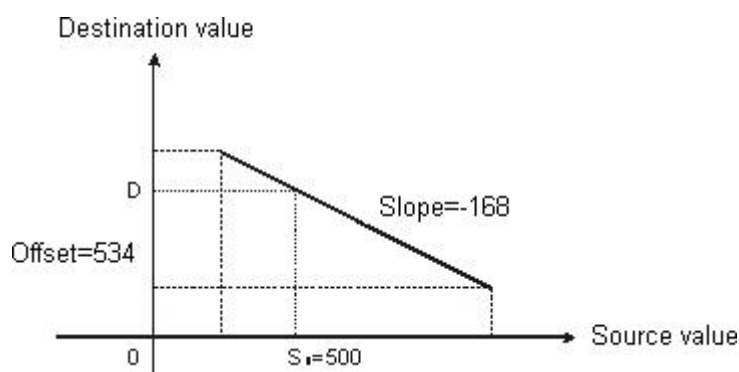
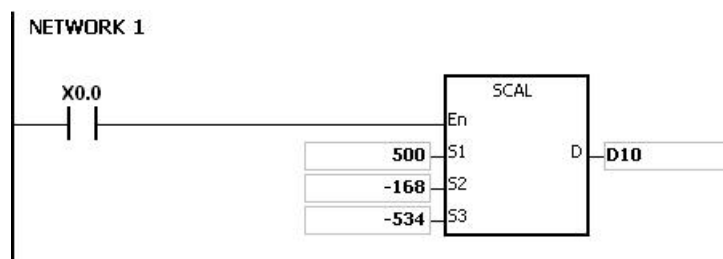
Приклад 1

1. Припустимо, що значення в S_1 , S_2 і S_3 дорівнюють 500, 168 і -4 відповідно. Коли X0.0 увімкнено, інструкція SCAL обчислює масштабоване значення та зберігає масштабоване значення в D0.
2. Для рівняння: $D0 = (500 \times 168) \div 1000 + (-4) = 80$



Приклад 2

1. Припустимо, що значення в S_1 , S_2 і S_3 дорівнюють 500, -168 і 534 відповідно. Коли X0.0 увімкнено, інструкція SCAL обчислює масштабоване значення та зберігає масштабоване значення в D10.
2. Для рівняння: $D10 = (500 \times -168) \div 1000 + 534 = 450$



Додаткові зауваження

1. Ви повинні знати нахил і зміщення, щоб використовувати SCAL. Якщо нахил і зсув невідомі, можна скористатися інструкцією SCLP.
2. Коли виконується 16-бітна інструкція, значення, введене в S_2 , має бути від -32 768 до 32 767. Якщо значення в S_2 перевищує діапазон, використовуйте інструкцію SCLP.
3. Коли виконується 32-розрядна інструкція, значення, введене в S_2 , має бути від -2 147 483 648 до 2 147 483 647. Якщо значення в S_2 перевищує діапазон, використовуйте інструкцію SCLP.
4. Коли ви використовуєте рівняння нахилу, зауважте, що максимальне значення джерела має бути більшим за мінімальне значення джерела. Однак максимальне значення призначення не обов'язково перевищує мінімальне значення призначення.
5. Для 16-бітної інструкції, якщо значення в D більше 32 767, значення, що зберігається в D , становить 32 767. Якщо значення в D менше ніж -32 768, значення, збережене в D , дорівнює -32 768.
6. Коли виконується 32-розрядна інструкція, якщо значення в D перевищує 2 147 483 647, значення, збережене в D , буде 2 147 483 647. Якщо значення в D менше за -2 147 483 648, значення, збережене в D , буде -2 147 483 648.

API	Код інструкції			Операнд							функція						
0217	D	SCLP	P	S_1, S_2, S_3, D							Знаходження масштабованого значення (два бали)						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		O	O	O	O		O
S_2	●	●			●	●	●	●	●							
D		●			●	●	●	●			O	O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●		●				
S_2		●	●			●	●		●				
D		●	●			●	●		●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

SCLP	SCLPP
En	En
S1	S1
S2	S2
D	D
DSCLP	DSCLPP
En	En
S1	S1
S2	S2
D	D

S_1 : Джерело даних

S_2 : Параметр

D : Пристрій призначення

Пояснення

- Ця інструкція знаходить масштабоване лінійне значення для значення в S_1 , використовуючи дві точки в S_2 для визначення коефіцієнта масштабування, і зберігає результат у D.
- Тільки 32-бітні інструкції можуть використовувати 32-бітний лічильник, але не пристрій E.
- У наведеній нижче таблиці наведено постійне використання для операнда S_1 .

Постійний	16-бітна інструкція	32-розрядна інструкція	
		SM685 ON	SM685 OFF
Постійний	O	X	O
Шістнадцятковий	O	X	O
Плаваюче число	X	O	X

Прапор SM685 (чи використовувати операцію з плаваючою комою чи ні) можна використовувати лише для 32-розрядних інструкцій.

4. Операнд S_2 , який використовується в 16-бітній інструкції, встановлюється, як показано в наступній таблиці.

Номер пристрою	Параметр	Діапазон налаштування
S_2	Максимальне вихідне значення	-32 768 до 32 767
S_2+1	Мінімальна вихідна вартість	-32 768 до 32 767
S_2+2	Максимальне значення призначення	-32 768 до 32 767
S_2+3	Мінімальне значення призначення	-32 768 до 32 767

5. Операнд S_2 , який використовується в 16-бітній інструкції, займає чотири пристрої.

6. Операнд S_2 , який використовується в 32-розрядній інструкції, встановлюється, як показано в наступній таблиці.

Номер пристрою	Параметр	Діапазон налаштування	
		Ціле число	Число з плаваючою комою
$S_2 \setminus S_2+1$	Максимальне вихідне значення	-2 147 483 648 до 2,147,483,647	Діапазон 32-розрядних чисел з плаваючою комою
$S_2+2, 3$	Мінімальна вихідна вартість		
$S_2+4, 5$	Максимальне значення призначення		
$S_2+6, 7$	Мінімальне значення призначення		

7. Операнд S_2 , який використовується в 32-розрядній інструкції, займає вісім пристроїв.

8. Якщо значення в 32-розрядній інструкції є числами з плаваючою комою, установіть SM658 на ON. Якщо значення десяткові цілі числа, встановіть для SM685 значення OFF.

9. Рівняння операції в інструкції таке:

$$D = [(S_1 - \text{Мінімальне значення джерела}) \times (\text{Максимальне значення призначення} - \text{Мінімальне значення призначення})] \div (\text{Максимальне значення джерела} - \text{Мінімальне значення джерела}) + \text{Мінімальне значення призначення}$$

10. Операційний зв'язок між значенням джерела та значенням призначення

такий: $y = kx + b$

y = цільове значення (D)

k = Нахил = $(\text{Максимальне значення призначення} - \text{Мінімальне значення призначення}) \div (\text{Максимальне значення джерела} - \text{Мінімальне значення джерела})$

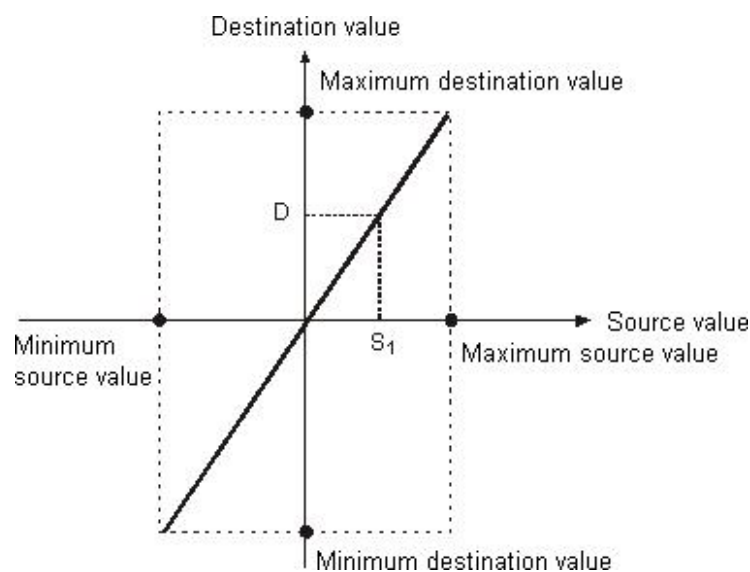
x = Вихідне значення (S_1)

b = Зсув = $\text{Мінімальне значення призначення} - \text{Мінімальне значення джерела} \times \text{Нахил}$

Наведені вище параметри замінюються на y , k , x і b у рівнянні $y = kx + b$, щоб отримати рівняння операції таким чином:

$$y = kx + b = D = k S_1 + b = \text{Нахил} \times S_1 + \text{Зсув} = \text{Нахил} \times S_1 + \text{Мінімальне цільове значення} - \text{Мінімальне значення джерела} \times \text{Нахил} = \text{Нахил} \times (S_1 - \text{Мінімальне значення джерела}) + \text{Мінімальне значення призначення} = (S_1 - \text{Мінімальне значення джерела}) \div (\text{Максимальне значення призначення} - \text{Мінімальне значення призначення}) \times (\text{Максимальне значення джерела} - \text{Мінімальне значення джерела}) + \text{Мінімальне значення призначення}$$

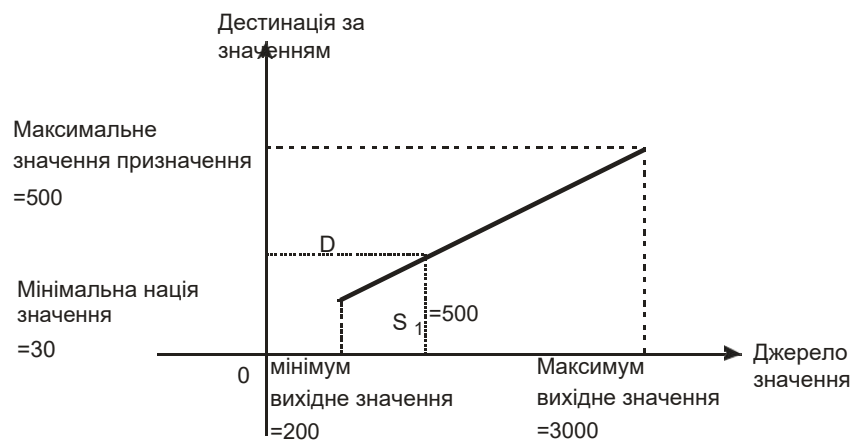
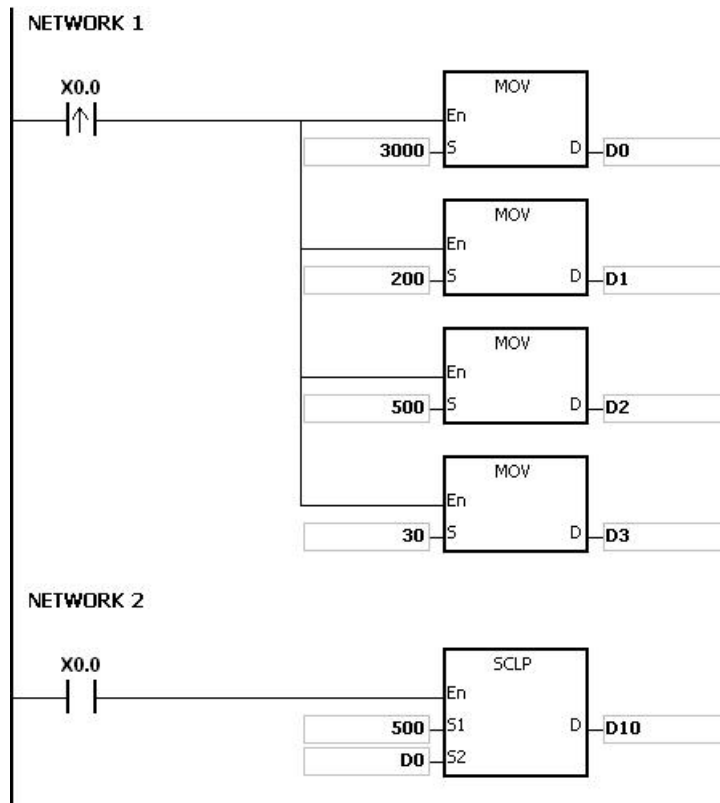
11. Якщо S_1 більше максимального значення джерела, максимальне значення джерела є значенням у S_1 . Якщо S_1 менше ніж мінімальне вихідне значення, мінімальне вихідне значення - це значення в S_1 . Вихідна крива показана нижче.



Приклад 1

- Припустимо, що значення в S_1 дорівнює 500, максимальне значення джерела в D0 становить 3000, мінімальне значення джерела в D1 становить 200, максимальне значення призначення в D2 становить 500, а мінімальне значення призначення в D3 дорівнює 30. Коли X0.0 ON, інструкція SCLP обчислює значення масштабу та зберігає його в D10.
- Операційне рівняння: $D10 = [(500 - 200) \times (500 - 30)] \div (3000 - 200) + 30 = 80,35$

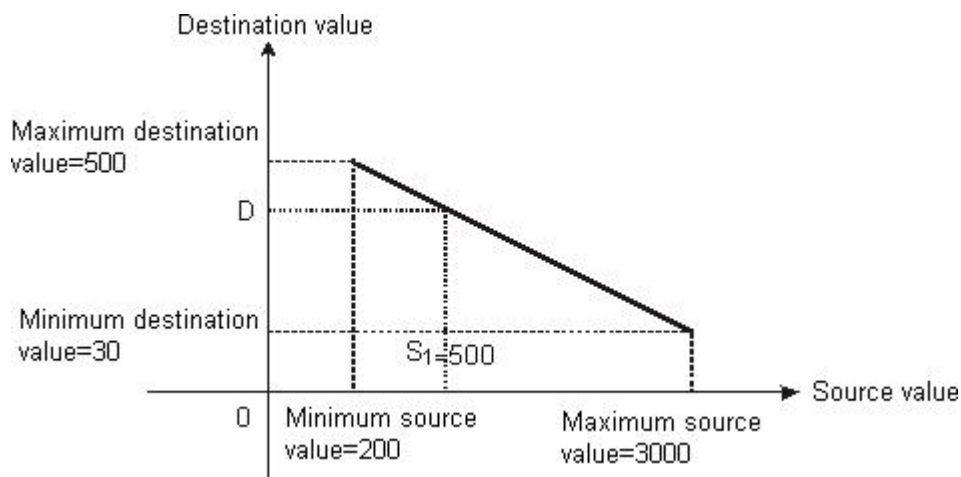
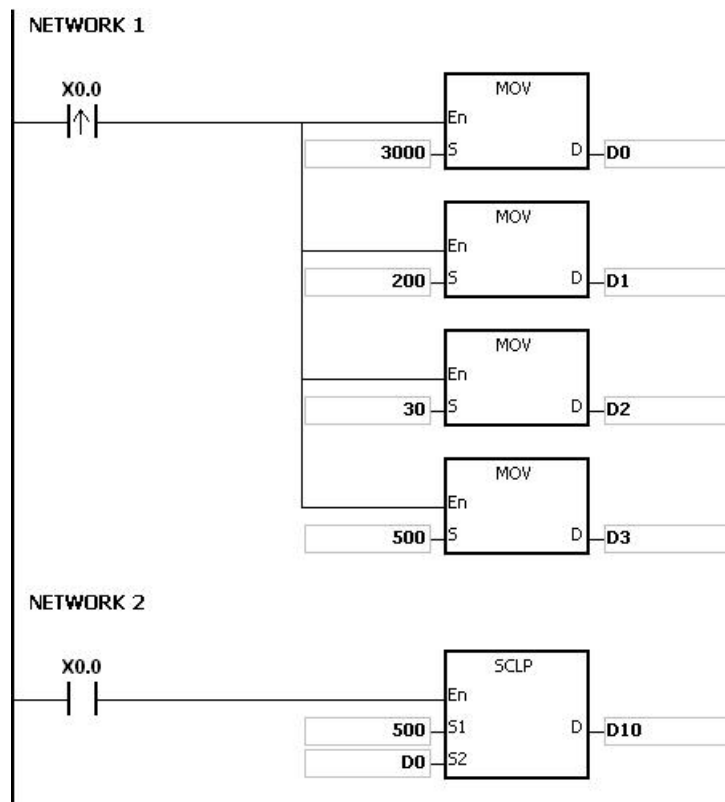
80,35 округлюється до найближчої цілої цифри і стає 80. 80 зберігається в D10.



Приклад 2

1. Припустимо, що значення в S_1 дорівнює 500, максимальне значення джерела в D0 становить 3000, мінімальне значення джерела в D1 становить 200, максимальне значення призначення в D2 становить 30, а мінімальне значення призначення в D3 становить 500. Коли X0.0 ON, інструкція SCLP обчислює значення шкали та зберігає його в D10.
2. Операційне рівняння: $D10 = [(500 - 200) \times (30 - 500)] \div (3000 - 200) + 500 = 449,64$

449,64 округлюється до найближчої цілої цифри і стає 450. 450 зберігається в D10.

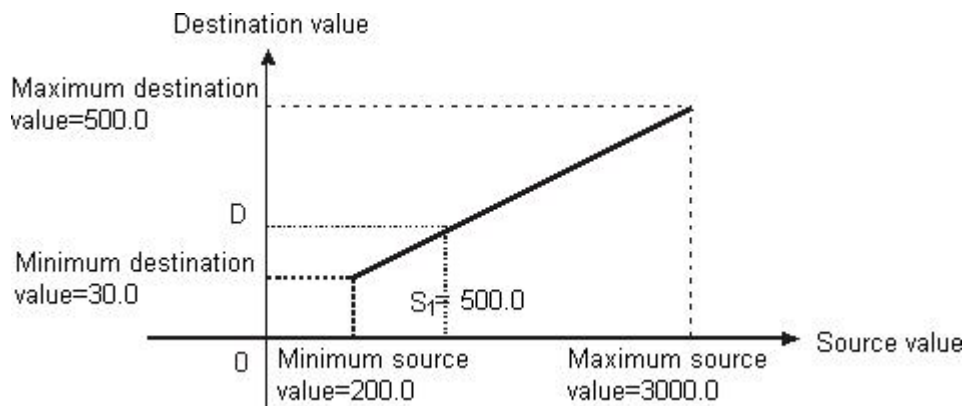
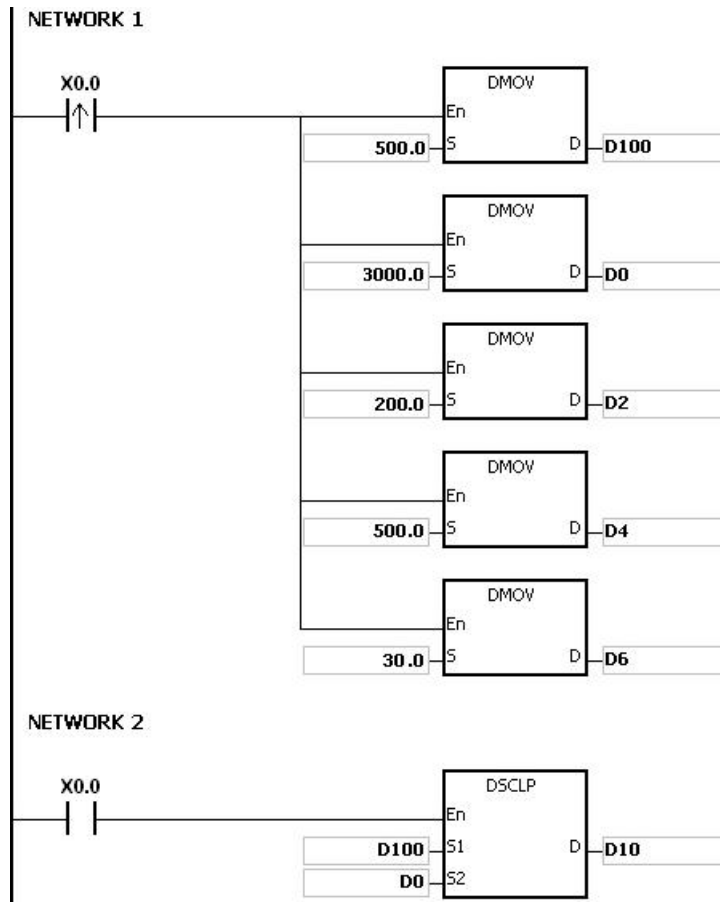


Приклад 3

1. Припустимо, що значення в S_1 дорівнює 500,0, максимальне значення джерела в D0 становить 3000,0, мінімальне значення джерела в D2 становить 200,0, максимальне значення призначення в D4 становить 500,0, а мінімальне значення призначення в D6 становить 30,0. Коли X0.0 увімкнено, SM685 увімкнено, інструкція DSCLP обчислює значення масштабу та зберігає його в D10.

2. Операційне рівняння: $D10 = [(500,0 - 200,0) \times (500,0 - 30,0)] \div (3000,0 - 200,0) + 30,0 = 80,35$

80,35 округлюється до найближчої цілої цифри і стає 80,0. 80,0 зберігається в D10.



Додаткові зауваження

1. Значення в S_1 для 16-бітових інструкцій має бути між мінімальним значенням джерела та максимальним значенням джерела; тобто від -32 768 до 32 767. Якщо значення перевищує граничне значення, обчислення використовує граничне значення.
2. Ціле число в S_1 для 32-розрядних інструкцій має бути між мінімальним значенням джерела та максимальним значенням джерела; тобто від -2 147 483 648 до 2 147 483 647. Якщо ціле число перевищує граничне значення, обчислення використовує граничне значення.

3. Число з плаваючою комою в S_1 для 32-розрядних інструкцій має бути між мінімальним значенням джерела та максимальним значенням джерела; тобто в діапазоні чисел з плаваючою комою. Якщо число з плаваючою комою перевищує граничне значення, обчислення використовує граничне значення.
4. Зауважте, що максимальне значення джерела має бути більшим за мінімальне значення джерела. Однак максимальне значення призначення не обов'язково перевищує мінімальне значення призначення.
5. Коли максимальне значення джерела збігається з мінімальним значенням джерела, інструкція виконується, SM0 увімкнено, а код помилки в SR0 – 16#2012.
6. Якщо ви оголошуєте S_2 для 16-бітної інструкції в ISPSOft, типом даних буде ARRAY [4] WORD.
7. Якщо ви оголошуєте S_2 для 32-розрядної інструкції в ISPSOft, типом даних є ARRAY [4] з DWORD.

API		Інструкція			Операнд							опис							
0222		D	SCLM		P	S ₁ , S ₂ , S ₃ , S ₄ , D							Багатоточкова операція співвідношення перетинів						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F			
S ₁					●	●	●	●											
S ₂					●	●	●	●					O	O					
S ₃					●	●	●	●	●										
S ₄					●	●	●	●	●										
D					●	●	●	●											

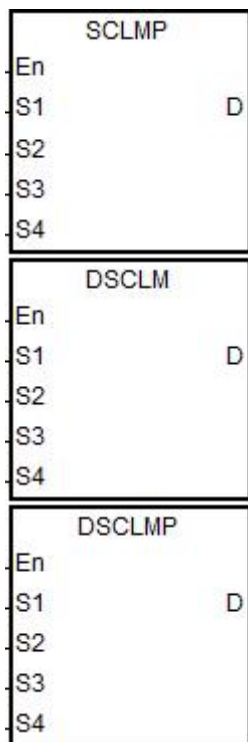
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●		●				
S ₂		●	●			●	●						
S ₃		●	●			●	●						
S ₄		●	●			●	●						
D		●	●			●	●		●				

Інструкція по пульсу						16-бітна інструкція				32-розрядна інструкція			
AS						AS				AS			

СИМВОЛ

SCLM	
En	
S1	D
S2	
S3	
S4	

- S_1 : Джерело даних
 S_2 : Кількість багатоточкових секцій
 S_3 : Значення порівняння в багатоточковому розрізі
 S_4 : Довідкове значення перетворення
D : Результат операції



Пояснення

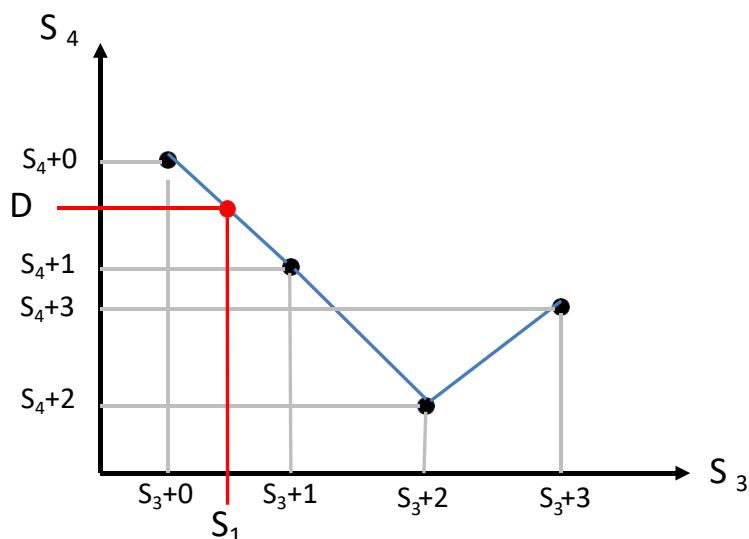
- Лише 32-розрядна інструкція може використовувати пристрої HC, але не пристрої E. Прошивка V1.04.00 і пізнішої версії для ПЛК AS300 підтримує цю інструкцію.
- Перегляньте наступну таблицю про типи даних, яким відповідають операнди S1, S3 і S4. (O означає «придатний для використання». X означає «непридатний для використання».)

Постійний	16-бітна інструкція	32-розрядна інструкція	
		SM685 ON	SM685 OFF
K	O	X	O
16#	O	X	O
F	X	O	X

Примітка: SM685=ON (операція з числами з плаваючою комою) працює лише для 32-розрядних інструкцій.

- S_1 є джерелом даних. S_2 – це кількість багатоточкових секцій, значення має бути від 2 до 50. Якщо значення перевищує діапазон, інструкція буде виконана автоматично при мініальному або максимальному значенні. S_3 – значення налаштування для порівняння в багатоточковій секції. S_4 є опорним значенням перетворення, якому відповідає значення порівняння багатоточкової секції, наприклад, кількість секцій, S_2 дорівнює 10. Тоді $S_3 \sim S_3 + 9$ є значеннями порівняння в 10 секціях. $S_4 \sim S_4 + 9$ є 10 відповідними еталонними значеннями перетворення.
- Порядок порівняння для багатоточкових ділянок 0, 1, 2 ... $S_2 - 1$. Правило порівняння $S_1 \geq S_3 + 0$ і $S_1 < S_3 + 1$. Якщо S_1 не належить розділу, порівняння перейде до наступного розділу. Наприклад, $S_1 \geq S_3 + 1$ і $S_1 < S_3 + 2$, порівняння продовжується, поки кількість разів порівняння не досягне $S_2 - 1$.

5. Інструкція порівнює розділи в порядку від малого до великого. Також розробіть порівняння значень розділу S_3 у порядку від малого до великого.
6. Дивіться малюнок нижче для перетворення значень багатоточкового розділу. (Установіть кількість секцій, S_2 до 4.)



7. Якщо значення S_1 знаходиться між S_3+0 і S_3+1 , формула перетворення: $D = ((S_1 - S_3 + 0) \times (S_4 + 1 - S_4 + 0)) / (S_3 + 1 - S_3 + 0) + S_4 + 0$.

8. Якщо значення S_1 не належить жодному вказаному розділу, результат виконання в D пояснюється нижче.

Якщо значення $S_1 >$ останній вказаний розділ, D збереже останнє контрольне значення перетворення S_4 , наприклад, якщо значення $S_1 > S_3+3$

значення на малюнку вище, $D = S_4 + 3$.

Якщо значення $S_1 <$ першого зазначеного розділу, D збереже перше контрольне значення перетворення S_4 , наприклад, якщо значення $S_1 < S_3+0$

значення на малюнку вище, $D = S_4 + 0$.

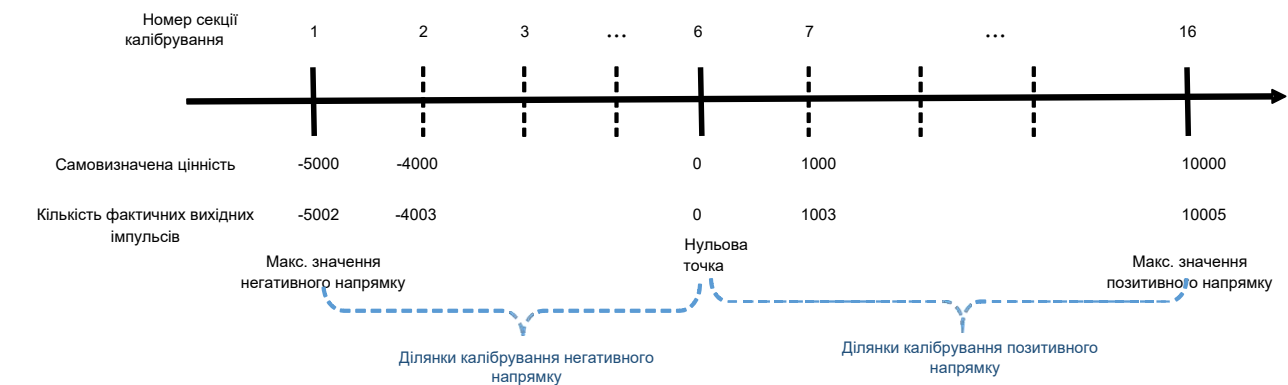
9. Якщо S_3 і S_4 16-бітної інструкції оголошено в ISPSOft, типом даних буде ARRAY [S2] WORD.

10. Якщо S_3 і S_4 32-розрядної інструкції оголошено в ISPSOft, тип даних – ARRAY [S2] DWORD (SM685=OFF) або ARRAY [S2] REAL (SM685=ON).

11. Для ЦП ПЛК із мікропрограмою версії 1.12.10 або новішої інструкція DSCLM (32-розрядна) підтримує застосування калібрування калібру різьби. І відмінності кільця різьби та діаметра кроку між фізичним параметром і кількістю вихідних імпульсів можна виправити.

12. Операційні кроки:

1) Виміряйте кожен діаметр кроку нарізного кільця та відповідну кількість вихідних імпульсів, а потім введіть виміряні значення в S_3 та S_4 відповідно.

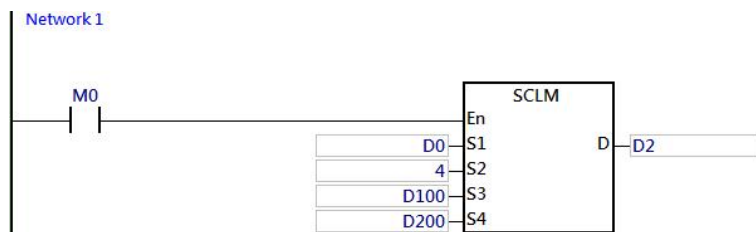


Номер секції калібрування	цільової позиції в S_3	Кількість фактичних вихідних імпульсів Значення в S_4
1	-5000 (D100, D101)	-5002 (D200, D201)
2	-4000 (D102, D103)	-4003 (D202, D203)
.		
6	0 (D110, D111)	0 (D210, D211)
7	1000 (D112, D113)	1003 (D212, D213)
.		
16	10000 (D130, D131)	10005 (D230 · D231)

2) Молодший байт у S_2 – це місце, де ви вказуєте номер розділу калібрування; можна встановити до 50 розділів. У той час як старший байт у S_2 є місцем, де ви вказуєте номер осі для виконання функції калібрування; значення у старшому байті може дорівнювати 0, що вказує на перетворення даних. Значення у старшому байті може бути від 1 до 12, що вказує на вихідні точки від Y0.0 до Y0.11, і виконується калібрування на вказаній вихідній точці. Операнди S_1 і D недейсні під час виконання калібрування.

3) Використовуйте контакт EN, щоб запустити інструкцію DSCLM (калібрування), і інструкція позиціонування діятиме відповідно до параметрів калібрування, щоб змінити кількість фактичних вихідних імпульсів у вказаній вихідній точці. Коли інструкція DSCLM закрита, калібрування закрито.

приклад



Порівняльні значення S_3 для багатоточкових перетинів наведені наступним чином.

пристрій	D100	D101	D102	D103
Зміст	100	200	300	400

Відповідні опорні значення перетворення S_4 наведені наступним чином.

пристрій	D200	D201	D202	D203
Зміст	4000	3000	1500	2000

Ось пояснення щодо значення в D2, отриманого шляхом перетворення на основі ресурсу даних D0.

Встановити $D0=10$,

Оскільки $D0 < D100$ (у першому розділі), $D2=D200=4000$ (перше контрольне значення перетворення)

Встановити $D0=K150$,

Значення $D0$ знаходиться між $(D100, D101) = (100, 200)$ і відповідним еталонним значенням $(D200, D201) = (4000, 3000)$ тому

$$D2 = (150-100) * (3000-4000) / (200-100) + 4000 = 3500$$

Встановити $D0=450$

Оскільки $D0 > D103$ (в останньому розділі), $D2=D203=2000$ (останнє контрольне значення перетворення)

Встановити $D0=K250$

Значення $D0$ знаходиться між $(D101, D102) = (200, 300)$ і відповідним еталонним значенням $(D201, D202) = (3000, 1500)$. тому

$$D2 = (250-200) * (1500-3000) / (300-200) + 3000 = 2250$$

Встановити $D0=K350$

$D0$ знаходиться між $(D102, D103) = (300, 400)$, а відповідне еталонне значення – $(D202, D203) = (1500, 2000)$ тому

$$D2 = (350-300) * (2000-1500) / (400-300) + 1500 = 1750$$

6.4 Інструкції з передачі даних

6.4.1 Список інструкцій щодо передачі даних

У наведеній нижче таблиці наведено інструкції з передачі даних, які розглядаються в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
0300	MOV	DMOV	✓	Передача даних
0302	\$MOV	–	✓	Перенесення рядка
0303	CML	DCML	✓	Інвертування даних
0304	BMOV	DBMOV	✓	Передача даних блоками
0305	NMOV	DNMOV	✓	Передача даних на кілька пристроїв
0306	XCH	DXCH	✓	Обмін даними
0307	BXCH	–	✓	Обмін даними блоками
0308	SWAP	DSWAP	✓	Обмін старшого байта на молодший
0309	SMOV	–	✓	Передача цифр блоками
0310	MOVB	–	✓	Передача бітів блоками

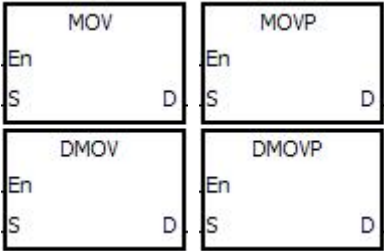
6.4.2 Пояснення інструкцій щодо передачі даних

API		Код інструкції			Операнд							функція						
0300		D	MOV		P	S, D							Передача даних					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S	●	●			●	●	●	●	●		○	○	○	○		○		
D		●			●	●	●	●			○	○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●		●		●	●	
D		●	●			●	●		●		●	●	

Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
AS		AS	AS

СИМВОЛ



S : Джерело даних

D : Призначення даних

Пояснення

- Ця інструкція передає дані з **S** в **D** .
- Ви повинні використовувати 32-розрядну інструкцію, якщо дані в **S** є числом з плаваючою комою.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

приклад

- Для передачі 16-бітних даних використовуйте MOV.
 - Коли X0.0 вимкнено, дані в D0 не змінюються. Коли X0.0 увімкнено, інструкція передає 10 до регістру даних D0.
 - Коли X0.1 вимкнено, дані в D10 не змінюються. Коли X0.1 увімкнено, інструкція передає поточне значення T0 до регістру даних D10.

2. Для 32-розрядних даних використовуйте DMOV.

- Коли X0.2 вимкнено, дані в (D31, D30) і (D41, D40) не змінюються. Коли X0.2 увімкнено, інструкція передає поточне значення з (D21, D20) до (D31, D30), а поточне значення HC0 передає до (D41, D40).

3. Для чисел з плаваючою комою використовуйте DMOV.

- Коли X0.3 вимкнено, дані в (D51, D50) не змінюються. Коли X0.3 увімкнено, інструкція перетворює число з плаваючою комою 3,450 у двійкове число з плаваючою комою та передає результат перетворення до (D51, D50).

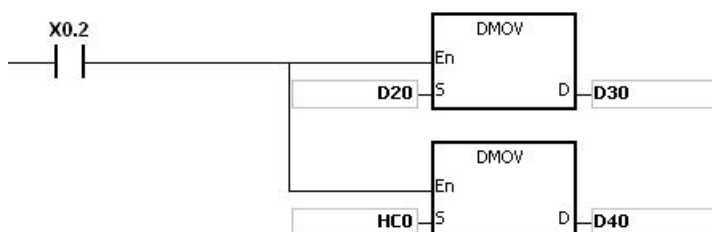
NETWORK 1



NETWORK 2



NETWORK 3



NETWORK 4



API	Код інструкції				Операнд						функція						
0302		\$MOV	P		S, D						Перенесення рядка						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●	●						○		
D		●			●	●		●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													●
D													●

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

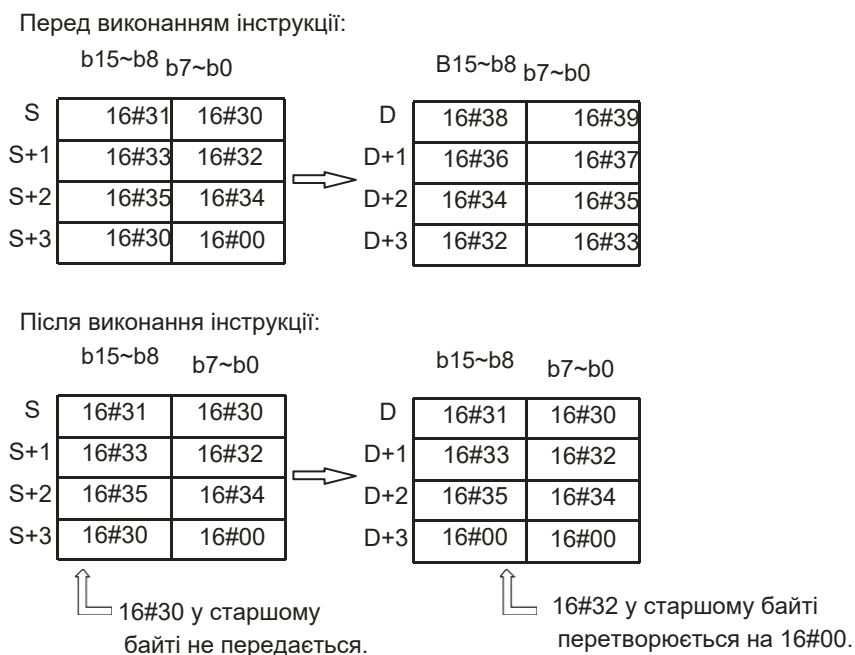


S : Джерело даних
D : Призначення даних

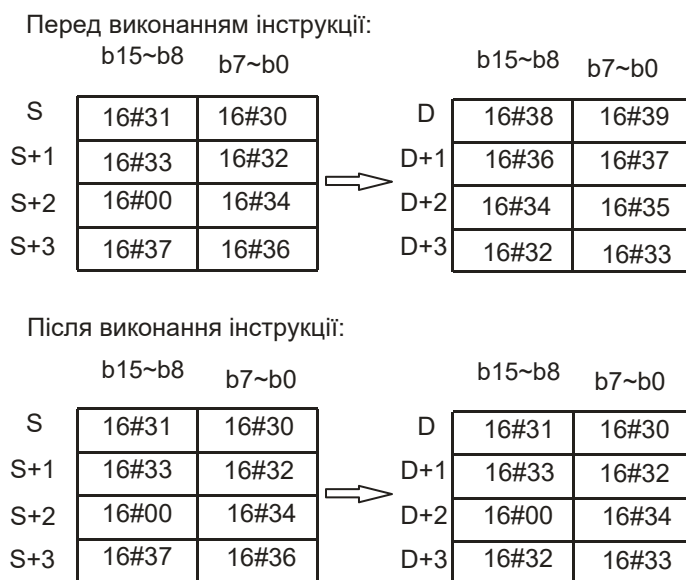
Пояснення

- Ця інструкція переносить рядок із **S** у **D** і додає код 16#00 у кінець рядка.
- Якщо операнд **S** не є рядком, інструкція додає код 16#00 у кінець переданих даних.
- Коли код закінчення 16#00 не може бути знайдений у **S** для 256 символів поспіль або навіть за межами діапазону пристрою,
вказівка не виконана; SM0 увімкнено, а код помилки в SR0 – 16#200E.
- Коли операнд **S** не є рядком і інструкція виконується, рядок, що починається з даних у пристрої, визначеному **S** (включаючи 16#00), передається в **D** . Коли інструкція не виконується, дані в **D** не змінюються.
- Якщо **D** недостатньо для вмісту рядка, що складається зі значень у **S** , інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
- Припустимо, що операнд **S** не є рядком. Коли інструкція виконується і перший символ у **S** є кодом 16#00, 16#00 все одно передається в **D** .

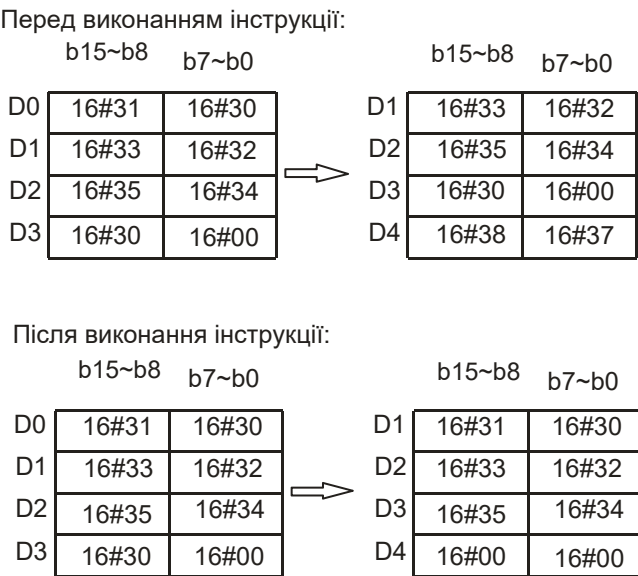
7. Коли в молодшому байті з'являється 16#00, виконання інструкції відбувається наступним чином.



8. Коли у старшому байті з'являється 16#00, виконання інструкції відбувається наступним чином. Передача припиняється, коли код 16#00, залишаючи залишок **D** без змін.

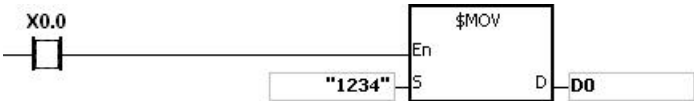


9. Коли **S** перекриває **D** і номер пристрою **S** менше, ніж номер пристрою **D** , передача даних на **D** починається з коду закінчення 16#00.



Приклад 1

Припустимо, що дані в **S** є рядком «1234» (парна кількість байтів). Коли X0.0 увімкнено, дані 1234 і кінцевий код 16#00 передаються в D0–D3, а 16#00 додається до старшого байта в **D** наступним чином.



Операнд **S** :

Рядок	'1'	'2'	'3'	'4'
Шістнадцятковий значення	16#31	16#32	16#33	16#34

Після виконання інструкції дані в **D** виглядають наступним чином.

пристрій	Старший байт	Молодший байт	Примітка
D0	16#32	16#31	'1'=16#31; '2'=16#32
D1	16#34	16#33	'3'=16#33; '4'=16#34
D2	16#00	16#00	Кінцевий код 16#00 знаходиться в молодшому байті. 16#00 автоматично додається до старшого байту.
D3	Без змін	Без змін	

Приклад 2

Припустимо, що дані в **S** є рядком «12345» (непарна кількість байтів). Коли X0.0 увімкнено, дані 12345 передаються в D0–D3 наступним чином.



Операнд **S** :

Рядок	'1'	'2'	'3'	'4'	"5"
Шістнадцятковий значення	16#31	16#32	16#33	16#34	16#35

Після виконання інструкції дані в операнді **D** виглядають наступним чином.

пристрій	Старший байт	Молодший байт	Примітка
D0	16#32	16#31	'1'=16#31; '2'=16#32
D1	16#34	16#33	'3'=16#33; '4'=16#34
D2	16#00	16#35	Кінцевий код 16#00 знаходиться у старшому байте.
D3	Без змін	Без змін	

Приклад 3

Коли дані в **S** не є рядком і кінцевий код 16#00 з'являється в молодшому байті, виконання інструкції відбувається наступним чином.



Операнд **S** :

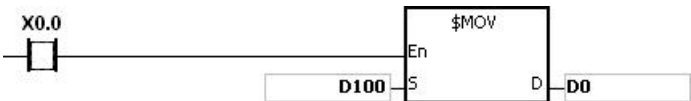
пристрій	Старший байт	Молодший байт	Примітка
D100	16#31	16#30	'1'=16#31; '0'=16#30
D101	16#33	16#32	'3'=16#33; '2'=16#32
D102	16#35	16#34	'5'=16#35; '4'=16#34
D103	16#30	16#00	'0'=16#30; 16#00 - код закінчення.

Після виконання інструкції дані в операнді **D** виглядають наступним чином.

пристрій	Старший байт	Молодший байт	Примітка
D0	16#31	16#30	'1'=16#31; '0'=16#30
D1	16#33	16#32	'3'=16#33; '2'=16#32
D2	16#35	16#34	'5'=16#35; '4'=16#34
D3	16#00	16#00	Кінцевий код 16#00 знаходиться в молодшому байті. 16#00 автоматично додається до старшого байту.
D4	Без змін	Без змін	

Приклад 4

Коли дані в **S** не є рядком і кінцевий код 16#00 з'являється у старшому байті, виконання інструкції відбувається наступним чином.



Операнд **S** :

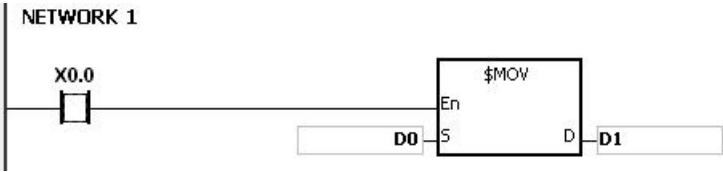
пристрій	Старший байт	Молодший байт	Примітка
D100	16#31	16#30	'1'=16#31; '0'=16#30
D101	16#33	16#32	'3'=16#33; '2'=16#32
D102	16#00	16#34	16#00 - код закінчення. '4'=16#34
D103	16#37	16#36	'7'=16#37; '6'=16#36

Після виконання інструкції дані в операнді **D** виглядають наступним чином.

пристрій	Старший байт	Молодший байт	Примітка
D0	16#31	16#30	'1'=16#31; '0'=16#30
D1	16#33	16#32	'3'=16#33; '2'=16#32
D2	16#00	16#34	16#00 - код закінчення. '4'=16#34
D3	Без змін	Без змін	

Приклад 5

Коли **S** перекриває **D** , а номер пристрою **S** менше, ніж номер пристрою **D** , передача даних до **D** починається з кінцевого коду 16#00.



Операнд **S** :

пристрій	Старший байт	Молодший байт	Примітка
D0	16#31	16#30	'1'=16#31; '0'=16#30
D1	16#33	16#32	'3'=16#33; '2'=16#32
D2	16#35	16#34	'5'=16#35; '4'=16#34
D3	16#30	16#00	'0'=16#30; 16#00 - код закінчення.
D4	16#38	16#37	'8'=16#38; '7'=16#37

6

Після виконання інструкції дані в **D** виглядають наступним чином.

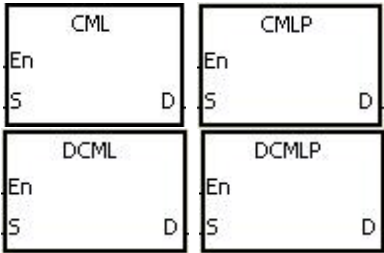
пристрій	Старший байт	Молодший байт	Примітка
D1	16#31	16#30	'1'=16#31; '0'=16#30
D2	16#33	16#32	'3'=16#33; '2'=16#32
D3	16#35	16#34	'5'=16#35; '4'=16#34
D4	16#00	16#00	Кінцевий код 16#00 знаходиться в молодшому байті. 16#00 автоматично додається до старшого байту.
D5	Без змін	Без змін	

API	Код інструкції			Операнд							функція						
0303	D	CML	P	S, D							Інвертування даних						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O	O	O	O			
D		●			●	●	●	●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



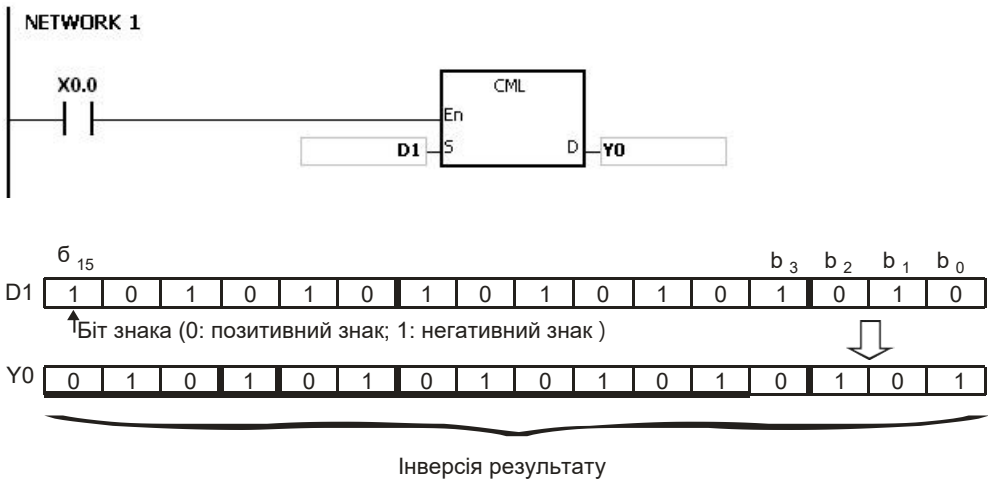
S : Джерело даних
D : Призначення даних

Пояснення

- Ця інструкція інвертує всі біти в **S** ; тобто 0 стає 1, а 1 стає 0, і зберігає результат інверсії в **D** . Якщо дані в **S** є константою, інструкція перетворює їх у двійкове значення.
- Тільки 32-бітні інструкції можуть використовувати 32-бітний лічильник, але не пристрій E.

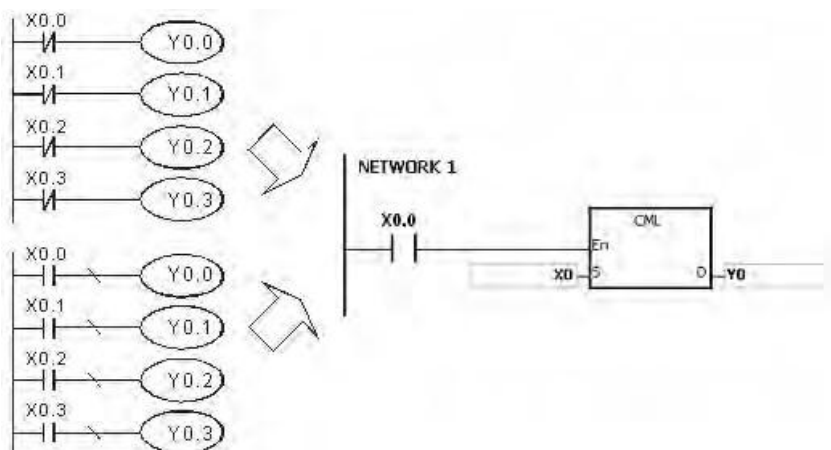
Приклад 1

Коли X0.0 увімкнено, інструкція інвертує всі біти в D1 і зберігає результат перетворення в Y0.0–Y0.15.



Приклад 2

Наведені нижче схеми можна представити за допомогою інструкції CML.

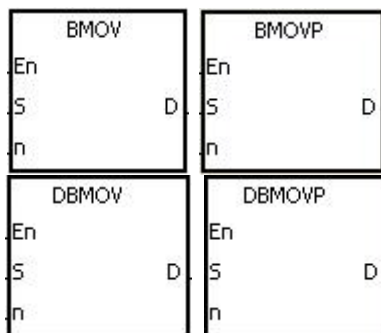


API	Код інструкції			Операнд						функція						
0304	D	BMOV	P	S, D, n						Передача даних блоками						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●							
D		●			●	●	●	●								
n	●	●			●	●	●	●	●				○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

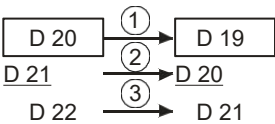


- S** : Джерело даних
D : Призначення даних
n : Довжина даних

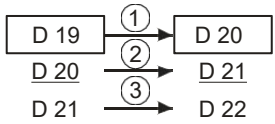
Пояснення

- Ця інструкція передає **n** фрагментів даних у блоці, починаючи з пристрою, зазначеного **S**, до пристроїв, починаючи з пристрою, зазначеного **D**.
- Значення в **n** має бути між 1–256.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.
- Щоб запобігти накладанню джерела та призначення, інструкція передає дані таким чином (використовуючи 16-бітну інструкцію як приклад).

Коли номер пристрою **S** більший за номер пристрою **D**, дані передаються в порядку від ① до ③.

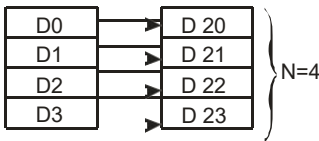
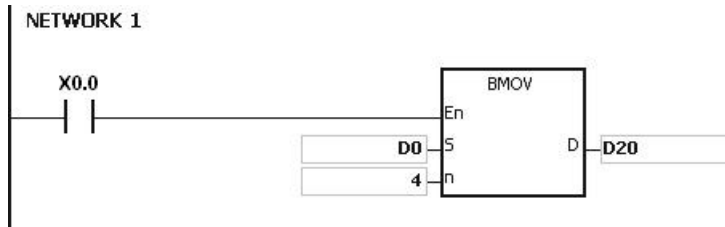


Коли номер пристрою **S** менше, ніж номер пристрою **D** , дані передаються в порядку від ③ до ① .



Приклад 1

Коли X0.0 увімкнено, інструкція передає дані з D0–D3 на D20–D23.

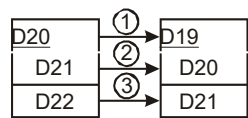
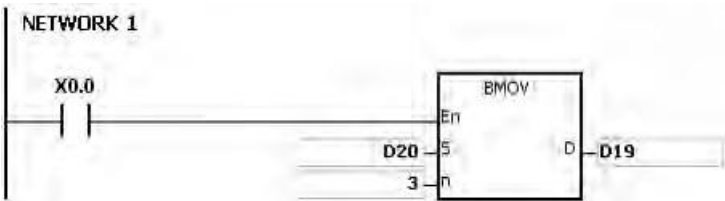


6

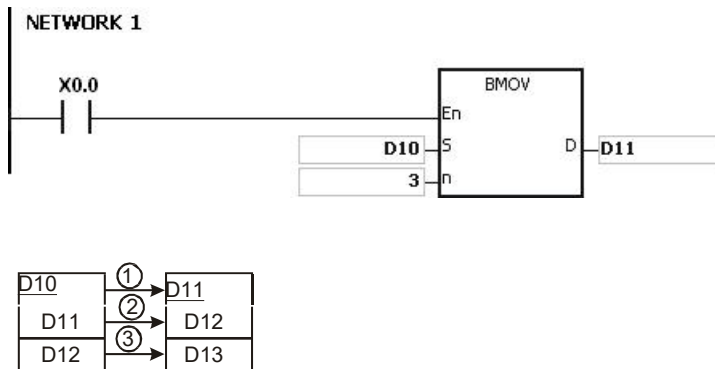
Приклад 2

Щоб запобігти накладенню джерела та призначення, дані передаються таким чином.

1. Якщо номер пристрою **S** перевищує номер пристрою **D** , дані передаються в порядку від ① до ③ .



2. Якщо номер пристрою **S** менше, ніж номер пристрою **D**, дані передаються в порядку від ③ до ①.



Додаткові зауваження

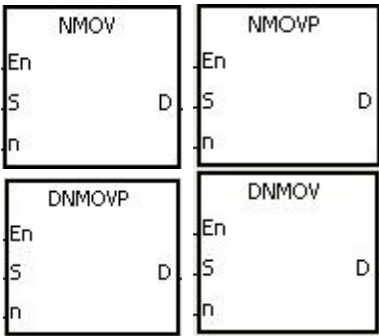
1. Якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо $S + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд							функція						
0305	D	NMOV	P	S, D, n							Передача даних на кілька пристроїв						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●				○	○			
D		●			●	●	●	●									
n	●	●			●	●	●	●	●				○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

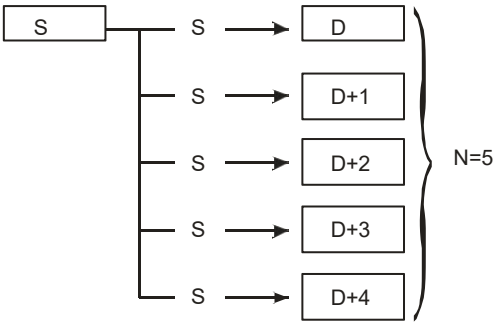
СИМВОЛ



- S : Джерело даних
- D : Призначення даних
- n : Довжина даних

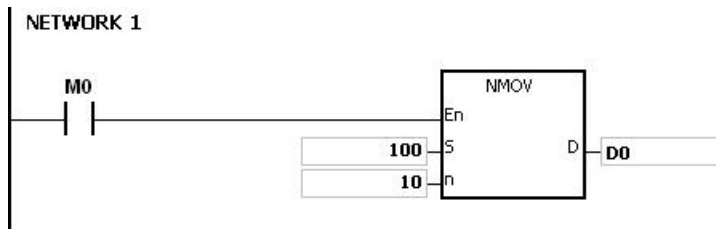
Пояснення

- Ця інструкція передає дані в S на n пристроїв, починаючи з пристрою, зазначеного D . Коли інструкція не виконується, дані в D не змінюються.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник.
- Значення n в інструкції NMOV має бути між 1–256.



приклад

Коли M0 увімкнено, 100 передається на D0-D9.

**Додаткові зауваження**

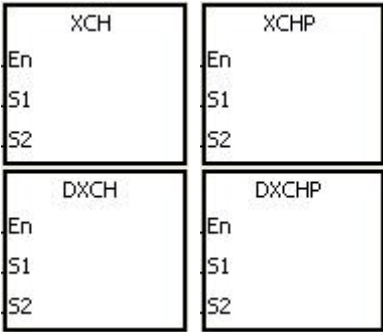
1. Якщо $D - D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо значення в **n** у 16-бітній інструкції не знаходиться в межах 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд							функція						
0306	D	XCH	P	S ₁ , S ₂							Обмін даними						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁		●			●	●	●	●				O					
S ₂		●			●	●	●	●				O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●						
S ₂		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



S₁ : Дані для обміну

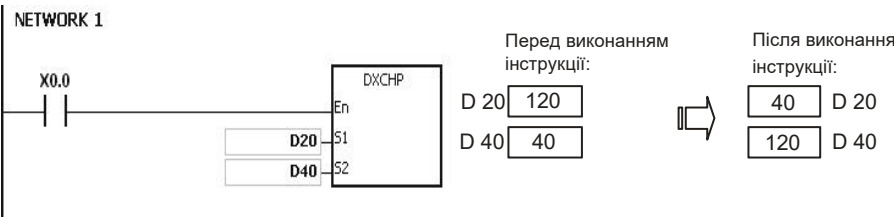
S₂ : Дані для обміну

Пояснення

- 1. Ця інструкція обмінюється даними в пристрої, визначеному S₁, даними в пристрої, визначеному S₂.
- 2. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

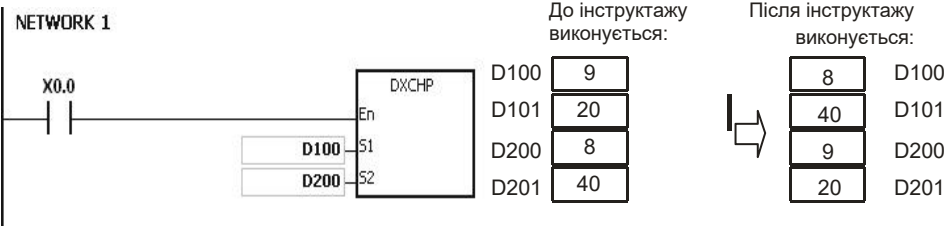
Приклад 1

Коли X0.0 перемикається з OFF на ON, інструкція обмінюється даними в D20 з даними в D40.



Приклад 2

Коли X0.0 перемикається з OFF. на ON., інструкція обмінюється даними в D100 з даними в D200.



API	Код інструкції				Операнд						функція						
0307		BXCH	P		S_1, S_2, n						Обмін даними блоками						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1		●			●	●		●									
S_2		●			●	●		●									
n	●	●			●	●		●	●				O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
n		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
AS		AS		-	

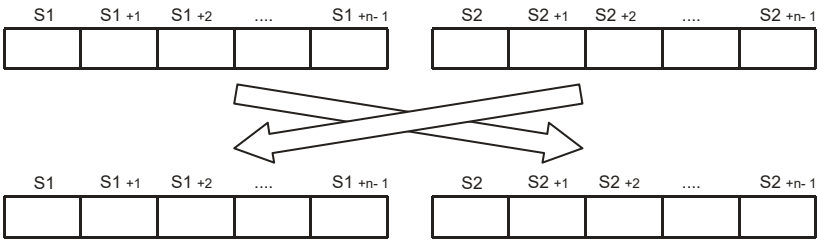
СИМВОЛ

BXCH		BXCHP	
En		En	
S1		S1	
S2		S2	
n		n	

- S_1 : Дані для обміну
- S_2 : Дані для обміну
- n : Довжина даних

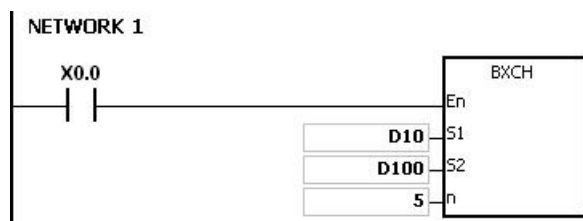
Пояснення

1. Ця інструкція обмінює дані в $S_1 - S_1 + n - 1$ з даними в $S_2 - S_2 + n - 1$.
2. Значення в n має бути між 1–256.



приклад

Коли X0.0 увімкнено, інструкція обмінюється даними в D10–D14 з даними в D100–D104.



Додаткові зауваження

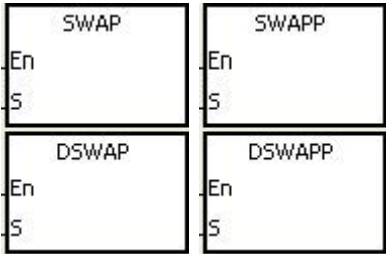
1. Якщо $S_1 + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо $S_2 + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо значення в n не знаходиться в межах 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 є 16#200Б.

API	Код інструкції			Операнд							функція						
0308	D	SWAP	P	S							Обмін старшого байта на молодший						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S		●			●	●	●	●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



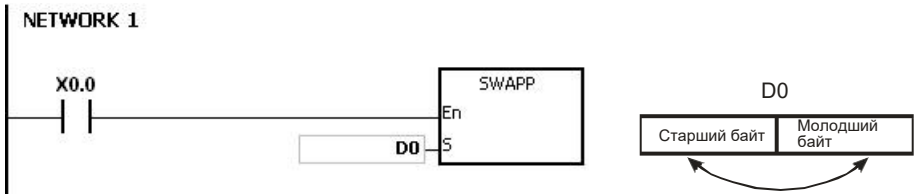
S : Джерело даних

Пояснення

- 1. 16-бітна інструкція обмінює дані в молодшому байті в S з даними в старшому байті в S .
- 2. 32-розрядна інструкція обмінюється даними в молодшому байті старшого слова в S з даними в старшому байті старше слово в S і обмінює дані в молодшому байті молодшого слова в S з даними в старшому байті молодшого слова в S .
- 3. Тільки 32-бітні інструкції можуть використовувати 32-бітний лічильник, але не пристрій E.

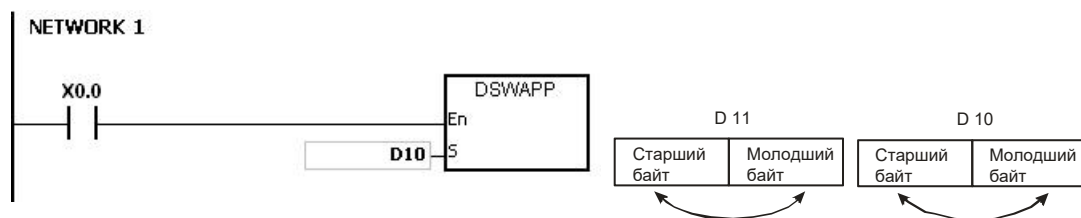
Приклад 1

Коли X0.0 увімкнено, інструкція обмінює дані в молодшому байті в D0 з даними у старшому байті в D0.



Приклад 2

Коли X0.0 увімкнено, інструкція обмінюється даними в молодшому байті в D11 з даними у старшому байті в D11, а також обмінюється даними в молодшому байті в D10 з даними у старшому байті в D10.

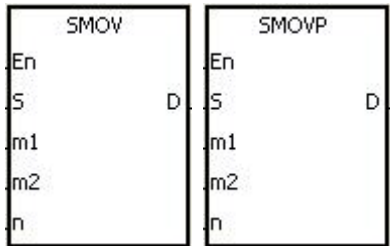


API	Код інструкції				Операнд						функція					
0309		SMOV	P	S, m ₁ , m ₂ , D, n						Передача цифр блоками						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●				○	○		
m ₁	●	●			●	●		●	●				○	○		
m ₂	●	●			●	●		●	●				○	○		
D		●			●	●		●								
n	●	●			●	●		●	●				○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
m ₁		●				●							
m ₂		●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



- S** : Джерело даних
- m₁** : Початкова цифра для передачі з вихідного пристрою
- m₂** : Кількість цифр для перенесення
- D** : Призначення даних
- n** : Початкова цифра, у якій зберігаються вихідні дані на пристрої призначення

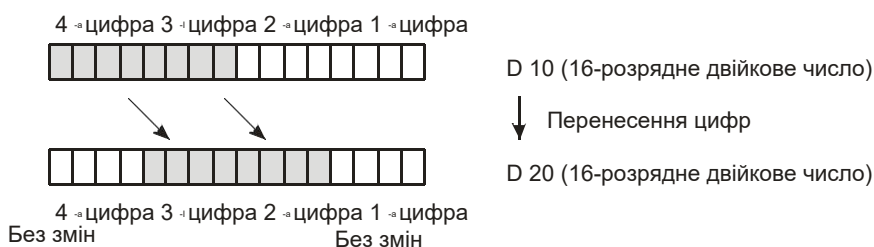
Пояснення

- Ця інструкція розподіляє та об'єднує дані. Інструкція передає **m₂** цифр числа, починаючи з **m₁** -ї цифри числа в **S** до **m₂** цифр числа, починаючи з **n** -ї цифри числа в **D**.
- Значення в **m₁** має бути в межах 1–4. Значення в **m₂** має бути між 1–**m₁**. Значення в **n** має бути між **m₂**–4 (інструкція розглядає чотири біти як одиницю.)
- Коли SM605 вимкнено, дані в **S** являють собою двійкові десятикові числа.



Припустимо, що число в **S** — K1234, а число в **D** — K5678. Після виконання інструкції число в **S** дорівнює 1234, а число в **D** дорівнює 5128.

4. Коли SM605 увімкнено, дані, задіяні в інструкції, є двійковими числами.

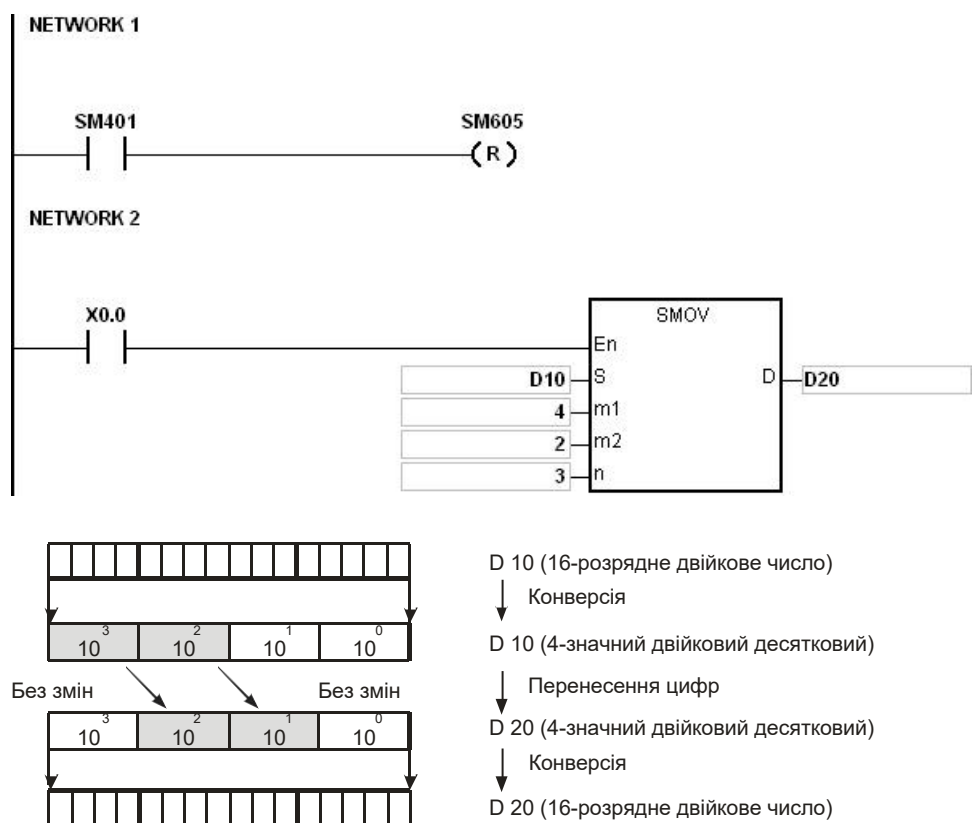


Припустимо, що число в **S** — 16#1234, а число в **D** — 16#5678. Після того, як інструкція буде виконана, число в **S** — 16#1234, а число в **D** — 16#5128.

6

Приклад 1

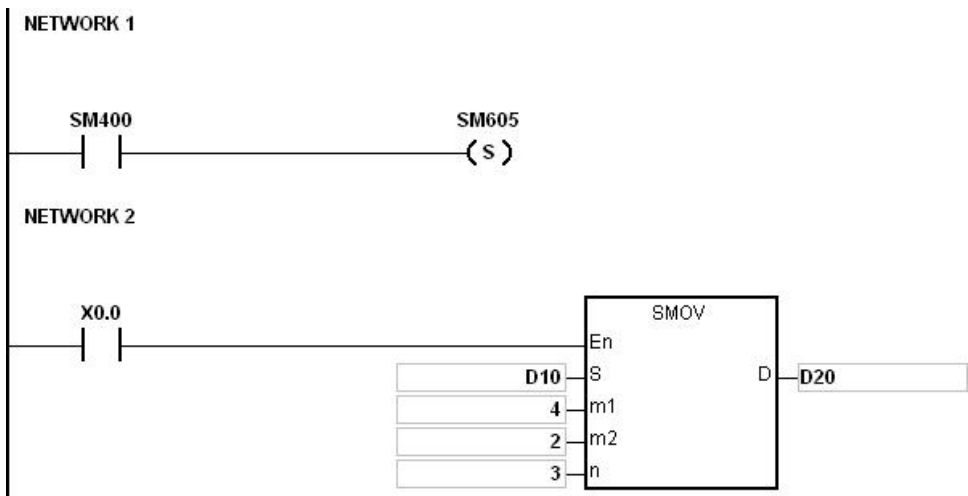
- Коли SM605 вимкнено, дані в **S** являють собою двійкові десяткові числа. Коли X0.0 увімкнено, інструкція передається дві цифри десяткового числа, починаючи з четвертої цифри десяткового числа (цифра в розряді тисяч десяткового числа) у D10 до двох цифр десяткового числа, починаючи з третьої цифри десяткового числа (цифра в розряд сотень десяткового числа) у D20. Після виконання інструкції цифри в розряді тисяч десяткового числа (10^3) і в розряді одиниць десяткового числа (10^0) у D20 залишаються незмінними.
- Якщо десяткове число в двійковому коді не знаходиться в межах 0–9999, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200D.

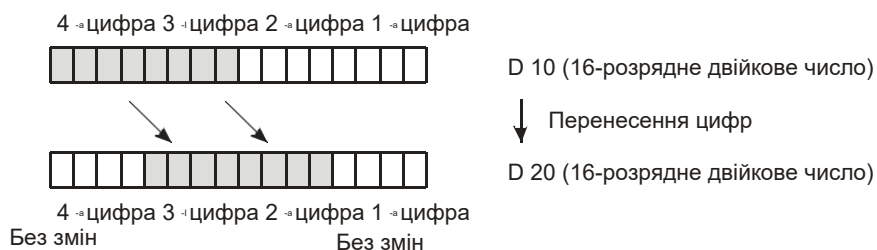


Припустимо, число в D10 дорівнює 1234, а число в D20 дорівнює 5678. Після виконання інструкції число у D10 не змінюється, а число в D20 дорівнює 5128.

Приклад 2

Коли SM605 увімкнено, дані є двійковими числами. Інструкція SMOV передає цифру, що складається з чотирьох бітів. інструкція не перетворює дані в двійкові десяткові числа.

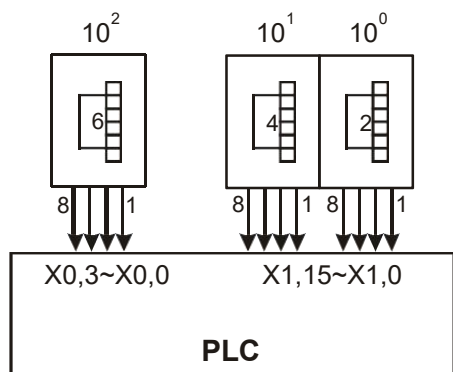


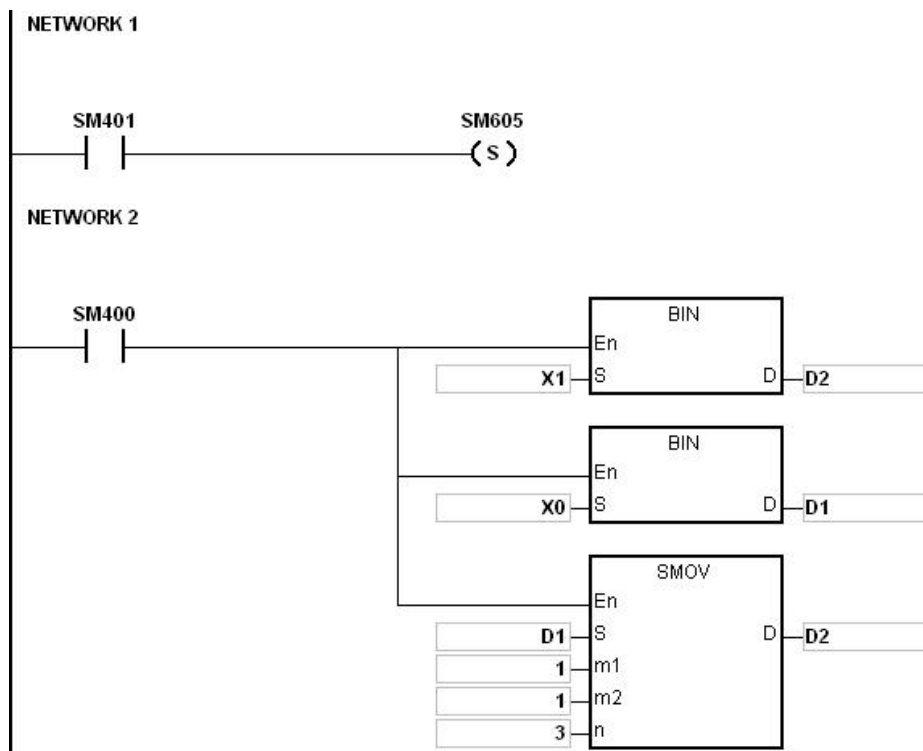


Припустимо, що число в D10 — 16#1234, а число в D20 — 16#5678. Після того, як інструкція буде виконана, число в D10 не змінилося, а число в D20 — 16#5128.

Приклад 3

1. Ви можете використати інструкцію для комбінування значень DIP-перемикачів, підключених до вхідних клем, номери яких не є послідовними.
2. Дві цифри значення DIP-перемикача праворуч переносяться до двох цифр числа, починаючи з другої цифри числа в D2, а одна цифра значення DIP-перемикача ліворуч переносяться до першої цифри числа в D1.
3. Ви можете використовувати інструкцію SMOV, щоб перенести першу цифру числа в D1 до третьої цифри числа в D2. Іншими словами, два DIP-перемикачі можна об'єднати в один DIP-перемикач за допомогою інструкції SMOV.



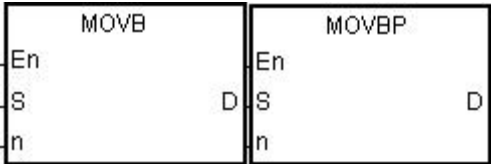


Додаткові зауваження

1. Припустимо, що дані є десятковими числами з двійковим кодом. Якщо число в **S** не між 0–9999 або якщо число в **D** не між 0–9999, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200D.
2. Якщо m_1 менше 1 або якщо m_1 більше 4, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
3. Якщо m_2 менше за 1 або якщо m_2 більше за m_1 , інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
4. Якщо n менше m_2 або якщо n більше 4, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.

API	Код інструкції				Операнд						функція						
0310		MOVB	P		S, n, D						Передача бітів блоками						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●	●	●	●	●	●	●		●							
n	●	●			●	●		●	●		O	O	O	O			
D		●	●	●				●									
Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S	●																
n		●				●											
D	●																
				Інструкція пульсу				16-бітна інструкція				32-розрядна інструкція					
				AS				AS				-					

СИМВОЛ



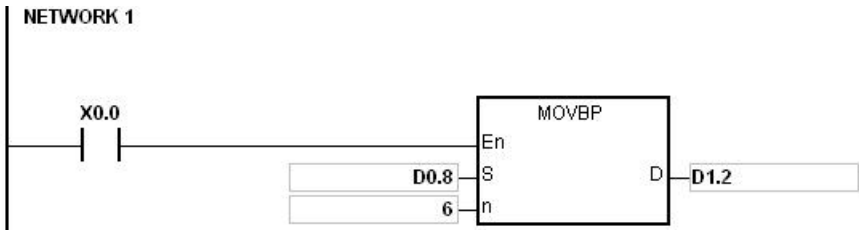
- S : Джерело даних
- n : Довжина даних
- D : Призначення даних

Пояснення

- 6
- Ця інструкція передає n фрагментів даних у пристроях, починаючи з пристрою, указанного S , до пристроїв, які починаються з пристрою, зазначеного D .
 - Коли S є T, C або HC, інструкція передає лише стан пристрою, але не передає поточне значення пристрою.
 - Значення в n має бути між 1–256. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

приклад

Коли X0.0 увімкнено, інструкція передає дані з D0.8–D0.13 на D1.2–D1.7.



Додаткові зауваження

1. Якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо $S + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

6.5 Інструкції зі стрибка

6.5.1 Список інструкцій зі стрибка

У наведеній нижче таблиці наведено інструкції щодо переходу, розглянуті в цьому розділі.

API	Код інструкції		Пульс інструкція	функція
	16-бітний	32-розрядний		
0400	CJ	–	✓	Умовний стрибок
0401	JMP	–	–	Безумовний стрибок
0402	GOEND	–	–	Стрибок до кінця програми

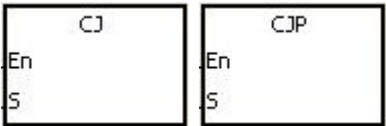
6.5.2 Пояснення інструкцій зі стрибка

API	Код інструкції			Операнд								функція					
0400		CJ	P	S								Умовний стрибок					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S																	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



S : Пункт призначення стрибка

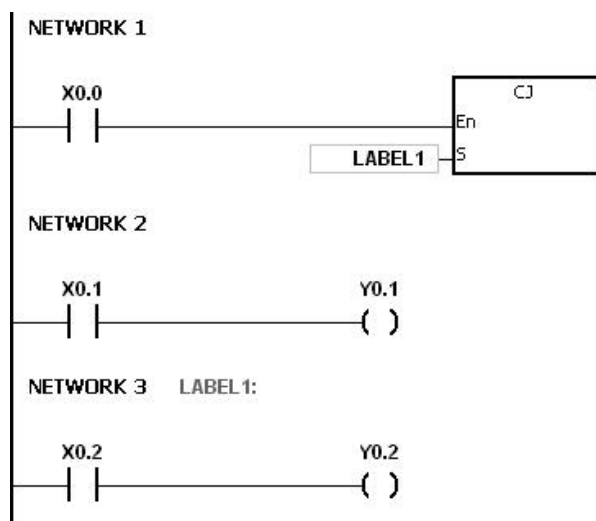
Пояснення

- Ця інструкція переходить від поточного виконання програми до мітки (пункту призначення) в іншій частині програми в ПЛК. Ви вказуєте мітку (вказівник) у **S** . Ви можете використовувати інструкцію CJ або CJP, щоб скоротити час сканування. Ви також можете використовувати інструкцію CJ або CJP, коли використовуєте подвійний вихід.
- Якщо програма, визначена призначенням переходу (міткою), передуватиме інструкції CJ, помилка сторожового таймера відбувається, і ПЛК припиняє виконання програми. Уважно користуйтеся цією інструкцією.
- Ви можете вказати ту саму мітку кілька разів за допомогою кількох різних інструкцій CJ.
- Коли інструкція виконується, дії пристроїв описані нижче.
 - Стан Y, стан M і стан **S** залишаються такими ж, як і до виконання стрибка.
 - Таймер продовжує відлік, і коли він досягає встановленого значення часу, програма запускає вихідну Т-котушку.
 - Для отримання додаткової інформації щодо інструкцій MC та MCR зверніться до прикладу 2 нижче.
 - Загальні прикладні інструкції не виконуються.

Приклад 1

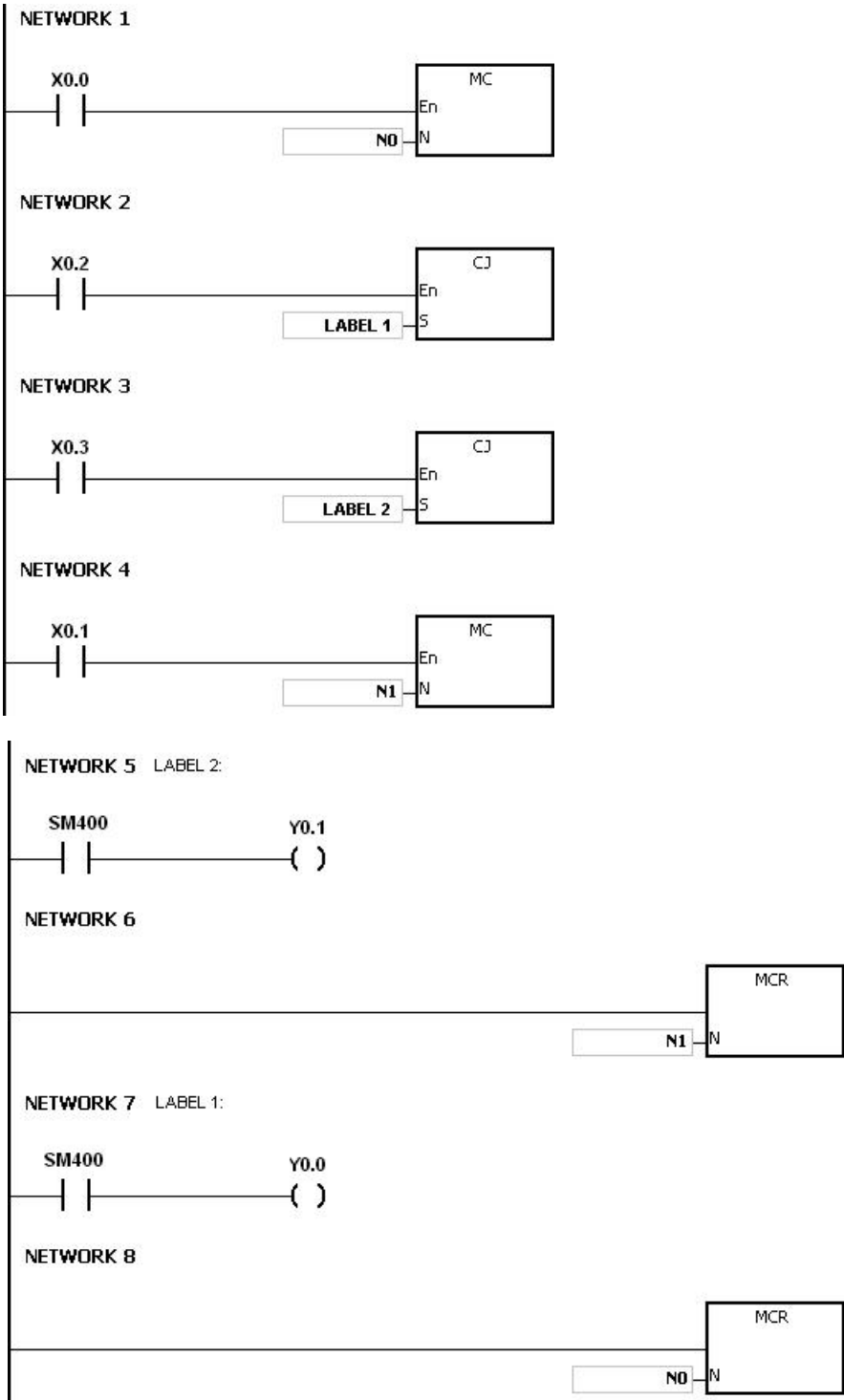
- Коли X0.0 увімкнено, виконання програми переходить від МЕРЕЖІ 1 до МІТКИ1 (МЕРЕЖА 3) і пропускає МЕРЕЖУ 2.

2. Коли X0.0 вимкнено, виконання програми послідовно переходить від МЕРЕЖІ 1 до МЕРЕЖІ 2 до МЕРЕЖІ 3, а інструкція CJ не виконується.



Приклад 2

1. Ви можете використовувати інструкцію CJ між інструкціями MC та MCR у п'яти наведених нижче умовах.
 - (a) Виконання програми переходить від частини програми за межами одного циклу MC/MCR до частини програми за межами іншого циклу MC/MCR.
 - (b) Виконання програми переходить від частини програми за межами циклу MC/MCR до частини програма всередині циклу MC/MCR.
 - (c) Виконання програми переходить від частини програми всередині циклу MC/MCR до частини програма всередині циклу MC/MCR.
 - (d) Виконання програми переходить від частини програми всередині циклу MC/MCR до частини програми за межами циклу MC/MCR.
 - (e) Виконання програми переходить від частини програми всередині одного циклу MC/MCR до частини програми всередині іншого циклу MC/MCR.
2. Коли ПЛК виконує інструкцію MC, він поміщає попередній стан контакту перемикача на вершину стека всередині ПЛК. Стек контролюється ПЛК і не може бути змінений. Коли ПЛК виконує інструкцію MCR, ПЛК знімає попередній стан контакту перемикача з вершини стека. За умов, перелічених у пунктах (b), (d) і (e) вище, кількість разів, коли елементи виштовхуються на стек, може відрізнятися від кількості разів, коли елементи викидаються зі стосу. Коли виникає така ситуація, до стека можна помістити не більше 32 елементів; елементи можна витягувати зі стека, доки стек не буде порожнім. Тому, коли ви використовуєте CJ або CJP з MC і MCR, будьте обережні з тим, як програма надсилає елементи на стек і викидає елементи зі стеку.

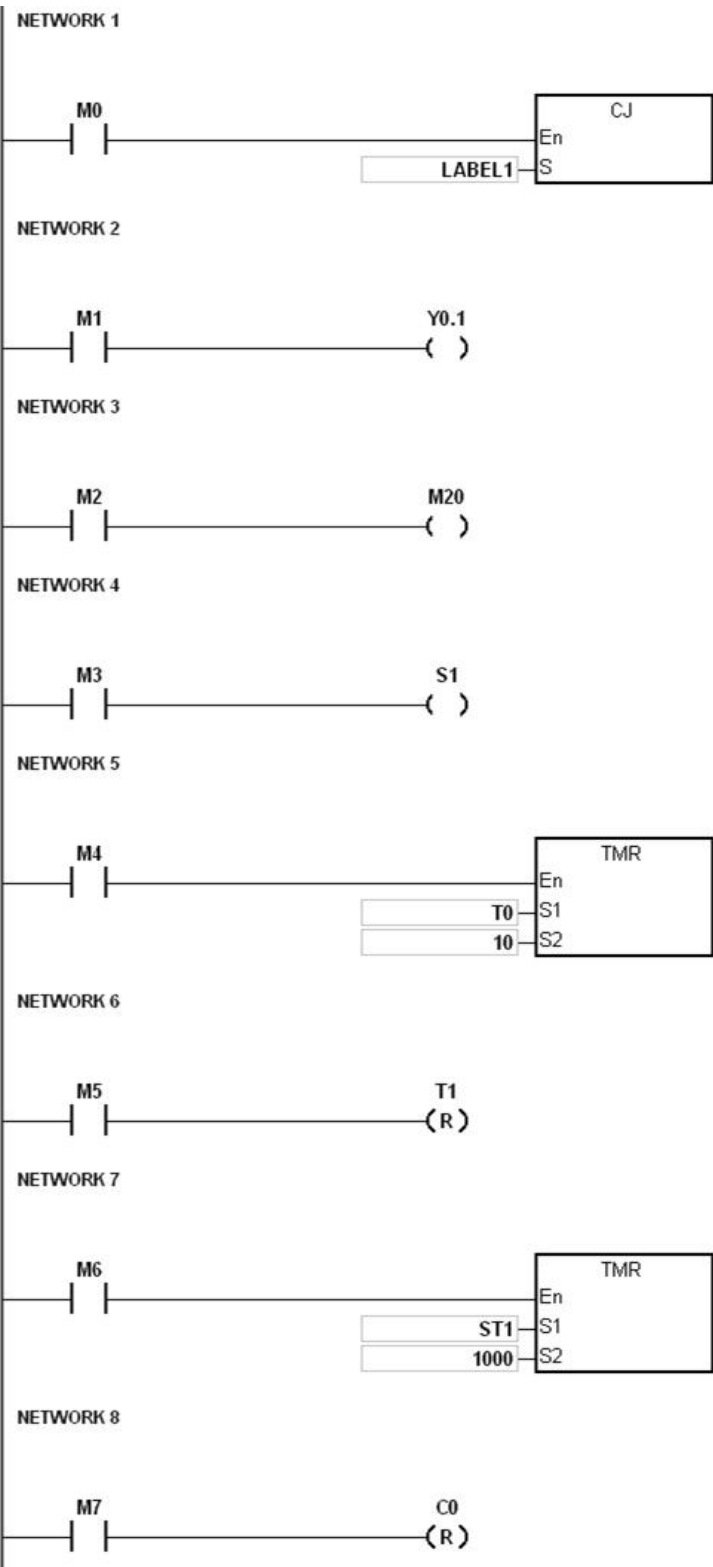


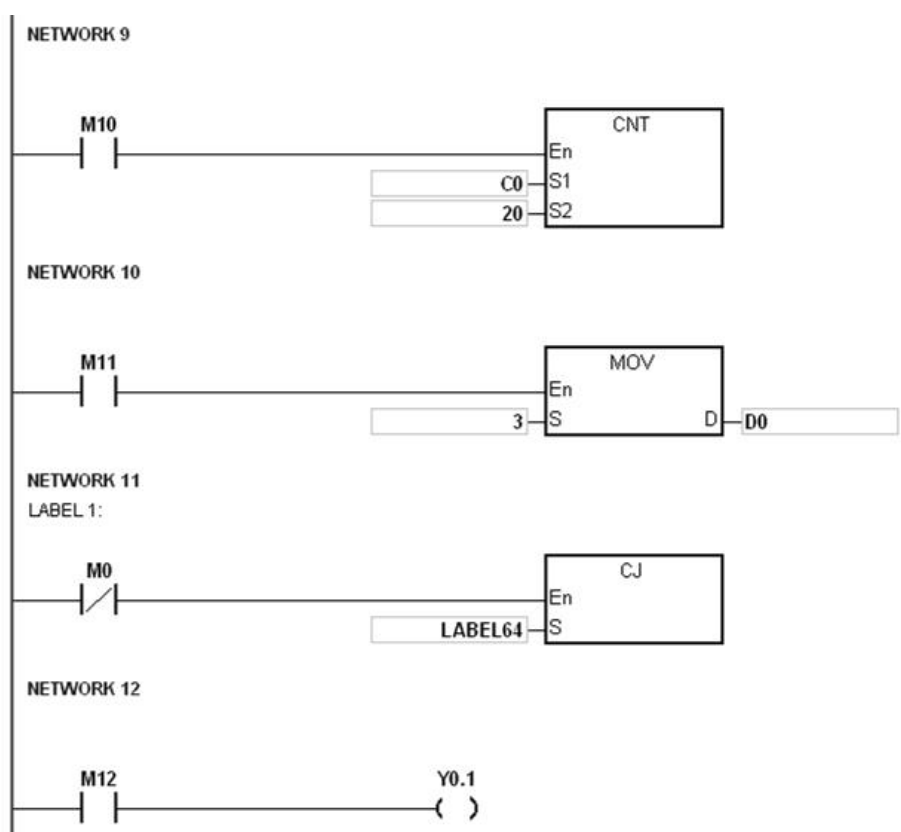
Приклад 3

Нижче наведено стани пристроїв.

пристрій	Стан контакту перед виконанням CJ M0=OFF	Стан контакту під час виконання CJ M0=ON	Стан вихідної котушки під час виконання CJ M0=ON
Y, M і S	M1, M2 і M3 є OFF.	M1, M2 і M3 перемикаються з OFF на ON.	Y0.1 *1, M20 і S1 вимкнено.
	M1, M2 і M3 увімкнено.	M1, M2 і M3 перемикаються з ON на OFF.	Y0.1 *1, M20 і S1 увімкнено.
Таймер	M4 вимкнено.	M4 перемикається з OFF. на ON.	Таймер не ввімкнено.
	M4 увімкнено.	M4 перемикається з ON. на OFF	Таймер продовжує відлік і час досягнуто встановленого значення таймера, він приводить в дію вихідну T- котушку.
Накопичувальний таймер	M6 вимкнено.	M6 перемикається з OFF. на ON.	ST1 не ввімкнено.
	M6 увімкнено.	M6 перемикається з ON. на OFF.	Тримає накопичувальний таймер підрахунок і коли значення налаштування таймера досягнуто, він приводить в дію вихідну T-котушку.
Лічильник	M7 і M10 є OFF.	M10 увімкнено/вимкнено.	Лічильник не включений.
	M7 вимкнено. M10 є ON/OFF.	M10 увімкнено/вимкнено.	C0 припиняє відлік. Коли M0 перемикається OFF, C0 продовжує відлік.
Прикладна інструкція	M11 вимкнено.	M11 перемикається з OFF. на ON	Прикладна інструкція - ні виконано.
	M11 увімкнено.	M11 перемикається з ON. на OFF	Застосовувана інструкція пропускається (не виконано).

*1:Y0.1 є подвійним виходом. Коли M0 вимкнено, Y0.1 керується M1. Коли M0 увімкнено, Y0.1 керується M12.





Додаткові зауваження

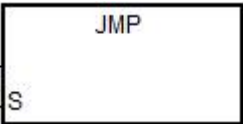
6

Зверніться до посібника користувача ISPSoft для отримання додаткової інформації про використання міток (вказівників) з інструкціями переходу.

API	Код інструкції				Операнд						функція						
0401			JMP		S						Безумовний стрибок						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S																	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													
					Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція		
					-			AS			-		

СИМВОЛ



S : Пункт призначення стрибка

Пояснення

- Ця інструкція змушує виконання програми переходити до частини програми, зазначеної міткою в S (вказівник) без жодної умови.
- Якщо програма, визначена міткою, передуватиме інструкції JMP, виникає помилка сторожового таймера, і ПЛК припиняє роботу програми. Уважно користуйтеся цією інструкцією.
- Зверніться до інструкції CJ (API 400) для отримання додаткової інформації про стани пристроїв під час виконання цієї інструкції.
- Зверніться до посібника користувача ISPSOFT для отримання додаткової інформації про використання міток (вказівників) з інструкціями переходу.

API	Код інструкції			Операнд	функція
0402		GOEND		—	Перехід до END

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



Пояснення

1. Ця інструкція змушує виконання програми переходити до END у програмі.
2. Функціональні блоки та завдання переривання не підтримують інструкцію GOEND. Ви не можете використовувати інструкцію між інструкціями FOR і інструкціями NEXT.
3. Коли ПЛК виконує інструкцію GOEND, пропущені інструкції не виконуються, дані в усіх пристроях не змінюються, і стани всіх пристроїв також не змінюються.

6.6 Інструкція по виконанню програми

6.6.1 Перелік інструкцій по виконанню програми

У наведеній нижче таблиці наведено інструкції щодо виконання програми, розглянуті в цьому розділі.

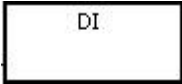
API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
0500	DI	–	–	Відключення функції переривання
0501	EI	–	–	Включення функції переривання
0503	EIX	–	–	Відключення конкретного переривання
0504	DIX	–	–	Включення певного переривання

6.6.2 Пояснення інструкцій щодо виконання програми

API	Код інструкції		Операнд	функція
0500		DI	-	Відключення функції переривання

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

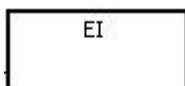
СИМВОЛ



Пояснення

Для отримання додаткової інформації зверніться до інструкції EI (API 0501).

API	Код інструкції			Операнд	функція
0501		EI		-	Включення функції переривання
				Інструкція пульсу	16-бітна інструкція
				-	AS
					32-розрядна інструкція
					-

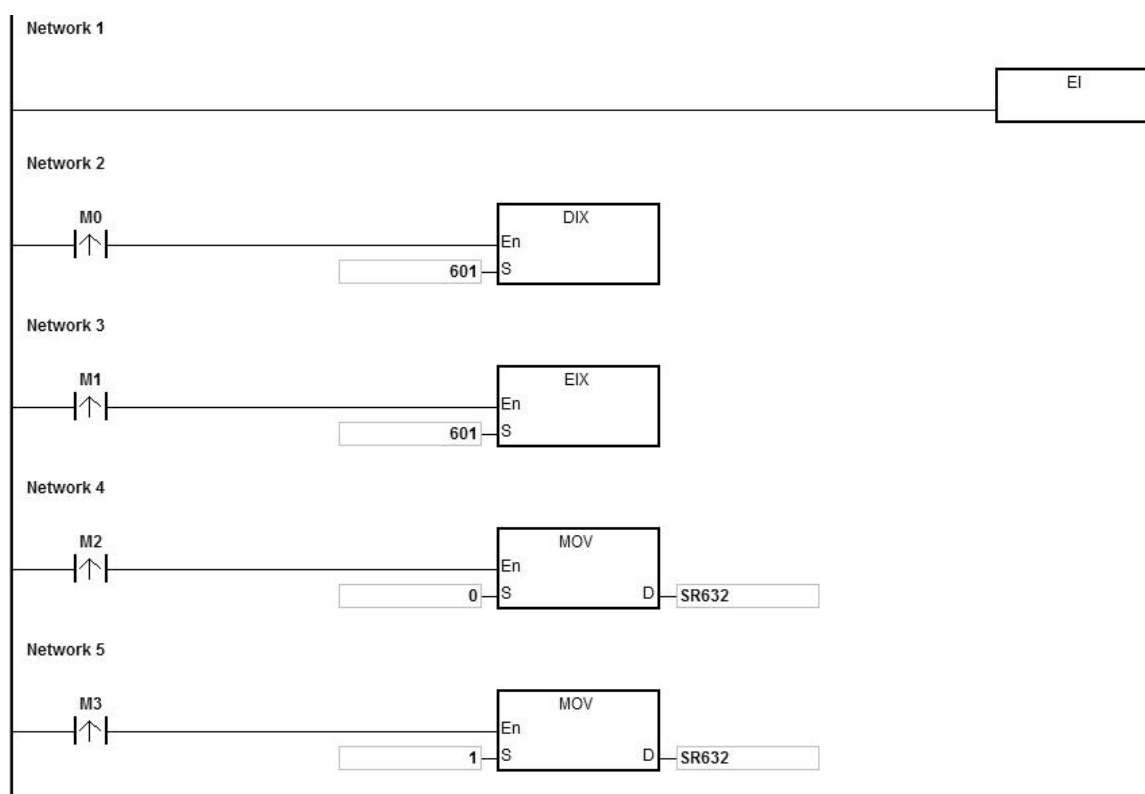
СИМВОЛ**Пояснення**

1. Використовуйте інструкцію EI, щоб увімкнути завдання переривання в програмі (додаткову інформацію про завдання див. на наступній сторінці).
2. Ви можете використовувати завдання переривання між інструкціями EI та інструкціями DI в програмі. Ви можете не використовувати інструкцію DI, якщо немає частини програми, в якій переривання вимкнено.
3. Під час виконання одного завдання переривання нове сформоване переривання не виконується, а зберігається. Після завершення виконання поточного завдання переривання виконується наступне завдання переривання. Наприклад, під час виконання I0 (за запуском порядком №1) генеруються 2 нових I0 (за запуском порядком №2, за запуском порядком №3), лише за запуском порядком №2 I0 буде збережено для подальшого виконання. ініційованим замовленням №3 I0 не зберігається для виконання.
4. При кількох перериваннях спочатку виконується завдання переривання з найвищим пріоритетом. Коли кілька переривань відбуваються одночасно, завдання переривання з найменшим номером показника виконується першим.
5. Коли між DI та EI виникає завдання переривання, воно не може бути виконане, а запит на переривання ігнорується. Це є запропоновувати не використовувати інструкцію DI для вимкнення переривань під час роботи ПЛК.
6. Якщо під час виконання завдання переривання потрібен негайний сигнал вводу/виводу, ви можете використовувати інструкцію REF або пристрій DX/DY у програмі, щоб оновити стан вводу/виводу.
7. Кожен номер переривання має тимчасову функцію, яку можна замаскувати. Нижче наведено список номерів переривань.

приклад:

- ◆ Налаштуйте завдання переривання I601 на 500 мс у HWCONFIG у ISPSoft.
- ◆ Коли ПЛК запускає програму Cyclic_0, він сканує інструкцію EI, вмикає завдання переривання, а потім виконує завдання переривання I601. Після завершення виконання завдання переривання виконується основна програма.
- ◆ Коли M0 увімкнено, завдання переривання таймера I601 вимкнено.
- ◆ Коли M1 увімкнено, завдання переривання таймера I601 увімкнено.
- ◆ Коли M2 увімкнено, SR632 дорівнює 0, а завдання переривання таймера I601 вимкнено.
- ◆ Коли M3 увімкнено, SR632 дорівнює 1, а завдання переривання таймера I601 увімкнено.

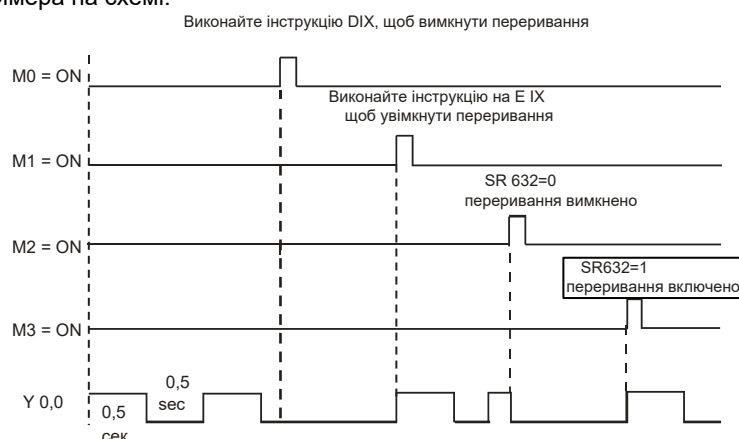
Програма Cyclic_0:



Завдання на переривання:



Переривання таймера на схемі:



Додаткові зауваження

Існує 7 типів завдань переривання:

1. Зовнішні переривання (I000–I115)

I000 вказує, що на вході X0.0 запускається спадаючий фронт. I100 вказує, що на вході X0.0 запускається наростаючий фронт. I001 вказує, що на вході X0.1 запускається спадаючий фронт.

I101 вказує, що на вході X0.1 запускається наростаючий фронт.

Решту можна зробити таким же чином.

2. Апаратні високошвидкісні переривання порівняння (I200–I253)

Цей тип переривань можна додатково розділити на 6 груп. Кожна група відповідає апаратному високошвидкісному лічильнику (для отримання додаткової інформації зверніться до API 1003 інструкції DCNT). Кожна група має 4 номери переривання (зверніться до API 1005 інструкції DHSCS для отримання додаткової інформації). Наприклад, номери переривань для першої групи I200-I203, а для другої групи I210-I213.

3. Програмні високошвидкісні переривання порівняння (I260–I267)

Є 8 переривань для програмного високошвидкісного порівняння, і ці 8 переривань спільно використовуються 8 високошвидкісними лічильниками.

4. Переривання зв'язку

Ви можете використовувати переривання зв'язку як інструкцію RS; тобто отримання певного символу викликає переривання, або ви можете використовувати його як загальне переривання. Для отримання додаткової інформації зверніться до інструкції COMRS (API1812).

COM1: I300

COM2: I302

Картка 1: I304

Картка 2: I306

5. Переривання модуля розширення (I400–I431)

Кожен модуль має одне переривання. Ви можете встановити 1 службу переривання для кожного модуля розширення.

6. Високошвидкісні вихідні переривання (I500–I519)

Після завершення виведення імпульсу надсилається запит на переривання. Переривання (I500–I505) для завершення виконання інструкції позиціонування працюють зі спеціальними пристроями (SM) для активації служби переривання. Наприклад, коли інструкція DDRVI завершує виконання першої осі, надсилається запит на переривання I500; ви можете встановити SM471 на ON, щоб активувати службу переривання. Переривання (I510–I519) для завершення виконання інструкції таблиці планування позиції працюють з інструкцією TPO. Після завершення виведення імпульсу надсилається запит на переривання.

7. Переривання таймера (I601–I604)

Встановіть переривання таймера, встановлені в HWCONFIG.

Для переривань таймера I601–I603: значення за замовчуванням становить 10 мілісекунд (одиниця виміру: 1 мс) (1–2000 мілісекунд).

Для переривань таймера I604: значення за замовчуванням становить 1 мілісекунда (одиниця виміру: 0,1 мс) (0,1–200 мілісекунд).

Повний список номерів переривань, описи та масковані переривання (SR) наведено в наступній таблиці.

Перервати номер	опис	Переривання з можливістю маскування SR	Номер біта
I000	Зовнішнє переривання: на вході X0.0 запускається спадаючий фронт.	SR623	0
I001	Зовнішнє переривання: на вході X0.1 запускається спадаючий фронт.		1
I002	Зовнішнє переривання: на вході X0.2 запускається спадаючий фронт.		2
I003	Зовнішнє переривання: на вході X0.3 запускається спадаючий фронт.		3
I004	Зовнішнє переривання: на вході X0.4 запускається спадаючий фронт.		4
I005	Зовнішнє переривання: на вході X0.5 запускається спадаючий фронт.		5
I006	Зовнішнє переривання: на вході X0.6 запускається спадаючий фронт.		6
I007	Зовнішнє переривання: на вході X0.7 запускається спадаючий фронт.		7
I008	Зовнішнє переривання: на вході X0.8 запускається спадаючий фронт.		8

Перервати номер	опис	Переривання з можливістю маскування SR	Номер біта
I009	Зовнішнє переривання: на вході X0.9 запускається спадаючий фронт.		9
I010	Зовнішнє переривання: на вході X0.10 запускається спадаючий фронт.		10
I011	Зовнішнє переривання: на вході X0.11 запускається спадаючий фронт.		11
I012	Зовнішнє переривання: на вході X0.12 запускається спадаючий фронт.		12
I013	Зовнішнє переривання: на вході X0.13 запускається спадаючий фронт.		13
I014	Зовнішнє переривання: на вході X0.14 запускається спадаючий фронт.		14
I015	Зовнішнє переривання: на вході X0.15 запускається спадаючий фронт.		15
I100	Зовнішнє переривання: вхід X0.0 запускається по наростаючому фронту.	SR624	0
I101	Зовнішнє переривання: вхід X0.1 запускається по передньому фронту.		1
I102	Зовнішнє переривання: вхід X0.2 запускається по наростаючому фронту.		2
I103	Зовнішнє переривання: вхід X0.3 запускається по наростаючому фронту.		3
I104	Зовнішнє переривання: вхід X0.4 запускається по наростаючому фронту.		4
I105	Зовнішнє переривання: вхід X0.5 запускається по наростаючому фронту.		5
I106	Зовнішнє переривання: вхід X0.6 запускається по наростаючому фронту.		6
I107	Зовнішнє переривання: вхід X0.7 запускається по наростаючому фронту.		7
I108	Зовнішнє переривання: вхід X0.8 запускається по наростаючому фронту.		8
I109	Зовнішнє переривання: вхід X0.9 запускається по наростаючому фронту.		9
I110	Зовнішнє переривання: вхід X0.10 запускається по наростаючому фронту.		10
I111	Зовнішнє переривання: вхід X0.11 запускається по наростаючому фронту.		11
I112	Зовнішнє переривання: вхід X0.12 запускається по наростаючому фронту.		12
I113	Зовнішнє переривання: вхід X0.13 запускається по наростаючому фронту.		13
I114	Зовнішнє переривання: вхід X0.14 запускається по наростаючому фронту.		14
I115	Зовнішнє переривання: вхід X0.15 запускається по наростаючому фронту.		15
I200	Переривання високошвидкісного порівняння 1 для апаратної високої швидкості лічильник 1	SR625	0
I201	Переривання високошвидкісного порівняння 2 для апаратної високої швидкості лічильник 1		1
I202	Переривання високошвидкісного порівняння 3 для апаратної високої швидкості лічильник 1		2
I203	Переривання високошвидкісного порівняння 4 для апаратної високої швидкості лічильник 1		3

Перервати номер	опис	Переривання з можливістю маскування SR	Номер біта
I210	Переривання високошвидкісного порівняння 1 для апаратної високої швидкості лічильник 2		4
I211	Переривання високошвидкісного порівняння 2 для апаратної високої швидкості лічильник 2		5
I212	Переривання високошвидкісного порівняння 3 для апаратної високої швидкості лічильник 2		6
I213	Переривання високошвидкісного порівняння 4 для апаратної високої швидкості лічильник 2		7
I220	Переривання високошвидкісного порівняння 1 для апаратної високої швидкості лічильник 3		8
I221	Переривання високошвидкісного порівняння 2 для апаратної високої швидкості лічильник 3		9
I222	Переривання високошвидкісного порівняння 3 для апаратної високої швидкості лічильник 3		10
I223	Переривання високошвидкісного порівняння 4 для апаратної високої швидкості лічильник 3		11
I230	Переривання високошвидкісного порівняння 1 для апаратної високої швидкості лічильник 4		12
I231	Переривання високошвидкісного порівняння 2 для апаратної високої швидкості лічильник 4		13
I232	Переривання високошвидкісного порівняння 3 для апаратної високої швидкості лічильник 4		14
I233	Переривання високошвидкісного порівняння 4 для апаратної високої швидкості лічильник 4		15
I240	Переривання високошвидкісного порівняння 1 для апаратної високої швидкості лічильник 5	SR626	0
I241	Переривання високошвидкісного порівняння 2 для апаратної високої швидкості лічильник 5		1
I242	Переривання високошвидкісного порівняння 3 для апаратної високої швидкості лічильник 5		2
I243	Переривання високошвидкісного порівняння 4 для апаратної високої швидкості лічильник 5		3

Перервати номер	опис	Переривання з можливістю маскування SR	Номер біта
I250	Переривання високошвидкісного порівняння 1 для апаратної високої швидкості лічильник 6		4
I251	Переривання високошвидкісного порівняння 2 для апаратної високої швидкості лічильник 6		5
I252	Переривання високошвидкісного порівняння 3 для апаратної високої швидкості лічильник 6		6
I253	Переривання високошвидкісного порівняння 4 для апаратної високої швидкості лічильник 6		7
I260	Переривання високошвидкісного порівняння 1 для програмного високошвидкісного лічильник	SR627	0
I261	Переривання високошвидкісного порівняння 2 для програмного високошвидкісного лічильник		1
I262	Переривання високошвидкісного порівняння 3 для програмного високошвидкісного лічильник		2
I263	Переривання високошвидкісного порівняння 4 для програмного високошвидкісного лічильник		3
I264	Переривання високошвидкісного порівняння 5 для програмного високошвидкісного лічильник		4
I265	Переривання високошвидкісного порівняння 6 для програмного високошвидкісного лічильник		5
I266	Переривання високошвидкісного порівняння 7 для програмного високошвидкісного лічильник		6
I267	Переривання високошвидкісного порівняння 8 для програмного високошвидкісного лічильник		7
I300	Отримання певного слова викликає переривання спілкування COM1	SR628	0
I301	Зарезервований		1
I302	Отримання певного слова викликає переривання спілкування COM1		2
I303	Зарезервований		3
I304	Отримання певного слова викликає переривання спілкування функціональна картка 1		4
I305	Зарезервований		5

Перервати номер	опис	Переривання з можливістю маскування SR	Номер біта
I306	Отримання певного слова викликає переривання спілкування функціональна картка 2		6
I307	Зарезервований		7
I500	Переривання високошвидкісного виведення: інструкція позиціонування 1-ї осі завершує	SR629	0
I501	Переривання високошвидкісного виведення: інструкція позиціонування 2-ї осі завершує		1
I502	Переривання високошвидкісного виведення: інструкція позиціонування 3-ї осі завершує		2
I503	Переривання високошвидкісного виведення: інструкція позиціонування 4-ї осі завершує		3
I504	Переривання високошвидкісного виведення: інструкція позиціонування 5-ї осі завершує		4
I505	Переривання високошвидкісного виведення: інструкція позиціонування 6-ї осі завершує		5
I510	Високошвидкісне вихідне переривання 1: таблиця планування позиції інструкція завершена	SR630	0
I511	Високошвидкісне вихідне переривання 2: таблиця планування позиції інструкція завершена		1
I512	Високошвидкісне вихідне переривання 3: таблиця планування позиції інструкція завершена		2
I513	Високошвидкісне вихідне переривання 4: таблиця планування позиції інструкція завершена		3
I514	Високошвидкісне вихідне переривання 5: таблиця планування позиції інструкція завершена		4
I515	Високошвидкісне вихідне переривання 6: таблиця планування позиції інструкція завершена		5
I516	Переривання високошвидкісного виведення 7: таблиця планування позиції інструкція завершена		6
I517	Високошвидкісне вихідне переривання 8: таблиця планування позиції інструкція завершена		7
I518	Високошвидкісне вихідне переривання 9: таблиця планування позиції інструкція завершена		8

Перервати номер	опис	Переривання з можливістю маскування SR	Номер біта
I519	Високошвидкісне вихідне переривання 10: таблиця планування позиції інструкція завершена		9
I601	Переривання таймера 1 (блок 1 мс)	SR632	0
I602	Переривання таймера 1 (блок 1 мс)		1
I603	Переривання таймера 1 (блок 1 мс)		2
I604	Переривання таймера 1 (одиниця 0,1 мс)		3

Примітка. Якщо кілька переривань виникають одночасно, першим виконується завдання переривання, номер показника якого є найменшим. ПЛК завершує поточне переривання, а потім виконує інші переривання відповідно до номерів показників. Наприклад, під час виконання переривання I400, якщо I500 та I300 відбуваються одночасно, ПЛК виконує переривання I300 (менший номер показника) після виконання переривання I400.

API	Код інструкції				Операнд					функція							
0503		EIX			S					Включення певного переривання							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S													O				
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S																	
					Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція						
					-			AS			-						

СИМВОЛ

	EIX
En	
S	

S : Джерело даних

Пояснення

- 6
- Джерело даних **S** може містити лише десяткове число, і це число має бути номером переривання. Якщо кількість не є номером переривання, інструкція не виконується і попередження не відображається. Наприклад, використовуйте EIX500 у **S** коли ви хочете увімкнути переривання I500. Зверніться до списку номерів переривань у поясненні інструкції EI.
 - За замовчуванням для завдань переривання в серії AS увімкнено. Якщо ви використовуєте інструкцію DIX для вимкнення переривань, ви повинні використовувати інструкцію EIX, щоб увімкнути переривання.
 - Ви можете використовувати цю інструкцію, щоб увімкнути завдання переривання в SR623–SR634.
 - Якщо ця інструкція не виконується, то вміст SR623–SR634 визначає, чи виконується завдання переривання чи ні.
 - Для отримання додаткової інформації зверніться до прикладів інструкції EI (API 0501).

API	Код інструкції				Операнд					функція							
0504		DIX			S					Відключення конкретного переривання							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S													O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S													

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

	DIX
En	
S	

S : Джерело даних

Пояснення

- Джерело даних **S** може містити лише десяткові числа, і число має бути номером переривання. Якщо число в **S** не є номером переривання, інструкція не виконується і попередження не відображається. Наприклад, використовуйте DIX500, коли ви хочете вимкнути переривання I500. Зверніться до списку номерів переривань у поясненні інструкції EI.
- За замовчуванням для завдань переривання в серії AS увімкнено. Використовуйте інструкцію DIX, щоб вимкнути переривання.
- Ви можете використовувати цю інструкцію, щоб вимкнути завдання переривання в SR623–SR634.
- Якщо ця інструкція не виконується, то вміст SR623–SR634 визначає, чи виконується завдання переривання чи ні.
- Для отримання додаткової інформації зверніться до прикладів інструкції EI (API 0501).

6.7 Інструкції з оновлення вводу/виводу

6.7.1 Список інструкцій щодо оновлення вводу/виводу

У наведеній нижче таблиці наведено інструкції з оновлення вводу/виводу, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32- розрядний		
0600	REF	–	✓	Оновлення введення-виведення
0601	–	DHSRF	✓	Негайне оновлення високошвидкісного порівняльного значення
0602	REFF	–	✓	Оновлення часу фільтрації введення/виведення

6.7.2 Пояснення інструкцій щодо оновлення вводу/виводу

API	Код інструкції				Операнд								функція			
0600		REF	P		D, n								Оновлення введення-виведення			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
D	O	O														
n								●	●		O	O	O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

REF	REFP
En	En
D	D
n	n

D : Стан точки введення/виведення для оновлення

n : Кількість станів точок введення/виведення для оновлення

Пояснення

- Стани введення/виведення зазвичай не оновлюються, доки ПЛК не виконає інструкцію END. Коли ПЛК запускається скануючи програму, вона зчитує та зберігає стани зовнішніх входів у пам'яті. Після виконання інструкції END ПЛК надсилає стани виходів у пам'яті на вихідні термінали. Таким чином, якщо вам потрібні останні дані введення/виведення під час процесу операції, ви можете скористатися цією інструкцією або скористатися пристроєм DX/DY для оновлення введення/виведення.
- Операнд **n** має бути кратним восьми, наприклад 8, 16, 24 і до 256. Якщо значення тут менше ніж кратне восьми, воно буде розглядатися як наступне кратне восьми. Наприклад, значення 20 буде розглядатися як його наступне кратне восьми, 24.
- Номер високошвидкісної вихідної точки зберігається в пристрої D для мікропрограми версії 1.04.00 і новішої. Якщо **n** дорівнює 1, це означає, що потрібно негайно оновити високошвидкісне вихідне значення відповідного SR. Якщо **n** дорівнює 0, це вказує на припинення високошвидкісного виведення та оновлення поточного значення SR. Наприклад, під час виконання цієї інструкції, якщо **n** дорівнює 0, а введення зовнішнього переривання отримано через X0.0, це означає, що зовнішнє переривання відбувається в X0.0, і високошвидкісний вихід через Y0.0 слід негайно припинити. ПЛК встановлює прапор зупинки SM463 на ON і оновлює поточну відповідну вихідну позицію в SR. Примітка: якщо для прапора автоматичного скидання завершення виведення встановлено значення ON,

ПЛК встановлює прапор автоматичної зупинки завершення виведення на OFF. і оновлює поточну відповідну позицію виводу в SR. Але ПЛК не встановлює прапор зупинки SM463 на ON.

Значення в n	D пристрій	Опис дій
n = кратне восьми	Y0.0 або X0.0	Негайно оновити введення-виведення Див. приклади 1 і 2
n = 1* 1	Високошвидкісна точка виведення	Оновити нове положення пульсу Див. приклад 3
n = 0* 1	Високошвидкісна точка виведення Без прапора автоматичного скидання завершення виведення	Зупинити високошвидкісне виведення, установити прапор зупинки SM463 на ON і оновити поточну відповідну позицію виведення. Дивіться приклад 3
n = 0* 1	Високошвидкісна точка виведення З прапором автоматичного скидання завершення виведення	Встановити для прапора автоматичного скидання завершення виведення значення OFF. і оновити поточну відповідну позицію виводу. Дивіться приклад 3.
n = -1* 2	Будь-які вхідні точки (X балів)	Оновити зіставлену область DS301 TxPDO (Master <= Slave)
n = -1* 2	Будь-які вихідні точки (Y точки)	Оновити зіставлену область DS301 RxPDO (Master => Slave)
n = -2* 3	Будь-які вхідні точки (X балів)	Відповісти на отриману команду зв'язку Modbus TCP (PLC діє як сервер)

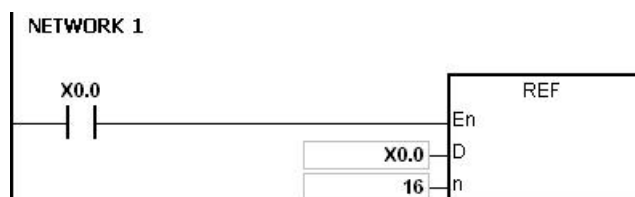
*1: Ця функція доступна лише для ЦП ПЛК із мікропрограмою версії 1.04.00 або новішої.

*2: Щоб ЦП ПЛК діяв як головний, його мікропрограмне забезпечення має бути версії 1.06.40 або новішої. Щоб ЦП ПЛК діяв як підлеглий, його мікропрограма має бути V1.12 або пізнішою. Ця функція не призначена для прискорення часу відображення даних PDO, і пропонується використовувати цю функцію, коли час сканування ПЛК довший за час оновлення PDO DS301.

*3: Ця функція доступна лише для ЦП PCL із мікропрограмою версії 1.10.30 або новішої. Це особливо корисно, коли час сканування ПЛК надто довгий. Ви можете використовувати цю функцію разом із тимчасовим перериванням, щоб відповісти на запит, надісланий через зв'язок Modbus TCP у встановлений час, щоб зв'язок між верхнім пристроєм і ЦП ПЛК почався швидше. Рекомендований час встановлення становить 5 мс. Якщо встановлений час для реакції ЦП ПЛК занадто короткий, час сканування ПЛК значно збільшиться. Оскільки відповідь на запит зв'язку Modbus TCP полягає в обробці запиту, це означає, що протягом часу сканування програми ПЛК буде додано більше завдань, включаючи читання або запис.

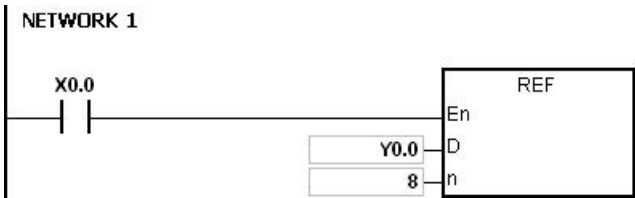
Приклад 1

1. Коли X0.0 увімкнено, ПЛК негайно зчитує стани входів X0.0 – X0.15 і оновлює вхідні сигнали без затримки.



Приклад 2

Коли X0.0 увімкнено, вихідні сигнали від Y0.0 до Y0.7 надсилаються на вихідні клеми. Вихідні сигнали оновлюються негайно, не чекаючи виконання інструкції END.



Додаткові зауваження

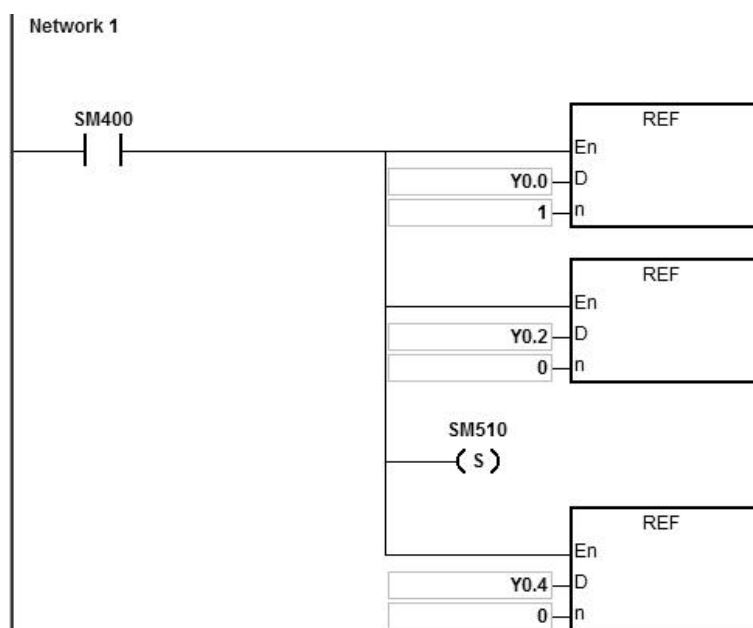
- 1. Якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

Приклад 3

- 1. Під час виконання цієї інструкції, якщо введення зовнішнього переривання отримано через X0.0

Значення в n	D пристрій	Опис дій
n = 1	Точка високошвидкісного виведення Y0.0	Негайно оновити нове положення пульсу Y0.0 (SR460)
n = 0	Високошвидкісна вихідна точка Y0.2 Без прапора автоматичного скидання завершення виведення	1. Зупинити високошвидкісне виведення 1. Встановить прапор зупинки SM483 на ON 2. Оновить поточну відповідну вихідну позицію SR480
n = 0	Високошвидкісна вихідна точка Y0.4 3 прапором автоматичного скидання завершення виведення SM510	1. Встановить для прапора автоматичного скидання завершення виведення SM510 значення OFF 2. Оновити поточну відповідну вихідну позицію. SR500

Програма зовнішнього переривання X0.0:



API	Інструкція			Операнд				опис								
0601	D	HSRF	P	S, S ₁				Негайне оновлення високошвидкісного порівняльного значення								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S							O									
S ₁								O								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S												●	
S ₁			●				●						

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	—	AS

СИМВОЛ

DHSRF	DHSRFP
En	En
S	S
S ₁	S ₁

S : Швидкісний лічильник для оновлення

S₁ : Кількість вводу/виводу для оновлення

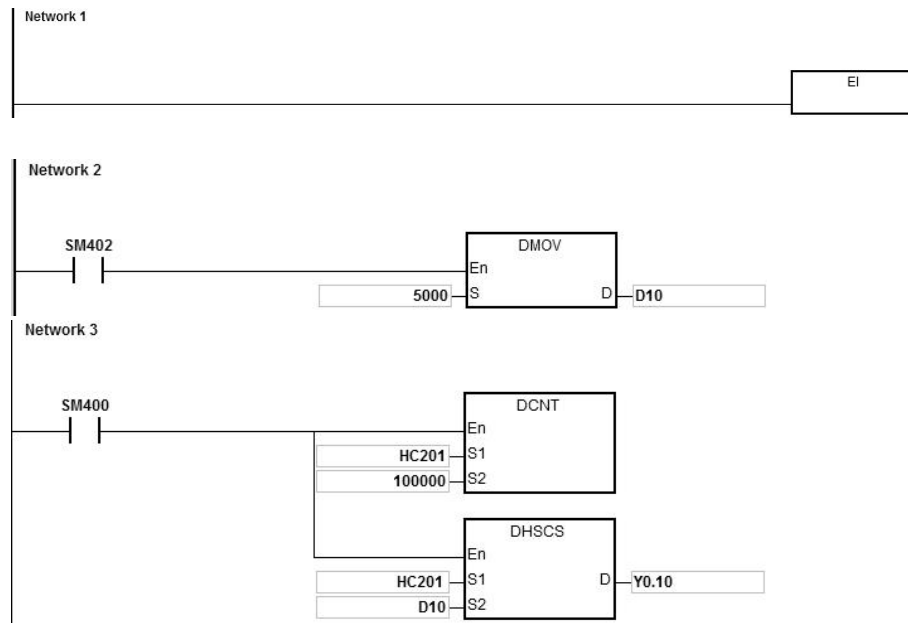
Пояснення

- Час, коли ПЛК оновлює порівняльне значення в своєму компараторі, коли DHSCS або DHSCR інструкція успішно просканована програмою. Однак оновлення може бути невдалим, якщо час сканування надто довгий або вхідний сигнал надходить надто швидко. У цьому випадку користувачі можуть використовувати інструкцію, щоб призначити нове порівняльне значення апаратному компаратору в ПЛК і виконати порівняння в реальному часі.
- S — номер зазначеного високошвидкісного лічильника, який потрібно оновити. S₁ має те саме ім'я компонента або змінної, що й що в інструкції DHSCS або DHSCR і негайне значення не можна встановити для S₁. Якщо S₁ не має того самого операнда, що й у ввімкненій інструкції високошвидкісного порівняння, виконання інструкції не матиме ефекту.
- Інструкцію підтримує лише мікропрограма версії 1.04.00 і пізніших.

приклад

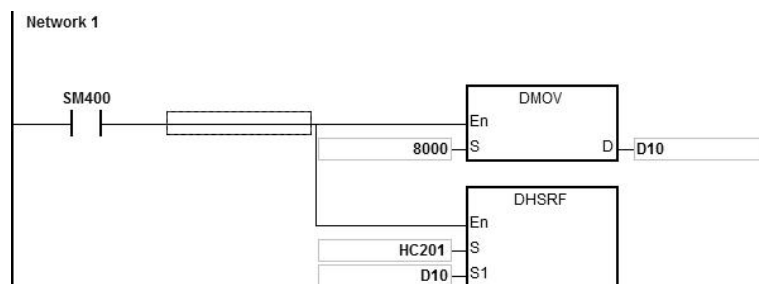
1. Коли ПЛК працює, порівняльне значення в інструкції DHSCS становить 5000.
2. Коли виникає зовнішнє переривання X0.0, порівняльне значення в DHSCS негайно встановлюється на 8000.

Основна програма:



Програма зовнішнього переривання X0.0:

Спочатку призначте нове порівняльне значення тій самій змінній (наприклад, D10 у прикладі), а потім виконайте інструкцію DHSRF для оновлення.



API	Код інструкції				Операнд								функція			
0602		REFF	P		Xno, Length, Filter								Оновлення часу фільтрації введення/виведення			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Xno	O															
Length					●	●		●	●				O	O		
Filter					●	●		●	●				O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Xno	●												
Length		●				●							
Filter		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

REFF	REFFP
En	En
Xno	Xno
Length	Length
Filter	Filter

- Xno** : Номер пускового пристрою X до оновити
- n** : Кількість станів точок введення/виведення для оновлення
- Filter** : Час фільтрації для оновлення (одиниця: мкс)

Пояснення

1. Ця інструкція працює лише з серією AS (FWV1.06.00) і ISPSOft V3.06 або новішими версіями.
2. **Xno** є початковим номером введення для оновлення. **Length** — це довжина, яку потрібно оновити. Якщо точка X є вбудованою вхідною точкою ЦП, коли **Xno** дорівнює X0,3, значення **Length** дорівнює 3, після виконання інструкції REFF час фільтрації вхідних даних X0,3 ~ X0,5 оновлюється.
- Оскільки вхідна точка X модуля розширення групується в 8 для одиниці, коли **Xno** дорівнює X1,9, а значення **Length** дорівнює 3, після виконання інструкції REFF час фільтрації вхідних даних X1,8 ~ X1,15 оновлюється. Коли **Xno** дорівнює 1,3, а значення **Length** дорівнює 3, після виконання інструкції REFF час фільтрації вхідних даних X1,0 ~ X1,7 оновлюється. Коли **Xno** дорівнює X1.3, а значення **Length** дорівнює 9, час фільтрації вхідних даних X1.0 ~ X1.15 оновлюється.
4. **Xno** і **Length** повинні бути встановлені на вході системи введення ПЛК. Якщо налаштування виходить за межі діапазону, навіть якщо ви виконали інструкцію REFF, I/O НЕ буде оновлено, і повідомлення про помилку НЕ буде показано.
5. Якщо точка X є вбудованою вхідною точкою ЦП, одиницею часу фільтрації є мкс. Тоді як, якщо точка X є вхідною точкою

модуль розширення, одиниця часу фільтрації мс. Ця інструкція призначена для оновлення поточного часу фільтрації; час фільтрації, встановлений у HWCONFIG, НЕ вплине. Але коли значення **фільтра** в інструкції перевищує діапазон налаштувань у HWCONFIG, ПЛК розглядає значення **фільтра** як максимальне або мінімальне значення, встановлене в HWCONFIG.

6. Під час увімкнення живлення час фільтрації ЦП і модуля розширення оновлюється відповідно до часу фільтрації, встановленого в HWCONFIG.

7. Інструкція REFF оновлює час фільтрації лише під час першого виконання.

6.8 Різні інструкції

6.8.1 Перелік різних інструкцій

У наведеній нижче таблиці наведено різні інструкції, описані в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16- розрядний	32- розрядний		
0700	ALT	–	✓	Чергування між ON і OFF
0701	TTMR	–	–	Таймер режиму навчання
0702	STMR	–	–	Спеціальний таймер
0703	RAMP	DRAMP	–	Сигнал циклічної рампи
0704	MTR	–	–	Матричний вхід
0705	ABSD	DABSD	–	Абсолютний барабанний секвенсор
0706	INCD	–	–	Інкрементний барабанний секвенсор
0708	–	DPIDE	–	ПІД-алгоритм
0709	XCMP	–	–	Налаштування для порівняння вхідних даних кількох робочих станцій
0710	YOUT	–	–	Порівняння результатів кількох робочих станцій
0711	–	DSUNRS	✓	Час сходу та заходу сонця

6.8.2 Пояснення різних інструкцій

API	Код інструкції				Операнд								функція				
0700		ALT	P		D								Чергування між ON і OFF				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D	●	●	●	●				●		O							

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												

Інструкція пульсу	16-розрядна інструкція	32-бітна інструкція
AS	AS	-

СИМВОЛ



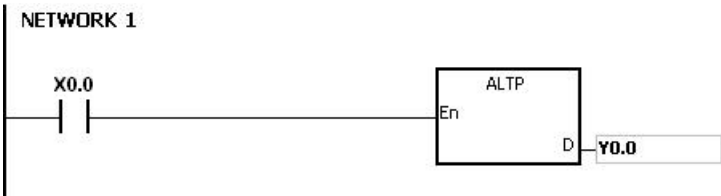
D : Пристрій призначення

Пояснення

- 1. Ця інструкція змінює стан пристрою, визначений **D** , між увімкненим і вимкненим.
- 2. Загалом використовуйте імпульсну інструкцію ALTP.

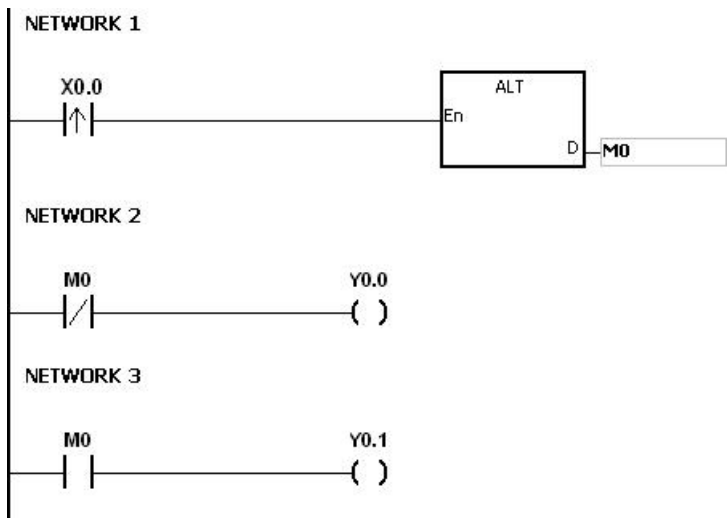
Приклад 1

Коли X0.0 перемикається з OFF на ON вперше, Y0.0 стає ON. Коли X0.0 перемикається з OFF на ON вдруге, Y0.0 стає OFF.



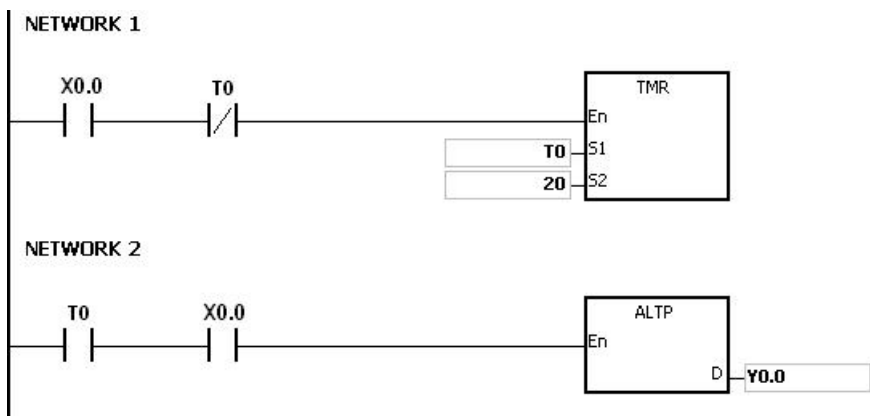
Приклад 2

На початку M0 вимкнено; тому Y0.0 увімкнено, а Y0.1 вимкнено. Коли X0.0 перемикається з OFF. на ON. вперше, M0 увімкнено; тому Y0.0 вимкнено, а Y0.1 увімкнено. Коли X0.0 перемикається з OFF. на ON. вдруге, M0 вимкнено; тому Y0.0 увімкнено, а Y0.1 вимкнено.



Приклад 3

Коли X0.0 увімкнено, T0 генерує імпульс кожні дві секунди. Вихід Y0.0 чергується між ON. і OFF до імпульсів, що генеруються T0.



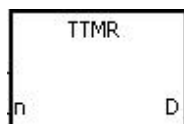
API	Код інструкції				Операнд						функція						
0701		TTMR			D, n						Таймер режиму навчання						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
D		●						●			○					
n	●	●						●	●		○		○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							
n		●				●							

Інструкція пульсу	16-розрядна інструкція	32-бітна інструкція
-	AS	-

СИМВОЛ



D :Записаний час

n :Множник

Пояснення

- Ця інструкція використовує секунди як одиницю часу. Час, протягом якого кнопковий перемикач був увімкнено, становить помножити на **n** , і добуток зберігається в **D** . **D** +1 лише для системного використання. Коли інструкція виконується, значення в **D** +1 не можна змінити. В іншому випадку час підраховується неправильно.
- Коли умовний контакт увімкнено, **D** скидається на 0.
- Встановлення множника: коли **n** дорівнює 0, **D** використовує секунду як одиницю часу. Коли **n** дорівнює 1, час, протягом якого кнопковий перемикач було увімкнено, множиться на 10, а **D** використовує 100 мілісекунд як одиницю часу. Коли **n** дорівнює 2, час, протягом якого кнопковий перемикач було увімкнено, множиться на 100, а **D** використовує 10 мілісекунд як одиницю часу. Чим більше значення в **n** , тим вище роздільна здатність синхронізації.

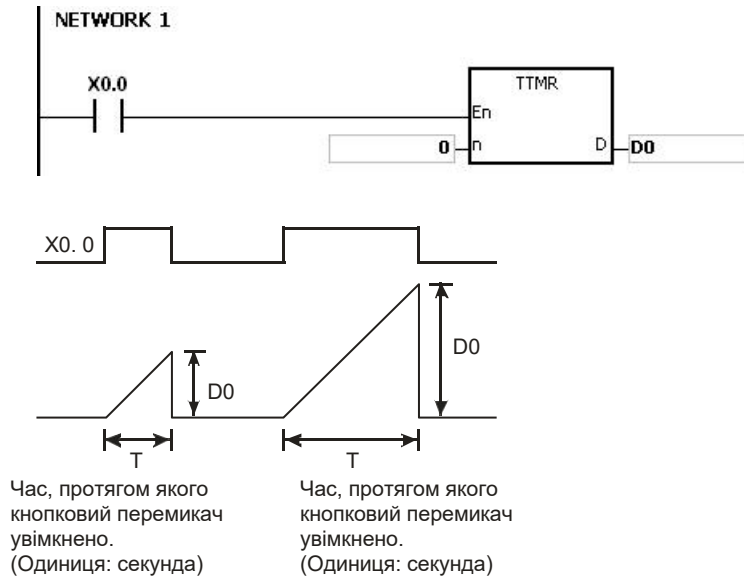
n	D
K0 (одиниця: 1 секунда)	1 × T
K1 (одиниця: 100 мілісекунд)	10 × T
K2 (одиниця: 10 мілісекунд)	100×T

- Якщо ви використовуєте редагування в режимі онлайн, скиньте умовний контакт, щоб ініціалізувати інструкцію.
- Значення в **n** має бути в діапазоні від 0 до 2.

Приклад 1

1. Інструкція множить час, протягом якого кнопковий перемикач X0.0 був увімкнений, на n , і зберігає продукт у D0. Для запису часу можна використовувати кнопковий перемикач (ON).

2. Коли X0.0 вимкнено, значення в D0 не змінюється.

**Додаткові зауваження**

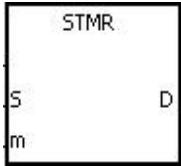
1. Якщо $D + 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16 № 2003.
2. Якщо значення в n не знаходиться в межах 0–2, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Якщо ви оголошуєте операнд **D** в ISPSOft, типом даних є ARRAY [2] із WORD/INT.

API	Код інструкції				Операнд						функція					
0702		STMR			S, m, D						Спеціальний таймер					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S					O											
m	●	●						●	●		○		○	○		
D		●	●	●				●		○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S											●		
m		●				●							
D	●												

Інструкція пульсу	16-розрядна інструкція	32-бітна інструкція
-	AS	-

СИМВОЛ



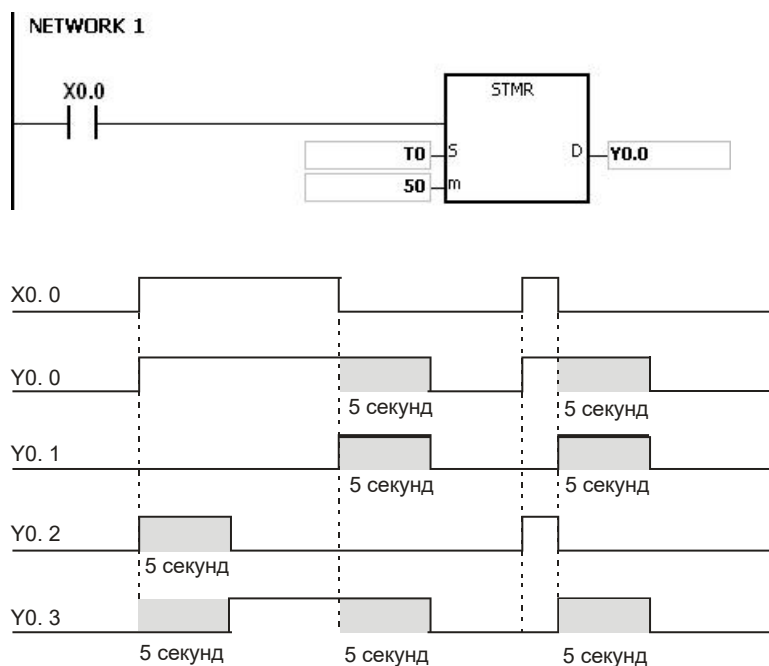
- S** : Номер таймера (T0-T511)
- m** : Установка значения таймера
- D** : Выводный пристрой

Пояснення

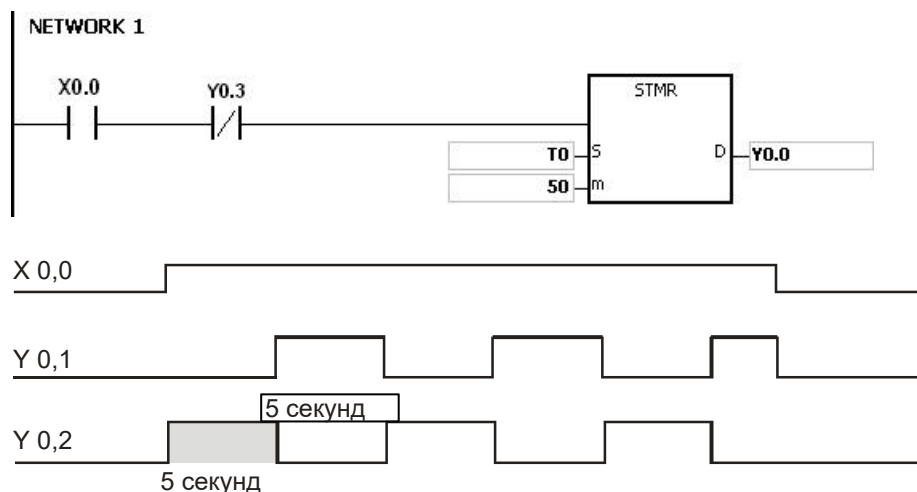
1. Ця інструкція генерує час для реле затримки вимкнення, одноразової схеми та схеми мигання.
2. Ця інструкція використовує 100 мілісекунд як одиницю часу. Якщо значення налаштування для **m** становить 50, значення часу дорівнює 5 секунд.
3. Ви не можете використовувати цей таймер повторно.
4. **D** займає чотири послідовні пристрої.
5. Перед виконанням інструкції скиньте **D** - **D** +3.
6. Коли умовний контакт не увімкнуто, а значення пристрою відповідає одній із двох умов, згаданих нижче, **D** , **D** +1 і **D** +3 увімкнені протягом **m** секунд, перш ніж їх вимкнути. Якщо умовний контакт не увімкнено, а значення пристрою не відповідає жодній із двох умов, згаданих нижче, **D** - **D** +3 залишаються ВИМКНЕНИМИ.
 - Значення таймера менше або дорівнює **m** , **D** увімкнено, а **D** +1 вимкнено.
 - Значення таймера менше ніж **m** , **D** +2 вимкнено, а **D** , **D** +1 і **D** +3 увімкнено.
7. Якщо використовується редагування в режимі онлайн, скиньте умовний контакт, щоб ініціалізувати інструкцію.
8. Значення в **m** має бути в діапазоні від 1 до 32767.

приклад

1. Коли X0.0 увімкнено, інструкція визначає таймер T0, а значення параметра T0 становить п'ять секунд.
2. Y0.0 – це контакт із затримкою вимкнення. Коли X0.0 перемикається на ON, Y0.0 стає ON. Через п'ять секунд після того, як X0.0 вимкнеться, Y0.0 вимкнеться.
3. Коли X0.0 вимикається, Y0.0 увімкнено протягом п'яти секунд.
4. Коли X0.0 вмикається, Y0.2 вмикається протягом п'яти секунд.
5. Через п'ять секунд після того, як X0.0 перемикається на ON, Y0.3 стає ON. Через п'ять секунд після того, як X0.0 вимкнеться, Y0.3 вимкнеться.



6. Коли за умовним контактом X0.0 слідує b-контакт Y0.3, ланцюг пробіску проходить через Y0.1 і Y0.2. Коли X0.0 вимикається, Y0.0, Y0.1 і Y0.3 вимикаються, а T0 скидається на 0.



Додаткові зауваження

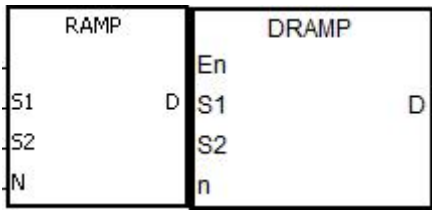
1. Якщо **D** +3 перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо значення в **m** менше 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
3. Якщо ви оголошуєте операнд **D** в ISPSOft, типом даних буде ARRAY [4] BOOL.

API	Код інструкції			Операнд						функція						
0703	D	RAMP		S ₁ , S ₂ , D, n						Сигнал циклічної рампи						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●					●	●	●		○	○	○	○		
S ₂	●	●					●	●	●		○	○	○	○		
D		●					●	●								
n	●	●					●	●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●						
S ₂		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу		16-розрядна інструкція	32-бітна інструкція
-		AS	AS

СИМВОЛ



- S_1 : Початкове значення сигналу рампи
- S_2 : Кінцеве значення сигналу рампи
- D : Тривалість сигналу рампи
- n : Кількість циклів сканування

Пояснення

- Ця інструкція отримує лінійний нахил, який має абсолютне співвідношення з часом сканування. Тому пропонується встановити фіксований час сканування або записати цю інструкцію в завдання переривання таймера.
- засдалегідь запише початкове значення та кінцеве значення сигналу рампи в S_1 та S_2 відповідно. Коли X0.0 є ON, D збільшується від значення налаштування в S_1 до значення налаштування в S_2 . Кількість циклів сканування зберігається в D +1. Коли значення в D дорівнює значенню в S_2 або коли значення в D +1 дорівнює n (числу циклів сканування), тоді SM687 увімкнено.
- Коли умовний контакт не увімкнуто, значення в D і D +1 дорівнюють 0, а SM687 вимкнено.
- Під час редагування в режимі онлайн скиньте умовний контакт, щоб ініціалізувати інструкцію.
- Зверніться до посібника користувача ISPSOFT, щоб дізнатися більше про встановлення фіксованого часу сканування.
- Значення n має бути в межах 1–32767. Якщо n виходить за межі діапазону, ця інструкція не виконується.

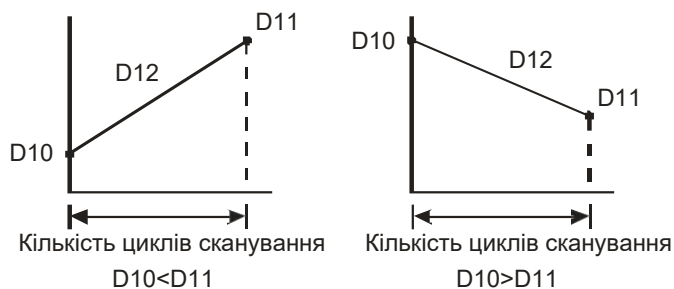
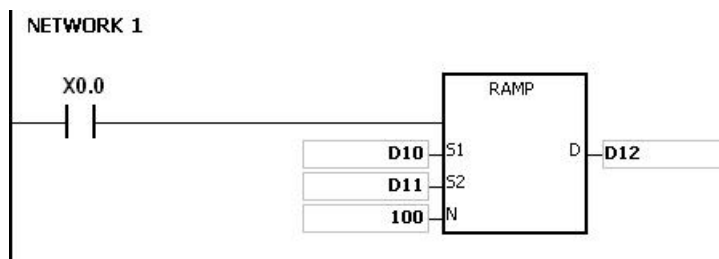
7. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій **E**.

8. Використовуйте прапор SM686, щоб скинути значення в **D** до 0. Зверніться до прикладів нижче, щоб отримати докладнішу інформацію.

приклад

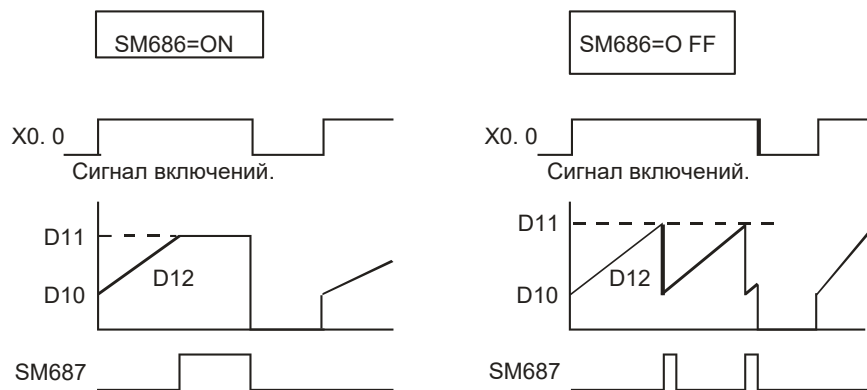
Коли ви використовуєте інструкцію з виходом аналогового сигналу, вона пом'якшує запуск і зупинку механізму.

1. Під час виконання, коли X0.0 вимикається, виконання інструкції припиняється. Коли X0.0 знову ввімкнеться, SM687 вимкнеться, D12 скинеться до значення налаштування в D10, D13 скинеться до 0, і розрахунок рампі розпочнеться заново.
2. Під час виконання SM686 OFF, і коли D12 досягає значення налаштування в D11, SM687 перемикається на ON як цикл сканування. Коли D12 скидається до значення налаштування в D10, D13 скидається до 0.



Кількість циклів сканування, що зберігається в D13.

3. Коли SM686 увімкнено, а D12 досягає значення налаштування в D11, значення в D12 не скидається на 0, а SM687 увімкнено. Поки умовний контакт замкнутий (ON), значення в D12 скидається на 0, а SM687 вимкнено. Коли SM686 увімкнено або вимкнено, значення в D12 змінюється, як показано нижче.



Додаткові зауваження

1. Якщо **D** +1 перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо **n** менше 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
3. Для 16-розрядної інструкції, якщо ви оголошуєте операнд **D** в ISPSOft, тип даних буде ARRAY [2] з WORD/INT.
4. Для 32-розрядної інструкції, якщо ви оголошуєте операнд **D** в ISPSOft, тип даних буде ARRAY [2] із DWORD/DINT.

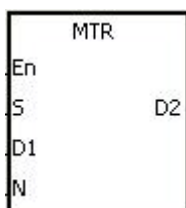
API	Код інструкції			Операнд							функція						
0704		MTR		S, D ₁ , D ₂ , n							Матричний вхід						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	○															
D ₁		○														
D ₂		○	○	○				○								
n	●	●						●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
D ₁	●												
D ₂	●												
n		●				●							

СИМВОЛ

Інструкція пульсу	16-розрядна інструкція	32-бітна інструкція
-	AS	-



- S** : Перший пристрій введення в матриці сканування
- D₁** : Перший пристрій виведення в матриці сканування
- D₂** : Перший відповідний пристрій у скануванні матриці
- n** : Кількість рядків для сканування

Пояснення

- Ця інструкція сканує та зберігає стани восьми послідовних пристроїв введення. **S** визначає перший пристрій введення в матричному скануванні.
- D₁** вказує транзисторний вихідний пристрій Y як перший пристрій у скануванні матриці. При умовному контакті є OFF, стани **n** пристроїв, починаючи з **D₁**, вимкнені.
- Один рядок вхідних даних оновлюється кожного циклу сканування. Є 16 входів посліпль, і сканування починається з першого ряду і переходить до **n**-го ряду.
- Вісім пристроїв введення, починаючи з пристрою, зазначеного **S**, підключені до **n** пристроїв виведення, починаючи з пристрою, зазначеного **D₁**, щоб утворити **n** рядів перемикачів. Матричне сканування зчитує стани **n** рядів перемикачів і зберігає стани в пристроях, починаючи з пристрою, визначеного **D₂**.
- Ви можете підключити до 8 рядів вхідних перемикачів паралельно, щоб отримати 64 входи (8×8=64).
- Інтервал між виконаннями цієї інструкції має бути довшим, ніж час, необхідний для оновлення станів точок введення/виведення на модулі. Інакше інструкція не зможе прочитати правильні стани входів. Побачити

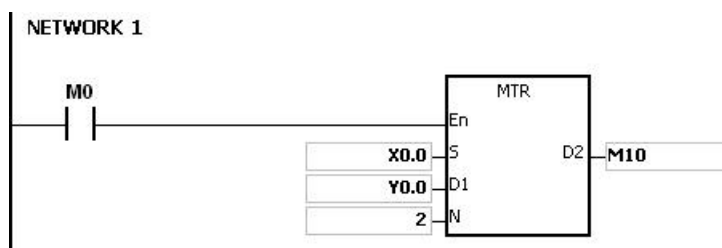
Додаткові зауваження нижче.

7. Загалом умовний контакт, який використовується в інструкції, — SM400: прапорець завжди ввімкнено, коли ЦП працює.

8. Значення в **n** має бути між 2–8.

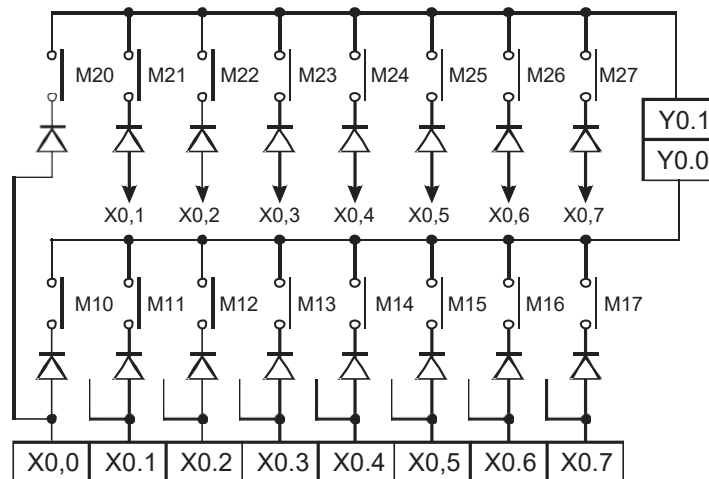
Приклад 1

1. Коли M0 увімкнено, виконується інструкція MTR. Інструкція зчитує стани двох рядів перемикачів по порядку та зберігає їх у внутрішніх реле M10–M17 і M20–M27 відповідно.



2. Наведена нижче схема зовнішнього з'єднання вхідної схеми матриці 2 на 8 для X0.0–X0.7 та Y0.0–Y0.1.

Відповідні внутрішні реле 16 перемикачів M10–M17 і M20–M27.

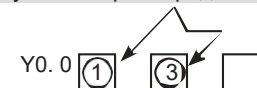


3. Інструкція підключає вісім пристроїв введення, починаючи з X0.0, до двох пристроїв виводу, починаючи з Y0.0, щоб сформувати

два ряди перемикачів. Матричне сканування зчитує стани двох рядів перемикачів і зберігає стани в пристроях, починаючи з M10 (вказано **D₂**). Тобто він зберігає стани першого ряду перемикачів у M10–M17 ;

зберігає стани другого ряду перемикачів у M20–M27.

Зчитується перший рядок вхідних сигналів.



Зчитується другий рядок вхідних сигналів.



Додаткові зауваження

1. Коли ця інструкція виконується, занадто довгий або занадто короткий час циклу призводить до неправильного зчитування стану перемикачів. Щоб вирішити цю проблему, скористайтеся наведеними нижче порадами.

- Якщо цикл сканування надто короткий, вхід/вихід може не відповісти вчасно, і правильні стани входів неможливо прочитати. Ви можете встановити фіксований час сканування, щоб вирішити цю проблему.
- Якщо цикл сканування надто довгий, комутатор може реагувати повільно. Ви можете записати цю інструкцію в завдання переривання таймера, щоб встановити фіксований час для виконання цієї інструкції.

2. Якщо $S + 7$, $D_1 + n - 1$ або $D_2 + (n * 8) - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

3. Якщо n не між 2–8, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

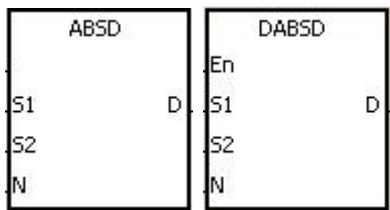
4. Якщо ви оголошуєте операнд S в ISPSOft, типом даних буде ARRAY [8] BOOL.

API	Код інструкції				Операнд							функція					
0705	D	ABSD			S ₁ , S ₂ , D, n							Абсолютний барабанний секвенсор					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●	●	●	●								
S ₂	●	●			●	●	●	●	●								
D		●	●	●				●									
n	●	●						●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●						
S ₂		●	●			●	●						
D	●												
n		●	●			●	●						

						Інструкція пульсу		16-розрядна інструкція		32-бітна інструкція	
						-		AS		AS	

СИМВОЛ



S_1 : Початковий пристрій для порівняння

S_2 : Значення порівняння

D : Результат порівняння

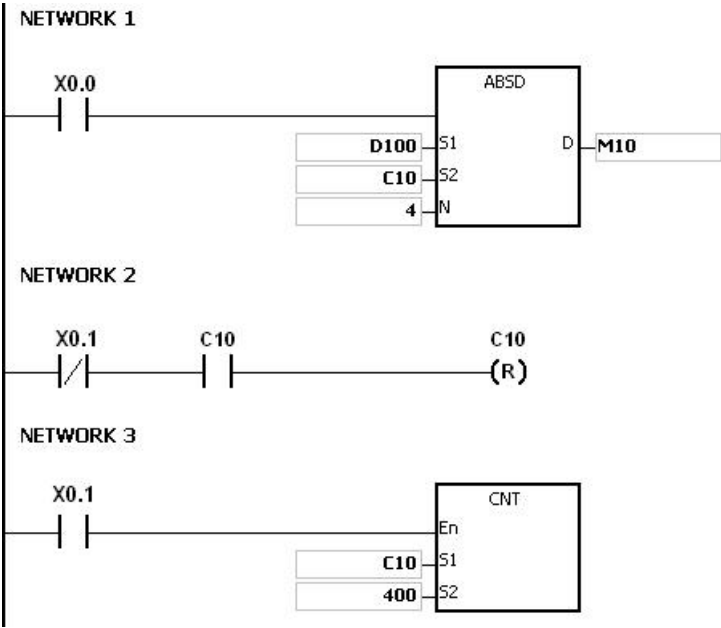
n : Кількість груп порівняння

Пояснення

- Використовуйте цю інструкцію для генерації кількох імпульсів, що відповідають поточним значенням лічильника.
- Тільки інструкція DABSD може використовувати 32-бітний лічильник, але не пристрій E.
- Якщо використовується інструкція ABSD, n має бути між 1–256.

Приклад 1

- Перед виконанням інструкції ABSD інструкція MOV записує значення параметрів у D100–D107. Значення в парних пристроях є мінімальними, а значення в непарних – максимальними.
- Коли X0.0 увімкнено, інструкція порівнює поточне значення лічильника C10 з максимальними значеннями та мінімальними значеннями в D100–D107 і зберігає результати порівняння в M10–M13.
- Коли X0.0 вимкнено, вихідні стани M10–M13 не змінюються.

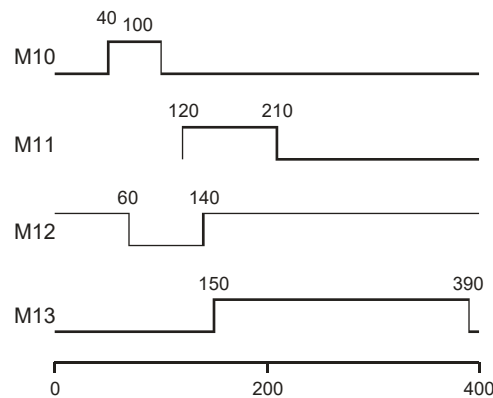


4. Коли поточне значення C10 знаходиться між мінімальним значенням і максимальним значенням, M10–M13 увімкнено.
В іншому випадку M10–M13 вимкнено.

Мінімальна вартість	Максимальне значення	Поточне значення C10	Вихід
D100=40	D101=100	$40 \leq C10 \leq 100$	M10=ON
D102=120	D103=210	$120 \leq C10 \leq 210$	M11=ON
D104=140	D105=170	$140 \leq C10 \leq 170$	M12=ON
D106=150	D107=390	$150 \leq C10 \leq 390$	M13=ON

5. Припустимо, мінімальне значення більше максимального. Коли поточне значення C10 менше, ніж максимальне значення ($C10 < 60$), або коли поточне значення C10 перевищує мінімальне значення ($C10 > 140$), M12 увімкнено. В іншому випадку M12 вимкнено.

Мінімальна вартість	Максимальне значення	Поточне значення C10	Вихід
D100=40	D101=100	$40 \leq C10 \leq 100$	M10=ON
D102=120	D103=210	$120 \leq C10 \leq 210$	M11=ON
D104=140	D105=60	$60 \leq C10 \leq 140$	M12=OFF
D106=150	D107=390	$150 \leq C10 \leq 390$	M13=ON



Додаткові зауваження

1. Для 16-розрядної інструкції, якщо $S + 2 * n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Для 32-розрядної інструкції, якщо $S + 4 * n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Для 16-розрядної інструкції, якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
4. Для 32-розрядної інструкції, якщо $D + 2 * n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
5. Як для 16-бітної, так і для 32-бітної інструкції, якщо n не знаходиться в межах 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

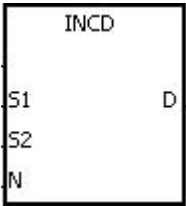
API	Код інструкції				Операнд					функція						
0706		INCD			S ₁ , S ₂ , n, D					Інкрементний барабанний секвенсор						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●							
S ₂	●	●			●	●		●	●							
D		●	●	●				●								
n	●	●						●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●				●							
S ₂		●				●							
D	●												
n		●				●							

Інструкція пульсу		16-розрядна інструкція		32-бітна інструкція	
-		AS		-	

СИМВОЛ

СИМВОЛ



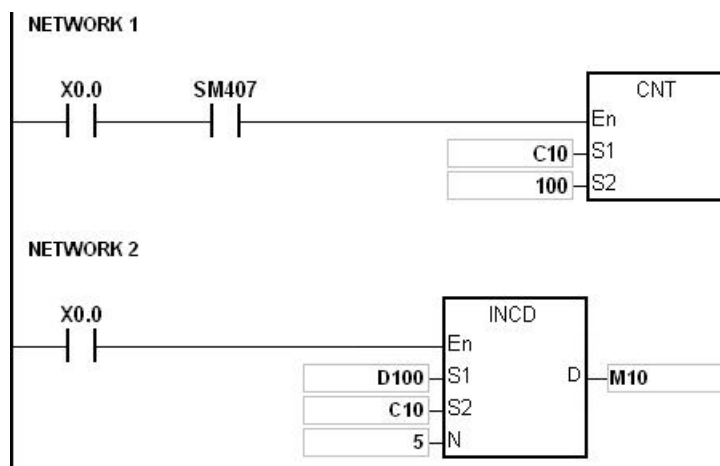
- S_1 : Початковий пристрій для порівняння
- S_2 : Номер лічильника
- D : Результат порівняння
- n : Кількість груп порівняння

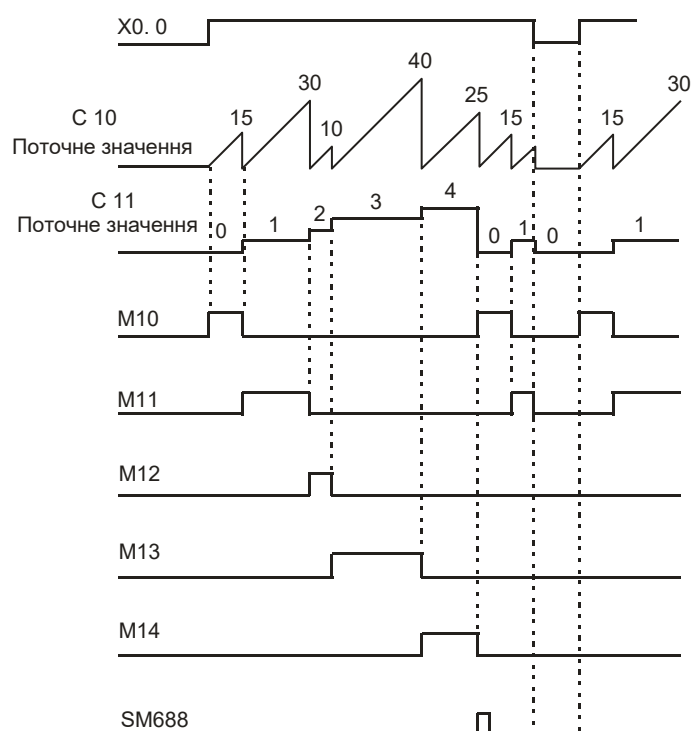
Пояснення

- Ця інструкція генерує декілька імпульсів для пари лічильників.
- Інструкція порівнює поточне значення S_2 із значенням налаштування в S_1 . Коли поточне значення збігається зі значенням налаштування, інструкція скидає поточне значення S_2 до 0 і зберігає поточний номер групи порівняння в $S_2 + 1$.
- Після завершення порівняння між поточними значеннями S_2 і n груп значень SM688 увімкнено для циклу сканування.
- Коли умовний контакт не увімкнено, значення в S_2 дорівнює 0, значення в $S_2 + 1$ дорівнює 0, D - D + n - 1 вимкнено, а SM688 вимкнено.
- Під час редагування в режимі онлайн скиньте умовний контакт, щоб ініціалізувати інструкцію.
- Значення в n має бути в діапазоні від 1 до 256.

приклад

1. Перед виконанням інструкції INCD інструкція MOV записує значення параметрів у D100 – D104. Значення в D100 – D104 становлять 15, 30, 10, 40 і 25 відповідно.
2. Інструкція порівнює поточні значення в C10 зі значеннями налаштувань у D100–D104. Коли поточне значення збігається зі значенням налаштування, інструкція скидає C10 на 0 і веде підрахунок знову.
3. Інструкція зберігає поточний номер групи порівняння в C11.
4. Коли значення в C11 змінюється на одиницю, M10 – M14 діють відповідно. Зверніться до наступної часової діаграми.
5. Після завершення порівняння між поточними значеннями в C10 і значеннями в D100 – D104 SM688 увімкнено для циклу сканування.
6. Коли X0.0 перемикається з ON на OFF, C10 і C11 скидаються на 0, а M10 – M14 вимикаються. Коли X0.0 знову перемикається в положення ON, виконання інструкції починається спочатку.





Додаткові зауваження

1. Якщо $S_2 + 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо $S_1 + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16 № 2003.
4. Якщо n не в діапазоні від 1 до 256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
5. Якщо ви оголошуєте операнд S_2 в ISPSOft, типом даних є ARRAY [2] із WORD/INT.

API	Код інструкції			Операнд									функція				
0708	D	PIDE		Як показано в наступній таблиці									ПІД-алгоритм				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
PID_RUN	●	●	●	●				●									
SV								●								○	
PV								●								○	
PID_MODE								●					○	○			
PID_MAN	●	●	●	●				●									
MOUT_AUTO	●	●	●	●				●									
CYCLE								●					○	○			
KC_Kp								●									
Ti_Ki								●									
Td_Kd								●									
Tf								●								○	
PID_EQ	●	●	●	●				●									
PID_DE	●	●	●	●				●									
PID_DIR	●	●	●	●				●									
ERR_DBW								●	●							○	
MV_MAX								●	●							○	
MV_MIN								●	●							○	
MOUT								●									
BIAS								●	●							○	
I_MV								●									
MV								●									

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING

Інструкція пульсу	16-розрядна інструкція	32-бітна інструкція
-	-	AS

СИМВОЛ

DPIDE
En
PID_RUN
SV
PV
PID_MODE
PID_MAN
MOUT_AUTO
CYCLE
Kc_Kp
Ti_Ki
Td_Kd
Tf
PID_EQ
PID_DE
PID_DIR
ERR_DBW
MV_MAX
MV_MIN
MOUT
BIAS
I_MV
MV

EN	: Увімкнути/вимкнути інструкцію
PID_RUN	: Увімкніть алгоритм PID
SV	: Цільове значення (SV)
PV	: Значення процесу (PV)
PID_MODE	: Режим ПІД-регулювання
PID_MAN	: Автоматичний/ручний режим PID
MOUT_AUTO	: Значення виведення вручну/авто
CYCLE	: Час вибірки (CYCLE)
Kc_Kp	: Пропорційне посилення
Ti_Ki	: Інтегральний коефіцієнт (сек або 1/сек)
Td_Kd	: Диференціальний коефіцієнт (с)
Tf	: Диференціальна постійна часу дії (с)
PID_EQ	: Типи формул PID
PID_DE	: Розрахунок похідної похибки ПІД
PID_DIR	: PID прямого/зворотного напрямку (PID_DIR)
ERR_DBW	: Діапазон, у якому знаходиться значення помилки : рахується як 0
MV_MAX	: Максимальне вихідне значення (MV_MAX)
MV_MIN	: Мінімальне вихідне значення (MV_MIN)
MOUT	: Ручне вихідне значення (MOUT)
BIAS	: Передача вихідного значення
I_MV	: Накопичене інтегральне значення
MV	: Вихідне значення (MV)

Пояснення

1. Ця інструкція реалізує алгоритм PID. Після досягнення часу вибірки інструкція застосовує алгоритм ПІД. PID означає пропорційний, інтегральний, диференціальний . ПІД-регулювання широко застосовується до механічного, пневматичного та електронного обладнання.
2. Налаштування параметрів наведено в наступній таблиці.

Операнд	Тип даних	функція	Діапазон налаштування	опис
PID_RUN	BOOL	Включення алгоритму PID	Правда: використовуйте алгоритм PID. False: скинути вихідне значення (MV) до 0 і припинити використання алгоритму PID.	
SV	REAL	SV	Діапазон чисел одинарної точності з плаваючою комою	Цільове значення
PV	REAL	PV	Діапазон чисел одинарної точності з плаваючою комою	Значення процесу
PID_MODE	DWORD/DINT	Режим ПІД-регулювання	0: Автоматичне керування Коли PID_MAN перемикається з True на False, викликає вихідне значення (MV) в автоматичному алгоритмі. Якщо MV перевищує межі MV_MAX або MV_MIN, значення I_MV залишається незмінним і не оновлюється. 1: Автоматичне налаштування параметрів для контролю температури. Після завершення налаштування система перебуває в режимі автоматичного керування (PID_MODE встановлено на 0) і заповнить відповідні параметри (Kc_Kp, Ti_Ki, Td_Kd і Tf) 2: Автоматичне керування (включаючи I_MV): коли PID_MAN перемикається з TRUE на FALSE, значення MV викликає вихідне значення (MV) у	

Операнд	Тип даних	функція	Діапазон налаштування	опис
			автоматичний алгоритм. Якщо MB перевищує межі MV_MAX або MV_MIN, I_MV скористайтесь формулою для розрахунку правильного значення та відповідно відкоригуйте його значення. Це може зменшити час відгуку зворотної дії. Примітка: коли режим встановлено на 1, автоматичне налаштування параметра, ви не можете використовувати числове значення для налаштування.	
PID_MAN	BOOL	Режим PID A/M	Правда: вручну Виведіть MV відповідно до MOUT, але воно все ще знаходиться між MV_MIN і MV_MAX. Цей параметр не діє, якщо PID_MODE встановлено на 1. False: Автоматично Виведіть MV відповідно до алгоритму PID, і вихідне значення буде між MV_MIN і MV_MAX.	
MOUT_AUTO	BOOL	Режим автоматичної зміни MOUT	Правда: автоматично MOUT змінюється в залежності від MV. False: Нормальний MOUT не змінюється в залежності від MV.	
CYCLE	DWORD/DINT	Час вибірки (T_s)	1–40 000 (одиниця: мс)	Коли інструкція сканується, використовуйте алгоритм PID відповідно до часу вибірки та оновіть MV. ПЛК вимагає, щоб інструкція була виконана; він не запустить

Операнд	Тип даних	функція	Діапазон налаштування	опис
				час відбору проб автоматично. Якщо T_s менше 1, воно вважається 1. Якщо T_s більше 40 000, воно вважається 40 000. Під час використання інструкції PID у завданні інтервального переривання час вибірки такий самий, як і інтервал між завданнями переривання. Налаштування циклу вибірки циклу вибірки тут ігнорується.
Kс_Kр	REAL	Розрахований пропорційний коефіцієнт (Kс або Kр, відповідно до налаштувань у PID_EQ)	Діапазон позитивних одно-точність з плаваючою комою чисел	Розраховано пропорційний коефіцієнт (Kс або Kр) Якщо коефіцієнт P менше 0, Kс_Kр дорівнює 0. Незалежно, якщо Kс_Kр дорівнює 0, він не контролюється P.
Ti_Ki	REAL	Інтегральний коефіцієнт (Ti або Ki, відповідно до налаштувань у PID_EQ)	Діапазон позитивний сингл-точність з плаваючою комою числа (одиниці: Ti = c; Ki = 1/c)	Якщо розраховано коефіцієнт I менше 0, Ti_Ki дорівнює 0. Якщо Ti_Ki дорівнює 0, він не контролюється I.

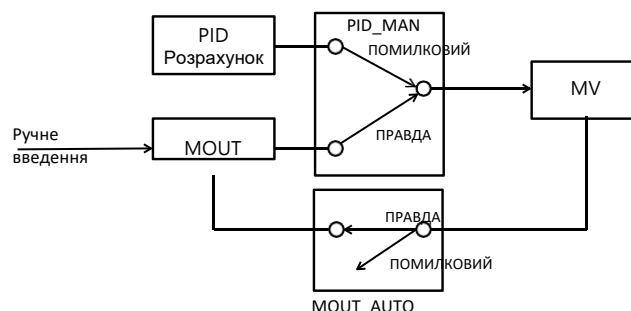
Операнд	Тип даних	функція	Діапазон налаштування	опис
Td_Kd	REAL	Коефіцієнт похідної (Td або Kd, відповідно до налаштувань у PID_EQ)	Діапазон позитивний сингл-точність з плаваючою комою числа (одиниця вимірювання: секунди)	Якщо розраховано коефіцієнт D менше 0, Td_Kd дорівнює 0. Якщо Ti_Ki дорівнює 0, він не контролюється D.
Tf	REAL	Похідна постійна часу дії	Діапазон позитивний сингл-точність з плаваючою комою числа (одиниця вимірювання: секунди)	Якщо похідна-дія постійна часу менше 0, Tf дорівнює 0 і не контролюється постійною часу дії похідної (згладжування похідної).
PID_EQ	BOOL	Типи формул PID	ІСТИНА: залежна формула FALSE: незалежна формула	
PID_DE	BOOL	Розрахунок похідної похибки ПІД	ПРАВДА: використовуйте варіації в PV для обчислити контрольне значення похідної (Derivative of the PV). НЕПРАВИЛЬНО: використовуйте варіанти помилки (E) для обчислення контрольного значення похідної (похідної помилки).	
PID_DIR	BOOL	PID напрямом вперед/назад	Вірно: пряма дія ($E=SV-PV$) False: зворотна дія ($E=PV-SV$)	
ERR_DBW	REAL	Діапазон, у межах якого значення помилки вважається 0.	Діапазон одинарної точності плаваючий-номери точок	Значення похибки (E) становить різниця між SV і PV. Якщо значення параметра дорівнює 0, функція вимкнена; інакше ЦП

Операнд	Тип даних	функція	Діапазон налаштування	опис
				модульні перевірки чи поточна помилка менша за абсолютне значення ERR_DBW, і перевіряє, чи поточна помилка відповідає умові перехресного статусу. Якщо поточна помилка менша за абсолютне значення ERR_DBW, і відповідає умові перехресного статусу, поточна помилка вважається 0, і ПЛК застосовує алгоритм PID; інакше поточна помилка вводяться в алгоритм PID відповідно до звичайної обробки.
MV_MAX	REAL	Максимальне вихідне значення	Діапазон одинарної точності плаваючий- номери точок	Припустимо, MV_MAX встановлено на 1000. Коли MV більше 1000, 1000 – вихід. Значення в MV_MAX має бути більшим за дуюм MV_MIN . В іншому випадку максимальні MB і мінімальний MV змінюється.
MV_MIN	REAL	Мінімальне вихідне значення	Діапазон	Припустимо, MV_MIN дорівнює

Операнд	Тип даних	функція		Діапазон налаштування	опис
				одно-точність з плаваючою комою чисел	встановити -1000. Коли MV менше - 1000, -1000 вихід.
MOUT	REAL	MV		Діапазон одно-точність з плаваючою комою чисел	Якщо встановлено PID Manual, значення MV виводиться як значення налаштування для MOUNT між MV_MAX і MV_MIN.
BIAS	REAL	Передача вихідного значення		Діапазон одно-точність з плаваючою комою чисел	Вихідне значення прямої подачі, що використовується для прямої подачі ПІД.
I_MV (займає 15 послідовних DWord пристрої)	REAL	I_MV	Накопичене інтегральне значення	Діапазон одинарної точності плаваючий-номери точок	Накопичений інтеграл значення, яке тимчасово зберігається і зазвичай для довідки. Ви все ще можете очистити або змінити його відповідно до своїх потреб. Режим PID в 0 (автоматичне керування): коли MV більше, ніж MV_MAX, або коли MV менше, ніж MV_MIN, накопичене інтегральне значення в I_MV не змінюється. Режим PID в 2

Операнд	Тип даних	функція		Діапазон налаштування	опис
					(Автоматичне керування, включаючи I_MV): Коли PID_MAN перемикається з TRUE на FALSE, значення MV викликає вихідне значення (MV) в автоматичному алгоритмі. Якщо MV перевищує межі MV_MAX або MV_MIN, I_MV використовує формулу для обчислення правильного накопиченого інтегрального значення та відповідного коригування його значення. Це може зменшити час відгуку зворотної дії.
		I_MV+1	Тут тимчасово зберігається попереднє значення помилки.		
		I_MV+2– I_MV+5	Тільки для системного використання		
		I_MV+6	Тут тимчасово зберігається попередній PV.		
		I_MV+7– I_MV+14	Тільки для системного використання		
MV	REAL	MV	MV знаходиться між MV_MIN і MV_MAX.		

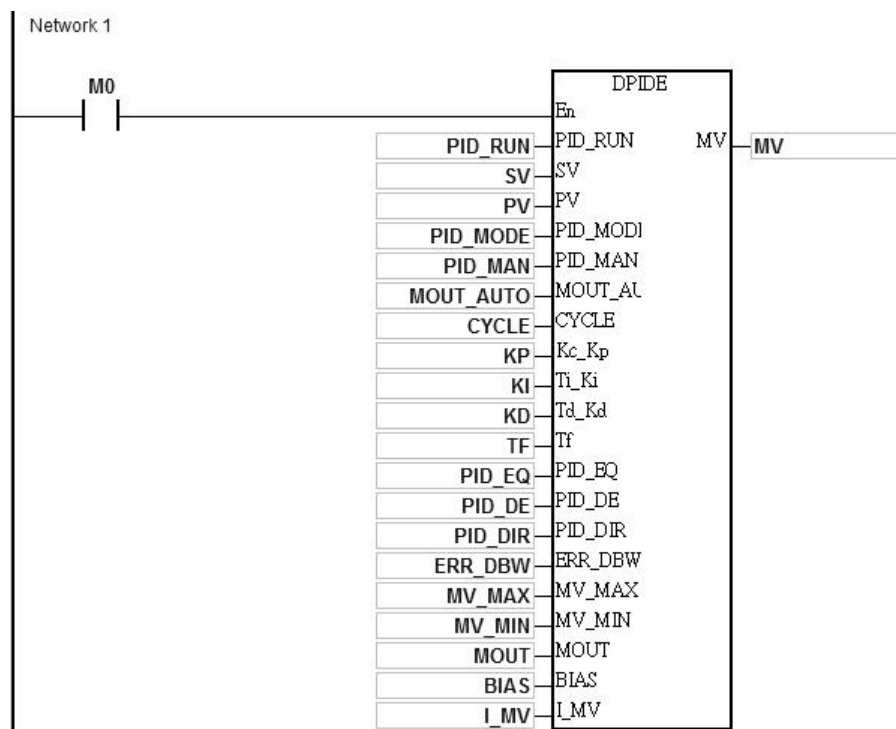
Схема перемикавання на PID_MAN / MOUT_AUTO:



1. Під час перемикавання режиму керування (**PID_MAN=0**) з автоматичного на ручний ви можете встановити прапор MOUT_AUTO на 1, а вихідне значення MOUT буде відповідати вихідному значенню MV. Після переходу в ручний режим (**PID_MAN=1**) ви можете встановити MOUT_AUTO на 0.
2. Коли **PID_RUN** змінюється з TRUE на FALSE, ПЛК скидає значення в MV до 0. Якщо значення в MV потрібно зберегти, ви можете встановити для EN значення FALSE, щоб відхилити інструкцію та зберегти вихідне значення в MV.

Приклад 1

1. Встановіть усі параметри перед виконанням цієї інструкції.
2. Коли M0 увімкнено, інструкція виконується. Коли PID_RUN має значення ON, інструкція застосовує алгоритм DPID. Коли PIC_RUN вимкнено, MV дорівнює 0 і зберігає значення в MV. Коли M0 перемикається у положення OFF, інструкція не виконується, а попередні дані залишаються незмінними.

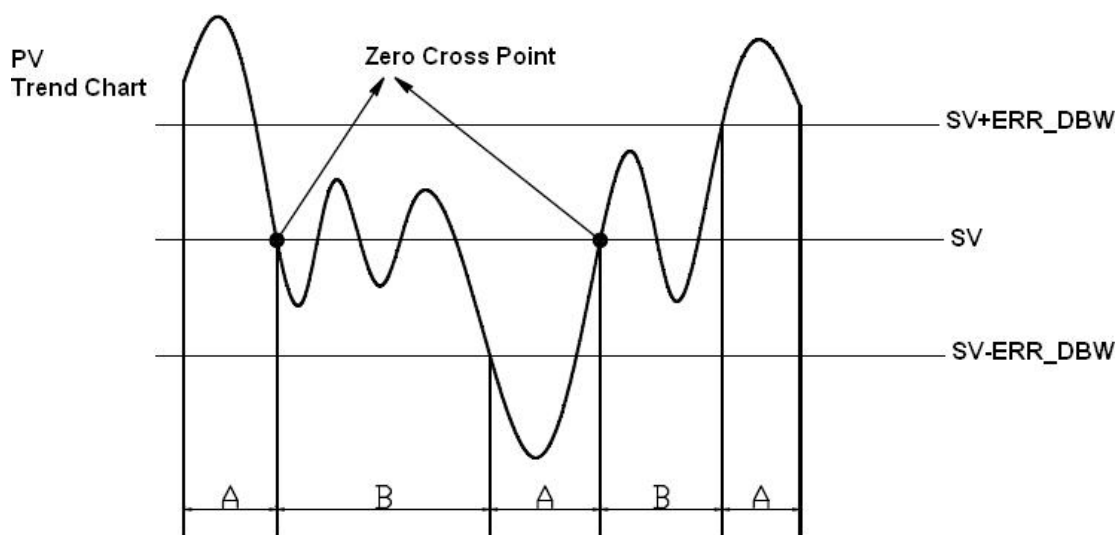


Додаткові зауваження

1. Інструкцію можна використовувати кілька разів, але регістри, визначені **I_MV – I_MV +14**, не можуть бути однаковими.
2. **I_MV** займає 30 регістрів.
3. Ви можете використовувати 32-розрядну інструкцію лише в циклічних завданнях і завданнях інтервального переривання. При використанні 32-розрядної інструкції в завданні інтервального переривання час дискретизації (цикл) такий самий, як і інтервал між завданнями переривання.
4. Коли інструкція сканується, 32-розрядний алгоритм PID застосовується відповідно до часу вибірки (цикл), і він оновлює MV. Коли ви використовуєте інструкцію в завданні переривання, час вибірки (цикл) такий самий, як і

інтервал між завданнями тимчасового переривання. Алгоритм PID застосовується відповідно до інтервалу між завданнями тимчасового переривання.

5. Перш ніж застосувати 32-розрядний алгоритм PID, значення процесу, що використовується в інструкції PID, має бути стабільним значенням. Якщо вам потрібне вхідне значення в модулі для реалізації алгоритму DPID, слід звернути увагу на час, потрібний для перетворення аналогового вхідного сигналу в цифровий.
6. Коли PV (значення процесу) знаходиться в діапазоні **ERR_DBW**, на початку поточна помилка вводиться в алгоритм PID відповідно до нормальної обробки, а потім модуль CPU перевіряє, чи поточна помилка відповідає умові перехресного статусу: PV (процесне значення) перевищує SV (цільове значення). Після виконання умови поточна помилка вважається 0 під час застосування алгоритму PID. Після того, як PV (значення процесу) виходить за межі діапазону **ERR_DBW**, поточна помилка знову вноситься в алгоритм PID. Якщо PID_DE має значення true, це означає, що він використовує варіації PV для обчислення контрольного значення похідної, і після виконання умови перехресного статусу ПЛК розглядає ΔPV як 0 для застосування алгоритму PID. (ΔPV = поточний **PV** – попередній **PV**). У наступному прикладі поточна помилка вводиться в алгоритм PID відповідно до звичайної обробки в розділі А, а поточна помилка або ΔPV вважається 0 для застосування алгоритму PID в розділі В.



Алгоритм PID:

1. Коли ви встановлюєте **PID_MODE** на 0, режим ПІД-регулювання є автоматичним.

- **Незалежна формула та похідна E** (**PID_EQ** =False & **PID_DE** =False)

$$MV = K_p E + Ki \int_0^t E dt + K_d * \frac{dE}{dt} + BIAS \quad E = SV - PV \text{ або } E = PV - SV$$

- **Незалежна формула та похідна PV** (**PID_EQ** =False & **PID_DE** =True)

$$MV = K_p E + Ki \int_0^t E dt - K_d * \frac{dPV}{dt} + BIAS \quad E = SV - PV$$

Або

$$MV = K_p E + Ki \int_0^t E dt + K_d * \frac{dPV}{dt} + BIAS \quad E = PV - SV$$

- **Залежна формула та похідна E** (**PID_EQ** = True & **PID_DE** = False)

$$MV = K_c \left[E + \frac{1}{T_i} \int_0^t E dt + T_d * \frac{dE}{dt} \right] + BIAS \quad E = SV - PV \text{ або } E = PV - SV$$

- **Залежна формула та похідна PV** (**PID_EQ** = True & **PID_DE** = True)

$$MV = K_c \left[E + \frac{1}{T_i} \int_0^t E dt - T_d * \frac{dE}{dt} \right] + BIAS \quad E = SV - PV$$

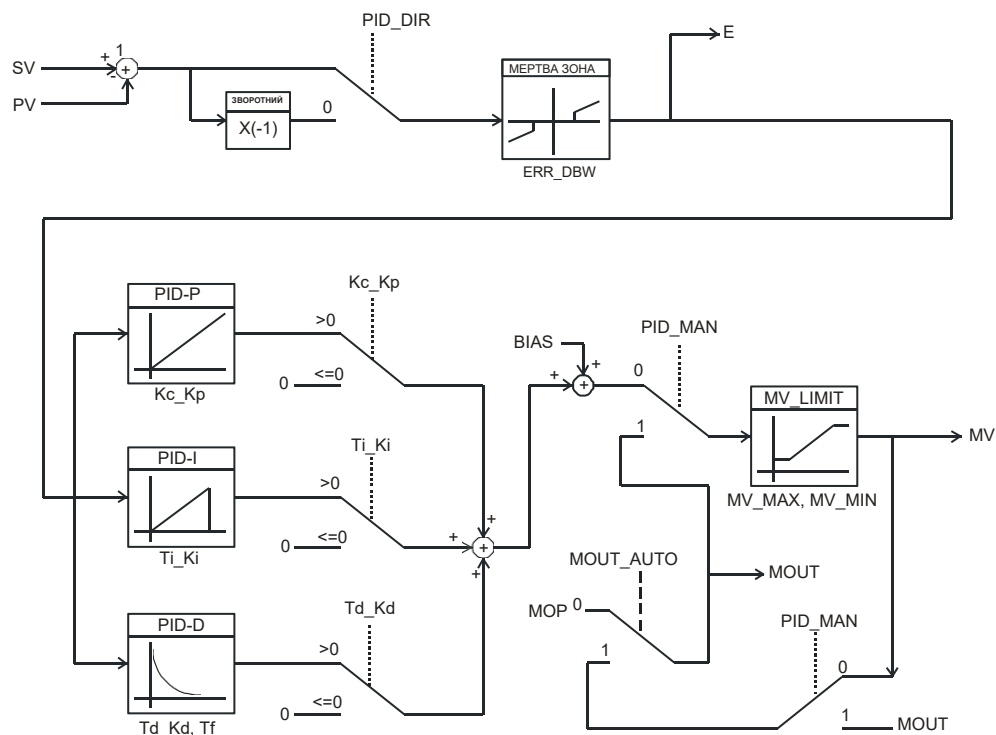
Або

$$MV = K_c \left[E + \frac{1}{T_i} \int_0^t E dt + T_d * \frac{dE}{dt} \right] + BIAS \quad E = PV - SV$$

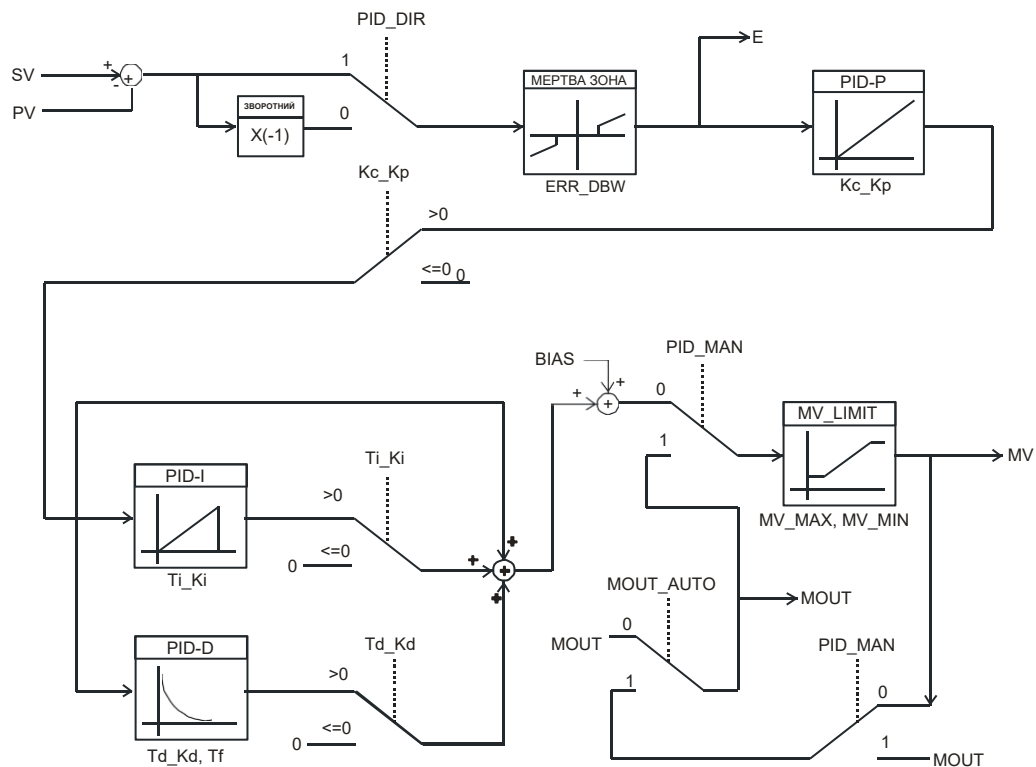
2. Коли ви встановлюєте **PID_MODE** на 1, режим ПІД-регулювання є режимом автоматичного налаштування. Після завершення налаштування параметра **PID_MODE** встановлюється на 0. Тоді режим ПІД-регулювання стає режимом автоматичного керування.

Блок-схема PID:

Блок-схема PID (незалежна)



Блок-схема PID (залежно)



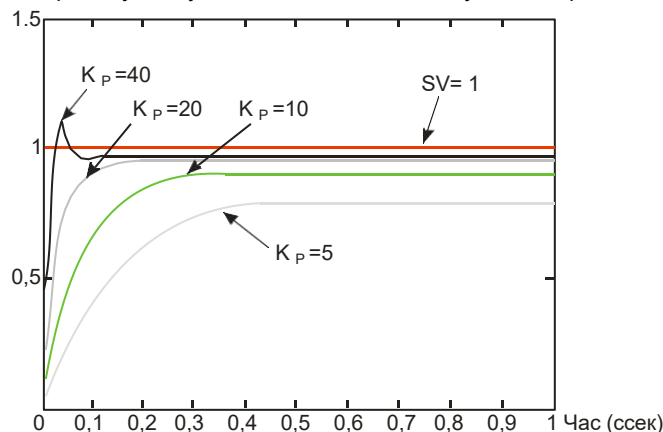
Пропозиції

- Оскільки ви можете використовувати 32-розрядну інструкцію в багатьох контрольованих середовищах, ви повинні вибрати відповідну функцію керування. Наприклад, щоб запобігти неправильному управлінню, не використовуйте **PID_MODE** в середовищі, керованому двигуном, якщо для нього встановлено значення 1.
- Коли ви налаштовуєте параметри **Kc_Kp**, **Ti_Ki** і **Td_Kd** (**PID_MODE** встановлено на 0), ви повинні спочатку налаштувати **KP** (на основі досвіду), а потім встановити **Ti_Ki** і **Td_Kd** на 0. Коли ви можете впоратися з керуванням, ви можете збільшити **Ti_Ki** і **Td_Kd**. Коли **Kc_Kp** дорівнює 1, це означає, що пропорційний приріст становить 100%. Тобто значення похибки збільшується в одиницю. Коли пропорційне посилення менше 100%, значення помилки зменшується. Коли пропорційне посилення перевищує 100%, значення похибки збільшується.
- Щоб запобігти зникненню автоматично налаштованих параметрів після втрати живлення, ви повинні зберегти параметри в зафіксованих регістрах даних, коли **PID_MODE** встановлено на 1. Параметри, які були автоматично налаштовані, не обов'язково підходять для кожного контрольованого середовища. Таким чином, ви можете змінити автоматично налаштовані параметри; однак пропонується модифікувати лише **Ti_Ki** та **Td_Kd**.
- Ви можете використовувати цю інструкцію з багатьма параметрами, але щоб запобігти неправильному контролю, не встановлюйте параметри випадково.

Приклад 2: Налаштування параметрів, що використовуються з інструкцією PID

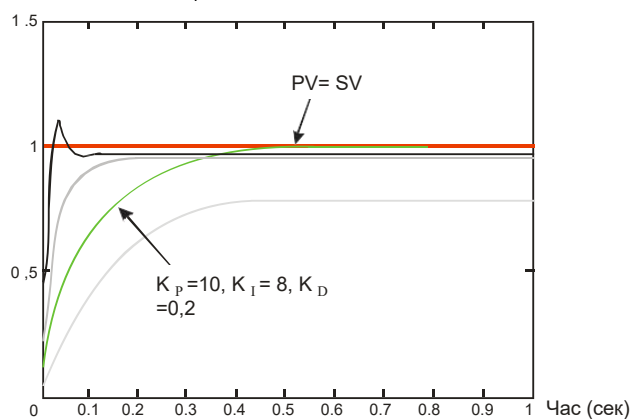
Припустимо, що передавальна функція установки є функцією першого порядку $G(s) = \frac{b}{s+a}$, SV дорівнює 1, час вибірки Ts дорівнює 10 мілісекунд. Під час налаштування параметрів рекомендується виконувати такі кроки.

Крок 1: спочатку встановіть K_I і K_D на 0. Потім послідовно встановіть K_P на 5, 10, 20 і 40 і запишіть цільові значення та значення процесу. Результати показані на наступній діаграмі.



Крок 2 : Коли K_P становить 40, спостерігається надмірна реакція. Коли K_P дорівнює 20, крива реакції PV близька до SV, і надмірної реакції немає. Однак через швидкий запуск перехідне вихідне значення (MV) є великим. Ні 40, ні 20 не підходять. Коли K_P дорівнює 10, крива реакції PV плавно наближається до SV. Коли K_P дорівнює 5, реакція йде надто повільно. Тому $K_P = 10$ є найкращим вибором.

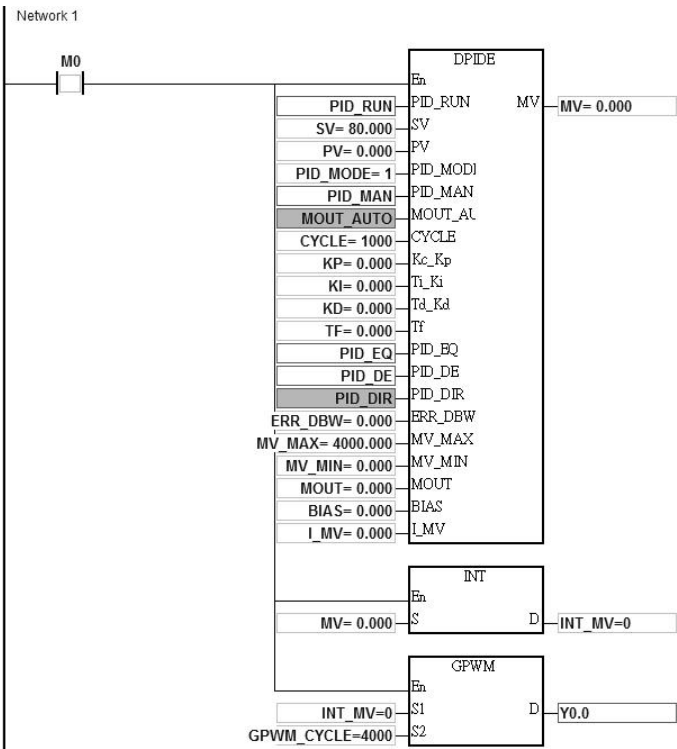
Крок 3 : після встановлення K_P на 10 збільште K_I . Наприклад, K_I послідовно встановлюється на 1, 2, 4 і 8. K_I не має бути більшим за K_P . Потім збільште K_D . Наприклад, послідовно встановіть K_D на 0,01, 0,05, 0,1 і 0,2. K_D не має перевищувати десяти відсотків K_P . Нарешті, співвідношення між PV і SV показано на наступній діаграмі.



Примітка. Цей приклад лише для довідки. Ви повинні правильно налаштувати параметри відповідно до фактичного стану системи керування.

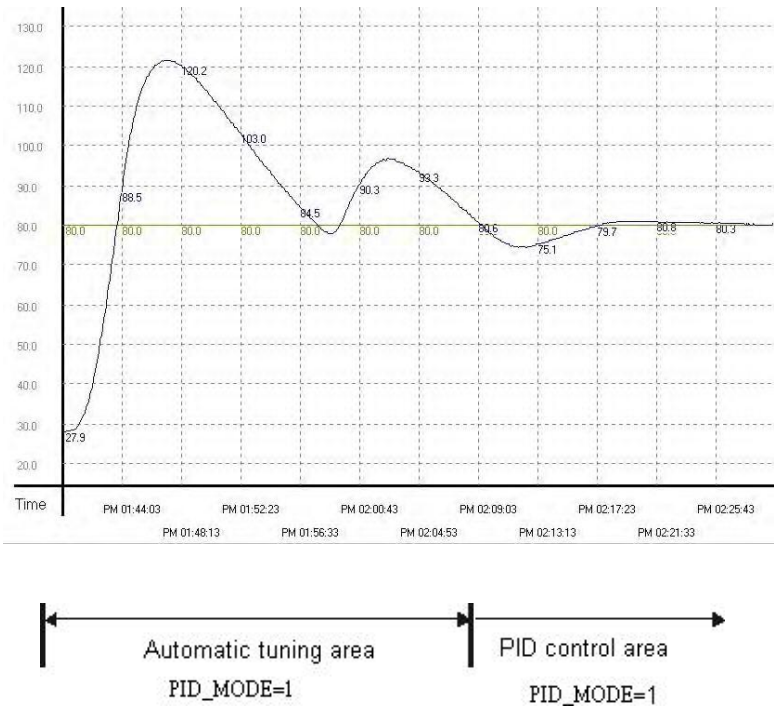
Приклад 3: Використання функції автоматичного налаштування для контролю температури

Оскільки ви можете бути не знайомі з характеристиками температурного середовища, яке потрібно контролювати, ви можете використовувати функцію автоматичного налаштування, щоб зробити початкове налаштування (**PID_MODE** встановлено на 1). Після завершення автоматичного налаштування параметра **PID_MODE** встановлюється на 0. Контрольоване середовище в цьому зразку – піч. У наступному прикладі програми показано значення налаштувань для інструкції.

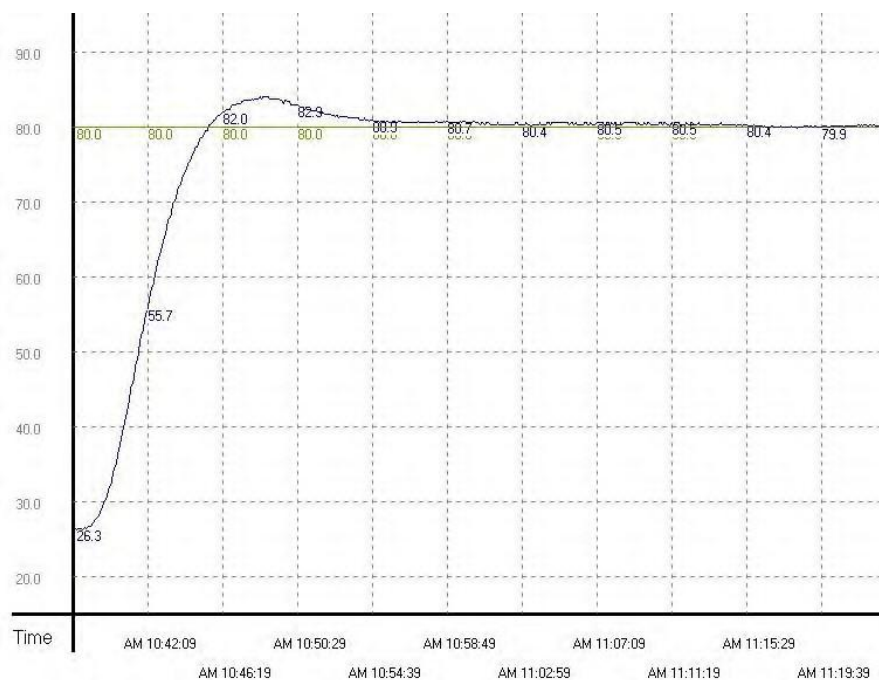


Експериментальний результат функції автоматичного налаштування показано на наступному графіку.

6

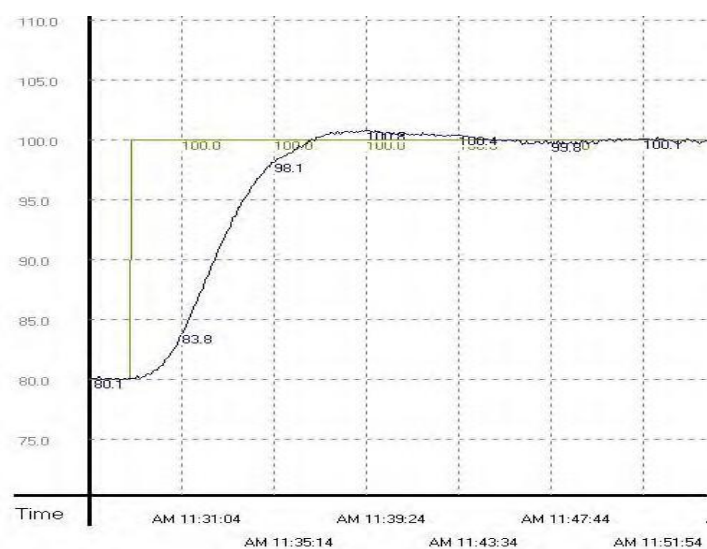


На наступному графіку показано результат використання автоматично налаштованих параметрів для контролю температури.



Цей графік показує, що використання автоматично налаштованих параметрів може призвести до хорошого результату контролю температури. Потрібно лише

близько двадцяти хвилин для контролю температури. На наступному графіку показано результат зміни цільової температури з 80°C до 100°C.



Цей графік показує, що коли цільова температура змінюється від 80°C до 100°C, автоматично налаштовані параметри залишаються

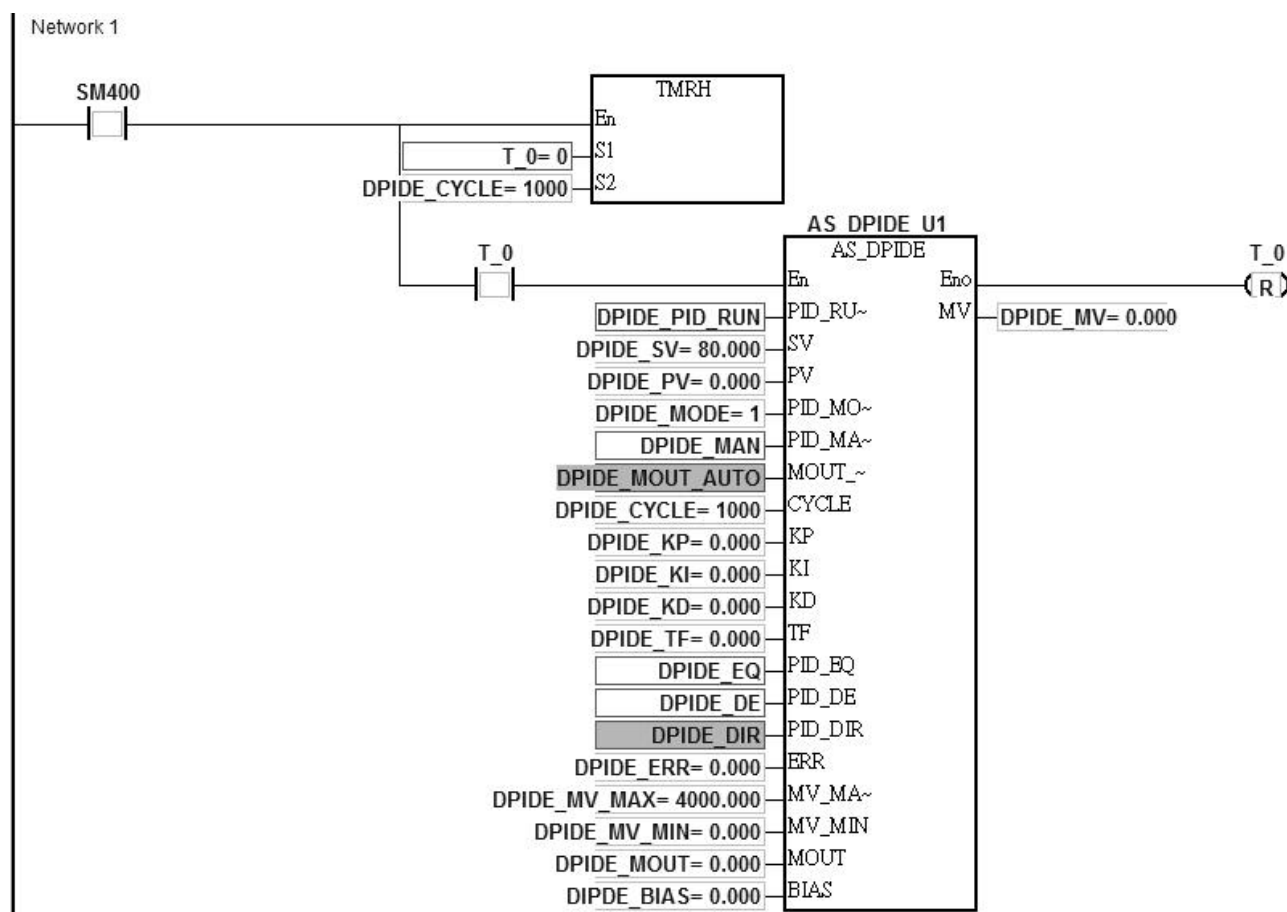
працювати над контролем температури в розумний проміжок часу.

Приклад 4: Створення інструкції DPIDE у функціональному блоці та встановлення режиму циклічного завдання для читання функціонального блоку, написаного за допомогою інструкції DPIDE, для керування температурою.

1. Встановіть значення DPIDE_CYCLE на 1000 мс і виконайте інструкцію DPIDE, прочитавши функціональний блок, написаний за допомогою інструкції DPIDE. Щоразу, коли функціональний блок сканується, алгоритм PID застосовується відповідно до часу вибірки (цикл) і оновлює вихідне значення (DPIDE_MV).

2. Встановіть DPIDE_MODE = 1 для автоматичного налаштування параметрів для контролю температури. Після завершення налаштування система перебуває в режимі автоматичного керування (PID_MODE встановлено на 0) і заповнить відповідні параметри (Kc_Kp, Ti_Ki, Td_Kd і Tf).

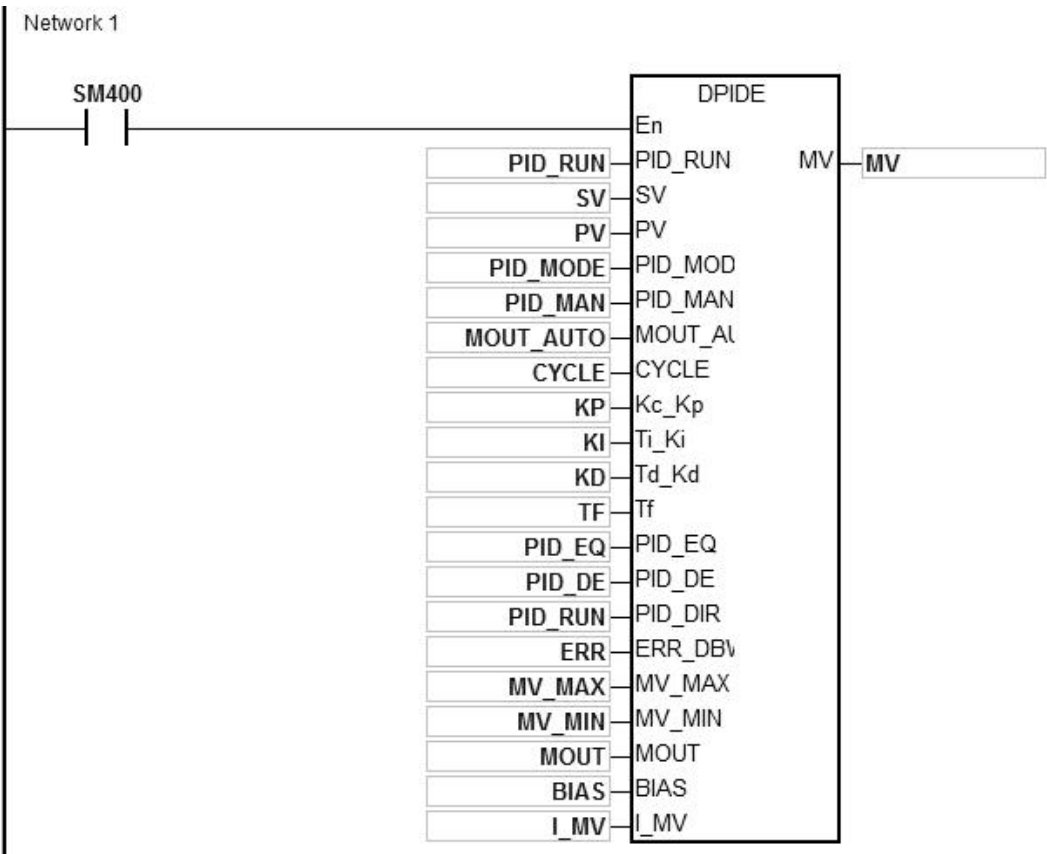
3. Основна програма (циклічне завдання): Оскільки ПЛК виконує інструкцію DPIDE лише під час її сканування. Якщо ми використовуємо TMRH для роботи з інструкцією DPIDE, наприклад, встановлюємо TMRH на 1000 мс, система викликає функціональний блок, написаний за допомогою інструкції DPIDE (AS_DPIDE) кожні 1000 мс. Дивіться приклад програми нижче.



4. Функціональний блок (AS_DPIDE): Виконайте інструкцію DPIDE, прочитавши функціональний блок, написаний за допомогою DPIDE

інструкція. Щоразу, коли функціональний блок сканується, алгоритм PID застосовується відповідно до часу вибірки (цикл) і оновлює вихідне значення (DPIDE_MV). (Детальніше про те, як створити функціональний блок, див. у посібнику ISPSOft.)

ПРИМІТКА: Параметри PID_MODE, Kc_Kp, Ti_Ki, Td_Kd, Tf і MV у функціональному блоці, написаному за допомогою інструкції DPIDE, повинні бути оголошені як VAR_IN_OUT.



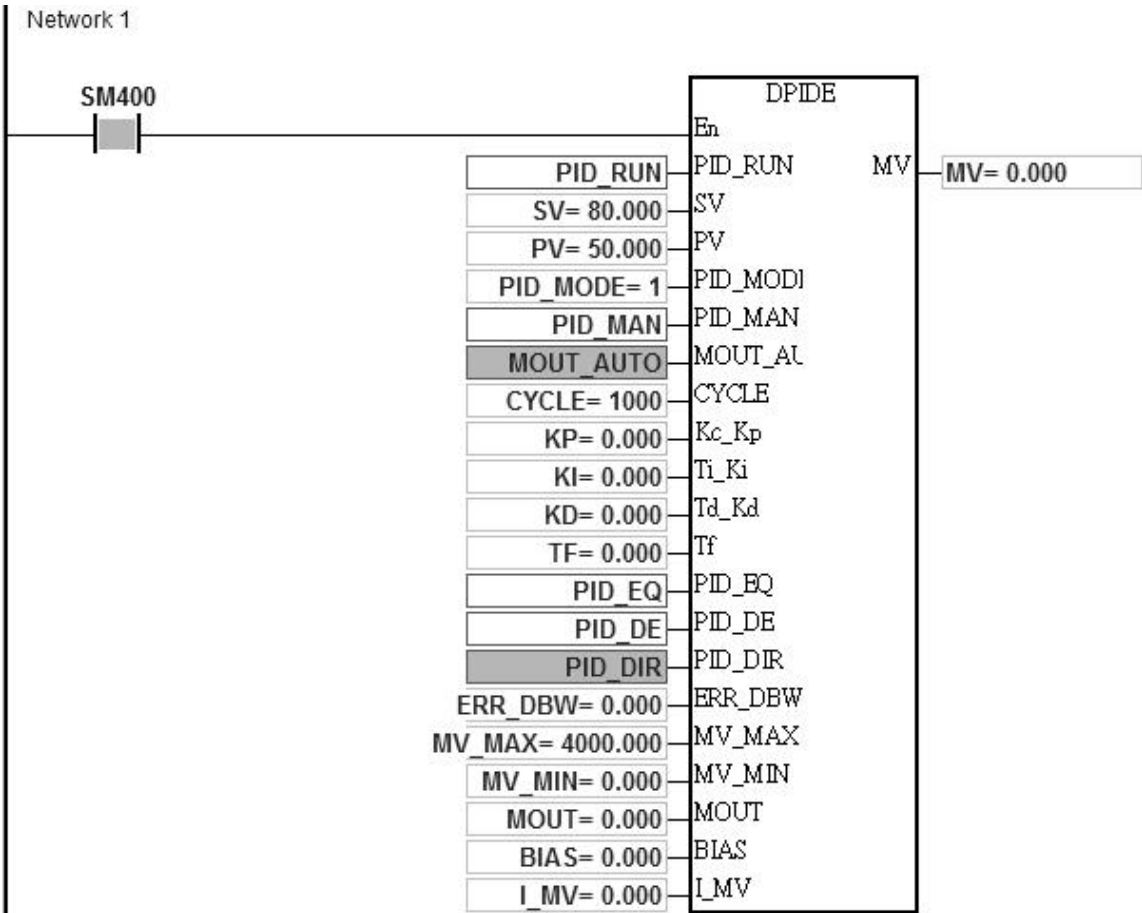
Приклад 5: Створення інструкції DPIDE у програмі переривання часу для керування температурою. (Примітка: використовуйте час переривання як час циклу DPIDE.)

1. Встановіть час переривання на 1000 мс у HWCONFIG.
2. Створіть інструкцію DPIDE у програмі переривання часу. Кожного разу, коли відбувається переривання часу, застосовується алгоритм PID. Параметр у DPIDE_CYCLE тут недійсний.
3. Встановіть DPIDE_MODE =1 для автоматичного налаштування параметрів для контролю температури. Після завершення налаштування система перебуває в режимі автоматичного керування (PID_MODE встановлено на 0) і заповнить відповідні параметри (Kc_Kp, Ti_Ki, Td_Kd і Tf).

Основна програма (циклічне завдання)



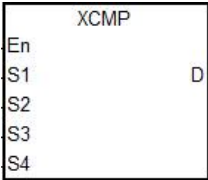
Програма переривання часу I601 і параметри налаштування



API	Код інструкції				Операнд						функція						
0709		XCMP			S_1, S_2, S_3, S_4, D						Налаштування для порівняння вхідних даних кількох робочих станцій						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	○																
S_2							○										
S_3								○									
S_4								○									
D								○									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2												●	
S_3		●				●							
S_4			●				●						
D			●				●						

СИМВОЛ	Інструкція пульсу	16-розрядна інструкція	32-бітна інструкція
	-	AS	-



- S_1 : Точка введення тригера
- S_2 : Швидкісний лічильник номера
- S_3 : Налаштування номерів робочого місця та об'єктів
- S_4 : Еталонне значення для порівняння та похибка спостереження
- D : Перший відповідний пристрій для результату порівняння в області стека

Пояснення

- Ця інструкція доступна лише для ПЛК серії AS із мікропрограмою версії 1.04.00 або новішої. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.
- Використовуйте цю інструкцію разом із API 0710 YOUT. Це дуже корисно для додатків складання та обробки з кількома робочими станціями, включаючи сектори фарбування, столярні роботи, слюсарні роботи та багато іншого.
- Використовуйте S_1 для встановлення вхідних точок тригера; високошвидкісні входи X0.0–X0.15, а інші входи загального типу. Виконання інструкції дозволяє зовнішні переривання для входів (X0.0–X0.15). Тому НЕ використовуйте входи із завданнями переривання; інакше, коли інструкція виконується, переривання вимикаються і відновлюються лише після завершення інструкції. На вхідні дані загального типу впливає час сканування, хоча вони підходять для середовищ, де вхідні дані не такі стабільні.

4. S_2 працює з 32-розрядними лічильниками (HC0–HC255). Якщо входи мають тип входу високошвидкісного тригера, реалізуйте апаратний високошвидкісний лічильник і використовуйте інструкцію DCNT, щоб увімкнути лічильник. Якщо вам потрібен високошвидкісний вихід, використовуйте інструкцію DMOV, щоб скопіювати поточну позицію виводу; наприклад копіювання осі SR460 до HC0 (DMOV SR460 HC0).

5. Для ПЛК із FW V1.04.00 (V1.04.00 включено) або попередніми версіями:

S_3 займає три послідовні 16-розрядні пристрої. $S_3 + 0$ – це n (номер робочої станції), а $S_3 + 1$ – це m (максимальна кількість об'єктів). $S_3 + 2$ – це результат фільтрації об'єкта. Діапазон для n і m становить від 1 до 32. Якщо це значення виходить за межі діапазону, використовується максимальне (32) або мінімальне значення (1). Діапазон для $S_3 + 2$ (номер фільтра) знаходиться між 0–32767. Нуль використовується для будь-якого значення менше 0, а значення 0 вимикає функцію фільтрації. Не забудьте оголосити масив із 3 слів або 3 послідовних змінних типу слова.

6. Для ПЛК із FW V1.06.00 (V1.06.00 включно) або новіших версій:

S_3 займає сім послідовних 16-розрядних пристроїв. $S_3 + 0$ – це n (номер робочої станції), а $S_3 + 1$ – це m (максимальна кількість об'єктів). $S_3 + 2$ – це результат фільтрації об'єкта. Діапазон для n і m становить від 1 до 64. Якщо це значення виходить за межі діапазону, використане значення розглядається як максимальне (32) або мінімальне (1). Діапазон для $S_3 + 2$ (номер фільтра) знаходиться між 0–32767. Нуль використовується для будь-якого значення менше 0; а значення 0 вимикає функцію фільтрації. $S_3 + 3$ (нижче слово) і $S_3 + 4$ (високе слово) призначені для наростаючих значень. $S_3 + 5$ (Low Word) і $S_3 + 6$ (High Word) призначені для спадаючих значень (32-біт). Не забудьте оголосити масив із 7 слів або 7 послідовних змінних типу слова.

7. Встановіть максимальну кількість для $S_3 + 1$ (m). Якщо $m < n$, зверніть увагу на об'єкти та переконайтеся, що їх достатньо на виробничій лінії.

8. S_4 займає $3n$ послідовних 32-розрядних пристроїв ($6n$ 16-розрядних пристроїв). Якщо необхідний простір перевищує діапазон пристрою D, інструкція не виконується. Значення n є номером робочої станції, встановленим в операнді S_3 . У наступній таблиці наведено функції для кожного пристрою та відповідний номер для S_4 . Не забудьте оголосити масив із $3n$ подвійних слів або 3 послідовних змінних типу подвійних слів для S .

функція	Робоча станція 1	Робоча станція 2	...	Робоча станція n
Еталонне значення для порівняння (32-розрядний)	$S_4 + 0$	$S_4 + 2$	...	$S_4 + (n-1) \times 2$
Похибка спостереження порівнюваного область стека для входу на робочу станцію (32-розрядна)	$S_4 + 2 \times n$	$S_4 + 2 \times n + 2$	...	$S_4 + (2 \times n - 1) \times 2$
Похибка спостереження порівнюваного область стеку для тих, що залишають робочу станцію (32-розрядна)	$S_4 + 4 \times n$	$S_4 + 4 \times n + 2$	...	$S_4 + (3 \times n - 1) \times 2$

Якщо для певної робочої станції встановлено значення 0, ця робоча станція припиняє працювати. Ви можете використовувати цю техніку для керування робочими станціями.

9. D є першим відповідним пристроєм для результату порівняння в області стека. D займає $2 \times n$ послідовних 16-розрядних пристроїв і $2 \times m \times n$ послідовних 32-розрядних пристроїв (або $4 \times m \times n$ послідовних 16-розрядних пристроїв). Якщо необхідний простір перевищує діапазон пристрою D, інструкція не виконується. У наступній таблиці наведено функції для кожного пристрою та відповідний номер для D .

функція	Робоча станція 1	Робоча станція 2	...	Робоча станція n
Значення головного індексу (16-біт)	$D + 0$	$D + 1$	...	$D + (n-1)$
Значення хвостового індексу (16-біт)	$D + n$	$D + (n+1)$	...	$D + (2xn-1)$
Порівняно область стека 1 для вхідні робочі станції (32-розрядні)	$D + 2xn$	$D + 2xn+2$	...	$D + 2xn+2(n-1)$
Порівняно область стека 1 для вихідні робочі станції (32-розрядні)	$D + 4xn$	$D + 4xn+2$	...	$D + 4xn+2(n-1)$
:	:	:	:	:
Порівняна площа стека m для вхідні робочі станції (32-розрядні)	$D + 4xmxn-2xn$		...	$D + 4xmxn-2$
Порівняна площа стека m для вихідні робочі станції (32-розрядні)	$D + 4xmxn$		...	$D + 4xmxn+2 (n-1)$

D має тенденцію займати більше місця в області стека. Якщо необхідний простір перевищує діапазон пристрою **D**, лише ПЛК

виконує те, що є дійсним у сховищі, і не показує попередження про відсутність. Пропонується оголосити масив із $2xn+4xmxn$ слів для **D**.

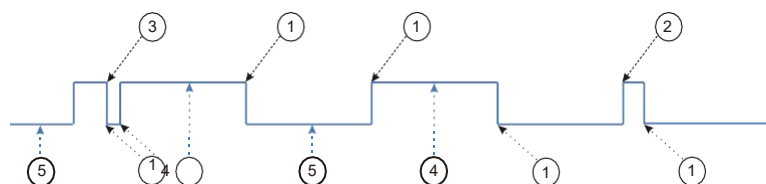
10. Немає обмежень на кількість разів, які ви можете виконати інструкцію, але за один раз можна виконати лише одне виконання.

11. Рекомендується використовувати цю інструкцію з інструкцією YOUT (API 0710) і використовувати той самий перший відповідний

пристрій для результату порівняння в області стека (**D**).

12. Для ПЛК із FW V1.04.00 (V1.04.00 включно) або попередніми версіями:

На наступній часовій діаграмі показано виконання високошвидкісного лічильника та фільтра (читання справа наліво).



① ПЛК зчитує поточне значення лічильника та зберігає значення в регістрі.

② Зменште значення лічильника зміщення: коли сигнал нестабільний (кількість фільтрів недостатня).

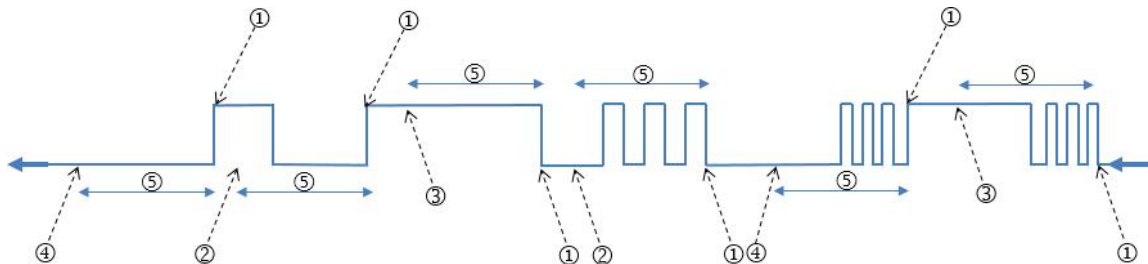
③ Зменште значення лічильника зміщення: коли сигнал нестабільний (кількість фільтрів недостатня).

④ Запишіть значення лічильника в порівнювану область стека для входу на робочу станцію, коли сигнал стабільний.

- ⑤ Запишіть значення лічильника в порівнювану область стека для тих, що залишають робочу станцію, коли сигнал стабільний.

13. Для ПЛК із FW V1.06.00 (V1.06.00 включно) або вищими версіями:

На наступній часовій діаграмі показано виконання високошвидкісного лічильника та фільтра (читання справа наліво).



- ① ПЛК зчитує поточне значення лічильника та зберігає значення в регістрі.
- ② Відкиньте значення лічильника: кількість зчитаних фільтрів менша за кількість встановлених фільтрів.
- ③ Запишіть значення лічильника в порівнювану область стека для входу на робочу станцію, коли сигнал стабільний.
- ④ Запишіть значення лічильника в порівнювану область стека для тих, що залишають робочу станцію, коли сигнал стабільний.
- ⑤ Кількість фільтрів

14. Коли сигнал запускається по висхідному або спадаючому фронту, і ПЛК завершує обробку фільтрів, ПЛК зчитує значення високошвидкісного лічильника та додає одиницю до значення головного індексу. Потім ПЛК записує результати лічильників входу та виходу для кожної робочої станції. Порівняний результат лічильника – це поточне значення лічильника + еталонне значення + помилка спостереження. Для спрацьовування по наростаючому або спадаючому фронту значення індексу голови збільшується. Максимальне значення індексу голови $mx2$ (максимальна кількість об'єктів).

15. Значення головного індексу циклічно збільшується, коли сигнал запускається по висхідному або спадаючому фронту та завершує обробку кількості фільтрів (за замовчуванням для входу тригера вимкнено). Максимальне значення індексу голови – $mx2$ (максимальна кількість об'єктів). Наприклад, якщо ви встановите кількість об'єктів на 10, значення головного індексу (за замовчуванням: 0) буде збільшено до 1, 2, 3–20, а потім повторно від 1, 2, 3 до 20. Якщо значення головного індексу дорівнює 0, це означає, що жоден об'єкт не ввійшов після виконання інструкції. ПЛК додає одиницю до значення головного індексу, а потім перевіряє значення хвостового індексу. Якщо значення (після додавання одиниці) у значенні головного індексу дорівнює значенню хвостового індексу, ПЛК скасовує додавання та записує результат лічильника.

16. Коли інструкція виконується і стан початкового введення **OFF**, **тригер по наростаючому фронту** відповідає непарним **числам** значення індексу голови, а **тригер по спадному фронту** відповідає **парним числам** значення індексу голови.

17. Коли ПЛК виконує інструкцію, а стан початкового введення є **ON** , **тригер по спадному фронту** відповідає непарним **числам** значення індексу голови, а **тригер по наростаючому фронту** відповідає **парним числам** значення індексу голови.

18. Коли ПЛК виконує інструкцію, він не очищає значення в накопиченій області та областях індексу. Якщо дані знаходяться в зафіксованій області та їх потрібно знову ввімкнути, скористайтеся інструкцією ZRST, щоб очистити значення в індексах голови та хвоста.

приклад

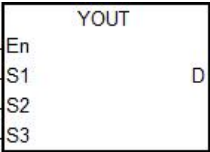
Для отримання додаткової інформації зверніться до прикладу в інструкції YOUT (API 0710).

API	Код інструкції				Операнд						функція						
0710		YOUT			S_1, S_2, S_3, D						Порівняння результатів кількох робочих станцій						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1							O										
S_2								O									
S_3								O									
D		O	O														

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2		●				●							
S_3			●				●						
D	●												

Інструкція пульсу		16-розрядна інструкція	32-бітна інструкція
-		AS	-

СИМВОЛ



- S_1 : Швидкісний лічильник номера
- S_2 : Налаштування кількості робочих місць і об'єктів
- S_3 : Перший відповідний пристрій для результату порівняння в області стека
- D : Перший відповідний пристрій для вихідної робочої станції

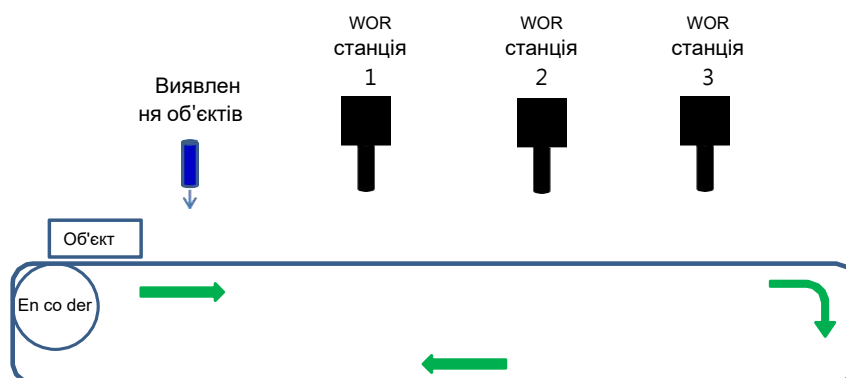
Пояснення

- Ця інструкція доступна лише для ПЛК серії AS із мікропрограмою версії 1.04.00 або вище. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.
- S_1 призначений для налаштування високошвидкісного лічильника. Використовуйте ті самі налаштування для високошвидкісного лічильника, що й для лічильник швидкості для інструкції XCMP.
- S_2 займає два послідовних 16-розрядних пристроїв. $S_2 + 0$ - це n (номер робочої станції), а $S_2 + 1$ - це m (максимальна кількість об'єктів). Діапазон для n і m становить від 1 до 32 для ПЛК із FW V1.04.00 (включно з V1.04.00) або попередніми версіями. Діапазон для n і m становить від 1 до 64 для ПЛК із FW V1.06.00 (включно з V1.06.00) або новішими версіями. Якщо значення виходить за межі діапазону, використовується максимальне (32) або мінімальне значення (1). Параметри для операндів мають бути такими ж, як і для інструкції XCMP.
- S_3 є першим відповідним пристроєм для результату порівняння в області стека. S_3 займає 2хn послідовних 16-біт і 2хmхn послідовних 32-розрядних пристроїв (або 4хmхn послідовних 16-розрядних пристроїв). Для отримання інформації про функції кожного пристрою та відповідний номер для D див. інструкцію XCMP (API 0709). Рекомендується використовувати ту саму змінну, що й для інструкції XCMP.

5. Немає обмежень щодо кількості разів, які ви можете виконати інструкцію, але за один раз можна виконати лише одне виконання.
6. Рекомендується використовувати з інструкцією XCMP і використовувати той самий перший відповідний пристрій для результату порівняння в області стеку (S_3).
7. D лише для виходів пристроїв Y і M; Y і M мають бути типом даних BOOL. Він займає послідовну кількість робочих станцій Xn. Якщо використовується як точка виведення пристрою Y або M, інструкція оновлює вихідні стани.
8. Значення індексу заголовка з непарними номерами (наприклад, 1, 3, 5,...) є порівнюваними результатами лічильника для об'єкта під час входу (порівняні області стека для входу робочої станції). Парні значення головного індексу (наприклад, 2, 4, 6,...) є результатом порівняння лічильника об'єкта під час виходу (порівняні області стека для вхідних робочих станцій).
9. Коли порівнюваний результат лічильника для входу та виходу в область стеку дорівнює 0, дії в цій області не виконуються, а стан відповідної робочої станції виводу OFF. Додайте 2 до значення хвостового індексу, і додана вартість хвостового індексу не повинна перевищувати значення головного індексу.
10. Коли виконується інструкція YOUT, кожна робоча станція перевіряє порівнюване значення для входу та виходу в індекс хвоста. Коли значення лічильника більше або таке ж, як порівнюване значення для введення, відповідна вихідна точка ввімкнена та додає 1 до значення хвостового індексу. Коли значення лічильника більше або таке саме, як порівнюване значення для виходу, відповідний вихід вимкнено та додає 1 до значення хвостового індексу; але значення хвостового індексу (після додавання 1) не перевищує значення головного індексу.

6

Приклад: три робочі станції і до чотирьох об'єктів



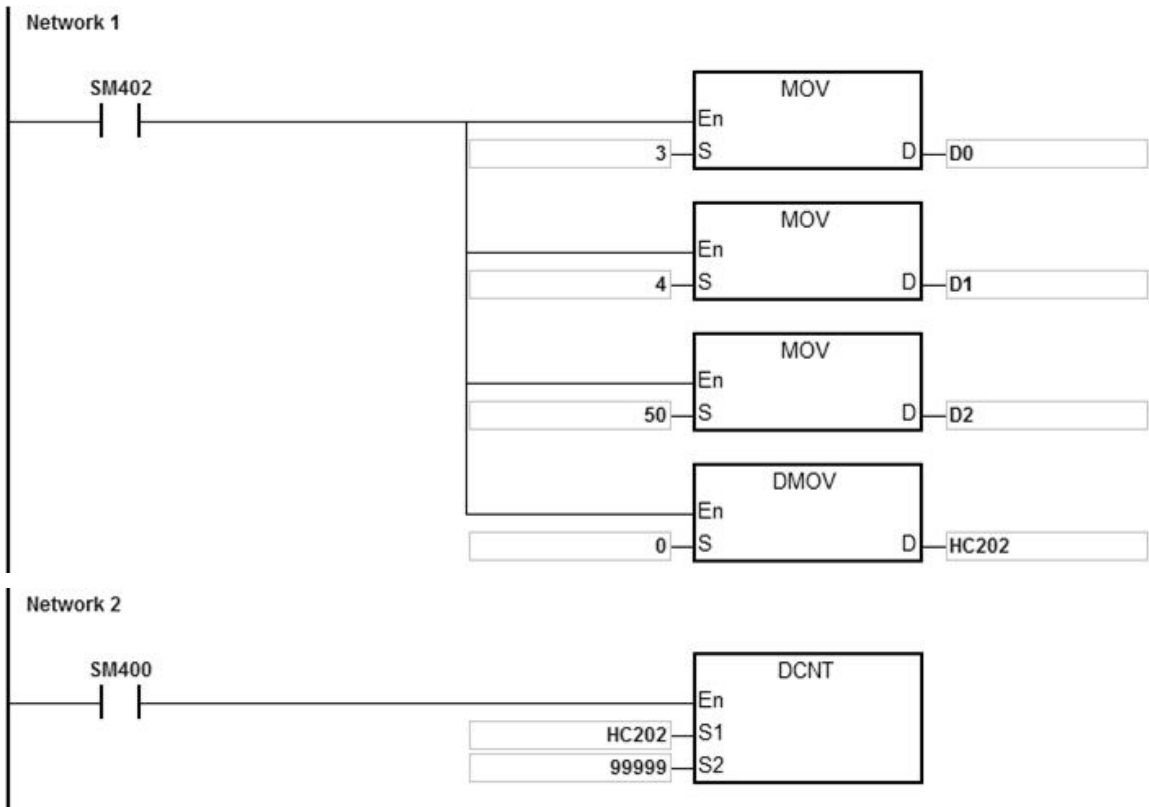
Крок 1: використовуйте вхідну точку X0.4 як переривання виявлення об'єкта, HC202 як високошвидкісний лічильник для кодера та

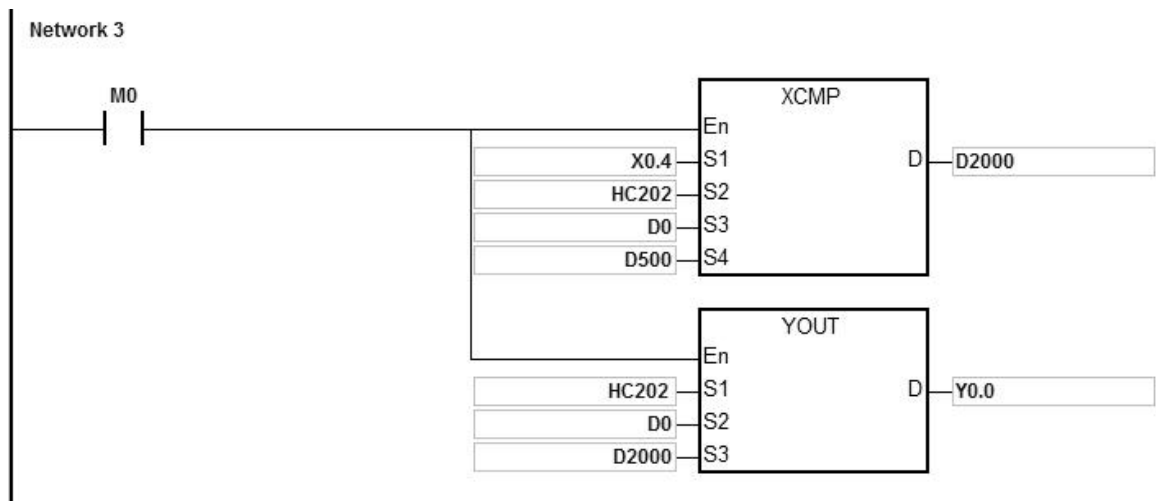
вихідна точка Y0.0 як перша вихідна точка для робочої станції.

Крок 2: відредагуйте реєстр, щоб встановити контрольні значення та помилку спостереження під час входу та виходу.

Пристрій D	D500	D502	D504
Еталонне значення для порівняння (32-розрядний)	K2000	K3000	K4000
Пристрій D	D506	D508	D510
Помилка спостереження при введенні (32-біт)	K100	K120	K130
Пристрій D	D512	D514	D516
Помилка спостереження при виході (32-біт)	K50	K-20	K20
Пристрій D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K0	K0	K0
Пристрій D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K0	K0	K0

Крок 3: встановіть початкові значення та напишіть програми.





Налаштуйте три робочі станції для D0, 4 об'єкти для D1 і 50 фільтрів для D2. Після спрацювання контакту M0 система встановлюється

виявлення об'єкта, порівнювані значення, порівняний результат лічильника входу та виходу об'єкта, а також контроль виведення для кожної робочої станції. Наприклад, система виявляє введення двох об'єктів, а потім чотири тригери для зчитування порівняних результатів лічильника: 3000, 3500, 4500 і 5000 у HC202 (HC202=K5060). У наведеній нижче таблиці показано порівняльне значення та індекс голови/кінця в області стеку.

Пристрій D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K4	K4	K4
Номер пристрою D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K1	K1	K1
Номер пристрою D	D2006	D2008	D2010
Порівняно область стека 1 для входу -work- станційні (32-розрядні)	K5100	K6120	K7130
Номер пристрою D	D2012	D2014	D2016
Порівняно область стека 1 для вихідної роботи- станційні (32-розрядні)	K5550	K6480	K7520
Номер пристрою D	D2018	D2020	D2022
Порівняно область стека 1 для входу -work- станційні (32-розрядні)	K6600	K7620	K8630
Номер пристрою D	D2024	D2026	D2028
Порівняна площа стека 2 для вихідної роботи- станційні (32-розрядні)	K7050	K7980	K9020
Номер пристрою D	D2030	D2032	D2034

Порівняно область стека 3 для входу -work- станційні (32-розрядні)	K0	K0	K0
Номер пристрою D	D2036	D2038	D2040
Порівняна площа стека 3 для вихідної роботи- станційні (32-розрядні)	K0	K0	K0

У наступній таблиці показано стан вихідної точки Y, коли високошвидкісний лічильник HC202 досягає 5200.

Вихідна точка Y число	Y0.0	Y0.1	Y0.2
16-бітне значення	ON	OFF	OFF
Номер пристрою D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K4	K4	K4
Номер пристрою D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K2	K1	K1

У наступній таблиці показано стан вихідної точки Y, коли високошвидкісний лічильник HC202 досягає 6200.

Вихідна точка Y число	Y0.0	Y0.1	Y0.2
16-бітне значення	OFF	ON	OFF
Номер пристрою D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K4	K4	K4
Номер пристрою D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K3	K2	K1

У наступній таблиці показано стан вихідної точки Y, коли високошвидкісний лічильник HC202 досягає 6800.

Вихідна точка Y число	Y0.0	Y0.1	Y0.2
16-бітне значення	ON	OFF	OFF
Номер пристрою D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K4	K4	K4
Номер пристрою D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K4	K3	K1

У наступній таблиці показано стан вихідної точки Y, коли високошвидкісний лічильник HC202 досягає 7300.

Вихідна точка Y число	Y0.0	Y0.1	Y0.2
16-бітне значення	OFF	OFF	ON
Номер пристрою D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K4	K4	K4
Номер пристрою D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K4	K3	K2

У наступній таблиці показано стан вихідної точки Y, коли високошвидкісний лічильник HC202 досягає 7700.

Вихідна точка Y число	Y0.0	Y0.1	Y0.2
16-бітне значення	OFF	ON	OFF
Номер пристрою D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K4	K4	K4
Номер пристрою D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K4	K4	K3

У наступній таблиці показано стан вихідної точки Y, коли високошвидкісний лічильник HC202 досягає 8000.

Вихідна точка Y число	Y0.0	Y0.1	Y0.2
Вихідний стан	OFF	OFF	OFF
Номер пристрою D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K4	K4	K4
Номер пристрою D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K4	K4	K3

У наступній таблиці показано стан вихідної точки Y, коли високошвидкісний лічильник HC202 досягає 8700.

Вихідна точка Y число	Y0.0	Y0.1	Y0.2
Вихідний стан	OFF	OFF	ON
Номер пристрою D	D2000	D2001	D2002
Значення головного індексу (16-біт)	K4	K4	K4
Номер пристрою D	D2003	D2004	D2005
Значення хвостового індексу (16-біт)	K4	K4	K4

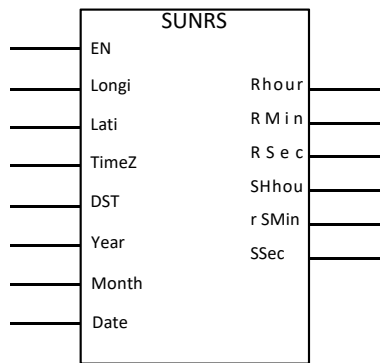
API	Код інструкції				Операнд						функція					
0711	D	SUNRS	P	Longi ~ SSec						Час сходу та заходу сонця						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Longi								●								○
Lati								●								○
TimeZ								●					○			
DST	●	●	●	●												
Year								●					○			
Month								●					○			
Date								●					○			
Rhour								●								
Rmin								●								
RSec								●								
SHour								●								
SMin								●								
SSec								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Longi									●				
Lati									●				
TimeZ		●				●							
DST	●												
Year		●				●							
Month		●				●							
Date		●				●							
Rhour		●				●							
Rmin		●				●							
RSec		●				●							
SHour		●				●							
SMin		●				●							
SSec		●				●							

Інструкція пульсу	16-розрядна інструкція	32-бітна інструкція
AS		AS

СИМВОЛ

Longi	:	Довгота (тип REAL)
Lati	:	Широта (тип REAL)
TimeZ	:	Часовий пояс (ціле число) (-12 ~ +14)
DST	:	Літній час
Year		рік
Month		місяць



Date	Дата
Rhour	Година до сходу сонця на встановлену дату (24-годинний формат часу)
Rmin	Хвилина до сходу сонця встановленої дати
RSec	Другий до сходу сонця встановленої дати
SHour	Година до заходу сонця на встановлену дату (24-годинний формат часу)
SMin	Хвилина до заходу встановленої дати
SSec	Другий до заходу сонця у встановлену дату

Пояснення

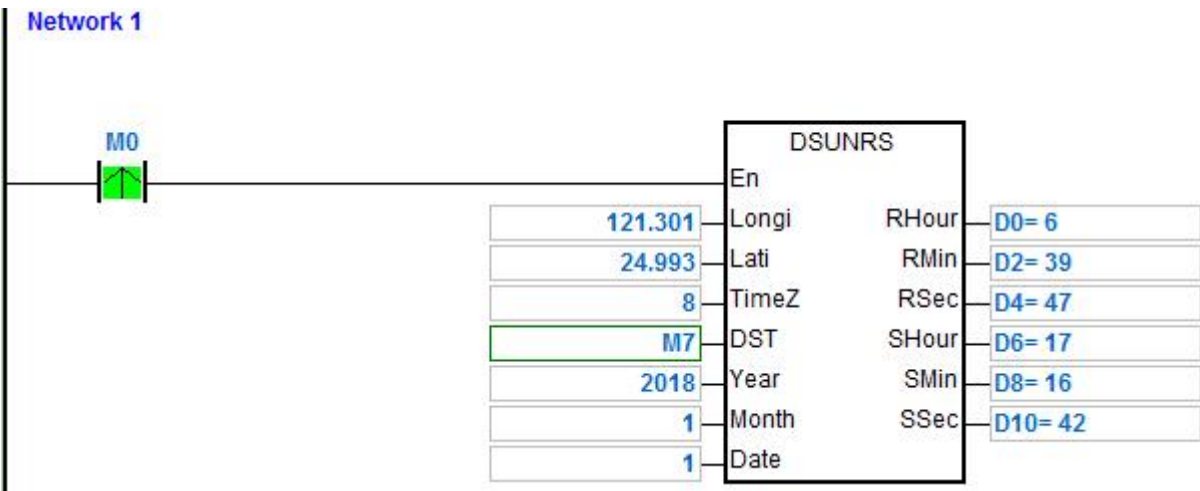
- Інструкція працює з прошивкою ПЛК серії AS версії 1.04.50 або новішої. Часи сходу та заходу сонця можуть не бути точніше, ніж публікує місцевий прогноз погоди, оскільки введені вами значення можуть бути неправильними або висота над рівнем моря, де встановлено пристрій, може перешкоджати точності. Якщо результат не такий точний, ви можете налаштувати значення вручну. Після самооцінки діапазон помилок цієї інструкції становить менше 5 хвилин.
- Введіть значення місцевої довготи та широти в цифрах. Наприклад, довгота та широта Таоюаня, Тайвань, становить 121,30098 і 24,99363. Широти на північ від екватора позначаються позитивним знаком. Широти півд екватора надаються від'ємні значення.
- Введіть значення для місцевого часового поясу в діапазоні від -12 до +14. Часовий пояс не можна обчислити за встановленою довготою та широтою; якщо налаштування виходить за межі діапазону або значення неправильне, повідомлення про помилку не відобразатиметься.
- Коли літній час увімкнено (ON), інструкція перевіряє, чи увімкнено літній час на ПЛК. Коли на ПЛК увімкнено літній час, літній час буде додано на схід і захід сонця разів.
- Введіть значення для місцевої дати, місяця та року в десятковому форматі. Переконайтеся, що ви ввели правильні значення. Інструкція не перевіряє, чи правильно введені значення.
- Після обчислення інструкція виводить години, хвилини та секунди до сходу та заходу сонця в цілих числах у десятковому форматі та 24-годинному форматі часу.

приклад

Розрахуйте час до сходу та заходу сонця в Таоюані, Тайвань, 1 січня ²⁰¹⁸ року. Офіційний часовий пояс на Тайвані визначається зсувом UTC на +08:00. І щоденне збереження часу NE реалізовано на Тайвані.

Після розрахунку, час сходу сонця - 06:39:47, а час заходу - 17:16:42. Дивіться приклад програми нижче.

Перевірено на офіційному веб-сайті погоди, фактичний схід сонця відбувся о 06:39:44, а фактичний захід відбувся о 17:16:45. Різниця між розрахунком PLC і фактичним випадком становить ± 3 секунди.



6.9 Логічні інструкції

6.9.1 Список логічних інструкцій

У наведеній нижче таблиці наведено логічні інструкції, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
0800	WAND	DAND	✓	Логічна операція AND
0801	MAND	–	✓	Матрична операція AND
0802	WOR	DOR	✓	Логічна операція OR
0803	MOR	–	✓	Матрична операція OR
0804	WXOR	DXOR	✓	Логічна операція виключаючого OR
0805	MXOR	–	✓	Матрична операція виняткового OR
0808	WINV	DINV	✓	Логічна зворотна операція INV
0809	LD&	DLD&	–	$S_1 \& S_2$
0810	LD	DLD	–	$S_1 S_2$
0811	LD^	DLD^	–	$S_1 \wedge S_2$
0812	AND&	DAND&	–	$S_1 \& S_2$
0813	AND	DAND	–	$S_1 S_2$
0814	AND^	DAND^	–	$S_1 \wedge S_2$
0815	OR&	DOR&	–	$S_1 \& S_2$
0816	OR	DOR	–	$S_1 S_2$
0817	OR^	DOR^	–	$S_1 \wedge S_2$

6.9.2 Пояснення логічних інструкцій

API	Код інструкції			Операнд							функція						
0800	D	WAND	P	S_1, S_2, D							Логічна операція AND						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●		○	○	○	○		
S_2	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

WAND	WANDP
En	En
S1 D	S1 D
S2	S2
DANDP	DAND
En	En
S1 D	S1 D
S2	S2

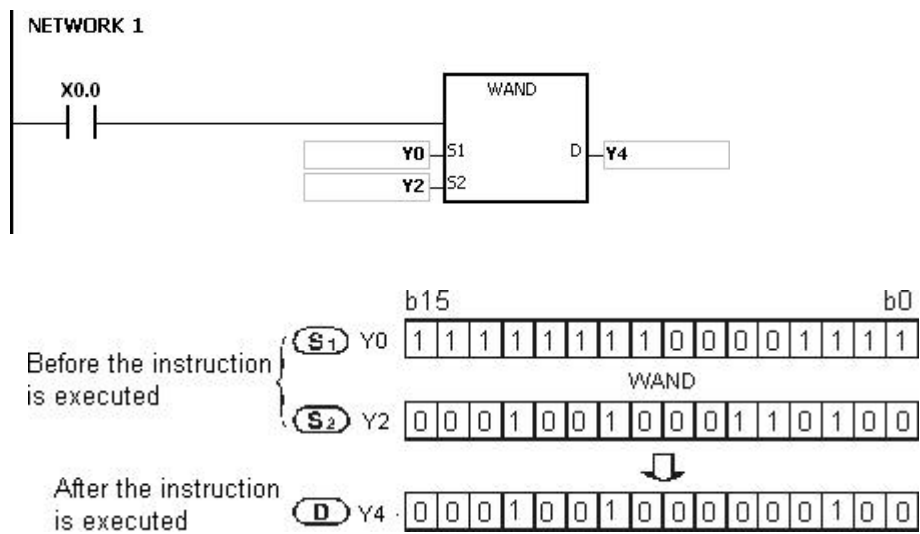
- S_1 : Джерело даних 1
- S_2 : Джерело даних 2
- D : Результат операції

Пояснення

1. Ця інструкція застосовує логічний оператор AND до двійкових представлень у S_1 і S_2 . Вона виконує логічну Операція AND над кожною парою відповідних бітів і зберігає результат у D .
2. Тільки інструкція DAND може використовувати 32-розрядний лічильник.
3. Результат у кожній позиції дорівнює 1, якщо перший біт дорівнює 1, а другий біт — 1; інакше результат дорівнює 0.

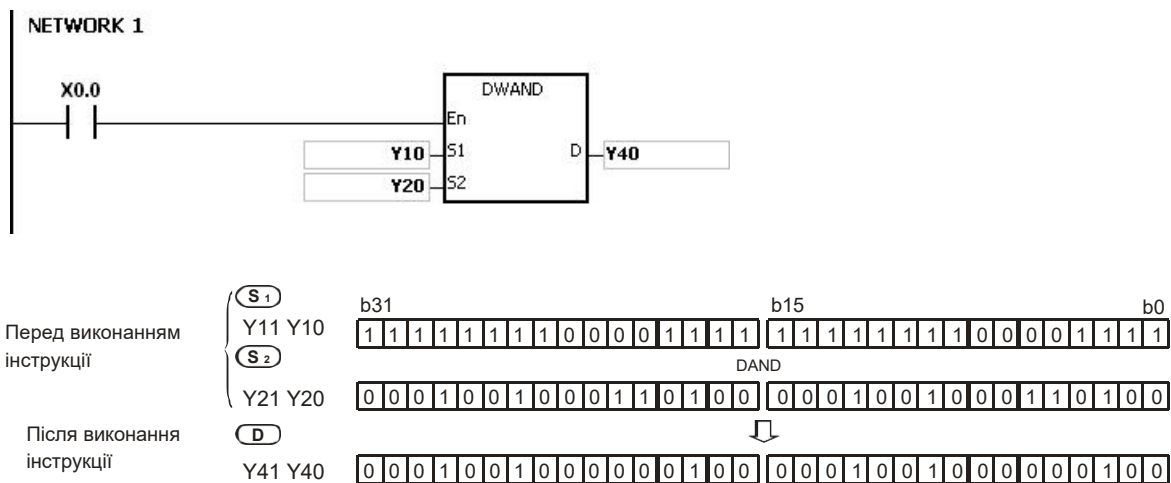
Приклад 1

Коли X0.0 увімкнено, інструкція виконує логічну операцію AND для кожної пари відповідних бітів у 16-розрядному пристрої Y0 та 16-розрядному пристрої Y2. Він зберігає результат у Y4.



Приклад 2

Коли X0.0 увімкнено, інструкція виконує логічну операцію AND для кожної пари відповідних бітів у 32-розрядному пристрої (Y11, Y10) і 32-розрядному пристрої (Y21, Y20). Він зберігає результат у (Y41, Y40).



API	Код інструкції			Операнд							функція						
0801		MAND	P	S_1, S_2, D, n							Матрична операція AND						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●							
S_2	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●				O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

MAND	MANDP
En	En
S1	S1
S2	S2
N	N

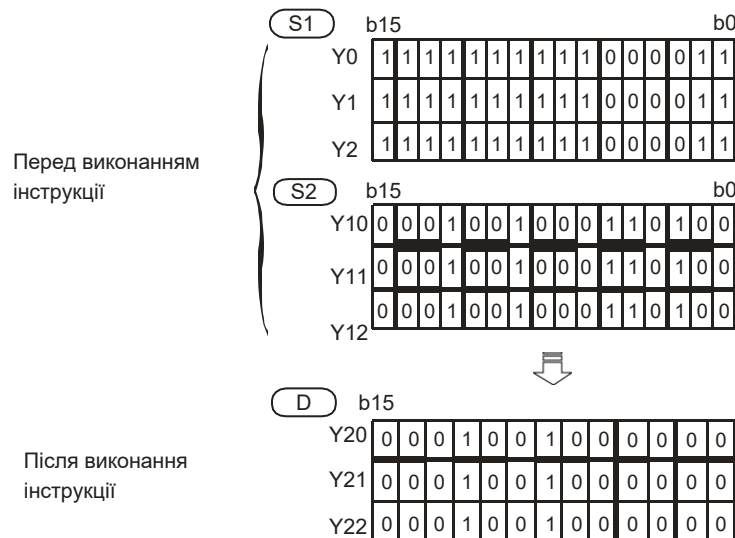
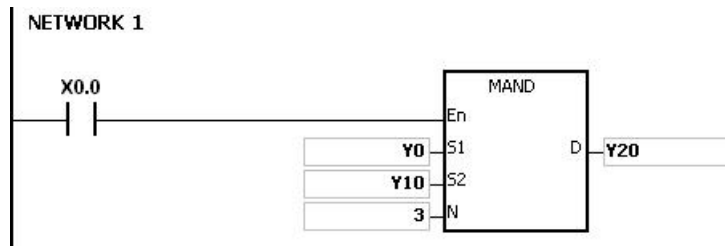
- S_1 : Джерело матриці 1
 S_2 : Джерело матриці 2
D : Результат операції
n : Довжина масиву

Пояснення

- Ця інструкція застосовує логічний оператор MAND до n рядків двійкових представлень у S_1 і n рядків двійкових представлень у S_2 . Він виконує матричну операцію AND над кожною парою відповідних бітів і зберігає результат операції в D.
- Результат у кожній позиції дорівнює 1, якщо перший біт дорівнює 1, а другий біт – 1; інакше результат дорівнює 0.
- Операнд n має бути між 1–256.

приклад

Коли X0.0 увімкнено, ця інструкція виконує матричну операцію AND для кожної пари відповідних бітів даних у 16-розрядних пристроях Y0–Y2 і даних у 16-розрядних пристроях Y10–Y12. Він зберігає результат у 16-розрядних пристроях Y20–Y22.



Додаткові зауваження

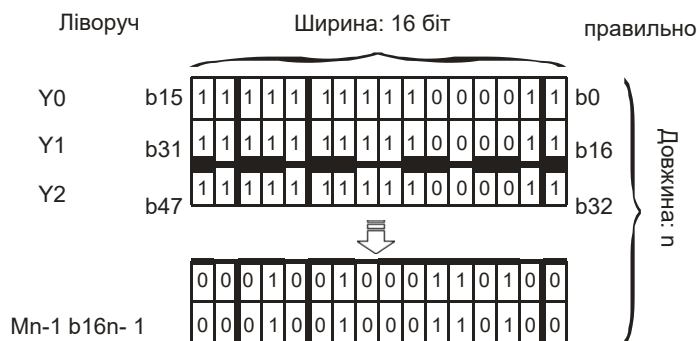
1. Якщо $S_1 + n - 1$, $S_2 + n - 1$ або $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, і помилка

код у SR0 – 16#2003.

2. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

3. Пояснення інструкцій матриці:

- Матриця складається з більш ніж одного 16-розрядного регістра. Кількість регістрів у матриці – це довжина масиву n . У матриці є $16 \times n$ бітів, і матрична операція виконується над одним бітом за раз.
- Інструкція матриці приймає $16 \times n$ бітів у матриці як рядок бітів, а не як значення. Матрична операція виконується над кожним бітом.
- Інструкції матриці в основному обробляють статус «один-до-багатьох» або «багато-до-багатьох», наприклад, переміщення, копіювання, порівняння та пошук по бітах.
- Ви повинні вказати 16-розрядний регістр для матричної інструкції. 16-розрядний регістр визначає певний біт серед $16 \times n$ розрядів у матриці для операції, і 16-розрядний регістр називається покажчиком. Значення в регістрі знаходиться між $0 - 16 \times n - 1$ і відповідає біту між $b0 - b16 \times n - 1$.
- Зміщення або обертання вказаних даних може бути залучено до операції матриці. Зауважте, що кількість бітів зменшується зліва направо, як показано нижче.



- Ширина матриці (C) становить 16 біт.
- P_r представляє показник. Коли значення P_r дорівнює 15, воно вказує на b₁₅.

Приклад: наступна матриця складається з трьох 16-розрядних пристроїв Y0, Y1 і Y2. Дані в Y0 – це 16#AAAA, дані в Y1 – 16#5555, а дані в Y2 – 16#AAFF.

b ₁₅	b ₁₄	b ₁₃	b ₁₂	b ₁₁	b ₁₀	b ₉	b ₈	b ₇	b ₆	b ₅	b ₄	b ₃	b ₂	b ₁	b ₀	
1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	Y0
0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	Y1
1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	Y2

Приклад: наступна матриця складається з трьох 16-розрядних пристроїв X0, X1 і X2. Дані в X0 є 16#37, дані в X1 – 16#68, а дані в X2 – 16#45.

b ₁₅	b ₁₄	b ₁₃	b ₁₂	b ₁₁	b ₁₀	b ₉	b ₈	b ₇	b ₆	b ₅	b ₄	b ₃	b ₂	b ₁	b ₀	
0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	X0
0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	X1
0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	X2

API	Код інструкції			Операнд							функція					
0802	D	WOR	P	S ₁ , S ₂ , D							Логічна операція OR					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		○	○	○	○		
S ₂	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●						
S ₂		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу					16-бітна інструкція					32-розрядна інструкція				
AS					AS					AS				

СИМВОЛ

WOR	WORP
En	En
S1	S1
S2	S2
D	D

DOR	DORP
En	En
S1	S1
S2	S2
D	D

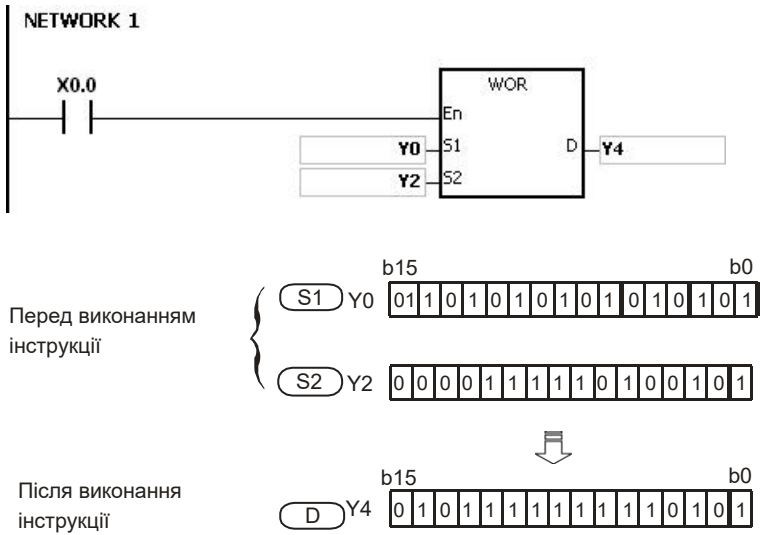
- S_1 : Джерело даних 1
- S_2 : Джерело даних 2
- D : Результат операції

Пояснення

1. Ця інструкція застосовує логічний оператор OR до двійкових представлень у S_1 і S_2 . Він виконує логічну операцію включення OR над кожною парою відповідних бітів і зберігає результат операції в D.
2. Тільки інструкція DOR може використовувати 32-бітний лічильник, але не пристрій E.
3. Результат у кожній позиції дорівнює 1, якщо перший біт дорівнює 1, другий біт дорівнює 1 або обидва біти дорівнюють 1; інакше результат дорівнює 0.

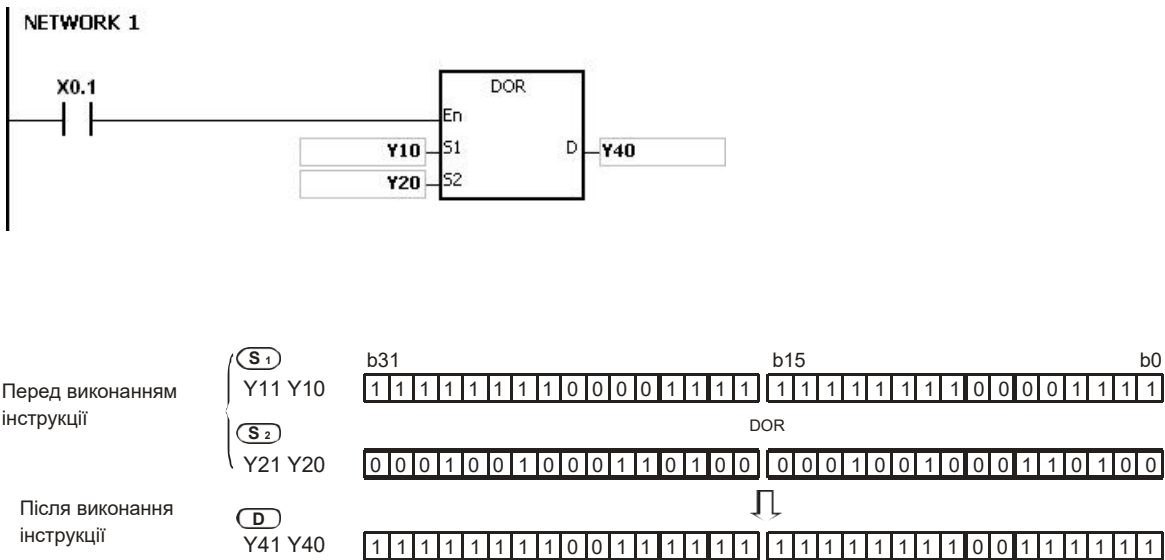
Приклад 1

Коли X0.0 увімкнено, ця інструкція виконує логічну операцію включення OR для кожної пари відповідних бітів у 16-розрядному пристрої Y0 та 16-розрядному пристрої Y2. Він зберігає результат операції в Y4.



Приклад 2

Коли X0.1 увімкнено, інструкція виконує логічну операцію включення OR для кожної пари відповідних бітів у 32-розрядний пристрій (Y11, Y10) і 32-розрядний пристрій (Y21, Y20). Він зберігає результат операції в (Y41, Y40).

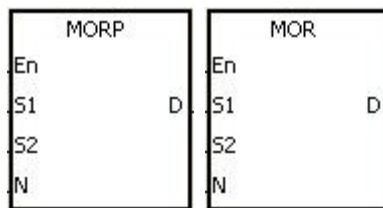


API	Код інструкції			Операнд						функція						
0803		MOR	P	S_1, S_2, D, n						Матрична операція OR						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●							
S_2	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●				O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

 S_1 : Джерело матриці 1 S_2 : Джерело матриці 2 D : Результат операції n : Довжина масиву

Пояснення

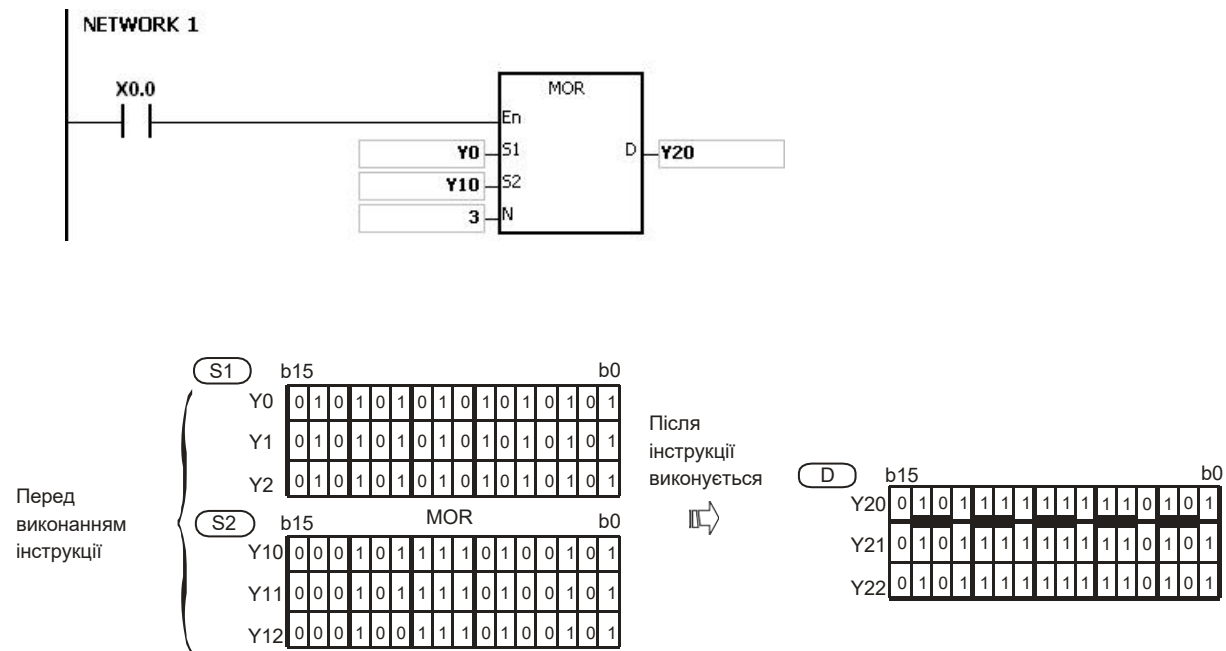
1. Ця інструкція застосовує логічний оператор OR до n рядків двійкових представлень у S_1 і n рядків двійкових представлень у S_2 . Він виконує матричну операцію OR над кожною парою відповідних бітів і зберігає результат операції в D .

2. Результат у кожній позиції дорівнює 1, якщо перший біт дорівнює 1, другий біт дорівнює 1 або обидва біти дорівнюють 1; інакше результат дорівнює 0.

3. Операнд n має бути між 1–256.

приклад

Коли X0.0 увімкнено, інструкція виконує матричну операцію OR над кожною парою відповідних бітів у 16-розрядних пристроях Y0–Y2 і даними в 16-розрядних пристроях Y10–Y12. Він зберігає результат операції в 16-розрядних пристроях Y20–Y22.



Додаткові зауваження

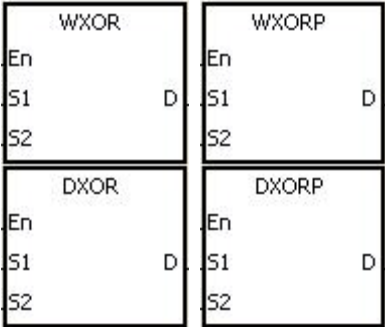
1. Якщо $S_1 + n - 1$, $S_2 + n - 1$ або $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, і помилка код у SR0 – 16#2003.
2. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд						функція							
0804	D	WXOR	P	S_1, S_2, D						Логічна операція виключаючого OR							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●	●	●	●		○	○	○	○			
S_2	●	●			●	●	●	●	●		○	○	○	○			
D		●			●	●	●	●			○	○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



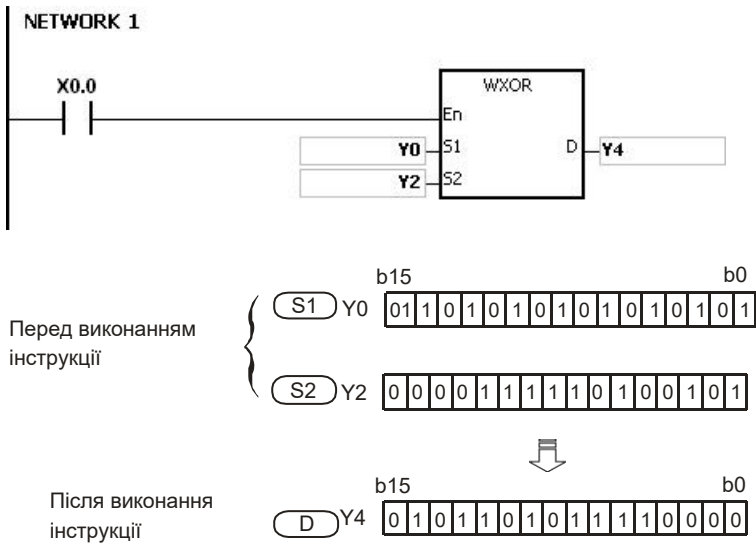
- S_1 : Джерело даних 1
- S_2 : Джерело даних 2
- D : Результат операції

Пояснення

- Ця інструкція застосовує логічний оператор XOR до двійкових представлень у S_1 та S_2 . Він виконує логічну виключаючу операцію OR над кожною парою відповідних бітів і зберігає результат операції в D.
- Тільки інструкція DXOR може використовувати 32-бітний лічильник, але не пристрій E.
- Результат у кожній позиції дорівнює 1, якщо два біти різні, і 0, якщо вони однакові.

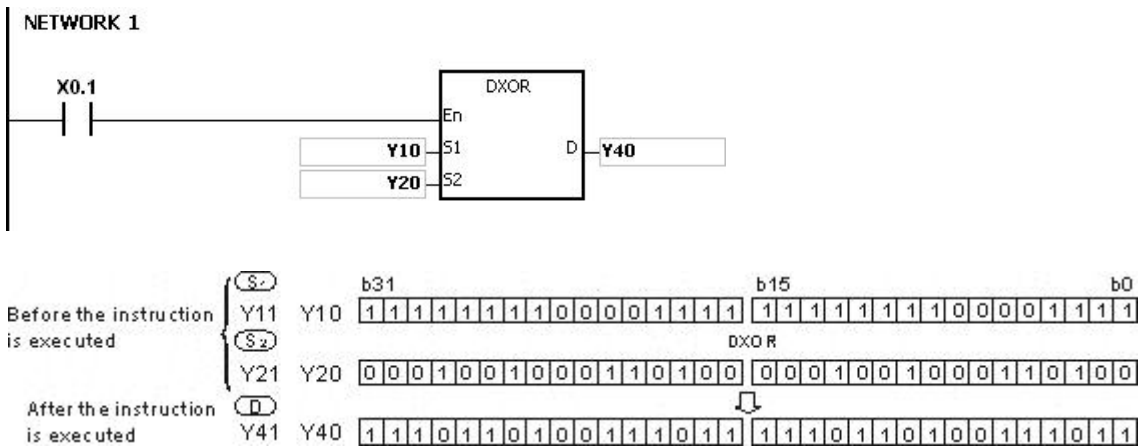
Приклад 1

Коли X0.0 увімкнено, інструкція виконує ексклюзивну операцію OR для кожної пари відповідних бітів у 16-розрядному пристрої Y0 та 16-розрядному пристрої Y2. Він зберігає результат операції в Y4.



Приклад 2

Коли X0.1 увімкнено, інструкція виконує логічну виключаючу операцію OR для кожної пари відповідних бітів у 32-розрядний пристрій (Y11, Y10) і 32-розрядний пристрій (Y21, Y20). Він зберігає результат операції в (Y41, Y40).



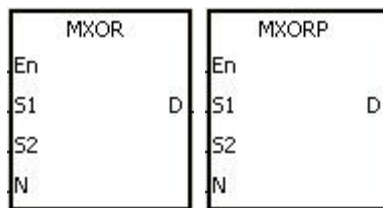
API	Код інструкції			Операнд						функція							
0805		MXOR	P	S_1, S_2, D, n						Матрична операція виняткового OR							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●							
S_2	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●				O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

 S_1 : Джерело матриці 1 S_2 : Джерело матриці 2

D : Результат операції

n : Довжина масиву

Пояснення

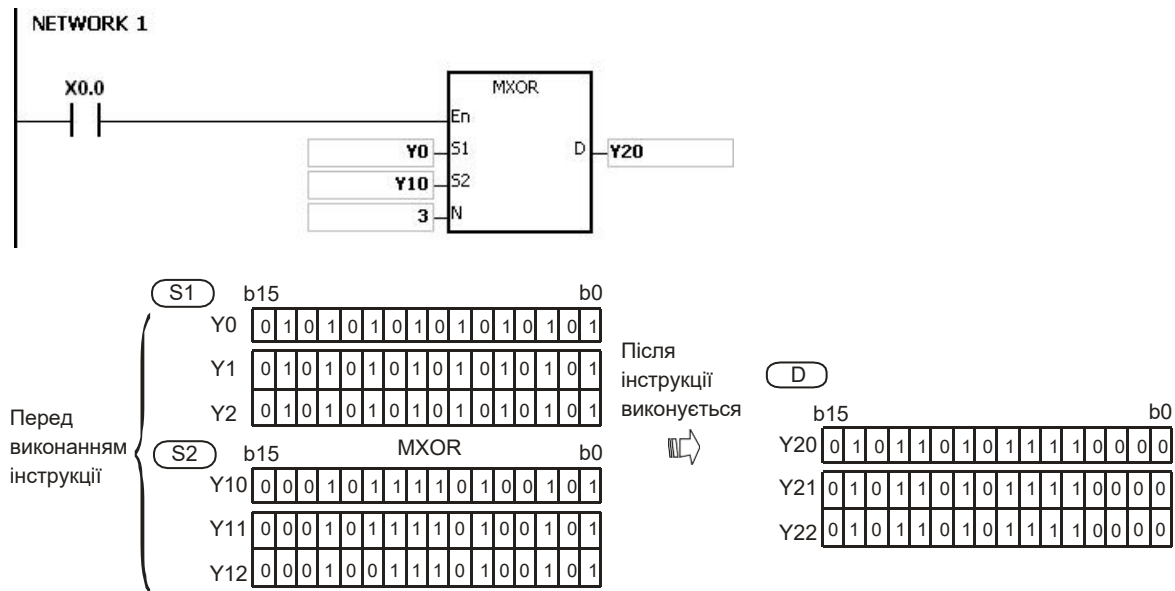
1. Ця інструкція застосовує логічний оператор XOR до n рядків двійкових представлень у S_1 і n рядків двійкових представлень у S_2 . Він виконує матричну виключаючу операцію OR над кожною парою відповідних бітів і зберігає результат операції в D.

2. Результат у кожній позиції дорівнює 1, якщо два біти різні, і 0, якщо вони однакові.

3. Операнд n має бути між 1–256.

приклад

Коли X0.0 увімкнено, інструкція виконує матричну ексклюзивну операцію OR для кожної пари відповідних бітів у 16-розрядних пристроях Y0–Y2 і даних у 16-розрядних пристроях Y10–Y12. Він зберігає результат операції в 16-розрядних пристроях Y20–Y22.



Додаткові зауваження

6

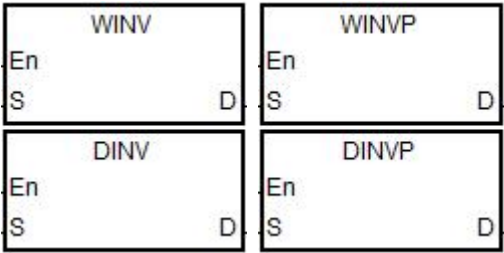
1. Якщо $S_1 + n - 1$, $S_2 + n - 1$ або $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції				Операнд						функція					
0808	D	WINV	P	S, D						Логічна зворотна операція INV						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O	O	O	O		
D	●	●			●	●	●	●			O	O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу						16-бітна інструкція				32-розрядна інструкція			
AS						AS				AS			

СИМВОЛ



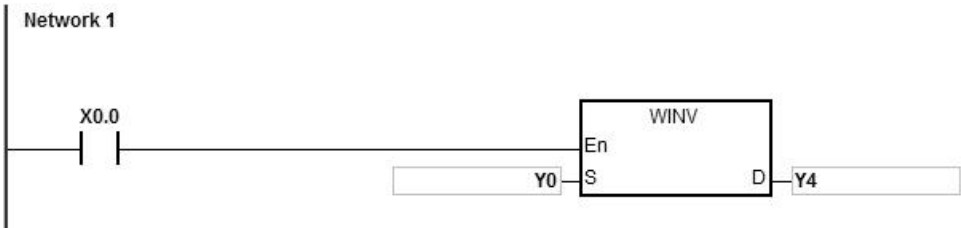
S : Джерело даних
D : Результат операції

Пояснення

- 6
- Ця інструкція застосовує інструкцію INV до даних у **S** та зберігає результат операції в **D**.
 - Тільки інструкція DINV може використовувати 32-бітний лічильник, але не пристрій E.
 - Ця інструкція виконує зворотну обробку даних у S. Якщо стан S дорівнює 0 перед виконанням INV стан змінюється на 1 у результаті виконання інструкції INV.

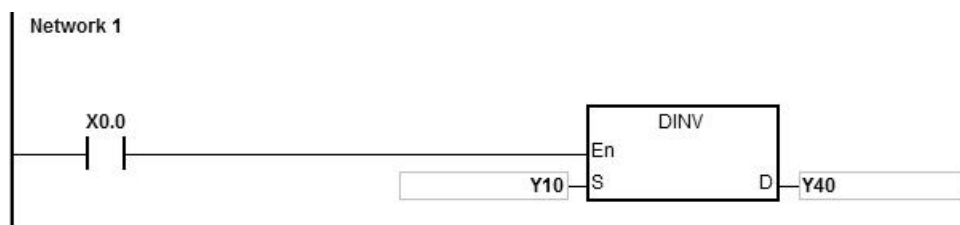
Приклад 1

Коли X0.0 увімкнено, інструкція виконує операцію INV на відповідних бітах у 16-розрядному пристрої Y0. Він зберігає результат операції в 16-бітному пристрої Y4.



Приклад 2

Коли X0.0 увімкнено, інструкція виконує операцію INV для кожної пари відповідних бітів у 32-розрядних пристроях Y11–Y10. Він зберігає результат операції в 32-розрядному пристрої Y41–Y40.

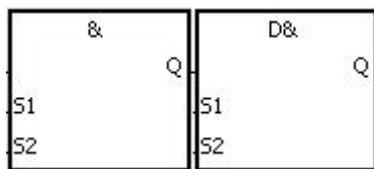


API		Код інструкції			Операнд						функція						
0809-0811		D	LD #		S ₁ , S ₂						Контактний тип логічної операції LD #						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●	●	●	●		○	○	○	○			
S ₂	●	●			●	●	●	●	●		○	○	○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●				●	●	
S ₂		●	●			●	●				●	●	

Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
-		AS	AS

СИМВОЛ

 S_1 : Джерело даних 1 S_2 : Джерело даних 2

Наприклад, LD& та DLD&

Пояснення

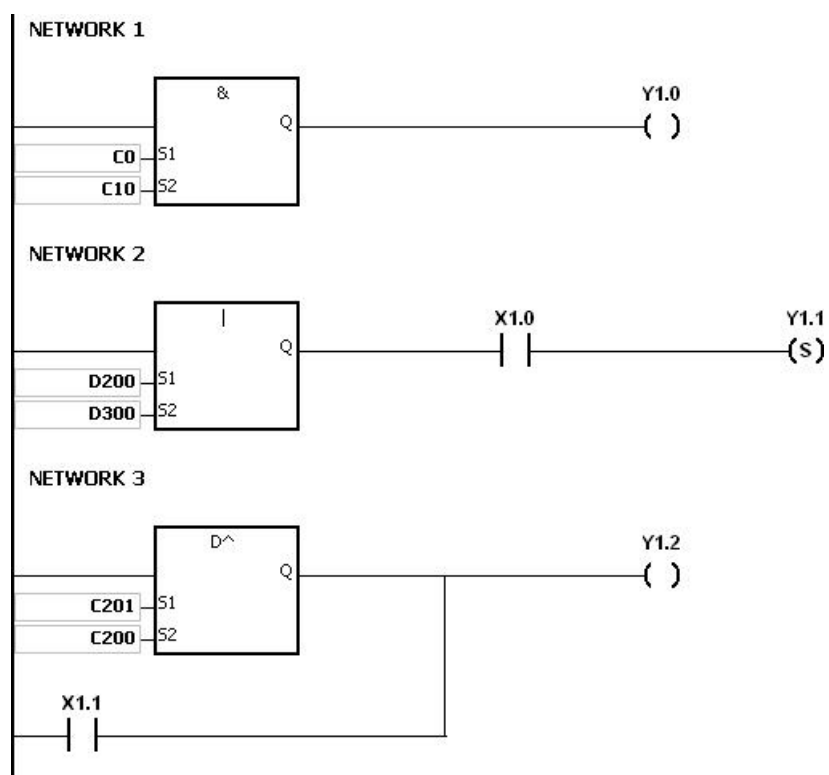
- Інструкція використовується для порівняння даних у S_1 з даними в S_2 . Якщо результат порівняння не дорівнює 0, умова інструкції виконано. Коли результат порівняння дорівнює 0, умова інструкції не виконується.
- Тільки інструкція DLD # може використовувати 32-бітний лічильник, але не пристрій E.
- Інструкцію LD # можна підключити безпосередньо до материнської лінії.

Номер API	16-бітна інструкція	32-розрядна інструкція	Результат операції порівняння	
			ON	OFF
0809	LD&	D LD&	$S_1 \& S_2 \neq 0$	$S_1 \& S_2 = 0$
0810	LD	D LD	$S_1 S_2 \neq 0$	$S_1 S_2 = 0$
0811	LD^	D LD^	$S_1 \wedge S_2 \neq 0$	$S_1 \wedge S_2 = 0$

- &: Логічна операція I
- |: логічна операція OR
- ^: логічна операція виключаючого OR

приклад

1. Логічний оператор AND приймає дані в C0 і C1 і виконує логічну операцію AND над кожною парою відповідних бітів. Якщо результат операції не дорівнює 0, Y1.0 увімкнено.
2. Інструкція виконує логічну операцію OR над кожною парою відповідних бітів у D200 і D300, коли результат операції не дорівнює 0 і X1.0 увімкнено, Y1.1 увімкнено.
3. Інструкція виконує логічну виключаючу операцію XOR над кожною парою відповідних бітів у C201 і C200, коли результат операції не дорівнює 0 або коли X1.1 має значення ON, Y1.2 має значення ON.



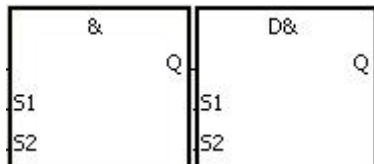
Додаткові зауваження

Якщо S_1 або S_2 недійсні, умова інструкції не виконується, SM0 увімкнено, а помилка в SR0 — 16#2003.

API	Код інструкції			Операнд							функція					
0812-0814	D	I #		S ₁ , S ₂							Контактний тип логічної операції I #					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		○	○	○	○		
S ₂	●	●			●	●	●	●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	
Інструкція пульсу		16-бітна інструкція				32-розрядна інструкція							
-		AS				AS							

СИМВОЛ

 S_1 : Джерело даних 1 S_2 : Джерело даних 2

Наприклад, AND& та DAND&

Пояснення

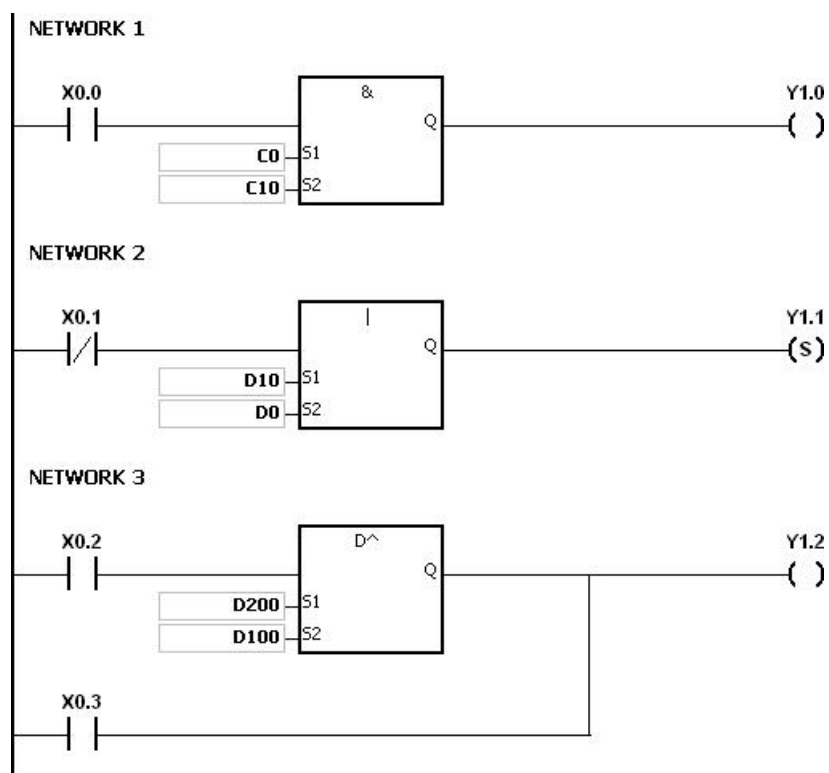
- Ця інструкція порівнює дані в S_1 з даними в S_2 . Коли результат порівняння не дорівнює 0, умова інструкції виконується. Коли результат порівняння дорівнює 0, умова інструкції не виконується.
- Тільки інструкція DAND # може використовувати 32-бітний лічильник, але не пристрій E.
- З'єднайте інструкцію I # і контакт послідовно.

Номер API	16-бітна інструкція	32-розрядна інструкція	Результат операції порівняння	
			ON	OFF
0812	I&	D I&	$S_1 \& S_2 \neq 0$	$S_1 \& S_2 = 0$
0813	I	D I	$S_1 S_2 \neq 0$	$S_1 S_2 = 0$
0814	I^	D I^	$S_1 \wedge S_2 \neq 0$	$S_1 \wedge S_2 = 0$

- &: Логічна операція I
- |: логічна операція OR
- ^: логічна операція виключаючого OR

приклад

1. Коли X0.0 увімкнено, інструкція виконує логічну операцію I для кожної пари відповідних бітів у C0 та C10. Якщо результат операції не дорівнює 0, Y1.0 увімкнено.
2. Коли X0.1 вимкнено, інструкція виконує логічну операцію OR над кожною парою відповідних бітів у D10 і D0. Якщо результат операції не дорівнює 0, Y1.1 увімкнено.
3. Коли X0.2 увімкнено, інструкція виконує логічну виключаючу операцію OR для кожної пари відповідних бітів у 32-бітному регістрі (D200, D201) і даних у 32-бітному регістрі (D100, D101). Якщо результат операції не дорівнює 0, або коли X0.3 увімкнено, Y1.2 увімкнено.



Додаткові зауваження

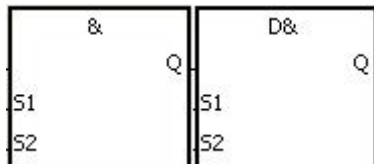
Якщо значення в S_1 або S_2 недійсне, умова інструкції не виконується, SM0 увімкнено, а помилка в SR0 — 16#2003.

API	Код інструкції			Операнд						функція						
0815-0817	D	OR #		S ₁ , S ₂						Контактний тип логічної операції OR #						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		○	○	○	○		
S ₂	●	●			●	●	●	●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●				●	●	
S_2		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ

 S_1 : Джерело даних 1 S_2 : Джерело даних 2

Наприклад, OR& та DOR&

Пояснення

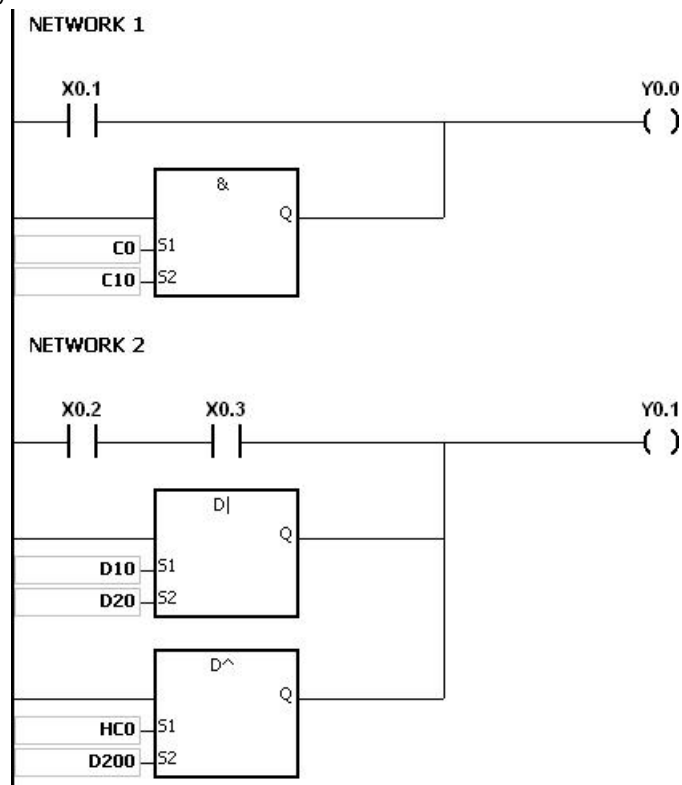
- Ця інструкція порівнює дані в S_1 з даними в S_2 . Якщо результат порівняння не дорівнює 0, умова інструкція виконана. Коли результат порівняння дорівнює 0, умова інструкції не виконується.
- Тільки інструкція DOR # може використовувати 32-бітний лічильник.
- З'єднайте інструкцію OR # і контакт паралельно.

Номер API	16-бітна інструкція	32-розрядна інструкція	Результат операції порівняння	
			ON	OFF
0815	OR&	D OR&	$S_1 \& S_2 \neq 0$	$S_1 \& S_2 = 0$
0816	OR	D OR	$S_1 S_2 \neq 0$	$S_1 S_2 = 0$
0817	OR^	D OR^	$S_1 \wedge S_2 \neq 0$	$S_1 \wedge S_2 = 0$

- &: Логічна операція І
- |: логічна операція OR
- ^: логічна операція виключаючого OR

приклад

1. Коли X0.1 увімкнено, Y0.0 увімкнено. Інструкція виконує логічну операцію I для кожної пари відповідних бітів у C0 і C10. Якщо результат операції не дорівнює 0, Y0.0 увімкнено.
2. Коли X0.2 і X0.3 увімкнено, Y0.1 увімкнено. Інструкція виконує логічну операцію OR над кожною парою відповідних бітів у 32-розрядному регістрі (D10, D11) і 32-розрядному регістрі (D20, D21). Якщо результат операції не дорівнює 0, Y0.1 увімкнено. Інструкція виконує логічну виключаючу операцію OR над кожною парою відповідних бітів у 32-розрядному лічильнику HC0 і 32-розрядний регістр (D200, D201). Якщо результат операції не дорівнює 0, Y0.1 увімкнено.



Додаткові зауваження

Якщо значення в S_1 або S_2 недійсне, умова інструкції не виконується, SM0 увімкнено, а помилка в SR0 — 16#2003.

6.10 Інструкції з ротації

6.10.1 Список інструкцій щодо ротації

У наведеній нижче таблиці наведено інструкції з ротації, які розглядаються в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16- бітний	32- розрядний		
0900	ROR	DROR	✓	Обертання бітів у групі вправо
0901	RCR	DRCR	✓	Обертання бітів у групі вправо з прапором перенесення
0902	ROL	DROL	✓	Обертання бітів у групі вліво
0903	RCL	DRCL	✓	Обертання бітів у групі вліво з прапором перенесення
0904	MBR	—	✓	Обертання бітів вправо або вліво в матриці

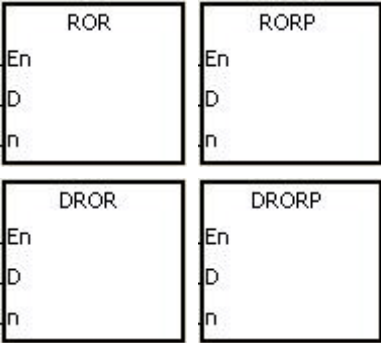
6.10.2 Пояснення інструкцій щодо ротації

API	Код інструкції				Операнд							функція					
0900	D	ROR		P	D, n							Обертання бітів у групі вправо					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			○	○					
n	●	●			●	●	●	●	●		○	○	○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



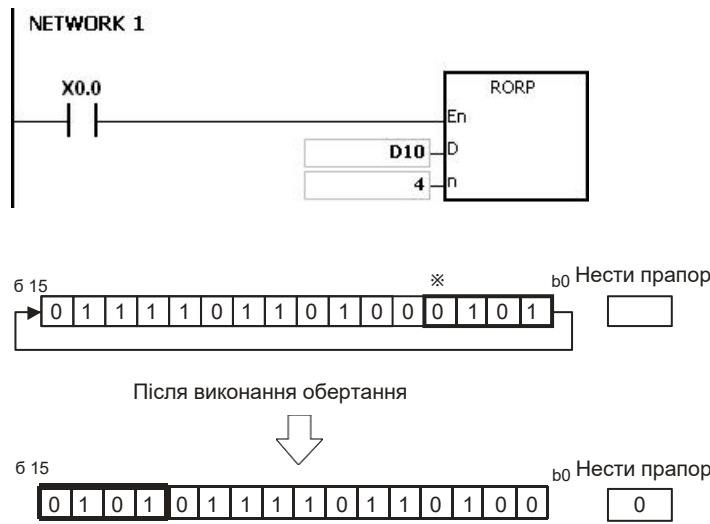
D : Пристрій для обертання
n : Кількість бітів у групі

Пояснення

- Ця інструкція ділить біти в пристрої, визначеному **D** , на групи (**n** бітів у групі), а потім повертає ці групи вправо без прапора переносу.
- Тільки інструкція DROR може використовувати 32-розрядний лічильник, але не пристрій E.
- Для 16-бітної інструкції використовуване значення **n** має бути між 1–6. Для 32-розрядної інструкції значення **n** має бути між 1–32. Якщо **n** менше 0, інструкція не виконується. Коли **n** перевищує діапазон, інструкція виконується з **n** при максимальному значенні (32) діапазону.
- Загалом використовуються імпульсні інструкції RORP та DRORP.

приклад

Коли X0.0 перемикається з OFF на ON, інструкція ділить значення бітів у D10 на групи (по чотири біти в групі) і повертає ці групи вправо. Значення біта, позначеного ※, передається на прапор переносу SM602.



Додаткові зауваження

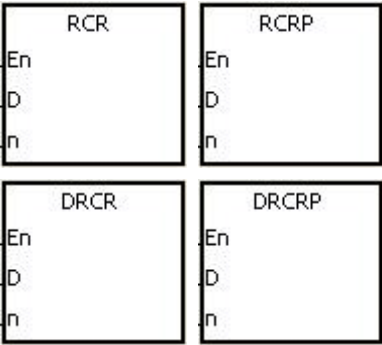
- 1. Якщо пристрій виходить за межі діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції				Операнд						функція						
0901	D	RCR		P	D, n						Обертання бітів у групі вправо з прапором перенесення						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			O	O					
n	●	●			●	●	●	●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



D : Пристрій для обертання

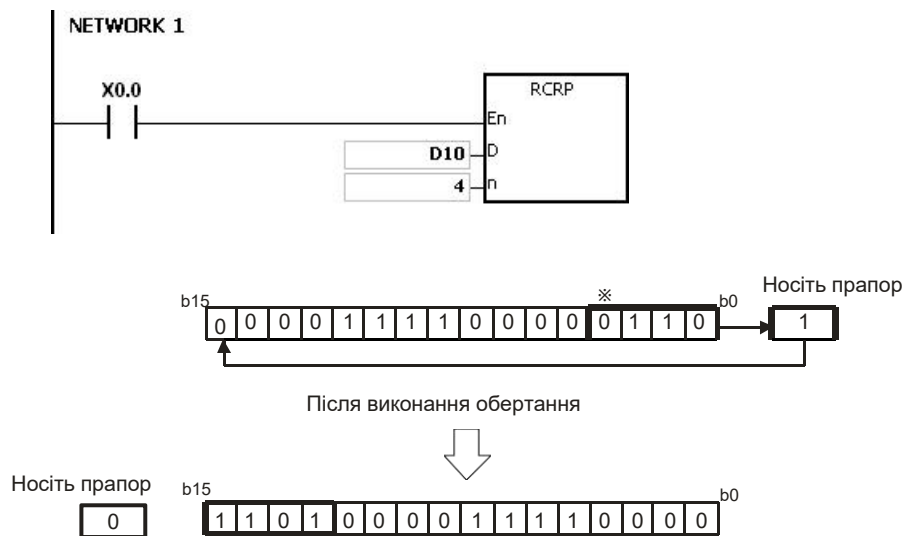
n : Кількість бітів у групі

Пояснення

- Ця інструкція розділяє біти в пристрої, визначеному **D** , на групи (**n** бітів у групі), а потім повертає ці групи вправо за допомогою прапора переносу SM602.
- Тільки інструкція DRCR може використовувати 32-бітний лічильник, але не пристрій E.
- Для 16-бітних інструкцій використовуване значення **n** має бути між 1–16. Для 32-розрядних інструкцій значення **n** має бути між 1–32. Якщо **n** менше 0, інструкція не виконується. Коли **n** перевищує діапазон, інструкція виконується з **n** при максимальному значенні (32) діапазону.
- Загалом використовуються імпульсні інструкції RCRP і DRCRP.

приклад

Коли X0.0 перемикається з OFF на ON, інструкція ділить значення бітів у D10 на групи (чотири біти як група), а потім обертає ці групи вправо за допомогою прапора переносу SM602. Значення біта, позначеного ※, передається на прапор переносу SM602.



Додаткові зауваження

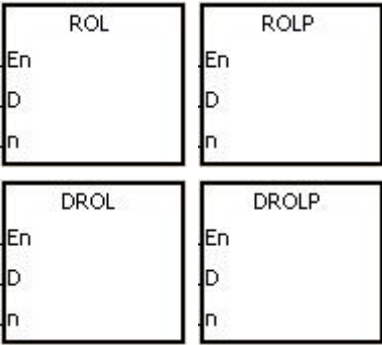
- 1. Якщо пристрій виходить за межі діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції				Операнд						функція						
0902	D	ROL	P		D, n						Обертання бітів у групі вліво						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			O	O					
n	●	●			●	●	●	●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



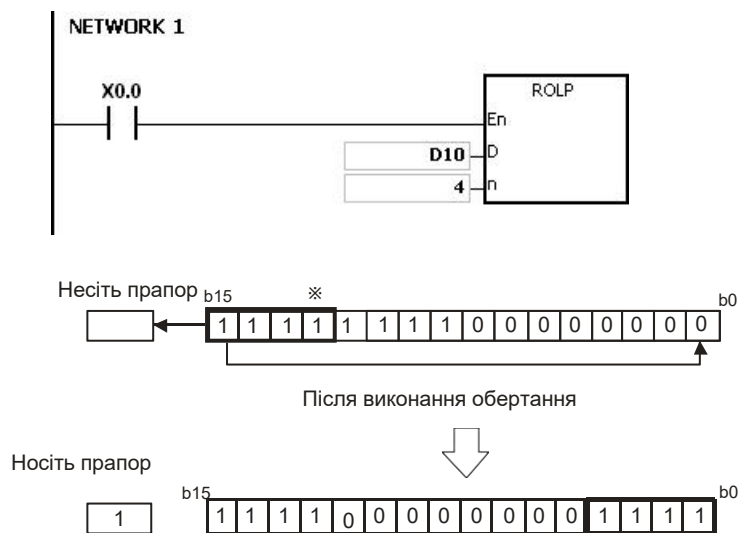
D : Пристрій для обертання
n : Кількість бітів у групі

Пояснення

- Ця інструкція ділить біти в пристрої, визначеному **D** , на групи (**n** бітів у групі), а потім повертає ці групи вліво.
- Тільки інструкція DROL може використовувати 32-розрядний лічильник, але не пристрій E.
- Для 16-бітних інструкцій значення **n** має бути між 1–16. Для 32-розрядних інструкцій значення **n** має бути між 1–32. Якщо **n** менше 0, інструкція не виконується. Коли **n** перевищує діапазон, інструкція виконується з **n** при максимальному значенні (32) діапазону.
- Загалом використовуються імпульсні інструкції ROLP і DROLP.

приклад

Коли X0.0 перемикається з OFF на ON, інструкція ділить значення бітів у D10 на групи (по чотири біти як група), а потім повертає ці групи вліво. Значення біта, позначеного *, передається на прапор переносу SM602.



Додаткові зауваження

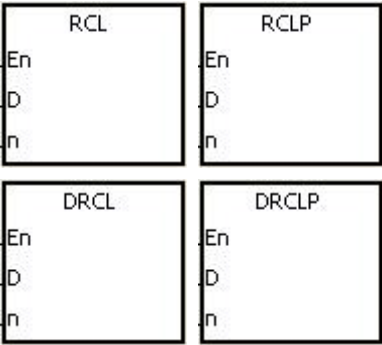
1. Якщо пристрій виходить за межі діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції			Операнд							функція						
0903	D	RCL	P	D, n							Обертання бітів у групі вліво з прапором перенесення						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			O	O					
n	●	●			●	●	●	●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



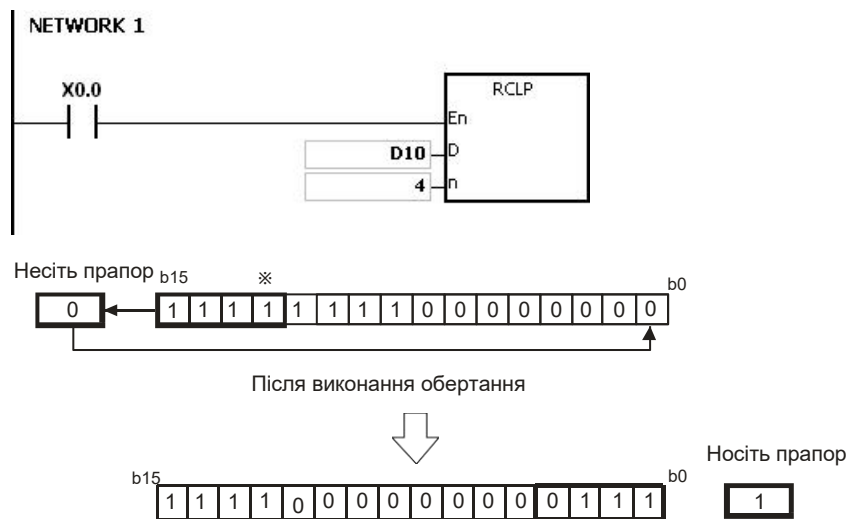
D : Пристрій для обертання
n : Кількість бітів у групі

Пояснення

- Ця інструкція розділяє біти в пристрої, визначеному **D** , на групи (**n** бітів у групі), а потім повертає ці групи вліво за допомогою прапора переносу SM602.
- Тільки інструкція DRCL може використовувати 32-розрядний лічильник, але не пристрій E.
- Для 16-бітних інструкцій значення в **n** має бути між 1–16. Для 32-розрядних інструкцій значення **n** має бути між 1–32. Якщо **n** менше 0, інструкція не виконується. Коли **n** перевищує діапазон, інструкція виконується з **n** при максимальному значенні (32) діапазону.
- Загалом використовуються імпульсні інструкції RCLP і DRCLP.

приклад

Коли X0.0 перемикається з OFF на ON, інструкція ділить значення бітів у D10 на групи (по чотири біти як група), а потім повертає ці групи вліво за допомогою прапора переносу SM602. Значення біта, позначеного ※, передається на прапор переносу SM602.



Додаткові зауваження

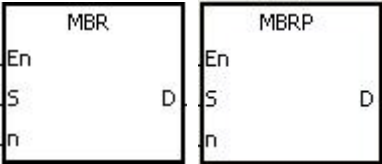
- 1. Якщо пристрій виходить за межі діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції				Операнд							функція					
0904		MBR	P		S, D, n							Обертання бітів у групі праворуч або ліворуч у матриці					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●	●								
D		●			●	●		●									
n	●	●			●	●		●	●				O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



- S : Джерело матриці
- D : Результат операції
- n : Довжина масиву

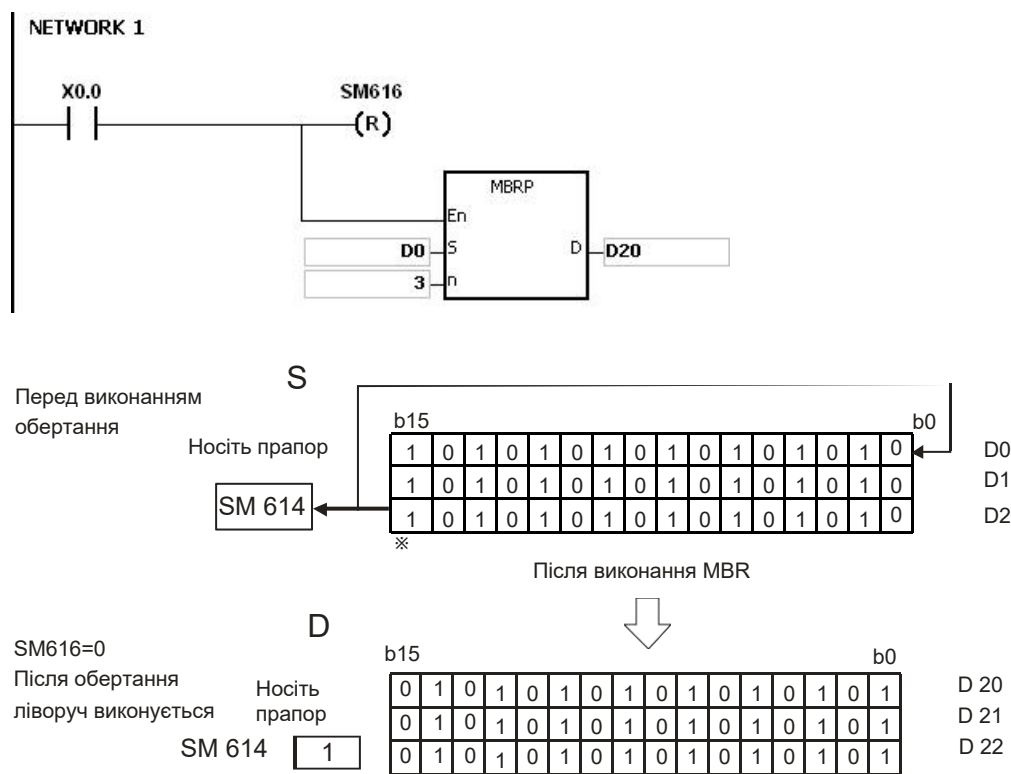
Пояснення

6

- Ця інструкція повертає значення **n** рядків бітів у **S** вправо або вліво. Коли SM616 вимкнено, інструкція повертає значення бітів вліво. Коли SM616 увімкнено, інструкція обертає значення бітів праворуч. Інструкція заповнює вакансію, що утворилася в результаті обертання, значенням біта, повернутого останнім, і зберігає результат операції в **D** . Значення біта, повернутого останнім, не тільки заповнює вакансію, але також передається на прапор переносу SM614.
- Для 16-бітних інструкцій значення **n** має бути між 1–16. Для 32-розрядних інструкцій значення **n** має бути між 1–32. Якщо **n** менше 0, інструкція не виконується. Коли **n** перевищує діапазон, інструкція виконується з **n** при максимальному значенні (32) діапазону.
- Загалом використовується імпульсна інструкція MBRP.

приклад 1:

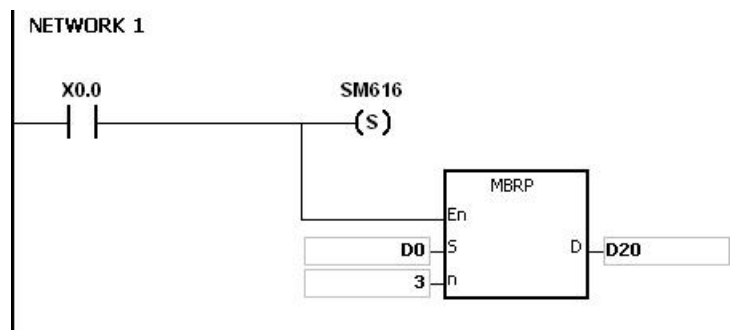
Коли X0.0 увімкнено, а SM616 вимкнено, інструкція обертає значення бітів у 16-розрядних регістрах D0–D2 вліво та зберігає результат операції в 16-розрядних регістрах D20–D22. Значення біта, позначеного *, передається на прапор переносу SM614.

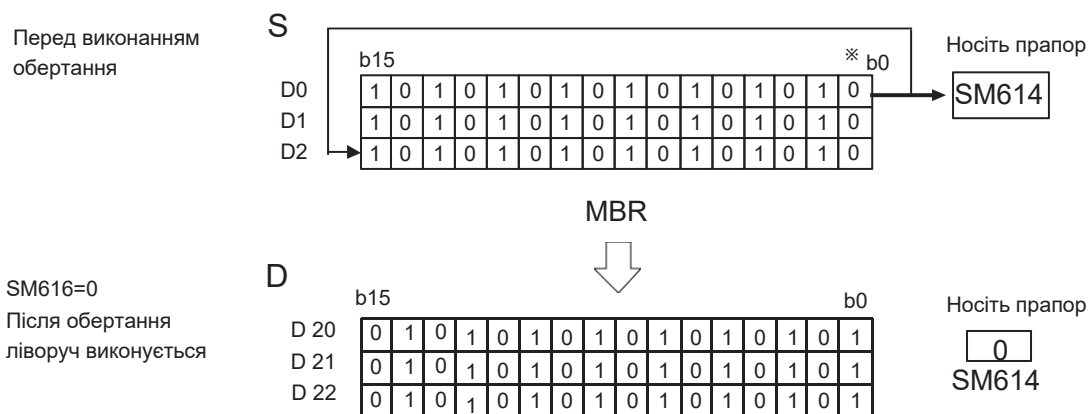


6

приклад 2:

Коли X0.0 увімкнено, а SM616 увімкнено, інструкція повертає значення бітів у 16-розрядних регістрах D0–D2 праворуч і зберігає результат операції в 16-розрядних регістрах D20–D22. Значення біта, позначеного *, передається на прапор переносу SM614.





Додаткові зауваження

1. Якщо **S + n - 1** або **D + n - 1** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

2. Прапори інструкцій:

SM614: Прапор переносу для повороту/зсуву/виведення матриці

SM616: Прапор напрямку для повороту/зсуву матриці

6.1 Інструкції з таймера та лічильника

6.11.1 Список інструкцій таймера та лічильника

У наведеній нижче таблиці наведено інструкції щодо таймера та лічильника, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1000	RST	DRST	–	Скидання контакту до ВІМК. або очищення значення в реєстрі.
1001	TMR	–	–	16-бітний таймер (одиниця: 100 мс)
1002	TMRH	–	–	16-бітний таймер (одиниця: 1 мс)
1003	CNT	–	–	16-розрядний лічильник
1004	–	DCNT	–	32-розрядний лічильник (включаючи використання високошвидкісних лічильників)
1005	–	DHSCS	–	Налаштування високошвидкісного порівняння
1006	–	DHSCR	–	Скидання високошвидкісного порівняння
1007	–	DHSZ	–	Швидке порівняння зон введення
1008	–	DSPD	–	Виявлення швидкості
1009	PWD	–	–	Визначення ширини імпульсу
1010	–	DCAP	–	Захоплення високошвидкісного значення підрахунку на зовнішньому вході переривати
1011	TMRM	–	–	16-бітний таймер (одиниця: 10 мс)
1012	IETS	–	✓	Початок вимірювання часу виконання інструкції
1013	IETE	–	✓	Кінець вимірювання часу виконання інструкції
1014	–	DHSCY	–	Циклічний підрахунок, порівняння та виведення
1015	PPDT	–	–	Виявлення різниці в часі між двома фазами
1016	ATMR	–	–	16-бітний контактний таймер (одиниця: 100 мс)
1017	ATMRH	–	–	16-бітний контактний таймер (одиниця: 1 мс)
1018	ATMRM	–	–	16-бітний контактний таймер (одиниця: 10 мс)

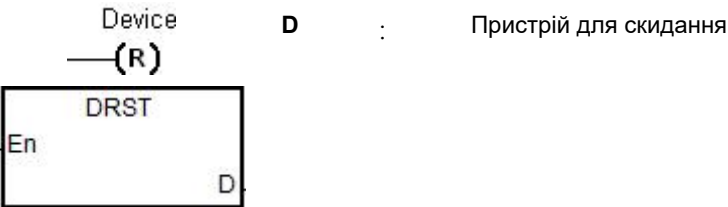
6.11.2 Пояснення інструкцій таймера та лічильника

API	Код інструкції				Операнд						функція						
1000	D	RST			D						Скидання контакту або очищення реєстру						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		O	O	O	O	O	O	O		O	O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●	●	●			●	●		●		●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ



Пояснення

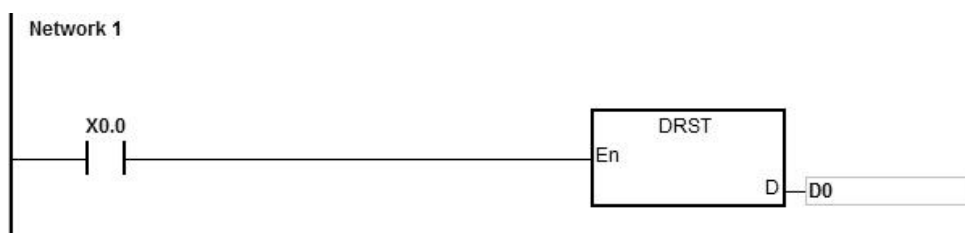
1. Ця інструкція очищає значення в 32-розрядному пристрої HC або двох послідовних 16-розрядних пристроях D. Для інших пристроїв використовуйте інструкція RST для очищення значень.
2. У наступній таблиці показано дії інструкції RST.
- | пристрій | Держава |
|-------------------|--|
| біт | Встановлює котушку та контакт у положення ВІМК. |
| T , S | Встановить поточне значення таймера та значення лічильника на 0, а також установить котушку та контакт у положення ВІМК. |
| Слово | Очищає значення 16-бітного вмісту до 0. |
| DWord , HC , Real | Очищає значення 32-розрядного вмісту, включаючи числа з плаваючою комою, до 0. |
3. Якщо інструкція RST не виконується, стан пристрою, визначений D , не змінюється.
4. Інструкція підтримує прямий вихід.

приклад

Коли X0.0 ON, інструкція встановлює Y0.5 на OFF.



Ця інструкція очищає 32-розрядні D1 і D0 до нуля, коли X0.0 увімкнено.



API	Код інструкції				Операнд								функція				
1001		TMR			S_1, S_2								16-бітний таймер (100 мс)				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1					O												
S_2								O				O	O	O			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1											●						
S_2		●				●											
								Інструкція пульсу				16-бітна інструкція				32-розрядна інструкція	
								-				AS				-	

СИМВОЛ



S_1 : Номер таймера
 S_2 : Установка значения таймера

Пояснення

Зверніться до пояснення інструкції TMRH (API 1002) для отримання детальної інформації.

API	Код інструкції				Операнд								функція				
1002		TMRH			S_1, S_2								16-бітний таймер (1 мс)				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1					O												
S_2								O				O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1											●		
S_2		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

	TMRH
En	
S1	
S2	

- S_1 : Номер таймера
- S_2 : Установка значення для таймера

Пояснення

- КоMANDa TMR використовує 100 мс як одиницю часу в таймері, тоді як інструкція TMRH використовує 1 мс як одиницю синхронізації в таймері.
- Значення S_2 для інструкцій TMR і TMRH знаходиться в межах 0–32767.
- Якщо ви повторно використовуєте той самий таймер у програмі, включно з використанням його в різних інструкціях TMR і TMRH, таймер, який першим завершить вимірювання, буде єдиним, що зраховується.
- Таймер T автоматично скидається на нуль, коли умовний контакт змінюється з УВІМК. на ВІМК.
- Коли ви додаєте букву S перед пристроєм T, таймер в інструкції TMR є накопичувальним таймером. Коли умовний контакт вимкнено, значення накопичувального таймера не очищається. Коли умовний контакт увімкнено, таймер відраховує поточне значення. Використовуйте інструкцію RST із накопичувальним таймером ST, якщо потрібно очистити значення таймера.
- Якщо ви використовуєте той самий таймер T у програмі, він буде OFF, коли один із умовних контактів OFF.
- Якщо ви використовуєте той самий таймер T для T і ST у програмі, T буде OFF, коли один із умовних контактів OFF.
- Коли виконується інструкція TMR, указана котушка таймера вмикається, і таймер починає відлік. Якщо значення таймера відповідає встановленому значенню, контакт увімкнено.

9. Таймери T0–T411 визначені як загальні таймери, а T412–T511 є таймерами підпрограм за замовчуванням. Скористайтесь програмним забезпеченням конфігурації обладнання HWCONFIG, якщо вам потрібно змінити діапазони двох типів таймерів.

10. Загальні таймери порівнюють значення часу під час сканування інструкції TMR. Система застосовує таймер до умови кожного разу, коли сканується статус інструкції TMR.

Для таймерів підпрограми система підраховує час і порівнює значення часу після виконання інструкції END. Використовуйте таймери підпрограм, коли інструкція TMR виконується не під час кожного сканування, але вам потрібні триваліші часи та порівняння.

Приклад 1 (з використанням загального типу лічильника; рахується для часового вікна 100 мс)

Коли X0.0 увімкнено, інструкція завантажує значення налаштування 50 до таймера T0, і T0 відраховує від 0 до 50. Коли значення T0 збігається з 50, контакт T0 увімкнено.



Приклад 2 (з використанням накопичувального типу лічильника; підрахунок для часового вікна 100 мс)

Коли X0.0 увімкнено, інструкція завантажує значення налаштування 50 до таймера ST0. Коли значення T0 дорівнює 25 і X0.0 перемикається з ВИМК. на УВИМК., потім T0 підраховує від 25 до 50, а контакт T0 увімкне (накопичується).



Приклад 3 (з використанням загального типу лічильника; рахується для вікна часу 1 мс)

Коли X0.0 увімкнено, інструкція завантажує значення налаштування 1000 до таймера T5, і T5 підраховує від 0 до 1000, а контакт T5 увімкнено.



Приклад 4 (з використанням накопичувального типу лічильника; підрахунок для часового вікна 1 мс)

Коли X0.0 увімкнено, інструкція завантажує значення налаштування 1000 до таймера T5. Коли значення T5 становить 500 і X0.0 перемикається з ВІМК. на УВІМК., T5 підраховує від 500 до 1000, а контакт T5 увімкнено (накопичується).



Приклад 5 (використання функціонального блоку в інструкції TMRH)

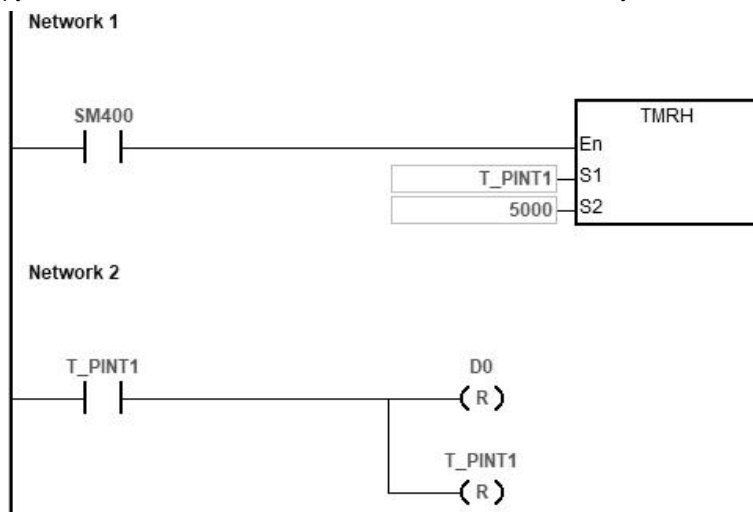
Крок 1: Оголошіть функціональний блок як T_Pointer.

Modify Symbol

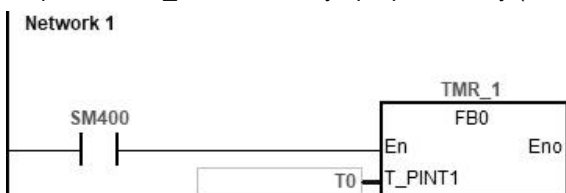
Identifier	Address	Type	Initial Value	Comment
T_PINT1		T_POINTER		

Class: VAR_IN_OUT ☒ Auto-close Dialog ☐ Define global Main Table

Цей функціональний блок очищає значення в D0 кожні 5 секунд.



Крок 2: Призначте T_Pointer як T0 у програмі блоку (POU).

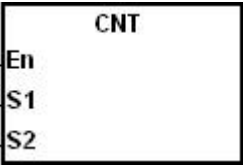


Додаткові зауваження

Коли ви оголошуєте операнд **S1** в ISPSOft, виберіть тип даних TIMER для загального T-таймера. Для накопичувального таймера ST вкажіть пристрій ST.

API	Код інструкції				Операнд							функція						
1003		CNT				S ₁ , S ₂							16-розрядний лічильник					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S ₁						O												
S ₂								O				O	O	O				
Тип даних	BOOL	WORD		DWORD		LWORD		UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING		
S ₁															●			
S ₂		●						●										
Інструкція пульсу											16-бітна інструкція			32-розрядна інструкція				
-											AS			-				

СИМВОЛ



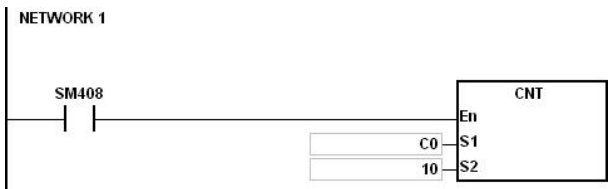
S₁ : Номер лічильника
S₂ : Установка значення для лічильника

Пояснення

1. Ця інструкція змінює вказану котушку лічильника з ВІМК. на УВІМК., а потім збільшує значення лічильника на 1. Коли значення лічильника відповідає встановленому значенню, контакт лічильника вмикається.
2. Коли значення лічильника збігається зі значенням налаштування, інструкція не змінює стан контакту та значення лічильника, якщо надходять додаткові імпульси підрахунку. Використовуйте інструкцію RST (API 1000), щоб скинути лічильник і знову ввімкнути підрахунок.
3. Зверніться до Прикладу 1 нижче, щоб продемонструвати використання інструкції CNT у програмі блоку (POU). Зверніться до прикладу 2 нижче для демонстрації використання інструкції CNT у функціональному блоці (FB).
4. Ви можете використовувати той самий номер пристрою C інструкції CNT, який використовувався в POU більше одного разу. Система буде рахувати відповідно до різного стану контакту, окремо OFF або ON.
5. Ви НЕ можете використовувати номер інструкції CNT того самого пристрою C, який використовувався у FB, більше одного разу. Рекомендується використовувати лише 1 номер пристрою C для FW V1.07.00 (за винятком) або попередніх версій. Для FW V1.07 або пізнішої версії ви можете використовувати до 8 ідентичних пристроїв C з інструкцією CNT, яка використовується у FB.

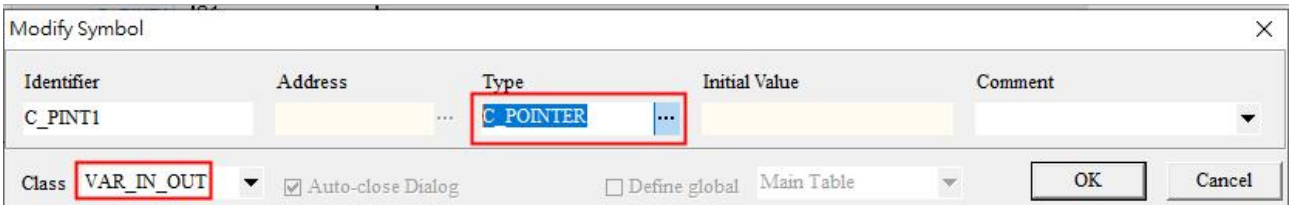
Приклад 1 (з використанням інструкції CNT у POU)

Коли SM408 увімкнено вперше, інструкція завантажує значення налаштування 10 у лічильник C0, і лічильник починає підрахунок. Після того, як SM408 десять разів перемикається з ВІМК. на УВІМК., значення в C0 становить 10, а контакт C0 увімкнено. Після того, як C0 увімкнено, якщо SM408 продовжує перемикатися з ВІМК. на УВІМК., інструкція не збільшує значення в C0 після досягнення значення налаштування для C0.

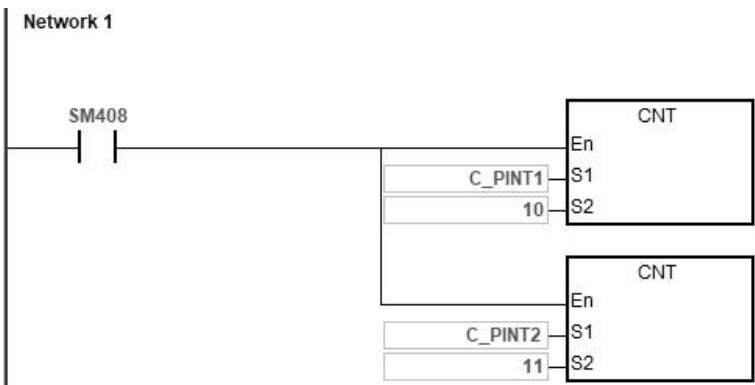


Приклад 2 (використання функціонального блоку в інструкції CNT)

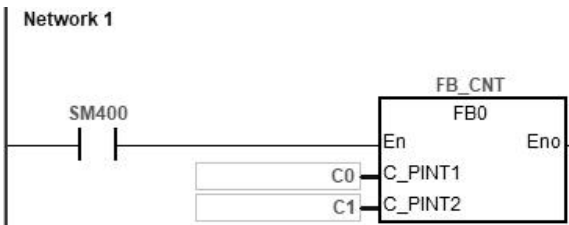
Крок 1: Оголошіть функціональний блок як C_Pointer.



Цей функціональний блок



Крок 2: Призначте C_Pointer як C0 і C1 у програмі блоку (POU).



Додаткові зауваження

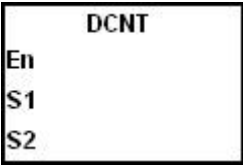
Коли ви оголошуєте операнд **S**₁ в ISPSOft, виберіть тип даних COUNTER.

API	Код інструкції				Операнд								функція				
1004		DCNT			S_1, S_2								32-розрядний лічильник				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1							O										
S_2								O					O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													
S_2			●				●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ



S_1 : Лічильник значення
 S_2 : Установка значення для лічильника

Пояснення

6

- Ця інструкція вмикає 32-бітний лічильник між HC0–HC255.
- Якщо ви оголошуєте операнд S_1 в ISPSOft, ви не можете вибрати тип даних CNT; замість цього вкажіть номер пристрою HC.
- Для лічильників зворотного/зворотного відліку HC0–HC63, коли умовний контакт цієї інструкції перемикається з ВІМК. до УВІМК., лічильники ведуть підрахунок, збільшуючи значення на 1, коли SM621–SM684 вимкнено, або здійснюють відлік шляхом зменшення значень на 1, коли SM621–SM684 увімкнено.
- Лічильники підрахунку HC64–HC199 підраховують, збільшуючи значення на 1, коли умовний контакт інструкції DCNT перемикається з ВІМК. на УВІМК.
- Лічильник припиняє підрахунок, коли інструкція DCNT вимкнена, але інструкція не очищає початкове значення підрахунку. Використовуйте інструкцію RST, щоб очистити значення підрахунку та скинути контакт у положення ВІМК.
- Зверніться до прикладу 2 інструкції CNT для демонстрації використання інструкції DCNT у функціональному блоці (FB).
- Зверніться до наступних сторінок, щоб дізнатися більше про високошвидкісний лічильник HC200–HC255. Щоразу, коли ПЛК сканує цю інструкцію, він оновлює призначений лічильник для його оновлених значень. Тому пропонується використовувати цю інструкцію в РОУ. ПРИМІТКА: номер високошвидкісного HE можна дублювати.

8. Оскільки ПЛК серії AS має різні вбудовані типи входів, пропонується різна максимальна частота входу. Перегляньте таблицю нижче для довідки. Можливо, максимальна вхідна частота не може бути досягнута для типу занурення та закисання. Якщо тип виходу нормально відкритий (NO) і струм приводу недостатній. Зверніться до розділу 4 Посібника з апаратного забезпечення та експлуатації серії AS, щоб отримати додаткові відомості про проводку.

Назва моделі	Тип введення	Введіть номер	Максимальна вхідна частота
AS324MT-A	Диференціал	X0,0 ~ X0,3	4 МГц
	Поглинання / Джерело	X0,4 ~ X0,11	200К Гц
AS332T-A AS332P-A	Поглинання / Джерело	X0,0 ~ X0,11	200К Гц
		X0,12 ~ X0,15	10К Гц
AS320T-B AS320P-B	Поглинання / Джерело	X0,0 ~ X0,7	200К Гц
AS300N-A		Немає вбудованих входів	
AS228T-A AS228P-A AS228R-A	Поглинання / Джерело	X0,0 ~ X0,7	200К Гц
		X0,8 ~ X0,15	10К Гц
AS218TX-A AS218PX-A AS218RX-A	Поглинання / Джерело	X0,0 ~ X0,7	200К Гц
AS132T-A AS132P-A AS132R-A AS148T-A AS148P-A AS148R-A AS164T-A AS164P-A AS164R-A	Поглинання / Джерело	X0,0 ~ X0,7	200 кГц
		X0,8 ~ X0,15	10 кГц

Приклад 1

МЕРЕЖА1:

Коли ПЛК працює, значення лічильника HC0 очищається, і лічильник рахується, оскільки SM621 вимкнено. У цей час SM408 увімкнено вперше. Отже, інструкція завантажує значення налаштування 10 у лічильник HC0, і лічильник починає підрахунок.

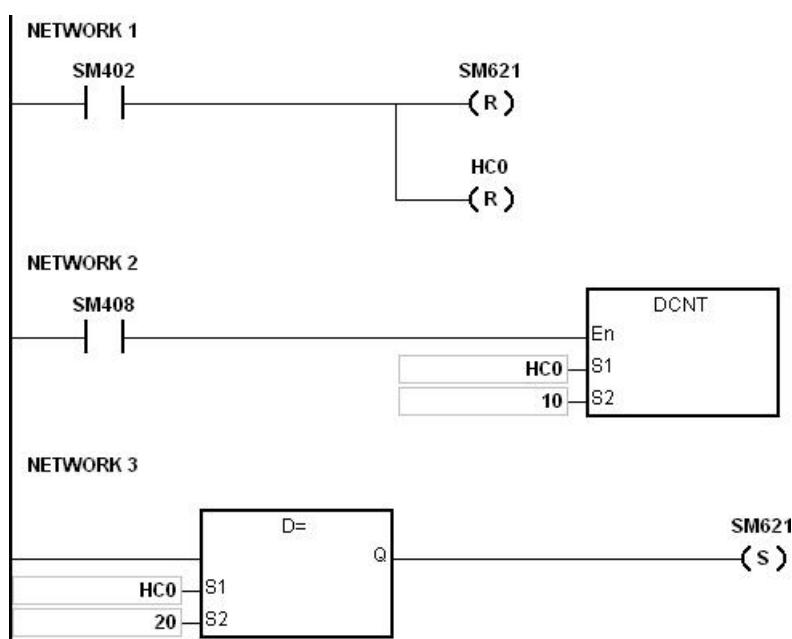
МЕРЕЖА2:

Після того, як SM408 десять разів перемикається з ВИМК. на УВИМК., значення лічильника HC0 відповідає встановленому значенню 10, а контакт HC0 увімкнено. Після того, як HC0 увімкнено, значення лічильника продовжує зростати, оскільки SM408 продовжує змінюватися з ВИМК. на УВИМК., навіть якщо значення HC0 досягло встановленого значення.

МЕРЕЖА3:

Коли HC0 продовжує відлік і значення досягає встановленого значення 20, лічильник починає зворотний відлік, оскільки SM621 увімкнено в програмі. Після того, як SM408 десять разів перемикається з ВИМК. на УВИМК. і значення HC0 зменшується з 10 до 9, контакт HC0 вимкнено.

Після того, як контакт HC0 вимкнено, значення HC0 все ще продовжує зменшуватися, оскільки SM408 продовжує змінюватися з ВИМК. на ON.



Додаткові зауваження

Щоб налаштувати режим SM621–SM684, зверніться до пояснення 32-розрядного лічильника HC у розділі 2.

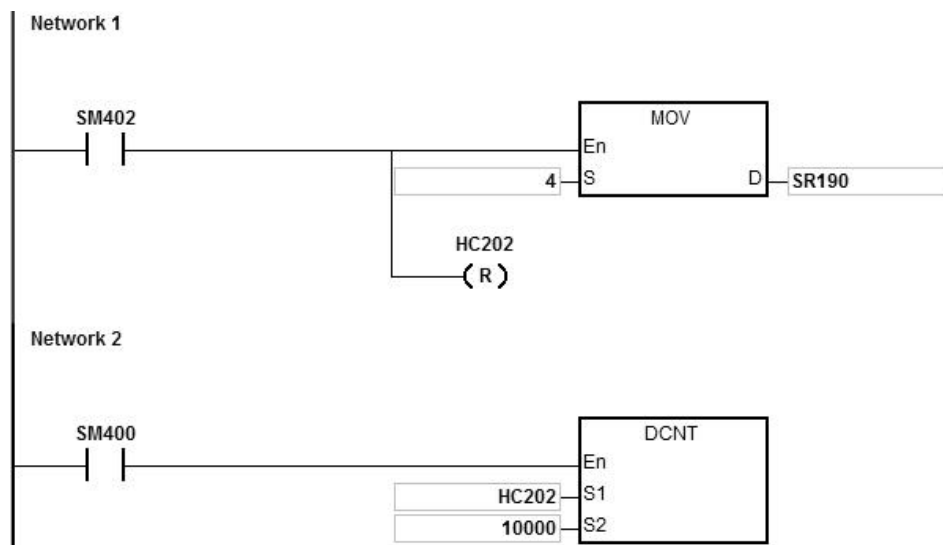
Приклад 2

МЕРЕЖА1:

Під час роботи ПЛК встановіть значення лічильника HC202 на чотири часові частоти (налаштування режиму слід встановити перед виконанням інструкції DCNT). Потім значення лічильника HC202 обнуляється.

МЕРЕЖА2:

Після того, як значення лічильника HC202 досягло встановленого значення 1000, контакт HC202 вмикається.



Пояснення високошвидкісного лічильника:

Високошвидкісні лічильники серії AS можна розділити на апаратні лічильники (максимальна вхідна частота 200 кГц і для диференціальних вхідних точок до 4 МГц) і програмні лічильники (максимальна частота 10 кГц). Зверніться до специфікації апаратного забезпечення, щоб дізнатися більше про обмеження вхідного сигналу.

Апаратний лічильник

Вхід HC №	X0.															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
HC200	P №1												R №4			
HC201	P	D №1											R			
HC202	A №1	B №1											R			
HC203	-- №2	--											--			
HC204			P											R		
HC205			P	D										R		
HC206			A	B										R		
HC207			--	--										--		
HC208					P										R	
HC209					P	D									R	
HC210					A	B									R	
HC211					--	--									--	
HC212							P									R
HC213							P	D								R
HC214							A	B								R
HC215							--	--								--
HC216									P							
HC217									P	D						
HC218									A	B						
HC219									--	--						
HC220											P					
HC221											P	D				
HC222											A	B				
HC223											--	--				

Примітка 1: P: однофазний імпульсний вхід, D: вхід сигналу напрямку, A і B: двофазний вхід, R: вхід сигналу скидання. Тільки один з чотирьох режимів введення можна використовувати в програмуванні ПЛК. Наприклад, якщо HC200 редагується, HC20- HC203 більше не можна редагувати.

Примітка 2: -- означає, що режим підрахунку зарезервований і зараз недоступний. Порожнє поле означає відсутність функції.

Примітка 3: зверніться до таблиці SM/SR для вибору стану підрахунку вгору/зниження та кількості разів для введення частоти. Примітка 4: функція R (вхід скидання) вимкнена за замовчуванням. Зверніться до порівняльної таблиці SM/SR, щоб дізнатися, як використовувати R.

Візьмемо, наприклад, HC200. SM291 перемикається в положення ON, щоб запустити функцію R, а потім наростаючий фронт X0.12 запускає очищення значення HC200. Якщо в ПЛК, який ви використовуєте, немає вбудованих вхідних точок, включаючи X0.12 ~ X0.15, ви можете використовувати зовнішні переривання або використовувати інструкції для стирання лічильників, щоб очистити лічильники.

Програмний лічильник:

Вхід HC №	X0.															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
HC232													P			
HC233													P	D		
HC234													A	B		
HC235													UP №5	DN №5		
HC236															P	
HC237															P	D
HC238															A	B
HC239															UP	DN
HC240		P														
HC241	UP	DN														
HC242				P												
HC243			UP	DN												
HC244						P										
HC245					UP	DN										
HC246							P									
HC247							UP	DN								
HC248									P							
HC249								UP	DN							
HC250										P						
HC251											UP	DN				
HC252													P			
HC253																P

Примітка 5: UP: однофазний вхід зворотного відліку (те саме, що CW), DN: однофазний вхід зворотного відліку (те саме, що CCW)

Високошвидкісні лічильники між HC200–HC255 є зарезервованими пристроями всередині ПЛК і не перераховані в цій таблиці. Не рекомендується використовувати ці лічильники в програмі. Якщо в ПЛК, який ви використовуєте, немає вбудованих точок введення, включаючи X0.12 ~ X0.15, НЕ використовуйте відповідні лічильники.

У наступній таблиці наведено високошвидкісний лічильник, функції, скидання, реверсування та режим підрахунку.

Функція зворотного/зворотного відліку				Запуск функції скидання	Зміна напрямку	Режим підрахунку
HC №	№ SM	Атрибут	Пояснення	№ SM	№ SM	CP №
HC200	SM300	R/W	Показати/налаштувати	SM291	SM281 (Застосовується для HC201, HC202 ^{#7})	SR190
HC201	SM301	R	Показати			
HC202	SM302	R	Показати			
HC204	SM304	R/W	Показати/налаштувати	SM292	SM282 (Застосовується для HC205, HC206 ^{#7})	SR191
HC205	SM305	R	Показати			
HC206	SM306	R	Показати			
HC208	SM308	R/W	Показати/налаштувати	SM293	SM283 (Застосовується для HC209, HC210 ^{#7})	SR192
HC209	SM309	R	Показати			
HC210	SM310	R	Показати			
HC212	SM312	R/W	Показати/налаштувати	SM294	SM284 (Застосовується для HC213, HC214 ^{#7})	SR193
HC213	SM313	R	Показати			
HC214	SM314	R	Показати			
HC216	SM316	R/W	Показати/налаштувати	--	SM285 (Застосовується для HC217, HC218 ^{#7})	SR194
HC217	SM317	R	Показати			
HC218	SM318	R	Показати			
HC220	SM320	R/W	Показати/налаштувати	--	SM286 (Застосовується для HC221, HC222 ^{#7})	SR195
HC221	SM321	R	Показати			
HC222	SM322	R	Показати			
HC232	SM332	R/W	Показати/налаштувати	--	SM287 (Застосовується для HC233, HC234 ^{#7})	SR196
HC233	SM333	R	Показати			
HC234	SM334	R	Показати			
HC235	SM335	R	Показати	--	SM288 (Застосовується для HC237, HC238 ^{#7})	SR197
HC236	SM336	R/W	Показати/налаштувати			
HC237	SM337	R	Показати			
HC238	SM338	R	Показати			
HC239	SM339	R	Показати	--	--	Підтримує одну часову частоту та наростаючий фронт – лише запущений підрахунок
HC240	SM340	R/W	Показати/налаштувати			
HC241	SM341	R	Показати			
HC242	SM342	R/W	Показати/налаштувати			
HC243	SM343	R	Показати			
HC244	SM344	R/W	Показати/налаштувати			
HC245	SM345	R	Показати	--	--	Підтримує одноразову частоту та висхідний фронт – тригер
HC246	SM346	R/W	Показати/налаштувати			

Функція зворотного/зворотного відліку				Запуск функції скидання	Зміна напрямку	Режим підрахунку
HC №	№ SM	Атрибут	Пояснення	№ SM	№ SM	CP №
						лише підрахунок
HC247	SM347	R	Показати	--	--	Підтримує одну часову частоту та наростаючий фронт – лише запущений підрахунок
HC248	SM348	R/W	Показати/налаштувати			
HC249	SM349	R	Показати			
HC250	SM350	R/W	Показати/налаштувати			
HC251	SM351	R	Показати			
HC252	SM352	R/W	Показати/налаштувати			
HC253	SM353	R	Показати			

Примітка 1. Усі спеціальні прапорці SM у таблиці вище OFF за замовчуванням.

Примітка 2. Коли SM у розділі «Функція підрахунку/зворотного відліку» OFF, це означає, що відповідний лічильник підраховує або відображає, що він підраховує. Якщо SM увімкнено, це вказує на те, що відповідний лічильник веде зворотний відлік, або відображає, що він веде зворотний відлік.

Примітка 3: « R » під атрибутом означає «Лише читання», а R/W вказує на «Читання/запис».

Примітка 4. Спеціальні регістри SR у розділі «Режим підрахунку» за замовчуванням мають 1 частотний вхід.

Використовуйте 2 як вхідне значення для подвійної частоти та 4 для чотирикратної частоти. Чотирикратна частота застосовна лише до двофазного вхідного лічильника A/B. Якщо вхідне значення не дорівнює 1, 2 або 4 у SR, це означає, що ПЛК використовує одноразову частоту.

Примітка 5. Усі однофазні лічильники в таблиці підраховують із використанням одноразової частоти та режиму підрахунку за висхідним фронтом

змінює точку введення з ВІМК. на УВІМК.

Примітка 6: Лічильники P (імпульсний вхід) і D (напрямок) можуть змінювати напрямок. Коли SM увімкнено, напрямок підрахунку

(вгору/вниз) перевертається. Наприклад, коли вхід попередньо встановленого напрямку OFF, лічильник підраховує вгору. Коли SM перемикається в положення ON, лічильник переходить у зворотний відлік.

Примітка 7: доступно для FW V1.08 або новішої версії.

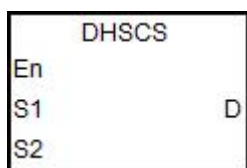
API	Код інструкції				Операнд					функція							
1005	D	HSCS			S_1, S_2, D					Налаштування високошвидкісного порівняння							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							O									
S_2								●					O	O		
D		O	O	O				O								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
D	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ



- S_1 : Номер лічильника
 S_2 : Порівняльна цінність
D : Результат порівняння

6

Пояснення

- Використовуйте цю інструкцію з високошвидкісними лічильниками з номерами HC200 і вище. Якщо значення у високошвидкісному лічильнику заданий S_1 змінюється збільшенням або зменшенням на 1, інструкція DHSCS виконує порівняння негайно. Коли поточне значення високошвидкісного лічильника дорівнює порівняльному значенню, зазначеному в S_2 , пристрій, вказаний D, переходить у стан ON. Після цього пристрій, визначений D, залишається увімкненим, навіть якщо результатом порівняння є те, що поточне значення та порівняльне значення не збігаються.
- Якщо пристрій, визначений D, є Y0.0–Y0.15, а значення S_2 дорівнює поточному значенню, результат порівняння високошвидкісного лічильника виводиться на вихідні клеми Y0.0–Y0.15. Цикл сканування впливає на інші пристрої Y, але ця інструкція миттєво оновлює всі пристрої та не впливає на цикл сканування. Для ПЛК із FW V1.06.00 (V1.06.00 включно) або пізнішими версіями цикл сканування впливає на пристрої Y0.12 ~ Y0.15.
- Операнд D також може вказувати пристрій переривання I між 1200–1267.
- Швидкісні лічильники поділяються на програмні та апаратні. Нумери доступних високошвидкісних компараторів і пристроїв переривання наведено в наступній таблиці.

Тип	Діапазон лічильника чисел	Швидкодіючий компаратор номер	Високошвидкісне переривання номер пристрою
Апаратний лічильник	HC200 - HC203	компаратор: HCC00-HCC03	I200-I203
	HC204 - HC207	компаратор: HCC04-HCC07	I210-I213
	HC208 - HC211	компаратор: HCC08-HCC11	I220-I223
	HC212 - HC215	компаратор: HCC12-HCC15	I230-I233
	HC216 - HC219	компаратор: HCC16-HCC19	I240-I243
	HC220 - HC223	компаратор: HCC20-HC223	I250-I253
Програмний лічильник	HC232 - HC253	-	I260-I267

5. Пояснення апаратних компараторів для інструкцій DHSCS, DHSCR, DHSCY і DHSZ:

- Кожна група апаратних лічильників спільно використовує 4 високошвидкісні компаратори. Одна інструкція DHSCS або DHSCR займає 1 високошвидкісний компаратор. Одна інструкція DHSZ, DHSCY використовує 2 високошвидкісних компаратора.
- Під час редагування програми кожна група апаратних лічильників може використовувати не більше 4 високошвидкісних компараторів для інструкцій DHSCS, DHSCR, DHSCY або DHSZ; інакше виникає синтаксична помилка.

6. Пояснення компараторів програмного забезпечення для інструкцій DHSCS, DHSCR і DHSCY:

- Є 8 компараторів програмного забезпечення для порівняння функції «Установити» або «Скинути». Кожна інструкція DHSCS, DHSCR або DHSCY використовує один високошвидкісний компаратор.
- Програмні компаратори порівнюють переривання, призначаючи відповідний програмний компаратор відповідно до номерів переривань. Зауважте, що один і той самий номер переривання не можна використовувати повторно.
- Для інструкцій DHSCS, DHSCR або DHSCY кількість компараторів Set або Reset не може перевищувати восьми випадків у програмі; інакше виникає синтаксична помилка.

7. Пояснення компараторів програмного забезпечення для інструкції DHSZ:

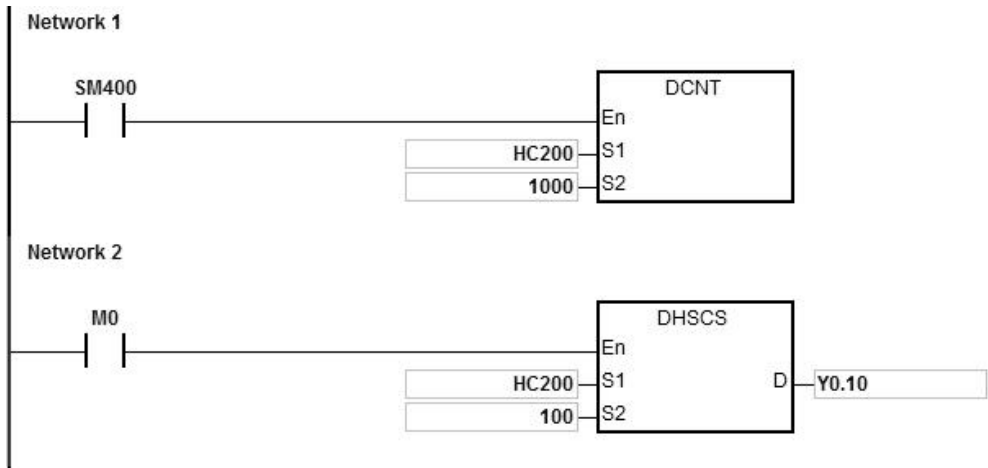
- Існує вісім програмних компараторів для порівняння зон. Одна інструкція DHSZ використовує один компаратор.
- Інструкція DHSZ може використовувати максимум вісім програмних компараторів; інакше виникає синтаксична помилка, якщо використовується більше восьми компараторів.

8. Інструкції DHSCS, DHSCR, DHSCY і DHSZ не можна використовувати в мові програмування ST, завданнях переривання або функціональному блоці, який викликається лише один раз.

Приклад 1

Коли M0 увімкнено, виконується інструкція DHSCS.

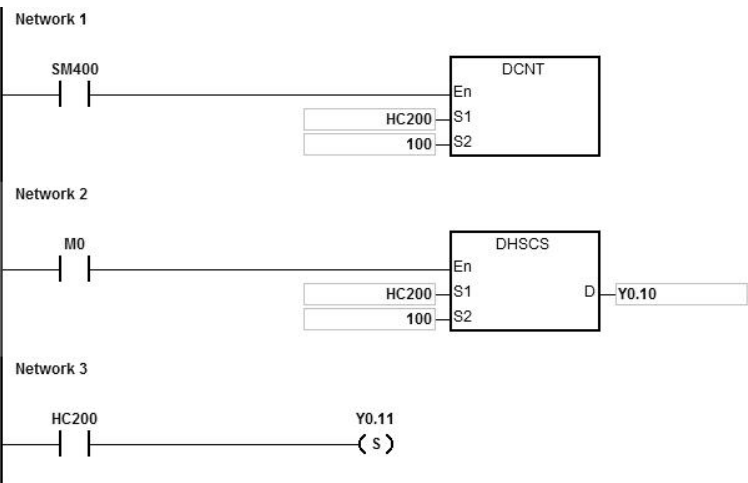
Коли поточне значення HC200 змінюється з 99 на 100 або зі 101 на 100, Y0.10 увімкнено, що виводить на зовнішній вихідний термінал Y0.10 у реальному часі та залишається увімкненим.



Приклад 2

Вихід Y інструкції DHSCS відрізняється від загального виводу Y.

1. Коли M0 увімкнено, виконується інструкція DHSCS. Коли поточне значення HC200 змінюється з 99 на 100 або зі 101 на 100, Y0.10 негайно виводить свій стан на зовнішній вихідний термінал, і на нього не впливає час сканування програми.
2. Коли поточне значення HC200 змінюється з 99 до 100, контакт HC200 негайно вмикається. Коли виконується SET Y0.11, на Y0.11 все ще впливає час сканування, і він виводить свій стан лише після проходження END.

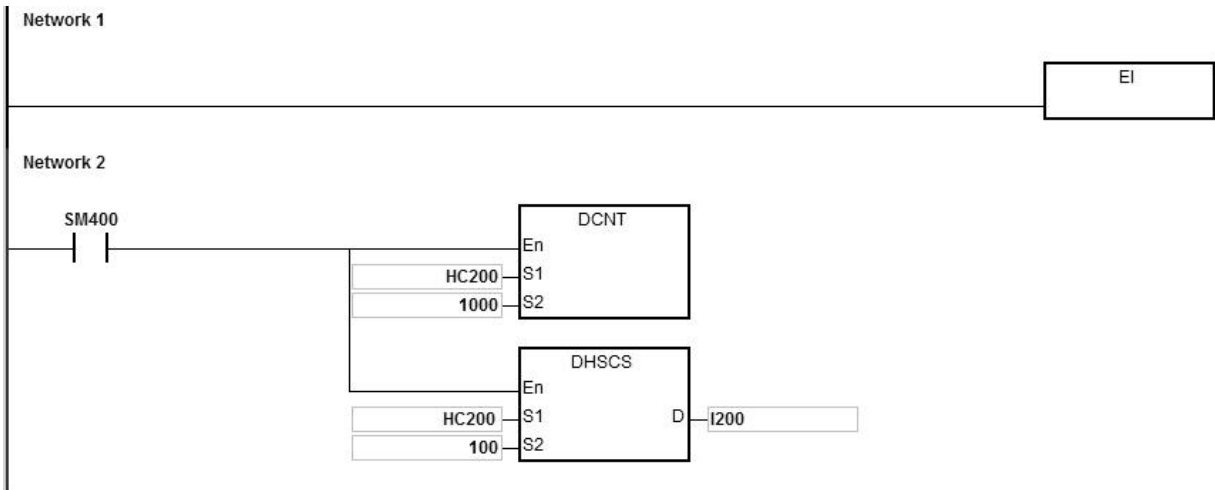


Приклад 3

Використання переривання в апаратному високошвидкісному порівнянні.

Коли поточне значення HC200 змінюється з 99 на 100 або зі 101 на 100, програма переходить до показника переривання, щоб виконати програму переривання, і Y0.10 увімкнено.

Основна програма:

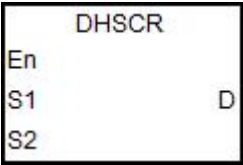


Програма переривання I200:



API	Код інструкції				Операнд							функція						
1006	D	HSCR			S ₁ , S ₂ , D							Скидання порівняння високошвидкісного введення						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S ₁							O											
S ₂								●					O	O				
D		O	O	O			O	O										
Тип даних	BOOL	WORD		DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S ₁													●					
S ₂				●				●										
D	●																	
Інструкція пульсу							16-бітна інструкція					32-розрядна інструкція						
-							-					AS						

СИМВОЛ



- S_1 : Номер лічильника
- S_2 : Порівняльна цінність
- D : Результат порівняння

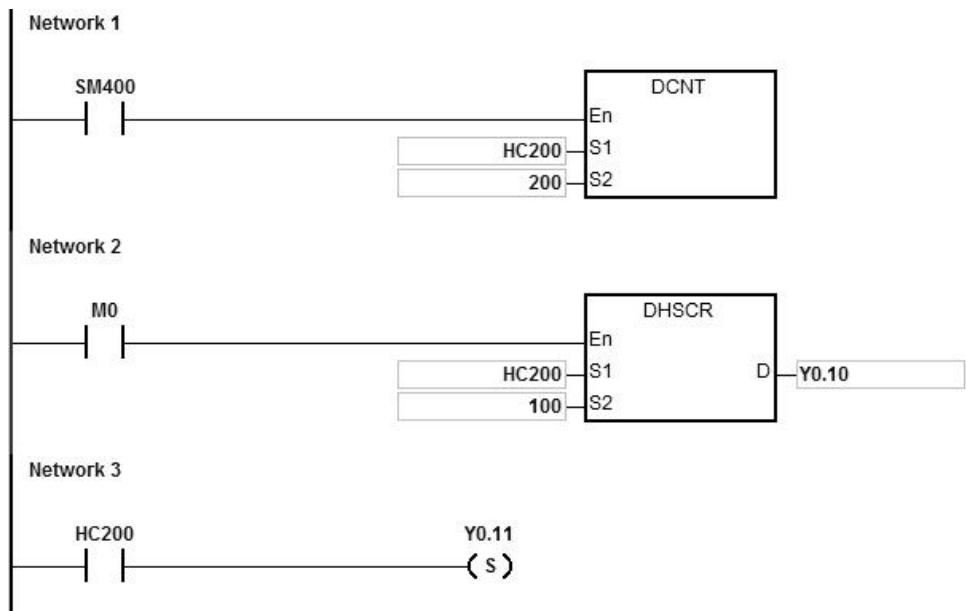
6

Пояснення

- Використовуйте цю інструкцію з високошвидкісним лічильником під номером HC200 і вище. Якщо значення у високошвидкісному лічильнику заданий S_1 змінюється шляхом збільшення або зменшення, інструкція DHSCR виконує порівняння негайно. Коли поточне значення високошвидкісного лічильника дорівнює порівняльному значенню, зазначеному в S_2 , пристрій, вказаний D, переходить у стан OFF. Після цього пристрій, визначений D, залишається ВИМКНЕНИМ, навіть якщо результатом порівняння є те, що поточне значення та порівняльне значення не збігаються.
- Якщо пристрій, визначений D, є Y0.0–Y0.15, а порівняльне значення S_2 дорівнює поточному значенню лічильника, результат порівняння виводиться на зовнішні вихідні клеми Y0.0–Y0.15. Цикл сканування впливає на інші пристрої Y, але ця інструкція миттєво оновлює всі пристрої та не впливає на цикл сканування.
- Операнд D також може вказувати пристрій HC для скидання та обмежується умовою, за якої число високошвидкісного лічильника таке ж, як і для S_1 .
- Для отримання додаткової інформації зверніться до інструкції DHSCS (API 1005).

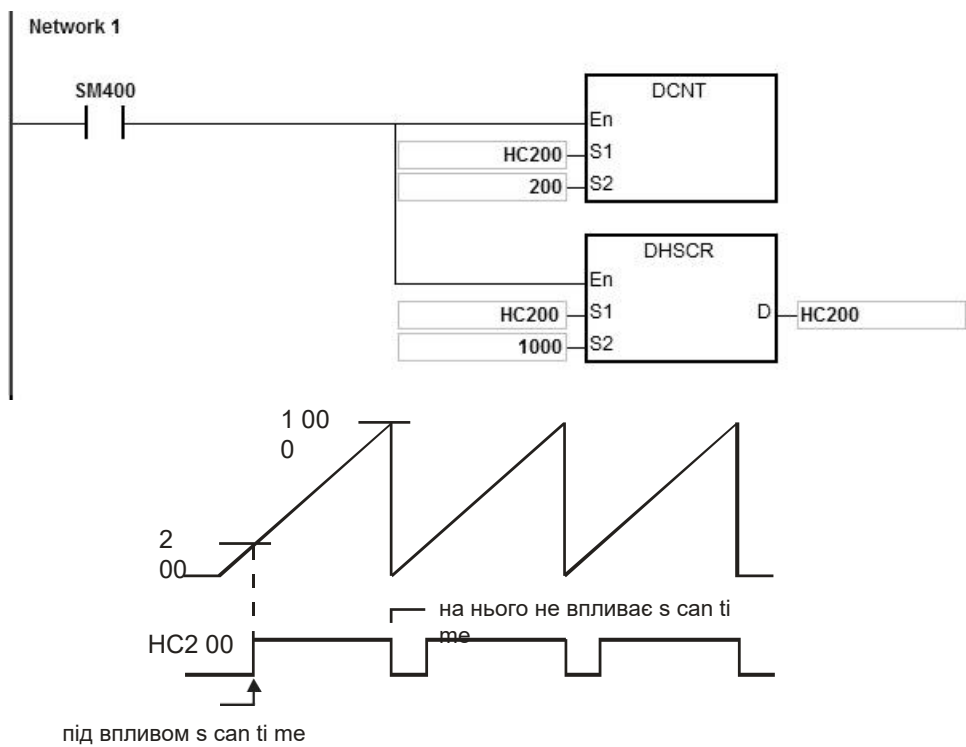
Приклад 1

- 1. Коли M0 увімкнено, а HC200 змінює своє поточне значення з 99 на 100 або зі 101 на 100, Y0.10 скидається на ВИМК.
- 2. Коли HC200 змінює своє поточне значення зі 199 на 200, контакт HC200 увімкнено, а Y0.11 увімкнено, але вихід затримується на час сканування програми.



Приклад 2

Якщо ви вкажете HC200 як апаратний високошвидкісний лічильник з таким же номером, контакт HC200 буде скинуто у положення OFF, коли HC200 змінює своє поточне значення з 999 на 1000 або з 1001 на 1000.



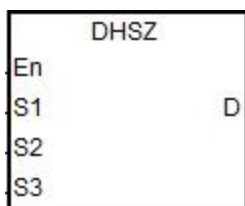
API	Код інструкції			Операнд								функція					
1007	D	HSZ		S_1, S_2, S_3, D								Швидке порівняння зон введення					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							O									
S_2								●					O	O		
S_3								●					O	O		
D		O	O	O				O								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
S_3			●				●						
D	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ



- S_1 : Номер лічильника
- S_2 : Нижня межа зони порівняння
- S_3 : Верхня межа зони порівняння
- D : Результат порівняння (3 пристрої підряд)

Пояснення

- Використовуйте цю інструкцію з високошвидкісними лічильниками з номерами HC200 і вище. Нижня межа S_2 повинна бути меншою ніж верхня межа S_3 . Якщо ви неправильно встановите граничні значення зони, ПЛК автоматично налаштує їх.
- Якщо S_1 визначає програмний лічильник і вказаний лічильник змінюється шляхом збільшення або зменшення на 1 значення,

Інструкція DHSZ виконує порівняння негайно. Умова порівняння та вихідний стан показані в наступній таблиці.

Умова порівняння	D +0 стан	D +1 стан	D +2 стан
Значення підрахунку S_1 < нижньої межі (S_2)	ON	OFF	OFF
Нижня межа (S_2) <= значення підрахунку S_1 < верхня межа (S_3)	OFF	ON	OFF
Значення підрахунку S_1 >= верхня межа (S_3)	OFF	OFF	ON

Примітка. Ви повинні встановити нижню межу (S_2) меншою за верхню межу (S_3). Якщо встановити межі зони неправильно, ПЛК автоматично виконає налаштування.

3. Якщо S_1 визначає апаратний лічильник і значення вказаного лічильника досягає нижньої межі (S_2) або верхньої межі (S_3), інструкція DHSZ виконує порівняння негайно відповідно до напрямку підрахунку (вгору/вниз) . умова порівняння та вихідний стан показані в наступній таблиці.

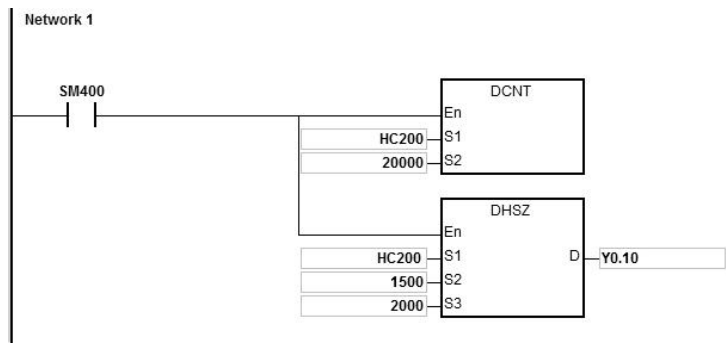
Напрямок відліку	Умова порівняння	D +0 стан	D +1 стан	D +2 стан
Порахуйте	Значення підрахунку S_1 == нижня межа (S_2)	OFF	ON	OFF
	Значення підрахунку S_1 == верхня межа (S_3)	OFF	OFF	ON
Зворотний відлік	Значення підрахунку S_1 == нижня межа (S_2)	ON	OFF	OFF
	Значення підрахунку S_1 == верхня межа (S_3)	OFF	ON	OFF

4. Якщо пристрій, визначений D , Y0.0–Y0.15, результат порівняння виводиться на зовнішні вихідні клеми Y0.0–Y0.15. Цикл сканування впливає на інші пристрої Y, але ця інструкція миттєво оновлює всі пристрої та не впливає на цикл сканування. Для ПЛК із FW V1.06.00 (V1.06.00 включно) або пізнішими версіями цикл сканування впливає на пристрої Y0.12 ~ Y0.15.

5. Зверніться до інструкції DHSCS (API 1005), щоб дізнатися більше про порівняння високошвидкісних зон.

приклад

- 1. Коли D вказано як Y0.10, Y0.11–Y0.12 також вказуються автоматично.
- 2. Інструкція порівнює поточне значення в HC200 з верхньою/нижньою межею (1500/2000) зони порівняння, і один із Y0.10–Y0.12 увімкнено відповідно до результату порівняння.
- 3. Коли поточне значення в HC200 <1500, Y0.10 увімкнено. Коли 1500<= поточне значення в HC200<2000, Y0.11 увімкнено.
Коли поточне значення в HC200>=2000, Y0.12 увімкнено.

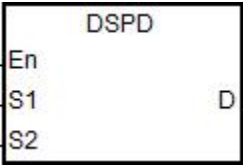


API	Код інструкції				Операнд							функція					
1008	D	SPD			S_1, S_2, D							Виявлення швидкості					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1							O										
S_2								O	O				O	O			
D								O									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1												●	
S_2			●				●						
D			●				●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ



- S_1 : Лічильник значення
- S_2 : Установка значення для часу циклу
- D : Кількість імпульсів з попереднього циклу сканування

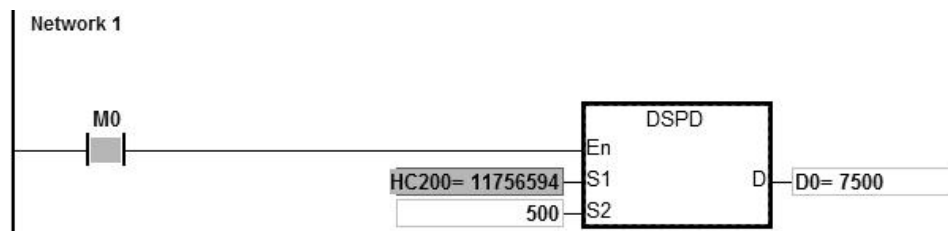
Пояснення

- Ця інструкція вимагає використання S_1 з інструкцією DCNT (API 1004), щоб увімкнути високошвидкісний лічильник з номери лічильника вище HC200 (включаючи HC200).
- Одиниці вимірювання часу для S_2 (налаштування часу циклу) — мілісекунди (мс). Налаштування має бути між 10–1000. Коли значення виходить за межі діапазону, ПЛК виконує інструкцію з S_2 при мінімальному значенні або максимальному значенні, і повідомлення про помилки відсутні.
- Коли підрахунок досягає встановленого значення в S_2 , ця інструкція зберігає кількість імпульсів у пристрої, визначеному D, і на неї не впливає цикл сканування ПЛК.
- Ця інструкція не має обмежень під час редагування, але вона дозволяє виконувати лише вісім наборів інструкцій визначення швидкості одночасно. Система ігнорує дев'ятий набір інструкцій визначення швидкості, і повідомлення про помилки відсутні. При виконанні цієї інструкції записуються значення параметрів для операнда, і під час виконання цієї інструкції ви не можете редагувати параметри.

приклад

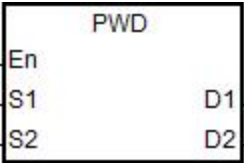
Ви можете використовувати інструкцію DSPD для визначення швидкості, якщо є вхідний імпульсний сигнал на X0.0. Коли M0 увімкнено, інструкція оновлює кількість імпульсів, підрахованих HC200 у D0 кожні 500 мс.

У наступному прикладі значення в D0 становить 7500, а фактична вхідна частота імпульсу X0.0 становить 15 кГц (7500/500 мс).



API	Код інструкції				Операнд							функція						
1009		PWD			S ₁ , S ₂ , D							Визначення ширини імпульсу						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S ₁	O																	
S ₂								O	O				O	O				
D ₁								O										
D ₂		O	O	O														
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING					
S ₁	●																	
S ₂		●					●											
D ₁				●				●										
D ₂	●																	
							Інструкція пульсу					16-бітна інструкція			32-розрядна інструкція			
							-					AS			-			

СИМВОЛ



- S_1 : Номер точки входу
- S_2 : Одиниця вимірювання
- D_1 : Час виявлення ширини імпульсу (32-бітове значення)
- D_2 : Прапор оновлення

Пояснення

1. S_1 підтримує наступні 12 входів, X0.0~X011, але S_1 не може спільно використовувати ті самі входи з високошвидкісним лічильником.
2. S_2 – одиниця вимірювання. Інструкція не виконується, якщо значення налаштування S_2 не є дійсним кодом S_2 із наступна таблиця.

S ₂ код	Одиниця вимірювання	Дальність виявлення	Діапазон частот	Зауваження
0	1 нас		1 Гц - 10 кГц	
1	1 мс		0,02 Гц - 100 Гц	
2	10нс		10 Гц - 1 МГц	Непарні входи NI підтримується.
4	1 нас		1 Гц - 10 кГц	
5	1 мс		0,02 Гц - 100 Гц	
Інші значення	Інструкція не виконується.			

3. Інструкція зберігає час виявлення ширини імпульсу (32-бітне значення) у D_1 , а діапазон виявлення становить 0–100 000 000. Якщо значення перевищує максимальне значення, воно обробляється як максимальне значення. Якщо значення дорівнює 0, це означає, що під час виконання цієї інструкції жоден вхід не перемикається з ON на OFF.

4. D_2 — це прапор оновлення. Кожного разу, коли виявлення вхідного сигналу S_1 завершується і інструкція сканується, оновлений прапор перемикається на ON на один час циклу сканування. Ви можете перевірити, чи оновлено значення виявлення, за допомогою позначки оновлення. Коли система виконує інструкцію вперше, прапор оновлення скидається на ВІМК.

5. Якщо значення в S_2 дорівнює 0, 1 або 2, зверніться до часової діаграми нижче для виконуваних процедур, таких як збереження значень виявлення та оновлення позначок під час виконання інструкції. Таймер запускається, коли вхід S_1 перемикається з ВІМК. на УВІМК., як показано в позиції ① наступної діаграми. Інструкція зберігає час виявлення, коли вхід S_1 перемикається з ON на OFF, як показано в позиції ② наступної діаграми.

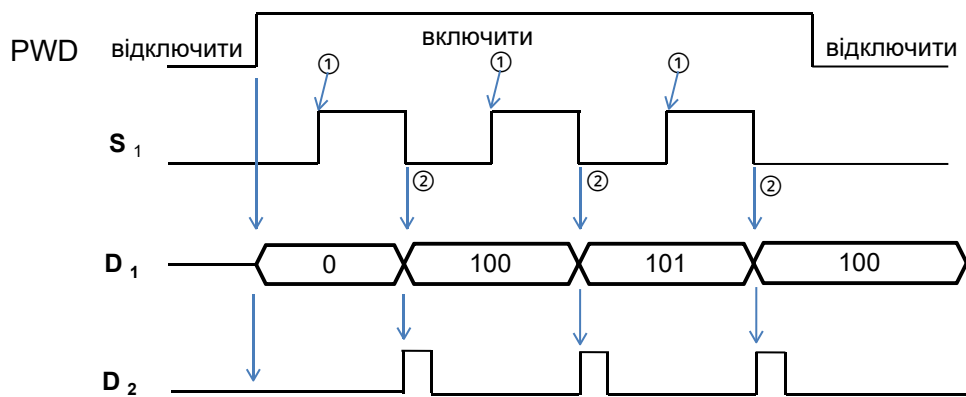


Рисунок 1. Режим виявлення, коли значення в S_2 дорівнює 0, 1 або 2

6. Якщо значення в S_2 дорівнює 4 або 5, зверніться до часової діаграми нижче для виконання процедур, таких як збереження

значення виявлення та прапори оновлення під час виконання інструкції. Таймер запускається, коли вхід S_1

перемикається з ВІМК. на УВІМК., як показано в позиції ① наступної діаграми. Інструкція зберігає час виявлення, коли вхід S_1 перемикається з ВІМК. на УВІМК., як показано в позиції ② наступної діаграми.

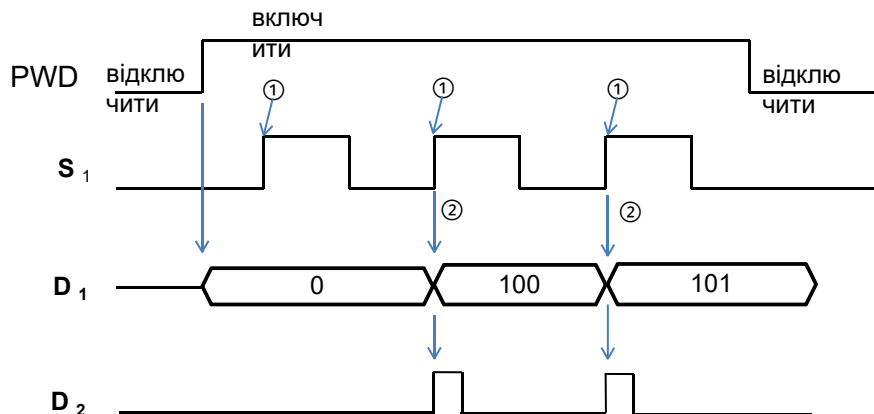


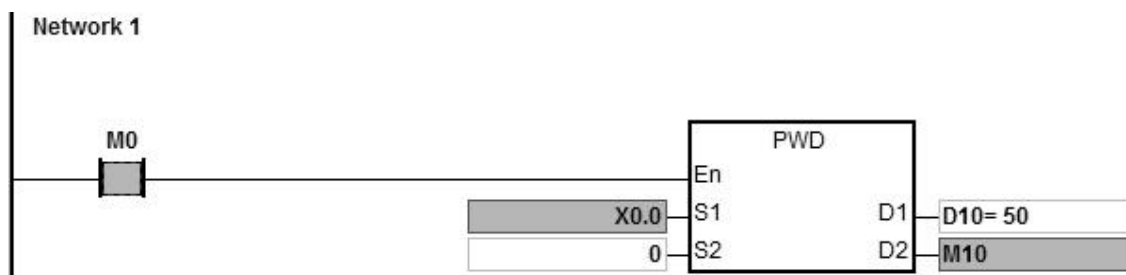
Рисунок 2 Режим виявлення, коли значення в S_2 дорівнює 4 або 5

7. Ця інструкція не має обмежень під час редагування, але вона дозволяє виконувати лише вісім наборів інструкцій визначення ширини імпульсу одночасно. Система ігнорує дев'ятий або пізніший набори інструкцій визначення ширини імпульсу, і повідомлення про помилки відсутні. Під час виконання цієї інструкції значення параметрів для операндів записуються, і ви не можете редагувати параметри під час виконання.

8. Перед виконанням цієї інструкції перевірте час відповіді апаратного введення та час імпульсу, встановлений у HWCONFIG. Наприклад, коли значення в S_2 встановлено на 0 або 2, це означає, що одиницею вимірювання часу є мікросекунди (мкс). Встановіть вхідне значення S_1 на 0, щоб вимкнути час фільтра точки введення в HWCONFIG.

приклад

Припустимо, що на вході X0.0 є імпульсний сигнал 10 кГц. Коли M0 увімкнено, інструкція PWD виявляє вхідний сигнал на X0.0 із шириною імпульсу, що зберігається в D10/D11 (32-розрядні дані), одиниця часу встановлена на 0, а виявлена ширина імпульсу з D10 становить 50 мкс.

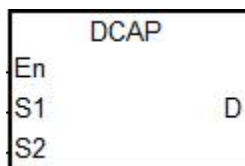


API	Код інструкції			Операнд					функція								
1010	D	CAP		S_1, S_2, D					Захоплення високошвидкісного значення підрахунку в зовнішньому вхідному перериванні								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	O																
S_2							O										
D								O									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2												●	
D			●				●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ



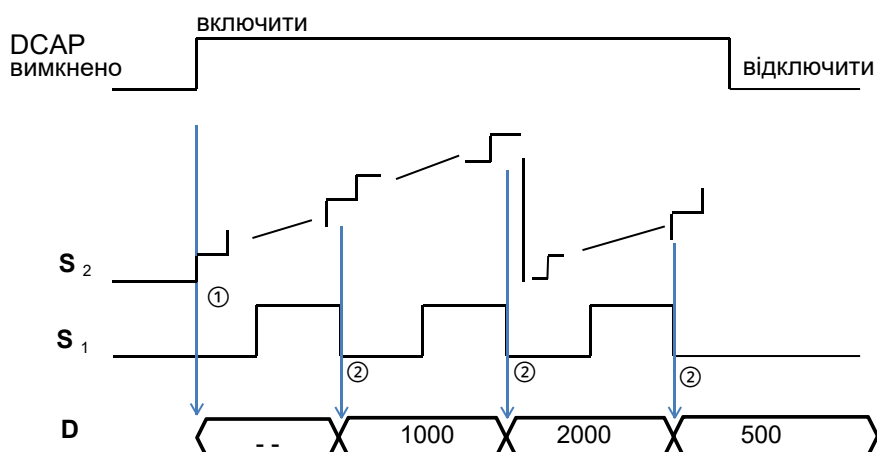
Пояснення

- S_1 : Точка входу зовнішнього переривання номер
 S_2 : Швидкісний лічильник номера
 D : Зареєструйтеся для збереження отриманого значення

- Ви можете використовувати лише 16 вхідних точок X0.0–X0.15 ПЛК у S_1 . Використовуйте одну з цих точок введення із зовнішнім перериванням службової програми для запуску функції. Зверніть увагу, що S_1 не може спільно використовувати ту саму вхідну точку з високошвидкісним лічильником.
- Виберіть пристрій високошвидкісного лічильника HC у S_2 . Ви повинні використовувати пристрій HC з інструкцією DCNT (API 1004), щоб запустити функцію підрахунку.
- Інструкція зберігає отримане значення з високошвидкісного лічильника (32-розрядний) у D , коли відбувається переривання. Інструкція зберігає дані, коли виникає переривання, і на неї не впливає сканування програми ПЛК.
- Операція інструкції показана нижче. Вхідне переривання викликається спадаючим фронтом.

① → Коли починається виконання інструкції, значення в D не змінюється, і ви можете ввести значення налаштування за замовчуванням.

② → Коли відбувається переривання в S_1 , інструкція негайно фіксує значення лічильника, визначене S_2 , і зберігає його в D .



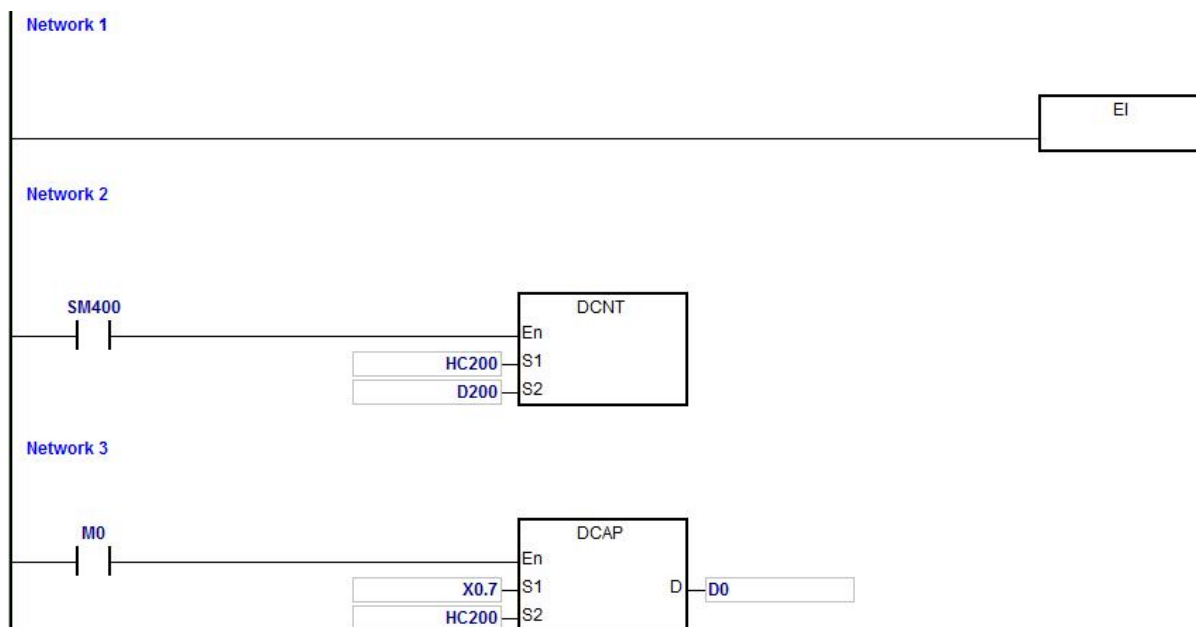
5. Ця інструкція може запускати інструкції DCAP щонайбільше для чотирьох різних точок введення. Якщо ви встановите одну точку введення як

зовнішнє переривання, викликане наростаючим фронтом і спадаючим фронтом, інструкція фіксує значення, коли вхід ініціюється наростаючим фронтом і спадаючим фронтом відповідно, і зберігає значення лічильника в пристрої, визначеному **D**. Коли дві інструкції вказують ту саму точку введення переривання, та, яка починається першою, першою використовує точку введення переривання.

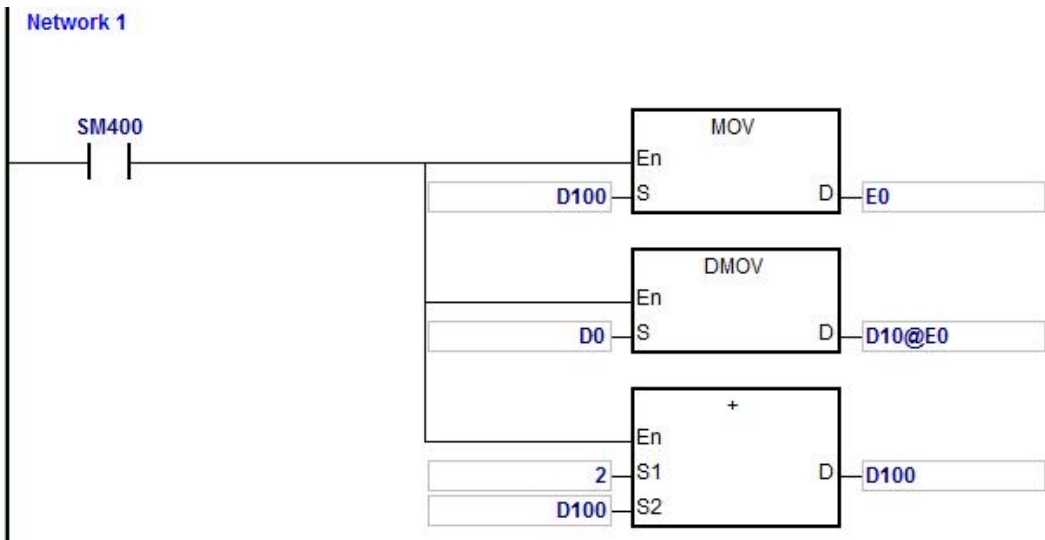
6. Встановіть номер пристрою HC у **S₂**. Рекомендовано використовувати високошвидкісні лічильники між HC200–HC255.

Докладніше про лічильники дивіться в поясненні інструкції DCNT (API 1004).

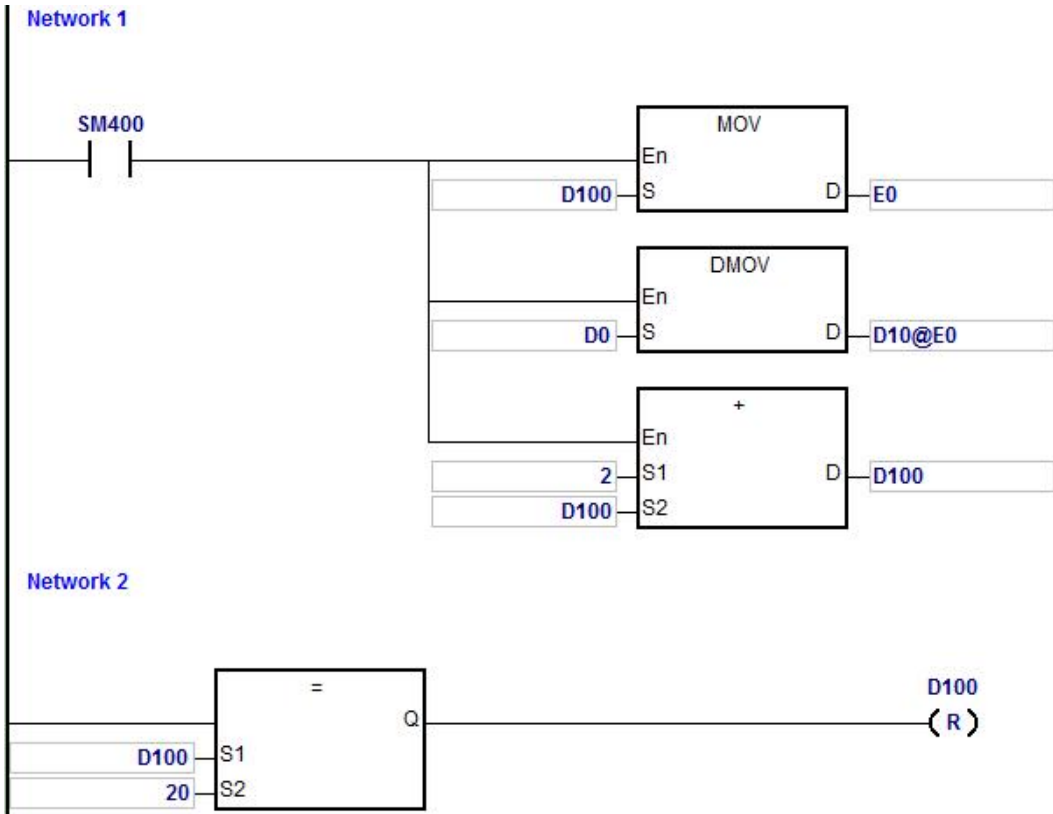
приклад



Зовнішнє переривання ініціюється наростаючим фронтом на X0.7.



Зовнішнє переривання ініціюється спадаючим фронтом на X0.7.



Додаткові зауваження

1. Коли M0 увімкнено, інструкція DCAP увімкнена. Коли зовнішнє переривання виникає в X0.7, інструкція фіксує значення в HC200 і зберігає його в (32-розрядному) D0.
2. Коли переривання зовнішнього входу запускається наростаючим фронтом один раз, інструкція змінює E0 на 0, встановлюючи D100, зберігає значення лічильника в D0 у D10, змінюючи E0, а значення в D100 становить 0+2.
3. Коли переривання зовнішнього входу запускається спадаючим фронтом один раз, інструкція змінює E0 на 2, встановлюючи D100, зберігає значення лічильника ($10+E0=12$) у D0 у D12, змінюючи E0, а значення в D100 дорівнює 0 +2. Коли значення в D100 дорівнює 20, D100 очищається до 0.
4. Якщо зовнішнє переривання викликано наростаючим фронтом і спадаючим фронтом п'ять разів відповідно, інструкція фіксує значення 10 разів і зберігає захоплені значення в D10, D12...D28.

1 -е зафіксоване значення=

D10 2 -е зафіксоване

значення= D12

...

...

10 -е зафіксоване

значення=D28 11 -е

зафіксоване значення=D10

API	Код інструкції				Операнд								функція				
1011		TMRM			S_1, S_2								16-бітний таймер (10 мс)				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1					O												
S_2								O				O	O	O			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1											●						
S_2		●				●											
										Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція			
										-		AS		-			

СИМВОЛ



S_1 : Номер таймера
 S_2 : Установка значения для таймера

Пояснення

Ця інструкція використовує 10 мс як одиницю часу.
Зверніться до пояснення інструкції TMRH (API 1002) для отримання детальної інформації.

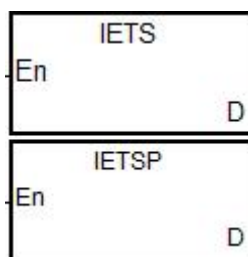
API	Інструкція			Операнд								опис					
1012		IETS	P	D								Початок вимірювання часу виконання інструкції					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
D								●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	—

СИМВОЛ



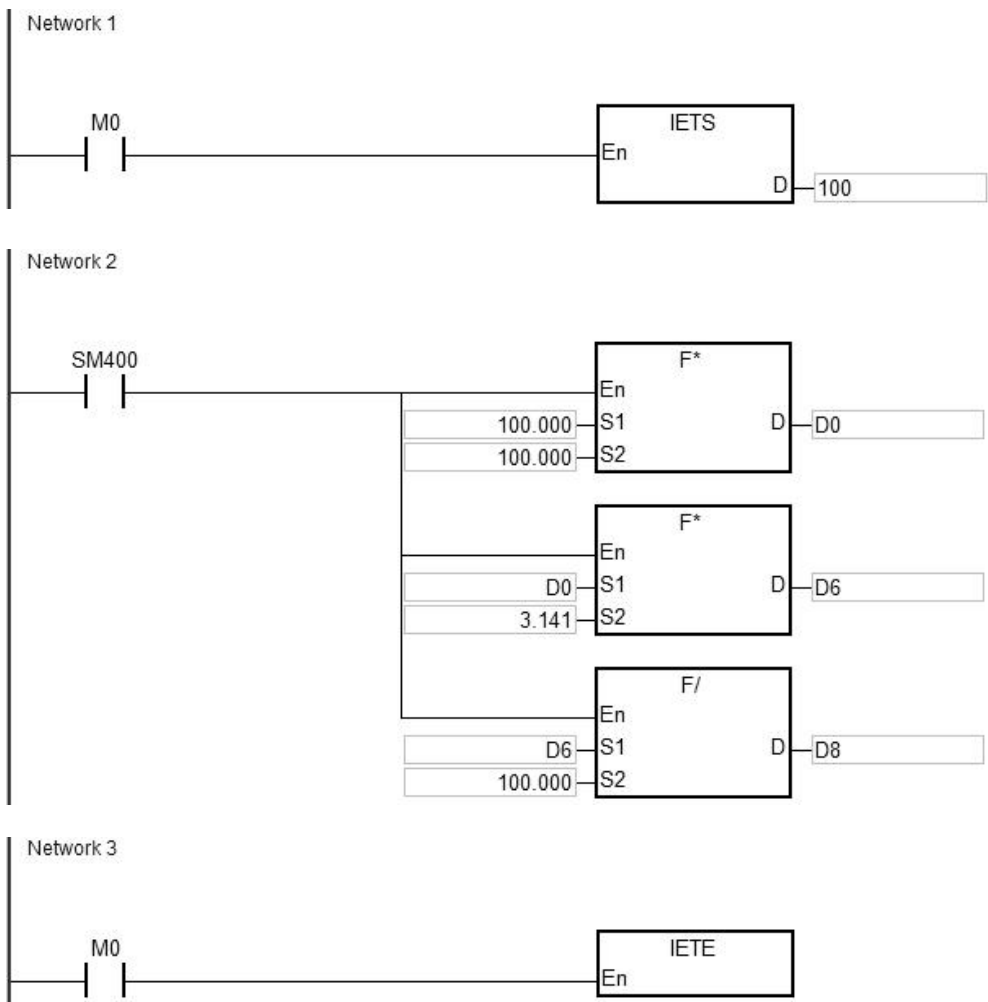
D : Результат вимірювання часу

Пояснення

- Інструкцію IETS потрібно використовувати разом з інструкцією API1013 IETE, щоб виміряти час для виконання інструкції в програмі ПЛК, яка призначена для виконання. Одиницею вимірювання часу є 1 мкс.
- Коли інструкцію IETS увімкнено, відлік часу починається негайно, доки також не буде виконано інструкцію IETE. Результат вимірювання зберігається в пристрої **D**.
- Результати вимірювання мінімального та максимального часу становлять 0us та 32767us відповідно. Після ввімкнення інструкції **IETS** ПЛК автоматично завершить вимірювання часу та збереже результат вимірювання в пристрої **D**, якщо жодна інструкція **IETE** не була відсканована, а сканування програми ПЛК досягає інструкції **END**.
- Для інструкцій IETS і IETE немає обмежень щодо кількості, написаної в програмі. Але лише один набір IETS і IETE можна вмикати щоразу, коли виконується сканування. Якщо IETS повторно вмикається для вимірювання часу, час увімкнення останньої інструкції IETS береться як початок вимірювання часу. Навпаки, якщо виконання кількох інструкцій IETE повністю завершено, ПЛК побачить точку, коли першу інструкцію IETE вимкнено, як кінцеву точку після завершення вимірювання часу.
- Інструкція IETS зазвичай використовується для вимірювання часу виконання програми ПЛК, наприклад, функціональних блоків програми обслуговування переривань тощо. Оскільки ресурс вимірювання часу ПЛК буде зайнято, коли функцію вимірювання часу увімкнено, ми пропонуємо дві інструкції видалити. після завершення вимірювання, щоб уникнути використання ресурсу ПЛК під час нормального виконання.

приклад

Розрахувати час виконання інструкції на основі формули для операції з числами з плаваючою комою, і результат операції зберігається в D100.



API	Інструкція			Операнд								опис					
1013		IETE	P	—								Кінець вимірювання часу виконання інструкції					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	—

СИМВОЛ

En	IETE
En	IETEP

Пояснення

Інструкцію IETE слід використовувати разом з інструкцією API1012 IETS. Для отримання додаткової інформації зверніться до пояснення інструкції API 1012.

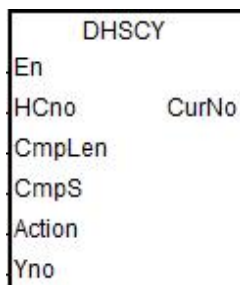
API	Інструкція			Операнд								опис					
1014	D	HSCY		HCno ~ CurNo								Циклічний підрахунок, порівняння та виведення					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
HCno							O									
CmpLen								●					O	O		
CmpS							●	●	●							
Action								●	●							
Yno								●	●							
CurNo								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
HCno												●	
CmpLen		●				●							
CmpS			●				●						
Action		●				●							
Yno		●				●							
CurNo		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



HCno : 32-розрядний вхідний лічильник

CmpLen : Довжина даних для порівняння

CmpS : Вихідне значення 32-розрядних даних для порівняння

Action : Дії, які необхідно взяти під час порівняння.

Yno : Номер вихідної точки Y

CurNo : Номер поточної групи, яка виконує порівняння

Пояснення

- Ця інструкція доступна для версії FW 1.06.10 або новішої.
- HCno**: якщо лічильник є загальним, порівняння не почнеться, навіть коли досягнуто часу для виконання порівняння. Порівняння починається лише після сканування інструкції. Якщо лічильник є високошвидкісним, порівняння починається, коли досягається час для виконання порівняння, на який не впливає цикл сканування ПЛК. Зверніться до інструкції DHSCS, щоб дізнатися більше про використання та обмеження високошвидкісного номера.
- CmpLen**: щоб вибрати набір груп для порівняння. Група набору порівнюється циклічно. Значення налаштування групи може бути від 4 до 32. Якщо значення налаштування перевищує прийнятний діапазон, ця інструкція не виконується. І SM0 буде переключено на ON. Код помилки SR0: 0x200B.

4. **CmpS** (імовірно встановлено як D100), **Action** (імовірно встановлено як D200), **Yno** (імовірно встановлено як D300), **CmpLen** (імовірно встановлено як 6), **CurNo** (як встановлено нижче).

CurNo (Номер поточної групи)	CmpS (Вихідне значення даних у 32-розрядному форматі для порівняння)	Дія (Дії, які необхідно взяти під час порівняння.)	Yno (Кількість вихідних точок Y)
1	D100 = 1000	D200 = 1 (набір)	D300 = 16#04 (Y0.4)
2	D102 = 2000	D201 = 1 (набір)	D301 = 16#05 (Y0.5)
3	D104 = 3000	D202 = 1 (набір)	D302 = 16#06 (Y0.6)
4	D106 = 4500	D203 = 0 (скидання)	D303 = 16#04 (Y0.4)
5	D108 = 5500	D204 = 0 (скидання)	D304 = 16#05 (Y0.5)
6	D110 = 6500	D205 = 0 (скидання)	D305 = 16#06 (Y0.6)

- **CmpS**: вихідне значення даних у вигляді 32-розрядного цілого числа. Це займає кілька місць у Dword у **CmLen**. Таким чином, пропонується оголосити масив у типі даних Dword. Порівнюване значення впорядковується в порядку зростання, якщо вихідне значення є додатним цілим числом. Порівнюване значення впорядковується в порядку спадання, якщо вихідне значення є цілими від'ємними числами. Якщо порівнюване значення організоване не за правилами, як зазначено, інструкція не виконується, а SMO буде переключено на ON. Код помилки SR0: 0x200B.
- **Action**: дія, яку необхідно виконати під час порівняння. Значення є 16-бітним цілим числом і займає одне слово в **CmLen**. Значення «1» тут означає встановлення вихідної точки на ON. Значення «0» тут означає встановлення вихідної точки на ВІМК. Використовуйте 16#8000, щоб очистити значення лічильника до 0 і перезавантажити циклічний підрахунок. Використовуйте тут лише значення 1, 0 або 16#8000. Пропонується оголосити масив у типі даних Word.
- **Yno**: номер вихідної точки Y, це значення має бути 16-бітним цілим числом. Це займає кілька місць у Word у **CmLen**. Формат тут шістнадцятковий. Наприклад, 16#0004 означає вихідну точку Y0.4, 16#0016 Y1.6, а 16#00AF Y10.15. Пропонується оголосити масив у типі даних Word.
- **CurNo**: поточний номер групи, яка виконує порівняння. З таблиці вище ви можете побачити, чи значення параметра в **CmLen** становить 1000, CurNo. Є 1.

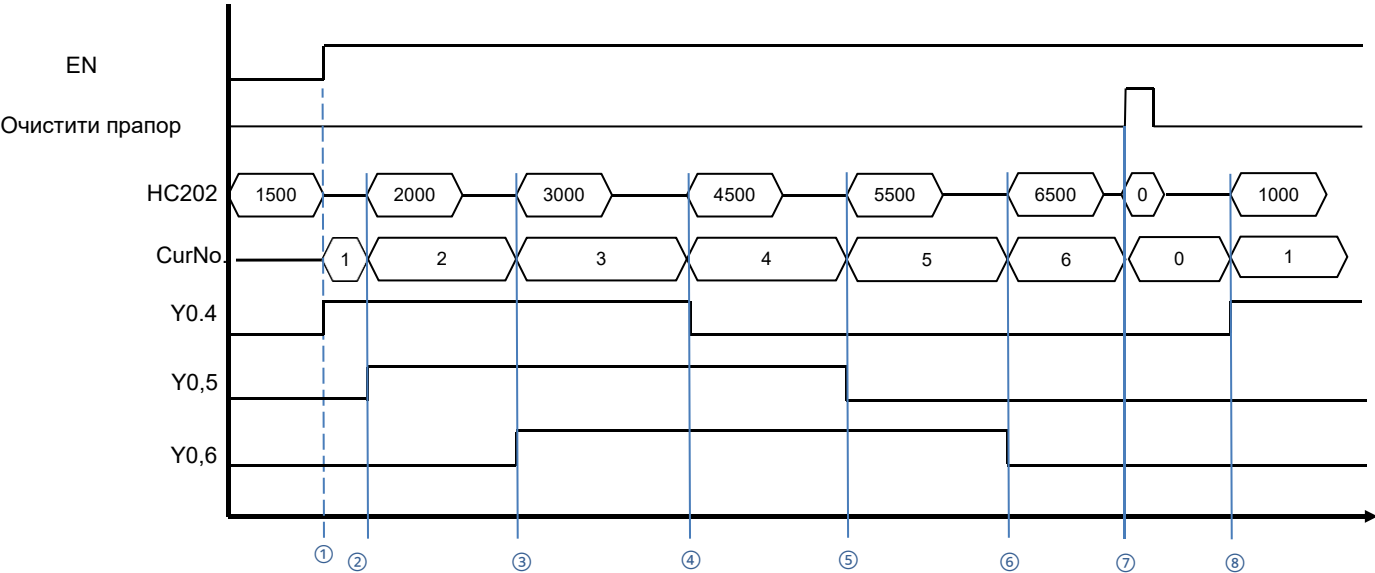
6

5. Якщо значення, яке ви використовуєте в **Yno**, не є вбудованими точками виводу ЦП, на виведення впливає цикл сканування ПЛК. Якщо вам потрібен негайний вихід, використовуйте вбудовані точки виведення процесора та апаратний високошвидкісний лічильник у **HCno**.
6. Оскільки ця інструкція призначена для циклічного порівняння та виведення, вам потрібно встановити спосіб очищення значення лічильника до 0, щоб встановити діапазон для циклічної операції. Рекомендується використовувати апаратний вхід, щоб встановити лічильник на 0. Або ви також можете використати налаштування, щоб очистити лічильник. Наприклад, вам потрібно 6 груп для виконання циклічного порівняння. Ви можете встановити значення параметра в **CmpLen** як 7, і ви матимете 7 груп **CruNo** (визначених **CmpS**). Потім використовуйте 16#8000 як значення налаштування в «Дії», щоб очистити HC. Перегляньте таблицю нижче для довідки.

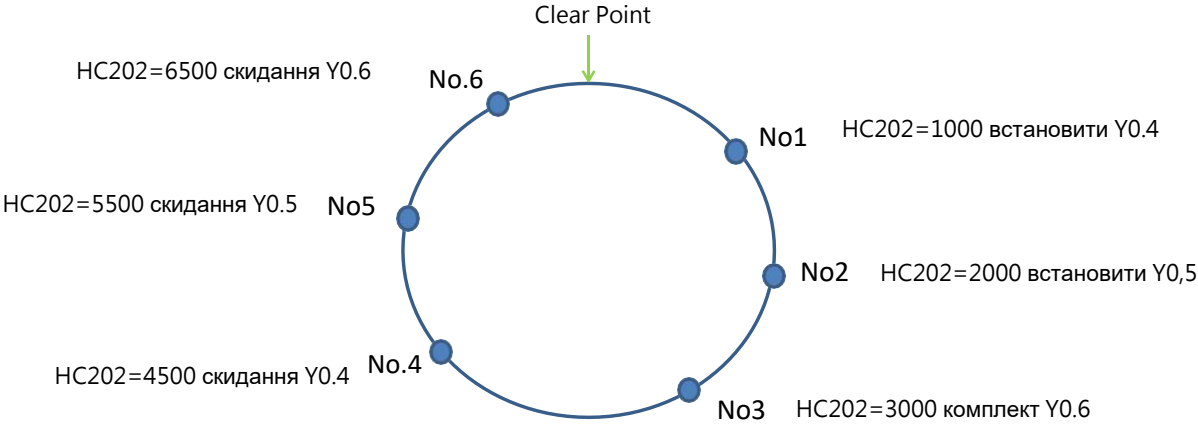
CurNo (Номер поточної групи)	CmpS (Вихідне значення даних у 32-розрядному форматі для порівняння)	Дія (Дії, які необхідно взяти під час порівняння.)	Yno (Кількість вихідних точок Y)
7 (CurNo = 0)	D112 = 8000	D206 = 16#8000 (Очистити HC)	D306 = 0 (жоден)

7. Коли інструкція виконується вперше, вона порівнює поточне значення лічильника зі значенням, встановленим у **CmpS**, і відповідно встановлює значення в **CurNo**. Див. приклад таблиці під номером 4. Якщо поточне значення лічильника дорівнює 2000, це відповідає значенню (D102 = 2000), встановленому в **CmpS**, а відповідне значення **CurNo** дорівнює 2. З цього ви знаєте, що Y0.5 увімкнено.
8. Дивіться приклад на хронологічній ілюстрації нижче.
- ① → Інструкція спочатку виконується, порівнюючи поточне значення лічильника (1500) і значення, встановлене в **CmpS** (HC202). І встановіть значення **CurNo** на 1, а вихід Y0.4 – на ON відповідно.
 - ② → Коли поточне значення лічильника досягає 2000, **CurNo** = 2 і Y0.5 увімкнено.
 - ③ → Коли поточне значення лічильника досягає 3000, **CurNo** = 3 і Y0.6 увімкнено.

- ④ → Коли поточне значення лічильника досягає 4500, **CurNo** = 4 і Y0.4 OFF.
- ⑤ → Коли поточне значення лічильника досягає 5500, **CurNo** = 5 і Y0.5 OFF.
- ⑥ → Коли поточне значення лічильника досягає 6500, **CurNo** = 6 і Y0.6 OFF.
- ⑦ → Зовнішнє переривання запускається для очищення. **CurNo** = 0 і HC202 = 0.
- ⑧ → Коли поточне значення лічильника досягає 1000, **CurNo** = 1 і Y0.4 увімкнено.



Циклічне порівняння лише для ілюстрації.



приклад:

1. Визначте наступні параметри, як показано нижче.

CurNo (Номер поточної групи)	CmpS (Вихідне значення даних у 32-розрядному форматі для порівняння)	Дія (Дії, які необхідно взяти під час порівняння.)	Yno (Кількість вихідних точок Y)
1	D100 = 1000	D200 = 1 (набір)	D300 = 16#04 (Y0.4)
2	D102 = 2000	D201 = 1 (набір)	D301 = 16#05 (Y0.5)
3	D104 = 3000	D202 = 1 (набір)	D302 = 16#06 (Y0.6)
4	D106 = 4500	D203 = 0 (скидання)	D303 = 16#04 (Y0.4)
5	D108 = 5500	D204 = 0 (скидання)	D304 = 16#05 (Y0.5)
6	D110 = 6500	D205 = 0 (скидання)	D305 = 16#06 (Y0.6)

2. Коли M0 увімкнено, інструкція виконується.

① → Інструкція спочатку виконується, порівнюючи поточне значення лічильника та значення, встановлене в **CmpS** (HC202).

І встановить значення **CurNo** на 1, а вихід Y0.4 – на ON відповідно.

② → Коли поточне значення лічильника досягає 2000, **CurNo** = 2 і Y0.5 увімкнено.

③ → Коли поточне значення лічильника досягає 3000, **CurNo** = 3 і Y0.6 увімкнено.

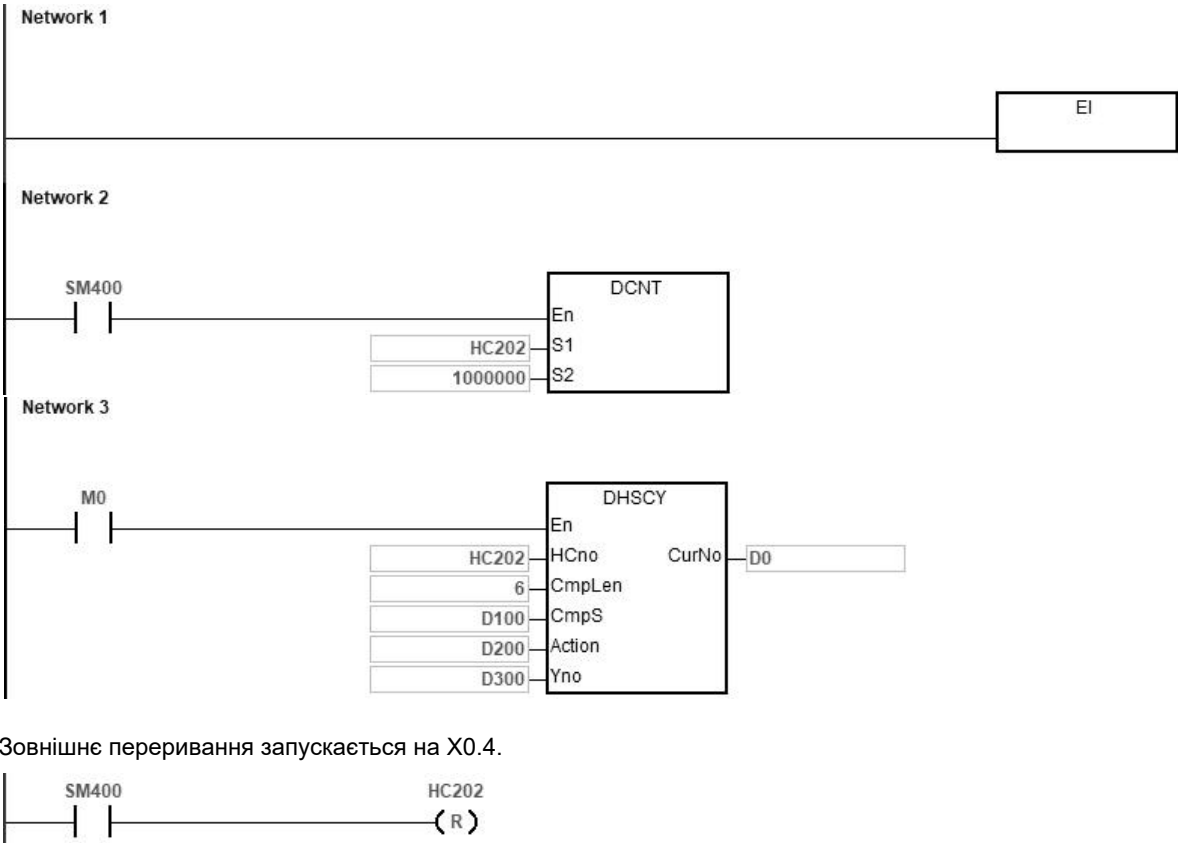
④ → Коли поточне значення лічильника досягає 4500, **CurNo** = 4 і Y0.4 OFF.

⑤ → Коли поточне значення лічильника досягає 5500, **CurNo** = 5 і Y0.5 OFF.

⑥ → Коли поточне значення лічильника досягає 6500, **CurNo** = 6 і Y0.6 OFF.

⑦ → Зовнішнє переривання запускається для очищення. **CurNo** = 0 і HC202 = 0.

⑧ → Коли поточне значення лічильника досягає 1000, **CurNo** = 1 і Y0.4 увімкнено.

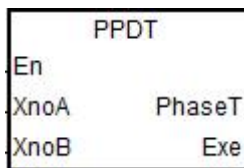


API	Інструкція				Операнд						опис						
1015		PPDT			HCNo ~ CurNo						Виявлення різниці в часі між двома фазами						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
XnoA	○																
XnoB	○																
PhaseT								●									
Exe		●	●	●													

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
XnoA	●												
XnoB	●												
PhaseT			●				●						
Exe	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ



XnoA : Вхід фази A

XnoB : Вхід фази B

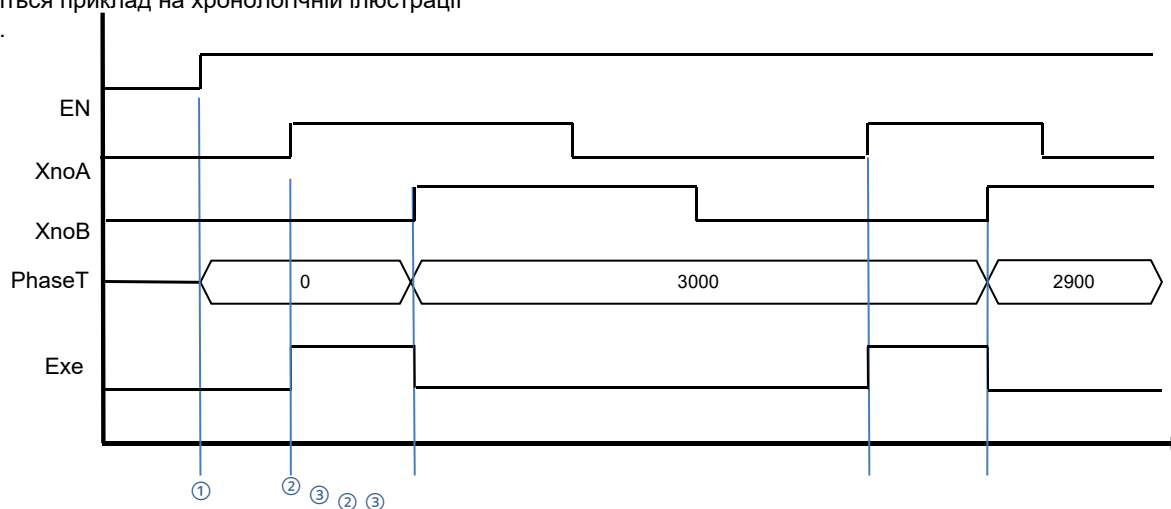
PhaseT : Різниця в часі між двома фазами (32-розрядне ціле число)

Exe : Прапор часу

Пояснення

- Ця інструкція доступна для версії FW 1.06.10 або новішої.
- Ви можете використовувати лише вбудовані в ЦП точки введення від X0.0 до X0.11 для XnoA та XnoB. І кількість вхідних точок НЕ можна дублювати. Якщо використані вхідні точки не від X0.0 до X0.11 або кількість вхідних точок дублюється, інструкція не виконується. І SM0 буде переключено на ON. Код помилки SR0: 16#2003.
- Коли **XnoA** спрацює по наростаючому фронту, таймер негайно починає відраховувати час, не впливаючи на цикл сканування ПЛК. Коли **XnoB** спрацює по наростаючому фронту, таймер негайно припиняє вимірювання часу, не впливаючи на цикл сканування ПЛК. І час, який вимірюється (одиниця вимірювання: 1 мкс), буде збережено у **PhaseT**, коли ця інструкція сканується ПЛК.
- Коли таймер запускається і ця інструкція сканується для виконання, прапор **Exe** буде ON. Коли таймер зупиняється і ця інструкція сканується для виконання, прапор **Exe** буде OFF.
- Щоразу, коли цю інструкцію активовано вперше, її початковою дією є очищення значення у **PhaseT** і встановлення позначки **Exe** на ВІМК.
- XnoA** завжди передуює **XnoB**. Навіть якщо **XnoB** запускається по наростаючому фронту перед тим, як запускається **XnoA**. Якщо вам потрібно **XnoB** вимірювався перед **XnoA**, вам потрібно повторно визначити **XnoA**, **XnoB** і точки введення в них.
- Немає обмежень щодо кількості виконання інструкції. І використовувати таку ж кількість вхідних точок теж не проблема. Але оскільки ПЛК використовує номер точки введення для **XnoA** як номер внутрішнього таймера. НЕ рекомендується використовувати два однакові номери точки введення для **XnoA** одночасно. Або система заплутається. І результат таймера буде спотворений.

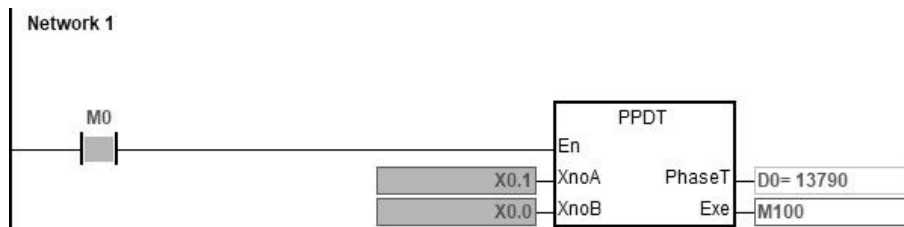
8. Дивіться приклад на хронологічній ілюстрації нижче.



- ① → Щоразу, коли цю інструкцію активовано вперше, вона очищає значення у **PhaseT** і встановлює прапор **Exe** на ВИМК.
- ② → Коли **XnoA** спрацьовує на підвищення, щоб почати відлік часу, прапор **Exe** буде ON.
- ③ → Коли **XnoB** спрацьовує на зростання, щоб зупинити хронометраж, прапор **Exe** буде OFF та зберігає хронометрований результат у **PhaseT**.

приклад

Коли M0 увімкнено, інструкція виконується. Ця інструкція обчислює різницю часу фази між X0.1 і X0.0 і зберігає результат у D0. (Тут розрахунковий результат становить 13790 нас.)

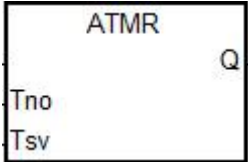


API	Код інструкції				Операнд				функція								
1016		ATMR			Tno, Tsv				16-бітний контактний таймер (одиниця: 100 мс)								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Tno					O												
Tsv								O				O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Tno											●		
Tsv		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



Tno : Номер таймера

Tsv : Установка значения для таймера

Пояснення

- 1. Інструкція доступна для мікропрограми ПЛК версії 1.12.00 і пізніших.
- 2. Інструкція ATMR використовує 100 мс як одиницю часу в таймері. Зверніться до пояснення до інструкції ATMRH (API 1017), щоб дізнатися більше.

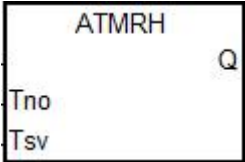
API	Код інструкції				Операнд					функція							
1017		ATMRH			Tno, Tsv					16-бітний контактний таймер (одиниця: 1 мс)							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Tno					O											
Tsv								O				O	O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Tno											●		
Tsv		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



Tno : Номер таймера
Tsv : Установка значення для таймера

Пояснення

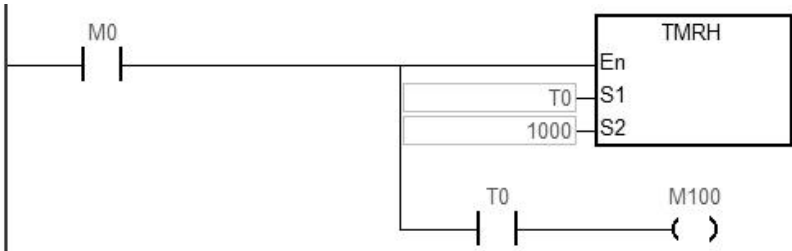
- 6
- Інструкція доступна для мікропрограми ПЛК версії 1.12.00 або новішої.
 - Інструкція ATMRH така ж, як комбінація інструкцій AND і TMRH.
Якщо умовний контакт виконано, котушка для вказаного таймера увімкнена, і таймер починає відлік часу. Коли задане значення часу досягнуто (значення часу >= значення налаштування), його контакт діятиме, як показано в наступній таблиці. Якщо умова для дії контакту I виконується, але умовний контакт не виконується, інструкція ATMRH автоматично очищає значення часу в таймері.

NO (нормально відкритий) контакт	Безперервність
NC (нормально замкнутий) контакт	Розривність

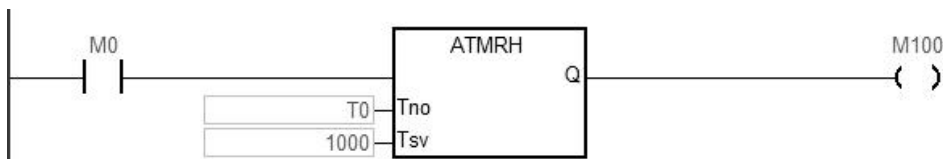
приклад

Програма виконується наступним чином:
Коли нормально відкритий контакт M0 увімкнено, таймер T0 починає відраховувати час, коли значення часу більше або дорівнює K1000, нормально відкритий контакт M100 увімкнено.

Сходова діаграма (з використанням інструкції TMRH)



Сходова діаграма (з використанням інструкції ATMRH)

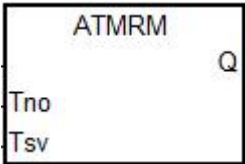


API	Код інструкції				Операнд				функція								
1018		ATMRM			Tno, Tsv				16-бітний контактний таймер (одиниця: 10 мс)								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Tno					O												
Tsv								O				O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Tno											●		
Tsv		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- Tno : Номер таймера
- Tsv : Установка значення для таймера

Пояснення

-
- Інструкція доступна для мікропрограми ПЛК версії 1.12.00 або новішої.
- Інструкція ATMRM використовує 10 мс як одиницю часу в таймері. Докладніше див. у поясненні інструкції ATMRH (API 1017).

6.12 Інструкції по зміні

6.12.1 Список інструкцій щодо зміни

У наведеній нижче таблиці наведено інструкції Shift, які розглядаються в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1100	SFTR	–	✓	Зсув станів пристроїв вправо
1101	SFTL	–	✓	Зсув станів пристроїв вліво
1102	WSFR	DWSFR	✓	Зсув даних в апаратах Word вправо
1103	WSFL	DWSFL	✓	Зсув даних в апаратах Word вліво
1104	SFWR	DSFWR	✓	Зсув даних і запис їх у текстовий пристрій
1105	SFRD	DSFRD	✓	Зсув даних і зчитування їх із пристрою Word
1106	SFPO	DSFPO	✓	Читання останніх даних зі списку даних
1107	SFDEL	DSFDEL	✓	Видалення даних зі списку даних
1108	SFINS	DSFINS	✓	Вставлення даних у список даних
1109	MBS	–	✓	Зміщення бітів матриці
1110	SFR	DSFR	✓	Зсув значень розрядів у регістрах на n розрядів вправо
1111	SFL	DSFL	✓	Зсув значень розрядів у регістрах на n розрядів вліво
1112	BSFR	–	✓	Зрушення станів n-розрядних пристроїв на один біт вправо
1113	BSFL	–	✓	Зрушення станів n-розрядних пристроїв на один біт вліво
1114	NSFR	DNSFR	✓	Зсув n регістрів вправо
1115	NSFL	DNSFL	✓	Зсув n регістрів вліво

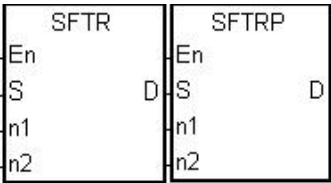
6.12.2 Пояснення інструкцій щодо зміни

API	Код інструкції				Операнд								функція				
1100		SFTR	P		S, D, n ₁ , n ₂								Зсув станів пристроїв вправо				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●	●	●				●									
D		●	●	●				●									
n ₁								●	●		○	○	○	○			
n ₂								●	●		○	○	○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
D	●												
n ₁		●				●							
n ₂		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



- S : Перший пристрій, на якому змінено значення
- D : Перший пристрій, на якому змінено значення
- n₁ : Довжина даних, які потрібно перемістити
- n₂ : Кількість бітів у групі

Пояснення

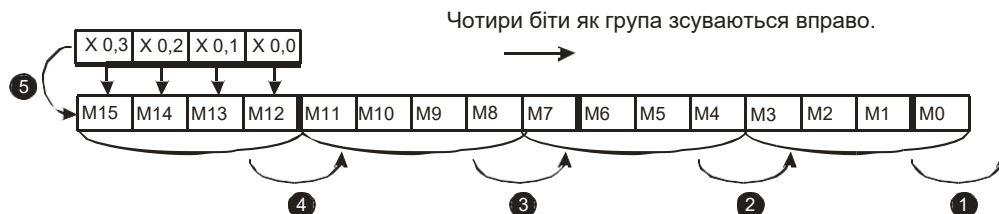
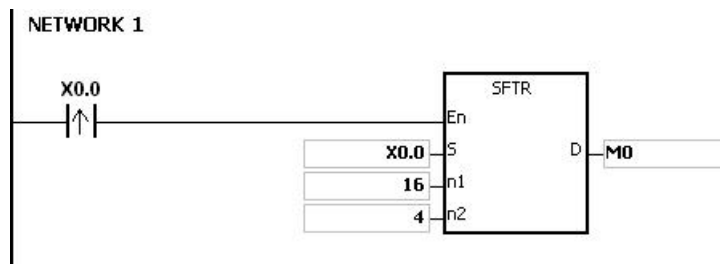
- Ця інструкція розділяє стани n₁ бітових пристроїв, починаючи з D, на групи (n₂ біт у групі) і зсуває ці групи праворуч. Потім ця інструкція зміщує стани n₂-розрядних пристроїв, починаючи з S, на пристрої, починаючи з D, щоб заповнити вакансію.
- Загалом використовується імпульсна інструкція SFTRP.
- Операнд n₁ має бути між 1–1024. Операнд n₂ має бути між 1– n₁. Для мікропрограми V1.12.00 або новішої версії операнд n₁ може бути між 1–4096.

Приклад 1

1. Коли X0.0 перемикається з OFF на ON, інструкція розділяє стани шістнадцятибітових пристроїв, починаючи з M0 до M15, на групи (по чотири біти в групі) і зсуває ці групи вправо.

2. Зсув станів бітових пристроїв вправо під час сканування показано нижче.

- ❶ M3-M0 → Несуть
- ❷ M7-M4 → M3-M0
- ❸ M11-M8 → M7-M4
- ❹ M15-M12 → M11-M8
- ❺ X0,3-X0,0 → M15-M12

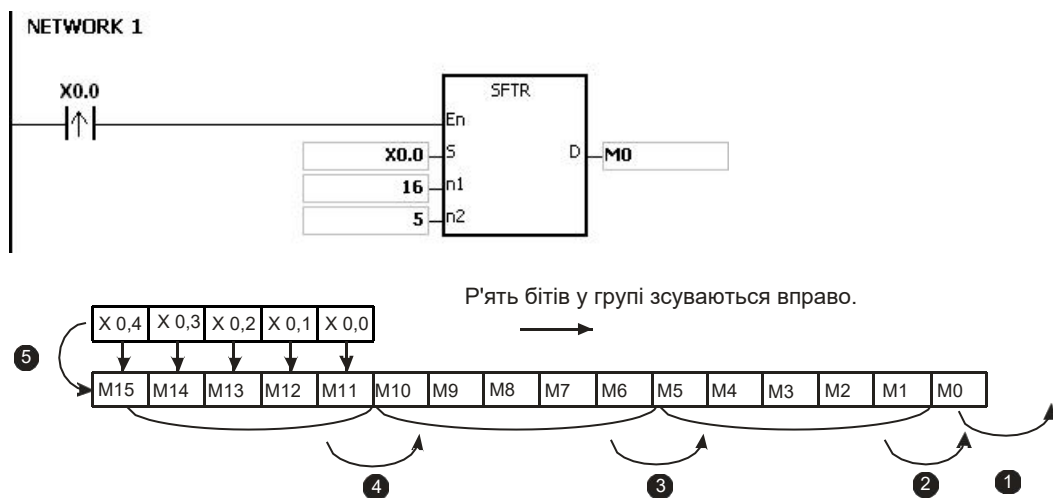
**Приклад 2**

1. Коли X0.0 перемикається з OFF на ON, інструкція розділяє стани шістнадцятибітових пристроїв, починаючи з M0 до M15, на групи (п'ять бітів як група) і зміщує ці групи вправо.

2. Зсув станів бітових пристроїв вправо під час сканування показано нижче.

- ❶ M0 → Несуть
- ❷ M5 → M0
- ❸ M10-M6 → M5-M1
- ❹ M15-M11 → M10-M6

⑤ X0,4-X0,0 → M15-M11



Додаткові зауваження

1. Якщо $S + n_2 - 1$ або $D + n_1 - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n_1 не між 1–1024, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Якщо n_2 не між 1– n_1 , інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API		Код інструкції			Операнд						функція					
1101			SFTL	P	S, D, n ₁ , n ₂						Зсув станів пристроїв вліво					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●	●	●				●								
D		●	●	●				●								
n ₁								●	●		○	○	○	○		
n ₂								●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
D	●												
n ₁		●				●							
n ₂		●				●							

Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
AS		AS	-

СИМВОЛ

SFTL				SFTLP			
En				En			
S		D	S	D			
n1			n1				
n2			n2				

- S** : Перший пристрій, на якому змінено значення
- D** : Перший пристрій, на якому змінено значення
- n₁** : Довжина даних, які потрібно перемістити
- n₂** : Кількість бітів у групі

Пояснення

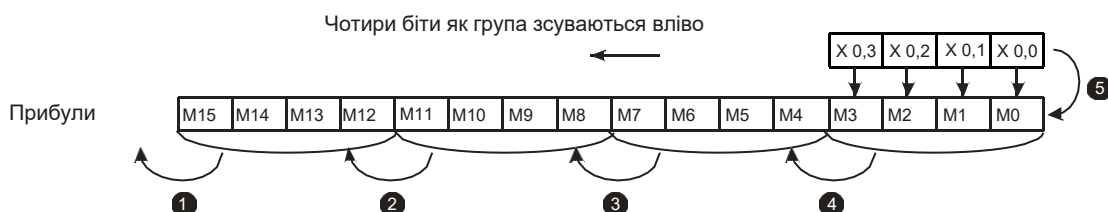
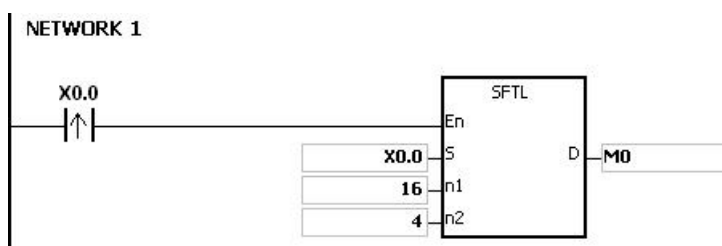
- Ця інструкція розділяє стани **n₁** бітових пристроїв, починаючи з **D**, на групи (**n₂** біт у групі) і зсуває ці групи вліво. Потім ця інструкція зміщує стани **n₂**-розрядних пристроїв, починаючи з **S**, на пристрої, починаючи з **D**, щоб заповнити вакансію.
- Загалом використовується імпульсна інструкція SFTLP.
- Операнд **n₁** має бути між 1–1024. Операнд **n₂** має бути між 1–**n₁**. Для мікропрограми версії 1.12.00 або пізнішої версії операнд **n₁** може бути між 1–4096.

Приклад 1

1. Коли X0.0 перемикається з OFF на ON, інструкція розділяє стани шістнадцятирозрядних пристроїв, починаючи з M0 до M15, на групи (по чотири біти в групі) і зсуває ці групи вліво.

2. Зсув станів бітових пристроїв вліво під час сканування показано нижче.

- ❶ M15-M12 → Несуть
- ❷ M11-M8 → M15-M12
- ❸ M7-M4 → M11-M8
- ❹ M3-M0 → M7-M4
- ❺ X0,3-X0,0 → M3-M0



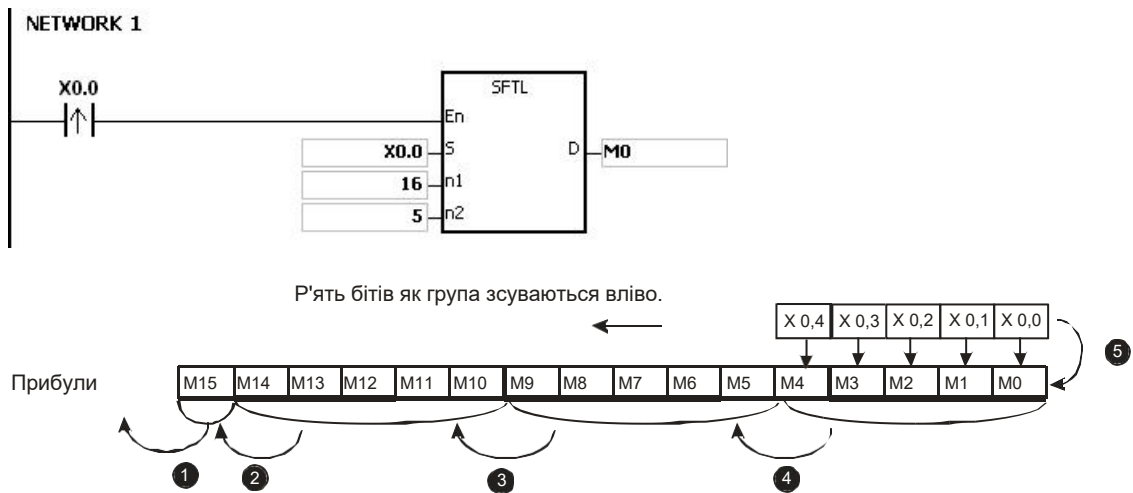
Приклад 2

1. Коли X0.0 перемикається з OFF на ON, інструкція розділяє стани шістнадцятибітових пристроїв, починаючи з M0 до M15, на групи (п'ять бітів у групі) і зсуває ці групи вліво.

2. Зсув станів бітових пристроїв вліво під час сканування показано нижче.

- ❶ M15 → Несуть
- ❷ M10 → M15
- ❸ M9-M5 → M14-M10

- ④ M4-M0 → M9-M5
- ⑤ X0,4-X0,0 → M4-M0



Додаткові зауваження

1. Якщо $S + n_2 - 1$ або $D + n_1 - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n_1 не між 1–1024, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Якщо n_2 не між 1– n_1 , інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.

API	Код інструкції			Операнд							функція						
1102	D	WSFR	P	S, D, n ₁ , n ₂							Зсув даних в апаратах Word вправо						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n ₁								●	●		○	○	○	○		
n ₂								●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n ₁		●	●			●	●						
n ₂		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

WSFR		WSFRP	
En		En	
S	D	S	D
n1		n1	
n2		n2	
DWSFR		DWSFRP	
En		En	
S	D	S	D
n1		n1	
n2		n2	

- S** : Перший пристрій, на якому змінено значення
- D** : Перший пристрій, на якому змінено значення
- n₁** : Довжина даних, які потрібно перемістити
- n₂** : Кількість бітів у групі

Пояснення

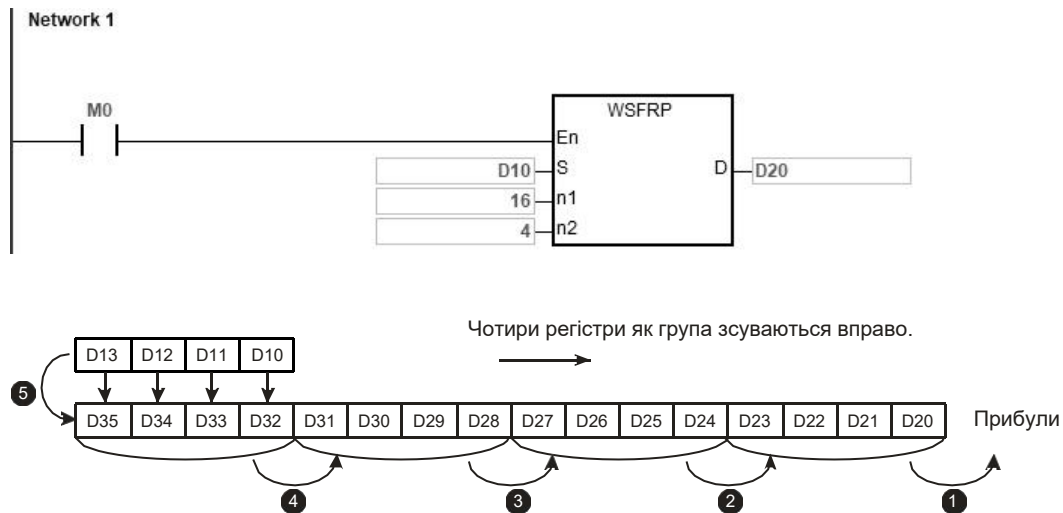
- Ця інструкція розбиває дані в пристроях з **n₁** слів, починаючи з **D**, на групи (**n₂** слова в групі), і зміщує ці групи вправо. Ця інструкція потім зміщує дані в **n₂** пристроях слова, починаючи з **S**, на пристрої, починаючи з **D**, щоб заповнити вакансію.
- Загалом використовується імпульсна інструкція WSFRP.
- Для 16-бітної інструкції операнд **n₁** має бути між 1–512. Для 32-розрядної інструкції операнд **n₁** має бути між 1–256. Для застосування 32-розрядних інструкцій вам потрібно використовувати ISPSOft V3.09.07 або пізнішої версії та AS PLC CPU FW V1.08 або пізнішої версії.
- Значення в операнді **n₂** має бути між 1–**n₁**.

Приклад 1

1. Коли M0 перемикається з OFF на ON, інструкція розділяє дані в пристроях з шістнадцятьма словами, починаючи з D20 до D35, на групи (по чотири слова в групі) і зміщує ці групи вправо.

2. Нижче показано зміщення даних у пристрої Word вправо під час сканування (M0 перемикається з ВІМК. на УВІМК.).

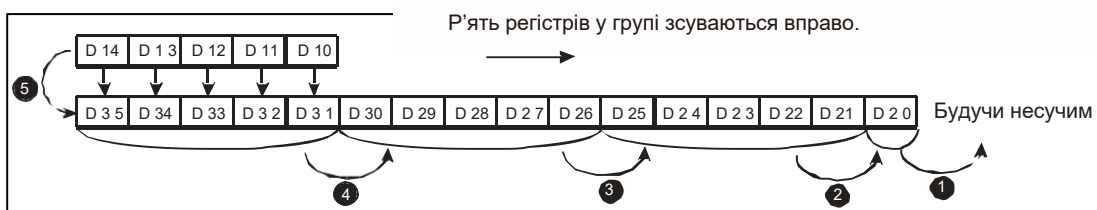
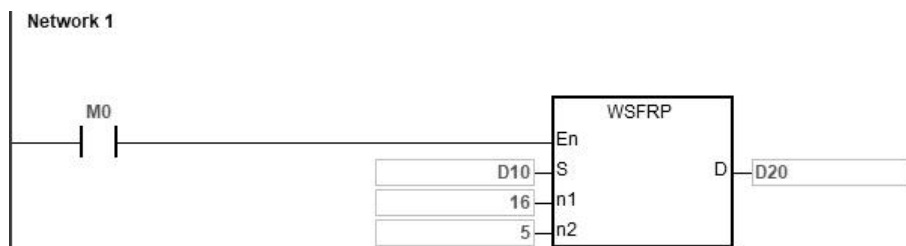
- ❶ D23-D20 → Несуть
- ❷ D27-D24 → D23-D20
- ❸ D31-D28 → D27-D24
- ❹ D35-D32 → D31-D28
- ❺ D13-D10 → D35-D32 завершено

**Приклад 2**

1. Коли M0 перемикається з OFF на ON, інструкція ділить дані в шістнадцяти слівних пристроях, починаючи з D20 до D35, на групи (n'ять слів у групі) і зміщує ці групи вправо.

2. Нижче показано зміщення даних у пристрої Word вправо під час сканування (M0 перемикається з ВІМК. на УВІМК.).

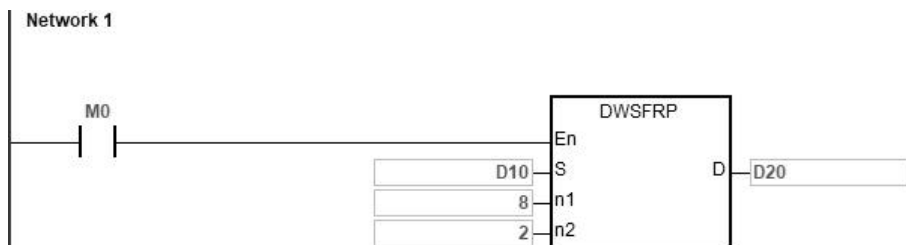
- ❶ D20 → Несуть
- ❷ D25 → D20
- ❸ D30-D26 → D25-D21
- ❹ D35-D31 → D30-D26
- ❺ D14-D10 → D35-D31 завершено



Приклад 3

1. Коли M0 перемикається з OFF на ON, інструкція розділяє дані у восьми пристроях слів, починаючи з D20 до D35, на групи (по два слова в групі) і зміщує ці групи вправо.
2. Нижче показано зміщення даних у пристрої Word вправо під час сканування (M0 перемикається з ВИМК. на УВИМК.).

- ➊ D23-D20 → Несуть
- ➋ D27-D24 → D23-D20
- ➌ D31-D28 → D27-D24
- ➍ D35-D32 → D31-D28
- ➎ D13-D10 → D35-D32 завершено





Додаткові зауваження

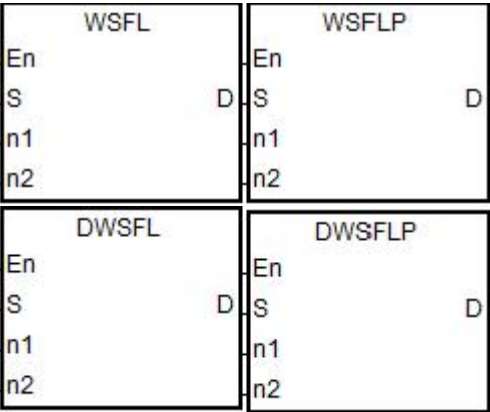
1. Якщо $S + n_2 - 1$ або $D + n_1 - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Значення n_1 16-бітної інструкції має бути між 1–512, а n_{32} -бітної інструкції має бути менше 256. Якщо ні, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 це 16#200B.
3. Значення n_2 має бути між 1– n_1 . Якщо ні, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд							функція						
1103	D	WSFL	P	S, D, n ₁ , n ₂							Зсув даних в апаратах Word вліво						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●	●								
D		●			●	●		●									
n ₁								●	●		○	○	○	○			
n ₂								●	●		○	○	○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n ₁		●	●			●	●						
n ₂		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



- S** : Перший пристрій, у якому значення зсувається
- D** : Перший пристрій, у якому значення зсувається
- n₁** : Довжина даних, які потрібно перемістити
- n₂** : Кількість бітів у групі

Пояснення

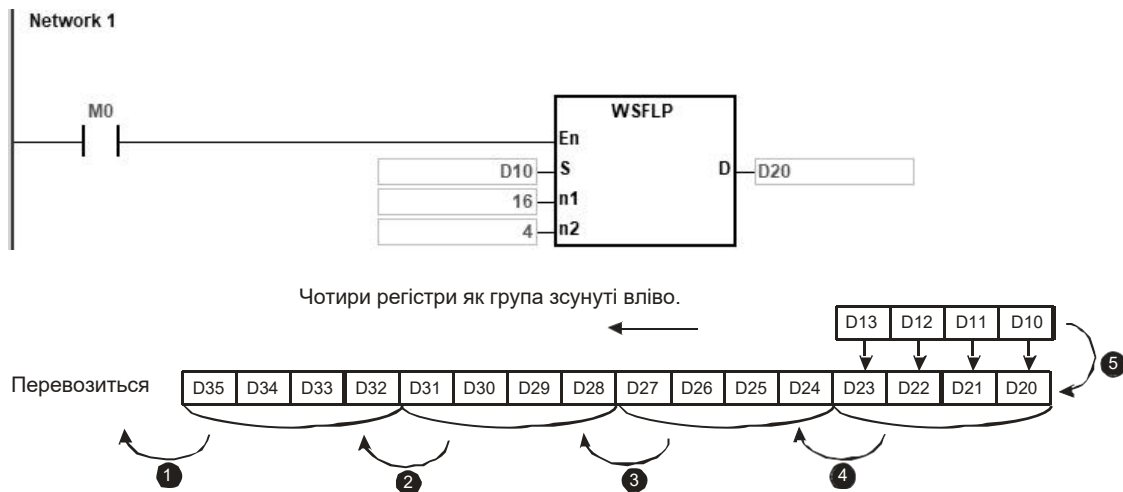
- Ця інструкція розбиває дані в пристроях з **n₁** слів, починаючи з **D**, на групи (**n₂** слова в групі), і зсуває ці групи ліворуч. Ця інструкція потім зміщує дані в **n₂** пристроях слова, починаючи з **S**, на пристрої, починаючи з **D**, щоб заповнити вакансію.
- Загалом використовується імпульсна інструкція WSFLP.
- Для 16-бітної інструкції значення в операнді **n₁** має бути між 1–512. Для 32-розрядної інструкції значення в операнді **n₁** має бути між 1–256. Для застосування 32-розрядних інструкцій вам потрібно використовувати ISPSOft V3.09.07 або пізнішої версії та AS PLC CPU FW V1.08 або пізнішої версії.
- Операнд **n₂** має бути між 1–**n₁**.

Приклад 1

1. Коли M0 перемикається з OFF на ON, інструкція розділяє дані в шістнадцяти слівних пристроях, починаючи з D20 до D35, на групи (по чотири слова в групі) і зміщує ці групи вліво.

2. Нижче показано зміщення даних у пристрої Word вліво під час сканування (M0 перемикається з ВІМК. на УВІМК.).

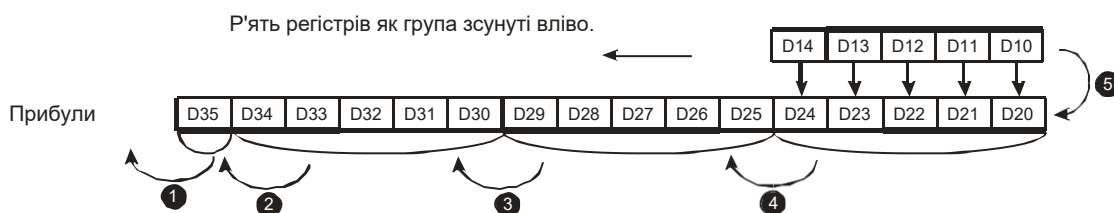
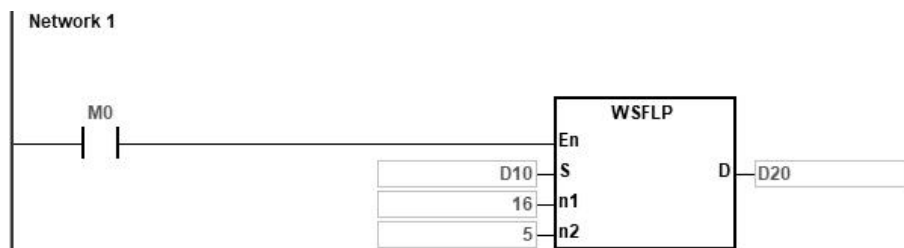
- ❶ D35-D32 → Несуть
- ❷ D31-D28 → D35-D32
- ❸ D27-D24 → D31-D28
- ❹ D23-D20 → D27-D24
- ❺ D13-D10 → D23-D20 завершено

**6****Приклад 2**

1. Коли M0 перемикається з OFF на ON, інструкція ділить дані в шістнадцяти слівних пристроях, починаючи з D20 до D35, на групи (п'ять слів у групі) і зміщує ці групи вліво.

2. Нижче показано зміщення даних у пристрої Word вліво під час сканування (M0 перемикається з ВІМК. на УВІМК.).

- ❶ D35 → Несуть
- ❷ D30 → D35
- ❸ D29-D25 → D34-D30
- ❹ D24-D20 → D29-D25
- ❺ D14-D10 → D24-D20 завершено



Додаткові зауваження

1. Якщо $S + n_2 - 1$ або $D + n_1 - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Значення n_1 16-бітної інструкції має бути між 1–512, а n_{32} -бітної інструкції має бути менше 256. Якщо ні, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 це 16#200B.
3. Значення n_2 має бути між 1– n_1 . Якщо ні, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 16#200B.

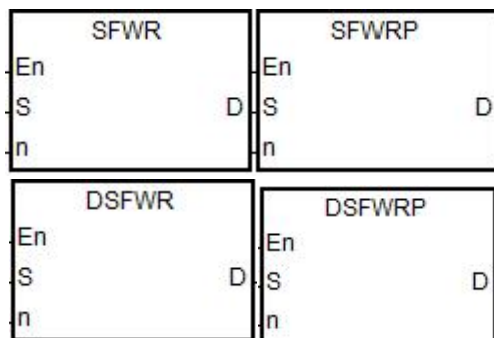
API	Код інструкції			Операнд						функція							
1104	D	SFWR	P	S, D, n						Зсув даних і запис їх у текстовий пристрій							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●		O	O	O	O		
D		●			●	●		●								
n	●	●			●	●		●	●		O	O	O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



S : Пристрій, на який передаються дані

D : Перший пристрій

n : Довжина даних

Пояснення

1. Ця інструкція визначає дані в **n**-слові пристроїв, починаючи з пристрою, визначеного **D** як тип списку першим увійшов, першим вийшов, і бере пристрій, вказаний **D**, як покажчик. Ця інструкція збільшує значення покажчика на одиницю та записує дані з пристрою, указанного **S**, у пристрій, вказаний покажчиком. Коли значення вказівника більше або дорівнює **n** - 1, інструкція припиняє запис даних і встановлює прапор переносу SM602 у положення ON.

2. Загалом використовується імпульсна інструкція SFWRP.

3. Ця інструкція широко використовується як функція PUSH. Якщо вам потрібна функція POP, зверніться до інструкцій SFRD і SFPO для отримання додаткової інформації.

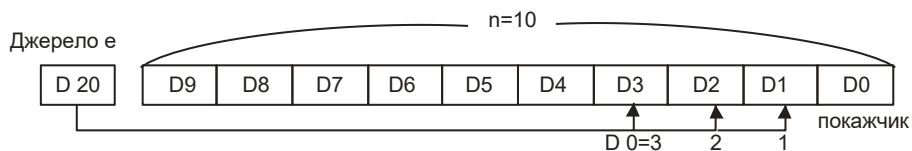
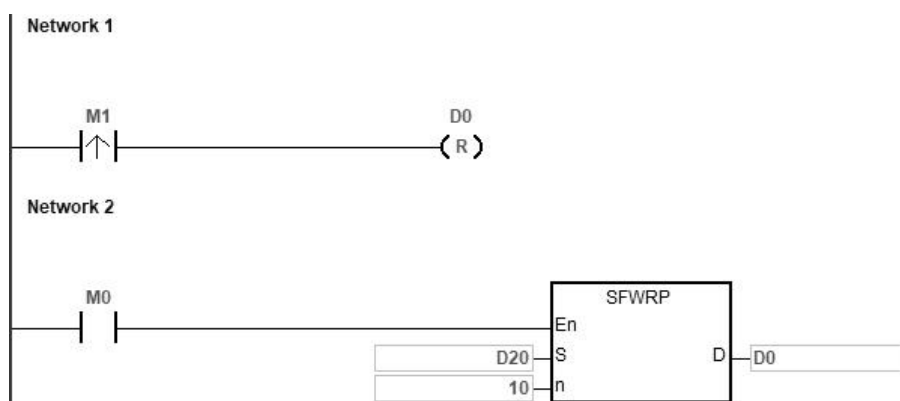
4. Для 16-бітної інструкції значення в операнді **n** має бути між 2–512. Для 32-розрядної інструкції значення в операнді **n** має бути між 2–256. Для застосування 32-розрядних інструкцій вам потрібно використовувати ISPSOft V3.09.07 або пізнішої версії та AS PLC CPU FW V1.08 або пізнішої версії.

Приклад 1

1. Інструкція спочатку очищає значення показчика D0 до 0. Коли M0 перемикається з OFF на ON, інструкція записує дані з D20 у D1 і збільшує значення в D0 до 1. Коли M0 знову перемикається з OFF на ON, інструкція записує дані з D20 у D2 та збільшує значення в D0 до 2.

2. Інструкція зсуває та записує дані в слово пристрій, як показано нижче.

- Дані в D20 записуються в D1.
- Значення в D0 стає 1.

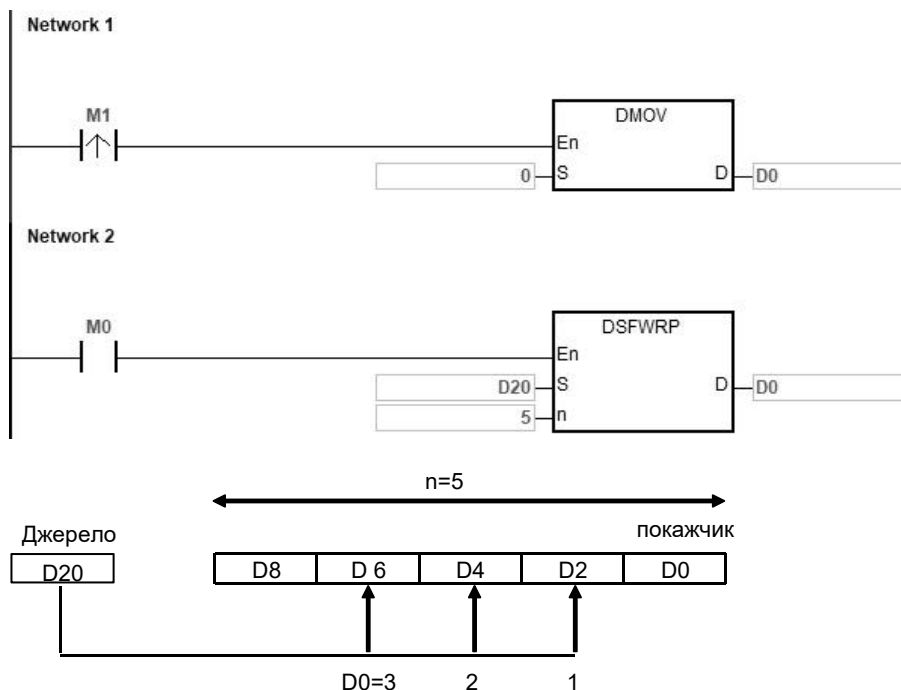


Приклад 2

1. Інструкція спочатку очищає значення покажчика D0 до 0. Коли M0 перемикається з OFF на ON, інструкція записує дані з D20/D21 у D2/D3 та збільшує значення в D0/D1 до 1. Коли M0 знову перемикається з OFF на ON, інструкція записує дані в D20/ D21 до D4/D5 і збільшує значення в D0/D1 до 2.

2. Інструкція зсуває та записує дані в слово пристрій, як показано нижче.

- Дані в D20/21 записуються в D2/D3.
- Значення в D0/D1 стає 1.

**Додаткові зауваження**

1. Якщо значення в **D** менше 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо **D + n -1** перевищує діапазон пристрою, інструкція не виконується. SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Для 16-бітної інструкції значення в операнді **n** має бути між 2–512. Для 32-розрядної інструкції значення в операнді **n** має бути між 2–256. Якщо ні, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
4. Ви можете використовувати інструкцію SFWR з інструкцією SFRD (API 1105), щоб записувати та читати дані методом FIFO (першим прийшов, першим вийшов).

API	Код інструкції			Операнд							функція						
1105	D	SFRD	P	S, D, n							Зсув даних і зчитування їх із пристрою Word						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S		●			●	●		●			O						
D		●			●	●		●									
n	●	●			●	●		●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

SFRD En S n	SFRDP En S D n	S : Перший пристрій
DSFRD En S n	DSFRDP En S D n	D : Пристрій, на який передаються дані
		n : Довжина даних

Пояснення

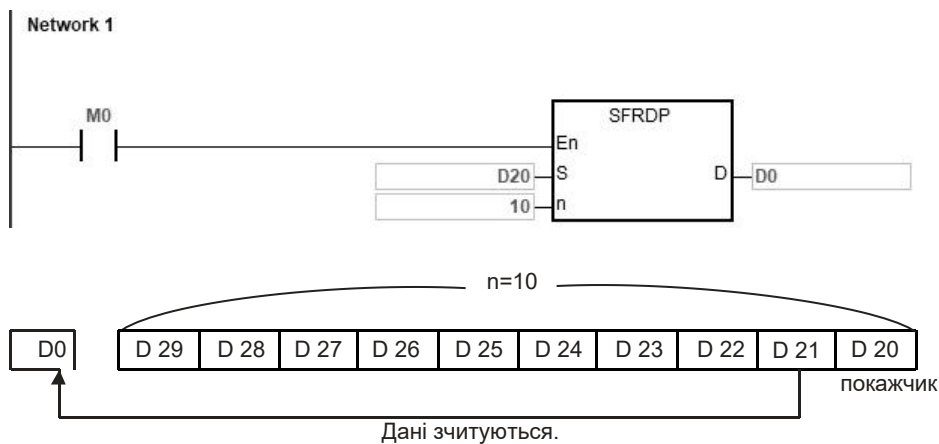
- 6
- Ця інструкція визначає дані в **n**-слові пристроїв, починаючи з пристрою, визначеного **S** як тип списку першим увійшов, першим вийшов, і бере пристрій, вказаний **S**, як покажчик. Ця інструкція зменшує значення в пристрої, визначеному **S**, на одиницю, записує дані в пристрої, визначеному **S + 1**, у пристрій, визначеному **D**, зсуває дані в пристроях, визначених **S + n - 1 – S + 2**, на праворуч і залишає дані в пристрої, визначеному **S + n - 1**, без змін. Коли значення в пристрої, визначеному **S**, дорівнює 0, інструкція припиняє читання даних і встановлює нульовий прапор SM600 у положення ON.
 - Загалом використовується імпульсна інструкція SFRDP.
 - Для 16-бітної інструкції значення в операнді **n** має бути між 2–512. Для 32-розрядної інструкції значення в операнді **n** має бути між 2–256. Для застосування 32-розрядних інструкцій вам потрібно використовувати ISPSOft V3.09.07 або пізнішої версії та AS PLC CPU FW V1.08 або пізнішої версії.

Приклад 1

1. Імовірно, значення в D20 дорівнює 10, коли M0 перемикається з ВІМК. на УВІМК., інструкція записує дані в D21 в D0, зміщує дані в D29–D22 на одне місце вправо, залишає дані в D29 незмінними, і зменшує значення значення в D20 на одиницю.

2. Дані в слові пристрій зсуваються та зчитуються, як показано нижче.

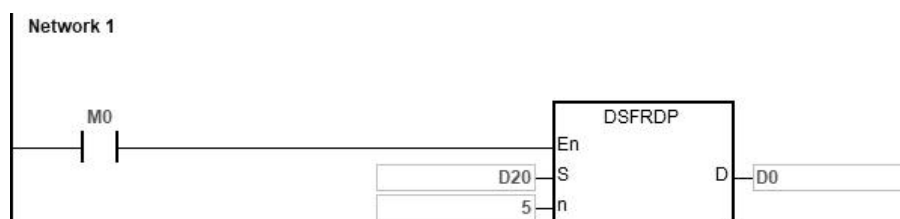
- Дані в D21 зчитуються і зміщуються в D0.
- Дані в D29-D22 зсунуті на одне місце вправо.
- Значення в D20 зменшується на одиницю.

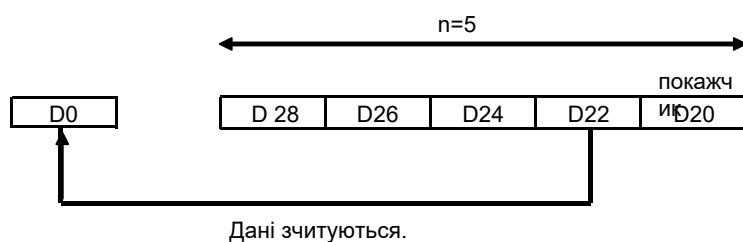
**Приклад 2**

1. Імовірно, значення в D20/D21 дорівнює 5, коли M0 перемикається з ВІМК. на УВІМК., інструкція записує дані з D22/D23 у D0/D1, зміщує 32-розрядні дані в D29–D22 на одне місце вправо, залишає дані в D28/D29 не змінюються, а значення в D20/D21 зменшується на одиницю.

2. Дані в слові пристрій зсуваються та зчитуються, як показано нижче.

- Дані в D22/D23 зчитуються та зміщуються до D0/D1.
- 32-розрядні дані в D29-D22 зсунуті на одне місце вправо.
- Значення в D20 зменшується на одиницю.





Додаткові зауваження

1. Якщо значення в **S** менше 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо **S + n - 1** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Для 16-бітної інструкції значення в операнді **n** має бути між 2–512. Для 32-розрядної інструкції значення в операнді **n** має бути між 2–256. Якщо ні, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
4. Ви можете використовувати цю інструкцію з інструкцією SFWR (API 1104) для запису та читання даних методом FIFO (першим прийшов, першим вийшов).

API		Код інструкції			Операнд						функція						
1106		D	SFPO	P	S, D						Читання останніх даних зі списку даних						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S		●			●	●		●									
D		●			●	●		●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція		
AS			AS			AS		

СИМВОЛ

En	SFPO	En	SFPOP
S	D	S	D

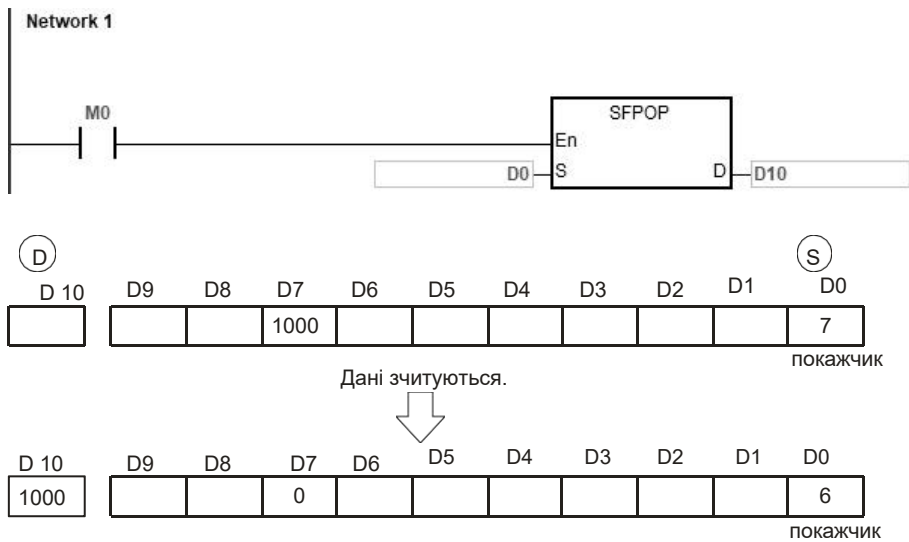
En	DSFPO	En	DSFPOP
S	D	S	D

S : Перший пристрій**D** : Пристрій, де зберігаються дані**Пояснення**

- Ця інструкція бере пристрій, вказаний **S**, як показчик. Ця інструкція записує дані у вказаний пристрій за значенням показчика в пристрій, визначений **D**, очищає його до 0 і зменшує значення в пристрої, визначеному **S**, на одиницю. Коли значення в пристрої, визначеному **S**, дорівнює 0, інструкція припиняє читання даних і встановлює нульовий прапор SM600 у положення ON.
- Загалом використовується імпульсна інструкція SFPOP.
- Ця інструкція широко використовується як функція PUSH. Якщо вам потрібна функція POP, зверніться до інструкцій SFRD і SFPO для отримання додаткової інформації.
- Для застосування 32-розрядних інструкцій вам потрібно використовувати ISPSOft V3.09.07 або пізнішої версії та AS PLC CPU FW V1.08 або пізнішої версії.

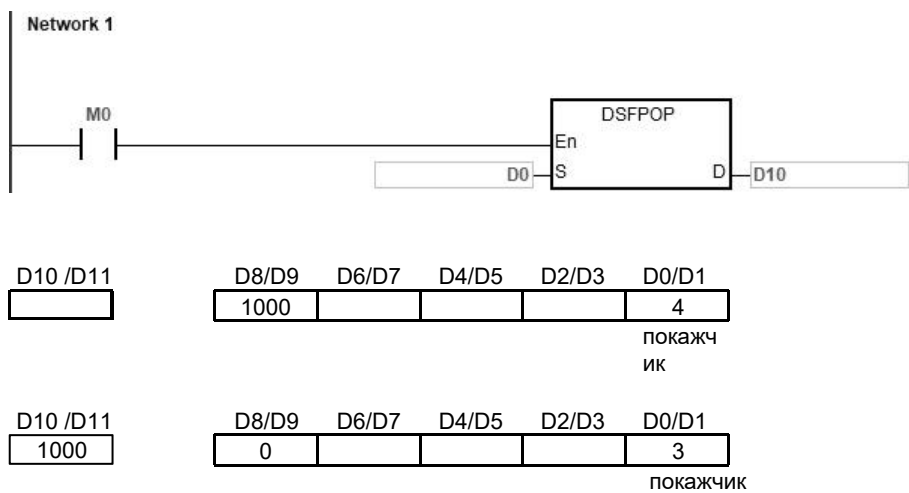
Приклад 1

Імовірно, значення в D0 дорівнює 7, коли M0 увімкнено, інструкція записує дані в пристрій, указаний D0, у D10. Після того, як інструкція зсуває дані, інструкція очищає дані в пристрої, визначеному D0, до 0 і зменшує значення в D0 на 1.



Приклад 2

Імовірно, значення в D0/D1 дорівнює 4, коли M0 увімкнено, інструкція записує дані в пристрій, визначений D0/D1, у D10/D11. Після того, як інструкція зсуває дані, інструкція очищає дані в пристрої, визначеному D0/D1, на 0, і зменшує значення в D0/D1 на 1.



Додаткові зауваження

1. Якщо значення в S менше 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо S + (значення в S) перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції			Операнд							функція					
1107	D	SFDEL	P	S, D, n							Видалення даних зі списку даних					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S		●			●	●		●								
D		●			●	●		●			O	O				
n	●	●			●	●		●	●		O	O	O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція		
AS			AS			AS		

СИМВОЛ

SFDEL		SFDELP	
En		En	
S	D	S	D
n		n	

DSFDEL		DSFDELP	
En		En	
S	D	S	D
n		n	

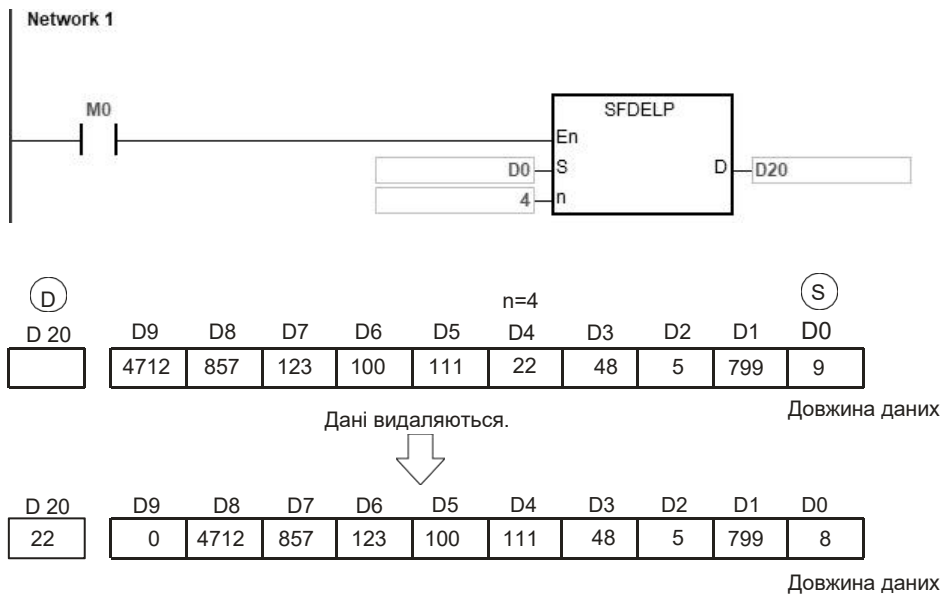
- S** : Перший пристрій
- D** : Пристрій, де зберігаються дані
- n** : Пристрій, на якому видаляються дані

Пояснення

- Для цієї інструкції довжиною даних є значення в пристрої, визначеному **S** , а самі дані знаходяться в пристроях, визначених **S + 1 – S + (значення в S)** . Ця інструкція зберігає дані в пристрої, визначеному **S + n** у **D** , і видаляє їх, зміщує дані в пристроях, указаних **S + n + 1 – S +(значення в S)** праворуч, очищає дані в пристрій, визначений **S +(значення в S)** до 0, і зменшує значення в пристрої, визначеному **S** , на одиницю. Коли значення в пристрої, визначеному **S** , дорівнює 0, інструкція припиняє видалення даних і встановлює нульовий прапор SM600 у положення ON.
- Загалом використовується імпульсна інструкція SFDELP.

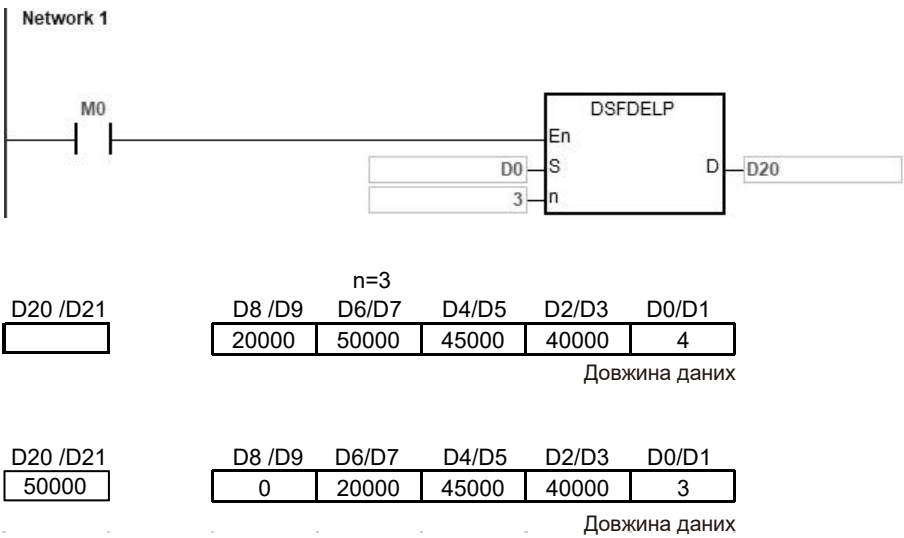
Приклад 1

Імовірно, значення в D0 дорівнює 9, а **n** дорівнює 4. Коли M0 увімкнено, інструкція зберігає дані в D4 до D20. Потім інструкція видаляє дані в D4. Після цього він зсуває дані в D5–D9 праворуч і зменшує значення в D0 на одиницю.



Приклад 2

Імовірно, значення в D0/D1 дорівнює 4, а **n** дорівнює 3. Коли M0 увімкнено, інструкція зберігає дані в D6/D7 до D20. І тоді інструкція видаляє дані в D6/D7. Після цього він зміщує дані в D8/D9 праворуч і зменшує значення в D0/D1 на одиницю.



Додаткові зауваження

1. Якщо значення в **S** менше 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо **S + n** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Якщо **S + (значення в S)** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
4. Якщо **n** не між 0– **S** , інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд							функція					
1108	D	SFINS	P	S, D, n							Вставлення даних у список даних					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S		●			●	●		●								
D	●	●			●	●		●	●		○	○	○	○		
n	●	●			●	●		●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

SFINS	SFINS P
En	En
S	S
n	n

DSFINS	DSFINS P
En	En
S	S
n	n

S : Перший пристрій

D : Дані для вставки

n : Пристрій, куди вставляються дані

Пояснення

1. Для цієї інструкції довжиною даних є значення в пристрої, визначеному **S**, а самі дані знаходяться в пристроях, визначених **S + 1 – S + (значення в S)**. Ця інструкція вставляє дані з **D** у **S + n**, зміщує вихідні дані в пристроях, визначених **S + n – S + (значення в S)**, вліво та збільшує значення в пристрої, визначеному **S**, на одиницю. Коли значення в пристрої, визначеному **S**, дорівнює 32767, інструкція припиняє запис даних, припиняє збільшення значення в пристрої, визначеному **S**, і встановлює прапор переносу SM602 у положення ON.

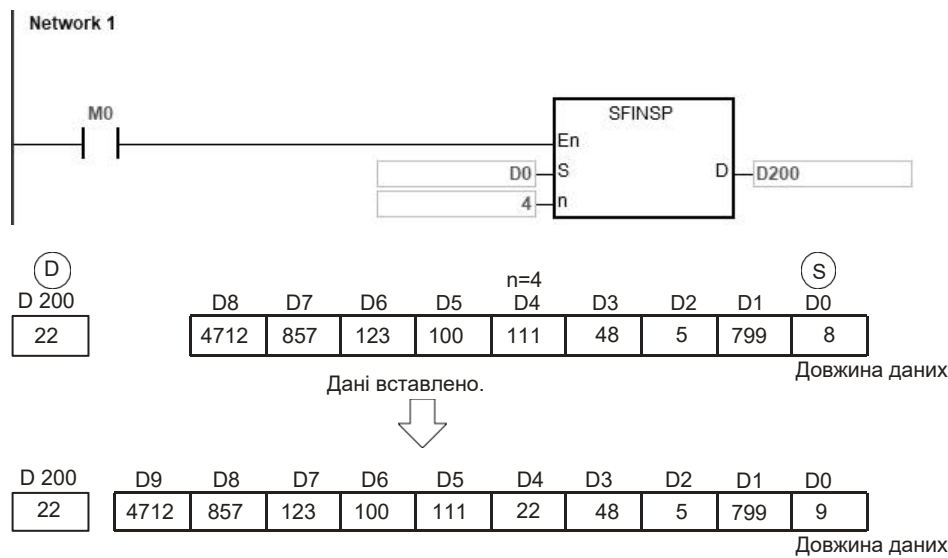
2. Загалом використовується імпульсна інструкція SFINS P.

3. Для 16-бітної інструкції значення в операнді **n** має бути в діапазоні від 1 до 32767. Для 32-бітної інструкції значення в операнді **n** має бути в діапазоні від 1 до 16383.

4. Для застосування 32-розрядних інструкцій вам потрібно використовувати ISPSOft V3.09.07 або пізнішої версії та AS PLC CPU FW V1.08 або пізнішої версії.

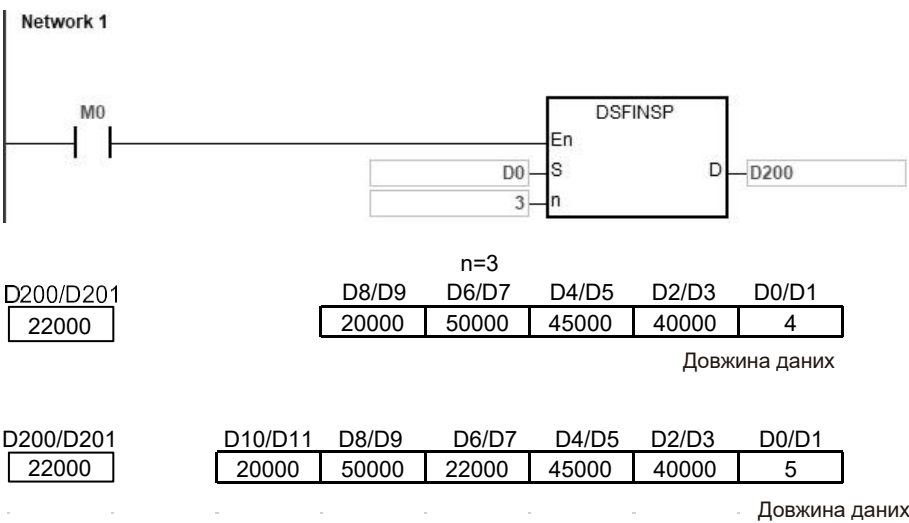
Приклад 1

Імовірно, значення в D0 дорівнює 8, а n дорівнює 4. Коли M0 увімкнено, інструкція вставляє дані з D200 у D4, зміщує вихідні дані з D4 – D8 на D5 – D9 та збільшує значення в D0 на одиницю.



Приклад 2

Імовірно, значення в D0/D1 дорівнює 4, а n дорівнює 3. Коли M0 увімкнено, інструкція вставляє дані з D200 у D6/D7, зміщує вихідні дані з D6 – D9 на D8 – D11 і збільшує значення в D0/D1 по одному.



Додаткові зауваження

1. Якщо значення в S менше 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо S + n перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003
3. Якщо S +(значення в S)+1 перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
4. Якщо n не між 0 – S , інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

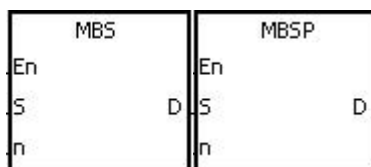
API	Код інструкції			Операнд						функція						
1109		MBS	P	S, D, n						Зміщення бітів матриці						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●		O		O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



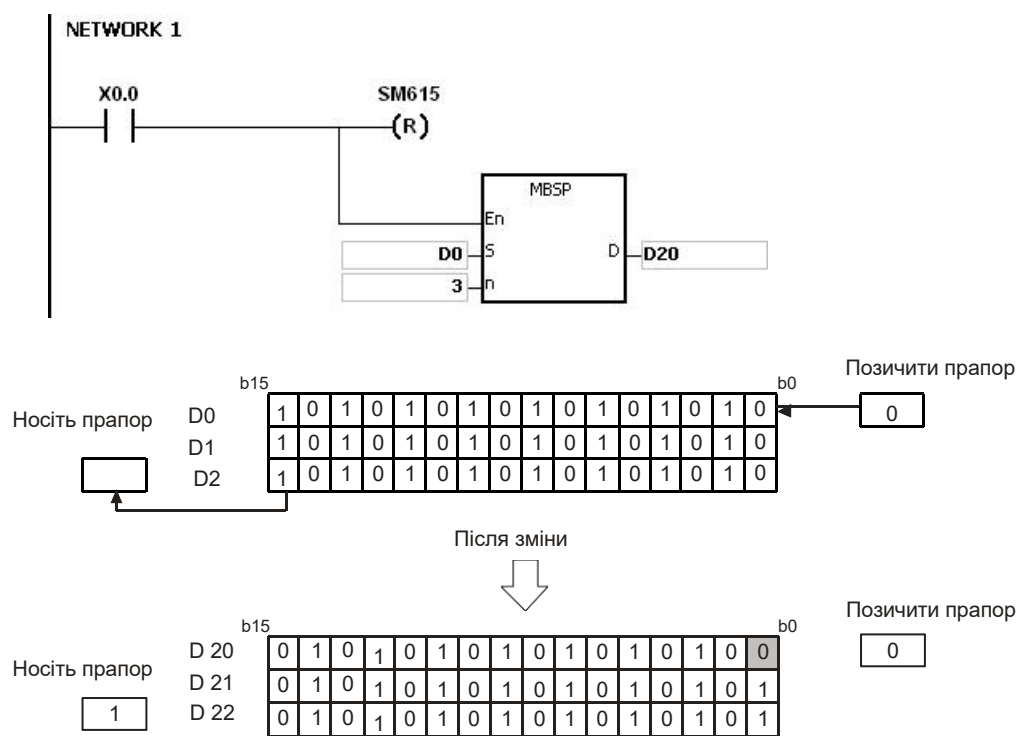
- S** : Джерело матриці
- D** : Результат операції
- n** : Довжина масиву

Пояснення

- Ця інструкція зсуває значення **n** рядків бітів у **S** вправо або вліво. Коли SM616 вимкнено, інструкція зсуває значення бітів вліво. Коли SM616 увімкнено, інструкція зсуває значення бітів праворуч. Інструкція заповнює вакансію (b0 при зсуві вліво та b16n-1 при зсуві вправо), що виникає в результаті зсуву станом прапора запозичення SM615. Інструкція передає значення біта, зсунутого останнім (від зсуву до лівий - b16n-1 і від зсуву вправо - b0) до прапора переносу SM614, і зберігає результат операції в **D**.
- Операнд **n** має бути між 1–256.
- Загалом використовується імпульсна інструкція MBSP.

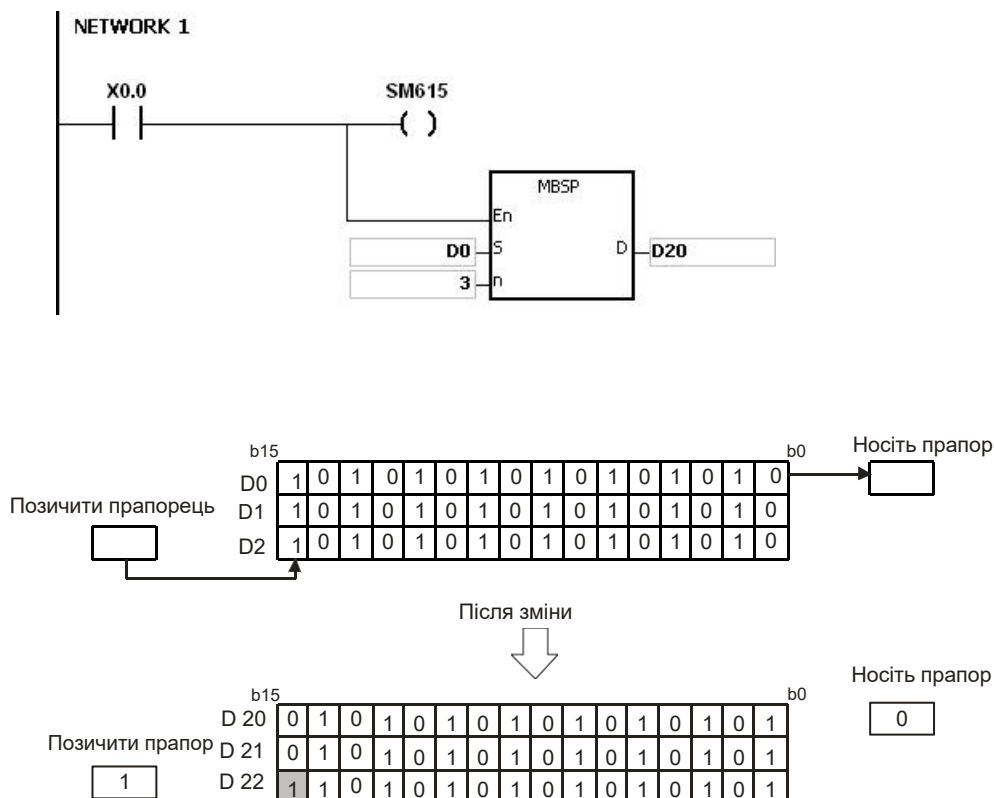
Приклад 1

Коли X0.0 увімкнено, SM616 вимкнено. Інструкція зсуває значення бітів вліво. Припустимо, SM615 вимкнено. Після того, як інструкція зсуває значення бітів у 16-розрядних регістрах D0–D2 вліво, вона зберігає результат операції в 16-розрядних регістрах D20–D22, а SM614 увімкнено.



Приклад 2

Коли X0.0 увімкнено, SM616 увімкнено. Інструкція зсуває значення бітів вправо. Припустимо, SM615 увімкнено. Після того, як інструкція зсуває значення бітів у 16-розрядних регістрах D0–D2 вправо, вона зберігає результат операції в 16-розрядних регістрах D20–D22, а SM614 вимкнено.



Додаткові зауваження

1. Якщо $S + n - 1$ або $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

2. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

3. Інструкційні прапорці

SM614: Прапор переносу для повороту/зсуву/виведення матриці.

SM615: Позичити позначку для зсуву/виведення матриці.

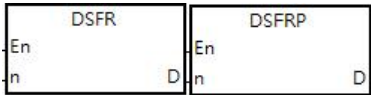
SM616: Позначка напрямку обертання/зсуву матриці.

API	Код інструкції			Операнд				функція								
1110	D	SFR	P	D, n				Зсув значень розрядів у регістрах на n розрядів вправо								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
D		●			●	●	●	●			O	O				
n	●	●			●	●		●	●		O	O	O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



D : Пристрій для зміни
n : Кількість біт

Пояснення

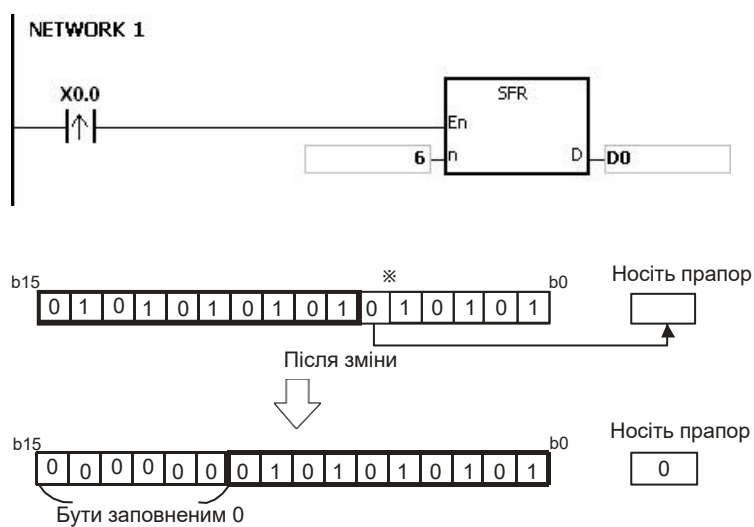
- Ця інструкція зсуває значення бітів у **D** на **n** бітів праворуч. 16-розрядна інструкція заповнює вакансії ($b_{15} - b_{15-n+1}$), що виникають в результаті зсуву, на 0, а 32-бітна інструкція заповнює вакансії ($b_{31} - b_{31-n+1}$), що виникають в результаті зсуву на 0. Значення b_{n-1} передається на SM602.
- Операнд **n** має бути між 1–16 для 16-бітних інструкцій, а операнд **n** має бути між 1–32 для 32-бітних інструкцій інструкції.
- Загалом, імпульсна інструкція SFRP/DSFRP використовується частіше.
- Тільки інструкція DSFR/DSFRP (32-розрядні інструкції) може використовувати 32-розрядний лічильник (високий лічильник), але не пристрій E.

приклад

Коли X0.0 увімкнено, інструкція зсуває значення b0–b15 у D0 на 6 біт вправо та передає значення b5 до SM602. Інструкція очищає значення b10–b15 до нуля після зсуву.

Зрушення значень бітів вправо під час сканування показано нижче.

- ❶ b5-b0 → Переноситься (Значення b5 передається на SM602.)
- ❷ b15-b6 → b9-b0
- ❸ 0 → b15-b10



Додаткові зауваження

Якщо n не знаходиться між 0–16 (для 16-бітних інструкцій) або між 0–32 (для 32-бітних інструкцій), інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд				функція								
1111	D	SFL	P	D, n				Зсув значень розрядів у регістрах на n розрядів вліво								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
D		●			●	●	●	●			O	O				
n	●	●			●	●		●	●		O	O	O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



- D : Пристрій для зміни
n : Кількість біт

Пояснення

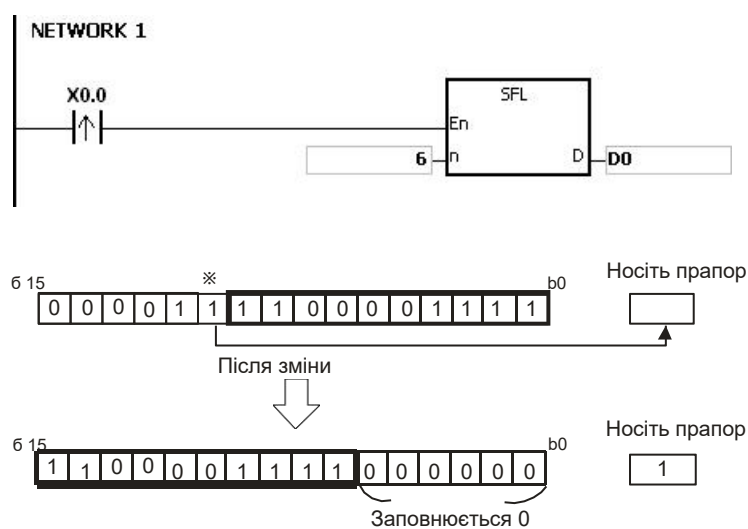
- Ця інструкція зсуває значення бітів у D на n бітів вліво. Інструкції заповнюють вакансії ($b_0 - b_{n-1}$), що виникають в результаті зсуву на 0, і передають значення b_{16-n} (для 16-бітної інструкції) або b_{32-n} (для 32-бітної інструкції) до SM602 . .
- Операнд n має бути між 1–16 для 16-бітних інструкцій, а операнд n має бути між 1–32 для 32-бітних інструкцій інструкції.
- Загалом, імпульсна інструкція SFRP/DSFRP використовується частіше.
- Тільки інструкція DSFR/DSFRP (32-розрядні інструкції) може використовувати 32-розрядний лічильник (високий лічильник), але не пристрій E.

приклад

Коли X0.0 увімкнено, інструкція зсуває значення b0 – b15 у D0 на 6 біт вправо та передає значення b10 до SM602. Інструкція заповнює значення b0 – b5 нулями після зсуву.

Зрушення значень бітів вліво під час сканування показано нижче.

- ❶ b15-b10 → Переноситься (Значення b10 передається на SM602.
- ❷ b9-b0 → b15-b6
- ❸ 0 → b5-b0



Додаткові зауваження

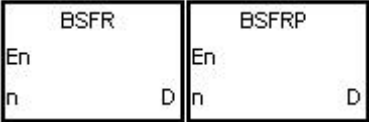
Якщо **n** не знаходиться між 0–16 (для 16-бітних інструкцій) або між 0–32 (для 32-бітних інструкцій), інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд						функція							
1112		BSFR	P	D, n						Зрушення станів n -розрядних пристроїв на один біт вправо							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●	●	●				●									
n	●	●			●	●		●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



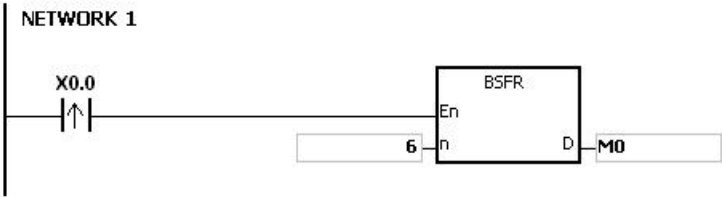
D : Перший апарат для зміни
n : Довжина даних

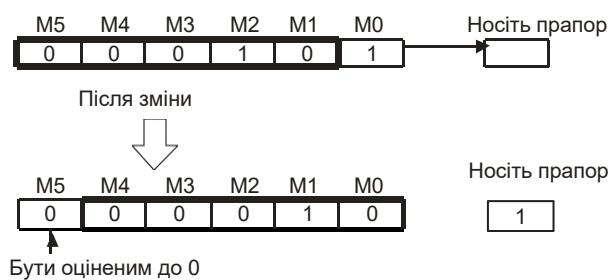
Пояснення

- Ця інструкція зсуває стани n -розрядних пристроїв, починаючи з D , на один біт вправо. Інструкція очищає стан D + n -1 до 0 і передає стан D до прапора перенесення SM602.
- Загалом використовується імпульсна інструкція BSFRP.
- Операнд n має бути між 1–1024.

приклад

Коли X0.0 увімкнено, інструкція зсуває стани M0–M5 на один біт вправо, очищає стан M5 до нуля та передає стан M0 на прапор переносу SM602.





Додаткові зауваження

1. Якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n не між 1–1024, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд				функція									
1113		BSFL	P	D, n				Зрушення станів n -розрядних пристроїв на один біт вліво									
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●	●	●				●									
n	●	●			●	●		●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D	●												
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



D : Перший апарат для зміни

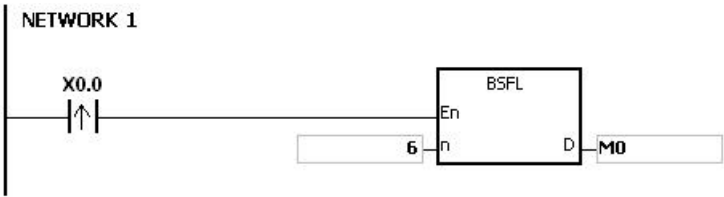
n : Довжина даних

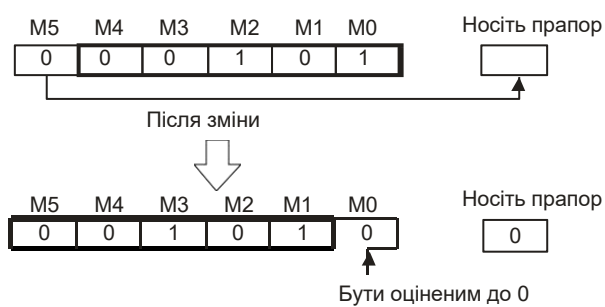
Пояснення

- 1. Ця інструкція зсуває стани n -розрядних пристроїв, починаючи з D , на один біт вліво. Інструкція очищає стан D до 0 і передає стан D + n -1 на прапор переносу SM602.
- 2. Загалом використовується імпульсна інструкція BSFLP.
- 3. Операнд n має бути між 1–1024.

приклад

Коли X0.0 увімкнено, інструкція зсуває стани M0–M5 на один біт вліво, очищає стан M0 до 0 і передає стан M5 на прапор переносу SM602.





Додаткові зауваження

1. Якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n не між 1–1024, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд							функція						
1114	D	NSFR	P	D, n							Зсув n регістрів вправо						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●		●									
n	●	●			●	●		●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



D : Перший апарат для зміни

n : Довжина даних

Пояснення

- Ця інструкція зсуває дані в n регістрах, починаючи з D , праворуч і очищає дані в D + n -1 до 0.
- Загалом використовується імпульсна інструкція NSFRP.
- Для 16-бітної інструкції значення в операнді n має бути між 1–512. Для 32-розрядної інструкції значення в операнд n має бути між 1–256. Для застосування 32-розрядних інструкцій вам потрібно використовувати ISPSOft V3.09.07 або пізнішої версії та AS PLC CPU FW V1.08 або пізнішої версії.

Приклад 1

Коли M0 перемикається з OFF на ON, інструкція зміщує дані в D1–D6 праворуч і очищає дані в D6 до 0.



D6	D5	D4	D3	D2	D1	D0
30	2235	9578	754	28	423	11

Після зміни



D6	D5	D4	D3	D2	D1	D0
0	30	2235	9578	754	28	423

Очищається до 0

Приклад 2

Коли M0 увімкнено, коли M0 перемикається з ВИМК. на УВІМК., інструкція зміщує 7 фрагментів 32-бітних даних у D0–D13 на одне місце вправо та очищає дані в D12 до 0.



D12	D10	D 8	D6	D4	D2	D0
90000	80000	70000	60000	50000	30000	40000



Після зміни

D12	D10	D 8	D6	D4	D2	D0
0	90000	80000	70000	60000	50000	30000

Додаткові зауваження

1. Якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Для 16-бітної інструкції значення в операнді n має бути між 1–512. Для 32-розрядної інструкції значення в операнді n має бути між 1–256. Якщо ні, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд							функція						
1115	D	NSFL	P	D, n							Зсув n регістрів вліво						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●		●									
n	●	●			●	●		●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

NSFL

En

n

D

NSFLP

En

n

D

DNSFL

En

n

D

DNSFLP

En

n

D

D : Перший апарат для зміни

n : Довжина даних

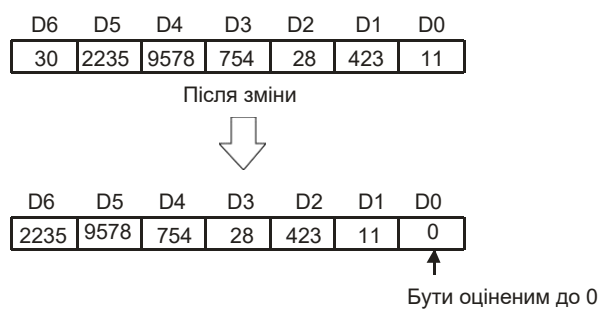
Пояснення

1. Ця інструкція зсуває дані в n регістрах, починаючи з D , вліво, і очищає дані в D до 0.
2. Загалом використовується імпульсна інструкція NSFLP.
3. Для 16-бітної інструкції значення в операнді n має бути між 1–512. Для 32-розрядної інструкції значення в операнд n має бути між 1–256. Для застосування 32-розрядних інструкцій вам потрібно використовувати ISPSOft V3.09.07 або пізнішої версії та AS PLC CPU FW V1.08 або пізнішої версії.

приклад

Коли M0 перемикається з OFF на ON, інструкція зміщує дані в D0–D5 вліво та очищає дані в D0 до 0.





Додаткові зауваження

1. Якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Для 16-бітної інструкції значення в операнді n має бути між 1–512. Для 32-розрядної інструкції значення в операнді n має бути між 1–256. Якщо ні, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

6.13 Інструкція з обробки даних

6.13.1 Перелік інструкцій з обробки даних

У наведеній нижче таблиці наведено інструкції з обробки даних, описані в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1200	SER	DSER	✓	Пошук даних
1201	SUM	DSUM	✓	Знаходження кількості бітів, стан яких ON
1202	DECO	–	✓	Декодування бітів
1203	ENCO	–	✓	Кодування бітів
1204	SEGD	–	✓	Семисегментное декодування
1205	SORT	DSORT	✓	Сортування даних
1206	ZRST	–	✓	Скидання зони
1207	BON	DBON	✓	Перевірка стану біта
1208	MEAN	DSEAN	✓	Знаходження середнього
1209	CCD	–	✓	Знаходження суми чека
1210	ABS	DABS	✓	Знаходження абсолютної величини
1211	MINV	–	✓	Інвертування бітів матриці
1212	MBRD	–	✓	Читання біта матриці
1213	MBWR	–	✓	Запис матричного біта
1214	MBC	–	✓	Підрахунок бітів зі значенням нуль або одиниця
1215	DIS	–	✓	Роз'єднання 16-бітних даних
1216	UNI	–	✓	Об'єднання 16-бітних даних
1217	WSUM	DWSUM	✓	Знаходження суми
1221	LIMIT	DLIMIT	✓	Обмеження значення в межах
1222	BAND	DBAND	✓	Контроль зони нечутливості
1223	ZONE	DZONE	✓	Контроль зони
1224	–	FMEAN	✓	Знаходження середнього значення чисел з плаваючою комою
1225	–	FSUM	✓	Знаходження суми чисел з плаваючою комою
1226	DTM	–	✓	Перетворення та переміщення даних
1228	ZSET	–	✓	Налаштування зони

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1229	–	FSORT	✓	Сортування даних у форматі з плаваючою комою

6.13.2 Пояснення інструкцій з обробки даних

API	Код інструкції				Операнд								функція				
1200	D	SER		P	S ₁ , S ₂ , D, n								Пошук даних				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁	●	●			●	●	●	●	●								
S ₂	●	●			●	●	●	●	●		O	O	O	O			
D		●			●	●	●	●									
n	●	●			●	●	●	●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●						
S ₂		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

SER		DSER	
En		En	
S1	D	S1	D
S2		S2	
N		N	
SERP		DSERP	
En		En	
S1	D	S1	D
S2		S2	
N		N	

- S₁ :Перший прилад для порівняння
- S₂ :Порівняні дані
- D :Перший пристрій, де зберігається результат порівняння
- n :Довжина даних

Пояснення

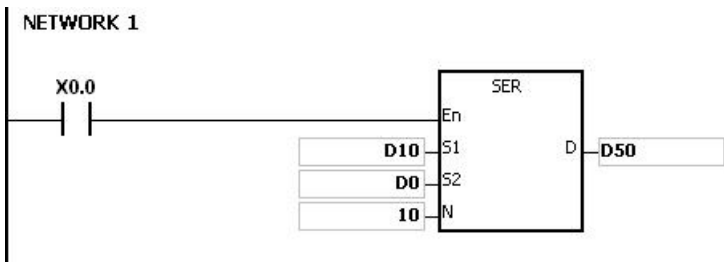
1. Ця інструкція порівнює n десяткових значень зі знаком у регістрах, починаючи з регістру, визначеного S₁, із десятковим значенням зі знаком у регістрі, визначеному S₂, і зберігає результати порівняння в регістрах D – D +4.

пристрій	опис
D	Кількість рівних значень
D +1	Номер даних першого рівного значення
D +2	Номер даних останнього дорівнює значення
D +3	Дані номер мінімуму значення
D +4	Кількість даних максимуму значення

- 2. Операнд **n** має бути в діапазоні від 1 до 256.
- 3. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

приклад

- 1. Коли X0.0 увімкнено, інструкція порівнює значення в D10 – D19 зі значенням у D0 та зберігає результати порівняння в D50 – D54. Якщо рівне значення не існує в D10 – D19, значення в D50 – D52 дорівнюють 0.
- 2. Інструкція зберігає номер даних мінімального значення в D53 і зберігає номер даних максимального значення в D54. Якщо існує більше одного мінімального або максимального значення, інструкція зберігає більше число даних.



n	S₁	Значення	Порівняно даних	Дані номер	Результат	D	Значення	опис
	D10	88	S₂ D0=100	0		D50	4	Кількість рівних значень
	D11	100		1	Рівні	D51	1	Номер даних першого рівне значення
	D12	110		2		D52	8	Номер даних остан рівне значення
	D13	150		3		D53	7	Номер даних мінімальне значення
	D14	100		4	Рівні	D54	9	Номер даних максимальне значення
	D15	300		5				
	D16	100		6	Рівні			
	D17	5		7	мінімум			
	D18	100		8	Рівні			
	D19	500		9	Максимум			

Додаткові зауваження

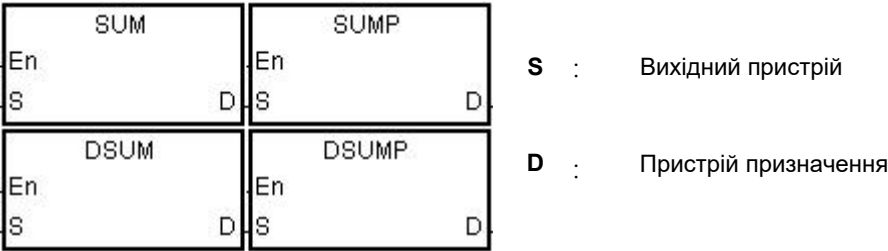
1. Якщо **S₁ + n - 1** або **D + 4** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 є 16#2003.
2. Для 16-розрядних інструкцій, якщо значення в **n** не знаходиться в межах від 1 до 256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Для 32-розрядних інструкцій, якщо значення в **n** не знаходиться в межах від 1 до 128, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
4. Для 16-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSOft, тип даних буде ARRAY [5] з WORD/INT.
5. Для 32-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSOft, тип даних буде ARRAY [5] з DWORD/DINT.

API	Код інструкції			Операнд							функція						
1201	D	SUM	P	S, D							Знаходження кількості бітів, стан яких ON						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O	O	O	O			
D		●			●	●	●	●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

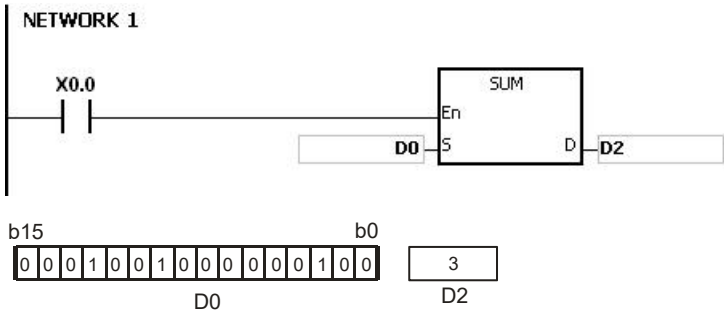


Пояснення

- 6
- Ця інструкція знаходить кількість бітів у **S** , значення яких є **ON** , і зберігає кількість бітів ON у **D**.
 - Коли значення всіх бітів у вихідному пристрої, визначеному **S** , дорівнюють 0, нульовий прапор SM600 увімкнено.
 - Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

приклад

Коли X0.0 увімкнено, інструкція зберігає кількість бітів, значення яких дорівнюють одиниці в D0 у D2.



Додаткові зауваження

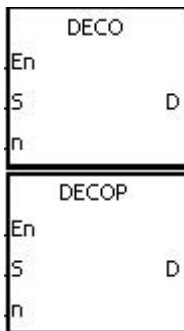
Якщо пристрій виходить за межі діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції			Операнд							функція						
1202		DECO	P	S, D, n							Декодування бітів						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●	●	●	●	●		●	●	○	○	○					
D		●	●	●	●	●		●				○					
n	●	●			●	●		●	●		○	○	○	○			

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●	●				●							
D	●	●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



S : Вихідний пристрій

D : Пристрій, де зберігаються розшифровані значення

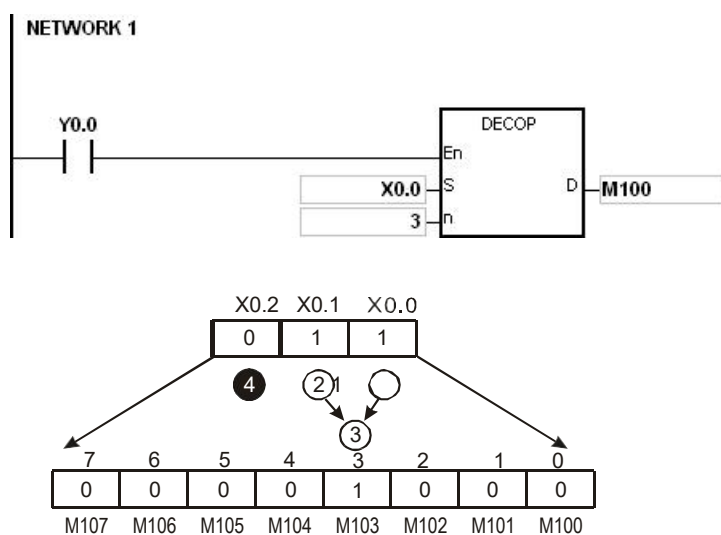
n : Кількість бітів, значення яких декодуються

Пояснення

1. Ця інструкція декодує значення молодших **n** бітів у вихідному пристрої, визначеному **S**, як значення молодших 2^n бітів у **D**.
2. Інструкція декодує значення послідовних **n** бітів у вихідному пристрої, визначеному **S**, як значення молодших 2^n бітів у **D**.
3. Коли вихідним пристроєм, визначеним **S**, є таймер або лічильник, інструкція розглядає пристрій як пристрій слова.
4. Коли **D** є бітовим пристроєм, **n** між 1 і 8. Коли **n** дорівнює 8, інструкція декодує значення восьми бітів як значення 256 бітів. Зверніть увагу, що пристрої, в яких зберігаються розшифровані значення, не можна використовувати повторно.
5. Коли **D** є словом, **n** становить від 1 до 4. Коли **n** дорівнює 4, інструкція декодує значення чотирьох бітів як значення 16 бітів.
6. Загалом використовується імпульсна інструкція DECOP.

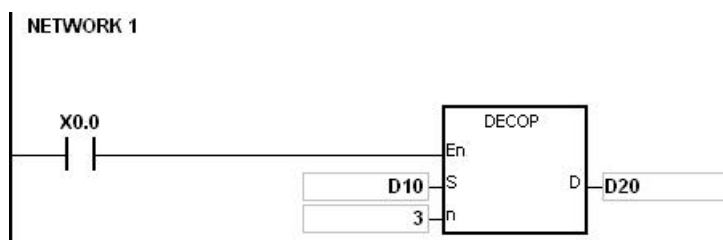
Приклад 1

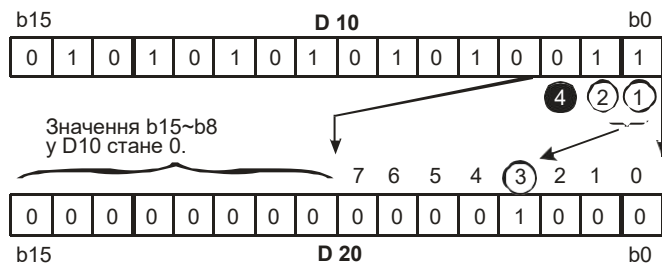
1. Коли Y0.0 перемикається з OFF на ON, інструкція DECO декодує значення 3 бітів у X0.0 – X0.2 як значення 8 бітів у M100 – M107.
2. Інструкція додає значення 3 бітів у X0.0–X0.2, щоб отримати значення 3. Інструкція встановлює третій біт у M10–M1007, тобто біт у M103, на 1.
3. Після виконання інструкції DECO і Y0.0 перемикається в положення OFF, значення восьми бітів у M100 – M107 залишаються незмінними.



Приклад 2

1. Коли X0.0 перемикається з OFF на ON, інструкція DECO декодує значення b2 – b0 у D10 як значення b7 – b0 у D20, і встановлює значення b15 – b8 у D10 на 0.
2. Інструкція декодує значення молодших трьох бітів у D10 як значення молодших восьми бітів у D20. Інструкція встановлює значення старших восьми бітів на 0.
3. Після виконання інструкції DECO і X0.0 переходить у положення ВИМК., дані в D20 не змінюються.





Додаткові зауваження

1. Якщо **D** є розрядним пристроєм і якщо **n** не в межах 1–8, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
2. Якщо **D** є текстовим пристроєм і якщо **n** не знаходиться в межах 1–4, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Якщо **S** є розрядним пристроєм і якщо **S + n - 1** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
4. Якщо **D** є розрядним пристроєм і якщо **D + (2ⁿ - 1)** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

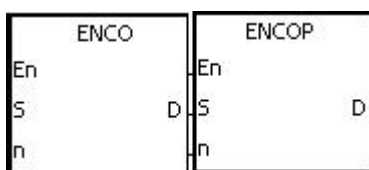
API	Код інструкції			Операнд							функція						
1203		ENCO	P	S, D, n							Кодування бітів						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●	●	●	●	●		●	●	○	○	○				
D		●			●	●		●			○	○				
n	●	●			●	●		●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●	●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



- S** : Вихідний пристрій
- D** : Пристрій, де зберігаються закодовані значення
- n** : Кількість бітів, значення яких кодуються

Пояснення

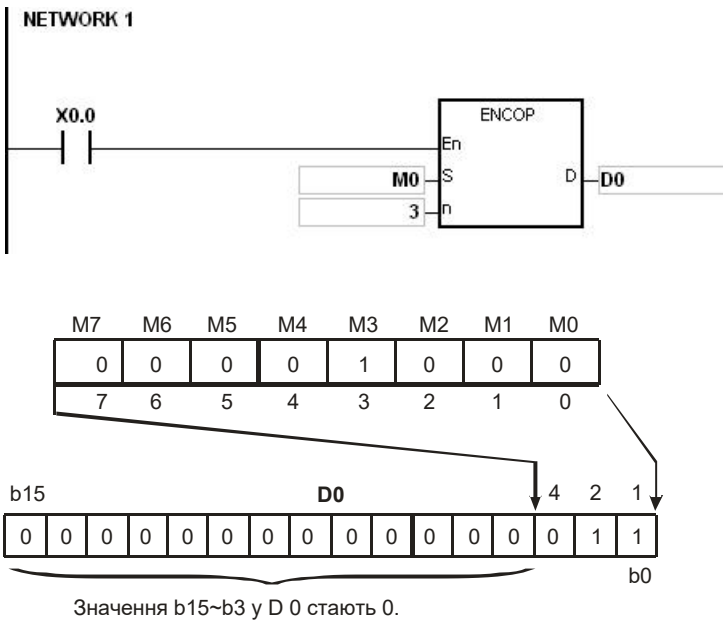
- Коли **S** є текстовим пристроєм, ця інструкція кодує значення молодших 2ⁿ бітів у вихідному пристрої, визначеному **S** як значення молодших **n** бітів у **D**.
- Коли **S** є бітовим пристроєм, інструкція обробляє старший біт зі значенням **S** + (**n** - 1) із молодших 2ⁿ бітів і зберігає результат у **D**.
- Коли вихідним пристроєм, визначеним **S**, є таймер або лічильник, інструкція розглядає пристрій як пристрій слова.
- Коли **S** є бітовим пристроєм, **n** знаходиться в межах від 1 до 8. Коли **n** дорівнює 8, інструкція кодує значення 256 біт як значення восьми біт.
- Коли **S** є текстовим пристроєм, **n** становить від 1 до 4. Коли **n** дорівнює 4, інструкція кодує значення 16 бітів як значення чотирьох бітів.
- Загалом використовується імпульсна інструкція ENCOP.

Приклад 1

- Коли X0.0 перемикається з OFF на ON, інструкція ENCO кодує значення 8 біт у M0 – M7 як

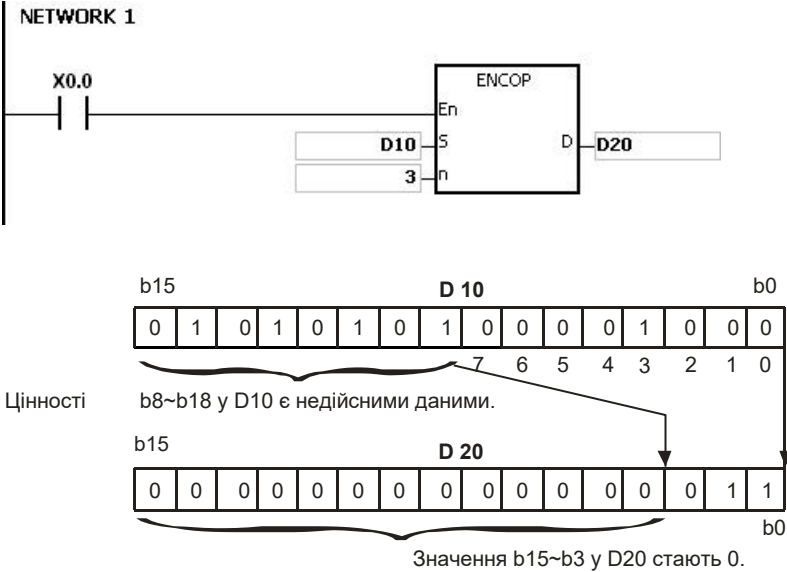
значення 3 молодших бітів у D0 та встановлює значення b15 – b3 у D0 на 0.

2. Після виконання інструкції ENCO і X0.0 перемикається в положення OFF, дані в **D** не змінюються.



Приклад 2

1. Коли X0.0 перемикається з OFF на ON, інструкція ENCO кодує значення b0 – b7 у D10 як значення b2 – b0 у D20 і встановлює значення b15 – b3 у D20 на нуль. Значення b8 – b18 у D10 є недійсними даними.
2. Після виконання інструкції ENCO та вимкнення X0.0 дані в **D** не змінюються.



Додаткові зауваження

1. Якщо в пристрої джерела, визначеному S, немає біта зі значенням одиниці, інструкція не виконується, SM0 увімкнено,

а код помилки в SR0 – 16#2003.

2. Якщо **S** є розрядним пристроєм і якщо **n** не знаходиться в межах від 1 до 8, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

3. Якщо **S** є текстовим пристроєм і якщо **n** не знаходиться в межах 1–4, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

4. Якщо **S** є розрядним пристроєм і якщо $S + (2^n) - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

5. Якщо **D** є розрядним пристроєм і якщо $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції				Операнд							функція					
1204		SEGD	P		S, D							Семисегментное декодування					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●	●		O	O	O	O			
D		●			●	●		●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



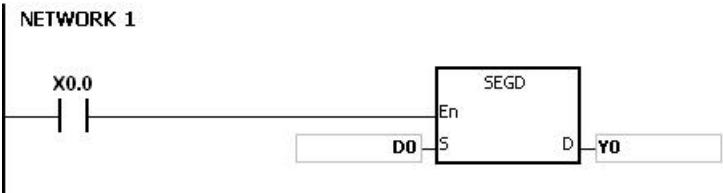
- S : Вихідний пристрій
- D : Пристрій, де зберігаються семисегментні дані

Пояснення

Інструкція декодує значення молодших чотирьох бітів (b0–b3) у вихідному пристрої, визначеному **S** як семи-дані сегмента, що зберігаються в **D** .

приклад

Коли X0.0 увімкнено, інструкція декодує значення b0–b3 у D0 як семисегментні дані, що зберігаються в Y0.0–Y0.15. Якщо дані в пристрої-джерелі перевищують чотири біти, інструкція декодує значення молодших чотирьох бітів.



У наведеній нижче таблиці показано співвідношення між семисегментними даними та бітовим шаблоном вихідних даних.

Hex	Bi t візерунок	Призначення відрізків	Стан сегмента							Відтворення
			B0(a)	B1(b)	B2(c)	B3(d)	B4(e)	B5(f)	B6(g)	
0	0000		ON	ON	ON	ON	ON	ON	OFF	0
1	0001		OFF	ON	ON	OFF	OFF	OFF	OFF	1
2	0010		ON	ON	OFF	ON	ON	OFF	ON	2
3	0011		ON	ON	ON	ON	OFF	OFF	ON	3
4	0100		OFF	ON	ON	OFF	OFF	ON	ON	4
5	0101		ON	OFF	ON	ON	OFF	ON	ON	5
6	0110		ON	OFF	ON	ON	ON	ON	ON	6
7	0111		ON	ON	ON	OFF	OFF	ON	OFF	7
8	1000		ON	ON	ON	ON	ON	ON	ON	8
9	1001		ON	ON	ON	ON	OFF	ON	ON	9
A	1010		ON	ON	ON	OFF	ON	ON	ON	A
B	1011		OFF	OFF	ON	ON	ON	ON	ON	B
C	1100		ON	OFF	OFF	ON	ON	ON	OFF	C
D	1101		OFF	ON	ON	ON	ON	OFF	ON	D
E	1110		ON	OFF	OFF	ON	ON	ON	ON	E
F	1111		ON	OFF	OFF	OFF	ON	ON	ON	F

API	Код інструкції			Операнд							функція						
1205	D	SORT	P	S, m_1, m_2, D, n							Сортування даних						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S								●								
m₁								●					○	○		
m₂								●					○	○		
D								●								
n								●					○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
m₁		●	●			●	●						
m₂		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

SORT		DSORT	
En		En	
S	D	S	D
m1		m1	
m2		m2	
n		n	
SORTP		DSORTP	
En		En	
S	D	S	D
m1		m1	
m2		m2	
n		n	

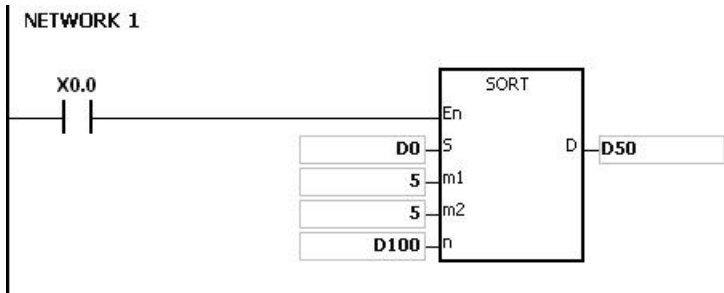
- S** : Перший пристрій, де зберігаються вихідні дані
m₁ : Кількість рядків даних
m₂ : Кількість стовпців даних
D : Перший пристрій, де зберігаються відсортовані дані
n : Довідкове значення для сортування даних

Пояснення

- Ця інструкція зберігає дані для сортування в регістрах **m₁ × m₂**, починаючи з регістра, визначеного **D**. Якщо **S** і **D** вказують той самий регістр, відсортовані дані є такими ж, як вихідні дані в регістрі, заданому **S**.
- Операнд **m₁** має бути від 1 до 32. Операнд **m₂** має бути від 1 до 6. Операнд **n** має бути від 1 до **m₂**.
- Коли SM604 вимкнено, інструкція сортує дані в порядку зростання. Коли SM604 увімкнено, інструкція сортує дані в порядку спадання.
- Рекомендується використовувати інструкцію імпульсного типу SORTP або DSORTP замість повторного сортування.
- Тільки 32-бітна інструкція може використовувати 32-бітний лічильник, але не пристрій E.

приклад

1. Припустимо, що SM604 вимкнено. Коли X0.0 перемикається з OFF на ON, інструкція сортує дані в порядку зростання.



2. Дані, які потрібно відсортувати, наведені в наступній таблиці.

		← m ₂ стовпці даних →				
		Колонка				
	Колонка	1	2	3	4	5
	рядок	Номер учня	китайський	англійська	математика	Фізика
↑ m ₁ рядків даних ↓	1	(D0) 1	(D5) 90	(D10) 75	(D15) 66	(D20) 79
	2	(D1) 2	(D6) 55	(D11) 65	(D16) 54	(D21) 63
	3	(D2) 3	(D7) 80	(D12) 98	(D17) 89	(D22) 90
	4	(D3) 4	(D8) 70	(D13) 60	(D18) 99	(D23) 50
	5	(D4) 5	(D9) 95	(D14) 79	(D19) 75	(D24) 69

3. Якщо значення в D100 дорівнює 3, дані сортуються, як у наведеній нижче таблиці.

		m_2 стовпці даних				
		Колонка				
	Колонка рядок	1	2	3	4	5
		Номер учня	китайський	англійська	математика	Фізика
↑ m_1 рядків даних	1	(D50) 4	(D55) 70	(D60) 60	(D65) 99	(D70) 50
	2	(D51) 2	(D56) 55	(D61) 65	(D66) 54	(D71) 63
	3	(D52) 1	(D57) 90	(D62) 75	(D67) 66	(D72) 79
	4	(D53) 5	(D58) 95	(D63) 79	(D68) 75	(D73) 69
	5	(D54) 3	(D59) 80	(D64) 98	(D69) 89	(D74) 90

4. Якщо значення D100 дорівнює 5, дані виглядають як у наведеній нижче таблиці.

		m_2 стовпці даних				
		Колонка				
	Колонка рядок	1	2	3	4	5
		Номер учня	китайський	англійська	математика	Фізика
↑ m_1 рядків даних	1	(D50) 4	(D55) 70	(D60) 60	(D65) 99	(D70) 50
	2	(D51) 2	(D56) 55	(D61) 65	(D66) 54	(D71) 63
	3	(D52) 5	(D57) 95	(D62) 79	(D67) 75	(D72) 69
	4	(D53) 1	(D58) 90	(D63) 75	(D68) 66	(D73) 79
	5	(D54) 3	(D59) 80	(D64) 98	(D69) 89	(D74) 90

Додаткові зауваження

1. Якщо пристрій виходить за межі діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо m_1 , m_2 або n перевищує діапазон, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.

API	Код інструкції				Операнд							функція					
1206		ZRST	P		D_1, D_2							Скидання зони					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D_1		●	●	●	●	●	●	●		○	○	○					
D_2		●	●	●	●	●	●	●		○	○	○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D_1	●	●				●							
D_2	●	●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

ZRST	ZRSTP
En	En
D1	D1
D2	D2

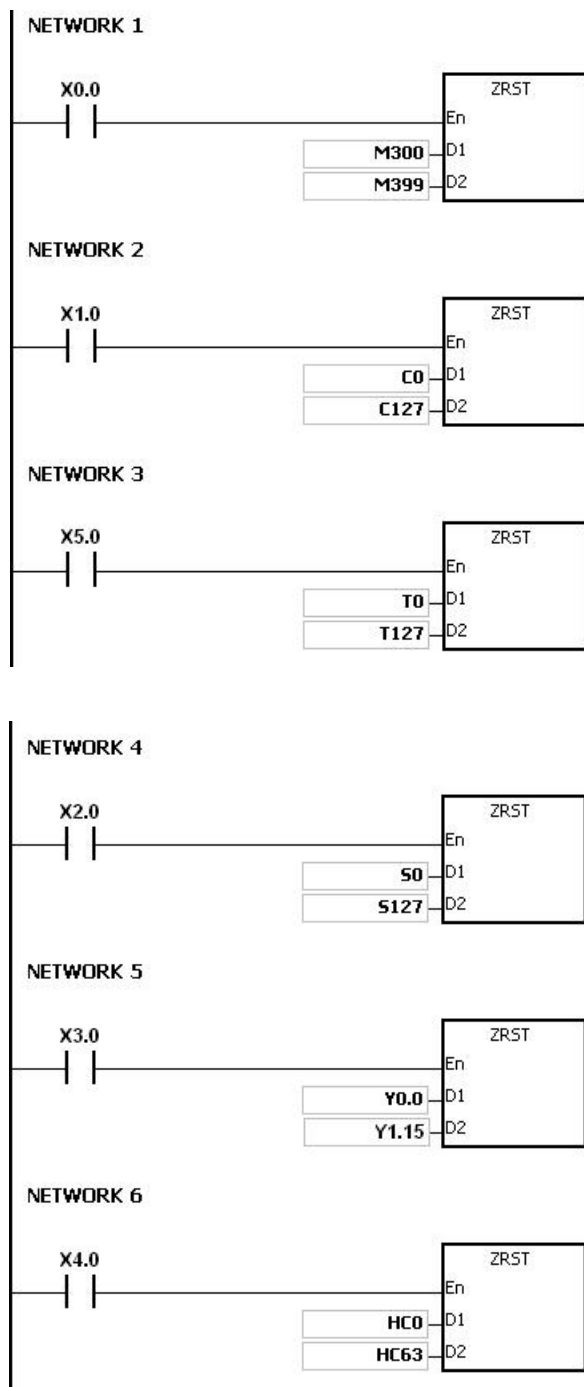
 D_1 : Перший пристрій для скидання D_2 : Останній пристрій для скидання

Пояснення

- Ця інструкція очищає значення в $D_1 - D_2$. Тип пристрою для $D_1 - D_2$ має бути однаковим для цієї інструкції.
- Коли номер пристрою D_1 більший за номер пристрою D_2 , інструкція скидає лише D_2 .
- Інструкція ZRST може використовувати 32-розрядний лічильник.

приклад

- Коли X0.0 увімкнено, інструкція скидає допоміжні реле M300–M399 у положення ВІМК.
- Коли X1.0 увімкнено, інструкція скидає 16-розрядні лічильники C0–C127. Значення C0–C127 скидаються до нуля, і контакт і котушка скидаються до ВІМК.
- Коли X2.0 увімкнено, інструкція скидає крокові реле S0–S127 у положення ВІМК.
- Коли X3.0 увімкнено, інструкція скидає вихідні реле Y0.0–Y1.15 у положення ВІМК.
- Коли X4.0 увімкнено, інструкція скидає 32-розрядні лічильники HC0–HC63. Значення HC0–HC63 скидаються до нуля, а контакт і котушка скидаються до ВІМК.
- Коли X5.0 увімкнено, інструкція скидає таймери T0–T127. Значення T0–T127 скидаються до 0, а контакт і котушка скидаються до ВІМК.



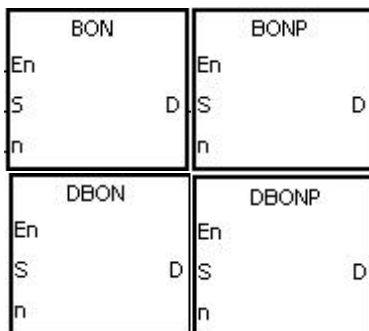
Додаткові зауваження

1. Якщо D_1 і D_2 — різні типи пристроїв, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2007.
2. Якщо D_1 і D_2 містять різні формати даних, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2007.

API	Код інструкції			Операнд							функція					
1207	D	BON	P	S, D, n							Перевірка стану біта					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		○	○	○	○		
D		●	●	●				●		○						
n	●	●			●	●	●	●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D	●												
n		●	●			●	●						

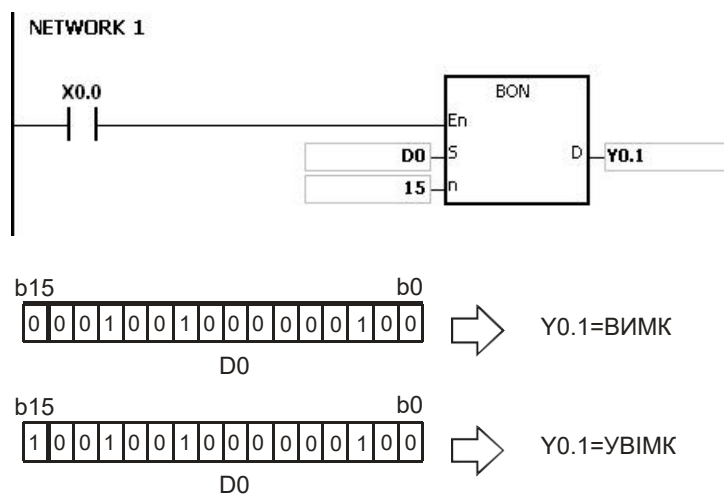
Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ**S** : Вихідний пристрій**D** : Пристрій, на якому зберігається результат перевірки**n** : Біт, стан якого перевіряється**Пояснення**

- Ця інструкція перевіряє стан **n**-го біта в **S** і зберігає результат у **D**.
- Операнд **n**, що використовується в 16-бітній інструкції, має бути між 0–15. Для 32-розрядних інструкцій **n** має бути між 0–31.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій **E**.

приклад

- Коли X0.0 увімкнено, Y0.1 увімкнено, якщо значення 15-го біта в D0 дорівнює одиниці. Коли X0.0 увімкнено, Y0.1 вимкнено, якщо значення 15-го біта в D0 дорівнює 0.
- Коли X0.0 перемикається у положення OFF, стан Y0.1 залишається таким самим, як і до того, як X0.0 перемикається у положення OFF.



Додаткові зауваження

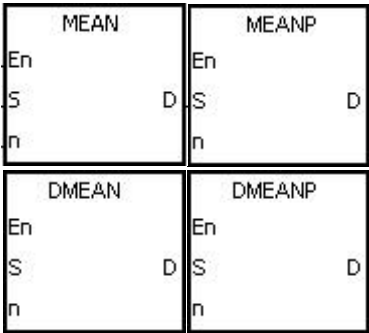
Якщо **n** перевищує діапазон, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд							функція						
1208	D	MEAN	P	S, D, n							Знаходження середнього						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●								
D		●			●	●	●	●			○	○					
n	●	●			●	●	●	●	●		○	○	○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



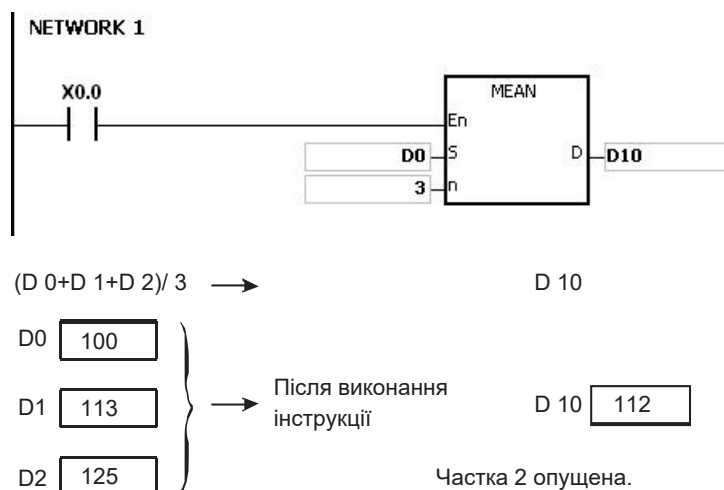
- S** : Перший пристрій
- D** : Пристрій, де зберігається середнє значення
- n** : Кількість пристроїв

Пояснення

1. Ця інструкція підсумовує значення в **n** пристроях, починаючи з пристрою, визначеного **S** , і зберігає середнє значення суми в **D**.
2. Якщо в обчисленні з'являється залишок, інструкція відкидає його.
3. Для 16-розрядних інструкцій **n** має бути між 1–256.
4. Для 32-розрядних інструкцій **n** має бути між 1–128.
5. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій **E**.

приклад

Коли X0.0 увімкнено, інструкція додає значення в трьох регістрах, починаючи з D0. Інструкція ділить суму на 3. Інструкція зберігає частку в D10 і пропускає залишок.



Додаткові зауваження

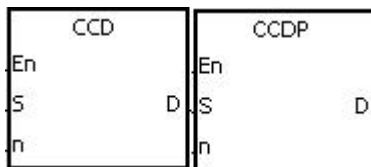
1. Для 16-розрядних інструкцій, якщо **n** не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
2. Для 32-розрядної інструкції, якщо **n** не знаходиться в межах 1–128, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Якщо **S + n - 1** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції			Операнд							функція					
1209		CCD	P	S, D, n							Знаходження суми чека					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



S: Перший пристрій

D: Пристрій, де зберігається сума

n: Кількість фрагментів даних

Пояснення

1/ Протоколи зв'язку використовують функцію перевірки суми для порівняння контрольних сум тих самих даних із різними

випадки або різні представлення даних, щоб перевірити цілісність даних.

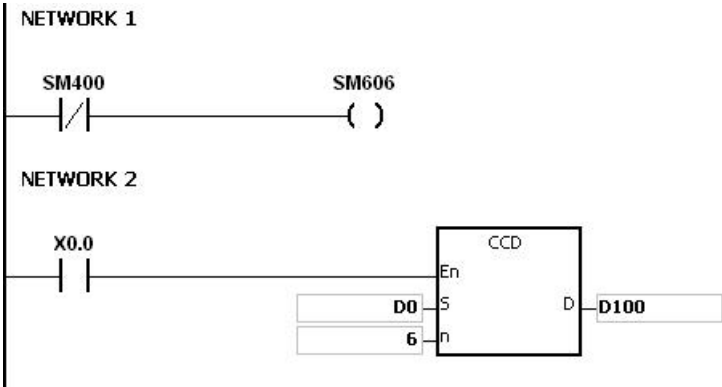
2. Коли SM606 вимкнено, інструкція використовує 16-бітний режим перетворення. Інструкція додає **n** фрагментів даних у регістрах, починаючи з регістра, визначеного **S** (вісім бітів як група). Інструкція зберігає суму в регістрі, визначеному **D**, і зберігає значення бітів парності в **D + 1**.

3. Коли SM606 увімкнено, інструкція використовує режим 8-бітного перетворення. Інструкція додає **n** фрагментів даних у регістрах, починаючи з регістра, заданого **S** (вісім бітів у групі, і тільки молодші вісім бітів є дійсними). Інструкція зберігає суму в регістрі, визначеному **D**, і зберігає значення бітів парності в **D + 1**.

4. Операнд **n** має бути між 1–256.

Приклад 1

- 1. Коли SM606 вимкнено, інструкція використовує 16-бітний режим перетворення.
- 2. Коли X0.0 увімкнено, інструкція додає шість фрагментів даних у D0–D2 (вісім бітів у групі). Інструкція зберігає суму в D100 і зберігає значення бітів парності в D101.



S	Дані
D0 Низький	100 = 0 1 1 0 0 1 0 0
D0 Високий	111 = 0 1 1 0 1 1 1 ① ←
D1 Низький	120 = 0 1 1 1 1 0 0 0
D1 Високий	202 = 1 1 0 0 1 0 1 0
D2 Низький	123 = 0 1 1 1 1 0 1 ① ←
D2 Високий	211 = 1 1 0 1 0 0 1 ① ←
D 100	867
D 101	0 0 0 1 0 0 0 ①

сума

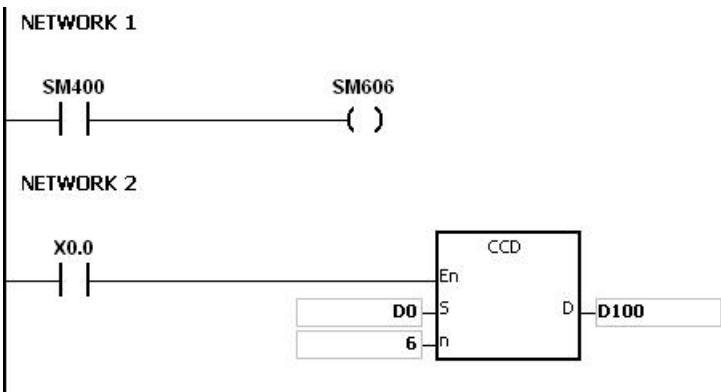
Біт парності встановлюється на 1, якщо кількість одиниць непарна.
Біт парності встановлюється на 0, якщо число одиниць парне.

D 100 0 0 0 0 0 0 1 1 0 1 1 0 0 0 1 1

D 101 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 ← Біти парності

Приклад 2

- 1. Коли SM606 увімкнено, інструкція використовує режим 8-бітного перетворення.
- 2. Коли X0.0 увімкнено, інструкція додає шість фрагментів даних у D0–D5 (вісім бітів у групі). Інструкція зберігає суму в D100 і зберігає значення бітів парності в D101.



S	Дані
D 0 Низький	100 = 0 1 1 0 0 1 0 0
D 1 Низький	111 = 0 1 1 0 1 1 1 ① ←
D 2 Низький	120 = 0 1 1 1 1 0 0 0
D 3 Низький	202 = 1 1 0 0 1 0 1 0
D 4 Низький	123 = 0 1 1 1 1 0 1 ① ←
D 5 Низький	211 = 1 1 0 1 0 0 1 ① ←
D 100	867
D 101	0 0 0 1 0 0 0 ① ←

сума

Біт парності встановлюється на 1, якщо кількість одиниць непарна.
Біт парності встановлюється на 0, якщо число одиниць парне.

D 100: 0 0 0 0 0 0 1 1 0 1 1 0 0 0 1 1

D 101: 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 ← Біти парності

Додаткові зауваження

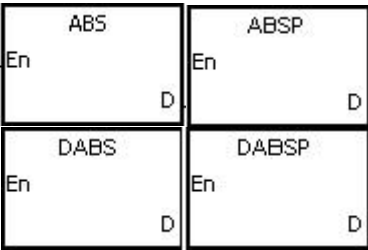
1. Припустимо, SM606 увімкнено. Якщо **S + n - 1** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Припустимо, SM606 вимкнено. Якщо **S + n / 2 - 1** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Якщо **n** не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
4. Якщо ви оголошуєте операнд **D** в ISPSOft, типом даних є ARRAY [2] із WORD/INT.

API		Код інструкції			Операнд					функція							
1210		D	ABS	P	D					Знаходження абсолютної величини							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●	●	●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●	●			●	●						

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
AS		AS		AS	

СИМВОЛ



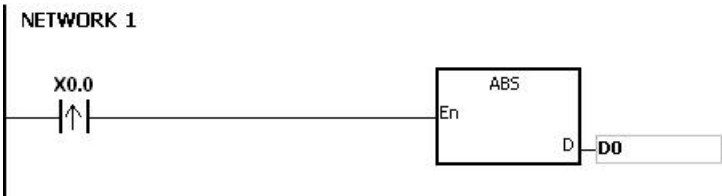
D : Пристрій, що зберігає початкове значення

Пояснення

- Ця інструкція знаходить абсолютне значення значення в пристрої, зазначеному D .
- Загалом використовується імпульсна інструкція ABSP.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.
- Якщо використане тут значення є найменшим від'ємним числом для 16-розрядного чи 32-розрядного числа, ця інструкція не виконано.

приклад

Передбачається, що початкове значення в D0 дорівнює -1234. Коли X0.0 перемикається з OFF на ON, інструкція знаходить абсолютне значення -1234 у D0. Тобто значення в D0 стає 1234 після виконання інструкції.

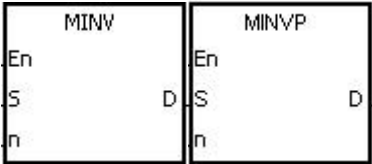


API	Код інструкції				Операнд						функція					
1211		MINV	P	S, D, n						Інвертування бітів матриці						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●							
D		●			●	●		●								
n	●	●			●	●		●	●		○		○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



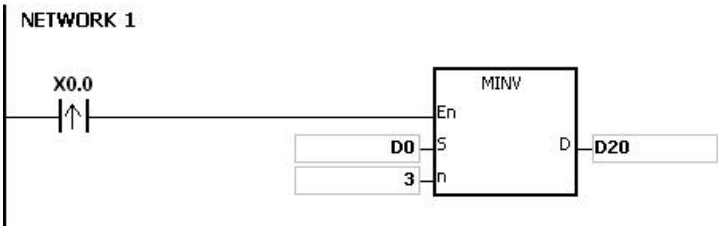
- S** : Джерело матриці
D : Результат операції
n : Довжина масиву

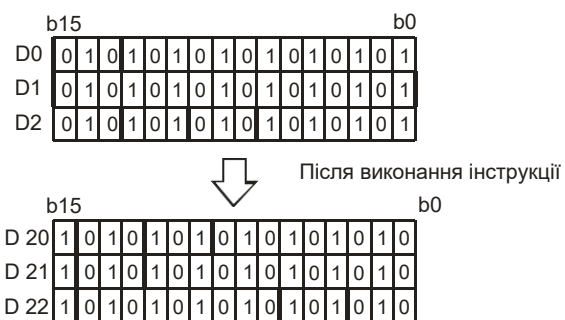
Пояснення

- Ця інструкція інвертує біти в **n** пристроях, починаючи з пристрою, визначеного **S** , і зберігає інверсію результат **D** .
- Операнд **n** має бути між 1–256.

приклад

Коли X0.0 увімкнено, інструкція інвертує біти в трьох 16-бітових регістрах D0–D2 і зберігає результат інверсії в 16-бітових регістрах D20–D22.





Додаткові зауваження

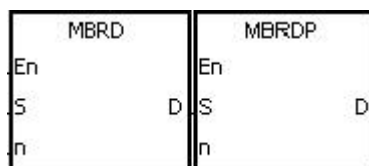
1. Якщо $S + n - 1$ або $D + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд							функція					
1212		MBRD	P	S, n, D							Читання біта матриці					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●							
n	●	●			●	●		●	●		○		○	○		
D		●			●	●		●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



S : Джерело матриці

n : Довжина масиву

D : покажчик

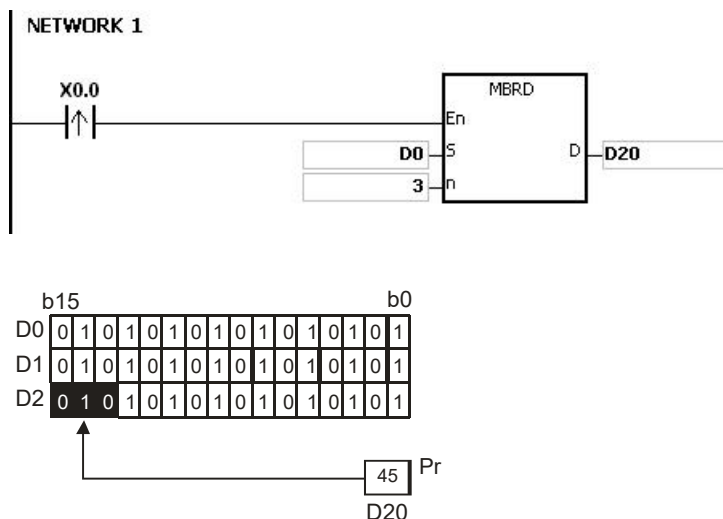
Пояснення

- Ця інструкція перевіряє стан SM613. Якщо SM613 увімкнено, інструкція очищає значення покажчика **D** до нуля. Інструкція зчитує значення біта, заданого значенням покажчика **D**, у SM614, а потім перевіряє стан SM612. Якщо SM612 увімкнено, інструкція збільшує значення покажчика **D** шляхом додавання одиниці.
- Коли інструкція зчитує значення останнього біта, SM608 увімкнено, і інструкція зберігає номер біта в покажчику **D**.
- Операнд **n** має бути між 1–256.
- Ви вказуєте значення вказівника. Значення знаходяться між 0–16 **n** - 1 і відповідають діапазону між b0– b16 **n** - 1. Якщо значення вказівника перевищує діапазон, SM611 встановлюється на одиницю, і інструкція не виконується.

приклад

- Припустимо, що SM613 вимкнено, а SM612 увімкнено, коли X0.0 перемикається з ВИМК. на ОН.
- Припустимо, поточне значення в D20 дорівнює 45. Коли X0.0 тричі перемикається з ВИМК. на УВИМК., інструкція дає наступні результати виконання.

- ❶ Значення в D20 дорівнює 46, SM614 вимкнено, а SM608 вимкнено.
- ❷ Значення в D20 дорівнює 47, SM614 увімкнено, а SM608 вимкнено.
- ❸ Значення в D20 дорівнює 47, SM614 вимкнено, а SM608 увімкнено.



Додаткові зауваження

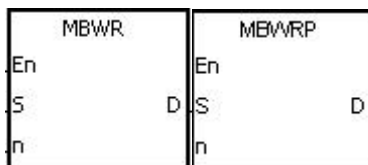
1. Якщо $S + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 16 № 2003.
2. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Прапори інструкцій:

- SM608: Матричне порівняння добігає кінця. Коли порівнюються останні біти, виявляється SM608 ON.
- SM611: Прапор помилки показчика матриці. Коли значення показчика перевищує діапазон порівняння, SM611 ON.
- SM612: Прапор збільшення показчика матриці. Поточне значення показчика збільшується на одиницю.
- SM613: Прапор очищення показчика матриці. Поточне значення вказівника скидається на нуль.
- SM614: Прапор переносу для повороту/зсуву/виведення матриці.

API	Код інструкції				Операнд						функція					
1213		MBWR		P	S, n, D						Запис матричного біта					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S		●			●	●		●								
n	●	●			●	●		●	●		○		○	○		
D		●			●	●		●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

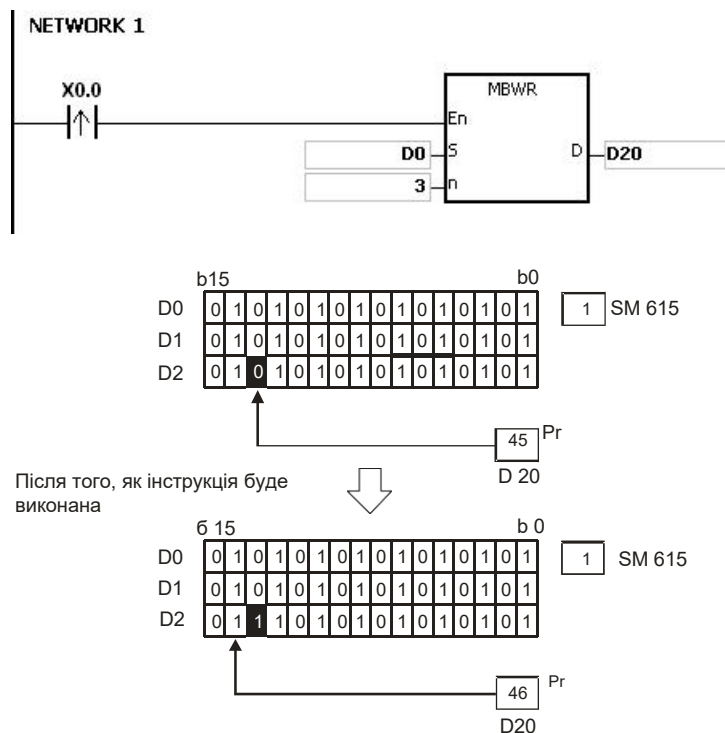
Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ**S** : Джерело матриці**n** : Довжина масиву**D** : покажчик**Пояснення**

- Ця інструкція перевіряє стан SM613. Якщо SM613 увімкнено, інструкція очищає значення покажчика **D** до 0. Інструкція записує стан SM615 у біт, визначений значенням покажчика **D**, а потім перевіряє стан SM612. Якщо SM612 увімкнено, інструкція збільшує значення в покажчику **D** на одиницю.
- Коли інструкція записує стан SM615 в останній біт, встановлює SM608 у положення ON і записує номер біта в вказівник **D**. _ Якщо значення покажчика **D** перевищує діапазон, SM611 увімкнено.
- Операнд **n** має бути між 1–256.
- Ви вказуєте значення вказівника. Значення знаходяться між 0–16 **n** - 1 і відповідають діапазону між b0– b16 **n** - 1. Якщо значення вказівника перевищує діапазон, SM611 встановлюється на одиницю, і інструкція не виконується.

приклад

- Припустимо, що SM613 вимкнено, а SM612 увімкнено, коли X0.0 перемикається з ВИМК. на ON.
- Припустимо, поточне значення в D20 дорівнює 45. Коли X0.0 один раз перемикається з ВИМК. на УВИМК., інструкція дає результат виконання, показаний нижче. Коли значення в D20 дорівнює 45, SM615 вимкнено, а SM608 вимкнено.



Додаткові зауваження

1. Якщо $S + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n не між 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Позначки інструкцій:

SM608: Матричне порівняння добігає кінця. Коли порівнюються останні біти, SM608 увімкнено.

SM611: Прапор помилки покажчика матриці. Коли значення покажчика перевищує діапазон порівняння, SM611 увімкнено.

SM612: Прапор збільшення покажчика матриці. Поточне значення покажчика збільшується на одиницю.

SM613: Прапор очищення покажчика матриці. Поточне значення вказівника скидається на нуль.

SM615: Позичити позначку для зсуву/виведення матриці.

API	Код інструкції			Операнд							функція						
1214		MBC	P	S, n, D							Підрахунок бітів зі значенням нуль або одиниця						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S		●			●	●		●									
n	●	●			●	●		●	●		○		○	○			
D		●			●	●		●			○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



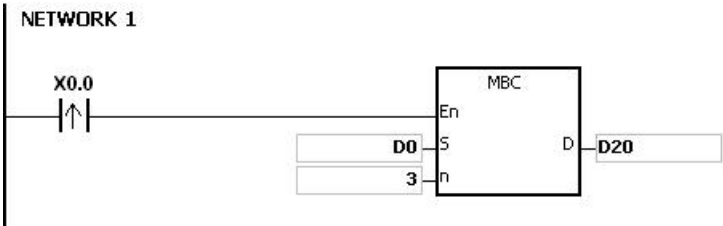
- S : Джерело матриці
- n : Довжина масиву
- D : Результат операції

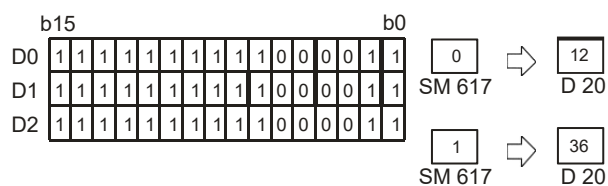
Пояснення

- Ця інструкція підраховує біти зі значенням одиниця або нуль у n пристроях, починаючи з пристрою, зазначеного S . Інструкція зберігає результат операції в D .
- Коли SM617 увімкнено, інструкція підраховує біти зі значенням одиниця. Коли SM617 вимкнено, інструкція підраховує біти зі значенням 0. Коли результат операції дорівнює 0, SM618 увімкнено..
- Значення в n має бути в діапазоні від 1 до 256.

приклад

Припустимо, SM617 увімкнено. Коли X0.0 увімкнено, інструкція підраховує біти зі значенням один і зберігає результат операції в D20. Припустимо, SM617 вимкнено. Коли X0.0 увімкнено, інструкція підраховує біти з нульовим значенням і зберігає результат операції в D20.





Додаткові зауваження

1. Якщо $S + n - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n не в діапазоні від 1 до 256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Прапори інструкцій:

SM617: Підраховуються біти зі значенням нуль або одиниця.

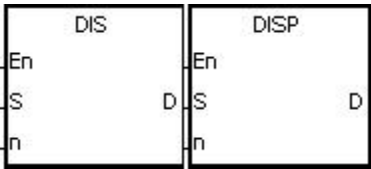
SM618: ON, коли результат підрахунку матриці дорівнює 0.

API	Код інструкції				Операнд						функція					
1215		DIS		P	S, n, D						Роз'єднання 16-бітних даних					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●		○	○				
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

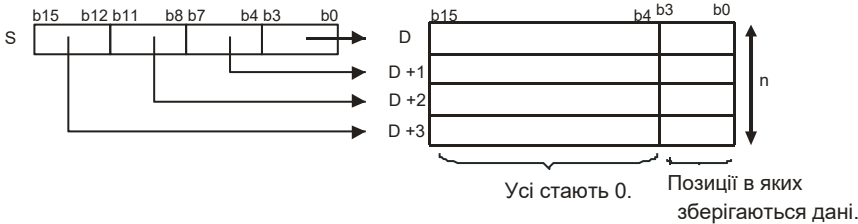
СИМВОЛ



- S : Джерело даних
n : Кількість пристроїв
D : Результат операції

Пояснення

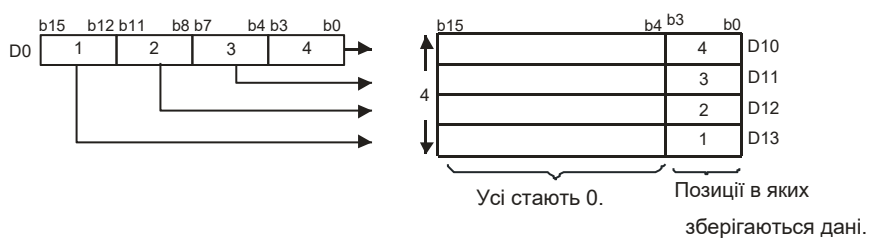
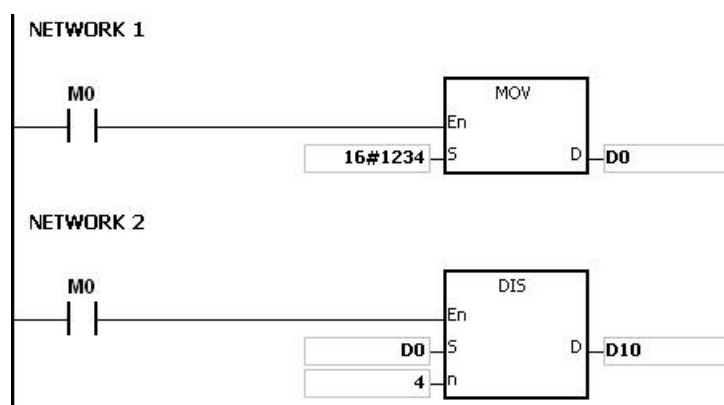
1. Ця інструкція ділить 16-бітне значення в регістрі, визначеному S, на чотири групи (чотири біти в групі) і зберігає ці групи в молодших чотирьох бітах кожного регістру (діапазон регістрів від D до D + (n - 1)).



2. Значення в n має бути між 1–4.

приклад

Припустимо, значення в D0 дорівнює 16#1234. Коли M0 увімкнено, інструкція ділить значення в D0 на чотири групи (чотири біти в групі) і зберігає ці групи в молодших чотирьох бітах у кожному регістрі (діапазон регістрів від D10 до D13).



Додаткові зауваження

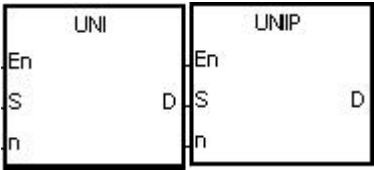
1. Якщо $D - D + (n - 1)$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо n не між 1–4, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд						функція						
1216		UNI	P	S, n, D						Об'єднання 16-бітних даних						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●							
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

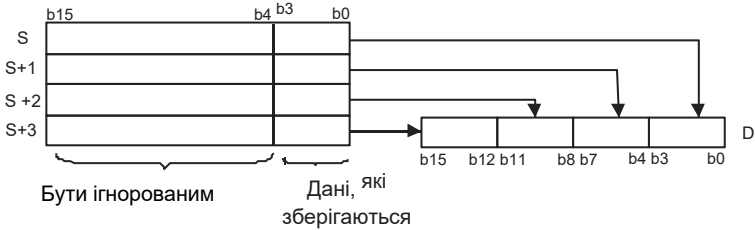
СИМВОЛ



- S : Джерело даних
- n : Довжина даних
- D : Результат операції

Пояснення

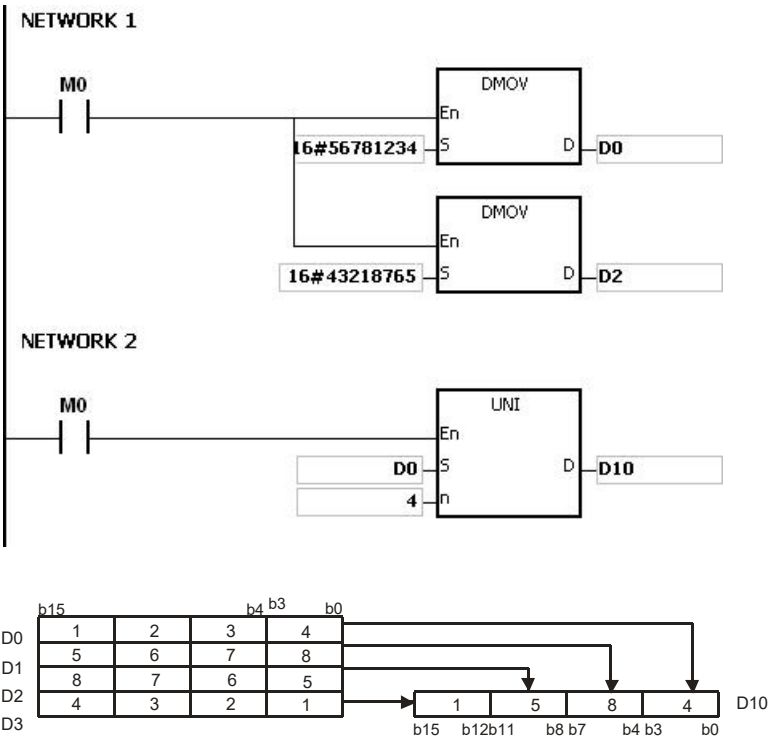
1. Ця інструкція розділяє 16-бітні значення в регістрах, визначених S – S + (n -1) на групи (по чотири біти в групі), і зберігає кожну групу, яка знаходиться в b0 – b3, у регістрі, визначеному D (b0–b15).



2. Значення в n має бути в діапазоні від 1 до 4.

приклад

Припустимо, значення в D0 – D3 16#1234, 16#5678, 16#8765 і 16#4321 відповідно. Коли M0 увімкнено, інструкція UNI розділяє значення в D0–D3 на групи (по чотири біти в групі) і зберігає кожну групу в b0 – b3 у D10(b0–b15).



Додаткові зауваження

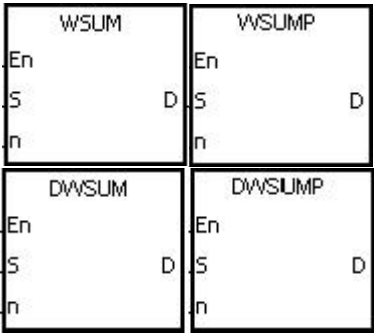
1. Якщо **S** до **S + (n -1)** перевищують діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо **n** не між 1 – 4, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

API	Код інструкції			Операнд						функція							
1217	D	WSUM	P	S, n, D						Отримання суми							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●								
n	●	●			●	●	●	●	●		O	O	O	O			
D		●			●	●	●	●			O						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
n		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

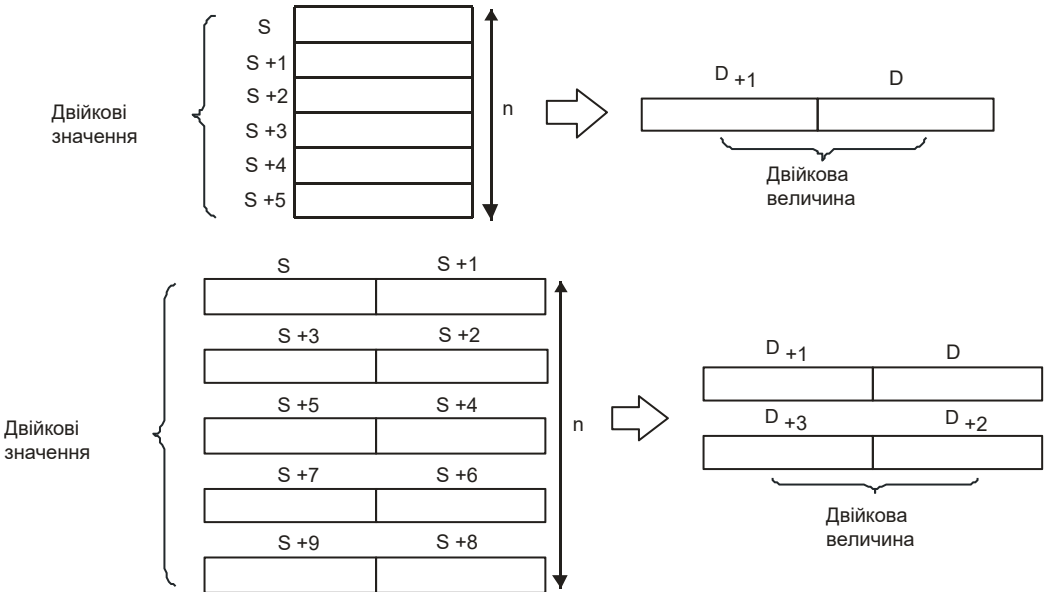
СИМВОЛ



- S: Джерело даних
- n: Довжина даних
- D: Результат операції

Пояснення

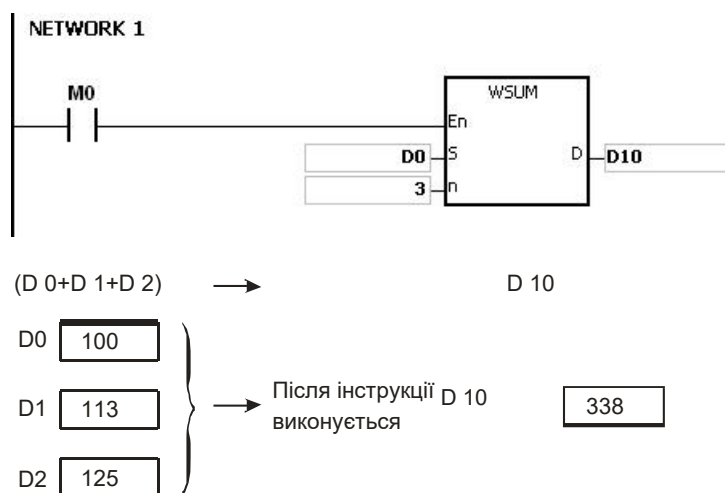
1. Інструкція додає десяткові значення зі знаком у S до S + n - 1 і зберігає суму в регістрі, визначеному D .



- Для 16-бітних інструкцій значення в **n** має бути в межах від 1 до 256.
- Для 32-розрядних інструкцій значення в **n** має бути в межах від 1 до 128.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

приклад

Інструкція WSUM додає значення в D0–D2 і зберігає суму (32 біти) у D10.



6

Додаткові зауваження

- Для 16-розрядних інструкцій значення в **n** не знаходиться в межах від 1 до 256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
- Для 32-розрядних інструкцій значення в **n** не знаходиться в межах від 1 до 128, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
- Якщо **S + n - 1** або **D** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
- Для 16-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSOft, тип даних буде DWORD або ARRAY [2] із WORD.
- Для 32-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSOft, типом даних є ARRAY [2] із DWORD.

API		Код інструкції			Операнд						функція					
1221		D	LIMIT	P	S ₁ , S ₂ , S ₃ , D						Обмеження значення в межах					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		○	○	○	○		
S ₂	●	●			●	●	●	●	●		○	○	○	○		
S ₃	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●						
S ₂		●	●			●	●						
S ₃		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу					16-бітна інструкція					32-розрядна інструкція				
AS					AS					AS				

СИМВОЛ

LIMIT	LIMITP
En	En
S1	S1
S2	S2
S3	S3
D	D

DLIMIT	DLIMITP
En	En
S1	S1
S2	S2
S3	S3
D	D

- S₁: Мінімальне вихідне значення
- S₂: Максимальне вихідне значення
- S₃: Вхідне значення
- D: Вихідне значення

Пояснення

1. Інструкція порівнює вхідне значення в S₃ з мінімальним вихідним значенням в S₁ і максимальним вихідним значенням в S₂ і зберігає результат порівняння в D .

Якщо вхідне значення в S3 менше, ніж мінімальне вихідне значення в S1, інструкції зберігають мінімум вихідне значення S₁ у D .

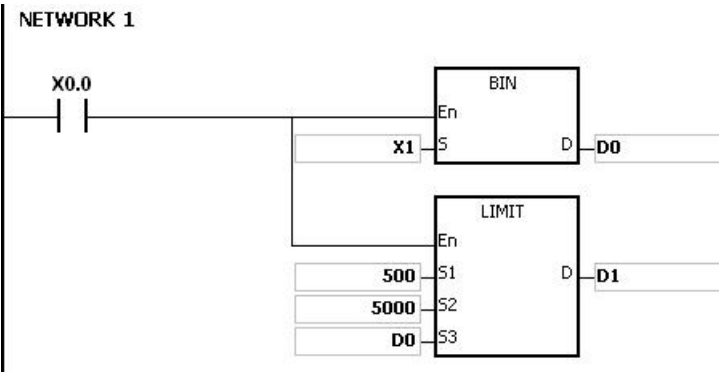
Якщо вхідне значення в S3 **більше** , ніж максимальне вихідне значення в S2 , інструкція зберігає максимальне **S2** в D.

Якщо вхідне значення в S₃ знаходиться між мінімальним вихідним значенням S₁ і максимальним вихідним значенням S₂ , інструкція зберігає вхідне значення S₃ в D .

- Якщо мінімальне вихідне значення в S_1 більше максимального вихідного значення в S_2 , інструкція не виконується.
2. Тільки 32-бітні інструкції можуть використовувати 32-бітний лічильник, але не пристрій E.

приклад

- Коли X0.0 увімкнено, інструкція перетворює стан X1 у двійкове значення та зберігає результат перетворення в D0. Потім інструкція порівнює значення в D0 з 500 і 5000 і зберігає результат порівняння в D1.



Мінімальний вихід значення	Максимальний вихід значення	Вихідне значення в D0	функція	Вихідне значення в D1
500	5000	499	$D0 < 500$	500
		5001	$D0 > 5000$	5000
		600	$500 \leq D0 \leq 5000$	600

Додаткові зауваження

Якщо мінімальне вихідне значення в $S1$ перевищує максимальне вихідне значення в $S2$, інструкція не виконується, $SM0$ увімкнено, а код помилки в $SR0$ — 16#2003.

API	Код інструкції			Операнд							функція					
1222	D	BAND	P	S ₁ , S ₂ , S ₃ , D							Контроль зони нечутливості					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		○	○	○	○		
S ₂	●	●			●	●	●	●	●		○	○	○	○		
S ₃	●	●			●	●	●	●	●		○	○	○	○		
D		●			●	●	●	●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●	●			●	●						
S ₂		●	●			●	●						
S ₃		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція		
AS			AS			AS		

СИМВОЛ

BAND		BANDP	
En		En	
S1	D	S1	D
S2		S2	
S3		S3	

DBAND		DBANDP	
En		En	
S1	D	S1	D
S2		S2	
S3		S3	

- S_1 : Мінімальне значення зони нечутливості
- S_2 : Максимальне значення зони нечутливості
- S_3 : Вхідне значення
- D: Вихідне значення

Пояснення

1. Ця інструкція віднімає мінімальне значення зони нечутливості в S_1 або максимальне значення зони нечутливості в S_2 від вхідного значення в S_3 і зберігає різницю в D .

Якщо вхідне значення в S_3 менше , ніж мінімальне значення зони нечутливості в S_1 , інструкція віднімає S_1 з S_3 і зберігає різницю в D .

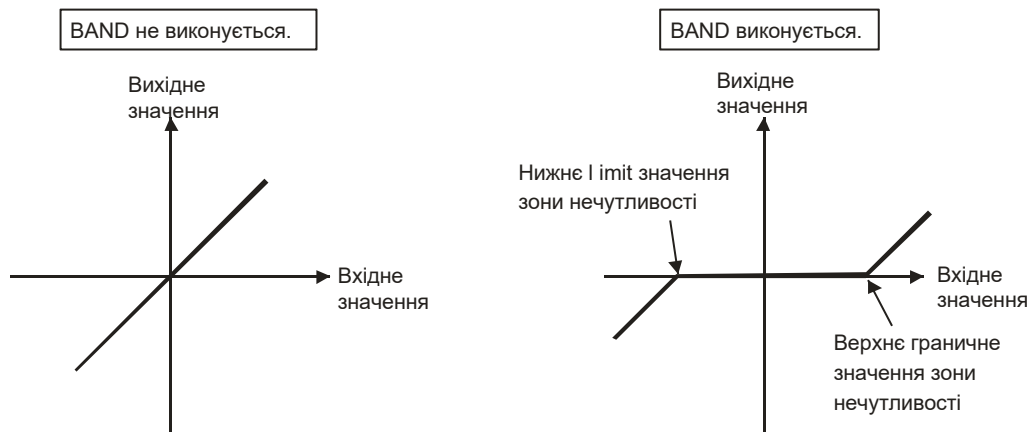
Якщо вхідне значення в S_3 більше , ніж максимальне значення зони нечутливості в S_2 , інструкція віднімає S_2 з S_3 і зберігає різницю в D .

Якщо вхідне значення в S_3 знаходиться між мінімальним значенням зони нечутливості в S_1 і максимальним значенням зони нечутливості в S_2 , інструкція зберігає нуль у D .

Якщо мінімальне значення зони нечутливості в S_1 більше, ніж максимальне значення зони нечутливості в S_2 , інструкція не виконується.

2. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

3. Наступні графіки показують, як ця інструкція використовує зону нечутливості.



4. Мінімальне значення зони нечутливості в S_1 , максимальне значення зони нечутливості в S_2 , вхідне значення в S_3 і вихідне значення в D має бути в діапазоні, описаному нижче.

5. Для інструкції BAND мінімальне значення зони нечутливості в S_1 , максимальне значення зони нечутливості в S_2 , вхідне значення в S_3 і вихідне значення в D мають бути між -32768 і 32767. Припустимо, що мінімальне значення зони нечутливості в S_1 дорівнює 10, а максимальне значення зони нечутливості в S_3 дорівнює -32768. Інструкція обчислює вихідне значення в D наступним чином.

$$\text{Вихідне значення в } D = -32768 - 10 = 16\#8000 - 16\#000A = 16\#7FF6 = 32758$$

6. Для інструкції DBAND мінімальне значення зони нечутливості в S_1 , максимальне значення зони нечутливості в S_2 , вхідне значення в S_3 і вихідне значення в D мають бути від -2147483648 до 2147483647. Припустимо, мінімальне значення зони нечутливості в $(S_1 + 1, S_1)$ становить 1000, а максимальне значення зони нечутливості в $(S_3 + 1, S_3)$ становить -2147483648. Інструкція обчислює вихідне значення в $(D + 1, D)$ наступним чином.

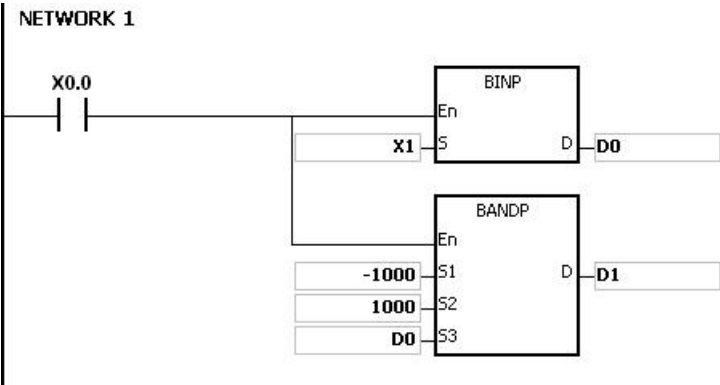
$$\text{Вихідне значення в } (D + 1, D)$$

$$= -2147483648 - 1000 = 16\#80000000 - 16\#000003E8 = 16\#7FFFC18$$

$$= 2147482648$$

Приклад 1

Коли X0.0 увімкнено, інструкція віднімає -1000 або 1000 із двійкового десяткового значення в X1 і зберігає різницю в D1.



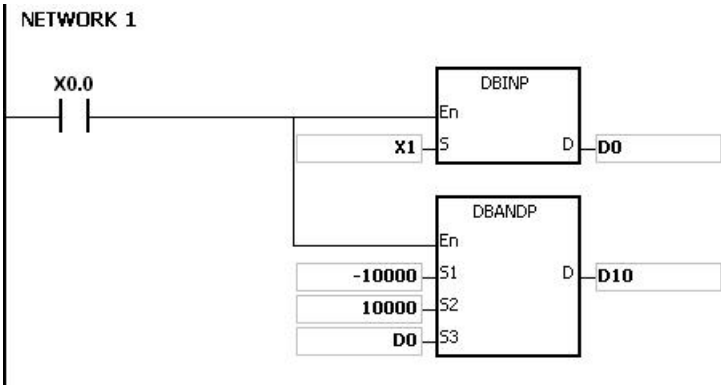
У наступній таблиці наведено результати виконання.

Мінімальне значення зони нечутливості	Максимум значення зони нечутливості	Введіть значення в D0	функція	Вихідне значення в D1
-1000	1000	-1200	$D0 < -1000 \Rightarrow D1 = D0 - (-1000)$	-200
		1200	$D0 > 1000 \Rightarrow D1 = D0 - 1000$	200
		500	$-1000 \leq D0 \leq 1000 \Rightarrow D0 = 0$	0

6

Приклад 2

Коли X0.0 увімкнено, інструкція віднімає -10000 або 10000 із двійкового десяткового значення в (X2, X1) і зберігає різницю в (D11, D10).



У наступній таблиці наведено результати виконання.

Мінімальне значення зони нечутливості	Максимум значення зони нечутливості	Вхідне значення в (D1, D0)	функція	Вихідне значення в (D11, D10)
-10000	10000	-12000	$(D1 \cdot D0) < -10000$ $\Rightarrow (D11 \cdot D10)$ $= (D1 \cdot D0) - (-10000)$	-2000
		12000	$(D1 \cdot D0) > 10000$ $\Rightarrow (D11 \cdot D10)$ $= (D1 \cdot D0) - 10000$	2000
		5000	$-10000 \leq (D1 \cdot D0) \leq 10000$ $\Rightarrow (D1 \cdot D0) = 0$	0

Додаткові зауваження

Якщо мінімальне значення зони нечутливості в S_1 перевищує максимальне значення зони нечутливості в S_2 , інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд							функція						
1223	D	ZONE	P	S_1, S_2, S_3, D							Контроль зони						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●	●	●	●		○	○	○	○			
S_2	●	●			●	●	●	●	●		○	○	○	○			
S_3	●	●			●	●	●	●	●		○	○	○	○			
D		●			●	●	●	●			○	○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
S_3		●	●			●	●						
D		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

ZONE		ZONEP	
En		En	
S1	D	S1	D
S2		S2	
S3		S3	

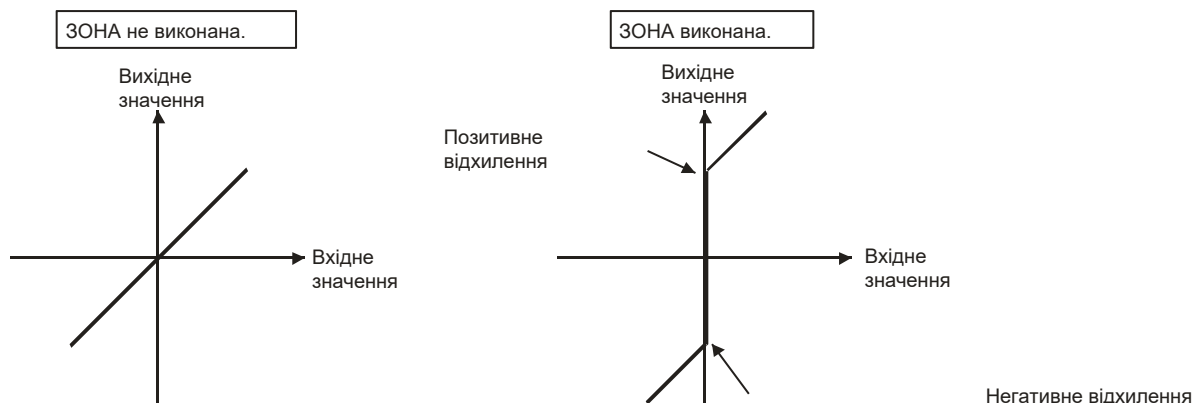
DZONE		DZONEP	
En		En	
S1	D	S1	D
S2		S2	
S3		S3	

- S_1 : Негативне відхилення
- S_2 : Позитивне відхилення
- S_3 : Вхідне значення
- D: Вихідне значення

Пояснення

1. Ця інструкція додає негативне відхилення в S_1 або позитивне відхилення в S_2 до вхідного значення в S_3 і зберігає суму в D .
- Якщо вхідне значення в S_3 менше 0, інструкція додає негативне відхилення в S_1 до вхідного значення в S_3 , і зберігає суму в D .
- Якщо вхідне значення в S_3 більше 0, інструкція додає позитивне відхилення в S_2 до вхідного значення в S_3 і зберігає суму в D .
- Якщо вхідне значення в S_3 дорівнює нулю, інструкція зберігає нуль в D .

2. Наступні графіки показують, як ця інструкція використовує зону:



3. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

4. Від'ємне відхилення в S_1 , додатне відхилення в S_2 , вхідне значення в S_3 і вихідне значення в D мають бути в межах діапазону, описаного нижче.

- Для інструкції ZONE від'ємне відхилення в S_1 , додатне відхилення в S_2 , вхідне значення в S_3 і вихідне значення в D мають бути від -32768 до 32767. Припустимо, від'ємне відхилення в S_1 становить -100 і вхідне значення в S_3 дорівнює -32768. Інструкція обчислює вихідне значення в D наступним чином.

$$\text{Вихідне значення в } D = (-32768) + (-100) = 16\#8000 + 16\#FF9C = 16\#7F9C = 32668$$

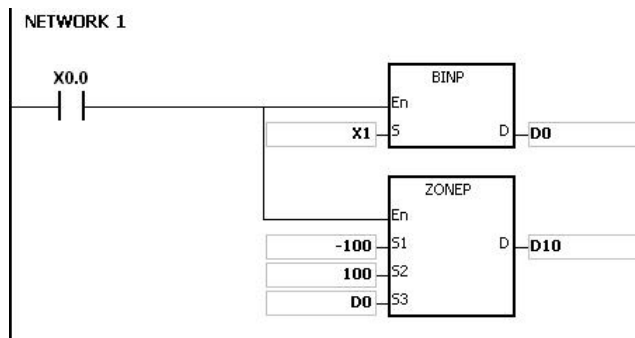
- Для інструкції DZONE негативне відхилення в S_1 , позитивне відхилення в S_2 , вхідне значення в S_3 , і вихідне значення в D має бути від -2147483648 до 2147483647. Припустимо, що негативне відхилення в $(S_1 + 1, S_1)$ дорівнює -1000, а вхідне значення в $(S_3 + 1, S_3)$ дорівнює -2147483648. Інструкція обчислює вихідне значення в $(D + 1, D)$ наступним чином.

Вихідне значення в $(D + 1, D)$

$$= -2147483648 + (-1000) = 16\#80000000 + 16\#FFFFFFC18 = 16\#7FFFFFFC18 = 2147482648$$

Приклад 1

Коли X0.0 увімкнено, інструкція додає -100 або 100 до двійкового десяткового значення в X1 і зберігає суму в D10.

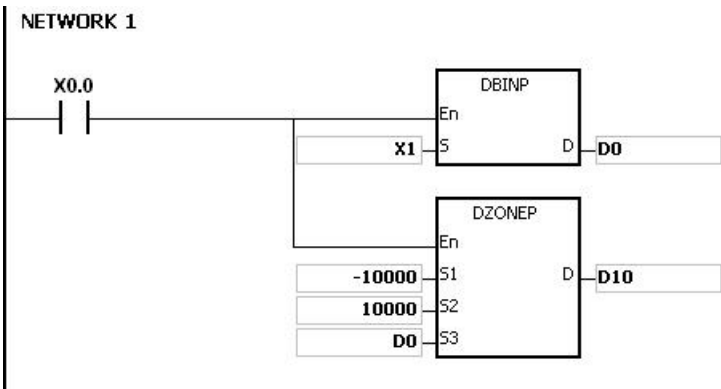


У наступній таблиці наведено результати виконання.

Негативний відхилення	Позитивний відхилення	Введіть значення в D0	функція	Вихідне значення в D10
-100	100	-10	$D0 < 0 \Rightarrow D10 = (-10) + (-100)$	-110
		0	$D0 = 0 \Rightarrow D10 = 0$	0
		50	$D0 > 0 \Rightarrow D10 = 50 + 100$	150

Приклад 2

Коли X0.0 увімкнено, інструкція додає -10000 або 10000 до двійкового десяткового значення в (X2, X1) і зберігає суму в (D11, D10).



6

У наступній таблиці наведено результати виконання.

Негативне відхилення	Позитивне відхилення	Вхідне значення в (D1, D0)	функція	Вихідне значення в (D11, D10)
-10000	10000	-10	$(D1 \cdot D0) < 0$ $\Rightarrow (D11 \cdot D10)$ $= (-10) + (-10000)$	-10010
		0	$(D1 \cdot D0) = 0$ $\Rightarrow (D11 \cdot D10) = 0$	0
		50	$(D1 \cdot D0) > 0$ $\Rightarrow (D11 \cdot D10) = 50 + 10000$	10050

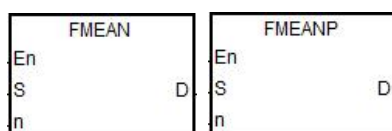
API	Код інструкції			Операнд							функція						
1224		FMEAN	P	S, D, n							Знаходження середнього значення чисел з плаваючою комою						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S							●	●								
n							●	●					○	○		
D							●	●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
n			●				●						
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ



S : Перший пристрій

D : Пристрій, де зберігається середнє значення

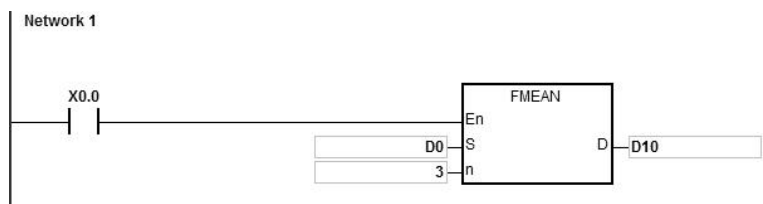
n : Кількість пристроїв

Пояснення

- Ця інструкція додає плаваючі крапки одинарної точності в **n** пристроях, починаючи з пристрою, визначеного **S**, ділить суму на значення в **n**, а потім зберігає середнє значення суми в **D**.
- Для 16-бітних інструкцій значення в **n** має бути між 1–256,
- Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор), SM602 (переносний прапор):
 - Коли результат операції дорівнює нулю, SM600 увімкнено. В іншому випадку він OFF.
 - Якщо значення під час додавання або абсолютний результат операції менші за число з плаваючою комою, яке може бути показано, D=16#FF800000 і прапор запозичення SM601 увімкнено.
 - Якщо значення під час додавання або абсолютний результат операції перевищує число з плаваючою комою, яке може бути показано, D=16#7F800000 і прапор переносу SM602 увімкнено.

приклад

Коли X0.0 увімкнено, інструкція додає значення 3 плаваючих точок одинарної точності в (D1, D0), (D3, D2), (D5, D4), а потім ділить результат додавання на 3, а потім зберігає результат в (D11, D10).



$$[(D1, D0) + (D3, D2) + (D5, D4)] / 3 \rightarrow (D11, D10)$$

(D1, D0) 100,1 } Після того, як інструкція буде виконана.

(D3, D2) 113.2 } → (D11, D10) 112.2

(D5, D4) 123.3 }

Додаткові зауваження

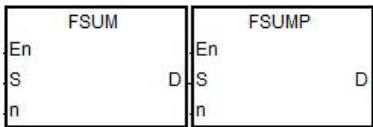
1. Якщо значення в **n** не знаходиться в межах 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
2. Якщо **S + 2* n -1** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо значення в **S** перевищує діапазон чисел з плаваючою комою, який можна показати, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції			Операнд							функція						
1225		FSUM	P	S, n, D							Знаходження суми чисел з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S							●	●									
n							●	●					○	○			
D							●	●			○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
n			●				●						
D									●				

Інструкція пульсу			16-бітна інструкція		32-розрядна інструкція	
AS			-		AS	

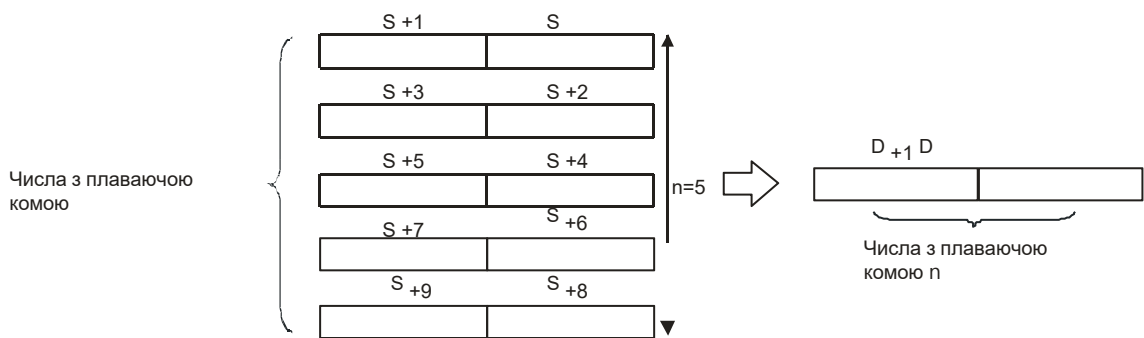
СИМВОЛ



- S : Джерело даних
- n : Довжина даних
- D : Результат операції

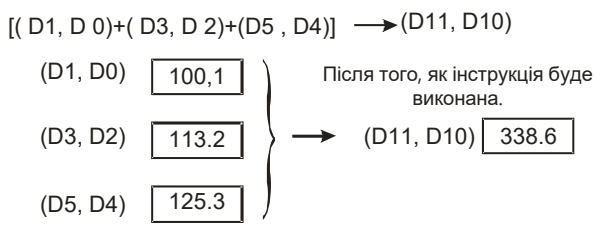
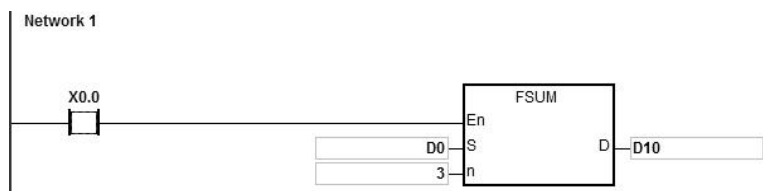
Пояснення

- Ця інструкція додає плаваючі крапки одинарної точності в n пристроях, починаючи з пристрою, визначеного S , потім зберігає суму в D .
- Для 16-бітних інструкцій значення в n має бути між 1–256,
- Прапори інструкцій: SM600 (нульовий прапор), SM601 (запозичений прапор), SM602 (переносний прапор):
 - Коли результат операції дорівнює нулю, SM600 увімкнено. В іншому випадку він OFF.
 - Якщо значення під час додавання або абсолютний результат операції менші за число з плаваючою комою, яке може бути показано, D=16#FF800000 і прапор запозичення SM601 увімкнено.
 - Якщо значення під час додавання або абсолютний результат операції перевищує число з плаваючою комою, яке може бути показано, D=16#7F800000 і прапор переносу SM602 увімкнено.



приклад

Інструкція FSUM додає значення 3 плаваючих точок одинарної точності в (D1, D0), (D3, D2), (D5, D4) і зберігає результат у (D11, D10).



Додаткові зауваження

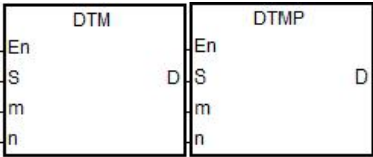
1. Якщо значення в **n** не знаходиться в межах 1–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
2. Якщо **S + 2* n - 1** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо значення в **S** перевищує діапазон чисел з плаваючою комою, які можна показати, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Інструкція				Операнд						опис						
1226		DTM	P		S , D , m , n						Перетворення та переміщення даних						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●									
D		●			●	●		●									
m								●					○	○			
n								●					○	○			

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							
m		●				●							
n		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



- S : Пристрій вихідних даних
- D : Пристрій, на якому зберігається результат перетворення
- m : Вибір режиму конвертації
- n : Довжина даних для виконання

Пояснення

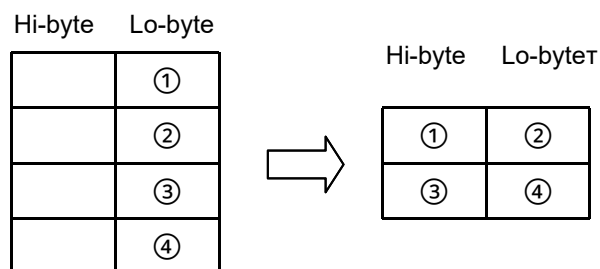
1. Параметр **m** призначений для вибору режиму перетворення з наступної таблиці. Дивіться докладну інформацію про режими в наступних розділах. Якщо значення параметра не є одним із значень у наведеній нижче таблиці, перетворення чи переміщення даних не відбуватиметься, а також повідомлення про помилку.

Параметр	опис
0	8-бітні дані перетворюються на 16-бітні (старші 8 біт, молодші 8 біт)
1	8-бітні дані перетворюються на 16-бітні (молодші 8 біт, старші 8 біт)
2	16-бітні дані (старші 8 біт, молодші 8 біт), перетворені на 8-бітні дані
3	16-бітні дані (молодші 8 біт, старші 8 біт) перетворюються на 8-бітні дані
4	8-бітні шістнадцяткові дані (старші 4 біти, молодші 4 біти), перетворені в дані ASCII
5	8-бітні шістнадцяткові дані (молодші 4 біти, старші 4 біти), перетворені в дані ASCII.
6	8-бітні дані ASCII, перетворені в шістнадцяткові (старші 4 біти, молодші 4 біти).)
7	8-бітні дані ASCII, перетворені в шістнадцяткові дані (молодші 4 біти, старші 4 біти).
18	Дані рядкового типу, перетворені на дані з плаваючою комою (доступно для версії FW 1.08 або новішої)
19	Дані з плаваючою комою, перетворені на дані рядкового типу (доступно для FW V1.08 або новішої версії)
42	Обчисліть тривалість дугової інтерполяції відносного положення 2 осей за годинниковою стрілкою
43	Обчисліть тривалість дугової інтерполяції 2-осьового відносного положення проти годинникової стрілки
44	Налаштуйте вбудований час фільтрації вхідних точок для ПЛК
46	Розрахувати час виведення імпульсу відносного положення при прискоренні та уповільненні (доступно для FW V1.08 або новішої)

2. **n** – значення параметра довжини даних. Діапазон значень налаштування становить 1~256. Якщо вхідне значення перевищує діапазон, ПЛК виконає інструкцію з мінімальним або максимальним значенням.
3. Режими перетворення та режими переміщення пояснюються нижче.

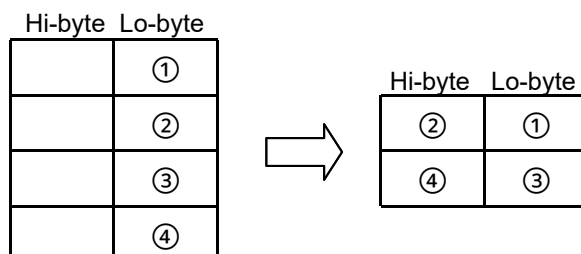
● **Коли m =0:**

Якщо **n =4**, 8-бітові дані перетворюються на 16-бітові (старші 8 бітів, молодші 8 бітів), перетворення відбувається так, як показано на наступному малюнку.



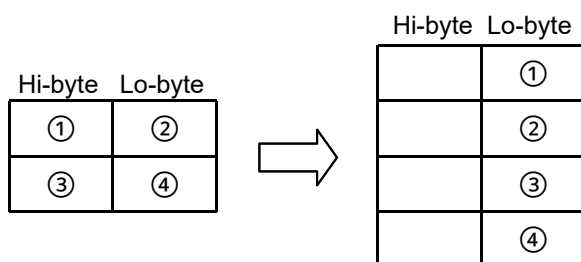
● **Коли $m = 1$:**

Якщо $n = 4$, 8-бітні дані перетворюються на 16-бітові дані (моладші 8-біт, старші 8-біт), перетворення відбувається так, як показано на наступному малюнку.



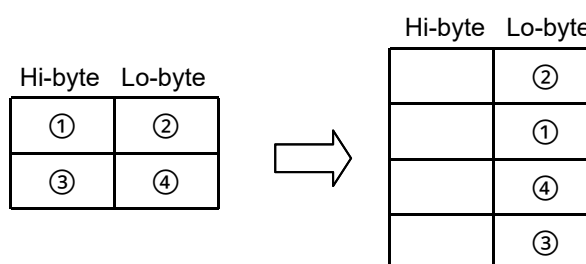
● **Коли $m = 2$:**

Якщо $n = 4$, 16-бітні дані (старші 8-біт, молодші 8-біт) перетворюються на 8-бітові дані, перетворення відбувається так, як показано на наступному малюнку.



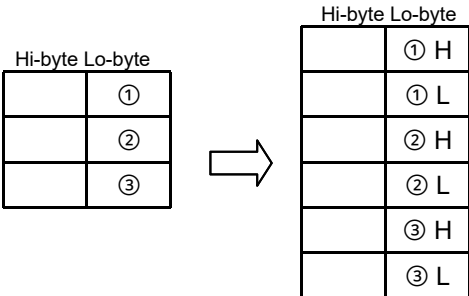
● **Коли $m = 3$:**

Якщо $n = 2$, 16-бітні дані (моладші 8-біт, старші 8-біт) перетворюються на 8-бітові дані, перетворення відбувається так, як показано на наступному малюнку.



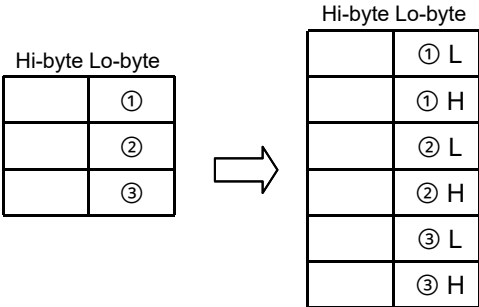
● Коли $m = 4$:

Якщо $n = 3$, 8-бітні шістнадцяткові дані (старші 4 біти, молодші 4 біти) перетворюються на дані ASCII, і перетворення відбувається так, як показано на наступному малюнку.



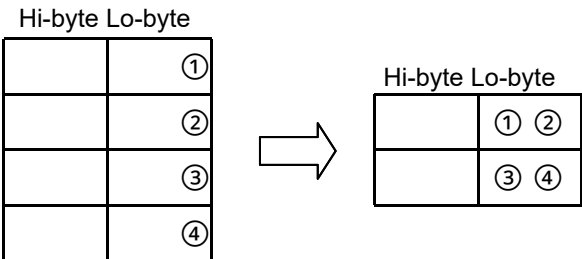
● Коли $m = 5$:

Якщо $n = 3$, 8-бітні шістнадцяткові дані (молодші 4 біти, старші 4 біти) перетворюються на дані ASCII, перетворення відбувається так, як показано на наступному малюнку.



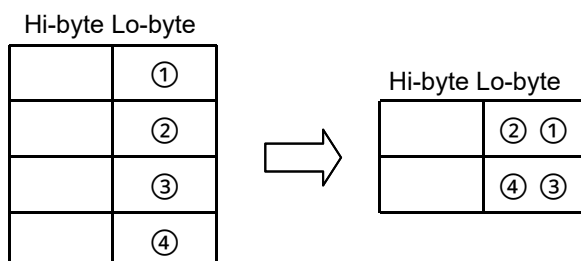
● Коли $m = 6$:

Якщо $n = 4$, 8-бітні дані ASCII перетворюються на шістнадцяткові дані (старші 4 біти, молодші 4 біти), перетворення відбувається так, як показано на наступному малюнку. Значення перетворення ASCII можуть бути: 0 ~ 9 (0x30 ~ 0x39), A ~ F (0x41 ~ 0x46), a ~ f (0x61 ~ 0x66).



● **Коли $m = 7$:**

Якщо $n = 4$, 8-бітні дані ASCII перетворюються на шістнадцяткові дані (молодші 4 біти, старші 4 біти), перетворення відбувається так, як показано на наступному малюнку.

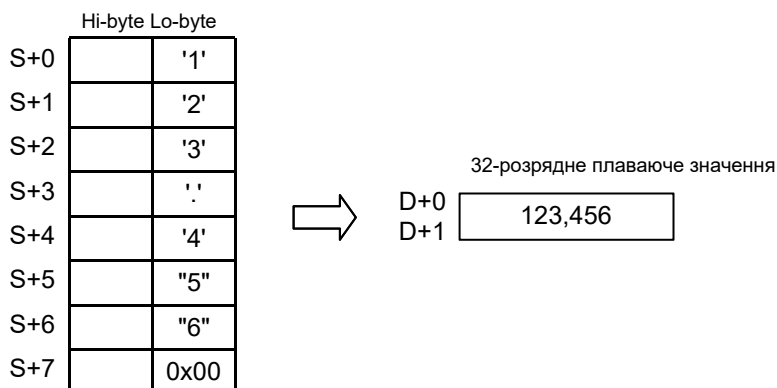


● **Коли $m = 18$:**

Перетворіть значення рядка в вихідному пристрої S (Lo-byte) на значення з плаваючою комою (операнд n ; десяткова крапка виключена) і збережіть це значення в пристрої D.

Примітка:

- Значення параметра в операнді n має бути менше 8. Якщо встановлене значення перевищує максимальне значення 8, воно розглядатиметься як максимальне значення 8. Наприклад, якщо $n = k6$, коли ви перетворюєте рядок 123,45678 у значення з плаваючою комою, результатом буде 123,456.
- Якщо вихідне значення містить не лише цифрові символи, нецифрові символи розглядатимуться як 0, коли вони відображаються перед десятковою комою, і розглядатимуться як кінцеві символи, коли вони відображаються після десяткової коми. Наприклад, якщо $n = k6$, коли ви перетворюєте рядок BC12.4K59 у значення з плаваючою комою, результатом буде 0012.4.
- Якщо в рядку немає десяткової коми, рядок перетворюється безпосередньо на значення з плаваючою комою. Перегляньте малюнок нижче для довідки.



● **Коли $m = 19$:**

Перетворіть значення з плаваючою комою в вихідному пристрої S (Lo-byte) на рядкове значення (операнд n; десяткова крапка виключається) і збережіть це значення в пристрої D.

Примітка:

1. Значення налаштування в операнді n має бути меншим за 8. Якщо встановлене вами значення перевищує максимальне значення 8, воно розглядатиметься як максимальне значення 8. Наприклад, якщо $n = k6$, під час перетворення значення з плаваючою комою F123.45678 у рядкове значення, результат 123,456.
2. Якщо вихідне значення більше за значення параметра в операнді n, буде перетворено лише значення перед десятковою комою. Наприклад, якщо значення з плаваючою комою дорівнює F123456,78, а n дорівнює k4, результат перетворення дорівнює 1234. Для довідки див. ілюстрацію нижче.



● **Коли $m = 42$:**

Операнд S1:

S1+0, S1+1: цільова координата осі X (відносно позиціонування)

S1+2, S1+3: цільова координата осі Y (відносно позиціонування)

S1+4, S1+5: зміщення центру (цілочисельний тип) або центрального кута (тип з плаваючою комою)

S1+6, S1+7: цільова опорна частота (1~200 000 Гц; у разі перевищення допустимого діапазону система замість цього використовує максимальне або мінімальне значення)

S1+8: Вибір функцій

(K0, K1, K2: зсув центрального режиму; зсув на 10 градусів, 5 градусів або 1 градус)

(K10, K11, K12: центральний кут; зсув на 10 градусів, 5 градусів або 1 градус)

Операнд D: D+0, D+1: обчислення тривалості дугової інтерполяції за годинниковою стрілкою відносного положення 2 осей; одиниця: мс

Операнд m: Код функції K42

Операнд n: Зарезервовано

● **Коли $m = 43$:**

Операнд S1:

S1+0, S1+1: цільова координата осі X (відносно позиціонування)

S1+2, S1+3: цільова координата осі Y (відносно позиціонування)

S1+4, S1+5: зміщення центру (цілочисельний тип) або центрального кута (тип з плаваючою комою)

S1+6, S1+7: цільова опорна частота (1~200 000 Гц; у разі перевищення допустимого діапазону система замість цього використовує максимальне або мінімальне значення)

S1+8: Вибір функцій

(K0, K1, K2: зсув центрального режиму; зсув на 10 градусів, 5 градусів або 1 градус)

(K10, K11, K12: центральний кут; зсув на 10 градусів, 5 градусів або 1 градус)

Операнд D: D+0, D+1: обчислення тривалості дугової інтерполяції 2-осьового відносного положення проти годинникової стрілки; одиниця: мс

Операнд m: Код функції K43

Операнд n: Зарезервовано

● **Коли $m = 44$:**

Операнд S1: початкова точка вхідних точок ПЛК (K0~K15 → X0.0~X0.15) Операнд D: час фільтрації; одиниця: мкс (0~20000)

Операнд m: код функції K44

Операнд n: загальна кількість вхідних точок для оновлення

● **$m = 46$:**

Операнд S1: S1+0, S1+1: кількість вихідних імпульсів (відносно положення)

S1+2, S1+3: частота імпульсного виходу (AS324MT Y0.0/Y0.2: 1~4 МГц; інші: 1~200 кГц) S1+4: пристрій імпульсного виходу (K0~K11 : Y0.0~ Y0) .11)

Операнд D: D+0, D+1: обчислити вихідний час імпульсу відносного положення при прискоренні та уповільненні; одиниця: мс

Операнд m: Код функції K46

Операнд n: Зарезервовано

Додаткові зауваження

1. Рекомендовано використовувати матричні змінні типу даних Word для S і D.

API	Інструкція			Операнд								опис					
122 8		ZSET	P	SBit, EBit								Налаштування зони					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
SBit		●	●	●				●		○							
EBit		●	●	●				●		○							

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
SBit	●												
EBit	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



SBit : Пусковий пристрій зони

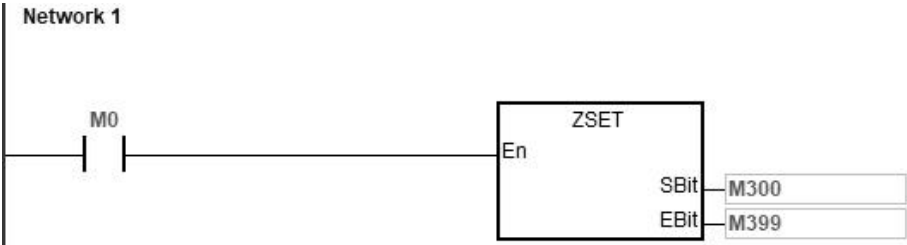
EBit : Кінцевий пристрій зони

Пояснення

1. Встановіть біт у зоні між SBit і EBit на ON. Тип пускового і кінцевого пристроїв повинен бути однаковим перед налаштуванням.
2. Якщо кількість SBit більша за кількість EBit, можна встановити лише EBit. Встановіть число SBit на менше.
3. За винятком пристроїв T, C і HC, ця інструкція доступна для пристроїв бітового типу та даних типу BOOL.

приклад

1. Коли M0 увімкнено, допоміжні реле M300~M399 увімкнені.



API	Код інструкції			Операнд							функція						
1229	F	SORT	P	src ~ ref							Сортування даних у форматі з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
src								●									
group								●					○	○			
column								●					○	○			
ref								●					○	○			
dst								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
src									●				
group			●				●						
column			●				●						
ref			●				●						
dst									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

FSORT	
En	
src	dst
group	
column	
ref	
FSORTP	
En	
src	dst
group	
column	
ref	

src : Перший пристрій, де зберігаються вихідні дані

group : Кількість рядків даних

column : Кількість стовпців даних

dst : Перший пристрій, де зберігаються відсортовані дані

ref : Довідкове значення для сортування даних

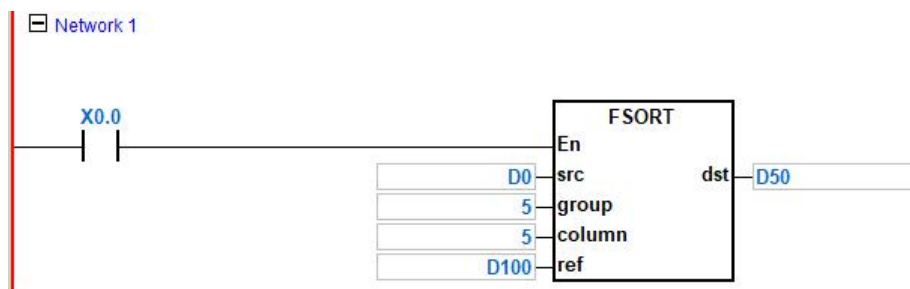
Пояснення

1. Ця інструкція зберігає дані для сортування в регістрах **group × стовпець × 2** , починаючи з регістра, визначеного **dst** . Якщо **src** і **dst** вказують той самий регістр, відсортовані дані будуть такими самими, як вихідні дані в регістрі, визначеному **src** .
2. Група операндів має бути між 1–32. **Стовпець** операнда має бути між 1–6. Операнд **ref** має бути між 1– стовпцем .

- 3. Коли SM604 вимкнено, інструкція сортує дані в порядку зростання. Коли SM604 увімкнено, інструкція сортує дані в порядку спадання.
- 4. Рекомендується використовувати інструкцію імпульсного типу, FSORTP, замість повторного сортування.

приклад

1. Припустимо, що SM604 вимкнено. Коли X0.0 перемикається з OFF на ON, інструкція сортує дані в порядку зростання.



2. Дані, які потрібно відсортувати, наведені в наступній таблиці.

		Кількість стовпців стовпця даних x 2				
		стовпці даних				
Колонка		1	2	3	4	5
рядок	студент номер	китайський	англійська	математика	Фізика	
	1	(D0,D1) 1,0	(D10,D11) 90,0	(D20,D21) 75,0	(D30, D31) 66,0	(D40, D41) 79,0
	2	(D2,D3) 2,0	(D12,D13) 55,0	(D22,D23) 65,0	(D32,D33) 54,0	(D42, D43) 63,0
	3	(D4,D4) 3,0	(D14,D15) 80,0	(D24,D25) 98,0	(D34,D35) 89,0	(D44, D45) 90,0
	4	(D6,D7) 4,0	(D16,D17) 70,0	(D26,D27) 60,0	(D36,D37) 99,0	(D46, D47) 50,0
	5	(D8, D9) 5,0	(D18,D19) 95,0	(D28, D29) 79,0	(D38, D39) 75,0	(D48, D49) 69,0

7. Якщо значення в D100 дорівнює 3, дані сортуються, як у наведеній нижче таблиці.

		Кількість стовпців стовпця даних x 2				
		стовпці даних				
Колонка рядок		1	2	3	4	5
		студент номер	китайський	англійська	математика	Фізика
Кількість рядків даних : group x 2	1	(D50,D51) 4,0	(D60, D61) 70,0	(D70, D71) 60,0	(D80, D81) 99,0	(D90, D91) 50,0
	2	(D52,D53) 2,0	(D62, D63) 55,0	(D72, D73) 65,0	(D82,D83) 54,0	(D92, D93) 63,0
	3	(D54,D55) 1,0	(D64, D65) 90,0	(D74,D75) 75,0	(D84, D85) 66,0	(D94, D95) 79,0
	4	(D56,D57) 5,0	(D66, D67) 95,0	(D76, D77) 79,0	(D86,D87) 75,0	(D96, D97) 69,0
	5	(D58,D59) 3,0	(D68, D69) 80,0	(D78, D79) 98,0	(D88, D89) 89,0	(D98, D99) 90,0

8. Якщо значення D100 дорівнює 5, дані виглядають як у наведеній нижче таблиці.

		Кількість стовпців стовпця даних x 2				
		стовпці даних				
Колонка рядок		1	2	3	4	5
		студент номер	китайський	англійська	математика	Фізика
Кількість рядків даних : group x 2	1	(D50,D51) 4,0	(D60, D61) 70,0	(D70, D71) 60,0	(D80, D81) 99,0	(D90, D91) 50,0
	2	(D52,D53) 2,0	(D62, D63) 55,0	(D72, D73) 65,0	(D82,D83) 54,0	(D92, D93) 63,0
	3	(D54, D55) 5,0	(D64, D65) 95,0	(D74, D75) 79,0	(D84,D85) 75,0	(D94, D95) 69,0
	4	(D56,D57) 1,0	(D66, D67) 90,0	(D76, D77) 75,0	(D86, D87) 66,0	(D96, D97) 79,0
	5	(D58,D59) 3,0	(D68, D69) 80,0	(D78, D79) 98,0	(D88, D89) 89,0	(D98, D99) 90,0

Додаткові зауваження

1. Якщо значення в пристрої перевищує діапазон, інструкція HE виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо значення в **групі** , **стовпці** або **посиланні** перевищує діапазон, інструкція HE виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
3. Якщо значення в **src** перевищує діапазон значення з плаваючою комою, інструкція HE виконується, SM0 увімкнено, а код помилки в SR0 – 16#2013.

16.14 Інструкція зі створення конструкції

16.14.1 Перелік інструкцій зі створення конструкції

У наведеній нижче таблиці наведено інструкції з обробки даних, описані в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32- розрядний		
1300	FOR	–	–	Запуск вкладеного циклу
1301	NEXT	–	–	Завершення вкладеного циклу
1302	ПЕРЕРВА	–	–	Завершення циклу FOR-NEXT

6.14.2 Пояснення інструкцій зі створення структури

API	Код інструкції				Операнд								функція				
1300		FOR			S								Запуск вкладки циклу				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●			●	O	O	O			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S		●				●											
								Інструкція пульсу				16-бітна інструкція				32-розрядна інструкція	
								-				AS				-	

СИМВОЛ



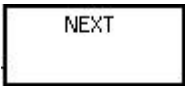
S : Кількість разів виконання циклу

Пояснення

Зверніться до інструкції NEXT (API 1301), щоб дізнатися більше.

API	Код інструкції			Операнд	функція
1301		NEXT		-	Завершення вкладки циклу
				Інструкція пульсу	16-бітна інструкція
				-	AS
					32-розрядна інструкція
					-

СИМВОЛ

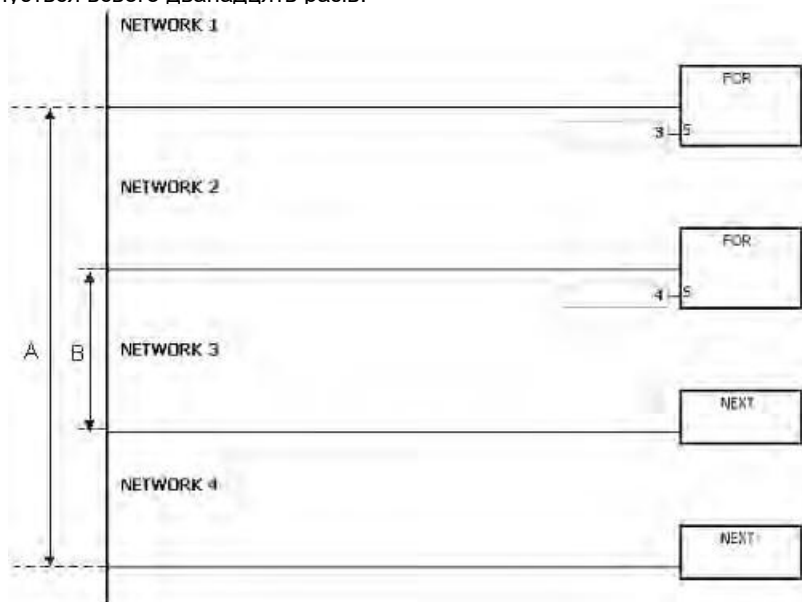


Пояснення

1. Ця інструкція виконує програму між інструкціями FOR і NEXT N разів, де N — це значення в **S** , указане для інструкції FOR (API 1300). Після виконання програми між інструкціями FOR і NEXT N разів виконується програма, що слідує за інструкцією NEXT. Інструкція FOR визначає кількість виконання програми між інструкціями FOR і NEXT.
2. N має бути в діапазоні від 1 до 32 767. Якщо N менше 1, інструкція обробляє його як 1.
3. Якщо ви не бажаєте виконувати програму між інструкціями FOR і NEXT, ви можете пропустити її за допомогою інструкції CJ (API 0400).
4. Наступні умови призводять до помилок.
 - Інструкція NEXT передує інструкції FOR.
 - Інструкція FOR існує, але інструкція NEXT не існує.
 - Інструкція NEXT слідує за інструкцією FEND або END.
 - Кількість разів використання інструкції FOR у програмі відрізняється від кількості разів використання інструкції NEXT у програмі.
5. Інструкції FOR і NEXT підтримують структуру вкладеної програми. Може бути не більше 32 рівнів вкладених програмних структур. Якщо цикл виконується багато разів, ПЛК потребує більше часу для сканування програми, і може виникнути помилка сторожового таймера. Ви можете скористатися інструкцією WDT (API 1900), щоб вирішити проблему.

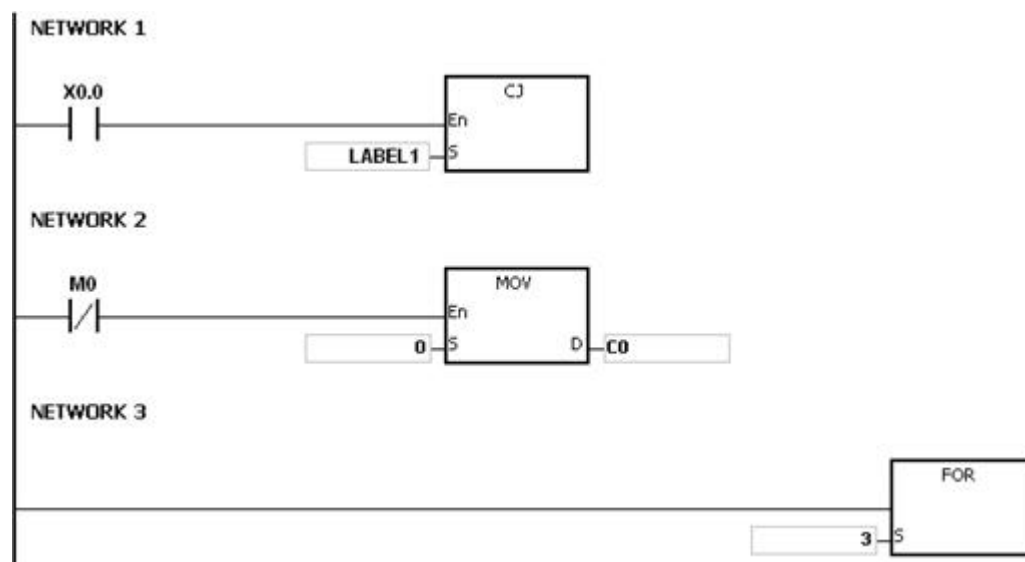
Приклад 1

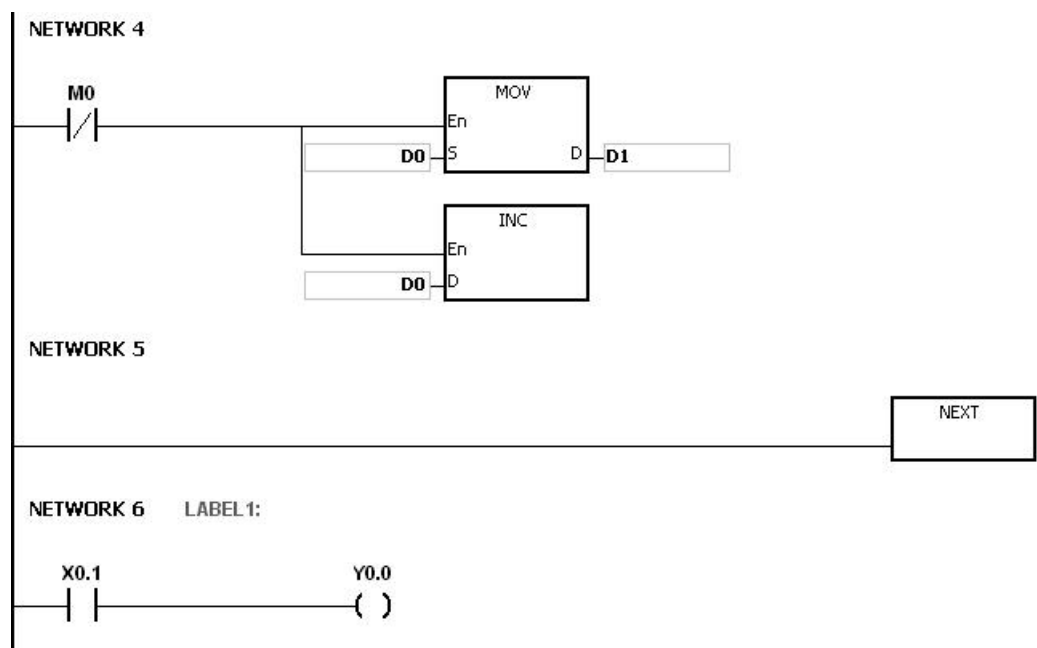
Після того, як програма А виконується три рази, виконується програма, яка слідує за інструкцією NEXT. Програма В виконується чотири рази кожного разу, коли виконується програма А. Таким чином, програма В виконується всього дванадцять разів.



Приклад 2

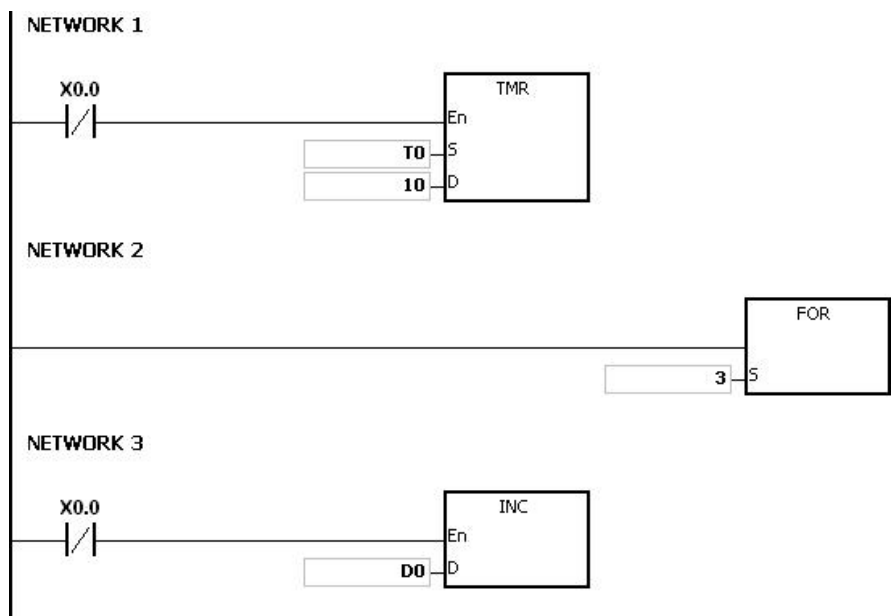
Коли X0.0 вимкнено, виконується програма між FOR і NEXT. Коли X0.0 увімкнено, виконується інструкція CJ. Виконання програми переходить до LABEL 1:, тобто мережа 6, а мережа 4 – 5 не виконується.

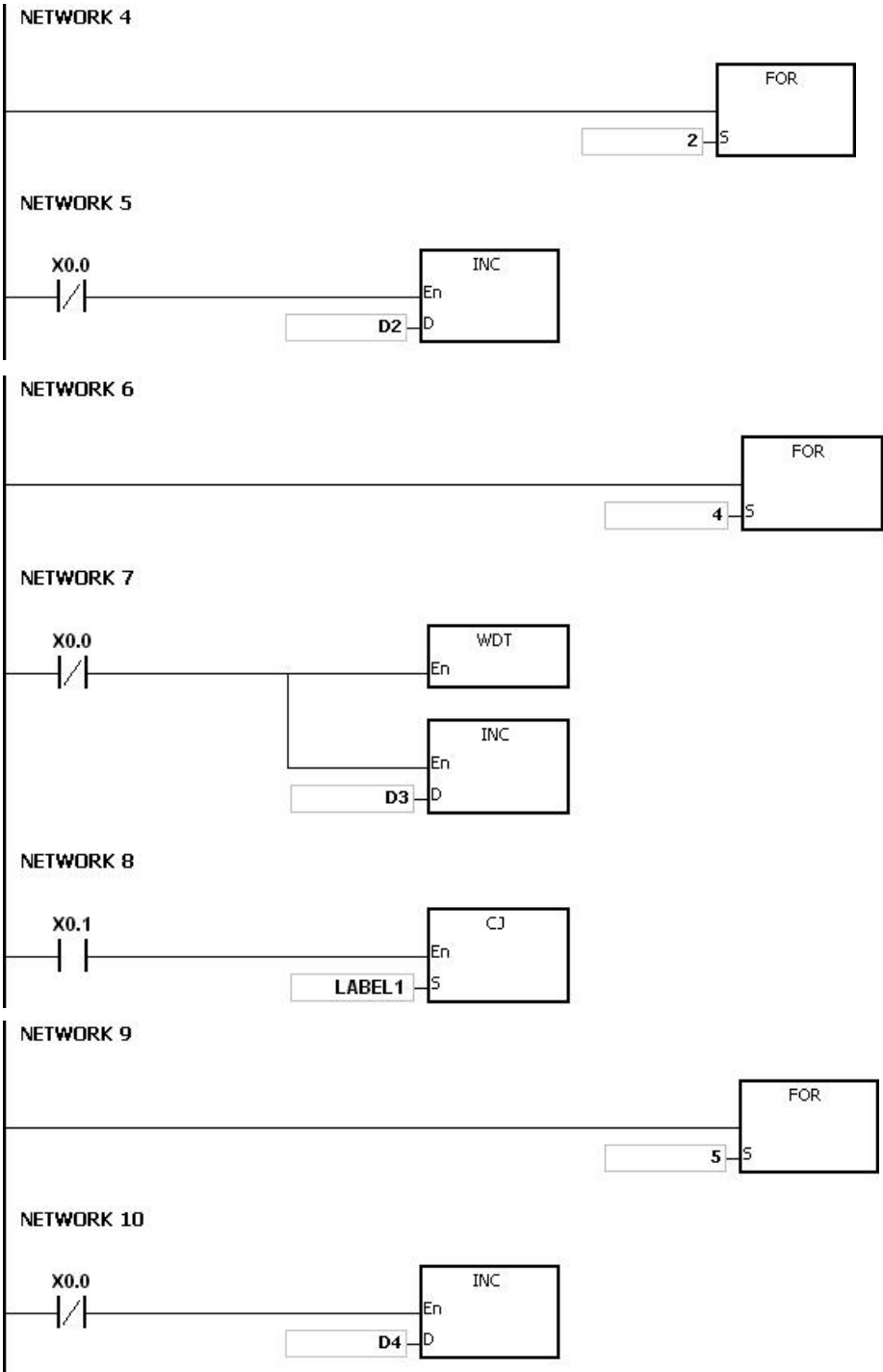


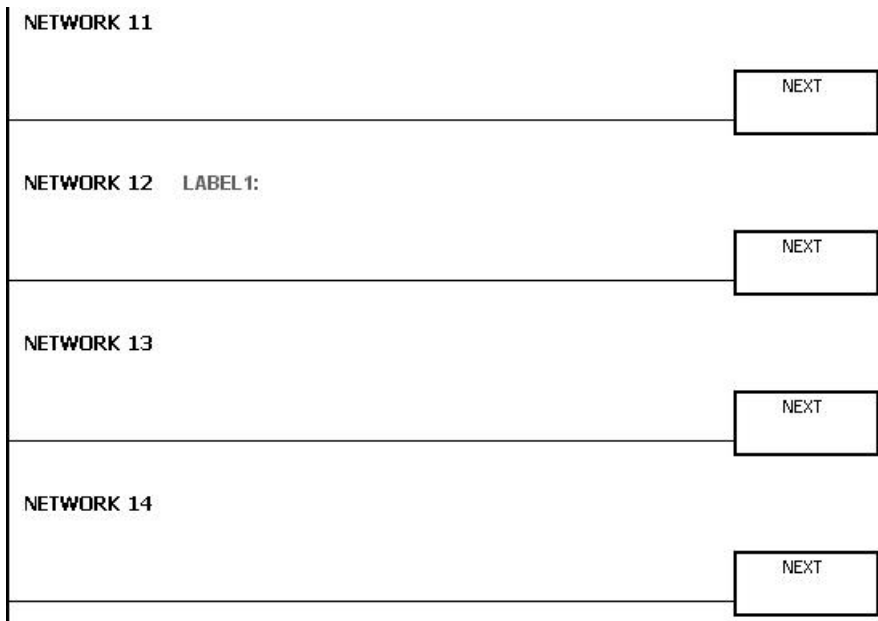


Приклад 3

Якщо програма між FOR і NEXT не повинна виконуватися, ви можете пропустити її за допомогою інструкції CJ. Коли X0.1 у мережі 8 увімкнено, виконується інструкція CJ. Виконання програми переходить до LABEL 1:, тобто мережа 12, а мережа 9 – 11 не виконується.







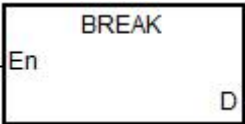
Додаткові зауваження

Додаткову інформацію про використання міток див. у посібнику користувача ISPSOft.

API	Код інструкції				Операнд								функція				
1302		ПЕРЕРВ A			D								Завершення циклу FOR-NEXT				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D		●			●	●		●			O	O					
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
D		●				●											

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



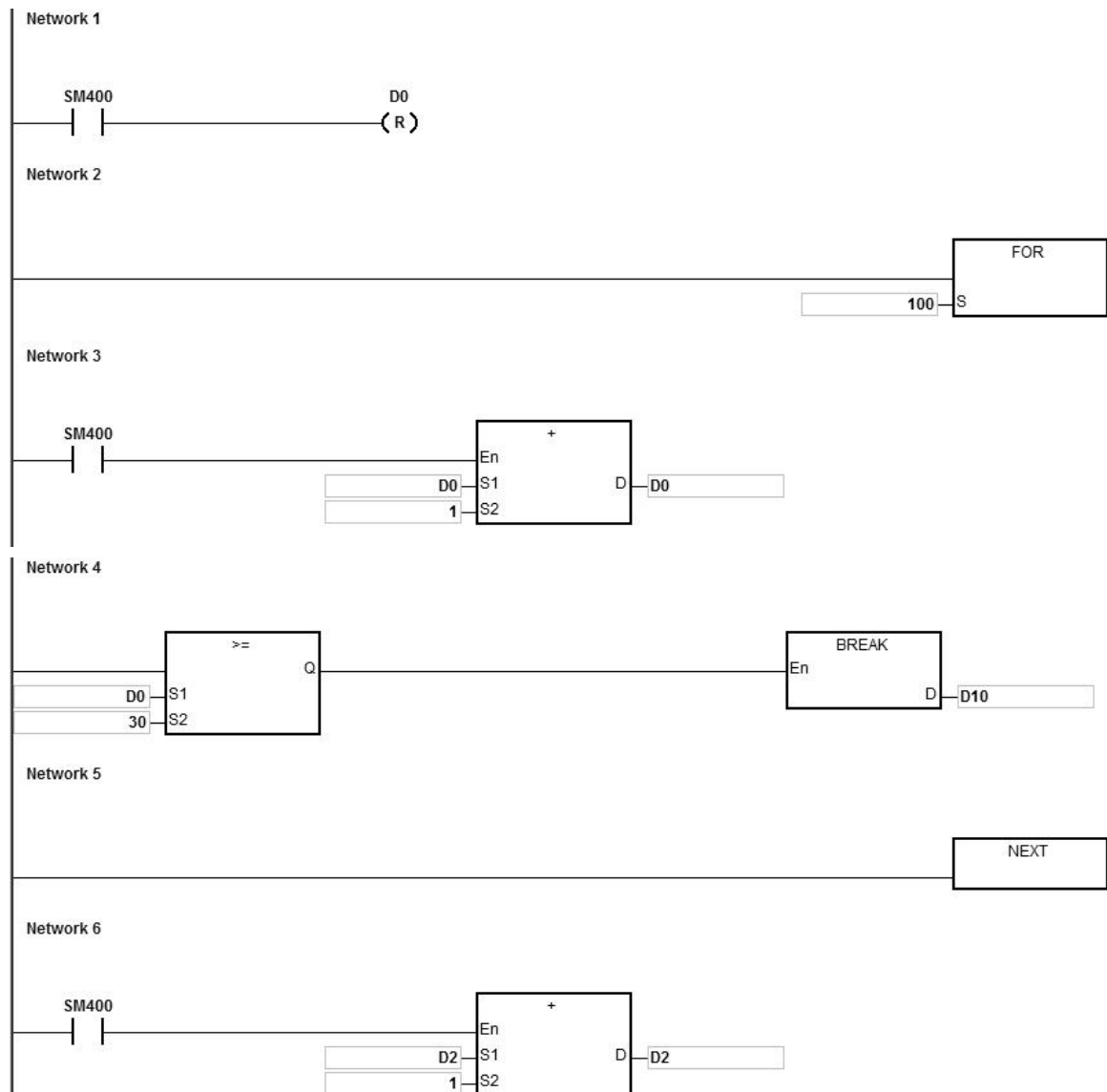
D : Пристрій, де залишилося число
кількість разів, коли цикл може
бути виконано, зберігається

Пояснення

- Ця інструкція завершує цикл FOR/NEXT. Залишок кількості повторів циклу FOR/NEXT зберігається в D .
Після виконання циклу D разів програма переходить до інструкції NEXT і виконує інструкцію після інструкції NEXT.
- Коли інструкція виконується, кількість повторів циклу FOR/NEXT, що залишилася, зберігається в D , включаючи цей раз виконується інструкція BREAK.
- Коли інструкція BREAK виконується вперше для завершення циклу FOR/NEXT, програма не вийти з циклу FOR/NEXT, щоб виконати наступну інструкцію. Якщо інструкція BREAK виконується більше одного разу для завершення циклу FOR/NEXT, програма переходить до інструкції NEXT і виконує інструкцію, наступну за інструкцією NEXT.

приклад

Коли виконується цикл FOR/NEXT, 1 додається до значення в D0. Коли значення в D0 дорівнює 30, цикл FOR/NEXT припиняється, а кількість разів, що залишилася для повторення циклу FOR/NEXT, тобто 71, зберігається в D10. Виконання програми переходить до LABEL 1:, тобто до мережі 6, а 1 додається до значення в D2.



Додаткові зауваження

1. Якщо інструкція BREAK знаходиться за межами циклу FOR/NEXT, це спричиняє помилку операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2017.
2. Додаткову інформацію про використання міток див. у посібнику користувача ISPSOft.

6.15 Інструкції модуля

6.15.1 Список інструкцій модуля

У наведеній нижче таблиці наведено інструкції для модуля, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1400	FROM	DFROM	✓	Читання даних з керуючого регістру в модулі розширення
1401	TO	DTO	✓	Запис даних у керуючий регістр у модулі розширення
1402	PUCONF	DPUCONF	✓	Налаштування вихідних параметрів керування модулем ПУ
1403	PUSTAT	–	–	Читання вихідного стану модуля ПУ
1404	–	DPUPLS	–	Вихідний імпульс модуля ПУ (без прискорення)
1405	–	DPUDRI	–	Вихід відносного положення модуля ПУ (з прискоренням і уповільненням)
1406	–	DPUDRA	–	Вихід абсолютної адресації модуля ПУ (з прискоренням і уповільненням)
1407	–	DPUZRN	–	Наведення модуля ПУ
1408	–	DPUJOG	–	Руховий вихід модуля ПУ
1409	–	DPUMPG	–	Вихід MPG модуля ПУ
1410	–	DPUCNT	–	Функція високошвидкісного лічильника модуля ПУ
1415	–	DLCCAL	–	Калібрування каналу модуля LC
1416	–	DLCWEI	–	Зчитування значення ваги через модуль LC
1417	–	DMPID	–	Алгоритм PID для модуля RTD/TC
1418	–	DHCCNT	–	Підраховано значення лічильника модуля HC
1419	–	DHCCAP	–	Зафіксовано значення лічильника модуля HC
1420	HCDO	–	–	Цифровий вихід модуля HC
1421	–	DHCCMP	–	Значення лічильника модуля HC у порівнянні для виведення
1422	–	DHCCMPT	–	Значення лічильника модуля HC у груповому порівнянні для виведення
1423	–	DHCMEAS	–	Модуль HC вимірювання частоти та обертів
1424	ADLOG	DADLOG	–	Запис даних модуля аналогового введення
1425	ADPEAK	DADPEAK	–	Запис пікових даних модуля аналогового введення
1426	IOLINKR	–	–	Зчитайте параметр із пристрою O-Link
1427	IOLINKW	–	–	Запишіть параметр у пристрій IO-Link
1430	MLPF	–	–	Цифровий фільтр низьких частот для аналогових вхідних сигналів

6.15.2 Пояснення інструкцій модуля

API	Код інструкції			Операнд								функція					
1400	D	FROM	P	$m_1, m_2, m_3, D_1, D_2, n$								Читання даних з керуючого регістру в модулі розширення					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
m_1					●	●	●	●	●		○	○	○	○			
m_2					●	●	●	●	●		○	○	○	○			
m_3					●	●	●	●	●		○	○	○	○			
D_1					●	●	●	●									
D_2					●	●	●	●									
n					●	●	●	●	●		○	○	○	○			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
m_1		●	●			●	●						
m_2		●	●			●	●						
m_3		●	●			●	●						
D_1		●	●			●	●						
D_2		●	●			●	●						
n		●	●			●	●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

FROM		FROMP	
En		En	
m1	D1	m1	D1
m2	D2	m2	D2
m3		m3	
n		n	

DFROM		DFROMP	
En		En	
m1	D1	m1	D1
m2	D2	m2	D2
m3		m3	
n		n	

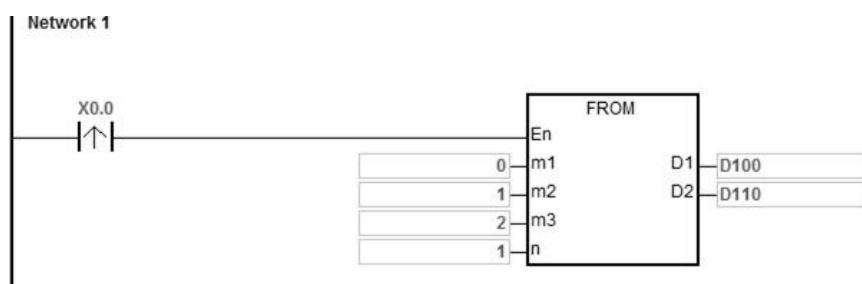
- m_1 : Номер модуля ЦП або пульта
номер модуля розширення
- m_2 : Номери замовлення внутрішнього номера
- m_3 : Номер контрольного реєстру
- D_1 : Пристрій, де зберігаються дані
- D_2 : Пристрій, де зберігається код помилки
- n : Довжина даних

Пояснення

1. Ця інструкція зчитує дані з керуючого регістру в модулі розширення. Усі регістри перераховані та детально описані в посібнику з модулів серії AS.
2. Значення в m_1 має бути в діапазоні від 0 до 16. Нуль означає модуль ЦП, а 1 – 16 – модулі розширення.
3. Операнд m_2 представляє кількість правих модулів розширення, які підключені до модуля CPU або до віддалених модулів. Перший пристрій номер 1, другий пристрій номер 2 і так далі. Враховуються будь-які типи підключених модулів і можна підключити до 32 пристроїв.
4. Операнд m_3 задає номер керуючого регістру.
5. Інструкція FROM встановлює для D_2 значення 0. Коли виникає помилка, інструкція не встановлює для D_2 значення 0. Див. Додаткові зауваження нижче для отримання додаткової інформації про коди помилок. Коли інструкція не виконується, D_2 не містить коду помилки.
6. Операнд n має бути в діапазоні від 1 до 8.
7. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

приклад

Коли X0.0 перемикається з ВІМК. на УВІМК., інструкція зчитує дані, що зберігаються в CR#2, з правого боку першого і зберігає дані в D100. Якщо помилка не виникає, код у D110 – 16#0000.



Додаткові зауваження

1. Якщо значення в m_1 і m_2 перевищують свій діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо D_1 до $D_1 + n - 1$ перевищують діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Якщо значення в n перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

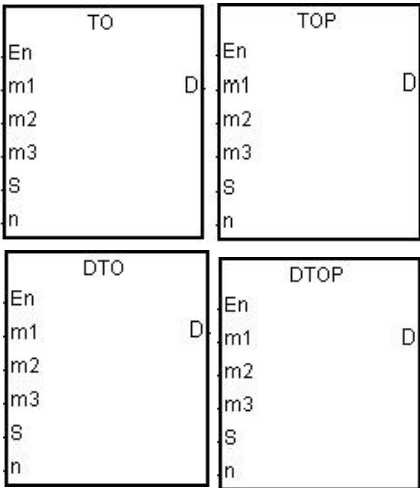
4. Через те, що інструкція FROM знижує ефективність виконання як модуля ЦП, так і модуля введення/виведення, пропонується використовувати інструкцію імпульсного типу для виконання одного тригера, як у наведеному вище прикладі.

5. Якщо є будь-яка відповідь на помилку від модулів, інструкція зберігає код помилки в D_2 . Описи кодів помилок наведено в наступній таблиці.

Код помилки	опис
16#1400	Спроба прочитати дані з керуючого регістру (CR) у модулі, але такого CR немає номер існує.
16#1401	Значення недійсне для модуля.
16#1402	Модуль не відповідає, тайм-аут зв'язку.

API		Код інструкції			Операнд						функція						
1401		D	TO	P	m ₁ , m ₂ , m ₃ , S, D, n						Запис даних у керуючий регістр у модулі розширення						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
m ₁					●	●	●	●	●		○	○	○	○			
m ₂					●	●	●	●	●		○	○	○	○			
m ₃					●	●	●	●	●		○	○	○	○			
S					●	●	●	●	●				○	○			
D					●	●	●	●									
n					●	●	●	●	●		○	○	○	○			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
m ₁		●	●			●	●										
m ₂		●	●			●	●										
m ₃		●	●			●	●										
S		●	●			●	●										
D		●	●			●	●										
n		●	●			●	●										
					Інструкція пульсу		16-бітна інструкція				32-розрядна інструкція						
					AS		AS				AS						

СИМВОЛ



- m₁

:

Номер модуля ЦП або номер модуля віддаленого розширення
- m₂

:

Номери замовлення внутрішнього номера
- m₃

:

Номер контрольного реєстру
- S

:

Пристрій, де зберігаються дані
- D

:

Пристрій, де зберігається код помилки
- n

:

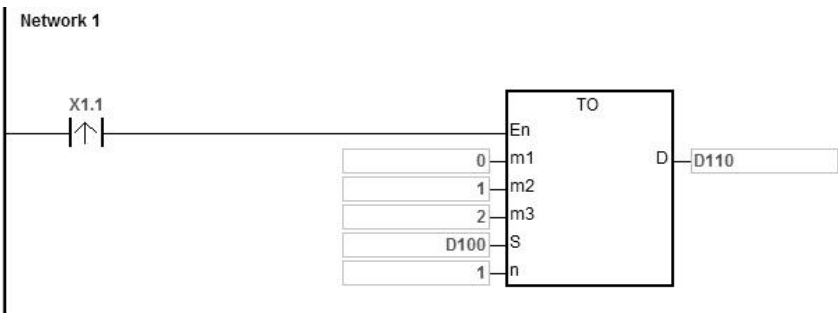
Довжина даних для запису

Пояснення

1. Ця інструкція записує дані в регістр керування в модулі розширення. Усі регістри перераховані та детально описані в посібнику з модулів серії AS.
2. Значення в m_1 має бути в межах 0–16. Нуль означає модуль центрального процесора, а 1–16 — модулі розширення.
3. Операнд m_2 представляє кількість правих модулів розширення, які підключені до модуля CPU або до віддалених модулів. Перший пристрій номер 1, другий пристрій номер 2 і так далі. Враховуються будь-які типи підключених модулів і можна підключити до 32 пристроїв.
4. Операнд m_3 задає номер керуючого регістру.
5. Інструкція TO встановлює **D** на 0. Коли виникає помилка, інструкція не встановлює **D** на 0. Зверніться до Додаткових зауважень нижче, щоб отримати додаткові відомості про коди помилок. Коли інструкція не виконується, **D** не містить коду помилки.
6. Операнд m_4 має бути між 1–8.
7. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.
8. Коли **S** є шістнадцятковим значенням, інструкція передає **n** шістнадцяткових значень до модуля введення/виведення. Припустимо, що **S** дорівнює 16#0001, а **n** дорівнює 3. Інструкція передає три 16#0001 на модуль введення/виведення.

приклад

Коли X0.0 перемикається з OFF на ON, інструкція TO записує дані, що зберігаються в D100, до CR#2 у правій частині першого модуля. Якщо помилки не виникає, код у D110 – 16#0000.



Додаткові зауваження

1. Якщо значення в m_1 і m_2 перевищують свій діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо $D_1 - D_1 + n - 1$ перевищують діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо значення в n перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
4. Через те, що інструкція TO знижує ефективність виконання як модуля ЦП, так і модуля введення/виведення, пропонується використовувати інструкцію імпульсного типу для виконання одного тригера, як у наведеному вище прикладі.
5. Якщо є будь-яка відповідь на помилку від модулів, інструкція зберігає код помилки в D_2 . Описи кодів помилок наведено в наступній таблиці.

Код помилки	опис
16#1400	Спроба прочитати дані з керуючого регістру (CR) у модулі, але такого CR немає номер існує.
16#1401	Значення недійсне для модуля.
16#1402	Модуль не відповідає, тайм-аут зв'язку.

API		Інструкція			Операнд						опис					
1402		D	PUCONF	P	Module ~ Error, ErrCode						Налаштування вихідних параметрів керування модулем ПУ					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Module								●					○	○		
Axis								●					○	○		
Mode								●					○	○		
SSpeed								●					○	○		
Atime								●					○	○		
Dtime								●					○	○		
MSpeed								●					○	○		
Z_no								●					○	○		
Offset								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
Mode		●				●							
SSpeed		●				●							
Atime		●				●							
Dtime		●				●							
MSpeed			●				●						
Z_no		●				●							
Offset		●				●							
Done	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	—	AS

СИМВОЛ

DPUCONF		DPUCONF	
En		En	
Module	Done	Module	Done
Axis	Error	Axis	Error
Mode	ErrCode	Mode	ErrCode
SSpeed		SSpeed	
Atime		Atime	
Dtime		Dtime	
MSpeed		MSpeed	
Z_no		Z_no	
Offset		Offset	

Module	: Номер модуля
Axis	: Номер вихідної осі
Mode	: Режим виведення
SSpeed	: Швидкість початкової/кінцевої частоти
Atime	: Час розгону
Dtime	: Час уповільнення
MSpeed	: Максимальна вихідна частота
Z_no	: Кількість фаз Z, які слід шукати після повернення у вихідне положення
Offset	: Вкажіть кількість виходів після повернення у вихідне положення
Done	: Прапор завершення
Error	: Прапор помилки
ErrCode	: Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із FW V1.08.00 або новішої версії. Для ISPSOFT ми рекомендуємо використовувати версію програмного забезпечення 3.12 і вище. Час для встановлення цієї інструкції – коли налаштування En змінюється з ВИМК. на УВИМК.

2. **Module** встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий

номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість — 32. Ця

інструкція стосується виключно модулів PU праворуч від ПЛК і не застосовується до модулів PU на

праворуч від віддаленого модуля. Якщо вказаний модуль не є модулем PU, прапор помилки **Error** зміниться на ON.

3. **Axis** встановлює номер вихідної осі для вказаного модуля PU. Значення параметрів 1~4 представляють вісь 1~4

вихід зазначеного модуля PU відповідно. Якщо модуль PU не має відповідного номера осі для виведення, прапор помилки **Error** зміниться на ON.

Дивіться наступну комбінацію номерів осей і відповідних вихідних точок модулів PU.

Назва модуля PU	Вісь 1	Вісь 2	Вісь 3	Вісь 4
AS02PU	Y0.0 / Y0.1	Y0,2 / Y0,3	NA	NA
AS04PU	Y0.0 / Y0.1	Y0,2 / Y0,3	Y0,4 / Y0,5	Y0,6 / Y0,7

4. **Mode** встановлює режим виведення вихідної осі, а значення налаштувань пояснюються в наступній таблиці.

Режим виведення значення	опис	Зауваження
0	Одноточковий вихідний імпульс (парне число точка лише для виведення)	Наприклад, Y0.0 або Y0.2 для виведення
1	Пульс (парна точка) + напрям	Наприклад, Y0.0 для пульсу, а Y0.1 для

	(Непарна точка)	напрямок. Y0.1: ON, негативний напрямок; Y0.1: ВІМК., позитивний напрямок
2	CW (точка парного числа) + CCW (Ан непарна точка)	Наприклад, Y0.0 для CW (позитивного напрямку) і Y0.1 для CCW (негативний напрямок)
3	Фаза А (точка з парним числом) + Фаза В (точка з непарним числом)	Наприклад, Y0.0 для фази А, а Y0.1 для фази В. Коли фаза А є провідною фаза В: позитивний напрямок; коли фаза В є провідною, фаза А: негативний напрямок
Інше значення	Автоматично перейти в режим 1 (значення за замовчуванням)	

5. SSpeed ~ Offset

Див. пояснення наступних незафіксованих параметрів і значень налаштувань. Якщо значення налаштування перевищують діапазону, інструкція буде автоматично виконана при мінімальному або максимальному значенні.

Параметр	функція	Діапазон	За замовчуванням	Зауваження
SSpeed	Початкова/кінцева частота	0 ~ 10 000 (Одиниця: Гц)	100	
Atime	Час розгону	0 ~ 10 000 (Одиниця вимірювання: мс)	100	
Dtime	Час уповільнення	0 ~ 10 000 (Одиниця вимірювання: мс)	100	
MSpeed	Максимальна вихідна частота	AS02PU: 100 ~ 200 000 (Одиниця: Гц)	200 тис	32-розрядне значення
		AS04PU: 100 ~ 100 000 (Одиниця: Гц)	100 тис	
Z_no	Кількість сигналів фази Z для пошуку після повернення до початку.	-100 ~ 100 (Одиниця виміру: раз)	0	0: вимкнено
Offset	Виводить позицію зсуву після наведення завершено, і пошук фази Z виконано.	-10 000 ~ 10 000 (Одиниця вимірювання: імпульси)	0	0: вимкнено

6. Done , вихід вказаного модуля PU встановлено як позначку завершення. Коли **Done** включено, це вказує

що налаштування параметра виконано успішно. Ви можете продовжувати виконувати виведення позиціонування на основі стану прапор завершення (ON). Очищення прапорця **Done** потрібно проводити вручну. Прапор «**Done**» змінюється на УВІМК. лише після завершення налаштування.

7. Error , вихід вказаного модуля PU є прапором помилки параметра. Більшість діапазонів параметрів автоматично фільтруються ПЛК. Таким чином, якщо прапорець помилки увімкнено, це означає, що не вказано модуль PU, або номер модуля PU неправильний, або номер вихідної осі неправильний.

8. Інструкція - це імпульсна інструкція. Навіть якщо контакт А прийнято як контакт умови, параметри модуля PU також встановлюються лише під час запуску інструкції. Таким чином, якщо значення параметра потрібно оновити, перезапустіть інструкцію, щоб знову встановити параметр.

9. Оскільки встановлені параметри доставляються через команду зв'язку модуля, підтвердьте стан виведення **«Done»** або **«Error»** , перш ніж значення параметра буде змінено, а потім продовжуйте відповідні операції.

10. **ErrCode** показує коди помилок. Опис див. нижче.

Код помилки	опис
16#1400	Модуль не підтримує функцію.
16#1401	Дані, що зберігаються в модулі, є незаконними або перевищують допустимий діапазон.
16#1402	Немає відповіді від модуля; відбувається тайм-аут зв'язку.

Приклад програмування: зверніться до опису інструкції DPUDRI (API 1405) для отримання додаткової інформації.

API	Інструкція				Операнд							опис					
1403		PUSTAT			Module ~ ErrCode							Читання вихідного стану модуля PU					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Module								●					○	○			
Axis								●					○	○			
ZeroS	●	●	●	●				●									
C_Posi								●									
Execute		●	●	●				●									
Pause		●	●	●				●									
Error		●	●	●				●									
ErrCode								●									

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
ZeroS	●												
C_Posi			●				●						
Execute	●												
Pause	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ

PUSTAT	
En	
Module	C_Posi
Axis	Execute
ZeroS	Pause
	Error
	ErrCode

- Module** : Номер модуля
- Axis** : Номер вихідної осі
- ZeroS** : Очистити поточну вихідну позицію до 0
- C_Posi** : Поточна вихідна позиція
- Execute** : Прапор виконання
- Pause** : Прапор паузи
- Error** : Прапор помилки
- ErrCode** : Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із FW V1.08.00 або новішої версії. Для ISPSoft ми рекомендуємо використовувати версію програмного забезпечення 3.12 і вище. Щоб постійно оновлювати статус вибраних осей, для параметра En потрібно встановити значення ON.

2. Module встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість — 32. Ця інструкція стосується виключно модулів PU праворуч від ПЛК і не застосовується до модулів PU праворуч від віддаленого модуля. Якщо вказаний модуль не є модулем PU, прапор помилки **Error** зміниться на ON.

3. Axis встановлює номер вихідної осі для вказаного модуля PU. Значення налаштування 1~4 представляють вихідні дані axis1~axis4 зазначеного модуля PU відповідно. Якщо модуль PU не має відповідного номера осі для виведення, прапор помилки **Error** зміниться на ON.

4. C_Posi встановлює поточне положення вихідної осі для вказаного модуля PU. Значення параметра є зафіксованим значенням і зберігається в модулі PU. Якщо значення потрібно скинути до 0, встановіть **ZeroS** з OFF на ON під час запуску інструкції.

5. Execute — це прапор лише для читання, який означає, що вихідна вісь зазначеного модуля PU виводить чи ні. Якщо **для Execute** встановлено значення On, це означає, що вивід виконується. Якщо **Execute** вимкнено, це означає, що вихідна вісь не використовується та може прийняти наступну команду виведення.

6. Pause — це прапор лише для читання для керування вихідною віссю зазначеного модуля PU, щоб призупинити його вихід. Коли **пауза** включена, це означає, що вихідний сигнал призупинено, поточна швидкість дорівнює 0 і поточний вихідний сигнал не досяг заданого цільового вихідного положення. Якщо ви відновите вихід, прапорець буде очищено автоматично.

6

Примітка: якщо **пауза** увімкнена, **функція «Execute»** також постійно увімкнена.

7. Error — це позначка помилки лише для читання, яка означає, що під час читання зазначеного модуля PU виникає помилка.

Зверніться до пояснення кодів помилок в **ErrCode**.

8. Після того, як інструкція PUSTAT дає команду паузи, прапорці **Execute**, **Pause** та **Error** стають прапорцями лише для читання, і на даний момент їх стани не можуть бути змінені. Прапори «**Execute**», «**Pause**» та «**Error**» можна встановлювати або скидати лише тоді, коли інструкцію PUSTAT вимкнено.

9. ErrCode показує коди помилок, а пояснення наведено в наступній таблиці.

Код помилки	опис
16#1400	Модуль не підтримує функцію.
16#1401	Дані, що зберігаються в модулі, є незаконними або перевищують допустимий діапазон.
16#1402	Немає відповіді від модуля; відбувається тайм-аут зв'язку.
16#1403	Такого номера вихідної осі в модулі ПУ немає.
16#1404	Вихідна частота модуля PU є недопустимою.
16#1405	Вивідна вісь, визначена модулем PU, виводить дані. Не дозволяється вказувати вихід повторно.

Приклад програмування: зверніться до опису інструкції DPUDRI (API 1405) для отримання додаткової інформації.

API	Інструкція			Операнд				опис							
1404	D	PUPLS		Module ~ ErrCode				Вихідний імпульс модуля PU (без прискорення)							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Module								●					○	○		
Axis								●					○	○		
TarPulse								●					○	○		
TarSpeed								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
TarPulse			●				●						
TarSpeed			●				●						
Done	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DPUPLS	
En	
Module	Done
Axis	Error
TarPulse	ErrCode
TarSpeed	

Module	:	Номер модуля
Axis	:	Номер вихідної осі
TarPulse	:	Цільова кількість вихідних імпульсів
TarSpeed	:	Цільова вихідна частота
Done	:	Прапор завершення
Error	:	Прапор помилки
ErrCode	:	Код помилки

Пояснення

- Ця інструкція доступна для ПЛК із FW V1.08.00 або новішої версії. Для ISPSOFT ми рекомендуємо використовувати версію програмного забезпечення 3.12 і вище. Якщо для параметра En встановлено значення ON, ця інструкція буде ефективною. Як тільки налаштування змінюється на ВИМК., вихідний імпульс буде негайно припинено.
- Module** встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість – 32.

Ця інструкція стосується виключно модулів PU праворуч від ПЛК і не застосовується до модулів PU праворуч від віддаленого модуля. Якщо вказаний модуль не є модулем PU, прапор помилки **Error** зміниться на ON.

3. Axis встановлює номер вихідної осі для вказаного модуля PU. Значення налаштування 1~4 представляють вихідні дані axis1~axis4 зазначеного модуля PU відповідно. Якщо модуль PU не має відповідного номера осі для виведення, прапор помилки **Error** зміниться на ON.

4. TarPulse встановлює кількість вихідних імпульсів. Номер імпульсу є 32-розрядним значенням із позитивним знаком. Коли значення дорівнює 0, це означає, що виведення виконується завжди, кількість вихідних імпульсів не обмежена і виведення не зупиняється, доки інструкцію не буде вимкнено. Коли значення менше 0, ПЛК автоматично використовує доповнення 2 с, щоб перетворити значення в позитивне ціле число як кількість вихідних імпульсів.

5. TarSpeed встановлює цільову вихідну швидкість (одиниця: Гц). Вхідне значення є 32-бітовим знаком. Ви можете змінити цільову частоту в будь-який час після ввімкнення інструкції, і модуль PU автоматично переключиться на щойно встановлену цільову частоту після виведення повного імпульсу.

Примітка. Перш ніж змінити цільову частоту, зверніть увагу на те, чи збігаються змінена швидкість і час сканування ПЛК.

Назва модуля	Діапазон налаштувань TarSpeed
AS02PU	-200 000 (-200K) ~ 200 000 (200K)
AS04PU	-100 000 (-100K) ~ 100 000 (100K)

6. Коли **TarSpeed** є додатним числом (>0), це означає, що вихідна точка «позитивного напрямку» вимкнена. Коли **TarSpeed** є від'ємним числом (<0), це означає, що вихідна точка «негативного напрямку» увімкнена. Коли **TarSpeed** дорівнює 0, це означає, що виведення буде призупинено після повного виведення імпульсу, що виконується.

6

7. Інструкція не підтримує функцію прискорення і уповільнення. Замість цього використовуйте інструкцію DPUDRI, якщо вам потрібна функція прискорення та уповільнення.

8. Інструкцію можна використовувати для зміни швидкості. Під час виконання інструкції ви можете змінити значення **TarSpeed**, щоб змінити вихідну швидкість.

9. Коли виходи досягають кількості імпульсів, визначеної **TarPulse**, прапор **Done** змінюється на ON. Done _ прапор потрібно очистити вручну. Інструкція встановлює прапор завершення на ON лише тоді, коли вивід завершено.

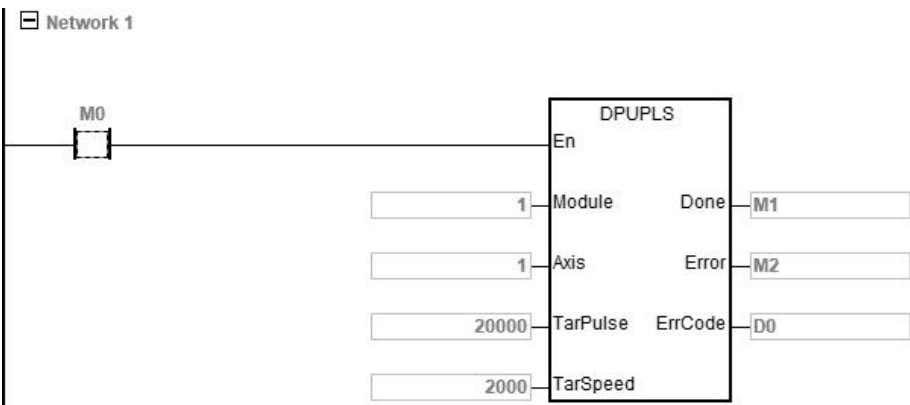
10. Якщо під час виведення інструкції виникає будь-яка помилка, прапорець **Error** змінюється на ON. Зверніться до кодів помилок **ErrCode** для усунення несправностей.

Коди помилок, які показує **ErrCode**, перераховані в наведеній нижче таблиці.

Код помилки	опис
16#1400	Модуль не підтримує функцію.
16#1401	Значення, збережене в модулі, є недопустимим або перевищує допустимий діапазон.
16#1402	Немає відповіді від модуля; відбувається тайм-аут зв'язку.
16#1403	Такого номера вихідної осі в модулі ПУ немає.
16#1404	Вихідна частота модуля PU є недопустимою.
16#1405	Вивідна вісь, визначена модулем PU, виводить дані. Не дозволяється вказувати виводити повторно.
16#1406	Модуль PU припиняє вихідний імпульс, коли досягається позитивна межа.
16#1407	Модуль PU припиняє вихідний імпульс при досягненні негативної межі.

Приклад програмування

1. Встановіть вихідну точку AS02PU як «Pulse Y0.0+ direction Y0.1» у HWCONFIG. Усі параметри осі 1 встановлені за замовчуванням.
2. Коли M0=ON, виконайте DPUPLS на осі 1, виводячи 20 000 імпульсів від Y0.0 на частоті 2 кГц (без прискорення та уповільнення). Коли Y0.1=OFF, напрям є позитивним.
3. Після завершення вихідних імпульсів M1 змінюється на ON.



API	Інструкція				Операнд						опис						
1405	D	PUDRI			Module ~ ErrCode						Вихід відносного положення модуля PU (з прискоренням і уповільненням)						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Module								●					○	○			
Axis								●					○	○			
RTarPosi								●					○	○			
TarSpeed								●					○	○			
Done		●	●	●				●									
Error		●	●	●				●									
ErrCode								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
RTarPosi			●				●						
TarSpeed			●				●						
Done	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DPUDRI
En
Module Done
Axis Error
RTarPosi ErrCode
TarSpeed

- Module** : Номер модуля
Axis : Номер вихідної осі
RTarPosi : Кількість вихідних імпульсів для відносного позиціонування
TarSpeed : Цільова вихідна частота
Done : Прапор завершення
Error : Прапор помилки
ErrCode : Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із FW V1.08.00 або новішої версії. Для ISPSOft ми рекомендуємо використовувати версію програмного забезпечення 3.12 і вище. Якщо для параметра En встановлено значення ON, ця інструкція буде ефективною. Коли налаштування змінюється на ВІМК., вихідний імпульс сповільнюється, доки не припиниться.

2. **Module** встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість — 32. Ця інструкція стосується виключно модулів PU праворуч від ПЛК і не застосовується до модулів PU праворуч від віддаленого модуля. Якщо вказаний модуль не є модулем PU, прапор помилки **Error** зміниться на ON.

3. Axis встановлює номер вихідної осі для вказаного модуля PU. Значення налаштування 1~4 представляють вихідні дані axis1~axis4 зазначеного модуля PU відповідно. Якщо модуль PU не має відповідного номера осі для виведення, прапор помилки **Error** зміниться на ON.

4. RTarPosi встановлює положення для відносного позиціонування. Номер імпульсу є 32-розрядним значенням зі знаком. Коли значення більше 0, вихід буде проходити в позитивному напрямку (а точка виведення напрямку вимкнена). Якщо значення менше 0, вихід буде здійснюватися у негативному напрямку (і точка виведення напрямку буде увімкнена). Коли значення дорівнює 0, прапорець завершення виведення **Done** змінюється на ON.

5. TarSpeed встановлює цільову вихідну частоту (одиниця: Гц). Значення частоти є додатним 32-розрядним цілим числом зі знаком. Якщо значення менше 0, інструкція автоматично використовуватиме доповнення до 2, щоб перетворити значення на додатне ціле число. Якщо значення дорівнює 0, інструкція сповістить модуль про перехід у режим паузи. Фактичний вихід сповільнюється зі швидкістю уповільнення, доки вихідна швидкість не дорівнюватиме 0, а прапор паузи зміниться на ON. Для отримання додаткової інформації зверніться до інструкції PUSTAT.

Назва модуля	Діапазон налаштувань TarSpeed
AS02PU	-200 000 (-200K) ~ 200 000 (200K)
AS04PU	-100 000 (-100K) ~ 100 000 (100K)

6. Після запуску вихідного сигналу цільову частоту можна будь-коли змінити. У фактичній зміні частоти

ПЛК автоматично змінить частоту на основі встановленої швидкості прискорення та уповільнення в інструкції DPUCONF.

7. Коли виходи досягають кількості імпульсів для відносного позиціонування, визначеного **RTarPosi**, прапор **Done**

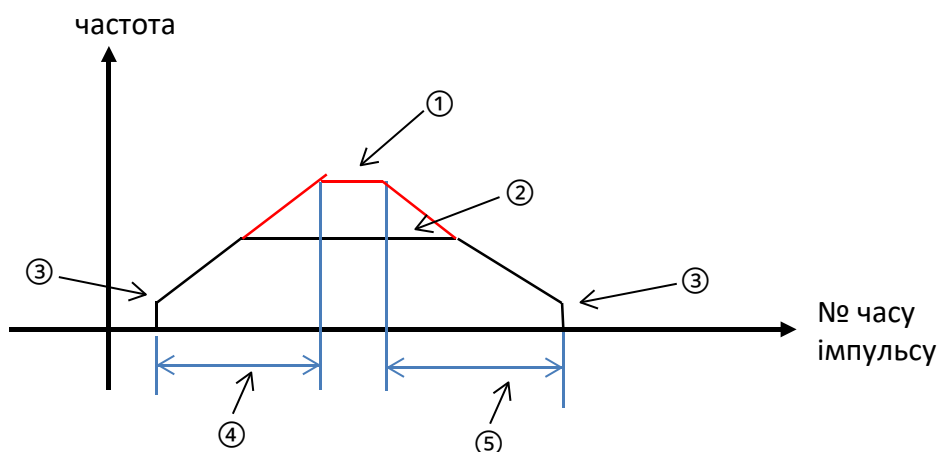
змінюється на ON. Прапорець **Done** потрібно очистити вручну. Інструкція встановлює прапор завершення на ON лише тоді, коли вивід завершено.

8. Якщо під час виведення інструкції виникає будь-яка помилка, прапорець **Error** змінюється на ON. Зверніться до кодів помилок, які показує **ErrCode** для усунення несправностей.

9. Коди помилок, які показує **ErrCode**, перераховані в наведеній нижче таблиці.

Код помилки	опис
16#1400	Модуль не підтримує функцію.
16#1401	Дані, що зберігаються в модулі, є незаконними або перевищують допустимий діапазон.
16#1402	Немає відповіді від модуля; відбувається тайм-аут зв'язку.
16#1403	Такого номера вихідної осі в модулі ПУ немає.
16#1404	Вихідна частота модуля PU є недопустимою.
16#1405	Хоча вихідна вісь, визначена модулем PU, виводить дані, це заборонено повторно вказуйте результат.
16#1406	Модуль PU припиняє вихідний імпульс, коли досягається позитивна межа.
16#1407	Модуль PU припиняє вихідний імпульс при досягненні негативної межі.

10. Ілюстрація кривої прискорення та уповільнення інструкції DPUDRI



① : значення максимальної вихідної частоти. Зверніться до налаштування в інструкції DPUCONF для налаштування параметра.

Або встановіть значення параметра через HWCONFIG.

② : цільова частота, визначена інструкцією виводу модуля PU. Цільова вихідна частота не повинна перевищувати максимальну вихідну частоту. Якщо максимальна вихідна частота перевищена, максимальна вихідна частота вважається вихідною частотою.

③ : Початкове/кінцеве значення вихідної частоти. Зверніться до налаштування в інструкції DPUCONF для налаштування параметра.

Або встановіть значення параметра через HWCONFIG.

④ : значення часу прискорення. Зверніться до налаштування в інструкції DPUCONF для налаштування параметра.

Або встановіть значення параметра через HWCONFIG.

⑤ : Значення часу уповільнення. Зверніться до налаштування в інструкції DPUCONF для налаштування параметра.

Або встановіть значення параметра через HWCONFIG.

Прискорення та уповільнення, якими керує модуль PU, виконується відповідно до фіксованого нахилу. Таким чином, фактичний час прискорення та час уповільнення змінюються залежно від вихідної цільової частоти. Формула для розрахунку швидкості прискорення та швидкості уповільнення відповідно показана нижче.

(Макс. вихідна частота - стартова частота)/час розгону; (Макс.

вихідна частота - кінцева частота)/час уповільнення.

Приклад програмування:

1. Налаштуйте параметри модуля AS02PU в HWCONFIG. Встановіть «Вісь 1, DOG, спрацьовує по наростаючому фронту» для точки введення X0.0, «Вісь 1, фаза Z, спрацьовує по наростаючому фронту» для точки введення X0.1 і «Фаза A Y0.0+ Фаза B Y0.1 » для вихідної точки. Інші параметри осі 1 встановлені за замовчуванням.

2. Коли M10 встановлено на ON, інструкція DPUCONF буде виконана для осі 1, щоб змінити вихідне налаштування на 1 (імпульс Y0.0+ напрямок Y0.1) у режимі. Встановіть SSpeed (швидкість для запуску) на 200 Гц, Atime (час прискорення) на 200 мс, Dtime (час уповільнення) на 200 мс, MSpeed (максимальна вихідна частота) на 100 кГц, Z_NO (кількість фаз Z, які слід шукати після повернення до вихідне положення) до 1, зміщення (кількість виходів після повернення у вихідне положення) до

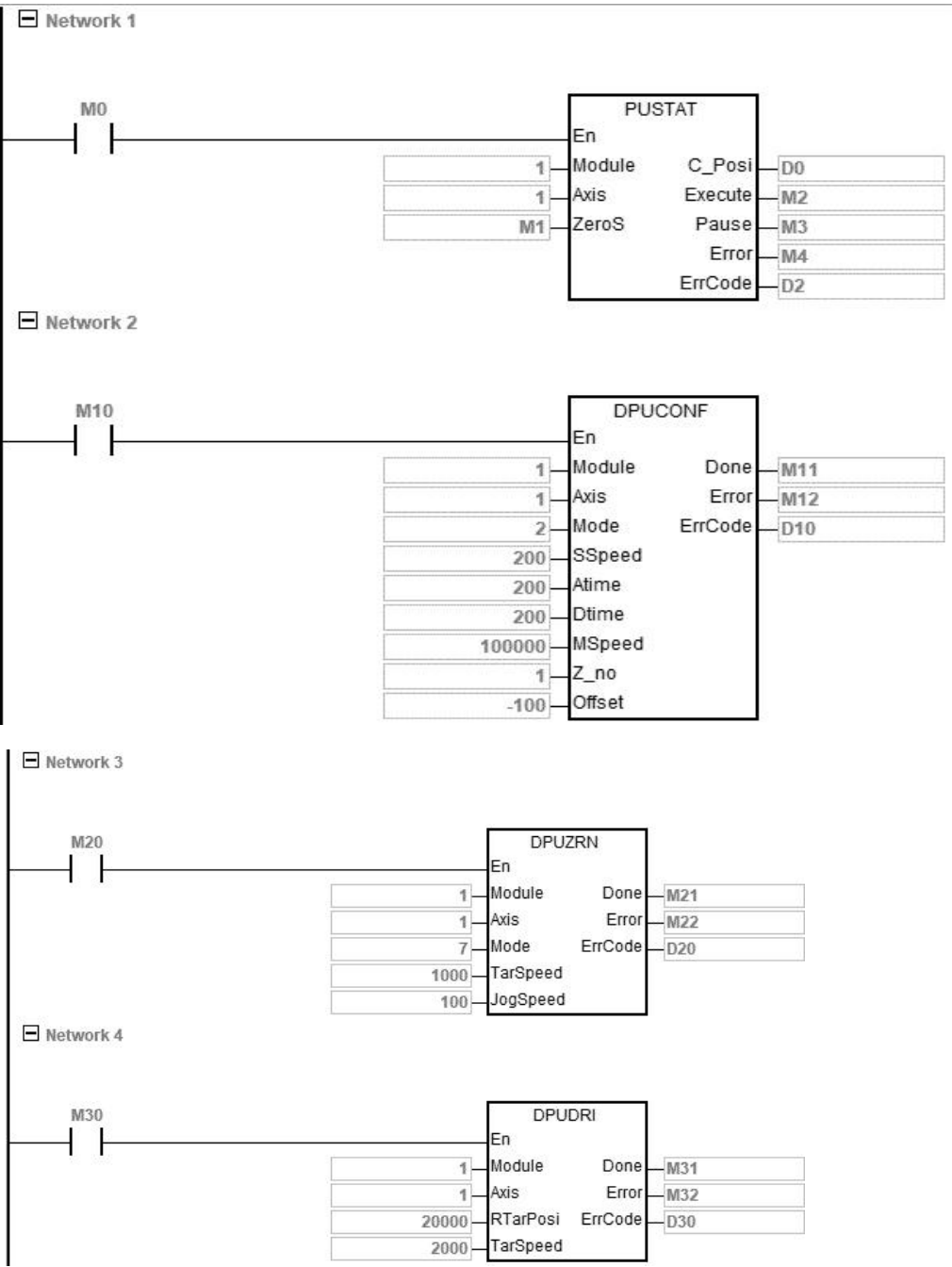
-100 і встановіть M11 на ON, коли прапор Done зміниться на ON.

3. Коли M0=ON, виконайте PUSTAT на осі 1, щоб прочитати вихідний стан модуля PU. Щоб очистити поточну вихідну позицію, ви можете встановити для M1 значення ON, щоб поточна позиція D0 (спрацьовує по наростаючому фронту) на осі 1 мала значення 0.

4. Коли M20=ON, виконайте DPUZRN, щоб виконати наведення на вісь 1, і виконайте DPUSTAT, щоб відобразити поточне положення в D0. Вихідна точка Y0.0 видає імпульси на частоті 1 кГц для пошуку вихідного сигналу в позитивному напрямку. Після досягнення сигналу близького вихідного сигналу (DOG) і X0.0=ON, вісь починає сповільнюватися та рухатися в негативному напрямку зі швидкістю Jog Speed 100 Гц. Коли X0.0=OFF, вісь рухається в позитивному напрямку для пошуку Z-фази, доки не буде виявлено перший висхідний фронт сигналу X0.1 (Z-фаза), потім рухається в негативному напрямку після завершення 100 вихідних імпульсів. Нарешті, M21 зміниться на ON після того, як прапор Done зміниться на ON.

5. Якщо M30=ON, DPUDRI буде виконано на осі 1 для виведення відносного позиціонування. І виконайте DPUSTAT, щоб відобразити поточну позицію в D0. І звідти вихідна точка Y0.0 видає 20 000 імпульсів на частоті 2 кГц (відносна адресація). Якщо Y0.1=OFF, напрям позитивний. Виконайте DPUSTAT, щоб відобразити поточну позицію в D0. Нарешті, M31 зміниться на ON після того, як прапор Done зміниться на ON.

Input Point Setting						
	Name	Setting Value	Unit	Default	Minimum	Maximum
▶	X0.0 Mode	Axis 1 DOG Raise edge trigger ▼		General Input	-	-
	X0.1 Mode	Axis 1 Z Raise edge trigger ▼		General Input	-	-
	X0.2 Mode	General Input ▼		General Input	-	-
	X0.3 Mode	General Input ▼		General Input	-	-
	X0.4 Mode	General Input ▼		General Input	-	-
	X0.0 Filter Time	10	ms	10	0	25
	X0.1 Filter Time	10	ms	10	0	25
	X0.2 Filter Time	10	ms	10	0	25
	X0.3 Filter Time	10	ms	10	0	25
	X0.4 Filter Time	10	ms	10	0	25
Output Point Setting						
	Name	Setting Value	Unit	Default	Minimum	Maximum
▶	Y0.0/Y0.1 Output Mode	A phase(Y0.0)+B phase(Y0... ▼		Pulse(Y0.0)+Directi...	-	-
	Y0.2/Y0.3 Output Mode	Pulse(Y0.2)+Direction(Y0.3) ▼		Pulse(Y0.2)+Directi...	-	-
Axis 1						
	Name	Setting Value	Unit	Default	Minimum	Maximum
▶	Start speed	100	Hz	100	0	10000
	Accelerate time	100	ms	100	0	10000
	Decelerate time	100	ms	100	0	10000
	Max speed	200000	Hz	200000	100	200000
	Seek the set number of Z phase after th...	0		0	-100	100
	Output the offset position after the homi...	0	Pulse	0	-10000	10000
	Home mode select	Mode 1 : the axis starts to ... ▼		Mode 1 : the axis st...	-	-
	Negative limited position	0	Pulse	0	-2147483648	2147483647
	Positive limited position	0	Pulse	0	-2147483648	2147483647

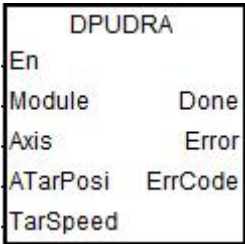


API	Інструкція				Операнд								опис				
1406	D	PUDRA			Module ~ ErrCode								Вихід абсолютної адресації модуля PU (з прискоренням і уповільненням)				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Module								●					○	○			
Axis								●					○	○			
ATarPosi								●					○	○			
TarSpeed								●					○	○			
Done		●	●	●													
Error		●	●	●													
ErrCode								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
ATarPosi			●				●						
TarSpeed			●				●						
Done	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



- Module** : Номер модуля
- Axis** : Номер вихідної осі
- ATarPosi** : Кількість вихідних імпульсів для абсолютної адресації
- TarSpeed** : Цільова вихідна частота
- Done** : Прапор завершення/паузи
- Error** : Прапор помилки
- ErrCode** : Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із FW V1.08.00 або новішої версії. Для ISPSOft ми рекомендуємо використовувати версію програмного забезпечення

3.12 і вище. Якщо для параметра En встановлено значення ON, ця інструкція буде ефективною. Коли налаштування змінюється на ВИМК., вихідний імпульс сповільнюється, доки не припиниться.

2. ATarPosi — позиція для абсолютної адресації. Номер вхідного імпульсу є 32-бітним значенням зі знаком. Module PU автоматично порівнює його з поточним положенням. Якщо результат порівняння більший за 0, вихід буде проведено в позитивному напрямку (а вихідна точка напрямку вимкнена). Якщо результат порівняння менший за 0, вихід буде проведено у негативному напрямку, а точка виведення напрямку буде включена). Коли значення дорівнює 0, інструкція встановлює прапор **Done** на ON.

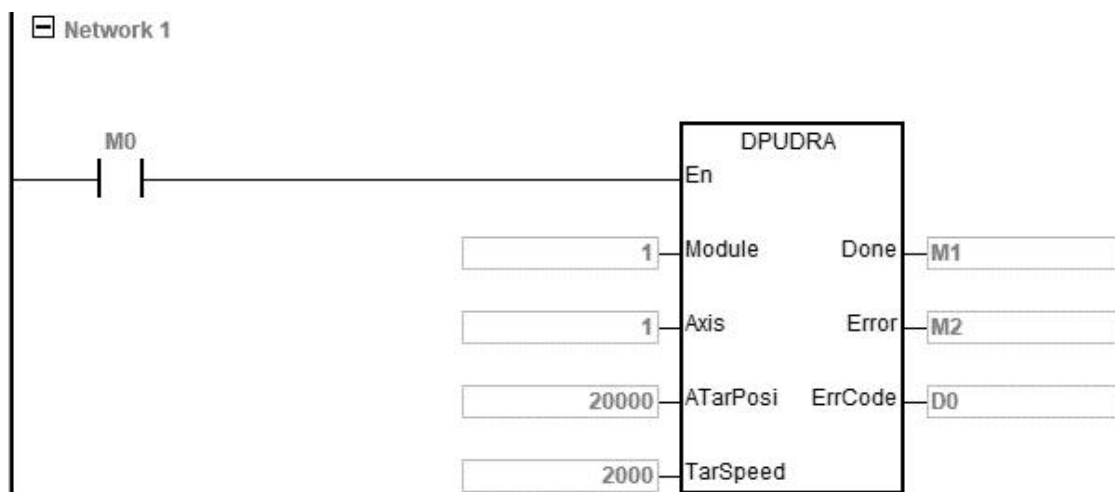
3. Зверніться до інструкції DPUDRI для пояснення інших параметрів.

Приклад програмування:

1. Налаштуйте параметри модуля AS02PU в HWCONFIG. Встановіть «Pulse Y0.0+ Direction 0.1» для вхідної точки X0.0. Інші параметри осі 1 встановлені за замовчуванням.

2. Коли M0=ON, виконайте DPUDRA на осі 1 для виведення імпульсів від Y0.0 на частоті 2 кГц до того, як поточна позиція досягне 20 000 (абсолютна адресація). Напрямок буде позитивним, якщо Y0.1=OFF.

3. Після завершення вихідного імпульсу M1 зміниться на ON.

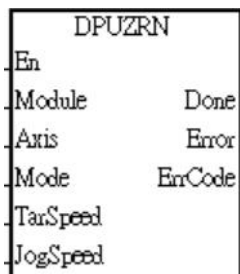


API	Інструкція				Операнд						опис					
1407	D	PUZRN			Module ~ ErrCode						Наведення модуля ПУ					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Module								●					○	○		
Axis								●					○	○		
Режим								●					○	○		
TarSpeed								●					○	○		
JogSpeed								●					○	○		
Done		●	●	●												
Error		●	●	●												
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
Режим		●				●							
TarSpeed			●				●						
JogSpeed		●				●							
Done	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	AS

СИМВОЛ

**Module** : Номер модуля**Axis** : Номер вихідної осі**Режим** : Вибір режиму самонаведення**TarSpeed** : Максимальна вихідна частота для самонаведення**JogSpeed** : Частота перемикавання для наведення**Done** : Прапор завершення**Error** : Прапор помилки**ErrCode** : Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із FW V1.08.00 або новішої версії. Для ISPSOFT ми рекомендуємо використовувати версію програмного забезпечення

3.12 і вище. Якщо для параметра En встановлено значення ON, ця інструкція буде ефективною. Після того, як налаштування змінюється на ВІМК., вихідний імпульс негайно припиняється, навіть якщо дію наведення не завершено.

2. Module встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість — 32. Ця інструкція стосується виключно модулів PU праворуч від ПЛК і не застосовується до модулів PU праворуч від віддаленого модуля. Якщо вказаний модуль не є модулем PU, прапор помилки **Error** зміниться на ON.

3. Axis встановлює номер вихідної осі для вказаного модуля PU. Значення налаштування 1~4 представляють вихідні дані axis1~axis4 зазначеного модуля PU відповідно. Якщо модуль PU не має відповідного номера осі для виведення, прапор помилки **Error** зміниться на ON.

4. Mode встановлює режим наведення. Пояснення режимів наведено в наступній таблиці.

Значення режиму	Опис функції	Використовуйте відповідний вхід балів (встановлюється в HWCONFIG)	Зауваження
0	Очистити поточну позицію до 0.	Жодного	
1	Налаштування на домашній вимикач; після самонаведення є зроблено, вісь зупиняється в негативному напрямку руху.	DOG	Точка DOG вказується в HWCONFIG, позитивний вихідний перемикач або негативний вихідний перемикач.
2	Налаштування на домашній вимикач; після самонаведення є завершено, вісь зупиняється в позитивному напрямку руху.		
3	Після завершення виконання режиму 1 шукайте кількість Z фаз.	Вхід фази DOG і Z	Використовуйте інструкцію DPUCONF, щоб встановити кількість фаз Z.
4	Після завершення виконання режиму 2 шукайте кількість Z фаз.		
5	Після завершення виконання режиму 1 виведіть офсетна позиція.	DOG	Використовуйте інструкцію DPUCONF для виведення положення зміщення.
6	Після завершення виконання режиму 2 виведіть офсетна позиція.		
7	Після завершення виконання режиму 1 шукайте кількість фаз Z, а потім виведіть позицію зсуву.	Вхід фази DOG і Z	Використовуйте інструкцію DPUCONF, щоб встановити кількість фаз Z, а потім вивести положення зміщення.
8	Після завершення виконання режиму 1 шукайте кількість фаз Z, а потім виведіть позицію зсуву.		
255	Змініть поточну вихідну позицію для осі.	Жодного	Використовуйте значення параметра TarSpeed
Інший	Зарезервований		

Примітка 1: Зазначена поведінка повернення до початкового місця може не реалізуватися, якщо точки введення для вибраного режиму не використовуються разом із налаштуваннями в HWCONFIG.

Примітка 2: Режим 1~4

FW V1.02.00: після виконання цієї інструкції поточне вихідне положення осі не буде скинуто до нуля. Щоб очистити його, потрібно скористатися інструкцією PUSTAT.

FW V1.02.10 або пізнішої версії: після виконання цієї інструкції поточне вихідне положення осі буде автоматично скинуто до нуля.

Примітка 3: Режим 5~8

V1.02.00: після виконання цієї інструкції поточна вихідна позиція осі не буде скинута до нуля. Вам потрібно використовувати режим 255, щоб змінити поточну вихідну позицію, щоб очистити її.

V1.02.10 або пізнішої версії: після виконання цієї інструкції поточна вихідна позиція осі буде автоматично скинута до нуля.

5. **TarSpeed** встановлює максимальну вихідну частоту для наведення, коли значення режиму встановлено від 1 до 8. Значення налаштування є 32-бітним значенням зі знаком. Негативний і позитивний знаки вказують на початковий напрямок самонаведення.

Нижче наведено відповідні значення в TarSpeed для модулів AS02/04PU.

Назва модуля	Діапазон налаштувань TarSpeed, коли режим становить від 1 до 8.
AS02PU	-200 000 (-200K) ~ 200 000 (200K)
AS04PU	-100 000 (-100K) ~ 100 000 (100K)

Якщо режим 255, значення в TarSpeed оновлюється як поточна вихідна позиція модуля PU.

6. **JogSpeed** — це частота пробіжок для досягнення вихідного положення. Значення налаштування — це 16-бітне значення зі знаком у діапазоні 1~10 000 (Гц).

7. Коли зазначена операція завершується під час виконання інструкції, прапор **Done** змінюється на ON. Позначку «**Done**» потрібно скинути вручну. Інструкція встановлює прапор завершення на ON лише тоді, коли вивід завершено.

8. Якщо під час виведення інструкції виникає будь-яка помилка, прапорець **Error** змінюється на ON. Зверніться до кодів помилок, які показує **ErrCode** для усунення несправностей.

9. Коди помилок, які показує **ErrCode**, перераховані в наведеній нижче таблиці.

Код помилки	опис
16#1400	Module не підтримує функцію.
16#1401	Дані, що зберігаються в модулі, є незаконними або перевищують допустимий діапазон.
16#1402	Немає відповіді від модуля; відбувається тайм-аут зв'язку.

16#1403	Такого номера вихідної осі в модулі ПУ немає.
16#1404	Вихідна частота модуля PU є недопустимою.
16#1405	Вивідна вісь, визначена модулем PU, виводить дані. Це не дозволено повторно вказуйте результат.
16#1406	Module PU припиняє вихідний імпульс, коли досягається позитивна межа.
16#1407	Module PU припиняє вихідний імпульс при досягненні негативної межі.

Приклад програмування: зверніться до опису інструкції DPUDRI (API 1405) для отримання додаткової інформації.

API	Інструкція				Операнд						опис					
1408	D	PUJOG			Module - ErrCode						Руховий вихід модуля PU					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Module								●					○	○		
Axis								●					○	○		
JogSpeed								●					○	○		
Busy		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
JogSpeed			●				●						
Busy	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DPUJOG	
En	
Module	Busy
Axis	Error
JogSpeed	ErrCode

Module : Номер модуля

Axis : Номер вихідної осі

JogSpeed : Вихідна частота Jog

Busy : Вихід у виконанні

Error : Прапор помилки

ErrCode : Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із FW V1.08.00 або новішої версії. Для ISPSOft ми рекомендуємо використовувати версію програмного забезпечення

3.12 і вище. Якщо для параметра En встановлено значення ON, ця інструкція буде ефективною. Коли налаштування змінюється на ВИМК., вихідний імпульс сповільнюється, доки не припиниться.

2. **Module** встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість — 32. Ця інструкція стосується виключно модулів PU праворуч від ПЛК і не застосовується до модулів PU праворуч від віддаленого модуля. Якщо вказаний модуль не є модулем PU, прапор помилки **Error** зміниться на ON.

3. **Axis** встановлює номер вихідної осі для вказаного модуля PU. Значення налаштування 1~4 представляють вихідні дані axis1~axis4 зазначеного модуля PU відповідно. Якщо модуль PU не має відповідного номера осі для виведення, прапор помилки **Error** зміниться на ON.

4. **JogSpeed** встановлює вихідну частоту Jog. Значення налаштування є 32-бітним значенням зі знаком. Коли значення більше 0, вихід буде проходити в позитивному напрямку (а точка виведення напрямку вимкнена). Якщо значення менше 0, вихід буде здійснюватися у негативному напрямку (і точка виведення напрямку буде увімкнена). Коли значення дорівнює 0, вихід припиняється. Значення JogSpeed можна змінювати під час виконання виходів, а діапазон налаштувань показано таким чином.

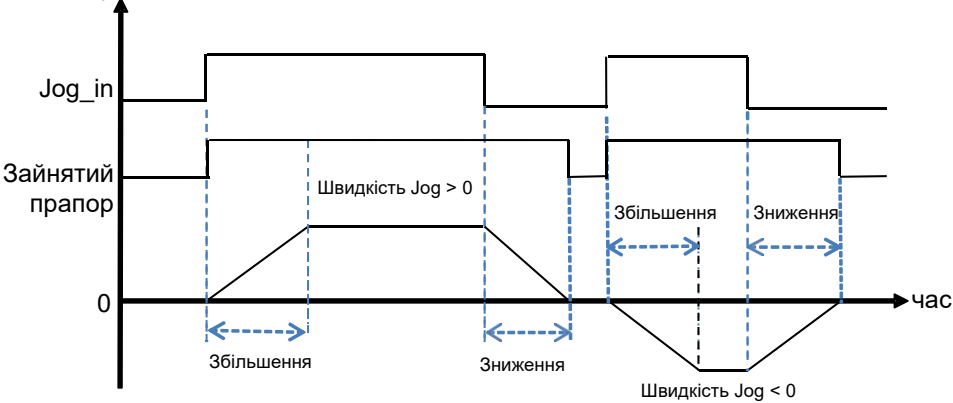
Назва модуля	Діапазон налаштування JogSpeed
AS02PU	-200 000 (-200K) ~ 200 000 (200K)
AS04PU	-100 000 (-100K) ~ 100 000 (100K)

5. Якщо під час виведення інструкції виникає будь-яка помилка, прапорець **Error** змінюється на ON. Зверніться до помилки коди, які показує **ErrCode** для усунення несправностей.

6. Коди помилок, які показує **ErrCode** , перераховані в наведеній нижче таблиці.

Код помилки	опис
16#1400	Module не підтримує функцію.
16#1401	Дані, що зберігаються в модулі, є незаконними або перевищують допустимий діапазон.
16#1402	Немає відповіді від модуля; відбувається тайм-аут зв'язку.
16#1403	Такого номера вихідної осі в модулі ПУ немає.
16#1404	Вихідна частота модуля PU є недопустимою.
16#1405	Вивідна вісь, визначена модулем PU, виводить дані. Це не дозволено повторно вказуйте результат.
16#1406	Module PU припиняє вихідний імпульс, коли досягається позитивна межа.
16#1407	Module PU припиняє вихідний імпульс при досягненні негативної межі.

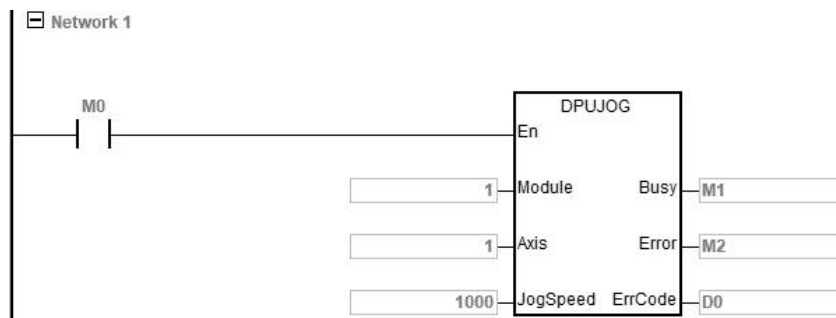
7. Дивіться часову діаграму виведення, як показано нижче. (Jog_in – це перемикач для запуску інструкції **En** , а прапорець Busy – це **зайнятості** .)



8. Після вимкнення інструкції PUJOG і вимкнення прапора «**Зайнято**» можна виконувати інший контроль виведення.

Приклад програмування

1. Налаштуйте параметри модуля AS02PU в HWCONFIG. Встановіть «Pulse Y0.0+ Direction 0.1» для вихідної точки. Параметри осі 1 встановлені за замовчуванням.
2. Коли M0=ON, виконайте DPUJOG на осі 1 для виконання виходів Jog. Імпульси будуть виводитися з Y0.0 на частоті 1 кГц. Напрямок буде позитивним, якщо Y0.1=OFF.
3. Коли інструкція виконується, M1 зміниться на ON.



API		Інструкція			Операнд						опис					
1409		D	PUMPG		Module ~ ErrCode						Вихід MPG модуля PU					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Module								●					○	○		
Axis								●					○	○		
InMode								●					○	○		
InPulse								●								
InSpeed								●								
Rate								●								○
OPulse								●								
OSpeed								●								
Error		●	●	●				●								
ErrCode								●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Axis		●				●							
InMode		●				●							
InPulse			●				●						
InSpeed			●				●						
Rate									●				
OPulse			●				●						
OSpeed			●				●						
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DPUMPG	
En	
Module	OPulse
Axis	OSpeed
InMode	Error
InPulse	ErrCode
InSpeed	
Rate	

Module	: Номер модуля
Axis	: Номер вихідної осі
InMode	: Режим введення кодера та множення частоти для підрахунку
InPulse	: Кількість імпульсів, які були введені
InSpeed	: Виявлена вхідна частота
Rate	: Швидкість введення/виведення (число з плаваючою комою)
OPulse	: Кількість імпульсів, які були виведені
OSpeed	: Частота, з якою виходять імпульси
Error	: Прапор помилки
ErrCode	: Код помилки

Пояснення

- Ця інструкція доступна для ПЛК із FW V1.08.20 або новішої версії. Для ISPSOFT ми рекомендуємо використовувати версію програмного забезпечення 3.12 і вище.
- Інструкція PUMPG застосовна лише до модуля AS02PU, а прошивка для модуля має бути V1.04.00 або вище. Якщо для параметра En встановлено значення ON, ця інструкція буде ефективною. Як тільки налаштування змінюється на ВІМК., вихідний імпульс буде негайно припинено.
- Module** встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість — 32. Ця інструкція стосується виключно модулів PU праворуч від ПЛК і не застосовується до модулів PU на праворуч від віддаленого модуля. Якщо вказаний модуль не є модулем PU, прапор помилки **Error** зміниться на ON.
- Axis** встановлює номер вихідної осі для вказаного модуля PU. Значення параметрів 1~4 представляють вісь 1~4 вихід зазначеного модуля PU відповідно. Якщо модуль PU не має відповідного номера осі для виведення, прапор помилки **Error** зміниться на ON.
- InMode** встановлює режим введення джерела кодера та множення частоти для підрахунку. Див. пояснення значення **InMode** у наведеній нижче таблиці.

Значення	Режими введення
	Режим введення; встановіть наступні значення, інакше для запуску модуль використовуватиме значення за замовчуванням.
16#0000	Чотирикратний частотний вхід фази A/B (за замовчуванням). Фаза A веде до фази B, вказуючи на підрахунок у позитивному напрямку. Фаза B веде до фази A, вказуючи на підрахунок у негативному напрямку.
16#0001	Вхід фази A/B одноразової частоти. Фаза A веде до фази B, вказуючи на підрахунок у позитивному напрямку. Фаза B веде до фази A, вказуючи на підрахунок у негативному напрямку.
16#0002	Подвійний частотний вхід фази A/B. Фаза A веде до фази B, вказуючи на підрахунок у позитивному напрямку.

	Фаза В веде до фази А, вказуючи на підрахунок у негативному напрямку.
16#0003	Зарезервований
16#0004	Чотирикратний частотний вхід фази А/В Фаза А веде до фази В, вказуючи на підрахунок у позитивному напрямку. Фаза В веде до фази А, вказуючи на підрахунок у негативному напрямку.
16#0005	Імпульсний + спрямований вхід (А+/А-: імпульсний вхід; В+/В-: спрямований вхід) Фаза В ON: підрахунок у негативному напрямку. Фаза В OFF: підрахунок у позитивному напрямку. Фаза А: підрахунок починається за допомогою спрацьовування наростаючого фронту.
16#0006	Імпульсний + спрямований вхід (А+/А-: імпульсний вхід; В+/В-: спрямований вхід) Фаза В ON: підрахунок у позитивному напрямку. Фаза В OFF: підрахунок у негативному напрямку Фаза А: підрахунок починається за допомогою спрацьовування по наростаючому фронту.
16#0007	Однофазний імпульсний вхід (А+/А-: імпульсний вхід) Фаза А: підрахунок починається за допомогою спрацьовування по наростаючому фронту.
інші	Зарезервований

6. **InPulse** відображає кількість уже введених імпульсів, яка є 32-бітним значенням зі знаком. Кожного разу інструкція є запущено, модуль PU автоматично скине значення до 0, а потім почне підрахунок.

7. **InSpeed** відображає вже визначену вхідну частоту, яка є 32-бітним значенням. Основний час для частоти виявлення становить 20 мс. Таким чином, виявлена вхідна частота дорівнює 0, якщо немає значення підрахунку протягом 20 мс. Якщо є значення підрахунку в межах 20 мс, вихід починається з мінімальної частоти 50 Гц. Навіть якщо значення **OSpeed** нижче за 50 Гц через перетворення на основі значення **швидкості**, вихід все одно виконується на 50 Гц.

8. **Rate** – це швидкість введення/виведення, а значення – число з плаваючою комою. Кількість фактичних вихідних імпульсів і частота відповідно дорівнює кількості вхідних імпульсів і частоті, помноженій на значення швидкості.

Наприклад: вхідна частота становить 100 Гц, а швидкість – 0,5. Отже, вихідна частота становить $100 \times 0,5 = 50$ Гц. Якщо максимальна вихідна частота після перетворення перевищує 100 кГц, вихідна частота обмежується 100 кГц.

Примітка. Довгий максимальний вихід частоти може призвести до того, що, оскільки MPG припинив роботу, кількість виходів все ще збільшується, і вихід потрібно продовжувати, доки він не завершиться.

9. **OPulse** показує кількість імпульсів, які були виведені. **OSpeed** відображає частоту, з якою виконується вихід. Вони мають 32-розрядні значення зі знаком.

10. Коли інструкцію **DPUMPG** вимкнено, перевірте частоту, на якій виконується вихід, і подивіться, чи досягла вона 0. Якщо інструкцію вимкнути до того, як частота досягне 0, модуль PU негайно припинить виведення, а вихід імпульсів які підраховуються на основі коефіцієнта конверсії, більше не триватимуть.

11. Коди помилок, які показує **ErrCode** , перераховані в наступній таблиці.

Код помилки	опис
16#1400	Module не підтримує функцію.
16#1401	Дані, що зберігаються в модулі, є незаконними або перевищують допустимий діапазон.
16#1402	Немає відповіді від модуля; відбувається тайм-аут зв'язку.
16#1403	Такого номера вихідної осі в модулі ПУ немає.
16#1404	Вихідна частота модуля ПУ є недопустимою.
16#1405	Вивідна вісь, визначена модулем ПУ, виводить дані. Це не дозволено повторно вказуйте результат.
16#1406	Module ПУ припиняє вихідний імпульс, коли досягається позитивна межа.
16#1407	Module ПУ припиняє вихідний імпульс при досягненні негативної межі.

12. Коли інструкція DPUMPG увімкнена або вимкнена, ПЛК повинен повідомити модуль, щоб увімкнути або вимкнути функція високошвидкісного лічильника. Таким чином, інструкцію не можна використовувати разом з API1410 DPUCNT. Інакше може статися, що дві інструкції вмикають або вимикають підрахунок модуля один з одним.

API	Інструкція				Операнд								опис				
1410	D	PUCNT			Module ~ ErrCode								Функція високошвидкісного лічильника модуля PU				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Module								●					○	○			
InMode								●					○	○			
Period								●					○	○			
ZeroS	●	●	●	●				●									
InPulse								●									
InSpeed								●									
Error		●	●	●				●									
ErrCode								●									

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
InMode		●				●							
Period		●				●							
ZeroS	●												
InPulse			●				●						
InSpeed			●				●						
Error	●												
ErrCode			●				●						

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DPUCNT	
En	
Module	InPulse
InMode	InSpeed
Period	Error
ZeroS	ErrCode

- Module :** Номер модуля
- InMode :** Режим введення кодера та множення частоти для підрахунку
- Period :** Період часу для захоплення частоти
- ZeroS :** Очистити лічильник до 0
- InPulse :** Кількість імпульсів, які були введені
- InSpeed :** Кількість імпульсів за цикл
- Error :** Прапор помилки
- ErrCode :** Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із FW V1.08.00 або новішої версії. Для ISPSOFT ми рекомендуємо використовувати версію програмного забезпечення 3.12 і вище.
2. Інструкція DPUCNT підтримує лише модуль AS02PU. Якщо для параметра En встановлено значення ON, ця інструкція буде ефективною. Як тільки налаштування змінюється на ВІМК., підрахунок буде негайно припинено.
3. **Module** встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість — 32. Ця інструкція стосується виключно модулів PU праворуч від ПЛК і не застосовується до модулів PU праворуч від віддаленого модуля. Якщо вказаний модуль не є модулем PU, прапор помилки **Error** зміниться на ON.
4. **InMode** встановлює режим введення джерела кодера та множення частоти для підрахунку. Див.

пояснення значення **InMode** у наведеній нижче таблиці.

Примітка: фаза A веде до фази B, вказуючи на підрахунок у позитивному напрямку.

Фаза B веде до фази A, вказуючи на підрахунок у негативному напрямку

Значення	Режими введення
	Режим введення; встановіть наступні значення, інакше для запуску модуль використовуватиме значення за замовчуванням.
16#0000	Зарезервований
16#0001	Вхід фази A/B одноразової частоти
16#0002	Подвійний частотний вхід фази A/B
16#0003	Зарезервований
16#0004	Чотирикратний частотний вхід фази A/B (за замовчуванням)
16#0005	Імпульсний + спрямований вхід (A+/A-: імпульсний вхід; B+/B-: спрямований вхід) Фаза B ON: підрахунок у негативному напрямку. Фаза B OFF: підрахунок у позитивному напрямку. Фаза A: підрахунок починається за допомогою спрацьовування наростаючого фронту.
16#0006	Імпульсний + спрямований вхід (A+/A-: імпульсний вхід; B+/B-: спрямований вхід) Фаза B ON: підрахунок у позитивному напрямку. Фаза B OFF: підрахунок у негативному напрямку Фаза A: підрахунок починається за допомогою спрацьовування по наростаючому фронту.
16#0007	Однофазний імпульсний вхід (A+/A-: імпульсний вхід) Фаза A: підрахунок починається за допомогою спрацьовування по наростаючому фронту.
інші	Зарезервований

5. **Period** – це значення налаштування часу циклу для захоплення частоти в діапазоні 10 мс ~ 1000 мс. Якщо значення параметра перевищує діапазон, максимальне або мінімальне значення буде автоматично прийнято ПЛК як значення параметра.

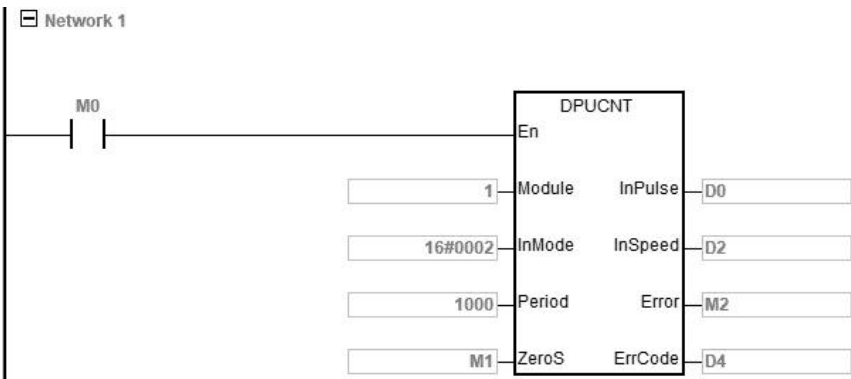
- 6. InPulse** — кількість уже введених імпульсів, яка є 32-бітним значенням зі знаком. Значення підрахунку є зафіксованим значенням. Якщо значення потрібно скинути до 0, просто встановіть **ZeroS** з OFF на ON під час виконання інструкції.
- 7. InSpeed** відображає значення підрахунку для кожного **періоду** часу, яке є 32-бітним значенням зі знаком. Якщо вам потрібно перевести його в значення з одиницею Гц, скористайтеся розрахунковою формулою для перерахунку самостійно.
8. Код помилок, які показує **ErrCode** , перераховані в наведеній нижче таблиці.

Код помилки	опис
16#1400	Module не підтримує функцію.
16#1401	Дані, що зберігаються в модулі, є незаконними або перевищують допустимий діапазон.
16#1402	Немає відповіді від модуля; відбувається тайм-аут зв'язку.
16#1406	Module PU не підтримує функцію підрахунку.

9. Коли інструкція DPUCNT увімкнена або вимкнена, ПЛК повинен повідомити модуль, щоб увімкнути або вимкнути функцію високошвидкісного лічильника. Таким чином, інструкцію не можна використовувати разом з API1409 DPUMPG. Інакше може статися, що дві інструкції вмикають або вимикають підрахунок модуля один з одним.

Приклад програмування

1. Коли M0=ON, буде виконано PUCNT, а **InMode** встановлено на вхід фази A/B з подвійною частотою.
2. Коли M1=ON, підрахована кількість імпульсів на осі 1 у **InPulse** буде очищена.
3. Коли кількість імпульсів становить 100, а вхідна частота становить 10 Гц, значення в **InPulse** та **InSpeed** відобразатимуться як 200 імпульсів і 20 Гц.



API	Інструкція				Операнд							опис					
1415	D	LCCAL			Group, модуль ~ ErrCode							Калібрування каналу модуля LC					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Group								●					○	○			
Module								●					○	○			
ChNo								●					○	○			
Trigger	●	●	●	●				●									
TPoint								●					○	○			
TBara								●									
CPoint								●									
Done		●	●	●				●									
Done		●	●	●				●									
Error		●	●	●				●									
ErrCode								●									

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
ChNo		●				●							
Trigger	●												
TPoint		●				●							
TBara									●				
CPoint		●				●							
Done	●												
Done	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

LCCAL	
En	
Group	CPoint
Module	Done
ChNo	ADone
Trigger	Error
TPoint	ErrCode
TWeight	

Group	: Номер групи
Module	: Номер модуля
ChNo	: Номер каналу
Trigger	: Запустить калібрування
TPoint	: Кількість усіх точок калібрування
TBara	: Значення калібрувальної ваги
CPoint	: Номер точки, для якої завершено калібрування.
Done	: Прапор завершення одного калібрування
Done	: Позначка завершення калібрування
Error	: Прапор помилки
ErrCode	: Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК з FW V1.08.00 або пізнішої версії та ISPSOft V3.09 або пізнішої версії.
2. Інструкція LCCAL підтримує лише модуль AS02LC. Перш ніж використовувати інструкцію, ви повинні дізнатися про позицію конфігурації поточного модуля з HWCONFIG.
3. **Group** – це номер групи зазначеного модуля LC, підключеного праворуч від ПЛК або віддаленого модуля. The номер ПЛК дорівнює 0, номер першого віддаленого модуля дорівнює 1 і так далі. Максимальний номер групи становить 15. Якщо вказаний модуль не є модулем LC, прапорець **Error** зміниться на ON.
4. **Module** встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість – 32. Якщо вказаний модуль не є модулем LC, прапор помилки **Error** зміниться на ON.
5. **ChNo** – номер каналу вказаного модуля LC. Якщо вхідне значення не є номером каналу модуля LC, прапор помилки **Error** зміниться на ON.
6. **Trigger** — це команда запуску одноточкового калібрування. Коли **тригер** змінюється з ВІМК. на УВІМК., модуль LC буде сповіщено про калібрування. **Done** змінюється на УВІМК. після завершення калібрування. Якщо було виконано калібрування за всіма точками зроблено, **ADone** також зміниться на ON. Перед калібруванням наступної точки калібрування потрібно переконатися, що **параметр «Done»** змінився на «УВІМК.», а потім встановити **«Запуск»** на «ВІМК.». Тоді інструкція зніме позначку **«Done»**, коли **тригер** зміниться з УВІМК. на ВІМК.
7. **TPoint** — це загальна кількість балів для калібрування. Після запуску інструкції значення не можна знову змінити, оскільки значення **TPoint** було передано до модуля LC для калібрування, коли інструкція запускається спочатку.
8. **TWeight** — це значення калібрувальної ваги, яке займає два пристрої. Якщо **TPoint** дорівнює 3 і початковим пристроєм для **TWeight** є D200, тут **TWeight** займає 6 пристроїв, D200~D205. Значення калібрувальної ваги для першого **TPoint** знаходиться в D200 (нижче слово) і D201 (високе слово). Значення калібрувальної ваги для другого **TPoint** знаходиться в D202 (нижче слово) і D203 (високе слово). Значення калібрувальної ваги для третьої **TPoint** знаходиться в D204 (Low Word) і D205

(Високе слово). Після початку виконання цієї інструкції значення **TWeight** надсилається до модуля LC для калібрування. Ви не можете змінити значення **TWeight** після виконання інструкції. значення першої точки для калібрувальної ваги повинно бути 0 (корекція нуля). Якщо ні, прапорець помилки буде ON.

9. Щойно **ADone** зміниться з ВІМК. на УВІМК., повне калібрування буде зупинено. Калібрування можна виконати знову, якщо інструкцію LCCAL знову ввімкнути після вимкнення.

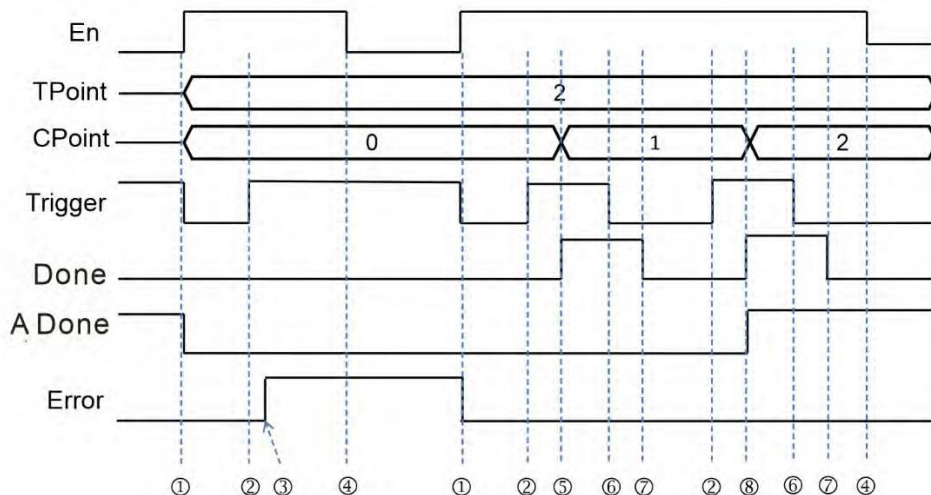
10. **CPoint** — це кількість точок, які були відкалібровані та не можуть бути змінені користувачами. Коли значення **CPoint** $\geq$ **TPoint**, вважається, що калібрування завершено, і прапор **ADone** змінюється на ON.

11. Коли інструкція LCCAL увімкнена спочатку, значення **CPoint** автоматично скидається на 0, а **тригери, готові та ADone** змінюються на ВІМК., і калібрування готується. Після того, як **тригер** встановлено з ВІМК. на УВІМК. і модуль LC завершить калібрування, значення **CPoint** буде автоматично додано на 1, а прапорець «**Done**» зміниться на УВІМК. За значенням можна спостерігати поточні точки, для яких було завершено калібрування. Наприклад, оскільки значення **CPoint** дорівнює 2, це означає, що **тригер** запускається двічі, і модуль завершив калібрування за 2 точками.

12. Коди помилок, які показує **ErrCode**, перераховані в наведеній нижче таблиці.

Код помилки	опис
16#1410	Error в номері групи LC або номері модуля
16#1411	Module LC не має такого номера каналу
16#1412	Error запису модуля LC або тайм-аут зв'язку
16#1413	Module LC завершив калібрування. Вимкніть інструкцію та повторно запустіть її.
16#1414	Перше значення для калібрувальної ваги не дорівнює 0.
16#1415	Значення калібрувальної ваги не є значенням з плаваючою комою.
16#1416	Значення перевищує діапазон вимірювання.
16#1417	Значення перевищує допустиму вагу.
16#1418	Error калібрування

13. Дивіться діаграму послідовності в прикладі калібрування.



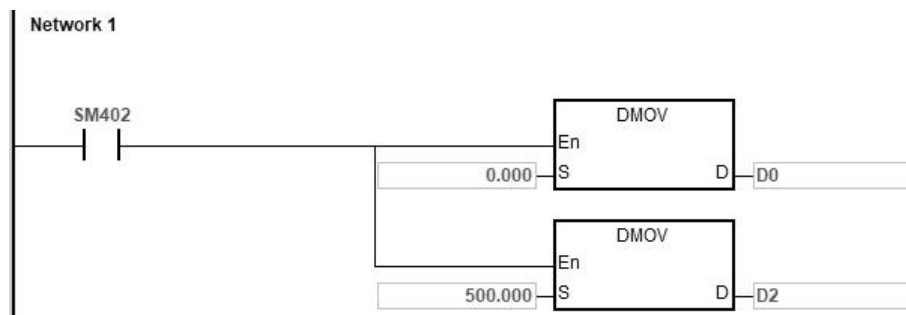
Пояснення часових точок на наведеній вище діаграмі послідовності:

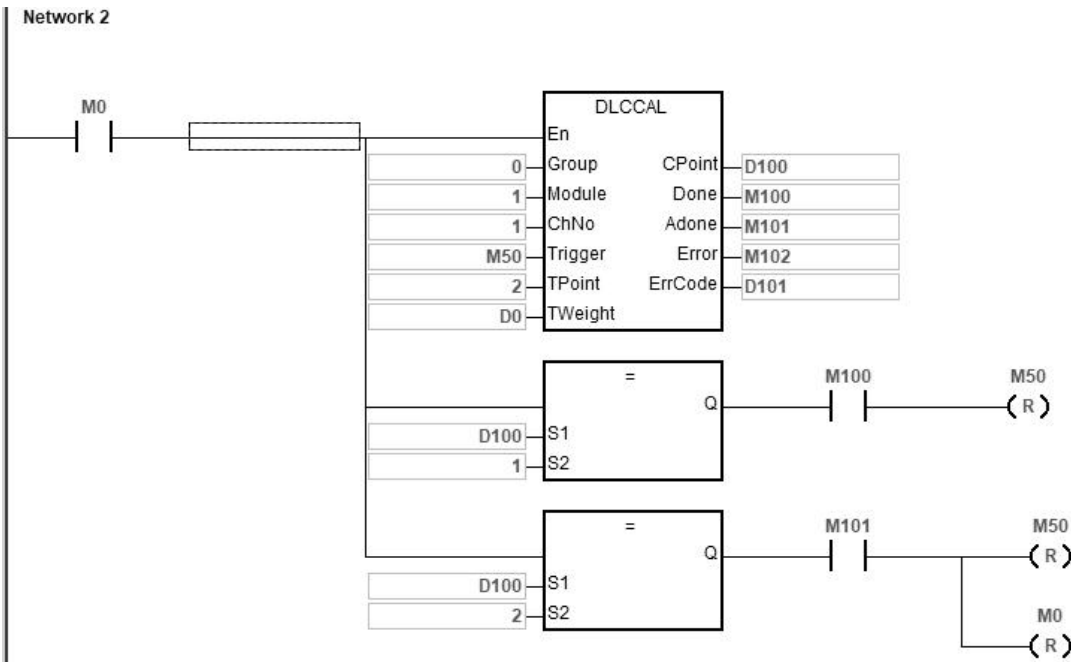
- ① → Інструкцію LCCAL увімкнено, а значення **CPoint** і прапорці **Trigger** , **Done** , **ADone** та **Error** скидаються автоматично.
- ② → Активуйте прапорець калібрування.
- ③ → Інструкція знаходить помилку номера модуля та відображає код помилки помилки номера модуля LC
- ④ → Вимкніть інструкцію LCCAL.
- ⑤ → Після тригера модуль LC завершує калібрування за однією точкою, і значення **CPoint** додається на 1, а «**Done**» змінено на «УВІМК.».
- ⑥ → Зніміть прапорець тригерного сигналу **Trigger**.
- ⑦ → Згодом прапорець **Done** знімається інструкцією.
- ⑧ → Після тригера модуль LC завершує повне калібрування, і значення **CPoint** додається на 1, а обидва прапорці **Done** і **ADone** змінюються на ON.

приклад

Коли ПЛК працює, значення першої точки для калібрувальної ваги дорівнює 0, записаному в D0 і D1. Значення другої точки для калібрувальної ваги становить 500,0, записане в D2 і D3.

- ① → Коли M0 перемикається на ON, ви можете використовувати інструкцію DLCCAL для калібрування.
- ② → Переконайтеся, що на платформі для зважування немає вантажу. Встановіть M50 на ON, а потім почніть корекцію нуля (встановіть значення першої точки на нуль). Коли D100 = 1 і M100 увімкнено, перша точка відкалібрована. M50 вимикається.
- ③ → Помістіть вагу 500 г на платформу для зважування, встановіть M50 на ON, а потім почніть калібрування 500 г для другої точки. Коли D100 = 2 і M101 увімкнено, другу точку відкалібровано. M50 перемикається в положення OFF, а M0 - у положення OFF.





API	Інструкція				Операнд								опис			
1416	D	LCWEI			Group, модуль ~ ErrCode								Зчитування значення ваги через модуль LC			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Group								●					○	○		
Module								●					○	○		
ChNo								●					○	○		
Stable								●					○	○		
ZeroS	●	●	●	●				●								
TareS	●	●	●	●				●								
TareW								●								
Weight								●								
Статус								●								
Error		●	●	●				●								
ErrCode								●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
ChNo		●				●							
Stable		●				●							
ZeroS	●												
TareS	●												
TareW									●				
Weight									●				
Статус		●				●							
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

- Group : Номер групи
- Module : Номер модуля
- ChNo : Номер каналу
- Stable : Укажіть діапазон стабільної ваги (0,0~100000,0)
- ZeroS : Очистити вагу до 0
- TareS : Прапорець для встановлення ваги тари

DLCWEI	
En	
Group	Weight
Module	Status
ChNo	Error
Stable	ErrCode
ZeroS	
TareS	
TareW	

TareW : значення маси тари

Weight : Поточне значення ваги

Status : Код стану модуля LC

Error : позначка помилки

ErrCode : Код помилки

Пояснення

- Ця інструкція доступна для ПЛК з FW V1.06.00 або пізнішої версії та ISPSOft V3.08 або пізнішої версії.
- Інструкція LCWEI підтримує лише модуль LC серії AS. Перш ніж використовувати інструкцію, ви повинні дізнатися про позицію конфігурації поточного модуля з HWCONFIG.
- Group** — це номер групи зазначеного модуля LC, підключеного праворуч від ПЛК або віддаленого модуля. Номер ПЛК - 0, номер першого віддаленого модуля - 1 і так далі. Максимальний номер групи становить 15. Якщо вказаний модуль не є модулем LC, прапорець **Error** зміниться на ON.
- Module** встановлює серійний номер модулів праворуч від ПЛК. Перший номер 1, другий номер 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Максимальна кількість – 32. Якщо вказаний модуль не є модулем LC, прапор помилки **Error** зміниться на ON.
- ChNo** – номер каналу зазначеного модуля LC. Якщо вхідне значення не є номером каналу модуля LC, прапор помилки **Error** зміниться на ON .
- Stable** встановлює значення в діапазоні стабільної ваги. Це необроблені дані. Значення може бути встановлено як значення з плаваючою комою
в діапазоні 0,0~100000,0, і максимальне або мінімальне значення буде автоматично прийнято як налаштування, якщо значення налаштування перевищує діапазон. Час, коли значення параметра стає дійсним, - коли інструкцію ввімкнено вперше. Якщо значення в модулі LC потрібно змінити, спочатку вимкніть інструкцію, установіть нове значення діапазону, а потім увімкніть інструкцію для нового налаштування.
- Після ввімкнення інструкції LCWEI вказаний канал автоматично переходить у режим відображення «маси нетто». Якщо вам потрібно знати вагу брутто (загальну вагу), додайте значення **TareW** і значення **ваги** самостійно.
- ZeroS** — це прапорець для встановлення поточної ваги на 0. Коли прапорець **ZeroS** змінюється з OFF на ON, значення **TareW** і **Вага** скидаються до 0.
- TareS** — це позначка для встановлення ваги тари. Коли **TareS** змінюється з OFF на ON, поточне значення **ваги** переміщується в **TareW** , а значення **ваги** скидається на 0. Коли **TareS** змінюється з ON на OFF, значення **TareW** повертається до поточного значення **ваги** , а значення **TareW** буде змінено. очищено до 0.
- Weight** – це значення ваги, виміряне шляхом вирахування ваги тари. Ви можете перевірити, чи існує значення **TareW** , щоб дізнатися, чи ввімкнено функцію ваги тари. Коли значення **TareW** дорівнює 0, це означає, що вагу тари не встановлено.

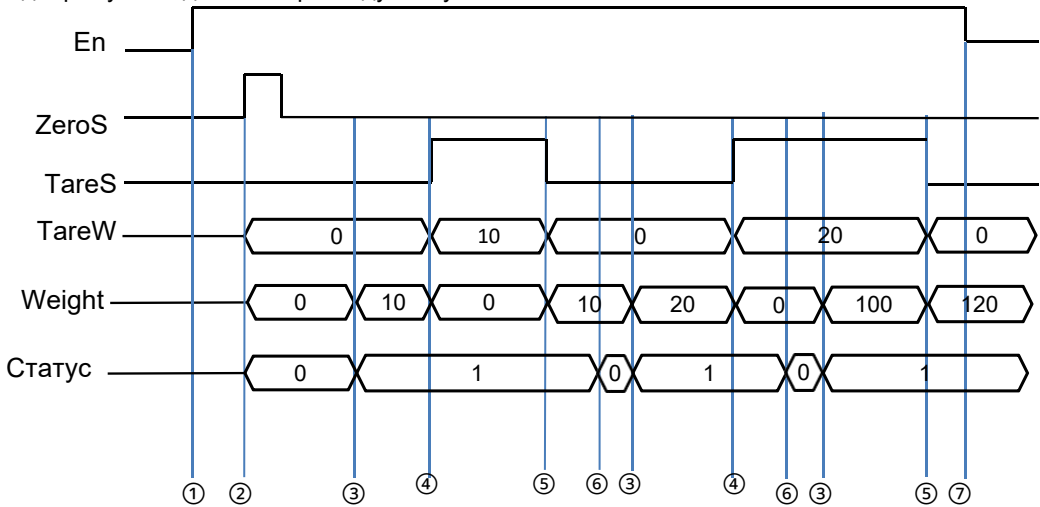
11. **Статус** — це код статусу, який зазвичай використовується для інструкцій щодо інтеграції модуля LC. Див. пояснення значень стану в наступній таблиці.

Значення	0	1	2	3	4	5
опис	Вимірювання ваги або без навантаження	Вага стабільна	Error обладнання/помилка калібрування	У калібруванні	Вага перевищує діапазон	Error номера модуля /помилка номера каналу

12. Під час зчитування ваги **«Статус»** відображатиме відповідний код помилки, а прапор **«Error»** змінюватиметься на **«УВІМК.»**.

виникає помилка в модулі LC. Коли стан повернеться до нормального стану, прапорець **Error** буде автоматично видалено. Докладніше про статус помилки див. у поясненні буферів контролю стану в посібнику з модуля LC.

13. Дивіться діаграму послідовності прикладу зчитування ваги.



Пояснення часових точок на наведеній вище діаграмі послідовності:

- ① → Увімкніть інструкцію LCWEI.
- ② → Після отримання команди очищення даних до 0, інструкція очистить значення **TareW** , **Weight** і **Status** .
- ③ → Покладіть вимірний матеріал на платформу для зважування. Коли вимірюється стабільне значення ваги, значення **Status** стає 1, а **Weight** відображає значення ваги.
- ④ → Якщо встановити **TareS** на ON, значення **Weight** переходить до **TareW** , а потім значення **Weight** очищається.
- ⑤ → Якщо встановити **TareS** на OFF, значення **TareW** повертається до **Weight** , а потім значення **TareW** очищається.
- ⑥ → Покладіть інший вимірний матеріал на платформу для зважування. На даний момент **Статус** переходить у статус вимірювання ваги.
- ⑦ → Вимкніть інструкцію LCWEI. **TareW** , **Weight** та **статус** містять останні значення статусу.

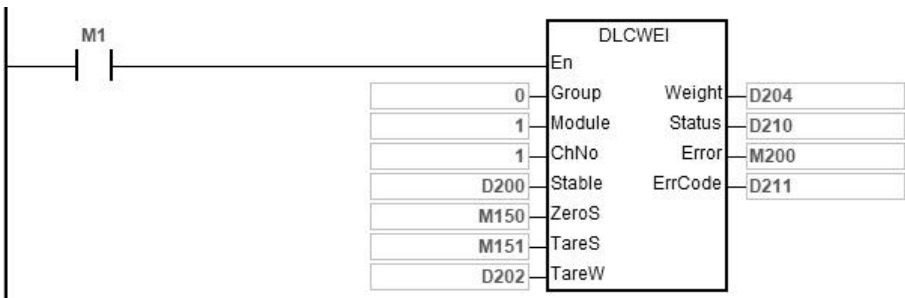
14. Коды помилок, які показує **ErrCode** , перераховані в наступній таблиці.

Код помилки	опис
16#1410	Error в номері групи LC або номері модуля
16#1411	Module LC не має такого номера каналу
16#1414	Error запису даних або час очікування зв'язку на модулі LC

приклад

Після завершення калібрування ви можете використовувати інструкцію DLCWEI для зважування.

- Зважування: Залиште гирю масою 500 г на платформі для зважування. Коли M1 перемикається в положення ON, відображається значення в D204
500,0.
- Встановіть вагу тари
 - Залишити упаковку (100 г) на платформі ваг. Значення в D204 показує 100,0.
 - Коли M151 перемикається в положення ON, значення **ваги** (D204) змінюється на значення **TareW** (D202).
Після переміщення очистіть значення ваги в D204.
 - D204 = 0,0 і D202 = 100,0. Налаштування ваги тари завершено.
- Очистіть вагу тари
 - Коли M151 вимикається, значення **TareW** (D202) повертається до значення **ваги** (D204). Після переміщення очистіть значення **TareW** у D202.
 - D202 = 0,0 і D204 = 100,0. Налаштування очищення ваги тари завершено.
- Вкажіть діапазон стабільної ваги (перевірка стабільності)
 - Перед виконанням цієї інструкції необхідно вказати діапазон стабільної ваги. Тут ми встановили D200 = 10,0.
 - Залиште вагу 500 г на платформі для зважування. Коли результат вимірювання становить 490 ~ 510 г, D210 =1 (Weight стабільна).



API		Інструкція			Операнд							опис					
1417		D	MPID		Group, модуль ~ Error							Алгоритм PID для модуля RTD/TC					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
GROUP								●					○	○			
MODULE								●					○	○			
CH								●					○	○			
UPDATE	●	●	●	●				●									
PID_RUN	●	●	●	●				●									
SV								●					○	○			
PID_MODE								●					○	○			
PID_MAN	●	●	●	●				●									
MOUT_AUTO	●	●	●	●				●									
AUTO_DBW								●					○	○			
Kc_Kp								●								○	
Ti_Ki								●								○	
Td_Kd								●								○	
Tf								●								○	
PID_EQ	●	●	●	●				●									
PID_DE	●	●	●	●				●									
PID_DIR	●	●	●	●				●									
ERR_DBW								●									
ALPHA								●					○	○			
BETA								●					○	○			
MOUT								●									
BIAS								●					○	○			
CYCLE								●					○	○			
MV								●									
PV								●									
I_MV								●									
ERROR								●									

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
----------	------	------	-------	-------	------	-----	------	------	------	-------	-----	-----	--------

Зверніться до наступних описів типів даних

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DMPID	
En	MV
GROUP	PV
MODULE	I_MV
CH	ERROR
UPDATE	
PID_RUN	
SV	
PID_MODE	
PID_MAN	
MOUT_AUTO	
AUTO_DBW	
Kc_Kp	
Ti_Ki	
Td_Kd	
Tf	
PID_EQ	
PID_DE	
PID_DIR	
ERR_DBW	
ALPHA	
BETA	
MOUT	
BIAS	
CYCLE	

GROUP	: Номер групи
MODULE	: Номер модуля
CH	: Номер каналу
UPDATE	: Оновити параметри PID
PID_RUN	: Увімкніть алгоритм PID
SV	: Цільове значення (SV)
PID_MODE	: Режим ПІД-регулювання
PID_MAN	: Автоматичний/ручний режим PID
MOUT_AUTO	: Значення виведення вручну/авто
AUTO_DBW	: Діапазон, в межах якого не відбувається автонастроювання працює
Kc_Kp	: Пропорційне посилення
Ti_Ki	: Інтегральний коефіцієнт (сек або 1/сек)
Td_Kd	: Коефіцієнт похідної (с)
Tf	: Похідна постійна часу дії (с)
PID_EQ	: Типи формул PID
PID_DE	: Розрахунок похідної похибки ПІД
PID_DIR	: PID прямого/зворотного напрямку (PID_DIR)
ERR_DBW	: Діапазон, у межах якого значення помилки вважається 0
ALPHA	: Компенсація початкового значення інтеграла обчислення (для розігріву)
BETA	: Компенсація початкового значення інтеграла конкрет (для охолодження)
MOUT	: Ручне вихідне значення (MOUT)
BIAS	: Передача вихідного значення
CYCLE	: Час вибірки (CYCLE)
MV	: Вихідне значення (MV)
PV	: Значення процесу (PV)
I_MV	: Накопичене інтегральне значення
ERROR	: Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із програмною програмою версії 1.06.00 або новішої та підтримує лише праві модулі. Для ПЛК із програмним забезпеченням версії 1.08.00 або пізнішим він також може підтримувати віддалені модулі правої сторони (AS00SCM-A з програмним забезпеченням версії 2.06 чи пізнішим).

2. Ця інструкція доступна для AS04RTD-A (V1.04 або пізнішої версії), AS06RTD-A (V1.00 або пізнішої версії), AS04TC-A (V1.04 або пізнішої версії) та AS08TC-A (V1.00 або пізнішої версії).

3. Налаштування параметрів наведено в наступній таблиці.

Операнд	Тип даних	функція	Діапазон налаштування	опис
GROUP	DWORD/DINT	Номер групи	Номер групи модулів RTD/TC який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перша підключена група модулів є групою номер 1, друга підключена група модулів є групою номер 2. Можна підключити до 15 груп модулів і порахували.	
MODULE	DWORD/DINT	Номер модуля	Номер підключеного модуля праворуч від ПЛК безпосередньо або підключеного до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль – це модуль номер 1, другий підключений модуль – це модуль номер 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулі повинні бути включені в підрахунок.	
CH	DWORD/DINT	Номер каналу	Номер каналу для алгоритму PID, наприклад, канал 1 є номером 1 для алгоритму PID, а канал 2 є номером 2 для алгоритму PID і так далі.	
UPDATE	BOOL	Прапори для оновлення параметрів, пов'язаних з PID	Увімкнувши цю інструкцію, ви можете введіть новий параметр і встановіть для прапора UPDATE значення TRUE, щоб оновити нове налаштування, і після оновлення прапорець автоматично перемикається на FALSE.	

Операнд	Тип даних	функція	Діапазон налаштування	опис
PID_RUN	BOOL	Включення алгоритму PID	Правда: використовуйте алгоритм PID. False: скинути вихідне значення (MV) до 0 і припинити використання алгоритму PID.	
SV	DWORD/DINT	Цільове значення	-32768~32767	Цільове значення
PID_MODE	DWORD/DINT	Режим ПІД-регулювання	0: Автоматичне керування Коли PID_MAN перемикається з True на False, викликає вихідне значення (MV) в автоматичному алгоритмі. 1: Автоматичне налаштування параметрів для контролю температури. Після завершення налаштування система перебуває в режимі автоматичного керування (PID_MODE встановлено на 0) і заповнить відповідні параметри (Kc_Kp, Ti_Ki, Td_Kd, Tf, APPHA та BETA) Примітка: коли режим встановлено на 1, автоматичне налаштування параметра, ви не можете встановити значення.	
PID_MAN	BOOL	Режим PID A/M	True: вручну Виведіть MV відповідно до MOUT. Цей параметр не діє, якщо PID_MODE встановлено на 1. False: Автоматично Виведіть MV відповідно до алгоритму ПІД.	
MOUT_AUTO	BOOL	Режим автоматичної зміни MOUT	Правда: автоматично Значення MOUT змінюється залежно від MV. False: Нормальний Значення MOUT не змінюється залежно від MV.	
AUTO_DBW	DWORD/DINT	Діапазон, у якому авто настройка не працює	0~32000	Автоматичне налаштування НЕ активовано, коли SV знаходиться в діапазоні $\pm$ ширина зони нечутливості

Операнд	Тип даних	функція	Діапазон налаштування	опис
Kc_Kp	REAL	Розрахований пропорційний коефіцієнт (Kc або Kp, відповідно до налаштувань у PID_EQ)	Діапазон позитивних чисел одинарної точності з плаваючою комою	Якщо коефіцієнт P менше 0, Kc_Kp дорівнює 0. Незалежно, якщо Kc_Kp дорівнює 0, це не контролюється P.
Ti_Ki	REAL	Інтегральний коефіцієнт (Ti або Ki, відповідно до налаштувань у PID_EQ)	Діапазон додатних чисел одинарної точності з плаваючою комою (одиниця: To = секунда; Ki = 1/сек)	Якщо розраховано коефіцієнт I менше 0, Ti_Ki дорівнює 0. Якщо Ti_Ki дорівнює 0, він не контролюється I.
Td_Kd	REAL	Коефіцієнт похідної (Td або Kd, відповідно до налаштувань у PID_EQ)	Діапазон додатних чисел із плаваючою комою одинарної точності (одиниця вимірювання: секунди)	Якщо розраховано коефіцієнт D менше 0, Td_Kd дорівнює 0. Якщо Ti_Ki дорівнює 0, він не контролюється D.
Tf	REAL	Похідна постійна часу дії	Діапазон додатних чисел із плаваючою комою одинарної точності (одиниця вимірювання: секунди)	Якщо похідна-постійна часу дії менша за 0, Tf дорівнює 0 і вона не контролюється постійною часу дії похідної (згладжування похідної).
PID_EQ	BOOL	Типи формул PID	TRUE: залежна формула FALSE: незалежна формула	

Операнд	Тип даних	функція	Діапазон налаштування	опис
PID_DE	BOOL	Розрахунок похідної похибки ПІД	ПРАВДА: використовуйте варіації в PV для обчислити контрольне значення похідної (Derivative of the PV). НЕПРАВИЛЬНО: використовуйте варіанти помилки (E) для обчислення контрольного значення похідної (похідної помилки).	
PID_DIR	BOOL	PID напрямом вперед/назад	ІСТИНА: дія вперед; охолодження (E=PV-SV) НЕПРАВДА: зворотна дія; нагрів (E=SV-PV)	
ERR_DBW	DWORD/DINT	Діапазон, у межах якого значення помилки вважається 0.	-32768~32767	Значення похибки (E) - це різниця між SV і PV. Якщо значення параметра ERR_DBW дорівнює 0, функція вимкнена; інакше ПЛК ЦП перевіряє, чи поточна помилка менша за абсолютне значення ERR_DBW, і перевіряє, чи поточна помилка відповідає умові перехресного статусу. Якщо поточна помилка менша за абсолютне значення ERR_DBW і відповідає умові перехресного статусу, поточна помилка зараховується як 0, і ПЛК застосовує алгоритм PID; інакше поточна помилка вноситься в алгоритм PID.

Операнд	Тип даних	функція	Діапазон налаштування	опис
ALPHA	DWORD/DINT	Інтегральне посилення значення ініціалізації (нагрівання)	0~100 Одиниця: 1%	Початкове значення компенсація інтегрального числення (за розігрів)
BETA	DWORD/DINT	Інтегральне посилення значення ініціалізації (охолодження)	0~100 Одиниця: 1%	Початкове значення компенсація інтегрального числення (для охолодження)
MOUT	REAL	MV	0~1000 (одиниця: 0,1%)	Коли встановлено PID Вручну (PID_MAN=True), значення MV виводиться як значення, встановлене вручну для MOUNT.
BIAS	REAL	Передача вихідного значення	-32768~32767	Вихідне значення прямої подачі, що використовується для прямої подачі ПІД.

Операнд	Тип даних	функція	Діапазон налаштування	опис
CYCLE	DWORD/DINT	Час вибірки (T_s)	1~1000 (одиниця: 100 мс)	Коли інструкція сканується, використовуйте алгоритм PID відповідно до часу вибірки та оновіть MV.
MV	REAL	Вихідне значення MV	0,0~100,0 (одиниця: 1%)	Вихідне значення MV
PV	DWORD/DINT/ REAL	Поточна вартість	Формат визначається в HWCONFIG.	
I_MV	REAL	Накопичене інтегральне значення	Накопичене інтегральне значення для посилення. Коли MV виходить за межі діапазону 0-100%, накопичене інтегральне значення в I_MV припиняє змінюватися.	
ERROR	DWORD/DINT	Коди помилок	16#0000 : виконується інструкція виконано і працює нормально 16#1400 : модуль, який ви використовуєте, не підтримує цю інструкцію. 16#1401 : помилка налаштування номера групи або номера модуля 16#1402 : модуль, який ви використовуєте, не відповідає, і зв'язок закінчився 16#1403 : помилка налаштування каналу 16#1404: функція PID триває під час спілкування без повторних налаштувань.	

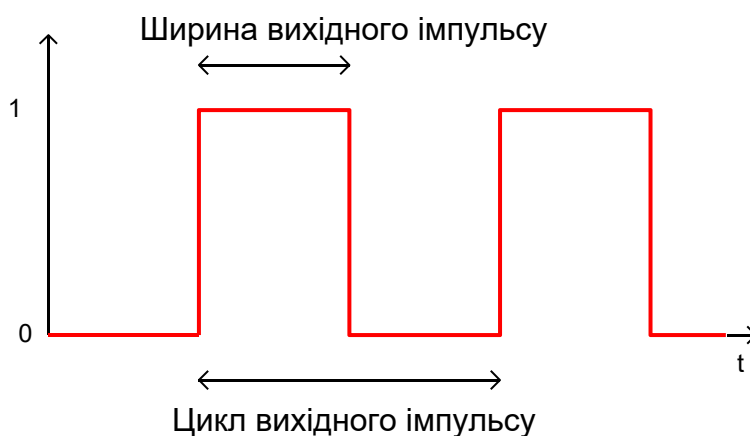
Примітка:

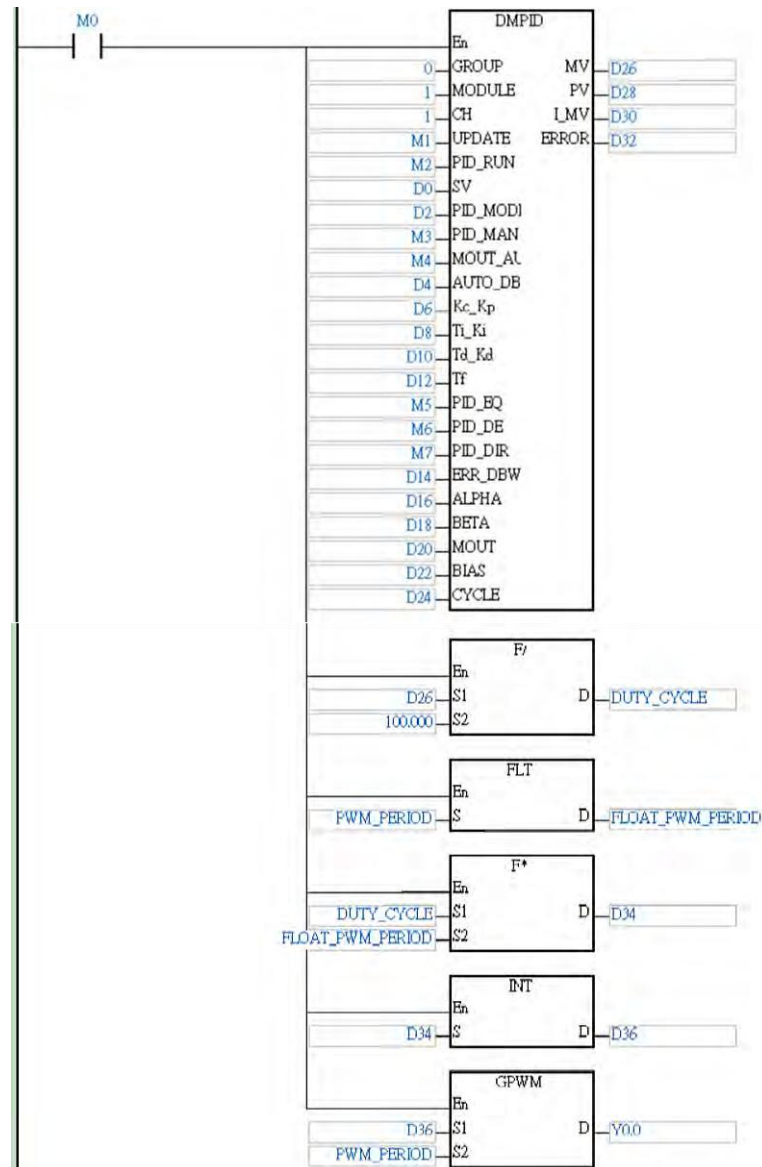
1. Якщо параметр PID перевищує верхню межу, у модуль можна записати лише максимальне значення, якщо параметр PID нижче нижньої межі, у модуль можна записати лише мінімальне значення.
2. Коли PID_RUN перемикається з True на False, він очищає вихідне значення MV до 0. Якщо вам потрібно зберегти останнє вихідне значення MV, ви можете змінити операнд EN на False, щоб закрити цю інструкцію та зберегти вихідне значення MV.

приклад

1. Вам потрібно налаштувати параметри перед виконанням DMPID. Перемикніть операнд EN з False на True, щоб виконати цю інструкцію, і коли інструкцію ввімкнено, параметри будуть застосовані до модулів. Якщо вам потрібно змінити параметри під час виконання, ви можете використовувати прапор UPDATE, щоб оновити параметри (включаючи **PID_RUN ~ CYCLE**) . Після оновлення параметрів ПЛК знімає прапор UPDATE.
2. Коли M0 увімкнено, інструкція виконується. Коли M2 увімкнено, DMPID починає оброблятися. Коли M2 вимкнено, значення MV дорівнює 0. А значення MV зберігається в D26. Коли M0 вимкнено, інструкція не виконується. І значення в інструкції не змінюються.
3. Коли режим встановлено на 1, система починає автоматично налаштовувати параметри для контролю температури. Після завершення налаштування система переходить у режим автоматичного керування (PID_MODE встановлено на 0) і заповнює відповідні параметри (Kc_Kp, Ti_Ki, Td_Kd, Tf, ALPHA та BETA) для пристроїв даних. Ви також можете використовувати пристрої збереження для збереження параметрів PID.
4. Вам слід конвертувати вихідне значення MV відповідно до ваших потреб у діапазоні від 0,0 до 100,0. У цьому прикладі вихідне значення MV перетворюється на вихід ШІМ. Використовуйте інструкцію поділу з плаваючою комою F/, щоб перетворити значення MV на робочий цикл в діапазоні від 0% до 100%, а потім використовуйте множення з плаваючою комою F*, щоб помножити робочий цикл і значення періоду ШІМ, щоб отримати ширину вихідного сигналу імпульсу.
5. Введення ширини та періоду вихідних імпульсів у параметри GPWM та керування ШІМ можна виконувати у призначеному пристрої вихідних імпульсів.

$$\text{Робочий цикл} = \frac{\text{Ширина вихідного імпульсу}}{\text{Цикл вихідного імпульсу}}$$



**Формула PID:**

1. Коли PID_MODE встановлено на 0, режим встановлюється на автоматичний:

- Незалежна формула та похідна від E (PID_EQ=False & PID_DE=False)

$$MV = K_p E + K_i \int_0^t E dt + K_d * \frac{dE}{dt} + BIAS \quad (E = SV - PV \text{ або } E = PV - SV)$$

- Незалежна формула та похідна PV (PID_EQ=False & PID_DE=True)

$$MV = K_p E + K_i \int_0^t E dt - K_d * \frac{dPV}{dt} + BIAS \quad (E = SV - PV)$$

Або

$$MV = K_p E + K_i \int_0^t E dt + K_d * \frac{dPV}{dt} + BIAS \quad (E = PV - SV)$$

- Залежна формула та похідна від E (PID_EQ=True & PID_DE=False)

$$MV = K_c \left[E + \frac{1}{T_i} \int_0^t E dt + T_d * \frac{dE}{dt} \right] + BIAS \quad (E = SV - PV \text{ або } E = PV - SV)$$

● Залежна формула та похідна PV (PID_EQ=Істинний & PID_DE=Істинний)

$$MV = K_c \left[E + \frac{1}{T_i} \int_0^m E dt - T_d * \frac{dE}{dt} \right] + BIAS (E = SV - PV)$$

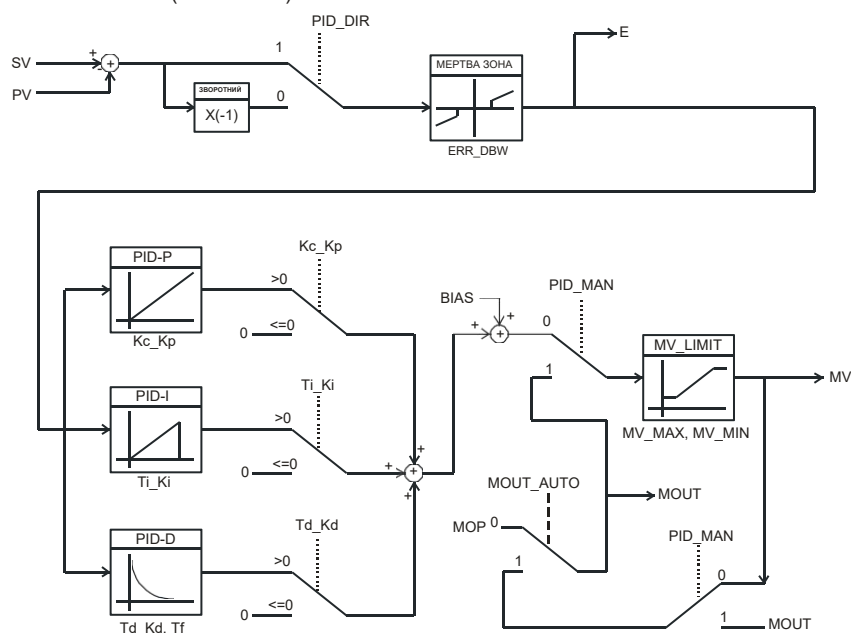
Або

$$MV = K_c \left[E + \frac{1}{T_i} \int_0^m E dt + T_d * \frac{dE}{dt} \right] + BIAS (E = PV - SV)$$

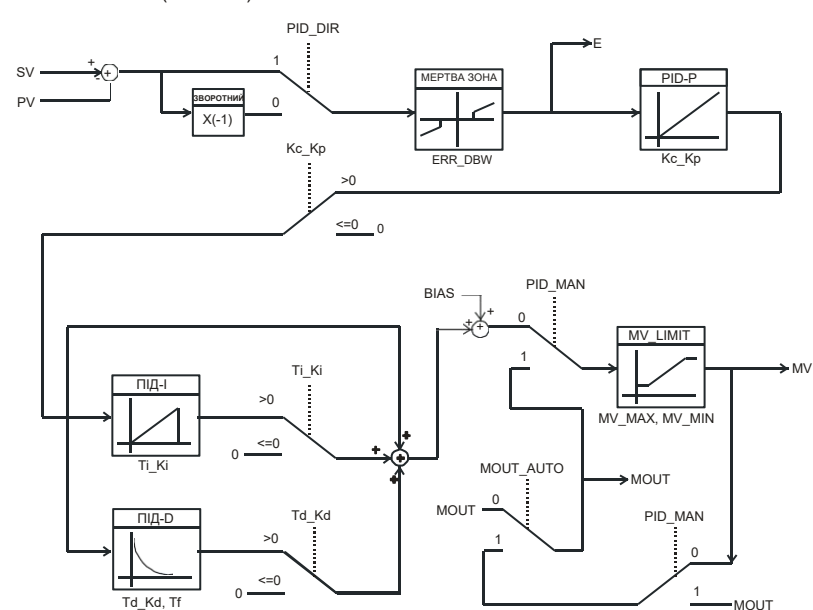
2. Коли ви встановлюєте PID_MODE на 1, вмикається режим автоматичного налаштування. Після завершення автоматичного налаштування значення стає 0 і режим автоматичного налаштування вимикається автоматично.

Блок-схема ПІД-контролю:

Блок-схема PID (незалежна)

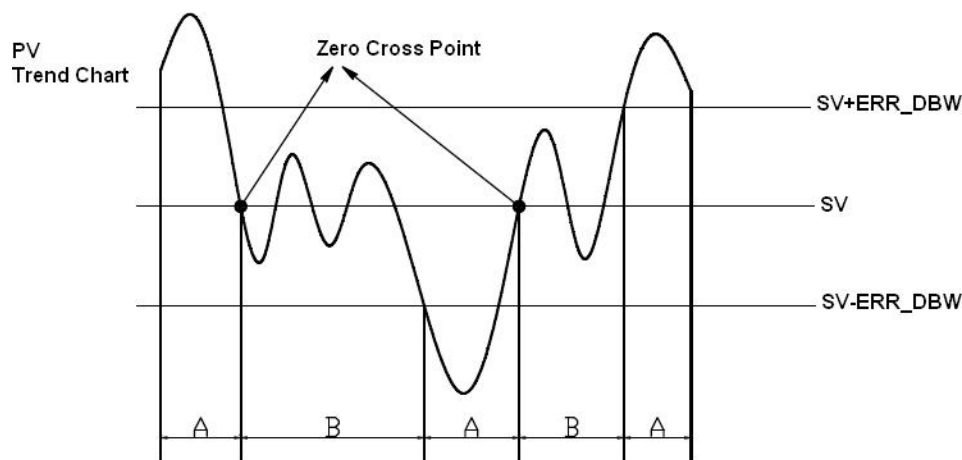


Блок-схема PID (залежна)

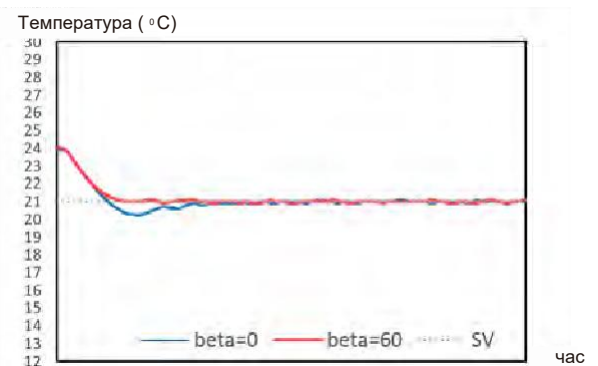
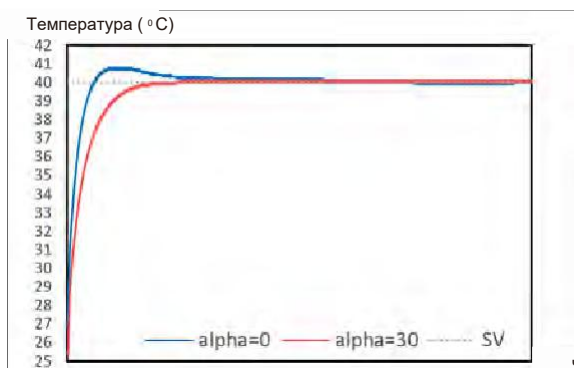


ERR_DBW

Коли PV (поточне значення) знаходиться в діапазоні **ERR_DBW**, на початку поточна помилка вводиться в алгоритм PID відповідно до нормальної обробки, а потім модуль CPU перевіряє, чи поточна помилка відповідає умові перехресного статусу: PV (теперішнє значення) перевищує SV (цільове значення). Після виконання умови поточна помилка вважається 0 під час застосування алгоритму PID. Після того як PV (поточне значення) виходить за межі діапазону **ERR_DBW**, поточна помилка знову вводиться в алгоритм PID. Якщо PID_DE має значення true, це означає, що він використовує варіації PV для обчислення контрольного значення похідної, і після виконання умови перехресного статусу ПЛК розглядає ΔPV як 0 для застосування алгоритму PID. (ΔPV = поточний PV – попередній PV). У наступному прикладі поточна помилка вводиться в алгоритм PID відповідно до звичайної обробки в розділі A, а поточна помилка або ΔPV вважається 0 для застосування алгоритму PID в розділі B.

**α, β Значення**

Щоб зменшити перевищення, ви можете використовувати параметри ALPHA або BETA на початку роботи ПІД-регулятора або коли SV (цільове значення) змінюється, щоб компенсувати початкове значення інтегрального обчислення (для нагрівання або охолодження). Дивіться зображення нижче. Використовуйте параметр ALPHA, щоб зменшити перевищення, коли температура підвищується. Використовуйте параметр BETA, щоб зменшити перевищення, коли температура падає.



Формула вихідного циклу:

Встановіть вихідний цикл відповідно до середовища. Якщо температура навколишнього середовища змінюється не настільки агресивно, ви можете встановити більший вихідний цикл.

- Ширина вихідного імпульсу = $MV (\%) \times \text{вихідний цикл}$

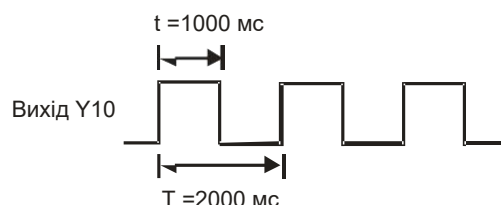
Виконайте загальний імпульс із інструкцією модуляції (GPWM), щоб встановити ширину вихідного імпульсу та час дискретизації вихідного циклу для керування циклом.

приклад

Якщо вихідний цикл становить 2000 мс, тоді вихідне значення становить 50% після реалізації алгоритму ПІД.

- Ширина вихідного імпульсу = $50\% \times 2000 \text{ мс} = 1000 \text{ мс}$

Іншими словами, інструкція GPWM може бути встановлена на ширину вихідного імпульсу = 1000 і вихідний цикл = 2000.



Примітка

1. Під час налаштування параметрів K_c , K_r , T_i , K_i і T_d , K_d (**PID_MODE** = 0) спочатку встановіть значення K_c , K_r , а потім значення T_i , K_i і T_d , K_d до 0. У контрольованому середовищі ви можете збільшити значення T_i , K_i (від меншого до більшого) і T_d , K_d (від більшого до меншого). Коли значення K_c , K_r дорівнює 1, пропорційний приріст становить 100%. Тобто значення похибок збільшуються в одиницю. Коли пропорційне посилення менше 100%, значення похибки зменшуються. Коли пропорційне посилення перевищує 100%, значення похибки збільшуються.
2. Параметри, які були автоматично налаштовані, не обов'язково підходять для кожного контрольованого середовища. Таким чином, ви можете додатково змінити автоматично налаштовані параметри, але рекомендується змінювати лише значення T_i , K_i або T_d , K_d .
3. Операнд CYCLE призначений для встановлення часу вибірки для використання алгоритму PID і оновлення MV.
4. Коли змінюється номер каналу для вимірювання, час оновлення виміряного значення також змінюється. Наприклад, виміряне значення оновлюється кожні 200 мс, якщо для вимірювання є лише 1 канал. Виміряне значення оновлюється кожні 800 мс, якщо є 4 канали для вимірювання. Параметри K_c , K_r , T_i , K_i , T_d , K_d можуть відрізнятися, якщо кількість каналів для вимірювання відрізняється.

API		Інструкція			Операнд							опис						
1418		D	HCCNT			Module ~ ErrCode							Лічильник модулів HC					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
Module								●					○	○				
ChNo								●					○	○				
Update		●	●	●				●										
Action								●					○	○				
Value							●	●					○	○				
CurCnt							●	●										
ST								●										
MT								●										
ASStat								●										
RefCnt								●										
Dir		●	●	●				●										
CntStat								●										
Error		●	●	●				●										
ErrCode								●										

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
Update	●												
Action		●				●							
Value			●				●						
CurCnt			●				●						
ST			●				●						
MT			●				●						
ASat		●				●							
RefCnt		●				●							
Dir	●												
CntStat		●				●							
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DHCCNT	
En	
Module	CurCnt
ChNo	ST
Update	MT
Action	AStat
Value	RefCnt
	Dir
	CntStat
	Error
	ErrCode

- Module** : Номер модуля
- ChNo** : Номер каналу
- Update** : Прапор оновлення параметрів
- Action** : Дії для дії лічильника
- Value** : Нове значення підрахунку / значення зміщення
- CurCnt** : Поточне значення лічильника
- ST** : Одноповоротні дані SSI-кодера абсолютного типу
- MT** : Багатооборотні дані SSI-кодера абсолютного типу
- AStat** : Дані про стан SSI-кодера абсолютного типу
- RefCnt** : Оновлення лічильника SSI-кодера абсолютного типу
- Dir** : Напрямок відліку
- CntStat** : Статус підрахунку
- Error** : Прапор помилки
- ErrCode** : Код помилки

Пояснення

- Ця інструкція стосується лише AS02HC-A. Ви можете запускати, зупиняти лічильник, а також налаштовувати та редагувати значення лічильника.
Ця інструкція доступна для ПЛК із програмним забезпеченням версії 1.08.або новішої та ISPSOft V3.10 чи новішої.
- Перед виконанням цієї інструкції вам потрібно вказати значення налаштувань у **Action** та **Value** . Після запуску **En** ПЛК негайно сканує значення в **діях** і **значеннях** . Якщо вам потрібно відредагувати значення в **«Action»** та **«Value»** , установіть нові значення в **«Action»** та **«Value»** та встановіть для **«Update»** значення «УВІМК.». Після оновлення значень у **«Action»** та **«Value»** параметр **«Update»** вимикається.
- Module** : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля
2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість.
- ChNo** : Номер каналу для лічильника, канал 1 є номером 1; канал 2 - номер 2.
- Режими та функції протидії**

Режим дії	функція	опис
0	Залишатися без змін	
1	Налаштувати/змінити поточне значення лічильника (CurCnt)	Налаштуйте нове значення лічильника в Value Примітка. Якщо ви використовуєте кодер SSI (послідовний синхронний інтерфейс) і для режиму лічильника встановлено значення Absolute Position, режим дії тут недейсний.
2	Встановіть значення зсуву в SSI-кодуювальнику як значення в Value	Якщо для режиму лічильника встановлено абсолютне положення, ви можете встановити значення зміщення лічильника кодера SSI. Якщо встановити значення зміщення у Value , поточне значення лічильника (CurCnt) є сумою

Режим дії	функція	опис
		значення лічильника кодера + значення у Value . Примітка: 1. Діапазон значень у значенні : $-2^{(MT+ST\ Length-1)} \leq \text{Value} < 2^{(MT+ST\ Length-1)}$. Якщо значення встановлено за межами діапазону, воно недійсне. 2. Якщо ви використовуєте кодувальник SSI (послідовний синхронний інтерфейс) і налаштування типу лічильника – Лічильник дзвінків, режим дії тут недійсний. 3. Після завантаження параметрів HWCONFIG значення зсуву скидається до 0.
3	Встановити/змінити абсолютне значення позиції датчика SSI	Коли для режиму лічильника встановлено значення Absolute Position, значення лічильника кодера SSI буде зміщено та збережено у Value . Якщо ви встановите значення зміщення у Value , поточне значення лічильника (CurCnt) — це значення у Value . (Поточний = значення) Примітка: 1. Діапазон значень у Value : $0 \leq \text{Value} < 2^{(MT+ST\ Length)}$. Якщо значення встановлено за межами діапазону, воно недійсне. 2. Якщо ви використовуєте кодувальник SSI (послідовний синхронний інтерфейс) і для типу лічильника встановлено значення «Лічильник дзвінків», режим дії тут недійсний. Ви можете використовувати Action=1 , щоб змінити значення налаштування. 3. Після завантаження параметрів HWCONFIG значення зсуву скидається до 0.
4	Скинути поточне значення лічильника (CurCnt)	1. Очистіть значення в CurCnt до 0. 2. Очистіть значення CurNo з інструкції DHCCMPT. 3. Зніміть прапорці Match1 і Match2 з інструкції DHCCMPT. Примітка. Якщо ви використовуєте кодер SSI (послідовний синхронний інтерфейс) і режим лічильника встановлено в абсолютну позицію, значення в CurCnt НЕ можна скинути до 0, але значення в CurNo , прапорці Match1 і Match2 з інструкції DHCCMP можна очистити. очищено.
5	Скинути поточне значення лічильника та призначені вихідні точки Y з інструкцій DHCCMP і DHCCMPT.	1. Очистіть значення в CurCnt до 0. 2. Очистіть значення CurNo з інструкції DHCCMPT. 3. Зніміть прапорці Match1 і Match2 з інструкції DHCCMPT. 4. Очистіть призначені вихідні точки Y (ON → OFF) з інструкцій DHCCMP і DHCCMPT.
6	Попередньо встановити поточне значення лічильника (CurCnt)	1. Установіть значення в CurCnt як значення у Value . 2. Очистіть значення CurNo з інструкції DHCCMPT. Після попереднього налаштування інструкція DHCCMPT виконує порівняння зверху. Якщо перше значення більше, ніж перше порівнюване значення в таблиці, порівняння припиняється. Якщо це станеться, ви можете використати прапор Оновлення (ON) з інструкції DHCCMPT, щоб виконати порівняння ще раз. 3. Зніміть прапорці Match1 і Match2 з інструкції DHCCMPT. Примітка. Якщо ви використовуєте кодер SSI (послідовний синхронний інтерфейс) і налаштування режиму лічильника встановлено в абсолютну позицію, значення в CurCnt НЕ можна змінити, але значення в CurNo , прапорці Match1 і Match2 з інструкції DHCCMP можна скинути.
7	Попередньо встановіть поточне значення лічильника та очистіть призначені вихідні точки Y з інструкцій DHCCMP і DHCCMPT.	1. Установіть значення в CurCnt як значення у Value . 2. Очистіть значення CurNo з інструкції DHCCMPT. Після попереднього налаштування інструкція DHCCMPT виконує порівняння зверху. Якщо перше значення більше, ніж

Режим дії	функція	опис
		перше значення для порівняння в таблиці, порівняння припиняється. Якщо це станеться, ви можете використати прапор Оновлення (ON) з інструкції DHCCMPT, щоб виконати порівняння ще раз. 3. Зніміть прапорці Match1 і Match2 з інструкції DHCCMPT. 4. Очистіть призначені вихідні точки Y (ON → OFF) з інструкцій DHCCMP і DHCCMPT.

6. **CurCnt** : поточне значення лічильника призначеного каналу модуля.

7. **ST** : одноповоротні дані SSI-кодера абсолютного типу. Якщо режим лічильника знаходиться в абсолютному положенні, значення **ST** є значенням зсуву. Для лічильника Ring значення **ST** є оригінальним показником SSI. Для інкрементального кодера значення **ST** дорівнює 0.

8. **MT** : Багатооборотні дані SSI-кодера абсолютного типу. Якщо режим лічильника знаходиться в абсолютному положенні, значення **MT** є значенням зміщення. Для лічильника Ring значення **MT** є оригінальним показником SSI. Для інкрементного кодера значення **MT** дорівнює 0.

9. **ASat** : Дані стану SSI-кодера абсолютного типу. Для інкрементного кодера значення **ASat** дорівнює 0.

10. **RefCnt** : оновлюваний лічильник SSI-кодера абсолютного типу в межах максимального діапазону 16 біт. Коли одні дані SSI захоплюються, додайте один до значення **RefCnt** . Після переповнення лічильник оновлення починає рахувати з 0. Для інкрементного кодера значення **RefCnt** дорівнює 0.

11. **Dir** : Напрямок підрахунку, коли значення ON, воно йде в позитивному напрямку; коли значення вимкнено, воно йде в негативному напрямку.

12. **CntStat** : статус підрахунку

Розрядне число	Статус	опис	Зауваження
15 ~ 11	Зарезервований		Зарезервований
10	Кодер абсолютного положення SSI відчуває перетин нуля	0: нормальний 1: ненормальний	У HWCONFIG ви можете налаштувати параметр виявлення чи не визначення перетину абсолютною позицією нульової точки. (за замовчуванням: виявлення вимкнено) Якщо режим лічильника знаходиться в абсолютному положенні, і ви встановили зміщення, виявлення починається після зсуву значення. Якщо типом лічильника є Ring Counter, значення зсуву буде скинуто до нуля. Виявлення починається з початкового положення SSI. Причина помилки: перехід через нуль Способи усунення помилки: скористайтеся інструкцією DHCCNT для скидання/попереднього налаштування або виконайте RESET за фазою Z.
9	SSI перевірка зв'язку	0: нормальний 1: ненормальний	Причина помилки: ненормальний зв'язок 5 разів поспіль Способи усунення помилки: зв'язок повертається до нормального
8	Перевірка парності SSI	0: нормальний 1: ненормальний	Причина помилки: помилка перевірки парності Способи усунення помилки: наступне проходження перевірки парності
7	Дані SSI перевищують встановлений допустимий діапазон	0: нормальний 1: ненормальний	Причина помилки: дані SSI перевищують встановлений допустимий діапазон Способи усунення помилки: коли дані SSI перестають перевищувати встановлений допустимий діапазон
6	Зарезервований	Зарезервований	

Розрядне число	Статус	опис	Зауваження
5	Кільцевий лічильник - Перелив	0: нормальний 1: ненормальний	У HWCONFIG ви можете встановити опцію виявляти або не виявляти. (за замовчуванням: виявлення вимкнено) Причина помилки: якщо переповнення ($> 2^{31} - 1$) або недоповнення ($< -2^{31}$)
4	Кільцевий лічильник - Underflow	0: нормальний 1: ненормальний	
3	Лінійний лічильник – значення перевищує верхню межу	0: нормальний 1: ненормальний	Причина помилки: значення лінійного лічильника перевищує встановлений користувачем діапазон Способи усунення помилки: коли значення лінійного лічильника знаходиться в діапазоні, визначеному користувачем
2	Лінійний лічильник – значення перевищує нижню межу	0: нормальний 1: ненормальний	
1	Лінійний лічильник - Перелив	0: нормальний 1: ненормальний	Примітка: коли значення підрахунку перевищує верхню або нижню межу, значення залишиться як значення у верхній/нижній межі. Після повернення значень до прийнятного діапазону підрахунок продовжується. Причина помилки: навіть якщо значення лінійного лічильника перевищує визначений користувачем діапазон, апаратний лічильник продовжує підрахунок. Але якщо підрахунок апаратного лічильника переповнений ($> 2^{31} - 1$) або недостатній ($< -2^{31}$), підрахунок припиняється. Після усунення помилки підрахунок відновлюється.
0	Лінійний лічильник - Підтікання	0: нормальний 1: ненормальний	

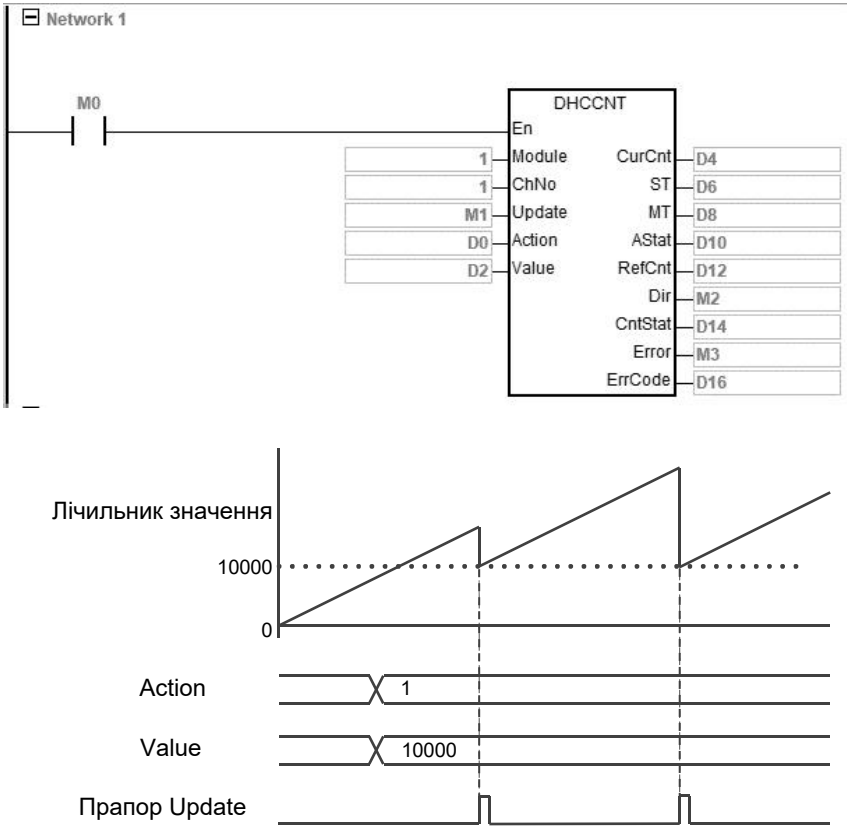
13. Якщо ви припинили виконання інструкції, значення каналу перестануть оновлюватися.

14. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів помилки увімкнено.

ErrCode	опис
16#1400	Ця функція НЕ доступна для цього модуля.
16#1401	Значення не дійсне або виходить за межі діапазону.
16#1402	Немає відповіді від модуля. Час очікування зв'язку.
16#1403	Error конфігурації HWCONFIG. Інструкція зупиняється.
16#1404	Для модуля НС такого каналу підрахунку немає.
16#1405	Інтерфейс введення не вибрано. Інструкція зупиняється.
16#1406	Недійсне значення в дії
16#1410	Один із каналів підрахунку модуля НС виконує підрахунок. Інші канали не можуть виконувати те саме завдання одночасно.

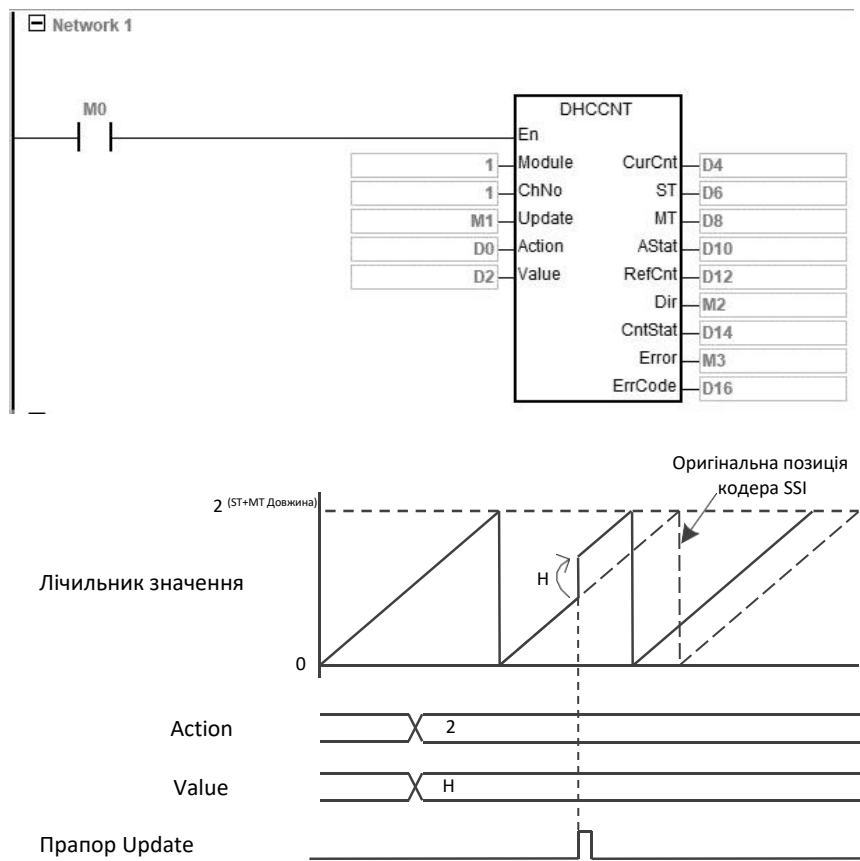
Приклад 1: Налаштування / зміна поточного налаштування (Action = 1)

- 1. Встановить M0=ON, лічильник починає рахувати.
- 2. У інструкції DHCCNT установить значення в Action на 1 і значення в Value на 10000. Коли прапор Update M1 увімкнено, значення в CurCnt оновлюється до 10000.
- 3. Після виконання прапор Update M1 автоматично вимикається.



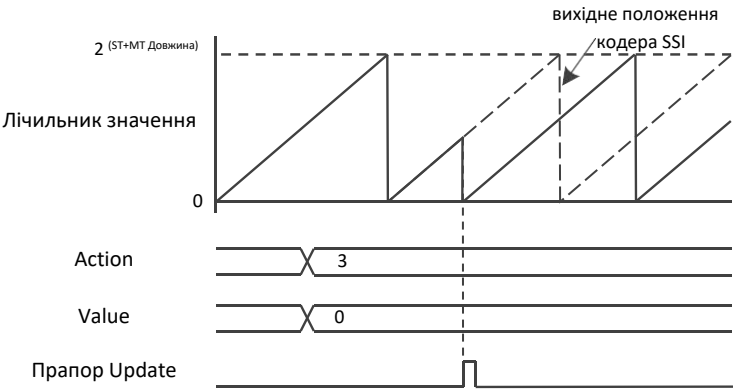
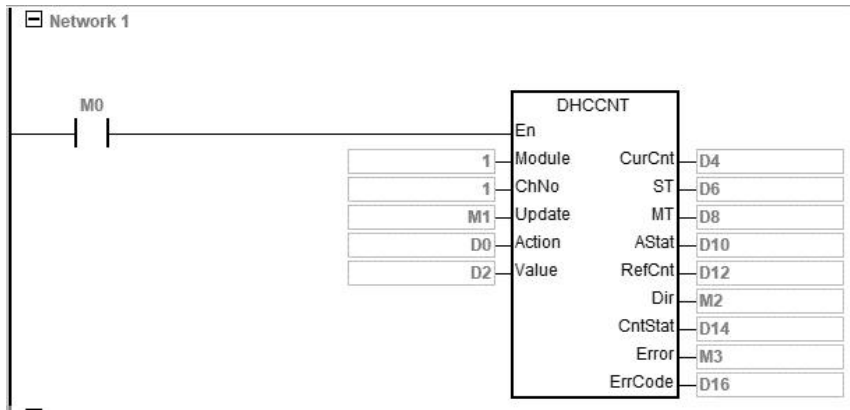
Приклад 2: установить значения зсуву в SSI-кодері як значения у **Value** (Action = 2)

- 1. У HWCONFIG встановить канал 1 як вхід SSI і встановить режим лічильника на абсолютне положення.
- 2. Встановить M0=ON, лічильник починає рахувати.
- 3. У інструкції DHCCNT встановить значення в **Action** на 2, а значення в **Value** на 500. Припустимо, що поточне значення підрахунку (**CurCnt**) становить 2500, і коли прапор **Update** M1 увімкнено, значення в **CurCnt** оновлюється до 3000.
- 4. Після виконання прапор **Update** M1 автоматично вимикається.



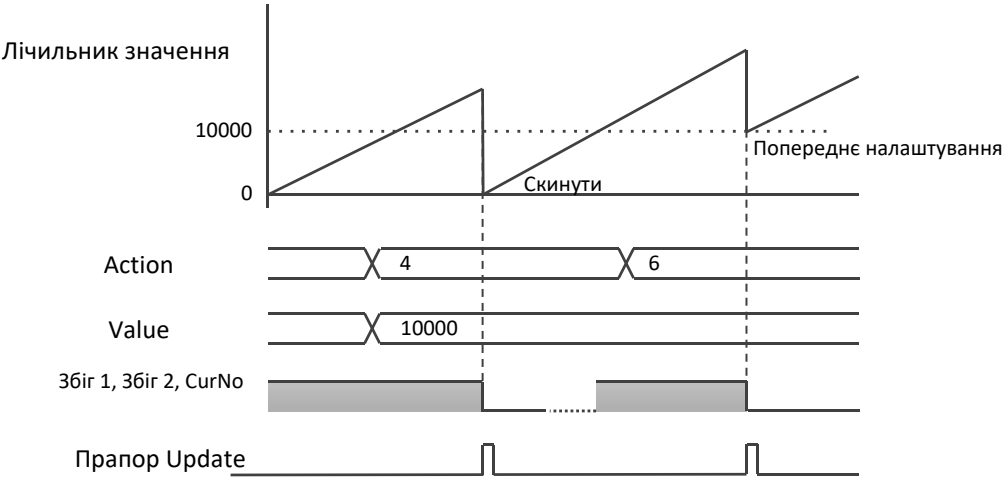
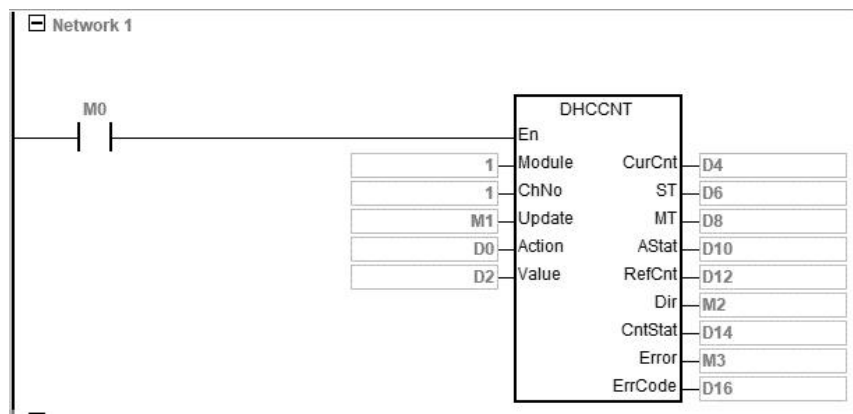
Приклад 3: встановити/змінити абсолютне значення позиції SSI-кодера (Action = 3)

- 1. У HWCONFIG встановити канал 1 як вхід SSI і встановити режим лічильника на абсолютне положення.
- 2. Встановити M0=ON, лічильник починає рахувати.
- 1. У інструкції DHCCNT встановити значення в **Action** на 3, а значення в **Value** на 0. Коли прапор **Update** M1 увімкнено, модуль HC автоматично обчислює значення зміщення, а значення в **CurCnt** зсувається на 0.
- 2. Після виконання прапор **Update** M1 автоматично вимикається.



Приклад 4: Скинути / попередньо встановити поточне значення лічильника (Action = 4, 6)

- 1. Встановить M0=ON, лічильник починає рахувати.
- 2. У інструкції DHCCNT встановить значення **Action** на 4. Коли прапор **Update** M1 увімкнено, значення **CurCnt** скидається до 0. Після виконання прапор **Update** M1 автоматично вимикається.
- 3. У інструкції DHCCNT встановить значення в **Action** на 6 і значення в **Value** на 10000. Коли прапор **Update** M1 увімкнено, значення в **CurCnt** попередньо встановлено на 10000. Після виконання прапор **Update** M1 автоматично вимикається.



API	Інструкція			Операнд								опис					
1419	D	HCCAP		Module ~ ErrCode								Зафіксовано значення лічильника модуля HC					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Module								●					○	○			
ChNo								●					○	○			
TrgSel								●					○	○			
Capt1							●	●									
Cmplt1		●	●	●				●									
Capt2							●	●									
Cmplt2		●	●	●				●									
Error		●	●	●				●									
ErrCode								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
TrgSel		●				●							
Capt1			●				●						
Cmplt1	●												
Capt2			●				●						
Cmplt2	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DHCCAP	
En	
Module	Capt1
ChNo	Cmplt1
TrgSel	Capt2
	Cmplt2
	Error
	ErrCode

- Module : Номер модуля
- ChNo : Номер каналу
- TrgSel : Вибір тригера
- Capt1 : Захоплення значення 1
- Cmplt1 : Змагання за значення 1
- Capt2 : Значення захоплення 2
- Cmplt2 : Змагання за значення 2
- Error : Прапор помилки
- ErrCode : Код помилки

Пояснення

1. Ця інструкція стосується лише AS02HC-A. Ви можете запускати, зупиняти лічильник, а також налаштовувати та редагувати значення лічильника. Ця інструкція доступна для ПЛК із програмним забезпеченням версії 1.08.або новішої та ISPSOft V3.10 чи новішої.

2. Використовуйте цю інструкцію з інструкцією DHCCNT. При виконанні інструкції DHCCNT починається підрахунок. І ви можете використовувати DHCCAP для захоплення значення лічильника. Коли DHCCNT зупинено, значення лічильника припиняє оновлюватися. І отримане значення залишається незмінним.

3. Перш ніж виконувати цю інструкцію, вам потрібно налаштувати **TrgSel**. Після активації **En ПЛК негайно сканує значення в TrgSel**. Якщо вам потрібно відредагувати значення в **TrgSel**, установіть нові значення в **TrgSel** і зупиніть інструкцію, а потім виконайте інструкцію знову.

4. Module : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість.

5. ChNo : Номер каналу для лічильника, канал 1 є номером 1; канал 2 - номер 2.

6. TrgSel : режими і функції запуску

TrgSel	функція	опис	
0	Запускається точками введення фази Z призначеного каналу	ChNo	Джерело тригера
		1	CH1 Z фаза
		2	CH2 Z фаза
		Щоб працювати з цим режимом тригера, потрібно встановити значення параметра для входу Z Phase призначеного каналу на Capture у HWCONFIG .	
1	Ініціюється інструкцією порівняння виводу іншого каналу (Збіг 1 / Збіг 2 інструкції DHCCMP)	Припустімо, що ви виконуєте інструкцію DHCCAP на каналі 2 і встановлюєте TrgSel = 1 і використовуєте канал 1 для виконання DHCCMP. Коли Match 1/ Match 2 OFF → ON на каналі 1, канал 2 запускається для захоплення поточного значення та зберігає його в Capt 1 або Capt 2. Примітка. Навіть якщо ви встановили значення параметра для входу Z Phase призначеного каналу на Capture у HWCONFIG, коли TrgSel = 1, ця функція захоплення зовнішньої точки введення не дійсна.	

7. Захопіть поточне значення лічильника та збережіть його в **Capt 1** або **Capt 2**. Коли **TrgSel = 0**, він фіксує значення лічильника

коли вхідна точка Z-фази спрацьовує по наростаючому фронту і зберігає значення в **Capt1**, і фіксує значення лічильника, коли вхідна точка Z-фази спрацьовує по спадаючому фронту, і зберігає значення в **Capt2**. Коли **TrgSel = 1**, він фіксує значення лічильника, коли **Match 1** вимкнено → ON в іншому каналі та зберігає отримане значення в **Capt1**, він фіксує значення лічильника, коли **Match 2** вимкнено → ON в іншому каналі та зберегти отримане значення в **Capt2**.

8. Прапори **Cmplt1** і **Cmplt2** є індикаторами, щоб побачити, чи завершено захоплення в **Capt1** і **Capt2** . Коли **Cmplt1/Cmplt2** OFF → ON , це означає, що значення в **Cmplt1/Cmplt2** оновлено. Ви можете зняти прапорець на ВИМК. Навіть якщо ви не вимкнули прапорці **Cmplt** , записані значення можна оновити.

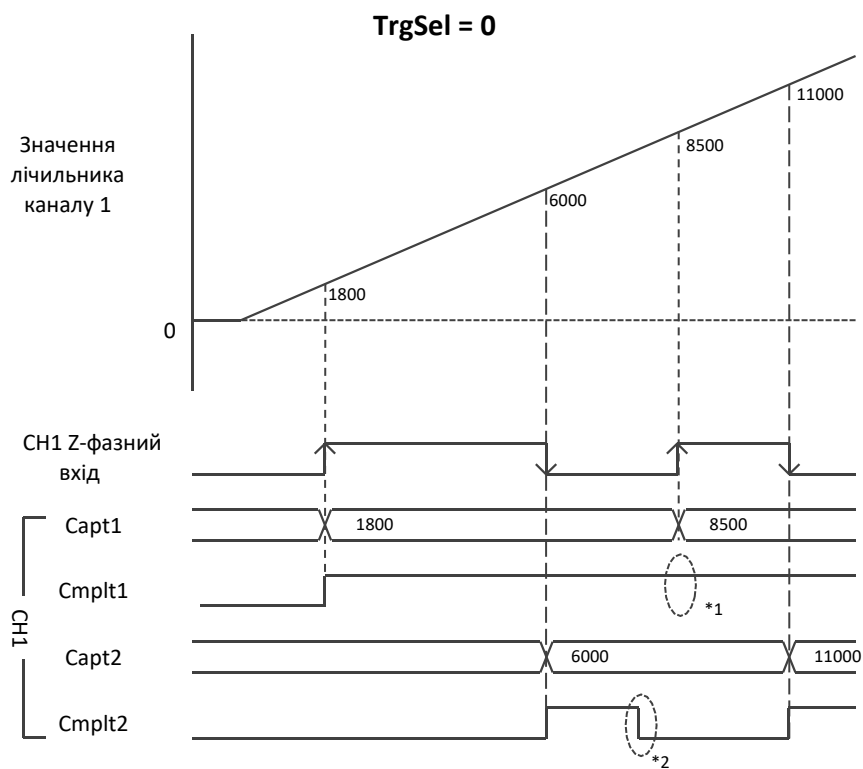
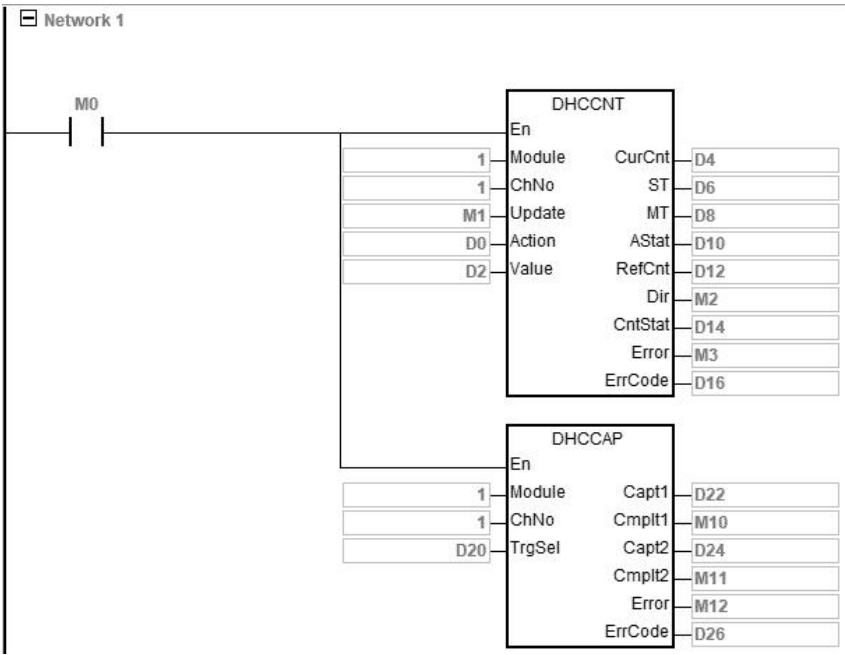
9. Якщо **En** ON → OFF, це означає, що функція захоплення вимкнена. Значення в **Capt1** і **Capt2** не будуть оновлені. Прапори **Cmplt1** і **Cmplt2** буде знято .

10. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів помилки увімкнено.

ErrCode	опис
16#1400	Ця функція НЕ доступна для цього модуля.
16#1401	Значення недійсне або виходить за межі діапазону.
16#1402	Немає відповіді від модуля. Час очікування зв'язку.
16#1403	Error конфігурації HWCONFIG. Інструкція зупиняється.
16#1404	Для модуля НС такого каналу підрахунку немає.
16#1405	Інтерфейс введення не вибрано. Інструкція зупиняється.
16#1407	Недійсне значення в TrgSel
16#1411	Один з рахункових каналів модуля НС виконує захоплення. Інші канали не можуть виконувати те саме завдання одночасно.

Приклад 1: Зафіксуйте значення лічильника, коли запускається точка введення фази Z.

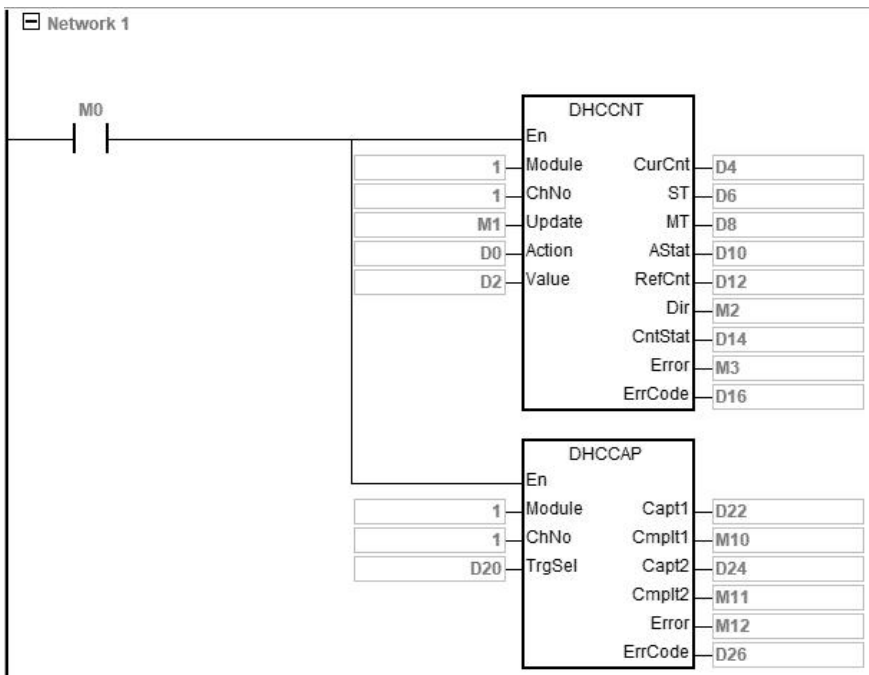
- Щоб працювати з цим режимом тригера, потрібно встановити значення параметра для фази Z каналу 1 для захоплення в HWCONFIG.
- Встановіть значення в **TrgSel** на 0 для інструкції DHCCAP. Коли M0 = ON, інструкція DHCCAP починає очікувати, поки вхідна точка фази Z буде активована по наростаючому або спадаючому фронту.
- Коли вхідна точка Z-фази каналу 1 запускається по висхідному фронту, вона фіксує значення лічильника 1800 і зберігає значення в **Capt1** . І прапор **Cmplt1** OFF → ON.
- Коли вхідна точка Z-фази каналу 1 запускається спадаючим фронтом, він фіксує значення лічильника 6000 і зберігає значення в **Capt2** . І прапор **Cmplt2** OFF → ON.
- Прапор **Cmplt** не вимкнено. Нове отримане значення все ще оновлюється в **Capt1** і **Capt2** . Тут ви можете побачити, що нове захоплене значення 8500 оновлюється в **Capt1** , коли вхідна точка Z-фази каналу 1 запускається спадаючим фронтом.
- Точка введення фази Z каналу 1 знову запускається за спадаючим фронтом, вона фіксує значення лічильника 11000 і зберігає значення в **Capt2** . І прапор **Cmplt2** OFF → ON.

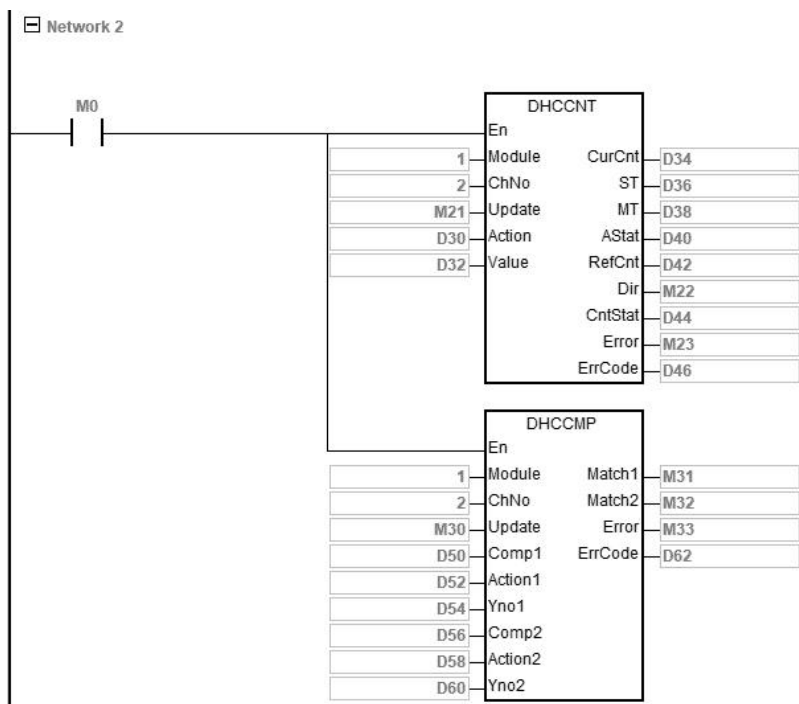


* 1: Cmpl1 не скидається на ВІМК., отримане значення все ще оновлюється в Capt1.
*2: для Cmpl2 встановлено значення OFF, отримане значення все ще оновлюється в Capt2.

Приклад 2: Зберіть значення лічильника, виконавши інструкцію DHCCMP і коли **Match 1/Match 2** вимкнено → ON в іншому каналі.

1. Припустімо, що ви виконуєте інструкцію DHCCAP на каналі 2 і встановлюєте **TrgSel** = 1 і використовуєте канал 1 для виконання DHCCMP. Коли **Match 1/ Match 2** OFF → ON на каналі 1, канал 2 запускається для захоплення поточного значення та зберігає його в **Capt 1** або **Capt 2** . Навіть якщо ви встановили значення налаштування для Z-фази призначеного каналу на **Capture** у HWCONFIG, коли **TrgSel** = 1, ця функція захоплення Z-фази вхідної точки недійсна.
2. Comp 1 (**Match 1**) OFF → ON на каналі 2, нове отримане значення 2000 на каналі 1 оновлюється в **Capt1** і **Cmplt 1** OFF → ON.
3. Comp 2 (**Match 2**) OFF → ON на каналі 2, нове отримане значення 8000 на каналі 1 оновлюється в **Capt2** і **Cmplt 2** OFF → ON.
4. Прапор **Cmplt** не вимкнено. Нове отримане значення все ще оновлюється в **Capt1** і **Capt2** .
5. Зовнішня вхідна точка каналу 1 знову запускається за спадаючим фронтом, вона фіксує значення лічильника 11000 і зберігає значення в **Capt2** . І прапор **Cmplt2** OFF → ON.





1

Module

Match1

M31

2

ChNo

Match2

M32

M30

Update

Error

M33

D50

Comp1

ErrCode

D62

D52

Action1

D54

Yno1

D56

Comp2

D58

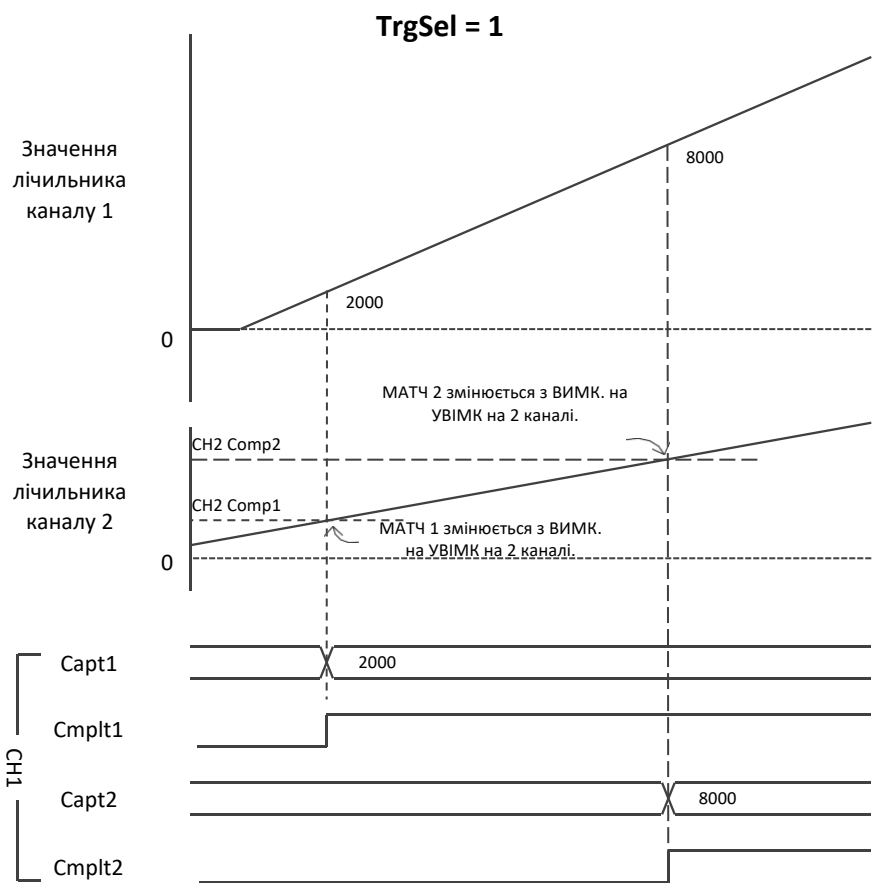
Action2

D60

Yno2

En

DHCCMP



API		Інструкція			Операнд						опис						
1420			HCDO			Module ~ ErrCode						Цифровий вихід модуля HC					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Module								●					○	○			
Update		●	●	●				●									
DOdata								●					○	○			
DOstat								●									
Error		●	●	●				●									
ErrCode								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
Update	●												
DOdata		●				●							
DOstat		●				●							
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ

HCDO	
En	
Module	Dostate
Update	Error
Dodata	ErrCode

- Module** : Номер модуля
- Update** : Номер каналу
- DOdata** : Цифрові вихідні дані
- DOstat** : Стан цифрового виходу
- Error** : Прапор помилки
- ErrCode** : Код помилки

Пояснення

- Ця інструкція стосується лише AS02HC-A. Ви можете запускати, зупиняти лічильник, а також налаштовувати та редагувати значення лічильника. Ця інструкція доступна для ПЛК із програмним забезпеченням версії 1.08.або новішої та ISPSoft V3.10 чи новішої.
- Перш ніж виконувати цю інструкцію, вам потрібно налаштувати **DOdata** . Після активації **En** ПЛК негайно сканує значення в **DOdata** . **DOdata** — початковий стан вихідних точок Y0.0 ~ Y0.3. Якщо вам потрібно змінити статус точки виведення, вам потрібно встановити нові значення в **DOdata** та встановити для параметра **Оновлення** значення ON. Після оновлення значень у **DOdata** параметр **«Оновлення»** вимикається.
- Module** : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість.

4. **DOdata** : налаштування для дії вихідних точок

b15~b4	b3	b2	b1	b0
NA	Action Y0.3	Action Y0.2	Action Y0.1	Action Y0.0
NA	0 : 3 1 : ON			

5. **DOstat** : налаштування стану вихідних точок

b15~b4	b3	b2	b1	b0
NA	Стан Y0.3	Стан Y0.2	Стан Y0.1	Стан Y0.0
NA	0 : ВИМК. 1 : УВИМК			

6. **DOstat** : якщо ви припинили виконання інструкції, **DOstat** припинить оновлення.

7. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів помилки увімкнено.

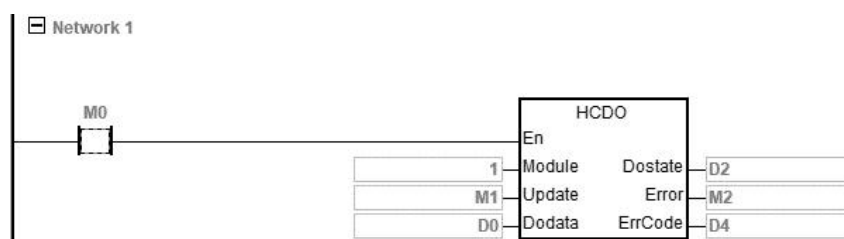
ErrCode	опис
16#1400	Ця функція НЕ доступна для цього модуля.
16#1401	Значення недійсне або виходить за межі діапазону.
16#1402	Немає відповіді від модуля. Час очікування зв'язку.
16#1403	Error конфігурації HWCONFIG. Інструкція зупиняється.
16#1411	Однією з точок виведення модуля HC є виконання виводу. Інші канали не працюють те саме завдання одночасно. (DOstat ще можна оновити.)

8. Коли виконуються інструкції **DHCCMP** або **DHCCMPT**, ви не можете змінити вихідні точки через **DOdata**, але **DOstat** може продовжувати оновлення.9. Щоб налаштувати початкові значення для вихідних точок **DHCCMP**, ви можете спочатку запустити інструкцію **HCDO**, а потім **DHCCMP**

або інструкція **DHCCMPT**. Якщо **вам** потрібно змінити вихідні точки, ви можете зупинити інструкцію **DHCCMP** або **DHCCMPT**, а потім використати **DOdata** **HCDO** для оновлення дій вихідних точок.

Приклад програмування: Встановіть зовнішню вихідну точку

- По-перше, визначте початковий стан Y0.0~Y0.3. У цьому прикладі стан Y0.0~Y0.3 встановлено на ON, тому ви повинні встановити вихідну точку **DOdata** інструкції **HCDO** на 15 перед перемиканням M0 на ON.
- Коли M0=ON, Y0.0~Y0.3 буде негайно змінено на ON.
- DOstat** показує поточний стан вихідної точки, яка тут дорівнює 15. Якщо M0=ON, значення **DOstat** буде оновлено в реальному часі, якщо стан Y0.0~Y0.3 зміниться в результаті керування з **DHCCMP** або **HCCMPT**.



API		Інструкція			Операнд							опис								
1421		D	HCCMP				Module ~ ErrCode							Значення лічильника модуля HC у порівнянні для виведення						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	I			
Module								●					○	○						
ChNo								●					○	○						
Update		●	●	●				●												
Comp1							●	●					○	○						
Action1								●					○	○						
Yno1								●					○	○						
Comp2							●	●					○	○						
Action2								●					○	○						
Yno2								●					○	○						
Match1		●	●	●				●									●			
Match2		●	●	●				●									●			
Error		●	●	●				●												
ErrCode								●												

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
Update	●												
Comp1			●				●						
Action1		●				●							
Yno1		●				●							
Comp2			●				●						
Action2		●				●							
Yno2		●				●							
Match1	●												
Match2	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ

DHCCMP	
En	
Module	Match1
ChNo	Match2
Update	Error
Comp1	ErrCode
Action1	
Yno1	
Comp2	
Action2	
Yno2	

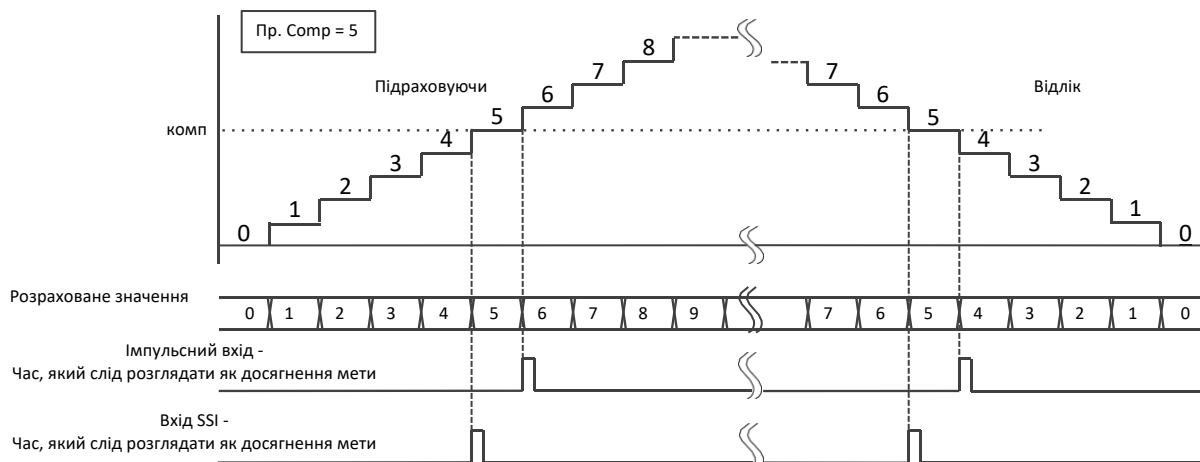
Module	: Номер модуля
ChNo	: Номер каналу
Update	: Прапор оновлення параметрів
Comp1	: Порівняльна вартість 1
Action1	: Дії, які потрібно виконати, коли Comp1 увімкнено
Yno1	: Кількість вихідних точок для виведення, коли Comp1 увімкнено
Comp2	: Порівняна вартість 2
Action2	: Дії, які потрібно виконати, коли Comp2 увімкнено
Yno2	: Кількість вихідних точок для виведення, коли Comp2 увімкнено
Match1	: Коли значення досягає встановленого значення 1, цей прапорець вмикається.
Match2	: Коли значення досягає встановленого значення 2, цей прапорець увімкнено.
Error	: Прапор помилки
ErrCode	: Код помилки

Пояснення

1. Ця інструкція стосується лише AS02HC-A. Значення лічильника шляхом підрахунку вгору або зворотного відліку досягло встановленого значення - **MATCH**. Якщо у вас є **MATCH**, система реагує відповідно до того, що ви встановили, включаючи дії лічильників і точку виведення для виведення. Ця інструкція доступна для ПЛК із програмним забезпеченням версії 1.08.або новішої та ISPSoft V3.10 чи новішої.
2. Використовуйте цю інструкцію з інструкцією **DHCCNT**. При виконанні інструкції **DHCCNT** починається підрахунок. Потім ви можете використовувати інструкцію **DHCCMP** для порівняння.
3. Перед виконанням цієї інструкції вам потрібно налаштувати **Comp1**, **Comp2**, **Action1**, **Action2**, **Yno1** та **Yno2**. Одного разу **En** активовано, ПЛК негайно сканує значення в **Comp1**, **Comp2**, **Action1**, **Action2**, **Yno1** і **Yno2**.
4. Ви можете редагувати значення в **Comp1**, **Comp2**, **Action1**, **Action2**, **Yno1** та **Yno2** під час роботи. Встановіть нові значення та встановіть для параметра **Оновлення значення** УВІМК. Після оновлення значень параметри **Update**, **Match1** і **Match2** перемикаються в положення OFF.
5. **Module**: номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість.
6. **ChNo**: Номер каналу для лічильника, канал 1 є номером 1; канал 2 - номер 2.
7. **Comp1** і **Comp2** є значеннями для порівняння (Значення в **Comp1** і **Comp2** мають бути різними.). Якщо значення лічильника **ChNo** відповідає встановленому значенню, його відповідний прапор **Match 1** / **Match2** буде увімкнено. Призначена вихідна точка Yno1 і Yno2 реагує відповідно до налаштувань для **Action**.

8. Якщо тип лічильника є лінійним, значення в **Comp1** і **Comp2** має задовольняти таку умову: *найменше значення лічильника* < **Comp** < *найбільше значення лічильника*. Якщо ви використовуєте кодер SSI (послідовний синхронний інтерфейс) і для режиму лічильника встановлено абсолютну позицію. Значення в **Comp1** і **Comp2** має задовольняти таку умову: $0 < \text{Comp} < 2^{(MT+ST\ Length)}$.

9. Час досягнення встановленого значення змінюється залежно від різних налаштувань вхідних інтерфейсів у HWCONFIG, наприклад, імпульсний вхід і вхід SSI.



Якщо вхідний інтерфейс має імпульсний вхід, час, який слід вважати таким, щоб відповідати встановленому значенню, передувє виходу підрахунку.

встановлене значення. Візьмемо, наприклад, $\text{Comp} = 5$, коли підрахунок йде від 5 до 6 або від 5 до 4, час тут вважається досягнутим встановленим значенням.

Якщо вхідним інтерфейсом є вхід SSI, час, який вважається досягненням встановленого значення, безпосередньо перед підрахунком торкається встановленого значення. Візьмемо, наприклад, $\text{Comp} = 5$, коли підрахунок йде від 4 до 5 або від 6 до 5, час тут вважається досягнутим встановленим значенням.

10. Якщо значення лічильника **ChNo** відповідає заданому значенню, його відповідний прапор **Match 1 / Match2** буде увімкнено. Призначений

вихідна точка **Yno1** і **Yno2** реагує відповідно до налаштувань для дій: **Action 1** (для призначеної вихідної точки **Yno1**) і **Action 2** (для призначеної вихідної точки **Yno2**).

Action 1	опис	Action 2	опис	Зауваження
0	Ніяких дій	0	Ніяких дій	
1	OFF	1	OFF	
2	ON	2	ON	
3	Перемикання між ON / OFF	3	Перемикання між ON / OFF	
4	ВИМК + скидання значення лічильника свого каналу	4	ВИМК + скидання значення лічильника свого каналу	Коли лічильник значення буде очищено, прапори відповідності 1 і відповідності 2 також будуть очищені.
5	ON + очистити значення лічильника свого каналу	5	ON + очистити значення лічильника свого каналу	
6	Перемикання між ON / OFF +	6	Перемикання між ON / OFF +	

	очистити значення лічильника канал		очистити значення лічильника канал	
7	Очистити значення лічильника канал	7	Очистити значення лічильника канал	
8	Очистити значення лічильника канал + ВІМК (призначені вихідні точки Yno1 і Yno2)	8	Очистити значення лічильника канал + ВІМК (призначені вихідні точки Yno1 і Yno2)	

Примітка: існує різниця в часі між досягненням встановленого значення та моментом виконання дії. До 100 нас часу можна очікувати різницю.

11. Установіть значення в **Yno1** і **Yno2**, щоб призначити вихідні точки Y0.0, Y0.1, Y0.2 і Y0.3. Ви можете мати два набори значень (**Yno1** і **Yno2**), щоб отримати два набори порівнянь. Після початку виконання цієї інструкції ви не зможете використовувати інструкцію HCDO для керування вихідними точками.

Yno1/Yno2	Призначені вихідні точки
0	Y0.0
1	Y0.1
2	Y0.2
3	Y0.3

12. Значення лічильника шляхом підрахунку вгору або зворотного відліку досягло встановленого значення - **MATCH**. Прапори **match 1** і

36ir 2 - індикатори. Коли значення лічильника відповідає встановленому значенню, його відповідний прапорець **Match 1** або **Match 2** буде увімкнено. Стається будь-яка з наступних подій: прапорці **36iry 1** і **36iry 2** будуть скинуті на ВІМК., коли ЦП ПЛК зупиняється, скидає лічильники, інструкція DHCCMP перемикається з ВІМК. на УВІМК. або прапор Update встановлюється на УВІМК.

13. Прапори **Match 1** і **Match 2** можуть використовуватися для пристроїв переривання I в діапазоні від I400 до I431.

Примітка. Частота пристрою переривання I з такою ж нумерацією не повинна перевищувати 800 Гц.

14. Якщо ви припинили виконання інструкції, її відносні порівнювані значення та її функція виводу перестануть оновлюватися.

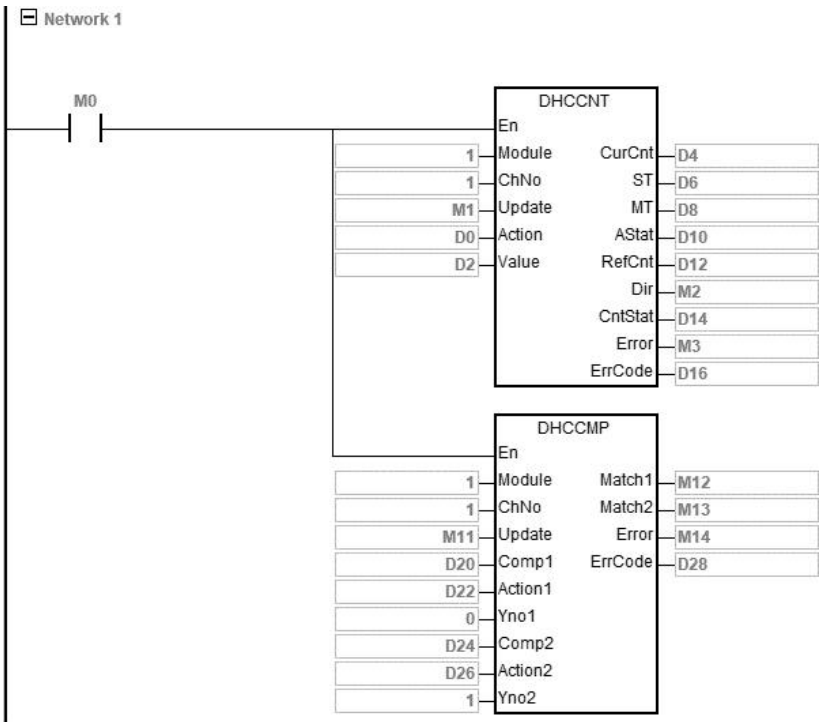
15. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів помилки увімкнено.

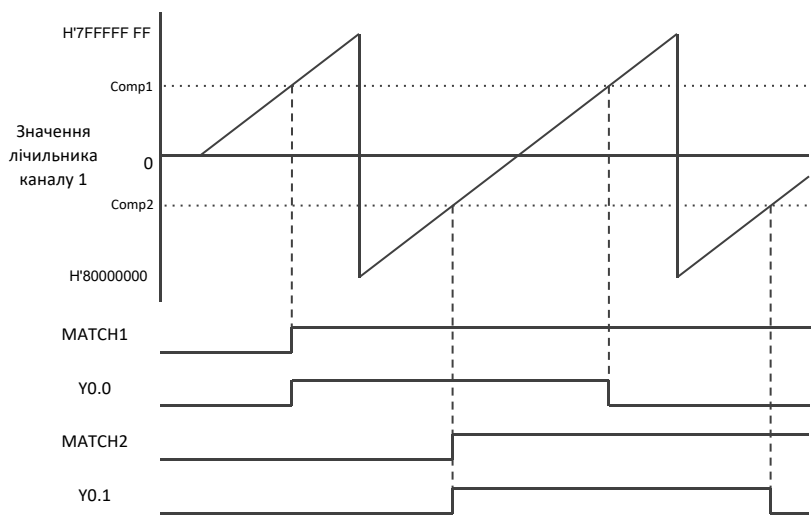
ErrCode	опис
16#1400	Ця функція НЕ доступна для цього модуля.
16#1401	Значення недійсне або виходить за межі діапазону.
16#1402	Немає відповіді від модуля. Час очікування зв'язу.
16#1403	Error конфігурації HWCONFIG. Інструкція зупиняється.
16#1404	Для модуля HC такого каналу підрахунку немає.

ErrCode	опис
16#1405	Інтерфейс введення не вибрано. Інструкція зупиняється.
16#1406	Недійсне значення в Action
16#1409	Недійсне значення в Yno
16#1413	Один з рахункових каналів модуля НС виконує вихід. Інші канали не можуть виконувати те саме завдання одночасно.

Приклад 1: Вихід із вихідної точки Y, коли значення лічильника досягло встановленого значення.

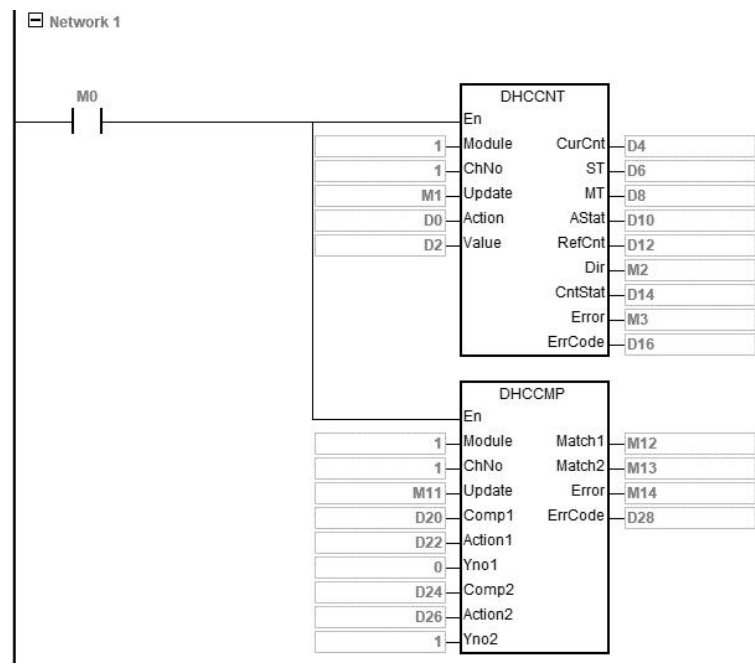
1. У HWCONFIG встановіть вхідний інтерфейс каналу 1 на імпульсний вхід або вхід SSI. І встановіть тип лічильника на лічильник дзвінків.
2. Встановіть значення в **Comp 1** на 16#5000000, значення в **Action1** на 3, а значення в **Yno1** на 0; встановіть значення **Comp 2** до 16#D000000, значення в **Action2** до 3, а значення в **Yno2** до 1.
3. Коли M0 = ON, лічильник починає підрахунок і встановлює параметри для інструкції DHCCMP, а потім чекає, поки значення лічильника досягне встановленого значення.
4. Коли значення лічильника досягає **Comp1**, прапорець **Match1** вмикається. **Y0.0 OFF → ON**.
5. Коли значення лічильника досягає **Comp2**, прапорець **Match2** вмикається. **Y0.1 OFF → ON**.
6. Коли значення лічильника знову досягає **Comp1**, **Y0.0** увімкнено → OFF. Коли значення лічильника досягне **Comp2** знову **Y0.1** увімкнено → OFF.

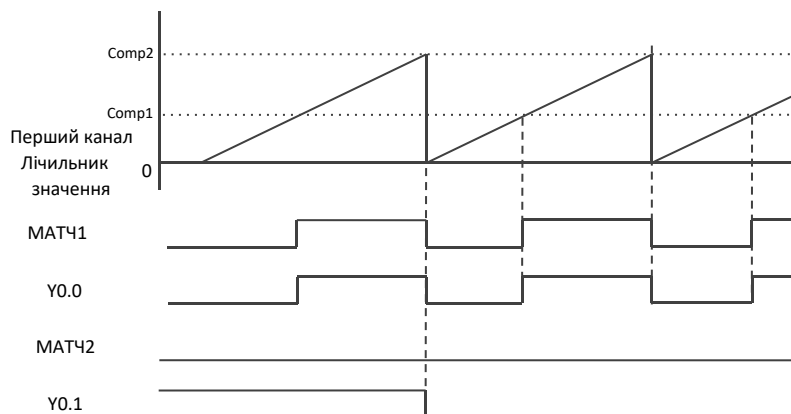




Приклад 2: Очистити значення лічильника його каналу, коли значення лічильника досягло встановленого значення.

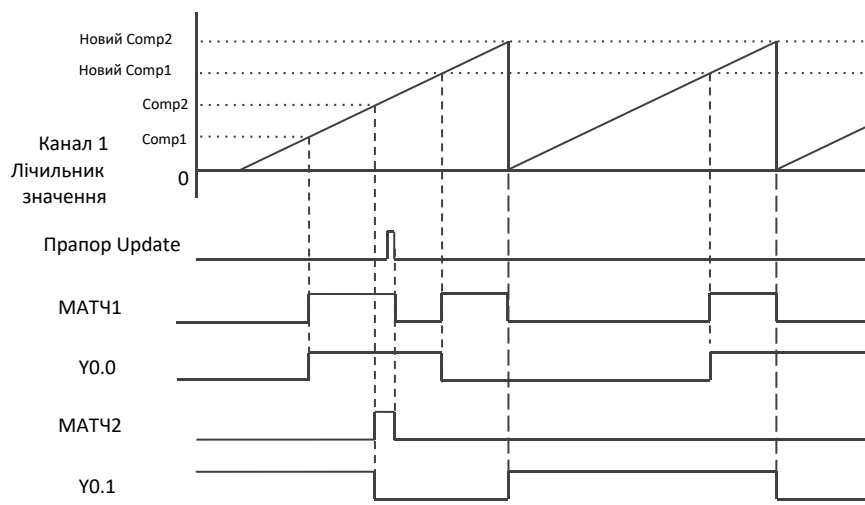
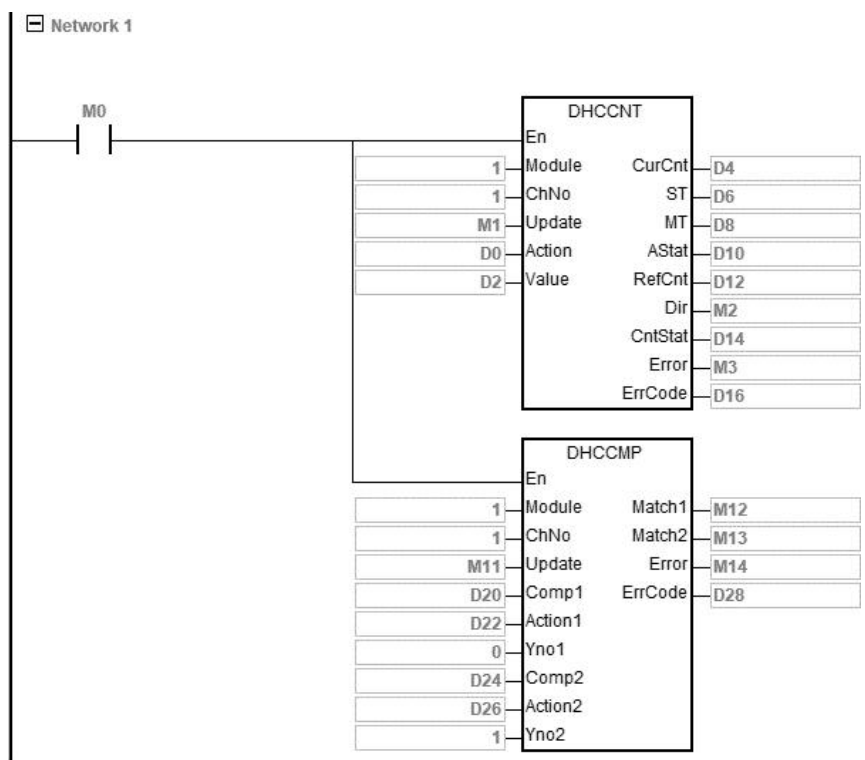
1. Встановіть значення в **Comp1** на 2500, значення в **Action1** на 2 і значення в **Yno1** на 0; встановіть значення в **Comp2** на 5000, значення в **Action2** на 8 і значення в **Yno2** на 1.
2. Коли M0 = ON, лічильник починає підрахунок і встановлює параметри для інструкції DHCCMP, а потім чекає, поки значення лічильника досягне встановленого значення.
3. Коли значення лічильника досягає **Comp 1** , прапорець **Match 1** вмикається. **Y0.0** OFF → ON.
4. Коли значення лічильника досягне **Comp 2** , почніть дію 2 = 8: очистіть значення лічильника його каналу, очистіть прапорці **Match 1 / Match 2** і **Y0.0 / Y0.1** на ВІМК.
5. Коли значення лічильника знову досягає **Comp1** , прапорець **Match1** вмикається. **Y0.0** OFF → ON.





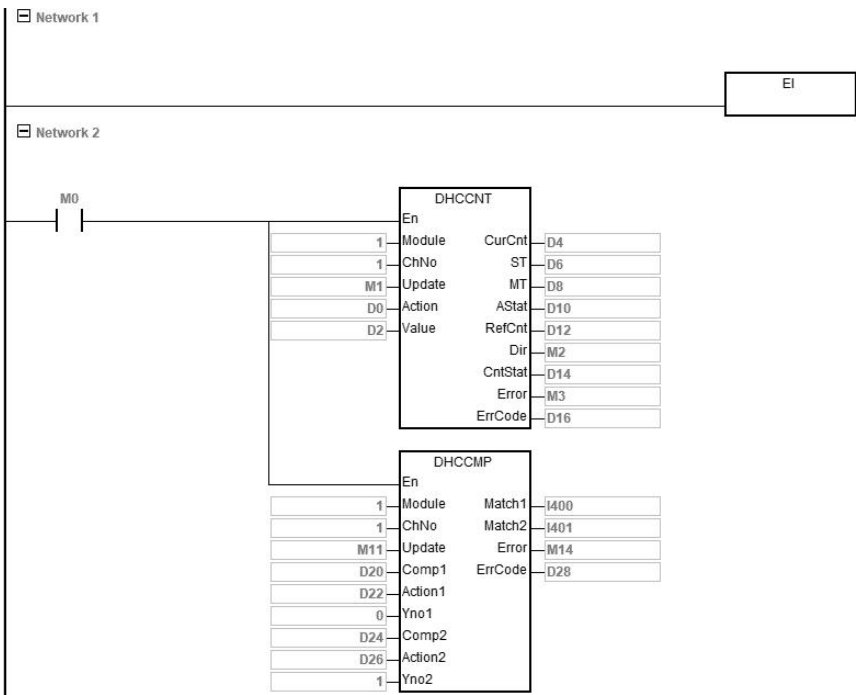
Приклад 3: оновіть значення лічильника свого каналу, коли значення лічильника досягне встановленого значення.

1. Встановіть значення в **Comp 1** на 5000, значення в **Action1** на 3 і значення в **Yno1** на 0; встановіть значення в **Comp 2** на 10000, значення в **Action2** на 3 і значення в **Yno2** на 1.
2. Коли M0 = ON, лічильник починає підрахунок і встановлює параметри для інструкції DHCCMP, а потім чекає, поки значення лічильника досягне встановленого значення.
3. Коли значення лічильника досягає **Comp1**, прапорець **Match1** вмикається. **Y0.0 OFF → ON**.
4. Коли значення лічильника досягає **Comp2**, прапорець **Match2** вмикається. **Y0.1 увімкнено → OFF**.
5. Встановіть нове значення для порівняння. Встановіть значення в **Comp 1** на 15000, значення в **Action1** на 3 і значення в **Yno1** до 0; встановіть значення в **Comp 2** на 20000, значення в **Action2** на 6 і значення в **Yno2** на 1.
6. Встановіть для прапора **Update** значення УВІМК. Після завершення оновлення прапорець автоматично скидається на ВІМК., і прапорці **Матчу 1 / Матчу 2** також будуть очищені.
7. Коли значення лічильника знову досягне **Comp1**. **Y0.0 ON → OFF**.
8. Коли значення лічильника знову досягне **Comp2**. **Y0.1 ON → OFF** та розпочніть дію 2 = 6: перемикайтеся між ON/OFF, очистіть значення лічильника його каналу, і прапорці Збігу 1 / Збігу 2 також будуть очищені.



Приклад 4: Виконайте переривання, коли значення лічильника досягло встановленого значення.

- 1. Для переривання можна використовувати інструкцію DHCCMP. Встановіть прапори **Match 1** і **Match 2** як переривання I400 і I401 відповідно.
- 2. Встановіть значення в **Comp1** на 5000, значення в **Action1** на 3 і значення в **Yno1** на 0; встановіть значення в **Comp2** на 10000, значення в **Action2** на 3 і значення в **Yno2** на 1.
- 3. Коли M0 = ON, лічильник починає підрахунок і встановлює параметри для інструкції DHCCMP, а потім чекає, поки значення лічильника досягне встановленого значення.
- 4. Коли значення лічильника досягає **Comp1**, виконується переривання I400 і починається дія 1 = 3: Y0.0 перемикається між ON / OFF.
- 5. Коли значення лічильника досягає **Comp2**, виконується переривання I401 і починається дія 2 = 3: Y0.1 перемикається між ON / OFF.



API	Інструкція				Операнд							опис					
1422	D	HCCMPT			Module ~ ErrCode							Значення лічильника модуля HC у груповому порівнянні для виведення					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Module								●					○	○		
ChNo								●					○	○		
Update		●	●	●				●								
CmpLen								●					○	○		
CompS							●	●								
ActionS								●								
YnoS								●								
InoS								●								
CurNo								●								
Error		●	●	●				●								
ErrCode								●								

пристрій	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Module		●				●							
ChNo		●				●							
Update	●												
CmpLen		●				●							
CompS			●				●						
ActionS		●				●							
YnoS		●				●							
InoS		●				●							
CurNo		●				●							
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ

DHCCMPT	
En	
Module	CurNo
ChNo	Error
Update	ErrCode
CmpLen	
CompS	
ActionS	
YnoS	
InoS	

Module	: Номер модуля
ChNo	: Номер каналу
Update	: Прапор оновлення параметрів
CmpLen	: Довжина даних для порівняння
CompS	: Порівняльне значення (32-розрядний)
ActionS	: Дії, які потрібно виконати, коли CompSis увімкнено
YnoS	: Кількість вихідних точок для виведення, коли CompSis увімкнено
InoS	: Кількість переривань, які будуть виконані, коли CompSis увімкнено
CurNo	: Номер поточної групи для порівняння
Error	: Прапор помилки
ErrCode	: Код помилки

Пояснення

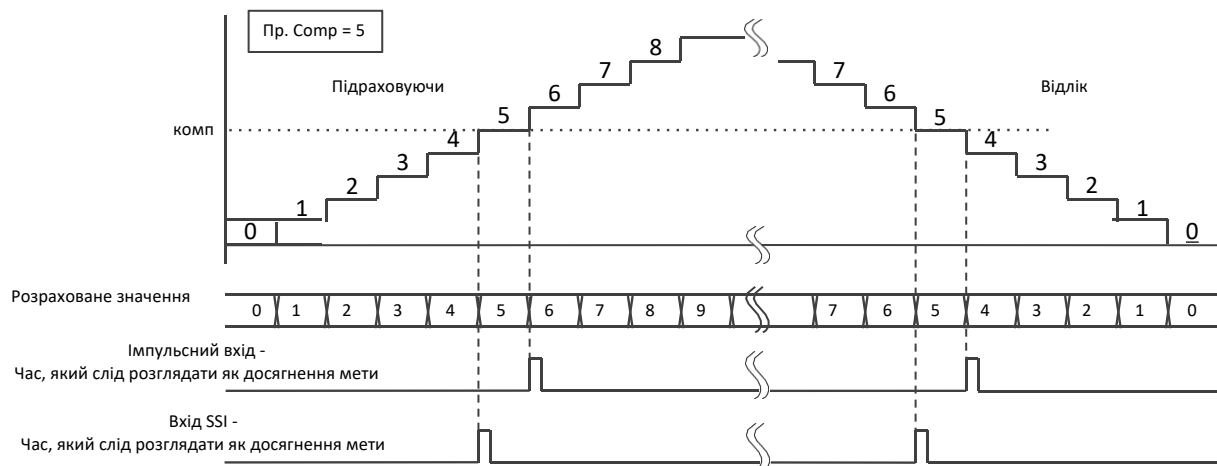
1. Ця інструкція стосується лише AS02HC-A. Він використовується для порівняння значення лічильника в циклічному порядку та одноразового значення збігається, виконується встановлена дія. Коли остання група завершена для порівняння, встановіть значення лічильника як перше порівнюване значення, і порівняння повернеться до першої групи.
2. Ця інструкція може працювати тільки в одному напрямку. Якщо змінити напрямок посередині виконання інструкції, результат може бути невірним. Можна встановити до 10 порівнюваних значень. Вам потрібно розташувати значення (мають бути неідентичними) у порядку зростання або спадання. У порядку зростання метод підрахунку є підрахунком угору, і використовувані значення мають бути додатними цілими числами. У порядку спадання метод підрахунку є зворотним відліком, і використовувані значення мають бути цілими від'ємними числами. Коли значення лічильника збігається з порівнюваним значенням, вихідні точки або лічильник реагують відповідно до ваших налаштувань.
3. Ця інструкція доступна для ПЛК із програмним забезпеченням версії 1.08.або новішої та ISPSoft V3.10 чи новішої.
4. Використовуйте цю інструкцію з інструкцією **DHCCNT** . При виконанні інструкції **DHCCNT** починається підрахунок. Потім ви можете використовувати інструкцію **DHCCMPT** для порівняння.
5. Перед виконанням цієї інструкції вам потрібно налаштувати **CmpLen** , **CompS** , **ActionS** , **YnoS** та **InoS** . Після активації **En** ПЛК негайно сканує значення в **CmpLen** , **CompS** , **ActionS** , **YnoS** та **InoS** .
6. Ви можете редагувати значення в **CmpLen** , **CompS** , **ActionS** , **YnoS** та **InoS** під час роботи. Встановіть нові значення та встановіть **Оновіть** на ON. Після оновлення значень прапорець **Оновлення** автоматично вимикається.
7. **Module** : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість.
8. **ChNo** : Номер каналу для лічильника, канал 1 є номером 1; канал 2 - номер 2.
9. **CmpLen** : довжина даних для порівняння в циклі; діапазон налаштувань 2~10. Коли значення налаштування виходить за межі діапазону, система не може виконати інструкцію, і з'являється код помилки.

10. **CompS** — це місце, де ви зберігаєте значення (у 32-розрядному цілочисельному форматі) для порівняння. Це займає кілька пробілів Dword у **CmpLen** послідовно. Вам потрібно розташувати значення (мають бути неідентичними) у порядку зростання або спадання. У порядку зростання метод підрахунку є підрахунком угору, і використовувані значення мають бути додатними цілими числами. У порядку спадання метод підрахунку є зворотним відліком, і використовувані значення мають бути цілими від'ємними числами. Для оголошення пропонується використовувати масив Dword.

Якщо тип лічильника є лінійним, значення в **CompS** має задовольняти наступну умову: *найменше значення лічильника* < **CompS** < *найбільше значення лічильника*.

Якщо ви використовуєте кодер SSI (послідовний синхронний інтерфейс) і для режиму лічильника встановлено абсолютну позицію. Значення в **CompS** має відповідати такій умові: $0 < \text{CompS} < 2^{(MT+ST \text{ Length})}$.

11. Час досягнення встановленого значення змінюється залежно від різних налаштувань вхідних інтерфейсів у HWCONFIG, наприклад, імпульсний вхід і вхід SSI.



Якщо вхідний інтерфейс має імпульсний вхід, час, який слід вважати таким, що відповідає встановленому значенню, передус підрахунку, що виходить з встановлене значення. Візьмемо, наприклад, **CompS = 5**, коли **підрахунок йде від 5 до 6 або від 5 до 4, час тут вважається досягнутим встановленим значенням**.

Якщо вхідним інтерфейсом є вхід SSI, час, який вважається досягненням встановленого значення, безпосередньо перед підрахунком торкається встановленого значення. Візьмемо, наприклад, **Comp = 5**, коли **підрахунок йде від 4 до 5 або від 6 до 5, час тут вважається досягнутим встановленим значенням**.

12. Якщо значення лічильника **ChNo** відповідає встановленому значенню, призначена вихідна точка **YnoS reaguє** відповідно до налаштувань для **ActionS**. Значення в **ActionS** має бути 16-бітним цілим числом і займати кілька місць Word у **CmpLen** послідовно.

Action	опис	Зауваження
0	Ніяких дій	
1	OFF	
2	ON	
3	Перемикання між ON / OFF	
4	OFF + очистити значення лічильника свого каналу	Коли значення лічильника скидається, значення в CurNo також обнулюється.
5	ON + очистити значення лічильника свого каналу	
6	Перемикання між ON / OFF + очищення значення лічильника свого каналу	
7	Очистити значення лічильника його каналу	
8	Очистити значення лічильника його каналу + ВІМК (призначено всі вихідні точки обома YnoS)	

Примітка: існує різниця в часі між досягненням встановленого значення та моментом виконання дії. До 100 нас часу можна очікувати різницю.

13. Значення в **YnoS** має бути 16-бітним цілим числом і займати кілька місць Word у **CmpLen** послідовно.

YnoS	Призначені вихідні точки
0	Y0.0
1	Y0.1
2	Y0.2
3	Y0.3

14. Значення в **InoS** - це номери переривань. Ви можете надсилати певні переривання до ЦП ПЛК. Допустимі переривання коливається від I400 до I431. Значення в **InoS** має бути 16-бітним цілим числом і займе кілька місць Word у **CmpLen** послідовно.

InoS	Дія
0	Без переривання
4 □ □	□ □ = 00~31; що відповідає I400~I431 (Коли задане значення досягнуто, ЦП ПЛК запитує конкретне переривання.)

Примітка. Частота пристрою переривання I з такою ж нумерацією не повинна перевищувати 800 Гц.

15. **CurNo** — номер поточної групи. Перегляньте таблицю нижче для довідки. Якщо значення лічильника дорівнює 200, підраховане число менше за значення в D100, а відповідний **CurNo** дорівнює 0. Якщо значення лічильника дорівнює 1000, перевірте таблицю нижче, щоб дізнатися, що значення в D100 також дорівнює 1000, а відповідний **CurNo** дорівнює 1.

16. **Comps** (припустимо D100), **ActionS** (припустимо D200), **YnoS** (припустимо D300), а значення в **CmpLen** дорівнює 6.

CurNo	CompS Вихідне значення для порівняння (32-розрядний)	ДіяS Взяти, коли значення досягнуто	YnoS (номер вихідної точки)	InoS (номер переривання)
1	D100 = 1000	D200 = 2 (увімк.)	D300 = 0 (Y0,0)	D400 = 400 (I400)
2	D102 = 2000	D201 = 2 (увімк.)	D301 = 1 (Y0.1)	D401 = 401 (I401)
3	D104 = 3000	D202 = 3 (перемикач)	D302 = 0 (Y0,0)	D402 = 402 (I402)
4	D106 = 4500	D203 = 2 (увімк.)	D303 = 2 (Y0,2)	D403 = 403 (I403)
5	D108 = 5500	D204 = 1 (Вимк.)	D304 = 3 (Y0,3)	D404 = 404 (I404)
6	D110 = 6500	D205 = 1 (Вимк.)	D305 = 1 (Y0,1)	D405 = 405 (I405)

17. Виконання інструкції (**En OFF** → УВІМК.) і встановить для **параметра Оновлення** значення УВІМК. Після оновлення значень інструкція використовує оновлені значення для порівняння. Для порівняння встановлюється поточне число, менше відповідного значення в **Comps**. Наприклад, якщо під час виконання інструкції поточне значення дорівнює 3500, відповідний **CurNo** дорівнює 3, а для порівняння встановлюється число 1~3 (значення лічильника 3000 < поточне значення 3500). Набір дій: Y0.0 = ON + запит на переривання I400, Y0.1 = ON + запит на переривання I401, Y0.0 = перемикач між ON і OFF) + запит на переривання I402, а значення в **CurNo** дорівнює 3.

18. Якщо **En ON** → OFF, це вказує на те, що функцію виведення порівняння вимкнено. Стани **CurNo** та **Output** припиняють оновлення. Лічильник продовжує рахувати, але порівняння не проводиться.

19. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів Error увімкнено.

ErrCode	опис
16#1400	Ця функція НЕ доступна для цього модуля.
16#1401	Значення не дійсне або виходить за межі діапазону.
16#1402	Немає відповіді від модуля. Час очікування зв'язку.
16#1403	Error конфігурації HWCONFIG. Інструкція зупиняється.
16#1404	Для модуля HC такого каналу підрахунку немає.
16#1405	Інтерфейс введення не вибрано. Інструкція зупиняється.
16#1406	Недійсне значення в дії
16#1408	Неправильний порядок у значенні CompS
16#1409	Недійсне значення в Yno
16#140C	Недійсне значення в CmpLen
16#1413	Один з рахункових каналів модуля HC виконує вихід. Інші канали не працюють те саме завдання одночасно.
16#1416	Недійсне значення в InoS

приклад

1. Налаштуйте наступні параметри.

CurNo	CompS Вихідне значення для порівняння (32-розрядний)	ДіяS Взяти, коли значення досягнуто	YnoS (номер вихідної точки)	InoS (номер переривання)
1	D50 = 1000	D70 = 3 (перемикач)	D80 = 0 (Y0,0)	D90 = 400
2	D52 = 2000	D71 = 3 (перемикач)	D81 = 1 (Y0,1)	D91 = 401
3	D54 = 3000	D72 = 3 (перемикач)	D82 = 2 (Y0,2)	D92 = 402
4	D56 = 4000	D73 = 3 (перемикач)	D83 = 3 (Y0,3)	D93 = 403
5	D58 = 5000	D74 = 3 (перемикач)	D84 = 0 (Y0,0)	D94 = 404
6	D60 = 6000	D75 = 3 (перемикач)	D85 = 1 (Y0,1)	D95 = 405
7	D62 = 7000	D76 = 3 (перемикач)	D86 = 2 (Y0,2)	D96 = 406
8	D64 = 8000	D77 = 3 (перемикач)	D87 = 3 (Y0,3)	D97 = 407
9	D66 = 9000	D78 = 3 (перемикання)	D88 = 0 (Y0,0)	D98 = 408
10	D68 = 10000	D79 = 3 (перемикач)	D89 = 1 (Y0,1)	D99 = 409

2. Виконання інструкції (**En OFF** → УВІМК.) і встановить для параметра **Оновлення** значення УВІМК. Після оновлення значень інструкція використовує оновлені значення для порівняння. Поточне підраховане число менше відповідного значення в **Comps** встановлено для порівняння. Значення початкового струму менше 1000, відповідний **CurNo** дорівнює 0, і жодних дій не виконується.

6

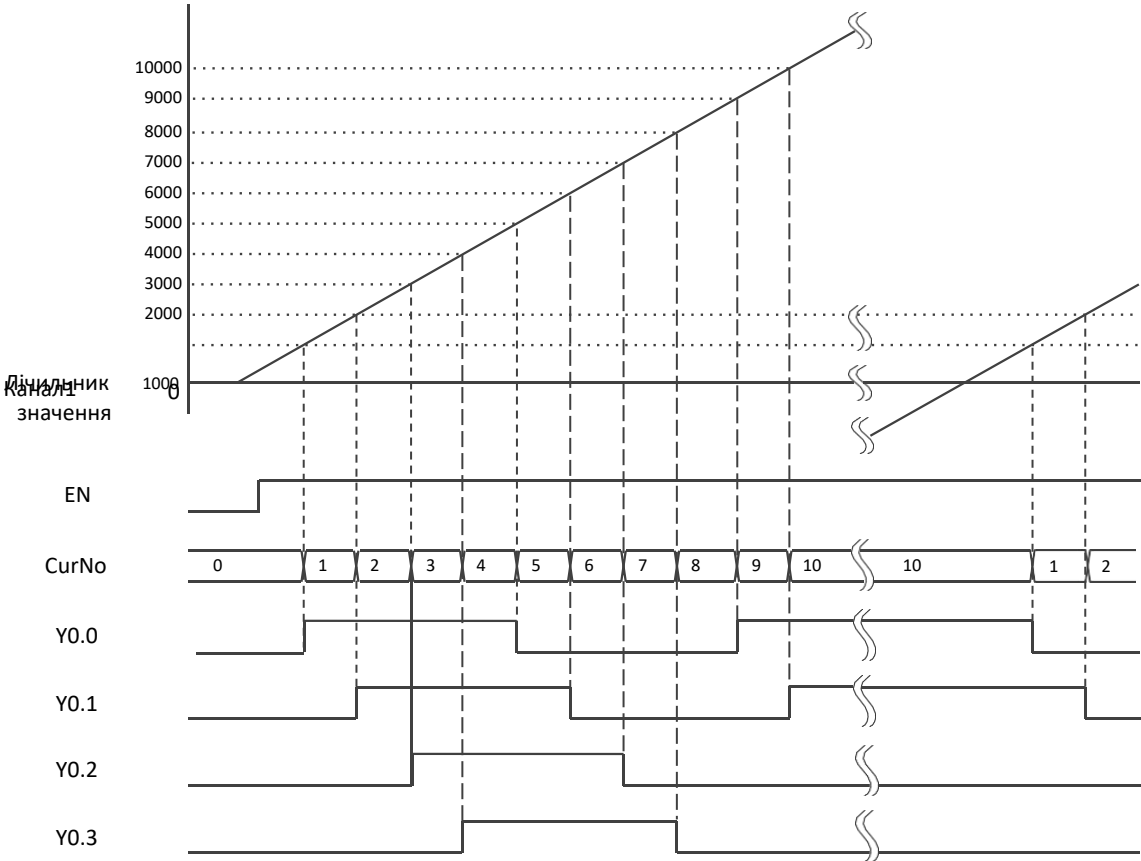
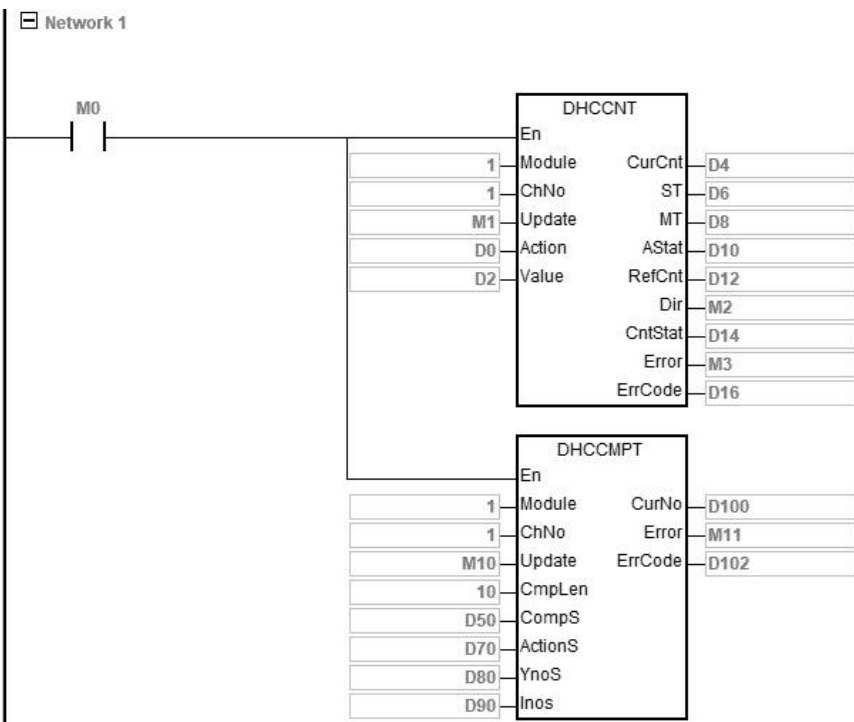
3. Коли значення лічильника досягає 1000 (999→1000), відповідний **CurNo** перемикається з 0 на 1, а **Y0.0** перемикається з ВІМК. на УВІМК., і виконується переривання I400. Якщо вхідний інтерфейс є імпульсним, порівняння починається, коли значення лічильника досягає 1001 (1000→1001).

4. Коли значення лічильника досягає 2000 (1999→2000), його відповідний **CurNo** перемикається з 1 на 2, а **Y0.1** перемикається з ВІМК. на УВІМК., і виконується переривання I401. Якщо вхідний інтерфейс є імпульсним, порівняння починається, коли значення лічильника досягає 2001 (2000→2001).

5. Коли значення лічильника досягає 3000 (2999→3000), його відповідний **CurNo** перемикається з 2 на 3, а **Y0.2** перемикається з ВІМК. на УВІМК., і виконується переривання I402. Якщо вхідний інтерфейс є імпульсним, порівняння починається, коли значення лічильника досягає 3001 (3000→3001).

6. Дотримуйтеся того самого правила, щоб виконати порівняння, доки **CurNo** не стане 10. Після останнього порівняння встановить значення лічильника на 1000, щоб **CurNo** міг повернутися до 1.

7. Якщо ви використовуєте лічильник викликів, коли значення лічильника знову досягає 1000, порівняння починається знову. **CurNo** дорівнює 1, **Y0.0** перемикається з **OFF** на **ON**, і виконується переривання I400. І дотримуйтеся того самого правила, щоб порівняти.



API	Інструкція				Операнд							опис						
1423	D	HCMEAS			Module ~ ErrCode							Module HC вимірювання частоти та обертів						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
Module								●					○	○				
ChNo								●					○	○				
Update		●	●	●				●										
Cnt/Rev								●					○	○				
Smpl								●					○	○				
Avg								●					○	○				
Freq								●										
RPM								●										
Error		●	●	●				●										
ErrCode								●										
пристрій	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING					
Module		●				●												
ChNo		●				●												
Update	●																	
Cnt/Rev			●				●											
Smpl		●				●												
Avg		●				●												
Freq			●				●											
RPM			●				●											
Error	●																	
ErrCode		●				●												
						Інструкція по пульсу		16-бітна інструкція				32-розрядна інструкція						
						-		-				AS						

СИМВОЛ

DHCMEAS	
En	
Module	Freq
ChNo	RPM
Update	Error
Cnt/Rev	ErrCode
Smpl	
Avg	

- Module : Номер модуля
- ChNo : Номер каналу
- Update : Прапор оновлення параметрів
- Cnt/Rev : Розраховане значення оборотів датчика за один раз
- Smpl : Цикл відбору проб
- Avg : Середній час
- Freq : Виміряна частота
- RPM : Виміряні оберти за хвилину (rpm)
- Error : Прапор помилки
- ErrCode : Код помилки

Пояснення

1. Ця інструкція стосується лише AS02HC-A. Ця інструкція доступна для ПЛК із програмним забезпеченням версії 1.08.або новішої та ISPSOft V3.10 чи новішої.
2. Використовуйте цю інструкцію з інструкцією **DHCCNT** . При виконанні інструкції **DHCCNT** починається підрахунок. Потім ви можете використовувати інструкцію **DHCCMP** для порівняння.
3. Перед виконанням цієї інструкції вам потрібно налаштувати **Cnt/Rev, Smpl i Avg** . Після активації **En** ПЛК негайно сканує значення в **Cnt/Rev, Smpl i Avg** .
4. Ви можете редагувати значення в **Cnt/Rev, Smpl i Avg** під час роботи. Встановіть нові значення та встановіть для параметра **Оновлення значення** УВІМК. Після оновлення значень прапор **Оновлення** вимикається.
5. **Module** : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість.
6. **ChNo** : Номер каналу для лічильника, канал 1 є номером 1; канал 2 - номер 2.
7. **Cnt/Rev** : Розраховане значення обороту кодера за один раз у діапазоні від 1 до 4294967295 (H'00000001~H'FFFFFFF).
8. **Smpl** : Цикл вибірки в діапазоні від 1 до 1000, одиниця: мс. Різні значення, встановлені в **Smpl i Freq** для потужності та RPM для виведення може спричинити різну роздільну здатність.

Роздільна здатність по частоті = $1000 \div \text{Smpl}$ (одиниця: Гц)

Роздільна здатність RPM = $60000 \div \left(\text{Smpl} \times \frac{\text{Cnt}}{\text{Rev}} \right)$ (одиниця: об/хв)

Початок циклу відбору проб і поточне підраховане число (**CurCnt**) є двома основними компонентами для розрахунку оборотів. Тому вам необхідно взяти до уваги наступні фактори, які можуть вплинути на результати вимірювань..

Тип кодера	Тип лічильника	Фактори, що впливають на результати вимірювань
Інкрементальний кодер	Лічильник кільця	Різниця між підрахованими значеннями в одній вибірці циклу перевищує межу 2^{31} .
	Лінійний лічильник	Різниця між підрахованими значеннями в одній вибірці циклу перевищує межу 2^{31} або підраховане значення перевищує верхню/нижню межу.
SSI кодер В Абсолютній Позиції	Абсолютна позиція	Різниця між підрахованими значеннями в одній вибірці циклу перевищує обмеження $2^{(MT+ST\ Length)-1}$.
	Лічильник кілець	Різниця між підрахованими значеннями в одній вибірці циклу перевищує межу 2^{31} .

9. **Avg** – це середнє значення часу та ковзного середнього результату вимірювання в діапазоні від 1 до 10.

10. **Freq** – результат вимірювання середньої частоти; одиниця: Гц. Формат наведений нижче.

$$\text{Частота (Гц)} = \frac{\text{CurCnt} (t + \mathbf{Smpl}) - \text{CurCnt} (t)}{\text{Smpl} (\text{мс}) \times 10^{-3}}$$

11. **RPM** – результат вимірювання середнього обороту; одиниця: об/хв. Формат наведений нижче.

$$\text{Оберти (об/хв)} = \frac{(\text{CurCnt} (t + \mathbf{Smpl}) - \text{CurCnt} (t)) \times 60}{\mathbf{Cnt/Rev} \times \mathbf{Smpl} (\text{мс}) \times 10^{-3}}$$

12. Якщо ви припинили виконання інструкції, її функція вимірювання припиниться. Значень **Freq** і **RPM** не буде оновлено.

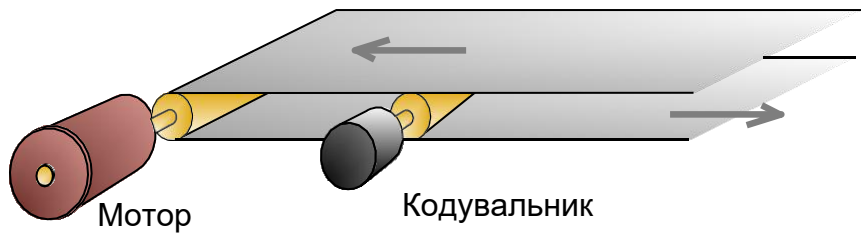
13. Зауважте, що якщо ви очистите або змініте значення лічильника, це вплине на результат вимірювання **частоти** та **обертів у циклі**.

14. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів помилки увімкнено.

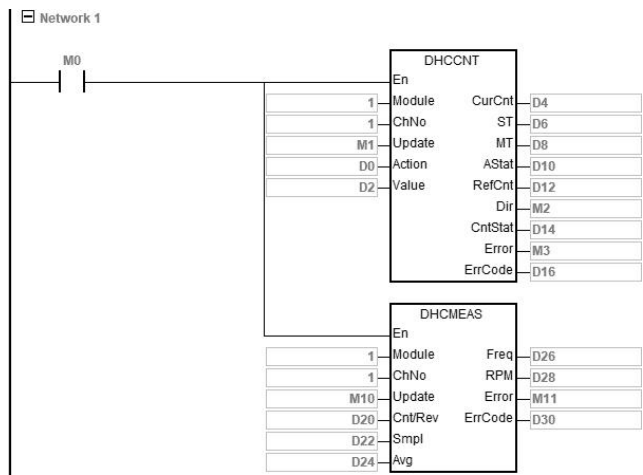
ErrCode	опис
16#1400	Ця функція НЕ доступна для цього модуля.
16#1401	Значення недійсне або виходить за межі діапазону.
16#1402	Немає відповіді від модуля. Час очікування зв'язку.
16#1403	Error конфігурації HWCONFIG. Інструкція зупиняється.
16#1404	Для модуля HC такого каналу підрахунку немає.
16#1405	Інтерфейс введення не вибрано. Інструкція зупиняється.
16#140D	Недійсне значення в Smpl
16#140E	Недійсне значення в сеп
16#140F	Недійсне значення в Cnt/Rev
16#1415	Один з лічильних каналів модуля HC виконує вимірювання частоти. Інші канали не можуть виконувати те саме завдання одночасно.

Приклад програмування 1: Інкрементальний кодер

Швидкість двигуна можна контролювати, виконавши інструкцію DHCMEAS, як описано нижче. Роздільна здатність кодера в цьому прикладі встановлена на 1000 PPR.



- 1. Встановіть вхідний інтерфейс каналу 1 на імпульсний вхід або вхід SSI у HWCONFIG. І встановіть тип лічильника на кільцевий лічильник.
- 2. Установіть значення в **Cnt/Rev** на 1000, значення в **Smpl** на 1000 і значення в **Avg** на 10.
- 3. Коли M0 перемикається в положення ON, лічильник починає підраховувати та налаштовувати параметри DHCMEAS у налаштуваннях модуля одночасно для вимірювання частот і швидкості, результати яких оновлюватимуться на дисплеї Freq і RPM кожні 1000 мс.
- 4. Коли двигун працює, а кодер отримує імпульси 5 кГц, результати відображатимуться як 5 кГц fin Freq і 300 об/хв у RPM. У цьому прикладі швидкість кодера є швидкістю двигуна, тому двигун тут буде 300 об/хв.



Приклад програмування 2: Absolute SSI Encoder (27-bit)

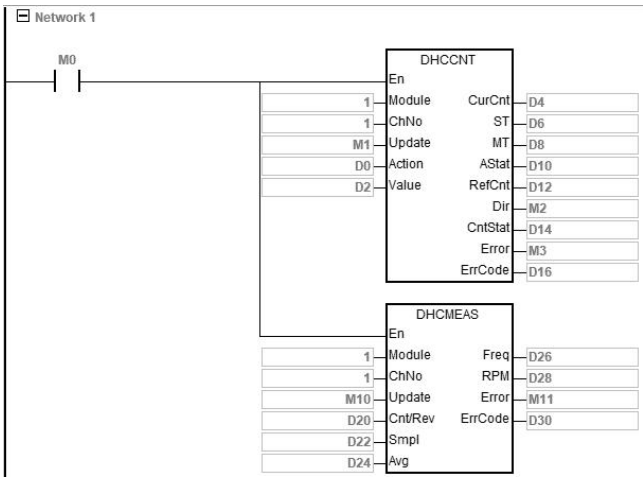
Продовжуючи попередній приклад, тепер, якщо змінити інкрементальний кодер на 13-бітний абсолютний кодер SSI, кроки налаштування параметрів DHCMEAS показані таким чином.

1. Встановіть вхідний інтерфейс каналу 1 на вхід SSI у HWCONFIG

2. Встановіть значення в **Cnt/Rev** на 8192, значення в **Smpl** на 1000 і значення в **Avg** на 10.

3. Коли M0 перемикається в положення ON, лічильник починає підраховувати та налаштовувати параметри DHCMEAS у налаштуваннях модуля одночасно для вимірювання частот і швидкості, результати яких оновлюватимуться на дисплеї Freq і RPM кожні 1000 мс.

4. Коли двигун працює, а швидкість зміни положення абсолютного кодера з SSI становить 4095 LSB/с, результати відображатимуться як 4095 кГц у Freq та 30 об/хв у RPM. У цьому прикладі швидкість кодера є швидкістю двигуна, тому двигун тут буде 30 об/хв.



API		Інструкція				Операнд						опис					
1424		D	ADLOG			Перераховані наступним чином						Журнал даних модулів аналогового введення					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Group								●					○	○			
Module								●					○	○			
ChNo								●					○	○			
Mode								●					○	○			
Period								●					○	○			
Points								●					○	○			
PostTrg								●					○	○			
Datalog								●									
CurNo								●									
Cmplt		●	●	●				●									
Error		●	●	●				●									
ErrCode								●									
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
Group		●	●			●	●										
Module		●	●			●	●										
ChNo		●	●			●	●										
Mode		●	●			●	●										
Period		●	●			●	●										
Points		●	●			●	●										
PostTrg		●	●			●	●										
Datalog		●	●			●	●		●								
CurNo		●				●											
Cmplt	●																
Error	●																
ErrCode		●	●			●	●										

ADLOG		DADLOG	
En		En	
Group	Datalog	Group	Datalog
Module	CurNo	Module	CurNo
ChNo	Cmplt	ChNo	Cmplt
Mode	Error	Mode	Error
Period	ErrCode	Period	ErrCode
Points		Points	
Postrig		Postrig	

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

- Group : Номер групи
- Module : Номер модуля
- ChNo : Номер каналу
- Mode : Режим запису
- Period : Період запису

Points	: Загальна кількість усіх записів
PostTrg	: Кількість записів після запуску
Datalog	: Пристрій для зберігання журналів даних
CurNo	: Кількість накопичених точок журналу
Cmplt	: Позначка завершення журналів запису
Error	: Прапор помилки
ErrCode	: Код помилки

Пояснення

- Ця інструкція призначена для модулів аналогового введення (AS04AD-A, AS06XA-A, AS02ADH-A), щоб увімкнути або вимкнути функцію запису та надіслати дані запису на призначені пристрої Datalog. Ця інструкція доступна для ПЛК з FW V1.10.00 або пізнішої версії та ISPSOft V3.13 або пізнішої версії.
- Опис періоду запису та точок журналу наведено в наступній таблиці. Зверніться до опису режиму для отримання додаткової інформації.

Модель	Період запису	Кількість записів
AS04AD-A	<u>Режим фіксованого періоду:</u>	Фіксоване число 500
AS06XA-A	Діапазон налаштувань від 1 до 100, а одиниця часу фіксована на 10 мс.	
AS02ADH-A	<u>Режим фіксованого періоду, фіксований період + режим запуску тригера, фіксований період +</u> <u>Режим призначення позиції тригера:</u> Діапазон налаштувань становить від 1 до 32000, а варіанти одиниці часу: 20 мкс, 40 мкс і 80 мкс. (Примітка: одиницею часу є період вибірки, встановлений у HWCONFIG.)	Діапазон налаштувань від 1 до 2000.
	<u>Режим реєстрації точок:</u> Точка буде записана після виявлення сигналу запуску на зовнішній вхідній точці не протягом фіксованого періоду. Час запуску для зовнішніх точок введення встановлюється в HWCONFIG. Час запуску каналу 1 – це коли наростаючий або спадаючий фронт виявляється на X0.0. Час запуску каналу 2 - це коли наростаючий або спадаючий фронт виявляється на X0.1.	

3. Group використовується для вказівки модуля аналогового введення, підключеного до правої сторони ПЛК безпосередньо або підключеного до віддаленого модуля, який діє як ПЛК, наприклад, ПЛК має номер 0, перший віддалений модуль матиме номер 1 і так далі. Максимальна кількість груп – 15 осіб.

4. Module : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість.

5. ChNo: вказаний номер каналу. Канал 1 матиме номер 1, а канал 2 – номер 2.

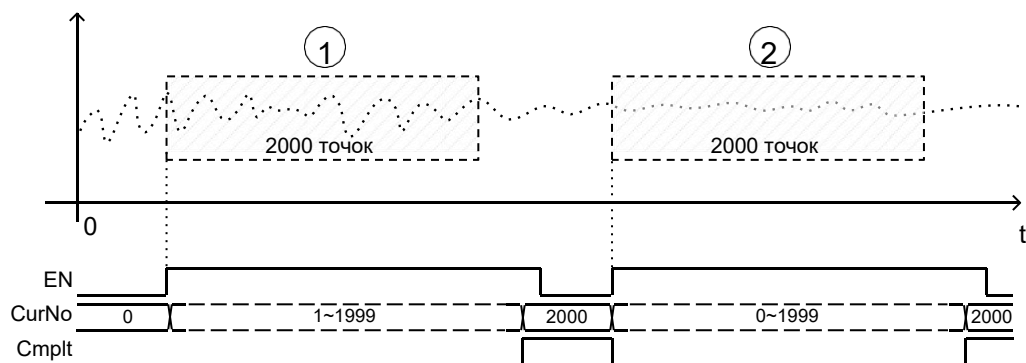
6. **Mode:** Режим запису. Нижче наведено режими, які підтримуються відповідними моделями.

Моделі	Mode	Назва режиму	опис
AS04AD-A AS06XA-A	K0	Фіксований період	Періодично виконуйте запис.
AS02ADH-A	K0	Фіксований період	Періодично виконуйте запис.
	K1	Фіксований період + старт тригера	Зачекайте, поки на зовнішніх входних точках буде виявлено тригерний сигнал. Після отримання здійснюватиметься періодичний запис.
	K2	Точковий журнал	Немає фіксованого періоду для запису. Одна точка журналу записуватиметься кожного разу, коли сигнал запуску буде виявлено на зовнішній входній точці.
	K3	Фіксований період + призначення позиції тригера	Ви можете вказати час зовнішнього запуску записуйте журнали даних протягом періодів до та після часу запуску.

(1) Режим фіксованого періоду: Установіть **режим** =0, запис даних виконуватиметься відповідно до попередньо визначеного запису періоду, коли **EN** перемикається на ON. Після завершення запису вказаної кількості точок журналу **Cmplt** прапор буде встановлено на High автоматично.

приклад:

Задані **точки** = 2000



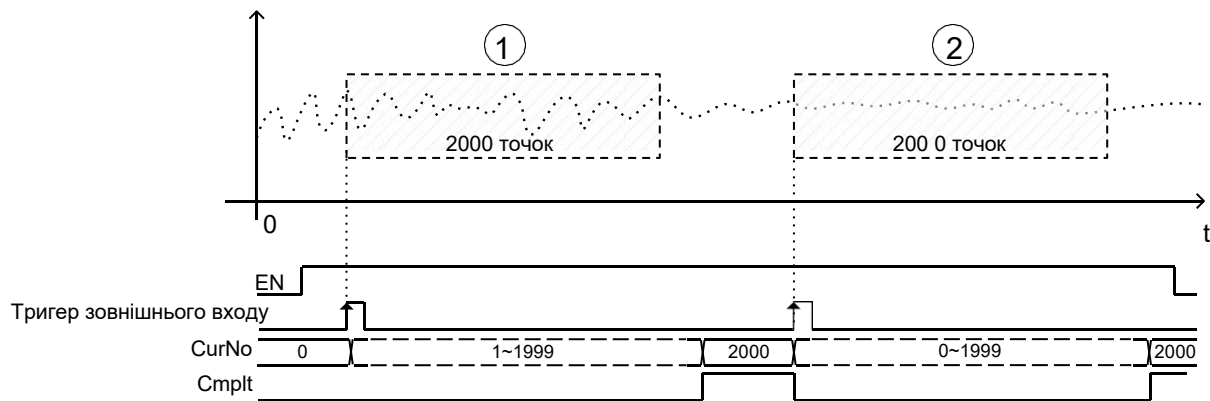
(2) Фіксований період + режим запуску тригера: встановіть **режим** =1 і встановіть інструкцію **EN** на ON перед початком запису.

Коли на зовнішній входній точці виявлено тригерний сигнал, почніть запис на основі попередньо визначеного запису періода. Після завершення прапор **Cmplt** автоматично встановлюється на High. До завершення запису будь-яка операція на зовнішніх точках введення НЕ впливає на продовження запису. Коли запис точок журналу завершено, а прапорець **Cmplt** має **значення High**, знову запустіть зовнішні точки введення, щоб почати новий цикл запису; інструкцію **EN** НЕ потрібно вимикати, а потім знову вмикати, щоб розпочати новий запис.

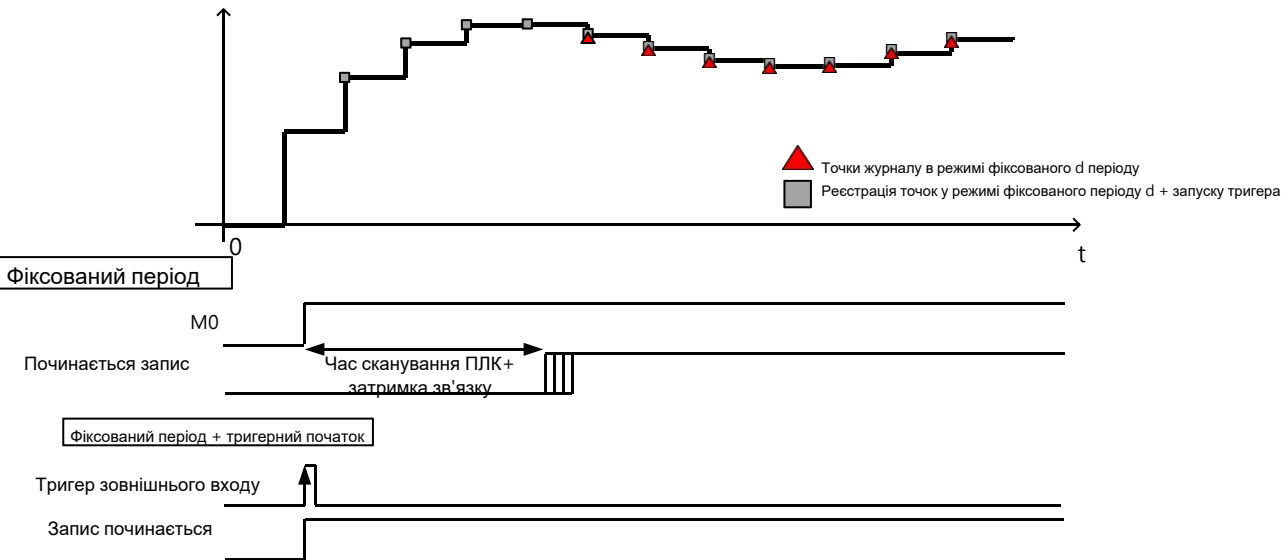
запис Канал	Джерело сигналу, що відповідає зовнішнім вхідним точкам (Установіть час для тригера зовнішнього введення в HWCONFIG)
Канал 1	X0.0 спрацьовує наростаючий або спадаючий фронт
Канал 2	X0.1 спрацьовує наростаючий або спадаючий фронт

приклад:

Set **Points** = 2000, час запуску для зовнішньої вхідної точки встановлено на спрацьовування по наростаючому фронту.



Функція «Фіксований період + тригерний початок» схожа на «Фіксований період». Але час початку запису в Fixed режим періоду буде відкладено в результаті часу сканування ПЛК і часу зв'язку модуля, що показано на наступній ілюстрації. Передбачається, що M0 є пристроєм для керування EN інструкції ADLOG. Ми бачимо, коли M0 перемикається з ВІМК. на УВІМК., модуль починає запис не відразу, а з невеликою затримкою.

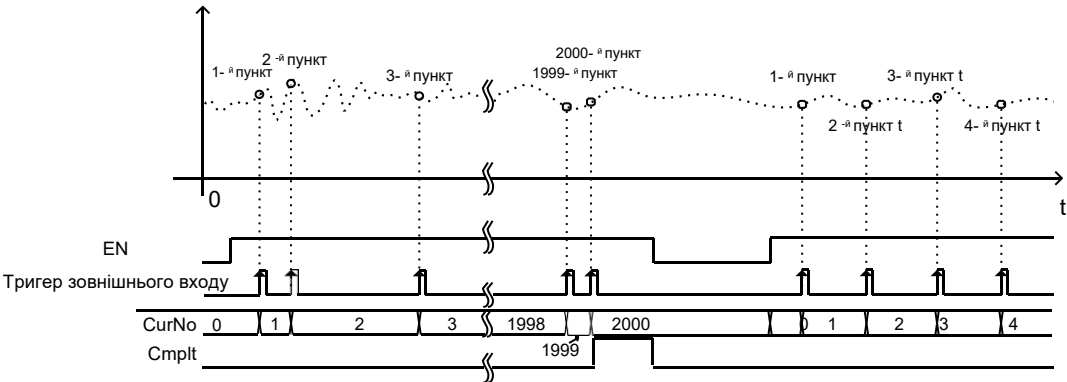


(3) Режим реєстрації точки: встановіть **режим** =2, увімкніть інструкцію EN перед початком запису. Одна точка журналу записуватиметься для кожного запуску на зовнішній вхідній точці, доки не досягне попередньо визначеного номера точки, прапорець **Cmplt** буде автоматично встановлено на високий. Якщо потрібно продовжити запис після того, як для прапорця **Cmplt** встановлено значення High, виконайте інструкцію ще раз.

запис Канал	Джерело сигналу, що відповідає зовнішнім вхідним точкам (Установіть час для тригера зовнішнього введення в HWCONFIG)
Канал 1	X0.0 спрацьовує наростаючий або спадаючий фронт
Канал 2	X0.1 спрацьовує наростаючий або спадаючий фронт

приклад:

Set **Points** = 2000, час запуску для точки зовнішнього введення встановлено на спрацьовування по наростаючому фронту.

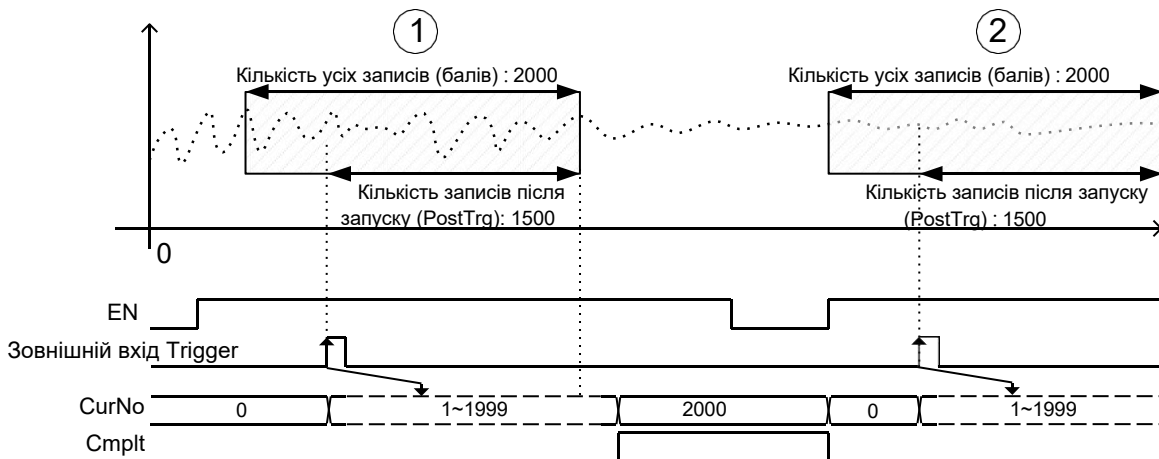


(4) Фіксований період + положення тригера Призначити режим: встановити **режим** =3 і встановити параметри **Points** і **PostTrg** відповідно до ваших вимог. Цей режим запускається у зовнішній точці введення та записує попередньо визначену кількість точок журналу до та після запуску. Якщо використовувати EN для ввімкнення цього режиму, AS02ADH-A почне очікувати сигналів, які будуть запускатися на зовнішньому вході. І вибірка розпочнеться одразу після цього, доки не досягне попередньо визначений номер точки, а потім прапорець **Cmplt** автоматично встановить значення High. Значення в **CurNo** дорівнює 0 перед ініціюванням, а після ініціювання модулі починають надсилати журнал даних до ініціювання до ЦП ПЛК. тому значення **CurNo** буде відповідати кількості накопичених точок журналу.

запис Канал	Джерело сигналу, що відповідає зовнішнім вхідним точкам (Установіть час для тригера зовнішнього введення в HWCONFIG)
Канал 1	X0.0 спрацьовує наростаючий або спадаючий фронт
Канал 2	X0.1 спрацьовує наростаючий або спадаючий фронт

приклад

Установіть **Mode** =3, **Points** = 2000 і **PostTrg** = 1500, щоб позиція точки 501 (**Points** – **PostTrg**) була першим записом після виявлення зовнішнього тригерного сигналу.



7. **Period** встановлюється для періоду запису, який деталізується наступним чином:

Модель	Встановлення діапазону для періоду	Одиниця періоду
AS04AD-A AS06XA-A	1~100	Одиниця виміру становить 10 мс, незмінна.
AS02ADH-A	1~32000	Період вибірки потрібно встановити в HWCONFIG з параметрами 20us, 40us і 80us. Якщо встановити « Mode =2:Реєстрація точок», значення Періоду буде недійсним.

8. **Points** – це загальна кількість записів, яка застосовна лише для AS02ADH-A та може встановити максимум 2000 записів. Для інших моделей значення параметра встановлено на 500.

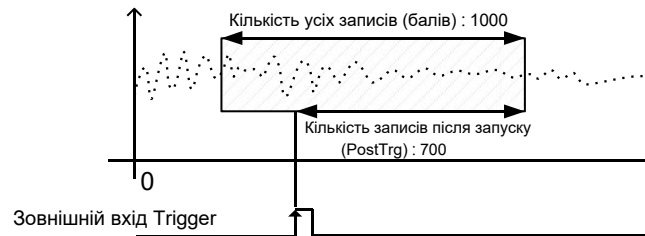
Модель	Діапазон налаштування для загальної кількості записів
AS04AD-A AS06XA-A	Зафіксовано значення 500, інше значення налаштування недійсне.
AS02ADH-A	Можливість налаштування, значення параметра в діапазоні від 1 до 2000.

9. **PostTrg** — це кількість точок журналу, записаних після запуску тригера, які слід застосувати з **Points** , щоб записати кількість усіх записів, включаючи дані до і після запуску. **PostTrg** застосовний лише для режиму призначення позиції фіксованого періоду + тригера (**Mode** =3) і недійсний для інших режимів. Зверніть увагу, що значення **PostTrg** не має перевищувати значення в **балах** (кількість усіх записів), інакше значення в **PostTrg** буде замінено на значення в **балах** . Наприклад, якщо **Points** = 100 і **PostTrg** = 200, значення в **PostTrg** розглядатиметься як значення в **Points**, яке дорівнює 100, оскільки значення в **PostTrg** більше за значення в **Points** (кількість усіх записів). Таким чином, лише 100 точок будуть записані після запуску.

Модель	Діапазон налаштування для значення PostTrg
AS02ADH-A	0 ~2000 балів (не має перевищувати значення в балах)

приклад

Set **Points** = 1000 і **PostTrg** = 700, а потім буде 1000 журналів, включаючи 300 фрагментів даних до запуску та 700 фрагментів даних після запуску.



10. **Datalog** — це пристрій для зберігання журналів даних. Якщо вибрати інтеграл як формат вихідних значень модуля AD, ми пропонуємо використовувати його з 16-розрядною інструкцією ADLOG. За допомогою AS04AD-A або AS06XA-A буде зайнято простір у 500 слів, а за допомогою моделі AS02ADH-A — простір від 1 до 2000 слів (залежно від загальної кількості записів). (якщо використовується 32-бітна інструкція DADLOG, 16-бітні дані будуть розширені до 32 бітів, причому простір даних займає подвійно.) Якщо вибрати формат вихідних значень модуля AD з плаваючою комою, ви повинні використовувати його з 32-бітна інструкція ADLOG. За допомогою AS04AD-A або AS06XA-A буде зайнято простір у 1000 слів, а за допомогою AS02ADH-A — простір від 2 до 4000 слів (залежно від загальної кількості записів).
11. **CurNo** — це кількість записів, які модуль надіслав назад до ПЛК. Коли запис триває, значення **CurNo** може бути недоступним для перегляду в послідовному порядку в результаті впливів протягом періоду сканування.
12. **Cmplt**, позначка завершення використовується для вказівки на завершення запису журналів. Коли **Cmplt** перемикається з ВИМК. на УВИМК., це означає, що всі дані журналу надіслано на призначений пристрій **реєстрації даних**. Коли **EN** перемикається з ВИМК. на УВИМК., прапорець **Cmplt** автоматично встановлюється на ВИМК.
13. Якщо ви припините виконання цієї інструкції, призначений канал припинить оновлення правої сторони значень.
14. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, коли будь-який із прапорів помилки увімкнено.

ErrCode	опис
16#1400	Ця функція НЕ доступна для цього модуля.
16#1401	Налаштування номера групи або номера модуля неправильне.
16#1402	Вхідне значення модуля є недопустимим або поза діапазоном.
16#1403	Немає відповідей від модуля. Час очікування зв'язку.
16#1404	Канал не знайдено в модулі AD.
16#1405	Канал, призначений модулем AD, триває запис. Інші канали не можуть виконувати те саме завдання одночасно.
16#1406	Формат даних модуля AD та інструкції не збігаються.
16#1407	Значення в режимі запису виходить за межі діапазону.
16#1408	Значення за період запису виходить за межі діапазону.
16#1409	Кількість усіх записів (Points) поза діапазоном.

API	Інструкція				Операнд								опис			
1425	D	ADPEAK			Перераховані наступним чином								Записуйте пікові значення модулів аналогового введення.			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Group								●					○	○		
Module								●					○	○		
ChNo								●					○	○		
МАКС								●								
ХВ								●								
Error		●	●	●				●								
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●	●			●	●						
Module		●	●			●	●						
ChNo		●	●			●	●						
МАКС		●	●			●	●		●				
ХВ		●	●			●	●		●				
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

ADPEAK				DADPEAK			
En				En			
Group	MAX			Group	MAX		
Module	MIN			Module	MIN		
ChNo	Error			ChNo	Error		
ErrCode				ErrCode			

- Group : Номер групи
- Module : Номер модуля
- ChNo : Номер каналу
- МАКС : Максимум
- ХВ : Мінімум
- Error : Прапори помилок
- ErrCode : Коды помилок

Пояснення

1. ADPEAK призначений для модулів аналогового введення (AS04AD-A, AS08AD-B, AS08AD-C, AS06XA-A, AS02ADH-A), щоб увімкнути та вимкнути запис пікових значень. Ця інструкція доступна для ПЛК з FW V1.10.00 або пізнішої версії та ISPSOft V3.13 або пізнішої версії.
2. **Group** використовується для вказівки модуля аналогового введення, підключеного безпосередньо до правої сторони ПЛК або підключеного до віддаленого модуля, який діє як ПЛК, наприклад, ПЛК має номер 0, перший віддалений модуль матиме номер 1 і так далі. Максимальна кількість груп – 15 осіб.
3. **Module** : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля
2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість.
4. **ChNo**: вказаний номер каналу. Канал 1 матиме номер 1, а канал 2 – номер 2.
5. **MAX** і **MIN** є максимальним і мінімальним значеннями відповідно. Коли EN перемикається з ВІМК. на УВІМК., значення **MAX** і **MIN** будуть ініціалізовані новим вимірним значенням і розпочнеться запис пікових значень. Значення **MAX** і **MIN** постійно оновлюватимуться під час виявлення максимальних і мінімальних значень. Якщо вибрати інтеграл як формат вихідних значень модуля AD, ми пропонуємо використовувати його з 16-бітною інструкцією ADPEAK. Якщо вибрати формат вихідних значень модуля AD з плаваючою точкою, ви повинні використовувати його з 32-розрядною інструкцією DADPEAK.
6. Якщо **En** перемикається з УВІМК. на ВІМК., функція запису пікових значень буде вимкнена. Тим часом цінності в **MAX** і **MIN** залишаться без змін і не оновлюються.
7. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів помилки увімкнено

ErrCode	опис
16#1400	Ця функція НЕ доступна для цього модуля.
16#1401	Налаштування номера групи або номера модуля неправильне.
16#1402	Вхідне значення модуля є недопустимим або поза діапазоном.
16#1403	Немає відповідей від модуля. Час очікування зв'язку.
16#1404	Канал не знайдено в модулі AD.
16#1405	Канал, призначений модулем AD, триває запис. Інші канали не можуть виконувати те саме завдання одночасно.
16#1406	Формат даних модуля AD та інструкції не збігаються.

API	Інструкція			Операнд	опис
1426		IOLINKR		Виконати ~ DataRead	Зчитайте параметр із пристрою O-Link

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Group								●					○	○		
Module								●					○	○		
Port								●					○	○		
Index								●					○	○		
SubIndex								●					○	○		
DataType								●					○	○		
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								
ReadLen								●								
DataRead								●								

тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
Port		●				●							
Index		●				●							
SubIndex		●				●							
DataType		●				●							
Done	●												
Error	●												
ErrCode		●				●							
ReadLen		●				●							
DataRead		●				●							
тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
Port		●				●							
Index		●				●							
SubIndex		●				●							
DataType		●				●							
Done	●												
Error	●												
ErrCode		●				●							
ReadLen		●				●							
DataRead		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

IOLINKR	
En	
Group	Done
Module	Error
Port	ErrCode
Index	ReadLen
SubIndex	DataRead
DataType	

Group	: Номер групи
Module	: Номер модуля
Port	: Номер комунікаційного порту
Index	: Індексний номер параметра
SubIndex	: Підіндексний номер параметра
DataType	: Тип даних
ReadLen	: Довжина зчитаних даних
DataRead	: Дані, які зчитуються
Done	: Зв'язок завершений
Error	: Прапор помилки
ErrCode	: Код помилки

Пояснення

- Ця інструкція доступна для ПЛК з FW V1.08.50 або пізнішої версії та ISPSOft V3.12 або пізнішої версії.
- Щоб дізнатися більше про параметри пристрою, перегляньте файл IODD пристрою IO-Link. Ви також можете використовувати цю інструкцію для читання параметрів пристрою IO-Link.
- Немає обмежень щодо кількості виконання інструкції, але одночасно може бути виконана лише одна інструкція (IOLINKR або IOLINKW).
- Під час зв'язку пристрій IO-Link відповість не менше 5 секунд. НЕ припиняйте і не починайте іншу інструкцію протягом цього часу. Після завершення зв'язку з IO-Link пропонується використовувати прапорці DONE або ERROR, щоб зупинити або запустити інструкцію.
- Group** : номер групи AS04SIL, який підключено до правої сторони ПЛК безпосередньо або підключено до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль – це модуль номер 1, другий підключений модуль – номер 2. Можна підключити до 15 модулів. Якщо підключений модуль не AS04SIL, прапор помилки перемикається на ON.
- Module** : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість. Якщо модуль не є AS04SIL, прапор помилки перемикається на ON.
- Port** : номер порту зв'язку в діапазоні від 1 до 4. Якщо номер порту, який використовується, не є портом для AS04SIL, прапор помилки вмикається.
- Index** — це номер індексу, який зчитується з параметра пристрою IO-Link.
- SubIndex** – це номер субіндексу, який зчитується з параметра пристрою IO-Link. Номер підіндексу можна встановити від 0 до 255. Якщо значення перевищує діапазон налаштувань, прапорець Error вмикається.
- DataType** — це тип даних параметра пристрою IO-Link. Вам потрібно встановити правильний тип даних перед виконанням цієї інструкції.

Якщо значення налаштування перевищує діапазон налаштування, прапорець Error перемикається на ON.

Код помилки	Тип даних	опис
0x0000	BooleanT	логічний; довжина даних: 1 байт
0x0001	UIntegerT	Беззнакове ціле число; довжина даних: 1, 2, 4, 8 байт
0x0002	Ціле числоT	Ціле число зі знаком; довжина даних: 1, 2, 4, 8 байт
0x0003	Float32T	з плаваючою комою; довжина даних: 4 байти
0x0004	StringT	рядок; довжина даних: 1~232 байт; Null (0x00) розглядається як кінцевий символ для рядка.
0x0005	OctetStringT	Рядок фіксованої довжини, визначений фіксованою довжиною IODD; довжина даних: 1~232 байт
0x0006	ЧасT	час; довжина даних: 4 байти (роздільна здатність: 1 секунда) + 4 байти (роздільна здатність: 2 ⁻³² секунд)
0x0007	TimeSpanT	час; довжина даних: 8 байт (роздільна здатність: 2 ⁻³² секунди)
0x0008	ArrayT	Цілий масив, який зчитується з підіндексу номер 0
0x0009	Запис T	Цілий запис, який зчитується з підіндексу номер 0
інші	Зарезервований	Зарезервований

11. **DataRead** — дані, зчитані з пристрою IO-Link; одиниця: байт, упорядкований старшим і молодшим байтами.

12. **ReadLen** — довжина даних, зчитаних із пристрою IO-Link; одиниця: байт.

13. **Done** — це позначка завершення зв'язку. Коли цей прапорець увімкнено, це означає, що параметр пристрою IO-Link зчитується повністю. Діапазон зчитаних даних (ReadData) використовуватиметься відповідно до довжини даних (READLen).

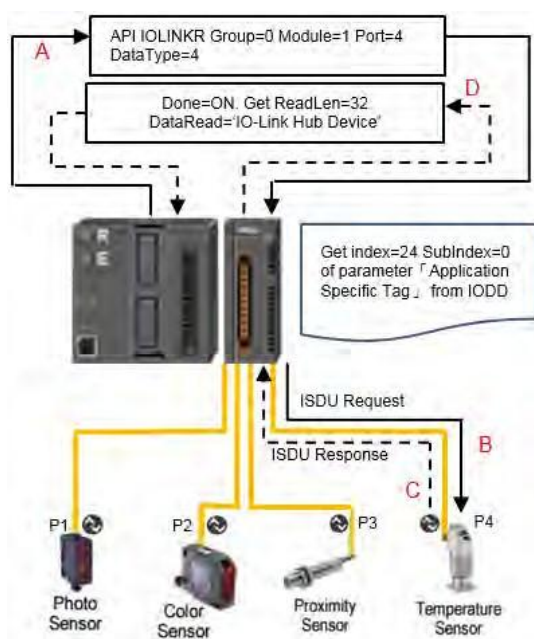
14. **Error** — це позначка помилки зв'язку. Якщо цей прапорець увімкнено, це означає, що під час зв'язку між ними сталася помилка ЦП ПЛК, AS04SIL і пристрій IO-Link. Зверніться до наведених нижче кодів помилок для усунення несправностей.

15. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів помилки увімкнено.

Код помилки	опис
0x8000	Error програми пристрою IO-Link
0x8011	Основного індексу НЕ існує.
0x8012	SubIndex НЕ існує.
0x8020	Наразі функція НЕ доступна.
0x8021	Функція наразі НЕ доступна – локальне керування.
0x8022	Функція наразі НЕ доступна – Контроль пристрою.
0x8023	Доступ заборонено.
0x8030	Параметр перевищує діапазон налаштування.
0x8031	Параметр перевищує верхню межу.
0x8032	Параметр перевищує нижню межу.
0x8033	Перевищення довжини параметра

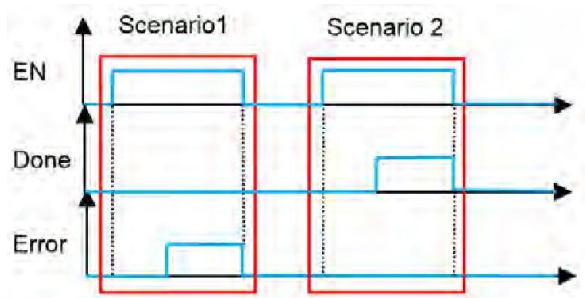
0x8034	Параметр заниження довжини
0x8035	Функція НЕ доступна.
0x8036	Наразі функція НЕ доступна.
0x8040	Встановлено недійсні параметри
0x8041	Непослідовний набір параметрів
0x8082	Програма НЕ готова до використання.
0x8101	Пристрій IO-Link НЕ в режимі IO-Link.
0x8102	Комунікаційний порт НЕ підключено до жодного пристрою IO-Link.
0x8103	Встановлюється підключення до пристрою IO-Link.
0x8104	Номер порту зв'язку перевищує діапазон налаштувань 1–4.
0x8105	Номер підіндексу параметра перевищує діапазон налаштування 0~255.
0x8106	Дані для читання та запису перевищують діапазон налаштувань 1~232.
0x8107	Зарезервований
0x8108	Тайм-аут зв'язку; відсутність відповіді від пристрою IO-Link більше 5000 мс.
0x8109	Неправильний номер модуля AS04SIL або номер групи
0x810A	Внутрішня помилка зв'язку
0x810B	Виконайте більше двох інструкцій читання/запису IO-Link на одному модулі AS04SIL.
0x810C	Початкова адреса пристрою читання/запису + довжина даних для читання/запису перевищує діапазон налаштування пристрою .
0x810D	Значення типу даних параметра перевищує діапазон налаштування.
0x810E	Тип даних параметра та довжина даних читання/запису не узгоджуються.

16. Процес зв'язку між ЦП ПЛК, AS04SIL і пристроєм IO-Link:



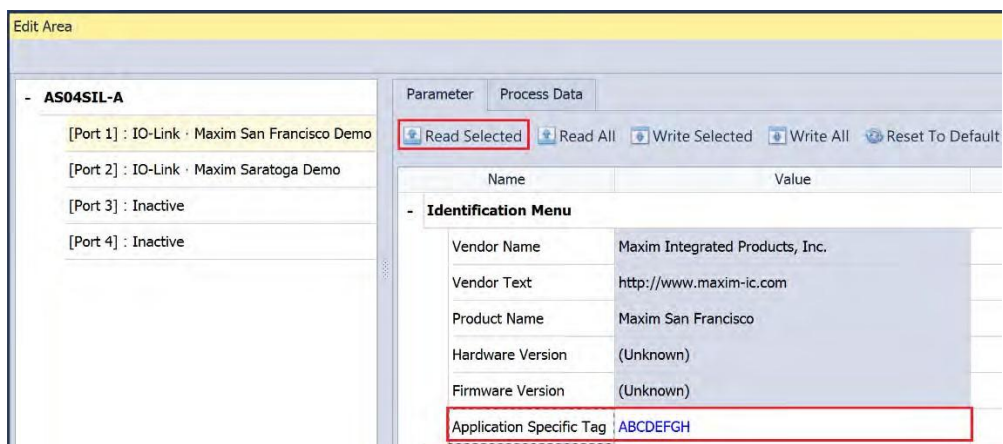
- B:** Центральний процесор ПЛК надсилає дані на AS04SIL.
- B:** AS04SIL надсилає дані на пристрій IO-Link.
- C:** пристрій IO-Link обробляє запит і надсилає дані на AS04SIL.
- D:** ЦП ПЛК оновлює статус (ГОТОВО / ERROR) і отримує дані.

17. Послідовність прапорів EN, Done та Error: коли виникає помилка під час зв'язку (сценарій 1) і коли зв'язок конкурує (сценарій 1).



Приклад 1: Зчитайте параметр спеціального тегу програми на першому пристрої першого порту зв'язку.

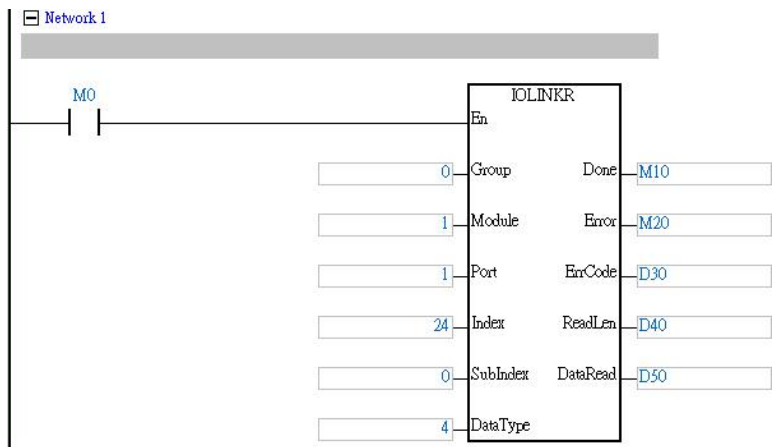
1. Перевірте HWCONFIG, щоб дізнатися, що значення параметра в спеціальному тегу програми – «ABCDEFGH».



2. AS04SIL є першим модулем, встановленим на правій стороні ЦП ПЛК. Значення налаштування в групі становить 0 і 1 дуюм як **модуль, так і порт** .

3. З файлу IODD або посібника з пристрою IO-Link ми дізналися, що його індекс дорівнює 24, субіндекс – 0, а тип даних – StringT. Встановіть значення налаштування в Index на 24, у Sub-index на 0, у типі даних на 4 (StringT).

4. Почніть надсилати запит на читання, коли M0 увімкнено. Коли M10 (ГОТОВО) горить, це означає, що пристрій відповів і завершив зчитування. ReadLen (D40) становить 32, а рядок даних зберігається в DataRead (D50~D65). Пристрій IO-Link зчитує дані та зберігає їх у порядку старшого та молодшого байтів.



Object	Identifiers	Device Name	Status	Data Type	Value (16bits)	Value (32bits)	Float	Radix	Comment...
		M1							
		M10							
		M20							
		D30			0000	00000000	0.0000	Hexadecimal	
		D40			32	32	0.0000	Signed Decima	
		D50			4241	44434241	781.0352	Hexadecimal	
		D51			4443	46454443	12625.0654	Hexadecimal	
		D52			4645	48474645	204057.0781	Hexadecimal	
		D53			4847	00004847	0.0000	Hexadecimal	
		D54			0000	00000000	0.0000	Hexadecimal	
		D55			0000	00000000	0.0000	Hexadecimal	
		D56			0000	00000000	0.0000	Hexadecimal	
		D57			0000	00000000	0.0000	Hexadecimal	
		D58			0000	00000000	0.0000	Hexadecimal	
		D59			0000	00000000	0.0000	Hexadecimal	
		D60			0000	00000000	0.0000	Hexadecimal	
		D61			0000	00000000	0.0000	Hexadecimal	
		D62			0000	00000000	0.0000	Hexadecimal	
		D63			0000	00000000	0.0000	Hexadecimal	
		D64			0000	00000000	0.0000	Hexadecimal	

Приклад 2: Зчитайте параметр регулювання посилення зеленого каналу на першому пристрої другого комунікаційного порту.

1. Перевірте HWCONFIG, щоб дізнатися, що значення параметра зеленого каналу регулювання посилення становить «-63».

AS04SIL-A

[Port 1] : IO-Link , Maxim San Francisco Demo

[Port 2] : IO-Link , Maxim Saratoga Demo

[Port 3] : Inactive

[Port 4] : Inactive

Parameter

Process Data

Read Selected

Read All

Write Selected

Write All

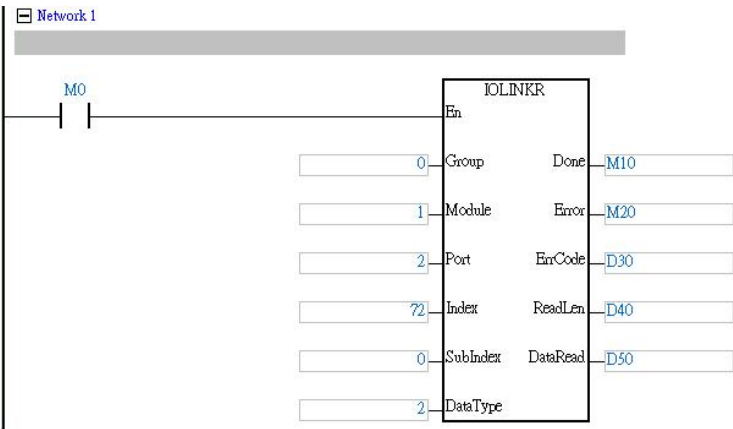
Reset To Default

Name	Value	Unit
gain trim green cha...	-63	
gain trim IR channel	-2	

2. AS04SIL є першим модулем, встановленим на правій стороні ЦП ПЛК. Значення налаштування в групі становить 0,1 дюйм Module і 2 в порту .

3. З його файлу IODD або посібника з пристрою IO-Link ми дізналися, що його індекс дорівнює 72, субіндекс – 0, а тип даних – IntegerT. Встановіть значення налаштування в Index на 72, у Sub-index на 0, у типі даних на 2 (IntegerT).

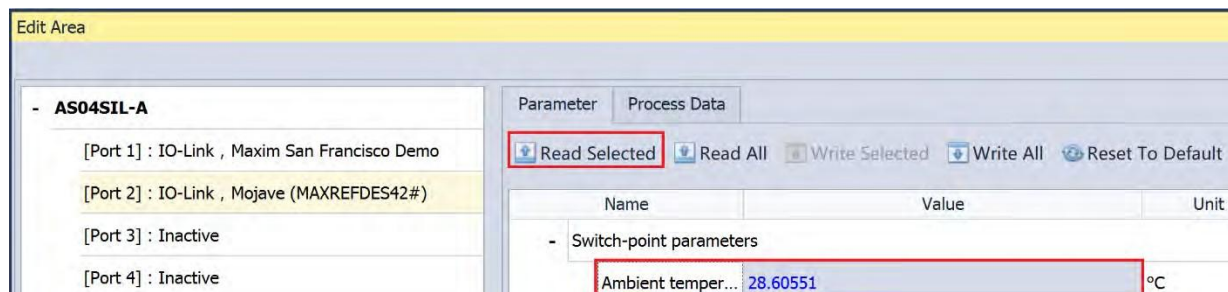
4. Почніть надсилати запит на читання, коли M0 увімкнено. Коли M10 (ГОТОВО) горить, це означає, що пристрій відповів і завершив зчитування. ReadLen (D40) дорівнює 1, а дані зберігаються в DataRead (D50 Low Byte). Пристрій IO-Link зчитує дані та зберігає їх у порядку старшого та молодшого байтів.



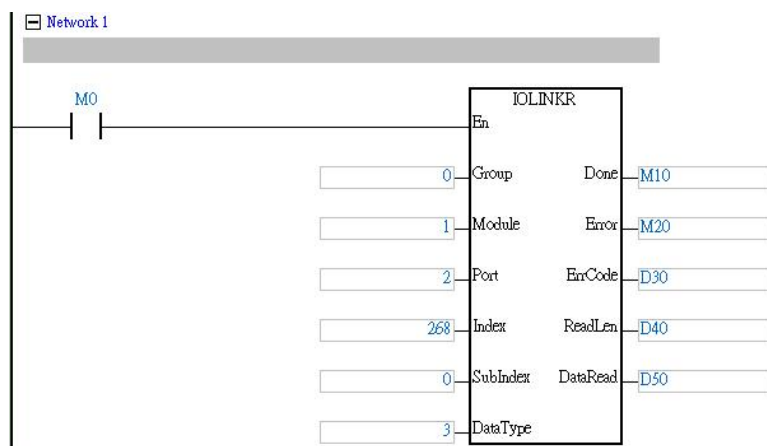
Object	Identifiers	Device Name	Status	Data Type	Value (16bits)	Value (32bits)	Float	Radix	Comment...
		M1							
		M10							
		M20							
		D30			0000	00000000	0.0000	Hexadecimal	
		D40			1	1	0.0000	Signed Decima	
		D50			00C1	000000C1	0.0000	Hexadecimal	

Приклад 3: Зчитайте параметр регулювання посилення зеленого каналу на першому пристрої другого комунікаційного порту.

1. Перевірте HWCONFIG, щоб дізнатися, що значення налаштування температури навколишнього середовища становить 28,60551.



2. AS04SIL є першим модулем, встановленим на правій стороні ЦП ПЛК. Значення налаштування в **групі** становить 0, **1 Module** і **2 в порту**.
3. З його файлу IODD або посібника з пристрою IO-Link ми дізналися, що його індекс дорівнює 268, субіндекс – 0, а тип даних – Float32T. Встановіть значення налаштування в Index на 268, у Sub-index на 0, у типі даних на 3 (Float32T).
4. Почніть надсилати запит на читання, коли M0 увімкнено. Коли M10 (ГОТОВО) горить, це означає, що пристрій відповів і завершив зчитування. ReadLen (D40) дорівнює 4, а дані зберігаються в DataRead (D50~D51). Пристрій IO-Link зчитує дані та зберігає їх у порядку старшого та молодшого байтів.



Object	Identifiers	Device Name	Status	Data Type	Value (16bits)	Value (32bits)	Float	Radix	Comment...
		M1							
		M10							
		M20							
		D30			0	0	0.0000	Signed Decima	
		D40			4	4	0.0000	Signed Decima	
		D50			-26209	1105566111	28.7000	Signed Decima	
		D51			16869	16869	0.0000	Signed Decima	

API	Інструкція				Операнд								опис			
1427		IOLINKW			Виконати ~ ErrCode								Запишіть параметр у пристрій ІО-Link			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Group								●					○	○		
Module								●					○	○		
Port								●					○	○		
Index								●					○	○		
SubIndex								●					○	○		
DataType								●					○	○		
WriteLen								●					○	○		
DataWrite								●								
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
Port		●				●							
Index		●				●							
SubIndex		●				●							
DataType		●				●							
WriteLen		●				●							
DataWrite		●				●							
Done	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

IOLINKW	
En	
Group	Done
Module	Error
Port	ErrCode
Index	
SubIndex	
DataType	
WriteLen	
DataWrite	

- Group** : Номер групи
- Module** : Номер модуля
- Port** : Номер комунікаційного порту
- Index** : Індексний номер параметра
- SubIndex** : Номер підіндексу параметра
- DataType** : Тип даних
- WriteLen** : Довжина записаних даних
- DataWrite** : Дані, які записуються
- Done** : Зв'язок завершений
- Error** : Прапор помилки
- ErrCode** : Код помилки

Пояснення

- Ця інструкція доступна для ПЛК з FW V1.08.50 або пізнішої версії та ISPSOft V3.12 або пізнішої версії.
- Щоб дізнатися більше про параметри пристрою, перегляньте файл IODD пристрою IO-Link. Ви також можете використовувати ці інструкції для запису параметрів у пристрій IO-Link.
- Немає обмежень щодо кількості виконання інструкції, але одночасно може бути виконана лише одна інструкція (IOLINKR або IOLINKW).
- Під час зв'язку пристрій IO-Link відповість не менше 5 секунд. НЕ зупиняйтеся і не починайте інший інструкції протягом цього часу. Після завершення зв'язку з IO-Link пропонується використовувати прапорці DONE або ERROR, щоб зупинити або запустити інструкцію.
- 5. Group** : номер групи AS04SIL, який підключено до правої сторони ПЛК безпосередньо або підключено до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль – це модуль номер 1, другий підключений модуль – номер 2. Можна підключити до 15 модулів. Якщо підключений модуль не AS04SIL, прапор помилки перемикається на ON.
- 6. Module** : номер модуля, який підключений до правої сторони ПЛК безпосередньо або підключений до віддаленого модуля, який діє як ПЛК, наприклад, перший підключений модуль має номер модуля 1, другий підключений модуль є номером модуля 2. Можна підключити та підрахувати до 32 модулів, і кожен тип модулів повинен бути включений у кількість. Якщо модуль не є AS04SIL, прапор помилки перемикається на ON.
- 7. Port** : номер порту зв'язку в діапазоні від 1 до 4. Якщо номер порту, який використовується, не є портом для AS04SIL, прапор помилки вмикається.
- 8. Index** — це номер індексу параметра, який призначено для запису в пристрій IO-Link.

9. SubIndex — це номер субіндексу параметра, який призначений для запису в пристрій IO-Link. Номер підіндексу можна встановити від 0 до 255. Якщо значення перевищує діапазон налаштувань, прапорець Error вмикається.

10. DataType — це тип даних параметра пристрою IO-Link. Вам потрібно встановити правильний тип даних перед виконанням цієї інструкції. Для отримання додаткової інформації зверніться до інструкції API1426 IOLINKR.

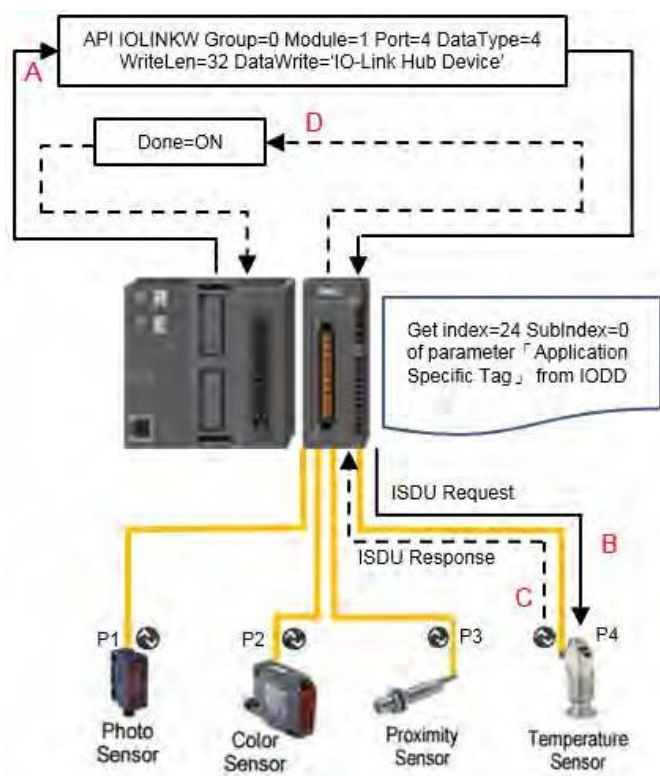
11. DataWrite — це дані, записані на пристрій IO-Link; одиниця: байт, упорядкований старшим і молодшим байтами.

12. WriteLen — це довжина даних, які будуть записані на пристрій IO-Link; одиниця: байт.

13. Done — це прапор завершення зв'язку. Якщо цей прапорець увімкнено, це означає, що параметри повністю записані в пристрій IO-Link.

14. Error — це позначка помилки зв'язку. Якщо цей прапорець увімкнено, це означає, що сталася помилка під час зв'язку між ЦП ПЛК, AS04SIL і пристроєм IO-Link. Зверніться до інструкції з **кодами помилок** API1426 IOLINKR для усунення несправностей.

15. Процес зв'язку між ЦП ПЛК, AS04SIL і пристроєм IO-Link:



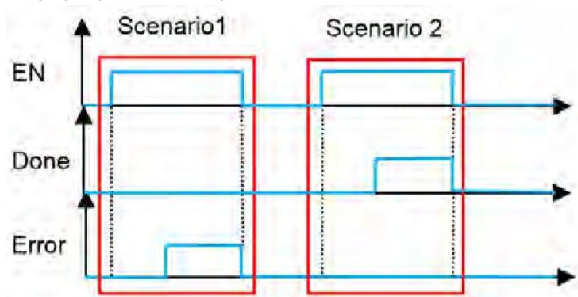
B: ЦП ПЛК надсилає дані на AS04SIL.

B: AS04SIL надсилає дані на пристрій IO-Link.

C: пристрій IO-Link обробляє запит і надсилає дані на AS04SIL.

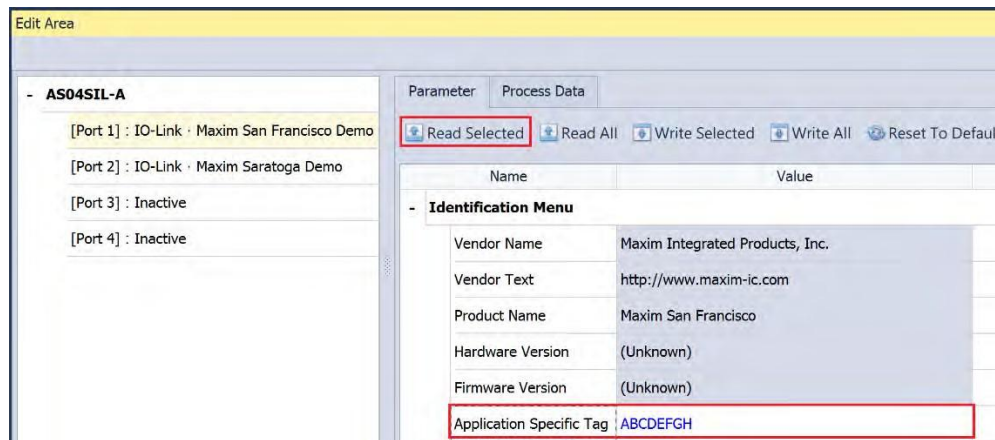
D: ЦП ПЛК оновлює статус (ГОТОВО / ERROR) і отримує дані.

16. Послідовність прапорів EN, Done та Error: коли виникає помилка під час зв'язку (сценарій 1) і коли зв'язок конкурує (сценарій 2).

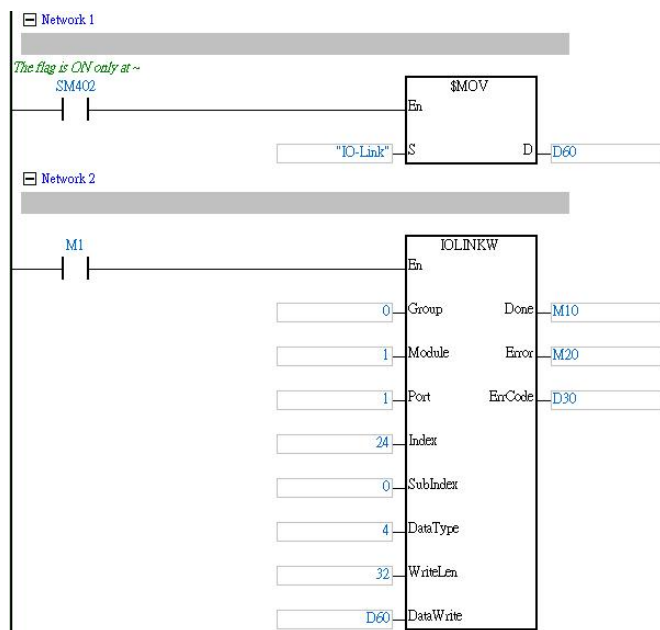


Приклад 1: Запишіть параметр спеціального тегу програми в перший пристрій першого порту зв'язку.

1. Перевірте HWCONFIG, щоб дізнатися, що значення параметра в спеціальному тегу програми – «ABCDEFGH».



- AS04SIL є першим модулем, встановленим на правій стороні ЦП ПЛК. Значення налаштування в **групі** становить 0,1 дюйм **Module** і 1 в **порту**.
- З файлу IODD або посібника з пристрою IO-Link ми дізналися, що його індекс дорівнює 24, субіндекс – 0, а тип даних – StringT. Встановіть значення налаштування в Index на 24, у Sub-index на 0, у типі даних на 4 (StringT).
- Використовуйте інструкцію \$MOV, щоб записати рядок «IO-Link» у пристрій даних, починаючи з D60.
- Почніть надсилати запит «на запис», коли M1 увімкнено. Коли M10 (ГОТОВО) увімкнено, це означає, що пристрій отримав відповідь і завершив запис. WriteLen становить 32, а рядок даних зберігається в DataWrite (D60~D75). Пристрій IO-Link записує дані в пристрої даних відповідно до порядку старшого та молодшого байтів.



Object	Identifiers	Device Name	Status	Data Type	Value (16bits)	Value (32bits)	Float	Radix	Comment...
		M1							
		M10							
		M20							
		D30			0000	00000000	0.0000	Hexadecimal	
		D60			4F49	4C2D4F49	45432100.0000	Hexadecimal	
		D61			4C2D	6E494C2D	15574644460718199940000	Hexadecimal	
		D62			6E49	006B6E49	0.0000	Hexadecimal	
		D63			006B	0000006B	0.0000	Hexadecimal	
		D64			0000	00000000	0.0000	Hexadecimal	
		D65			0000	00000000	0.0000	Hexadecimal	
		D66			0000	00000000	0.0000	Hexadecimal	
		D67			0000	00000000	0.0000	Hexadecimal	
		D68			0000	00000000	0.0000	Hexadecimal	
		D69			0000	00000000	0.0000	Hexadecimal	
		D70			0000	00000000	0.0000	Hexadecimal	
		D71			0000	00000000	0.0000	Hexadecimal	
		D72			0000	00000000	0.0000	Hexadecimal	
		D73			0000	00000000	0.0000	Hexadecimal	
		D74			0000	00000000	0.0000	Hexadecimal	

6. Перевірте HWCONFIG, щоб дізнатися, що значення параметра в тегу Application Specific Tag оновлено на «IO-Link».

AS04SIL-A

- [Port 1] : IO-Link , Maxim San Francisco Demo
- [Port 2] : IO-Link , Mojave (MAXREFDES42#)
- [Port 3] : Inactive
- [Port 4] : Inactive

Parameter Process Data

Read Selected
Read All
Write Selected
Write All
Reset To Default

Name	Value	Unit
Firmware Version	1.0	
Application Specific...	IO-Link	

Приклад 2: Запишіть параметр регулювання посилення зеленого каналу в перший пристрій другого комунікаційного порту.

1. Перевірте HWCONFIG, щоб дізнатися, що значення параметра зеленого каналу регулювання посилення становить «-63».

AS04SIL-A

- [Port 1] : IO-Link , Maxim San Francisco Demo
- [Port 2] : IO-Link , Maxim Saratoga Demo
- [Port 3] : Inactive
- [Port 4] : Inactive

Parameter Process Data

Read Selected
Read All
Write Selected
Write All
Reset To Default

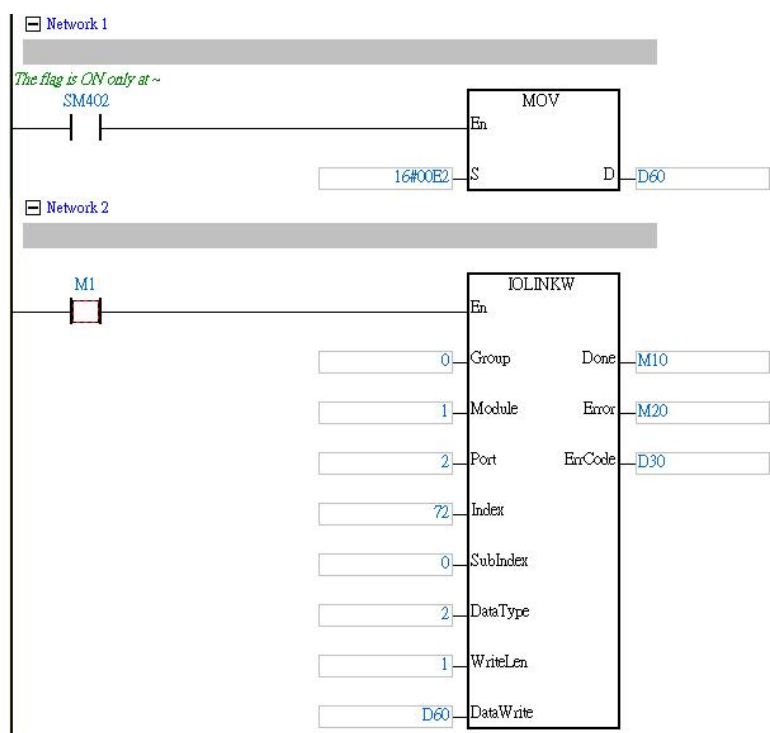
Name	Value	Unit
gain trim green cha...	-63	
gain trim IR channel	-2	

2. AS04SIL є першим модулем, встановленим на правій стороні ЦП ПЛК. Значення налаштування в групі становить 0, 1 у **модулі** та 2 у порту.

3. З його файлу IODD або посібника з пристрою IO-Link ми дізналися, що його індекс дорівнює 72, субіндекс – 0, а тип даних – IntegerT. Встановіть значення налаштування в Index на 72, у Sub-index на 0, у типі даних на 2 (IntegerT).

4. Використовуйте інструкцію \$MOV, щоб записати 0xE2 (-30) у пристрій даних, починаючи з D60.

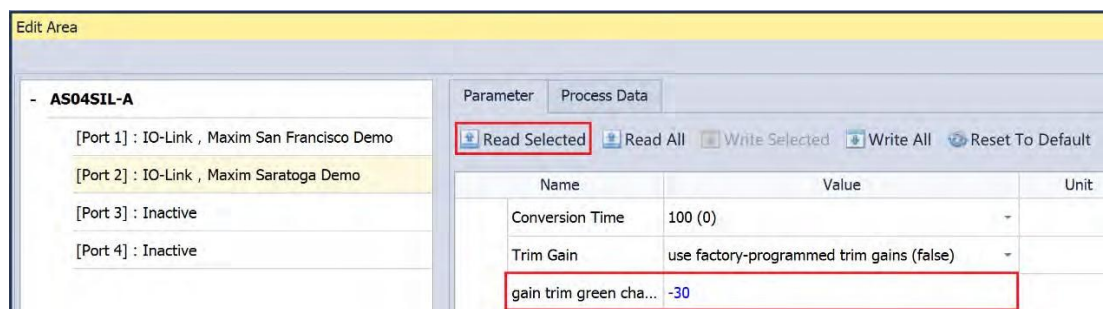
5. Почніть надсилати запит «на запис», коли M1 увімкнено. Коли M10 (ГОТОВО) увімкнено, це означає, що пристрій отримав відповідь і завершив запис. WriteLen дорівнює 1, а дані зберігаються в DataWrite (молодший байт D60). Пристрій IO-Link записує дані в пристрої даних відповідно до порядку старшого та молодшого байтів.



Object	Identifiers	Device Name	Status	Data Type	Value (16bits)	Value (32bits)	Float	Radix	Comment...
		M1							
		M10							
		M20							
		D30			0000	00000000	0.0000	Hexadecimal	
		D60			00E2	000000E2	0.0000	Hexadecimal	

7. Перевірте HWCONFIG, щоб дізнатися значення параметра підсилення зеленого каналу оновлено до «-30».

6



API	Код інструкції				Операнд				функція								
1430		MLPF			Group ~ ErrCode				Фільтр низьких частот для аналогових вхідних сигналів								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Group								●					○	○			
Module								●					○	○			
ChNo								●					○	○			
Fcutoff								●					○	○			
Error		●	●	●				●									
ErrCode								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Group		●				●							
Module		●				●							
ChNo		●				●							
Fcutoff		●				●							
Error	●												
ErrCode		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

MLPF	
En	
Group	Error
Module	ErrCode
ChNo	
Fcutoff	

- Group : Номер групи
- Module : Номер модуля
- ChNo : Номер каналу
- Error : Error частоти
- Fcutoff : зрізу
- ErrCode : Код помилки

Пояснення

1. Інструкція передбачає функцію цифрового фільтра низьких частот для аналогових вхідних сигналів. Цю інструкцію слід використовувати разом із ISPSOFT V3.16 або пізнішою.

Версії прошивки необхідних пристроїв:

Пункт	Модель	Прошивка	Зауваження
Функціональна карта AD	AS-F2AD	V1.40 або новішої версії	Прошивка ПЛК версії 1.14 або новішої
Модулі розширення	AS04AD-A, AS08AD-A, AS06XA-A	V1.02 або новішої версії	Версія мікропрограми ПЛК має бути V1.14 або пізніше. Крім того, мікропрограма підключеного ПЛК має бути версії 1.14 або новішої, мікропрограма для AS00SCM-A має бути версії 2.08 або новішої.
	AS02LC-A	V1.06 або новішої версії	
ПЛК із вбудованими каналами AD	AS218	Прошивка: V1.12 або новіша	

2. Після ввімкнення інструкції MLPF вхідне значення, що відповідає каналу, змінюється від початкового середнього значення до результату цифрового фільтра.
3. Для призначеного пристрою, якщо вказівку MLPF увімкнено для будь-якого з його каналів, час дискретизації всіх аналогових каналів буде вказано відповідно до їхніх обов'язкових значень, а початкове налаштування часу дискретизації буде недійсним. Якщо інструкцію MLPF вимкнено, вибірка буде проводитися відповідно до початкового налаштування часу вибірки.

Назва моделі	AS-F2AD	AS04AD-A, AS08AD-A, AS06XA-A	AS02LC-A	PLC вбудований AD
Час відбору проб вказується обов'язково	2 мс	2 мс	10 мс	2 мс

4. **Group** вказує номер групи для модулів аналогового введення, підключених до правої сторони ПЛК або віддалені модулі. Номер ЦП ПЛК дорівнює 0, номер першого віддаленого модуля дорівнює 1 і так далі. Максимальна кількість груп – 15 осіб.
5. **Module** вказує номер модуля. Якщо вхідний канал є аналоговим вхідним каналом, вбудованим у ЦП ПЛК, значення має бути встановлено на 0. Якщо вхідний канал є аналоговим вхідним каналом функціональної карти AS-F2AD, установіть операнд на 33 або 34 відповідно до слот функціональної карти, де розташований модуль.

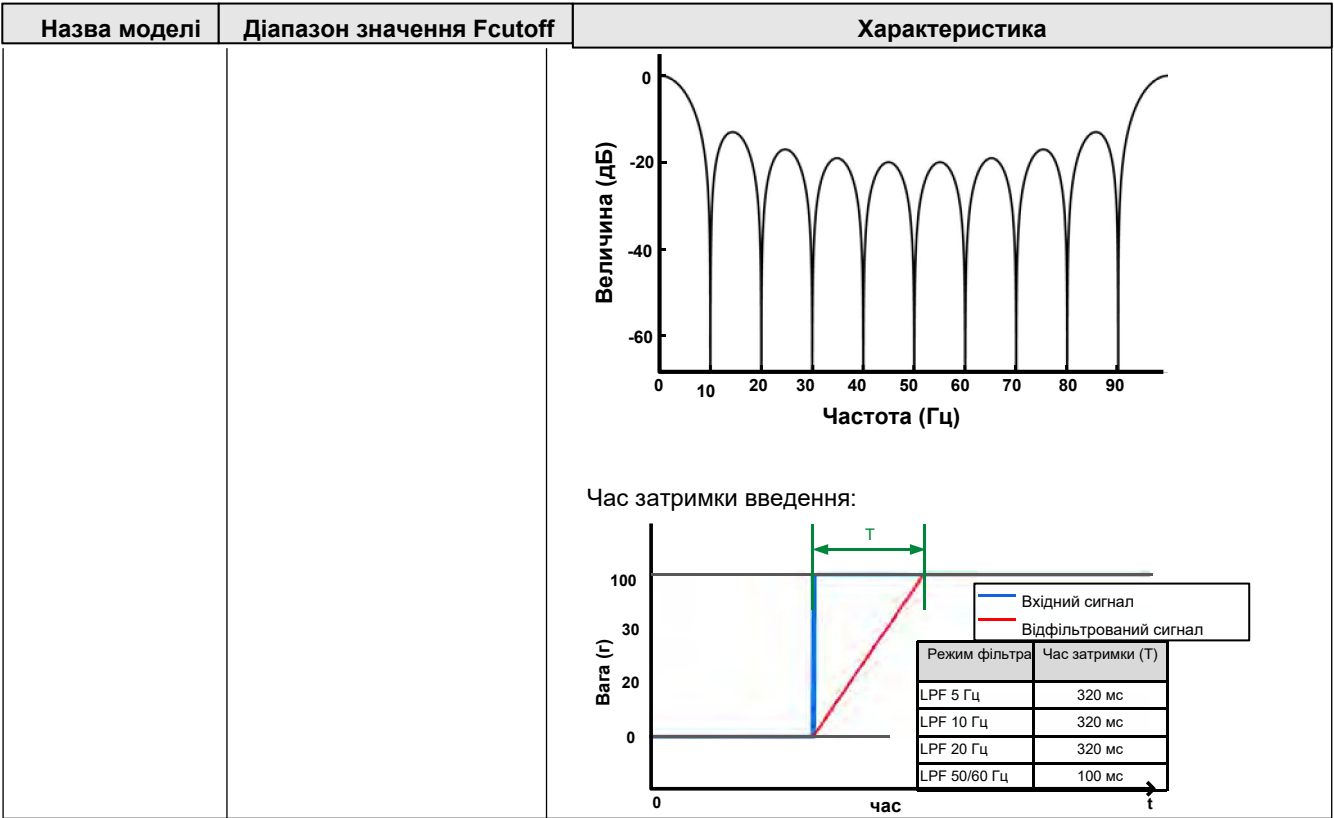
Module	опис
0	PLC вбудовані аналогові вхідні канали
1	1-го модуля розширення праворуч від ПЛК
2	2-го модуля розширення праворуч від ПЛК
3 ~ 32	Згідно з наведеним вище правилом розміщення, до правої сторони ПЛК можна підключити максимум 32 модулі розширення.
33	Слот функціональної карти 1
34	Слот для функціональної карти 2

6. **ChNo** вказує номер каналу для використання фільтра низьких частот.

Module	ChNo	опис
0	1	Аналоговий вхідний канал 1 вбудований в ЦП ПЛК
	2	Аналоговий вхідний канал 2 вбудований в ЦП ПЛК
1	1	Вхідний канал 1 1-го модуля розширення
	2 ~ 8	Вхідні канали 2 ~ 8 1-го модуля розширення
2	1	Вхідний канал 1 2-го модуля розширення
	2 ~ 8	Вхідні канали 2 ~ 8 2-го модуля розширення
3 ~ 32		Дотримуйтеся тих же правил, що й попередні, показані вище.
33	1	Аналоговий вхідний канал 1 AS-F2AD у слоті функціональної карти 1
33	2	Аналоговий вхідний канал 2 AS-F2AD у слоті функціональної карти 1
34	1	Аналоговий вхідний канал 1 AS-F2AD у гнізді функціональної карти 2
34	2	Аналоговий вхідний канал 2 AS-F2AD у гнізді функціональної карти 2

7. **Fcutoff** – частота зрізу фільтра низьких частот; діапазон налаштувань наступний:

Назва моделі	Діапазон значення Fcutoff	Характеристика
AS-F2AD, AS218 (з вбудовані канали AI/AO), AS04AD-A, AS08AD-B, AS08AD-C, AS06XA-A	5~100, одиниця: Гц	3 дБ (Співвідношення амплітуд: 0,707) Нахил: -40 дБ/декаду
AS02LC-A	0: 5 Гц 1: 10 Гц 2: 20 Гц 3: 50/60 Гц (Для ослаблення форми хвилі 50/60 Гц)	ФНЧ 5 Гц, 10 Гц і 20 Гц відповідають величинам: ФНЧ 5 Гц ФНЧ 10 Гц ФНЧ 20 Гц ФНЧ 50/60 Гц відповідає величині:

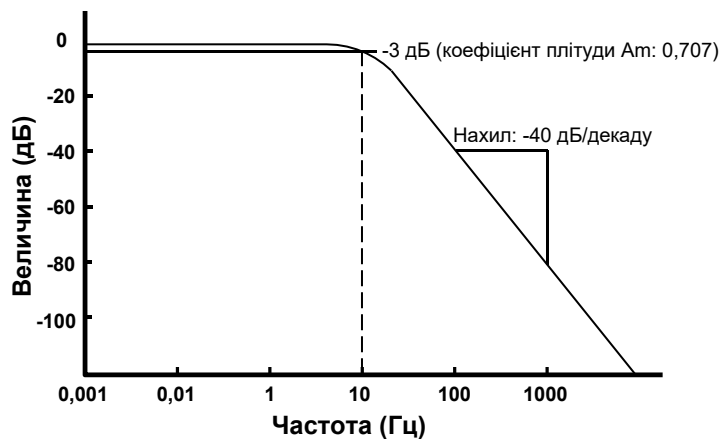


8. Перш ніж змінювати будь-які параметри під час виконання, вам потрібно спочатку вимкнути інструкцію MLPF. Після завершення модифікації увімкніть інструкцію знову.
9. Зверніться до наступної таблиці для опису кодів помилок (**ErrCode**), коли прапор помилки **Error** увімкнено.

Код помилки	опис
0x1400	Функція не підтримується пристроєм.
0x1401	Номер групи або номер модуля встановлено неправильно.
0x1402	Канал не існує в модулі.
0x1403	Fcutoff перевищує встановлений діапазон.
0x1404	Немає відповіді від модуля та тайм-аут зв'язку
0x1405	Для вказаного каналу ввімкнено функцію цифрового фільтра низьких частот. Повторне вказівку одного і того ж каналу заборонено.

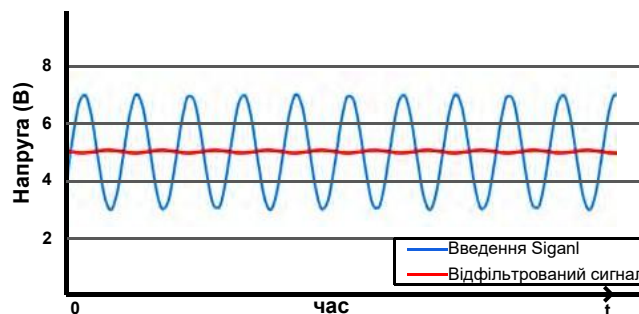
Приклад 1

1. Візьмемо для прикладу канал 1 вбудованого аналогового входу AS218, встановить **Group = 0, Module = 0, ChNo = 1, Fcutoff = 10 (Гц)**.
2. Частота цифрового фільтра нижніх частот відповідає величині наступним чином.



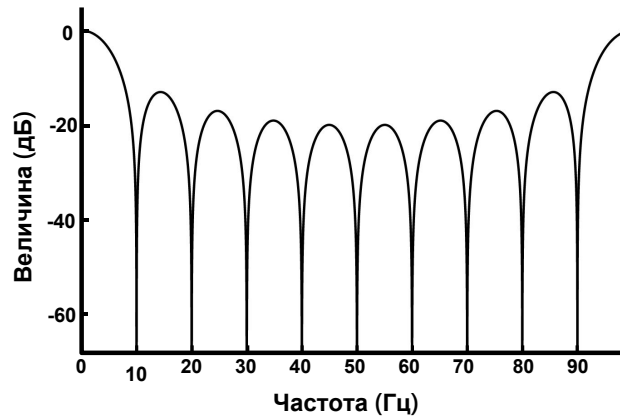
3. Введіть сигнал для перевірки:

Синусоїдальна хвиля 5 В постійного струму + 60 Гц має синю форму, як показано нижче (від піку до піку: 4 В). Після проходження сигналу через фільтр низьких частот на виході виходить сигнал червоного кольору. Можна побачити, що синусоїдальна хвиля 60 Гц послаблюється, і на кривій залишається постійний струм 5 В постійного струму.



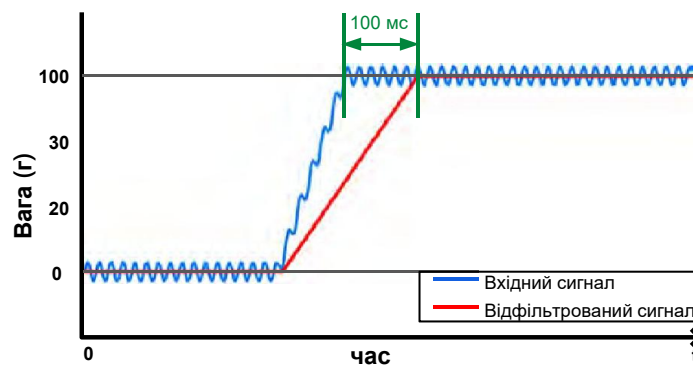
Приклад 2

1. Візьмемо, наприклад, вхідний канал 2 першого AS02LC-A з правого боку центрального процесора ПЛК, встановить **Group = 0, Module = 1, ChNo = 2, Fcutoff = 3 (50/60 Гц)**.
2. Частота цифрового фільтра нижніх частот відповідає величині наступним чином.



3. Введіть сигнал для перевірки:

Сигнал тензодатчика плюс синусоїдальний шум 50 Гц показано на синій формі сигналу нижче. Після того, як він проходить через фільтр низьких частот (ФНЧ), на виході виходить червоний сигнал, і можна побачити, що синусоїдальна хвиля 50 Гц значно ослаблена



6.16 Інструкції з числами з плаваючою комою

6.16.1 Список інструкцій для чисел з плаваючою комою

У наведеній нижче таблиці наведено інструкції для модуля, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1500	–	FSIN	✓	Синус числа з плаваючою комою
1501	–	FCOS	✓	Косинус числа з плаваючою комою
1502	–	FTAN	✓	Тангенс числа з плаваючою комою
1503	–	FASIN	✓	Арксинус числа з плаваючою комою
1504	–	FACOS	✓	Аркусинус числа з плаваючою комою
1505	–	FATAN	✓	Арктангенс числа з плаваючою комою
1506	–	FSINH	✓	Гіперболічний синус числа з плаваючою комою
1507	–	FCOSH	✓	Гіперболічний косинус числа з плаваючою комою
1508	–	FTANH	✓	Гіперболічний тангенс числа з плаваючою комою
1509	–	FRAD	✓	Перетворення градусів у радіани
1510	–	FDEG	✓	Перетворення радіанів у градуси
1511	SQR	DSQR	✓	Квадратний корінь з двійкового числа
1512	–	FSQR	✓	Квадратний корінь із числа з плаваючою комою
1513	–	FEXP	✓	Піднесення числа з плаваючою комою до степеня
1514	–	FLOG	✓	Логарифм числа з плаваючою комою
1515	–	FLN	✓	Натуральний логарифм двійкового числа з плаваючою комою
1516	–	FPOW	✓	Степінь числа з плаваючою комою
1517	RAND	–	✓	Генерація випадкового числа

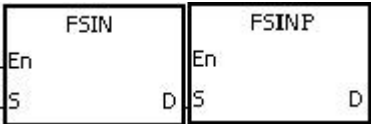
6.16.2 Пояснення інструкцій для чисел з плаваючою комою

API		Код інструкції			Операнд						функція						
1500			FSIN		P	S, D						Синус числа з плаваючою комою					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O					O	
D		●			●	●	●	●			O						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

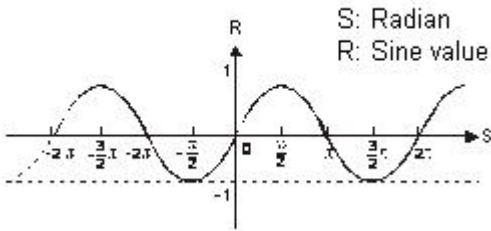
СИМВОЛ



S : Вихідне значення
D : Значення синуса

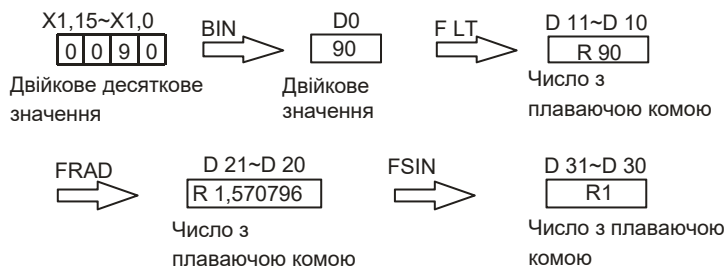
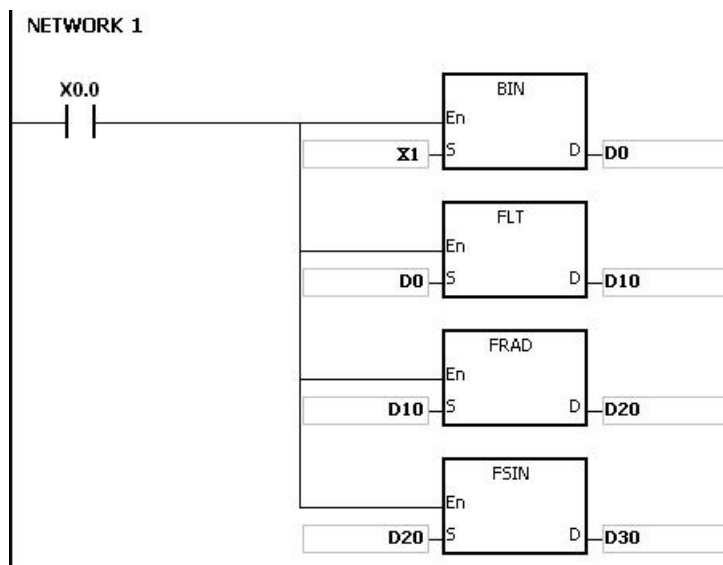
Пояснення

- Ця інструкція знаходить синус значення в **S** і зберігає його в **D**. Стан SM695 визначає, чи є вихідне значення в **S** в радіанах або градусах.
- Якщо SM695 вимкнено, вихідним значенням є **S** у радіанах
Радіан=градус×π/180.
- Якщо SM695 увімкнено, вихідне значення в **S** вказано в градусах.
Градус=Радіан×180/π. (0 ° ≤ градус ≤ 360 °)
- Якщо результат перетворення дорівнює нулю, SM600 увімкнено.
- На наступному графіку показано співвідношення між значеннями радіана та синуса.



приклад

Коли X0.0 увімкнено, інструкція BIN перетворює двійкове десяткове значення в X1.15–X1.0 у двійкове значення та зберігає результат перетворення в D0. Інструкція FLT перетворює двійкове значення в D0 на число з плаваючою комою та зберігає результат перетворення в (D11, D10). Інструкція FRAD перетворює число з плаваючою комою в (D11, D10) у радіани та зберігає результат перетворення в (D21, D20). Інструкція FSIN знаходить синус значення радіана в (D21, D20) і зберігає його в (D31, D30). Значення синуса є числом з плаваючою комою.



Додаткові зауваження

1. Якщо значення в S перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.
2. Якщо SM695 увімкнено, а значення в S не знаходиться в межах 0–360, інструкція не виконується, SM0 увімкнено, а код помилки — 16#2003.

API		Код інструкції			Операнд						функція					
1501			FCOS	P	S, D						Косинус числа з плаваючою комою					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O					O
D		●			●	●	●	●			O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу						16-бітна інструкція				32-розрядна інструкція			
AS						-				AS			

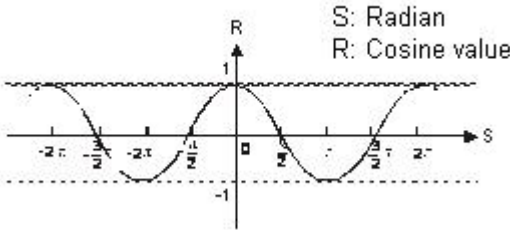
СИМВОЛ



S : Вихідне значення
D : Значення косинуса

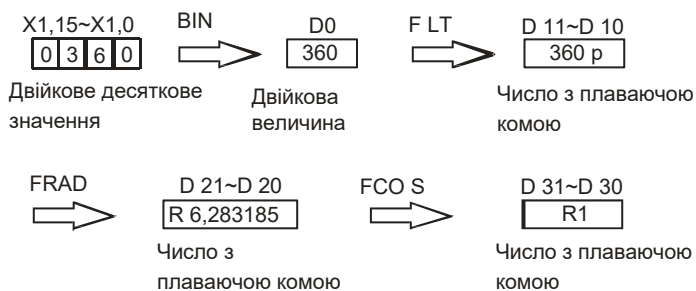
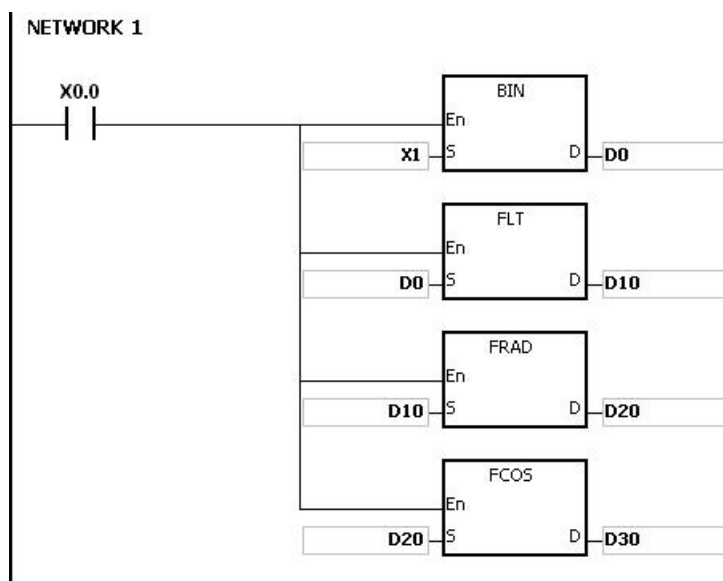
Пояснення

- Ця інструкція знаходить косинус значення в **S** і зберігає його в **D**. Стан SM695 визначає, чи є вихідне значення в **S** у радіанах або градусах.
- Якщо SM695 вимкнено, вихідне значення в **S** виражається в радіанах.
 $\text{Радіан} = \text{градус} \times \pi / 180$.
- Якщо SM695 увімкнено, вихідне значення в **S** вказано в градусах.
 $\text{Градус} = \text{Радіан} \times 180 / \pi$. ($0^\circ \leq \text{градус} \leq 360^\circ$)
- Якщо результат перетворення дорівнює нулю, SM600 увімкнено.
- На наступному графіку показано співвідношення між значеннями радіанів і косинусів.



приклад

Коли X0.0 увімкнено, інструкція BIN перетворює двійкове десяткове значення в X1.15–X1.0 у двійкове значення та зберігає результат перетворення в D0. Інструкція FLT перетворює двійкове значення в D0 на число з плаваючою комою та зберігає результат перетворення в (D11, D10). Інструкція FRAD перетворює число з плаваючою комою в (D11, D10) у радіани та зберігає результат перетворення в (D21, D20). Інструкція FCOS знаходить косинус значення радіана в (D21, D20) і зберігає його в (D31, D30). Значення косинуса є числом з плаваючою комою.



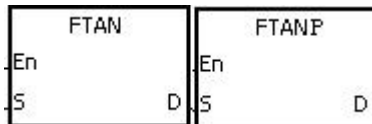
Додаткові зауваження

1. Якщо значення в S перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.
2. Якщо SM695 увімкнено, а значення в S не знаходиться в межах 1–360, інструкція не виконується, SM0 увімкнено, а код помилки — 16#2003.

API	Код інструкції				Операнд						функція					
1502		FTAN	P	S, D						Тангенс числа з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ**S** : Вихідне значення**D** : Значення тангенсу**Пояснення**

1. Ця інструкція знаходить тангенс значення в **S** і зберігає його в **D**. Стан SM695 визначає, чи є вихідне значення в **S** виражається в радіанах або градусах.

2. Якщо SM695 вимкнено, вихідне значення в **S** виражається в радіанах.

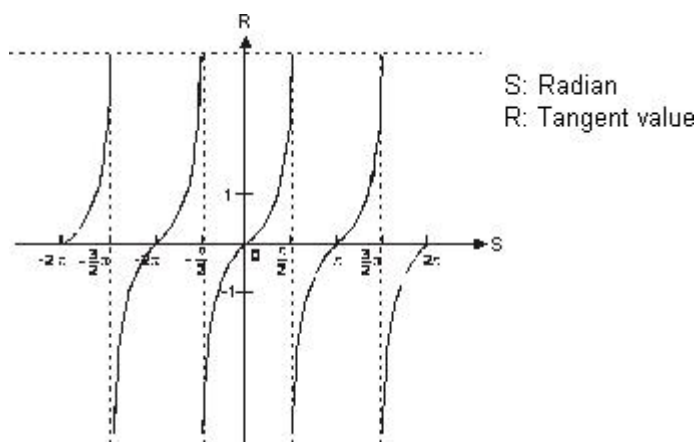
$$\text{Радіан} = \text{градус} \times \pi / 180.$$

3. Якщо SM695 увімкнено, вихідне значення в **S** вказано в градусах.

$$\text{Градус} = \text{Радіан} \times 180 / \pi. (0^\circ \leq \text{градус} \leq 360^\circ)$$

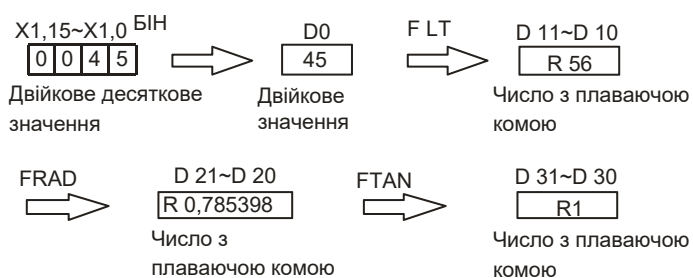
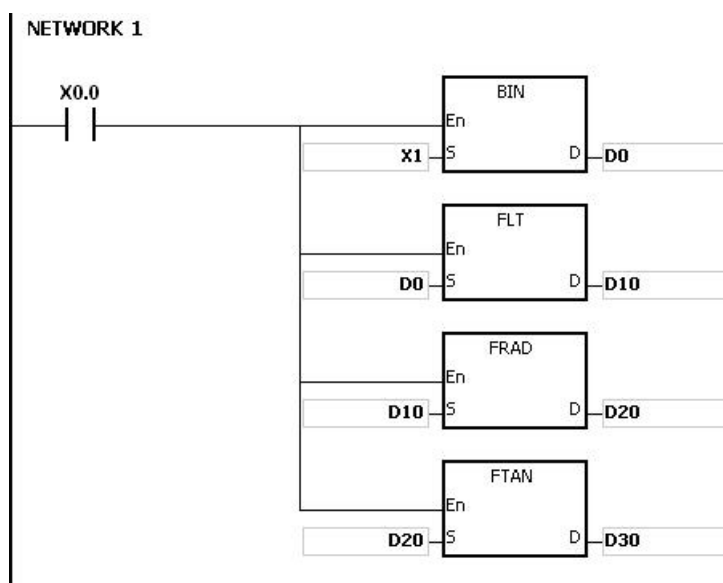
4. Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

5. На наступному графіку показано співвідношення між радіанами та значеннями тангенса.



приклад

Коли X0.0 увімкнено, інструкція BIN перетворює двійкове десяткове значення в X1.15–X1.0 у двійкове значення та зберігає результат перетворення в D0. Інструкція FLT перетворює двійкове значення в D0 на число з плаваючою комою та зберігає результат перетворення в (D11, D10). Інструкція FRAD перетворює число з плаваючою комою в (D11, D10) у радіани та зберігає результат перетворення в (D21, D20). Інструкція FTAN знаходить тангенс значення радіана в (D21, D20) і зберігає його в (D31, D30). Значення тангенса є числом з плаваючою комою.



Додаткові зауваження

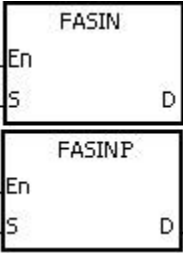
1. Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.
2. Якщо SM695 увімкнено, а значення в **S** не знаходиться в межах 0–360, інструкція не виконується, SM0 увімкнено, а код помилки — 16#2003.

API	Код інструкції				Операнд						функція					
1503		FASIN	P	S, D						Арксинус числа з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ



S: Вихідне значення

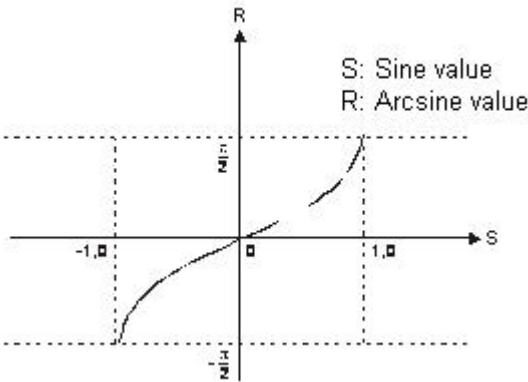
D: Значення арксинуса

6

Пояснення

1. Ця інструкція знаходить арксинус значення в S і зберігає його в D . Значення арксинуса= $\sin^{-1}$

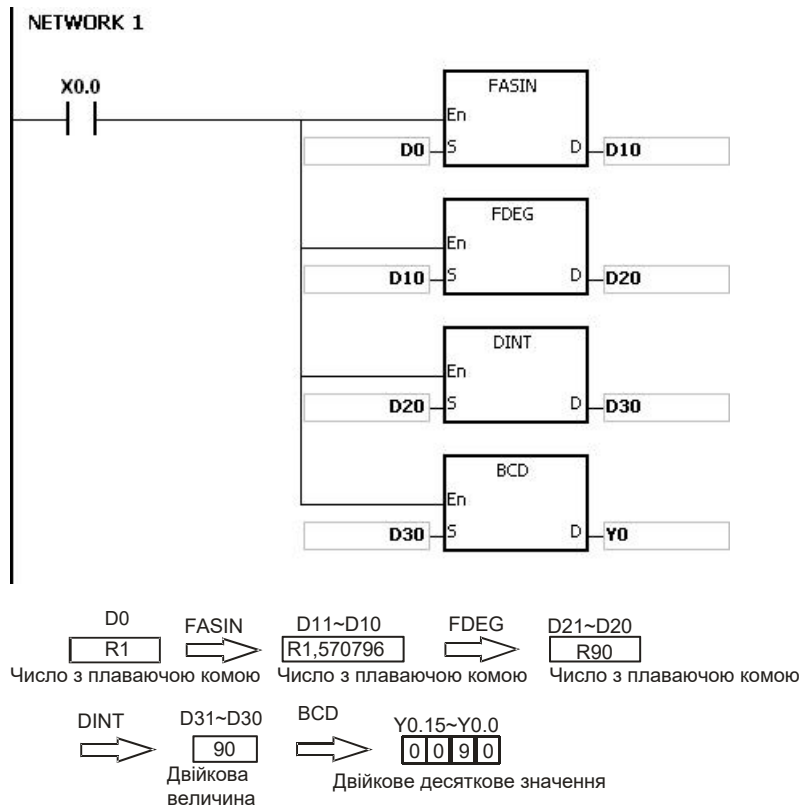
На наступному графіку показано співвідношення між значеннями синуса та арксинуса.



2. Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

приклад

Коли X0.0 увімкнено, інструкція знаходить арксинус числа з плаваючою комою в (D1, D0) і зберігає його в (D11, D10). Інструкція FDEG перетворює значення арксинуса в (D11, D10) у градуси та зберігає результат перетворення в (D21, D20). Інструкція DINT перетворює значення ступеня в (D21, D20) у ціле число та зберігає результат перетворення в (D31, D30). Інструкція BCD перетворює ціле число в (D31, D30) у двійкове десятикове значення та зберігає результат перетворення в Y0.15–Y0.0.



Додаткові зауваження

1. Число з плаваючою комою, визначене операндом **S**, має бути від –1,0 до 1,0. Якщо число з плаваючою комою не входить до цього діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції				Операнд						функція					
1504		FACOS	P	S, D						Аркосинус числа з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

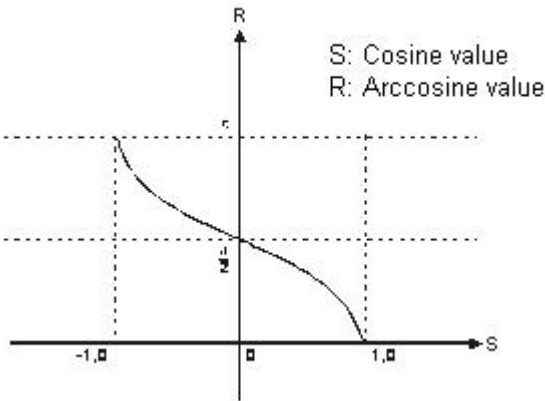
FACOS		FACOSP	
En		En	
S	D	S	D

- S : Вихідне значення
- D : Значення аркосинуса

Пояснення

1. Ця інструкція знаходить аркосинус значення в S і зберігає його в D . Значення аркосинуса=cos⁻¹

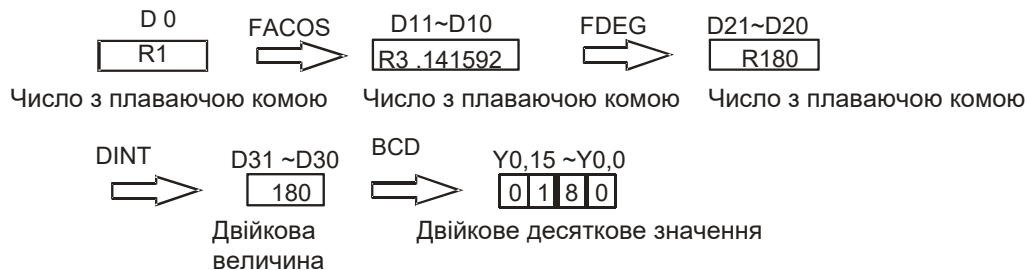
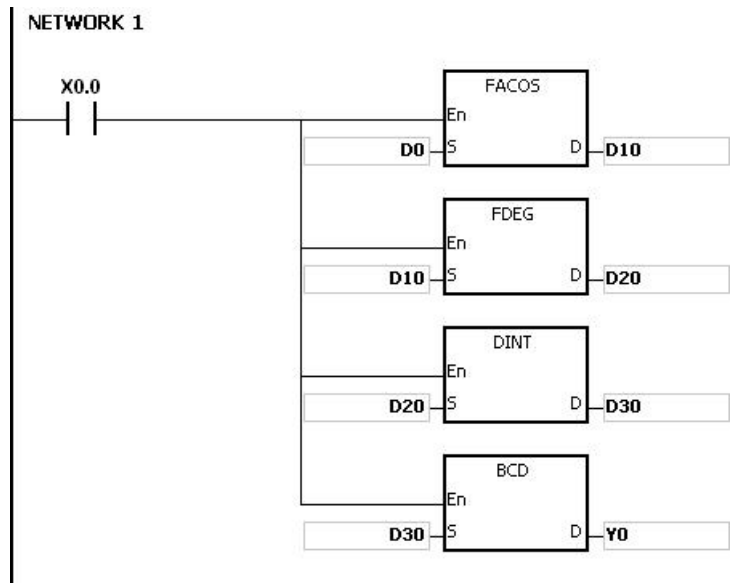
На наступному графіку показано співвідношення між значеннями косинуса та аркосинуса.



2. Якщо абсолютне значення результату перетворення перевищує значення, яке можна представити максимумом число з плаваючою комою, SM602 увімкнено.
3. Якщо абсолютне значення результату перетворення менше значення, яке може бути представлене мінімальним числом з плаваючою комою, SM601 увімкнено.
- 4., Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

приклад

Коли X0.0 увімкнено, інструкція FACOS знаходить аркосинус числа з плаваючою комою в (D1, D0) і зберігає його в (D11, D10). Інструкція FDEG перетворює значення аркосинуса в (D11, D10) у градуси та зберігає результат перетворення в (D21, D20). Інструкція DINT перетворює значення градусів у (D21, D20) у ціле число та зберігає результат перетворення в (D31, D30). Інструкція BCD перетворює ціле число в (D31, D30) у двійкове десятикове значення та зберігає результат перетворення в Y0.15–Y0.0.



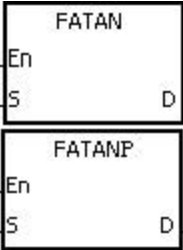
Додаткові зауваження

- 1. Число з плаваючою комою, визначене операндом **S**, має бути від –1,0 до 1,0. Якщо число з плаваючою комою не входить у діапазон, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
- 2. Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції			Операнд						функція						
1505		FATAN	P	S, D						Арктангенс числа з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O					O
D		●			●	●	●	●			O					
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING			
S										●						
D										●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

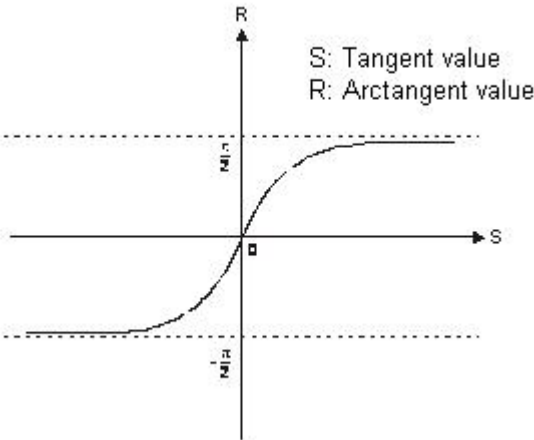


S : вихідне значення

D : значення арктангенса

Пояснення

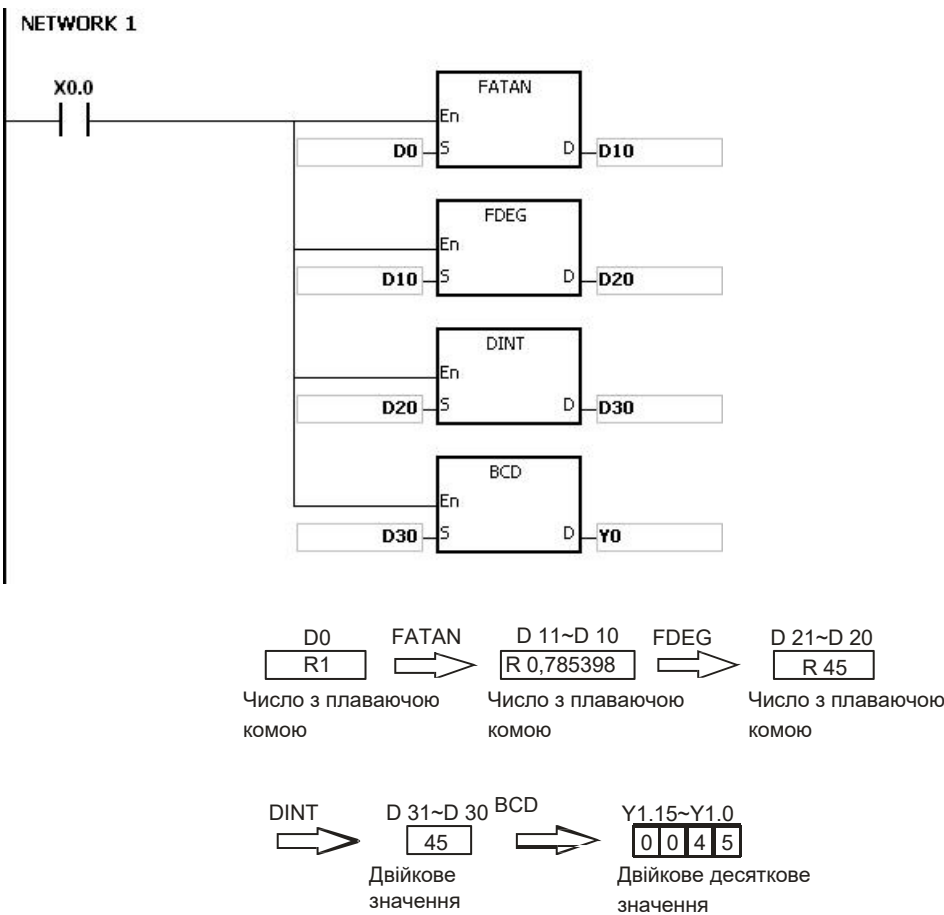
1. Ця інструкція знаходить арктангенс значення в S і зберігає його в D . Значення арктангенса=tan⁻¹
2. На наступному графіку показано співвідношення між значеннями тангенса та арктангенса.



3. Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

приклад

Коли X0.0 увімкнено, інструкція FATAN знаходить арктангенс числа з плаваючою комою в (D1, D0) і зберігає його в (D11, D10). Інструкція FDEG перетворює значення арктангенса в (D11, D10) на градуси та зберігає результат перетворення в (D21, D20). Інструкція DINT перетворює ступінь у (D21, D20) у ціле число та зберігає результат перетворення у (D31, D30). Інструкція BCD перетворює ціле число в (D31, D30) у двійкове десяткове значення та зберігає результат перетворення в Y0.15–Y0.0.



Додаткові зауваження

Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API		Код інструкції			Операнд				функція							
1506			FSINH	P	S, D				Гіперболічний синус числа з плаваючою комою							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O					O
D		●			●	●	●	●			O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу						16-бітна інструкція				32-розрядна інструкція			
AS						-				AS			

СИМВОЛ



S : вихідне значення

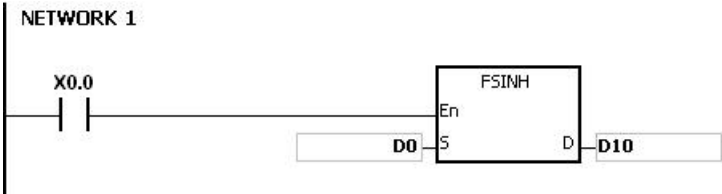
D : Гіперболічний синус

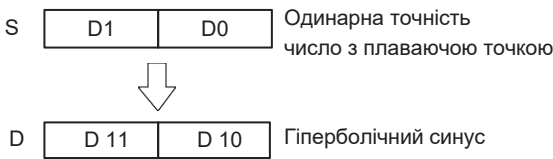
Пояснення

- Ця інструкція знаходить гіперболічний синус значення в **S** і зберігає його в **D**. Гіперболічне значення синуса= $(e^s - e^{-s})/2$.
- Якщо абсолютне значення результату перетворення перевищує значення, яке можна представити числами з плаваючою комою, значення в **D** дорівнює 16#7F800000, а SM602 увімкнено.
- Якщо абсолютне значення результату перетворення менше значення, яке можна представити числами з плаваючою комою, значення в **D** дорівнює 16#FF800000, а SM601 увімкнено.
- Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

приклад

- Коли X0.0 увімкнено, інструкція знаходить гіперболічний синус числа з плаваючою комою в (D1, D0) і зберігає його в (D11, D10). Гіперболічне значення синуса в (D11, D10) є числом із плаваючою комою.





- 2. Якщо абсолютне значення результату перетворення перевищує значення, яке можна представити за допомогою числа з плаваючою комою, SM602 увімкнено.
- 3. Якщо абсолютне значення результату перетворення менше значення, яке можна представити числами з плаваючою комою, SM601 увімкнено.
- 4. Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

Додаткові зауваження

Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API		Код інструкції			Операнд			функція								
1507			FCOSH	P	S, D			Гіперболічний косинус числа з плаваючою комою								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O					O
D		●			●	●	●	●			O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу						16-бітна інструкція				32-розрядна інструкція			
AS						-				AS			

СИМВОЛ



S: Вихідне значення

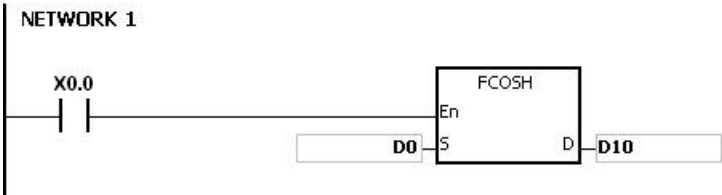
D: Гіперболічний косинус

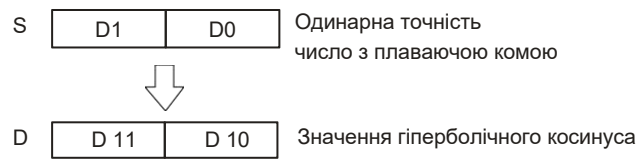
Пояснення

- Ця інструкція знаходить гіперболічний косинус значення в S і зберігає його в D . Гіперболічний косинус = $(e^s + e^{-s})/2$.
- Якщо абсолютне значення результату перетворення перевищує значення, яке можна представити числами з плаваючою комою, значення в D дорівнює 16#7F800000, а SM602 увімкнено.
- Якщо абсолютне значення результату перетворення менше значення, яке можна представити числами з плаваючою комою, значення в D дорівнює 16#FF800000, а SM601 увімкнено.
- Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

приклад

- Коли X0.0 увімкнено, інструкція знаходить гіперболічний косинус числа з плаваючою комою в (D1, D0) і зберігає його в (D11, D10). Значення гіперболічного косинуса в (D11, D10) є числом із плаваючою комою.





2. Якщо абсолютне значення результату перетворення перевищує значення, яке можна представити числами з плаваючою комою, SM602 увімкнено.
3. Якщо абсолютне значення результату перетворення менше значення, яке можна представити числами з плаваючою комою, SM601 увімкнено.
- 4,. Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

Додаткові зауваження

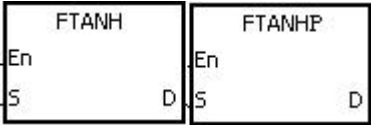
Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API		Код інструкції			Операнд						функція						
1508			FTANH	P	S, D						Гіперболічний тангенс числа з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O						O
D		●			●	●	●	●			O						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ



S : Вихідне значення
D : Значення гіперболічного тангенса

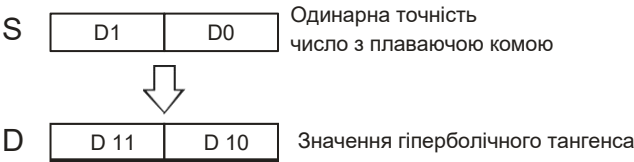
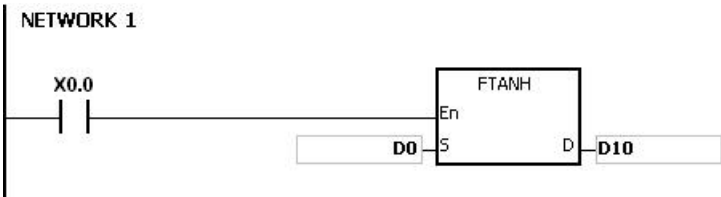
Пояснення

1. Ця інструкція знаходить гіперболічний тангенс значення в **S** і зберігає його в **D** .
Значення гіперболічного тангенса= $(e^S - e^{-S}) / (e^S + e^{-S})$.

2. Якщо результат перетворення дорівнює 0, SM600 увімкнено.

приклад

1. Коли X0.0 увімкнено, інструкція знаходить гіперболічний тангенс числа з плаваючою комою в (D1, D0) і зберігає його в (D11, D10). Значення гіперболічного тангенса в (D11, D10) є числом з плаваючою комою.



2. Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

Додаткові зауваження

Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції				Операнд						функція						
1509		FRAD	P		S, D						Перетворення градусів у радіани						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O						O
D		●			●	●	●	●			O						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ



S : Вихідне значення (у градусах)

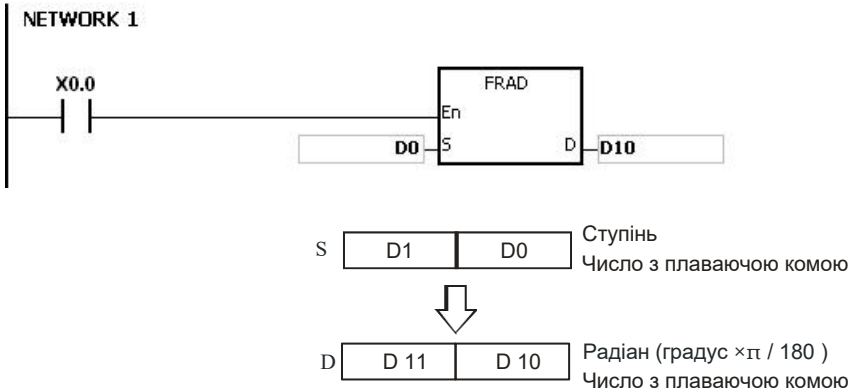
D : Результат перетворення (у радіанах)

Пояснення

- 1. Ця інструкція перетворює значення градусів у **S** у радіани та зберігає його в **D**.
- 2. Радіан = градус × (π/180).
- 3. Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

приклад

Коли X0.0 увімкнено, інструкція перетворює значення градуса в (D1, D0) на значення в радіанах і зберігає результат перетворення в (D11, D10). Радіан у (D11, D10) є числом із плаваючою комою.



Додаткові зауваження

Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції				Операнд						функція					
1510		FDEG	P	S, D						Перетворення радіанів у градуси						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ



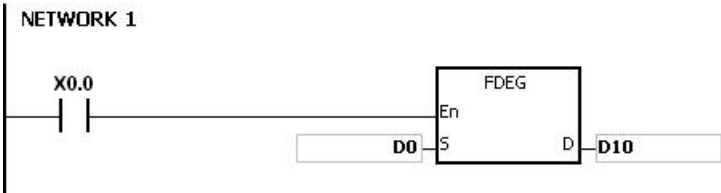
S : Вихідне значення (у радіанах)
D : Результат перерахунку (у градусах)

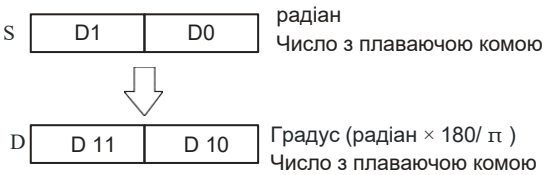
Пояснення

- Ця інструкція перетворює значення радіанів у **S** у градуси та зберігає його в **D** .
- Градус = Радіан × (180/π).
- Якщо абсолютне значення результату перетворення перевищує значення, яке можна представити числами з плаваючою комою, значення в D дорівнює 16#7F7FFFFF.
- Якщо абсолютне значення результату перетворення менше значення, яке можна представити за допомогою числа з плаваючою комою чисел, значення в **D** дорівнює 16#7F7FFFFF.
- Якщо результат перетворення дорівнює нулю, SM600 увімкнено.

приклад

Коли X0.0 увімкнено, інструкція перетворює радіанні значення в (D1, D0) на значення градусів і зберігає перетворення результат (D11, D10). Степінь у (D11, D10) є числом із плаваючою комою.





Додаткові зауваження

Якщо значення в **S** перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції				Операнд						функція						
1511	D	SQR	P		S, D						Квадратний корінь з двійкового числа						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O	O	O	O			
D		●			●	●	●	●			O	O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●				●	●	
		●	●			●	●				●	●	

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



S: Вихідне значення

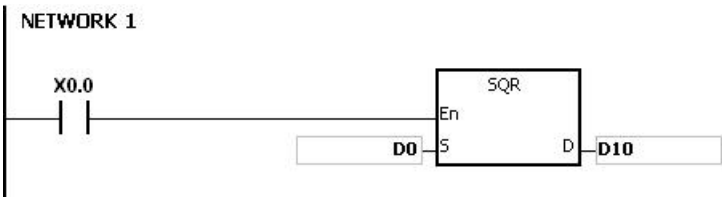
D: Пристрій, де зберігається результат

Пояснення

- Ця інструкція обчислює квадратний корінь із значення в **S** і зберігає результат у пристрої, визначеному **D**.
- Результат операції, що зберігається в **D**, є цілим числом. Якщо число з плаваючою комою округлено до найближчої цілої цифри, SM601 увімкнено.
- Якщо результат операції, збережений у **D**, дорівнює 0, SM600 увімкнено.
- Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

приклад

Коли X0.0 увімкнено, інструкція обчислює квадратний корінь із значення в D0 і зберігає результат у D10.



Додаткові зауваження

значенням може бути лише значення **S**. Якщо значення в **S** є від'ємним значенням, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції				Операнд						функція						
1512		FSQR	P		S, D						Квадратний корінь із числа з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O						O
D		●			●	●	●	●			O						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ



S : Вихідне значення

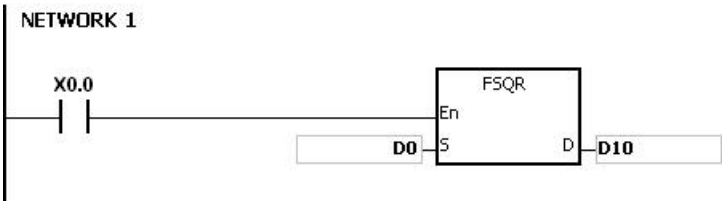
D : Пристрій, де зберігається результат

Пояснення

- Ця інструкція обчислює квадратний корінь із числа з плаваючою комою в **S** і зберігає результат у пристрої, визначеному **D**.
- Якщо результат операції, збережений у **D**, дорівнює 0, SM600 увімкнено.

Приклад 1

Коли X0.0 увімкнено, інструкція обчислює квадратний корінь із числа з плаваючою комою в (D1, D0) і зберігає результат в (D11, D10).



Додаткові зауваження

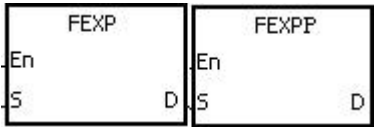
значенням може бути лише значення **S**. Якщо значення в **S** є від'ємним значенням, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції				Операнд				функція							
1513		FEXP	P		S, D				Піднесення числа з плаваючою комою до степеня							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O					O
D		●			●	●	●	●			O					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ



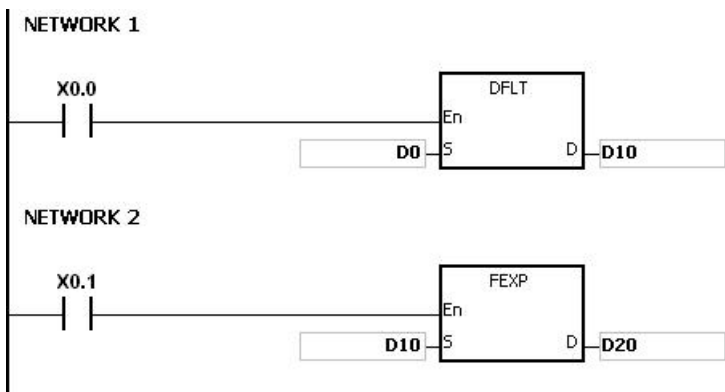
- S**: Вихідне значення (експонента)
- D**: Пристрій, де зберігається результат операції

Пояснення

- Ця функція обчислює значення числа e , зведеного до степеня **S**. Піднесення до степеня включає два числа, основа e , яка представляє 2,71828, і показник степеня в **S**.
- $EXP[D + 1, D] = [S + 1, S]$.
- Число в **S** може бути позитивним або негативним числом. Пристрій, визначений **D**, має бути 32-розрядним регістр, а число в пристрої, визначене **S**, має бути числом з плаваючою комою.
- Значення в регістрі, визначене **D**, є e^S (e дорівнює 2,71828, а **S** представляє вихідне значення).
- Якщо абсолютне значення результату перетворення перевищує значення, яке можна представити числами з плаваючою комою, значення в регістрі, визначеному **D**, є 16#7F800000, а SM602 увімкнено.
- Якщо результат операції, збережений у **D**, дорівнює 0, SM600 увімкнено.

приклад

- Коли X0.0 увімкнено, інструкція DFLT перетворює значення в (D1, D0) на число з плаваючою комою та зберігає результат перетворення в (D11, D10).
- Коли X0.1 увімкнено, інструкція FEXP виконує піднесення до степеня зі значенням у (D11, D10) і зберігає результат числа з плаваючою комою (D21, D20).



API	Код інструкції				Операнд						функція					
1514		FLOG	P	S ₁ , S ₂ , D						Логарифм числа з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●	●	●	●		○					○
S ₂	●	●			●	●	●	●	●		○					○
D		●			●	●	●	●			○					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

FLOG				FLOGP			
En				En			
S1		D		S1		D	
S2				S2			

S_1 : Основа для логарифма

S_2 : Вихідне значення

D : Пристрій, де зберігається результат операції

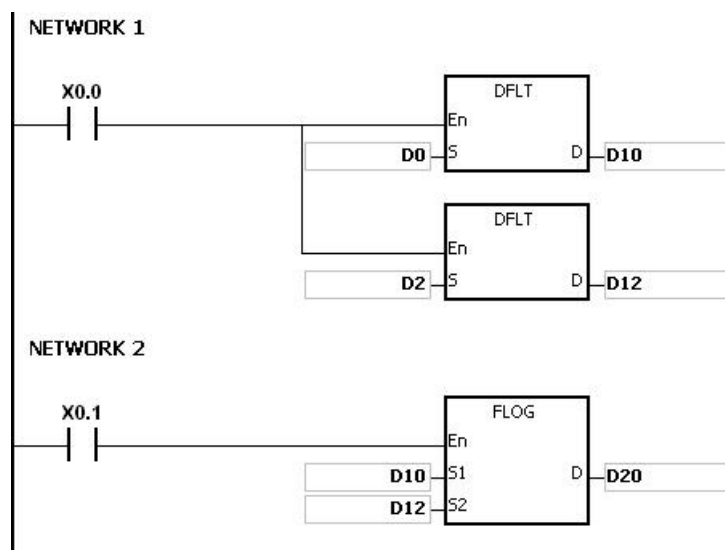
Пояснення

- Ця інструкція обчислює логарифм значення в S_2 відносно значення в S_1 і зберігає точний результат операції з плаваючою комою в D .
- Значення в S_1 і S_2 можуть бути лише позитивними значеннями.
- $S_1 \neq 0 \rightarrow D = \log_{S_1} S_2$.
- Приклад: припустимо, що значення в S_1 і S_2 дорівнюють 5 і 125 відповідно. Знайдіть логарифм $_5 125$ (логарифм з основою 5 числа 125).
- $S_1 \neq 0 \rightarrow 5 \neq 125 \rightarrow D = \log_5 125 = 3$.
- Якщо результат операції, збережений у D , дорівнює 0, SM600 увімкнено.

приклад

- Коли X0.0 увімкнено, інструкція DFLT перетворює значення в (D1, D0) і (D3, D2) на числа з плаваючою комою та зберігає результати перетворення в (D11, D10) і (D13, D12) відповідно .
- Коли X0.1 увімкнено, інструкція FLOG обчислює логарифм числа з плаваючою комою в (D13, D12) за допомогою

відносно числа з плаваючою комою в (D11, D10) і зберігає результат операції в (D21, D20).



Додаткові зауваження

Якщо значення в S_1 менше або дорівнює 1, або якщо значення в S_2 менше або дорівнює 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції				Операнд					функція							
1515		FLN	P		S, D					Натуральний логарифм двійкового числа з плаваючою комою							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O					O	
D		●			●	●	●	●			O						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ



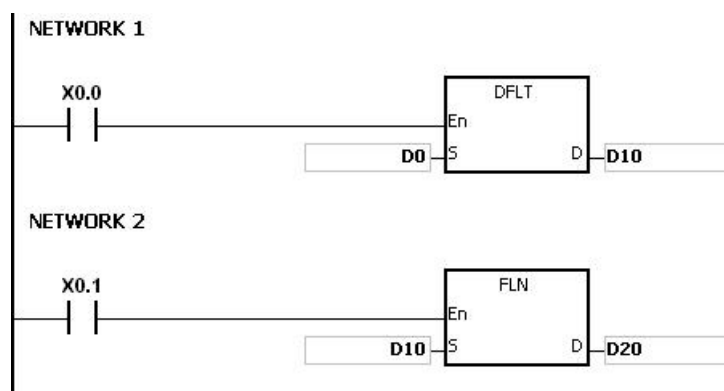
- S:** Вихідне значення
- D:** Пристрій, де зберігається результат операції

Пояснення

- Ця інструкція обчислює натуральний логарифм операнда **S** в операції з плаваючою комою одинарної точності.
- значенням може бути лише значення **S**.
- $e^D = S \rightarrow$ Значення в **D** $= \ln S$.
- Якщо результат операції, збережений у **D**, дорівнює 0, SM600 увімкнено.

приклад

- Коли X0.0 увімкнено, інструкція DFLT перетворює значення в (D1, D0) на число з плаваючою комою та зберігає результат перетворення в (D11, D10).
- Коли X0.1 увімкнено, інструкція FLN обчислює натуральний логарифм числа з плаваючою комою в (D11, D10) і зберігає результат операції в (D21, D20).



Додаткові зауваження

Якщо значення в S менше або дорівнює 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд						функція							
1516		FPOW	P	S_1, S_2, D						Степінь числа з плаваючою комою							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●	●	●	●							O
S_2	●	●			●	●	●	●	●							O
D		●			●	●	●	●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2									●				
D									●				

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

FPOW	FPOWP
En	En
S1	S1
S2	S2

 S_1 : Вихідне значення S_2 : Показник ступеня

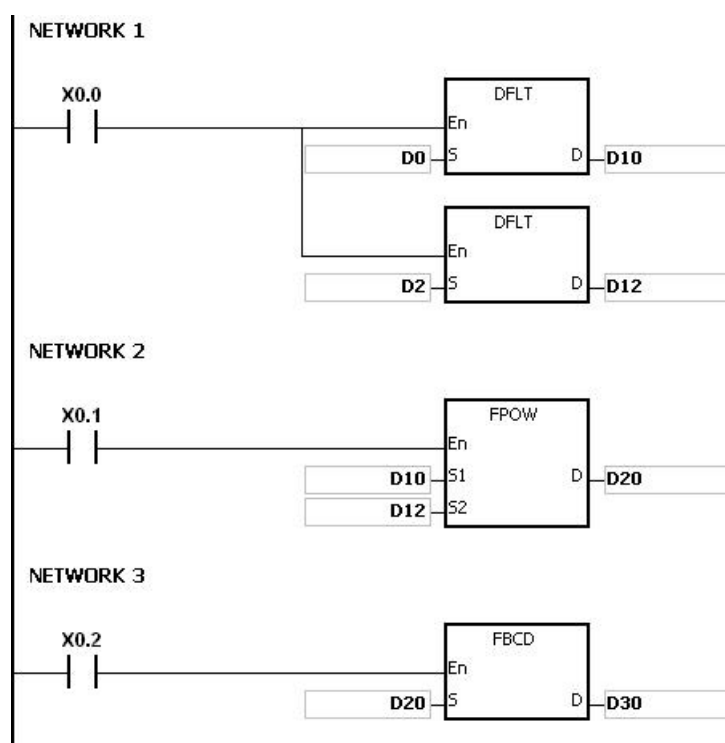
D : Пристрій, де зберігається результат операції

Пояснення

- Ця інструкція підвищує число одинарної точності з плаваючою комою в S_1 до ступеня значення в S_2 і зберігає результат операції з плаваючою комою одинарної точності в D.
- $D = \text{POW}[S_1 + 1 \cdot S_1][S_2 + 1 \cdot S_2]$
- Для ПЛК із мікропрограмою V1.10.00 або попередніми версіями значення в S_1 може бути лише позитивним числом, але значення в S_2 може бути як позитивним, так і негативним числом.
- Для ПЛК із мікропрограмою V1.10.10 або пізніших версій, якщо значення в S_1 є додатним числом, значення в S_2 може бути додатне або від'ємне число. Якщо значення в S_1 є від'ємним числом, значення в S_2 має бути цілим числом або додатним, або від'ємним.
- Припустимо, що значення в S_1 і S_2 дорівнюють 5 і 3 відповідно: $D = 5^3 = 125$.
- Якщо абсолютне значення результату операції перевищує значення, яке можна представити числами з плаваючою комою, значення в D дорівнює 16#7F7FFFFF, а SM602 увімкнено.
- Якщо абсолютне значення результату операції менше значення, яке можна представити числами з плаваючою комою, значення в D дорівнює 16#FF800000, а SM601 увімкнено.
- Якщо результат операції, збережений у D, дорівнює 0, SM600 увімкнено.

приклад

1. Коли X0.0 увімкнено, інструкція DFLT перетворює значення в (D1, D0) і (D3, D2) на числа з плаваючою комою, і зберігає результати перетворення в (D11, D10) і (D13, D12) відповідно.
2. Коли X0.1 увімкнено, інструкція FPOW підносить число з плаваючою комою в (D11, D10) до ступеня числа з плаваючою комою в (D13, D12) і зберігає результат операції в (D21, D20).
3. Коли X0.2 увімкнено, інструкція FBCD перетворює двійкове число з плаваючою комою в (D21, D20) у двійкове десяткове число з плаваючою комою та зберігає результат перетворення в (D31, D30).



Додаткові зауваження

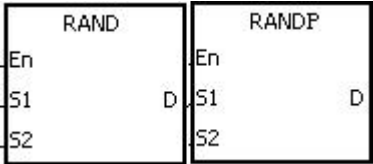
1. Для ПЛК із мікропрограмою V1.10.00 або попередніми версіями, якщо значення в S_1 менше 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Для ПЛК із мікропрограмою V1.10.10 або пізніших версій, якщо значення в S_1 менше 0, а значення в S_2 НЕ є цілим числом, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 є 16#2003.
3. Якщо значення в S_1 дорівнює 0, а значення в S_2 менше або дорівнює 0, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції				Операнд						функція					
1517		RAND		P	S ₁ , S ₂ , D						Генерація випадкового числа					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●		○	○	○	○		
S ₂	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



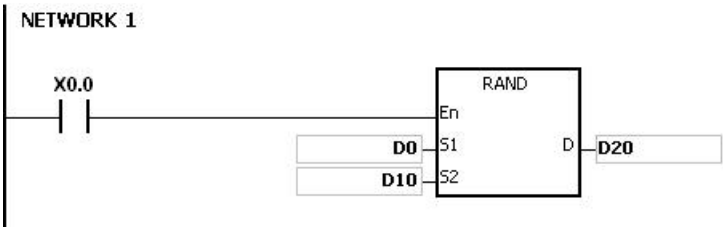
- S_1 : Мінімальна вартість
- S_2 : Максимальне значення
- D : Пристрій, де зберігається результат

Пояснення

- Ця інструкція генерує випадкове число між мінімальним значенням в S_1 і максимальним значенням в S_2 , і потім зберігає результат у D.
- Якщо значення в S_1 більше, ніж значення в S_2 , інструкція приймає значення в S_1 і S_2 як максимальне значення і мінімальне значення відповідно.

приклад

Коли X0.0 увімкнено, інструкція генерує випадкове число між мінімальним значенням у D0 та максимальним значенням у D10 і зберігає результат у D20.



Додаткові зауваження

Значення в S_1 і S_2 мають бути між 0–32767. Якщо значення в S_1 або S_2 перевищує діапазон, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

6.17 Інструкції годинника реального часу

6.17.1 Список інструкцій годинника реального часу

У наведеній нижче таблиці наведено інструкції щодо Годинника реального часу, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1600	TRD	—	✓	Читання часу
1601	TWR	—	✓	Написання часу
1602	T+	—	✓	Додавання часу
1603	T-	—	✓	Віднімання часу
1604	HOUR	—	—	Лічильник часу роботи
1605	TCMP	—	✓	Порівняння часу
1606	TZCP	—	✓	Порівняння часових поясів
1607	DST	—	✓	Літній час
1608	WON	—	—	Налаштування щотижневого робочого часу

6.17.2 Пояснення інструкцій годинника реального часу

API	Код інструкції			Операнд								функція					
1600		TRD	P	D								Читання часу					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D					●	●		●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



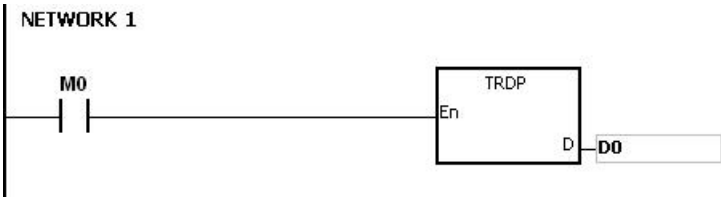
D : Пристрій, де зберігається результат

Пояснення

- 1. Ця інструкція зчитує поточний час із вбудованого годинника реального часу в модулі ЦП і зберігає поточний час в D.
- 2. Операнд D займає сім послідовних пристроїв.
- 3. Вбудований годинник реального часу показує рік, тиждень, місяць, день, хвилину та секунду. Дані зберігаються в SR391–SR397.
- 4. Дві останні цифри номера року для AD зберігаються в SR391.

приклад

Коли M0 увімкнено, інструкція зчитує поточний час із годинника реального часу в D0–D6. Значення 1 у SR397 означає понеділок, значення 2 — вівторок, а значення 7 — неділю.



Особливі дані регістр	Пункт	Значення		Загальні дані регістр	Пункт
SR391	Рік (AD)	00-99	→	D0	Рік (AD)
SR392	місяць	1-12	→	D1	місяць
SR393	День	1-31	→	D2	День
SR394	годину	0-23	→	D3	годину
SR395	хвилина	0-59	→	D4	хвилина
SR396	секунда	0-59	→	D5	секунда
SR397	тиждень	1-7	→	D6	тиждень

Додаткові зауваження

1. Якщо **D +6** перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Коли SM220 увімкнено, годинник реального часу калібрується в межах ± 30 секунд. Якщо значення секунд, зчитане з годинника реального часу, становить 0–29, інструкція очищає значення секунд до нуля. Якщо значення секунд, зчитане з годинника реального часу, становить 30–59, інструкція збільшує значення хвилини на одиницю та очищає значення секунд до нуля.

API	Код інструкції				Операнд						функція						
1601		TWR	P		S						Запис часу						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S					●	●		●	●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



S : Джерело даних

Пояснення

1. Ця інструкція регулює вбудований годинник реального часу в модулі ЦП, записуючи правильний поточний час у **S** у вбудований годинник реального часу.
2. Операнд **S** займає сім послідовних пристроїв. (Використовуйте 24-годинний формат часу)
3. Інструкція миттєво записує новий встановлений час у годинник реального часу в ПЛК.
4. Переконайтеся, що під час виконання інструкції новий час налаштування в **S** відповідає фактичному часу.
5. Пропонується використовувати його як інструкцію пульсу. Якщо контакт нормально розімкнутий, то виконується вказівка на постійний запис часу. Але ПЛК записує час лише під час першого сканування. Якщо потрібно оновити вбудований годинник реального часу, ви можете закрити контакт на час сканування, а потім виконати цю інструкцію знову, щоб оновити годинник.

приклад

Коли M0 увімкнено, інструкція записує правильний поточний час у вбудований годинник реального часу в ПЛК.



Новий час схоплювання	Загальні дані регістр	Пункт	Значення	→	Особливі дані регістр	Пункт	Годинник реального часу
	D20	Рік (AD)	00-99	→	SR391	Рік (AD)	
	D21	місяць	1-12	→	SR392	місяць	
	D22	День	1-31	→	SR393	День	
	D23	годину	0-23	→	SR394	годину	
	D24	хвилина	0-59	→	SR395	хвилина	
	D25	секунда	0-59	→	SR396	секунда	
	D26	тиждень	1-7	→	SR397	тиждень	

Додаткові зауваження

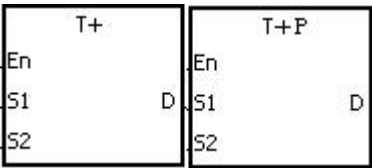
1. Якщо значення в **S** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM увімкнено, а код помилки в SR — 16#2003.
2. Якщо **S** +6 перевищує діапазон пристрою, виникає помилка операції, інструкція не виконується, SM увімкнено, а код помилки в SR — 16#2003.
3. Якщо ви оголошуєте операнд **S** в ISPSOft, типом даних буде ARRAY [7] WORD/INT.

API	Код інструкції				Операнд						функція						
1602		T+	P		S ₁ , S ₂ , D						Додавання часу						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁					●	●		●	●		O						
S ₂					●	●		●	●		O						
D					●	●		●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●				●							
S ₂		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



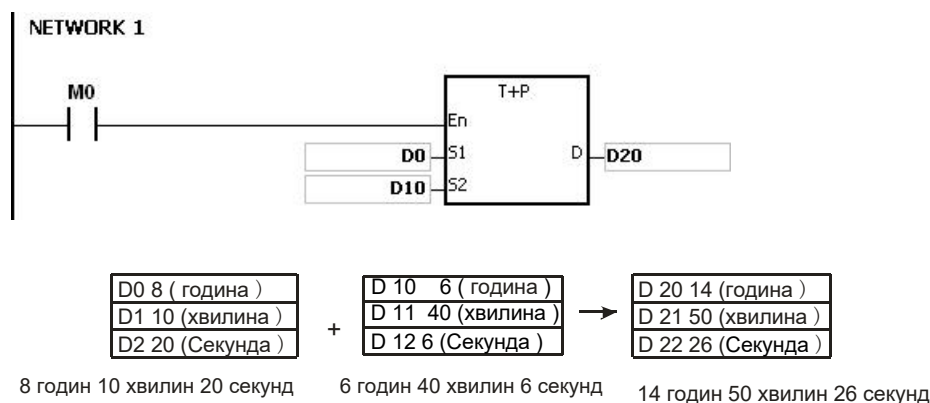
- S₁ : Пристрій-джерело 1
- S₂ : Пристрій-джерело 2
- D : Пристрій, де зберігається результат

Пояснення

- Ця інструкція додає значення години, хвилини та секунди в годиннику реального часу, визначеному S₂, до значення години, хвилини та секунди в годиннику реального часу, визначеному S₁, а потім зберігає суму в регістрі, визначеному D.
- Кожен з операндів S1, S2 і D займає три послідовних пристрої.
- Якщо сума більша або дорівнює 24 годинам, SM602 увімкнено, і інструкція віднімає 24 години із суми перед збереженням результату в D.
- Якщо сума дорівнює нулю (нуль годин нуль хвилин нуль секунд), SM600 увімкнено.

приклад

Коли M0 увімкнено, інструкція додає значення години, хвилини та секунди в D10 – D12 до значення години, хвилини та секунди в D0 – D2 і зберігає суму в D20 – D22.



Додаткові зауваження

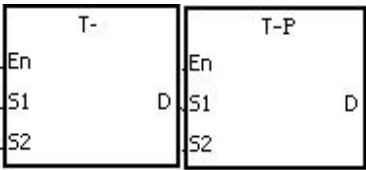
1. Якщо значення в **S₁** або **S₂** перевищує діапазон, виникає помилка операції, інструкція не виконується, **SM0** увімкнено, а код помилки в **SR0** — 16#2003.
2. Якщо **S₁ + 2**, **S₂ + 2** або **D + 2** перевищує діапазон пристрою, виникає помилка операції, інструкція не виконується, **SM0** увімкнено, а код помилки в **SR0** — 16#2003.
3. Якщо ви оголошуєте операнд **S₁** в **ISPSoft**, типом даних буде **ARRAY [3] WORD/INT**.
4. Якщо ви оголошуєте операнд **S₂** в **ISPSoft**, типом даних є **ARRAY [3] з WORD/IN**.
5. Якщо ви оголошуєте операнд **D** в **ISPSoft**, типом даних є **ARRAY [3] з WORD/INT**.

API	Код інструкції				Операнд						функція					
1603		T-	P	S ₁ , S ₂ , D						Віднімання часу						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁					●	●		●	●		O					
S ₂					●	●		●	●		O					
D					●	●		●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●				●							
S ₂		●				●							
D		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
AS		AS		-	

СИМВОЛ



- S_1 : Пристрій-джерело 1
- S_2 : Пристрій-джерело 2
- D : Пристрій, де зберігається результат

Пояснення

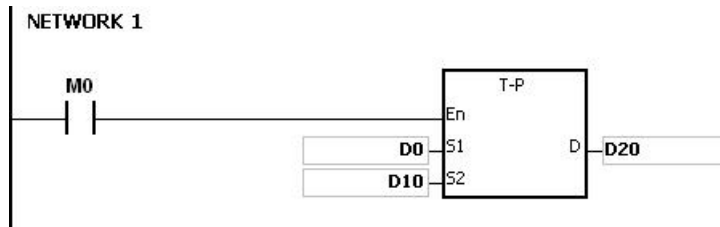
- Ця інструкція віднімає значення години, хвилини та секунди в годиннику реального часу, визначеному S_2 від значення години, хвилини та секунди в годиннику реального часу, визначеному S_1 , і зберігає різницю в регістрі, визначеному D .
- операнди S_1 , S_2 і D займають три послідовних пристрої.
- Якщо різниця негативна, SM601 увімкнено, і інструкція додає 24 години до різниці, а потім зберігає результат у D .
- Якщо різниця дорівнює нулю (нуль годин нуль хвилин нуль секунд), SM600 увімкнено.

приклад

- Коли M0 увімкнено, інструкція віднімає значення години, хвилини та секунди в D10–D12 із значення години, хвилини та секунди в D0–D2 та зберігає різницю в D20 –D22.



2. Якщо різниця негативна, SM601 увімкнено.



Додаткові зауваження

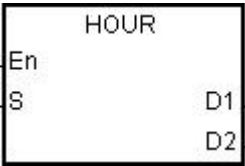
1. Якщо значення в S_1 або S_2 перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо $S_1 + 2$, $S_2 + 2$ або $D + 2$ перевищує діапазон пристрою, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо ви оголошуєте операнд S_1 в ISPSOft, типом даних буде ARRAY [3] WORD/INT.
4. Якщо ви оголошуєте операнд S_2 в ISPSOft, типом даних буде ARRAY [3] WORD/INT.
5. Якщо ви оголошуєте операнд D в ISPSOft, типом даних є ARRAY [3] з WORD/INT.

API	Код інструкції				Операнд						функція						
1604		HOUR			S, D ₁ , D ₂						Лічильник часу роботи						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S					●	●		●	●		O	O	O	O			
D ₁								●									
D ₂		●	●	●				●		O							

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D ₁		●				●							
D ₂	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	

СИМВОЛ



- S** : Час, через який вихід пристрій увімкнено
- D₁** : Поточний час
- D₂** : Вивідний пристрій

Пояснення

1. Ця інструкція перемикає вихідний пристрій, визначений **D₂**, у положення ON через час у **S**.

2. S : Час, після якого вихідний пристрій увімкнено (Одиниця вимірювання: Години)

Операнд **S**, який використовується в 16-бітній інструкції, має бути між 1–32 767.

3. **D₁** : поточний час (одиниця: година). Значення в **D₁** має бути між 0–32 767.

D₁ +1: поточний час, менший за одну годину (одиниця: секунди). Значення в **D₁ +1** має бути між 0–3599.

D₁ +2 лише для системного використання. Значення в ньому не можна змінити під час виконання інструкції; інакше виникає помилка.

Коли поточний час становить 32 767 годин 3 599 секунд, таймер припиняє відлік. Після того, як значення в **D₁** і **D₁ +1** скидаються до 0, таймер починає відлік знову.

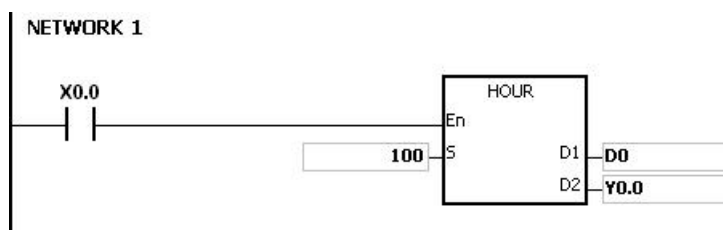
4. Коли час, протягом якого вхідний контакт був увімкненим, досягає встановленого часу в **S**, вихідний пристрій увімкнено. До цього вихідний пристрій не увімкнено. Ця функція дозволяє керувати часом роботи машини та обслуговуванням.

5. Після увімкнення вихідного пристрою таймер продовжує відлік.

6. Під час редагування в режимі онлайн скиньте умовний контакт, щоб ініціалізувати інструкцію.

Приклад 1

Коли X0.0 увімкнено, таймер інструкцій починає відлік. Коли час, протягом якого X0.0 був увімкнено, досягає 100 годин, Y0.0 увімкнено. Поточний час записується в D0, а поточний час, який менше однієї години, записується в D1. D2 призначений для системного використання. Значення в ньому не можна змінити; інакше виникає помилка.



Додаткові зауваження

1. Коли **S** менше або дорівнює 0, інструкція не виконується, а стан пристрою виведення не змінюється.
2. Якщо значення в **D₁** менше 0, стан пристрою виведення не змінюється.
3. Якщо **D₁ + 2** перевищує діапазон пристрою, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
4. Якщо ви оголошуєте операнд **D₁** в ISPSOft, типом даних є ARRAY [3] з WORD/INT.

API	Код інструкції			Операнд						функція						
1605		TCMP	P	S_1, S_2, S_3, S, D						Порівняння часу						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2	●	●			●	●		●	●		○	○	○	○		
S_3	●	●			●	●		●	●		○	○	○	○		
S					●	●		●								
D		●	●	●				●		○						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S		●				●							
D	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

TCMP	TCMPP
En	En
S1	S1 D
S2	S2
S3	S3
S	S

- S_1 : Година для часу захоплення
- S_2 : Хвилина для часу захоплення
- S_3 : Другий для часу захоплення
- S : Поточний час
- D : Результат порівняння

Пояснення

- Ця інструкція порівнює значення години, хвилини та секунди, визначені $S_1 - S_3$, зі значенням годину, хвилину та секунду в пристроях, починаючи з пристрою, визначеного S , і зберігає результат порівняння в D .
- Година поточного часу знаходиться в пристрої, визначеному S , а значення години між 0 – 23. Хвилина поточний час знаходиться в пристрої, визначеному $S + 1$, а значення хвилин має бути в межах 0 – 59. Секунда поточного часу знаходиться в пристрої, зазначеному $S + 2$, і значення секунди має бути від 0 до 59.
- Операнд D займає три послідовних пристроїв. Результат порівняння зберігається в D , $D + 1$ і $D + 2$.
- Загалом, спочатку використовуйте інструкцію TRD (API 1600), щоб зчитати поточний час із годинника реального часу, а потім використовуйте інструкцію TCMP, щоб порівняти час.
- Якщо час налаштування в $S_1 - S_3$ більший за поточний час у S , D увімкнено, $D + 1$ вимкнено, а $D + 2$ вимкнено.

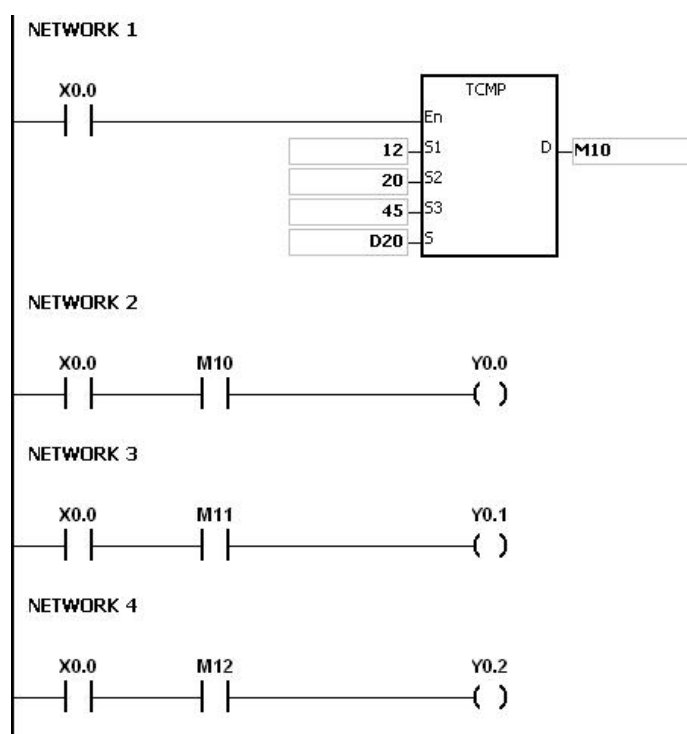
6. Якщо встановлений час у $S_1 - S_3$ дорівнює поточному часу в S , D вимкнено, $D + 1$ увімкнено, а $D + 2$ вимкнено.

7. Якщо час налаштування в $S_1 - S_3$ менший за поточний час у S , D буде OFF, $D + 1$ OFF, а $D + 2$ ВІМКНЕНО.

приклад

1. Коли X0.0 увімкнено, інструкція порівнює встановлений час 12 годин 20 хвилин 45 секунд із поточним часом у D20 – D22 і зберігає результат порівняння в M10 – M12. Коли X0.0 перемикається з ON на OFF, інструкція не виконується, а стани M10, M11 і M12 залишаються такими ж, якими вони були до того, як X0.0 перемикається на ON.

2. Якщо ви хочете отримати результат порівняння $\geq$, $\leq$ або $\neq$, ви можете з'єднати M10–M12 послідовно або паралельно.



Додаткові зауваження

1. Якщо $S + 2$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

2. Якщо $D + 2$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

3. Якщо значення в S перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

4. Якщо значення в $S_1 - S_3$ перевищують діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

5. Якщо ви оголошуєте операнд **S** в ISPSoft, типом даних є ARRAY [3] WORD.
6. Якщо ви оголошуєте операнд **D** в ISPSoft, типом даних буде ARRAY [3] BOOL.

API	Код інструкції			Операнд							функція						
1606		TZCP	P	S_1, S_2, S, D							Порівняння часових поясів						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1					●	●		●	●							
S_2					●	●		●	●							
S					●	●		●								
D		●	●	●				●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S		●				●							
D	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

TZCP		TZCPP	
En		En	
S1	D	S1	D
S2		S2	
S		S	

- S_1 : Нижня межа часу
 S_2 : Верхній граничний час
 S : Поточний час
 D : Результат порівняння

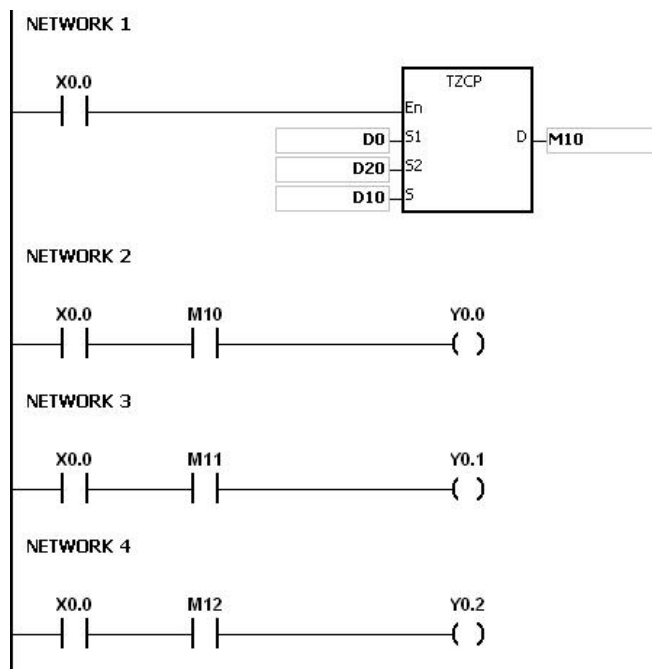
Пояснення

- Ця інструкція порівнює поточний час, визначений S , з нижнім граничним часом, визначеним S_1 , і з верхнім граничним часом, визначеним S_2 , і зберігає результат порівняння в D .
- Година нижнього граничного часу знаходиться в пристрої, визначеному S_1 , хвилина нижнього граничного часу – у пристрої, визначеному $S_1 + 1$, а секунда нижнього граничного часу – у пристрої, визначеному $S_1 + 2$.
- Година верхнього граничного часу знаходиться в пристрої, зазначеному S_2 , хвилина верхнього граничного часу - в пристрої, зазначеному $S_2 + 1$, а секунда верхнього граничного часу - в пристрої, зазначеному $S_2 + 2$.
- Година поточного часу знаходиться в пристрої, визначеному S , хвилина поточного часу – у пристрої, визначеному $S + 1$, а секунда поточного часу – у пристрої, визначеному $S + 2$.
- Час у пристрої, визначеному S_1 , повинен бути меншим за час у пристрої, визначеному S_2 . Якщо час у пристрої, визначеному S_1 , більший за час у пристрої, визначеному S_2 , інструкція приймає час у пристрої, визначений S_1 , як верхній граничний час під час виконання інструкції.
- Загалом, спочатку використовуйте інструкцію TRD (API 1600), щоб прочитати поточний час із годинника реального часу, а потім використовуйте інструкції TZCP для порівняння часу.

7. Якщо поточний час у пристрої, визначеному S_1 , менший ніж нижня межа часу в пристрої, визначеному S_2 , D увімкнено. Якщо поточний час у пристрої, визначеному S_1 , перевищує нижній граничний час у пристрої, визначеному S_2 , $D+2$ увімкнено; інакше $D+1$ увімкнено.

приклад

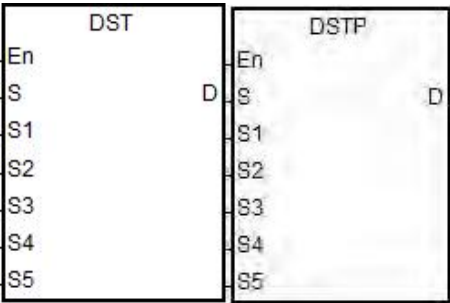
Коли X0.0 увімкнено, виконується інструкція TZCP. M10, M11 або M12 увімкнено. Коли X0.0 вимкнено, інструкція не виконується, стан M10, стан M11 і стан M12 залишаються такими ж, як і до того, як X0.0 перейшов у положення ON.



Додаткові зауваження

1. Якщо $S_1 + 2$, $S_2 + 2$, $S + 2$ або $D + 2$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо значення в S_1 , S_2 і S перевищують діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003
3. Якщо ви оголошуєте операнд S_1 в ISPSOft, типом даних буде ARRAY [3] WORD/INT.
4. Якщо ви оголошуєте операнд S_2 в ISPSOft, типом даних буде ARRAY [3] WORD/INT.
5. Якщо ви оголошуєте операнд S в ISPSOft, типом даних є ARRAY [3] із WORD/INT.
6. Якщо ви оголошуєте операнд D в ISPSOft, типом даних буде ARRAY [3] BOOL.

API	Код інструкції				Операнд							функція					
1607		DST	P		S, S ₁ , S ₂ , S ₃ , S ₄ , S ₅ , D							Літній час					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S					●	●		●					O	O			
S ₁					●	●		●					O	O			
S ₂					●	●		●					O	O			
S ₃					●	●		●					O	O			
S ₄					●	●		●					O	O			
S ₅					●	●		●					O	O			
D		●	●	●				●									
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S		●				●											
S ₁		●				●											
S ₂		●				●											
S ₃		●				●											
S ₄		●				●											
S ₅		●				●											
D	●																
				Інструкція пульсу				16-бітна інструкція				32-розрядна інструкція					
СИМВОЛ				AS				AS				-					



- S : Коды функций летнего времени
- S₁ : Месяц начала летнего времени
- S₂ : Дата начала летнего времени
- S₃ : Месяц окончания летнего времени
- S₄ : Дата окончания летнего времени
- S₅ : Изменение через летнее время (минут)
- D : Статус функции летнего времени

Пояснення

1. Операнди, які використовуються в цій інструкції,
описані нижче: S: функціональні коди літнього часу

Версія мікропрограми до V1.04 (за винятком V1.04) підтримує такі коди функцій:

Коди функцій	опис
0	Вимкнути літній час
1	Увімкніть режим літнього часу 1
2	Прочитайте літній час
3~	Зарезервовано або переглядається як зчитування літнього часу

Версія мікропрограми після V1.04 (включно з V1.04) підтримує такі коди функцій:

Коди функцій	опис
0	Вимкнути літній час
1	Увімкніть режим літнього часу 1
2	Прочитайте літній час
3	Увімкніть режим літнього часу 1
4	Вимкнути літній час (задається системою)
5	Увімкнено режим літнього часу 1 (встановлюється системою)
7	Увімкнено режим літнього часу 2 (задається системою)
6, 8~	Зарезервовано або переглядається як зчитування літнього часу

Примітка 1: коли код у S дорівнює 4, 5 або 7, виконання інструкції не має користі.

Примітка 2. Додаткову інформацію про різні режими читайте в наступних розділах.

S₁: налаштування місяця для переходу на літній час

S₂: встановлення дати переходу на літній час

S =1 (увімкнено режим переходу на літній час 1), **S₂**: налаштування дати початку переходу на літній час

S =3 (увімкнено режим переходу на літній час 2), **S₂**: налаштування тижня для початку переходу на літній час, **S₂**
+1: день тижня з **S₂**

S₃: налаштування місяця для завершення літнього часу

S₄: налаштування дати завершення літнього часу

S =1 (увімкнено режим переходу на літній час 1), **S₄**: налаштування дати завершення переходу на літній час

$S = 3$ (режим переходу на літній час 2 увімкнено), S_4 : налаштування тижня, до якого завершується літній час,
 $S_2 + 1$: день тижня з S_4

S_5 : налаштування для зміни через літній час; одиниця: хвилина

D : зберігає стан літнього часу; коли значення в D вимкнено, літній час вимкнено. Коли значення в D є ON, літній час увімкнено.

2. Опис значення в коді функції S для функцій літнього часу

D.S.T State	Код функції S	опис
Вимкнено	0	Вимкнено функцію переходу на літній час
Увімкнено	1, 3	Увімкнено функцію переходу на літній час
Прочитати	2	Прочитайте налаштування літнього часу

· Вимкнено функцію переходу на літній час (див. приклад 1 нижче)

Коли операнд S дорівнює 0, функція переходу на літній час вимкнена. Коли S налаштовано на відключення переходу на літній час, значення в $S_1 - S_5$ не мають значення, а операнд D показує стан переходу на літній час як OFF.

· Увімкнено функцію переходу на літній час (див. приклади 2 і 3)

Якщо значення в S дорівнює 1 або 3, функція переходу на літній час увімкнена: S_1 і S_2 : налаштування місяця для початку переходу на літній час; S_3 і S_4 : налаштування місяця для завершення літнього часу; S_5 : налаштування для зміни через літній час; одиниця: хвилина; операнд D показує стан літнього часу. Коли функцію переходу на літній час увімкнено та система запускається вперше протягом часу запуску (S_1 , S_2), системний час додає значення, встановлене в S_5 один раз. Коли функція переходу на літній час вимкнена і система запускається вперше протягом часу завершення (S_3 , S_4), системний час один раз віднімає значення, встановлене в S_5 .

Режими для літнього часу

Режим	Код функції S	правила	Доступний для
Режим 1	1	За місяцем і числом	V1.00 або новішої версії
Режим 2	3	По місяцях і тижнях	V1.04 або новішої версії

Режим 1 (S=1): увімкнено за місяцем і числом (див. приклад 2)

Операнд	опис
S_1	Місяць переходу на літній час Діапазон: 1-12
S_2	Дата початку переходу на літній час Діапазон: 1-31
S_3	Місяць закінчення літнього часу Діапазон: 1-12
S_4	Дата закінчення переходу на літній час Діапазон: 1-31
S_5	Час, який змінився через перехід на літній час; одиниця: хвилина Діапазон: 1-1439 (протягом 1 дня)

Примітка 1. Якщо цю функцію ввімкнено, значення в **D** є ON.

Примітка 2. Якщо дату встановлено неправильно, функцію переходу на літній час увімкнути неможливо. SM0 увімкнено, а код помилки в SR0 – 16#200B. Наприклад, якщо встановлено неіснуючу дату, як-от 31 квітня, або початкову дату встановлено меншою за кінцеву дату в календарному році, наприклад, початкова дата – 1 жовтня, а кінцева – 1 квітня.

Примітка 3: якщо S_5 встановлено поза діапазоном, функцію літнього часу не можна ввімкнути. SM0 увімкнено, і помилка код у SR0 – 16#200B.

Режим 2 (S=3): увімкнено за тижнем і днем тижня (див. приклад 3)

Операнд	опис
S_1	Місяць переходу на літній час Діапазон: 1-12
S_2 S_2+1	S_2 : налаштування тижня для початку літнього часу; діапазон: 1-4 S_2+1 : в який день тижня S_2 ; діапазон: 1-7 (понеділок: 1, вівторок: 2 ..., неділя: 7)
S_3	Місяць закінчення літнього часу Діапазон: 1-12
S_4 S_4+1	S_4 : налаштування тижня до кінця літнього часу; діапазон: 1-4 S_4+1 : в який день тижня S_4 ; діапазон: 1-7 (понеділок: 1, вівторок: 2 ..., неділя: 7)
S_5	Час, який змінився через перехід на літній час; одиниця: хвилина Діапазон: 1-1439 (протягом 1 дня)

Примітка 1. Якщо цю функцію увімкнено, значення в **D** є ON.

Примітка 2: Діапазон налаштувань для **S₂** і **S₄** становить 1—4 або -1—-4. Значення -1 вказує на останній тиждень місяця, а -2 вказує на останній 2-й тиждень. Якщо значення в **S₂** дорівнює -2, а **S₂ + 1** дорівнює 7, це означає останню 2 неділю місяця. Якщо дату встановлено неправильно, функцію літнього часу увімкнути неможливо. SM0 увімкнено, а код помилки в SR0 – 16#200B.

Примітка 3: Якщо значення в **S₂ + 1** / **S₄ + 1** виходить за межі діапазону, значення за замовчуванням становить 7, що вказує на неділю.

Примітка 4: якщо **S₅** встановлено поза діапазоном, функцію літнього часу не можна увімкнути. SM0 увімкнено, а код помилки в SR0 – 16#200B.

Примітка 5. Якщо пристрій для операндів **S₂** і **S₄** — K або 16#, значення не зберігаються, SM0 увімкнено, а код помилки в SR0 — 16#2003.

Прочитайте функцію літнього часу (див. приклад 1-3)

Коли операнд **S** дорівнює 2, зчитується функція літнього часу. **S₁** і **S₂**: налаштування місяця для початку переходу на літній час; **S₃** і **S₄**: налаштування місяця для завершення літнього часу; **S₅**: налаштування для зміни через літній час; одиниця: хвилина. Коли **S** налаштовано на зчитування стану функції переходу на літній час, а вихідний стан **D** увімкнено, ПЛК зберігає значення налаштувань в операндах **S₁ – S₅**. Пристрій налаштовано на **D**, а **S** налаштовано на читання. Якщо пристрій налаштовано на K або 16#, значення не зберігаються, SM0 увімкнено, а код помилки в SR0 – 16#2003.

Пристрої з версією мікропрограми після V1.04 (включаючи V1.04) додають 4 до кодів функцій у **S** після зчитування стану літнього часу. Наприклад, після зчитування стану літнього часу коди функцій 0, 1, 3 стають 4, 5 і 7.

Коли стан літнього часу OFF, операнд і описи показані нижче.

Операнд	опис
S	До мікропрограми V1.04 (за винятком V1.04) функціональний код фіксувався на 2. Після версії мікропрограми V1.04 (V1.04 включено) код функції дорівнює 4, що вказує на те, що літній час OFF.
S₁ - S₅	Недійсний операнд
D	Літній час вимкнено.

Коли літній час увімкнено та в режимі 1, операнд і опис наведено нижче.

Операнд	опис
S	До мікропрограми V1.04 (за винятком V1.04) функціональний код фіксувався на 2. Після версії мікропрограми V1.04 (V1.04 включено) код функції дорівнює 5, що вказує на те, що літній час увімкнено та знаходиться в режимі 1.
S₁	Місяць переходу на літній час
S₂	Дата початку переходу на літній час
S₃	Місяць закінчення літнього часу
S₄	Дата закінчення переходу на літній час
S₅	Час, який змінився через перехід на літній час; одиниця: хвилина

Операнд	опис
D	Перехід на літній час увімкнено (увімкнено).

Коли літній час увімкнено та в режимі 2, операнд і опис наведено нижче.

Операнд	опис
S	Код функції – 7, що вказує на те, що літній час увімкнено та знаходиться в режимі 2.
S₁	Місяць переходу на літній час
S₂	S₂ : налаштування тижня для переходу на літній час
S₂+1	S₂+1 : в який день тижня S₂
S₃	Місяць закінчення літнього часу
S₄	S₄ : налаштування тижня до кінця літнього часу
S₄+1	S₄+1 : в який день тижня S₄
S₅	Час, який змінився через перехід на літній час; одиниця: хвилина
Операнд	Перехід на літній час увімкнено (увімкнено).

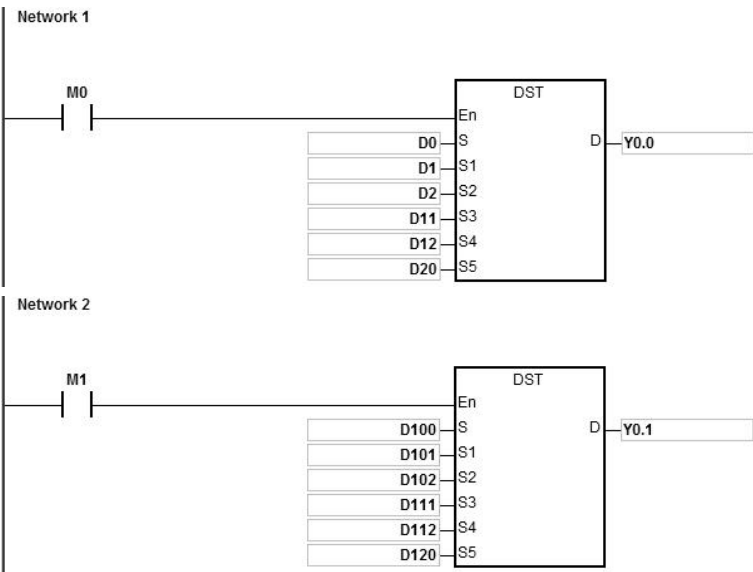
3. Ця інструкція призначена для ввімкнення/вимкнення функції переходу на літній час. Нормально розімкнутий чи закритий контакт не вплине на налаштування літнього часу. (див. приклад 2, щоб дізнатися більше про те, як перемкнути контакт M0 OFF=>ON) Ви можете скинути літній час, виконавши інструкцію ще раз. Немає необхідності вимикати, а потім увімкнути цю функцію, щоб скинути літній час.

4. При встановленні літнього часу з 1 квітня до 1 вересня тривалістю 60 хвилин;
годинник реального часу працює, як показано нижче.

Функція переходу на літній час вимкнена	Увімкнено функцію переходу на літній час
1 березня, о 3 годині	1 березня, о 3 годині
31 березня, о 3 годині	31 березня, о 3 годині
1 квітня, о 3 годині	1 квітня, о 4 годині
1 травня, о 3 годині	1 травня, о 4 годині
1 червня, о 3 годині	1 червня, о 4 годині
1 липня, о 3 годині	1 липня, о 4 годині
1 серпня, о 3 годині	1 серпня, о 4 годині
31 серпня, о 3 годині	31 серпня, о 4 годині
1 вересня, о 3 годині	1 вересня, о 3 годині

Приклад 1

Вимкніть функцію літнього часу та прочитайте стан літнього часу.



Значення параметрів і описи:

пристрій	Значення налаштування	опис
D0	0	Вимкніть функцію літнього часу
D1	X	Недійсний операнд
D2	X	Недійсний операнд
D11	X	Недійсний операнд
D12	X	Недійсний операнд
D20	X	Недійсний операнд

Увімкнути контакт M0

Y0.0=ВИМК., що означає, що функція літнього часу вимкнена.

D100=K2, що вказує на зчитування стану літнього часу. Увімкнути контакт M1

Значення параметрів і описи:

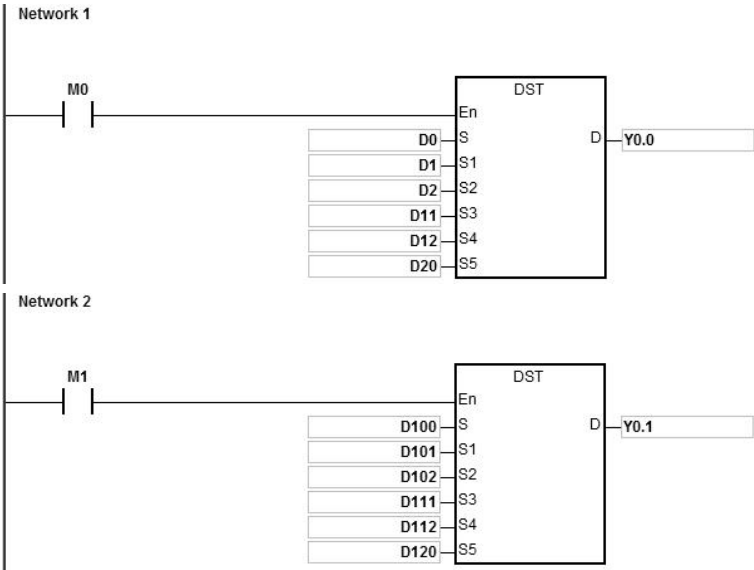
пристрій	Значення налаштування	опис
D100	2	До мікропрограми V1.04 (за винятком V1.04) функціональний код фіксувався на 2, вказуючи на зчитування стану літнього часу.
	4	Після версії мікропрограми V1.04 (V1.04 включено) код функції дорівнює 4, що вказує стан літнього часу OFF.
D101	X	Недійсний операнд
D102	X	Недійсний операнд
D111	X	Недійсний операнд

D112	X	Недійсний операнд
D120	X	Недійсний операнд
Y0.1	OFF	Стан вузла OFF.

Приклад 2

Увімкніть функцію літнього часу та прочитайте стан літнього часу.

Встановіть літній час на 1 квітня та 3 вересня на 60 хвилин.



Значення параметрів і описи:

пристрій	Значення налаштування	опис
D0	1	Літній час увімкнено та знаходиться в режимі 1.
D1	4	Місяць початку: квітень
D2	1	Дата початку: 1-е
D11	9	Кінцевий місяць: вересень
D12	3	Дата закінчення: 3 числа
D20	60	Тривалість: 60 хвилин

Увімкнути контакт M0

Y0.0=ON, що вказує на те, що функцію літнього часу увімкнено.

Системний час PLC додає 60 хвилин, коли настає дата 1 квітня, і віднімає 60 хвилин, коли настає дата 3 вересня, щоб закінчити літній час.

D100=K2, що вказує на зчитування стану літнього часу. Увімкнути контакт M1

Значення параметрів і описи:

пристрій	Налаштування Значення	опис
D100	2	До мікропрограми V1.04 (за винятком V1.04) функціональний код фіксувався на 2, вказуючи на зчитування стану літнього часу.
	5	Після мікропрограми V1.04 (V1.04 включено), код функції 5, що вказує на літній час увімкнено та в режимі 1.
D101	4	Місяць початку: квітень
D102	1	Дата початку: 1 ^{ст}
D111	9	Кінцевий місяць: вересень
D112	3	Дата закінчення: 3 число
D120	60	Тривалість: 60 хвилин
Y0.1	ON	Стан вузла ON.

Використовуйте інструкцію DST або HWCONFIG в ISPSOft, щоб прочитати стан літнього часу. HWCONFIG перетворює результат від номера тижня до відповідних дат і місяців автоматично.

Знову включити контакт M0 OFF=>ON; ця дія не впливає на літній час. Літній час не скидається.

Значення параметрів і описи:

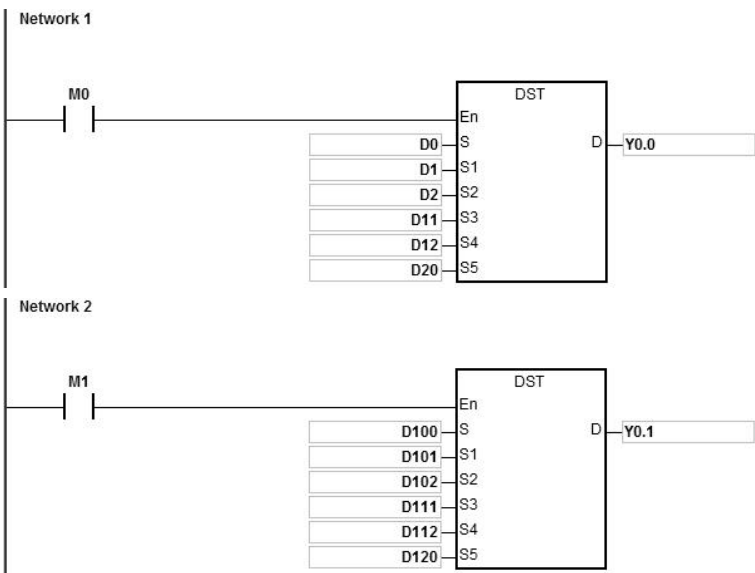
6

пристрій	Налаштування Значення	опис
D100	2	До мікропрограми V1.04 (за винятком V1.04) функціональний код фіксувався на 2, вказуючи на зчитування стану літнього часу.
	5	Після мікропрограми V1.04 (V1.04 включено), код функції 5, що вказує на літній час увімкнено та в режимі 1.
D101	4	Місяць початку: квітень
D102	1	Дата початку: 1 ^{ст}
D111	9	Кінцевий місяць: вересень
D112	3	Дата закінчення: 3 число
D120	60	Тривалість: 60 хвилин
Y0.1	ON	Стан вузла ON.

Якщо літній час починається з 3 години 1 квітня, додається 60 хвилин; годинник реального часу показує 4 години 1^{ст} квітень. Незалежно від того, скільки разів контакт M0 вимикався або вмикався, годинник реального часу зберігає той самий літній час.

Приклад 3

Увімкніть функцію літнього часу та в режимі 2.
Встановіть літній час так, щоб він починався з 2 -ї середи травня до 3 -ї п'ятниці вересня, а тривалість – 60 хвилин.



Значення параметрів і описи:

пристрій	Значення налаштування	опис
D0	3	Літній час увімкнено та знаходиться в режимі 2.
D1	5	Місяць початку: травень
D2	2	Номер стартового тижня: 2 -й тиждень
D3	3	День початку: середа
D11	9	Кінцевий місяць: вересень
D12	3	Кінцевий номер тижня: 3 -й тиждень
D13	5	День закінчення: п'ятниця
D20	60	Тривалість: 60 хвилин

Увімкнути контакт M0

Y0.0=ON, що вказує на те, що функцію літнього часу увімкнено.

У 2017 році 2 -га середа травня – 10 травня , а 3 -ья п'ятниця вересня – 15 вересня . Системний час PLC додає 60 хвилин, коли настає дата 10 травня, і віднімає 60 хвилин, коли настає дата 15 вересня, щоб закінчити літній час.

D100=K2, що вказує на зчитування стану літнього часу.

Увімкнути контакт M1

Значення параметрів і описи:

пристрій	Налаштування Значення	опис
D100	2	До мікропрограми V1.04 (за винятком V1.04) функціональний код фіксувався на 2, вказуючи на зчитування стану літнього часу.
	7	Після версії мікропрограми V1.04 (V1.04 включено), код функції дорівнює 4, що вказує на літній час увімкнено та в режимі 2.
D101	5	Місяць початку: травень
D102	2	Номер стартового тижня: 2 -й тиждень
D103	3	День початку: середа
D111	9	Кінцевий місяць: вересень
D112	3	Кінцевий номер тижня: 3 -й тиждень
D113	5	День закінчення: п'ятниця
D120	60	Тривалість: 60 хвилин
Y0.1	ON	Стан вузла ON.

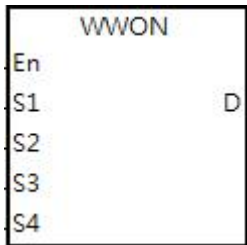
Використовуйте інструкцію DST або HWCONFIG в ISPSOft, щоб прочитати стан літнього часу. HWCONFIG перетворює результат від номера тижня до відповідних дат і місяців автоматично.

API	Код інструкції				Операнд					функція							
1608		WON			S_1, S_2, D					Встановлення тижневого робочого часу							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1								O									
S_2								O									
S_3								O									
S_4								O									
D		O	O	O													

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
D	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- S_1 : Година для початку роботи (займає 7 послідовних пристроїв)
- S_2 : Хвилина початку роботи (займає 7 пристроїв посліпль)
- S_3 : Година припинення роботи (займає 7 пристроїв посліпль)
- S_4 : Хвилина до припинення роботи (займає 7 пристроїв посліпль)
- D : Вихідний контроль

Пояснення

- Ця інструкція дозволяє встановити час початку роботи на тиждень. $S_1 - S_1 + 6$ дозволяє встановити час у неділю / понеділок / вівторок / середа / четвер / п'ятниця / субота відповідно. Цей операнд займає 7 послідовних пристроїв. Ви можете використовувати змінні в ARRAY для оголошення операндів.
- $S_2 - S_2 + 6$ $S_1 - S_1 + 6$ дозволяє встановити хвилини для початку роботи в неділю / понеділок / вівторок / середу / Четвер/п'ятниця/субота відповідно. Цей операнд займає 7 послідовних пристроїв. Ви можете використовувати змінні в ARRAY для оголошення операндів.
- $S_3 - S_3 + 6$ дозволяє встановити годину припинення роботи в неділю / понеділок / вівторок / середу / четвер / п'ятницю / субота відповідно. Цей операнд займає 7 послідовних пристроїв. Ви можете використовувати змінні в ARRAY для оголошення операндів.
- $S_4 - S_4 + 6$ дозволяє встановити хвилини для припинення роботи в неділю / понеділок / вівторок / середу / четвер /

п'ятниця/субота відповідно. Цей операнд займає 7 послідовних пристроїв. Ви можете використовувати змінні в ARRAY для оголошення операндів.

5. Якщо значення години в S_1 більше, ніж значення, встановлене в S_3 , це означає, що час припинення роботи наступний день. Наприклад, якщо ви встановлюєте час початку роботи о 18:00 у понеділок і час припинення роботи о 6:00, це означає, що час припинення роботи – 6:00 вівторка.

	Початок робочого часу					Припинити робочий час					
День	старт	годину		старт	хвилина		Stop	годину		Stop	хвилина
неділя	S₁	24		S₂	00		S₃	24		S₄	00
понеділок	S₁+1	18		S₂+1	00		S₃+1	06		S₄+1	00

6. Значення налаштування для години становить 0–23. Якщо значення налаштування виходить за межі діапазону, ця функція не активна.

Значення налаштування для хвилин становить 0–59. Якщо значення налаштування виходить за межі діапазону, ця функція вмикається, але використовує 0 як значення налаштування.

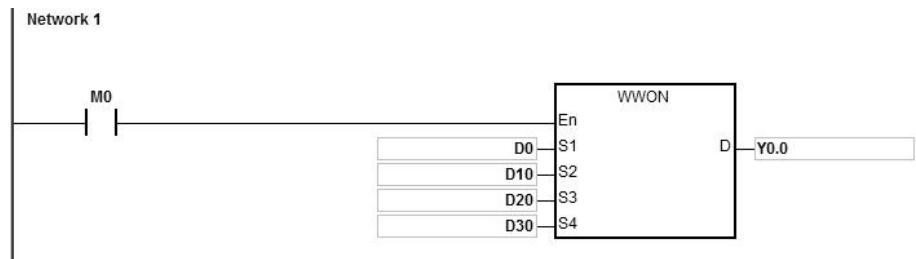
7. Якщо потрібно встановити робочий час більше ніж 1 день, ви можете встановити 24 години, що означає, що система не перевіряє час початку та закінчення роботи. Наприклад, щоб встановити час початку роботи на 8 ранку понеділка, а час закінчення роботи на 20:00 вівторка, використовуйте $S_1+1=8$, $S_3+1=24$, $S_1+2=24$ і $S_3+2=20$. Дивіться формулу в наступній таблиці.

	Початок робочого часу						Припинити робочий час					
День	старт	годину		старт	хвилина		Stop	годину		Stop	хвилина	
неділя	S₁	24		S₂	00		S₃	24		S₄	00	
понеділок	S₁+1	08		S₂+1	00		S₃+1	24		S₄+1	00	
вівторок	S₁+2	24		S₂+2	00		S₃+2	20		S₄+2	00	

8. Ця інструкція повинна працювати з годинником реального часу в ПЛК. Перед початком роботи переконайтеся, що батарея ПЛК справна надійно встановлений і працює правильно.
9. Немає обмежень на кількість разів, які ви можете виконати інструкцію, але пристрій керування виведенням **D** не можна використовувати повторно. Якщо ви використовуєте пристрій **D** повторно, виконується лише останній результат виводу з інструкції WWON.
10. Якщо потрібно більше ніж 1 набір робочих годин, використовуйте інструкцію WWON неодноразово за потреби. Зауважте, що ви не можете повторно використовувати пристрій керування виходом **D**.

Приклад 1

Встановіть робочий час з 8:00 до 18:00 з понеділка по п'ятницю та без роботи в суботу та неділю.



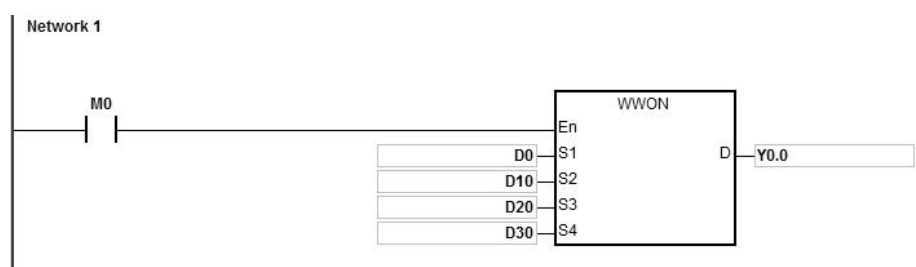
У наведеній нижче таблиці наведено налаштування для пристрою D .

	Початок робочого часу						Припинити робочий час				
День	старт	година		старт	хвилина		Stop	година		Stop	хвилина
неділя	D0	24		D10	00		D20	24		D30	00
понеділок	D1	08		D11	00		D21	18		D31	00
вівторок	D2	08		D12	00		D22	18		D32	00
Середа	D3	08		D13	00		D23	18		D33	00
четвер	D4	08		D14	00		D24	18		D34	00
П'ятниця	D5	08		D15	00		D25	18		D35	00
Субота	D6	24		D16	00		D26	24		D36	00

Коли M0 увімкнено, Y0.0 увімкнено з 8:00 до 18:00 з понеділка по п'ятницю; в інший час Y0.0 OFF.

Приклад 2

Встановіть робочий час з 18:00 понеділка до 08:00 вівторка та з 18:00 вівторка до 08:00 середи. Дотримуйтеся цієї схеми до 08:00 суботи та не працюєте в неділю.



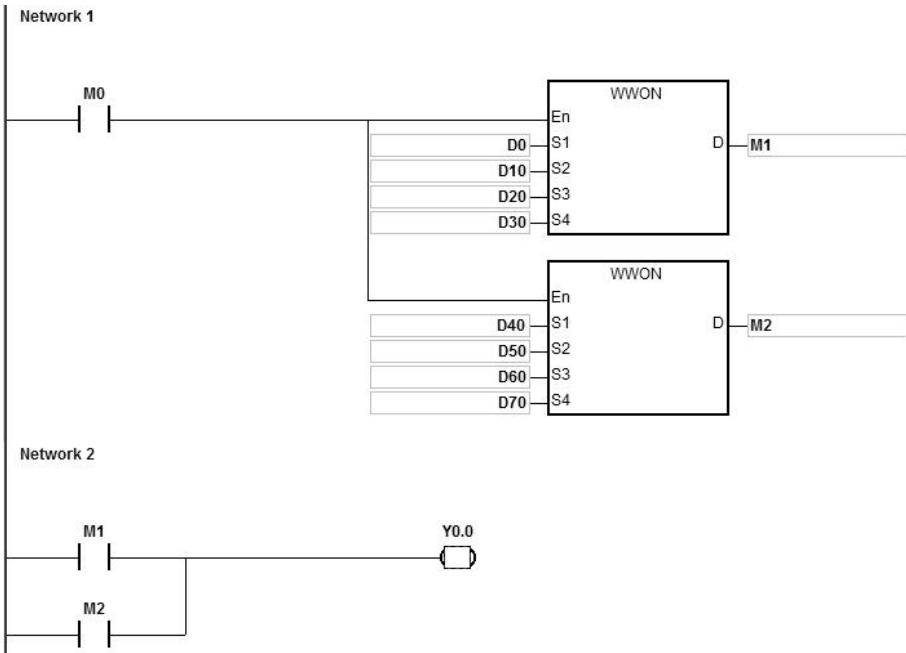
У наведеній нижче таблиці наведено налаштування для пристрою **D** .

	Початок робочого часу						Припинити робочий час					
День	старт	година		старт	хвилина		Stop	година		Stop	хвилина	
неділя	D0	24		D10	00		D20	24		D30	00	
понеділок	D1	18		D11	00		D21	08		D31	00	
вівторок	D2	18		D12	00		D22	08		D32	00	
Середа	D3	18		D13	00		D23	08		D33	00	
четвер	D4	18		D14	00		D24	08		D34	00	
П'ятниця	D5	18		D15	00		D25	08		D35	00	
Субота	D6	24		D16	00		D26	24		D36	00	

Коли M0 увімкнено, Y0.0 увімкнено з 18:00 до 8:00 наступного дня з понеділка по п'ятницю, а в інший час Y0.0 вимкнено.

Приклад 3

Встановити робочий час з 08:00 до 12:00 та з 14:00 до 17:30 з понеділка по п'ятницю. У суботу та неділю не працює.



У наступній таблиці наведено ранкові налаштування для пристрою **D**.

День	Початок робочого часу					Припинити робочий час				
	старт	годину		старт	хвилина	СТІЙ	годину		СТІЙ	хвилина
неділя	D0	24		D10	00	D20	24		D30	00
понеділок	D1	08		D11	00	D21	12		D31	00
вівторок	D2	08		D12	00	D22	12		D32	00
Середа	D3	08		D13	00	D23	12		D33	00
четвер	D4	08		D14	00	D24	12		D34	00
П'ятниця	D5	08		D15	00	D25	12		D35	00
Субота	D6	24		D16	00	D26	24		D36	00

У наступній таблиці перераховані налаштування вдень для пристрою **D**.

День	Початок робочого часу					Припинити робочий час				
	старт	годину		старт	хвилина	СТІЙ	годину		СТІЙ	хвилина
неділя	D40	24		D50	00	D60	24		D70	00
понеділок	D41	14		D51	00	D61	17		D71	30
вівторок	D42	14		D52	00	D62	17		D72	30
Середа	D43	14		D53	00	D63	17		D73	30
четвер	D44	14		D54	00	D64	17		D74	30
П'ятниця	D45	14		D55	00	D65	17		D75	30
Субота	D46	24		D56	00	D66	24		D76	00

Коли M0 увімкнено, Y0.0 увімкнено з 08:00 до 12:00 та з 14:00 до 17:30 з понеділка по п'ятницю, а в інший час Y0.0 вимкнено.

6.18 Інструкції з периферійних пристроїв

6.18.1 Список периферійних інструкцій

У наведеній нижче таблиці наведено інструкції щодо периферійних пристроїв, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1700	TKY	DTKY	–	Десятиклавішна клавіатура
1701	HKY	DHKY	–	Шістнадцятиклавішна клавіатура
1702	DSW	–	–	DIP-перемикач
1703	ARWS	–	–	Клавіші зі стрілками
1704	SEGL	–	–	Семисегментний дисплей із засувками

6.18.2 Пояснення периферійних інструкцій

API	Код інструкції				Операнд				функція							
1700	D	TKY			S, D ₁ , D ₂				Десятиклавішна клавіатура							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	○															
D ₁					●	●	●	●								
D ₂		○	○	○				○								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S	●												
D ₁		●	●			●	●						
D ₂	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ

TKY		DTKY	
En		En	
S	D1	S	D1
	D2		D2

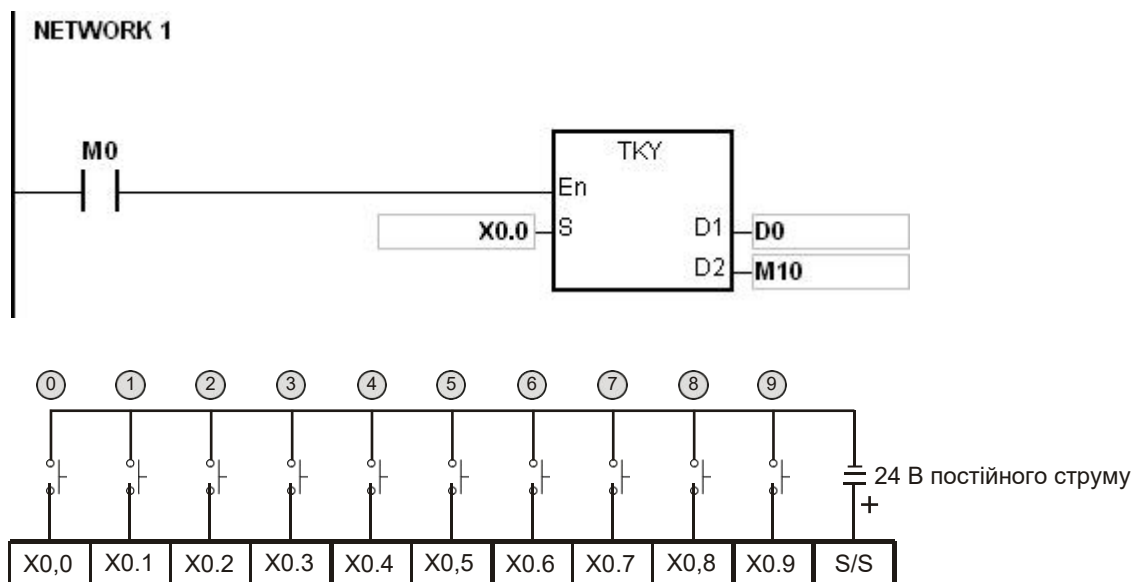
- S** : Перший пристрій введення
D₁ : Пристрій, де зберігається значення
D₂ : Вихідний сигнал

Пояснення

- Десять зовнішніх входів, починаючи з входу, визначеного **S**, представляють 0–9 у десятковій системі числення. Вони є підключений до десяти ключів. Ви можете ввести чотиризначне десяткове значення 0–9999 (16-бітна інструкція) або восьмизначне десяткове значення 0–99 999 999 (32-бітна інструкція), натискаючи клавіші по порядку. Інструкція зберігає десятковий знак значення в **D₁** і зберігає вихідні сигнали в **D₂**.
- Операнд **S** займає десять біт.
- Операнд **D₂** займає одинадцять біт. Будь ласка, не змінюйте стани бітів під час виконання інструкції.
- Коли умовний контакт не ввімкнуто, одинадцять бітів, починаючи з біта, визначеного **D₂**, вимкнені.
- Під час редагування в режимі онлайн скиньте умовний контакт, щоб ініціалізувати інструкцію.
- Ви можете використовувати 32-розрядний лічильник лише тоді, коли **D₁** використовує 32-розрядні інструкції.

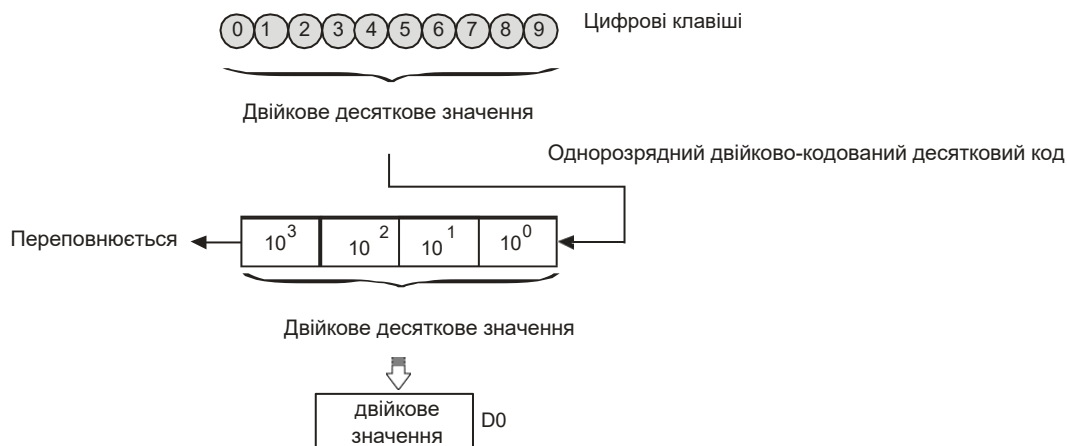
приклад

- Десять зовнішніх входів, починаючи з X0.0, підключені до десяти клавiш, які представляють 0–9 у десятичній системі числення. Коли M0 увімкнено, інструкція зберігає значення, яке ви вводите як двійкове значення в D0, і зберігає вихідні сигнали в M10–M19.



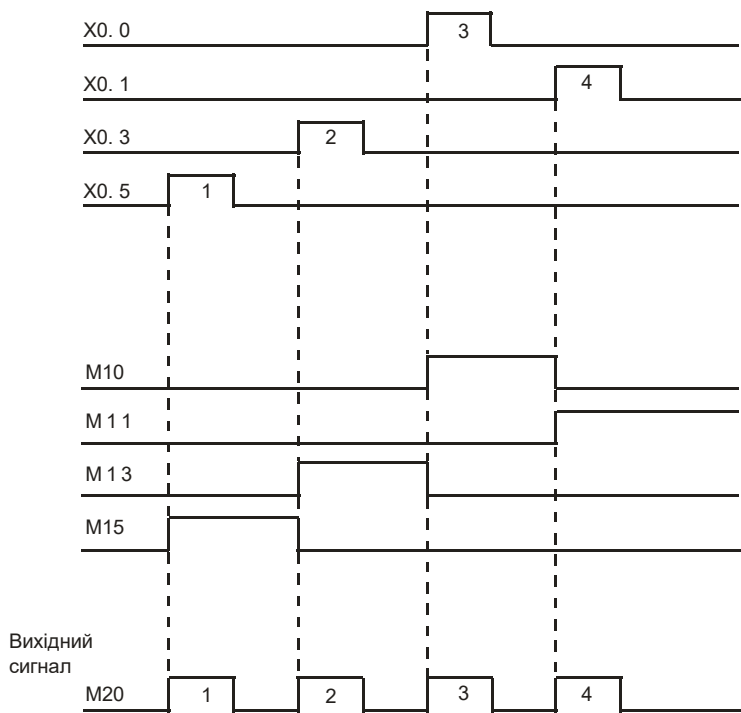
Примітка. У цьому прикладі використовується модуль цифрового введення AH16AM10N-5A.

6



- Якщо клавiші, підключені до X0.5, X0.3, X0.0 і X0.1, натиснути в порядку, зазначеному в часовій діаграмі, інструкція зберігає результат 5301 у D0. Максимальне значення, яке можна зберегти в D0, становить 9999. Якщо значення перевищує чотири цифри, перша цифра зліва переповнюється.
- Після натискання клавiші, підключеної до X0.2, і до натискання інших клавiш M12 увімкнено. Те саме стосується й інших клавiш.

- 4. Коли натискається клавіша, під'єднана до входу в діапазоні від X0.0 до X0.9, відповідний вихід у діапазоні від M10 до M19 увімкнено.
- 5. При натисканні однієї з клавіш M20 увімкнено.
- 6. Коли умовний контакт M0 вимкнено, значення, що зберігається в D0, не змінюється; однак M10–M20 вимкнено.



Додаткові зауваження

- 1. Якщо ви оголошуєте операнд **S** в ISPSOft, типом даних буде ARRAY [10] BOOL.
- 2. Якщо ви оголошуєте операнд **D₂** в ISPSOft, типом даних буде ARRAY [11] BOOL.

API	Код інструкції			Операнд						функція						
1701	D	HKY		S_1, S_2, D_1, D_2, D_3						Шістнадцятиклавівна клавіатура						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	○															
S_2								●								
D_1		○														
D_2					●	●	●	●								
D_3		○	○	○				○								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2		●				●							
D_1	●												
D_2		●	●			●	●						
D_3	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	AS

СИМВОЛ

HKY		DHKY	
En		En	
S1	D1	S1	D1
S2	D2	S2	D2
	D3		D3

- S_1 : Перший пристрій введення
- S_2 : Тільки для системного використання
- D_1 : Перший вихідний пристрій
- D_2 : Пристрій, де зберігається значення
- D_3 : Вихідний сигнал

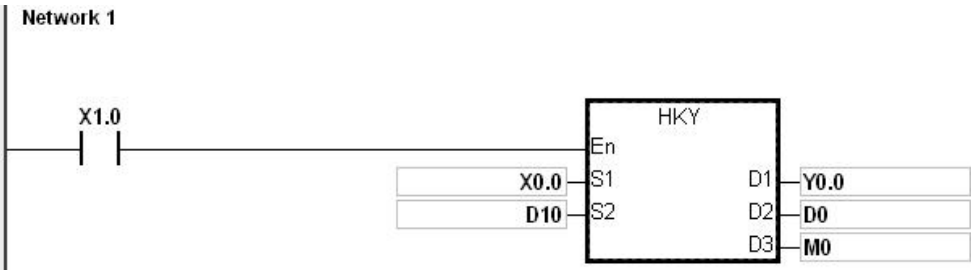
Пояснення

- Чотири зовнішні входи, починаючи з входу, визначеного S , з'єднані з чотирма зовнішніми виходами, починаючи з виходу, визначеного D_1 , щоб утворити клавіатуру з 16 клавішами. Інструкція зберігає значення, яке ви вводите, натискаючи клавіші в D_2 , і зберігає вихідні сигнали в D_3 . Якщо одночасно натиснути кілька клавіш, зберігається менше значення.
- Значення, яке ви вводите, натискаючи клавіші, тимчасово зберігається в D_2 . Для 16-бітної інструкції HKY максимальне значення, яке можна зберегти в D_2 , становить 9999. Якщо значення перевищує чотири цифри, перша цифра зліва переповнюється. Для 32-розрядної інструкції DHKY максимальне значення, яке можна зберегти в D_2 , становить 9999. Якщо значення перевищує вісім цифр, перша цифра зліва переповнюється.
- Після завершення інструкції SM692 увімкнено. Тобто SM692 увімкнено для циклу сканування після завершення сканування матриці.

4. Ви можете використовувати 32-розрядний лічильник лише тоді, коли **D₂** використовує 32-розрядні інструкції.

приклад

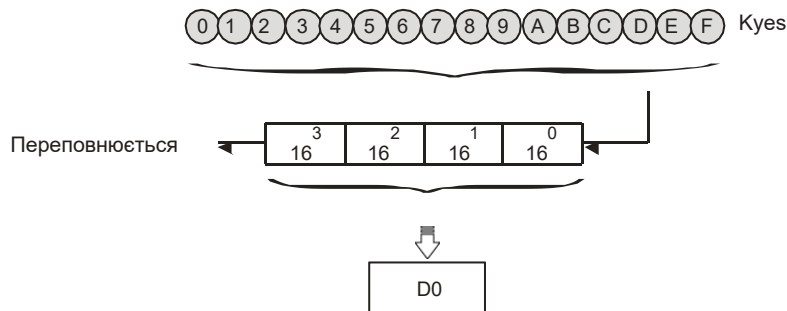
1. Чотири зовнішні входи X0.0–X0.3 з’єднані з чотирма зовнішніми виходами Y0.0–Y0.3 для формування 16-клавішної клавіатури. Коли X1.0 увімкнено, інструкція зберігає значення, яке ви вводите як двійкове значення в D0, і зберігає вихідні сигнали в M0–M7.



Функція SM691:

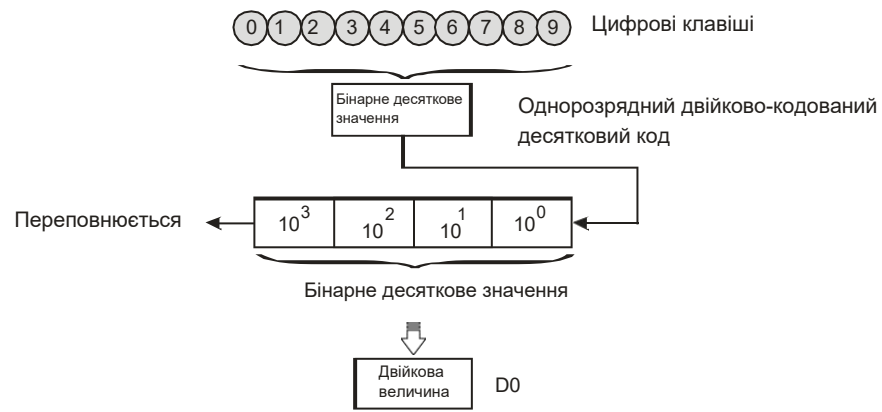
- Якщо SM691 увімкнено, 16-бітна інструкція приймає 0–F як шістнадцяткові значення.

■ Цифрові клавіші:



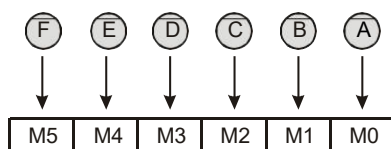
- Якщо SM691 вимкнено, 16-бітна інструкція приймає A–F як функціональні клавіші.

■ Цифрові клавіші:



■ Функціональні клавіші:

- ◆ Коли натиснуто A, M0 залишається УВІМКНЕНИМ. Коли натиснуто D, M0 вимикається, а M3 залишається увімкненим.
- ◆ Якщо натиснуто кілька функціональних клавіш, пріоритет має клавіша, яка натиснута першою.

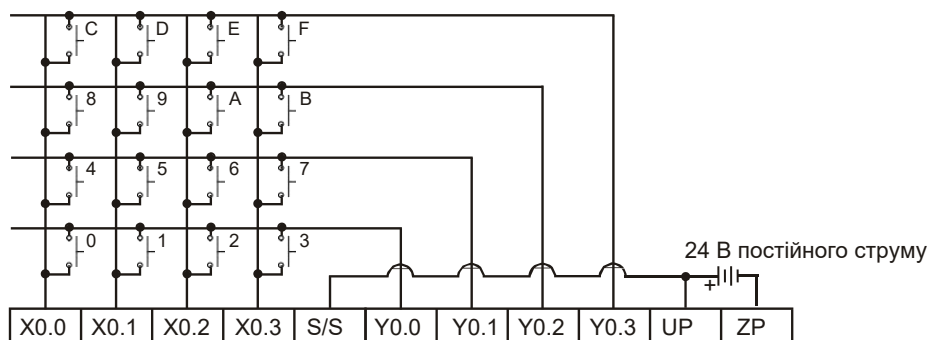


2. Вихідні сигнали:

- Коли натиснуто клавішу в межах від A до F, M6 увімкнено.
- Коли натискається клавіша в діапазоні від 0 до 9, M7 увімкнено.

3. Коли умовний контакт X1.0 вимикається, значення, збережене в D0, не змінюється. Однак M0–M7 вимкнено.

4. Зовнішня проводка:



Примітка. У цьому прикладі використовується транзисторний вихідний модуль AH16AP11T-5A.

Додаткові зауваження

1. Під час виконання цієї інструкції занадто довгий або занадто короткий час циклу сканування призведе до неправильного зчитування стану перемикачів. Щоб вирішити проблему, скористайтеся наведеними нижче порадами.

- Якщо цикл сканування надто короткий, пристрій вводу/виводу може не відповісти вчасно та не зможе прочитати правильний стан входів. Ви можете встановити фіксований час сканування, щоб вирішити цю проблему.
- Якщо цикл сканування надто довгий, комутатор може повільно реагувати. Ви можете записати цю інструкцію в завдання переривання таймера, щоб встановити фіксований час для виконання цієї інструкції.

2. Якщо ви оголошуєте операнд **S** в ISPSOft, типом даних буде ARRAY [4] BOOL.

3. Якщо ви оголошуєте операнд **D1** в ISPSOft, типом даних буде ARRAY [4] BOOL.

4. Якщо ви оголошуєте операнд **D3** в ISPSOft, типом даних буде ARRAY [8] BOOL.

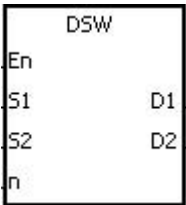
API		Код інструкції			Операнд						функція							
1702			DSW				S_1, S_2, D_1, D_2, n						DIP-перемикач					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S_1	○																	
S_2								●										
D_1		○																
D_2					●	●		●										
n					●	●		●	●		○	○	○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2		●				●							
D_1	●												
D_2		●				●							
n		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
-		AS		-	

СИМВОЛ

СИМВОЛ



- S_1 : Перший пристрій введення
- S_2 : Тільки для системного використання
- D_1 : Перший вихідний пристрій
- D_2 : Пристрій, де зберігається значення
- n : Кількість DIP-перемикачів

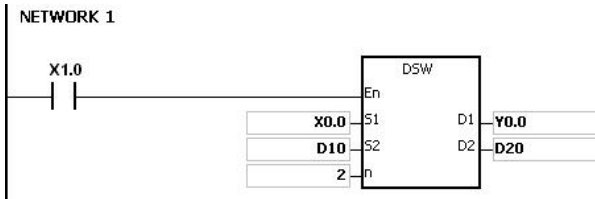
Пояснення

- Чотири або вісім зовнішніх входів, починаючи з входу, визначеного S_1 , з'єднані з чотирма зовнішніми виходами, починаючи з виходу, визначеного D_1 , щоб утворити чотирирозрядний DIP-перемикач або два чотиризначні DIP-перемикачі. Інструкція зберігає значення DIP-перемикача в D_2 . Від n залежить наявність одного чотиризначного DIP-перемикача чи двох чотиризначних DIP-перемикачів.
- Якщо n дорівнює 1, операнд D_2 займає один регістр. Якщо n дорівнює 2, операнд D_2 займає два регістри.
- S_2 і $S_2 + 1$ призначені лише для системного використання та займають два пристрої. Будь ласка, не змінюйте значення в цих пристроях.
- Після завершення інструкції SM694 увімкнено для циклу сканування.
- Коли умовний контакт не активований, чотири зовнішні виходи, починаючи з виходу, визначеного D_1 , залишаються ВИМКНЕНИМИ.

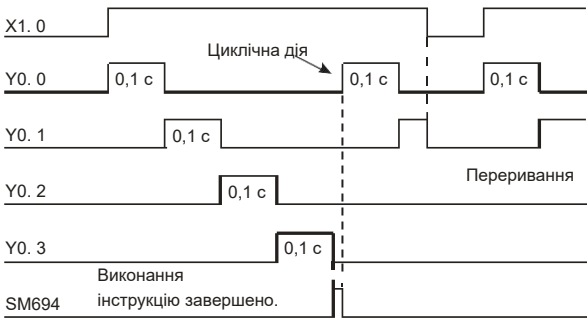
6. Під час редагування в режимі онлайн скиньте умовний контакт, щоб ініціалізувати інструкцію.

приклад

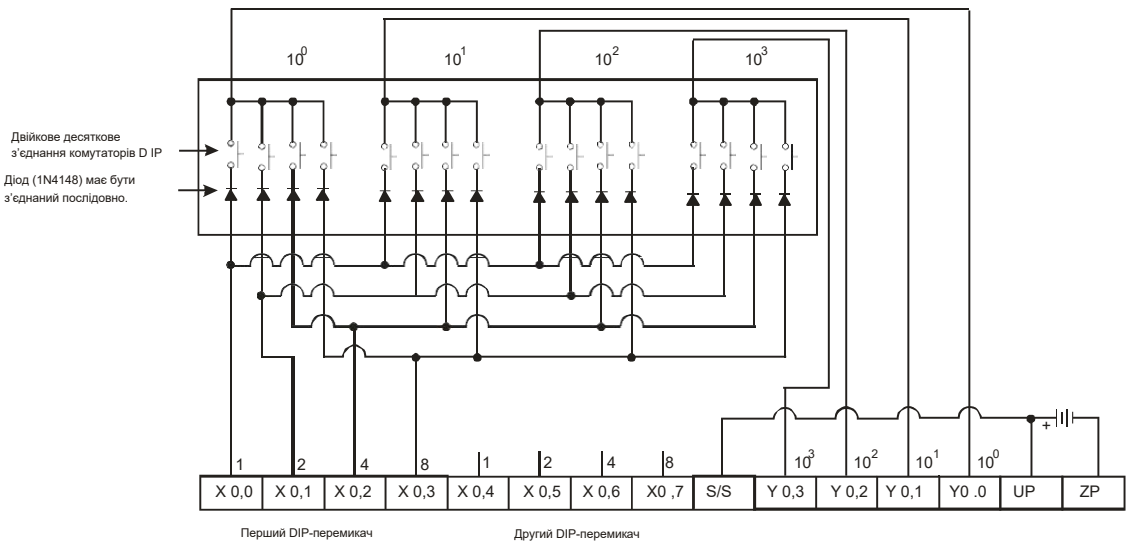
1. X0.0–X0.3 з’єднані з Y0.0–Y0.3 для формування першого DIP-перемикача, а X0.4–X0.7 з’єднані з Y0.0–Y0.3 для формування другого DIP-перемикача . Коли X1.0 увімкнено, інструкція перетворює значення, яке ви вводите за допомогою першого DIP-перемикача, у двійкове значення та зберігає результат перетворення в D20. Ця інструкція перетворює значення, яке ви вводите за допомогою другого DIP-перемикача, у двійкове значення та зберігає результат перетворення в D21.



2. Коли X1.0 увімкнено, Y0.0–Y0.3 циклічно вмикаються. Після завершення інструкції SM694 увімкнено для циклу сканування.
3. На наступному графіку показано результати. Y0.0–Y0.3 повинні бути транзисторами.



4. На наступному малюнку показано DIP-перемикачі.



Примітка. У цьому прикладі використовується транзист або вихідний модуль AH16AP11T-5A.

Додаткові зауваження

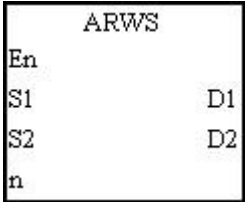
1. Якщо **n** перевищує діапазон, інструкція не виконується, **SM0** увімкнено, а код помилки в **SR0** – 16#200B.
2. Якщо ви оголошуєте операнд **D₁** у **ISPSoft**, типом даних є **ARRAY [4]** із **BOOL**.

API		Код інструкції				Операнд						функція					
1703			ARWS			S_1, S_2, D_1, D_2, n						Клавіші зі стрілками					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	O																
S_2								●									
D_1					●	●		●									
D_2		O															
n					●	●		●	●		O	O	O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2		●				●							
D_1		●				●							
D_2	●												
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



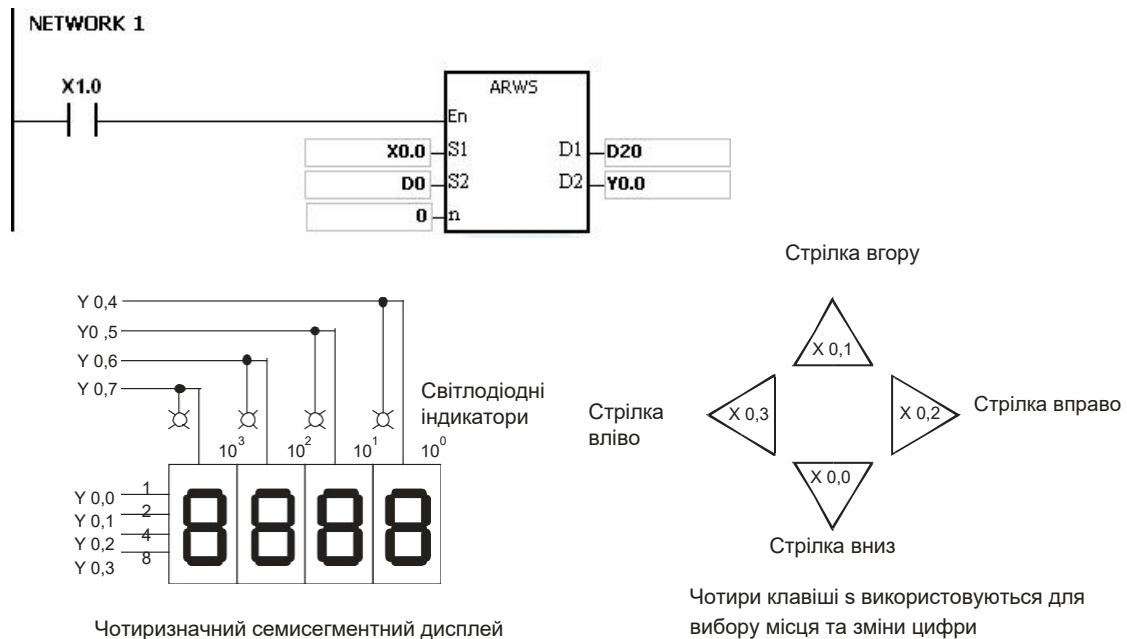
- S_1 : Перший пристрій введення
- S_2 : Тільки для системного використання
- D_1 : Пристрій, де зберігається значення налаштування
- D_2 : Перший вихідний пристрій
- n : Позитивна/Негативна логіка

Пояснення

- Ця інструкція визначає S_1 як стрілку вниз, $S_1 + 1$ як стрілку вгору, $S_1 + 2$ як стрілку вправо та $S_1 + 3$ як стрілку вліво. Інструкція зберігає значення параметра в D_1 , і значення має бути між 0–9999.
- Операнд S_1 займає чотири послідовні бітові пристрої.
- S_2 лише для системного використання. Будь ласка, не змінюйте значення в ньому.
- Операнд D_2 займає вісім послідовних бітових пристроїв.
- Коли умовний контакт вимкнено, восьмирозрядні пристрої, починаючи з розрядного пристрою, визначеного D_2 , залишаються ВИМКНЕНИМИ.
- Операнд n має бути між 0–3.
- Під час редагування в режимі онлайн скиньте умовний контакт, щоб ініціалізувати інструкцію.

приклад

- Інструкція визначає X0.0 як стрілку вниз, X0.1 як стрілку вгору, X0.2 як стрілку вправо та X0.3 як стрілку вліво. Інструкція зберігає значення параметра в D20, і значення параметра має бути між 0–9999.
- Коли X1.0 увімкнено, вибирається цифра на місці 10^3 . Якщо натиснути стрілку вліво, місця вибираються послідовно ($10^3 \rightarrow 10^0 \rightarrow 10^1 \rightarrow 10^2 \rightarrow 10^3 \rightarrow 10^0$).
- Якщо натиснути стрілку вправо, місця вибираються послідовно ($10^3 \rightarrow 10^2 \rightarrow 10^1 \rightarrow 10^0 \rightarrow 10^3 \rightarrow 10^2$). Світлодіодні індикатори для відповідних місць підключені до Y0.4–Y0.7. Коли цифри на місцях вибираються послідовно, світлодіодні індикатори горять послідовно.
- Якщо натиснути стрілку вгору, цифра у вибраному місці зміниться ($0 \rightarrow 1 \rightarrow 2 \rightarrow \dots 8 \rightarrow 9 \rightarrow 0 \rightarrow 1$). Якщо натиснути стрілку вниз, цифра у вибраному місці зміниться ($0 \rightarrow 9 \rightarrow 8 \rightarrow \dots 1 \rightarrow 0 \rightarrow 9$). Нова цифра відображається на семисегментному дисплеї.

**Додаткові зауваження**

- Якщо n перевищує діапазон, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
- Якщо ви оголошуєте операнд S_1 в ISPSOft, тип даних буде ARRAY [4] BOOL.
- Якщо ви оголошуєте операнд D_2 в ISPSOft, тип даних буде ARRAY [8] з BOOLL.

API	Код інструкції				Операнд				функція							
1704		SEGL			S_1, S_2, D, n				Семисегментний дисплей із засувками							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●		O	O				
S_2								●								
D		O														
n					●	●		●	●		O	O	O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
D	●												
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- S_1 : Вихідний пристрій
- S_2 : Тільки для системного використання
- D : Перший вихідний пристрій
- n : Позитивна/Негативна логіка

Пояснення

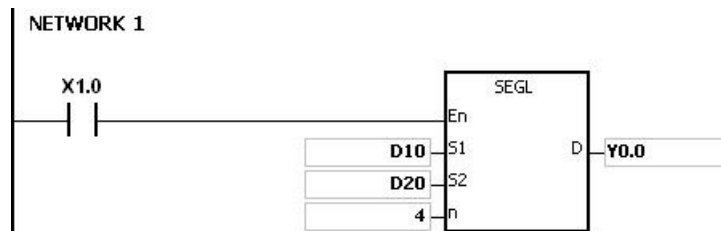
- Вісім зовнішніх виходів, починаючи з виходу, визначеного D , підключені до чотиризначного семисегментного дисплея; або дванадцять зовнішніх виходів, починаючи з виходу, визначеного D , підключені до двох чотирирозрядних семисегментних дисплеїв. Кожне місце оснащено драйвером, який перетворює двійкове десятикове значення в семисегментні дані, і кожен драйвер оснащено засувкою, яку можна використовувати для зберігання інформації про стан.
- Значення в S_1 — це значення, яке відобразиться на першому семисегментному дисплеї, а значення в $S_1 + 1$ — це значення, яке відобразиться на другому семисегментному дисплеї.
- S_2 лише для системного використання. Будь ласка, не змінійте значення в ньому.
- Операнд n має бути між 0–7. Для отримання додаткової інформації зверніться до додаткового зауваження.
- Чи є один чотирирозрядний семисегментний дисплей або два чотирирозрядні семисегментні дисплеї, і чи є вихід позитивним чи негативним логічним виходом, залежить від n .
- При наявності одного чотирирозрядного семисегментного дисплея зайнято вісім виходів. Якщо є два чотирирозрядних семисегментних дисплея, зайнято дванадцять виходів.

7. Коли інструкція виконується, виходи циклічно вмикаються. Якщо під час виконання інструкції умовний контакт перемикається з ВІМК. на УВІМК., виходи циклічно знову вмикаються.

8. Після завершення виконання інструкції SM693 увімкнено для циклу сканування.

приклад

1. Коли X1.0 увімкнено, інструкція виконується. Y0.0–Y0.4 утворюють ланцюг. Ця інструкція перетворює значення в D10 у двійкове десятичне значення та показує результат перетворення на першому семисегментному дисплеї. Ця інструкція перетворює значення в D11 у двійкове десятичне значення та відображає результат перетворення на другому семисегментному дисплеї. Якщо значення в D10 або D11 перевищує 9999, виникає помилка операції.



2. Коли X1.0 увімкнено, Y0.4–Y0.7 циклічно вмикаються. Щоб увімкнути Y0.4–Y0.7, потрібно дванадцять циклів сканування. Після

інструкція завершена, SM693 увімкнено для циклу сканування.

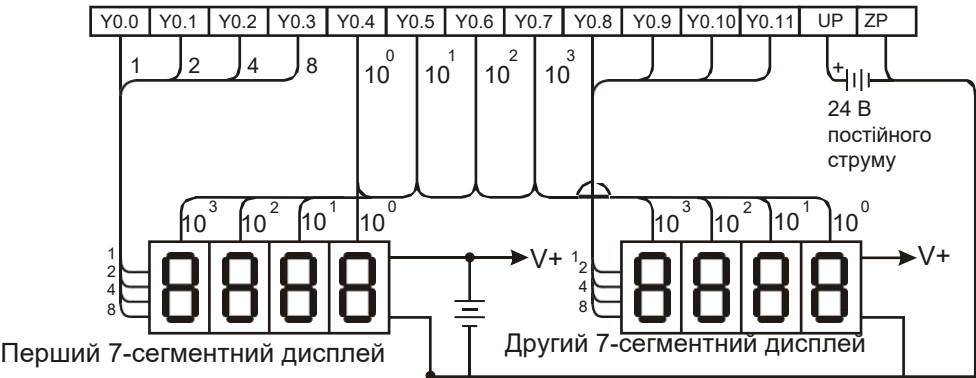
3. Якщо є один чотиризначний семисегментний дисплей, **n** знаходиться в межах 0–3.

- З'єднайте контакти 1, 2, 4 і 8 паралельно, потім підключіть їх до Y0.0–Y0.3 на ПЛК і підключіть засувки до Y0.4–Y0.7 на ПЛК.
- Коли X1.0 увімкнено, інструкція виконується. Y0.4–Y0.7 циклічно вмикаються, а значення в D10 відображається семисегментний дисплей.

4. Якщо є два чотиризначних семисегментних дисплея, **n** знаходиться в межах 4–7.

- З'єднайте контакти 1, 2, 4 і 8 паралельно, потім під'єднайте їх до Y0.8–Y0.11 на ПЛК, а засувки – до Y0.4–Y0.7 на ПЛК.
- Значення в D10 відображається на першому семисегментному дисплеї, а значення в D11 відображається на другому семисегментному дисплеї. Якщо значення в D10 і D11 становлять 1234 і 4321 відповідно, на другому семисегментному дисплеї відображається 1234.

5. Електропроводка:



Примітка. У цьому прикладі використовується транзисторний вихідний модуль AH16AN01T-5A.

Додаткові зауваження

- **n** залежить, чи є вихід позитивним чи негативним виходом, і чи є один чотиризначний семисегментний дисплей або два чотиризначних семисегментних дисплеї .
- Виходи на ПЛК повинні бути NPN транзисторами, колектори яких є відкритими. Крім того, вихід повинен підключити підтягуючий резистор до джерела постійного струму (менше 30 В постійного струму). Тому, коли вихід увімкнено, виводиться сигнал низького потенціалу.
- У наступній таблиці показано негативну логіку.

Двійкове десяткове значення				Вихід (Двійковий десятковий код)				Сигнал			
b ₃	b ₂	b ₁	b ₀	8	4	2	1	A	B	C	D
0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	1	0	0	0	1	1	1	1	0
0	0	1	0	0	0	1	0	1	1	0	1
0	0	1	1	0	0	1	1	1	1	0	0
0	1	0	0	0	1	0	0	1	0	1	1
0	1	0	1	0	1	0	1	1	0	1	0
0	1	1	0	0	1	1	0	1	0	0	1
0	1	1	1	0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	0	0	1	1	1
1	0	0	1	1	0	0	1	0	1	1	0

- Наступна таблиця демонструє позитивну логіку.

Двійкове десятикове значення				Вихід (Двійковий десятиковий код)				Сигнал			
b ₃	b ₂	b ₁	b ₀	8	4	2	1	A	B	C	D
0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	1	1	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	0	0	1	0
0	0	1	1	1	1	0	0	0	0	1	1
0	1	0	0	1	0	1	1	0	1	0	0
0	1	0	1	1	0	1	0	0	1	0	1
0	1	1	0	1	0	0	1	0	1	1	0
0	1	1	1	1	0	0	0	0	1	1	1
1	0	0	0	0	1	1	1	1	0	0	0
1	0	0	1	0	1	1	0	1	0	0	1

- У наступній таблиці показано засув.

Позитивна логіка		Негативна логіка	
Засувка	Сигнал	Засувка	Сигнал
1	0	0	1

- У наступній таблиці показано значення параметра **n**.

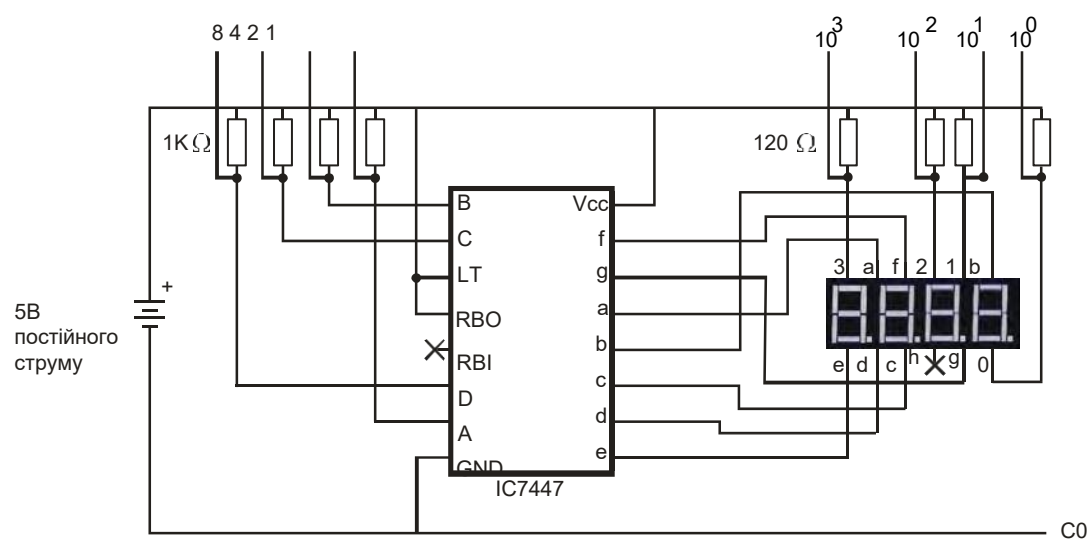
Кількість семисегментних дисплеїв	Один				Два			
Вихід (Двійковий десятиковий код)	+		-		+		-	
Засувка	+	-	+	-	+	-	+	-
n	0	1	2	3	4	5	6	7

' + ': Позитивна логіка

' - ': Негативна логіка

- Ви можете редагувати параметри в **n**, щоб змінити логіку для вихідного транзистора та входу семи-сегментний дисплей.

- На наступному малюнку показано підключення чотиризначного семисегментного дисплея із загальним анодом до IC 7447.



6.19 Інструкції щодо спілкування

6.19.1 Перелік інструкцій щодо спілкування

У наведеній нижче таблиці наведено інструкції зі зв'язку, описані в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
1806	LRC	—	—	Поздовжня перевірка парності
1807	CRC	—	—	Перевірка циклічної надлишковості
1808	MODRW	—	—	Читання та запис даних Modbus
1812	COMRS	—	—	Надсилання та отримання комунікаційних даних
1813	COMDF	—	✓	Встановлення формату зв'язку для послідовного зв'язку порт
1814	VFDRW	—	—	Інструкція по послідовному зв'язку виключно для Delta AC моторні приводи
1815	ASDRW	—	—	Інструкція по послідовному зв'язку виключно для сервоприводу Delta
1816	CCONF	—	✓	Встановлення параметрів у таблиці обміну даними для а комунікаційний порт
1817	MODRWE	—	—	Читання та запис даних Modbus без використання будь-яких прапорів
1818	DNETRW	—	—	Читання та запис даних зв'язку DeviceNet
1819	CANRS	—	—	Надсилання та отримання користувацького зв'язку CAN
1820	DMVSH	—	—	Увімкнення виявлення та зв'язку Delta DMV
1821	DESO	—	—	Виконайте призначений номер зв'язку даних обмінний стіл один раз

6.19.2 Пояснення інструкцій щодо спілкування

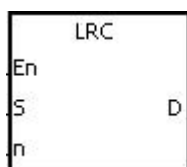
API	Код інструкції			Операнд						функція							
1806		LRC		S, n, D						Поздовжня перевірка парності							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S								●	●							
n								●	●				O	O		
D								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- S** : Перший пристрій, до якого застосовано LRC
- n** : Кількість байтів
- D** : Перший пристрій, де зберігається результат операції

6 Пояснення

- Ця інструкція виконує поздовжню перевірку парності на **n** байтах у пристрої, визначеному **S**. Будь ласка, зверніться до додаткового зауваження до цієї інструкції (нижче), щоб дізнатися більше про контрольний код LRC.
- Операнд **n** має бути парним числом і мати значення від 1 до 1000. Якщо **n** не входить у діапазон, виникає помилка операції, інструкція не виконується, SM0 і SM1 увімкнено, а код помилки в SR0 – 16#200B.
- Режим 16-бітного перетворення: коли SM606 OFF, інструкція розділяє шістнадцяткові дані в пристрої, визначеному **S**, на старші 8-бітні дані та молодші 8-бітні дані. Інструкція застосовує LRC до кожного байта та зберігає результат операції у старших 8-бітах і молодших 8-бітах у пристрої, визначеному **D**. Кількість байтів залежить від **n**.
- Режим 8-бітного перетворення: коли SM606 увімкнено, інструкція розділяє шістнадцяткові дані в пристрої, визначеному **S**, на старші 8-бітні дані (недійсні дані) і молодші 8-бітні дані. Інструкція застосовує LRC до кожного байта та зберігає результат операції в молодшому 8-бітному значенні в двох регістрах. Кількість байтів залежить від **n**. Значення старших 8 бітів у двох регістрах дорівнюють 0.

приклад

1. ПЛК підключено до приводу змінного струму серії VFD-S (режим ASCII SM210 вимкнено; 8-розрядний режим: SM606 увімкнено). ПЛК надсилає команду та зчитує дані в шести пристроях за адресами, починаючи з 16#2101, у приводі змінного струму серії VFD-S.

PLC ⇄ VFD-S

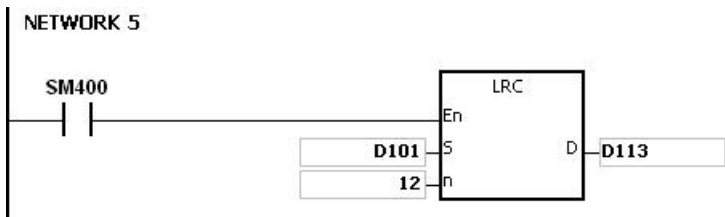
ПЛК надсилає « : 01 03 2101 0006 D4 CR LF».

ПЛК надсилає дані в наступній таблиці.

регістр	Дані		опис	
D100 Низькі 8 біт	' : '	16#3A	STX	
D101 Низькі 8 біт	'0'	16#30	ADR 1	AD (1, 0) – адреса станції приводу двигуна змінного струму.
D102 Низькі 8 біт	'1'	16#31	ADR 0	
D103 Низькі 8 біт	'0'	16#30	CMD 1	CMD (10) — код команди.
D104 Низькі 8 біт	'3'	16#33	CMD 0	
D105 Низькі 8 біт	'2'	16#32	Початкова адреса даних	
D106 Низькі 8 біт	'1'	16#31		
D107 Низькі 8 біт	'0'	16#30		
D108 Низькі 8 біт	'1'	16#31		
D109 Низькі 8 біт	'0'	16#30	Кількість даних (підраховується словом)	
D110 Низькі 8 біт	'0'	16#30		
D111 Низькі 8 біт	'0'	16#30		
D112 Низькі 8 біт	"6"	16#36		
D113 Низькі 8 біт	"D"	16#44	LRC CHK 0	LRC CHK (01) — це код перевірки помилок.
D114 Низькі 8 біт	'4'	16#34	LRC CHK 1	
D115 Низькі 8 біт	CR	16#0D	END	
D116 Низькі 8 біт	LF	16#0A		

LRC CHK (01) вище є кодом перевірки помилок. Ви можете використовувати інструкцію LRC, щоб обчислити його (8-розрядний режим:

SM606 увімкнено).



Контрольний код LRC: 16#01+16#03+16#21+16#01+16#00+16#06=16#2C
Доповненням до 16#2C є 16#D4. «D» (16#44) зберігається в молодшому 8-бітному коді D113, а «4» (16#34) зберігається в молодшому 8-бітному коді D114.

Додаткові зауваження

1. У наступній таблиці наведено формат даних зв'язку в режимі ASCII.

STX	' : '	Початковий символ тексту — « : » (16#3A).
Адреса Привіт	'0'	Адреса зв'язку:
Адреса Lo	'1'	8-бітна адреса складається з двох кодів ASCII.
Функція Привіт	'0'	Код функції:
Функція Lo	'3'	8-бітний код функції складається з двох кодів ASCII.
ДАНІ (n-1) ДАНІ 0	"2"	Дані: n×8-бітові дані складаються з 2n кодів ASCII.
	'1'	
	'0'	
	"2"	
	'0'	
	'0'	
	'0'	
	"2"	
LRC CHK Привіт	'D'	Контрольний код LRC:
LRC CHK Lo	'7'	8-бітний контрольний код складається з двох кодів ASCII.
END Привіт	CR	Символ кінця тексту:
END Lo	LF	END Hi=CR (16#0D), END Lo=LF (16#0A)

2. Контрольний код LRC: інструкція додає значення, починаючи від адреси зв'язку до даних, а потім інструкція обчислює доповнення до суми, яка є кодом перевірки LRC.

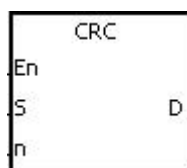
Приклад 16#01+16#03+16#21+16#02+16#00+16#02=16#29
Доповненням до 16#29 є 16#D7.

API	Код інструкції				Операнд					функція						
1807		CRC			S, n, D					Перевірка циклічної надлишковості						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S								●	●							
n								●	●				○	○		
D								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- S** : Перший пристрій, до якого застосовано CRC
- n** : Кількість байтів
- D** : Перший пристрій, в якому зберігається результат операції

Пояснення

- Ця інструкція виконує циклічну перевірку надлишковості **n** байтів, починаючи з пристрою, заданого **S**. Будь ласка, зверніться до Додаткове зауваження до цієї інструкції (нижче) для отримання додаткової інформації про контрольний код CRC.
- Операнд **n** має бути між 1–1000. Якщо **n** не в діапазоні, виникає помилка операції, інструкції немає виконано, SM0 і SM1 увімкнені, а код помилки в SR0 – 16#200B.
- Режим 16-бітного перетворення: коли SM606 OFF, інструкція розділяє шістнадцяткові дані в пристрої, визначеному **S**, на старші 8-бітні дані та молодші 8-бітні дані. Інструкція застосовує CRC до кожного байта, і зберігає результат операції у старших 8-бітах і молодших 8-бітах у пристрої, визначеному **D**. Кількість байтів залежить від **n**.
- Режим 8-бітного перетворення: коли SM606 увімкнено, інструкція розділяє шістнадцяткові дані в пристрої, визначеному **S**, на старші 8-бітні дані (недійсні дані) і молодші 8-бітні дані. Інструкція застосовує CRC до кожного байта та зберігає результат операції в молодших 8 бітах у двох регістрах. Кількість байтів залежить від **n**.

приклад

1. ПЛК підключено до приводу змінного струму серії VFD-S (режим RTU SM210 увімкнено; 16-розрядний режим: SM606 увімкнено). Значення 16#12, яке має бути записане в пристрій на 16#2000 в приводі змінного струму серії VFD-S, спочатку записується в пристрій у ПЛК.

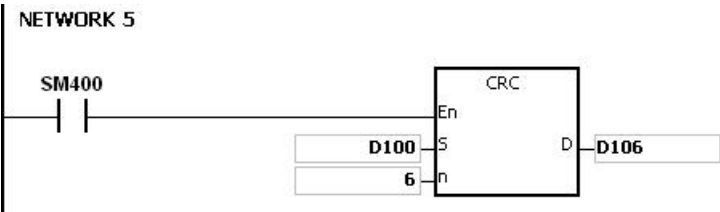
PLC ⇄ VFD-S

PLC надсилає 01 06 2000 0012 02 07.

ПЛК надсилає дані в наступній таблиці.

регістр	Дані	опис
D100 Низькі 8 біт	16#01	Адреса
D101 Низькі 8 біт	16#06	функція
D102 Низькі 8 біт	16#20	Адреса даних
D103 Низькі 8 біт	16#00	
D104 Низькі 8 біт	16#00	Дані
D105 Низькі 8 біт	16#12	
D106 Низькі 8 біт	16#02	CRC CHK 0
D107 Низькі 8 біт	16#07	CRC CHK 1

CRC CHK (01) вище є кодом перевірки помилок. Ви можете обчислити його за допомогою інструкції CRC (8-розрядний режим: SM606 увімкнено).



Контрольний код CRC: 16#02 зберігається в молодшому 8-бітному коді D106, а 16#07 зберігається в молодшому 8-бітному коді D107.

Додаткові зауваження

1. У наступній таблиці показано формат даних зв'язку в режимі RTU.

START	Проміжок часу
Адреса	Адреса зв'язку: 8-бітна двійкова адреса
функція	Функціональний код: 8-бітний двійковий код
ДАНІ (n-1)	Дані: $n \times 8$ -бітні дані
.....	
ДАНІ 0	
CRC CHK Низький	Контрольний код CRC: 16-бітний контрольний код складається з двох 8-бітних двійкових кодів.
CRC CHK Високий	
END	Проміжок часу

2. Контрольний код CRC: контрольний код починається від адреси до даних. Правило роботи показано в наступній таблиці.

- Крок 1: Припустимо, що дані в 16-бітному регістрі (регістрі, де зберігається контрольний код CRC) мають значення 16#FFFF.
- Крок 2: Логічний оператор XOR приймає перше 8-бітне повідомлення та молодші 8-бітні дані в 16-бітному регістрі та виконує логічну операцію виключаючого АБО над кожною парою відповідних бітів. Результат операції зберігається в 16-розрядному регістрі.
- крок 3: Значення розрядів у 16-розрядних регістрах зсуваються на один біт вправо. Значення старшого біта стає 0.
- крок 4: Якщо значення крайнього правого біта, який зсувається вправо, дорівнює 0, дані з кроку 3 зберігаються в 16-бітному регістрі. В іншому випадку логічний оператор XOR приймає 16#A001 і дані в 16-бітному регістрі та виконує операцію логічного виключаючого АБО над кожною парою відповідних бітів. Результат операції зберігається в 16-розрядному регістрі.
- крок 5: Повторіть кроки 3 і 4 і виконайте операцію з 8-бітним повідомленням.
- Крок 6: Повторіть кроки 2–5, а потім отримайте наступне 8-бітне повідомлення. Виконайте операції з усіма повідомленнями. Кінцевим результатом у 16-розрядному регістрі є контрольний код CRC. Зауважте, що молодші 8-бітові дані в 16-бітному регістрі замінюються старшими 8-бітовими даними в 16-бітовому регістрі перед тим, як контрольний код CRC буде вставлено в контрольний код повідомлення.

API	Код інструкції			Операнд						функція						
1808		MODRW		S_1, S_2, S_3, S, n						Читання та запис даних Modbus						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S								●								
n								●	●				○	○		

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S	●	●				●							
n		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

MODRW
En
S1
S2
S3
S
n

- S_1 : Адреса підрозділу
- S_2 : Код функції
- S_3 : Адреса пристрою
- S : Зареєструватися для читання/запису даних
- n : Довжина даних

Пояснення

1. Операнд S_1 має бути між 0–254; 0 – режим трансляції.
2. S_2 — код функції.

У наступній таблиці показано приклад.

функція код	опис	Дані довжина	Пристрої, які підтримують пристрої для підлеглих
16#01	ПЛК зчитує дані з кількох розрядних пристроїв.	1-1600	X, Y, M, SM, S, T, C, HC
16#02	ПЛК зчитує дані з кількох розрядних пристроїв.	1-1600	X, Y, M, SM, S, T, C, HC
16#03	PLC зчитує дані з кількох слів пристроїв.	1-100	X, Y, SR, D, T, C, HC, E

функція код	опис	Дані довжина	Пристрої, які підтримують пристрої для підлеглих
16#04	PLC зчитує дані з кількох слів пристроїв.	1-100	X
16#05	PLC записує стан у бітовий пристрій.	1	Y, M, SM, S, T, C, HC
16#06	ПЛК записує дані в пристрій Word.	1	Y, SR, D, T, C, HC, E
16#0F	ПЛК записує стани в кілька розрядних пристроїв.	1-1600	Y, M, SM, S, T, C, HC
16#10	PLC записує дані в кілька слів пристроїв.	1-100	Y, SR, D, T, C, HC, E

Ця інструкція підтримує лише коди функцій, згадані вище, і не може виконувати інші коди функцій.

Перегляньте наведені нижче приклади.

3. **S₃** – адреса пристрою. Якщо адреса недійсна для призначеного пристрою зв'язку, зв'язок

пристрій відповідає повідомленням про помилку. Наприклад, адреса пристрою 16#8000 недійсна в DVP-ES2.

4. **S** – це регістр, який бере участь у читанні/записі даних.

Дані для запису в зовнішнє обладнання зберігаються в реєстрі заздалегідь.

Дані для зчитування із зовнішнього обладнання зберігаються в реєстрі.

5. **N** — довжина даних

Для кодів функцій зв'язку словесного типу довжина даних не може перевищувати 100 слів.

Для кодів функцій зв'язку бітового типу (BOOL) довжина даних становить від 1 до 1600 біт.

6. У наведеній нижче таблиці показано, як функції **S₃**, **S** та **n** змінюються залежно від використовуваного коду функції.

функція код	S₃	S	n
16#01	Адреса, звідки дані зчитуються	Зареєструватися там, де зчитані дані зберігаються	Довжина зчитування даних
16#02	Адреса, звідки дані зчитуються	Зареєструватися там, де зчитані дані зберігаються	Довжина зчитування даних
16#03	Адреса, звідки дані зчитуються	Зареєструватися там, де зчитані дані зберігаються	Довжина зчитування даних
16#04	Адреса, звідки дані зчитуються	Зареєструватися там, де зчитані дані зберігаються	Довжина зчитування даних
16#05	Адреса, де	Записано значення статусу	Жодного сенсу

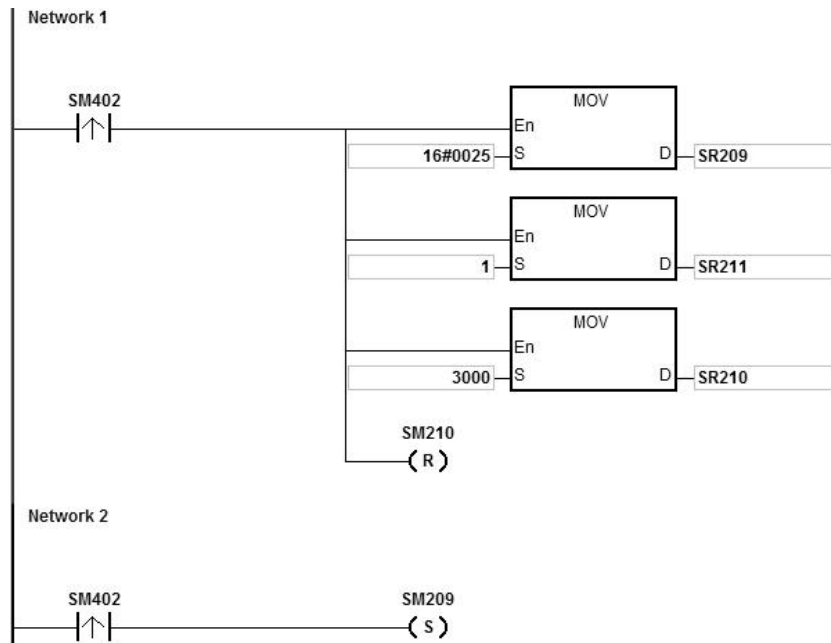
функція код	S_3	S	p
	записуються дані		
16#06	Адреса, де записуються дані	Зареєструватися там, де записані дані зберігаються	Жодного сенсу
16#0F	Адреса, де записуються дані	Зареєструватися там, де записані дані зберігаються	Довжина записаних даних
16#10	Адреса, де записуються дані	Зареєструватися там, де записані дані зберігаються	Довжина записаних даних

7. Кількість разів використання цієї інструкції не обмежена; однак може бути лише одна інструкція виконується на тому самому COM-порту одночасно. Перед виконанням цієї інструкції вам потрібно вирішити та використовувати прапор надсилання для COM. В іншому випадку дані з 2 різних комунікаційних COM можуть бути переплутані.
8. Якщо відбувається тайм-аут зв'язку, прапорці тайм-ауту ввімкнені. Після вирішення проблеми потрібно скинути прапорці часу очікування на ВІМК. Під час використання цієї інструкції значення часу очікування не може дорівнювати 0. Встановіть значення між 100–32767 мс; коли значення встановлено на 0, воно обробляється як 200 мс.
9. У режимі Modbus ASCII вам потрібно налаштувати лише дані (режим не ASCII) для передачі. Інструкція конвертує режим не-ASCII у режим ASCII, що складається з головного коду (:), перетвореного коду ASCII, контрольної суми (LRC) і кінцевого коду (CRLF). Інструкція зберігає отримані дані в символі ASCII у внутрішньому регістрі. ПЛК автоматично перетворює дані в шістнадцяткове значення, і якщо дані зв'язку правильні, зберігає результат перетворення в S . і встановлює для прапора завершення SM значення ON.
10. У режимі Modbus RTU вам потрібно налаштувати лише дані для передачі. Інструкція додає контрольну суму (CRC) і зберігає отримані дані в символах ASCII у внутрішньому регістрі. ПЛК автоматично перетворює дані в шістнадцяткове значення, і якщо дані зв'язку правильні, зберігає результат перетворення, збережений у S .
11. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.

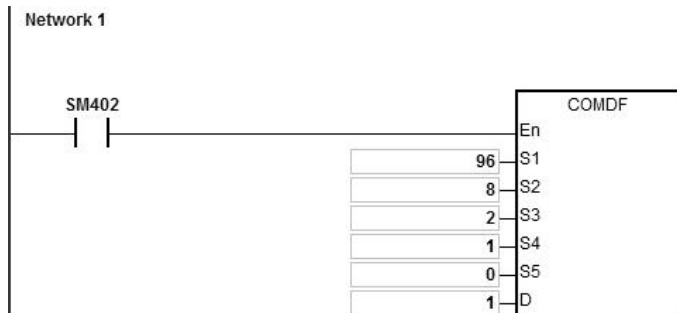
Приклад налаштування протоколу зв'язку

1. У наступних прикладах використовується порт зв'язку ПЛК 1 і спеціальні регістри, щоб продемонструвати, як налаштувати протокол зв'язку.
2. Ви можете налаштувати порт зв'язку ПЛК за допомогою HWCONFIG в ISPSOFT або за допомогою відповідних спеціальних регістрів, або ви можете використати інструкцію COMDF (API 1813) для налаштування зв'язку. Зверніться до посібника ISPSOFT, щоб налаштувати його в HWCONFIG. Для налаштувань регістру зв'язку (SM, SR) зверніться до розділу 6.19.3 для отримання додаткової інформації.

3. Налаштування зв'язку для цього прикладу – RS485 ASCII, 9600, 8, E, 1 (SR209=16#0025).
4. Встановить час очікування зв'язку на 3000 мс (SR210=3000).
5. Встановить режим зв'язку на режим ASCII (SM210=ВИМК.).
6. Увімкніть протокол зв'язку (SM209=ON).



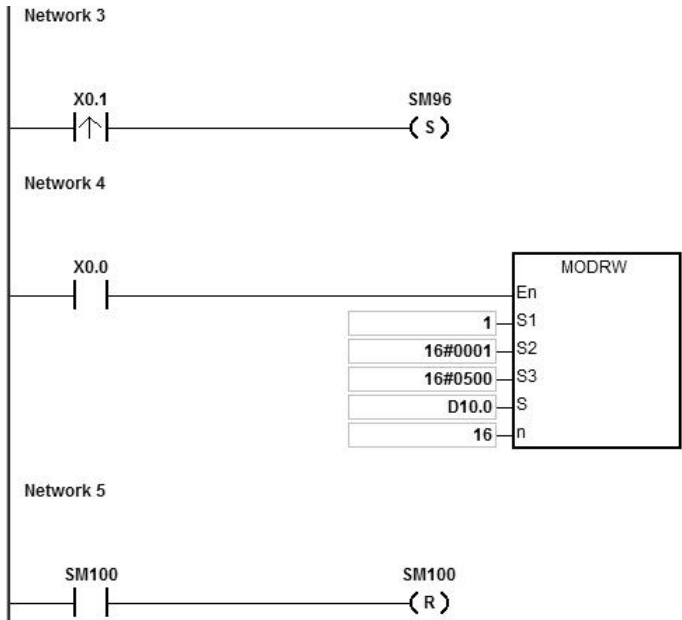
Якщо ви налаштували порт зв'язку за допомогою інструкції COMDF (API 1813), ви можете проігнорувати цей крок.



Якщо ви налаштували порт зв'язку в ISPSOft - > HWCONFIG - > COM Port, ви можете проігнорувати цей крок.

Приклад 1

1. Код функції 01 (16#01): ПЛК зчитує дані з кількох бітових пристроїв, які не є дискретними пристроями введення (у цьому прикладі зчитується 16 фрагментів даних). Для коду функції 02 дія така ж, як і для коду функції 01.



Module ЦП серії AS підключається до ПЛК серії DVP-ES2.
Коли SM96 і X0.0 увімкнено, ЦП серії AS надсилає та отримує команди Y0–Y17 від DVP-ES2. Коли адреса Y0 становить 16#0500, стани Y0–Y17 у DVP-ES2 перераховані в наступній таблиці.

пристрій	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
Держава	ON	ON	OFF	ON	OFF	OFF	ON	OFF
Значення	D				2			
пристрій	Y17	Y16	Y15	Y14	Y13	Y12	Y11	Y10
Держава	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
Значення	0				4			

У наведеній нижче таблиці наведено операнди для інструкції MODRW.

Операнд	опис	пристрій
S ₁	Адреса підрозділу	16#0001
S ₂	Код функції	16#0001
S ₃	Адреса пристрою	16#0500
S	Зареєструватися для читання та запису даних	D10.0
n	Довжина даних	16

Режим ASCII

Вам не потрібно перетворювати коди ASCII, і всі вони виражені у значеннях 16#.

●AS надсилає команду зв'язку: « : 01 01 05 00 00 10 E9 CR LF».

●AS отримує команду зв'язку: « : 01 01 02 D2 04 26 CR LF».

Режим RTU

●AS надсилає команду зв'язку: «01 01 05 00 00 10 3D 0A».

●AS отримує команду зв'язку: «01 01 02 D2 04 E4 9F».

Якщо формат правильний, SM100 увімкнено.

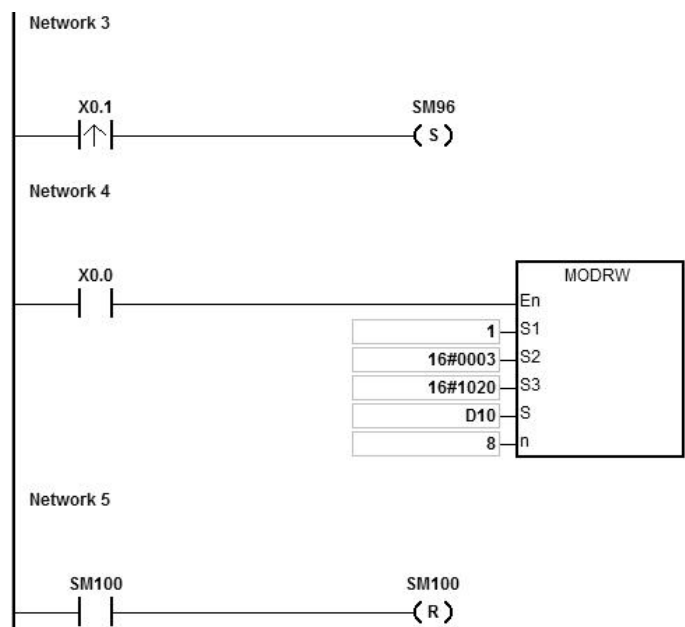
2. Відповідні повідомлення від DVP-ES2 зберігаються в регістрах від D10.0 до 10.15 (зчитування даних D10.15–D10.0=16#04D2).

пристрій	D10.7	D10.6	D10.5	D10.4	D10.3	D10.2	D10.1	D10.0
Держава	ON	ON	OFF	ON	OFF	OFF	ON	OFF
Значення	D				2			
пристрій	D10.15	D10.14	D10.13	D10.12	D10.11	D10.10	D10.9	D10.8
Держава	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
Значення	0				4			

3. Після отримання даних, надісланих назад від DVP-ES2, ПЛК підтверджує формат даних, надісланих назад від DVP-ES2 і визначає, чи це правильно. Якщо у форматі немає помилок, відповідні спеціальні прапорці SM100 увімкнено; якщо ні, SM102 увімкнено.

Приклад 2

1. Код функції 03 (16#03): ПЛК зчитує дані з кількох бітових пристроїв, які не є дискретними пристроями введення (у цьому прикладі зчитується вісім фрагментів даних). Для коду функції 04 дія така ж, як і для коду функції 03.



2. Module центрального процесора серії AS підключено до ПЛК серії DVP-ES2.
Коли SM96 і X0.0 увімкнено, модуль ЦП серії AS надсилає та отримує D32–D39 від DVP-ES2.
3. Якщо адреса D32 дорівнює 16#1020, значення D32–D39 у DVP-ES2 наведено в наступній таблиці.

пристрій	D32	D33	D34	D35	D36	D37	D38	D39
Значення (16#)	1234	5678	1122	3344	5566	7788	99AA	BBCC

У наведеній нижче таблиці наведено операнди інструкції MODRW.

Операнд	опис	пристрій
S ₁	Адреса підрозділу	16#0001
S ₂	Код функції	16#0003
S ₃	Адреса пристрою (D32)	16#1020
S	Реєстр, задіяний для читання та запису даних	D10
n	Довжина даних	8

Режим ASCII

Вам не потрібно перетворювати коди ASCII, і всі вони виражаються значеннями 16#.

● AS надсилає команду зв'язку: « : 01 03 10 20 00 08 C4 CR LF».

● AS отримує команду зв'язку: « : 01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC AA CR LF».

Режим RTU

● AS надсилає команду зв'язку: «01 03 10 20 00 08 41 06».

● AS отримує команду зв'язку: «01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC 90 FE».

Якщо формат правильний, SM100 увімкнено.

● Відповідні повідомлення від DVP-ES2 зберігаються в регістрах від D10 до D17.

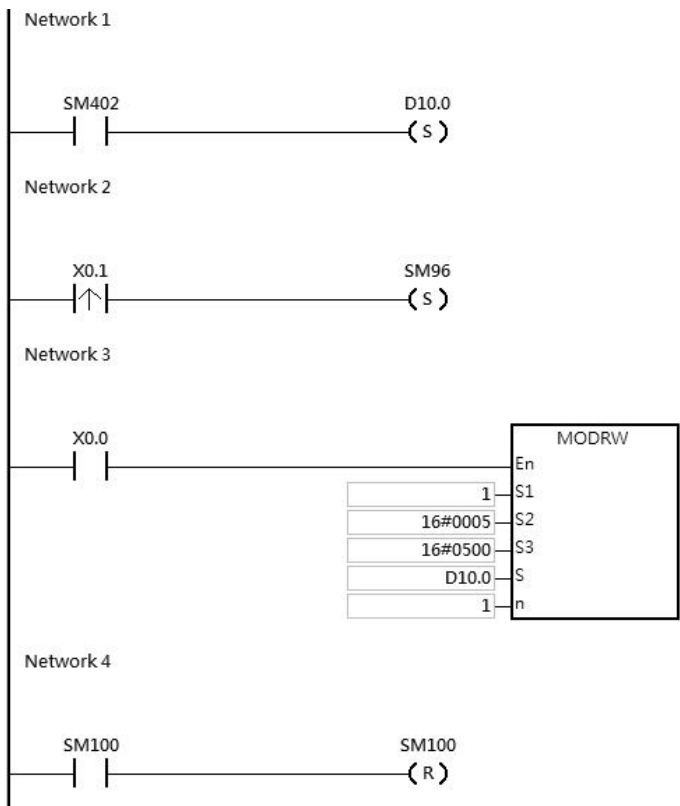
● У таблиці нижче наведено значення в D10–D17.

пристрій	D10	D11	D12	D13	D14	D15	D16	D17
Значення (16#)	1234	5678	1122	3344	5566	7788	99AA	BBCC

4. Після отримання даних, надісланих назад від DVP-ES2, ПЛК підтверджує формат даних, надісланих назад від DVP-ES2 і визначає, чи це правильно. Якщо у форматі немає помилок, відповідні спеціальні прапорці SM100 увімкнено; якщо ні, SM102 увімкнено.

Приклад 3

1. Код функції 05 (16#05): ПЛК записує стан у бітовий пристрій. У цьому прикладі пристрій увімкнено.



2. Module ЦП серії AS підключено до ПЛК серії DVP-ES2. D10.0 увімкнено, а Y0 у серії DVP-ES2 PLC також увімкнено. Коли SM96 і X0.0 увімкнені, ПЛК може встановити стан Y0.

У наведеній нижче таблиці наведено операнди для інструкції MODRW.

Операнд	опис	пристрій
S ₁	Адреса підрозділу	1
S ₂	Код функції	16#0005
S ₃	Адреса пристрою	16#0500
S	Зареєструватися для читання та запису даних	D10.0
n	Довжина даних (не використовується з цим кодом функції)	1

Режим ASCII

Наведені нижче номери лише для довідки. Замість показу значень у кодах ASCII тут вирази показано у значеннях 16#.

- AS надсилає команду зв'язку: « : 01 05 05 00 FF 00 F6 CR LF»
- AS отримує команду зв'язку: « : 01 05 05 00 FF 00 F6 CR LF»

Режим RTU

● AS надсилає команду зв'язку: «01 05 05 00 FF 00 8C F6»

● AS отримує команду зв'язку: «01 05 05 00 FF 00 8C F6». Якщо

формат правильний, SM100 увімкнено.

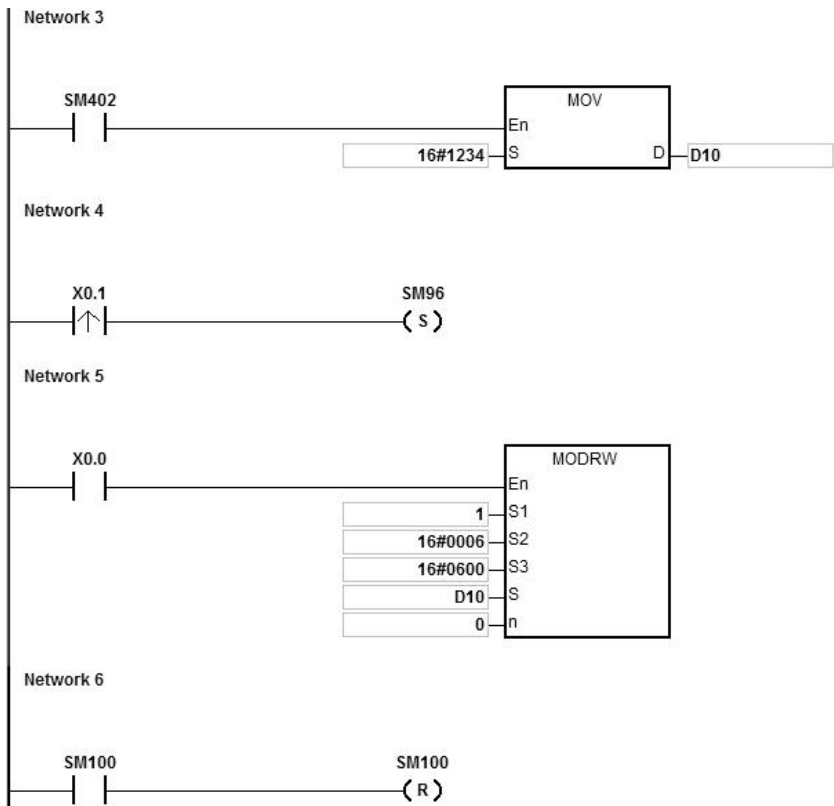
3. Після отримання даних від DVP-ES2 ПЛК підтверджує формат даних, надісланий назад від DVP-ES2, і визначає, чи він правильний. Якщо у форматі немає помилок, відповідні спеціальні прапорці SM100 увімкнено; якщо ні, SM102 увімкнено.

4. Коли DVP-ES2 отримує цю інструкцію, Y0 увімкнено.

5. Оскільки цей код функції записує дані, операнд *n* ігнорується.

Приклад 4

1. Код функції 06 (16#06): ПЛК записує стан у текстовий пристрій.



2. Module центрального процесора серії AS підключається до ПЛК серії DVP-ES2.

3. Припустимо, що D10 — це 16#55AA (очікування запису даних на пристрій T0 DVP-ES2).

Коли SM96 і X0.0 увімкнені, ПЛК може записувати дані в T0 ПЛК серії DVP-ES2. Адреса T0 – 16#0600.

У наведеній нижче таблиці наведено операнди для інструкції MODRW.

Операнд	опис	пристрій
S ₁	Адреса підрозділу	1
S ₂	Код функції	16#0006
S ₃	Адреса пристрою T0	16#0600
S	Регістр T0 для читання та запису даних	D10
n	Довжина даних (не використовується з цим кодом функції)	0

Режим ASCII

Вам не потрібно перетворювати коди ASCII, і всі вони виражаються значеннями 16#.

●AS надсилає команду зв'язку: « : 01 06 06 00 55 AA F4 CR LF»

●AS отримує команду зв'язку: « : 01 06 06 00 55 AA F4 CR LF»

Режим RTU

●AS надсилає команду зв'язку: «01 06 06 00 55 AA 36 6D»

●AS отримує команду зв'язку: «01 06 06 00 55 AA 36 6D». Якщо

формат правильний, SM100 увімкнено.

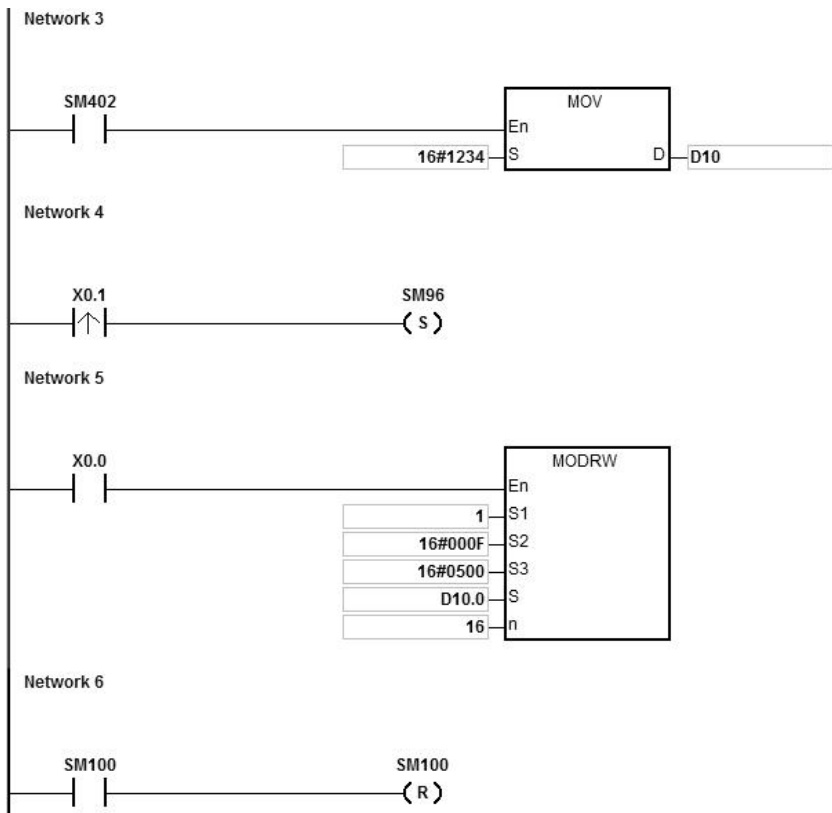
4. Після отримання даних від DVP-ES2 ПЛК підтверджує формат даних, надісланий назад від DVP-ES2, і визначає, чи він правильний. Якщо у форматі немає помилок, відповідні спеціальні прапорці SM100 увімкнено; якщо ні, SM102 увімкнено.

5. Коли DVP-ES2 отримує цю інструкцію, він записує дані, що зберігаються в пристрої D10, на пристрій T0 у DVP-ES2.

6. Оскільки цей код функції записує дані, операнд *n* ігнорується.

Приклад 5

1. Код функції 0F (16#0F): ПЛК записує стани в кілька бітових пристроїв.



6

2. Module центрального процесора серії AS підключається до ПЛК серії DVP-ES2.

● Припустимо, D10.15-D10.0=16#04D2 (очікування на запис стану Y0-Y17 DVP-ES2)

пристрій	D10.7	D10.6	D10.5	D10.4	D10.3	D10.2	D10.1	D10.0
Держава	ON	ON	OFF	ON	OFF	OFF	ON	OFF
Значення	D				2			
пристрій	D10.15	D10.14	D10.13	D10.12	D10.11	D10.10	D10.9	D10.8
Держава	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
Значення	0				4			

Коли SM96 і X0.0 увімкнені, ПЛК може встановити стан Y0-Y17 у DVP-ES2. Адреса Y0 є 16#0500.

У наведеній нижче таблиці наведено операнди для інструкції MODRW.

Операнд	опис	пристрій
S_1	Адреса підрозділу	1
S_2	Код функції	16#000F
S_3	Адреса пристрою Y0	16#0500
S	Регістри Y0–Y17 для читання та запису даних	D10.0
n	Довжина даних	16

Режим ASCII

Вам не потрібно перетворювати коди ASCII, і всі вони виражаються значеннями 16#.

● AS надсилає команду зв'язку: « : 01 0F 0500 0010 02 D2 04 03 CR LF»

● AS отримує команду зв'язку: « : 01 0F A0 00 00 10 40 CR LF»

Режим RTU

● AS надсилає команду зв'язку: «01 0F 05 00 00 10 02 D2 04 EA 43»

● AS отримує команду зв'язку: «01 0F A0 00 00 10 76 07». Якщо формат

правильний, SM100 увімкнено.

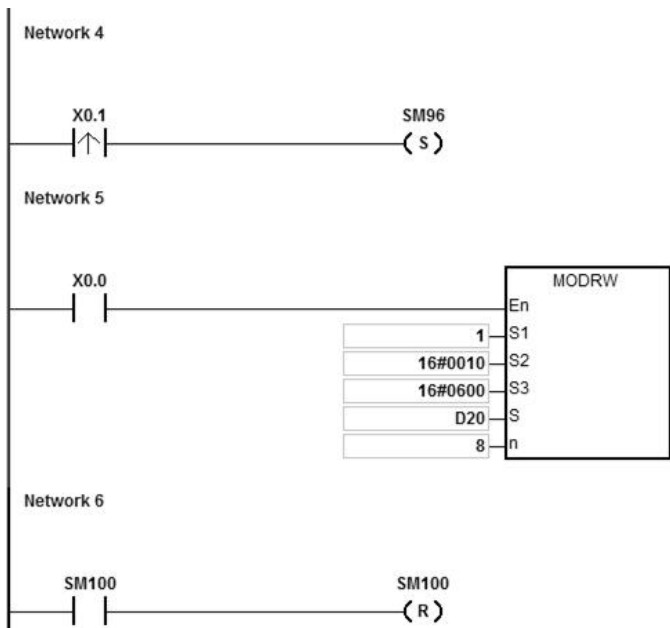
3. Після отримання даних, надісланих назад від DVP-ES2, ПЛК підтверджує формат даних, надісланих назад від DVP-ES2 і визначає, чи це правильно. Якщо у форматі немає помилок, відповідні спеціальні прапорці SM100 увімкнено; якщо ні, SM102 увімкнено.

пристрій	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
Держава	ON	ON	OFF	ON	OFF	OFF	ON	OFF
Значення	D				2			
пристрій	Y17	Y16	Y15	Y14	Y13	Y12	Y11	Y10
Держава	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
Значення	0				4			

4. Оскільки цей код функції записує дані, операнд n ігнорується.

Приклад 6

1. Функціональний код 10 (16#10): ПЛК записує стани в кілька пристроїв слова.



2. Module ЦП серії AS підключається до ПЛК серії DVP-ES2.
3. Припустимо, що значення для D20–27 наведено в наступній таблиці (очікування запису даних на пристрої T0–7 DVP-ES2).

пристрій	D20	D21	D22	D23	D24	D25	D26	D27
Значення (16#)	1234	5678	1122	3344	5566	7788	99AA	BBCC

Коли SM96 і X0.0 увімкнені, ПЛК може записувати дані в T0–7 ПЛК серії DVP-ES2. Адреса T0 є 16#0600.

У наведеній нижче таблиці наведено операнди інструкції MODRW.

Операнд	опис	пристрій
S ₁	Адреса підрозділу	1
S ₂	Код функції	16#0010
S ₃	Адреса пристрою T0	16#0600
S	Регістр T0–7 для читання та запису даних	D20
n	Довжина даних (не використовується з цим кодом функції)	8

Режим ASCII

Вам не потрібно перетворювати коди ASCII, і всі вони виражаються значеннями 16#.

● AS надсилає команду зв'язку: « : 01 10 0600 00 08 10 1234 5678 1122 3344 5566 7788 99AA BBCC 8F CR LF»

● AS отримує команду зв'язку: « : 01 10 06 00 00 08 E1 CR LF»

Режим RTU

● AS надсилає команду зв'язку: «01 10 06 00 00 08 10 1234 5678 1122 3344 5566 7788 99AA BBCC 0B 0C»

● AS отримує команду зв'язку: «01 10 06 00 00 08 C1 47». Якщо формат

правильний, SM100 увімкнено.

5. Після отримання даних, надісланих з DVP-ES2, ПЛК підтверджує формат даних, надісланих з DVP-ES2, і визначає, чи він правильний. Якщо у форматі немає помилок, відповідні спеціальні прапорці SM100 увімкнено; якщо ні, SM102 увімкнено. Коли DVP-ES2 отримує цю інструкцію, він записує дані, що зберігаються в пристроях D20–27, на пристрій T0–7 у DVP-ES2.

пристрій	T0	T1	T2	T3	T4	T5	T6	T7
Значення (16#)	1234	5678	1122	3344	5566	7788	99AA	BBCC

6. Оскільки цей код функції записує дані, операнд **n** ігнорується.

Додаткові зауваження

1. Якщо значення в **S**₁ або **S**₂ перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо пристрою, зазначеного **S**, недостатньо для зберігання **n** фрагментів даних, інструкція не виконується, SM0 є ON, а код помилки в SR0 – 16#2003.
3. Якщо **n** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
4. Якщо функціональний код, визначений **S**₂, відноситься до бітових пристроїв, пристрій, визначений **S**, має бути бітовим пристроєм;
інакше виникає помилка операції, інструкція не виконується, а код помилки в SR0 — 16#2003.
5. Якщо функціональний код, визначений **S**₂, відноситься до словесних пристроїв, пристрій, визначений **S**, повинен бути словесним пристроєм;
інакше виникає помилка операції, інструкція не виконується, а код помилки в SR0 — 16#2003.

6. Якщо комунікаційна команда 0x05 або 0x06, значення в **n** можна проігнорувати. Довжина даних становить лише один біт або одне слово.
7. Інструкція MODRW не виконується, якщо прапори надсилання SM96 і SM97 не ввімкнено.
8. Якщо відбувається тайм-аут зв'язку, прапорці тайм-ауту SM104 і SM105 увімкнуті, а прапорці отримання SM98 і SM99 вимкнуті.
9. Якщо під час отримання даних виникає помилка, прапорці помилки SM102 і SM103 увімкнуті, а прапорці отримання SM98 і SM99 вимкнені.
10. Якщо функціональний код, визначений **S₂**, пов'язаний із текстовими пристроями, пристрій у зовнішньому обладнанні, з яким ПЛК спілкується, має бути текстовим пристроєм. Якщо функціональний код, визначений **S₂**, відноситься до бітових пристроїв, пристрій у зовнішньому обладнанні, з яким ПЛК спілкується, має бути бітовим пристроєм.
11. Будь ласка, зверніться до розділу 6.19.3, щоб дізнатися більше про налаштування регістру зв'язку (SM, SR).

API	Код інструкції			Операнд				функція								
1812		COMRS		S_1, S_2, S_3, D_1, D_2				Надсилання та отримання комунікаційних даних								
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●	●				○	○		
S_2								●	●							
S_3								●	●				○	○		
D_1								●								
D_2								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
D_1		●				●							
D_2		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

COMRS
En
S1
S2
S3

S_1 :Номер комунікаційного порту (1–2, 11–12)

S_2 :Джерело даних для надсилання

S_3 :Довжина даних для надсилання

D_1 :Перший пристрій, де зберігаються отримані комунікаційні дані

D_2 Умова припинення отримання даних

Пояснення

- S_1 – номер порту зв'язку: COM1 – номер 1, COM 2 – номер 2, карта 1 – номер 11, а карта 2 – номер 12. Якщо дані виходять за межі діапазону комунікаційного порту, інструкція не виконує надсилання чи отримання.

- Якщо ви використовуєте певний символ або символи для завершення отримання даних, пропонується застосувати інструкцію до

Дані ASCII. Якщо ви не застосовуєте інструкцію до даних ASCII, пропонується використовувати період очікування, щоб завершити отримання даних.

- S_2 є джерелом даних, які потрібно надіслати.

S_3 — довжина даних, які потрібно надіслати.

Якщо S_2 дорівнює D100, а S_3 дорівнює 10, інструкція надсилає значення в молодших байтах у D100–D109 через порт зв'язку, визначений S_1 .

4. Рядки не надсилаються, якщо значення налаштування в S_3 дорівнює 0. Максимальна кількість символів, яку можна надіслати, становить 256 слів.

5. D_1 — довжина отриманих даних.

$D_1 + 1 - D_1 + n$ — це пристрої для зберігання отриманих даних

Якщо D_1 дорівнює D200, значення в D_2 дорівнює 3, а значення в $D_2 + 1$ дорівнює 16#0D0A, інструкція зберігає отримані дані в молодших байтах у пристроях, починаючи з D201 (старші байти не змінюються). Інструкція продовжує отримувати дані, доки не отримає послідовні символи зупинки 16#0D і 16#0A. Інструкція записує довжину даних, отриманих у D200 після отримання 16#0D і 16#0A, і встановлює прапор завершення на ON після припинення отримання даних.

6. D_2 — режим отримання даних

$D_2 + 1$ — умова, яка завершує отримання даних

D_2 і $D_2 + 1$ описані в наступній таблиці.

D_2	Режим отримання даних	Значення налаштування в $D_2 + 1$	Зауваження
0	Не отримує даних зв'язку	Невикористаний	Після надсилання даних є завершення, установіть прапор завершення на ON.
1	Коли час після останньої отриманої частини даних перевищує час, встановлений у $D_2 + 1$, отримання даних завершено.	Значенням налаштування в $D_2 + 1$ є час. Одиницю вимірювання є 1 мілісекунда. Значення налаштування в $D_2 + 1$ становить 5–3000.	Якщо встановлений вами час є перевищує 3000 мілісекунд, значення в $D_2 + 1$ дорівнює 3000. Якщо встановлений час менший за 5 мілісекунд, значення в $D_2 + 1$ дорівнює 5.
2	Отримані дані закінчуються на а специфічний характер.	Значення налаштування в $D_2 + 1$ є певним символом.	Якщо конкретний символ 16#0A, значення в $D_2 + 1$ становить 16#000A.
3	Отримані дані закінчуються два послідовних конкретних символи.	Значення налаштування в $D_2 + 1$ — це два певних символи.	Якщо є два конкретних символи 16#0D і 16#0A, значення в $D_2 + 1$ дорівнює 16#0D0A.
4	Отримані дані починаються з певного символу. Коли час після останнього фрагмента даних	Певний символ зберігається в старший байт в $D_2 + 1$, а час зберігається в молодшому байті в	Якщо початковий символ 16#3A, а час становить 15 мілісекунд, значення в $D_2 + 1$ дорівнює 16#3A0F.

	отримано перевищує час встановлено в $D_2 + 1$, отримання даних завершено.	$D_2 + 1$. Час встановлено в низ байт у $D_2 + 1$ знаходиться в діапазоні 5–255 мілісекунд.	
5	Отримані дані починаються з а конкретний символ і закінчується конкретним символом.	Значення налаштування в $D_2 + 1$ є певним початковим символом і а специфічний кінцевий характер.	Якщо початковий символ 16#3A, а символ зупинки — 16#0A, значення в $D_2 + 1$ — 16#3A0A.
6	Отримано певну кількість даних.	Значення налаштування в $D_2 + 1$ — це довжина даних, які потрібно отримати. Значення параметра становить від 1 до 256.	Якщо ви хочете отримати 10 символів, встановіть значення в $D_2 + 1$ на 10.
7	Отримані дані закінчуються на а специфічний характер і створює переривання зв'язку.	Значення параметра в $D_2 + 1$ є конкретним кінцевим символом.	Якщо кінцевий символ — 16#0A, значення в $D_2 + 1$ — 16#000A.
8	Встановіть кількість даних отримані, а потім генерувати переривання зв'язку.	Значення налаштування в $D_2 + 1$ є довжиною отриманих даних. значення параметра становить від 1 до 256.	Якщо ви хочете отримати 10 символів, встановіть значення в $D_2 + 1$ до 10.
9	Отримані дані закінчуються певним символом або певною кількістю отриманих даних; коли виконується будь-яка умова, передача завершена.	Специфічний кінцевий символ є зберігається у старшому байті в $D_2 + 1$, а час зберігається в молодшому байті в $D_2 + 1$. Час, встановлений у молодшому байті в $D_2 + 1$, має становити від 1 до 255 мілісекунд.	Якщо кінцевий символ — 16#0A, а час — 15 мілісекунд, довжина даних — 15 слів, значення в $D_2 + 1$ — 16#0A0F.
інші	Якщо використовуваний режим не підтримується, інструкція не виконується.		

7. За винятком режимів 6 і 8, коли дані, отримані в D_2 , перевищують максимальний діапазон довжини отриманих даних (256 слова) і не отримано кінцевий символ, інструкція припиняє виконання та розглядає цю операцію як помилку отримання. $D_1 + 0$ дорівнює 0, а $D_1 + 1$ – $D_1 + n$ не зберігають отримані дані.

8. Взаємодія між комунікаційним портом, відповідними спеціальними допоміжними реле та відповідними спеціальними даними

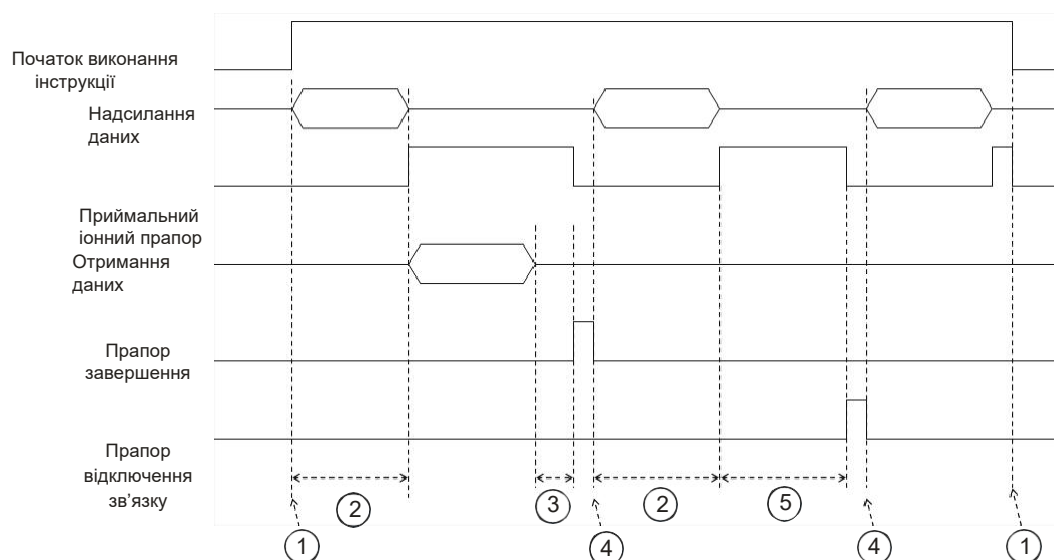
реєстру описано в розділі 6.19.3.

9. Часові діаграми

● Режим отримання даних: 0

Коли дані надсилаються, ви не можете скасувати надсилання даних. Якщо умовний контакт перед інструкцією не ввімкнуто, дані все одно буде надіслано, але прапор завершення не буде встановлено на ON після завершення надсилання даних.

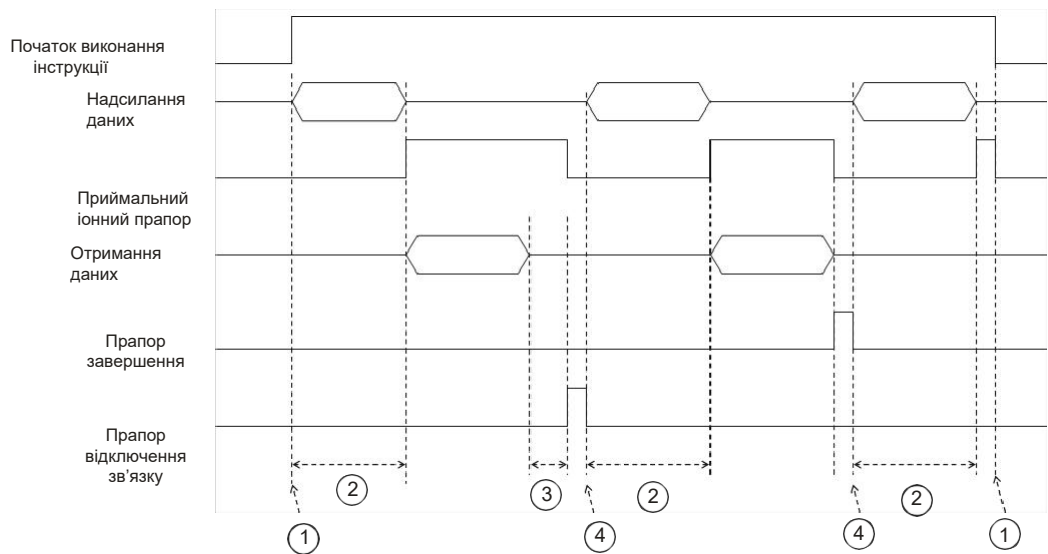
● Режим отримання даних: 1 або 4



опис:

- ① → Почати/зупинити виконання інструкції.
- ② → Час, протягом якого дані надсилаються. Період часу, протягом якого надсилаються дані, не вимірюється.
- ③ → Після отримання першого символу вимірюється час, який мине до отримання наступного символу. Щоразу, коли символ отримано, інструкція очищає виміряний час. Прапор завершення не встановлюється на ON, доки виміряний час не перевищить значення налаштування в $D_2 + 1$.
- ④ → Якщо інструкція все ще ввімкнена після скидання прапора завершення або прапора зв'язку, наступні дані зв'язку надсилаються автоматично під час сканування інструкції в наступному циклі.
- ⑤ → Коли ПЛК починає отримувати дані, він починає вимірювати час, що минає. Він не встановлює для тайм-ауту зв'язку значення ON, доки виміряний час не перевищить період тайм-ауту. Рекомендується встановити період очікування довшим за час, встановлений у $D_2 + 1$.

● Режим отримання даних: 2, 3, 5, 6 або 9.



опис:

- ① → Почати/зупинити виконання інструкції.
- ② → Час, протягом якого дані надсилаються. Період часу, протягом якого надсилаються дані, не вимірюється.
- ③ → Після отримання першого символу час, який проходить до наступного символу отримане вимірюється. Щоразу, коли символ отримано, інструкція очищає виміряний час. Прапор тайм-ауту зв'язку не встановлюється на ON, доки виміряний час не перевищить період тайм-ауту.
- ④ → Якщо інструкція все ще ввімкнена після скидання прапора завершення або прапора зв'язку, наступні дані зв'язку надсилаються автоматично, коли інструкція сканується в наступному циклі.

10. Режим передачі даних / Режим отримання даних

8-бітний режим: команда, яка редагується, зберігається в пристрої початкової передачі, а команда, яка надсилається, включає головний код і кінцевий код. Ця інструкція розділяє 16-бітні дані на старші 8-бітні дані та молодші 8-бітні дані. Інструкція ігнорує старші 8-бітні дані, і може надсилати або отримувати молодші 8-бітові дані, які можна надсилати або отримувати. Візьмемо, наприклад, стандартний Modbus. Надсилання даних: (PLC→зовнішнє обладнання)

D10 Low	D11Low	D12Low	D13Low	D14Low	D15 Low	D16Low
Head code		Initial transmission device: The low 8-bit data in D10			Tail code 1 Tail code 2	
Length=7						

Отримання даних: (Зовнішнє обладнання→PLC)

D100Low	D101Low	D102Low	D103Low	D104Low	D105Low	D106Low
Head code		Initial reception device: The low 8-bit device in D100			Tail code 1 Tail code 2	
Length=7						

16-бітний режим: команда, яка редагується, зберігається в пристрої початкової передачі, а команда, яка надсилається

включає код голови та код хвоста. Коли SM106/SM107 вимкнено, інструкція розділяє 16-бітні дані на старші 8-бітні дані та молодші 8-бітні дані.

Надсилання даних: (PLC→зовнішнє обладнання)

D10Low	D10High	D11Low	D11High	D12Low	D12 High	D13Low
Head code		Initial transmission device: The low 8-bit data in D10			Tail code 1 Tail code 2	
Length=7						

Отримання даних: (Зовнішнє обладнання→PLC)

D100Low	D100High	D101 Low	D101High	D102 Low	D102High	D103 Low
Head code		Initial reception device: The low 8-bit device in D100			Tail code 1 Tail code 2	
Length=7						

Дані, які ПЛК отримує від зовнішнього обладнання, включають головний і кінцевий коди; отже, ти потрібно знати про налаштування довжини.

11. Якщо режим 7 або 8, відповідний комунікаційний порт і номер переривання вказані в наступній таблиці.

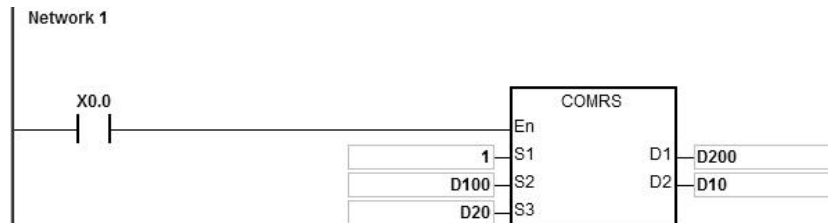
спілкування номер порту	COM1	COM2	Картка1	Картка2
Номер переривання	I300	I302	I304	I306

12. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.

У наступних прикладах використовується COM1 (RS485).

Приклад 1

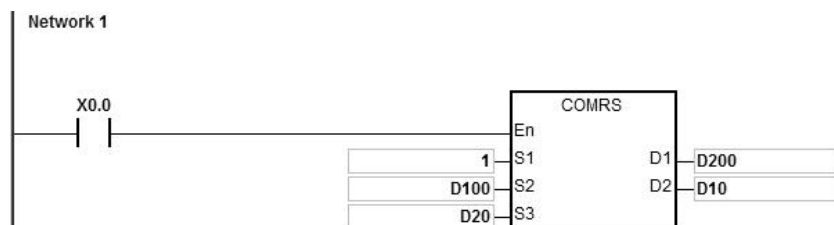
Режим у D_2 дорівнює 0 (не отримує даних зв'язку), і ви встановлюєте 8-бітний режим для надсилання та отримання даних (SM106=ON).



1. Довжина даних, які потрібно надіслати: D20=4.
2. Вміст даних, які потрібно надіслати: D100=16#0031, D101=16#0032, D102=16#0033, D103=16#0034.
3. Установіть D10=16#0000 (лише надсилання даних, а не отримання даних).
4. Увімкніть контакт X0.0.
5. ПЛК надсилає 4 частини даних.
6. Надсилання даних: PLC→зовнішнє обладнання 31 32 33 34.
7. Оскільки отримання даних не потрібне, після того, як ПЛК надішле дані, операція завершується, і SM100=0.
8. Щоб надіслати більше даних, установіть прапорець SM100 на ВИМК., щоб розпочати операцію знову.

Приклад 2

Режим у **D₂** дорівнює 1 (встановлення значення тайм-ауту 5–3000 мс), і ви встановлюєте 16-бітний режим для надсилання та отримання даних (SM106=ВИМК.).

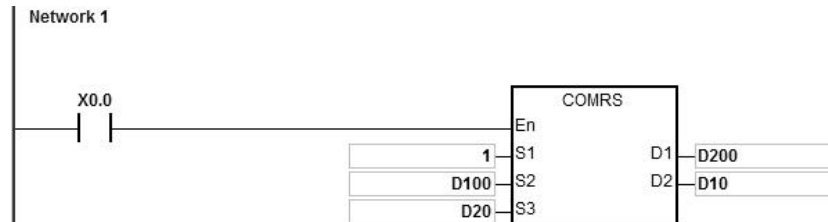


1. Довжина даних для надсилання: D20=4.
2. Вміст даних, які потрібно надіслати: D100=16#3231, D101=16#3433.
3. Встановіть D10=16#0001 (режим: 1), D11=300 (встановіть значення тайм-ауту 300 мс).
4. Увімкніть контакт X0.0.
5. PLC надсилає 4 частини даних.
6. Надсилання даних: PLC→зовнішнє обладнання 31 32 33 34.
7. Після того, як зовнішнє обладнання отримує дані від ПЛК, воно надсилає 5 послідовних даних на ПЛК, кожна з яких надсилається менш ніж за 20 мс. Зовнішнє обладнання→PLC 35 36 37 38 39.
8. D200=5 (кількість отриманих даних) і вміст отриманих даних: D201=16#3635, D202=16#3837, D203=16#0039.
9. SM100=ON: прийом даних завершено.
10. Щоб надіслати більше даних, установіть прапорець SM100 на ВИМК., щоб розпочати операцію знову.

ПРИМІТКА. Після завершення надсилання даних прапор отримання SM98 увімкнено, а потім ПЛК починає отримувати дані. Ви встановлюєте час очікування між кожним прийомом даних у D11. Коли інтервал часу перевищує встановлений тайм-аут і дані не надходять, SM100 вмикається.

Приклад 3

Режим у D_2 — це 2 (отримані дані закінчуються певним символом), і ви встановлюєте режим надсилання та отримання даних на 8-бітний режим (SM106=ON).



1. Встановіть довжину даних для надсилання: D20=0, тобто ПЛК не надсилатиме дані, а лише отримуватиме дані.
2. Встановіть D10=16#0002 (режим: 2), D11=16#000A (кінцевий символ 16#0A).
3. Увімкніть контакт X0.0.
4. ПЛК очікує отримання даних від зовнішнього обладнання. D20=0 означає, що ПЛК не надсилає дані на зовнішнє обладнання.

5. Зовнішнє обладнання надсилає дані на ПЛК.

Зовнішнє обладнання → PLC 31 32 33 34 35 0A.

6. D200=6 (номер отриманих даних), вміст отриманих даних: D201=16#0031, D202=16#0032, D203=16#0033, D204=16#0034, D205=16#0035, D206=16#000A

7. SM100=ON: прийом даних завершено.

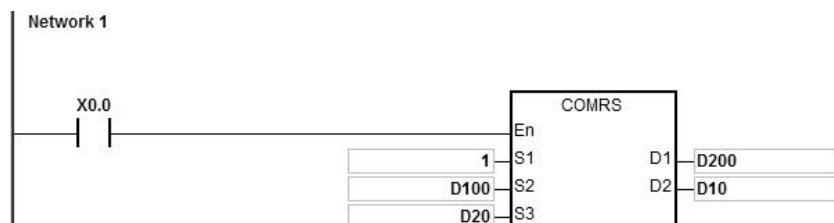
8. Щоб надіслати більше даних, встановіть прапорець SM100 на ВИМК., щоб розпочати операцію знову.

ПРИМІТКА. Після завершення надсилання даних прапор отримання SM98 увімкнено, а потім ПЛК починає отримувати дані до отримання кінцевого символу (16#0A). Після завершення прийому даних SM100 увімкнено. Якщо тайм-аут зв'язку виникає, але кінцевий символ (16#0A) все ще не отримано, прапор тайм-ауту зв'язку SM104 увімкнено.

Приклад 4

Режим у D_2 дорівнює 3 (отримані дані закінчуються двома певними символами), і ви встановлюєте 16-бітний режим для надсилання та отримання даних (SM106=ВИМК.).

У цьому прикладі використовується DVP-ES2 як зовнішнє обладнання та запис H1234 у D100 у DVP-ES2.



1. Довжина даних, які потрібно надіслати: D20=17.
2. Вміст даних, які потрібно надіслати: D100=16#303A, D101=16#3031, D102=16#3136, D103=16#3630, D104=16#3134, D105=16#3332, D106=16 #3334, D107=16#0D46, D108=16#000A.
3. Встановіть D10=16#0003 (режим: 3), D11=16#0D0A (кінцеві символи 16#0D і 16#0A).
4. Увімкніть контакт X0.0.
5. ПЛК надсилає 17 фрагментів даних.

Надсилання даних: PLC→зовнішнє обладнання 3A 30 31 30 36 31 30 36 34 31 32 33 34 33 46 0D 0A

(ASCII-код: 0106106412343FCRLF)

6. Зовнішнє обладнання отримує дані від ПЛК, а останні 2 дані – 16#0D і 16#0A.

Зовнішнє обладнання → PLC 3A 30 31 30 36 31 30 36 34 31 32 33 34 33 46 0D 0A

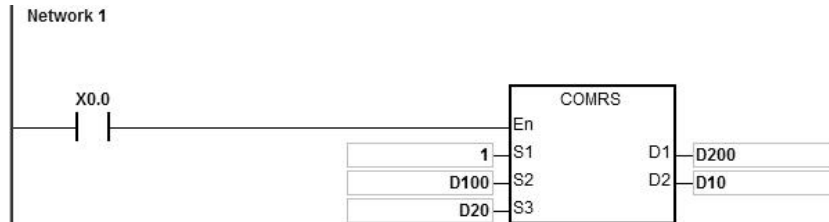
(код ASCII: 0106106412343FCRLF)

7. D200=17 (номер отриманих даних), а вміст отриманих даних: D201=16#303A, D202=16#3031, D203=16#3136, D204=16#3630, D205=16#3134, D206 =16#3332, D207=16#3334, D208=16#0D46, D209=16#000A.
8. SM100=ON: прийом даних завершено.
9. Щоб надіслати більше даних, установіть прапорець SM100 на ВИМК., щоб розпочати операцію знову.

ПРИМІТКА. Після завершення надсилання даних прапор отримання SM98 увімкнено, а потім ПЛК починає отримувати дані, доки не отримає кінцевий символ (16#0D0A). Після завершення прийому даних SM100 увімкнено. Якщо тайм-аут зв'язку виникає, але кінцевий символ (16#0D0A) все ще не отримано, прапор тайм-ауту зв'язку SM104 увімкнено.

Приклад 5

Режим у D_2 дорівнює 4 (отримані дані починаються з певного символу, і ви встановлюєте значення тайм-ауту 5–255 мс.), і ви встановлюєте режим надсилання та отримання даних у 8-бітний режим (SM106=ON).



1. Довжина даних, які потрібно надіслати: D20=4.
2. Вміст даних, які потрібно надіслати: D100=16#0031, D101=16#0032, D102=16#0033, D103=16#0034.
3. Встановіть D10=16#0004 (режим: 4), D11=16#3A0F (початковий символ — 16#3A, а значення часу — 16#0F, тобто 15 мс).
4. Увімкніть контакт X0.0.
5. ПЛК надсилає 4 частини даних.

Надсилання даних: PLC→зовнішнє обладнання 31 32 33 34

6. Зовнішнє обладнання отримує дані від ПЛК, а потім надсилає 7 послідовних слів до ПЛК із інтервал 1 мс між кожною відправкою.

Зовнішнє обладнання → PLC 30 3A 35 36 37 38 39

7. D200=6 (номер отриманих даних), а вміст отриманих даних: D201=16#003A, D202=16#0035, D203=16#0036, D204=16#0037, D205=16#0038, D206=16#0039.

8. SM100=ON: прийом даних завершено.

9. Щоб надіслати більше даних, установіть прапорець SM100 на ВИМК., щоб розпочати операцію знову.

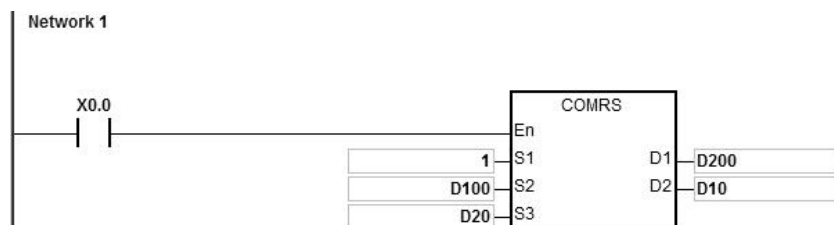
ПРИМІТКА: Коли надсилання даних завершено, прапор отримання SM98 увімкнено, і тоді ПЛК готовий приймати дані.

Коли ПЛК отримує початковий символ 16#3A, він починає отримувати дані. Інтервал очікування між отриманням кожної частини даних встановлюється в D11. Коли інтервал перевищує встановлений тайм-аут 16#0F (15 мс) і дані не надходять, SM100 увімкнено.

Приклад 6

Режим у D_2 дорівнює 5 (отримані дані починаються й закінчуються певним символом), і ви встановлюєте 16-бітний режим надсилання та отримання даних (SM106=ВИМК.).

У цьому прикладі DVP-ES2 використовується як зовнішнє обладнання та зчитує дані з D100 у DVP-ES2.



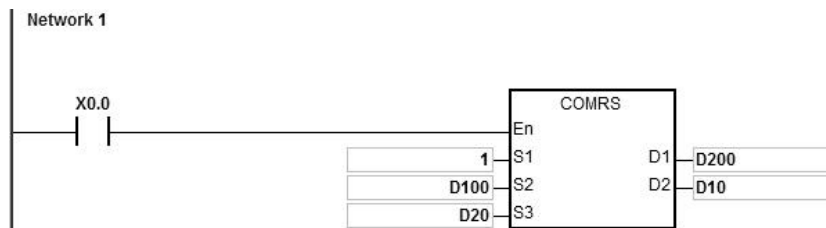
1. Довжина даних, які потрібно надіслати: D20=17.
2. Вміст даних, які потрібно надіслати: D100=16#303A, D101=16#3031, D102=16#3133, D103=16#3630, D104=16#3034, D105=16#3030, D106=16 #3831, D107=16#0D37, D108=16#000A
3. Встановіть D10=16#0005 (режим: 5), D11=16#3A0A (початковий символ — 16#3A, а кінцевий — 16#0A).
4. Увімкніть контакт X0.0.
5. ПЛК надсилає 17 фрагментів даних.
Надсилання даних: PLC→зовнішнє обладнання 3A 30 31 30 36 31 30 36 34 31 32 33 34 33 46 0D 0A
(ASCII-код: 0106106412343FCRLF)
6. Зовнішнє обладнання отримує дані від ПЛК, а останні 2 дані: 16#0D і 16#0A.
Зовнішнє обладнання → PLC 3A 30 31 30 36 31 30 36 34 31 32 33 34 33 46 0D 0A (код
ASCII: 0106106412343FCRLF)
7. D200=15 (номер отриманих даних), а вміст отриманих даних: D201=16#303A, D202=16#3031, D203=16#3033, D204=16#3132, D205=16#3332, D206 =16#4234, D207=16#0D34, D208=16#000A.
8. SM100=ON: прийом даних завершено.
9. Щоб надіслати більше даних, установіть прапорець SM100 на ВИМК., щоб розпочати операцію знову.

ПРИМІТКА: Коли надсилання даних завершено, прапор отримання SM98 увімкнено, і тоді ПЛК готовий до прийому даних.

Коли ПЛК отримує початковий символ 16#3A, він починає отримувати дані до отримання кінцевого символу 16#0A, а SM100 увімкнено. Якщо час очікування зв'язку виникає, але початковий символ 16#3A або кінцевий символ 16#0A все ще не отримано, прапор тайм-ауту зв'язку SM104 увімкнено.

Приклад 7

Режим у D_2 становить 6 (довжина отриманих даних), і ви встановлюєте режим надсилання та отримання даних на 8-бітний режим (SM106=ON).



1. Довжина даних, які потрібно надіслати: D20=4.
2. Вміст даних, які потрібно надіслати: D100=16#0031, D101=16#0032, D102=16#0033, D103=16#0034.
3. Установіть D10=16#0006 (режим: 6), D11=16#0008 (отримано 8 фрагментів даних).
4. Увімкніть контакт X0.0.
5. ПЛК надсилає 4 частини даних.

Надсилання даних: PLC → зовнішнє обладнання 31 32 33 34

6. Зовнішнє обладнання отримує дані від ПЛК, а потім надсилає 8 послідовних даних до ПЛК.

Зовнішнє обладнання → PLC 32 33 34 35 36 37 38 39

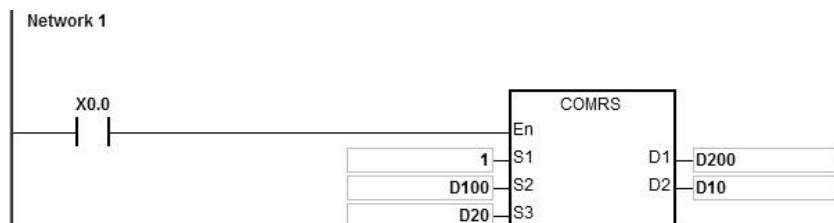
7. D200=8 (номер отриманих даних), а вміст отриманих даних: D201=16#0032, D202=16#0033, D203=16#0034, D204=16#0035, D205=16#0036, D206=16#0037, D207=16#0038, D208=16#0039.
8. SM100=ON: прийом даних завершено.
9. Щоб надіслати більше даних, установіть прапорець SM100 на ВИМК., щоб розпочати операцію знову.

ПРИМІТКА: Коли надсилання даних завершено, прапор отримання SM98 увімкнено, і тоді ПЛК готовий до прийому даних.

При отриманні встановленої кількості даних SM100 увімкнено. Якщо тайм-аут зв'язку відбувається, але встановлена кількість даних все ще не отримано, прапор тайм-ауту зв'язку SM104 увімкнено.

Приклад 8

Режим у D_2 становить 7 (отримані дані закінчуються певним символом і генерують переривання зв'язку), і ви встановлюєте режим надсилання та отримання даних на 8-бітний режим (SM106=ON).



Програми переривання зв'язку:



1. Очистити переривання: $D30=0$.
2. Довжина даних для надсилання: $D20=4$.
3. Вміст даних, які потрібно надіслати: $D100=16\#0031$, $D101=16\#0032$, $D102=16\#0033$, $D103=16\#0034$.
4. Встановіть $D10=16\#0007$ (режим: 7), $D11=16\#000A$ ($16\#0A$ — символ закінчення).
5. Увімкніть контакт $X0.0$.
6. ПЛК надсилає 4 частини даних.

Надсилання даних: PLC → зовнішнє обладнання 31 32 33 34

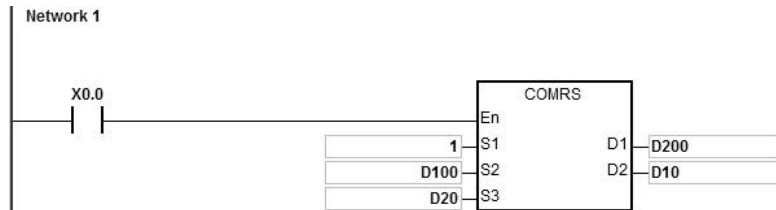
7. $D30=0$ (програми в перериванні не виконуються).
8. Зовнішнє обладнання надсилає дані на ПЛК. Зовнішнє обладнання → PLC 31 32 33 34 35 0A
9. $D200=6$ (номер отриманих даних), а вміст отриманих даних: $D201=16\#0031$, $D202=16\#0032$, $D203=16\#0033$, $D204=16\#0034$, $D205=16\#0035$, $D206=16\#000A$.
10. SM100=ON: прийом даних завершено.
11. $D30=1$ (спрацьовує переривання, а потім виконується INC $D30$).
12. Щоб надіслати більше даних, установіть прапор SM100 на ВІМК., щоб розпочати операцію знову.

ПРИМІТКА: Коли надсилання даних завершено, прапор отримання SM98 увімкнено, і тоді ПЛК готовий до прийому даних.

Під час отримання встановленого кінцевого символу ($16\#06$) SM100 увімкнено. Якщо тайм-аут зв'язку виникає, але встановлений кінцевий символ все ще не отримано, прапор тайм-ауту зв'язку SM104 увімкнено.

Приклад 9

Режим у D_2 становить 8 (встановлена кількість даних отримується та створює переривання зв'язку), і ви встановлюєте режим надсилання та отримання даних на 8-бітний режим (SM106=ON).



Програми переривання зв'язку:



1. Очистити переривання: D30=0
2. Довжина даних для надсилання: D20=4.
3. Вміст даних, які потрібно надіслати: D100=16#0031, D101=16#0032, D102=16#0033, D103=16#0034.
4. Встановіть D10=16#0008 (режим: 8), D11=16#0008 (8 фрагментів даних для отримання).
5. Увімкніть контакт X0.0.
6. ПЛК надсилає 4 частини даних.

Надсилання даних: PLC→зовнішнє обладнання 31 32 33 34

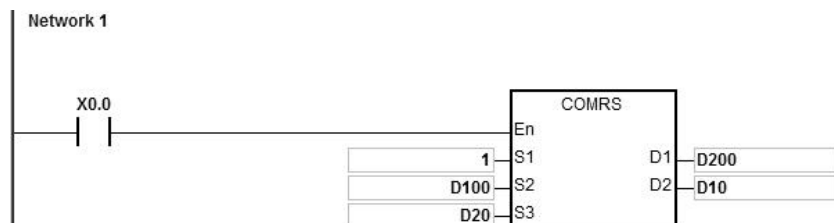
7. D30=0 (програми в перериванні не виконуються).
8. Зовнішнє обладнання отримує дані від ПЛК, а потім надсилає 8 послідовних даних до ПЛК.
9. Зовнішнє обладнання → PLC 32 33 34 35 36 37 38 39
10. D200=8 (номер отриманих даних), а вміст отриманих даних: D201=16#0032, D202=16#0033, D203=16#0034, D204=16#0035, D205=16#0036, D206=16#0037, D207=16#0038, D208=16#0039.
11. SM100=ON: прийом даних завершено.
12. D30=1 (спрацьовує переривання, а потім виконується INC D30).
13. Щоб надіслати більше даних, установіть прапорець SM100 на ВІМК., щоб розпочати операцію знову.

ПРИМІТКА: Коли надсилання даних завершено, прапор отримання SM98 увімкнено, і тоді ПЛК готовий до прийому даних.

При отриманні встановленої кількості даних SM100 увімкнено. Якщо тайм-аут зв'язку відбувається, але встановлена кількість даних все ще не отримано, прапор тайм-ауту зв'язку SM104 увімкнено.

Приклад 10

Режим у D_2 дорівнює 9 (встановлений кінцевий символ або отримано встановлену кількість даних) і встановіть режим надсилання даних/режим отримання даних на 8-бітний режим (SM106=ON).



1. Довжина даних, які потрібно надіслати: D20=4.
2. Вміст даних, які потрібно надіслати: D100=16#0031, D101=16#0032, D102=16#0033, D103=16#0034.
3. Встановіть D10=16#0009 (режим: 9), D11=16#0A0F (кінцевий символ — 16#0A, а встановлена довжина даних — 16#0F).
4. Увімкніть контакт X0.0.
5. ПЛК надсилає 4 частини даних.

Надсилання даних: PLC→зовнішнє обладнання 31 32 33 34

6. Зовнішнє обладнання отримує дані від ПЛК, а потім надсилає 15 фрагментів даних до ПЛК. Зовнішнє обладнання → PLC 31 32 33 34 35 0A 41 42 43 44 45 46 47 48 49

7. D200=6 (номер отриманих даних) і вміст отриманих даних: D201=16#0031, D202=16#0032, D203=16#0033, D204=16#0034, D205=16# 0035 , D206=16#000A.

ПЛК припиняє отримувати дані після отримання 6 -го фрагмента даних.

8. SM100=ON: прийом даних завершено.
9. Щоб надіслати більше даних, установіть прапорець SM100 на ВІМК., щоб розпочати операцію знову.

ПРИМІТКА: Коли надсилання даних завершено, прапор отримання SM98 увімкнено, і тоді ПЛК готовий до прийому даних.

При отриманні встановленого кінцевого символу або встановленої кількості даних SM100 увімкнено. Якщо тайм-аут зв'язку виникає, але встановлений кінцевий символ або встановлена кількість даних все ще не отримано, прапор тайм-ауту зв'язку SM104 увімкнено.

Додаткові зауваження

1. Немає обмежень на кількість разів, які ви можете виконати інструкцію зв'язку COMRS. Проте кожен порт зв'язку можна ввімкнути лише однією інструкцією зв'язку, а подальші інструкції зв'язку, які слідують, не виконуються.
2. Інструкція не використовує контрольну суму, коли ви виконуєте цю інструкцію. Якщо вам потрібна контрольна сума, використовуйте COMRS та іншу доступну інструкцію.
Якщо значення в D_2 дорівнює 2, 3, 5, 6 або 9, пропонується встановити період очікування. Після встановлення періоду очікування інструкція намагається знову надіслати дані, якщо символ зупинки не отримано.
4. Інструкція не очищає автоматично значення в $D_1 - D_1 + n$ щоразу, коли інструкція виконується або ПЛК починає отримувати нові дані зв'язку. Ви можете знати, чи отримує ПЛК даних і скільки даних, тільки після того, як прапор завершення перемикається з ВІМК. на УВІМК. Якщо ви хочете очистити значення в $D_1 - D_1 + n$, скористайтеся інструкцією ZRST (API 1206).
5. Якщо значення в S_1 виходить за межі діапазону, інструкція не виконується.
6. Якщо кількість пристроїв, починаючи з S_2 , не дорівнює значенню в S_3 , інструкція виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
7. Якщо значення в D_2 не між 0–9, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
8. Якщо значення в D_2 дорівнює 6, 8 або 9, а кількість пристроїв, починаючи з D_1 , не дорівнює значенню в $D_2 + 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 є 16#2003.
9. Якщо кількість отриманих даних перевищує кількість пристроїв, починаючи з D_1 , дані, які неможливо зберегти, ігноруються.
10. Якщо прапор завершення ввімкнено, ПЛК припиняє отримувати дані. Якщо комунікаційний порт отримує дані після завершення прапор ON, дані не отримані.
11. Якщо значення налаштування в S_3 не знаходиться в межах 0–256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#200B.
12. Коли режим D_2 дорівнює 6 або 8, довжина $D_2 + 1$ не між 1–256, інструкція не виконується. SM0 увімкнено, а код помилки в SR0 — 16#200B.

API	Код інструкції			Операнд						функція							
1813		COMDF	P	$S_1, S_2, S_3, S_4, S_5, D$						Налаштування формату зв'язку для серіалу комунікаційний порт							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1													☐	☐			
S_2													☐	☐			
S_3													☐	☐			
S_4													☐	☐			
S_5													☐	☐			
D													☐	☐			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													
S_2													
S_3													
S_4													
S_5													
D													

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

COMDF	COMDFP
En	En
S1	S1
S2	S2
S3	S3
S4	S4
S5	S5
D	D

- S_1 : Швидкість передачі (одиниця: 100 біт/с)
- S_2 : Кількість бітів даних
- S_3 : Біт парності
- S_4 : Кількість кінцевих бітів
- S_5 : Вибір формату Modbus
- D : Номер комунікаційного порту

Пояснення

- Ця інструкція надає спосіб безпосереднього встановлення значень параметрів замість оголошення змінних.
- S_1 встановлює швидкість передачі з одиницями 100 біт/с. Наприклад, значення 96 означає 9600 біт/с.
- S_2 встановлює кількість бітів даних. Значення 7 означає 7 бітів даних, а 8 означає 8 бітів даних. Якщо значення не дорівнює 7 або 8 в S_2 інструкція використовує значення за замовчуванням.

4. **S₃** встановлює біт парності. Значення 0 вказує на відсутність (без біта парності). Значення 1 вказує на перевірку непарних бітів. Значення 2 відповідає перевірці навіть бітів. Якщо значення в **S₃** не дорівнює 0, 1 або 2, інструкція використовує значення за замовчуванням.

5. **S₄** встановлює кількість кінцевих бітів. Значення 1 (попереднє налаштування) означає 1 біт. Значення 2 означає 2 біти. Якщо значення в **S₄** не дорівнює 1 або 2, інструкція використовує значення за замовчуванням.

6. **S₅** встановлює режим зв'язку для зв'язку Modbus. Значення 0 означає ASCII (значення за замовчуванням). Значення 1 означає RTU. Якщо значення в **S₅** не дорівнює 0 або 1, інструкція використовує значення за замовчуванням.

7. **D** встановлює номер порту зв'язку. Номер для COM1 – 1, COM2 – 2, Card1 – 11 і Card2 – 12. Якщо значення налаштування виходить за межі допустимого діапазону, інструкція не встановлює формат порту зв'язку.

8. Ви також можете безпосередньо встановити порт зв'язку в HWCONFIG в ISPSOFT (параметри COM-порту) або за допомогою спеціальних регістрів. Щоб дізнатися більше про HWCONFIG, перегляньте посібник користувача ISPSOFT. Зверніться до Розділу 6.19.3 щодо налаштування регістрів SR і SM, пов'язаних із зв'язком.

9. Зв'язок на фактичному порту зв'язку змінюється одразу після зміни налаштування інструкції. Якщо в даний момент здійснюється якийсь зв'язок, його примусово скасувати. Крім того, відповідно змінюється відповідне значення налаштування в SM/SR. Докладніше про SM/SR див. у розділі 6.19.3.

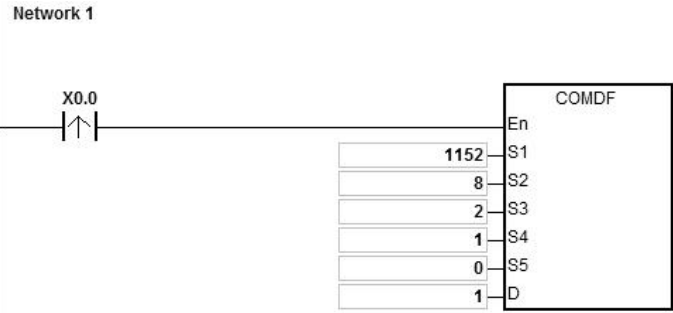
10. Ця інструкція не змінює жодних налаштувань для фактичного порту зв'язку, якщо налаштування формату зв'язку збігаються з попередніми налаштуваннями.

приклад

- У цьому прикладі використовується порт COM1 ПЛК. Інші комунікаційні порти ПЛК схожі за способом налаштування зв'язку.
- Контактом для умови запуску є X0.0.
- Встановіть (RS485) формат зв'язку ПЛК COM1 на 115200, 8, E та 1.
- Встановіть режим зв'язку (RS485) ПЛК COM1 на ASCII.
- У наведеній нижче таблиці пояснюються операнди COMDF для прикладу.

Операнд	опис		Цінність змісту
S₁	Швидкість передачі даних	115200 біт/с	1152
S₂	Кількість бітів даних	8	8
S₃	Біт парності	E	2
S₄	Кількість кінцевих бітів	1	1

S ₅	Вибір формату Modbus	ASCII	0
D	Номер комунікаційного порту	ПЛК COM1	1



API	Код інструкції			Операнд						функція							
1814		VFDRW		S_1, S_2, S_3, S						Ексклюзивна інструкція по послідовному зв'язку для Привід змінного струму Delta							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●	●		O	O	O	O		
S_2								●	●		O	O	O	O		
S_3								●	●		O	O	O	O		
S								●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

	VFDRW
En	
S1	
S2	
S3	
S	

- S_1 : Номер комунікаційного порту
- S_2 : Адреса станції VFD
- S_3 : Код функції
- S : Джерело та отримані дані

Пояснення

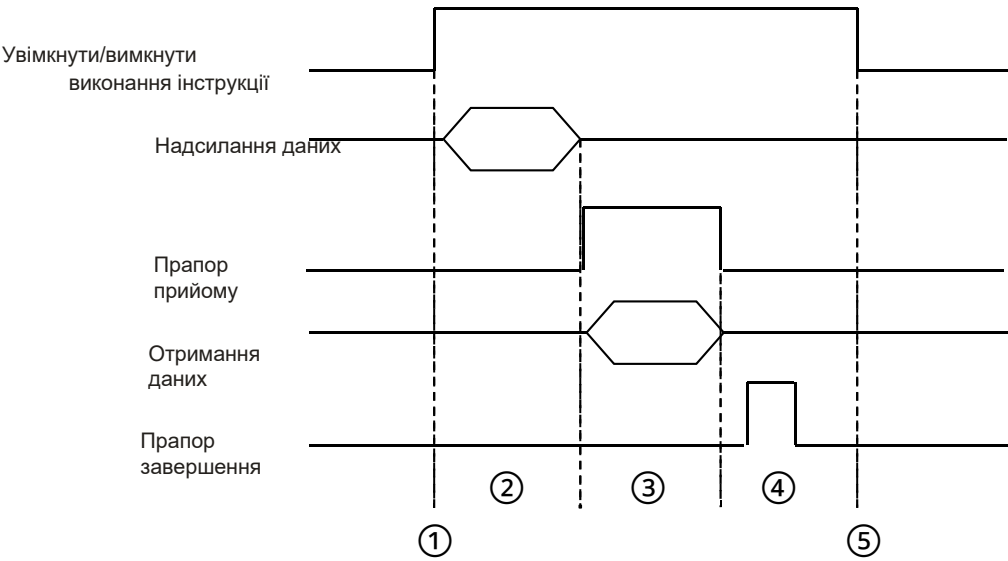
1. S_1 встановлює номер порту зв'язку. Номер для COM1 дорівнює 1, COM2 дорівнює 2, Card1 дорівнює 11 і Card2 дорівнює 12. Якщо значення перевищує допустимий діапазон, інструкція не отримує жодних даних зв'язку.
2. S_2 встановлює адресу станції для приводу двигуна змінного струму VFD. 0 означає, що інструкція використовує широкомовний режим. Діапазон становить від 0 до 254, і інструкція не виконується, якщо це значення виходить за межі допустимого діапазону.

3. **S₃** — код функції зв'язку, а **S** — джерело або отримані дані, як пояснюється в наступній таблиці.

S₃ код функції	Назва функції S₃	S джерело та отримані дані	Зауваження
0	Скидання через несправність	Невикористаний	Будь-яке значення може зберігатися в S .
1	Команда руху за годинниковою стрілкою	Значення швидкості	Зверніться до посібника користувача приводу змінного струму, щоб дізнатися значення налаштування та одиницю.
2	Біг проти годинникової стрілки команда	Значення швидкості	
3	СТІЙ	Невикористаний	Будь-яке значення може зберігатися в S .
4	Біг підтюпцем за годинниковою стрілкою команда	Невикористаний	Зверніться до посібника користувача приводу змінного струму, щоб налаштувати швидкість поштовху.
5	Біг підтюпцем проти годинникової стрілки команда	Невикористаний	
6	Читання держ	Отримав державні цінності	Зверніться до користувача приводу змінного струму керівництво щодо значення значень стану 5-бітових адрес H2100–H2104 для VFD.

6

4. Наступна діаграма показує час надсилання та отримання даних.



опис:

- ① → Почати або зупинити виконання інструкції.
- ② → Починається передача даних. Протягом цього часу час очікування зв'язку не вимірюється.
- ③ → Прапор прийому встановлено. Від моменту отримання першого символу до моменту отримання наступного символу вимірюється період часу. Щоразу, коли отримано символ, вимірний час обнуляється до 0. Інструкція генерує прапор тайм-ауту зв'язку, якщо вимірний час перевищує значення параметра тайм-ауту зв'язку.
- ④ → Коли отримання даних завершено, інструкція встановлює позначку завершення. Ви повинні зняти прапорець перед отриманням додаткових даних.
- ⑤ → Інструкція зупиняється на один цикл після встановлення прапора завершення. Тоді навчання можна розпочати в наступному циклі.

5. Немає обмежень щодо кількості виконання інструкції. Інструкція може використовувати лише один комунікаційний порт для виведення та виконання однієї інструкції зв'язку щоразу. Якщо отримання та надсилання даних завершено, ви повинні вимкнути інструкцію, щоб правильно звільнити керування зв'язком.

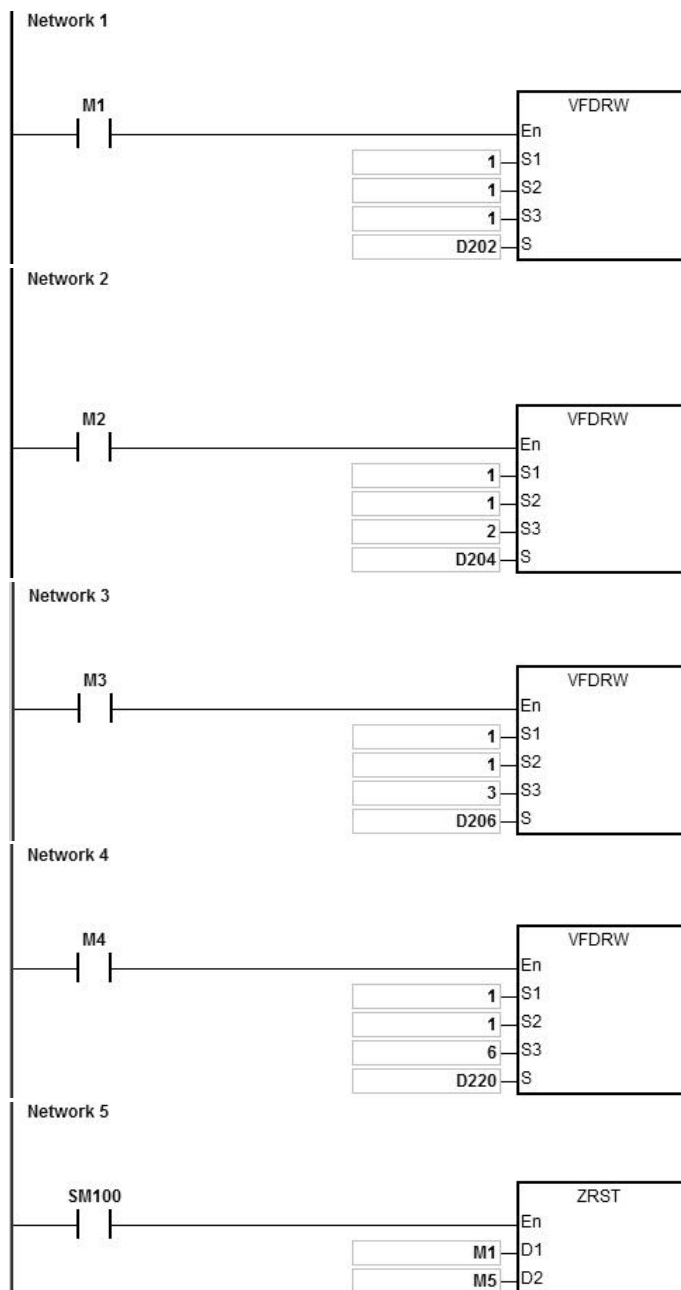
6. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.

Приклад налаштування протоколу зв'язку

1. Встановіть для порту ПЛК COM1 (RS485) адресу станції 2 і формат зв'язку в HWCONFIG за допомогою цих значення: ASCII, 115200, 7, N, 2.
2. Встановіть параметри приводу двигуна за допомогою панелі на приводі двигуна змінного струму Delta C2000 відповідно до наступних кроків.
 - A. Установіть 09-00 на 1: адресу станції приводу змінного струму встановлено на 1.
 - B. Установіть 09-01 на 115.2: Швидкість передачі RS485 приводу двигуна змінного струму становить 115200.
 - C. Установіть 09-04 на 1: Формат зв'язку RS485 приводу двигуна змінного струму – 7, N, 2.
 - D. Встановіть 09-20 на 1: вказівка частоти вводиться через RS485.
 - E. Встановіть 09-21 на 2: інструкція для виконання вводиться через RS485.

приклад

Використовуйте інструкцію VFDRW для керування швидкістю: змусьте VFD працювати вперед із частотою 120 Гц, потім рухатися назад із частотою 180 Гц, а потім зупинити роботу.



1. Підключіть AS COU до VFD.

Спочатку встановіть D202=12000. Коли M1 увімкнено, VFD починає прискорюватися після отримання команди руху за годинниковою стрілкою та працює за годинниковою стрілкою з частотою 120 Гц.

2. Спочатку встановіть D204=18000. Коли M2 увімкнено, VFD починає сповільнюватися, доки не зупиниться після отримання команди на рух проти годинникової стрілки, і працює проти годинникової стрілки з частотою 180 Гц.

3. Коли M3 увімкнено (у цей час значення в D206 ігнорується), VFD сповільнюється для зупинки після отримання команди зупинки.

4. Коли M4 увімкнено, інструкція зчитує значення H2100–H2104 VFD і зберігає їх у D220–224.

пристрій	D220	D221	D222	D223	D224
Зміст	Код помилки	стан VFD	Команда частоти	Вихідна частота	Вихідний струм

Стан VFD:

Біт2=1 змушує VFD виконувати команду Jog. Біт4–3=11В змушує VFD працювати проти годинникової стрілки.

Команда частоти становить 18000, і вона змушує VFD працювати на частоті 180 Гц. Щоб дізнатися про визначення адрес параметрів у протоколі зв'язку, зверніться до посібника користувача Delta AC Motor Drive.

5. Прапор завершення прийому SM100 увімкнено, а значення M1–M5 скидаються, щоб уникнути перешкод для наступної команди зв'язку.

Після отримання даних, які повертає VFD, інструкція перевіряє формат даних, надісланих назад від VFD.

Якщо формат даних правильний, SM100 увімкнено; інакше SM102 увімкнено.

API		Код інструкції			Операнд					функція							
1815			ASDRW		S_1, S_2, S_3, S					Ексклюзивна інструкція по послідовному зв'язку для Дельта сервопривід							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1								●	●		○	○	○	○			
S_2								●	●		○	○	○	○			
S_3								●	●		○	○	○	○			
S								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

ASDRW
En
S1
S2
S3
S

- S_1 : Номер комунікаційного порту
- S_2 : Адреса станції сервоприводу
- S_3 : Код функції
- S : Джерело та отримані дані

Пояснення

1. S_1 встановлює номер порту зв'язку. Номер для COM1 дорівнює 1, COM2 дорівнює 2, Card1 дорівнює 11 і Card2 дорівнює 12. Якщо значення перевищує допустимий діапазон, інструкція не отримує жодних даних зв'язку.
2. S_2 встановлює адресу станції для сервоприводу. 0 означає, що інструкція використовує широкомовний режим. Діапазон значень – від 0 до 254. Інструкція не виконується, якщо значення виходить за допустимий діапазон.
3. Докладніше про параметри сервоприводу див. у посібнику з експлуатації Delta Servo.
4. S_3 — код функції зв'язку, а S — вихідні або отримані дані, як пояснюється в наступній таблиці.

S₃ і S для серій A, AB, A+, B			
S₃ код функції	Назва функції S₃	S джерело або отримані дані	Зауваження
0	Зчитування значення стану сервоприводу	Займає 5 пристроїв посліпль S – S +4	Читання значення стану з P0-04–P0-08
1	Читання регістра сервоприводу значення	Займає 8 пристроїв посліпль S – S +7	Читання значення з регістри P0-09–P0-16
2	Запис значення регістра сервоприводу	Займає 8 пристроїв посліпль S – S +7	Запис даних у реєстри P0-09–P0-16
3	Введення швидкості бігового режиму, біг за годинниковою стрілкою, біг проти годинникової стрілки та зупинка	Діапазон швидкості: 1–3000; 4999 (за годинниковою стрілкою) 4998 (біг проти годинникової стрілки) 5000 (стоп)	Запис даних у регістри P4-05
4	Увімкнення/вимкнення сервоприводу	1: Серво увімкнено Будь-яке інше значення: Servo OFF	Запис даних у регістри P2-30
5	Команда швидкості	Дійсний діапазон: від -5000 до 5000	Запис даних у регістри P1-09–P1-11
6	Команда крутного моменту	Дійсний діапазон: від -300 до 300	Запис даних у регістри P1-12–P1-14

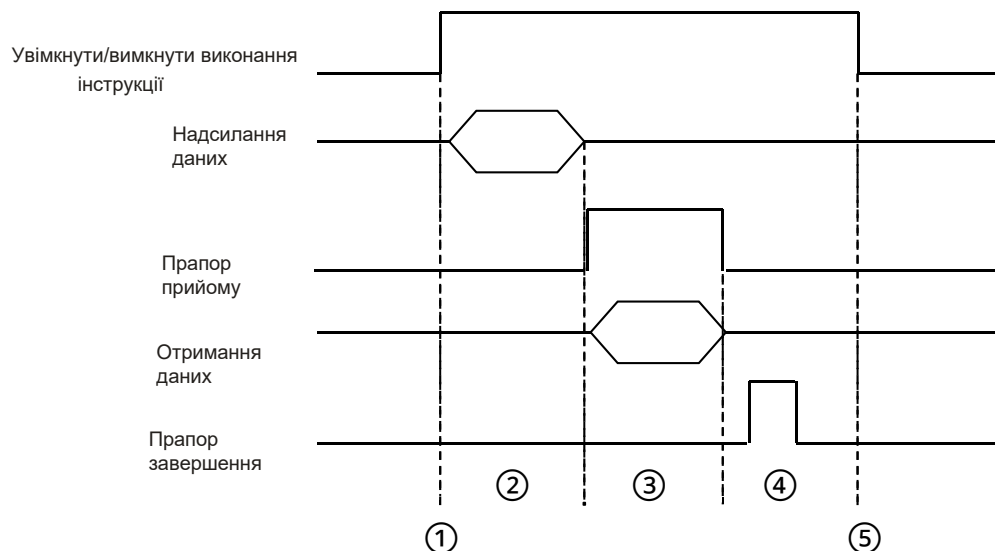
6

S₃ і S для серій A2, B2			
S₃ код функції	Назва функції S₃	S джерело або отримані дані	Зауваження
16	Зчитування значення стану сервоприводу	Займає 10 посліпль прилади S – S +9	Читання значення стану з регістри P0-09–P0-13 (32-розрядне значення)
17	Запис даних у сервопривід регістр	Займає 8 пристроїв посліпль S – S +7	Запис даних у регістри P0-17–P0-20 (32-розрядне значення)
18	Запис параметра відображення значення *1	Займає 8 пристроїв посліпль S – S +7	Запис даних у регістри P0-25–P0-28 (32-розрядне значення)
19	Введення швидкості підтюпця, біг за годинниковою стрілкою, бігти	Дійсний діапазон швидкості: дійсний діапазон 1–5000;	Запис даних в реєстри P4-05

	проти годинникової стрілки, СТІЙ	4999 (за годинниковою стрілкою) 4998 (біг проти годинникової стрілки) 0 (стоп)	
20	Увімкнення/вимкнення сервоприводу	1: Серво увімкнено Будь-яке інше значення: Servo OFF	Запис даних у регістри P2-30
21	Команда швидкості (3 набори)	Займає 6 пристроїв посліп із допустимим діапазоном значень налаштування: від - 60000 до 60000	Запис даних в реєстри P1-09–P1-11 (32-розрядне значення)
22	Команда крутного моменту (3 набори)	Займає 6 пристроїв посліп із допустимим діапазоном значень налаштування: від - 300 до 300	Запис даних у регістри P1-12–P1-14 (32-розрядне значення)
23	Встановлення цілей відображення параметри сервоприводу	Займає 8 пристроїв посліп S – S + 7	Запис даних у регістри P0-35–P0-38 (32-розрядне значення)

*1: Серія B2 не підтримує код функції 18 і 23.

5. Наступна діаграма показує час надсилання та отримання даних.



опис:

- ① → Почати або зупинити виконання інструкції.
- ② → Розпочинається передача даних. Протягом цього періоду час очікування зв'язку не вимірюється.
- ③ → Прапор прийому встановлено. Від моменту отримання першого символу до моменту отримання наступного символу вимірюється період часу. Кожного разу, коли отримано символ, виміряне значення скидається до 0. Інструкція генерує прапор очікування зв'язку, якщо виміряний час перевищує

значення параметра часу очікування зв'язку.

④ → Коли отримання даних завершено, інструкція встановлює позначку завершення. Ви повинні зняти прапорець перед отриманням додаткових даних.

⑤ → Інструкція вимикається на один цикл після встановлення прапора завершення. Тоді навчання можна розпочати в наступному циклі.

6. Немає обмежень щодо кількості виконання інструкції. Інструкція може використовувати лише один комунікаційний порт для виведення та виконання однієї інструкції зв'язку щоразу. Якщо отримання та надсилання даних завершено, ви повинні вимкнути інструкцію, щоб правильно звільнити керування зв'язком.

7. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.

Приклад налаштування протоколу зв'язку:

1. Встановіть порт ПЛК COM1 (RS485) у HWCONFIG із такими значеннями: ASCII, 115200, 8, E, 1

2. Встановіть параметри сервоприводу за допомогою панелі на сервоприводі Delta ASDA-A2 відповідно до наступних кроків.

A. Встановіть P2-08 на 10, щоб відновити заводські налаштування.

B. Вимкніть і знову ввімкніть сервопривід.

C. Встановіть P1-01 на 0001 (режим PR).

D. Встановіть P3-00 на 2: адресу станції сервоприводу встановлено на 2.

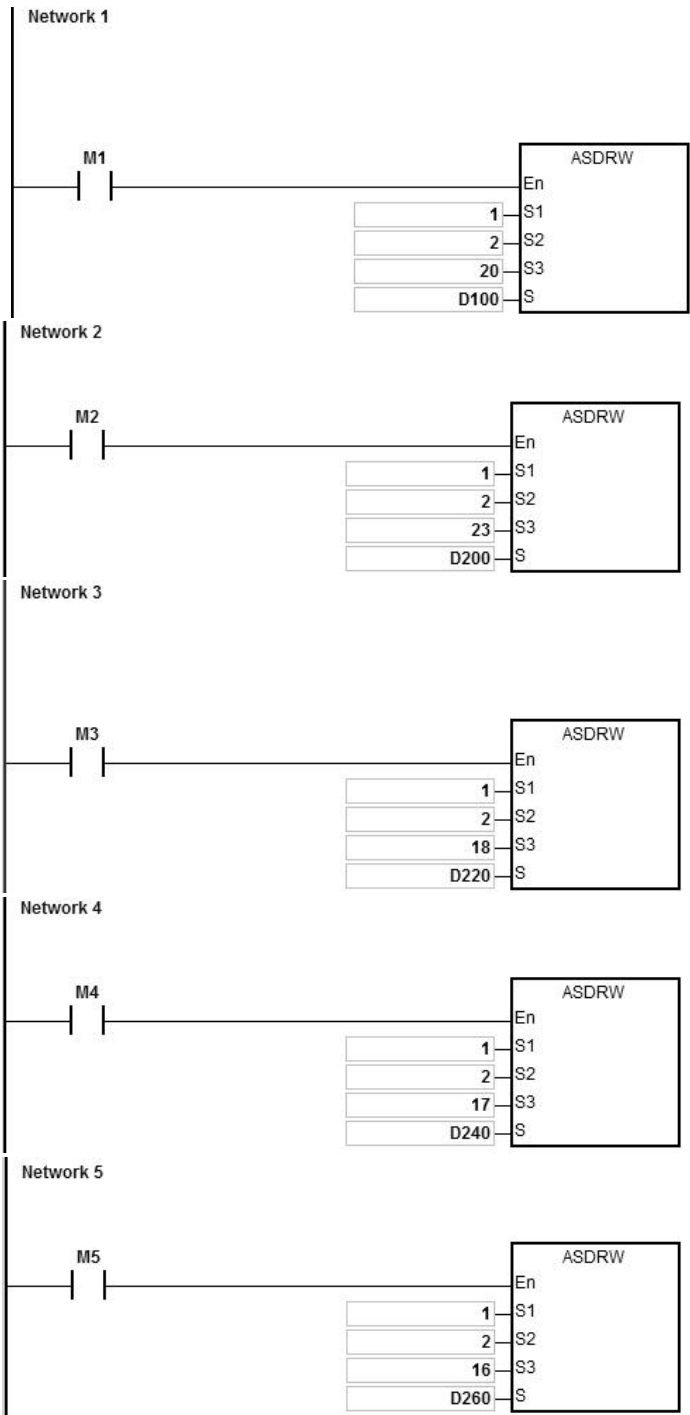
E. Встановіть P3-01 на 0205: швидкість передачі RS485 сервоприводу встановлена на 115200.

F. Встановіть P3-02 на 0004: формат зв'язку RS485 сервоприводу встановлено на 8, E, 1.

G. Після завершення цих налаштувань знову ввімкніть живлення сервоприводу.

приклад

Використовуйте інструкцію ASDRW для керування швидкістю: змусьте сервопривід працювати до відносного положення 5000000 PUU, прискорюючи на 400 мс до швидкості 3000,0 об/хв, а потім уповільнюючи на 200 мс.





1. Підключіть AS CPU до ASDA-A2.

Спочатку встановіть D100=1. Коли M1 увімкнено, ASDA-A2 увімкнено сервопривод.

2. Установіть значення в D200–D207, як у наведеній нижче таблиці, які записані в P0-35–P0-38 в ASDA-A2. Значення займає 8 послідовних пристроїв.

пристрій	D200	D201	D202	D203	D204	D205	D206	D207
Значення налаштування (16#)	05140515		06020602		06030603		053C0507	

Коли M2 увімкнено, значення в D200–D207 записуються в P0-35–P0-38 в ASDA-A2.

Значення параметрів P0-35–P0-38 встановлюють ціль відображення для P0-25–P0-28. Ви можете встановити ціль відображення; зверніться до посібника з експлуатації сервоприводу Delta.

ASDA-A2	P0-35	P0-36	P0-37	P0-38
Значення налаштування (16#)	0514 0515	0602 0602	0603 0603	053C 0507

Встановлення значення P0-38 на 053C 0507 вказує на те, що цільовим параметром відображення для P0-28 є P5-60 (16 біт) і P5-07 (16 біт).

6

ASDA-A2	P0-25	P0-26	P0-27	P0-28
Цільовий параметр відображення	P5-20, P5-21	P6-02	P6-03	P5-60, P5-07
Назва параметра	Час розгону/гальмування1 Час розгону/гальмування2	Тип шляху 1	Дані Path1	Тригер швидкості цілі та команди PR

3. У наступній таблиці наведено значення налаштувань для D220–D227.

пристрій	D220	D221	D222	D223	D224	D225	D226	D227
Значення налаштування (16#)	0190 00C8		0000 1083		4C4B40		7530 0001	

Коли M3 увімкнено, значення в D220–D227 записуються в P0-25–28 в ASDA-A2.

ASDA-A2	P0-25	P0-26	P0-27	P0-28
Значення налаштування (16#)	0190 00C8	0000 1083	4C4B40	7530 0001

Сервопривід починає працювати з параметрами: час прискорення=0190 (400 мс), час уповільнення=00C8 (200 мс), тип шляху=1083, команда позиції=4C4B40 (5000000 PUU), цільова швидкість=7530 (3000,0 об/хв) і PR тригер команди =1.

4. Установіть значення в D240-D247, як показано в наведеній нижче таблиці, для запису в P0-17–P0-20 в ASDA-A2.

Значення займають 8 послідовних пристроїв.

пристрій	D240	D241	D242	D243	D244	D245	D246	D247
Значення налаштування (10#)	41	0	0	0	0	0	0	0

Коли M4 увімкнено, значення в D240–D247 записуються в P0-17–20 в ASDA-A2.

Значення параметрів P0-17–20 задають вміст P0-09–12.

Ви можете встановити вміст, який буде відображатися; зверніться до посібника з експлуатації Delta Servo.

ASDA-A2	P0-17	P0-18	P0-19	P0-20
Значення налаштування (10#)	41	0	0	0

Установка P0-17 на 41 вказує на те, що вміст P0-09 є станом диска.

ASDA-A2	P0-09	P0-10	P0-11	P0-12
Зміст	Стан диска	Кількість імпульсів зворотного зв'язку двигуна	Кількість імпульсів зворотного зв'язку двигуна	Кількість імпульсів зворотного зв'язку двигуна

5. Коли M5 встановлено на ON, значення P0-17-P0-20 ASDA-A2 зчитуються на D260-D267.

пристрій	D260	D262	D264	D266
Зміст	Стан диска	Кількість імпульсів зворотного зв'язку двигуна	Кількість імпульсів зворотного зв'язку двигуна	Кількість імпульсів зворотного зв'язку двигуна

Коли біт стану накопичувача (біт 4) дорівнює одиниці, це вказує на те, що цільове положення досягнуто.

Зверніться до посібника з експлуатації сервоприводу Delta для пояснення P0-46.

6. Коли прапор завершення прийому SM100 встановлено на ON, інструкція очищає значення M1–M5, щоб уникнути перешкод для наступної команди зв'язку.

Після отримання даних, які ASDA-A2 надсилає назад, ПЛК перевіряє формат даних, надісланих назад з ASDA-A2. Якщо формат даних правильний, SM100 увімкнено; інакше SM102 увімкнено, якщо формат даних неправильний.

API	Код інструкції				Операнд				функція							
1816		CCONF		P	$S_1 - S_{11}$				Налаштування параметрів у таблиці обміну даними для комунікаційного порту							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●	●				○	○		
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●	●				○	○		
S_6								●	●				○	○		
S_7			●					●								
S_8								●	●				○	○		
S_9								●	●				○	○		
S_{10}								●	●				○	○		
S_{11}			●					●								
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING			
$S_1 - S_{11}$	Зверніться до пояснення до інструкції.															
Інструкція пульсу								16-бітна інструкція				32-розрядна інструкція				
AS								AS				-				

СИМВОЛ

CCONF		CCONFP	
En		En	
S1		S1	
S2		S2	
S3		S3	
S4		S4	
S5		S5	
S6		S6	
S7		S7	
S8		S8	
S9		S9	
S10		S10	
S11		S11	

$S_1 - S_{11}$: Зверніться до пояснення до інструкції.

Пояснення

1. У наступній таблиці наведено назви та описи S_1 – S_{11} .

пристрій	Ім'я	опис	Тип даних	Зауваження
S_1	Номер комунікаційного порту	1=COM1, 2=COM2. 11=Функція картки 1, 12 = функціональна карта 2. ПЛК із програмним забезпеченням версії 1.06.00 або новішої підтримує функціональні карти 1 і 2. Будь-яке інше значення, інструкція не змінює операнд.	WORD	
S_2	Номер позиції в таблиці обміну даними	Дійсний діапазон: 1–32. Якщо значення виходить за межі діапазону, інструкція не змінює операнд.	WORD	
S_3	Адреса станції віддаленого обладнання	Допустимий діапазон: 0 і 1–240. Якщо значення більше 240, інструкція використовує 240 як значення. Якщо значення менше або дорівнює 0, вихідна адреса станції не змінюється.	WORD	А від'ємне число обробляється як 0.
S_4	Код функції для читання даних	Підтримує 16#01, 02, 03, 04 і 17. Якщо значення перевищує допустимий діапазон, інструкція не змінює операнд. Якщо S_4 =16#17, він обробляється як 16#17.	WORD	
S_5	Читання пульта адреса зв'язку	16#0000–16#FFFF	WORD	
S_6	Довжина даних для читання	Код функції біта, 16#01, 02: підтримує 1–256 біт; Код функції Word, 16#03, 04, 17: Підтримує 1–100 слів. Якщо значення дорівнює 0 або менше за 0, це означає, що дані не читаються. Якщо значення більше максимального, інструкція використовує максимальне значення.	WORD	
S_7	Локальний реєстр для зберігання отриманих даних	Бітовий код функції: може бути лише пристрій M вибрано. Код функції Word: можна вибрати лише пристрій D.	BOOL WORD	

пристрій	Ім'я	опис	Тип даних	Зауваження
S₈	Код функції для запису даних	Підтримує 16#05, 06, 0F, 10. Якщо значення виходить за межі діапазону, інструкція не змінює операнд.	WORD	
S₉	Написання пульта адреса зв'язку	16#0000–16#FFFF	WORD	
S₁₀	Довжина запису даних	Код функції біта, 16#0F: підтримує 1–256 біти; Код функції Word, 16#10: підтримує 1–100 слів. Для коду функції 16#06 значення може бути лише 1. Якщо значення дорівнює 0 або менше за 0, це означає, що дані не читаються. Якщо значення більше максимального, інструкція використовує максимальне значення.	WORD	
S₁₁	Місцевий реєстр в які записуються віддалені дані	Код функції біта: можна вибрати лише M. Код функції Word: можна вибрати лише D.	BOOL WORD	

2. Рекомендується використовувати імпульсну інструкцію.

3. Подробиці про функціональні коди Modbus дивіться в S₄ і S₈ нижче.

6

Команда для AS, яка зчитує дані з кількох бітових пристроїв (які не є дискретними пристроями введення), дорівнює 1 (16#01).

Команда для AS зчитує дані з кількох бітових пристроїв (які є лише дискретними пристроями введення) — 2 (16#02).

Команда для AS для читання даних з кількох пристроїв вводу (які не є вхідними регістрами) — 3 (16#03).

Команда для AS зчитувати дані з кількох пристроїв вводу (які є лише вхідними регістрами) — 4 (16#04).

Команда AS для запису стану в бітовий пристрій — 5 (16#05).

Команда для AS для запису стану в текстовий пристрій є 6 (16#06).

Команда для AS для запису стану в кілька бітових пристроїв — 15 (16#0F).

Команда для AS для запису даних у кілька пристроїв введення даних є 16 (16#10).

Команда для AS для синхронного читання та запису даних у декілька пристроїв введення даних є 23 (16#17).

Підтримуються лише перелічені вище коди функцій. Інші значення налаштувань коду функції, наприклад 0, недійсні

(включаючи адресу зв'язку, довжину та початковий регістр), а інструкція виконує функцію обміну даними на основі початкових налаштувань параметрів зв'язку.

4. Коли ви вибираєте 16#17 (для синхронного читання та запису) у S_4 (код функції для читання), операнд S_8 (код функції для запису) обробляється як недійсний, а 16#17 автоматично обробляється для запису даних.

5. Для ПЛК із FW V1.04.00 або попередніми версіями: якщо вибрати 16#05 або 16#06 у S_8 (код функції для запису), операнд S_{10} (довжина даних) є недійсним, а 16#05 або 16#06 вказує на запис частини даних.

6. Значення параметрів, вказані в інструкції, дійсні лише під час роботи ПЛК. Якщо ви повторно ввімкнете живлення ПЛК, дані в таблиці обміну даними, налаштовані в HWCONFIG, приймаються як значення за замовчуванням. Якщо ви хочете змінити значення параметрів, скористайтеся цією інструкцією.

7. Ви можете скористатися інструкцією, щоб відразу встановити параметри для зв'язку, коли функція обміну даними не запущена. Нові параметри зв'язку не використовуються до наступного циклу, якщо запущена функція обміну даними; однак інструкція змінює параметри зв'язку для номера підключення.

Наприклад, коли функція обміну даними працює з параметрами, встановленими для з'єднання 3, ви можете вказати нові параметри для з'єднання 3. Нові параметри не використовуватимуться до наступного циклу, коли запуститься з'єднання 3.

8. Інструкція дозволяє лише змінювати параметри зв'язку. Зверніться до наступної таблиці для прапорів для запуску та закриття функції зв'язку, якщо ви використовуєте програму PLC для виконання цих функцій.

Коли ви встановлюєте функцію автоматичного сканування через ISPSoft, прапор запуску/зупинки номера підключення

автоматично оновлює стан запуску/зупинки після завершення сканування функцією обміну даними.

У наступній таблиці описано прапори SM.

№ SM	Атрибут	Пояснення до параметрів обміну даними COM1
SM750	R/W	Дозволяє обмін даними
SM752–SM783	R/W	Вмикає підключення для обміну даними 1–32
SM784–SM815	R	Успішне читання для з'єднань обміну даними 1–32
SM816–SM847	R	Прапори помилок для з'єднань обміну даними 1–32
№ SM	Атрибут	Пояснення параметрів обміну даними COM2
SM862	R/W	Дозволяє обмін даними
SM864–SM895	R/W	Вмикає підключення для обміну даними 1–32
SM896–SM927	R	Успішне читання для з'єднань обміну даними 1–32
SM928–SM959	R	Прапори помилок для з'єднань обміну даними 1–32

№ SM	Атрибут	Пояснення щодо параметрів обміну даними функціональної карти 1
SM1710	R/W	Дозволяє обмін даними
SM1712 ~ SM1743	R/W	Вмикає підключення для обміну даними 1–32
SM1744 ~ SM1775	R	Успішне читання для з'єднань обміну даними 1–32
SM1776 ~ SM1807	R	Прапори помилок для з'єднань обміну даними 1–32
№ SM	Атрибут	Пояснення щодо параметрів обміну даними функціональної карти 2
SM1822	R/W	Дозволяє обмін даними
SM1824 ~ SM1855	R/W	Вмикає підключення для обміну даними 1–32
SM1856 ~ SM1887	R	Успішне читання для з'єднань обміну даними 1–32
SM1888 ~ SM1919	R	Прапори помилок для з'єднань обміну даними 1–32

ПЛК встановлює прапорець на ON, коли прапорець успішного читання вказує, що отримання даних завершено і дані

перевірені є правильними. Якщо виникає помилка під час отримання даних або час очікування зв'язку, прапор помилки встановлюється на ON (див. коди помилок). ПЛК не скидає жодного з прапорів успішного читання чи помилки на ВІМК під час обміну даними, оскільки прапор успішного читання та прапор помилки кожного номера з'єднання не вмикаються одночасно.

9. У наступній таблиці описано SR (реєстри лише для читання) у функції обміну даними.

CP №	опис
SR1335	Цикл фактичних з'єднань 1–32 в обміні даними COM1
SR1336	Номер циклічного підключення в поточному обміні даними COM1
SR1340–SR1371	Коди помилок з'єднань 1–32 в обміні даними COM1
SR1375	Цикл фактичних з'єднань 1–32 в обміні даними COM2
SR1376	Номер циклічного підключення в поточному обміні даними COM2
SR1380–SR1411	Коди помилок для підключень 1–32 в обміні даними COM2
SR1435	Цикл фактичних з'єднань 1–32 в обміні даними функціональної карти 1
SR1436	Номер циклічного підключення в поточному обміні даними функціональної карти 1
SR1440 ~ SR1471	Коди помилок з'єднань 1–32 в обміні даними функціональної карти 1
SR1475	Цикл фактичних з'єднань 1–32 в обміні даними функціональної карти 2
SR1476	Номер циклічного підключення в поточному обміні даними функціональної карти 2
SR1480 ~ SR1511	Коди помилок з'єднань 1–32 в обміні даними функціональної карти 2

10. Функція обміну даними не забезпечує прапор успішного запису. Рекомендується звернутися до підключення номер під час виконання, щоб судити про успішність запису даних чи ні.

Наприклад, коли номер виконаного з'єднання дорівнює 3 у SR1336, послідовні дії зв'язку полягають у тому, щоб спочатку прочитати дані зв'язку, а потім записати дані зв'язку після завершення читання. Після завершення запису номер підключення збільшується до 4.

11. Якщо значення довжини даних для зчитування на **S6** і вихідний пристрій на **S7** виходять за межі діапазону пристрою **D** або **M**, значення довжини на **S6** **автоматично** встановлюється **таким**, щоб знаходився в допустимому діапазоні. Наприклад, якщо значення довжини в **S6** дорівнює **100**, а початковим пристроєм **S7** є **M8182**, значення в **S6** автоматично встановлюється на **10**.

12. У наведених нижче випадках інструкція не виконується, і налаштування параметрів у HWCONFIG не змінюються для порту зв'язку, а для **SM0** встановлено значення **ON**, а код помилки в **SR0** – **16#200B**.

- Помилка введення виникає, коли значення параметрів **S₁**, **S₂**, **S₄** і **S₈** виходять за межі вказаного діапазону.
- Коли функціональний код **S₄** або **S₈** вибирає тип біта для читання або запису даних, локальний пристрій для зберігання даних **S₇** або **S₁₁** повинні вибрати пристрій **M**. Помилка введення виникає, якщо вибір не **M**.
- Коли код функції **S₄** або **S₈** вибирає тип слова для читання або запису даних, локальний пристрій для зберігання даних **S₇** або **S₁₁** повинен вибрати пристрій **D**. Помилка введення виникає, якщо вибір не **D**.

Приклад: AS COM1 (RS485)

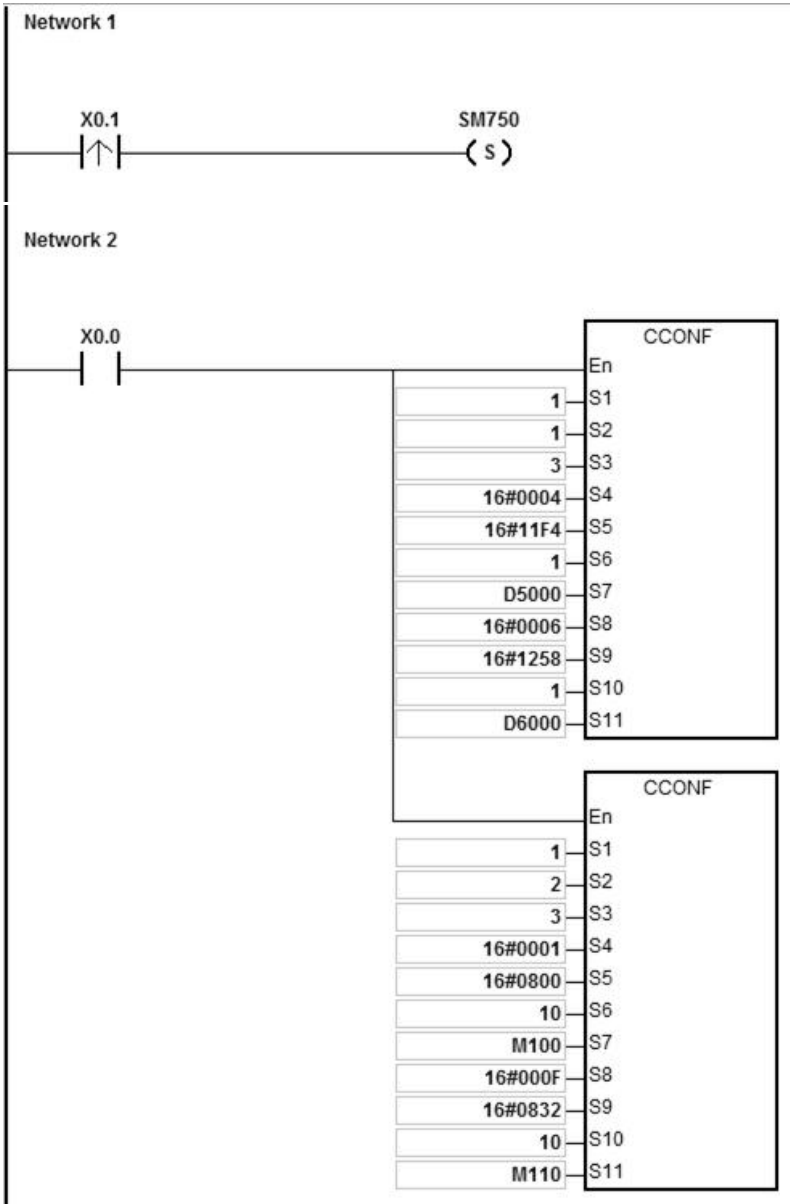
1. Для обміну даними між ЦП серії AS і ЦП DVP-ES2 у наступній таблиці показано таблицю обміну даними COM1 у HWConfig в ISPSoft.

Пункт №.	Спосіб виявлення	Адреса віддаленої станції	Локальний пристрій		Віддалений пристрій	Довжина	Код функції	Як почати
1	Вказане підключення	2	D50	←	D50	50	H03	Програмний контроль
			D100	→	D100	50	H10	
2	Вказане підключення	2	D150	←	D150	50	H03	Програмний контроль
			D200	→	D200	50	H10	

2. Перед початком обміну даними припустимо, що відповідні дані між ЦП серії AS і DVP-ES2 ЦП перераховані в наступній таблиці.

AS PLC (головний)	Цінність змісту	ПЛК ES2 (підлеглий)	Цінність змісту
D50-D99	0	D50-D99	1-50
D100-D149	100-149	D100-D149	0
D150-D199	0	D150-D199	150-199
D200-D249	200-249	D200-D249	0

3. Обмін даними між ЦП серії AS і ЦП DVP-ES2 починається, коли X0.1 увімкнено.



4. Після початку обміну даними відповідні дані між ЦП серії AS і ЦП DVP-ES2 змінюються як показано в наступній таблиці.

AS PLC (головний)	Цінність змісту	ПЛК ES2 (підлеглий)	Цінність змісту
D50-D99	1-50	D50-D99	1-50
D100-D149	100-149	D100-D149	100-149
D150-D199	150-199	D150-D199	150-199
D200-D249	200-249	D200-D249	200-249

5. Коли X0.0 увімкнено, параметри таблиці обміну даними COM1 у ЦП серії AS змінюються, як показано в наступній таблиці.

Пункт №.	Спосіб виявлення	Адреса віддаленої станції	Локальний пристрій		Віддалений пристрій	Довжина	Код функції	Як почати
1	Вказане підключення	3	D5000	←	D500	1	H04	Програмний контроль
			D6000	→	D600	1	H06	
2	Вказане підключення	3	M100	←	M0	10	H01	Програмний контроль
			M110	→	M50	10	H0F	

6. Через зміну параметрів таблиці обміну даними COM1 відповідні дані між ЦП серії AS і ЦП DVP-ES2 змінено, як показано в наступній таблиці.

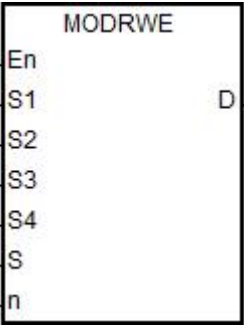
AS PLC (головний)	Цінність змісту	ПЛК ES2 (підлеглий)	Цінність змісту
D5000	3000	D500	3000
D6000	4000	D600	4000
M100-M109	ON	M0-M9	ON
M110-M119	OFF	M50-M59	OFF

API	Код інструкції			Операнд						функція						
1817		MODRWE		S ₁ , S ₂ , S ₃ , S ₄ , S, n, D						Читання та запис даних Modbus без використання будь-яких прапорів						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁								●	●				○	○		
S ₂								●	●				○	○		
S ₃								●	●				○	○		
S ₄								●	●				○	○		
S								●								
n								●	●				○	○		
D		●	●	●				●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●				●							
S ₂		●				●							
S ₃		●				●							
S ₄		●				●							
S	●	●				●							
n		●				●							
D	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- S_1 : Номер комунікаційного порту
- S_2 : Адреса підрозділу
- S_3 : Код функції
- S_4 : Адреса пристрою
- S : Зареєструватися для читання та запису даних
- n : Довжина даних
- D : Прапор завершення читання та запису даних

Пояснення

- S_1 встановлює серійний номер комунікаційного порту. Кількість COM1 – 1, COM2 – 2, Card1 – 11 і Card2 – це 12. Якщо значення перевищує допустимий діапазон, інструкція не отримує жодних даних зв'язку.
- Зверніться до пояснення інструкції MODRW (API 1808), щоб дізнатися про значення операндів S_2, S_3, S_4, S і n .
- D встановлює прапорці стану зв'язку, коли інструкція завершує зв'язок, і прапорці займають 3 послідовних пристроїв. Ви повинні скинути прапорці на OFF У наступній таблиці наведено пояснення держав прапора.

Операнд	опис
D	Успішне отримання даних.
D +1	Під час отримання даних сталася помилка.
D +2	Прапор тайм-ауту прийому

ПРИМІТКА. Серед трьох прапорів стану лише один прапор встановлено на ON, а також встановлено відповідні спеціальні прапори (SM).

щоразу, коли зв'язок завершується. Зверніться до розділу 6.19.3 щодо використання спеціальних позначок.

4. Відлік часу для надсилання інструкції починається з моменту її початку. Ви повинні вимкнути інструкцію для циклу сканування після завершення зв'язку. Наступну інструкцію зв'язку можна надіслати, щойно інструкція буде перезапущена.

5. Комунікаційна дія та керуюча послідовність інструкції подібні до інструкції MODRW (API 1808). Єдина відмінність між двома інструкціями полягає в тому, що ви можете надіслати команду зв'язку без контролю над прапором для надсилання даних.

6. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.

приклад

Ви можете порівняти програму MODRW і програму MODRWE, наприклад, за допомогою ПЛК COM1 і коду функції 03, щоб зчитувати вісім фрагментів даних із D20 у DVP-ES2. Те саме стосується інших комунікаційних портів ПЛК. Зверніться до інструкції MODRW (API 1808) і наступного прикладу щодо використання інших кодів функцій.

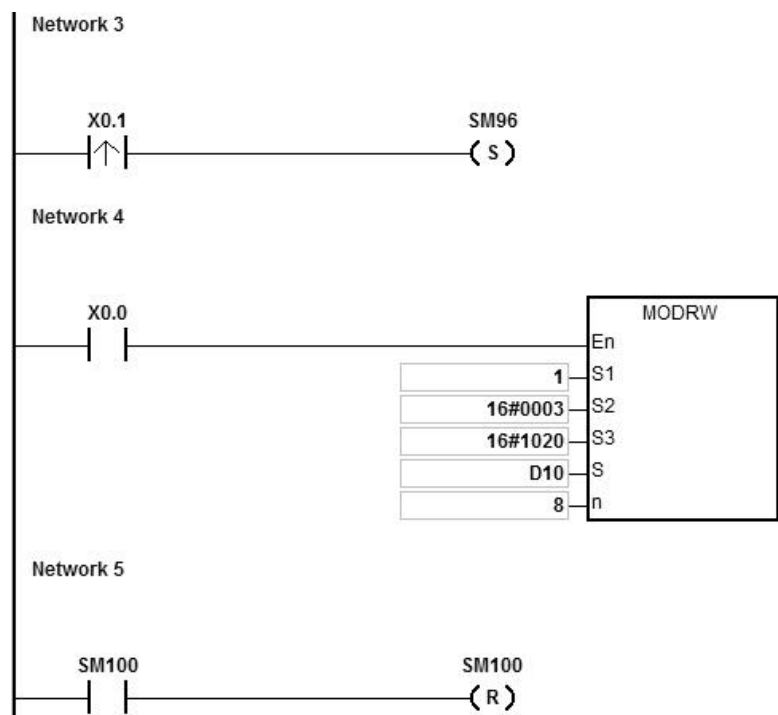
1. Адреса пристрою D20 у ЦП DVP-ES2 (16#1020) і значення вмісту в D20–D27 показані в наступна таблиця.

пристрій	D20	D21	D22	D23	D24	D25	D26	D27
Значення (16#)	1234	5678	1122	3344	5566	7788	99AA	BBCC

2. ПЛК серії AS зчитує значення вмісту з D20–D27 у ЦП DVP-ES2 через зв'язок.

Спосіб 1: Використання інструкції MODRW

Дані в D20-D27 DVP-ES2 зчитуються, коли SM96 увімкнено, а X0.0 увімкнено.



У наведеній нижче таблиці пояснюються операнди MODRW.

Операнд	опис	пристрій
S ₁	Адреса підрозділу	16#0001
S ₂	Код функції	16#0003
S ₃	Читання адреси пристрою D20	16#1020
S	Перший регістр для збереження прочитаних даних	D10
n	Читання довжини даних	8

Відповідь на зв'язок між ЦП серії AS і DVP-ES2 залежить від режиму.

Режим ASCII: вам не потрібно перетворювати коди ASCII, і вони виражені у значеннях 16#.

- AS надсилає команду зв'язку: «01 03 10 20 00 08 C4 CR LF»
- AS отримує команду зв'язку: «01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC AA CR LF»

Режим RTU

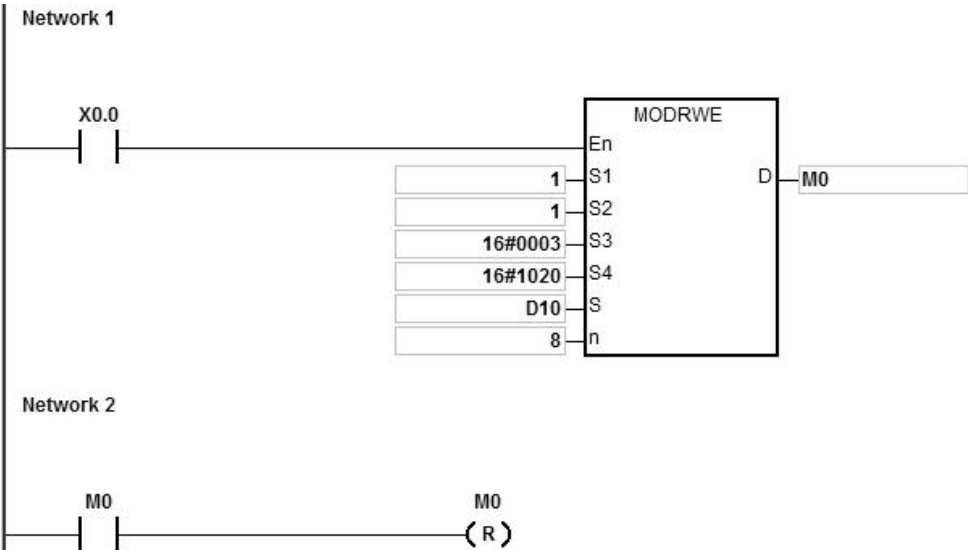
- AS надсилає команду зв'язку: "01 03 10 20 00 08 41 06"
- AS отримує команду зв'язку: «01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC 90
FE"

SM100 увімкнено, якщо в даних немає помилок.

Після отримання даних, надісланих з DVP-ES2, ПЛК підтверджує формат даних, надісланих назад з DVP-ES2. Якщо у форматі немає помилок, SM100 увімкнено; інакше SM102 увімкнено. SM104 увімкнено, якщо немає відповіді.

Спосіб 2: Використання інструкції MODRWE

Дані в D20-D27 DVP-ES2 зчитуються, коли X0.0 увімкнено.



У наведеній нижче таблиці пояснюються операнди MODRWE.

Операнд	опис	пристрій
S1	Номер комунікаційного порту	16#0001
S2	Адреса підрозділу	16#0001
S3	Код функції	16#0003
S4	Читання адреси пристрою D20	16#1020
S	Перший регістр для збереження прочитаних даних	D10
n	Читання довжини даних	8
D	Позначка завершення читання та запису даних	M0

Відповідь зв'язку між центральним процесором серії AS і DVP-ES2 залежить від режиму.

Режим ASCII: вам не потрібно перетворювати коди ASCII, і вони виражені у значеннях 16#.

- AS надсилає команду зв'язку: «01 03 10 20 00 08 C4 CR LF»
- AS отримує команду зв'язку: «01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC AA CR LF»

Режим RTU

- AS надсилає команду зв'язку: «01 03 10 20 00 08 41 06»
- AS отримує команду зв'язку: «01 03 10 12 34 56 78 11 22 33 44 55 66 77 88 99 AA BB CC 90 FE»

M0 увімкнено, якщо в даних немає помилок.

Після отримання даних, надісланих з DVP-ES2, ПЛК підтверджує формат даних, надісланих з DVP-ES2.

Якщо у форматі немає помилок, M0 увімкнено; інакше M1 увімкнено. M2 увімкнено, якщо немає відповіді.

Відповідні спеціальні прапорці SM100, SM102 і SM104 також увімкнені.

3. У наведеній нижче таблиці наведено значення вмісту в D10–D17 AS CPU.

пристрій	D10	D11	D12	D13	D14	D15	D16	D17
Значення (16#)	1234	5678	1122	3344	5566	7788	99AA	BBCC

Додаткові зауваження

1. Якщо ви оголошуєте операнд у IPSSoft, типом даних є ARRAY [3] із BOOL.
2. Якщо D +2 перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API		Інструкція			Операнд							опис					
1818		DNETRW			S ₁ , S ₂ , S ₃ , S ₄ , S ₅ , S ₆ , S ₇ , S ₈ , S ₉ , S ₁₀ , D ₁ , D ₂ , D ₃ , D ₄ , D ₅							Читання та запис даних зв'язку DeviceNet					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁								●	●				○	○			
S ₂								●	●				○	○			
S ₃								●	●				○	○			
S ₄								●	●				○	○			
S ₅								●	●				○	○			
S ₆								●	●				○	○			
S ₇								●	●				○	○			
S ₈								●									
S ₉								●	●				○	○			
S ₁₀								●	●				○	○			
D ₁		●	●	●													
D ₂		●	●	●													
D ₃								●									
D ₄								●									
D ₅								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●				●							
S ₂		●				●							
S ₃		●				●							
S ₄		●				●							
S ₅		●				●							
S ₆		●				●							
S ₇		●				●							
S ₈		●				●							
S ₉		●				●							
S ₁₀		●				●							
D ₁	●												
D ₂	●												
D ₃		●				●							
D ₄		●				●							
D ₅		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ

DNETRW	
En	
S1	D1
S2	D2
S3	D3
S4	D4
S5	D5
S6	
S7	
S8	
S9	
S10	

- S₁** : Номер модуля, який надсилає DeviceNet спілкування
- S₂** : DeviceNet MAC ID
- S₃** : Код послуги
- S₄** : ID класу
- S₅** : Ідентифікатор екземпляра
- S₆** : ID атрибута
- S₇** : Довжина записаних даних
- S₈** : Пристрій для зберігання записаних даних
- S₉** : Тайм-аут зв'язку
- S₁₀** : Кількість разів повторної передачі
- D₁** : Прапор завершення
- D₂** : Прапор помилки
- D₃** : Код помилки
- D₄** : Довжина прочитаних даних
- D₅** : Пристрій для зберігання прочитаних даних

Пояснення

- S₁** — серійний номер модуля праворуч від ПЛК. Перший під номером 1, другий під номером 2 і так далі. Будь-які модулі праворуч від ПЛК повинні бути пронумеровані. Діапазон налаштувань 1~32. Якщо значення налаштування (<1 або >32) перевищує діапазон, інструкція виконуватиметься з мінімальним або максимальним значенням.
- S₂** — ідентифікатор DeviceNet Mac у діапазоні: 0~63. Це може бути Mac ID підлеглого пристрою, який має прочитати головний запису, а також Mac ID головного пристрою, що означає читання та запис даних у головному пристрої.
- S₃** — код служби DeviceNet.

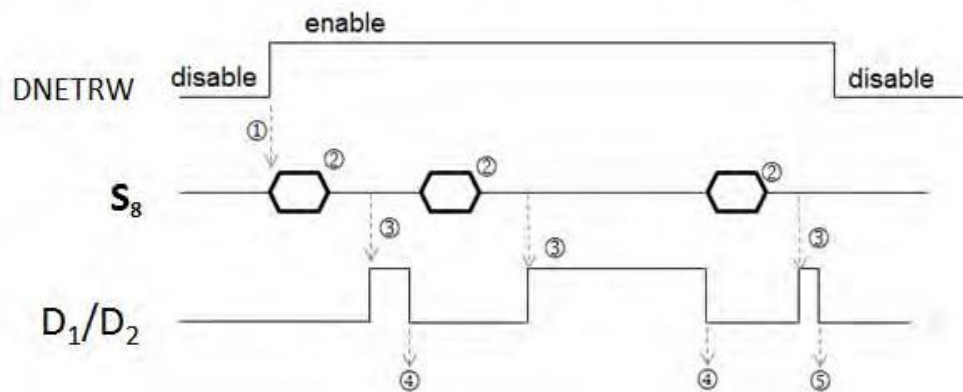
Сервісний код	опис
0x01	Читати всі атрибути (Get_Attribute_All)
0x02	Установити всі атрибути (Set_Attribute_All)
0x0E	Прочитайте один атрибут (Get_Attribute_Single)
0x10	Установіть один атрибут (Set_Attribute_Single)

- S₄**, **S₅** і **S₆** — це відповідно ідентифікатор класу, ідентифікатор екземпляра та ідентифікатор атрибута для визначення шляху параметра в EPATH протоколу DeviceNet.

5. S_7 — довжина записаних даних з одиницею вимірювання байт.
6. S_8 є початковою адресою пристроїв, де записані дані зберігаються в порядку від молодшого біта до старшого біта.
7. S_9 - час очікування зв'язку. Діапазон значень налаштування становить 1~100, а одиниця – 0,1 с (секунди).
8. S_{10} – це кількість разів повторної передачі, а діапазон становить 0–3. Після закінчення часу очікування зв'язок буде передано знову.
9. D_3 відображає код помилки читання або запису.

Код помилки		опис
Код 1 (старший байт)	Код 2 (молодший байт)	
XX	FF	Не відповідає стандарту DeviceNet
20	01	Цільовий підпорядкований пристрій не існує.
20	02	Не вдалося встановити з'єднання з підлеглим.
20	03	Не вдалося надіслати явне повідомлення.
16	00	Час очікування зв'язку

10. D_4 — довжина зчитованих даних у байтах.
11. D_5 є початковою адресою пристроїв, де зчитані дані зберігаються в порядку від молодшого до старшого біта.
12. D_1 — це позначка завершення зв'язку, а D_2 — позначка помилки. Дивіться детальну процедуру надсилання повідомлення і діаграма послідовності.
- ① Інструкцію DNETRW увімкнено вперше, і команда буде надіслано негайно, якщо інструкція не може бути зайнята іншим зв'язком DeviceNet.
 - ② Надсилається команда.
 - ③ Надсилання даних завершено, а позначка завершення або позначка помилки змінюється на Увімк. відповідно до відповіді.
 - ④ Наступне повідомлення готове до надсилання. Наступна команда надсилається одразу після видалення позначки завершення або помилки.
 - ⑤ Відправка команди завершена, інструкція DNETRW вимкнена.



13. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який називається

тільки один раз.

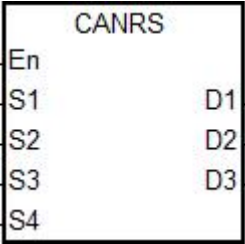
API	Інструкція			Операнд						опис							
1819		CANRS		$S_1, S_2, S_3, S_4, D_1, D_2, D_3$						Надсилання та отримання користувацького зв'язку CAN							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●	●				○	○		
S_2								●	●							
S_3								●	●				○	○		
S_4								●	●							
D_1								●								
D_2		●	●	●				●								
D_3		●	●	●				●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
D_1		●				●							
D_2	●												
D_3	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

6
СИМВОЛ



- S_1 : Налаштування режиму зв'язку
- S_2 : Ідентифікатор зв'язку (MsgID)
- S_3 : Довжина надісланих даних
- S_4 : Пусковий пристрій, на якому зберігаються надіслані вихідні дані
- D_1 : Пусковий пристрій, де зберігаються отримані дані
- D_2 : Прапор завершення зв'язку
- D_3 : Прапор помилки зв'язку

Пояснення

- Інструкція CANRS застосовна до ПЛК серії AS із мікропрограмою версії 1.04.00 і новішої. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.
- Перед виконанням інструкції CANRS переконайтеся, що HWCONFIG для конфігурації апаратного забезпечення вибрав CANopen DS301 як режим функціональної карти 2.

3. S_1 встановлює режим зв'язку. Перегляньте наступні режими, які підтримує інструкція. Якщо значення параметра неправильне, позначка помилки D_3 змінюється на Увімк., а код помилки SR659 встановлюється на 1.

спілкування		Значення параметра S_1	Формат спілкування	опис
Високий 8 біти	Низький 8 біти			
0	0	16#0000	2.0 A 11-бітний ідентифікатор	Головний режим. Майстер буде чекати отримання відповідне повідомлення після надсилання даних зв'язку.
	1	16#0001		Головний режим. Після надсилання широкомовної команди ведучий увійде в режим прийому повідомлень від кількох підлеглих, поки не настане тайм-аут отримання або кількість отриманих повідомлень не досягне 100.
	2	16#0002		Підлеглий режим. Підлеглий отримує повідомлення першим і потім надсилає повідомлення відповіді.
	3	16#0003		Підлеглий режим. Підлеглий тільки отримує повідомлення.
1	0	16#0100	2.0B 29-бітний ідентифікатор	Головний режим. Майстер буде чекати отримання відповідь після надсилання повідомлення.
	1	16#0101		Головний режим. Після надсилання широкомовної команди ведучий увійде в режим прийому повідомлень від кількох підлеглих, поки не настане тайм-аут отримання або кількість отриманих повідомлень не досягне 100.
	2	16#0102		Підлеглий режим. Підлеглий отримує повідомлення першим і потім надсилає повідомлення відповіді.
	3	16#0103		Підлеглий режим. Підлеглий тільки отримує повідомлення.

4. S_2 - ідентифікатор переданого повідомлення. Відповідно до протоколу 2.0A або 2.0B дані передаються автоматично

займає D буферних регістрів.

Якщо вибрано 2.0A, S_2 – це 11 біт ідентифікаційного коду з таким форматом передачі даних.

S_2 №	S_2
опис	повідомлення ID

Якщо вибрано 2.0B, S_2 (Lo-слово) і S_2+1 (Hi-слово) є 29 бітами ідентифікаційного коду.

S_2 №	S_2	S_2+1
опис	повідомлення ID (Lo-word)	повідомлення ID (Привіт-слово)

5. S_3 - довжина переданого повідомлення. Значення параметра має бути в діапазоні 0~8 з одиницею вимірювання байт (8 біт). Якщо значення налаштування (<0 або >8) перевищує діапазон, інструкція виконуватиметься з мінімальним значенням 0 або максимальним значенням 8. Якщо довжина переданого повідомлення дорівнює 0, режим зв'язку автоматично зміниться на підлеглий режим на отримувати повідомлення без надсилання даних. Режим можна використовувати для моніторингу пакету зв'язку.

6. S_4 є початковим пристроєм, де зберігаються передані дані та використовуються лише наступні 8 бітів даних.

Наприклад, 4 повідомлення передаються з D10 як пусковим пристроєм. Послідовність передачі даних див. нижче.

S_4 №	D10	D11	D12	D13
опис	Дані1	Дані2	Дані3	Дані4

7. Якщо S_1 є головним режимом, у якому головний буде чекати отримання даних після надсилання даних, або підлеглим режимом,

отримані дані будуть безпосередньо збережені в пристрої, визначеному D_1 . D100 визначається D_1 . Тут див. формат збереженого вмісту.

Налаштування режиму 2.0A:

D_1 №	D100	D101	D102 ~ D109 (молодші 8 бітів)
опис	повідомлення ID	Довжина даних	Дані1 ~ Дані8

Налаштування режиму 2.0B

D_1 №	D100	D101	D102	D103 ~ D110 (молодші 8 біт)
опис	повідомлення ID (Lo-word)	повідомлення ID (Привіт-слово)	Довжина даних	Дані1 ~ Дані8

Примітка. Якщо повідомлення Msg. ID, який потрібно отримати, необхідно вказати на етапі отримання даних, встановити значення $D_1 / D_1 + 1$

на основі режиму 2.0A/2.0B. Якщо повідомлення Msg. ID не вказано, будь ласка, очистіть значення $D_1 / D_1 + 1$ до 0 перед отриманням даних.

8. Якщо S_1 є основним режимом широкомовної передачі, отримані дані будуть зберігатися на пристрої, зазначеному D_1 . D100 тут визначено D_1 . Формат зберігання див. нижче.

Вибір режиму 2.0A: (тут показано отримання даних від 2 підлеглих пристроїв. Для інших даних збільште номер пристрою, визначений D_1)

Відповідь послідовність		Дані першого раба			Дані з другого раб		Дані з третій раб
D_1 №	D100	D101	D102	D103 ~ D110 (Молодші 8 біт)	D111	D112~D120	D121~130
опис	Отримання Номер пакета	повідомле ння ID	Довжина даних	Дані1 ~ Дані8	повідомлення ID	Довжина, дані	ID, довжина, дані

Вибір режиму 2.0B: (тут показано отримання даних від 2 підлеглих пристроїв. Для інших даних збільште кількість D_1)

Відповідь послідовність	Дані першого раба					Дані з другого раб
D_1 №	D100	D101	D102	D103	D104 ~ D111 (Молодші 8 біт)	D112~D122
опис	Отримання Номер пакета	повідомлення ID (Людне слово)	повідомлення ID (Привіт-слово)	Довжина даних	Дані1 ~ Дані8	ID, довжина, дані

9. D_2 і D_3 є прапором завершення отримання та прапорцем помилки відповідно. Два прапорці будуть оцінювати отримання даних

завершується або виникає помилка на основі поточного стану зв'язку під час сканування інструкції. Якщо отримання даних завершено або сталася помилка, прапорці зміняться на Увімк. Два прапорці будуть скинуті та змінюватимуться на Вимкнено автоматично щоразу, коли ввімкнено інструкцію.

10. Коли для інструкції встановлено головний режим, рекомендується використовувати налаштування тайм-ауту отримання в HWCONFIG. Отже, D_3 зміниться на Увімк., а код помилки 2 буде записаний у SR659, якщо пакет зв'язку не було отримано повністю протягом зазначеного періоду часу.

Якщо для тайм-ауту отримання встановлено значення 0, це вказує на те, що тайм-аут зв'язку не обмежений і статус можна застосувати до підлеглого режиму.

Примітка : Право на керування зв'язком не може бути звільнено, доки інструкцію не буде вимкнено вручну, коли

використовується метод, у якому немає обмежень на час очікування зв'язку.

11. Кількість разів використання інструкції не обмежена. Але є лише одна команда зв'язку CAN дозволено надсилати кожного разу. Якщо в даний момент надсилається або отримується одна команда, наступну інструкцію CANRS не можна ввімкнути.

12. Ось формат CAN BUS і кожен фрагмент вмісту для Msg. ID пояснюється нижче.

Оскільки вибрано протокол 2.0A і значення S_2 становить 16#0123, повідомлення Msg. Вміст ідентифікатора показано в наступній таблиці.

Номер біта	15 ~ 11	10 ~ 8	7 ~ 4	3 ~ 0
Значення S_2 (16 біт)	-	1	2	3

Оскільки вибрано протокол 2.0B, значення S_2 встановлено на 16#1234 (Lo-слово), а $S_2 + 1$ — 16#0567 (Hi-слово), Msg.

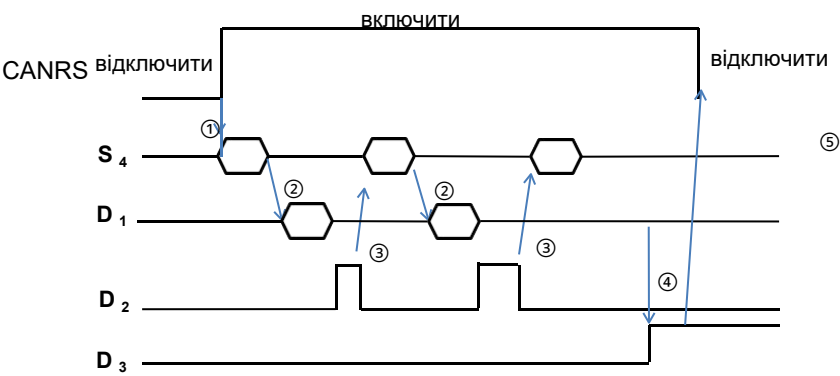
Вміст ідентифікатора показано в наступній таблиці.

Номер біта	31 ~ 29	28	27 ~ 24	23 ~ 20	19 ~ 16	15 ~ 0
S_2 (32 біти)	-	0	5	6	7	1234

13. Якщо операнд D_1 оголошено у змінній, рекомендується використовувати матрицю типу Word.

14. Див. схему послідовності зв'язку CANRS і пояснення.

- ① : Інструкцію CANRS увімкнено. Якщо жодна інша інструкція CANRS не займає права керування, прапорці **D₂** і **D₃** очищаються, а потім дані надсилаються негайно.
- ② : Надсилання даних завершено, а тим часом починається прийом даних. Після завершення прийому прапор завершення **D₂** змінюється на Увімк.
- ③ : інструкція надсилає дані знову після того, як ви знімете прапорець **D₂**.
- ④ : У міру закінчення часу очікування **D₃** змінюється на Оп.
- ⑤ : Якщо виявлено помилку, ви можете вимкнути інструкцію. Крім того, ви можете мати інструкцію для повторного надсилання даних після очищення позначки **D₃**.



6

Приклад 1

Головний режим

За допомогою інструкції CANRS 6-байтові дані в D10~D15 передаються на підлеглий пристрій, а дані відповіді з підлеглого пристрою зберігаються в пристроях, починаючи з D100. Після завершення надсилання та отримання M100 змінюється на Увімк.

Див. пояснення відповідних параметрів.

Головний режим_MsgID=2	
Режим зв'язку	0= головний режим, очікує на отримання відповіді від підлеглого після відправки даних.
ID зв'язку	MsgID надісланого повідомлення D0=1
Довжина надісланих даних	6 БАЙТ
Пусковий пристрій, де надіслано дані зберігаються	D10

Пусковий пристрій, де зберігаються отримані дані	D100 для зберігання MsgID, D101 для зберігання кількості отриманих пакетів D102.... для зберігання отриманих даних
Завершення спілкування прапор	M100
Прапор помилки зв'язку	M101

1. Оскільки M1 увімкнено, установіть MsgID надісланого повідомлення на 1 та MsgID отриманого повідомлення на 2. Потім дані починається відправка.

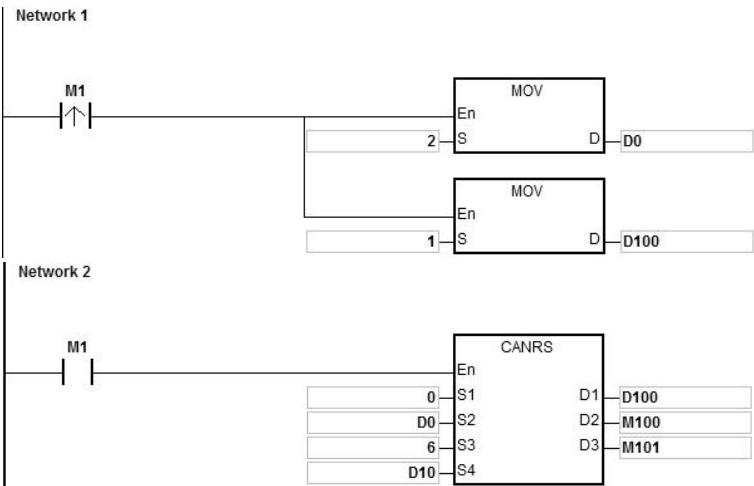
A. Виконується надсилання даних у D10~D15 (6 байтів) на підлеглий пристрій.

пристрій	D10~D15
Зміст	H0A (визначено користувачами)

B. Отримані дані відповіді від підлеглого пристрою зберігаються в D100~D109. M100 змінюється на Увімк., коли відбувається отримання зроблено.

пристрій	D100	D101	D102~D109
Зміст	2	8	H0B

2. Майстер-приклад програмування



Приклад 2 Режим

Slave

Використовуючи інструкцію CANRS, отримані дані від головного пристрою зберігаються в пристроях, починаючи з D120, а 8-байтові дані в D20~D27 надсилаються назад. Після завершення надсилання та отримання M110 змінюється на Увімк.

Див. пояснення відповідних параметрів.

Підлеглий режим_MsgID=1	
Режим зв'язку	2= Підлеглий режим, надсилає дані після отримання даних.
ID зв'язку	MsgID надісланого повідомлення, D0=2
Довжина надісланих даних	8 БАЙТ
Пусковий пристрій де надіслані дані зберігаються	D20
Пусковий пристрій, де зберігаються отримані дані	D120 для зберігання MsgID D121 для зберігання кількості отриманих пакетів D122.... для зберігання отриманих даних
спілкування прапор завершення	M110
Прапор помилки зв'язку	M111

- Оскільки M1 увімкнено, установіть MsgID надісланого повідомлення на 2 та MsgID отриманого повідомлення на 1, а потім зачекайте, отримати дані.

А. Отримані 6-байтові дані від майстра поміщаються в D120~D127.

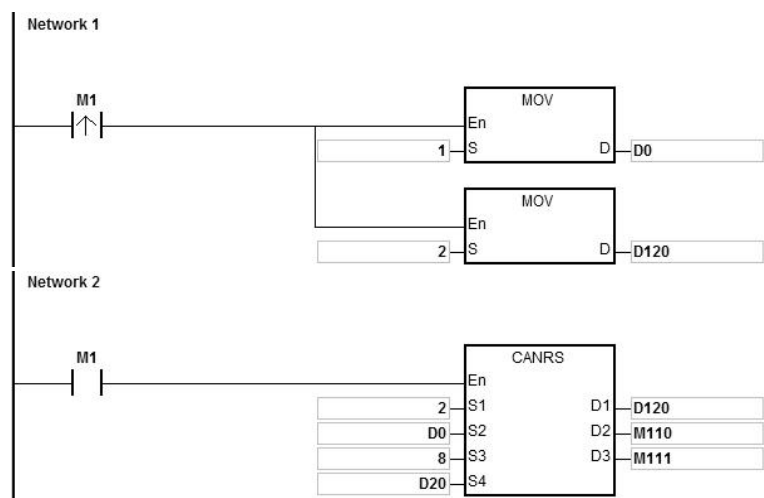
пристрій	D120	D121	D122~D127
Зміст	1	6	H0A

В. Потім 8-байтові дані в D20~D27 починають надсилатися назад головному. M110 змінюється на Увімк., коли надсилаються дані

зроблено.

пристрій	D10~D15
Зміст	H0B (визначено користувачами)

2. Приклад підлеглого програмування



API	Інструкція			Операнд					опис							
1820		DMVSH		Mode ~ ErrCode					Увімкнення виявлення та зв'язку Delta DMV							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Mode								●					○	○		
Start1		●	●	●				●								
Start2		●	●	●				●								
Ready	●															
Com No								●					○	○		
Id_lp								●					○	○		
Addres								●					○	○		
Length								●					○	○		
Shoot1		●														
Shoot2		●														
RdData								●								
OK		●	●	●				●								
Err		●	●	●				●								
ErrCode								●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Mode		●											
Start1	●												
Start2	●												
Ready	●												
Com No		●				●							
Id_lp		●				●							
Addres		●				●							
Length		●				●							
Shoot1	●												
Shoot2	●												
RdData		●				●							
OK	●												
Err	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

DMVSH	
En	
Mode	Shoot1
Start1	Shoot2
Start2	RdData
Ready	Ok
ComNo	Err
Id_Ip	ErrCode
Address	
Length	

Mode	: Спосіб запуску та отримання
Start1	: Тригерний сигнал для запуску set1
Start2	: Тригерний сигнал для запуску set2
Ready	: Сигнал завершення при отриманні даних відповіді від DMV є повний
Com No	: Послідовний порт або порт мережевого зв'язку для ПЛК для надсилання або отримати дані
Id_Ip	: Адреса станції послідовного зв'язку або IP-адреса мережі DMV
Addres	: Пусковий пристрій, де зберігаються дані DMV, які потрібно зчитати
Length	: Довжина даних для читання
Shoot1	: Вихідний сигнал, коли ПЛК повідомляє DMV, щоб увімкнути виявлення set1
Shoot2	: Вихідний сигнал, коли ПЛК повідомляє DMV, щоб увімкнути виявлення set2
RdData	: Пусковий пристрій, де дані, які ПЛК отримує від DMV зберігаються
OK	: Прапор успішного спілкування
Err	: Прапор помилки зв'язку
ErrCode	: Код помилки зв'язку

Пояснення

1. Інструкція DMVSH застосовна до ПЛК серії AS із мікропрограмою V1.06.00 і пізніших.

2. **Mode** встановлює метод, за допомогою якого ПЛК запускає DMV, включаючи DMV1000 і DMV2000, і метод отримання. Є 2 режими: 0 і 1 для опції. Якщо налаштування виходить за межі діапазону, ПЛК автоматично працюватиме в режимі 0.

3. Будь ласка, зверніться до наступного прикладу пояснення функції **Start1** , **Start2** , **Ready** , **Shoot1** , **Shoot2** , **RdData**

і **Ok** і процес запуску виявлення.

4. **ComNo** встановлює номер комунікаційного порту ПЛК. Цифра 1 позначає COM1, цифра 2 – COM2, цифра 11 – Card1, цифра 12 – Card2, а цифра 4 – порт Ethernet. Якщо налаштування не є одним із згаданих вище чисел або означає порт зв'язку, який ПЛК не підтримує, прапор помилки **Err** зміниться на On і з'явиться код помилки 16#0001.

5. **Id_Ip** встановлює адресу станції зв'язку (Mac ID) підпорядкованого DMV або IP-адресу мережі. **Address** – це адреса зв'язку, де зчитується результат виявлення DMV. **Length** — це довжина даних виявлення зчитування.

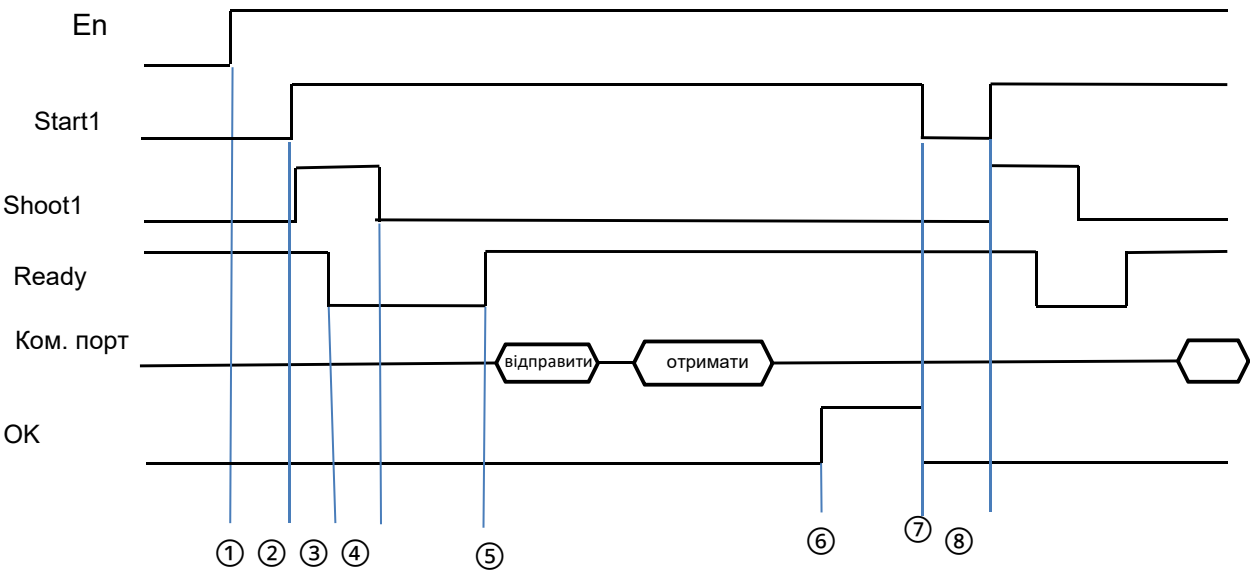
6. Див. пояснення значень **ErrCode** у таблиці нижче.

Код помилки	опис	Виправлення
16#0001	Вказаний порт зв'язку неправильний.	Вкажіть новий номер порту зв'язку.
16#0002	Помилка відповіді DMV	Перевірте адресу зв'язку
16#0003	Немає відповіді від DMV	Перевірте, чи правильно під'єднано кабель або DMV увімкнено.
16#0004	Очікування відповіді від DMV; DMV виявлення не можна вмикати повторно.	Змініть час увімкнення виявлення DMV.

Приклад 1

Mode=0

Тільки set1 використовується для ініціювання виявлення DMV і отримання даних відповіді DMV. Дивіться схему контролю послідовності нижче.



Пояснення таймінгу:

- ① Інструкція DMVSH включена.
- ② Встановіть для Start1 значення On, щоб сповістити інструкцію про надсилення вихідного сигналу **Shoot1** (ON) протягом приблизно 10 мс) і повідомити DMV, щоб увімкнути функцію виявлення.
- ③ Після того, як DMV отримає тригерне повідомлення, для параметра **Ready** змінено значення On (Увімкнено) на Off (Вимкнено).
- ④ ПЛК автоматично вимикає **Shoot1** на Off.

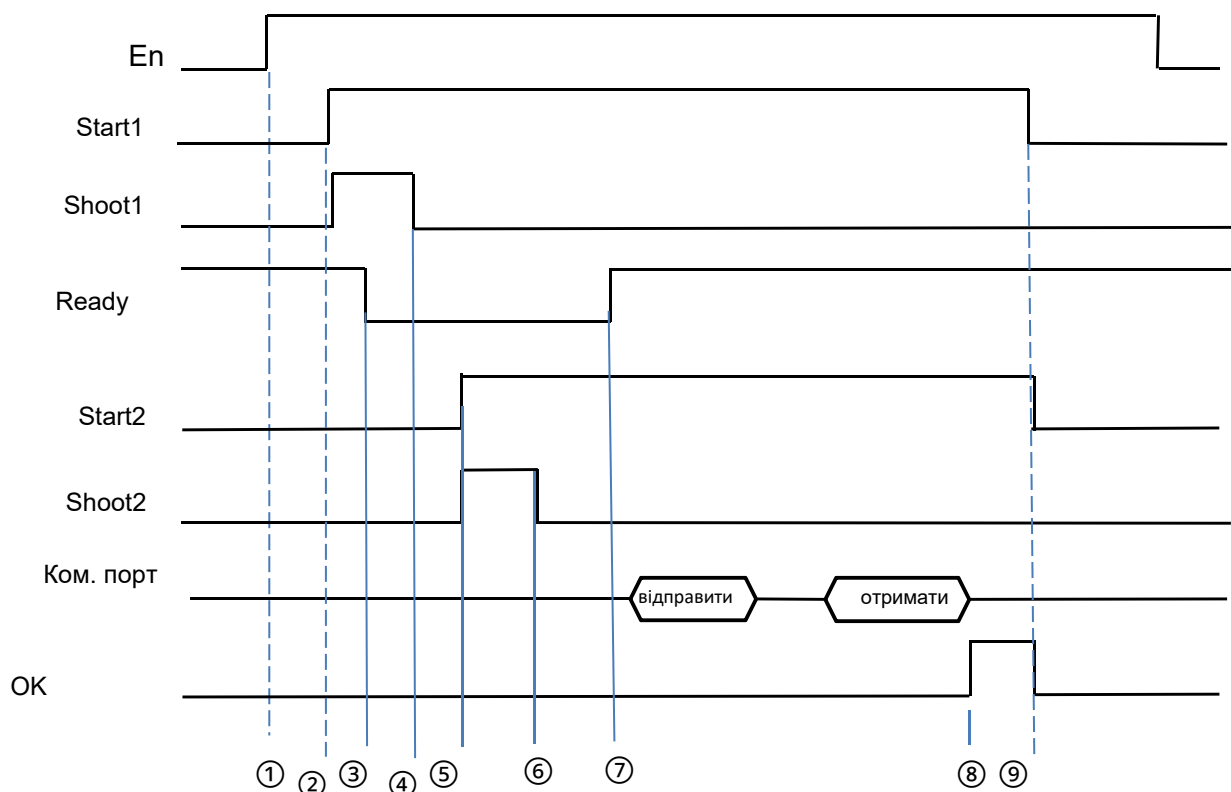
- ⑤ Після завершення виявлення DMV для параметра **Done** встановлено значення Вимк. на Увімк. ПЛК починає оцінювати, чи включено **Ready**, після очікування в 1,5 рази більше часу фільтрації вхідних даних. ПЛК отримує повідомлення про те, що **Ready** змінюється з Off на On, і надсилає команду читання через Modbus 0x03.
- ⑥ Прапор **Ok** встановлюється на On після того, як ПЛК отримує дані зв'язку, надіслані назад від DMV. На даний момент вже отримані дані можуть бути оброблені.
- ⑦ Прапорці **Start1** і **Ok** можна скинути на Off вручну.
- ⑧ Якщо для виконання виявлення потрібно повторно запустити, безпосередньо встановіть для **Start1** значення On, щоб розпочати наступне виявлення DMV. (Поверніться до кроку 2, щоб продовжити.)

Приклад 2

Mode = 1

Set1 і set2 використовуються для запуску виявлення DMV окремо. Але команду читання даних зв'язку не можна надіслати, доки обидва параметри **Ready** не зміняться на On. (Приклад застосовний до виявлення DMV1000 із двома камерами.)

Дивіться схему контролю послідовності нижче.



- ① Інструкція DMVSH включена.
- ② Встановіть для параметра **Start1** значення On, щоб повідомити про вказівку надсилання вихідного сигналу **Shoot1** (який увімкнено протягом приблизно 10 мс) і повідомити DMV, щоб увімкнути функцію виявлення першої камери.
- ③ Після того, як DMV отримає тригерне повідомлення, для параметра **Ready** змінено значення On (Увімкнено) на Off (Вимкнено).
- ④ ПЛК автоматично вимикає **Shoot1** на Off.
- ⑤ Встановіть для **Start2** значення On, щоб повідомити про вказівку надсилання вихідного сигналу **Shoot2** (який увімкнено протягом приблизно 10 мс) і повідомити DMV, щоб увімкнути функцію виявлення другої камери.
- ⑥ ПЛК автоматично вимикає **Shoot2**.
- ⑦ Після завершення виявлення DMV для параметра **Ready** встановлено значення Off (Вимк.) на On (Увімк.), і обидва виявлення завершено. Потім ПЛК надсилає команду читання через Modbus 0x03.
- ⑧ Прапор **Ok** встановлюється на On після того, як ПЛК отримує дані зв'язку, надіслані назад від DMV. На даний момент вже отримані дані можуть бути оброблені.
- ⑨ Прапорці **Start1**, **Start2** і **Ok** можна скинути на Off вручну. Якщо для виконання виявлення потрібно повторно запустити, поверніться до кроку 2, щоб продовжити.

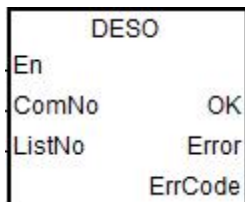
API	Інструкція			Операнд						опис							
1821		DESO		ComNo ~ ErrCode						Виконайте призначений номер зв'язку таблиці обміну даними один раз							

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
ComNo								●					○	○		
IListNo								●					○	○		
OK		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
ComNo		●				●							
IListNo		●				●							
OK	●												
Error	●												
ErrCode		●				●							

СИМВОЛ

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-



- ComNo** : Комунікаційний порт для обміну даними
- IListNo** : Номер зв'язку таблиці обміну даними
- OK** : Прапор завершення
- Error** : Прапор помилки
- ErrCode** : Код помилки

Пояснення

- Ця інструкція доступна для ПЛК із FW V1.08.20 або пізнішої версії та ISPSOft V3.09.07 або пізнішої версії.
- Перед виконанням цієї інструкції переконайтеся, що таблицю обміну даними ЦП ПЛК уже створено. Якщо ні, прапорець Error перемикається на ON, а ErrCode записується як 1.
- ComNo** — комунікаційний порт для обміну даними.

Комунікаційний порт	COM1	COM2	Ethernet	Картка1	Картка2
Число в десятковому форматі	1	2	3 FW CPU ПЛК V1.12 або новішої версії	11	12

- ListNo** — номер зв'язку таблиці обміну даними.

1	1	D100	<<	D0	1
		D200	>>	D0	1
2	1	D300	<<	D0	1
		D400	>>	D0	1

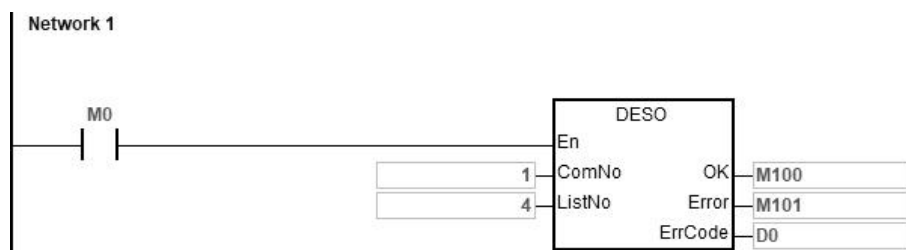
5. Коли інструкція включена, буде виконано лише призначений номер зв'язку таблиці обміну даними. Коли зв'язок завершено і підлеглий пристрій відповідає ОК, прапор ОК буде увімкнено. Відповідний SM номера зв'язку буде увімкнено, а код помилки записаний як 0.
6. Перед виконанням цієї інструкції система перевіряє, чи номер зв'язку, який потрібно виконати, знаходиться в режимі циклічного зв'язку. Якщо так, ця інструкція не буде виконана. Прапор помилки увімкнено. Код помилки записаний як 2.
7. Зверніться до наведеної нижче таблиці для отримання додаткової інформації, якщо будь-який із прапорів помилки увімкнено.

Код помилки	опис
0	Немає помилок або передача завершена.
1	Таблиці обміну даними або номера зв'язку НЕ існує.
2	Розпочато підключення таблиці обміну даними; НЕ запускайте інше підключення знову.
3	Жодної відповіді від рабів.
4	Slave відповідає помилкою зв'язку.
5	Використовується той самий комунікаційний порт.
6	Комунікаційний порт НЕ існує або комунікаційна карта (AS-F232/422/485) Не встановлено. .
7	Номер зв'язку таблиці обміну даними перевищує встановлений діапазон (1~32).

8. Час очікування зв'язку встановлюється в таблиці обміну даними.
9. Ця інструкція пропонується для додатків, де ведений пристрій краще не отримувати або регулярно надсилати із запитом на зв'язок. Оскільки під час виконання інструкції буде відкрито лише один зв'язок, вам потрібно закрити інструкцію, щоб почати інший зв'язок, якщо використовується той самий порт комунікатора.
10. Немає обмежень щодо кількості виконання інструкції, але для кожного порту зв'язку можна встановити лише один зв'язок. Якщо ви знову виконаєте інструкцію, коли використовується той самий комунікаційний порт, прапорець помилки буде ON, а код помилки буде записано як 5.

приклад:

Коли M0 увімкнено, інструкція починає виконувати передачу номер чотири таблиці обміну даними в COM1.



Таблиця обміну даними COM1

	Enable	Remote Station Address	Local Address	Direction	Remote Address	Quantity
1	☐	1	D400	←	D0	10
			D0	→	D0	10
2	☐	1	D410	←	D10	10
			D10	→	D10	10
3	☐	1	D420	←	D20	10
			D20	→	D20	10
4	☒	1	D430	←	D30	10
			D30	→	D30	10
5	☐	1	D440	←	D40	10
			D40	→	D40	10

6.19.3 Описи пов'язаних із зв'язком прапорів і реєстрів

Прапори зв'язку (SM)

Прапор				опис	Дія
COM1	COM2	Картка 1	Картка 2		
SM96	SM97	SM76	SM77	Прапор запиту надсилання даних Якщо ви хочете використовувати інструкцію для надсилання та отримання даних, ви повинні використовувати інструкцію імпульсу, щоб встановити прапорець у положення ON. Коли інструкція виконується, ПЛК надсилає та отримує дані. Після надсилання даних система автоматично скидає прапор на OFF	Ви встановлюєте прапорець на ON, і система автоматично скидає його на OFF.
SM98	SM99	SM78	SM79	Коли прапорець увімкнено, ПЛК очікує на отримання даних.	Система автоматично встановлює прапор на ON і скидає його на OFF.
SM100	SM101	SM80	SM81	Прапор повного прийому Після отримання даних система автоматично встановлює прапор на ON. Коли прапорець увімкнено, отримані дані можуть бути оброблені. Після обробки отриманих даних необхідно скинути прапорці на OFF	Система автоматично встановлює прапорець на ON, а ви скидаєте його на OFF.
SM102	SM103	SM82	SM83	Прапор помилки отримання даних Під час отримання даних за допомогою інструкції отримання даних виникає помилка.	Система автоматично встановлює прапорець на ON, а ви скидаєте його на OFF.
SM104	SM105	SM84	SM85	Прапорець помилки тайм-ауту зв'язку Якщо ви встановлюєте тайм-аут зв'язку (у SR) і дані не надходять після періоду тайм-ауту, прапорець ON. Після вирішення проблеми потрібно скинути прапорець на OFF	Система автоматично встановлює прапорець на ON, а ви скидаєте його на OFF.
SM106	SM107	SM86	SM87	Вибір між 8-бітним режимом обробки і 16-бітний режим обробки ON: 8-бітний режим обробки OFF: 16-бітний режим обробки	Ви встановлюєте прапорець на ON і скидаєте його на OFF.

Прапор				опис	Дія
COM1	COM2	Картка 1	Картка 2		
SM210	SM212	-	-	Режим зв'язку ON: режим RTU OFF: режим ASCII Ви можете встановити це значення в HWCONFIG в ISPSOft. Недоступно для карток 1 і карток 2.	Ви встановлюєте прапорець на ON і скидаєте його на OFF.
SM209	SM211	SM90	SM91	Прапорець зміни протоколу зв'язку Протокол зв'язку змінюється відповідно до значень параметрів у SR. Якщо прапор встановлено на ON, протокол зв'язку змінюється відповідно до значень налаштувань у SR, а потім система автоматично скидає прапор на OFF. ПРИМІТКА: ця зміна не впливає на параметри, встановлені в HWCONFIG. Коли ПЛК знову вмикається, він працює відповідно до протоколу зв'язку, встановленого в HWCONFIG.	Ви встановлюєте прапорець на ON і скидаєте його на OFF.

ПРИМІТКА: наведені вище прапорці не фіксуються.

Комунікаційні реєстри (SR)

Реєстр спеціальних даних				опис
COM1	COM2	Картка 1	Картка 2	
SR201	SR202	SR176	SR178	Адреса комунікаційного порту
SR209	SR212	SR177	SR179	Протокол зв'язку Для отримання додаткової інформації зверніться до наступної таблиці, щоб налаштувати формат зв'язку для послідовного порту зв'язку.
SR210	SR213	SR182	SR183	Тайм-аут зв'язку, одиниці: мс Припустімо, що значення параметра більше 0. Коли ПЛК виконує інструкцію зв'язку та перебуває в стані отримання, але дані не отримані після періоду очікування або проміжний час між двома символами перевищує значення налаштування, прапор тайм-ауту ввімкнено. Ви можете встановити реєстр на 0, щоб вимкнути моніторинг часу очікування зв'язку. Перегляньте набір інструкцій MODRW між 100–32767 (мс).

Встановлення формату зв'язку для послідовного порту зв'язку

b0	Довжина даних			7 (значення=0)	8 (значення=1)
b2-b1	Біти парності			00	: Жодного
				01	: Непарний
				10	: Навіть
b3	стоп-біти			1 біт (значення=0)	2 біти (значення=1)
b7-b4	0001	(16#1)	:	4800	
	0010	(16#2)	:	9600	
	0011	(16#3)	:	19200	
	0100	(16#4)	:	38400	
	0101	(16#5)	:	57600	
	0110	(16#6)	:	115200	
	0111	(16#7)	:	230400	Недоступно для RS-232
	1000	(16#8)	:	500000	Недоступно для RS-232
	1001	(16#9)	:	921000	Недоступно для RS-232
	1111	(16#F)	:	Самовизначений*1	
b8-b15	Не визначено (зарезервовано)				

*1: Ви можете встановити швидкість передачі в HWCONFIG в ISPSOft. Швидкість

передачі даних вказана в наступній таблиці.

Швидкість передачі (біт/с)	Таймер очікування RTU (мс)	Швидкість передачі (біт/с)	Таймер очікування RTU (мс)
4800	9	115200	1
9600	5	230400	1
19200	3	-	-
38400	2	-	-
57600	1	-	-

6.20 Інші інструкції

6.20.1 Список інших інструкцій

У наведеній нижче таблиці наведено інші інструкції, описані в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16- бітний	32- розрядний		
<u>1900</u>	WDT	–	✓	Сторожовий таймер
<u>1901</u>	DELAY	–	✓	Затримка виконання програми
<u>1902</u>	GPWM	–	–	Загальна широтно-імпульсна модуляція
<u>1904</u>	EPUSH	–	✓	Зберігання вмісту індексних регістрів
<u>1905</u>	EPOP	–	✓	Зчитування даних в індексні регістри
<u>1906</u>	INFO	–	✓	Читання системних даних

6.20.2 Пояснення інших інструкцій

API	Код інструкції			Операнд	функція
1900		WDT	P	—	Сторожовий таймер

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	—

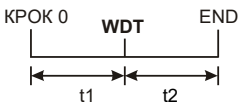
СИМВОЛ



Пояснення

1. Ця інструкція скидає сторожовий таймер для моніторингу роботи системи ПЛК серії AS.
2. Якщо час сканування програми в ПЛК перевищує 200 мілісекунд, світиться світлодіодний індикатор помилки, і ПЛК припиняє роботу.
3. Сторожовий таймер спрацьовує в наступних випадках.
 - Система поводить ся ненормально.
 - Виконання програми займає надто багато часу, тому час сканування довший, ніж таймер сторожового таймера налаштування значення. Ви можете покращити ситуацію двома способами.

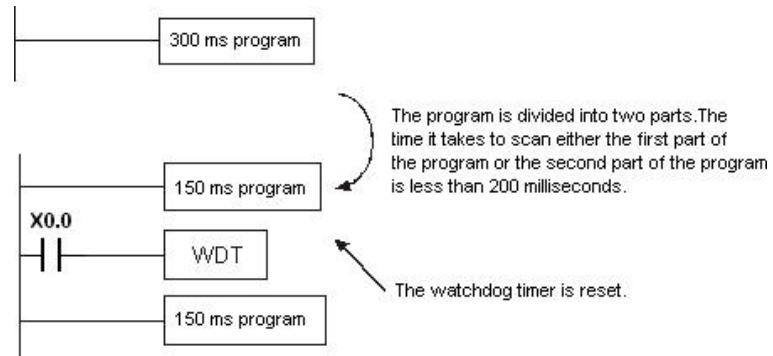
(а) Використовуйте інструкцію WDT, щоб скинути сторожовий таймер.



(а) Зверніться до посібника користувача ISPSoft для отримання додаткової інформації про зміну значення налаштування сторожового таймера.

приклад

Припустимо, час сканування програми становить 300 мілісекунд. Після того, як програму розділено на дві частини, і ви вставите інструкцію WDT між цими двома частинами, час, необхідний для сканування першої чи другої частини програми, має становити менше 200 мілісекунд.



API	Код інструкції				Операнд						функція						
1901		DELAY	P		S						Затримка виконання програми						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●	●		○	○	○	○			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S		●				●											

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	—

СИМВОЛ

DELAY

En

S

DELAYP

En

S

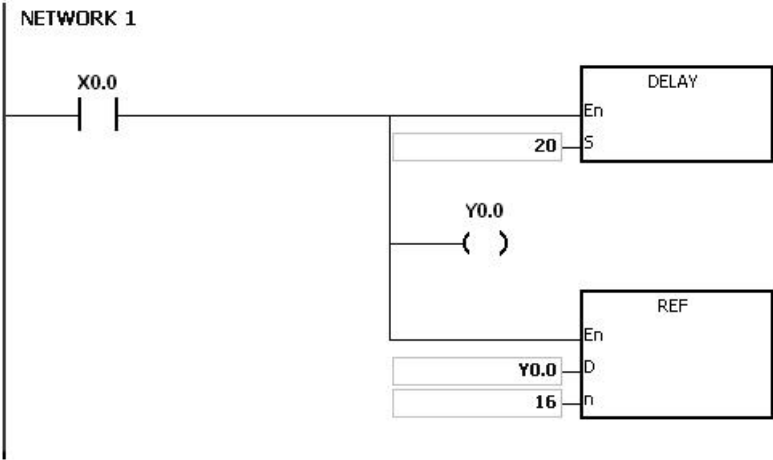
S : Затримка

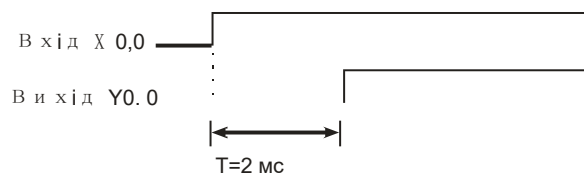
Пояснення

Ця інструкція затримує виконання програми після інструкції DELAY на період часу, вказаний у **S** . Одиницею **S** є 0,1 мілісекунди. (від 1 до 1000)

приклад

Коли X0.0 увімкнено, інструкція затримує виконання програми після інструкції DELAY на два мілісекунд. Тобто Y0.0 увімкнено, а стани Y0.0–Y0.15 оновлюються через дві мілісекунди після виконання інструкції DELAY.

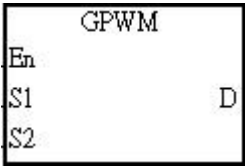


**Додаткові зауваження**

1. Якщо **S** ≤ 0 , затримки немає.
2. Якщо **S** більше 1000, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Ви можете налаштувати затримку відповідно до фактичних умов вашого обладнання.
4. Затримка може бути більшою, ніж **S** через зв'язок або інші впливи.

API	Код інструкції				Операнд							функція					
1902		GPWM			S_1, S_2, D							Загальна широтно-імпульсна модуляція					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●		●			O	O					
S_2		●			●	●		●									
D		●	●	●				●									
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1		●				●											
S_2		●				●											
D	●																
Інструкція пульсу								16-бітна інструкція				32-розрядна інструкція					
—								AS				—					

СИМВОЛ



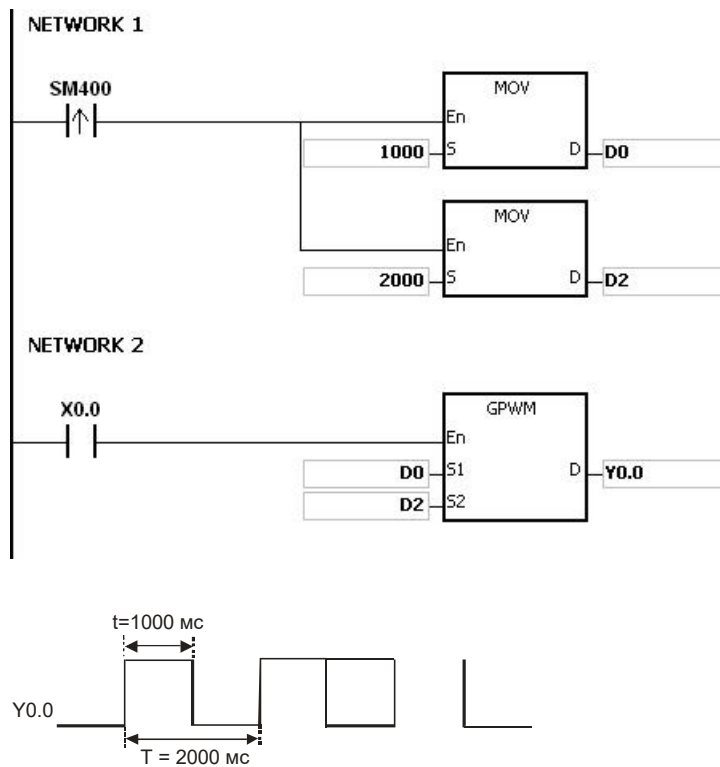
- S_1 : Ширина імпульсу
- S_2 : Пульсовий цикл
- D : Вивідний пристрій

Пояснення

- із пристрою кожен імпульс із шириною, визначеною S_1 , і циклом, визначеним S_2 вказаний D. _
- Ширина імпульсу, визначена S_1 , дорівнює t. t має бути в межах 0–3276 мілісекунд.
- Імпульсний цикл, визначений S_2 , дорівнює T. T має бути в межах 1–32767 мілісекунд, а S_1 має бути менше S_2 .
- Параметри S_2+1 і S_2+2 призначені для системного використання. Будь ласка, не змінюйте їх.
- Якщо S_1 менше 0, імпульсний вихід відсутній. Якщо S_1 більше за S_2 , вихідний пристрій залишається УВІМКНЕНИМ.
- Ви можете змінити S_1 і S_2 під час виконання цієї інструкції.
- Якщо умовний контакт не активовано, імпульсний вихід відсутній.
- Під час редагування в режимі онлайн скиньте умовний контакт, щоб ініціалізувати інструкцію.

приклад

Коли програма виконується, значення в D0 і D2 становлять 1000 і 2000 відповідно. Коли X0.0 увімкнено, пристрій Y0.0 видає імпульси, зображені нижче. Коли X0.0 вимкнено, Y0.0 вимкнено.



Додаткові зауваження

1. Інструкція розраховується за циклом сканування. Тому максимальна помилка становить один цикл сканування. Крім того, S_1, S_2

($S_2 - S_1$) має бути більше циклу сканування; інакше під час виконання інструкції GPWM виникає помилка.

2. Якщо ви використовуєте інструкцію у функціональному блоці або завданні переривання, виникає неточний імпульсний вихід.

3. Якщо ви оголошуєте операнд S_2 в ISPSOft, типом даних буде ARRAY [3] WORD/INT.

API	Код інструкції			Операнд							функція						
1904		EPUSH	P	D							Зберігання вмісту індексних регістрів						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
D								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	—

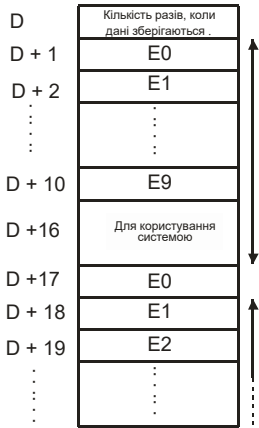
СИМВОЛ



D : Пристрій для зберігання значення в індексних регістрах

Пояснення

- Ця інструкція зберігає значення в E0–E9 у пристроях, визначених D . Допустимий діапазон для D становить 0–99. Інструкція не виконується для значень, які перевищують діапазон.
- Інструкція використовує шістнадцять пристроїв, і останні шість пристроїв призначені для системного використання. Якщо виконати інструкцію і кількість разів збереження даних дорівнює n (значення в D), інструкція зберігає дані в E0–E9 у D +(16*n+1)– D +(16*n+16) і збільшує значення в D до n+1.
- Пам'ять для пристрою D становить 1601 (16x100+1).
- Ця інструкція використовує імпульсну інструкцію для взаємодії зі стеком, надаючи значення в стек. Тому необхідно скинути контакт перед наступною операцією.
- Коли ви використовуєте цю інструкцію з інструкцією EPOP (API 1905), значення, що зберігається останнім у пристрої, визначеному D читається першим, дотримуючись принципу LIFO (останній прийшов, перший вийшов).



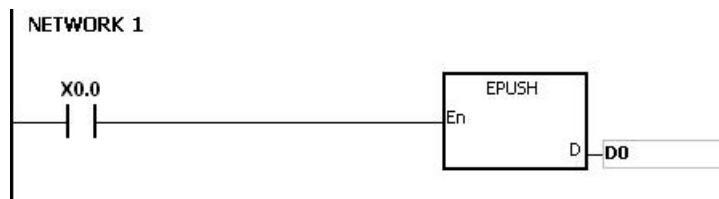
приклад

Припустимо, що значення в D0 дорівнює 0.

Коли X0.0 увімкнено вперше, інструкція передає дані з E0–E9 на D1–D10 і збільшує значення в D0 до 1.

Коли X0.0 перемикається з OFF на ON вдруге, інструкція передає дані з E0–E9 на D17–D26 і збільшує значення в D0 до 2.

Коли X0.0 перемикається з OFF на ON в n -й раз, інструкція передає дані в E0–E9 до $(n*16)+1-(n*16)+10$.

**Додаткові зауваження**

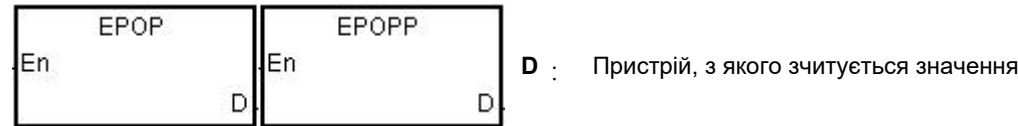
1. Якщо значення в **D** не між 0–99, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Якщо діапазон пристроїв недостатній для $\mathbf{D} + ((\text{значення в } \mathbf{D}) + 1) * 16 - 1$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції			Операнд							функція						
1905		EPOP	P	D							Зчитування даних в індексні регістри						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
D								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
D		●				●							

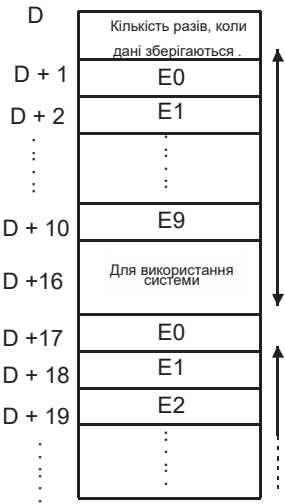
Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	—

СИМВОЛ



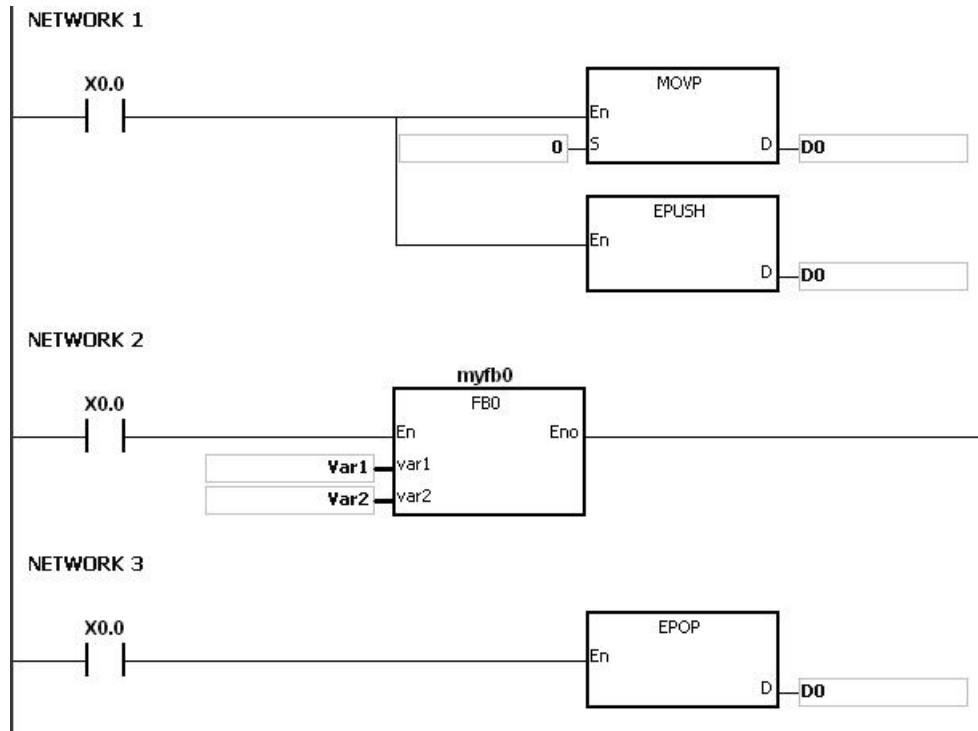
Пояснення

1. Ця інструкція зчитує значення в пристроях, визначених **D** , у E0–E9 і зменшує значення в **D** на одиницю.
Дійсне значення для **D** становить від 1 до 100. Інструкція не виконує значення, які перевищують діапазон.
2. Інструкція включає шістнадцять пристроїв, і останні шість пристроїв для системного використання. Якщо ви виконуєте інструкцію
а кількість разів збереження даних дорівнює n (значення в **D**) , інструкція зберігає дані в E0–E9 у
 $D + 16 * (n - 1) + 1 - D + 16 * (n - 1) + 10$, і зменшує значення в **D** до n-1.
3. Ця інструкція використовує імпульсну інструкцію для взаємодії зі стеком, беручи ПЕРШЕ ЗНАЧЕННЯ зі стеку та присвоєння його вказаній змінній; тому ви повинні скинути контакт перед наступною операцією.
4. Значення, яке інструкція зберігає останнім у пристрої, визначеному **D** , зчитується першим, дотримуючись принципу LIFO (останній прийшов, перший вийшов).



приклад

Коли X0.0 увімкнено, інструкція MOV встановлює значення в D0 на 0, а інструкція EPUSH передає значення з E0–E9 до D1–D10. Після завершення виконання FB0 інструкція EPOP зчитує значення з D1–D9 у E0–E9.

**Додаткові зауваження**

1. Якщо значення в **D** не в діапазоні 0–100, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 16 № 2003.
2. Якщо діапазон пристрою недостатній для $D + ((\text{значення в } D) * 16 - 1)$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

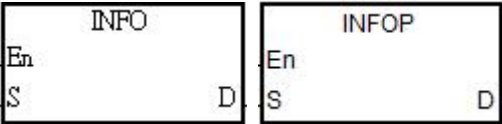
API	Інструкція				Операнд							опис						
1906		INFO	P		S , D							Читання системних даних						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S													O					
D								●										

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	—

СИМВОЛ

S : Код читання системних даних



D : Пусковий пристрій, де зберігаються зчитані системні дані

Пояснення

1. S – код зчитування системної інформації, коди наведено в наступній таблиці.

Код S	опис	Число послідовний D пристроїв (Кількість слів)	Зауваження	Підтримувана версія
0	Читає серійний номер продукту	9	код ASCII	V1.04 і новіші версії
1	Читає Ethernet Mac ID	3	Шістнадцяткове значення	V1.04 і новіші версії
2	Читає вже живлення час	2	32-розрядне значення, одиниця вимірювання: хвилин	V1.04 і новіші версії
інші	Зарезервований	0	Інструкція ні виконано.	

2. Якщо серійний номер продукту. – AS332T-AW16360001, код читання інструкції – 0, а зчитані дані зберігаються в пристроях, починаючи з D0, серійний номер продукту. зберігається в такому порядку.

D0	D1	D2	D3	D4	D5	D6	D7	D8
16#5341	16#3333	16#5432	16#412D	16#3157	16#3336	16#3036	16#3030	16#0031
"SA"	'33'	"T2"	"A-"	"1W"	'36'	'06'	'00'	'1'

3. Якщо Mac ID 16#123456789ABC, код читання інструкції дорівнює 1, а зчитані дані зберігаються на пристроях починаючи з D10, дані Mac ID зберігаються в такому порядку.

D10	D11	D12
16#1234	16#5678	16#9ABC

4. Якщо час увімкнення живлення становить 70000 хвилин (16#00011170), код читання інструкції дорівнює 2, а зчитування

дані зберігаються в пристроях, починаючи з D20, дані про час увімкнення живлення зберігаються в такому порядку.

D20	D21
16#1170	16#0001

5. Тільки перше виконання інструкції INFO є ефективним, коли умовний контакт зустрічається. Тому так і є рекомендується використовувати імпульсну інструкцію INFOP. Якщо інструкція INFO використовується разом із нормально відкритим контактом, інструкція буде автоматично виконана лише один раз.

Додаткове зауваження:

Якщо **D** оголошено в змінній і код 0, 1 і 2, вони можуть бути оголошені відповідно в ARRAY [9], [3], [2] WORD.

6.21 Інструкції з обробки рядка

6.21.1 Список інструкцій з обробки рядків

У наведеній нижче таблиці наведено інструкції з обробки рядків, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
<u>2100</u>	BINDA	DBINDA	✓	Перетворення десяткового числа зі знаком у код ASCII
<u>2101</u>	BINHA	DBINHA	✓	Перетворення двійкового шістнадцяткового числа в шістнадцятковий код ASCII
<u>2102</u>	BCDDA	DBCDDA	✓	Перетворення двійкового десяткового числа в ASCII код
<u>2103</u>	DABIN	DDABIN	✓	Перетворення десяткового коду ASCII зі знаком у знаковий десяткове двійкове число
<u>2104</u>	HABIN	DABIN	✓	Перетворення шістнадцяткового коду ASCII у шістнадцятковий двійкове число
<u>2105</u>	DABCD	DDABCD	✓	Перетворення коду ASCII у двійковий десятковий код номер
<u>2106</u>	\$LEN	—	✓	Обчислення довжини рядка
<u>2109</u>	\$FSTR	—	✓	Перетворення числа з плаваючою комою в рядок
<u>2110</u>	\$FVAL	—	✓	Перетворення рядка в число з плаваючою комою
<u>2111</u>	\$RIGHT	—	✓	Отримання символів із рядка, починаючи з правого боку
<u>2112</u>	\$LEFT	—	✓	Отримання символів із рядка, починаючи зліва
<u>2113</u>	\$MIDR	—	✓	Отримання частини рядка
<u>2115</u>	\$SER	—	✓	Пошук рядка
<u>2116</u>	\$RPLC	—	✓	Заміна символів у рядку
<u>2117</u>	\$DEL	—	✓	Видалення символів у рядку
<u>2118</u>	\$CLR	—	✓	Очищення рядка
<u>2119</u>	\$INS	—	✓	Вставка рядка
<u>2122</u>	SPLIT	—	✓	Розбиття струни
<u>2123</u>	MERGE	—	✓	Об'єднання рядків

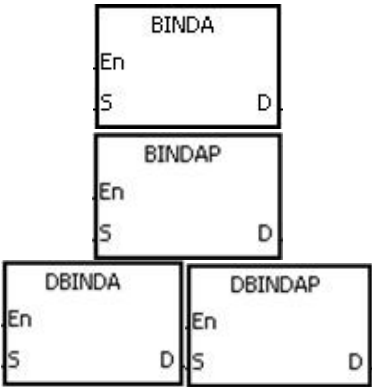
6.21.2 Пояснення інструкцій з обробки рядків

API	Код інструкції				Операнд								функція			
2100	D	BINDA		P	S, D								Перетворення десяткового числа зі знаком у код ASCII			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●	●	●	●		O	O	O	O		
D		●			●	●		●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

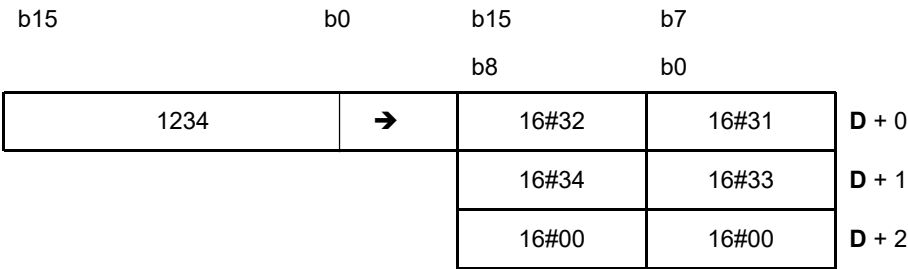
СИМВОЛ



- S** : Вихідне значення
- D** : Пристрій, на якому зберігається результат перетворення

Пояснення

- Ця інструкція перетворює десяткове двійкове число зі знаком у **S** на код ASCII і зберігає результат перетворення в **D**.
- Інструкція підтримує SM690 для керування кінцевим символом.
- Для 16-розрядних інструкцій значення в **S** має бути двійковим числом у діапазоні від -32768 до 32767. Операнд **D** займає чотири апарати слів. Інструкція перетворює дані перетворюються наступним чином.



Якщо SM690 вимкнено, інструкція зберігає кінцевий символ 16#0000 у **D** +2. Якщо SM690 увімкнено, значення в **D** +2 є без змін без символу закінчення.

Якщо значення в **S** є додатним значенням, інструкція зберігає лише значення, але не символ знака в **D**.

Якщо значення в **S** є від'ємним значенням, інструкція зберігає знак «-» в **D** (16#2D). Наприклад, якщо значення в **S** становить - 12345, а SM690 вимкнено, результат перетворення буде таким.

b15	b0	b15	b7	
		b8	b0	
-1234	→	16#31	16#2D (-)	D + 0
		16#33	16#32	D + 1
		16#00	16#34	D + 2

4. Для 32-розрядних інструкцій значення в **S** має бути двійковим числом від -2147483648 до 2147483647.

операнд **D** займає шість пристроїв слів. Дані перетворюються наступним чином.

b31	b0	b15	b7	
		b8	b0	
12345678	→	16#32	16#31	D + 0
		16#34	16#33	D + 1
		16#36	16#35	D + 2
		16#38	16#37	D + 3
		16#00	16#00	D + 4

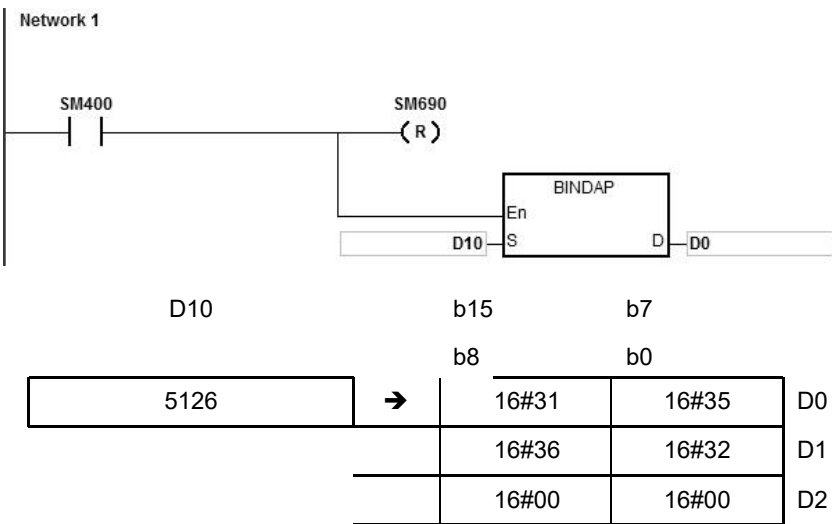
Якщо SM690 вимкнено, інструкція зберігає кінцевий символ 16#0000 у **D** +4. Якщо SM690 увімкнено, значення в **D** +4 є без змін без символу закінчення.

Крім того, якщо значення в **S** є додатним значенням, інструкція зберігає лише значення, але не символ знака в **D**. Якщо значення в **S** є від'ємним значенням, інструкція зберігає знак «-» в **D** (16#2D). Наприклад, якщо значення в **S** становить -12345678, а SM690 вимкнено, результат перетворення буде таким.

b31	b0	b15	b7	
		b8	b0	
-12345678	→	16#31	16#2D	D + 0
		16#33	16#32	D + 1
		16#35	16#34	D + 2
		16#37	16#36	D + 3
		16#00	16#38	D + 4

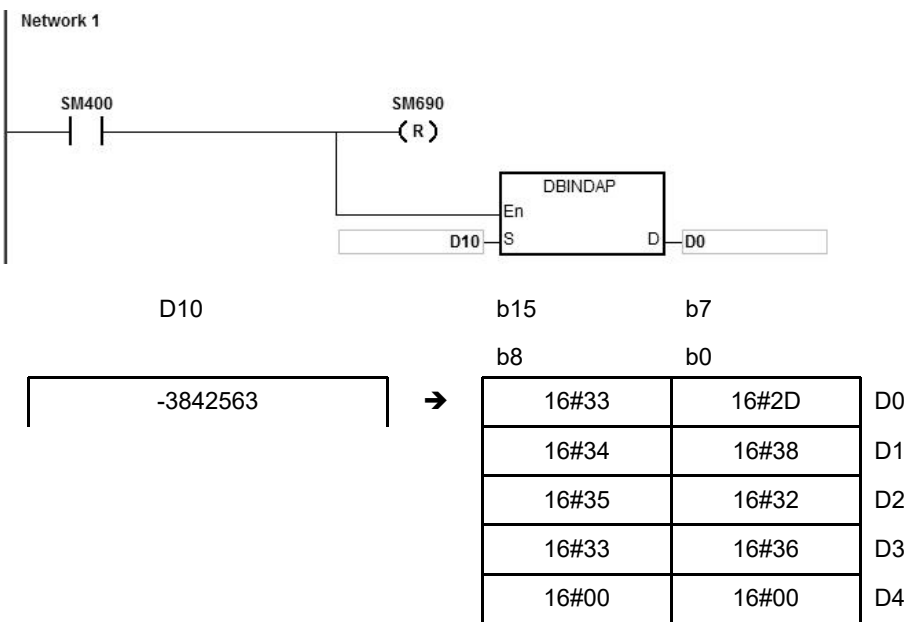
Приклад 1

Припустимо, значення в L0 становить 5126, а SM690 вимкнено. Коли інструкція виконується, значення D0=16#3135, D1=16#3632, D2=16#0000.



Приклад 2

Припустимо, значення в D10 дорівнює -3842563, а SM690 вимкнено. Коли інструкція виконується, значення D0=16#332D, D1=16#3438, D2=16#3532, D3=16#3336, D4=16#0000.



Додаткові зауваження

1. Якщо значення в пристрої **D** недостатньо для перетворення, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Для 16-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSOft, типом даних є ARRAY [4] з WORD/INT.
3. Для 32-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSOft, типом даних буде ARRAY [6] WORD/INT.

API	Код інструкції				Операнд								функція				
2101	D	BINHA		P	S, D								Перетворення двійкового шістнадцяткового числа в шістнадцятковий код ASCII				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O	O	O	O			
D		●			●	●		●									

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ

BINHA		BINHAP	
En		En	
S	D	S	D

DBINHA		DBINHAP	
En		En	
S	D	S	D

S : Вихідне значення

D : Пристрій, на якому зберігається результат перетворення

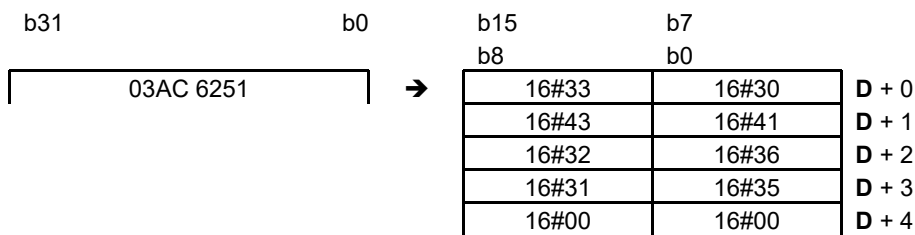
6

Пояснення

1. Ця інструкція перетворює шістнадцяткове двійкове число в S на код ASCII і зберігає результат перетворення в D .
2. Інструкція підтримує SM690 для керування кінцевим символом.
3. Для 16-розрядних інструкцій значення в S має бути між 16#0000–16#FFFF і має бути чотиризначним двійковим кодом номер. Операнд D займає три слова.
- Якщо SM690 вимкнено, 16#0000 зберігається в D +2. Якщо SM690 увімкнено, значення в D +2 не змінюється.
- Наприклад, якщо значення в S дорівнює 16#02A6, а SM690 вимкнено, результат перетворення буде таким.

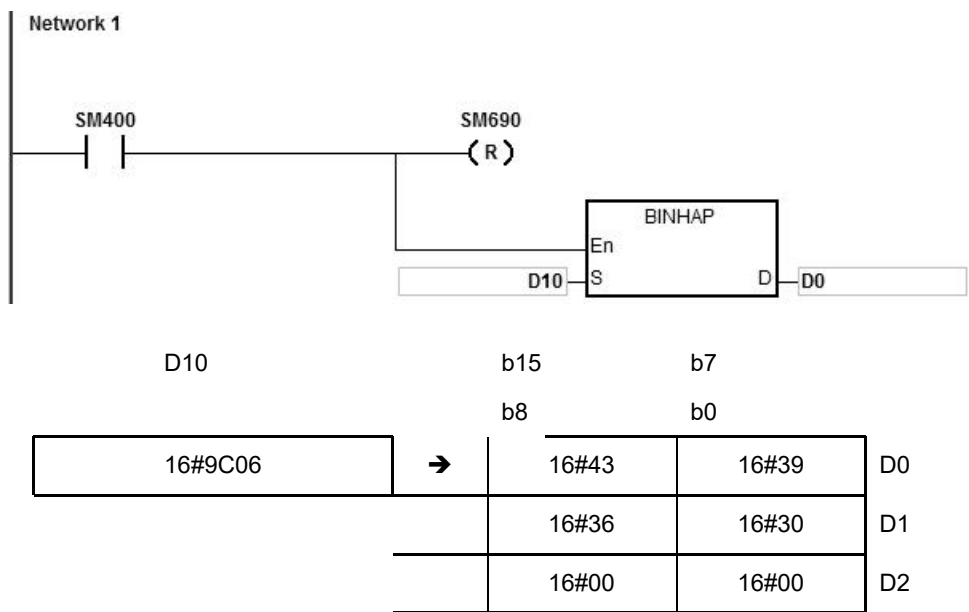
b15	b0	b15	b7	
		b8	b0	
02A6		→	16#32	16#30
			16#36	16#41
			16#00	16#00
				D + 0
				D + 1
				D + 2

4. Для 32-розрядних інструкцій значення в S має бути між 16#00000000–16#FFFFFFFF і бути восьмизначним двійкове число. Операнд D займає п'ять пристроїв слів.
- Якщо SM690 вимкнено, 16#0000 зберігається в D +4. Якщо SM690 увімкнено, значення в D +4 не змінюється.
- Наприклад, якщо значення в S дорівнює 16#03AC625E, а SM690 вимкнено, результат перетворення буде таким.



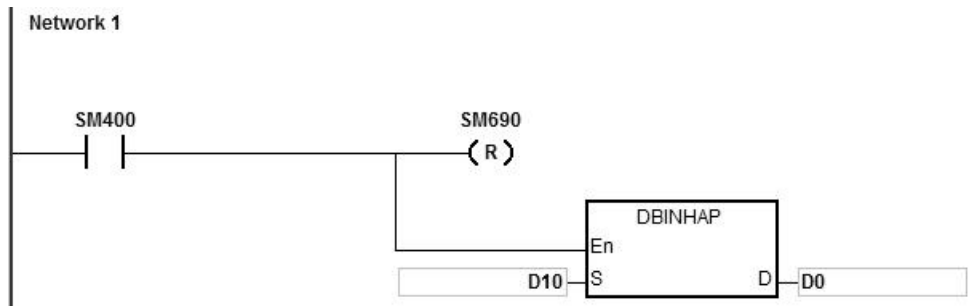
Приклад 1

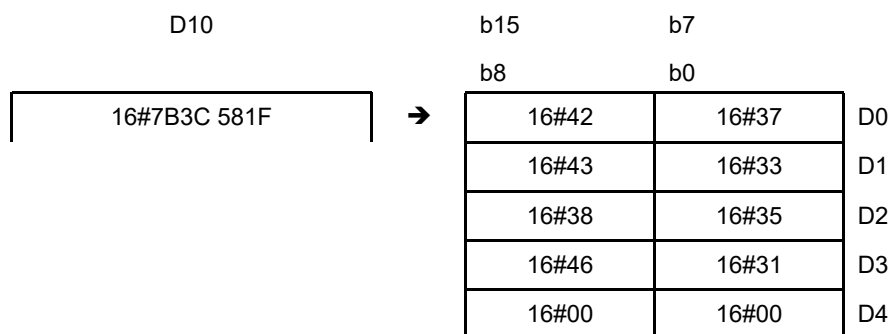
Припустимо, що значення в D10 дорівнює 16#9C06, а SM690 вимкнено. Коли інструкція виконується, значення D0=16#4339, D1=16#3630, D2=16#0000.



Приклад 2

Припустимо, що значення в D10 дорівнює 16#7B3C581F, а SM690 вимкнено. Коли інструкція виконується, значення D0=16#4237, D1=16#4333, D2=16#3835, D3=16#4631, D4=16#0000.





Додаткові зауваження

1. Для 16-бітних інструкцій, якщо **D** +2 перевищує діапазон пристрою, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Для 32-розрядних інструкцій, якщо **D** +4 перевищує діапазон пристрою, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Для 16-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSOft, тип даних буде ARRAY [3] з WORD/INT.
4. Для 32-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSOft, типом даних буде ARRAY [5] WORD/INT.

API	Код інструкції			Операнд						функція							
2102	D	BCDDA	P	S, D						Перетворення двійкового десятичного числа в код ASCII							
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●	●	●	●		O	O	O	O			
D		●			●	●		●									

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●	●			●	●						
D		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
AS		AS		AS	

СИМВОЛ

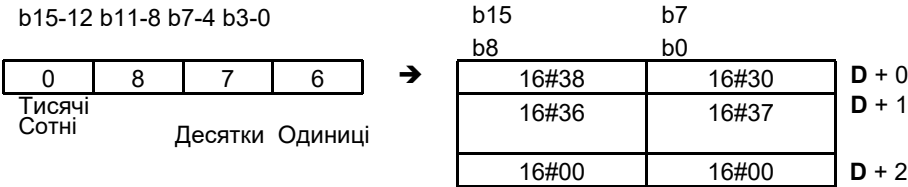
BCDDA		BCDDAP	
En		En	
S	D	S	D

DBCDDA		DBCDDAP	
En		En	
S	D	S	D

- S**: Вихідне значення
- D**: Пристрій, на якому зберігається результат перетворення

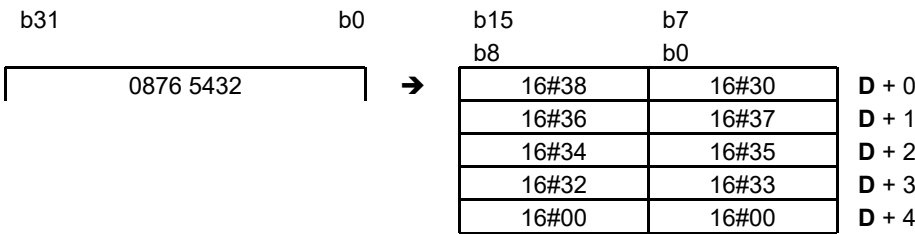
Пояснення

1. Ця інструкція перетворює двійкове десятичне число в **S** на код ASCII і зберігає результат перетворення в **D**.
2. Інструкція підтримує SM690 для керування кінцевим символом.
3. Двійкове десятичне значення в **S**, яке використовується в 16-бітній інструкції, має бути між 0–9999 і має бути чотиризначним двійковим десятичним значенням. Операнд **D** займає три слова. Дані перетворюються наступним чином.



Якщо SM690 вимкнено, інструкція зберігає кінцевий символ 16#0000 у **D** +2. Якщо SM690 увімкнено, значення в **D**+2 не змінюється без символу закінчення.

4. Для 32-розрядних інструкцій двійкове десятичне значення в **S** має бути між 0–99999999 і бути вісімкою. digit двійкове десятичне значення. Операнд **D** займає п'ять пристроїв слів. Дані перетворюються наступним чином.

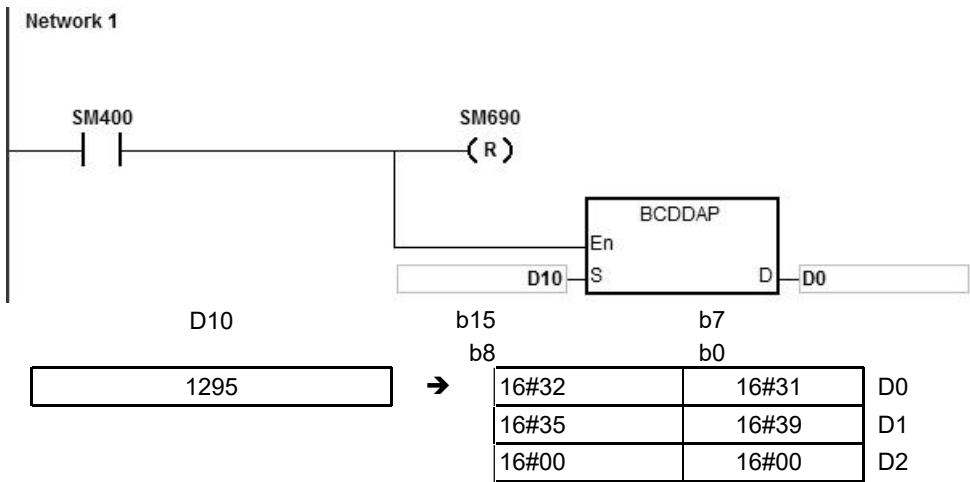


5. Якщо SM690 вимкено, інструкція зберігає кінцевий символ 16#0000 у D +5. Якщо SM690 увімкнено, значення в D +5 є без змін без символу закінчення.

6. Навіть якщо перша цифра двійкового десятикового значення в S дорівнює 0, інструкція перетворює його на код ASCII 0 (16#30).

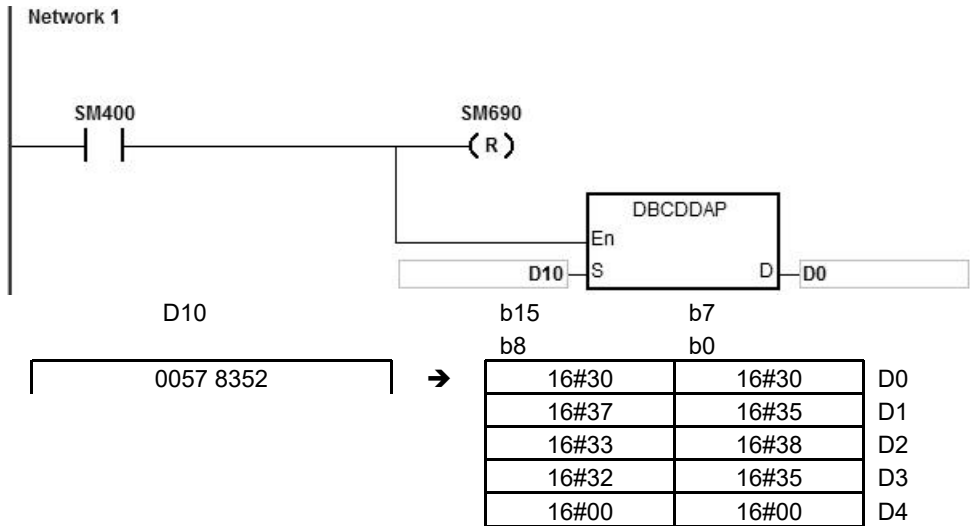
Приклад 1

Припустімо, що двійкове десятикове значення в D10 дорівнює 1295, а SM690 вимкнено. Коли інструкція виконується, значення D0=16#3231, D1=16#3539, D2=16#0000.



Приклад 2

Припустімо, що двійкове десятикове значення в D10 дорівнює 00578352, а SM690 вимкнено. Коли інструкція виконується, значення D0=16#3030, D1=16#3735, D2=16#3338, D3=16#3235, D4=16#0000.



Додаткові зауваження

1. Для 16-бітних інструкцій, якщо значення в **S** не знаходиться в межах 0–9999, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200D. Двійкове десяткове значення представлено шістнадцятковим числом, але одна з цифр не знаходиться в діапазоні від 0 до 9.
2. Для 32-розрядних інструкцій, якщо значення в **S** не між 0–99999999, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200D. Двійкове десяткове значення представлено шістнадцятковим числом, але одна з цифр не знаходиться в діапазоні від 0 до 9.
3. Для 16-бітних інструкцій, якщо **D** +2 перевищує діапазон пристрою, SM0 увімкнено, а код помилки в SR0 – 16#2003.
4. Для 32-розрядних інструкцій, якщо **D** +4 перевищує діапазон пристрою, SM0 увімкнено, а код помилки в SR0 – 16#2003.
5. Для 16-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSoft, тип даних буде ARRAY [3] з WORD/INT.
6. Для 32-розрядних інструкцій, якщо ви оголошуєте операнд **D** в ISPSoft, типом даних буде ARRAY [5] WORD/INT.

API	Код інструкції				Операнд								функція				
2103	D	DABIN	P		S, D								Перетворення десяткового коду ASCII зі знаком у десяткове двійкове число зі знаком				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S		●			●	●		●	●						O		
D		●			●	●	●	●			O	O					
Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S		●				●										●	
D		●	●			●	●										
								Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція			
								AS			AS			AS			

СИМВОЛ

DABIN		DABINP	
En		En	
S	D	S	D
DDABIN		DDABINP	
En		En	
S	D	S	D

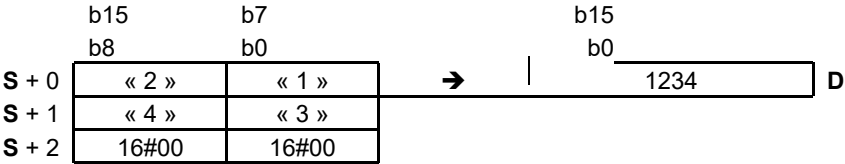
S : Вихідне значення

D : Пристрій, на якому зберігається результат перетворення

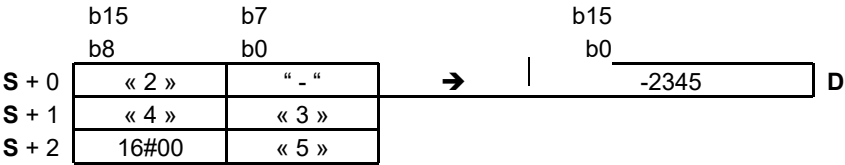
Пояснення

6

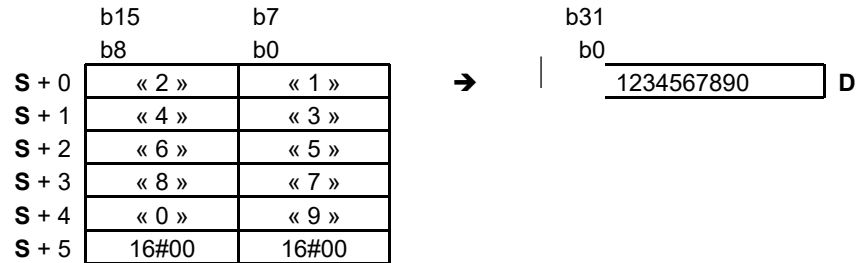
1. Ця інструкція перетворює десятковий ASCII-код зі знаком у S на десяткове двійкове число зі знаком і зберігає результат перетворення в D .
2. Для 16-розрядних інструкцій операнд S займає три слова, а десятковий ASCII-код у S повинен бути від -32768 до 32767. Якщо значення в S є рядком і не містить кінцевого символу 16#00, перетворення може містити до 5 цифр (без урахування знака).



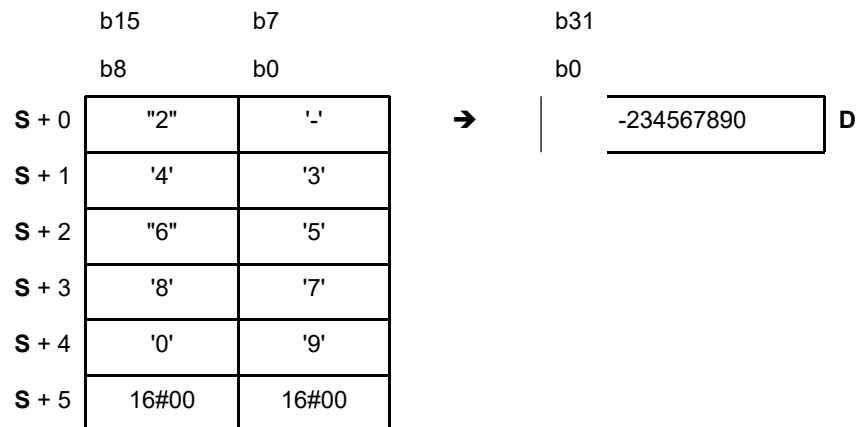
3. Якщо першим символом є « » (пробіл), знак є позитивним. Якщо перший символ «-», це знак мінус. Візьмемо, наприклад, рядок «2345».



4. Для 32-розрядних інструкцій операнд **S** займає шість пристроїв слів, а десятковий код ASCII у **S** повинен бути від -2147483648 до 2147483647. Якщо значення в **S** є рядком і не містить кінцевого символу 16#00, перетворення може містити до 10 цифр (без знака).



5. Якщо перший символ « » (пробіл), знак є додатним. Якщо перший символ «-», це знак мінус. Візьмемо, наприклад, рядок «-234567890».



6. Якщо перша цифра рядка в пристрої **S** пуста (16#20) + знак (16#2B), значення в **S** обробляється як 0. Для друга цифра, якщо число не 0–9, інструкція розглядає його як кінець рядка, і повідомлення про помилку не буде показано. Наприклад, якщо порядок слів 16#20 → 16#31 → 16#32 → 16#2B, результатом перетворення буде 12.

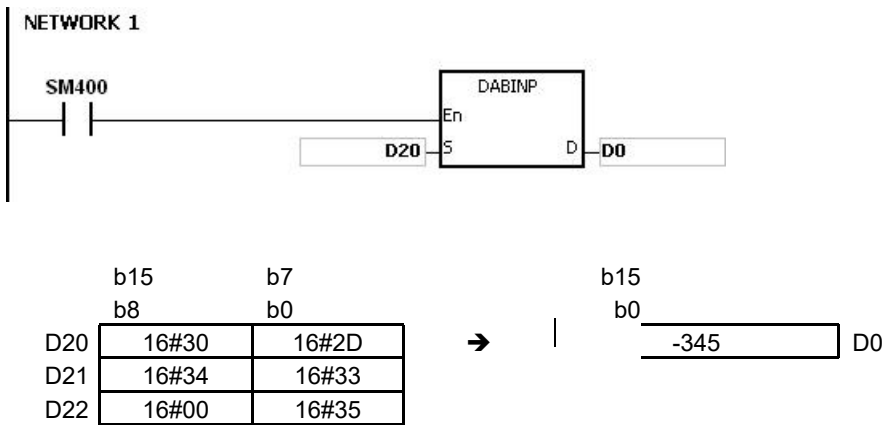
7. Для 16-розрядних інструкцій діапазон рядків у пристрої **S** становить 1–6 (включаючи позитивні або негативні знаки), а для 32-розрядних

діапазон інструкції 1–11 (включаючи позитивні негативні знаки).

8. Тільки 32-розрядні інструкції можуть використовувати 32-розрядний лічильник, але не пристрій E.

Приклад 1

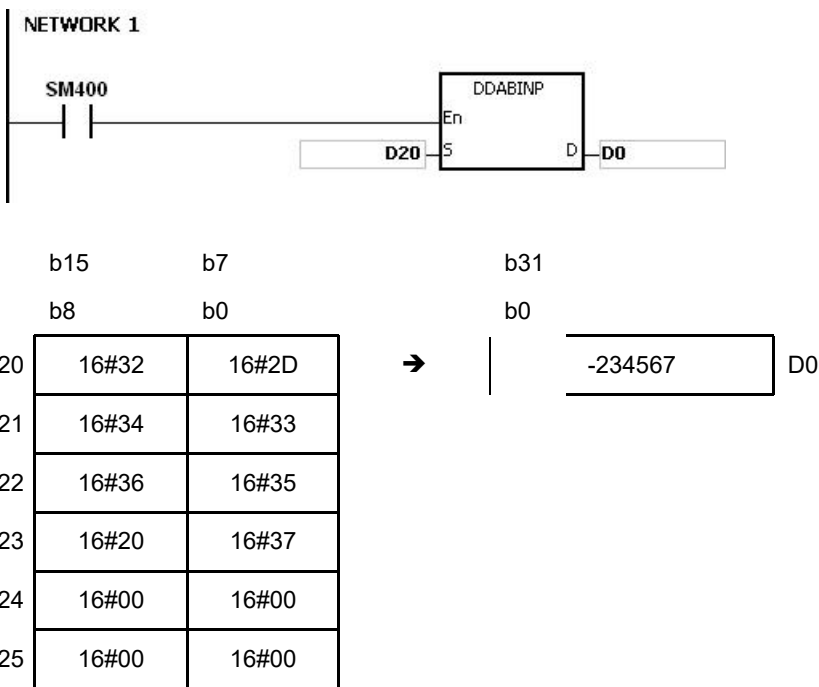
Припустимо, значення D20=16#302D, D21=16#3433, D22=16#0035. Коли інструкція виконується, значення D0=- 345.



Приклад 2

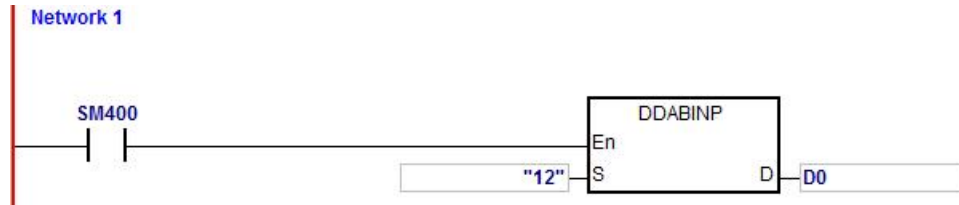
Припустимо, що значення D20=16#322D, D21=16#3433, D22=16#3635, D23=16#2037, D24=16#0000, D25=16#0000.

Коли інструкція виконується, значення D0=-234567.



Приклад 3

Припустимо, значення рядка в **S** дорівнює 12. Коли інструкція виконується, значення D0=12.



Додаткові зауваження

1. Якщо значення першого слова в **S** є символом закінчення (16#00), значення обробляється як 0 (16#30).
2. Якщо значення першої цифри в **S** дорівнює 16#20 (пробіл), 16#2B (+) або 16#2D (-), а другої цифри дорівнює 16#00, значення в **S** обробляється як 0 (16# 30).
3. Навіть якщо перша цифра двійкового десятичного значення в **S** дорівнює 0, воно перетворюється на код ASCII 0 (16#30).
4. Значення першої цифри в **S** підтримує лише коди ASCII, 16#30–16#39 (0–9), 16#200 (пробіл), 16#2D (мінус), 16#2B (позитивний знак), 16 #00 (кінцевий символ). Якщо значення першої цифри в **S** не є кодом ASCII, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
5. За винятком першої цифри, якщо значення інших цифр у **S** не є кодами ASCII, 16#30–16#39 або 16#00, значення в **S** є оброблено як 16#00.
6. Якщо значення в **S** перевищує діапазон пристрою, SM0 увімкнено, а код помилки в SR0 – 16#2003. Інструкція є не виконано.
7. Для 16-розрядних інструкцій, якщо ви оголошуєте операнд **S** в ISPSOft, типом даних буде ARRAY [3] WORD/INT.
8. Для 32-розрядних інструкцій, якщо ви оголошуєте операнд **S** в ISPSOft, типом даних буде ARRAY [6] WORD/INT.

API	Код інструкції			Операнд								функція					
2104	D	HABIN	P	S, D								Перетворення шістнадцяткового коду ASCII у шістнадцяткове двійкове число					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S		●			●	●		●	●						O		
D		●			●	●	●	●			O	O					
Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S		●				●										●	
D		●	●			●	●										
								Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція			
								AS			AS			AS			

СИМВОЛ

HABIN		HABINP	
En		En	
S	D	S	D
DHABIN		DHABINP	
En		En	
S	D	S	D

S : Вихідне значення

D : Пристрій, на якому зберігається результат перетворення

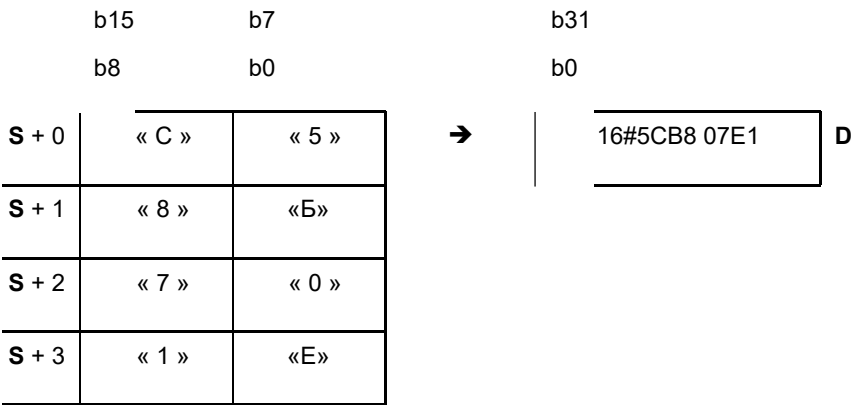
6

Пояснення

1. Ця інструкція перетворює шістнадцятковий код ASCII у **S** на шістнадцяткове двійкове число та зберігає результат перетворення в **D**.
2. Для 16-бітних інструкцій операнд **S** займає два пристрої слова. Якщо значення в **S** є рядком і не містить кінцевого символу 16#00, конвертація може містити до 4 цифр (без знака). Шістнадцятковий код ASCII у **S** має бути між 0000–FFFF. Якщо **S** є рядком, рядок має бути між «0»–«FFFF».

	b15	b7		b15
	b8	b0		b0
S + 0	"A"	« 5 »	➔	16#5A8D
S + 1	"D"	« 8 »		

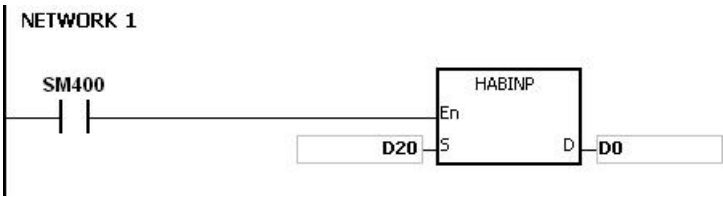
3. Для 32-розрядних команд операнд **S** займає чотири пристрої слів. Якщо значення в **S** є рядком і не включає кінцевий символ 16#00, конвертація може складатися з 8 цифр (без знака). Шістнадцятковий код ASCII в **S** має бути між 00000000–FFFFFFFF. Якщо **S** є рядком, рядок має бути між «0»–«FFFFFFFF».



4. Діапазон допустимих рядків у пристрої **S** для 16-бітної інструкції становить 1–4, а для 32-бітної інструкції – 1–8.

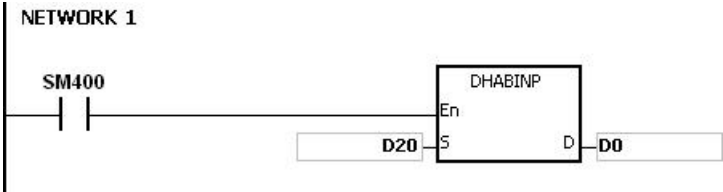
Приклад 1

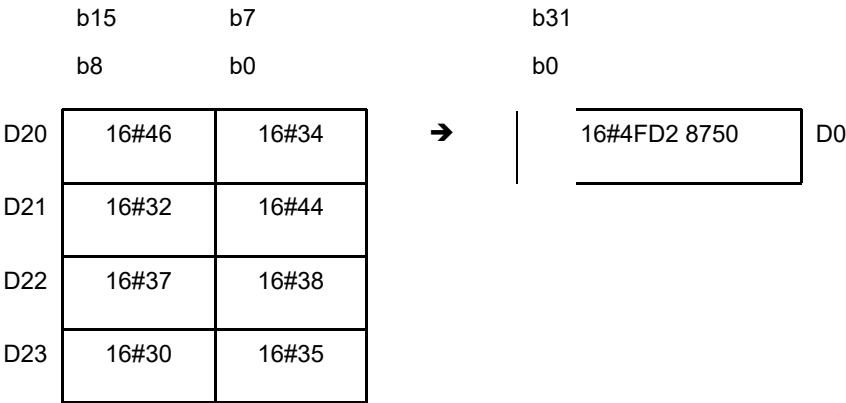
Припустимо, значення D20=16#3641, D21=16#4633 (ASCII 16#A63F). Коли інструкція виконується, значення D0=-22977.



Приклад 2

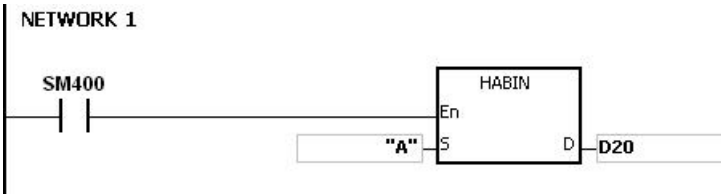
Припустимо, що значення D20=16#4634, D21=16#3244, D22=16#3738, D23=16#3035 (ASCII 16#4FD28750). Коли інструкція виконується, значення (D1, D0)=16#4FD28750.





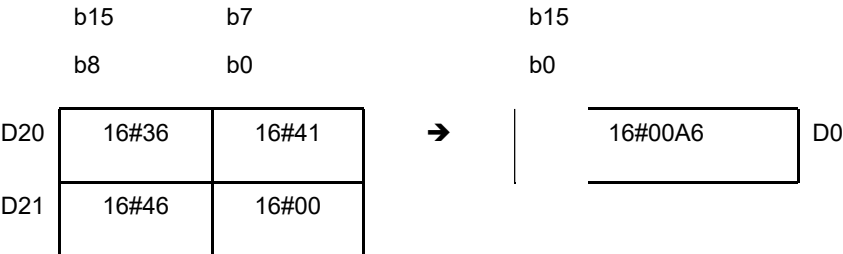
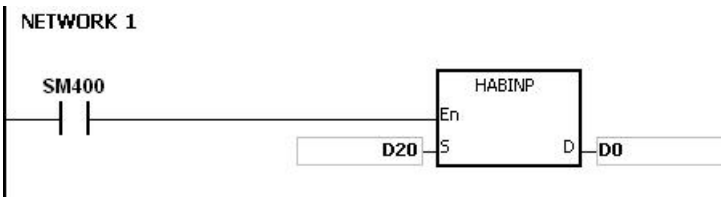
Приклад 3

Припустимо, значення рядка в S дорівнює A. Коли інструкція виконується, значення D20=16#A=10.



Приклад 4

Припустимо, значення D20=16#3641, D21=16#4600 (ASCII 16#00A6). Коли інструкція виконується, значення в D0=166.



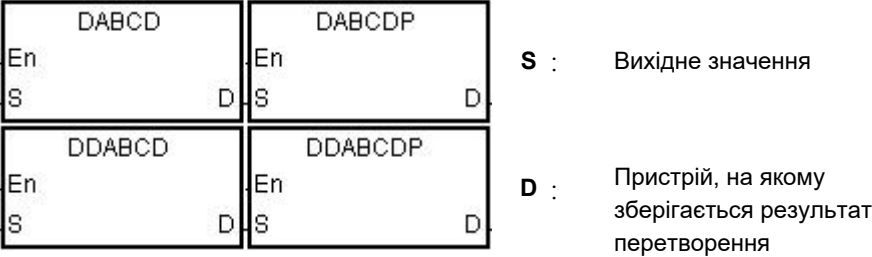
Додаткові зауваження

1. Якщо код ASCII у **S** не між 16#30–16#39 («0»–»9») або між 16#41–16#46 («A»–«F»), інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
2. Для 16-розрядних інструкцій, якщо ви оголошуєте операнд **S** в ISPSOft, типом даних буде ARRAY [2] WORD/INT.
3. Для 32-розрядних інструкцій, якщо ви оголошуєте операнд **S** в ISPSOft, типом даних є ARRAY [4] з WORD/INT.

API	Код інструкції				Операнд						функція						
2105	D	DABCD		P	S, D						Перетворення ASCII-коду в двійкове десятичне число						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S		●			●	●		●	●						O		
D		●			●	●	●	●			O	O					
Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S		●				●										●	
D		●	●			●	●										

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	AS

СИМВОЛ



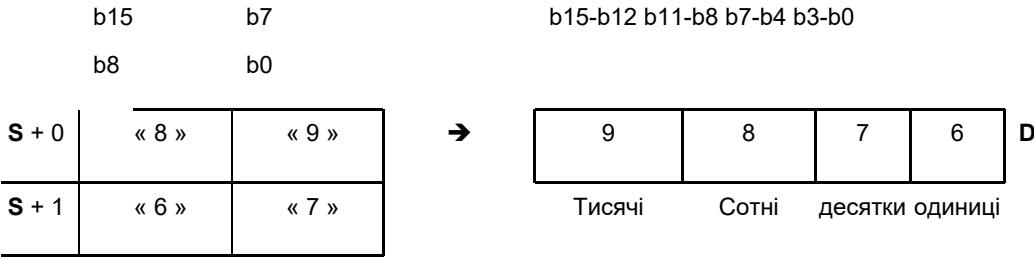
S : Вихідне значення

D : Пристрій, на якому зберігається результат перетворення

Пояснення

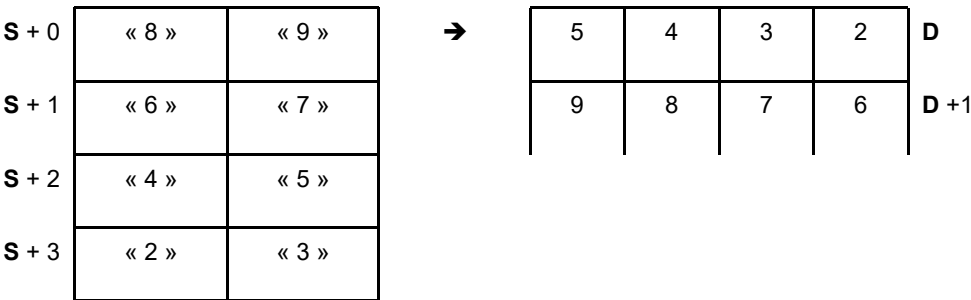
- 6
1. Ця інструкція перетворює код ASCII у **S** на двійкове десятичне число та зберігає результат перетворення в **D**.

2. Для 16-бітних інструкцій операнд **S** займає два пристрої слів, а код ASCII у **S** має бути між 0000–9999. Якщо **S** є рядком, рядок має бути між «0»–«9999».



3. Для 32-розрядних інструкцій операнд **S** займає чотири слова, а код ASCII в **S** повинен бути між 0000000–99999999. Якщо **S** є рядком, рядок має бути між «0»–«99999999».

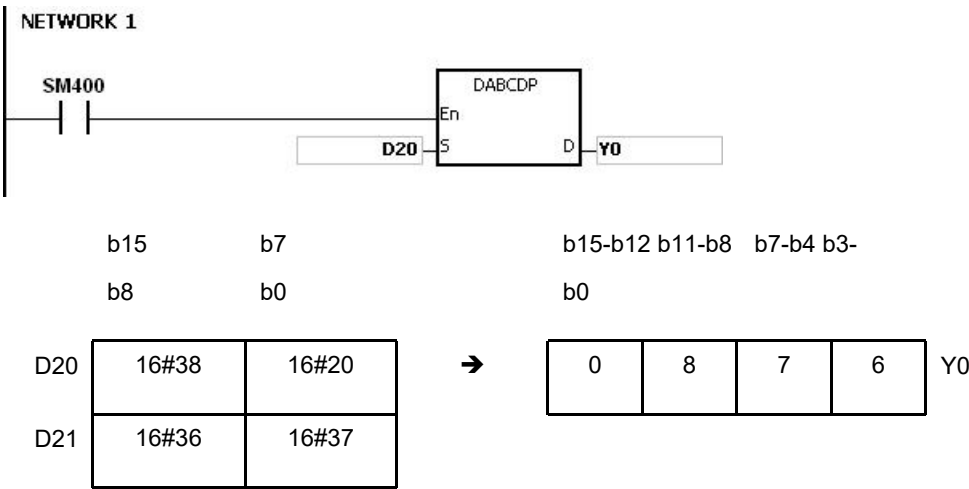




4. Якщо значення в **S** дорівнює 16#20, значення обробляється як 16#30. Якщо значення в **S** дорівнює 16#00, це значення обробляється як символ закінчення.
5. Для 16-бітних інструкцій, якщо **S** є рядком, кількість символів у рядку має бути між 1–4. Для 32-розрядних інструкцій, якщо **S** є рядком, кількість символів у рядку має бути від 1 до 8.

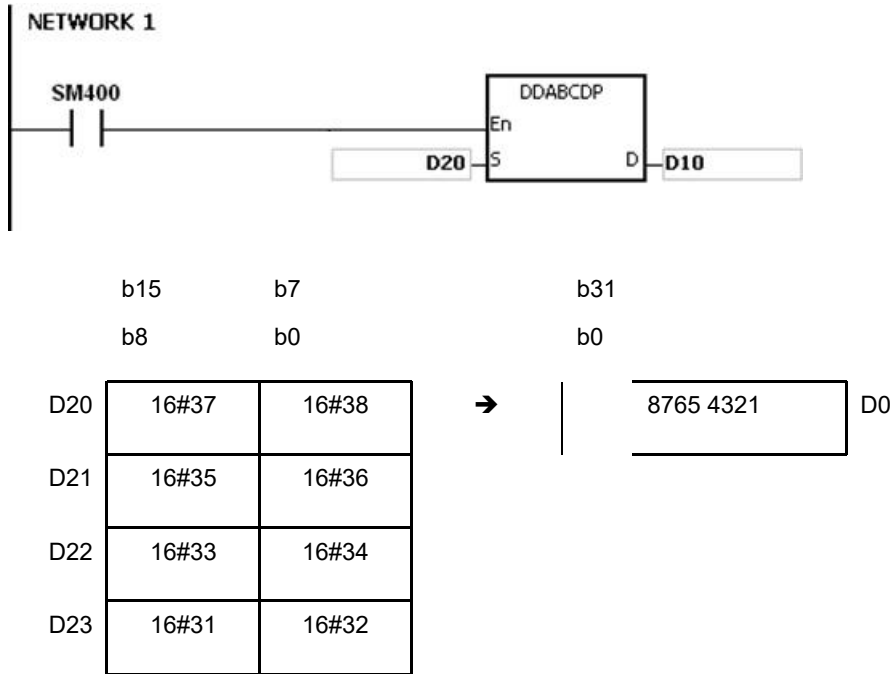
Приклад 1

Припустимо, значення D20=16#3820, D21=16#3637 (ASCII 876). Коли інструкція виконується, інструкція перетворює значення на Y0=16#876.



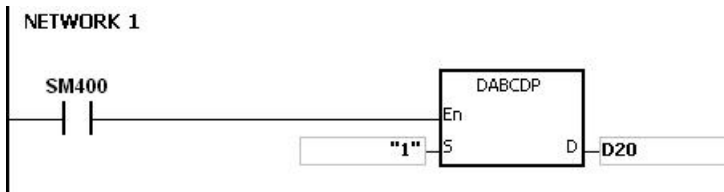
Приклад 2

Припустімо, що значення D20=16#3738, D21=16#3536, D22=16#3334, D23=16#3132 (ASCII 87654321). Коли інструкція виконується, значення (D11, D10)= 16#87654321.



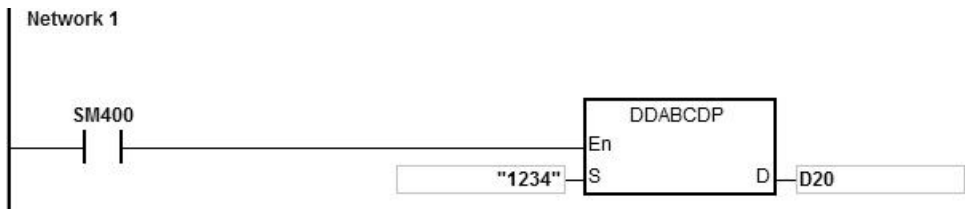
Приклад 3

Припустімо, значення рядка в S дорівнює 1. Коли інструкція виконується, значення D20=16#0001.



Приклад 4

Припустімо, значення рядка в S дорівнює 1234. Коли інструкція виконується, значення (D21, D20)= 16#00001234.



Додаткові зауваження

1. Якщо код ASCII у **S** не є кодами ASCII 16#30–16#39, 16#20 або 16#00, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо **S** є рядком і кількість символів, що містяться в рядку, перевищує діапазон, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Для 16-розрядних інструкцій, якщо ви оголошуєте операнд **S** в ISPSOft, типом даних буде ARRAY [2] WORD/INT.
4. Для 32-розрядних інструкцій, якщо ви оголошуєте операнд **S** в ISPSOft, типом даних є ARRAY [4] з WORD/INT.

API	Код інструкції				Операнд						функція						
2106		\$LEN	P		S, D						Обчислення довжини рядка						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●									
D		●			●	●		●			O	O					

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



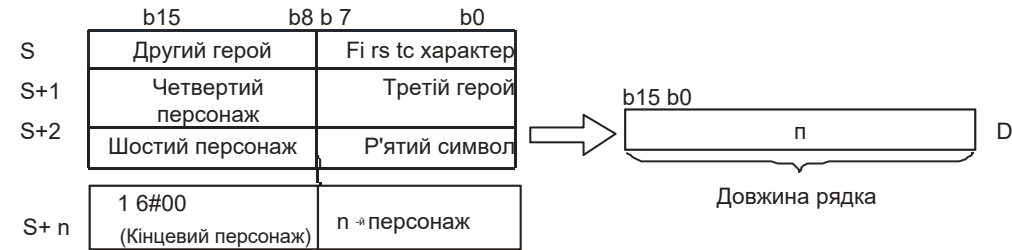
S : Рядок

D : Довжина рядка

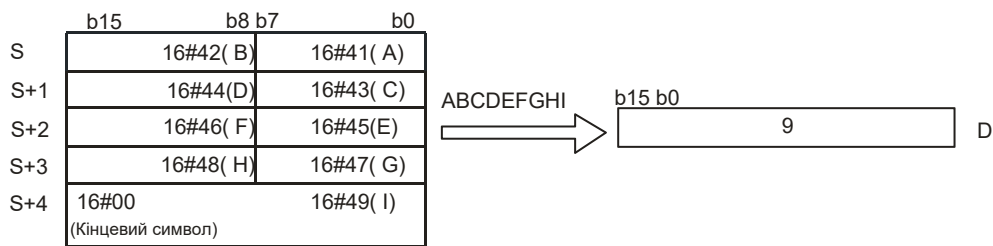
6

Пояснення

- 1. Ця інструкція обчислює довжину рядка в S без символу закінчення (16#00) і зберігає довжина рядка в D .
- 2. Значення, що зберігається в D, має бути між 0–32767. Якщо воно перевищує цей діапазон, значення в D обробляється як 32767.



Якщо дані в S - S +4 дорівнюють ABCDEFGHI, результат обчислення буде таким.

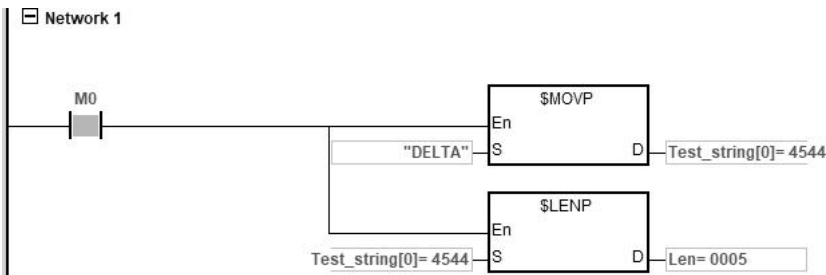


Приклад 1

1. Оголошіть два символи Test String і Len.

Local Symbols				
Class	Identifiers	Address	Type	Initial Value (Acti...
VAR	Test_string	N/A [Auto]	ARRAY [10] OF WORD	N/A
VAR	Len	N/A [Auto]	WORD	N/A

2. Коли M0 перемикається з OFF на ON, інструкція \$LEN підраховує та зберігає довжину рядка у значенні Len.



Додаткові зауваження

- 1. Якщо рядок не закінчується на 16#00, інструкція виконується до максимальної довжини 32767.
- 2. Якщо довжина значення перевищує діапазон пристрою, останній символ обробляється як кінцевий.

API		Код інструкції			Операнд							функція						
2109			\$FSTR	P	S_1, S_2, D							Перетворення числа з плаваючою комою в рядок						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S_1	●	●			●	●		●	●								O	
S_2	●	●			●	●		●	●									
D		●			●	●		●										

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1									●				
S_2		●				●							
D		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
AS		AS		-	

СИМВОЛ

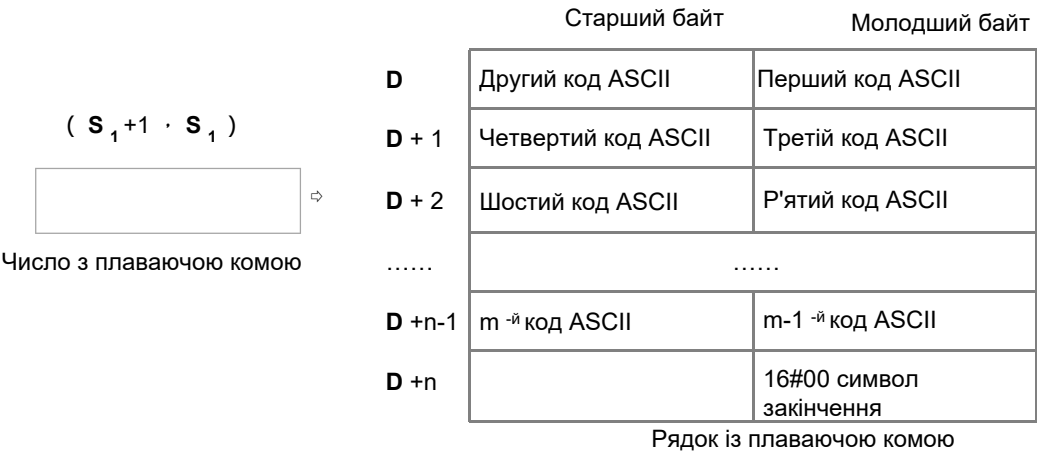
\$FSTR		\$FSTRP	
En		En	
S1	D	S1	D
S2		S2	

- S_1 : Вихідне значення
- S_2 : Перший пристрій, де зберігається формат
- D : Перший пристрій, на якому зберігається результат перетворення

6

Пояснення

- Ця інструкція перетворює число з плаваючою комою в S_1 на рядок відповідно до значення налаштування формату в S_2 і зберігає результат перетворення в D .
- Ця інструкція перетворює число з плаваючою комою в S_1 на рядок і додає кінцевий символ 16#00 до кінця рядка, а потім зберігає результат перетворення в D .



3. Результат перетворення змінюється в залежності від налаштування S_2 .

4. Значення в S_2+1 має бути між $2 \leq S_2+1 \leq 20$; знак (+,-) і цифри цілого або десяткового числа є включено, але десяткова кома в десятковому числі та число степеня не включені.

Операнд	опис
S_2	0: десятковий формат 1: експоненціальний
S_2+1	Номер символу

5. Якщо значення в S_1 є додатним числом, перший код ASCII можна зберегти в D ; якщо значення в S_1 є від'ємним числом, спочатку зберігається знак 16#2D (-), а потім зберігається другий ASCII.

6. Десятковий формат ($S_2=0$)

Після перетворення рядок з плаваючою комою в пристрої D.

Знак	Ціле число	Десяткова
------	------------	-----------

- Значення в S_2+1 має бути між $2 \leq S_2+1 \leq 20$; знак (+,-) і цифри в цілих чи десяткових числах число включено, але десяткова кома в десятковому числі та число степеня не включені.

- Приклад 1

Припустимо, кількість символів дорівнює восьми. Два приклади чисел з плаваючою комою: -1,2345678 і 123456. Розрахунок такий.

D номер	Число з плаваючою комою -1,2345678		Число з плаваючою комою 123456	
	Старший байт	Молодший байт	Старший байт	Молодший байт
D	16#31 (1)	16#2D (-)	16#32 (2)	16#31 (1)
D + 1	16#32 (2)	16#2E (.)	16#34 (4)	16#33 (3)
D + 2	16#34 (4)	16#33 (3)	16#36 (6)	16#35 (5)
D + 3	16#36 (6)	16#35 (5)	16#00 символ закінчення	
D + 4	16#00 символ закінчення	16#38 (8)		

Після перетворення, якщо число з плаваючою комою зберігається в S_2+1 , але якщо довжина перевищує значення в S_2+1 , інструкція округлює цифри в десятковому числі. Рядок із плаваючою комою не обов'язково заповнює весь S_2+1 .

● Приклад 2

Після перетворення, якщо число з плаваючою комою зберігається в $S_2 + 1$, інструкція використовує експоненціальний формат для перетворення. Наприклад, кількість символів – це п'ять цифр, а число з плаваючою комою – 1234567, результат перетворення – 1,2346E+06.

D номер	Число з плаваючою комою 1234567	
D	16#2E (.)	16#31 (1)
D + 1	16#33 (3)	16#32 (2)
D + 2	16#36 (6)	16#34 (4)
D + 3	16#2B (+)	16#45 (E)
D + 4	16#32 (6)	16#30 (0)
D + 5	16#0000 символ закінчення	

● Приклад 3

Після перетворення, якщо число з плаваючою комою зберігається в $S_2 + 1$, інструкція використовує експоненціальний формат для перетворення. Наприклад, кількість символів – це дві цифри, а число з плаваючою комою – 0,00012345, результат перетворення – 1,2E-04.

D номер	Число з плаваючою комою 0,00012345	
D	16#2E (.)	16#31 (1)
D + 1	16#45 (E)	16#32 (2)
D + 2	16#30 (0)	16#2D (-)
D + 3	16#00 символ закінчення	16#34 (4)

● Приклад 4

Після перетворення, якщо абсолютне значення числа з плаваючою комою $\leq 10^{-5}$, інструкція використовує експоненціальний формат для перетворення. Наприклад, кількість символів — це чотири цифри, а число з плаваючою комою — 0,00001234, результат перетворення — 1,234E-05.

D номер	Число з плаваючою комою 0,00001234	
D	16#2E (.)	16#31 (1)
D + 1	16#33 (3)	16#32 (2)
D + 2	16#45 (E)	16#34 (4)
D + 3	16#30 (0)	16#2D (-)
D + 4	16#00 символ закінчення	16#35 (5)

7. Експоненціальний формат ($S_2=1$)

Після перетворення,
рядок floating -point у пристрої D.

Знак	Ціле число	.	Decimal	Експоненціальний
------	------------	---	---------	------------------

- Значення в $S_2 + 1$ має бути між $2 \leq S_2 + 1 \leq 20$; знак (+,-) і цифри в цілих і десяткових числах число включено, але десяткова кома в десятковому числі та число степеня не включені. Після обчислення довжина додає експоненту (чотири цифри) і десяткову крапку десяткового числа.
- Кількість символів у області для цілого числа дорівнює одній цифрі.
- Кількість символів у області експоненти становить чотири символи.
Якщо експонента є додатним числом, інструкція додає 16#2B (+) у область експоненти в **D**. Якщо експонента є від'ємним числом, інструкція додає 16#2D (-) у область експоненти в **D**. Номер символу в області експоненти складається з двох цифр. Якщо в результаті перетворення є лише одна цифра, інструкція додає 16#30 (0) як першу цифру області експоненти.

● приклад

Припустимо, кількість символів дорівнює восьми. Два приклади чисел з плаваючою комою: - 123,456789 і 123456. Обчислення виглядає наступним чином.

D номер	Число з плаваючою комою -123,456789		Число з плаваючою комою 12345	
	Старший байт	Молодший байт	Старший байт	Молодший байт
D	16#31 (1)	16#2D (-)	16#2E (.)	16#31 (1)
D + 1	16#32 (2)	16#2E (.)	16#33 (3)	16#32 (2)
D + 2	16#34 (4)	16#33 (3)	16#35 (5)	16#34 (4)
D + 3	16#36 (6)	16#35 (5)	16#2B (+)	16#45 (E)
D + 4	16#45 (E)	16#38 (8)	16#34 (4)	16#30 (0)
D + 5	16#30 (0)	16#2B (+)	16#00 символ закінчення	
D + 6	16#00 символ закінчення	16#32 (2)		

Після перетворення, якщо число з плаваючою комою зберігається в $S_2 + 1$, а інструкція округлює додаткове цифри.

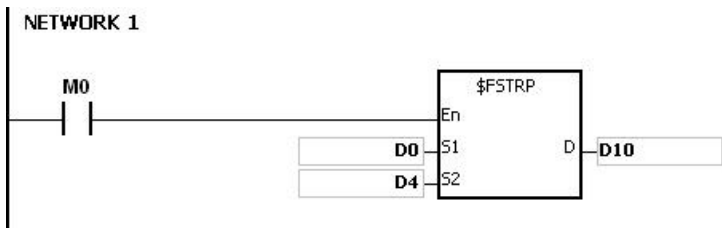
Результат перетворення довжини рядка для плаваючого числа -123,456789 у восьми символах дорівнює 13 (символ закінчення виключений).

Результат перетворення довжини рядка для плаваючого числа 12345 у восьми символах дорівнює 10 (кінець символ виключено).

Рядок із плаваючою комою не обов’язково має заповнювати всі символи.

Приклад 1

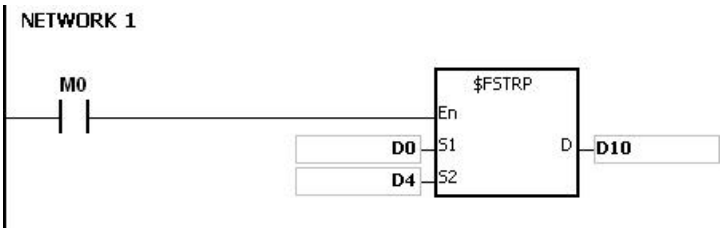
Число з плаваючою комою в (D1, D0= 12,3456) перетворюється на десятковий формат у рядку (D4=0, D5=8).



D10	16#32 (2)	16#31 (1)
D11	16#33 (3)	16#2E (.)
D12	16#35 (5)	16#34 (4)
D13	16#00 символ закінчення	16#36 (6)

Приклад 2

Число з плаваючою комою в (D1, D0 = 0,0012345678) перетворюється на експоненціальний формат у рядку (D4=1, D5=8).



D10	16#2E (.)	16#31 (1)
D11	16#33 (3)	16#30 (2)
D12	16#35 (5)	16#34 (4)
D13	16#37 (7)	16#36 (6)
D14	16#45 (E)	16#38 (8)
D15	16#30 (0)	16#2D (-)
D16	16#00 символ закінчення	16#33 (3)

Додаткові зауваження

1. Якщо значення в S_1 перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.
2. Якщо значення в S_2 не дорівнює ні 0, ні 1, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо значення в $S_2 + 1$ не знаходиться в межах $2 \leq S_2 + 1 \leq 20$, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
4. Якщо ви оголошуєте операнд S_2 в ISPSOft, типом даних буде ARRAY [2] WORD/INT.

API		Код інструкції			Операнд						функція						
2110			\$FVAL	P	S, D						Перетворення рядка в число з плаваючою комою						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●	●						O		
D		●			●	●		●									

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D									●				

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
AS		AS		-	

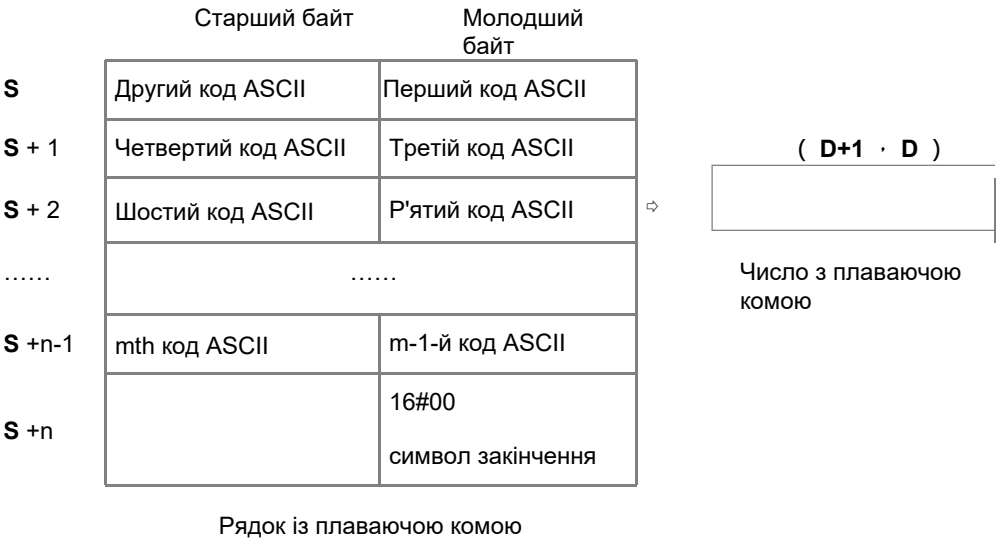
СИМВОЛ

\$FVAL		\$FVALP	
En		En	
S	D	S	D

- S** : Вихідне значення
- D** : Пристрій, на якому зберігається результат перетворення

Пояснення

1. Ця інструкція перетворює рядок у **S** на число з плаваючою комою та зберігає результат перетворення в **D**.



- Зверніться до наступних розділів щодо використання коду ASCII.
- Для десяткового чи експоненціального форматів максимальна довжина рядка з плаваючою комою (m) становить 24 символи (за винятком символу закінчення 16#00), а максимальна довжина n становить 13 символів.

2. Формат значення в **S** може бути десятковим або експоненціальним. Інструкція автоматично визначає формат відповідно до значення в **S**.

- Десятковий формат: довжина рядка з плаваючою комою дорівнює дев'яти; символ закінчення 16#00 не включається.

	Старший байт	Молодший байт	
S	16#31 (1)	16#32 (2)	
S + 1	16#32 (2)	16#2E (.)	
S + 2	16#34 (4)	16#33 (3)	⇒ D 21,234567
S + 3	16#36 (6)	16#35 (5)	
S + 4	16#00 символ закінчення	16#37 (7)	

АБО

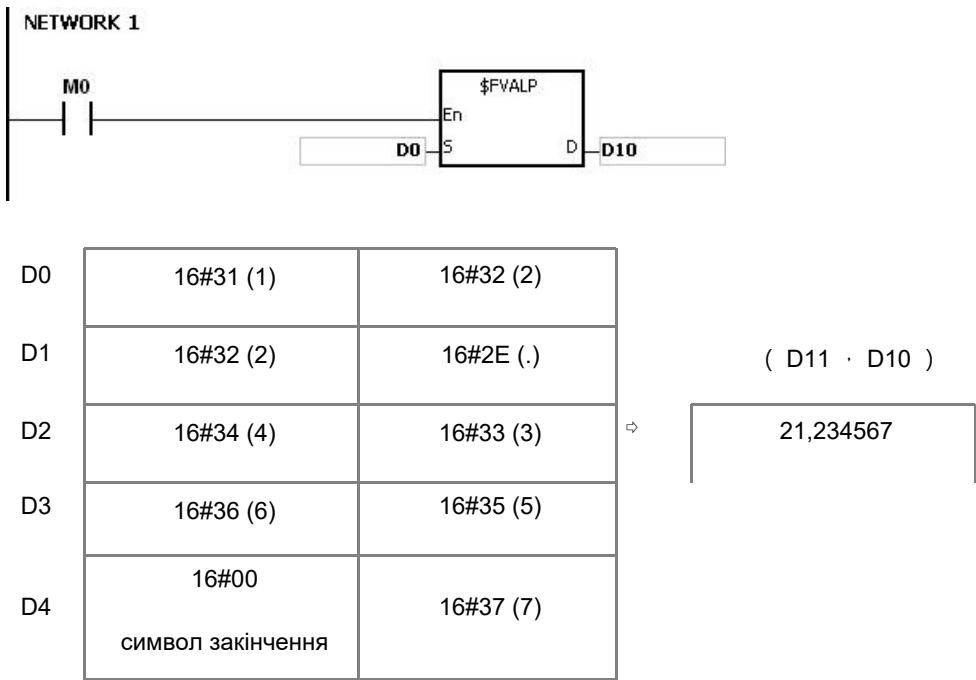
	Старший байт	Молодший байт	
S	16#31 (1)	16#2D (-)	
S + 1	16#32 (2)	16#2E (.)	
S + 2	16#34 (4)	16#33 (3)	⇒ D -1,234568
S + 3	16#36 (6)	16#35 (5)	
S + 4	16#00 символ закінчення	16#38 (8)	

- Експоненціальний формат: довжина рядка з плаваючою комою дорівнює 9; символ закінчення 16#00 не включається.

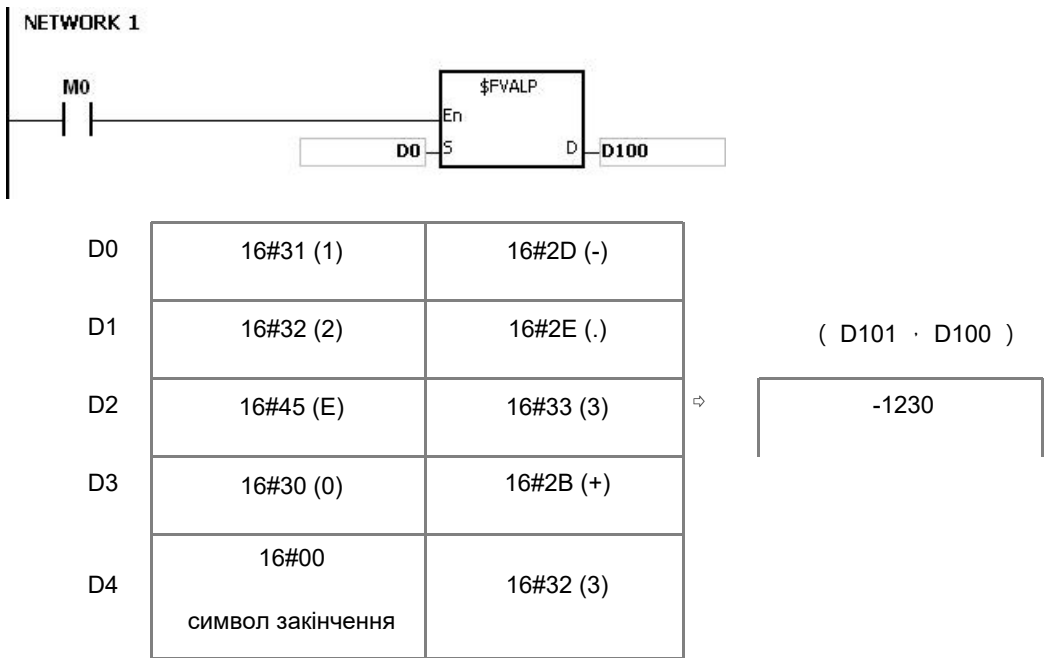
	Старший байт	Молодший байт	
S	16#31 (1)	16#2D (-)	
S + 1	16#32 (2)	16#2E (.)	
S + 2	16#45 (E)	16#33 (3)	⇒ D -1230
S + 3	16#30 (0)	16#2B (+)	
S + 4	16#00 символ закінчення	16#32 (3)	

3. Якщо знаковий код у **S** дорівнює 16#20, 16#30 або 16#2B, то результатом перетворення є додатне значення. Якщо код знака в **S**, дорівнює 16#2D, то результат перетворення є від'ємним значенням.

Приклад 1



Приклад 2



Додаткові зауваження

1. Якщо довжина рядка в **S** перевищує 25 байт і не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#2003.

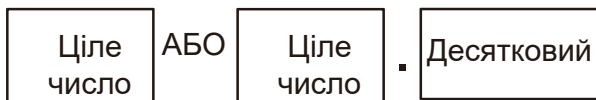
2. Є деякі правила для значення в **S**. якщо значення в **S** не відповідає правилам, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

Перший код ASCII: дозволені знаки 16#2B(+), 16#2D(-), пропуск 16#20 і числа 16#30(0)–16#39(9).

Якщо перший код ASCII є знаком або пробілом, другий код ASCII має бути числом.

Другий код ASCII може бути в десятковому або експоненціальному форматі.

- Десятковий формат:



Десятна крапка «.» (16#2E) може зустрічатися лише один раз, і до та після коми мають бути числа.

- Експоненціальний формат:



Десятна крапка «.» (16#2E) може зустрічатися лише один раз, і до та після коми мають бути числа.

Перед експонентом має бути число.

Цілі числа: дозволені лише числа «0» (16#30)–«9» (16#39).

Десяткові знаки: дозволені лише числа «0» (16#30)–«9» (16#39). Експоненти:

Формат чотиризначного коду ASCII наведено нижче.



Має бути буква «Е» (16#45) або «е» (16#65), і вона може зустрічатися лише один раз.

Має бути знак «+» (16#2B) або «-» (16#2D), і він може бути лише один раз. Має бути дві

цифри; цифри «0» (16#30)–«9» (16#39).

3. Якщо рядок у **S** виходить за межі діапазону, інструкція не виконується, SM увімкнено, а код помилки в SR0 — 16#2003.

- Якщо першим символом у рядку є число "0"–"9" (16#30–16#39), дійсний діапазон для рядка з плаваючою комою становить 1–24. Мінімальна довжина рядка дорівнює одиниці.
- Якщо першим символом у рядку є пробіл (16#20) або знак («+» (16#2B) або «-» (16#2D)), допустимим діапазоном для рядка з плаваючою комою є 2– 24. Мінімальна довжина рядка дорівнює двом («+1»).

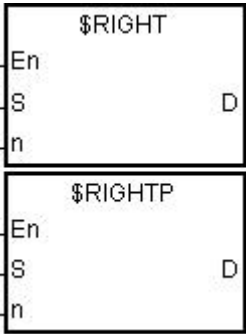
4. Якщо результат перетворення перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2013.

API	Код інструкції				Операнд								функція			
2111		\$RIGHT	P		S, n, D								Отримання символів із рядка, що починається справа.			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●	●						○	
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

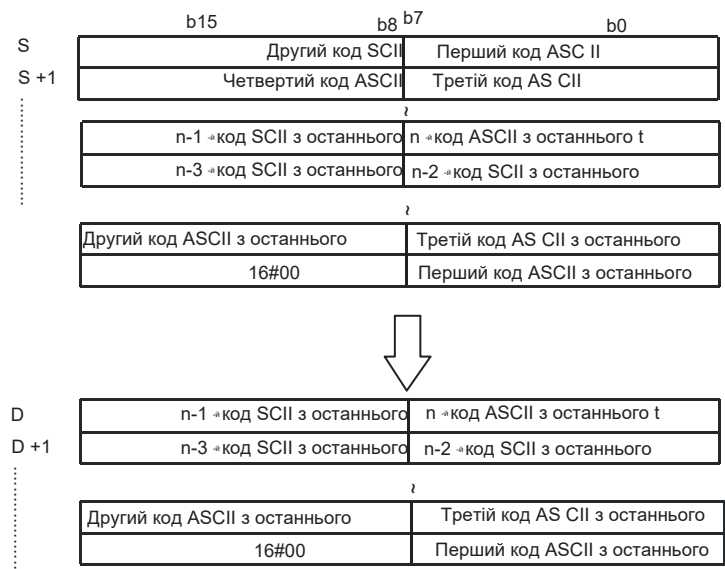
СИМВОЛ



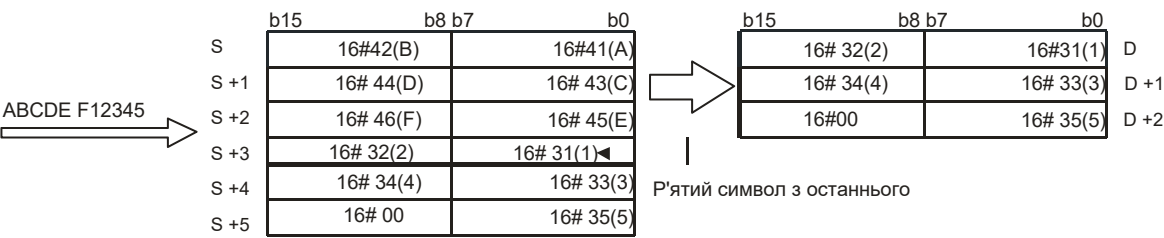
- S :** Рядок
- n :** Кількість символів, які потрібно отримати
- D :** Пристрій, де зберігаються отримані символи

Пояснення

1. Ця інструкція отримує **n** символів із рядка в **S** справа та зберігає отримані символи в **D** . Якщо тип даних **S** (вихідне значення) — рядок, максимальна довжина значення в **S** становить 31 символ. Якщо тип даних **S** (вихідне значення) не є рядком, максимальна довжина значення в **S** становить 255 символів.
2. Якщо **n** дорівнює 0, значення в **D** дорівнює 0. Максимальна довжина для **n** становить 255 символів. .

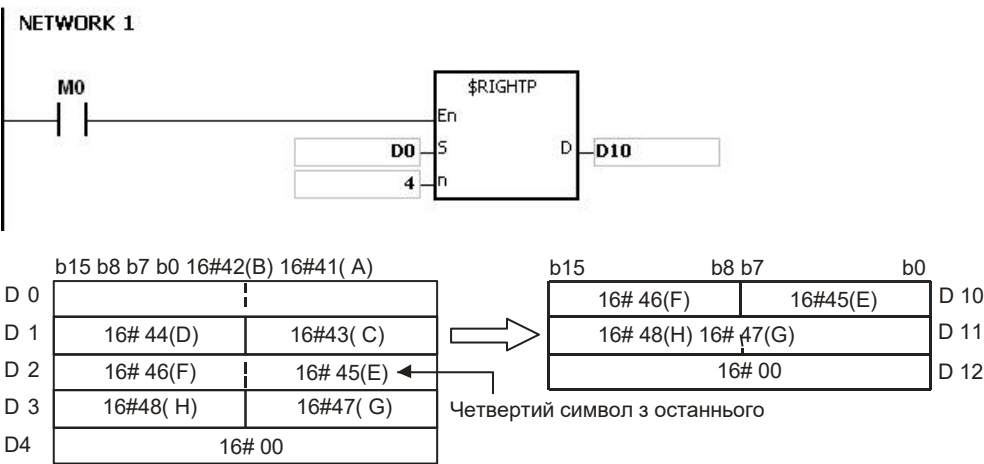


Якщо дані в **S** — ABCDEF12345, а **n** — 5, інструкція отримує п'ять символів у рядку в **S** справа.
Результат перетворення виглядає наступним чином.



6

приклад



Додаткові зауваження

1. Якщо операнд **S** не є рядком (\$), а пристроєм із рядком, рядок у **S** може містити до 256 символів (включаючи символ закінчення 16#00). Якщо рядок у **S** не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
2. Якщо **n** менше 0, воно обробляється як 0. Якщо **n** більше за довжину рядка в **S**, воно обробляється як довжина рядка **S**.
3. Якщо **D** недостатньо, щоб містити **n** символів, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції				Операнд						функція						
2112		\$LEFT	P		S, n, D						Отримання символів із рядка, що починається зліва.						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●			●	●		●	●						○		
n	●	●			●	●		●	●		○	○	○	○			
D		●			●	●		●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
n		●				●							
D		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
AS		AS		-	

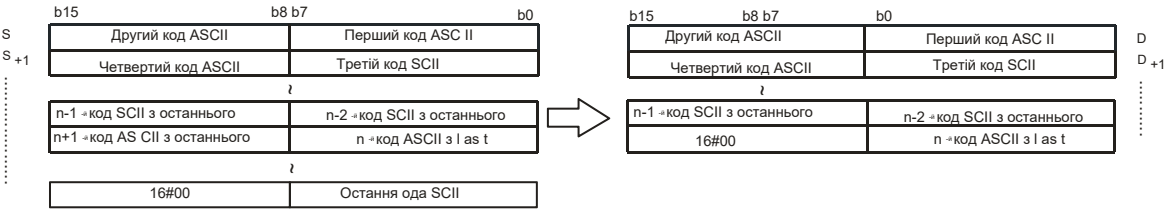
СИМВОЛ



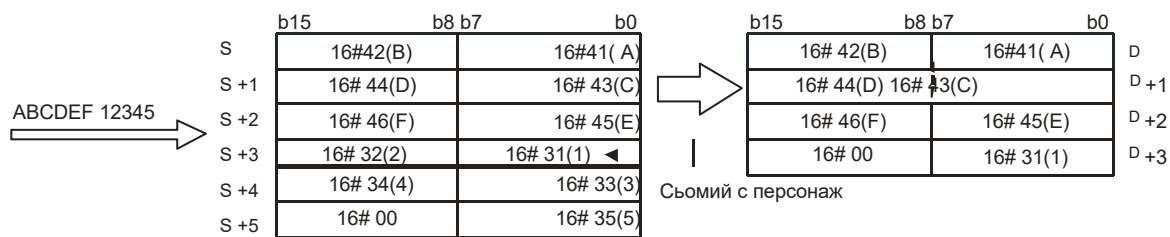
- S** : Рядок
- n** : Кількість символів для отримання
- D** : Пристрій, де зберігаються отримані символи

Пояснення

1. Ця інструкція отримує **n** символів із рядка в **S**, починаючи зліва, і зберігає отримані символи в **D**. Коли **S** є рядковим пристроєм, максимальна довжина значення в **S** становить 31 символ; якщо **S** не є рядковим пристроєм, максимальна довжина значення в **S** становить 255 символів.
2. Якщо **n** дорівнює 0, значення в **D** дорівнює 0.

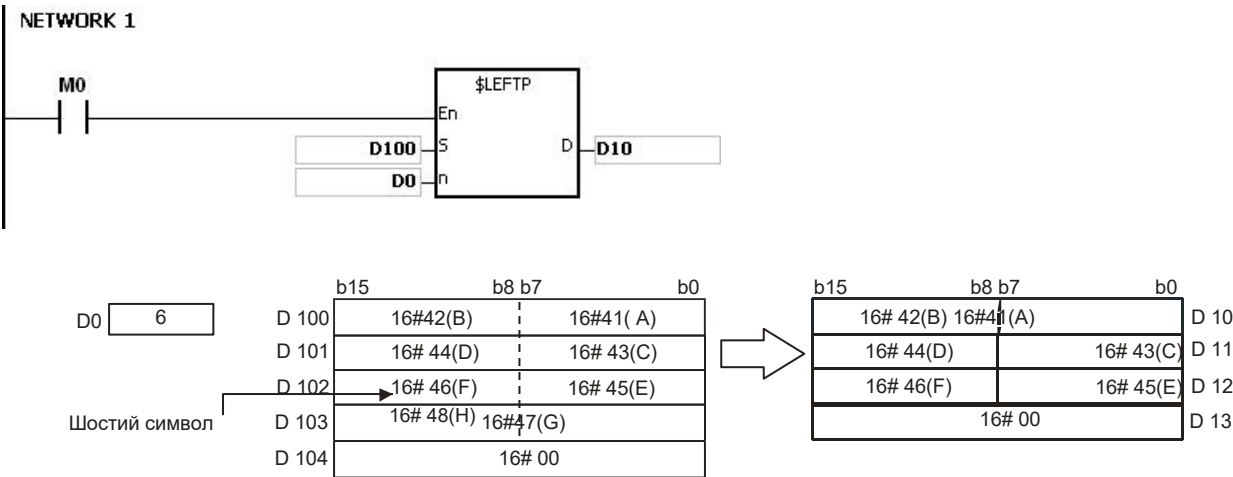


Якщо дані в **S** є ABCDEF12345 і **n** дорівнює 7, інструкція отримує сім символів у рядку в **S** зліва. Результат перетворення виглядає наступним чином.



приклад

Коли M0 увімкнено, інструкція отримує шість символів, починаючи з символу в D100, і зберігає їх у D10.



Додаткові зауваження

1. Якщо операнд **S** не є рядком (\$), а пристроєм із рядком, рядок у **S** може містити до 256 символів (16#00 включений символ закінчення). Якщо рядок у **S** не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 – 16#200E.
2. Якщо **n** менше 0, воно обробляється як 0. Якщо **n** більше за довжину рядка в **S**, воно обробляється як довжина рядка **S**.
3. Якщо **D** недостатньо, щоб містити **n** символів, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд							функція						
2113		\$MIDR	P	S_1, S_2, D							Отримання частини рядка						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●		●	●						○		
S_2	●	●			●	●		●	●		○						
D		●			●	●		●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							●
S_2		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

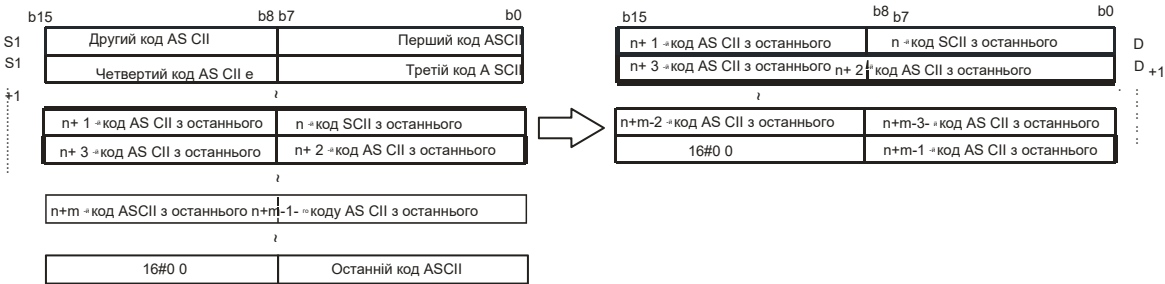
\$MIDR		\$MIDRP	
En		En	
S1	D	S1	D
S2		S2	

- S_1 : Рядок
- S_2 : Частина рядка, яку потрібно отримати
- D : Пристрій, де зберігаються отримані символи

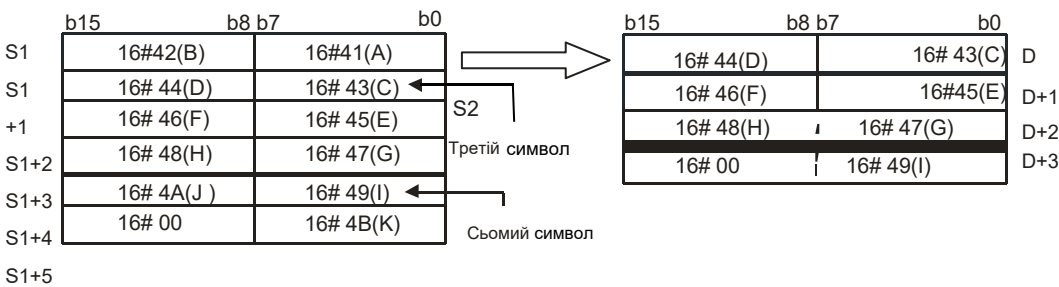
6

Пояснення

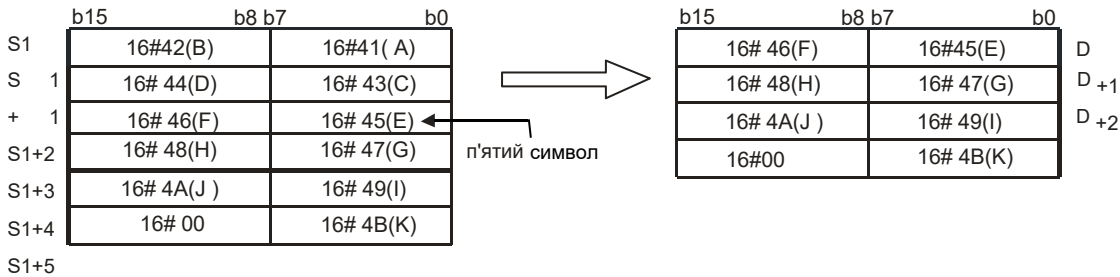
1. Припустимо, що значення в S_2 і $S_2 + 1$ дорівнюють n і m відповідно. Ця інструкція отримує m символів, починаючи з n -го символу у рядку в S_1 і збирає їх у D.



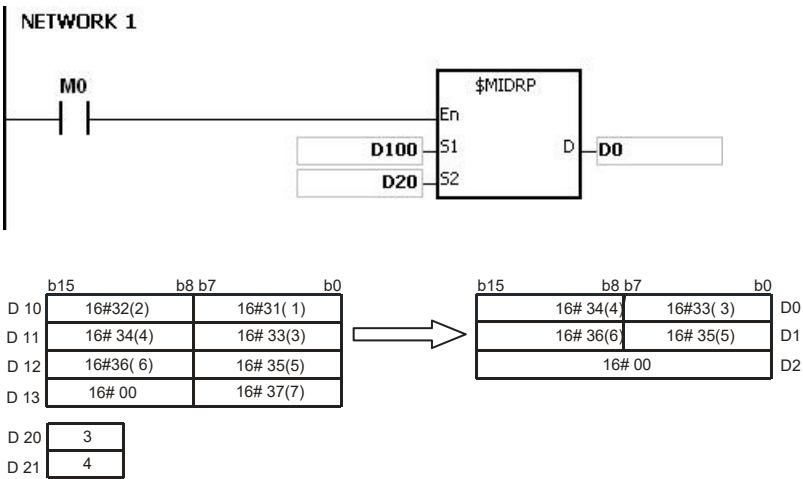
2. Якщо дані в S_1 — ABCDEFGHIJK, значення в S_2 — 3, а значення в $S_2 + 1$ — 7, інструкція отримує сім символів, починаючи з третього символу в рядку зліва. Результат перетворення виглядає наступним чином.



3. Якщо значення S_2 дорівнює $S_2 \leq 0$, $S_2 + 1 < -1$ або $S_2 + 1 = 0$, інструкція не виконується.
4. Якщо значення в $S_2 + 1$ дорівнює -1, інструкція отримує символи в S_1 , починаючи від символу, зазначеного значенням в S_2 , до останнього символу в S_1 .
5. Якщо дані в S_1 є ABCDEFGHIJK, значення в S_2 дорівнює 5, а значення в $S_2 + 1$ дорівнює -1, результат перетворення буде таким.



приклад



Додаткові зауваження

1. Якщо операнд S_1 не є рядком (\$), а пристроєм із рядком, рядок у S_1 може містити до 256 символів (включаючи символ закінчення 16#00). Якщо рядок у S_1 не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
2. Якщо значення S_2 дорівнює $S_2 \leq 0$ або $S_2 + 1 < -1$, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо значення в S_2 або $S_2 + S_2 + 1$ більше, ніж довжина рядка в S_1 , SM0 увімкнено, а код помилки в SR0 — 16#2003.
4. Якщо значення в $S_2 + 1$ більше, ніж кількість символів, які можна отримати з рядка в S_1 , SM0 увімкнено, а код помилки в SR0 — 16#2003.
5. Якщо D недостатньо, щоб містити $S_2 + 1$ символів, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
6. якщо ви оголошуєте операнд S_2 в ISPSOft, типом даних буде ARRAY [2] з WORD/INT.

API	Код інструкції				Операнд						функція					
2115		\$SER	P	S ₁ , S ₂ , n, D						Пошук рядка						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●						○	
S ₂	●	●			●	●		●	●						○	
n	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●			○	○				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							●
S_2		●				●							●
n		●				●							
D		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

\$SER				\$SERP			
En				En			
S1			D	S1			D
S2				S2			
N				N			

- S_1 : Рядок для пошуку
- S_2 : Рядок для пошуку
- n : n-й символ у S_2 , де починається пошук
- D : Результат пошуку

Пояснення

- Ця інструкція шукає рядок у S_2 , починаючи з n-го символу рядка в S_1 , і зберігає результат пошуку в D.
- Коли n-й символ перевищує довжину рядка в S_2 , або $S_1 > S_2$, D дорівнює 0.
- Припустимо, що рядок у S_2 – «ABCDEFGHIJK», рядок у S_1 – «EFGH», а n дорівнює 3. Пошук починається з третього символу в S_2 , а значення в D дорівнює 5.

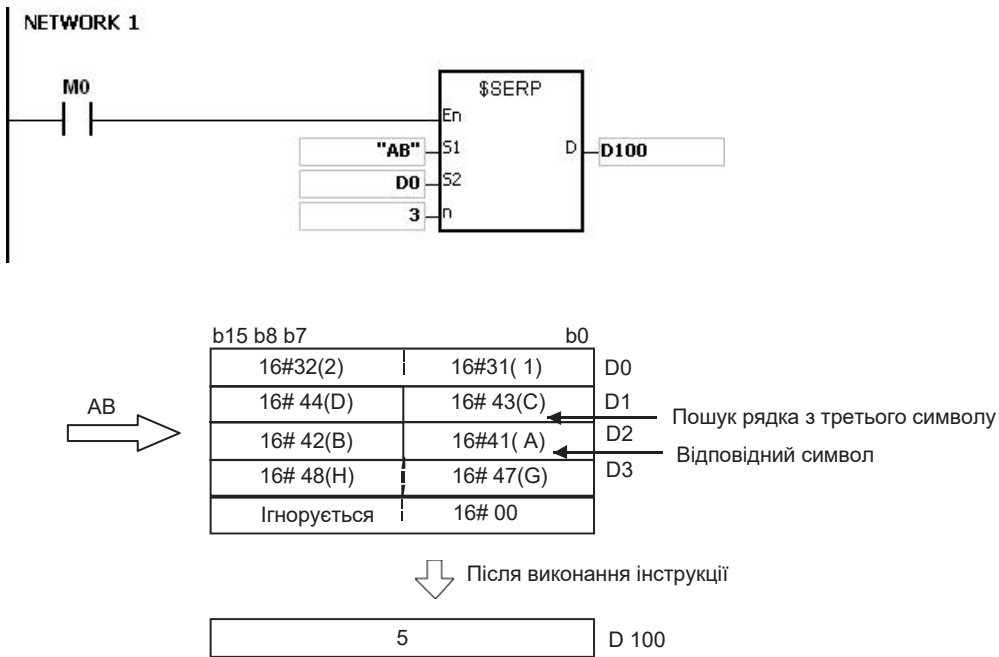
S1	b15	b8	b7	b0	b15	b8	b7	b0	S2
	16#46(F)			16#45(E)	16#42(B)			16#41(A)	
	16#48(H)			16#47(G)	16#44(D)			16#43(C)	
	Ігнорується			16 №00	16#46(F)			16#45(E)	
S1+1					16#48(H)			16#47(G)	S2+1
S1+2					16#4A(J)			16#49(I)	S2+2
					16#00			16#4B(K)	S2+3
									S2+4
									S2+5

Пошук рядка з третього символу
Відповідний символ

Після того, як інструкція буде виконана

5 D

приклад



Додаткові зауваження

1. Якщо операнд S_1 не є рядком (\$), а пристроєм із рядком, рядок у S_1 може містити до 256 символів (16#00 включений символ закінчення). Якщо рядок у S_1 не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
2. Якщо операнд S_1 не є рядком (\$), а пристроєм із рядком, рядок у S_2 може містити до 256 символів (16#00 включений символ закінчення). Якщо рядок у S_1 не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
3. Якщо n менше або дорівнює 0, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API		Код інструкції			Операнд						функція					
2116			\$RPLC	P	S ₁ , S ₂ , S ₃ , S ₄ , D						Заміна символів у рядку					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●						○	
S ₂	●	●			●	●		●	●						○	
S ₃	●	●			●	●		●	●		○	○	○	○		
S ₄	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●				●							●
S ₂		●				●							●
S ₃		●				●							
S ₄		●				●							
D		●				●							

Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
AS		AS	-

СИМВОЛ

\$RPLC		\$RPLCP	
En		En	
S1	D	S1	D
S2		S2	
S3		S3	
S4		S4	

- S_1 : Рядок для пошуку та відповідності символів замінено
- S_2 : Новий рядок
- S_3 : Кількість символів у S_1 , які потрібно замінити
- S_4 : Рядок S_2 замінює символи в S_1 , починаючи з символу, зазначеного значенням у S_4 .
- D : Пристрій, де зберігається результат

Пояснення

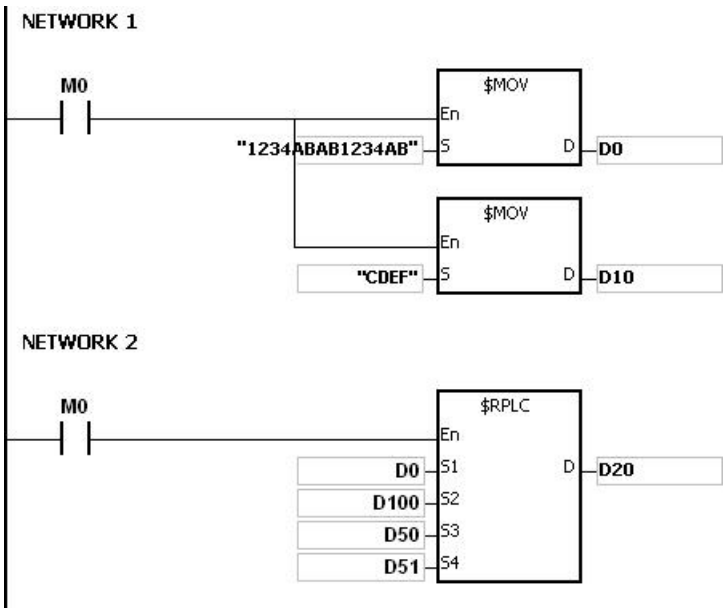
1. Ця інструкція замінює символи в S_1 на символи в S_2 , починаючи з символу в S_1 , який відповідає символу в S_4 . Кількість символів, які замінюються, вказується значенням у S_3 , а інструкція зберігає результат у D.
2. Наприклад, інструкція замінює чотири символи, починаючи з шостого символу в рядку «12345 6789 0» на «MN», і результатом є «12345 MN 0».



- 3. Якщо рядок у **S₂** є 16#00, інструкція видаляє відповідні символи.
- 4. Якщо значення в **S₃** більше, ніж кількість символів, яку інструкція може замінити в рядку в **S₁**, інструкція замінює символи в **S₁**, починаючи від символу, зазначеного значенням у **S₄**, до останнього символу в **S₁**.
- 5. Якщо значення в **S₃** дорівнює 0, інструкція не виконується.

приклад

Коли M0 увімкнено, дані в D0–D7 мають значення «1234ABAB1234AB», а дані в D10–D11 мають значення «CDEF». Коли \$RPLC інструкція виконується, інструкція замінює символи в D0–D7, починаючи з символу, зазначеного значенням у D51, на символи в D10–D11. Кількість символів для заміни вказується значенням у D50, а інструкція зберігає результат у D20–D27.



Якщо значення в D50 і D51 дорівнюють 3 і 4 відповідно, результат виконання буде таким.

	b 15	b8 b7	b 0		b15	b8 b7	b0	
D 0	16# 32 (2)	16#31(1)			16#32(2)	16#31(1)		D20
D 1	16# 34 (4)	16#33(3)			16#43(C)	16#33(3)		D21
D2	16# 42(B)	16 #41(A)			16#45(E)	16#44(D)		D22
D3	16#42(B)	16 #41(A)			16#41(A)	16#46(F)		D23
D4	16#32(2)	16#31(1)			16#31(1)	16#42(B)		D24
D5	16#34(4)	16#33(3)			16#33(3)	16#32(2)		D25
D6	16#42(B)	16#41(A)			16#41(A)	16#34(4)		D26
D7	Ігнорується	16 №00			16#00	16#42(B)		D27
після виконання								
D10	16# 44 (D)	1 6#43(C)						
D11	16# 45 (F)	1 6#45(E)						
D12	Ігнорується	16 №00						忽略
D50	3		Кількість героїв, які замінено		取代數量			
D51	4		Характер, який його замінив		開始取代的位置			

Якщо значення в D50 і D51 дорівнюють 4 і 4 відповідно, результат виконання буде таким.

	b15	b8 b7	b0		b15	b8 b7	b0	
D0	16#32(2)	16#31(1)		➡ Після виконання інструкції	16# 32(2)	16#31(1)		D 20
D 1	16# 34(4)	16# 33(3)			16# 43(C)	16# 33(3)		D 21
D 2	16# 42(B)	16# 41(A)			16# 45(E)	16# 44(D)		D 22
D 3	16# 42(B)	16# 41(A)			16# 42(B)	16# 46(F)		D 23
D 4	16# 32(2)	16# 31(1)			16#32 (2)	16#31 (1)		D 24
D 5	16# 34(4)	16# 33(3)			16# 34(4)	16# 33(3)		D 25
D 6	16# 42(B)	16# 41(A)			16# 42(B)	16#41 (A)		D 26
D7	Ігнорується	16#00			16# 0000			D 27
D 10	16# 44(D)	16# 43(C)						
D 11	16# 45(F)	16# 45(E)						
D 12	Ігнорується	16#00						
D 50	4		Кількість символів, які замінюються					
D 51	4		Символ, який відтворюється					

Якщо значення в D50 і D51 дорівнюють 20 і 4 відповідно, результат виконання буде таким.

	b15	b8 b7	b0		b15	b8 b7	b0	
D0	16#32(2)		16#31(1)		16# 32(2)		16#31(1)	D 20
D1	16# 34(4)		16# 33(3)		16# 43(C)		16# 33(3)	D 21
D2	16# 42(B)		16# 41(A)		16# 45(E)		16# 44(D)	D 22
D3	16# 42(B)		16# 41(A)		16# 00		16# 46(F)	D 23
D4	16# 32(2)		16# 31(1)					
D5	16# 34(4)		16# 33(3)					
D6	16# 42(B)		16# 41(A)					
D7	Ігнорується		16# 00					

Після інструктажу виконується

D 10	16# 44(D)	16# 43(C)
D 11	16# 45(F)	16# 45(E)
D 12	Ігнорується	16# 00

D 50	20	Кількість символів, які замінюються
D 51	4	Символ, який замінюється

Якщо значення в D50, D51 і D10 дорівнюють 3, 4 і 16#00 відповідно, результат виконання буде таким. Трійка символи в D0-D7, починаючи з четвертого символу, видаляються.

	b15	b8 b7	b0		b15	b8 b7	b0	
D 0	16#32(2)		16#31(1)		16# 32(2)		16#31(1)	D 20
D 1	16# 34(4)		16# 33(3)		16# 41(A)		16# 33(3)	D 21
D 2	16# 42(B)		16# 41(A)		16# 31(1)		16# 42(B)	D 22
D 3	16# 42(B)		16# 41(A)		16# 33(3)		16# 32(2)	D 23
D 4	16# 32(2)		16# 31(1)		16# 41(A)		16# 34(4)	D 24
D 5	16# 34(4)		16# 33(3)		16# 00		16#42(B)	D 25
D 6	16# 42(B)		16# 41(A)					
D7	16# 00							

Після виконання інструкції

D 10	16# 00
------	--------

D 50	3	Кількість символів, які замінюються
D 51	4	Символ, який замінюється

Додаткові зауваження

1. Якщо рядок у **S₁** не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
1. Якщо рядок у **S₂** не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
3. Якщо значення в **S₃** <0 або **S₄** <=0 або значення в **S₄** більше, ніж довжина рядка в **S₁**, SM0 увімкнено, і помилка код у SR0 – 16#2003.
4. Якщо значення в рядку (16#00 разом із символом закінчення) у **S1** після заміни перевищує 256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції				Операнд						функція					
2117		\$DEL	P	S ₁ , S ₂ , S ₃ , D						Видалення символів у рядку						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●						○	
S ₂	●	●			●	●		●	●		○	○	○	○		
S ₃	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING			
S ₁		●				●										●
S ₂		●				●										
S ₃		●				●										
D		●				●										
						Інструкція пульсу		16-бітна інструкція				32-розрядна інструкція				
СИМВОЛ						AS		AS				-				

\$DEL				\$DELP			
En				En			
S1			D	S1			D
S2				S2			
S3				S3			

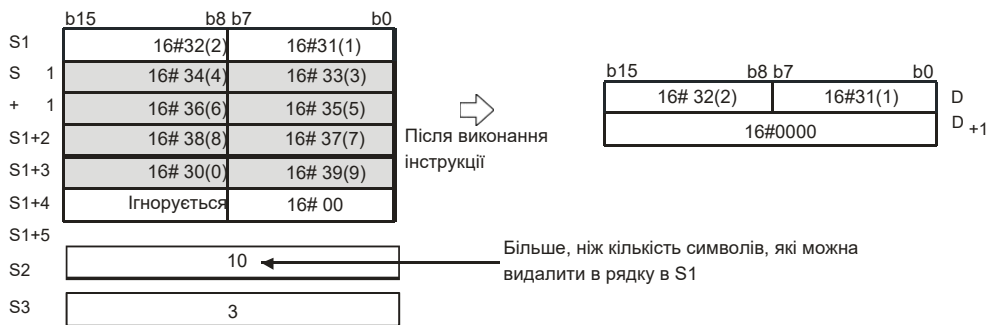
- S_1 : Рядок
- S_2 : Кількість символів, які потрібно видалити
- S_3 : Символи в S_1 видаляються, починаючи з символу, зазначеного значенням у S_3 .
- D : Пристрій, де зберігається результат виконання

Пояснення

1. Ця інструкція видаляє символи в S_1 , починаючи з символу, зазначеного значенням у S_3 . Кількість символів, які видаляються, позначаються значенням у S_2 , а інструкція зберігає результат у D.
2. Наприклад, інструкція видаляє чотири символи, починаючи з третього символу в рядку «12 3456 7890» в S_1 і зберігає результат «127890» в D.



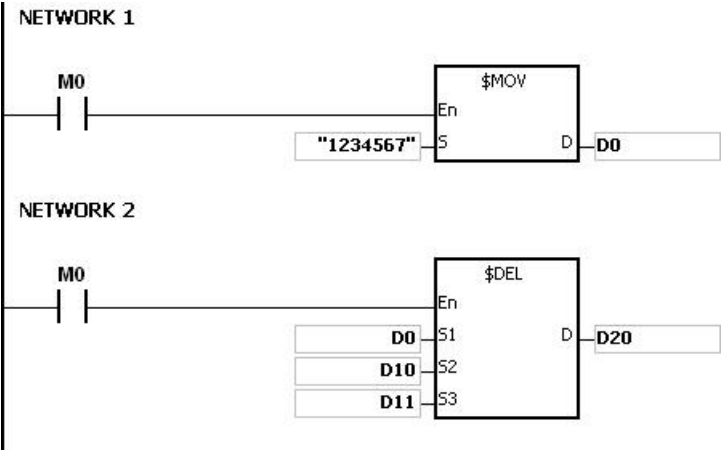
3. Якщо значення в S_2 більше, ніж кількість символів, які можна видалити в рядку в S_1 , інструкція видаляє символи в S_1 , починаючи від символу, зазначеного значенням у S_3 , до останнього символу в S_1 , і зберігає 16#00 у D .



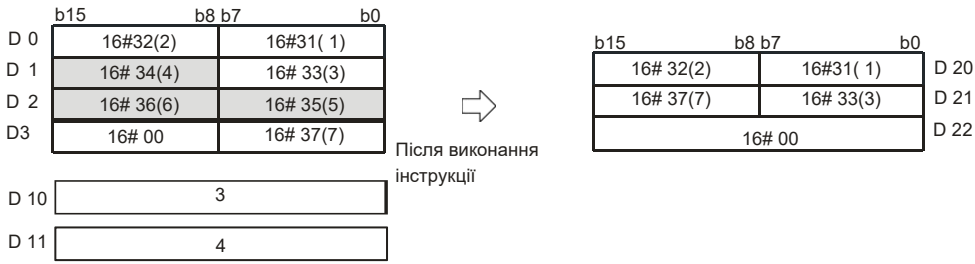
4. Якщо значення в S_2 дорівнює 0, інструкція не виконується.

приклад

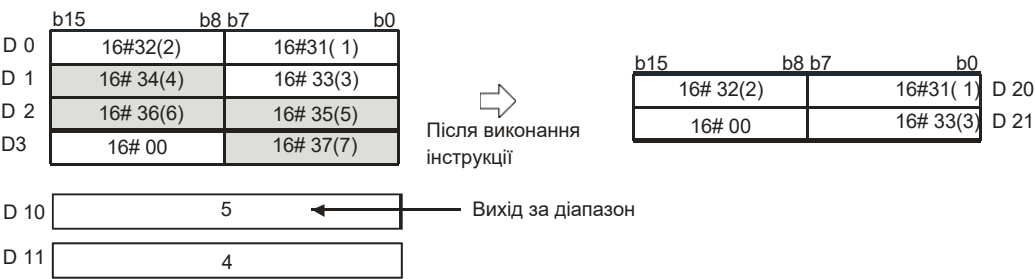
Коли $M0$ увімкнено, дані в $D0$ – $D3$ є «1234567». Коли виконується інструкція $\$DEL$, вона видаляє символи в $D0$ – $D3$, починаючи з символу, зазначеного значенням у $D11$. Кількість символів для видалення вказується значенням у $D10$, а інструкція зберігає результат у $D20$ – $D23$.



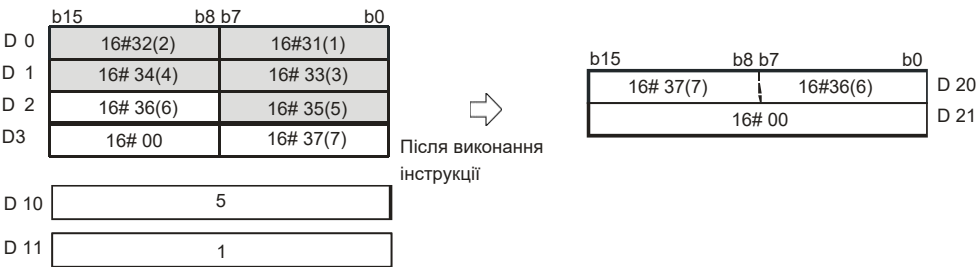
Якщо значення в $D10$ і $D11$ дорівнюють 3 і 4 відповідно, результат виконання буде таким.



Якщо значення в $D10$ і $D11$ дорівнюють 5 і 4 відповідно, результат виконання буде таким. Зверніть увагу, що кількість символів для видалення перевищує діапазон, інструкція видаляє символи в $D0$ – $D3$, починаючи з четвертого символу до останнього символу.



Якщо значення в D10 і D11 дорівнюють 5 і 1 відповідно, результат виконання буде таким.



Додаткові зауваження

1. Якщо рядок у S_1 не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
2. Якщо значення в S_2 менше 0, значення в S_3 менше або дорівнює 0, або значення в S_3 більше за довжину рядка в S_1 , SM0 увімкнено, а код помилки в SR0 – 16#2003.

API		Код інструкції			Операнд						функція					
2118			\$CLR	P	S						Очищення рядка					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S		●			●	●		●								
Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING			
S		●				●										
							Інструкція пульсу			16-бітна інструкція			32-розрядна інструкція			
							AS			AS			-			

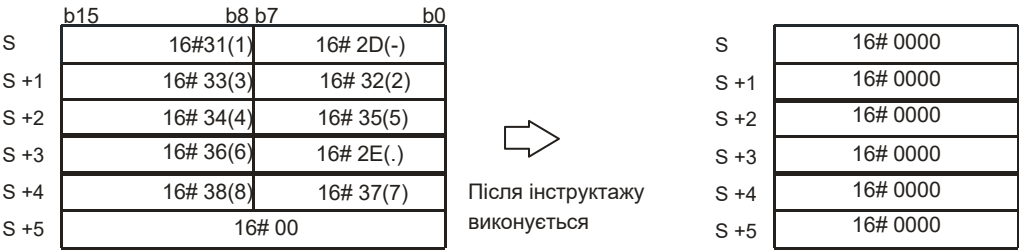
СИМВОЛ



S : Рядок для очищення

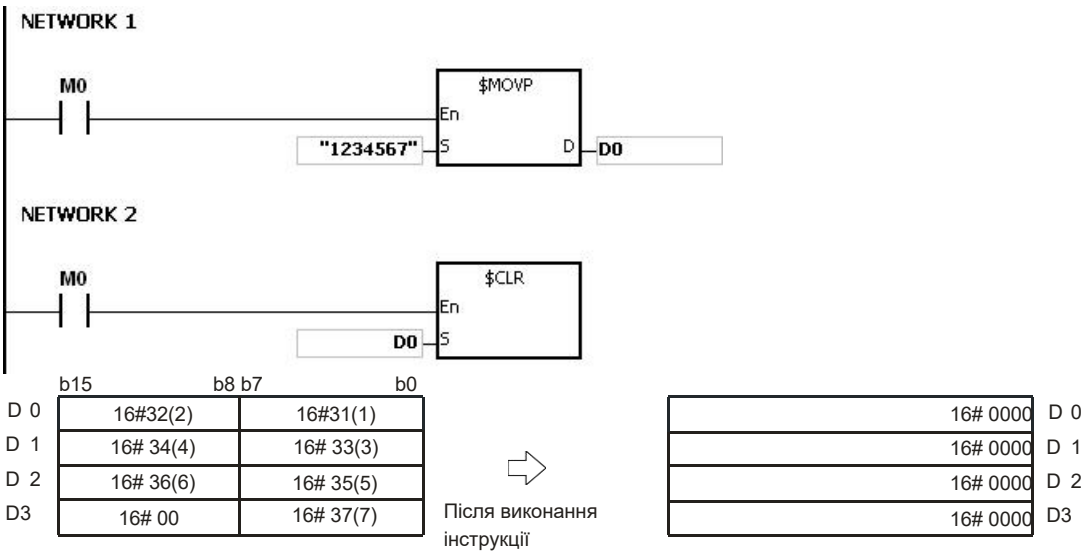
Пояснення

1. Ця інструкція очищає рядок у S . Якщо рядок у S не закінчується на 16#00, інструкція очищається до 255 персонажів.



приклад

Інструкція очищає рядок у D0, як показано нижче.



API	Код інструкції				Операнд						функція					
2119		\$INS	P	S ₁ , S ₂ , S ₃ , D						Вставка рядка						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●						○	
S ₂	●	●			●	●		●	●						○	
S ₃	●	●			●	●		●	●		○	○	○	○		
D		●			●	●		●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁		●				●							●
S ₂		●				●							●
S ₃		●				●							
D		●				●							

Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
AS		AS	-

СИМВОЛ

\$INS		\$INSP	
En		En	
S1	D	S1	D
S2		S2	
S3		S3	

- S_1 : Рядок
- S_2 : Рядок для вставки
- S_3 : Рядок S_2 вставляється в S_1 після символу, зазначеного значенням у S_3 .
- D : Пристрій, в якому зберігається результат виконання

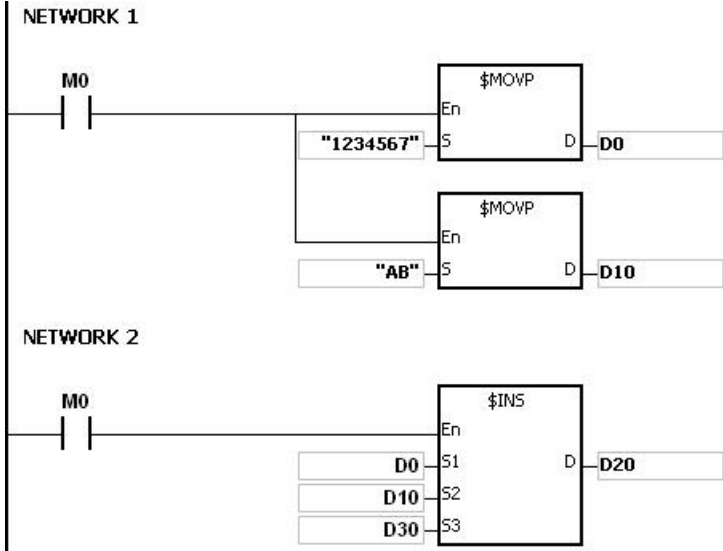
Пояснення

- Ця інструкція вставляє рядок у S_2 у рядок у S_1 після символу, зазначеного значенням у S_3 , і зберігає результат у D.
- Якщо рядок у S_1 або S_2 є нульовим рядком, інструкція зберігає інший рядок (S_1 або S_2), який не є нульовим рядком в D.
- Якщо рядки в S_1 і S_2 є нульовими рядками, інструкція зберігає 16#0000 у D.



приклад

Коли M0 увімкнено, дані в D0–D3 мають значення «1234567», а дані в D10 — «AB». Коли виконується інструкція \$INS, вона вставляє «AB» у рядок у D0–D3 після символу, зазначеного значенням у D30, і зберігає результат у D20–D24.



Якщо значення в D30 дорівнює 1, результат виконання буде таким.

	b15	b8	b7	b0		b15	b8	b7	b0	
D 0	16#32(2)		16#31(1)		Після виконання інструкції	16# 41(A)		16#31(1)		D 20
D 1	16# 34(4)		16# 33(3)			16# 32(2)		16# 42(B)		D 21
D 2	16# 36(6)		16# 35(5)			16# 34(4)		16# 33(3)		D 22
D 3	16# 00		16# 37(7)			16# 36(6)		16# 35(5)		D 23
						16#00		16# 37(7)		D 24
D 10	16# 42(B)		16# 41(A)							
D 11	Ігнорується		16#00							
D 30	1									

Якщо значення в D30 дорівнює 0, результат виконання буде таким.

	b15	b8	b7	b0		b15	b8	b7	b0	
D0	16#32(2)		16#31(1)		Після втрати	16#42(B)		16 #41(A)		D20
D1	16#34(4)		16#33(3)			16#32(2)		16 #31(1)		D21
D2	16#36(6)		16#35(5)			16#34(4)		16 №33(3)		D22
D3	16#00		16#37(7)			16# 36(6)		16 №35(5)		D23
						16#00		16 №37(7)		D24
D10	16#42(B)		16#41(A)							
D11	Ігнорується		16#00							
D30	0									

Додаткові зауваження

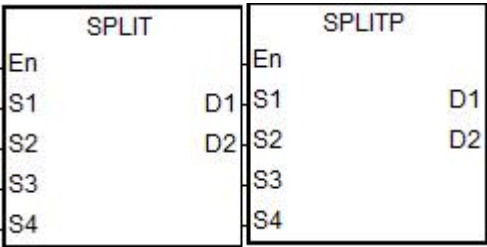
1. Якщо рядок у S_1 не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
2. Якщо рядок у S_2 не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
3. Якщо значення в S_3 менше 0 або якщо значення в S_3 більше, ніж довжина рядка в S_1 , SM0 увімкнено, а код помилки в SR0 — 16#2003.
4. Якщо вставлене значення в рядок (16#00, включаючи символ закінчення) у $S1$ перевищує 256, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції				Операнд						функція						
2122		SPLIT	P		$S_1 \cdot S_2 \cdot S_3 \cdot S_4 \cdot D_1 \cdot D_2$						Розбиття струни						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1								●	●								
S_2								●					○	○	○		
S_3								●					○	○	○		
S_4								●					○	○			
D_1								●									
D_2								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							●
S_3		●				●							●
S_4		●				●							
D_1		●				●							
D_2		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядний	
AS		AS		-	

СИМВОЛ



- S_1 : Рядок
- S_2 : Спліттер
- S_3 : Призначений символ закінчення розділювача
- S_4 : Зберігання довжини рядка після розділювача
- D_1 : Пристрій, у якому зберігається рядок після розділювача
- D_2 : Пристрій, в якому зберігається кількість розбитих струн

Пояснення

- Ця інструкція доступна лише для мікропрограми версії 1.04 або новішої.
- S_1 є джерелом рядка. Для символів у рядку – лише символи перед кінцевим символом або значення в S_3 використовуватиметься як дійсний рядок.
- S_2 є роздільником, який розділяє струни. S_3 — призначений кінцевий символ розділювача. Лише значення в нижніх 8- байт дійсні для S_2 і S_3 . Однак, якщо ви введете значення, відмінне від значення в молодших 8 байтах, ПЛК не надсилає повідомлення про помилку. Коли S_2 і S_3 використовують пристрої D для зберігання значень, використовуйте лише молодші 8 байт для значень у форматі ASCII. Значення в S_2 і S_3 можуть бути постійним числом, ASCII або рядком.
- S_4 - довжина збереження рядка після розділювача; одиниця: WORD. Кінцеві символи, наприклад 16#00 і 16#0000

слід включити в довжину. Максимальна кількість СЛІВ, які може містити рядок, становить S_4 , а WORD дорівнює 2 байтам. Кінцевий символ 16#00 займає байт. Наприклад, якщо S_4 дорівнює 2, максимальна кількість СЛІВ, які може містити рядок, становить 2 СЛОВА, а 2 СЛОВА дорівнюють 4 байтам. А це означає, що рядок може містити 3 символи та 1 кінцевий символ (16#00). Діапазон налаштувань для S_4 становить 1-100. (Якщо вхідне значення виходить за межі діапазону, значення буде розглядатися як мінімальне або максимальне під час роботи, і ПЛК не надсилає жодного повідомлення про помилку.)

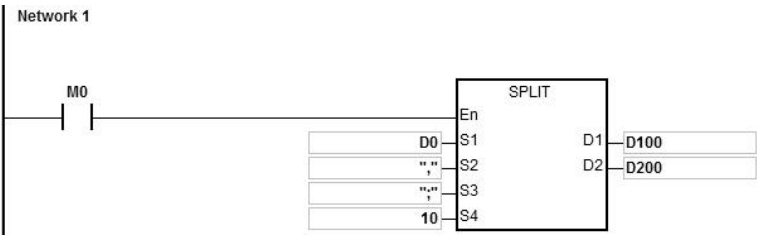
5. D_2 є сумою всіх розділених рядків.
6. Результатом $D_1 \sim D_1 + S_4 * D_2 - 1$ є кількість D пристроїв, які зайняті.
- Наприклад, якщо D_2 дорівнює 3, це означає, що 1-й рядок буде збережено з $D_1 + 0$, а 2-й рядок буде збережено в $D_1 + S_4$ а 3-й рядок буде збережено в $D_1 + S_4 * 2$.
7. Інструкція шукає рядок довжиною, визначеною в S_4 (WORD), доки не буде кінцевого символу або розділювача. додати 1 до значення в D_2 . Щоразу, коли знайдено один рядок,
8. S_1 можна зберегти до 256 символів, а в D_2 – до 20 наборів рядків. При будь-якому обмеженні досягнуто, навіть якщо немає кінцевого символу, поділ припиняється.

приклад

Використовуйте «.» як роздільник, щоб розділити рядок «abcd,1234,5.67,8910;ef» і використовувати «;» як кінцевий символ, який зберігається в D0-D11. I

зберігати результати в реєстрі, починаючи з D100 в одиниці 10 СЛІВ.

Примітка: рядок, який потрібно розділити, не містить символів «.».



Пояснення

1. Результати цього прикладу: D0~D11="abcd,1234,5.67,89103;edf".

пристрій	Значення	Рядок
D0	16#6261	"ба"
D1	16#6463	«DC»
D2	16#312C	«1»,
D3	16#3332	32
D4	16#2C34	“,4”
D5	16#2E35	«.5»
D6	16#3736	«76»
D7	16#382C	«8»,
D8	16#3139	«19»
D9	16#3330	«30»
D10	16#453B	"e,"
D11	16#4644	"fd"

2. Початковий контакт M0

3. Кількість розрізних струн D200=4

4. Розділені струни, як показано нижче.

1 -й набір: D100~D109 є зайнятий; D100-D101="abcd"	D100	D101	D102	D103	D104	D105	D106	D107	D108	D109
HEX	6261	6463	0000	0000	0000	0000	0000	0000	0000	0000
Рядок	"ба"	«DC»	16#0000	-	-	-	-	-	-	-
2 -й набір: D110~D119 є зайнятий; D110~D111=«1234»	D110	D111	D112	D113	D114	D115	D116	D117	D118	D119
HEX	3231	3433	0000	0000	0000	0000	0000	0000	0000	0000
Рядок	«21»	«43»	16#0000	-	-	-	-	-	-	-

3-й набір: D120~D129 зайняті; D120~D121=«5,67»

пристрій	D120	D121	D122	D123	D124	D125	D126	D127	D128	D129
HEX	2E35	3736	0000	0000	0000	0000	0000	0000	0000	0000
Рядок	«.5»	«76»	16#0000	-	-	-	-	-	-	-

4-й набір: D130~D139 зайняті, D130~D131=“89103”

пристрій	D130	D131	D132	D133	D134	D135	D136	D137	D138	D139
HEX	3938	3031	0033	0000	0000	0000	0000	0000	0000	0000
Рядок	«98»	«01»	Верхній: 3” Нижній: 16#00	-	-	-	-	-	-	-

Додаткові зауваження

S₁ можна зберегти до 256 символів, якщо немає кінцевого символу (16#00) або розділювача, інструкція виконуватиметься для 256 символів. SM0 не буде ввімкнено.

2. Кінцевий символ 16#00 або 16#0000 буде додано в рядок після його розділення (**D₁**). Дивіться приклад вище.

3. Кінцевий символ 16#00 або 16#0000 буде додано в рядок після його розділення (**D₁**), навіть якщо позаду немає рядка розділений рядок.

API	Код інструкції				Операнд								функція				
2123		MERGE	P		$S_1 \cdot S_2 \cdot S_3 \cdot D_1 \cdot D_2$								Об'єднання рядків				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1								●	●						○		
S_2								●	●						○		
S_3								●					○	○	○		
D_1								●									
D_2								●									
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1		●				●										●	
S_2		●				●										●	
S_3		●				●										●	
D_1		●				●											
D_2		●				●											
Інструкція пульсу								16-бітна інструкція				32-розрядний					
AS								AS				-					

СИМВОЛ

MERGE				MERGEP			
En				En			
S1		D1		S1		D1	
S2		D2		S2		D2	
S3				S3			

- S_1 : Джерело рядка 1
- S_2 : Джерело рядка 2
- S_3 : Спліттер
- D_1 : Пристрій, на якому зберігається об'єднаний рядок
- D_2 : Пристрій, в якому зберігається кількість об'єднаних рядків

Пояснення

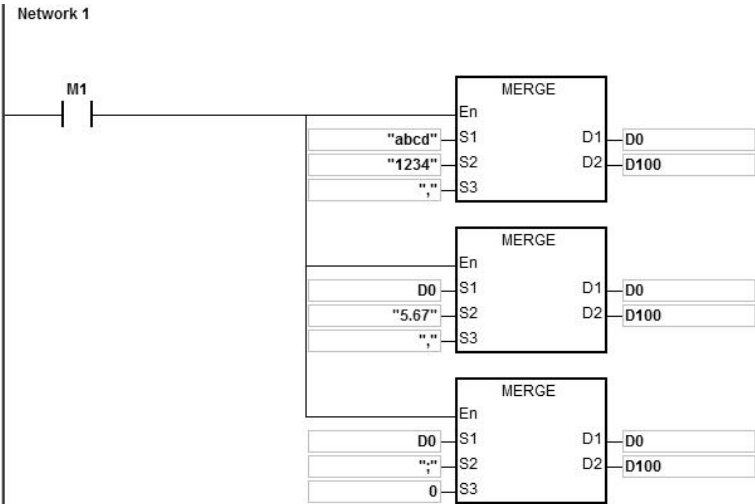
- Ця інструкція доступна лише для мікропрограми версії 1.04 або новішої.
- S_1 і S_2 є джерелом рядка 1 і 2 відповідно. S_3 - це роздільник, який розділяє струни. Лише значення в нижньому 8-байт дійсні для S_3 . S_1 безпосередньо об'єднує S_2 , коли S_3 дорівнює 0 (16#00). S_1 додає роздільник, а потім об'єднує S_2 , коли S_3 не дорівнює 0.
- D_1 — місце для зберігання об'єданого рядка. Порядок об'єднання: рядок 1 + роздільник + рядок 2 + кінцевий символ 0 (16#00).

Ви можете самостійно визначити кінцевий символ. Цей самовизначений кінцевий символ буде додано перед офіційним, 0 (16#00).
- Якщо всі значення в S_1 , S_2 , S_3 дорівнюють 0, D_1 дорівнює 16#0000.
- D_2 — це сума всіх об'єднаних рядків (за винятком символу закінчення 16#00). Накопичена довжина рядка становить довжина рядка 1 або довжина рядка 1 + роздільник + довжина рядка 2.

6. Якщо джерела рядків для S_1 і S_2 дорівнюють 0 або коли рядки об'єднано, довжина об'єданого рядка становить до 255 слів (за винятком символу закінчення 16#00). Наприклад, є 250 слів у рядку 1 і 100 слів у рядку 2. Після їх об'єднання довжина об'єданого рядка становить 250 слів із рядка 1 + роздільник *1 + перші 4 слова з рядка 2.

приклад

Використовуйте «,» як роздільник, щоб розділити рядок, і використовуйте «;» як кінцевий символ, а потім об'єднайте 3 рядки «abcd», «1234» і «5.67».



І збережить результати «abcd,1234,5.67;» D100=15, в регістрі починаючи з D0-D7.

6

Пояснення

1. Початковий контакт M0
2. 1-ша програма: об'єднайте 1-й рядок «abcd» і 2-й рядок «1234» і помістіть між ними роздільник «,» і потім збережіть результат у регістрах, починаючи з D0.

Об'єднаний результат: D0~D4= «abcd,1234»; кількість об'єднаних рядків: D100=9

пристрій	D0	D1	D2	D3	D4
HEX	6261	6463	312C	3332	0034
Рядок	"ба"	«DC»	«1»,	«32»	Старший 8-байт: 16#00 ; молодший 8-байт: 4

3. 2-га програма: об'єднайте 1-й рядок D0=«abcd,1234» і 2-й рядок «5.67» разом і помістіть між ними роздільник «,» і потім збережіть результат у регістрах, починаючи з D0.

Об'єднаний результат: D0~D7= «abcd,1234,5.67»; кількість об'єднаних рядків: D100=14

пристрій	D0	D1	D2	D3	D4	D5	D6	D7
HEX	6261	6463	312C	3332	2C34	2E35	3736	0000
Рядок	"ба"	«DC»	«1»,	«32»	“,4”	«.5»	«76»	16#0000

4. 3-тя програма: об'єднайте 1-й рядок D0="abcd,1234,5.67" і 2-й рядок "","" разом і поставте роздільник "0" між ними (об'єднайте рядок 1 і 2 безпосередньо), а потім збережіть результат в регістрах, починаючи з D0.

Об'єднаний результат: D0~D7="abcd,1234,5.67;"; кількість об'єднаних рядків: D100=15.

пристрій	D0	D1	D2	D3	D4	D5	D6	D7
HEX	6261	6463	312C	3332	2C34	2E35	3736	003B
Рядок	"ба"	«DC»	«1»,	«32»	“,4”	«.5»	«76»	Старший 8-байт: 16#00 ; молодший 8-байт: “,”

Додаткові зауваження

1. Якщо рядок у S_1 не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
2. Якщо рядок у S_2 не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 — 16#200E.
3. Якщо $D_1 + (D_2 + 1)/2 - 1$ перевищує діапазон пристрою, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
4. За об'єднаним рядком буде додано кінцевий символ 16#00 або 16#0000. Дивіться приклади вище.

6.22 Інструкції по Ethernet

6.22.1 Список інструкцій Ethernet

У наведеній нижче таблиці наведено інструкції з обробки рядків, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
2200	SOPEN	–	✓	Відкриття розетки
2201	SSEND	–	✓	Надсилання даних через сокет
2203	SCLOSE	–	✓	Закриття розетки
2204	MSEND	–	✓	Надсилання електронного листа
2206	INTOA	–	✓	Перетворення цілої IP-адреси в рядкову IP-адресу
2207	ЯТОН	–	✓	Перетворення рядкової IP-адреси в цілу IP-адресу
2208	EIPRW	–	–	Читання та запис даних EtherNet/IP
2209	SCONF	–	✓	Налаштування параметрів сокета TCP/UDP
2210	MCONF	–	✓	Читання та запис даних Modbus TCP
2211	EMCONF1	–	✓	Налаштування значень параметрів сервера електронної пошти
2212	EMCONF2	–	✓	Встановлення електронної адреси
2214	MQTT_Connect	–	–	Встановлення підключення MQTT
2215	MQTT_Publish	–	–	Публікація повідомлення MQTT
2216	MQTT_Subscribe	–	–	Підписка на тему MQTT

6.22.2 Пояснення інструкцій Ethernet

API	Код інструкції			Операнд								функція					
2200		SOPEN	P	S_1, S_2, S_3								Відкриття розетки					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●		●	●			○	○	○			
S_2	●	●			●	●		●	●			○	○	○			
S_3	●	●			●	●		●	●			○	○	○			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1		●				●											
S_2		●				●											
S_3		●				●											
				Інструкція пульсу				16-бітна інструкція				32-розрядна інструкція					
				AS				AS				-					

СИМВОЛ

SOPEN		SOPENP	
En		En	
S1		S1	
S2		S2	
S3		S3	

S_1 : Режим розетки
 S_2 : Номер розетки
 S_3 : Режим зв'язку

Пояснення

- Ця інструкція відкриває сокет. Встановіть S_1 на 1, щоб відкрити TCP-сокет, і S_1 на 0, щоб відкрити UDP-сокет.
- S_2 - номер розетки. Він повинен відповідати номеру Ethernet-розетки, встановленому в розширених налаштуваннях; до 4 можна встановити.
- ПЛК серії AS функціонує як клієнт і надсилає запит TCP-з'єднання ($S_1 = 1$) на сервер, якщо S_3 дорівнює 1 або 3 (доступно для V1.06.30 або новішої версії), а ПЛК серії AS функціонує як сервер і очікує на запит TCP-з'єднання ($S_1 = 1$) від клієнта, якщо S_3 дорівнює 0 або 2 (доступно для V1.06.30 або новішої версії).
- Щоб розпочати з'єднання UDP ($S_1 = 0$), встановіть S_3 на 0~3.
- Операнд S_1 має бути 0 або 1 (за замовчуванням: 0); операнд S_2 має бути між 1–4 (за замовчуванням: 1); операнд S_3 має бути 0 або 1 (за замовчуванням: 0). У разі перевищення діапазону на ПЛК відображається код помилки H'2003.
- Перед використанням інструкції встановіть такі параметри в HWCONFIG в ISPSOft.
 - Налаштування параметрів PLC → Ethernet-Basic → Налаштування IP-адреси та адреси маски мережі
 - Налаштування параметрів PLC → Ethernet-Advanced → Socket → Enable Socket Function

- Налаштування параметрів PLC → Ethernet-Advanced → Socket → TCP/UDP Socket Connection and Setting sockets to use

7. Зверніться до інструкцій SCONF (API 2209), щоб дізнатися більше про параметри сокетів TCP і UDP.

8. Якщо вам потрібно надіслати кілька даних окремо, ви можете скористатися цією новою функціональністю: $S_3 = 2$ або 3 (доступно для версії 1.06.30 або новішої). Перший пристрій у пункті призначення зберігає довжину даних, які потрібно отримати, а дані, які потрібно отримати, зберігаються з другого пристрою й далі. Якщо вам потрібно отримати інші дані, ви можете очистити значення в першому пристрої та почати іншу передачу даних.

9. Коли відкрито сокет TCP ($S_1 = 1$), використовуйте такі параметри для S_3 , IP-адреси сокета та портів зв'язку.

Режим запуску	S_3	Віддалений IP	Місцевий Комунікаційний порт	Дистанційний Комунікаційний порт	опис
Клієнт	1 або 3	Конкретний IP адресу	Не дорівнює 0	Не дорівнює 0	Вказує IP-адресу, локальну та порт віддаленого зв'язку
		Конкретний IP адресу	Не дорівнює 0	0	Вказує IP-адресу, але не локальну комунікаційний порт
		Будь-яке значення	0	Будь-яке значення	Незаконний; коли ЦП ПЛК перебуває в режимі клієнта, порт зв'язку з сервером і IP-адреса повинні бути призначені.
		0.0.0.0	Будь-яке значення	Будь-яке значення	
Сервер	0 або 2	Конкретний IP адресу	0	Не дорівнює 0	Вказує IP-адресу, але не віддалену комунікаційний порт
		Конкретний IP адресу	Не дорівнює 0	Не дорівнює 0	Вказує IP-адресу та локальну адресу комунікаційний порт
		0.0.0.0	0	Не дорівнює 0	Не вказувати IP-адресу та локальну комунікаційний порт
		0.0.0.0	Не дорівнює 0	Не дорівнює 0	Не вказати IP-адресу, а вказати порт віддаленого зв'язку
		Будь-яке значення	Будь-яке значення	0	Незаконний; коли ЦП ПЛК знаходиться на сервері, має бути призначений комунікаційний порт сервера.

Примітка. Нові режими 2 і 3 доступні для V1.06.30 або новішої версії. У нових режимах використовується перший пристрій у пункті призначення

для збереження довжини отриманих даних. Ви можете використовувати значення тут, щоб визначити, чи завершено отримання. Для режимів 0 і 1 довжина даних не записується.

10. Якщо дані надсилаються через сокет TCP і після виконання інструкції не виникає жодної помилки, сокет починає встановлювати з'єднання з віддаленим пристроєм, а прапор для початку з'єднання ввімкнено. Якщо з'єднання виконано успішно, прапор успішного з'єднання ввімкнуто, а прапор початку з'єднання – вимкнено. Якщо виникає помилка, прапор помилки ввімкнено.

ТСР Номер розетки	Підключення розпочато	Дані отримано	Дані надіслано	Підключення починається	Підключення закрито	Надсилан ня даних	Прапор помилки
1	SM1270	SM1271	SM1272	SM1273	SM1274	SM1275	SM1277
2	SM1278	SM1279	SM1280	SM1281	SM1282	SM1283	SM1285
3	SM1286	SM1287	SM1288	SM1289	SM1290	SM1291	SM1293
4	SM1294	SM1295	SM1296	SM1297	SM1298	SM1299	SM1301

11. Якщо дані передаються через сокет UDP і після виконання інструкції не виникає помилка, прапор для розпочате з'єднання ввімкнено. Якщо виникає помилка, прапор помилки ввімкнено.

Номер UDP- сокета	Підключення розпочато	Дані отримано	Надсилання даних	Підключення ЗАЧИНЕНО	Прапор помилки
1	SM1334	SM1335	SM1336	SM1337	SM1338
2	SM1339	SM1340	SM1341	SM1342	SM1343
3	SM1344	SM1345	SM1346	SM1347	SM1348
4	SM1349	SM1350	SM1351	SM1352	SM1353

12. Якщо дані передаються через сокет TCP у режимі сервера або через сокет UDP, не дозволяється використовувати наступний номер порту зв'язку для ПЛК серії AS.

Тип розетки	Номер порту зв'язку	Протокол зв'язку
TCP	80	HTTP
TCP	502	MODBUS TCP
TCP	44818	EtherNet/IP
UDP	68	DHCP/BOOTP
UDP	2222	EtherNet/IP
UDP	44818	EtherNet/IP
UDP	20006	Самовизначений протокол для серії AS

13. Загалом використовується імпульсна інструкція SOPENP.

Приклад 1 ($S_3 = 1$)

1. У цьому прикладі показано, як встановити TCP-з'єднання між комп'ютером як сервером і ПЛК серії AS як клієнтом.



2. ISPSOft → HWCONFIG (Ethernet-Basic)

Конфігурація Ethernet	
Режим IP-адреси	Статичний
IP-адреса	192.168.1.111
Адреса маски мережі	255.255.255.0
Адреса шлюзу	192.168.1.1

3. a) ISPSOft → HWCONFIG (Ethernet-Advanced → Socket → 1st TCP Socket Connection)

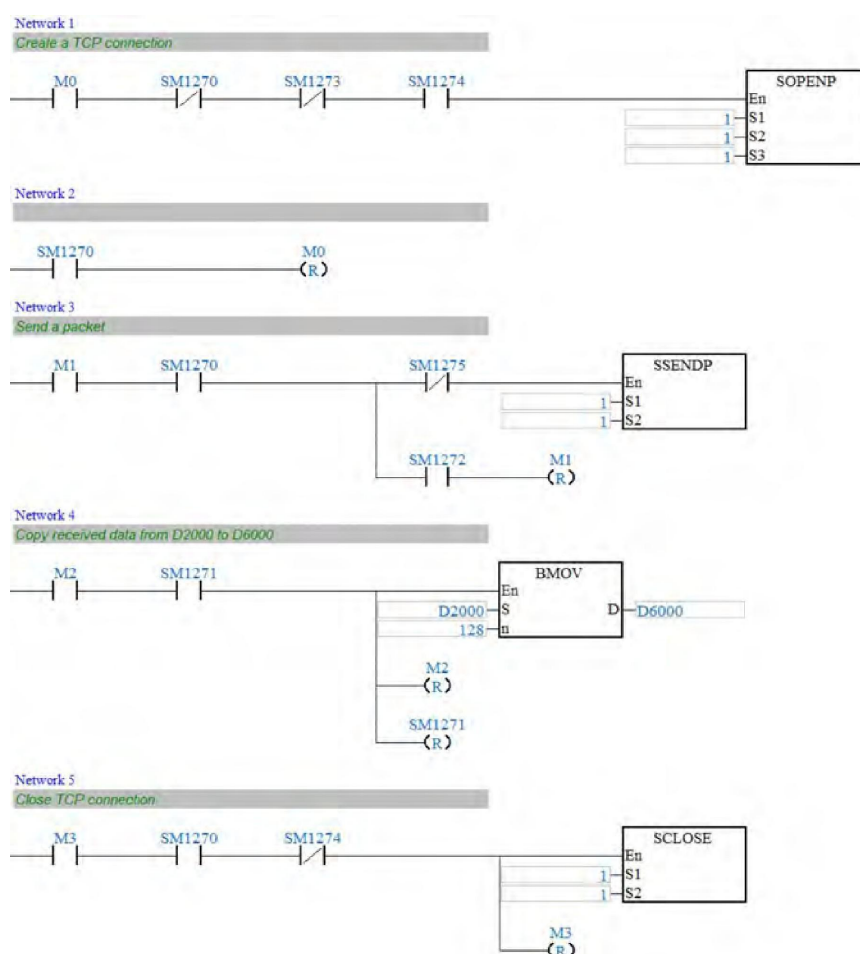
б) Інструкція SCONF

1-е з'єднання TCP-сокет	
Віддалений IP	192.168.1.110
Віддалений порт	3201
Місцевий порт	3200
Надіслати адресу даних	D1000
Визначена довжина даних для надсилання #1	200
Адреса отримання даних	D2000
Визначена довжина даних для отримання #1	200
Таймер Keep Alive	10

#1 : Ця довжина даних визначається налаштуваннями в інструкції SCONF; для V1.06.20 або попередніх версій: 512 байт доступний і для V1.06.30 або новіших версій: доступно 4096 байт.

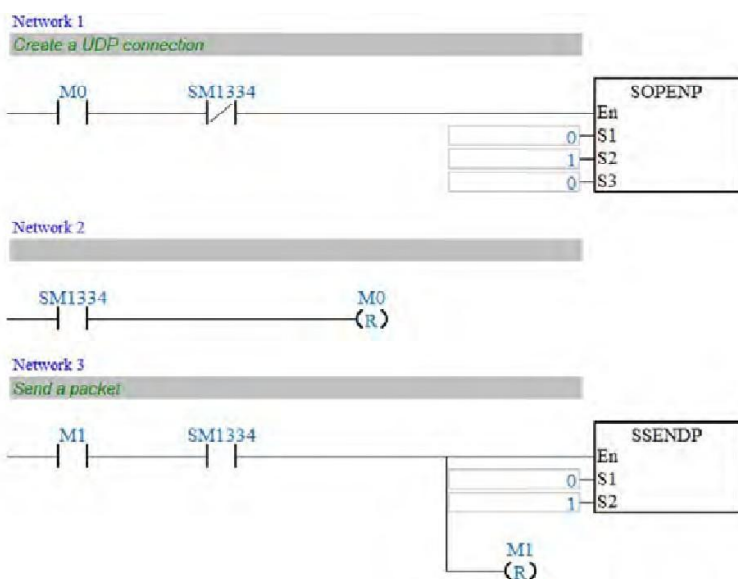
4. Спочатку встановіть параметри, пов'язані з TCP Socket1, у HWCONFIG у ISPSOft або в інструкції SCONF.

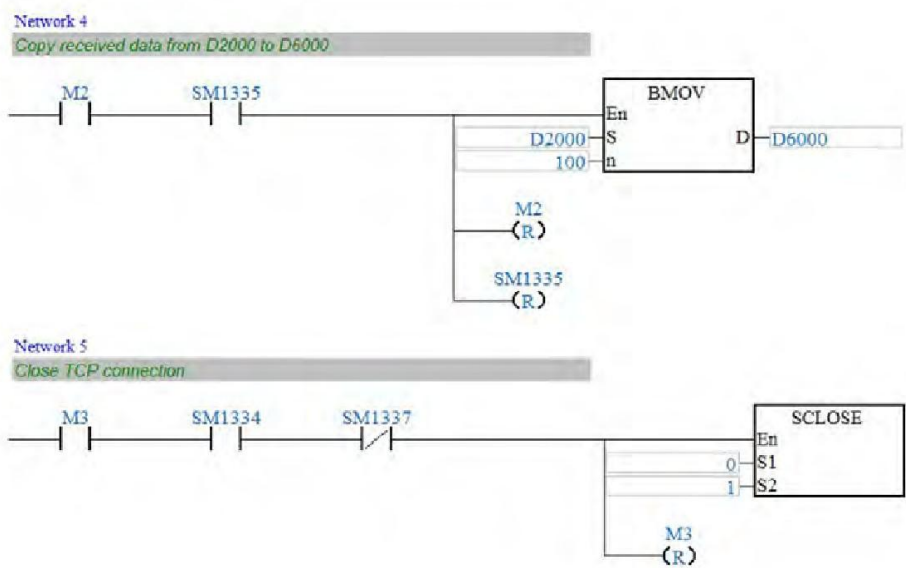
5. Встановлення з'єднання (Мережа 1): Коли M0 увімкнено, система перевіряє, чи розетка закрита (SM1274 = ON), НЕ підключено (SM1270 = OFF) або НЕ підключено (SM1273 = OFF). Якщо розетка не закрита, не підключалася і не підключається, система виконує процедуру підключення. Після підключення розетки (SM1270 = ON), M0 перемикається в положення OFF.
6. Надсилання даних (Мережа 3): коли M1 увімкнено, система перевіряє, чи підключено розетку (SM1270) = ON), і дані ще не надсилаються (SM1275 = OFF). Якщо сокет підключено, а дані не надсилаються, система починає надсилати дані. Якщо розетка не підключена, система не виконує інструкцію. Після надсилання даних (SM1272 = ON), M1 перемикається в положення OFF.
7. Отримання даних (Мережа 4): коли M2 увімкнено, система перевіряє, чи підключено розетку та чи встановлено прапор (SM1271) для отриманих даних. Якщо розетка була підключена і прапор для отримання даних увімкнено, то дані вже отримані. Але це НЕ обов'язково означає, що він досяг максимальної тривалості отримання. Після отримання даних (SM1271 = ON), M2 і SM1271 перемикаються в положення OFF.
8. Закриття з'єднання (Мережа 5): Коли M3 увімкнено, система перевіряє, чи під'єднано розетку. Якщо розетка підключена (SM1270 = ON), а з'єднання ще не закрито (SM1274 = OFF), система збирається закрити з'єднання. Якщо розетка не підключена, система не виконує інструкцію. Після закриття з'єднання (SM1274 = ON), M3 перемикається в положення OFF.



Приклад 2 ($S_3 = 0$)

- Цей приклад ілюструє, як встановити з'єднання UDP між комп'ютером і ПЛК серії AS.
 - ISPSOft → HWCONFIG (Ethernet-Advanced → Socket → 1st UDP Socket Connection)
 - Інструкція SCONF
 - Зверніться до пункту 3 прикладу 1, щоб налаштувати підключення.
- Встановлення з'єднання (Мережа 1): Коли M0 увімкнено, система перевіряє, чи розетка закрита (SM1334 = OFF), а потім виконує процедуру підключення. Після підключення розетки (SM1334 = ON), M0 перемикається в положення OFF.
- Надсилання даних (Мережа 3): коли M1 увімкнено, система перевіряє, чи підключено розетку (SM1334 = увімкнено), а потім система починає надсилати дані. Після надсилання даних M1 вимикається.
- Отримання даних (Мережа 4): коли M2 увімкнено, система перевіряє, чи роз'єм підключено та прапор (SM1335) для отриманих даних увімкнено. Якщо так, то дані вже отримані. Але це НЕ обов'язково означає, що він досяг максимальної тривалості отримання. Після отримання даних (SM1335 = ON), M2 і SM1335 перемикаються в положення OFF.
- Закриття з'єднання (Мережа 5): коли M3 увімкнено, система перевіряє, чи був роз'єм, який потрібно закрити, підключено (SM1334 = ON), і якщо з'єднання ще не закрито (SM1337 = OFF). Якщо так, система збирається закрити з'єднання. Якщо розетка не підключена, система не виконує інструкцію. Після закриття з'єднання M3 вимикається.





Приклад 3 (S₃ = 2)

- 1. Цей приклад ілюструє, як сервер отримує дані про 2K bytes від TCP-з'єднання в режимі S₃ = 2.
- 2. Інструкція SCONF

1 -е з'єднання TCP-сокет	
Віддалений IP	192.168.1.106
Віддалений порт	0
Місцевий порт	2020
Надіслати адресу даних	D10000
Визначена довжина даних для надсилання #1	4096
Адреса отримання даних	D15000
Визначена довжина даних для отримання #1	4096
Таймер Keep Alive	30

#1 : Ця довжина даних визначається налаштуваннями в інструкції SCONF; для V1.06.20 або попередніх версій: 512 байт доступний і для V1.06.30 або новіших версій: доступно 4096 байт.

- 3. Коли ЦП ПЛК переходить зі стану STOP на RUN, виконайте інструкцію SCONF. Зверніться до API 2209 SCONF для налаштування сокета.
- 4. Встановлення з'єднання (Мережа 3); Коли M0 увімкнено, система перевіряє, чи розетка закрита (SM1274 = ON), НЕ підключена (SM1270 = OFF) або НЕ підключена (SM1273 = OFF). Якщо розетка не закрита, не підключалася і не підключається, система виконує процедуру підключення. Після підключення розетки (SM1270 = ON), M0 перемикається в положення OFF.

5. Підключення до віддаленого клієнта (IP = 192.168.1.106) підключено та надсилає 2 КБ даних.
6. Відкрийте таблицю моніторингу та почніть моніторинг початкового пристрою, де зберігається довжина отриманих даних.

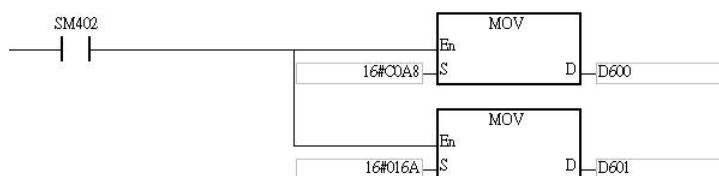
D15000		2000
--------	--	------

Якщо отримано 2 КБ даних, ви побачите, що значення в D15000 дорівнює 2000. Якщо вам потрібно почати інші дані

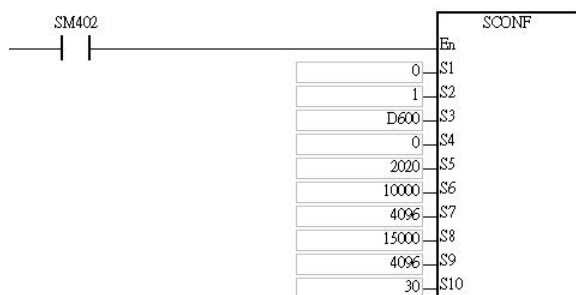
передачі, ви можете встановити значення в D15000 на 0, щоб почати інший підрахунок довжини даних.

7. Закриття з'єднання (Мережа 5): Коли М3 увімкнено, система перевіряє, чи роз'єм під'єднано. Якщо розетка була підключена (SM1270 = ON), система збирається закрити підключення. Якщо розетка не підключена, система не виконує інструкцію. Після закриття з'єднання М3 вимикається.

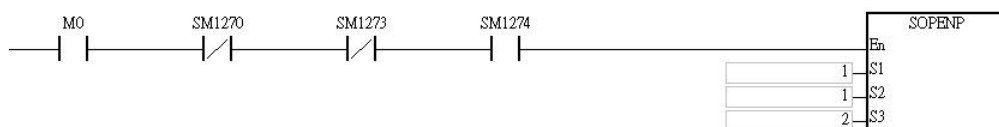
Network 1
Set TCP Client IP



Network 2
Set Sconfig



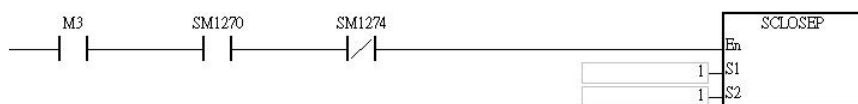
Network 3
Create TCP Connection



Network 4



Network 5
Close TCP connection



Додаткові зауваження

1. У наведеній нижче таблиці перераховано помилки підключення TCP.

Код помилки (SR180)	Прапор помилки	опис
16 № 2003	SM0	S_1 , S_2 або S_3 перевищує діапазон
16#600C	SM1109	Локальний сокет не використовується
16#600D	SM1100	Мережа Ethernet не підключена
16#6200	Прапор помилки TCP-сокета	Незаконна IP-адреса
16#6201	Прапор помилки TCP-сокета	Незаконне налаштування режиму зв'язку TCP-сокета
16#6202	Прапор помилки TCP-сокета	Незаконне налаштування режиму сокета TCP
16#6203	Прапор помилки TCP-сокета	Незаконна адреса для надсилання даних
16#6204	Прапор помилки TCP-сокета	Довжина надісланих даних перевищує діапазон
16#6205	Прапор помилки TCP-сокета	Пристрій вихідних даних перевищує діапазон
16#6206	Прапор помилки TCP-сокета	Незаконна адреса для отримання даних
16#6207	Прапор помилки TCP-сокета	Довжина отриманих даних перевищує діапазон
16#6208	Прапор помилки TCP-сокета	Пристрій даних цілі перевищує дальність
16#6212	Прапор помилки TCP-сокета	Тайм-аут зв'язку сокета TCP
16#6213	Прапор помилки TCP-сокета	Розмір отриманих даних перевищує набір даних
16#6214	Прапор помилки TCP-сокета	Віддалене обладнання відхилило підключення через сокет TCP
16#6215	Прапор помилки TCP-сокета	Сокет TCP не підключений
16#6217	Прапор помилки TCP-сокета	Створено з'єднання TCP-сокета
16#6218	Прапор помилки TCP-сокета	Надсилання даних через TCP-сокет активовано
16#621A	Прапор помилки TCP-сокета	Вимкнено з'єднання TCP-сокета

2. У наступній таблиці пояснюються помилки підключення UDP.

Код помилки (SR180)	Прапор помилки	опис
16 № 2003	SM0	S_1 , S_2 або S_3 перевищує діапазон
16#600C	SM1109	Використовується локальна розетка
16#600D	SM1100	Мережа Ethernet не підключена
16#6209	Прапор помилки UDP-сокета	Незаконна IP-адреса
16#620A	Прапор помилки UDP-сокета	Незаконний режим спілкування
16#620C	Прапор помилки UDP-сокета	Незаконна адреса для надсилання даних
16#620D	Прапор помилки UDP-сокета	Довжина надісланих даних перевищує діапазон
16#620E	Прапор помилки UDP-сокета	Пристрій вихідних даних перевищує діапазон
16#620F	Прапор помилки UDP-сокета	Незаконна адреса для отримання даних
16#6210	Прапор помилки UDP-сокета	Довжина отриманих даних перевищує діапазон
16#6211	Прапор помилки UDP-сокета	Пристрій даних цілі перевищує дальність
16#6213	Прапор помилки UDP-сокета	Розмір отриманих даних перевищує набір даних
16#6215	Прапор помилки UDP-сокета	Сокет UDP не підключено
16#6217	Прапор помилки UDP-сокета	Створено з'єднання через сокет UDP
16#6218	Прапор помилки UDP-сокета	Активовано надсилання даних через сокет UDP
16#621A	Прапор помилки UDP-сокета	Вимкнення з'єднання через сокет UDP

3. Якщо клієнт і сервер є ПЛК серії AS і параметри часу очікування зв'язку однакові,

Сервер автоматично розриває з'єднання, якщо спочатку на сервері настане тайм-аут. У цьому випадку прапор помилки на клієнті не ввімкнено. З іншого боку, якщо тайм-аут спочатку виникає на клієнті, прапор помилки на клієнті ввімкнено, і з'єднання розривається.

API		Код інструкції			Операнд						функція					
2201			SSEND	P	S ₁ , S ₂						Надсилання даних через сокет					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁	●	●			●	●		●	●			○	○	○		
S ₂	●	●			●	●		●	●			○	○	○		

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

SSEND	SSENDP
En	En
S1	S1
S2	S2

 S_1 : Режим розетки S_2 : Номер розетки

Пояснення

- Ця інструкція надсилає дані через сокет, визначений S_1 . Встановіть S_1 на 1, щоб відкрити TCP-сокет, і S_1 на 0, щоб відкрити UDP-сокет.
- S_2 - номер розетки. Він повинен відповідати номеру Ethernet-сокета; можна встановити до 4.
- Операнд S_1 має бути 0 або 1 (за замовчуванням: 0); операнд S_2 має бути між 1–4 (за замовчуванням: 1). У разі перевищення діапазону на ПЛК відображається код помилки H'2003.
- Перед використанням цієї інструкції скористайтеся інструкцією SOPEN (API 2200), щоб відкрити сокет. Якщо прапор успішного підключення TCP-сокета ON, або прапор початку підключення UDP-сокета ON, використовуйте цю інструкцію.
- Якщо дані надсилаються через TCP-сокет і після виконання інструкції не виникає помилка, дані надсилаються, а прапор для надсилання даних увімкнено. Якщо дані надіслано успішно, прапорець для надісланих даних увімкнено, а прапорець для надсилання даних – OFF. Якщо виникає помилка, прапор помилки ввімкнено.

Номер TCP-сокета	Надсилання даних	Дані надіслано	Прапор помилки
1	SM1275	SM1272	SM1277
2	SM1283	SM1280	SM1285
3	SM1291	SM1288	SM1293
4	SM1299	SM1296	SM1301

6. Якщо дані надсилаються через сокет UDP і після виконання інструкції не виникає жодної помилки, прапор для надісланих даних увімкнено. Якщо виникає помилка, прапор помилки ввімкнено.

Номер UDP-сокета	Надсилення даних	Прапор помилки
1	SM1336	SM1338
2	SM1341	SM1343
3	SM1346	SM1348
4	SM1351	SM1353

7. Загалом використовується імпульсна інструкція SSENDP.

приклад

Зверніться до прикладу в інструкціях SOPEN (API 2200).

Додаткові зауваження

1. У наступній таблиці пояснюються помилки підключення TCP.

Код помилки	Прапор помилки	опис
16 № 2003	SM0	S_1 або S_2 перевищує діапазон
16#600D	SM1100	Мережа Ethernet не підключена
16#6202	Прапор помилки TCP-сокета	Незаконне налаштування режиму сокета TCP
16#6203	Прапор помилки TCP-сокета	Незаконна адреса для надсилення даних
16#6204	Прапор помилки TCP-сокета	Довжина надісланих даних перевищує діапазон
16#6205	Прапор помилки TCP-сокета	Пристрій вихідних даних перевищує діапазон
16#6212	Прапор помилки TCP-сокета	Тайм-аут підключення TCP-сокета
16#6214	Прапор помилки TCP-сокета	Віддалене обладнання відхилило підключення через сокет TCP
16#6215	Прапор помилки TCP-сокета	Сокет TCP не підключений
16#6218	Прапор помилки TCP-сокета	Розпочато надсилення даних

2. У наступній таблиці пояснюються помилки підключення UDP.

Код помилки	Прапор помилки	опис
16 № 2003	SM0	S_1 або S_2 перевищує діапазон
16#600D	SM1100	Мережа Ethernet не підключена
16#620A	Прапор помилки UDP-сокета	Незаконне налаштування режиму зв'язку UDP-сокета
16#620C	Прапор помилки UDP-сокета	Незаконна адреса для надсилання даних
16#620D	Прапор помилки UDP-сокета	Довжина надісланих даних перевищує діапазон
16#620E	Прапор помилки UDP-сокета	Пристрій вихідних даних перевищує діапазон
16#6218	Прапор помилки UDP-сокета	Розпочато надсилання даних

API	Код інструкції				Операнд						функція						
2203		SCLOSE	P		S_1, S_2						Закриття розетки						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●		●	●			○	○	○			
S_2	●	●			●	●		●	●			○	○	○			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1		●				●											
S_2		●				●											
							Інструкція пульсу		16-бітна інструкція				32-розрядна інструкція				
СИМВОЛ							AS		AS				-				

SCLOSE				SCLOSEP			
En				En			
S1				S1			
S2				S2			

S_1 : Режим розетки

S_2 : Номер розетки

Пояснення

- Ця інструкція закриває сокет, визначений S_1 . Встановіть S_1 на 0, щоб закрити TCP-сокет, і S_1 на 1, щоб закрити UDP-сокет.
- S_2 - номер розетки. Він повинен відповідати номеру Ethernet-сокета; можна встановити до 4.
- Операнд S_1 має бути 0 або 1 (за замовчуванням: 0); операнд S_2 має бути між 1–4 (за замовчуванням: 1). У разі перевищення діапазону на ПЛК відображається код помилки H'2003.

Перед закриттям розетки переконайтеся, що розетка підключена, інакше інструкція не виконується.

- Якщо клієнт закриває сокет TCP, сервер продовжує підключатися до порту локального зв'язку (прапор успішного підключення ввімкнено). Якщо сервер закриває TCP-сокет, сервер більше не підключається до порту локального зв'язку після виконання інструкції. Коли TCP-сокет закритий, відповідні прапорці ВИМКНЕНІ.
- Після виконання інструкції для закриття UDP-сокета відповідні прапорці OFF.
- Якщо TCP-сокет закритий і після виконання інструкції не виникає жодної помилки, з'єднання з віддаленим пристроєм починає закриватися, а прапорець для закриття з'єднання ввімкнено. Якщо з'єднання успішно закрито, прапорець для закриття з'єднання OFF. Якщо виникає помилка, прапор помилки ввімкнено.

Номер TCP-сокета	Підключення закрито	Прапор помилки
1	SM1274	SM1277
2	SM1282	SM1285
3	SM1290	SM1293
4	SM1298	SM1301

8. Якщо UDP-сокет закритий і після виконання інструкції не виникає жодної помилки, прапор для розпочатого з'єднання OFF. Якщо виникає помилка, прапор помилки ввімкнено.

Номер UDP-сокета	Прапор помилки
1	SM1338
2	SM1343
3	SM1348
4	SM1353

9. Загалом використовується імпульсна інструкція SCLOSEP.

приклад

Зверніться до прикладу інструкції SOPEN (API 2200).

Додаткові зауваження

1. У наведеній нижче таблиці перераховано помилки підключення TCP.

Код помилки	Прапор помилки	опис
16 № 2003	SM0	S_1 або S_2 перевищує діапазон
16#600D	SM1100	Мережа Ethernet не підключена
16#6212	Прапор помилки TCP-сокета	Тайм-аут зв'язку сокета TCP
16#6214	Прапор помилки TCP-сокета	Віддалений пристрій відхилив підключення через сокет TCP
16#621A	Прапор помилки TCP-сокета	Вимкнуто з'єднання TCP-сокета

2. У наступній таблиці пояснюються помилки підключення UDP.

Код помилки	Прапор помилки	опис
16 № 2003	SM0	S_1 або S_2 перевищує діапазон
16#600D	SM1100	Мережа Ethernet не підключена
16#621A	Прапор помилки UDP-сокета	Вимкнення з'єднання через сокет UDP

API	Код інструкції			Операнд							функція						
2204		MSEND	P	S_1, S_2, S_3, D							Надсилання електронного листа						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●			O	O	O		
S_2	●	●			●	●		●	●							
S_3	●	●			●	●		●	●							
D	●	●	●	●				●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
D	●												

СИМВОЛ

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

MSEND				MSENDP			
En				En			
S1			D	S1			D
S2				S2			
S3				S3			

- S_1 : Віддалений номер електронної пошти
- S_2 : Тема електронного листа
- S_3 : Текст електронної пошти
- D : Завершення інструкції

Пояснення

- Ця інструкція створює та надсилає електронний лист, встановивши S_1 , S_2 та S_3 .
- Перед використанням інструкції встановіть наступні значення в ISPSOft.
 - Налаштування параметрів PLC → Ethernet-Basic → Налаштування IP-адреси та адреси маски мережі
 - Налаштування параметрів ПЛК → Ethernet-Advanced → Електронна пошта → Налаштування SMTP-сервера, порту, локальної електронної адреси, і тему SMTP
 - Налаштування параметрів ПЛК → Ethernet-Advanced → Конфігурація електронної пошти та тригера → Налаштування адреси електронної пошти
 - Якщо потрібна ідентифікація облікового запису: PLC Parameter Setting → Ethernet-Advanced → Email → Setting user
ім'я та пароль

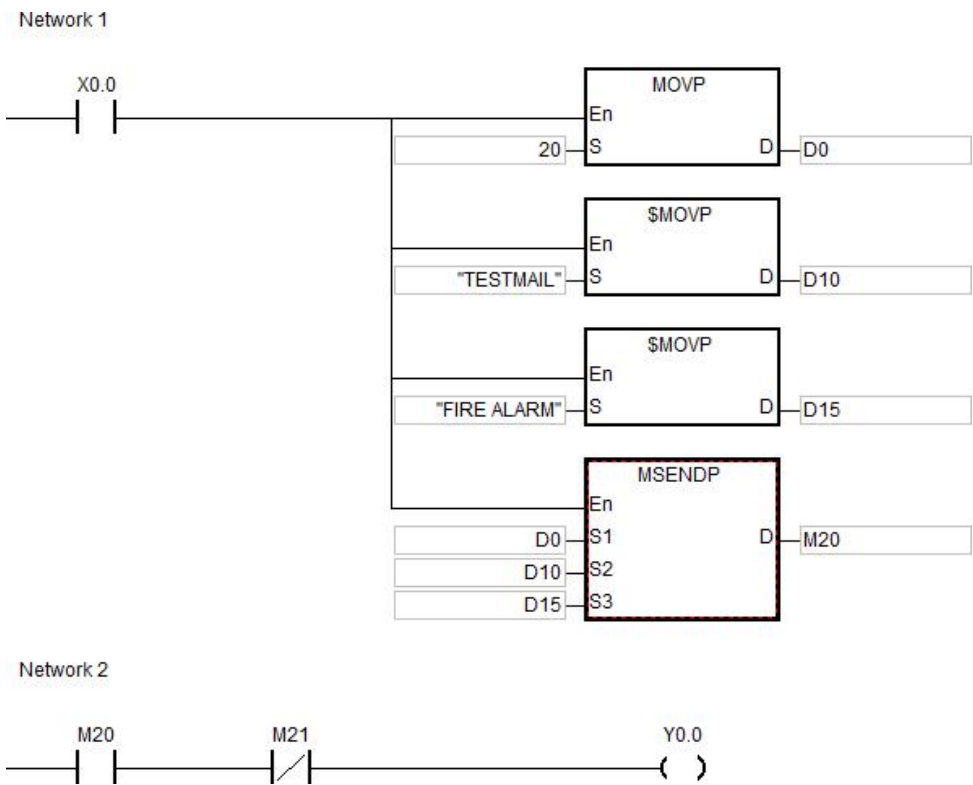
3. Електронна пошта встановлюється наступним чином.

Операнд	опис	Діапазон налаштування
S_1	Віддалений номер електронної пошти	Значення в S_1 має бути між 1–15. Встановить біт 0–біт 3 у ISPSOft, щоб вказати віддалені адреси електронної пошти. У ISPSOft можна встановити до чотирьох електронних адрес. Біт 0 відповідає електронній пошті 1, біт 1 відповідає електронній пошті 2 і так далі. Якщо ви хочете надіслати електронний лист, ви повинні встановити відповідний біт в ISPSOft.
S_2	Тема електронного листа	Розмір теми листа: до 16 слів
S_3	Текст електронної пошти	Розмір тіла електронного листа: до 64 слів
D	Завершення інструкція	Після завершення виконання інструкції біт увімкнено. Якщо виконання інструкції ненормальне, наступний біт (D+1) увімкнено.

4. Загалом використовується імпульсна інструкція MSENDP.

приклад

Припустімо, що значення в D0 дорівнює 00000101. Коли X0.0 увімкнено, електронний лист надсилається на номери електронної пошти 1 і 3. Після завершення зв'язку з сервером SMTP M20 увімкнено. Якщо під час зв'язку не виникає помилка, M21 вимкнено, а Y0.0 – увімкнено.



Додаткові зауваження

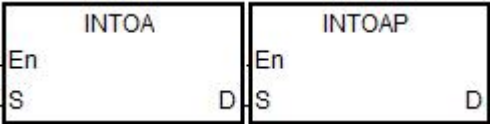
1. Для довжини рядка в S_2 або S_3 система фіксує дані з кінцем 16#00. Якщо довжина рядка більша за максимальне значення (з 16#00 як кінець), довжина рядка в S_2 або S_3 обробляється як максимальне значення.
2. Якщо ви оголошуєте операнд D в ISPSoft, типом даних є ARRAY [2] із WORD/INT.
3. Зарезервуйте одне слово між S_2 і S_3 для символу переривання.
4. У наведеній нижче таблиці пояснюються помилки під час виконання електронної інструкції.

Код помилки	Прапор помилки	опис
16 № 2003	SM0	1. $D + 1$ перевищує радіус дії пристрою 2. $S_1 < 1$ або $S_1 > 15$
16#600D	SM1100	Мережа Ethernet не підключена
16#6100	D+1	Конфлікти спілкування
16#6107	D+1	Тайм-аут спілкування електронною поштою
16#6108	D+1	Помилка автентифікації пароля облікового запису сервера SMTP
16#6111	D+1	Недійсна віддалена електронна адреса

API	Код інструкції				Операнд						функція						
2206		INTOA	P		S , D						Перетворення цілої IP-адреси в рядкову IP-адресу						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S								●	●								
D								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D		●				●							

СИМВОЛ	Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
	AS	AS	-



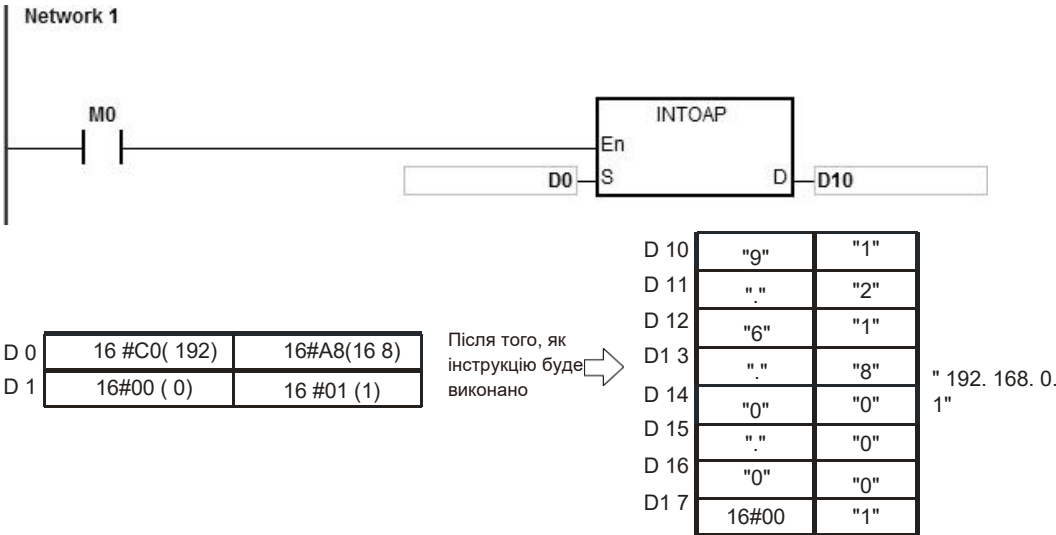
S : Вихідне значення

D : Результат перетворення

Пояснення

1. Ця інструкція перетворює IP-адресу в **S** і **S** + 1 на рядкову IP-адресу та зберігає результат перетворення в **D**.
2. Операнд **D** займає вісім послідовних пристроїв.

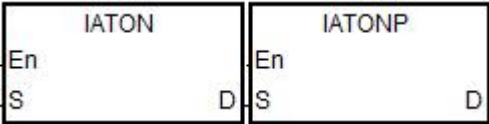
приклад



API	Код інструкції				Операнд								функція				
2207		ЯТОН	P		S, D								Перетворення рядкової IP-адреси в цілу IP-адресу				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S								●	●						O		
D								●									

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							●
D		●				●							

СИМВОЛ	Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
	AS		AS	-



S : Вихідне значення

D : Результат перетворення

Пояснення

1. Ця інструкція перетворює рядок IP-адреси в **S** на цілу IP-адресу та зберігає результат перетворення в **D** і **D + 1**.
2. Операнд **S** займає вісім послідовних пристроїв.

"192.168.0.1"

S+0	"9"	"1"
S+1	"."	"2"
S+2	"6"	"1"
S+3	"."	"8"
S+4	"0"	"0"
S+5	"."	"0"
S+6	"0"	"0"
S+7	16# 00	"1"

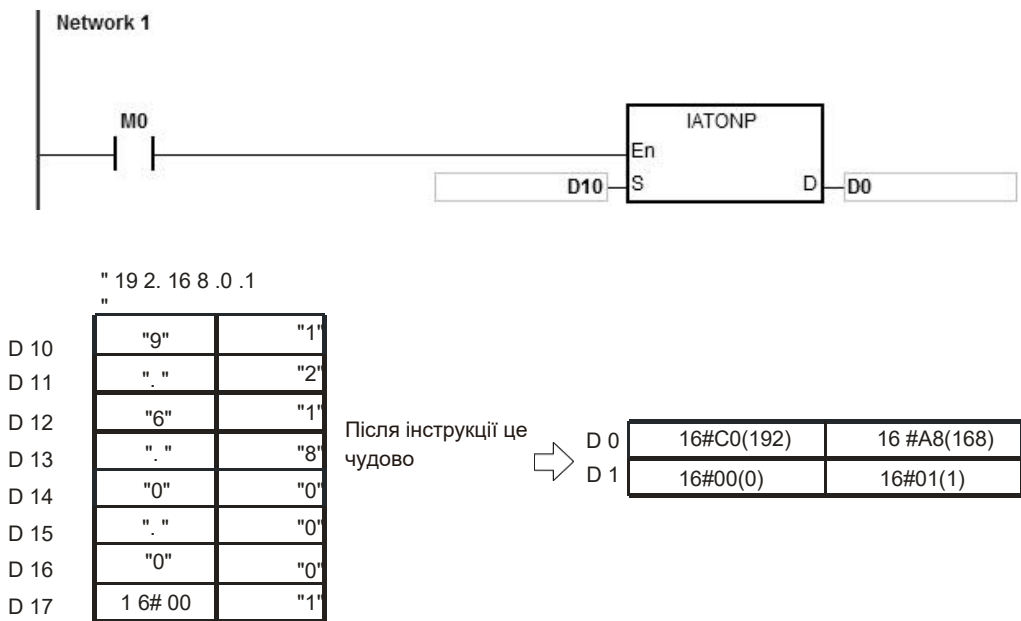
Після того, як інструкція буде виконана



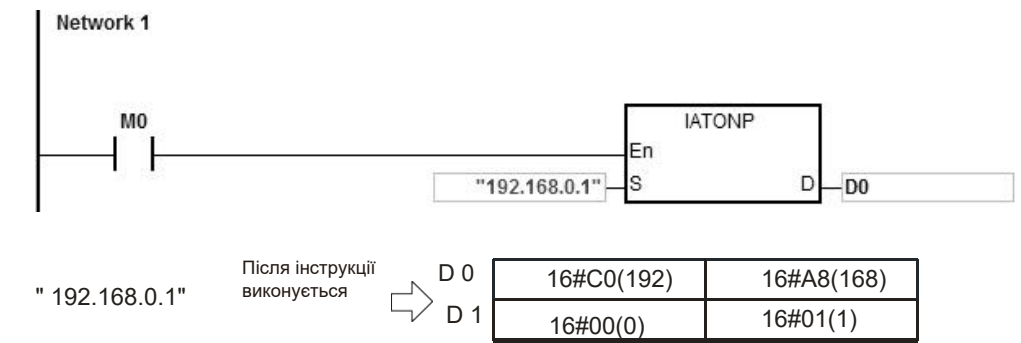
D+0	16#C0(192)	16#A8(168)
D+1	16#00(0)	16#01(01)

3. У кожному розділі рядка IP-адреси в **S** є **1–3 символи**. Ці розділи розділені знаком «.» (16#2E). Наприклад, ви можете ввести «192.168.0.1» замість «192.168.000.001».
4. Значення кожної частини рядка IP-адреси в **S** має бути між 0–255.

Приклад 1



Приклад 2



Додаткові зауваження

1. Якщо рядок у **S** не закінчується на 16#00, SM0 увімкнено, а код помилки в SR0 – 16#200E.
2. У рядку в **S**, за винятком коду, що представляє десяткову крапку, двійкові коди мають бути між 16#30– 16#39, інакше інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
3. Якщо немає трьох десяткових знаків «.» у рядку в **S** інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
4. Якщо значення будь-якої частини рядка IP-адреси в **S** не між 0–255, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.
5. Кількість символів у будь-якому розділі рядка IP-адреси в **S** має бути між 1–3, інакше інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

6. Якщо ви оголошуєте операнд **S** в ISPSoft, типом даних буде ARRAY [8] WORD/INT.
7. Якщо ви оголошуєте операнд **D** в ISPSoft, типом даних є ARRAY [2] із WORD/INT.

API	Код інструкції				Операнд						функція					
2208		EIPRW			S_1-S_7, n, S, D_1, D_2						Читання та запис даних EtherNet/IP					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●	●							
S_2								●	●				○	○		
S_3								●	●				○	○		
S_4								●	●				○	○		
S_5								●	●				○	○		
S_6	●	●	●	●												
S_7								●	●				○	○		
n								●	●				○	○		
S								●								
D ₁								●								
D ₂								●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁ -D ₂	Див. Пояснення в інструкції.												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

EIPRW	
En	
S1	D1
S2	D2
S3	
S4	
S5	
S6	
S7	
n	
S	

S_1-S_7, n, S : Дивіться Пояснення нижче.

D_1-D_2 : Дивіться Пояснення нижче.

Пояснення

- Ця інструкція читає та записує дані EtherNet/IP. У наступній таблиці наведено назви та пояснення S_1-S_7 , n , S і D_1-D_2 .

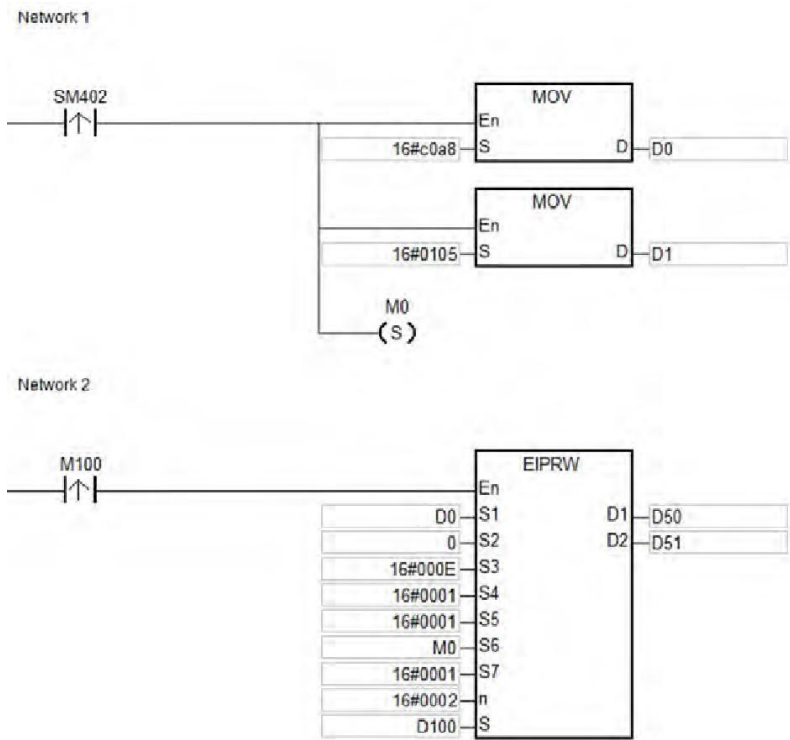
Операнд	Ім'я	опис	Тип даних	Зауваження
S_1	IP-адреса	Перші два розділи IP-адреси займають перше слово, а решта дві частини IP-адреси займають друге слово. Наприклад, якщо IP адреса 192.168.1.5, $S_1=16\#C0A8$ і $S_1+1=16\#0105$.	WORD[2]	Займає два послідовних слова
S_2	Режим підключення	0: УКММ	WORD	
S_3	Код функції (сервісний код)	Діапазон: 16#0000–00FF. Якщо функція код перевищує діапазон, інструкція не виконується.	WORD	
S_4	ID класу	Зверніться до протоколу EtherNet/IP.	WORD	
S_5	Ідентифікатор екземпляра	Зверніться до протоколу EtherNet/IP.	WORD	
S_6	Перемикач ID атрибута	ON: Увімкнути; OFF: Вимкнути	BOOL	
S_7	ID атрибута	Зверніться до протоколу EtherNet/IP.	WORD	
n	Тривалість читання та запису даних	Розмір даних для запису або читання; одиниця: байт; максимум: 200 байт	WORD	
S	Зареєструватися для читання/запис даних	Вихідний реєстр для надісланих даних або де зберігаються отримані дані	WORD[n]	
D_1	Статус зв'язку	0 : зв'язок не запущено 1 : Зв'язок у процесі 2 : Зв'язок завершився без errors 3 : Помилка зв'язку 4 : Помилка в налаштуванні параметрів	WORD	
D_2	Код помилки	Основний код помилки та розширений код помилки	WORD[2]	Займає два слова поспіль

2. Коли ви вмикаєте цю інструкцію вперше, вона надсилає команду зв'язку. Якщо кількість підключених підлеглих пристроїв досягає верхньої межі, значення статусу зв'язку в **D₁** дорівнює 3, і команда зв'язку не надсилається. Код помилки показує 16#6306 у **D₂[1]**.
3. Коли параметр виходить за межі діапазону, інструкція не виконується, а значення статусу зв'язку в **D₁** становить 4.
4. Ця інструкція повинна працювати з тригером по висхідному фронту. Якщо він повторно запускається, код помилки показує 16#02
D₂[0]. Зачекайте, поки виконання інструкції завершиться, перш ніж починати іншу інструкцію.

Приклад 1

Інструкція читає код постачальника пристрою за адресою 192.168.1.5 і зберігає значення в D100. Він встановлює параметри EtherNet/IP Object, як показано нижче.

- (1) Ідентифікатор класу = 1
- (2) Ідентифікатор екземпляра = 1
- (3) Ідентифікатор атрибута = 1



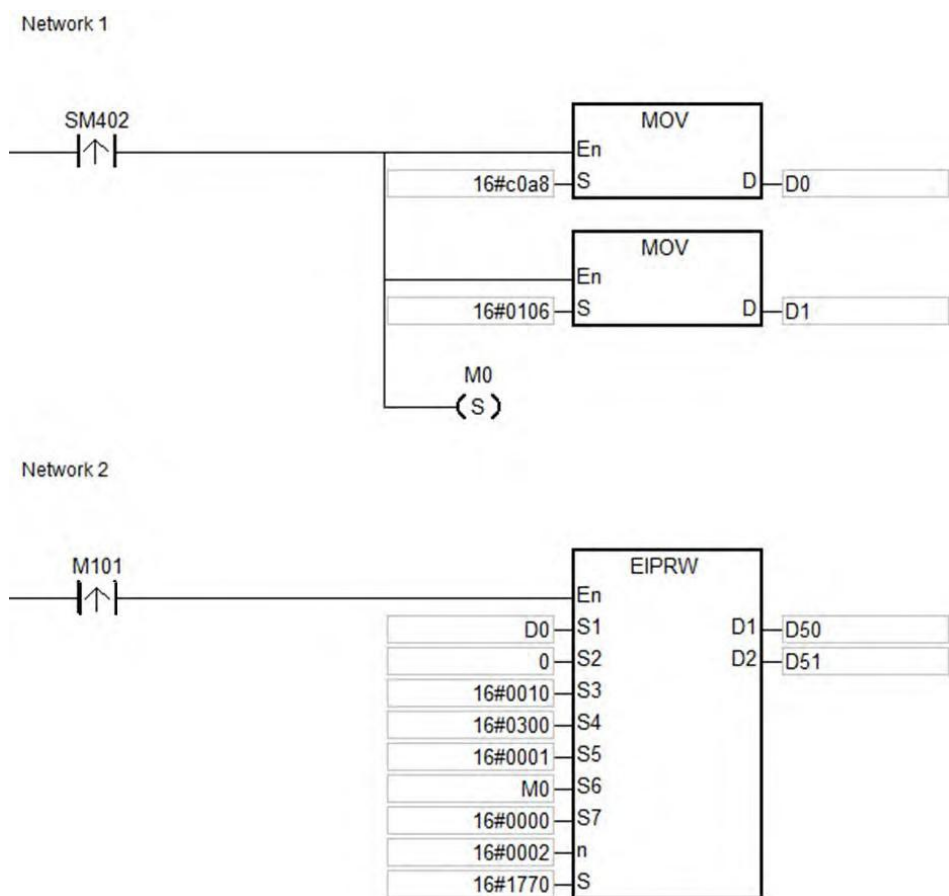
Приклад 2

Максимальна частота (01-00) приводу двигуна змінного струму (за адресою 192.168.1.6) встановлена на 60,00 Гц. Параметри EtherNet/IP Object встановлюються, як показано нижче.

(1) Ідентифікатор класу = 16#0300

(2) Ідентифікатор екземпляра = 16#0001

(3) Ідентифікатор атрибута = 16#0000



Додаткові зауваження

1. Якщо ви оголошуєте операнд **S**₁ в ISPSOft, типом даних є ARRAY [2] із WORD.
2. Якщо ви оголошуєте операнд **S** в ISPSOft, типом даних буде ARRAY [**n**] of WORD, де **n** — це розмір даних, що читаються/записуються.
3. Якщо ви оголошуєте операнд **D**₂ в ISPSOft, типом даних є ARRAY [2] із WORD.

4. У наступній таблиці пояснюються коди помилок у **D₂[0]**.

Статус Код	опис	Що робити
16#00	Підключення успішне	
16#01	Помилка з'єднання	Переконайтеся, що файл EDS підпорядкованого пристрою правильний.
16#02	Підключені пристрої недоступні	1. Переконайтеся, що кількість пристроїв, підключених до головного, відповідає не перевищувати ліміт. 2. Переконайтеся, що кількість пристроїв, підключених до веденого пристрою, не перевищує обмеження. 3. Інструкція запускається повторно. Зачекайте, поки виконання інструкції завершиться, перш ніж починати іншу інструкцію.
16#03	Помилка в параметрі	Переконайтеся, що дані для читання/запису в S правильні.
16#04	Помилка шляху	Переконайтеся, що налаштування правильні для ідентифікатора класу (S₄), ідентифікатора екземпляра (S₅) та ідентифікатора атрибута (S₇).
16#05	Шлях призначення – ні існують	Переконайтеся, що налаштування правильні для ідентифікатора класу (S₄), ідентифікатора екземпляра (S₅) та ідентифікатора атрибута (S₇).
16#07	З'єднання розірване	1. Переконайтеся, що порт Ethernet підлеглого пристрою підключено належним чином. 2. Переконайтеся, що налаштування таймера підтримки активності правильні для підлеглого пристрою.
16#08	Сервісний код відсутній підтримується	Переконайтеся, що код функції (S₃) правильний.
16#09	Недійсне значення атрибута	Переконайтеся, що регістри, залучені до читання/запису даних, і їх вміст правильний.
16#0A	Помилка в списку атрибутів	Переконайтеся, що атрибут об'єкта підлеглого пристрою дозволяє ПЛК виконувати Функції <code>Get_Attribute_List</code> і <code>Set_Attribute_List</code>.
16#0B	Конфлікти передачі	Переконайтеся, що сервісне налаштування повторюється.
16#0C	Конфлікти статусу об'єкта	Переконайтеся, що підключення власника вводу-виводу встановлено.
16#0D	Об'єкт існував	Переконайтеся, що підлеглий пристрій підтримує визначений об'єкт. Сервіс не робить необхідно виконати, якщо визначений об'єкт підтримується.
16#0E	Атрибут не доступний для запису	Переконайтеся, що атрибут об'єкта підтримує функцію запису.

Статус Код	опис	Що робити
16#0F	Без привілеїв для виконання код послуги	Переконайтеся, що підпорядкованому пристрою дозволено виконувати службовий код.
16#10	Пристрій наразі не може виконувати послугу	Переконайтеся, що підключення власника вводу-виводу встановлено.
16#11	Також розмір даних відповіді великий	Переконайтеся, що довжина даних в атрибуті об'єкта не перевищує межа (100 слів).
16#12	Послідовність доступу до даних помилка під час доступу до тегу	Переконайтеся, що довжина та тип даних правильні.
16#13	Передані дані теж короткий	Переконайтеся, що довжина (n) прочитаних/записаних даних правильна.
16#14	Значення атрибута не підтримується	Переконайтеся, що перемикач ідентифікатора атрибута (S₆) і ідентифікатор атрибута (S₇) правильні.
16#15	Передані дані теж довго	Переконайтеся, що довжина даних для читання/запису (n) правильна.
16#16	Об'єкт не існує	Переконайтеся, що ідентифікатор класу (S₄) правильний.
16#17	Послідовність доступу до даних помилка під час доступу до тегу	1. Переконайтеся, що підключення до мережі Ethernet правильне. 1. Переконайтеся, що жодні пакети не втрачаються під час з'єднання Ethernet.
16#18	Значення атрибута не збережено	Переконайтеся, що на підлеглому пристрої не виникло жодного стану помилки.
16#19	Зберігання значення атрибута помилка	Переконайтеся, що на підлеглому пристрої не виникло апаратної помилки.
16#1A	Помилка маршрутизатора: довжина пакет запиту перевищує обмеження	Переконайтеся, що довжина даних для читання/запису (n) не перевищує обмеження маршрутизатора.
16#1B	Помилка маршрутизатора: довжина пакет відповіді перевищує обмеження	Переконайтеся, що довжина даних для читання/запису (n) не перевищує обмеження маршрутизатора.
16#1F	Визначений користувачем об'єкт помилка доступу	Зверніться до визначення помилки веденого пристрою.
16#20	Неправильне значення параметра	Переконайтеся, що значення читання/запису в S правильне.

5. У наведеній нижче таблиці пояснюються коди помилок у $D_2[0]$.

Код помилки	Прапор помилки	опис
16 № 2003	SM0 / D_1	1. Значення в S_1 , S_2 , S_3 або S_6 перевищує діапазон 2. $S + n$ поза діапазоном адреси пристрою
16#200Б	SM0	Операнд n поза діапазоном
16#600D	SM1100	Мережа Ethernet не підключена
16#6301	D_1	З'єднання з віддаленим пристроєм порушено
16#6302	D_1	Час очікування відповіді віддаленого пристрою
16#6303	D_1	Незаконна IP-адреса
16#6304	D_1	Помилка у сервісному коді команди відповіді
16#6305	D_1	Помилка в довжині команди відповіді
16#6306	D_1	Конфлікти спілкування

API	Код інструкції			Операнд							функція						
2209		SCONF	P	$S_1 - S_{10}$							Налаштування параметрів сокета TCP/UDP						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●	●				O	O		
S_2								●	●				O	O		
S_3								●	●				O	O		
S_4								●	●				O	O		
S_5								●	●				O	O		
S_6								●	●				O	O		
S_7								●	●				O	O		
S_8								●	●				O	O		
S_9								●	●				O	O		
S_{10}								●	●				O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
$S_1 - S_{10}$	Див. Пояснення в інструкції.												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

SCONF		SCONFP	
En		En	
S1		S1	
S2		S2	
S3		S3	
S4		S4	
S5		S5	
S6		S6	
S7		S7	
S8		S8	
S9		S9	
S10		S10	

$S_1 - S_{10}$: Дивіться Пояснення нижче

Пояснення

1. Ця інструкція встановлює параметри сокета TCP/UDP. У наступній таблиці наведено назви та пояснення $S_1 - S_{10}$.

Операнд	Ім'я	опис	Тип даних	Зауваженн я
S_1	Вибір TCP/UDP	0: TCP, 1: UDP	WORD	
S_2	Номер розетки	Діапазон: 1–4	WORD	

Операнд	Ім'я	опис	Тип даних	Зауваження
S_3	Віддалена IP-адреса	Перші два розділи займають IP-адресу перше слово, а інші два розділи IP-адреси займають друге слово. Наприклад, якщо IP-адреса 192.168.1.5, $S_3 = 16\#C0A8$ і $S_3 + 1 = 16\#0105$.	WORD[2]	Займає два послідовних слова
S_4	Віддалений порт	Діапазон: 0–65535; 0 означає будь-який порт	WORD	
S_5	Місцевий порт	Діапазон: 0–65535; 0 означає будь-який порт	WORD	
S_6	Зареєструйте джерело надсилання даних	Визначає номер пристрою D. Діапазон: 0–29999. Приклад: Значення налаштування 100 означає, що S_7 байтів даних надсилаються з регістрів, починаючи з D100 (від молодшого байта до старшого).	WORD	
S_7	Розмір надісланих даних	Максимум: 512 байт Для ПЛК із програмною програмою версії 1.06.30 або новішої; максимум: 4096 байт	WORD	
S_8	Зареєструватися, де зберігаються отримані дані	Визначає номер пристрою D. Діапазон: 0–29999. Приклад: Значення налаштування 200 означає, що S_9 байтів даних приймаються та зберігаються в регістрах, починаючи з D200 (від молодшого байта до старшого).	WORD	
S_9	Розмір отриманих даних	Максимум: 512 байт Для ПЛК із програмною програмою версії 1.06.30 або новішої; максимум: 4096 байт	WORD	
S_{10}	Час підключення	Діапазон: 1–30000, одиниця вимірювання: секунда	WORD	Застосовується лише в режимі TCP

- Загалом використовується імпульсна інструкція SCONF.
- Параметри Socket за замовчуванням використовують значення налаштувань у HWCONFIG. Використовуйте інструкцію SCONF, коли потрібно змінити параметри під час зв'язку.
- Якщо ви встановлюєте параметри під час обміну даними з розеткою, налаштування не набудуть чинності, доки не буде завершено зв'язок. Налаштування слід змінити після того, як розетка більше не використовується.
- Якщо будь-який із параметрів виходить за межі діапазону, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд							функція						
2210		MCONF	P	S ₁ - S ₁₁							Читання/запис даних Modbus TCP						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁								●	●				○	○		
S ₂								●	●							
S ₃								●	●				○	○		
S ₄								●	●				○	○		
S ₅								●	●				○	○		
S ₆								●	●				○	○		
S ₇			●					●								
S ₈								●	●				○	○		
S ₉								●	●				○	○		
S ₁₀								●	●				○	○		
S ₁₁			●					●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁ - S ₁₁	Див. Пояснення в інструкції.												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

MCONF		MCONF	
En		En	
S1		S1	
S2		S2	
S3		S3	
S4		S4	
S5		S5	
S6		S6	
S7		S7	
S8		S8	
S9		S9	
S10		S10	
S11		S11	

S₁ - S₁₁ : Дивіться Пояснення нижче

Пояснення

1. Ця інструкція читає та записує дані Modbus TCP. У наступній таблиці наведено назви та пояснення $S_1 - S_{11}$.

Операнд	Ім'я	опис	Дані Тип	Зауваження
S_1	Таблиця обміну даними №	Номер таблиці обміну даними Modbus TCP, діапазон: 1–32	WORD	
S_2	Віддалена IP-адреса	Перші два розділи IP-адреси займають перше слово, а решта дві частини IP-адреси займають друге слово. Наприклад, якщо IP адреса 192.168.1.5, $S_2 = 16\#C0A8$ і $S_2 + 1 = 16\#0105$.	WORD[2]	Займає два послідовних слова
S_3	Адреса станції віддаленого підпорядкованого	Діапазон: 0–255. Якщо налаштування перевищує значення в молодшому байті використовується автоматично.	WORD	
S_4	Прочитайте код функції	Діапазон: 16#01, 02, 03, 04 і 17. Якщо код функції виходить за межі діапазону, вхідні параметри, включаючи адресу та довжину даних, не змінюються.	WORD	
S_5	Читати дистанційно адреса зв'язку	16#0000–16#FFFF	WORD	
S_6	Прочитати довжину даних	Код функції 16#01-02 для читання біта пристрою, в діапазоні від 1 до 256 Код функції 16#03-04 для читання пристрою Word, в діапазоні від 1 до 100 Якщо значення налаштування дорівнює 0 або менше 0, ця функція не активна. Якщо значення налаштування більше максимального значення, воно буде розглядатися як максимальне значення.	WORD	
S_7	Локальний реєстр, де зберігаються отримані дані	Код функції для читання даних у бітах пристрій; Лише пристрій M Код функції для читання даних у пристрої Word; Лише пристрій D	BOOL WORD	

Операнд	Ім'я	опис	Дані Тип	Зауваження
S₈	Напишіть код функції	Діапазон: 16#05, 06, 0F і 10. Якщо код функції виходить за межі діапазону, вихідні параметри, включаючи адресу та довжину даних, не змінюються.	WORD	
S₉	Пишіть дистанційно адреса зв'язку	Діапазон: 16#0000–16#FFFF	WORD	
S₁₀	Записати довжину даних	Код функції 16#0F для запису бітового пристрою, в діапазоні від 1 до 256; значення в коді функції 16#05 може бути лише 1; для ПЛК із FW V1.06.20 або новішої версії: 0 можна використовувати в 16#05 Код функції 16#10 для запису пристрою Word, в діапазоні від 1 до 100; значення в коді функції 16#06 може бути лише 1; для ПЛК із FW V1.06.20 або новішої версії: 0 можна використовувати в 16#05 Якщо значення налаштування дорівнює 0 або менше 0, ця функція не активна. Якщо значення налаштування більше максимального значення, воно буде розглядатися як максимальне значення.	WORD	
S₁₁	Локальний реєстр, куди записуються віддалені дані	Функціональний код для запису даних у бітовий пристрій; Лише пристрій M Код функції для запису даних у пристрій Word; Лише пристрій D	BOOL WORD	

2. Загалом використовуйте імпульсну інструкцію MCONFP.

3. Наступні пункти пояснюють коди функцій зв'язку **Modbus y S4 та S8**.

Коли ПЛК серії AS зчитує дані з кількох розрядних пристроїв (не дискретних пристроїв введення), код функції надіслано 1 (16#01).

Коли ПЛК серії AS зчитує дані з кількох розрядних пристроїв (тільки дискретні пристрої введення), надсилається код функції 2 (16#02).

Коли ПЛК серії AS зчитує дані з кількох пристроїв введення даних (не вхідних регістрів), надсилається код функції 3 (16#03).

Коли ПЛК серії AS зчитує дані з кількох пристроїв слова (тільки вхідні регістри), надсилається код функції 4 (16#04).

Коли ПЛК серії AS записує статус на однорозрядний пристрій, надсилається код функції 5 (16#05).

Коли ПЛК серії AS записує дані на пристрій з одним словом, надсилається код функції 6 (16#06).

Коли ПЛК серії AS записує статус на багаторозрядні пристрої, надсилається код функції 15 (16#0F). Коли

ПЛК серії AS записує дані на кілька пристроїв Word, надсилається код функції 16 (16#10). Коли ПЛК серії

AS записує дані на кілька пристроїв Word, надсилається код функції 23 (16#17). Наразі ПЛК серії AS

підтримує лише перелічені вище коди функцій.

4. Параметри в інструкції дійсні лише під час роботи ПЛК. Коли ПЛК знову вмикається після вимкнення, він приймає налаштування з таблиці обміну даними за замовчуванням. Якщо вам потрібно змінити деякі параметри під час обміну даними, ця інструкція змінює ці параметри.

5. Якщо вказаний сокет уже з'єднується, нові параметри стають дійсними після завершення поточного з'єднання.

6. Якщо будь-який параметр не входить у допустимий діапазон, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.

7. ПЛК серії AS не підтримує комунікаційні інструкції Modbus TCP або щось подібне

інструкція ПЛК серії ETHRW DVP. Якщо ви використовуєте програму PLC для керування зв'язком Ethernet Modbus TCP, спочатку увімкніть режим «Програмне керування» в таблиці обміну даними та використовуйте номери SM із наступної таблиці.

Номер SM	Атрибут	Пояснення параметрів обміну даними Ethernet
SM1167	R/W	Позначте, щоб увімкнути обмін даними
SM1168–SM1199	R/W	Прапорець для ввімкнення підключень для обміну даними 1–32
SM1200–SM1231	R	Позначка успіху для з'єднань обміну даними 1–32
SM1232–SM1263	R	Прапор помилки для з'єднань обміну даними 1–32

Коли дані отримано успішно, прапор успіху увімкнено. Коли виникає помилка при отриманні даних, відображається позначка помилки

ON. Прапор успіху та прапор помилки будь-якого одного роз'єму з'єднання ніколи не вмикаються одночасно.

8. У наступній таблиці пояснюються номери SR (тільки для читання), які використовуються в обміні даними.

Номер SR	Пояснення
SR1120–SR1151	Вказує фактичний час зв'язку для з'єднань 1–32.
SR1152–SR1183	Вкажіть коди помилок зв'язку для з'єднань 1–32.

У наведеній нижче таблиці визначено коди помилок у SR1152–SR1183.

Код помилки	Пояснення
16#00XX	Помилка відповіді віддаленого пристрою
16#F000	Мережа Ethernet не підключена
16#F001	Час очікування відповіді віддаленого пристрою
16#F003	Тайм-аут з'єднання TCP
16#F007	Помилка команди відповіді
16#F009	З'єднання з віддаленим пристроєм розірвано

API	Інструкція				Операнд							опис					
2211		EMCONF1	P		Server ~ Passw							Налаштування значень параметрів сервера електронної пошти					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Server								●	●						○		
Port								●	●				○	○			
LMail								●	●						○		
Sub								●	●						○		
Verify								●	●				○	○			
User								●	●						○		
Passw								●	●						○		

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
	Зверніться до пояснення наступної інструкції												
Інструкція по пульсу		16-бітна інструкція		32-розрядна інструкція									
AS		AS		-									

СИМВОЛ

EMCONF1	EMCONF1P
En	En
Server	Server
Port	Port
LMail	LMail
Sub	Sub
Verify	Verify
User	User
Passw	Passw

Server ~Passw : Зверніться до наступного пояснення

Пояснення

1. Ця інструкція доступна для ПЛК із FW V1.06.00.
2. Перегляньте ім'я операнда, використовуйте пояснення та тип даних у наступній таблиці.

Операнд	Ім'я	Пояснення	Тип даних	Зауваження
Server	Сервер електронної пошти	Server +0: довжина даних, одиниця: байти (0 або 4; 0: без змін) Server +1~ Server +2: дані (Дані є розміщені з молодших 8 бітів.) Наприклад: 192.168.0.1, D0 = 16#0004, D1= 16#0001, D2= 16#C0A8;	WORD[N]	Зверніться до Сервера +0 за зайнятість адреси. Можуть бути зайняті не більше 3 послідовних слів.
		В одному рядку можна використовувати до 15 слів; кожен розділ в одному рядку може містити 1-3 слова та використовуйте крапку «.» відокремити кожен рядок, наприклад 192.168.0.1	Рядок	
Port	спілкування порт	Діапазон: 1~65535, 0: без змін	WORD	
LMail	Місцева електронна адреса	LMail +0: довжина даних, одиниця: байти (діапазон: 0~63, 0: без змін.) LMail +1~ LMail +32: дані (Дані є розміщується з молодших 8 бітів.) Наприклад: <u>ex@deltaww.com</u> D0 = 16#000E, D1 = 16#7865, D2 = 16#6440, D3 = 16#6C65, D4 = 16#6174, D5 = 16#7777, D6 = 16#2E63, D7 = 16#6D6F	WORD[N]	Зверніться до LMail +0 за зайнятість адреси. Може бути зайнято не більше 33 послідовних слів.
		В одному рядку можна використовувати до 63 слів	Рядок	
Sub	Тема електронного листа	Sub +0: довжина даних, одиниця: байти (діапазон: 0~31, 0: без змін.) Sub +1~ Sub +16: вміст даних (Дані розміщуються з молодших 8 бітів.) Наприклад: Sub 0 = 16#0003, D1 = 16#7553, D2 = 16#0062	WORD[N]	Зверніться до Sub +0 для зайнятості адреси. Може бути зайнято не більше 17 послідовних слів.
		В одному рядку можна використовувати до 31 слова	Рядок	
Verify	Увімкнути/ вимкнути перевірка	0: без змін, 1: вимкнути перевірку, 2: Увімкніть перевірку	WORD	

Операнд	Ім'я	Пояснення	Тип даних	Зауваження
User	Ім'я користувача	User +0: довжина даних, одиниця: байти (діапазон: 0~31, 0: без змін.) User +1~ User +16: вміст даних (Дані розміщуються з молодших 8 бітів.) Наприклад: користувач D0 = 16#0004, D1 = 16#7355, D2 = 16#7265	WORD[N]	Зверніться до користувача +0 для зайнятості адреси. Може бути зайнято не більше 17 послідовних слів.
		В одному рядку можна використовувати до 31 слова	Рядок	
Passw	Пароль	Passw +0: довжина даних, одиниця: байти (діапазон: 0~20, 0: без змін.) Passw +1~ Passw +10: вміст даних (Дані розміщуються з молодших 8 бітів.) Наприклад: Passw D0 = 16#0005 , D1 = 16#5061 , D2 = 16#7373 , D3 = 16#0077	WORD[N]	Зверніться до Passw +0 за зайнятість адреси. Може бути зайнято не більше 11 послідовних слів.
		В одному рядку можна використовувати до 20 слів	Рядок	

Ви можете налаштувати параметри електронної пошти в ISPSOft > HWCONFIG > Device Setting > Ethernet Port Advance Налаштування > Електронна пошта. Звідси ви також можете перевірити параметри, встановлені за допомогою інструкцій. Але після вимкнення живлення зберігаються лише налаштування, встановлені в ISPSOft.

Device Setting

Options Data Exchange -COM1 Data Exchange -COM2 Data Exchange -Ethernet Data Exchange -FEN02 Data Exchange -Function Card1 Data Exchange -Function Card2

AS332T-A

- System settings
 - COM1 Port Setting
 - COM2 Port Setting
 - Ethernet Port Basic Setting
 - Ethernet Port Advance Setting
 - IP Filter
 - NTP
 - Email
 - 1st Trigger Setting
 - 1st Trigger Recipient
 - 2nd Trigger Setting
 - 2nd Trigger Recipient
 - 3rd Trigger Setting
 - 3rd Trigger Recipient
 - 4th Trigger Setting
 - 4th Trigger Recipient

Email

Parameter name	Value	Unit	Default	Minimum	Maximum
Email Function Enable	☒		☐	-	-
SMTP Server	192.168.0.1		1.1.1.1	1.1.1.1	223.255.255.255
Port	123		25	1	65535
Local Email	ex@deltaww.com		-	-	-
Mail Subject	Sub		-	-	-
Account Identification	☒		☐	-	-
User name	User		-	-	-
Password	*****		-	-	-
1st Remote Address			-	-	-
2nd Remote Address			-	-	-
3rd Remote Address			-	-	-
4th Remote Address			-	-	-

Default Import Export

OK

API	Інструкція			Операнд							опис						
2212		EMCONF2	P	Index , Mail							Встановлення електронної адреси						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Index								●	●				○	○		
Mail								●	●						●	

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
	Зверніться до пояснення наступної інструкції												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

EMCONF2
En
Index
Mail

EMCONF2P
En
Index
Mail

Index, Mail

: Зверніться до наступного пояснення

Пояснення

- Ця інструкція доступна для ПЛК із FW V1.06.00.
- Перегляньте ім'я операнда, використовуйте пояснення та тип даних у наступній таблиці.

Операнд	Ім'я	Пояснення	Тип даних	Зауваження
Index	Номер поштової адреси	Index : номер адреси електронної пошти змінено (діапазон: 0~4, 0: не змінювати адресу електронної пошти.)	WORD	
Mail	Поштова адреса	Mail +0 : довжина даних значення індексу з одиниця байтів (діапазон: 0~63, 0: не змінювати адреса електронної пошти.) Mail +1~ Mail +32 : вміст даних Index (Дані розміщуються з нижньої 8 біти.) Наприклад: ex@deltaww.com D0 = 16#000E, D1 = 16#7865, D2 = 16#6440, D3 = 16#6C65, D4 = 16#6174, D5 = 16#7777, D6 = 16#632E, D7 = 16#6D6F В одному рядку можна використовувати до 63 слів	WORD[N]	, зверніться до Mail +0 професія. Щонайбільше 33 слова поспіль можна зайняти.
			Рядок	

Ви можете налаштувати параметри, пов'язані з електронною поштою, у ISPSOft > HWCONFIG > Налаштування пристрою > Розширені налаштування порту Ethernet > Електронна пошта. Звідси ви також можете перевірити параметри, встановлені за допомогою інструкцій. Але після вимкнення живлення зберігаються лише налаштування, встановлені в ISPSOft.

Device Setting

Options Data Exchange -COM1 Data Exchange -COM2 Data Exchange -Ethernet Data Exchange -FEN02 Data Exchange -Function Card1 Data Exchange -Function Card2

AS332T-A

- System settings
- COM1 Port Setting
- COM2 Port Setting
- Ethernet Port Basic Setting
- Ethernet Port Advanced
 - IP Filter
 - NTP
 - Email
 - 1st Trigger Setting
 - 1st Trigger Recipient
 - 2nd Trigger Setting
 - 2nd Trigger Recipient
 - 3rd Trigger Setting
 - 3rd Trigger Recipient
 - 4th Trigger Setting
 - 4th Trigger Recipient

Email

Parameter name	Value	Unit	Default	Minimum	Maximum
Email Function Enable	☒		☐	-	-
SMTP Server	1.1.1.1		1.1.1.1	1.1.1.1	223.255.255.255
Port	25		25	1	65535
Local Email				-	-
Mail Subject				-	-
Account Identification	☐		☐	-	-
User name	...			-	-
Password				-	-
1st Remote Address	mail1@deltaww.com			-	-
2nd Remote Address	mail2@deltaww.com			-	-
3rd Remote Address	mail3@deltaww.com			-	-
4th Remote Address	mail4@deltaww.com			-	-

Default Import Export

OK

API	Інструкція				Операнд							функція					
2214		MQTT_Connect			Execute ~ Status							Встановлення підключення MQTT					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Execute	●	●	●	●				●		●	●						
SID								●	●		●		●				
BrokerNum								●	●		●		●				
ClientID								●	●		●				●		
CleanSession	●	●	●	●				●		●	●						
WillTopic								●	●		●				●		
WillMessage								●	●		●				●		
WillRetain	●	●	●	●				●		●	●						
WillQos								●	●		●		●				
WillFlag	●	●	●	●				●		●	●						
KeepAlive								●	●		●		●				
Status								●			●						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Execute ~ Status	Зверніться до наступного пояснення інструкцій.												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

MQTT_Connect
En
Execute Status
SID
BrokerNum
ClientID
CleanSession
WillTopic
WillMessage
WillRetain
WillQos
WillFlag
KeepAlive

Execute ~ Status : Дивіться пояснення інструкцій нижче.

Пояснення

1. Ця інструкція доступна для ISPSOft V3.16 або пізнішої версії та ПЛК AS300 із мікропрограмою версії 1.12 або пізнішої, якщо на ПЛК встановлено функціональну карту AS-FFTP01.
2. Зверніться до опису функцій на AS-FFTP01 у Посібнику з модулів серії AS або прикладі інструкцій нижче та налаштуйте параметри MQTT для AS-FFTP01 у HWCONFIG перед використанням.
3. Імена та описи операндів наведено в наступній таблиці.

Операнд	Ім'я	опис	Тип даних	Зауваження
Execute	Виконати	● Коли Execute змінюється на ON, create з'єднання. ● Коли Execute змінюється на OFF, створюється відключення.	BOOL	
SID	Номер слота	Фіксоване значення: 34 (представляє функціональну картку AS-FFTP01)	WORD	
BrokerNum	Сервер (Брокер) ID	Фіксоване значення: 1	WORD	
ClientID	Ім'я клієнта	Дані розташовані в порядку від старшого байта до молодшого. Максимальна довжина – 20 символів, а в кінці обов'язковий термінатор рядка (16#00).	WORD	Див. пояснення ідентифікатора клієнта нижче.
CleanSession	Чиста сесія	● ON: сервер очищає дані сеансу після відключення. ● OFF: сервер зберігатиме дані сеансу після відключення.	BOOL	Дивіться опис сеансу нижче.
WillTopic	Буде тема повідомлення	Дані розташовані в порядку від старшого байта до молодшого. Максимальна довжина – 100 символів, а в кінці обов'язковий термінатор рядка (16#00).	WORD	Дивіться пояснення до повідомлення про заповіт нижче.

Операнд	Ім'я	опис	Тип даних	Зауваження
WillMessage	Буде повідомлення	Дані розташовані в порядку від старший байт до молодшого байта. Максимальна довжина – 100 символів, а в кінці обов'язковий термінатор рядка (16#00).	WORD	
WillRetain	Зберегти волю повідомлення	Встановить, чи зберігатиметься повідомлення Will на сервер чи ні.	BOOL	
WillQos	Сервіс якість для повідомлення Will	Встановить якість обслуговування (Qos) для повідомлення Will. Діапазон налаштувань: 0, 1	WORD	
WillFlag	Воля прапор повідомлення	● ON: Увімкнути функцію повідомлення Will ● OFF: Вимкнути функцію повідомлення Will		
KeepAlive	Час збереження зв'язку	Коли час простою перевищує цей показник у 1,5 рази налаштування (одиниця: секунда), з'єднання буде закрито. ● Діапазон налаштувань: 0~65535. ● Рекомендоване значення: 60. ● Якщо значення налаштування дорівнює 0, з'єднання не буде закрито, незалежно від того, чи перевищує час простою 1,5 встановленого значення чи ні. Фактичний час підключення визначається налаштуваннями облікового запису сервера. Зверніться до посібника розробника серверної платформи для роботи.	DWORD	
Status	Держава	● Status[0] Код статусу 0: Не підключено/відключено 1: Підключено 2: Підключення/відключення 4: Сталася помилка, і з'явився код помилки. ● Status [1] Код помилки Перегляньте опис коду помилки нижче.	WORD[2]	Два посліпль слова зайняті. Перегляньте часові діаграми нижче, щоб дізнатися про час оновлення.

4. Перш ніж запускати **MQTT_Publish** або **MQTT_Subscribe** , переконайтеся, що підключення успішно встановлено. Ви можете використовувати код стану **MQTT_Connect** , щоб побачити, чи можна запустити **MQTT_Publish** або **MQTT_Subscribe** . Докладніше див. у прикладах операцій у **MQTT_Publish** (API2215) і **MQTT_Subscribe** (API2216).

5. Пояснення ідентифікатора клієнта

- (1) Використовуйте унікальне ім'я клієнта як його ідентифікатор клієнта, який використовується сервером для ідентифікації. Ідентифікатор клієнта має бути рядком, а його вміст обмежений цифрами та літерами.
- (2) Якщо **ClientID** встановлено на 0, для **CleanSession** має бути встановлено значення ON.

6. Опис сеансу

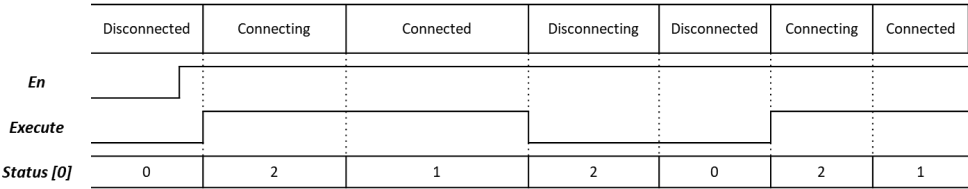
- (1) **Чистий сеанс** : вказує на те, що повідомлення від клієнта буде видалено сервером після відключення, а попередні сеанси буде видалено. Рекомендується для сценаріїв, де є повідомлення постійно публікуються або де втрата повідомлень прийнятна.
- (2) **Постійний сеанс** : вказує на те, що після відключення сервер збереже сеанс, включаючи незавершені повідомлення. Після відновлення з'єднання всі підписки буде автоматично відновлено, а збережені повідомлення автоматично доставлено. Рекомендується для сценаріїв, коли абоненти не можуть втратити повідомлення , а повідомлення абонента потрібно зберігати .
- (3) Фактичні повідомлення та вміст у сеансі визначаються обліковим записом сервера. Для налаштування зверніться до посібника розробника серверної платформи.

7. Пояснення до заповіту

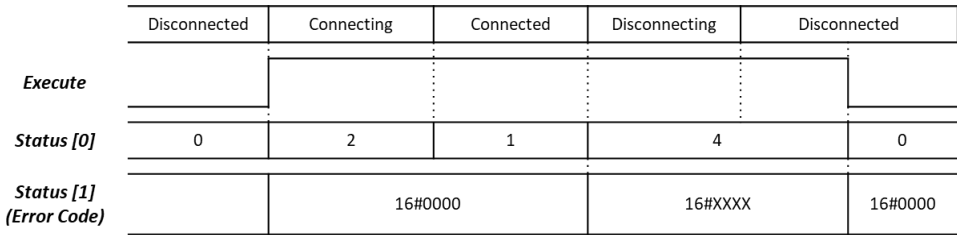
- (1) Повідомлення **Will** охоплює параметри **WillTopic** , **WillMessage** , **WillRetain** , **WillQos** і **WillFlag** .
- (2) Щоб запустити функцію **Will Message** , вам потрібно налаштувати зазначені вище параметри, встановити **WillFlag** на ON, а потім встановити з'єднання. Коли пристрій ненормально від'єднано (сервер не отримує команди роз'єднання, але виявляє, що з'єднання перервано), сервер надішле повідомлення Will іншим пристроям, які підписалися на тему повідомлення Will.
- (3) Характеристики **WillTopic** і **WillMessage** такі ж, як і для загальної публікації повідомлень. Додаткову інформацію див. у **темі** та **повідомленні** в **MQTT_Publish** (API2215).
- (3) Характеристика **WillRetain** така ж, як і загальне збереження повідомлень. Зверніться до **Retain in MQTT_Publish** (API2215) для деталей.
- (5) Характеристика **WillQos** така ж, як Qos (якість обслуговування) загальних повідомлень. Зверніться до **Qos в MQTT_Publish** (API2215) для деталей.
- (6) Коли **WillFlag** має значення FALSE, для **WillRetain** має бути встановлено значення OFF, а для **WillQos** має бути встановлено значення 0.

8. Часові діаграми

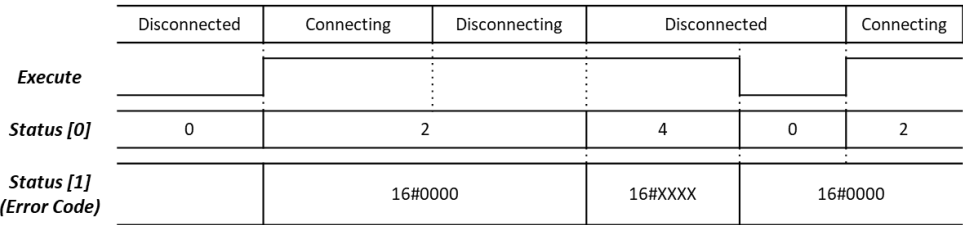
(1) Нормально підключений і відключений



(2) Відключено під час спілкування



(3) Не вдалося встановити з'єднання



9. Коды помилок і пояснення наведені в наступній таблиці:

(Для **MQTT_Connect** (API2214), **MQTT_Publish** (API2215) і **MQTT_Subscribe** (API2216))

Код помилки	Пояснення	Рішення
0x2200	SID поза межами діапазону.	Змініть значення вхідної змінної SID .
0x2201	Функціональна карта 2 не є AS-FFTP01.	Перевірте, чи правильний тип картки функціональної картки 2.
0x2202	BrokerNum поза діапазоном.	Змініть значення вхідної змінної BrokerNum .

Код помилки	Пояснення	Рішення
0x2207	Для зв'язку виконуються відповідні інструкції MQTT.	Перевірте порядок виконання інструкцій MQTT; чекати для завершення виконання (Status[0] = 0, 1 або 4), а потім виконати наступну інструкцію MQTT.
0x2208	Тайм-аут відбувається під час спілкування між ПЛК і AS-FFTP01.	Налаштуйте час очікування зв'язку в HWCONFIG для функціональної карти 2.
0x2209	AS-FFTP01 зайнятий.	Спробуйте ще раз пізніше.
0x220A	AS-FFTP01 не готовий (SM172 OFF).	Експортуйте файл системного журналу за допомогою діагностики системи веб-сторінки AS-FFTP01 і зверніться до виробника для аналізу.
0x220B	Довжина рядка теми поза межами діапазон.	Змініть довжину рядка вхідної змінної Тема .
0x220C	Довжина рядка повідомлення виходить за межі діапазону.	Змініть довжину рядка вхідної змінної повідомлення .
0x220D	Qos поза діапазоном.	Змініть значення вхідної змінної Qos .
0x8001	Принаймні один параметр недійсний.	Перевірте, чи відповідають усі параметри інструкції API до специфікації використання.
0x8002	Буферна зона заповнена.	Розмір отриманих пакетів перевищує 1024 байт.
0x8003	Не вдалося надіслати пакети MQTT.	Перевірте, чи нормально працює мережа Ethernet.
0x8004	Не вдалося отримати пакети MQTT.	Перевірте, чи нормально працює мережа Ethernet.
0x8005	Від сервера отримано недійсний пакет.	● Перевірте, чи працює сервер нормально. ● Експортуйте файл системного журналу через системну діагностику веб-сторінки AS-FFTP01 і зверніться до виробника для аналізу.
0x8006	Сервер відмовив у CONNECT або ПІДПИСАТИСЯ.	Перевірте, чи нормальний робочий стан сервера.
0x8008	Зафіксовано незаконний стан	Експортуйте файл системного журналу за допомогою діагностики системи веб-сторінки AS-FFTP01 і зверніться до виробника для аналізу.
0x8009	Отриманий пакет не узгоджений з тим, що записано.	Використовуйте MQTT_Connect , щоб від'єднатися, а потім повторно підключіться, якщо стан з'єднання нестабільний.
0x800A	Час очікування PINGRESP	● Перевірте, чи нормально працює мережа Ethernet. ● Перевірте, чи нормально працює сервер.

Код помилки	Пояснення	Рішення
0x800B	Не вдалося встановити підключення TLS	● Перевірте, чи нормально працює мережа Ethernet. ● Перевірте, чи правильна адреса кінцевої точки сервера. ● Перевірте, чи правильні імпортований кореневий сертифікат, сертифікат пристрою та закритий ключ. ● Перевірте, чи працює сервер. ● Перевірте, чи налаштування мережевого брандмауера дозволяють використовувати порт зв'язку 8883.
0x800C	Не вдалося встановити підключення MQTT	Перевірте стан мережі та повторно виконайте MQTT_Connect для встановлення з'єднання.
0x800D	Відсутні дані в адресі кінцевої точки сервера, кореневому сертифікаті, сертифікаті пристрою чи закритому ключі	Для налаштування перейдіть до параметрів користувача MQTT у HWCONFIG адресу кінцевої точки сервера, кореневий сертифікат, сертифікат пристрою та закритий ключ, а потім їх завантаження.
0x800E	Тема підписки порожня.	Тема підписки не може бути пустою.
0x800F	Досягнуто максимальної кількості підписаних тем.	Після скасування підписки на непотрібні теми знову активуйте функцію підписки.
0x8010	QoS 2 не підтримується.	Змініть рівень QoS на 1 або 0.
0x8011	Тема вже підписана.	На ту саму тему не можна підписатися двічі.

6

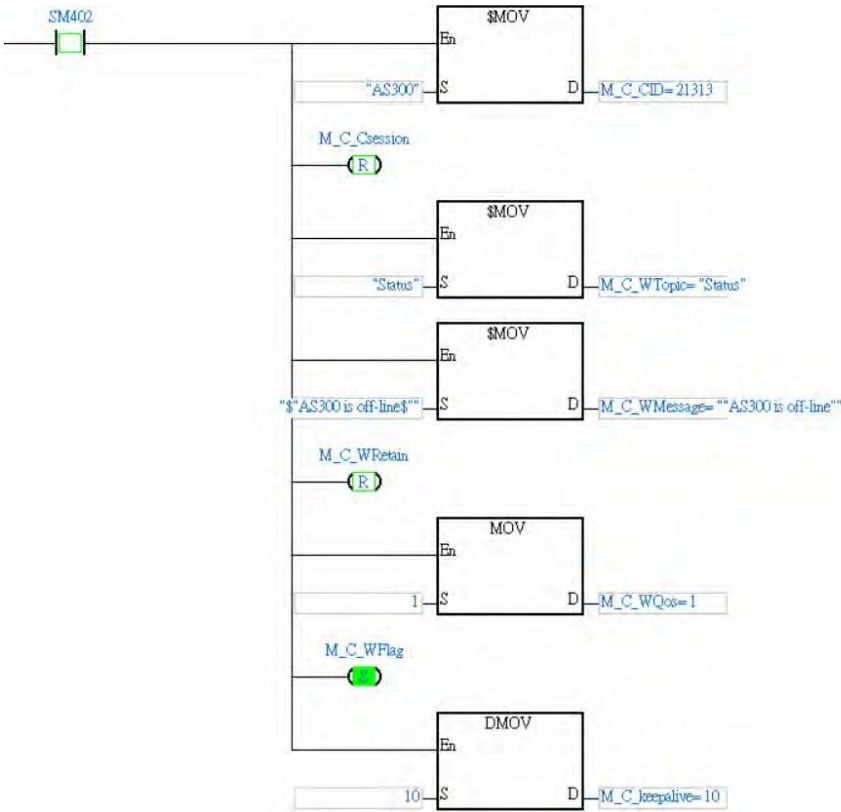
приклад

1. Зверніться до опису функцій AS-FFTP01 у Посібнику модуля серії AS, щоб налаштувати адресу кінцевої точки сервера MQTT, сертифікат ключа та сертифікат пристрою в HWCONFIG, а потім завантажте параметри для функціональної карти AS-FFTP01.

2. Налаштування та опис параметрів

Параметр	Налаштування	опис
SID	34	Фіксовані значення параметрів
BrokerNum	1	
ClientID	"AS300"	Сервер підключить сеанс до ідентифікатора клієнта: "AS300".
CleanSession	OFF	Сеанс не очищається, а вміст підписки з останнє з'єднання зберігається після повторного встановлення з'єднання.
WillTopic	«Status»	● Клієнти, які підписалися на тему «Статус» повідомлення Will, отримають повідомлення «AS300 is off-line» у разі втрати зв'язку.
WillMessage	"\$AS300 вимкнено\$"	

Параметр	Налаштування	опис
		Під час надсилання рядкових символів слід брати подвійні лапки. Вам потрібно додати \$ перед подвійними лапками (" "), щоб визначити подвійні лапки як рядкові символи, тобто "\$"AS300 вимкнено".
WillRetain	OFF	Сервер не зберігатиме повідомлення Will.
WillQos	1	Повідомлення Will використовує QoS рівня 1 для зв'язку та абоненти отримують повідомлення Will принаймні один раз.
WillFlag	ON	Функцію повідомлення про волю ввімкнено.
KeepAlive	10	Коли час відключення перевищує 15 секунд (1,5 рази KeepAlive), сервер активно закриє з'єднання, і підписники, які підписалися на тему повідомлення Will, отримують повідомлення Will.



3. Тригер для виконання інструкції: послідовно встановіть для **En** і **Execute** значення **ON**, коли **Status[0] = 1**, це означає, що підключення завершено.



4. Публікація та підписка: **Status [0] = 1** можна встановити як умову для **En** MQTT_Publish (API2215) і **MQTT_Subscribe** (API2216). Функції «Опублікувати» та «Підписатися» можна ввімкнути лише після встановлення з'єднання. Див. **MQTT_Publish** (API2215) і **MQTT_Subscribe** (API2216). для встановлення параметрів і прикладів роботи.
5. Якщо для параметра **Execute** встановлено значення OFF, AS-FFTP01 надсилатиме запит на відключення. Коли **Status[0] = 0**, це означає, що від'єднання виконано. Це стандартний процес відключення.
6. Коли AS-FFTP01 відключається від сервера без попередження, клієнти, які підписалися на тему «Status», отримують повідомлення «AS300 is offline».

API	Інструкція				Операнд							функція						
2215		MQTT_Publish				Execute ~ ErrCode							Публікація повідомлення MQTT					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
Execute	●	●	●	●				●		●	●							
SID								●	●		●		●					
BrokerNum								●	●		●		●					
Topic								●	●		●				●			
Message								●	●		●				●			
Size								●	●		●		●					
Qos								●	●		●		●					
Retain	●	●	●	●				●		●	●							
Dup	●	●	●	●				●		●	●							
Done		●	●	●				●		●	●							
Busy		●	●	●				●		●	●							
Error		●	●	●				●		●	●							
ErrCode								●			●							

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Execute ~ ErrCode	Зверніться до пояснення наступної інструкції.												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

MQTT_Publish
En
Execute
Done
SID
Busy
BrokerNum
Error
Topic
ErrCode
Message
Size
Qos
Retain
Dup

Execute ~ ErrCode : Дивіться інструкцію пояснення нижче.

Пояснення

1. Ця інструкція доступна для ISPSOft V3.16 або новішої версії та ПЛК AS300 із мікропрограмою версії 1.12 або пізнішої, якщо на ПЛК встановлено функціональну карту AS-FFTP01.
2. Зверніться до опису функцій на AS-FFTP01 у Посібнику з модуля серії AS або прикладі інструкцій нижче та налаштуйте параметри MQTT для AS-FFTP01 у HWCONFIG перед використанням.
3. Перш ніж запускати **MQTT_Publish** , переконайтеся, що підключення успішно встановлено. Ви можете використовувати код стану **MQTT_Connect** , щоб побачити, чи можна запустити **MQTT_Publish** . Для отримання детальної інформації зверніться до наведених нижче прикладів операцій.
4. Імена та описи операндів наведено в наступній таблиці.

Операнд	Ім'я	опис	Тип даних	Зауваження
Execute	Виконати	● Коли Execute змінюється на ON, start MQTT_Опублікувати	BOOL	
SID	Номер слота	Фіксоване значення: 34 (Представляє функціональну картку AS-FFTP01)	WORD	
BrokerNum	ID сервера (брокера).	Фіксоване значення:1	WORD	
Topic	Тема	● Максимальна довжина 100 символів, включаючи символ закінчення рядка (16#00). ● Вміст теми не може бути порожнім.	STRING[100]	
Message	повідомлення	● Максимальна довжина 256 символи, включно з символом закінчення рядка (16#00) Опис розташування даних: ● Size = 0: від молодшого до старшого байта ● Size ≠ 0: від старшого до молодшого байта	WORD[128]	
Size	Довжина даних повідомлення	● 0: вказує на тип повідомлення даними є рядок, який визначається термінатором (16#00). ● 1-256: вказує, що тип даних повідомлення є вихідними даними в шістнадцятковій системі.	WORD	

Операнд	Ім'я	опис	Тип даних	Зауваження
Qos	Якість обслуговування	Якість обслуговування повідомлення, Діапазон налаштувань: 0, 1	WORD	Дивіться Qos пояснення нижче.
Retain	Зберігати	Встановіть, чи зберігати повідомлення на сервері чи ні.	BOOL	Див. Зберегти пояснення функції нижче.
Dup	Прапор DUP	Користувачі можуть зробити позначку, щоб вказати, чи є опубліковане повідомлення є дублікатом повідомлення чи ні. Як правило, встановлено значення OFF. Коли QoS встановлено на 0, Dup має бути встановлено на OFF. ● OFF: повідомлення публікується вперше. ● ON: повідомлення позначено як дублікат.	BOOL	
Done	Виконано	Спілкування завершено.	BOOL	
Busy	Зайняте	Зв'язок зайнятий.	BOOL	
Error	Помилка	Виникає помилка та відображається код помилки вгору.	BOOL	
ErrCode	Код помилки	Щоб отримати коди помилок, зверніться до MQTT_Connect (API2214).	WORD	

5. QoS (Якість обслуговування)

QoS — це симетрична комунікаційна угода. Видавці та передплатники можуть встановлювати власні рівні QoS, а різні рівні QoS визначають різні передачі або прийоми. AS-FFTP01 підтримує Qos рівня 1 і 2.

(1) QoS=0: Це не гарантує, що повідомлення буде отримано. Отримання повідомлення залежить від стану мережі.

- Видавець надсилає повідомлення лише один раз.
- Абонент не відповідає, отримано повідомлення чи ні.
- Під час публікації повідомлення з **QoS**, встановленим на 0, **для Dup** має бути встановлено значення OFF.

(2) QoS=1: гарантує, що повідомлення буде отримано принаймні один раз.

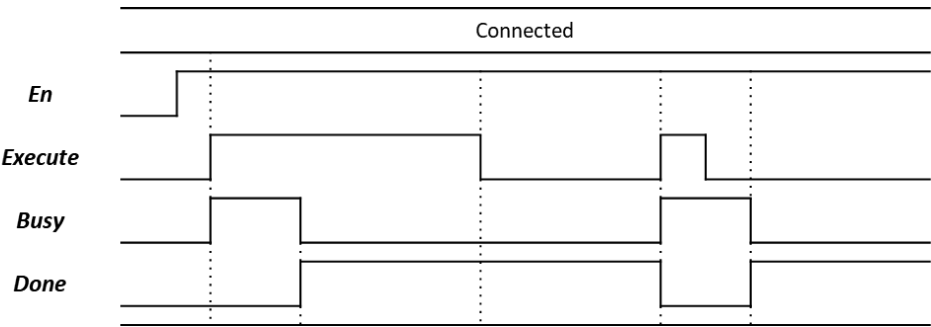
- Після отримання повідомлення абонент відповість пакетом підтвердження (PUBACK Packet).

- Видавець і сервер зберігатимуть повідомлення, доки не буде отримано пакет підтвердження від передплатника.
- Повторне надсилання повідомлення: якщо підписник не отримує повідомлення після публікації повідомлення, повідомлення буде автоматично надіслано повторно, якщо сеанс не буде очищено (для **CleanSession** встановлено значення OFF), коли з'єднання буде успішно відновлено.

6. Зберігає функцію.

- (1) Коли опубліковане повідомлення зберігається (**Retain** = ON), сервер збереже повідомлення та його дані QoS і доставить останнє збережене повідомлення всім новим підписникам теми в майбутньому.
- (2) Коли опубліковане повідомлення зберігається (**Retain** = ON), сервер НЕ збереже повідомлення, а збережені повідомлення НЕ будуть видалені чи замінені.

7. Часова діаграма

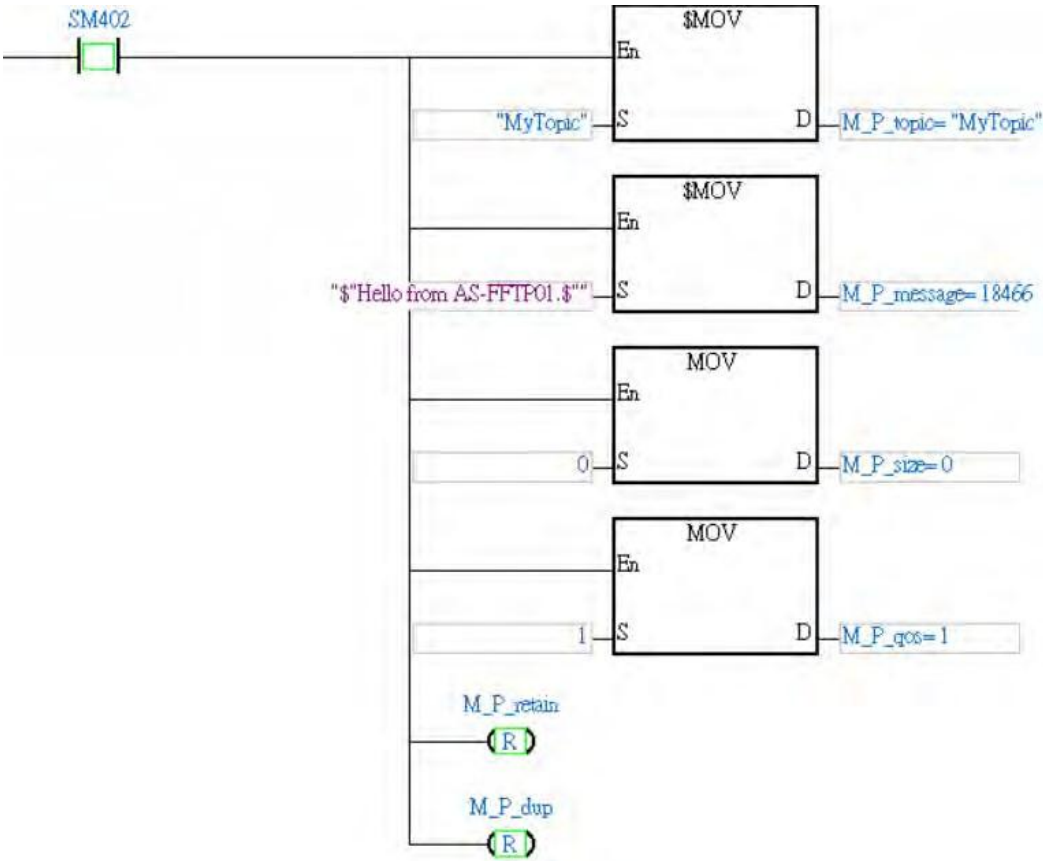


приклад

1. Щоб встановити з'єднання, зверніться до **MQTT_Connect (API2214)**.
2. Налаштування та опис параметрів

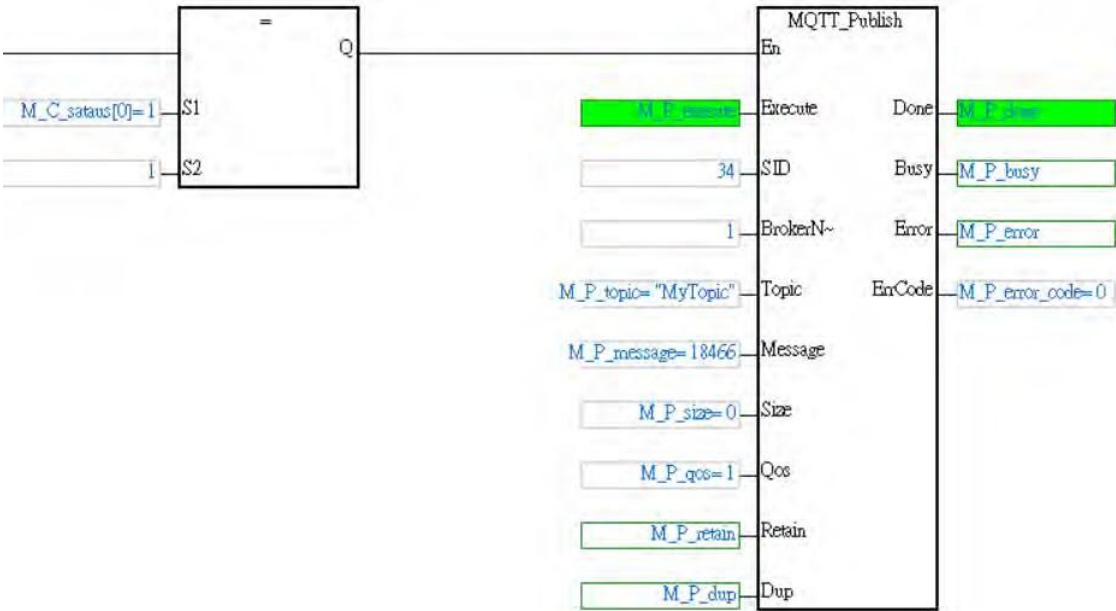
Параметр	Налаштування	опис
SID	34	Фіксовані значення параметрів
BrokerNum	1	
Topic	«Моя тема»	● Клієнти, які підписалися на тему "MyTopic", отримають повідомлення "Привіт від AS- FFTP01". ● Під час надсилання рядкових символів слід брати подвійні лапки. Вам потрібно додати \$ перед подвійними лапками (" "), щоб ідентифікувати подвійні лапки як символи рядка.
повідомлення	"\$Привіт від AS-FFTP01.\$"	

Параметр	Налаштування	опис
Size	0	Значення параметра 0 означає надсилання повідомлення рядка.
Qos	1	Це гарантує, що повідомлення буде отримано принаймні один раз.
Retain	OFF	Повідомлення НЕ буде збережено.
Dup	OFF	Повідомлення НЕ позначено як дублікат.



3. Тригер для виконання інструкції

Коли **Status[0] = 1 MQTT_Connect** , **En** увімкнено; це означає, що з'єднання встановлено. Коли **En** увімкнено, ви можете виконати **MQTT_Publish** . Після того, як для параметра «**Execute**» встановлено значення «ON», публікація даних завершена, а « **Done** » ввімкнено.



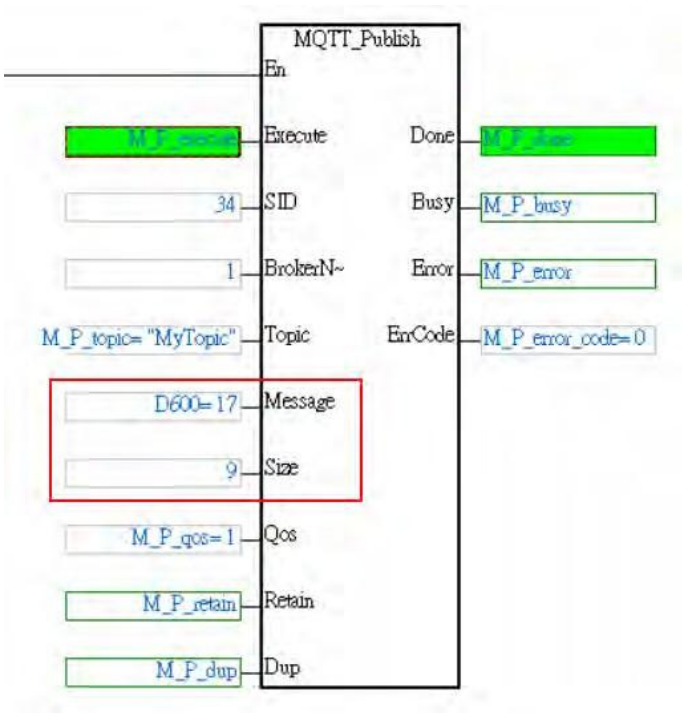
4. Щоб надіслати рядкове повідомлення (**Size** = 0), спочатку будуть передані дані молодшого байту, а потім дані старшого байту. передається. Дивіться приклад передачі даних, показаний нижче.

M_P_message[0]		WORD	H"	leH"	ASCII	▼
M_P_message[1]		WORD	le	olle	ASCII	▼
M_P_message[2]		WORD	ol	f ol	ASCII	▼
M_P_message[3]		WORD	f	orf	ASCII	▼
M_P_message[4]		WORD	or	mor	ASCII	▼
M_P_message[5]		WORD	m	SA m	ASCII	▼
M_P_message[6]		WORD	SA	F-SA	ASCII	▼
M_P_message[7]		WORD	F-	TFF-	ASCII	▼
M_P_message[8]		WORD	TF	0PTF	ASCII	▼
M_P_message[9]		WORD	0P	.10P	ASCII	▼
M_P_message[10]		WORD	.1	*1.1	ASCII	▼
M_P_message[11]		WORD	*1	***1	ASCII	▼
M_P_message[12]		WORD	**	****	ASCII	▼

Клієнти, які підписалися на тему "MyTopic", отримають повідомлення "Привіт від AS-FFTP01". Візьміть дані, отримані за допомогою **MQTT_Subscribe** як приклад, показаний нижче.

M_S_R_message[1]		WORD	"H	el"H	ASCII	▼
M_S_R_message[2]		WORD	el	loel	ASCII	▼
M_S_R_message[3]		WORD	lo	flo	ASCII	▼
M_S_R_message[4]		WORD	f	ro f	ASCII	▼
M_S_R_message[5]		WORD	ro	m ro	ASCII	▼
M_S_R_message[6]		WORD	m	A.Sm	ASCII	▼
M_S_R_message[7]		WORD	AS	-FAS	ASCII	▼
M_S_R_message[8]		WORD	-F	FT-F	ASCII	▼
M_S_R_message[9]		WORD	FT	P0FT	ASCII	▼
M_S_R_message[10]		WORD	P0	1.P0	ASCII	▼
M_S_R_message[11]		WORD	1.	"*1.	ASCII	▼
M_S_R_message[12]		WORD	"*	***"	ASCII	▼

5. Навіть якщо абонент відключено під час публікації повідомлення та не може отримати повідомлення, поки рівень QoS абонента дорівнює 1, повідомлення «Привіт від AS-FFTP01». буде повторно надіслано абоненту, якщо сеанс не буде очищено (для **CleanSession** встановлено значення OFF) після відновлення з'єднання.
6. Якщо для параметра **Retain** встановлено значення ON, з'явиться повідомлення «Привіт від AS-FFTP01». отримають нові передплатники теми "MyTopic" після підключення.
7. Для передачі нерядкового повідомлення (**розмір** $\neq$ 0) спочатку передадуться дані старшого байту, а потім дані молодшого байту. Ви можете вказати довжину даних у **M_P_size**. Змініть наведений вище приклад, щоб опублікувати 9 байтів, починаючи з D600, і порівняйте вихідні дані з даними, отриманими через **MQTT_Subscribe** , як показано нижче.



	D600		0011	22330011
	D601		2233	44552233
	D602		4455	66774455
	D603		6677	88006677
	D604		8800	00008800
M_S_R_message[0]		WORD	9	1114121
M_S_R_message[1]		WORD	0011	22330011
M_S_R_message[2]		WORD	2233	44552233
M_S_R_message[3]		WORD	4455	66774455
M_S_R_message[4]		WORD	6677	88006677
M_S_R_message[5]		WORD	8800	00008800
M_S_R_message[6]		WORD	0000	00000000

API	Інструкція				Операнд							функція					
2216		MQTT_Subscribe				Execute ~ RecvTopic							Підписка на тему MQTT				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Execute	●	●	●	●				●		●	●						
SID								●	●		●		●				
BrokerNum								●	●		●		●				
Topic								●	●		●				●		
Qos								●	●		●		●				
Status								●			●						
Message								●			●						
RecvTopic								●			●						

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Execute ~ RecvTopic	Зверніться до пояснення наступної інструкції.												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

MQTT_Subscribe	
En	
Execute	Status
SID	Message
BrokerNum	RecvTopic
Topic	
Qos	

Execute ~ RecvTopic : Дивіться пояснення інструкцій нижче.

Пояснення

- 1. Ця інструкція доступна для ISPSOft V3.16 або новішої версії та ПЛК AS300 із мікропрограмою версії 1.12 або пізнішої, якщо на ПЛК встановлено функціональну карту AS-FFTP01.
- 2. Зверніться до опису функцій на AS-FFTP01 у Посібнику з модуля серії AS або прикладі інструкцій нижче та налаштуйте параметри MQTT для AS-FFTP01 у HWCONFIG перед використанням.

3. Перед запуском **MQTT_Subscribe** переконайтеся, що підключення успішно встановлено. Ви можете використовувати код стану **MQTT_Connect**, щоб побачити, чи можна запустити **MQTT_Subscribe**. Для отримання детальної інформації зверніться до наведених нижче прикладів операцій.

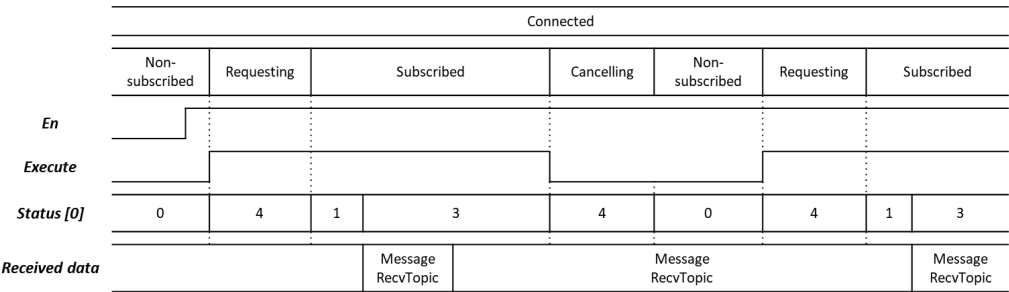
4. Імена та описи операндів наведено в наступній таблиці.

Операнд	Ім'я	опис	Тип даних	Зауваження
Execute	Виконати	- Коли Execute змінюється на ON, запустить MQTT_Subscribe. - Коли Execute змінюється на OFF, зупинить MQTT_Subscribe.	BOOL	
SID	Номер слота	Фіксоване значення: 34 (Представляє функціональну картку AS-FFTP01)	WORD	
BrokerNum	Сервер (Брокер) ID	Фіксоване значення 1	WORD	
Topic	Тема	- Максимальна довжина 100 персонажів. Знак закінчення (16#00) обов'язковий у кінці рядка. - Вміст теми не може бути порожнім.	STRING[100]	
Qos	Якість обслуговування	Якість обслуговування повідомлення, Діапазон налаштувань: 0, 1	WORD	Зверніться до Qos пояснення в MQTT_Publish (API2215).
Status	Держава	- Status[0]: код стану 0: Скасовано підписку 1: Підписано 3: Отримано нове повідомлення 4: Підписка/скасування підписки 8: Сталася помилка та з'явився код помилки. - Status [1]: код помилки Перегляньте опис коду помилки в MQTT_Connect (API 2214).	WORD[2]	Два послідовних слова зайняті. Див . Час діаграми нижче для часу оновлення.
Message	повідомлення	Повідомлення[0]: Розмір отриманого повідомлення;	WORD[129]	

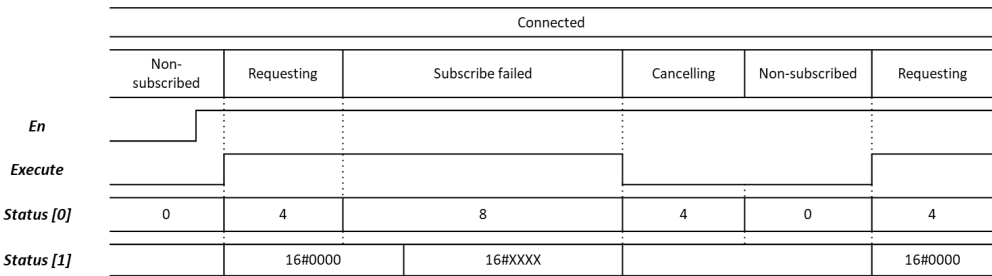
Операнд	Ім'я	опис	Тип даних	Зауваження
		● Повідомлення[1]: Зміст отримане повідомлення; Максимум 256 символів; Дані розташовуються в порядку від старшого байта до молодшого байта, а в кінці ставиться знак закінчення (16#00).		
RcvTopic	Отримана тема	При підписці на кілька тем на водночас, необхідно визначити тему кожного джерела повідомлення. ● RcvTopic[0]: Розмір теми джерела повідомлення; ● RcvTopic[1]: Зміст отриманої теми; Максимум 100 символів. Дані розташовуються в порядку від старшого байта до молодшого байта, а в кінці ставиться знак закінчення (16#00).	WORD[51]	

Часові діаграми

(1) Підписався, отримав і скасував підписку



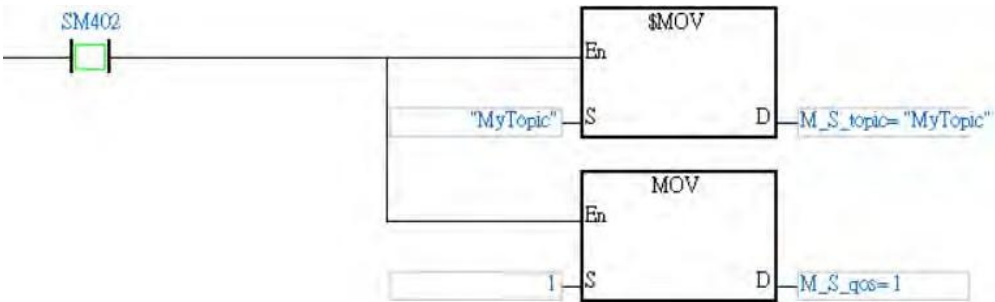
(2) Не вдалося підписатися



приклад

- 1. Зверніться до **MQTT_Connect** (API2214), щоб встановити з'єднання.
- 2. Налаштування та опис параметрів

Параметр	Налаштування	опис
SID	34	Фіксовані значення параметрів
BrokerNum	1	
Topic	«Моя тема»	Тема, на яку варто підписатися
Qos	1	Це гарантує, що повідомлення буде отримано принаймні один раз.



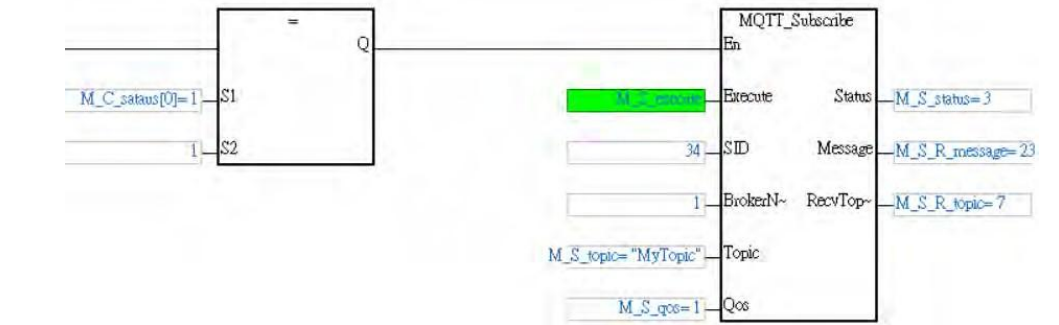
3. Тригер для виконання інструкції

Коли **Status[0] = 1 MQTT_Connect** , **En** увімкнено; це означає, що з'єднання встановлено. Коли **En** увімкнено, ви можете виконати **MQTT_Subscribe** . Після того, як для параметра **«Execute»** встановлено значення **«ON»**, підписка є успішною, а **«Status [0]» = 1**



4. Отримати дані

Після того як видавець надішле повідомлення до «MyTopic» (див. приклад у **MQTT_Publish** (API2215)), функціональна картка AS-FFTP01 отримає повідомлення «Привіт від AS-FFTP01.». Довжина даних теми та повідомлення зберігається в байтах у **Message[0]** та **RecvTopic[0]**, а назва теми та вміст повідомлення зберігаються в **Message[1]** та **RecvTopic[1]** за порядком.



MQTT	M_S_R_topic[0]		WORD	7	Signed Decimal	▼
MQTT	M_S_R_topic[1]		WORD	My	ASCII	▼
MQTT	M_S_R_topic[2]		WORD	To	ASCII	▼
MQTT	M_S_R_topic[3]		WORD	pi	ASCII	▼
MQTT	M_S_R_topic[4]		WORD	c*	ASCII	▼
MQTT	M_S_R_topic[5]		WORD	**	ASCII	▼
MQTT	M_S_R_message[0]		WORD	23	Signed Decimal	▼
MQTT	M_S_R_message[1]		WORD	"H	ASCII	▼
MQTT	M_S_R_message[2]		WORD	el	ASCII	▼
MQTT	M_S_R_message[3]		WORD	lo	ASCII	▼
MQTT	M_S_R_message[4]		WORD	f	ASCII	▼
MQTT	M_S_R_message[5]		WORD	ro	ASCII	▼
MQTT	M_S_R_message[6]		WORD	m	ASCII	▼
MQTT	M_S_R_message[7]		WORD	AS	ASCII	▼
MQTT	M_S_R_message[8]		WORD	-F	ASCII	▼
MQTT	M_S_R_message[9]		WORD	FT	ASCII	▼
MQTT	M_S_R_message[10]		WORD	P0	ASCII	▼
MQTT	M_S_R_message[11]		WORD	1.	ASCII	▼
MQTT	M_S_R_message[12]		WORD	**	ASCII	▼

5. Скасувати підписку: якщо для **параметра «Execute»** встановлено значення OFF, на сервер буде надіслано пакет «Скасувати підписку» на тему, указану в « **Темі** ». Якщо **Status[0]** дорівнює 0, підписку на тему успішно скасовано.
6. Якщо ви НЕ виконали функцію скасування підписки, навіть якщо з'єднання припиняється або ПЛК переходить у стан STOP, ви все одно можете отримувати повідомлення теми, на яку раніше підписалися, після відновлення з'єднання ПЛК повертається до стану RUN і **CleanSession** встановлено на OFF. Якщо ви знову виконаєте **MQTT_Subscribe**, коли станеться втрата з'єднання, ви отримаєте код помилки 16#8011, який повідомляє, що на цю тему вже підписані.
7. Якщо рівень QoS для повідомлення теми, на яку ви підписані, дорівнює 1, повідомлення буде отримано принаймні один раз після втрати з'єднання. Переконайтеся, що сеанс зберігається (для **CleanSession** встановлено значення OFF), коли з'єднання буде відновлено, а рівень QoS опублікованого повідомлення також дорівнює 1; повідомлення буде доставлено повторно.

6.23 Інструкції з реєстрації карти пам'яті/файлу

6.23.1 Список інструкцій щодо реєстрації картки пам'яті/файлу

У наведеній нижче таблиці наведено інструкції щодо картки пам'яті/реєстрації файлів, які розглядаються в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
<u>2300</u>	MWRIT	–	✓	Запис даних з ПЛК на карту пам'яті
<u>2301</u>	MREAD	–	✓	Зчитування даних з карти пам'яті в ПЛК
<u>2302</u>	MTWRIT	–	✓	Записування рядка на карту пам'яті
<u>2303</u>	MEMW	–	✓	Запис даних у файловий реєстр
<u>2304</u>	MDEL	–	✓	Видалення файлів на карті пам'яті
<u>2305</u>	Record_Trigger	–	–	Запис і запис даних у файл

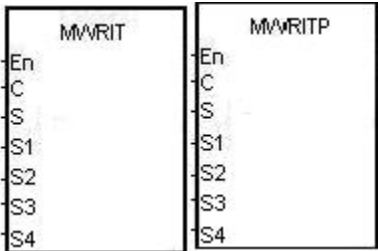
6.23.2 Пояснення інструкцій щодо картки пам'яті/реєстрації файлів

API		Код інструкції			Операнд					функція						
2300			MWRIT	P	C, S, S ₁ , S ₂ , S ₃ , S ₄					Запис даних з ПЛК на карту пам'яті						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
C								●	●				○	○		
S								●	●							
S ₁								●	●				○	○		
S ₂								●	●				○	○		
S ₃								●	●						○	
S ₄								●	●				○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
C		●											
S		●											
S ₁			●										
S ₂		●											
S ₃													●
S ₄			●										

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

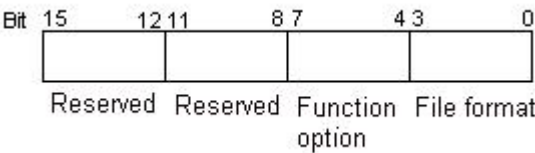


- C :Контрольний параметр
- S :Джерело даних
- S₁ :Довжина даних
- S₂ :Попередня лінія
- S₃ :Ім'я файлу
- S₄ :Адреса даних у файлі

Пояснення

1. Ця інструкція записує дані з ПЛК на карту пам'яті. Операнди описані наступним чином.

- C : контрольний параметр



Пункт	Код	опис
Формат файлу	0	Двійкова величина
		Значення за замовчуванням
		Розширення імені файлу – .dmd.
		Одиницею величини є слово.
	1	Значення розділяються комою.
		Одиницею величини є слово.
		Розширення імені файлу – .cvs.
		Використовуйте коди ASCII..
		Значення, яке зберігається, має шістнадцятковий формат.
	2	Значення розділяються комою.
		Одиницею величини є подвійне слово.
		Розширення імені файлу – .cvs.
		Використовуйте коди ASCII.
		Значення, яке зберігається, має шістнадцятковий формат.
	3	Значення розділені табуляцією.
		Одиницею величини є слово.
		Розширення імені файлу – .txt.
		Використовуйте коди ASCII.
		Значення, яке зберігається, має шістнадцятковий формат.
	4	Значення розділені табуляцією.
		Одиницею величини є подвійне слово.
		Розширення імені файлу – .txt.
		Використовуйте коди ASCII.
		Значення, яке зберігається, має шістнадцятковий формат.
	5	Значення не розділені жодним знаком.
		Одиницею величини є слово.
		Розширення імені файлу – .txt.
		Використовуйте коди ASCII.
		Значення, яке зберігається, має шістнадцятковий формат.
	6	Значення не розділені жодним знаком.

Пункт	Код	опис
		Одиницею величини є подвійне слово.
		Розширення імені файлу – .txt.
		Використовуйте коди ASCII.
		Значення, яке зберігається, має шістнадцятковий формат.
	7	Значення розділяються комою. (Підтримується версія 1.04.00 або новіша)
		Одиницею значення є слово.
		Розширення імені файлу – .csv.
		Використовуйте коди ASCII.
		Значення, яке зберігається, має десятковий формат.
	Варіант функції	Додавання Дані, які записуються на карту пам'яті, додаються після останнього значення у файлі. Це значення за умовчанням Якщо файл не існує, він створюється автоматично.
		1 Перезапис Дані, записані на карту пам'яті, замінюють значення у файлі, починаючи зі значення, зазначеного значенням у S_4 . Якщо файл не існує, він створюється автоматично.
		Зарезервований
		-
		Значення бітів 8 – бітів 15 дорівнюють 0.

- S : джерело даних
- S_1 : довжина даних, які інструкція записує у файл

Якщо значення в S_1 дорівнює 0, інструкція не записує жодних даних у файл.

Пункт	опис
Одиниця величини	Якщо формат файлу 0, 1, 3, 5 або 7, одиницею значення є слово. Якщо формат файлу 2, 4 або 6, одиницею значення є подвійне слово.
Одиниця параметра	Подвійне слово
Довжина даних	Пристрої, на яких інструкція зберігає дані, не можуть перевищувати пристрій діапазон, а розмір даних, які інструкція записує у файл, не може перевищувати чотири гігабайти. Зверніться до розділу 2 для отримання додаткової інформації про пристрої.

- S_2 : просування лінії; значення в S_2 має бути між 0–256.
- Прошивка AS CPU версії 1.04.00 або новішої підтримує використання рядка (менше 9 символів) у імені файлу S_3 . Версія мікропрограми 1.08.30 або новіша підтримує до 200 символів (включно з 16#00) в імені файлу S_3 .
- $S_3 - S_3 + 4$: S_3 займає п'ять пристроїв. Ім'я файлу містить максимум дев'ять символів, включаючи 16#00. Якщо рядок не закінчується на 16#00, виникає помилка. Коли інструкція читає символ закінчення, вона припиняє читання та перевіряє, чи ім'я файлу є правильним. Символи в імені файлу можуть бути A–Z, a–z та 0–9. Розширення імені файлу залежить від формату файлу. Файл, створений інструкцією, знаходиться в папці за замовчуванням. Якщо ім'я файлу «Test1», інструкція запише символи в пристрої таким чином.

ASCII					
S_3	'e'	"T"	Код →	S_3	16#65
$S_3 + 1$	't'	's'		$S_3 + 1$	16 #74
$S_3 + 2$	НУЛЬ	'1'		$S_3 + 2$	16 №00
					16#31

- Шлях до папки за замовчуванням:

Назва моделі	Шлях до папки
Серія AS300	PLC CARD\AS300\UserProg
Серія AS200	PLC CARD\AS200\UserProg

- S_4 : значення у файлі, яке інструкція перезаписує, позначається значенням у S_4 .

Пункт	опис
Одиниця величини	Якщо формат файлу 0, 1, 3, 5 або 7, одиницею значення є слово. Якщо формат файлу 2, 4, або 6, одиницею значення є подвійне слово. (див. операнд C)
Одиниця параметра	Подвійне слово.
Використання	Якщо параметр функції дорівнює 0, S_4 не використовується.
	Якщо параметр функції дорівнює 1, інструкція замінює значення у файлі даними записувати на карту пам'яті, починаючи зі значення, зазначеного значенням у S_4 .
	Значення в S_4 має вказувати на значення у файлі.
	Якщо значення в S_4 дорівнює 0, інструкція перезаписує дані у файлі, починаючи з першого значення у файлі.

2. Позначки інструкцій:

Прапор	опис
SM450	ON: карта пам'яті знаходиться в модулі CPU.
SM451	Стан перемикача захисту від запису на карті пам'яті ON: карта пам'яті захищена від запису. OFF: карта пам'яті не захищена від запису.
SM452	Дані записуються з ПЛК на карту пам'яті або дані зчитуються з карти пам'яті на ПЛК.
SM453	ON: під час роботи карти пам'яті виникає помилка. Якщо прапорець увімкнено, ви потрібно скинути його на OFF Код помилки зберігається в SR453.

3. Пов'язані коди помилок (SR453):

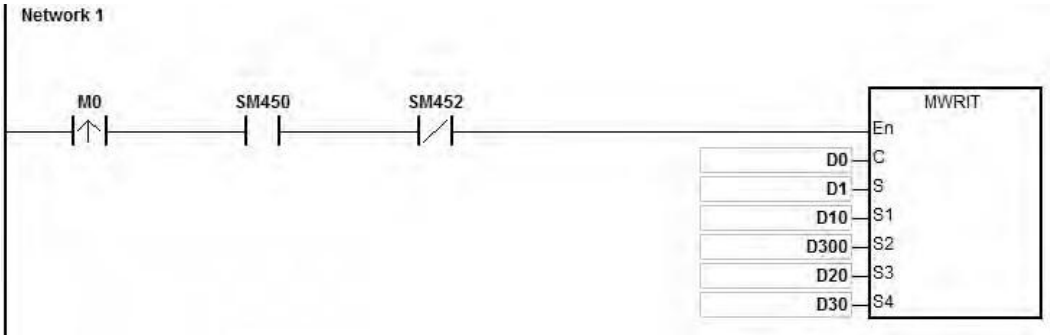
Код помилки	опис
16#005E	Під час ініціалізації карти пам'яті виникає помилка.
16#005F	Шлях неправильний, або файл не існує.
16#0060	Неможливо створити папку за замовчуванням.
16#0061	Пам'яті недостатньо.
16#0062	Карта пам'яті захищена від запису.
16#0063	Під час запису даних у файл виникає помилка.
16#0064	Неможливо прочитати дані з карти пам'яті.
16#0065	Файл призначений лише для читання.

4. Якщо формат цільового файлу на карті пам'яті дорівнює 0, формат вихідного файлу в ПЛК також має бути 0.

Інакше інструкція не може прочитати дані, а SM453 увімкнено. Те ж саме стосується інших форматів файлів.

приклад

SM450 увімкнено, коли ви вставляєте карту пам'яті в модуль ЦП; SM452 увімкнено, коли виконується інструкція MWRIT; SM452 вимкнено, коли інструкція MWRIT завершується. Ви не можете використовувати MWRITP (імпульсну інструкцію) постійно. Виконання цієї імпульсної інструкції для безперервного запису даних на картку пам'яті може перевищити обмеження запису та призвести до пошкодження карти пам'яті.



Операнд	Значення налаштування	опис
D0	16#0011	Файл, у який записуються дані Формат файлу: Значення розділяються комою. Одиницею величини є слово. Розширення імені файлу – .cvs. Використовуйте коди ASCII.
D1	-	Дані, які записуються у файл
D10, D11	16#00000030	Розмір даних, які записуються у файл, становить 48 слів.
D300	16#000A	У кожному рядку записано десять значень.
D20	D20=16#6554 D21=16#7473 D22=16#0031	Ім'я файлу «Test1».
D30, D31	16#00000000	Дані, записані на карту пам'яті, замінюють значення у файлі починаючи з першого значення.

Додаткові зауваження

1. Якщо значення в **C** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо значення в **S₁** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, і код помилки в SR0 – 16#2003.
3. Якщо значення в **S₂** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, і код помилки в SR0 – 16#2003.
4. Якщо значення в **S₃** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, і код помилки в SR0 – 16#2003.

API		Код інструкції			Операнд							функція						
2301			MREAD	P	C, S, S ₁ , S ₂ , S ₃ , D							Зчитування даних з карти пам'яті в ПЛК						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
C								●	●				○	○				
S								●	●						○			
S ₁								●	●				○	○				
S ₂								●	●				○	○				
S ₃								●	●				○	○				
D								●										

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
C		●											
S													●
S ₁			●										
S ₂		●											
S ₃			●										
D		●											

Інструкція пульсу		16-бітна інструкція	32-розрядна інструкція
AS		AS	-

СИМВОЛ

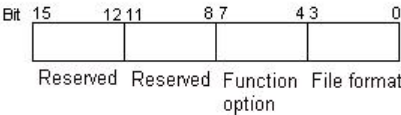


- C : Контрольний параметр
- S : Ім'я файлу
- S₁ : Адреса даних у файлі
- S₂ : Зарезервований
- S₃ : Довжина даних
- D : Призначення даних

Пояснення

1. Ця інструкція зчитує дані з карти пам'яті в ПЛК. Операнди описані наступним чином.

- C : контрольний параметр



Пункт	Код	опис
Формат файлу	0	Двійкова величина
		Значення за замовчуванням 0.
		Розширення імені файлу – .dmd.
		Одиницею величини є слово.
	1	Значення розділяються комою.
		Одиницею величини є слово.
		Розширення імені файлу – .cvs.
		Використовуйте коди ASCII.
		Значення, яке зберігається, є шістнадцятковим.
	2	Значення розділяються комою.
		Одиницею величини є подвійне слово.
		Розширення імені файлу – .cvs.
		Використовуйте коди ASCII.
		Значення, яке зберігається, є шістнадцятковим.
	3	Значення розділені табуляцією.
		Одиницею величини є слово.
		Розширення імені файлу – .txt.
		Використовуйте коди ASCII.
		Значення, яке зберігається, є шістнадцятковим.
	4	Значення розділені табуляцією.
		Одиницею величини є подвійне слово.
		Розширення імені файлу – .txt.
		Використовуйте коди ASCII.
		Значення, яке зберігається, є шістнадцятковим.
	5	Значення не розділені жодним знаком.
		Одиницею величини є слово.
		Розширення імені файлу – .txt.
		Використовуйте коди ASCII.
		Значення, яке зберігається, є шістнадцятковим.
	6	Значення не розділені жодним знаком.
		Одиницею величини є подвійне слово.

Пункт	Код	опис
		Розширення імені файлу – .txt.
		Використовуйте коди ASCII.
		Значення, яке зберігається, є шістнадцятковим.
	7	Значення розділяються комою. (Підтримується версія 1.04.00 або новіша)
		Одиницею значення є слово.
		Розширення імені файлу – .csv.
		Використовуйте коди ASCII.
		Значення, яке зберігається, має десятковий формат.
Варіант функції	0	Зчитуються значення у файлі, починаючи зі значення, зазначеного значенням у S 1.
		Значення за замовчуванням 0.
	1	Кількість значень зберігається в D і D +1.
		Якщо формат файлу 0, 1, 3 або 5, одиницею значення є слово. Якщо формат файлу 2, 4, або 6, одиницею значення є подвійне слово.
Зарезервований	-	Значення біта 8–біта 15 дорівнюють 0.

- Прошивка AS CPU версії 1.04.00 або новішої підтримує використання рядка (менше 9 символів) у імені файлу **S**.
- **S – S + 4**: **S** займає п'ять пристроїв. Ім'я файлу містить максимум дев'ять символів, включаючи 16#00. Якщо рядок не закінчується на 16#00, виникає помилка. Коли інструкція читає кінцевий символ, інструкція припиняє читати символи та перевіряє, чи ім'я файлу є правильним. Символи в імені файлу можуть бути A–Z, a–z та 0–9. Розширення імені файлу залежить від формату файлу. Файл, створений інструкцією, знаходиться в папці за замовчуванням. Якщо ім'я файлу «Test1», інструкція записує символи в пристрої таким чином.

S	"e"	"T"	Код →	S	16#6 5	16 #54
S + 1	't'	's '		S + 1	16 #74	16 #73
S+ 2	НУЛЬ	'1'		S+ 2	16 №00	16 #31

- Шлях до папки за замовчуванням:

Назва моделі	Шлях до папки
Серія AS300	PLC CARDIAS300\UserProg
Серія AS200	PLC CARDIAS200\UserProg

- **S**₁: Значення у файлі, яке зчитує інструкція, позначається значенням у **S**₁.

Пункт	опис
Одиниця величини	Зверніться до операнда С для його формату файлу. Якщо формат файлу 0, 1, 3, 5 або 7, одиниця значення - це слово. Якщо формат файлу 2, 4 або 6, одиницею значення є подвійне слово.
Одиниця параметра	Одиницею параметра є подвійне слово.
Використання	Значення в S_1 має вказувати на значення у файлі. Якщо значення в S_1 дорівнює 0, інструкція читає з першого значення у файлі.

- S_3 : довжина даних, які інструкція зчитує з файлу

Пристрої, на яких інструкція зберігає дані, не можуть перевищувати діапазон пристрою. Якщо значення в S_3 більше, ніж кількість значень у файлі, довжина даних, зчитаних із файлу, дорівнює кількості значень у файлі. Одиниця S_3 є подвійним словом.

Програмне забезпечення AS CPU версії 1.08.30 або новішої підтримує до 200 символів (включаючи 16#00) в імені файлу S_3 .

- D : початковий пристрій, на якому зберігаються дані.

2. Позначки інструкцій:

Прапор	опис
SM450	ON: карта пам'яті знаходиться в модулі CPU
SM451	Перемикач захисту від запису на картці пам'яті ON: карта пам'яті захищена від запису. OFF: карта пам'яті не захищена від запису.
SM452	Дані записуються з ПЛК на карту пам'яті або дані зчитуються з карту пам'яті в ПЛК.
SM453	ON: під час роботи карти пам'яті виникає помилка. Якщо прапорець увімкнено, потрібно скинути налаштування його на OFF Код помилки зберігається в SR453.

3. Пов'язані коди помилок (SR453):

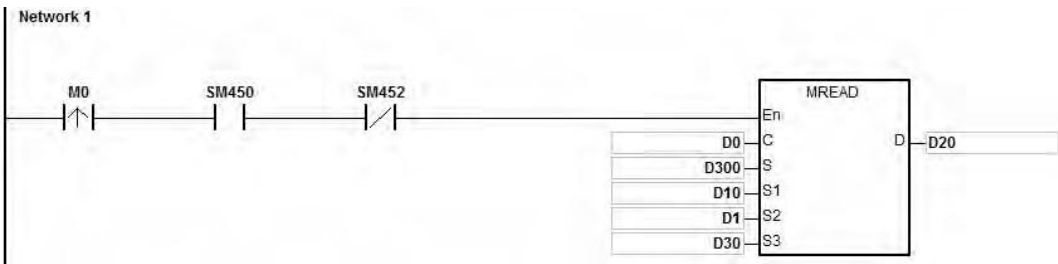
Код помилки	опис
16#005E	Під час ініціалізації карти пам'яті виникає помилка.
16#005F	Шлях неправильний, або файл не існує.
16#0060	Неможливо створити папку за замовчуванням.
16#0061	Пам'яті недостатньо.

Код помилки	опис
16#0062	Карта пам'яті захищена від запису.
16#0063	Під час запису даних у файл виникає помилка.
16#0064	Неможливо прочитати дані з карти пам'яті.

4. Якщо формат цільового файлу, в який інструкція записує дані, дорівнює 0, формат файлу, з якого Інструкція зчитує дані також має бути 0. Інакше інструкція не зможе прочитати дані, а SM453 увімкнено. Те ж саме стосується інших форматів файлів.

приклад

SM450 увімкнено, коли картку пам'яті вставлено в модуль ЦП; SM452 увімкнено, коли виконується інструкція MREAD; SM452 OFF після завершення інструкції MREAD.



Операнд	Значення налаштування	опис
D0	16#0011	Файл, з якого зчитуються дані Формат файлу: Значення розділяються комою. Одиницею величини є слово. Розширення імені файлу – .csv. Використовуйте коди ASCII.
D300	D300=16#6554 D301=16#7473 D302=16#0031	Ім'я файлу «Test1».
D10, D11	16#00000000	Зчитуються значення у файлі, починаючи з першого значення.
D1	16#000A	Десять значень зчитуються з кожного рядка.
D30, D31	16#00000020	Розмір даних, які зчитуються з файлу, становить 32 слова.
D20	-	Зчитані дані зберігаються в D20.

Додаткові зауваження

1. Якщо значення в **C** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо значення в **S₂** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо значення в **S₃** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
4. Якщо значення в **D** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд								функція					
2302		MTWRIT	P	C, S, S ₁ , S ₂ , S ₃								Записування рядка на карту пам'яті					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
C								●	●				○	○			
S								●	●								
S ₁								●	●				○	○			
S ₂								●	●				○	○			
S ₃								●	●						○		

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
C		●											
S		●											
S ₁		●											
S ₂		●											
S ₃													●

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

MTWRIT

En

C

S

S1

S2

S3

MTWRITP

En

C

S

S1

S2

S3

C : Контрольний параметр

S : Джерело даних

S₁ : Довжина даних

S₂ : Розділовий знак

S₃ : Ім'я файлу

Пояснення

1. Ця інструкція записує рядок на карту пам'яті. Операнди описані наступним чином.

- C : контрольний параметр

Параметр значення	опис
0 (Додавання)	Якщо файл існує, дані, які записуються на карту пам'яті, додаються після останнього байта у файлі.
	Якщо файл не існує, він створюється автоматично.
	Якщо файл існує, нові дані, записані на картку пам'яті, замінюють старі дані на карті пам'яті

1	файл. Розмір файлу відповідає розміру нових даних.
(Перезапис)	Якщо файл не існує, він створюється автоматично.

- **S** : джерело даних

Якщо рядок, який інструкція записує у файл, є «12345», інструкція зберігає символи в пристроях наступним чином. Основною одиницею є байт, тому інструкція зберігає перший символ у молодшому байті D300. Інструкція зберігає другий символ у старшому байті в D300. Те ж саме стосується й інших персонажів. Інструкція зберігає «16#00» у старшому байті в D300+2 і вказує на кінець рядка.

S300		S300+1		S300+2	
байт 2	байт 1	байт 4	байт 3	байт 6	байт 5
16#32	16#31	16#34	16#33	16#00	16#35

- **S₁** : Довжина даних, які записуються на карту пам'яті

Основною одиницею є байт. Пристрої, на яких інструкція зберігає дані, не можуть перевищувати діапазон пристроїв, а довжина даних, які інструкція записує на карту пам'яті, не може перевищувати 255 байт.

- **S₂** : Знак розділення

Якщо значення в **S₁** дорівнює N, інструкція записує значення в **S₂** на карту пам'яті наступним чином.

Операнд S₂		опис
Старший байт	Молодший байт	
16#00	16#00 або Не 16#00	N-байтові дані записуються у файл.
ні 16#00	16#00	У файл записуються N+1-байтові дані. Значення в старшому байті в S₂ є значенням в N+1 -му байті.
ні 16#00	ні 16#00	N+2-байтові дані записуються у файл. Значення у старшому байті в S₂ є значенням у сьомому байті, а значення в молодшому байті в S₂ є значенням у N+2 -му байті.

- Прошивка AS CPU версії 1.04.00 або новішої підтримує використання рядка (менше 9 символів) у імені файлу **S₃**. Версія мікропрограми 1.08.30 або новіша підтримує до 200 символів (включно з 16#00) в імені файлу **S₃**.
- **S₃–S₃+4**: **S₃** займає п'ять пристроїв. Ім'я файлу містить максимум дев'ять символів, включаючи 16#00. Якщо рядок не закінчується на 16#00, виникає помилка. Коли інструкція читає кінцевий символ, інструкція припиняє читати символи та перевіряє, чи ім'я файлу є правильним. Символи в імені файлу можуть бути A–Z, a–z та 0–9. Розширення імені файлу залежить від формату файлу. Файл, створений інструкцією, знаходиться в папці за замовчуванням. Якщо ім'я файлу «Test1», інструкція записує символи в пристрої таким чином.

S3	'e'	"T"	ASCII Код ➔	S3	16#65	16#54
S3 + 1	't'	's'		S3 + 1	16 #74	16#73
S3+2	НУЛЬ	'1'		S3+ 2	16 №00	16#31

- Шлях до папки за замовчуванням:

Назва моделі	Шлях до папки
Серія AS300	PLC CARD\AS300\UserProg
Серія AS200	PLC CARD\AS200\UserProg

2. Інструкційні прапорці

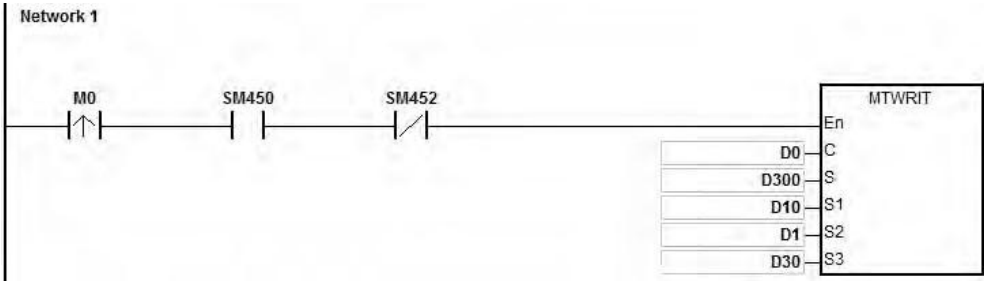
Прапор	опис
SM450	ON: карта пам'яті знаходиться в модулі CPU.
SM451	Перемикач захисту від запису на картці пам'яті ON: карта пам'яті захищена від запису. OFF: карта пам'яті не захищена від запису.
SM452	Дані записуються з ПЛК на карту пам'яті або дані зчитуються з карти пам'яті в ПЛК.
SM453	ON: під час роботи карти пам'яті виникає помилка. Якщо прапорець увімкнено, ви повинні скиньте його на OFF Код помилки зберігається в SR453.

3. Пов'язані коди помилок (SR453):

Код помилки	опис
16#005E	Під час ініціалізації карти пам'яті виникає помилка.
16#005F	Шлях неправильний, або файл не існує.
16#0060	Неможливо створити папку за замовчуванням.
16#0061	Пам'яті недостатньо.
16#0062	Карта пам'яті захищена від запису.
16#0063	Під час запису даних у файл виникає помилка.
16#0064	Неможливо прочитати дані з карти пам'яті.
16#0065	Файл призначений лише для читання.

приклад:

SM450 увімкнено, коли картку пам'яті вставлено в модуль ЦП; SM452 увімкнено, коли виконується інструкція MTWRIT;
SM452 вимкнено, коли інструкція MTWRIT завершується.



Операнд	Значення налаштування	опис
D0	16#0001	Файл, у який записуються дані Формат файлу: Одиницею символу є байт. Розширення імені файлу – .txt. Використовуйте коди ASCII. Дані в D300 записуються у файл.
D300	-	Дані, які записуються у файл
D10	16#000A	Розмір рядка, який записується у файл, становить 10 байт.
D1	16#0A00	Після того, як дані записані у файл, знак розділення додається після останнього байт у файлі.
D30	D30=16#6554 D31=16#7473 D32=16#0031	Ім'я файлу «Test1».

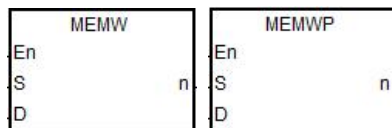
Додаткові зауваження

1. Якщо значення в **C** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
2. Якщо значення в **S₁** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.
3. Якщо значення в **S₃** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 — 16#2003.

API	Код інструкції			Операнд							функція					
2303		MEMW	P	S, D, n							Запис даних у файловий реєстр					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S	●	●			●	●		●								
D									●							
n	●	●			●	●		●				●	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●											
D		●											
n		●											

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ**S** : Перша адреса джерела даних**D** : Перша адреса, де зберігаються дані**n** : Довжина даних**Пояснення**

- Ця інструкція записує дані у файловий реєстр. Опис операндів виглядає наступним чином.
 - S** : початкова адреса джерела даних; пропонується оголосити змінну типу масиву.
 - D** : початкова адреса, де інструкція зберігає дані. Пропонується призначити адресу файлу зареєструвати та оголосити змінну типу масиву.
 - n** : довжина даних, які інструкція записує в реєстр файлу, між 1 і 2048. Якщо значення перевищує діапазон, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#200B.
 - Якщо пристрій **S** або **D** перевищує дозволений діапазон, SM0 увімкнено, а код помилки в SR0 — 16#2003.
- Оскільки для інструкції потрібно 60–120 мс для запису даних, пропонується використовувати цю інструкцію, коли ПЛК неактивний. Наприклад, коли немає зовнішнього завдання переривання, немає високошвидкісного виведення або будь-яких негайних подій для обробки ПЛК.
- Інструкція записується лише тоді, коли контакт перемикається з OFF на ON, і записується лише один раз. ПРИМІТКА. Ви можете записувати в цей файловий реєстр максимум 100 000 разів.

API	Код інструкції			Операнд							функція						
2304		MDEL	P	ctrl, fname							Видалення файлів на карті пам'яті						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
ctrl								●					○	○		
fname								●	●						○	

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
ctrl		●				●							
fname													●

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ

MDEL	MDELP
En	En
ctrl	ctrl
fname	fname

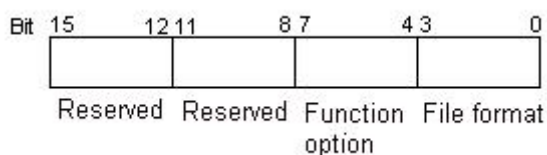
ctrl : Контрольний параметр

fname : Джерело даних

Пояснення

1. Ця інструкція записує дані з ПЛК на карту пам'яті. Операнди описані наступним чином.

- **ctrl** : параметр керування



Пункт	Код	опис
Формат файлу	0	Розширення імені файлу – .dmd.
	1, 2, 7	Розширення імені файлу – .cvs.
	3, 4, 5, 6	Розширення імені файлу – .txt.
Зарезервований	-	Значення бітів 8 – бітів 15 дорівнюють 0.

- Ви можете використовувати рядки (менше 9 символів) у **fname** (ім'я файлу).
- **fname** займає п'ять пристроїв. Ім'я файлу містить максимум дев'ять символів, включаючи 16#00. Якщо рядок не закінчується на 16#00, виникає помилка. Коли інструкція читає символ закінчення, вона припиняє читання та перевіряє, чи ім'я файлу є правильним. Символи в імені файлу можуть бути A–Z, a–z та 0–9. Розширення імені файлу залежить від формату файлу. Файл, створений інструкцією, знаходиться в папці за замовчуванням. Якщо ім'я файлу «Test1», інструкція записує символи в пристрої таким чином.

S3	'e'	"T"	ASCII Код ➔	S3	16#65	1 6#54
S3 + 1	't'	's'		S3 + 1	16 #74	16#73
S3+2	НУЛЬ	'1'		S3+ 2	16 №00	16#31

- Шлях до папки за замовчуванням:

Назва моделі	Шлях до папки
Серія AS300	PLC CARD\AS300\UserProg
Серія AS200	PLC CARD\AS200\UserProg

- Програмне забезпечення AS CPU версії 1.08.30 або новішої підтримує до 200 символів (включаючи 16#00) в імені файлу **S**₃.

2. Позначки інструкцій:

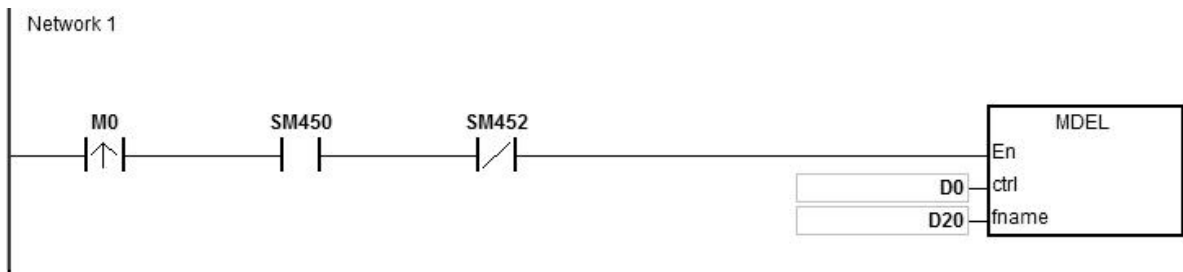
Прапор	опис
SM450	ON: карта пам'яті знаходиться в модулі CPU.
SM451	Стан перемикача захисту від запису на карті пам'яті ON: карта пам'яті захищена від запису. OFF: карта пам'яті не захищена від запису.
SM452	Дані записуються з ПЛК на карту пам'яті або дані зчитуються з карти пам'яті на ПЛК.
SM453	ON: під час роботи карти пам'яті виникає помилка. Якщо прапорець увімкнено, ви потрібно скинути його на OFF Код помилки зберігається в SR453.

3. Пов'язані коди помилок (SR453):

Код помилки	опис
16#005E	Під час ініціалізації карти пам'яті виникає помилка.
16#005F	Шлях неправильний, або файл не існує.
16#0060	Неможливо створити папку за замовчуванням.
16#0061	Пам'яті недостатньо.
16#0062	Карта пам'яті захищена від запису.
16#0063	Під час запису даних у файл виникає помилка.

приклад

SM450 увімкнено, коли ви вставляєте карту пам'яті в модуль ЦП; SM452 увімкнено, коли ця інструкція виконується; Після виконання цієї інструкції SM452 OFF.



Операнд	Значення налаштування	опис
D0	16#0001	Розширення імені файлу – .cvs.
D20	D20=16#6554 D21=16#7473 D22=16#0031	Ім'я файлу «Test1».

Додаткові зауваження

1. Якщо значення в **ctrl** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено та код помилки в SR0 — 16#2003.
2. Якщо значення в **fname** перевищує діапазон, виникає помилка операції, інструкція не виконується, SM0 увімкнено, а код помилки в SR0 – 16#2003.

API	Код інструкції				Операнд					функція						
2305		Record_Trigger			Execute ~ ErrCode					Запись и записывание данных в файл						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Execute	●	●	●	●	●	●	●	●		●	●					
SID	●	●			●	●	●	●	●		●		●			
Tb_num	●	●			●	●	●	●	●		●		●			
W_mode	●	●	●	●	●	●	●	●		●	●					
Done		●	●	●	●	●	●	●		●	●					
Busy		●	●	●	●	●	●	●		●	●					
Error		●	●	●	●	●	●	●		●	●					
ErrCode		●			●	●	●	●			●					

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Execute	●												
SID		●				●							
Tb_num		●				●							
W_mode	●												
Done	●												
Busy	●												
Error	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

Record_Trigger
En
Execute
Done
SID
Busy
Tb_num
Error
W_mode
ErrCode

Execute ~ W_mode : Зверніться до пояснення інструкцій нижче.

Done ~ ErrCode : Зверніться до пояснення інструкцій нижче.

Пояснення

- Ця інструкція доступна лише для ISPSOft V3.16 або новішої версії.
- Ця інструкція доступна для ПЛК AS300 із мікропрограмою версії 1.12 або новішої, якщо на ПЛК встановлено функціональну карту AS-FFTP01. Перед використанням перегляньте опис функції AS-FFTP01 у посібнику з модуля серії AS або приклад нижче, щоб завершити налаштування параметрів журналу для AS-FFTP01 у HWCONFIG.
- Перш ніж використовувати інструкцію, у HWCONFIG установіть для режиму журналу значення Program Control.

Налаштування відповідних параметрів описано нижче.

Режим журналу	Журнал циклу	Налаштування параметрів
Запуск ПЛК Завжди вмикати	Використовуйте цикл журналу в HWCONFIG	Використовуйте параметри в HWCONFIG
Контроль програми	«Execute» ініціюють користувачі	Цикли журналу, зокрема секунди, хвилини, години, дати та місяці, не підтримуються в наступних налаштуваннях «Час створення нового файлу», «Параметр створення нового файлу» та «Час збереження на SD-карті».

4. Імена та описи операндів наведено в наступній таблиці.

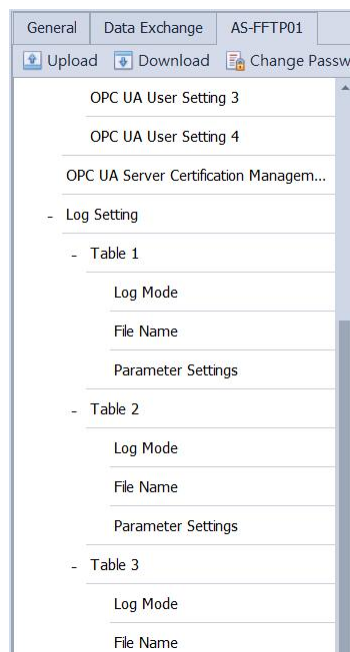
Операнд	Ім'я	опис	Тип даних	Зауваження
Execute	Виконати	Коли Execute змінюється на ON, починайте запис.	BOOL	
SID	ID слота	Фіксоване значення: 34 (Представляє функціональну картку AS-FFTP01)	WORD	
Tb_num	Номер таблиці	Вкажіть номер якої таблиці (від 1 до 4). HWCONFIG для запису.	WORD	
W_mode	Режим запису	Перш ніж Execute зміниться на ON, необхідно встановити режим. ● OFF: після того, як для параметра Execute встановлено значення ON, дані спочатку записуються в область буфера. Дані в області буфера будуть записані у файл лише після виконання умов, установлених у HWCONFIG. ● ON: після того, як для параметра Execute встановлено значення ON, дані спочатку записуються в область буфера. Дані в області буфера будуть безпосередньо записані у файл.	BOOL	
Done	Прапор завершення	Прапор завершення виконання ● OFF: не виконано/незавершено ● ON: Виконано успішно	BOOL	
Busy	Зайнятий прапор	Прапор зайнятості виконання ● OFF: не виконується ● ON: Виконується	BOOL	
Error	Прапор помилки	Прапор помилки ● OFF: немає помилок ● ON: сталася помилка	BOOL	
ErrCode	Код помилки	Зверніться до опису коду помилки нижче.	WORD	

5. Коды та описи помилок

Код помилки	опис
16#220Б	Значення параметра Tb_num виходить за межі діапазону.
16#0001	Кількість підкаталогів досягла верхнього ліміту (500), і більше неможливо створити підкаталог.
16#0002	SD-карта не встановлена. Перевірте та знову вставте SD-карту.
16#0003	Не вдалося записати дані у файл.
16#0004	Не вдалося записати дані.
16#0005	Неправильний режим журналу. Установіть для режиму журналу значення «Програмне керування» в HWCONFIG.
16#0006	Помилка налаштування параметра

приклад

- Використання програмного забезпечення FTP-клієнта, щоб перевірити, чи можна правильно прочитати SD-карту. Щоб дізнатися про налаштування FTP, зверніться до опису функції на AS-FFTP01 у розділі 10 Посібника з модуля серії AS.
- Переконайтеся, що налаштування часового поясу (HWCONFIG -> Ethernet Port Advanced Setting -> NTP -> Time Zone) і налаштування RTC (ISPSoft -> Tools-> Set RTC) для ПЛК AS300 встановлено правильно.
- Перейдіть до HWCONFIG -> AS-FFTP01 для налаштування параметрів



- (1) На сторінці Таблица 1 заповніть назву параметра, тип реєстру, адресу реєстру, тип даних тощо. Поля 1 і 2 закріплені для відображення дати та часу, а самовизначені поля починаються з ідентифікатора 3. У цьому прикладі дані, що зберігаються у двох регістрах даних, мають формати цілого числа та з плаваючою комою відповідно.

Table 1						
Add Move Up Move Down Delete Delete All Preview CSV File Enter text to search...						
Column Id	Column Name	Register Type	Register Address	Data Type	Decimal Places	
1	Date					
2	Time					
3	VFD Frequency	D	2103	INT16	0	
4	Temperature	D	0	Float	2	

(2) Встановіть режим журналу на програмне керування та ігноруйте параметр циклу журналу.

Log Mode						
	Name	Setting Value	Unit	Default	Minimum	Maximum
▶	Log Mode	Program Control		Program Control	-	-
	Log Cycle	100	0.1 sec	100	5	1200

(3) Встановіть назву батьківського каталогу, назву підкаталогу та назву файлу на сторінці імені файлу. Дані будуть збережені в: SD card/Delta/MyLog_date/Device_log_date_time.csv

File Name						
	Name	Setting Value	Unit	Default	Minimum	Maximum
▶	Parent directory Name (SD Card/)	Delta			-	-
	Subdirectory Name	MyLog		LOG1	-	-
	Automatically Appended Subdirectory Name	Date		Disable	-	-
	File Name (.csv)	Device_log		log1	-	-
	Automatically Appended File Name	Data + Time		Disable	-	-

(4) На сторінці «Параметри параметрів» налаштуйте наступні параметри « Час створення нового файлу », « Параметр створення нового файлу », « Час збереження на SD-карту » та « Параметр збереження на SD-карту ».

Parameter Settings						
	Name	Setting Value	Unit	Default	Minimum	Maximum
▶	New Directory Creating Timing	Automatically create a new dire...		Automatically create a ...	-	-
	New Directory Creating Parameter	1		1	1	31
	New File Creating Timing	Set maximum records of creat...		Overwrite existing file	-	-
	New File Creating Parameter	10	Count	100	1	10000
	Save to SD Card Timing	Count		Count	-	-
	Save to SD Card Parameter	5	Count	100	1	500

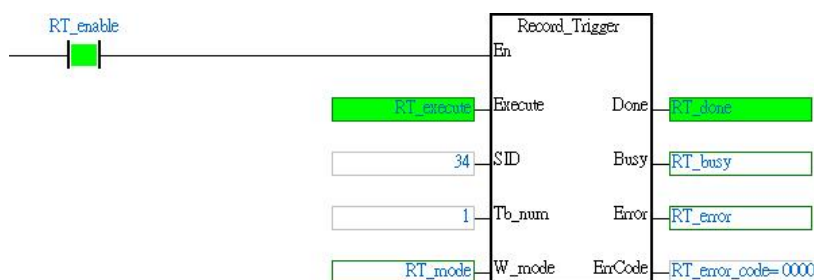
Кожен тригер записує один фрагмент даних. В одному файлі буде збережено не більше 10 записів. Запуск 5 разів означає, що 5 фрагментів даних буде записано в один файл. Інші 5 тригерів генерують ще 5 фрагментів даних, і дані будуть записані в той самий файл. Тобто після 15-го запису автоматично буде створено два файли.

4. Пояснення параметрів

Параметр	Значення налаштування	опис
SID	34	Фіксоване значення
Tb_num	1	Укажіть номер таблиці (від 1 до 4) із HWCONFIG для запису.
W_mode	OFF	

5. Тригер для виконання інструкції

Після того, як **Execute** встановлено на ON, дані записуються в буферну область, і якщо дані успішно записані в буферну область, **Done** стає ON. Перш ніж виконувати другий тригер, вам потрібно встановити для параметра «**Execute**» значення «OFF», а потім розпочати інше виконання. Щоразу, коли **Execute** змінюється з OFF на ON, починається новий запис.



6. W_mode:

(1) Якщо **W_mode** встановлено на ON, поточні дані в області буфера будуть записані безпосередньо у файл, коли **Execute** встановлено на ON, що можна використовувати для програм, у яких журнал не може бути втрачений для важливих подій або журнал повинен бути негайно збереженим у разі виникнення аномалії пристрою.

(2) Якщо для параметра **W_mode** встановлено значення OFF, поточні дані в області буфера будуть записані у файл лише після того, як умови, установлені в HWCONFIG, будуть виконані після зміни **Execute** на ON. Коли «**Execute**» ввімкнено втретє, у буферній області буде 3 записи даних, які ще не збережено у файлі. Після цього, якщо для **W_mode** встановлено значення ON перед виконанням 4-го тригера, 4-й запис і попередні 3 записи (а це всього 4 записи) будуть безпосередньо збережені у файлі на SD-карті після того, як **Execute** встановлено на ON для 4-й раз.

7. Після виконання 15-го тригера вручну в цьому прикладі створюється два файли, і всі записи зберігаються у форматі .csv. Перші 10 записів зберігаються в першому файлі, а останні 5 записів зберігаються в другому файлі. Файли журналу можна передавати на комп'ютер через FTP-клієнт.

Filename	Filesize	Filetype	Last modified	Permissions	Owner/G...
Device_log_20220425_184031_0000.csv	406	Microso...	4/25/2022 6:40:00 P...	-rwxrwxrwx	0 0
Device_log_20220425_183434_0000.csv	775	Microso...	4/25/2022 6:39:00 P...	-rwxrwxrwx	0 0

8. На наступному малюнку показано вміст першого відкритого файлу. Перший рядок показує заголовки. Виконайте дії, описані в кроці 3 вище, щоб налаштувати параметри на сторінці Таблиця 1. Дата й час у стовпцях A і B фіксовані. Дані в реєстрах записуються в рядки, починаючи з другого рядка.

	A	B	C	D	E
1	Date	Time	VFD frequency	Temperature	
2	2022/4/25	18:34:12	5723	20.97	
3	2022/4/25	18:34:16	3128	21	
4	2022/4/25	18:34:28	3631	20.97	
5	2022/4/25	18:34:30	5407	20.97	
6	2022/4/25	18:34:34	4754	20.97	
7	2022/4/25	18:39:24	2220	21.12	
8	2022/4/25	18:39:28	4114	21.16	
9	2022/4/25	18:39:31	5604	21.16	
10	2022/4/25	18:39:33	5061	21.16	
11	2022/4/25	18:39:42	372	21.09	
12					

6.24 Контрольні завдання Інструкції

6.24.1 Перелік інструкцій з контролю завдань

У наведеній нижче таблиці наведено інструкції щодо керування завданнями, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
<u>2400</u>	TKON	–	✓	Включення циклічного завдання
<u>2401</u>	TKOFF	–	✓	Відключення циклічного завдання

6.24.2 Пояснення Інструкції з контролю завдань

API	Код інструкції				Операнд								функція				
2400		TKON	P		S								Включення циклічного завдання				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●						●	●		O	O	O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
AS	AS	-

СИМВОЛ



S: Номер завдання

Пояснення

- 1. Ця інструкція вмикає циклічне завдання, визначене S .
- 2. Коли ПЛК працює, виконання циклічних завдань залежить від налаштування циклічних завдань у ISPSOft.
- 3. Операнд S має бути між 0-31.
- 4. Зверніться до посібника користувача ISPSOft для отримання додаткової інформації про створення та активацію завдань.

приклад

Коли ПЛК працює, увімкнено циклічне завдання (0). Коли виконується інструкція TKON у циклічному завданні (0), це вмикає циклічне завдання (1), а Y0.0 увімкнено.

Ви створюєте циклічні завдання в ISPSOft і визначаєте їх стандартний увімкнений/вимкнений стан у ISPSOft. Циклічне завдання (0) увімкнено під час роботи ПЛК, а циклічне завдання (1) не увімкнено під час роботи ПЛК.

Циклічне завдання (1) увімкнено інструкцією TKON у циклічному завданні (0).



Виконується циклічне завдання (1).



Додаткові зауваження

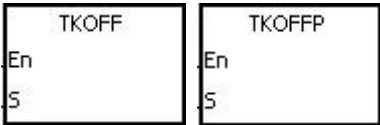
Додаткову інформацію про завдання див. у посібнику користувача ISPSOft.

API	Код інструкції				Операнд						функція						
2401		TKOFF	P		S						Відключення циклічного завдання						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S	●	●						●	●		O	O	O				

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
AS		AS		-	

СИМВОЛ



S : Номер завдання

Пояснення

1. Ця інструкція вимикає циклічне завдання, визначене S .
2. Коли ПЛК працює, виконання циклічних завдань залежить від налаштування циклічних завдань у ISPSoft.
3. Операнд S має бути між 0-31.
4. Зверніться до посібника користувача ISPSoft для отримання додаткової інформації про створення та активацію завдань.

приклад

Коли ПЛК працює, увімкнено циклічне завдання (0) і циклічне завдання (1). Коли виконується інструкція TKOFF у циклічній задачі (0), вона вимикає циклічну задачу (1), а Y0.0 вимкнено.

Ви створюєте циклічні завдання в ISPSoft і визначаєте їх стандартний увімкнений/вимкнений стан у ISPSoft. Циклічне завдання (0) і циклічне завдання (1) увімкнено під час роботи ПЛК, а циклічне завдання (1) вимкнено, коли інструкція TKOFF у циклічному завданні (0) виконується. Циклічне завдання (1) вимикається виконанням інструкції TKOFF у циклічному завданні (0).



Циклічне завдання (1) не виконується.



Додаткові зауваження

Додаткову інформацію про завдання див. у посібнику користувача ISPSOft.

6.25 Інструкції SFC

6.25.1 Список інструкцій SFC

У наведеній нижче таблиці наведено інструкції щодо SFC (схема послідовної функції), розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція
	16-бітний	32-розрядний		
<u>2500</u>	SFCRUN	–	–	SFC Run
<u>2501</u>	SFCPSE	–	–	SFC Пауза
<u>2502</u>	SFCSTP	–	–	Зупинка SFC

6.25.2 Пояснення Інструкції з контролю завдань

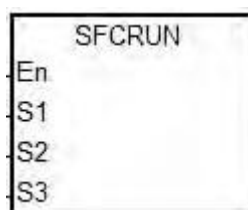
API	Код інструкції				Операнд								функція				
2500		SFCRUN			S_1, S_2, S_3								SFC Run				

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1																
S_2								●					O	O		
S_3																

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1													
S_2		●				●							
S_3													

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



S_1 : Назва SFC POU

S_2 : Код функції

S_3 : Адреса пристрою

Пояснення

- Ця інструкція запускає програму SFC (схема послідовних функцій), визначену S_1 відповідно до коду функції в S_2 .
- Інструкція запускає SFC POU (Програмна організаційна одиниця), визначену S_1 , лише коли ПЛК сканує SFC POU.
- Операнди описані наступним чином.
 - S_1 визначає назву SFC POU.
 - Коли виконується призначений SFC POU S_1 , інструкція очищає параметри (такі як SFC/STEP/ACTION/TRANSITION) для програми SFC, коли $S_2=0$ або 1, і виконання SFC починається відповідно до значення, зазначеного в S_2 .
 - ЯКЩО $S_2=0$, система виконує SFC POU з початкового кроку.
 - ЯКЩО $S_2=1$, система виконує SFC POU з визначеного кроку на S_3 .

- Якщо $S_2 = 2$, інструкція HE очищає статус і параметри (наприклад, SFC/КРОК/ДІЯ/ПЕРЕХІД) для SFC, і система починає виконання з того місця, де була призупинена.
- S_3 визначає, з чого почати (який крок) у програмі SFC на S_4 .

4. Діапазон S_2 знаходиться між 0–2. Якщо він виходить за межі діапазону, він оцінюється як 0.

5. Коли стан SFC POU є RUN, виконання цієї інструкції є недійсним.

приклад

Налаштуйте один LD (сходовий) POU та вкажіть його ім'я POU як «Main», а потім налаштуйте два SFC POU з іменами «TestSFC1» і «TestSFC2».

1. Коли ПЛК виконує основну програму (RUN), TestSFC1 і TestSFC2 виконують інструкцію SFCSTP (API 2502), а два SFC POU припиняють виконання.

2. Коли M0 перемикається з OFF на ON, TestSFC1/TestSFC2 POU виконує інструкцію SFCRUN*. Дивіться нижче ^{вміст} TestSFC1 і TestSFC2 для деталей виконання. Коли $S_2 = 0$, стан SFC і параметри очищаються і починають виконуватися з КРОКУ 1. Коли $S_2 = 1$, стан і параметри SFC очищаються і починають виконуватися з призначеного КРОКУ на S_3 .

3. Коли M1 перемикається з OFF на ON, TestSFC1 POU призупиняється. Коли $S_2 = 1$, усі дії та виходи, що виконуються SFC

очищаються, і система виконує остаточне сканування.

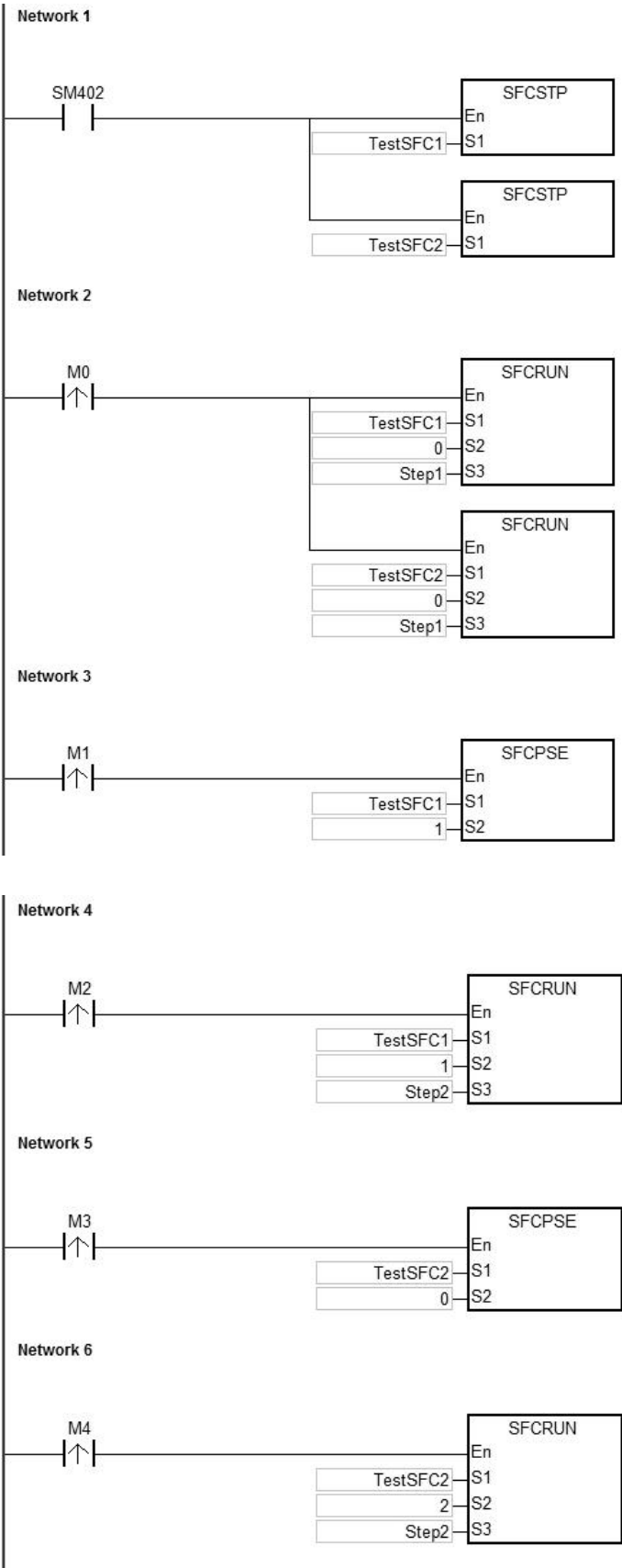
4. Коли M2 перемикається з OFF на ON, виконується TestSFC1 POU. Коли $S_2 = 1$, статус і параметри SFT є очищаються, і система починає виконуватися з КРОКУ 2.

5. Коли M3 перемикається з OFF на ON, TestSFC2 POU призупиняється. Коли $S_2 = 0$, усі дії та виходи, що виконуються SFC, зберігаються, і система не запускає остаточне сканування.

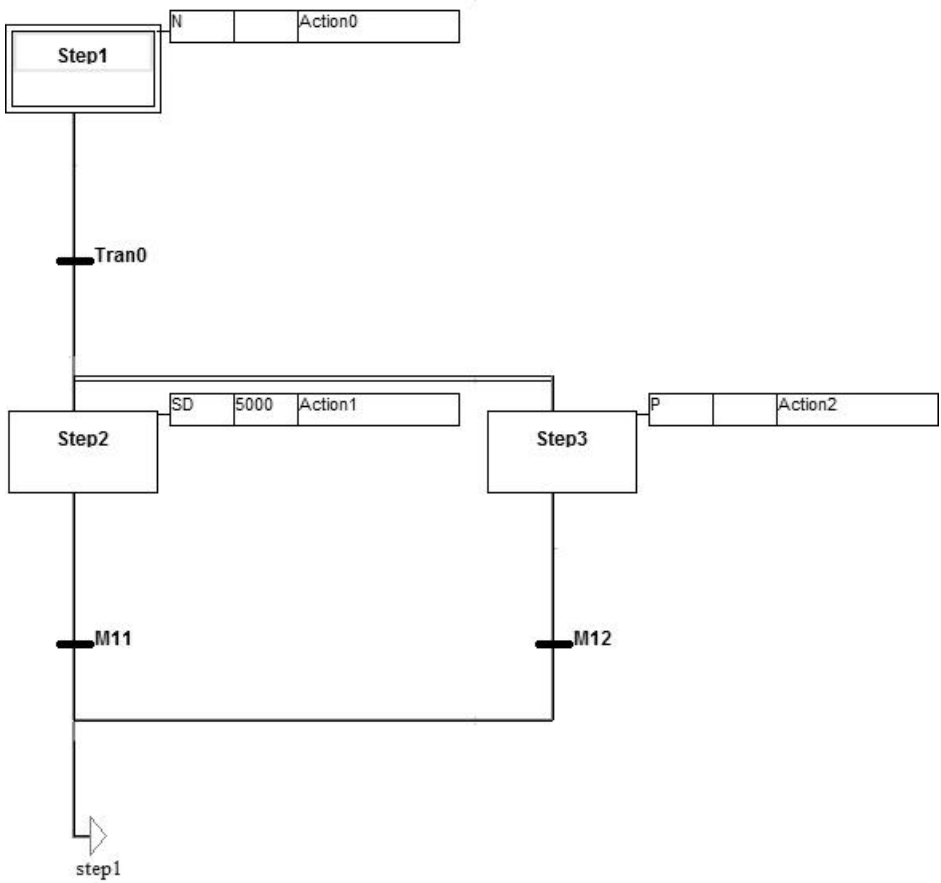
6. Коли M4 перемикається з OFF на ON, виконується TestSFC1 POU. Коли $S_2 = 2$, стан і параметри SFC є, і SFC починає виконуватися з того місця, де було призупинено.

*SFCRUN активує SPC POU під час наступного сканування.

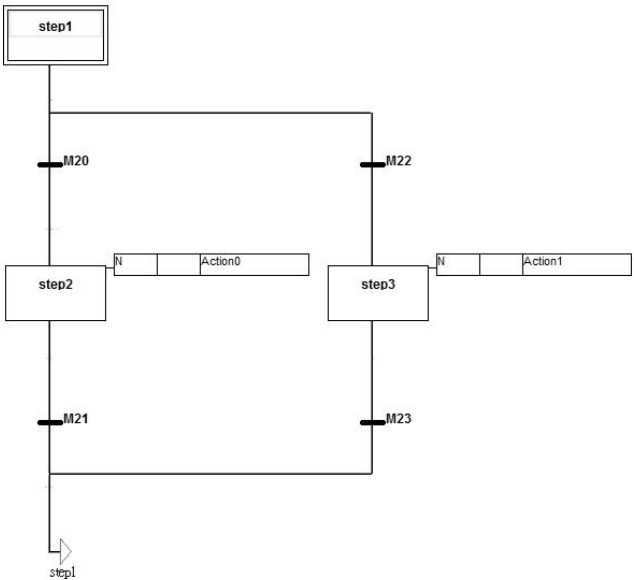
Головне ПОУ



TestSFC1 POU



TestSFC2 POU

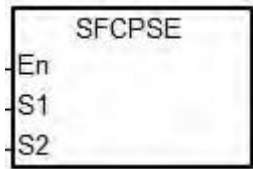


Додаткові зауваження

Зверніться до посібника користувача ISPSOft для отримання додаткової інформації про послідовні функціональні схеми.

API	Код інструкції				Операнд								функція				
2501		SFCPSE			S_1, S_2								SFC Пауза				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1																	
S_2								●					O	O			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1																	
S_2		●				●											
								Інструкція пульсу				16-бітна інструкція				32-розрядна інструкція	
								-				AS				-	

СИМВОЛ



S_1 : Назва SFC POU
 S_2 : Код функції

Пояснення

- Ця інструкція призупиняє SFC POU, визначений S_1 відповідно до коду функції на S_2 .
- Інструкція може призупинити SFC POU, визначений S_1 , лише коли ПЛК сканує SFC POU.
- Під час паузи статус SFC і параметри (такі як SFC/STEP/ACTION/TRANSITION) зберігаються.
- Операнди описані наступним чином.
 - S_1 визначає назву SFC POU.
 - Коли $S_2=0$, інструкція зберігає всі дії SFC і виходи, і система не запускає остаточне сканування.
 - Коли $S_2=1$, інструкція очищає всі дії, що виконуються, і виходи SFC POU, і система запускає остаточне сканування.
- Діапазон S_2 становить 0–1. Коли він виходить за межі діапазону, він обробляється як 0.
- Коли стан SFC POU є ПАУЗА/СТОП, виконання цієї інструкції є недійсним.

приклад

Для отримання додаткової інформації зверніться до прикладу програмування SFCRUN (API 2500).

Додаткові зауваження

Додаткову інформацію про SFC див. у посібнику користувача ISPSOft.

API	Код інструкції				Операнд								функція			
2502		SFCSTP			S								Зупинка SFC			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S																

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁													

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



S : Назва SFC POU

Пояснення

- Ця інструкція зупиняє SFC POU, визначений **S**.
- Ця інструкція зупиняє SFC POU, визначений **S₁**, лише коли ПЛК сканує SFC POU.
- Після зупинки інструкція очищає стан і параметри SFC, і система запускає остаточне сканування.
- Коли стан SFC POU є STOP, виконання цієї інструкції є недейсним.

приклад

Для отримання додаткової інформації зверніться до прикладу програмування SFCRUN (API 2500).

Додаткові зауваження

Додаткову інформацію про SFC див. у посібнику користувача ISPSOft.

6.26 Інструкції щодо високошвидкісного виведення

6.26.1 Список інструкцій щодо високошвидкісного виведення

У наведеній нижче таблиці наведено інструкції щодо високошвидкісного виведення, розглянуті в цьому розділі.

API	Інструкція код		Пульт інструктувати іон	функція	Парна вісь *1	Непарна вісь *2
	16- бітний	32- розрядний				
2700	–	DPLSY	–	Високошвидкісний імпульсний вихід (без процесу підвищення/зниження)	V	V
2701	–	DPLSR	–	Високошвидкісний імпульсний вихід (з процесом підвищення/зниження)	V	V
2702	PWM	DPWM	–	Широтно-імпульсна модуляція	V	V
2703	JOG	DJOG	–	Вихід JOG	V	V
2704	–	DZRN	–	Нульова віддача	V	–
2705	–	DPLSV	–	Регульований імпульсний вихід	V	V
2706	–	DDRVI	–	Контроль відносного положення	V	V
2707	–	DDRVA	–	Абсолютний контроль позиції	V	V
2708	CSFO	–	–	Швидкість улову та пропорційний вихід	V	–
2709	–	DDRVM	–	Розташування вирівнювання позначок	V	–
2710	–	DPPMR	–	2-осьовий синхронізований рух від точки до точки з відносною координатою	V	–
2711	–	DPPMA	–	2-осьовий синхронізований рух від точки до точки з абсолютною координатою	V	–
2712	–	DCICR	–	2-осьова дугова інтерполяція відносного положення за годинниковою стрілкою	V	–
2713	–	DCICA	–	2-осьова дугова інтерполяція за годинниковою стрілкою в абсолютному положенні	V	–
2714	–	DCICCR	–	2-осьова дугова інтерполяція відносного положення проти годинникової стрілки	V	–
2715	–	DCICCA	–	2-осьова дугова інтерполяція проти годинникової стрілки	V	–
2716	–	DCCMR	–	Креслення кола відносного положення	V	–
2717	–	DCCMA	–	Малювання кола в абсолютному положенні	V	–
2718	TPO	–	–	Таблиця планування позиції контролює вихід	V	–
2719	–	DTPWS	✓	Налаштування вихідних параметрів однієї осі в плануванні позиції стіл	V	–
2720	–	DTPWL	✓	Встановлення параметрів лінійної інтерполяції в плануванні позиції стіл	V	–

API	Інструкція код		Пульт інструктувати іон	функція	Парна вісь *1	Непарна вісь *2
	16- бітний	32- розрядний				
<u>2721</u>	–	DTPWC	✓	Налаштування параметрів дугової інтерполяції в плануванні позиції стіл	V	–
<u>2723</u>	–	DPPGB	–	Від точки до точки ходити вперед і назад	V	–
<u>2724</u>	–	DZRN2	–	Нульове повернення 2 (спрямований вихід можна визначити)	V	V

ПРИМІТКА:

*1: Осі з парними номерами включають Y0.0, Y0.2, Y0.4, Y0.6, Y0.8 і Y0.10.

*2: Осі з непарними номерами включають Y0.1, Y0.3, Y0.5, Y0.7, Y0.9 та Y0.11.

6.26.2 Пояснення інструкцій високошвидкісного виведення

API	Інструкція			Операнд	опис
2700	D	PLSY		S_1 , S_2 , D	Високошвидкісний імпульсний вихід (без процесу підвищення/зниження)

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		O		O	O		
S_2							●	●	●		O		O	O		
D		O														

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
D	●												

СИМВОЛ

	DPLSY	
En		
S1		D
S2		

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

- S : Вихідна частота імпульсів
 S_1
 S : Кількість імпульсів для виведення
 S_2
 D : Пристрій виведення імпульсів

Пояснення

- Ця інструкція вказує імпульсний вихід (S_1 , S_2) для пристрою, зазначеного в D , без наростання чи зменшення при зміні частоти. S_1 вказує діапазон вихідної імпульсної частоти: для моделей вихідного сигналу лінійного драйвера діапазон становить 0–4 МГц; для моделей виходу з відкритим колектором діапазон становить від 0 Гц до 200 кГц. Одиницею вихідної частоти є 1 Гц. Коефіцієнт помилок для 200 кГц становить приблизно 0,02%, а для 100 кГц – близько 0,01%. Рівень помилок зменшується зі зменшенням частоти. Наприклад, якщо вихідна частота становить 199990 Гц, фактична вихідна частота становить 199960 Гц. Якщо налаштування вихідної частоти становить 99999 Гц, фактична вихідна частота становить 99990 Гц. Якщо вихідна частота виходить за допустимий діапазон, ПЛК автоматично обробляє її як максимальну або мінімальну вихідну частоту імпульсу.
- Ви можете змінити вихідну частоту, визначену S_1 , під час виконання інструкції без використання процесу підвищення/зниження. Час для зміни частоти настає, коли ПЛК сканує інструкцію, і інструкція завершила виведення повного імпульсу.
- S_2 задано кількість вихідних імпульсів. Діапазон між 0–2 147 483 647. Коли ви встановлюєте кількість вихідних імпульсів на 0, тоді кількість імпульсів не обмежується, і імпульси виводяться постійно, доки інструкцію не буде вимкнено. Якщо кількість вихідних імпульсів менше 0, імпульси не виводяться.
- Після того, як інструкція починає виводити імпульси, кількість вихідних імпульсів, визначену S_2 , не може бути змінена.

5. D дозволяє лише Y0.0–Y0.11 як вихідні пристрої. Після виконання інструкції її функція виводу стає високошвидкісним виведенням, а керування точкою виведення базової інструкції стає недійсним. Рекомендується не використовувати загальну функцію виведення після використання функції високошвидкісного виведення в програмі.

6. Співвідношення часу вимкнення та часу увімкнення для імпульсного виходу становить 1:1.

7. Немає обмежень щодо того, скільки разів ви можете використовувати інструкцію програми. Але лише одна інструкція виведення, яка використовує ту саму точку виведення, може бути виконана в одному циклі сканування. Коли кілька інструкцій високошвидкісного імпульсного виведення починають ту саму точку виведення в програмі, ПЛК спочатку виконує виведення на основі інструкції, яка виконується першою.

8. Після виконання кожної інструкції високошвидкісного виведення для кожної точки виведення інша інструкція, що займає ту саму точку виведення, не може початися, доки інструкція високошвидкісного виведення у виконанні не буде вимкнена.

9. Коли інструкція високошвидкісного виведення виконується в програмі переривання або не знаходиться в основному процесі, пропонується використовувати інструкцію з функцією автоматичного скидання, коли виведення буде завершено, а ПЛК оновить вихідний стан у інструкції END .

10. Після встановлення прапора зупинки ПЛК припиняє вихід і знімає прапор зайнятості лише після другого виконання інструкції запуску та виведення повного імпульсу. ПЛК продовжує виводити імпульси, коли прапор зупинки скидається, а раніше зупинений вихід імпульсу не завершено. Ви встановлюєте та скидаєте прапор зупинки, встановлений у програмі.

11. Точки високошвидкісного виходу та відповідні SM/SR перераховані в наступній таблиці.

Номер точки виведення	Атрибут №2	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0,5
Зайнятий прапор	R	SM460	SM472	SM480	SM492	SM500	SM512
Прапор завершення №3	R/W	SM461	SM473	SM481	SM493	SM501	SM513
Прапор зупинки	R/W	SM463	SM474	SM483	SM494	SM503	SM514
Завершення виходу автоскидання #4	R/W	SM470	SM475	SM490	SM495	SM510	SM515
Поточний вихід позиція №1 (32-розрядний)	R/W	SR460 SR461	SR474 SR475	SR480 SR481	SR494 SR495	SR500 SR501	SR514 SR515
Номер точки виведення	Атрибут №2	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
Зайнятий прапор	R	SM520	SM532	SM540	SM552	SM560	SM572
Прапор завершення №3	R/W	SM521	SM533	SM541	SM553	SM561	SM573
Прапор зупинки	R/W	SM523	SM534	SM543	SM554	SM563	SM574

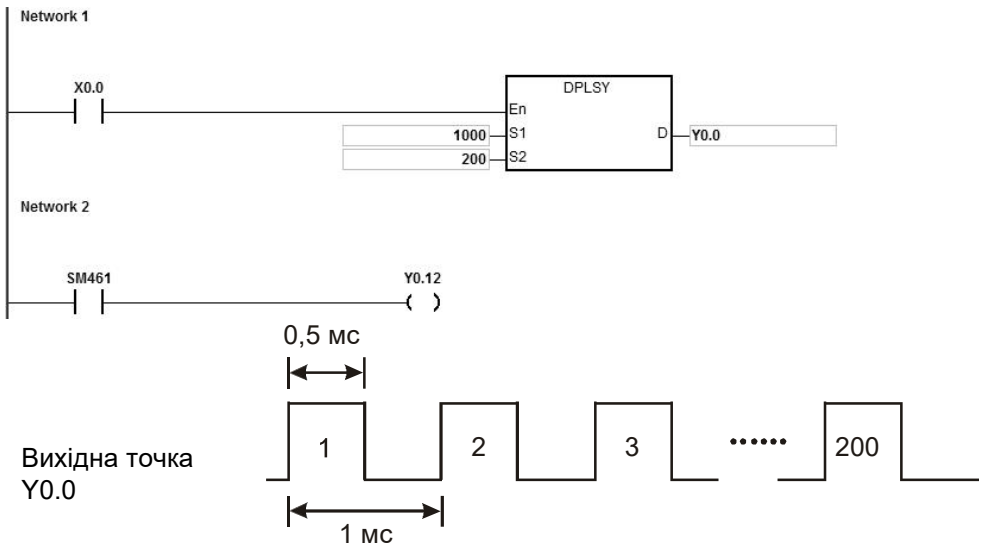
Номер точки виведення	Атрибут №2	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0,5
Завершення виходу автоскидання #4	R/W	SM530	SM535	SM550	SM555	SM570	SM575
Поточна вихідна позиція #1 (32-розрядний)	R/W	SR520 SR521	SR534 SR535	SR540 SR541	SR554 SR555	SR560 SR561	SR574 SR575

Примітки:

- #1: Усі наявні вихідні позиції фіксуються, коли живлення вимкнено.
- #2: R означає регістри лише для читання, і дані в регістрах не можна змінювати. R/W означає, що дані в регістрах можна читати та записувати.
- #3: Рекомендується очистити позначку завершення. Якщо прапор завершення не скинуто, він скидається автоматично, коли наступного разу виконується інструкція високошвидкісного виведення .
- #4: Ви встановлюєте позначку автоматичного скидання завершення виведення. ПЛК автоматично знімає прапор після завершення виведення.

Приклад 1

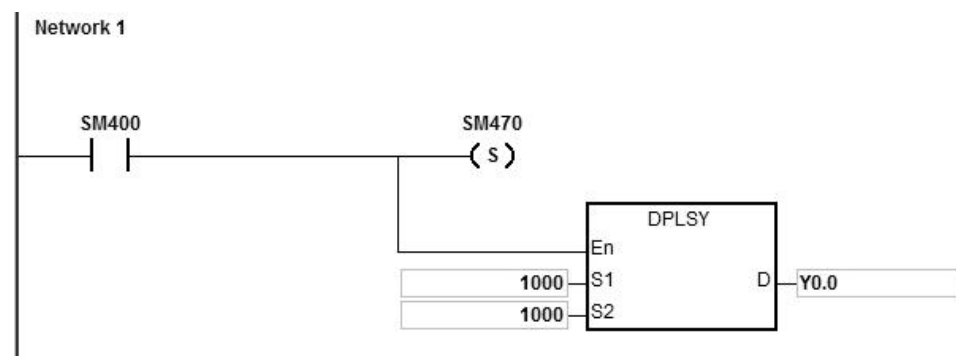
1. Коли X0.0 увімкнено, Y0.0 видає 200 імпульсів на 1 кГц. SM461 увімкнено, коли завершено вихід імпульсу, а потім Y0.12 увімкнено.
2. Коли X0.0 вимкнено, Y0.0 зупиняє вихід. Він перезапускає імпульсний вихід, коли X0.0 знову вмикається.



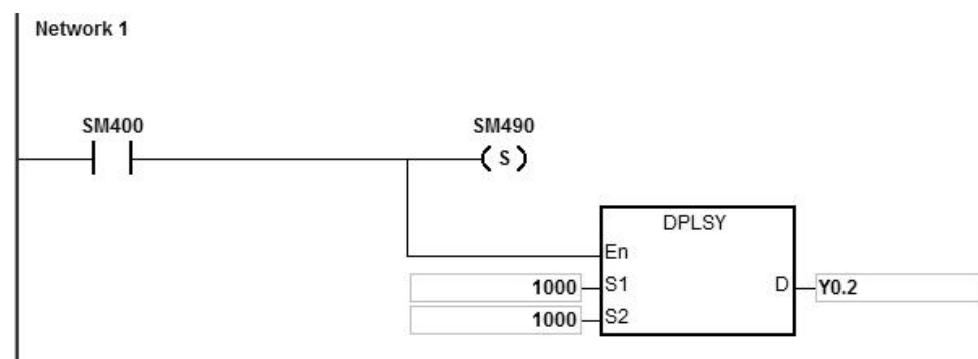
Приклад 2



Вхід зовнішнього переривання X0.0



Вхід зовнішнього переривання X0.1



Пояснення

1. Y0.0 видає 1000 імпульсів, коли X0.0 отримує один зовнішній сигнал переривання. Y0.2 видає 1000 імпульсів, коли X0.1 отримує один зовнішній сигнал переривання.
2. Коли вхід зовнішнього переривання X запускає вихід імпульсу з Y, інтервал часу між завершенням виведення імпульсу Y і наступним тригером входу X зовнішнього переривання має становити один або більше циклів сканування ПЛК.

Приклад 3 (програма ST)

```
0001 IF M0 THEN
0002     DPLSY(1000,1000,Y0.0);
0003     M1 := TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
```

Пояснення

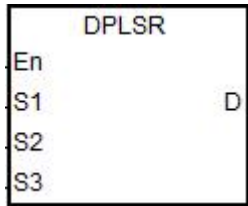
1. Коли M0 увімкнено, Y0.0 видає 1000 імпульсів на 1 кГц. Коли M1 увімкнено, це означає, що Y0.0 використовується для високошвидкісного імпульсного виведення.
2. Коли вихід імпульсу завершено, SM461 увімкнено. І якщо M1 також увімкнено, він запускає SM470 (автоматичне скидання завершення виведення для Y0.0) у положення ON, і Y0.0 стає вільним і готовим до використання.
3. Коли M0 перемикається з OFF на ON, імпульсний вихід перезапускається.
4. Якщо в програмі є більше двох інструкцій високошвидкісного виведення, не використовуйте назву змінної M1 повторно.
5. Рекомендується не використовувати мову ST у програмі, якщо є ймовірність зупинити виведення.

API	Інструкція			Операнд	опис
2701	D	PLSR		S_1 , S_2 , S_3 , D	Високошвидкісний імпульсний вихід (з процесом підвищення/зниження)

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
D		○														

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
D	●												

СИМВОЛ	Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
	—	—	AS

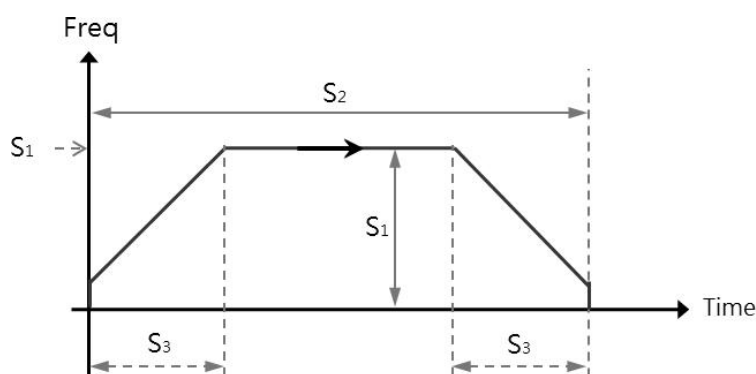


- S_1 : Цільова вихідна частота
- S_2 : Кількість вихідних імпульсів. Час
- S_3 : наростання/вимкнення. Імпульсний
- D : вихідний пристрій

Пояснення

- Ця інструкція визначає імпульсний вихід (S_1 , S_2) для пристрою, зазначеного в D, із збільшенням або зменшенням у змінна частоти. S_1 вказує вихідну частоту імпульсу як 4 МГц для моделей виходу лінійного драйвера та вихідну частоту імпульсу між 0 Гц–200 кГц для моделей виходу з відкритим колектором. Одиницею вихідної частоти є 1 Гц. Допустима частота помилок для 200 кГц становить приблизно 0,02%, а для 100 кГц – приблизно 0,01%. Рівень помилок зменшується зі зменшенням частоти. Наприклад, якщо вихідна частота становить 199990 Гц, фактична вихідна частота становить 199960 Гц. Якщо налаштування вихідної частоти становить 99999 Гц, фактична вихідна частота становить 99990 Гц. Якщо вихідна частота виходить за допустимий діапазон, ПЛК автоматично обробляє її як максимальну або мінімальну вихідну частоту імпульсу.
- Після виконання інструкції ви можете змінити цільову вихідну частоту, визначену S_1 , і ПЛК виконує процес нарощування або зниження відповідно до налаштування часу наростання або зниження S_3 . Час для зміни частоти настає, коли ПЛК сканує інструкцію, і інструкція завершила виведення повного імпульсу.
- S_2 – кількість вихідних імпульсів. Діапазон між 0–2 147 483 647. Коли ви встановлюєте кількість імпульсів на 0, тоді кількість імпульсів не обмежується, і імпульси виводяться безперервно, доки інструкція не буде вимкнена.

4. Після того як інструкція починає виводити імпульси, кількість вихідних імпульсів, яку вказує S_2 , не може бути змінена.
5. S_3 встановлює час розгону або затиску з одиницею 1 мс. Значення діє, коли інструкція виконується вперше. Якщо цільова частота, визначена S_1 , змінена в процесі наростання, час наростання або спаду перезавантажується для виконання. Але якщо цільова частота змінюється, коли вихідний сигнал переходить у процес зниження, інструкція ігнорує цю зміну.
6. D допускає лише Y0.0–Y0.11 як вихідні імпульсні пристрої. Після виконання інструкції її функція виведення стає високошвидкісним виведенням, а контроль точки виведення загальної інструкції стає недійсним. Рекомендується не використовувати загальну функцію виведення після використання функції високошвидкісного виведення в програмі.
7. Цільова вихідна частота S_1 , кількість вихідних імпульсів S_2 і час наростання/зниження S_3 проілюстровані на наступному графіку.

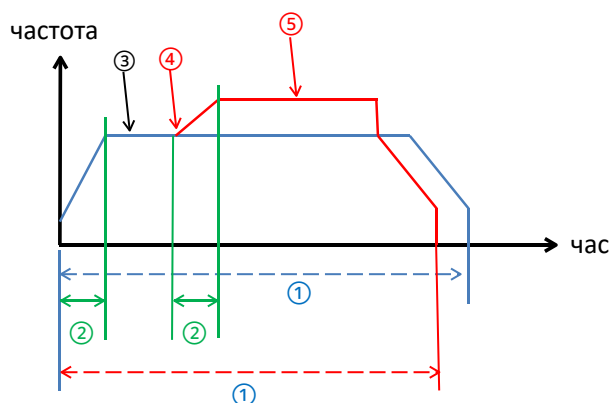


8. Співвідношення часу вимкнення та часу увімкнення для імпульсного виходу становить 1:1.
9. Немає обмежень щодо того, скільки разів ви можете використовувати інструкцію програми. Але лише одна інструкція виведення, яка використовує та сама вихідна точка може бути виконана в одному циклі сканування. Коли кілька інструкцій високошвидкісного імпульсного виведення починають ту саму точку виведення в програмі, ПЛК спочатку виконує виведення на основі інструкції, яка виконується першою.
10. Після ввімкнення інструкції високошвидкісного виведення для однієї точки виведення інші інструкції, які займають ту саму точку виведення, не можуть запуститися, доки не буде вимкнено інструкцію високошвидкісного виведення під час виконання.
11. Коли інструкція високошвидкісного виведення виконується в програмі переривання або відсутня в основній програмі, пропонується використовувати інструкцію з функцією автоматичного скидання, коли виведення буде завершено, а ПЛК оновить вихідний стан у END інструкція.
12. Після того, як прапор зупинки встановлено, ПЛК виконує зупинку з уповільненням; або він негайно припиняє вихід тільки після того, як інструкція запуску буде виконана вдруге, і введе повний імпульс.
13. Зверніться до інструкції PLSY (API 2700) для пояснення високошвидкісних вихідних точок і відповідних SM/SR.
14. Зверніться до прикладу 3 з інструкції PLSY (API 2700) для програмування мовою ST.

15. Коли цільова вихідна частота досягнута, ви можете змінити встановлену цільову вихідну частоту. Частоти для підвищення та зменшення частоти вже встановлені, якщо ви змінюєте встановлену цільову вихідну частоту під час виконання інструкції, частота буде або збільшуватися, або зменшуватися. Якщо встановлено більше однієї цільової вихідної частоти, пропонується встановити останню нову цільову частоту такою ж, як початкова цільова частота, щоб уникнути різких змін, коли потрібно зупинитися.

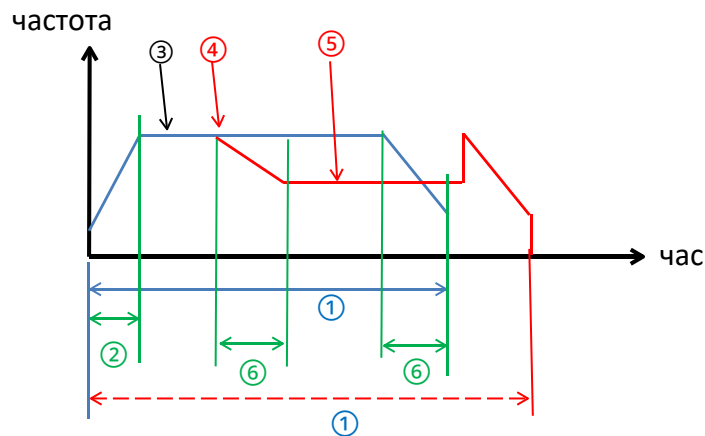
Дивіться приклад нижче. Синя лінія вказує на початкову цільову частоту, а червона лінія вказує на нову цільову частоту.

● Ситуація А: нарощування



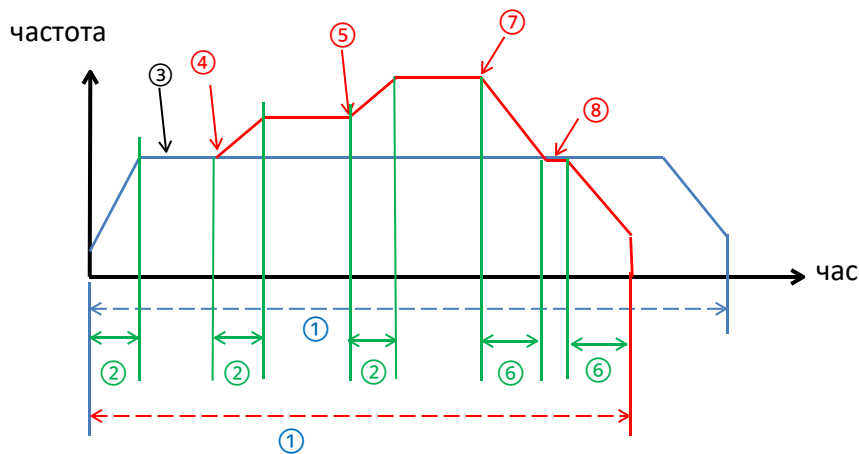
- ① Кількість вихідних імпульсів
- ② Нарощувати час
- ③ Перша цільова частота
- ④ Позиція для зміни на нову цільову частоту
- ⑤ Нова цільова частота

● Ситуація Б: зниження



- ① Кількість вихідних імпульсів
- ② Нарощувати час
- ③ Перша цільова частота
- ④ Позиція для зміни на нову цільову частоту
- ⑤ Нова цільова частота
- ⑥ Час зниження

- Пропонований дизайн: пропонується встановити останню нову цільову частоту такою ж, як початкова цільова частота щоб уникнути різких змін, коли потрібно зупинитися.



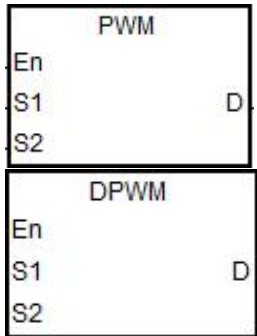
- ① Кількість вихідних імпульсів
- ② Нарощувати час
- ③ Перша цільова частота
- ④ Позиція для зміни першої нової цільової частоти
- ⑤ Позиція для зміни другої нової цільової частоти
- ⑥ Час зниження
- ⑦ Положення для зміни третьої нової цільової частоти
- ⑧ Позиція для зміни вихідної заданої цільової частоти, а потім почніть зменшуватись до зупинки.

API	Інструкція			Операнд							опис						
2702	D	PWM		S_1, S_2, D							Широтно-імпульсна модуляція						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1	●	●			●	●	●	●	●		○		○	○			
S_2	●	●			●	●	●	●	●		○		○	○			
D		○															

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
D	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	AS

СИМВОЛ



S_1 : ширина вихідного імпульсу

S_2 : Цикл вихідного імпульсу

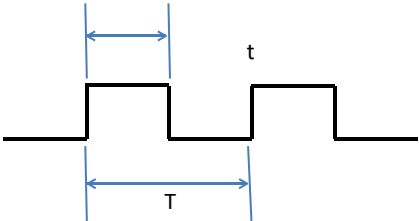
D : пристрій виведення імпульсів

Пояснення

1. Ця інструкція вказує ширину вихідного імпульсу. 16-розрядна інструкція використовує 100 мкс як вихідну одиницю, тоді як 32-бітна інструкція використовує 1 мкс як вихідну одиницю.

Вихідна інструкція	PWM	DPWM
Діапазон ширини вихідного імпульсу S_1	0–30000	0–60000
Діапазон імпульсного вихідного циклу S_2	0–30000	0–60000

2. S_1 (широта вихідного імпульсу — режим увімкнено) визначається як t , S_2 (час циклу вихідного імпульсу — час циклу) — це T , як показано нижче. Рекомендується $S_1 \leq S_2$.



3. **D** дозволяє лише Y0.0–Y0.11 як вихідний імпульсний пристрій.
4. Немає обмежень щодо того, скільки разів ви можете використовувати інструкцію програми. Але лише одна інструкція виведення, яка використовує ту саму точку виведення, може бути виконана в одному циклі сканування. Коли кілька інструкцій високошвидкісного імпульсного виведення починають ту саму точку виведення в програмі, ПЛК спочатку виконує виведення на основі інструкції, яка виконується першою.
5. Якщо $S_1 \leq 0$ або $S_2 \leq 0$, він обробляється як 0 (імпульсний вихід OFF). Якщо $S_1 > S_2$, він обробляється як $S_1 = S_2$. Коли $S_1 = S_2$ і S_2 не дорівнює 0, імпульсний вихід завжди ввімкнено.
6. Ви можете змінити ширину вихідного імпульсу S_1 і цикл вихідного імпульсу S_2 під час виконання інструкції PWM.
7. Відповідні спеціальні регістри SR перераховані в наступній таблиці.

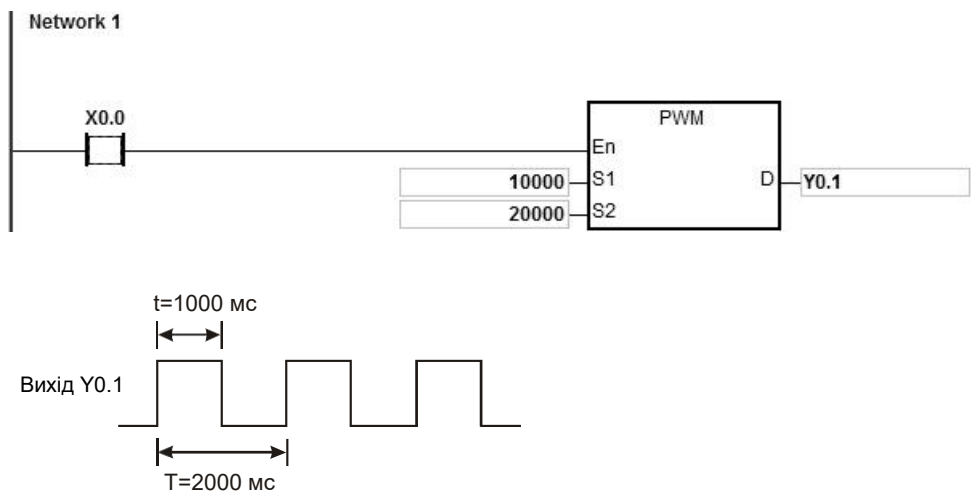
Номер точки виведення	Атрибут	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0,5
Поточний вихід позиція №1 (32-розрядний)	R/W	SR460	SR474	SR480	SR494	SR500	SR514
		SR461	SR475	SR481	SR495	SR501	SR515
Номер точки виведення	Атрибут	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
Поточна вихідна позиція #1 (32-розрядний)	R/W	SR520	SR534	SR540	SR554	SR560	SR574
		SR521	SR535	SR541	SR555	SR561	SR575

#1 : усі поточні вихідні позиції фіксуються, коли живлення вимкнено.

8. Зверніться до прикладу 3 з інструкції DJOG (API 2703) для програмування на мові ST.

приклад

Коли X0.0 увімкнено, Y0.1 видає наступні імпульси. Коли X0.0 вимикається, вихід Y0.1 змінюється на OFF



API		Інструкція			Операнд							опис							
2703		D	JOG				S_1, S_2, S_3, D_1, D_2							Вихід JOG					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1					●	●	●	●	●		○		○	○		
S_2					●	●	●	●	●		○		○	○		
S_3					●	●	●	●	●		○		○	○		
D_1		○														
D_2		○	○													

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●	●			●	●						
S_2		●	●			●	●						
S_3		●	●			●	●						
D_1	●												
D_2	●												

Інструкція по пульсу		16-бітна інструкція	32-розрядна інструкція
—		AS	AS

СИМВОЛ



- S_1 : Час розгону
- S_2 : Цільова вихідна частота
- S_3 : Час зниження
- D_1 : Пристрій виведення імпульсів
- D_2 : Допоміжний вихідний пристрій

Пояснення

1. Ця інструкція визначає вихід JOG (S_1, S_2, S_3) для пристроїв, зазначених у D_1 і D_2 . S_1 — це час розгону з одиницею 10 мс. Наприклад, значення 10 означає, що час розгону становить 100 мс. Після виконання інструкції вихідну частоту можна поділити на десять, і вона прискорюється кожні 10 мс. Він досягає цільової частоти JOG, визначеної S_2 , у час розгону S_1 .
2. Коли інструкцію вимкнено, її вихідна частота зменшується один раз кожні 10 мс на основі часу спаду в S_3 . Вихід не припиняється, доки не досягнеться час зниження. Якщо встановити час зниження на 0, вихід негайно припиняється.

3. Діапазон часу розгону S_1 і часу спаду S_3 становить 0–3000 (0–30 секунд). Якщо значення налаштування виходить за межі діапазону, ПЛК обробляє його як мінімальне або максимальне вихідне значення. На час розгону та час спаду впливає час сканування. Якщо вам потрібен точний час наростання або зниження, рекомендується використовувати інструкцію виводу з указаним часом наростання або зниження, наприклад інструкцію DDRVI (API 2706).
4. Діапазон цільової вихідної частоти S_2 становить від -200 кГц до 200 кГц. Якщо значення налаштування виходить за межі діапазону, ПЛК обробляє його як мінімальне або максимальне вихідне значення. Додатне число для вихідної частоти вказує на прямий вихід. Від'ємне число для вихідної частоти вказує на зворотний вихід.
5. D_1 дозволяє лише Y0.0–Y0.11 для вихідної точки. Для допоміжної вихідної точки, визначеної D_2 , зверніться до вихідних точок у наступній таблиці. Якщо ви виберете інші вихідні точки або пристрій M, значення в SR, що вказує на вихідний режим, буде недійсним, а режим Pulse+direction буде дійсним за замовчуванням. D_2 – вихідна точка напрямку.

D_1 вибирає парний номер вихідної точки.

Вихідна точка для D_1	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
Точка виведення напрямку для D_2	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Зайнятий прапор	SM460	SM480	SM500	SM520	SM540	SM560
Режим виведення	SR462	SR482	SR502	SR522	SR542	SR562
Поточна вихідна позиція	SR460	SR480	SR500	SR520	SR540	SR560
	SR461	SR481	SR501	SR521	SR541	SR561
Люфт компенсація	SR478	SR498	SR518	SR538	SR558	SR578

D_1 вибирає непарний номер вихідної точки.

Вихідна точка для D_1	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Точка виведення напрямку для D_2	Y1.0 або вище або будь-який пристрій M (BOOL)					
Зайнятий прапор	SM472	SM492	SM512	SM532	SM552	SM572
Поточна вихідна позиція	SR474	SR494	SR514	SR534	SR554	SR574
	SR475	SR495	SR515	SR535	SR555	SR575
Люфт компенсація	SR479	SR499	SR519	SR539	SR559	SR579

6. Коли значення дорівнює 0 у SR462, SR482, SR502, SR522, SR542 і SR562, це вказує на вихідний режим Pulse+direction. Коли значення дорівнює 1 у SR462, SR482, SR502, SR522, SR542 і SR562, це вказує на вихідний режим фази A/B.

Примітка: ви можете вибрати режим виведення лише тоді, коли D_1 використовує рівну точку виведення, а D_2 використовує точку виведення рекомендованого напрямку.

7. Якщо виходи напрямків не використовують виходи за замовчуванням для виведення, ви можете звернутися до таблиці нижче, щоб налаштувати час виведення, коли імпульсний вивід йде першим перед виведенням напрямку. Щоб ви могли бути впевнені, що виведення відбувається лише під час перемикавання на виходи напрямку. Одиницею для SR є мілісекунда, а діапазон – 0-20 мс. За замовчуванням 0, що вказує на неактивність.

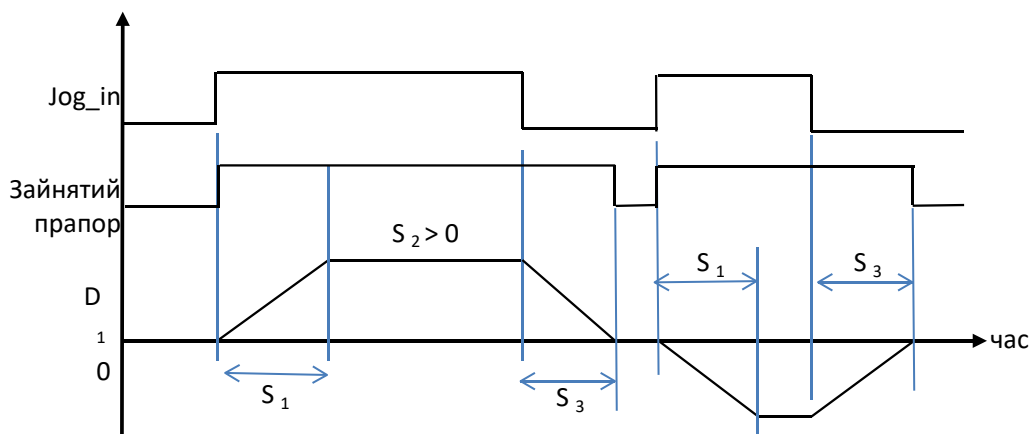
Вихідна точка для D_1	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5
Номер SR	SR640	SR641	SR642	SR643	SR644	SR645
Вихідна точка для D_1	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
Номер SR	SR646	SR647	SR648	SR649	SR650	SR651

Примітка: джерелом часу тут є час сканування ПЛК. (доступно для мікропрограми версії 1.02.20 або новішої)

8. Коли починається вихід, прапор зайнятості SM увімкнено. Після завершення виводу прапор «Зайнято» автоматично скидається на OFF, а прапор «Завершення» не вмикається.

9. На наступному графіку показано часову діаграму виведення. Позначку «Зайнятий» на наступному графіку дивіться на «Зайнятий».

вісь прапора.

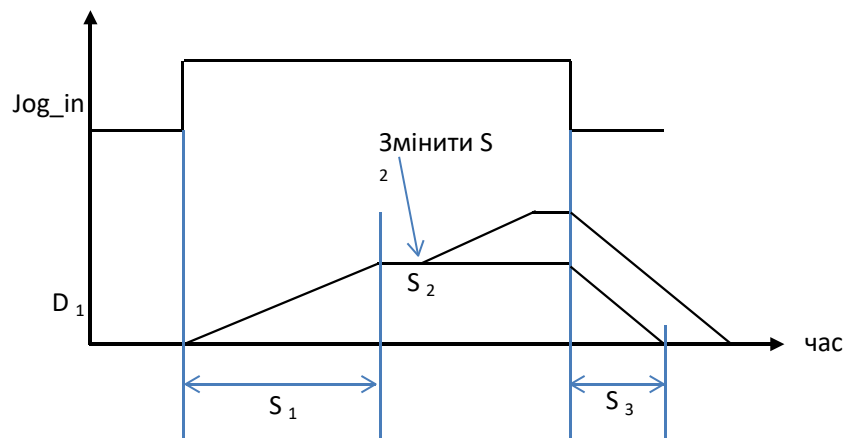


$S_2 < 0$

10. Ви можете змінювати цільову вихідну частоту під час виконання інструкції, але НЕ можете змінювати час наростання або спаду. Коли нова цільова частота більша за попередню, інструкція використовує нахил наростання. Коли нова частота менша за попередню, інструкція використовує нахил спаду.

11. Інструкція встановлює нахил підвищення/зниження шляхом перетворення встановленого часу та цільової частоти під час виконання інструкції. Нахил не змінюється зі зміною цільової частоти в процесі виведення. Наприклад, вихідна цільова вихідна частота становить 1 кГц, а потім вона змінюється до 2 кГц. Фактичний час зниження

відрізняється від початкового часу. Пунктирна лінія – це діаграма часу наростання та зменшення частоти після зміни цільової частоти.



12. Якщо відбувається будь-яка зміна частоти під час підвищення або зниження частоти, зміна може бути здійснена лише після

вихідна частота завершує вихід. Наприклад, якщо нова цільова частота становить 100 Гц, а час наростання встановлено на 100 (1 секунда), після завершення виведення вихідної частоти наступною буде виведена нова цільова частота 100 Гц. (Незважаючи на те, що цільова частота змінюється кожні 10 мс, але якщо імпульс НЕ завершує виведення, зміну НЕ можна застосувати.)

Приклад (програма ST):

У цій інструкції як мову програмування пропонується використовувати сходові діаграми. Якщо потрібно використовувати структуровані

текстів, вам також потрібно використовувати прапор автоматичного скидання завершення виведення, щоб ПЛК знав, що точка виводу вільна та готова до використання знову, коли виведення завершиться. Взявши Y0.0 як вихідну точку, програму слід написати, як показано нижче. Дивіться наступний приклад.

```

0001 IF M0 THEN
0002     DJOG(1000,5000,100,Y0.0,Y0.1);
0003     M1 := TRUE;
0004 ELSIF (NOT M0) AND M1 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END IF;

```

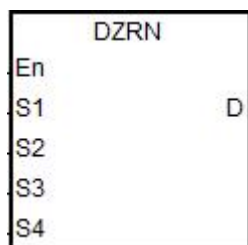
API	Інструкція			Операнд	опис
2704	D	3PH		S ₁ , S ₂ , S ₃ , S ₄ , D	Нульова віддача

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
S_4	○															
D		○														

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4	●												
D	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



- S_1 : Цільова частота для нульової віддачі
- S_2 : Частота JOG для DOG
- S_3 : Режим нульового повернення
- S_4 : Пристрій введення для DOG
- D : Пристрій виведення імпульсів

Пояснення

- Ця інструкція повертає машину до нульової точки. Діапазон цільової частоти для нульового повернення S_1 становить від 1 Гц до 200 кГц. Частота JOG S_2 має бути меншою за цільову частоту S_1 . Частота JOG S_2 – початкова частота. Якщо S_1 менше S_2 , S_1 автоматично переглядається і обробляється як рівний S_2 .
- Вхідна точка для S_4 і вихідна точка для D повинні збігатися. Не змінюйте їх під час виконання інструкції. Вхідною точкою для S_4 пропонується використовувати 16 високошвидкісних точок X0.0–X0.15. PLC не вплине на них час сканування інструкції. Якщо ви використовуєте послідовні точки введення X1.0 або пристрої M (потрібна мікропрограма версії 1.02.10 або вище), на них впливатиме час сканування інструкцій ПЛК. Зверніться до наступної таблиці для вибору D вихідна точка і вихідна точка напрямку. Якщо D не є попередньо встановленим вихідним сигналом Pulse+direction (за замовчуванням: 0), змініть режим на вихід фази A/B, встановивши SR на 1.

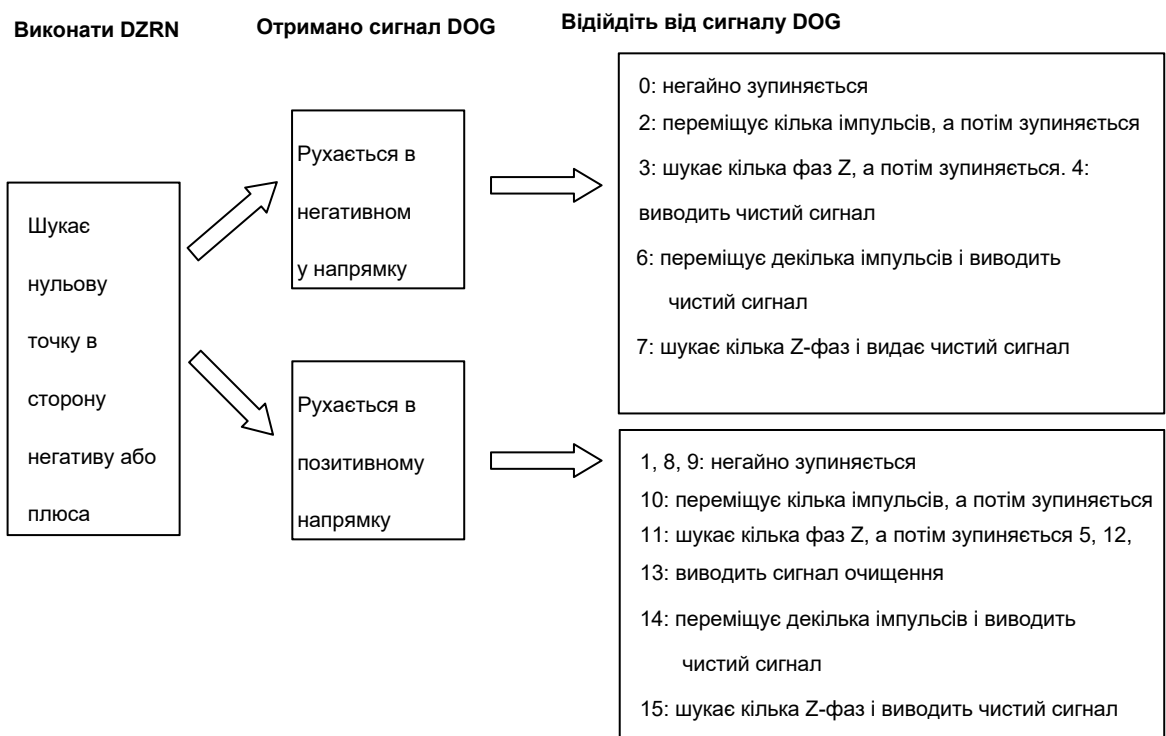
Номер осі	Вісь 1	Вісь 2	Вісь 3	Вісь 4	Вісь 5	Вісь 6
Вхідна точка для S_4	Може бути будь-якою з вхідних точок X0.0–X0.15. Але ту саму вхідну точку не можна вибрати для виведення різних осей. Якщо точка DOG тремтить або перемикач підстрибує, встановіть час фільтра точки введення в HWCONFIG.					
Вихідна точка для D	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
Точка виведення напрямку	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Режим виведення	SR462	SR482	SR502	SR522	SR542	SR562
Зайнятий прапор	SM460	SM480	SM500	SM520	SM540	SM560
Прапор завершення	SM461	SM481	SM501	SM521	SM541	SM561
Поточна вихідна позиція	SR460 SR461	SR480 SR481	SR500 SR501	SR520 SR521	SR540 SR541	SR560 SR561

3. За допомогою S_3 виберіть режим повернення до нуля. Код функції встановлюється двома старшими та нижчими 16-бітними параметрами. Подробиці див. у наступній таблиці.

S_3 : виберіть режим повернення до нуля				
Високий 16-біт	Низький 16-біт			
b31~b16	b15~b6	b5	b4	b3~b0
Кількість імпульсів для руху	зарезервований	Налаштування напрямку* 1	Налаштування сигналу DOG	Налаштування режиму 0~15 (F)
Кількість фаз Z		0: у негативному напрямку 1: у позитивному напрямку	0: контакт А 1: контакт Б	

Примітка 1. Ця функція доступна лише для мікропрограми версії 1.04.30 або новішої.

Налаштування режиму дивіться на діаграмі нижче.



4. За допомогою **S₃** виберіть режим повернення до нуля. Код функції встановлюється двома старшими та нижчими 16-бітними параметрами. Подробиці див. у наступній таблиці.

Функції	Код		Пояснення
	Високий 16-біт	Низький 16-біт	
Покидає нульову точку в негативному напрямку, а потім зупиняється (Режим 0)	0	0	Коли інструкція виконується, пошук нульової точки відбувається в негативному напрямку з цільовою частотою. Коли нульова точка ввімкнена (сигнал нульової точки змінюється з OFF на ON), частота зменшується до швидкості JOG, а рух у негативному напрямку продовжується та не зупиняється, доки сигнал нульової точки не зміниться з ON на OFF
Покидає нульову точку в позитивному напрямку, а потім зупиняється (Режим 1)	0	1	Коли інструкція виконується, пошук нульової точки відбувається в негативному напрямку з цільовою частотою. Коли нульова точка ввімкнена (сигнал нульової точки змінюється з OFF на ON), частота негайно зменшується до 0, а потім рух відбувається в позитивному напрямку зі швидкістю JOG і не зупиняється, доки сигнал нульової точки не зміниться з ON до OFF.
Режим 0 Переміщується знову після повернення до нульової точки	Кількість імпульсів для руху	2	Повернення до нульової точки відбувається так само, як і для молодшого 16-бітного коду. Після ввімкнення нульової точки рух продовжується відповідно до кількості заданих імпульсів. Коли старший 16-бітний код є додатним числом, пошук здійснюється в додатному напрямку. Від'ємне значення означає, що пошук здійснюється в негативному напрямку.
Режим 0 Пошук фази Z після повернення до нульової точки (Точка введення фази Z встановлюється в HWCONFIG)	Кількість фаз Z	3	Повернення до нульової точки відбувається так само, як і для молодшого 16-бітного коду. Після повернення в нульову точку рух продовжується за кількістю Z фаз. Коли старший 16-бітний код є позитивним числом, пошук відбувається в позитивному напрямку. Від'ємне значення означає, що пошук здійснюється в негативному напрямку. Припустімо, ви вказали тригер X0.1 по наростаючому фронту як умову для введення фази Z у HWCONFIG. Підрахунок виконується один раз, коли виникає тригер по наростаючому фронту для X0.1.
Режим 0 Виводить чистий сигнал після повернення до нульової точки. (Точка очищення виводу встановлюється в HWCONFIG)	Кількість імпульсів для руху або кількість Z-фаз	4+0=4 4+1=5 4+2=6 4+3=7 (біт 2=ON)	Вибір значення між 4–7 означає вибір кодів функцій 0–3 відповідно, а зазначена вихідна точка надсилає сигнал ON шириною близько 20 мс, коли виконання функції завершується. Діапазон вихідної точки становить Y0,12–Y0,15 та Y1,0–Y1,15. Наприклад, якщо ви вказуєте Y1.12 як вихідну точку в HWCONFIG, це вказує, що Y1.12 призначений для виведення чітких сигналів.
Покидає нульову точку в позитивному напрямку, а потім зупиняється	0	8+0=8 8+1=9	Операція повернення нульової точки така ж, як і для коду 1 (режим 1).

(Режим 1)		(біт 3=ON)	
Режим 1 видає кількість імпульсів після повернення до нульової точки	Кількість імпульсів для руху	8+2=10 (біт 3=ON)	Операція повернення нульової точки така ж, як і для молодшого 16-бітного коду 1. Після повернення в нульову точку рух продовжується відповідно до кількості заданих імпульсів. Коли значення старшого 16-бітного коду є позитивним числом, рух відбувається в позитивному напрямку. Від'ємне число означає, що рух відбувається у негативному напрямку.
Режим 1 Пошук фази Z після повернення до нульової точки (Точка введення фази Z встановлюється в HWCONFIG)	Кількість фаз Z	8+3=11 (біт 3=ON)	Операція повернення нульової точки така ж, як і для молодшого 16-бітного коду 1. Після повернення до нульової точки рух продовжується відповідно до кількості Z-фаз для пошуку. Коли значення старшого 16-бітного коду є позитивним числом, рух відбувається в позитивному напрямку. Від'ємне число означає, що рух відбувається у негативному напрямку. Якщо тригер по наростаючому фронту для X0.1 є умовою для введення фази Z, підрахунок виконується один раз, коли виникає тригер по наростаючому фронту для X0.1.
Режим 1 Виводить чіткий сигнал після повернення до нульової точки (Точка очищення виводу встановлюється в HWCONFIG)	0 або кількість імпульсів або кількість фаз Z	12–15 (біт 3=біт 2 = ON)	Після повернення до нульової точки в режимі 1 виводиться сигнал очищення тривалістю 20 мс.
Точка DOG є точкою B		+16 (біт 4 = ON)	Коли в молодшому 16-бітному коді біт 4 увімкнено, це означає, що нульова точка увімкнена, коли точка DOG змінюється з ON на OFF, а нульова точка залишається, коли точка DOG змінюється з OFF на ON.

5. Послідовність виконання базується на значенні молодшого 16-бітного коду в таблиці вище та описано нижче.

- Напрямок сигналу DOG визначається значенням біта 5.
- Сигнал DOG визначається значенням біта 4.
- Режим 0 або режим 1 для повернення нульової точки, вибирається відповідно до значення біта 3.
- Операція повернення нульової точки виконується відповідно до значень біта 1 і біта 0.
- Виконується операція очищення сигналу, визначена бітом 2.

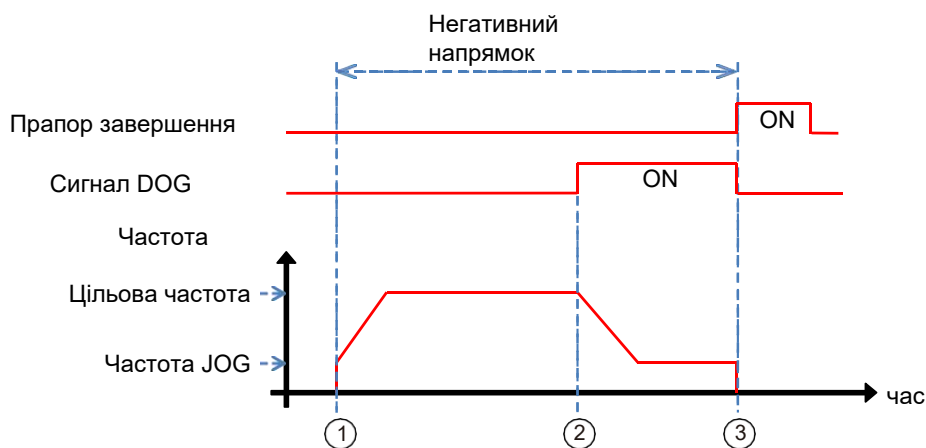
6. Встановіть точку введення та умову тригера наростаючого фронту в HWCONFIG, коли системі керування положенням потрібні позитивні та негативні граничні точки введення. Зауважте, що граничні вхідні точки не повинні збігатися з нульовою точкою або вхідними точками фази Z.

Для мікропрограми V1.10 або попередніх версій, коли виконується зворотний рух до нуля та активована функція перевірки граничної точки, прапор граничної сигналізації не встановлюється на ON після досягнення граничної точки. Для мікропрограми V1.12 або пізнішої версії прапорець тривоги обмеження буде автоматично увімкнено для нагадування, а потім ви зможете вирішити подальшу операцію на основі позначки тривоги. (Примітка: після того, як прапорець граничної тривоги увімкнено, його потрібно скинути самостійно.)

7. Прапор завершення встановлюється на ON після того, як інструкція завершує виконання вказаної функції. Наприклад, для коду функції 6 ПЛК встановлює прапор завершення на ON лише тоді, коли пошук фази Z завершено.
8. Після виконання інструкції DZRN програма обслуговування переривання не виконується, доки інструкція DZRN не буде вимкнена, якщо вказана точка введення для нульової точки збігається з точкою для зовнішнього переривання введення в програмі.
9. Якщо кінцевий вимикач вказано в HWCONFIG і існує програма обслуговування переривання зовнішнього введення, програма переривання буде виконана одночасно.
10. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз. Прошивка версії 1.12 або новішої підтримує мову програмування ST. Для отримання детальної інформації зверніться до опису в прикладі.

Кроки для виконання функцій наведені нижче, код функції 0:

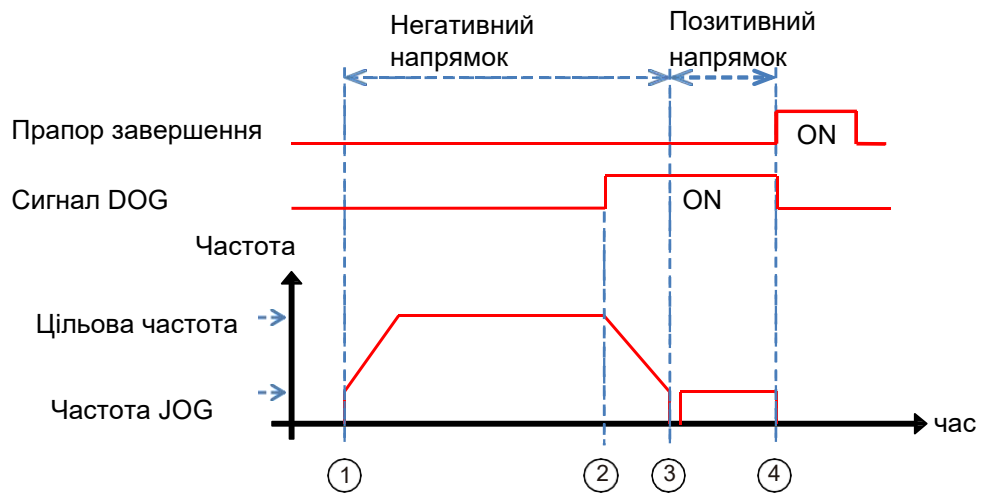
- ① Функція DZRN виконується, і пошук нульової точки відбувається в негативному напрямку з цільовою частотою.
- ② Після отримання сигналу DOG вихідна частота зменшується до частоти JOG. Вихід продовжує працювати в негативному напрямку і не припиняється, доки сигнал нульової точки не зміниться з ON на OFF
- ③ Вихід припиняється, коли сигнал змінюється з ON на OFF, а вісь віддаляється від сигналу DOG.



Код функції 1:

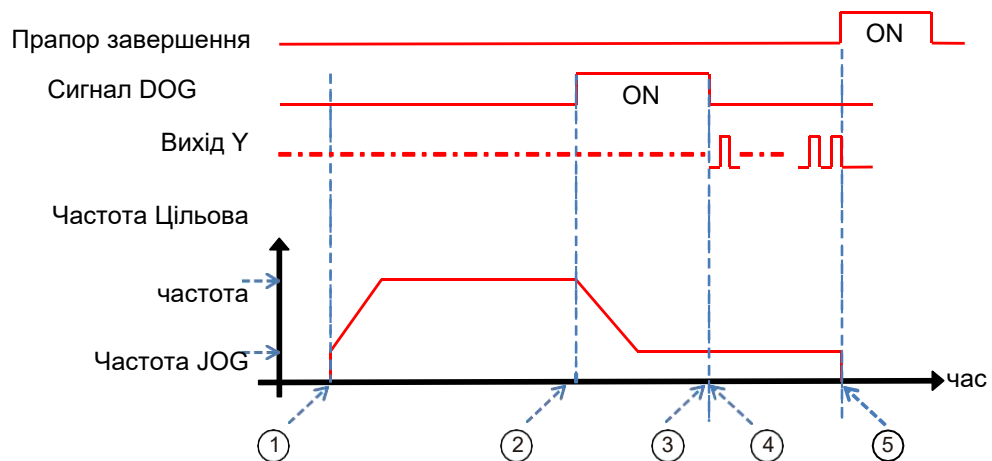
- ① Функція DZRN виконується, і пошук нульової точки відбувається в негативному напрямку з цільовою частотою.
- ② Після отримання сигналу DOG вихідний сигнал має позитивний напрямок із частотою JOG після того, як вихідна частота зменшується, і напрямок руху змінюється на протилежний. Вихід не припиняється, доки сигнал нульової точки не зміниться з ON на OFF

- ③ Вісь відходить від сигналу DOG, і ПЛК зупиняється, коли сигнал змінюється з ON на OFF



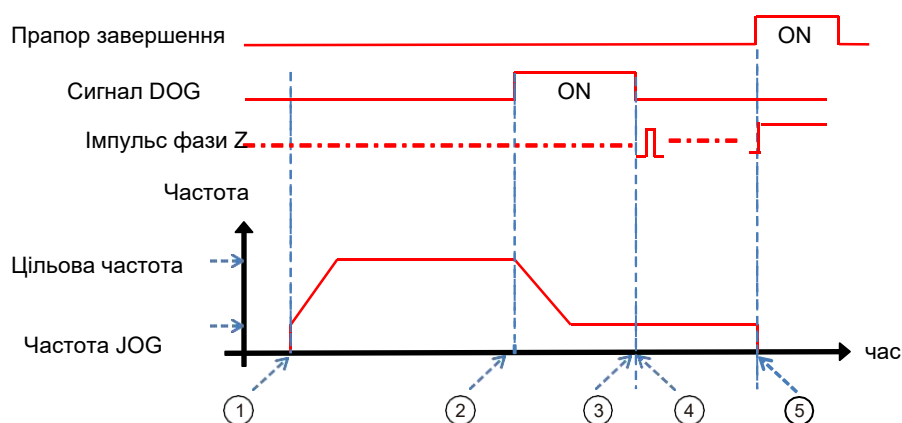
Код функції 2:

- ① Функція DZRN виконується, і пошук нульової точки відбувається в негативному напрямку з цільовою частотою.
- ② Після отримання сигналу DOG вихідний сигнал зменшує частоту до частоти JOG і продовжує рух у негативному напрямку.
- ③ Коли сигнал DOG залишається, і сигнал змінюється з ON на OFF, виводиться задана кількість імпульсів.
- ④ Починається вихід першого імпульсу.
- ⑤ Після завершення виведення 100-го імпульсу ПЛК зупиняється, а прапор завершення вмикається.



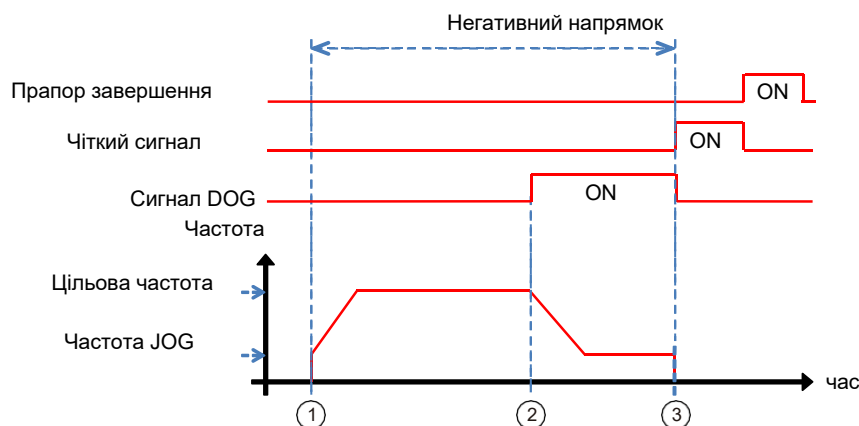
Код функції 3:

- ① Функція DZRN виконується, і пошук нульової точки відбувається в негативному напрямку з цільовою частотою.
- ② Після отримання сигналу DOG вихідна частота зменшується до частоти JOG і рух продовжується у негативному напрямку.
- ③ Рух продовжується відповідно до кількості фаз Z, коли сигнал змінюється з ON на OFF після того, як вісь відходить від сигналу DOG.
- ④ Перший імпульс фази Z
- ⑤ Рух припиняється після завершення $2^{-1}Z$ -фази та ввімкнення прапора завершення.



Код функції 4:

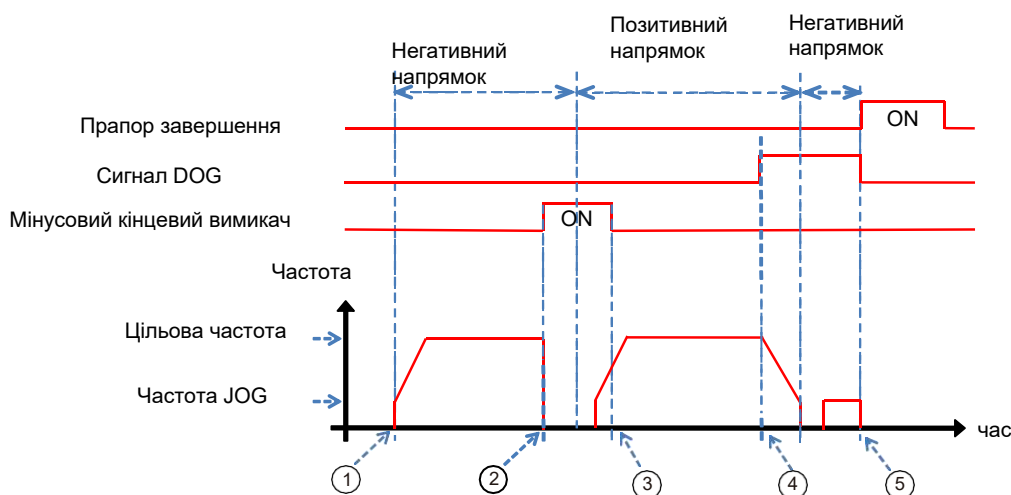
- ① Виконується DZRN і пошук нульової точки йде в негативному напрямку з цільовою частотою.
- ② Після отримання сигналу DOG вихідна частота зменшується до частоти JOG і продовжує рухатися у негативному напрямку. Вихід не припиняється, доки сигнал нульової точки не зміниться з ON на OFF
- ③ Після того, як вісь відійде від сигналу DOG, вихід припиняється, коли сигнал змінюється з ON на OFF, а сигнал очищення буде ON протягом приблизно 20 мілісекунд.



Код функції 0+ функція негативного обмеження, що дозволяє:

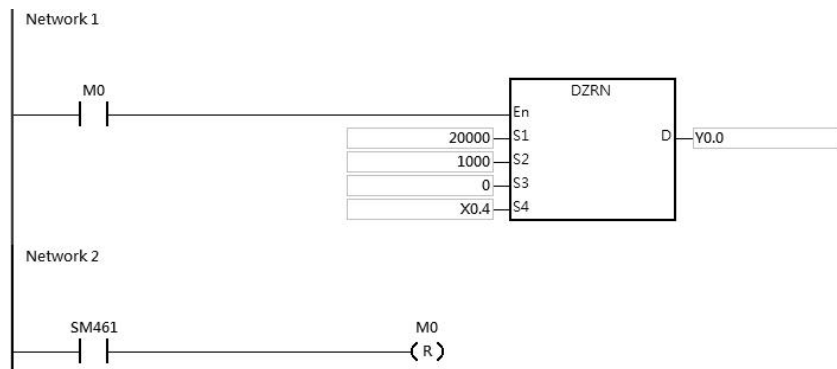
Ви встановлюєте вхідну точку негативного обмеження в HWCONFIG, а потім завантажуєте налаштування в ПЛК. ПЛК автоматично розраховує функцію негативного обмеження під час виконання інструкції.

- ① Функція DZRN виконується, і пошук нульової точки відбувається в негативному напрямку з цільовою частотою.
- ② Після увімкнення негативного кінцевого вимикача рух припиняється, а потім повертається в позитивному напрямку після зміни напрямку.
- ③ Після виходу з негативного кінцевого вимикача рух продовжується в позитивному напрямку.
- ④ Вихідна частота зменшується після отримання сигналу DOG. Зворотний вихід виконується з частотою JOG після зміни напрямку.
- ⑤ Вихід припиняється, коли сигнал змінюється з ON на OFF після того, як вісь відходить від сигналу DOG.



Приклад 1

Коли M0 увімкнено, виведення імпульсу з Y0.0 з частотою 20 кГц для пошуку нульової точки в негативному напрямку. Коли отримано сигнал DOG і X0.4 увімкнено, він продовжує рухатися в негативному напрямку з частотою JOG 1 кГц. Вихід припиняється відразу після зміни X0.4 з ON на OFF.



Пояснення

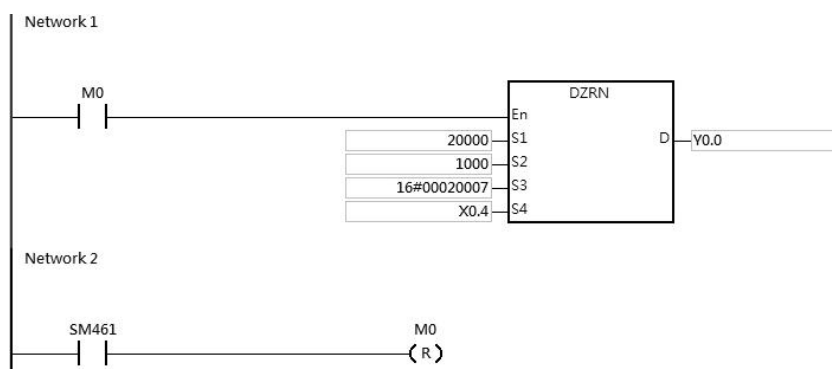
S_3 встановлено на 0.

Старший 16-бітний [0000] вмикає функцію переміщення кількості імпульсів або пошуку Z-фаз.

Нижчий 16-бітний [0000] — це режим 0; при отриманні сигналу DOG вісь рухається в негативному напрямку; після того, як вісь відійде від сигналу DOG, він негайно зупиниться.

Приклад 2

Коли M0 увімкнено, виведення імпульсу з Y0.0 з частотою 20 кГц для пошуку нульової точки в негативному напрямку. Коли отримано сигнал DOG і X0.4 увімкнено, ПЛК знижує частоту до частоти JOG 1 кГц і продовжує рухатися в позитивному напрямку з частотою JOG 1 кГц. Коли X0.4 вимкнено, ПЛК починає шукати імпульс фази Z у позитивному напрямку. Коли X0.5 отримує два імпульси, ПЛК зупиняється, а Y1.4 видає імпульс тривалістю 20 мс.



Пояснення

1. Якщо ви вкажете тригер по наростаючому фронту на X0.5 як умову для введення фази Z у HWCONFIG, кількість збільшується один раз, коли виникає тригер по наростаючому фронту на X0.5.

2. Y1.4 вказано як вихідну точку для виведення сигналу очищення в HWCONFIG.

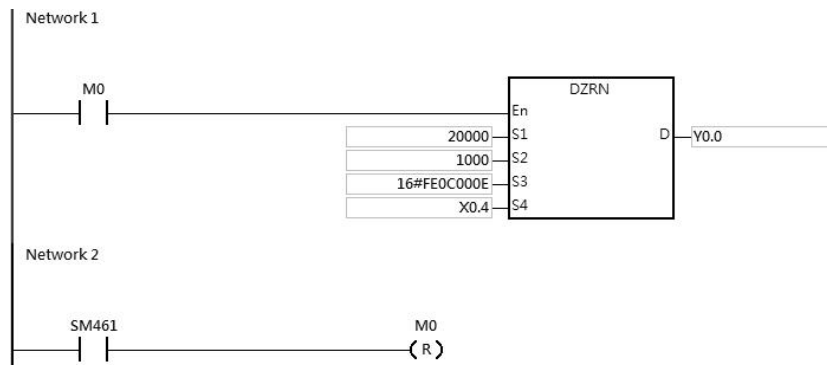
3. S_3 встановлено як 16#FE0C000E.

Старший 16-бітний [0002] означає двічі шукати фазу Z, коли вісь відходить від сигналу DOG.

Нижчий 16-бітний [0007] є режимом 7; при отриманні сигналу DOG вісь рухається в негативному напрямку; після того, як вісь відійде від сигналу DOG і знайде фазу Z, виводиться чистий сигнал (20 мс).

Приклад 3

Коли M0 увімкнено, виведення імпульсу з Y0.0 з частотою 20 кГц для пошуку нульової точки в негативному напрямку. Коли отримано сигнал DOG і X0.4 увімкнено, ПЛК знижує частоту до частоти JOG 1 кГц і продовжує рухатися в позитивному напрямку з частотою JOG 1 кГц. Коли X0.4 вимкнено, вісь починає рухатися після завершення виведення 500 плюсів у негативному напрямку. Y1.4 видає імпульс тривалістю 20 мс, а потім припиняє виведення.

**Пояснення**

1. Y1.4 вказано як вихідну точку для виведення сигналу очищення в HWCONFIG.

2. S₃ встановлено як 16#FE0C000E.

Високий 16-біт [FE0C=-500], коли вісь віддаляється від сигналу DOG. Вісь починає рухатися після завершення виведення 500 плюсів у негативному напрямку.

Нижчий 16-бітний [000E] – це режим 14; при отриманні сигналу DOG вісь рухається в позитивному напрямку; після того, як вісь відійде від сигналу DOG, виводиться чистий сигнал (20 мс).

6**Приклад 4 (мовою програмування ST)**

Загалом, під час використання цієї інструкції для редагування програми пропонується використовувати мову сходової діаграми. Якщо ви використовуєте мову ST (яка доступна лише для мікропрограми версії 1.12 або новішої), ви повинні дочекатися завершення руху повернення до нуля (прапор завершення SM461 увімкнено), а потім вимкнути інструкцію. Крім того, ви повинні використовувати прапор автоматичного скидання разом, щоб повідомити ПЛК про звільнення права керування високошвидкісним виходом.

Взявши, наприклад, вихід Y0.0, написання програми показано на малюнку нижче

```

0001 IF M0 THEN
0002     DZRn(1000,100,0,X0.0,Y0.0);
0003     M1:=TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;

```

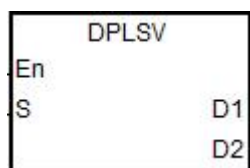
API	Інструкція			Операнд							опис						
2705	D	PLSV		S, D ₁ , D ₂							Регульований імпульсний вихід						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S							●	●	●		O		O	O		
D ₁		O														
D ₂		O	O													

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S			●				●						
D ₁	●												
D ₂	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



S : вихідна частота імпульсів

D₁ : пристрій імпульсного виведення

D₂ : Пристрій виведення напрямку імпульсу

Пояснення

1. Ця інструкція регулює вихідну частоту імпульсу та напрям (S) для пристроїв, зазначених у D та D₂. S уточнює

вихідна частота імпульсів: для лінійних моделей з вихідним драйвером діапазон становить від -4 МГц до +4 МГц; для моделей виходу з відкритим колектором діапазон становить від -200 кГц до +200 кГц. Знаки мінус і плюс вказують на позитивний напрямок і негативний напрямок імпульсу. Ви можете змінювати частоту вихідного імпульсу під час виведення імпульсу; але якщо нова частота відрізняється від попередньої частоти за напрямком, інструкція зупиняє вихід, а через один цикл знову виводить цільову частоту в новому напрямку.

2. Зверніться до наступної таблиці для вибору пристроїв імпульсного виведення для D₁ і D₂. Якщо режим виведення для D₁ і D₂ не є виходом за замовчуванням Pulse+direction (0 є значенням за замовчуванням), змініть режим, встановивши SR на 1, що змінює режим на вихід фази A/B.

Номер осі	Вісь 1	Вісь 2	Вісь 3	Вісь 4	Вісь 5	Вісь 6
Вихідна точка для D ₁	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
Вихідна точка напрямку для D ₂	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Режим виведення	SR462	SR482	SR502	SR522	SR542	SR562
Зайнятий прапор	SM460	SM480	SM500	SM520	SM540	SM560

Вихід поточного регістру	SR460	SR480	SR500	SR520	SR540	SR560
	SR461	SR481	SR501	SR521	SR541	SR561

3. Вихідна точка для пристрою виводу напрямку імпульсу **D₂** зазнає впливу циклу сканування, якщо вона не рекомендована
- вихідну точку в таблиці вище.
4. Пристрій **D₂** виведення напрямку імпульсу змінює свій власний стан відповідно до знака мінус або плюс у **S**. **D₂** вимкнено, якщо **S** плюс (+), а **D₂** увімкнено, якщо **S** мінус (-).
5. В інструкції немає налаштувань збільшення або зниження. Інструкція не виконує нарощування робота на початку або зниження при зупинці. Якщо вам потрібна функція підвищення та зниження частоти, використовуйте інструкцію DRAMP (API 0703) для збільшення та зменшення частоти вихідного імпульсу.
6. Поки інструкція виконує імпульсний вихід, вихід негайно припиняється, якщо стан приводу змінюється на OFF
7. Якщо виходи спрямованості не використовують виходи за замовчуванням для виведення, ви можете звернутися до таблиці нижче, щоб налаштувати час для виведення спрямованого виведення перед імпульсним виведенням. Щоб ви могли бути впевнені, що виведення відбувається лише під час перемикання на виходи напрямку. Одиницею для SR є мілісекунда, а діапазон – 0-20 мс. За замовчуванням 0, що вказує на неактивність.

Вихідна точка для D₁	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5
Номер SR	SR640	SR641	SR642	SR643	SR644	SR645
Вихідна точка для D₁	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
Номер SR	SR646	SR647	SR648	SR649	SR650	SR651

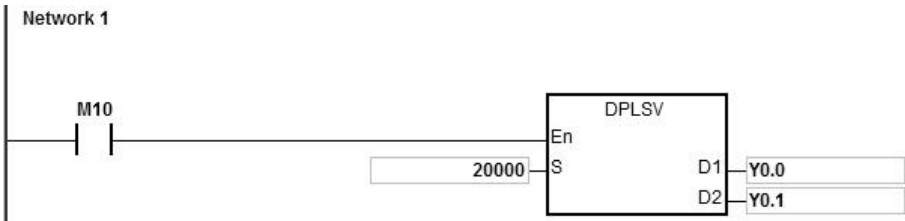
6

Примітка: джерелом часу тут є час сканування ПЛК. (доступно для мікропрограми версії 1.02.20 або новішої)

8. Зверніться до прикладу 3 з інструкції DJOG (API 2703) для програмування на мові ST.

приклад

Коли M10 увімкнено, Y0.0 видає імпульс із частотою 20 кГц. Y0.1 = ВИМК означає позитивний напрямок вихідного імпульсу.



API	Інструкція			Операнд							опис						
2706	D	DRV1		S_1 , S_2 , D_1 , D_2							Контроль відносної позиції						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
D_1		○														
D_2		○	○													

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
D_1	●												
D_2	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DDRVI	
En	
S1	D1
S2	D2

- S_1 : Кількість вихідних імпульсів (відносно позиціонування)
- S_2 : Вихідна частота імпульсів
- D_1 : Пристрій виведення імпульсів
- D_2 : вихідний пристрій напрямку імпульсу

Пояснення

1. Ця інструкція визначає налаштування вихідного імпульсу для відносного позиціонування. S_1 – кількість вихідних імпульсів (відносно позиціонування). Діапазон становить від -2 147 483 648 до +2 147 483 647, а знаки +/- вказують на позитивний і негативний напрямок імпульсу. Коли інструкція виконується і S_1 дорівнює 0, це вказує на те, що виведення не виконується, а прапор завершення виведення не має значення ON. Якщо 0 є можливим вихідним числом у вашій програмі, пропонується додати більше умов у вашу програму, щоб виключити таку можливість.
2. S_2 — частота імпульсного виходу, а діапазон становить від 0 Гц до 200 кГц для моделей виходу з відкритим колектором і 0—4 МГц для моделей виходу лінійного драйвера. Якщо значення в S_2 менше початкової/кінцевої частоти (Гц), встановленої в SR (додаткову інформацію див. у таблиці SR нижче), ПЛК працює відповідно до значень, встановлених у SR.
3. Зверніться до наведеної нижче таблиці для вибору пристроїв імпульсного виведення для D_1 і D_2 . Якщо режим виведення для D_1 і D_2 не є вихідним сигналом за замовчуванням Pulse+direction (0 є значенням за замовчуванням), змініть режим, встановивши SR на 1, що змінює режим на вихід фази A/B.

Номер осі	Вісь 1	Вісь 2	Вісь 3	Вісь 4	Вісь 5	Вісь 6
Вихідна точка для D_1	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
Напрямок вихідної точки для D_2	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Режим виведення	SR462	SR482	SR502	SR522	SR542	SR562

DDRVI та DDRVA підтримують вихідні пристрої для D_1 , які можуть бути непарними точками, як показано в наступній таблиці.

Підтримується лише режим виведення Pulse+direction. Не рекомендується використовувати вхідні точки Y0.0, Y0.2–Y0.10 як високошвидкісні вихідні точки.

Номер осі	Вісь 7	Вісь 8	Вісь 9	Вісь 10	Вісь 11	Вісь 12
Вихідна точка для D_1	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Напрямок вихідної точки для D_2	Y1.0 або вище або будь-який пристрій M (BOOL)					

4. На вихідну точку для пристрою виводу напрямку імпульсу D_2 впливає цикл сканування, якщо він не рекомендований вихідну точку в таблиці вище.

5. Коли D_2 використовує вихідний режим Pulse+direction і S_1 має знак мінус, D_2 увімкнено. Коли S_1 має знак плюс, D_2 є OFF. D_2 не вимикається відразу після завершення виведення імпульсу в негативному напрямку. D_2 перемикається на ВІМК, коли контакт для виконання інструкції OFF.

6. Для виконання інструкцій ви повинні використовувати такі параметри, як початкова/кінцева частота та наростання/зниження

час. Зверніться до наступної таблиці для SM/SR, що відповідає осям.

Наступна таблиця застосовується, коли D_1 вибирає парні вихідні точки як вихідні пристрої.

Вихідна точка для D_1	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
Зайнятий прапор	SM460	SM480	SM500	SM520	SM540	SM560
Прапор завершення	SM461	SM481	SM501	SM521	SM541	SM561
Реверсування виходу напрямку	SM462	SM482	SM502	SM522	SM542	SM562
Прапор зупинки	SM463	SM483	SM503	SM523	SM543	SM563
Включення фіксованого нахилу підвищення/зниження	SM469	SM489	SM509	SM529	SM549	SM569
Завершення виходу автоматичне скидання	SM470	SM490	SM510	SM530	SM550	SM570

Виконання переривання запрограмувати, коли вихід імпульсу закінчується	SM471	SM491	SM511	SM531	SM551	SM571
Відповідний перервати І число	I500	I501	I502	I503	I504	I505
Вихід негайно зупиняється, коли інструкція вимкнена або зупиняється	SM476	SM496	SM516	SM536	SM556	SM576
Поточна вихідна позиція	SR460	SR480	SR500	SR520	SR540	SR560
	SR461	SR481	SR501	SR521	SR541	SR561
Початкова/кінцева частота (Гц)	SR463	SR483	SR503	SR523	SR543	SR563
Час розгону (мс)	SR464	SR484	SR504	SR524	SR544	SR564
Час зниження (мс)	SR465	SR485	SR505	SR525	SR545	SR565
Цільова частота фіксованого нахилу	SR472	SR492	SR512	SR532	SR552	SR572
	SR473	SR493	SR513	SR533	SR553	SR573
Поточна вихідна частота	SR610	SR612	SR614	SR616	SR618	SR620
	SR611	SR613	SR615	SR617	SR619	SR621

Наступна таблиця застосовується, коли D_1 вибирає непарні вихідні точки як вихідні пристрої.

Вихідна точка для D_1	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Зайнятий прапор	SM472	SM492	SM512	SM532	SM552	SM572
Прапор завершення	SM473	SM493	SM513	SM533	SM553	SM573
Прапор зупинки виведення	SM474	SM494	SM514	SM534	SM554	SM574
Завершення виходу автоматичне скидання	SM475	SM495	SM515	SM535	SM555	SM575
Вихід негайно зупиняється, коли інструкція вимкнена або зупиняється	SM477	SM497	SM517	SM537	SM557	SM577
Поточна вихідна позиція	SR474	SR494	SR514	SR534	SR554	SR574
	SR475	SR495	SR515	SR535	SR555	SR575

Початок/кінець частота (Гц)	SR476	SR496	SR516	SR536	SR556	SR576
Час наростання/вимкнення (мс)	SR477	SR497	SR517	SR537	SR557	SR577

Деякі прапорці, такі як фіксований нахил підйому/зменшення та виконання програми переривання після завершення імпульсного виведення,

не підтримується непарними вихідними точками.

Якщо вихідний сигнал триває, попередньо встановленим вихідним сигналом є зупинка темпу зниження, коли інструкція вимкнена або тимчасово зупиняється. Якщо вихід потрібно негайно зупинити, установіть прапор негайної зупинки на ON.

7. Для ПЛК із FW V1.06.00 або новішої версії додано нові SR652, SR653 для відображення кількості імпульсів у зоні підвищення та нові SR654, SR655 для відображення кількості імпульсів у зоні зниження.

Зверніться до наступної таблиці для атрибутів, заводських налаштувань і фіксованих значень SM/SR, що відповідають осям.

Пункт	Атрибут	Stop → Run	Run → Стоп	Увімкнути	Фабрика за замовчуванням	Засув
Зайнятий прапор	R	OFF	OFF	OFF	OFF	N
Прапор завершення	R / W	OFF	OFF	OFF	OFF	N
Прапор зупинки	R / W	OFF	OFF	OFF	OFF	N
Завершення виходу автоматичне скидання	R / W	OFF	OFF	OFF	OFF	N
Включення фіксованого нахилу підвищення/зниження	R / W	OFF	OFF	OFF	OFF	N
Виконання переривання запрограмувати, коли вихід імпульсу закінчується	R / W	OFF	OFF	OFF	OFF	N
Вихід негайно зупиняється, коли інструкція вимкнена або зупиняється	R / W	OFF	OFF	OFF	OFF	N
Поточний вихід положення	R / Wd	Без змін	Без змін	Без змін	0	Y
Початок/кінець частота (Гц)	R / Wd	Без змін	Без змін	Без змін	200	Y
Час розгону (мс)	R / Wd	Без змін	Без змін	Без змін	200	Y

Час зниження (мс)	R / Wd	Без змін	Без змін	Без змін	200	Y
Цільова частота фіксований нахил	R / Wd	Без змін	Без змін	Без змін	0	Y
Поточний вихід частота	R	0	0	0	0	N

Примітка: R означає, що пристрій читається. W означає, що пристрій доступний для запису. Wd означає, що пристрій доступний для запису в будь-який час

за винятком виконання високошвидкісного виведення.

8. Після ввімкнення автоматичного скидання завершення виведення прапорець автоматично скидається на OFF, коли виведення завершується. Ви повинні знову встановити прапор на ON, якщо функція автоматичного скидання потрібна для наступної операції виведення. Рекомендується використовувати наростаючий або спадаючий фронт, щоб активувати прапор кожного разу. Ця функція зазвичай використовується, коли вихідна інструкція DDRVI не може бути просканована та виконана програмою ПЛК після виконання DDRVI. Наприклад, коли інструкція виконується в програмі переривання.

9. Якщо ви використовуєте прапор завершення для встановлення функції переривання I, прапор автоматично скидається на ВИМК, коли виведення завершується і запускається програма переривання. Ви повинні знову встановити прапор на ON для наступного виводу, у якому виникає переривання. Рекомендується використовувати наростаючий або спадаючий фронт, щоб активувати прапор кожного разу.

10. Якщо виходи напрямку не використовують виходи за замовчуванням для виведення, ви можете звернутися до таблиці нижче для налаштування

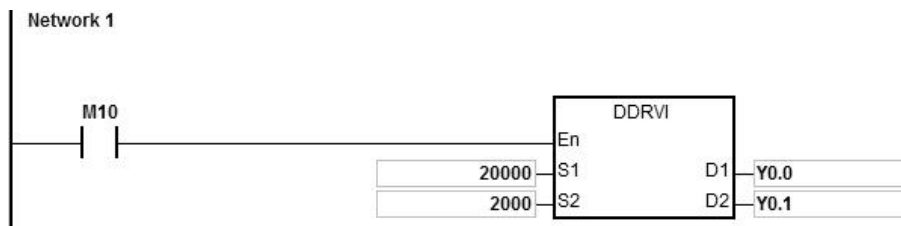
час спрямованого виведення йде першим перед виведенням імпульсу. Щоб ви могли бути впевнені, що виведення відбувається лише під час перемикавання на виходи напрямку. Одиницею для SR є мілісекунда, а діапазон – 0-20 мс. За замовчуванням 0, що вказує на неактивність.

Вихідна точка для D ₁	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0,5
Номер SR	SR640	SR641	SR642	SR643	SR644	SR645
Вихідна точка для D ₁	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
Номер SR	SR646	SR647	SR648	SR649	SR650	SR651

Примітка: джерелом часу тут є час сканування ПЛК. (доступно для мікропрограми версії 1.02.20 або новішої)

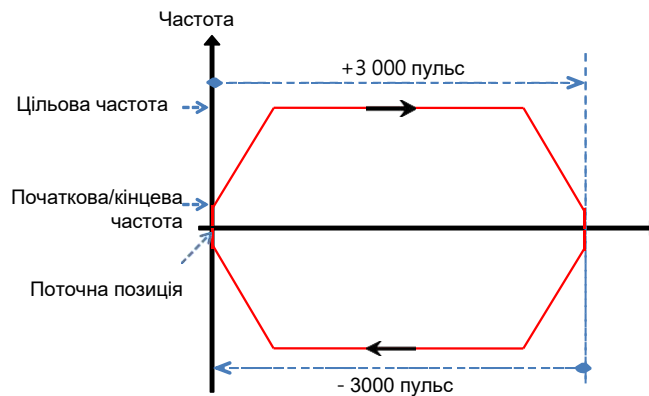
Приклад 1

Коли M10 увімкнено, Y0.0 видає 20 000 імпульсів на 2 кГц (відносно позиціонування). Y0.1 = ВІМК означає позитивний напрям.



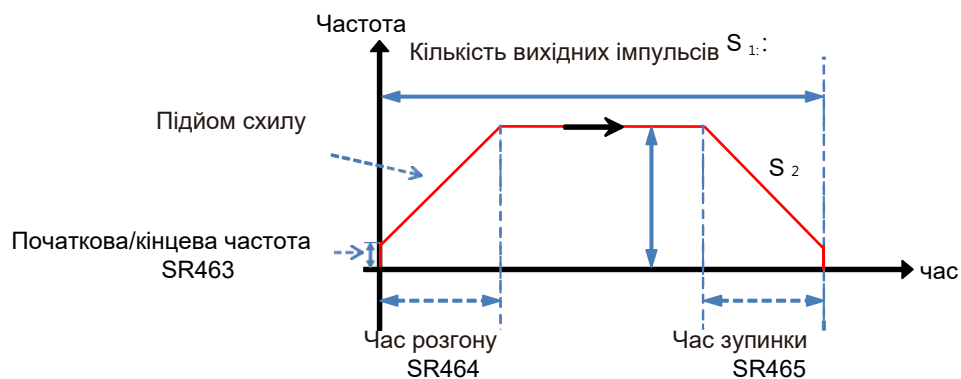
Додаткові зауваження

- На наступному графіку пояснюється відносно позиціонування шляхом визначення відстані шляху від поточного положення за допомогою знаки +/-.



- Налаштування рампи вгору/вниз і пунктів для відносного позиціонування

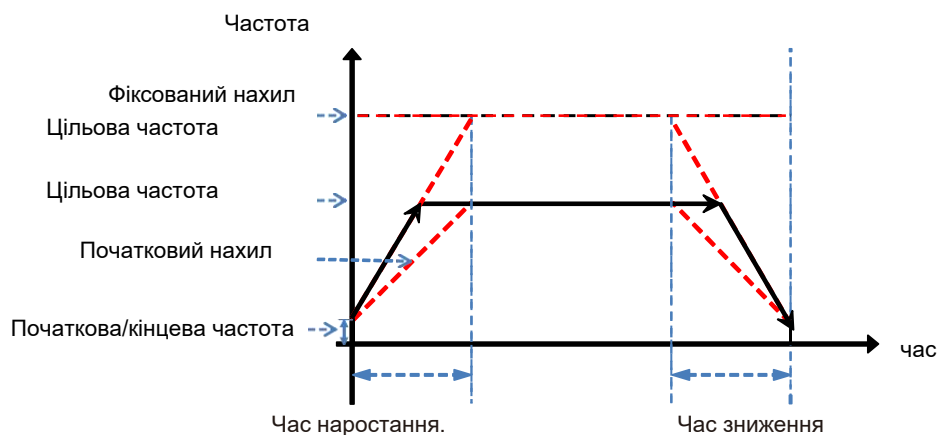
а) Діаграма вихідної кривої Y0.0



- Ви можете створити кілька інструкцій DRVI, використовуючи ту саму точку виведення в програмі. ПЛК може запускатися лише одна інструкція DRVI при виконанні програми. Наприклад, коли одна інструкція запускає Y0.0 для виведення, тоді інші інструкції, які також використовують Y0.0 для виведення, не виконуються. Інструкція, яка першою запускає точку виведення, спочатку використовує точку.
- Після виконання інструкції будь-які зміни параметрів не приймаються, доки інструкцію не буде вимкнено.

3. Збільшення та зниження для фіксованих схилів

а) Наступний графік показує діаграму вихідної кривої Y0.0.



б) Запустіть прапор фіксованого нахилу SM469 і запишіть цільову частоту фіксованого нахилу в SR472.

с) Новий нахил займає місце початкового нахилу, а інструкція DDRVI виконує позиціонування, як показано вище.

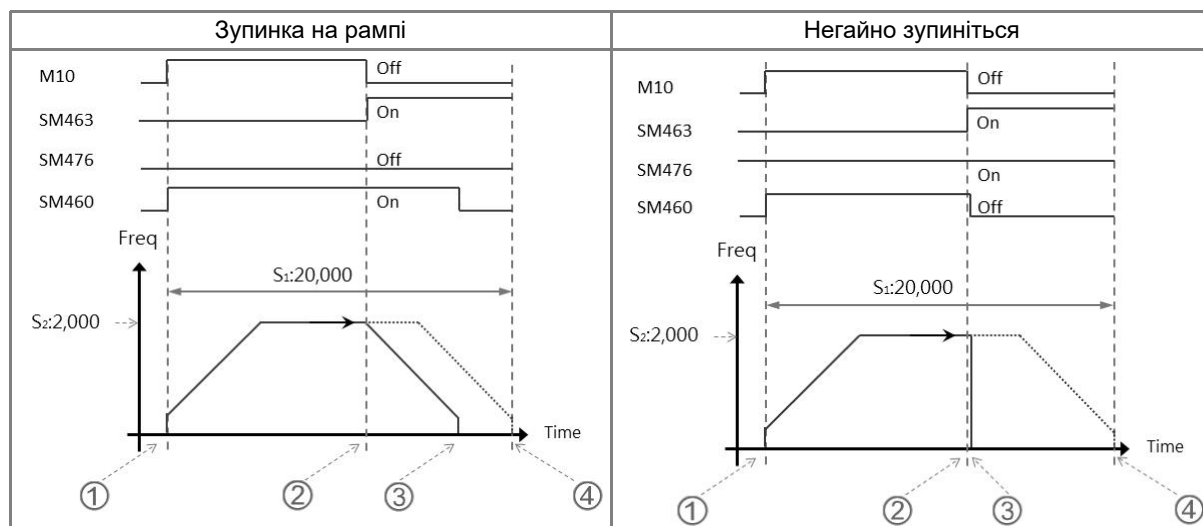
б) Загальний нахил визначається початковою/кінцевою частотою, цільовою частотою та часом наростання та спаду.

Фіксований нахил визначається початковою/кінцевою частотою, цільовою частотою фіксованого нахилу та часом наростання та зниження.

е) Якщо цільова частота фіксованого нахилу менша за цільову частоту, функція фіксованого нахилу не почався.

4. Прапор зупинки та негайно зупинити прапор

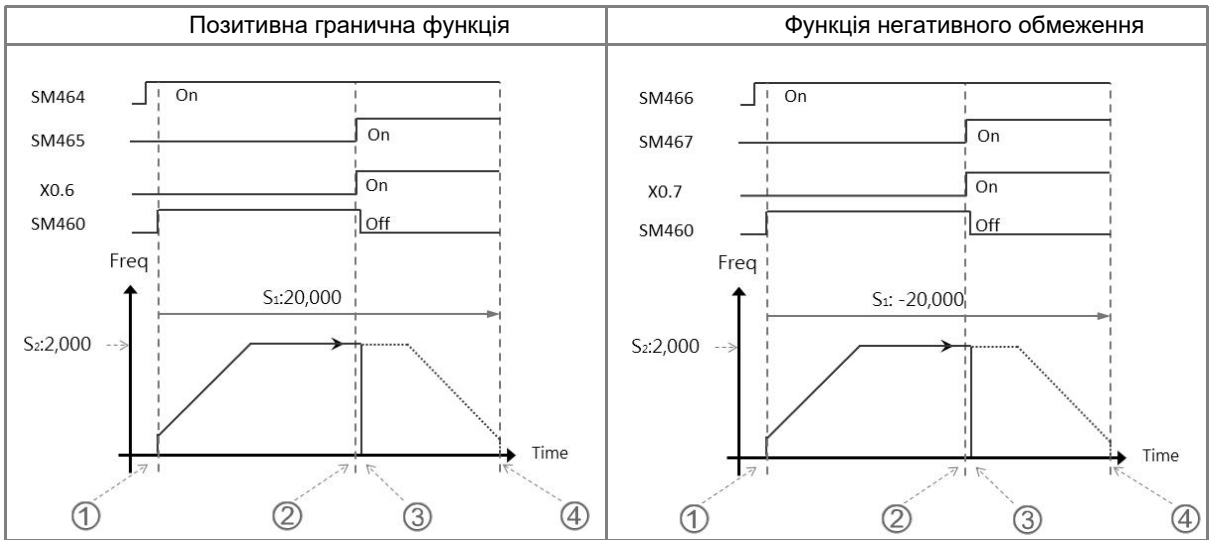
а) Наступна графіка показує вихідну діаграму на основі прикладу 1.



- b) Зупинка на рампі
- ① Коли M10 увімкнено, інструкція DDRVI виконується та починає виводити імпульси, а на SM460 прапор зайнято увімкнено.
 - ② Коли інструкцію вимкнено (M10 вимкнено) або прапор зупинки SM463 увімкнено, стан прапора негайної зупинки SM476 підтверджується.
 - ③ Якщо прапор негайної зупинки SM476 OFF, вихідний сигнал зменшується до зупинки, а потім прапор зайнятості SM460 OFF.
 - ④ Якщо M10 не вимкнено, а прапор зупинки SM463 не увімкнено після виконання інструкції, Y0.0 зупиняє вихід після 20 000 імпульсів.
- c) Негайно зупиниться
- ① Коли M10 увімкнено, інструкція DDRVI виконується та починає виводити імпульси, а на SM460 прапор зайнято увімкнено.
 - ② Якщо інструкцію вимкнено (M10 вимкнено) або прапор зупинки SM463 увімкнено, стан прапора негайної зупинки SM476 підтверджується.
 - ③ Якщо прапор негайної зупинки SM476 увімкнуто, вихід негайно припиняється, а прапор зайнятості SM460 вимкнено.

5. Функція апаратного обмеження

а) На наступному малюнку показано діаграму вихідної кривої осі 1 (Y0.0/Y0.1).



- б) Операція позитивної межі
- ① Точка введення позитивного обмеження X0.6 встановлюється в HWCONFIG і завантажується в ПЛК.
- Коли SM464 встановлено на ON, запускається функція позитивного обмеження.

① DDRVI видає 20 000 імпульсів у позитивному напрямку. **Примітка:** інструкція не виводиться в позитивному напрямку, якщо позитивна гранична точка ввімкнена під час запуску функції.

② Коли вхідна точка обмеження X0.6 запускається зовнішнім механізмом і ввімкнена, умова для апаратної функції позитивного обмеження виконується.

③ Інструкція DDRVI негайно припиняє вихід, а сигнал позитивного обмеження SM465 увімкнено.

④ Якщо умова функції апаратного обмеження не виконується, Y0.0 припиняє вихід після 20 000 імпульсів.

c) Спрацьовування негативного ліміту

① Від'ємна межа вхідної точки X0.7 встановлюється в HWCONFIG і завантажується в ПЛК. Коли SM466 встановлено на ON, запускається функція негативного обмеження.

① DDRVI видає 20 000 імпульсів у негативному напрямку. **Примітка:** інструкція не виводиться в негативному напрямку, якщо негативна гранична точка ввімкнена під час запуску функції.

② Коли вхідна точка обмеження X0.7 запускається зовнішнім механізмом і ввімкнена, умова для апаратної функції негативного обмеження виконується.

③ Інструкція DDRVI негайно припиняє виведення, а сигнал негативного обмеження SM467 увімкнено.

④ Якщо умова функції апаратного обмеження не виконується, Y0.0 припиняє вихід після -20 000 імпульсів.

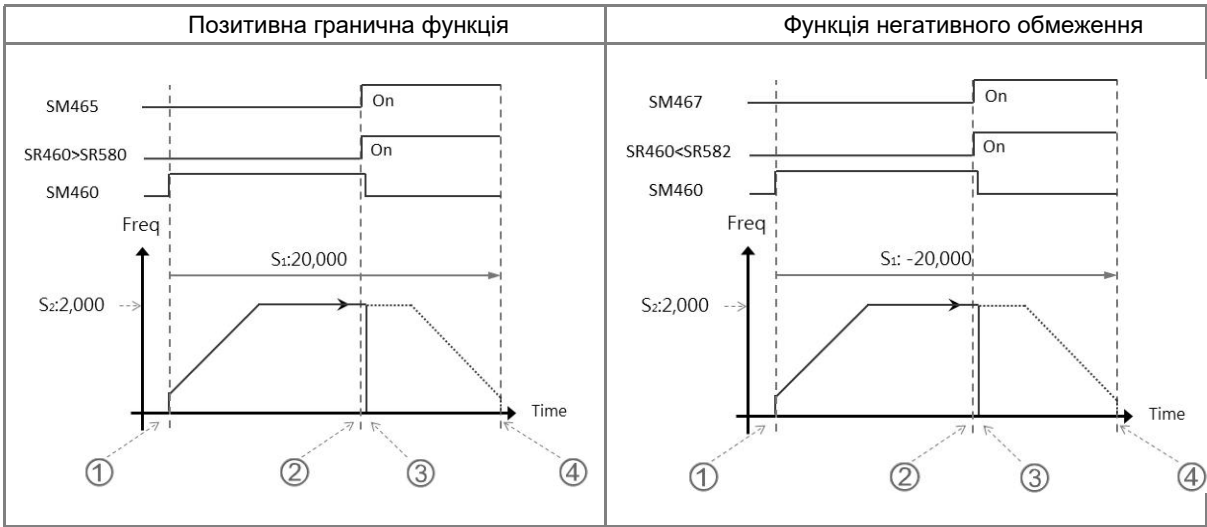
d) Прапори SM, що стосуються функції обмеження

Номер осі	Вісь 1	Вісь 2	Вісь 3	Вісь 4	Вісь 5	Вісь 6
Увімкнення апаратна позитивна межа	SM464	SM484	SM504	SM524	SM544	SM564
Позитивний граничний сигнал прапор	SM465	SM485	SM505	SM525	SM545	SM565
Увімкнення апаратне негативне обмеження	SM466	SM486	SM506	SM526	SM546	SM566
Сигналізація негативного ліміту прапор	SM467	SM487	SM507	SM527	SM547	SM567

Примітка. Прапор тривоги граничної точки встановлюється або скидається автоматично ПЛК відповідно до стану прапора запуску.

6. Функція програмного обмеження

а) Наступна діаграма показує діаграму вихідної кривої осі 1 (Y0.0/Y0.1).



б) Операція позитивної межі

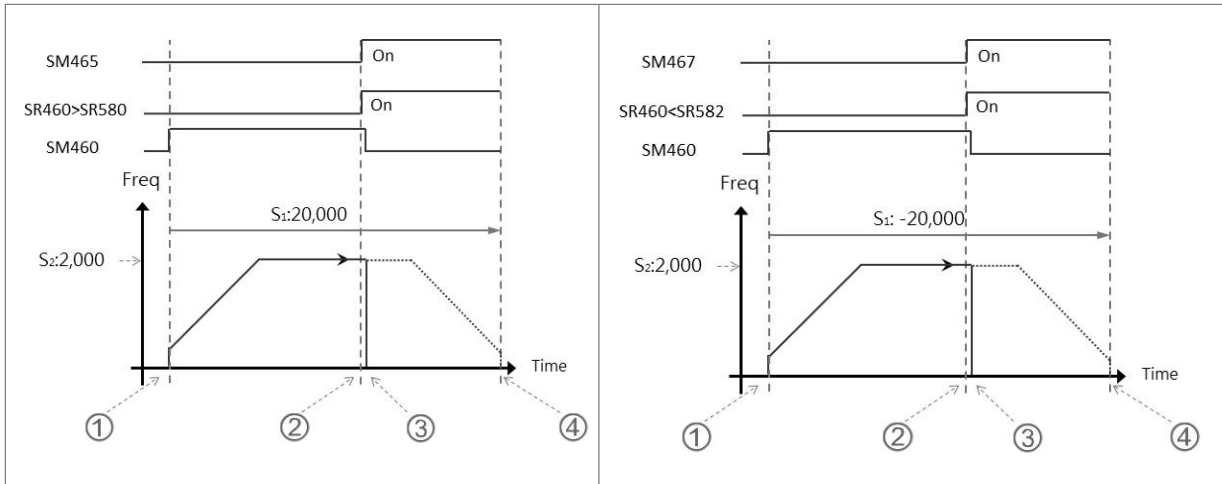
① Позитивне програмне обмеження встановлюється в HWCONFIG (наприклад, установкою SR580=11 000). Після завантаження налаштування в ПЛК, це означає, що функція позитивного обмеження запускається.

② DDRVI починає видавати 20 000 імпульсів у позитивному напрямку. **Примітка:** інструкція не виводиться в позитивному напрямку, якщо позитивна межа перевищена під час запуску функції.

③ Коли поточне положення SR460 > граничне положення SR580, це означає, що умова програмного обмеження функція виконана.

④ Інструкція DDRVI негайно припиняє вихід, а сигнал позитивного обмеження SM465 увімкнено.

⑤ Якщо умова функції обмеження не виконується, Y0.0 припиняє вихід після 20 000 імпульсів.



с) Спрацьовування негативного ліміту

① Негативне програмне обмеження встановлюється в HWCONFIG (наприклад, установкою SR582 = -11 000). Після завантаження налаштування в ПЛК запускається функція негативного обмеження.

① DDRVI починає видавати 20 000 імпульсів у негативному напрямку. **Примітка:** інструкція не виводиться в негативному напрямку, якщо негативний ліміт перевищено під час запуску функції.

② Коли поточне положення SR460 < граничного положення SR582, виконується умова для програмної функції негативного обмеження.

③ Інструкція DDRVI негайно припиняє виведення, а сигнал негативного обмеження SM467 увімкнено.

④ Якщо умова функції програмного обмеження не виконується, Y0.0 припиняє вихід після -20 000 імпульсів.

д) Тригер по висхідному фронту

Гранична функція	Контакт А	Контакт Б
Типи	Тригер по висхідному фронту	Тригер із спадною кромкою
Описи	 При запуску: від низького до високого і підтримується на високому	 При запуску: від високого до низького і підтримується на низькому

д) СР, пов'язана з граничною функцією

Номер осі	Вісь 1	Вісь 2	Вісь 3	Вісь 4	Вісь 5	Вісь 6
Позитивне граничне положення програмного забезпечення	SR580	SR584	SR588	SR592	SR596	SR600
	SR581	SR585	SR589	SR593	SR597	SR601
Негативне програмне обмеження	SR582	SR586	SR590	SR594	SR598	SR602
	SR583	SR587	SR591	SR595	SR599	SR603

Позитивний вихідний ліміт: вихід негайно припиняється, якщо поточне положення перевищує позитивне граничне положення.

Обмеження від'ємного виходу: вихід припиняється негайно, якщо поточна позиція менша за від'ємне граничне положення.

Коли і позитивне, і негативне обмеження дорівнюють 0, функція програмного обмеження не може запуститися. Оскільки функція програмного обмеження перевіряє вихідну позицію під час сканування вихідної інструкції, зупинка виведення залежить від сканування ПЛК. Щоб швидко зупинити вихід у реальному часі, використовуйте зовнішню точку введення як граничну точку.

7. Приклад ST

```

0001 IF M0 THEN
0002     DDRVI (1000,1000,Y0.0,Y0.1);
0003     M1 := TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;

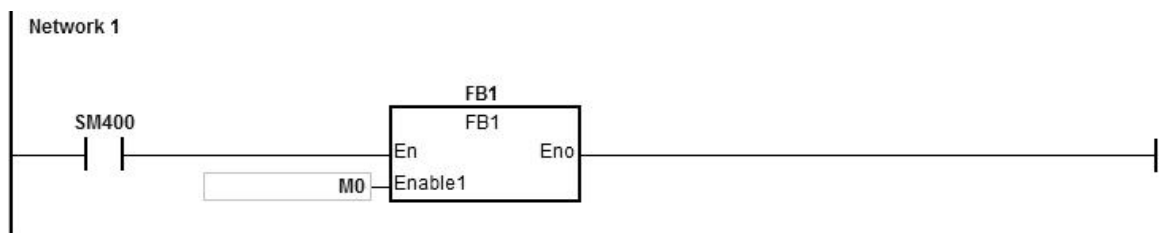
```

Пояснення

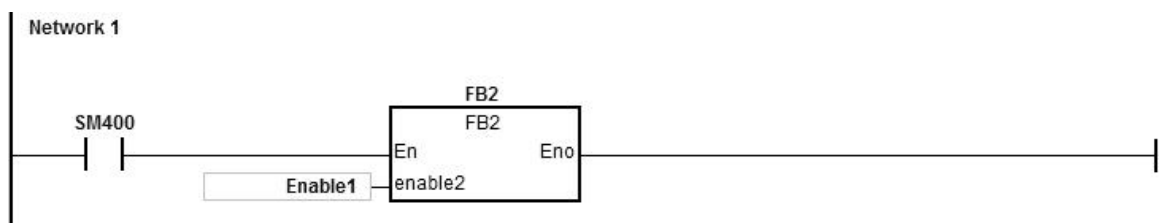
1. Коли M0 увімкнено, Y0.0 видає 1000 імпульсів на 1 кГц.
2. Коли імпульсний вихід завершується, SM461 увімкнено та спрацьовує SM470=ON.
3. Імпульсний вихід перезапускається, коли M0 знову змінюється з OFF. на ON
4. Якщо в програмі є більше двох інструкцій високошвидкісного виведення, не використовуйте назву змінної M1 повторно.
5. Рекомендується не використовувати мову ST у програмі, якщо є ймовірність зупинити виведення.

8. Приклад функціонального блоку

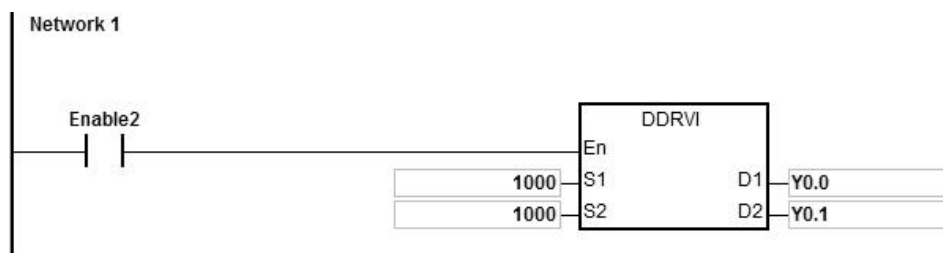
Основна програма



Функціональний блок 1



Функціональний блок 2



Пояснення

Коли M0 увімкнено, Y0.0 видає 10000 імпульсів на 1 кГц. Y0.1 = OFF означає, що вихідний імпульс позитивний.

9. Обов'язково зверніться до інструкції PLSR (API2701), щоб дізнатися більше про зміну цільової частоти під час виведення. Пам'ятайте, що якщо є будь-які зміни на цільовій частоті, зміна може бути здійснена лише після завершення виведення вихідної цільової частоти.

10. Зміна цільової позиції під час виведення:

10.1 Ця функція доступна лише для мікропрограми версії 1.02.30 або новішої. Ви не можете змінити напрямок виведення навіть після зміни цільової позиції. Наприклад, якщо вихідна цільова позиція знаходиться в позитивному напрямку, після зміни цільової позиції вона все ще виводиться в позитивному напрямку.

10.2 Після того як ви встановили прапорець зміни, зміна відбувається під час сканування інструкції. І після того, як його змінено, прапорець зміни зникає. Але якщо цільову позицію встановлено неправильно, прапорець зміни залишається увімкненим.

- 10.3 Зверніться до наступної таблиці для відповідних виходів і змініть прапор SM.

Вихідна точка	Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5
Номер SM	SM478	SM479	SM498	SM499	SM518	SM519
Вихідна точка	Y0.6	Y0.7	Y0.8	Y0.9	Y0.10	Y0.11
Номер SM	SM538	SM539	SM558	SM559	SM578	SM579

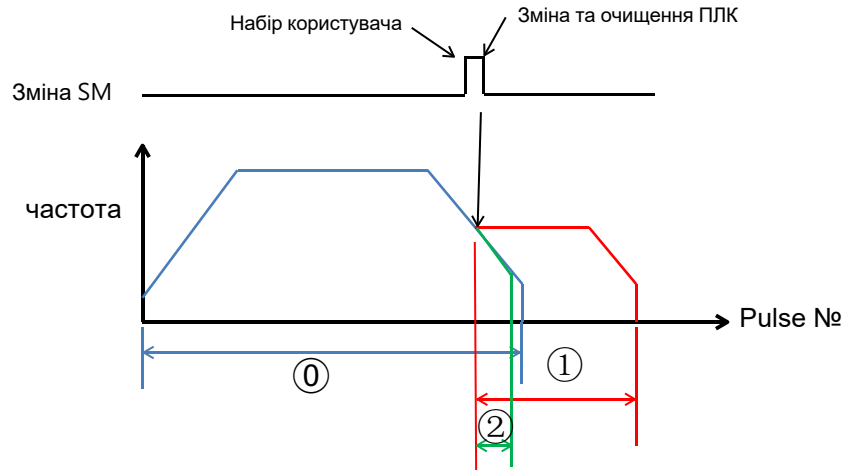
- 10.4 Ця функція зміни цільової позиції доступна для пристроїв DDRVI та DDRVA. Коли інструкція

Використовується DDRVI і встановлено позначку зміни, зміни буде виконано. Навіть якщо вихідний номер дорівнює 0, операція буде виконана та припинить виведення. Коли використовується інструкція DDRVA, якщо змінена цільова позиція така ж, як і попередня, ця зміна не буде виконана.

- 10.5 При зміні позиції цілі під час виведення можуть виникнути три ситуації.

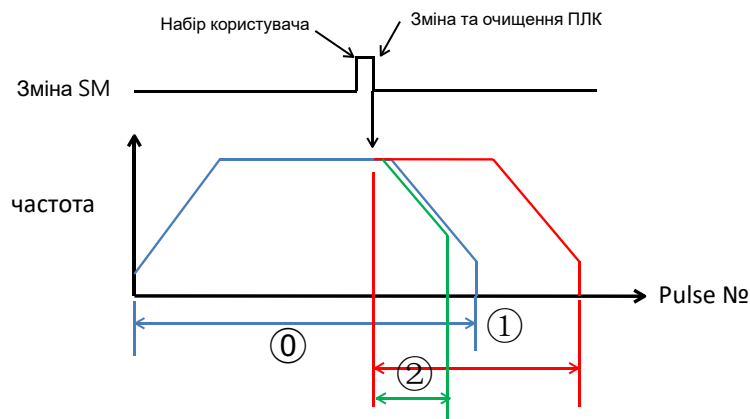
Ситуація А: зміна часу відбувається під час виведення в зоні зниження.

- ① вихідна цільова позиція під час запуску (синя)
- ① нова кількість вихідних імпульсів >= залишилася кількість вихідних імпульсів, його вихідна крива (червоний).
- ② нова кількість вихідних імпульсів < кількість вихідних імпульсів, що залишилася, його вихідна крива (зелений).



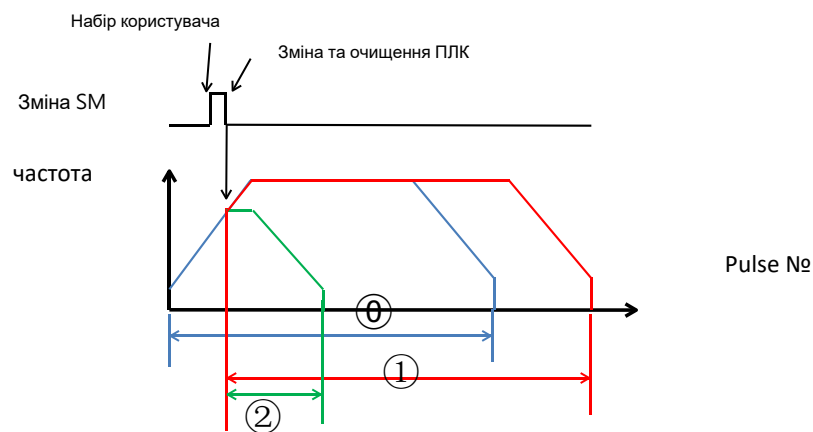
Ситуація В: зміна синхронізації відбувається під час виведення в цільовій частотній області.

- ① вихідна цільова позиція під час початкового старту (синій)
- ① нова кількість вихідних імпульсів $\geq$ кількість вихідних імпульсів у зоні зниження, його вихідна крива (червона).
- ② нова кількість вихідних імпульсів $<$ кількість вихідних імпульсів у зоні спаду, його вихідна крива (зелений)



Ситуація С: зміна часу відбувається під час виведення в зоні наростання.

- ① вихідна цільова позиція під час початкового старту (синій)
- ① нова кількість вихідних імпульсів $\geq$ кількість вихідних імпульсів у зоні зниження, його вихідна крива (червона).
- ② нова кількість вихідних імпульсів $<$ кількість вихідних імпульсів у зоні спаду, його вихідна крива (зелений)



API	Інструкція				Операнд								опис			
2707	D	DRVA			S_1, S_2, D_1, D_2								Абсолютний контроль позиції			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
D_1		○														
D_2		○	○													

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
D_1	●												
D_2	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



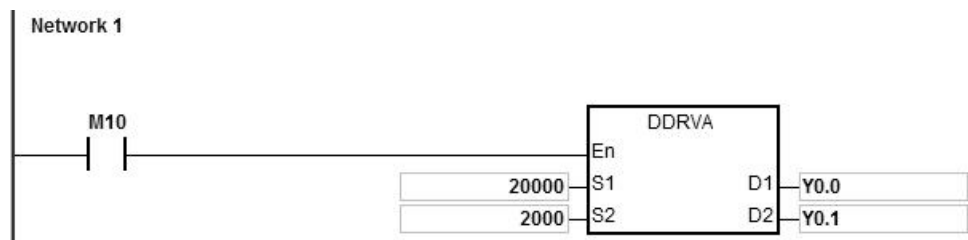
- S_1 : Кількість вихідних імпульсів (абсолютне позиціонування)
- S_2 : Вихідна частота імпульсів
- D_1 : Пристрій виведення імпульсів
- D_2 : Пристрій виведення напрямку імпульсу

Пояснення

- Ця інструкція вказує налаштування вихідного імпульсу для абсолютного позиціонування. S_1 – кількість вихідних імпульсів (абсолютне позиціонування). Діапазон становить від -2 147 483 648 до +2 147 483 647, а знаки +/- вказують на позитивні та негативні напрямки пульсу. Коли інструкція виконується, вона перевіряє поточну позицію за допомогою S_1 . Якщо S_1 дорівнює 0, це вказує на відсутність виведення, а прапор завершення виведення не має значення ON. Якщо 0 є можливим вихідним числом у вашій програмі, пропонується додати більше умов у вашу програму, щоб виключити таку можливість.
- S_2 — частота імпульсного виходу, а діапазон становить від 0 Гц до 4 МГц для моделей з вихідним сигналом лінійного драйвера та від 0 Гц до 200 кГц для моделей з відкритим колектором. Якщо значення в S_2 менше початкової/кінцевої частоти (Гц), встановленої в SR (додаткову інформацію див. у таблиці SR інструкції DRV1), ПЛК працює відповідно до значень, встановлених у SR.
- Абсолютне позиціонування означає, що інструкція видає імпульси для переміщення від поточної позиції до досягнення заданої цільової позиції. Наприклад, кількість вихідних імпульсів у поточній позиції дорівнює 100, а кількість імпульсів у цільовій позиції S_1 встановлено на 1000. Отже, кількість фактичних вихідних імпульсів становить 1000-100=900.
- Зверніться до інструкції DDRVI (API 2706), щоб отримати додаткові пояснення.

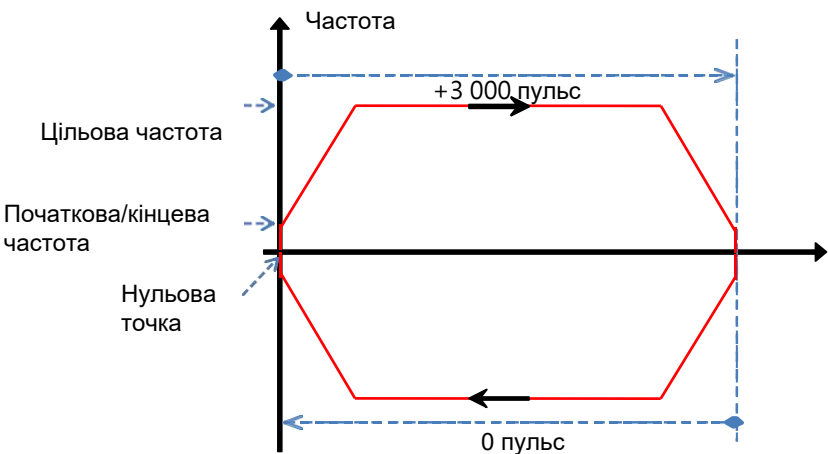
приклад

Якщо значення поточної вихідної позиції SR460 (32-розрядний) становить 100, а M10 увімкнено, тоді інструкція DDRVA змушує Y0.0 виводити імпульси з частотою 2 кГц, доки значення в SR460 не стане 20 000 (абсолютне позиціонування). Y0.1 = OFF означає позитивний напрямок.



Додаткові зауваження

1. На наступному графіку показано абсолютне позиціонування: спосіб визначення відстані від центру (нульової точки).

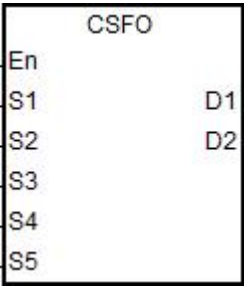


API	Інструкція				Операнд								опис			
2708		CSFO				$S_1, S_2, S_3, S_4, S_5, D_1, D_2$								Швидкість улову та пропорційний вихід		
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	○															
S_2								●								
S_3								●								
S_4								●								
S_5								●								
D_1		○														
D_2								●								

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1	●												
S_2			●				●						
S_3			●				●						
S_4		●											
S_5		●											
D_1	●												
D_2			●				●						

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	-

СИМВОЛ



- S_1 : Пристрій введення сигналу
- S_2 : Кількість вхідних імпульсів
- S_3 : Вхідна частота імпульсів
- S_4 : Знаменник (вихідна частота) для пропорції вхідної частоти та вихідної частоти
- S_5 : Чисельник (вхідна частота) для пропорції вхідної частоти та вихідної частоти
- D_1 : Пристрій виведення імпульсів
- D_2 : Вихідна частота імпульсів

Пояснення

1. Ця інструкція визначає швидкість лову та пропорційний вихід. S_1 може вказувати лише X0.0, X0.2, X0.4, X0.6, X0.8 і X0.10 як вхідні точки, а операнд займає дві послідовні вхідні точки. Ви не можете виконати інструкція, якщо вхідними точками є точки, вказані для S_1 вище. Після вибору вхідних точок автоматично вказується високошвидкісний лічильник. Якщо є інструкція DCNT (API 1004) з таким же високошвидкісним лічильником у

У програмі ПЛК спочатку виконує інструкцію, яка першою запускає лічильник. Точки введення та відповідні високошвидкісні лічильники показані в наступній таблиці.

Номер групи	1	2	3	4	5	6
S₁ +0 вхідна точка (Фаза А)	X0.0	X0.2	X0.4	X0.6	X0.8	X0.10
S₁ +1 вхідна точка (Фаза В)	X0.1	X0.3	X0.5	X0.7	X0.9	X0.11
Швидкісний лічильник номер	HC202	HC206	HC210	HC214	HC218	HC222
Прапор для руху заднім ходом напрямок введення	SM270	SM271	SM272	SM273	SM274	SM275

2. Якщо високошвидкісні лічильники для інструкції можуть використовувати лише режим введення фази А/В, встановіть прапор для реверсування напрямку введення на ON, коли MPG підключений, але ще не обертався, а точка введення ПЛК увімкнена. Встановіть для функції визначення стану циклу виконання ПЛК значення OFF.

3. **S₂** – кількість вхідних імпульсів. Використовуйте 32-розрядну змінну, щоб оголосити параметр.

4. **S₃** – частота вхідних імпульсів. Використовуйте 32-розрядну змінну, щоб оголосити параметр з одиницею 1 Гц.

5. **S₄** – це знаменник (вихідна частота) пропорції вхідної та вихідної частоти. **S₅** – це чисельник (вхідна частота) для співвідношення вхідної та вихідної частоти. Діапазон **S₄** і **S₅** становить від 1 до 255. Якщо значення налаштування перевищує діапазон, інструкція обробляється як максимальне або мінімальне значення. Наприклад, якщо вхідна частота: вихідна частота = 5:3, вихідна частота знаменника дорівнює К3, а вхідна частота чисельника дорівнює К5. Якщо вхідна частота: вихідна частота=1:2, знаменник вихідної частоти дорівнює К2, а вхідна частота чисельника дорівнює К1.

6. **D₁** може вказати лише Y0.0, Y0.2, Y0.4, Y0.6, Y0.8 та Y0.10 як вихідні точки та займає дві послідовні вихідні точки.

Точки виходу та відповідний вихідний режим SR наведені в наступній таблиці.

Номер вихідної осі	1	2	3	4	5	6
D₁ +0 вихідна точка	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
D₁ +1 вихідна точка	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Режим виведення	SR462	SR482	SR502	SR522	SR542	SR562

7. **D₂** – частота вихідних імпульсів. Використовуйте 32-розрядну змінну, щоб оголосити параметр з одиницею 1 Гц.

8. Кількість разів використання цієї інструкції не обмежена; але високошвидкісні точки введення та виведення не можуть використовуватися іншими інструкціями, коли ця інструкція виконується. Інакше інструкція не може бути виконана.
9. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.

Примітки

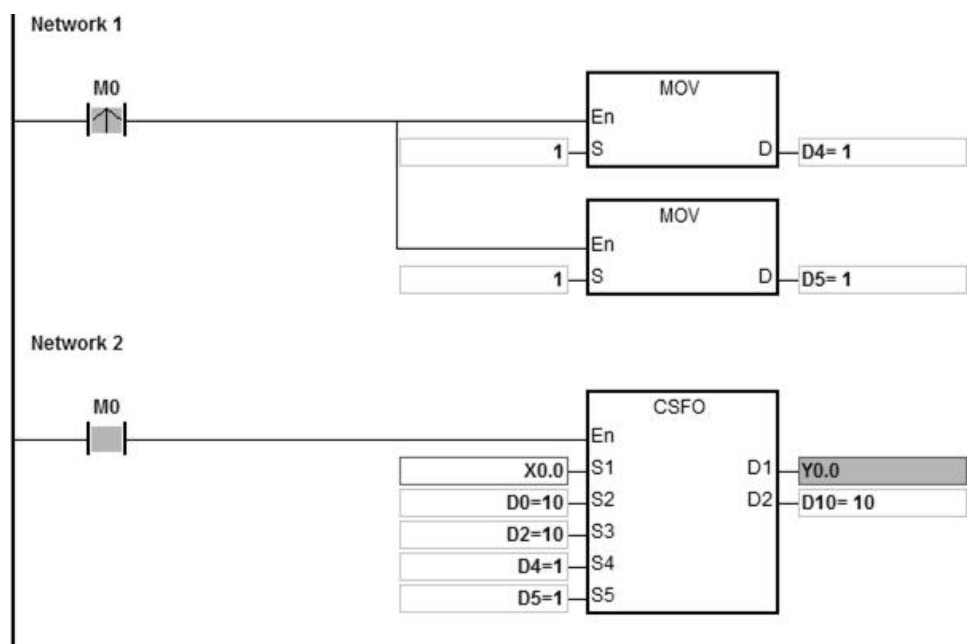
1. ПЛК обчислює частоту вхідного імпульсу на основі ширини вхідного імпульсу (ON) у позитивному півперіоді. Якщо пропорція ширини імпульсу для ON:широти імпульсу для OFF не становить 1:1, ПЛК приймає ширину ON як стандарт для перетворення за замовчуванням.
2. Вхідний імпульс=ON означає, що світлодіодний індикатор вхідної точки горить. Якщо використовується MPG, але він ще не обертався, ви можете перевірити, чи світлодіод точки введення OFF та прапорець для зміни напрямку введення активований.
3. Для ПЛК із програмним забезпеченням версії 1.04.70 (версія 1.04.70 включно) або пізнішої мінімальна частота плюса для входу та виходу становить 50 Гц. Якщо фактичний вхідний імпульс становить 10 Гц, а значення в S_3 показує 10 Гц. Але оскільки мінімальна вихідна частота імпульсів становить 50 Гц, інструкція використовує 50 Гц замість 10 Гц для обчислення. Для співвідношення 1:2 після перетворення вихідний імпульс становить 100 Гц. Для співвідношення 2:1
4. після перетворення вихідний імпульс становить 50 Гц замість 25 або 5 Гц.

Ви можете змінювати значення в S_4 або S_5 під час виконання цієї інструкції. Але цю інструкцію потрібно виконати повторно актуалізувати цю модифікацію. Інакше значення в S_2 буде недостатньо точним, щоб використовувати його для обчислення поточної вихідної позиції SR.

5. Після перетворення, якщо частота вихідного імпульсу перевищує максимальну межу, код помилки записується як 16#2030 у Файл журналу помилок у ISPSOFT і SM29 увімкнено. SR29 - номер осі. Ця функція доступна для ПЛК із FW V1.06.20 (V1.06.20 включено) або новішої версії.
6. Якщо вхідний імпульс становить 0 Гц і триває більше 2 секунд, ПЛК автоматично припиняє виведення, а код помилки записується як 16#2031 у файлі журналу помилок у ISPSOFT, а SM29 увімкнено. SR29 - номер осі. Ця функція доступна для ПЛК із FW V1.06.20 (V1.06.20 включено) або новішої версії.

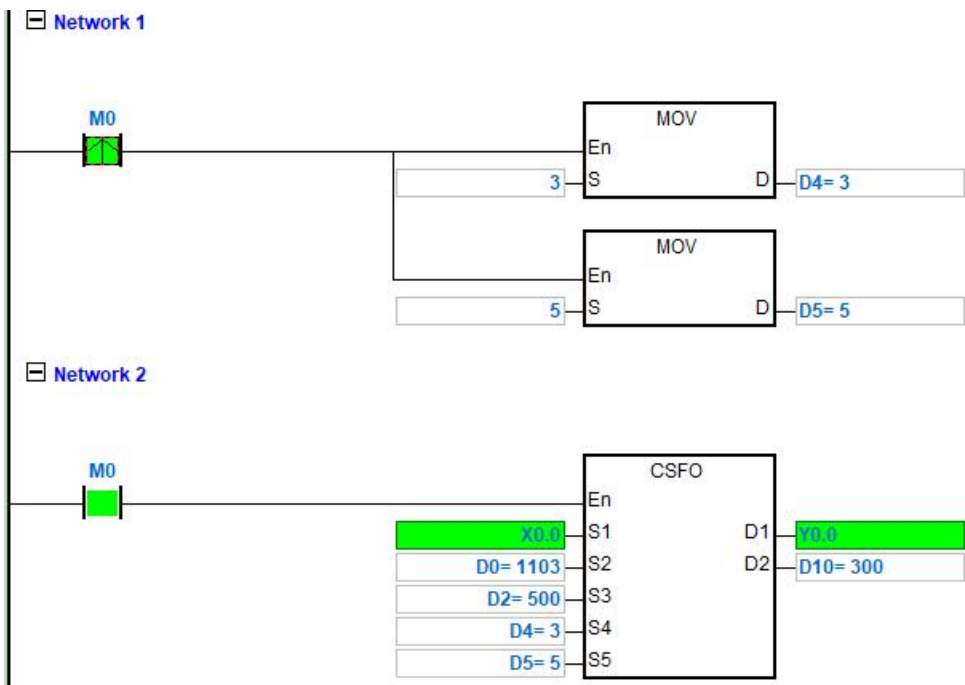
Приклад 1

Вхідні імпульси X0.0/X0.1 для визначення MPG. Коли M0 увімкнено, значення параметрів S_4 і S_5 обидва дорівнюють 1 ($D4=1$ і $D5=1$). Коли вхідна частота становить 10 Гц ($D2, D3=10$) і кількість імпульсів становить 10 ($D0, D1=10$), вихідна вісь ($Y0.0/Y0.1$) видає 10 імпульсів ($SR460, SR461=10$).) при 10 Гц ($D10, D11=10$).



Приклад 2

Коли M0 увімкнено, значення параметрів S4 і S5 становлять 3 і 5 (D4=3, D5=5) відповідно. Коли вхідна частота MPG становить 500 Гц (D2, D3=500), а кількість імпульсів становить 2500 (D0, D1=2500), вихідна вісь (Y0.0/Y0.1) видає 1500 імпульсів (SR460, SR461=1500) при 300 Гц (D10, D11=300).



API	Інструкція				Операнд								опис				
2709	D	DRVМ			S ₁ , S ₂ , S ₃ , S ₄ , S ₅ , S ₆ , D ₁ , D ₂								Розташування вирівнювання позначок				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S ₁								●	●				○	○			
S ₂								●	●				○	○			
S ₃	○		○														
S ₄								●	●				○	○			
S ₅								●	●				○	○			
S ₆								●	●				○	○			
D ₁		○															
D ₂		○	○														

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁			●				●						
S ₂			●				●						
S ₃	●												
S ₄			●				●						
S ₅			●				●						
S ₆			●				●						
D ₁	●												
D ₂	●												

6

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DDRМ	
En	
S1	D1
S2	D2
S3	
S4	
S5	
S6	

- S₁ : Цільова кількість вихідних імпульсів
- S₂ : Цільова вихідна частота
- S₃ : Номер точки входу для отримання зовнішнього сигналу переривання
- S₄ : Кількість вихідних імпульсів у кінці для секції передньої маски
- S₅ : Кількість вихідних імпульсів на початку для секції задньої маски
- S₆ : Кількість вихідних імпульсів у процесі зниження після отримання сигналу переривання
- D₁ : Пристрій виведення імпульсів
- D₂ : Пристрій виведення імпульсів напрямку

Пояснення

1. Ця інструкція виконує позиціонування позначки. Коли значення S_1 дорівнює 0, це вказує, що на виході є максимальне 32-бітне значення (зі знаком), і на виході не виконується зупинка зниження до отримання сигналу позначки.
Якщо значення S_1 не дорівнює 0 і зовнішнє переривання не відбувається, виводяться цільові вихідні імпульси. Діапазон S_1 становить від -2 147 483 648 до +2 147 483 647, а знак +/- означає позитивний або негативний напрям.
2. Коли значення S_2 менше або дорівнює 0, вихід не вмикається. Якщо вихідна частота перевищує максимальну частоту, ПЛК обробляє її як максимальну вихідну частоту.
3. S_3 може вказати пристрій M або X як точку введення. На сигнал джерела вхідного сигналу впливає цикл сканування, якщо вибрана вхідна точка не знаходиться в X0.0–X0.15 у ПЛК.
4. Ви повинні використовувати інструкцію, комбінуючи номер точки введення та зовнішню програму переривання, щоб досягти вихідного сигналу в режимі реального часу. Для тригера по наростаючому або спадаючому фронту для зовнішнього переривання виберіть тригери в програмі зовнішнього переривання в ISPSOft.
 - A. Використання зовнішнього переривання (наприклад, I0xx, I1xx): коли виникає зовнішнє переривання, після отримання сигналу позначки виконується зупинка з нахилом. Жодна операція не виконується, якщо зовнішнє переривання не ввімкнено.
 - B. Без використання зовнішнього переривання (I0xx, I1xx): коли інструкція виконується, зупинка наростання виконується після отримання сигналу позначки, якщо наростаючий фронт виникає в точці X або пристрої M. На це впливає час сканування.
5. Кількість вихідних імпульсів у секції передньої маски становить від 1 до S_4 . Коли значення налаштування для S_4 дорівнює 0, це означає функцію розділу передньої маски вимкнено. Коли кількість імпульсів, що підлягають маскуванню, перевищує значення S_4 , інструкція бере цільову кількість вихідних імпульсів S_4 як кількість, що підлягає маскуванню. Якщо тригер зовнішнього входу відбувається в межах кількості вихідних імпульсів, які потрібно маскувати, переривання зовнішнього входу автоматично стає недійсним.
6. Кількість вихідних імпульсів у секції задньої маски становить від S_5 до S_1 . Коли $S_5 = 0$ або $S_5 \geq S_1$, функція секції задньої маски вимкнена. Коли зовнішній вхідний тригер виникає в замаскованих секціях, зовнішнє вхідне переривання автоматично стає недійсним. Коли $S_4 > S_5$, це вказує на те, що зовнішнє переривання введення недійсне в процесі виведення.
7. Якщо передня і задня секції маски встановлені і $S_4 < S_5$, дійсне переривання введення відбувається в секції між $S_4 + 1$ і $S_5 - 1$.
8. Для ПЛК із FW V1.04.00 (V1.04.00 включено) або попередніми версіями:

S_6 : кількість вихідних імпульсів при зниженні	опис
Якщо S_6 встановлено на 0 після отримання сигналу позначки	Зупинка на рампі виконується на основі час простою

Якщо $S_6 = -1$ або < 0 після отримання сигналу позначки	вихід негайно припиняється
Якщо значення параметра S_6 недостатньо для досягнення зупинки зниження протягом часу зниження,	Інструкція обмежує цільову частоту і виконує гальмівну зупинку відповідно до встановленого часу гальмування
Якщо S_6 більше, ніж кількість вихідних імпульсів протягом часу зниження,	Інструкція виводить номер зайвого імпульси з тією самою вихідною частотою, коли спрацьовує переривання, а потім виконує процес зменшення

9. Для ПЛК із програмним забезпеченням версії 1.06.00 (версія 1.06.00 включно) або новішої: якщо S_6 встановлено на 0 після отримання сигналу позначки, це означає

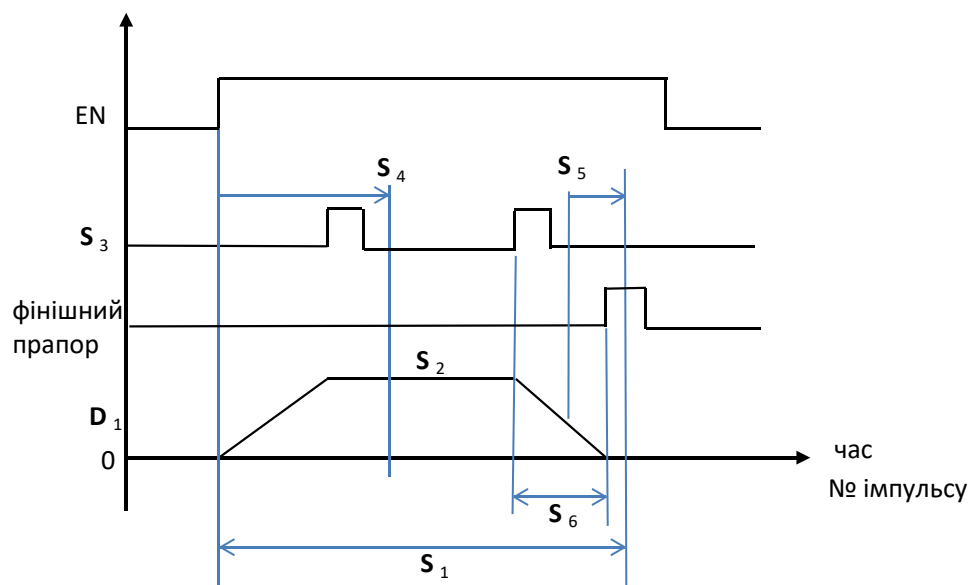
що зупинка при зниженні темпу виконується на основі часу зниження. Якщо S_6 встановлено на -1 після отримання сигналу позначки, це означає, що вихід негайно припиняється. Якщо S_6 встановлено на >0 після отримання сигналу позначки, і якщо значення налаштування S_6 недостатньо для досягнення зупинки зниження протягом часу зниження, інструкція обмежує цільову частоту та виконує зупинку вниз відповідно до встановленого часу зниження. Якщо S_6 більше, ніж кількість імпульсів, виведених протягом часу зниження, інструкція виводить кількість надлишкових імпульсів з тією ж вихідною частотою, коли спрацьовує переривання, а потім виконує процес зниження.

S_6 : кількість вихідних імпульсів при зниженні	опис
Якщо $S_6 = > 0$ після отримання сигналу позначки	Інструкція обмежує цільову частоту та виконує зупинку зниження відповідно до встановленого часу зниження
Якщо значення параметра S_6 недостатньо для досягнення зупинки зниження протягом часу зниження,	
Якщо $S_6 = > 0$ після отримання сигналу позначки	Інструкція виводить номер зайвого імпульси з тією самою вихідною частотою, коли спрацьовує переривання, а потім виконує процес зменшення
Якщо S_6 більше, ніж кількість вихідних імпульсів протягом часу зниження,	
Якщо S_6 встановлено на 0 після отримання сигналу позначки	Зупинка загальмування виконується на основі часу загасання
Якщо $S_6 = -1$ після отримання сигналу позначки	Вивід негайно припиняється
Якщо $S_6 = < -1$ після отримання сигналу позначки	Інструкція НЕ обмежує цільову частоту, і після виведення кількості вихідних імпульсів вихід негайно припиняється
Якщо значення параметра S_6 недостатньо для досягнення зупинки зниження протягом часу зниження,	
Якщо $S_6 = < -1$ після отримання сигналу позначки	Інструкція виводить номер зайвого імпульси з тією ж вихідною частотою, коли

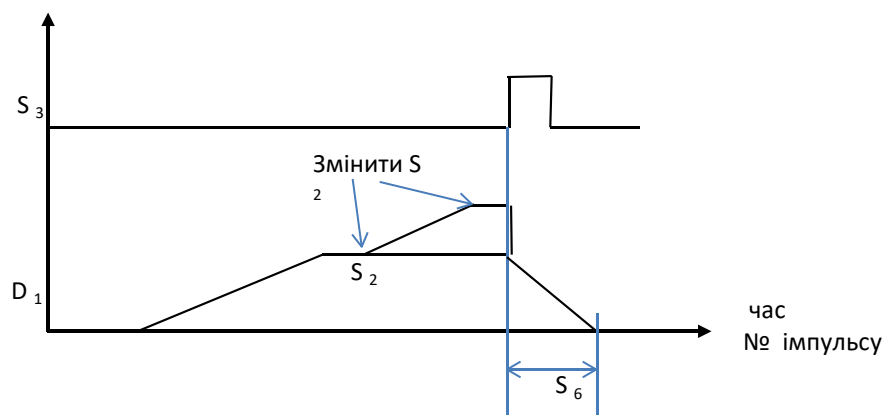
Якщо S_6 більше, ніж кількість вихідних імпульсів в межах час зниження,

Переривання запускається, а потім виконується вниз процес

- Додано нові SR652, SR653 для відображення кількості імпульсів у зоні підвищення та нові SR654, SR655 для відображення кількості імпульсів у зоні зниження.
10. Якщо вихідний сигнал увійшов у процес зниження, коли виникає тригер переривання зовнішнього введення, інструкція завершує виведення кількості імпульсів, визначеної S_6 .
11. Перегляньте наступну часову діаграму про вихід, тригер переривання, маску та відповідні прапори. Щоб дізнатися прапор завершення, зверніться до прапора завершення SM для осей у інструкції DDRVI (API 2706).



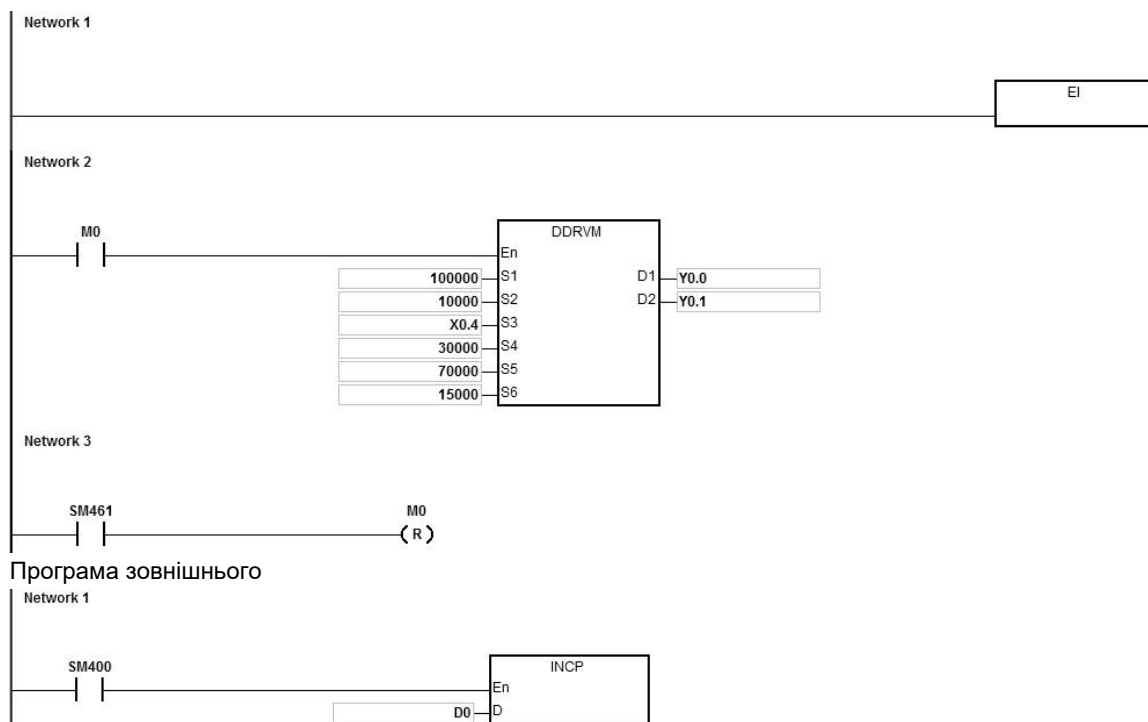
12. Цільова частота S_2 інструкції може бути змінена в процесі виведення імпульсу. З числа вихідних імпульсів у ділянці спаду встановлено, коли інструкція виконується, часова діаграма для зміни частоти показана наступною пунктирною лінією.



13. Зверніться до інструкції DDRVI (API 2706) щодо вибору пристроїв імпульсного виводу для D_1 і D_2 . Вихідні точки для D_1 можуть мати лише парне число, наприклад Y0.0, Y0.2 або Y0.10. Якщо вихідні точки не є рекомендованими, цикл сканування впливає на напрямок D_2 .

14. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.

Приклад 1



Примітки

1. Коли M0 змінюється з OFF на ON, Y0.0 починає видавати імпульси. Після виведення понад 30 000 імпульсів, зовнішнє переривання виявлено на X0.4. Значення в D0 збільшується на 1, а імпульсний вихід припиняється після 15 000 імпульсів ($S_6 = 15\,000$). Якщо переривання не відбувається, вихід не припиняється, доки не буде виведено 100 000 імпульсів.
2. Якщо кількість вихідних імпульсів становить від 1 до 30 000, вони знаходяться в секції передньої маски. Відбувається зовнішнє переривання на X0.4 на даний момент, і інструкція не виконує операцію припинення.
3. Якщо кількість вихідних імпульсів становить 70 000–100 000, вони знаходяться в секції задньої маски. Зовнішнє переривання відбувається в X0.4 в даний момент, і інструкція все ще не виконує операцію припинення.
4. Коли імпульсний вихід завершується, SM461 увімкнено, а M0 скидається.

Додаткові зауваження

1. Функція позначки вирівнювання та функція маски:

(1) Коли DDRVM виконується, починається імпульсний вихід.

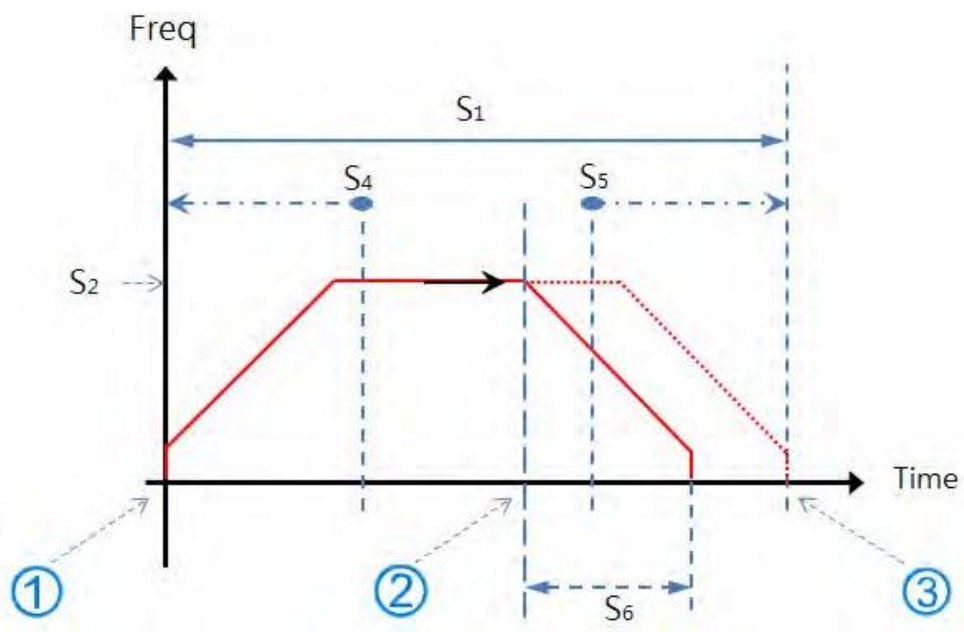
(2) Коли відбувається переривання для вирівнювання позначки, починається процес зниження, і вихід припиняється після кількості імпульсів, визначених S_6 .

(3) Коли переривання для вирівнювання позначки не відбувається або не має ефекту, DDRVM припиняє виводити імпульси після цільової кількості імпульсів, визначеної S_1 .

S_4 : кількість вихідних імпульсів у секції передньої маски. Якщо переривання відбувається в цьому розділі, переривання є неефективним для вирівнювання позначки.

S_5 : кількість вихідних імпульсів у секції задньої маски. Якщо переривання відбувається в цьому розділі, переривання є неефективним для вирівнювання позначки.

S_2 : Цільова вихідна частота

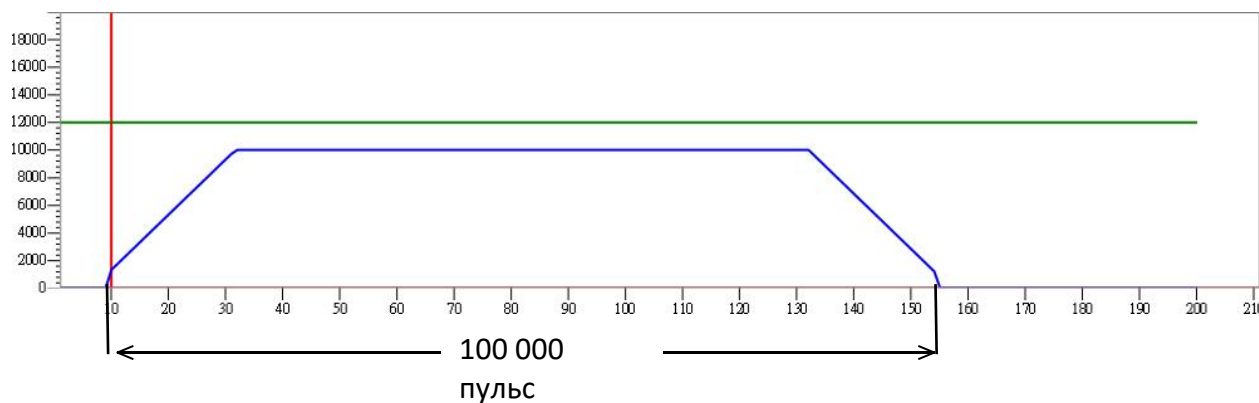


2. Функцію вирівнювання позначок увімкнено в розділах наростання, цільової частоти та спаду. (Горизонтальна координата (вісь x) представляє ЧАС, а вертикальна координата (вісь y) представляє ПУЛЬС.)

Якщо функцію вирівнювання позначок не ввімкнено, установіть $S_1 = 100\ 000$; вихід припиняється після 100 000 імпульсів.

● Зелена лінія: сигнал вирівнювання позначки

● Синя лінія: фактична вихідна швидкість



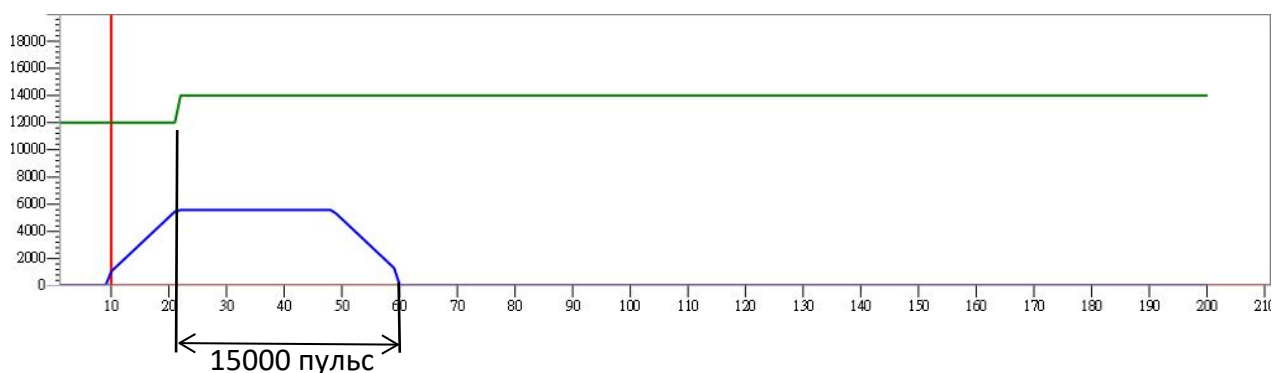
Випадок 1: коли вирівнювання позначок відбувається на ділянці наростання

● Зелена лінія: сигнал вирівнювання позначки

● Синя лінія: фактична вихідна швидкість

Операція вирівнювання позначки (зелена лінія: OFF. -> ON.) починається, коли кількість імпульсів становить 3000.

Цільова частота 10 кГц не може бути досягнута, навіть якщо S_6 становить 15 000.

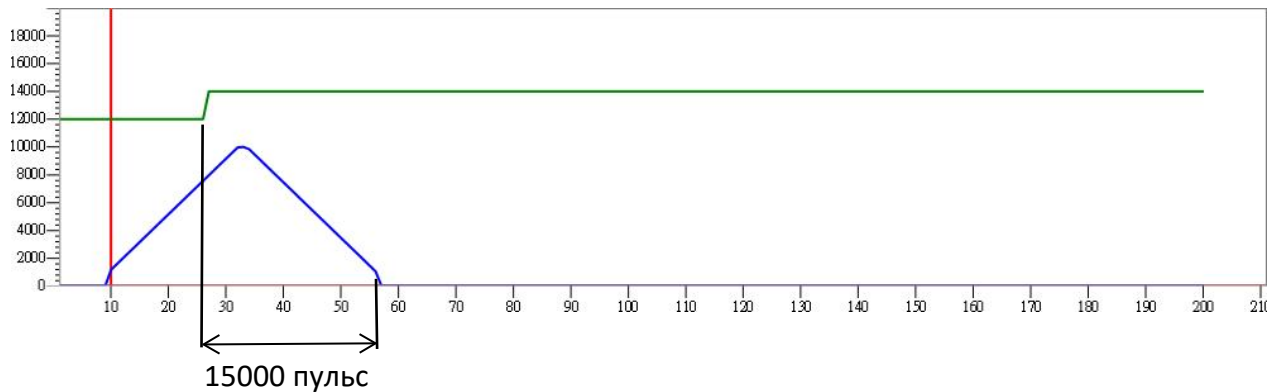


Випадок 2: коли вирівнювання позначок відбувається на ділянці наростання

- Зелена лінія: сигнал вирівнювання позначки
- Синя лінія: фактична вихідна швидкість

Операція вирівнювання позначки (зелена лінія: OFF. -> ON.) починається, коли кількість імпульсів становить 6000.

Цільова частота 10 кГц може бути досягнута, якщо S_6 становить 15 000.

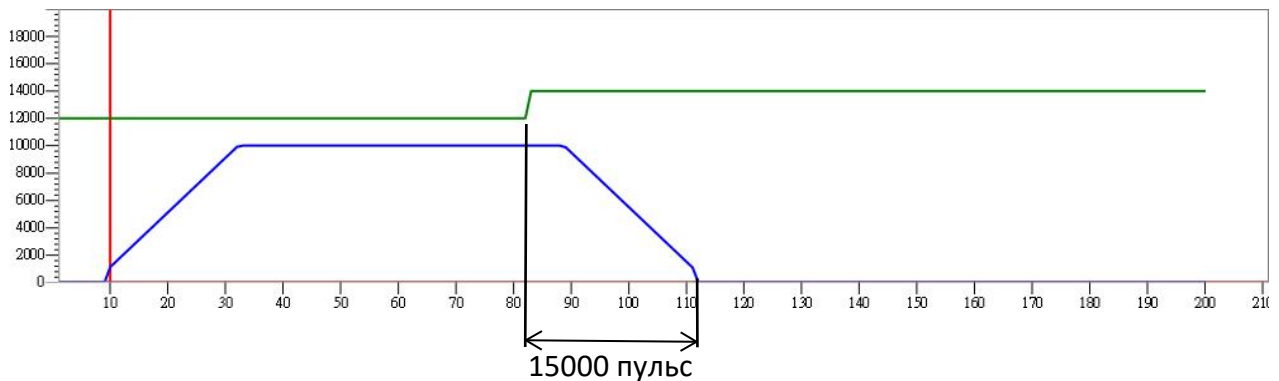


Коли вирівнювання позначок відбувається в ділянці цільової частоти

- Зелена лінія: сигнал вирівнювання позначки
- Синя лінія: фактична вихідна швидкість

Операція вирівнювання позначки (зелена лінія: OFF. -> ON.) починається, коли кількість імпульсів становить 50 000. Набір

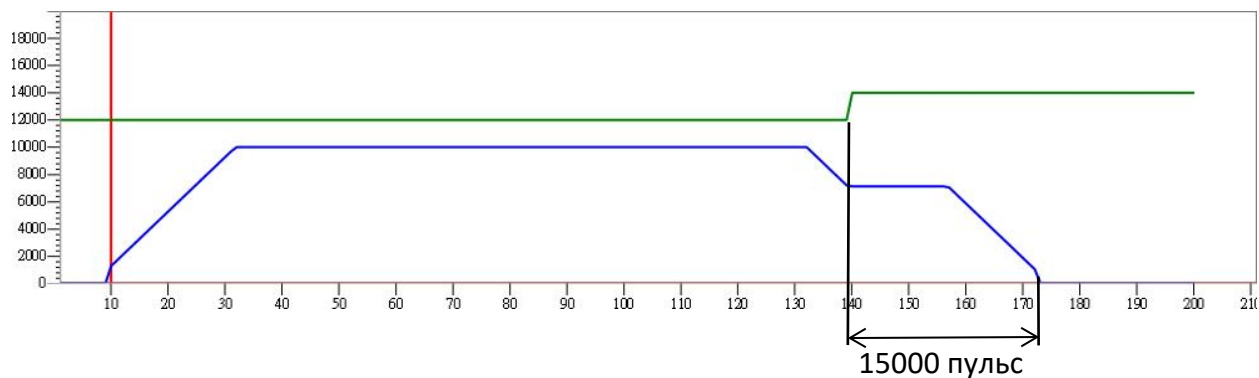
$S_6 = 15\,000$; вихід припиняється після 15 000 імпульсів.



Випадок 1: коли вирівнювання позначки відбувається в секції нахилу

- Зелена лінія: сигнал вирівнювання позначки
- Синя лінія: фактична вихідна швидкість

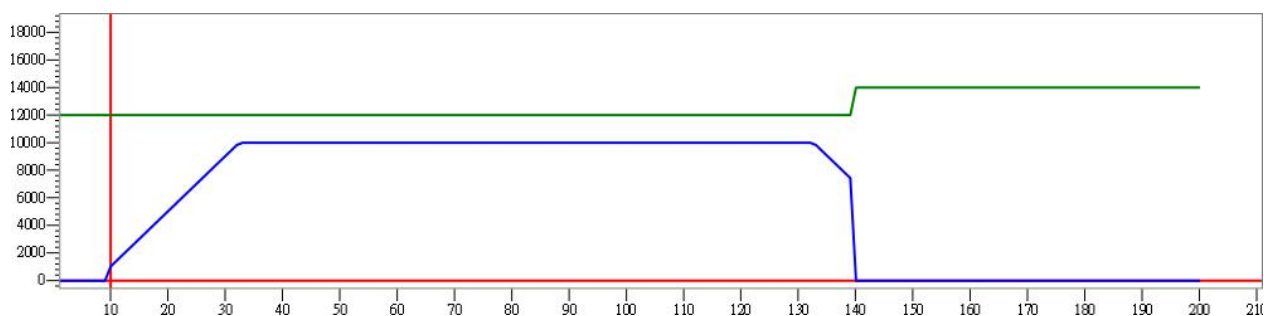
Операція вирівнювання позначки (зелена лінія: OFF. -> ON.) починається, коли кількість імпульсів становить 95 000. Набір $S_6 = 15\ 000$; вихід припиняється після 15 000 імпульсів на поточній швидкості.



Випадок 2: коли вирівнювання позначки відбувається в секції спуску

- Зелена лінія: сигнал вирівнювання позначки
- Синя лінія: фактична вихідна швидкість

Операція вирівнювання позначки (зелена лінія: OFF. -> ON.) починається, коли кількість імпульсів становить 95 000. Після вирівнювання позначки встановіть S_6 на -1, вихід негайно припиняється.



API		Інструкція			Операнд							опис					
2710		D	PPMR			S_1, S_2, S_3, D_1, D_2							2-осьовий синхронізований рух від точки до точки з відносною координатою				

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		O		O	O		
S_2							●	●	●		O		O	O		
S_3							●	●	●		O		O	O		
D_1		O														
D_2		O														

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
D_1	●												
D_2	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DPPMR	
En	
S1	D1
S2	D2
S3	

- S_1 : Кількість вихідних імпульсів для осі X
- S_2 : Кількість вихідних імпульсів для осі Y
- S_3 : Максимальна вихідна частота точка-точка
- D_1 : Пристрій виведення імпульсів по осі X
- D_2 : Пристрій виведення імпульсів для осі Y

Пояснення

- Ця інструкція виконує двоосьовий синхронізований рух із відносною координатою від точки до точки. S_1 і S_2 визначають кількість вихідних імпульсів (відносно позиціонування) для осей X і Y. Діапазон між: -2,147,483,648 до +2,147,483,647, а знак +/- вказує на позитивний/негативний напрямок.
- S_3 визначає максимальну вихідну частоту «точка-точка». Діапазон становить від 1 Гц до 200 кГц.
- Для мікропрограми V1.08.70 або новішої версії ви можете змінити цільову частоту під час виконання інструкції. Після початкової цільової частоти завершує виведення, можна використовувати нову цільову частоту. І час наростання та зменшення від початкової цільової частоти до нової цільової частоти розраховується відповідно до частоти інтервалу та часу інтервалу з моменту ввімкнення інструкції. Зверніться до інструкції DPLSR, щоб отримати детальну інформацію про фактичний рух у секції спаду, яка знаходиться між новою цільовою частотою та наступною цільовою частотою.

4. D_1 і D_2 є вихідними пристроями для осей X і Y відповідно. Ви можете призначити наступні шість осей для виведення, але ви не можете змінити вихідні точки з фіксованим напрямком. Сигнал напрямку: OFF означає в позитивному напрямку, а ON – у негативному напрямку. Якщо ви не використовуєте попередньо встановлений режим виведення Pulse+direction (за замовчуванням: 0), змініть режим виведення, встановивши SR на 1.

Номер осі	Вісь 1	Вісь 2	Вісь 3	Вісь 4	Вісь 5	Вісь 6
Вихідні точки для D_1 і D_2	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
Вихідна точка з фіксованим напрямком	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Режим виведення	SR462	SR482	SR502	SR522	SR542	SR562

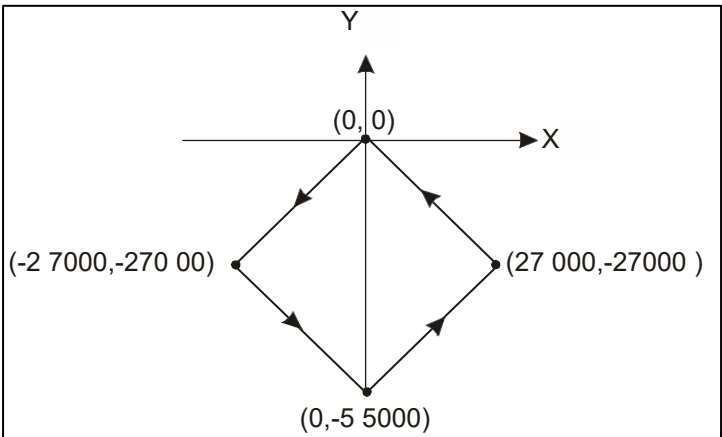
5. ПЛК призначає значення S_3 осі X або Y залежно від того, який виводить найбільшу кількість імпульсів. Якщо кількість вихідних імпульсів для двох осей надмірно різна, тому неможливо розрахувати належну вихідну частоту через кількість вихідних імпульсів, ПЛК автоматично зменшує максимальну частоту від точки до точки без появи будь-яких повідомлень про помилку.
6. Інструкція використовує такі параметри, як початкова та кінцева частоти, а також час розгону та спаду для осі, які виконує інструкція. Замість початково налаштованих параметрів для осей параметри осі X беруться як джерело посилання. Наприклад, коли вісь X вибирає вісь 1, а вісь Y вибирає вісь 3 для вихідного сигналу, параметри осі 1 стають джерелами параметрів для початкової частоти та часу наростання та зниження.
7. У наступній таблиці показано позначки для осей і відповідних SM/SR.

Номер осі	Вісь 1	Вісь 2	Вісь 3	Вісь 4	Вісь 5	Вісь 6
Зайнятий прапор	SM460	SM480	SM500	SM520	SM540	SM560
Прапор завершення	SM461	SM481	SM501	SM521	SM541	SM561
Поточна вихідна позиція	SR460	SR480	SR500	SR520	SR540	SR560
	SR461	SR481	SR501	SR521	SR541	SR561
Початкова/кінцева частота	SR463	SR483	SR503	SR523	SR543	SR563
Час розгону	SR464	SR484	SR504	SR524	SR544	SR564
Час зниження	SR465	SR485	SR505	SR525	SR545	SR565

8. Немає обмежень щодо того, скільки разів ви можете використовувати інструкцію в програмі. Але під час виконання інструкції він не може виконати вихід (якщо використовується вихід по осі Y), доки вісь Y не завершить свій вихід і не буде звільнено.
9. Коли вихід синхронізованого імпульсу по 2 осях завершується, встановлюються відповідні позначки завершення для двох осей. Оскільки неможливо забезпечити щоразу однаковий час для завершення виведення синхронізованого імпульсу по 2 осях, програма повинна перевіряти позначки завершення обох осей перед виконанням наступної програми користувача.

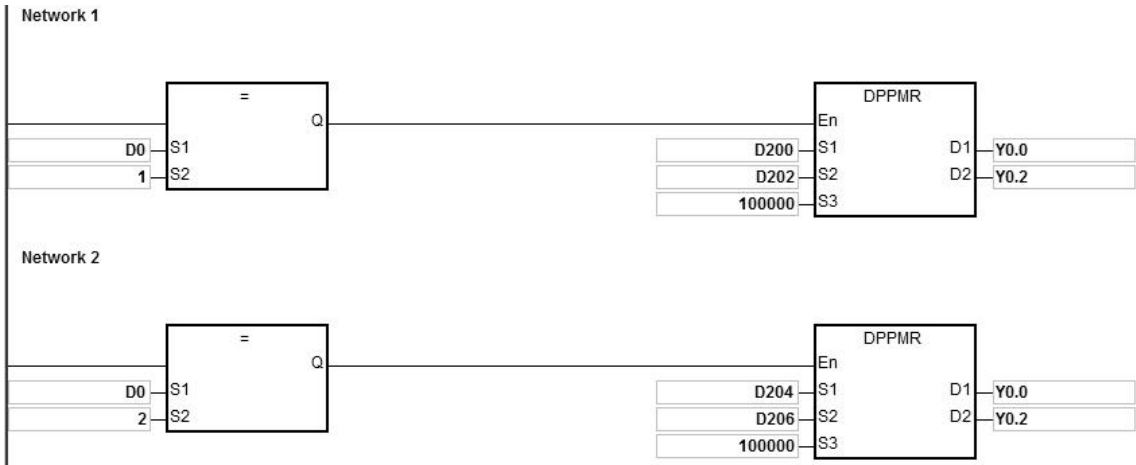
Приклад (Сходові діаграми)

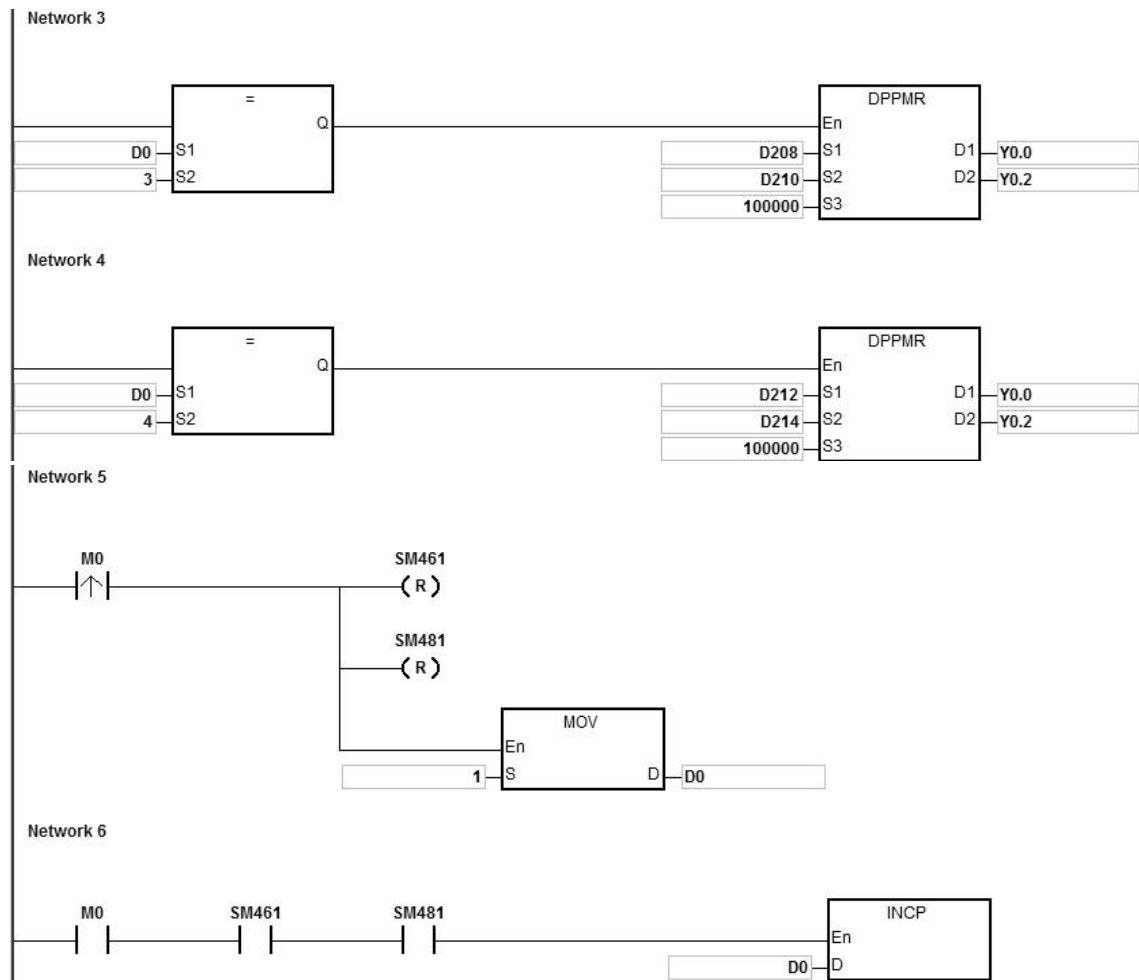
1. Намалюйте ромб, як показано нижче.



2. Кроки:

- Встановіть координати чотирьох точок (0, 0), (-27000, -27000), (0, -55000), (27000, -27000), як на малюнку вище. Обчисліть відносні координати чотирьох точок і отримайте (-27000, -27000), (27000, -28000), (27000, 28000) і (-27000, 27000). Розмістіть їх у 32-розрядних регістрах (D200, D202), (D204, D206), (D208, D210), (D212, D214).
- ЗАПУСТИТИ програму в ПЛК. Встановіть M0 на ON, щоб розпочати креслення 2-осьової лінії.





6

3. Операція:

Коли ПЛК запускає програму і M0 увімкнено, ПЛК починає перший рух від точки до точки на частоті 100 кГц. D0 збільшується на 1 кожного разу, коли рух «точка-точка» завершується, і другий рух «точка-точка» починає виконуватися автоматично. Шаблон операції повторюється, доки не завершиться четвертий рух від точки до точки.

Приклад 2 (програма ST)

```

0001 IF M0 THEN
0002   DPPMR(1000,1000,1000,Y0.0,Y0.2);
0003   M1 := TRUE;
0004 ELSIF M1 AND SM461 AND SM481 THEN
0005   SM470 := TRUE;
0006   SM490 := TRUE;
0007   M1 := FALSE;
0008 END_IF;

```

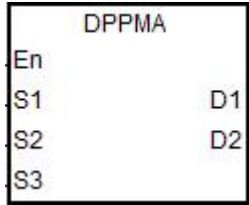
Пояснення

1. Коли M0 увімкнено, Y0.0 і Y0.2 видають 1000 імпульсів на 1 кГц одночасно. Коли M1 увімкнено, це означає, що Y0.0 і Y0.2 використовуються для високошвидкісного імпульсного виведення.
2. Коли вихід імпульсу Y0.0 завершено, SM461 увімкнено. Коли вихід імпульсу Y0.2 завершено, SM481 увімкнено. І якщо M1 також увімкнено, це запускає SM470 (автоматичне скидання завершення виведення для Y0.0) і SM490 (автоматичне скидання завершення виведення для Y0.2) у стані ON. Y0.0 і Y0.2 вільні та готові до повторного використання, а M1 скидається на OFF.
3. Коли M0 перемикається з OFF. на ON., імпульсний вихід перезапускається.
4. Якщо в програмі є більше двох інструкцій високошвидкісного виведення, не використовуйте назву змінної M1 повторно.
5. Рекомендується не використовувати мову ST у програмі, якщо є ймовірність зупинити виведення.

API		Інструкція			Операнд							опис						
2711		D	PPMA			S ₁ , S ₂ , S ₃ , D ₁ , D ₂							2-осьовий абсолютний координатний синхронізований рух від точки до точки					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S ₁							●	●	●		○		○	○				
S ₂							●	●	●		○		○	○				
S ₃							●	●	●		○		○	○				
D ₁		○																
D ₂		○																
Тип дані	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING					
S ₁			●				●											
S ₂			●				●											
S ₃			●				●											
D ₁	●																	
D ₂	●																	

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



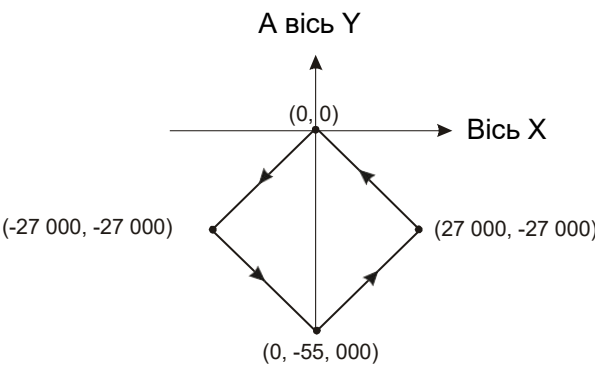
- S_1 : Кількість вихідних імпульсів для осі X
- S_2 : Кількість вихідних імпульсів для осі Y
- S_3 : Максимальна вихідна частота точка-точка
- D_1 : Пристрій виведення імпульсів по осі X
- D_2 : Пристрій виведення імпульсів для осі Y

Пояснення

- Ця інструкція виконує двоосьовий синхронізований рух від точки до точки з абсолютними координатами. **S1** і **S2** визначають кількість вихідних імпульсів (абсолютне позиціонування) для осей X і Y.
- Для отримання додаткової інформації зверніться до інструкції DPPMR (API 2710).

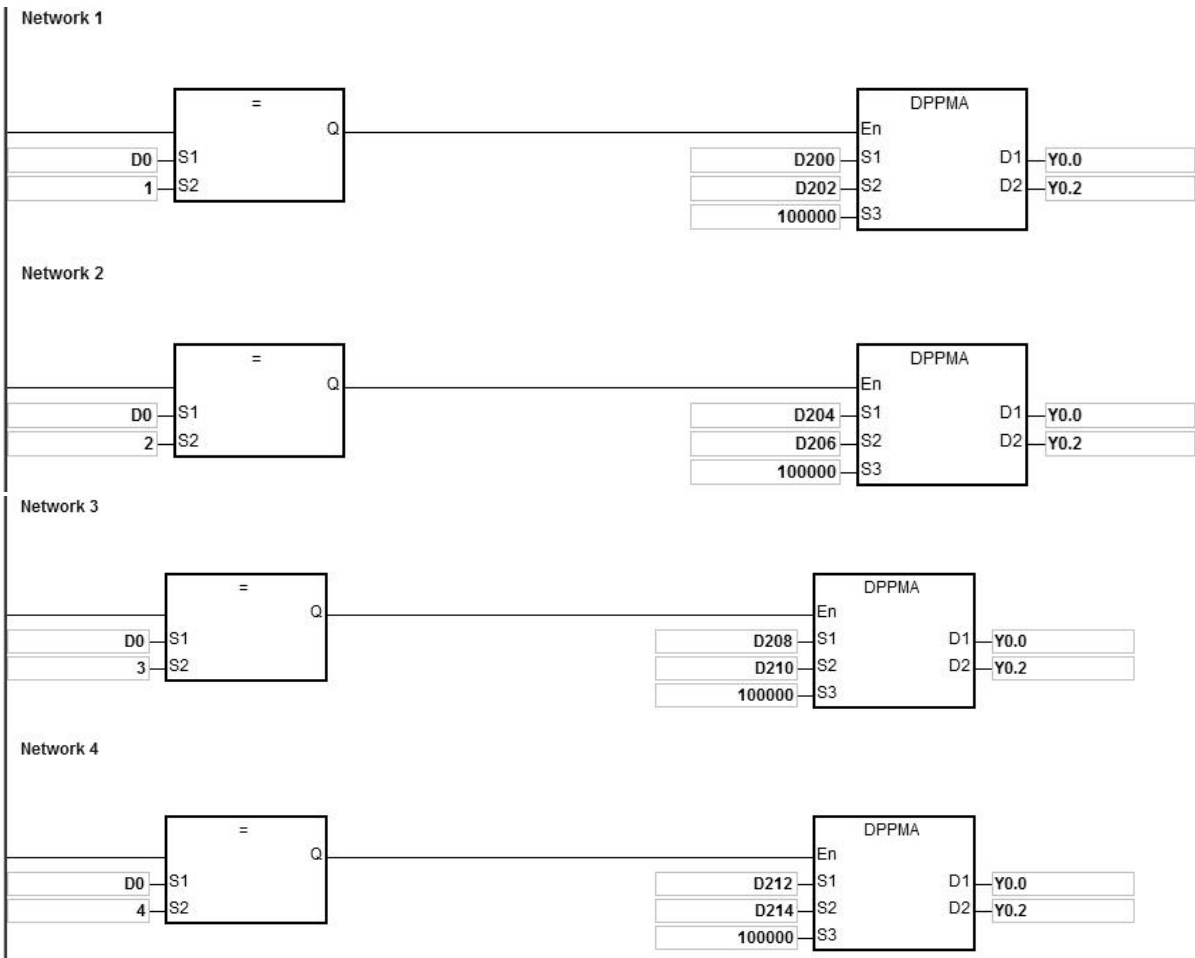
приклад

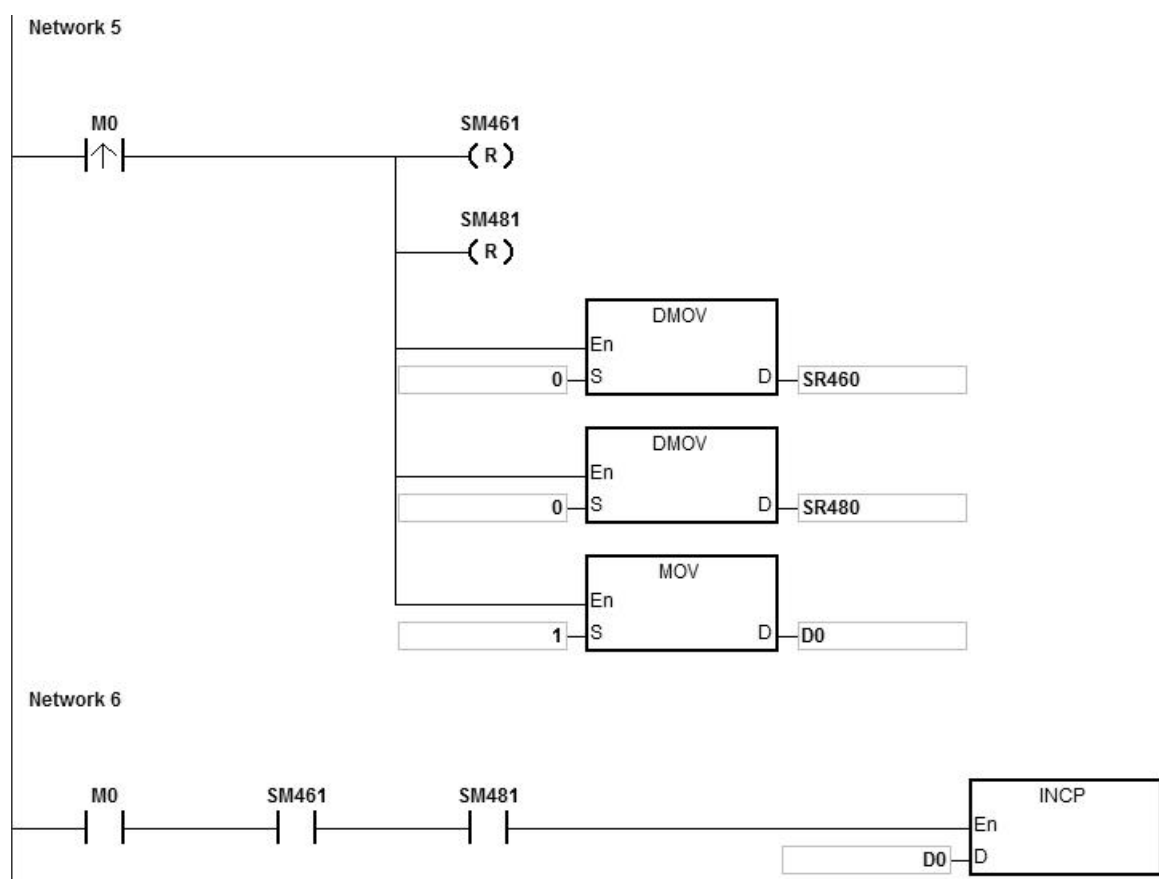
1. Намалуйте ромб, як на малюнку нижче.



2. Кроки:

- Встановіть чотири координати (-27000, -27000), (0, -55000), (27000, -27000) і (0, 0), як показано на малюнку вище. Розмістіть їх у 32-розрядних регістрах (D200, D202), (D204, D206), (D208, D210), (D212, D214).
- ЗАПУСТИТИ програму в ПЛК. Встановіть M0 на ON, щоб розпочати креслення 2-осьової лінії.





3. Операція:

6

Коли ПЛК запускає програму і M0 увімкнено, ПЛК починає перший рух від точки до точки на частоті 100 кГц. D0 збільшується на 1

кожного разу, коли рух «точка-точка» завершується, а другий рух «точка-точка» починає виконуватися автоматично. Шаблон операції повторюється, доки не завершиться четвертий рух від точки до точки.

API	Інструкція				Операнд								опис					
2712	D	CICR			S ₁ , S ₂ , S ₃ , S ₄ , S ₅ , D ₁ , D ₂								2-осьова дугова інтерполяція відносного положення за годинниковою стрілкою					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S ₁							●	●	●		O		O	O				
S ₂							●	●	●		O		O	O				
S ₃							●	●	●		O		O	O		●		
S ₄							●	●	●		O		O	O				
S ₅							●	●	●		O		O	O				
D ₁		O																
D ₂		O																

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁			●				●						
S ₂			●				●						
S ₃			●				●		●				
S ₄			●				●						
S ₅			●				●						
D ₁	●												
D ₂	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DCICR	
En	
S1	D1
S2	D2
S3	
S4	
S5	

- S_1 : Цільова координата осі X (відносно розташування)
- S_2 : Цільова координата осі Y (відносно розташування)
- S_3 : Зміщення центру або центрального кута
- S_4 : Цільова опорна частота
- S_5 : Вибір функції
- D_1 : Пристрій виведення імпульсів по осі X
- D_2 : Пристрій виведення імпульсів для осі Y

Пояснення

1. S_1, S_2 і S_3 відповідно позначають цільову координату осі X, цільову координату осі Y (відносно положення) і зміщення центру або центрального кута кола. Зверніться до наведеної нижче операції дуги за годинниковою стрілкою, щоб дізнатися більше.
2. D_1 і D_2 - пристрої виведення імпульсів для осей X і Y відповідно. Зверніться до інструкції DPPMR (API 2710) щодо вибору вихідних точок і режимів виведення для осей.

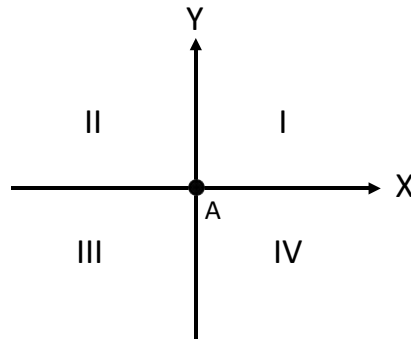
3. S_4 встановлює цільову частоту для опорної частоти. Цільова опорна частота використовується для попереднього розрахунку, коли ПЛК планує шлях руху дуги після виконання інструкції. Але якщо розрахунковий процес розрахунку не може досягти шляху проходження дуги, вихідна частота автоматично зменшується для виконання функції синхронізованого малювання дуги.
4. S_5 – значення налаштування для вибору функції. Дивіться описи нижче.

Значення налаштування в S_5	S_3 : Зміщення центру або центрального кута	опис
0	Зсув центру (32-розрядний тип цілого числа)	Для роздільної здатності дуги використовується 10° дуга як основний кут для руху.
1	Зсув центру (32-розрядний тип цілого числа)	Для роздільної здатності дуги використовується 5° дуга як основний кут для руху.
2	Зсув центру (32-розрядний тип цілого числа)	Для роздільної здатності дуги використовується 1° дуга як основний кут для руху. (FW V1.04.60 або пізнішої версії)
10	Зміщення центрального кута (тип з плаваючою комою)	Для роздільної здатності дуги використовується 10° дуга як основний кут для руху. (FW V1.04.40 або пізнішої версії)
11	Зміщення центрального кута (тип з плаваючою комою)	Для роздільної здатності дуги використовується 5° дуга як основний кут для руху. (FW V1.04.40 або пізнішої версії)
12	Зміщення центрального кута (тип з плаваючою комою)	Для роздільної здатності дуги використовується 1° дуга як основний кут для руху. (FW V1.04.60 або пізнішої версії)
20	Недійсний	Для роздільної здатності дуги використовується 10° arcs як основний кут для руху, щоб створити дугу еліпса під кутом 90° градусів і завершитися в точці, де осі X і Y спрямовуються на вказані координати. (FW V1.10.10 або новіша)
21	Недійсний	Для роздільної здатності дуги використовується 5° arcs як основний кут для руху, щоб створити дугу еліпса під кутом 90° градусів і завершитися в точці, де осі X і Y спрямовуються на вказані координати. (FW V1.10.10 або новіша)
22	Недійсний	Для роздільної здатності дуги використовується 1° arcs як основний кут для руху, щоб створити дугу еліпса під кутом 90° градусів і завершитися в точці, де осі X і Y спрямовуються на вказані координати. (FW V1.10.10 або новіша)
Інший	Зсув центру (32-розрядний тип цілого числа)	Значення налаштування в S_5 дорівнює 0 Для роздільної здатності дуги використовується 10° дуга як основний кут для руху.

5. Зверніться до прикладу 2 інструкції DPPMR, щоб дізнатися більше про програмування мовою ST.

Малювання дуги за годинниковою стрілкою (для значень параметрів 0~12 у S_5)

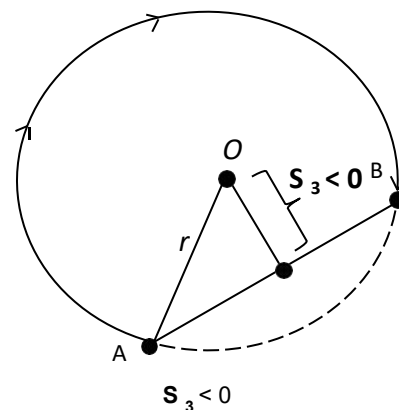
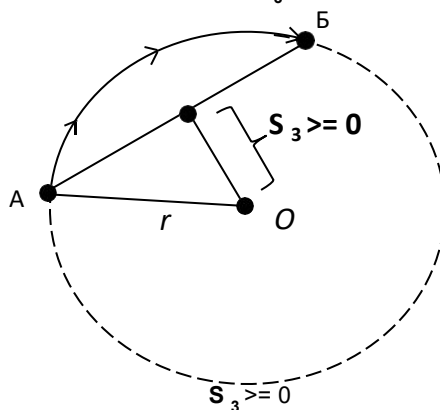
1. Визначте чотири квадранти (QI, QII, QIII і QIV) системи координат, як показано на графіку нижче. Точка А вказує поточні координати на осях X і Y.



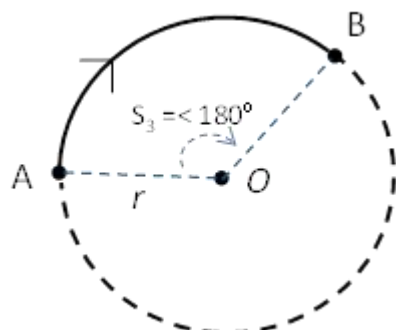
2. Точка В є цільовими координатами, визначеними S_1 і S_2 . Точка О є центром кола, що охоплює точки А і В.
3. Для ПЛК із програмним забезпеченням V1.04.59 або попередніми версіями S_3 — це відстань для зміщення центральної точки О. Для ПЛК із програмним забезпеченням V1.04.60 або новішої версії, S_3 також може бути відстанню для зсуву центральної точки або центрального кута, якщо він визначений як центральний кут, це включений кут АОВ.
4. Точка А позначає початок осей X і Y на площині. Цільова точка В може бути в квадранті I, II, III, IV літак. Використовуйте цільову точку В, щоб з'єднатися з точкою А, щоб намалювати лінію, а потім використовуйте значення параметрів (зміщення центральної точки або центрального кута) у S_3 , щоб визначити траєкторію руху дуги.

На малюнках нижче показано 2 приклади траєкторії руху по дузі (суцільні лінії), коли точка В знаходиться в квадранті I і IV з

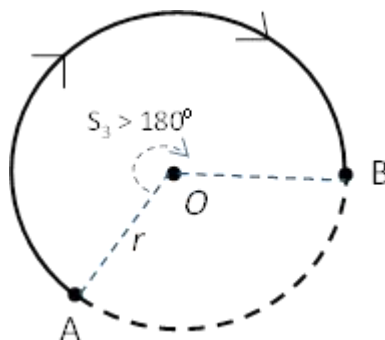
різні значення налаштувань у S_3 (зміщення центральної точки). (r : радіус)



5. На малюнках нижче показано 2 приклади траєкторії переміщення по дузі (суцільні лінії), коли точка В знаходиться в квадранті I і IV з різними значеннями параметрів у S_3 (зсув центрального кута). (r : радіус)

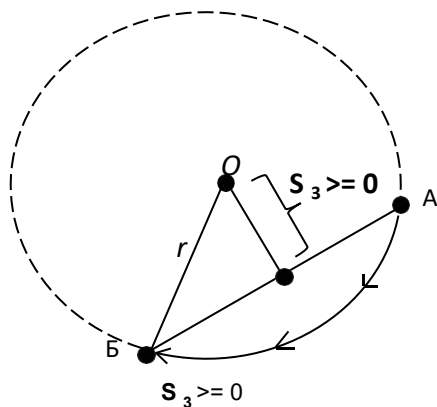


$$S_3 \leq 180,0$$

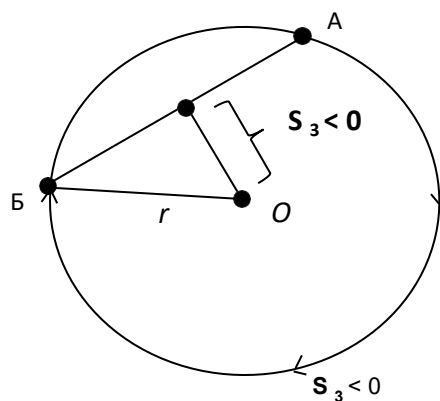


$$S_3 > 180,0$$

6. На малюнках нижче показано 2 приклади траєкторії переміщення по дузі (суцільні лінії), коли точка В знаходиться в квадранті II і III з різними значеннями налаштувань у S_3 (зміщення центральної точки). (r : радіус)

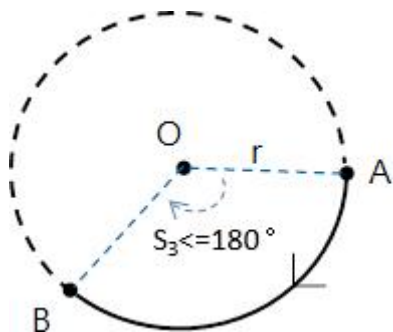


$$S_3 \geq 0$$

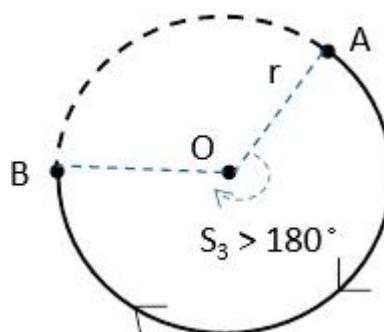


$$S_3 < 0$$

7. На малюнках нижче показано 2 приклади траєкторії переміщення по дузі (суцільні лінії), коли точка В знаходиться в квадранті II і III з різними значеннями параметрів у S_3 (зсув центрального кута). (r : радіус)



$$S_3 \leq 180,0$$

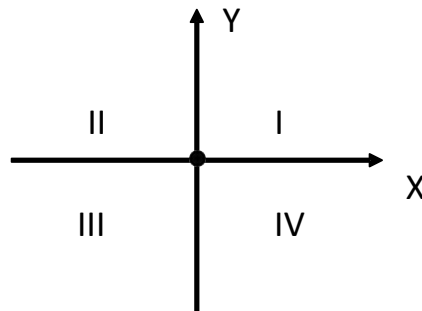


$$S_3 > 180,0$$

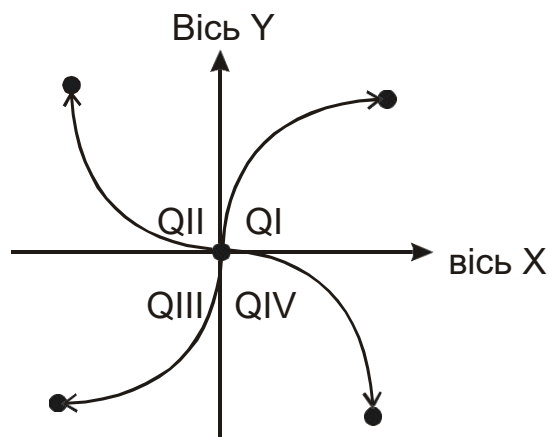
8. Коли точка В знаходиться в точці, де значення на осі X дорівнює 0, а значення $\geq$ на осі Y, точка В знаходиться в квадранті I. Тоді як, якщо точка В знаходиться в точці, де значення на осі X дорівнює 0, а значення < 0 на осі Y, точка В знаходиться в квадранті III.

Малювання дуги під кутом 90 градусів за годинниковою стрілкою (для налаштувань значень 20~22 у S_5) для процесора серії AS FW V1.10.10 або новішої версії

1. Визначте чотири квадранти (QI, QII, QIII і QIV) системи координат, як показано на графіку нижче.



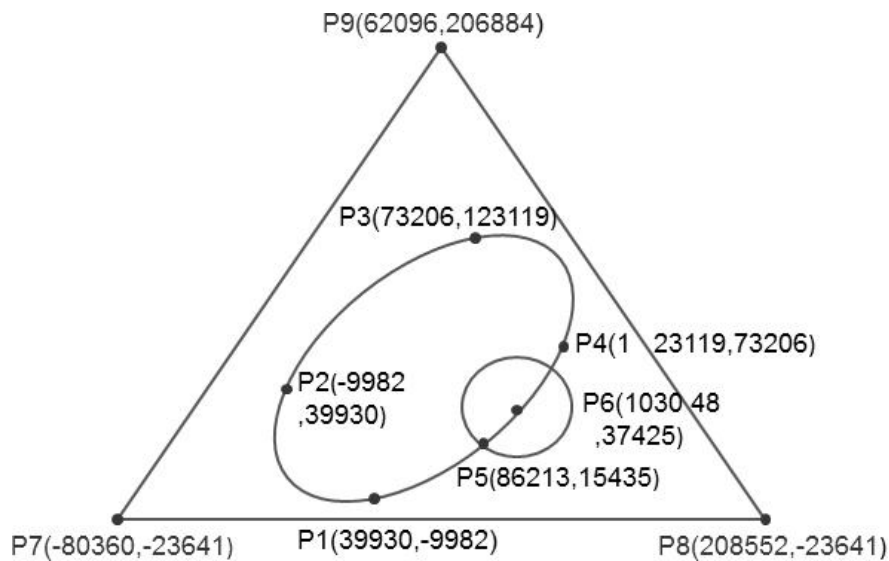
2. Використовуйте значення в S_1 (цільова координата осі X) і в S_2 (цільова координата осі Y), щоб визначити цільове положення в один із наступних чотирьох квадрантів.



3. Оскільки відстані між початковими позиціями та цільовими позиціями по осях X і Y неоднакові, траєкторія руху буде утворена дугою еліпса під кутом 90 градусів. Ця функція корисна для програм, які малюють еліптичні дуги.

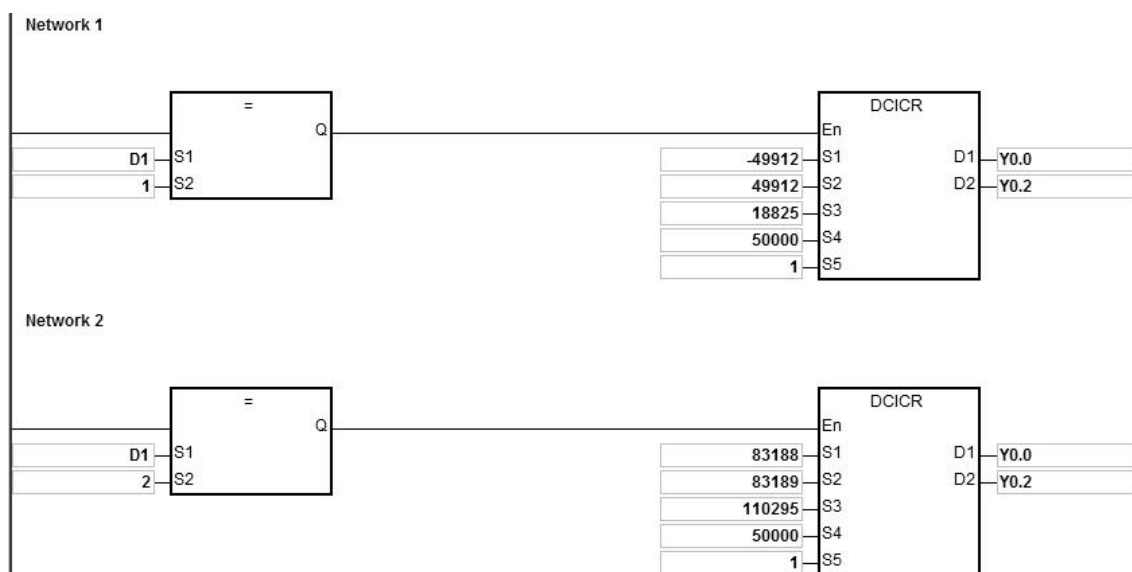
приклад

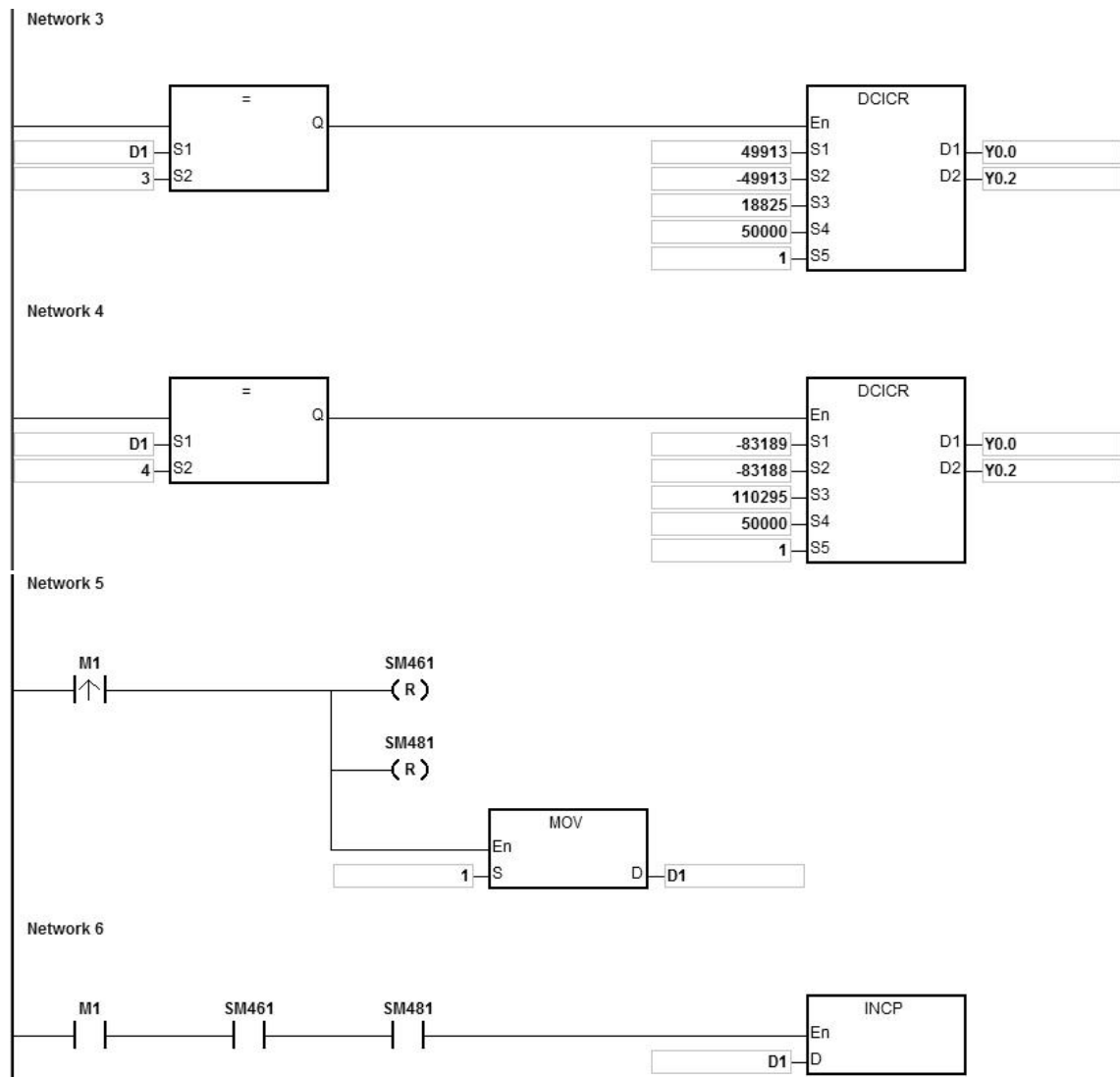
1. Намалюйте ЛОГОТИП DELTA, як показано на малюнку нижче.



2. Етапи: розділіть логотип на три частини.

- Для еліпса: використовуйте інструкцію DCICR для дугової інтерполяції відносного положення за годинниковою стрілкою.
- Абсолютні координати еліпса: P1 (39930 , -9982) , P2 (-9982 , 39930) , P3 (73206 , 123119) і P4 (123119 , 73206) .
- Взявши (39930 , -9982) як початкову точку, ви отримаєте такі відносні координати : (-49912 , 49912) (83188 , 83189) , (49913 , -49913) і (-83189 , -83188) .





Коли ПЛК запускає програму, а M1 увімкнено, ПЛК починає малювати перший сегмент дуги. D1 збільшується на одиницю, коли сегмент дуги завершено, а другий сегмент дуги починає автоматично виконуватися. Ця модель операції повторюється до завершення четвертого сегмента дуги.

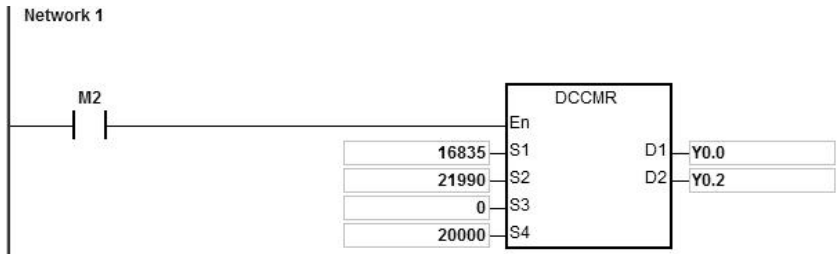
Коли D1=1, використовуючи DCICR, ПЛК малює дугу першого сегмента від P1 до P2 зі зміщенням центру кола: 18825 і роздільною здатністю дуги: 5.

Коли D1=2, використовуючи DCICR, ПЛК малює дугу другого сегмента від P2 до P3 зі зміщенням центру кола: 110295 і роздільною здатністю дуги: 5.

Коли D1=3, використовуючи DCICR, ПЛК малює дугу третього сегмента від P3 до P4 зі зміщенням центру кола: 18825 і роздільною здатністю дуги: 5.

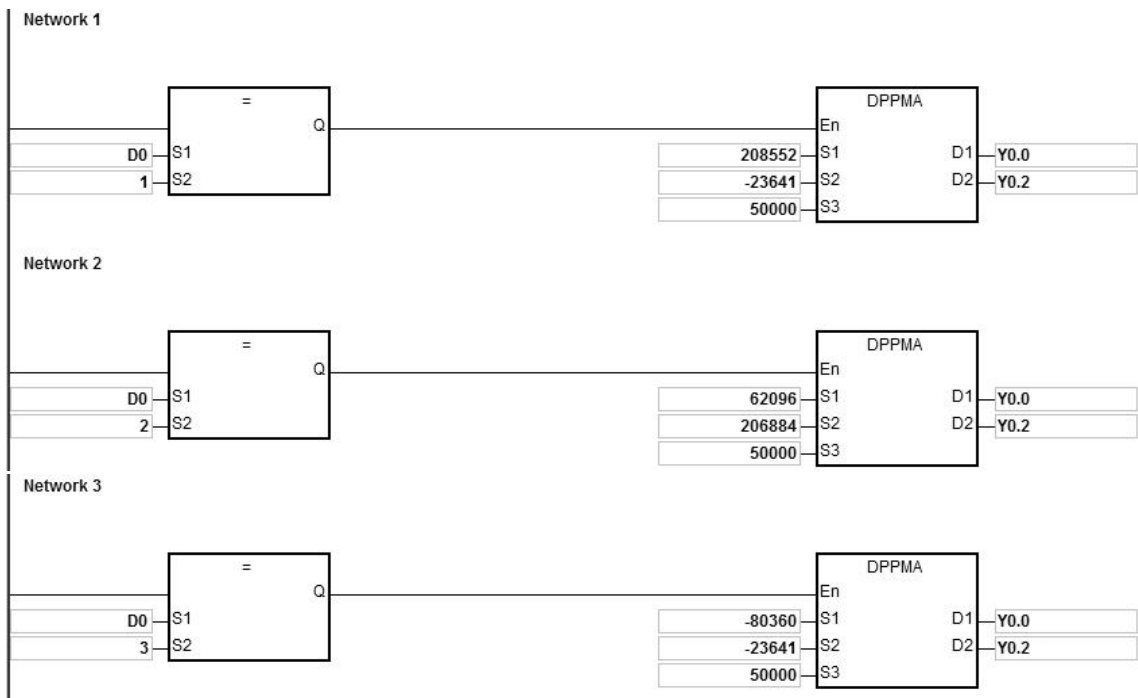
Коли D1=4, використовуючи DCICR, ПЛК малює дугу четвертого сегмента від P4 до P1 зі зміщенням центру кола: 110295 і роздільною здатністю дуги: 5.

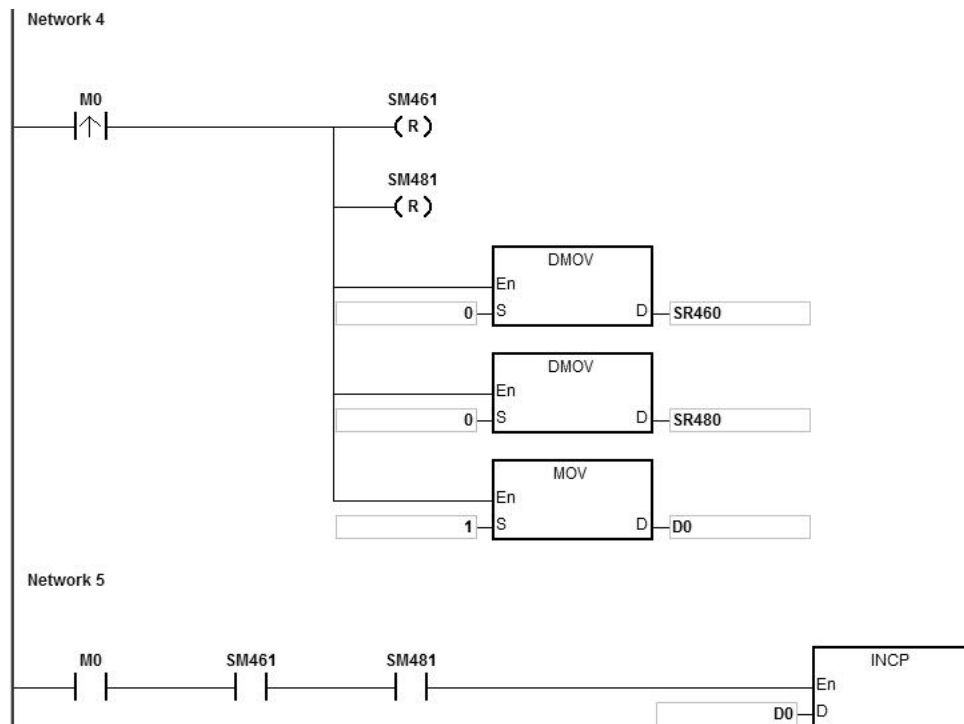
- Для кола = використовуйте інструкцію DCCMR (API 2716) для креслення кола відносного положення.
- Абсолютні координати кола: P5 (86213 , 15435) і P6 (103048 , 37425) . Беручи (86213 , 15435) як початкова точка, дає відносні координати центру кола як (16835 , 21990) .



Коли ПЛК запускає програму, а M2 увімкнено, ПЛК починає малювати коло відносного положення з метою опорна частота 20 кГц.

- Для дельти використовуйте DPPMA (API 2711) для 2-осьового синхронізованого руху в абсолютному положенні.
- Абсолютні координати дельти = початкова точка P7 (-80360 , 23641), P8 (208552 , -23641) і P9 (62096 , 206884).





Коли ПЛК запускає програму і M0 увімкнено, ПЛК починає малювати перший сегмент 2-осьової синхронізованої рух на частоті 50 кГц. D1 збільшується на одиницю, коли сегмент 2-осьового синхронізованого руху завершується, а другий сегмент 2-осьового синхронізованого руху починає автоматично виконуватися. Шаблон операції повторюється, доки не завершиться третій сегмент двовісного синхронізованого руху.

Коли D1=1, ПЛК починає малювати лінію від P7 до P8 за допомогою інструкції DPPMA.

Коли D1=2, ПЛК починає малювати лінію від P8 до P9 за допомогою інструкції DPPMA.

Коли D1=3, ПЛК починає малювати лінію від P9 до P7 за допомогою інструкції DPPMA.

API	Інструкція				Операнд							опис					
2713	D	CICA			$S_1, S_2, S_3, S_4, S_5, D_1, D_2$							2-осьова дугова інтерполяція абсолютного положення за годинниковою стрілкою					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1							●	●	●		○		○	○			
S_2							●	●	●		○		○	○			
S_3							●	●	●		○		○	○		○	
S_4							●	●	●		○		○	○			
S_5							●	●	●		○		○	○			
D_1		○															
D_2		○															
Тип дан	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1			●				●										
S_2			●				●										
S_3			●				●		●								
S_4			●				●										
S_5			●				●										
D_1	●																
D_2	●																

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DCICA	
En	
S1	D1
S2	D2
S3	
S4	
S5	

- S_1 : Цільова координата осі X (абсолютне позиціонування)
- S_2 : Цільова координата осі Y (абсолютне позиціонування)
- S_3 : Зміщення центру або центрального кута
- S_4 : Цільова опорна частота
- S_5 : функція
- D_1 : Пристрій виведення імпульсів по осі X
- D_2 : Пристрій виведення імпульсів для осі Y

Пояснення

1. Ця інструкція виконує дугову інтерполяцію за годинниковою стрілкою по 2-вісній абсолютній позиції. S_1 і S_2 відповідно позначають цільові координати осі X і Y. Щоб отримати додаткові пояснення, зверніться до інструкції DCICR (API 2712).
2. Зверніться до прикладу 2 з інструкції DPPMR (API 2710) для програмування мовою ST.

API	Інструкція				Операнд								опис			
2714	D	CICCR				$S_1, S_2, S_3, S_4, S_5, D_1, D_2$								2-осьова дугова інтерполяція відносного положення проти годинникової стрілки		
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		○
S_4							●	●	●		○		○	○		
S_5							●	●	●		○		○	○		
D_1		○														
D_2		○														

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●		●				
S_4			●				●						
S_5			●				●						
D_1	●												
D_2	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DCICCR	
En	
S1	D1
S2	D2
S3	
S4	
S5	

- S_1 : Цільова координата осі X (відносно розташування)
- S_2 : Цільова координата осі Y (відносно розташування)
- S_3 : Зміщення центру або центрального кута
- S_4 : Цільова опорна частота
- S_5 : функція
- D_1 : Пристрій виведення імпульсів по осі X
- D_2 : Пристрій виведення імпульсів для осі Y

Пояснення

- Ця інструкція виконує 2-осьову дугову інтерполяцію за годинниковою стрілкою відносного положення. S_1 , S_2 і S_3 відповідно позначають цільову координату по осі X, цільову координату по осі Y і зміщення центру кола. Зверніться до наступної операції для малювання дуги проти годинникової стрілки, щоб отримати докладнішу інформацію.
- D_1 і D_2 є вихідними пристроями для осей X і Y відповідно. Зверніться до інструкції DPPMR (API 2710) щодо вибору вихідних точок і режиму виведення для осей.
- S_4 встановлює цільову частоту для опорної частоти. Цільова опорна частота використовується для попереднього розрахунку, коли ПЛК планує шлях руху дуги після виконання інструкції. Якщо розрахунковий процес розрахунку не може досягти запланованої траєкторії руху дуги, вихідна частота автоматично зменшується, щоб виконати функцію синхронізованого малювання дуги.
- S_5 – значення налаштування для вибору функції. Дивіться описи нижче.

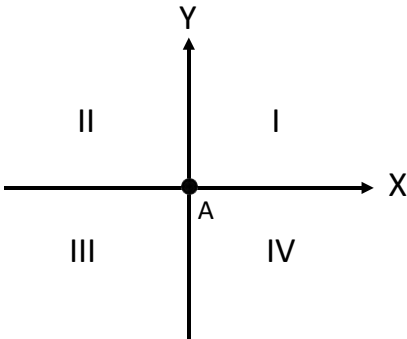
Значення налаштування в S_5	S_3 : Зміщення центру або центрального кута	опис
0	Зсув центру (32-розрядний тип цілого числа)	Для роздільної здатності дуги використовується 10° дуга як основний кут для руху.
1	Зсув центру (32-розрядний тип цілого числа)	Для роздільної здатності дуги використовується 5° дуга як основний кут для руху.
2	Зсув центру (32-розрядний тип цілого числа)	Для роздільної здатності дуги використовується 1° дуга як основний кут для руху. (FW V1.04.60 або пізнішої версії)
10	Зміщення центрального кута (тип з плаваючою комою)	Для роздільної здатності дуги використовується 10° дуга як основний кут для руху. (FW V1.04.40 або пізнішої версії)
11	Зміщення центрального кута (тип з плаваючою комою)	Для роздільної здатності дуги використовується 5° дуга як основний кут для руху. (FW V1.04.40 або пізнішої версії)
12	Зміщення центрального кута (тип з плаваючою комою)	Для роздільної здатності дуги використовується 1° дуга як основний кут для руху. (FW V1.04.60 або пізнішої версії)
20	Недійсний	Для роздільної здатності дуги використовується 10° arcs як основний кут для руху, щоб створити дугу еліпса під кутом 90 градусів і завершитися в точці, де осі X і Y спрямовуються на вказані координати. (FW V1.10.10 або новіша)
21	Недійсний	Для роздільної здатності дуги використовується 5° arcs як основний кут для руху, щоб створити дугу еліпса під кутом 90 градусів і завершитися в точці, де осі X і Y спрямовуються на вказані координати. (FW V1.10.10 або новіша)
22	Недійсний	Для роздільної здатності дуги використовується 1° дуга як основний кут для руху для створення 90

		градусна дуга еліпса і закінчуватися в точці, де осі X і Y спрямовуються на вказані координати. (FW V1.10.10 або новіша)
Інший	Зсув центру (32-розрядний тип цілого числа)	Значення налаштування в S_5 дорівнює 0 Для роздільної здатності дуги використовується 10° дуга як основний кут для руху.

5. Зверніться до прикладу 2 з інструкції DPPMR (API 2710) для програмування на мові ST.

Малювання дуги проти годинникової стрілки (для значень параметрів 0~12 у S_5)

1. Визначте чотири квадранти (QI, QII, QIII і QIV) системи координат, як показано нижче. Точка А вказує поточні координати по осях X і Y.



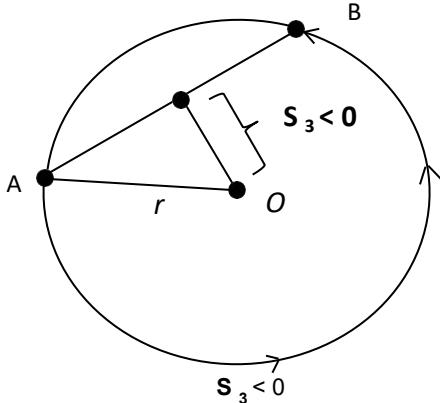
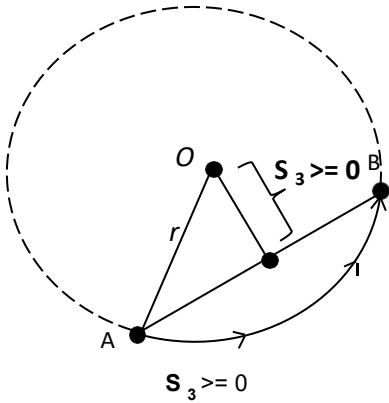
2. Точка В є цільовими координатами, визначеними S_1 і S_2 . Точка О є центром кола, що містить точки А і В.

3. Для ПЛК із програмним забезпеченням V1.04.59 або попередніми версіями S_3 — це відстань для зміщення центральної точки О. Для ПЛК із програмним забезпеченням

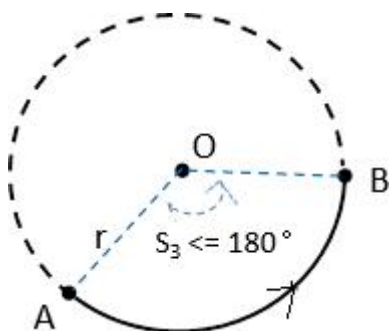
6

V1.04.60 або новішої версії, S_3 також може бути відстанню для зсуву центральної точки або центрального кута, якщо він визначений як центральний кут, це включений кут АОВ.

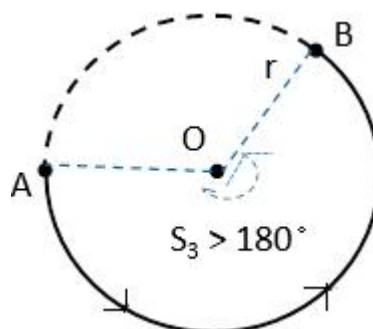
4. Точка А позначає початок осей X і Y на площині. Цільова точка В може бути в квадранті I, II, III, IV літак. Використовуйте цільову точку В, щоб з'єднатися з точкою А, щоб намалювати лінію, а потім використовуйте значення параметрів (зміщення центральної точки або центрального кута) у S_3 , щоб визначити траєкторію руху дуги. На малюнках нижче показано 2 приклади траєкторії руху по дузі (суцільні лінії), коли точка В знаходиться в квадранті I і IV з різні значення налаштувань у S_3 (зміщення центральної точки). (r: радіус)



5. На малюнках нижче показано 2 приклади траєкторії переміщення по дузі (суцільні лінії), коли точка В знаходиться в квадранті I і IV з різними значеннями параметрів у S_3 (зсув центрального кута). (r: радіус)

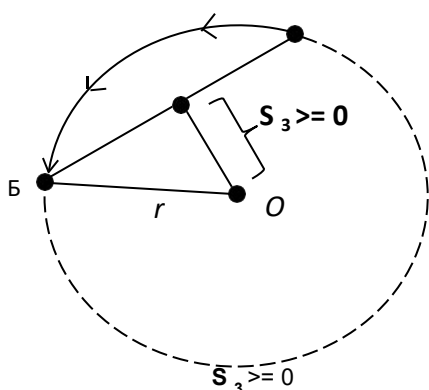


$$S_3 \leq 180,0$$

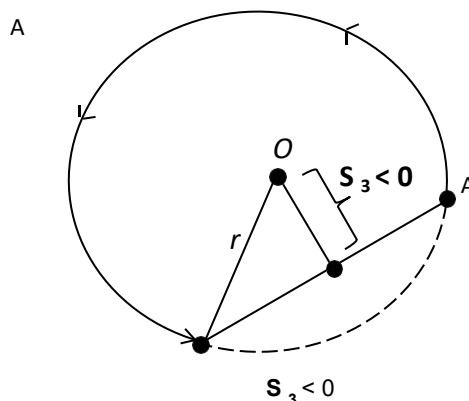


$$S_3 > 180,0$$

6. На малюнках нижче показано 2 приклади траєкторії руху по дузі (суцільні лінії), коли точка В знаходиться в квадранті II і III з різними значеннями налаштувань у S_3 (зміщення центральної точки). (r: радіус)

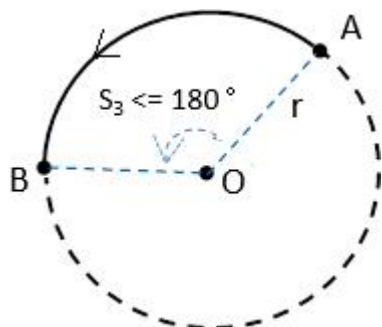


$$S_3 \geq 0$$

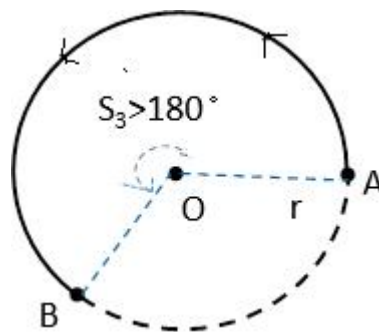


$$S_3 < 0$$

7. На малюнках нижче показано 2 приклади траєкторії переміщення по дузі (суцільні лінії), коли точка В знаходиться в квадранті II і III з різними значеннями параметрів у S_3 (зсув центрального кута). (r: радіус)



$$S_3 \leq 180,0$$

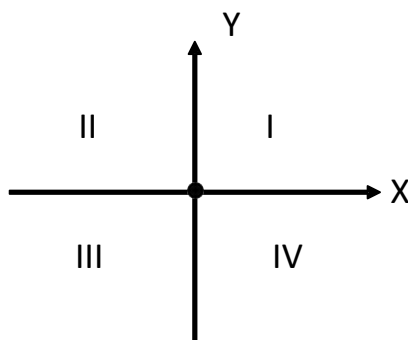


$$S_3 > 180,0$$

8. Коли точка В знаходиться в точці, де значення на осі X дорівнює 0, а значення $\geq$ на осі Y, точка В знаходиться в квадранті
- I. Тоді як, якщо точка В знаходиться в точці, де значення на осі X дорівнює 0, а значення < 0 на осі Y, точка В знаходиться в квадранті III.

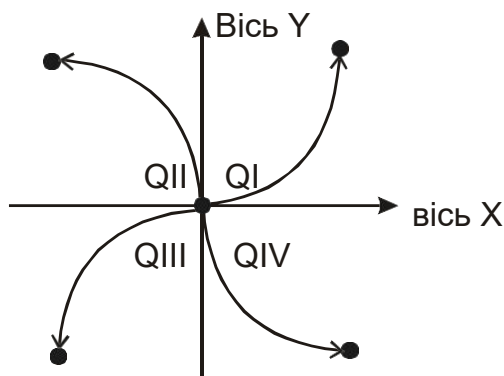
Намалюйте дугу під кутом 90 градусів проти годинникової стрілки (для значень параметрів 20~22 у S_5) для FW CPU серії AS V1.10.10 або пізніше

1. Визначте чотири квадранти (QI, QII, QIII і QIV) системи координат, як показано на графіку нижче.



2. Використовуйте значення в S_1 (цільова координата осі X) і в S_2 (цільова координата осі Y), щоб визначити цільове положення в

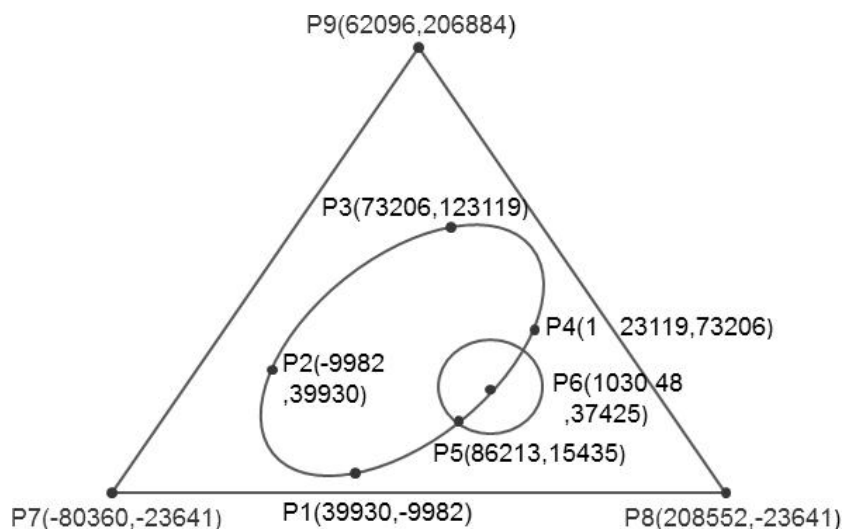
один із наступних чотирьох квадрантів.



3. Оскільки відстані між початковими позиціями та цільовими позиціями по осях X і Y неоднакові, траєкторія руху буде утворена дугою еліпса під кутом 90 градусів. Ця функція корисна для програм, які малюють еліптичні дуги.

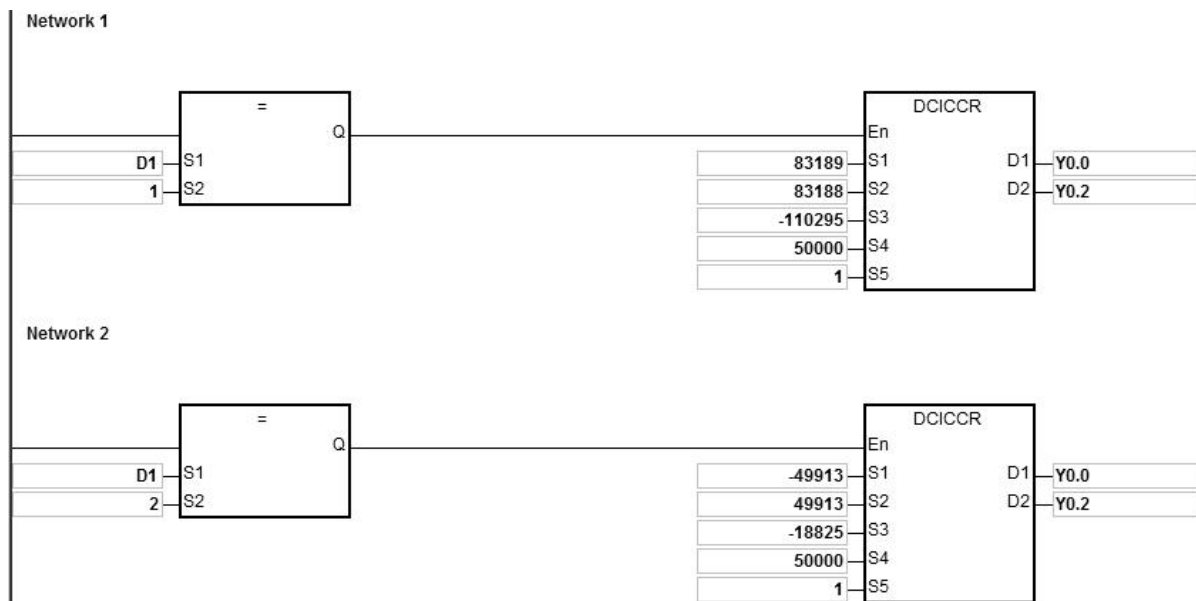
приклад

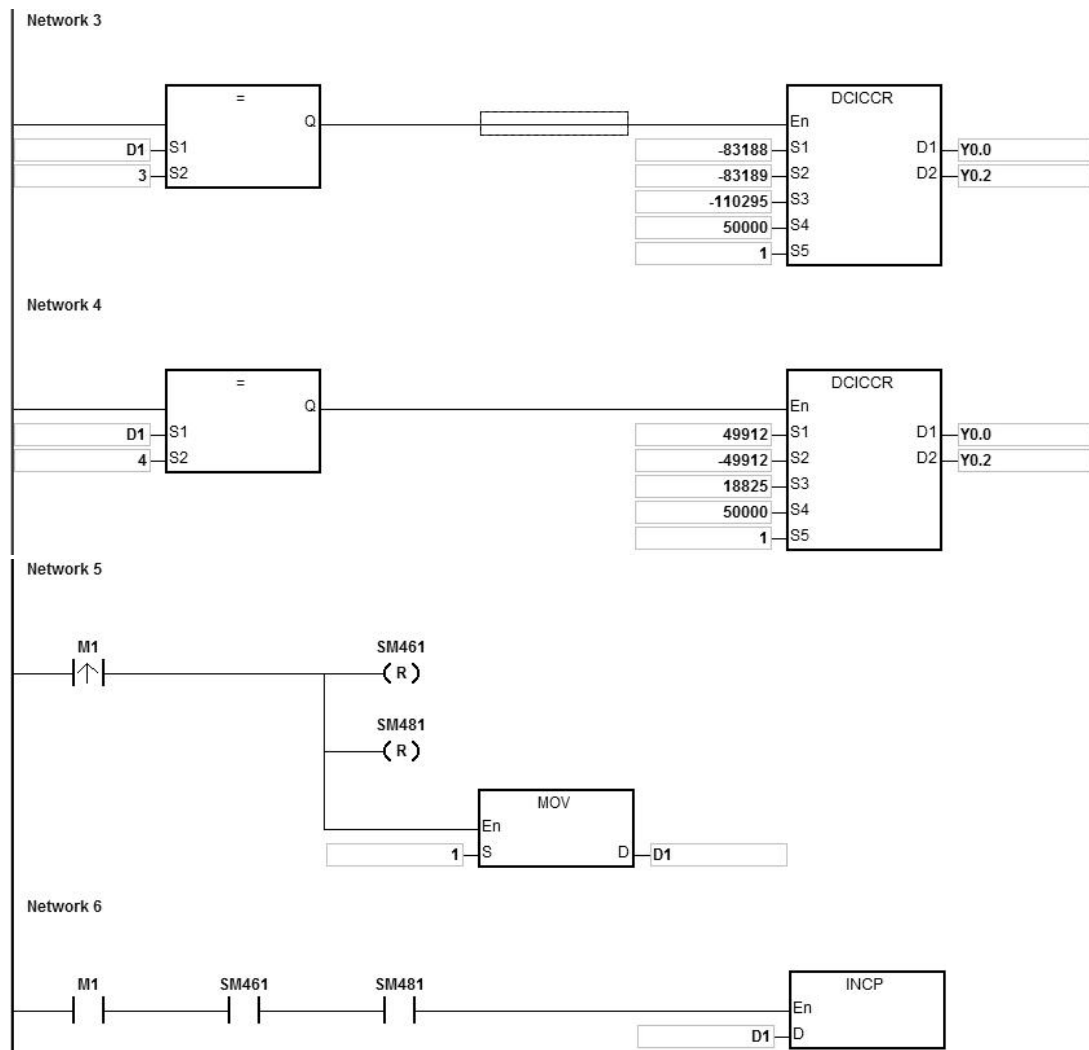
1. Намалюйте ЛОГОТИП DELTA, як показано на малюнку нижче.



2. Етапи: розділіть логотип на три частини.

- Для еліпса: використовуйте інструкцію DCICR (API 2712) для дугової інтерполяції відносного положення проти годинникової стрілки.
- Абсолютні координати еліпса: P1(39930, -9982), P4 (123119, 73206), P3 (73206, 123119) і P2 (-9982, 39930)
- Взявши (39930, -9982) як початкову точку, отримані відносні координати: (83189, 83188), (-49913, 49913), (-83188, -83189) і (49912, -49912).





Коли ПЛК запускає програму, а M1 увімкнено, ПЛК починає малювати перший сегмент дуги за годинниковою стрілкою з частотою 50 кГц. D1 збільшується на 1, коли сегмент дуги завершено, а другий сегмент дуги починає автоматично виконуватися. Схема операції повторюється до завершення четвертого сегмента дуги.

Коли D1=1, використовуючи DCICCR, ПЛК малює дугу першого сегмента від P1 до P4 зі зміщенням центру кола: -110295 і роздільною здатністю дуги: 5.

Коли D1=2, використовуючи DCICCR, ПЛК малює дугу другого сегмента від P4 до P3 зі зміщенням центру кола: -18825 і роздільною здатністю дуги: 5.

Коли D1=3, використовуючи DCICCR, ПЛК малює дугу третього сегмента від P3 до P2 зі зміщенням центру кола: -110295 і роздільною здатністю дуги: 5.

Коли D1=4, використовуючи DCICCR, ПЛК малює дугу четвертого сегмента від P2 до P1 зі зміщенням центру кола: -18825 і роздільною здатністю дуги: 5.

- Коло = використовуйте інструкцію DCCMR (API 2716) для креслення кола відносного положення. Зверніться до прикладу в API 2712 DCICR для детальної роботи.
- Дельта = використовувати DPPMA (APIU 2711) для 2-осьового синхронізованого руху в абсолютному положенні. Для отримання додаткової інформації зверніться до прикладу в DCICR (API 2712).

API	Інструкція				Операнд								опис			
2715	D	CICCA			$S_1, S_2, S_3, S_4, S_5, D_1, D_2$								2-осьова інтерполяція абсолютна позиція проти годинникової стрілки			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		○
S_4							●	●	●		○		○	○		
S_5							●	●	●		○		○	○		
D_1		○														
D_2		○														

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●		●				
S_4			●				●						
S_5			●				●						
D_1	●												
D_2	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DCICCA	
En	
S1	D1
S2	D2
S3	
S4	
S5	

- S_1 : Цільова координата осі X (абсолютне позиціонування)
- S_2 : Цільова координата осі Y (абсолютне позиціонування)
- S_3 : Зміщення центру або центрального кута
- S_4 : Цільова опорна частота
- S_5 : функція
- D_1 : Пристрій виведення імпульсів по осі X
- D_2 : Пристрій виведення імпульсів для осі Y

Пояснення

1. S_1 і S_2 відповідно позначають цільову координату на осі X і цільову координату на осі Y. Щоб отримати пояснення до інструкції, зверніться до інструкції DCICCR (API 2714).
2. Зверніться до прикладу 2 з інструкції DPPMR (API 2710) для програмування мовою ST.

API	Інструкція			Операнд								опис				
2716	D	CCMR		$S_1, S_2, S_3, S_4, D_1, D_2$								Креслення кола відносного положення				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		O		O	O		
S_2							●	●	●		O		O	O		
S_3							●	●	●		O		O	O		
S_4							●	●	●		O		O	O		
D_1		O														
D_2		O														

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4			●				●						
D_1	●												
D_2	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DCCMR	
En	
S1	D1
S2	D2
S3	
S4	

- S_1 : Координата центру осі X (відносне розташування)
- S_2 : Координата центру осі Y (виносне розташування)
- S_3 : Вибір функції
- S_4 : Цільова опорна частота
- D_1 : Пристрій виведення імпульсів по осі X
- D_2 : Пристрій виведення імпульсів для осі Y

Пояснення

1. Ця інструкція малює коло на основі відносного розташування. S_1 і S_2 відповідно координати осі X і Y центру кола (відносно положення).
2. S_3 – значення налаштування для вибору функції. Дивіться описи нижче.

S_3 : Вибір функції	Описи
0	Для роздільної здатності дуги використовується 10° дуга як основний кут для руху за годинниковою стрілкою.
1	Для роздільної здатності дуги використовується 5° дуга як основний кут для руху за годинниковою стрілкою.
2	Для роздільної здатності дуги використовується 10° дуга як основний кут для руху проти годинникової стрілки.
3	Для роздільної здатності дуги використовується 5° дуга як основний кут для руху проти годинникової стрілки.
4	Для роздільної здатності дуги використовується 1° Дуга як основний кут для руху за годинниковою стрілкою. (FW V1.04.60 або пізнішої версії)
5	Для роздільної здатності дуги використовується 1° дуга як основний кут для руху проти годинникової стрілки. (FW V1.04.60 або пізнішої версії)
Інший	Значення налаштування в S_5 дорівнює 0 Для роздільної здатності дуги використовується 1° дуга як основний кут для руху за годинниковою стрілкою.

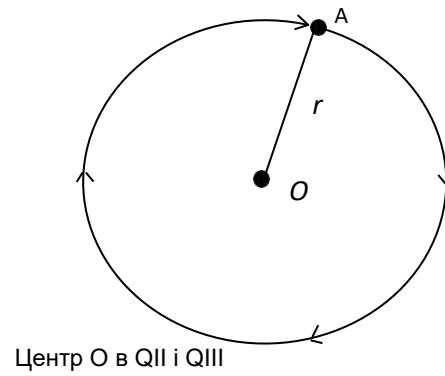
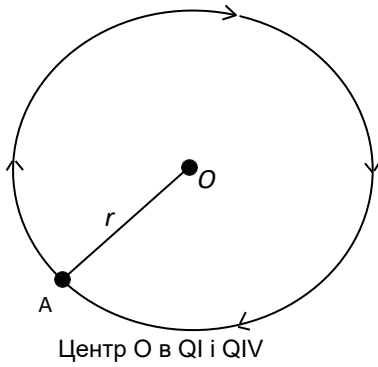
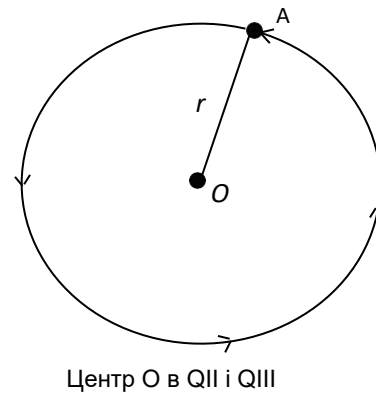
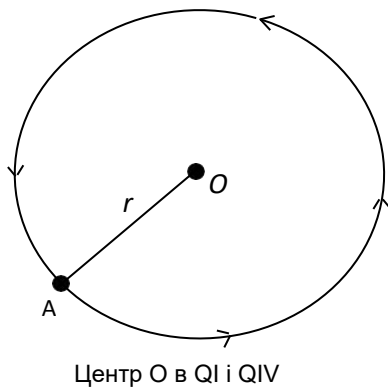
6

3. D_1 і D_2 - пристрої виведення імпульсів для осей X і Y відповідно. Див. інструкцію DPPMR (API 2710) для вибору вихідних точок і режиму виведення для осей.
4. S_4 встановлює цільову частоту для опорної частоти. Цільова опорна частота використовується для попереднього розрахунку, коли

ПЛК планує шлях руху дуги після виконання інструкції. Але якщо процес розрахунку, який оцінюється, не може досягти запланованої траєкторії руху дуги, вихідна частота автоматично зменшується для виконання функції синхронізованого малювання дуги.
5. Коли координата центральної точки осі X $O=0$, точка O визначається як точка, що залишається в QI точки A, якщо вісь Y цільова координата ≥ 0 . Тоді як, якщо цільова координата осі Y < 0 , точка B визначається як точка, що залишається в QIII точки A.
6. Зверніться до прикладу 2 з інструкції DPPMR (API 2710) для програмування мовою ST.

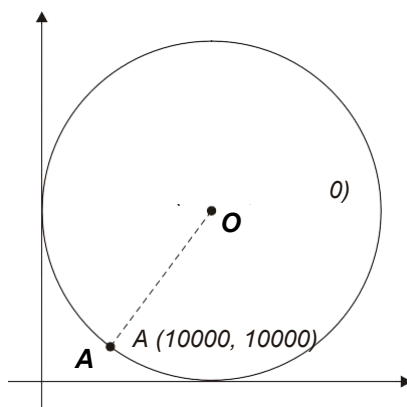
Малюнок кола за годинниковою стрілкою:

Точка A — поточне положення, а точка O — цільовий центр, r — радіус кола, як показано на малюнках нижче.

**Малюнок кола проти годинникової стрілки:**

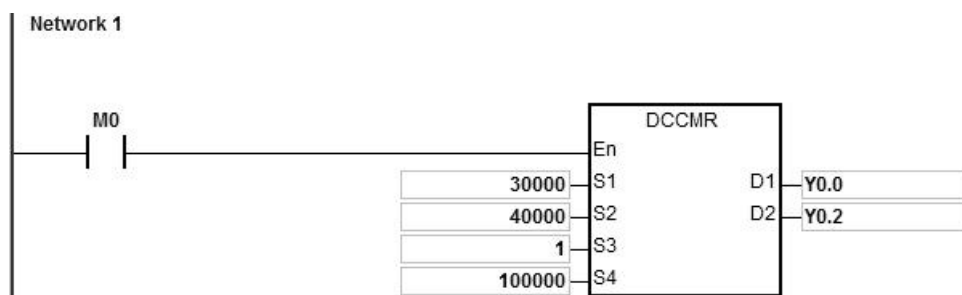
приклад

1. Взявши (40000, 50000) за центр O і (10000, 10000) за точку A, намалюйте коло, як показано нижче.



2. Кроки:

- Розраховані відносні координати точки O: (30000, 40000)
- Координати точки A — поточне положення; SR460=10000 і SR480=10000
- $S_3 = 1$ і креслення кола за годинниковою стрілкою виконується з кутом 5° як основний кут руху.
- Коли ПЛК запускає програму і для M0 встановлено значення ON, починається креслення кола відносного положення.



3. Операція:

- Коли ПЛК запускає програму і M0 увімкнено, ПЛК починає малювати коло за годинниковою стрілкою з частотою 100 кГц.
- Після завершення малювання кола SM461 і SM481 увімкнені.

API	Інструкція				Операнд								опис			
2717	D	CCMA			$S_1, S_2, S_3, S_4, D_1, D_2$								Креслення кола з абсолютним положенням			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1							●	●	●		○		○	○		
S_2							●	●	●		○		○	○		
S_3							●	●	●		○		○	○		
S_4							●	●	●		○		○	○		
D_1		○														
D_2		○														

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4			●				●						
D_1	●												
D_2	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DCCMA	
En	
S1	D1
S2	D2
S3	
S4	

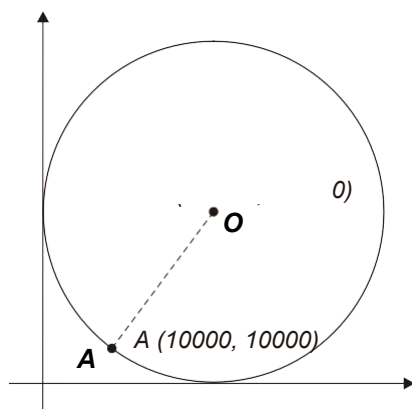
- S_1 : Координата центру осі X (абсолютне позиціонування)
- S_2 : Координата центру осі Y (абсолютне положення)
- S_3 : Вибір функції
- S_4 : Цільова частота
- D_1 : Пристрій виведення імпульсів по осі X
- D_2 : Пристрій виведення імпульсів для осі Y

Пояснення

1. S_1 і S_2 відповідно координати осі X і Y центру кола (абсолютне позиціонування). Для отримання додаткової інформації зверніться до інструкції DCCMR (API 2716).
2. Зверніться до прикладу 2 з інструкції DPPMR (API 2710) для програмування мовою ST.

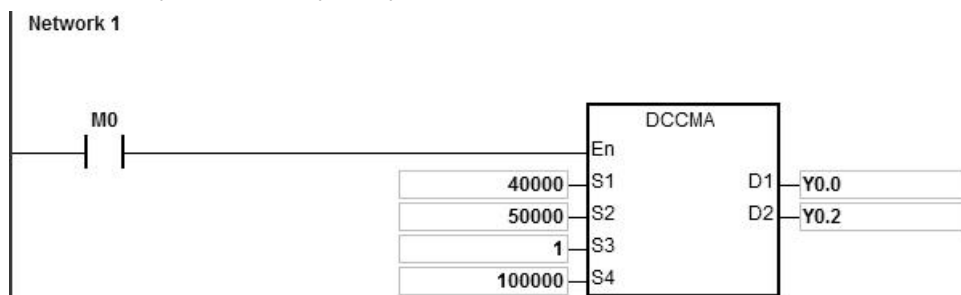
приклад

1. Взявши (40000, 50000) за центр O і (10000, 10000) за точку A, намалюйте коло, як показано нижче.



3. Кроки:

- Встановіть абсолютну координату X і координату Y (40000, 50000) в інструкції
- Точка A є поточною позицією, SR460=10000 і SR480=10000
- S_3 дорівнює 1, а коло за годинниковою стрілкою накреслено з кутом 5° як основний кут руху.
- Коли ПЛК запускає програму і M0 увімкнено, починається малювання кола абсолютного положення.



4. Операція:

- Коли ПЛК запускає програму і M0 увімкнено, ПЛК починає малювати коло за годинниковою стрілкою з частотою 100 кГц.
- Після завершення малювання кола SM461 і SM481 увімкнені.

API		Інструкція				Операнд							опис					
2718			TPO				S , D ₁ , D ₂							Таблиця планування позиції контролює вихід				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F		
S								●				○	○	○				
D ₁								●										
D ₂		●	●	●				●										

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●				●							
D₁		●				●							
D₂	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	-

СИМВОЛ



- S** : Перше вихідне число в таблиці планування позицій
- D₁** : Вихідний номер, який виводиться
- D₂** : Прапор перемикач в кінці виведення послідовного номера

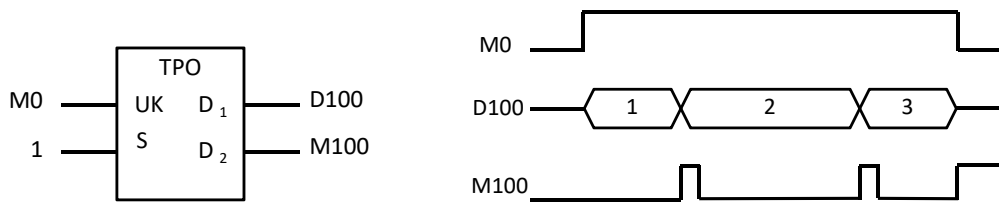
Пояснення:

- Ця інструкція використовує таблицю планування позицій для керування виходом. **S** – число, зазначене в таблиці планування посадки. Якщо номер не існує в таблиці, інструкція не виконується, SM0 увімкнено, а код помилки SR0=16#2027.

Single-axis point-to-point motion Single-axis multi-segment motion 2-axis linear interpolation 2-axis arc interpolation							
No.	Axis No.	Output	ABS/REL mode	Target position	Target speed	Bias speed(Vbi...	Acceleratio
1	Axis 1	Pulse	Reletive	1000000	100000	1000	3000

- S** - це перше вихідне число під час роботи ПЛК. Зверніться до значення в **D₁**, якщо під час виведення виводяться послідовні числа.
- Коли виводиться одне число або перше з послідовних чисел, прапор перемикача **D₂** встановлюється на OFF. Коли вихід послідовного числа досягає номера виходу, на який потрібно переключитися, **D₂** вмикається протягом циклу сканування, а потім знову вмикається, доки не буде завершено виведення останнього числа.

Нижче наведено приклад і часову діаграму для перемикання стану прапора.



4. Прапор перемикання D_2 залежить від часу сканування ПЛК. Якщо час переходу на наступний вихідний номер менший ніж час сканування, прапорець для перемикання на вихідний номер може не скинутися на OFF.
5. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.
6. Якщо використовується функція Одноосьовий багатосегментний рух і версія мікропрограми нижче V1.07.10, ви не можете використовувати прапор зупинки під час виведення. Якщо вам потрібно припинити виведення, скористайтесь цією інструкцією.
7. Для мікропрограми версії 1.08.00 або пізнішої версії ви можете використовувати позначку зупинки під час виведення, щоб зупинити виведення. Якщо ви використовуєте позначку зупинки та цю інструкцію, виведення буде зупинено після завершення виведення цього сегмента. Наприклад, припустимо, що сегменти 1, 2 і 3 для виведення в послідовному порядку, і ви використовуєте позначку зупинки під час виведення сегмента 2. Система завершить виведення сегмента 2, а потім, коли ви вимкнете позначку зупинки, виконання ця інструкція буде продовжена в сегменті 3.
8. Якщо для виведення використовується функція одноосного руху від точки до пінти та використовується позначка зупинки під час виведення, це
instruction зупинить виконання та відновить виконання та виведе залишки після того, як прапор зупинки буде OFF.
9. Якщо використовується функція 2-осьової дугової інтерполяції та використовується позначка зупинки під час виведення, ця інструкція зупиниться
виконання після однієї основної одиниці дугової інтерполяції (1 градус), і ця інструкція відновлює виконання після того, як прапор зупинки OFF, і виводить залишки.
10. Для мікропрограми версії 1.12.10 або новішої підтримується зупинка поточної послідовності виведення та початок нової під час виконання інструкції TPO. Коли відповідний SM змінюється на ON, ЦП ПЛК діятиме відповідно до налаштувань у відповідному SR, щоб вивести певну кількість сегментів або наступний сегмент у таблиці позиціонування. Два сценарії:
 - A. Якщо відповідний SM змінюється на ON, а значення в SR1291-SR1296 є дійсним, ЦП ПЛК діятиме відповідно, виводячи певну кількість сегментів у таблиці позиціонування. Після цього ЦП ПЛК очищає відповідні SM і SR.
 - B. Якщо відповідний SM змінюється на ON, але значення в SR1291-SR1296 дорівнює 0 або HE є дійсним значенням, або таблиця позиціонування HE існує, ЦП ПЛК припинить виведення. Після очищення відповідного SM ЦП ПЛК продовжить виводити наступний сегмент у таблиці позиціонування.

Відповідна вісь, номери SM і SR

Вісь для виведення	Прапор, щоб зупинити поточну послідовність виведення	Конкретний номер сегмента
Вісь 1 (Y0.0/Y0.1)	SM591	SR1291
Вісь 2 (Y0.2/Y0.3)	SM592	SR1292
Вісь 3 (Y0.4/Y0.5)	SM593	SR1293
Вісь 4 (Y0.6/Y0.7)	SM594	SR1294
Вісь 5 (Y0.8/Y0.9)	SM595	SR1295
Вісь 6 (Y0.10/Y0.11)	SM596	SR1296

приклад 1:

1. Увімкніть функцію перетворення механічних одиниць

Axis Settings | Upload Password Settings | Positioning Settings

Setting	Displacement on axis for one motor rotation(lead)			Pulses for one motor rotation(resolution)
Axis	Output	Enable	Value Setting	Unit Name
Axis 1	Pulse	☒	1	mm

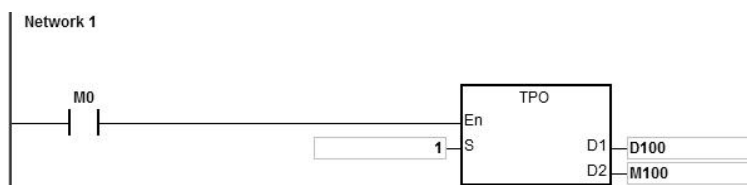
Value Setting: 200

2. Налаштуйте таблицю планування позиції та завантажте параметри

Цільове положення становить 2000 (одиниця: мм). Після перетворення в таблиці планування положення 400 000 вихідних імпульсів завантажуються та зберігаються в ПЛК.

No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	Bias speed(Vbi...	Acceleration ti...
1	Axis 1	Relative	2000	100000	200	200

Коли M0 увімкнено, Y0.0 видає 400 000 імпульсів на 100 кГц. Після завершення виведення M100 увімкнено.



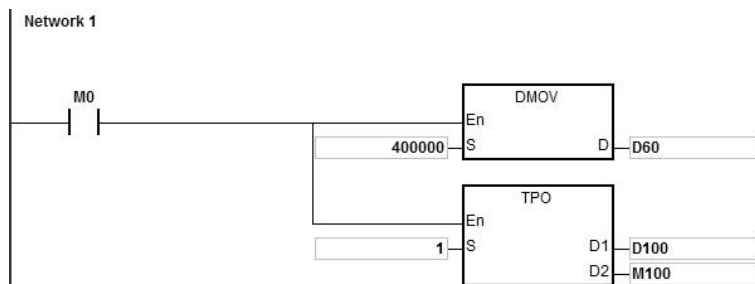
Приклад 2: використовуйте глобальний символ або пристрій D у цільовій позиції

1. Використовуйте пристрій D у Target Position таблиці планування позиції та завантажте параметри.

Одиницею для D60 у ПЛК є імпульс за замовчуванням. Якщо вам потрібно використовувати пристрої D у цільовому положенні та в механічному блоці, вам потрібно перетворити механічний блок на імпульсний.

No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	Bias speed(Vbi...	Acceleration ti...
1	Axis 1	Relative(Symbol/Address)	D60	100000	200	200

2. Коли M0 увімкнено, а D60 становить 400 000. Y0.0 видає 400 000 імпульсів на 100 кГц. Після завершення виведення M100 увімкнено.



Примітка: одиниця є імпульсною, якщо ви використовуєте глобальні символи, пристрої D або інструкції, включаючи DTPWS, DTPWL, і DTPWC для введення значень у Target Position.

3. Якщо в цільовій позиції використовуються глобальні символи

A. Вам потрібно призначити адресу для глобального символу.

Global Symbols				
	Class	Identifiers	Address	Type... Initial Value (Activ...
VAR		Target_Axis1	D60	DWORD N/A

B. Використовуйте глобальні символи в Цільовій позиції таблиці планування.

Single-axis point-to-point motion		Single-axis multi-segment motion		2-axis linear interpolation	2-axis arc interpolation
No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	Bias speed(Vbi... Acceleration ti...
1	Axis 1	Relative(Symbol/Address)	Target axis1	100000	200 200

Приклад 3: використання символу скасування та прапора завершення

1. Налаштуйте таблицю планування позиціонування, як показано нижче.

Single-axis point-to-point motion			Single-axis multi-segment motion		2-axis linear interpolation	2-axis arc interpolation			
No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	ACC/DEC time(ms)	Abort signal(I/M)	Completion action	Completion flag(I/M)	No. of next segment
3	Axis 1	Relative	10000	5000	300	M100	Go to next segment		4
4	Axis 1	Relative	15000	8000	100		Completion	M777	---

2. Сигнал переривання: коли № 3 раптово припиняє виведення, M100 увімкнено, і система починає виводити № 4 негайно.

Символ переривання застосовний для таких умов: зовнішні переривання, високошвидкісні переривання порівняння та пристрої M (M0~8191).

3. Прапор завершення: коли номер 4 завершує виведення, M777 увімкнено. Коли виконання цієї таблиці планування позиціонування завершено, система викликає переривання або запускає пристрій M.

Ви можете використовувати номери переривань I510~519 і пристрій M (M0~M8191) у таблиці планування позиціонування.

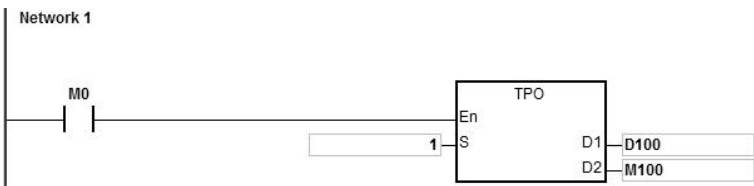
Примітка. Якщо ви використовуєте переривання замість пристроїв M, це не впливає на цикл сканування ПЛК.

Приклад 4: Багатосегментний рух по одній осі

1. Налаштуйте таблицю планування позиціонування, як показано нижче.

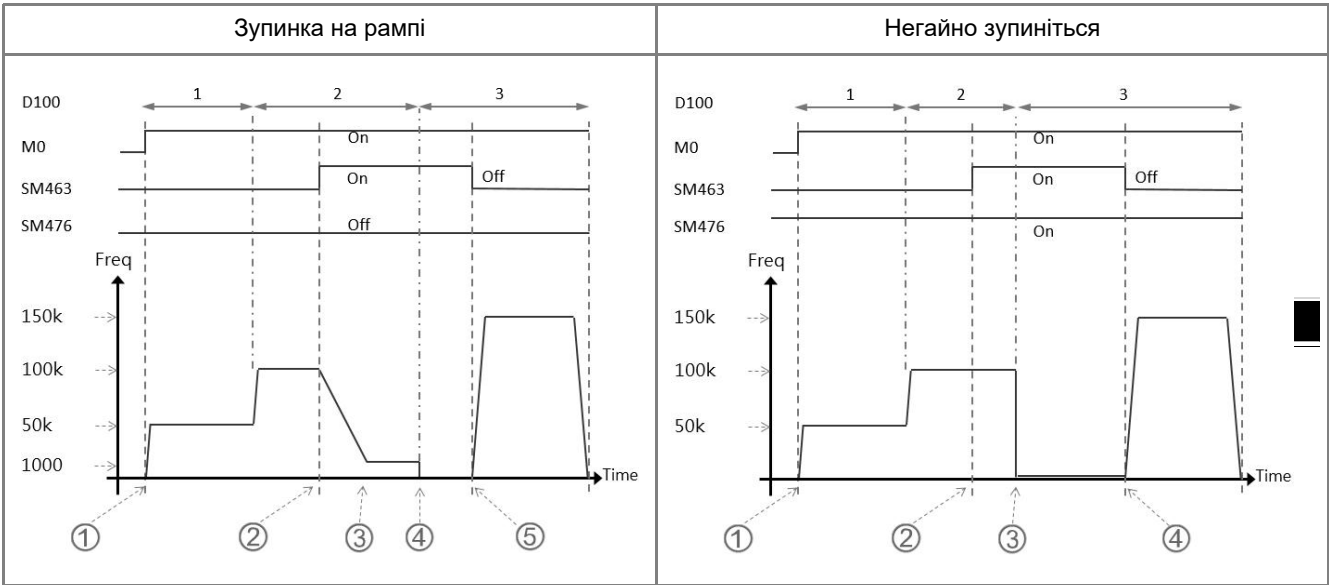
Single-axis point-to-point motion		Single-axis multi-segment motion		2-axis linear interpolation		2-axis arc interpolation	
No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	ACC/DEC time(...)	Abort signal(I/...	Completion acti...
1	Axis 1	Relative	200000	50000	200		Go to next segmer
2	Axis 1	Relative	400000	100000	200		Go to next segmer
3	Axis 1	Relative	600000	150000	200		Go to next segmer

2. Коли M0 увімкнено, Y0.0 послідовно виводить номери від №1 до №3. Після завершення виведення M100 увімкнено.



1. Прапор зупинки та негайно зупинити прапор

а) Наступна графіка показує вихідну діаграму на основі прикладу 4.



б) Зупинка при зниженні

- (1) Коли M0 увімкнено, інструкція TPO виконується та починає виводити сегмент № 1.
- (2) Коли прапор зупинки SM463 увімкнено під час виведення сегмента № 2, стан прапора негайної зупинки SM476 підтверджується.
- (3) Якщо прапор негайної зупинки SM476 OFF, вихідний сигнал зменшується до зупинки відповідно до часу зниження SR465 на осі 1, доки він не впаде до найнижчої частоти 1000 Гц.
- (4) Вихід на найнижчій частоті 100 Гц і поки вихід № 2 не буде завершено та зупинено. (імпульс = 600 К)
- ⑤ Після того, як прапор зупинки SM463 перемикається у положення OFF., виконання інструкції TPO починає виводити сегмент №3 до завершення виведення. (імпульс = 1200 К)

с) Негайно зупиніться

- (1) Коли M0 увімкнено, інструкція TPO виконується та починає виводити сегмент № 1.
- (2) Коли прапор зупинки SM463 увімкнено під час виведення сегмента № 2, стан прапора негайної зупинки SM476 підтверджується.
- (3) Якщо прапор негайного припинення SM476 увімкнено, вихід припиняється після завершення виведення № 2. (імпульс = 600 К).
- (4) Після того, як прапор зупинки SM463 перемикається у положення OFF., виконання інструкції TPO починає виводити сегмент №3 до завершення виведення. (імпульс = 1200 К)

API		Інструкція			Операнд								опис					
2719		D	TPWS		P	S ₁ , S ₂ , S ₃								Налаштування вихідних параметрів однієї осі в таблиці планування положення				

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S ₁								●	●				O	O		
S ₂								●	●				O	O		
S ₃								●	●				O	O		

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S ₁			●				●						
S ₂			●				●						
S ₃			●				●						

Інструкція по пульсу		16-бітна інструкція	32-розрядна інструкція
AS		-	AS

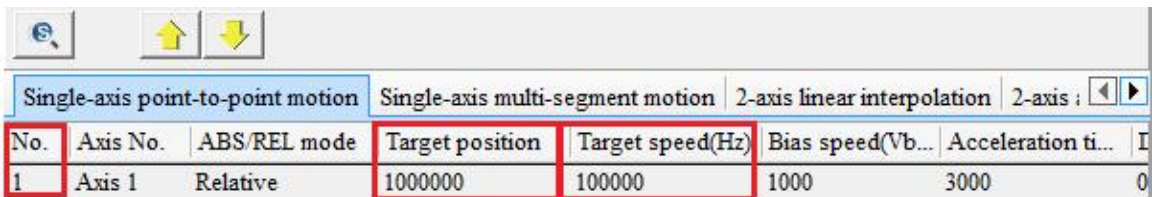
СИМВОЛ

DTPWS	DTPWSP
En	En
S1	S1
S2	S2
S3	S3

- S_1 : Число, указане в таблиці планування посадки
- S_2 : Цільова позиція
- S_3 : Цільова швидкість

Пояснення

- Ця інструкція встановлює вихідні параметри однієї осі в таблиці планування положення. S_1 – номер, зазначений у таблиці планування посадки. Якщо число не існує в таблиці, або число не під рухом по одній осі «точка-точка», або під рухом по одній осі з кількома сегментами, інструкція не виконується, SM0 увімкнено, а код помилки SR0=16#2027.



No.	Axis No.	ABS/REL mode	Target position	Target speed(Hz)	Bias speed(Vb...	Acceleration ti...	I
1	Axis 1	Relative	1000000	100000	1000	3000	0

- S_2 — цільова позиція (має бути 32-розрядним цілим числом). Якщо ви використовуєте перетворення механічних одиниць під час редагування таблиці планування позиції в ISPSoft, спершу використовуйте інструкцію з перетворення для модифікації.
- S_3 — цільова швидкість. Діапазон становить від 1 до 200 000 Гц.

Примітка 1. Максимальна швидкість вихідної осі Y0 і Y2 становить 4 МГц для AS324MT.

Примітка 2: якщо значення налаштування виходить за межі діапазону, інструкція автоматично змінює налаштування на мінімальне або максимальне значення.

- Коли виконується інструкція для зміни параметрів для однієї осі, яка виводить імпульси, змінені параметри зберігаються в таблиці та не діють, доки не почнеться наступний вихід.
- Параметри, змінені інструкцією, можна змінювати лише під час роботи ПЛК. Останній записаний параметр не зберігається, коли живлення вимикається. Таблиця, яку ви редагуєте в ISPSoft і завантажуєте в ПЛК, обробляється як таблиця планування позицій за замовчуванням, коли живлення ON.

6. Ця інструкція HE змінює час прискорення або уповільнення. Якщо вам потрібно змінити час для прискорення або уповільнення, ви можете встановити прапор SM585 на ON. А потім змініть час прискорення та уповільнення, встановлений у SR, щоб змінити час прискорення в русі від точки до точки по одній осі та багатосегментному русі по одній осі або час уповільнення в русі від точки до точки по одній осі. Повторно виконайте цю інструкцію, щоб актуалізувати цю модифікацію. Ця функція доступна для ПЛК із FW V1.06.00.
7. Під час виконання інструкції TPO зверніть увагу на час зміни параметрів за допомогою цієї інструкції. Рекомендується уникати зміни використовуваного номера для результату, який виконується або буде виконано незабаром.

Приклад 1: Змініть параметри осі 3 номеру 5 під час виконання в русі від точки до точки по одній осі.

Single-axis point-to-point motion			Single-axis multi-segment motion			2-axis linear interpolation		2-axis arc interpolation		
No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	Bias speed(Vbias)(Hz)	Acceleration time(Tacc)(ms)	Deceleration time(Tdec)(ms)	Completion action	Completion flag(I/M)	No. of next segment
1	Axis 2	Relative	1000	1000	100	100	200	Go to next segment		2
5	Axis 3	Relative	2000	2000	50	50	100	Completion		---

- 1) Використовуйте інструкцію DMOV, щоб визначити нове положення цілі та швидкість цілі.
- 2) Використовуйте інструкцію MOV, щоб визначити початкову/кінцеву частоту в SR503, час прискорення в SR504 і час уповільнення в SR505 для осі 3. Після цього оновлюються відносні параметри в русі від точки до точки по одній осі.
- 3) Встановіть прапорець SM585 на ON.
- 4) Повторно виконайте інструкцію DTPWS, щоб актуалізувати модифікацію.
- 5) Після цього SM585 автоматично вимикається.

Приклад 2: Змініть параметри осі 3 номеру 2 під час виконання одновісного багатосегментного руху.

Single-axis point-to-point motion			Single-axis multi-segment motion			2-axis linear interpolation		2-axis arc interpolation		
No.	Axis ...	ABS/REL mode	Target position	Target speed(Hz)	ACC/DEC time(ms)	Abort signal(I/M)	Completion action	Completion flag(I/M)	No. of next segment	
2	Axis 3	Relative	1000	1000	100		Go to next segment		2	
3	Axis 3	Relative	2000	2000	50		Go to next segment		3	
4	Axis 3	Relative	1500	1500	2000		Completion	M100	---	

- 1) Використовуйте інструкцію DMOV, щоб визначити нове положення цілі та швидкість цілі.
- 2) Використовуйте інструкцію MOV, щоб визначити час прискорення в SR504. Після цього оновлюються відносні параметри в одновісному багатосегментному русі.
- 3) Встановіть прапорець SM585 на ON.
- 4) Повторно виконайте інструкцію DTPWS, щоб актуалізувати модифікацію.
- 5) Після цього SM585 автоматично вимикається.

API	Інструкція			Операнд								опис					
2720	D	TPWL	P	S_1, S_2, S_3, S_4								Встановлення параметрів лінійної інтерполяції в таблиці планування позиції					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●	●				O	O		
S_2								●	●				O	O		
S_3								●	●				O	O		
S_4								●	●				O	O		

Тип дані	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4			●				●						

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

DTPWL	DTPWLP
En	En
S1	S1
S2	S2
S3	S3
S4	S4

- S_1 : Число, указане в таблиці планування посадки
- S_2 : Цільове положення осі X
- S_3 : Цільове положення осі Y
- S_4 : Цільова швидкість

Пояснення

- Ця інструкція встановлює параметри 2-осьової лінійної інтерполяції в таблиці планування положення. S_1 – номер, зазначений у таблиці планування посадки. Якщо число не існує в таблиці або вихідне число не належить до 2-осьової лінійної інтерполяції, інструкція не виконується, SM0 увімкнено, а код помилки SR0=16#2027.

Single-axis point-to-point motion							
Single-axis multi-segment motion				2-axis linear interpolation		2-axis arc interpolation	
No.	Coord.	Axis No.	Output	ABS/REL mode	Target position	Target speed	ACC/DEC time
1	X	Axis 1	Pulse/Dir	Absolute	1000	50000	100
T1	Y	Axis 2	Pulse/Dir		-1000		

- S_2 і S_3 відповідно є цільовими положеннями осей X і Y, які можуть бути лише 32-розрядними цілими числами. Якщо ви використовуєте

перетворення механічних одиниць під час редагування таблиці планування позиції в ISPSOft спочатку

використовуйте інструкцію з перетворення для модифікації.

3. S_4 — цільова швидкість. Діапазон становить від 1 до 200 000 Гц. (Примітка: якщо значення налаштування виходить за межі діапазону, інструкція автоматично змінює налаштування на мінімальне або максимальне значення.)
4. Коли інструкція виконує лінійну інтерполяцію, цільова частота S_4 автоматично відповідає виходу осі, яка знаходиться найдалше від її цільового положення. Якщо вісь X і Y не можуть одночасно досягти цільових положень, ПЛК автоматично сповільнює частоту, щоб обидві осі досягли цільових положень одночасно.
5. Коли виконується інструкція для зміни параметрів для двох осей (кожна з яких виводиться), змінені параметри двох осей зберігаються в таблиці та не діють, доки не почнеться наступний вивід 2 осей.
6. Параметри, змінені інструкцією, можна змінювати лише під час роботи ПЛК. Останній записаний параметр не зберігається, коли живлення вимикається. Таблиця, яку ви редагуєте в ISPSOft і завантажуєте в ПЛК, обробляється як таблиця планування позицій за замовчуванням, коли живлення ON.
7. Щоб змінити час прискорення/уповільнення, вам слід змінити час прискорення/уповільнення в SR, якому відповідає вказана вихідна вісь осі X, встановити прапорець SM585 на ON, а потім виконати цю інструкцію. Після завершення зміни відповідних параметрів SM585 буде автоматично очищено та змінено на OFF.
- Наприклад, якщо вихідна вісь Y0.0 призначена для осі X, вам слід змінити значення в SR464 (час прискорення вихідного сигналу Y0.0), щоб змінити час прискорення/уповільнення. Щоб отримати докладні відомості про етапи роботи, зверніться до прикладу 1 інструкції TPWS.
- Примітка:
- Час прискорення та уповільнення призначено лише для роботи ПЛК із двоосьовим синхронізованим рухом. Якщо одночасне прибуття до цільових позицій не може бути досягнуто після розрахунку, ПЛК автоматично виконає інструкцію з найбільш правильним часом прискорення/уповільнення, і інформація про помилку не з'явиться.
 - Функція доступна для мікропрограми ПЛК версії V1.12.00 і пізніших.
8. Під час виконання інструкції TPO зверніть увагу на час зміни параметрів за допомогою цієї інструкції. Рекомендується уникати зміни використовуваного номера для результату, який виконується або буде виконано незабаром.

API	Інструкція				Операнд								опис				
2721	D	TPWC		P	S_1, S_2, S_3, S_4, S_5								Налаштування параметрів дугової інтерполяції в таблиці планування положення				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1								●	●				○	○			
S_2								●	●				○	○			
S_3								●	●				○	○			
S_4								●	●				○	○			
S_5								●	●				○	○			
Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1			●				●										
S_2			●				●										
S_3			●				●										
S_4			●				●										
S_5			●				●										

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
AS	-	AS

СИМВОЛ

DTPWC
En
S1
S2
S3
S4
S5

DTPWCP
En
S1
S2
S3
S4
S5

- S_1 : Число, указане в таблиці планування посадки
- S_2 : Цільове положення осі X
- S_3 : Цільове положення осі Y
- S_4 : Зміщення центру
- S_5 : Цільова швидкість

Пояснення

1. Ця інструкція встановлює параметри інтерполяції 2-осьової дуги в таблиці планування положення. S_1 – номер, зазначений у таблиці планування посадки. Якщо число не існує в таблиці або виведене число не належить до 2-осьової дугової інтерполяції, інструкція не виконується, SM0 увімкнено, а код помилки SR0=16#2027.

  							
Single-axis point-to-point motion		Single-axis multi-segment motion		2-axis linear interpolation		2-axis arc interpolation	
No.	Coord.	Axis No.	Output	ABS/REL mode	Target position	Target speed	Center shift value
1	X	Axis 1	Pulse/Dir	Absolute	100000	15000	-50000
T1	Y	Axis 2	Pulse/Dir		100000		

2. S_2 і S_3 є відповідно цільовим положенням на осях X і Y. S_4 - відстань до зміщення центру. Трійка параметри можуть бути лише 32-розрядними цілими числами. Якщо ви використовуєте перетворення механічних одиниць під час редагування таблиці планування позиції в ISPSOft, спершу використовуйте інструкцію з перетворення для модифікації.
3. S_5 — цільова швидкість. Діапазон становить від 1 до 200 000 Гц. Це опорна швидкість для фактичного виходу. ПЛК автоматично сповільнює швидкість, якщо неможливо досягти 2-осьового синхронізованого виходу.
4. Коли виконується інструкція для зміни параметрів для двох осей (кожна з яких виводиться), змінені параметри двох осей зберігаються в таблиці та не діють, доки не почнеться наступний вивід 2 осей.
5. Параметри, змінені інструкцією, можна змінювати лише під час роботи ПЛК. Останнє написане параметр не зберігається, коли живлення вимикається. Таблиця, яку ви редагуєте в ISPSOft і завантажуєте в ПЛК, обробляється як таблиця планування позицій за замовчуванням, коли живлення ON.
6. Ця інструкція не підтримує зміну напрямку руху за або проти годинникової стрілки. Для зміни напрямку, додайте або змініть вихідні параметри для дугової інтерполяції в таблиці планування положення в ISPSOft.
7. Під час виконання інструкції TPO зверніть увагу на час зміни параметрів за допомогою цієї інструкції. Рекомендується уникати зміни використовуваного номера для результату, який виконується або буде виконано незабаром.

API	Інструкція			Операнд								опис				
2723	D	ППГБ		$S_1, S_2, S_3, S_4, D_1, D_2$								Точка-точка перейти вперед і назад				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●								
S_2								●								
S_3								●								
S_4								●								
D_1		○														
D_2			●													

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						
S_4									●				
D_1	●												
D_2	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	-	AS

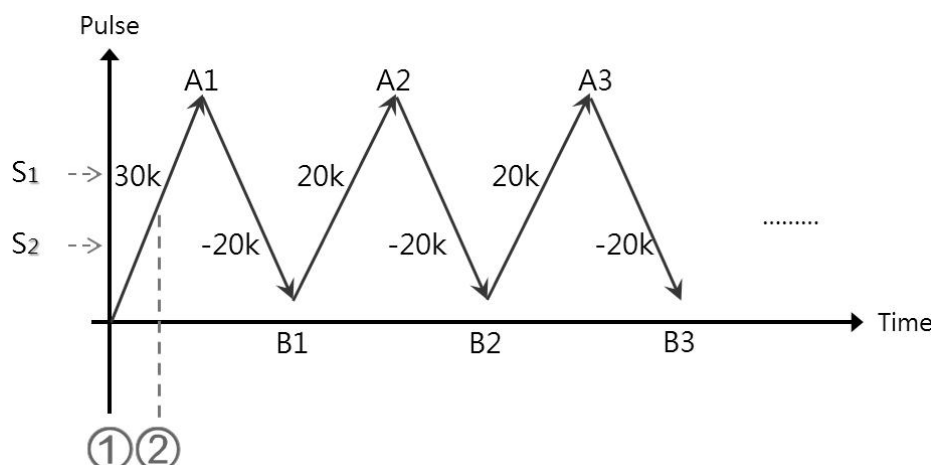
СИМВОЛ

DPPGB	
En	
S1	D1
S2	D2
S3	
S4	

- S_1 : Відносне цільове положення A
- S_2 : Відносна цільова позиція B
- S_3 : Цільова швидкість
- S_4 : Скориговане значення цільового співвідношення швидкості (значення з плаваючою комою)
- D_1 : Пристрій виведення імпульсів
- D_2 : Індикатор цільової швидкості може змінюватися

Пояснення

1. Ця інструкція високошвидкісного виводу визначає рух вперед і назад між двома цільовими позиціями, які перетворюються з двох відносних цільових положень, визначених S_1 і S_2 . Ця інструкція застосовна до таких машин, як деформаційні машини в текстильній промисловості та машини для намотування та зв'язування в кабельній промисловості.
2. Коли інструкція починає виведення, спочатку повинні бути вказані відносні позиції S_1 і S_2 , щоб інструкція може зробити розрахунок для переходу на наступний вихід. Після виконання інструкції ви можете змінити цільові позиції для виконання, але ви не можете змінити вихідні цільові позиції.



(1) Інструкція DPPGB виконана. Коли S_1 становить 30 000, а S_2 -30 000, рух йде до цільової позиції $A1=30\,000$.

(2) Якщо позиція імпульсу >15 000, відносні позиції змінюються на S_1 :20 000 і S_2 :-20 000, а цільове положення миттєво становить $A1=30\,000$. Цільові позиції, отримані шляхом розрахунку, $B1=10\,000 (A1+ S_2)$ і $A2=30\,000 (B1+ S_1)$, оскільки вихідні цільові позиції не можна змінити.

3. S_3 — цільова швидкість (має бути 32-розрядним цілим числом), а S_4 — скориговане значення цільового співвідношення швидкості (має бути плаваючим числом).

номер точки). Фактична швидкість є результатом множення значень у $S3$ та $S4$, а потім округлення результату обчислення до 32-розрядного цілого числа. Прийнятний діапазон вхідної швидкості 1–200 кГц. Коли значення виходять за межі діапазону, ПЛК автоматично обробляє значення як мінімальне або максимальне значення робочого діапазону. Формула співвідношення для фактичної цільової швидкості $S_3 \times S_4$. Наприклад, якщо цільова швидкість становить 1 кГц, скоригована плаваюча кома дорівнює 1,2345, а фактичне значення – 1234 Гц.

4. Коли інструкція виводиться, цільову швидкість і відкориговане співвідношення можна змінити, і результат є оновлюється до фактичної швидкості виведення після сканування інструкції. Але рекомендується не надто змінювати цільову швидкість, якщо це вплине на розраховане уповільнення.

5. Вихідними точками для D_1 можуть бути лише Y0.0, Y0.2, Y0.4, Y0.6, Y0.8 і Y0.10, і він займає 2 послідовні вихідні точки.

Набори вихідних даних і режими спеціальних регістрів вихідних сигналів перераховані в наступній таблиці.

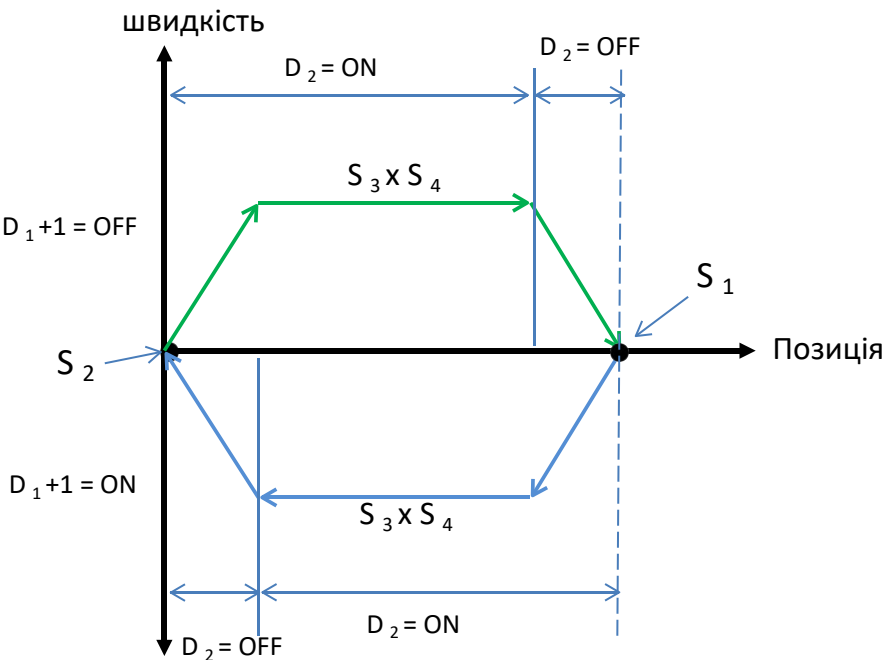
Номер вихідної осі	1	2	3	4	5	6
$D_1 +0$ вихідних балів	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
$D_1 +1$ вихідні точки	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Режими виведення	SR462	SR482	SR502	SR522	SR542	SR562

6. D_2 — це прапорець, що вказує на цільову швидкість і відкоригований коефіцієнт, який можна змінити. Коли цей прапорець увімкнено,

цільову швидкість виконання можна змінити. Коли прапорець перемикається з ON. на OFF., він уповільнюється, і поточна цільова швидкість буде оброблена як цільова швидкість для наступного виходу.

7. Немає обмежень на кількість разів, коли ви можете виконати цю інструкцію, але під час виконання призначена вісь високої швидкості не може бути зайнята жодною іншою інструкцією, а інша інструкція не виконується.

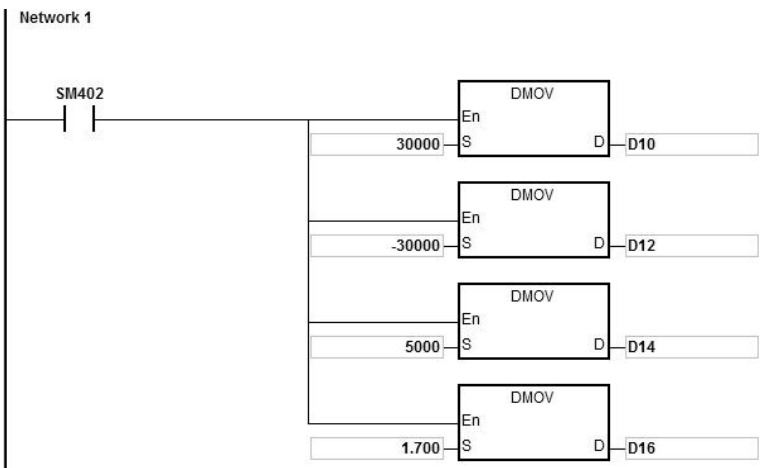
На наступному малюнку показано часову діаграму виведення.

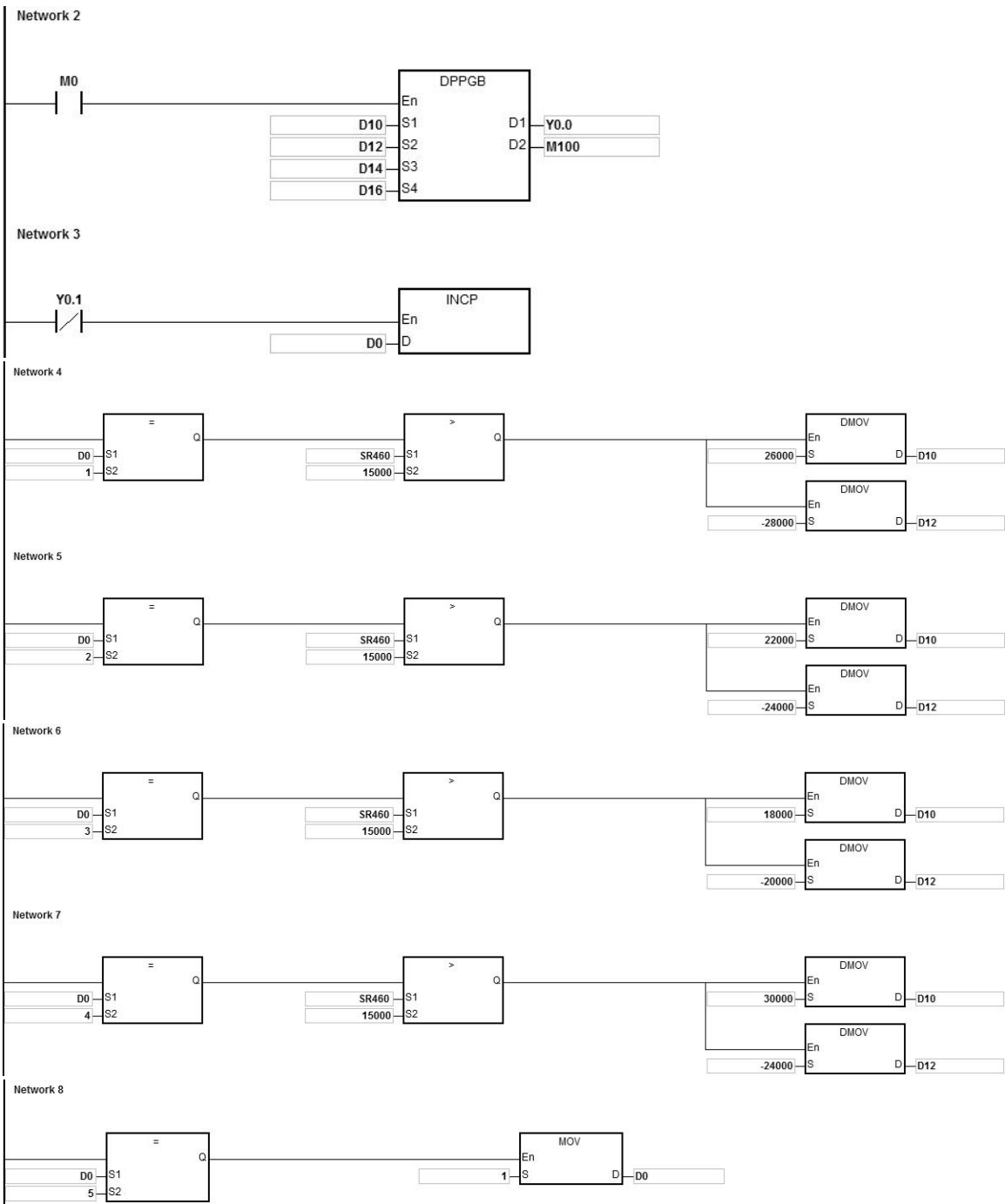


приклад

Коли M0 увімкнено, інструкція DPPGB починає рухатися вперед і назад між обчисленими цільовими позиціями від двох відносних цільових положень, визначених S_1 і S_2 , з фактичною цільовою швидкістю 8500 Гц ($5000 \times 1,7$).

Y0.0 — вихідна точка. Y0.1=OFF означає позитивний напрям.

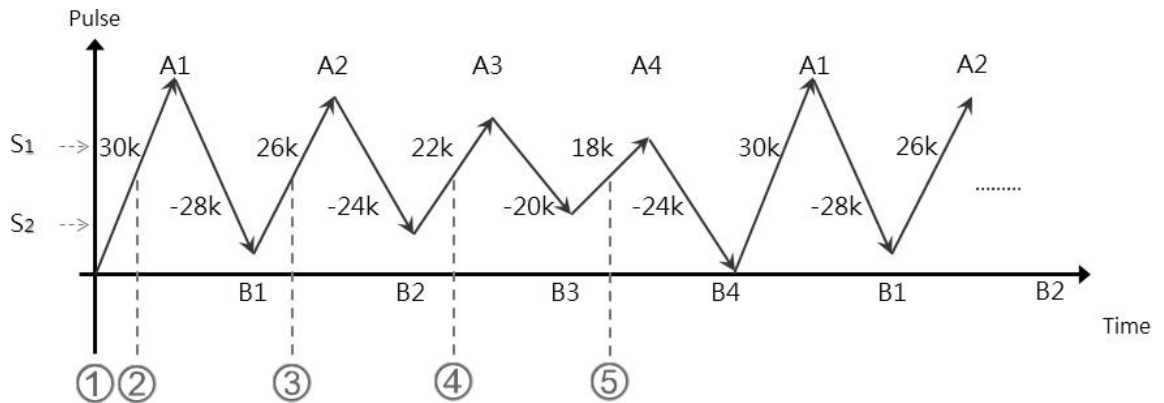




Пояснення

1. Після того, як інструкція почне виводити, змініть відносні цільові позиції А для S_1 і В для S_2 .

а) Вихідний графік Y0.0:



б) Пояснення:

(1) Коли S_1 :30 000, S_2 :-30 000 і M0 вмикає інструкцію DPPGB, рух йде до позиції A1=30 000.

(2) Коли D0=1 і позиція: кількість імпульсів у SR460 >15 000, відносні позиції змінюються на S_1 :26 000 і S_2 :-28 000.

На даний момент цільова позиція становить A1=30 000, оскільки вихідна цільова позиція не може бути змінена. The розраховані цільові позиції B1=2000 (A1-28000), A2=28000 (B1+26000).

(3) Коли D0=2 і позиція: кількість імпульсів у SR460 >15 000, відносні позиції змінюються на S_1 :22 000 і S_2 :-24 000.

Наразі цільова позиція становить A2=28 000, оскільки вихідна цільова позиція не може бути змінена.

Розраховані цільові позиції B2=4000 (A2-24000), A3=26000 (B2+22000).

(4) Коли D0=3 і позиція: кількість імпульсів у SR460 >15 000, відносні позиції змінюються на $S_{1:18}$:000 і S_2 :-20 000.

Розраховані цільові позиції B3=6000 (A3-20000), A4=24000 (B3+18000).

(5) Коли D0=4 і позиція: кількість імпульсів у SR460 >15 000, відносні позиції змінюються на S_1 :30 000 і S_2 :-24 000.

Розраховані цільові позиції B4=0(A4-24 000) і A1=30 000 (B4+30 000).

Крок	S_1 : Відносне положення	S_2 : Відносне положення	Цільова позиція B	Цільова позиція A
(1)	30 000	-30 000	-	A1=30 000
(2)	26 000	-28 000	B1=2000	A2=28 000
(3)	22 000	-24 000	B2=4000	A3=26 000
(4)	18 000	-20 000	B3=6000	A4=24 000
(5)	30 000	-24 000	B4=0	A1=30 000

API	Інструкція				Операнд								опис				
2724	D	ZRN2			TFreq, JFreq, Mode, DOG, NL, Pulse, Dir								Нульове повернення 2 (можна визначити спрямований вихід)				

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
TFreq							●	●	●		○		○	○		
JFreq							●	●	●		○		○	○		
Mode							●	●	●		○		○	○		
DOG	○		○													
NL	○		○													
Pulse		○														
Dir		○	○													

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
TFreq			●				●						
JFreq			●				●						
Mode			●				●						
DOG	●												
NL	●												
Pulse	●												
Dir	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



- TFreq : Цільова частота для нульової віддачі
- JFreq : Частота JOG для DOG
- Mode : Режим нульового повернення
- DOG : Пристрій введення для DOG
- NL : Пристрій введення для точки негативного обмеження
- Pulse : Пристрій виведення імпульсів
- Dir : Пристрій вихідного напрямку

Пояснення

- Ця інструкція доступна для ПЛК із FW V1.04.90 (V1.04.90 включено) або новішої версії.
- Ця інструкція повертає машину до нульової точки. Діапазон цільової частоти (**TFreq**) для нульового повернення становить від 1 Гц до 200 кГц. Частота JOG (**JFreq**) має бути меншою за цільову частоту (**TFreq**). Частота JOG (**JFreq**) є початковою частотою. Якщо цільова частота (**TFreq**) менша за (**JFreq**), цільова частота (**TFreq**) автоматично переглядається і обробляється як рівна (**JFreq**).
- Не змінюйте операнди **DOG**, **NL**, **Pulse** або **Dir** під час виконання інструкції. Для точки введення для **DOG** і **NL** пропонується використовувати 16 високошвидкісних точок X0.0–X0.15. На них не вплине час сканування інструкцій ПЛК. Якщо ви використовуєте послідовні точки введення X1.0 або пристрої M, на них впливатиме час сканування інструкцій ПЛК.
- Дивіться наведені нижче описи для **DOG**, **NL** та **Dir**.

DOG точка	Ви можете використовувати пристрій X або M, але НЕ вибирайте ту саму точку введення для різних осей. Коли точка DOG використовує X0.0–X0.15 для входів, перейдіть до HWCONFIG, щоб налаштувати час фільтрації для входів, якщо перемикач починає деренчати.
NL точка	Ви можете використовувати пристрій X або M, але НЕ вибирайте ту саму точку введення для різних осей.
Dir контакт	Ви можете використовувати пристрій Y або M.

Для мікропрограми V1.10 або ранішої версії, коли виконується повернення на нуль і активована функція перевірки граничної точки,

прапор граничної тривоги не встановлюється на ON після досягнення граничної точки. Для мікропрограми V1.12 або пізнішої версії прапорець тривоги обмеження буде автоматично увімкнено для нагадування, а потім ви зможете вирішити подальшу операцію на основі позначки тривоги. (Після увімкнення прапорця тривоги його потрібно скинути самостійно.)

Примітка: DZRN2 не підтримує функцію позитивного обмеження.

- Дивіться наведену нижче таблицю SR і SM для виведення імпульсів.

Точки виходу імпульсу	Y0.0	Y0.2	Y0.4	Y0.6	Y0.8	Y0.10
Встановлення часу для спрямоване виведення йде спочатку перед виведенням імпульсу (одиниця: 1 мс)	SR640	SR642	SR644	SR646	SR648	SR650
Зайнятий прапор	SM460	SM480	SM500	SM520	SM540	SM560
Прапор завершення	SM461	SM481	SM501	SM521	SM541	SM561
Поточна вихідна позиція	SR460	SR480	SR500	SR520	SR540	SR560
	SR461	SR481	SR501	SR521	SR541	SR561

Точки виходу імпульсу	Y0.1	Y0.3	Y0.5	Y0.7	Y0.9	Y0.11
Встановлення часу для спрямоване виведення йде спочатку перед виведенням імпульсу (одиниця: 1 мс)	SR641	SR643	SR645	SR647	SR649	SR651
Зайнятий прапор	SM472	SM492	SM512	SM532	SM552	SM572
Прапор завершення	SM473	SM493	SM513	SM533	SM553	SM573
Поточна вихідна позиція	SR474	SR494	SR514	SR534	SR554	SR574
	SR475	SR495	SR515	SR535	SR555	SR575

6. Виберіть режим повернення нуля. Код функції встановлюється двома старшими та нижчими 16-бітними параметрами. Дивіться наступне

таблиця для деталей.

Виберіть режим повернення на нуль					
Високий 16-біт	Низький 16-біт				
b31~b16	b15~b6	b6	b5	b4	b3~b0
Кількість імпульсів для руху	зарезервований	Налаштування DOG маска сигналу 0: вимкнено 1: увімкнено	Установка NL сигнал 0: контакт А 1: контакт Б	Налаштування DOG сигнал 0: контакт А 1: контакт Б	Налаштування режиму 0~2; 8~10

7. Код функції встановлюється двома 16-бітними параметрами старшого та нижчого рівнів. Подробиці див. у наступній таблиці.

Функції	Код		опис
	Високий 16-біт	Низький 16-біт	
Покидає нульову точку в негативному напрямку, а потім зупиняється (Mode 0)	0	0	Коли інструкція виконується, пошук нульової точки відбувається в негативному напрямку з цільовою частотою. Коли нульова точка увімкнена (сигнал нульової точки змінюється з OFF. на ON.), частота зменшується до швидкості JOG, а рух у негативному напрямку продовжується та не зупиняється, доки сигнал нульової точки не зміниться з ON. на OFF.
Покидає нульову точку в позитивному напрямку, а потім зупиняється (Mode 1)	0	1	Коли інструкція виконується, пошук нульової точки відбувається в негативному напрямку з цільовою частотою. Коли нульова точка увімкнена (сигнал нульової точки змінюється з OFF. на ON.), частота негайно зменшується до 0, а потім рух відбувається в позитивному напрямку зі швидкістю JOG і не зупиняється, доки сигнал нульової точки не зміниться з ON до OFF.
Mode 0 Знову рухається	Число	2	Повернення до нульової точки таке ж, як і для

Функції	Код		опис
	Високий 16-біт	Низький 16-біт	
після повернення до нульової точки	імпульси для руху		низький 16-бітний код. Після ввімкнення нульової точки рух продовжується відповідно до кількості заданих імпульсів. Коли старший 16-бітний код є позитивним числом, пошук відбувається в позитивному напрямку. Від'ємне значення означає, що пошук здійснюється в негативному напрямку.
Покидає нульову точку в позитивному напрямку, а потім зупиняється (Mode 1)	0	8+0=8 8+1=9 (bit3=ON)	Зверніться до режиму 1. Операція для повернення нульової точки така ж, як і для коду 1 (режим 1).
Mode 1 видає кількість імпульсів після повернення до нульової точки	Кількість імпульсів для руху	8+2=10 (bit3=ON)	Операція повернення нульової точки така ж, як і для молодшого 16-бітного коду 1. Після повернення в нульову точку рух продовжується відповідно до кількості заданих імпульсів. Коли значення старшого 16-бітного коду є позитивним числом, рух відбувається в позитивному напрямку. Від'ємне число означає, що рух відбувається у негативному напрямку.
Точка DOG є точкою В		+16 (bit4=ON)	Коли в молодшому 16-бітному коді біт 4 увімкнено, це означає, що нульова точка увімкнена, коли точка DOG змінюється з ON на OFF, а нульова точка залишається, коли точка DOG змінюється з OFF на ON.
Точка NL є точкою В		+32 (біт 5=ON.)	Коли в молодшому 16-бітному коді біт 5 увімкнено, це означає, що нульова точка увімкнена, коли точка NL змінюється з ON. на OFF., а нульова точка залишається, коли точка DOG змінюється з OFF. на ON.

8. Послідовність виконання базується на значенні молодшого 16-бітного коду в таблиці вище та описано нижче.

- Напрямок сигналу DOG визначається значенням біта 5.
- Негативний ліміт NL і сигнал DOG визначаються значенням біта 5 і біта 4.
- Mode 0 або режим 1 для повернення нульової точки, вибирається відповідно до значення біта 3.
- Операція повернення нульової точки виконується відповідно до значень біта 1 і біта 0.

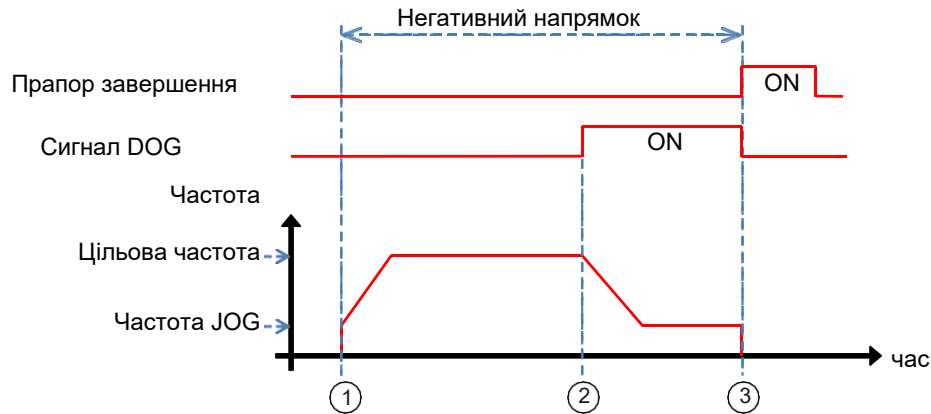
Функція сигнальної маски DOG доступна лише для мікропрограми V1.12. або пізніше. Коли функцію маскування ввімкнено (встановлення біта 6 на 1), **Mode+1** – це кількість сигналів DOG, які потрібно маскувати. Якщо значення менше або дорівнює 0, функція маски точки DOG вважається вимкненою.

Ділянка завдання визначається кількістю вихідних імпульсів (незалежно від позитивного чи негативного напрямку) після виконання інструкції DZR2. Оскільки початкова позиція для пошуку нульової позиції може щоразу відрізнятися, налаштування секції маски може призвести до того, що нульова позиція буде пропущена. Тому обов'язково оцініть, чи встановити функцію маски чи встановити функцію від'ємного обмеження на основі фактичного застосування.

Наприклад, встановіть **режим** D0=64 (біт 6 встановлений на 1) і встановіть D2=120000 (кількість імпульсів, які потрібно маскувати). Після виконання DZR2 всі сигнали DOG для кількості вихідних імпульсів менше 120 000 будуть замасковані, і, отже, сигнали DOG не працюють.

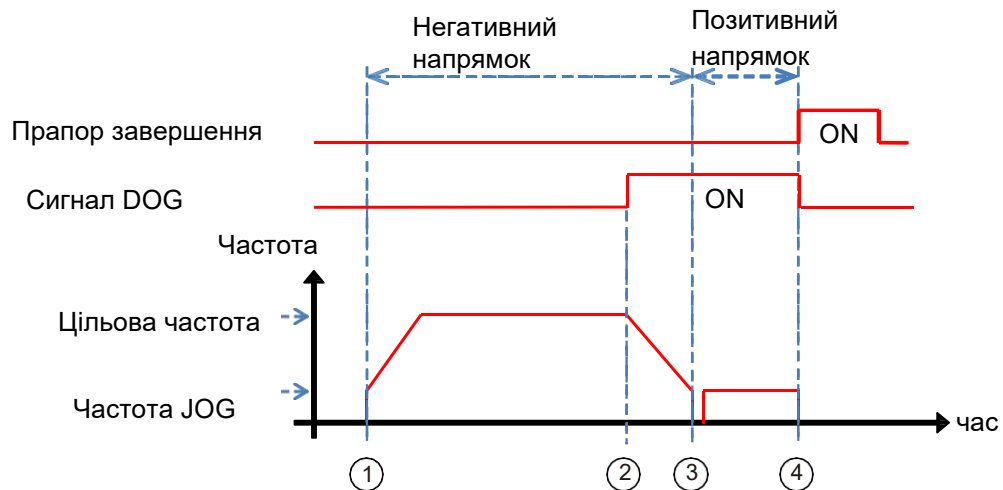
9. Прапор завершення встановлюється на ON після того, як інструкція завершує виконання вказаної функції. Наприклад, для функціонального коду 2 ПЛК встановлює прапор завершення на ON лише тоді, коли завершено виведення вказаної кількості імпульсів.
10. Після виконання інструкції DZRN2 програма обслуговування переривання не виконується, доки інструкція DZRN2 не буде вимкнена, якщо вказана точка введення для нульової точки збігається з точкою зовнішнього введення в програмі.
11. Не рекомендується використовувати інструкцію на мові програмування ST, завдання переривання або функціональний блок, який викликається лише один раз. Прошивка версії 1.12 або новішої підтримує мову програмування ST. Для отримання детальної інформації зверніться до опису в прикладі.
12. Кроки для виконання функцій наведені нижче, код функції 0:

- (1) Функція DZRN2 виконується, і пошук нульової точки відбувається в негативному напрямку з цільовою частотою (**TFreq**).
- (2) Після отримання сигналу DOG вихідна частота зменшується до частоти JOG (**JFreq**). Вихід продовжує працювати в негативному напрямку і не припиняється, доки сигнал нульової точки не зміниться з ON. на OFF.
- (3) Вихід припиняється, коли сигнал змінюється з ON. на OFF., а вісь віддаляється від сигналу DOG.



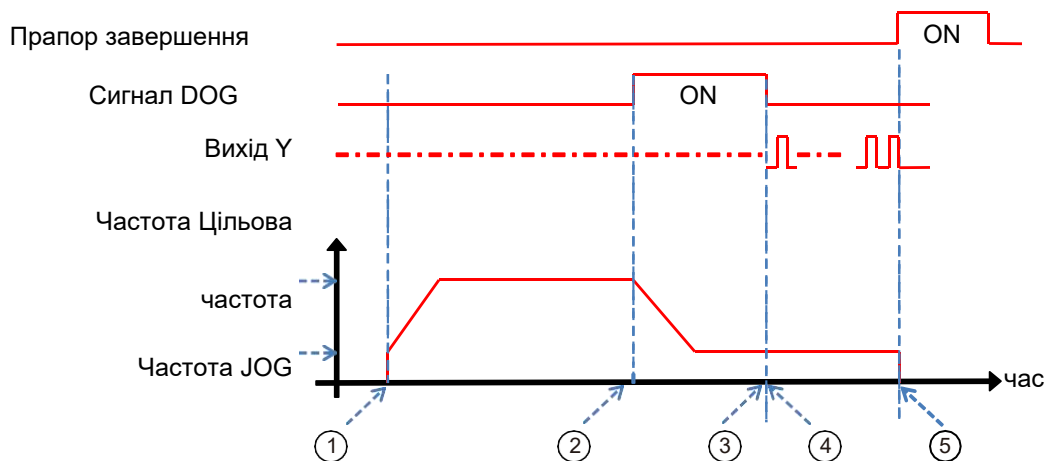
Код функції 1:

- (1) Функція DZRN2 виконується, і пошук нульової точки відбувається в негативному напрямку з цільовою частотою (**TFreq**).
- (2) Після отримання сигналу DOG вихідний сигнал має позитивний напрямок із частотою JOG (**JFreq**), після того як вихідна частота зменшується, а напрямок руху змінюється на зворотний. Вихід не припиняється, доки сигнал нульової точки не зміниться з ON. на OFF.
- (3) Вісь відходить від сигналу DOG, і ПЛК зупиняється, коли сигнал змінюється з ON. на OFF.



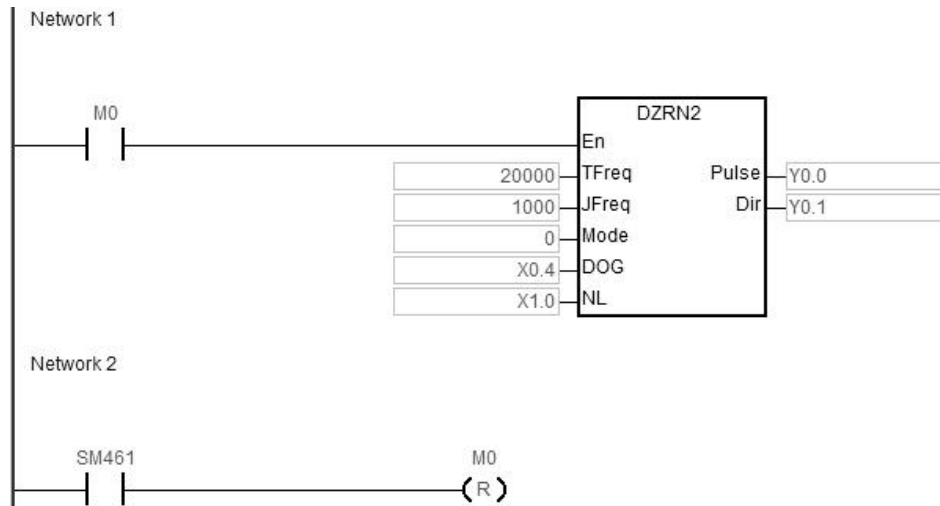
Код функції 2:

- (1) Функція DZRN2 виконується, і пошук нульової точки відбувається в негативному напрямку з цільовою частотою (**TFreq**).
- (2) Після отримання сигналу DOG вихідний сигнал зменшує частоту до частоти JOG (**JFreq**) і продовжує рух у негативному напрямку.
- (3) Коли сигнал DOG залишається, і сигнал змінюється з ON на OFF, виводиться задана кількість імпульсів.
- (4) Починається вихід першого імпульсу.
- (5) Після завершення виведення 100-го імпульсу ПЛК зупиняється, а прапор завершення вмикається.



Приклад 1

Коли M0 увімкнено, виведення імпульсу з Y0.0 з частотою 20 кГц для пошуку нульової точки в негативному напрямку. Коли отримано сигнал DOG і X0.4 увімкнено, він продовжує рухатися в негативному напрямку з частотою JOG 1 кГц. Вихід припиняється відразу після зміни X0.4 з ON на OFF.

**Пояснення**

Режим встановлено на 0.

Старший 16-бітний [0000] вимикає функцію переміщення кількості імпульсів.

Нижчий 16-бітний [0000] — це режим 0; при отриманні сигналу DOG вісь рухається в негативному напрямку; після того, як вісь відійде від сигналу DOG, він негайно зупиниться.

Приклад 2 (мовою програмування ST)

Загалом, під час використання цієї інструкції для редагування програми пропонується використовувати мову сходової діаграми. Якщо ви використовуєте мову ST (яка доступна для мікропрограми версії 1.12 або пізнішої), ви повинні дочекатися завершення руху повернення до нуля (прапор завершення SM461 увімкнено), а потім вимкнути інструкцію. Крім того, ви повинні використовувати прапор автоматичного скидання разом, щоб повідомити ПЛК про звільнення права керування високошвидкісним виходом.

Взявши, наприклад, вихід Y0.0, написання програми показано на малюнку нижче

```

0001 IF M0 THEN
0002     DZRN2 (1000,100,0,X0.0,X1.0,Y0.0,Y0.1);
0003     M1:=TRUE;
0004 ELSIF M1 AND SM461 THEN
0005     SM470 := TRUE;
0006     M1 := FALSE;
0007 END_IF;

```

6.27 Інструкції щодо зв'язку Delta CANopen та EtherCAT

6.27.1 Перелік інструкцій щодо спілкування

У наведеній нижче таблиці наведено інструкції щодо високошвидкісного виведення, розглянуті в цьому розділі.

API	Код інструкції		Інструкція пульсу	функція	Особливість дельти		EtherCAT	
	16-бітний	32-розрядний			Сервопривод	Інвертор	Сервопривод	Інвертор
2800	INIC	–	–	Ініціалізація сервоприводу для CANopen спілкування	V	V	–	–
2801	ASDON	–	–	Servo-ON і Servo-OFF	V	V	V	V
2802	CASD	–	–	Встановлення часу розгону і час уповільнення для сервоприводу	V	V	V	V
2803	–	DDRVIC	–	Контроль відносного положення сервоприводу	V	–	V	–
2804	–	DDRVAC	–	Абсолютний контроль положення сервоприводу	V	–	V	–
2805	–	DPLSVC	–	Серворегулювання швидкості	V	V	V	V
2806	ZRNC	DZRNC	–	Сервонаведення	V	–	V	–
2807	COPRW	–	–	Читання та запис CANopen комунікаційні дані	V	V	–	–
2808	COPWL	DCOPWL	–	Запис кількох параметрів CANopen значення	V	V	–	–
2809	RSTD	–	–	Надсилання команди Reset або NMT	V	V	V	V
2810	ZRNM	–	–	Налаштування режиму наведення для сервоприводу Delta	V	–	V	–
2811	EMER	–	–	Читання екстреного повідомлення	V	V	V	V
2812	–	DCSFOC	–	Керування функцією відстеження сервоприводу через зв'язок	V	–	V	–
2813	ECAMD	–	–	Завантаження діаграми E-CAM на Slave	V	–	–	–
2814	–	DECAMS	–	Налаштування та планування параметрів E-CAM	V	–	–	–
2815	ECAMA	–	–	Активізація керування та відображення E-CAM статус	V	–	–	–
2816	ECAMC	–	–	Зміна початкової адреси для E-CAM діаграми та довжини діаграми E-CAM у сервоприводі	V	–	–	–
2817	–	DTQC	–	Контроль крутного моменту	V	V	V	–
2818	–	DTQLC	–	Швидкість регулювання крутного моменту	V	V	V	–

API	Код інструкції		Інструкція пульсу	функція	Особливість дельти		EtherCAT	
	16-бітний	32- розрядний			Сервопр ивод	Інвертор	Сервопр ивод	Інвертор
<u>2820</u>	INITEC	–	–	Ініціалізація підлеглих пристроїв для зв'язку EtherCAT	–	–	V	V
<u>2821</u>	ECATRW		–	Читання та запис даних EtherCAT SDO	–	–	V	V
<u>2822</u>	MOTST	–	–	Читання стану руху підлеглого EtherCAT	–	–	V	V
<u>2823</u>	ECATST	–	–	Читання стану підлеглого EtherCAT	–	–	V	V

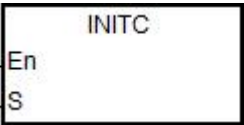
6.27.2 Пояснення інструкцій щодо зв'язку CANopen та EtherCAT

API	Інструкція				Операнд						опис						
2800		INIC			S						Ініціалізація сервоприводу для зв'язку CANopen						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S													O	O			

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S		●											

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ



S : Номер станції для ініціалізації

Пояснення

- Перед виконанням інструкції, якщо процесором ПЛК є AS300, обов'язково виберіть **Delta Special Driver & AS Remote Mode** як робочий режим для функціональної карти 2 у програмному забезпеченні конфігурації обладнання (HWCONFIG) у ISPSOft. Якщо процесором ПЛК є AS200 або AS100, переконайтеся, що режим роботи для вбудованого порту зв'язку CAN вибрано Delta Serve Drives у HWCONFIG.
- Діапазон сервоприводу **S** становить 1–8. Коли вхідне значення перевищує 8, ПЛК автоматично обробить 8 як значення **S** для ініціалізації. Ідентифікатор вузла має починатися з 1, а наступні адреси не можна пропускати або резервувати.
- Якщо програмне забезпечення ПЛК версії 1.02.00 або новішої, для інверторів додається **S у діапазоні від 21 до 28**. Для вхідного значення, яке не знаходиться в діапазоні від 21 до 28, ця інструкція не буде виконана. Те саме правило стосується сервоприводу; ідентифікатор вузла має починатися з 21, а наступна адреса має бути в послідовному порядку.
- Якщо встановлювати номер станції, яку потрібно ініціалізувати, рівним 5, ця інструкція ініціалізує ідентифікатор вузла від 1 до 5. А якщо встановлювати номер станції, яку потрібно ініціалізувати, рівним 23, ця інструкція ініціалізує ідентифікатор вузла від 21 до 23. Якщо є 5 сервоприводів систем і 3 інверторів, які потрібно ініціалізувати, цю інструкцію потрібно виконати двічі. Один — INIC 5 для 5 сервосистем, а інший — INIC 23 для 3 інверторів.
- Коли ви виконуєте цю інструкцію (SM1686 = OFF.) на сервоприводах (**S** дорівнює 1–8), SM1681 встановлюється на OFF., а стани сервоприводів 1–8 скидаються до неініціалізованих. SM1681 встановлюється на ON, коли ініціалізація на сервоприводах 1-8 завершена.

SM1683 встановлено на OFF, а **S** становить 21–28, стани сервоприводів 21–28 скидаються до неініціалізованих.

SM1683 встановлено на ON, коли ініціалізація на сервоприводах 21-28 завершена.

Примітка. Прапорці ініціалізації використовуються для сповіщення про те, виконано чи ні ініціалізацію. Після цього, якщо ви використовуєте параметри в цій інструкції, ці два прапорці використовуються як прапори завершення для виконання інструкції.

6. SM1682 встановлюється на ON, коли під час зв'язку виникає помилка. Крім того, SR658 зберігає ідентифікаційний номер, у якому сталася помилка, а SR659 зберігає код помилки. (Прапорець і код тут спеціально для інструкцій із зв'язку Delta CANOpen.)

7. Для мікропрограми V1.06.40 або новішої версії додається новий функціональний SM1686 для ініціалізації лише одного зазначеного підлеглого пристрою. Наприклад, якщо ви встановите для SM1686 значення OFF і виконаєте INITC 3, це означає, що підлеглі 1, 2 і 3 будуть ініціалізовані. Після цього, якщо ви встановите для SM1686 значення ON, виконання INITC 6 вказує, що буде ініціалізовано лише підлеглий пристрій 6. І тоді разом у вас будуть ініціалізовані підлеглі 1, 2, 3 і 6.
 Примітка: якщо вам потрібно ініціалізувати кілька підлеглих пристроїв і один підлеглий пристрій, вам потрібно спочатку встановити SM1686 на OFF., щоб ініціалізувати кілька підлеглих пристроїв, а потім встановити SM1686 на ON. для однієї ініціалізації, але не навпаки.

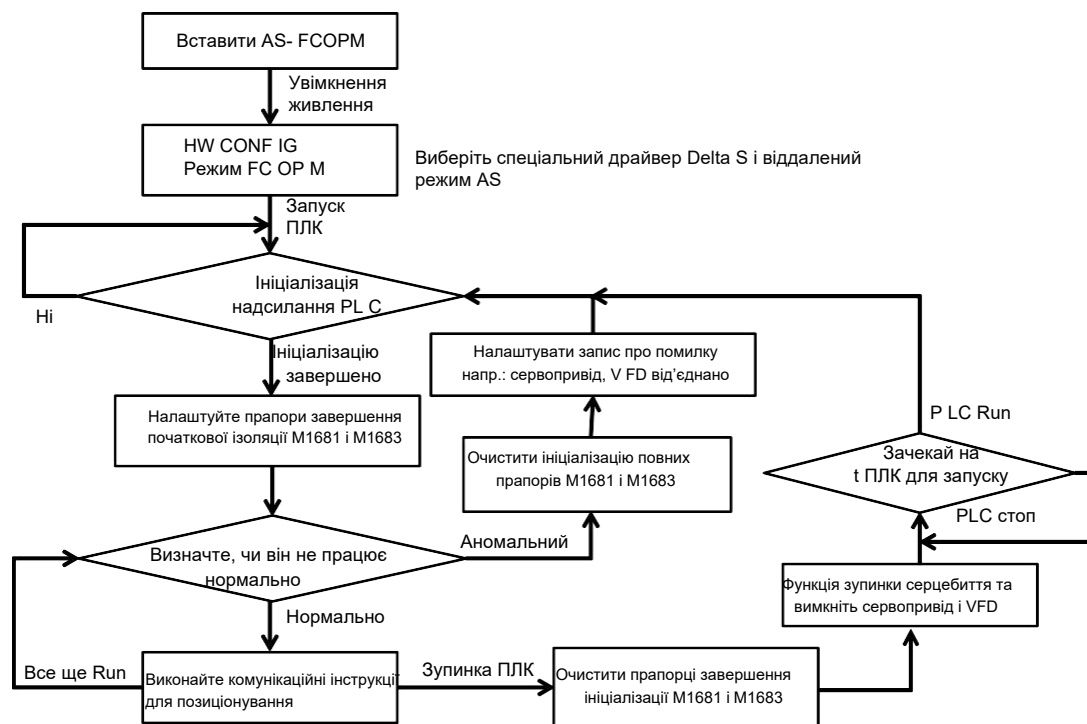
8. Якщо помилка серцевого ритму (SR0 = 0x19 00 ~0x19 1C; останні 2 коди є підлеглим ідентифікатором) виникає після завершення ініціалізації та SM1684 OFF (за замовчуванням, вказуючи, що коли один виходить з ладу, усі сервоприводи OFF), ініціалізація повні прапори SM1681 і SM1683 буде скинуто на OFF., а пов'язані дії на інших підлеглих також буде призупинено. Після усунення всіх несправностей вам потрібно ініціалізувати кожне підпорядковане пристрій, щоб перезапустити роботу. ПЛК вимкне індикатор помилки після того, як ПЛК підтвердить завершення ініціалізації. Ця помилка буде зареєстрована в журналі помилок.

9. Якщо кожен ідентифікатор вузла працює незалежно і зв'язок працює належним чином, ви можете налаштувати SM1684 на ON (вказує, що коли один з них вимикається, лише несправний сервопривід OFF), щоб повідомити ПЛК про запис конкретної помилки в журналі помилок, а інші підлеглі пристрої можуть продовжувати працювати.

10. Вам потрібно вручну встановити ідентифікатор вузла та швидкість сервоприводів, вимкнути та знову увімкнути сервоприводи.
 активуйте нові параметри.

11. Цю інструкцію не слід використовувати в мові програмування структурованих текстів або в програмах переривання чи програмах FB, які викликаються лише один раз.

Схема процесу ініціалізації та роботи



Приклад зв'язку з сервоприводом Delta ASD-A2M

- Перед використанням інструкції Delta CANopen встановіть плату AS-FCOPM у ПЛК серії AS (тільки слот 2) і обов'язково виберіть **Delta Special Driver & AS Remote Mode** як робочий режим для карти AS-FCOPM 2 у конфігураційному програмному забезпеченні (HWCONFIG в ISPSoft). Встановіть швидкість передачі даних на 125 Кбіт/с, 250 Кбіт/с, 500 Кбіт/с або 1000 Кбіт/с.

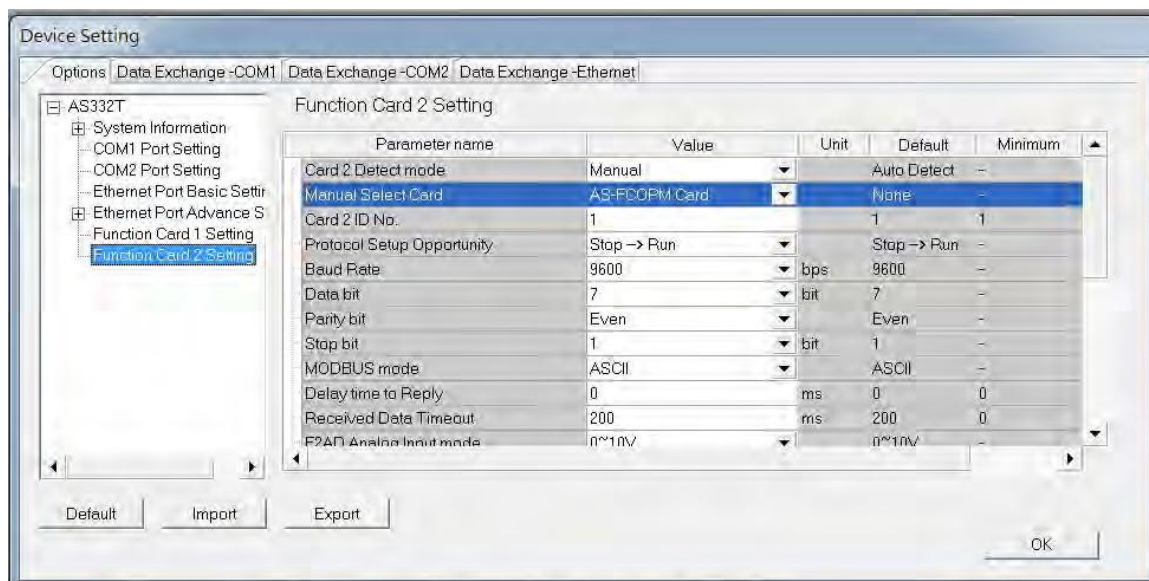


Рисунок 1 Налаштування функціональної картки 2

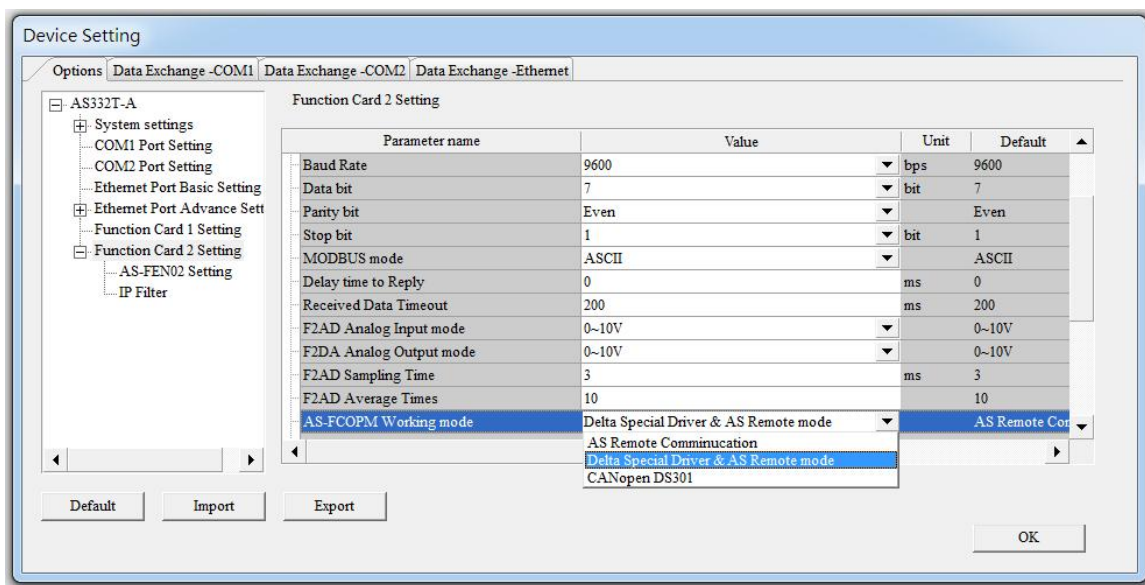
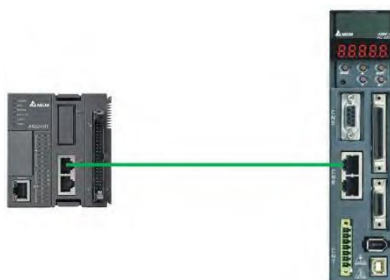


Рисунок 2 Вибір режиму роботи

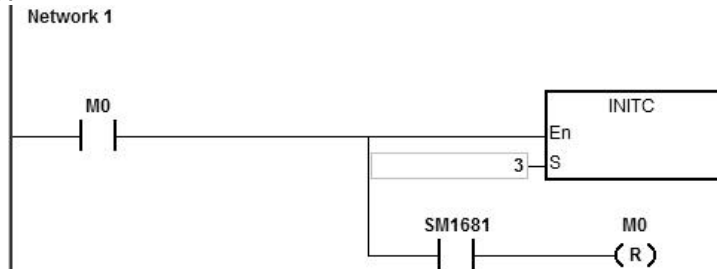
2. Під'єднайте ПЛК серії AS до серії ASDA за допомогою кабелю зв'язку CANopen, як показано на малюнку нижче. (Змініть 120 Ω на платі AS-FCOPM, щоб увімкнути кінцевий резистор.)



3. Виконайте наведені нижче кроки для основних налаштувань на панелі ASD-A2.
- Встановіть параметр сервоприводу P2-08 на 10, щоб відновити заводські налаштування.
 - Вимкніть і знову увімкніть сервопривід.
 - Встановіть P1-01 на 0001 (режим PR).
 - Встановіть P3-01 на 0400 і швидкість передачі сервоприводу для зв'язку CANopen на 1,0 Мбіт/с. Швидкість передачі даних має бути такою ж, як у ПЛК.
 - Встановіть ідентифікатор вузла для кожного сервоприводу на основі кількості сервоприводів. Встановіть P3-00 кожного сервоприводу на 1, 2 і 3 по порядку. Ви можете встановити максимум вісім сервоприводів.
 - Вимкніть і знову увімкніть сервопривід.
 - Почніть роботу після завершення основного налаштування.

4. Завантажте зразок програми та встановіть для M0 значення ON (за замовчуванням: SM1686 = OFF). Ця інструкція ініціалізує сервоприводи на вузлі ID 1–3. Коли SM1681 увімкнено, ініціалізація завершена. Коли сервопривід успішно переходить у режим CANopen, відображається інформація Co-Ld.

Схорова схема:



Мова програмування структурованих

```

0001 IF M0 = TRUE AND SM1681 = FALSE THEN
0002     INITC (3) ;
0003 END_IF;
0004 IF M0 = TRUE AND SM1681 = TRUE THEN
0005     M0 := FALSE;
0006 END IF;
    
```

5. У наведеному нижче списку показано налаштування для ініціалізації сервоприводу серії AS.

- a. Встановіть P2-30 (допоміжна функція) на 5, щоб вказати, що сервоприводу не потрібно постійно зберігати налаштування в EEPROM. Це може продовжити термін служби сервоприводу.
- b. Скиньте P6-02 (PATH#1) на 0 і P6-06 (PATH#3) на 0. Це означає, що PATH#1 і #3 в режимі PR очищено.
- c. Встановіть P3-06 (джерело SDI) на 16#0100. Це вказує на те, що DI1–DI8 контролюються апаратним забезпеченням, EDI9 – програмним забезпеченням, а EDI10–EDI14 – апаратним забезпеченням.
- d. Скиньте P4-07 (статус SDI контролюється вручну) до 0.
- e. Встановіть P2-36 (EDI9) на 16#0101. Це вказує на те, що функція EDI9 встановлена на Servo ON.
- f. Встановіть P0-17 (CM1A) на 1. Це вказує на те, що параметром відображення є вихідний регістр імпульсної команди CMD_O.
- g. Встановіть P0-18 (CM2A) на 64. Це вказує на те, що параметр відображення є регістром імпульсних команд CMD_E.
- h. Встановіть P5-20–P5-35 (час прискорення) на 1. Це означає, що час прискорення становить 1 мс.
- i. Встановіть P5-60–P5-75 (цільова швидкість) на 1. Це означає, що цільова швидкість становить 0,1 об/хв.
- j. Встановіть PDO1 відповідно до P5-07 (команда PR), P0-01 (код помилки), P0-46 (стан точки DO) і P4-07 (стан точки DI).

k. Встановіть PDO2 відповідно до P0-09 (стан CM1: CMD_O) і P0-10 (стан CM2: CMD_E).

l. Встановіть PDO3 відповідно до P0-11 (стан CM3: стан поточного крутного моменту)

m. Доступно для ПЛК із програмною програмою версії 1.04.00 або пізнішої: установіть час для пульсу та надлишкового сканування ПЛК на 200 мс (за замовчуванням). У разі переривання зв'язку ПЛК надішле повідомлення про помилку (світлодіодний індикатор помилки повільно блимає; подробиці див. у коді помилки), і сервопривід автоматично вимкнеться.

n. Якщо SM1685 увімкнено, PDO3 зчитує самовизначені параметри з сервоприводу P0-12 і зберігає дані в SR791-SR806. Перед виконанням потрібно налаштувати P0-20 (CM4A). Для кожного ідентифікатора сервоприводу зарезервовано 2 SR, і ви можете визначити їхні типи даних.

Примітка: ПЛК із мікропрограмою версії 1.04.10 або новішої підтримує стани поточного крутного моменту (PD03) і точки DI (PD01). ПЛК із вбудованим програмним забезпеченням версії 104.60 або новішої підтримує читання самовизначених параметрів (PD03)

6. Не використовуйте інструкцію COPRW (API 2807) для зміни параметрів сервоприводу восьми елементів A, B, F, G, J, K і N після ініціалізації.

7. Якщо ви використовуєте сервопривід абсолютного типу, використовуйте інструкцію зв'язку COPRW для запису 16#0100 у P3-12, який записує відповідні параметри сервоприводу абсолютного типу в EEPROM у момент вимкнення сервоприводу.

8. Встановіть відповідні параметри конфігурації сигналу DI вручну або за допомогою інструкції COPRW, щоб змінити параметри апаратного сигналу DI ASD-A2. Використовуйте COPRW, щоб змінити конфігурацію після завершення виконання інструкції INITC і до ввімкнення сервоприводу.

9. Коли ініціалізація завершена, сервопривід знаходиться в режимі PR. Не керуйте зв'язком на сервоприводі P5-18.

10. Додаткову інформацію про параметри сервоприводу див. у посібнику з експлуатації Delta Servo.

Приклад зв'язку з Delta Inverter

1. Вручну встановіть параметр інвертора P09-36 (від 21 до 28), P09-37 (швидкість передачі даних за замовчуванням становить 1 МГц) і P09-40 на стандартний протокол CANopen (за замовчуванням).

2. Вручну встановіть джерело команди частоти P00-20 на 6, а джерело команди роботи P00-21 на 3 та керуйте ними за допомогою зв'язку CANopen.

3. У наведеному нижче списку показано налаштування для ініціалізації інвертора серії AS.

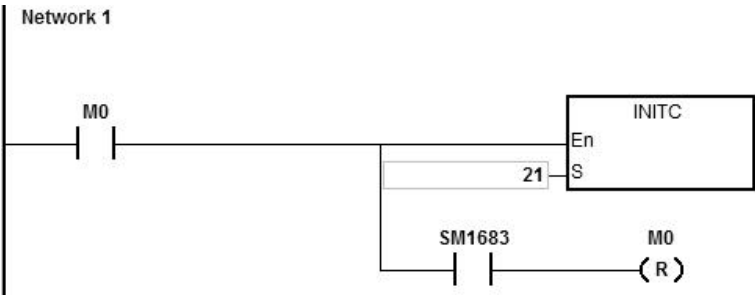
A. Встановіть область відображення PDO: робочі частоти, поточний стан і стани крутного моменту та точки DI відповідають SR PLC.

B. Встановіть час для серцевого ритму та додаткового часу сканування ПЛК на 200 мс (за замовчуванням). При спілкуванні понаднормово

ПЛК надішле повідомлення про помилку (світлодіод помилки повільно блимає; подробиці див. у коді помилки), і сервопривід автоматично вимкнеться.

4. Додаткову інформацію про параметри сервоприводу див. у посібнику з експлуатації дельта-інвертора.

приклад



1. Коли M0 змінюється з OFF на ON, інструкція INITC починає ініціалізувати інвертор на вузлі ID 21, доки SM1683 увімкнено. Коли SM1683 увімкнено, це означає, що ініціалізацію завершено.

Додаткові зауваження:

У наступній таблиці показано спеціальні прапори (SM) і регістри (SR), пов'язані з ініціалізацією.

Прапор	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
Увімкнути певну функцію	R/W	Прапор у HWCONFIG встановлюється на ON, коли ПЛК переходить від зупинки до роботи.							
Ініціалізація та зв'язок завершені (INITC і CASD)	R	SM1681							
Помилка зв'язку CANopen	R	SM1682							
Ініціалізація одного веденого пристрою (для прошивки V1.06.40 або пізнішої версії)	R/W	SM1686 = OFF. (за замовчуванням; ініціалізація кількох підлеглих пристроїв) SM1686 = ON. (ініціалізація одного підлеглого пристрою)							
Ідентифікаційний номер із помилкою зв'язку	R	SR658							
Код помилки зв'язку	R	SR659							

Примітка 1: після того, як вісь починає виводити, ПЛК очищає відповідний прапор завершення.

У наступній таблиці наведено коди помилок спеціальної сушарки Delta CANopen. (код помилки в SR659)

Код помилки	причина
0x0002	Підлеглий пристрій не відповідає на повідомлення SDO.
0x0003	Сталася помилка в повідомленні, отриманому підлеглим. Ця помилка часто виникає, коли параметри інструкції COPRW недійсні, через що підлеглий пристрій не отримує повне повідомлення.

0x0004	Підпорядковане повідомлення PDO не отримано.
0x0005	Під час використання операнда інструкції сталася помилка.
0x0006	Під час виконання інструкції INITC використовується одна зі станцій.
0x0007	Ідентифікатор підлеглого вузла перевищує прийнятний діапазон інструкції INITC
0x0008	Slave не може бути скинуто; перевірте, чи під'єднано кабель зв'язку та чи зв'язок підлеглого пристрою працює належним чином.

Якщо ви використовуєте Delta Special Driver & AS Remote mode / Delta Special Driver & CANopen DS301 і виконуєте INITC інструкції, час оновлення обміну даними PDO між станціями з 1 по 15 і з 21 по 28 показано нижче. Наприклад, коли швидкість передачі даних становить 1 МБ, час оновлення обміну даними PDO становить 10 мс X 5 разів = 50 мс.

Швидкість передачі (біт/с)	1М	500 тис	250 тис	125 тис	50 тис
Попередньо встановлений час для синхронізації (мс) #1	10	15	30	60	80
Попередньо встановлена кількість разів синхронізації #2	5	5	5	5	5

№1: Якщо ви використовуєте режим Delta Special Driver & CANopen DS301, ви можете використовувати програмне забезпечення CANopen Builder для редагування часу, щоб

синхронізувати для підлеглих. Але відредагований час може бути лише більшим за час, указаний у таблиці вище. Якщо ви встановите час, менший за вказаний вище, система ігноруватиме редагування та працюватиме відповідно до зазначеного вище часу.

№2: якщо ви використовуєте спеціальний драйвер Delta & режим CANopen DS301, ви можете використовувати програмне забезпечення CANopen Builder для редагування номера

часу синхронізації для заданої кількості часів синхронізації в таблиці DS301. Для попередньо встановленої кількості часу синхронізації його не можна редагувати.

Якщо ви використовуєте Delta Special Driver & AS Remote mode / Delta Special Driver & CANopen DS301 і виконуєте Інструкція COPPRW (зв'язок SDO), час очікування зв'язку показано нижче.

Швидкість передачі (біт/с)	1М	500 тис	250 тис	125 тис	50 тис
Час очікування зв'язку (мс) #1	100	100	200	400	500

Коли встановлено режим CANopen DS301, ви можете встановити час тайм-ауту COPRW у налаштуваннях зв'язку CANopen опція з HWCONFIG; тайм-аут за замовчуванням становить 100 мс.

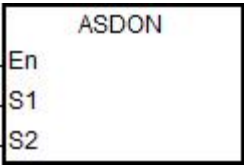
API		Інструкція				Операнд							опис							
2801			ASDON				S_1, S_2							Servo-ON і Servo-OFF						

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2													○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							

Інструкція пульсу		16-бітна інструкція		32-розрядна інструкція	
—		AS		—	

СИМВОЛ



S_1 : Ідентифікатор вузла сервоприводу

S_2 : Серво ON. і OFF

Пояснення

6

- Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а також Функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для зв'язку CAN інструкція INITC має бути виконана перед виконанням цієї інструкції. Для ECAT комунікації, інструкція INITEC повинна бути завершена до виконання цієї інструкції.
- Для зв'язку CAN діапазон S_1 становить 1–8 (для сервоприводу) та 21–28 (для інвертора). Виконання не відбудеться, якщо введене значення виходить за межі діапазону. Для зв'язку ECAT діапазон S_1 становить 1–16. Виконання не відбудеться, якщо введене значення виходить за межі діапазону. Використовуйте інструкцію INITEC для встановлення ідентифікаційного номера та типу пристрою.
- Діапазон S_1 (для сервоприводу) становить 1–8 (для зв'язку CAN) і 1-16 (для зв'язку ECAT), а S_2 є ненульовим значенням, сервопривід увімкнено (Servo-ON). Якщо S_2 дорівнює 0, сервопривід вимкнено (Servo-OFF).
- Діапазон S_1 (для інвертора) становить 21–28 (для зв'язку CAN) і 1–16 (для зв'язку ECAT). Коли S_2 дорівнює 1, це означає, що він працює в режимі швидкості інвертора. Коли S_2 дорівнює 2, це вказує на те, що він працює в режимі інверторного моменту. Коли S_2 дорівнює 0, це вказує на припинення роботи. Оскільки режими керування (режим швидкості та режим крутного моменту) для інверторів можна перемикаати безпосередньо, робота може продовжуватися. Перед перемикаанням режимів керування переконайтеся, що всі параметри інвертора встановлені.

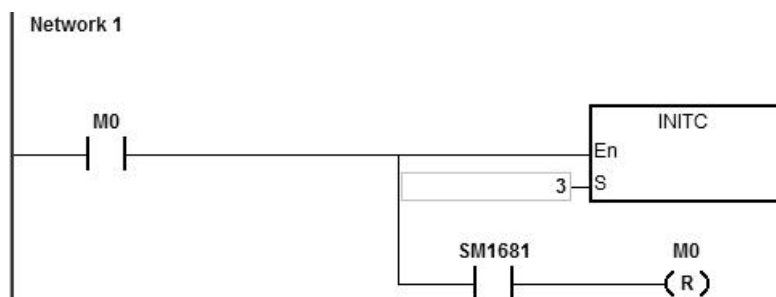
6. Таблиця останніх 2 пунктів:

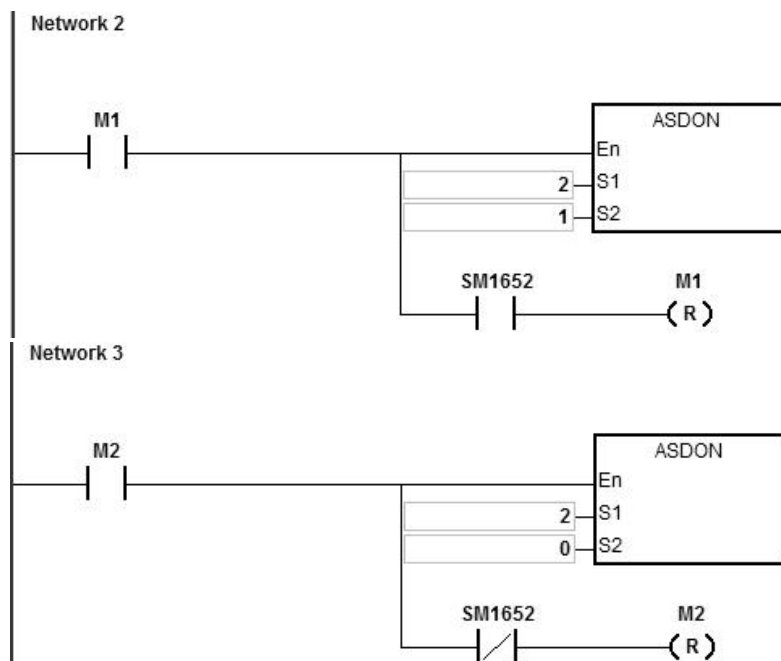
Тип приводу	Діапазон S ₁	Описи S ₂
Сервопривод	1-8 (CAN) 1-16 (ECAT)	1 = СЕРВОУВІМК 0 = СЕРВО OFF
Інвертор	21-28 (CAN) 1-16 (ECAT)	0 = припинити роботу 1 = Швидкісний режим 2 = Крутний момент

- Для зв'язку CAN кожен сервопривод (1-8) має прапорець (SM1651–M1658) для відображення свого стану. Для ECAT комунікації, кожен ідентифікатор вузла сервоприводу (1-16) має прапор (SM1651–M1658, SM1621-SM1628) для відображення свого стану. Ви можете прочитати фактичний стан сервоприводу з прапора. Коли прапорець увімкнено, сервопривід увімкнено. Коли прапорець OFF, сервопривід вимкнено.
- Для зв'язку CAN кожен інвертор (21-28) має прапорець (SM1621–M1628) для відображення свого стану. Для зв'язку ECAT кожен ідентифікатор вузла інвертора (1-16) має прапор (SM1651–M1658, SM1621-SM1628) для відображення його стану. Ви можете прочитати фактичний стан інвертора з прапора. Коли прапорець увімкнено, інвертор увімкнено. Коли прапор вимкнено, інвертор вимкнено.
- SM1682 увімкнено, якщо під час зв'язку виникає помилка. Крім того, SR658 зберігає ідентифікаційний номер, у якому сталася помилка, а SR659 зберігає код помилки.

Приклад 1 (Сходова діаграма):

- Коли M0 змінюється з OFF на ON, інструкція INITC починає ініціалізувати сервоприводи на вузлі ID 1–3 до SM1681 увімкнено.
- Коли M1 змінюється з OFF на ON, інструкція ASDON починає вмикати серво на вузлі ID 2. Коли SM1652 увімкнено, це вказує на Servo-ON.
- Коли M2 змінюється з OFF на ON, інструкція ASDON починає вимикати серво на вузлі ID 2. Коли SM1652 вимкнено, це вказує на Servo-OFF.





Приклад 2 (мова програмування структурованих текстів):

```

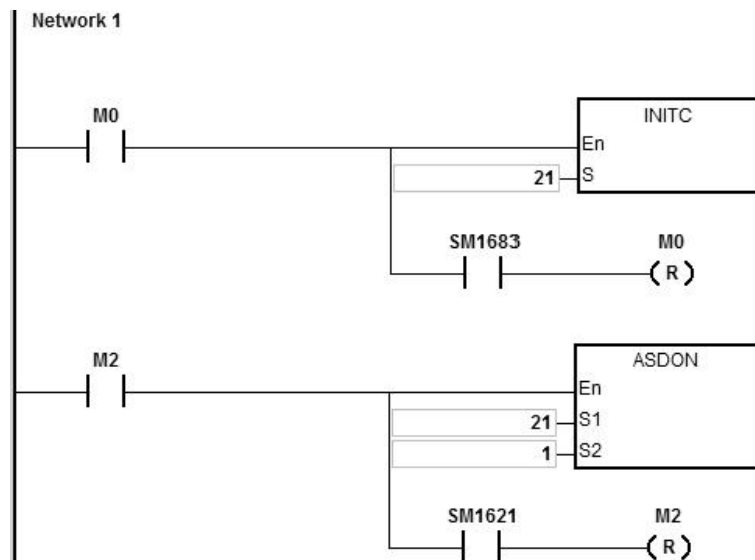
0001 IF M1 THEN
0002     ASDON(2,1);
0003 END_IF;
0004 IF M1 AND SM1652 THEN
0005     SM1582 := TRUE;
0006     M1 := FALSE;
0007 END_IF;
0008
0009 IF M2 THEN
0010     ASDON(2,0);
0011 END_IF;
0012 IF M2 AND (not SM1652) THEN
0013     SM1582 := TRUE;
0014     M2 := FALSE;
0015 END_IF;

```

1. Коли M1 змінюється з OFF на ON, ця інструкція починає вмикати сервопривід на вузлі ID 2. Коли SM1652 увімкнено, він вказує на Servo-ON.
2. Переконайтеся, що M1 і SM1652 увімкнено, а потім установіть прапор права керування зв'язком автоматичного повернення SM1582 у значення ON для ідентифікатора вузла 2. Це звільнить право керування зв'язком, і вузол з ідентифікатором 2 зможе отримувати та виконувати інші інструкції зв'язку.
3. Коли M2 змінюється з OFF на ON, ця інструкція починає вимикати серво на вузлі ID 2. Коли SM1652 вимкнено, це вказує на Servo-OFF.
4. Переконайтеся, що M2 увімкнено, а SM1652 вимкнено, а потім встановіть правий прапор керування зв'язком автоматичного повернення SM1582 у значення ON для ідентифікатора вузла 2. Це звільнить право керування зв'язком, і вузол з ідентифікатором 2 зможе отримувати та виконувати інші інструкції зв'язку.

Приклад 3 (зв'язок з інверторами Delta):

1. Коли M0 змінюється з OFF на ON, ця інструкція починає ініціалізацію вузла ID 21. Коли SM1683 є ON, це вказує на завершення ініціалізації.
2. Коли M2 змінюється з OFF на ON, ця інструкція починає встановлювати режим керування на вузлі ID 21. Коли SM1621 увімкнено, це означає, що інвертор працює.



API	Інструкція				Операнд								опис			
2802		CASD			S_1, S_2, S_3								Встановлення часу розгону та часу уповільнення для сервоприводу			

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1	●	●			●	●		●	●		○	○	○	○		
S_2	●	●			●	●		●	●		○	○	○	○		
S_3	●	●			●	●		●	●		○	○	○	○		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	INT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ

	CASD
En	
S1	
S2	
S3	

- S_1 : Ідентифікатор вузла сервоприводу
- S_2 : Час прискорення (мс)
- S_3 : Час уповільнення (мс)

Пояснення

- Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOFT V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для зв'язку CAN інструкція INITC має бути виконана перед виконанням цієї інструкції. Для зв'язку ECAT інструкція INITEC має бути завершена до виконання цієї інструкції.
- Для зв'язку CAN діапазон S_1 становить 1–8 (для сервоприводу) та 21–28 (для інвертора). Виконання не відбудеться, якщо введене значення виходить за межі діапазону. Для зв'язку ECAT діапазон S_1 становить 1–16. Виконання не відбудеться, якщо введене значення виходить за межі діапазону. Використовуйте інструкцію INITEC для встановлення ідентифікаційного номера та типу пристрою.
- Таблиця для опису S_2 і S_3

Тип приводу	Діапазон S_1	Діапазон S_2 і S_3	Описи S_2 і S_3
Сервопривод	1-8 (CAN) 1-16 (ECAT)	1-32767 (мс)	S_2 : Час прискорення – це період часу під час якого сервопривід обертається від 0 до 3000,0 об/хв.

			S₃ : Час уповільнення – це період часу під час якого сервопривід обертається з 3000,0 об/хв до 0.
Інвертор	21-28 (CAN зв'язок) 1-16 (комунікація ECAT)	100-32700 (мс) Введене значення потрібно помножити на 100.	S₂ : Час прискорення – це період часу під час якого інвертор обертається від 0,0 Гц до найвищої робочої частоти. S₃ : Час уповільнення – це період часу , протягом якого сервопривід обертається вниз від найвищої робочої частоти до 0.

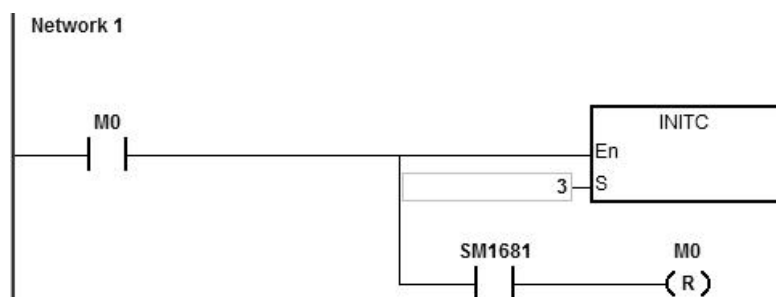
5. Для зв'язку CAN, коли ви виконуєте цю інструкцію та діапазон **S₁** становить 1–8 (для сервоприводу), SM1681 встановлюється на OFF (параметри редагуються). Коли сервопривід відповідає на отриману команду, SM1681 увімкнено (параметри налаштовані успішно). Коли ви виконуєте цю інструкцію, а діапазон **S₁** становить 21–28 (для інвертора), SM1683 встановлюється на OFF. (параметри редагуються). Коли інвертор відповідає на отриману команду, SM1683 увімкнено (параметри налаштовані успішно). Для зв'язку ECAT, коли ви виконуєте цю інструкцію та діапазон **S₁** становить 1–16, SM1681 встановлюється на OFF (параметри редагуються). Коли серво/інвертор відповідає на отриману команду, SM1681 увімкнено (параметри налаштовані успішно).
6. SM1682 встановлюється на ON, коли під час зв'язку виникає помилка. Крім того, SR658 зберігає ідентифікаційний номер, у якому сталася помилка, а SR659 зберігає код помилки.

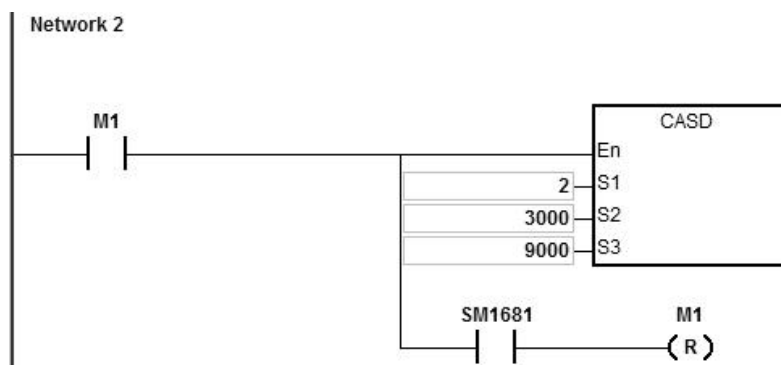
Приклад 1 (зв'язок із Delta ASD-A2M)

- Коли M0 змінюється з OFF на ON, інструкція INITC ініціалізує сервоприводи на вузлах ID 1–3, поки SM1681 не стане ON.
- Коли M1 змінюється з OFF на ON і цільова швидкість сервоприводу на вузлі ID 2 становить 3000 об/хв, інструкція CASD встановлює час прискорення сервоприводу 2 на 3000 мс і час уповільнення на 9000 мс.
- Якщо цільова швидкість сервоприводу 2 становить 1000 об/хв, час прискорення та час уповільнення показано нижче.

Час розгону: $[3000 \text{ мс} / 3000 \text{ об/хв}] \times 1000 \text{ об/хв} = 1000 \text{ мс}$

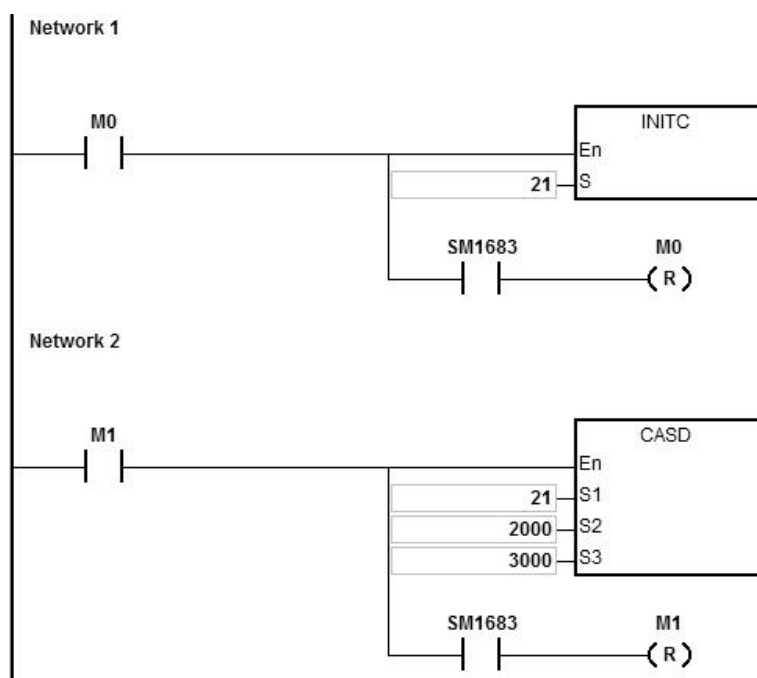
Час уповільнення: $[9000 \text{ мс} / 3000 \text{ об/хв}] \times 1000 \text{ об/хв} = 3000 \text{ мс}$





Приклад 2 (зв'язок з інверторами Delta)

1. Коли M0 змінюється з OFF на ON, ця інструкція починає ініціалізацію вузла ID 21. Коли SM1683 є ON, це вказує на завершення ініціалізації.
2. Коли M1 змінюється з OFF на ON, ця інструкція починає встановлювати час прискорення на 2000 мс і час уповільнення на 3000 мс у вузлі ID 21.



API	Інструкція				Операнд								опис			
2803	D	DRVIC			S_1, S_2, S_3								Контроль відносного положення сервоприводу			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●					O	O		
S_2								●	●				O	O		
S_3								●	●				O	O		
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING			
S_1			●				●									
S_2			●				●									
S_3			●				●									

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DDR VIC
En
S1
S2
S3

S_1 : Ідентифікатор вузла сервоприводу

S_2 : Відносна цільова позиція

S_3 : Цільова швидкість

6

Пояснення

- Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для зв'язку CAN інструкції INITC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції. Для зв'язку ECAT інструкції INITEC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції.
- Для зв'язку CAN діапазон S_1 становить 1–8. Виконання не відбудеться, якщо введене значення виходить за межі діапазону. Для зв'язку ECAT діапазон S_1 становить 1–16. Виконання не відбуватиметься, якщо вхідне значення виходить за межі діапазону або якщо пристрій із призначеним ідентифікатором не є сервоприводом.
- Діапазон S_2 становить від -2147483648 до +2147483647. Знак +/- вказує на напрямок вперед/назад. Цільова позиція є відносною.
- Одиницею величини S_3 є 0,1 об/хв. Діапазон 1–60000, що означає 0,1–6000,0 об/хв.

6. Коли цільове положення досягнуто, відповідні прапори завершення осей SM1631–SM1638 для вузла CAN ID 1-8 і відповідні прапори завершення осей SM1631–SM1638, SM1921-SM1928 для вузла ECAT ID 1-16 вмикаються. Після початку виведення осі ПЛК знімає відповідний прапор завершення. Якщо потрібне позиціонування кілька разів на одній вихідній осі, вам потрібно самостійно очистити відповідні позначки завершення, щоб заощадити час і уникнути плутанини.
7. Зверніться до наступної таблиці для відповідних SM і SR осей.
8. SM1682 увімкнено, коли під час зв'язку виникає помилка. Крім того, SR658 зберігає ідентифікаційний номер, у якому сталася помилка, а SR659 зберігає код помилки.
9. Ця інструкція використовує SDO (Service Data Object) протоколу CANopen для зв'язку. Оскільки протягом певного часу може бути надіслано лише одне повідомлення SDO, НЕ використовуйте цю інструкцію для виведення позиціонування під час синхронізованого руху по різних осях.
10. Для ЦП ПЛК із мікропрограмою версії 1.12.10 або новішої підтримується зміна цільової швидкості та цільової позиції під час виконання інструкцій.

Модифікація цільової швидкості: сервопривід діятиме відповідно до часу прискорення та уповільнення.

Якщо нова цільова швидкість дорівнює 0, сервопривід уповільниться, а потім зупиниться.

Якщо нову цільову швидкість змінено з 0 на будь-яке число, відмінне від 0, сервопривід відновить будь-який рух.

Модифікація цільової позиції:

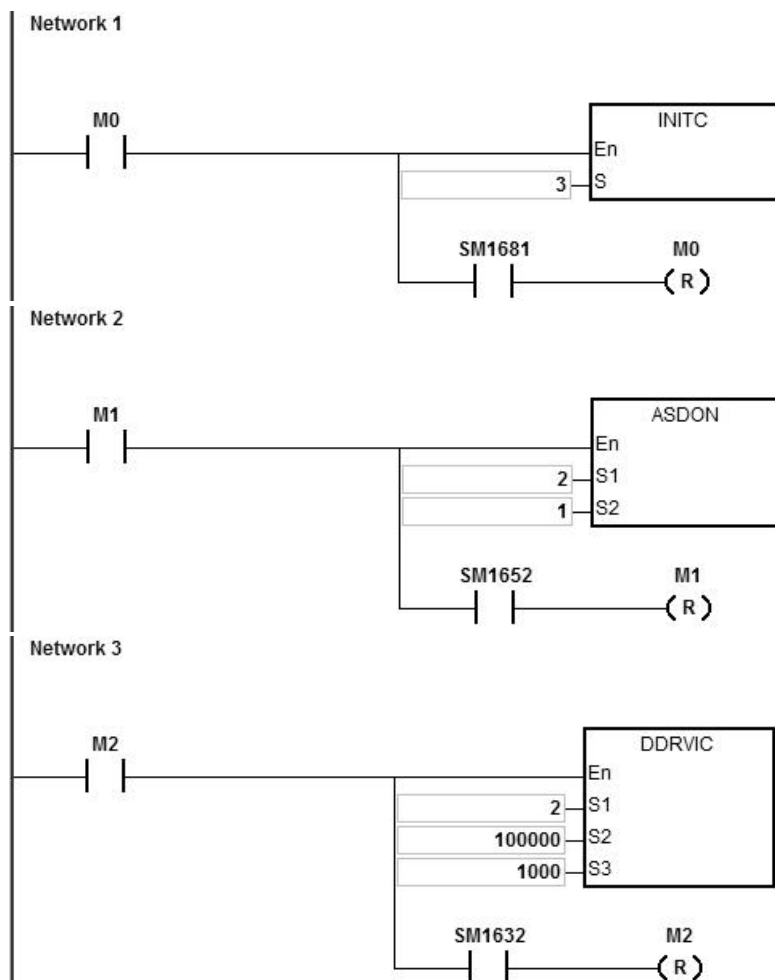
Для руху в негативному напрямку не дозволяється змінювати цільове положення.

Для відносного позиціонування, коли нове цільове положення дорівнює 0, сервопривід сповільнюється, а потім негайно зупиняється, і позиціонування вважається досягнутим.

Приклад 1 (Сходова діаграма):

1. Коли M0 змінюється з OFF на ON, інструкція INITC починає ініціалізувати сервоприводи на вузлі ID1–3, доки SM1681 не стане ON.
2. Коли M1 змінюється з OFF на ON, інструкція ASDON починає вмикати серво на вузлі ID2. Коли SM1652 увімкнено, це означає, що сервопривід увімкнено.
3. Коли M2 змінюється з OFF. на ON., сервопривід 2 переміщується у відносне положення 100000 PUU при 100,0 об/хв.

Прапор фінішу SM1632 увімкнено, коли досягнуто цільової позиції.



Приклад 2 (мова програмування структурованих текстів):

```

0001 IF M2 THEN
0002     DDRVIC (2, 100000, 1000);
0003 END_IF;
0004 IF M2 AND SM1632 THEN
0005     SM1582 := TRUE;
0006     M2 := FALSE;
0007 END_IF;

```

1. Коли M2 змінюється з OFF. на ON., сервопривід 2 переміщується у відносне положення 100000 PUU при 100,0 об/хв.

Прапор фінішу SM1632 увімкнено, коли досягнуто цільової позиції.

2. Переконайтеся, що M2 і SM1632 увімкнено, а потім установіть правий прапорець автоматичного повернення зв'язку SM1582 у положення ON для вузла ID2. Це звільнить право керування зв'язком, і вузол ID2 зможе отримувати та виконувати інші команди зв'язку.

Пояснення спеціальних прапорів (SM) і регістрів (SR) для спеціальних інструкцій зв'язку Delta CANopen

У наведеній нижче таблиці показано спеціальні позначки (SM) і регістри (SR), пов'язані зі спеціальним зв'язком Delta CANopen.

Прапор	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
Увімкнути певну функцію	R/W	Прапорець у HWCONFIG встановлюється на ON, коли ПЛК змінює режим зупинки на бігти.							
Ініціалізація сервоприводу та зв'язок завершені (INITC і CASD)	R	SM1681							
Помилка зв'язку CANopen	R	SM1682							
Ініціалізація одного веденого пристрою (для прошивки V1.06.40 або пізнішої версії)	R/W	SM1686 = OFF. (за замовчуванням; ініціалізація кількох підлеглих пристроїв) SM1686 = ON. (ініціалізація одного підлеглого пристрою)							
Позиціонування завершено	R/W №1	SM1631	SM1632	SM1633	SM1634	SM1635	SM1636	SM1637	SM1638
СТІЙ	R/W	SM1641	SM1642	SM1643	SM1644	SM1645	SM1646	SM1647	SM1648
Серво-ON	R	SM1651	SM1652	SM1653	SM1654	SM1655	SM1656	SM1657	SM1658
Увімкнено хід назад/уперед Підтримується лише DDRVAC.	R/W	SM1661	SM1662	SM1663	SM1664	SM1665	SM1666	SM1667	SM1668
Показчик напрямку руху вперед/назад Підтримується лише DDRVAC.	R	SM1671	SM1672	SM1673	SM1674	SM1675	SM1676	SM1677	SM1678
Автоматичне повернення права керування зв'язком	R/W	SM1581	SM1582	SM1583	SM1584	SM1585	SM1586	SM1587	SM1588
Код помилки Heartbeat	R	SM1691	SM1692	SM1693	SM1694	SM1695	SM1696	SM1697	SM1698
Обробка помилок Heartbeat	R/W	SM1684 = OFF. (за замовчуванням; коли один виходить з ладу, усі сервоприводи вимикаються.) SM1684 = ON (коли один опускається, лише несправний сервопривід вимикається.)							
ID номер з помилкою зв'язку	R	SR658							
Код помилки зв'язку	R	SR659							

Примітка 1: після того, як вісь починає виводити, ПЛК очищає відповідний прапор завершення.

У наведеній нижче таблиці показано, як параметри сервоприводу Delta осей відповідають спеціальним прапорцям і регістрам у зв'язку CANopen.

Назва параметра сервоприводу (номер)	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
PR команда (P5-07)	R	SR661	SR662	SR663	SR664	SR665	SR666	SR667	SR668
Код тривоги (P0-01) (шістнадцятковий)	R	SR671	SR672	SR673	SR674	SR675	SR676	SR677	SR678
Стан DO (P0-46)	R	SR681	SR682	SR683	SR684	SR685	SR686	SR687	SR688
Сервокомандне положення CMD_O (P0-09)	R	SR691	SR693	SR695	SR697	SR699	SR701	SR703	SR705
		SR692	SR694	SR696	SR698	SR700	SR702	SR704	SR706
Цільове положення сервоприводу CMD_E (P0-10)	R	SR711	SR713	SR715	SR717	SR719	SR721	SR723	SR725

		SR712	SR714	SR716	SR718	SR720	SR722	SR724	SR726
Стан DI (P4-07)	R	SR731	SR732	SR733	SR734	SR735	SR736	SR737	SR738
Поточний крутний момент (одиниця: 0,1%)	R	SR741	SR742	SR743	SR744	SR745	SR746	SR747	SR748
Самовизначений (P0-12) * Робота з SM1685 Enable/Disable	R	SR791 SR792	SR793 SR794	SR795 SR796	SR797 SR798	SR799 SR800	SR801 SR802	SR803 SR804	SR805 SR806
Налаштування діапазону завершення позиціонування	R/W	SR811	SR812	SR813	SR814	SR815	SR816	SR817	SR818

*Наприклад, якщо запитується реальне положення сервоприводу, встановіть значення серо в P0-20, увімкніть SM1685, а потім реальне положення сервоприводу.

положення можна прочитати в SR791-SR806. Додаткову інформацію див. у Посібнику користувача серії ASDA-A2 для сервоприводу змінного струму високої роздільної здатності Delta для програм мережевого зв'язку.

У наведеній нижче таблиці показано, як параметри дельта-інвертора осей відповідають спеціальним прапорцям і регістрам у зв'язку CANopen.

опис		ID. 21	ID. 22	ID. 23	ID. 24	ID. 25	ID. 26	ID. 27	ID. 28
Прапор завершення зв'язку INITC і CASD інвертора	R	SM1683							
Прапор запуску інвертора	R	SM1621	SM1622	SM1623	SM1624	SM1625	SM1626	SM1627	SM1628
Прапор помилки Heartbeat інвертора	R	SM1611	SM1612	SM1613	SM1614	SM1615	SM1616	SM1617	SM1618
Автоматичне скидання контролю зв'язку	R/W	SM1601	SM1602	SM1603	SM1604	SM1605	SM1606	SM1607	SM1608
Статус інвертора (індекс 6041H-00H)	R	SR751	SR752	SR753	SR754	SR755	SR756	SR757	SR758
Швидкість струму інвертора (індекс 6043H-00H)	R	SR761	SR762	SR763	SR764	SR765	SR766	SR767	SR768
Крутний момент струму інвертора (одиниця: 0,1%) (Індекс 2021H-0CH)	R	SR771	SR772	SR773	SR774	SR775	SR776	SR777	SR778
Стан цифрового входу інвертора (індекс 2022H-11H)	R	SR781	SR782	SR783	SR784	SR785	SR786	SR787	SR788

У наведеній нижче таблиці наведено коди помилок зв'язку CANopen для сервоприводів і інверторів Delta. (код помилки в SR659)

Код помилки	причина
0x0002	Підлеглий пристрій не відповідає на повідомлення SDO.
0x0003	Сталася помилка в повідомленні, отриманому підлеглим. Ця помилка часто виникає, коли параметри інструкції COPRW недійсні, через що підлеглий пристрій не отримує повне повідомлення.
0x0004	Підпорядковане повідомлення PDO не отримано.
0x0005	Під час використання операнда інструкції сталася помилка.
0x0006	Під час виконання інструкції INITC використовується одна зі станцій.

0x0007	ID підлеглого вузла перевищує прийнятний діапазон інструкції INITC
0x0008	Slave не може бути скинуто; перевірте, чи під'єднано кабель зв'язку та чи зв'язок підлеглого пристрою працює належним чином.

- У наведеній нижче таблиці показано спеціальні прапори (SM) і регістри (SR), пов'язані зі зв'язком ECAT.

Прапор	R/W	ID. 1 до ID. 16
Ініціалізація та зв'язок завершені (INITC і CASD)	R	SM1681
Помилка зв'язку	R	SM1682
Обробка втрати з'єднання	R/W	SM1684 = OFF. (за замовчуванням; коли один вимикається, усі диски вимикаються.) SM1684 = ON (коли один вимикається, лише несправний диск OFF.)
Код скасування ECAT SDO (32-розрядний)	R/W	SR656 SR657
ID номер з помилкою зв'язку	R/W	SR658
Код помилки зв'язку	R/W	SR659

функція	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
Позиціонування сервоприводу завершено №1	R/W	SM1631	SM1632	SM1633	SM1634	SM1635	SM1636	SM1637	SM1638
Сервопривід	R/W	SM1641	SM1642	SM1643	SM1644	SM1645	SM1646	SM1647	SM1648
Серво-ON, інвертор-ON	R	SM1651	SM1652	SM1653	SM1654	SM1655	SM1656	SM1657	SM1658
Увімкнуто хід назад/уперед Підтримується лише DDRVAC.	R/W	SM1661	SM1662	SM1663	SM1664	SM1665	SM1666	SM1667	SM1668
Показчик напрямку руху вперед/назад Підтримується лише DDRVAC.	R	SM1671	SM1672	SM1673	SM1674	SM1675	SM1676	SM1677	SM1678
Автоматичне повернення права керування зв'язком. Підтримується лише DDRVAC.	R/W	SM1581	SM1582	SM1583	SM1584	SM1585	SM1586	SM1587	SM1588
Код помилки втрати з'єднання №2	R/W	SM1691	SM1692	SM1693	SM1694	SM1695	SM1696	SM1697	SM1698

функція	R/W	ID. 9	ID. 10	ID. 11	ID. 12	ID. 13	ID. 14	ID. 15	ID. 16
Позиціонування сервоприводу завершено №1	R/W	SM1921	SM1922	SM1923	SM1924	SM1925	SM1926	SM1927	SM1928
Сервопривід	R/W	SM1931	SM1932	SM1933	SM1934	SM1935	SM1936	SM1937	SM1938
Серво-ON, інвертор-ON	R	SM1621	SM1622	SM1623	SM1624	SM1625	SM1626	SM1627	SM1628
Увімкнуто хід назад/уперед Підтримується лише DDRVAC.	R/W	SM1941	SM1942	SM1943	SM1944	SM1945	SM1946	SM1947	SM1948
Показчик напрямку руху вперед/назад Підтримується лише DDRVAC.	R	SM1951	SM1952	SM1953	SM1954	SM1955	SM1956	SM1957	SM1958
Автоматичне повернення права керування зв'язком. Підтримується лише DDRVAC.	R/W	SM1601	SM1602	SM1603	SM1604	SM1605	SM1606	SM1607	SM1608
Код помилки втрати з'єднання №2	R/W	SM1611	SM1612	SM1613	SM1614	SM1615	SM1616	SM1617	SM1618

#1: Час автоматичного вимкнення прапора завершення позиціонування сервоприводу відбувається, коли ввімкнено виведення осі. Якщо вам потрібно використовувати інструкції позиціонування на певній осі кілька разів поспіль, вам потрібно самостійно очистити прапорець завершення позиціонування сервоприводу. Якщо ви не очистите прапор завершення позиціонування сервоприводу, а потім знову і знову використовуєте інструкцію позиціонування, можливо, що перед виконанням наступної інструкції позиціонування буде виявлено прапор завершення позиціонування сервоприводу, і тоді виконання інструкції позиціонування буде зупинено.

2: Оскільки серцебиття недоступне для зв'язку ECAT, робочий лічильник (WKC) використовується як спосіб визначити, чи втрачено з'єднання підпорядкованого пристрою.

● У наступній таблиці показано спеціальні прапори (SM) і регістри (SR), пов'язані з комунікаціями Servo ECAT.

Назва параметра (номер)	R/W	ID. 1	ID. 2	ID. 3	ID. 4	ID. 5	ID. 6	ID. 7	ID. 8
Команда Servo PR (P5-07) / Стан інвертора (індекс 6041H-00H)	R	SR661	SR662	SR663	SR664	SR665	SR666	SR667	SR668
Код сигналу сервоприводу (P0-01)	R	SR671	SR672	SR673	SR674	SR675	SR676	SR677	SR678
Стан сервоприводу DO (P0-46)	R	SR681	SR682	SR683	SR684	SR685	SR686	SR687	SR688
Сервокомандне положення CMD_O (P0-09)	R	SR691	SR693	SR695	SR697	SR699	SR701	SR703	SR705
		SR692	SR694	SR696	SR698	SR700	SR702	SR704	SR706
Цільове положення сервоприводу CMD_E (P0-10)	R	SR711	SR713	SR715	SR717	SR719	SR721	SR723	SR725
		SR712	SR714	SR716	SR718	SR720	SR722	SR724	SR726
Стан сервоприводу DI (P4-07)	R	SR731	SR732	SR733	SR734	SR735	SR736	SR737	SR738
Поточний крутний момент (P0-11)	R	SR741	SR742	SR743	SR744	SR745	SR746	SR747	SR748
Сервопривод Self-defined (P0-12)	R	SR791	SR793	SR795	SR797	SR799	SR801	SR803	SR805
		SR792	SR794	SR796	SR798	SR800	SR802	SR804	SR806
Налаштування діапазону завершення позиціонування	R/W	SR811	SR812	SR813	SR814	SR815	SR816	SR817	SR818

Назва параметра (номер)	R/W	ID. 9	ID. 10	ID. 11	ID. 12	ID. 13	ID. 14	ID. 15	ID. 16
Команда Servo PR (P5-07) / Стан інвертора (індекс 6041H-00H)	R	SR751	SR752	SR753	SR754	SR755	SR756	SR757	SR758
Код сигналу сервоприводу (P0-01)	R	SR761	SR762	SR763	SR764	SR765	SR766	SR767	SR768
Стан сервоприводу DO (P0-46)	R	SR1191	SR1192	SR1193	SR1194	SR1195	SR1196	SR1197	SR1198
Сервокомандне положення CMD_O (P0-09)	R	SR1201	SR1203	SR1205	SR1207	SR1209	SR1211	SR1213	SR1215
		SR1202	SR1204	SR1206	SR1208	SR1210	SR1212	SR1214	SR1216
Цільове положення сервоприводу CMD_E (P0-10)	R	SR1221	SR1223	SR1225	SR1227	SR1229	SR1231	SR1233	SR1235
		SR1222	SR1224	SR1226	SR1228	SR1230	SR1232	SR1234	SR1236
Стан сервоприводу DI (P4-07)	R	SR781	SR782	SR783	SR784	SR785	SR786	SR787	SR788
Поточний крутний момент (P0-11)	R	SR771	SR772	SR773	SR774	SR775	SR776	SR777	SR778
Сервопривод Self-defined (P0-12)	R	SR1241	SR1243	SR1245	SR1247	SR1249	SR1251	SR1253	SR1255
		SR1242	SR1244	SR1246	SR1248	SR1250	SR1252	SR1254	SR1256
Налаштування діапазону завершення позиціонування	R/W	SR1261	SR1262	SR1263	SR1264	SR1265	SR1266	SR1267	SR1268

У наступній таблиці показано коди помилок зв'язку ECAT для сервоприводів і інверторів Delta. (код помилки в SR659)

Код помилки	причина
0x0002	Підлеглий пристрій не відповідає на повідомлення SDO.
0x0003	Сталася помилка в повідомленні, отриманому підлеглим. Ця помилка часто виникає, коли параметри інструкції COPRW недійсні, через що підлеглий пристрій не отримує повне повідомлення.
0x0004	Підпорядковане повідомлення PDO не отримано.
0x0005	Під час використання операнда інструкції сталася помилка.
0x0006	Під час виконання інструкції INITC використовується одна зі станцій.
0x0007	ID підлеглого вузла перевищує прийнятний діапазон інструкції INITC
0x0008	Slave не може бути скинуто; перевірте, чи під'єднано кабель зв'язку та чи зв'язок підлеглого пристрою працює належним чином.
0x0009	AS-FECAT HE відповідає.
0x000A	Команда DCSOFC одночасно керує більш ніж чотирма підлеглими.

API	Інструкція				Операнд								опис			
2804	D	DRVAC			S_1, S_2, S_3								Абсолютний контроль положення сервоприводу			

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●					O	O		
S_2								●	●				O	O		
S_3								●	●				O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1			●				●						
S_2			●				●						
S_3			●				●						

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DDRAC
En
S1
S2
S3

- S_1 : Ідентифікатор вузла сервоприводу
- S_2 : Абсолютна цільова позиція
- S_3 : Цільова швидкість

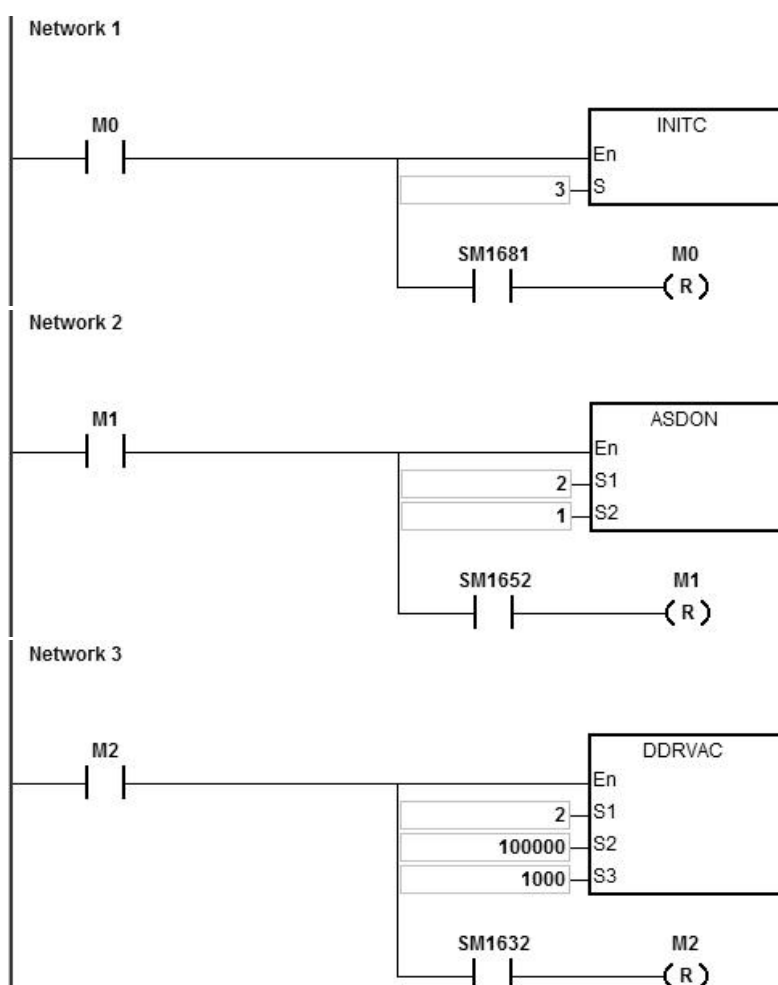
Пояснення

- Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для зв'язку CAN інструкції INITC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції. Для зв'язку ECAT інструкції INITEC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції.
- Для зв'язку CAN діапазон S_1 становить 1–8. Виконання не відбудеться, якщо введене значення виходить за межі діапазону. Для зв'язку ECAT діапазон S_1 становить 1–16. Виконання не відбуватиметься, якщо вхідне значення виходить за межі діапазону або якщо пристрій із призначеним ідентифікатором не є сервоприводом.
- Діапазон S_2 становить від -2147483648 до +2147483647. Знак +/- вказує на напрямок вперед/назад. Цільова позиція є абсолютною.
- Зверніться до інструкції DRVIC (API 2803) для отримання додаткової інформації та прикладів.

Приклад 1

1. Коли M0 змінюється з OFF на ON, інструкція INITC починає ініціалізувати сервоприводи на вузлі ID1–3, доки SM1681 не стане ON.
2. Коли M1 змінюється з OFF на ON, інструкція ASDON починає вмикати серво на вузлі ID2. Коли SM1652 увімкнено, це означає, що сервопривід увімкнено.
3. Коли M2 змінюється з OFF на ON., сервопривід 2 переміщується з поточного положення в абсолютне положення 100 000 PUU при 100,0 об/хв.

Прапор фінішу SM1632 увімкнено, коли досягнуто цільової позиції.



Приклад 2

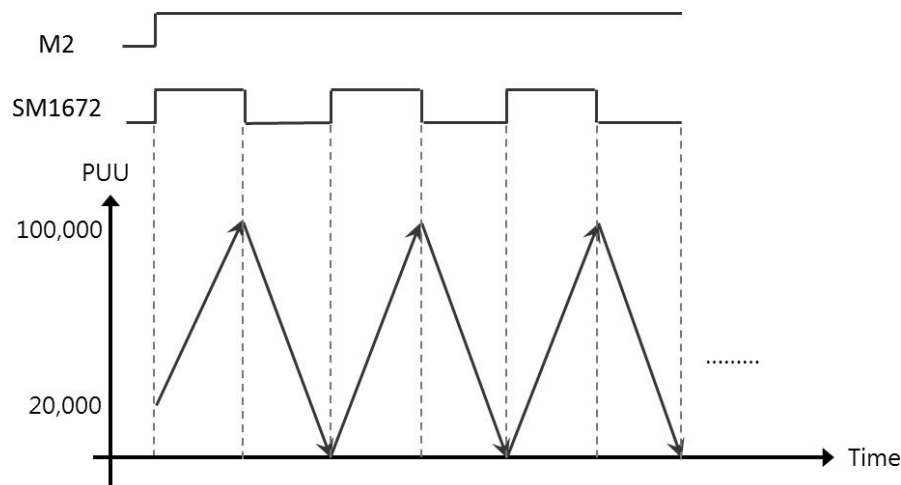
1. Додайте один рядок до програми в прикладі 1. Коли ПЛК працює, а SM1662 увімкнено, увімкнено функцію для сервоприводу 2, щоб повертатися вперед і назад.



2. Як показано на малюнку нижче, сервопривід переміщується зі свого поточного положення (20 000) до абсолютного цільового положення (100 000) після того, як M2 увімкнено. Після цього він повертається вперед і назад між абсолютною позицією 100 000 і 0.

Прапор індикації напрямку SM1672 увімкнено, коли сервопривід рухається до цільової позиції вперше після увімкнення сервоприводу. Після цього прапорець повторює стан, змінюючи з ON на OFF.

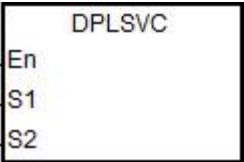
3. Ви можете змінити цільове положення в будь-який момент руху, але нове цільове положення дійсне лише для наступного циклу вперед і назад.



API	Інструкція				Операнд								опис				
2805	D	PLSVC			S_1, S_2								Серворегулювання швидкості				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1								●					O	O			
S_2								●	●				O	O			
Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING				
S_1			●				●										
S_2			●				●										

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



S_1 : Ідентифікатор вузла сервоприводу

S_2 Цільова швидкість

Пояснення

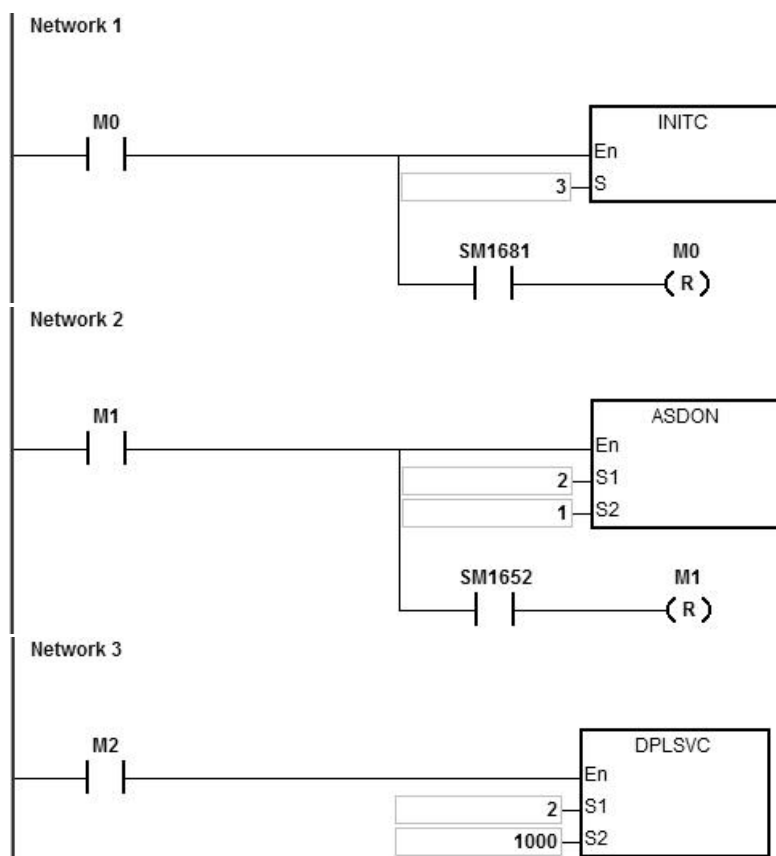
- Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а також Функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для зв'язку CAN інструкції INITC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції. Для зв'язку ECAT інструкції INITEC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції.
- Для зв'язку CAN діапазон S_1 становить 1–8. Виконання не відбудеться, якщо введене значення виходить за межі діапазону. Для зв'язку ECAT діапазон S_1 становить 1–16. Виконання не відбудеться, якщо введене значення виходить за межі діапазону.
- Таблиця для опису S_2

Тип сервоприводу	Діапазон S_1	Діапазон S_2	Описи S_2
Сервопривод	1-8 (CAN) 1-16 (ECAT)	-60000 - 60000 (0,1 об/хв)	Знак +/- вказує на напрямок вперед/назад. для Наприклад, при введенні 6005 у S_2 значення буде 600,5 об/хв.
Інвертор	21-28 (CAN) 1-16 (ECAT)	-32768 - 32767 (1 об/хв)	Знак +/- вказує на напрямок вперед/назад. Наприклад , при введенні 605 у S_2 значення буде 605 об/хв.

5. Для відповідних SM і SR осей зверніться до інструкції DRVIC (API 2803).
6. SM1682 встановлюється на ON, коли під час зв'язку виникає помилка. Крім того, SR658 зберігає номер осі, на якій сталася помилка, а SR659 зберігає код помилки.

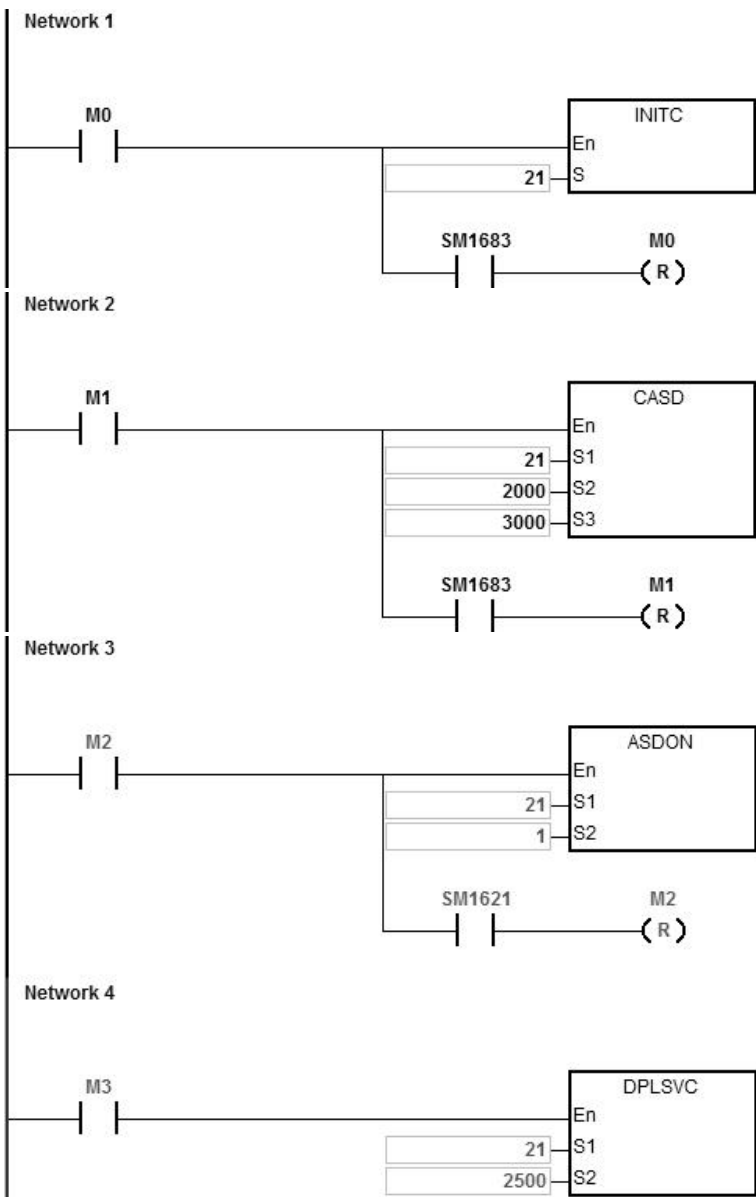
Приклад 1 (зв'язок із сервосистемами Delta)

5. Коли M0 змінюється з OFF на ON, інструкція INITC починає ініціалізувати сервоприводи на вузлі ID 1–3, доки SM1681 не стане ON.
6. Коли M1 змінюється з OFF на ON, інструкція ASDON починає вмикати серво на вузлі ID 2. Коли SM1652 увімкнено, це вказує на Servo-ON.
7. Коли M2 змінюється з OFF на ON, сервопривід 2 рухається зі швидкістю 100,0 об/хв, доки M2 не вимкнеться.



Приклад 2 (зв'язок з інверторами Delta)

- 1. Коли M0 змінюється з OFF на ON, ця інструкція починає ініціалізацію вузла ID 21. Коли SM1683 є ON, це вказує на завершення ініціалізації.
- 2. Коли M1 змінюється з OFF на ON, ця інструкція починає встановлювати час прискорення та час уповільнення на ідентифікаторі вузла 21.
- 3. Коли M2 змінюється з OFF на ON, ця інструкція починає встановлювати режим керування на вузлі ID 21.
- 4. Коли M3 змінюється з OFF на ON, інвертор рухається зі швидкістю 2500 об/хв, поки M3 не вимкнеться.



API	Інструкція			Операнд								опис					
2806	D	ZRNC		S_1, S_2, S_3								Сервонаведення					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●					O	O		
S_2								●	●				O	O		
S_3								●	●				O	O		

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●	●			●							
S_3		●	●			●							

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	AS

СИМВОЛ

ZRNC
En
S1
S2
S3

S_1 : Ідентифікатор вузла сервоприводу

S_2 : 1 -й – сегмент швидкості

S_3 : 2 -й – відріzkова швидкість

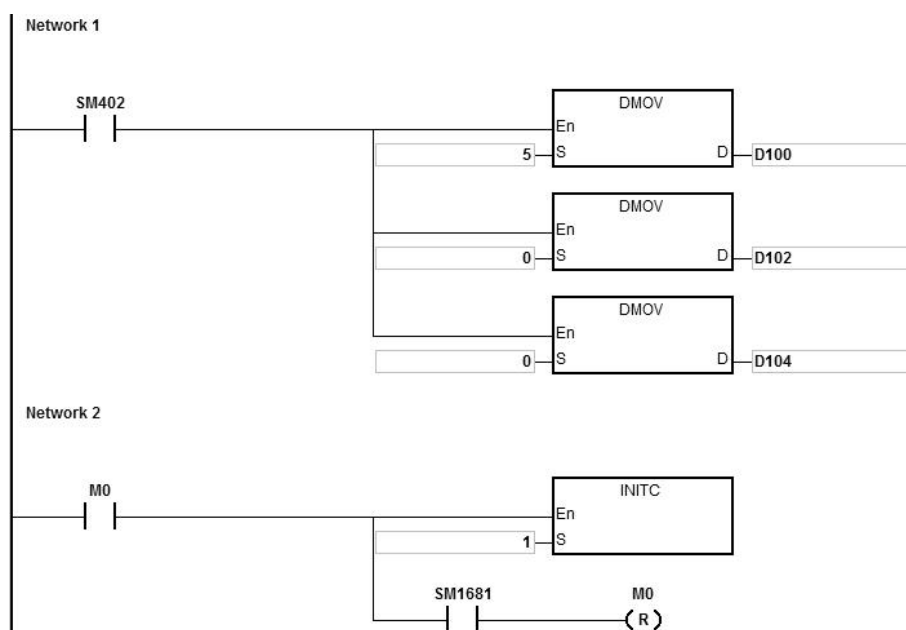
Пояснення

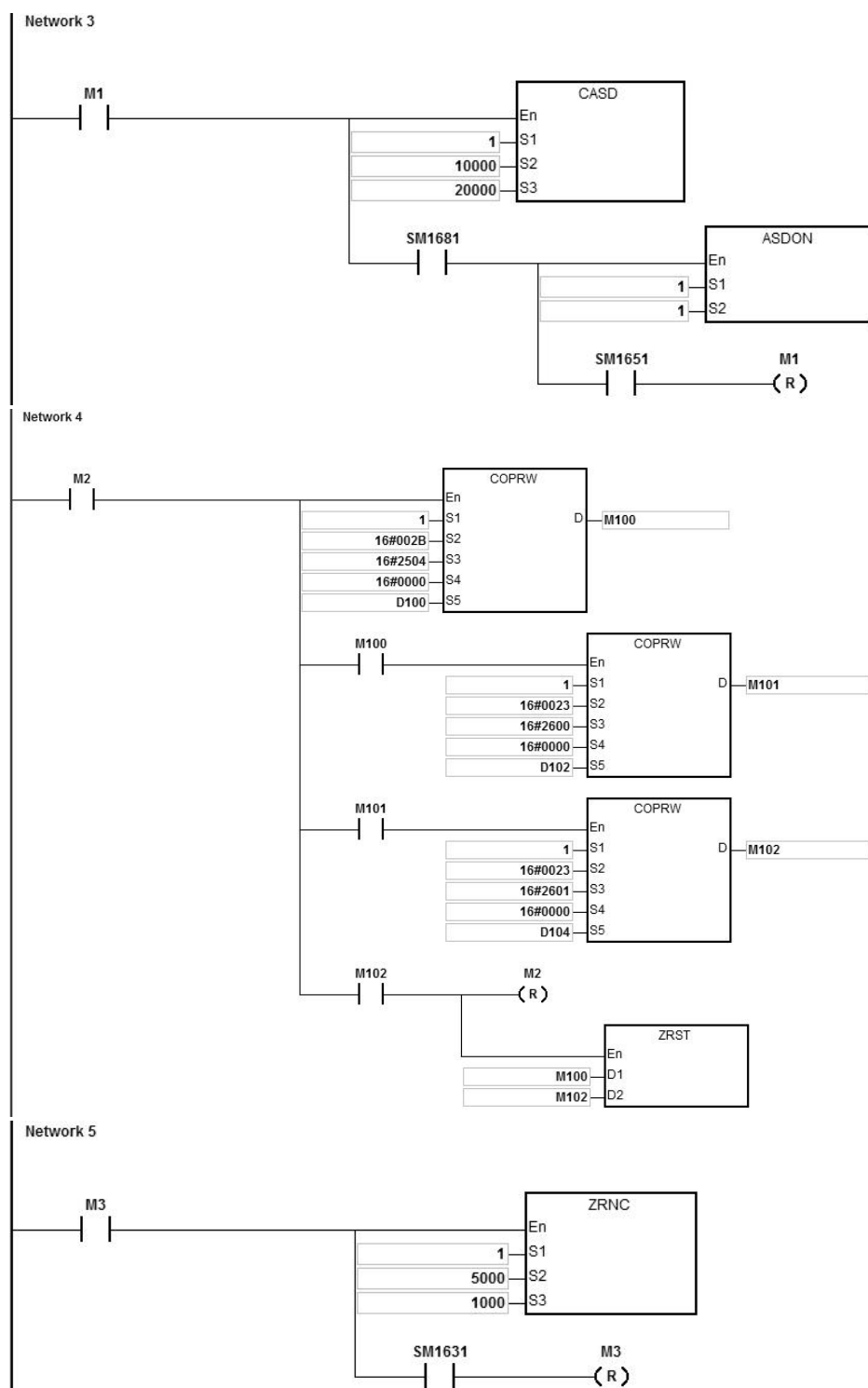
- Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а також
Функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для зв'язку CAN інструкції INITC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції. Для зв'язку ECAT інструкції INITEC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції. Пропонується використовувати ZRNM (API 2810) для встановлення всіх осей сервоприводу в режим наведення.
- Для зв'язку CAN діапазон S_1 становить 1–8. Виконання не відбудеться, якщо введене значення виходить за межі діапазону. Для зв'язку ECAT діапазон S_1 становить 1–16. Виконання не відбуватиметься, якщо вхідне значення виходить за межі діапазону або якщо пристрій із призначеним ідентифікатором не є сервоприводом.
- Діапазон S_2 становить 1–20000. Одиниця: 0,1 об/хв (параметр сервоприводу P5-05); якщо вам потрібно використовувати привід лінійного двигуна, а перша секція перевищує 32767, замість цього потрібно використовувати інструкцію DZRNC (доступно лише для PLC FW V1.04 або новішої версії).
- Діапазон S_3 становить 1–5000. Одиниця: 0,1 об/хв (параметр сервоприводу P5-06)
- Коли сервопривод повертається у вихідне положення, відповідні прапорці фінішу осей увімкнені. Зверніться до інструкції DRVIC (API 2803) для пояснення спеціальних прапорців (SM) і регістрів (SR).

7. SM1682 встановлюється на ON, коли під час зв'язку виникає помилка. Крім того, SR658 зберігає номер осі, на якій сталася помилка, а SR659 зберігає код помилки.

приклад

1. Коли M0 змінюється з OFF на ON, інструкція INITC починає ініціалізацію сервоприводу 1, поки SM1681 не стане ON.
2. Коли M1 змінюється з OFF на ON, час прискорення (10000) і час уповільнення (20000) сервоприводу 1 встановлюється за допомогою інструкції CASD (API 2802). Коли SM1651 увімкнено, це означає, що сервопривід увімкнено.
3. Коли M2 змінюється з OFF. на ON.,
 2-байтові дані записуються на P5-04 у сервоприводі 1. Коли D100 = 5 і дані записуються, M100 увімкнено. 4-байтові дані записуються в P6-00 у сервоприводі 1. Коли D102 = 0 і дані записуються, M101 увімкнено. 4-байтові дані записуються на P6-01 у сервоприводі 1. Коли D104 = 0 і дані записуються, M102 увімкнено. P5-04 встановлює режим наведення.
 P6-00 і P6-01 є визначеннями наведення.
4. Коли M3 змінюється з OFF. на ON., функція наведення вмикається для сервоприводу 1.





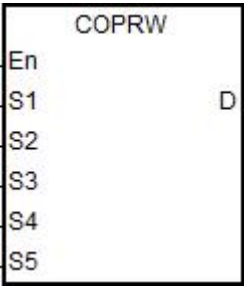
API	Інструкція				Операнд								опис				
2807		COPRW			$S_1, S_2, S_3, S_4, S_5, D$								Читання та запис даних зв'язку CANopen				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
S_1								●					○	○			
S_2								●	●				○	○			
S_3								●	●				○	○			
S_4								●	●				○	○			
S_5								●									
D		●	●	●				●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●				●							
S_3		●				●							
S_4		●				●							
S_5		●				●							
D	●												

Інструкція пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

6

СИМВОЛ



- S_1 : Ідентифікатор вузла серво або підлеглого
- S_2 : Код запиту
- S_3 : Індекс
- S_4 : Підіндекс
- S_5 : Пристрій читання/запису
- D : Прапор завершення зв'язку

Пояснення

1. Ця інструкція читає та записує дані зв'язку CANopen до сервоприводу за адресою, зазначеною в S_1 . Діапазон S_1 становить 1–127. Якщо значення виходить за межі діапазону (<1 або >127), мінімальне або максимальне значення автоматично обробляється інструкцією як значення S_1 .
1. S_2 може вказати лише чотири типи кодів запиту, як показано в наступній таблиці. Якщо ви встановили неправильний код запиту, він НЕ виконується; SM1682 увімкнено, а SR659 — 16#0005.

S₂ (доступно лише для значень молодших 8 біт)	опис
16#0023	Запис 4-байтових даних; Прискорений SDO
16#002B	Запис 2-байтових даних; Прискорений SDO
16#002F	Запис 1-байтових даних; Прискорений SDO
16#0040	Читання даних. Довжина даних міститься у повідомленні відповіді SDO; Прискорений SDO
16#0021	Дані та їх довжина визначені S ₅ ; Звичайний SDO
16#0041	Зчитування даних і збереження зчитаних даних і їх довжини в S ₅ ; Звичайний SDO

- Для **S₃** і **S₄** зверніться до словника об'єктів у посібнику з експлуатації серво/інвертора Delta або до словника об'єктів із пристрою стороннього виробника, який сумісний із CANopen DS301.
- Інструкцію COPRW слід виконувати лише після завершення інструкції INITC, якщо параметри перезаписані інструкцією INITC. Використовуйте інструкцію COPRW для зв'язку CANopen SDO. Якщо ідентифікатор підлеглого вузла перевищує діапазон 1–8 або 21–28, ви можете використовувати інструкцію COPRW для встановлення, читання та запису параметрів, пов'язаних із підлеглим.

Якщо ви використовуєте Delta Special Driver & AS Remote mode / Delta Special Driver & CANopen DS301 і виконуєте інструкцію COPRW (зв'язок SDO), час очікування зв'язку показано нижче.

Швидкість передачі (біт/с)	1М	500 тис	250 тис	125 тис	50 тис
Час очікування зв'язку (мс) #1	100	100	200	400	500

Коли встановлено режим CANopen DS301, ви можете встановити час очікування COPRW у зв'язку CANopen параметр налаштування з HWCONFIG; тайм-аут за замовчуванням становить 100 мс.

- SM1682 встановлюється на ON, коли під час зв'язку виникає помилка. Крім того, SR658 зберігає ідентифікаційний номер, у якому сталася помилка, а SR659 зберігає код помилки. Додаткову інформацію дивіться в інструкції DRVIC.

Примітка. Коли ви використовуєте інструкцію COPRW, ви повинні відредагувати процес обробки помилок зв'язку, щоб уникнути виникнення недійсного зв'язку внаслідок неочікуваних помилок зв'язку.

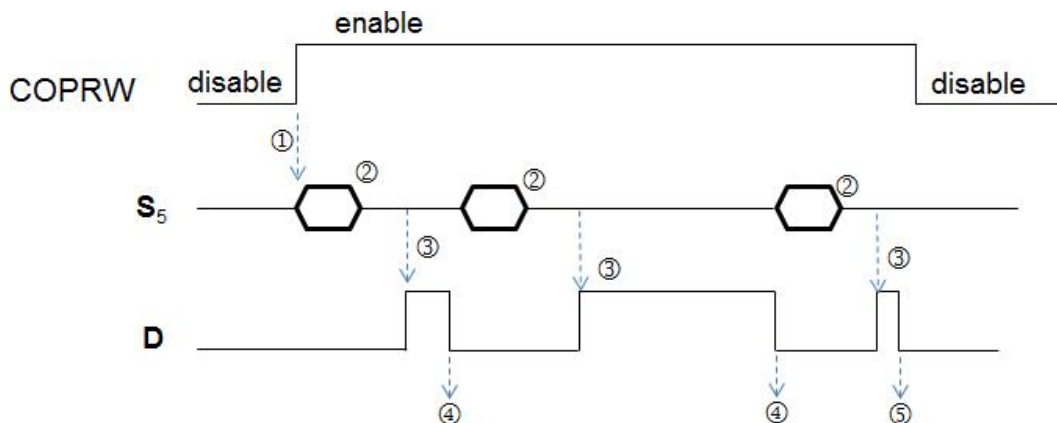
- На діаграмі нижче показано час виконання інструкції COPRW.

(1) Коли ви вперше вмикаєте інструкцію COPRW, інструкція негайно надсилає код команди, якщо інший зв'язок CANopen не використовує її.

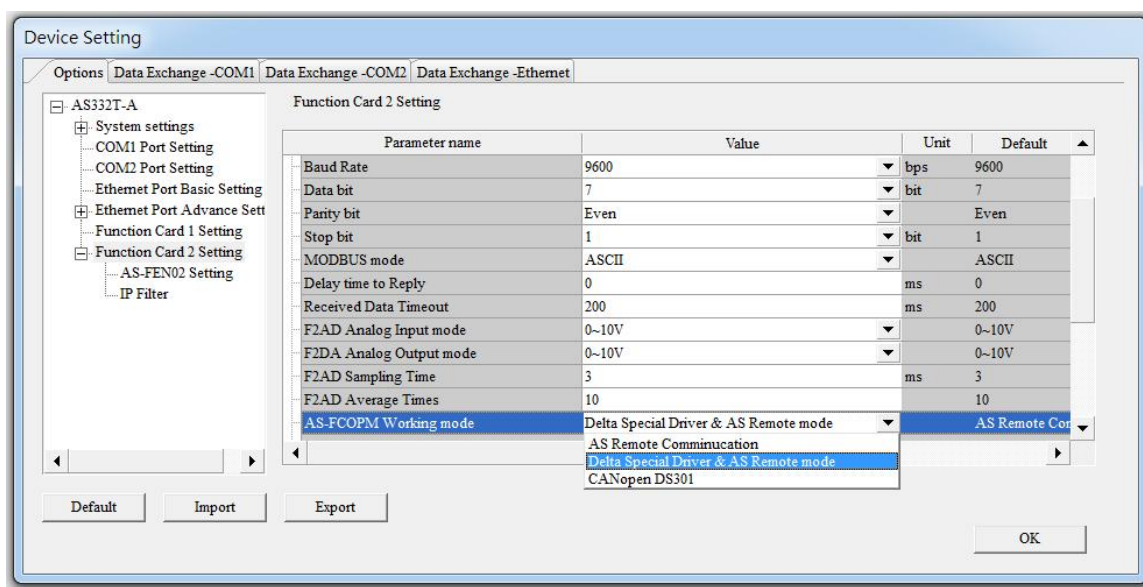
(2) Інструкція надсилає код команди.

(3) Код надіслано, і прапор завершення встановлено на ON.

- (4) Ви змінюєте наступні дані для надсилання. Код наступної команди надсилається відразу після того, як прапор завершення встановлено на OFF.
- (5) Код надіслано, а інструкцію COPRW вимкнено.



7. Інструкція COPRW підтримує лише **Delta Special Driver & AS Remote Mode** та **CANopen DS301** у налаштування апаратної конфігурації (HWCONFIG в ISPSOft).



8. Більшість параметрів у Delta ASDA-A2 відображаються в десятковому форматі. Ви можете конвертувати параметри в

індексні адреси, див. приклад нижче. 0 є фіксованим значенням для адреси підіндексу.

Приклад: адреса індексу $PX-YY=0x2000 + (X \ll 8) + YY$ P2-10 =

$$0x2000 + (0x0002 \ll 8) + 0x000A = 0x220A \text{ P5-04} =$$

$$0x2000 + (0x0005 \ll 8) + 0x0004 = 0x2504 \text{ P1-44} =$$

$$0x2000 + (0x0001 \ll 8) + 0x002C = 0x212C$$

9. Більшість параметрів інвертора Delta також відображаються в десятковому форматі. Використовуйте наступну формулу для перетворення параметрів.

приклад: Адреса індексу $PXX-YY=0x2000 + XX$ (шістнадцяткове);

Адреса підіндексу – $YY+1$ (шістнадцяткове)

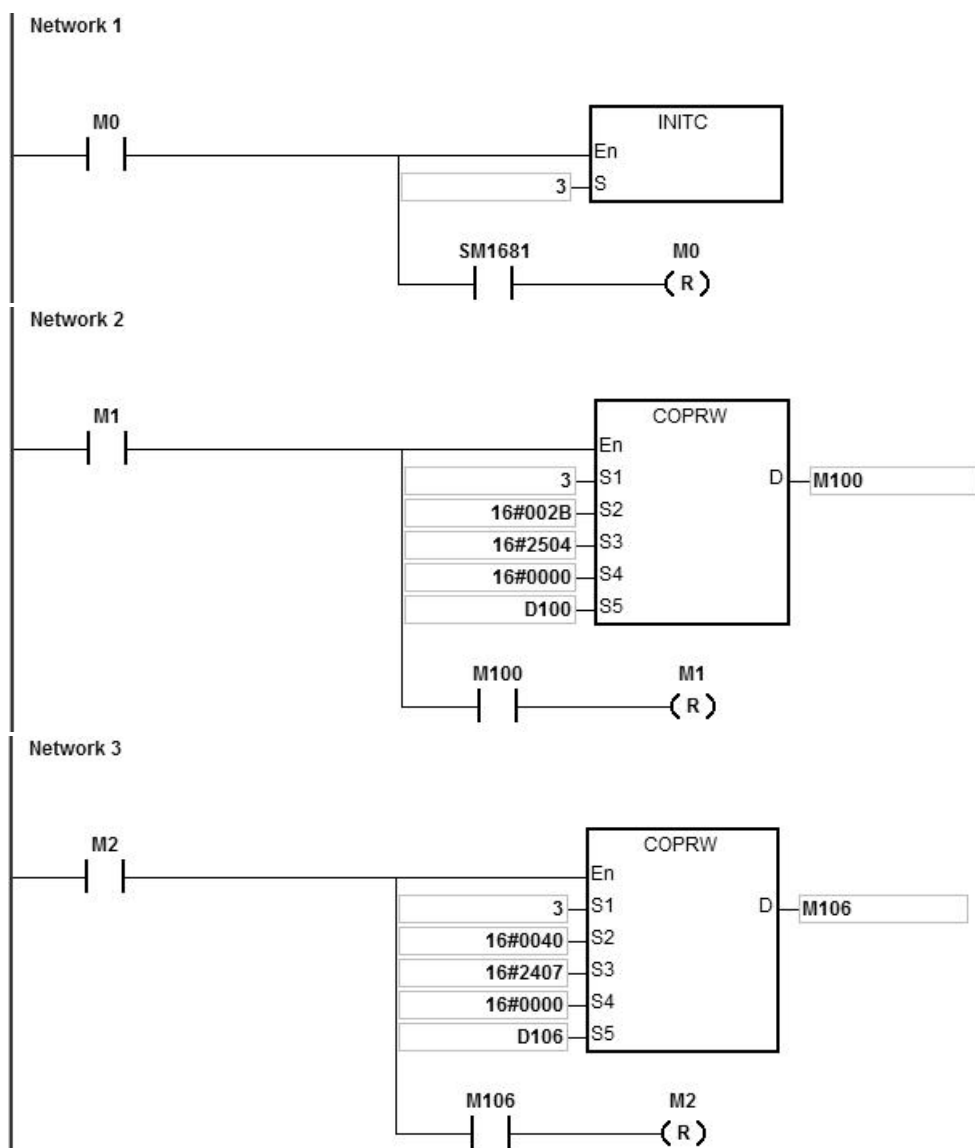
Індексна адреса $P10-15 = 0x2000 + 0x000A = 0x200A$

Адреса підіндексу $0x0F+1= 0x10$

10. Оскільки в цій інструкції використовується SDO (об'єкти службових даних), режим зв'язку — це одна передача (одне надсилання/одне отримання) за раз, а право зв'язку спільно з інструкціями високошвидкісного виведення. Під час написання програм PLC майте на увазі наступне.
- * Рекомендується НЕ використовувати 2 інструкції COPRW або зв'язок одночасно в одному циклі сканування.
 - * Рекомендується НЕ використовувати цю інструкцію у функціональному блоці. Тому що, коли ця інструкція записана у функціональному блоці, у програмі ПЛК буде 2 POU, і коли функціональний блок буде викликано, 2 інструкції будуть увімкнені одночасно.
11. Коди запиту для звичайного SDO: 16#0021 і 16#0041, доступні для ЦП ПЛК із мікропрограмою версії 1.04.80 або новішої.
12. Коли код запиту 16#0021, S_5 є довжиною даних («n» байт) для запису в підлеглий пристрій. $S_5+1 \sim S_5+(n+1)/2$ є дані для запису в slave. Наприклад, якщо S_5 вказано як D100, а значення в S_5 дорівнює 16. Ця інструкція надсилає дані в D101 ~ D108 на підлеглий пристрій у такому порядку: молодші 8 біт у D101, старші 8 біт у D101, молодші 8-біт у D102, старші 8-біт у D102 і так далі.
13. Коли код запиту 16#0041, коли отримані відповіді від підлеглого пристрою, ця інструкція зберігає довжину даних («n» байт) в S_5 і дані в $S_5+1 \sim S_5+(n+1)/2$. Наприклад, якщо отримано 15 байтів довжини даних, D100 дорівнює 15 і збереже дані Byte[0], Byte[1].. Byte[14] у такому порядку, нижні 8 біт у D101, старші 8 біт у D101, молодші 8-біт у D102, старші 8-біт у D102 і так далі.
14. Максимальна довжина даних становить 255 байт. Щоб запобігти занадто великій кількості нормальних передач SDO, якщо довжина даних перевищує 255 байт, ПЛК надсилає або отримує лише 255 байт.

Приклад (з використанням Expedited SDO)

1. Коли M0 змінюється з OFF на ON, інструкція INITC починає ініціалізувати сервоприводи на вузлі ID 1–3, доки SM1681 не стане ON.
2. Коли M1 змінюється з OFF на ON, ПЛК записує 2-байтові дані ($S_2=16\#002B$) у D100 до P5-04 ($S_3=16\#2504$) сервоприводу 3, використовуючи інструкцію COPRW. Після завершення запису M100 увімкнено.
3. Коли M2 змінюється з OFF на ON, ПЛК зчитує значення P4-07 ($S_3=16\#2407$) сервоприводу 3 і зберігає значення ($S_2=16\#0040$) у D106. Коли зчитування завершено, M106 увімкнено.



API	Інструкція			Операнд								опис					
2808	D	COPWL		S_1, S_2, S_3, D								Запис кількох значень параметрів CANopen					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
S_1								●					O	O		
S_2								●	●							
S_3								●					O	O		
D		●	●	●				●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
S_1		●				●							
S_2		●	●			●	●						
S_3		●				●							
D	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	AS

СИМВОЛ

COPWL		DCOPWL	
En		En	
S1	D	S1	D
S2		S2	
S3		S3	

S_1 : Ідентифікатор вузла (ідентифікатор Mac)

S_2 : Початковий вихідний пристрій, де зберігаються записані дані

S_3 : Кількість повідомлень для послідовного запису даних

D : Прапор завершення зв'язку

Пояснення

1. S_1 встановлює ідентифікатор вузла в діапазоні 1~127. Якщо значення параметра перевищує діапазон (< 1 або >127), інструкція автоматично надсилатиме дані з мінімальним або максимальним значенням відповідно.

2. S_2 — початковий вихідний пристрій, де зберігаються записані дані, а S_3 — кількість повідомлень для послідовного запису даних. Наприклад, S_2 визначає D10 як початковий пристрій, а кількість повідомлень для послідовного запису даних дорівнює 3. Ось докладне пояснення в наступній таблиці.

Назва інструкції	Повідомлення №	Адреса індексу	Підіндексна адреса	Письмові вихідні дані
COPWL (Записує 16-бітні значення)	1	D10	D11	D12
	2	D13	D14	D15
	3	D16	D17	D18
DCOPWL (Записує 32-розрядні значення)	1	D10	D11	D12, D13
	2	D14	D15	D16, D17
	3	D18	D19	D20, D21

Значення S_3 знаходиться в діапазоні 1~100.

3. Індексні адреси та підіндексні адреси сервоприводу Delta та приводу змінного струму див. у поясненні до інструкції COPRW. В принципі, значення параметрів сервоприводу Delta та приводу змінного струму є як 16-бітними, так і 32-бітними, включаючи числа з плаваючою комою. Якщо вам потрібно записати 8-бітне значення, використовуйте інструкцію COPRW.

4. **D** — позначка завершення зв'язку. **D** увімкнеться після завершення надсилання кількох повідомлень.

Перегляньте детальний процес надсилання та схему послідовності нижче.

(1) Інструкція COPWL увімкнена та починає надсилати дані.

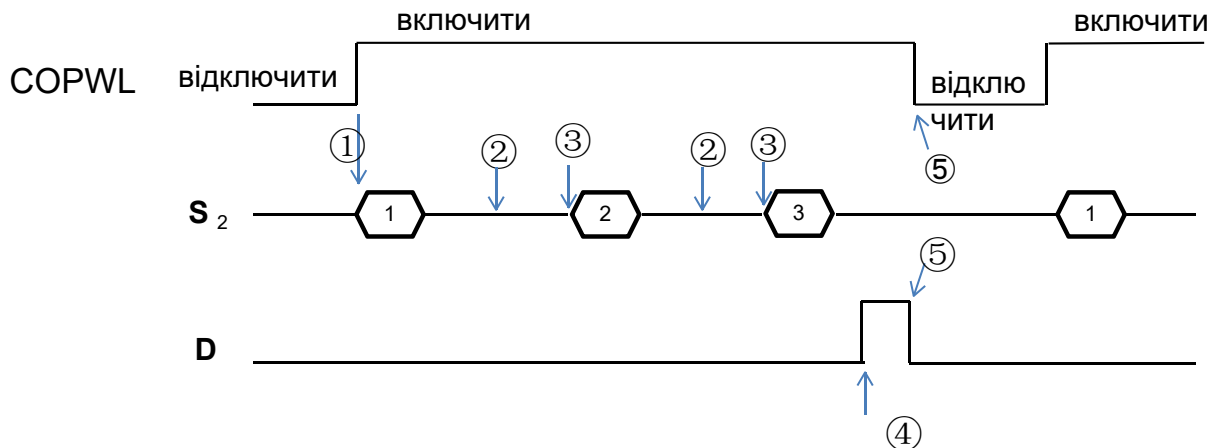
(2) Після того, як інструкція COPWL надішле одне повідомлення, наступна інструкція ПЛК продовжує виконуватися.

(3) Коли інструкція COPWL сканується ще раз і попереднє повідомлення було отримано підлеглим, інструкція COPWL надсилає наступне повідомлення.

(4) Після останнього надсилання записаних даних інструкція встановить прапор завершення на ON.

(5) Коли ввімкнеться позначка завершення, інструкцію COPWL потрібно вимкнути вручну, щоб наступна інструкція COPWL або COPRW могла продовжувати працювати.

Примітка: якщо вимкнути інструкцію, прапор завершення буде автоматично очищено відповідно.



Примітка. На діаграмі послідовності вище показано надсилання 3 частин письмових даних.

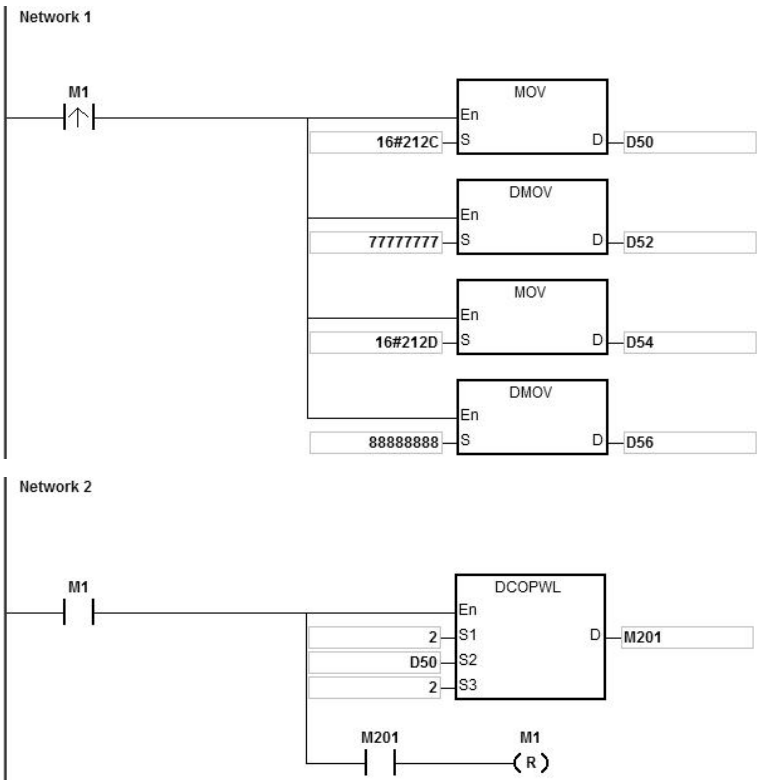
5. Після ввімкнення інструкції дочекайтеся завершення запису, а потім вимкніть інструкцію. Якщо під час виконання сталася помилка зв'язку, усуньте несправність, а потім повторно ввімкніть інструкцію для запису всіх даних.

приклад

1. Коли M1 = OFF → ON, дані записуються на пристрій D.

Назва інструкції	Дані №	Адреса індексу	Підіндексн а адреса	Source письмових даних
DCOPWL (Записує 32-розрядні значення)	1	D50 = 16#212C (Чисельник електронного передавального числа)	D51 = 0	D52, D53 = 77777777
	2	D54 = 16#212D (Знаменник електронного передавального числа)	D55 = 0	D56, D57 = 88888888

2. Коли M1 = OFF → ON, інструкція записує 32-бітне значення для P1-44 сервоприводу, ідентифікатор якого дорівнює 2, а
записане значення 77777777 зберігається в D52. Інструкція записує 32-розрядне значення для P1-45, а
записане значення 88888888 зберігається в D56. Після завершення запису вмикається M201.



API	Інструкція				Операнд				опис			
2809		RSTD			Node , Para , OK , Err				Надсилання команди Reset або NMT			

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Node								●					○	○		
Para								●					○	○		
OK		●	●	●												
Err		●	●	●												

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●											
Para		●											
OK	●												
Err	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ

RSTD			
En			
Node		Ok	
Para		Err	

- Node** : Ідентифікатор вузла, який скидається
- Para** : Установка значення параметра
- OK** : Прапор завершення скидання
- Err** : Прапор помилки скидання

Пояснення

- Лише мікропрограмне забезпечення ПЛК версії 1.04.10 і пізніших і програмне забезпечення ISPSOft версії 3.03.08 і пізніших підтримують цю інструкцію. Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для зв'язку CAN перед використанням інструкції RSTD у режимі спеціальних інструкцій Delta переконайтеся, що всі сервоприводи/інвертори Delta були ініціалізовані за допомогою інструкції INITC і колись вони працювали нормально.
- Для зв'язку ECAT перед використанням інструкції RSTD у режимі спеціальних інструкцій Delta переконайтеся, що всі сервоприводи/інвертори Delta були ініціалізовані за допомогою інструкції INITEC і колись вони працювали нормально.
- Для зв'язку CAN, коли використовується в режимі CANopen DS301, інструкція RSTD працює як функція зв'язку NMT і може перемикає стани мережі за допомогою параметра **Para**.
- Для зв'язку CAN, коли порт зв'язку CAN вказано для роботи в режимі спеціального драйвера Delta, значення **вузла** може бути 0 (для функції трансляції) та 1–8, які призначені лише для ідентифікатора сервовузла, і 21–28 для вузла

Ідентифікатор лише приводів змінного струму. Коли ідентифікатор вузла виходить за межі діапазону, ПЛК не виконуватиме дію скидання та

Вмикається прапор **Err** . (Коди помилок див. у поясненні до SR659)

6. Для зв'язку CAN, коли порт зв'язку CAN вказано для роботи в режимі CANopen DS301, значення **Node** знаходиться в діапазоні 1~64 і 0 (для функції трансляції). Коли значення виходить за межі діапазону, ПЛК не виконуватиме зв'язок NMT, і вмикається прапор **Err** . (Коди помилок див. у поясненні до SR659)

7. Для зв'язку ECAT значення **Node** може бути 0 (для функції трансляції) та 1~16, які призначені для ідентифікатора вузла серво/інвертора. Коли ідентифікатор вузла перевищує діапазон, ПЛК не виконуватиме дію скидання, і вмикається прапор **Err** . (Коди помилок див. у поясненні до SR659)

8. Значення параметра **Para** застосовується лише до режиму CANopen DS301. Параметр не має сенсу, якщо інструкція використовується в спеціальному режимі драйвера Delta та віддаленому режимі AS. Параметри для **Para** (сервісний код NMT) наведено в наступній таблиці. Якщо значення налаштування не є одним із значень у таблиці, увімкнеться позначка **Err** .

NMT код послуги	16#01	16#02	16#80	16#81	16#82
Опис функції	Запустіть раб	Зупинить раба	Перейти в передексплуатаційний стан	Скинути прикладний рівень	Скинути зв'язок

9. Інструкція RSTD може щоразу виконувати дію команди лише на одному диску або підлеглому пристрої. Якщо декілька RSTD

інструкції ввімкнено одночасно, ПЛК автоматично матиме пріоритет для виконання інструкції, яка ввімкнена раніше.

10. Інструкція RSTD виконується для надсилання команди, коли вона включена. Якщо інструкцію вимкнено перед **OK** увімкнено, ПЛК не встановлюватиме прапор **OK** на ON.

11. Окрім сповіщення вказаного диска про очищення стану помилки, інструкція також перевірятиме повторно, якщо це необхідно

значення параметрів зв'язку є правильними та повторно встановить правильні значення параметрів зв'язку.

Наприклад, через відключення підлеглого вузла ID 2 вся система припиняє працювати. Після усунення проблеми підлеглий вузол з ідентифікатором 2 може повернутися до стану керованості за допомогою інструкції RSTD для скидання лише підлеглого вузла з ідентифікатором 2. Таким чином зберігається час повторної ініціалізації всіх дисків.

12. Якщо підлеглий пристрій відповідає, надсилаючи назад будь-яку помилку команди зв'язку до ПЛК під час зв'язку, інструкція RSTD увімкне прапор **Err** і зупинить наступні дії. (Зверніться до пояснення SR659 для кодів помилок.)

API	Інструкція			Операнд								опис					
2810		ZRNM		Node , Mode , OK , Err								Налаштування режиму наведення для сервоприводу Delta					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Node								●					○	○		
Mode								●					○	○		
OK		●	●	●												
Err		●	●	●												

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●											
Mode		●											
OK	●												
Err	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ

ZRNM	
En	
Node	Ok
Mode	Err

- Node** : Вказаний ідентифікатор вузла
- Mode** : Код режиму самонаведення
- OK** : Прапор завершення
- Err** : Прапор помилки

Пояснення

- Лише мікропрограмне забезпечення ПЛК версії 1.04.10 і пізніших і програмне забезпечення ISPSOFT версії 3.03.08 і пізніших підтримують цю інструкцію. Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOFT V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Перед виконанням інструкції скидання переконайтеся, що всі диски Delta ініціалізовано.
- Для зв'язку CAN значення **Node** знаходиться в діапазоні від 1 до 8 (за винятком ідентифікаторів сервоузелів) або в діапазоні від 1 до 16 для зв'язку ECAT. Якщо значення налаштування виходить за межі діапазону або якщо пристрій із призначеним ідентифікатором не є сервоприводом, ПЛК не виконуватиме дію режиму наведення та встановлюватиме прапор **Err** на ON. (Зверніться до пояснення SR659 для кодів помилок.)
- Інструкція ZRNM може щоразу встановлювати режим наведення тільки для одного диска. Якщо кілька інструкцій увімкнено одночасно, ПЛК матиме пріоритет для виконання інструкції, яка була увімкнена раніше.
- Інструкція ZRNM виконується для надсилання команди, коли вона увімкнена. Якщо вказівку вимкнено або НЕ буде скановано знову до того, як буде увімкнено прапорець **Ok**, ПЛК НЕ зможе встановити прапорець **Ok** або прапорець **Err** на ON.

6. Ця інструкція буде активована лише один раз, коли контакт EN перемикається з OFF. на ON. Рекомендується використовувати цю інструкцію в програмуванні сходів. Якщо використовується ця інструкція в програмуванні SFC або ST, коли контакт EN вимкнено, його НЕ можна сканувати.

7. **Mode** встановлює режим наведення. Якщо значення налаштування виходить за межі діапазону, ПЛК все одно надсилатиме команду, а сервер сам вирішуватиме, отримувати команду чи ні. Режим налаштування — це режим наведення, якому відповідає параметр сервоприводу ASDA P5-04.

Значення параметра режиму наведення дельта-сервоприводу є шістнадцятковим. Значення визначається як формат 0xWZYX. Дивіться пояснення відповідних кодів, наведені нижче.

Наведення код режиму	Діапазон	Опис функції та коду	Зауважен ня
W	0 ~ 1	Виберіть кінцеве положення, де зупиниться сервопривід. 0 = сервопривід залишає вихідне положення, сповільнюється та зупиняється, а потім автоматично повертається у фактичне вихідне положення. 1 = Після виходу з вихідного положення, уповільнення та зупинки сервопривід більше не виконуватиме жодних дій.	
Z	0 ~ 1	Механізм обробки, коли зустрічається обмеження. 0 = вихід припиняється. 1 = Вивід здійснюється у зворотному напрямку.	
Y	0 ~ 2	Налаштування імпульсного сигналу Z (використовується для коду X 0~8) 0 = Шукайте імпульс Z під час повернення. Не шукайте фазу Z, коли рухається вперед. 1 = Перехід до імпульсу Z. Не шукайте Z pulse, коли повертається. 2 = Не шукайте імпульс Z. Спосіб обробки сигналу Z-імпульсу (застосовується до коду X: 9~A) 0 =Зверніть увагу на Z-імпульс під час повернення. 1 = Не шукайте імпульс Z як при поверненні, так і при русі вперед.	
X	0 ~ A	Метод наведення: 0~8 0 = Наведення в прямому напрямку; PL - вихідна позиція. 1 = Наведення в зворотному напрямку; NL - вихідна позиція. 2 = Наведення в прямому напрямку; ORGP : OFF > ON, як початкове положення. 3 = Наведення в зворотному напрямку; ORGP : OFF > ON	

		початкове положення. 4 = Наведення в прямому напрямку; шукайте імпульс Z і розглядайте його як початкове положення. 5 = Наведення в зворотному напрямку; шукайте імпульс Z і розглядайте його як початкове положення. 6 = Наведення в прямому напрямку; ORGP: ON > OFF, як вихідне положення. 7 = Наведення в зворотному напрямку; ORGP: ON > OFF, як вихідне положення. 8 = Поточна позиція є вихідною. Метод наведення: 9~A 9 = Наведення в прямому напрямку; точкою зіткнення є вихідна позиція. A = Наведення в зворотному напрямку; точкою зіткнення є вихідна позиція.	
--	--	--	--

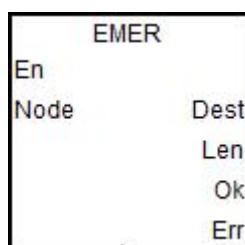
API	Інструкція				Операнд								опис			
2811		EMER			Node , Dest , Len , OK , Err								Читання екстреного повідомлення			

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Node								●					O	O		
Dest								●								
Len								●								
OK		●	●	●												
Err		●	●	●												

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●											
Dest		●											
Len		●											
OK	●												
Err	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ



- Node** : Вказаний ідентифікатор вузла
- Dest** : Цільовий пристрій для зберігання даних
- Len** : Загальна кількість прочитаних 4 слів даних
- OK** : Прапор завершення
- Err** : Прапор помилки

Пояснення

- Лише мікропрограмне забезпечення ПЛК версії 1.04.10 і пізніших і програмне забезпечення ISPSOft версії 3.03.08 і пізніших підтримують цю інструкцію. Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Інструкція EMER застосовна до режиму CANOpen DS301, спеціального режиму Delta та зв'язку ECAT.
- Після отримання екстреного повідомлення від підлеглого **вузла** ПЛК автоматично збереже дані на вказаному пристрої зберігання та встановить прапор **OK** у значення ON.
- Рекомендується, щоб значення **вузла** вказувалося з ідентифікаторів підлеглих вузлів, які вже існували. Якщо значення не є ідентифікатором одного існуючого вузла або підлеглий пристрій було від'єднано, ПЛК не зможе отримати жодне повідомлення, установить прапор **Err** на ON і відобразить код помилки часу очікування зв'язку. (Див. пояснення SR659)

для кодів помилок.)

5. Спосіб, у який інструкція EMER читає екстрені повідомлення, такий самий, як метод екстреного зв'язку в посібнику з експлуатації AS. Просто виберіть із них один спосіб зв'язку під час читання екстрених повідомлень.
6. Інструкція EMER може прочитати щонайбільше 5 екстрених повідомлень. Кожного разу, коли читання завершується успішно, вмикається прапорець **Ok i Len** відображає загальну кількість прочитаних повідомлень. Ви можете оцінити, скільки послідовних слів займає **Dest** на основі довжини. Кожне повідомлення містить 4 слова. Дані зберігаються в порядку від молодших 8 бітів до старших 8 бітів. Формат зберігання показаний нижче. (Наприклад, **Dest** — D10, **Len** — 2, тобто кількість повідомлень, що зберігаються в D5.)

D пристрій №	Значення
D5	2

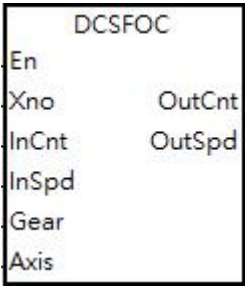
D пристрій №	Вищі 8 біт	Молодші 8 біт
D10	Другий байт у першому повідомленні	Перший байт у першому повідомленні
D11	Четвертий байт у першому повідомленні	Третій байт у першому повідомленні
D12	Шостий байт у першому повідомленні	П'ятий байт у першому повідомленні
D13	Восьмий байт у першому повідомленні	Сьомий байт у першому повідомленні
D14	Другий байт у другому повідомлення	Перший байт у другому повідомленні
D15	Четвертий байт у другому повідомленні	Третій байт у другому повідомленні
D16	Шостий байт у другому повідомленні	П'ятий байт у другому повідомленні
D17	Восьмий байт у другому повідомленні	Сьомий байт у другому повідомлення

API	Інструкція				Операнд								опис			
2812	D	CSFOC			Xno ~ OutSpd								Керування функцією відстеження сервоприводу через зв'язок			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Xno	O															
InCnt								●								
InSpd								●								
Gear								●								●
Axis								●					●	●		
OutCnt								●								
OutSpd								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Xno	●												
InCnt			●				●						
InSpd			●				●						
Gear									●				
Axis		●											
OutCnt			●				●						
OutSpd			●				●						

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ



- Xno** : Пристрій введення сигналу
- InCnt** : Кількість вхідних імпульсів
- InSpd** : Вхідна частота імпульсів
- Gear** : Пропорція вхідної частоти та вихідної частоти (значення з плаваючою комою)
- Axis** : Ідентифікатор вузла CANopen сервоприводу Delta
- OutCnt** : Кількість вихідних імпульсів (одиниця: на одиницю користувача (PUU))
- OutSpd** : Вихідна частота імпульсів

Пояснення

1. Для зв'язку по CAN мікропрограмане забезпечення ПЛК версії 1.06 і пізніших і програмне забезпечення ISPSOft версії 3.07 і пізніших підтримують інструкцію. Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
2. Для зв'язку CAN інструкції INITC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції. Для зв'язку ECAT інструкції INITEC і ASDON (увімкнення сервоприводу) мають бути виконані перед виконанням цієї інструкції.

1. **Xno** може вказати лише X0.0, X0.2, X0.4, X0.6, X0.8 і X0.10 як точки введення, а операнд займає дві послідовні точки введення. Ви не можете виконати інструкцію, якщо точки введення є точками, вказаними для **Xno** вище. Після вибору вхідних точок автоматично вказується високошвидкісний лічильник. Якщо в програмі є інструкція DCNT (API 1004) або інструкція CSFO (API 2708) з таким же високошвидкісним лічильником, ПЛК спочатку виконує інструкцію, яка першою запускає лічильник. Точки введення та відповідні високошвидкісні лічильники показані в наступній таблиці.

Номер групи	1	2	3	4	5	6
Xno +0 вхідна точка (фаза A)	X0.0	X0.2	X0.4	X0.6	X0.8	X0.10
Xno +1 вхідна точка (фаза B)	X0.1	X0.3	X0.5	X0.7	X0.9	X0.11
Швидкісний лічильник номер	HC202	HC206	HC210	HC214	HC218	HC222
Прапор для реверсивного введення напрямку	SM270	SM271	SM272	SM273	SM274	SM275

4. Якщо високошвидкісні лічильники для інструкції можуть використовувати лише режим введення фази A/B, встановіть прапор для реверсування напрямку введення на ON, коли MPG підключений, але ще не обертався, а точка введення ПЛК увімкнена. Встановіть для функції визначення стану циклу виконання ПЛК значення OFF.
5. **InCnt** — кількість вхідних імпульсів. Використовуйте 32-розрядну змінну, щоб оголосити параметр.
5. **InSpd** — частота вхідних імпульсів. Використовуйте 32-розрядну змінну, щоб оголосити параметр з одиницею 1 Гц.
5. **Gear** — це пропорція вхідної частоти та вихідної частоти (значення з плаваючою комою). Формула полягає в тому, що вихідна частота дорівнює результату множення вхідної частоти Gear. Наприклад, якщо Gear — 2,5, а вхідна частота — 50, вихід — 125 (50*2,5). Примітка: вихідна кількість і вихідна частота округлюються до найближчої цілої цифри під час виведення.

8. **Axis** — це ідентифікатор вузла в діапазоні від 1 до 8 для зв'язку CAN, і він має бути в межах ідентифікатора вузла, який ініціюється інструкцією INITC. Для зв'язку ECAT ідентифікатор вузла коливається від 1 до 16 і він має бути в межах ідентифікатора вузла, ініційованого інструкцією INITEC, а пристроєм призначеного ідентифікатора має бути сервопривід.
8. **OutCnt** – кількість вихідних імпульсів. Використовуйте 32-розрядну змінну, щоб оголосити параметр з одиницею PUU.
8. **OutSpd** — це частота вихідних імпульсів. Використовуйте 32-розрядну змінну, щоб оголосити параметр з одиницею 1 Гц.
9. Кількість разів використання цієї інструкції не обмежена. Кілька інструкцій можуть використовувати одну групу вхідних точок як джерело підрахунку вхідних даних. Але вам потрібно очистити значення в лічильнику введення перед виконанням цієї інструкції; особливо коли ви використовуєте один вхід і кілька виходів. І ця інструкція виконується відповідно до номера осі, щоб записати інформацію про її вхід і вихід. Ту саму вісь можна використовувати кілька разів, але ви не повинні виводити ту саму вісь одночасно. Якщо ви використовуєте ту саму вісь для виведення кілька разів, PLC виводить перший початковий, а потім другий. Для зв'язку ECAT можна одночасно керувати до чотирьох підлеглих пристроїв. Якщо перевищено обмеження, ця інструкція НЕ виконується, а SM0 увімкнено, код помилки, що зберігається в SR0, становить 16#2003.
8. Інструкція не може бути використана мовою програмування ST, завданнями переривання або функціональним блоком, який викликається лише один раз.

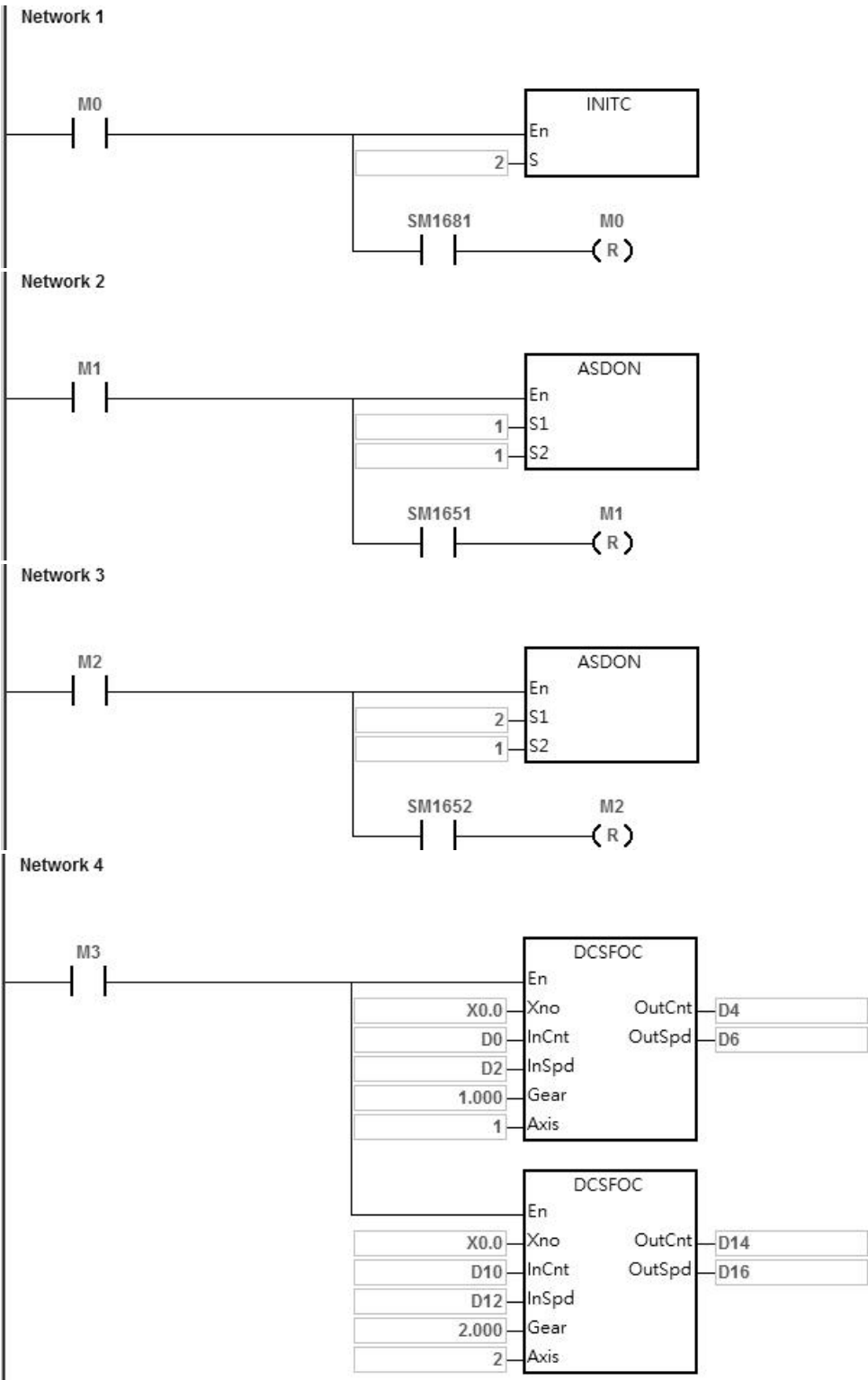
Примітка

1. ПЛК обчислює частоту вхідного імпульсу на основі ширини вхідного імпульсу (ON) у позитивному півперіоді. Якщо співвідношення ширини імпульсу для ON:широти імпульсу для OFF не є 1:1, ПЛК приймає ширину ON як стандарт для перетворення за замовчуванням.
2. Вхідний імпульс=ON означає, що світлодіодний індикатор вхідної точки горить. Якщо ви використовуєте MPG, але він ще не обертався, ви можете перевірити якщо світлодіод точки введення OFF та прапорець для реверсування напрямку введення активований.
3. Під час виконання цієї інструкції існує 20 мс різниці в часі зв'язку CANopen / ECAT. Якщо вас це турбує, замість цього використовуйте інструкцію CSFO, керуючи сервоприводом за допомогою імпульсного виходу.

приклад

- Коли M0 перемикається з OFF. на ON., ПЛК починає виконувати цю інструкцію для ініціалізації ідентифікаторів 1 і 2 сервовузла, доки SM1681 не буде ON.
- Коли M1 перемикається з OFF. на ON., ПЛК починає виконувати цю інструкцію для запуску сервоузла з ідентифікатором 1, а коли сервопривід запускається, SM1651 увімкнено, що означає, що сервопривід увімкнено.
- Коли M2 перемикається з OFF. на ON., ПЛК починає виконувати цю інструкцію, щоб запустити сервоузел з ідентифікатором 1, і коли сервопривід запускається, SM1652 увімкнено, що означає, що сервопривід увімкнено.

- Коли M3 перемикається з OFF. на ON., вхідні точки X0.0 і X0.1 підключаються до MPG для керування 2 сервоприводами. Пропорції для 1-го сервоприводу дорівнюють 1, а для 2-го сервоприводу — 2.



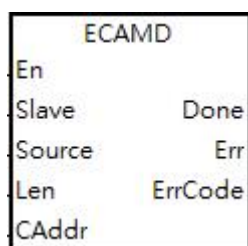
API	Інструкція				Операнд	опис
2813		ECAMD		Slave ~ ErrCode	Завантаження діаграми E-CAM на Slave	

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Slave								●					○	○		
Source								●	●				○	○		
Len								●					○	○		
CAddr								●					○	○		
Done		●	●	●												
Err		●	●	●												
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
Source		●				●							
Len		●				●							
CAddr		●				●							
Done	●												
Err	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ



- Slave** : Номер веденої станції сервоприводу
- Source** : З якої діаграми E-CAM почати завантаження
- Len** : Довжина діаграми E-CAM
- CAddr** : З якої адреси почати збереження діаграми E-CAM у сервоприводі
- Done** : Завантаження повного прапора

Пояснення

- Для виконання цієї інструкції вам потрібно використовувати PLC FW V1.06 або пізнішої версії та ISPSOFT V3.07 або пізнішої версії.
- Slave** – це місце, де ви зберігаєте завантажену діаграму E-CAM; кількість підтримуваних підлеглих станцій варіюється від 1 до 8. Після завершення завантаження прапорець **Done** увімкнено. Якщо номер підлеглої станції, який ви використовуєте, виходить за межі підтримуваного діапазону, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено.

3. **Source** — це місце, де ви вказуєте, з якої діаграми E-CAM почати завантаження; підтримувана довжина діаграми E-CAM становить від 1 до 16. Якщо номер діаграми E-CAM, який ви використовуєте, виходить за межі підтримуваної довжини, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено. Ви також можете використовувати номер регістру даних ПЛК, щоб вказати, з якої діаграми E-CAM почати завантаження в сервопривід, і вона приймає **Len+1** як довжину завантаження. Після завершення налаштування система починає послідовно зчитувати серво з указанного номера регістру даних (номера пристрою) до **Len +1 (32 біти)**. Якщо **Source** — D10, а **Len** — 10, система починає зчитувати від D10 до довжини 10+1, починаючи від D10/D11, D12/D13 до D30/D31 до сервоприводу.
3. **Len** — довжина діаграми E-CAM для завантаження; діапазон налаштувань від 1 до 720. Якщо довжина, яку ви використовуєте, виходить за межі підтримуваного діапазону, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено. Сервопривід підтримує до 800; якщо **Len +1** перевищує 800, ПЛК припиняє завантаження та вмикається прапор **Err** (помилка).
3. **CAddr** — це початкова адреса для збереження схеми E-CAM у сервоприводі; підтримуваний діапазон налаштувань становить від 0 до 798. Якщо початкова адреса, яку ви використовуєте, виходить за межі підтримуваного діапазону, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено.
3. Якщо підтримуваних 16 діаграм E-CAM недостатньо, ви можете завантажити діаграми E-CAM у реєстр файлів ПЛК під час редагування в редакторі E-CAM.
7. Ви можете використовувати 1-^е ~ 800 -^е сховище в сервоприводі. При створенні діаграм E-CAM з не такою кількістю точок (менше ніж 800), ви можете завантажити діаграми E-CAM за один раз, щоб заощадити час спілкування. Наприклад, якщо ви використовуєте точки 1-200 на діаграмі A та 201-400 на діаграмі B, точок на діаграмі A плюс точки на діаграмі B дорівнює 400 (менше 800), ви можете завантажити ці дві діаграми одночасно.
8. Після виконання цієї інструкції спочатку знімається позначка «**Done**», а потім надсилається діаграма E-CAM. Після передачі завершено, прапорець **Done** увімкнено. Якщо будь-яка помилка виникає від підлеглих пристроїв, прапор **Err** (помилка) вмикається. Якщо вам потрібно повторно надіслати діаграму E-CAM, вам потрібно зачекати один час сканування, перш ніж виконувати цю інструкцію знову.
9. Якщо прапорець **Err** (помилка) увімкнено, зверніться до наведених нижче визначень.

ErrCode	опис
0001	Цей номер веденої станції НЕ входить до підтримуваного діапазону.
0002	Цей номер діаграми E-CAM НЕ входить до підтримуваного діапазону.
0003	Цей CAddr. Адреса або довжина НЕ входять до підтримуваного діапазону.
0004	Немає відповіді від раба.
0005	Повідомлення про помилку, надіслане підлеглим під час завантаження або налаштування.
0006	ПЛК НЕ перебуває ні в режимі Delta Special Driver & AS Remote Mode, ні в Delta Special Драйвер і режим CANopen DS301.

API	Інструкція				Операнд								опис			
2814	D	ECAMS			Slave ~ ErrCode								Налаштування та планування параметрів E-CAM			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Slave								●					○	○		
Smode								●					○	○		
CAddr								●					○	○		
CLen								●					○	○		
Rpuu								●					○	○		
Cmode								●					○	○		
WbPuu								●					○	○		
OutT								●					○	○		
CaPuu								●					○	○		
CwPuu								●					○	○		
Opate								●					○	○		
Done		●	●	●												
Err		●	●	●												
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
Smode		●				●							
CAddr		●				●							
CLen		●				●							
RPuu			●				●						
Cmode		●				●							
WbPuu			●				●						
OutT		●				●							
CaPuu			●				●						
CwPuu			●				●						
ORate			●				●						
Done	●												
Err	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	—	AS

СИМВОЛ

DECAMS	
En	
Slave	Done
Smode	Err
CAddr	ErrCode
CLen	
Rpuu	
Cmode	
WbPuu	
OutT	
CaPuu	
CwPuu	
Orate	

Slave	: Номер веденої станції сервоприводу
Smode	: Source команд для кодера головного вісь
CAddr	: З якої адреси почати зберігання E-CAM діаграму в сервоприводі
CLen	: Вказана довжина діаграми E-CAM для виконання
RPuu	: Кількість імпульсів для головного E-CAM до обертати на повний цикл на 360
Cmode	: Приемний стан
WbPuu	: Кількість імпульсів, які E-CAM очікує перед тим залучення вперше (32-розрядне значення)
OutT	: Стан розчеплення
CaPuu	: Кількість імпульсів, які E-CAM очікує перед тим відключення (32-розрядне значення)
CwPuu	: Кількість імпульсів, які E-CAM очікує перед тим залучення знову (32-розрядне значення)
ORate	: Вихідний коефіцієнт для діаграми E-CAM (32-розрядне значення)
Done	: Завантаження повного прапора
Err	: Прапор помилки
ErrCode	: Код помилки

Пояснення

- Для виконання цієї інструкції вам потрібно використовувати PLC FW V1.06 або пізнішої версії та ISPSOft V3.07 або пізнішої версії.
- Ця інструкція призначена для першого налаштування параметрів, пов'язаних з E-CAM, або для налаштування діаграми E-CAM перед першим залученням. Якщо вам потрібно змінити діаграму E-CAM під час залучення, замість цього потрібно використовувати інструкцію ECAMC (API2816).
- Slave** — це підпорядкована сервостанція в діапазоні від 1 до 8. Якщо номер підлеглої станції, який ви використовуєте, виходить за межі підтримуваного діапазону, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено. Після модифікації та правильної відповіді вказаної веденої сервостанції прапор **Done** ввімкнеться.
- Smode** можна встановити джерело команд для кодера головної осі; варіанти, як показано нижче.

Source команд головної осі	опис
0	Вісь CAP (визначається функцією CAP)
1	Допоміжний кодер (P5-17)
2	Імпульсна команда (фазовий імпульс A/B) (P5-18)
3	Команда PR: значення абсолютної координати з команди положення
4	Вісь часу (1 мс)
5	Вісь синхронного захоплення (P5-77)
6	Аналоговий канал 1 (віртуальна вісь; одиниця: 10 В відповідає 1 М імпульсу/с)
12	Імпульсна команда (CW/CCW імпульс) (P5-18)
22	Імпульсна команда (Pulse/Dir pulse) (P5-18)
Інші номери	Враховується як варіант 2: імпульсна команда (фазовий імпульс A/B) (P5-18)

5. PLC встановлює параметр сервоприводу P5-88.Y відповідно до джерела команд головної осі. Коли джерелом команди є Pulse Command, вам потрібно налаштувати тип імпульсу за допомогою параметра сервоприводу P1-00, коли сервопривід вимкнено.
6. **CAddr** — це початкова адреса для збереження схеми E-CAM у сервоприводі; підтримуваний діапазон налаштувань становить від 0 до 794. Якщо початкова адреса, яку ви використовуєте, виходить за межі підтримуваного діапазону, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено. PLC встановлює параметр сервоприводу P5-81 (початкова адреса масиву даних).
7. **CLen** — задана довжина діаграми E-CAM для виконання; діапазон налаштувань від 5 до 720. Діаграма E-CAM розділена у **CLen**, а діаграма має містити дані **CLen** +1. Сервопривід підтримує 1-й ~ 800-й простір; якщо початкова адреса дорівнює 501, а для **CLen** встановлено значення 400, це розташування займає 902 (501+401) простору, що перевищує 800-й, інструкція НЕ виконується, а прапор **Err** (помилка) увімкнено. PLC встановлює параметр сервоприводу P5-82. Записуйте цей параметр, коли E-CAM зупиняється (P5-88, X=0).
8. **RPuu** — це місце, де ви встановлюєте кількість імпульсів для головного E-CAM для обертання на повний цикл на 360; сервоблок: ПУУ (пер блок користувача). Значення в **RPuu** має бути 32-розрядним і варіюватись від 10 до 1073741823. Коли значення перевищує верхню або нижню межу, воно розглядатиметься як максимальне або мінімальне значення. PLC встановлює параметр сервоприводу P5-83 і P5-84. Запишіть параметри, коли E-CAM зупиняється (P5-88, X0=0).
9. **Cmode** — це місце, де ви можете встановити умови залучення; варіанти, як показано нижче.

Режим	опис
0	Займіться негайно
1	DI: KAMEP ON
2	Будь-який із захоплення
Інші номери	Розглядається як режим 0, для негайного включення

10. PLC встановлює параметр сервоприводу P5-88.Z (час зачеплення).

11. WbPuu – це місце, де ви встановлюєте кількість імпульсів, які E-CAM чекає перед першим увімкненням. Значення в **WbPuu** має бути 32-розрядним і варіюватись від -1073741824 до +1073741823. Коли значення перевищує верхню або нижню межу, воно буде розглядатися як максимальне або мінімальне значення. Якщо значення дорівнює 0, це означає відсутність очікування перед першою взаємодією. PLC встановлює параметр сервоприводу P5-87.

12. OutT – це місце, де ви можете встановити умову роз'єднання; варіанти, як показано нижче. Варіанти 2, 4 і 6 не можна вибрати одночасно.

Режим	опис	Після відключення
0	Ніколи не відключався	-
1	DI: КАМЕРА ВИМКНЕНА	У статусі СТОП
2	Головна вісь отримує номер імпульсу який встановлюється CaPuu (P5-89).	У статусі СТОП
4	Головна вісь перевищує встановлене значення в CaPuu (P5-89).	Повернення до попереднього статусу заручених. Попередньо задіяний імпульс – CwPuu (P5-92).
6	Те саме, що 2; головна вісь отримує імпульс число, яке встановлюється CaPuu (P5-89), але тут E-CAM зберігає свою швидкість під час відключення. Головна вісь трохи перевищує значення налаштування в CaPuu (P5-89). Підходить для використання команди позиції PR одразу після роз'єднання.	У статусі СТОП
8	Робота з варіантом 1, 2 або 6; вимкніть E-CAM після її від'єднання.	Встановіть параметр сервоприводу P5-88.X0 на 0.
	Працюйте з варіантом 4, щоб уникнути тремтіння, коли він повернеться до попереднього стану	-
Інший чисел	Розглядається як режим 0, для негайного включення	

13. PLC встановлює параметр сервоприводу P5-88.U (час відключення).

14. CaPuu – це місце, де ви встановлюєте кількість імпульсів, які E-CAM чекає перед відключенням. Значення в **CaPuu** має бути 32-розрядним і варіюватись від -1073741824 до +1073741823. Коли значення перевищує верхню або нижню межу, воно буде розглядатися як максимальне або мінімальне значення. Якщо значення дорівнює 0, це означає відсутність очікування перед відключенням. PLC встановлює параметр сервоприводу P5-89.

15. CwPuu – це місце, де ви встановлюєте кількість імпульсів, які E-CAM чекає перед повторним увімкненням. Значення в **CwPuu** має бути 32-розрядним і варіюватись від -2147483648 до +2147483647. Якщо значення дорівнює 0, це означає відсутність очікування перед повторним залученням. PLC встановлює параметр сервоприводу P5-92.

16. **ORate** – це місце, де ви встановлюєте вихідний коефіцієнт для діаграми E-CAM. Значення в **ORate** має бути 32-розрядним цілим числом у діапазоні від -2147000000 до +2147000000. Коли значення перевищує верхню або нижню межу, воно буде розглядатися як максимальне або мінімальне значення. PLC встановлює параметр сервоприводу P5-19. Наприклад, встановіть **ORate** на 123000, і 123000 буде поділено на 1000000 сервоприводом. І у вас буде 0,123 як вихідний коефіцієнт. Встановіть **ORate** на 1500000, і 1500000 буде поділено на 1000000 сервоприводом. І у вас буде 1,5 як вихідний коефіцієнт.

17. Після виконання цієї інструкції спочатку знімається позначка «**Done**», а потім надсилається запит на встановлення параметрів E-CAM. Якщо передача працює правильно, прапорець **Done** ввімкнено. Якщо будь-яка помилка виникає від підлеглих пристроїв, прапор **Err** (помилка) вмикається. Якщо вам потрібно змінити параметри E-CAM, вам потрібно зачекати один час сканування, перш ніж виконувати цю інструкцію знову.

18. Якщо прапорець **Err** (помилка) увімкнено, зверніться до наведених нижче визначень.

ErrCode	опис
0001	Цей номер веденої станції HE входить до підтримуваного діапазону.
0003	Ця адреса завантаження (CAddr) або довжина (CLen) HE входить до підтримуваного діапазону.
0004	Немає відповіді від раба.
0005	Повідомлення про помилку, надіслане підлеглим під час завантаження або налаштування.
0006	ПЛК HE перебуває ні в режимі Delta Special Driver & AS Remote Mode, ні в Delta Special Драйвер і режим CANopen DS301.

API	Інструкція				Операнд						опис						
2815		ECAMA			Slave ~ ErrCode						Активация управління E-CAM і відображення стану						
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Slave								●					○	○			
CAMEn	●	●	●	●													
CAMSt								●									
MIPuu								●									
SOPuu								●									
Err		●	●	●													
ErrCode								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
CAMEn	●												
CAMSt		●				●							
MIPuu			●				●						
SOPuu			●				●						
Err	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ

ECAMA	
En	
Slave	MIPuu
CAMEn	SOPuu
CAMSt	Err
	ErrCode

- Slave : Номер веденої станції сервоприводу
- CAMEn : Активуйте керування E-CAM
- CAMSt : Відображення стану E-CAM
- MIPuu : Кількість імпульсів, введених Master (одиниця: PUU)
- SOPuu : Кількість імпульсів, виданих вказаним підлеглим (одиниця: PUU)
- Err : Прапор помилки
- ErrCode : Код помилки

Пояснення

1. Для виконання цієї інструкції вам потрібно використовувати PLC FW V1.06 або пізнішої версії та ISPSoft V3.07 або пізнішої версії.

2. Інструкції INITC і ASDON (Servo-ON) мають бути виконані перед виконанням цієї інструкції.

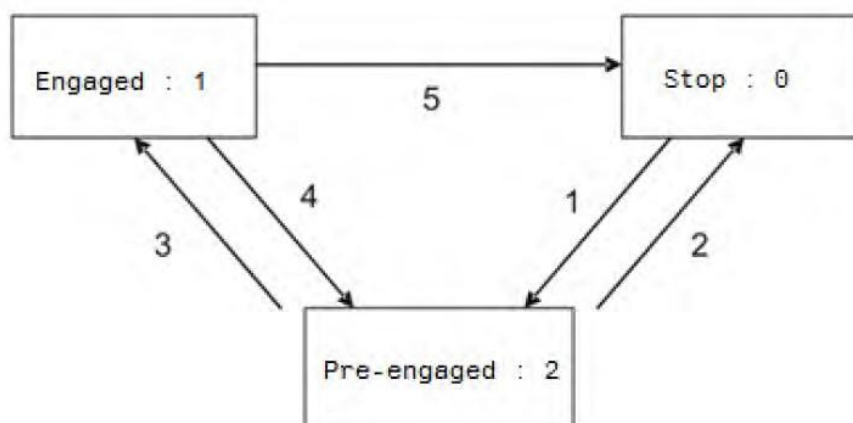
3. **Slave** — це сервопривідна станція, де ви можете вказати прийом команд; діапазон підтримуваних підлеглих станцій від 1 до 8. Якщо номер підлеглої станції, який ви використовуєте, виходить за межі підтримуваного діапазону, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено. Якщо зазначена ведена сервостанція не може зв'язатися, прапор **Err** (помилка) також увімкнено.

4. **CAMEn** можна активувати керування E-CAM. Коли значення BOOL у **CAMEn** увімкнено, це означає, що керування E-CAM активовано. Коли значення BOOL у **CAMEn** вимкнено, це означає, що керування E-CAM НЕ активовано. PLC встановлює параметр сервоприводу P5-88, X=0 відповідно до статусу в **CAMEn**.

5. **CAMSt** — це місце, де ви можете знайти статус E-CAM у сервоприводі відповідно до параметра сервоприводу P5-88. S: показує статус залучення (лише для читання, налаштування неможливе)

0: Стоп

1: Статус залучення



6. Опис стану:

- **Зупинка:** це початковий стан E-CAM. E-CAM не працює з головним імпульсом. Якщо функцію E-CAM вимкнено (P5-88, X=0), вона повертається до цього стану.
- **Попереднє залучення:** коли встановлено стан залучення (шлях 1), він переходить у цей статус. E-CAM все ще не працює з головним імпульсом.
- **Залучення:** коли він досягає статусу попереднього залучення (шлях 3), він переходить у цей статус. E-CAM починає працювати з головним імпульсом.

7. Опис шляху:

- **Шлях 1 :** коли встановлено стан зчеплення (P5-88.Z), стан – Зупинка → Попередньо зчеплено. Відведення пульсу визначається за R5-87.
- **Шлях 2 :** коли функцію E-CAM вимкнено (P5-88.X=0), вона повертається до стану зупинки.

- **Шлях 3** : коли він перебуває в статусі попереднього залучення, він має статус Попередньо залучений → Залучений.
- **Шлях 4** : коли встановлено стан роз'єднання (P5-88, U=4), стан «Задіяно → Попередньо задіяно». Випереджаючий імпульс визначається P5-92 (доступно для FM V1.006sub04 або пізніших).
- **Шлях 5**: коли встановлено роз'єднаний стан (P5-88, U=1, 2, 6) або функцію E-CAM вимкнено (P5-88, X=0), стан «Задіяно → Зупинити » . .

8. MIPuu можна знайти кількість імпульсів, отриманих вказаним підлеглим від головного. Одиниця на одиницю користувача (PUU) сервоприводу. Це значення відображається відповідно до параметра сервоприводу P5-86. Коли інструкція зупиняється, це значення також НЕ оновлюється.

9. SOPuu — це місце, де ви можете знайти кількість імпульсів, виданих вказаним Slave. Одиниця на одиницю користувача (PUU) сервоприводу. Це значення відображається відповідно до параметра сервоприводу CMD_O. Коли інструкція зупиняється, це значення також НЕ оновлюється.

10. Якщо прапорець **Err** (помилка) увімкнено, зверніться до наведених нижче визначень.

ErrCode	опис
0001	Цей номер веденої станції HE входить до підтримуваного діапазону.
0004	Немає відповіді від раба.
0005	Повідомлення про помилку, надіслане підлеглим під час завантаження або налаштування.
0006	ПЛК HE перебуває ні в режимі Delta Special Driver & AS Remote Mode, ні в Delta Special Драйвер і режим CANopen DS301.
0007	Підлеглий не ініціюється.

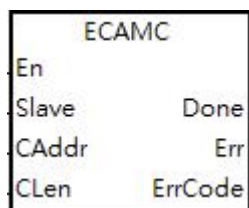
API	Інструкція				Операнд								опис			
2816		ECAMC			Slave ~ ErrCode								Зміна початкової адреси для діаграми E-CAM і довжини діаграми E-CAM у сервоприводі			

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Slave								●					○	○		
CAddr								●					○	○		
CLen								●					○	○		
Done		●	●	●												
Err		●	●	●												
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
CAddr		●				●							
CLen		●				●							
Done	●												
Err	●												
ErrCode		●				●							

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
—	AS	—

СИМВОЛ



- Slave** : Номер веденої станції сервоприводу
- CAddr** : З якої адреси почати збереження діаграми E-CAM у сервоприводі
- CLen** : Вказана довжина діаграми E-CAM для виконання
- Done** : Прапор завершення модифікації
- Err** : Прапор помилки
- ErrCode** : Код помилки

Пояснення

1. Для виконання цієї інструкції вам потрібно використовувати PLC FW V1.06 або пізнішої версії та ISPSOFT V3.07 або пізнішої версії.

2. Ця інструкція призначена для зміни параметрів діаграми E-CAM. Перед використанням цієї інструкції вам потрібно завантажити схему E-CAM на вказаний сервопривід.

3. **Slave** — це сервопривідна станція, де ви можете вказати прийом команд; діапазон підтримуваних підлеглих станцій від 1 до 8. Якщо номер підлеглої станції, який ви використовуєте, виходить за межі підтримуваного діапазону, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено. Якщо зазначена ведена сервостанція не може зв'язатися, прапор **Err** (помилка) також увімкнено.

Після зміни параметрів E-CAM і правильної відповіді вказаної веденої станції сервоприводу прапор **Done** ввімкнено.

4. CAddr — це початкова адреса для збереження схеми E-CAM у сервоприводі; підтримуваний діапазон налаштувань становить від 0 до 794. Якщо початкова адреса, яку ви використовуєте, виходить за межі підтримуваного діапазону, інструкція НЕ виконується. І прапор **Err** (помилка) увімкнено. PLC встановлює параметр сервоприводу P5-81 (початкова адреса масиву даних).

5. CLen — указана довжина діаграми E-CAM для виконання; його діапазон налаштувань становить від 5 до 720. Діаграма E-CAM розділена на **CLen**, і діаграма повинна містити дані **CLen** +1. Сервопривод підтримує 1- й ~ 800- й простір; якщо початкова адреса дорівнює 501, а для **CLen** встановлено значення 400, це розташування займає 902 (501+401) простору, що перевищує 800- й, інструкція НЕ виконується, а прапор **Err** (помилка) увімкнено. PLC встановлює параметр сервоприводу P5-82. Записуйте цей параметр, коли E-CAM зупиняється (P5-88, X=0).

6. Після виконання цієї інструкції спочатку знімається позначка **«Done»**, а потім надсилається запит на модифікацію. Після зміни параметрів E-CAM і правильної відповіді вказаної веденої станції сервоприводу прапор **Done** ввімкнено. Якщо будь-яка помилка виникає від підлеглих пристроїв, прапор **Err** (помилка) вмикається. Якщо вам потрібно змінити початкову адресу та довжину E-CAM, вам потрібно зачекати один час сканування, перш ніж виконувати цю інструкцію знову.

7. Якщо прапорець **Err** (помилка) увімкнено, зверніться до наведених нижче визначень.

ErrCode	опис
0001	Цей номер веденої станції НЕ входить до підтримуваного діапазону.
0002	Цей номер діаграми E-CAM НЕ входить до підтримуваного діапазону.
0003	Ця адреса завантаження (CAddr) або довжина (CLen) НЕ входить до підтримуваного діапазону.
0004	Немає відповіді від раба.
0005	Повідомлення про помилку, надіслане підлеглим під час завантаження або налаштування.
0006	ПЛК НЕ перебуває ні в режимі Delta Special Driver & AS Remote Mode, ні в Delta Special Драйвер і режим CANopen DS301.

API	Інструкція			Операнд								опис					
2817	D	TQC		Node, TarTQ, ..., Err								Контроль крутного моменту					
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Node								●					○	○			
TarTQ								●					○	○			
LSpeed								●					○	○			
CurTQ								●									
Reac		●	●	●				●									
Err		●	●	●				●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●				●							
TarTQ		●				●							
LSpeed			●				●						
CurTQ		●				●							
Reac	●												
Err	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

СИМВОЛ

DTQC	
En	
Node	CurTQ
TarTQ	Reac
Lspeed	Err

Node : Ідентифікатор вузла дисків Delta

TarTQ : Зазначений цільовий крутний момент

LSpeed : Обмеження максимальної швидкості

CurTQ : Поточне значення крутного моменту

Reac : Цільовий крутний момент досягнуто

Err : Прапор помилки зв'язку

Пояснення

- Перед виконанням цієї інструкції переконайтеся, що всі диски Delta ініціалізовано.
- Для зв'язку по CAN мікропрограмне забезпечення ПЛК версії 1.06 і пізніших і програмне забезпечення ISPSOFT версії 3.07 і пізніших підтримують інструкцію. Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOFT V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для сервоприводів Delta: встановіть параметр P1-01 на 3 (T: режим керування крутним моментом), вимкніть і знову увімкніть живлення, щоб налаштування було ефективним.

Для приводів Delta Vector Control: використовуйте інструкцію ASDON, щоб установити режим керування крутним моментом і встановіть параметр P11-33 (Source команди крутного моменту) на 3 (CANopen) на приводах двигунів змінного струму Delta, які підтримують режим керування крутним моментом.

4. **Node** : для зв'язку CAN ідентифікатор вузла сервоприводів Delta та інверторів коливається від 1 до 8 (сервоприводи Delta) і від 21 до 28 (інвертори Delta). Якщо налаштування виходить за межі діапазону, прапор Err буде увімкнено. Для зв'язку ECAT діапазон ідентифікатора вузла від 1 до 16, а пристрій із призначеним ідентифікатором має бути сервоприводом. Якщо налаштування виходить за межі діапазону, прапор Err буде увімкнено.
5. **TarTQ** : заданий цільовий крутний момент; пропонується перевірити посібники Delta Servo Drives і AC Motor Drives перед налаштуванням. Ви можете змінити налаштування під час виконання цієї інструкції.

Привід Типи	Ідентифікатор вузла Діапазон	TarTQ
Сервопривід	1~8 (CAN) 1-16 (ECAT)	Налаштування параметра сервоприводу P1-12 (внутрішнє обмеження крутного моменту 1), допустимий діапазон: -300~+300, одиниця: 1%; значення, що перевищує верхню/нижню межу, буде розглядатися як значення верхньої/нижньої межі. Для FW V1.08.40 або пізнішої версії підтримується режим крутного моменту високої роздільної здатності ASDA-A3/B3; дійсний діапазон: -3000~+3000, одиниці виміру: 0,1%, значення, що перевищує верхню/нижню межу, розглядатиметься як значення верхньої/нижньої межі. Знак +/- вказує на напрямок вперед/назад. Сервопривод працює відповідно до параметра режиму керування крутним моментом P1-01 (напрям вихідного крутного моменту); встановіть DI сервоприводу на TCM0 = ON і TCM1 = OFF, щоб вибрати внутрішню команду крутного моменту на 1; пропонується використовувати інструкцію SOPRW для очищення значення в параметрі сервоприводу P1-12 перед увімкненням. Оскільки значення P1-12 за замовчуванням HE 0%, він почне працювати, коли сервопривід увімкнено.
АС Моторний привід	21~28 (CAN)	Налаштування параметра приводу двигуна змінного струму P11-27 (команда максимального моменту); індекс 6071-00 (цільовий крутний момент) та індекс 6072-00 (найбільший вихідний крутний момент); допустимий діапазон: - 500~+500; одиниця: 1%; значення, що перевищує верхню/нижню межу, буде розглядатися як значення верхньої/нижньої межі.

6. **LSpeed** : максимальна швидкість; це використовується для запобігання перевищенню обмеження швидкості, якщо приводи не можуть завершити

задані команди крутного моменту. Ви можете змінити налаштування під час виконання цієї інструкції.

Привід Типи	Ідентифікатор вузла Діапазон	LSpeed
Сервопривід	1~8 (CAN) 1-16 (ECAT)	Установка параметра сервоприводу P1-55 (обмеження максимальної швидкості); дійсний діапазон від 10 до рейтингу швидкості кожної моделі; блок: 1 об/хв (роторний).
АС Моторний привід	21~28 (CAN)	Налаштування параметрів електроприводу змінного струму 1. P01-00 = LSpeed (максимальна вихідна частота); допустимий діапазон: 0~59900; одиниця: 0,01 Гц; значення, що перевищує верхню/нижню межу, буде розглядатися як верхня/нижня межа 2. P11-37=100% (обмеження швидкості руху) 3. P11-38=100% (обмеження швидкості заднього ходу) Вам потрібно встановити параметр P11-36 (вибір обмеження швидкості) на 0, а коли параметр P11-36 встановлено на 0, обмеження швидкості визначається налаштуваннями в P11-37 і P11-38.

7. **CurTQ** : поточне значення крутного моменту; одиниця: 0,1%; значення тут таке саме, як значення, показане у відповідному SR. Зверніться до інструкції API 2803 DRVIC, щоб дізнатися більше про відповідний SR.

8. **Reach** – це результат порівняння між **CurTQ** і **TarTQ** , показаний у 0,1% як його одиниця. Коли **CurTQ** $\geq$

TarTQ , прапор **Reach** увімкнено. Коли **CurTQ** $<$ (**TarTQ** – 5), прапор **Reach** OFF. Значення затримки відгуку становить 5.

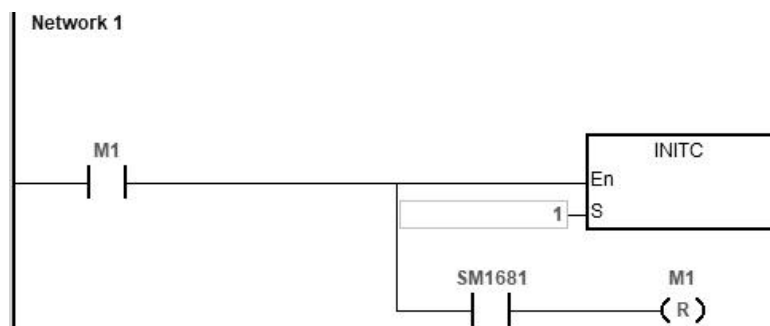
9. Якщо вузли приводів відповідають повідомленнями про помилку або не відповідають, ПЛК встановить прапор **Err** на ON.

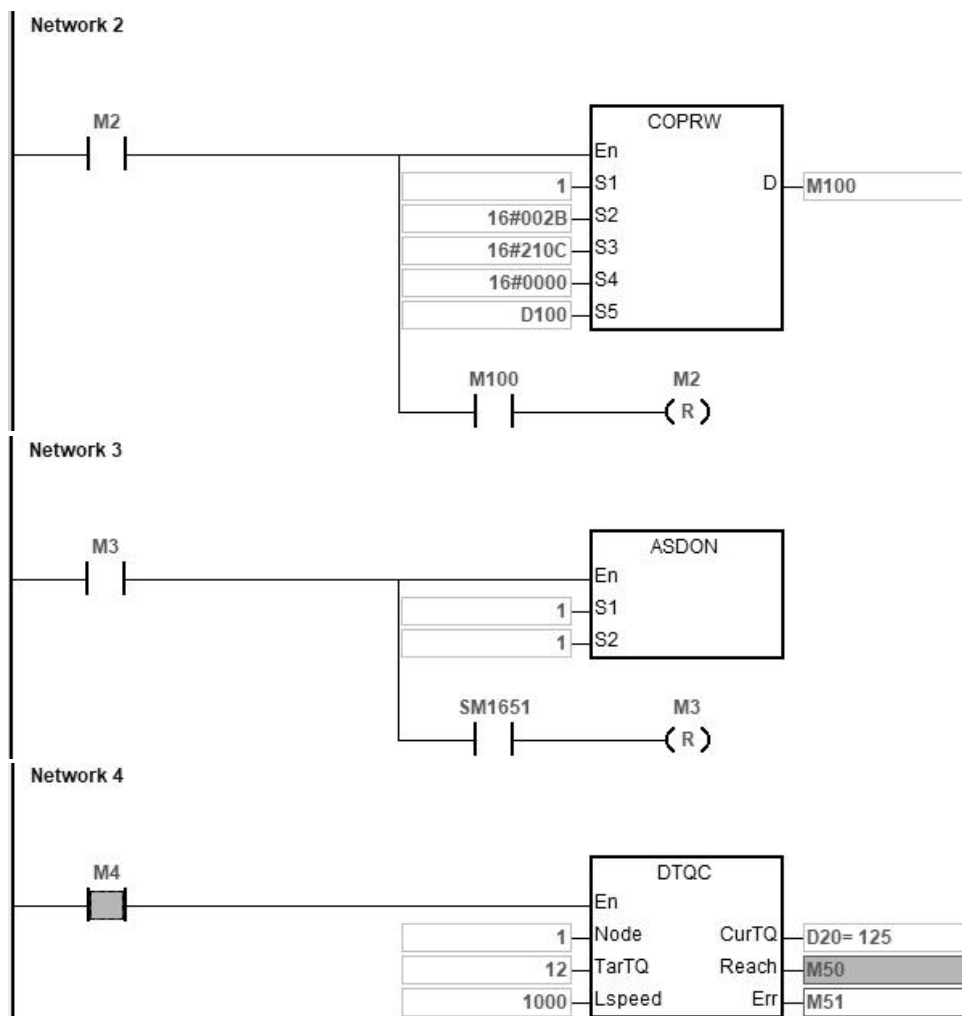
Для отримання додаткової інформації про коди помилок зверніться до SR658 і SR659 з інструкції API 2803 DRVIC.

10. Так як значення в команді крутного моменту і швидкості можуть відрізнятися. Рекомендується перевірити посібники для сервоприводів Delta та приводів змінного струму перед налаштуванням. Ця інструкція НЕ перевіряє та не обмежує наявність недейсних значень.

Приклад 1 (зв'язок із Delta servo ASD-A2M)

1. Налаштуйте ASD-A2
 - A. Встановіть параметр P1-01 на 3 (T: режим керування моментом).
 - B. Встановіть сервопривід DI: TCM0 = ON і TCM1 = OFF, щоб вибрати внутрішню команду крутного моменту на 1.
Якщо значення в DI є значенням за замовчуванням, ви можете встановити DI3 на ON, а потім команда крутного моменту встановлюється DI3 і DI4.
 - C. Вимкніть і знову увімкніть живлення, щоб налаштування було ефективним.
2. Коли M1 змінюється з OFF на ON, інструкція INITC починає ініціалізувати сервоприводи на вузлі ID 1, доки SM1681 увімкнено.
3. Коли M2 змінюється з OFF на ON, встановіть параметр сервоприводу P1-12 на 0. Використовуйте інструкцію COPRW, щоб записати значення з D100 у P1-12 (**S3** =16#210B). Пам'ятайте, що використовуйте інструкцію COPRW, щоб очистити значення в параметрі сервоприводу P1-12 перед увімкненням. Оскільки значення P1-12 за замовчуванням НЕ 0%, він почне працювати, коли сервопривід увімкнено.
4. Коли M3 змінюється з OFF на ON, інструкція ASDON починає вмикати серво на вузлі ID 1. Коли SM1651 увімкнено, це вказує на Servo-ON.
5. Коли M4 змінюється з OFF на ON, параметр сервоприводу P1-12 = 12%, сервопривід у вузлі ID 2 рухається зі швидкістю 1000 об/хв, доки M4 не вимкнеться..

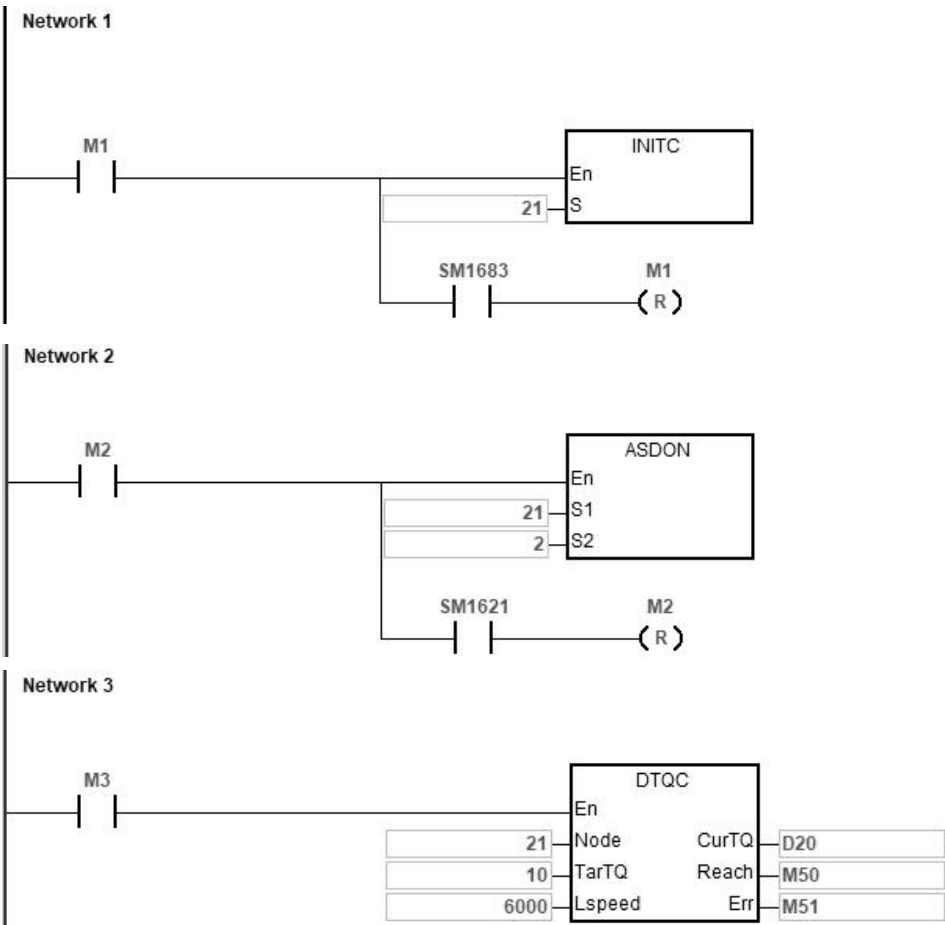




6

Приклад 2 (зв'язок з інверторами Delta)

1. Налаштуйте VFD-C2000 вручну
 - A. Встановіть параметр P09-36 (підпорядкована адреса CANopen) у діапазон 21~28 і P09-37 (швидкість CANopen) у значення 0 (1 МГц) і P09-40 (метод декодування CANopen) у значення 1 (увімкнати стандартний протокол CANopen DS402).
 - B. Встановіть параметр P00-20 (джерело команди головної частоти (AUTO)) на 6 (карта зв'язку CANopen), P00-21 (джерело команди роботи) на 3 (карта зв'язку CANopen).
 - C. Встановіть параметр P11-36 (вибір обмеження швидкості) на 0, а коли параметр P11-36 встановлено на 0, обмеження швидкості визначається налаштуваннями в P11-37 і P11-38.
2. Коли M1 змінюється з OFF на ON, інструкція INITC починає ініціалізувати привод двигуна змінного струму на вузлі ID 21, доки SM1683 не буде ON.
3. Коли M2 змінюється з OFF на ON, почніть керування крутним моментом на приводі двигуна змінного струму на вузлі ID 21.
4. Коли M3 змінюється з OFF на ON, цільовий крутний момент приводу двигуна змінного струму встановлюється на 10%, а привід двигуна змінного струму на вузлі ID 21 рухається з максимальною вихідною частотою 60,00 Гц, доки M3 не вимкнеться.



API	Інструкція			Операнд								опис					
2818	D	TQLC		Node 、 TarSpd 、 ... 、 Err								Швидкість регулювання крутного моменту					

пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Node								●					○	○		
TarSpd								●					○	○		
LimitTQ								●					○	○		
CurTQ								●								
Reac		●	●	●				●								
Err		●	●	●				●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Node		●				●							
TarSpd			●				●						
LimitTQ		●				●							
CurTQ		●				●							
Reac	●												
Err	●												

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	-	AS

Символ :

DTQLC
En
Node
TarSpd
LimitTQ
CurTQ
Reac
Err

Node : Ідентифікатор вузла дисків Delta

TarSpd : Зазначена цільова швидкість

LimitTQ : Максимальний крутний момент

CurTQ : Цільовий крутний момент досягнуто

Reac : Прапор помилки зв'язку

Err : Цільовий крутний момент досягнуто

Пояснення

- Перед виконанням цієї інструкції переконайтеся, що всі диски Delta ініціалізовано.
- Для зв'язку по CAN мікропрограмне забезпечення ПЛК версії 1.06 і пізніших і програмне забезпечення ISPSOFT версії 3.07 і пізніших підтримують інструкцію. Зв'язок ECAT доступний для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOFT V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Для сервоприводів Delta: встановіть параметр P1-01 на 3 (T: режим керування крутним моментом), вимкніть і знову ввімкніть живлення, щоб налаштування було ефективним.

Для приводів Delta Vector Control: використовуйте інструкцію ASDON, щоб установити режим керування крутним моментом і встановіть параметр P00-20 (джерело команди головної частоти (AUTO)) на 6 (карта зв'язку CANopen) на приводах двигунів змінного струму Delta, які підтримують режим керування крутним моментом.

4. Node : для зв'язку CAN ідентифікатор вузла сервоприводів Delta та інверторів коливається від 1 до 8 (сервоприводи Delta) і від 21 до 28 (інвертори Delta). Якщо налаштування виходить за межі діапазону, прапор Err буде увімкнено. Для зв'язку ECAT діапазон ідентифікатора вузла від 1 до 16, а пристрій із призначеним ідентифікатором має бути сервоприводом. Якщо налаштування виходить за межі діапазону, прапор Err буде увімкнено.

5. TarSpd : задана цільова швидкість

Привід Типи	Ідентифікатор вузла Діапазон	TarSpd Діапазон налаштування	TarSpd
Сервопривід	1~8 (CAN)	-60000 ~ 60000 (0,1 об/хв)	Знак +/- вказує на напрямок вперед/назад.
	1-16 (ECAT)	Приклад: введення 6005 вказує на 600,5 об./хв	
Двигун змінного струму Привід	21~28 (CAN)	-32768 ~ 32767 (1 об/хв) Приклад: введіть 605, що вказує на 605 об/хв	

6. LimitTQ : максимальне значення крутного моменту; після запуску інструкції налаштування застосовуються до сервоприводу. Ви НЕ можете змінювати налаштування під час виконання цієї інструкції.

Встановлення параметра сервоприводу P1-12 (внутрішнє обмеження крутного моменту 1), допустимий діапазон: -300~+300, одиниці: 1%; значення, що перевищує верхню/нижню межу буде розглядатися як значення верхньої/нижньої межі. Для FW V1.08.40 або пізнішої версії підтримується режим крутного моменту високої роздільної здатності ASDA-A3/B3; дійсний діапазон: -3000~+3000, одиниці виміру: 0,1%, значення, що перевищує верхню/нижню межу, розглядатиметься як значення верхньої/нижньої межі. Параметр P1-02 (налаштування обмеження швидкості та крутного моменту (сили); Встановіть серво DI на TCM0 = ON і TCM1 = OFF, щоб вибрати внутрішню команду крутного моменту на 1. Рекомендується використовувати інструкцію COPRW, щоб очистити значення в параметрі сервоприводу P1-12 перед увімкненням. Оскільки значення P1-12 за замовчуванням НЕ 0%, він почне працювати, коли сервопривід увімкнено. Припустимо, що в P1-12 введено значення 30, і ви отримаєте наступне, як показано в таблиці:

Допустимий діапазон крутного моменту (сила).	Обмеження прямого моменту (сили).	Обмеження зворотного моменту (сили).
-30% ~ +30%	30%	-30%

Налаштування параметра приводу двигуна змінного струму P11-17 (обмеження крутного моменту двигуна прямого ходу, квадрант I) і P11-19 (двигун заднього ходу обмеження крутного моменту, квадрант III), допустимий діапазон: 0~+500, одиниця: 1%; значення, що перевищує верхню/нижню межу, буде розглядатися як значення верхньої/нижньої межі. Припустимо, що в P11-17 і P11-19 введено значення 30, і ви отримаєте наступне, як показано в таблиці:

Елемент налаштування	P11-17	P11-19	Допустимий діапазон крутного моменту (сила).	Обмеження прямого моменту (сили).	Обмеження зворотного моменту (сили).
Значення	+30	+30	-30% ~ +30%	30%	-30%

7. CurTQ : поточне значення крутного моменту; одиниця: 0,1%; значення тут таке саме, як значення, показане у відповідному SR.

Зверніться до інструкції API 2803 DRVIC, щоб дізнатися більше про відповідний SR.

8. Reach – це результат порівняння між **CurTQ** і **TarTQ**, показаний у 0,1% як його одиниця. Коли **CurTQ** >=

TarTQ, прапор **Reach** увімкнено. Коли **CurTQ** < (**TarTQ** – 5), прапор **Reach** OFF. Значення затримки відгуку становить 5.

9. Якщо вузли приводів відповідають повідомленнями про помилку або не відповідають, ПЛК встановить прапор **Err** на ON. Для отримання додаткової інформації про коди помилок зверніться до SR658 і SR659 з інструкції API 2803 DRVIC.

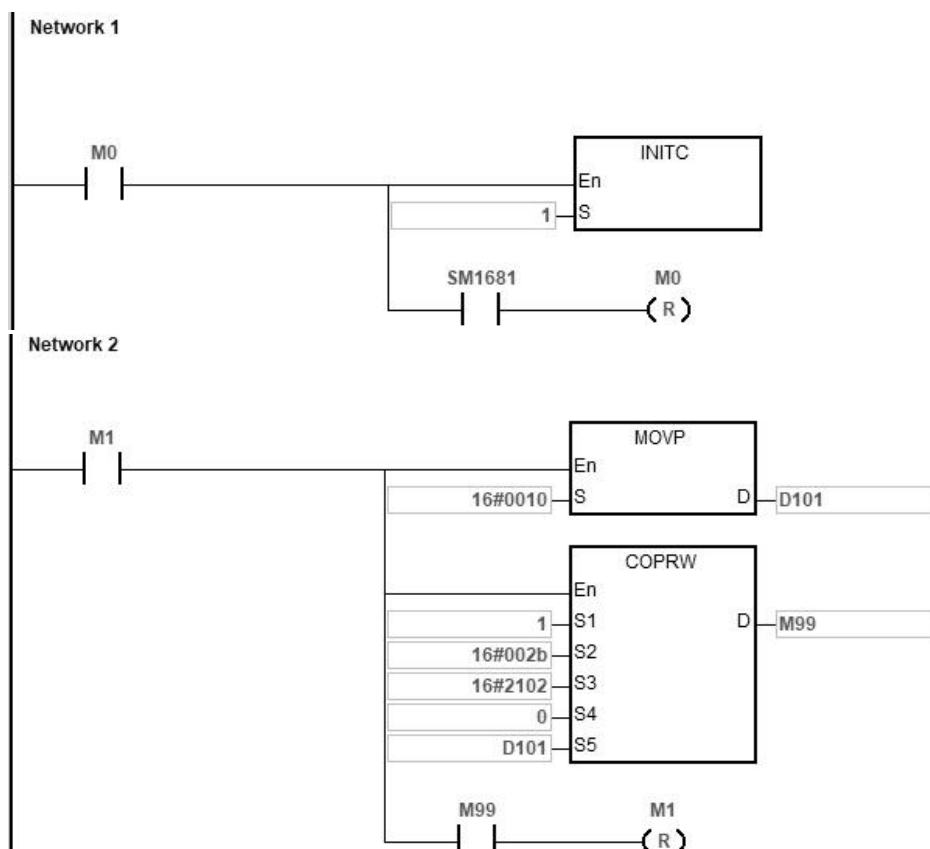
10. Так як значення в команді крутного моменту і швидкості можуть відрізнятися. Рекомендується перевірити посібники для сервоприводів Delta та приводів змінного струму перед налаштуванням. Ця інструкція НЕ перевіряє та не встановлює жодних обмежень, якщо є недійсне налаштування чи значення.

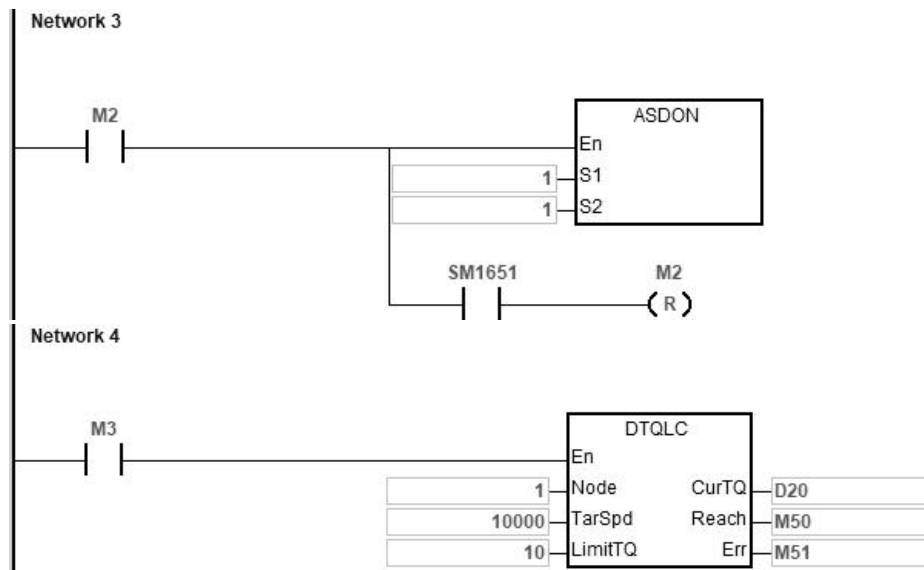
11. У момент, коли інструкція закрита, усі обмеження на значення крутного моменту будуть зняті, щоб привід міг виконувати інші команди позиціонування.

Приклад 1 (зв'язок із Delta servo ASD-A2M)

1. Налаштуйте ASD-A2
 - A. Встановіть параметр P1-01 (режим роботи) на 0001 (режим позиції)
 - B. Встановіть параметр P1-02 (обмеження швидкості та крутного моменту (сили)) на 0010 (увімкнено) або скористайтесь інструкцією COPRW, щоб встановити цей параметр.
 - C. Встановіть сервопривід DI: TCM0 = ON і TCM1 = OFF, щоб вибрати внутрішню команду крутного моменту на 1. Якщо значення в DI є значенням за замовчуванням, ви можете встановити DI3 на ON, а потім команда крутного моменту встановлюється DI3 і DI4.
2. Коли M0 змінюється з OFF на ON, інструкція INITC починає ініціалізувати сервоприводи на вузлі ID 1, доки SM1681 увімкнено.
3. Коли M1 змінюється з OFF на ON, використовуйте інструкцію COPRW, щоб установити параметр сервоприводу P1-02 (обмеження швидкості та крутного моменту (сили)) на 0010 (увімкнено).
4. Коли M2 змінюється з OFF на ON, інструкція ASDON починає вмикати серво на вузлі ID 1. Коли SM1651 увімкнено, це вказує на Servo-ON.
5. Коли M3 змінюється з OFF на ON, параметр сервоприводу P1-12 = 10%, сервопривід у вузлі ID 1 рухається зі швидкістю 1000 об/хв, доки M3 не вимкнеться.

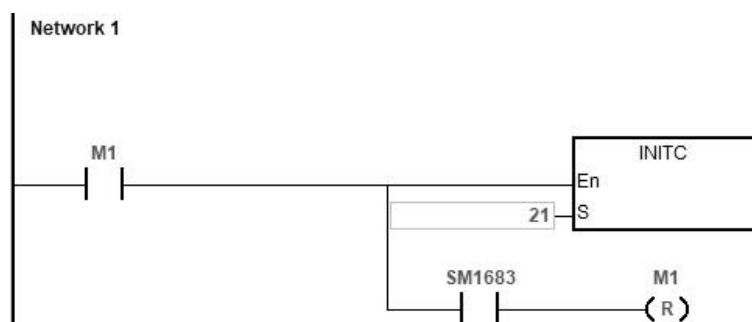
6

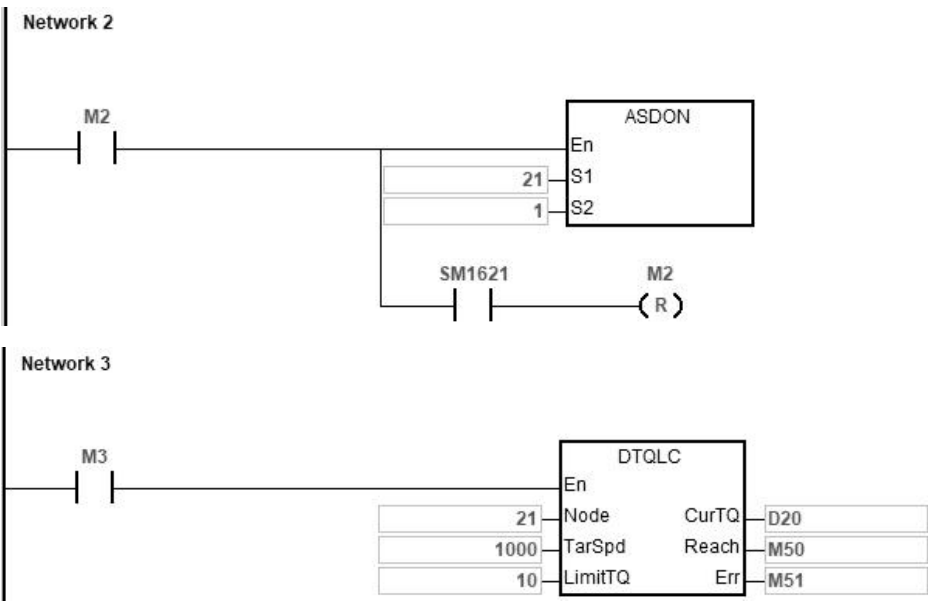




Приклад 2 (зв'язок з інверторами Delta)

1. Налаштуйте VFD-C2000 вручну
 - A. Встановіть параметр P09-36 (підпорядкована адреса CANopen) у діапазон 21~28 і P09-37 (швидкість CANopen) у значення 0 (1 МГц) і P09-40 (метод декодування CANopen) у значення 1 (увімкнути стандартний протокол CANopen DS402).
 - B. Встановіть параметр P00-20 (джерело команди головної частоти) на 6 (карта зв'язку CANopen), P00-21 (джерело команди роботи) на 3 (карта зв'язку CANopen).
2. Коли M1 змінюється з OFF на ON, інструкція INITC починає ініціалізувати привод двигуна змінного струму на вузлі ID 21, доки SM1683 увімкнено.
3. Коли M2 змінюється з OFF. на ON., почніть керування крутним моментом на приводі двигуна змінного струму на вузлі ID 21.
4. Коли M3 змінюється з OFF. на ON., обмеження крутного моменту приводу двигуна змінного струму встановлюється на 10%, а привід двигуна змінного струму на вузлі ID 21 рухається зі швидкістю 1000 об/хв, доки M3 не вимкнеться.



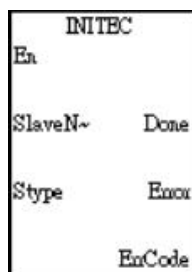


API	Інструкція				Операнд								опис			
2820		INITEC			SlaveNo , SType , Done , Error , ErrCode								Ініціалізація підлеглих пристроїв для зв'язку EtherCAT			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
SlaveNo								●					O	O		
SType								●	●							
Done		●	●	●				●								
Error		●	●	●				●								
ErrCode								●								

Тип дани	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
SlaveNo		●				●							
SType		●				●							
Done	●												
Error	●												
ErrCode		●											

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



SlaveNo : Кількість підлеглих пристроїв для ініціалізації

SType : Рабський тип

Done : Прапор завершення

Error : Прапор помилки

ErrCode : Код помилки

Пояснення

- Ця інструкція доступна для ПЛК AS із вбудованим програмним забезпеченням версії 1.14.00 або пізнішої та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Перед виконанням цієї інструкції переконайтеся, що плату зв'язку AS-FECAT встановлено на ПЛК.
- SlaveNo** має вказувати значення в діапазоні від 1 до 16. (AS-FECAT підтримує максимум 16 підлеглих пристроїв EtherCAT.) Якщо значення налаштування перевищує цей діапазон, ПЛК вважатиме це помилкою ініціалізації.
Примітка. Ідентифікатори підлеглих станцій повинні починатися з 1, а спосіб встановлення ідентифікатора підлеглих шляхом пропуску деяких номерів станцій неприпустимий.
- SType** — це тип підлеглих пристроїв і це змінна Array of Word, наприклад, якщо для **SlaveNo** встановлено значення 8, для цієї змінної також має бути встановлено змінну Array[8]. Коди типів і відповідні диски Delta перераховані в наступній таблиці.

Введіть код	1	2	3	інші
Модель приводу	ASDA-A2-E	ASDA-A3-E ASDA-B3-E	C2000 CH2000	Зарезервований

Примітка. Якщо код типу не збігається з фактично підключеним приводом, ПЛК встановить прапор помилки на ON і повідомити про код помилки.

5. **Done** – прапор завершення ініціалізації. Рекомендується почати наступне керування положенням після завершення ініціалізації.
6. **Error** — це позначка помилки ініціалізації, а **ErrCode** — код помилки. Перегляньте коди помилок і причини в наступній таблиці.

Код помилки	причина
16#0000	Помилки немає
16#0001	Кількість підлеглих поза діапазоном.
16#0003	Помилка налаштування коду підлеглого типу
16#0004	Час очікування під час спілкування з AS-FECAT
16#0005	Підлеглий SDO використовується іншою інструкцією.
16#0006	AS-FECAT не встановлено.
16#0007	Помилка ініціалізації веденого пристрою (AL STATUS UNKNOWN)
16#0008	Помилка ініціалізації веденого пристрою (AL STATUS ERROR)
16#0009	Кількість фактично підключених підлеглих пристроїв менше, ніж кількість підлеглих пристроїв встановити в INITEC.
16#000A	Відсутнє фізичне з'єднання між AS-FECAT і підлеглим
16#000B	Час очікування ініціалізації AS-FECAT

7. Коли проблема з втратою з'єднання (SR0 = 0x19 01 ~0x19 10, останні два коди представляють ідентифікатор підлеглого) виникає після ініціалізація завершена, SM1684 OFF (за замовчуванням, вказуючи, що коли один відключається, усі підлегли вимкнені), і відповідні дії інших підлеглих також буде призупинено. Після усунення проблеми вам потрібно ініціалізувати всі підлегли пристрої, щоб перезапустити роботу. ПЛК автоматично вимкне миготливий світлодіод помилки після підтвердження завершення ініціалізації, і помилка буде записана в журнал помилок.
8. Якщо підлегли пристрої повинні працювати незалежно, не зазнаючи впливу збою будь-якого іншого підпорядкованого пристрою, коли кабель зв'язку під'єднано належним чином, вам потрібно встановити SM1684 у положення ON після ініціалізації, щоб повідомити ПЛК про запис помилки та встановити відповідний прапор помилки на ON, без необхідності зупиняти роботу інших підлеглих пристроїв.

API	Інструкція				Операнд								опис			
2821		ECATRW			Slave , RWCode , ... ErrCode								Читання та запис даних EtherCAT SDO			
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F
Slave								●					○	○		
RWCode								●					○	○		
Index								●								
SubIndex								●								
Length								●					○	○		
Data								●								
Done		●	●	●												
Error		●	●	●												
ErrCode								●								

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
RWCode		●				●							
Index		●				●							
SubIndex		●				●							
Length		●				●							
Data		●	●			●	●						
Done	●												
Error	●												
ErrCode		●											

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ

ECATRW	
En	
Slave	Done
RWCode	ErrCode
Index	ErrCode
SubIndex	
Length	
Data	

Slave : ID раба

RWCode : Код функції

Index : Індекс

SubIndex : Підіндекс

Length : Розмір даних

Data : Регістр призначення для даних, які потрібно зчитати або
Регістр джерела для запису даних

Done : Прапор завершення

Error : Прапор помилки

ErrCode : Код помилки

Пояснення

1. Ця інструкція доступна для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
2. Ця інструкція використовується для читання та запису значень параметрів підпорядкованого пристрою в мережі EtherCAT через зв'язок SDO.
3. **Slave** визначає значення в діапазоні 1~16. (AS-FECAT підтримує максимальну кількість 16 підлеглих пристроїв EtherCAT.) Якщо значення перевищує діапазон налаштування для ініціалізації, ПЛК вважатиме це помилкою налаштування.
4. **RWCode** вказує код функції для читання або запису даних. Перегляньте коди функцій читання/запису в наведеній нижче таблиці.

Код функції	16#0000	16#0001	16#1000	16#1001
опис	Прочитайте один сингл Значення параметра EtherCAT	Прочитайте кілька Значення параметрів EtherCAT	Напишіть один сингл Значення параметра EtherCAT	Напишіть кілька Значення параметрів EtherCAT

Примітка: якщо ви встановите **RWCode** на код функції, який не визначений у таблиці вище, ПЛК розглядатиме це як помилку.

Коли використовуються коди функцій для читання та запису одного значення параметра, параметр **Length** інструкції недійсний. Коли використовуються функціональні коди для читання та запису кількох значень параметрів, інструкція виконується відповідно до налаштування параметра **Length**.

5. **Index** і **SubIndex** — це індекс і підіндекс для зв'язку SDO. **Length** — це розмір кількох значень параметрів, які потрібно прочитати або записати, а діапазон налаштувань становить 1 ~ 50. Якщо значення налаштувань перевищує діапазон, воно автоматично регулюється до верхньої або нижньої межі. Якщо ви збираєтеся читати та записувати кілька значень параметрів, змінні **Index** і **SubIndex** мають бути оголошені як тип Array.
6. **Data** — це пристрій призначення для зберігання даних для читання або вихідний пристрій, на якому знаходяться дані для запису. Читати і записати кілька значень параметрів, слід зазначити, що тип даних **Data** має бути оголошений як Array. Кожне прочитане або записане значення параметра займає 2 слова.
7. **Done** — це позначка завершення читання або запису. Під час виконання операції читання вам слід дочекатися, поки прапор завершення ввімкнеться, перш ніж використовувати дані читання.
8. **Error** — це позначка помилки зв'язку, а **ErrCode** — код помилки.

Перегляньте коди помилок і причини в наступній таблиці.

Код помилки	причина
16#0000	Помилки немає
16#0001	Ідентифікатор підпорядкованого пристрою виходить за межі діапазону або вказаний підлеглий пристрій не ініціалізовано.
16#0002	Немає відповіді від раба
16#0003	Неправильний код функції читання та запису.
16#0004	Відповідно до встановленої довжини даних для читання та запису, дані в Index / SubIndex поза діапазоном.

Код помилки	причина
16#0005	Відповідно до встановленої довжини даних для читання та запису дані в Data пристрій знаходиться поза діапазоном.
16#0006	Функціональна карта AS-FECAT не встановлена
16#0007	Сталася помилка в повідомленні, отриманому підлеглим. Ця помилка часто виникає, коли параметри інструкції ECATRW недійсні, тому підлеглий пристрій не може отримати повні повідомлення.

API	Інструкція				Операнд								опис				
2822		MOTST			Slave , Reset , ... ErrCode								Читання стану руху підлеглого EtherCAT				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Slave								●					O	O			
Reset			●														
Position								●									
State								●									
Done		●	●	●													
Error		●	●	●													
ErrCode								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
Reset	●												
Position			●				●						
State		●				●							
Done	●												
Error	●												
ErrCode		●											

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- Slave : ID раба
- Reset : Скиньте регулятор положення
- Position : Позиція підлеглого (32 біта)
- State : Рабський стан
- Done : Прапор завершення
- Error : Прапор помилки
- ErrCode : Код помилки

Пояснення

- Ця інструкція доступна для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- За допомогою методу зв'язку PDO ця інструкція регулярно оновлює статус і положення вказаного підлеглого пристрою. Зазначений підлеглий пристрій сканується, а його статус і позиція автоматично оновлюються відповідно до змінних інструкції. Рекомендується виконати інструкцію для оновлення останнього стану підлеглого пристрою після завершення ініціалізації.

3. **Slave** визначає значення в діапазоні 1~16. (AS-FECAT підтримує до 16 підлеглих пристроїв EtherCAT.) Якщо значення перевищує діапазон налаштування для ініціалізації, ПЛК вважатиме це помилкою налаштування.
4. Коли **Reset** увімкнено, контроль зайнятої позиції над вказаним підлеглим скидається, щоб інші інструкції позиції (включно з DDRVIC / DDRVAC / DPLSVC / ZRNC / DZRNC / DCSFOC / DTQC / DTQLC) могли перезапустити підлеглий пристрій для виконання керування положенням. Після відпускання регулятора положення **Reset** автоматично зміниться на OFF.

Примітка. У звичайному управлінні положенням вихідне позиціонування буде зупинено, коли буде виконано цю дію «**Reset**».

5. Якщо тип пристрою зазначеного підлеглому пристрою — серво, **Position** — це поточна позиція (32 біти) зазначеного підлеглому пристрою. Якщо тип пристрою зазначеного підпорядкованого пристрою — це електропривод змінного струму, значення **Position** є фіксованим значенням 0.
6. **Стан** - це поточний статус зазначеного підлеглому.

Дивіться визначення бітів, коли всі біти значення стану ON, у наведеній нижче таблиці.

Номер біта	Біт 15 ~ Біт 4	Біт 3	Біт 2	Біт 1	Біт 0
Визначення	Зарезервований	Серво увімкнено	Пауза	Позиція досягнута	Позиція на виході

7. **Done** — це позначка завершення оновлення статусу. У звичайних обставинах прапор має постійно горіти. Якщо він вимкнений, будуть збережені останні оновлені дані про поточний стан, положення та швидкість підлеглому пристрою.
 8. **Error** — це позначка помилки зв'язку, а **ErrCode** — код помилки.
- Перегляньте коди помилок і причини в наступній таблиці.

Код помилки	причина
16#0000	Помилки немає
16#0001	Ідентифікатор підпорядкованого пристрою виходить за межі діапазону або вказаний підлеглий пристрій не ініціалізовано.
16#0003	Функціональна карта AS-FECAT не встановлена.

API		Інструкція			Операнд							опис					
2823			ECATST			Slave , State , Done , Error , ErrCode							Читання стану підлеглого EtherCAT				
пристрій	X	Y	M	S	T	C	HC	D	FR	SM	SR	E	K	16#	«\$»	F	
Slave								●					○	○			
State								●									
Done		●	●	●				●									
Error		●	●	●				●									
ErrCode								●									

Тип даних	BOOL	WORD	DWORD	LWORD	UINT	IHT	DINT	LINT	REAL	LREAL	TMR	CNT	STRING
Slave		●				●							
State		●				●							
Done	●												
Error	●												
ErrCode		●											

Інструкція по пульсу	16-бітна інструкція	32-розрядна інструкція
-	AS	-

СИМВОЛ



- Slave** : ID раба
- State** : Статус веденого пристрою EtherCAT
- Done** : Прапор завершення
- Error** : Прапор помилки
- ErrCode** : Код помилки

Пояснення

- Ця інструкція доступна для ПЛК із мікропрограмою V1.14.00 або пізнішої версії та ISPSOft V3.16 або пізнішої версії, а функціональна карта AS-FECAT повинна бути встановлена на ПЛК.
- Виконання інструкції INITEC має бути завершено до виконання цієї інструкції.
- Кількість разів використання інструкції не обмежена. Однак для одного підлеглого пристрою дозволяється одночасно виконувати лише одну інструкцію ECATST.
- Slave** визначає значення в діапазоні 1~16. (AS-FECAT підтримує максимальну кількість 16 підлеглих пристроїв EtherCAT.) Якщо значення перевищує діапазон налаштувань або вказаний підлеглий пристрій не ініціалізовано, інструкція не буде виконана.

5. Стан : стан зв'язку підлеглого пристрою **EtherCAT**

Номер біта	опис
Біт 0~3	1 : стан ініціалізації 2 : Передопераційний стан 3 : Завантажувальний стан 4 : безпечний робочий стан 8 : робочий стан
Біт 4	0 : стан пристрою відповідає необхідному стану або очищено командою. 1 : Стан пристрою не відповідає необхідному стану або змінено підлеглим.
Біт 5	0 : Ідентифікація пристрою не ефективна. 1 : Ідентифікація пристрою ефективна.
Біт 6~15	Зарезервований

6. Done — це позначка завершення оновлення статусу. За звичайних обставин прапор має бути ON весь час. Якщо він вимкнений,

будуть збережені останні оновлені дані підлеглого стану EtherCAT, згадані вище.

7. Error — це позначка помилки зв'язку, а **ErrCode** — код помилки.

Перегляньте коди помилок і причини в наступній таблиці.

Код помилки	причина
16#0000	Помилки немає
16#0001	Ідентифікатор підпорядкованого пристрою виходить за межі діапазону або вказаний підлеглий пристрій не ініціалізовано.
16#0002	Немає відповіді від раба
16#0003	Функціональна карта AS-FECAT не встановлена.
16#0004	Дві або більше інструкцій ECATST запускаються з однаковим ID підлеглого.

6.27.3 Часті запитання в Delta Special CANopen Communication and Troubleshooting

Запитання 1: після того, як ПЛК запускає інструкцію INITC, прапор завершення ініціалізації SM1681 не встановлюється на ON.

Кроки:

- 1) Переконайтеся, що для карти 2 встановлено плату зв'язку AS-FCOPM в AS HWCONFIG і зв'язок режим — це режим спеціального драйвера Delta. Потім перевірте, чи швидкість зв'язку відповідає вказаній швидкості передачі даних. Після того, як наведені вище параметри будуть правильні, повторно завантажте таблицю конфігурації апаратного забезпечення, а потім ще раз перевірте, чи працює ініціалізація. Якщо ініціалізація все ще не працює, перейдіть до наступного кроку.
- 2) Переконайтеся, що ініціалізацію сервоприводу встановлено вручну. Потім знову увімкніть сервопривід після вимкнення. Після цього перейдіть до кроку 3.
- 3) Переконайтеся, що ідентифікатор сервоузла (P3-00), швидкість передачі даних і режим (P3-01) встановлено вручну, а режим зв'язку (P1-01) встановлено в режим PR вручну. Потім перейдіть до кроку 4.
- 4) Переконайтеся, що підключення кабелю зв'язку CAN є правильним, кінцевий резистор на платі AS-FCOPM ПЛК увімкнено, а останній сервопривод під'єднано до кінцевого резистора. Потім перейдіть до кроку 5.
- 5) Зверніть увагу, чи слово: **код** тимчасово з'являється на цифровому дисплеї сервоприводу, коли ПЛК вмикає інструкцію ініціалізації. Якщо код тимчасово не відображається, поверніться до кроку 1 або замініть сервопривід. Якщо тимчасово з'являється **код слова**, перейдіть до кроку 6.
- 6) Перевірте, чи фактичні стани DI0~DI7 (P2-10 ~ P2-17) сервоприводу відповідають функціональним налаштуванням параметрів сервоприводу. Наприклад, DI0 – вхід негативного обмеження (контакт А). Перевірте, чи фактичний стан входу OFF (контакт В). Якщо функцію DI було ввімкнено, вимкніть її. Якщо функцію не ввімкнено, перейдіть до кроку 7.
- 7) Режим зв'язку комунікаційної карти AS-FCOMP у HWCONFIG знову перемикається на режим DS301. Використовуйте програмне забезпечення CANopen Builder для пошуку підпорядкованого пристрою та спостерігайте, який підпорядкований пристрій не було проскановано. Видаліть той, який не був просканований, а потім знову виконайте ініціалізацію, налаштувавши ідентифікатор сервоузла та ідентифікатор в INITC у ПЛК.

Розділ 7 Усунення несправностей

Зміст

7.1 Усунення несправностей	7-3
7.1.1 Основні дії з усунення несправностей	7-3
7.1.2 Видалення станів помилок	7-3
7.1.3 Усунення несправностей SOP	7-4
7.1.4 Системний журнал і позиції кроку помилки	7-5
7.2 Усунення несправностей для модулів процесора	7-7
7.2.1 Світлодіодні індикатори ERROR	7-7
7.2.2 Світлодіодні індикатори ERROR блимають кожні 0,5 секунди	7-7
7.2.3 Світлодіодні індикатори ERROR швидко блимають кожні 0,2 секунди	7-9
7.2.4 Світлодіодні індикатори ERROR Повільно блимають кожні 3 секунди та світяться протягом 1 секунда	7-9
7.2.5 BAT. Світлодіодні індикатори LOW	7-10
7.2.6 BAT. LOW Світлодіодні індикатори блимають кожні 0,5 секунди.....	7-10
7.2.7 Світлодіодні індикатори RUN і ERROR блимають одночасно 0,5 секунди	7-11
7.2.8 Індикатори RUN і LED блимають один за одним 0,5 секунди.	7-11
7.2.9 Інші помилки (без світлодіодних індикаторів).....	7-11
7.3 Усунення несправностей для інших модулів вводу/виводу.....	7-19
7.3.1 Усунення несправностей для аналогових модулів (AD/DA/XA) і температури Модулі (RTD/TC)	7-19
7.3.2 Усунення несправностей для модуля позиціонування AS02/04PU	7-20
7.3.3 Пошук і усунення несправностей модуля високошвидкісного лічильника AS02HC	7-20
7.3.4 Пошук і усунення несправностей модуля тензодатчика AS02LC.....	7-21
7.3.5 Усунення несправностей для модуля AS04SIL IO-Link як зв'язку Модуль	7-22
7.3.6 Усунення несправностей для модуля AS00SCM як модуля зв'язку	7-25
7.3.7 Пошук і усунення несправностей для модуля AS00SCM як віддаленого модуля.....	7-26
7.3.8 Пошук та усунення несправностей для модулів AS01DNET	7- 27
7.4 Коди помилок і світлодіодні індикатори для модулів.....	7-28
7.4 .1 Коди помилок і світлодіодні індикатори для модулів центрального процесора	7-28
7.4 .2 Коди помилок і світлодіодні індикатори для аналогових/температурних модулів	7-35
7.4 .3 Коди помилок та світлодіодні індикатори для модуля позиціонування AS02 / 04PU.....	7-36
7.4 .4 Коди помилок і світлодіодні індикатори для модуля високошвидкісного лічильника AS02HC.....	7-36
7.4 .5 Коди помилок та світлодіодні індикатори для модуля тензодатчика AS02LC	7-37
7.4 .6 Коди помилок і світлодіодні індикатори для модуля AS04SIL IO-Link as комунікаційний модуль	7-37
7.4 .7 Коди помилок і світлодіодні індикатори для модуля AS00SCM як зв'язку Модуль	7-38
7.4 .8 Коди помилок і світлодіодні індикатори для модуля AS00SCM як пульта дистанційного керування Модуль.....	7-38
7.4 .9 Коди помилок і світлодіодні індикатори для модуля AS01DNET (Режим Master/Slave)	7-39
7.4 .10 Коди помилок і світлодіодні індикатори для модуля AS01DNET (режим RTU)	7-40

7.1 Вирішення проблем

7.1.1 Основні дії з усунення несправностей

У цьому розділі описано можливі помилки, які можуть виникнути під час роботи, їх причини та дії щодо усунення.

(1) Перевірте наступне:

- ПЛК має працювати в безпечному середовищі (враховуйте безпеку навколишнього середовища, електроніку та вібрацію).
- Правильно підключіть джерело живлення до ПЛК.
- Закріпіть модуль, термінал і кабельні установки.
- Всі світлодіодні індикатори світяться правильно.
- Встановіть усі перемикачі правильно.

(2) Перевірте такі робочі функції:

- Перемкніть стан RUN/STOP
- Перевірте параметри RUN/STOP для серії AS
- Перевірити та усунути помилки зовнішніх пристроїв
- Використовуйте функцію системного журналу в ISPSoft, щоб перевірити роботу системи та журнали

(3) Визначте можливі причини:

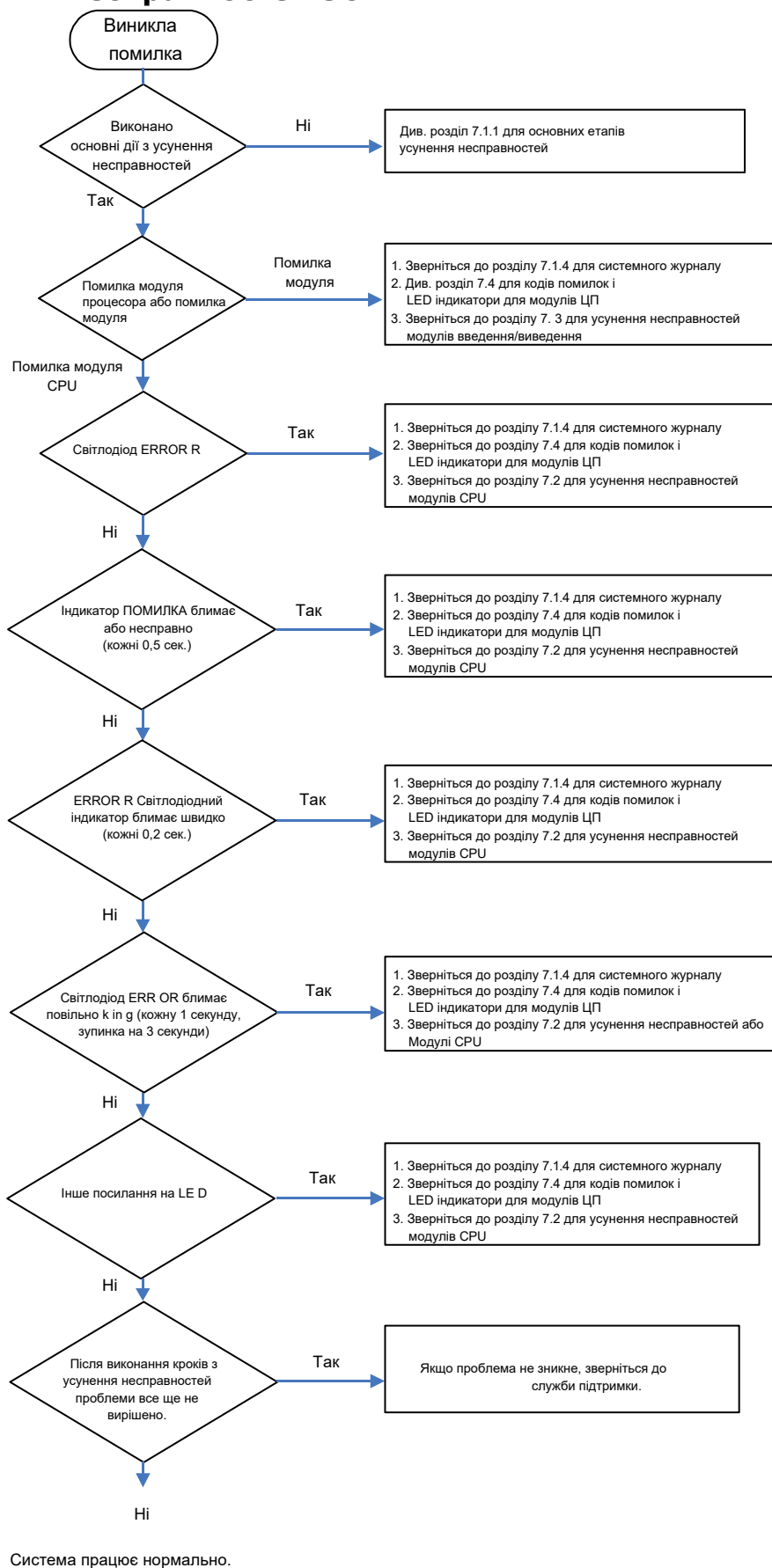
- Серія AS або зовнішній пристрій
- ЦП або модулі розширення
- Параметри або налаштування програми

7.1.2 Очистіть стани помилок

Використовуйте наведені нижче методи, щоб очистити стани помилок. Якщо джерело помилки не виправлено, система продовжує показувати помилки.

- (1) Перемкніть стан моделі CPU на STOP, а потім на RUN.
- (2) Вимкніть процесор і ввімкніть його знову.
- (3) Використовуйте ISPSoft, щоб очистити журнали помилок.
- (4) Скиньте налаштування процесора до стандартних і завантажте програму знову.

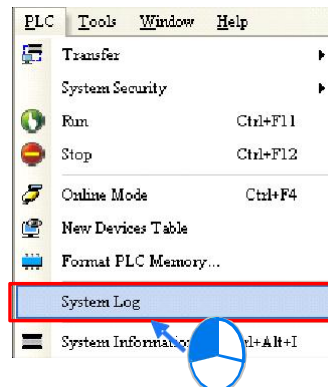
7.1.3 Усунення несправностей SOP



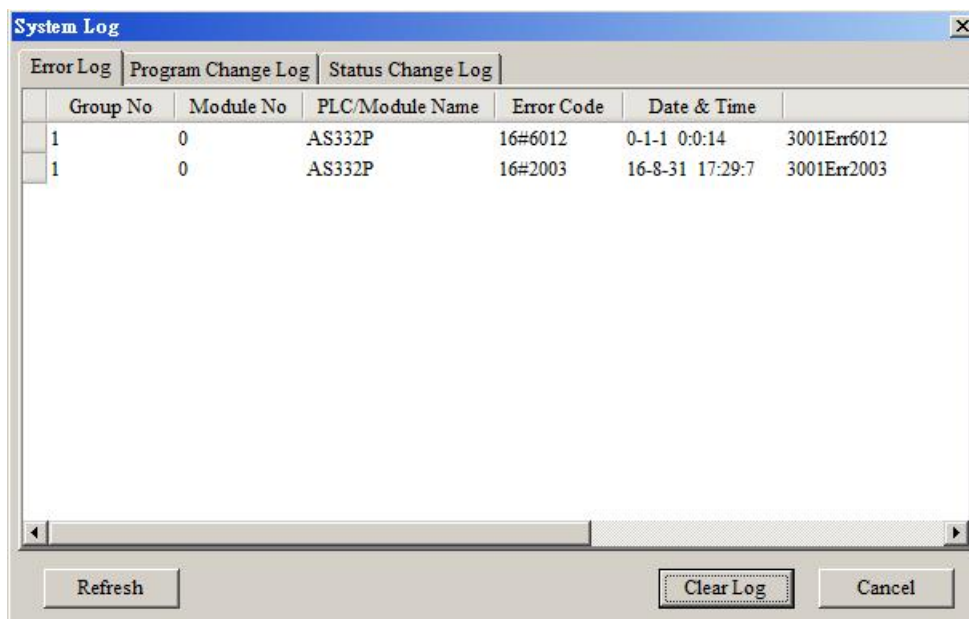
7.1.4 Системний журнал і позиції кроку помилки

Якщо ISPSoft підключено до серії AS, ви можете переглянути дії та помилки в серії AS, натиснувши « **Системний журнал** » у меню **ПЛК** . ЦП може зберігати до 20 наборів журналів помилок. Після того, як 20 наборів буде збережено, 1 -й журнал замінюється на 21-й, якщо надходять нові журнали, а старі журнали замінюються на нові послідовно. Коли картку пам'яті встановлено в модуль ЦП, на карті пам'яті створюється резервна копія 20 наборів старих журналів, і можна записати до 1000 журналів. Якщо збережені журнали перевищують 1000, найстаріші 20 журналів замінюються на 20 найновіших журналів на карті пам'яті.

(1) У меню **ПЛК** натисніть **Системний журнал** .



(2) З'явиться вікно **системного журналу** . Натисніть **Очистити журнал** , щоб очистити журнал помилок у вікні та журнал помилок у модулі ЦП, а також скинути модуль ЦП.

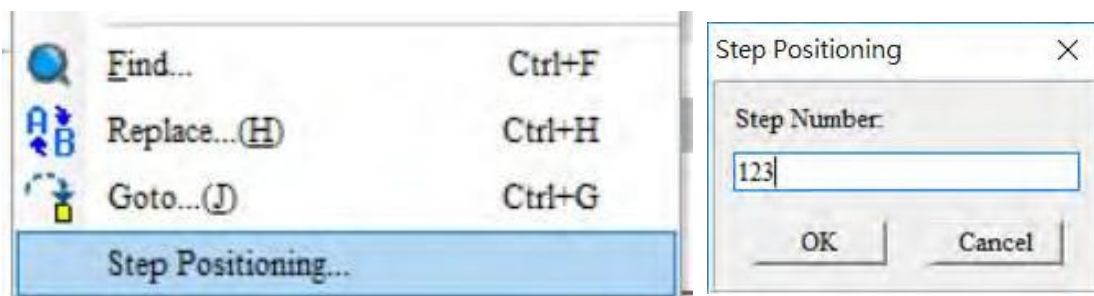


- Номер групи: цифра 1 вказує на те, що помилка сталася в модулі ЦП або правому модулі 1. Цифри 2–16 вказують на помилку у віддалених модулях 1–15.
- Номер модуля: цифра 0 вказує на те, що помилка сталася в модулі ЦП або віддаленому модулі. Цифри 1–32 вказують на те, що помилка сталася в правому модулі ЦП / віддаленого модуля. Цифра 1 позначає модуль, найближчий до модуля ЦП або віддаленого модуля. Це число збільшується від найближчого до найдалшого від модуля ЦП або віддаленого модуля. Примітка: до правої сторони віддаленого модуля можна підключити до восьми модулів розширення.
- Назва ПЛК/модуля: назви моделей ЦП, дистанційного керування та модулів розширення.

- Код помилки: коди помилок у журналі помилок.
- Дата й час: дата й час, коли сталася помилка. Помилка, яка сталася останньою, вказана вгорі.
- Останній стовпець містить описи помилок.

(3) Якщо операція або операнд виходить за межі дозволеного діапазону (SM0 = ON, зберігається в SR0) або виникає помилка перевірки інструкції (SM5 = ON, зберігається в SR4), ви можете використовувати наступний метод, щоб побачити, де проблема.

- Коли SM0 = ON, перевірте значення в SR1, щоб побачити крок помилки. SR1: адреса помилки операції (32-біт).
- Коли SM5 = ON, перевірте значення в SR5, щоб побачити крок помилки. SR5: адреса помилки перевірки граматики (32-біт).
- Коли ви дізнаєтесь номер кроку помилки, ви можете перейти до ISPSoft -> Edit -> Step positioning. Введіть номер кроку помилки та натисніть OK. Система переведе вас у позицію, де крок пішов не так.



7.2 Усунення несправностей для модулів ЦП

Перевірте світлодіодні індикатори та коди помилок модуля ЦП і зверніться до наступної таблиці для усунення несправностей. V у стовпці Журнал означає, що помилка зареєстрована в журналі. X у стовпці Журнал означає, що помилка не зареєстрована в журналі. Н у стовпці Журнал вказує, чи можна налаштувати запис помилки в журнал у HWCONFIG.

Примітка. Журнал помилок також зберігається в SR. Зверніться до глави 2 для отримання додаткової інформації про SR.

7.2.1 Світлодіодні індикатори ERROR

Код помилки (16#)	опис	Рішення	Прапор	Журнал
000A	Тайм-аут сканування	1. Перевірте налаштування сторожового таймера в HWCONFIG. 2. Перевірте, чи програма довго сканує	SM8	V

7.2.2 Світлодіодні індикатори ERROR блимають кожні 0,5 секунди

Код помилки (16#)	опис	Рішення	Прапор	Журнал
000C	Програма в ПЛК пошкоджена.	Завантажте програму ще раз.	SM9	V
000D	Початкове значення пристрою та програма ПЛК завантажено не повністю.	Завантажте програму ще раз.	SM9	V
0010	Пам'ять ЦП заборонено.	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM9	V
0020	Час сканування ПЛК довший, ніж очікувалося, через ненормальну швидкість роботи ЦП.	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM9	V
0021	Не вдалося відновити параметри пристрою Delta до значень за замовчуванням	1. Перевірте, чи файл EDS пристрою збігається з пристроєм, для якого потрібно відновити стандартні налаштування. 2. Перевірте, чи нормальне мережеве з'єднання для ЦП ПЛК і пристрою, який потрібно відновити до стандартних значень. 3. Перевірте сумісність пристрою, який потрібно відновити до значень за замовчуванням.	SM9	V
002E	Доступ до зовнішньої пам'яті ЦП заборонено.	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM9	V
002F	Програми ПЛК не відповідають системним журналам.	Завантажте програму ще раз.	SM34	V
0070	Розташування функціональних карток не відповідає налаштуванням.	Перевірте, чи параметри в HWCONFIG узгоджуються з розташуванням функціональних карток.	SM10	V
0102	Номер переривання перевищує діапазон.	Перевірте позицію, де сталася помилка, і змініть програму. Скопіюйте програму та завантажте її знову.	SM5	X
0202	Інструкція MC перевищує діапазон.		SM5	X
0302	Інструкція MCR перевищує діапазон.		SM5	X

Код помилки (16#)	опис	Рішення	Прапор	Журнал
0D03	Операнди, які використовуються в DHSCS, не використовуються належним чином.		SM5	X
0E05	Операнди HCXXX, які використовуються в DCNT, не використовуються належним чином.		SM5	X
1000 - 10FF	Системна помилка	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	-	V
1300 - 130F	У віддалених модулях виникли помилки	Додаткову інформацію про коди помилок для віддалених модулів див. у розділі 7.3.4.	SM30	V
1402	Розташування модулів введення/виведення не відповідає налаштуванням.	Перевірте, чи параметри в HWCONFIG узгоджуються з розташуванням модулів введення/виведення.	SM10	V
140B	Більше чотирьох комунікаційних модулів.	Перевірте загальну кількість комунікаційних модулів (максимум чотири).	SM10	V
140C	Кількість підключених модулів позиціонування перевищує максимальну кількість 8.	Перевірте загальну кількість модулів позиціонування.	SM10	V
140D	Більше 32 модулів розширення.	Перевірте загальну кількість модулів розширення (максимум 32).	SM10	V
140E	Більше восьми віддалених модулів з правого боку від модуля ЦП.	Перевірте загальну кількість віддалених модулів праворуч від модуля ЦП (максимум 8).	SM30	V
1600	Ідентифікатор модуля розширення перевищує діапазон.	1.Переконайтеся, що модуль належним чином підключено до модуля ЦП, і знову ввімкніть модулі. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM10	V
1601	Ідентифікатор модуля розширення не може бути встановлений.	1.Переконайтеся, що модуль належним чином підключено до модуля ЦП, і знову ввімкніть модулі. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM10	V
1602	Ідентифікатор модуля розширення дублюється.	1.Переконайтеся, що модуль належним чином підключено до модуля ЦП, і знову ввімкніть модулі. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM10	V
1603	Модуль розширення не працює.	1.Переконайтеся, що модуль належним чином підключено до модуля ЦП, і знову ввімкніть модулі. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM10	V
1604	Час очікування зв'язку модуля розширення	1.Переконайтеся, що модуль належним чином підключено до модуля ЦП, і знову ввімкніть модулі. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM10	V
1605	Збій обладнання	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM10	V
1606	Помилки налаштувань функціональної карти модуля зв'язку	Переконайтеся, що функціональну плату правильно підключено до модуля ЦП, і знову ввімкніть модулі.	SM10	V

Код помилки (16#)	опис	Рішення	Прапор	Журнал
1607	Зовнішня напруга ненормальна.	Перевірте, чи нормальне зовнішнє джерело живлення 24 V до модуля.	SM10	V
1608	Внутрішнє заводське калібрування або CJC не відповідає нормі.	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM10	V
1609 - 160F	Зарезервовано (коди помилок для модулів розширення)			
200A	Недійсна інструкція	Перевірте позицію, де сталася помилка, і змініть програму. Скопіюйте програму та завантажте її знову.	SM5	V
6010	Кількість підключень MODBUS TCP перевищує діапазон.	Перевірте кількість вищих пристроїв (максимум 32).	SM 1092	V
6011	Кількість підключень EtherNet/IP перевищує допустимий діапазон.	Перевірте кількість підключень (максимум 16).	SM 1093	V
C000 - CFFF	Неправильний синтаксис програми ПЛК.	Збережіть програму PLC і зверніться до свого дистриб'ютора.	-	V

7.2.3 Світлодіодні індикатори ERROR швидко блимають кожні 0,2 секунди

Це трапляється, коли джерело живлення 24 В постійного струму модуля ЦП від'єднано, або джерело живлення є недостатнім, нестабільним або ненормальним.

Код помилки (16#)	опис	Рішення	Прапор	Журнал
002A	Зовнішня напруга ненормальна.	Перевірте, чи нормальне зовнішнє джерело живлення 24 В до модуля.	SM7	V

7.2.4 Світлодіодні індикатори ERROR Повільно блимають кожні 3 секунди та світяться на 1 секунду

Код помилки (16#)	опис	Рішення	Прапор	Журнал
0040	Перетворене значення у функціональній картці 1 (AS-F2AD) або у вбудованому вхідному каналі AD 1 перевищує діапазон налаштування.	Перевірте фактичний вхідний сигнал	SM27	X
0041	Перетворене значення у функціональній картці 1 (AS-F2AD) або у вбудованому вхідному каналі AD 2 перевищує діапазон налаштування.	Перевірте фактичний вхідний сигнал	SM27	X
0042	Перетворене значення у вхідному каналі 1 функціональної карти 2 (AS-F2AD) перевищує діапазон налаштування.	Перевірте фактичний вхідний сигнал	SM27	X
0043	Перетворене значення у вхідному каналі 2 функціональної карти 2 (AS-F2AD) перевищує налаштування	Перевірте фактичний вхідний сигнал	SM27	X

	діапазон.			
0044	З'єднання втрачено в режимі струмового входу 4 мА - 20 мА для функціональної карти 1 (AS-F2AD) або для вбудованого вхідного каналу AD 1.	Перевірте кабель підключення.	SM27	X
0045	З'єднання втрачено в режимі струмового входу 4 мА - 20 мА для функціональної карти 1 (AS-F2AD) або для вбудованого вхідного каналу AD 2.	Перевірте кабель підключення.	SM27	X
0046	З'єднання втрачено в режимі струмового входу 4 мА - 20 мА для вхідного каналу 1 функціональної карти 2 (AS-F2AD).	Перевірте кабель підключення.	SM27	X
0047	З'єднання втрачено в режимі струмового входу 4 мА - 20 мА для вхідного каналу 2 функціональної карти 2 (AS-F2AD).	Перевірте кабель підключення.	SM27	X
1500	З'єднання втрачено у віддалених модулях	Перевірте кабель підключення до мережі.	SM30	V
1502 - 150F	У віддалених модулях виникли помилки	Додаткову інформацію про коди помилок віддаленого модуля див. у розділі 7.3.4.	SM30	V
1800 - 180F	Виникли помилки в модулях розширення	Зверніться до розділу 7.3 для отримання додаткової інформації про коди помилок модуля розширення.	SM10	V
1900 - 191C	У веденому модулі зв'язку CANopen/ECAT виникли помилки серцевого ритму/втрати з'єднання.	1. Перевірте з'єднувальний кабель CANopen / ECAT. 2. Перевірте, чи певний підлеглий пристрій працює належним чином. Примітка. Останні 2 цифри коду помилки представляють ідентифікаційний номер підпорядкованого пристрою (конвертуйте шістнадцяткове число в десяткове).	-	V

7.2.5 BAT. Світлодіодні індикатори LOW горять

Це трапляється, коли немає батареї (CR1620) або низький заряд. Вимкніть цю опцію в HWCONFIG > CPU >

Device Setting > Show Battery Low Voltage Error CPU, коли вам не потрібна функція RTC для відстеження

7 поточний час (за замовчуванням увімкнено).

Код помилки (16#)	опис	Рішення	Прапор	Журнал
0027	Низький заряд батареї	Замініть акумулятор або вимкніть цю опцію	SM219	X

7.2.6 BAT. Світлодіодні індикатори LOW блимають кожні 0,5 секунди

Це трапляється, коли RTC не може відстежувати поточний час.

Код помилки (16#)	опис	Рішення	Прапор	Журнал
0026	RTC не може відстежувати поточний час	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM218	V
0028	Немає відповіді від RTC	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM217	V

7. Світлодіодні індикатори RUN і ERROR блимають одночасно кожні 0,5 секунди

Це трапляється під час оновлення мікропрограми модуля ЦП. Якщо це відбувається після подачі живлення на модуль ЦП, це означає, що під час попереднього оновлення мікропрограми виникли помилки. Користувачам потрібно знову оновити мікропрограму або звернутися до місця придбання.

7. Індикатори RUN і LED блимають один за одним кожні 0,5 секунди.

Це трапляється, коли карта пам'яті модуля ЦП виконує резервне копіювання, відновлення або збереження. Якщо ви використовуєте ЦП серії AS218 без встановленої SD-карти, індикатори RUN і світлодіодні індикатори блимають один за одним кожні 0,5 секунди, що означає оновлення мікропрограми АІО. Якщо це станеться після подачі живлення на модуль ЦП, це означає, що оновлення мікропрограми АІО не вдалося, і його потрібно оновити знову. Якщо ви не оновлювали мікропрограму АІО, блимання означає, що компонент АІО несправний і вам потрібно зв'язатися з пунктом покупки.

7. Інші помилки (без світлодіодних індикаторів)

Код помилки (16#)	опис	Рішення	Прапор	Журнал
0011	Ідентифікатор ПЛК неправильний.	Перевірте ідентифікатор ПЛК.	SM34	V
0012	Пароль ПЛК неправильний.	Перевірте пароль ПЛК.	SM34	V
002D	Перевищено максимальну кількість спроб пароля ПЛК.	Скиньте модуль ЦП або відновіть заводські налаштування модуля ЦП.	SM34	V
0050	Пам'ять у зафіксованих спеціальних допоміжних реле є ненормальною.	1.Скиньте модуль ЦП або відновіть заводські налаштування модуля ЦП, а потім знову завантажте програму та параметри. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM6	V
0051	Зафіксовані спеціальні регістри даних є ненормальними.	1.Скиньте модуль ЦП або відновіть заводські налаштування модуля ЦП, а потім знову завантажте програму та параметри. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM6	V
0052	Пам'ять у зафіксованих допоміжних реле ненормальна.	1.Скиньте модуль ЦП або відновіть заводські налаштування модуля ЦП, а потім знову завантажте програму та параметри. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM6	V
0054	Заблоковані лічильники ненормальні.	1.Скиньте модуль ЦП або відновіть заводські налаштування модуля ЦП, а потім знову завантажте програму та параметри. 2.Якщо помилка все ще виникає, зверніться до свого дистриб'ютора.	SM6	V
0055	Зафіксовані 32-розрядні лічильники є ненормальними.	1.Скиньте модуль ЦП або відновіть заводські налаштування модуля ЦП, а потім знову завантажте програму та параметри. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM6	V

0056	Заблоковане спеціальне допоміжне реле є ненормальним.	1.Скиньте модуль ЦП або відновіть заводські налаштування модуля ЦП, а потім знову завантажте програму та параметри. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM6	V
0059	Зафіксовані регістри даних є ненормальними.	1.Скиньте модуль ЦП або відновіть заводські налаштування модуля ЦП, а потім знову завантажте програму та параметри. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	SM6	V
005D	Модуль процесора не виявляє карту пам'яті.	Перевірте, чи правильно вставлено картку пам'яті в модуль ЦП.	SM453	V
005E	Картка пам'яті ініціалізована неправильно.	Перевірте, чи не зламана карта пам'яті.	SM453	V
0063	Під час запису даних на карту пам'яті виникає помилка.	Перевірте, чи правильний шлях до файлу, чи несправна карта пам'яті.	SM453	V
0064	Неможливо прочитати файл на карті пам'яті.	Перевірте, чи правильний шлях до файлу, чи файл пошкоджено.	SM453	V
1950	Ініціалізація комунікаційного модуля CANopen ще не завершена, інструкції CANopen не можуть бути виконані.	1. Перевірте з'єднувальний кабель CANopen. 2. Перевірте, чи певний підлеглий пристрій працює належним чином. 3.Якщо нічого не так, знову ініціалізуйте комунікаційний модуль CANopen.	-	V
19B0	У підпорядкованому модулі зв'язку CANopen стався тайм-аут серцевого ритму.	Перевірте, чи добре під'єднано з'єднувальний кабель CANopen.	-	V
19B1	Довжина даних PDO (об'єкт даних процесу) у підпорядкованому режимі не відповідає налаштуванню.	Перегляньте налаштування довжини даних PDO в підпорядкованому режимі та завантажте налаштування знову.	-	V
Перевірте значення в SR830~SR893 (додаткову інформацію див. у розділі 2.2.14 цього посібника), щоб побачити, який підпорядкований пристрій (1~64) має помилку, і зверніться до наступних кодів помилок 19E1~19E8, щоб перевірити деталі.				
19E1	Довжина даних PDO (об'єкт даних процесу) у підлеглому режимі не узгоджується з параметрами списку сканування. Зверніться до розділу 10.5 Описи зв'язку CANopen у Посібнику з апаратного забезпечення та експлуатації серії AS, щоб дізнатися більше про коди помилок 19E1~19E8.	Перегляньте налаштування довжини даних PDO в підпорядкованому режимі та завантажте налаштування знову.	-	V
19E2	PDO в веденому режимі не отримано.	Перевірте, чи правильно встановлено конфігурації.	-	V
19E3	Функція автоматичного завантаження SDO не працює під час першого запуску.	Перевірте, чи правильний вміст SDO для автоматичного завантаження.	-	V
19E4	Конфігурації PDO встановлено неправильно.	Переконайтеся, що конфігурації PDO встановлено правильно.	-	V
19E5	Основні налаштування не узгоджуються з підключеним веденим.	Переконайтеся, що підключені підлеглі пристрої налаштовані в ISPSofT.	-	V
19E6	Slave HE існує в мережі.	Переконайтеся, що джерело живлення підлеглого пристрою є нормальним і підпорядкований пристрій правильно підключено до мережі.	-	V
19E7	Час очікування підлеглого керування помилками	Переконайтеся, що джерело живлення підлеглого пристрою є нормальним і підпорядкований пристрій правильно підключено до мережі.	-	V
19E8	Ідентифікатори вузлів головного та підлеглого дублюються.	Знову встановіть ідентифікатор вузла головного та підлеглого та переконайтеся, що їхні ідентифікатори вузлів унікальні.	-	V
19F3	Помилка в конфігурації	1. Знову завантажте конфігурацію параметрів. 2. Якщо проблема не зникає, зверніться до місцевого	-	V

		авторизовані дистриб'ютори.		
19F4	Зв'язок CANopen знаходиться в стані BUS-OFF.	1. Перевірте, чи мережевий кабель справний, а екранований кабель заземлений. 2. Перевірте, чи початок і кінець мережевого кабелю підключено до кінцевого резистора 121 Ом. 3. Перевірте, чи всі вузлові пристрої працюють з однаковою швидкістю передачі даних у мережі.	-	V
19FB	Регістри надсилання перевищують діапазон.	Перегляньте час для синхронізації (пропонується використовувати довший час).	-	V
19FC	Приймальні реєстри перевищують діапазон.	Перегляньте час для синхронізації (пропонується використовувати довший час).	-	V
2001	Не використовується карта FCOMP або не в правильному режимі для ASDA-A2 під час використання інструкції зв'язку CANopen.	1. Використовуйте плату FCOMP у гнізді функціональної карти 2 і переконайтеся, що її режим роботи – зв'язок CANopen. 2. Перевірте позицію, де сталася помилка, і змініть програму. Скопіюйте програму та завантажте її знову.	SM0	V
2003	Пристрій, який використовується в програмі, перевищує діапазон пристрою або є незаконним.	Перевірте позицію, де сталася помилка, і змініть програму. Скопіюйте програму та завантажте її знову.	SM0	V
200B	Операнд n або інші постійні операнди K/H перевищують діапазон.		SM0	V
200C	Операнди перекриваються.		SM0	V
200D	Двійкове перетворення десяткового числа в двійковий код неправильне.		SM0	V
200E	Рядок не закінчується на 00.		SM0	V
2012	Некоректна операція ділення		SM0	V
2013	Значення перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою.		SM0	V
2014	Завдання, визначене інструкцією TKON або YKOFF, є неправильним або виходить за межі діапазону.		SM0	V
2017	Інструкція BREAK записана поза циклом FOR-NEXT.		SM0	V
2027	Такого номера таблиці планування позицій немає або формат неправильний.	1. Перевірте позицію, де сталася помилка, і змініть програму. Скопіюйте програму та завантажте її знову. 2. Перевірте налаштування в таблиці планування позицій.	SM0	V
2028	Виконується високошвидкісна інструкція виведення. Одночасно може бути виконана лише одна інструкція.	Зверніться до SR28 для запису номера осі та зміни порядку процедур керування виходом.	-	V
2030	Вихідна частота, що виконується інструкцією CSFO, перевищує максимальну вихідну частоту.	Змініть співвідношення вхідної та вихідної частот або зменште вхідну частоту.	-	V
2031	Якщо вхідний імпульс становить 0 Гц і триває більше 2 секунд, ПЛК автоматично припиняє виведення.	Змініть співвідношення вхідної та вихідної частот або зменште вхідну частоту.	-	V
6004	Фільтр IP-адрес налаштовано неправильно.	Знову встановіть параметр Ethernet для модуля ЦП у HWCONFIG.	SM1108	X
600D	Порт RJ45 не підключено.	Перевірте підключення.	SM1100	X

6012	Є пристрої, які використовують ту саму IP-адресу.	1. Перевірте, чи є пристрої, які використовують ту саму IP-адресу. 2. Перевірте, чи є в мережі більше одного сервера DHCP або BOOTP.	SM1101	V
6100	З'єднання електронної пошти зайняте.	Спробуйте підключитися до електронної пошти пізніше. Ця помилка не призводить до припинення роботи ПЛК. Вирішіть задачу за допомогою відповідного прапорця в програмі.	SM1113	X
6103	Режим активації вкладення в електронному листі встановлено неправильно.	Налаштуйте режим підключення тригера в HWCONFIG > CPU Module > Device Setting > Options > Ethernet Port Advanced > Email > Trigger Setting > Активувати режим приєднання.	SM1113	X
6104	Вкладення в електронному листі не існує.	Перевірте, чи є вкладення на карті пам'яті.	SM1113	X
6105	Вкладення в електронному листі завелике.	Перевірте розмір вкладення. Якщо розмір перевищує 2 МБ, файл не можна надіслати як вкладення.	SM1113	X
6106	Тайм-аут відповіді сервера SMTP.	Перевірте правильну адресу та налаштуйте сервер SMTP у HWCONFIG > CPU Module > Device Setting > Options > Ethernet Port Advanced > Email ще раз.	SM1113	X
6107	Тайм-аут відповіді сервера SMTP.	1. Перевірте, чи нормальний стан SMTP-сервера. 2. Спробуйте надіслати електронний лист пізніше. Ця помилка не призводить до припинення роботи ПЛК. Вирішіть задачу за допомогою відповідного прапорця в програмі.	SM1113	X
6108	Помилка перевірки SMTP	Перевірте правильний ідентифікатор/пароль і налаштуйте його в HWCONFIG > CPU Module > Device Setting > Options > Ethernet Port Advanced > Email ще раз.	SM1113	X
6200	IP-адреса віддаленого зв'язку, встановлена у функції TCP-сокета, є незаконною.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > TCP Socket.	-	X
6201	Порт локального зв'язку, встановлений у функції TCP-сокета, є недопустимим.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > TCP Socket.	-	X
6202	Порт віддаленого зв'язку, встановлений у функції TCP-сокета, є незаконним.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > TCP Socket.	-	X
6203	Пристрій, з якого надсилаються дані у функції TCP-сокета, є незаконним.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > TCP Socket.	-	X

6206	Пристрій, який отримує дані у функції TCP-сокета, є незаконним.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > TCP Socket.	-	X
6208	Дані, отримані через сокет TCP, перевищують діапазон пристрою.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > TCP Socket.	-	X
6209	IP-адреса віддаленого зв'язку, встановлена у функції сокета UDP, є недопустимою.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > UDP Socket.	-	X
620A	Порт локального зв'язку, встановлений у функції сокета UDP, є недопустимим.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > UDP Socket.	-	X
620C	Пристрій, з якого надсилаються дані у функції сокета UDP, є незаконним.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > UDP Socket.	-	X
620F	Пристрій, який отримує дані у функції сокета UDP, є незаконним.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > UDP Socket.	-	X
6210	Дані, отримані через сокет UDP, перевищують діапазон пристрою.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > UDP Socket.	-	X
6212	Немає відповіді від віддаленого пристрою після періоду очікування.	Переконайтеся, що віддалений пристрій підключено.	-	X
6213	Отримані дані перевищують ліміт.	1. Перевірте програму та відповідні регістри спеціальних даних. 2. Установіть параметр Ethernet для модуля CPU у HWCONFIG CPU Module > Device Setting > Options > Ethernet Port Advanced > UDP Socket.	-	X
6214	Віддалений пристрій відмовляється підключитися.	Переконайтеся, що віддалений пристрій працює нормально.	-	X
6215	Розетка не відкрита.	Перевірте, чи є послідовність операцій у програмі	-	X

		правильно.		
6217	Розетка відкрита.	Перевірте, чи правильна послідовність операцій у програмі.	-	X
6218	Дані надіслано через сокет.	Перевірте, чи правильна послідовність операцій у програмі.	-	X
6219	Дані були отримані через сокет.	Перевірте, чи правильна послідовність операцій у програмі.	-	X
621A	Розетка закрита.	Перевірте, чи правильна послідовність операцій у програмі.	-	X
7011	Код функції зв'язку пристрою в COM1 неправильний.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7012	Адреса зв'язку пристрою, яка використовується в COM1, неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7013	Пристрій, який використовується в COM1, перевищує діапазон пристрою.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7014	Довжина даних пристрою в COM1 перевищує обмеження.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7017	Контрольна сума пристрою для послідовного порту зв'язку COM1 неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7021	Код функції зв'язку пристрою в COM2 неправильний.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7022	Адреса зв'язку пристрою, яка використовується в COM2, неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7023	Пристрій, який використовується в COM2, перевищує діапазон пристрою.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7024	Довжина даних пристрою в COM2 перевищує обмеження.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7027	Контрольна сума пристрою для послідовного порту зв'язку COM2 неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7031	Код функції зв'язку пристрою в Ethernet неправильний.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7032	Адреса зв'язку пристрою, яка використовується в Ethernet, неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7033	Пристрій, який використовується в Ethernet, перевищує діапазон пристрою.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7034	Довжина даних пристрою в Ethernet перевищує обмеження.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X

7037	Контрольна сума пристрою для послідовного порту зв'язку Ethernet неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7041	Код функції зв'язку пристрою в USB неправильний.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7042	Адреса зв'язку пристрою, яка використовується в USB, неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7043	Пристрій, який використовується в USB, перевищує діапазон пристрою.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7044	Довжина комунікаційних даних пристрою на USB перевищує обмеження.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7047	Контрольна сума пристрою для послідовного порту зв'язку USB неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70B1	Код функції зв'язку пристрою у функціональній картці 1 неправильний.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70B2	Адреса зв'язку пристрою, яка використовується у функціональній картці 1, неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70B3	Пристрій, який використовується у функціональній картці 1, перевищує діапазон пристрою.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70B4	Довжина даних пристрою у функціональній картці 1 перевищує обмеження.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70B7	Контрольна сума пристрою для послідовного порту зв'язку функціональної карти 1 неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70C1	Код функції зв'язку пристрою у функціональній картці 2 неправильний.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70C2	Адреса зв'язку пристрою, яка використовується у функціональній картці 2, неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70C3	Пристрій, який використовується у функціональній картці 2, перевищує діапазон пристроїв.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70C4	Довжина даних пристрою у функціональній картці 2 перевищує обмеження.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
70C7	Контрольна сума пристрою для послідовного порту зв'язку функціональної карти 2 неправильна.	1. Перевірте налаштування зв'язку в головному та підпорядкованому. 2. Перевірте кабель зв'язку.	-	X
7203	Недійсний код функції зв'язку	1. Зверніться до кодів функцій, визначених протоколами зв'язку 2. Перевірте програмне забезпечення та мікропрограму, які ви використовуєте	-	X

		це останні версії. 3. Занотуйте робочі процедури та повідомлення про помилки, а потім зверніться до свого дистриб'ютора.		
8105	Вміст завантаженої програми неправильний. Неправильний синтаксис програми.	1. Повторне редагування програми та повторне завантаження програми та параметрів. 2. Перевірте, чи надійно підключено комунікаційні кабелі. 3. Збережіть і стисніть проект, а потім надішліть його своєму дистриб'ютору.	-	X
8106	Вміст завантаженої програми неправильний. Довжина коду виконання перевищує обмеження.	1. Повторне редагування програми та повторне завантаження програми та параметрів. 2. Збережіть і стисніть проект, а потім надішліть його своєму дистриб'ютору.	-	X
8107	Вміст завантаженої програми неправильний. Довжина вихідного коду перевищує обмеження.	1. Повторне редагування програми та повторне завантаження програми та параметрів. 2. Збережіть і стисніть проект, а потім надішліть його своєму дистриб'ютору.	-	X
8000 - 8FFF	Помилки редагування ПЛК в ISPSOft	1. Перевірте, чи програмне забезпечення та мікропрограми, які ви використовуєте, мають останню версію. 2. Занотуйте робочі процедури та повідомлення про помилки, а потім зверніться до свого дистриб'ютора.	-	X

7.3 Усунення несправностей для інших модулів введення/виведення

● Знайомство з модулями усунення несправностей

Цифровий вхід/вивід, аналоговий вхід/вихід, вимірювання температури, позиціонування, високошвидкісний лічильник, датчик навантаження та мережеві модулі можуть бути встановлені в систему серії AS. Є 2 типи кодів помилок; помилка та попередження. Модуль центрального процесора та його модулі припиняють роботу, коли виникають помилки. Модулі центрального процесора та його модулі не припиняють працювати, коли спрацьовують попередження.

7.3.1 Усунення несправностей для аналогових модулів (AD/DA/XA) і температурних модулів (RTD/TC)

7.3.1.1 Світлодіодні індикатори ERROR

Ви можете налаштувати HWCONFIG, щоб наведені нижче помилки відображалися як попередження, коли вони виникають. В іншому випадку, коли виникає помилка, з'являється лише повідомлення про помилку.

Код помилки	опис	Рішення
16#1605	Збій обладнання	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1607	Зовнішня напруга ненормальна.	Перевірте джерело живлення.
16#1608	Заводське калібрування або CJC є ненормальними.	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.

7.3.1.2 Світлодіодні індикатори ERROR блимають кожні 0,2 секунди

Наступні помилки вказуються як попередження, щоб гарантувати, що модуль ЦП може працювати, навіть якщо попередження викликано його модулями. Наступні перші 4 коди помилок встановлено як попередження за замовчуванням у HWCONFIG; ЦП повинен ЗУПИНИТЕ роботу негайно, коли виникають перші 4 помилки.

Код помилки	опис	Рішення
16#1801	Зовнішня напруга ненормальна.	Перевірте джерело живлення.
16#1802	Збій обладнання	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1804	Заводське калібрування є ненормальним.	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1807	CJC ненормальний.	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1808	Сигнал, отриманий каналом 1, перевищує діапазон аналогових входів (температура).	Перевірте сигнал, отриманий каналом 1
16#1809	Сигнал, отриманий каналом 2, перевищує діапазон аналогових входів (температура).	Перевірте сигнал, отриманий каналом 2
16#180A	Сигнал, отриманий каналом 3, перевищує діапазон аналогових входів (температура).	Перевірте сигнал, отриманий каналом 3
16#180B	Сигнал, отриманий каналом 4, перевищує діапазон аналогових входів (температура).	Перевірте сигнал, отриманий каналом 4

Код помилки	опис	Рішення
16#180C	Сигнал, отриманий каналом 5, перевищує діапазон аналогових входів (температура).	Перевірте сигнал, отриманий каналом 5
16#180D	Сигнал, отриманий каналом 6, перевищує діапазон аналогових входів (температура).	Перевірте сигнал, отриманий каналом 6
16#180E	Сигнал, отриманий каналом 7, перевищує діапазон аналогових входів (температура).	Перевірте сигнал, отриманий каналом 7
16#180F	Сигнал, отриманий каналом 8, перевищує діапазон аналогових входів (температура).	Перевірте сигнал, отриманий каналом 8
-	Після ввімкнення живлення модуль HE отримує жодних запитів на виявлення від ЦП ПЛК.	Перевірте з'єднання між процесором і модулем або знову під'єднайте їх.

7.3.2 Усунення несправностей для модуля позиціонування AS02/04PU

7.3.2.1 Світлодіодні індикатори ERROR блимають кожні 0,2 секунди

Код помилки	опис	Рішення
16#1802	Збій обладнання	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.

7.3.3 Усунення несправностей для модуля високошвидкісного лічильника AS02HC

7.3.3.1 Світлодіодні індикатори ERROR

Код помилки	опис	Рішення
16#1605	Підрахований результат у зафіксованій області не зберігається (основна помилка)	Підраховані дані втрачаються. Вимкніть і знову ввімкніть живлення модуля. Код помилки видається системою. Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1606	Налаштування модуля в решітчастій зоні не зберігаються. (основна помилка)	Дані налаштувань модуля втрачено. Вимкніть і знову ввімкніть живлення модуля. Завантажте налаштування HWCONFIG ще раз, щоб очистити код помилки. Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1607	Помилка налаштування модуля (основна помилка)	Перевірте, чи параметр у HWCONFIG відповідає фактичному розміщенню. Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.

7.3.3.2 Світлодіодні індикатори ERROR блимають кожні 0,5 секунди

Наступні помилки вказуються як попередження, щоб гарантувати, що модуль ЦП може працювати, навіть якщо попередження викликані його модулями AIO.

Код помилки	опис	Рішення
16#1800	Лічильник переповнення / недоповнення на CH1	Перевірте результат лічильника. Якщо сигналізація не потрібна, вимкніть функцію виведення тривоги в HWCONFIG. Щоб очистити код помилки, скористайтеся будь-яким із наведених нижче способів: очистіть, скиньте, попередньо налаштуйте лічильник, перезапустіть модуль або знову виконайте інструкцію DHCCNT.
16#1801	Лічильник переповнення / недоповнення на CH2	
16#1802	Лінійний рахунок перевищує встановлену верхню/нижню межу на CH1	Перевірте сигнал, отриманий каналами 1 і 2. Апаратний лічильник все ще підраховує; коли число накопичено до максимального значення та повернеться до мінімуму, код помилки буде видалено.
16#1803	Лінійне число перевищує встановлену верхню/нижню межу на CH2	
16#1804	Зміни щодо положення кодера SSI, що перевищує обмеження на CH1	Перевірте, чи немає перерв, і перевірте специфікацію пристрою, щоб перевірити, чи зміщення збігається з фактичним розташуванням. Коли наступне зчитування буде нормальним, код помилки буде видалено.
16#1805	Зміни відносно положення кодера SSI, що перевищує обмеження на CH2	
16#1806	Ненормальний зв'язок SSI на каналі CH1	Перевірте виконання інструкції DHCCNT. Якщо це перевірка парності, перевірте, чи є переривання, і перевірте, чи правильний формат даних. Перевірте надійність електропроводки пристрою та нормальне живлення кодера.
16#1807	Ненормальний зв'язок SSI на каналі CH2	
16#1808	Абсолютне положення SSI перетинає нульову точку на CH1	Перевірте специфікацію абсолютного кодера SSI та відповідно змініть налаштування. Якщо сигналізація не потрібна, вимкніть функцію виведення тривоги в HWCONFIG. Щоб очистити код помилки, скористайтеся будь-яким із наведених нижче способів: очистіть, скиньте, попередньо налаштуйте лічильник, перезапустіть модуль або знову виконайте інструкцію DHCCNT.
16#1809	Абсолютне положення SSI перетинає нульову точку на CH2	

7.3.4 Усунення несправностей для модуля тензодатчика AS02LC

7.3.4.1 Світлодіодні індикатори ERROR

Ви можете налаштувати HWCONFIG, щоб наведені нижче помилки відображалися як попередження, коли вони виникають. В іншому випадку, коли виникає помилка, з'являється лише повідомлення про помилку.

Код помилки	опис	Рішення
16#1605	Збій обладнання (тобто плати драйвера)	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1607	Зовнішня напруга ненормальна.	Перевірте джерело живлення.

7.3.4.2 Світлодіодні індикатори ERROR блимають кожні 0,2 секунди

Наступні помилки вказуються як попередження, щоб гарантувати, що модуль ЦП може працювати, навіть якщо попередження викликано його модулями. Наступні перші 3 коди помилок встановлено як попередження за замовчуванням у HWCONFIG; ЦП повинен негайно ЗУПИНИТИ роботу, коли виникають перші 3 помилки.

Код помилки	опис	Рішення
16#1801	Зовнішня напруга ненормальна.	Перевірте джерело живлення.
16#1802	Збій обладнання	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1807	Несправність дайверської дошки	Перевірте, чи на клемах не впливають перешкоди чи короткого замикання (перевірте EXC+ і EXC-). Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1808	Сигнал, отриманий каналом 1, перевищує діапазон аналогових входів або напруга SEN є ненормальною.	Перевірте сигнал, отриманий каналом 1, і кабельні з'єднання.
16#1809	Сигнал, отриманий каналом 1, перевищує ліміт ваги.	Перевірте значення, введене в канал 1, і налаштування максимальної ваги.
16#180A	Заводське калібрування в каналі 1 неправильне.	Перевірте калібрування ваги в каналі 1.
16#180B	Сигнал, отриманий каналом 2, перевищує діапазон аналогових входів або напруга SEN є ненормальною.	Перевірте сигнал, отриманий каналом 2, і кабельні з'єднання.
16#180C	Сигнал, отриманий каналом 2, перевищує обмеження ваги.	Перевірте значення, введене в канал 2, і налаштування максимальної ваги.
16#180D	Заводське калібрування каналу 2 неправильне.	Перевірте калібрування ваги в каналі 1.
-	Після ввімкнення живлення модуль НЕ отримує жодних запитів на виявлення від ЦП ПЛК.	Перевірте з'єднання між процесором і модулем або знову під'єднайте їх.

7

7.3.5 Усунення несправностей для модуля AS04SIL IO-Link як модуля зв'язку

7.3.5.1 Світлодіодні індикатори ERROR (статус модуля) блимають кожні 0,2 секунди

Наступні коди помилок ідентифікують можливі помилки, коли модуль AS04SIL встановлено з правого боку модуля ЦП або RTU і діє як модуль зв'язку.

Код помилки	опис	Рішення
16#1605	Збій обладнання	1. Перевірте надійність встановлення модуля. 2. Встановіть новий AS04SIL або, якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.
16#1606	Електроживлення 24 В постійного струму недостатнє, а потім відновлюється від низької напруги менше ніж за 10 мс.	Перевірте, чи нормальне живлення 24 В до модуля.

7.3.5.2 Світлодіодні індикатори ERROR (статус мережі) увімкнені

Наступні коди помилок ідентифікують можливі помилки, коли модуль AS-SIL встановлено з правого боку модуля ЦП або RTU і діє як модуль зв'язку.

Код помилки	опис	Рішення
16#1802	Відсутнє зовнішнє джерело живлення	Перевірте зовнішнє джерело живлення

7.3.5.3 Світлодіодні індикатори ERROR (статус мережі) блимають кожні 1 секунду

Наступні коди помилок ідентифікують можливі помилки, коли модуль AS-SIL встановлено з правого боку модуля ЦП або RTU і діє як модуль зв'язку.

Код помилки	опис	Рішення
16#1800	Сталася помилка в IO-Link Master	Щоб отримати додаткові відомості, перегляньте наступну таблицю кодів подій IO-Link.
16#1801	У пристрої IO-Link сталася помилка	Щоб отримати додаткові відомості, перегляньте наступну таблицю кодів подій IO-Link.
16#1803	Помилка під час завантаження таблиць зіставлення пристроїв IO-Link	Перезавантажте конфігурацію програмним забезпеченням.
16#1804	Не вдалося переключити набір параметрів даних процесу	Перевірте, чи конфігурації фактично підключеного пристрою відповідають конфігураціям у програмному забезпеченні.
16#1805	У IO-Link через комунікаційний порт 1 сталася помилка підключення	1. Вимкніть зовнішнє живлення на 3 секунди, а потім увімкніть його знову. 2. Перезавантажте конфігурацію програмним забезпеченням.
16#1806	Виникає помилка з'єднання в IO-Link через комунікаційний порт 2	
16#1807	У IO-Link через комунікаційний порт 3 сталася помилка підключення	
16#1808	Виникає помилка з'єднання в IO-Link через комунікаційний порт 4	
16#1809	Помилка сканування пристрою, і сканування примусово зупинено	1. Вимкніть зовнішнє живлення на 3 секунди, а потім увімкніть його знову. 2. Повторно проскануйте весь пристрій

Таблиця кодів подій IO-Link

Ви можете знайти коди подій IO-Link в області обміну даними модуля, де зберігаються дані про стан пристрою комунікаційного порту 1~4. Перегляньте посібник з експлуатації пристрою IO-Link, якщо подія сталася з пристроїв IO-Link.

Коди подій IO-Link	Тип			Подія	Рішення	Source	
	УВАГА	Помилка	Сповідення			IO-Link Master	Пристрій IO-Link
16#4000		V		Температурне перевантаження пристрою	Менше навантаження		V
16#4210	V			Перевищення температури пристрою	Прозоре джерело тепла		V
16#5101		V		Перегорів запобіжник пристрою	Замінити запобіжник		V
16#5110	V			Перевищення напруги джерела живлення	Перевірити допуск		V
16#5111	V			Занижена напруга живлення	Перевірити допуск		V
16#6320		V		Помилка параметра	Перевірте технічні характеристики пристрою		V
16#6321		V		Параметр відсутній	Перевірте характеристики пристрою		V
16#7710		V		Коротке замикання пристрою	Перевірте встановлення		V

Коди подій IO-Link	Тип			Подія	Рішення	Source	
	УВАГА	Помилка	Сповіщення			IO-Link Master	Пристрій IO-Link
16#8C10	V			Перевищення діапазону змінної процесу	Перевірте дані процесу		V
16#8C20		V		Перевищення діапазону вимірювання	Перевірити заявку		V
16#8C30	V			Зниження діапазону змінної процесу	Перевірте дані процесу		V
16#8CA0	V			Немає підключеного пристрою IO-Link	Перевірте встановлення	V	
16#8CA1	V			Версія протоколу IO-Link відрізняється від налаштованої.	Використовуйте відповідний файл IODD і налаштуйте його знову.	V	
16#8CA2	V			Підключений пристрій відрізняється від налаштованого в програмному забезпеченні	Перевірте налаштування та встановлення	V	
16#8CA3				Зарезервований		V	
16#8CA4		V		Коротке замикання технологічного кабелю пристрою IO-Link	Перевірте встановлення	V	
16#8CA5	V			Головна температура перевищує 135°C	Прозоре джерело тепла	V	
16#8CA6		V		Головна температура перевищує 160°C	Чисте джерело тепла та менше навантаження	V	
16#8CA7	V			Низька напруга живлення пристрою L+ (<18 V)	Перевірте зовнішнє джерело живлення	V	
16#8CA8		V		Низька напруга живлення пристрою L+ (<9 V)	Перевірте зовнішнє джерело живлення	V	
16#8CA9	V			Незаконний ідентифікатор пристрою	Перевірте характеристики пристрою	V	
16#8CAA	V			Перевищення діапазону даних процесу IO-Link	Перевірте характеристики пристрою	V	
16#8CAB	V			Перевищення діапазону даних процесу	Проскануйте пристрій і знову завантажте конфігурацію	V	
16#8CAC		V		Помилка зберігання даних	Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.	V	
0xFF21			V	Новий підключений пристрій		V	
0xFF22			V	Пристрій відключено	Перевірте встановлення	V	
0xFF23			V	Невідповідність ідентифікації зберігання даних	Зabloкуйте доступ до сховища даних і встановіть резервне копіювання/відновлення, а потім резервне копіювання даних відповідно до фактичного розміщення.	V	
0xFF24			V	Пам'яті даних недостатньо	Перевірте характеристики пристрою	V	
0xFF25			V	Доступ до параметрів зберігання даних заборонено	Перевірте характеристики пристрою	V	

7.3.6 Усунення несправностей для модуля AS00SCM як модуля зв'язку

7.3.6.1 Світлодіодний індикатор ERROR

Наступні коди помилок ідентифікують можливі помилки, коли модуль AS00SCM встановлено з правого боку модуля ЦП і діє як модуль зв'язку.

Код помилки	опис	Рішення
16#1605	Збій обладнання	3. Перевірте надійність встановлення модуля. 4. Встановіть новий AS00SCM або зверніться до свого дистриб'ютора.
16#1606	Помилки налаштувань функціональної карти модуля зв'язку	1. Перевірте, чи надійно встановлено функціональну карту. 2. Встановіть нову функціональну картку або зверніться до свого дистриб'ютора. 3. Перевірте, чи параметр у HWCONFIG узгоджується з налаштуванням функціональної карти. 4. Встановіть новий AS00SCM або зверніться до свого дистриб'ютора.

7.3.6.2 Світлодіодні індикатори ERROR блимають кожні 0,5 секунди

Наступні коди помилок ідентифікують можливі помилки, коли модуль AS00SCM встановлено з правого боку модуля ЦП і діє як модуль зв'язку.

Код помилки	опис	Рішення
16#1802	Неправильні параметри	Перевірте параметр у HWCONFIG. Завантажте параметр ще раз.
16#1803	Час очікування зв'язку	1. Перевірте, чи правильно під'єднано кабель зв'язку. 2. Перевірте, чи правильно встановлено номер станції та формат зв'язку. 3. Перевірте, чи правильно працює з'єднання з функціональною карткою.
16#1804	Налаштування UD Link неправильне.	1. Перевірте налаштування UD Link. 2. Перевірте налаштування попереджень у ПЛК.

Наступні коди помилок можна переглянути лише за допомогою SCMSoft; коли виникають наведені нижче помилки, вони не відображаються на світлодіодних індикаторах, і система не надсилає повідомлення про помилки модулю ЦП.

Код помилки	опис	Рішення
16#0107	Налаштування в HWCONFIG і налаштування вручну не відповідають функціональній картці 1.	Перевірте налаштування в HWCONFIG і ручні налаштування для функціональної карти 1.
16#0108	Налаштування в HWCONFIG і ручні налаштування не узгоджуються для функціональної карти 2.	Перевірте налаштування в HWCONFIG і ручні налаштування для функціональної карти 2.
16#0201	Неправильні параметри	Перевірте параметр у HWCONFIG. Завантажте параметр ще раз.
16#0301	Тайм-аут зв'язку функціональної картки 1	1. Перевірте, чи правильно встановлено номер станції та формат зв'язку. 2. Перевірте, чи правильно працює з'єднання з функціональною карткою.

Код помилки	опис	Рішення
16#0302	Час очікування зв'язку функціональної карти 2	1. Перевірте, чи правильно встановлено номер станції та формат зв'язку. 2. Перевірте, чи правильно працює з'єднання з функціональною карткою.
16#0400	Недійсний ідентифікатор UD Link Group для функціональної карти 1	1. Перевірте налаштування UD Link. 2. Перевірте налаштування попереджень у ПЛК.
16#0401	Недійсний ідентифікатор UD Link Group для функціональної карти 2	1. Перевірте налаштування UD Link. 2. Перевірте налаштування попереджень у ПЛК.
16#0402	Недійсна команда UD Link для функціональної карти 1	1. Перевірте налаштування UD Link. 2. Перевірте налаштування попереджень у ПЛК.
16#0403	Недійсна команда UD Link для функціональної карти 1	1. Перевірте налаштування UD Link. 2. Перевірте налаштування попереджень у ПЛК.

7.3.7 Усунення несправностей для модуля AS00SCM як віддаленого модуля

Помилки від віддалених модулів розглядаються як попередження для модулів ЦП серії AS. Світлодіодний індикатор модуля ЦП блимає, і модуль ЦП продовжує працювати. Використовуйте прапор SM30 для керування представленням помилок у віддалених модулях.

7.3.7.1 Світлодіодні індикатори ERROR

Коди помилок:

Код помилки	опис	Рішення
16#1301	Збій обладнання	1. Перевірте, чи надійно встановлено модуль. 2. Змініть та встановіть новий AS00SCM або зверніться до свого дистриб'ютора.
16#1302	Помилки налаштувань функціональної карти модуля зв'язку	1. Перевірте, чи надійно встановлено функціональну плату з картою AS-FCOPM. 2. Змініть та встановіть нову функціональну карту або зверніться до свого дистриб'ютора. 3. Перевірте, чи параметр у HWCONFIG узгоджується з налаштуванням функціональної карти. 4. Встановіть новий AS00SCM або зверніться до свого дистриб'ютора.
16#1304	Більше восьми віддалених модулів з правого боку від модуля ЦП.	Перевірте загальну кількість віддалених модулів праворуч від модуля ЦП (максимум 8).

7.3.7.2 Світлодіодні індикатори ERROR блимають кожні 1 секунду

Тип попередження кодів помилок:

Код помилки	опис	Рішення
16#1506	Віддалений модуль було зупинено.	Цей код помилки має працювати з налаштуваннями віддаленого модуля серії AS у ISPSOFT. Коли з'являється цей код помилки, це вказує на те, що віддалений модуль було зупинено: головний модуль відключено, головний повторно підключено, аварійний сигнал модуля вводу-виводу або тайм-аут модуля вводу-виводу. Перевірте та усуньте проблему, а потім вимкніть, а потім увімкніть віддалений модуль, щоб оновити його стан. Додаткову інформацію див. у розділі 9.4.3 посібника з модулів серії AS.

7.3.7.3 Світлодіодні індикатори ERROR блимають кожні 0,5 секунди

Коди помилок типу попередження:

Код помилки	опис	Рішення
16#1500	Час очікування зв'язку віддаленого модуля	Перевірте мережевий кабель віддалених модулів.
16#1502	Неправильні параметри	Перевірте параметр у HWCONFIG. Завантажте параметр ще раз.
16#1503	Час очікування зв'язку модуля розширення	Переконайтеся, що модуль належним чином підключено до модуля ЦП, і знову ввімкніть модулі.
16#1505	Фактичне розташування модуля розширення не відповідає конфігураціям.	Перевірте фактичне розташування та конфігурації.
16#1604	Час очікування зв'язку модуля розширення	1.Переконайтеся, що модуль належним чином підключено до модуля ЦП, і знову ввімкніть модулі. 2.Якщо проблема не зникає, зверніться до місцевих авторизованих дистриб'юторів.

7.3.7.4 Світлодіодні індикатори ERROR блимають кожні 0,2 секунди

Це трапляється, коли джерело живлення 24 В постійного струму для віддаленого модуля недостатнє. Перевірте джерело живлення. Якщо джерело живлення нормальне, вийміть модуль розширення з модуля ЦП, а потім перевірте, чи не працює віддалений модуль SCM.

Коди помилок:

Код помилки	опис	Рішення
16#1303	Електроживлення 24 В постійного струму недостатнє, а потім відновлюється від низької напруги менше ніж за 10 мс.	Перевірте, чи нормальне живлення 24 В до модуля.

7.3.8 Усунення несправностей для модулів AS01DNET

Зверніться до розділів 7.4.9 і 7.4.10 для отримання додаткової інформації.

7.4 Коди помилок і світлодіодні індикатори для модулів

7.4.1 Коди помилок і світлодіодні індикатори для модулів ЦП

А. Стовпці

- a. Код помилки: якщо в системі виникає помилка, генерується код помилки.
- a. Опис: Опис помилки
- b. Статус ЦП: якщо виникає помилка, ЦП припиняє роботу, продовжує працювати або відображає стан, який ви визначили для помилки.
 - Зупинка: ЦП припиняє роботу, коли виникає помилка.
 - Продовжити: ЦП продовжує працювати, коли виникає помилка.
- a. Статус світлодіодного індикатора: якщо виникає помилка, світлодіодний індикатор увімкнено, вимкнено або блимає.
 - ПОМИЛКА: системна помилка

● Описи

Тип модуля	світлодіодний індикатор	Описи
ЦП	Світлодіод помилки	Існує п'ять типів станів індикатора помилок для помилок модуля ЦП, включаючи світлодіодний індикатор ON, OFF, швидко блимає, нормально блимає та повільно блимає. Коли світлодіодний індикатор горить, блимає швидко/нормально, спочатку усуньте проблеми, щоб запустити модуль ЦП. Якщо світлодіодний індикатор повільно блимає, вказуючи на тип попередження про коди помилок, це не потребує негайної дії. Усуньте проблеми, коли модуль вимкнено. Тип помилки: ON: в модулі сталася серйозна помилка. Швидко блимає (кожні 0,2 секунди): нестабільне живлення або апаратний збій. Блимає нормально (кожні 0,5 секунди): помилки системної програми або система не працює. Тип попередження: Повільно блимає (кожну секунду і гасне на три секунди): попередження спрацьовує, але система все ще може працювати. OFF.: спрацьовує попередження, але система все ще може працювати. Ви можете змінити правила ініціювання попередження або використовувати SM/SR для показу попереджень.

Код помилки	опис	ЦП статус	Статус світлодіодного індикатора ERROR				
			ON	Швидко моргає	Моргає нормально	Повільно моргає	OFF
000A	Тайм-аут сканування	СТІЙ	V				
000C	Програма в ПЛК пошкоджена.	СТІЙ			V		
000D	Початкове значення пристрою та програма ПЛК завантажено не повністю.	СТІЙ			V		
0010	Доступ до пам'яті в ЦП заборонено.	СТІЙ			V		
0011	Ідентифікатор ПЛК неправильний.	Продовжити					V
0012	Пароль ПЛК неправильний.	Продовжити					V
0020	Швидкість роботи процесора не на найвищому рівні.	СТІЙ			V		
0021	Не вдалося відновити параметри пристрою Delta до значень за замовчуванням	X			X		
Примітка. У HWCONFIG у разі виникнення цієї помилки можна налаштувати, щоб ЦП ПЛК припиняв роботу або ні. Якщо налаштування полягає в тому, щоб ЦП ПЛК зупинявся, коли виникає помилка 16#0021, ЦП ПЛК припинить роботу, а світлодіод ERROR почне блимати. Якщо налаштування полягає в тому, щоб ЦП ПЛК продовжував працювати, помилка буде вказана в журналі помилок, але світлодіод ERROR не блиматиме.							
0026	Час реального часу не може відстежувати поточний час (світлодіод акумулятора блимає).	Продовжити					
0027	Батарея розряджена (індикатор батареї світиться.)	Продовжити					
0028	Немає відповіді від RTC	Продовжити					V
002A	Напруга живлення 24 В постійного струму недостатня, а потім відновлюється від низької напруги менше ніж за 10 мс.	Продовжити		V			
002D	Перевищено максимальну кількість спроб пароля ПЛК.	Продовжити					V
002E	Доступ до зовнішньої пам'яті процесора заборонено.	СТІЙ			V		
002F	Програми ПЛК не відповідають системним журналам.	СТІЙ			V		
0040	Перетворене значення у функціональній картці 1 (AS-F2AD) або у вбудованому вхідному каналі AD 1 перевищує діапазон налаштування.	Продовжити				V	
0041	Перетворене значення у функціональній картці 1 (AS-F2AD) або у вбудованому вхідному каналі AD 2 перевищує діапазон налаштування.	Продовжити				V	
0042	Перетворене значення у вхідному каналі 1 функціональної карти 2 (AS-F2AD) перевищує діапазон налаштування.	Продовжити				V	
0043	Перетворене значення у вхідному каналі 2 функціональної карти 2 (AS-F2AD) перевищує діапазон налаштування.	Продовжити				V	
0044	З'єднання втрачено в режимі струмового входу 4 mA - 20 mA для функціональної карти 1 (AS-F2AD) або для вбудованого вхідного каналу AD 1.	Продовжити				V	
0045	З'єднання втрачено в режимі струмового входу 4 mA - 20 mA для функціональної карти 1 (AS-F2AD) або для вбудованого вхідного каналу AD 2.	Продовжити				V	
0046	З'єднання втрачено в режимі струмового входу 4 mA - 20 mA для вхідного каналу 1 функціональної карти 2 (AS-F2AD).	Продовжити				V	
0047	З'єднання втрачено в режимі струмового входу 4 mA - 20 mA для вхідного каналу 2 функціональної карти 2 (AS-F2AD).	Продовжити				V	
0050	Пам'ять у зафіксованих спеціальних допоміжних реле є ненормальною.	Продовжити					V
0051	Зафіксовані спеціальні регістри даних є ненормальними.	Продовжити					V

Код помилки	опис	ЦП статус	Статус світлодіодного індикатора ERROR				
			ON	Швидко моргає	Моргає нормально	Повільно моргає	OFF
0052	Пам'ять у зафіксованих допоміжних реле ненормальна.	Продовжити					V
0054	Заблоковані лічильники ненормальні.	Продовжити					V
0055	Зафіксовані 32-розрядні лічильники є ненормальними.	Продовжити					V
0056	Заблоковане спеціальне допоміжне реле є ненормальним.	Продовжити					V
0059	Зафіксовані регістри даних є ненормальними.	Продовжити					V
005D	Модуль процесора не виявляє карту пам'яті.	Продовжити					V
005E	Картка пам'яті ініціалізована неправильно.	Продовжити					V
0063	Під час запису даних на карту пам'яті виникає помилка.	Продовжити					V
0064	Неможливо прочитати файл на карті пам'яті.	Продовжити					V
0070	Фактичне розташування функціональних карток не відповідає налаштуванням.	СТІЙ			V		
0102	Номер переривання перевищує діапазон.	СТІЙ			V		
0202	Інструкція MC перевищує діапазон.	СТІЙ			V		
0302	Інструкція MCR перевищує діапазон.	СТІЙ			V		
0D03	Операнди, які використовуються в DHSCS, не використовуються належним чином.	СТІЙ			V		
0E05	Операнди HCXXX, які використовуються в DCNT, не використовуються належним чином.	СТІЙ			V		
1000 - 10FF	Системна помилка	СТІЙ			V		
1300 - 130F	У віддалених модулях виникли помилки	Продовжити				V	
1402	Розташування модулів введення/виведення не відповідає налаштуванням.	СТІЙ			V		
140B	Кількість комунікаційних модулів перевищує ліміт у чотири.	СТІЙ			V		
140C	Кількість підключених модулів позиціонування перевищує максимальну кількість 8.	СТІЙ			V		
140D	Кількість модулів розширення перевищує 32.	СТІЙ			V		
140E	Кількість віддалених модулів перевищує ліміт у вісім праворуч від модуля ЦП.	СТІЙ			V		
1500	З'єднання втрачено у віддалених модулях	Продовжити				V	
1502 - 150F	У віддалених модулях виникли помилки	Продовжити				V	
1600	Ідентифікатор модуля розширення перевищує діапазон.	СТІЙ			V		
1601	Ідентифікатор модуля розширення не може бути встановлений.	СТІЙ			V		
1602	Ідентифікатор модуля розширення дублюється.	СТІЙ			V		
1603	Модуль розширення не працює.	СТІЙ			V		
1604	Час очікування зв'язку модуля розширення	СТІЙ			V		
1605	Збій обладнання	СТІЙ			V		
1606	Помилки налаштувань функціональної карти модуля зв'язку	СТІЙ			V		

Код помилки	опис	ЦП статус	Статус світлодіодного індикатора ERROR				
			ON	Швидко моргає	Моргає нормально	Повільно моргає	OFF
1607	Зовнішня напруга ненормальна.	СТІЙ			V		
1608	Внутрішнє заводське калібрування або CJC не відповідає нормі.	СТІЙ			V		
1609 - 160F	Зарезервовано (коди помилок для модулів розширення)	СТІЙ			V		
1800 - 180F	Виникли помилки в модулях розширення	Продовжити				V	
1900 - 191C	У веденому модулі зв'язку CANopen/ECAT виникли помилки серцевого ритму/втрати з'єднання.	Продовжити				V	
1950	Ініціалізація комунікаційного модуля CANopen ще не завершена, інструкції CANopen не можуть бути виконані.	Продовжити					V
19B0	У підпорядкованому модулі зв'язку CANopen стався тайм-аут серцевого ритму.	Продовжити					V
19B1	Довжина даних PDO (об'єкт даних процесу) у підпорядкованому режимі не відповідає налаштуванню.	Продовжити					V
19E1	Довжина даних PDO (об'єкт даних процесу) у підлеглому режимі не узгоджується з параметрами списку сканування. Зверніться до розділу 10.5 Описи зв'язку CANopen у Посібнику з апаратного забезпечення та експлуатації серії AS, щоб дізнатися більше про коди помилок 19E1–19E8.	Продовжити					V
19E2	PDO в веденому режимі не отримано.	Продовжити					V
19E3	Функція автоматичного завантаження SDO не працює під час першого запуску.	Продовжити					V
19E4	Конфігурації PDO встановлено неправильно.	Продовжити					V
19E5	Основні налаштування не узгоджуються з підключеним веденим.	Продовжити					V
19E6	Slave HE існує в мережі.	Продовжити					V
19E7	Час очікування підлеглого керування помилками	Продовжити					V
19E8	Ідентифікатори вузлів головного та підлеглого дублюються.	Продовжити					V
19F3	Помилка в конфігурації	Продовжити					V
19F4	Зв'язок CANopen знаходиться в стані BUS-OFF.	Продовжити					V
19FB	Регістри надсилання перевищують діапазон.	Продовжити					V
19FC	Приймальні регістри перевищують діапазон.	Продовжити					V
2001	Не використовується карта FCOMP або не в правильному режимі для ASDA-A2 під час використання інструкції зв'язку CANopen.	Продовжити					V
2003	Пристрій, який використовується в програмі, перевищує діапазон пристрою або є незаконним.	Продовжити					V
200A	Недійсна інструкція	СТІЙ			V		
200B	Операнд n або інші постійні операнди К/Н перевищують діапазон.	Продовжити					V
200C	Операнди перекриваються.	Продовжити					V

Код поми лки	опис	ЦП статус	Статус світлодіодного індикатора ERROR				
			ON	Швидко моргає	Моргає нормально	Повільно моргає	OFF
200D	Двійкове перетворення десяткового числа в двійковий код неправильне.	Продовжити					V
200E	Рядок не закінчується на 00.	Продовжити					V
2012	Некоректна операція ділення	Продовжити					V
2013	Значення перевищує діапазон значень, які можуть бути представлені числами з плаваючою комою.	Продовжити					V
2014	Завдання, визначене інструкцією TKON або YKOFF, є неправильним або виходить за межі діапазону.	Продовжити					V
2017	Інструкція BREAK записана поза циклом FOR-NEXT.	Продовжити					V
2027	Такого номера таблиці планування позицій немає або формат неправильний.	Продовжити					V
2028	Виконується інструкція високошвидкісного виведення. Одночасно може бути виконана лише одна інструкція.	Продовжити					V
2030	Вихідна частота, що виконується інструкцією CSFO, перевищує максимальну вихідну частоту.	Продовжити					V
2031	Якщо вхідний імпульс становить 0 Гц і триває більше 2 секунд, ПЛК автоматично припиняє виведення.	Продовжити					V
6004	Фільтр IP-адрес налаштовано неправильно.	Продовжити					V
600D	Порт RJ45 не підключено.	Продовжити					V
6010	Кількість підключень MODBUS TCP перевищує діапазон.	Продовжити			V		
6011	Кількість підключень EtherNet/IP перевищує допустимий діапазон.	Продовжити			V		
6012	Є пристрої, які використовують ту саму IP-адресу.	Продовжити					V
6100	З'єднання електронної пошти зайняте.	Продовжити					V
6103	Режим активації вкладення в електронному листі встановлено неправильно.	Продовжити					V
6104	Вкладення в електронному листі не існує.	Продовжити					V
6105	Вкладення в електронному листі завелике.	Продовжити					V
6106	Тайм-аут відповіді сервера SMTP.	Продовжити					V
6107	Тайм-аут відповіді сервера SMTP.	Продовжити					V
6108	Помилка перевірки SMTP	Продовжити					V
6200	IP-адреса віддаленого зв'язку, встановлена у функції TCP-сокета, є незаконною.	Продовжити					V
6201	Порт локального зв'язку, встановлений у функції TCP-сокета, є недопустимим.	Продовжити					V
6202	Порт віддаленого зв'язку, встановлений у функції TCP-сокета, є незаконним.	Продовжити					V
6203	Пристрій, з якого надсилаються дані у функції TCP-сокета є незаконним.	Продовжити					V
6206	Пристрій, який отримує дані у функції TCP-сокета, є незаконним.	Продовжити					V
6208	Дані, отримані через сокет TCP, перевищують діапазон пристрою.	Продовжити					V
6209	IP-адреса віддаленого зв'язку, встановлена у функції сокета UDP, є недопустимою.	Продовжити					V

Код поми лки	опис	ЦП статус	Статус світлодіодного індикатора ERROR				
			ON	Швидко моргає	Моргає нормально	Повільно моргає	OFF
620A	Порт локального зв'язку, встановлений у функції сокета UDP, є недопустимим.	Продовжити					V
620C	Пристрій, з якого надсилаються дані у функції сокета UDP, є незаконним.	Продовжити					V
620F	Пристрій, який отримує дані у функції сокета UDP, є незаконним.	Продовжити					V
6210	Дані, отримані через сокет UDP, перевищують діапазон пристрою.	Продовжити					V
6212	Немає відповіді від віддаленого пристрою після періоду очікування.	Продовжити					V
6213	Отримані дані перевищують ліміт.	Продовжити					V
6214	Віддалений пристрій відмовляється підключитися.	Продовжити					V
6215	Розетка не відкрита.	Продовжити					V
6217	Розетка відкрита.	Продовжити					V
6218	Дані надіслано через сокет.	Продовжити					V
6219	Дані були отримані через сокет.	Продовжити					V
621A	Розетка закрита.	Продовжити					V
7011	Код функції зв'язку пристрою в COM1 неправильний.	Продовжити					V
7012	Адреса зв'язку пристрою, яка використовується в COM1, неправильна.	Продовжити					V
7013	Пристрій, який використовується в COM1, перевищує діапазон пристрою.	Продовжити					V
7014	Довжина даних пристрою в COM1 перевищує обмеження.	Продовжити					V
7017	Контрольна сума пристрою для послідовного порту зв'язку COM1 неправильна.	Продовжити					V
7021	Код функції зв'язку пристрою в COM2 неправильний.	Продовжити					V
7022	Адреса зв'язку пристрою, яка використовується в COM2, неправильна.	Продовжити					V
7023	Пристрій, який використовується в COM2, перевищує діапазон пристрою.	Продовжити					V
7024	Довжина даних пристрою в COM2 перевищує обмеження.	Продовжити					V
7027	Контрольна сума пристрою для послідовного порту зв'язку COM2 неправильна.	Продовжити					V
7031	Код функції зв'язку пристрою в Ethernet неправильний.	Продовжити					V
7032	Адреса зв'язку пристрою, яка використовується в Ethernet, неправильна.	Продовжити					V
7033	Пристрій, який використовується в Ethernet, перевищує діапазон пристрою.	Продовжити					V
7034	Довжина даних пристрою в Ethernet перевищує обмеження.	Продовжити					V
7037	Контрольна сума пристрою для послідовного порту зв'язку Ethernet неправильна.	Продовжити					V
7041	Код функції зв'язку пристрою в USB неправильний.	Продовжити					V

Код помилки	опис	ЦП статус	Статус світлодіодного індикатора ERROR				
			ON	Швидко моргає	Моргає нормально	Повільно моргає	OFF
7042	Адреса зв'язку пристрою, яка використовується в USB, неправильна.	Продовжити					V
7043	Пристрій, який використовується в USB, перевищує діапазон пристрою.	Продовжити					V
7044	Довжина комунікаційних даних пристрою в USB перевищує обмеження.	Продовжити					V
7047	Контрольна сума пристрою для послідовного порту зв'язку USB неправильна.	Продовжити					V
70B1	Код функції зв'язку пристрою у функціональній картці 1 неправильний.	Продовжити					V
70B2	Адреса зв'язку пристрою, яка використовується у функціональній картці 1, неправильна.	Продовжити					V
70B3	Пристрій, який використовується у функціональній картці 1, перевищує діапазон пристроїв.	Продовжити					V
70B4	Довжина даних пристрою у функціональній картці 1 перевищує обмеження.	Продовжити					V
70B7	Контрольна сума пристрою для послідовного порту зв'язку функціональної карти 1 неправильна.	Продовжити					V
70C1	Код функції зв'язку пристрою у функціональній картці 2 неправильний.	Продовжити					V
70C2	Адреса зв'язку пристрою, яка використовується у функціональній картці 2, неправильна.	Продовжити					V
70C3	Пристрій, який використовується у функціональній картці 2, перевищує діапазон пристроїв.	Продовжити					V
70C4	Довжина даних пристрою у функціональній картці 2 перевищує обмеження.	Продовжити					V
70C7	Контрольна сума пристрою для послідовного порту зв'язку функціональної карти 2 неправильна.	Продовжити					V
7203	Недійсний код функції зв'язку	Продовжити					V
8105	Вміст завантаженої програми неправильний. Неправильний синтаксис програми.	Продовжити					V
8106	Вміст завантаженої програми неправильний. Довжина коду виконання перевищує обмеження.	Продовжити					V
8107	Вміст завантаженої програми неправильний. Довжина вихідного коду перевищує обмеження.	Продовжити					V
8000 - 8FFF	Помилки редагування ПЛК в ISPSOft	Продовжити					V
C000 - CFFF	Синтаксична помилка в програмі PLC	СТІЙ			V		

7.4.2 Коды помилок і світлодіодні індикатори для аналогових/температурних модулів

Код помилки	опис	Статус світлодіодного індикатора ERROR	
		$A \rightarrow D /$ $D \rightarrow A /$ $A \leftrightarrow D$	ПОМИЛКА
16#1605	Збій обладнання	OFF	ON
16#1607	Зовнішня напруга ненормальна.	OFF	ON
16#1608	Заводське калібрування або CJC є ненормальними.	OFF	ON
16#1801*1	Зовнішня напруга ненормальна.	OFF	Блимає
16#1802*1	Збій обладнання	OFF	Блимає
16#1804*1	Заводське калібрування є ненормальним.	RUN: блимає STOP: OFF	Блимає
16#1807*1	CJC ненормальний.	OFF	Блимає
16#1808	Сигнал, отриманий каналом 1, перевищує діапазон аналогових входів (температура).	RUN: блимає STOP: OFF	Блимає
16#1809	Сигнал, отриманий каналом 2, перевищує діапазон аналогових входів (температура).		
16#180A	Сигнал, отриманий каналом 3, перевищує діапазон аналогових входів (температура).		
16#180B	Сигнал, отриманий каналом 4, перевищує діапазон аналогових входів (температура).		
16#180C	Сигнал, отриманий каналом 5, перевищує діапазон аналогових входів (температура).		
16#180D	Сигнал, отриманий каналом 6, перевищує діапазон аналогових входів (температура).		
16#180E	Сигнал, отриманий каналом 7, перевищує діапазон аналогових входів (температура).		
16#180F	Сигнал, отриманий каналом 8, перевищує діапазон аналогових входів (температура).		
-	Після ввімкнення живлення модуль HE отримує жодних запитів на виявлення від ЦП ПЛК.	OFF	Моргає один-два рази. Після припинення блимання протягом двох секунд починає блимати знову.

*1: наведені нижче помилки вказуються як попередження, щоб гарантувати, що модуль ЦП може працювати, навіть якщо попередження викликані його модулями AIO. Ви можете налаштувати HWCONFIG, щоб наступні перші 4 коди помилок відображалися як помилки, коли вони виникають.

7.4.3 Коди помилок і світлодіодні індикатори для модуля позиціонування AS02 / 04PU

Код помилки	опис	Статус світлодіодного індикатора ERROR	
		A → D	ПОМИЛКА
16#1802	Збій обладнання	OFF	Блимає

7.4.4 Коди помилок і світлодіодні індикатори для високошвидкісного модуля лічильника AS02HC

Код помилки	опис	Статус світлодіодного індикатора ERROR	
		ON	Блимає
16#1605	Підрахований результат у зафіксованій області не зберігається (основна помилка)	V	
16#1606	Налаштування модуля в решітчастій зоні не зберігаються. (основна помилка)	V	
16#1607	Помилка налаштування модуля (основна помилка)	V	
16#1800	Лічильник переповнення / недоповнення на CH1		V
16#1801	Лічильник переповнення / недоповнення на CH2		V
16#1802	Лінійний рахунок перевищує встановлену верхню/нижню межу на CH1		V
16#1803	Лінійне число перевищує встановлену верхню/нижню межу на CH2		V
16#1804	Зміни щодо положення кодера SSI, що перевищує обмеження на CH1		V
16#1805	Зміни відносно положення кодера SSI, що перевищує обмеження на CH2		V
16#1806	Ненормальний зв'язок SSI на каналі CH1		V
16#1807	Ненормальний зв'язок SSI на каналі CH2		V
16#1808	Абсолютне положення SSI перетинає нульову точку на CH1		V
16#1809	Абсолютне положення SSI перетинає нульову точку на CH2		V

7.4.5 Коди помилок і світлодіодні індикатори для модуля тензодатчика AS02LC

Код помилки	опис	Статус світлодіодного індикатора ERROR	
		A → D	ПОМИЛКА
16#1605	Збій апаратного забезпечення (дайверська плата включно)	OFF	ON
16#1607	Зовнішня напруга ненормальна.	OFF	ON
16#1801*1	Зовнішня напруга ненормальна.	OFF	Блимає
16#1802*1	Збій обладнання	OFF	Блимає
16#1807*1	Несправність дайверської дошки	OFF	Блимає
16#1808	Сигнал, отриманий каналом 1, перевищує діапазон аналогових входів або напруга SEN є ненормальною.	RUN: блимає STOP: OFF	Блимає
16#1809	Сигнал, отриманий каналом 1, перевищує ліміт ваги.		
16#180A	Заводське калібрування в каналі 1 неправильне.		
16#180B	Сигнал, отриманий каналом 2, перевищує діапазон аналогових входів або напруга SEN є ненормальною.		
16#180C	Сигнал, отриманий каналом 2, перевищує обмеження ваги.		
16#180D	Заводське калібрування каналу 2 неправильне.		
-	Після ввімкнення живлення модуль НЕ отримує жодних запитів на виявлення від ЦП ПЛК.	OFF	Моргає один-два рази. Після припинення блимання протягом двох секунд починає блимати знову.

*1: наведені нижче помилки вказуються як попередження, щоб гарантувати, що модуль ЦП може працювати, навіть якщо попередження викликані його модулями. Ви можете налаштувати HWCONFIG, щоб наступні перші 3 коди помилок відображалися як помилки, коли вони виникають.

7.4.6 Коди помилок і світлодіодні індикатори для модуля AS04SIL IO-Link як модуля зв'язку

Код помилки	опис	Світлодіодний індикатор MS Status	
		ON	Блимає
16#1605	Збій обладнання		✓
16#1606	Електроживлення 24 В постійного струму недостатнє, а потім відновлюється від низької напруги менше ніж за 10 мс.		✓

Код помилки	опис	NS світлодіодний індикатор	
		ON	Блимає
16#1800	Сталася помилка в IO-Link Master		✓
16#1801	У пристрої IO-Link сталася помилка		✓
16#1802	Відсутнє зовнішнє джерело живлення	✓	
16#1803	Помилка під час завантаження таблиць зіставлення пристроїв IO-Link		✓

Код помилки	опис	NS світлодіодний індикатор	
		ON	Блимає
16#1804	Не вдалося переключити набір параметрів даних процесу		V
16#1805	У IO-Link через комунікаційний порт 1 сталася помилка підключення		V
16#1806	Виникає помилка з'єднання в IO-Link через комунікаційний порт 2		V
16#1807	У IO-Link через комунікаційний порт 3 сталася помилка підключення		V
16#1808	Виникає помилка з'єднання в IO-Link через комунікаційний порт 4		V
16#1809	Помилка сканування пристрою, і сканування примусово зупинено		V

7.4.7 Коди помилок і світлодіодні індикатори для модуля AS00SCM як модуля зв'язку

Код помилки	опис	Статус світлодіодного індикатора ERROR	
		ON	Блимає
16#1605	Збій обладнання	V	
16#1606	Помилки налаштувань функціональної карти модуля зв'язку	V	
16#1802	Неправильні параметри		V
16#1803	Час очікування зв'язку		V
16#1804	Налаштування UD Link неправильне.		V

7.4.8 Коди помилок і світлодіодні індикатори для модуля AS00SCM як віддаленого модуля

Код помилки	опис	Статус світлодіодного індикатора ERROR			
		ON	Блимає	Повільно моргає	Швидко моргає
16#1301	Збій обладнання	V			
16#1302	Налаштування функціональної картки неправильне.	V			
16#1303	Джерела живлення 24 В постійного струму недостатньо, а потім відновлюється після ситуації низької напруги менше 10 мс.				V
16#1304	Включено більше восьми віддалених модулів праву сторону модуля ЦП.	V			
16#1500	Час очікування зв'язку віддаленого модуля		V		
16#1502	Неправильні параметри		V		
16#1503	Час очікування зв'язку модуля розширення		V		
16#1505	Фактичне розташування модулів розширення НЕ збігається з заданим.		V		
16#1506	Віддалений модуль було зупинено.			V	
16#1604	Час очікування зв'язку модуля розширення		V		

7.4.9 Коды помилок і світлодіодні індикатори для модуля AS01DNET (режим головного/підлеглого)

Код	Пояснення	Виправлення
0~63	Адреса вузла AS01DNET-A (у нормальній роботі)	--
80	AS01DNET-A знаходиться в стані STOP.	Переведіть ПЛК у режим RUN і почніть обмін даними введення/виведення
F0	Ідентифікатор вузла AS01DNET збігається з ідентифікатором іншого вузла або перевищує допустимий діапазон.	1. Переконайтеся, що адреса вузла AS01DNET унікальна. 2. Повторне живлення AS01DNET.
F1	У списку сканування немає налаштованого підлеглого пристрою.	Налаштуйте список сканування, а потім завантажте конфігурацію в AS01DNET.
F2	Занадто низька напруга робочого джерела живлення	Перевірте, чи справне джерело живлення для AS01DNET і ПЛК.
F3	AS01DNET переходить в тестовий режим	Перемикач функціональний перемикач IN1 із положення «Увімк.» у положення «Вимк.» та знову увімкніть AS01DNET-A.
F4	АВТОБУС ВИКЛ	1. Перевірте, чи мережевий кабель справний, а екранований кабель заземлений. 2. Перевірте, чи однакові швидкості передачі всіх вузлів у мережі. 3. Перевірте, чи початок і кінець мережевого кабелю підключено до кінцевого резистора 121 Ом. 4. Повторне живлення AS01DNET-A.
F5	Відсутнє живлення мережі	1. Перевірте, чи нормальний мережевий кабель. 2. Переконайтеся, що мережеве живлення в нормі.
F6	Внутрішня помилка; Помилка перевірки Flash або RAM	Якщо помилка все ще існує після повторного живлення, надішліть AS01DNET-A назад для ремонту.
F8	Помилка, виникла на заводі	Якщо помилка все ще існує після повторного живлення, надішліть AS01DNET-A назад для ремонту.
F9	Внутрішня помилка; Помилка доступу до EEPROM	Якщо помилка все ще існує після повторного живлення, надішліть AS01DNET-A назад для ремонту.
FA	Недійсні конфігураційні дані	1. Правильно налаштуйте мережу та повторно завантажте її на AS01DNET-A. 2. Перевірте, чи адреса вузла одного підлеглого пристрою в списку сканування збігається з адресою AS01DNET-A.
E0	Параметри ідентифікації, надіслані від підлеглого пристрою, не відповідають даним конфігурації.	1. Перевірте, чи є будь-які зміни в ідентифікаторі підлеглого вузла в мережі. 2. Перевірте, чи не замінено якийсь вузловий пристрій у мережі. 3. Переналаштуйте мережу.
E1	Розмір повернутих даних введення/виведення не відповідає розміру в списку сканування.	Знову налаштуйте розмір даних введення/виведення підлеглого пристрою, завантажте конфігурацію в AS01DNET-A та запустіть ПЛК.
E2	Підлеглий пристрій у списку сканування не існує або перебуває в автономному режимі, коли AS01DNET-A перебуває в режимі головного. З'єднання вводу/виводу між підлеглим AS01DNET-A і головним розривається, коли AS01DNET-A перебуває в підлеглому режимі.	1. Перевірте, чи відбулася зміна адреси підлеглого вузла. 2. Перевірте, чи кабель зв'язку від'єднано чи під'єднано нещільно. 3. Перевірте, чи довжина кабелю шини перевищує максимальну відстань передачі. Якщо так, система може бути нестабільною.
E3	AS01DNET-A не може передати дані.	1. Переконайтеся, що з'єднання між AS01DNET-A та мережею є нормальним.

Код	Пояснення	Виправлення
		2. Перевірте, чи швидкість передачі даних AS01DNET-A збігається з іншими вузлами мережі.
E4	Виявлено помилку в послідовності фрагментованих даних вводу-виводу від підлеглого пристрою.	Перевірте, чи ведений пристрій працює нормально.
E5	Підлеглий пристрій повертає помилку, коли AS01DNET-A намагається з'єднатися з ним.	Перевірте, чи ведений пристрій працює нормально.
E6	Розмір даних вводу/виводу, що повертаються від підлеглого, більший, ніж налаштований у списку сканування.	Переконайтеся, що розмір даних вводу-виводу підлеглого пристрою має бути таким самим, як налаштований у списку сканування.
E7	AS01DNET-A перевіряє MAC ID.	Якщо код відображається довго, виконайте усунення несправностей відповідно до наступних кроків. 1. Переконайтеся, що принаймні два вузли працюють нормально в мережі. 2. Перевірте, чи будь-який кінець мережі підключено до кінцевого резистора 121 Ом. 3. Перевірте, чи однакові швидкості передачі даних вузлових пристроїв у мережі. 4. Перевірте, чи кабель зв'язку справний, щоб уникнути його від'єднання або нещільного приєднання. 5. Перевірте, чи довжина кабелю шини перевищує максимальну відстань передачі. Якщо так, система може бути нестабільною. 6. Перевірте, чи заземлений екранований дрот мережевого кабелю. 7. Повторне живлення модуля сканера AS01DNET-A.

7.4.10 Коды помилок і світлодіодні індикатори для модуля AS01DNET (режим RTU)

Код	Індикація	Як боротися з
0~63	Ідентифікатор вузла модуля сканера (у стані RUN)	Виправлення не потрібно
F0	Ідентифікатор вузла повторюється або перевищує допустимий діапазон.	1. Переконайтеся, що ідентифікатор вузла AS01DNET (RTU) є унікальним у мережі DeviceNet у діапазоні 0~63. 2. Знову увімкніть його після зміни ідентифікатора вузла.
F1	Жоден модуль вводу/виводу не налаштований на AS01DNET (RTU) у Програмне забезпечення DeviceNet Builder.	Додайте модулі вводу/виводу в AS01DNET (RTU) у програмному забезпеченні DeviceNet Builder і завантажте конфігураційні дані в AS01DNET (RTU) після завершення налаштування.
F2	Робоча напруга AS01DNET (RTU) занадто низька.	Перевірте, чи блок живлення для AS01DNET (RTU) працює нормально.
F3	AS01DNET (RTU) переходить в тестовий режим.	Repower AS01DNET (RTU).
F4	AS01DNET (RTU) – стан шини вимкнено.	1. Перевірте, чи мережевий комунікаційний кабель справний і екранований кабель заземлений. 2. Переконайтеся, що швидкість передачі всіх вузлів мережі однакова. 3. Перевірте, чи обидва кінці мережі з'єднані за допомогою кінцевого резистора 120 Ом. 4. Повторно увімкніть модуль сканера.

Код	Індикація	Як боротися з
F5	Немає мережевого живлення для AS01DNET(RTU)	1. Перевірте, чи нормальний мережевий кабель. 2. Перевірте, чи нормальне живлення мережі. (Зовнішній мережевий блок живлення 24 В постійного струму підключається між червоним V+ і чорним V- AS01DNET (RTU).)
F6	Внутрішня помилка; Помилка у внутрішніх накопичувачах AS01DNET (RTU)	Поверніть виріб для ремонту, якщо помилка не зникає після повторного увімкнення.
F7	Внутрішня помилка; Помилка в блоках обміну даними AS01DNET (RTU)	Поверніть виріб для ремонту, якщо помилка не зникає після повторного увімкнення.
F8	Помилка виробництва	Поверніть виріб для ремонту, якщо помилка не зникає після повторного увімкнення.
F9	Внутрішня помилка; Помилка доступу до Flash AS01DNET (RTU)	Поверніть виріб для ремонту, якщо помилка не зникає після повторного увімкнення.
E4	Помилка модуля	Перевірте, чи не виникає помилка в модулях, підключених до правого боку AS01DNET (RTU); Перевірте, чи існує модуль; Перевірте, чи поточний модуль відповідає налаштованому в програмному забезпеченні; Перевірте, чи додано неналаштований модуль.
E7	Повторне виявлення ідентифікатора вузла	Якщо код з'явився протягом тривалого часу, вирішуйте проблеми за допомогою наведених нижче методів. 1. Переконайтеся, що в мережі принаймні два вузли працюють нормально. 2. Перевірте, чи обидва кінці мережі з'єднані за допомогою кінцевого резистора 121 Ом. 3. Переконайтеся, що швидкість передачі всіх вузлів мережі однакова. 4. Перевірте, чи немає в мережевому кабелі проблем, наприклад його від'єднання та ослаблення. 5. Перевірте, чи не перевищує довжина кабелю зв'язку шини максимальну відстань передачі. Якщо перевищено максимальну відстань передачі, стабільність системи не може бути забезпечена. 6. Перевірте, чи заземлений екранований провід кабелю мережевого зв'язку. 7. Знову увімкніть живлення AS01DNET (RTU).
E9	Кількість модулів введення/виведення, підключених до AS01DNET (RTU), перевищує максимум 8.	Перевірте, чи кількість модулів введення/виведення, підключених до AS01DNET (RTU), перевищує 8.
80	AS01DNET (RTU) знаходиться в стані STOP стан.	1. Перевірте, чи перемикач RUN/STOP ПЛК, підключеного до головного пристрою DeviceNet, переведено в положення RUN. 2. Перевірте, чи є значення контрольного слова AS01DNET (RTU). 1. Докладніше див. у розділі 11.5.4.3.4.
83	AS01DNET (RTU) конфігурація програмного забезпечення завантажувється.	Зачекайте, доки завершиться завантаження конфігураційних даних AS01DNET (RTU).

ПАМ'ЯТКА