
Версія: V1.1

Дата: 2021-05-15

Suzhou Veichi Electric Co., Ltd. надає клієнтам повний спектр послуг технічної підтримки. Ви можете звернутися до місцевого офісу або центру обслуговування клієнтів нашої компанії, або безпосередньо зв'язатися з нашою штаб-квартирою.

Suzhou Veichi Electric Co., Ltd.

Усі права захищено. Вміст посібника може бути змінено без попереднього

повідомлення. Suzhou Veichi Electric Co., Ltd.

Адреса: вул. Songjia, буд. 1000, вул. Guoxiang, Економічна та технологічна зона розвитку Wuzhong, Suzhou

Гаряча лінія: +86-0755-29685680

Веб-сайт: www.veichi.com

Передмова

- Призначені читачі

Цей посібник призначений для техніків з автоматизації, щоб допомогти їм опанувати програмування, проектування систем та введення в експлуатацію VEICHI програмованих логічних контролерів PLCs, надаючи довідкову інформацію як для нових, так і для досвідчених користувачів програмування PLC.

- Вміст

Цей посібник описує принципи програмування, програмні та апаратні ресурси програмування, підтримувані мови програмування та детальні інструкції для VC серії PLCs; надає технічну довідкову інформацію, таку як високошвидкісні входи/виходи та дані про зв'язок; а також представляє методи застосування всіх функцій із наданням прикладів використання.

- Розташування

Розділи в цьому посібнику організовані від загального до деталей. Вміст кожного розділу є незалежним від інших. Ви можете прочитати посібник повністю, щоб поступово зрозуміти серію VC PLCs у повному обсязі, або ви можете просто прочитати окремі розділи в будь-який час, щоб отримати технічну довідку.

- Читання посібника

1. Для читачів, не знайомих з PLCs

Якщо ви ніколи раніше не використовували PLCs, рекомендується спочатку прочитати розділи з 1 по 4. У цих розділах пояснюються основи PLCs, включаючи описи функцій, мови програмування, елементи програм, типи даних, режими адресації, визначення програмних елементів, функцію анотування програм та програмування, застосування головної програми та підпрограм.

2. Для читачів, знайомих з PLCs

Якщо ви знайомі з основними концепціями та інструментами програмування PLCs, ви можете безпосередньо прочитати розділ 5 "Основні інструкції" та розділ 6 "Прикладні інструкції". Ці два розділи описують усі інструкції серії VEICHI VC PLCs. Щоб зрозуміти, як використовувати діаграми послідовних функцій, високошвидкісні I/O, переривання та функції зв'язку, зверніться до розділів з 7 по 10. Щоб зрозуміти функції керування позиціонуванням, зверніться до розділу 11 "Посібник з функцій позиціонування". Крім того, додаток K "Таблиця покажчика інструкцій за порядком" та додаток L "Таблиця покажчика інструкцій за класифікацією" дозволяють шукати описи відповідних інструкцій за типом інструкції та алфавітним порядком англійських назв інструкцій

відповідно, що полегшує читання.

- Пов'язані документи та посилання

Ви також можете звернутися до наступних посібників:

- Посібник користувача серії VC1 PLC

Зміст

Передмова.....	1
Розділ 1 Огляд продукції.....	1
1.1 Вступ до продукту.....	1
1.1.1 Продуктивність виробу та технічні характеристики.....	1
1.1.2 Зовнішній вигляд VC1 та основних модулів серії VC1.....	5
1.2 Програмне забезпечення Auto Studio.....	5
1.2.1 Базова конфігурація.....	6
1.2.2 Процес встановлення програмного забезпечення Auto Studio.....	6
1.2.3 Інтерфейс запуску Auto Studio.....	6
1.2.4 Кабель програмування.....	7
1.3 Комунаційна функція.....	7
1.3.1 Комунаційна мережа Modbus.....	7
1.3.2 N:N комунаційна мережа.....	7
1.3.3 Вільний порт мережі.....	8
Розділ 2 Опис функцій.....	11
2.1 Ресурси та принципи програмування.....	12
2.1.1 Програмування ресурсів.....	12
2.1.2 PLC механізм керування (цикл сканування модель).....	14
2.1.3 Виконання програми користувача сторожовий таймер.....	15
2.1.4 Режим роботи постійного сканування mode.....	15
2.1.5 Завантаження та зберігання файлів користувача.....	15
2.1.6 Ініціалізація елемента.....	15
2.1.7 Збереження даних під час вимкнення живлення.....	15
2.1.8 Цифрова фільтрація для вхідних точок.....	16
2.1.9 Режим без акумулятора mode.....	16
2.1.10 Програма користувача захист.....	16
2.2 Конфігурація системи.....	17
2.2.1 Системний блок.....	17
2.2.2 Блоки даних.....	23
2.2.3 Глобальна змінна таблиця.....	24
2.3 Режим роботи та керування станом.....	24
2.3.1 Концепції станів RUN та STOP системи.....	24
2.3.2 Стан RUN та STOP стану.....	24
2.3.3 Встановлення стану вихідних точок у стані STOP стану.....	25
2.4 Введення системи в експлуатацію.....	25
2.4.1 Завантаження програми та вивантаження.....	25
2.4.2 Механізм звітування про помилки.....	26
2.4.3 Зміна користувацької програми в режимі онлайн.....	28
2.4.4 Очищення та форматування.....	28
2.4.5 Перевірка PLC інформації онлайн.....	29
2.4.6 Запис та примусове встановлення значень елементів, а також таблиця моніторингу елементів table	29
2.4.7 Генерація блоків даних з RAM.....	32
Розділ 3 Програмний елемент та дані.....	34
3.1 Тип та функція програмних елементів.....	35
3.1.1 Короткий огляд програмних елементів.....	35
3.1.2 М'який список елементів.....	35

3.1.3 точки входу та виходу.....	37
3.1.4 Допоміжні реле.....	37
3.1.5 Стан реле.....	38
3.1.6 Таймер.....	38
3.1.7 Лічильник.....	39
3.1.8 Регістр даних.....	39
3.1.9 Спеціальне допоміжне реле.....	40
3.1.10 Спеціальний регістр даних регістр.....	40
3.1.11 Індикація реєстру.....	41
3.1.12 Локальне допоміжне реле.....	41
3.1.13 Локальний регістр даних регістр.....	41
3.1.14 Режим комбінованої бітової адресації (Kn адресація режим).....	41
3.1.15 Режим індикації (Z режим адресації).....	42
3.1.16 Комбінована індикація бітових рядків режим.....	43
3.1.17 Зберігання та адресація 32-бітних даних у D, R та V елементах.....	44
3.2 Дані.....	44
3.2.1 Тип даних.....	44
3.2.2 Кореляція між елементами та типами даних.....	44
3.2.3 Константа.....	45
Розділ 4 Концепція програмування.....	46
4.1 Вступ до мов програмування.....	47
4.1.1 LAD.....	47
4.1.2 IL.....	48
4.1.3 SFC.....	48
4.2 Програмування Елементів.....	49
4.2.1 Програма.....	49
4.2.2 Системний блок.....	50
4.2.3 Блок даних.....	50
4.3 Блоковий коментар та коментар змінної програми.....	50
4.3.1 Блок коментаря.....	50
4.3.2 Змінна Коментар.....	51
4.4 Підпрограма.....	52
4.4.1 Концепція.....	52
4.4.2 Примітка щодо використання SBRs.....	52
4.4.3 Визначення змінної SBR таблиця.....	53
4.4.4 SBR параметр передачі.....	54
4.4.5 SBR екземпляр застосування.....	54
4.5 Загальна інформація інструкцій.....	56
4.5.1 операнди.....	56
4.5.2 Прапорець біта.....	56
4.5.3 Обмеження щодо використання інструкцій.....	56
Розділ 5 Основні інструкції.....	57
5.1 Інструкції контактної логіки.....	59
5.1.1 LD: NO контакт інструкція.....	59
5.1.2 LDI: NC контакт інструкція.....	59
5.1.3 AND: NO контакт та інструкція.....	59
5.1.4 ANI: NC контакт та інструкція.....	60
5.1.5 OR: NO контакт або інструкція.....	60

5.1.6 ORI: NO контакт або інструкція.....	61
5.1.7 OUT: Вихід котушки інструкція.....	61
5.1.8 ANB: Блок потоку енергії та інструкція.....	61
5.1.9 ORB: Блок потоку енергії або інструкція.....	62
5.1.10 MPS: Інструкція з виштовхування потоку енергії на виході інструкція.....	62
5.1.11 MRD: Інструкція для зчитування значення верхівки стека потоку вихідної енергії.....	63
5.1.12 MPP: Інструкція виштовхування стека потоку енергії виходу.....	63
5.1.13 EU: Інструкція виявлення переднього фронту сигналу.....	64
5.1.14 ED: Інструкція виявлення спадного фронту.....	64
5.1.15 LDP: Передній фронт контакту.....	65
5.1.16 LDF: Спадаючий фронт контакту.....	65
5.1.17 ANDP: Передній фронт контакту.....	66
5.1.18 ANDF: Задній фронт контакту.....	66
5.1.19 ORP: Передній фронт контакту.....	67
5.1.20 ORF: Задній фронт контакту.....	67
5.1.21 PLP: Вихід за переднім фронтом інструкція.....	68
5.1.22 PLF: Вихід за спадним фронтом інструкція.....	68
5.1.23 INV: Інструкція заперечення потоку енергії.....	69
5.1.24 SET: Інструкція щодо комплекту котушок.....	69
5.1.25 RST: інструкція скидання котушки	69
5.1.26 NOP: Відсутня операція інструкція.....	70
5.2 Основні інструкції з керування.....	70
5.2.1 MC: Основна інструкція з керування.....	70
5.2.2 MCR: Інструкція скидання головного керування інструкція.....	70
5.3 SFC інструкції.....	71
5.3.1 STL: SFC стан інструкція із завантаження.....	71
5.3.2 SET Sxx: SFC перехід стану інструкція.....	72
5.3.3 OUT Sxx: SFC стан інструкція переходу.....	72
5.3.4 RST Sxx: SFC скидання стану інструкція.....	72
5.3.5 RET: SFC кінець сегмента програми інструкція.....	73
5.4 Інструкції таймера.....	73
5.4.1 TON: Час затримки увімкнення інструкція.....	73
5.4.2 TONR: Інструкція затримки увімкнення типу пам'яті інструкція.....	73
5.4.3 TOF: Інструкція щодо затримки вимкнення.....	74
5.4.4 TMON: Неперезапускальна моностабільна часова інструкція.....	75
5.5 Інструкції лічильника.....	76
5.5.1 CTU: 16-бітний інкрементний лічильник інструкцій.....	76
5.5.2 CTR: 16-бітна інструкція циклічного підрахунку.....	76
5.5.3 DCNT: 32-розрядна інструкція інкременту та декременту лічильника.....	77
Розділ 6 інструкції із застосування.....	79
6.1 Інструкції керування потоком програми.....	85
6.1.1 FOR: Інструкція циклу.....	85
6.1.2 NEXT: Інструкція повернення циклу інструкція.....	85
6.1.3 LBL: Визначення мітки переходу інструкція.....	86
6.1.4 CJ: Умовна команда переходу інструкція.....	88
6.1.5 CFEND: Інструкція для умовного повернення основної програми користувача.....	88
6.1.6 WDT: Інструкція для скидання сторожового таймера користувача.....	89
6.1.7 EI: Увімкнути інструкцію переривання.....	89
6.1.8 DI: Вимкнути переривання інструкції.....	89

6.1.9 CIRET: Інструкція для умовного повернення переривання користувача програми.....	89
6.1.10 STOP: Інструкція для зупинки програми користувача.....	89
6.1.11 CALL: Інструкція для виклику підпрограми користувача.....	90
6.1.12 CSRET: Інструкція для умовного повернення підпрограми користувача.....	90
6.2 Інструкції з передачі даних інструкції.....	91
6.2.1 MOV: Інструкція з передачі даних word.....	91
6.2.2 DMOV: Інструкція передачі даних подвійного слова інструкція.....	91
6.2.3 RMOV: Інструкція передачі даних з рухомою комою інструкція.....	92
6.2.4 BMOV: Інструкція передачі даних блоком інструкція.....	92
6.2.5 FMOV: Інструкція заповнення блоку даних instruction.....	92
6.2.6 DFMOV: Інструкція заповнення подвійного слова блоку даних instruction.....	93
6.2.7 SWAP: MSB/LSB замінити інструкція.....	93
6.2.8 XCH: Інструкція з перестановки слів.....	94
6.2.9 DXCH: Інструкція обміну подвійними словами.....	94
6.2.10 PUSH: Інструкція передачі даних.....	94
6.2.11 FIFO: Інструкція першим прийшов — першим обслужений.....	95
6.2.12 LIFO: Інструкція останній прийшов — перший пішов.....	96
6.2.13 WSFR: Інструкція зсуву рядка символів праворуч instruction.....	96
6.2.14 WSFL: Інструкція зсуву рядка символів вліво instruction.....	97
6.3 Інструкції цілочисельних арифметичних операцій.....	98
6.3.1 ADD: Інструкція додавання цілих чисел instruction.....	98
6.3.2 SUB: Інструкція віднімання цілих чисел instruction.....	99
6.3.3 MUL: Інструкція множення цілих чисел instruction.....	99
6.3.4 DIV: Інструкція цілочисельного ділення instruction.....	99
6.3.5 SQT: Інструкція для добування квадратного кореня з SQT цілого числа.....	100
6.3.6 INC: Ціле число плюс один інструкція.....	100
6.3.7 DEC: Ціле число мінус один інструкція.....	101
6.3.8 VABS: Інструкція для отримання абсолютного значення цілого числа.....	101
6.3.9 NEG: Інструкція заперечення цілого числа instruction.....	102
6.3.10 DADD: Додавання довгих цілих чисел інструкція.....	102
6.3.11 DSUB: Віднімання довгих цілих чисел інструкція.....	102
6.3.12 DMUL: Інструкція множення довгих цілих чисел instruction.....	103
6.3.13 DDIV: Ділення довгих цілих чисел інструкція.....	103
6.3.14 DSQT: Інструкція для добування квадратного кореня з довгого цілого числа.....	104
6.3.15 DINC: Довге ціле число плюс один інструкція.....	104
6.3.16 DDEC: Довге ціле число мінус один інструкція.....	105
6.3.17 DVABS: Інструкція для отримання абсолютного значення цілого числа типу long.....	105
6.3.18 DNEG: Інструкція заперечення довгого цілого числа.....	106
6.3.19 SUM: Інструкція накопичення цілих чисел.....	106
6.3.20 DSUM: Інструкція накопичення довгого цілого числа instruction.....	108
6.4 Інструкції операцій з плаваючою комою.....	108
6.4.1 RADD: Інструкція додавання чисел з рухомою комою інструкція.....	108
6.4.2 RSUB: Інструкція віднімання чисел з рухомою комою instruction.....	109
6.4.3 RMUL: Інструкція множення чисел з рухомою комою інструкція.....	109
6.4.4 RDIV: Інструкція ділення чисел з рухомою комою інструкція.....	110
6.4.5 RSQT: Інструкція для добування квадратного кореня з числа з рухомою комою.....	110
6.4.6 RVABS: Інструкція для отримання абсолютного значення числа з рухомою комою числа.....	111
6.4.7 RNEG: Інструкція заперечення числа з рухомою комою instruction.....	111
6.4.8 SIN: Інструкція для отримання SIN числа з рухомою комою числа.....	111

6.4.9 COS: Інструкція для отримання COS числа з плаваючою комою.....	112
6.4.10 TAN: Інструкція для отримання TAN числа з рухомою комою number.....	112
6.4.11 POWER: Інструкція для піднесення до степеня числа з плаваючою комою числа.....	113
~~6.4.12 LN: Інструкція з отримання натурального логарифма числа з плаваючою комою.....~~	~~113~~
6.4.13 EXP: Інструкція для отримання степеня натурального числа для числа з рухомою комою.....	114
6.4.14 RSUM: Інструкція накопичення чисел з рухомою комою instruction.....	114
6.4.15 ASIN: Інструкція для отримання ASIN числа з плаваючою комою number.....	115
6.4.16 ACOS: Інструкція для отримання ACOS числа з рухомою комою числа.....	115
6.4.17 ATAN: Інструкція для отримання ATAN числа з рухомою комою числа.....	115
6.4.18 LOG: Інструкція для отримання десятичного логарифма числа з рухомою комою.....	116
6.4.19 RAD: Інструкція для перетворення кута з градусів у радіани conversion.....	116
6.4.20 DEG: Інструкція для перетворення кута в радіанах з плаваючою комою conversion.....	116
6.5 Інструкції перетворення значень.....	117
6.5.1 DTI: Інструкція щодо перетворення довгого цілого числа на ціле число.....	117
6.5.2 ITD: Інструкція для перетворення цілого числа на довге ціле число.....	117
6.5.3 FLT: Інструкція для перетворення цілого числа у число з плаваючою комою.....	118
6.5.4 DFLT: Інструкція для перетворення довгого цілого числа у число з плаваючою комою.....	118
6.5.5 INT: Інструкція для перетворення числа з рухомою комою на ціле число.....	118
6.5.6 DINT: Інструкція для перетворення числа з рухомою комою у ціле число типу long.....	119
6.5.7 BCD: Інструкція з перетворення слова у 16-бітний BCD код.....	119
6.5.8 DBCD: Інструкція для перетворення подвійного слова у 32-бітний BCD code.....	120
6.5.9 BIN: Інструкція для перетворення 16-бітного BCD коду у слово.....	120
6.5.10 DBIN: Інструкція для перетворення 32-бітного BCD коду на подвійне слово.....	121
6.5.11 GRY: Інструкція з перетворення слова у 16-бітний код Грея.....	121
6.5.12 DGRY: Інструкція для перетворення подвійного слова у 32-бітний код Грея code.....	121
6.5.13 GBIN: Інструкція з перетворення 16-бітового коду Грея у слово.....	122
6.5.14 DGBIN: Інструкція для перетворення 32-бітного коду Грея у подвійне слово.....	122
6.5.15 SEG: Інструкція з перетворення слова на 7-сегментний код.....	123
6.5.16 ASC: ASCII перетворення коду інструкція.....	123
6.5.17 ITA: Інструкція з перетворення 16-бітових шістнадцяткових даних у ASCII код.....	124
6.5.18 ATI: Інструкція для перетворення коду ASCII у 16-бітні шістнадцяткові дані.....	124
6.5.19 LCNV: Інструкція з перетворення проєкту instruction.....	125
6.5.20 RLCNV: Інструкція з конвертації проєкту з рухомою комою інструкція.....	126
6.6 Логічна операція над словами інструкції.....	127
6.6.1 WAND: Слово AND інструкція.....	127
6.6.2 WOR: INT OR інструкція.....	129
6.6.3 WXOR: Слово XOR інструкція.....	129
6.6.4 WINV: Слово INV інструкція.....	130
6.6.5 DWAND: Подвійне слово AND інструкція.....	130
6.6.6 DWOR: Подвійне слово OR інструкція.....	130
6.6.7 DWXOR: Подвійне слово XOR інструкція.....	131
6.6.8 DWINV: Подвійне заперечення слова інструкція.....	131
6.7 Інструкції побітового зсуву та циклічного зсуву.....	132
6.7.1 ROR: 16-бітна циклічна операція зсуву вправо інструкція.....	132
6.7.2 ROL: 16-бітна циклічна команда зсуву вліво інструкція.....	132
6.7.3 RCR: Інструкція для 16-бітного циклічного зсуву праворуч через прапор перенесення біт.....	133
6.7.4 RCL: Інструкція для 16-бітного циклічного зсуву вліво через прапор переносу біт.....	134
6.7.5 DROR: 32-бітна циклічна операція зсуву праворуч інструкція.....	134
6.7.6 DROL: інструкція циклічного зсуву вліво на 32 біти інструкція.....	135

chastotnik.ua

6.11.5 ABSD: Абсолютне керування кулачком інструкція.....	181
6.11.6 DABSD: Інструкція з абсолютного керування кулачком подвійного слова instruction.....	183
6.11.7 ALT: Альтернативна інструкція виводу інструкція.....	184
6.12 Інструкції зв'язку.....	184
6.12.1 MODBUS: Інструкція з обміну даними головної станції інструкція.....	184
6.12.2 XMT: Інструкція передавання через вільний порт instruction.....	185
~~6.12.3 RCV: Приймання інструкції через вільний порт інструкція.....~~	~~186~~
6.12.4 MODRW: Інструкція з читання/запису Modbus інструкція.....	187
6.13 Перевірте інструкції.....	191
6.13.1 CCITT: Перевірте інструкцію.....	191
6.13.2 CRC16: Перевірте інструкцію.....	191
6.13.3 LRC: Перевірте інструкцію.....	192
6.14 Розширені інструкції бітової обробки.....	193
6.14.1 ZRST: Інструкція зі скидання бітів у 0 у пакетному режимі.....	193
6.14.2 ZSET: Інструкція для скидання бітів у пакеті.....	193
6.14.3 DECO: Декодування інструкції.....	193
6.14.4 ENCO: Інструкція з кодування інструкція.....	194
6.14.5 BITS: Інструкція для підрахунку бітів у <A>слові.....	194
6.14.6 DBITS: Інструкція для підрахунку бітів у подвійному слові.....	195
6.14.7 BON: Інструкція для визначення біта у слові.....	195
6.15 Контактні інструкції.....	195
6.15.1 BLD: Контакт слова біт LD інструкція.....	195
6.15.2 BLDI: Контакт біта слова LDI інструкція.....	196
6.15.3 BAND: Контакт біта слова AND інструкція.....	196
6.15.4 BANI: Контакт біта слова ANI інструкція.....	197
6.15.5 BOR: Контакт біта слова OR інструкція.....	197
6.15.6 BORI: Контакт біта слова ORI інструкція.....	198
6.15.7 BOUT: Інструкція виходу бітової котушки слова.....	198
6.15.8 BSET: Бітова котушка слова інструкція встановлення.....	198
6.15.9 BRST: Інструкція скидання бітової котушки слова instruction.....	199
6.16 Інструкції контакту порівняння.....	199
6.16.1 LD (=, <, >, <>, >=, <=): Порівняння цілих чисел LD*інструкція.....	199
6.16.2 AND (=, <, >, <>, >=, <=): Порівняння цілих чисел AND *інструкція.....	200
6.16.3 OR (=, <, >, <>, >=, <=): Порівняння цілих чисел OR*інструкція.....	201
6.16.4 LDD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел LDD*інструкція.....	202
6.16.5 ANDD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел ANDD*інструкція.....	203
6.16.6 ORD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел ORD*інструкція.....	204
6.16.7 LDR: Інструкція порівняння чисел з рухомою комою instruction.....	205
6.16.8 ANDR: Інструкція порівняння чисел з рухомою комою instruction.....	206
6.16.9 ORR: Інструкція порівняння чисел з рухомою комою instruction.....	207
6.16.10 CMP: Інструкція для встановлення порівняння цілих чисел на ON.....	209
6.16.11 LCMP: Інструкція для встановлення порівняння довгих цілих чисел на ON.....	209
6.16.12 RCMP: Інструкція для встановлення порівняння чисел з рухомою комою на ON.....	210
6.17 Інструкції з масової обробки даних.....	210
6.17.1 BKADD: Інструкція з операції масового додавання даних інструкція.....	210
6.17.2 BKSUB: Операція віднімання масиву даних інструкція.....	211
6.17.3 BKCMP=>,<,<>,<=,>=: Інструкція порівняння масиву даних instruction.....	211
6.18 Технічний паспорт інструкції.....	212
6.18.1 LIMIT: Інструкція з керування верхньою/нижньою межею інструкція.....	212

6.18.2 DBAND: Інструкція з керування зоною нечутливості instruction.....	213
6.18.3 ZONE: Інструкція з керування зоною.....	213
6.18.4 SCL: Інструкція з налаштування координат.....	214
6.18.5 SER: Інструкція з пошуку даних інструкція.....	215
6.19 Рядок символів інструкції.....	216
6.19.1 STRADD: Комбінація рядків інструкція.....	216
6.19.2 STRLEN: Інструкція для визначення довжини рядка.....	217
6.19.3 STRRIGHT: Інструкція для зчитування рядка зправа.....	217
6.19.4 STRLEFT: Інструкція для зчитування рядка з лівого боку.....	218
6.19.5 STRMIDR: Інструкція для зчитування будь-яких символів рядка.....	219
6.19.6 STRMIDW: Інструкція щодо заміни будь-яких символів рядка.....	219
6.19.7 STRINSTR: Пошук рядка інструкція.....	220
6.19.8 STRMOV: Інструкція з передачі рядка instruction.....	221
6.20 Інструкції з реєстрації файлів розширення instructions.....	221
6.20.1 LOADR: Інструкція для зчитування даних з файлу розширення регістр.....	221
6.20.2 SAVER: Інструкція для запису даних у регістр файла розширення.....	222
6.20.3 INITR: Інструкція з ініціалізації регістра розширення register.....	223
6.20.4 LOGR: Інструкція для входу в систему на розширенні реєстр.....	224
6.20.5 ITER: Інструкція з ініціалізації файлу розширення реєстру.....	225
6.21 Інструкції з позиціонування.....	227
6.21.1 ZRN: Інструкція повернення до нуля інструкція.....	227
6.21.2 PLSV: Інструкція щодо виходу імпульсів зі змінною швидкістю instruction.....	228
6.21.3 DRVI: Інструкція з керування відносним положенням інструкція.....	229
6.21.4 DRVA: Інструкція керування абсолютним положенням інструкція.....	230
6.21.5 DSZR: Інструкція для повернення до нуля з DOG.....	231
6.21.6 DVIT: Інструкція переривання позиціонування instruction.....	232
6.21.7 STOPDV: Зупинка імпульсного виходу інструкція.....	233
6.21.8 LIN: Інструкція лінійної інтерполяції траєкторії instruction.....	234
6.21.9 CW: Траєкторія дуги за годинниковою стрілкою інтерполяція.....	237
6.21.10 CCW: Інструкція інтерполяції дугової траєкторії проти годинникової стрілки instruction.....	238
6.21.11 MOVELINK: Інструкція синхронного керування інструкція.....	241
6.21.12 GEARBOX: Інструкція з електронного редуктора instruction.....	243
6.22 Інструкції з обробки даних.....	244
6.22.1 MEAN: Означає інструкцію.....	244
6.22.2 WTOP: Інструкція з розділення даних у байтових одиницях instruction.....	244
6.22.3 BTOW: Інструкція комбінування даних байтових одиниць instruction.....	245
6.22.4 UNI: Інструкція для об'єднання 4bits 16-бітних даних.....	246
6.22.5 DIS: Інструкція для розділення 4bitsof 16-бітових даних.....	247
6.22.6 ANS: Інструкція встановлення сигналу тривоги instruction.....	248
6.22.7 ANR: Інструкція скидання сигналу тривоги інструкція.....	249
6.23 Інші інструкції.....	249
6.23.1 RND: Інструкція для генерування випадкових чисел.....	249
6.23.2 DUTY: Інструкція для генерування часових імпульсів.....	250
Розділ 7 SFCНаставник.....	251
7.1 Вступ до SFC.....	252
7.1.1 Що таке SFC.....	252
7.1.2 Що таке SFC серії VC PLCs.....	252
7.1.3 Базова концепція SFC.....	252

7.1.4 Символи програмування та їхнє використання.....	252
7.1.5 SFC структура програми.....	253
7.1.6 Виконання SFC програми.....	257
7.2 Зв'язок між SFC та LAD.....	258
7.2.1 STL інструкція та етап станів.....	258
7.2.2 SET інструкція.....	259
7.2.3 RET інструкція та SFC програмний сегмент.....	259
7.2.4 OUT інструкція та RST інструкція.....	259
7.2.5 SFC гілка вибору, паралельна гілка та об'єднання.....	259
7.3 Як програмувати за допомогою SFC.....	259
7.4 Примітки в SFC програмуванні.....	260
7.4.1 Типові помилки програмування.....	260
7.4.2 Програмування хитрощів.....	263
7.5 Приклади SFC програмування.....	264
7.5.1 Проста послідовна структура.....	265
7.5.2 Гілка вибору структури.....	266
7.5.3 Паралельна гілка структури.....	269
Розділ 8 Посібник з експлуатації для високошвидкісного входу функція.....	274
8.1 Високошвидкісний лічильник.....	275
8.1.1 Конфігурація.....	275
8.1.2 Зв'язок між високошвидкісним лічильником та SM допоміжним реле.....	276
8.1.3 Як використовувати високошвидкісні лічильники.....	278
8.1.4 Примітки щодо високошвидкісних лічильників серії VC1.....	281
8.2 Функція захоплення зовнішнього імпульсу функція.....	282
8.3 Примітки щодо застосування високошвидкісного входу.....	282
Розділ 9 Використання переривань.....	283
9.1 Перервати програму.....	284
9.2 Обробка переривання події.....	286
9.3 Використання переривання за таймером interrupt.....	286
9.4 Використання зовнішніх переривань.....	289
9.5 Використання високошвидкісного лічильника переривання.....	290
9.6 Використовуючи РТО виведіть завершення переривання.....	291
9.7 Використання переривання на основі послідовного порту.....	292
Розділ 10 Комунікаційна функція посібник.....	295
10.1 Комунікаційні ресурси.....	296
10.2 Протокол зв'язку порту програмування protocol.....	296
10.3 Протокол вільного порту зв'язку.....	296
10.3.1 Вступ.....	296
10.3.2 Налаштування параметрів вільного порту.....	296
10.3.3 Інструкція вільного порту.....	298
10.4 Протокол зв'язку Modbus protocol.....	299
10.4.1 Вступ.....	299
10.4.2 Характеристики посилок.....	299
10.4.3 RTU режим передачі.....	299
10.4.4 Підтримувані коди функцій Modbus.....	300
10.4.5 Режим адресації PLC елементів.....	300
10.4.6 Ведучий пристрій Modbus станція.....	301
10.4.7 Читання та запис елементів.....	301

10.4.8 Обробка подвійного слова елемента.....	302
10.4.9 Обробка LONG INT даних.....	302
10.4.10 Діагностичний код функції.....	302
10.4.11 Код помилки.....	303
10.4.12 Налаштування параметра Modbus налаштування.....	303
10.4.13 Інструкція Modbus інструкція.....	304
10.5 N:N протокол зв'язку протокол.....	306
10.5.1 N:N вступ.....	306
10.5.2 N:N форма передачі даних мережею form.....	306
10.5.3 N:N мережева структура.....	307
10.5.4 N:N режим оновлення mode.....	308
10.5.5 Покращений режим оновлення.....	312
10.5.6 N:N налаштування параметра параметра.....	313
10.6 Декілька стратегій керування.....	315
10.6.1 Визначення головної станції.....	315
10.6.2 Максимальна кількість інспекційних станцій.....	315
10.6.3 Мультимайстер та підпорядкований (M: N).....	315
10.6.4 Приклади використання N:N.....	315
Розділ 11 Посібник з функції позиціонування guide.....	316
11.1 Система позиційного керування system.....	317
11.1.1 Абсолютна позиція системи.....	317
11.1.2 Система позиційного керування system.....	318
11.1.3 Процес позиціонування керування.....	319
11.2 Огляд функцій позиціонування серії VC PLC.....	319
11.3 Примітка щодо використання інструкцій позиціонування.....	322
11.4 Спеціальні елементи, пов'язані з інструкціями позиціонування.....	323
11.4.1 Вихідні осі серії VC PLC.....	323
11.4.2 Вихідні осі серії VC1 PLC.....	324
11.5 Екземпляр застосунку.....	324
11.5.1 Метод конфігурації PLS оболонки інструкції.....	324
Додаток А Спеціальне допоміжне реле.....	328
1. PLC робочий стан прапорець.....	328
2. Біт роботи годинника.....	328
3. Помилка виконання програми користувача error.....	329
4. Керування перериванням.....	329
5. Периферійна інструкція.....	331
6. Operation прапорець біт.....	331
7. DHST/DHSP Завершення порівняння таблиці прапорець.....	331
8. ASCII інструкція перетворення коду прапорець.....	331
9. Біт моніторингу захоплення імпульсів bit.....	331
10. Чотиричастотний.....	332
11. Вільний порт (PORT0).....	332
12. Вільний порт (PORT1).....	332
13. Розширений вільний порт (PORT2).....	333
14. N:N зв'язок.....	334
15. Помилка системної шини прапорець.....	334
16. Прапорець помилки годинника реального часу.....	335
17. Увімкнути прапорець інтегрованого аналогового каналу.....	335

18. Напрямок рахунку та моніторинг високошвидкісних лічильників.....	335
19. Високошвидкісний вихід та інструкція позиціонування.....	336
1) Прапорець кореляції Y0 біт.....	336
2) Прапорець кореляції Y1 біт.....	337
3) Прапорець кореляції Y2 біт.....	338
4) Прапорець кореляції Y3 біт.....	338
5) Прапорець кореляції Y4 біт.....	339
6) Y5 Прапорець кореляції біт.....	340
~~7) Прапорець кореляції Y6 біт.....~~	~~341~~
8) Прапорець кореляції Y7 біт.....	341
20. Вихід за таймером інструкція.....	342
21. Сигнал тривоги.....	342
22. CANopen інструкція.....	342
Додаток В Спеціальний регістр даних регістр.....	344
1. PLC робочий стан дані.....	344
2. Код помилки операції FIFO зона.....	344
3. FROM/TO помилка.....	345
4. Сканувати час.....	345
5. Константа фільтрації вхідного сигналу налаштування.....	345
6. Циклічне переривання за таймеромцикл.....	346
7. Годинник реального часу clock.....	346
8. Використання DHSP та DHST інструкцій.....	346
9. Керування та стан приймання через вільний порт (PORT0).....	347
10. Керування прийомом через вільний порт та стан (PORT1).....	348
11. Керування прийомом через вільний порт та стан (PORT2).....	350
12. Високошвидкісний імпульсний вихід та позиціонування.....	352
1) Кореляція Y0 регістр.....	352
2) Кореляція Y1 регістр.....	353
3) Кореляція Y2 регістр.....	354
4) Кореляція Y3 регістр.....	355
5) Y4 Кореляція регістру.....	355
6) Кореляція Y5 регістр.....	356
7) Y6 Кореляція регістру.....	357
8) Y7 Кореляція регістр.....	358
13. Вихід за часом інструкція.....	358
14. Сигнал тривоги інструкція.....	359
Додаток С Зарезервовані елементи.....	360
Додаток D Коды помилок зв'язку Modbus.....	361
Додаток Е Коды системних помилок кодів.....	362
Додаток F Протоколи зв'язку Modbus протоколи.....	364
1. Огляд протоколу зв'язку Modbus protocol.....	364
2. Підтримувані коди функцій Modbus та адресація елементів режим.....	364
3. Код функції Modbus опис.....	366
3.1 Зчитування стану котушки (0x01).....	366
3.2 (0x02)Зчитування стану дискретного входу (0x02).....	366
3.3 Читання регістрів зберігання (0x03).....	366
3.4 Примусовий запис (запис) окремої котушки (0x05).....	367

3.5 Попереднє встановлення (запис) одного регістра (0x06).....	367
3.6 Повернення діагностичної перевірки (0x08).....	367
3.7 Примусовий запис (запис) декількох котушок(0x0F).....	370
3.8 Попереднє встановлення (запис) декількох регістрів (0x10 шістнадцятковий).....	370
3.9 Кадр відповіді про помилку (0x80+код функції).....	370
3.10 Примітка.....	370
4. Приклад керування зв'язком Modbus.....	371
5. Опис трансляції.....	372
~~Додаток С ASCII-таблиця кодування символів коду.....~~	~~374~~
Додаток Н Показчик порядку інструкцій таблиця.....	375
Додаток І Показчик класифікації інструкцій таблиця.....	385

Розділ 1 Огляд продукції

У цьому розділі стисло описано елементи продукту, програмну платформу для програмування, а також конфігурацію мережі та застосування VC1 серії PLCs.

Передмова.....	1
Розділ 1 Огляд продукції.....	1
1.1 Вступ до продукту.....	1
1.1.1 Продуктивність виробу та технічні характеристики.....	1
1.1.2 Зовнішній вигляд VC1 та основних модулів серії VC1.....	5
1.2 Програмне забезпечення Auto Studio.....	5
1.2.1 Базова конфігурація.....	6
1.2.2 Процес встановлення програмного забезпечення Auto Studio.....	6
1.2.3 Інтерфейс запуску Auto Studio.....	6
1.2.4 Кабель програмування.....	7
1.3 Комунікаційна функція.....	7
1.3.1 Комунікаційна мережа Modbus.....	7
1.3.2 N:N комунікаційна мережа.....	7
1.3.3 Вільний порт мережі.....	8

1.1 Вступ до продукту

Серії VC — це інтегровані PLCs з вбудованими високопродуктивними мікропроцесорами та основними системами обчислювального керування, що об'єднують точки вводу та виводу, шину модулів розширення тощо. У ці серії також включено модулі розширення I/O та спеціальні модулі. Основний модуль інтегрує від 2 до 3 комунікаційних портів. Серії VC2/3/5 PLCs можуть бути безпосередньо підключені до мережі, тоді як основні модулі інших серій PLC можуть бути підключені до мережі польової шини через модулі розширення польової шини. I/O, налаштовані на основних модулях, також включають канали високошвидкісного підрахунку та високошвидкісного імпульсного виводу, які можна використовувати для точного позиціонування. Вони оснащені великою кількістю вбудованих ресурсів програмування, використовують три стандартизовані мови програмування та можуть виконувати введення в експлуатацію та моніторинг за допомогою потужного програмного забезпечення для програмування Auto Studio. Крім того, для користувацьких програм передбачено оптимальні механізми захисту безпеки.

1.1.1 Продуктивність виробу та технічні характеристики

Таблиця 1-1 Продуктивність та технічні характеристики PLC основних модулів

Ім'я		VC 5 (Планування)	VC3 (Планування)	VC2 (Планування)	VC1	VC1S
I/O	Цифровий I/O точка	16 вхід./16 вихід.	16 вх/16 вих 32 входи/32 виходи	16 вх/16 вих 32 входи/32 виходи	8 входів/6 виходів 12 вх./8 вих. 14 вхід/10 вихід 16 вх/14 вих 24 входи/16 виходів 36 вх/24 вих 16 вх./14 вих./ 2 AI/1 AO	8 входів/6 виходів 12 вхід/8 вихід 14 вх/10 вих 16 вх/14 вих 24 входи/16 виходів 36 вх/24 вих 16 вх./14 вих./ 2 AI/1 AO
	Макс. кількість яких логічні I/O точки	512	512	128	128	60
	Макс. кількість яких спеціальні модулі	8	8	8	8	
	Високошвидкісний імпульсний вихід		8×200 кГц (лише для транзисторного виходу тільки)	3×100 кГц (тільки для транзисторного виходу)	3×100 кГц (тільки для транзисторного виходу)	2×100 кГц (тільки для виходу транзистора)
	Однофазний рахунковий канал	8×200 кГц		2×50 кГц + 6×10 кГц		6×10 кГц
	Двофазний рахунковий канал	4×200 кГц		1×30 кГц + 3×5 кГц		1×5 кГц
	Макс. частота сума високого лічильники швидкості	1600 кГц	1600 кГц	80 кГц	60 кГц	60 кГц
	Цифрове фільтрування	Застосування цифрової фільтрації для X0 до X7. Діапазон постійної фільтрації вхідного сигналу: 0–60	Застосування цифрової фільтрації для X0 до X7. Діапазон постійної фільтрації вхідного сигналу: 0–60	Застосування цифрової фільтрації для X0–X5. Діапазон константи фільтрації вхідного сигналу: 0–	Застосування цифрової фільтрації для X0–X7. Діапазон сталої фільтрації вхідного сигналу: 0–60 мс	Застосування цифрової фільтрації для X0–X7. Діапазон сталої фільтрації вхідного

			мс	мс	60000 мкс		сигналу: 0–60 мс
Макс. струм реле виходу реле виходу у точку	Резистивне навантаження	2 A/ 1 точка 8 A/ 4-точковий груповий спільний вивід 8 A/ 8-точковий груповий спільний вивід					
	Індуктивний навантаження	220 V AC, 80 VA					
	Лампове навантаження	220 V AC, 100 W					
	Макс. струм при той	Резистивне навантаження	Точка виходу: 0.5 A/1 точка Інші: 0.5 A/1 точка; 0.8 A/4 точки; 1.6 A/8 точок Вище 8 точок, загальний струм збільшується на 0,1 A для кожної додаткової точки				

Ім'я			VC5(Планування)	VC3 (Планування)	VC2 (Планування)	VC1	VC1S
транзисторний вихідний контакт	Індуктивне навантаження		12 W/24 V DC	Y0–Y7: 7.2 W/24 V DC Інші: 12 W/24 V DC	Y0, Y1,Y2: 7.2 W/24 V DC Інші: 12 W/24 V DC		Y0, Y1: 7.2 W/24 V DC Інші: 12 W/24 V DC
	Лампове навантаження		1.5 W/24 V DC	Y0–Y7: 0.9 W/24 V DC Інше: 1.5 W/24 V DC	Y0, Y1,Y2: 0.9 W/24 V DC Інші: 1.5 W/24 V DC		Y0, Y1: 0.9 W/24 V DC Інші: 1.5 W/24 V DC
Пристрій зберігання даних	Користувальська програма		64 кроків (128 кБ)	64 кроків (128 кБ)	32 кроків (64 кБ)	16 кроків (32 кБ)	6 кроків (12 кБ)
	Постійний зберігання після зникнення живлення		Так				
	Макс. кількість елементів, для яких дані зберігаються при вимкненні живлення відключення		Усі програмні елементи, окрім R	Усі програмні елементи, крім R	Усі програмні елементи	Бітовий елемент: повний діапазон Елемент слова: 2000	Бітовий елемент: повний діапазон Слово елемент: 2000
	Апаратна підтримка та витривалість час		Резервна батарея для зберігання 3 роки	Резервна батарея для зберігання 3 роки	Резервна батарея для зберігання 3 роки	EEPROM, постійна пам'ять	EEPROM, постійна пам'ять
	Таймер		Точність 100 мс: T0–T209 Точність 10 мс: T210–T479 Точність 1 мс: T480–T511		Точність 100 мс: T0–T209 точність 10 мс: T210–T251 Точність 1 мс: T252–T255		
			16-бітний лічильник інкременту:				

М'який елемент ресурс uk	Лічильник	C0—C199 32-бітний лічильник інкременту та декременту: C200—C235 32-бітний високошвидкісний лічильник: C236—C263		16-бітний лічильник інкрементів: C0—C199 32-бітний лічильник інкременту та декременту: C200—C235 32-бітний високошвидкісний лічильник: C236—C263		
	Регістр даних	D0—D7999, 0—R32767		D0—D7999		
	Локальні дані зареєструвати	V0—V63				
	Індексація реєстру	Z0—Z15				
	Спеціальні дані зареєструвати	SD0—SD1023	SD0—SD1023	SD0—SD1023	SD0—SD511	SD0—SD511
	Допоміжне реле	M0—M10239	M0—M10239	M0—M2047		M0—M2047
	Локальні йдопоміжний реле	LM0—LM63				
	Спеціальний допоміжний реле	SM0—SM1023	SM0—SM1023	SM0—SM1023	SM0—SM511	SM0—SM511
	Стан реле	S0— S4095	S0— S4095	S0— S1023		S0— S1023
Переривання ресурсу	Внутрішній таймер переривання	3	3	3		3
	Зовнішній переривання	16	16	16		16
	На основі високошвидкісного лічильника переривання	8	8	8		8

Ім'я		VC5 (Планування)	VC3 (Планування)	VC2 (Планування)	VC1	VC1S
	Послідовний порт-на основі переривання	12	12	6		8
	Переривання після РТО вихід	/	8	3		2
	Переривання після інтерполяція	/	1	/		/
	Переривання під час проходження позиція	/	8	/		/
	Переривання під час вимкнення живлення	1	1	/		1
Звичайний	Час роботи базового привода інструкція	0.09 мкс	0.09 мкс	0.09 мкс	0,2 мкс	0.2 мкс
	Годинник реального часу clock	Підтримка мінімум безперервного виходу протягом 3	Підтримка мінімум безперервного виходу протягом 3 років після	Підтримка мінімум безперервного виводу протягом 3 років після	Підтримка мінімального безперервного вихідного сигналу	

		років після збою живлення	вимкнення живлення	зникнення живлення	протягом 3 років після подачі живлення відключення	
	Аналоговий потенціометр					
Комунікація	Комунікаційний порт	PORT0: RS232 PORT1: RS485 PORT2: RS485 PORT3: CAN PORT4: Мережевий порт PORT5: USB PORT : Мережевий порт	PORT0: RS232 PORT1: RS485 PORT2: RS485 PORT3: CAN PORT4: Мережевий порт PORT5: USB	PORT0: RS232 PORT1: RS485 PORT2: RS485 PORT3: USB (Примітка: Порт 2 є розширенням комунікаційного порту)		PORT0: RS232 PORT1: RS485
	Протокол зв'язку	EtherCAT CANopen, Modbus-TCP, Modbus, вільний порт, N:N, програмування порт протокол	Modbus, free-port, N:N, та протоколи порту програмування protocols			
Керування доступом та захист програм і користувача ON	Тип пароля	Пароль завантаження Пароль вивантаження Моніторинг пароль Підпрограма пароль				
	Вимкнути завантаження	Підтримується				
	Вимкнути форматування	Підтримується				

Ім'я		VC5(Планування)	VC3 (Планування)	VC2 (Планування)	VC1	VC1S
	Годинник реального часу, інструкція тактового сигналу	Доступно	Доступно	Доступно	Доступно	
	Дані та тактовий сигнал порівняння інструкція	Доступно	Доступно	Доступно	Доступно	
	Операція з числами з рухомою комою інструкція	Доступно	Доступно	Доступно	Доступно	
	Позиціонування інструкція	Доступно	Доступно	Доступно		
	Високошвидкісна I/O інструкція	Доступно	Доступно	Доступно	Доступно	PLS не підтримується
	Modbus та інструкція інвертора	Доступно	Доступно	Доступно	Доступно	Доступно

	Читання/запис EEPROM інструкція					
	Керування та обчислення інструкція	Доступно	Доступно	Доступно	Доступно	Підтримка PID інструкція лише
	Символьний рядок інструкція	Доступно	Доступно			
	Пакетна обробка даних інструкція	Доступно	Доступно			
	Технічний паспорт інструкція	Доступно	Доступно			
	Пам'ять картка інструкції					
Середній час між відмовами (MTBF)	Релейний вихід	200 000 годин, закріплений на землі, з механічним навантаженням, що наближається до нуля та керування температурою та вологістю control				
		100 000 годин, закріплений на землі, з механічним навантаженням, що наближається до нуля, ні керування температурою та вологістю control				
	Транзисторний вихід	300 000 годин, закріплений на землі, з механічним навантаженням, що наближається до нуля та керування температурою та вологістю контроль				
		150 000 годин, за умови механічного навантаження, що наближається до нуля, без контролю температури та вологості				
Термін служби контактів вихідного реле	220 V AC/15 VA/чутливий	1s ON/1s OFF, 3 200 000 разів				
	220 V AC /30 VA/чутливий	1s ON/1s OFF, 1 200 000 разів				
	220 V AC/72 VA/чутливий	1s ON/1s OFF, 300 000 разів				
Характеристики джерела живлення характеристики	Діапазон вхідної напруги	85 V AC–264 V AC (нормальна робота)				
Ім'я		VC5(Планування)	VC3 (Планування)	VC2 (Планування)	VC1	VC1S

Примітки:

1. Для отримання детальної інформації щодо технічних характеристик продукту, інструкцій з монтажу, а також експлуатації та технічного обслуговування VC1 серії PLCs, зверніться до *Посібника користувача VC серії PLC*.
2. Щоб отримати детальну інформацію про технічні характеристики виробу, інструкції з монтажу, а також експлуатацію та технічне обслуговування VC2 серії PLCs, зверніться до *Посібника користувача PLC серії VC2*.
3. Щодо деталей про технічні характеристики виробу, інструкції з монтажу, а також експлуатацію та технічне обслуговування VC3 серії PLCs, зверніться до *Посібника користувача PLC серії VC3*.
4. Щоб отримати детальну інформацію про технічні характеристики виробу, інструкції з монтажу, а також експлуатацію та технічне обслуговування VC5 серії PLCs, зверніться до *Посібника користувача PLC серії VC5*.
5. Резервна батарея може підтримувати зберігання протягом 3 років у робочих середовищах з температурою 25 °C.

1.1.2 Зовнішній вигляд VC1 та основних модулів серії VC1

На рисунку 1-1 показано зовнішній вигляд та структуру головних модулів серії VC1 (використовуючи VC1-1614MAT як приклад застосування).

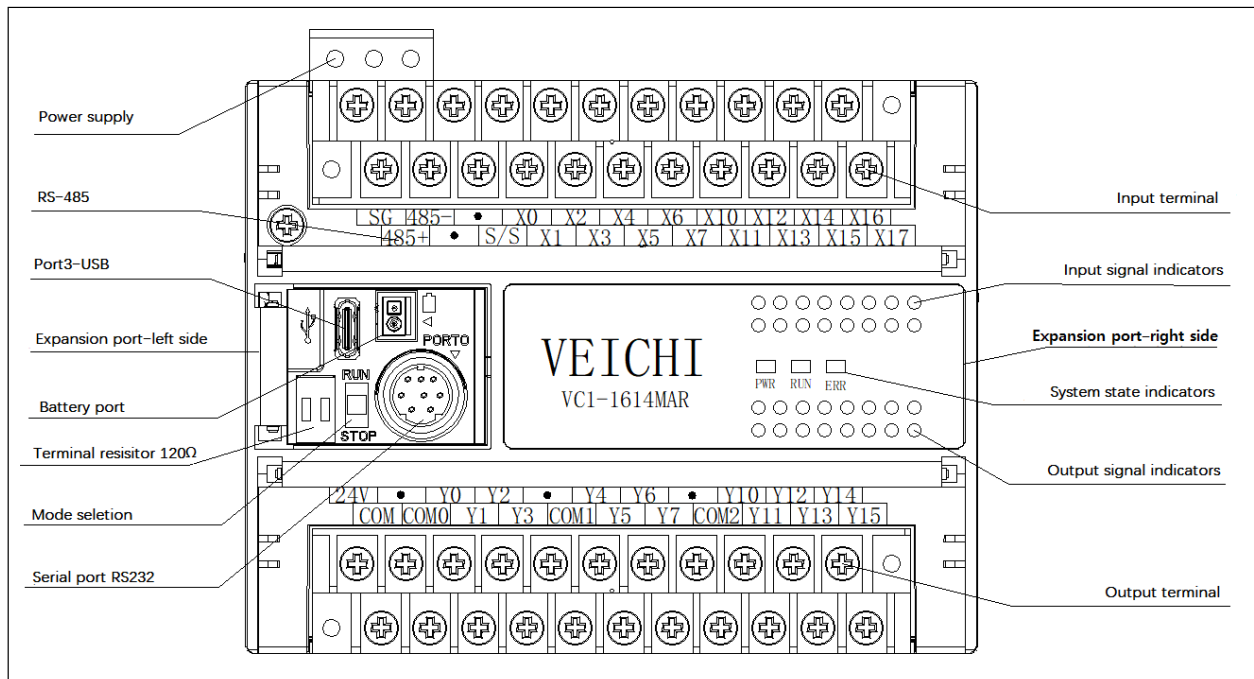


Рисунок 1-1 Зовнішній вигляд та структура VC1, а також основних модулів серії VC1

PORT0 та PORT1 є комунікаційними портами. PORT0 використовує рівень RS232 та роз'єм Mini DIN8. PORT1 та PORT2 серії VC1 використовують рівень RS485, PORT3 — це порт типу C.

1.2 Програмне забезпечення Auto Studio

Auto Studio — це програмне забезпечення для програмування, розроблене для серій VC1S, VC1, VC2, VC3 та VC5 продукції PLC. Це програмне забезпечення можна завантажити на вебсайті компанії.

Програмне забезпечення Auto Studio є стандартним програмним забезпеченням для Windows та графічним PLC інструментом програмування, що керується за допомогою миші та клавіатури. Для програмування доступні три стандартні мови: релейно-контактна схема (LAD), мова інструкцій (IL) та послідовна функціональна схема (SFC).

Програмне забезпечення Auto Studio підключається до PLC за допомогою послідовного кабелю програмування. Програмування мережі Modbus також може бути реалізоване через перетворення послідовного порту, а віддалене програмування може бути реалізоване через Modbus. Детальну інформацію про програмування Modbus та віддалений моніторинг див. у *Посібнику користувача програмного забезпечення для програмування Auto Studio*.

1.2.1 Базова конфігурація

Програмне забезпечення Auto Studio працює на IBM PC мікрокомп'ютерах або сумісних із ними пристроях і потребує встановлення в операційній системі серії Microsoft Windows. Сумісні операційні системи включають Windows 98,

Windows Me, NT 4.0, Windows 2000 та Windows XP.

Таблиця 1-2 описує мінімальну та рекомендовану конфігурацію, необхідну для Auto Studio.

Таблиця 1-2 Умови базової конфігурації Auto Studio

Елемент	Мінімальна конфігурація	Рекомендована конфігурація
CPU	Еквівалент Intel Pentium 233 або вище	Еквівалент Intel Pentium 1G або вище
Пам'ять	64 MB	128 MB
Графіка картка	Може працювати з роздільною здатністю 640 x 480 у 256-колірний режим	Може працювати з роздільною здатністю 800 x 600 у 65535-колірний режим
Зв'язок інтерфейс інтерфейс	Потрібен RS232 послідовний порт, що забезпечується роз'ємом типу DB9 (або ви можете використовувати USB інтерфейс, але USB-RS232 перетворювач потрібен).	
Інше пристрої	Кабелі програмування, розроблені для VEICHI PLCs	

1.2.2 Процес встановлення програмного забезпечення Auto Studio

Інсталяційний пакет Auto Studio, випущений компанією Suzhou Veichi Electric Co.,Ltd., є окремою виконуваною програмою. Двічі клацніть на ній, щоб розпочати процес інсталяції, а потім встановлюйте її крок за кроком згідно з майстром інсталяції. Ви можете вибрати шлях інсталяції за потреби.

Після завершення інсталяції VEICHI група програм з'являється в меню «Пуск», а інсталятор також створює ярлик Auto Studio на робочому столі. Ви можете двічі клацнути по ярлику, щоб запустити програму.

Видалення: Програмне забезпечення можна видалити через Панель керування Windows. Для оновлення та встановлення нової версії Auto Studio попередню версію необхідно спочатку видалити.

1.2.3 Інтерфейс запуску Auto Studio

Головний інтерфейс програми загалом складається із семи частин: меню, панель інструментів, вікно менеджера проєктів, вікно дерева інструкцій, вікно повідомлень, рядок стану та робоча область.

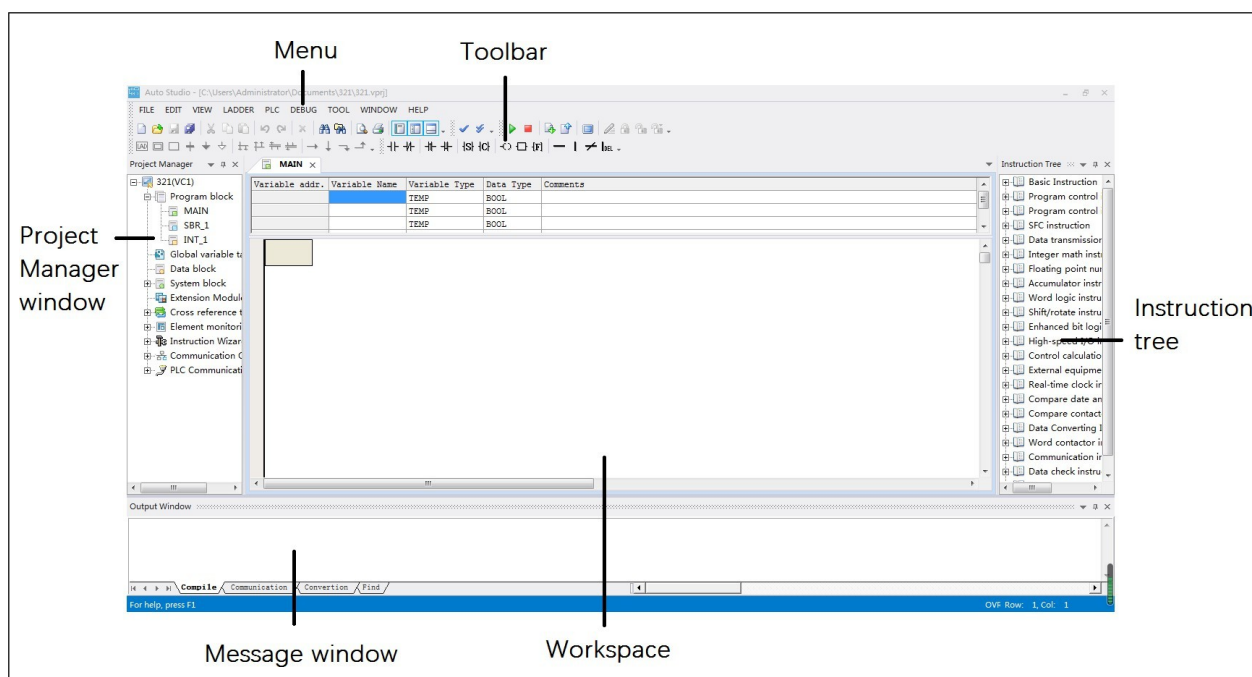


Рисунок 1-4 Головний інтерфейс Auto Studio

Зверніться до *Посібника користувача програмного забезпечення для програмування Auto Studio* для отримання детальної інформації про те, як використовувати програмне забезпечення для програмування Auto Studio.

1.2.4 Кабель програмування

Клієнти можуть використовувати послідовний кабель програмування, наданий VEICHI, для програмування та виконання пусконаладжувальних робіт на PLC. На рисунку 1-5 показано підключення кабелю програмування.



Рисунок 1-5 Підключення кабелю програмування

1.3 Комунікаційна функція

Основний модуль мікро-PLC серії VC1/VC1S інтегрує два послідовні порти: PORT0, PORT1 та один порт USB. PORT0 та PORT1 підтримують оновлення прошивки та програмування. Ці послідовні порти використовують протоколи зв'язку Modbus та N: N, також застосовний визначений користувачем протокол вільного порту. Порт PORT0 не може бути використаний для мережевого підключення N:N.

1.3.1 Комунікаційна мережа Modbus

Привод серії VC PLC може утворювати мережу Modbus RS485 з іншими інтелектуальними приладами або пристроями, такими як інвертори та PLCs, через інтерфейси RS485 PORT1 та PORT2 або через PORT0 (при використанні PORT0 потрібен перетворювач RS232-RS485). Максимальна відстань зв'язку становить 1200 метрів, а найвища швидкість передачі даних — 115200 біт/с. Ви можете вибрати режим передачі RTU.

Привод серії VC PLC можна підключити до інвертора, PLC, сенсорного екрана або приладу через інтерфейс RS232 PORT0. Максимальна відстань зв'язку становить 15 метрів, а найвища швидкість передачі даних — 115200 біт/с.

Щодо деталей про зв'язок Modbus, зверніться до розділу 10 "Посібник з функцій зв'язку" та додатка G "Протоколи зв'язку Modbus".

1.3.2 N:N комунікаційна мережа

Серії VC1S, VC1, VC2, VC3 та VC5 PLCsare, налаштовані з використанням протоколу зв'язку N:N, розробленого VEICHI, який можна застосувати для формування мережі зв'язку N:N через інтерфейси RS485 PORT1 та PORT2. Завдяки перетворенню RS232 на RS485, порт PORT0 також можна використовувати для мережевого підключення N:N.

Комунікаційний протокол N:N дозволяє від 2 до 32 станціям PLC обмінюватися даними між собою. Найвища швидкість передачі даних становить 115200 біт/с. Цей протокол може бути застосований для створення однорівневої або

дворівневої мережі.

Для отримання детальної інформації про зв'язок N:N зверніться до розділу 10 "Посібник з функцій зв'язку".

1.3.3 Вільний порт мережі

Протокол вільного порту — це режим протоколу, у якому обмін даними виконується на основі визначених користувачем форматів даних зв'язку. Він підтримує два формати даних: ASCII та двійковий. У цьому режимі комунікаційного порту PLC можна використовувати для зв'язку з різними пристроями, що використовують визначені користувачем формати, такими як інвертори, сканери штрих-кодів, вимірювальні прилади та інші інтелектуальні пристрої, що використовують вільні протоколи зв'язку. А PLC може здійснювати зв'язок з одним пристроєм у режимі RS232 або RS485, а також може утворювати мережу RS485 з кількома пристроями.

Детальну інформацію щодо протоколу зв'язку free-port див. у розділі 10 "Посібник з функцій зв'язку".

Розділ 2 Опис функцій

Цей розділ стисло описує ресурси та принципи програмування, а також метод конфігурації системи серії VC PLCs, як налаштувати режим роботи та функціонування PLC, а також різноманітні функції та програмні інструменти, пов'язані з введенням системи в експлуатацію. Детальнішу інформацію див. у *Посібнику користувача програмного забезпечення для програмування Auto Studio*.

Розділ 2 Опис функцій.....	11
2.1 Ресурси та принципи програмування.....	12
2.1.1 Програмування ресурсів.....	12
2.1.2 PLC механізм керування (цикл сканування модель).....	14
2.1.3 Виконання програми користувача сторожовий таймер.....	15
2.1.4 Режим роботи постійного сканування mode.....	15
2.1.5 Завантаження та зберігання файлів користувача.....	15
2.1.6 Ініціалізація елемента.....	15
2.1.7 Збереження даних під час вимкнення живлення.....	15
2.1.8 Цифрова фільтрація для вхідних точок.....	16
2.1.9 Режим без акумулятора mode.....	16
2.1.10 Програма користувача захист.....	16
2.2 Конфігурація системи.....	17
2.2.1 Системний блок.....	17
2.2.2 Блоки даних.....	23
2.2.3 Глобальна змінна таблиця.....	24
2.3 Режим роботи та керування станом.....	24
2.3.1 Концепції станів RUN та STOP системи.....	24
2.3.2 Стан RUN та STOP стану.....	24
2.3.3 Встановлення стану вихідних точок у стані STOP стану.....	25
2.4 Введення системи в експлуатацію.....	25
2.4.1 Завантаження програми та вивантаження.....	25
2.4.2 Механізм звітування про помилки.....	26
2.4.3 Зміна користувацької програми в режимі онлайн.....	28
2.4.4 Очищення та форматування.....	28
2.4.5 Перевірка PLC інформації онлайн.....	29
2.4.6 Запис та примусове встановлення значень елементів, а також таблиця моніторингу елементів table	29
2.4.7 Генерація блоків даних з RAM.....	32

2.1 Ресурси та принципи програмування

2.1.1 Програмування ресурсів

Таблиця 2-2 VC1 програмні ресурси

Ім'я		Специфікація та опис
I/O Конфігурація	Макс. кількість яких I/O точки	128 (теоретичне значення)
	Кількість розширень модулі	15
Ємність файлу користувача capacity	Користувальська програма потужність	16 кроків
	Розмір блока	8000 D елементи
Швидкість інструкції	Базова інструкція	0.2 мкс/інструкція
	Застосування інструкція	A кілька мкс – сотні мкс /інструкція
Кількість інструкцій	Базова інструкція	32
	Застосування інструкція	234
Ресурси програмних елементів ВПримітка 7	I/O точки	128 входів/128 виходів (входи: X0–X177; виходи: Y0–Y177)Примітка 1
	Допоміжне реле	2048 (M0–M2047)
	Локальний допоміжний реле	64 (LM0–LM63)
	Спеціальний допоміжний реле	512 (SM0–SM511)
	Стан реле	1024 (S0–S1023)
	Таймер	256 (T0–T255)Примітка 2
	Лічильник	256 (C0–C255)Примітка 3
	Регістр даних	8000 (D0–D7999)
	Локальні дані зареєструвати	64 (V0–V63)
	Індексація реєстру	16 (Z0–Z15)
	Спеціальні дані зареєструвати	512 (SD0–SD511)
Ресурси переривань	Переривання на основі зовнішнього входу	16 (фронти спрацювання переривання можуть визначатися користувачами, що відповідають передньому та задньому фронтам сигналів на клеммах X0–X7)
	На основі високошвидкісного лічильника переривання	8
	Внутрішній таймер переривання	3
	Послідовний порт-базоване переривання	6

	Переривання після РТО вихід	2
	Переривання під час вимкнення живлення	
Комунікаційна функція	Зв'язок інтерфейс	Два асинхронні порти послідовного зв'язку: PORT0: RS232 PORT1 Один USD порт
	Зв'язок протокол	Modbus, вільний порт, N:N (VEICHI пропрієтарний протокол, який можна застосувати для створення 1:N або N:N комунікаційні мережі)

Ім'я		Специфікація та опис	
Спеціальна функція	Високошвидкісний лічильник	X0, X1	Одиночний вхід: 50 кГц Одночасні входи від X0 до X7: загальна частота не перевищує 80 кГц.
		X2–X5	Одиночний вхід: 10 кГц
	Високошвидкісний імпульсний вихід	Y0, Y1, Y2	Три незалежні виходи: 100 кГц (тільки для транзисторного типу виходу)
	Цифрове фільтрування	Застосування цифрової фільтрації для X0–X7 та апаратної фільтрації для інших портів	
	Аналоговий потенціометр		
	Примітка 4		
Спеціальна функція	Виклик підпрограми	А Дозволено максимум 64 користувацькі підпрограми та шість рівнів вкладених підпрограм. Підтримуються локальні змінні, і кожна підпрограма може надавати максимум 16 параметрів для передачі інформації. Псевдоніми змінних також підтримуються.	
	Програма користувача захист	Завантажити пароль	Передбачено три типи паролів. А пароль має довжину не більше 8 символів, кожен символ є алфавітно-цифровим та чутливим до регістру.
		Завантажити пароль	
		Годинник пароль	
	Підпрограма шифрування	Інше захисні заходи	А пароль становить не більше 16 символів, і кожен символ є буквено-цифровий та регістрозалежний регістрозалежний.
		Інше захисні заходи	Функції вимкнення форматування та завантаження передбачені.
	Режим програмування примітка 5	Auto Studio програмування програмне забезпечення ЯПримітка 6	Його необхідно встановити та запустити на мікрокомп'ютері IBM PC або сумісних пристроях.
	Годинник реального часу clock	Вбудований, та живлення від резервної батареї.	

Примітки:

Примітка 1: Адреси елементів X та Y пронумеровані у вісімковій системі числення, наприклад: Адреса X10 представляє 8-му вхідну точку.

Примітка 2: Адреси елементів T класифікуються на три категорії відповідно до точності синхронізації:

VC1

(1) Точність 100 мс: T0–T209

(2) Точність 10 мс: T210–T251

(3) Точність 1 мс: T252–T255

Примітка 3: Адреси елементів C класифікуються на три категорії відповідно до ширини та функції значень лічильника:

VC1

- (1) 16-бітний лічильник інкрементів: C0–C199
- (2) 32-бітний лічильник інкременту та декременту: C200–C235
- (3) 32-бітний високошвидкісний лічильник: C236–C263

Примітка 5: Функція примусового встановлення елементів надається для полегшення введення в експлуатацію та аналізу користувацької програми, що підвищує ефективність пусконаладжувальних робіт. А Одночасно можна примусово встановити максимум 128 бітових елементів та 16 словних елементів.

Примітка 6: Користувацьку програму можна змінювати в режимі онлайн, коли PLC працює.

Примітка 7: Деякі внутрішні ресурси програмних елементів PLCs зарезервовані для внутрішнього використання. По можливості не використовуйте ці елементи в користувацькій програмі. Детальнішу інформацію див. у Додатку С "Зарезервовані елементи".

2.1.2 PLC механізм керування (цикл сканування модель)

Головні модулі серії VC PLC працюють відповідно до моделі циклу сканування.

Система виконує чотири завдання у послідовному та циклічному порядку: виконання програми користувача, обмін даними, обслуговування та оновлення I/O. Кожен цикл завдань називається циклом сканування.

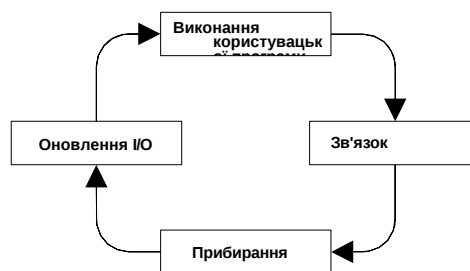


Рисунок 2-1PLC робочий механізм

- Виконання користувацької програми
Система послідовно виконує інструкції користувацької програми, починаючи від першої інструкції до кінцевої інструкції головної програми.
- Зв'язок
Система здійснює обмін даними з програмним забезпеченням для програмування та реагує на команди зв'язку, такі як команди завантаження, RUN та STOP, що надсилаються програмним забезпеченням для програмування.
- Прибирання
Система виконує різноманітні завдання з обслуговування, такі як оновлення індикаторів панелі, оновлення значень часу програмного таймера, оновлення спеціальних допоміжних реле та спеціальних регістрів даних.
- Оновлення I/O
I/O оновлення включає фазу оновлення виходу та фазу оновлення входу.

Фаза оновлення виходу: вмикання або вимикання відповідної апаратної вихідної точки згідно зі значенням елемента Y (ON або OFF).

Фаза оновлення входу: Перетворення стану ON/OFF апаратного вхідного пункту на відповідне значення елемента X (ON або OFF).

2.1.3 Виконання програми користувача сторожовий таймер

Система відстежує час виконання користувацької програми в кожному циклі сканування. Як тільки виявляється, що час виконання користувацької програми перевищує встановлене значення, система зупиняє користувацьку програму. Ви можете встановити час сторожового таймера на вкладці **Set Time** діалогового вікна системного блоку в інтерфейсі програмного забезпечення Auto Studio.

2.1.4 Режим роботи постійного сканування mode

У режимі роботи з постійним циклом сканування час кожного циклу сканування є однаковим під час роботи системи. Ви можете активувати режим постійного сканування та встановити час постійного сканування на вкладці **Встановити час** діалогового вікна системного блоку інтерфейсу фонового програмного забезпечення Auto Studio. Типовий цикл постійного сканування дорівнює 0, тобто постійне сканування вимкнено. Коли фактичний цикл сканування довший за цикл постійного сканування, програма виконується з фактичним циклом сканування.

□ Примітка

Налаштування константи часу сканування не може перевищувати час сторожового таймера.

2.1.5 Завантаження та зберігання файлів користувача

Ви можете програмувати та керувати головним модулем шляхом завантаження спеціального файлу користувача до нього.

Існує чотири типи файлів користувача: файли програм користувача, файли блоків даних, файли системних блоків та файли довідкової інформації користувача. Файли довідкової інформації користувача включають таблиці глобальних змінних та файли джерел даних користувача.

Ви можете обрати завантаження файлу програми користувача, файлу блоку даних або файлу системного блоку. Коли файл вибрано, відповідний файл довідкової інформації користувача додається до пакету та завантажується.

Усі файли користувача серій VC1S та VC1 записуються в область FLASH головного модуля для постійного зберігання.

□ Примітка

1. Переконайтеся, що джерело живлення головного модуля працює належним чином протягом певного часу (довше ніж 30s) після завантаження файлу, щоб файл міг бути коректно записаний у головний модуль.

2.1.6 Ініціалізація елемента

Коли PLC переходить у стан роботи (STOP→RUN), пристрій ініціалізує відповідні програмні елементи згідно з даними, збереженими під час вимкнення живлення, даними, що зберігаються на EEPROM, блоками даних та значеннями елементів. У таблиці 2-6 описано пріоритети даних.

Таблиця 2-6 PLC пріоритети ініціалізації даних

Тип пристрою зберігання	Живлення OFF → ON	STOP → RUN
Дані збережено під час зникнення живлення	Найвищий	Найвищий
Зберігання даних на EEPROM	Високий	Високий
Блоки даних (коли вибрано "Datablock enabled" в розширених налаштуваннях системи блоку)	Середній	Середній
Значення елементів (коли вибрано "Збереження значення елемента" в параметрі розширені налаштування системного блоку)	—	Низький

2.1.7 Збереження даних під час вимкнення живлення

- Умови для збереження даних при вимкненні живлення
При виявленні зникнення живлення система зупиняє користувацьку програму та зберігає значення елементів, встановлені в межах діапазону зберігання в системному блоці, у файл резервної копії на випадок зникнення живлення.
- Відновлення елемента при увімкненні живлення-ON

Якщо файл резервної копії після зникнення живлення є коректним після ввімкнення живлення, значення вказаних програмних елементів відновлюються до значень, збережених під час останнього зникнення живлення.

Після ввімкнення живлення система видаляє значення даних елементів, що виходять за межі діапазону зберігання. Якщо файл резервної копії втрачено або він пошкоджений, дані всіх елементів видаляються.

- Діапазон зберігання налаштування

Діапазон елементів, для яких необхідно зберегти значення даних, можна встановити в **Діапазоні збереження** системного блоку, як показано на Рисунку 2-2.

Діапазон зберігання серій VC1S та VC1 можна встановити лише для групи.

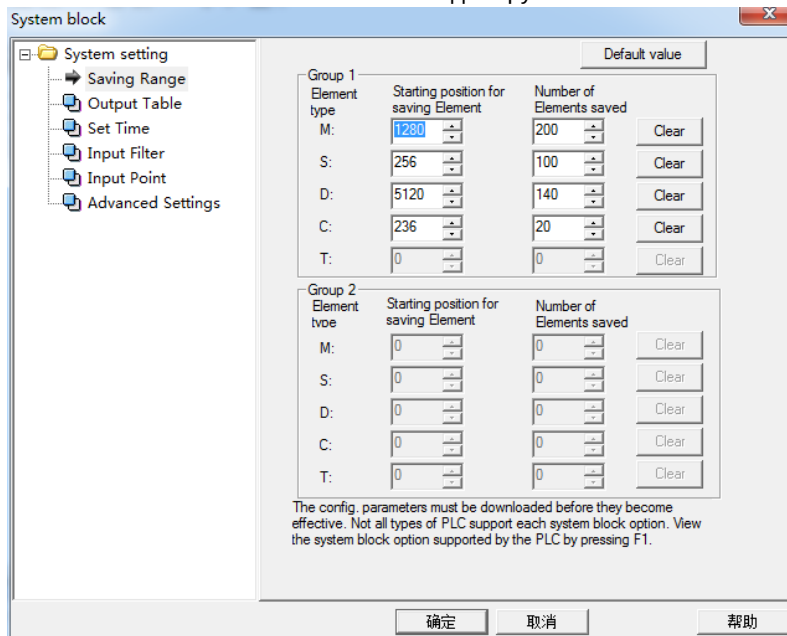


Рисунок 2-2 Встановлення діапазону збереження

□ Примітка

1. Коли VC1S/VC1 серії PLC стикається з вимкненням живлення, дані її елементів, встановлені в межах діапазону зберігання, записуються в EEPROM.

2.1.8 Цифрова фільтрація для вхідних точок

Вхідні точки X0–X7 головних модулів серій VC1S, VC1, VC2, VC3 та VC5 налаштовані з функцією цифрової фільтрації, яку можна використовувати для фільтрації сигналу переход на портах. Ви можете змінити константу вхідної фільтрації, налаштувавши **Вхідний фільтр** у системному блоці.

2.1.9 Режим без акумулятора mode

Основні модулі серії VC1 можуть працювати без акумуляторів. Коли ви обираєте режим без акумулятора, система не повідомляє про системну помилку, спричинену відсутністю акумулятора (втрата даних елементів, втрата таблиць примусового керування, помилка файлу користувачської програми).

Зверніться до опису конфігурації **режиму без акумулятора** в **Розширених налаштуваннях** системного блоку.

2.1.10 Програма користувача захист

Серія VC PLCs розроблена з багаторівневим захистом паролем та іншими стратегіями безпеки.

Таблиця 2-7 Захист користувачської програми

Користувачська програма захист	Опис

Вимкнути форматування	Після вимкнення форматування в конфігурації системного блоку та завантаження системного блоку в PLC, ви не можете видалити користувацьку програму, системний блок та блок даних на PLC шляхом форматування. Щоб увімкнути форматування, потрібно завантажити новий системний блок, у якому форматування не вимкнено.
Користувацька програма захист	Опис
Завантажити пароль	Використовується для обмеження функції завантаження
Вимкнути завантаження	Під час операції завантаження виберіть опцію вимкнення завантаження у діалоговому вікні завантаження, і тоді ви не зможете вивантажити дані пізніше, навіть якщо введете пароль для вивантаження. Щоб увімкнути завантаження, вам потрібно знову завантажити дані користувача та вибрати опцію увімкнення завантаження в діалогове вікно завантаження.
Завантажити пароль	Використовується для обмеження функції завантаження
Годинник пароль	Використовується для обмеження функції годинника
Програмування пароля	Програмісти можуть встановлювати паролі для шифрування головної програми, підпрограм та підпрограм переривань. Під час відкриття проекту в програмному забезпеченні для програмування ви не зможете переглядати та редагувати вміст зашифрованих програм. Щоб переглянути та відредагувати його, потрібно відкрити діалогове вікно дешифрування та ввести правильний пароль для розшифрування. Метод шифрування: натисніть правою кнопкою миші на програму, яку потрібно зашифрувати, виберіть Зашифрувати/Розшифрувати у контекстному меню, а потім введіть пароль і підтвердьте його. Метод дешифрування: натисніть правою кнопкою миші на програму, яку потрібно дешифрувати, виберіть Шифрування/Дешифрування у контекстному меню, а потім введіть правильний пароль.

□ Примітка

Якщо ви введете неправильний пароль п'ять разів поспіль, мікро-PLC серії VC блокує введення пароля на п'ять хвилин.

2.2 Конфігурація системи

2.2.1 Системний блок

PLC інформація про конфігурацію, налаштована в системному блоці, компілюється як файл системного блоку, що є важливим PLC файлом користувача. Перед початком роботи з PLC необхідно скопіювати та завантажити файл системного блоку.

Конфігурація системного блоку включає наступні елементи:

- Діапазон збереження (елемент збереження діапазону)
- Встановлення часу (сторожовий таймер, час постійного сканування та час виявлення зникнення живлення)
- Точка входу (режим запуску точки входу)
- Комунікаційні порти (комунікаційні порти та налаштування протоколів)
- Пріоритет переривань (конфігурація пріоритету переривань)
- Конфігурація привода
- Конфігурація Ethernet
- Вихідна таблиця (вихідна таблиця налаштувань)
- Фільтр
- Розширені налаштування (блок даних, значення елемента зберігається, режим без батареї та вимкнення форматування)
- Конфігурація спеціального модуля
- Комунікаційний модуль
- CANopen конфігурація

Після налаштування системного блоку ви можете вибрати меню **PLC/Компілювати Все**, щоб скопіювати файл системного блоку, а потім завантажити його.

- Збереження діапазону

У разі зникнення живлення, VC серії PLCs може зберігати дані елементів, встановлених у межах діапазону збереження, у зону збереження при зникненні живлення, і ці дані можуть бути збережені та використані після повторного увімкнення живлення пристрою.

На рисунку 2-3 показано діапазон адрес елементів, для яких дані мають бути збережені на першій вкладці діалогового вікна.

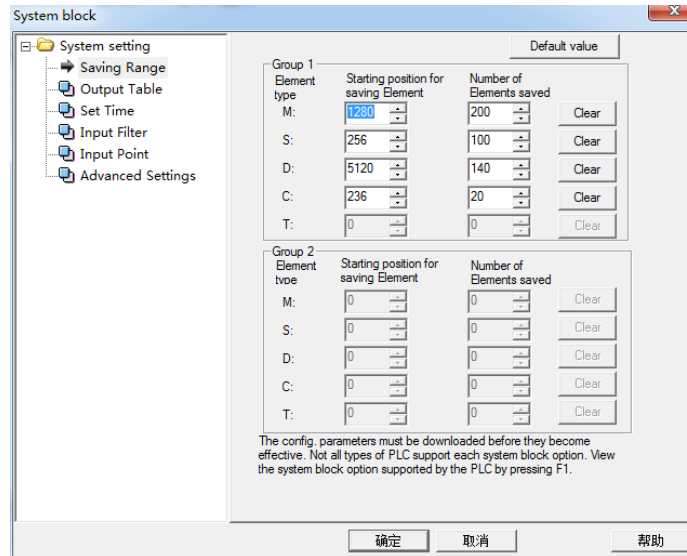


Рисунок 2-3 Налаштування діапазону адрес елементів, дані для яких мають бути збережені

□ Примітка

Діапазони адрес та групи елементів, для яких можна зберігати дані, відрізняються залежно від PLC моделей.

За замовчуванням певний діапазон зберігання автоматично встановлюється для кожного з елементів D, M, S, T та C.

Ви можете змінювати діапазон адрес елементів, для яких потрібно зберегти дані на вкладці, за власним бажанням. Натисніть кнопку Clear праворуч, щоб встановити кількість елементів (для яких потрібно зберегти дані) на нуль.

Лише одну групу можна встановити для кожної з серій VC1S та VC1.

□ Примітка

Для серій VC1S та VC1 PLCs, жоден елемент T не може бути встановлений у межах діапазону зберігання.

Робота системи під час зникнення живлення: PLC зберігає дані елементів, встановлених у межах діапазону пам'яті, у файл резервної копії на випадок зникнення живлення.

Робота системи під час увімкнення живлення: PLC перевіряє дані в області зберігання на випадок зникнення живлення. Якщо дані, збережені в області зберігання на випадок зникнення живлення, правильні, область зберігання на SRAM залишається без змін. Якщо дані неправильні, PLC видаляє дані всіх елементів (включаючи елементи, що знаходяться в межах та поза діапазоном зберігання) на SRAM.

● Вихідна таблиця

Клацніть вкладку **Вихідна таблиця**, після чого ви зможете встановити стан, у який переходить вихідна точка після зупинки PLC, як показано на Рисунку 2-4.

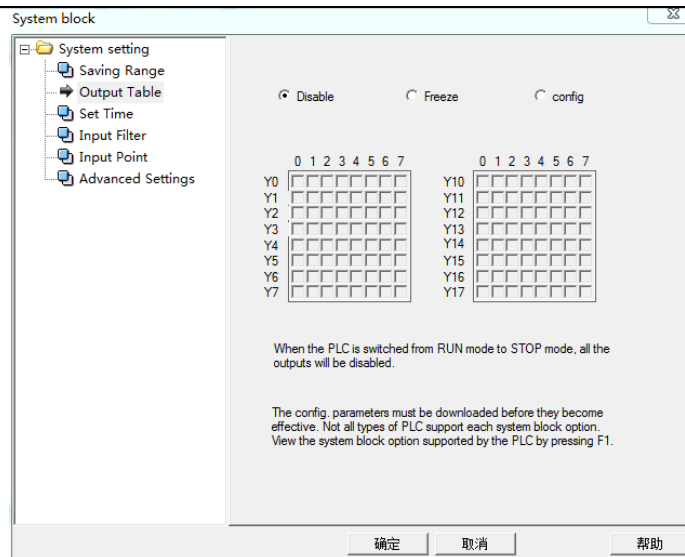


Рисунок 2-4 Налаштування вихідної таблиці

Функція налаштування вихідної таблиці використовується для встановлення стану, в який переходить вихідна точка після зупинки PLC. Вихідна точка може перейти в один із наступних станів після зупинки CPU привода PLC:

- (1) Вимкнення: PLC вимикає всі вихідні точки під час перемикання зі стану роботи у стан зупинки.
- (2) Фіксація: PLC фіксує всі вихідні точки у їхньому кінцевому стані під час зупинки.
- (3) Конфігурація: PLC встановлює всі вихідні точки у відомий стан під час зупинки. Стан за замовчуванням для всіх вихідних точок — вимкнено (0).

- Встановити час

На рисунку 2-5 показано вкладку встановлення часу.

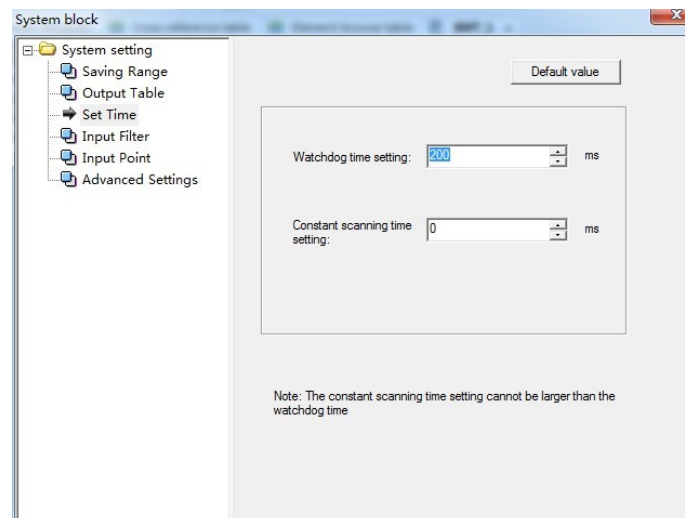


Рисунок 2-5 Встановлення часу

1. Налаштування часу сторожового таймера

Ви можете встановити час роботи сторожового таймера користувацької програми. Час сторожового таймера — це максимально дозволений час виконання користувацької програми. Коли фактичний час виконання користувацької програми перевищує час сторожового таймера, PLC зупиняє виконання користувацької програми, вмикає індикатор аварії програми (червоний) і видає вихідний сигнал відповідно до конфігурації системи. Час сторожового таймера можна встановити в діапазоні від 0ms до 1000ms, а значення за замовчуванням становить 200ms.

2. Налаштування сталого часу циклу сканування

Сталий час сканування — це сталий проміжок часу, протягом якого система сканує регістр. Регістр налаштування сталого часу сканування системи зчитується, і користувацька програма сканується лише один раз за цей сталий час. Сталий час можна встановити в діапазоні від 0ms до 1000ms, значенням за замовчуванням є 0ms. Значення за замовчуванням дорівнює 0ms, і сталий час сканування не активовано. Активує встановлений сталий час сканування, якщо він не дорівнює нулю.

- фільтр

Натисніть вкладку **Вхідний фільтр**, щоб встановити константу фільтрації входу для точок входу PLC. Зовнішні завади, що надходять через точки входу, можна відфільтрувати за допомогою цифрової фільтрації. X0–X7 — це цифрові точки входу з функцією цифрової фільтрації. Інші цифрові точки входу використовують апаратну фільтрацію. Фільтрацію входу серій VC1S/VC1 можна налаштувати окремо для кожного вхідного порту, а константи фільтрації можна встановлювати послідовно від 0ms до 60ms. На рисунку 2-6 показано налаштування вхідної фільтрації для серії VC1.

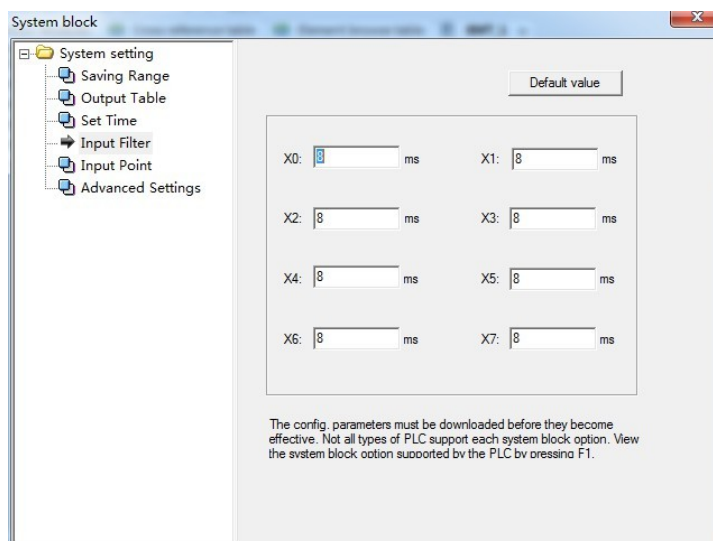


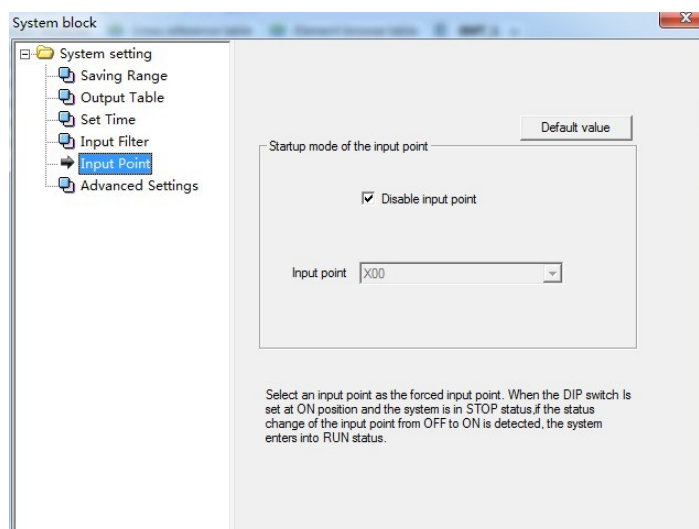
Рисунок 2-6 Налаштування вхідного фільтра

- точка

На рисунку 2-7 показано вкладку налаштування вхідного сигналу.

1. Встановіть вхідну точку для запуску PLC

Коли **точку входу вимкнення** не вибрано, ви можете встановити точку входу (від X0 до X17), щоб примусово перевести PLC у стан RUN. PLC перемикається зі стану STOP у стан RUN, коли виявлено, що точка входу має



рівень ON.

2. Вимкнути вхідну точку

Виберіть **вимкнення вхідної точки**, щоб вимкнути функцію пуску на основі вхідної точки.

- Додаткові налаштування

Розширені налаштування включають увімкнення блоку даних, збереження значення елемента, режим без батареї, тощо.

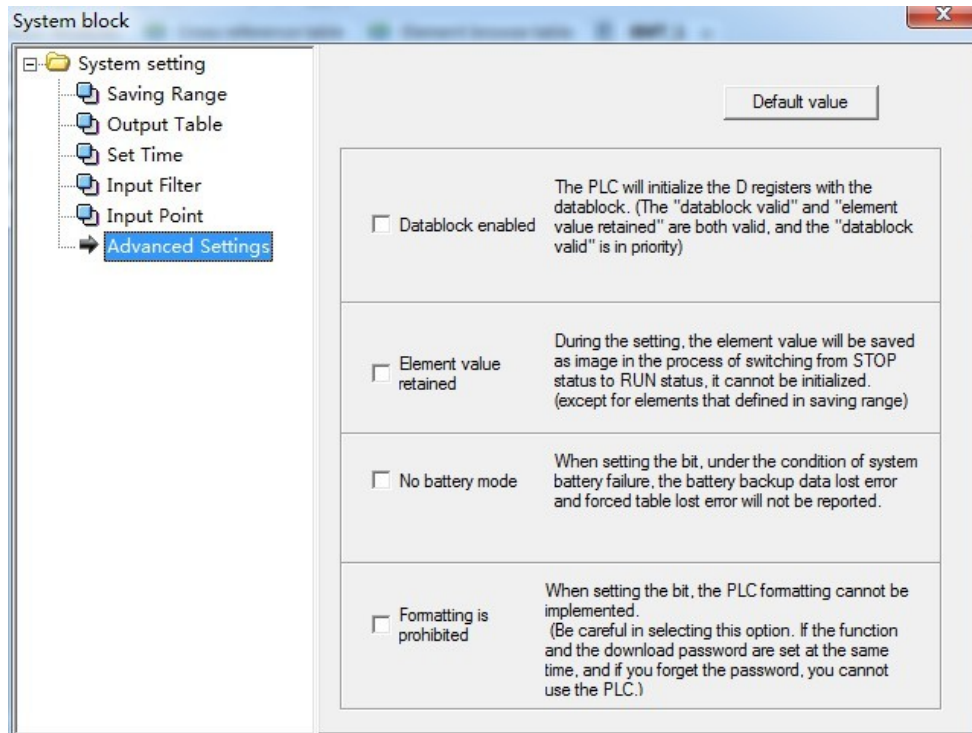


Рисунок 2-8 Додаткові налаштування

4. Блок даних увімкнено

Виберіть увімкнення блоку даних, і PLC використовує блоки даних для ініціалізації елементів D під час перемикання з STOP на RUN.

5. Значення елемента збережено

Значення елемента вибрано, і PLC не ініціалізує значення елемента, а зберігає їх як зображення під час перемикання з STOP на RUN.

□ Примітка

Коли "**Блок даних увімкнено**" та "**Значення елемента збережено**" є дійсними, "**Блок даних увімкнено**" має пріоритет. Детальнішу інформацію див. у розділі 2.1.6 "Ініціалізація елемента".

6. Режим без акумулятора mode

Виберіть **режим без акумулятора**, і система не повідомлятиме про помилки, спричинені втратою даних резервного акумулятора або втратою таблиць примусового керування у разі виходу з ладу резервного акумулятора.

- Комунікаційний порт

Ви можете налаштувати три комунікаційні порти на вкладці комунікаційних портів системного блоку. На рисунку 2-9 показано налаштування комунікаційного порту. Елемент включає вибір протоколу зв'язку та налаштування параметрів конкретного протоколу.

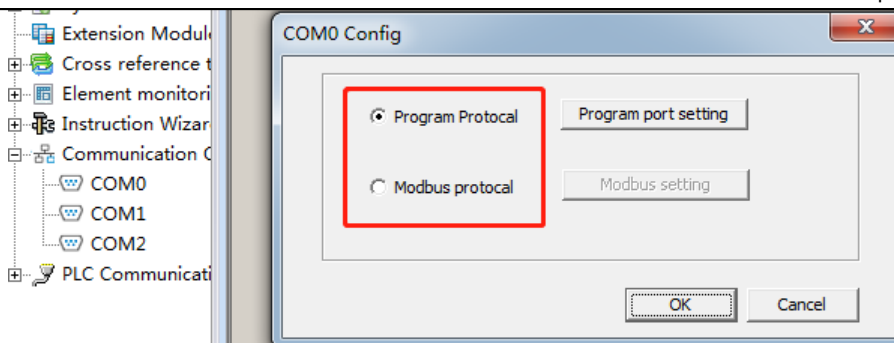


Рисунок 2-9 Налаштування комунікаційного порту налаштування

За замовчуванням PORT0 використовує протокол порту програмування, тоді як PORT1 та PORT2 не використовують жодного протоколу. Ви можете налаштувати PORT0, PORT1 та PORT2 відповідно до своїх потреб.

1. Порт програмування протокол

За замовчуванням PORT0 використовує протокол порту програмування, спеціалізований протокол для зв'язку програмного забезпечення для програмування серії VC PLC. За допомогою цього протоколу ви можете встановити швидкість передачі даних між PC та PORT0 через інструмент конфігурації послідовного порту Auto Studio.

2. Протокол вільного порту protocol

Протокол вільного порту — це режим зв'язку, що використовує формати файлів даних, визначені користувачем. Режим зв'язку через вільний порт підтримує два формати даних, а саме ASCII та двійковий код. А PLC може використовувати зв'язок через вільний порт лише у стані RUN. При використанні режиму вільного порту PLC не може обмінюватися даними з програмувальними пристроями.

Конфігуровані параметри включають швидкість передачі даних, біт даних, перевірку на парність, стоповий біт, виявлення початкового символу, виявлення кінцевого символу, тайм-аут між символами та тайм-аут між кадрами.

3. Протокол зв'язку Modbus protocol

Пристрої зв'язку Modbus включають головні та підпорядковані станції. А Головна станція може обмінюватися даними з підпорядкованою станцією (такою як перетворювач частоти) і надсилати керуючі кадри на підпорядковану станцію відповідно до функціональних кодів протоколу зв'язку Modbus, а підпорядкована станція відповідає на запит головної станції.

PORT0 може бути налаштований як підпорядкована станція, тоді як PORT1 та PORT2 можуть бути налаштовані як головна або підпорядкована станція.

Налаштовувані параметри включають швидкість передачі даних, біт даних, перевірку на парність, стоповий біт, режим ведучий-ведений, номер станції, режим передачі, а також час очікування та кількість спроб повтору в режимі ведучого.

4. N:N протокол зв'язку протокол

N:N — це протокол зв'язку N:N, розроблений компанією Suzhou Veichi Electric Co.,Ltd. для мікромереж PLC. PLCs у мережі N:N можуть автоматично обмінюватися деякими значеннями своїх елементів D та M.

Усі PORT0, PORT1 та PORT2 можуть використовувати N:N комунікаційний протокол.

□ Примітка

Щоб дізнатися детальніше про використання протоколів free-port, Modbus та N:N, зверніться до розділу 10 "Посібник з комунікаційних функцій guide".

● Конфігурація модуля розширення

Ви можете встановити властивість модуля на вкладці конфігурації модуля розширення, як показано на Рисунку 2-10.

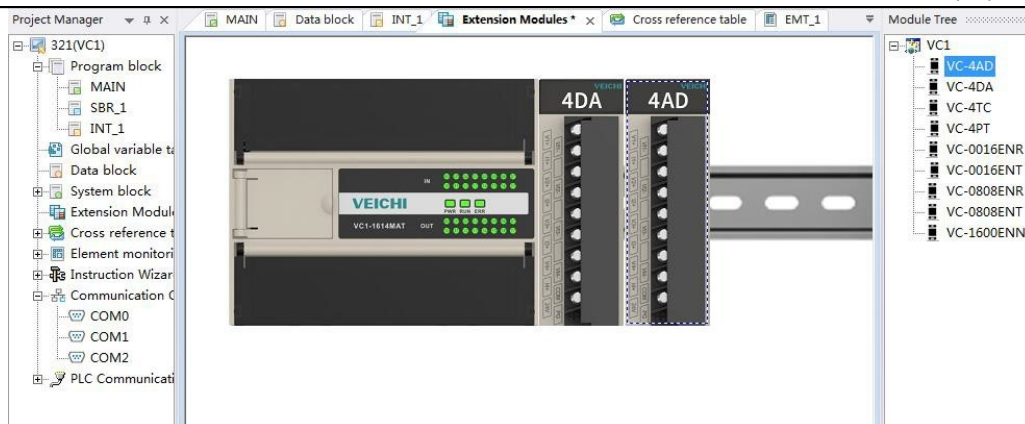


Рисунок 2-10 Конфігурація модуля розширення

1. Тип модуля

Ви можете вибрати тип модуля для VC-4AD, VC-4DA, VC-4PT, VC-4RT та іншого модуля, як показано на Рисунку 2-10.

2. Властивість модуля

Після вибору **Типу модуля** активується відповідна **Властивість модуля** і може бути відображене наступне діалогове вікно.

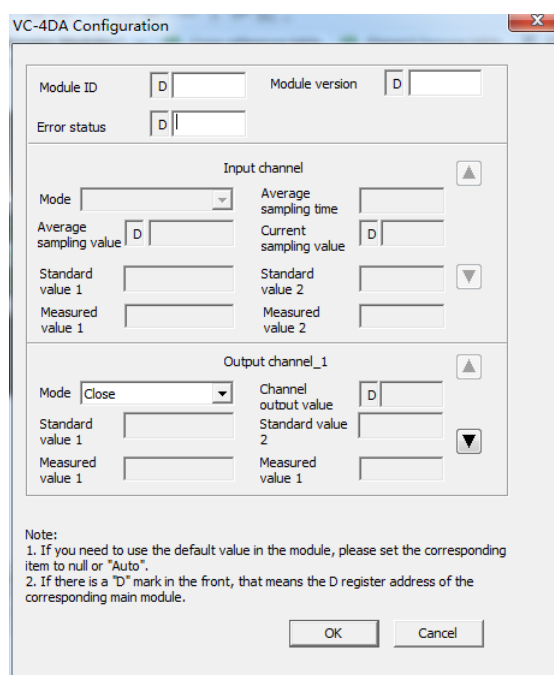


Рисунок 2-11 Налаштування властивостей модуля setting

У діалоговому вікні, показаному на Рисунку 2-11, ви можете налаштувати канал для спеціального модуля, включаючи **Режим** (характеристики сигналу), **Цифрове значення при нулі**, **Верхня межа цифрового значення**, **Усереднення вибірки** та **Поточне значення вибірки**. Для отримання детальної інформації про значення та методи налаштування параметрів зверніться до посібників користувача відповідних спеціальних модулів.

2.2.2 Блоки даних

Блоки даних використовуються для встановлення значень за замовчуванням у D елементах. Якщо ви завантажуєте скопійовані налаштування блоку даних у PLC, PLC використовує блок даних для ініціалізації відповідних D елементів під час запуску.

Редактор блоку даних дозволяє призначати початкові дані у D регістр (пам'ять даних). Ви можете призначати дані

словом або подвійним словом, але не байтам. Ви також можете додавати коментарі, вводячи "/" перед рядком символів.

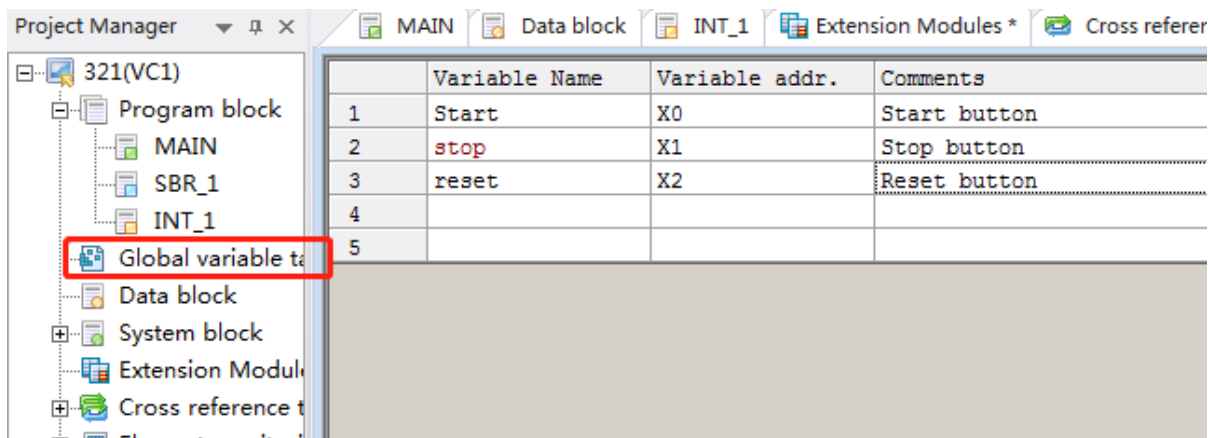
Детальну інформацію щодо інструкцій з експлуатації блоків даних див. у *Посібнику користувача програмного забезпечення для програмування Auto Studio*.

2.2.3 Глобальна змінна таблиця

Глобальна змінна — це значуще символічне ім'я, визначене для певної PLC адреси, до якого можна звертатися в межах усього проєкту, що еквівалентно використанню відповідного елемента змінної. Глобальна змінна визначається в таблиці глобальних змінних. Глобальна змінна містить три атрибути, а саме: **Ім'я змінної**, **Адреса змінної** та **Коментарі**.

Правило визначення глобальної змінної: складається з A–Z, a–z, 0–9, підкреслення та китайських ієрогліфів, причому назва змінної не може починатися з цифр і не може бути окремими цифрами. Назва може бути написана великими або малими літерами, а довжина не може перевищувати 8 байтів. Жоден елемент типу літери+цифри не може бути використаний як назва програми чи змінної. Не повинно бути ні пробілів, ні назв, що збігаються з ключовими словами. Зарезервовані ключові слова включають назви базових типів даних, назви інструкцій та символи операцій у мові списку інструкцій.

Для мікро-приводів серії VC1 PLCs дозволено завантажувати не більше 140 глобальних змінних відповідно (згідно з максимальною кількістю коментарів). В іншому випадку їх можна лише зберегти локально, але неможливо завантажити.



	Variable Name	Variable addr.	Comments
1	Start	X0	Start button
2	stop	X1	Stop button
3	reset	X2	Reset button
4			
5			

На рисунку 2-16 показано таблицю глобальних змінних.

Рисунок 2-16 Таблиця глобальних змінних

2.3 Режим роботи та керування станом

Ви можете запустити або зупинити PLCs будь-яким із трьох наступних способів.

1. Використовуючи перемикач вибору режиму.
2. Налаштування режиму запуску вхідних точок та зовнішніх терміналів у системних блоках, що керуються вказаними терміналами.
3. Використання програмного забезпечення для запуску або зупинки PLC, якщо перемикач вибору режиму встановлено в ON.

2.3.1 Концепції станів RUN та STOP системи

Робочі стани основних модулів включають стани RUN та STOP.

- **RUN**
Коли головний модуль перебуває у стані RUN, PLC виконує користувацьку програму. Тобто цикл сканування включає чотири завдання, а саме: виконання користувацької програми, обмін даними, обслуговування та оновлення I/O.
- **STOP**
Коли головний модуль перебуває у стані STOP, PLC не виконує користувацьку програму, проте виконує три інші

завдання у кожному циклі сканування, а саме: обмін даними, обслуговування та оновлення I/O.

2.3.2 Стан RUN та STOP стану

- Як змінити з STOP на RUN
 1. Скидання PLC

Якщо перемикач вибору режиму встановлено в ON, а PLC скинуто (включаючи скидання при ввімкненні живлення системи), система автоматично переходить у стан RUN.

□ Примітка

Якщо елемент конфігурації системи **режиму керування точкою входу** в головному модулі є дійсним, стан відповідної вхідної клеми дорівнює ON. В іншому разі система не зможе перейти у стан RUN після скидання.

2. Встановлення перемикача вибору режиму вручну

Система перемикається зі стану STOP у стан RUN, коли ви встановлюєте перемикач вибору режиму з OFF у ON.

3. Режим запуску вхідних точок

Коли елемент конфігурації системи **вхідної точки режиму запуску** на вкладці системного блоку є дійсним, головний модуль змінюється з STOP на RUN, як тільки система виявляє, що вказані вхідні точки (X0 до X17) змінюються з OFF на ON.

□ Примітка

Коли ви обираєте режим керування вхідною точкою, перемикач вибору режиму має бути встановлений у положення ON одночасно, інакше PLC не зможе перейти у стан RUN.

● Як змінити з RUN на STOP

1. Скидання PLC

Якщо перемикач вибору режиму встановлено в положення OFF, виконайте скидання PLC (включаючи скидання при ввімкненні живлення), і система автоматично перейде в стан RUN.

□ Примітка

Коли перемикач вибору режиму встановлено в ON, система також може перейти у стан зупинки після скидання, якщо елемент конфігурації системи **Режим керування вхідною точкою** є дійсним, а призначена вхідна точка перебуває в OFF.

2. Встановлення перемикача вибору режиму вручну

Система перемикається зі стану RUN у стан STOP, коли ви встановлюєте перемикач вибору режиму з ON на OFF.

3. Використовуючи режим керування

Система переходить з RUN на STOP після виконання інструкції STOP у користувацькій програмі.

4. Автозупинка у разі виникнення несправностей

Система зупиняє виконання користувацької програми при виявленні серйозної помилки (такої як помилки користувацької програми або перевищення часу виконання користувацької програми).

2.3.3 Встановлення стану вихідних точок у стані STOP стану

Ви можете встановити вихідний стан вихідної точки (Y), коли PLC зупинено. На ваш вибір є три режими: Вимкнуті режим виходу — Коли PLC зупинено, усі вихідні точки є OFF.

Режим фіксації виходу — Коли PLC зупинено, усі вихідні точки фіксуються в останньому стані. Режим конфігурації виходу — Ви можете встановити стан вихідних точок за потреби, коли PLC зупинено.

Ви можете керувати вищезазначеними налаштуваннями у вихідній таблиці на вкладці системного блоку. Детальнішу інформацію див. у налаштуваннях вихідної таблиці у розділі 2.2.1 "Системний блок".

2.4 Введення системи в експлуатацію

2.4.1 Завантаження програми та вивантаження

● Завантажити

Функція завантаження використовується для завантаження системного блоку, блоку даних та користувацької програми, створених у програмному забезпеченні Auto Studio, до PLC через послідовні порти, при цьому PLC має бути у стані STOP під час завантаження.

Під час завантаження, якщо програму було змінено після останньої компіляції, ви отримаєте сповіщення про необхідність повторної компіляції програми, як показано на Рисунку 2-17.

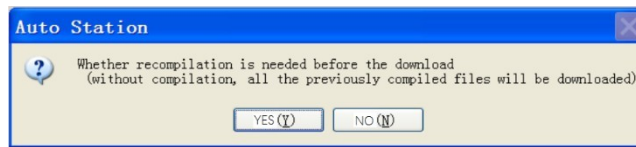


Рисунок 2-17 Поради щодо перекомпіляції програми

□ Примітка

Вибір кнопки **N** означає, що вам не потрібно виконувати перекомпіляцію, і програмне забезпечення застосовує попередні відредаговані результати, проте програма, що виконується та завантажена у PLC, відрізняється від тієї, що відображається в інтерфейсі програмного забезпечення.

Під час завантаження програми програмне забезпечення відкриває вікно пароля з запитом на введення пароля для завантаження, якщо його налаштовано, а пароль завантаження не було введено після запуску програмного забезпечення. Завантаження розпочнеться після успішної перевірки введеного пароля, в іншому разі система запропонує вам повторно ввести пароль або натиснути кнопку Cancel для виходу із завантаження.

● Завантажити

Функція вивантаження використовується для вивантаження системного блоку, блоку даних та користувацької програми з PLC до PC через послідовні порти, а також для їх збереження для нових проєктів. Коли дані резервного копіювання від акумулятора є дійсними, вибирається файл, після чого відповідний файл довідкової інформації для користувача об'єднується та вивантажується. На рисунку 2-18 показано діалогове вікно вивантаження.

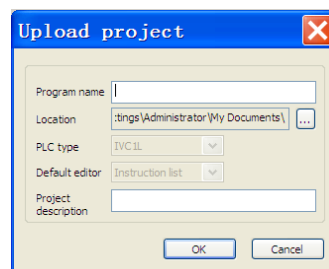


Рисунок 2-18 Вікно діалогу завантаження

Під час завантаження програми ви можете завантажити її безпосередньо, якщо не налаштовано пароль для завантаження. Програмне забезпечення відкриває вікно пароля з запитом на введення пароля для вивантаження, якщо його налаштовано і пароль не було введено після запуску програмного забезпечення. Вивантаження розпочинається після введення правильного пароля, в іншому разі програмне забезпечення видає підказку та повертається до інтерфейсу діалогового вікна вивантаження.

Якщо ви виберете функцію **Вимкнути Вивантаження** під час завантаження програми, PLC не зможе вивантажити програму пізніше, якщо ви не введете правильний пароль для скасування функції Вимкнути Вивантаження.

2.4.2 Механізм звітування про помилки

А система може виявляти та повідомляти про два типи помилок: системна помилка та помилка виконання користувацької програми. А системна помилка спричинена ненормальною роботою системи.

А помилка виконання користувацької програми спричинена аномальним виконанням користувацької програми.

Усі помилки пронумеровані одномоментно, і кожен код помилки відповідає певній помилці. Детальнішу інформацію див. у Додатку F "Системні коди помилок".

● Механізм звітування для системної помилки

Коли система виявляє виникнення системної помилки, код системної помилки записується у спеціальний регістр даних SD3, і одночасно встановлюється спеціальне реле SM3. Ви можете отримати інформацію про системну помилку, оцінивши код помилки, що зберігається в SD3.

Якщо одночасно виникає кілька системних помилок, система записує лише код найсерйознішої помилки в SD3.

Коли виникає серйозна системна помилка, програма користувача зупиняється, а індикатор ERR на головному модулі вмикається на тривалий час.

- Механізм звітування про помилку виконання користувацької програми

Коли виникає помилка виконання програми користувача, система встановлює спеціальне реле SM20 і записує поточний код помилки у спеціальний регістр даних SD20.

Коли наступна прикладна інструкція виконується коректно, SM20 скидається, тоді як SD20 все ще зберігає попередній код помилки.

Система послідовно записує помилки виконання користувацької програми у стек. Спеціальні регістри даних SD20 — SD24 утворюють стек помилок, який фіксує коди п'яти останніх помилок виконання користувацької програми.

When a user program execution error occurs and the current error code is different from one stored in SD20, the error code is added to the error stack. Figure 2-19 shows the stack push process of error codes when a user program execution error occurs.

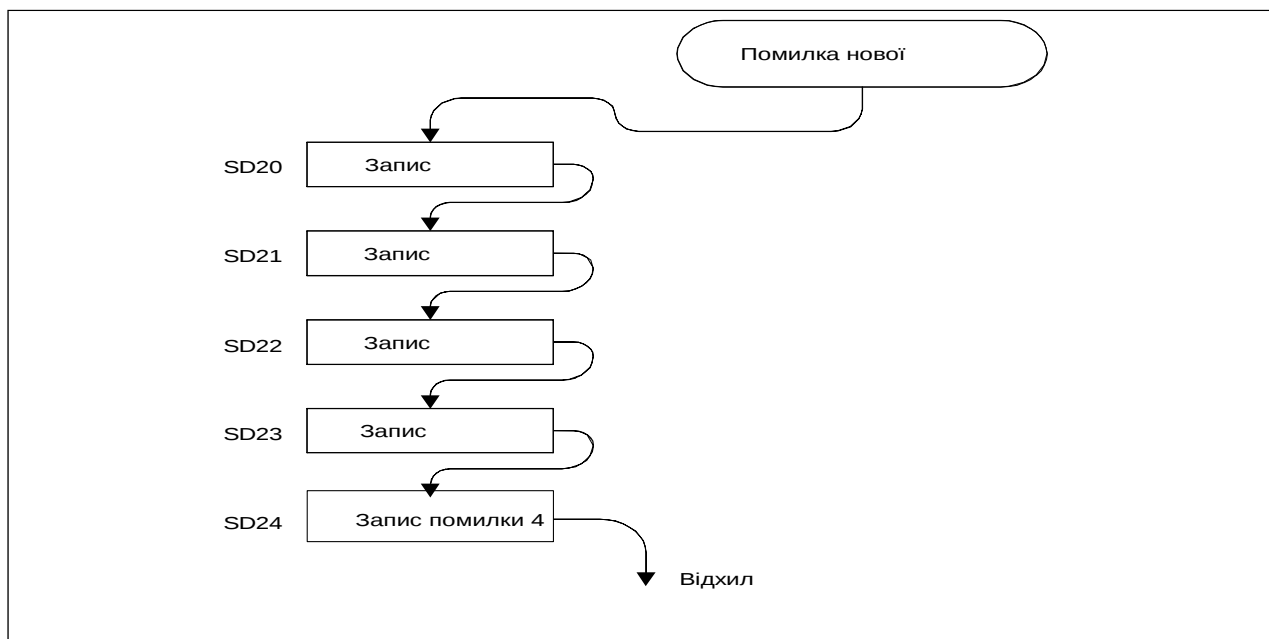


Рисунок 2-19 Процес розміщення в стеку кодів помилок

Коли виникає серйозна помилка виконання користувацької програми, користувацька програма зупиняється, а індикатор ERR на головному модулі вмикається на тривалий час. У менш серйозних випадках індикатор ERR на головному модулі не вмикається.

- Перевірка інформації про помилку онлайн

Підключивши PLC до вашого PC через послідовний порт, ви можете зчитувати різноманітну інформацію про стан PLC через Auto Studio, включаючи коди та описи вищезгаданих системних помилок та помилок виконання користувацької програми.

В основному інтерфейсі Auto Studio ви можете натиснути елемент **PLC->PLC інформація** , щоб перевірити інформацію PLC, як показано на Рисунок 2-20.

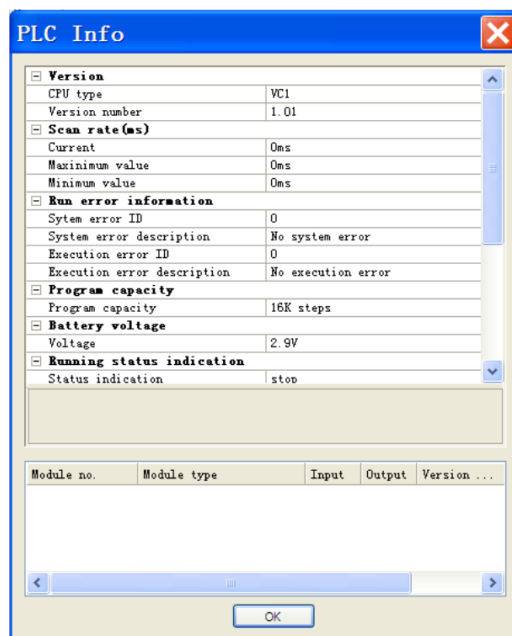


Рисунок 2-20 PLC інформація

Системна помилка ID — це код системної помилки, що зберігається у SD3, тоді як **помилка виконання ID** — це код помилки виконання користувацької програми, що зберігається у SD20. Відповідні описи помилок, що відображаються обома, наведені для вашої довідки.

2.4.3 Зміна користувацької програми в режимі онлайн

Ви можете використовувати функцію онлайн-модифікації, коли потрібно змінити програму користувача без зупинки PLC.

□ Попередження

У випадках, коли можливі людські жертви або втрата майна, функція онлайн-модифікації програми повинна використовуватися професіоналами з належними заходами захисту.

- **Метод експлуатації**

Після того, як ви переконалися, що зв'язок PC-PLC успішно налаштовано, а PLC перебуває у стані RUN, ви можете натиснути **Налагодження->Оперативна модифікація** в головному інтерфейсі Auto Studio, щоб увійти в стан оперативної модифікації.

У стані оперативної модифікації ви можете змінювати головну програму, підпрограму та підпрограму переривання як зазвичай. Після внесення змін ви натискаєте меню **PLC->Завантажити**, і змінена програма автоматично компілюється та завантажується у PLC. Після завершення завантаження PLC виконує нову завантажену програму.

- **Межі**

1. У стані оперативної модифікації таблиця глобальних змінних та таблиця локальних змінних будь-якої програми не можуть бути змінені, також неможливо додати або видалити будь-яку підпрограму чи підпрограму переривання.

2. Коли програма перебуває у стані оперативного редагування, Auto Studio автоматично виходить зі стану оперативного редагування, якщо PLC зупинено.

2.4.4 Очищення та форматування

Операції очищення включають PLC очищення значення елемента, PLC очищення програми та PLC очищення блоку даних. Форматування призначене для очищення всіх даних та програм у PLCs.

- **PLC значення елемента очистити**

Функція очищення значень елементів PLC може очистити всі значення елементів у PLC. Значення елементів слід очищати, коли PLC перебуває у стані STOP.

Очищення значень елементів у PLC призводить до неправильної роботи PLC або втрати проміжних робочих даних, тому цю функцію слід використовувати з Увага. Під час процесу виконання програмне забезпечення відображає вікно підтвердження, щоб ви могли вибрати, чи продовжувати, чи скасувати поточну операцію.

- **PLC програма очистити**

Функція очищення програми PLC може видаляти користувацькі програми у PLC. Користувацькі програми слід видаляти, коли PLC перебуває у стані STOP.

Очищення користувацьких програм у PLC призводить до невиконання користувацьких програм приводом PLC, тому цю функцію слід використовувати з Увага. Під час процесу роботи програмне забезпечення відображає вікно підтвердження, щоб ви могли вибрати, чи продовжувати, чи скасувати поточну операцію.

- **PLC блок даних очистити**

Функція очищення блоку даних PLC може очистити всі налаштування блоків даних у PLC. Блоки даних слід очищати, коли PLC перебуває у стані STOP.

Очищення блоків даних у PLC призводить до того, що PLC припиняє використання попередньо встановленого значення блоку даних для ініціалізації елементів D, тому цю функцію слід використовувати з Увага. Під час процесу роботи програмне забезпечення відображає вікно підтвердження, щоб ви могли вибрати, чи продовжувати, чи скасувати поточну операцію.

- PLC форматування

Функція форматування PLC може формувати всі дані PLC, включаючи очищення користувацької програми, відновлення конфігурації за замовчуванням та очищення блоків даних. Блоки даних слід очищати, коли PLC перебуває у стані STOP.

Ця операція очищає всі завантажені та налаштовані дані, тому цю функцію слід використовувати з Увага. Під час виконання операції програмне забезпечення відображає вікно підтвердження, щоб ви могли вибрати, чи продовжувати, чи скасувати поточну операцію.

2.4.5 Перевірка PLC інформації онлайн

- PLC інформація

Функцію PLC інформації можна використовувати для отримання та відображення різноманітних даних про роботу та важливої інформації про PLCs. Ви можете побачити важливу інформацію про поточну роботу PLC у вікні інформаційного дисплея, як показано на Рисунок 2-21.

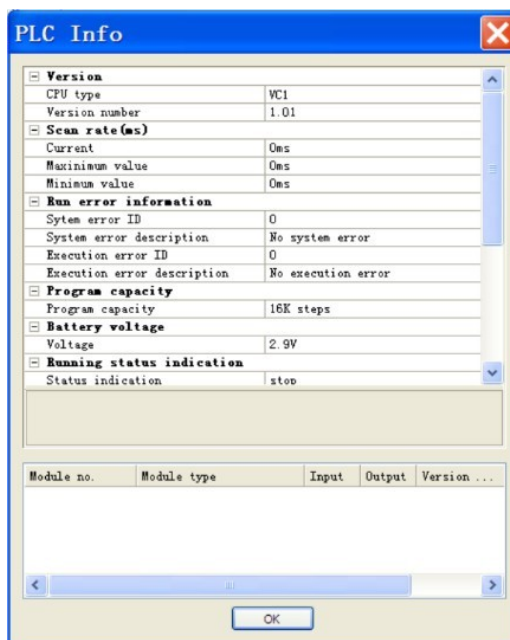


Рисунок 2-21 Поточна інформація про роботу PLC

- PLC час

Функцію часу PLC можна використовувати для відображення та встановлення поточного часу PLCs, як показано на рисунку 2-22.

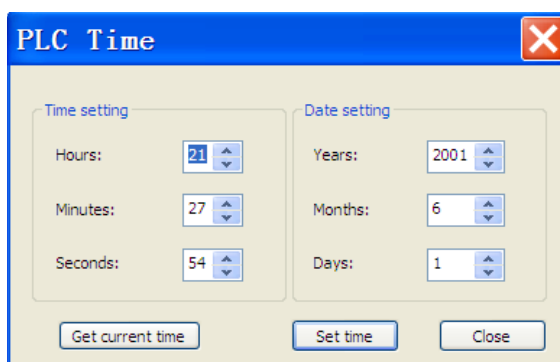


Рисунок 2-22 Налаштування PLC часу

У вікні відображається поточна дата та час з PLC. Ви можете налаштувати параметри часу та натиснути кнопку **Встановити Час**, щоб встановити новий час для PLC.

2.4.6 Запис та примусове встановлення значень елементів, а також таблиця моніторингу елементів table

- Запис та примусові значення елементів

Під час процесу введення в експлуатацію ви можете вручну змінити значення певних програмних елементів для досягнення певних умов, запис значення елемента та примусове встановлення надають вам цю функцію. Різниця між записом та примусовим встановленням полягає в тому, що: значення елемента при записі є дійсним лише один раз, і записане значення може змінюватися під час роботи програми, але примусово встановлене значення елемента записується в PLC апаратне забезпечення до моменту його скасування вами.

Для виконання функції запису або примусового встановлення значень, спочатку потрібно вибрати елементи, які

необхідно записати або примусово встановити, а потім натиснути праву кнопку миші на меню, щоб вибрати **Write** або **Force**. Після цього з'явиться відповідне діалогове вікно, у якому буде наведено перелік усіх адрес програмних елементів, на які посилаються вибрані елементи. Ви можете вибірково записати або примусово встановити певні значення програмних елементів, після підтвердження ці значення будуть

надіслано до PLC апаратного забезпечення. Коли ці значення набувають чинності в апаратному забезпеченні, ви можете побачити результати змін під час подальшого процесу введення в експлуатацію.

На рисунку 2-23 показано діалогове вікно для запису значення елемента.

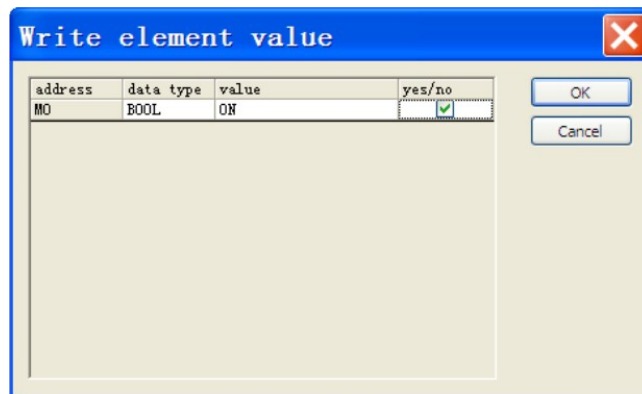


Рисунок 2-23 Діалогове вікно для запису значення

елемента Рисунок 2-24 показує діалогове вікно для примусового встановлення значення елемента.

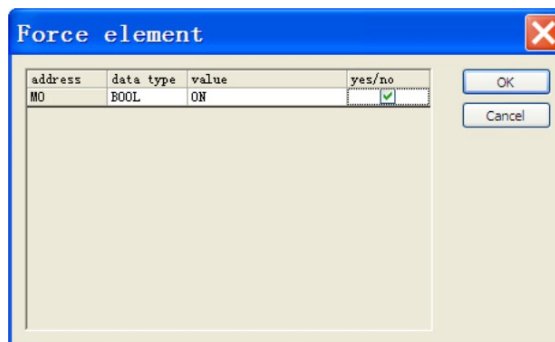


Рисунок 2-24 Діалогове вікно для примусового

встановлення значення елемента Примусово встановлений програмний елемент має позначку блокування в I

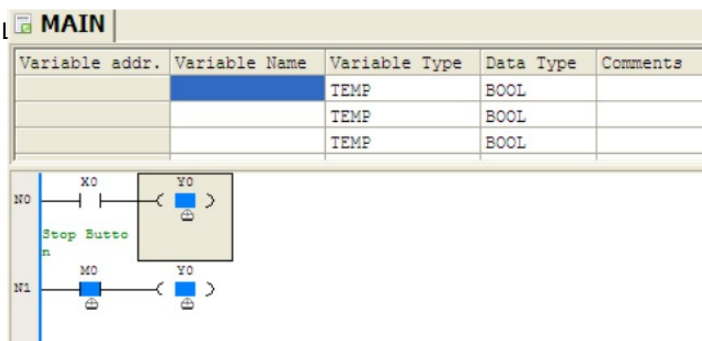


Рисунок 2-25 Позначка блокування примусового програмного елемента

- Значення елемента примусове вимкнення

Для значень елементів, які більше не потрібно примусово встановлювати, ви можете скасувати примусове встановлення. Коли ви видаляєте функцію примусового встановлення, вам потрібно спочатку вибрати цільові елементи, натиснути праву кнопку миші та вибрати **Скасувати примусове встановлення**, після чого з'явиться діалогове вікно, у якому відображаються всі примусово встановлені програмні елементи. Ви можете вибірково скасувати примусове встановлення певних програмних елементів, натиснути кнопку OK для підтвердження, і тоді ці примусові значення будуть видалені з PLC, як і відповідні позначки блокування. На рисунку 2-26 показано діалогове вікно скасування примусового встановлення.

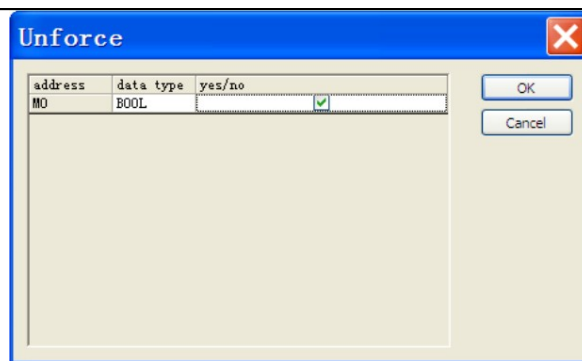


Рисунок 2-26 Скасування примусового керування

- Таблиця моніторингу елементів table

Таблиця моніторингу елементів забезпечує функцію відстеження значень елементів під час введення в експлуатацію. Програмні вхідні та вихідні елементи, регістри та словні елементи можна додати до таблиці моніторингу елементів, щоб їх можна було відстежувати після завантаження програми в PLCs.

Таблиця моніторингу елементів працює у двох режимах: режимі редагування та режимі моніторингу. У режимі редагування можна виконувати всі функції редагування, але неможливо виконувати функції моніторингу. У режимі моніторингу можна виконувати як функції моніторингу, так і функції редагування.

У режимі моніторингу значення елементів, що відображаються, оновлюються автоматично, після чого змінені або примусово встановлені значення елементів можуть бути оновлені своєчасно.

Таблиця моніторингу стану елементів забезпечує функції, що включають редагування, послідовність, пошук, автооновлення та відображення поточних значень визначених елементів, запис значень елементів, примусове встановлення значень елементів/змінних та скасування примусового встановлення значень.

На рисунку 2-27 показано таблицю моніторингу елементів.

MAIN EMT_1 *					
	Element Name	data type	display format	current value	new value
1		WORD	Decimal		
2	X0	BOOL	Binary	OFF	
3	Y0	BOOL	Binary	ON	
4	M0	BOOL	Binary	ON	
5		WORD	Decimal		

Рисунок 2-27 Моніторинг стану елементів таблиця

2.4.7 Генерація блоків даних з RAM

Значення даних 500 D регістрів (максимум), зчитаних з PLC послідовно, відображаються, і результати можуть бути об'єднані з блоком даних або накладені на початковий блок даних.

Відкрийте вікно для створення блоків даних з RAM, як показано на Рисунку 2-28.

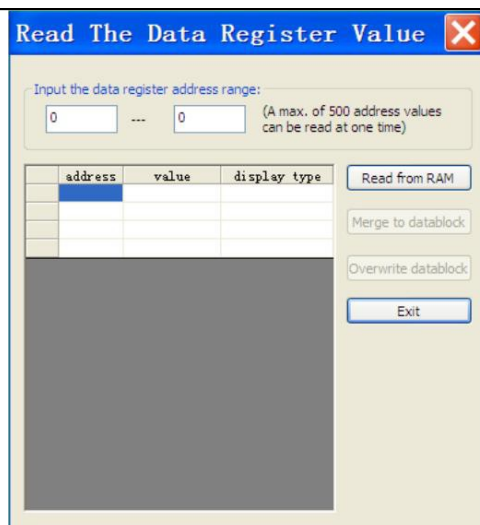


Рисунок 2-27 Window для генерування блоків даних з RAM

Ввівши діапазон блоків даних для зчитування, натисніть кнопку **Зчитати з RAM**, і дані будуть зчитані у список після успішного виконання інструкції.

Ви можете вибрати шістнадцятковий, десятковий, вісімковий або двійковий формат у полі **типу відображення** для виведення даних.

Після успішного зчитування даних стають доступними кнопки **Об'єднати з блоком даних** та **Перезаписати блок даних**. Натисніть кнопку **Об'єднати з блоком даних**, щоб додати згенеровані результати до кінцевої частини поточних блоків даних, і натисніть кнопку **Перезаписати блок даних**, щоб замінити наявний вміст у блоці даних згенерованими результатами. Після виходу з вікна зчитування значень регістрів програмне забезпечення видає запит про те, що блок даних було змінено, і вікно блоку даних відкривається автоматично.

Розділ 3 Програмний елемент та дані

У цьому розділі детально описано визначення, класифікацію та функції програмних елементів серії VC

PLCs.Розділ 3 Програмний елемент та дані.....	34
3.1 Тип та функція програмних елементів.....	35
3.1.1 Короткий огляд програмних елементів.....	35
3.1.2 М'який список елементів.....	35
3.1.3 точки входу та виходу.....	37
3.1.4 Допоміжні реле.....	37
3.1.5 Стан реле.....	38
3.1.6 Таймер.....	38
3.1.7 Лічильник.....	39
3.1.8 Регістр даних.....	39
3.1.9 Спеціальне допоміжне реле.....	40
3.1.10 Спеціальний регістр даних регістр.....	40
3.1.11 Індикація реєстру.....	41
3.1.12 Локальне допоміжне реле.....	41
3.1.13 Локальний регістр даних регістр.....	41
3.1.14 Режим комбінованої бітової адресації (Kn адресація режим).....	41
3.1.15 Режим індикації (Z режим адресації).....	42
3.1.16 Комбінована індикація бітових рядків режим.....	43
3.1.17 Зберігання та адресація 32-бітних даних у D, R та V елементах.....	44
3.2 Дані.....	44
3.2.1 Тип даних.....	44
3.2.2 Кореляція між елементами та типами даних.....	44
3.2.3 Константа.....	45

3.1 Тип та функція програмних елементів

3.1.1 Короткий огляд програмних елементів

PLCs налаштовуються з використанням різноманітних віртуальних елементів у системному проектуванні для заміни реальних загальних реле, реле часу та інших пристроїв у схемах релейного керування. Ці віртуальні елементи разом називаються програмними елементами. PLCs використовують програмні елементи для роботи програми та налаштування функцій системи для реалізації всіх функцій керування та контролю. Завдяки своїй віртуальній природі програмні елементи можуть використовуватися в програмі багаторазово без теоретичного обмеження кількості (фактично це залежить від обсягу пам'яті програми). Крім того, програмні елементи не мають механічних або електричних проблем, притаманних реальним пристроям. Такі особливості роблять PLCs набагато надійнішими за схеми релейного керування. Крім того, програмувати та змінювати логіку стало простіше.

The types and functions of VC series PLC soft elements are shown in the following figure.



Рисунок 3-1 Типи та функції програмних

елементів У цьому посібнику програмні елементи названі відповідно до їхніх типів.

Наприклад:

- Точка вводу X, або "елемент X" для короткого
- Вихідна точка Y, або "елемент Y" для короткого
- Допоміжне реле M, або «елемент M» для скорочення
- Регістр даних D, або "елемент D" для скорочення
- Реле стану S, або "елемент S" для короткого позначення

3.1.2 М'який список елементів

Програмні елементи серії VC мікро-PLCs скопійовані та класифіковані відповідно до їхніх функцій. Різні елементи виконують різні функції та є легкодоступними.

На рисунку 3-1 наведено перелік елементів мікро-PLC серії VCsoft.

Рисунок 3-1 VC серія PLC список програмних елементів

	VC1S	VC1	VC2(Плануванн я)	VC3(Плануван ня)	VC5(Плануван ня)	Пронумеро ваний в

Програмні ресурси елементів В Примітка 4	I/O точки	36 входів 24 виходи (входи:X0–X47 виходи:Y0–Y27) Примітка 1	128 входів 128 виходів (входи:X0–X177; виходи:Y0–Y177) Примітка 1	256 входів 256 виходів (входи:X0–X377; виходи:Y0–Y377) Примітка 1	512 входів 512 виходів (входи:X0–X777; виходи:Y0–Y777) Примітка 1	512 входів 512 виходів (входи:X0–X777; виходи:Y0–Y777) Примітка 1	Вісімков а

		VC1S	VC1	VC2(Планування)	VC3(Планування)	VC5(Планування)	Пронумерований
	Допоміжне реле	2048 (M0–M2047)	2048 (M0–M2047)	2048 (M0–M2047)	10240 (M0–M10239)	10240 (M0–M10239)	Десятковий
	Локальний допоміжний реле ^{Примітка 5}	64 (LM0–LM63)	64 (LM0–LM63)	64 (LM0–LM63)	64 (LM0–LM63)	64 (LM0–LM63)	Десятковий
	Спеціальний допоміжне реле	512 (SM0–SM511)	512 (SM0–SM511)	1024 (SM0–SM1023)	1024 (SM0–SM1023)	1024 (SM0–SM1023)	Десятковий
	Стан реле	1024 (S0–S1023)	1024 (S0–S1023)	1024 (S0–S1023)	4096 (S0–S4095)	4096 (S0–S4095)	Десятковий
	Таймер	256 (T0–T255) ^{Примітка 2}	256 (T0–T255) ^{Примітка 2}	256 (T0–T255) ^{Примітка 2}	512 (T0–T511) ^{Примітка 2}	512 (T0–T511) ^{Примітка 2}	Десятковий
	Лічильник	256 (C0–C255) ^{Примітка 3}	264 (C0–C263) ^{Примітка 3}	256 (C0–C255) ^{Примітка 3}	307 (C0–C307) ^{Примітка 3}	307 (C0–C307) ^{Примітка 3}	Десятковий
	Регістр даних	8000(D0–D7999)	8000 (D0–D7999)	8000 (D0–D7999)	8000 (D0–D7999)	8000 (D0–D7999)	Десятковий
	Регістр даних R	/	/	/	32768 (R0–R32767)	32768 (R0–R32767)	Десятковий
	Локальні дані регістр ^{Примітка 5}	64 (V0–V63)	64 (V0–V63)	64 (V0–V63)	64 (V0–V63)	64 (V0–V63)	Десятковий
	Індексація зареєструвати	16 (Z0–Z15)	16 (Z0–Z15)	16 (Z0–Z15)	16 (Z0–Z15)	16 (Z0–Z15)	Десятковий
	Спеціальні дані зареєструвати	512 (SD0–SD511)	512 (SD0–SD511)	1024 (SD0–SD1023)	1024 (SD0–SD1024)	1024 (SD0–SD1024)	Десятковий

Примітки:

Примітка 1: Адреси елементів X та Y пронумеровані у вісімковій системі числення, наприклад, адреса X10 позначає точку входу 8th. Максимальне значення точок I/O тут є ємністю системи. Фактична кількість апаратних точок, що розширюються, повинна залежати від конфігурації системи PLC (включаючи доступні типи та кількість точок модулів розширення, обмеження потужності тощо).

Примітка 2: Т адреса елемента класифікується на три категорії залежно від точності синхронізації:

- Точність 100 мс: T0–T209
- точність 10 мс: T210–T251
- Точність 1 мс: T252–T255

Примітка 3: С адреса елемента класифікується на три категорії залежно від ширини та функції значень лічильника:

- 16-бітний лічильник інкрементів: C0–C199
- 32-бітний лічильник інкременту та декременту: C200–C235
- 32-бітний високошвидкісний лічильник: C236–C263

Примітка 4: Деякі внутрішні ресурси програмних елементів PLCs зарезервовані лише для внутрішнього використання. Вам слід, по можливості, не використовувати ці елементи в користувацькій програмі. Докладніше див. у Додатку С "Зарезервовані елементи".

Примітка 5: Ці два типи м'яких елементів є локальними змінними, які не можуть бути визначені в таблиці глобальних змінних. Коли користувацька програма викликає підпрограми або повертається до головної програми, вони скидаються до нуля для отримання значень параметрів або станів відповідно до функція передачі параметрів інтерфейсу функція.

3.1.3 точки входу та виходу

- Елемент мнемоніки
 - X елементи (дискретні вхідні точки)
 - Y елементи(дискретні вихідні точки)
- Функція

Елементи X та Y представляють відповідно стан входу апаратного терміналу X та стан виходу апаратного

терміналу Y.

The state of X elements is obtained through the input image register, while the state of Y elements is output through the output circuit driven by the output image register. The two operations are carried out in the I/O refresh phase of PLC scan cycle mode, as shown in Figure 3-2. For details, refer to section 2.1.2 "PLC operating mechanism (Scan cycle model)". It is obvious that there is a brief delay in PLC's response to the I/O. The delay is related to the input filter, communication, housekeeping, and scan cycle.

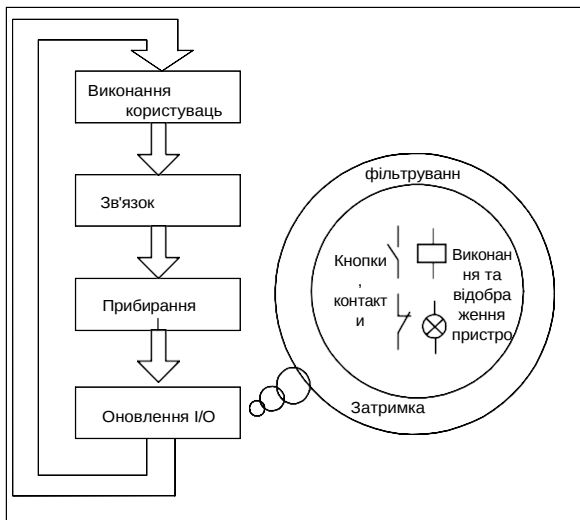


Рисунок 3-2 Схематична діаграма I/O оновлення

- **Класифікація**
Вихідний канал, що відповідає X елементам: X0–X17 налаштовані з функцією цифрової фільтрації. Ви можете встановити час фільтрації через системний блок. Інші X вхідні точки використовують апаратну фільтрацію. X0–X7 можуть використовуватися як вхідні точки підрахунку для програмних елементів високошвидкісних лічильників. Крім того, X0–X7 також можуть використовуватися як вхідні клеми для зовнішнього переривання, захоплення імпульсів та інструкції SPD.
Y елементи використовуються як високошвидкісні вихідні клеми або клеми загального призначення.
- **Елементи, що стосуються режиму**
Вісімкова система, починається з 0. Елементи X та Y головного модуля та модулі розширення I/O адресуються послідовно. Елементи X адресуються послідовно в X0–X7, X10–X17, X20–X27 тощо. Елементи Y адресуються послідовно в Y0–Y7, Y10–Y17, Y20–Y27 тощо.
- **Тип даних**
Обидва елементи X та Y є логічними (значення елементів — ON або OFF).
- **Доступні форми**
Ви можете використовувати NO та NC контакти елементів X під час програмування (залежно від двох інструкцій). Контакти NO та NC мають протилежні значення стану. Їх іноді називають "а-контакт" та "b-контакт".
Ви також можете використовувати контакти NO та NC елементів Y під час програмування.
- **Присвоєння значення**
 1. X елементи приймають лише значення стану апаратного входу та стану примусової роботи, які неможливо змінити за допомогою вихідних або налаштувальних інструкцій, а також неможливо встановити під час введення системи в експлуатацію.
 2. Ви можете призначати значення стану елементам Y за допомогою OUT instruction, встановлювати значення стану елементів Y або навіть примусово задавати чи записувати значення стану елементів Y під час введення системи в експлуатацію.
 3. Через системний блок ви можете встановити вихідні стани елементів Y у стані STOP.

3.1.4 Допоміжні реле

- **Елемент**
мнемоніки M
елементи
- **Функція**
Система надає вам тип дискретних елементів стану,

що подібний до проміжного реле в реальному

електричне коло керування. Ви можете використовувати їх для збереження різних проміжних станів у користувацькій програмі.

- Режим адресації елементів вісімкові, починаються з 0.
- Тип даних

M elements are Boolean (the element values are ON or OFF).

Доступні форми

NO та NC контакти

Присвоєння значення

- Через операції з інструкціями.
- Примусове встановлення або запис значень стану під час введення системи в експлуатацію.

Збереження даних при зникненні живлення

Стан	М елементи, для яких дані зберігаються при вимкненні живлення	М елементи, для яких дані не зберігаються при вимкненні живлення
		відключення

3.1.5 Стан реле

Елемент мнемоніки S елементи

Псевдонім прапорця кроку

Функція

Як прапорець кроку використовуються елементи S у SFC.

Детальнішу інформацію див. у розділі 7 «Посібник користувача SFC».

Класифікація

S0–S19: прапорець початкового

кроку Інші: прапорець загального кроку

Режими адресації елементів десяткові, починаючи з 0

Тип даних

Булеві елементи (значення елементів — ON або OFF).

Доступні форми

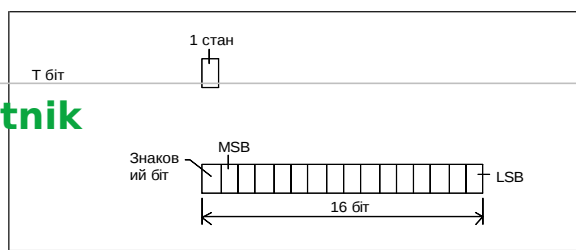
- Відображення стану кроку (коли інструкцію STL запрограмовано через SFC)

3.1.6 Таймер

Елемент мнемоніки T елементи

Функція

T element is a composite soft element that contains a word element (2 bytes) and a bit element. The T word element records a 16-bit timing value that can be used as a value in the program. The T bit element reflects the state of the timer coil, which can be used for logic



Зникнення живлення	Дані збережено	Дата видалена
RUN → STOP	Дані збережено	Дата видалена
STOP → RUN	Без змін	Очищено
Примітка: Діапазон збереження при зникненні живлення встановлюється через системний блок. Детальніше див. у розділі 2.2.1 "Система"		

Примітка

Під час використання протоколу N:N, системою викликаються деякі елементи M. Вам необхідно звернути на це увагу під час програмування та внесення змін до програми.

2. Контакти NO та NC (коли інструкція STL не запрограмована для SFC). Подібно до характеристик елементів M, контакти NO та NC елементів S можуть бути використані для програмування.

Присвоєння значення

- Через операції з інструкціями.
- Примусове встановлення або запис значень стану під час введення системи в експлуатацію.

Безперервна робота після зникнення живлення

Стан	S елементи, для яких дані зберігаються при вимкненні живлення	S елементи, для яких дані не зберігаються при вимкненні живлення
Зникнення живлення	Без змін	Очищено
RUN → STOP	Без змін	Без змін
STOP → RUN	Без змін	Очищено
Примітка: Діапазон збереження під час зникнення живлення встановлюється через системний блок. Детальнішу інформацію див. у розділі 2.2.1 "Система"		

блок

control.

Рисунок 3-3 T елементи

Класифікація

Відповідно до точності синхронізації, T елементи є

класифіковано на три типи. У наступній таблиці описано елементи T та відповідну точність синхронізації в різних сегментах адреси. Вам слід звернути на це увагу під час використання.

T елемент	Точність синхронізації
T0–T209	100 мс точність
T210–T251	10 мс точність
T252–T255	1 мс точність

Для елементів T з точністю таймінгу 1 мс, вони активуються за допомогою переривання і не залежать від циклу сканування PLC. Тому час їхньої дії є найбільш точним. Для елементів T з точністю таймінгу 10 мс та 100 мс, оновлення та час дії їхніх значень таймінгу залежать від циклу сканування PLC.

- Режими адресації елементів десяткові, починаючи з 0
- Тип даних Булеве значення (значення елементів ON або OFF), та слово.

- Available forms

Часові параметри та режими роботи Т елементів визначаються часовими інструкціями. Існує чотири типи часових інструкцій, включаючи TON, TOF, TONR та TMON інструкції. Детальнішу інформацію див. у розділі 5 "Основні інструкції".

- Присвоєння значення
 - Через операції з інструкціями.
 - Примусове встановлення або запис значень стану під час введення системи в експлуатацію.
- Безперервний вихідний сигнал після збою живлення

Стан	Т елементи для яких дані збережено у	Т елементи для які дані не є

3.1.7 Лічильник

- Елемент мнемоніки C елементи
- Функція

А C елемент — це композитний програмний елемент, що містить бітовий елемент та елемент слова або подвійного слова (2 або 4 байти). C елемент слова записує 16-бітне або 32-бітне значення часу, яке можна використовувати як значення в програмі, тоді як C бітовий елемент відображає стан котушки таймера для логічного керування.

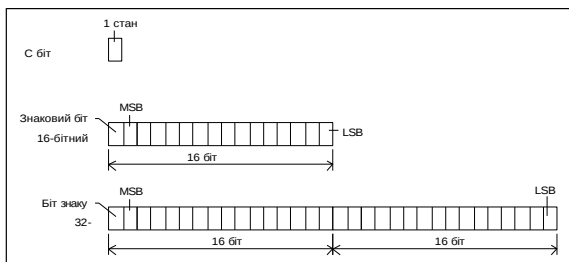


Рисунок 3-4 C елементи

- Класифікація

Два типи: 16-бітний лічильник та 32-бітний лічильник
- Режими адресації елементів десяткові, починаючи з 0
- Тип даних

Булеве значення (значення елемента ON або OFF), одне слово або два слова.
- Доступні форми

Інструкції підрахунку, що викликають елементи C, класифікуються на 4 типи, а саме: 16-бітний інкрементний підрахунок

3.1.8 Регістр даних

- Мнемоніка

	зниження напруги (для VC2/3/5series тільки)	збережено при живленні відключення
Зниження живлення	Без змін	Очищено
RUN → STOP	Без змін	Без змін
STOP → RUN	Без змін	Очищено

Примітка: Діапазон збереження під час зниження живлення встановлюється через the

□ Примітка

Максимальне значення часу елемента Т становить 32767. Попередньо встановлене значення становить від -32768 до -32767. Оскільки елемент Т діє лише тоді, коли значення лічильника досягає або перевищує попередньо встановлене значення, встановлення попередньо встановленого значення як від'ємного числа фактично не має сенсу.

інструкція, 16-бітна інструкція циклічного підрахунку, 32-бітна інструкція підрахунку інкременту та декременту, а також високошвидкісна інструкція I/O. Детальнішу інформацію див. у розділі 5 "Базові інструкції" та розділі 6 "Прикладні інструкції".

The classification of C elements is shown in the table below:

C елементи	Функція рахування	Застосовна інструкція типу
C0-C199	16-бітний інкрементний лічильник	16-бітна інструкція інкрементного підрахунку 16-бітний циклічний підрахунок інструкція
C200-C235	32-бітний лічильник інкременту та декременту	32-бітна інструкція інкременту та декременту

- Присвоєння значення
 - Через операції з інструкціями.
 - Примусово або записати елемента D та елементи R
- Функція

Як елементи даних, елементи D або R використовуються як операнди в багатьох інструкціях обчислення та керування.

значення стану під час введення системи в експлуатацію.

- Безперервна робота після зникнення живлення

Стан	С елементи для яких дані зберігаються при зникненні живлення	С елементи для яких дані не зберігаються під час зникнення живлення
Зникнення живлення	Без змін	Очищено
RUN → STOP	Без змін	Без змін
STOP → RUN	Без змін	Очищено
Примітка: Діапазон збереження при зникненні живлення встановлюється через системний блок. Детальніше див. у розділі 2.2.1 "Система блок"		

- Режими адресації елементів десяткові, починаючи з 0
- Тип даних
Кожен елемент D або R є 16-бітним регістром, який може зберігати 16-бітні дані, наприклад 16-бітне ціле число.

Two D or R elements can form a double word element that can store 32-bit data, such as long integer data or floating-point data.

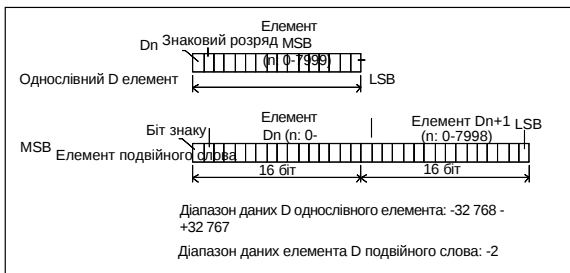


Рисунок 3-5 D або R елементи

Примітка

У D або R елементі подвійного слова, старші 16 біт знаходяться у першому D або R елементі, а молодші 16 біт — у другому D або R елементі.

- Доступні форми
Елементи D або R використовуються як операнди в багатьох інструкціях обчислення та керування.

- Присвоєння значення
1. Ініціалізація блоку даних. 2. Через інструкції операцій. 3. Примусове встановлення або запис значень стану під час введення системи в експлуатацію.
- Безперервна робота після зникнення живлення

Стан	D елементи, для яких зберігаються дані під час вимкнення живлення	D елементи, для яких дані не зберігаються при вимкненні живлення
Зникнення живлення	Без змін	Очищено
RUN → STOP	Без змін	Без змін
STOP → RUN	Без змін	Очищено

Примітка: Діапазон збереження при зникненні живлення встановлюється через системний блок. Детальнішу інформацію див. у розділі 2.2.1 "Системний блок".
R елементи не можуть бути збережені при вимкненні живлення

Примітка

Деякі D елементи викликаються системою, коли застосовується інструкція привода або протокол N:N. Вам потрібно звернути на це увагу під час програмування та модифікації програми.

3.1.9 Спеціальне допоміжне реле

- Мнемоніка елемента
SMelements
- Функція
SM елементи тісно пов'язані з функцією PLCsystem, відображаючи системну функцію та стан PLCs. Детальнішу інформацію див. у Додатку А "Спеціальні допоміжні реле" у цьому посібнику.
- Класифікація
Часто використовувані SM елементи включають:
 - SM0: біт монітора роботи PLC. Він ON, коли PLC перебуває у стані RUN.
 - SM1: біт імпульсу початкової роботи. Він є ON під час циклу першого сканування роботи PLC.
 - SM3: системна помилка. Це ON, якщо будь-яка системна помилка виявляється після ввімкнення живлення PLC або коли PLC переходить з STOP у RUN.
 - SM10–SM12: відповідно прямокутний тактовий сигнал із періодом 10 мс, 100 мс та 1 с (змінює стан двічі за цикл).

3.1.10 Спеціальний регістр даних

- Елемент

мнемоніки SD
елементи

- Функція
Елементи SD тісно пов'язані з функцією системи PLC, відображаючи параметри функції системи, значення кодів стану та дані операцій інструкцій PLCs. Детальнішу інформацію див. у Додатку В "Спеціальний регістр даних" у цьому посібнику.

- SM25 – SM71: Біт прапорця керування перериванням. Встановлення цих SM елементів активує відповідні функції переривання.

- Режим адресації елементів десятковий, починається з 0.
- Тип даних Булеві елементи (значення елементів — ON або OFF).
- Доступні форми NO та NC контакти
- Присвоєння значення 1. Через операції з інструкціями. 2. Примусове встановлення або запис значень стану під час введення системи в експлуатацію.

□ Примітка

Ви не можете присвоювати значення елементам SD лише для читання.

- Режим адресації елементів десятковий, починається з 0.
- Тип даних Слово та подвійне слово (ціле число) елементи.
- Доступні форми Зберігання та обчислення цілих чисел
- Присвоєння значення

1. Через операції з інструкціями. 2. Примусове встановлення або запис значень стану під час введення системи в експлуатацію.

□ Примітка

Ви не можете присвоювати значення SD елементам, доступним лише для читання.

3.1.11 Індексція реєстру

- Елемент мнемоніки Z елементи
- Функція 16-бітні елементи реєстрів можуть використовуватися для зберігання символічних цілочисельних даних. Щодо деталей індексції, зверніться до розділу 3.1.15 "Режим індексції (Z адресація Режим)".
- Режим адресації елементів десятковий, починається з 0.

- Елементи типу даних Word
- Доступні форми Елементи Z використовуються для індексції. Вам потрібно записати зміщення адреси в елементи Z перед їх використанням.
- Присвоєння значення 1. Через операції з інструкціями. 2. Примусове встановлення або запис значень стану під час введення системи в експлуатацію.

3.1.12 Локальне допоміжне реле

- Елемент мнемоніки LM елементи
- Функція LM елементи — це локальні варіанти, які можуть застосовуватися в головній програмі та підпрограмах. Це змінні елементи, які є локально дійсними в кожному окремому тілі програми (головна програма, підпрограма та програма переривання). Тому неможливо безпосередньо спільно використовувати стан будь-якого LM елемента між різними тілами програм. Коли система залишає тіло програми під час виконання користувацької програми, LM елемент перевизначається. При поверненні до головної програми або виклику підпрограми значення перевизначеного LM елемента видаляється або відповідний стан має бути отриманий згідно з функцією передачі параметрів інтерфейсу.

Елементи LM можна використовувати для визначення параметрів інтерфейсу підпрограм з метою реалізації функції передачі параметрів інтерфейсу. Детальніше див. у розділі

4.4 "Підпрограма".

- Режим адресації елементів десятковий, починається з 0.
- Тип даних Булеве значення (значення елементів є ON або OFF).
- Доступні форми NO та NC контакти
- Присвоєння значення 1. Через інструкцію операцій.

3.1.13 Локальний регістр даних

- Елемент мнемоніки V елементи
- Функція V елементи — це локальні варіанти, які можна застосовувати в головній програмі та підпрограмах. Це змінні елементи, які є локально дійсними в кожному окремому тілі програми (головній програмі та підпрограмі). Тому неможливо безпосередньо спільно використовувати дані будь-якого V елемента між різними тілами програм. Коли система залишає тіло програми під час виконання користувацької

програми, V елемент перевизначається. При поверненні до головної програми або виклику підпрограми значення перевизначеного V елемента видаляється або відповідні дані мають бути отримані відповідно до функції передачі параметрів інтерфейсу.

V елементи можуть бути використані для визначення параметрів інтерфейсу підпрограм для реалізації функції передачі параметрів інтерфейсу. Детальніше див. у розділі 4.4 "Підпрограма".

- Режим адресації елементів десятковий, починається з 0.
- Тип даних Булеві елементи (значення елементів — ON або OFF).
- Доступні форми Елемент слова, який можна використовувати для зберігання числової інформації
- Присвоєння значення
 1. Через інструкцію операцій.

3.1.14 Режим комбінованої бітової адресації (Kn адресація режим)

- Концепція

Режим адресації бітових рядків, або режим адресації Kn, об'єднує рядки бітових елементів у слова або подвійні слова.

- Метод

Формат такий: "K (n) (U)", де "n" — це ціле число від одного до восьми, що вказує на те, що довжина бітового рядка становить $n \times 4$ біти. U вказує на початкову адресу бітового елемента рядка елементів.

Застосунок екземпляр:

1. K1X0 вказує на слово, що складається з бітового рядка з 4 бітових елементів (X0, X1, X2 та X3), починаючи з X0.
2. K3Y0 вказує на слово, що складається з бітового рядка з 12 бітових елементів (Y0, Y01, Y02 та Y03), (Y04, Y05, Y06 та Y07) і (Y10, Y11, Y12 та Y13), починаючи з Y0.
3. K4M0 вказує на слово, що складається з бітового рядка з 16 бітових елементів: M0, M1, M2, M3... та M15.
4. K8M0 вказує на подвійне слово, що складається з бітового рядка з 32 бітових елементів: M0, M1, M2, M3... та M31.

- Формат зберігання даних у режимі адресації Kn режим

Наступний приклад описує, як дані зберігаються в режимі адресації Kn:

MOV 2#10001001 K2M0 (що є еквівалентом MOV 16#89 K2M0, або MOV 137 K2M0). Після виконання інструкції, K2M0 зберігається. У наступній таблиці описано конкретний формат зберігання K2M0.

Дані	MSB	Проміжний біт						LSB
K2M0	M7	M6	M5	M4	M3	M2	M1	M0
16#89	1	0	0	0	1	0	0	1

- Примітка щодо комбінованої адресації бітових рядків режиму

Якщо операнд призначення інструкції використовує режим адресації Kn, тоді як ширина даних, що зберігаються в операнді призначення, перевищує ширину, визначену в режимі адресації Kn, система зберігає LSB та відкидає MSB даних.

Для прикладу:

Виконання інструкції "DBITS 16# FFFFFFFF0 K1M0".

Після виконання інструкції операнд 2 (K1M0) має зберігати результат обчислення 16#1с (28). Однак K1M0 має ширину лише 4 біти, чого недостатньо для 16#1с. Шляхом відкидання старших бітів фактичний результат обчислення операнда 2 становить K1M0=16#с (12).

3.1.15 Режим індексації (Z режим адресації)

- Концепція індексації

Серія VC PLCs забезпечує режим індексації (режим адресації Z). Ви можете використовувати елементи Z (регістри індексації) для отримання непрямого доступу до цільових елементів.

- Z метод адресації:

Цільова адреса=Адреса базового елемента+Зміщення адреси, що

зберігається в елементі Z. Наприклад:

У режимі індексації для D0Z0 (де Z0=3) цільовою адресою є D3. D0 — це базова адреса, а адреса зміщення, що зберігається в елементі Z0, дорівнює 3.

Отже, коли Z0=3, інструкція "MOV 45 D0Z0" за результатом дорівнює "MOV 45 D3", і в обох випадках D3 отримує значення 45 після виконання інструкції.

- Індексація прикладу

1. Приклад індексації бітового

елемента LD M01

MOV 6 Z1

SFTR X0Z1 M0 8 2

Попередні інструкції рівнозначні: LD M01

SFTR X6 M0 8 2

Процес адресації є таким:

Z1=6

$X0Z1 = X (0 + Z1) = X6$

2. Приклад індексації елемента

слова LD M01

MOV 30 Z20

MOV D100Z20 D0

Попередні інструкції рівнозначні: LD M01

MOV D130 D0

Процес адресації виглядає наступним

чином: $Z20=30$

$D100 Z20 = D (100 + Z20) = D130$

● Примітки до режиму індексації

1. У режимі індексації (Z режим адресації), Z елементи зберігають зміщення адреси. Вони підтримують цілі числа зі знаком, що означає підтримку від'ємного зміщення.

Для прикладу:

MOV -30 Z20

MOV D100Z20 D0

Попередні інструкції є рівнозначними до:

MOV D70 D0

2. Елементи SM та SD не підтримують індексацію режиму.

3. Вам необхідно звертати увагу на діапазон адрес при використанні режиму адресації Z. Наприклад, D7999Z0 (Z0=9) знаходиться поза межами діапазону адрес елементів D (макс. адреса елементів D становить 7999).

3.1.16 Комбінована індексація бітових рядків режим

Режим комбінованої адресації бітових рядків можна використовувати разом із режимом індексації. Наприклад: K1X0Z10.

У цьому режимі вам потрібно визначити адресу елемента початкового біта через Z адресацію, а також визначити довжину бітового рядка через Kп адресацію.

Наприклад:

LD M1

MOV 3 Z10

MOV K1X0Z10 D0

Попередні інструкції рівнозначні: LD M1

MOV K1X3 D0

Процес адресації наступний: $Z10 = 3$

$K1X0Z10 = K1X (0 + Z10) = K1X3$

3.1.17 Зберігання та адресація 32-бітних даних у D, R та V елементах

- Зберігання 32-бітних даних у D, R та V елементах

Дані типу DINT, DINT та REAL є 32-бітними, а елемент D, R або V має лише 16 біт. Для зберігання 32-бітних даних потрібно два послідовні елементи D, R або V.

Серія VC PLCs зберігає 32-бітні дані у режимі Big Endian, тобто елементи з меншими адресами використовуються для зберігання старших бітів 32-бітних даних, тоді як елементи з більшими адресами використовуються для зберігання молодших бітів 32-бітних VCданих.VC

Для екземпляра Application, беззнакове довге ціле число "16# FEA8_67DA" зберігається в елементі (D0, D1). Фактичний формат зберігання наступний:

D0	0xFE A8
D1	0x 67 DA

- Адресація 32-бітних даних у D, R та V елементах

Ви можете використовувати елемент D або V для пошуку 16-бітних даних, таких як дані типу INT або WORD, або 32-бітних даних, таких як дані DINT або DINT.

Якщо в операнді інструкції використовується адреса елемента D, R або V, тип даних операнда визначає, чи є дані 16-бітними чи 32-бітними.

Для прикладу:

В інструкції "MOV 16#34 D0" адреса D0 вказує на один елемент D0, оскільки операнд 2 інструкції MOV має тип даних WORD.

В інструкції "DMOV 16# FEA867DA D0" адреса D0 вказує на два послідовні слова: D0 та D1, оскільки операнд 2 інструкції DMOV належить до типу даних DINT.

3.2 Дані

3.2.1 Тип даних

Усі операнди інструкцій належать до певного типу даних. Усього існує шість типів даних, як зазначено в наступній таблиці.

Таблиця 3-2 Типи даних операндів

Тип даних	Опис типу	Ширина даних	Діапазон
BOOL	Біт	1	ON, OFF (1, 0)
INT	Знакове ціле число	16	−32768–+32767
DINT	Знакове довге ціле число	32	−2147483648–+2147483647
REAL	Рухома-кома номер	32	±1.175494E-38–±3.402823E+38

3.2.2 Кореляція між елементами та типами даних

Типи елементів, що використовуються операндами інструкцій, повинні відповідати їхнім типам даних. Кореляції між застосовними елементами та типами даних наведено в наступній таблиці.

Таблиця 3-3 Відповідність між елементами та типами даних

Тип даних	Програмні елементи													
BOOL	X	Y	M	S	LM	SM				C	T			
	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R
DINT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R

REAL	Константа								D				V		R

Тип даних	Програмні елементи													

Якщо інструкція використовує операнд із невідповідним типом даних, інструкція вважається недопустимою. Наприклад, інструкція "MOV 10 X0" є недопустимою, оскільки операнд 2 інструкції MOV має тип цілого числа зі знаком, тоді як елемент X0 може зберігати лише логічні дані.

□ Примітка

1. Коли операнд належить до типу INT, застосовні елементи включають KnX, KnY, KnM, KnS, KnLM та KnSM, де $1 \leq n \leq 4$.
2. Коли операнд має тип DINT, застосовні елементи включають KnX, KnY, KnM, KnS, KnLM та KnSM, де $5 \leq n \leq 8$.
3. Коли операнд має тип INT, застосовними елементами C є C0–C199.
4. Коли операнд має тип DINT, застосовними елементами C є C200–C263

3.2.3 Константа

Ви можете використовувати константи як операнди інструкцій. Серія VC PLCs підтримує різноманітні режими введення констант. Вирази констант наведено в наступній таблиці.

Таблиця 3-4 Вирази констант

Постійний тип	Застосування екземпляр	Діапазон значень	Примітки
Десяткова константа (16-бітне знакове ціле число)	–8949	–32768–+32767	
Десяткова константа (32-бітне знакове ціле число)	–2147483646	–2147483648–+2147483647	
Шістнадцяткова константа (16-бітна)	16#1FE9	16#0–16#FFFF	Шістнадцяткові, вісімкові та двійкові константи не мають додатного чи від'ємного значення. Якщо ви обираєте шістнадцяткові, вісімкові або двійкові константи, що використовуються як операнди, додатні та від'ємні значення цих констант, а також їхні розміри визначаються типами даних цих операнди
Шістнадцяткова константа (32-бітна)	16#FD1EAFE9	16#0–16#FFFFFFFF	
Вісімкова константа (16-біт)	8#7173	8#0–8#177777	
Вісімкова константа (32-біт)	8#71732	8#0–8#3777777777	
Бінарна константа (16-біт)	2#10111001	2#0–2#1111111111111111	
Бінарна константа (32-біт)	2#101110011111	2#0–2#11111111111111111111111111111111	
Константи з рухомою комою одинарної точності constants	–3.1415E-16 3.1415E + 3 0.016	±1.175494E-38–±3.402823E + 38	Відповідність IEEE-754 стандарту. Програмне забезпечення для програмування може відображати та вхідні константи з плаваючою комою з 7-бітна ефективна точність.

Розділ 4 Концепція програмування

У цьому розділі детально описано вміст програмування мікро-PLC серії VC, включаючи мову програмування та елементи програми. Також представлено програмування та використання підпрограм, а наприкінці наведено загальні пояснення до інструкцій.

Розділ 4 Концепція програмування.....	46
4.1 Вступ до мов програмування.....	47
4.1.1 LAD.....	47
4.1.2 IL.....	48

4.1.3 SFC.....	48
4.2 Програмування Елементів.....	49
4.2.1 Програма.....	49
4.2.2 Системний блок.....	50
4.2.3 Блок даних.....	50
4.3 Блоковий коментар та коментар змінної програми.....	50
4.3.1 Блок коментаря.....	50
4.3.2 Змінна Коментар.....	51
4.4 Підпрограма.....	52
4.4.1 Концепція.....	52
4.4.2 Примітка щодо використання SBRs.....	52
4.4.3 Визначення змінної SBR таблиця.....	53
4.4.4 SBR параметр передачі.....	54
4.4.5 SBR екземпляр застосунок.....	54
4.5 Загальна інформація інструкцій.....	56
4.5.1 операнди.....	56
4.5.2 Прапорець біта.....	56
4.5.3 Обмеження щодо використання інструкцій.....	56

4.1 Вступ до мов програмування

Існує три мови програмування: LAD, IL та SFC.

4.1.1 LAD

- Концепції

LAD — це широко вживана графічна мова програмування PLC, подібна до електричної (релейної) схеми керування. Її основні особливості включають:

1. Він налаштований з лівою шиною, а права шина опущена.
2. Усі елементи керуючого виходу (котушки) та функціональні блоки (прикладні інструкції) мають спільний вхідний термінал потоку енергії.

Електрична схема керування та LAD є еквівалентними певною мірою, як показано на наступному рисунку.

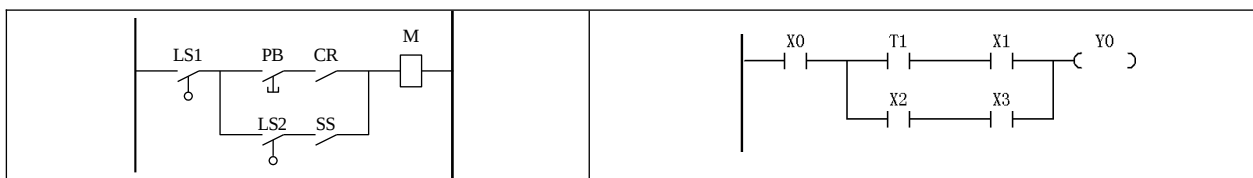


Рисунок 4-1 Еквівалентність між схемою електричного керування та LAD

- LAD основні елементи програмування

Відповідно до принципів, викладених у схемі електричного (релейного) керування, для LAD: абстраговано кілька базових елементів програмування

1. Ліва шина: відповідає шині керування на схемі електричного керування та забезпечує живлення керування для кола керування.
2. Лінія підключення (—|): Це позначає електричне з'єднання на схемі електричного керування та використовується для підключення інших взаємопов'язаних елементів.
3. Контакт (—|): Він представляє вхідний контакт на схемі електричного керування, він керує ON/OFF та напрямком керуючих струмів у колі. Паралельне та послідовне з'єднання контактів по суті відображає операційний взаємозв'язок вхідної логіки кола керування та контролює передачу потоку енергії.

4. Котушка (🔌): Він представляє релейний вихід на електричній схемі керування.

5. Функціональний блок (🔧): Вона також відома як інструкція застосування і відповідає виконавчому механізму або функціональному пристрою, що виконує спеціальні функції на схемі електричного керування. Вона може виконувати специфічні функції керування або функції обчислення керування (такі як передача даних, обчислення даних, таймер, лічильник тощо).

- Потік енергії

Будучи важливим поняттям у LAD, потік енергії використовується для живлення елементів котушки та інструкцій застосування, що схоже на керуючий струм, який видається котушкою приводу, і виконується виконавчим блоком на схемі електричного керування.

У LAD котушки або вхідна частина інструкції застосунку мають бути підключені до потоку енергії. Елемент котушки може бути вихідним, а інструкції застосунку можуть бути ефективно виконані лише тоді, коли потік енергії є дійсним.

На наступному рисунку показано передачу потоку енергії в LAD та привід потоку енергії до котушок або функціональних блоків.

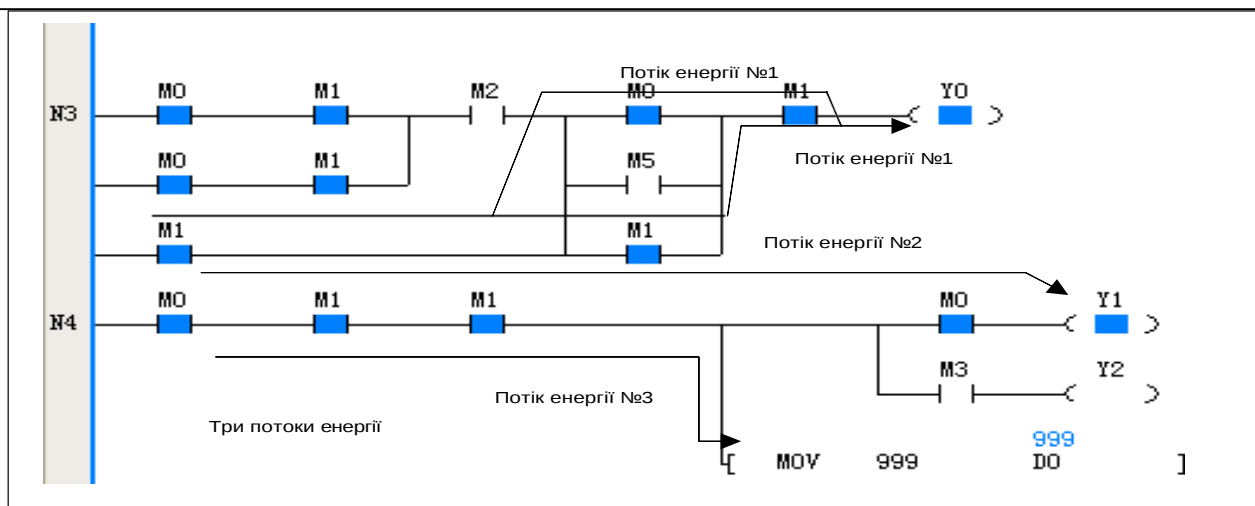


Рисунок 4-2 Потік енергії та його функція приводу

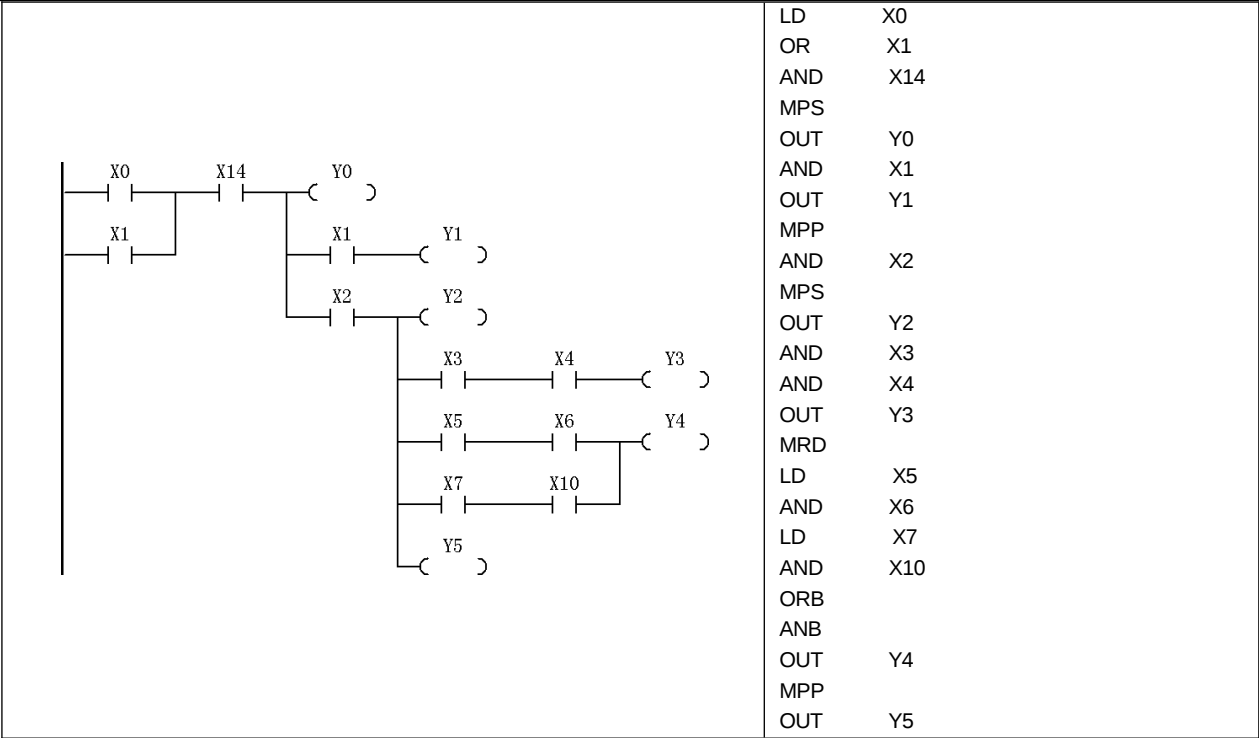
4.1.2 IL

IL — це текстова користувацька програма або набір послідовностей інструкцій, написаних користувачами.

Користувацька програма, що зберігається в PLC головному модулі для виконання, фактично є послідовністю інструкцій, розпізнаваною головним модулем, і система виконує кожну інструкцію в послідовності одну за одною для реалізації функції керування користувацькою програмою.

Наступний рисунок є прикладом застосування перетворення LAD на IL.

LAD	IL



4.1.3 SFC

SFC — це графічна мова проектування для структури користувацької програми, яка зазвичай використовується для реалізації функцій послідовного керування.

Послідовне керування — це процес керування, який можна розділити на кілька процедур (етапів обробки) і виконувати їх згідно з певною робочою послідовністю.

Користувацька програма, розроблена за допомогою SFC, є відносно простою та зрозумілою, оскільки її структура відповідає фактичному процесу послідовного керування.

Наступний рисунок є прикладом застосування простого SFC.

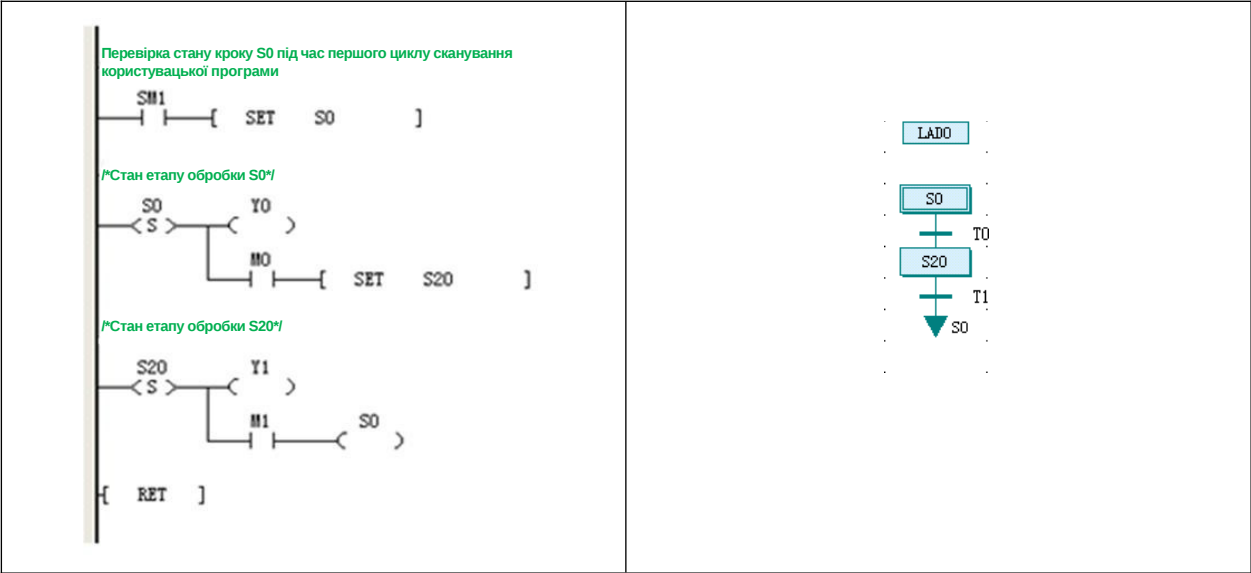


Рисунок 4-3 Приклад застосування SFC

4.2 Програмування Елементів

Елементи програми включають користувацьку програму, системний блок та блок даних. Ви можете змінювати ці елементи програми за допомогою програмування.

4.2.1 Програма

Користувацька програма — це програмний код, написаний користувачем. Він компілюється у послідовність виконуваних інструкцій, завантажується у контролер, і контролер виконує функцію керування користувацької програми.

Користувацька програма складається з трьох програмних організаційних одиниць (POU): головної програми (MAIN), підпрограми (SBR) та програми переривання (INT).

- MAIN

Головна програма є основним тілом та каркасом користувацької програми. Коли система перебуває в RUN, головна програма виконується циклічно.

Одна користувацька програма має лише одну головну програму.

- SBR

А Підпрограма — це структурно та функціонально незалежна користувацька програма, яку можуть викликати інші програмні блоки. Підпрограми зазвичай мають інтерфейси операндів виклику, які виконуються лише під час виклику.

А Користувацька програма може мати довільну кількість підпрограм або не мати жодної підпрограми.

- INT

Програма переривання — це сегмент користувацької програми, що обробляє конкретну подію переривання. А конкретна подія переривання завжди відповідає конкретній програмі переривання.

У разі виникнення події переривання звичайний цикл сканування переривається, і потік користувацької програми автоматично переходить до виконання програми переривання, доки інструкція повернення з переривання не поверне систему до звичайного циклу сканування.

А Користувацька програма може мати довільну кількість програм переривань або не мати жодної програми переривань взагалі.

4.2.2 Системний блок

Системний блок містить декілька опцій конфігурації системи. Ви можете змінювати, компілювати та завантажувати системний блок для налаштування режиму роботи головного модуля.

Детальну інформацію щодо використання елементів конфігурації системи див. у розділі 2.2.1 "Системний блок" посібника або у відповідному описі в *Посібнику користувача програмного забезпечення Auto Studio Programming*.

4.2.3 Блок даних

Блок даних містить задане значення елемента D або R. Коли блок даних завантажується в контролер, призначеному елементу D або R присвоюється задане значення, завдяки чому досягається мета групового встановлення значення елемента D або R.

Якщо параметр "Datablock enabled" у вкладці розширених налаштувань системного блоку активовано, елементи D або R ініціалізуються блоком даних до того, як користувацька програма перейде у стан RUN.

4.3 Блоковий коментар та коментар змінної програми

4.3.1 Блок коментаря

Під час програмування ви можете додавати блокові коментарі до програми. Блокові коментарі текстово описують відповідні програми. Кожен блоковий коментар займає цілий рядок простору.

У програмі натисніть праву кнопку миші та виберіть **Вставити Рядок**, щоб вставити рядок над поточним рядком. Ви можете використовувати порожній рядок для розділення двох програмних блоків.

Щоб створити блочний коментар, потрібно спочатку вибрати порожній рядок, потім натиснути праву кнопку миші та вибрати **Вставити блочний коментар**, як показано нижче.

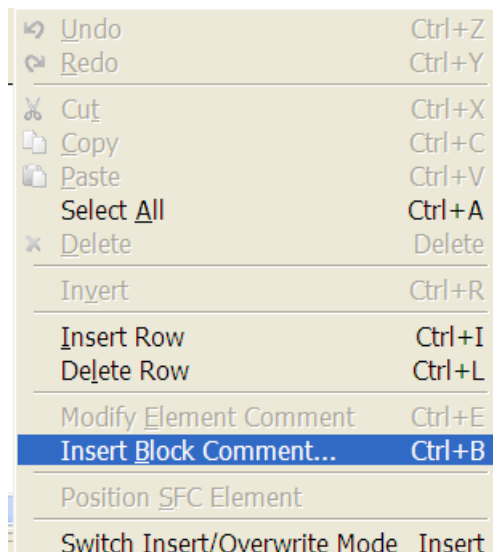


Рисунок 4-4 Додавання блоку коментаря

Введення вашого коментаря у діалогове вікно коментарів, що з'являється, та натискання ОК кнопки.

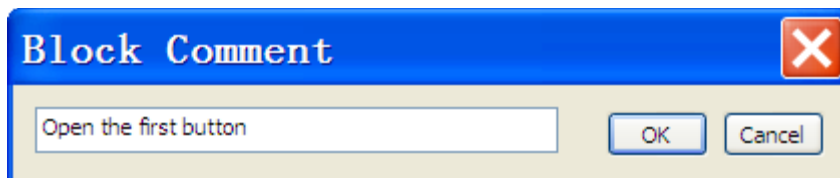


Рисунок 4-5 Діалогове вікно коментаря блоку

Програмне забезпечення автоматично додає "/" та "*" з обох боків введеного тексту і відображає їх зеленим кольором, як показано нижче:

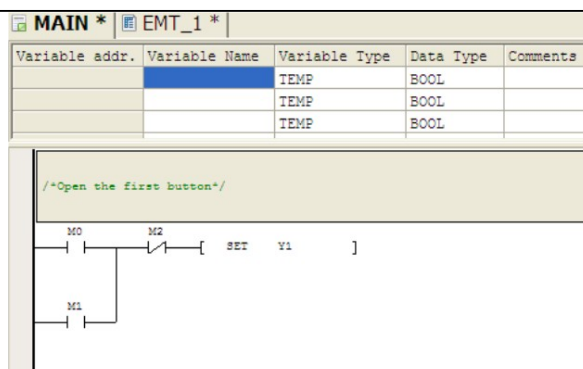


Рисунок 4-6 Коментар до годинника в програмі

А Блочний коментар займає цілий рядок. Ви не можете додати блокувальний коментар до зайнятого рядка, так само як і рядок, зайнятий коментарем, не може бути використаний для інших цілей.

4.3.2 Змінна Коментар

Ви можете визначити змінні у таблиці глобальних змінних та таблиці локальних змінних. (Детальніше див. розділ 2.2.3 "Таблиця глобальних змінних" та розділ 4.4.3 "Визначення таблиці SBR змінних"), і правильно визначені змінні можна використовувати у LAD. Змінна A може означати

певну адресу для підвищення читабельності програми. На рисунку 4-7 показано деякі змінні, визначені в глобальній таблиці

ЗМІННИХ.

	Variable Name	Variable addr.	Comments
1	StopButton	X0	Stop Button
2	Timer0	T0	Timer
3	Start1	M0	
4	Start2	M1	
5	Stop1	M2	
6	Switch1	Y1	Control Switch

Рисунок 4-7 Змінні, визначені у глобальній таблиці змінних

- Символ адресації

Коли визначені змінні використовуються, ви можете перемикатися між назвою змінної та адресою елемента, вибравши меню **Символьна адресація**.

На наступному рисунку показано той самий LAD в обох режимах відображення.

На наступному рисунку показано стан LAD, коли символічна адресація не вибрана.

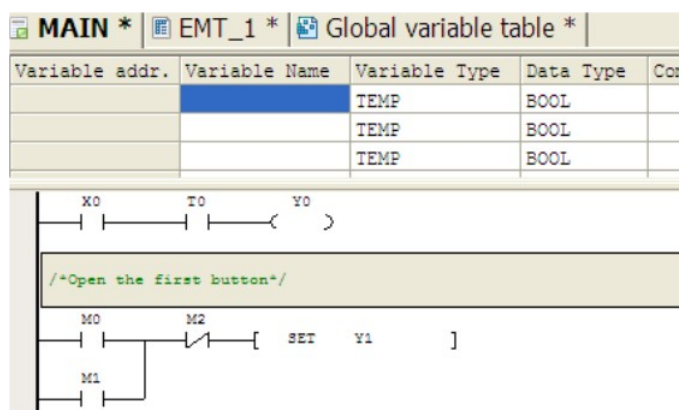


Рисунок 4-8 Стан символічної адресації не вибрано

на наступному рисунку показано стан LAD, коли вибрано символічну адресацію.

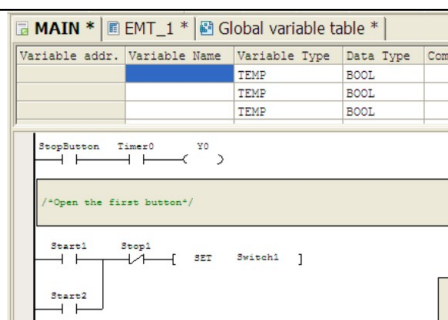
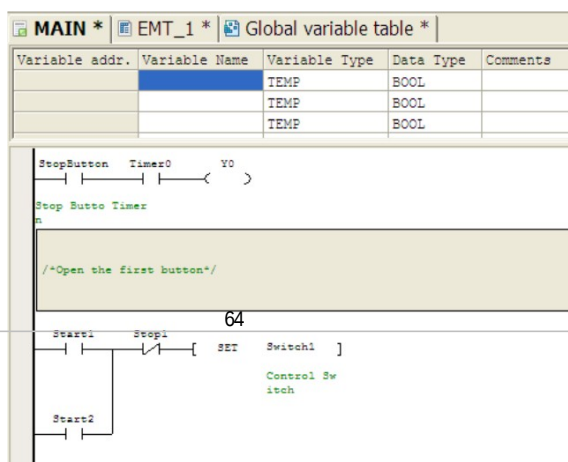


Рисунок 4-9 Стан символічної адресації вибрано

- Коментар до елемента

Ви можете керувати відображенням коментарів до елементів у LAD, вибравши меню **Коментар елемента**, як показано на Рисунку 4-10.



4.4 Підпрограма

4.4.1 Концепція

Будучи необов'язковою частиною користувацької програми, підпрограма (SBR) є незалежною програмною організаційною одиницею (POU), яку може викликати головна програма або інші SBRs.

Ви можете використовувати SBRs у вашій користувацькій програмі для:

1. Зменште розмір користувацької програми. Ви можете записати повторювану частину програми як SBR та викликати її, коли це необхідно.
2. Уточніть структуру програми, зокрема структуру головної програми.
3. Зробіть програму користувача більш портативною.

4.4.2 Примітка щодо використання SBRs

Під час написання або виклику SBR, Вам необхідно звернути увагу на таке:

1. PLCs підтримує максимум 6 рівнів вкладеності SBR викликів.

Нижче наведено вдалий приклад застосування 6-рівневого SBR вкладення виклику:

MAIN→SBR1→SBR2→SBR3→SBR4→SBR5→SBR6

(де "→" означає виклик відповідного SBRs з CALL інструкцією)

2. PLCs не підтримують рекурсивний виклик та циклічний виклик SBRs.

Наступні два екземпляри застосунку демонструють два недопустимі SBR виклики.

-
- MAIN→SBR0→SBR0 (рекурсивний виклик, неприпустимо)
 - MAIN→SBR0→SBR1→SBR0 (циклічний виклик, неприпустимо)

3. А максимум 64 SBRs можна визначити у користувацькій програмі.
4. А У таблиці змінних SBR можна визначити максимум 16 типів 16-бітних та 16-слівних змінних.
5. При виклику SBR, необхідно зауважити, що тип операнда інструкції CALL має збігатися з типом змінної, визначеним у таблиці змінних SBR, і компілятор перевіряє правильність цього збігу.
6. Програмам переривань заборонено викликати SBRs

4.4.3 Визначення змінної SBR таблиця

- SBR таблиця змінних
Таблиця змінних SBR використовується для відображення всіх параметрів інтерфейсу SBR та локальних змінних (разом званих змінними) і визначення їхніх властивостей.

- SBR властивості параметрів
Змінні SBR (включаючи параметри інтерфейсу та локальні змінні) мають такі властивості:

1. Адреса змінної

Залежно від типу даних змінної, програмне забезпечення автоматично призначає фіксовану адресу елемента LM або V для кожної змінної SBR послідовно.

2. Змінна назва

Ви можете надати кожній SBR змінній назву (псевдонім). Ви можете використовувати змінну в програмі, беручи в лапки її назву.

3. Тип змінної

Змінні SBR класифікуються на чотири типи: IN, OUT, IN_OUT та TEMP.

- IN: Змінна типу IN використовується для передачі вхідного значення SBR, коли SBR викликається.
- OUT: Змінна типу OUT використовується для виклику значення, що повертається, для передачі SBR, коли SBR повертається.
- Змінна типу IN_OUT використовується для передачі вхідного значення SBR під час виклику SBR, або для виклику значення, що повертається, для передачі SBR, коли SBR завершує роботу.
- Типові змінні TEMP використовуються лише як локальні змінні, що є дійсними лише в межах області видимості SBR.

4. Змінний тип даних

Властивості типів даних змінних визначають розрядність даних та діапазон значень змінних. Типи даних змінних наведено у наступній таблиці.

Таблиця 4-1 Типи змінних даних

Тип даних	Опис	Зайнятий LM/V елемент адреса
BOOL	Біт тип	Один LM елемент адреса
INT	Знакове ціле число	Один V елемент адреса
DINT	Знакове ціле число подвійної точності тип	Дві послідовні V адреси елементів
REAL	Тип з рухомою комою	Дві послідовні V адреси елементів

4.4.4 SBR параметр передачі

Якщо локальні вхідні або вихідні змінні визначені в SBR, коли SBR викликається в головній програмі, вам потрібно ввести відповідні значення змінних, глобальні або тимчасові елементи змінних у параметри інтерфейсу SBR. Вам слід зауважити, що локальна змінна повинна мати той самий тип даних, що й параметр інтерфейсу.

4.4.5 SBR екземпляр застосунку

Ось приклад застосування того, як написати та викликати SBR.

- Вступ до функцій екземпляра застосунку
Виклик SBR_1 у головній програмі для виконання операції додавання (3+2) двох цілих констант та присвоєння результату операції 5 до D0.

- Процедури експлуатації екземпляра застосунку

Крок 1: Створення SBR у проєкті та надання йому назви SBR_1.

Крок 2: Запис SBR_1.

1. Створення інтерфейсу операнда виклику SBR через таблицю змінних SBR_1.

1) Визначення змінної 1: Надання їй назви IN1 (тип змінної: IN). Вона використовується як дані типу INT і послідовно призначається елементній адресі V V0.

2) Визначення змінної 2: Надання їй назви IN2 (тип змінної: IN). Вона використовується як дані типу INT і послідовно призначається адресі елемента V V1.

3) Визначення змінної 3: Надання їй назви OUT1 (тип змінної: OUT). Вона використовується як дані типу INT і послідовно призначається елементній адресі V V2.

2. Запис SBR_1 як:

LD SM0

ADD #IN1 #IN2 #OUT1

Процес запису SBR_1 показано на наступному рисунку.

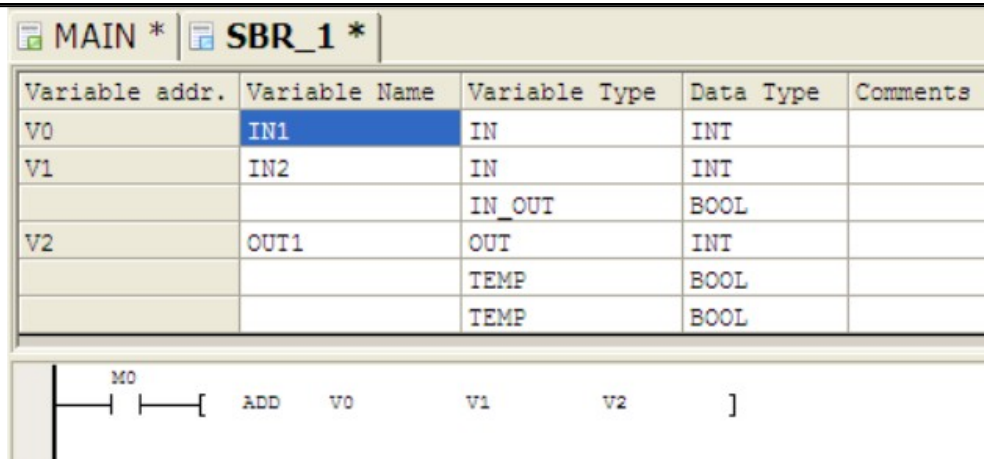


Рисунок 4-11 Процес запису SBR_1

Крок 3: Написання основної програми та виклик SBR

Використання інструкції CALL у головній програмі для виклику

SBR_1. Відповідна головна програма виглядає так:

LD M0

CALL SBR_1 3 2 D0

Ви можете використовувати параметри для передачі таблиці відповідних зв'язків та заповнення параметрів, що передаються або повертаються під час виклику SBR.

- Параметр IN1 вводиться для передачі цілої константи 3
- Параметр IN2 вводиться для передачі цілої константи 2
- Значення, що повертається OUT1, зберігається в D0

Вищевказана програма показана на наступному рисунку.

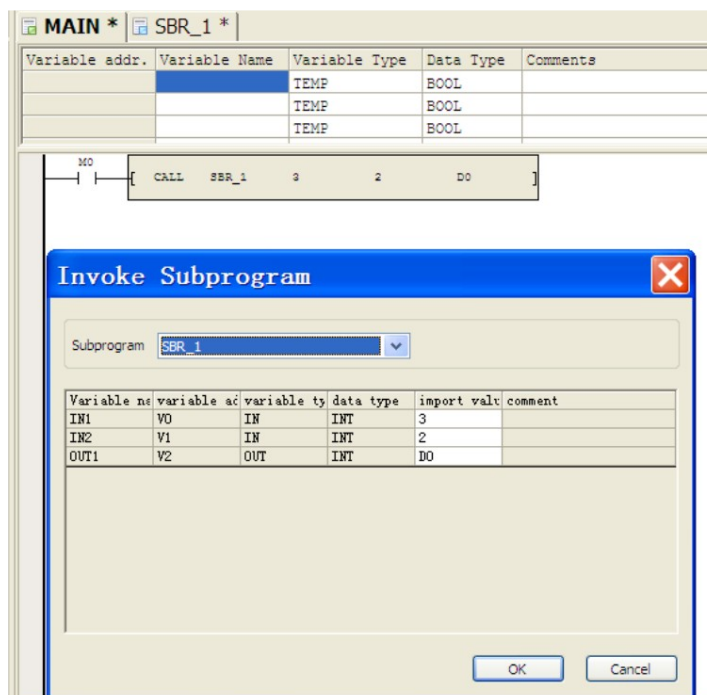


Рисунок 4-12 Виклик SBR

Крок 4: Компіляція, завантаження та запуск користувацької програми для перевірки логіки підпрограми.

- Результат виконання застосунку екземпляра

Коли M0 дорівнює ON, викликається SBR_1. Значення 2 та 3 передаються в операнди IN1 та IN2 для виконання операції додавання. Результат 5 потім повертається в основну програму, тобто D0 дорівнює 5.

4.5 Загальна інформація інструкцій

4.5.1 операнди

Операнди інструкцій можна класифікувати на наступні два типи:

- Операнди джерела: Або S (або S1, S2, S3 ... коли їх декілька в одній інструкції). Інструкція зчитує значення з операндів джерела для обчислення.
- Операнди призначення: або D (або D1, D2, D3 ... коли їх декілька в одній інструкції). Інструкція керує операндами призначення або виводить на них значення.

Операндами можуть бути бітові елементи, словні елементи, елементи подвійного слова або константи. Детальну інформацію щодо опису інструкцій див. у розділі 5 та розділі 6.

4.5.2 Прапорець біта

Операції інструкцій можуть впливати на наступні три типи прапорців бітів.

- Прапорець нуля SM80
Встановлення прапорця нуля, коли результат операції інструкції дорівнює нулю.
- Прапорець переносу SM81
Встановлення прапорця перенесення, коли результат операції інструкції є перенесенням.
- Прапорець позики SM82
Встановлення прапорця позики, коли результат операції інструкції є позикою.

4.5.3 Обмеження щодо використання інструкцій

Існують певні обмеження щодо застосування деяких інструкцій, деякі з яких наведено нижче. Для отримання детальної інформації зверніться до докладних інструкцій відповідних команд.

- Ексклюзивні апаратні ресурси
Деякі інструкції потребують апаратних ресурсів. Коли певна інструкція використовує конкретне обладнання, доступ до нього забороняється для інших інструкцій, оскільки зайняття ресурсу є ексклюзивним.

Використання інструкцій високошвидкісного підрахунку та інструкції SPD для прикладу застосування. Будь-яка з цих інструкцій може займати певні вхідні точки серед X0–X7. Обмежені ресурси унеможливають виконання цих інструкцій одночасно.
- Ексклюзивний час
Виконання певних інструкцій може зайняти певний час. Тому під час використання цих інструкцій необхідно переконаватися, що інструкції мають достатньо часу для виконання функції. Лише одна може бути виконана в певний момент часу, коли система працює.

Використання інструкції ХМТ для екземпляра застосунку. Через своєчасність зв'язку інструкція ХМТ надсилається на вільний порт, і лише одна може виконуватися одночасно. Аналогічно, вільний порт може виконувати лише одну інструкцію RCV за раз. Щоразу, коли виконується інструкція Modbus, канал зв'язку стає недоступним для інших інструкцій на певний час. Те саме стосується інших інструкцій, таких як інструкція високошвидкісного виходу, інструкції позиціонування.
- Межі застосування інструкцій
Деякі інструкції не можуть бути використані у певних ситуаціях через обмежену сферу їхнього застосування.

Наприклад, пару інструкцій MC/MCR не можна використовувати у стані кроку, запрограмованому за допомогою SFC.

Розділ 5 Основні інструкції

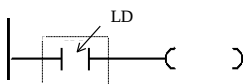
У цьому розділі детально описано базові інструкції мікро-PLCs серії VC, включаючи формат інструкції, операнд, прапорець впливу, функцію, приклад застосування та часову діаграму.

Розділ 5 Основні інструкції.....	57
5.1 Інструкції контактної логіки.....	59
5.1.1 LD: NO контакт інструкція.....	59
5.1.2 LDI: NC контакт інструкція.....	59
5.1.3 AND: NO контакт та інструкція.....	59
5.1.4 ANI: NC контакт та інструкція.....	60
5.1.5 OR: NO контакт або інструкція.....	60
5.1.6 ORI: NO контакт або інструкція.....	61
5.1.7 OUT: Вихід котушки інструкція.....	61
5.1.8 ANB: Блок потоку енергії та інструкція.....	61
5.1.9 ORB: Блок потоку енергії або інструкція.....	62
5.1.10 MPS: Інструкція з виштовхування потоку енергії на виході інструкція.....	62
5.1.11 MRD: Інструкція для зчитування значення верхівки стека потоку вихідної енергії.....	63
5.1.12 MPP: Інструкція виштовхування стека потоку енергії виходу.....	63
5.1.13 EU: Інструкція виявлення переднього фронту сигналу.....	64
5.1.14 ED: Інструкція виявлення спадного фронту.....	64
5.1.15 LDP: Передній фронт контакту.....	65
5.1.16 LDF: Спадаючий фронт контакту.....	65
5.1.17 ANDP: Передній фронт контакту.....	66
5.1.18 ANDF: Задній фронт контакту.....	66
5.1.19 ORP: Передній фронт контакту.....	67
5.1.20 ORF: Задній фронт контакту.....	67
5.1.21 PLP: Вихід за переднім фронтом інструкція.....	68
5.1.22 PLF: Вихід за спадним фронтом інструкція.....	68
5.1.23 INV: Інструкція заперечення потоку енергії.....	69
5.1.24 SET: Інструкція щодо комплексу котушок.....	69
5.1.25 RST: інструкція скидання котушки.....	69
5.1.26 NOP: Відсутня операція інструкція.....	70
5.2 Основні інструкції з керування.....	70
5.2.1 MC: Основна інструкція з керування.....	70
5.2.2 MCR: Інструкція скидання головного керування інструкція.....	70
5.3 SFC інструкції.....	71
5.3.1 STL: SFC стан інструкція із завантаження.....	71
5.3.2 SET Sxx: SFC перехід стану інструкція.....	72
5.3.3 OUT Sxx: SFC стан інструкція переходу.....	72
5.3.4 RST Sxx: SFC скидання стану інструкція.....	72
5.3.5 RET: SFC кінець сегмента програми інструкція.....	73
5.4 Інструкції таймера.....	73
5.4.1 TON: Час затримки увімкнення інструкція.....	73
5.4.2 TONR: Інструкція затримки увімкнення типу пам'яті інструкція.....	73
5.4.3 TOF: Інструкція щодо затримки вимкнення.....	74
5.4.4 TMON: Неперезапускальна моностабільна часова інструкція.....	75
5.5 Інструкції лічильника.....	76

5.5.1 STU: 16-бітний інкрементний лічильник інструкцій.....	76
5.5.2 CTR: 16-бітна інструкція циклічного підрахунку.....	76
5.5.3 DCNT: 32-розрядна інструкція інкременту та декременту лічильника.....	77

5.1 Інструкції контактної логіки

5.1.1 LD: NO контакт інструкція

LAD:		Екземпляр застосування	Застосовна модель	LD	VC1S VC1 VC2 VC3 VC5	
			Зумовлений прапор біт	OUT Y0		
IL: LD (S)			Довжина кроку	1		
Операнд	Тип	Застосовний програмний елемент				Індексація
Д						я

Operand description

S: Джерельний операнд

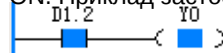
Опис функції

Підключення до лівої шини для підключення (стан: ON) або відключення (стан: OFF) потоку енергії.



Коли M0 дорівнює ON, вихід Y0 дорівнює

ON. Приклад застосування

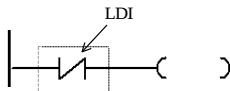


LD D1.2

OUT Y0

Коли другий біт D1 дорівнює 1, вихід Y0 має стан ON.

5.1.2 LDI: NC контакт інструкція

LAD:										Застосовна модель		VC1S VC1 VC2 VC3 VC5						
										Прапор впливу біт								
IL: LDI (S)										Довжина кроку		1						
Операнд		Тип		Застосовний програмний елемент										Індексація				
S		BOOL		X	Y	M	S	LM	SM		Dx.y		C	T				

Операнд опис

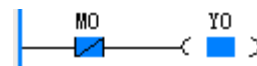
S: Опис функції

вихідного

операнда

Підключення до лівої шини для підключення (стан: ON) або відключення (стан: OFF) потоку енергії.

Екземпляр застосування

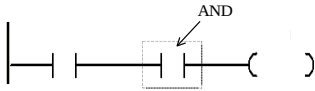


LDI M0

OUT Y0

Коли M0 дорівнює OFF, вихід Y0 має стан ON.

5.1.3 AND: NO контакт та інструкція

LAD: 		Застосовна модель		VC1S VC1 VC2 VC3 VC5									
		Прапорець впливу біт											
IL: AND (S)		Довжина кроку		1									
Операнд	Тип	Застосовний програмний елемент											Індексація
S	BOOL	X	Y	M	S	LM	SM		Dx.y		C	T	

Операнд опис

S: Функція

операнда джерела

опис

Виконання операції "AND" над станом ON/OFF визначеного контакту (**S**) та потоком енергії, і присвоєння отриманого результату до поточний енергетичний потік.

Екземпляр застосунку

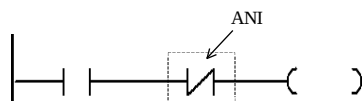


```
LD M0
AND M1
OUT Y0
```

Коли M0 дорівнює ON, а M1 дорівнює ON, вихід Y0 становить ON.

5.1.4 ANI: NC контакт та інструкція

LAD:



Застосовна модель

VC1S VC1 VC2 VC3 VC5

Прапор впливу біт

IL: ANI (S)

Довжина кроку

1

Операнд	Тип	Застосовний програмний елемент										Індексація	
D													
S	BOOL	X	Y	M	S	LM	SM		Dx.y		C	T	

Операнд опис

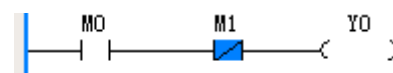
S: Функція

операнда джерела

опис

Виконання операції "NOT" над станом ON/OFF вказаного контакту (**S**), а потім виконання операції "AND" над отриманим результатом та поточним значенням потоку енергії, і присвоєння отриманого результату поточний енергії потік

Екземпляр застосунку

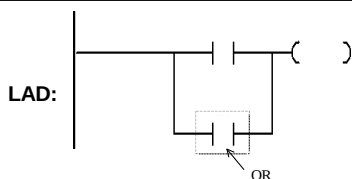


```
LD M0
ANI M1
OUT Y0
```

Коли M0 дорівнює ON, а M1 дорівнює OFF, вихід Y0 становить ON.

5.1.5 OR: NO контакт або інструкція

LAD:



Застосовна модель

VC1S VC1 VC2 VC3 VC5

Прапор впливу біт

IL: OR (S)

Довжина кроку

1

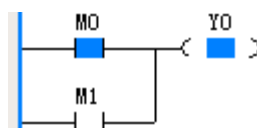
Операнд	Тип	Застосовний програмний елемент										Індексація	
D													
S	BOOL	X	Y	M	S	LM	SM		Dx.y		C	T	

Опис операнда **S**:

Опис функції

операнда джерела

Екземпляр застосунку

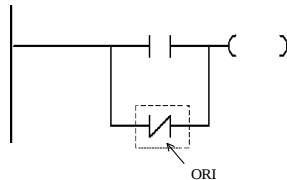


```
LD M0
OR M1
OUT Y0
```

Виконання операції "OR" над станом ON/OFF визначеного контакту (S) та поточним потоком енергії, і призначення отриманого результат поточного енергетичного потоку.

Коли M0 або M1 дорівнює ON, вихід Y0 становить ON.

5.1.6 ORI: NO контакт або інструкція

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапор впливу біт							
IL: ORI (S)										Довжина кроку		1					
Операнд		Тип		Застосовний програмний елемент										Індексація			
S		BOOL		X	Y	M	S	LM	SM		Dx.y		C	T			

Операнд опис

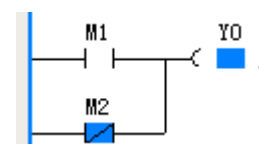
S: Функція

операнда джерела

опис

Виконання операції "NOT" над станом ON/OFF визначеного контакту (S), а потім виконання операції "OR" над отриманим результатом та поточним значенням потоку енергії, і присвоєння отриманого результату поточній енергії потік

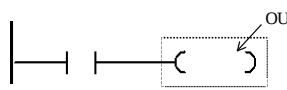
Екземпляр застосунку



LD M1
ORI M2
OUT Y0

Коли M1 є ON або M2 є OFF, вихід Y0 є ON.

5.1.7 OUT: Вихід котушки інструкція

LAD:										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт							
IL: OUT (S)										Довжина кроку		1					
Операнд	Тип	Застосовний програмний елемент										Індексація					
S	BOOL	X	Y	M	S	LM	SM		Dx.y		C	T					

Операнд опис

S: Опис функції

вихідного

операнда

Присвоєння значення поточного енергопоту до призначена котушка (D).

Екземпляр застосунку

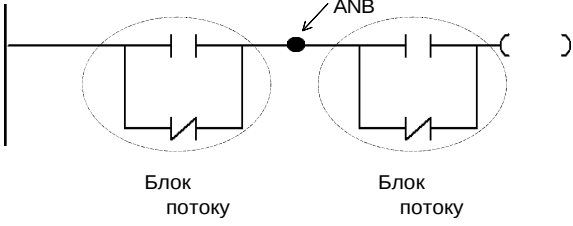


LD M1
OUT Y0

Коли M1 дорівнює ON, вихід Y0 становить ON.

5.1.8 ANB: Блок потоку енергії та інструкція

LAD:	Застосовна	VC1S VC1 VC2 VC3 VC5

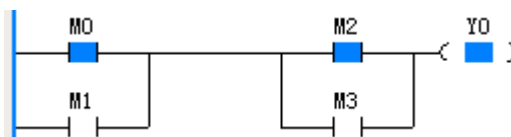
	модель	
	Прапор впливу біт	
IL: ANB	Довжина кроку	1

Опис операнда

Функція опис

Виконання "AND" операції над цим значення потоку енергії двох блоків потоку енергії, а потім присвоєння отриманого результату поточному потоку енергії.

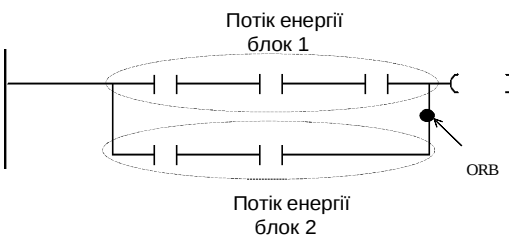
Екземпляр застосунку



LD M0
OR M1
LD M2
OR M3
ANB
OUT Y0

Коли або M0, або M1 увімкнено, і або M2, або M3 є ON, Y0 ON.

5.1.9 ORB: Блок потоку енергії або інструкція

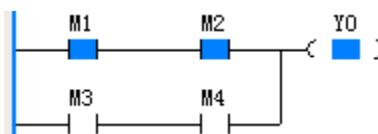
LAD:	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Прапор впливу біт	
IL: ORB	Довжина кроку	1

Опис операнда

Функція опис

Виконання операції "OR" над значення потоку енергії двох блоків потоку енергії, а потім присвоєння отриманого результату поточному потоку енергії.

Екземпляр застосунку

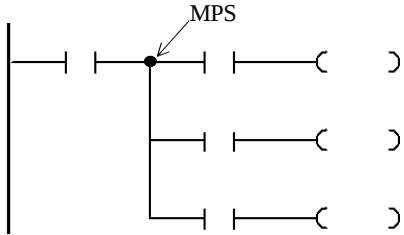


LD M1
AND M2
LD M3
AND M4
ORB
OUT Y0

Коли обидва M1 та M2 є ON, або обидва M3 та M4 є ON, Y0 ON.

5.1.10 MPS: Інструкція з виштовхування потоку енергії на виході інструкція

LAD:	Застосовна модель	VC1S VC1 VC2 VC3 VC5

	Прапорець впливу біт	
IL: MPS	Довжина кроку	1

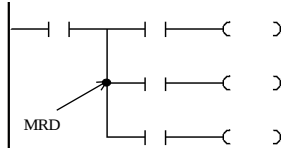
Опис функції

Збереження та запис поточних значень енергії для розрахунку потоку енергії наступного виходу філії

Примітка:

Заборонено використовувати інструкції MPS (без інструкцій MPP між ними) більше 8 разів поспіль в одній мережі релейної діаграми, інакше стек виходу потоку енергії може переповнитися.

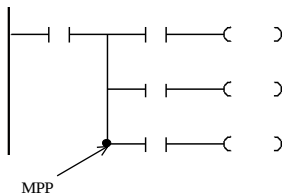
5.1.11 MRD: Інструкція для зчитування значення верхівки стека потоку вихідної енергії

LAD: 	Застосовна модель Прапорець впливу біт	VC1S VC1 VC2 VC3 VC5
IL: MRD	Довжина кроку	1

Опис функції

Призначення верхнього значення стека виходу потоку енергії поточному потоку енергії.

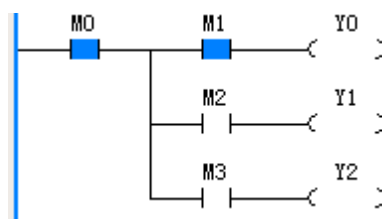
5.1.12 MPP: Вихідний стек потоку енергії, виштовхування інструкції

LAD: 	Застосовна модель Прапорець впливу біт	VC1S VC1 VC2 VC3 VC5
IL: MPP	Довжина кроку	1

Опис функції

Виконання операції Pop над тим вихідний стек потоку енергії, та призначення the вилучене значення до поточного енергетичного потоку.

Екземпляр застосунку



LDM0

MPS

AND M1

OUT Y0

MRD

AND M2

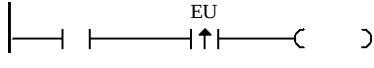
OUT Y1

MPP

AND M3

OUT Y2

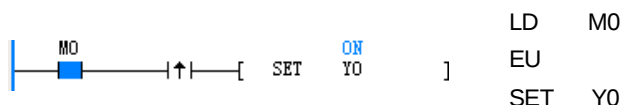
5.1.13 EU: Інструкція виявлення переднього фронту сигналу

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Зумовлений прапор біт	
IL: EU	Довжина кроку	2

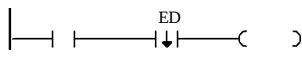
Опис функції

Порівняння стану потоку вхідної енергії у поточному циклі сканування з попереднім. Якщо потік енергії зростає (OFF→ON), вихід у поточному циклі сканування є дійсним.

Екземпляр застосунку



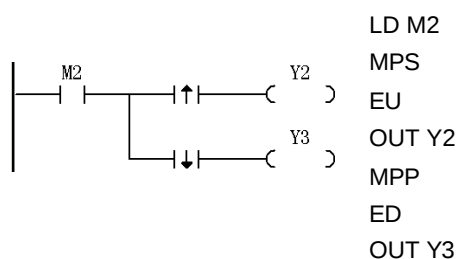
5.1.14 ED: Інструкція виявлення спадного фронту сигналу

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Зумовлений прапор біт	
IL: ED	Довжина кроку	2

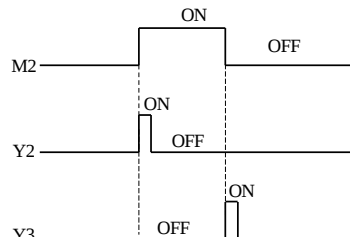
Опис функції

Порівняння стану потоку вхідної енергії у поточному циклі сканування зі станом у попередньому циклі. Якщо потік енергії падає (OFF→ON), вихід у поточному циклі сканування є дійсним.

Екземпляр застосунку



Графік послідовності застосунку екземпляра



Примітка

1. Протягом двох послідовних циклів сканування стани контакту M2 дорівнюють OFF та ON відповідно, а інструкція EU виявляє зміну переднього фронту, завдяки чому Y2 видає стан ON тривалістю в один цикл сканування.

2. Протягом двох послідовних циклів сканування стани контакту M2 дорівнюють ON та OFF відповідно, а інструкція ED виявляє спадний фронт, завдяки чому Y3 видає стан ON з тривалістю в один цикл

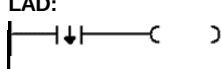
У LAD інструкції контакту за переднім фронтом та контакту за заднім фронтом повинні використовуватися послідовно, а не паралельно з іншими контактними елементами.

У LAD інструкції контакту за переднім фронтом та контакту за заднім фронтом не можуть бути підключені безпосередньо до лівої шини потоку енергії.

На наступному рисунку показано приклад неправильного використання інструкцій EU та ED у LAD.

chastotnik.ua

5.1.16 LDF: Спадаючий фронт контакту

LAD: 										Застосовно модель		VC1S VC1 VC2 VC3 VC5					
										Прапор впливу біт							
IL: LD (S)										Довжина кроку		1					
Операнд	Тип	Застосовний програмний елемент														Індексація	
S	BOOL	X	Y	M	S	LM	SM					C	T				

Операнд опис

S: Опис функції

вихідного

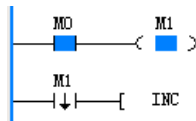
операнда

Команда LDF використовується для зчитування спадного фронту сигналу контакту

Виявляється спадний фронт відповідного сигналу, після чого контакт стає активним,

Під час наступного сканування контакт стає недійсним

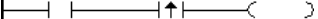
Екземпляр застосунку



```
LD    M0
OUT   Y0
LDF   M1
INC    D0
```

Коли M0 дорівнює ON, вихід M1 дорівнює ON. У цей момент, якщо M1 змінюється з ON на OFF і залишається дійсним протягом 1 циклу сканування, D0 виконує інкремент на 1. У наступному циклі сканування M1 стане недійсним, а D0 залишиться рівним 1.

5.1.17 ANDP: Передній фронт контакту

LAD: 										Застосовно модель		VC1S VC1 VC2 VC3 VC5					
										Прапор впливу біт							
IL: LD (S)										Довжина кроку		1					
Операнд	Тип	Застосовний програмний елемент													Індексація		
S	BOOL	X	Y	M	S	LM	SM					C	T				

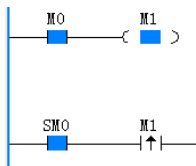
Опис операнда **S:**

Опис функції

операнда джерела

ANDP інструкція призначена для участі в AND операції переднього фронту контакту.

Екземпляр застосунку



```
LD    M0
OUT   Y0
LD    SM0
ANDP  M1
INC    D0
```

Коли M0 дорівнює ON, вихід M1 дорівнює ON. У цей момент, якщо M1 змінюється з OFF на ON і залишається дійсним протягом 1 циклу сканування, D0 виконує інкремент на 1. У наступному циклі сканування M1 стане недійсним, а D0 залишиться 1.

5.1.18 ANDF: Задній фронт контакту

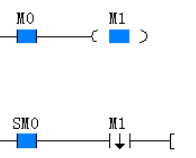
LAD: </									

Опис операнда **S:**

Опис функції

операнда джерела

Екземпляр застосунку



```
LD    M0
OUT   Y0
LD    SM0
```

ANDF

M1 INC

D0

Інструкція ANDF призначена для участі в операції AND за спадним фронтом контакту.

Коли M0 дорівнює ON, вихід M1 дорівнює ON. У цей час, якщо M1 змінюється з ON на OFF і залишається дійсним протягом 1 циклу сканування, D0 виконує інкремент на 1. У наступному циклі сканування M1 буде недійсний і D0 залишається 1.

5.1.19 ORP: Передній фронт контакту

LAD:										Застосовно модель		VC1S VC1 VC2 VC3 VC5					
										Зумовлений прапор біт							
IL: LD (S)										Довжина кроку		1					
Операнд	Тип	Застосовний програмний елемент												Індексація			
S	BOOL	X	Y	M	S	LM	SM					C	T				

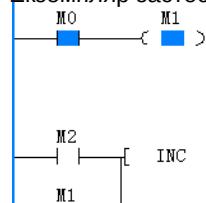
Опис операнда S:

Опис функції

операнда джерела

Інструкція ORP полягає в участі в операції OR переднього фронту контакту.

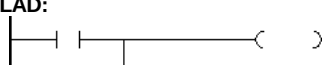
Екземпляр застосунку



```
LD M0
OUT Y0
LD M2
ORP M1
INC D0
```

Коли M0 дорівнює ON, вихід M1 дорівнює ON. У цей момент, якщо M1 змінюється з OFF на ON і залишається дійсним протягом 1 циклу сканування, D0 виконує інкремент на 1. У наступному циклі сканування M1 стане недійсним, а D0 залишиться 1.

5.1.20 ORF: Задній фронт контакту

LAD: 										Застосовно модель		VC1S VC1 VC2 VC3 VC5					
										Прапор впливу біт							
IL: LD (S)										Довжина кроку		1					
Операнд	Тип	Застосовний програмний елемент														Індексація	
S	BOOL	X	Y	M	S	LM	SM					C	T				

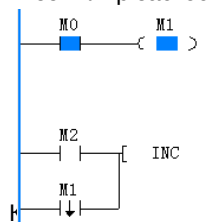
Опис операнда S:

Опис функції

операнда джерела

Інструкція ORF полягає в участі в операції OR за спадним фронтом контакту.

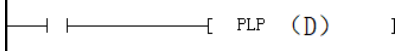
Екземпляр застосунку



```
LD M0
OUT Y0
LD M2
ORF M1
INC D0
```

Коли M0 дорівнює ON, вихід M1 дорівнює ON. У цей момент, якщо M1 змінюється з ON на OFF і залишається дійсним протягом 1 циклу сканування, D0 виконує інкремент на 1. У наступному циклі сканування M1 стане недійсним, а D0 залишиться рівним 1.

5.1.21 PLP: Вихід за переднім фронтом інструкція

LAD: 									Застосовно модель				VC1S VC1 VC2 VC3 VC5					
									Зумовлений прапор біт									
IL: LD (S)									Довжина кроку				1					
Операнд	Тип	Застосовний програмний елемент														Індексація		
S	BOOL	X	Y	M	S	LM	SM					C	T					

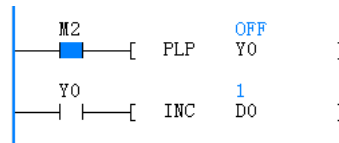
Опис операнда **S**:

Опис функції

операнда джерела

PLP інструкція полягає у зчитуванні переднього фронту вихідного сигналу. Якщо під час цього сканування виявлено передній фронт вихідного сигналу, вихідний контакт буде ON. У наступному циклі сканування контакт виходу стане недійсним.

Екземпляр застосунку



LD M2

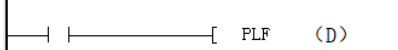
PLP Y0

LD Y0

INC D0

Y0 дорівнює ON і залишається дійсним протягом 1 циклу сканування, D0 виконує інкремент на 1. У наступному циклі сканування M1 стане недійсним, а D0 залишиться 1.

5.1.22 PLF: Вихід за спадним фронтом інструкція

LAD: 										Застосовно модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт							
IL: LD (S)										Довжина кроку		1					
Операнд	Тип	Застосовний програмний елемент												Індексація			
S	BOOL	X	Y	M	S	LM	SM					C	T				

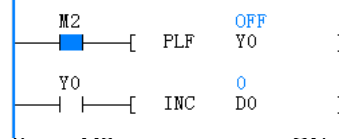
Опис операнда **S**:

Опис функції

операнда джерела

Інструкція PLP полягає у використанні спадного фронту вихідного сигналу. Якщо під час цього сканування виявлено спадний фронт вихідного сигналу, вихідний контакт буде ON. У наступному циклі сканування контакт виходу стане недійсним.

Екземпляр застосунку



LD M2

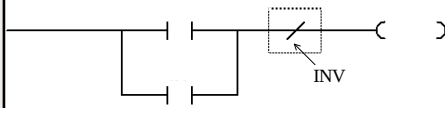
PLF Y0

LD Y0

INC D0

Коли M2 змінюється з ON на OFF, вихід Y0 стає ON і залишається дійсним протягом 1 циклу сканування, D0 виконує інкремент на 1. У наступному циклі сканування M1 стане недійсним, а D0 залишиться 1.

5.1.23 INV: Інструкція заперечення потоку енергії

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Прапорець впливу біт	
IL: INV	Довжина кроку	1

Опис функції

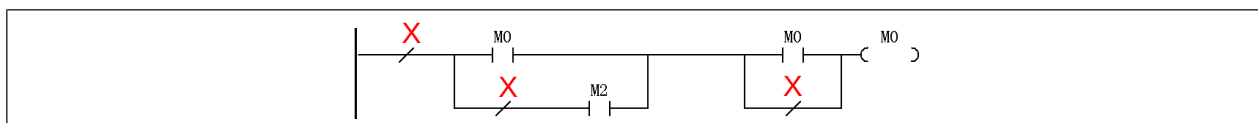
Виконання операції "INV" над поточним значенням енергетичного потоку з подальшим присвоєнням отриманого результату поточному енергетичному потоку.

Примітка

У LAD інструкцію INV необхідно використовувати послідовно, а не паралельно з іншими контактними елементами. INV не можна використовувати як першу інструкцію у паралельних вхідних гілках.

In LAD, the INV instruction cannot be directly connected to the left energy flow bus.

The following figure shows an example of incorrect use of the INV instruction in LAD.



5.1.24 SET: Інструкція щодо комплекту котушок

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Зумовлений прапор біт							
IL: SET (S)										Довжина кроку		1					
Операнд	Тип	Застосовний програмний елемент													Індексація		
S	BOOL		Y	M	S	LM	SM		Dx.y		C	T					

Опис операнда **S**:

Опис функції

операнда джерела

Встановлення бітового елемента, призначеного **D** коли потік енергії дійсний.

Екземпляр застосунку



5.1.25 RST: інструкція скидання котушки

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5				
										Зумовлений прапор біт						
IL: RST (S)										Довжина кроку		1				
Операнд	Тип	Застосовний програмний елемент													Індексація	
D S	BOOL		Y	M	S	LM	SM		Dx.y		C	T				

Операнд опис

S: Опис функції

вихідного

операнда

Скидання вказаного бітового елемента (**D**) до нуля, коли потік енергії є дійсним.

Екземпляр застосунку



LD M0

RST M1

Примітка

Якщо D є елементом C, відповідне значення лічильника видаляється. Якщо D є елементом T, відповідне значення часу видаляється.

5.1.26 NOP: Відсутня операція інструкція

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Зумовлений прапор біт	
IL: NOP	Довжина кроку	1

Опис функції

Виконання операції

відсутнє.

Примітка

У LAD ця інструкція не може бути безпосередньо підключена до лівої шини потоку енергії.

5.2 Основні інструкції з керування

5.2.1 MC: Основна інструкція з керування

LAD: 		Застосовна модель	VC1S VC1 VC2 VC3 VC5											
		Зумовлений прапор біт												
IL: MC (S)		Довжина кроку	3											
Операнд	Тип	Застосовний програмний елемент												Індексація
S	INT	Складається немає												

Операнд опис

S: Джерельний операнд

5.2.2 MCR: Інструкція скидання головного керування інструкція

LAD: 		Застосовна модель	VC1S VC1 VC2 VC3 VC5											
		Зумовлений прапор біт												
IL: MCR (S)		Довжина кроку	1											
Операнд	Тип	Застосовний програмний елемент												Індексація
S	INT	Складається немає												

Опис операнда **S**:

Опис функції

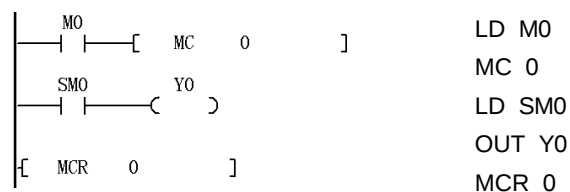
операнда джерела

1. Узгодження інструкцій MC та MCR для формування структури MC-MCR. Інструкція MC представляє початок структури MC-MCR, а її операнд S є міткою структури MC-MCR. Значення операнда S є константою в діапазоні від 0 до 7. MCR представляє кінець структури MC-MCR.

2. Виконання інструкції посередині структури MC-MCR, коли потік енергії перед інструкцією MC є дійсним,

3. Пропуск інструкції всередині структури MC-MCR, коли потік енергії перед інструкцією MC є недійсним, а також виконання її та видалення операндів призначення, що відповідають OUT, TON, TOF, PWM, HCNT, PLSY, PLSR, DHSCS, SPD, DHSCI, DHSCR, DHSZ, DHST,

DHSP та BOUT у структурі після того, як програма безпосередньо переходить до структури



Коли M0 = ON, інструкції в структурі MC 0–MCR 0 виконуються, і Y0 = ON. Коли M0 = OFF, інструкції в структурі MC 0–MCR 0 не виконуються, а бітовий елемент Y0, визначений операндом призначення інструкції OUT в тому структурі видалено, Y0 = OFF.

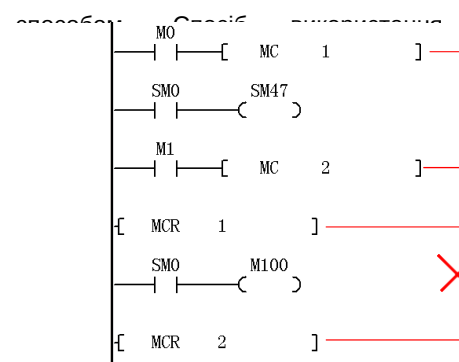
Примітка

1. У LAD, інструкція MCR має бути безпосередньо підключена до лівої шини потоку енергії.

2. У LAD, інструкція MCR не може з'єднуватися паралельно або послідовно з іншими інструкціями.

3. У вкладеному режимі можна використовувати кілька різних пронумерованих структур MC-MCR, але кількість рівнів вкладеності не може перевищувати 8. Структури MC-MCR з однаковим номером не можна використовувати у вкладеному режимі.

4. Дві структури MC-MCR не можуть використовуватися перехресним



Примітка: Його не можна використовувати в SFC програмуванні.

5.3 SFC інструкції

5.3.1 STL: SFC стан інструкція із завантаження

LAD: 										Застосовна модель			VC1S VC1 VC2 VC3 VC5				
										Зумовлений прапор біт							
IL: STL (S)										Довжина кроку			3				
Операнд	Тип	Застосовний програмний елемент													Індексація		
S	BOOL							S									

Опис операнда S:

Опис функції

операнда джерела

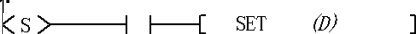
1. Вказує на початок процесу стану кроку (**S**).
2. Виконання вбудованих інструкцій кроку, коли стан кроку є дійсним (ON).
3. Якщо стан кроку змінюється з ON на OFF (зміна по спадному фронту), вбудовані інструкції кроку не виконуються, а операнди призначення видаляються

що відповідають OUT, TON, TOF, PWM, HCNT, PLSY, PLSR, DHSCS, SPD, DHSCI, DHSCR, DHSZ, DHST, DHSP та BOUT.

4. Невиконання вбудованих інструкцій кроку, коли стан кроку є недійсним (OFF).

5. Визначення структури паралельного об'єднання шляхом послідовних інструкцій STL (послідовне з'єднання елементів STL), де інструкції STL можуть бути використані максимум 16 разів поспіль, тобто максимальна кількість гілок у структурі паралельного об'єднання становить 16.

5.3.2 SET Sxx: SFC перехід стану інструкція

LAD:											Застосовна модель		VC1	VC2	VC3	VC5
											Зумовлений прапор					
	SET (D)										Довжина кроку					
	Операнд	Тип	Застосовний програмний елемент										Індексація			
D	BOOL							S								

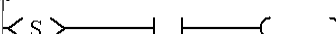
Операнд опис

Опис функції

:Призначення операнд

Встановлення стану вказаного кроку (**D**) як дійсного, коли потік енергії є дійсним, та встановлення поточного дійсного кроку як неможливо завершити перехід стану.

5.3.3 OUT Sxx: SFC стан інструкція переходу

LAD:						Застосовна модель		VC1	VC2	VC3	VC5	
						Зумовлений прапор						
OUT (D)					Довжина кроку							
Операнд	Тип	Застосовний програмний елемент								Індексація		
D	BOOL						S					

Операнд опис

Опис функції

:Призначення операнд

Встановлення стану вказаного кроку (**D**) як дійсного, коли потік енергії є дійсним, та встановлення поточного дійсного кроку як недійсного для завершення переходу стану.

5.3.4 RST Sxx: SFC скидання стану інструкція

LAD:						Застосовна модель		VC1	VC2	VC3	VC5
						Зумовлений прапор					
RST (D)						Довжина кроку					
Операнд	Тип	Застосовний програмний елемент								Індексація	
D	BOOL						S				

Операнд опис

Опис функції

: Цільовий операнд

Встановлення стану вказаного кроку (**D**) як недійсного, коли потік енергії є дійсним.

5.3.5 RET: SFC кінець сегмента програми інструкція

LAD:		Застосовна модель					
		VC1	S	VC1	VC2	VC3	VC5
	RET]	Зумовлений прапор					
RET		Довжина кроку					

Опис функції

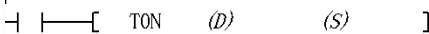
Примітка

Вказування на кінець сегмента SFC програми.

Його можна використовувати лише в основній програмі.

5.4 Інструкції таймера

5.4.1 TON: Часова затримка увімкнення інструкція

LAD:										Застосовна модель VC1S VC1 VC2 VC3 VC5									
										Зумовлений прапор									
TON (D) (S)										Довжина кроку									
Операнд		Тип		Застосовний програмний елемент												Індексація			
D		INT													T				
S		INT		Складається немає KnX		KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	

Опис операнда **D**:

Операнд

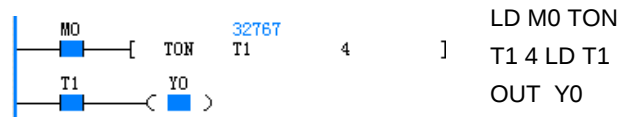
призначення **S**:

Операнд джерела

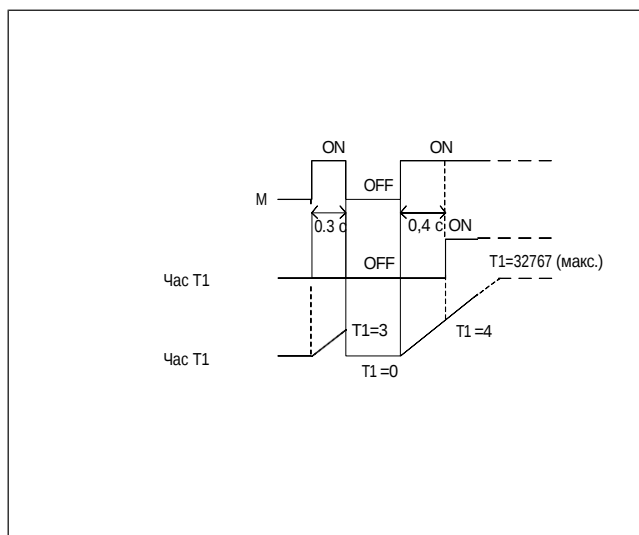
Опис функції

- Відлік часу для вказаного T елемента (D), коли потік енергії є дійсним, а значення часу менше 32 767 (значення часу збільшується з часом), і залишається на рівні 32 767 після досягнення цього значення.
- Встановлення виходу котушки таймера визначеного Telement у ON, коли значення таймера більше або дорівнює попередньо встановленому значенню (S).
- Зупинка відліку часу, скидання значення часу до нуля та встановлення виходу котушки таймера в OFF, коли потік енергії дорівнює OFF.
- Встановлення значення котушки таймера визначеного елемента T на OFF та скидання значення таймера до нуля, коли система виконує інструкцію вперше.

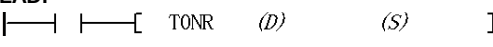
Екземпляр застосунку



Sequence chart of the application instance



5.4.2 TONR: Інструкція затримки увімкнення типу пам'яті інструкція

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт										
IL: TONR (D) (S)									Довжина кроку				5						
Операнд	Тип	Застосовний програмний елемент														Індексація			
D	INT												T						
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√			

Операнд опис

D: Операндпризначення **S:**

Операнд джерела

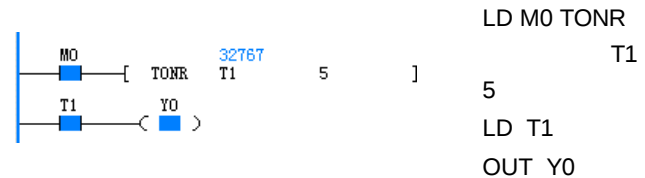
Опис функції

1. Відлік часу для вказаного T елемента (**D**), коли потік енергії є дійсним, а значення часу менше 32 767 (значення часу збільшується з часом), і залишається на рівні 32 767 після досягнення цього значення.

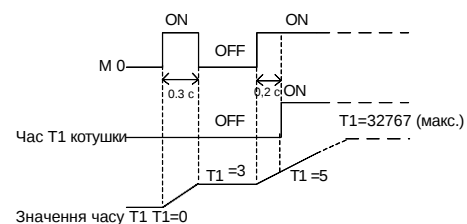
2. Встановлення виходу котушки таймера визначеного T елемента на ON, коли значення таймера більше або дорівнює попередньо встановленому значенню (**S**).

3. Зупинка таймінгу, та збереження поточного таймінгу котушки та значення таймінгу, коли потік енергії дорівнює OFF.

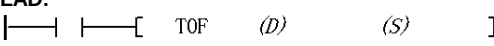
Екземпляр застосунку



Графік послідовності застосунку екземпляра



5.4.3 TOF: Часова затримка вимкнення інструкція

LAD: 								Застосовна модель				VC1S VC1 VC2 VC3 VC5							
								Прапорець впливу біт											
IL: TOF (D) (S)								Довжина кроку				5							
Операнд	Тип	Застосовний програмний елемент														Показчик			
D	INT											T							
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√			

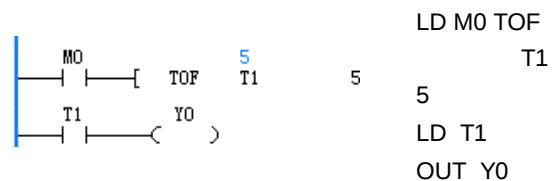
Операнд опис

D: Операндпризначення **S:**

Операнд джерела

Опис функції

Екземпляр застосунку



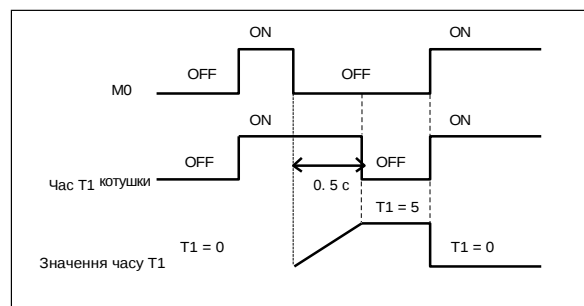
1. Запуск відліку часу вказаного таймера Т (**D**), коли потік енергії змінюється з ON на OFF (зміна за спадним фронтом).

2. Збереження відліку часу, коли потік енергії дорівнює OFF, але вказаний таймер Т було запущено, і встановлення виходу котушки відліку часу елемента Т у стан OFF, коли значення часу досягає попередньо встановленого значення (**S**), після чого залишається на попередньо встановленому значенні.

3. Не відраховується час, коли відлік часу не розпочато, але вхідний потік енергії становить OFF.

4. Зупинка відліку часу, скидання значення часу до нуля та встановлення виходу котушки таймера в ON, коли потік енергії дорівнює ON.

Графік послідовності застосунку екземпляра



5.4.4 TMON: Неперезапускальна моностабільна часова інструкція

LAD:								Застосовна модель				VC1S VC1 VC2 VC3 VC5					
								Прапорець впливу біт									
IL: TMON (D) (S)								Довжина кроку				5					
Операнд	Тип	Застосовний програмний елемент														Показник	
D	INT											T				к по	
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	

Операнд опис

D: Цільовий операнд

S: Джерельний операнд

Опис функції

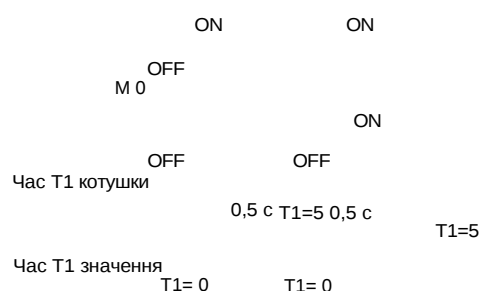
Екземпляр застосунку



Графік послідовності застосунку екземпляра

1. Запуск відліку часу визначеного таймера Т (**D**) (на основі поточного значення), коли потік вхідної енергії змінюється з OFF на ON (зміна за переднім фронтом), і відлік часу ще не розпочато, та утримання виходу котушки таймера ON під час стану відліку часу (тривалість відліку визначається **S**).

2. Збереження синхронізації та підтримка виходу котушки синхронізації ON незалежно від того, як змінюється потік енергії, коли він перебуває у стані синхронізації (тривалість синхронізації визначається **S**).



3. Зупинка відліку часу, скидання значення часу до нуля, і встановлення виходу котушки таймера на OFF, коли значення таймера досягає попередньо встановленої точки.

5.5 Інструкції лічильника

5.5.1 CTU: 16-бітний інкрементний лічильник інструкцій

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: CTU (D) (S)									Довжина кроку			5						
Операнд	Тип	Застосовний програмний елемент														Індексація		
D	INT											C						
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		

Операнд опис

D: Операнд

призначення **S:**

Операнд джерела

Опис функції

- Збільшення значення лічильника вказаного 16-бітного лічильника C (D) на 1, коли вхідний потік енергії змінюється з OFF на ON (зміна за переднім фронтом).
- Залишається у значенні лічильника, коли він досягає 32767.
- Встановлення котушки підрахунку на ON, коли значення лічильника більше або дорівнює заданій точці (**S**).

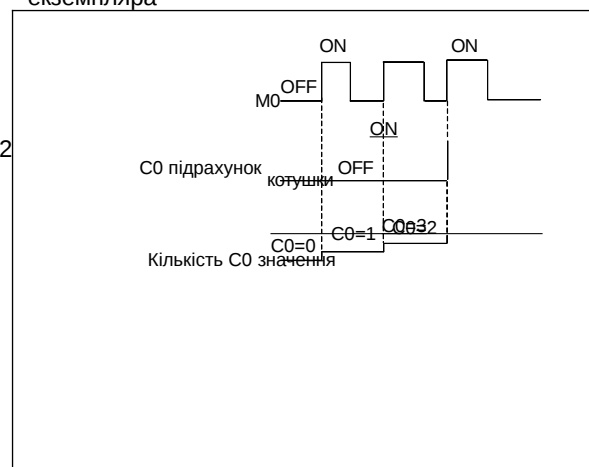
Примітка

Встановлення адреси 16-бітного лічильника C (**D**) у діапазоні значень від C0 до C199.

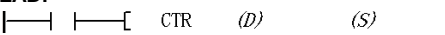
Екземпляр застосунку



Діаграма часової послідовності екземпляра



5.5.2 CTR: 16-бітна інструкція циклічного підрахунку

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5							
									Зумовлений прапор біт											
IL: CTR (D) (S)									Довжина кроку				5							
Операнд	Тип	Застосовний програмний елемент														Індексація				
D	INT												C							
S	INT	Складається	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√				

немає

Операнд опис

D: Операндпризначення **S:**

Операнд джерела

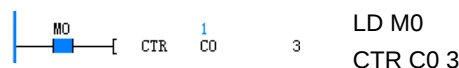
Опис функції

1. Збільшення значення лічильника вказаного 16-бітного лічильника **C (D)** на 1, коли потік вхідної енергії змінюється з OFF на ON (зміна за переднім фронтом).

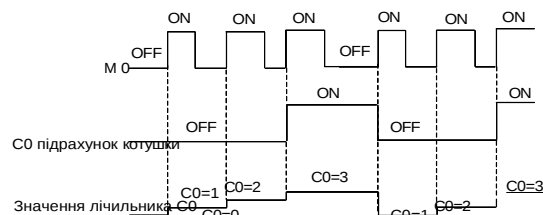
2. Встановлення катушки лічильника на ON, коли значення лічильника дорівнює заданій точці (**S**).

3. Встановлення значення лічильника на 1, а катушки лічильника на OFF, коли значення лічильника досягає попередньо встановленої точки (**S**), і вхідний потік енергії змінюється з OFF на ON

Екземпляр застосунку



Діаграма послідовності екземпляра застосунку

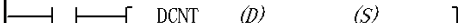


Примітка

1. Виконання жодної дії підрахунку, коли попередньо встановлене значення підрахунку (**S**) менше або дорівнює 0.

2. Встановлення адреси 16-бітного лічильника **C (D)** у діапазоні значень від C0 до C199.

5.5.3 DCNT:32-розрядна інструкція інкременту та декременту лічильника

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: DCNT (D) (S)									Довжина кроку			7						
Операнд	Тип	Застосовний програмний елемент														Індексація		
D	DINT											C						
S	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		

Операнд опис

D: Операндпризначення **S:**

Операнд джерела

Опис функції

1. Збільшення або зменшення значення лічильника вказаного 32-бітного лічильника **C** (**D**) на 1, коли потік вхідної енергії змінюється з OFF на ON (зміна по передньому фронту) (збільшення та зменшення залежать від відповідного прапорця SM біта).

2. Для лічильників інкременту, встановлення котушки підрахунку на ON, коли значення підрахунку більше або дорівнює заданій точці (**S**).

3. Для лічильників на зменшення, встановлення котушки підрахунку на OFF, коли значення лічильника менше або дорівнює заданій точці (**S**).

4. Зміна значення лічильника на – 2147483648, коли значення лічильника становить 2147483647 і збільшується на 1.

5. Зміна значення лічильника на 2147483647, коли значення лічильника дорівнює –2147483648 і зменшується на 1.

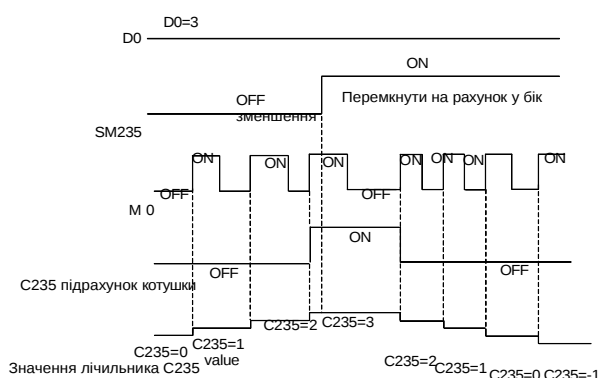
Примітка

Адреса елемента **C** (**D**) повинна бути в діапазоні від C200 до C235.

Екземпляр застосунку



Послідовнісна діаграма екземпляра застосунку



Розділ 6 інструкції із застосування

У цьому розділі детально описано інструкції із застосування мікро-PLCs серії VC, включаючи формат інструкції (форму), операнд, біт прапорця, на який здійснюється вплив, функцію, приклад застосування та діаграму послідовності.

Розділ 6 інструкції із застосування..... 79

6.1 Інструкції керування потоком програми.....	85
6.1.1 FOR: Інструкція циклу.....	85
6.1.2 NEXT: Інструкція повернення циклу інструкція.....	85
6.1.3 LBL: Визначення мітки переходу інструкція.....	86
6.1.4 CJ: Умовна команда переходу інструкція.....	88
6.1.5 CFEND: Інструкція для умовного повернення основної програми користувача.....	88
6.1.6 WDT: Інструкція для скидання сторожового таймера користувачської програми.....	89
6.1.7 EI: Увімкнути інструкцію переривання.....	89
6.1.8 DI: Вимкнути переривання інструкції.....	89

6.1.9 CIRET: Інструкція для умовного повернення переривання користувача програми.....	89
6.1.10 STOP: Інструкція для зупинки програми користувача.....	89
6.1.11 CALL: Інструкція для виклику підпрограми користувача.....	90
6.1.12 CSRET: Інструкція для умовного повернення підпрограми користувача.....	90
6.2 Інструкції з передачі даних інструкції.....	91
6.2.1 MOV: Інструкція з передачі даних word.....	91
6.2.2 DMOV: Інструкція передачі даних подвійного слова інструкція.....	91
6.2.3 RMOV: Інструкція передачі даних з рухомою комою інструкція.....	92
6.2.4 BMOV: Інструкція передачі даних блоком інструкція.....	92
6.2.5 FMOV: Інструкція заповнення блоку даних instruction.....	92
6.2.6 DFMov: Інструкція заповнення подвійного слова блоку даних instruction.....	93
6.2.7 SWAP: MSB/LSB замінити інструкція.....	93
6.2.8 XCH: Інструкція з перестановки слів.....	94
6.2.9 DXCH: Інструкція обміну подвійними словами.....	94
6.2.10 PUSH: Інструкція передачі даних.....	94
6.2.11 FIFO: Інструкція першим прийшов — першим обслужений.....	95
6.2.12 LIFO: Інструкція останній прийшов — перший пішов.....	96
6.2.13 WSFR: Інструкція зсуву рядка слів праворуч.....	96
6.2.14 WSFL: Інструкція зсуву рядка символів вліво instruction.....	97
6.3 Інструкції цілочисельних арифметичних операцій.....	98
6.3.1 ADD: Інструкція додавання цілих чисел instruction.....	98
6.3.2 SUB: Інструкція віднімання цілих чисел instruction.....	99
6.3.3 MUL: Інструкція множення цілих чисел instruction.....	99
6.3.4 DIV: Інструкція цілочисельного ділення instruction.....	99
6.3.5 SQT: Інструкція для добування квадратного кореня з SQT цілого числа.....	100
6.3.6 INC: Ціле число плюс один інструкція.....	100
6.3.7 DEC: Ціле число мінус один інструкція.....	101
6.3.8 VABS: Інструкція для отримання абсолютного значення цілого числа.....	101
6.3.9 NEG: Інструкція заперечення цілого числа instruction.....	102
6.3.10 DADD: Додавання довгих цілих чисел інструкція.....	102
6.3.11 DSUB: Віднімання довгих цілих чисел інструкція.....	102
6.3.12 DMUL: Інструкція множення довгих цілих чисел instruction.....	103
6.3.13 DDIV: Ділення довгих цілих чисел інструкція.....	103
6.3.14 DSQT: Інструкція для добування квадратного кореня з довгого цілого числа.....	104

6.3.15 DINC: Довге ціле число плюс один інструкція.....	104
6.3.16 DDEC: Довге ціле число мінус один інструкція.....	105
6.3.17 DVABS: Інструкція для отримання абсолютного значення цілого числа типу long.....	105
6.3.18 DNEG: Заперечення довгого цілого числа інструкція.....	106
6.3.19 SUM: Інструкція накопичення цілих чисел.....	106
6.3.20 DSUM: Інструкція накопичення довгого цілого числа instruction.....	108
6.4 Інструкції операцій з плаваючою комою.....	108
6.4.1 RADD: Інструкція додавання чисел з рухомою комою інструкція.....	108
6.4.2 RSUB: Інструкція віднімання чисел з рухомою комою instruction.....	109
6.4.3 RMUL: Інструкція множення чисел з рухомою комою інструкція.....	109
6.4.4 RDIV: Інструкція ділення чисел з рухомою комою інструкція.....	110
6.4.5 RSQT: Інструкція для добування квадратного кореня з числа з рухомою комою.....	110
6.4.6 RVABS: Інструкція для отримання абсолютного значення числа з рухомою комою числа.....	111
6.4.7 RNEG: Інструкція заперечення числа з рухомою комою instruction.....	111
6.4.8 SIN: Інструкція для отримання SIN числа з рухомою комою числа.....	111
6.4.9 COS: Інструкція для отримання COS числа з плаваючою комою.....	112
6.4.10 TAN: Інструкція для отримання TAN числа з рухомою комою number.....	112
6.4.11 POWER: Інструкція для піднесення до степеня числа з плаваючою комою числа.....	113
6.4.12 LN: Інструкція з отримання натурального логарифма числа з плаваючою комою.....	113
6.4.13 EXP: Інструкція для отримання степеня натурального числа для числа з рухомою комою.....	114
6.4.14 RSUM: Інструкція накопичення чисел з рухомою комою instruction.....	114
6.4.15 ASIN: Інструкція для отримання ASIN числа з плаваючою комою number.....	115
6.4.16 ACOS: Інструкція для отримання ACOS числа з рухомою комою числа.....	115
6.4.17 ATAN: Інструкція для отримання ATAN числа з рухомою комою числа.....	115
6.4.18 LOG: Інструкція для отримання десятичного логарифма числа з рухомою комою.....	116
6.4.19 RAD: Інструкція для перетворення кута з градусів у радіани conversion.....	116
6.4.20 DEG: Інструкція для перетворення кута в радіанах з плаваючою комою conversion.....	116
6.5 Інструкції перетворення значень.....	117
6.5.1 DTI: Інструкція щодо перетворення довгого цілого числа на ціле число.....	117
6.5.2 ITD: Інструкція для перетворення цілого числа на довге ціле число.....	117
6.5.3 FLT: Інструкція для перетворення цілого числа у число з плаваючою комою.....	118
6.5.4 DFLT: Інструкція для перетворення довгого цілого числа у число з плаваючою комою.....	118
6.5.5 INT: Інструкція для перетворення числа з рухомою комою на ціле число.....	118
6.5.6 DINT: Інструкція для перетворення числа з рухомою комою у ціле число типу long.....	119
6.5.7 BCD: Інструкція з перетворення слова у 16-бітний BCD код.....	119
6.5.8 DBCD: Інструкція для перетворення подвійного слова у 32-бітний BCD code.....	120
6.5.9 BIN: Інструкція для перетворення 16-бітного BCD коду у слово.....	120
6.5.10 DBIN: Інструкція для перетворення 32-бітного BCD коду на подвійне слово.....	121
6.5.11 GRY: Інструкція з перетворення слова у 16-бітний код Грея.....	121
6.5.12 DGRY: Інструкція для перетворення подвійного слова у 32-бітний код Грея code.....	121
6.5.13 GBIN: Інструкція з перетворення 16-бітового коду Грея у слово.....	122
6.5.14 DGBIN: Інструкція для перетворення 32-бітного коду Грея у подвійне слово.....	122
6.5.15 SEG: Інструкція з перетворення слова на 7-сегментний код.....	123
6.5.16 ASC: ASCII перетворення коду інструкція.....	123
6.5.17 ITA: Інструкція з перетворення 16-бітових шістнадцяткових даних у ASCII код.....	124
6.5.18 ATI: Інструкція для перетворення коду ASCII у 16-бітні шістнадцяткові дані.....	124
6.5.19 LCNV: Інструкція з перетворення проєкту instruction.....	125
6.5.20 RLCNV: Інструкція з конвертації проєкту з рухомою комою інструкція.....	126
6.6 Логічна операція над словами інструкції.....	127

6.6.1 WAND: Слово AND інструкція.....	127

6.6.2	WOR: INT OR інструкція.....	129
6.6.3	WXOR: Слово XOR інструкція.....	129
6.6.4	WINV: Слово INV інструкція.....	130
6.6.5	DWAND: Подвійне слово AND інструкція.....	130
6.6.6	DWOR: Подвійне слово OR інструкція.....	130
6.6.7	DWXOR: Подвійне слово XOR інструкція.....	131
6.6.8	DWINV: Подвійне заперечення слова інструкція.....	131
6.7	Інструкції побітового зсуву та циклічного зсуву.....	132
6.7.1	ROR: 16-бітна циклічна операція зсуву вправо інструкція.....	132
6.7.2	ROL: 16-бітна циклічна команда зсуву вліво інструкція.....	132
6.7.3	RCR: Інструкція для 16-бітного циклічного зсуву праворуч через прапор перенесення біт.....	133
6.7.4	RCL: Інструкція для 16-бітного циклічного зсуву вліво через прапор переносу біт.....	134
6.7.5	DROR: 32-бітна циклічна операція зсуву праворуч інструкція.....	134
6.7.6	DROL: інструкція циклічного зсуву вліво на 32 біти інструкція.....	135
6.7.7	DRCR: Інструкція для 32-бітного циклічного зсуву праворуч через прапор переносу біт.....	135
6.7.8	DRCL: Інструкція для 32-бітного циклічного зсуву вліво через прапор переносу біт.....	136
6.7.9	SHR: 16-бітова інструкція зсуву вправо instruction.....	136
6.7.10	SHL: 16-бітовий зсув вліво інструкція.....	137
6.7.11	DSHR: 32-бітна інструкція зсуву праворуч instruction.....	138
6.7.12	DSHL: 32-бітна інструкція логічного зсуву вліво.....	138
6.7.13	SFTR: Інструкція зсуву бітового рядка вправо.....	139
6.7.14	SFTL: Інструкція зсуву бітового рядка вліво instruction.....	139
6.8	Периферійні інструкції.....	141
6.8.1	FROM: Інструкція для зчитування слів із буфера спеціального модуля регістр.....	141
6.8.2	DFROM: Інструкція для зчитування подвійних слів із буфера спеціального модуля регістр.....	141
6.8.3	TO: Інструкція для запису слів із буфера спеціального модуля регістр.....	142
6.8.4	DTO: Інструкція для запису подвійних слів із буфера спеціального модуля регістр.....	143
6.8.5	VRRD: Інструкція для зчитування значення аналогового потенціометра.....	143
6.8.6	REFF: Інструкція щодо встановлення константи фільтрації вхідного сигналу.....	144
6.8.7	REF: Інструкція для негайного оновлення I/O.....	144
6.8.8	EROMWR: EEPROM напишіть інструкцію.....	145
6.8.9	PR: Друк інструкції.....	145
6.8.10	TKY: Інструкція з введення цифрових значень інструкція.....	146
6.9	Інструкції годинника реального часу.....	148
6.9.1	TRD: Інструкція зчитування годинника реального часу instruction.....	148
6.9.2	TWR: Інструкція запису годинника реального часу instruction.....	149
6.9.3	TADD: Інструкція додавання годинника інструкція.....	150
6.9.4	TSUB: Інструкція віднімання тактових імпульсів instruction.....	151
6.9.5	HOURL: Інструкція до хронографа інструкція.....	152
6.9.6	DCMP: (= < > <> >= <=)Інструкція порівняння дати інструкція.....	153
6.9.7	TCMP: (= < > <> >= <=)Інструкція порівняння часу інструкція.....	154
6.9.8	HTOS: Інструкція для перетворення даних у форматі години-хвилини-секунди на секунди.....	155
6.9.9	STOH: Інструкція для перетворення секунд у години-хвилини-секунди дані.....	155
6.10	Високошвидкісні I/O інструкції.....	156
6.10.1	HCNT: Інструкція для керування високошвидкісним лічильником.....	156
6.10.2	DHSCS: Інструкція з налаштування порівняння високошвидкісного лічильника.....	156
6.10.3	DHSCI: Інструкція для ініціювання переривання на основі порівняння високошвидкісного рахунку.....	157
6.10.4	DHSPI:Запуск переривання за інструкцією на основі порівняння абсолютних високошвидкісних вихідних позицій.....	159

6.10.5 DHSCR: Інструкція щодо скидання порівняння високошвидкісного лічильника.....	160
6.10.6 DHSZ: Інструкція порівняння діапазону високошвидкісного підрахунку інструкція.....	161

6.10.7 DHST: Інструкція порівняння таблиці високошвидкісного лічильника instruction.....	162
6.10.8 DHSP: Інструкція для імпульсного виходу на основі порівняння таблиці високошвидкісного лічильника	164
6.10.9 SPD: Інструкція з вимірювання частоти імпульсу.....	166
6.10.10 PLSY: Високошвидкісний імпульсний вихід імпульсу.....	167
6.10.11 PLSR: Інструкція для виходу імпульсного лічильника з прискоренням/уповільненням.....	168
6.10.12 PLS: Інструкція імпульсного виходу огинаючої сигналу.....	171
6.10.13 PWM: Інструкція імпульсного виходу ходу.....	172
6.11 Інструкції з розрахунку керування.....	174
6.11.1 PID: Інструкція функції.....	174
6.11.2 RAMP: Інструкція виходу сигналу рампи імпульсу.....	178
6.11.3 HACKLE: Інструкція щодо виходу сигналу пилоподібної хвилі імпульсу.....	179
6.11.4 TRIANGLE: Інструкція виходу сигналу трикутної хвилі імпульсу.....	180
6.11.5 ABSD: Абсолютне керування кулачком імпульсу.....	181
6.11.6 DABSD: Інструкція з абсолютного керування кулачком подвійного слова instruction.....	183
6.11.7 ALT: Альтернативна інструкція виводу імпульсу.....	184
6.12 Інструкції зв'язку.....	184
6.12.1 MODBUS: Інструкція з обміну даними головної станції імпульсу.....	184
6.12.2 XMT: Інструкція передавання через вільний порт instruction.....	185
6.12.3 RCV: Приймання імпульсу через вільний порт імпульсу.....	186
6.12.4 MODRW: Інструкція з читання/запису Modbus імпульсу.....	187
6.13 Перевірте імпульс.....	191
6.13.1 CCITT: Перевірте імпульс.....	191
6.13.2 CRC16: Перевірте імпульс.....	191
6.13.3 LRC: Перевірте імпульс.....	192
6.14 Розширені імпульси бітової обробки.....	193
6.14.1 ZRST: Інструкція зі скидання бітів у 0 у пакетному режимі.....	193
6.14.2 ZSET: Інструкція для скидання бітів у пакеті.....	193
6.14.3 DECO: Декодування імпульсу.....	193
6.14.4 ENCO: Інструкція з кодування імпульсу.....	194
6.14.5 BITS: Інструкція для підрахунку бітів у <A>слові/A>.....	194
6.14.6 DBITS: Інструкція для підрахунку бітів у подвійному слові.....	195
6.14.7 BON: Інструкція для визначення біта у слові.....	195
6.15 Контактні імпульси.....	195
6.15.1 BLD: Контакт слова біт LD імпульсу.....	195
6.15.2 BLDI: Контакт біта слова LD імпульсу.....	196
6.15.3 BAND: Контакт біта слова AND імпульсу.....	196
6.15.4 BANI: Контакт біта слова ANI імпульсу.....	197
6.15.5 BOR: Контакт біта слова OR імпульсу.....	197
6.15.6 BORI: Контакт біта слова ORI імпульсу.....	198
6.15.7 BOUT: Інструкція виходу бітової котушки слова.....	198
6.15.8 BSET: Бітова котушка слова імпульсу встановлення.....	198
6.15.9 BRST: Інструкція скидання бітової котушки слова instruction.....	199
6.16 Інструкції контакту порівняння.....	199
6.16.1 LD (=, <, >, <>, >=, <=): Порівняння цілих чисел LD*імпульсу.....	199
6.16.2 AND (=, <, >, <>, >=, <=): Порівняння цілих чисел AND *імпульсу.....	200
6.16.3 OR (=, <, >, <>, >=, <=): Порівняння цілих чисел OR*імпульсу.....	201
6.16.4 LDD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел LDD*імпульсу.....	202
6.16.5 ANDD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел ANDD*імпульсу.....	203

6.16.6 ORD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел ORD*інструкція.....	204
6.16.7 LDR: Інструкція порівняння чисел з рухомою комою instruction.....	205
6.16.8 ANDR: Інструкція порівняння чисел з рухомою комою instruction.....	206
6.16.9 ORR: Інструкція порівняння чисел з рухомою комою instruction.....	207
6.16.10 CMP: Інструкція для встановлення порівняння цілих чисел на ON.....	209
6.16.11 LCMP: Інструкція для встановлення порівняння довгих цілих чисел на ON.....	209
6.16.12 RCMP: Інструкція для налаштування порівняння чисел з рухомою комою на ON.....	210
6.17 Інструкції з масової обробки даних інструкції.....	210
6.17.1 BKADD: Інструкція з операції масового додавання даних інструкція.....	210
6.17.2 BKSUB: Операція віднімання масиву даних інструкція.....	211
6.17.3 BKCMP=,>,<,<>,<=,>=: Інструкція порівняння масиву даних instruction.....	211
6.18 Технічний паспорт інструкції.....	212
6.18.1 LIMIT: Інструкція з керування верхньою/нижньою межею інструкція.....	212
6.18.2 DBAND: Інструкція з керування зоною нечутливості instruction.....	213
6.18.3 ZONE: Інструкція з керування зоною.....	213
6.18.4 SCL: Інструкція з налаштування координат.....	214
6.18.5 SER: Інструкція з пошуку даних інструкція.....	215
6.19 Рядок символів інструкції.....	216
6.19.1 STRADD: Комбінація рядків інструкція.....	216
6.19.2 STRLEN: Інструкція для визначення довжини рядка.....	217
6.19.3 STRRIGHT: Інструкція для зчитування рядка зправа.....	217
6.19.4 STRLEFT: Інструкція для зчитування рядка з лівого боку.....	218
6.19.5 STRMIDR: Інструкція для зчитування будь-яких символів рядка.....	219
6.19.6 STRMIDW: Інструкція щодо заміни будь-яких символів рядка.....	219
6.19.7 STRINSTR: Пошук рядка інструкція.....	220
6.19.8 STRMOV: Інструкція з передачі рядка instruction.....	221
6.20 Інструкції з реєстрації файлів розширення instructions.....	221
6.20.1 LOADR: Інструкція для зчитування даних з файлу розширення регістр.....	221
6.20.2 SAVER: Інструкція для запису даних у регістр файла розширення.....	222
6.20.3 INITR: Інструкція з ініціалізації регістра розширення register.....	223
6.20.4 LOGR: Інструкція для входу в систему на розширенні реєстр.....	224
6.20.5 ITER: Інструкція з ініціалізації файлу розширення реєстру.....	225
6.21 Інструкції з позиціонування.....	227
6.21.1 ZRN: Інструкція повернення до нуля інструкція.....	227
6.21.2 PLSV: Інструкція щодо виходу імпульсів зі змінною швидкістю instruction.....	228
6.21.3 DRVI: Інструкція з керування відносним положенням інструкція.....	229
6.21.4 DRVA: Інструкція керування абсолютним положенням інструкція.....	230
6.21.5 DSZR: Інструкція для повернення до нуля з DOG.....	231
6.21.6 DVIT: Інструкція переривання позиціонування instruction.....	232
6.21.7 STOPDV: Зупинка імпульсного виходу інструкція.....	233
6.21.8 LIN: Інструкція лінійної інтерполяції траєкторії instruction.....	234
6.21.9 CW: Траєкторія дуги за годинниковою стрілкою інтерполяція.....	237
6.21.10 CCW: Інструкція інтерполяції дугової траєкторії проти годинникової стрілки instruction.....	238
6.21.11 MOVELINK: Інструкція синхронного керування інструкція.....	241
6.21.12 GEARBOX: Інструкція з електронного редуктора instruction.....	243
6.22 Інструкції з обробки даних.....	244
6.22.1 MEAN: Означає інструкцію.....	244
6.22.2 WTOB: Інструкція з розділення даних у байтових одиницях instruction.....	244
6.22.3 BTOW: Інструкція комбінування даних байтових одиниць instruction.....	245

6.22.4 UNI: Інструкція для об'єднання 4bits 16-бітних даних.....	246

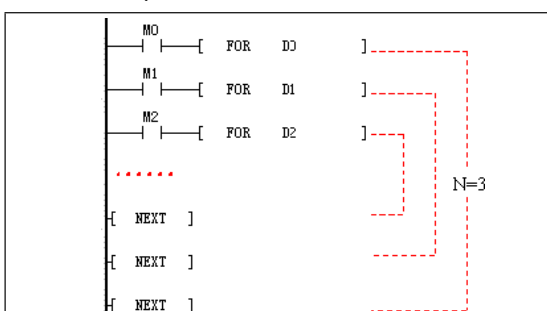
6.22.5 DIS: Інструкція для розділення 4bitsof 16-бітових даних.....	247
6.22.6 ANS: Інструкція встановлення сигналу тривоги instruction.....	248
6.22.7 ANR: Інструкція скидання сигналу тривоги інструкція.....	249
6.23 Інші інструкції.....	249
6.23.1 RND: Інструкція для генерування випадкових чисел.....	249
6.23.2 DUTY: Інструкція для генерування часових імпульсів.....	250

послідовно виконується 100 разів. D0 збільшується на одиницю 100 разів. Коли цикл завершується, D0=100.

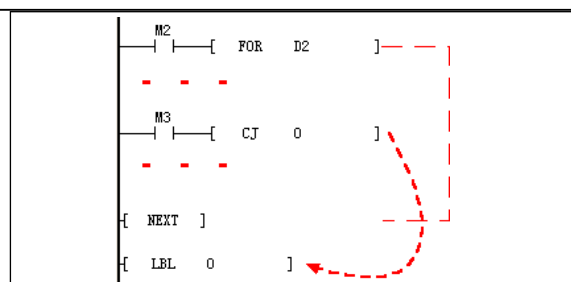
- Примітка

1. Інструкцію FOR-NEXT необхідно використовувати парами в тілі програми (POU), інакше користувацька програма не може бути скомпільована належним чином.

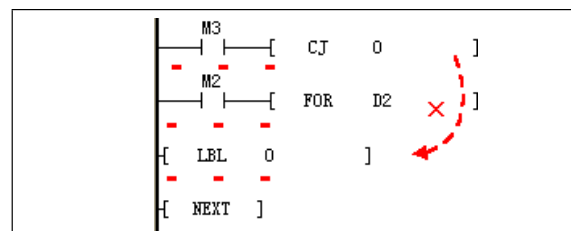
2. Підтримка вкладеності кількох структур FOR-NEXT. Пристрій CPU серії VC2 підтримує вкладеність максимум 8 структур FOR-NEXT. (На наступному рисунку показано вкладеність 3 структур FOR-NEXT.)



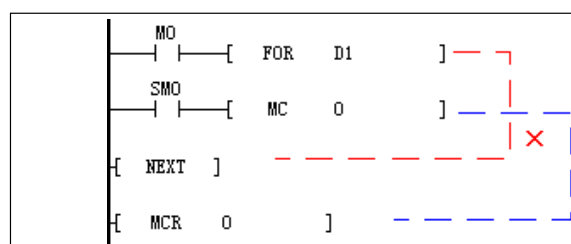
3. Ви можете використовувати інструкцію умовного переходу (CJ) для виходу з тіла циклу, щоб завчасно завершити виконання тіла циклу, як показано в LAD нижче:



4. Заборонено використовувати оператор CJ для переходу в тіло наступного циклу, а наведений нижче LAD не може бути скомпільований коректно:



5. Перетин тіла структури MC-MCR та тіла структури FOR-NEXT заборонено, і LAD нижче не може бути скомпільовано коректно:



- Примітка

Виконання циклу FOR-NEXT потребує багато часу. Чим більше циклів або чим більше інструкцій міститься в тілі циклу, тим більше часу це займає. Щоб запобігти помилці тайм-ауту операції, необхідно використовувати інструкцію WDT всередині а трудомісткий цикл корпусу.

6.1.3 LBL: Визначення мітки переходу інструкція

LAD:		Застосовна модель		VC1S VC1 VC2 VC3 VC5									
[LBL (S)]		Зумовлений прапор біт											
IL: LBL (S)		Довжина кроку		3									
Операнд	Тип	Застосовний програмний елемент											Індексація
S	INT	Складається											
		має											

- Операнд опис
S:Маркування значення

- Опис функції
1. Визначення мітки, значенням якої є S.

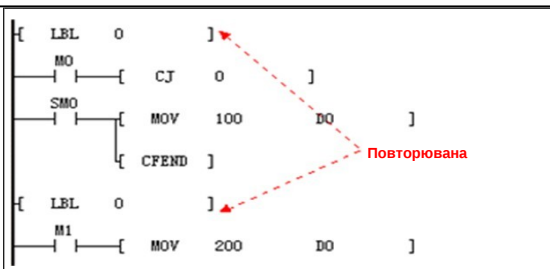
- 2. Не використовується для фактичної роботи. Він лише вказує конкретну позицію переходу для інструкції CJ.

- Примітка

1. Діапазон значення маркування S: $0 \leq S \leq 127$.

2. Дві мітки з однаковим визначенням не дозволяються в межах одного ROU, інакше програму користувача неможливо буде скомпілювати. Проте повторні визначення міток дозволені в різних ROUs (наприклад, у різних підпрограмах).

- Екземпляр застосунку неправильної програми



6.1.4 CJ: Умовна команда переходу інструкція

LAD: [CJ (S)]					Застосовна модель		VC1S VC1 VC2 VC3 VC5							
					Зумовлений прапор біт									
IL: CJ (S)					Довжина кроку		3							
Операнд	Тип	Застосовний програмний елемент											Індексація	
S	INT	Складається немає												

- Операнд опис
S: Маркування значення

- Опис функції
 - Коли потік енергії є дійсним, програма користувача переходить до виконання інструкції з номером S, допустимою міткою.
 - Коли потік енергії недійсний, операція переходу відсутня, і система виконує інструкцію після CY послідовно.

- Примітка
 - Мітка S ($0 \leq S \leq 127$), на яку здійснюється перехід інструкцією CJ, повинна бути дійсною та визначеною міткою, інакше користувацька програма не зможе бути коректно скомпільована.
 - Заборонено використовувати інструкцію CJ для переходу до структури FOR-NEXT.
 - Дозволяється використовувати інструкцію CJ для виходу з або входу в структуру MC-MCR або стану SFC. Однак така операція порушує логіку структури MC-MCR та стану SFC і ускладнює програму. Тому така операція не рекомендується.

- Екземпляр застосунку



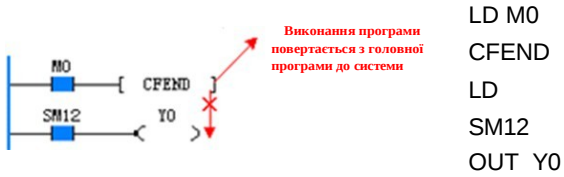
- Коли початкова умова M0=OFF та M1=ON, інструкція CJ 0 не пропускається, а D0=100. Після виконання інструкції CFEND програма достроково виходить з основної програми, інструкції LD M1 та MOV 200 D0 не виконуються.
- Коли M0=ON, M1=ON, програма виконує інструкцію CJ 0, пропускає MOV 100 D0 та CFEND інструкції. Після переходу до інструкції LBL 0, програма виконує інструкцію MOV 200 D0, і тепер D0=200.

6.1.5 CFEND: Інструкція для умовного повернення основної програми користувача

LAD:	Застосовна модель		VC1S VC1 VC2 VC3 VC5				
	Зумовлений прапор						
[CFEND]		Довжина кроку					
CFEND							

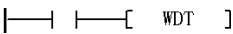
- Опис функції
 - Коли потік енергії інструкції є дійсним, головна програма повертається до системи з поточного циклу сканування (головна програма користувача виконується системою багаторазово відповідно до циклу сканування), а інструкції в наступній головній програмі не виконуються.
 - Якщо потік енергії інструкції є недійсним, ця інструкція не виконує жодної дії, а наступні інструкції виконуються в порядку черги.
- Примітка
Інструкцію CFEND необхідно використовувати в основній програмі користувача, інакше програму неможливо буде скомпілювати.

● Екземпляр застосунку



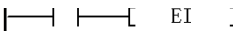
Коли програма виконується, якщо M0=OFF, інструкція CFEND не генерує жодної дії, наступні інструкції LD SM12 та OUT Y0 виконуються, і ви можете бачити, як Y0 періодично блимає. Якщо M0=ON, інструкція CFEND генерує дію, хід програми завчасно повертається до системи з основної програми, наступні інструкції LD SM12 та OUT Y0 не виконуються, і Y0 більше не блимає періодично.

6.1.6 WDT: Інструкція для скидання сторожового таймера користувацької WDTпрограми

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Зумовлений прапор біт	
IL: WDT	Довжина кроку	1

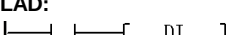
- Опис функції
Коли потік енергії є дійсним, ця інструкція очищує значення таймінгу сторожового таймера в користувацькій програмі, і сторожовий таймер системної користувацької програми перезавантажує відлік часу.

6.1.7 EI: Увімкнути інструкцію переривання

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Зумовлений прапор біт	
IL: EI	Довжина кроку	1

- Опис функції
 - Коли потік енергії є дійсним, переривання у поточному циклі сканування дозволені.
 - Коли інструкція EI є дійсною, запит на переривання дозволяється додавати до черги запитів на переривання, очікуючи на відповідь системи.

6.1.8 DI: Вимкнути переривання інструкції

LAD: 	Застосовно модель	VC1S VC1 VC2 VC3 VC5
	Під впливом прапорця біта	
IL: DI	Крок довжина	1

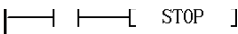
- Опис функції
 - Коли потік енергії є дійсним, прапорець дозволу глобального переривання є недійсним, тобто глобальне переривання вимкнено.
 - Коли прапорець дозволу глобальних переривань недійсний, події переривань не можуть генерувати жодного запиту на переривання.
- Примітка
Коли інструкція запиту на заборону переривань є дійсною, система все ще реагує на необроблені запити на переривання в черзі запитів на переривання, але нові події переривання не можуть генерувати запити на переривання.

6.1.9 CIRET: Інструкція для умовного повернення переривання користувача програми

LAD: 	Застосовно модель	VC1S VC1 VC2 VC3 VC5
	Під впливом прапорця біта	
IL: CIRET	Крок довжина	1

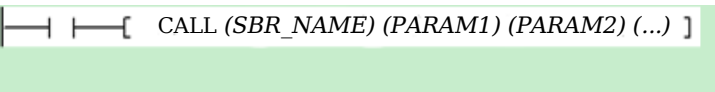
- Опис функції
Коли потік енергії є дійсним, система виходить із поточної програми переривання заздалегідь.

6.1.10 STOP: Інструкція для зупинки програми користувача

LAD: 	Застосовно модель	VC1S VC1 VC2 VC3 VC5
	Під впливом прапорця біта	
IL: STOP	Крок довжина	1

- Опис функції
Коли потік енергії є дійсним, система зупиняє виконання користувацької програми негайно.

6.1.11 CALL: Інструкція для виклику підпрограми користувача

LAD: 	Придатна модель	VC1S VC1 VC2 VC3 VC5
	Під впливом прапорця біта	
IL: CALL (назва підпрограми) (параметр підпрограми 1) (параметр підпрограми 2) ...	Крок довжина	Визначається параметрами виконується підпрограмою

- Опис функції
Коли потік енергії є дійсним, система викликає призначену підпрограму для виконання, а після виконання повертається до інструкцій, що йдуть після інструкції CALL, щоб продовжити виконання.

- Примітка

- Підпрограма, що викликається інструкцією CALL, має бути заздалегідь визначена у користувацькій програмі. Якщо в інструкції CALL зустрічається невизначена підпрограма, програму неможливо скомпілювати.
- Тип елемента операндів в інструкції CALL має збігатися з типом даних, визначеним у таблиці локальних змінних підпрограми, інакше програму неможливо буде скомпілювати.

Цей екземпляр застосування демонструє деякі недопустимі збіги.

Екземпляр застосування 1: У таблиці локальних змінних підпрограми SBR1 тип даних операнда 1 — DINT/DWORD.

Наступні способи використання є незаконними:

- CALL SBR1 Z0 (Тип даних елемента Z не може бути DINT/DWORD)
- CALL SBR1 C199 (Тип даних елемента від C0 до C199 не може бути DINT/DWORD)
- CALL SBR1 K2X0 (адресація Kn, де $1 \leq n \leq 3$, тип даних не може бути DINT/DWORD)

Приклад застосування 2: У таблиці локальних змінних підпрограми SBR1 тип даних операнда 1 — INT/WORD.

Наступні способи використання є незаконними:

- CALL SBR1 C200 (Тип даних елементів від C200 до C255 не може бути INT/WORD)
- CALL SBR1 K2X0 (адресація Kn $4 \leq n \leq 8$, тип даних не може бути INT/WORD)

- Тип елемента операндів в інструкції CALL має збігатися з типом даних, визначеним у таблиці локальних змінних підпрограми, інакше програму неможливо буде скомпілювати.

Цей екземпляр застосування демонструє деякі недопустимі збіги.

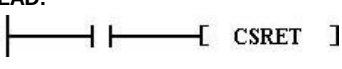
Приклад застосування: у таблиці локальних змінних SBR1 підпрограми тип даних операнда 1 є OUT або IN_OUT.

Наступні способи використання є незаконними:

- CALL SBR1 321 (константу неможливо змінити, тому вона не збігається з OUT або типом IN_OUT операнда)
- CALL SBR1 K4X0 (K4X0 є доступним лише для читання, тому він не збігається з OUT або типом IN_OUT операнда)
- CALL SBR1 SD0 (SD0 є лише для читання, тому вона не збігається з OUT або типом IN_OUT операнда)

- Кількість операндів в інструкції CALL повинна збігатися з таблицею локальних змінних підпрограми, інакше програму неможливо буде скомпілювати.

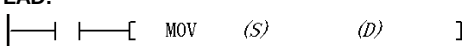
6.1.12 CSRET: Інструкція для умовного повернення підпрограми користувача

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Зумовлений прапор біт	
IL: CSRET	Довжина кроку	1

- Опис функції
Коли потік енергії є дійсним, система виходить із поточної підпрограми та повертається до підпрограми попереднього рівня.

6.2 Інструкції з передачі даних інструкції

6.2.1 MOV: Інструкція з передачі даних word

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: MOV (S) (D)									Довжина кроку			5						
Операнд	Тип	Застосовний програмний елемент														Індексація		
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓		
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	Z	R	✓		

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
 Присвоєння вмісту **S** до **D** та збереження значення **S**, коли потік енергії є дійсним.
- Примітка
 1. MOV інструкція підтримує два типи цілих чисел: одне зі знаком, а інше без знака. Якщо обидва операнди інструкції є програмними елементами, то тип даних — ціле число зі знаком. Якщо операнд-джерело інструкції є довгим цілим числом зі знаком, наприклад (−10, +100), то операнд-призначення є цілим числом зі знаком. Якщо операнд-джерело є довгим цілим числом без знака, наприклад (100, 45535), то операнд-призначення також є цілим числом без знака.
 2. Відповідний програмний елемент C підтримує C0 до C199 тільки.
- Екземпляр застосунку



Присвоєння вмісту D0 до D10, коли X0=ON. У цьому випадку, D10=500.

6.2.2 DMOV: Інструкція передачі даних подвійного слова інструкція

LAD: [DMOV (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: DMOV (S) (D)									Довжина кроку			7						
Операнд		Тип	Застосовний програмний елемент													Індексація		
S		DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V	Z	R	✓	
D		DINT			KnY	KnM	KnS	KnLM		D	SD	C		V		R	✓	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
 Присвоєння вмісту **S** до **D** та збереження значення **S** за умови, що потік енергії є дійсним.

- Примітка
 1. DMOV інструкція підтримує два типи довгих цілих чисел: одне зі знаком, а інше без знака. Якщо обидва операнди інструкції є програмними елементами, то тип даних — ціле число зі знаком. Якщо вихідний операнд інструкції є довгим цілим числом зі знаком, як-от (−10, +100), то цільовий операнд також є цілим числом зі знаком. Якщо вихідний операнд є довгим цілим числом без знака, як-от (100, 45535), то цільовий операнд також є цілим числом без знака.
 2. Відповідний програмний елемент S підтримує 32-бітний елемент S лише.
- Екземпляр застосунку



Дуп копіювання вмістуL (D0, D1) доD (D10, D11)X колиD X0=ON. У цьому а випадку, (D10, D11)=D50M0O00V. D0 D10

6.2.3 RMOV: Інструкція передачі даних з рухомою комою інструкція

LAD:										Застосовна модель VC1S VC1 VC2 VC3 VC5									
— [RMOV (S) (D)]										Зумовлений прапор									
RMOV (S) (D)										Довжина кроку									
Операнд		Тип		Застосовний програмний елемент												Індексація			
S	REAL	Складається немає								D				V		R	✓		
D	REAL									D				V		R	✓		

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Екземпляр застосунку



Присвоєння вмісту (D0, D1) до (D10, D11), коли X0=ON. У цьому випадку (D10, D11)=50000.5.

- Опис функції
Присвоєння вмісту **S** до **D** та збереження значення **S** за умови, що потік енергії є дійсним.

6.2.4 BMOV: Інструкція передачі даних блоком інструкція

LAD: [BMOV (S1) (D) (S2)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: BMOV (S1) (D) (S2)									Довжина кроку			7						
Операнд		Тип	Застосовний програмний елемент													Індексація		
S1		INT		KnX	KnY	KnM	KnS	KnLM		D	SD	C	T	V		R	√	
D		INT			KnY	KnM	KnS	KnLM		D		C	T	V		R	√	
S2		INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	

- Операнд опис
S1: Джерело операнда, початкова одиниця даних блоку
D: Операнд призначення, початкова одиниця блоку даних
S2: Розмір блоку даних block

починаючи з **D** та зберігаючи вміст одиниць **S2**, починаючи з **S1**, коли потік енергії є дійсним.

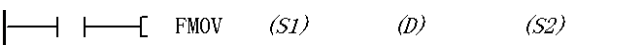
- Екземпляр застосунку



Присвоєння вмісту 10 одиниць, починаючи з D0, 10 одиницям, починаючи з D100, коли X0=ON. У цьому випадку D100=D0, D101=D1, ..., D109=D9

- Опис функції
Призначення той вміст від **S2** одиниці починаючи з **S1** до **S2** одиниці

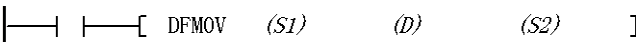
6.2.5 FMOV: Інструкція заповнення блоку даних instruction

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: FMOV (S1) (D) (S2)									Довжина кроку			7						
Операнд		Тип	Застосовний програмний елемент													Індексація		
S1		INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
D		INT			KnY	KnM	KnS	KnLM		D		C	T	V		R	√	
S2		INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	

- Операнд опис
S1: Вихідний операнд, початкова одиниця даних блоку
D: Операнд призначення, початкова одиниця блоку даних
S2: Розмір блоку даних block
- Опис функції
 Заповнення вмісту S1 у пристроях S2, починаючи з D, та збереження вмісту S1, коли потік енергії є дійсним.
- Примітка
 1. Коли **S1**, **D** та **S2** використовують елементи C, допустимий діапазон становить від C0 до C199.
 2. **S2** більше за 0.
 3. Коли **S1** та **D** обидва є адресацією Kn, Kn має бути рівним.
- Екземпляр застосунку

 LD X0
 FMOV D0 D100 10
 Заповнення вмісту D0 у 10 одиницях, починаючи з D100, коли X0=ON. У цьому випадку D100=D101=.....=D109=D0=500.

6.2.6 DFMOV: Інструкція заповнення подвійного слова блоку даних instruction

LAD: 										Застосовна модель			VC1S VC1 VC2 VC3 VC5					
										Зумовлений прапор біт								
IL: DFMOV (S1) (D) (S2)										Довжина кроку			9					
Операнд	Тип	Застосовний програмний елемент														Індексація		
S1	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√		
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√		
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		

- Операнд опис
S1: Початок операнда джерела
D: Операнд призначення, початкова одиниця блоку даних
S2: Розмір блоку даних block
- Опис функції
 Заповнення вмісту **S1** в одиницях **S2** починаючи з **D** та збереження вмісту **S1** коли потік енергії є дійсним.
- Примітка
 1. Коли **S1**, **D** та **S2** використовують елементи C, підтримуються лише 32-розрядні елементи C.
 2. **S2** більше за 0.
 3. Коли **S1** та **D** обидва є адресацією Kn, Kn має бути рівним.
- Екземпляр застосунку

 LD X0
 DFMOV D0 D10 10
 Заповнення вмісту (D0,D1) у 10×2 одиницях, починаючи з D10, коли X0=ON. У цьому випадку (D10,D11)= (D12,D13)=.....= (D28,D29)=

(D0,D1)=100000.

6.2.7 SWAP: MSB/LSB замінити інструкція

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт										
IL: SWAP (D)									Довжина кроку				3						
Операнд		Тип		Застосовний програмний елемент													Індексація		
D		INT				KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	

- Операнд опис

D: Цільовий операнд, елемент-слово, у якого старший байт (MSB) та молодший байт (LSB) місцями змінені

- Опис функції
- Збереження значення, отриманого після MSB та LSB вмісту D є

перемкнuto в D, коли потік енергії дійсний.

- Екземпляр застосунку



Збереження значення, отриманого після того, як MSB та LSB вмісту D0=0x1027 (4135) міняються місцями в D0, коли X0=ON. У цьому випадку D0=0x2710 (10000).

6.2.8 XCH: Інструкція щодо заміни слів інструкція

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: XCH (D1) (D2)									Довжина кроку			5						
Операнд		Тип	Застосовний програмний елемент													Індикація		
D1		INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	
D2		INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	

- Операнд опис
D1: Операнд призначення 1
D2: Операнд призначення 2
- Опис функції
Обмін вмістом **D1** із вмістом **D2**, а потім збереження отриманого значення в **D1** та **D2** одиницях, коли потік енергії є дійсним.
- Примітка

При використанні режиму адресації Kn, значення Kn у **D1** та **D2** мають бути однаковими.

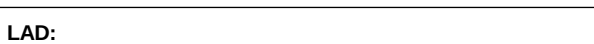
- Екземпляр застосунку



Обмін вмістом D0 та D10, коли X0=ON. Перед виконанням: D0=5000, D10=1000.

Після виконання: D0=1000, D10=5000.

6.2.9 DXCH: Інструкція обміну подвійними словами

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: DXCH (D1) (D2)									Довжина кроку			7						
Операнд		Тип		Застосовний програмний елемент													Індикація	
D1		DINT				KnY	KnM	KnS	KnLM		D		C	T	V		R	√
D2		DINT				KnY	KnM	KnS	KnLM		D		C	T	V		R	√

- Операнд опис
D1: Операнд призначення 1;
D2: Операнд призначення 2
- Опис функції
Обмін вмістом **D1** із вмістом **D2**, а потім збереження отриманого значення в **D1** та **D2** одиницях, коли потік енергії є дійсним.
- Примітка

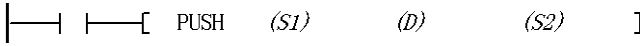
При використанні режиму адресації Kn, значення Kn у **D1** та **D2** мають бути однаковими.

- Екземпляр застосунку



Обмін вмістом (D0,D1) з (D10,D11) коли X0=ON. Перед виконанням: (D0,D1) =5000000, (D10,D11) =1000000. Після виконання: (D0,D1) =1000000, (D10,D11) =5000000.

6.2.10 PUSH: Інструкція передачі даних

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Зумовлений прапор біт	

IL: PUSH (S1) (D) (S2)									Довжина кроку			7					
Операнд	Тип	Застосовний програмний елемент														Індексція	
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	
D	INT								D				V		R	✓	
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	

- Операнд опис

S1: Натисніть значення

D: Кількість елементів у стеку зберігання. Номер елемента вказує на позицію дна стека.

S2: Розмір стека

- Опис функції

1. Занесення значення **S1** на вершину стека, основою якого є **D**, та збільшення значення **D** на 1, коли потік енергії є дійсним. У цьому випадку номер елемента вершини стека дорівнює номеру **D** + значення **D**.

2. Встановлення біта прапора перенесення операції (SM81) в 1, і невиконання операції запису в стек, коли значення **D** дорівнює значенню **S2** і все ще виконуються інструкції запису в стек.

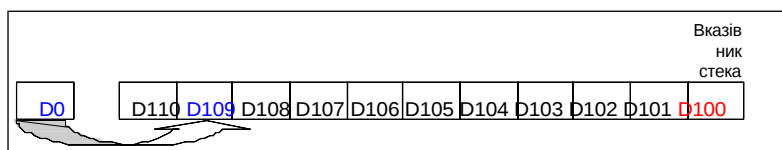
- Примітка

1. Коли стек визначено як недопустимий (наприклад, коли розмір стека не перевищує 0, кількість елементів у стеку менша за 0 або розмір стека виходить за межі ліміту), система повідомляє про помилку, вказуючи на те, що визначення стека, з яким виконується операція, є недійсним.

2. Розмір стека не включає нижній елемент стека (елемент, позначений **D**).

3. **S2** вказує на розмір стека, і діапазон більший за 0.

- Екземпляр застосунку



1. Перенесення вмісту D0 у стек, дном якого є D100, коли M0=ON.

2. Перед виконанням: D0=1000, D100=8, D109=0.

3. Після виконання: D0=1000, D100=9, D109=1000.

6.2.11 FIFO: Інструкція першим прийшов — першим обслужений

LAD:									Застосовна модель VC1S VC1 VC2 VC3 VC5							
FIFO FIFO (D1) (D2) (S) Зумовлений прапор																
FIFO (D1) (D2) (S)									Довжина кроку							
Операнд	Тип	Застосовний програмний елемент													Індексація	
D1	INT								D				V		R	✓
D2	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓
S	INT	Складається немає KnX		KnY	KnM	KnS	KnLM	KnSM	D SD		C	T	V	Z	R	✓

- Операнд опис

D1: Кількість елементів у стеку, а початковий елемент стека — це його порядковий номер плюс 1.

D2: Регістр зберігання для вилученого значення

S: Розмір черги

- Опис функції

1. Присвоєння початкового значення стека слів, що починається з **D1** (вміст блоку після

D1), до **D2**, віднімання 1 від значення **D1**, переміщення вмісту **S** блоків після **D1** вперед та заповнення 0 останнього блоку, коли потік енергії дійсний.

2. Встановлення біта прапорця нуля (SM80) у 1, коли D1 дорівнює 0, тобто стек порожній.

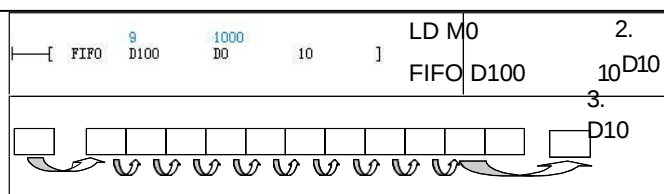
- Примітка

1. Коли стек визначено як недопустимий (наприклад, коли розмір стека не перевищує 0, кількість елементів у стеку менша за 0 або розмір стека виходить за межі ліміту), система повідомляє про помилку, вказуючи на те, що визначення стека, з яким виконується операція, є недійсним.

2. Розмір стека не включає нижній елемент стека (елемент, позначений **D1**).

3. **S** вказує на розмір стека, і діапазон більший за 0.

- Екземпляр застосунку



1. Заповнення вмісту D101in D0, переміщення вмісту D101 до D110forward, та заповнення 0 у D110, коли M0=ON.

Перед виконання: D0=0, D100=10, D101=1000, 2=2000, ..., D109=9000, D110=10000.

Після виконання: D0=1000, D100=9, D101=2000, 2=3000, ..., D109=10000, D110=0.

6.2.12 LIFO: Інструкція останній прийшов — перший пішов

LAD: LIFO (D1) (D2) (S)									Застосовна модель			VC1S VC1 VC2 VC3 VC5					
									Зумовлений прапор біт								
IL: LIFO (D1) (D2) (S)									Довжина кроку			7					
Операнд	Тип	Застосовний програмний елемент														Індексація	
D1	INT								D				V		R	√	
D2	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	

- Операнд опис
 - D1**: Кількість елементів у черзі, а початковим елементом стека є його номер елемента плюс 1
 - D2**: Регістр зберігання для вилученого значення
 - S**: Розмір черги
- Опис функції
 - Присвоєння вмісту вершини стека з **D1** як основою стека до **D2** та віднімання 1 від значення **D1**, коли потік енергії дійсний.
 - Встановлення біта нульового прапорця (SM80) у 1, коли **D1** дорівнює 0, тобто стек порожній.
- Примітка
 - Коли стек визначено як недопустимий (наприклад, коли розмір стека не перевищує 0, кількість елементів у стеку менша за 0 або розмір стека виходить за межі ліміту), система повідомляє про помилку, вказуючи на те, що визначення стека, з яким виконується операція, є недійсним

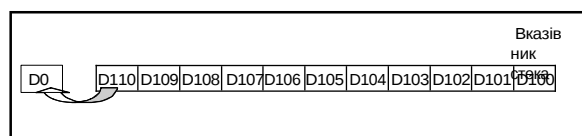
2. Розмір стека не включає нижній елемент стека (елемент, позначений **D1**).

3. **S** вказує на розмір стека, і діапазон більший за 0.

- Екземпляр застосунку



LD M0
LIFO D100 D0 10



- Присвоєння вмісту D110 до D0 та збереження вмісту D101 до D110, коли M0=ON.
- Перед виконанням: D0=0, D100=10, D101=1000, D102=2000, ..., D109=9000, D110=10000.
- Після виконання: D0=10000, D100=9, D101=1000, D102=2000, ..., D109=9000, D110=10000.

6.2.13 WSFR: Інструкція зсуву рядка символів праворуч instruction

LAD:									Застосовна модель										VC1	S	VC1	VC2	VC3	VC5		
									Зумовлений прапор										Прапор перенесення, прапор позики							
WSFR								(S1)	(D)	(S2)	(S3)	Довжина кроку														
Операнд	Тип	Застосовний програмний елемент															Індексація									
S1	INT			KnX	KnY	KnM	KnS	KnLM		D	SD	C	T	V			R	√								
D	INT				KnY	KnM	KnS	KnLM		D			C	T	V		R	√								
S2	INT	Складається немає		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z		R	√								

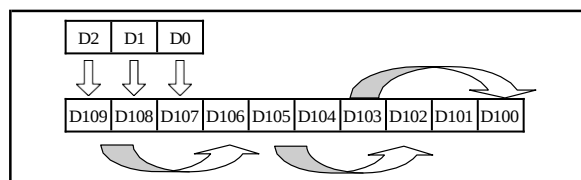
S3	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис
S1: Джерело операнда
D: Операнд призначення, початковий елемент рядка string
S2: Розмір черги цільового слова
S3: Кількість слів, заповнених при зсуві рядка слів праворуч
- Опис функції
Зсув вмісту **S2** одиниць, починаючи з **D**, праворуч на **S3** одиниць у вимірі слова, відкидаючи **S3** дані з крайнього правого боку, та зсув вмісту **S3** одиниць, починаючи з **S1**, до лівого краю рядка слів, коли потік енергії є дійсним.
- Примітка
 - Порядок зліва направо. Елементи з малими номерами знаходяться справа, а елементи з великими номерами — зліва.
 - Обидва **S2** та **S3** не менше 0.
 - S2** є не меншим за **S3**.
 - Коли **S1** та **D** обидва використовують адресацію Kn, значення Kn має бути однаковим.
- Екземпляр застосунку



LD X0

WSFR D0 D100 10 3



- Зсув вмісту 10 одиниць, починаючи з D100, праворуч на три одиниці у форматі слова, з відкиданням даних одиниць від D102 до D100 на крайньому правому боці, та зсув вмісту трьох одиниць, починаючи з D0, до лівого краю рядка слів, коли M0=ON.
- Перед виконанням: D2=300, D1=200, D0=100. D109=10000, D108=9000, D107=8000, D106=7000, D105=6000, D104=5000, D103=4000, D102=3000, D101=2000, D100=1000.
- Після виконання: збереження вмісту від D0 до D2. D2=300, D1=200, D0=100. D109=300, D108=200, D107=100, D106=10000, D105=9000, D104=8000, D103=7000, D102=6000, D101=5000, D100=4000.

6.2.14 WSFL: Інструкція зсуву рядка символів вліво instruction

LAD:										Застосовна модель					
										VC1	VC1	VC2	VC3	VC5	
										Зумовлений прапорець					
										Прапорець нуля, прапорець перенесення та прапорець позики прапор					
WSFL (S1) (D) (S2) (S3)										Довжина кроку					
Операнд	Тип	Застосовний програмний елемент								Індикація					
S1	INT		KnX	KnY	KnM	KnS	KnLM		D	SD	C	T	V		✓
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V		✓
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	✓
S3	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	✓

- Операнд-опис

S1: Джерело операнда

D: Операнд призначення, початковий елемент рядка слів

S2: Розмір черги слів призначення **S3:** Кількість слів, заповнених при зсуві рядка слів праворуч

- Опис функції

Зсув вмісту **S2** одиниць починаючи з **D** ліворуч на **S3** одиниць у словах, відкидаючи дані **S3** з лівого краю, та зсув вмісту **S3** одиниць починаючи з **S1** до правого кінця рядка слів, коли потік енергії дійсний.

- Примітка

1. Порядок зліва направо. Елементи з малими номерами знаходяться праворуч, а елементи з великими номерами — ліворуч.

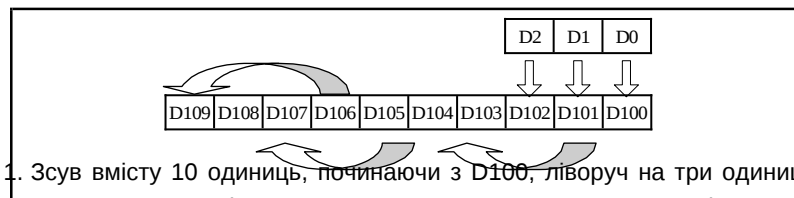
2. І **S2**, і **S3** не менше 0.

3. **S2** не менше ніж **S3**.

4. Коли **S1** та **D** обидва використовують адресацію Kn, Kn має бути однаковим.

- Екземпляр застосування

LD X0
WSFL D0 D100 10 3]



1. Зсув вмісту 10 одиниць, починаючи з D100, ліворуч на три одиниць в одиницях слова, з відкиданням даних одиниць D109–D107 з лівого краю, та зсув вмісту трьох одиниць, починаючи з D0, до правого кінця рядка слів, коли X0=ON.

2. Перед виконанням: D0=100, D1=200, D2=300. D109=10000, D108=9000, D107=8000, D106=7000, D105=6000, D104=5000, D103=4000, D102=3000, D101=2000, D100=1000.

3. Після виконання: Збереження вмісту від D0 до D2. D2=300, D1=200, D0=100. D109=7000, D108=6000, D107=5000, D106=4000, D105=3000, D104=2000, D103=1000, D102=300, D101=200, D100=100.

6.3 Інструкції цілочисельних арифметичних операцій

6.3.1 ADD: Інструкція додавання цілих чисел instruction

LAD:		Застосовна модель													
		VC1S VC1 VC2 VC3 VC5													
		Зумовлений прапорець нуля, прапорець перенесення та позики прапор													
ADD (S1) (S2) (D)		Довжина кроку													
Операнд	Тип	Застосовний програмний елемент													
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R

- Опис операнда **S1:**

Операнд джерела 1

S2: Операнд

джерела 2

D: Цільовий операнд

- Опис функції

1. Збільшення **S1** на **S2** та присвоєння результату операції **D** коли потік енергії є дійсним.

2. Встановлення біта прапора перенесення (SM81), коли результат операції (**D**) більший за 32767. Встановлення біта нульового прапора (SM80), коли результат операції дорівнює 0. Встановлення

прапора позики

біт прапорця (SM82), коли результат операції менший за — 32768.

● Екземпляр застосунку

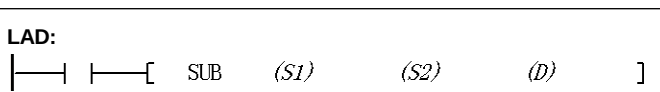


LD X0

ADD D0 D1 D10

Збільшення D0 (1000) на D1 (2000) та присвоєння результату операції D10, коли X0=ON. У цьому випадку, D10=3000.

6.3.2 SUB: Інструкція віднімання цілих чисел instruction

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Зумовлений прапор біт		Прапорець нуля, прапорець перенесення та прапорець позики прапор					
IL: SUB (S1) (S2) (D)										Довжина кроку		7					
Операнд	Тип	Застосовний програмний елемент														Індексация	
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓	

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
- Опис функції
 - Віднімання **S1** від **S2** та присвоєння результату операції **D** за умови валідного потоку енергії.
 - Встановлення біта прапора перенесення (SM81), коли результат операції (**D**) більший за 32767. Встановлення біта нульового прапора (SM80), коли результат операції дорівнює 0. Встановлення позики біт прапорця (SM82), коли результат операції менший за – 32768.
- Екземпляр застосунку


```
LD X0
SUB D0 D1 D10
```

Віднімання D0 (1000) від D1 (2000) та присвоєння результату операції D10, коли X0=ON. У цьому випадку, D10= – 1000.

6.3.3 MUL: Інструкція множення цілих чисел instruction

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
IL: MUL (S1) (S2) (D)									Довжина кроку			8						
Операнд Д	Тип	Застосовний програмний елемент														Індекс ація		
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√		

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
- Опис функції
Множення **S1** на **S2** та присвоєння результату операції **D** за умови, що потік енергії є дійсним.
- Примітка
Результатом операції інструкції MUL є 32-бітні дані.
- Екземпляр застосунку


```
LD X0
MUL D0 D1 D10
```

Множення D0 (1000) на D1 (2000) та присвоєння результату операції (D10,D11), коли X0=ON. У цьому

випадку, (D10,D11)=2000000.

6.3.4 DIV: Інструкція цілочисельного ділення instruction

LAD:	Застосовна модель	VC1S VC1 VC2 VC3 VC5

									Зумовлений прапор біт							
IL: DIV (S1) (S2) (D)									Довжина кроку				7			
Операнд	Тип	Застосовний програмний елемент														Індекс ація
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓

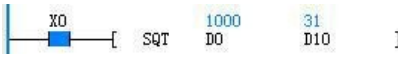
- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
- Опис функції
Ділення **S1** на **S2** та присвоєння результату операції **D** (D містить дві одиниці, одна зберігає значення частки, а інша зберігає значення залишку), коли потік енергії є дійсним.
- Примітка
Якщо **S2** ≠ 0, система повідомляє про помилку, вказуючи на те, що 0 не може бути використаний як дільник, і не виконує операцію ділення.
- Екземпляр застосунку


LD X0
DIV D0 D1 D10

Ділення D0 (2500) на D1 (1000) та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку D10=2, D11=500.

6.3.5 SQT: Інструкція для добування квадратного кореня з цілого числа

LAD: [SQT (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та позики прапор						
IL: SQT (S) (D)									Довжина кроку			5						
Операнд		Тип	Застосовний програмний елемент														Індекс ація	
S		INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	
D		INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
1. Видобування квадратного кореня з **S** та присвоєння результату операції **D** за умови, що потік енергії є дійсним.
2. Встановлення біта прапора нуля (SM80) Коли (**D**) дорівнює 0. Встановлення прапора переносу біт (SM82), коли результат операції округлено в меншу сторону.
- Примітка
Якщо **S** ≥ 0, система повідомляє про помилку операнда і не виконує операцію добування квадратного кореня.
- Екземпляр застосунку


LD X0
SQT D0 D10

Присвоєння результату квадратного кореня з D0 (1000) до D10, коли X0=ON. У цьому випадку D10=31.

6.3.6 INC: Ціле число плюс один інструкція

LAD:			Застосовна модель	VC1S VC1 VC2 VC3 VC5				
			Зумовлений прапор біт					
IL: INC (D)			Довжина кроку	3				
Операнд	Тип	Застосовний програмний елемент					Індексація	

D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√

- Операнд опис
D: Цільовий операнд
- Опис функції
Автоматичне збільшення **D** на 1, коли потік енергії є дійсним.
- Примітка
Ця інструкція є інструкцією циклічного додавання, діапазон якої становить від -32768 до 32767. Підтримуваний діапазон елемента **C** становить від C0 до C199.
- Екземпляр застосунку

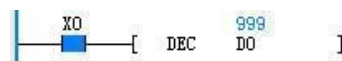
LD X0
INC D0

Автоматичне збільшення D0 (1000) на 1, коли X0=ON. Після виконання, D0=1001.

6.3.7 DEC: Ціле число мінус один інструкція

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5						
										Зумовлений прапор біт								
IL: DEC (D)										Довжина кроку		3						
Операнд	Тип	Застосовний програмний елемент															Індекс ація	
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√		

- Операнд опис
D: Цільовий операнд
- Опис функції
Зменшення **D** за допомогою 1
автоматичноКоли потік енергії є дійсним.
- Примітка
Ця інструкція є циклічною мінус інструкцією, і діапазон становить від -32768 до 32767.
- Екземпляр застосунку



LD X0
DEC D0

Автоматичне зменшення D0 (1000) на 1, коли X0=ON. Після виконання, D0=999.

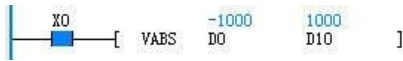
6.3.8 VABS: Інструкція для отримання абсолютного значення цілого числа

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5							
									Зумовлений прапор біт											
IL: VABS (S) (D)									Довжина кроку				5							
Операнд		Тип	Застосовний програмний елемент														Індексація			
S		INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√			
D		INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√			

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
Взяття абсолютного значення **S** та присвоєння отриманого результату **D** коли потік енергії є дійсним.
- Примітка

Діапазон **S** має бути від -32767 до 32767. Коли значення **S** становить -32768, система повідомляє про помилку, вказуючи, що значення операнда інструкції є недійсним, і інструкція не виконує жодної дії.

- Екземпляр застосунку



Обчислення абсолютного значення D0 (-1000), присвоєння отриманого результату D10, коли X0=ON. У цьому випадку D10=1000.

6.3.9 NEG: Інструкція заперечення цілого числа instruction

LAD: [NEG (S) (D)]									Застосовна модель		VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт								
IL: NEG (S) (D)									Довжина кроку		5						
Операнд	Тип	Застосовний програмний елемент														Індекс ація	
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
Діапазон **S** має бути від -32767 до 32767. Коли значення **S** становить -32768, система повідомляє про помилку, вказуючи, що значення операнда інструкції є недійсним, і інструкція не виконує жодної дії.
- Опис функції
Взяття від'ємного значення **S** та присвоєння отриманого результату **D** за умови, що потік енергії є дійсним.
- Примітка
- Екземпляр застосунку

LD X0
NEG D0 D10
Взяття від'ємного значення D0 (1000) та присвоєння отриманого результату D10, коли X0=ON. У цьому випадку D10=-1000.

6.3.10 DADD: Додавання довгих цілих чисел інструкція

LAD: 									

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
менший за -2147483648.

6.3.11 DSUB: Віднімання довгих цілих чисел інструкція

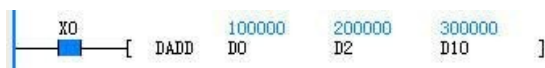
- Опис функції
1. Збільшення **S1** на **S2** та присвоєння результату операції **D** за умови, що потік енергії є дійсним.
2. Встановлення біта прапора перенесення (SM81), коли результат операції (**D**) більший за 2147483647. Встановлення біта прапора нуля (SM80), коли результат операції дорівнює 0. Встановлення біта прапора позики (SM82), коли результат операції

● Екземпляр застосунку

```
LD X0
```

```
DADD D0 D2 D10
```

Збільшення значення (100000) (D0,D1) на значення (200000) (D2,D3) та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку, (D10,D11)=300000.



LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Зумовлений прапор біт	Прапорець нуля, прапорець перенесення та прапорець позики
IL: DSUB (S1) (S2) (D)	Довжина кроку	10

Операнд	Тип	Застосовний програмний елемент														Індексація
S1	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
S2	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓

- Опис операнда **S1**:

Операнд джерела 1

S2: Операнд

джерела 2

D: Цільовий операнд

- Опис функції

1. Віднімання **S1** від **S2** та присвоєння результату операції **D** за умови, що потік енергії є дійсним.

2. Встановлення біта прапора перенесення (SM81), коли результат операції (**D**) більший за 2147483647. Встановлення біта прапора нуля (SM80), коли результат операції дорівнює 0. Встановлення біта прапора позики (SM82), коли результат операції менший за -2147483648.

- Екземпляр застосунку



LD X0

DSUB D0 D2 D10

Віднімання значення (100000) (D0,D1) від значення (200000) (D2,D3) та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку (D10,D11)= - 100000.

6.3.12 DMUL: Інструкція множення довгих цілих чисел instruction

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5					
									Зумовлений прапор біт								
IL: DMUL (S1) (S2) (D)									Довжина кроку			10					
Операнд	Тип	Застосовний програмний елемент														Індекс ація	
S1	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
S2	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓	

- Опис операнда **S1**:

Операнд джерела 1

S2: Операнд

джерела 2

D: Цільовий операнд

- Опис функції

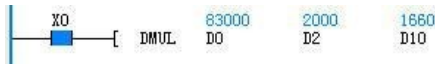
Множення **S1** на **S2** та

присвоєння результату операції **D** за умови, що потік енергії є дійсним.

- Примітка

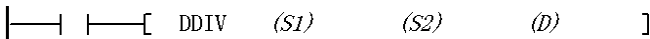
Будь ласка, зверніть увагу, що результатом операції інструкції DMUL є 32-бітні дані, що може призвести до переповнення.

- Екземпляр застосунку



Множення значення (83000) (D0,D1) на значення (2000) (D2,D3) та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку, (D10,D11)=1660000000.

6.3.13 DDIV: Ділення довгих цілих чисел інструкція

LAD: 		Застосовна модель	VC1S VC1 VC2 VC3 VC5
		Зумовлений прапор біт	
IL: DDIV (S1) (S2) (D)		Довжина кроку	10
Операнд	Тип	Застосовний програмний елемент	Індексація

S1	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√
S2	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√

- Опис операнда **S1**:

Операнд джерела 1

S2: Операнд

джерела 2

D: Цільовий операнд

- Опис функції

Ділення **S1** на **S2** та присвоєння результату операції **D** (**D** включає чотири одиниці, перші дві зберігають значення частки, а останні дві — значення залишку), коли потік енергії є дійсним.

- Примітка

Якщо $S2 \neq 0$, система повідомляє про помилку, вказуючи на те, що 0 не може бути використаний як дільник, і не виконує операцію ділення.

- Екземпляр застосунку



Ділення значення (83000) (D0,D1) на значення (2000) (D2,D3) із присвоєнням результату операції (D10,D11) та (D12,D13), коли X0=ON. У цьому випадку (D10,D11)=41 та (D12,D13)=1000.

6.3.14 DSQT: Інструкція для добування квадратного кореня з довгого цілого числа

LAD: [DSQT (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5					
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та позики прапор					
IL: DSQT (S) (D)									Довжина кроку			7					
Операнд	Тип	Застосовний програмний елемент														Індексація	
S	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓	

- Операнд опис

S: Джерельний операнд

D: Цільовий операнд

- Опис функції

1. Видобування квадратного кореня з **S** та присвоєння результату операції **D** за умови, що потік енергії є дійсним.

2. Встановлення біта прапорця нуля (SM80), коли (**D**) дорівнює 0. Встановлення біта прапорця переносу (SM82), коли результат операції округлюється.

- Примітка

Якщо $S \geq 0$, система повідомляє про помилку операнда і не виконує операцію добування квадратного кореня.

- Екземпляр застосунку

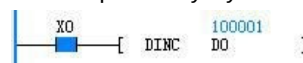


Видобування квадратного кореня зі значення (83000) регістрів (D0,D1) та присвоєння результату квадратного кореня регістрам (D10,D11), коли X0=ON. У цьому випадку (D10,D11)=288.

6.3.15 DINC: Довге ціле число плюс одна інструкція

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт									
IL: DINC (D)									Довжина кроку			4						
Операнд		Тип		Застосовний програмний елемент													Індекс ація	
D		DINT				KnY	KnM	KnS	KnLM		D		C		V		R	√

- Операнд опис
D: Цільовий операнд
- Опис функції
Автоматичне збільшення **D** на 1, коли потік енергії є дійсним.
- Примітка
1. Ця інструкція є інструкцією циклічного додавання, а діапазон становить від -2147483648 до 2147483647.
2. Підтримка лише 32-бітного C елемента серед C елементів.
- Екземпляр застосунку



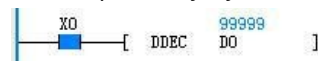
LD X0
DINC D0

Автоматичне збільшення значення (100000) (D0,D1) при X0=ON. Після виконання, (D0,D1)=100001.

6.3.16 DDEC: Довге ціле число мінус один інструкція

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5							
									Зумовлений прапор біт											
IL: DDEC (D)									Довжина кроку				4							
Операнд		Тип		Застосовний програмний елемент															Індекс ація	
D		DINT				KnY	KnM	KnS	KnLM		D		C		V		R	√		

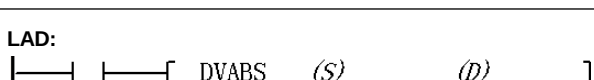
- Операнд опис
D: Цільовий операнд
- Опис функції
Автоматичне зменшення **D** на 1, коли потік енергії є дійсним.
- Примітка
Ця інструкція є циклічною інструкцією віднімання, і діапазон становить від -2147483648 до 2147483647.
- Екземпляр застосунку



LD X0
DDEC D0

Автоматичне зменшення значення (100000) (D0,D1), коли X0=ON. Після виконання (D0,D1)=99999.

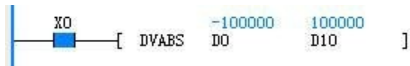
6.3.17 DVABS: Інструкція для отримання абсолютного значення цілого числа типу long

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт				Прапорець нуля, прапорець перенесення та позики прапор						
IL: DVABS (S) (D)									Довжина кроку				7						
Операнд		Тип		Застосовний програмний елемент														Індкса ція	
S		DINT		Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√	
D		DINT				KnY	KnM	KnS	KnLM		D		C		V		R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
Взяття абсолютного значення S та присвоєння отриманого результату **D** коли потік енергії є дійсним.

- Примітка
Діапазон **S** має бути від -2147483647 до 2147483647. Коли значення **S** становить -2147483648, система видає помилку, вказуючи на те, що значення операнда інструкції є недійсним, і інструкція не виконує жодної дії.

- Екземпляр застосунку



Обчислення абсолютного значення (- 100000) для (D0,D1) та присвоєння отриманого результату до (D10,D11), коли X0=ON. У цьому випадку (D10,D11)=100000.

6.3.18 DNEG: Заперечення довгого цілого числа інструкція

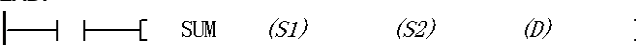
LAD: [DNEG (S) (D)]									Застосовна модель				VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт				Прапорець нуля, прапорець перенесення та позики прапор						
IL: DNEG (S) (D)									Довжина кроку				7						
Операнд		Тип	Застосовний програмний елемент													Індексація			
S		DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√		
D		DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√		

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
Взяття від'ємного значення **S** та присвоєння отриманого результату **D** за умови, що потік енергії є дійсним.
- Примітка
Діапазон **S** має бути від -2147483647 до 2147483647 . Коли значення **S** дорівнює -2147483648 , система видає помилку, що вказує на недійсне значення операнда інструкції, і інструкція не виконує жодної дії.
- Екземпляр застосунку


LD X0
DNEG D0 D10

Взяття від'ємного значення (100000) від (D0,D1) та присвоєння отриманого результату до (D10,D11), коли X0=ON. У цьому випадку, (D10,D11) = -100000 .

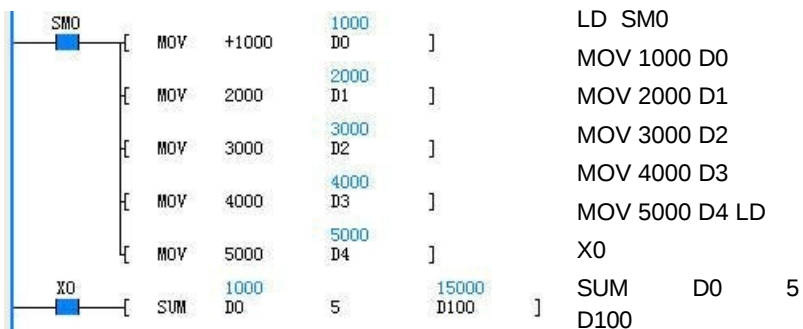
6.3.19 SUM: Інструкція накопичення цілих чисел

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та позики прапор						
IL: SUM (S1) (S2) (D)									Довжина кроку			8						
Операнд	Тип	Застосовний програмний елемент														Індексація		
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√		

- Операнд опис
S1: Джерело операнда, початковий блок для накопичення
S2: Джерело операнда, кількість одиниць даних для накопичення
D: Призначення операнд, результат накопичення
- Опис функції
Накопичення вмісту одиниць **S2**, починаючи з початкової одиниці (**S1**), та присвоєння отриманого результату **D**, коли потік енергії є дійсним.

- Примітка
 1. Результат операції інструкції SUM є 32-бітними даними.
 2. Якщо $0 \leq S2 \leq 255$, система повідомляє про помилку операнда.
 3. Оскільки D є 32-бітними даними, прапорці перенесення та позики завжди дорівнюють 0, тоді як біт нульового прапорця визначається кінцевим результатом накопичення.

- Application instance



Накопичення даних 5 одиниць, починаючи з D0, та присвоєння отриманого результату до (D100,D101), коли X0=ON. Увімк. це випадок, (D100,D101)=D0+.....+D4=15000.

6.3.20 DSUM: Інструкція накопичення довгого цілого числа instruction

LAD:								Застосовна модель VC1S VC1 VC2 VC3 VC5								
┌ ─																

- Операнд опис
S1: Джерело операнда, початковий блок для накопичення

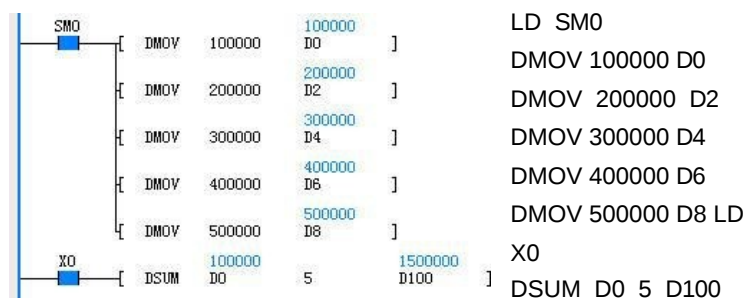
S2: Джерело операнда, кількість одиниць даних для накопичення
D: Призначення

операнд, результат накопичення

- Опис функції
Накопичення вмісту **S2**×2 одиниць, починаючи з початкової одиниці (**S1**), виконання інструкції DSUM та присвоєння отриманого результату **D**, коли потік енергії є дійсним.
- Примітка

Якщо $0 \leq S2 \leq 255$, система повідомляє про помилку </В операнда.

- Екземпляр застосунку



Накопичення даних 5×2 одиниць, починаючи з D0, та присвоєння отриманого результату (D100,D101), коли X0=ON.

У цьому випадку, (D100,D101)= (D0,D1) + ... + (D8,D9)=1500000.

6.4 Інструкції операцій з плаваючою комою

6.4.1 RADD: Інструкція додавання чисел з рухомою комою інструкція

LAD: ┌──┐ ┌──┐ [RADD (S1) (S2) (D)]										Застосовно модель		VC1S VC1 VC2 VC3 VC5					
										Під впливом прапорця біта		Прапорець нуля, прапорець переносу, та позичити прапор					
IL: RADD (S1) (S2) (D)										Крок довжина		10					
Операн Д		Тип		Застосовний програмний елемент											Індкса ція		
S1		REAL		Склад ається						D				V		R	√

		немає														
S2	REAL	Склад ається немає							D				V		R	√
D	REAL								D				V		R	√

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд
джерела 2
D: Цільовий операнд

- Опис функції
1. Збільшення **S1** на **S2** та
присвоєння результату операції **D**
за умови, що потік енергії є
дійсним.



- 2. Встановлення біта прапора переносу (SM81), коли результат операції (**D**) більший за 1.701412e + 038 або менший за -1.701412e + 038. Встановлення біта нульового прапора (SM80), коли результат операції дорівнює 0.

- Екземпляр застосунку

LD X0
RADD D0 D2 D10

Збільшення значення (-10000.2)
(D0,D1) на значення (2000.5) від

(D2,D3), та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку (D10,D11)=-7999.7.

6.4.2 RSUB: Інструкція віднімання чисел з рухомою комою instruction

LAD: 										Застосовно модель		VC1S VC1 VC2 VC3 VC5					
										Під впливом прапорець біта		Прапорець нуля, прапорець переносу, та позичити прапор					
IL: RSUB (S1) (S2) (D)										Довжина кроку		10					
Операнд	Тип	Застосовний програмний елемент														Показчик	
S1	REAL	Складається							D				V		R	✓	
S2	REAL	Складається							D				V		R	✓	
D	REAL	немає							D				V		R	✓	

- Опис операнда **S1**:

Операнд джерела 1

S2: Операнд

джерела 2

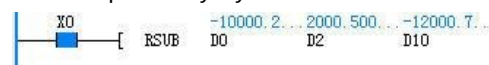
D: Цільовий операнд

- Опис функції

1. Віднімання **S1** від **S2** та присвоєння результату операції **D** за умови, що потік енергії є дійсним.

2. Встановлення біта прапора перенесення (SM81), коли результат операції (**D**) більший за $1.701412e + 038$ або менший за $-1.701412e + 038$. Встановлення біта нульового прапора (SM80), коли результат операції дорівнює 0.

- Екземпляр застосунку



LD X0

RSUB D0 D2 D10

Віднімання значення (- 10000.2) (D0,D1) від значення (2000.5) (D2,D3) та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку (D10,D11)=- 12000.7.

6.4.3 RMUL: Інструкція множення чисел з рухомою комою інструкція

LAD: 									Застосовна модель		VC1S VC1 VC2 VC3 VC5					
									Зумовлений прапор біт		Прапорець нуля, прапорець перенесення та прапорець позики прапор					
IL: RMUL (S1) (S2) (D)									Довжина кроку		10					
Операнд	Тип	Застосовний програмний елемент													Індексація	
S1	REAL	Складається немає							D				V		R	✓
S2	REAL	Складається немає							D				V		R	✓
D	REAL								D				V		R	✓

- Опис операнда **S1**:

Операнд джерела 1

S2: Операнд

джерела 2

D: Цільовий операнд

- Опис функції

1. Множення **S1** на **S2** та присвоєння результату операції **D** за умови, що потік енергії є дійсним.
2. Встановлення біта прапора переносу (SM81), коли результат операції (D) більший за $1.701412e + 038$ або менший за $-1.701412e + 038$. Встановлення біта нульового прапора (SM80), коли результат операції дорівнює 0.

● Екземпляр застосунку

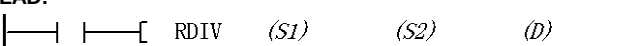


LD X0

RMUL D0 D2 D10

Множення значення (-10000.2) (D0,D1) на значення (2000.5) (D2,D3) та присвоєння результату операції (D10,D11), коли $X0=ON$. У цьому випадку (D10,D11)=-20005400.0 (фактично результат становить -20005400.1, але він округлюється відповідно до точності обчислень).

6.4.4 RDIV: Інструкція ділення чисел з рухомою комою інструкція

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5				
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор				
IL: RDIV (S1) (S2) (D)									Довжина кроку			10				
Операнд	Тип	Застосовний програмний елемент												Індексація		
S1	REAL	Склад ається немає							D				V		R	✓
S2	REAL	Склад ається немає							D				V		R	✓
D	REAL								D				V		R	✓

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
- Опис функції
 - Ділення **S1** на **S2** та присвоєння результату операції **D**.
 - Встановлення біта прапора переносу (SM81), коли результат операції (D) більший за 1.701412e + 038 або менший за -1.701412e + 038. Встановлення біта нульового прапора (SM80), коли результат операції дорівнює 0.
- Примітка
Якщо $S2 \neq 0$, система повідомляє про помилку, вказуючи, що 0 не може бути використаний як дільник, і не виконує операцію ділення.
- Екземпляр застосунку


Ділення значення (-10000.2) регістрів (D0,D1) на значення (2000.5) регістрів (D2,D3) із записом результату операції в (D10,D11), коли $X0=ON$. У цьому випадку, (D10,D11)=-4.998850.

6.4.5 RSQT: Інструкція для добування квадратного кореня з числа з рухомою комою числа

LAD: [RSQT (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5				
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор				
IL: RSQT (S) (D)									Довжина кроку			7				
Операнд	Тип	Застосовний програмний елемент												Індексація		
S	REAL	Склад ається немає							D				V		R	√
D	REAL								D				V		R	√

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
 - Добування квадратного кореня з **S** та присвоєння результату операції **D**, коли потік енергії є дійсним.
 - Встановлення біта прапорця нуля (SM80), коли (D) дорівнює 0.
- Примітка

Якщо $S \geq 0$, система повідомляє про помилку операнда та не виконує добування квадратного кореня.

- Екземпляр застосунку



Видобування квадратного кореня зі значення (10000.2) регістрів (D0,D1) та присвоєння результату квадратного кореня регістрам (D10,D11), коли $X0=ON$. У цьому випадку, $(D10,D11)=100.000999$.

6.4.6 RVABS: Інструкція для отримання абсолютного значення числа з рухомою комою числа

LAD: [RVABS (S) (D)]										Застосовна модель			VC1S VC1 VC2 VC3 VC5				
										Зумовлений прапор біт							
IL: RVABS (S) (D)										Довжина кроку			7				
Операнд		Тип	Застосовний програмний елемент												Індексація		
S		REAL	Складається немає							D				V		R	✓
D		REAL								D				V		R	✓

- Операнд опис

S: Джерельний операнд

D: Цільовий операнд

- Опис функції

Обчислення абсолютного значення **S** та присвоєння отриманого результату **D**, коли потік енергії є дійсним.

- Екземпляр застосунку



LD X0

RVABS D0 D10

Обчислення абсолютного значення (-10000.2) для (D0,D1) та присвоєння отриманого результату до (D10,D11), коли X0=ON. У цьому випадку (D10,D11)=10000.2.

6.4.7 RNEG: Інструкція заперечення числа з рухомою комою instruction

LAD:									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
[RNEG (S) (D)]									Зумовлений прапор біт									
IL: RNEG (S) (D)									Довжина кроку			7						
Операнд		Тип		Застосовний програмний елемент											Індексація			
S		REAL		Складається	немає						D				V		R	✓
D		REAL									D				V		R	✓

- Операнд опис

S: Джерельний операнд

D: Цільовий операнд

- Опис функції

Взяття від'ємного значення від **S** та присвоєння отриманого результату **D**, коли потік енергії є дійсним.

- Екземпляр застосунку



LD X0

RNEG D0 D10

Взяття від'ємного значення (10000.2) з (D0,D1) та присвоєння отриманого результату до (D10,D11), коли X0=ON. У цьому випадку, (D10,D11)=-10000.2.

6.4.8 SIN: Інструкція для отримання SIN числа з плаваючою комою number

LAD: 			Застосовна модель	VC1S VC1 VC2 VC3 VC5				
			Прапорець впливу біт	Прапорець нуля, прапорець перенесення та прапорець позики				
IL: SIN (S) (D)			Довжина кроку	7				
Операнд	Тип	Застосовний програмний елемент						Індексація

																ія
S	REAL	Склад ається немає							D				V		R	√
D	REAL								D				V		R	√

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Опис функції

1. Отримання значення SIN **S** (одиниця: радіан) та присвоєння отриманого результату **D** коли потік енергії є дійсним.

2. Встановлення біта прапорця нуля (SM80), коли результат операції (**D**) дорівнює 0.

- Екземпляр застосунку

LD X0
SIN D0 D10

Беручи значення SIN (1.57) для (D0,D1) та присвоюючи отриманий результат до (D10,D11), коли X0=ON. У цьому випадку, (D10,D11) =1.

6.4.9 COS: Інструкція для отримання COS числа з рухомою комою number

LAD: COS (S) (D)]									Застосовна модель		VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики					
IL: COS (S) (D)									Довжина кроку		7					
Операнд	Тип	Застосовний програмний елемент													Індексація	
S	REAL	Складається немає							D				V		R	√
D	REAL								D				V		R	√

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

2. Встановлення біта нульового прапорця (SM80), коли результат операції (**D**) дорівнює 0.

- Опис функції
1. Отримання значення COS для **S** (одиниця виміру: радіан), та присвоєння отриманого результату **D** коли потік енергії є дійсним.

- Екземпляр застосунку

LD X0
COS D0 D10

Беручи значення COS (3.14) з (D0,D1) та присвоюючи отриманий результат до (D10,D11), коли X0=ON. У цьому випадку (D10,D11) =-0.9999999.

6.4.10 TAN: Інструкція для отримання TAN числа з рухомою комою number

LAD: TAN (S) (D)									Застосовна модель			VC1S VC1 VC2 VC3 VC5				
									Прапорець впливу біт			Прапорець нуля, прапорець перенесення та прапорець позики				
IL: TAN (S) (D)									Довжина кроку			7				
Операнд	Тип	Застосовний програмний елемент												Індекс		
S	REAL	Складається немає							D				V		R	√
D	REAL								D				V		R	√

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

2. Встановлення біта прапора переносу (SM80), коли результат операції (**D**) більший за 1.701412e + 038 або менший за -1.701412e + 038. Встановлення біта нульового прапора (SM81), коли результат операції дорівнює 0.

- Опис функції
1. Отримання значення TAN для **S** (одиниця: радіан) та присвоєння отриманого результату **D**, коли потік енергії є дійсним.

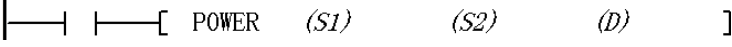
- Екземпляр застосунку

LD X0
TAN D0 D10

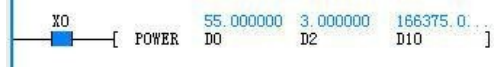
LD X0
TAN D0 D10

Беручи значення TAN (1.57) з
(D0,D1) та присвоюючи
отриманий результат (D10,D11),
коли X0=ON. У цьому випадку
(D10,D11) =1255.848398.

6.4.11 POWER: Інструкція для піднесення до степеня числа з плаваючою комою числа

LAD: 										Застосовно модель				VC1S VC1 VC2 VC3 VC5			
										Зумовлений прапор біт				Прапорець нуля, прапорець переносу, та позичити прапор			
IL: POWER (S1) (S2) (D)										Довжина кроку				10			
Операн д	Тип	Застосовний програмний елемент														Індексац я	
S1	REAL	Склад ається немає							D				V		R	√	
S2	REAL	Склад ається немає							D				V		R	√	
D	REAL								D				V		R	√	

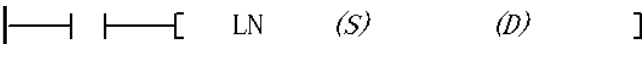
- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
- Опис функції
 - Отримання **S2** степеня **S1**, а потім присвоєння результату операції **D**, коли потік енергії є дійсним.
 - Встановлення біта прапора перенесення (SM81), коли результат операції (**D**) більший за $1.701412e + 038$ або менший за $-1.701412e + 038$. Встановлення біта нульового прапора (SM81), коли результат операції дорівнює 0.
- Примітка
 - Якщо **S1**=0 та **S2**≤0, система повідомляє про помилку значення операнда і не виконує операцію.
 - Якщо **S1**<0 і мантиса **S2** не дорівнює 0, система повідомляє про помилку значення операнда і не виконує операцію.
- Екземпляр застосунку



LD X0
POWER D0 D2 D10

Отримання степеня (D2,D3) для (D0,D1), а саме (ступінь 3.0 для 55.0), та присвоєння результату (D10,D11), коли X0=ON. У цьому випадку, (D10,D11) =166375.0.

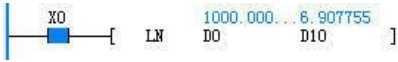
6.4.12 LN: Інструкція для отримання натурального логарифма числа з рухомою комою числа

LAD:									Застосовна модель				VC1S VC1 VC2 VC3 VC5				
									Прапорець впливу біт				Прапорець нуля, прапорець перенесення та прапорець позики				
IL: LN (S) (D)									Довжина кроку				7				
Операнд	Тип	Застосовний програмний елемент													Індексація		
S	REAL	Складається немає							D				V		R	√	
D	REAL								D				V		R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
 - Отримання значення LN для **S** та присвоєння результату операції **D**, коли потік енергії є дійсним.
 - Встановлення біта прапорця переносу (SM81), коли результат операції (**D**) є

більше ніж $1.701412e+038$ або
менше ніж $-1.701412e+038$.
Встановлення біта прапорця
нуля (SM80), коли результат
операції дорівнює 0.

- Екземпляр застосунку



Приймаючи значення LN
(1000.0) для (D0,D1) та
присвоюючи отриманий
результат (D10,D11), коли
X0=ON. У цьому випадку
(D10,D11) =6.907755.

6.4.13 EXP: Інструкція для отримання степеня натурального числа з числа з рухомою комою number

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Зумовлений прапор біт		Прапорець нуля, прапорець перенесення та прапорець позики прапор					
IL: EXP (S) (D)										Довжина кроку		7					
Операнд		Тип	Застосовний програмний елемент													Індексація	
S		REAL	Склад ається немає						D				V		R	√	
D		REAL							D				V		R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
 більше ніж $1.701412e+038$ або менше ніж $-1.701412e+038$.
 Встановлення біта нульового прапорця (SM80), коли результат операції дорівнює 0.
- Опис функції
 1. Отримання значення EXP для **S** та присвоєння результату операції **D**, коли потік енергії є дійсним.
 2. Встановлення біта прапора переносу (SM81), коли результат операції (**D**) є
- Екземпляр застосунку

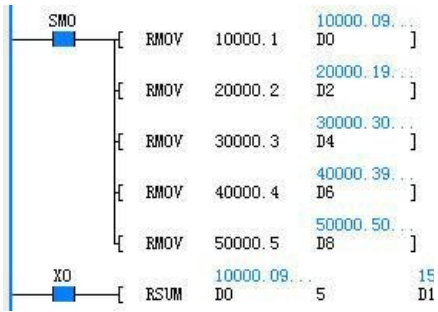
 Беручи значення EXP (10.0) з (D0,D1) та присвоюючи отриманий результат до (D10,D11), коли X0=ON. У цьому випадку (D10,D11) =22026.464844.

6.4.14 RSUM: Інструкція накопичення чисел з рухомою комою instruction

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Зумовлений прапор біт		Прапорець нуля, прапорець перенесення та прапорець позики прапор					
IL: RSUM (S1) (S2) (D)										Довжина кроку		9					
Операнд	Тип	Застосовний програмний елемент													Індексація		
S1	REAL								D				V		R	✓	
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D				V		R	✓	
D	REAL								D				V		R	✓	

- Операнд опис
S1: Джерело операнда, початковий блок для накопичення
S2: Джерело операнда, номер про дані, що підлягають накопиченню
D: Операнд призначення, накопичення результату
 дійсним.
- Примітка
 1. Якщо $0 \leq S2 \leq 255$, система повідомляє про помилку операнда.
 2. Система не виконує операцію накопичення після виникнення переповнення.
- Опис функції
 Накопичення вмісту **S2** × 2 одиниць, починаючи з початкової одиниці (**S1**), виконання інструкції RSUM та присвоєння отриманого результату **D**, коли потік енергії є

● Екземпляр застосунку



Накопичення даних 5×2
одиниць, починаючи з D0, та
присвоєння отриманого
результату (D100,D101), коли
X0=ON.

У цьому випадку, (D100,D101)= (D0,D1)
+ + (D8,D9)=150001.5.

6.4.15 ASIN: Інструкція для отримання ASIN числа з плаваючою комою number

LAD: [ASIN (S) (D)]								Застосовна модель			VC1S VC1 VC2 VC3 VC5					
								Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор					
IL: ASIN (S) (D)								Довжина кроку			7					
Операнд		Тип	Застосовний програмний елемент										Індексація			
S		REAL	Складається немає						D				V	R	√	
D		REAL							D				V	R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
 - Отримання SIN^{-1} значення **S** та присвоєння результату операції **D** коли потік енергії є дійсним.
 - Встановлення біта прапорця нуля (SM80), коли результат операції (**D**) дорівнює 0.
- Примітка
 Якщо $S > 1$ або $S < -1$, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.
- Екземпляр застосунку



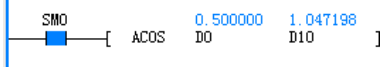
LD SM0
 ASIN D0 D10

Беручи значення SIN^{-1} (0.500000) для (D0,D1) та присвоюючи отриманий результат (D10,D11), коли SM0=ON. У цьому випадку (D10,D11) = 0.523599.

6.4.16 ACOS: Інструкція для отримання ACOS числа з рухомою комою числа

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5				
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та позики прапор				
IL: ACOS (S) (D)									Довжина кроку			7				
Операнд		Тип	Застосовний програмний елемент											Індексація		
S		REAL	Складається немає							D				V	R	√
D		REAL								D				V	R	√

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
 - Отримання COS^{-1} значення **S** та присвоєння результату операції **D** коли потік енергії є дійсним.
 - Встановлення біта нульового прапорця (SM80), коли результат операції (**D**) дорівнює 0.
- Примітка
 Якщо $S > 1$ або $S < -1$, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.
- Екземпляр застосунку



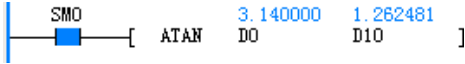
LD SM0
 ACOS D0 D10

Беручи значення COS^{-1} (0.500000) для (D0,D1) та присвоюючи отриманий результат (D10,D11), коли SM0=ON. У цьому випадку (D10,D11) = 1.047198.

6.4.17 ATAN: Інструкція для отримання ATAN числа з рухомою комою числа

LAD:										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
 ATAN (S) (D)]										Зумовлений прапор біт		Прапорець нуля, прапорець перенесення та прапорець позики прапор					
IL: ATAN (S) (D)										Довжина кроку		7					
Операнд		Тип		Застосовний програмний елемент										Індексація			
S		REAL		Складається немає							D				V	R	√

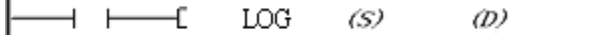
D	REAL								D				V	R	√

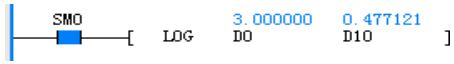
- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
1. Отримання значення TAN^{-1} для **S** та присвоєння результату операції **D**, коли потік енергії є дійсним.
- 2. Встановлення біта прапорця нуля (SM80), коли результат операції (**D**) дорівнює 0.
- Екземпляр застосунку


```
LD SM0
ATAN D0 D10
```

Беручи значення TAN^{-1} (3.14) для (D0,D1) та присвоюючи отриманий результат (D10,D11), коли SM0=ON. У цьому випадку (D10,D11) =1.262481.

6.4.18 LOG: Інструкція для отримання десяткового логарифма числа з рухомою комою числа

LAD: 										Застосовна модель		VC2 VC3 VC5				
										Зумовлений прапор біт		Прапорець нуля, прапорець перенесення та прапорець позики прапор				
IL: LOG (S) (D)										Довжина кроку		7				
Операнд	Тип	Застосовний програмний елемент												Індексація		
S	REAL	Складається							D				V	R	√	
D	REAL	немає							D				V	R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
1. Отримання значення LOG для **S** та присвоєння результату операції до **D**, коли потік енергії є дійсним. LOG — це загальна логарифмічна операція за основою 10.
- 2. Встановлення прапорця перенесення (переповнення) (SM81), коли результат операції (**D**) переповнюється. Встановлення прапорця нуля (SM80), коли результат операції дорівнює 0.
- Екземпляр застосунку


```
LD SM0
LOG D0 D10
```

Прийняття значення (3.0) D0(D1) та присвоєння отриманого результату D10(D11), коли SM0=ON. У цьому випадку D10(D11) =0.477121.

6.4.19 RAD: Інструкція для перетворення кута з градусів у радіани conversion

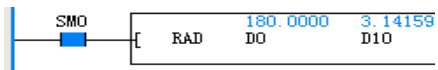
LAD: 										Застосовна модель		VC2 VC3 VC5				
										Зумовлений прапор біт		Прапорець нуля, прапорець перенесення та прапорець позики прапор				
IL: RAD (S) (D)										Довжина кроку		7				
Операнд	Тип	Застосовний програмний елемент													Індексація	
S	REAL	Складається							D				V	R	√	
D	REAL	немає							D				V	R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
1. Перетворення значення кута **S** одиничного числа з рухомою комою у значення радіана та присвоєння

результат операції до **D**, коли потік енергії є дійсним.

2. Встановлення біта прапорця нуля (SM80), коли результат операції дорівнює 0.

● Екземпляр застосунку



Взяття значення (180.0) D0(D1) та присвоєння отриманого результату D10(D11), коли SM0=ON. У цьому випадку D10(D11) = 3.141593.

6.4.20 DEG: Інструкція для перетворення кута в радіанах з плаваючою комою conversion

LAD: 	Застосовна модель	VC2 VC3 VC5
	Зумовлений прапор біт	Прапорець нуля, прапорець перенесення та прапорець позики прапор

IL: DEG (S) (D)									Довжина кроку		7				
Операнд	Тип	Застосовний програмний елемент												Індексація	
S	REAL	Складається немає						D				V	R	√	
D	REAL							D				V	R	√	

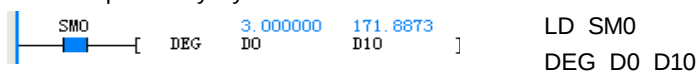
- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Опис функції
 1. Перетворення радіанного значення **S** одиничного числа з рухомою комою у значення кута та присвоєння

результат операції до **D**, коли потік енергії є дійсним.

2. Встановлення біта прапорця нуля (SM80), коли результат операції дорівнює 0. Встановлення біта прапорця переносу (переповнення) (SM81), коли результат операції (D) переповнюється.

- Екземпляр застосунку



Присвоєння значення (3.0) з D0(D1) до D10(D11), коли SM0=ON. У цьому випадку D10(D11) =171.8873.

6.5 Інструкції перетворення значень

6.5.1 DTI: Інструкція щодо перетворення довгого цілого числа на ціле число

LAD: [DTI(S)(D)]									Застосовна модель		VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики						
IL: DTI (S) (D)									Довжина кроку		6						
Операнд	Тип	Застосовний програмний елемент														Індексація	
S	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√	
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Опис функції
 Перетворення **S** з довгого цілого числа на ціле число та присвоєння результату операції **D** за умови, що потік енергії є дійсним.

Якщо $S > 32767$ або $S < -32768$, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.

- Екземпляр застосунку



Перетворення (D0, D1) =10000 з довгого цілого числа на ціле число та присвоєння результату операції в D10, коли X0=ON. У цьому випадку D10=10000.

- Примітка

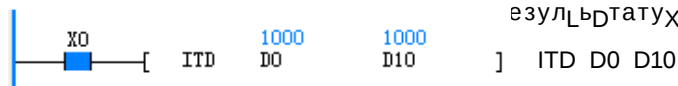
6.5.2 ITD: Інструкція для перетворення цілого числа на довге ціле число

LAD:			ITD (S) (D)	Застосовна модель	VC1S VC1 VC2 VC3 VC5				
				Зумовлений прапор біт	Прапорець нуля, прапорець перенесення та прапорець позики прапор				
IL: ITD (S) (D)			Довжина кроку	6					
Операнд	Тип	Застосовний програмний елемент							Індексація

S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
Перетворення **S** з цілого числа на довге ціле число та присвоєння результату операції **D** за умови, що потік енергії є дійсним.

● Екземпляр застосунку



Перетворення D0=1000 з цілого числа у довге ціле число та присвоєння результату виконання D10, коли X0=ON. У цьому випадку (D10,D11)=1000.

6.5.3 FLT: Інструкція для перетворення цілого числа у число з плаваючою комою

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
IL: FLT (S) (D)									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор						
									Довжина кроку			6						
Операнд	Тип	Застосовний програмний елемент														Індексація		
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓		
D	REAL								D				V		R	✓		

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Опис функції
Перетворення S з цілого числа у число з рухомою комою та присвоєння результату операції D за умови, що потік енергії є дійсним.

● Екземпляр застосунку



Перетворення D0=10005 з цілого числа у число з рухомою комою та присвоєння результату операції до (D10,D11), коли X0=ON. У цьому випадку, (D10,D11)=10005.0.

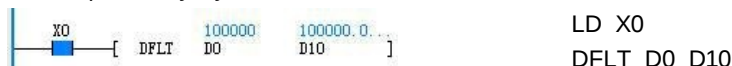
6.5.4 DFLT: Інструкція для перетворення довгого цілого числа у число з плаваючою комою

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор						
IL: DFLT (S) (D)									Довжина кроку			7						
Операнд		Тип	Застосовний програмний елемент													Індексація		
S	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓		
D	REAL								D				V		R	✓		

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Опис функції
Перетворення S з цілого числа великого формату у число з рухомою комою та присвоєння результату операції D, коли потік енергії є дійсним.

● Екземпляр застосунку

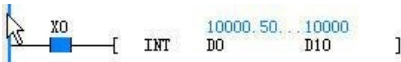


Перетворення (D0,D1)=100000 з довгого цілого числа у число з рухомою комою та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку, (D10,D11)=100000.0.

6.5.5 INT: Інструкція для перетворення числа з рухомою комою на ціле число

LAD: 		Застосовна модель	VC1S VC1 VC2 VC3 VC5
		Прапорець впливу біт	Прапорець нуля, прапорець перенесення та прапорець позики
IL: INT (S) (D)		Довжина кроку	6
Операнд	Тип	Застосовний програмний елемент	
Д			Індексація

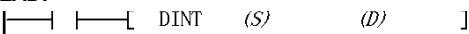
S	REAL	Склад ається немає							D				V		R	√
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
1. Перетворення **S** з числа з рухомою комою на ціле число та присвоєння результату операції **D**, коли потік енергії є дійсним.
2. Ця інструкція впливає на прапорець нуля та прапорець позики. Встановлення біта прапорця нуля(SM80), коли результат перетворення дорівнює 0. Встановлення прапорця позики, коли десяткові дробу результату є округлено. Встановлення біта прапорця перенесення (SM81), коли результат перевищує діапазон даних довгого цілого числа.
- Примітка
Коли **S** > 32767, **D**=32767. Коли **S** < -32768, **D**=-32768, і встановить біт прапорця перенесення (переповнення) (SM81).
- Екземпляр застосунку


```
LD X0
INT D0 D10
```

Перетворення (D0,D1)=10000.5from числа з рухомою комою на ціле число та присвоєння результату операції D10, коли X0=ON. У цьому випадку, (D0,D1)
=10000.5.

6.5.6 DINT: Інструкція для перетворення числа з рухомою комою у довге ціле число

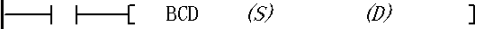
LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт			Прапорець нуля, прапорець перенесення та прапорець позики					
IL: DINT (S) (D)									Довжина кроку			7					
Операнд	Тип	Застосовний програмний елемент														Індексація	
S	REAL	Складається немає							D				V		R	√	
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
1. Перетворення **S** з числа з рухомою комою на довге ціле число та присвоєння результату операції **D** за умови, що потік енергії є дійсним.
2. Встановлення біта нульового прапорця (SM80), коли результат перетворення дорівнює 0. Встановлення прапорця позики, коли десяткові дробу результату округлюються. Встановлення прапорця перенесення (переповнення) (SM81), коли результат перевищує діапазон даних довгого цілого числа.
- Примітка
Коли **S** > 2147483647, **D**=2147483647. Коли **S** < -2147483648, **D**=-2147483648, та встановить прапорець перенесення (переповнення) (SM81) біти одночасно.
- Екземпляр застосунку


```
LD X0
DINT D0 D10
```

Перетворення (D0,D1)=100000.5from числа з плаваючою комою у довге ціле число та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку (D10,D11)=100000.

6.5.7 BCD: Інструкція з перетворення слова у 16-бітний BCD код

LAD: 		Застосовна модель		VC1S VC1 VC2 VC3 VC5												
		Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики												
IL: BCD (S) (D)		Довжина кроку		5												
Операнд	Тип	Застосовний програмний елемент														Індекс ація
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓

- Операнд опис
S: вихідний операнд, ≤ 9999
D: Цільовий операнд
 - Опис функції
Перетворення **S** з цілого числа у 16-бітний код BCD та присвоєння отриманого результату **D**, коли потік енергії є дійсним.
 - Примітка
- Якщо **S** > 9999, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.
- Екземпляр застосунку



LD X0
BCD D0 D10

Перетворення D0=0x0D05 (3333) з цілого числа у 16-бітний BCD код та присвоєння результату операції до D10, коли X0=ON. У цьому випадку D10=0x3333 (13107).

6.5.8 DBCD: Інструкція для перетворення подвійного слова у 32-бітний BCD code

LAD: [DBCD (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт			Прапорець нуля, прапорець перенесення та прапорець позики					
IL: DBCD (S) (D)									Довжина кроку			7					
Операнд	Тип	Застосовний програмний елемент														Індексація	
S	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓	

- Операнд опис
S: вихідний операнд, ≤ 99999999
D: Цільовий операнд
 - Опис функції
Перетворення **S** з довгого цілого числа у 32-бітний код BCD та присвоєння отриманого результату **D**, коли потік енергії є дійсним.
 - Примітка
- Якщо **S** > 99999999, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.
- Екземпляр застосунку



LD X0
DBCD D0 D10

Перетворення (D0,D1)=0x3F940AA (66666666) з довгого цілого числа у 32-бітний код BCD та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку (D10,D11)=0x66666666 (1717986918).

6.5.9 BIN: Інструкція для перетворення 16-бітного BCD коду у слово

LAD: [BIN (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт			Прапорець нуля, прапорець перенесення та прапорець позики						
IL: BIN (S) (D)									Довжина кроку			5						
Операнд	Тип	Застосовний програмний елемент														Індексація		
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓		
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓		

- Операнд опис
S: Вихідний операнд, його формат даних повинен відповідати формату коду BCD **D**: Операнд призначення.
- Опис функції
Перетворення **S** з 16-бітного коду BCD на ціле число та присвоєння цього

отриманий результат до **D**, коли потік енергії є дійсним.

- Примітка

Якщо формат даних **S** не відповідає формату коду BCD, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.

- Application instance



Перетворення D0=0x5555 (21845) з 16-бітного BCD коду в ціле число, та

присвоєння D0 значення D10, коли X0=ON. У цьому випадку, D10=0x15B3 (5555).

6.5.10 DBIN: Інструкція для перетворення 32-бітного BCD коду на подвійне слово

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт				Прапорець нуля, прапорець перенесення та прапорець позики прапор						
IL: DBIN (S) (D)									Довжина кроку				7						
Операнд	Тип	Застосовний програмний елемент														Індексація			
S	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√			
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√			

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

Якщо формат даних S не відповідає формату коду BCD, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст D.

- Опис функції
1. Перетворення S з 32-бітного коду BCD у довге ціле число та присвоєння отриманого результату D за умови, що потік енергії є дійсним.
2. Формат даних S має відповідати формату коду BCD.

- Екземпляр застосунку



Перетворення (D0,D1)=0x99999999 (2576980377) з 32-бітного BCD коду у довге ціле число та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку, (D10,D11)=0x5F5E0FF (99999999).

- Примітка

6.5.11 GRY: Інструкція щодо перетворення слова у 16-бітний сірий код

LAD: [GRY (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт			Прапорець нуля, прапорець перенесення та прапорець позики					
IL: GRY (S) (D)									Довжина кроку			5					
Операнд	Тип	Застосовний програмний елемент														Індексація	
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Екземпляр застосунку



Перетворення D0=0xAAAA (43690) з цілого числа у 16-бітний код Грея та присвоєння результату операції D10, коли X0=ON. У цьому випадку D10=0xFFFF (65535).

- Опис функції
Перетворення S з цілого числа у 16-бітний код Грея та присвоєння отриманого результату D, коли потік енергії є дійсним.

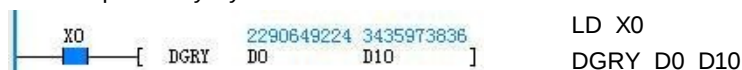
6.5.12 DGRY: Інструкція для перетворення подвійного слова у 32-бітний код Грея code

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Прапорець впливу біт	Прапорець нуля, прапорець перенесення та прапорець позики

IL: DGRY (S) (D)									Довжина кроку			7				
Операнд	Тип	Застосовний програмний елемент														Індексація
S	DINT	Складається	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
D	DINT	немає		KnY	KnM	KnS	KnLM		D		C		V		R	✓

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Екземпляр застосунку



- Опис функції
Перетворення **S** з довгого цілого числа на 32-бітний код Грея та присвоєння отриманого результату **D**, коли потік енергії є дійсним.

Перетворення (D0,D1)=0x88888888 (2290649224) з довгого цілого числа у 32-бітний код Грея та присвоєння результату операції (D10,D11), коли X0=ON. У цьому випадку (D10,D11)=0xCCCCCCCC (3435973836).

6.5.13 GBIN: Інструкція з перетворення 16-бітового коду Грея у слово

LAD: [GBIN (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор						
IL: GBIN (S) (D)									Довжина кроку			5						
Операнд	Тип	Застосовний програмний елемент														Індексація		
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓		
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓		

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд

- Екземпляр застосунку



- Опис функції
Перетворення **S** з 16-бітового коду Грея в ціле число та присвоєння отриманого результату **D**, коли потік енергії є дійсним.

Перетворення D0=0xFFFF (65535) з 16-бітового коду Грея у ціле число та присвоєння результату операції D10, коли X0=ON. У цьому випадку D10=0xAAAA (43690).

6.5.14 DGBIN: Інструкція з перетворення 32-бітного коду Грея у подвійне слово

LAD: DGBIN (S) (D)									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт			Прапорець нуля, прапорець перенесення та прапорець позики						
IL: DGBIN (S) (D)									Довжина кроку			7						
Операнд	Тип	Застосовний програмний елемент														Індексація		
S	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√		
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√		

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції

Перетворення S з 32-бітного коду Грея у довге ціле число та присвоєння отриманого результату **D** коли потік енергії є дійсним.

- Application instance



LD X0
DGBIN D0 D10

Перетворення (D0,D1)=0хCCCCCCCC
(3435973836) з 32-бітового коду Грея у довге
ціле число та призначення

результат операції у (D10,D11), коли X0=ON. У цьому випадку
(D10,D11)=0х88888888 (2290649224).

6.5.15 SEG: Інструкція з перетворення слова на 7-сегментний код

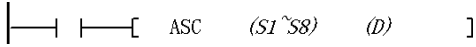
LAD: [SEG (S) (D)]									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор						
IL: SEG (S) (D)									Довжина кроку			5						
Операнд	Тип	Застосовний програмний елемент														Індексація		
S	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√		

- Операнд опис
S: Вихідний операнд,
S≤15 **D**: Цільовий операнд
- Опис функції
Перетворення **S** з цілого числа на 7-сегментний код та присвоєння отриманого результату **D**, коли потік енергії є дійсним.
- Примітка
Якщо **S > 15**, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.
- Екземпляр застосунку


LD X0
SEG D0 D10

Перетворення D0=0x0F (15) з цілого числа у 7-сегментний код та присвоєння результату операції до D10, коли X0=ON. У цьому випадку D10=0x71 (113).

6.5.16 ASC: ASCII перетворення коду інструкція

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5							
									Прапорець впливу біт				Прапорець нуля, прапорець перенесення та прапорець позики							
IL: ASC (S1–S8) (D)									Довжина кроку				19							
Операнд		Тип	Застосовний програмний елемент															Індексація		
S1		INT	Складається немає																	
S2		INT	Складається немає																	
S3		INT	Складається немає																	
S4		INT	Складається немає																	
S5		INT	Складається немає																	
S6		INT	Складається немає																	
S7		INT	Складається немає																	

S8	INT	Складається немає														
D	INT								D		C	T	V	Z	R	√

- Операнд опис
S1-S8: Операнд джерела (якщо кількість символів менша за 8, решта символів заповнюється 0)
Підтримка лише тих символів, код ASCII яких становить від 0x21 до 0x7E (введення з клавіатури, якщо кількість яких символів менше ніж 8, решта символів заповнюються 0)
D: Цільовий операнд
- Опис функції

Перетворення рядка символів **S1-S8** у код ASCII та присвоєння отриманого результату елементам, починаючи з **D**, коли потік енергії є дійсним. Збереження двох даних коду ASCII у старшому/молодшому байті кожного елемента **D**, коли SM85=OFF. Збереження одного даних коду ASCII у молодшому байті кожного елемента **D**, коли SM85=ON.

- Екземпляр застосунку



LD M0

ASC 12345678 D0

Виконання ASCII перетворення, коли M0=ON, а дані зберігаються у двох режимах:

- Якщо SM85=OFF, результат виконання: D0=0x3231, D1=0x3433, D2=0x3635, D3=0x3837.
- Якщо SM85=ON, результат виконання: D0=0x31, D1=0x32, D2=0x33, D3=0x34, D4=0x35, D5=0x36, D6=0x37, D7=0x38.

6.5.17 ITA: Інструкція для перетворення 16-бітових шістнадцяткових даних у ASCII код

LAD:									Застосовна модель									VC1S VC1 VC2 VC3 VC5				
									Прапорець впливу біт					Прапорець нуля, прапорець перенесення та прапорець позики								
— [ITA (S1) (D) (S2)]																						
ITA (S1) (D) (S2)									Довжина кроку													
Операнд		Тип	Застосовний програмний елемент														Індикація					
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLN	KnSM	D	SD	C	T	V	Z	R	✓						
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓						
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLN	KnSM	D	SD	C	T	V	Z	R	✓						

- Операнд опис

S1: Вихідні шістнадцяткові дані для перетворення

D: Цільовий операнд

S2: Кількість кодів ASCII ($1 \leq S2 \leq 256$)

- Опис функції

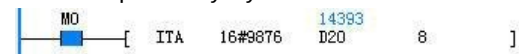
Перетворення шістнадцяткових даних, починаючи з елемента **S1**, у коди S2 ASCII та присвоєння отриманого результату елементам, починаючи з **D**, коли потік енергії є дійсним. Збереження даних двох кодів ASCII у старшому/молодшому байті кожного елемента **D**, коли SM85=OFF. Збереження даних одного коду ASCII у молодшому байті кожного елемента **D**, коли SM85=ON.

- Примітка

- Якщо **S1** та **D** використовують адресацію Kn, Kn=4.
- Якщо **S2** не знаходиться в діапазоні від 1 до 256, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.

3. Якщо **S1** є константою, коли **S2** ≥ 4, системним значенням за замовчуванням є **S2=4**, і система не повідомляє про помилку операнда після застосування значення за замовчуванням.

- Екземпляр застосунку



Вихідні дані: 0x9876.

LD M0

ITA 16#9876 D20 8

Виконання ITA операції перетворення, коли M0=ON and дані зберігаються у двох режимах:

- Якщо SM85=OFF, результат виконання: D20=0x3839, D21=0x3637.
- Якщо SM85=ON, результат виконання: D20=0x39, D21=0x38, D22=0x37, D23=0x36.

6.5.18 ATІ: Інструкція для перетворення коду ASCII у 16-бітні шістнадцяткові дані

LAD:					Застосовна модель		VC1S	VC1	VC2	VC3	VC5
	[ATІ (S1) (D) (S2)]				Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики				
АТІ (S1) (D) (S2)				Довжина кроку							
Операнд			Застосовний програмний елемент								Індексація
	Тип										

S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√

- Операнд опис

S1: Вихідні ASCII дані коду для перетворення $0x30 \leq S1 \leq 0x39$ або $0x41 \leq S1 \leq 0x46$ (коли SM85=OFF, необхідно, щоб старший/молодший байти **S1** були в цьому діапазоні) **D:** Операнд призначення
S2: Кількість кодів ASCII ($1 \leq S2 \leq 256$)

- Опис функції

Коли потік енергії є дійсним, перетворить дані коду ASCII першого **S2** елемента **S1** у шістнадцяткові дані та збережить результат по 4 біти в початковий елемент **D**. Коли SM85=OFF, старший/молодший байт кожного елемента **D** зберігає два дані коду ASCII, коли SM85=ON, молодший байт кожного елемента **D** зберігає одні дані коду ASCII.

Перетворення кодів S2 ASCII, починаючи з елемента **S1**, у шістнадцяткові дані та збереження отриманого результату кожні 4 біти в елементах, починаючи з **D**, коли потік енергії є дійсним. Збереження даних двох кодів ASCII у старшому/молодшому байті кожного елемента **D**, коли SM85=OFF. Збереження даних одного коду ASCII у молодшому байті кожного елемента **D**, коли SM85=ON.

- Примітка

1. Якщо **S1** та **D** використовують адресацію Kn, Kn=4.

2. Якщо **S1** не знаходиться в діапазоні від 0x30 до 0x39 або від 0x41 до 0x46, або **S2** не знаходиться в діапазоні від 1 до 256, система повідомляє про помилку операнда, не виконує операцію перетворення та зберігає вміст **D**.

3. Якщо **S1** є константою, коли SM85=OFF та $S2 \geq 2$, системним значенням за замовчуванням є $S2=2$. Коли SM85=ON та $S2 \geq 1$, системним значенням за замовчуванням є $S2=1$. Система не повідомляє про помилку операнда після застосування значення за замовчуванням.

- Екземпляр застосунку



LD M0

ATI D10 D30 4

D10=0x3938, D11=0x3736, D12=0x3534, D13=0x3332.

Виконання операції перетворення ІТА, коли M0=ON and, результати, згенеровані відповідно до режиму зберігання даних, наведені нижче:

- Якщо SM85=OFF, результат виконання: D30=0x8967.
- Якщо SM85=ON, результат виконання: D30=0x8642.

6.5.19 LCNV: Інструкція з перетворення проєкту instruction

LAD: 									

- Операнд опис

S1: Початкова адреса операнда джерела, який потрібно перетворити
S2: Початкова адреса таблиці перетворення table
D: Початкова адреса, де зберігається результат перетворення результату
S3: Кількість даних для перетворення ($1 \leq S3 \leq 64$)

- Опис функції

Коли температурні або аналогові модулі використовуються для вимірювання температури або аналогових величин, якщо значення температури або значення, зчитане проєктом, виміряне за допомогою PLC, відхиляється від результату, виміряного стандартним термометром або відповідними стандартними приладами, цю інструкцію можна використовувати для виконання

лінійної модифікації з метою калібрування фактичного вимірювання.

При використанні модуля аналогового входу для зчитування сигналу

зовнішній аналоговий сигнал, ви можете використовувати цю інструкцію для

перетворити початкове аналогове зчитане значення на те

відповідне значення проєкту value.

Заповнення таблиці перетворення чотирма параметрами, а саме

вимірювання нижньої точки V_{ML} , висока точка значення

значення V_{MH} та відповідна нижня точка вимірювання

стандартне значення V_{SL} та стандартне значення верхньої точки V_{SH} .

Коли виконується лінійне перетворення, ціль

стандартне значення генерується шляхом застосування наведеної нижче формули LDI M1

вихідні дані, у яких S_n — це початкові вхідні дані, а

MOV 282 D1000

 D_n is conversion result data.

MOV 3530 D1001

$$A = (V_{SL} - V_{SH}) / (V_{ML} - V_{MH}) * 10000$$

MOV 260 D1002

$$B = V_{SL} - (V_{ML} * A / 10000)$$

MOV 3650 D1003

$$D_n = (S_n * A / 10000) + B$$

LDI M4

- Примітка

Ці чотири дані в таблиці перетворення мають свої фактичні значення, наприклад, значення вимірювання нижньої точки має бути меншим за значення вимірювання верхньої точки. Результат є неточним, якщо результат перетворення перевищує діапазон цілих чисел. Якщо D_n більше за 32767, воно становить 32767; якщо воно менше за -32768, воно дорівнює -32768.

MOV 282 D100

MOV 3530 D101

MOV 1906 D102

MOV 0 D103

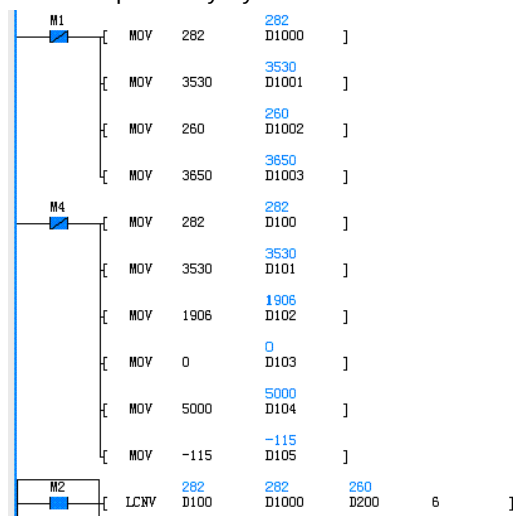
MOV 5000 D104

MOV -115 D105

LD M2

LGNV D100 D100 D1000 D200 6

- Екземпляр застосунку



Виконання інструкції LGNV, коли M2=ON. Результати, згенеровані відповідно до режиму зберігання даних, наведено нижче:

D200 = 260

D201 = 3650

D202 = 1955

D203 = -34

D204 = 5184

D205 = -154

6.5.20 RLCNV: Інструкція з конвертації проєкту з рухомою комою інструкція

LAD:										Застосовна модель	VC2 VC3 VC5			
										Прапорець впливу біт	Прапорець нуля, прапорець перенесення та прапорець позики			
IL: RLCNV (S1) (S2) (D) (S3)										Довжина кроку	12			
Операнд	Тип	Застосовний програмний елемент										Індексація		
S1	REAL									D			V	R
S2	REAL									D			V	R
D	REAL									D			V	R
S3	INT	Складається немає								D			V	R

- Операнд опис

S1: Початкова адреса операнда-джерела, який потрібно перетворити

S2: Початкова адреса таблиці перетворення table

D: Початкова адреса, де зберігається результат перетворення результату

S3: Кількість даних для перетворення ($1 \leq S3 \leq 64$)

- Опис функції

При використанні модуля аналогового входу для зчитування зовнішнього аналогового сигналу ви можете скористатися цією інструкцією для перетворення початкового значення аналогового зчитування у відповідне значення зчитування проекту.

Коли температурні або аналогові модулі використовуються для застосувань вимірювання температури або аналогових величин, якщо значення температури або проекту, зчитане PLC, відхиляється від результату, виміряного стандартом

термометром або відповідними стандартними приладами, цю інструкцію можна використовувати для виконання лінійної модифікації з метою калібрування фактичного вимірювання.

Заповнення таблиці перетворення чотирма параметрами, а саме

вимірювання нижньої точки значення

V_{ML} , верхня точка

значення вимірювання

V_{MH} та відповідна нижня точка

standard value V_{SL}

and high point standard value V_{SH} .

Коли виконується лінійне перетворення, цільове стандартне значення генерується шляхом застосування наведеної нижче формули вихідні дані, у яких S_n — це початкові вхідні дані, а

D_n — це результат перетворення даних.

$$A = (V_{SL} - V_{SH}) / (V_{ML} - V_{MH}) * 10000$$

$$B = V_{SL} - (V_{ML} * A / 10000)$$

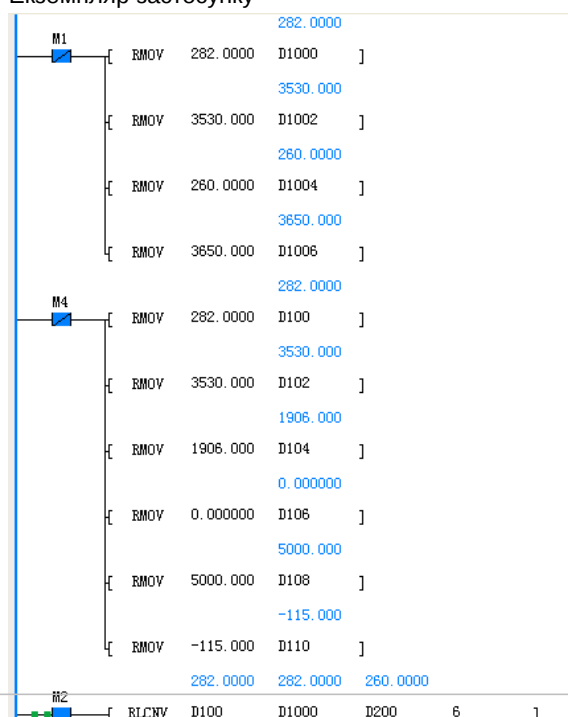
$$D_n = (S_n * A / 10000) + B$$

- При мітк а

Ці чотири дані в таблиці перетворення мають свої фактичні значення, наприклад, значення вимірювання нижньої точки має бути меншим за значення вимірювання верхньої точки. Результат є неточним, якщо результат перетворення перевищує діапазон цілих D_n більший за 32767, він чисел. Якщо становить 32767;

якщо воно менше за -32768, це -32768.

- Екземпляр застосунку



```
LDI M1
RMOV 282 D1000
RMOV 3530 D1002
RMOV 260 D1004
```

```
RMOV 3650 D1006
```

```
LDI M4
RMOV 282 D100
RMOV 3530 D102
RMOV 1906 D104
RMOV 0 D106
RMOV 5000 D108
RMOV -115 D110
LD M2
```

```
RLCNV D100 D1000 D200 6
```

Виконання інструкції RLCNV, коли M2=ON. Ця результати, згенеровані відповідно до режиму зберігання даних, наведено нижче:
D200(D201) = 260

D202(D203) = 3650

D204(D205) = 1955

D206(D207) = -34.3288

D208(D209) = 5184.267

6.6 Логічні операції над словами інструкції

6.6.1 WAND: Слово AND інструкція

D210(D211) = -154.357

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Прапорець впливу біт	

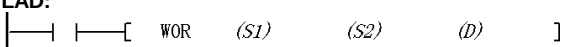
IL: WAND (S1) (S2) (D)										Довжина кроку	7					
Робота d	Тип	Застосовний програмний елемент														Показчик
S1	INT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
S2	INT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓

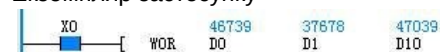
- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
- Опис функції Виконання операції AND побітово над **S1** та **S2**, і присвоєння отриманий результат до **D**, коли потік енергії є дійсним.
- Екземпляр застосунку


LD X0
WAND
D0 D1 D10

Виконання операції AND над D0=2#1011011010010011 (46739) та D1=2#1001001100101110 (37678) побітово, і присвоєння отриманого результату до D10, коли X0=ON. У цьому випадку, D10=2#1001001000000010 (37378).

6.6.2 WOR: INT OR інструкція

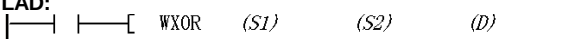
LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт								
IL: WOR (S1) (S2) (D)									Довжина кроку			7					
Робота d	Тип	Застосовний програмний елемент														Індексація	
S1	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
S2	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
- Опис функції Виконання побітової операції OR над **S1** та **S2**, і присвоєння отриманий результат до **D**, коли потік енергії є дійсним.
- Екземпляр застосунку


LD X0
WOR D0 D1 D10

Виконання операції OR побітово над D0=2#1011011010010011 (46739) та D1=2#1001001100101110 (37678), і присвоєння отриманого результату в D10, коли X0=ON. У цьому випадку, D10=2#1011011110111111 (47039).

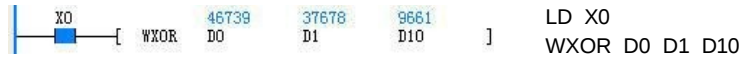
6.6.3 WXOR: Слово XOR інструкція

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт									
IL: WXOR (S1) (S2) (D)									Довжина кроку			7						
Робота d	Тип	Застосовний програмний елемент															Показник	
S1	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
S2	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√		

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд
джерела 2
D: Цільовий операнд
- Опис функції
Виконання побітової операції
XOR над **S1** та **S2**, і присвоєння
результату

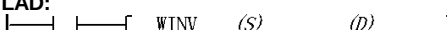
отриманий результат до **D**, коли потік енергії є дійсним.

- Екземпляр застосунку



Виконання операції XOR побітово над D0=2#1011011010010011 (46739) та D1=2#1001001100101110 (37678), і присвоєння отриманого результату до D10, коли X0=ON. У цьому випадку D10=2#0010010110111101 (9661).

6.6.4 WINV: Слово INV інструкція

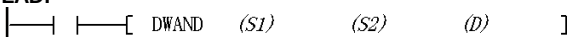
LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт									
IL: WINV (S) (D)									Довжина кроку			7						
Робота d		Тип	Застосовний програмний елемент													Індексація		
S		INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
D		INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	

- Операнд опис
S: Джерельний операнд
D: Цільовий операнд
- Опис функції
Виконання операції INV по бітах на **S**, та призначення отриманого результату до **D**, коли потік енергії є дійсним.
- Екземпляр застосунку

LD X0
WINV D0 D10

Виконання операції INV побітово над D0= (46739) та присвоєння отриманого результату у D10, коли X0=ON. У цьому випадку, D10=(18796).

6.6.5 DWAND: Подвійне слово AND інструкція

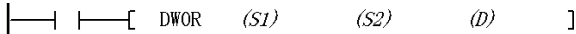
LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт									
IL: DWAND (S1) (S1) (D)									Довжина кроку			7						
Робота d	Тип	Застосовний програмний елемент														Індексація		
S1	DINT	Стант	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√		
S2	DINT	Стант	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√		
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	√		

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд
- Опис функції
Виконання операції AND по бітах на **S1** та **S2**, і присвоєння отриманого результату до **D** коли потік енергії є дійсним.
- Екземпляр застосунку

LD X0
DWAND D0 D2 D10

Виконання той AND побітова операція над (D0,D1)=2#10110010101001101110011001010010 (2997282386) та (D2, D3)=2#00111010001110110011000100110011 (976957747) з присвоєнням отриманий результат у (D10, D11), коли X0=ON. У цьому випадку (D10, D11)=2#00110010001000100010000000010010 (841097234).

6.6.6 DWOR: Подвійне слово OR інструкція

LAD:		Застосовна модель		VC1S VC1 VC2 VC3 VC5				
		Прапорець впливу біт						
IL: DWOR (S1) (S2) (D)		Довжина кроку		10				
Робота d	Тип	Застосовний програмний елемент						Показчик

S1	DINT	Стан	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
S2	DINT	Стан	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓

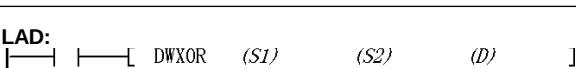
- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд

- Опис функції
Виконання операції OR побітово над **S1** та **S2** і присвоєння отриманого результату **D**, коли потік енергії є valid.

D3)=2#00111010001110110011000100110011

Виконання той OR експлуатація за допомогою біт ON D11)=2#101110101011111111011101110011 (2997282386) та (D2, (D0, D1)=2#10110010101001101110011001010010 (3133142899).

6.6.7 DWXOR: Подвійне слово XOR інструкція

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт									
IL: DWXOR (S1) (S2) (D)									Довжина кроку			10						
Робота d	Тип	Застосовний програмний елемент															Показник	
S1	DINT	Стан	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓		
S2	DINT	Стан	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓		
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓		

- Опис операнда **S1**:
Операнд джерела 1
S2: Операнд джерела 2
D: Цільовий операнд

- Екземпляр застосунку

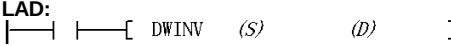


Виконання той XOR експлуатація за допомогою біт ON

(D0,D1)=2#10110010101001101110011001010010 (2997282386) та (D2,D3)=2#00111010001110110011000100110011 (976957747), і призначення отриманий результат у (D10, D11), коли X0=ON. У цьому випадку, (D10, D11)=2#10001000100111011101011101100001 (2292045665).

- Опис функції
Виконання операції XOR побітово над **S1** та **S2**, і присвоєння отриманого результату **D**, коли потік енергії є дійсним.

6.6.8 DWINV: Подвійне заперечення слова інструкція

LAD: 			Застосовна модель	VC1S VC1 VC2 VC3 VC5				
			Прапорець впливу біт					
IL: DWINV (S) (D)			Довжина кроку	7				
Робота d	Тип	Застосовний програмний елемент						Індексація

S	DINT	Стат	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓

- Операнд опис

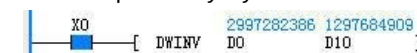
S: Джерельний операнд

D: Цільовий операнд

- Опис функції

Виконання операції INV побітово над **S** та присвоєння отриманого результату **D**, коли потік енергії є дійсним.

- Екземпляр застосунку



Виконання той INV експлуатація за допомогою біт увімк.

(D0,D1)=2#10110010101001101110011001010010 (2997282386), та призначення

той отримано результат до (D10,D11) коли X0=ON.
У це корпус,
(D10,D11)=2#01001101010110010001100110101101 (1297684909).

6.7 Інструкції з побітового зсуву та циклічного зсуву інструкції

6.7.1 ROR: 16-бітна циклічна операція зсуву вправо інструкція

LAD: 										Придатна модель	VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт	Прапорець переносу SM81					
IL: ROR (S1) (D) (S2)										Довжина кроку	7					
Робота	Тип	Застосовний програмний елемент														Індекс ація
S1	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓
S2	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Опис операнда **S1**:

Операнд джерела 1

D: Операнд призначення

S2: Джерело операнда 2

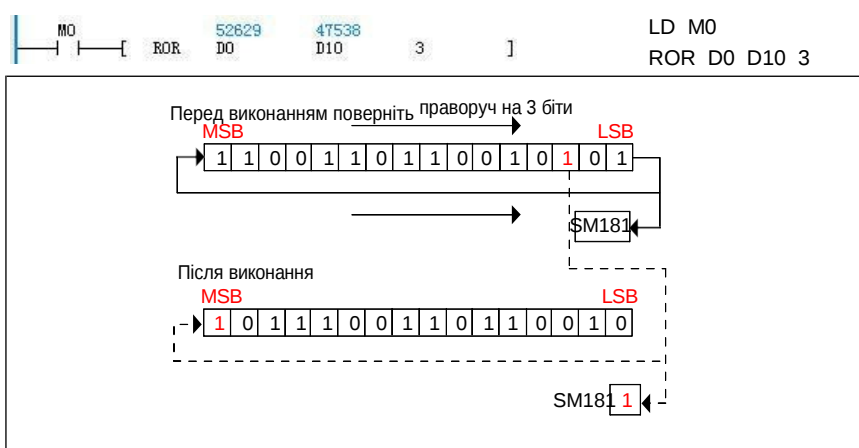
- Опис функції

Циклічне зміщення даних **S1** праворуч на **S2** бітів, присвоєння отриманого результату **D**, та збереження біта призначення у прапорці переносу (SM81), коли потік енергії дійсний.

- Примітка

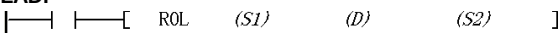
S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 4.

- Екземпляр застосунку



Циклічне зміщення D0=2#1100110110010101 (52629) праворуч на 3 біти, присвоєння отриманого результату D10 та збереження біта призначення у прапорці переносу, коли M0=ON. У цьому випадку D10=2#1011100110110010 (47538) та SM81=ON.

6.7.2 ROL: 16-бітна циклічна команда зсуву вліво інструкція

LAD: 	Застосовна модель	VC1S VC1 VC2 VC3 VC5
	Прапорець впливу біт	Прапорець переносу SM81

IL: ROL (S1) (D) (S2)									Довжина кроку			7				
Робота d	Тип	Застосовний програмний елемент														Показник
S1	INT	Конс-та а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓
S2	INT	Конс-та а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Опис операнда **S1**:
Операнд джерела 1
D: Операнд призначення
S2: Джерело операнда 2
- Опис функції
- Примітка
S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 4.

Циклічне зсування D0=2#1100110110010101 (52629)

- Екземпляр застосування



зсув ліворуч на 15 бітів, присвоєння отриманого результату D10 та збереження біта призначення у прапорці переносу, коли M0=ON. У цьому випадку D10=2#1110011011001010 (59082) та SM81=OFF.

6.7.3 RCR: Інструкція для 16-бітного циклічного зсуву праворуч через прапор переносу біт

LAD: [RCR (S1) (D) (S2)]										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт		Прапорець переносу SM81					
IL: RCR (S1) (D) (S2)										Довжина кроку		7					
Робота d	Тип	Застосовний програмний елемент														Індекс ація	
S1	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√	
S2	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	

- Опис операнда **S1**:
Операнд-джерело 1
D: Операнд-призначення
- Опис функції
Циклічне зміщення даних **S1** разом із прапорцем переносу (SM81) праворуч на **S2** бітів та присвоєння отриманого результату **D** за умови,

що потік енергії є дійсним.

- Примітка

S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 4.

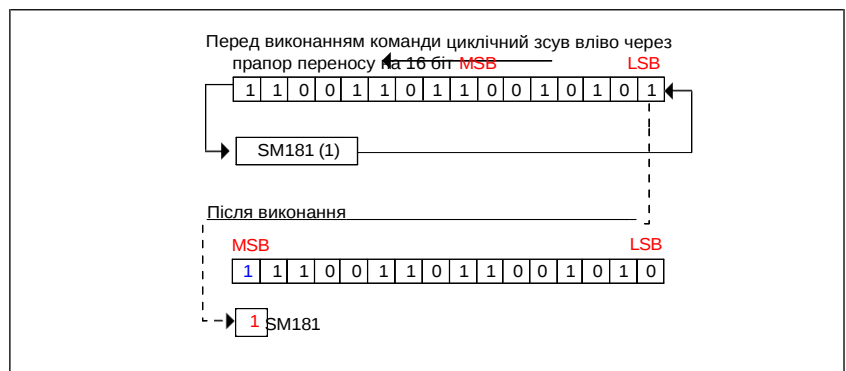
Екземпляр застосунку



6.7.4 RCL: Інструкція для 16-бітного циклічного зсуву вліво через прапорець переносу біт

LAD: [RCL (S1) (D) (S2)]									Застосовна модель				VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт				Прапорець переносу SM81					
IL: RCL (S1) (D) (S2)									Довжина кроку				7					
Робота d	Тип	Застосовний програмний елемент														Індекс ація		
S1	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	√		
S2	INT	Конс тант а	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		

- Опис операнда **S1**:
Операнд-джерело 1
D: Операнд-призначення
S2: Джерело операнда 2
- Опис функції
Циклічне зсування даних **S1** через прапорець переносу (SM81) вліво на **S2** бітів та присвоєння отриманого результату **D** за умови, що потік енергії є дійсним.
- Примітка
S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 4.
- Екземпляр застосунку



Циклічне зсування D0=2#1100110110010101 (52629) з прапорцем переносу (SM81=ON) праворуч на 16 бітів та присвоєння отриманого результату до D10, коли M0=ON. У цьому випадку D10=2#1110011011001010 (59082) та SM81=ON.

6.7.5 DROR: 32-бітна циклічна операція зсуву праворуч інструкція

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт		Прапорець переносу SM81					
IL: DROR (S1) (D) (S2)										Довжина кроку		9					
Робота d	Тип		Застосовний програмний елемент													Показчик ng	
S1	DINT		Постійний	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
D	DINT				KnY	KnM	KnS	KnLM		D		C		V		R	✓

S2	INT	П ості йни й	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Опис операнда **S1**:
Операнд-джерело 1
D: Операнд-призначення
S2: Джерело операнда 2

- Опис функції
Циклічне зсування даних **S1** праворуч на **S2** бітів, присвоєння отриманого результату **D**, та збереження біта призначення у прапорці переносу (SM81), коли потік енергії є дійсним.

- Примітка

S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 8.

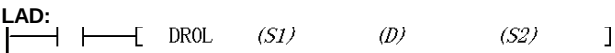
- Екземпляр застосунку



Циклічне зсування D0 (D1)=2#10110011100110001001110010101100 (3013123244) зсув праворуч на 7 біт, присвоюючи отриманий результат для (D10, D11), та зберігаючи біт призначення у прапорці переносу, коли M0=ON. У цьому випадку, (D10,D11)=2#01011001011001110011000100111001 (1499935033) та SM81=OFF.

Детальнішу інформацію див. на схемі ROR інструкції.

6.7.6 DROL: інструкція циклічного зсуву вліво на 32 біти інструкція

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт		Прапорець переносу SM81					
IL: DROL (S1) (D) (S2)										Довжина кроку		9					
Робота d	Тип	Застосовний програмний елемент														Показчик ng	
S1	DINT	Стат	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓	
S2	INT	Стат	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	

- Опис операнда **S1**:
Операнд-джерело 1
D: Операнд-призначення
S2: Джерело операнда 2

- Опис функції
Циклічне зсування даних **S1** ліворуч на **S2** бітів, присвоєння отриманого результату **D**, та збереження біта призначення у прапорці переносу (SM81), коли потік енергії є дійсним.

- Примітка

S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 8.

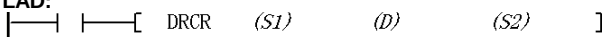
- Екземпляр застосунку



Циклічне зсування (D0,D1)=2#10110011100110001001110010101100 (3013123244) зсув праворуч на 30 біт, присвоєння отриманого результату до (D10, D11), та збереження біта призначення у прапорці переносу, коли M0=ON. У цьому випадку, (D10,D11)=2#00101100111001100010011100101011 (753280811) та SM81=ON.

Для отримання деталей зверніться до діаграми ROL інструкції.

6.7.7 DRCR: Інструкція для 32-бітного циклічного зсуву праворуч через прапор переносу біт

LAD: 		Застосовна модель	VC1S VC1 VC2 VC3 VC5				
		Прапорець впливу біт	Прапорець переносу SM81				
IL: DRCR (S1) (D) (S2)		Довжина кроку	9				
Робота d	Тип	Застосовний програмний елемент					Показчик ng

S1	DINT	Статус	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
D	DINT			KnY	KnM	KnS	KnLM		D		C		V		R	✓
S2	INT	Статус	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Опис операнда **S1**:
Операнд-джерело 1

D: Операнд-призначення

S2: Джерело операнда 2

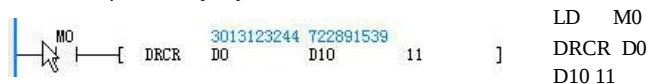
- Опис функції

Циклічне зміщення даних **S1** разом із прапорцем зсуву біта прапорця переносу (SM81=OFF) праворуч на 11 біт та переносу (SM81) праворуч на **S2** бітів із присвоєнням отриманого результату **D** за умови валідного потоку енергії.

- Примітка

S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 8.

- Екземпляр застосунку



- Циклічно зсув (D0,D1)=2#10110011100110001001110010101100 (3013123244) за допомогою

біта прапорця переносу (SM81=OFF) праворуч на 11 біт та присвоєнням отриманого результату (D10,D11), коли M0=ON. У цьому випадку (D10,D11)=2#00101011000101100111001100010011 (722891539) та SM81=ON.

- Для отримання деталей зверніться до діаграми RCR інструкції.

6.7.8 DRCL: Інструкція для 32-бітного циклічного зсуву вліво через прапорець переносу біт

LAD:		Застосовна модель VC1S VC1 VC2 VC3 VC5														
		Прапорець переносу SM81														
		Прапорець впливу біт														
DRCL		(S1)	(D)	(S2)	Довжина кроку 9											
Працює	Тип	Застосовний програмний елемент														Показники
S1	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
D	DINT		KnY	KnM	KnS	KnLM		D		C		V		R	✓	
S2	Постійна	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	

- Опис операнда **S1**:
Операнд-джерело 1

D: Операнд-призначення

S2: Джерело операнда 2

- Опис функції

Циклічне зсування даних **S1** через прапорець переносу (SM81) ліворуч на **S2** бітів та присвоєнням отриманого результату **D** за умови, що потік енергії є дійсним.

- Примітка

S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 8.

- Екземпляр застосунку



- Циклічне зсування (D0,D1)=2#10110011100110001001110010101100 (3013123244) з бітом прапорця переносу (SM81=ON) праворуч на 25 бітів, та присвоєнням отриманого результату до (D10,D11), коли M0=ON. У цьому випадку, (D10,D11)=2#00101100010110011100110001001100 (1488165020) та SM81=ON.
- Для отримання деталей зверніться до діаграми RCL інструкції.

6.7.9 SHR: 16-бітова інструкція зсуву вправо instruction

LAD:			Застосовна модель				
			VC1S	VC1	VC2	VC3	VC5
			Прапорець впливу біт				

IL: SHR (S1) (D) (S2)									Довжина кроку		7					
Робота d	Тип	Застосовний програмний елемент														Пока жчи к ng
S1	INT	Ста нт	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓
S2	INT	Ста нт	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

Перед виконанням поверніть праворуч на 5 біт

MSB

LSB

Після виконання

0	1	1	1	1	0	1	0	1	1	0	1	1	1	0	0
0	0	0	0	0	0	1	1	1	1	0	1	0	1	1	0

- Опис операнда **S1**:
Операнд джерела 1
D: Операнд призначення
S2: Джерело операнда 2
- Опис функції
Зсув даних **S1** праворуч на **S2** бітів та присвоєння отриманого результату **D** за умови валідного потоку енергії.
- Примітка
S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 4.
- Екземпляр застосунку

MO	SHR	31452	D0	982	D10	LD	MO	SHR	D0	D10	5

Shifting D0=2#0111101011011100 (31452) зсув праворуч на 5 бітів, і присвоєння отриманого результату до D10, коли M0=ON. У цьому випадку, D10=2#0000001111010110 (982).

6.7.10 SHL: 16-бітовий зсув вліво інструкція

LAD: [SHL (S1) (D) (S2)]									Застосовна модель		VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт							
IL: SHL (S1) (D) (S2)									Довжина кроку		7					
Робота d	Тип	Застосовний програмний елемент														Показчик ng
S1	INT	Стан t	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓
S2	INT	Стан t	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Опис операнда **S1**:
Операнд джерела 1
D: Операнд призначення
S2: Джерело операнда 2
- Опис функції
Зсув даних **S1** ліворуч на **S2** бітів та присвоєння отриманого результату **D**, коли потік енергії є дійсним.
- Примітка
S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 4.

● Екземпляр застосунку



Зсув D0=2#0111101011011100
(31452) ліворуч на 7 бітів та
присвоєння отриманого
результату до D10, коли
M0=ON. У цьому випадку
D10=2#0110111000000000
(28160).

6.7.11 DSHR: 32-бітна інструкція зсуву праворуч instruction

LAD: DSHR (S1) (D) (S2)									Застосовна модель			VC1S VC1 VC2 VC3 VC5											
									Прапорець впливу біт														
IL: DSHR									(S1)			(D)			(S2)			Довжина кроку			9		
Робота d		Тип		Застосовний програмний елемент														Показчик ng					
S1		DINT		Стан t	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓					
D		DINT				KnY	KnM	KnS	KnLM		D		C		V		R	✓					
S2		INT		Констант	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓					

- Опис операнда **S1**:
Операнд джерела 1
D: Операнд призначення
S2: Джерело операнда 2
 - Опис функції
Зсув даних **S1** праворуч на **S2** бітів та присвоєння отриманого результату **D** за умови, що потік енергії є valid.
 - Примітка
- S2** більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 8.
- Екземпляр застосунку

 LD M0
DSHR D0 D10 10

- Зсув (D0,D1)=2#01110011100110001001110010101100 (1939381420) зсув праворуч на 10 бітів, і присвоєння отриманого результату до (D10,D11), коли M0=ON. У цьому випадку, (D10,D11)=2#00000000000111001110011000100111 (1893927).
- Для отримання деталей зверніться до діаграми SHR інструкції.

6.7.12 DSHL: 32-бітна інструкція логічного зсуву вліво

LAD:	 DSHL (S1) (D) (S2)]									Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт							
DSHL			(S1)	(D)	(S2)	Довжина кроку				9							
	Робота d	Тип	Застосовний програмний елемент														Показчик ng
S1		Констант	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
D		DINT		KnY	KnM	KnS	KnLM		D		C		V		R	✓	
S2		Стан т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	

- Опис операнда **S1**:
Операнд-джерело 1
D: Операнд-призначення
S2: Джерело операнда 2
- Опис функції
Зсув даних **S1** ліворуч на **S2** бітів та присвоєння отриманого результату **D**, коли потік енергії є дійсним.
- Примітка

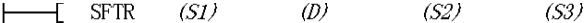
S2 більше або дорівнює 0. Коли **S1** використовує адресацію Kn, Kn має дорівнювати 8.

● Екземпляр застосунку



- Зсув
(D0,D1)=2#0111001110011000100111001
0101100 (1939381420)
ліворуч на 15 бітів, і присвоєння
отриманого результату до
(D10,D11), коли M0=ON. У цьому
випадку,
(D10,D11)=2#0100111001010110000
0000000000000 (1314258944).
- Детальнішу інформацію див. на схемі
SHR інструкції.

6.7.13 SFTR: Інструкція зсуву бітового рядка праворуч instruction

LAD:											Застосовна модель							VC1	VC1	VC2	VC3	VC5					
											Прапорець впливу біт																
	SFTR										(S1)	(D)	(S2)	(S3)	Довжина кроку							9					
	Робота d	Тип	Застосовний програмний елемент														Пока жчи к ng										
S1		BOOL		X	Y	M	S	LM	SM				C	T					✓								
D		BOOL			Y	M	S	LM					C	T					✓								
S2		INT	Стан т Kn	X	KnY	KnM	KnS	KnLM	KnSM		D	SD	C	T	V	Z	R		✓								
S3		INT	Стан т Kn	X	KnY	KnM	KnS	KnLM	KnSM		D	SD	C	T	V	Z	R		✓								

- Опис операнда **S1**:

Операнд-джерело 1

D: Операнд-призначення

S2: Джерело операнда 2

S3: Джерело операнда 3

- Опис функції

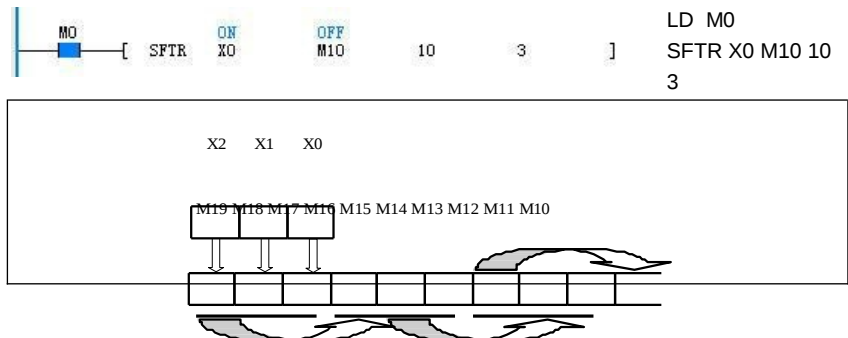
Зсув вмісту **S2** одиниць, починаючи з **D** праворуч на **S3** одиниць, відкидаючи **S3** дані на крайньому правому кінці, та зсув вмісту **S3** одиниць, починаючи з **S1** до лівого кінця рядка слів, коли потік енергії є дійсним.

- Примітка

Порядок зліва направо, елементи з великим номером елемента знаходяться зліва, а елементи з малим номером елемента знаходяться справа.

І **S2**, і **S3** є більшими або рівними нулю.

- Application instance



- Зсув вмісту 10 одиниць, починаючи з M10, на 3 одиниці праворуч, з відкиданням M10–M12 на крайньому правому кінці, та зсув вмісту 3 одиниць, починаючи з X0, до лівого кінця бітового рядка, коли M0=ON.
- Перед виконанням: X0=1, X1=0, X2=1. M10=0, M11=1, M12=1, M13=0, M14=0, M15=1, M16=0, M17=0, M18=0 та M19=1.
- Після виконання: вміст X0 до X2 зберігається. M10=0, M11=0, M12=1, M13=0, M14=0, M15=0, M16=1, M17=1, M18=0 та M19=1.

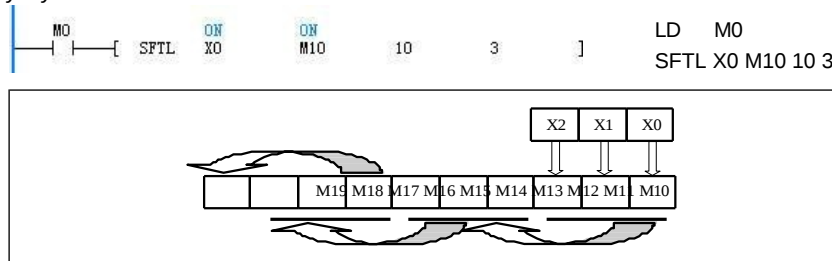
6.7.14 SFTL: Інструкція зсуву бітового рядка вліво instruction

LAD:	Застосовна модель										VC1S VC1 VC2 VC3 VC5					
	Прапорець впливу біт															
SFTL (S1) (D) (S2) (S3)										Довжина кроку 9						
S1	Робота d	Тип	Застосовний програмний елемент													Показчик
	BOOL		X	Y	M	S	LM	SM			C	T				✓
	BOOL			Y	M	S	LM				C	T				✓
S2	INT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	\$D	C	T	V	Z	R	✓
	INT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	\$D	C	T	V	Z	R	✓

- Опис операнда **S1**:
Операнд-джерело 1
D: Операнд-призначення
S2: Джерело операнда 2

S3: Джерело операнда 3

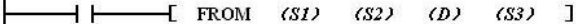
- Примітка
 1. Порядок зліва направо, елементи з великим номером елемента знаходяться ліворуч, а елементи з малим номером елемента знаходяться праворуч.
 2. І **S2**, і **S3** є більшими або рівними нулю.
- Опис функції
Зсув вмісту **S2** одиниць, починаючи з **D** праворуч на **S3** одиниць, відкидаючи **S3** дані на крайньому лівому кінці, та зсув вмісту **S3** одиниць, починаючи з **S1** до правого кінця рядка слів, коли потік енергії є дійсним.
- Екземпляр застосунку



1. Зсув вмісту 10 одиниць, починаючи з M10, ліворуч на 3 одиниці, з відкиданням M17–M19 на крайньому лівому кінці, та зсув вмісту 3 одиниць, починаючи з X0, до правого кінця бітового рядка, коли M0=ON.
2. Перед виконанням: X0=1, X1=0, X2=1. M10=0, M11=1, M12=1, M13=0, M14=0, M15=1, M16=0, M17=0, M18=0, та M19=1.
3. Після виконання: вміст X0 до X2 зберігається. M10=1, M11=0, M12=1, M13=0, M14=1, M15=1, M16=0, M17=0, M18=1 та M19=0.

6.8 Периферійні інструкції

6.8.1 FROM: Інструкція для зчитування слів із буфера спеціального модуля регістр

LAD: 										Застосовна модель		VC2 VC3 VC5					
										Прапорець впливу біт							
IL: FROM (S1) (S2) (D) (S3)										Довжина кроку		9					
Робота d	Тип	Застосовний програмний елемент														Показники	
S1	INT	Константа															
S2	INT	Константа															
D	INT							D				V		R	✓		
S3	INT	Константа															

- Операнд опис

S1: SN спеціальних модулів, що відвідуються. Діапазон налаштування: від 0 до 7. При зверненні до спеціального модуля, якого немає в системі, привод

система повідомляє, що адреса спеціального модуля є недійсною.

S2: Початкова адреса буфера BFM спеціального модуля, який потрібно зчитати

Діапазон налаштування: від 0 до 32767. При зверненні до недопустимої BFM адреси система повідомляє, що BFM одиниця оцінюваного спеціального модуля виходить за межі діапазону.

D: Елемент, у якому зберігаються дані, зчитані зі спеціального модуля.

S3: Кількість послідовних BFM буферів (одне слово), які потрібно зчитати

Діапазон даних: від 1 до 32767.

При спробі доступу до недійсної адреси буфера BFM система повідомляє, що буфер BFM оцінюваного спеціального модуля виходить за межі діапазону.

- Опис функції
Послідовне зчитування **S3** слів даних, починаючи з буфера **S2**

у визначеній BFM визначеного спеціального модуля (SN: **S1**), та послідовне збереження цих даних у **S3** елементах слів, починаючи з **D**.

- Примітка
Час виконання інструкції FROM є відносно тривалим і залежить від значення **S3**.

- Екземпляр застосунку



Послідовне зчитування 2 даних, починаючи з буфера 3 у BFM спеціального модуля № 0, та збереження цих даних у D100 та D101 відповідно, коли M0 є ON.

6.8.2 DFROM: Інструкція для зчитування подвійних слів із буфера спеціального модуля регістр

LAD: [DFROM (S1) (S2) (D) (S3)]										Застосовна модель		VC2 VC3 VC5					
										Прапорець впливу біт							
IL: DFROM (S1) (S2) (D) (S3)										Довжина кроку		10					
Робота d	Тип	Застосовний програмний елемент														Показник	
S1	INT	Константа															
S2	INT	Константа															
D	DINT							D				V		R	✓		
S3	INT	Константа															

Операнд опис

S1: SN спеціальних модулів відвідано

Діапазон налаштування: від 0 до 7. При зверненні до спеціального модуля, який відсутній у системі, система повідомляє, що адреса спеціального модуля є недійсною.

S2: Початкова адреса буфера BFM спеціального модуля, який потрібно зчитати

Діапазон налаштування: від 0 до 32767. При зверненні до недопустимої BFM адреси система повідомляє, що BFM одиниця оцінюваного спеціального модуля виходить за межі діапазону.

D: Елемент, у якому зберігаються дані, зчитані зі спеціального модуля. **S3:** Кількість послідовних BFM буферів (подвійне слово) для зчитування

Діапазон даних: від 1 до 32767. При зверненні до недійсної адреси буфера BFM система повідомляє, що буфер BFM оцінюваного спеціального модуля виходить за межі діапазону.

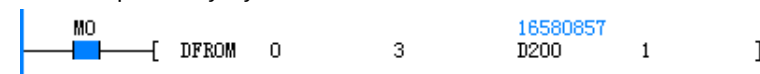
Опис функції

Послідовне зчитування **S3** даних слів, починаючи з буфера **S2** у визначеному BFM спеціального модуля (SN: **S1**), та послідовне збереження цих даних у **S3** елементах подвійних слів, починаючи з **D**.

Примітка

Час виконання інструкції DFROM є відносно довгим і залежить від значення **S3**.

Екземпляр застосунку



LD M0

DFROM 0 3 D200 1

Зчитування подвійних даних, починаючи з буфера 3 у BFM спеціального модуля № 0, та збереження цих подвійних даних у D200 та D201 відповідно, коли M0 є ON.

6.8.3 TO: Інструкція для запису слів із буфера спеціального модуля регістр

LAD:										Застосовна модель	VC2 VC3 VC5			
										Прапорець впливу біт				
IL: TO (S1) (S2) (S3) (S4)										Довжина кроку	9			
Робота d	Тип	Застосовний програмний елемент												Показник
S1	INT	Константа												
S2	INT	Константа												
S3	INT	Константа							D			V	R	✓
S4	INT	Константа												

Операнд опис

S1: SN спеціальних модулів відвідано

Діапазон налаштування: від 0 до 7. При спробі доступу до спеціального модуля, який відсутній у системі, система повідомляє, що адреса

спеціального модуля є недійсною.

S2: Початкова адреса буфера BFM спеціального модуля, що підлягає запису

Діапазон налаштування: від 0 до 32767. При зверненні до недопустимої BFM адреси система повідомляє, що BFM одиниця оцінюваного спеціального модуля виходить за межі діапазону.

S3: Дані для запису у спеціальні модулі

S4: Кількість послідовних буферів BFM (одне слово), які потрібно записати.

Діапазон даних: від 1 до 32767.

При зверненні до недійсної адреси буфера BFM, система повідомляє, що буфер BFM оцінюваного спеціального модуля виходить за межі діапазону.

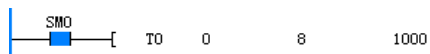
- Опис функції

Запис даних із послідовних **S4** слів, починаючи з **S3**, у **S4** елементи слів, починаючи з визначеного BFM буфера (адреса **S2**) визначеного спеціального модуля (SN: **S1**).

- Примітка

Час виконання інструкції TO є відносно тривалим і залежить від значення **S4**.

- Екземпляр застосунку



Запис значення 1000 відповідно у буфери BFM #8 та BFM #9 спеціального модуля № 0, коли PLC працює.

6.8.4 DTO: Інструкція для запису подвійних слів із буфера спеціального модуля регістр

LAD:  [DTO (S1) (S2) (S3) (S4)]										Застосовна модель	VC2 VC3 VC5					
										Прапорець впливу біт						
IL: DTO (S1) (S2) (S3) (S4)										Довжина кроку	10					
Робота d	Тип	Застосовний програмний елемент														Показчик ng
S1	INT	Константа														
S2	INT	Константа														
S3	DINT	Константа							D				V		R	✓
S4	INT	Константа														

- Операнд опис

S1: SN спеціальних модулів, що відвідуються. Діапазон налаштування: від 0 до 7. При зверненні до спеціального модуля, якого немає в системі, система повідомляє, що the адреса спеціального модуля недійсна.

S2: Початкова адреса буфера BFM спеціального модуля для запису. Діапазон налаштування: від 0 до 32767. При зверненні до недійсної адреси BFM привод система повідомляє, що BFM одиниця цього оцінений спеціальний модуль перевищує діапазон.

S3: Дані для запису у спеціальні модулі

S4: Кількість послідовних буферів BFM (подвійне слово), що підлягають запису.

Діапазон даних: від 1 до 32767. При спробі доступу до недійсної адреси буфера BFM система повідомляє, що буфер BFM оцінюваного спеціального модуля виходить за межі діапазону.

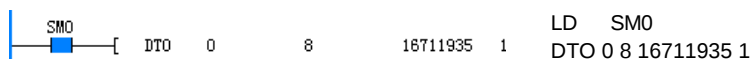
- Опис функції

Запис даних з послідовних **S4** подвійних слів, починаючи з **S3**, у **S4** елементи подвійних слів, починаючи з визначеного BFM буфера (адреса **S2**) визначеного спеціального модуля (SN: **S1**).

- Примітка

Час виконання інструкції DTO є відносно довгим і залежить від значення **S4**.

- Екземпляр застосунку



Запис одного подвійного слова даних (16711935) у буфери BFM #8 та BFM #9 (вони можуть утворювати одне подвійне слово) спеціального модуля № 0, коли PLC перебуває у стані RUN.

6.8.5 VRRD: Інструкція для зчитування значення аналогового потенціометра

LAD:  VRRD (S) (D)										Застосовна модель	Резерв					
										Прапорець впливу біт						
IL: VRRD (S) (D)										Довжина кроку	5					
Робота d	Тип	Застосовний програмний елемент														Показчик

														к ng
S	WORD	Стан												
D	WORD								D				V	√

- Операнд опис

S: Призначений аналоговий потенціометр №

Діапазон налаштування: від 0 до 255. Якщо **S** встановлено поза цим діапазоном, система повідомляє про помилку операнда.

D: Елемент, у якому зберігається зчитане значення аналогового потенціометра. Діапазон: 0–255.

- Опис функції

Зчитування значення призначеного аналогового потенціометра та його збереження у призначений елемент.

- Екземпляр застосунку



 LD M0

VRRD 0 D10

запис зчитаного значення у D10, коли M0 — ON.

Зчитування значення аналогового потенціометра № 0 у системі, та

6.8.6 REFF: Інструкція щодо встановлення константи фільтрації вхідного сигналу

LAD: [REFF (S)]									Застосовна модель			VC1S VC1						
									Прапорець впливу біт									
IL: REFF (S)									Довжина кроку			3						
Робота d		Тип		Застосовний програмний елемент													Показник ng	
S		INT		Постійна т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

S: Постійна фільтрації вхідного сигналу constant

- VC1: Фактичними дійсними даними є 0, 8, 16, 32 та 64. Будь-який параметр менше 8 обробляється як 0, а менше 16 — як 8. Параметр менше 32 обробляється як 16, а менше 64 — як 32, інші дані обробляються як 64.

- Опис функції

Налаштування сталої фільтрації входу X0–X7.

- Примітка

Константа фільтрації входу дійсна лише для порту, що використовується як звичайний вхід, і вона недейсна для порту, що використовується як високошвидкісний вхід.

- Екземпляр застосунку



Встановлення часу сталої фільтрації входу на 30ms, коли X10 дорівнює ON.

6.8.7 REF: Інструкція для негайного оновлення I/O

LAD: REF (D) (S)										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт							
IL: REF (D) (S)										Довжина кроку		5					
Робота d	Тип	Застосовний програмний елемент													Показник		
D	BOOL		X	Y													
S	INT	Постійна т															

- Операнд опис

D: Початковий X або Y елемент, який потрібно оновити

Призначення початкового номера програмного елемента як цілого числа, кратного 8. Наприклад, X0, X10, X20... або Y0, Y10, Y20.... Мінімальний біт — 0.

S: Кількість портів I/O, які потрібно оновити

Це завжди має бути число, кратне 8, наприклад, 8,

16, ..., 256 тощо. Інші значення (крім кратних 8) є помилковими.

- Опис функції

Зазвичай PLC не оновлює свої точки I/O до завершення роботи користувацької програми. Однак, якщо під час процесу роботи потрібно зчитати найновіший стан входу або негайно оновити стан виходу, можна скористатися REF інструкцією.

- Примітка

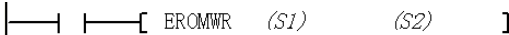
1. Індекс вхідного порту (X_n , Y_n) має бути цілим числом, кратним 8.
2. Номер оновленого (порту) також повинен бути цілим числом, кратним 8.
3. Загалом, інструкція REF використовується для негайного оновлення I/O між інструкцією FOR-NEXT та інструкцією CJ.
4. Ви також можете використовувати інструкцію REF для отримання найновішої вхідної інформації та виведення результату операції без затримки під час виконання підпрограми переривання за допомогою I/O.
5. Щоб оновити вихід реле, необхідно врахувати час відгуку.

● Екземпляр застосунок



Негайне виведення станів на Y0–Y7 незалежно від циклу сканування, коли M0 дорівнює ON.

6.8.8 EROMWR: EEPROM напишіть інструкцію

LAD:												Застосовна модель	VC1			
												Прапорець впливу біт				
IL: EROMWR (S1) (S2)												Довжина кроку	6			
Робота d	Тип	Застосовний програмний елемент														Показки ng
S1	INT								D						R	
S2	INT	Констант	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

S1: Початкова адреса елементів, що записуються (Діапазон: D6000–D6999) **S2:** Кількість елементів, що записуються ($S2 < 16$, та $S1 + S2 < D7000$)

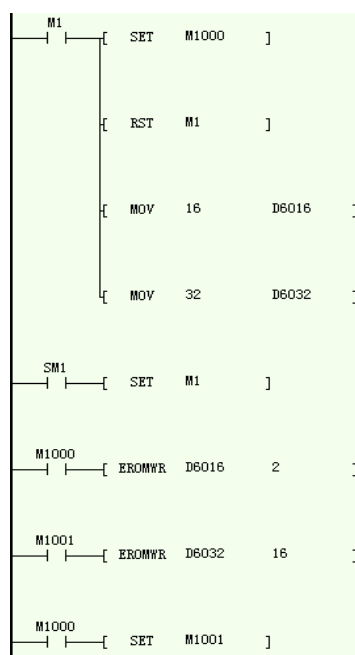
- Примітка

Одна EEPROM операція інструкції продовжує цикл сканування на 2-5 мс. Рекомендується встановити **S1** на 6000 плюс ціле число, кратне 16, наприклад D6000, D6016 та D6032.

- Опис функції

1. Загалом, часткові дані PLC можна зберігати за допомогою функції збереження даних під час зникнення живлення. Однак під час обчислення ви можете зберегти проміжні дані у EEPROM за допомогою інструкції EROMWR.
2. Ця інструкція виконується за переднім фронтом сигналу.

- Екземпляр застосунку



```
LD      M1
SET     M1000
RST     M1
MOV     16 D6016
MOV     32 D6032
LD      SM1
SET     M1
LD      M1000
EROMWR D6016 2
LD      M1001
EROMWR D6032 16
LD      M1000
SET     M1001
```

У попередньому прикладі два набори елементів D зберігаються в EEPROM. SM1 та M1 змушують M1000 згенерувати передній фронт у другому циклі сканування та ініціюють виконання першої інструкції EROMWR. M1001 створює другий передній фронт та ініціює виконання другої EROMWR інструкції

6.8.9 PR: Друк інструкції

LAD:												Придатна модель	VC2 VC3 VC5			
												Під впливом				
												прапорцевий біт				
IL: PR (S) (D)												Довжина кроку	5			
Робота d	Тип	Застосовний програмний елемент														Показки ng

S	INT								D		C	T		R	√
D	BOOL			Y											

- Операнд опис

S: Початковий номер програмних елементів, що зберігають дані.

D: Початковий номер Y елементів, що виводять дані.

- Опис функції

виконується інструкція друку. SM79 вмикається під час процесу друку та скидається після завершення друку.



Скидання SM79, коли потік енергії недейсний.

Примітка

1. Printing only once when the energy flow is valid.
2. SM79 (який вказує на те, що інструкція друку виконується) може бути використаний для керування ON/OFF інструкції друку.

Екземпляр застосунку

- Примітка
- 1. Ця інструкція застосовується лише до модулів транзисторного виходу.
- 2. Ця інструкція виконується з циклом сканування.
- 3. Одночасно можна виконати лише одну інструкцію. Після завершення друку SM79 скидається.

6.8.10 ТКУ: Введення цифрових значень інструкція

LAD:		Придатна модель																	
		VC2 VC3 VC5																	
		Під впливом																	
		прапорцевий біт																	
TKY		Довжина кроку																	
Робота	Тип	Застосовний програмний елемент																	
S	BOOL	X	Y	M	S	SM	LM												
D1	INT							KnY	KnM	KnLM									
D2	BOOL		Y	M	S	SM	LM												

- Операнд опис
 - S:** Стартовий біт даних, введенних за допомогою цифрової клавіші (займає 10 біт).
 - D1:** Блок зберігання даних unit.
 - D2:** Номер програмного елемента, що відповідає стану ON/OFF вхідної клавіші (займає 11 точок).
- Опис функції
 1. S до S+9 використовуються для введення з клавіатури, а вхідні дані зберігаються в D1; D2 до D2+9 виводять інформацію

введено через клавіатуру; D2+10 виявляє введення з клавіатури, тобто, коли будь-який один вхід є ON, цей біт встановлюється.

1) Введення значення D1

На рисунку нижче значення 2130 зберігається в D1 після натискання цифрових клавіш у порядку 1, 2, 3 та 4.

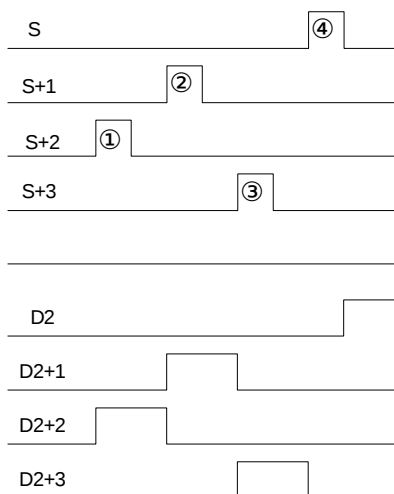
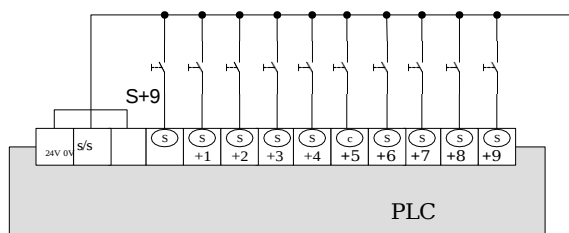
2) Ключова інформація про D2-D2+10

Чи є ключова інформація D2-D2+9 ON чи OFF, залежить від клавіші, яку ви натискаєте.

При натисканні будь-якої з клавіш 0-9 вихід виявлення клавіатури D2+10 дорівнює ON.

Цифрові клавіші

0 1 2 3 4 5 6 7 8 9



● Екземпляр застосунку

```

M0
┌───[ TKY X0 D7999 M1000 ]─── TKY X0 D7999
                                M1000

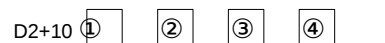
```

Коли ви натискаєте X2, X1, X3 та X0 послідовно, вміст D7999 стає 2130. Після натискання X2, M1002 встановлюється в ON до моменту натискання іншої клавіші, так само відбувається і при натисканні інших клавіш. Після натискання будь-якої з клавіш, вихід виявлення клавіатури M1010 стає ON лише під час натискання.

● Note

1. When multiple keys are pressed simultaneously, only the information input by the key pressed first is valid.
2. When the energy flow is OFF, the content of D1 does not change, but D2 to D2+10 are set to OFF.
3. If the inputted data exceeds 9999, the MSB overflows first.
4. After an input key is pressed, the keypad detection output D2 corresponding to the key is set to ON until another input key.
5. Only one TKY instruction can be used in the program, and but you can use it multiple times through indexing.

D2+9



6.9 Інструкції годинника реального часу

6.9.1 TRD: Інструкція зчитування годинника реального часу instruction

LAD: 										Застосовна модель				VC1 VC2 VC3 VC5					
										Прапорець впливу біт									
IL: TRD (D)										Довжина кроку				3					
Робота d		Тип		Застосовний програмний елемент														Індекс ація	
D		INT									D					V		R	√

- Операнд опис

D: Зчитування початкової одиниці, в якій зберігається системний час, та запис зчитаного часу у 7 послідовних одиниць, починаючи з одиниці, призначеної **D**.

- Опис функції

Зчитування системного часу та його збереження в одиниці пам'яті, призначеній **D**.

- Примітка

Інструкція TRD не виконується, якщо є помилка в налаштуванні системного годинника.

- Екземпляр застосунку



Надсилення системного часу на 7 одиниць, починаючи з D10 відповідно, коли M0 у стані ON.

Результати виконання інструкції наведено нижче:

Спеціальний регістр даних для годинника реального часу	Елемент	Проект	Тактові дані		Елемент	Проект
	SD60	Рік	2000–2099		D10	Рік
	SD61	Місяць	1–12		D11	Місяць
	SD62	День	1–31		D12	День
	SD63	Година	0–23		D13	Година
	SD64	Хвилина	0–59		D14	Хвилина
	SD65	Другий	0–59		D15	Другий
	SD66	Тиждень	0–6		D16	Тиждень

6.9.2 TWR: Інструкція запису годинника реального часу instruction

LAD: [TWR (S)]										Застосовна модель		VC1 VC2 VC3 VC5						
										Прапорець впливу біт								
IL: TWR (S)										Довжина кроку		3						
Робота d		Тип		Застосовний програмний елемент												Індексація		
S		INT								D					V		R	✓

Операнд опис

S: М'який елемент, у який має бути записаний системний час

Дані для налаштування годинника	Елемент	Проект	Тактові дані		Елемент	Проект
	D10	Рік	2000–2099		SD60	Рік
	D11	Місяць	1–12		SD61	Місяць
	D12	День	1–31		SD62	День
	D13	Година	0–23		SD63	Година
	D14	Хвилина	0–59		SD64	Хвилина
	D15	Другий	0–59		SD65	Другий
	D16	Тиждень	0–6		SD66	Тиждень

Опис функції

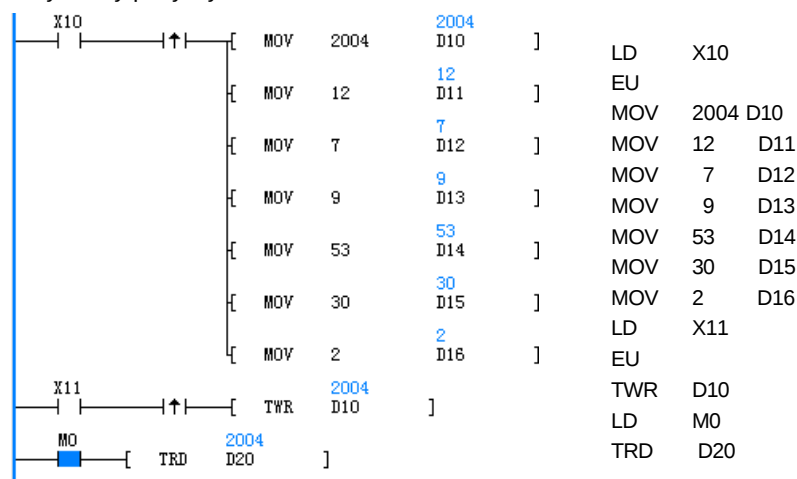
Ви можете використовувати інструкцію TWR для коригування системного часу, коли системний час відрізняється від фактичного.

Примітка

1. Записані дані часу повинні використовувати сонячний календар, інакше інструкцію неможливо виконати успішно.
2. Рекомендовано використовувати передній фронт як умову виконання цієї інструкції.

Екземпляр застосунку

Зміна системного часу за допомогою інструкції TWR, як показано на наступному рисунку.



1. Запис значень встановленого часу у 7 послідовних регістрів, починаючи з D10, при виявленні переднього фронту сигналу X10.
2. Запис значень 7 послідовних одиниць, починаючи з D10, у системний час при виявленні переднього фронту X11.
3. Зчитування системного часу та його збереження в D20, коли M0 є ON.

6.9.3 TADD: Інструкція додавання годинника інструкція

LAD:	[TADD (S1) (S2) (D)]									Застосовна модель		VC1	VC2	VC3	VC5	
										Прапорець впливу біт		Прапорець нуля SM80, прапорець переносу SM81				
	TADD (S1) (S2) (D)			Довжина кроку		7										
	Робота d	Тип	Застосовний програмний елемент										Індикація			
S1		INT							D	\$D			V		R	✓
S2		INT							D	\$D			V		R	✓
D		INT							D				V		R	✓

- Операнд опис

S1: Дані годинника 1

Збереження даних часу у трьох одиницях пам'яті, призначених для **S1**. Якщо дані не відповідають формату часу, система повідомляє про помилку, вказуючи на те, що значення операнда інструкції є недейсним.

S2: Дані годинника 2

Збереження іншого значення часу у трьох комірках пам'яті, призначених для **S2**. Якщо дані не відповідають формату часу, система повідомляє про помилку, вказуючи на те, що значення операнда інструкції є недейсним.

D: Одиниця зберігання часу результат

Дані, отримані після операції додавання часу, зберігаються у трьох одиницях пам'яті, призначених **D**. Результат впливає на прапорець переносу SM81 та прапорець нуля SM80.

- Опис функції

Виконання операції додавання над даними у форматі часу та виконання правила операції у форматі часу.

- Примітка

Дані часу під час роботи повинні відповідати формату часу:
Діапазон налаштування «Години»: 0–23.

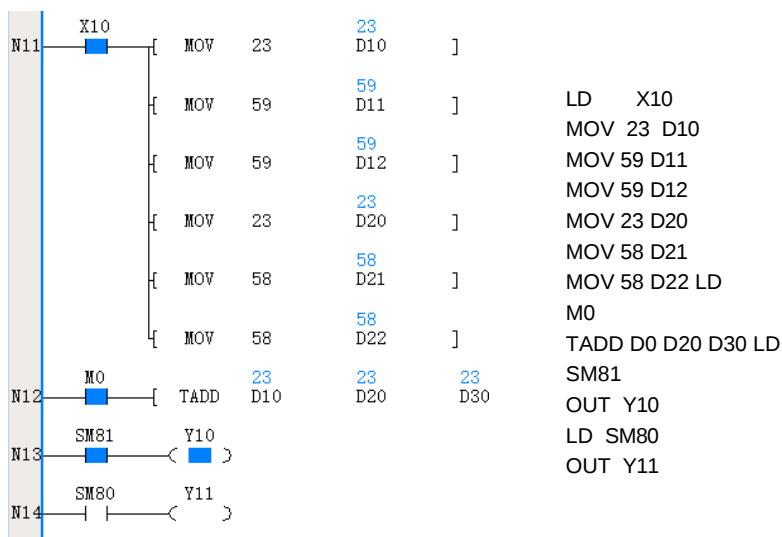
Діапазон налаштування «Хвилини»:

0–59 Діапазон налаштування

«Секунда»: 0–59

- Екземпляр застосунку

S1		+	S2		=	D	
D10	23 години		D20	23 години		D30	23 години
D11	59 хвилини		D21	58 хвилини		D31	58 хвилин
D12	59 секунди		D22	58 секунди		D32	57 секунди



1. Надсилання даних часу до трьох одиниць зберігання, починаючи з D10, та трьох одиниць зберігання, починаючи з D20, коли X10 у стані ON.

2. Збільшення значень трьох комірок пам'яті, починаючи з D10, на значення трьох комірок пам'яті, починаючи з D20, із записом отриманого результату у три комірки пам'яті, починаючи з D30, коли M0 у стані ON.

3. Встановлення прапорця переносу (SM81) у ON та прапорця нуля (SM80) у OFF.

6.9.4 TSUB: Інструкція віднімання тактових імпульсів instruction

LAD:	Застосовна модель										VC1	VC2	VC3	VC5	
											Прапорець впливу біт				Прапорець нуля SM80, прапорець переносу SM82
	[TSUB (S1) (S2) (D)]														
TADD	(S1)	(S2)	(D)	Довжина кроку 7											
Робота	Тип	Застосовний програмний елемент										Індексація			
S1	INT								D	\$D		V	R	✓	
S2	INT								D	\$D		V	R	✓	
D	INT								D			V	R	✓	

- Операнд опис

S1: Дані годинника 1

Збереження даних часу у трьох одиницях пам'яті, призначених для **S1**. Якщо дані не відповідають формату часу, система повідомляє про помилку, вказуючи на те, що значення операнда інструкції є недійсним.

S2: Дані годинника 2

Збереження іншого значення часу у трьох комірках пам'яті, призначених для **S2**. Якщо дані не відповідають формату часу, система повідомляє про помилку, вказуючи на те, що значення операнда інструкції є недопустимим.

D: Одиниця зберігання часу результат

Дані, отримані після операції додавання часу, зберігаються у трьох одиницях пам'яті, призначених **D**. Результат впливає на прапорець переносу SM81 та прапорець нуля SM80.

- Опис функції

Виконання операції віднімання над даними у форматі часу та виконання правила операції у форматі часу.

- Примітка

Дані про час під час роботи мають відповідати формату часу:

Діапазон налаштування "Години":

0–23 Діапазон налаштування

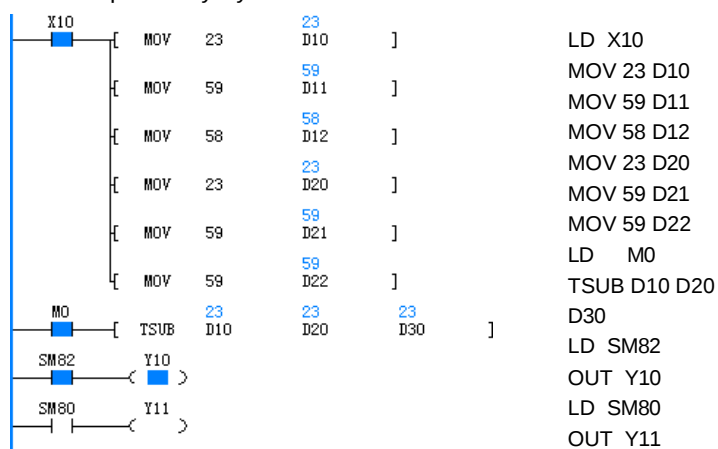
"Хвилини": 0–59 Діапазон

налаштування "Секунди": 0–59

- Екземпляр застосунку

S1		—	S2		=	D	
D10	23 години		D20	23 години		D30	23 години
D11	59 хвилин		D21	59 хвилина		D31	59 хвилина
D12	58 секунд		D22	59 другий		D32	59 секунд

- Екземпляр застосунку



1. Надсилання даних часу до трьох блоків пам'яті, починаючи з D10, та до трьох блоків пам'яті, починаючи з D20, коли X10 у стані ON.

2. Віднімання значень трьох комірок пам'яті, починаючи з D10, від значень трьох комірок пам'яті, починаючи з D20, із записом отриманого результату у три комірки пам'яті, починаючи з D30, коли M0 у стані ON.

3. Встановлення прапорця переносу (SM82) у ON та прапорця нуля (SM80) у OFF.

6.9.5 HOUR: Інструкція до хронографа інструкція

LAD:										Застосовна модель		VC1	VC2	VC3	VC5			
[HOUR (S) (D1) (D2)]										Прапорець впливу біт								
HOUR (S) (D1) (D2)										Довжина кроку		8						
Робота	Тип	Застосовний програмний елемент													Індикація			
S	INT	Кон стан	KnX		KnY	KnM	KnS	KnLM	KnSM	D	SD		C	T	V	Z	R	✓
D1	INT									D					V		R	✓
D2	BOOL				Y	M	S	LM										

- Операнд опис

S: Дані порівняння годин. Діапазон даних: 0–32767

D1: Одиниця зберігання часу unit
Одиниця даних **D1** зберігає години, тоді як одиниця даних **D1** +1 зберігає секунди.

D2: Вихід аварійного сигналу адреса

Коли дані **D1** більші або дорівнюють даним, визначеним **S**, точка спрацювання аварійного сигналу змінюється на ON вихід.

- Опис функції

Визначення часу, протягом якого вхідний контакт перебуває у стані ON (оддиниця виміру: година).

- Примітка

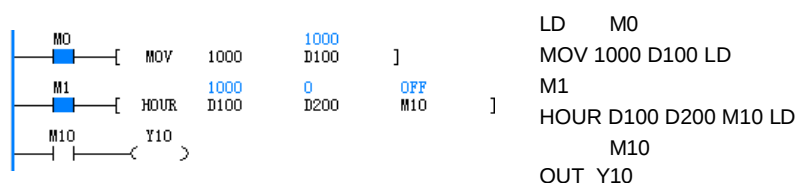
1. Призначте **D1** як одиницю програмного елемента для збереження при вимкненні живлення, щоб забезпечити

поточні дані можна використовувати після того, як PLC живлення вимкнено. Якщо використовується звичайний блок м'яких елементів, поточні дані буде очищено, коли PLC живлення вимкнено або RUN → STOP виконується операція.

2. Навіть якщо вихід аварійного сигналу **D2** має стан ON, він може продовжувати рахунок.

3. Ця година інструкції є 16-бітними цілочисельними даними. Коли дані години більші за 32767, відлік знову починається з 0.

- Екземпляр застосунку



1. Налаштування даних порівняння інструкції HOUR, коли M0 — ON.

2. HOUR виконує операцію накопичення часу на вхідних контактах, коли M1 у стані ON.

3. Коли час накопичення, тобто час перебування M1 у стані ON, дорівнює або перевищує 1000, M10 перебуває у стані ON.

6.9.6 DCMP: (= < > <> >= <=) Інструкція порівняння дати instruction

LAD:	┌───┐ ┌───┐ [DCMP= (S1) (S2) (D)]		Застосовна модель	VC1	VC2	VC3	VC5										
	┌───┐ ┌───┐ [DCMP< (S1) (S2) (D)]																
	┌───┐ ┌───┐ [DCMP> (S1) (S2) (D)]																
	┌───┐ ┌───┐ [DCMP<> (S1) (S2) (D)]																
	┌───┐ ┌───┐ [DCMP>= (S1) (S2) (D)]																
	┌───┐ ┌───┐ [DCMP<= (S1) (S2) (D)]		Прапорець впливу біт														
	DCMP= (S1) (S2) (D)																
	DCMP< (S1) (S2) (D)																
	DCMP> (S1) (S2) (D)																
	DCMP<> (S1) (S2) (D)																
DCMP>= (S1) (S2) (D)																	
DCMP<= (S1) (S2) (D)		Довжина кроку	7														
DCMP= (S1) (S2) (D)																	
DCMP< (S1) (S2) (D)																	
DCMP> (S1) (S2) (D)																	
DCMP<> (S1) (S2) (D)																	
DCMP>= (S1) (S2) (D)																	
DCMP<= (S1) (S2) (D)																	
Робота	Тип	Застосовний програмний елемент										Індексація					
S1	INT									D	\$D			V		R	√
S2	INT									D	\$D			V		R	√
D	BOOL			Y	M	S	LM					C	T				

- Операнд опис

S1: Дані порівняння дати 1, які займають три початкові одиниці слів, визначені **S1**. Дані цих трьох одиниць повинні відповідати формату сонячного календаря, інакше система повідомить про помилку операнда.

S2: Дані порівняння дати 2, які займають три початкові одиниці слів, визначені **S2**. Дані цих трьох одиниць повинні відповідати формату сонячного календаря, інакше система повідомить про помилку операнда.

D: Вихід стану порівняння. Коли дані відповідають умові порівняння, встановлюється D у ON, інакше — OFF.

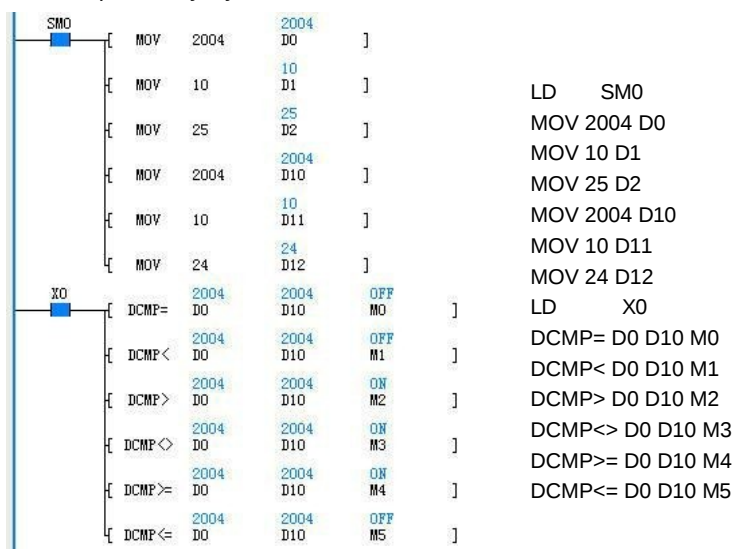
- Опис функції

Виконання порівняння BIN для даних дати, починаючи з **S1** та **S2** відповідно, і присвоєння результату порівняння до **D**.

- Примітка

Дані дати, починаючи з **S1** та **S2**, повинні відповідати формату сонячного календаря, інакше система видасть помилку операнда (наприклад: 2004, 9, 31 та 2003, 2, 29 є неприпустимими).

- Екземпляр застосунку



Виконання порівняння BIN даних дати, починаючи з D0 та D10 відповідно, і присвоєння результату порівняння цільовим даним (M0, тощо).

6.9.7 TCMP: (= < > <> >= <=) Інструкція порівняння часу instruction

LAD:										Застосовна модель		VC1 VC2 VC3 VC5					
[]	[]	TCMP=	(S1)	(S2)	(D)												
[]	[]	TCMP<	(S1)	(S2)	(D)												
[]	[]	TCMP>	(S1)	(S2)	(D)												
[]	[]	TCMP<◇	(S1)	(S2)	(D)												
[]	[]	TCMP>=	(S1)	(S2)	(D)												
[]	[]	TCMP<=	(S1)	(S2)	(D)												
IL:										Прапорець впливу біт							
TCMP= (S1) (S2) (D)																	
TCMP< (S1) (S2) (D)																	
TCMP> (S1) (S2) (D)																	
TCMP<> (S1) (S2) (D)																	
TCMP> = (S1) (S2) (D)																	
TCMP< = (S1) (S2) (D)																	
										Довжина кроку		7					
Робота d	Тип	Застосовний програмний елемент														Індексація	
S1	INT								D	SD			V		R	√	
S2	INT								D	SD			V		R	√	
D	BOOL			Y	M	S	LM				C	T					

- Операнд опис

S1: Дані порівняння часу 1, які займають початкові три одиниці слів, визначені **S1**. Дані цих трьох одиниць повинні відповідати 24-годинному формату часу, інакше система повідомить про помилку операнда.

S2: Дані порівняння часу 2, які займають початкові три слова, визначені **S2**. Дані цих трьох одиниць повинні відповідати 24-годинному формату часу, інакше система повідомить про помилку операнда.

D: Вихід стану порівняння. Коли дані відповідають умові порівняння, встановлюється D у ON, інакше — OFF.

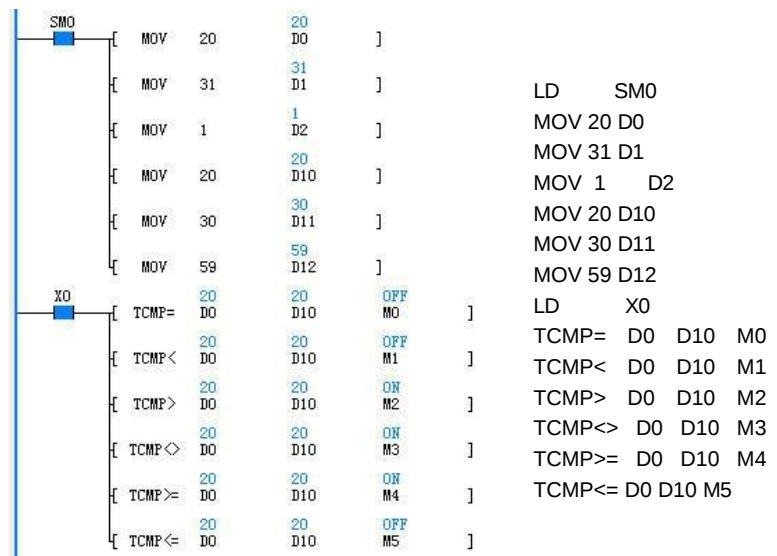
- Опис функції

Виконання порівняння BIN для даних часу, починаючи з **S1** та **S2** відповідно, і призначення результату порівняння до **D**.

- Примітка

Дані часу, що починаються з **S1** та **S2**, повинні відповідати 24-годинному формату часу, інакше система повідомляє про помилку операнда (наприклад: 24, 10, 31 та 13, 59, 60 є недопустимими).

- Екземпляр застосунку



Виконання порівняння BIN для даних часу, починаючи з D0 та D10 відповідно, і призначення результату порівняння цільовим даним (M0, тощо).

6.9.8 HTOS: Інструкція для перетворення даних у форматі години-хвилини-секунди на секунди

LAD: <								

- Операнд опис

S: Номер початкового програмного елемента, у якому зберігаються дані часу, що підлягають перетворенню.

D: Номер програмного елемента, у якому зберігаються перетворені дані часу.

- Опис функції

Перетворення даних часу (годин, хвилин і секунд) S-S+2 у секунди та збереження отриманого результату в D.

- Екземпляр застосунку



1. Перетворення даних часу (години, хвилини та секунди), починаючи з D0, у секунди та збереження отриманого результату в D10, коли M1=ON. D10=11415, коли D0=3, D1=10 та D2=15.

6.9.9 STON: Інструкція для перетворення секунд у години-хвилини-секунди дані

LAD:											Застосовна модель	VC2	VC3	VC5	
											Прапорець впливу біт				
	STON (S)		(D)		Довжина кроку 5										
Робота d	Тип	Застосовний програмний елемент												Індексація	
S	INT	KnX		KnY	KnM	KnS	T	C	D	SD				P	√
D	INT			KnY	KnM	KnS	T	C	D	SD				R	√

- Операнд опис

S: Номер програмного елемента, у якому зберігаються дані часу, що підлягають перетворенню.

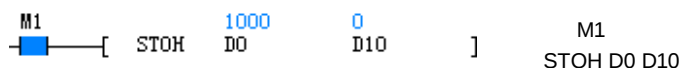
D: Номер початкового програмного елемента, у якому зберігаються перетворені дані.

- Опис функції

Перетворення другого значення даних S у години, хвилини та секунди, і збереження отриманого результату у D, D+1 та D+2 відповідно.

- Примітка

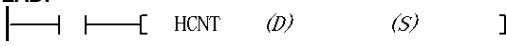
- Екземпляр застосунку



1. Перетворення другого значення D0 у дані часу (години, хвилини та секунди) та збереження отриманого результату у трьох регістрах, починаючи з D10, коли M1=ON. D10=0, D11=16 та D12=40, коли D0=1000.

6.10 Високошвидкісні I/O інструкції

6.10.1 HCNT: Інструкція для керування високошвидкісним лічильником

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт							
IL: HCNT (D) (S)										Довжина кроку		7					
Робота d	Тип	Застосовний програмний елемент															Індексація
D	DINT										C						
S	DINT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	

- Операнд опис

D: Призначення номера лічильника. Діапазон налаштування: C236—C263

S: Призначення константи порівняння, 32-бітні дані зі знаком. Діапазон даних: -2147483648 — +2147483647.

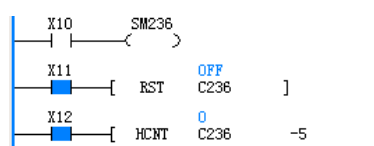
- Опис функції

Керування призначеним апаратним високошвидкісним лічильником. Виконання апаратного високошвидкісного підрахунку на всіх високошвидкісних лічильниках лише за умови безперервного керування ними, та визначення дії NO контактів високошвидкісного лічильника відповідно до **S**.

- Примітка

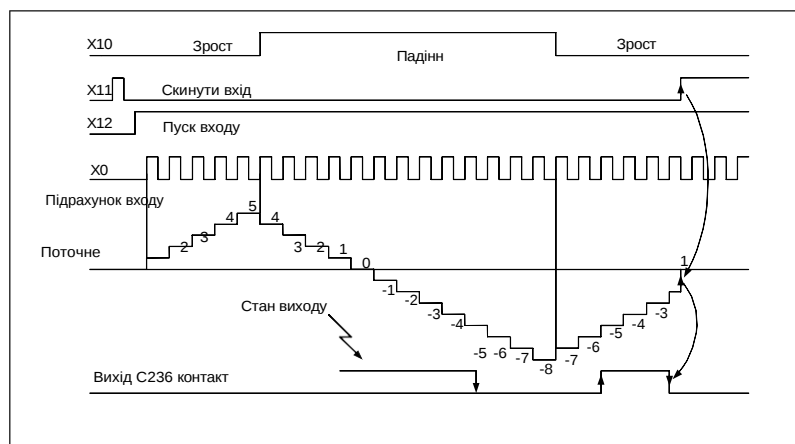
Існують апаратні конфлікти між інструкцією HCNT, інструкцією SPD, перериванням від зовнішнього входу та захопленням імпульсів. Вам необхідно звернути увагу на умови використання всіх високошвидкісних I/O. Зверніться до розділу 8 "Посібник з експлуатації функції високошвидкісного вводу" для використання.

- Екземпляр застосунку



```
LD X10
OUT SM236
LD X11
RST C236
LD X12
HCNT C236 -5
```

Часову послідовність роботи програмного керування наведено нижче:



6.10.2 DHSCS: Інструкція з налаштування порівняння високошвидкісного лічильника

LAD:					Застосовна модель	VC1	S	VC1	VC2	VC3	VC5
					Прапорець впливу біт						
DHSCS		(S1)	(S2)	(D)	Довжина кроку	10					
Робота	Тип	Застосовний програмний елемент									Індексація

S1	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
S2	DINT										C					
D	BOOL			Y	M	S										

- Операнд опис

S1: Дані для порівняння високошвидкісним лічильником, 32-бітні DINT дані, в діапазоні від -2147483648 до +2147483647.

S2: Високошвидкісний лічильник. Діапазон застосування: C236–C263

D: об'єкт виходу бітового елемента, що негайно встановлює вихід Y, M, та S незалежно від періоду сканування.

- Примітка

1. Інструкція DHSCS має працювати разом з інструкцією HCNT. DHSCS може бути виконана коректно лише тоді, коли високошвидкісний лічильник приводиться в дію за допомогою HCNT.

2. Результат, порівняний інструкцією DHSCS, набуває чинності, коли надходять зовнішні імпульси високошвидкісного лічильника. Тому інструкція DHSCS не виконує жодної дії, навіть якщо значення високошвидкісного лічильника змінено за допомогою інструкції DMOV або MOV інструкції.

3. Як і загальні інструкції, декілька DHSCS (DHSCI, DHSCR, DHSZ, DHSP та DHST) інструкції можна використовувати одночасно, але загальна кількість цих інструкцій не може перевищувати 8. Перші 8 інструкцій виконуються послідовно, а 9th або пізніші

інструкції є неефективними і тому не виконуються.

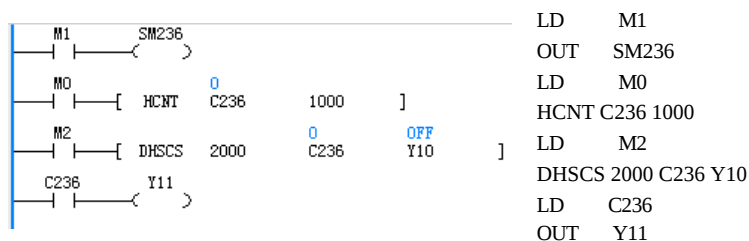
4. Макс. частота, що підтримується високошвидкісним лічильником PLCs. Якщо ви використовуєте інструкцію DHSCS, DHSCI, DHSCR, DHSZ, DHSP або DHST, вона обмежується макс. частотою відгуку та комплексною частотою. Детальніше див. у розділі 8 «Посібник з експлуатації функції високошвидкісного входу».

- Опис функції

1. А високошвидкісний лічильник рахує в режимі переривання лише тоді, коли він керується інструкцією HCNT, а вхідний сигнал підрахунку змінюється з OFF на ON. Коли значення високошвидкісного лічильника дорівнює **S1** в інструкції DHSCS, бітовий елемент, визначений **D**, встановлюється негайно, або у випадку елемента Y, елемент Y виводиться негайно.

2. Ця інструкція може бути використана для негайного виведення результату порівняння на основі налаштувань порівняння для поточного значення високошвидкісного лічильника.

- Екземпляр застосунку



1. Коли M1 має значення ON, C236 виконує підрахунок у режимі переривання, коли X0 змінюється з OFF на ON (щодо вхідної частоти X0 див. інструкцію для високошвидкісного I/O). Коли C236 змінюється з 999 на 1000, контакт C236 встановлюється, а коли C236 змінюється з 1001 на 1000, контакт C236 скидається. Коли Y11 активується за допомогою C236, виконання Y11 визначається циклом сканування користувацької програми.

2. Коли M2 має значення ON, а високошвидкісна інструкція DHSCS відповідає вимогам, зазначеним у попередній "Примітка", Y10 видається негайно, якщо C236 досягає 2000, незалежно від циклу сканування.

3. Коли M0 дорівнює ON, SM236 активується, а лічильник C236 рахує у зворотному напрямку. Коли M0 дорівнює OFF, SM236 не активується, а лічильник C236 рахує у прямому напрямку.

6.10.3 DHSCI: Інструкція для ініціювання переривання на основі порівняння високошвидкісного рахунку

LAD:	[DHSCI (S1) (S2) (S3)]			Застосовна модель	VC1	VC2	VC3	VC5
				Прапорець впливу біт				
DHSCI		(S1)	(S2)	(S3)	Довжина кроку	10		
Робота	Тип	Застосовний програмний елемент						Індексація

S1	DINT	Складається немає KPX	KnY	KnM	KnS	KnLM	KnSM		D	SD		C		V		R	√
S2	DINT											C					
S3	INT	Складається немає															

- Операнд опис

S1: Дані для порівняння високошвидкісним лічильником, 32-бітні дані DINT, у діапазоні від -2147483648 до +2147483647.

S2: Високошвидкісний лічильник
Застосовний діапазон:
C236–C263

S3: Номер переривання. Діапазон номера переривання: 33-40

- Опис функції

А високошвидкісний лічильник може рахувати в режимі переривання лише тоді, коли він керується інструкцією HCNT, а вхідний сигнал підрахунку змінюється з OFF на ON. Коли значення високошвидкісного лічильника дорівнює **S1** в інструкції DHSCS, він переходить до підпрограми переривання, визначеної **S3**. Ви можете написати програму, яка буде виконана негайно в підпрограмі переривання.

- Примітка

1. Інструкція DHSCI повинна працювати разом з інструкцією HCNT. DHSCI може бути виконана правильно лише тоді, коли високошвидкісний лічильник приводиться в дію за допомогою HCNT.

2. Результат, порівняний інструкцією DHSCI, набуває чинності, коли надходять зовнішні імпульси високошвидкісного лічильника. Тому інструкція DHSCI не виконує жодної дії, навіть якщо значення високошвидкісного лічильника змінено за допомогою інструкції DMOV або MOV.

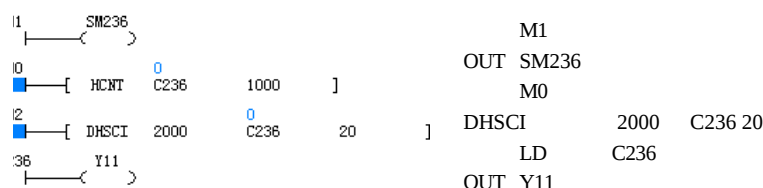
3. Як і загальні інструкції, декілька DHSCI (DHSCS, DHSCR, DHSZ, DHSP, та DHST) інструкції можуть

можуть використовуватися одночасно, але загальна кількість цих інструкцій не може перевищувати 8. Перші 8 інструкцій виконуються по порядку, а 9-та або наступні інструкції не є дійсними і тому не виконуються.

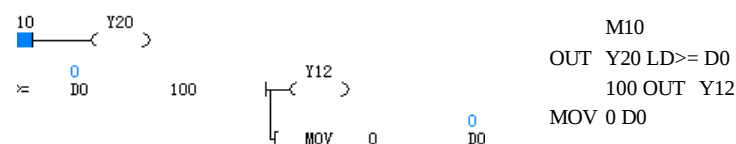
4. Макс. частота, що підтримується високошвидкісним лічильником PLCs. Якщо ви використовуєте інструкцію DHSCS, DHSCI, DHSCR, DHSZ, DHSP або DHST, вона обмежена макс. частотою відгуку та загальною частотою. Детальніше див. у розділі 8 "Посібник з експлуатації функції високошвидкісного вводу".

- Екземпляр застосунку

Головна програма користувача наведена нижче:



Програма переривання, номер користувацького переривання якої 20, показана нижче:



1. Коли M1 має значення ON, C236 виконує підрахунок у режимі переривання, коли X0 змінюється з OFF на ON (щодо вхідної частоти X0 див. інструкцію для високошвидкісного I/O). Коли C236 змінюється з 999 на 1000, контакт C236 встановлюється, а коли C236 змінюється з 1001 на 1000, контакт C236 скидається. Коли Y11 активується за допомогою C236, виконання Y11 визначається циклом сканування користувацької програми.

2. Коли M2 дорівнює ON, а високошвидкісна інструкція DHSCI відповідає вимогам, зазначеним у попередній "Примітка", підпрограма переривання, номер переривання якої 20, реагує негайно та виконує користувацьку програму в програмі переривання, коли C236 досягає 2000.

3. Коли M1 має значення ON, SM236 приводиться в дію, а лічильник C236 здійснює зворотний відлік. Коли M1 має значення OFF, SM236 не приводиться в дію, а лічильник C236 здійснює прямий відлік.

4. Вхід у програму переривання №20, коли C236 має імпульсний вхід, а C236 дорівнює 2000. Керування Y20, коли M10 дорівнює ON. Проте вихід Y20 залежить від циклу сканування користувацької програми. Водночас керування Y12 та очищення даних D0, коли виявлено, що D0 більше 100.

6.10.4 DHSPi: Ініціювання переривання за інструкцією на основі порівняння абсолютних високошвидкісних вихідних позицій

LAD:		Застосовна модель										VC2		VC3			
		Прапорець впливу біт															
DHSPi		(S1)	(S2)	(S3)	Довжина кроку										10		
Робота d	Тип	Застосовний програмний елемент												Індексація			
S1	DINT	Кон стан	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C		V		R	√
S2	DINT										SD						
S3	INT	Кон стан т															

- Операнд опис

S1: Дані для порівняння елементом позиціонування високошвидкісного виходу, 32-бітні DINT дані, діапазон від -2147483648 до +2147483647.

S2: Елемент позиції високошвидкісного виходу. Застосовний діапазон:

S3: Номер переривання. Діапазон номера переривання: 53, 54, 55, 56, 57, 58, 59 та 60.

- Опис функції

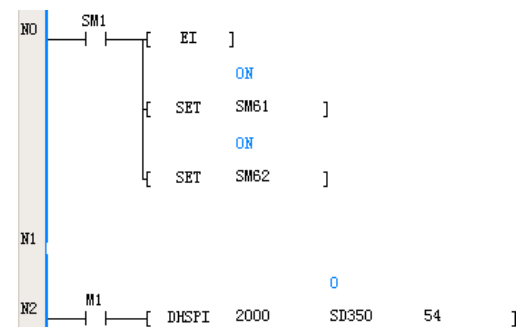
Коли значення елемента позиції високошвидкісного виходу дорівнює **S1** в інструкції DHSPi, він переходить до підпрограми переривання, визначеної **S3**. Ви можете написати програму, яка буде негайно виконана в підпрограмі переривання.

- Примітка

1. Коли дані записуються в елемент SD, переривання за позицією не ініціюється. Після запису даних переривання за позицією ініціюється за вказаною позицією.

- Екземпляр застосунку

Головна програма користувача наведена нижче:



Номер переривання підпрограми переривання можна встановити на 54 або інше джерело переривання високошвидкісного виходу, що базується на позиції, а потім програму, яка має виконуватися під час проходження позиції, можна записати в підпрограму переривання.

6.10.5 DHSCR: Інструкція щодо скидання порівняння високошвидкісного лічильника

LAD: DHSCR (S1) (S2) (D)										Застосовна модель		VC1S VC1 VC2 VC3 VC5											
												Прапорець впливу біт											
IL: DHSCR										(S1)		(S2)		(D)		Довжина кроку		10					
Робота d		Тип		Застосовний програмний елемент														Індексація					
S1		DINT		Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓					
S2		DINT											C										
D		BOOL				Y	M	S					C										

- Операнд опис

S1: Дані для порівняння високошвидкісним лічильником, 32-бітні DINT дані, в діапазоні від -2147483648 до +2147483647.

S2: Високошвидкісний лічильник. Діапазон застосування: C236–C263.

D: Об'єкт виходу бітового елемента, що негайно скидає вихід Y, M, S та C незалежно від періоду сканування. Якщо D є елементом C, він має бути **S2**.

- Опис функції

А високошвидкісний лічильник може рахувати в режимі переривання лише тоді, коли він керується інструкцією HCNT, а вхідний сигнал підрахунку змінюється з OFF на ON. Коли значення високошвидкісного лічильника дорівнює **S1** в інструкції DHSCR, бітовий елемент, визначений **D**, негайно скидається, або у випадку елемента Y, елемент Y виводиться негайно. 2. Ця інструкція може бути використана для негайного виведення результату порівняння на основі налаштування порівняння для поточного значення високошвидкісного лічильника.

- Примітка

1. Інструкція DHSCR повинна працювати разом з інструкцією HCNT. DHSCR може бути

виконана коректно лише тоді, коли високошвидкісний лічильник приводиться в дію за допомогою HCNT.

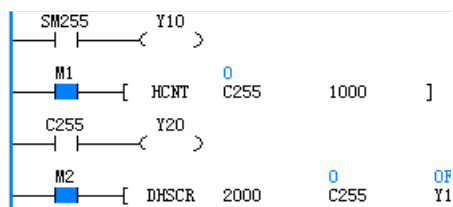
2. Результат, порівняний інструкцією DHSCR, набуває чинності, коли

вводяться зовнішні імпульси високошвидкісного лічильника. Тому інструкція DHSCR не виконує жодної дії, навіть якщо значення високошвидкісного лічильника змінено за допомогою інструкції DMOV або MOV.

3. Як і загальні інструкції, декілька DHSCR (DHSCI, DHSCS, DHSZ, DHSP, та DHST) інструкцій можуть бути використані одночасно, але загальна кількість цих інструкцій не може перевищувати 8. Перші 8 інструкцій виконуються по порядку, а 9-та або пізніші інструкції не є ефективними і тому не виконуються.

4. Максимальна частота, що підтримується високошвидкісним лічильником PLCs. Якщо ви використовуєте інструкцію DHSCS, DHSCI, DHSCR, DHSZ, DHSP або DHST, вона обмежується максимальною частотою відгуку та комплексною частотою. Детальнішу інформацію див. у розділі 8 "Посібник з експлуатації функції високошвидкісного вводу".

● Екземпляр застосунку



1. Коли M1 та X7 обидва ON, C255 рахує різницю фаз X3 та X4 у режимі переривання (щодо вхідної частоти X0, зверніться до інструкції для високошвидкісного I/O). Коли C255 змінюється з 999 на 1000, контакт C255 встановлюється, а коли C255 змінюється з 1001 на 1000, контакт C255 скидається. Коли Y20 керується приводом C255, виконання Y20 визначається циклом сканування користувачької

програми.

2. Коли M2 дорівнює ON, а високошвидкісна інструкція DHSCR відповідає вимозі, зазначеній у попередній "Примітка", Y1 видається негайно, якщо C255 досягає 2000, незалежно від циклу сканування.

3. Коли вхідний імпульс X3 випереджає імпульс X4, SM255 — це ON; коли вхідний імпульс X4 випереджає імпульс X3, SM255 — це OFF.

4. Коли X7 (сигнал запуску C255) має значення OFF, лічильник C255 не може рахувати.

5. Коли M1 та X7 обидва ON, якщо X5 ON, лічильник C255 скидається, а допоміжний контакт C255 також скидається.

6.10.6 DHSZ: Інструкція порівняння діапазону високошвидкісного підрахунку інструкція

LAD:	Застосовна модель										VC1S VC1 VC2 VC3 VC5					
	Прапорець впливу біт															
DHSZ (S1) (S2) (S3) (D) Довжина кроку 13																
	Робота	Тип	Застосовний програмний елемент												Індексація	
S1		DINT	Констант	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V	R	√
S2		DINT	Постійна	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V	R	√
S3		DINT										C				
D		BOOL			Y	M	S									

- Операнд опис

S1: Дані 1 для порівняння високошвидкісним лічильником, 32-бітні DINT дані, в діапазоні від -2147483648 до +2147483647.

S2: Дані 2 для порівняння високошвидкісним лічильником, 32-бітним DINT дані, в діапазоні від 2147483648 до +2147483647.

S3: Високошвидкісний лічильник. Діапазон застосування: C236–C263

D: об'єкт виходу бітового елемента, що негайно встановлює вихід Y, M, та S незалежно від періоду сканування.

- Опис функції

1. А Високошвидкісний лічильник може рахувати в режимі переривання лише тоді, коли він керується інструкцією HCNT, а вхідний сигнал підрахунку змінюється з OFF на ON.

2. Встановлення бітового елемента, визначеного **D**, та скидання двох бітових елементів, що йдуть після того, який визначено **D**, коли значення високошвидкісного лічильника менше за **S1** в інструкції.

3. Скидання бітового елемента, визначеного **D**, та встановлення двох бітових елементів, що йдуть за тим, який визначено **D**, коли

значення високошвидкісного лічильника більше або дорівнює **S1** і менше або дорівнює **S2**.

4. Скидання бітового елемента, визначеного **D**, та скидання двох бітових елементів, що йдуть після того, який визначений **D**, коли значення

високошвидкісний лічильник більший за S2 в DHSZ інструкції.

5. Якщо це елемент Y, елемент Y негайно видає відповідний стан, а вихідна дія не залежить від циклу сканування програми.

- Примітка

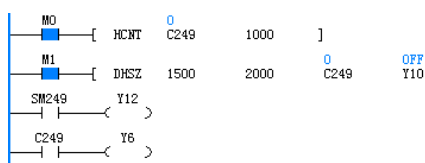
1. Інструкція DHSZ повинна працювати разом з інструкцією HCNT. DHSZ може бути виконана коректно лише тоді, коли високошвидкісний лічильник приводиться в дію за допомогою HCNT.

2. Результат, порівняний інструкцією DHSZ, набуває чинності при надходженні зовнішніх імпульсів високошвидкісного лічильника. Тому інструкція DHSZ не виконує жодної дії, навіть якщо значення високошвидкісного лічильника змінено за допомогою інструкції DMOV або MOV.

3. Як і загальні інструкції, декілька інструкцій DHSCZ (DHSCI, DHSCS, DHSCR, DHSP та DHST) можна використовувати одночасно, але загальна кількість цих інструкцій не може перевищувати 8. Перші 8 інструкцій виконуються по порядку, а 9-та або наступні інструкції є неефективними і тому не виконуються.

4. Макс. частота, що підтримується високошвидкісним лічильником PLCs. Якщо ви використовуєте інструкцію DHSCS, DHSCI, DHSCR, DHSZ, DHSP або DHST, вона обмежена максимальною частотою відгуку та загальною частотою. Для отримання деталей зверніться до розділу 8 "Посібник з експлуатації функції високошвидкісного вводу".

- Екземпляр застосування



1. Коли M0 дорівнює ON, C249 рахує вгору, коли X2 змінюється з OFF на ON, або рахує вниз, коли X3 змінюється з OFF на ON (щодо вхідної частоти X0, зверніться до інструкції для високошвидкісного I/O). Коли C249 змінюється з 999 на 1000, контакт C249 встановлюється, а коли C249 змінюється з 1001 на 1000, контакт C249 скидається. Коли Y6 активується приводом C249, виконання Y6 визначається циклом сканування користувацької програми.

2. Коли M1 дорівнює ON, а високошвидкісна інструкція DHSZ відповідає вимогам, зазначеним у попередній "Примітка", стан Y10, Y11 та Y12 є таким, як наведено нижче:

(1) $C249 < 1500$: Y10: ON; Y11, Y12:

OFF °

Виходи Y10, Y11 та Y12 не залежать від циклу сканування.

(2) $2000 \geq C249 \geq 1500$: Y10, Y12:

OFF; Y11: ON °

3. Коли M0 дорівнює ON, якщо X2 рахує вгору від OFF до ON, SM249 скидається. Якщо X3 рахує вниз від OFF до ON, SM249 встановлюється.

(3) $C249 > 2000$: Y10, Y11: OFF; Y12:

ON °

6.10.7 DHST: Інструкція порівняння таблиці високошвидкісного лічильника instruction

LAD:	Застосовна модель										VC1S VC1 VC2 VC3 VC5				
	Прапорець впливу біт														
DHST (S1) (S2) (S3) Довжина кроку 10															
Робота d	Тип	Застосовний програмний елемент													Індексація
S1	DINT								D					R	
S2	INT	Константа мурха													
S3	DINT									C					

- Операнд опис

S1: Початкова одиниця даних для порівняння таблиць (початковий номер елемента D). Наступні три елементи D з послідовними порядковими номерами використовуються для визначення даних, що порівнюються високошвидкісним лічильником, порядкового номера та відповідного стану виходу елемента Y. Елементи D, пов'язані з цими чотирма послідовними порядковими номерами, разом називаються одним записом.

S2: Кількість записів для порівняння. Діапазон даних: 1–128

S3: Високошвидкісний лічильник. Діапазон застосування: C236–C263.

- Опис функції

1. А Високошвидкісний лічильник може рахувати в режимі переривання лише тоді, коли він керується інструкцією HCNT, а вхідний сигнал підрахунку змінюється з OFF на ON.

2. Коли значення високошвидкісного лічильника дорівнює даним, що підлягають порівнянню в поточний момент, стан відповідного елемента Y виводиться згідно із записаними даними, а об'єктом виводу може бути лише елемент Y.

3. Вихідна дія не залежить від циклу сканування, і елемент Y, визначений поточним записом, негайно видає визначений стан.

4. Якщо бажано, щоб користувачка програма виконувала операцію негайного виводу відповідно до порівняння

дані, визначені певною таблицею та Y елементами, ви можете використовувати інструкцію DHST.

- Примітка

1. Інструкція DHST повинна працювати разом з інструкцією HCNT. DHST може бути виконана коректно лише тоді, коли високошвидкісний лічильник приводиться в дію за допомогою HCNT.

2. Результат, порівняний інструкцією DHST, набуває чинності при надходженні зовнішніх імпульсів високошвидкісного лічильника. Тому інструкція DHST не виконує жодної дії, навіть якщо значення високошвидкісного лічильника змінено за допомогою інструкції DMOV або MOV.

3. Як і загальні інструкції, декілька інструкцій DHST (DHSCI, DHSCS, DHSCR, DHSP та DHSZ) можуть використовуватися одночасно, але загальна кількість цих інструкцій не може перевищувати 8. Перші 8 інструкцій виконуються по порядку, а 9-та або наступні інструкції є неефективними і тому не виконуються.

Якщо DHSP є дійсною інструкцією в інструкції користувача, DHST не виконується. Навпаки, якщо DHST є дійсною інструкцією, DHSP не виконується. Лише одна інструкція (DHST або DHSP) може бути дійсною одночасно в програмі користувача.

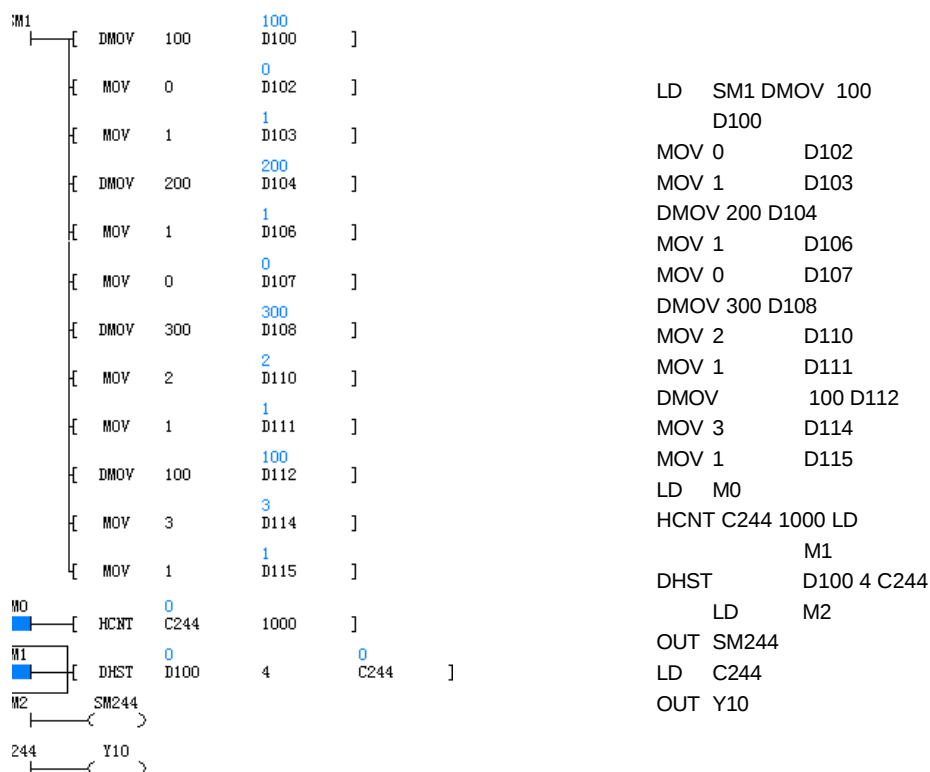
4. Макс. частота, що підтримується високошвидкісним лічильником PLCs. Якщо ви використовуєте інструкцію DHSCS, DHSCI, DHSCR, DHSZ, DHSP або DHST, вона обмежується макс. частотою відгуку та комплексною частотою. Детальнішу інформацію див. у розділі 8 "Посібник з експлуатації функції високошвидкісного входу".

● Екземпляр застосунку

Дані таблиці наведено нижче:

Порівняльні дані		Y число	Встановити/Скинути	Операція процесу
MSB	LSB			
D100=0	D101=100	D102=0	D103=1	1 ↓
D104=0	D105=200	D106=1	D107=0	2 ↓
D108=0	D109=300	D110=2	D111=1	3 ↓
D112=0	D113=300	D114=3	D115=1	4 ↓
Повернутися до 1				

LAD показано нижче:



1. Присвоєння початкового значення D100–D115 та створення таблиці для порівняння під час першого циклу сканування користувацької програми.

2. Коли M0 дорівнює ON, C244 веде підрахунок, коли X0 змінюється з OFF на ON (щодо вхідної частоти X0, зверніться до інструкції для високошвидкісного I/O). Коли C244 змінюється з 999 на 1000, C244 встановлюється, а коли C244 змінюється з 1001 на 1000, C244 скидається. Коли Y10 активується за допомогою C244, виконання Y10 визначається циклом сканування користувацької програми.

3. Коли M1 має значення ON, а високошвидкісна інструкція DHST відповідає вимогам, зазначеним у попередній «Примітці», високошвидкісна інструкція DHST починається із запису №1 таблиці та переходить до порівняння запису №2 лише після завершення запису №1. До порівняння наступного запису можна перейти лише після завершення попередньої інструкції. Коли порівняння останнього запису завершено, воно знову повертається до порівняння першого запису та встановлює SM83. SD90 вказує номер запису, що порівнюється в даний момент, тоді як SD88 та SD89 вказують дані, що порівнюються в даний момент. Результат порівняння виводиться негайно, незалежно від циклу сканування.

4. Коли M2 дорівнює ON, SM244 дорівнює ON, а C244 здійснює зворотний відлік. Якщо M2 дорівнює OFF, SM244 дорівнює OFF, а C244 здійснює відлік вгору.

6.10.8 DHSP: Інструкція для імпульсного виходу на основі таблиці високошвидкісного підрахунку порівняння

LAD:				Застосовна модель	VC1	S	VC1	VC2	VC3	VC5
	[DHSP (S1) (S2) (S3)]			Прапорець впливу біт						
	DHSP	(S1)	(S2)	(S3)	Довжина кроку					

Робота d	Тип	Застосовний програмний елемент												Індексація	
S1	DINT								D					R	
S2	INT	Пос- тійна													
S3	DINT										C				

- Операнд опис

S1: Початкова одиниця даних для порівняння таблиць (початковий номер елемента D). Наступні три елементи D з послідовними серійними номерами використовуються для визначення даних, що порівнюються високошвидкісним лічильником, та даних, що виводяться на SD86 та SD87. Елементи D, пов'язані з цими чотирма послідовними серійними номерами, разом називаються одним записом.

S2: Кількість записів для порівняння. Діапазон даних: 1–128.

S3: Високошвидкісний лічильник. Діапазон застосування: C236–C263.

- Опис функції

1. А Високошвидкісний лічильник може рахувати в режимі переривання лише тоді, коли він керується інструкцією HCNT, а вхідний сигнал підрахунку змінюється з OFF на ON.

2. Коли значення високошвидкісного лічильника дорівнює даним порівняння поточного запису, SD86 та SD87 змінюються відповідно до вихідних даних поточних даних.

3. Якщо необхідно, щоб користувачка програма визначала вихід високої швидкості або інше значення призначення даних відповідно до певної таблиці, використовуйте інструкцію порівняння таблиць DHSP. Наприклад, користувач може призначити SD86 та SD87 (подвійне слово) як операнд вихідної частоти інструкції PLSY, щоб змусити PLSY регулювати вихідну частоту відповідно до результату порівняння таблиць.

3. Ви можете використовувати інструкцію DHSP, коли очікуєте, що програма користувача визначить призначення високошвидкісного виходу або інших даних на основі певної таблиці. Наприклад, SD86 та SD87 (подвійне слово) можуть бути вказані як операнди вихідної частоти інструкції PLSY, щоб налаштувати вихідну частоту PLSY відповідно до результату порівняння таблиць.

- Примітка

1. Інструкція DHSP повинна працювати разом з інструкцією HCNT. DHSP може бути виконана коректно лише тоді, коли високошвидкісний лічильник приводиться в дію за допомогою HCNT.

2. Коли DHSP використовується разом із PLSY, дані, що передаються на SD86 та SD 181, повинні відповідати вимогам до вихідної частоти PLSY. Детальнішу інформацію див. в інструкції до PLSY.

3. Якщо ви бажаєте зупинити порівняння на останньому рядку, вам необхідно встановити дані, що надсилаються до SD86 та SD87 останньою коміркою таблиці порівняння, на 0. У цьому випадку інші інструкції DHST та DHSP є недійсними, проте інструкція DHSP не враховується у загальній кількості інструкцій інших високошвидкісних інструкцій.

4. Результат, порівняний інструкцією DHSP, набуває чинності при надходженні зовнішніх імпульсів високошвидкісного лічильника. Тому інструкція DHSP не виконує жодної дії, навіть якщо значення високошвидкісного лічильника змінено за допомогою інструкції DMOV або MOV.

5. Як і загальні інструкції, декілька інструкцій DHSP (DHSCI, DHSCS, DHSCR, DHST та DHSZ) можна використовувати одночасно, але загальна кількість цих інструкцій не може перевищувати 8. Перші 8 інструкцій виконуються по порядку, а 9-та або наступні інструкції є неефективними і тому не виконуються.

6. Якщо DHSP є дійсною інструкцією в інструкції користувача, DHST не виконується, навпаки, якщо DHST є дійсною інструкцією, DHSP не виконується. Лише одна інструкція (DHST або DHSP) може бути дійсною одночасно в програмі користувача, а інші є недійсними.

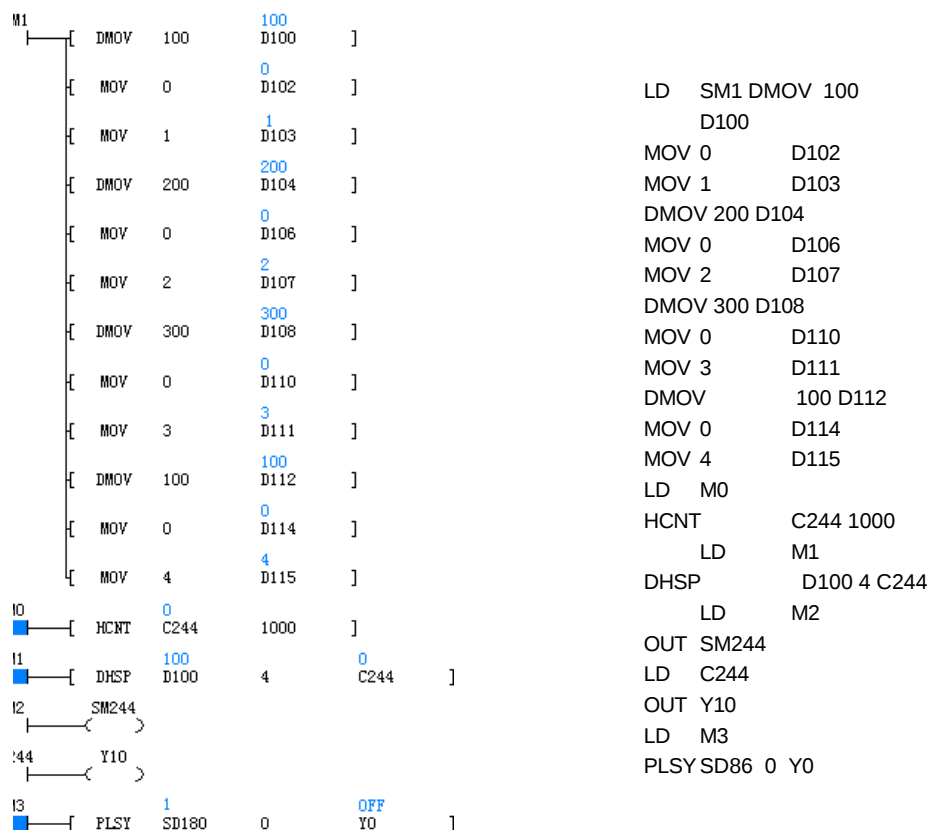
7. Макс. частота, що підтримується високошвидкісним лічильником PLCs. Якщо ви використовуєте інструкцію DHSCS, DHSCI, DHSCR, DHSZ, DHSP або DHST, вона обмежується макс. частотою відгуку та комплексною частотою. Детальнішу інформацію див. у розділі 8 "Посібник з експлуатації функції високошвидкісного входу".

● Екземпляр застосунку

Дані таблиці наведено нижче:

Порівняльні дані		Дані виведено на SD86 та SD87		Операція процесу
MSB	LSB	MSB	LSB	
D100=0	D101=100	D102=0	D103=1	1 ↓
D104=0	D105=200	D106=0	D107=2	2 ↓
D108=0	D109=300	D110=0	D111=3	3 ↓
D112=0	D113=100	D114=0	D115=4	4 ↓ Повернення з 1

LAD показано нижче:



1. Присвоєння початкового значення D100–D115 та створення таблиці для порівняння під час першого циклу сканування користувацької програми.

2. Коли M0 дорівнює ON, C244 виконує підрахунок, коли X0 змінюється з OFF на ON (щодо вхідної частоти X0, зверніться до інструкції для високошвидкісного I/O). Коли C244 змінюється з 999 на 1000, C244 встановлюється, а коли C244 змінюється з 1001 на 1000, C244 скидається. Коли Y10 активується за допомогою C244, виконання Y10 визначається циклом сканування користувацької програми.

3. Коли M1 дорівнює ON, а високошвидкісна інструкція DHSP відповідає вимогам, зазначеним у попередній «Примітці», високошвидкісна інструкція DHSP починається із запису №1 таблиці та переходить до порівняння запису №2 лише після завершення запису №1. До порівняння наступного запису можна перейти лише після завершення попередньої інструкції. Коли порівняння останнього запису завершено, воно знову повертається до порівняння першого запису та встановлює SM83. SD90 вказує номер запису, що порівнюється в даний момент, тоді як SD88 та SD89 вказують дані, що порівнюються в даний момент. Результат порівняння виводиться негайно, незалежно від циклу сканування. Вихідні операнди, отримані після порівняння, поміщаються в SD86 та SD87, на що не впливає цикл сканування. Якщо ви хочете зупинитися на останньому рядку в таблиці порівняння, вам потрібно встановити дані, що надсилаються до SD86 та SD87 фінальною таблицею, на 0.

4. Коли M2 дорівнює ON, SM244 дорівнює ON, і C244 здійснює зворотний відлік. Якщо M2 дорівнює OFF, SM244 дорівнює OFF, і C244 здійснює відлік вгору.

6.10.9 SPD: Інструкція з вимірювання частоти інструкція

LAD:	Застосовна модель										VC1S VC1 VC2 VC3 VC5								
	Прапорець впливу біт																		
SPD		(S1)	(S2)	(D)	Довжина кроку														
	Робота	Тип	Застосовний програмний елемент										Індексація						
S1		BOOL		X															
S2		INT	Кон	стан	т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√
D		INT										D				V		R	√

- Operand description

S1: Вхідна точка. Діапазон налаштування: X0–X7

S2: Одиниця часу для виявлення вхідної точки. Одиниця: мс. Операнд S2>0

D: Блок зберігання даних імпульсного детектування. Коли значення лічильника перевищує 65535, виконується автоматичне переповнення.

- Опис функції

Detecting the number of input pulses of X0–X7 within the specified time (ms), and storing the obtained result in the designated soft element unit.

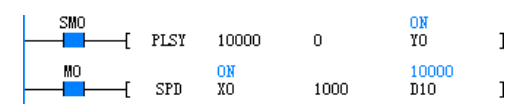
- Примітка

1. Існує апаратний конфлікт між SPD та HCNT, зовнішнім входом переривання та захопленням імпульсів. Для отримання деталей зверніться до розділу 8 "Інструкція з експлуатації функції високошвидкісного входу" для використання.

2. For VC1, the input points of SPD are within the range of X0 to X7.

3. Максимальна частота імпульсного входу SPD становить 10kHz, і може виникнути помилка виявлення, якщо вхідна частота перевищує 10kHz.

- Екземпляр застосунку



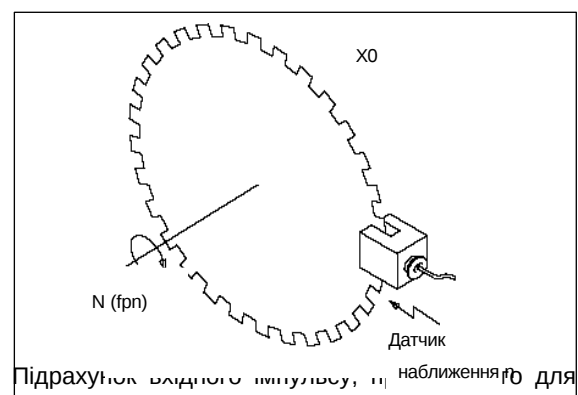
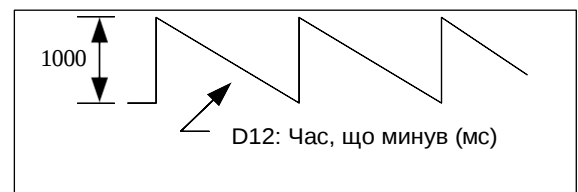
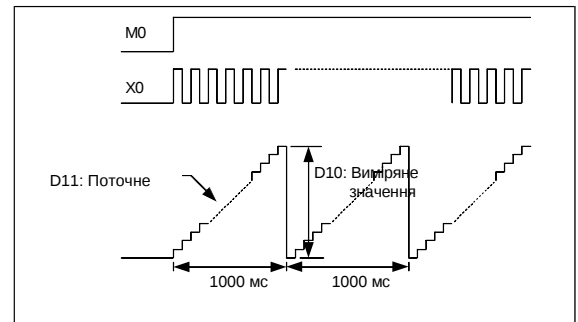
LD SMO

PLSY 10000 0 Y0 LD

MO

SPD X0 1000 D10

Часову послідовність роботи програми наведено нижче:



1. Підрахунок вхідного імпульсу, що наближення до для X0 протягом 1000ms, та збереження результату підрахунку в комірку пам'яті D10, коли M0 є ON, де D11 — це поточне значення підрахунку протягом 1000ms, а D12 — це час, що минув протягом 1000ms.
2. Дані D10 пропорційні швидкості обертання на наведеній вище діаграмі.
3. Підрахунок кожного разу, коли X0 змінюється з OFF на ON, і збереження значень підрахунку в D10 при кожному 1000ms.

6.10.10 PLSY: Високошвидкісний імпульсний вихід інструкція

LAD:	Застосовна модель										VC1S VC1 VC2 VC3							
	Прапорець впливу біт																	
PLSY	(S1)	(S2)	(D)	Довжина кроку														
	Робота d	Тип	Застосовний програмний елемент										Індексація					
S1		DINT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√	
S2		DINT	Постійна	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	√	
D		BOOL			Y													

- Операнд опис

S1: Задана частота (Гц)

Коли **S1** не входить у діапазон налаштування, система повідомляє, що операнд інструкції є неприпустимим, і жоден системний апаратний ресурс не займається. Коли вміст **S1** змінюється під час виконання інструкції, вихідна частота змінюється відповідно.

S2: Кількість згенерованих імпульсів (PLS).

Якщо заданий операнд знаходиться поза межами діапазону налаштування, система повідомляє, що операнд інструкції є недопустимим, імпульс не виводиться, і жоден апаратний ресурс системи не займається. Коли **S2** дорівнює 0, імпульс завжди виводиться, коли інструкція є дійсною.

Якщо вміст **S2** змінюється під час виконання інструкції, зміна набуває чинності в наступному циклі роботи привода.

D: Вихід високошвидкісних імпульсів точка.

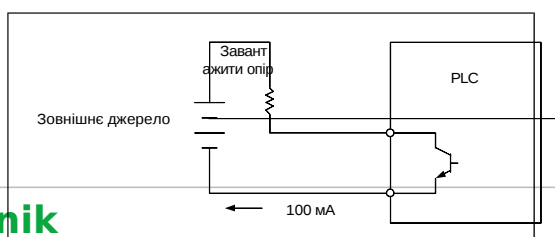
- Опис функції

Генерація заданої кількості високошвидкісних імпульсів виходу відповідно до частоти, вказаної в інструкції. Для виведення високошвидкісних імпульсів струм навантаження на вихідному транзисторі привода PLCs має бути достатньо великим, але він не повинен перевищувати номінальний струм навантаження.

- Примітка

1. PLCs потребує використання режиму транзисторного виходу. 2. Коли PLCsare виконує високошвидкісний імпульсний вихід, необхідно використовувати струм навантаження, вказаний для вихідного транзистора PLC, описаного нижче.

3. The output circuit (transistor) for PLSY, PWM and PLSR is shown as follows.



246

інструкції, вимагається, щоб вихідний порт транзистора був підключений до відповідного навантаження. Коли вихідна форма сигналу не відповідає операнду інструкції, доцільно збільшити струм навантаження транзистора (навантаження транзистора $\leq 100\text{mA}$).

5. Коли високошвидкісна інструкція виконується ефективно (включаючи завершення виводу), інші операції на тому самому порту є недейсними. Тільки коли інструкція високошвидкісного імпульсного виводу є недейсною, інші інструкції можуть керувати цим портом.

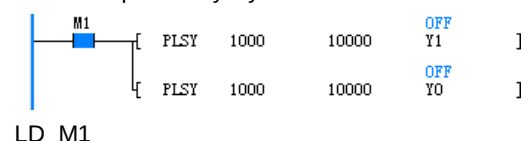
6. Використання декількох інструкцій PLSR дозволяє отримати окремий високошвидкісний імпульсний вихід на точках високошвидкісного виходу, або використання інструкції PWM (чи PLSR) дозволяє отримати окремий високошвидкісний імпульсний вихід на різних вихідних точках.

7. Коли декілька інструкцій PWM, PLSY або PLSR діють на один і той самий порт, перша дійсна інструкція керує вихідним станом порту, а наступні дійсні інструкції не впливають на стан вихідних точок.

8. Подібно до інших високошвидкісних інструкцій (DHSCS, DHSCR, DHSZ, DHSP, DHST та

HCNT), інструкція PLSY повинна відповідати вимогам до високошвидкісного виходу та високошвидкісного імпульсного виходу в системі.

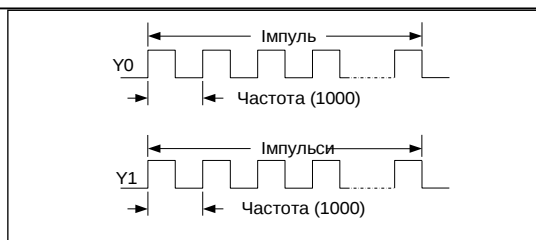
- Екземпляр застосунку



PLSY 1000 10000 Y1

PLSY 1000 10000 Y0

4. При великому струмі навантаження OFF час роботи транзистора є відносно тривалим. Під час виконання PWM, PLSY та PLSR



1. Коли M1 має значення ON, 10000 імпульсів виводяться через порти Y0 та Y1 з частотою 1000 Гц, і вивід імпульсів зупиняється після того, як 10000 імпульсів було виведено. Коли M1 змінюється з OFF на ON, починається наступний цикл виводу. Коли M1 має значення OFF, вивід порту становить OFF.

2. Робочий цикл імпульсу становить 50% ON та 50% OFF. Вихідне керування не залежить від циклу сканування та обробляється перериваннями. При високочастотному виході робочий цикл виходу на портах Y залежить від навантаження. Форми сигналів, отримані з вихідних клем, залежать від вихідного навантаження користувача. Коли навантаження не може перевищувати номінальний струм навантаження, чим менше навантаження, тим ближче вихідна форма сигналу до заданого операнда.

3. SM270 використовується для керування виходом Y0, тоді як SM290 використовується для керування виходом Y1, і коли SM290 дорівнює 1, вихідний імпульс Y1 вимкнено.

4. SM271 та SM291 вказують на вихідні прапорці Y0 та Y1 відповідно. Прапорець скидається, коли вихід завершено або M0 має значення OFF.

5. SD160 вказує на MSB кількості імпульсів виходу Y0 в інструкціях PLSY та PLSR.

SD161 вказує на LSB кількості імпульсів виходу Y0 в інструкціях PLSY та PLSR.

SD180 вказує на MSB кількості імпульсів виходу Y1 в інструкціях PLSY та PLSR.

SD181 вказує на LSB кількості імпульсів виходу Y1 в інструкціях PLSY та PLSR.

6. SD160–SD180 можна змінити через "DMOV x x x SD160" або "MOV x x x SD180", або через моніторинг.

7. Якщо ви бажаєте використовувати кількість вхідних імпульсів для керування частотою вихідних імпульсів PLSY, зверніться до інструкції DHSP для отримання деталей

6.10.11 PLSR: Інструкція для виходу імпульсного лічильника з прискоренням/уповільненням

LAD:	PLSR				(S1) (S2) (S3) (D)				Застосовна модель									
	Прапорець впливу біт				VC1S VC1 VC2 VC3													
PLSR				(S1) (S2) (S3) (D)				Довжина кроку										
Робота d	Тип	Застосовний програмний елемент														Індикація		
S1	INT	Складається немає	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓			
S2	DINT	Складається немає	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓			
S3	INT	Складається немає	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓			
D1	BOOL		Y															

● Операнд опис

S1: Максимальна частота (Гц). Діапазон налаштування: 10–20000 (Гц). Якщо операнд, вказаний опосередковано, більший за 20000, він обробляється як 20000, якщо він менший за 10, він обробляється як 10. У такому разі система видає помилку, вказуючи, що значення операнд інструкції є недійсним, і за замовчуванням виводиться високошвидкісний імпульс.

S2: Загальна кількість вихідних імпульсів (PLS). Діапазон налаштування: 110–2147483647. Якщо встановлений операнд знаходиться поза межами діапазону, система повідомляє про помилку,

вказуючи, що значення операнда інструкції є недійсним, імпульси не виводяться, і жодні апаратні ресурси, що відповідають цій інструкції, не займаються.

S3: Час розгону/гальмування (мс)

Якщо $S1 \times S3 < 100000$, $S3$ вважається рівним $100000/S1$, система повідомляє про помилку параметра інструкції PLSR, а час розгону/гальмування є невизначеним.

Якщо $S1 \times S3 > S2 \times 909$, $S3$ вважається як $S2 \times 909/S1$, система повідомляє про помилку параметра інструкції PLSR, а час розгону/гальмування є невизначеним.

Зміна швидкості рівномірно розподіляється на 10 кроків під час обробки прискорення/уповільнення, і кожен крок становить $S1/10$.

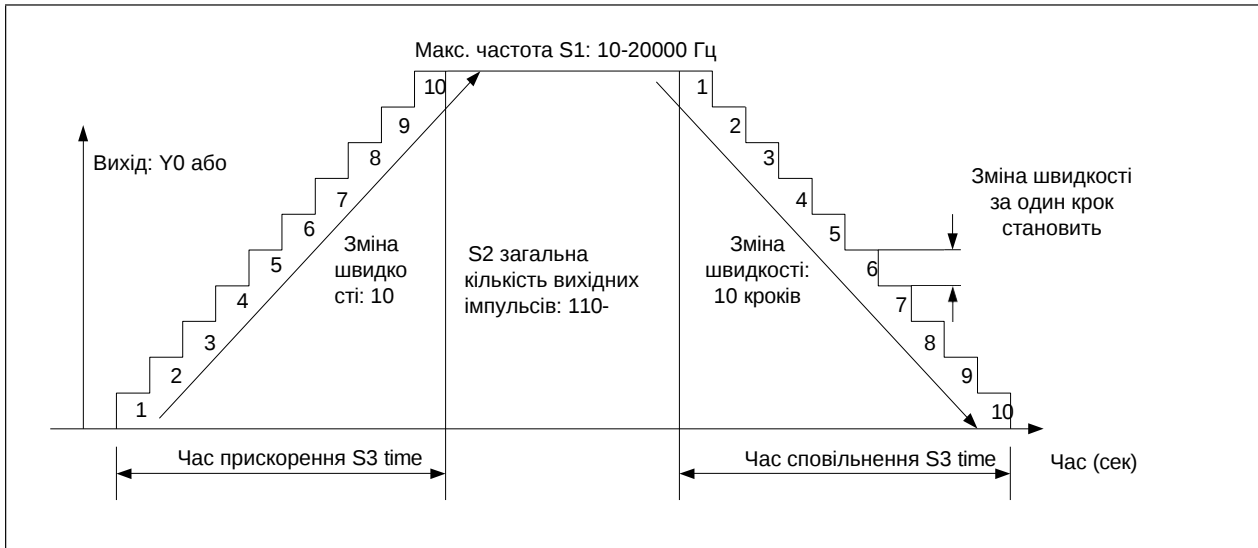
D: Вихід високошвидкісних імпульсів точка.

- Опис функції

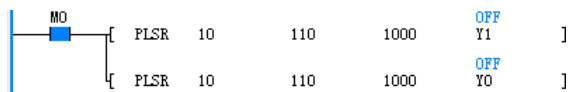
Інструкція PLSR — це інструкція високошвидкісного імпульсного виходу з функцією прискорення/сповільнення для передачі фіксованого розміру. Орієнтуючись на вказану макс. частоту, імпульсний вихід прискорюється рівномірно та

після того, як кількість вихідних імпульсів досягне заданого значення, вихідний імпульсний сигнал рівномірно сповільнюється.

Процес роботи показано на наступному рисунку:



● Примітка



1. Вихідна частота цієї інструкції становить 10–20000Hz. Якщо зміна швидкості під час розгону/гальмування перевищує цей діапазон, операнд автоматично коригується в межах діапазону. На цю інструкцію не впливає цикл сканування.

2. PLCs необхідно використовувати транзисторний вихід. Під час високошвидкісного імпульсного виходу, вихідний транзистор підключається відповідно до вказаного струму навантаження. Форми сигналів, отримані з вихідних клем, залежать від вихідного навантаження користувача. Коли навантаження не може перевищувати номінальний струм навантаження, чим менше навантаження, тим ближче форма вихідного сигналу до заданого операнда.

3. Коли високошвидкісна інструкція виконується ефективно (включаючи завершення виводу), інші операції на тому самому порту є недійсними. Тільки коли інструкція високошвидкісного імпульсного виводу є недійсною, інші інструкції можуть працювати з цим портом.

4. Використання двох інструкцій PLSR дозволяє отримати окремий вихід високошвидкісних імпульсів на вихідних точках, або використання інструкції PWM (чи PLSR) дозволяє отримати окремий вихід високошвидкісних імпульсів на різних вихідних точках.

5. Коли декілька інструкцій PWM, PLSY або PLSR працюють з одним і тим самим портом, перша дійсна інструкція керує станом виходу порту, а наступні дійсні інструкції не впливають на стан вихідних точок.

6. Подібно до інших високошвидкісних інструкцій

(DHSCS, DHSCR, DHSZ, DHSP, DHST, та HCNT), інструкція PLSR повинна відповідати вимогам до високошвидкісного входу та високошвидкісного імпульсного виходу в системі.

● Екземпляр застосування

```
LD M0
PLSR 10 110 1000 Y1
PLSR 10 110 1000 Y0
```

1. Коли M0 дорівнює ON, імпульси виводяться через порти Y0 та Y1 із заданою частотою, а вивід імпульсів зупиняється після 110

імпульси виводяться. Коли M0 змінюється з OFF на ON, починається наступний цикл виводу імпульсів. Коли M0 дорівнює OFF, вихід порту дорівнює OFF.

2. Усі операнди не змінюються під час виконання інструкції, вони обробляються відповідно до операнда, який стає дійсним першим. Новий операнд стає дійсним лише тоді, коли M0 змінюється з ON → OFF → ON.

3. SM270 використовується для керування виходом Y0, тоді як SM290 використовується для керування виходом Y1, і коли SM270 або SM290 дорівнює 1, вихідні імпульси, що відповідають вихідним точкам, вимикаються.

4. SM271 та SM291 вказують на вихідні прапорці Y0 та Y1 відповідно. Відповідний SM271 або SM291 має значення OFF, коли вихід завершено або M0 має значення OFF. Відповідний SM271 або SM291 має значення ON, коли вихід увімкнено.

5. SD160–SD180 відповідають наступному:

1) SD160: MSB кількості імпульсів виходу Y0 у PLSY та PLSR інструкціях.

2) SD161: LSB кількості імпульсів виходу Y0 у PLSY та PLSR інструкціях.

3) SD180: MSB кількості імпульсів виходу Y1 у PLSY та PLSR інструкціях.

4) SD181: LSB кількості імпульсів виходу Y1 у PLSY та PLSR інструкціях.

6. SD160–SD180 можна змінити за допомогою "DMOV x x x SD160" або "MOV x x x SD180", або через моніторинг.

6.10.12 PLS: Інструкція імпульсного виходу огинаючої сигналу

LAD: 										Застосовна модель		VC1 VC2 VC3					
										Прапорець впливу біт							
IL: PLS (S1) (S2) (D1)										Довжина кроку		7					
Робота d	Тип	Застосовний програмний елемент														Індексація	
S1	INT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	
S2	DINT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
D1	BOOL			Y													

- Операнд опис

S1: Початкова адреса елемента D, визначеного параметром

S2: Кількість вихідних сегментів. Діапазон: 0–255.

D1: Вихід високошвидкісних імпульсів точка.

- Опис функції

1. Використання майстра інструкцій Auto Studio для створення інструкції PLS, яку можна викликати як підпрограму. Коли потік енергії увімкнено, система видає відповідні імпульси згідно з конфігурацією. Ви можете керувати ON або OFF генератора імпульсів, а також встановити частоту та кількість імпульсів.

2. Немає виводу, коли номер сегмента дорівнює 0.

3. Високошвидкісний імпульсний вихід можна вимкнути шляхом встановлення SM270, SM290 та SM310 у значення ON, а інші біти прапорців є спільними з високошвидкісними виходами.

4. Вміст підпрограми PLS_SET згенеровано

за допомогою програмного забезпечення Auto Studio є такими (встановіть п як його D

номер елемента та M як загальна кількість сегментів):

LD SM0

DMOV Частота процедури першого сегмента Dn

DMOV Кількість імпульсів процедури першого сегмента Dn+2

DMOV Частота процедури другого сегмента Dn+4

DMOV Кількість імпульсів процедури другого сегмента Dn+6

DMOV Частота процедури третього сегмента Dn+8

- При мітк а

DMOV Макс. швидкість Dn+

4M MOV Мін. швидкість Dn+

4M+2

MOV Час розгону Dn+4M+3 MOV

Час гальмування Dn+4M+4

1. Рекомендовано використовувати інструкцію PLS, згенеровану майстром інструкцій PTO. Якщо ви пишете інструкцію PLS вручну, зверніть увагу, що кількість імпульсів у кожній процедурі не може бути занадто малою. При заданій швидкості прискорення кількість імпульсів у кожній процедурі повинна бути більшою за мінімальну кількість імпульсів необхідно для частотного перетворення.

2. P вказує на кількість імпульсів у певній процедурі, F_N вказує на частоту сегмента N, $F_{\text{макс.}}$ та $F_{\text{мін.}}$ вказують на макс. швидкість та мін. швидкість,

та T_a
вгор
у

T_v вказіть прискорення та

DMOV Кількість імпульсів третього сегмента

час уповільнення в мс.

1) Коли швидкість процедури N більша за швидкість процедури N-1, кількість імпульсів у процедурі N повинна відповідати наступним умовам:

$$P \geq \frac{(F_N + F_{N-1}) \times (F_N - F_{N-1}) \times T_{copy}}{2000 \times (F_{max} - F_{min})}$$

процедура Dn + 10

.....

DMOV Частота процедури сегмента M Dn + 4M - 4

DMOV Кількість імпульсів M сегмента процедури Dn + 4M - 2

2) Коли швидкість процедури N менша за швидкість процедури N-1, кількість імпульсів у процедурі N повинна відповідати таким умовам:

$$P \geq \frac{(F_N + F_{N-1}) \times (F_N - F_{N-1}) \times T_{min}}{2000 \times (F_{max} - F_{min})}$$

3. Зокрема: зокрема:

1) Коли N=1, частота процедури N - 1

бере $F_{хв}$, а потім $F_{хв}$ встановлюється у вищезгадане формула

2) Коли кількість процедур дорівнює 1, а саме є лише один сегмент, кількість імпульсів повинна відповідати такій умові:

$$P \geq \frac{(F_1 + F_{\min}) \times (F_1 - F_{\min}) \times (T_{up} + T_{down})}{2000 \times (F_{\max} - F_{\min})}$$

3) Кількість імпульсів у останній процедурі повинна відповідати наступній формулі:

$$P \geq \frac{(F_M + F_M) \times (F_M - F_M) \times (T_{вых} + T_{внут})}{2000 \times (F_{\max} - F_{\min})}$$

4) Частота, визначена в кожній процедурі, повинна бути в межах діапазону налаштування макс. або мін. швидкості.

4. PLCs необхідно використовувати транзисторний вихід. Під час високошвидкісного імпульсного виходу транзисторний вихід підключається відповідно до вказаного струму навантаження. Форми сигналів, отримані з вихідних клем, пов'язані

до вихідного навантаження користувача. Коли навантаження не може перевищувати номінальний струм навантаження, чим менше навантаження, тим ближче вихідна форма сигналу до заданого операнда.

5. Коли високошвидкісну інструкцію виконано ефективно (включаючи вихід завершення), інше операції на одному й тому самому порту є неприпустимими.

інструкція високошвидкісного імпульсного виходу є недейсною, чи можуть інші інструкції керувати цим портом.

6. PLSY, PLSR, PLS та інструкції позиціонування можуть виводити високошвидкісні імпульси через порти. Заборонено використовувати ці інструкції для високошвидкісного імпульсного виводу на одному й тому самому порту одночасно.

6.10.13 PWM: Інструкція імпульсного виходу ходу

LAD:	Застосовна модель										VC1S VC1 VC2 VC3												
	Прапорець впливу біт																						
PWM (S1) (S2) (D)										Довжина кроку													
Опера	Тип	Застосовний програмний елемент														Індикація							
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C	T	V	Z	R	✓						
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C	T	V	Z	R	✓						
D	BOOL			Y																			

● Операнд опис

S1: Призначена ширина імпульсу (мс/мкс)

Діапазон налаштування: 0–32767 (мс), якщо **S1** більше за 32767, система повідомляє, що операнд інструкції є недопустимим, і жоден апаратний ресурс системи не займається.

Якщо вміст **S1** змінюється під час виконання інструкції, вихідний імпульс також змінюється. Коли SM272 дорівнює 0, **S1** вимірюється в **ms**; коли SM272 дорівнює 1, **S1** вимірюється в **us**.

S2: Призначений цикл імпульсів (мс)

Діапазон налаштування: 1–32767, якщо заданий операнд не входить у діапазон налаштування, система повідомляє, що операнд інструкції є неприпустимим, імпульс не виводиться, і жоден апаратний ресурс системи не займається.

Якщо вміст **S2** змінюється під час виконання інструкції, вихідний імпульс також змінюється. Коли SM84 дорівнює 0, **S1** вимірюється в **ms**; коли SM84 дорівнює 1, **S1** вимірюється в **us**.

S2 має бути більшим або дорівнювати **S1**, інакше система повідомить про помилку операнда, і жоден системний ресурс не буде зайнятий.

D: Вихід високошвидкісних імпульсів точка.

● Опис функції

Імпульс PWM, ширина виходу якого становить **S1**, а цикл імпульсу є

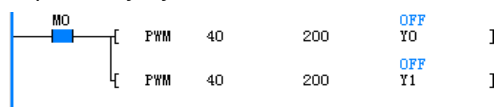
S2 виводиться на порт, визначений **D**. Примітка

1. Коли **S1** дорівнює 0, вихід високошвидкісного порту виводу завжди OFF. Коли **S1=S2**, вихід високошвидкісного порту виводу завжди ON.

2. Форми сигналів, отримані з вихідних клем, залежать від вихідного навантаження користувача. Коли досягається макс. вихідний струм, чим менше навантаження, тим ближчою є вихідна форма сигналу до заданого операнда. Для виведення високошвидкісних імпульсів струм навантаження на вихідних транзисторах PLCs має бути достатньо великим у межах діапазону номінального струму навантаження.
3. Коли високошвидкісна інструкція виконується ефективно (включаючи завершення виводу), інші операції на тому самому порту є недійсними. Тільки коли інструкція високошвидкісного імпульсного виводу є недійсною, інші інструкції можуть керувати цим портом.
4. Використання двох інструкцій PWM дозволяє отримати окремий імпульсний вихід на вихідних портах, або використання інструкції PLSY (чи PLSR) дозволяє отримати окремий високошвидкісний імпульсний вихід на різних вихідних портах.
5. Коли декілька інструкцій PWM, PLSY або PLSR діють на один і той самий порт, перша дійсна інструкція керує вихідним станом порту, а наступні дійсні інструкції не впливають на стан вихідних точок.
6. Подібно до інших високошвидкісних інструкцій (DHSCS, DHSCR, DHSZ, DHSP, DHST та

HCNT), інструкція PWM повинна відповідати вимогам для високошвидкісного I/O у системі.

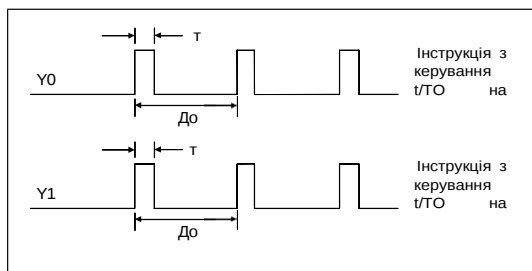
● Екземпляр застосунку



LD M0

PWM 40 200 Y0

PWM 40 200 Y1



Де "t" — це ширина імпульсу, а T0 — цикл імпульсу.

1. Коли M0 дорівнює ON, Y0 та Y1 видають PWM імпульси з шириною 40ms та циклом 200 мс. Коли M0 дорівнює OFF, вихід дорівнює OFF. На стан виходу не впливає цикл сканування.

2. SM270 використовується для вимкнення виходу Y0, тоді як SM290 використовується для вимкнення виходу Y1. Коли SM270 та SM290 знаходяться у стані ON, вихід зупинено.

3. SM271 та SM291 вказують на вихідний прапор Y0 та Y1 відповідно. Коли M0 дорівнює OFF, SM271 та SM291 дорівнюють OFF.

6.11 Інструкції з розрахунку керування

6.11.1 PID: Інструкція функції

LAD:	Застосовна модель					VC1S VC1 VC2 VC3 VC5				
	Прапорець впливу біт									
PID	(S1)	(S2)	(S3)	(D)	Довжина кроку					
Робота	Тип	Застосовний програмний елемент								Індексація
S1	INT									R ✓
S2	INT									R ✓
S3	INT									R ✓
D	INT									R ✓

- Операнд опис

D: Вихід результату операції (MV) під час виконання програми

S1: Встановить цільове значення (SV)

S2: Поточне виміряне значення (PV)

S3: Час вибірки (Ts). Діапазон: 1–32767 (мс). Його необхідно встановити більшим за час обчислення.

S3+1: Слово для налаштування функції дії, аварійного сигналу та верхньої/нижньої межі.

(від'ємна зміна). Діапазон: 0–32767 (якщо біт 1 параметра **S3+1** дорівнює 1).

S3 + 17: Уставка аварійного сигналу варіації виходу PID (позитивна зміна). Діапазон: 0 – 32767 (коли біт 2 та біт 5 **S3 + 1** дорівнюють 1 та 0 відповідно).

Встановить значення верхньої межі виходу. Діапазон: -32768–+32767 (коли біт 2 та біт 5 **S3 + 1** дорівнюють 0 та 1 відповідно).

Біт	Встановить значення та визначення	
	0	1
0	Позитивний зворотний зв'язок	Негативний зворотний зв'язок
1	аварійний сигнал недійсний	Тривога варіації вхідного сигналу дійсна
2	Варіант виходу аварійний сигнал недійсний	Сигнал тривоги відхилення виходу дійсний
3 ~ 4	Зарезервовано	
5	Вихідне значення налаштування значення верхньої/нижньої межі є недейсним	Налаштування верхньої/нижньої межі вихідного значення є дійсним
6 ~ 15	Зарезервовано	

S3+2: Стала фільтрації входу (α). Діапазон: 0–99[%]. Якщо значення дорівнює 0, фільтрація входу відсутня.

S3+3: Пропорційний коефіцієнт підсилення (Kp). Діапазон: 1–32767[%].

S3+4: Час інтегрування (TI). Діапазон: 0–32767 (×100ms). Якщо значення дорівнює 0, воно обробляється як ∞ (без інтегрування).

S3+5: Диференціальний коефіцієнт підсилення (KD). Діапазон: 0–100[%]. Якщо значення дорівнює 0, диференціальний коефіцієнт підсилення відсутній.

S3+6: Диференціальний час (TD). Діапазон: 0–32767 (×10ms). Якщо значення дорівнює 0, воно обробляється як відсутність диференціала.

S3+7–S3+14: Внутрішній регістр даних для PID операції.

S3+15: Значення уставки аварійного сигналу для PID варіації AI-входу (позитивна зміна). Діапазон: 0–32767 (якщо біт 1 у **S3+1** дорівнює 1).

S3 + 16: Уставка аварійного сигналу для зміни PID AI-входу

S3 + 18: Значення уставки аварійного сигналу для відхилення виходу PID (від'ємна зміна). Діапазон: 0 – 32767 (коли біт 2 та біт 5 **S3+1** дорівнюють 1) та 0 відповідно).

Встановіть значення нижньої межі виходу. Діапазон: - 32768–+32767 (коли біт2 та біт5 S3 +1 дорівнюють 0 та 1 відповідно).

S3 + 19: PID вихід аварійного сигналу.

- Біт 0: Варіація вхідного сигналу (позитивна зміна) переповнюється.
- Біт 1: варіація входу (від'ємна зміна) переповнюється.
- Біт 2: Переповнення вихідного відхилення (додатна зміна).
- Біт 3: Переповнення варіації виходу (від'ємна зміна).

У якому S3–S3 + bare є обов'язковими операндами, що задаються користувачем, тоді як S3 + 15 – S3 + 19 є необов'язковими операндами, що задаються користувачем. Ви можете задати операнди через майстер інструкцій PID у Auto Studio.

● Опис функції

1. Виконання операції PID, коли потік енергії є дійсним і досягнуто часу вибірки.
2. Кілька PID інструкцій можуть виконуватися одночасно багато разів (без обмеження кількості циклів). Але вам слід зауважити, що номери програмних елементів S1, S2, S3 або D, які використовуються операцією, не можуть бути накладені.
3. Інструкція PID застосовна до підпрограм переривань за таймером, загальних підпрограм та головних програм. За цієї умови необхідно підтвердити одиницю набору операндів та очистити внутрішні дані обробки **S3 + 7** перед виконанням інструкції PID . °

4. The input filtering constant can smooth the changes of measured value.

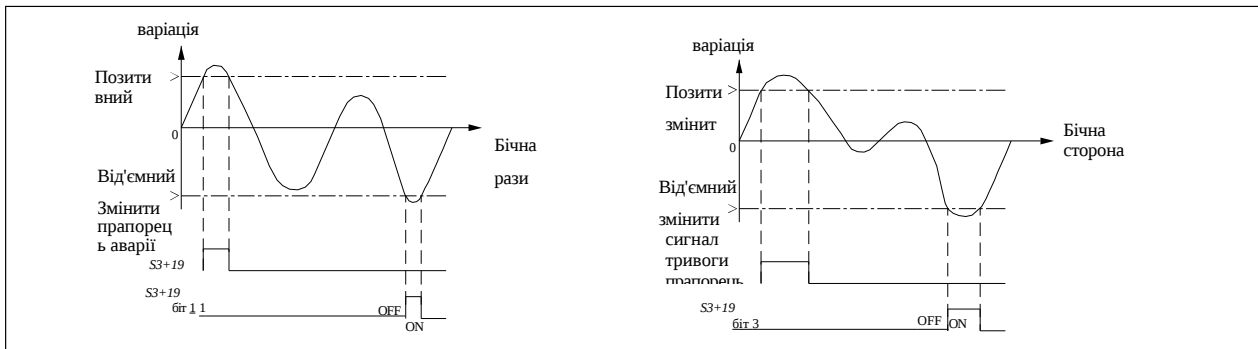
5. Диференціальне підсилення може пом'якшити різку зміну вихідного значення.

6. Action direction: Setting the forward (positive feedback) and reverse actions (adverse feedback) of the system via bit 0 of **S3+1**.

7. Налаштування верхньої/нижньої межі виходу: Коли ви встановлюєте верхню/нижню межу виходу як дійсну (біт 5 та біт 2 **S3+1** є ON та OFF відповідно), інтегральну складову PID можна запобігти від надмірного зростання. Вихідне значення показано нижче:



8. Alarm setting: When you set the output upper/lower limit to be valid (in **S3+1**, bit 1 is ON, bit 2 is ON, and bit 5 is OFF), the PID instruction compares the I/O variation with the preset value in **S3+15-S3+18**. If the current value is larger than the preset value, PID reports an alarm, and the corresponding function bits in **S3+19** are set immediately after executing the PID instruction. In this way, you can monitor the IO variation. The current output values are shown as below:



9. Базовий вираз PID інструкції:

Дія напрямком	PID вираз
Позитивна дія	$= KP \left\{ (EV_n - EV_{n-1}) + \frac{T_s}{T_i} EV_n + D_n \right\}$ $EV_n = PV_{nf-1} - SV$ $= \frac{T_D}{T_s + \alpha_D * T_D} (PV_{nf} + PV_{nf-2} - 2 PV_{nf-1}) + \frac{\alpha_D * T_D}{T_s + \alpha_D * T_D} * D_{n-1}$ $MV_n = \sum \Delta MV$
Зворотна дія	$= KP \left\{ (EV_n - EV_{n-1}) + \frac{T_s}{T_i} EV_n + D_n \right\}$ $EV_n = SV - PV_{nf-1}$ $= \frac{T_D}{T_s + \alpha_D * T_D} (2 PV_{nf-1} - PV_{nf} - PV_{nf-2}) + \frac{\alpha_D * T_D}{T_s + \alpha_D * T_D} * D_{n-1}$ $MV_n = \sum \Delta MV$

Опис символів наведено нижче:

Символ	Опис	Символ	Опис
EV_n	Відхилення вибірки струму	n	Диференціальний струм елемент
EV_{n-1}	Відхилення перед одним циклом	$n-1$	Диференціальний елемент перед одним циклом
SV	Цільове значення	KP	Пропорційний коефіцієнт підсилення
PV_{nf}	Значення вибірки струму (після фільтрації)		Цикл опитування
PV_{nf-1}	Значення вибірки перед одним циклом (після фільтрації)		Інтегральний час
PV_{nf-2}	Значення вибірки перед двома циклами (після фільтрування)		Диференціальний час
ΔMV	варіація		Диференціальне підсилення
MV	Поточне значення величини		

● Екземпляр застосунок

PID ініціалізація, якщо операнди керування однакові, ви можете ініціалізувати операнди лише один раз.

SM1

//Ініціалізація, виконується лише один

MOV	1000	D500	раз //Встановлення цільового значення

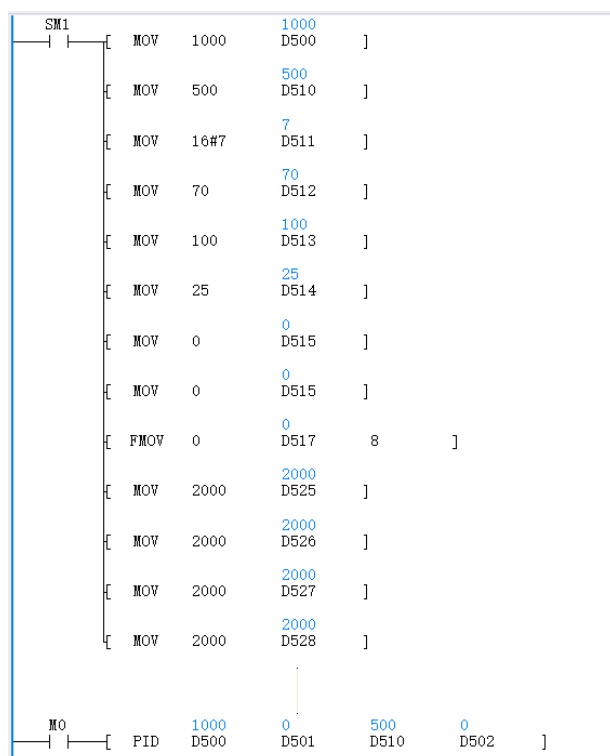
MOV	500	D510	Час дискретизації (Ts). Діапазон: 1–32767 (мс). Він має бути більшим за час //розрахунок часу
MOV	7	D511	//Дія напрямок
MOV	70	D512	Константа фільтрації входу (α). Діапазон: 0–99[%]. Коли значення дорівнює 0, вхідна фільтрація відсутня
MOV	100	D513	Пропорційний коефіцієнт підсилення (Kp). Діапазон: 1–32767[%]
MOV	25	D514	Інтегральний час (TI). Діапазон: 0–32767 (×100ms). Коли він дорівнює 0, він обробляється як //∞ (не інтеграл)
MOV	0	D515	Диференціальне підсилення (KD). Діапазон: 0–100[%]. Коли воно дорівнює 0, диференціальне підсилення відсутнє
MOV	63	D516	Диференціальний час (TD). Діапазон: 0–32767 (×10 мс). Коли він дорівнює 0, це означає відсутність диференціальної складової обробка
FMOV	0	D517	8 //Очищення області для зберігання транзитних даних PID операції
MOV	2000	D525	Значення уставки аварійного сигналу для варіації входу (позитивна зміна). Діапазон: 0–32767
MOV	2000	D526	Значення уставки аварійного сигналу для варіації входу (від'ємна зміна). Діапазон: 0–32767
MOV	2000	D527	Значення уставки аварійного сигналу для зміни виходу (додатна зміна). Діапазон: 0–32767
MOV	2000	D528	Уставка аварійного сигналу варіації виходу (від'ємна зміна). Діапазон: 0–32767

PID виконання інструкції операція

LD M0 Програма операцій PID, що керується
користувачем PID D500 D501 D510 D502 PID інструкція:

PID S1 S2 S3

LAD вищевказаної інструкції показано нижче:



виконується операція PID. Результат операції перетворюється на аналогові сигнали через зовнішні модулі D/A (фактична ситуація може відрізнятися) і подається до керованої системи.

- Примітка

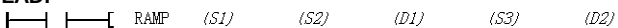
PLCs ініціалізують операнди PID лише під час першого циклу сканування. Коли M0 у стані ON, поточне виміряне значення зчитується із зовнішніх модулів A/D (фактична ситуація може відрізнятися), присвоюється одиниці вимірювання D501, і

1. Для **D** необхідно призначити його в регістр даних за межами діапазону збереження. В іншому випадку його потрібно очистити (LD SM0 MOV 0 D****) під час першої операції.
2. Інструкція PID повинна займати 20 послідовних регістрів даних, починаючи з **S3**.
3. Максимальна похибка часу вибірки TS становить — (один цикл сканування + 1ms)– + (один цикл сканування). Коли значення TSis відносно мале, ефект PID погіршується. Рекомендується використовувати інструкцію PID у перериванні за таймером interrupt.
4. Коли ви встановлюєте верхню/нижню межу виходу PID як дійсну, якщо значення верхньої межі менше за значення нижньої межі, система повідомляє про помилку операнда і не виконує операцію PID.
5. Коли ви встановлюєте сигнал I/O варіації як дійсний, встановлене значення **S3 + 15 – S3 + 18** не може бути від'ємним, інакше система повідомить про помилку операнда і не виконає операцію PID.
6. Коли біт 2 та біт 5 **S3 + 1** встановлені в ON, система визначає, що налаштування є недейсним (це еквівалентно тому, що біт 2 та біт 5 встановлені в OFF), і тому не повідомляє про аварійні сигнали щодо перевищення верхньої межі, нижньої межі або відхилення.
7. Коли задане значення операнда керування PID (блок **S3-S3 + 6**) виходить за межі допустимого діапазону, система повідомляє про помилку операнда і не виконує операцію PID.
8. Коли час вибірки менший або дорівнює одному циклу сканування, якщо під час операції виникає переповнення даних або переповнення результату, сигнал тривоги не генерується, і PID операція продовжується.
9. Усі PID операнди повинні бути ініціалізовані перед початковим виконанням PID інструкції. Якщо операнди залишаються

те саме під час роботи, і відповідні елементи операндів не охоплюються іншими програмами, ви можете ініціалізувати PID операнди лише один раз. Якщо дані в тому

реєстри даних транзиту змінюються під час PID операції, результат операції є некоректним.

6.11.2 RAMP: Інструкція виходу сигналу рампи інструкція

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт							
IL: RAMP (S1) (S2) (D1) (S3) (D2)										Довжина кроку		12					
Робота d	Тип	Застосовний програмний елемент														Індексація	
S1	INT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	
S2	INT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	
D1	INT								D				V		R	✓	
S3	INT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	
D2	BOOL			Y	M	S	LM				C	T					

- Операнд опис

S1: Початкове значення **S2:**

Кінцеве

значення **D1:**

Вихідне

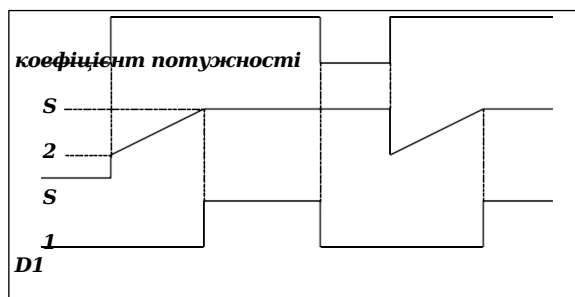
значення

S3: Кількість кроків ($S3 > 0$, інакше система видає помилку операнда і не виконує операцію)

D2: Стан виходу 0

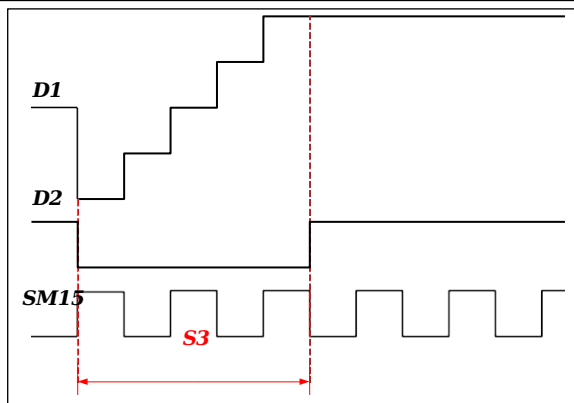
- Опис функції

When a rising edge occurred to the energy flow and keeps ON, each scan cycle, based on the height and mobile scan times of the ramps, determines the incremental quantity and current output value. After reaching **S2**, the output value (**D1**) stays in current state, and the state output position is ON. If a falling edge occurred to the energy flow, the output state (**D2**) is OFF, the output value (**D1**) stays in current state until the rising edge occurs to the energy flow again, the output value (**D1**) is initialized to be the value of **S1**, and continues to generate next ramp operation, as shown



below:

Процес виконання інструкції рампи показано нижче ($S3=5$):



- Примітка

1. Якщо результат не ділиться без залишку при розрахунку довжини кроку, ви можете застосувати метод "округлення".
2. Інструкція генерує дані рампі лише один раз за кожним переднім фронтом сигналу.
3. Коли **S1=S2**, **D1=S2**, та **D2=ON**.
4. Загальна кількість інструкцій RAMP, HACKLE та TRIANGLE у програмі не може перевищувати 100.

- Екземпляр застосунку

Ініціалізація регістра під час першого циклу сканування після увімкнення живлення

```
LD      SM1
MOV     0 D0
MOV     2000 D1
```

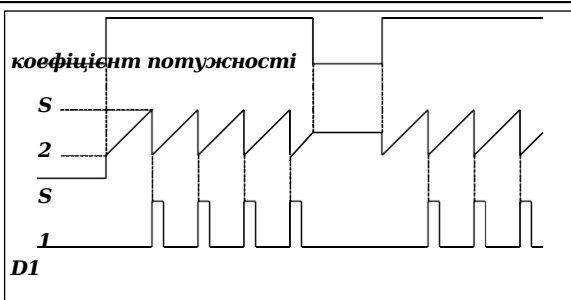
Виконання інструкції функції рампі, коли X0=ON

```
LD      X0
RAMP     D0 D1 D10 1000 M0
```

Надсилення вихідного значення функції рампі до зовнішнього DA модуля, коли X1=ON, для генерування хвильових форм рампі

```
LD      X1
TO      0 6 D10 1
```

LAD вищезазначених інструкцій показано нижче:



The execution process of a sawtooth wave instruction is shown as follows (**S3=5**).



- Примітка
- 1. Якщо результат не ділиться без залишку під час обчислення довжини кроку, можна застосувати метод округлення.

2. Інструкція генерує послідовність даних пилкоподібної хвилі, доки потік енергії є дійсним.

3. Коли **S1=S2**, **D1=S2**, а **D2=ON** (без підрахунку імпульсів).

4. Загальна кількість інструкцій RAMP, HACKLE та TRIANGLE у програмі не може перевищувати 100.

- Екземпляр застосунку

Ініціалізація регістру під час першого циклу сканування після увімкнення живлення

LD SM1

MOV0 D0

MOV2000 D1

Виконання інструкції функції пилкоподібної хвилі, коли X0=ON

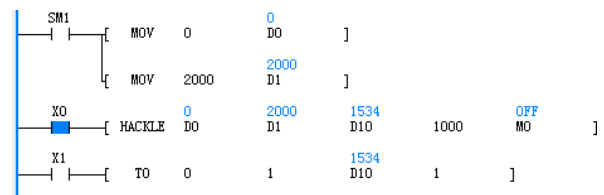
LD X0

HACKLE D0 D1 D10 1000 M0

Надсилання вихідного значення функції рампи на зовнішній DA модуль, коли X1=ON, для генерування пилкоподібних сигналів LD X1

TO 0 1 D10 1

LAD вищезазначених інструкцій наведено нижче.



1. Коли X0=ON, D10 (у першому циклі, D10=D0=0) збільшується на 2 (2000/1000) у кожному циклі сканування. Коли D10=D1=2000, M0=ON. У наступному циклі сканування, якщо X0 залишається ON, D10=D0=0, а M0=OFF, починається наступний процес пилкоподібної хвилі. Якщо потік енергії зникає під час роботи, вихідний стан **D2** дорівнює OFF, вихідне значення **D1** зберігає свій поточний стан до надходження наступного переднього фронту, вихідне значення **D1** ініціалізується як S1, і починається новий процес пилкоподібної хвилі.

2. Ви можете використовувати зовнішній спеціальний модуль для перетворення даних в аналогову форму сигналу.

6.11.4 TRIANGLE: Інструкція виходу сигналу трикутної хвилі інструкція

LAD:	Застосовна модель									VC1S VC1 VC2 VC3 VC5											
	Прапорець впливу біт																				
IL: TRIANGLE (S1)	(S2)	(D1)	(S3)	(D2)	Довжина кроку																
Робота d	Тип	Застосовний програмний елемент															Індексація				
S1	INT	Констант	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C	T	V	Z	R	√				
S2	INT	Постійна	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C	T	V	Z	R	√				
D1	INT									D				V		R	√				
S3	INT	Постійна	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C	T	V	Z	R	√				
D2	BOOL			Y	M	S	LM					C	T								

- Операнд опис

S1: Початкове

значення **S2:**

Кінцеве

значення **D1:**

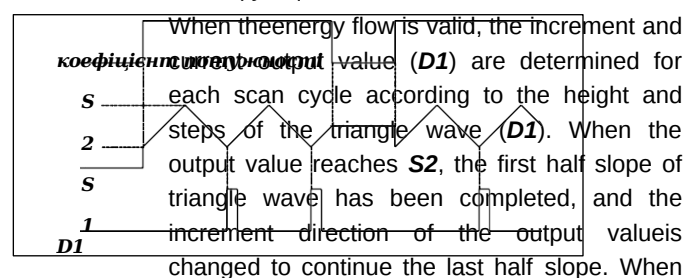
Вихідне

значення

S3: Кількість кроків (**S3**>0, інакше система повідомляє про помилку операнда і не виконує операцію)

D2: Стан виходу state

- Опис функції

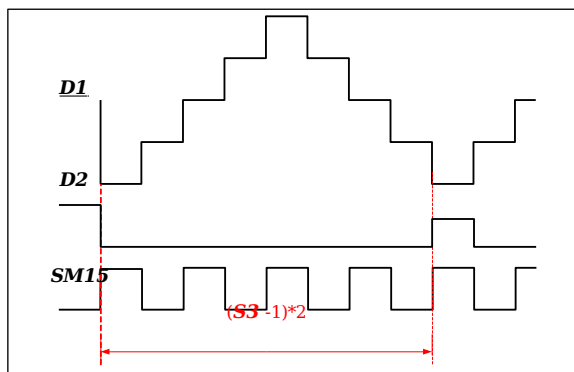


the output value (**D1**) reaches **S1** again,

Біт виходу стану (**D2**) встановлюється в ON. У наступному циклі сканування, якщо потік енергії зберігає ON, біт виходу стану (**D2**) встановлюється в OFF, і генерується наступна трикутна хвиля. Під час процесу генерації функції трикутної хвилі, якщо на потоці енергії виникає спадний фронт, вихідний стан (**D2**) дорівнює OFF, а вихідне значення (**D1**) залишається в поточному стані. Коли на потоці енергії знову виникає передній фронт, вихідне значення (**D1**) ініціалізується в **S1** і починається новий процес генерації трикутної хвилі, як показано нижче.

pf: стан потоку енергії

The execution process of a triangle wave instruction is shown as follows ($S3=5$).



- Примітка
 - Якщо результат не ділиться без залишку під час обчислення довжини кроку, можна застосувати метод округлення.
 - Інструкція генерує послідовність даних трикутної хвилі, доки потік енергії є дійсним.
 - Коли $S1=S2$, $D1=S2$, а $D2=ON$ (без підрахунку імпульсів).
 - Цикл трикутної хвилі становить $(S3-1) \times 2$.
 - Загальна кількість інструкцій RAMP, HACKLE та TRIANGLE у програмі не може перевищувати 100.
- Екземпляр застосунку

Ініціалізація регістра під час першого циклу сканування після увімкнення живлення

```
LD    SM1
MOV   0 D0
```

MOV 2000 D1

Виконання інструкції функції трикутної хвилі, коли $X0=ON$

LD X0

TRIANGLE D0 D1 D10 1000 M0

Надсилання вихідного значення функції рампи на зовнішній DA модуль, коли $X1=ON$, для генерування трикутної форми сигналу

LD X1

TO 0 1 D10 1

LAD вищезазначених інструкцій показано нижче:

- Коли $X0=ON$, $D10=0$ (у першому циклі, $D10=D0=0$) збільшується на 2 ($2000/1000$) у кожному циклі сканування. Коли $D10=D1=2000$, половина трикутної хвилі завершена, і $D10$ зменшується на 2 у кожному наступному циклі сканування. Коли $D10=D0=0$, повна трикутна хвиля завершена, і $M0=ON$. У наступному циклі сканування, якщо $X0$ залишається ON , а $M0=OFF$, починається процес наступної трикутної хвилі. Якщо потік енергії зникає, вихідний стан $D2$ дорівнює OFF , вихідне значення $D1$ ініціалізується як $S1$, і починається новий процес трикутної хвилі.
- Ви можете використовувати зовнішній спеціальний модуль для перетворення даних в аналогову форму сигналу.

6.11.5 ABSD: Абсолютне керування кулачком інструкція

LAD:	Придатна модель						VC2 VC3 VC5										
	Прапорець впливу біт						Прапорець нуля, прапорець перенесення та прапорець позики										
	ABSD (S1) (S2) (D) (S3)																
ABSD (S1) (S2) (D) (S3)						Довжина кроку											
Працює	Тип	Застосовний програмний елемент												Показники			
D														нг			
S1	INT	KnX	KnY	KnM	KnS			D		C	T		R	√			
S2	INT									C				√			
D	BOOL		Y	M	S												
S3	INT	Складається немає															

- Операнд опис

S1: Початковий елемент номера для збереження даних таблиці (передній фронт та задній фронт). $n=4$.

S2: Номер лічильника, що використовується для моніторингу поточного значення порівняно з даними таблиці.

D: Виведення номера початкового біта елемента.

S3: Кількість рядків у таблиці та кількість точок вихідних бітових елементів. Діапазон: $1 \leq S3 \leq 64$.

- Опис функції

1. Порівняння n рядків даних таблиці, починаючи з **S1** (що займають n рядків $\times$ 2 точки), з поточним значенням **S2** лічильника, та виконання керування ON/OFF на послідовному n -точковому **D** виході.

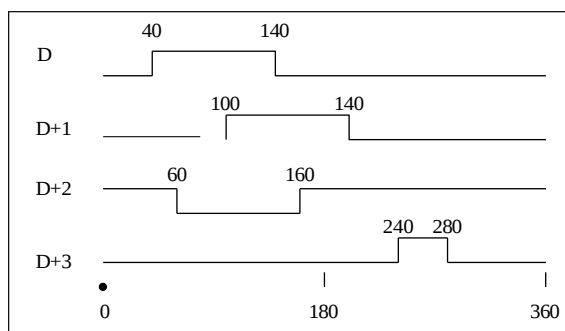
2. Кожну точку наростання/спадання можна змінити шляхом перезапису даних **S1–S1** $+n\times 2$.

3. Спочатку запишіть дані, наведені нижче, у $S1-S1+2n$

+1:

Точка зростання	Приклад значення	Точка падіння	Значення прикладу	Вихід
$S1$	40	$S1+1$	140	D
$S1+2$	100	$S1+3$	200	$D+1$
$S1+4$	160	$S1+5$	60	$D+2$
$S1+6$	240	$S1+7$	280	$D+3$
...	...	...	...	...
$S1+2n$	...	$S1+2n+1$	...	$D+n-1$

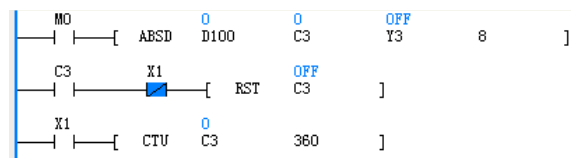
4. After the instruction input is ON, the n point starting from D also changes as follows.



● Екземпляр застосунку

Виведення ON/OFF шляхом одноразового обертання платформи (0–

360 градусів, сигнал кута повороту, що вказує 1 градус на імпульс).

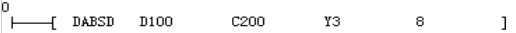


Серед них M0 — це вхід потоку енергії, а X1 — сигнал кута повороту, що вказує 1 градус на імпульс.

● Примітка

1. Коли номер біта бітових логічних елементів призначено в $S1$, $k=4$.
2. Коли номер лічильника призначено в $S2$, для $S2$ необхідно призначити C0–C199.
3. Ця інструкція залежить від циклу сканування.

6.11.6 DABSD: Інструкція з абсолютного керування кулачком подвійного слова instruction

LAD:	Придатна модель										VC2	VC3	VC5				
											Прапорець впливу біт				Прапорець нуля, прапорець перенесення та прапорець позики		
IL: DABSD (S1) (S2) (D) (S3)	Довжина кроку 11																
Працює	Тип	Застосовний програмний елемент										Показує					
D												шки					
S1	DINT	KnX		KnY	KnM	KnS			D		C	T		R	√		
S2	DINT										C				√		
D	BOOL			Y	M	S											
S3	INT	Складається немає															

- Операнд опис

S1: Початковий елемент номера для збереження даних таблиці (передній фронт та задній фронт). $n = 8$.

S2: Номер лічильника, що використовується для моніторингу поточного значення порівняно з даними таблиці. Діапазон: C200–C255 та C256–C263.

D: Виведення номера початкового біта елемента.

S3: Кількість рядків у таблиці та кількість точок вихідних бітових елементів. Діапазон: $1 \leq S3 \leq 64$.

- Опис функції

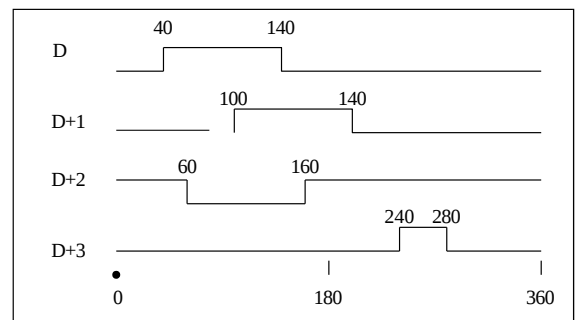
1. Порівняння n рядків даних таблиці, починаючи з **S1** (що займають n рядків×4 точки), з поточним значенням **S2** лічильника, та виконання керування ON/OFF на послідовному n -точковому виході **D**.

2. Кожну точку наростання/спадання можна змінити шляхом перезапису даних $[S1+1, S1] - [S1+(n \times 2)+3, S1+(n \times 2)+2]$.

3. По-перше, використовуючи інструкцію передачі, запишіть дані, наведені нижче, у $[S1, S1+1] - [S1, S1+1]+4n+3$.

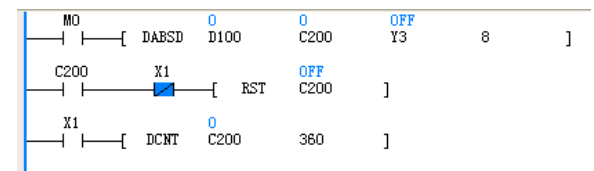
Точка зростання	Значення прикладу	Десяткова крапка	Значення прикладу	Вихід
$[S1+1, S1]$	40	$[S1+3, S1+2]$	140	D
$[S1+5, S1+4]$	100	$[S1+7, S1+6]$	200	D+1
$[S1+9, S1+8]$	160	$[S1+11, S1+10]$	60	D+2
$[S1+13, S1+12]$	240	$[S1+15, S1+14]$	280	D+3
.....				
$[S1+4n+1, S1+4n]$		$[S1+4n+3, S1+4n+2]$		D+n-1

4. Після введення інструкції ON, n -та точка, що починається з **D**, також зміниться, як показано нижче.



- Екземпляр застосунку

Виведення ON/OFF шляхом одноразового обертання платформи (0–360 градусів, сигнал кута повороту, що вказує 1 градус на імпульс).



Серед них **M0** — це вхід потоку енергії, а **X1** — сигнал кута повороту, що вказує 1 градус на імпульс.

Серед них **M0** — це вхід інструкції, а **X1** — сигнал кута повороту, що вказує 1 градус на імпульс. У цей час значення за замовчуванням SM200 дорівнює OFF, інструкція DCNT інкрементується.

- Примітка

1. Коли номер біта для бітових програмних елементів призначено в **S1**, $n=8$.

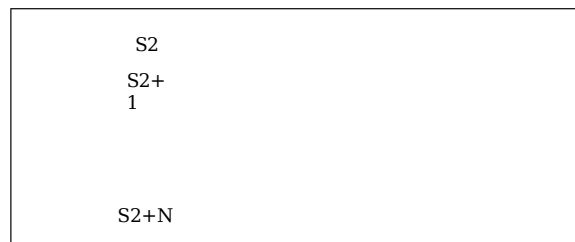
2. Коли номер лічильника визначено в **S2**, C200–C255 необхідно призначити для **S2**.

3. Ця інструкція залежить від циклу сканування.

Контрольні суми не зберігаються. У процесі передачі необхідний початковий символ, кінцевий символ та контрольний біт автоматично додаються до даних.

2. You do not need to set the length for the data to be transmitted. The system automatically transmits the data according to the internally specified length based on the instruction, as shown below:

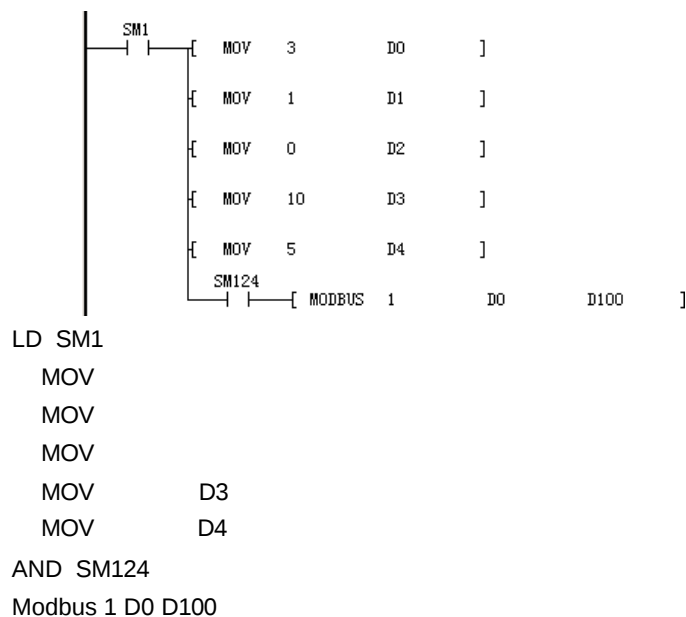
Адреса підпорядкованого пристрою
Функціональний код
Дані 1
... ..
Дані N



3. Дані отримуються через Modbus. Незалежно від того, чи встановите ви режим RTU чи ASCII, отримані дані зберігаються в

~~RTUformat. Тобто, коли ви встановлюєте режим на ASCII,~~
система автоматично перетворює дані у шістнадцятковий формат, видаляє початкові та кінцеві символи, і зберігає їх у зоні даних, починаючи з **S3**.
4. Надіслані та отримані дані зберігаються в LSB елемента слова, а MSB не використовуються.

- Екземпляр застосунку

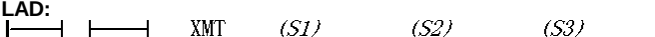


- Збереження даних, надісланих через інструкцію Modbus, в елемент, починаючи з D0.
- Збереження отриманих даних в елементах, починаючи з D100.
- Після отримання даних через Modbus система виконує перевірку CRC, перевірку адреси та перевірку коду функції. Якщо виникає будь-яка помилка, встановлюється прапорець помилки (SM136), а деталі помилки записуються у спеціальний регістр SD139.
- Прапорці очікування послідовного порту SM114 та SM124 також можуть використовуватися в Modbus для індикації стану зв'язку Modbus. Таблиця кодів помилок зв'язку Modbus наведена нижче.

Аномальний код	Значення коду несправності
0x01	Неприпустимий код функції
0x02	Неприпустима адреса регістра
0x03	Неправильна кількість даних
0x10	Тайм-аут зв'язку. зв'язок час перевищує встановлений ліміт часу зв'язку ліміт
0x11	Помилка під час отримання даних кадру
0x12	Помилка операнда, операнд (режим ведучий/ведений) налаштування помилки
0x13	Помилка виникає, коли номер локальної станції є таким такий самий, як встановлений інструкцією.

Щодо деталей про екземпляр застосунку зверніться до розділу 10 "Посібник з комунікаційних функцій".

6.12.2 XMT: Інструкція передавання через вільний порт instruction

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт							
IL: XMT (S1) (S2) (S3)										Довжина кроку		7					
Робота d	Тип	Застосовний програмний елемент														Індекс ація	
S1	INT	Констант а															
S2	INT	D	V												R		
S3	INT	Констант а	KnX	KnY	KnM	KnS	KnLM		D	SD	C	T	V	Z	R		

- Операнд опис

S1: Призначений канал зв'язку. Для VC1 діапазон значень становить 1 та 2.

S2: Початкова адреса даних, що мають бути надіслані

S3: Кількість байтів для надсилання

- Опис функції

Коли живлення увімкнено і умови зв'язку виконані, дані надсилаються через призначений канал та адресу.

- Примітка

- Розмір кадру зв'язку: Розмір кадру зв'язку:

Кінцевий символ переданого кадру не може перевищувати D7999 або V63, залежно від вибраного типу елемента (D або V).

- У разі вимкнення привід зупиняється.

- Спеціальний регістр

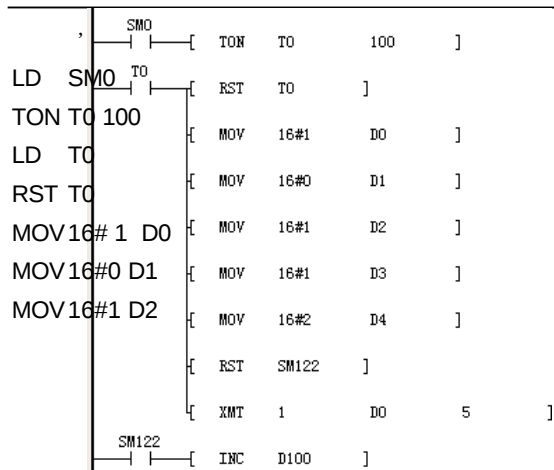
1. SM110/SM120: Прапорець дозволу передачі. Цей біт встановлюється при використанні інструкції ХМТ, скидається після завершення передачі та встановлюється у початковий стан, коли поточна передача переривається.

2. SM112/SM122: Прапорець завершення передачі. Коли визначається, що передачу завершено, встановлюється прапорець завершення передачі.

3. SM114/SM124: Прапор очікування. Встановлюється, коли послідовний порт не має завдань для обміну даними. Може використовуватися як контрольний біт для зв'язку.

4. Детальну інформацію про екземпляр застосунку див. у розділі 10 "Посібник з функцій зв'язку".

- Екземпляр застосунку



MOV 16#1 D3

MOV 16#2 D4

RST SM122

XMT 1 D0 5 LD

SM122

INC D100

Процедура полягає у відправленні одного кадру даних з інтервалом 10s. Через послідовний порт 1 передаються такі дані:

1. Спочатку встановіть порт 1 у системному блоці як вільний порт, а також налаштуйте швидкість передачі даних, перевірку парності, біт даних та стоп біт.
2. Запис даних для відправлення у буферну область передачі. Для VC2, лише молодші байти елемента слова надсилаються.
3. Очищення прапорця завершення передачі (SM122) перед відправленням даних.
4. Встановлення прапорця завершення передачі (SM122) після виконання передачі.

6.12.3 RCV: Приймання інструкції через вільний порт інструкція

LAD:	Застосовна модель										VC1S	VC1	VC2	VC3	VC5		
	Прапорець впливу біт																
IL: RCV (S1)	(D)	(S2)	Довжина кроку														
Робота	Тип	Застосовний програмний елемент														Показник	
S1	INT	Складається немає															
D	INT	D V													R		
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM			DSD	C	T	V	Z	R		

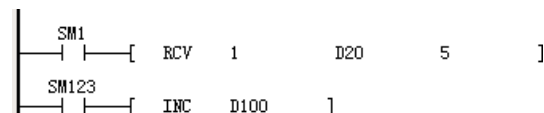
S2: Макс. кількість отриманих байтів

- Опис функції

- Примітка

1. Розмір кадру зв'язку: кінцевий символ отриманого кадру не може перевищувати D7999 або V63, залежно від вибраного типу елемента (D або V).
2. У разі вимкнення приймання зупиняється.
3. Діапазон значень **S1**: 0, 1 та 2.

- Екземпляр застосунку



LD SM1

RCV 1 D20 5

LD SM123

INC D100

1. Коли потік енергії проводиться, інструкція RCV є дійсною безперервно. Якщо ви бажаєте отримати дані лише один раз, ви можете використати передній фронт або спеціальний регістр, який є дійсним лише один раз, наприклад SM1, як вхід потоку енергії.

2. Детальну інформацію щодо екземпляра застосунку див. у розділі 10 "Посібник з функцій зв'язку".

- Спеціальні реєстри

SM111 (SM121):Отримання прапорця дозволу. Цей біт встановлюється, коли використовується інструкція RCV, і скидається після того, як

SM113 (SM123): Отримання прапорця завершення.
Коли отримання завершено, встановлюється прапорець завершення отримання.

SM114 (SM124): Прапорець очікування. Цей прапорець встановлюється, коли немає завдань обміну даними через послідовний порт, і його можна використовувати як контрольний біт для обміну даними.

SD111 (SD121): Початковий символ, який можна встановити у системному блоці.

SD112 (SD122): Кінцевий символ, який можна встановити в системному блоці.

SD113 (SD123): Час очікування між символами, а саме макс. інтервал між отриманням двох символів, який можна встановити в системному блоці.

SD114 (SD124): Час очікування кадру, а саме час від моменту передачі потоку енергії до моменту отримання, який можна встановити в системному блоці. SD115 (SD125): Код інформації про завершення отримання.

Визначення бітів даних наведено нижче:

Прапо рець для завер шення отрим ання корист уваче м даних	Прапо рець для отрим ання вказ аного кінцев ого слова	Прапо рець для отрима ння ма ксима льного кільк ості симво лів привод и	Прапо р тайм- ауту між символ ами	(Типор озмір) отрима ння прапор ця тайм- ауту	Поми лка пере вірки парн ості прап ор	Зареє строван о (Вам не потріб о брати їх до уваги).
Біт 0	Біт 1	Біт 2	Біт 3	Біт 4	Біт 5	Біт 6– 15

SD116 (SD126): Символи, що отримані на даний

момент SD117 (SD127): Загальна кількість символів, отриманих на даний момент.

6.12.4 MODRW: Інструкція з читання/запису Modbus інструкція

[illegible]

S2	INT	Складається немає	D	V										R	✓
S3	INT	Складається немає	D	V										R	✓
S4	INT	Складається немає	D	V										R	✓
S5	INT	Складається немає	D	V										R	✓
D	INT		D											R	✓

Операнд-опис

S1: Призначений канал зв'язку, для VC1 діапазон значень становить 0, 1 та 2

S2: Адреса (Діапазон налаштування адреси підпорядкованого пристрою: 1–247, а широкомовна адреса застосовується до елемента запису).

S3: Код функції. VC1S/1L підтримує 01 (читання котушки), 02 (читання дискретного входу), 03 (читання регістру), 04 (читання вхідного регістру), 05 (запис однієї котушки), 06 (запис одного регістру), 15 (запис кількох котушок), 16 (запис кількох регістрів) VC2/3/5 підтримують 01 (читання котушки), 03 (читання регістру), 15 (запис кількох котушок) та 16 (запис кількох регістрів).

S4: Початкова адреса елементів привода, що підлягають зчитуванню/запису. **S5:** Кількість елементів, що підлягають зчитуванню/запису. Кількість елементів VC1S/1L, що підлягають зчитуванню/запису, обмежена максимальною довжиною кадру (256) RTU, як показано нижче.

Функційний код	Ім'я	Номер елементів в елементі	Номер необхідних D елементів
01	Зчитати котушки	1–2000	(S5 + 15)/16
02	Зчитати дискретний вхід	1–2000	(S5 + 15)/16
03	Зчитати регістри	1–125	S5
04	Прочитайте вхідні дані регістри	1–125	S5
05	однорядковий котушка	0(фіксоване)	1
06	однорядковий зареєструвати	0(фіксоване)	1
15	декілька котушки	1–1968	(S5 + 15)/16
16	декілька регістри	1–123	S5

*Значення 05, 06 та S5 має бути 0

Кількість слівних та бітових елементів не може перевищувати 16 відповідно, і всі бітові елементи зберігаються як одне слово.

D1: Адреса зберігання елементів для зчитування/запису. Детальну інформацію про кількість елементів, необхідних для VC1, див. у таблиці вище.

- Опис функції
Надсилання повідомлення та отримання повернутих даних, коли потік енергії є дійсним.
- Примітка
 - Кількість елементів не може перевищувати 16.
 - Зчитування максимум 16 бітових елементів. Бітові елементи з малими адресами зберігаються в LSB, і один байт зберігає 16 бітів.
 - Повернутий код помилки збігається з інструкцією Modbus.

Екземпляр застосунку

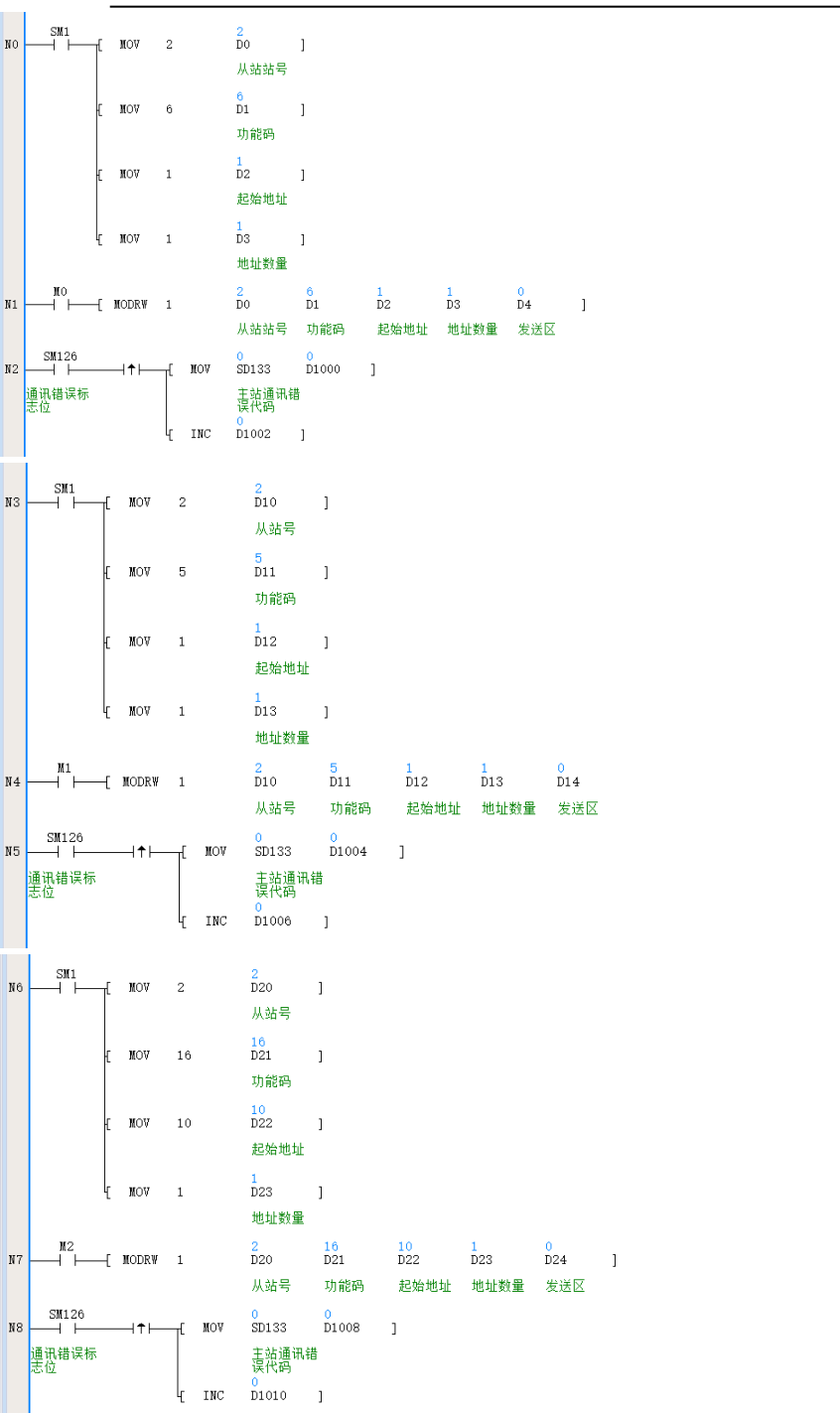
* Наведений нижче екземпляр застосунку наразі дійсний лише для серії VC1 PLCs.

1. Стандартне опитування

Цей екземпляр є простим екземпляром опитування, встановлює M0, M1 та M2, і три MODRW інструкції по черзі звертаються до пристрою відповідно до налаштувань.

Під час виконання, якщо будь-який з M елементів скинуто, відповідна йому MODRW інструкція виходить з опитування, але інші MODRWs все ще виконуються в режимі опитування. Наприклад, якщо M2 скинуто, то MODRW інструкції, що відповідають M0 та M1, звертаються до пристрою по черзі.

Аналогічно, можливо вставити одну MODRW інструкцію під час роботи. Наприклад, якщо M2 встановлено знову, то три MODRW інструкції звертаються до пристрою по черзі.



Коли виникає помилка в інструкції MODRW, встановлюється SM126, SD133 вказує її коди помилок. Значення SM126, SD133 можуть бути змінені іншими інструкціями MODRW, тому їхній стан необхідно записати перед наступним MODRW інструкцію виконано.

MODRW код помилки інструкції наведено нижче:

Код	Помилка назва	Детальний опис
1	Неприпустимий код функції	
2	Неприпустима адреса	
3	Неприпустимі дані	
4-15	Зарезервовано	

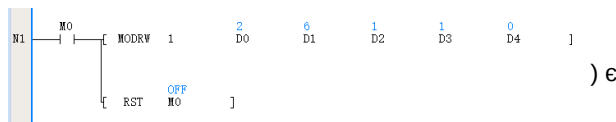
17	Зарезервовано	
18	Помилка налаштування параметрів error	Налаштування параметра (режим або ведучий/ведений) помилка
19	Помилка параметра S2, а саме помилка параметра адреси підпорядкованого пристрою	Номер станції збігається з номером, встановленим інструкцією, або помилка перевищення адреси виникає
20	Параметр D помилка, а саме переповнення адреси елемента	Переповнення адреси елемента (Обсяг отриманих або переданих даних перевищує обсяг пам'яті простір елемента)
21	Інструкція виконання не вдалося	
22	Адреса не збігається	Отримана адреса підлеглого пристрою не збігається із запитаною адресою підлеглого пристрою, а елемент коду помилки зберігає отримана адреса підпорядкованого пристрою
23	Код функції не збігається	Отриманий код функції не збігається із запитаним кодом функції, а елемент коду помилки зберігає отримано функціональний код
24	Інформаційний кадр помилка	Помилка кадру інформації: це означає, що початкова адреса елемента не збігається, елемент коду помилки зберігає отриманий початок елемента адреса
25	Довжина даних не збігається	Довжина отриманих даних не відповідає протоколу, або кількість елементів перевищує максимальне обмеження, вказане цим функція code
26	CRC/LRC перевірте помилка	
27	Зарезервовано	
28	Помилка параметра S3, а саме помилка параметра елемента	Помилка налаштування початкової адреси параметра інструкції

16	Тайм-аут зв'язку	Час зв'язку перевищує встановлений максимальний час зв'язку користувачем користувачем

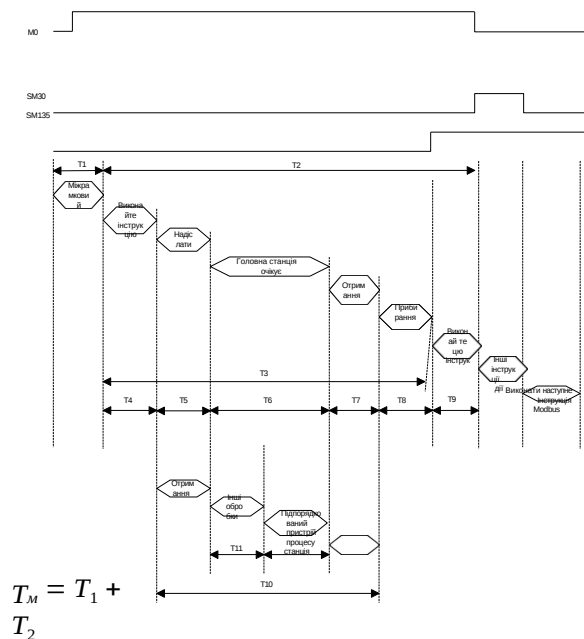
33	Параметр є незмінним під час роботи	Параметр не можна змінити під час роботи (тільки EV3000 підтримка)
34	Параметр захищено за допомогою паролю	Параметр захищено за допомогою паролю

2. Посилання час

Ви можете використовувати наступне LAD для реалізації зв'язку між майстром Modbus та підпорядкованими станціями. Для повного зв'язку необхідний час для кожного етапу наведено нижче:



складається з T_1 та T_2 , а саме



У якому T_1 гарантується користувачем, згідно з тим Протокол зв'язку Modbus, міжфреймовий інтервал час має бути щонайменше 3,5 символи. Довжина одного символу становить: старт-біт (1 біт) + довжина даних (7 бітів або 8 бітів) + контроль (0 бітів або 1 біт) + стоп-біт (1 біт або 2 біти).

$$T_2 = \left(\text{INT} \left(\frac{T_3}{T_c} \right) + 1 \right) T_s$$

У якому, T_s — це макс. цикл сканування PLC.

$$T_3 = T_4 + T_5 + T_6 + T_7 + T_8 + T_9$$

T_4 , T_8 , та T_9 потребують менше ніж 1 мс часу.

$$T_5 = \frac{\text{Кількість байтів для відправлення} \times \text{Довжина символу}}{\text{Швидкість передачі даних (біт/с)}} \times 1000 (\text{мс}) + 1 \text{мс}$$

T_6 : Час очікування головної станції залежить від того підпорядкована станція, яка не може перевищувати час очікування попередньо встановленого головного режиму.

$$T_7 = \frac{\text{Кількість байтів для отримання} \times \text{Довжина символу}}{\text{Швидкість передачі даних (біт/с)}} \times 1000 (\text{мс}) + 1 \text{мс}$$

Час обробки підпорядкованої станції можна розрахувати за такою формулою:

$$T_{10} = T_5 + T_{11} + T_{12} + T_7$$

У якому, макс. час сканування T_{11} якого

T_{12} потребує менше ніж 1 мс часу

Наприклад: Встановлення параметрів зв'язку як 19200, парна перевірка, 8-бітні дані, 1 стоповий біт, RTU режим передачі, 10 символів для відправлення та 20 символів для отримання. Тоді час обробки головної станції розраховується наступним чином:

$$T_5 = \frac{10 \times 10}{19200} \times 1000 + 1 = 6.2 \text{мс}$$

$$T_7 = \frac{20 \times 10}{19200} \times 1000 + 1 = 11.4 \text{мс}$$

$$T_4 = T_8 = T_9 \approx 1 \text{мс}$$

Припустимо $T_6 = 35 \text{мс}$, потім

$$T_3 = 1 + 6.2 + 35 + 11.4 + 1 + 1 = 55.6 \text{мс}$$

Припустимо, що макс. цикл сканування становить 10 мс Тоді

$$T_2 = \left(\text{INT} \left(\frac{55.6}{15} \right) + 1 \right) 15 = 60 \text{мс}$$

Час обробки підпорядкованої станції

$$T_{10} = 6.2 + 15 + 1 + 11.4 = 33.6 \text{мс}$$

6.13 Перевірте інструкції

6.13.1 ССІТТ: Перевірте інструкцію

LAD: CCITT (S1) (S2) (D)										Застосовна модель		VC1S VC1 VC2 VC3 VC5						
										Прапорець впливу біт								
IL: CCITT (S1) (S2) (D)										Довжина кроку		7						
Робота d	Тип	Застосовний програмний елемент															Показник	
S1	INT									D				V		R	√	
S2	INT	Мінус и натяг	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C	T	V	Z	R	√	
D	INT									D				V		R	√	

- Операнд опис

S1: Початкова одиниця даних для перевірки

S2: Кількість даних для перевірки ($S2 \geq 0$, інакше система повідомить про помилку операнда) **D:** Результат перевірки

- Опис функції

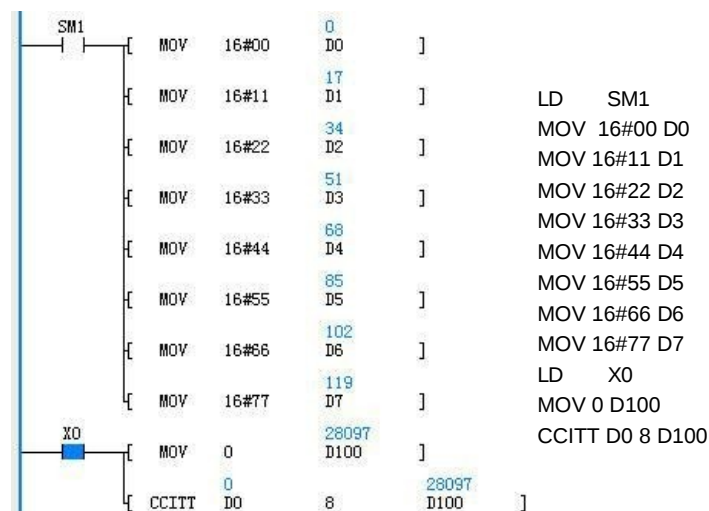
- Виконання операції перевірки ССІТТ для даних **S2**, починаючи з (**S1**), та призначення результату операції для **D**.
- Поліном алгоритму перевірки ССІТТ має вигляд: $X^{16} + X^{12} + X^5 + 1$.

- Примітка

- Система зазвичай переносить вміст **D** перед виконанням інструкції в операцію під час кожного виконання інструкції, тому **D** потрібно ініціалізувати перед виконанням інструкції.
- Дані в області перевірки даних, починаючи з **S2** одиниці за замовчуванням зберігаються у байтовому режимі, тобто,

старший байт приймається за 0, а результат перевірки — 16bits.

- Екземпляр застосунку



Виконання операції перевірки ССІТТ для 8 даних, починаючи з D0, та присвоєння отриманого результату до D100, коли X0=ON.

6.13.2 CRC16: Перевірте інструкцію

LAD: CRC16 (S1) (S2) (D)										Застосовна модель		VC1S VC1 VC2 VC3 VC5						
										Прапорець впливу біт								
IL: CRC16 (S1) (S2) (D)										Довжина кроку		7						
Робота d		Тип		Застосовний програмний елемент													Індекс ація	
S1		INT								D				V		R	√	
S2		INT		Пос тійн а т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√

D	INT								D				V		R	√

- Операнд опис

S1: Початкова одиниця даних для перевірки

S2: Кількість даних для перевірки (**S2**≥0, інакше система видасть помилку операнда)

D: Перевірте результат

- Опис функції

1. Виконання операції перевірки CRC16 для даних **S2**, починаючи з (**S1**), та присвоєння результату операції **D**.

2. Поліном алгоритму перевірки CRC16 має вигляд: $X^{16} + X^{15} + X^2 + 1$.

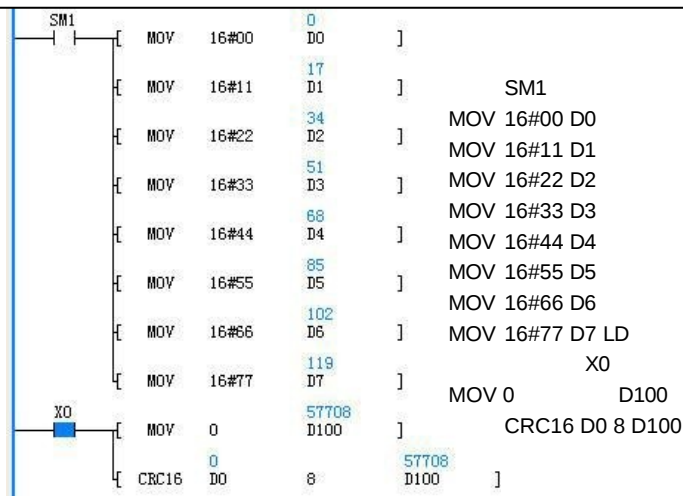
- Примітка

1. Система зазвичай переносить вміст **D** перед виконанням інструкції в операцію під час кожного виконання інструкції, тому **D** потрібно ініціалізувати перед виконанням інструкції.

2. Присвоєння 16#FFFF початковому значенню елемента D (контрольна сума) та перестановка старшого/молодшого байтів (старші 8 біт та молодші 8 біт) під час використання стандартної перевірки Modbus CRC.

3. Дані в області перевірки даних, починаючи з пристрою **S2**, за замовчуванням зберігаються в байтовому режимі, тобто старший байт приймається за 0, а результат перевірки дорівнює 16bits.

- Екземпляр застосунку



Виконання операції перевірки CRC16 для 8 даних, починаючи з D0, та присвоєння отриманого результату D100, коли X0=ON.

6.13.3 LRC: Перевірте інструкцію

LAD:		Застосовна модель										VC1 VC2 VC3 VC5					
[LRC (S1) (S2) (D)]		Прапорець впливу біт															
LRC		(S1)		(S2)		(D)		Довжина кроку									
Робота	тип	Застосовний програмний елемент													Показник		
S1	INT								D				V		R	✓	
S2	INT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD		C	T	V	Z	R	✓
D	INT								D				V		R	✓	

- Операнд опис

S1: Початкова одиниця даних для перевірки

S2: Кількість даних для перевірки (**S2**≥0, інакше система повідомить про помилку операнда)

D: Результат перевірки

- Опис функції

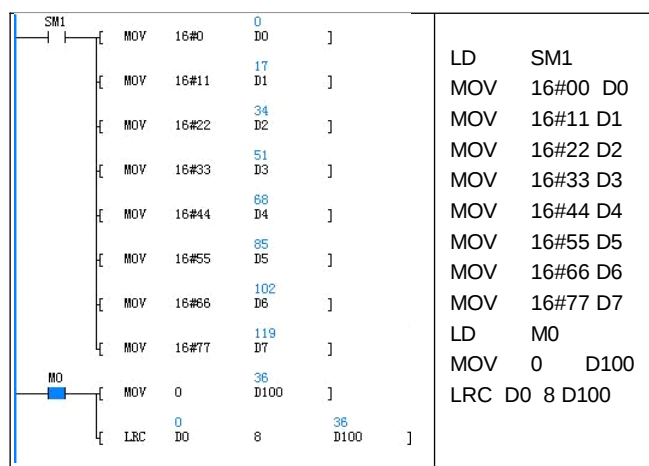
Виконання операції перевірки LRC для даних **S2**, починаючи з (**S1**), та присвоєння результату операції **D**.

- Примітка

1. Система зазвичай переносить вміст **D** перед виконанням інструкції в операцію під час кожного виконання інструкції, тому **D** потрібно ініціалізувати перед виконанням інструкції.

2. Дані в області перевірки даних, починаючи з пристрою **S2**, за замовчуванням зберігаються в байтовому режимі, тобто старший байт приймається за 0, а результат перевірки — це 8 бітів, що зберігаються в молодших байтах **D**.

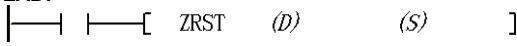
- Екземпляр застосунку



Performing the LRC check operation on 8 data starting from D0, and assigning the obtained result to D100 when X0=ON.

6.14 Розширені інструкції бітової обробки

6.14.1 ZRST: Інструкція зі скидання бітів у 0 у пакетному режимі

LAD: 									Застосовна модель				VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт										
IL: ZRST (D) (S)									Довжина кроку				5						
Робота d	Тип	Застосовний програмний елемент														Показчик			
D	BOOL			Y	M	S	LM				C	T				✓			
S	INT	Константа муреха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓			

- Опис операнда **D**:

Операнд

призначення **S**:

Операнд джерела

- Опис функції

Очищення **S** послідовних бітових елементів, починаючи з

D коли потік енергії є дійсним.

- Примітка

1. Очищення значення лічильника в елементі C, коли елемент C очищено.

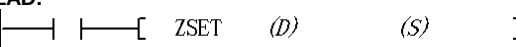
2. Очищення значення часу в елементі T, коли елемент T очищується.

- Екземпляр застосунку



Очищення даних 10 одиниць, починаючи з M10 (а саме M10, M11, M12...M19), коли SM0=ON.

6.14.2 ZSET: Інструкція щодо скидання бітів у пакеті

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт									
IL: ZSET (D) (S)									Довжина кроку			5						
Робота d		Тип	Застосовний програмний елемент													Індекс ація		
D		BOOL			Y	M	S	LM				C	T				✓	
S		INT	Мінус и натяг	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	

- Опис операнда **D**:

Операнд

призначення **S**:

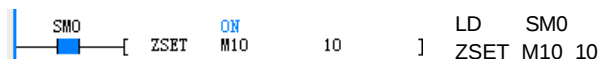
Операнд джерела

- Опис функції

Налаштування **S** послідовних бітових елементів, починаючи

від **D** до 1when потік енергії є дійсним.

- Екземпляр застосунку



Встановлення даних 10 одиниць, починаючи з M10 (а саме M10, M11, M12...M19), на 1, коли SM0=ON.

6.14.3 DECO: Декодування інструкції

LAD: 											Застосовна модель		VC1S VC1 VC2 VC3 VC5				
											Прапорець						

									впливу біт							
IL: DECO (S) (D)									Довжина кроку			5				
Робота d	Тип	Застосовний програмний елемент														Індексація
S	INT	Постійна т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√
D	INT			KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√

- Операнд опис

S: Джерельний операнд

D: Цільовий операнд

- Опис функції

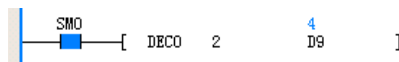
Встановлення біта **S** у слові елемента **D** в 1 та скидання інших бітів, коли потік енергії є дійсним.

- Примітка

- Допустимий діапазон **S** становить від 0 до 15.

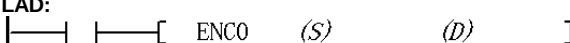
1. Коли **S** більше 15 або менше 0, а потік енергії є дійсним, значення **D** не змінюється, але система повідомляє про помилку значення операнда інструкції.

- Екземпляр застосунку



Встановлення біта 2 у D9 в 1 та скидання інших бітів, коли потік енергії є дійсним.

6.14.4 ENCO: Інструкція з кодування інструкція

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5						
										Прапорець впливу біт								
IL: ENCO (S) (D)										Довжина кроку		5						
Робота d	Тип	Застосовний програмний елемент															Індексація	
S	INT	Пос тійн а т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	Z	R	√		

- Операнд опис

S: Джерельний операнд;

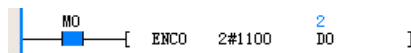
D: Цільовий операнд

- Опис функції

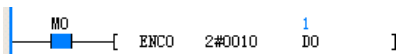
Запис номера біта, значення якого дорівнює "1", у словному елементі **S** до **D**, коли потік енергії є дійсним.

- Примітка

Коли значення кількох бітів у **S** дорівнюють "1", найменший номер позиції буде записаний у **D**, як показано на наступному рисунку.



- Екземпляр застосунку



LD MO
ENCO 2#0010 D0

Операнд 1 дорівнює 2#0010, а біт 1 є "1", отже результат дорівнює 1, який записується в D0, коли потік енергії є дійсним.

6.14.5 BITS: Інструкція для підрахунку бітів у <A>слові

LAD: [BITS (S) (D)]										Застосовна модель		VC1S VC1 VC2 VC3 VC5						
										Прапорець впливу біт								
IL: BITS (S) (D)										Довжина кроку		5						
Робота d		Тип	Застосовний програмний елемент													Індекс ація		
S	INT	Пос тійн а т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		
D	INT			KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√		

- Операнд опис

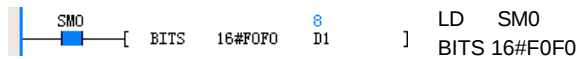
S: Джерельний операнд; **D:** Цільовий операнд

- Опис функції

Підрахунок кількості бітів "1" в операнді **S** та збереження статистичного результату

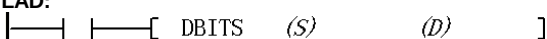
в операнді **D** коли потік енергії є дійсним.

- Екземпляр застосунку



Коли потік енергії дійсний, **S** в інструкції BITS є константою 16#F0F0, яка містить 8 бітів, значення яких дорівнює "1" (стан ON). Результат обчислення дорівнює 8 і зберігається в **D** (D1).

6.14.6 DBITS: Інструкція для підрахунку бітів у подвійному слові

LAD: 									Застосовна модель			VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт								
IL: DBITS (S) (D)									Довжина кроку			6					
Робота d		Тип	Застосовний програмний елемент													Індексація	
S	DINT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
D	INT			KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓	

- Операнд опис

S: Вихідний операнд; **D:** Цільовий операнд

- Опис функції

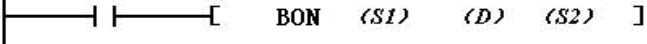
Підрахунок кількості бітів "1" у подвійному слові **S** та збереження статистичного результату в **D**, коли потік енергії є дійсним.

- Екземпляр застосунку



Коли потік енергії дійсний, **S** в інструкції DBITS є константою 16#FF0FF, яка містить 16 бітів, значення яких дорівнює "1" (стан ON). Результат обчислення дорівнює 16 і зберігається в **D** (D10).

6.14.7 BON: Інструкція щодо визначення біта у слові word

LAD:										Застосовна модель		VC2 VC3 VC5					
										Прапорець впливу біт							
IL: BON (S1) (D) (S2)										Довжина кроку		7					
Робота d		Тип	Застосовний програмний елемент													Індексація	
S1	INT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V			✓	
D	BOOL		Y	M	S												
S2	INT								D				V		R		

- Операнд опис

S: Вихідний операнд; **D:** Цільовий операнд

- Опис функції

Підрахунок стану біта **S2** у слові **S1** та виведення отриманого результату на **D** за умови валідного потоку енергії.

- Екземпляр застосунку



Коли потік енергії дійсний, **S1** в інструкції BON є константою D0, а стан її п'ятого біта (ON) виводиться на D (Y0).

6.15 Контактні інструкції

6.15.1 BLD: Контакт біта слова LD інструкція

LAD: BLD (S1) (S2)									Застосовна модель		VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт								
IL: BLD (S1) (S2)									Довжина кроку		5						
Робота d	Тип	Застосовний програмний елемент														Індексація	
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
S2	INT	Мінуси натяг	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	

- Операнд опис

S1: Джерело операнда

S2: Призначений біт, $0 \leq S2 \leq 15$, в іншому випадку система повідомляє про помилку операнда.



- Опис функції

Взяття стану біта **S2** у **S1** та використання отриманого значення для керування подальшою операцією.

Взяття стану біта 5 (ON) у D0 (1000: 2#0000001111101000) та використання отриманого значення для визначення стану Y0 у наступній операції.

- Екземпляр застосунку

6.15.2 BLDI: Контакт біта слова LDI інструкція

LAD: BLDI (S1) (S2)									Застосовна модель			VC1S VC1 VC2 VC3 VC5						
									Прапорець впливу біт									
IL: BLDI (S1) (S2)									Довжина кроку			5						
Робота d	Тип	Застосовний програмний елемент														Індексація		
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓		
S2	INT	Пос тій н а т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓		

- Операнд опис

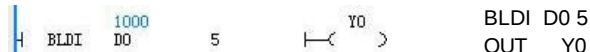
S1: Джерело операнда

S2: Призначений біт, $0 \leq S2 \leq 15$, інакше система повідомляє про помилку операнда.

- Опис функції

Виконання операції NOT над станом біта **S2** у **S1** та використання отриманого значення для керування наступною операцією.

- Екземпляр застосунку



Виконання NOT операції (OFF) над станом біта 5 (ON) у D0 (1000: 2#0000001111101000), та використання отриманого значення для визначення вихідного стану Y0 у подальшій операції.

6.15.3 BAND: Контакт слова біт AND інструкція

LAD Примітка: 			Застосовна модель		VC1S VC1 VC2 VC3 VC5				
Примітка: Логічний взаємозв'язок було відображено на діаграмі, тому інструкцію BAND відображено як BLD у LAD			Прапорець впливу біт						
IL: BAND (S1) (S2)			Довжина кроку		5				
Робота d	Тип	Застосовний програмний елемент					Індексація		

S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√
S2	INT	Пос тійн а т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√

- Операнд опис

S1: Джерело операнда

S2: Призначений біт ($0 \leq S2 \leq 15$, в іншому випадку система повідомляє про помилку операнда)

- Опис функції

Враховуючи стан біта **S2** у **S1**, та використовуючи отримане значення у serial

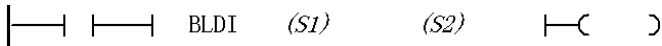
з'єднання з іншими вузлами для керування наступною операцією.

- Екземпляр застосунку



Беручи стан біта 5 (ON) у D0 (1000: 2#0000001111101000) та використовуючи отримане значення у послідовному з'єднанні з іншими вузлами ($X0=ON$) для визначення вихідного стану Y0 у наступній операції.

6.15.4 BANI: Контакт біта слова ANI інструкція

LAD Примітка:  Примітка: Логічний взаємозв'язок було відображено на діаграмі, тому інструкцію BANI відображено як BLDI у LAD										Застосовна модель	VC1S VC1 VC2 VC3 VC5					
IL: BANI (S1) (S2)										Прапорець впливу біт						
										Довжина кроку	5					
Робота d	Тип	Застосовний програмний елемент														Індексація
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
S2	INT	Постійна	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

S1: Джерело операнда

S2: Призначений біт ($0 \leq S2 \leq 15$, в іншому випадку система повідомляє про помилку операнда)

- Опис функції

Виконання операції NOT над станом біта **S2** у **S1** та використання отриманого значення у serial

з'єднання з іншими вузлами для керування наступною операцією.

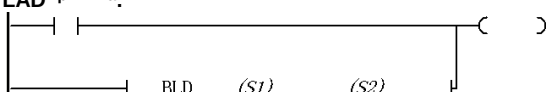
- Екземпляр застосунку



```
LD X0
BANI D0 5
OUT Y0
```

Виконання операції NOT (OFF) над станом біта 5 (ON) у D0(1000: 2#0000001111101000) та використання отриманого значення у послідовному з'єднанні з іншими вузлами (X0=ON) для визначення вихідного стану Y0 у наступній операції.

6.15.5 BOR: Інструкція ORконтакту бітового слова

LAD Примітка:  Примітка: Логічний взаємозв'язок було відображено на схемі, тому інструкцію BOR відображено як BLD у LAD										Застосовна модель	VC1S VC1 VC2 VC3 VC5					
IL: BOR (S1) (S2)										Прапорець впливу біт						
										Довжина кроку	5					
Робота d	Тип	Застосовний програмний елемент														Індексація
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
S2	INT	Постійна	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

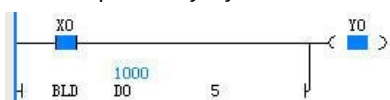
S1: Джерело операнда

S2: Біт призначення ($0 \leq S2 \leq 15$, в іншому разі буде повідомлено про помилку операнда)

- Опис функції

Враховуючи стан біта **S2** у **S1** та використовуючи отримане значення у паралельному з'єднанні з іншими вузлами для керування наступною операцією.

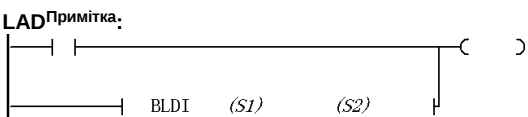
- Екземпляр застосунку



```
LD X0
BOR D0 5
OUT Y0
```

Беручи стан біта 5 (ON) у D0 (1000: 2#0000001111101000) та використовуючи отримане значення у паралельному з'єднанні з іншими вузлами (X0=ON) для визначення вихідного стану Y0 у наступній операції.

6.15.6 BORI: Контакт біта слова ORI інструкція

LAD Примітка: 										Застосовна модель	VC1S VC1 VC2 VC3 VC5					
Примітка: Логічний взаємозв'язок було відображено на діаграмі, тому інструкцію BORI відображено як BLDI у LAD										Прапорець впливу біт						
IL: BORI (S1) (S2)										Довжина кроку	5					
Робота d	Тип	Застосовний програмний елемент														Індексація
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
S2	INT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

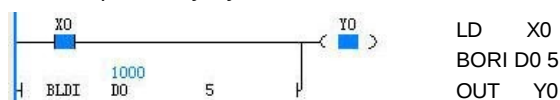
S1: Джерело операнда

S2: Призначений біт ($0 \leq S2 \leq 15$, в іншому випадку система повідомляє про помилку операнда)

- Опис функції

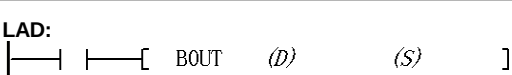
Виконання операції NOT над станом біта **S2** у **S1** та використання отриманого значення у паралельному з'єднанні з іншими вузлами для керування наступною операцією.

- Екземпляр застосунку



Виконання операції NOT (OFF) над станом біта 5 (ON) в D0 (1000: 2#0000001111101000), та використання отриманого значення у паралельному з'єднанні з іншими вузлами ($X0=ON$) для визначення вихідного стану $Y0$ у наступній операції.

6.15.7 BOUT: Інструкція виходу бітової котушки слова

LAD: 										Застосовна модель	VC1S VC1 VC2 VC3 VC5					
										Прапорець впливу біт						
IL: BOUT (D) (S)										Довжина кроку	5					
Робота d	Тип	Застосовний програмний елемент														Індексація
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓
S	INT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

S1: Джерело операнда

S2: Призначений біт ($0 \leq S2 \leq 15$, інакше система повідомляє про помилку операнда)

- Опис функції

Призначення стану поточного потоку енергії біту **S** у **D**.

- Екземпляр застосунку



Присвоєння стану поточного енергетичного потоку ($X0=ON$) біту 4 у D0 (1000:2#0000001111101000). Після виконання: $D0=1016$ (2#000000111111000).

6.15.8 BSET: Бітова катушка слова інструкція встановлення

LAD:			Застосовна модель	VC1S VC1 VC2 VC3 VC5				
			Прапорець впливу біт					
IL: BSET (D) (S)			Довжина кроку	5				
Робота d	Тип	Застосовний програмний елемент					Індексація	

D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓
S	INT	Пос тійн а т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

D: Цільовий операнд

S2: Призначений біт ($0 \leq S2 \leq 15$, інакше система повідомляє про помилку операнда)

- Опис функції

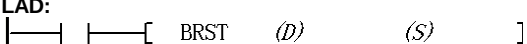
Встановлення біта **S** елемента **D**.

- Екземпляр застосунку



Встановлення 15-го біта в D0 (1000: 2#0000001111101000), коли потік енергії є дійсним. Після виконання D0=33768 (2#1000001111101000).

6.15.9 BRST: Інструкція скидання бітової котушки слова instruction

LAD: 									Застосовна модель		VC1S VC1 VC2 VC3 VC5					
									Прапорець впливу біт							
IL: BRST (D) (S)									Довжина кроку		5					
Робота d	Тип	Застосовний програмний елемент														Індексація
D	INT			KnY	KnM	KnS	KnLM		D		C	T	V	Z	R	✓
S	INT	Пос тійн а т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

D: Цільовий операнд

S2: Призначений біт ($0 \leq S2 \leq 15$, інакше система повідомляє про помилку операнда)

- Опис функції

Очищення біта **S** елемента **D**.

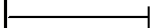
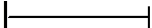
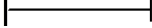
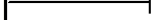
- Екземпляр застосунку



Очищення 8-го біта в D0 (1000: 2#0000001111101000), коли потік енергії є дійсним. Після виконання, D0=744 (2#0000001011101000).

6.16 Інструкції контакту порівняння

6.16.1 LD (=, <, >, <>, >=, <=): Порівняння цілих чисел LD-інструкція

LAD:						Застосовна модель	VC1S VC1 VC2 VC3 VC5					
						=	(S1)	(S2)		)	Прапорець впливу біт	
						<	(S1)	(S2)		)		
						>	(S1)	(S2)		)		
						<>	(S1)	(S2)		)		
						>=	(S1)	(S2)		)		
						<=	(S1)	(S2)		)		
IL: LD=						(S1)	(S2)	Довжина кроку				5
LD<						(S1)	(S2)					
LD>						(S1)	(S2)					
LD<>						(S1)	(S2)					
LD>=						(S1)	(S2)					
LD<=						(S1)	(S2)					
Робота d	Тип	Застосовний програмний елемент									Індексація	

S1	INT	Мінус и н а т я г	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
S2	INT	Мінус и н а т я г	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

S1: Параметр порівняння 1

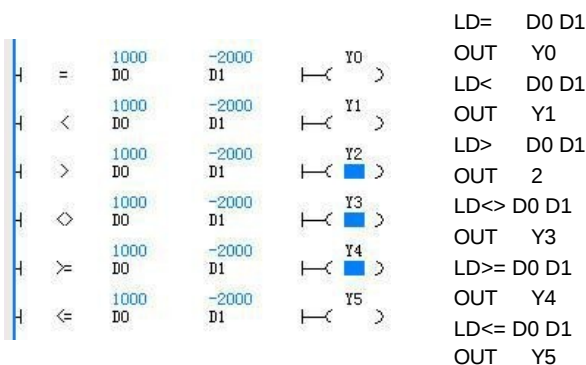
S2: Параметр порівняння 2

- Опис функції

Виконання BIN порівняння вмісту

S1 та **S2** і використання результату порівняння для керування подальшою операцією.

- Екземпляр застосунку



Виконання BIN порівняння даних D0 та D1, і використання результату порівняння для визначення вихідного стану наступних елементів.

6.16.2 AND (=, <, >, <>, >=, <=): Порівняння цілих чисел AND жінструкція

LAD:										Застосовна модель			VC1S VC1 VC2 VC3 VC5				
		=	(S1)	(S2)		)	Прапорець впливу біт										
		<	(S1)	(S2)		)											
		>	(S1)	(S2)		)											
		<>	(S1)	(S2)		)											
		>=	(S1)	(S2)		)											
		<=	(S1)	(S2)		)											
IL: AND=							(S1)	(S2)	Довжина кроку			5					
AND<							(S1)	(S2)									
AND>							(S1)	(S2)									
AND<>							(S1)	(S2)									
AND>=							(S1)	(S2)									
AND<=							(S1)	(S2)									
Робота d	Тип	Застосовний програмний елемент														Індекс ія	
S1	INT	Кон ста нт	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	
S1	INT	Пос тійн а т	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	√	

- Екземпляр застосунку

Виконання BY 3IN порівняння даних D0 та D1, а також використання результату порівняння у послідовному з'єднанні з іншими вузлами для визначення стану вищих даних для наступних елементів.

```

OUT    Y4
LD      X5
AND<=  D0 D1
OUT     Y5

```

6.16.3 OR (=, <, >, <>, >=, <=): Порівняння цілих чисел OR-інструкція

LAD:		Застосовна модель		VC1	VC2	VC3	VC5									
		Прапорець впливу біт														
OR= (S1) (S2) OR< (S1) (S2) OR> (S1) (S2) OR<> (S1) (S2) OR>= (S1) (S2) OR<= (S1) (S2)		Довжина кроку														
Робота d	Тип	Застосовний програмний елемент										Індексація				
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓

- Операнд опис

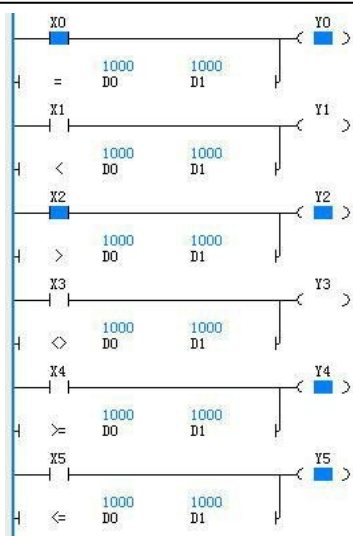
S1: Параметр порівняння 1

S2: Параметр порівняння 2

- Опис функції

Виконання порівняння вмісту **S1** та **S2** і використання результату порівняння в паралельному з'єднанні з іншими вузлами для керування наступною операцією.

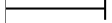
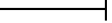
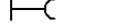
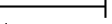
- Екземпляр застосунку



ЛД проводиться порівняння даних D0 та D1, і використовуючи результат у паралельному з'єднанні з іншими вузлами для визначення стану наступних елементів.

```
LD X1
OR< D0 D1 OUT Y1
LD X2
OR<> D0 D1
OUT Y2
LD X3
OR>= D0 D2
OUT Y3
LD X4
OR>= D0 D1
OUT Y4
LD X5
OR<= D0 D1
OUT Y5
```

6.16.4 LDD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел LDD інструкція

LAD:		Застосовна модель										VC1S VC1 VC2 VC3 VC5					
	D=	(S1)	(S2)			Прапорець впливу біт											
	D<	(S1)	(S2)														
	D>	(S1)	(S2)														
	D<>	(S1)	(S2)														
	D>=	(S1)	(S2)														
	D<=	(S1)	(S2)			Довжина кроку											
LDD=																(S1)	(S2)
LDD<																(S1)	(S2)
LDD>																(S1)	(S2)
LDD<>																(S1)	(S2)
LDD>=						(S1)	(S2)										
LDD<=						(S1)	(S2)										
Робота d	Тип	Застосовний програмний елемент										Індикація					
S1	DINT	Пос- тійна	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	V	R	✓			
S2	DINT	Пос- тійна	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	V	R	✓			

● Operand description

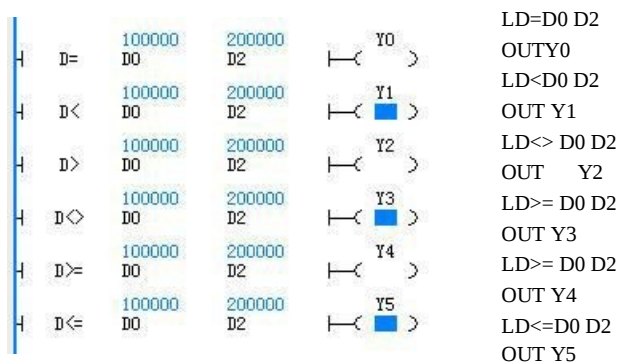
S1: Параметр порівняння 1

S2: Параметр порівняння 2

● Опис функції

Виконання порівняння вмісту **S1** та **S2** і використання результату порівняння для керування подальшою операцією.

● Екземпляр застосунку



```
LD=D0 D2
OUTY0
LD<D0 D2
OUT Y1
LD<> D0 D2
OUT Y2
LD>= D0 D2
OUT Y3
LD>= D0 D2
OUT Y4
LD<=D0 D2
OUT Y5
```

Виконання порівняння для (D0,D1) та (D2,D3) і використання результату порівняння для визначення вихідного стану наступних елементів.

6.16.5 ANDD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел ANDD-інструкція

LAD:										Застосовна модель		VC1S VC1 VC2 VC3 VC5				
		D=	(S1)	(S2)		)	Прапорець впливу біт									
		D<	(S1)	(S2)		)										
		D>	(S1)	(S2)		)										
		D<>	(S1)	(S2)		)										
		D>=	(S1)	(S2)		)										
		D<=	(S1)	(S2)		)										
IL: ANDD= (S1) (S2) ANDD< (S1) (S2) ANDD> (S1) (S2) ANDD< > (S1) (S2) ANDD > = (S1) (S2) ANDD<= (S1) (S2)								Довжина кроку		7						
Робота d	Тип	Застосовний програмний елемент														Індексація
S1	DINT	Склад ається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
S2	DINT	Склад ається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓

● Operand description

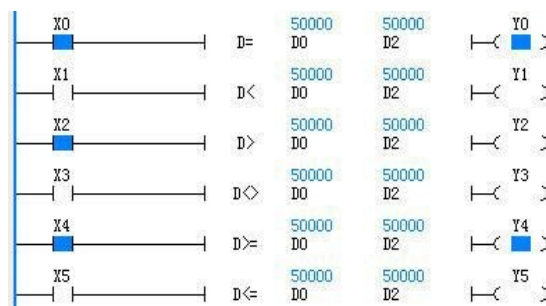
S1: Параметр порівняння 1

S2: Параметр порівняння 2

● Опис функції

Виконання порівняння вмісту **S1** та **S2** і використання результату порівняння в послідовному з'єднанні з іншими вузлами для керування подальшою операцією.

● Екземпляр застосунку



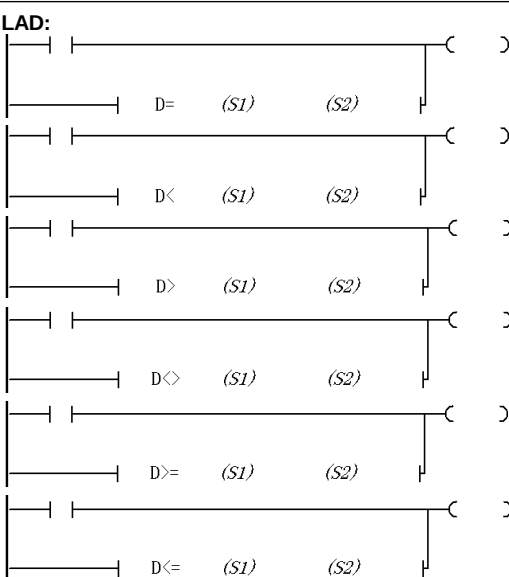
Виконання порівняння для (D0,D1) та (D2,D3) і використання результату порівняння у послідовному з'єднанні з іншими вузлами для визначення вихідного стану наступних елементів.

```

LD      X0
LDD=    D0 D2
OUT     Y0
LD      X1
LDD<    D0 D2
OUT     Y1
LD      X2
LDD<>   D0 D2
OUT     Y2
LD      X3
LDD<>   D0 D2
OUT     Y3
LD      X4
LDD>=   D0 D2
OUT     Y5

```

6.16.6 ORD (=, <, >, <>, >=, <=): Порівняння довгих цілих чисел ORD-інструкція

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3 VC5						
										Прапорець впливу біт								
IL: ORD= ORD< (S1) (S2) ORD> (S1) (S2) ORD<> (S1) (S2) ORD>= (S1) (S2) ORD<= (S1) (S2)										Довжина кроку		7						
Робота d	Тип	Застосовний програмний елемент														Індексація		
S1	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓		
S2	DINT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓		

● Operand description

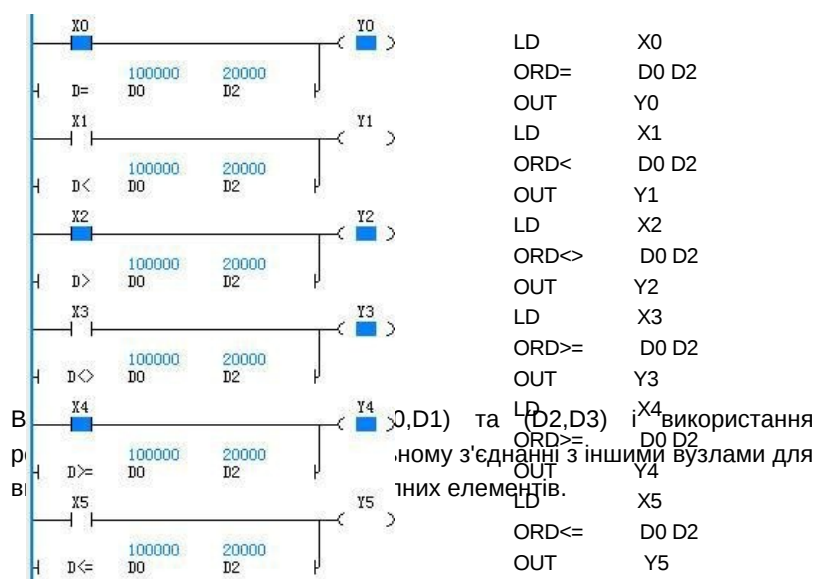
S1: Параметр порівняння 1

S2: Параметр порівняння 2

● Опис функції

Виконання порівняння вмісту **S1** та **S2** і використання результату порівняння в паралельному з'єднанні з іншими вузлами для керування наступною операцією.

● Екземпляр застосування



6.16.7 LDR: Інструкція порівняння чисел з рухомою комою instruction

LAD:								Застосовна модель		VC1S VC1 VC2 VC3 VC5					
		R=	(S1)	(S2)	┌─┐										
		R<	(S1)	(S2)	┌─┐										
		R>	(S1)	(S2)	┌─┐										
		R<>	(S1)	(S2)	┌─┐										
		R>=	(S1)	(S2)	┌─┐			Прапорець впливу біт							
		R<=	(S1)	(S2)	┌─┐										
IL: LDR=						(S1) (S2)									
						LDR<		(S1) (S2)							
						LDR>		(S1) (S2)							
						LDR<>		(S1) (S2)							
						LDR>=		(S1) (S2)							
						LDR<=		(S1) (S2)							
								Довжина кроку		7					

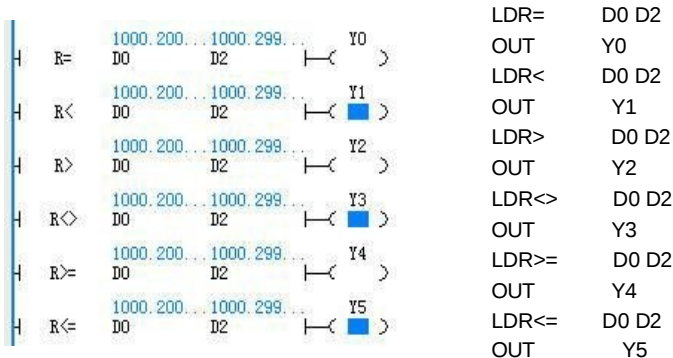
Робота d	Тип	Застосовний програмний елемент													Індексація	
S1	REAL	Мінус и натяг							D				V		R	√
S2	RAEL	Мінус и натяг							D				V		R	√

● Operand description

S1: Параметр порівняння 1

S2: Параметр порівняння 2

- Опис функції
Виконання порівняння вмісту **S1** та **S2** і використання результату порівняння для керування подальшою операцією.
- Екземпляр застосунку



Виконання порівняння для (D0,D1) та (D2,D3) і використання результату порівняння для визначення вихідного стану наступних елементів.

6.16.8 ANDR: Інструкція порівняння чисел з рухомою комою instruction

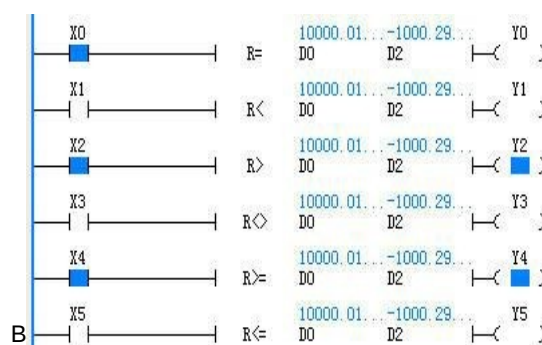
LAD:										Застосовна модель		VC1S VC1 VC2 VC3 VC5				
		R=	(S1)	(S2)					Прапорець впливу біт							
		R<	(S1)	(S2)												
		R>	(S1)	(S2)												
		R<>	(S1)	(S2)												
		R>=	(S1)	(S2)												
		R<=	(S1)	(S2)												
IL: ANDR= (S1) (S2)										Довжина кроку	7					
ANDR< (S1) (S2)																
ANDR> (S1) (S2)																
ANDR<> (S1) (S2)																
ANDR>= (S1) (S2)																
ANDR<= (S1) (S2)																
Робота d	Тип	Застосовний програмний елемент													Індексація	
S1	REAL	Пос тійн а т							D				V		R	✓
S2	REAL	Пос тійн а т							D				V		R	✓

● Operand description

S1: Параметр порівняння 1

S2: Параметр порівняння 2

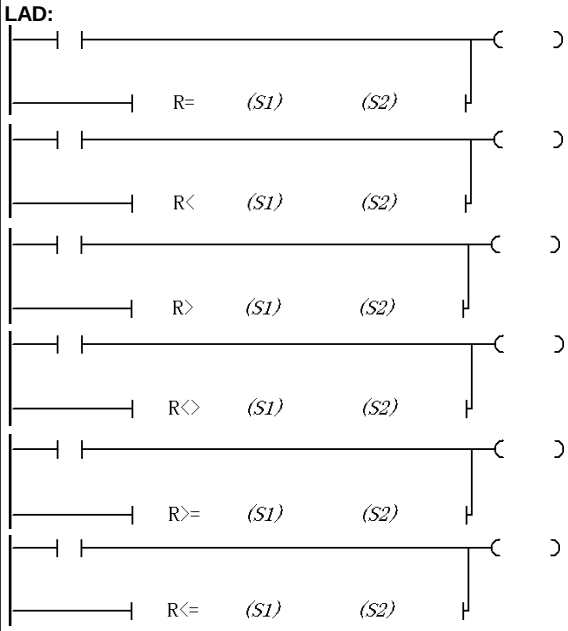
- Опис функції
Виконання порівняння вмісту **S1** та **S2** і використання результату порівняння у послідовному з'єднанні з іншими вузлами для керування подальшою операцією.
- Екземпляр застосунку



і використання результату порівняння у послідовному з'єднанні з іншими вузлами для визначення вихідного стану наступних елементів.

```
LD      X0
ANDR=   D0 D2
OUT     Y0
LD      X1
ANDR<   D0 D2
OUT     Y1
LD      X2
ANDR<>  D0 D2
OUT     Y2
LD      X3
ANDR<>  Y3
LD      X4
ANDR>=  D0 D2
OUT     Y5
```

6.16.9 ORR: Інструкція порівняння чисел з рухомою комою instruction

LAD:									Застосовна модель				VC1S VC1 VC2 VC3 VC5				
									Прапорець впливу біт								
IL: ORR= (S1) (S2)									Довжина кроку				7				
ORR< (S1) (S2)																	
ORR> (S1) (S2)																	
ORR<> (S1) (S2)																	
ORR>= (S1) (S2)																	
ORR<= (S1) (S2)																	

Робота d	Тип	Застосовний програмний елемент												Індексація		
S1	REAL	Постійна							D				V		R	✓
S2	REAL	Постійна							D				V		R	✓

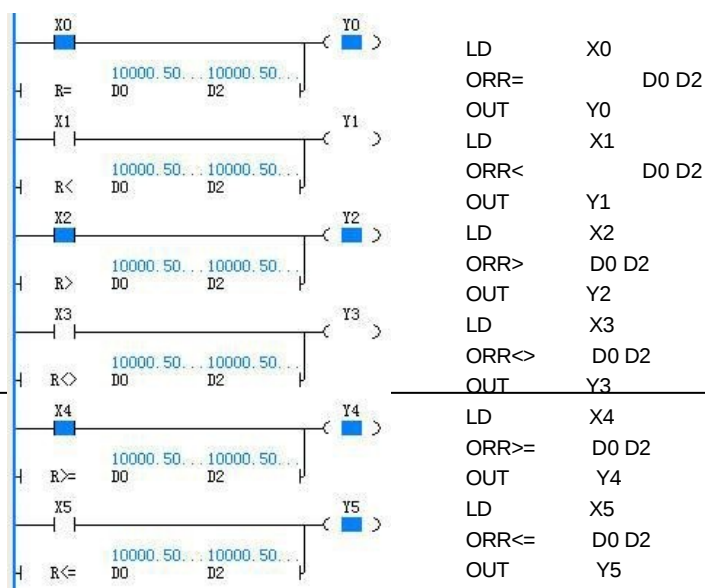
● Operand description

S1: Параметр порівняння 1

S2: Параметр порівняння 2

- Опис функції
Виконання порівняння вмісту **S1** та **S2** і використання результату порівняння в паралельному з'єднанні з іншими вузлами для керування наступною операцією.
- Екземпляр застосунку

Виконання порівняння для (D0,D1) та (D2,D3) і використання результату порівняння у паралельному з'єднанні з іншими вузлами для визначення вихідного стану



наступних елементів.

6.16.10 CMP: Інструкція для встановлення порівняння цілих чисел на ON

LAD: 							

- Операнд опис

S1: Номер даних або програмного елемента для порівняння.

S2: Номер даних або програмного елемента, що є джерелом для порівняння.

D: Номер початкового елемента, що використовується для виведення результату

- Опис функції

Виконання інструкції та порівняння **S1** і **S2** за умови, що потік енергії є дійсним. Вибір одного з (**D**) (**D**+1) (**D**+2) ON відповідно до результату (менше, дорівнює або більше).

- Application instance



6.16.11 LCMP: Інструкція для налаштування порівняння довгих цілих чисел на ON

LAD:	Придатна модель								VC2 VC3 VC5							
	</															

- Операнд опис

S1: Порівняльне значення 1.

S2: Порівняльне значення 2.

D: Номер початкового елемента, який використовується для виведення результату.

Виконання інструкції та порівняння **S1** і **S2**, коли потік енергії є дійсним. Вибір одного з (**D**) (**D**+1) (**D**+2) ON відповідно до результату (менше, дорівнює або більше).

- Екземпляр застосунку

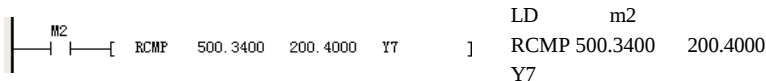
- Опис функції



6.16.12 RCMP: Інструкція для встановлення порівняння чисел з рухомою комою на ON

LAD: [RCMP (S1) (S2) (D)]								Придатна модель		VC2 VC3 VC5						
								Прапорець впливу біт								
IL: RCMP (S1) (S2) (D)								Довжина кроку		9						
Робота d		Тип	Застосовний програмний елемент												Показник	
S1		REAL	Складається немає						D					R	✓	
S2		REAL	Складається немає						D					R	✓	
D		BOOL			Y	M	S									

- Операнд опис
S1: Порівняльне значення 1.
S2: Порівняльне значення 2.
D: Номер початкового елемента, який використовується для виведення результату.
- Виконання інструкції та порівняння **S1** і **S2**, коли потік енергії є дійсним. Вибір одного з (**D**) (**D**+1) (**D**+2) ON відповідно до результату (менше, дорівнює або більше).
- Application instance



6.17 Інструкції з масової обробки даних інструкції

6.17.1 BKADD: Інструкція з операції масового додавання даних інструкція

LAD:	Придатна модель									VC3 VC5							
	[BKADD (S1) (S2) (D) (S3)]									Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики					
	BKADD (S1) (S2) (D) (S3)									Довжина кроку							
	Робота d	Тип	Застосовний програмний елемент												Показник		
S1	INT								D	SD	C	T	V	R	✓		
S2	INT	Складається немає							D	SD	C	T	V	R	✓		
D	INT								D	SD	C	T	V	R	✓		
S3	INT	Складається немає							D				V	R			

- Операнд опис
S1: Номер початкового програмного елемента для збереження даних, над якими має виконуватися операція додавання
S2: Константа для виконання операції додавання або номер початкового програмного елемента, що зберігає дані, над якими має бути виконана операція
- додавання
D: Номер початкового програмного елемента для збереження результату операції
S3: Кількість даних
- Опис функції

1. Виконання інструкції, збільшення 16-бітних даних S3 одиниць, починаючи з S1, на 16-бітні дані S3 одиниць (BIN), починаючи з S2, та збереження результату операції в S3 одиницях, починаючи з D, коли потік енергії є дійсним.

2. Ви можете безпосередньо вказати 16-бітну константу в S2. При послідовному виконанні, коли S2 є константою, операція додавання 16-бітних даних S3 одиниць, починаючи з S1 та S2, і збереження результату операції в одиницях S3, починаючи з D.

- Примітка

Коли результат операції переповнюється, біт прапора переносу не встановлюється в ON.

- Екземпляр застосунку

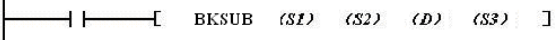


LD M1

BKADD D10 D100 D1000 5

Послідовне додавання вмісту 5 одиниць, починаючи з D10, до вмісту 5 одиниць, починаючи з D100, із записом результатів операції у 5 одиниць, починаючи з D1000, коли M1=ON. У цьому випадку D1000=D10+D100, D1001=D11+D101,....., D1004=D14+D104.

6.17.2 BKSUB: Операція віднімання масиву даних інструкція

LAD:									Придатна модель		VC3 VC5						
									Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики						
IL: BKSUB (S1) (S2) (D) (S3)									Довжина кроку		9						
Робота d	Тип	Застосовний програмний елемент													Показник		
S1	INT								D	SD	C	T	V	R	✓		
S2	INT	Складається немає							D	SD	C	T	V	R	✓		
D	INT								D	SD	C	T	V	R	✓		
S3	INT	Складається немає							D				V	R			

- Операнд опис

S1: Номер початкового програмного елемента для збереження даних, над якими має виконуватися операція віднімання

S2: Константа для виконання операції додавання або номер початкового програмного елемента, що зберігає дані, над якими необхідно виконати операцію віднімання

D: Номер початкового програмного елемента для збереження результату операції

S3: Кількість даних

- Опис функції

1. Виконання інструкції, віднімання 16-бітних даних у кількості S3, починаючи з S1, від 16-бітних даних у кількості S3 (BIN), починаючи з S2, та збереження результату операції в одиницях S3, починаючи з D, коли потік енергії є дійсним.

2. Ви можете безпосередньо вказати 16-бітну константу в S2. При послідовному виконанні, коли S2 є константою, операція віднімання над 16-бітними даними блоків S3 починається

з S1 та S2, і зберігаючи результат операції в одиницях S3, починаючи з D.

- Примітка

Коли результат операції переповнюється, біт прапорця переносу не встановлюється в ON.

- Екземпляр застосунку



LD M1

BKSUB D10 D100 D1000 5

Послідовне віднімання вмісту 5 одиниць, починаючи з D10, від вмісту 5 одиниць, починаючи з D100, та збереження результатів операції у 5 одиницях, починаючи з D1000, коли M1=ON. У цьому випадку D1000=D10-D100, D1001=D11-D101,....., D1004=D14-D104.

314

S1	INT	Складається немає							D	SD	C	T	V	R	✓
S2	INT								D	SD	C	T	V	R	✓
D	BOOL		Y	M	S	LM	SM								
S3	INT	Складається немає							D				V	R	

- Операнд опис

S1: Номер початкового програмного елемента для порівняння або збереження

S2: Номер елемента startsoft, у якому зберігається джерело порівняння

D: Номер початкового м'якого елемента, у якому зберігається результат порівняння

S3: Кількість даних

- Опис функції

1. Порівняння 16-бітних даних блоків S3, починаючи з S1, з 16-бітними даними блоків S3 (BIN), починаючи з S2, та збереження результату операції в блоках S3, починаючи з D.

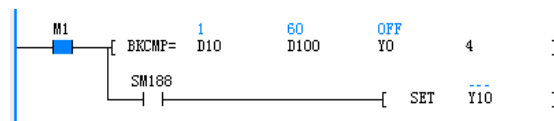
2. Ви можете безпосередньо вказати 16-бітну константу в S1. При послідовному виконанні, коли S1 є константою, порівняння 16-бітних даних S3, що починаються з S1, з даними S2, та збереження результату операції в одиницях S3, що починаються з D.

3. Встановлення прапорця порівняння набору блоку даних (SM188), коли результати порівняння одиниць S3, починаючи з D, дорівнюють ON.

- Примітка

Коли результат операції переповнюється, прапорець переносу не встановлюється ON.

- Екземпляр застосунку



LD M1

BKMP= D10 D100 Y0 4 LD

SM188

SET Y10

Коли M1=ON, ON вміст чотирьох одиниць, починаючи з D10, порівнюється з вмістом чотирьох одиниць, починаючи з D100, і результат зберігається в чотирьох одиницях, починаючи з Y0. Крім того, коли результат порівняння дорівнює ON, Y10 стає ON.

Послідовне порівняння вмісту 4 одиниць, починаючи з D10, із вмістом 4 одиниць, починаючи з D100, та збереження результатів операції у 4 одиницях, починаючи з Y0, коли M1=ON. У цьому випадку Y0 встановлюється ON.

6.18 Технічний паспорт інструкції

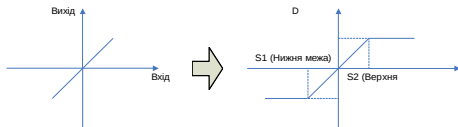
6.18.1 LIMIT: Інструкція з керування верхньою/нижньою межею інструкція

LAD:									Придатна модель	VC3 VC5					
									Прапорець впливу біт	Прапорець нуля, прапорець перенесення та прапорець позики					
IL: LIMIT (S1) (S2) (S3) (D)									Довжина кроку	9					
Оператор	Тип	Застосовний програмний елемент													Показник
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
S3	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓

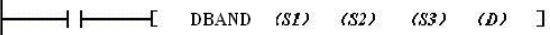
- Опис операнда **S1**:
Значення нижньої
межі **S2**: Значення
верхньої межі
S3: Вхідне значення для керування
верхньою та нижньою межами
- D**: Номер початкового м'якого елемента, у якому зберігається
вихідне значення, що було обмежене верхньою та нижньою
межами

Опис функції

Ви можете визначити, чи значення входу, вказане у S3, знаходиться в межах діапазону верхньої та нижньої меж, визначених S1 та S2, щоб керувати та зберігати значення у D. У цьому випадку, коли $S3 < S1$, $D=S1$; коли $S3 > S2$, $D=S2$; коли $S1 \leq S3 \leq S2$, $D=S2$.



6.18.2 DBAND: Інструкція з керування зоною нечутливості instruction

LAD: 								Придатна модель		VC3 VC5					
								Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики					
IL: DBAND (S1) (S2) (S3) (D)								Довжина кроку		9					
Оператор	Тип	Застосовний програмний елемент													Показники
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
S3	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓

Операнд опис

S1: Нижнє граничне значення зони нечутливості

S2: Верхнє граничне значення зони нечутливості **S3:** Вхідне значення, що керується зоною нечутливості

D: Номер початкового програмного елемента, у якому зберігається вихідне значення, що керувалося за допомогою зони нечутливості

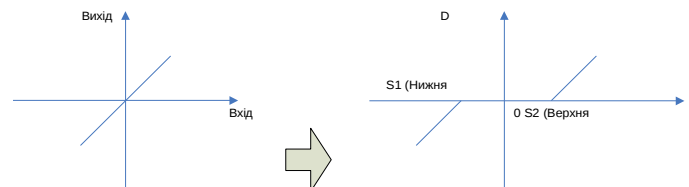
Опис функції

Ви можете визначити, чи знаходиться вхідне значення, задане у S3, у межах діапазону зони нечутливості, заданої S1 та S2, щоб керувати значенням та зберігати його в D. У цьому випадку, коли $S3 < S1$, $D = S3-S1$; коли $S3 > S2$, $D = S3-S2$; коли $S1 \leq S3 \leq S2$, $D=0$.

Екземпляр застосунку



Виконання обмеження керування D0-D10 для вмісту D100 та збереження отриманого результату в D1000, коли M1=ON. У цьому випадку $D0(10) \leq D100(30) \leq D10(100)$, і $D1000=30$.



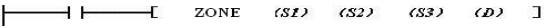
Екземпляр застосунку



DBAND D0 D10 D100 D1000

Виконання керування зоною нечутливості D0-D10 для вмісту D100 та збереження отриманого результату в D1000, коли M1=ON. У цьому випадку $D0 (-100) < D100(30) < D10(100)$, і $D1000=0$.

6.18.3 ZONE: Інструкція з керування зоною інструкція

LAD: 		Придатна модель	VC3 VC5	
		Прапорець впливу біт	Прапорець нуля, прапорець перенесення та прапорець позики	
IL: ZONE (S1) (S2) (S3) (D)		Довжина кроку	9	
Оператор	Тип	Застосовний програмний елемент		Показки

S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
S2	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
S3	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓

- Операнд опис

S1: Від'ємне значення відхилення, яке додається до вхідного значення

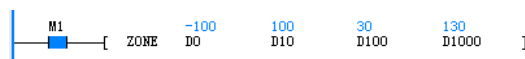
S2: Значення додатного відхилення, що додається до вхідного значення

S3: Вхідне значення для керування за допомогою зонального керування **D:** Номер початкового програмного елемента, у якому зберігається вихідне значення, що було піддане керуванню byzone

- Опис функції

Ви можете оцінити вхідне значення, вказане в S3, плюс значення відхилення, вказане в S1 або S2, щоб керувати та зберігати значення в D. У цьому випадку, коли $S3 < 0$, $D = S3 + S1$; коли $S3 > 0$, $D = S3 + S2$; коли $S3 = 0$, $D = 0$.

- Екземпляр застосування



LD M1

ZONE D0 D10 D100 D1000

Виконання зонального керування D0-D10 над вмістом D100 та збереження отриманого результату в D1000, коли M1=ON. У цьому випадку $D100(30) > 0$, $D1000 = D100(30) + D10(100)$, і $D1000 = 130$.

6.18.4 SCL: Інструкція з налаштування координат

LAD:		Придатна модель		VC3 VC5											
		Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики											
IL: SCL (S1) (S2) (S3) (D)		Довжина кроку		7											
Опера-та	Тип	Застосовний програмний елемент													Пока-жчи-к ng
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
S2	INT								D				V	R	✓
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓

- Операнд опис

S1: Вхідне значення, за яким встановлюються координати, або номер програмного елемента, що зберігає введені значення.

S2: Номер програмного елемента таблиці перетворення для запуску, що використовується для встановлення координат.

D: Номер програмного елемента, у якому зберігається вихідне значення, для якого було встановлено координати.

- Опис функції

1. Встановлення координат для вхідних значень, визначених S1, відповідно до заданих характеристик

перетворення, та збереження отриманих результатів у номерах програмних елементів, визначених D.

2. Перетворення для встановлення координат виконується на основі таблиці даних, що зберігається в програмних елементах, початковий програмний елемент яких задається через S2. Однак, якщо вихідні дані не є цілим числом, дані виводяться після округлення першого десяткового знака.

3. Координати встановлюються через таблицю перетворення.

Точки координат		S2
Пункт 1	X координата	S2+1
	Y координата	S2+2
Пункт 2	X координата	S2+3
	Y координата	S2+4
...	...	...
Точка n (Кінець)	X координата	S2+2n-1
	Y координата	S2+2n

- Примітка

1. Дані таблиці даних мають бути впорядковані за зростанням. Якщо лише частина даних не впорядкована за зростанням і виявлення розпочинається з кнопки, операція перед цією частиною все одно виконується;
2. S1 має бути в межах діапазону, встановленого даними таблиці;

- Екземпляр застосунку



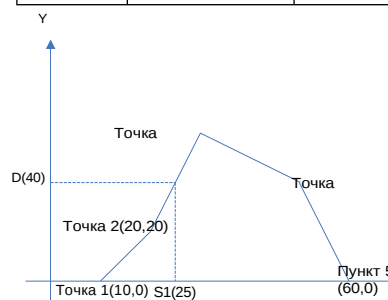
LD M1

SCL D10 D100 D1000

Setting the coordinates on the content of D10, and збереження отриманих результатів у D1000, коли M1=ON.

Точки координат		D100	5
Пункт 1	X координата	D101	10
	Y координата	D102	0
Пункт 2	X координата	D103	20
	Y координата	D104	20
Пункт 3	X координата	D105	30
	Y координата	D106	60

Пункт 4	X координата	D107	50
	Y координата	D108	40
Пункт 5	X координата	D109	60
	Y координата	D110	0



6.18.5 SER: Інструкція з пошуку даних інструкція

LAD:	Придатна модель							VC3 VC5													
								Прапорець впливу біт							Прапорець нуля, прапорець перенесення та прапорець позики						
	SER (S1) (S2) (S3) (D)							Довжина кроку													
	Опера-та	Тип	Застосовний програмний елемент												Пока-жчик по						
S1	INT		KnX	KnY	KnM	KnSKnLM	KnSM	D	SD	C	T	V	R	✓							
S2	INT	Складається немає	KnX	KnY	KnM	KnSKnLM	KnSM	D	SD	C	T	V	R	✓							
D	INT			KnY	KnM	KnSKnLM		D	SD	C	T	V	R	✓							
S3	INT	Складається немає						D				V	R								

Операнд опис

S1: Номер початкового програмного елемента для пошуку. **S2:** Еталонне значення або номер програмного елемента, у якому зберігається цільове значення для пошуку.

D: Номер початкового програмного елемента, у якому зберігається кількість одиниць даних, позиція максимального значення та позиція мінімального значення. **S3:** Кількість одиниць даних для пошуку ($1 \leq S3 \leq 256$).

Опис функції

- Пошук даних S3, починаючи з S1, пошук тих самих даних, що й S2, та збереження отриманих результатів у D-D+4.
- Коли існують однакові дані, кількість однакових даних, а також позиції початкового/кінцевого значення, мінімального та максимального значень зберігаються у 5 програмних елементах, починаючи з D.
- Якщо однакові дані відсутні, перші 3 програмні елементи зберігають 0, а інші два такі самі, як вище.

Екземпляр застосунку



LD M1

SER D0 D10 D100 D1000 8

Пошук вмісту 8 одиниць, починаючи з D10, та збереження результатів пошуку у 5 одиницях, починаючи з D1000, коли M1=ON.

Значення елемента для пошуку S1	Значення	Значення елемента для порівняння	Дані положення	Результат пошуку D	Значення
D10	100	100	0	D1000	3
D11	78		1	D1001	0
D12	92		2	D1002	7
D13	100		3	D1003	5
D14	110		4	D1004	6
D15	-20		5		
D16	145		6		
D17	100		7		

6.19 Рядок символів інструкції

6.19.1 STRADD: Комбінація рядків інструкція

LAD:	Придатна модель								VC3 VC5							
									Прапорець впливу біт Прапорець нуля, прапорець перенесення та прапорець позики							
IL:	STRADD (S1)			(S2) (D)			Довжина кроку									
	Опера та	Тип	Застосовний програмний елемент												Пока жчи к по	
	S1	INT	Рядок	KnX	KnY	KnM	KnSKnLM	KnSM	D	SD	C	T	V	R	√	
	S2	INT	Рядок	KnX	KnY	KnM	KnSKnLM	KnSM	D	SD	C	T	V	R	√	
	D	INT			KnY	KnM	KnSKnLM		D	SD	C	T	V	R	√	

- Опис операнда **S1**:

Перша одиниця

рядка **S2**: Друга

одиниця рядка

D: Блок зберігання рядків після підключення

- Опис функції

1. Підключення рядкових елементів, починаючи з S1 та S2, і збереження їх у програмних елементах, починаючи з D, коли потік енергії є дійсним;

2. Комбінація рядків вказує на те, що перший символ рядка S2 з'єднаний з останнім символом рядка S1, а кінцевий ідентифікатор S1string ігнорується.

3. Дійсні дані рядкового елемента вказують на дані, що починаються із зазначеного програмного елемента для рядкового елемента, до перших даних, у яких

виявлено "00H".

4. Якщо кількість підключених символів є непарною, '00H' додається до старшого байта програмного елемента останнього символу. Якщо вона парна, '0000H' додається до наступного елемента програмного елемента останнього символу.

- Примітка

1. Коли S1 та S2 визначають рядок символів, дозволено максимум 32 символи. Кома та подвійні лапки представляють роздільники у програмному забезпеченні головного комп'ютера, тому цей символ не може бути розпізнаний програмним забезпеченням головного комп'ютера;
2. Якщо '00H' збережено як у S1, так і в S2, тоді '0000H' можна додати безпосередньо до D;
3. Коли адреси м'яких елементів рядкового блоку S1 та D або S2 та D перетинаються, система повідомляє про помилку "The value of

the Instruction operand is illegal".

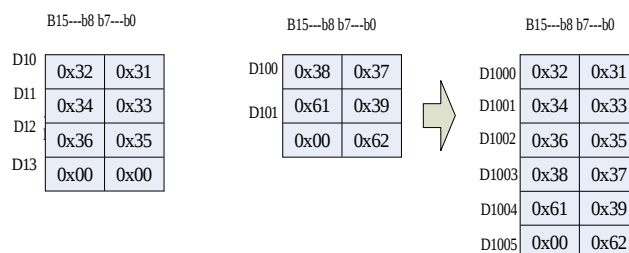
4. Якщо '00H' не з'являється у відповідному діапазоні програмних елементів одиниці рядка, що починається з S1 або S2, система повідомляє про помилку "Номер елемента операнда інструкції поза межами діапазону".

- Екземпляр застосування

LD M1

STRADD D10 D100 D1000

З'єднання рядкового блоку, що починається з D10, та рядкового блоку, що починається з D100, і збереження отриманого результату в блоці, що починається з D1000, коли M1=ON.



6.19.2 STRLEN: Інструкція для визначення довжини рядка

LAD: 									Придатна модель		VC3 VC5						
									Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики						
IL: STRLEN (S) (D)									Довжина кроку		5						
Опера та		Тип		Застосовний програмний елемент												Показчик	
S		INT			KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
D		INT				KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓

- Операнд опис

S: Рядок пристрій

D: Довжина рядка одиниці

- Опис функції

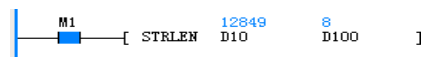
1. Визначення довжини рядка S та збереження отриманого значення у D, коли потік енергії є дійсним.

2. Дійсні дані рядкового елемента вказують на дані, що починаються із зазначеного програмного елемента для рядкового елемента, до перших даних, у яких виявлено "00H".

- Примітка

Коли '00H' не з'являється у відповідному діапазоні програмних елементів одиниці рядка, що починається з S, система повідомляє про помилку "Номер елемента операнда інструкції поза межами діапазону".

- Екземпляр застосунку



LD M1

STRLEN D10 D100

Визначення довжини рядка, що починається з D10, та збереження отриманого результату в D100, коли M1=ON.

6.19.3 STRRIGHT: Інструкція для зчитування рядка зправа

LAD: 								Придатна модель		VC3 VC5							
								Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики							
IL: STRRIGHT (S1) (D) (S2)								Довжина кроку		7							
Опера та	Тип	Застосовний програмний елемент													Пока жчи к по		
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓		
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓		
S2	INT	Конста нта мура ха							D				V	R			

- Операнд опис

S1: Рядок одиниці

D: Збереження видобутого рядка одиниця

S2: Кількість витягнутих символів

- Опис функції

1. Вилучення символів S2, починаючи з останнього дійсного символу рядка S1 (не

враховуючи '00H'), та їх збереження в м'якому елементі, починаючи з D, коли потік енергії є дійсним;

2. Збереження "00H" у програмному елементі D, коли S2 дорівнює нулю;

3. Додавання 00H до старшого байта програмного елемента, що містить останній символ, якщо кількість вилучених символів є непарною. Додавання 0000H до наступного елемента програмного елемента, що містить останній символ, якщо вона є парною.

4. Дійсні дані рядкового елемента вказують на дані, що починаються із заданого програмного елемента для рядкового елемента, до перших даних, у яких виявлено "00H".

- Примітка

1. Коли '00H' не з'являється у відповідному діапазоні програмних елементів одиниці рядка, що починається з S1, система повідомляє про помилку "Номер елемента операнда інструкції поза межами діапазону".

2. S2 більше або дорівнює 0.

3. S2 має бути меншим або дорівнювати кількості символів у рядку S1.

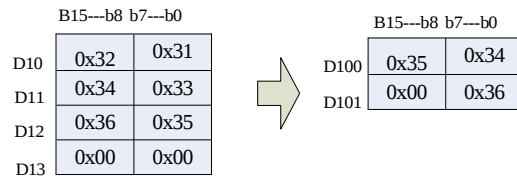
● Екземпляр застосунку



LD M1

STRRIGHT D10 D100 3

Вилучення 3 символів з правого боку рядка, що починається з D10, та їх збереження в блоці, що починається з D100, коли M1=ON.



6.19.4 STRLEFT: Інструкція для зчитування рядка з лівого боку

LAD: 								

● Операнд опис

S1: Рядок одиниць

D: Збереження видобутого рядка одиниць

S2: Кількість вилучених символів

● Опис функції

1. Вилучення S2 символів праворуч, починаючи з лівого боку рядка S1, та їх збереження у програмному елементі, починаючи з D, коли потік енергії є дійсним;

2. Збереження "00H" у програмному елементі D, коли S2 дорівнює нулю;

3. Додавання 00H до старшого байта програмного елемента, що містить останній символ, якщо кількість вилучених символів є непарною. Додавання 0000H до наступного елемента програмного елемента, що містить останній символ, якщо вона є парною.

4. Дійсні дані рядкового елемента вказують на дані, що починаються із зазначеного програмного елемента для рядкового елемента

до перших даних, у яких "00H" виявлено.

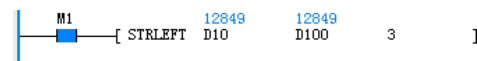
● Примітка

1. Коли '00H' не з'являється у відповідному діапазоні м'яких елементів одиниці рядка, що починається з S1, система повідомляє про помилку "Номер елемента операнда інструкції поза межами діапазону".

2. S2 більше або дорівнює 0.

3. S2 має бути меншим або дорівнювати кількості символів у рядку S1.

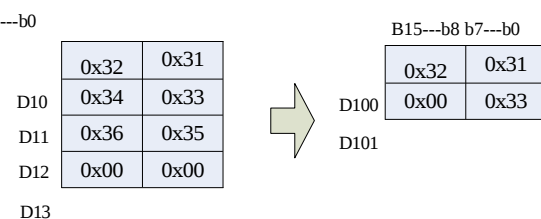
● Екземпляр застосунку



LD M1

STRLEFT D10 D100 3

Вилучення 3 символів з лівого боку рядка, що починається з D10, і збереження їх у блоці, що починається з D100, коли M1=ON.



6.19.5 STRMIDR: Інструкція для зчитування будь-яких символів рядка

LAD:									Придатна модель	VC3 VC5						
[STRMIDR (S1) (D) (S2)]									Прапорець впливу біт	Прапорець нуля, прапорець перенесення та прапорець позики						
IL: STRMIDR (S1) (D) (S2)									Довжина кроку	7						
Оператор	Тип	Застосовний програмний елемент														Показники
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓	
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓	
S2	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓	

- Операнд опис

S1: Рядок одиниця

D:Витягнутий рядок одиниця

S2: Початкова позиція символів, які потрібно витягти

S2+1: Кількість символів для вилучення (n)

- Опис функції

1. Вилучення n символів даних з рядка S1, починаючи з символу S2, та їх збереження у програмному елементі, починаючи з D, коли потік енергії є дійсним;

2. Додавання 00H до старшого байта програмного елемента, що містить останній символ, якщо кількість вилучених символів є непарною. Додавання 0000H до наступного елемента програмного елемента, що містить останній символ, якщо вона є парною.

3. Дійсні дані рядкової одиниці вказують на дані, що починаються із зазначеного програмного елемента для рядкової одиниці, до перших даних, у яких виявлено "00H".

4. Виконання жодної дії, коли n дорівнює 0.

5. Вилучення всіх символів даних рядка S1 та їх збереження у програмному елементі, починаючи з D, коли n дорівнює -1.

- Примітка

1. S2 має бути меншим або дорівнювати кількості символів у рядку S1.

2. n більше за 2.

3. S2 більше або дорівнює 1.

4. Коли "00H" не з'являється у відповідному діапазоні м'яких елементів одиниці рядка, що починається з S1, система повідомляє про помилку "Номер елемента операнда інструкції поза межами діапазону".

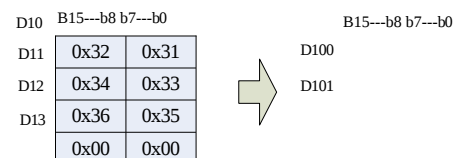
- Екземпляр застосунку



LD M1

STRMIDR D10 D100 D0

Зчитування D1 (D1=3) символів з D0 (D0=2) рядкового блоку, що починається з D10, та їх збереження у блоці, що починається з D100, коли M1=ON.



0x33	0x32
0x00	0x34

6.19.6 STRMIDW: Інструкція щодо заміни будь-яких символів рядка

LAD:									Придатна модель	VC3 VC5						
									Прапорець впливу біт	Прапорець нуля, прапорець перенесення та прапорець позики						
IL: STRMIDW (S1) (D) (S2)									Довжина кроку	7						
Оператор	Тип	Застосовний програмний елемент														Показник
S1	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓	
D	INT			KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓	
S2	INT		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓	

● Операнд опис

S1: Рядковий елемент, що має використовуватися**D:** Рядок, який потрібно замінити**S2:** Початкова позиція заміни

S2+1: Кількість символів, що підлягають заміні (n)

- Опис функції
 - Заміна n символів даних, починаючи з символу S2 рядка D, на n символів даних рядка S1, коли потік енергії є дійсним;
 - Дійсні дані рядкового елемента вказують на дані, що починаються із зазначеного програмного елемента для рядкового елемента, до перших даних, у яких виявлено "00H".
 - Виконання жодної дії, коли n дорівнює 0.
 - Коли n дорівнює -1, вміст до кінцевого символу даних, вказаного S1, зберігається в програмних елементах після програмного елемента, вказаного D.
- Примітка
 - S2 має бути меншим або дорівнювати кількості символів у рядку S1.
 - n більше за 2.
 - S2 більше або дорівнює 1.

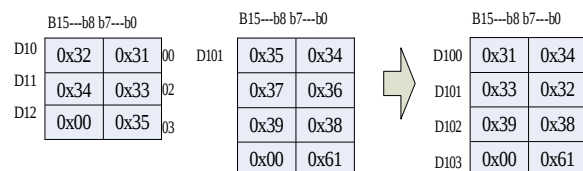
4. Коли кількість символів для заміни перевищує останній символ одиниці рядка, що починається з D, зайві символи не зберігаються.

5. Коли '00H' не з'являється у відповідному діапазоні програмних елементів одиниці рядка, що починається з S1 та D, система повідомляє про помилку "Номер елемента операнда інструкції поза межами діапазону".

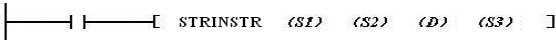
- Екземпляр застосунку



Заміна D1 (D1=3) символів після D0 (D0=2) символів рядкової одиниці, що починається з D100, на перші D1 (D1=3) символи рядкової одиниці, що починається з D10, коли M1=ON.



6.19.7 STRINSTR: Пошук рядка інструкція

LAD: 								Придатна модель		VC3 VC5					
								Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики					
IL: STRINSTR (S1) (S2) (D) (S3)								Довжина кроку		9					
Оператор	Тип	Застосовний програмний елемент													Показник
S1	INT	Рядок							D	SD	C	T	V	R	✓
S2	INT								D	SD	C	T	V	R	✓
D	INT								D	SD	C	T	V	R	✓
S3	INT	Константа муреха							D				V	R	

- Операнд опис

S1: Рядок, у якому виконується пошук

S2: Джерело пошуку

D: Результати пошуку

S3: Початкова позиція пошуку
- Опис функції
 - Пошук того самого рядка, що й рядок S1, починаючи з символу S3 рядка S2, та збереження інформації про позицію рядка результатів пошуку в D, коли потік енергії є дійсним.
 - Збереження "0" у D, коли немає узгодженого рядка.
 - Жодна операція не виконується, якщо позиція S3,

з якої починається пошук, є від'ємним числом або "0".

4. Дійсні дані рядкового елемента вказують на дані, що починаються із зазначеного програмного елемента для рядкового елемента, до перших даних, у яких виявлено "00H".

- Примітка
 1. Якщо '00H' не з'являється у відповідному діапазоні програмних елементів одиниці рядка, що починається з S1 або S2, система повідомляє про помилку "Номер елемента операнда інструкції поза межами діапазону".
 2. S3 менше або дорівнює кількості символів у рядку S2.
 3. Коли S1 визначає рядок символів, дозволено максимум 32 символи. Кома та подвійні лапки є роздільниками у програмному забезпеченні головного комп'ютера, тому цей символ не може бути розпізнаний програмним забезпеченням головного комп'ютера.
 4. Коли S1 є порожнім рядком символів ('00H'), визначається позиція рядка символів S2 '00H' (якщо S2 містить парну кількість символів, це позиція першого '00H').
- Екземпляр застосунку


```

M1 ──┬── [STRINSTR "45" D10 D100 2 ]

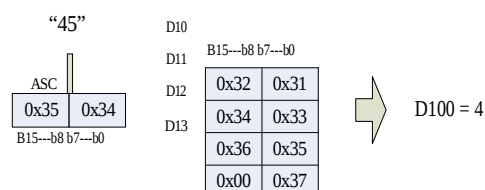
```

LD M1

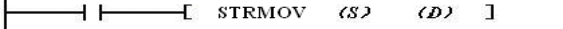
STRINSTR "45" D10 D100 2

Пошук того самого рядка, що й "45", починаючи з другого символу одиниці символного рядка, починаючи з

D10, та збереження результатів пошуку в D100 коли M1=ON.



6.19.8 STRMOV: Інструкція з передачі рядка instruction

LAD:									Придатна модель		VC3 VC5						
									Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики						
IL: STRMOV (S) (D)									Довжина кроку		5						
Опера та		Тип		Застосовний програмний елемент												Показники	
S		INT		Рядок	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	R	✓
D		INT				KnY	KnM	KnS	KnLM		D	SD	C	T	V	R	✓

- Опис операнда **S**:
Одиниця вихідного рядка **D**: Одиниця призначення
- Опис функції
 - Перенесення всіх даних S рядкового модуля, включаючи '00H', до елементного модуля, починаючи з D.
 - Дійсні дані рядкової одиниці вказують на дані, що починаються із заданого програмного елемента для рядкової одиниці, до перших даних, у яких виявлено "00H".
- Примітка
 - Коли '00H' не з'являється у відповідному діапазоні програмних елементів рядкової одиниці, що починається з S, то Система повідомляє про помилку "Кількість елементів операнда інструкції поза межами діапазону".
 - Коли кількість символів одиниці рядка S є парною, '00H' зберігається в LSB, і як MSB, так і LSB відповідної позиції в D зберігають '00H'.

3. Коли S1 визначає рядок символів, дозволено максимум 32 символи. Кома та подвійні лапки є роздільниками в програмному забезпеченні головного комп'ютера, тому ці символи не можуть бути розпізнані програмним забезпеченням головного комп'ютера.

- Екземпляр застосунку

```

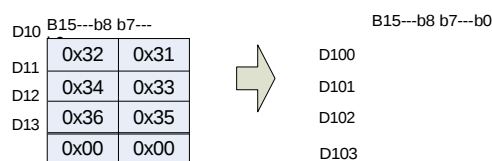
M1 ──┬── [STRMOV D10 D100 ]

```

LD M1

STRMOV D10 D100

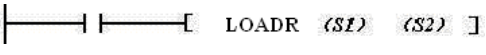
Передача даних рядка символів, починаючи з D10, до пристрою, починаючи з D100, коли M1=ON.



0x32	0x31
0x34	0x33
0x36	0x35
0x00	0x00

6.20 Інструкції з реєстрації файлів розширення instructions

6.20.1 LOADR: Інструкція для зчитування даних з файлу розширення регістр

LAD: 								Застосовна модель		Прапорець нуля, прапорець перенесення та прапорець позики							
								Під впливом прапорцевий біт									
IL: LOADR (S1) (S2)								Довжина кроку		5							
Опера та	Тип	Застосовний програмний елемент														Пока жчи к ng	
S1	INT														R	✓	

S2	INT	Складається немає								D						

- Операнд опис

S1: Блок м'яких елементів регістра розширення для зберігання даних

S2: Кількість даних, що були зчитані ($1 \leq S2 \leq 1024$)

- Опис функції

Зчитування даних S2 з S1 файлового регістра розширення, що зберігається у блоці пам'яті, до м'яких елементів, починаючи з S1 регістра розширення у PLC;

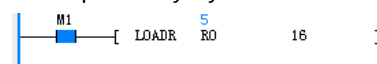
- Примітка

1. Жодна операція не виконується, коли для S2 вказано 0.

2. Якщо накопичувач не підключено, при виконанні цієї інструкції система повідомляє про відсутність карти пам'яті.

3. Коли S2=1024, час виконання інструкції становить приблизно 80ms. Вам потрібно правильно встановити час сторожового таймера у практичних застосуваннях.

- Екземпляр застосунку



LD M1

LOADR R0 16

Зчитування 16 даних, починаючи з R0 області резервного копіювання карти пам'яті R, та збереження цих даних у 16 одиницях програмних елементів, починаючи з R0, коли M1=ON.

6.20.2 SAVER: Інструкція для запису даних у файл розширення регістр

LAD:										Застосовна модель						
										Під впливом прапорцевий біт	Прапорець нуля, прапорець перенесення та прапорець позики					
IL: SAVER (S1) (S2) (D)										Довжина кроку	7					
Операнда	Тип	Застосовний програмний елемент														Показник
S1	INT														R	✓
S2	INT	Постійна														
D	INT									D						✓

- Операнд опис

S1: Блок м'яких елементів регістра розширення для зберігання даних (можна вказати лише початкову адресу м'яких елементів у сегменті)

S2: Кількість даних, записаних за один робочий цикл ($1 \leq S2 \leq 2048$)

D: Зберігання кількості даних, що були записані

- Опис функції

1. Запис даних з S1 – S1+2047 у регістрі розширення до того самого пристрою в блоці пам'яті за 2048/S2 (якщо 2048 не ділиться на S2, частка збільшується на 1) робочий цикл.

2. Збереження кількості даних, що були записані в D під час процесу запису.

3. Встановлення прапорця завершення виконання інструкції SM189, коли виконання інструкції завершується.

- Примітка

1. Коли $S2=2048$, час виконання інструкції становить близько кількох сотень мс. Вам необхідно правильно встановити час сторожового таймера.

2. Вам необхідно виконати інструкцію INITER або INITR для ініціалізації керованих сегментів перед запуском інструкції SAVER. Якщо дані, записані в регістр файлу розширення, не збігаються з даними регістра розширення, система повідомляє, що операція на карті пам'яті некоректна.

3. Якщо в $S2$ задано значення 0, воно виконується відповідно до $S2 = 2048$.

4. Дані в S1 мають бути номером початкового програмного елемента в сегменті регістрів розширення. Номер початкового програмного елемента в кожному сегменті наведено нижче.

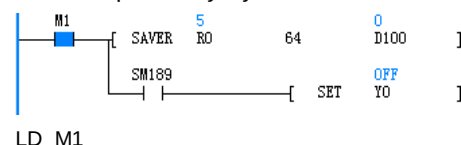
Номер сегмента	Кількість the запуск м'якого елемента	Діапазон запису до карта пам'яті R зона резервного копіювання	Номер сегмента	Кількість the запуск м'якого елемента	Діапазон запису на карту пам'яті R резервна область

0	R0	R0–R2047	8	R16384	R16384–R18431
1	R2048	R2048–R4095	9	R18432	R18432–R20479
2	R4096	R4096–R6143	10	R20480	R20480–R22527
3	R6144	R6144–R8191	11	R22528	R22528–R24575
4	R8192	R8192–R10239	12	R24576	R24576–R26623
5	R10240	R10240–R12287	13	R26624	R26624–R28671
6	R12288	R12288–R14335	14	R28672	R28672–R30719
7	R14336	R14336–R16383	15	R30720	R30720–R32767

5. Якщо пам'ять не підключено, система повідомляє про відсутність карти пам'яті.

6. Коли виконується апаратний захист від запису для карти пам'яті, система повідомляє про помилку операції з картою пам'яті.

● Екземпляр застосунку



SAVER R0 64 D100 LD

SM189

SET Y0

Збереження даних 2048 програмних елементів, починаючи з R0, у 2048 одиницях, починаючи з ER0, та збереження кількості одиниць, які вже збережені в D100, коли M1=ON. Коли виконання інструкції завершено, SM189=ON, а Y0=ON.

6.20.3 INTR: Інструкція з ініціалізації регістра розширення register

LAD: 								Придатна модель		Прапорець нуля, прапорець перенесення та прапорець позики							
								Прапорець впливу біт									
IL: INTR (S1) (S2)								Довжина кроку		5							
Опера та	Тип	Застосовний програмний елемент												Пока жчи к п			
S1	INT														R	✓	
S2	INT	Склад ається немає															

● Операнд опис

S1: Одиниці регістра розширення та регістра файлу розширення, що підлягають ініціалізації (можна вказати лише адресу початкового програмного елемента в сегменті)

S2: Сегменти регістра розширення та регістра файлу розширення, що підлягають ініціалізації (S2=1)

● Опис функції

1. Ініціалізація регістрів розширення S2 та регістрів файлів розширення, починаючи з S1, і запис початкового значення 0xFFFF.

2. Ініціалізація виконується за допомогою сегментів.

● Примітка

1. S1 має вказувати номер початкового програмного елемента в сегменті. Номер початкового програмного елемента в кожному сегменті наведено нижче.

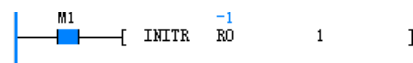
Номер сегмента	Номер програмного елемента пуску	Діапазон запису на карту пам'яті R резервна копія зона	Номер сегмента	Номер програмного елемента пуску	Діапазон запису на карту пам'яті R резервна область
0	R0	R0–R2047	8	R16384	R16384–R18431
1	R2048	R2048–R4095	9	R18432	R18432–R20479
2	R4096	R4096–R6143	10	R20480	R20480–R22527
3	R6144	R6144–R8191	11	R22528	R22528–R24575
4	R8192	R8192–R10239	12	R24576	R24576–R26623
5	R10240	R10240–R12287	13	R26624	R26624–R28671
6	R12288	R12288–R14335	14	R28672	R28672–R30719
7	R14336	R14336–R16383	15	R30720	R30720–R32767

2. Якщо карта пам'яті не використовується, ініціалізація регістра файлу розширення не виконується.

3. Якщо карту пам'яті підключено, і для неї активовано апаратний захист від запису, система повідомляє про помилку операції з картою пам'яті.

4. Інструкція може ініціалізувати лише один сегмент для кожного виконання. Коли використовується карта пам'яті, час операції ініціалізації кожного сегмента становить приблизно 100ms. У фактичному застосуванні вам потрібно правильно встановити час сторожового таймера.

● Екземпляр застосунку



LD M1 INITR

R0 1

Ініціалізація регістрів розширення R-R2047 у сегменті 0, коли M1=ON. У разі використання карти пам'яті також ініціалізуються регістри файлів розширення ER-ER2047.

6.20.4 LOGR: Інструкція для входу в систему на розширенні реєстр

LAD: 								Придатна модель		Прапорець нуля, прапорець перенесення та прапорець позики					
								Прапорець впливу біт							
IL: LOGR (S1) (S2) (S3) (S4) (D)								Довжина кроку		11					
Опера-та	Тип	Застосовний програмний елемент													Показчик перенесення
S1	INT								D		C	T			✓
S2	INT	Складається немає							D						
S3	INT													R	
S4	INT	Складається немає													
D	INT								D						✓

● Операнд опис

S1: Об'єктний блок, що виконує вхід

S2: Кількість зареєстрованих об'єктних одиниць (1–1024)

S3: Адреса початкового елемента програмного модуля, що використовується для входу **S4:** Кількість сегментів програмних елементів, що використовуються для входу (1–16)

D: Кількість зареєстрованих даних

● Опис функції

1. Безперервний вхід у програмні елементи S2, починаючи з S1 до сегментів S4 регістру розширення, а також регістру файлу розширення, починаючи з S3, заповнюються, коли потік енергії є дійсним.

2. Виконання входу під час кожного циклу.

3. Збереження кількості даних входу в D.

● Примітка

1. Під час використання карти пам'яті необхідно ініціалізувати сегменти для реєстрації. Якщо дані, записані у файловий регістр розширення, не збігаються з даними, що починаються з S1, система повідомляє про помилку операції з картою пам'яті. У такому разі ви можете ініціалізувати сегменти для реєстрації по черзі за допомогою команди INITR або INITER, або ініціалізувати всі елементи ER через інструкцію очищення меню фонові карти пам'яті (необхідно одночасно вибрати користувацьку програму, глобальні змінні, блок даних та системний блок). Щоб уникнути втрати даних, перед ініціалізацією необхідно створити резервну копію вмісту файлу карти пам'яті за допомогою фонового програмного забезпечення.

2. S3 потребує вказання номера початкового програмного елемента в сегменті. Номер початкового програмного елемента в кожному сегменті наведено нижче.

Номер сегмента	Номер програмного забезпечення для запуску елемента	Діапазон запису на карту пам'яті R резервна копія зона	Номер сегмента	Номер програмного елемента пуску	Діапазон запису на карту пам'яті R резервна область
0	R0	R0–R2047	8	R16384	R16384–R18431
1	R2048	R2048–R4095	9	R18432	R18432–R20479
2	R4096	R4096–R6143	10	R20480	R20480–R22527
3	R6144	R6144–R8191	11	R22528	R22528–R24575
4	R8192	R8192–R10239	12	R24576	R24576–R26623
5	R10240	R10240–R12287	13	R26624	R26624–R28671
6	R12288	R12288–R14335	14	R28672	R28672–R30719
7	R14336	R14336–R16383	15	R30720	R30720–R32767

3. Якщо встановлено апаратний захист пам'яті, система повідомляє, що операція з картою пам'яті є некоректною.

4. Формат даних для входу

S3–S3+S2-1	Адреса зберігання перших даних, що увійшли в систему S3 – S3+S2-1	D=S2	Область запису даних 1926×S4	Область зберігання даних для входу
S3+S2–S3+2S2-1	Адреса зберігання других даних, що увійшли в систему S3 – S3+S2-1	D=2S2		
S3+2S2–S3+3S2-		D=3S2		
.....				
S3+1926×S4-1 – S3+2048×S4-1	Запис у зону керування розташуванням area При використанні області запису даних одного слова послідовність змінюється з ON на OFF, починаючи з 0 бітів S3+1926×S4-1, а наступний програмний елемент S3+1926×S4 використовується, коли S3+1926×S4-1 — це все OFF.	Зона керування 122 × S4		

● Екземпляр застосунку

LD M1



LOGR D0 5 R0 1 D100

Реєстрація даних пристрою D0–D4 у блоці R0–R2047 сегмента 0 та запис кількості зареєстрованих даних у D100, коли M1=ON.

6.20.5 INITER: Інструкція з ініціалізації файлу розширення реєстру

LAD:	Придатна модель																	
											Прапорець впливу біт		Прапорець нуля, прапорець перенесення та прапорець позики					
IL: INITER (S1) (S2)	Довжина кроку																	
	Опера- та	Тип	Застосовний програмний елемент										Пока- жи к ng					
	S1	INT												R	√			
	S2	INT	Складається немає															

- Операнд опис

S1: Той самий блок регістрів розширення, що має спільну адресу з блоком файлових регістрів розширення, які необхідно ініціалізувати (можна вказати лише початкову адресу регістру розширення в сегменті)

S2: Сегменти регістра розширення та регістра файлу розширення, що підлягають ініціалізації (S2=1)
- Примітка
- Опис функції
 1. Ініціалізація сегментів S2 файлових регістрів розширення, починаючи з S1 у блоці пам'яті, та запис значення 0xFFFF у них.
 2. Ініціалізація виконується за допомогою сегментів.

1. S1 має вказувати номер початкового програмного елемента в сегменті. Номер початкового програмного елемента в кожному сегменті наведено нижче.

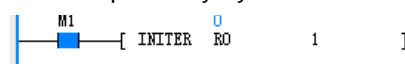
Номер сегмента	Кількість the запуск м'якого елемента	Діапазон запису до карта пам'яті R зона резервного копіювання	Номер сегмента	Номер програмного елемента пуску	Діапазон запису до карта пам'яті R зона резервного копіювання
0	R0	R0–R2047	8	R16384	R16384–R18431
1	R2048	R2048–R4095	9	R18432	R18432–R20479
2	R4096	R4096–R6143	10	R20480	R20480–R22527
3	R6144	R6144–R8191	11	R22528	R22528–R24575
4	R8192	R8192–R10239	12	R24576	R24576–R26623
5	R10240	R10240–R12287	13	R26624	R26624–R28671
6	R12288	R12288–R14335	14	R28672	R28672–R30719
7	R14336	R14336–R16383	15	R30720	R30720–R32767

2. Якщо встановлено апаратний захист пам'яті, система повідомляє, що операція з картою пам'яті є некоректною.

3. Якщо накопичувач не підключено, система повідомляє про відсутність карти пам'яті.

4. Інструкція може лише ініціалізувати один сегмент для кожного виконання, і час операції ініціалізації кожного сегмента становить близько 100 мс. У фактичному застосуванні вам потрібно правильно встановити час сторожового таймера.

● Екземпляр застосунку



LD M1 INITER

R0 1

Ініціалізація файлу розширення реєструє ER0–ER2047 у сегменті 0, коли M1=ON.

6.21 Інструкції з позиціонування

6.21.1 ZRN: Інструкція повернення до нуля інструкція

LAD: 										Застосовна модель		VC1S VC1 VC2 VC3					
										Прапорець впливу біт							
IL: ZRN (S1) (S2) (S3) (D)										Довжина кроку		11					
Робота d		Тип	Застосовний програмний елемент													Індексація	
S1	DINT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
S2	DINT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓	
S3	BOOL		X	Y	M	S											
D	BOOL			Y													

- Операнд опис

S1: Швидкість повернення до нуля, що визначає швидкість, з якої починається повернення до нуля. 32-бітна інструкція.

S2: Швидкість повзання, знижена швидкість після того, як сигнал кінцевого вимикача наближення (DOG) перемикається в ON.

S3: DOG для введення в X елемент.

Коли елемент, відмінний від вхідного реле (X) є зазначено, зміщення позиції нульової точки збільшується через вплив PLC циклу обчислень.

D: Початкова адреса виходу високошвидкісних імпульсів.

- Опис функції

Беручи Y0 як приклад. Коли SM276 сигнал скидання є дійсним, а елементи, визначені SM281 сигналом скидання, є недійсними, Y10 є вихідним портом сигналу скидання. Коли елементи, визначені SM281 сигналом скидання, є недійсними, Y (N), визначений SD175, є вихідним портом сигналу скидання.

- Примітка

1. Оскільки інструкція ZRN не здатна автоматично виконувати пошук DOG, необхідно розпочати операцію повернення в нуль далі за передній край пристрою виявлення DOG.

- Часова діаграма послідовності

1. Під час процесу повернення до нуля значення регістра поточного значення зменшується.

2. The min. output pulse frequency that can be outputted actually, is determined by the formula below:

$$F_{\text{min\_приск}} = \sqrt{\frac{F_{\text{макс.}} \times}{500}}$$

У наведеній вище формулі, $F_{\text{макс.}}$ вказує на макс. швидкість

поки T вказує час розгону/гальмування

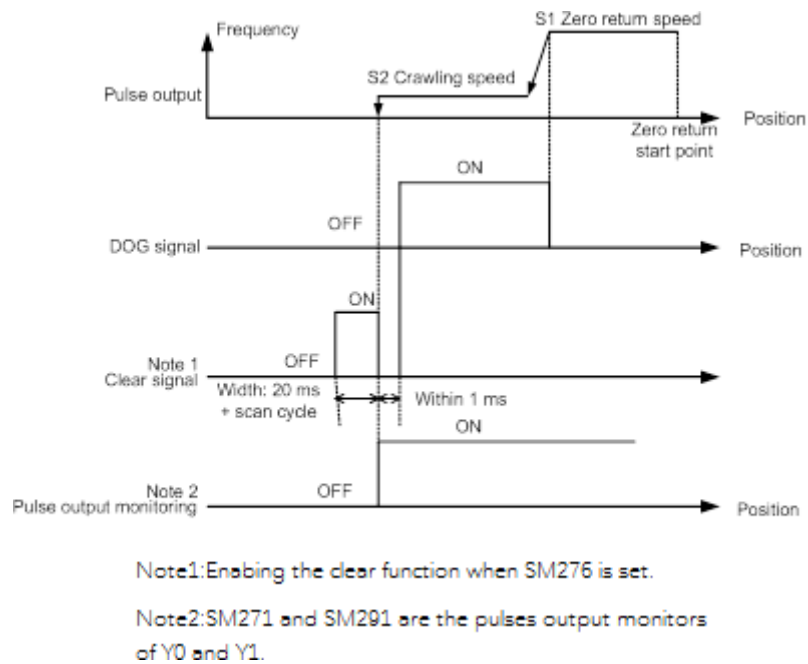
з одиницею вимірювання мс. Результат обчислення

$F_{\text{min\_acc}}$ є граничним значенням мін. вихідної частоти.

3. Для частоти імпульсного виходу розрахована частота виводиться, навіть якщо призначено значення, менше за розраховане. Частота початкової ділянки ACC та кінцевої частини DEC не може бути меншою за вищевказаний результат розрахунку. Якщо макс. швидкість нижча за вищевказаний результат розрахунку, імпульсний вихід відсутній.

4. Швидкість повзання має бути більшою за нуль і меншою за одну десяту від максимальної швидкості.

5. Детальну інформацію див. у розділі 11 "Посібник користувача для функції позиціонування".



6.21.2 PLSV:Інструкція щодо виходу імпульсів зі змінною швидкістю instruction

LAD:	Застосовна модель										VC1S VC1 VC2 VC3				
	[PLSV (S) (D1) (D2)]										Прапорець впливу біт				
	PLSV (S) (D1) (D2)										Довжина кроку 8				
Робота	Тип	Застосовний програмний елемент												Індексація	
S	DINT	Константа	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	V	R	√	
D1	BOOL	мураха		Y											
D2	BOOL			Y	M	S									

- Операнд опис

S: Вихідна частота імпульсів (Гц), 32-бітна інструкція.

D1: Початкова адреса високошвидкісного імпульсного виходу.

D2: Початкова адреса виходу сигналу напрямку обертання. Відповідно до умови додатної/від'ємної величини **S**, він діє наступним чином:

- Коли **S** є додатним: **D2** є ON.
- Коли **S** від'ємне: **D2** — це OFF.

- Опис функції

1. Ви можете вільно змінювати частоту вихідних імпульсів (**S**) навіть у стані високошвидкісного імпульсного виходу.

2. Оскільки під час пуску/зупину немає прискорення або сповільнення, якщо під час пуску або зупину потрібен буфер, рекомендується використовувати інструкцію RAMP для зміни значення вихідної частоти імпульсів (**S**).

3. Під час процесу високошвидкісного імпульсного виходу, коли потік енергії, що керується інструкцією, стає OFF, вихід зупиняється без сповільнення.

4. Якщо моніторинг високошвидкісного імпульсного виходу дорівнює ON, потік енергії, що керується інструкцією, не керується інструкцією знову після того, як потік енергії стає OFF.

5. Напрямок визначається додатним або від'ємним значенням **S**.

- Примітка

1. PLSY, PLS та інструкції позиціонування можуть виводити сигнал через порт високошвидкісних імпульсів. Заборонено використовувати ці інструкції для виводу високошвидкісних імпульсів на один і той самий порт одночасно.

2. Детальну інформацію див. у розділі 11 "Посібник користувача для функції позиціонування".

6.21.3 DRVI: Інструкція з керування відносним положенням інструкція

LAD:	Застосовна модель										VC1S VC1 VC2 VC3									
	Прапорець впливу біт																			
DRVI	(S1)	(S2)	(D1)	(D2)	Довжина кроку										11					
Робота	Тип	Застосовний програмний елемент													Індексація					
S1	DINT	Константа мураха	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C		V		R	√			
S2	DINT	Константа мураха	KnX	KnY	KnM	KnS	KnLM	KnSM		D	SD	C		V		R	√			
D1	BOOL			Y																
D2	BOOL			Y	M	S														

- Операнд опис

S1: Кількість вихідних імпульсів (відносно задана)

S2: Частота вихідних імпульсів (Гц).

D1: Початкова адреса високошвидкісного імпульсного виходу.

D2: Початкова адреса виходу сигналу напрямку обертання. Відповідно до умови додатного/від'ємного значення **S**, він діє наступним чином:

- Коли S1 є додатним: D2 є ON.
- Коли S1 від'ємний: D2 — OFF.

- Опис функції

- S1** відповідає наступним регістрам поточного значення як відносна позиція.

SD162	Поточне значення позиції в позиціонуванні інструкція виведена через Y0 (MSB)
SD163	Поточне значення позиції в позиціонуванні інструкція виведена через Y0 (LSB)
SD182	Поточне значення позиції в позиціонуванні інструкція виведена через Y1 (MSB)
SD183	Поточне значення позиції в позиціонуванні інструкція виведена через Y1 (LSB)
SD202	Поточне значення позиції в позиціонуванні інструкцію виведено через Y2 (MSB)
SD203	Поточне значення позиції в позиціонуванні інструкція виведена через Y2 (LSB)
SD222	Поточне значення позиції в позиціонуванні інструкція виведена через Y3 (MSB)
SD223	Поточне значення позиції в позиціонуванні інструкція виведена через Y3 (LSB)
SD242	Поточне значення позиції в позиціонуванні інструкція виведена через Y4 (MSB)
SD243	Поточне значення позиції в позиціонуванні інструкція виведена через Y4 (LSB)
SD262	Поточне значення позиції в позиціонуванні інструкція виведена через Y5 (MSB)
SD263	Поточне значення позиції в позиціонуванні інструкція виведена через Y5 (LSB)
SD282	Поточне значення позиції в позиціонуванні інструкція виведена через Y6 (MSB)
SD283	Поточне значення позиції в позиціонуванні інструкція виведена через Y6 (LSB)
SD302	Поточне значення позиції в позиціонуванні інструкція виведена через Y7 (MSB)

SD303	Поточне значення позиції в позиціонуванні інструкція виведена через Y7 (LSB)

2. Під час зворотного обертання значення регістра поточного значення зменшується.

3. Напрямок обертання визначається додатним або від'ємним значенням **S1**.

4. Під час виконання інструкції, навіть якщо вміст операндів змінено, ці зміни не можуть бути відображені в поточній операції та набудуть чинності лише під час наступного виконання інструкції.

5. Під час виконання інструкції вихідний сигнал сповільнюється до зупинки, коли потік енергії, що керується інструкцією, стає OFF. Прапорець завершення виконання SM у цей момент не діє.

6. Якщо прапорець високошвидкісного імпульсного виходу дорівнює ON, потік енергії, що керується інструкцією, не керується інструкцією знову після того, як потік енергії стає OFF.

- Примітка

1. Мінімальна вихідна частота імпульсів, яку фактично можна вивести, визначається за наведеною нижче формулою:

$$F_{\min\_acc} = \sqrt{\frac{F_{\max}}{500}}$$

У наведеній вище формулі, $F_{\max}$ вказує макс. швидкість

тоді як T вказує на час розгону/гальмування з одиницею вимірювання мс. Результат обчислення $F_{\min\_acc}$ є граничним значенням мін. вихідної частоти.

2. Для частоти імпульсного виходу розрахована частота виводиться, навіть якщо призначено значення, менше за розраховане. Частота початкової ділянки ACC та кінцевої частини DEC не може бути меншою за вищевказаний результат розрахунку. Якщо макс. швидкість нижча ніж вищезгаданий результат розрахунку, немає імпульсу вихід.

3. Швидкість повзання має бути більшою за нуль і меншою за одну десяту від максимальної швидкості.

4. Детальну інформацію див. у розділі 11 "Посібник користувача для функції позиціонування".

6.21.4 DRVA: Інструкція керування абсолютним положенням інструкція

LAD:	Застосовна модель										VC1S VC1 VC2 VC3				
	Прапорець впливу біт														
DRVA (S1) (S2) (D1) (D2) Довжина кроку 11															
Робота	Тип	Застосовний програмний елемент													Індексація
S1	DINT	Константа мурха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V	R	√
S2	DINT	Константа мурха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V	R	√
D1	BOOL			Y											
D2	BOOL			Y	M	S									

- Операнд опис

S1: Цільова позиція (задана абсолютно)

S2: Частота вихідних імпульсів (Гц), 32-бітна інструкція.

D1: Початкова адреса високошвидкісного імпульсного виходу. Для VC1S можна призначити лише Y0 або Y1.

Вихід PLC має використовувати режим транзисторного виходу. **D2:** Початкова адреса виходу сигналу напрямку обертання. Якщо імпульс передається у прямому напрямку, **D2** дорівнює ON, в іншому випадку **D2** дорівнює OFF.

- Опис функції

1. **S1** відповідає наступним регістрам поточного значення як відносна позиція.

SD162	Поточне значення позиції в позиціонуванні інструкція виведена через Y0 (MSB)
SD163	Поточне значення позиції в позиціонуванні інструкція виведена через Y0 (LSB)
SD182	Поточне значення позиції в позиціонуванні інструкція виведена через Y1 (MSB)
SD183	Поточне значення позиції в позиціонуванні інструкція виведена через Y1 (LSB)
SD202	Поточне значення позиції в позиціонуванні інструкція виведена через Y2 (MSB)
SD203	Поточне значення позиції в позиціонуванні інструкція виведена через Y2 (LSB)
SD222	Поточне значення позиції в позиціонуванні інструкція виведена через Y3 (MSB)
SD223	Поточне значення позиції в позиціонуванні інструкція виведена через Y3 (LSB)
SD242	Поточне значення позиції в позиціонуванні інструкція виведена через Y4 (MSB)
SD243	Поточне значення позиції в позиціонуванні інструкція виведена через Y4 (LSB)
SD262	Поточне значення позиції в позиціонуванні інструкція виведена через Y5 (MSB)
SD263	Поточне значення позиції в позиціонуванні інструкція виведена через Y5 (LSB)
SD282	Поточне значення позиції в позиціонуванні інструкція виведена через Y6 (MSB)
SD283	Поточне значення позиції в позиціонуванні інструкція виведена через Y6 (LSB)
SD302	Поточне значення позиції в позиціонуванні інструкція виведена через Y7 (MSB)

SD303	Поточне значення позиції в позиціонуванні інструкція виведена через Y7 (LSB)

2. Під час зворотного обертання значення регістра поточного значення зменшується.

3. Напрямок обертання визначається додатним або від'ємним значенням **S1**.

4. Під час виконання інструкції, навіть якщо вміст операндів змінено, ці зміни не можуть бути відображені в поточній операції та набудуть чинності лише під час наступного виконання інструкції.

5. Під час виконання інструкції вихідний сигнал сповільнюється до зупинки, коли потік енергії, що керується інструкцією, стає OFF. Прапорець завершення виконання SM у цей момент не діє.

6. Якщо прапорець високошвидкісного імпульсного виходу дорівнює ON, потік енергії, що керується інструкцією, не керується інструкцією знову після того, як потік енергії стає OFF.

- Примітка

1. Мінімальна вихідна частота імпульсів, яку фактично можна вивести, визначається за наведеною нижче формулою:

$$F_{\min\_acc} = \sqrt{\frac{F_{\max} \times}{500}}$$

У наведеній вище формулі, $F_{\max}$ вказує макс. швидкість

тоді як T вказує час розгону/гальмування з одиниця вимірювання мс. Результат чи це ліміт обчислення $F_{\min\_acc}$ значення мін. вихідної частоти.

2. Для частоти імпульсного виходу виводиться розрахована частота, навіть якщо призначено значення, менше за розраховане. Частота пускового Секція ACC та кінцева частина DEC не можуть бути меншими за результат розрахунку вище. Якщо макс. швидкість нижча за результат розрахунку вище, імпульсний вихід відсутній.

3. Швидкість повзання має бути більшою за нуль і меншою за одну десяту від максимальної швидкості.
4. Детальну інформацію див. у розділі 11 "Посібник користувача для функції позиціонування".

6.21.5 DSZR: Інструкція для повернення до нуля з DOG

LAD:	Придатна модель										VC1 VC2 VC3				
	DSZR (S1) (S2) (D1) (D2)										Прапорець впливу біт				
DSZR (S1) (S2) (D1) (D2)										Довжина кроку					
Робота	Тип	Застосовний програмний елемент													Показник
S1	BOOL		X	Y	M	S									
S2	BOOL		X												
D1	BOOL			Y											
D2	BOOL			Y	M	S									

- Операнд опис

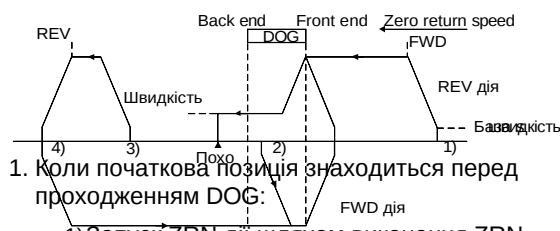
S1: Визначення номера програмного елемента DOG. Коли призначено вхідний програмний елемент, зміщення положення нульової точки збільшується через вплив циклу обчислення PLC.

S2: Визначення номера програмного елемента для вхідного нульового сигналу. Діапазон: X0–X7.

D1: Визначення кількості імпульсів вихідних імпульсів. **D2:** Визначення номера вихідного об'єкта сигналу напрямку обертання.

- Опис функції

ZRN інструкція, що дозволяє використання DOG сигналу та нульового сигналу. Існують FWD та REV обмеження. ZRN дія залежить від ZRN положення. А сигнал скидання надсилається після завершення інструкції.



1. Коли початкова позиція знаходиться перед проходженням DOG:

- 1) Запуск ZRN дії шляхом виконання ZRN інструкції.
- 2) Рух у напрямку ZRN на швидкості повернення до нуля.
- 3) Початок сповільнення до швидкості повзання після виявлення передньої частини DOG.
- 4) Зупинка після повторного виявлення першого нульового сигналу після виявлення заднього краю DOG.

2. Коли початкове положення знаходиться в межах DOG:

- 1) Запуск дії ZRN шляхом виконання інструкції ZRN.

2) Рух у напрямку, протилежному до напрямку ZRN, зі швидкістю повернення до нуля.

3) Уповільнення до зупинки після виявлення переднього краю DOG. (залишення DOG)

4) Рух у напрямку ZRN з нульовою швидкістю повернення. (повторний вхід у DOG)

5) Початок сповільнення до швидкості повзання після виявлення передньої частини DOG.

6) Зупинка після виявлення першого нульового сигналу після детектування заднього краю DOG.

3. Коли початкова позиція знаходиться в DOG OFF (після проходження DOG):

1) Запуск ZRN дії шляхом виконання ZRN інструкції.

2) Рух у напрямку ZRN на швидкості повернення до нуля.

3) Уповільнення до зупинки після виявлення зворотного кінцевого вимикача.

4) Рух у напрямку, протилежному до напрямку ZRN, зі швидкістю повернення до нуля.

5) Уповільнення до зупинки після виявлення переднього краю DOG. (виявлення (вихід) DOG)

6) Рух у напрямку ZRN при нульовій швидкості.

7) Початок сповільнення до швидкості повзання після виявлення передньої частини DOG.

8) Зупинка після виявлення першого нульового сигналу після детектування заднього краю DOG.

4. Коли початкове положення знаходиться в положенні кінцевого вимикача (кінцевий вимикач FWD або REV):

1) Запуск ZRN дії шляхом виконання ZRN інструкції.

2) Рух у напрямку, протилежному до напрямку

ZRN, зі швидкістю повернення нуль.

3) Уповільнення до зупинки після виявлення
переднього краю

DOG.(виявлення (залишити) DOG)

- При 4) Рух у напрямку ZRN з нульовою швидкістю повернення. (повторний вхід у DOG)
 - а 5) Початок сповільнення до швидкості повзання після виявлення передньої частини DOG.
 - 6) Зупинка після виявлення першого нульового сигналу після виявлення заднього краю DOG.

1. PLSY, PLS, та інструкції позиціонування можуть виводити сигнали через порт високошвидкісних імпульсів. Не дозволяється використовувати ці інструкції для високошвидкісного імпульсного виходу на одному й тому самому порту одночасно.

2. Мінімальна вихідна частота імпульсів, яку фактично можна вивести, визначається за наведеною нижче формулою:

$$F_{\min\_acc} = \sqrt{\frac{F_{\max} \times}{500}}$$

У наведеній вище формулі $F_{\max}$ позначає макс. швидкість, тоді як T вказує на час розгону/гальмування, з одиницею вимірювання мс. Результат обчислення $F_{\min\_acc}$ є граничним значенням мін. вихідної частоти.

3. Для частоти імпульсного виходу розрахована частота виводиться, навіть якщо призначено значення, менше за розраховане. Частота початкової ділянки ACC та кінцевої частини DEC не може бути меншою за вищевказаний результат розрахунку. Якщо макс. швидкість нижча

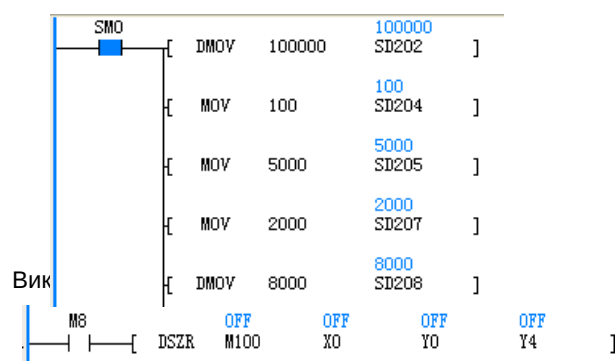
ніж вищевказаний результат обчислення, імпульсний вихід відсутній.

4. Швидкість повзання має бути більшою за нуль і меншою за одну десяту від максимальної швидкості.

5. Детальну інформацію див. у розділі 11 "Посібник користувача для функції позиціонування".

Екземпляр застосунку

Такі параметри, як макс. швидкість, базова швидкість, час розгону/гальмування, швидкість повернення до нуля та швидкість повзання, можуть приймати значення за замовчуванням або бути скинуті через призначення програмних елементів.



6.21.6 DVIT: Інструкція переривання позиціонування instruction

LAD:		Придатна модель	VC1													
		Прапорець впливу біт														
IL: DVIT (S1) (S2) (D1) (D2)		Довжина кроку	11													
Робота d	Тип	Застосовний програмний елемент														Показники
S1	DINT	Постійний	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
S1	DINT	Постійний	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		R	✓
D1	BOOL			Y												
D2	BOOL			Y	M	S										

- Операнд опис
 - S1:** Визначення кількості вихідних імпульсів після переривання (відносна адреса).
 - S2:** Визначення вихідної частоти імпульсів. **D1:** Визначення вихідної кількості вихідних імпульсів. **D2:** Визначення вихідного номера об'єкта сигналу

напрямку обертання.

- Опис функції
 - 1. Коли генерується переривання, імпульс виводиться із заданою частотою та заданою кількістю

імпульси. SM284 є чинним для функції входу переривання; SD176 — це позначення функції входу переривання; SM283 — це програмний елемент логічної інверсії сигналу входу переривання. Логічна інверсія визначає, чи генерує програмний елемент переривання переривання за ON або OFF.

2. Метод позначення входу переривання є таким: SM284 встановлено на ON.

3. Призначення вхідного номера (X0–X7) як входу переривання у SD176, або призначення користувацького переривання

інструкційні програмні елементи, у яких LSB з SD176

відповідає входу переривання, що використовується імпульсним виходом

Y0, а MSB відповідає входу переривання, що використовується для імпульсного виходу Y1.

Встановлення значення	Налаштування вмісту								
0	Призначення X0 як вхідного сигналу переривання								
1	Призначення X1 як вхідного сигналу переривання								
.....									
7	Призначення X7 як вхідного сигналу переривання								
8×1	Призначення програмного елемента інструкції переривання користувача ^{x1} як вхідного сигналу переривання								
	<table> <tr> <th>Програмний елемент імпульсного виходу</th><th>Переривання користувача інструкція soft елемент</th></tr> <tr> <td>Y0</td><td>SM285</td></tr> <tr> <td>Y1</td><td>SM305</td></tr> <tr> <td>Y2</td><td>SM325</td></tr> </table>	Програмний елемент імпульсного виходу	Переривання користувача інструкція soft елемент	Y0	SM285	Y1	SM305	Y2	SM325
Програмний елемент імпульсного виходу	Переривання користувача інструкція soft елемент								
Y0	SM285								
Y1	SM305								
Y2	SM325								

- Примітка

1. PLSY, PLS та інструкції позиціонування можуть виводити сигнал через порт високошвидкісних імпульсів. Заборонено використовувати ці інструкції для виводу високошвидкісних імпульсів на один і той самий порт одночасно.

2. Мінімальна вихідна частота імпульсів, яку фактично можна вивести, визначається за наведеною нижче формулою:

$$F_{\min\_acc} = \sqrt{\frac{F_{\max} \times}{500}}$$

6.21.7 STOPDV: Зупинка імпульсного виходу інструкція

LAD: [STOPDV (S1) (S2) (S3) (D)]								Придатна модель		VC3					
								Зумовлений прапор біт							
IL: STOPDV (S1) (S2) (S3) (D)								Довжина кроку		12					
Робота d	Тип	Застосовний програмний елемент													Індексація
S1	DINT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		✓
S2	DINT	Константа мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C		V		✓

У наведеній вище формулі, макс. вказує на макс. швидкість,

поки T вказує час розгону/гальмування,

з одиницею вимірювання мс. Результат обчислення $F_{\min\_acc}$ є граничним значенням мін. вихідної частоти.

3. Для частоти імпульсного виходу розрахована частота виводиться, навіть якщо призначено значення, менше за розраховане. Частота початкової ділянки ACC та кінцевої частини DEC не може бути меншою за вищевказаний результат розрахунку. Якщо макс. швидкість нижча за вищевказаний результат розрахунку, імпульсний вихід відсутній.

4. Коли кількість вихідних імпульсів менша за кількість імпульсів, необхідних для сповільнення, частоту можна знижувати.

5. Детальну інформацію див. у розділі 11 "Посібник користувача для функції позиціонування".

- Екземпляр застосунку

3 M9 [DINT 9000 1000 OFF Y0 OFF Y1]

S3	DINT	Конст анта мур аха	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	V	✓
D	BOOL			Y									

- Операнд опис

S1: Кількість вихідних імпульсів після виконання інструкції (відносна адреса).

S2: Базова швидкість під час уповільнення.

S3: Час уповільнення від початкової вихідної швидкості до базової швидкості.

D: Номер вихідної точки, що відповідає високошвидкісним імпульсам.

- Опис функції

1. Виконання інструкції STOPDV. Її можна запустити під час процесу виконання інструкцій PLSY, PLS та позиціонування, вона зупиняє вихідну дію вказаної осі.

2. Коли потік енергії приводу інструкції дорівнює ON, імпульсний вихід зупиняється після виконання вказаної кількості імпульсів. Коли вказана кількість імпульсів дорівнює 0, інструкція негайно припиняє свою вихідну дію. Коли вказана кількість імпульсів більша за 0, інструкція спочатку продовжує початкову вихідну дію, а потім

сповільнюється до базової швидкості та зупиняє вихідну дію, коли досягається базова швидкість.

3. Базова швидкість та час розгону/гальмування також встановлюються у спеціальному регістрі даних вихідної осі. Виконання інструкції не змінює налаштування спеціального регістру даних. Базова швидкість та час розгону/гальмування під час виконання інструкції виконуються відповідно до встановленого операнда інструкції, а не з використанням конфігурації у спеціальному регістрі даних.

4. Сигнал напрямку вихідної осі не потребує визначення, а сигнали напрямку, вказані в оригінальних PLSY, PLS та інструкціях позиціонування, розпізнаються автоматично. Стан ON/OFF сигналу напрямку не змінюється під час виконання інструкції.

- Примітка

1. The min. output pulse frequency that can be outputted actually, is determined by the formula below:

$$F_{\text{min\_приск}} = \sqrt{\frac{F_{\text{макс.}} \times}{500}}$$

У наведеній вище формулі, F_{max} вказує на макс. швидкість, встановлено

через SD85 та SD86 поки T вказує той час розгону/гальмування, встановлюється через SD87, з одиницею вимірювання мс. Результат обчислення $F_{\text{min\_acc}}$ є граничним значенням мін. вихідної частоти.

2. Для частоти імпульсного виходу S2 розрахована частота виводиться, навіть якщо призначено значення, менше за розраховане. Частота початкової секції ACC та кінцевої частини DEC не може бути меншою за вищевказаний результат розрахунку. Якщо макс. швидкість нижча за вищевказаний результат розрахунку, імпульсний вихід відсутній.

3. Коли кількість вихідних імпульсів менша за кількість імпульсів, необхідних для сповільнення, частота може бути знижена.

4. Номер вихідної точки D, що відповідає високошвидкісним імпульсам, може визначати Y0, Y1, Y2, Y3, Y4, Y5, Y6 та Y7.

5. Коли інструкція виконується, якщо вихідна вісь уже перебуває у стані STOP, жодна операція не виконується; якщо вихідна вісь виконує інструкції LIN, CW або CCW, жодна операція не виконується.

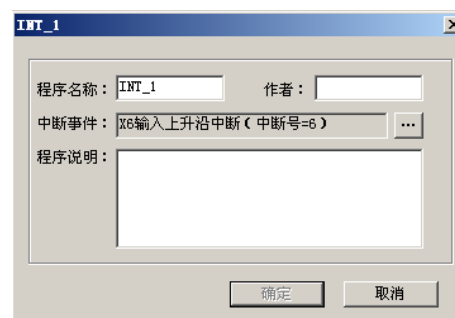
6. Детальну інформацію див. у розділі 11 "Посібник користувача для функції позиціонування".

- Екземпляр застосунку

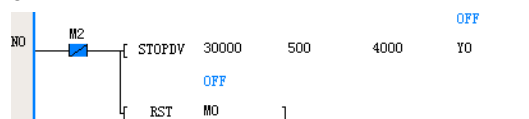
У головній програмі ви можете випробувати інструкцію PLSY для керування Y0. Потік енергії контролюється елементом M0 елемент:



Налаштування джерела переривання для підпрограми переривання, наприклад:

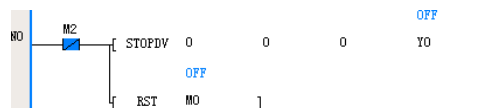


Додавання наступного твердження до підпрограми переривання subprogram:



У наведеній вище інструкції, коли X6 встановлено в ON, Y0 видає певну кількість імпульсів, а потім сповільнюється до зупинки. Якщо X6 встановлено в ON до повної зупинки Y0, видається 30 000 імпульсів, на що не впливає цикл сканування.

Наприклад, операнд 1 в інструкції STOPDV в INT_1 дорівнює 0, діаграма наведена нижче:



Коли виникає подія джерела переривання (далі мається на увазі передній фронт X6), Y0 негайно зупиняється і не залежить від циклу сканування.

Примітка: коли викликається інструкція STOPDV, необхідно перервати потік енергії високошвидкісної інструкції, що виконується для Y0 в основній функції, щоб запобігти перезапуску високошвидкісного виходу через сканування інструкції в основній функції після того, як Y0 було зупинено.

6.21.8 LIN: Інструкція лінійної інтерполяції траєкторії instruction

LAD: 		Придатна модель	VC3	
		Зумовлений прапор біт		
IL: LIN (S) (D1) (D2) (D3) (D4)		Довжина кроку 12		
Робота d	Тип	Застосовний програмний елемент		Індексація

S	DINT									D						
D1	BOOL			Y												
D2	BOOL			Y												
D3	BOOL			Y												
D4	BOOL			Y												

- Операнд опис

S: Початкова адреса області зберігання таблиці параметрів. **D1:** Номер вихідної точки, що відповідає імпульсному сигналу осі X (або сигналу додатного імпульсу). Можна вказати лише Y0.

D2: Номер вихідної точки, що відповідає сигналу напрямку X-осі (або сигналу від'ємного імпульсу). Можна вказати лише Y1.

D3: Номер вихідної точки, що відповідає імпульсному сигналу осі Y (або сигналу позитивного імпульсу). Можна вказати лише Y2.

D4: Номер вихідної точки, що відповідає сигналу напрямку осі Y (або сигналу від'ємного імпульсу). Можна вказати лише Y3.

- Опис функції

1. Переміщення до цільової позиції вздовж прямої траєкторії із заданою векторною швидкістю.

2. Таблиця параметрів визначення

D елемент	Вміст		
S	Зарезервовано		
S+1	Взаємозв'язок режиму дії та логіки виходу (код конфігурації у десятковій системі)		
	Конфігурація	Режим	логіка
	00	Інкремент типу a1	Імпульс+напрямок (Вперед: ON/реверс: OFF)
	01	Інкремент типу a1	Прямий імпульс+зворотний імпульс
	10	Абсолютне значення типу	Імпульс+напрямок (Вперед: ON/назад: OFF)
	11	Абсолютна величина типу	Прямий імпульс+зворотний імпульс
S+2	Результуюча швидкість початкова швидкість Fmin (Гц) (MSB)		
S+3	Результуюча швидкість початкова швидкість Fmin (Гц) (LSB)		
S+4	Результуюча швидкість макс. швидкість Fmax (Гц) (MSB)		
S+5	Результуюча швидкість макс. швидкість Fmax (Гц) (LSB)		
S+6	Час розгону/гальмування T (мс) (MSB)		
S+7	Час розгону/гальмування T (мс) (LSB)		

S+8	Цільова позиція осі X (відстань переміщення) (MSB)
S+9	Цільова позиція осі X (відстань переміщення) (LSB)
S+10	Цільове положення осі Y (відстань переміщення) (MSB)
S+11	Цільова позиція осі Y (відстань переміщення) (LSB)

Серед яких:

(1) В інкрементальному режимі ціль траєкторії приймає відносну адресу, яка стосується відстані переміщення від поточного положення до осей X та Y протягом цільового періоду.

(2) У режимі абсолютних значень ціль траєкторії використовує абсолютну адресу, яка стосується абсолютних координат положення цільової позиції на X та Y осях.

- Примітка

1. Номери вихідних точок D1 та D3, що відповідають двом вихідним імпульсним сигналам осі (або сигналам прямого імпульсу) в інструкції, повинні використовуватися групами. Коли вихідна група може бути призначена лише як Y0 та Y2, Y1 та Y3 використовуються в комбінації з Y0 та Y2 відповідно для забезпечення сигналу напрямку або сигналу виходу від'ємного імпульсу.

2. Вихідну групу (Y0 та Y2) можна налаштувати в режимі "Імпульс + напрямок" або "Позитивний

імпульс + негативний імпульс", максимальна швидкість однієї осі становить 200k, тоді як результуюча швидкість становить максимум 200K.

3. Діапазон налаштування відстані переміщення для кожної осі становить від -8388608 до +8388607 імпульсів.

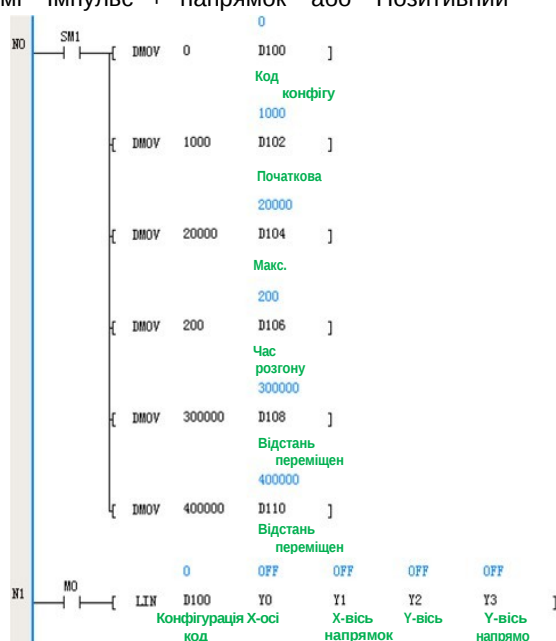
4. Заборонено одночасно використовувати декілька інструкцій PLSY, PLS або інструкцій позиціонування на одному вихідному порту високошвидкісних імпульсів.

5. Діапазон часу розгону та гальмування становить 5-5000ms.

6. Підтримується лише трапецеїдальне прискорення/сповільнення, і час сповільнення неможливо налаштувати окремо.

7. Переривання завершення підтримує лише одне (Y0Y1) переривання.

- Екземпляр застосунку



6.21.9 CW: Траєкторія дуги за годинниковою стрілкою інтерполяція

LAD: <							

- Операнд опис

S: Початкова адреса області зберігання таблиці параметрів. **D1:** Номер вихідної точки, що відповідає імпульсному сигналу осі X (або сигналу позитивного імпульсу). Можна вказати лише Y0

D2: Номер вихідної точки, що відповідає сигналу напрямку X-осі (або сигналу від'ємного імпульсу). Можна вказати лише Y1.

D3: Номер вихідної точки, що відповідає імпульсному сигналу осі Y (або сигналу позитивного імпульсу). Можна вказати лише Y2.

D4: Номер вихідної точки, що відповідає сигналу напрямку осі Y (або сигналу від'ємного імпульсу). Можна вказати лише Y3.

- Опис функції

1. Переміщення до цільової позиції вздовж дугової траєкторії за годинниковою стрілкою із заданою лінійною швидкістю.

2. Parameter table definition

D елемент	Вміст		
S	Метод утворення дуги		
	Конфігурація	Режим	
	0	Центральне положення позначення	
	1	Передача позначення позиції	
S+1	Взаємозв'язок режиму дії та логіки виходу (код конфігурації у десятковій системі)		
	Конфігурація співвідношення	Режим	логіка
	00	Інкремент типу al	Імпульс+напрямок
			(Прямий хід: ON/зворотний хід: OFF)

S+4	Результуюча швидкість (MSB)
S+5	Результуюча швидкість (LSB)
S+6	Час розгону/гальмування T (мс) (MSB)
S+7	Час розгону/гальмування T (мс) (LSB)
S+8	Відстань переміщення осі X (цільова позиція) (MSB)
S+9	Відстань переміщення осі X (цільова позиція) (LSB)
S+10	Відстань переміщення осі Y (цільова позиція) (MSB)
S+11	Відстань переміщення осі Y (цільова позиція) (LSB)
S+12	X-координата осі положення центру кола/точки проходження (MSB)
S+13	X-координата осі положення центру кола/точки проходження (LSB)
S+14	Y-координата осі положення центру кола/точки проходження (MSB)
S+15	Y-координата осі положення центру кола/точки проходження (LSB)

Серед яких:

(1) В інкрементальному режимі ціль траєкторії використовує відносну адресу, яка стосується відстані переміщення від поточного положення до осей X та Y протягом цільового періоду.

(2) У режимі абсолютних значень ціль траєкторії використовує абсолютну адресу, що стосується координат абсолютного положення цільової позиції на X та Y осях.

- Примітка

1. Номери вихідних точок D1 та D3, що відповідають двом вихідним імпульсним сигналам осі (або сигналам прямого імпульсу) в інструкції, повинні використовуватися групами. Коли вихідна група може бути призначена лише як Y0 та Y2, Y1 та Y3 використовуються в комбінації з Y0 та Y2 відповідно для забезпечення сигналу напрямку або вихідного сигналу від'ємного імпульсу.

2. Вихідну групу (Y0 та Y2) можна налаштувати в режимі "Імпульс + напрямок" або "Позитивний

імпульс + негативний імпульс", а максимальна швидкість становить 200к.

3. Діапазон налаштування відстані переміщення для кожної осі становить -8388608-8388607 імпульсів.

4. ~~Заборонено одночасно використовувати декілька PLSY, PLS або інструкцій позиціонування на одному вихідному порту високошвидкісних імпульсів.~~

5. The coordinates of the passing point specified in the passing position mode refer to the passing position of the entire circle, but the user-defined trajectory path may not pass through this point.

6. Діапазон часу розгону та гальмування становить 5-5000ms.

7. Підтримується лише трапецеїдальне прискорення/сповільнення, і час сповільнення неможливо налаштувати окремо.

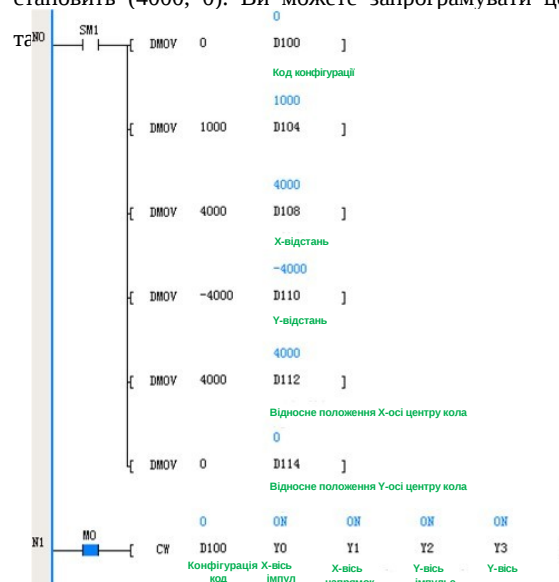
8. Переривання завершення підтримує лише одне (Y0Y1) переривання.

- Екземпляр застосунку

Assuming the current position value of the SD element is (2000, 5000), and you want to draw the arc shown in the following figure:



Ви можете застосувати інкрементальний тип, зміщення кінцевої точки відносно початкової становить (4000, -4000), а зміщення центру відносно початкової точки становить (4000, 0). Ви можете запрограмувати це



6.21.10 CCW: Інструкція інтерполяції дугової траєкторії проти годинникової стрілки instruction

LAD: 							

- Операнд опис

S: Початкова адреса області зберігання таблиці параметрів. D1: Номер вихідної точки, що відповідає

імпульсному сигналу осі X (або сигналу додатного імпульсу). Можна вказати лише Y0.

D2: Номер вихідної точки, що відповідає сигналу напрямку X-осі (або сигналу від'ємного імпульсу). Можна вказати лише Y1.

D3: Номер вихідної точки, що відповідає імпульсному сигналу осі Y (або сигналу позитивного імпульсу). Можна вказати лише Y2.

D4: Номер вихідної точки, що відповідає сигналу напрямку осі Y (або сигналу від'ємного імпульсу). Можна вказати лише Y3.

- Опис функції

1. Переміщення до цільової позиції вздовж дугової траєкторії проти годинникової стрілки із заданою лінійною швидкістю.

2. Таблиця параметрів визначення

D елем ент	Вміст		
S	Метод утворення дуги		
	Кон фігу раці я н	Режим	
	0	Центральне положення позначення	
	1	Передача позначення позиції	
S+1	Взаємозв'язок режиму дії та логіки виходу (код конфігурації у десятковій системі)		
	Кон фігу раці я н	Режим	логіка
	00	Інкремент типу al	Імпульс+напрямок (Вперед: ON/реверс: OFF)
	01	Інкремент типу al	Прямий імпульс+зворотний імпульс
	10	Абсолютний значення типу	Імпульс+напрямок (Прямий хід: ON/зворотний хід: OFF)
	11	Абсолютне значення типу	Прямий імпульс+зворотний імпульс
S+2	Результуюча швидкість початкова швидкість (MSB)		
S+3	Результуюча швидкість початкова швидкість (LSB)		
S+4	Результуюча швидкість (MSB)		
S+5	Результуюча швидкість (LSB)		
S+6	Час розгону/гальмування T (мс) (MSB)		
S+7	Час розгону/гальмування T (мс) (LSB)		
S+8	Відстань переміщення осі X (цільова позиція) (MSB)		
S+9	Відстань переміщення осі X (цільова позиція) (LSB)		
S+10	Відстань переміщення осі Y (цільова позиція) (MSB)		
S+11	Відстань переміщення осі Y (цільова позиція) (LSB)		
S+12	X-координата осі положення центру кола/точки проходження (MSB)		
S+13	X-координата осі положення центру кола/точки проходження (LSB)		
S+14	Y-координата осі положення центру кола/точки проходження (MSB)		
S+15	Y-координата осі положення центру кола/точки проходження (LSB)		

Серед яких:

(1) В інкрементальному режимі ціль траєкторії приймає відносну адресу, яка стосується відстані переміщення від поточного положення

до осей X та Y протягом цільового періоду.

(2) У режимі абсолютних значень ціль траєкторії використовує абсолютну адресу, що стосується координат абсолютного положення цільової позиції на осях X та Y.

сигнали) в інструкції повинні використовуватися групами. Коли вихідна група може бути позначена лише як Y0 та Y2, Y1 та Y3 використовуються в комбінації з Y0 та Y2 відповідно для забезпечення сигналу напрямку або вихідного сигналу від'ємного імпульсу.

2. Вихідну групу (Y0 та Y2) можна налаштувати в режимі "Імпульс + напрямок" або "Позитивний імпульс + негативний імпульс", а максимальна швидкість становить 200k.

3. Діапазон налаштування відстані переміщення для кожної осі становить -8388608-8388607 імпульсів.

4. Заборонено одночасно використовувати

- Примітка

1. Номери вихідних точок D1 та D3, що відповідають двом вихідним імпульсним сигналам осі (або імпульсу прямого ходу

декілька інструкцій PLSY, PLS або інструкцій позиціонування на одному вихідному порту високошвидкісних імпульсів.

5. Координати точки проходження, вказані в режимі позиції проходження, стосуються позиції проходження всього кола, і визначена користувачем траєкторія руху може не проходити через цю точку.

6. Діапазон часу розгону та гальмування становить 5-5000ms.

7. Підтримується лише трапецеїдальне прискорення/сповільнення, і час сповільнення неможливо налаштувати окремо.

8. Переривання завершення підтримує лише одне (Y0Y1) переривання.

● Екземпляр застосу

Assuming the current position value of the SD element is (6000, 1000), and you want to draw the arc shown in the following figure:

зміщення центру до початкової точки становить (0, 4000).

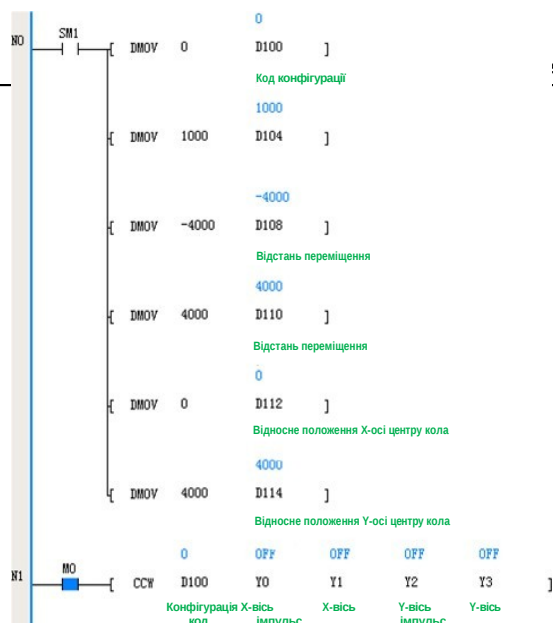
Ви можете запрограмувати це так:



Вісь Y (SD320)

X-axis (SD200)

Ви можете застосувати інкрементальний тип, зміщення кінцевої точки відносно початкової становить (4000, -4000), а це



Micro-PLC

6.21.11 MOVELINK: Інструкція синхронного керування інструкція

LAD: 								

- Операнд опис

S1: Загальна кількість імпульсів, згенерованих підпорядкованою віссю під час слідування за віссю шпинделя.

S2: Загальна кількість імпульсів, згенерованих віссю шпинделя, коли вона супроводжується веденою віссю.

S3: Кількість імпульсів, згенерованих віссю шпинделя до досягнення синхронізації постійної швидкості

S4: Кількість імпульсів, згенерованих віссю шпинделя після завершення синхронізації постійної швидкості

S5: Номер вихідної точки, що відповідає імпульсному сигналу підпорядкованої осі. Діапазон: Y4 та Y5.

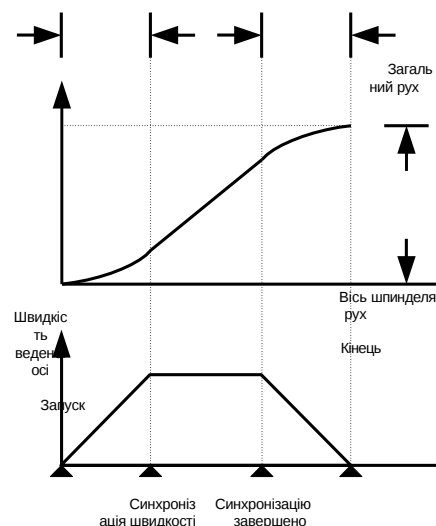
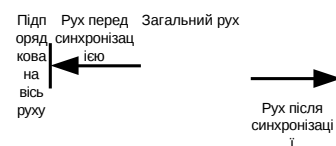
S6: Номер вихідної точки, що відповідає імпульсному сигналу осі шпинделя або номеру високошвидкісного лічильника. Діапазон: Y0–Y7, C236–C263.

- Опис функції

1. Ця інструкція реалізує просту функцію синхронізації для двоосового руху. Ведена вісь вимірює та відстежує швидкість і положення осі шпинделя.

Вісь шпинделя може використовуватися як високошвидкісний вихід або

вхідний порт модуля.



2. Як показано на рисунку вище, після запуску цієї інструкції ведена вісь починає рух зі стану зупинки та

слідую за
головною
віссю після
періоду
розгону та
гальмування.
Коли головна
вісь видає
вказану
кількість
імпульсів **S3**,
швидкість
веденої осі
стає такою ж,
як у головної
осі, і
починається
синхронізація
на постійній
швидкості.
Коли до
досягнення
цільової
позиції
головній осі
залишається
згенерувати
S4 імпульсів,
синхронізація
на постійній
швидкості
припиняється,
і ведена вісь
починає
сповільнювати
ся. Коли
головна вісь
досягає
цільової
позиції, ведена
вісь
зупиняється.

3. Коли вісь
шпинделя
визначена як
вихідна вісь,
операнд **S6**
встановлюєтьс
я на номер
вихідної точки
Y0–Y7;

якщо визначено як вхідну вісь, операнд **S6** встановлюється на C236–C263.

4. Коли потік енергії інструкції увімкнено (ON), підпорядкована вісь починає роботу. Ця інструкція підтримує режим виклику та підпрограму, а також метод виклику програми переривання в основній програмі.

5. Сигнал напрямку вихідної осі не потребує визначення, і стан ON/OFF сигналу напрямку не змінюється під час наступного процесу.

- Примітка

1. Заборонено одночасно використовувати декілька PLSY, PLS або інструкцій позиціонування на одному вихідному порту високошвидкісних імпульсів.

2. Номер вихідної точки D, що відповідає високошвидкісним імпульсам, може бути вказаний як: Y0, Y2, Y4, Y5, Y6, Y7. Y1 та Y3 використовуються в комбінації з Y0 та Y2 відповідно для забезпечення сигналу напрямку або вихідного сигналу від'ємного імпульсу.

- Екземпляр застосунку



6.21.12 GEARBOX: Інструкція з електронного редуктора instruction

LAD:	Придатна модель															
	GEARBOX (D1) (S1) (D2) (S2)								Зумовлений прапор біт							
GEARBOX (D1) (S1) (D2) (S2)				Довжина кроку				9								
Робота d	Тип	Застосовний програмний елемент												Індексація		
D1	BOOL			Y												
S1	DINTКонстанта мураха	KnX	KnY	KnM	KnS	KnLM	KnSM				C		V		√	
D2	BOOL			Y												
S2	DINTКонстанта мураха	KnX	KnY	KnM	KnS	KnLM	KnSM				C		V		√	

- Операнд опис

D1: Номер вихідної точки, що відповідає сигналу осі шпинделя. Діапазон: Y0–Y7.

S1: Цей операнд разом із **S2** визначає коефіцієнт електронного редуктора.

D2: Номер вихідної точки, що відповідає імпульсному сигналу веденої осі. Діапазон: Y4 та Y5.

S2: Цей операнд разом із **S1** визначає коефіцієнт електронного редуктора.

Діапазон коефіцієнта електронного редуктора (**S1/S2**): 1/10000–10000.

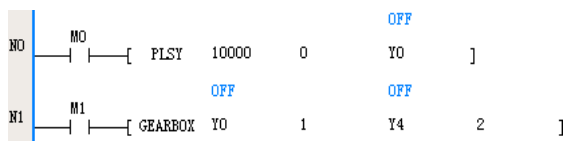
- Опис функції

- Увімкнення веденої осі для слідування за віссю шпинделя відповідно до передавального відношення електронного редуктора (S1/S2). Коли вісь шпинделя надсилає N імпульсів, ведена вісь надсилає $N \times S1/S2$ імпульсів. Коли вісь шпинделя видає імпульси з частотою F, ведена вісь видає імпульси з частотою $F \times S1/S2$.
- Коли потік енергії команди вмикається (ON), підпорядкована вісь починає працювати.
- Сигнал напрямку вихідної осі не потребує визначення, і він не змінює свій ON/OFF стан сигналу напрямку під час процесу відстеження.
- Коли коефіцієнт електронного редуктора менший за 0, тобто одне значення S1, S2 є додатним, а інше від'ємним, відповідні SD елементи веденої осі поступово зменшуються разом із вихідним імпульсом.

- Примітка

- Заборонено одночасно використовувати декілька PLSY, PLS або інструкцій позиціонування на одному вихідному порту високошвидкісних імпульсів.
- Якщо коефіцієнт електронного редуктора виходить за межі вказаного діапазону, імпульсний вихід відсутній.
- Номер вихідної точки D, що відповідає високошвидкісним імпульсам, можна вказати як: Y0, Y2, Y4, Y5, Y6 та Y7. Y1 та Y3 використовуються в комбінації з Y0 та Y2 відповідно для забезпечення сигналу напрямку або сигналу виходу від'ємного імпульсу.

- Екземпляр застосунку



6.22 Інструкції з обробки даних

6.22.1 MEAN: Означає інструкцію

LAD:				Застосовна модель VC3 VC5	
	[MEAN (S1) (D) (S2)]			Зумовлений прапорПрапорець нуля, прапорець перенесення та прапорець позики прапор	
	: MEAN (S1) (D) (S2)			Довжина кроку	

Операнд	Тип	Застосовний програмний елемент											Індикація		
S1	INT	Складається немає	KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T		R	✓
S2	INT	Складається немає							D					R	✓
D	INT			KnY	KnM	KnS	KnLM		D	SD	C			R	✓

- Операнд опис

S1: Номер початкового елемента слова, у якому зберігаються дані, над якими необхідно виконати операцію обчислення середнього значення

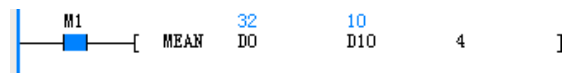
S2: Кількість одиниць даних, над якими необхідно виконати операцію усереднення (1–64)

D: Номер елемента слова, у якому зберігається отримане середнє значення

- Опис функції

1. Збереження середнього значення 16-бітних даних S2, починаючи з S1, у D, з округленням залишку.

- Екземпляр застосунку

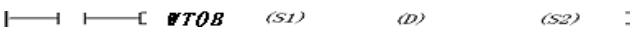


LD M1

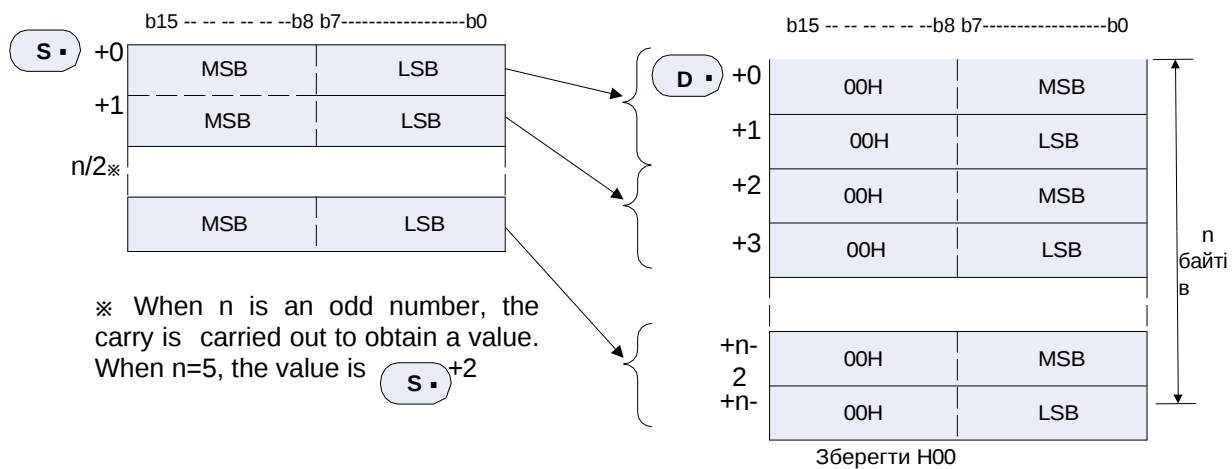
MEAN D0 D10 4

Обчислення середнього значення чотирьох одиниць дані sof починаючи з D0, і збереження отриманого значення в D10, коли M1 = ON. Коли D0=32, D1=10, D2=15, і D3=-14, D10=10.

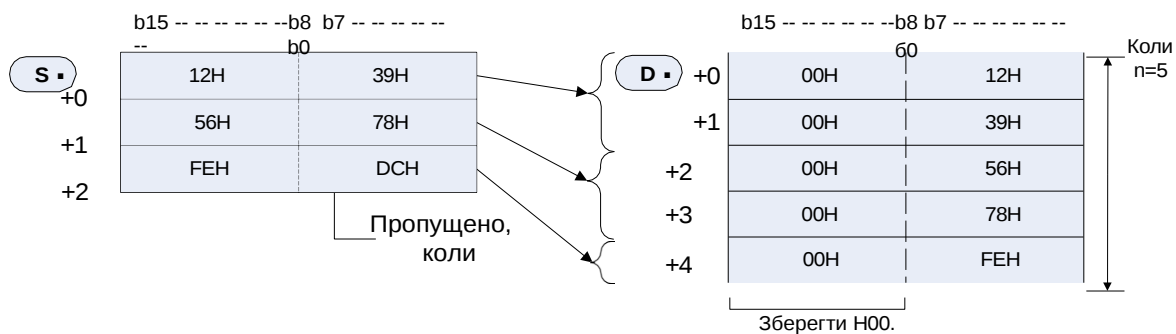
6.22.2 WTOB: Інструкція з розділення даних у байтових одиницях instruction

LAD: 									Застосовна модель			VC3 VC5				
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор				
IL: WTOB (S1) (D) (S2)									Довжина кроку			7				
Операнд	Тип	Застосовний програмний елемент													Індексація	
S1	INT								D	SD	C	T		R	√	
S2	INT								D	SD	C	T		R	√	
D	INT	Складається немає							D					R	√	

- Операнд опис
 - S1:** Номер початкового програмного елемента, у якому зберігаються дані для розділення в байтовий блок
 - S2:** Кількість байтів даних, що підлягають розділенню ($S2 \geq 0$)
 - D:** Номер початкового м'якого елемента, у якому зберігається результат, розділений побайтово unit
- Опис функції
 - Відокремлення 16-бітних даних, що зберігаються в програмних елементах S2/2, з S1 у байти S2, їх збереження в молодших байтах програмних елементів S2, починаючи з D, та очищення старших байтів.

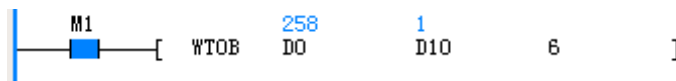


2. Коли S2 є непарним, лише старший байт (8 біт) є об'єктними даними в останніх даних розділеного джерела. Наприклад, коли $n=5$, дані молодших байтів $S-S+2$ зберігаються в $D-D+4$.



3. Коли S2=0, інструкція не виконується.
4. Джерело та цільові операнди не можуть перекриватися.

- Екземпляр застосунку



LD M1

WTON D0 D10 6

Розділення трьох одиниць даних, починаючи з D0, на шість одиниць даних відповідно до старших та молодших байтів, із подальшим збереженням отриманих даних у шести одиницях, починаючи з D10, коли M1=ON. Коли D0=0x102, D1=0x304 та D2=0x506, тоді D10=0x01, D11=0x02, D12=0x03, D13=0x04, D14=0x05 та D15=0x06.

6.2.2.3 BTOW: Інструкція комбінування даних байтових одиниць instruction

LAD: 									Застосовна модель			VC3 VC5				
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор				
IL: BTOW (S1) (D) (S2)									Довжина кроку			7				
Операнд	Тип	Застосовний програмний елемент													Індексація	
S1	INT								D	SD	C	T		R	✓	
S2	INT								D	SD	C	T		R	✓	
D	INT	Складається немає							D					R	✓	

- Операнд опис

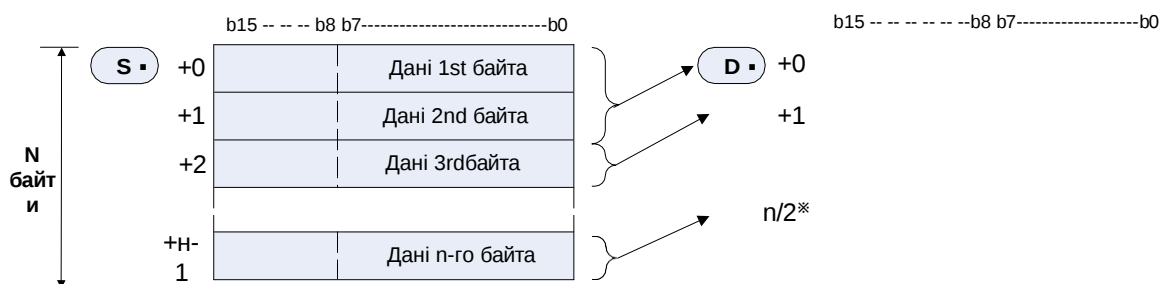
S1: Номер початкового програмного елемента, що зберігає дані для об'єднання в байтову одиницю

S2: Кількість байтів даних для об'єднання ($S2 \geq 0$)

D: Номер початкового програмного елемента, у якому зберігається результат, об'єднаний у байт unit

- Опис функції

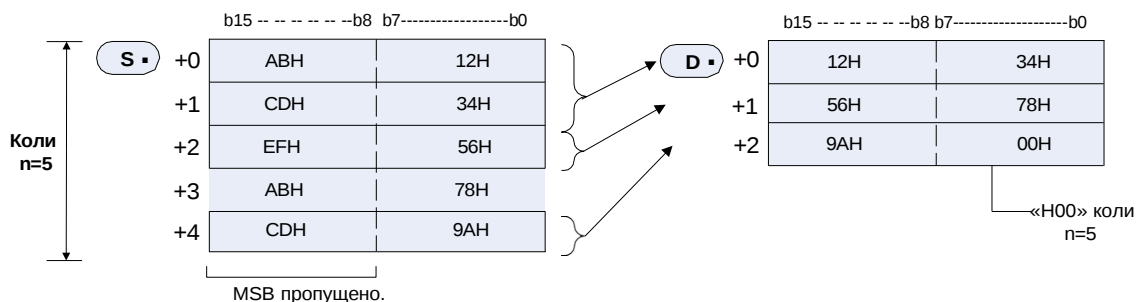
1. Об'єднання молодших байтів (8 біт) 16-бітних даних S2, починаючи з S1, у 16-бітні дані та збереження цих даних у D програмних елементах $S2/2$, починаючи з D. Старші байти (8 біт після S1) 16-бітних даних у джерелі даних ігноруються.



1st байт	2nd байт
3rd байт	4th байт
n-1th байт	n-ний байт

MSB пропущено.

2. Коли S2 є непарним, найменш значущий байт, що об'єднується в кінцевому результаті, є очищеним.



3. Коли S2=0, інструкція не виконується.

4. Джерело та цільові операнди не можуть перекриватися.

● Екземпляр застосунку

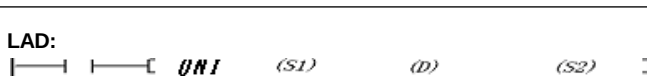


LD M1

BTOW D0 D10 6

Об'єднання шести одиниць даних, починаючи з D0, у три одиниці даних та збереження отриманих даних у трьох одиницях, починаючи з D10, коли M1=ON. Коли D0=0x01, D1=0x02, D2=0x03, D3=0x04, D4=0x05 та D5=0x06, D10=0x102, D11=0x304 та D12=0x506.

6.22.4 UNI: Інструкція для об'єднання 4bits 16-бітних даних

LAD: 										Застосовна модель				VC3 VC5					
										Зумовлений прапор біт				Прапорець нуля, прапорець перенесення та прапорець позики прапор					
IL: UNI (S1) (D) (S2)										Довжина кроку				7					
Операнд		Тип		Застосовний програмний елемент												Індексація			
S1		INT								D	SD	C	T		R	√			
S2		INT								D	SD	C	T		R	√			
D		INT		Складається немає						D					R	√			

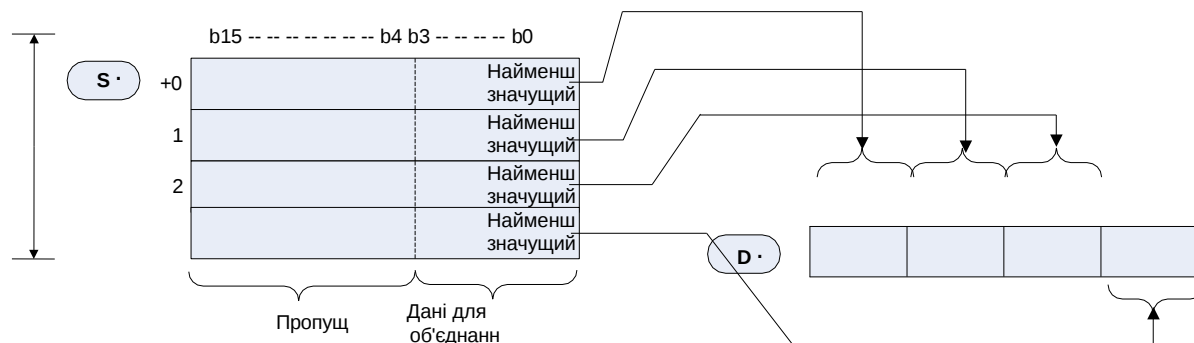
S1: Номер початкового програмного елемента, у якому зберігаються дані для об'єднання

S2: Кількість комбінованих даних (якщо діапазон **S2** становить 0—4, а **S2=0**, обробка відсутня)

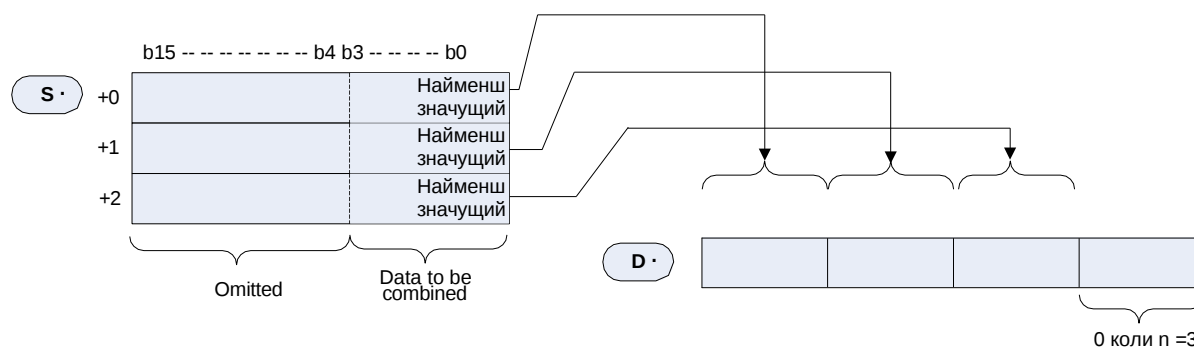
D: Номер програмного елемента, у якому зберігаються об'єднані дані

● Опис функції

1. Збереження 16-бітних даних **S2**, починаючи з **S1**, у програмних елементах **S2**, починаючи з **D**.



2. Коли **S2** знаходиться в діапазоні від 1 до 3, одиниці LSB $\{4 \times (4-S2)\}$ **D** дорівнюють нулю.



3. Коли діапазон **S2** становить 1-4, а **S2=0**, інструкція не виконується.

4. Джерело та цільові операнди не можуть перекриватися.

● Екземпляр застосунку



LD M1

UNI D0 D10 4

Об'єднання молодших чотирьох бітів чотирьох одиниць даних, починаючи з D0, та їх збереження у D10. Коли D0=0x01, D1=0x02, D2=0x03 та D3=0x04, D10=0x1234.

6.22.5 DIS: Інструкція для розділення 4bitsof 16-бітових даних

LAD: 									Застосовна модель			VC3 VC5				
									Зумовлений прапор біт			Прапорець нуля, прапорець перенесення та прапорець позики прапор				
IL: DIS (S1) (D) (S2)									Довжина кроку			7				
Операнд	Тип	Застосовний програмний елемент												Індексація		
S1	INT								D	SD	C	T		R	√	
S2	INT								D	SD	C	T		R	√	
D	INT	Складається немає							D					R	√	

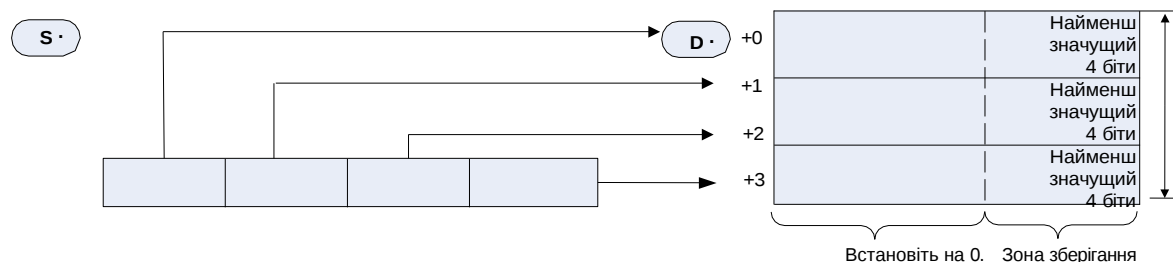
S1: Номер початкового програмного елемента, у якому зберігаються дані, що підлягають розділенню

S2: Номер розділених даних (коли діапазон **S2** становить 0—4, а **S2=0**, обробка відсутня)

D: Номер програмного елемента, у якому зберігаються розділені дані

- Опис функції

1. Збереження 16-бітних даних S2, починаючи з S1, у програмних елементах S2, починаючи з D.

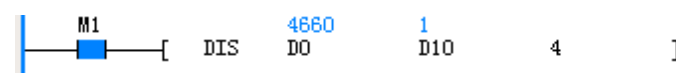


2. Допустимий діапазон S2 становить 1—4, а решта даних не виконують інструкцію.

3. Старші 12 біт програмних елементів S2, починаючи з D, скидаються.

4. Джерело та цільові операнди не можуть перекриватися.

- Екземпляр застосунку



LD M1

DIS D0 D10 4

Розділення кожних 4 бітв даних у блоці D0 та збереження цих даних у чотирьох блоках, починаючи з D10, коли M1=ON.
Коли D0=0x1234, D10=0x01, D11=0x02, D12=0x03 та D13=0x04.

6.22.6 ANS: Інструкція встановлення сигналу тривоги instruction

LAD: </								

- Операнд опис

S1: Номер таймера для визначення часу, застосовується лише для таймера 100ms з діапазоном T0-T209

S2: Дані для визначення часу (1—32767)

D: Налаштування програмних елементів сигналізації. Діапазон: S900—S999

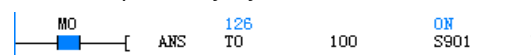
- Опис функції

1. Setting D when the energy flow duration is greater than S2; resetting the timer S1 and not setting D when the energy flow duration of the instruction is less than S2; resetting S1 when the energy flow is invalid, S1 reset.

Адреса № No.	Ім'я	функція
SM400	Увімкніть сигнал аварії	Коли SM400 встановлено на ON, наступні SM401 та SD401 робота

SM401	Сигнальна тривога спрацьовує	SM401 встановлено на ON, коли виконується будь-яка дія у стані S900-S999
SD401	Мінімальна кількість з стан ON стан	Збереження мінімальної кількості від дії у S900-S999

- Екземпляр застосунку

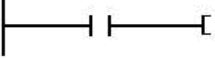


LD M0

ANS T0 100 S901

Встановлення S901 у стан ON, коли потік енергії є дійсним, і не відключати протягом 10 секунд.

6.23.2 DUTY: Інструкція для генерування часових імпульсів

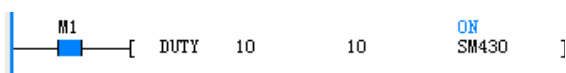
LAD:									Застосовна модель				VC2 VC3 VC5						
									Прапорець впливу біт										
IL: DUTY (S1) (S2) (D)									Довжина кроку				7						
Робота d	Тип	Застосовний програмний елемент														Індексація			
S1	INT	Мінуси натяг		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓		
S2	INT	Мінуси натяг		KnX	KnY	KnM	KnS	KnLM	KnSM	D	SD	C	T	V	Z	R	✓		
D	BOOL	SM																	

- Опис операнда **S1**: ON
часів сканування **S2**:
OFF часів сканування
D: Адреса призначення для виходу імпульсів з часовою затримкою output
- Опис функції
 - Блок виходу імпульсів за часом D змінюється залежно від часу сканування ON та OFF (заданих у S1 та S2, відповідно).
 - SM одиниця. Діапазон: SM430-SM434.

Адреса призначення для імпульсний вихід із витримкою часу output	Програмні елементи для підрахунку кількості сканувань
SM430	SD430
SM431	SD431
SM432	SD432
SM433	SD433
SM434	SD434

3. Цю інструкцію можна використовувати 5 разів, але одна й та сама адреса призначення вихідного сигналу тактового генератора не може бути використана в кількох інструкціях DUTY.

- Екземпляр застосунку



Коли M1=ON, 10 циклів сканування дорівнюють ON, а 10 циклів сканування дорівнюють OFF у SM430, і значення лічильника кількості циклів сканування одночасно зберігається в D430.

- Примітка
Коли робота починається за переднім фронтом команди, потік енергії не зупиняється, навіть якщо її вимкнено. Він зупиняється в STOP або у разі зникнення живлення.

Розділ 7 SFC Наставник

Цей розділ знайомить з основними концепціями програмування, базовими та складними методами

програмування, а також примітками щодо SFC. Розділ 7 SFC Наставник.....251

7.1 Вступ до SFC.....252

7.1.1 Що таке SFC.....252

7.1.2 Що таке SFC серії VC PLCs.....252

7.1.3 Базова концепція SFC.....252

7.1.4 Символи програмування та їхнє використання.....252

7.1.5 SFC структура програми.....	253
7.1.6 Виконання SFC програми.....	257
7.2 Зв'язок між SFC та LAD.....	258
7.2.1 STL інструкція та етап станів.....	258
7.2.2 SET інструкція.....	259
7.2.3 RET інструкція та SFC програмний сегмент.....	259
7.2.4 OUT інструкція та RST інструкція.....	259
7.2.5 SFC гілка вибору, паралельна гілка та об'єднання.....	259
7.3 Як програмувати за допомогою SFC.....	259
7.4 Примітки в SFC програмуванні.....	260
7.4.1 Типові помилки програмування.....	260
7.4.2 Програмування хитрощів.....	263
7.5 Приклади SFC програмування.....	264
7.5.1 Проста послідовна структура.....	265
7.5.2 Гілка вибору структури.....	266
7.5.3 Паралельна гілка структури.....	269

7.1 Вступ до SFC

7.1.1 Що таке SFC

Sequential Function Chart, або SFC, — це мова програмування, яка розвинулася та набула популярності останніми роками. SFC може перетворити проєкт програмування PLC на структуровану блок-схему. Використовуючи елементи програмування та структуру мови, що відповідають стандарту IEC61131-3, вона розділяє складний системний процес на послідовні багатоетапні процедури та переходи між процедурами, і таким чином реалізує послідовне керування.

SFC програмування є прямим та послідовним. Кожна процедура та умова перетворення після декомпозиції є відносно простим програмним процесом, що ідеально підходить для застосунків послідовного керування. Ці переваги пояснюють, чому воно було широко використане.

7.1.2 Що таке SFC серії VC PLCs

SFC серії VC PLCs — це мова програмування, що використовується у продуктах PLC серії VC компанії VEICHI. На додаток до стандартних функцій SFC, в SFC серії VC PLCs можна створювати один або кілька програмних блоків LAD.

Програму, написану за допомогою SFC серії VC PLCs, можна перетворити на відповідну LAD або IL програму.

SFC серії VC PLCs також може підтримувати максимум 20 незалежних процесів. Ці незалежні процеси можуть виконуватися незалежно, тобто стан кроку в межах кожного процесу сканується та передається окремо за процесом. Переходи між незалежними процедурами дозволені.

7.1.3 Базова концепція SFC

SFC має два основні поняття: стан кроку та перехід. Інші поняття, такі як стрибок, розгалуження та множинні незалежні процеси, походять від цих двох основних понять.

- Стан кроку

1. Визначення кроку стану

А стан кроку є фактично незалежною програмою, що представляє робочий стан або операцію в процесі послідовного керування. Об'єднання кількох станів кроків органічним способом може сформувати повну SFC програму.

2. Виконання кроку стану

У програмі SFC стан кожного кроку представлений фіксованим S елементом.

А стан кроку є дійсним, коли він виконується. Для дійсного стану кроку стан відповідного йому елемента S є ON, а PLC сканує та виконує всі послідовності інструкцій у ньому. А стан кроку є недійсним, коли він не виконується. Для недійсного стану кроку стан відповідного йому елемента S є OFF, а PLC не сканує та не виконує його послідовності інструкцій.

- Передача

Процес керування послідовністю — це фактично серія процесів перемикання станів кроків. А PLC виконання стану кроку означає вихід із поточного стану кроку, перехід до нового стану кроку та його виконання, якщо виконано певні логічні умови. Цей процес перемикання називається передачею стану кроку.

Логічна умова, що ініціює перехід, називається умовою переходу кроку.

7.1.4 Символи програмування та їхнє використання

- Програмування символів

Мова програмування VC серії PLC SFC складається з наступних базових символів:

Таблиця 7-1 Програмування символів

Символ назва	Символ	Опис
Початковий крок		Початковий стан кроку, пронумерований як Sn. «n» не може повторюватися. Виконання програми SFC має починатися з символу початкового кроку. Діапазон адрес програмного елемента S, що відповідає символу початкового кроку, становить S0–S19
Нормальний крок		А стан звичайного кроку, пронумерований як Sn. «n» не може повторюватися. Діапазон адрес S логічного елемента, що відповідає символу звичайного кроку, — S20–S991
Передача		А перехід. Він може бути вбудованим із умовою переходу (вбудований LAD). Ви можете скопіювати умову переходу так, щоб стан елемента S, підключеного до цього переходу, встановлювався при виконанні умови, і здійснювався перехід до стану наступного кроку. Символ переходу необхідно використовувати між кроками
Перехід		А символ переходу, що використовується після символу передачі. Він може встановити вказаний S елемент у ON, коли виконуються умови передачі. Він використовується для циклічного виконання або переходу між станами кроків.
Скидання		А символ скидання, що використовується після символу передачі. Він може встановити вказаний S елемент у OFF, коли виконані умови передачі. Він використовується для завершення SFC програми.
Вибір гілки		Кілька незалежних умов переходу, що використовуються після символу кроку. Коли виконується умова переходу однієї з гілок, стан попереднього кроку завершується і починається стан наступного кроку відповідної гілки. Використовується для вибору однієї з кількох гілок кроків. Після вибору однієї гілки інші гілки не можуть бути вибрані.
Вибір об'єднання		А злиття крокових гілок, з'єднаних у вузлі гілок вибору. Коли виконується умова переходу однієї з гілок, стан останнього кроку завершується, і набуває чинності стан наступного кроку.
Паралельна гілка		Підключено після символу кроку, і наступні паралельні гілки мають спільну умову переходу. Коли умова переходу встановлена, наступні паралельні гілки перевіряються та виконуються одночасно.
Паралельне об'єднання		А злиття паралельних гілок, з'єднаних у точці розгалуження паралельних гілок. Коли виконуються кілька паралельних крокових гілок і дотримані умови переходу, стан наступного кроку набуває чинності.
LAD блок		Блок LAD використовується для представлення LAD інструкцій, відмінних від потоку SFC. Його можна використовувати для запуску початкового кроку та інших загальних операцій.

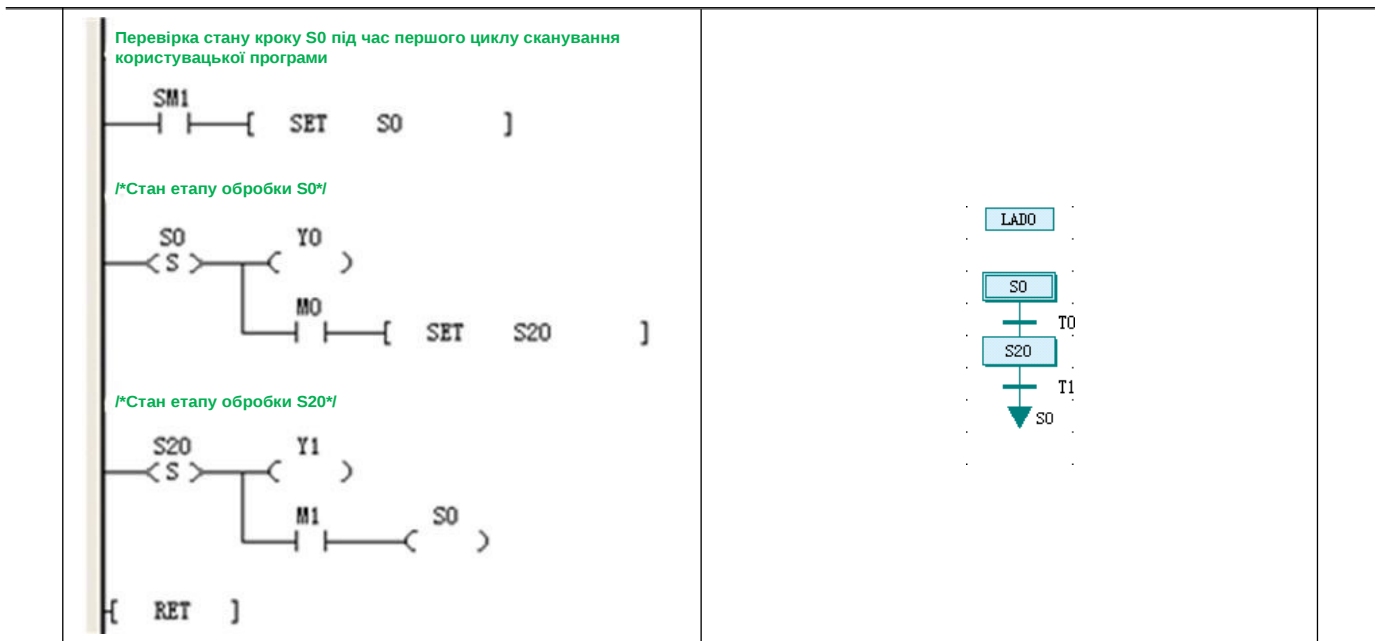
- Використання символів програмування
 1. Символ початкового кроку може використовуватися окремо. Він не може передувати іншим символам і може супроводжуватися лише символами передачі.
 2. Блок A LAD не може бути підключений до інших символів.
 3. А Символ звичайного кроку може бути з'єднаний лише із символом переходу. Символ звичайного кроку не може існувати окремо на діаграмі.
 4. Символи скидання та переходу повинні передувати символам передачі, а за ними має йти нічого.
 5. Ні символ передачі, ні символ переходу не можуть існувати окремо в програмі.

7.1.5 SFC структура програми

Процесна структура SFC класифікується на три типи: проста послідовна структура, структура гілки вибору та структура паралельної гілки. Крім того, структура переходу також є особливою формою структури вибору.

- Проста послідовна структура

На наступному рисунку показано просту структуровану програму SFC та її LAD аналог.



У простій структурованій програмі SFC, коли умови переходу між кроками виконані, програма послідовно переходить від поточного стану кроку до наступного стану кроку без будь-якої розгалуженої структури. На останньому кроці, коли умови переходу виконані, секція програми SFC або завершується, або переходить до початкового стану кроку.

1. LADблок

Блок LAD використовується для запуску програмної секції SFC. Зокрема, встановлення елемента S символу початкового кроку на ON. На попередньому рисунку програма використовує режим запуску при ввімкненні живлення.

Блок LAD також можна використовувати в інших загальних розділах програми, окрім SFC програми.

2. Початковий крок стан

Як показано на рисунку вище, початковий стан кроку запускається блоком LAD. Діапазон елементів S для стану початкового кроку становить 0–19.

3. Звичайний крок символ

Символ нормального кроку використовується для програмування у послідовних процесах. Діапазон елементів S для стану нормального кроку становить 20–1023.

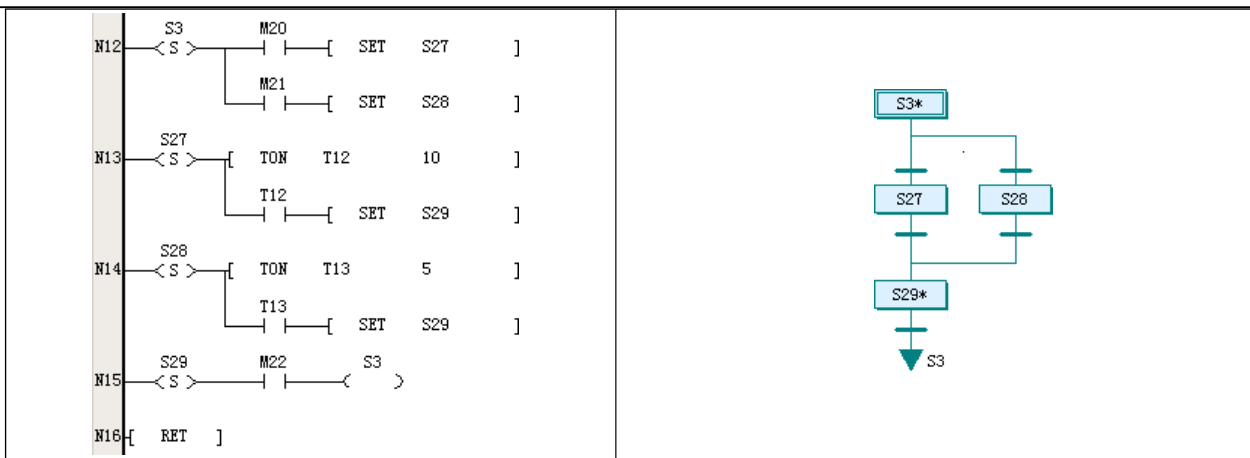
4. Перенести або скинути

Останній символ переходу програми, показаної на рисунку вище, з'єднаний із символом переходу, який спрямовує програму до початкового стану кроку. Це циклічна програма.

Однак останній символ переходу програми можна з'єднати із символом скидання, який може скинути стан останнього кроку. Після скидання програма завершується і очікує на наступний цикл виконання.

- Гілка вибору структури

На наступному рисунку показано структуру гілки вибору, з LAD ліворуч та SFC праворуч right.



1. Вибір гілки

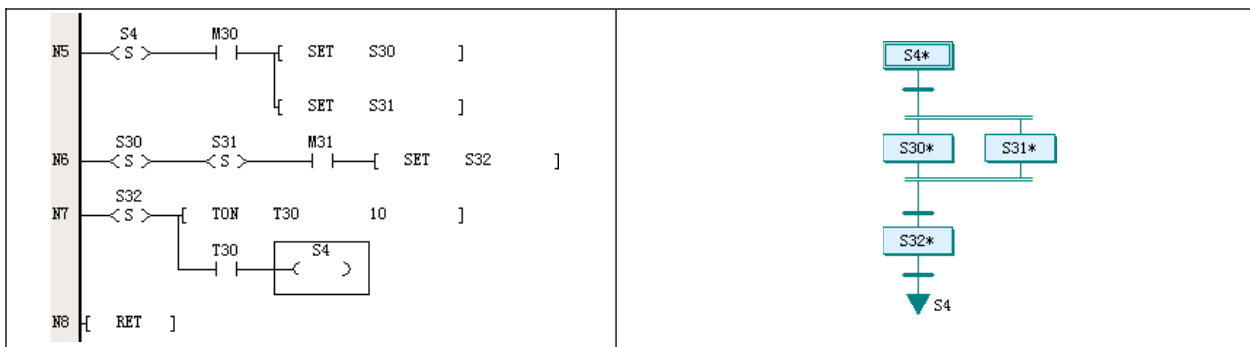
Стан кроку гілки перевіряється, коли виконуються відповідні умови переходу. Вам необхідно забезпечити, щоб умови переходу різних гілок були взаємовиключними, щоб гарантувати, що кожного разу вибирається лише одна гілка. Як показано на попередньому рисунку, S27 та S28 у рядку N12 програми LAD переходять за умов M20 та M21 відповідно. Умови M20 та M21 не повинні бути встановлені одночасно, щоб гарантувати, що S27 та S28 не будуть вибрані одночасно.

2. Вибір об'єднання

Об'єднання вибору — це структура, де всі гілки вибору зливаються в один стан кроку. Умови переходу встановлені відповідно. Як показано на попередньому рисунку, умовою переходу в стані кроку S27 у рядку N13 є закінчення часу T12, тоді як у стані кроку S28 у рядку N14 — закінчення часу T13. Проте результати однакові: стан кроку S29 запускається.

- Паралельна гілка структури

На наступному рисунку показано структуру паралельних гілок, з LAD ліворуч та SFC праворуч.



1. Паралельна гілка

Коли умови переходу паралельної гілкової структури виконані, кожен стан кроку, підключений до паралельної гілкової структури, активується одночасно. Це також поширена структура послідовного керування, яка дозволяє PLC обробляти кілька процедур одночасно. Як показано у рядку N5 програми на попередньому рисунку, S30 та S31 підтверджуються одночасно, коли умова переходу M30 виконана і M30 встановлено.

2. Паралельне об'єднання

Коли виконуються умови переходу структури паралельного об'єднання, кожен стан кроку, підключений до структури паралельного об'єднання, одночасно стає недійсним. Як показано в рядку N6 програми на попередньому рисунку, коли програма одночасно виконує S30 та S31, і встановлено M31, програма запускає S32 та завершує S30 і S31.

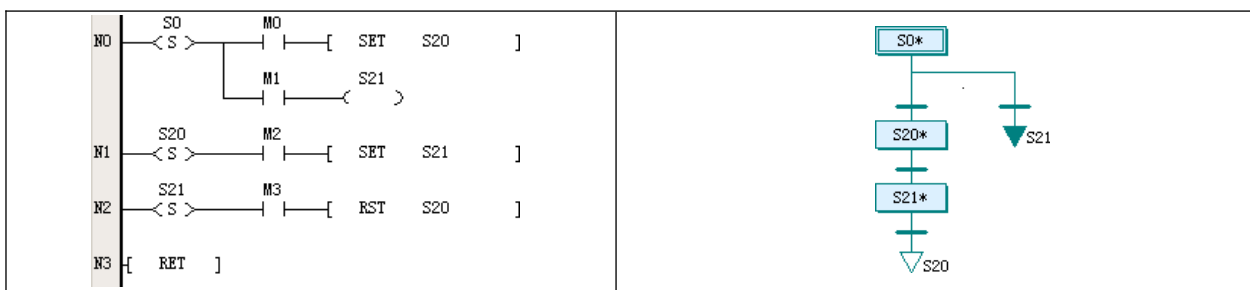
Умова переходу для паралельного злиття полягає в тому, що дія переходу може бути виконана лише тоді, коли всі окремі кроки завершені перед злиттям.

- **Перехід структура**

Структура переходу часто використовується для таких цілей: для пропуску певних станів кроку, для циклічного повернення до початкового стану кроку або нормального стану кроку, а також для переходу до інших процесів.

1. Пропуск певних етапів станів

У процесі, коли послідовне виконання не є обов'язковим за певних умов передачі, програма може переходити безпосередньо до необхідного стану кроку та пропускати непотрібні стани кроків, як показано на наступному малюнку, з LAD увімкненим ліворуч та SFC увімкненим праворуч.



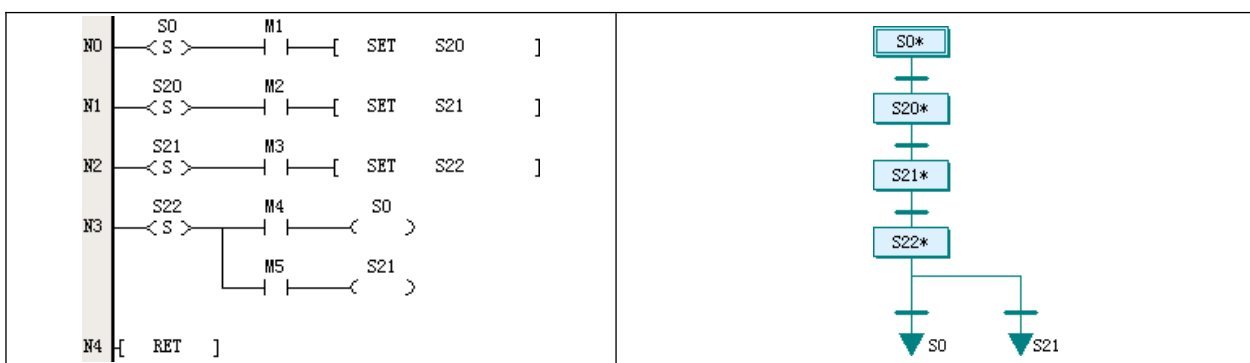
У програмі SFC, показаній на попередньому рисунку, символ переходу S21 використовується як перехід, тоді як стан кроку S20 пропускається. Фактично, перед виконанням переходу вибирається структура розгалуження.

У LAD друга гілка в рядку N0 є інструкцією переходу, яка приймає форму котушки OUT замість форми інструкції SET, що передається послідовно. Під час роботи в кроковому стані S0, коли M1 має стан ON, програма переходить до крокового стану S21.

2. Велоспорт

У процесі, коли необхідно циклічно повторити частину або всі стани кроків за певних умов переходу, можна використати символ переходу для реалізації функції циклу. Наприкінці цього процесу частину станів кроків можна циклічно повторити, якщо програма переходить до попереднього символу звичайного кроку, або всі стани кроків можна циклічно повторити, якщо програма переходить до символу початкового кроку.

На наступному рисунку показано програму, яка реалізує вищезгадані дві циклічні структури одночасно, з LAD ліворуч та SFC праворуч.



У SFC, коли стан кроку S22 є дійсним і виконано одну з умов переходу, програма переходить до S21 і повторно запускає стан кроку S21. За іншої умови переходу програма переходить до початкового стану кроку S0 і повторно запускає всі стани кроків.

У LAD ці два типи переходів реалізовані в рядку N3, де ви можете побачити котушку OUT інструкції переходу.

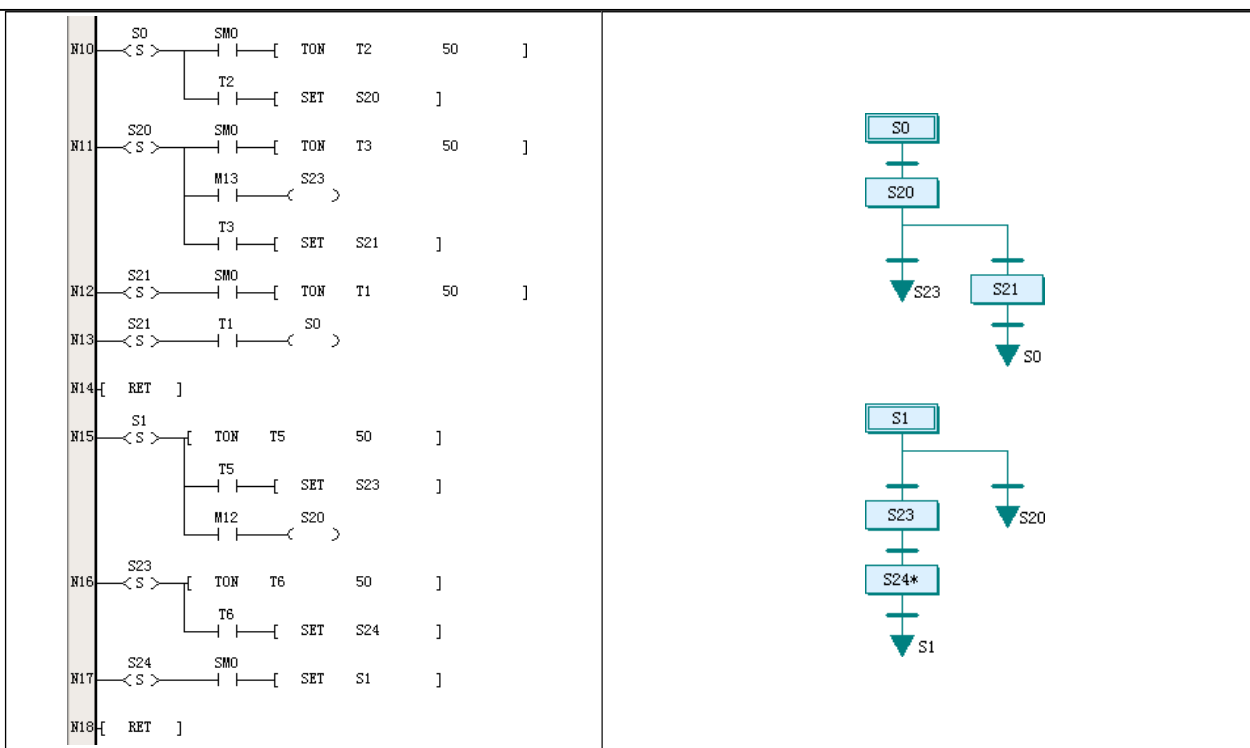
3. Перемикання між різними незалежними процесами

SFC серії VC PLCs підтримують кілька незалежних процесів одночасно, і перемикання між цими процесами є можливим. Ви можете встановити певні умови передачі в незалежному процесі, і коли умови виконані, він безпосередньо передається до іншого незалежного процесу або переходить до випадкового стану кроку (початкового або нормального) іншого незалежного процесу.

Примітка

Перемикання між багатьма незалежними процесами ускладнює PLC програму. Вам потрібно використовувати це з обережністю.

На наступному рисунку показано програму, що реалізує перехід від одного незалежного процесу до іншого, з LAD ліворуч та SFC праворуч.



У SFC, коли стан кроку S0 у першому процесі є дійсним, програма може перейти до стану кроку S23 у другому процесі

за певних умов переходу; поки стан кроку S1 у другому процесі є дійсним, програма також може перейти до стану кроку S20 у першому процесі за певних умов переходу.

Як показано на попередньому рисунку, перехід базується на структурі гілки вибору. Коли програма переходить до іншого процесу, усі стани кроків у початковому процесі стають недійсними. Як показано в прикладі, якщо програма переходить до стану кроку S23 у другому процесі зі стану кроку S20 у першому процесі, S20 встановлюється в OFF, а всі стани кроків (S0, S20 та S21) у першому процесі стають недійсними (OFF).

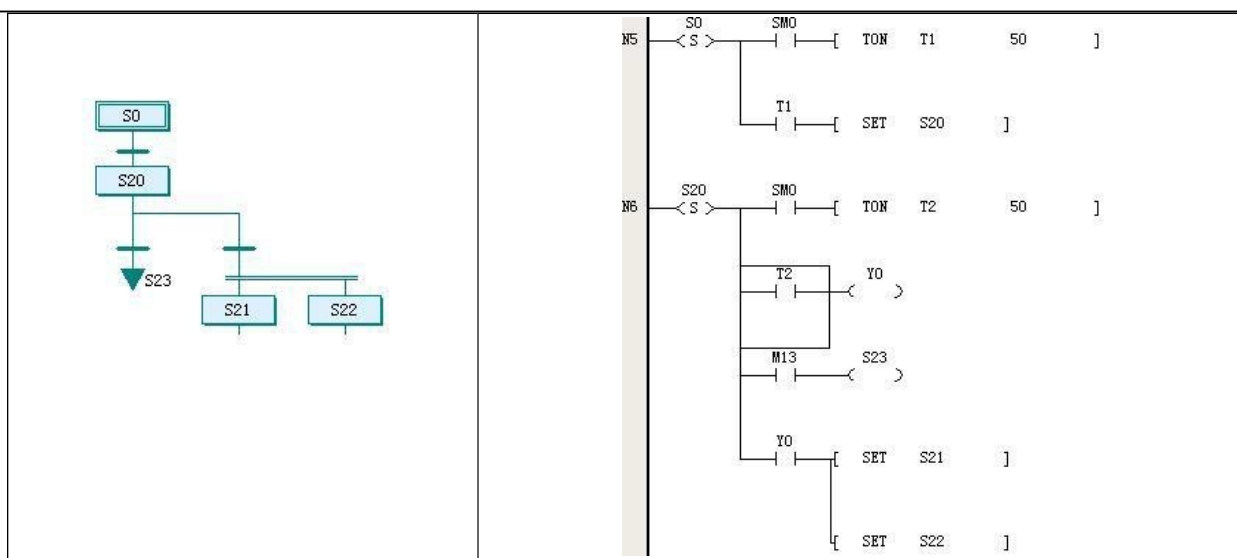
7.1.6 Виконання SFC програми

Подібність між виконанням програми SFC та програми LAD полягає в тому, що обидві вони здійснюють безперервне циклічне сканування зверху вниз і зліва направо.

З іншого боку, їхня відмінність полягає в наступному: у програмі SFC дійсність станів кроку може змінюватися відповідно до умов послідовності, і лише послідовності внутрішніх інструкцій у дійсних станах кроку можуть бути проскановані та виконані. Тоді як у головній програмі звичайного LAD усі програми скануються та виконуються в кожному циклі сканування.

Як показано на наступному рисунку, програма праворуч є LAD аналогом програми SFC ліворуч. Коли стан кроку S20 є

дійсним, таймер T2 сканується і починає відлік часу. Стани кроків S21 та S22 не виконуються, доки таймер T2 не досягне попередньо встановленого значення, а S23 не виконується, коли M13 має стан OFF.



Стан елементів S перемикається між ON та OFF відповідно до умов переходу, таким чином змушуючи програму переходити з одного крокового стану в інший. Коли елемент S змінюється з ON на OFF, вихідні елементи відповідного крокового стану скидаються або очищуються. Детальніше див. у розділі 5.3.1 "STL: інструкція завантаження стану SFC".

Примітка

1. Програма SFC серії VC PLCs зазвичай містить блоки SFC та LAD. Блоки LAD використовуються для обробки операцій поза процесом, включаючи запуск SFC. Блоки LAD не контролюються елементами S, а рядки програми цих блоків LAD, що скануються PLCs, виконуються у кожному циклі сканування.
2. Зміна стану елемента S впливає на вбудовані інструкції відповідних станів кроків, а перемикання між двома станами кроків потребує певного часу, тому необхідно дотримуватися певних правил щодо роботи певних програмних елементів та використання інструкцій під час програмування SFC.

7.2 Зв'язок між SFC та LAD

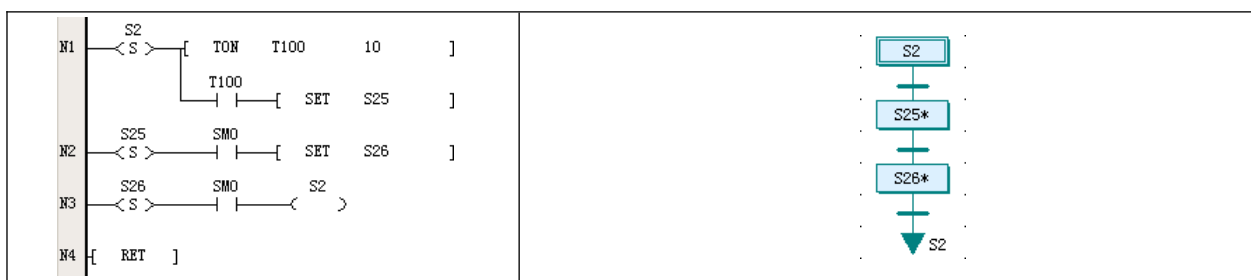
А SFC програма може набувати форми LAD, що може допомогти вам зрозуміти практичне значення структури програми SFC.

У LAD символи SFC замінюються на різні відповідні інструкції SFC, тоді як відповідні процеси представлені різними структурами.

7.2.1 STL інструкція та етап станів

У LAD, стан кроку запускається інструкцією STL. Усі стани кроків представлені елементами S.

Наведена ліворуч діаграма є LAD програмою простої послідовної структури, а діаграма праворуч — це відповідна їй SFC програма.



Як показано у програмі LAD, стан кроку S2 починається з інструкції STL, а наступна інструкція TON є внутрішньою послідовністю інструкцій S2. Внутрішня інструкція стану кроку може складатися з кількох операторів, що фактично є відносно повною програмною секцією, яка майже збігається зі звичайним аналогом LAD.

Різниця між станом початкового кроку та станом звичайного кроку полягає лише в діапазоні, у якому вибираються елементи S.

Для отримання деталей щодо інструкції STL зверніться до розділу 5.3.1 "STL: інструкція завантаження стану SFC". Вам слід зауважити, що коли стан кроку змінюється з ON на OFF, операнди призначення, що відповідають вбудованим інструкціям (OUT, TON, TOF, PWM, HCNT, PLSY, PLSR, DHSCS, SPD, DHSCI, DHSCR, DHSZ, DHST, DHSP та BOUT), очищуються.

□ Примітка

Оскільки PLCs виконується у циклах безперервного сканування, після переходу стану кроку вбудовані інструкції початкового стану кроку не зазнають впливу переходу ON до OFF до наступного сканування. Детальнішу інформацію див. у розділі 7.4.1 "Типові помилки програмування".

7.2.2 SET інструкція

Як показано на попередній схемі, символи передачі в програмі SFC праворуч реалізуються за допомогою інструкції передачі стану SFC.

Умови передачі складаються з контактів NO перед інструкціями SET. Контакти NO керуються вбудованими інструкціями або через зовнішні операції.

Встановлення вказаного стану кроку як дійсного, коли потік енергії інструкції переходу стану SFC є дійсним, та встановлення поточного дійсного стану кроку як недійсного для завершення переходу стану.

7.2.3 RET інструкція та SFC програмний сегмент

Як показано на попередній схемі, програма SFC праворуч починається з початкового кроку S2 і повертається до S2 після проходження двох звичайних кроків. Тоді як у LAD кінець сегмента програми SFC має бути позначений RET інструкцією.

Інструкцію RET можна використовувати лише в основній програмі.

7.2.4 OUT інструкція та RST інструкція

Як показано на попередній діаграмі, перехід до S2 реалізується в LAD рядком N3, який використовує інструкцію OUT. Операнд призначення інструкції OUT (перехід) може знаходитися в будь-якій незалежній процедурі.

Якщо використовується символ скидання S26, рядок N3 у LAD є інструкцією RST, яка може скинути стан останнього кроку S26.

7.2.5 SFC гілка вибору, паралельна гілка та об'єднання

Щодо деталей про LAD лічильну частину SFC гілок вибору, зверніться до структури гілки вибору розділу 7.1.5 "SFC програмна структура".

Щодо деталей про LAD аналог SFC паралельних гілок, зверніться до структури паралельних гілок у розділі 7.1.5 "SFC структура програми".

7.3 Як програмувати за допомогою SFC

1. Аналіз робочого процесу та визначення структури програми

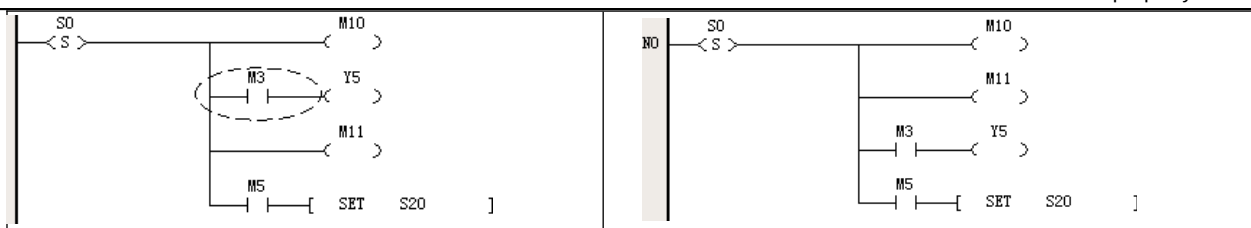
Структуру програми SFC класифікують на три типи: проста послідовна структура, структура гілки вибору та структура паралельної гілки. Крім того, структура переходу також є особливою формою структури гілки вибору. Для програмування з SFC першим кроком є визначення структури потоку. Наприклад, один об'єкт, що проходить через послідовний потік, є простою послідовною структурою. Кілька об'єктів з різними параметрами, які потрібно обробити асинхронно, потребують структури гілки вибору. Тоді як для взаємодії кількох незалежних механічних елементів може знадобитися структура паралельної гілки.

2. Визначення основних процедур та умов передачі для складання чернетки блок-схеми

Після визначення структури вам потрібно з'ясувати основні процедури та умови передачі. Розділивши робочий процес

на менші етапи операцій, ви можете отримати процедури. Завершуючи кожну процедуру умовою передачі, ви зможете отримати чернетку робочого процесу.

3. Створення SFC програми згідно з чернеткою блок-схеми

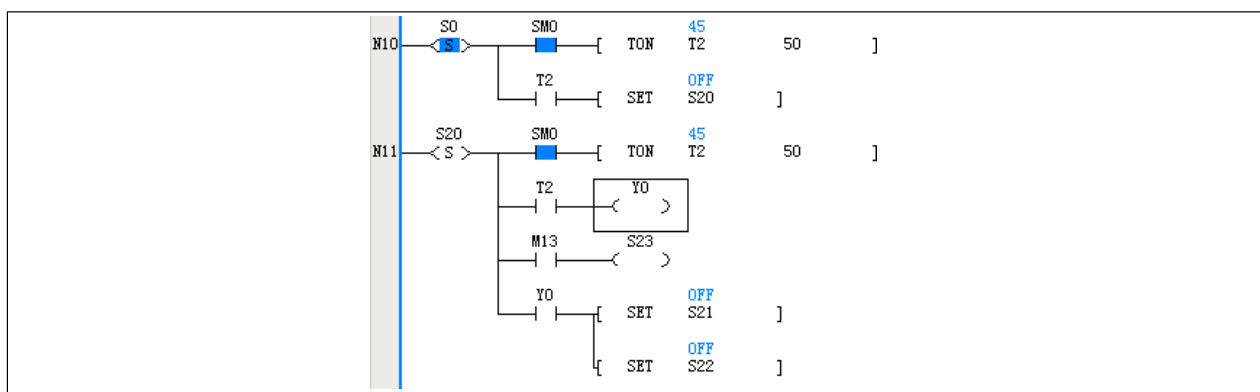


4. Повторне використання того самого програмного елемента у сусідньому кроці станів

Коли PLC виконує програму, інструкції циклічно скануються відповідно до їхнього порядку відображення. Коли програма переходить зі стану попереднього кроку до стану наступного кроку, послідовність інструкцій у стані попереднього кроку щойно завершує сканування, а послідовність інструкцій у стані наступного кроку також була увімкнена для формування керуючого виходу.

Отже, коли інструкцію STL перемикають з OFF на OFF, певні внутрішні елементи інструкції скидаються (детальніше див. розділ 5.3.1 "STL: інструкція завантаження стану SFC"), проте скидання може бути виконане лише під час наступного циклу сканування. Тобто, у момент передачі стану кроку внутрішні елементи попереднього стану кроку зберігають свої початкові дані та стани доти, доки стан кроку не буде просканий у наступному циклі.

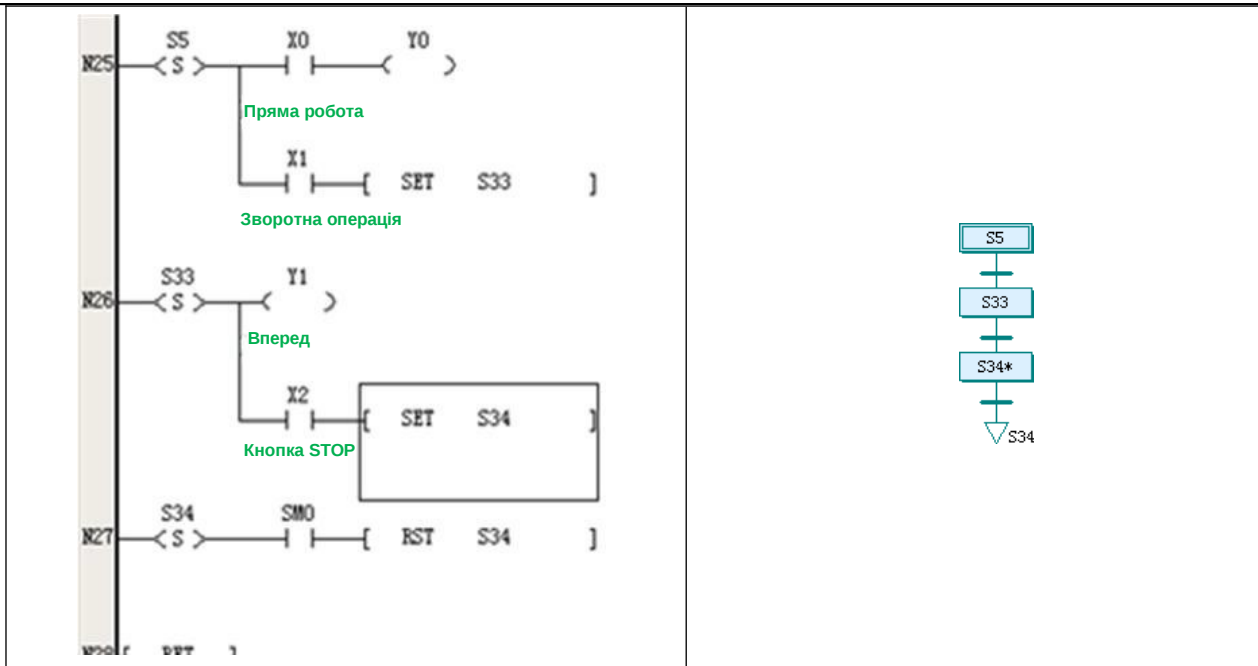
Як показано на наступній діаграмі, два сусідні стани кроків використовують один і той самий таймер: T2. Коли стан кроку перемикається з S0 на S20, елемент T2 зберігає своє значення часу та стан, через що стан кроку S20 не може виконати операцію відліку часу, як це було спочатку заплановано користувачем. Програма перескочить безпосередньо до S21 та S22. Тому вам слід звернути увагу на те, що повторне використання програмних елементів у програмі не заборонено, але найкраще не використовувати їх повторно у сусідніх станах кроків, інакше можуть виникнути неочікувані результати.



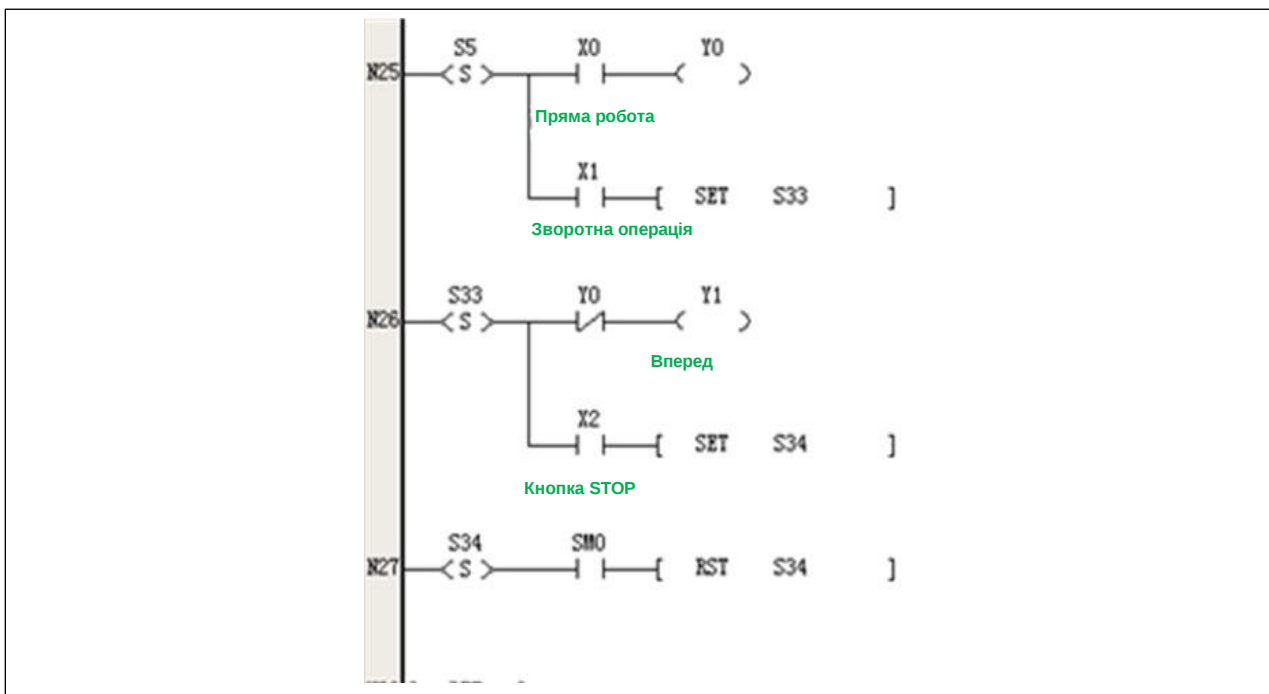
5. Нездатність заблокувати м'які елементи

Під час SFC програмування певні програмні елементи можуть суперечити один одному за деяких особливих умов переходу станів кроків. У такому разі необхідне блокування.

Взявши за приклад наступну програму прямого та зворотного ходу, де Y0 та Y1 є відповідно виходами керування прямим та зворотним ходом. X0 — це команда прямого ходу, X1 — команда зворотного ходу, а X2 — кнопка STOP. Вимагається, щоб Y0 та Y1 були заблоковані, тобто вони не можуть бути ON одночасно. Однак у цьому прикладі, коли пристрій працює у прямому напрямку, а X1 перемикається в ON для зміни стану кроку з S5 на S33, Y0 та Y1 обидва стають ON протягом одного циклу сканування програми.



Therefore, you need to add the interlock statements to the program by adding a Y0 NC contact before the Y1 output coil, as shown in the following diagram.



6. Плутанина стрибків із передачами

Переходи здебільшого використовуються між різними процесами та станами кроків, що не є сусідніми, тоді як передачі використовуються між сусідніми станами кроків. Забороняється змінювати вихідну котушку на інструкцію SET там, де слід використовувати перехід, або змінювати інструкцію SET на вихідну котушку там, де слід використовувати передачу.

7. Вибір структури паралельної гілки як точки передачі гілки, що спричиняє збій завершення процесу

A branch is selected from multiple ones. If parallel branches are included, the process may never be ended, as shown in the following figure. In the program on the left, when process 1 runs to the step state S41, the transfer condition is a parallel branch, and therefore the system does not run process 2, causing the failure to implement the transfer condition, and thus causing a process error.



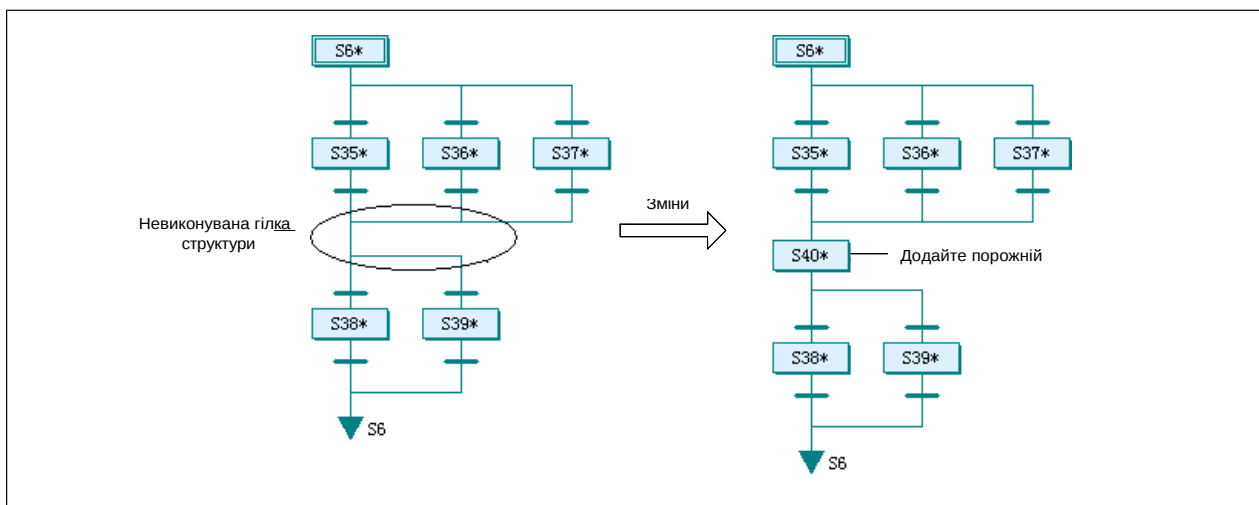
Метод модифікації показано у програмі праворуч, вам потрібно додати стан кроку S42, функція якого така сама, як у S41. Потім потрібно додати порожній стан кроку S43, який слугує структурним елементом програмування без фактичної функції. Умови переходу для S38, S41 та S43 мають бути розроблені програмістами, наприклад, усі вони можуть використовувати умови переходу початкового S41.

7.4.2 Програмування хитрощів

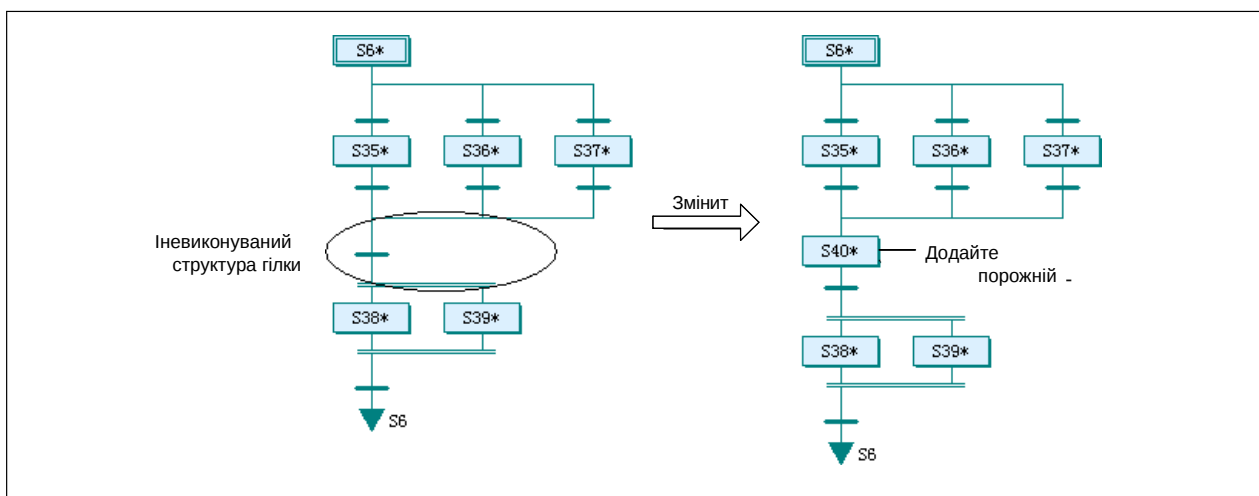
1. Використання порожніх крокових станів

Ви можете використовувати порожні стани кроків для роботи з гілками, що мають граматичні проблеми. Порожні стани кроків не забезпечують фактичної операції, але є необхідним вузлом у структурі перед наступним переходом. Ви можете побачити наступний приклад.

In the left diagram below, the selection merge is connected immediately with another selection branch structure, which is prohibited. You can modify it by adding an empty step state as shown in the right diagram.



In the left diagram below, the selection merge is connected immediately with a parallel branch structure, which is prohibited too. You can also modify it by adding an empty step state as shown in the right diagram.

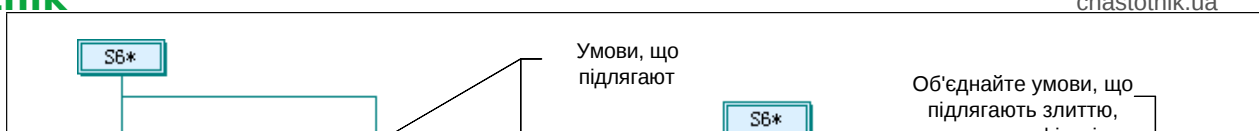


Ви можете опрацьовувати інші складні структури, такі як паралельне злиття, з'єднане з паралельними гілками, або паралельні гілки, з'єднані з гілками вибору, шляхом додавання стану порожнього кроку.

2. Об'єднання гілок та перенесення умов

Деякі гілки, що здаються складними, насправді спричинені неналежним проектуванням і можуть бути об'єднані або спрощені належним чином.

As shown in the following diagram, the designer set a selection branch first, and then set two selection branches. In fact, only



one selection branch of four branches can achieve the same, and the two levels of transfer symbols of the original design can be merged into one.

3. Використання функції для збереження даних під час зникнення живлення

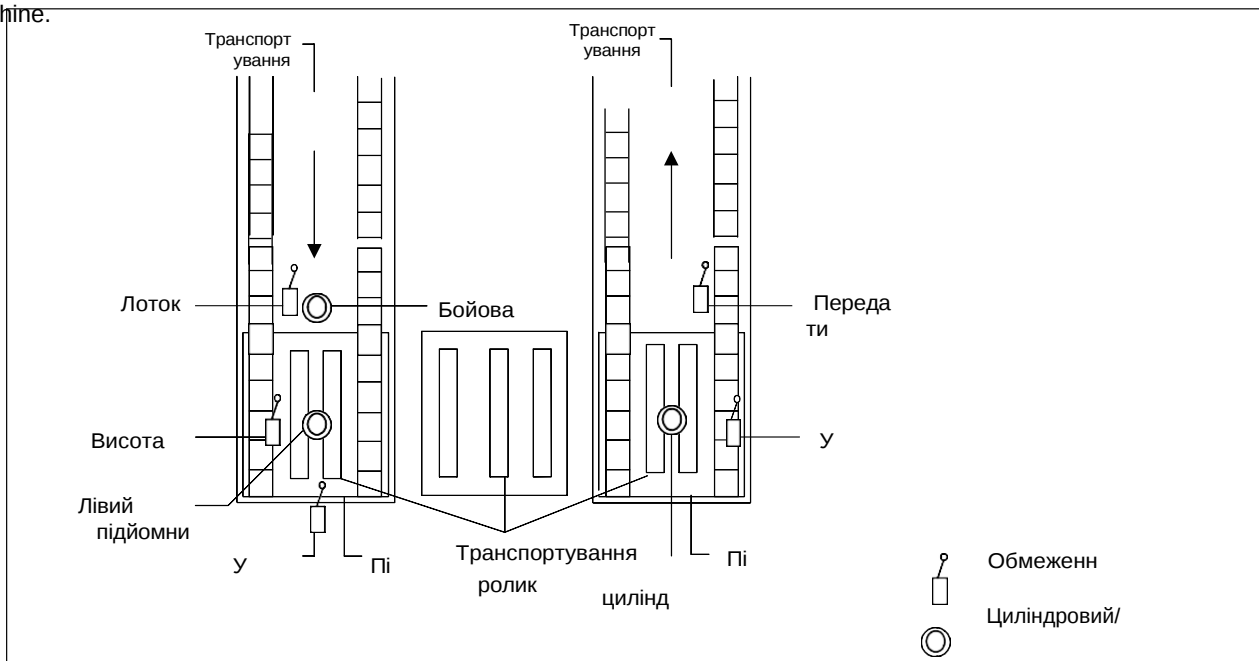
Значення елементів S можуть зберігатися функцією для збереження даних під час зникнення живлення. Таким чином, після ввімкнення живлення програма може продовжити роботу з того стану кроку, у якому відбулося зникнення живлення.

7.5 Приклади SFC програмування

Приклади у цьому розділі є лише ілюстраціями SFC програмування зі спрощеними операціями та умовами. Конфігурація пристрою є концептуальною і призначена лише для вивчення. Ви не можете застосовувати приклади програм для фактичного використання.

7.5.1 Проста послідовна структура

The following example is a workpiece tray lifting and conveying machine. This machine uses cylinder lifting devices and conveying rollers to convey the workpiece tray from one conveying belt to another. The following figure shows a top view of the machine.



Після запуску машини лоток із заготовками транспортується вздовж конвеєрної стрічки з лівого боку до входу в машину та активує кінцевий вимикач "Tray in". Якщо жоден інший лоток не займає машину, "Baffle plate" опускається, щоб дозволити лотку із заготовками увійти в машину. Коли лоток повністю увійшов у лівий підйомник, він активує кінцевий вимикач "In lift 1", підйомник піднімає лоток, доки не спрацює кінцевий вимикач "Height OK". Потім ролики переміщують лоток до підйомника з правого боку, доки не спрацює кінцевий вимикач "In lift 2". Після цього підйомник опускається, щоб перемістити лоток на конвеєрну стрічку праворуч. Коли кінцевий вимикач "Convey complete" скидається, повний процес підйому та транспортування завершується, і машина готова до наступного циклу.

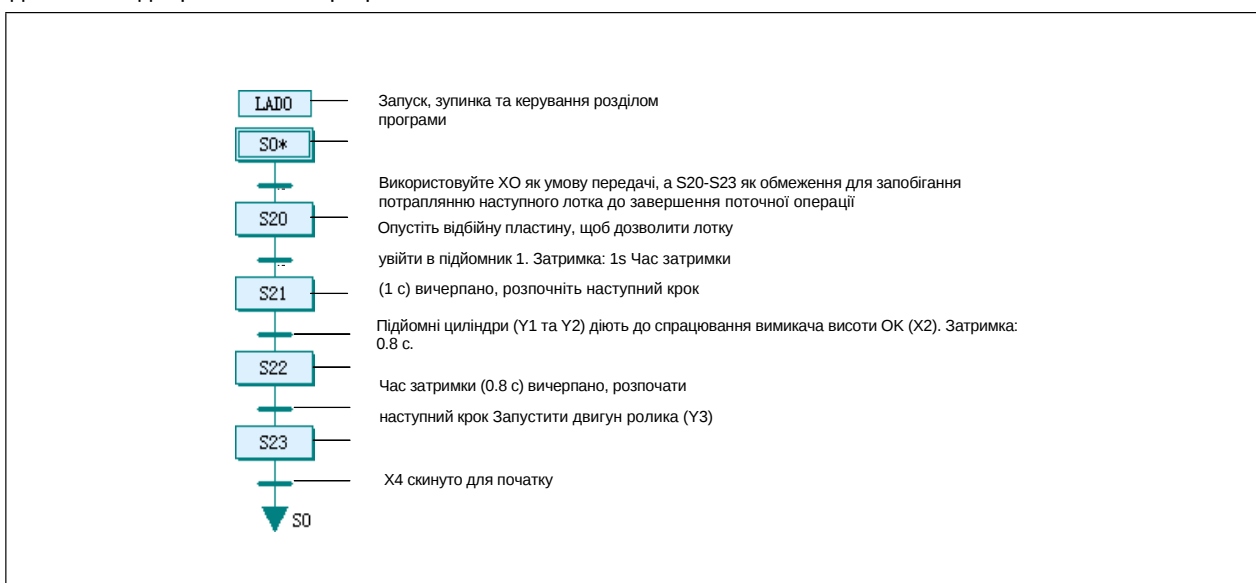
Точки I/O перелічені в наступній таблиці.

SN	Адреса	Об'єкти, що відстежуються	SN	Адреса	Об'єкти, що відстежуються
1	X0	Кінцевий вимикач лотка	8	Y0	Циліндровий електромагнітний клапан для цього дефлекторна пластина
2	X1	У ліфті 1 кінцевий вимикач	9	Y1	Циліндровий електромагнітний клапан для цього лівий підйом
3	X2	Границя ОК вимикача	10	Y2	Циліндровий електромагнітний клапан для цього

		кінцевого				правий підйом
4	X3	У ліфті 2 кінцевий вимикач		11	Y3	Двигун конвеєрного ролика двигун контактор
5	X4	Виконати повне перемикавання		12	Y4	Контактор двигуна для лівого конвеєрної стрічка
6	X5	Запустити перемикач		13	Y5	Контактор двигуна для правого конвеєрної стрічка
7	X6	Допоміжний сигнал з аварійний вимикач				

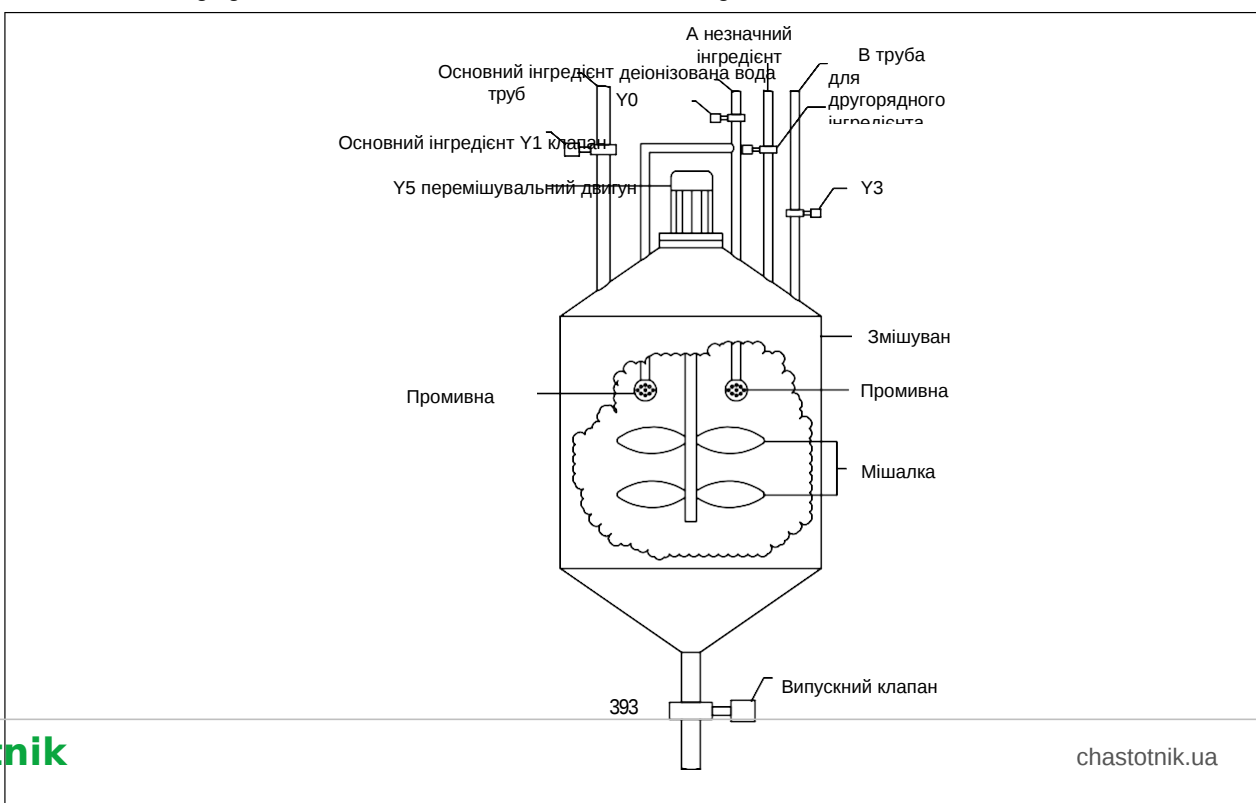
Це простий послідовний процес. Кожен лоток транспортується через кілька послідовних процедур, без інших варіантів чи паралельних процедур, і без паралельного потоку. Написання програми з SFC було б швидшим і зрозумілішим, ніж традиційний логічний метод.

Наведені нижче діаграми є SFC програмою та її LAD аналогом.



7.5.2 Гілка вибору структури

The following example is a material mixing process flow. Through this flow, two kinds of products, namely A and B, are produced. The following figure shows the illustration of the manufacturing device.



Щоб розпочати роботу, першим кроком є вибір типу продукту, А або В, через сенсорний екран для наступної партії продуктів, а потім запуск виробництва. Другим кроком є додавання основних інгредієнтів, доки вага доданого інгредієнта не досягне 2000kg. Третім кроком є додавання другорядних інгредієнтів, а саме додавання А другорядних інгредієнтів для продукту типу А або В другорядних інгредієнтів для продукту типу В, доки доданий другорядний інгредієнт А або В не досягне відповідно 500kg. Четвертим кроком є змішування інгредієнтів протягом 20 хвилин. П'ятим кроком є вивантаження матеріалів, доки залишок матеріалу не стане меншим за 20kg і час затримки не мине. Як тільки ці кроки будуть завершені, машина готова до наступного циклу.

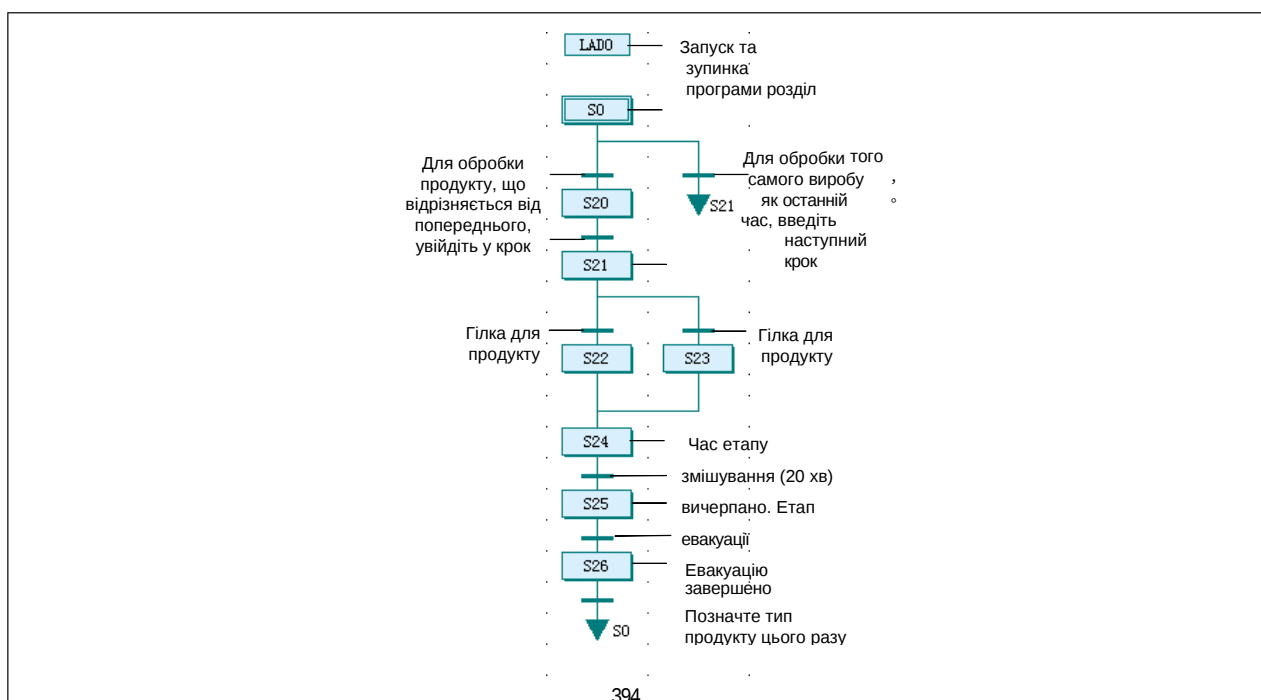
Якщо це перший запуск машини у виробництво або тип продукту попередньої партії відрізняється від того, що буде вироблятися, необхідно відкрити клапан деіонізованої води та клапан спорожнення, щоб промити машину протягом 5 хвилин перед додаванням основних інгредієнтів.

Точки I/O перелічені в наступній таблиці.

SN	Адреса	Об'єкти, що відстежуються	SN	Адреса	Об'єкти, що відстежуються
1	X0	Клапан деіонізованої води відкритий	10	X11	Клапан евакуації відкритий
2	X1	Клапан деіонізованої води закритий	11	X12	Евакуаційний клапан закритий
3	X2	Клапан основного інгредієнта відкритий	12	Y0	Електромагнітний клапан для деіонізованої води
4	X3	Клапан основного інгредієнта закрито	13	Y1	Електромагнітний клапан для головного інгредієнта
5	X4	Допоміжний інгредієнт А клапан відкритий	14	Y2	Електромагнітний клапан для незначного інгредієнта А
6	X5	Додатковий інгредієнт А клапан закритий	15	Y3	Електромагнітний клапан для незначних інгредієнтів В
7	X6	Допоміжний інгредієнт В клапан відкритий	16	Y4	Електромагнітний клапан для евакуації
8	X7	Додатковий інгредієнт В клапан закритий	17	Y5	Контактор двигуна змішування contactor
9	X10	Змішувальний двигун працює			

Очевидно, що це потік із розгалуженням вибору. Ви можете вибрати лише один тип продукту, А або В, протягом одного циклу виробництва продукції. Ви можете змінити тип продукту лише після завершення циклу виробництва. Водночас у процесі передбачено структуру вибору та переходу, а саме етап промивання.

Наведені нижче діаграми є програмою SFC та її аналогом LAD part.

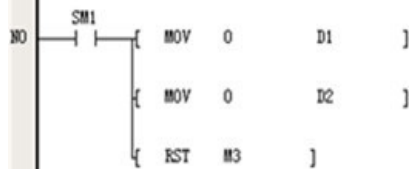


/*Скидання M1-M3*/

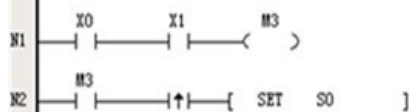
/*D1 означає останній виріб. 0: виріб А. 1: виріб В.*/

/*D2 означає наступний виріб. 0: виріб А. 1: виріб В.*/

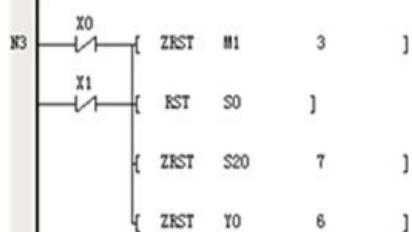
/*M3 — це прапорець запуску.*/



/*Початок роботи*/



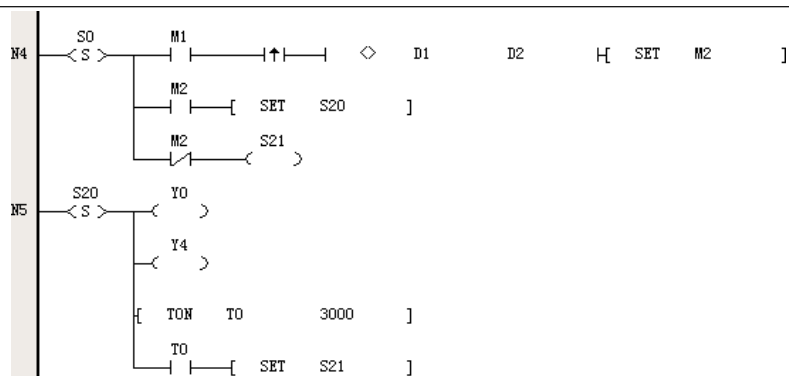
/*Зупинить роботу*/



/*Вибір продукту робота*/

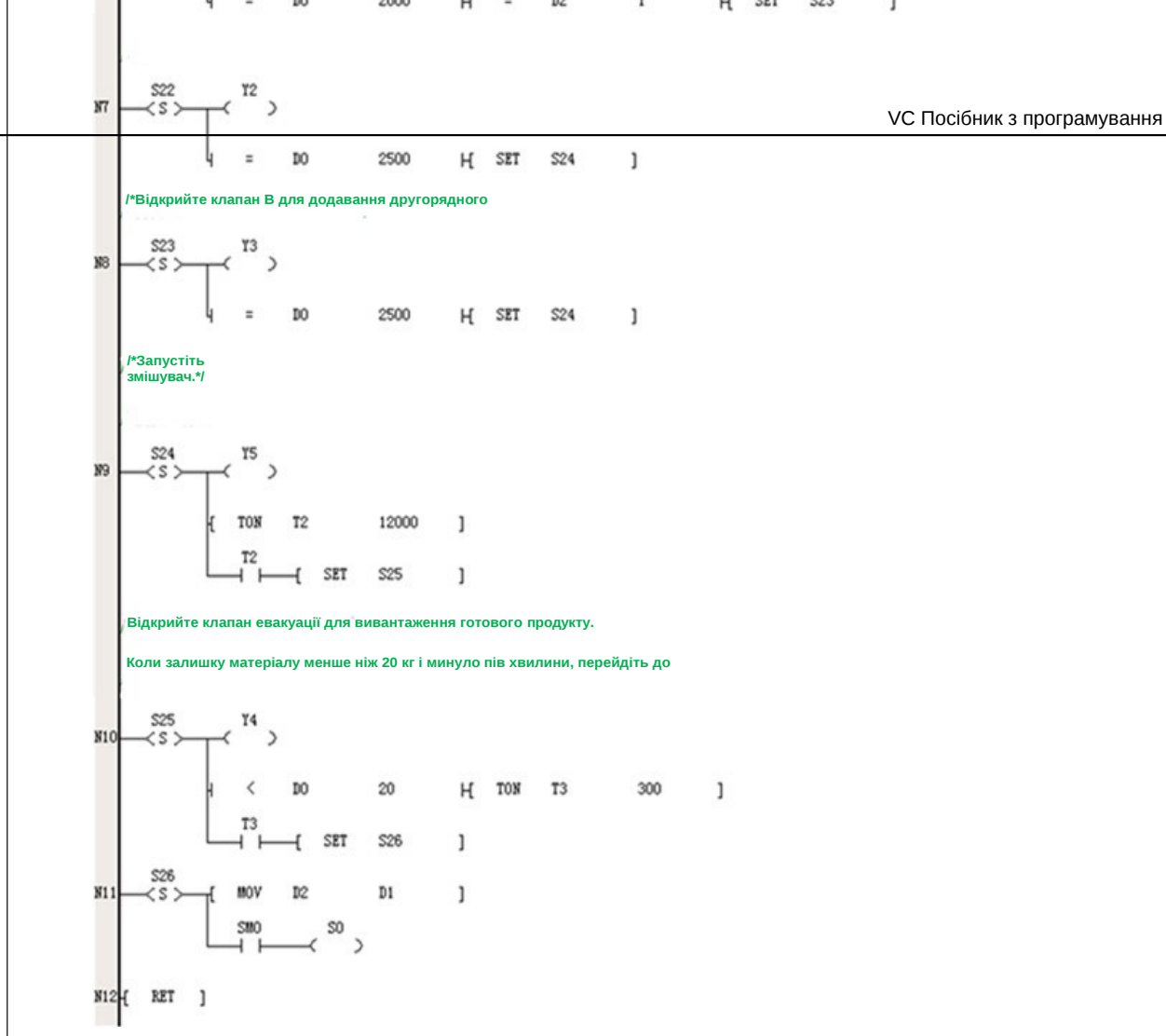
/*M1 — це біт керування екраном HMI інтерфейсу.*/

/*M2 означає, що два сусідні вироби різні.*/



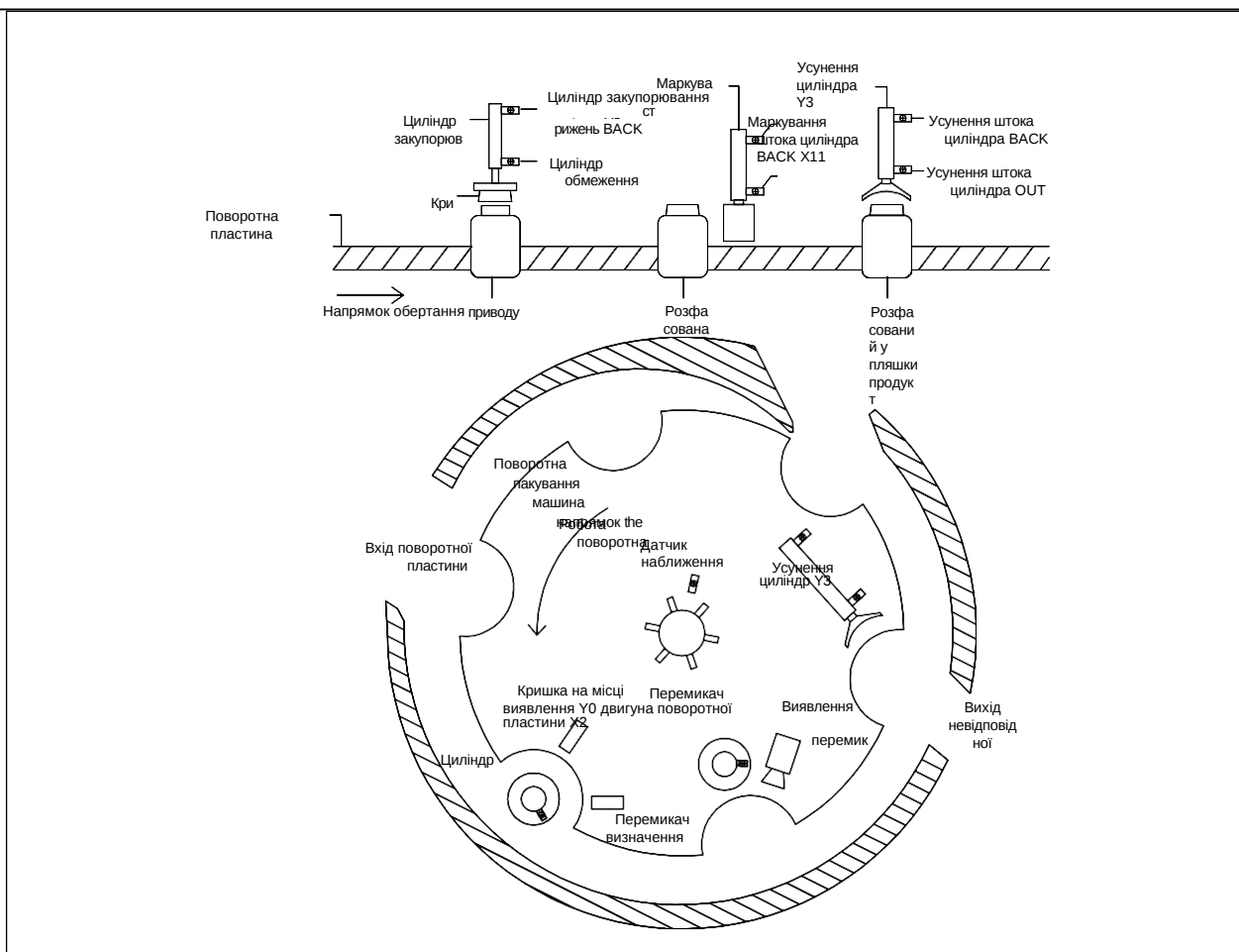
/*Відкрийте значення основного інгредієнта для

/*Відкрийте клапан А для додавання другорядного



7.5.3 Паралельна гілка структури

Наступний приклад — це пакувальник пляшок. Цей пакувальник герметизує пляшки та наклеює на них етикетки. Тим часом він перевіряє кришки пляшок та етикетки, щоб браковані вироби були усунені під час третьої процедури, тоді як якісні вироби продовжують рух до наступного етапу робочого процесу. Якщо з попереднього етапу робочого процесу не надходить жодної пляшки, пакувальник не виконує жодного герметизування чи маркування. Ці три процедури виконуються одночасно, і кожна пляшка переміщується з однієї позиції в іншу щоразу, коли обертається поворотна плита. На наступному рисунку показано ілюстрацію пакувальника.



During the operation, the rotary plate rotates one step each time, which is detected by the X0 limit switch. The rotary plate stays at each step long enough for all the three procedures, driven by cylinders, finished. The cylinder rod OUT signal and cylinder rod BACK signal are monitored respectively.

Точки I/O перелічені в наступній таблиці.

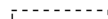
SN	Адреса	Об'єкти, що відстежуються	SN	Адреса	Об'єкти, що відстежуються
1	X0	Станція поворотної пластины датчик наближення перемикач	8	X10	Шток циліндра затискання BAC
2	X1	Пляшка в позиції виявлення перемикач	9	X11	Маркування штока циліндра BAC
3	X2	Виявлення положення ковпачка перемикачем	10	X12	Усунення штока циліндра BAC
4	X3	Виявлення маркування перемикача	11	Y0	Поворотна пластина двигуна
5	X5	Усунення штока циліндра OUT	12	Y1	Закриття циліндра
6	X6	Маркування штока циліндра OUT	13	Y2	Маркування циліндра
7	X7	Шток циліндра затискання OUT	14	Y3	Усунення циліндра

Очевидно, що це потік із структурою паралельних гілок. Після повороту поворотної пластины виконуються операції на трьох станціях. Після того, як усі операції виконані на трьох станціях, пристрій виконує наступні операції. Наступні діаграми є SFC програмою та її LAD аналогом.

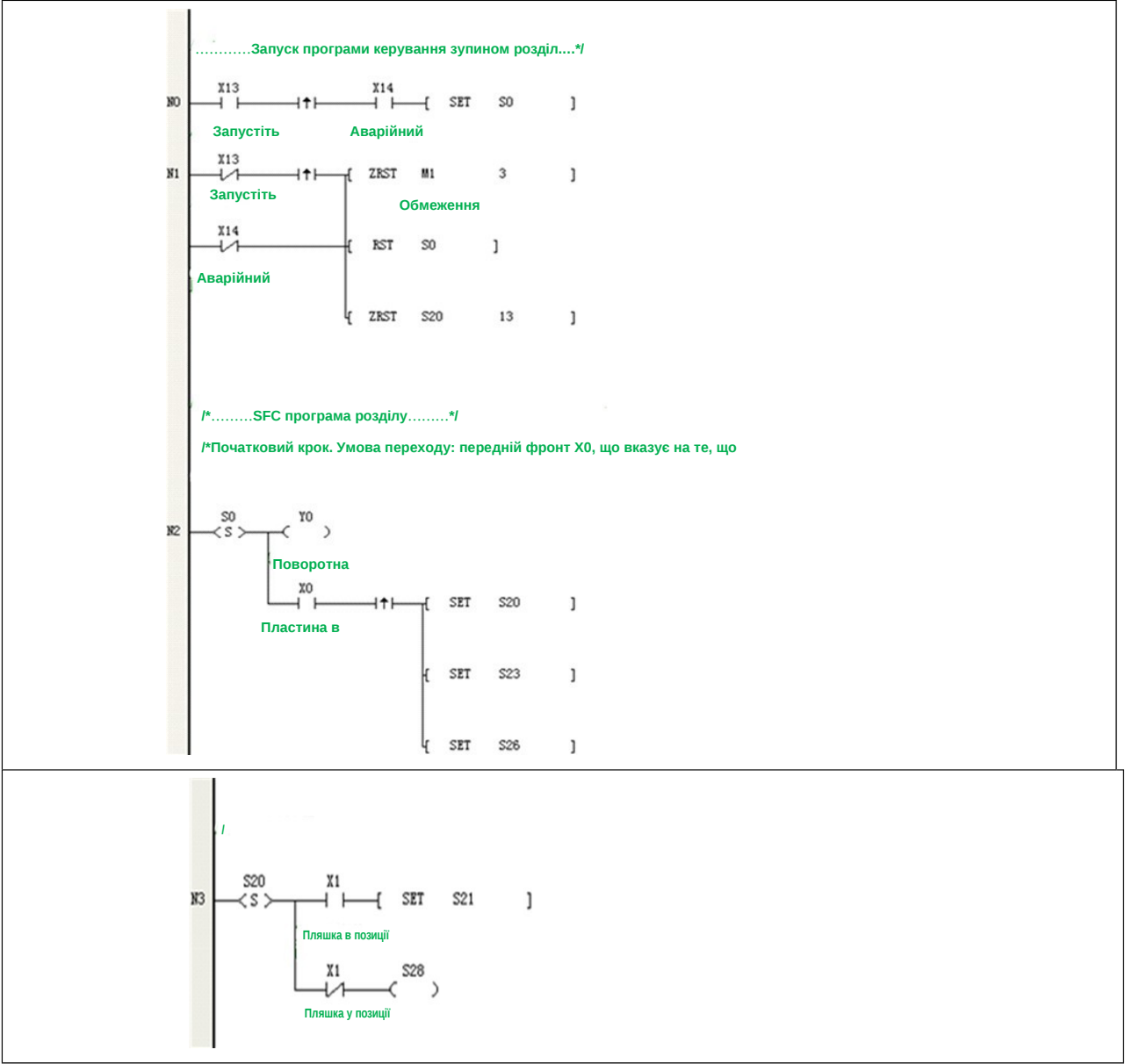
У програмі M1-M3 є прапорцями кваліфікації для процедур закупорювання, маркування та усунення відповідно.

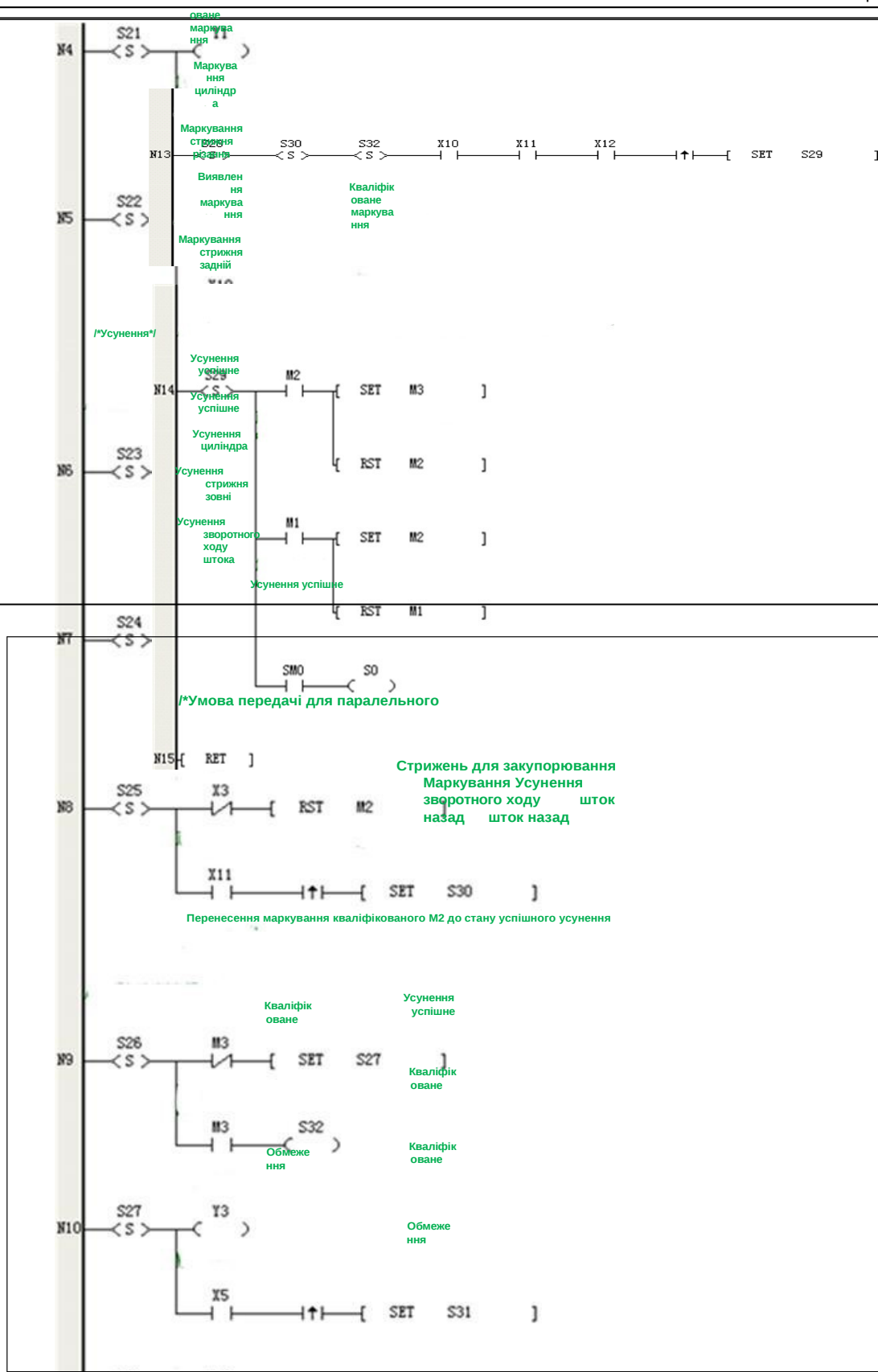
Коли процедура закупорювання виконується до S22, X2 перевіряє, чи є закупорювання якісним. Якщо так, встановлюється відповідний прапор якості M1. Коли процедура маркування виконується до S25, X3 перевіряє, чи є маркування якісним. Якщо ні, M2 скидається. Після завершення всіх процедур, на кроці S29, стан M2 передається до M3, а стан M1 передається до M2.

Процедура закупорювання діє відповідно до стану X1. Якщо X1 вказує на те, що пляшка не знаходиться в позиції, закупорювання не виконується. Процедура маркування діє відповідно до стану M2. Якщо M2 дорівнює OFF, це вказує на те, що пляшка в позиції є бракованою, і маркування не виконується. Процедура відбраковування діє відповідно до M3. Якщо M3 дорівнює ON, це вказує на те, що пляшка є якісною, і відбраковування не виконується, і навпаки. В обох випадках M3 скидається в S32 для підготовки до наступної процедури.



Паралельне злиття
Перенести стани прапорців
кваліфікації
від M1 до M2 та від M2 до M3





Розділ 8 Посібник з експлуатації для високошвидкісного входу функція

У цьому розділі детально описано використання та примітки щодо функції високошвидкісного входу, включаючи високошвидкісні лічильники та захоплення імпульсів.

Розділ 8 Посібник з експлуатації для високошвидкісного входу функція.....	274
8.1 Високошвидкісний лічильник.....	275
8.1.1 Конфігурація.....	275
8.1.2 Зв'язок між високошвидкісним лічильником та SM допоміжним реле.....	276
8.1.3 Як використовувати високошвидкісні лічильники.....	278
8.1.4 Примітки щодо високошвидкісних лічильників серії VC1.....	281
8.2 Функція захоплення зовнішнього імпульсу функція.....	282
8.3 Примітки щодо застосування високошвидкісного входу.....	282

8.1 Високошвидкісний лічильник

8.1.1 Конфігурація

Вбудований високошвидкісний лічильник для мікро-приводів серії VC PLCs налаштовується наступним чином:

● Таблиця конфігурації високошвидкісного лічильника

Вхідна точка	Лічильник	X0	X1	X2	X3	X4	X5	X6	X7	Макс. частота (кГц)		
										VC5 VC3	VC2 VC1	VC1S
Одноточковий вхідний лічильник	C236	Вгору/Вниз								200	50	10
	C237		Вгору/Вниз									
	C238			Вгору/Вниз								
	C239				Вгору/Вниз							
	C240					Вгору/Вниз					10	/
	C241						Вгору/Вниз					
	C242							Вгору/Вниз			10	/
	C243								Вгору/Вниз			
	C244	Вгору/Вниз		Скидання							50	10
	C245					Вгору/Вниз		Скидання			10	
	C246	Вгору/Вниз		Скидання	Початок						50	
	C247					Вгору/Вниз		Скидання	Початок		10	
Двополюсний лічильник	C248	Вгору	Вниз							200	30	10
	C249			Вгору	Вниз						5	
	C250					Вгору	Вниз				5	
	C251							Вгору	Вниз		5	
	C252	Вгору	Вниз	Скидання							30	10
	C253					Вгору	Вниз	Скидання			5	
	C254	Вгору	Вниз	Скидання	Початок						30	
	C255					Вгору	Вниз	Скидання	Початок		5	
Трифазний лічильник	C256	А фаза	В фаза							200	30	5
	C257			А фаза	В фаза							
	C258					А фаза	В фаза					
	C259							А фаза	В фаза		5	/

Два фазовий лічильник входу	C260	фаза А	фаза В	Скидання						30	5
	C261				фаза А	фаза В	Скидання			5	
	C262	фаза А	фаза В	Скидання	Початок					30	
	C263					фаза А	фаза В	Скидання	Початок	5	
Імпульс функція захоплення	200									10	10
	200									10	
Зовнішнє переривання номер		0/8	1/9	2/10	3/11	4/12	5/13	6/14	7/15		

У режимах, наведених у попередній таблиці, високошвидкісні лічильники діють відповідно до певного входу та обробляють високошвидкісну дію згідно з перериваннями. Дія підрахунку не пов'язана з PLC циклом сканування.

Цей тип високошвидкісних лічильників належить до 32-бітових двонаправлених лічильників. Залежно від різних методів перемикавання рахунку вгору/вниз, їх можна розділити на наступні чотири категорії:

Режим рахунку	Підрахунок дії
Однофазний один порт рахувати вхід	Лічильники C236–C247 здійснюють прямий/зворотний відлік, коли SM236–SM247 мають значення ON/OFF.
Однофазний двонаправлений підрахунок вхід	Відповідно до дії рахунку вгору або вниз, лічильники C248–C255 автоматично інкрементуються/декрементуються. Ви можете дізнатися поточний напрямок рахунку відповідного лічильника через SM248–SM255. Лічильник рахує вгору, коли елемент SM є OFF, або рахує вниз, коли OFF.
Двофазний лічильний вхід	Коли SM100–SM103 встановлено в OFF, лічильники C256–C263 автоматично інкрементуються та декрементуються відповідно до двофазного входу. Ви можете дізнатися поточний напрямок рахунку відповідного лічильника через SM256–SM263. Лічильник рахує вгору, коли елемент SM дорівнює OFF, або рахує вниз, коли OFF. Напрямок рахунку визначено наступним чином:
Двофазний квадратурний вхід підрахунку частоти	Коли SM100–SM103 встановлено в ON, лічильники C256–C263 автоматично виконують рахунок вгору та вниз відповідно до двофазного входу. Ви можете дізнатися поточний напрямок рахунку відповідного лічильника через SM256–SM263. Лічильник рахує вгору, коли елемент SM має значення OFF, або рахує вниз, коли OFF. Напрямок рахунку визначено наступним чином:

8.1.2 Зв'язок між високошвидкісним лічильником та SM допоміжним реле

- Спеціальне допоміжне реле для керування напрямком лічильника

Тип	Лічильник SN	Керування вгору/вниз

	C236	SM236
	C237	SM237
	C238	SM238
	C239	SM239

Однофазний одноточковий лічильний вхід	C240	SM240
	C241	SM241
	C242	SM242
	C243	SM243
	C244	SM244
	C245	SM245
	C246	SM246
	C247	SM247

- Спеціальне допоміжне реле для моніторингу напрямку лічби

- Спеціальне допоміжне реле для керування чотирьох-частотним режимом

Тип	Лічильник SN	Чотиричастотний налаштування
Двофазний лічильний вхід	C256	SM100
	C257	SM101
	C258	SM102
	C259	SM103
	C260	SM100
	C261	SM102
	C262	SM100
	C263	SM102

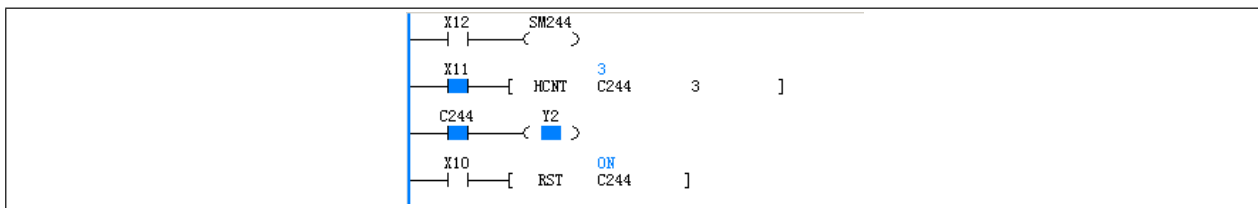
Тип	Лічильник SN	Вгору/вниз монітор
Однофазний двонаправлений лічильний вхід	C248	SM248
	C249	SM249
	C250	SM250
	C251	SM251
	C252	SM252

Тип	Лічильник SN	Вгору/вниз монітор
Двофазний лічильний вхід	C253	SM253
	C254	SM254
	C255	SM255
	C256	SM256
	C257	SM257
	C258	SM258
	C259	SM259
	C260	SM260
	C261	SM261
	C262	SM262
	C263	SM263

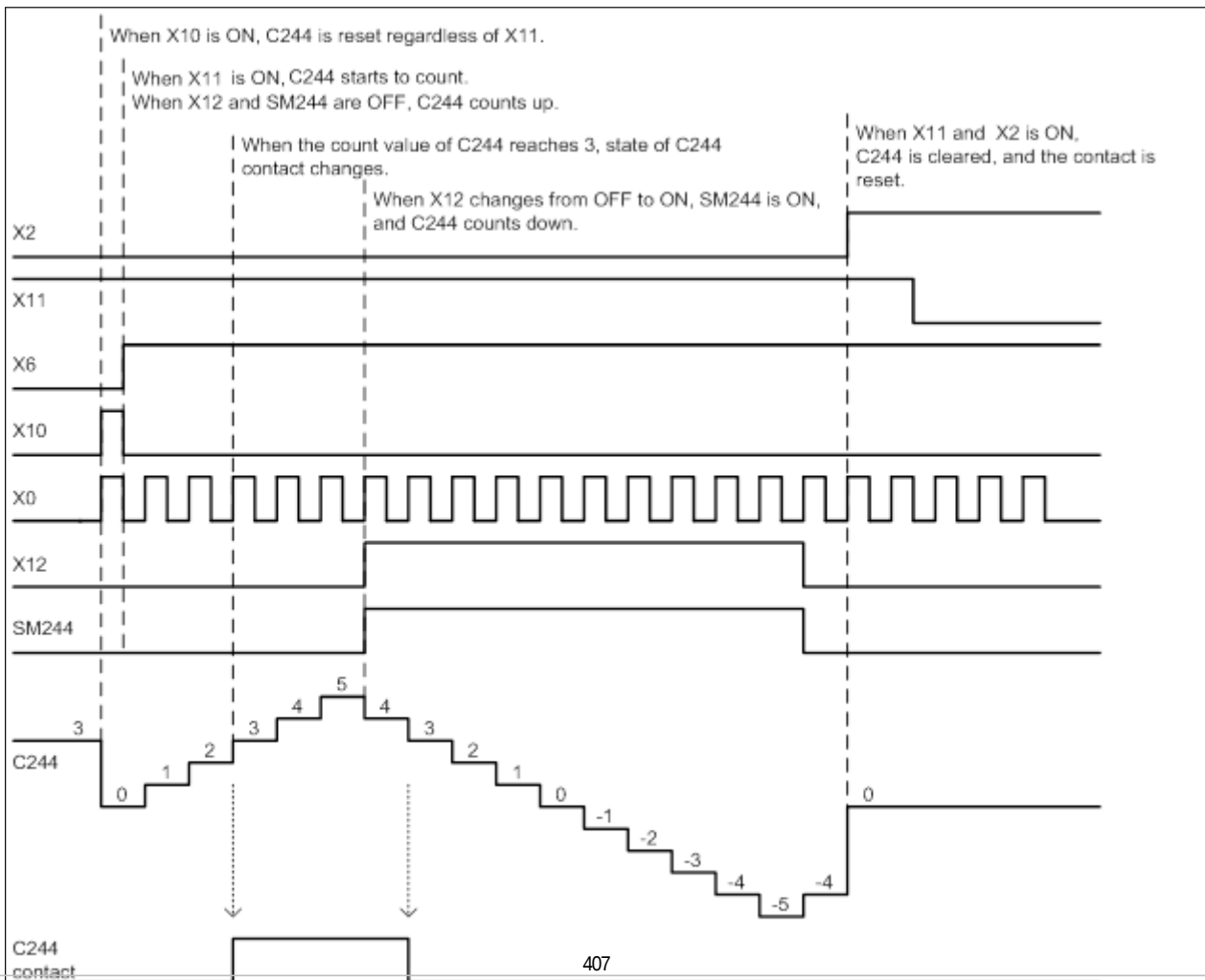
8.1.3 Як використовувати високошвидкісні лічильники

- Як використовувати однофазні одноточкові вхідні високошвидкісні лічильники

Характеристики: А однофазний одноточковий вхід високошвидкісного лічильника починає рахувати лише тоді, коли імпульсний вхід змінюється з OFF на ON, а напрямок рахунку визначається відповідним спеціальним допоміжним реле SM. Приклад дії показано на наступному рисунку.



Часову діаграму послідовності дій контактів у програмі наведено на наступному рисунку.



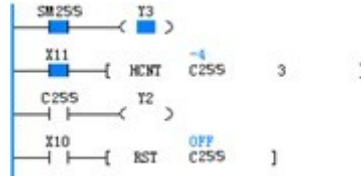
Note:

- Counter input points: X0.
- High-Speed counters, when used in DHSCS, DHSCR, DHSZ, DHSP, and DHST instructions, can trigger operations or controls free from the scan cycle.

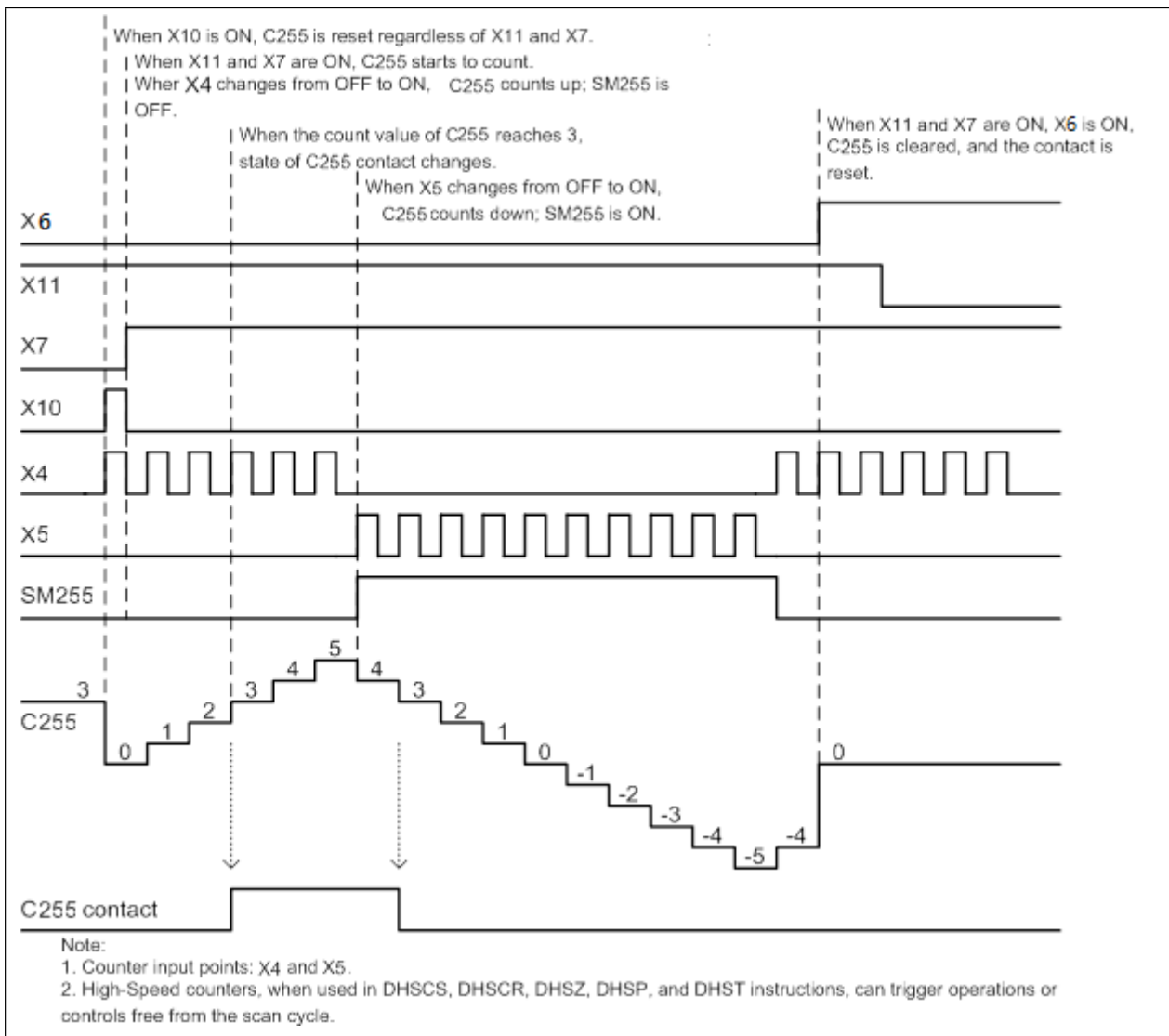
- Як використовувати однофазний двонаправлений лічильний вхід високошвидкісних лічильників

Характеристики: А однофазний двонаправлений лічильний вхід високошвидкісного лічильника починає рахувати лише тоді, коли імпульсний вхід змінюється з OFF на ON, а напрямок рахунку визначається відповідними двома вхідними точками. Стан високошвидкісного лічильника визначається відповідним спеціальним допоміжним реле SM.

Приклад дії показано на наступному рисунку.



Часова діаграма послідовності дій контактів у програмі наведена на наступному рисунку.



- Як використовувати високошвидкісні лічильники з двофазним лічильним входом

Характеристики: А високошвидкісний лічильник з двофазним імпульсним входом починає рахувати лише тоді, коли вхідний імпульс змінюється з OFF на ON, а напрямок рахунку визначається різницею фаз між двома вхідними точками. Стан високошвидкісного лічильника визначається його відповідним спеціальним допоміжним реле SM.

Приклад цієї дії показано на наступному рисунку.

When X10 is ON, C262 is reset regardless of X11 and X3.

When X11 and X3 are ON, C262 starts to count.

When X0 is ON and X1 changes from OFF to ON, C262 counts up; SM262 is OFF.

When the count value of C262 reaches 3, state of C262 contact changes.

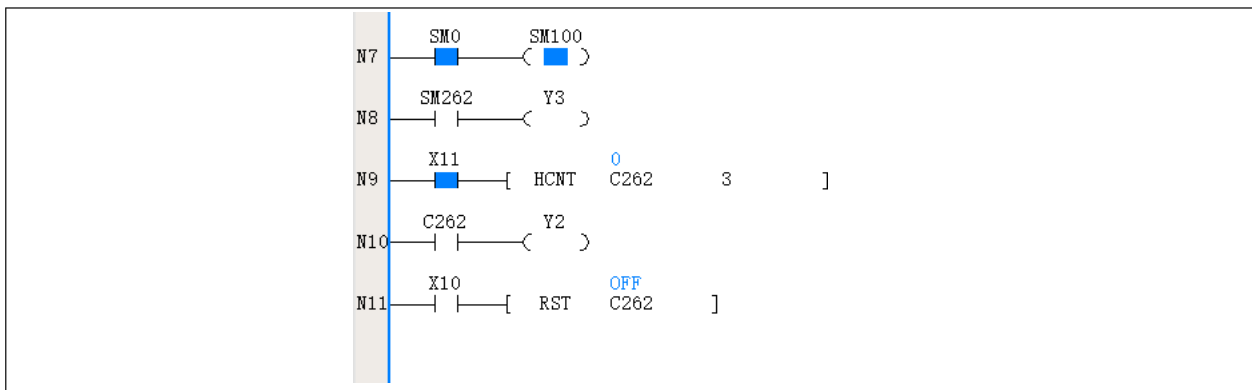
When X0 is ON and X1 changes from ON to OFF, C262 counts down; SM262 is ON.

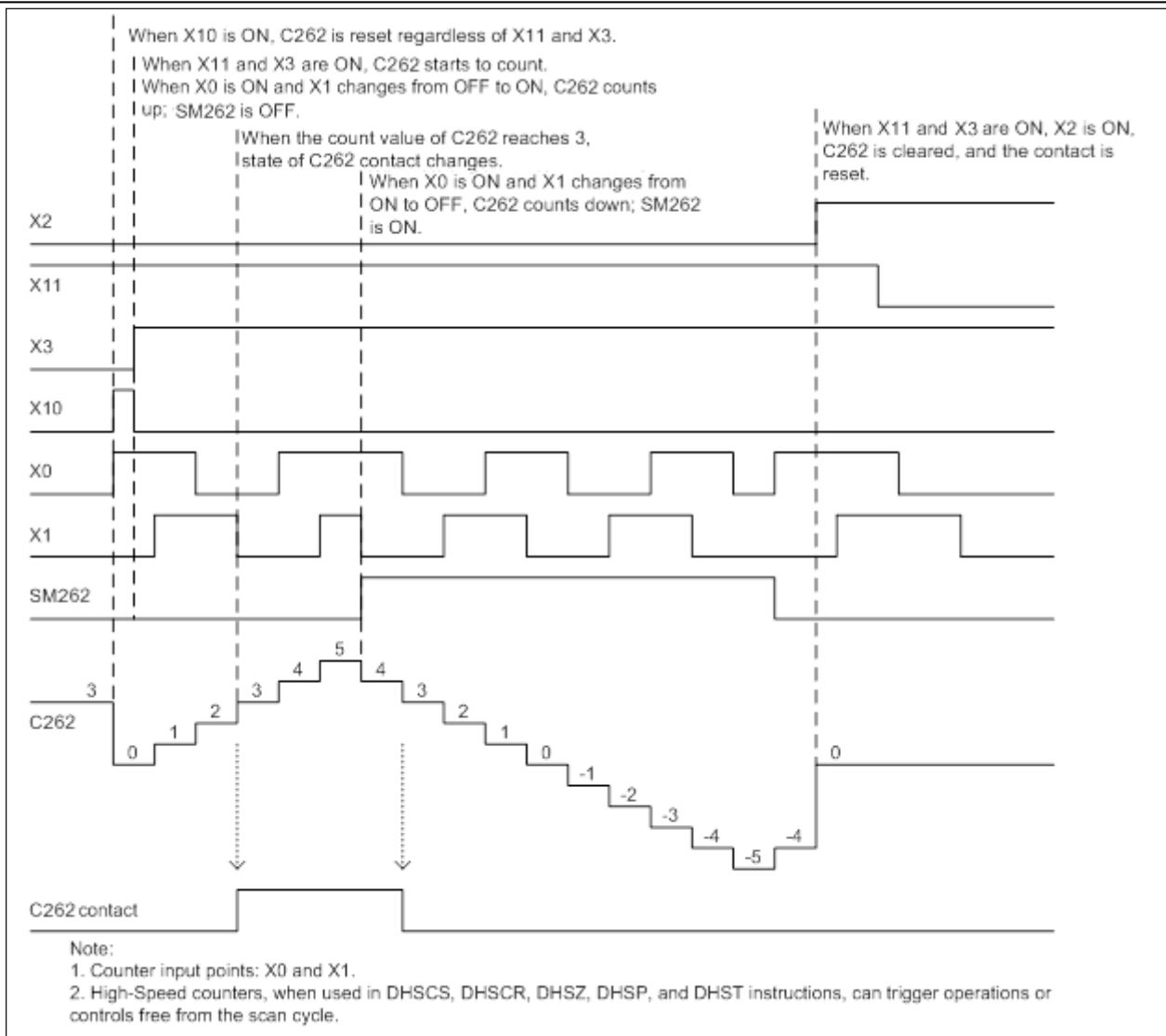
When X11 and X3 are ON, X2 is ON, C262 is cleared, and the contact is reset.

- Як використовувати високошвидкісні лічильники з двофазним квадратурним частотним входом

Характеристики: А двофазний квадратурний частотний лічильний вхід високошвидкісного лічильника починає рахунок, коли імпульсний вхід змінюється з OFF на ON або з ON на OFF, а напрямок рахунку визначається різницею фаз між двома вхідними точками. Стан високошвидкісного лічильника визначається його відповідним спеціальним допоміжним реле SM.

Приклад дії показано на наступному рисунку.





8.1.4 Примітки щодо високошвидкісних лічильників серії VC1

- Класифікація високошвидкісних лічильників

C0-C235 — це програмні лічильники, C236-C263 — це апаратні лічильники. Будь ласка, виберіть їх відповідно до режиму використання.

- Макс. частота сума

Максимальна сумарна частота не може перевищувати 60kHz, коли одночасно використовуються кілька високошвидкісних лічильників (апаратний режим підрахунку) або коли одночасно використовуються високошвидкісні лічильники (апаратний режим підрахунку) та інструкція SPD.

Сумарна вхідна частота при одночасному використанні кількох програмних високошвидкісних лічильників або при одночасному використанні високошвидкісних лічильників та інструкції SPD наведена в наступній таблиці.

умови використання	Загальна вхідна частота
DHSCS, DHSCR, DHSCI, DHSZ, DHSP та DHST не використовуються	≤60 кГц
DHSCS, DHSCR, DHSCI, DHSP, DHST використовуються	≤30 кГц
DHSZ використовується	≤20 кГц

- Макс. частота апаратних лічильників

Лічильники C236, C237 та C256 є апаратними лічильниками. Серед яких:

- C236 та C237 — це однофазні лічильники з макс. частотою підрахунку 50kHz.

- C256 — це двофазний лічильник з максимальною частотою підрахунку 30kHz.

● Макс. частота програмних лічильників

Високошвидкісні лічильники, що використовуються в інструкціях DHSCS, DHSCR, DHSCI, DHSP або DHST, працюють у режимі програмного лічильника. Максимальна вхідна частота для однофазних лічильників та двофазних лічильників становить відповідно 10kHz та 5kHz.

При використанні в інструкції DHSZ максимальна вхідна частота для однофазних лічильників та двофазних лічильників становить 5 кГц та 4 кГц відповідно.

8.2 Функція захоплення зовнішнього імпульсу функція

Вхідні точки, що забезпечують функцію захоплення зовнішніх імпульсів, це X0–X7. Відповідні програмні елементи SM наведено нижче:

Апаратна точка	SM елемент
X0	SM90
X1	SM91
X2	SM92
X3	SM93
X4	SM94
X5	SM95
X6	SM96
X7	SM97

□ Примітка

1. Коли сигнали зовнішніх входів змінюються з OFF на ON, програмні елементи SM відповідних портів встановлюються в ON.
2. SM90–SM97 очищуються, коли користувач запускає програму.
3. Під час використання захоплення імпульсів лічильники повинні дотримуватися граничної суми частоти вхідних імпульсів кожної серії PLC, інакше може виникнути відхилення.
4. Коли високошвидкісний лічильник або інструкція SPD, що відповідає HCNT, використовуються на одній і тій самій вхідній точці, захоплення імпульсів стає недейсним після першого циклу сканування, незалежно від того, чи є інструкція дійсною.

8.3 Примітки щодо застосування високошвидкісного входу

Вхідні точки X0–X7 використовуються як вхідні сигнали у таких функціях, як високошвидкісний лічильник, SPD інструкція, захоплення імпульсів та зовнішнє переривання. Проте ці функції не можуть використовуватися одночасно, оскільки багато різних функцій можуть використовувати одну або декілька вхідних точок. Тому під час PLC програмування доступна лише одна з кількох функцій, які може забезпечити вхідна точка. Якщо вхідні точки X0–X7 використовуються повторно у користувацькій програмі, програма користувача не може бути скопійована.

Розділ 9 Використання переривань

У цьому розділі детально описано механізми, процедури обробки та способи використання різноманітних переривань.

Розділ 9 Використання переривань.....	283
9.1 Перервати програму.....	284
9.2 Обробка переривання події.....	286
9.3 Використання переривання за таймером interrupt.....	286
9.4 Використання зовнішніх переривань.....	289
9.5 Використання високошвидкісного лічильника переривання.....	290
9.6 Використовуючи РТО виведіть завершення переривання.....	291
9.7 Використання переривання на основі послідовного порту.....	292

9.1 Перервати програму

Коли виникає подія переривання, звичайний цикл сканування переривається, а програма переривання викликається та виконується в пріоритетному порядку, що називається механізмом обробки переривань. Для завдань керування, критичних до часу та таких, що запускаються подіями, часто потрібно використовувати цей спеціальний механізм обробки системи.

Система надає багато видів програмованих ресурсів переривань. Кожен вид ресурсу переривань може ініціювати тип подій переривання, і кожен тип подій переривання має незалежну нумерацію.

Для обробки певної події переривання необхідно скопіювати програму обробки, тобто програму переривання, яка є незалежним POU, що становить частину користувацької програми.

Номер події має бути призначений для кожної програми переривання, щоб пов'язати користувацьку програму переривання з подіями переривання, визначеними номером події. Під час відповіді на запит переривання події переривання система викликає відповідну користувацьку програму переривання на основі номера події переривання.

Наступний перелік показує ресурси переривань, що надаються мікро-VC серії PLCs:

Подія номер	Подія переривання event	Увімкнення SM	Подія число	Подія переривання event	Увімкнення переривання SM
0	Передній фронт сигналу на вході X0 імпульсу переривання	SM25	33	Високошвидкісний лічильник переривання 0	SM58
1	Передній фронт сигналу на вході X1 імпульсу переривання	SM26	34	Високошвидкісний лічильник переривання 1	SM58
2	Передній фронт сигналу на вході X2 імпульсу переривання	SM27	35	Високошвидкісний лічильник переривання 2	SM58
3	Передній фронт входу X3 сигналу переривання	SM28	36	Високошвидкісний лічильник переривання 3	SM58
4	Передній фронт сигналу на вході X4 імпульсу переривання	SM29	37	Високошвидкісний лічильник переривання 4	SM58
5	Передній фронт сигналу на вході X5 імпульсу переривання	SM30	38	Високошвидкісний лічильник переривання 5	SM58
6	Передній фронт входу X6 сигналу переривання	SM21	39	Високошвидкісний лічильник переривання 6	SM58
7	Передній фронт вхідного сигналу X7 імпульсу переривання	SM32	40	Високошвидкісний лічильник переривання 7	SM58
8	Задній фронт сигналу на вході X0 імпульсу переривання	SM33	41	Інтерполяція завершення переривання 1	SM69
9	Спадаючий фронт входу X1 переривання	SM34	42	Інтерполяція завершення переривання 2	SM69
10	Вхід X2 по спадному фронту переривання	SM35	43	Інтерполяція завершення переривання 3	SM69
11	Задній фронт входу X3 переривання	SM36	44	Інтерполяція завершення переривання 4	SM69
12	Вхід X4 по спадаючому фронту переривання	SM37	45	Переривання за позицією 0 від високошвидкісний вихід	SM61
13	Вхід X5 по спадному фронту	SM38	46	Переривання за положенням 1 з	SM62

	переривання			високошвидкісний вихід	
14	Задній фронт входу X6 переривання	SM39	47	Переривання за положенням 2 з високошвидкісний вихід	SM105
15	Задній фронт входу X7 переривання	SM40	48	Переривання за позицією 3 з високошвидкісний вихід	SM106
16	СОМ переривання передачі 0 кадру	SM41	49	Переривання за позицією 4 з високошвидкісний вихід	SM107
17	СОМ 0 переривання прийому кадру interrupt	SM42	50	Переривання за позицією 5 з високошвидкісний вихід	SM108
18	СОМ переривання передачі 1 кадру	SM43	51	Переривання на основі положення 6 з високошвидкісний вихід	SM98
19	СОМ 1 переривання прийому кадру interrupt	SM44	52	Переривання за положенням 7 з високошвидкісний вихід	SM99
20	СОМ 2 типорозмір переривання передачі	SM45			
21	СОМ 2 кадр приймання переривання	SM46			
22	Переривання за таймером 0	SM47			
23	Переривання за таймером 1	SM48			
24	Переривання за таймером 2	SM49			
25	РТО (Y0) вихід переривання завершення	SM50			
26	РТО (Y1) вихід переривання завершення	SM51			
27	РТО (Y2) вихід переривання завершення	SM52			
28	РТО (Y3) вихід переривання завершення	SM53			
29	РТО (Y4) вихід переривання завершення	SM54			
30	РТО (Y5) вихід переривання завершення	SM55			
31	РТО (Y6) вихід переривання завершення	SM56			
32	РТО (Y7) вихід переривання завершення	SM57			

9.2 Обробка переривання події

1. Коли виникає певна подія переривання, якщо її було активовано, відповідний номер події додається до запису черги запитів на переривання, яка є чергою типу FIFO довжиною 8 записів.

2. Обробка запиту переривання системою:

- (1) Якщо система виявляє будь-який запит у черзі переривань, вона зупиняє нормальне виконання користувацької програми.
- (2) Система зчитує головний запис у черзі запитів переривань, який фіксує номер події переривання, що виникла першою. Викликається та виконується програма користувацького переривання, що відповідає номеру події.
- (3) Після виконання відповідної програми переривання (виконано інструкцію повернення з переривання), запит на переривання обробляється, відповідний головний запис черги запитів видаляється з черги, а всі наступні записи зміщуються на один крок вперед.

(4) Система знову перевіряє, чи порожня черга запитів на переривання. Якщо вона не порожня, система повторює вищезазначені процедури, доки черга не стане порожньою.

(5) Коли черга запитів на переривання порожня, система продовжує виконувати перервану основну програму.

3. Система може обробляти лише один запит на переривання за один раз. Коли система обробляє запит на переривання, нова подія переривання додається до черги запитів на переривання, а не обробляється негайно. Система обробляє її після того, як будуть опрацьовані всі запити, що стоять перед нею в черзі.

4. Коли в черзі запитів на переривання є 8 записів, система автоматично маскує нову подію переривання, щоб жоден новий запит не додавався до черги. Маска не скасовується, доки всі запити в черзі не будуть оброблені, а перервана основна програма не буде виконана.

□ Примітка

1. Переривання мають бути короткими, інакше можуть виникнути відхилення, включаючи маскування інших подій переривання (пропуск запитів на переривання), перевищення часу сканування системи та низьку ефективність виконання основної програми.

2. Заборонено викликати інші підпрограми в програмі переривання.

3. Якщо ви бажаєте оновити I/O негайно під час переривання, ви можете використати інструкцію REF. Вам слід зауважити, що час виконання інструкції REF залежить від кількості точок I/O, які потрібно оновити.

4. Подія переривання може генерувати запит на переривання лише тоді, коли відповідний прапорець події переривання активовано (кожен тип керування увімкненням/вимкненням подій переривання відповідає пов'язаним SM елементам. Для увімкнення певних типів подій переривання потрібно встановити відповідний SM елемент у ON), а глобальний прапорець дозволу переривань увімкнено (ON).

5. Коли генерується запит на переривання, для якого у користувацькій програмі немає відповідної програми обробки переривання, система реагує на запит на переривання, але жодних дій не виконує.

9.3 Використання переривання за таймером interrupt

● Опис переривання за таймером interrupt

Таймерне переривання — це подія переривання, що генерується системою періодично на основі встановлених інтервалів.

Програма переривання за таймером головним чином застосовується у ситуаціях, що потребують своєчасної та негайної обробки системою, як-от періодичне опитування AI-входів та періодичне оновлення аналогового виходу на основі певних форм сигналу.

Ви можете встановити інтервали (одиниця: мс) для таймерних переривань шляхом налаштування відповідних SD елементів. Система генерує подію переривання лише тоді, коли досягається встановлений часовий інтервал (рекомендований мін. інтервал: > 4ms).

Стан ON/OFF певних елементів SM може вмикати/вимикати відповідні переривання за

таймером. Серія VC PLCs надає вам три ресурси переривань за таймером, як зазначено

нижче.

Таблиця 9-1 Перелік ресурсів переривань за таймером

Переривання за таймером interrupt	Подія переривання номер	(SD)Інтервали часові переривання (SD)	Увімкнуті керування (SM)
0	22	SD47	SM47
1	23	SD48	SM48
2	24	SD49	SM49

□ Примітка

1. Налаштування елементів керування дозволом не може впливати на виконання переривань за таймером у черзі запитів на переривання.

2. Відлік часу для повторно активованого переривання починається з нуля.

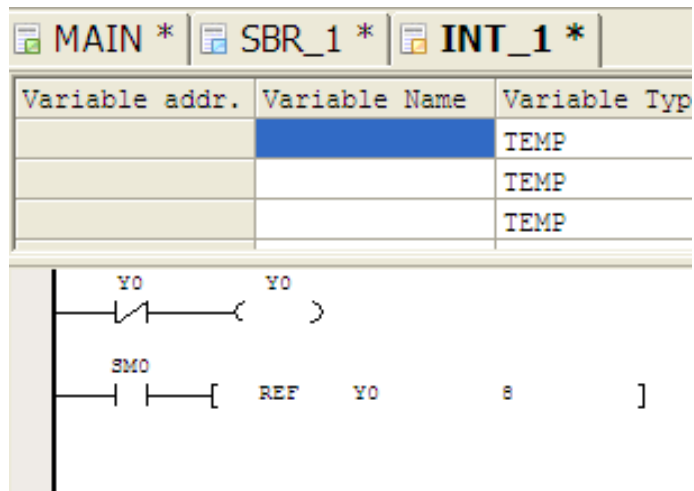
Щоб змінити інтервал таймерного переривання під час роботи програми, рекомендується виконати такі дії: 1.

Вимкнути таймерне переривання. 2. Змінити інтервал. 3. Увімкнути таймерне переривання.

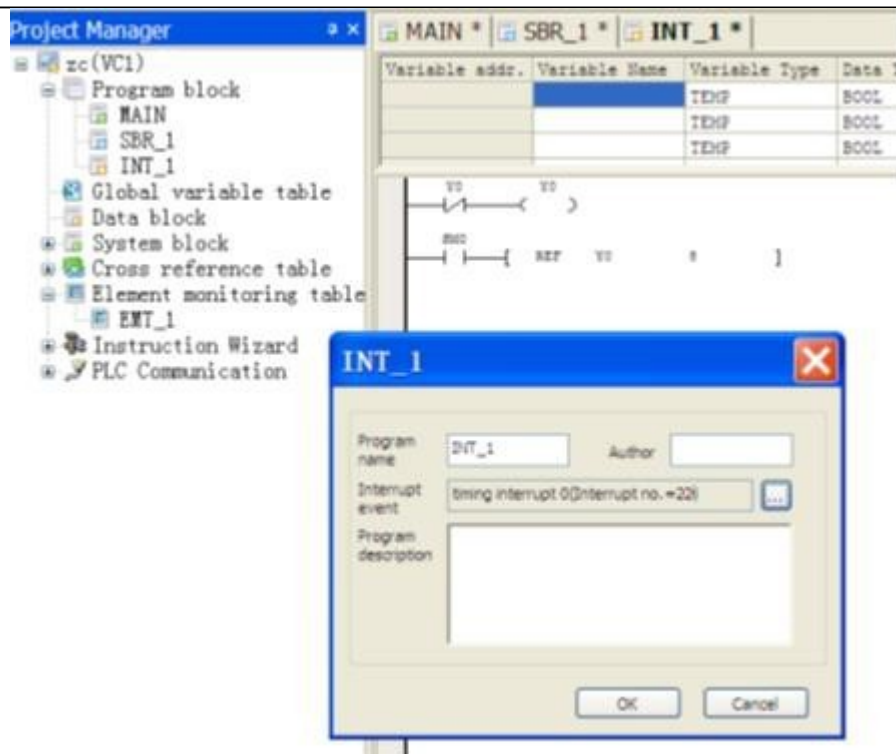
- Приклад застосування таймерного переривання

У прикладі застосування система змінює стан виходу Y0 один раз на секунду на основі переривання за таймером 0, що змушує Y0 періодично блимати.

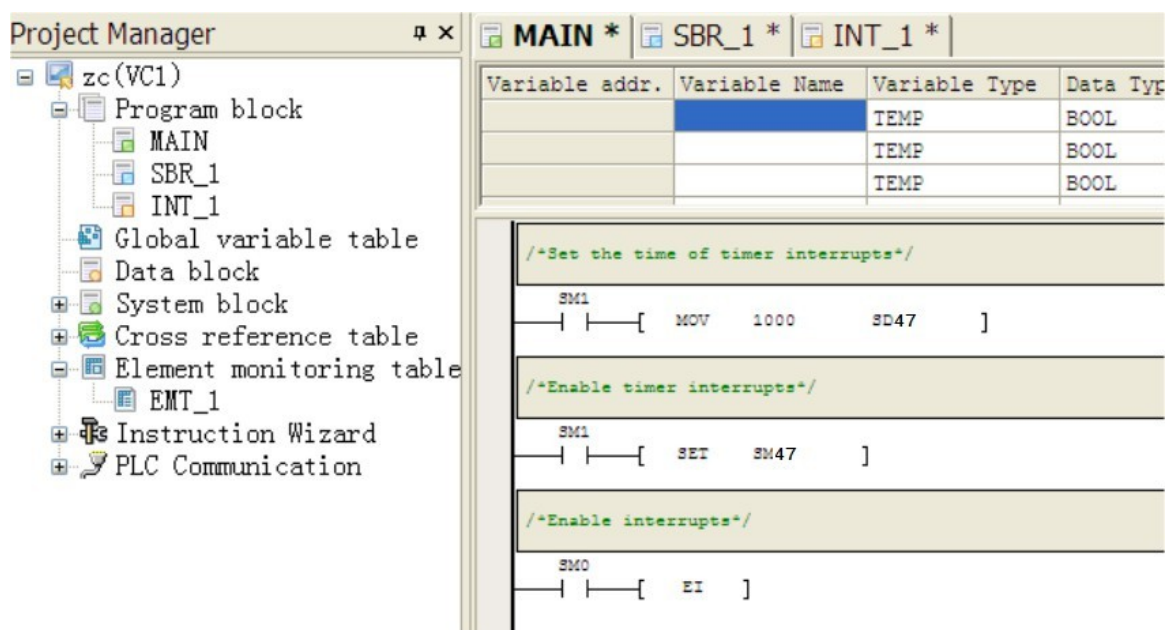
- Написання програми переривання для події переривання.



- Визначення номера події переривання для програми переривання:



- Встановлення інтервалу для таймерного переривання та активація таймерного переривання в основній програмі (MAIN).



9.4 Використання зовнішніх переривань

- Опис зовнішніх переривань

Зовнішні переривання пов'язані з фактичними вхідними точками PLCs, які класифікуються на переривання за переднім фронтом вхідного сигналу та переривання за заднім фронтом вхідного сигналу. У користувацькій програмі вони додають дії, пов'язані із зовнішніми подіями, до програми зовнішніх переривань.

Найвища частота реакції системи на зовнішню подію становить 1K. Зовнішні події з частотою понад 1K можуть бути втрачені.

Переривання за переднім фронтом та переривання за заднім фронтом не можуть використовуватися на одному порту одночасно. Усі зовнішні переривання є дійсними лише тоді, коли глобальне керування перериваннями EI та відповідне дозвільне SM є активними.

Детальний взаємозв'язок такий:

Переривання номер	Увімкнення елемента	Переривання номер	Увімкнення елемента
0 або 8	SM25/SM33	4 або 14	SM29/SM37
1 або 9	SM26/SM34	5 або 15	SM30/SM38
2 або 10	SM27/SM35	6 або 16	SM31/SM39
3 або 11	SM28/SM36	7 або 17	SM32/SM40

Зовнішні переривання пронумеровані наступним чином:

Переривання номер	Джерело переривання	Переривання номер	Джерело переривання
0	Передній фронт входу X0 переривання	9	Задній фронт входу X1 переривання
1	Передній фронт входу X1 переривання	10	Вхід X2 по спадному фронту переривання
2	Передній фронт входу X2 переривання	11	Задній фронт входу X3 переривання
3	Вхід X3 по передньому фронту переривання	12	Вхід X4 по спадаючому фронту переривання
4	Передній фронт входу X4 переривання	13	Вхід X5 по спадному фронту переривання
5	Передній фронт входу X5 переривання	14	Задній фронт входу X6 переривання
6	Передній фронт входу X6 переривання	15	Задній фронт входу X7 переривання
7	Передній фронт входу X7 переривання	16	Зарезервовано
8	Вхід X0 по спадному фронту	17	Зарезервовано

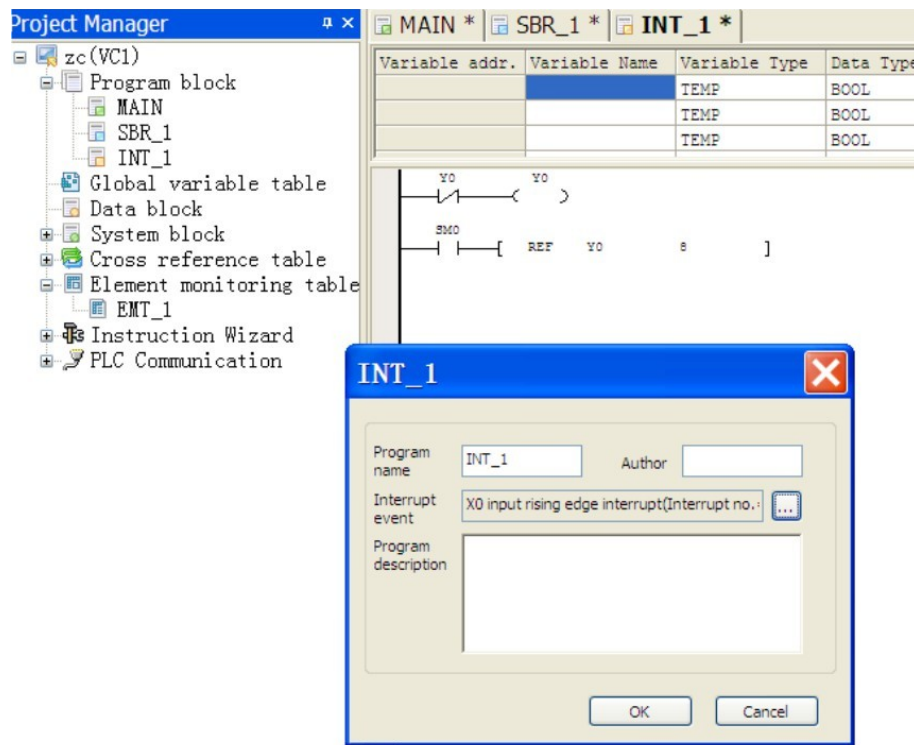
переривання

Частота одиничного вхідного імпульсу X0 - X7 менша за 200Hz.

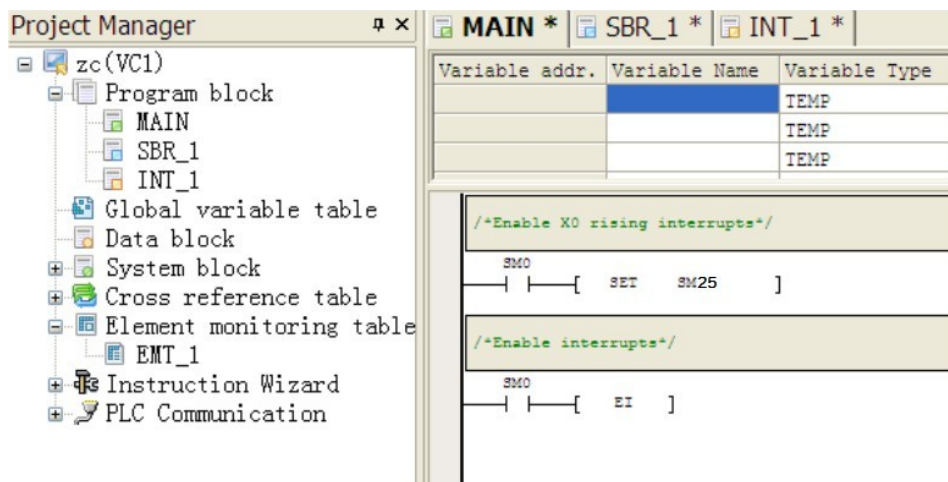
- Приклад застосування зовнішнього переривання

У прикладі застосування система змінює стан виходу Y0 на основі відповідної функції зовнішнього переривання 0 та події вхідного сигналу за переднім фронтом X0.

1. Написання програми переривання для зміни стану Y0 один раз під час кожного переривання та негайного виводу. Щоб скористатися перериванням, ви повинні вибрати відповідний номер переривання. Детальнішу інформацію можна побачити у конкретній операції на наступному рисунку.



2. Запис інструкції EI у MAIN та встановлення SM25, прапорця дозволу переривання, що відповідає перериванню за переднім фронтом входу X0, у стан дійсного.



9.5 Використання високошвидкісного лічильника переривання

- Опис високошвидкісного лічильника переривання

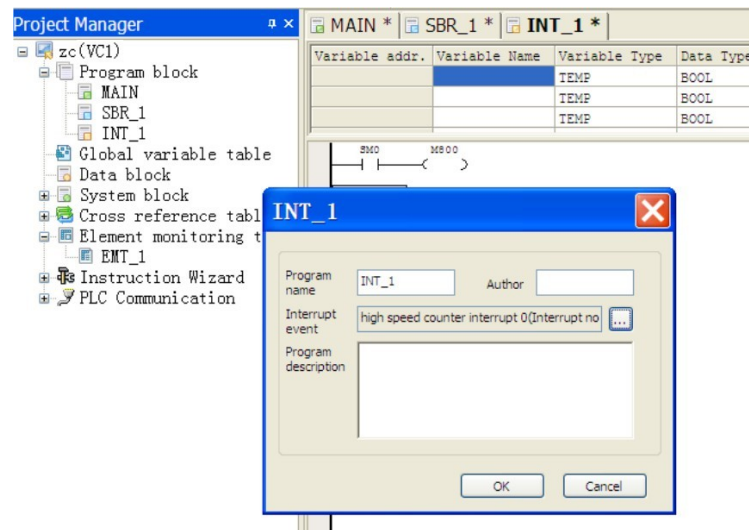
Переривання високошвидкісного лічильника є дійсним лише тоді, коли воно використовується разом з інструкцією NCNT або інструкцією DHSCI, і генерує переривання високошвидкісного лічильника на основі значення підрахунку

високошвидкісного лічильника. Ви можете компілювати програми, пов'язані із зовнішнім імпульсним входом, у програмі високошвидкісного переривання. Усі переривання високошвидкісного лічильника (33-40) є дійсними лише тоді, коли інструкція EI та відповідний прапор дозволу переривання є дійсними.

- Приклад застосування високошвидкісного лічильника переривання

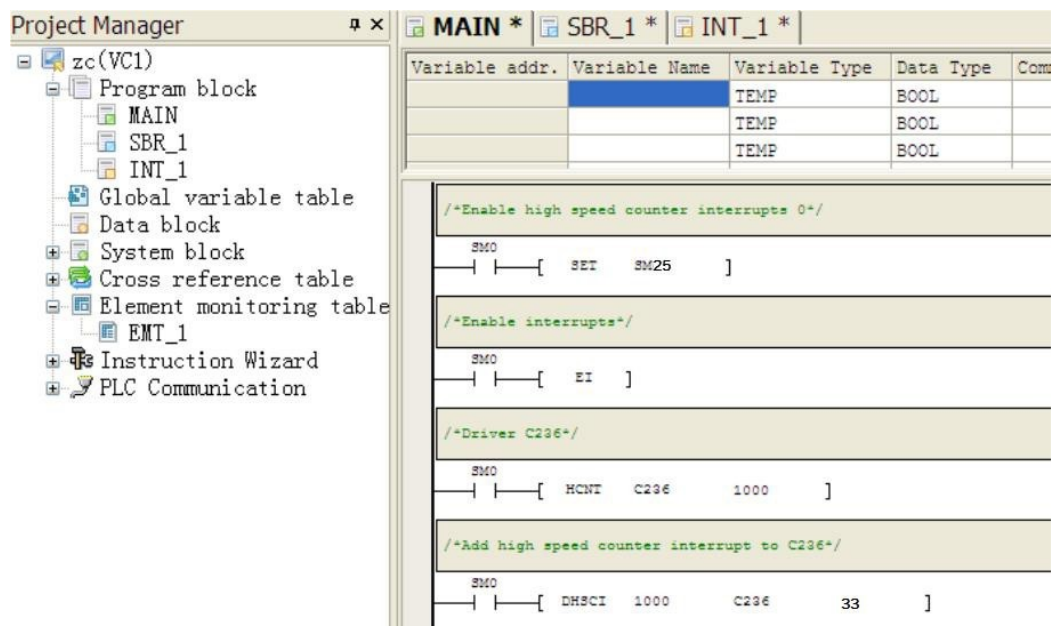
У прикладі застосування система використовує інструкцію переривання високошвидкісного лічильника X0 для реакції на програму переривання (номер 33), коли значення зовнішнього лічильника C236 досягає значення, вказаного через інструкцію DHSCI.

1. Написання підпрограми переривання. Щоб використовувати підпрограму переривання, потрібно вибрати відповідний номер переривання. Детальнішу інформацію про конкретну операцію можна знайти на наступному



рисунку.

2. Запис інструкції EI у MAIN та встановлення SM58, прапорця дозволу переривання, що відповідає перериванню високошвидкісного лічильника, як дійсного. Керування високошвидкісним лічильником C236 та інструкцією переривання високошвидкісного лічильника.



9.6 Використовуючи PTO виведіть завершення переривання

- Опис PTO виходу переривання завершення

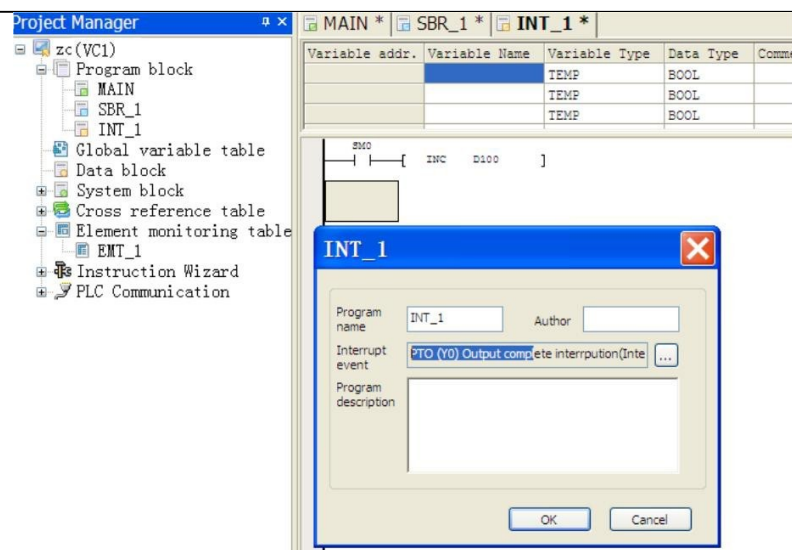
Переривання завершення виводу PTO спрацьовує, коли встановлено прапорець дозволу (SM50,SM51,SM52) і

високошвидкісний імпульсний вихід на Y0,Y1,Y2 завершено. Ви можете виконати відповідну обробку в підпрограмі переривання.

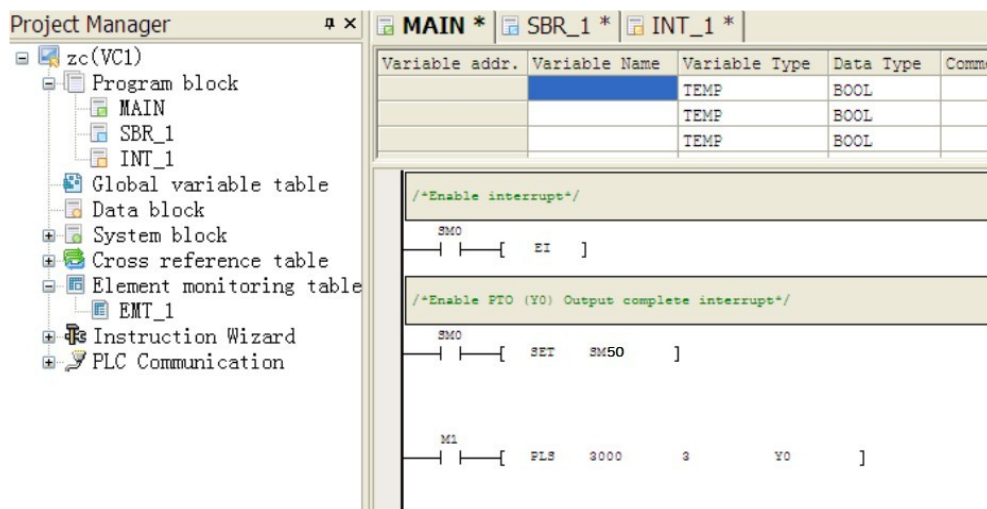
- Примірник застосунку PTO переривання виводу

У цьому прикладі застосування система використовує високошвидкісний імпульсний вихід Y0 для реагування на програму переривання (номер 25) після завершення високошвидкісного імпульсного виходу Y0.

1. Написання програми переривання (INT_1): Написання програми для коду переривання для реалізації керування. Щоб використовувати підпрограму переривання, потрібно вибрати відповідний номер переривання. Детальнішу інформацію можна знайти в описі конкретної операції на наступному рисунку.



2. Запис функцій у MAIN: Увімкнення глобального переривання системи та встановлення прапора дозволу переривання завершення виводу PTO SM50 як дійсного. Використання інструкції PLS.



9.7 Використання переривання на основі послідовного порту

- Опис переривання на основі послідовного порту

Коли послідовний порт перебуває в режимі протоколу вільного порту, система генерує подію переривання відповідно до подій надсилання або отримання послідовного порту.

Для кожного послідовного порту система надає вам два ресурси переривань. Програма переривань на основі послідовного порту головним чином застосовна до сценарію, що потребує спеціальної обробки прийому та передачі символу/кадру (XMT та RCV) через послідовний порт і своєчасної обробки системою. Вона може негайно реагувати на вимоги до обробки після того, як символ або кадр передано чи отримано, незалежно від часу

сканування.

Ви можете встановити стан ON/OFF відповідних елементів SM для увімкнення або вимкнення переривання на основі послідовного порту. Коли переривання на основі послідовного порту вимкнено, ті, що були додані до черги переривань, продовжують виконуватися. Інструкцію ХМТ не можна викликати в підпрограмі обробки переривання передачі символів, коли потік живлення нормально увімкнений. В іншому випадку це може призвести до вкладки підпрограм переривання і, таким чином, заблокувати виконання користувацької програми.

Переривання прийому та передачі кадру стосуються подій переривання, що ініціюються після виконання інструкцій ХМТ та RCV.

Список ресурсів переривань послідовного порту :

Переривання номер	Подія переривання event	Увімкнення переривання SM
16	Переривання передачі кадру PORT0	SM41

Переривання номер	Подія переривання event	Увімкнення переривання SM
17	Переривання прийому кадру PORT0	SM42
18	Переривання надсилання кадру PORT1	SM43
19	Переривання прийому кадру PORT1	SM44
20	Переривання передачі кадру PORT2	SM45
21	Переривання прийому кадру PORT2	SM46

- Приклад застосування переривання на основі послідовного порту

У прикладі, з функцією надсилання кадру через послідовний порт, система один раз збуджує вихід Y3 під час надсилання кадру, що створює ефект миготіння залежно від частоти надсилання символів кадру.

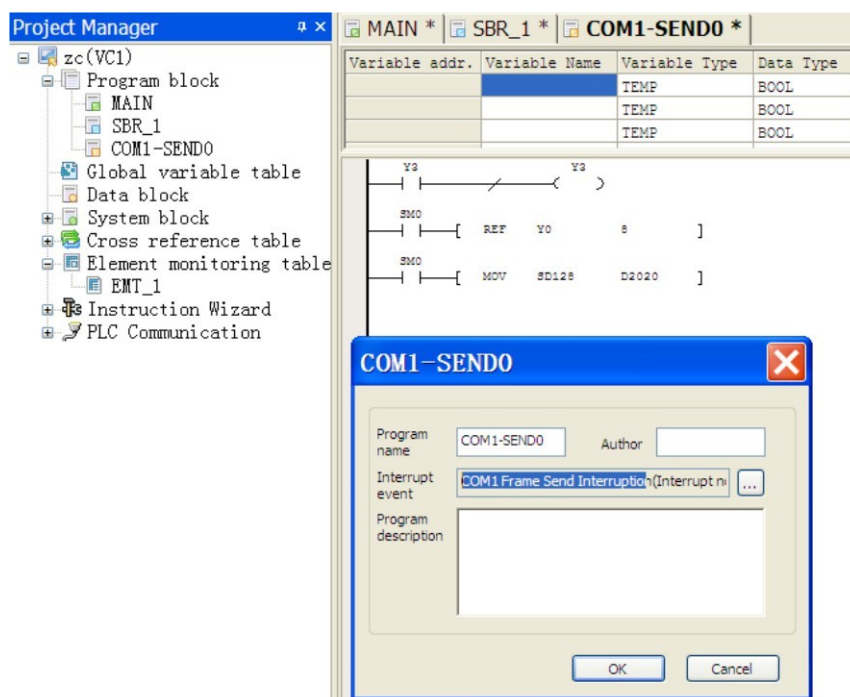
1. Написання програми переривання та коду обробки після завершення надсилання кадру послідовним портом та спрацювання переривання.

Variable addr.	Variable Name	Variable Type	Data
		TEMP	BOOL
		TEMP	BOOL
		TEMP	BOOL

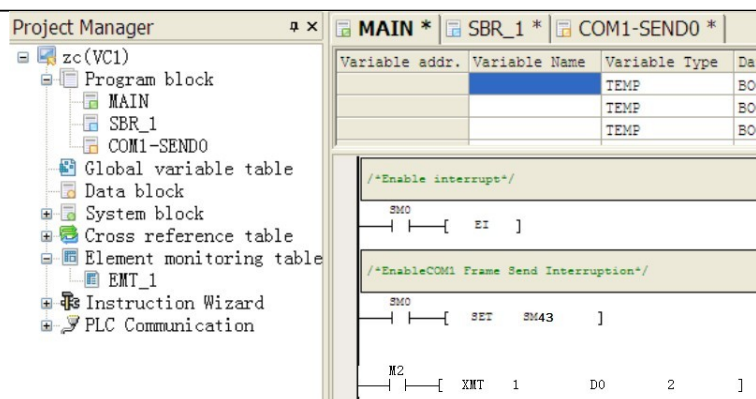
```

Network 1:
Y3 --- ( ) --- Y3
Network 2:
SM0 --- [ RST Y0 S ]
Network 3:
SM0 --- [ MOV SD128 D2020 ]
  
```

2. Визначення відповідного номера події переривання для програми переривання:



3. Написання коду для ввімкнення переривання передачі кадру послідовного порту в MAIN.



Детальну інформацію щодо прикладу застосування переривання послідовного порту див. у розділі 10 "Посібник з функцій зв'язку"

Розділ 10 Комунікаційна функція посібник

У цьому розділі детально представлено комунікаційні функції VC серії мікро-PLCs, включаючи комунікаційні ресурси та протоколи зв'язку, а також наведено приклади застосування для ілюстрації.

Розділ 10 Комунікаційна функція посібник.....	295
10.1 Комунікаційні ресурси.....	296
10.2 Протокол зв'язку порту програмування protocol.....	296
10.3 Протокол вільного порту зв'язку.....	296
10.3.1 Вступ.....	296
10.3.2 Налаштування параметрів вільного порту.....	296
10.3.3 Інструкція вільного порту.....	298
10.4 Протокол зв'язку Modbus protocol.....	299
10.4.1 Вступ.....	299
10.4.2 Характеристики посилення.....	299

10.4.3 RTU режим передачі.....	299
10.4.4 Підтримувані коди функцій Modbus.....	300
10.4.5 Режим адресації PLC елементів.....	300
10.4.6 Ведучий пристрій Modbus станція.....	301
10.4.7 Читання та запис елементів.....	301
10.4.8 Обробка подвійного слова елемента.....	302
10.4.9 Обробка LONG INT даних.....	302
10.4.10 Діагностичний код функції.....	302
10.4.11 Код помилки.....	303
10.4.12 Налаштування параметра Modbus налаштування.....	303
10.4.13 Інструкція Modbus інструкція.....	304
10.5 N:N протокол зв'язку протокол.....	306
10.5.1 N:N вступ.....	306
10.5.2 N:N форма передачі даних мережею form.....	306
10.5.3 N:N мережева структура.....	307
10.5.4 N:N режим оновлення mode.....	308
10.5.5 Покращений режим оновлення.....	312
10.5.6 N:N налаштування параметра параметра.....	313
10.6 Декілька стратегій керування.....	315
10.6.1 Визначення головної станції.....	315
10.6.2 Максимальна кількість інспекційних станцій.....	315
10.6.3 Мультимайстер та підпорядкований (M: N).....	315
10.6.4 Приклади використання N:N.....	315

10.1 Комунікаційні ресурси

Швидкості передачі даних, що застосовуються для малогабаритних приводів серії VC PLC, наведено в наступній таблиці:

Протокол зв'язку	Застосовна швидкість передачі даних
Протокол вільного порту, протокол Modbus протокол	115200, 57600, 38400, 19200, 9600, 4800, 2400, 1200
N:N протокол	115200, 57600, 38400, 19200, 9600, 4800, 2400, 1200

Комунікаційні протоколи, що підтримуються мікро-приводом серії VC PLCs, наведені в наступній таблиці.

Головний модуль	Зв'язок увімкнено порт	Тип порту	Підтримуваний протокол
VC1S (Планування)	PORT0	RS232	Протокол порту програмування, протокол вільного порту, протокол зв'язку Modbus протокол (підпорядкована станція), та N:N протокол зв'язку (головна та підпорядкована станція)
	PORT1	RS485	Протокол вільного порту, комунікаційний протокол Modbus (головна станція та підпорядкована станція), та N:N протокол зв'язку (головна та підпорядкована станція)
VC1	PORT0	RS232	Протокол порту програмування, протокол зв'язку Modbus (підпорядкована станція)
	PORT1	RS485	Протокол вільного порту, протокол зв'язку Modbus (головна станція та підпорядкована станція), та N:N протокол зв'язку (головна та підпорядкована станція)
	PORT2	RS485	Протокол вільного порту, комунікаційний протокол Modbus (головна станція та підпорядкована станція), та N:N протокол зв'язку (головна та підпорядкована станція)
	PORT3	USB	Порт програмування протокол
VC2 VC3 VC5 (Планування)	PORT0	RS232	Протокол порту програмування, протокол вільного порту, комунікаційний протокол Modbus протокол (підпорядкована станція), та N:N протокол зв'язку (головна та підпорядкована станція)
	PORT1	RS485	Протокол вільного порту, комунікаційний протокол Modbus (головна станція та підпорядкована станція), та N:N протокол зв'язку (головна та підпорядкована станція)
	PORT2	RS485	Протокол вільного порту, протокол зв'язку Modbus (головна станція та підпорядкована станція)

		станція), та N:N протокол зв'язку (головна та підпорядкована станція)
PORT3	USB	Порт програмування протокол
PORT4	Ethernet	Протокол порту програмування, Modbus_TCP (ведуча та ведена станція)
PORT5	CAN	CANopen(головна та підпорядкована станція)

Крім того, ви можете встановити перемикач вибору режиму серії VC micro-PLCs (RUN-STOP) для примусового переведення PORT0 у протокол порту програмування.

10.2 Протокол зв'язку порту програмування protocol

Протокол порту програмування — це внутрішній протокол, призначений для зв'язку між PC програмним забезпеченням Auto Studio та головним модулем. USB, мережевий порт та послідовний порт можуть використовуватися для зв'язку з головним модулем. Для моніторингу можна вибрати лише один із них, і їх не можна використовувати одночасно.

10.3 Протокол вільного порту зв'язку

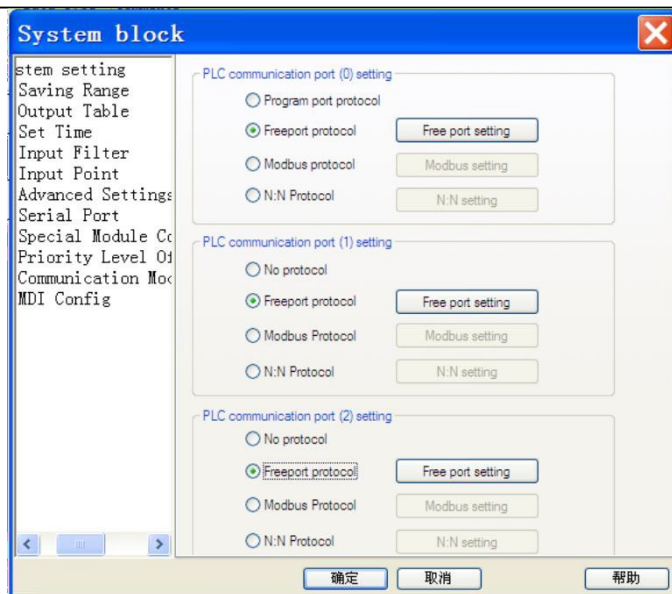
10.3.1 Вступ

Протокол вільного порту — це режим зв'язку з визначеним користувачем форматом файлу даних, який може реалізувати надсилання та отримання даних за допомогою інструкцій. Він підтримує два формати даних: ASCII та двійковий. Зв'язок через вільний порт доступний лише тоді, коли PLC перебуває у стані RUN.

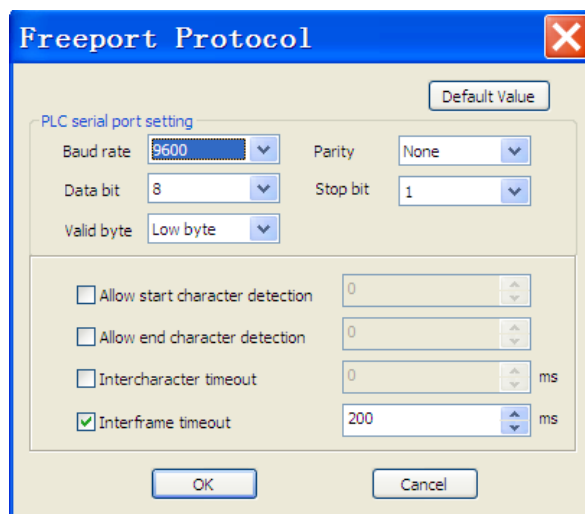
Інструкції для вільного порту зв'язку включають XMT (інструкція передавання через вільний порт) та RCV (інструкція приймання через вільний порт).

10.3.2 Налаштування параметрів вільного порту

Виберіть **Комунікаційний порт** у діалоговому вікні системного блоку та виберіть **Протокол вільного порту** у відповідній області налаштувань, щоб увімкнути кнопку налаштування вільного порту, як показано на наступному рисунку.



Натисніть будь-яку з кнопок налаштування вільного порту, щоб увійти в інтерфейс налаштування параметрів вільного порту, як показано на наступному рисунку.



Налаштовувані елементи перелічені у наступній таблиці.

Елемент	Налаштування вмісту	Примітка
Швидкість передачі даних	115200, 57600, 38400, 19200, 9600, 4800, 2400, 1200. Типово: 9600	—
Біт даних	7 або 8 (за замовчуванням)	—
Перевірка парності	Немає (за замовчуванням), непарний, парний	—
Зупинка біта	1 (за замовчуванням) або 2	—
Дозволити запуск виявлення символів	Увімкнено чи вимкнено (за замовчуванням)	—
Початковий символ виявлення	0–255 (відповідає 00–FF)	Починайте прийом після виявлення визначеного початкового символу та зберігайте отримані символи (включаючи початковий символ) у визначеному буфер
Дозволити кінець виявлення символів	Увімкнено чи вимкнено (за замовчуванням)	—
Кінцевий символ виявлення	0–255 (відповідає 00–FF)	Зупинка приймання після заданого кінцевого символу є отримано, і збереження кінцевого символу в буфер
Дозволити міжфреймовий тайм-аут	Увімкнено чи вимкнено (типово)	—
Тайм-аут між символами	0–65535 мс	Припинити приймання, якщо інтервал між двома отриманими символами довший за встановлений міжсимвольний тайм-аут

Міжкадровий тайм-аут увімкнути	Увімкнено чи вимкнено (за замовчуванням)	—
Міжкадровий тайм-аут	0–65535 мс	Коли живлення увімкнено і умови зв'язку виконані, тобто відлік часу починається, коли послідовний порт зв'язку починає отримувати кадри. Якщо прийом одного кадру завершено, коли час вичерпано, прийом є завершено.

10.3.3 Інструкція вільного порту

- Примітка

Інструкції вільного порту XMT та RCV можна використовувати для надсилання/отримання даних до/з призначеного комунікаційного порту. Детальну інформацію щодо використання інструкцій вільного порту див. у розділі 6.12.11

"XMT: Інструкція надсилання вільного порту" та розділі

6.12.12 "RCV: Приймання через вільний порт інструкції".

Вам слід зауважити, що для використання інструкції вільного порту на певному порту, необхідно встановити протокол вільного порту та параметри зв'язку для комунікаційного порту через системний блок Auto Studio. Після налаштування необхідно завантажити системні параметри у PLCs та перезапустити його.

- Екземпляр застосунку

Приклад 1: Надсилання 5-байтових даних, а потім отримання 6-байтових даних через PORT1.

Дані, що мають бути надіслані:	01	FF	00	01	02		Дані отримано	01	FF	02	03	05	FE

Збереження отриманих даних у D елементах, починаючи з D10. Кожен байт займає один D елемент, як показано нижче:

01	FF	02	03	05	FE
D10	D11	D12	D13	D14	D15

<pre> SM1 ├── [MOV 16#1 D0] ├── [MOV 16#FF D1] ├── [MOV 16#0 D2] ├── [MOV 16#1 D3] ├── [MOV 16#2 D4] ├── [RST SM122] ├── [XMT 1 D0 5] ├── SM122 │ ├── [RST SM123] │ └── [RCV 1 D10 6] ├── SM123 │ └── [BLD SD125 2] ├── X5 │ ├── [RST SM120] │ └── [RST SM121] </pre>	- Спочатку потрібно змінити налаштування комунікаційного порту в системному блоці на режим вільного порту та встановити відповідні параметри, такі як швидкість передачі даних та парність. - Збереження даних, що підлягають відправленню, у буфері зв'язку BFM, починаючи з D0, відправлення даних за допомогою інструкції XMT та скидання SM122 (біта прапорця завершення відправлення) перед відправленням, коли SM1 є дійсним. - Встановлення SM122 після надсилання даних та початок отримання даних за переднім фронтом. Макс. довжина для отриманих символів становить 6. - Налаштування SM123 після отримання даних та виконання відповідних операцій на основі регістра інформації про завершення отримання (SD125). - Використання X5 як біта дозволу для переривання надсилання та отримання.

Приклад 2: Надсилання даних через PORT1, а потім отримання даних.

<pre> SM1 ├── [MOV 16#1 D0] ├── [MOV 16#FF01 K4M0] ├── [MOV K2M0 D1] ├── [MOV K2M8 D2] ├── [MOV 16#1 D3] ├── [MOV 16#2 D4] ├── [RST SM122] ├── [XMT 1 D0 5] ├── SM122 │ ├── [RST SM123] │ └── [RCV 1 D10 6] ├── SM123 │ └── [BLD SD125 2] ├── X5 │ ├── [RST SM120] │ └── [RST SM121] </pre>	На відміну від "Приклад 1", перед надсиланням старшого та молодшого байтів елемента слова, цей елемент слова потрібно розділити на дві частини: старший та молодший байти. Наприклад, якщо ви хочете надіслати вміст D2, ви можете зберегти його старший та молодший байти окремо в D3 та D4, а потім надіслати D3 та D4. Ви також можете зберегти дані в елементі K4MX (як-от K4M0 у цій програмі), а потім взяти K2M0 як старший байт, а K2M8 як молодший байт.

10.4 Протокол зв'язку Modbus protocol

10.4.1 Вступ

Для послідовного зв'язку мікро-PLCs серії VC доступний протокол зв'язку Modbus. Підтримується режим RTU. PLCs можна налаштувати як головну або підпорядковану станцію.

10.4.2 Характеристики послань

1. Фізичний рівень: RS-232 та RS-485

2. Канальний рівень: асинхронний режим передачі

(1) Біт даних: 8 біт (RTU)

(2) Швидкість передачі: 1200, 2400, 4800, 9600, 19200, 38400, 57600 та 115200

(3) Метод парності: парна, непарна або відсутня

(4) Стоповий біт: 1 або 2 стопові біти

3. Мережева конфігурація: максимум 31 комплект обладнання. Діапазон адрес: 1–247. Широкомовлення підтримується.

10.4.3 RTU режим передачі

1. Шістнадцяткові дані.

2. Інтервал між двома символами повинен бути не меншим за час 1,5 символів.

3. Немає заголовка або кінця кадру, а інтервал між двома кадрами становить принаймні час 3,5 символів.

4. Використовуючи CRC16 перевірку.

5. Максимальна довжина RTU кадру становить 256 байтів, а структура кадру показана нижче:

Структура корпусу типорозміру	Адреса	Функція код код	Дані	CRC
Кількість від байти	1	1	0–252	2

6. Розрахунок інтервалу між символами:

Якщо швидкість передачі даних становить 19200, інтервал 1,5 символу дорівнює $1/19200 \times 11 \times 1,5 \times 1000$

$= 0.86\text{ms}$. Інтервал 3,5 символу дорівнює $1/19200 \times 11 \times 3,5 \times 1000 = 2\text{ms}$.

10.4.4 Підтримувані коди функцій Modbus codes

Підтримувані коди функцій Modbus включають 01, 02, 03, 05, 06, 08, 15 та 16.

10.4.5 Режим адресації PLC елементів

1. Відображення між функціональними кодами елементів читання-запису та елементами:

Функціональний код	Назва функції код	Адреса даних Modicon address	Тип керованого елемента	Примітка
01	Зчитати котушки	0 ^{Примітка 1} :xxxx	Y, X, M, SM, S, T та C	Біт зчитування
02	Зчитати дискретний вхід	1 ^{Примітка 2} :xxxx	X	Біт зчитування
03	Зчитати регістри	4 ^{Примітка 3} :xxxx ^{Примітка 4}	D, SD, Z, T, C та R	Слово зчитати
05	Запишіть одну котушку	0:xxxx	Y, M, SM, S, T та C	Біт запису
06	однорядковий зареєструвати	4:xxxx	D, SD, Z, T, C та R	Слово записати

15	Запишіть декілька котушок	0:xxxx	Y, M, SM, S, T та C	Біт запису
16	декілька регістри	4:xxxx	D, SD, Z, T, C та R	Слово записати

Примітки:

- 0 вказує на "котушку".
- 1 вказує на "дискретний вхід".
- 4 вказує на "регістр".
- xxxx вказує діапазон "1–9999". Кожен тип має незалежний діапазон логічних адрес від 1 до 9999 (адреса протоколу починається з 0).
- 0, 1 та 4 не мають фізичного значення і не задіяні у фактичній адресації.
- Користувачі не повинні записувати елемент X з кодами функцій 05 та 15, інакше система не надає зворотну інформацію про помилку, якщо записані операнди та дані є правильними, але система не виконує жодної операції за інструкцією запису.

2. Відображення між елементами PLC та адресою протоколу зв'язку Modbus :

Елемент немає	Тип	Фізичний елемент	Протокол адреса	Підтримується функція code	Примітка
Y	Біт	Y0–Y777 (вісімковий код) 512 точок у загальній кількості	0000–0511	01, 05, та 15	Стан виходу, номер елемента: Y0–Y7 та Y10–Y17
X	Біт	X0–X777 (вісімковий код) 512 точок загалом	1200–01711	01, 02, 05 та 15	Стан входу, він підтримує два типи адрес та номер елемента є таким самим, як вище
M	Біт	M0–M2047 M2048–M10239	2000–4047 12000–20191	01, 05 та 15	
SM	Біт	SM0–SM255 SM256–SM1023	4400–4655 30000–30767	01, 05, та 15	
S	Біт	S0–S1023 S1024–S4095	6000–7023 31000–34071	01, 05 та 15	
T	Біт	T0–T255 T256–T511	8000–8255 11000–11255	01, 05 та 15	Стан T елемента
C	Біт	C0–C255 C256–C511	9200–9455 10000–10511	01, 05 та 15	Стан C елемента
D	Слово	D0–D7999	0000–7999	03, 06 та 16	
SD	Слово	SD0–SD255 SD256–SD1023	8000–8255 12000–12767	03, 06 та 16	
Z	Слово	Z0–Z15	8500–8515	03, 06 та 16	
T	Слово	T0–T255 T256–T511	9000–9255 11000–11255	03, 06 та 16	Поточне значення T елемента
C	Слово	C0–C199	9500–9699	03, 06 та 16	Поточне значення елемента C (INT)
C	Подвійний слово	C200–C255	9700–9811	03 та 16	Поточне значення елемента C (DINT)

Елемент немає	Тип	Фізичний елемент	Протокол адреса	Підтримується функція code	Примітка
C	Подвійний слово	C256–C306	10000–10101	03 та 16	Поточне значення елемента C (DINT)
R	Слово	R0–R32767	13000–45767	03, 06 та 16	

Примітка:

Протокольна адреса — це адреса, що використовується під час передавання даних і відповідає логічній адресі даних Modicon. Протокольна адреса починається з 0, тоді як логічна адреса даних Modicon починається з 1, тобто протокольна адреса + 1 = логічна адреса даних Modicon. Наприклад, якщо протокольна адреса M0 дорівнює 2000, то відповідна їй логічна адреса даних Modicon — 0:2001. На практиці читування та запис M0 виконуються через протокольну адресу, наприклад, кадр для читування елемента M0 (надісланий головною станцією).

01 01 07 D0 00 01 FD 47

CRC перевірити код
Кількість зчитаних елементів
Початкова адреса. Десятькове значення 07D0 дорівнює 2000 Код функції
Станція №

10.4.6 Ведучий пристрій Modbus станція

Ведена станція Modbus відповідає головній станції відповідно до отриманого повідомлення локальної адресації, а не надсилає повідомлення активно. Ведена станція підтримує лише коди функцій Modbus 01, 02, 03, 05, 06, 08, 15 та 16, а інші коди є «неприпустимими кодами функцій» (за винятком ширококомовного кадру).

10.4.7 Читання та запис елементів

Усі функціональні коди, крім 08, використовуються для читання та запису елементів. У принципі, в одному кадрі можна зчитати максимум 2000 бітових елементів та 125 слів, а записати максимум 1968 бітових елементів та 120 слів. Однак фактичні адреси протоколу для різних типів елементів відрізняються та є розривними (наприклад, адреса протоколу Y377 — 255, тоді як адреса протоколу X0 — 1200). Тому під час читання або запису елемента можна зчитувати лише елементи одного типу, а максимальна кількість елементів, що зчитуються, залежить від кількості фактично визначених елементів. Наприклад, під час читання елемента Y (Y0 — Y377, всього 256 точок) діапазон адрес протоколу становить від 0 до 255, відповідна логічна адреса даних Modicon — від 1 до 256, а максимальна кількість елементів Y, які можна зчитати, дорівнює 256.

Приклади є такими:

1. Надсилання з головної станції: 01 01 00 00 01 00 3D 9A

01 — адреса; 01 — код функції; 00 00 — початкова адреса; 01 00 — кількість елементів для зчитування; 3D

9A — перевірка відповіді підпорядкованої станції: надання коректної відповіді

2. Надсилання з головної станції: 01 01 00 00 01 01 FC 5A

Головна станція зчитує 01 01 (257) елементів, починаючи з 0000, що виходить за межі визначеної кількості Y елементів. Відповідь підпорядкованої станції: 01 81 03 00 51

Відповідь підпорядкованої станції містить недопустиме значення даних, оскільки $257 > 256$, а 256 — це максимально дозволена кількість елементів. Y

3. Надсилання з головної станції: 01 01 00 64 00 A0 7D AD

00 64 (десятькове 100) — це початкова адреса для зчитування головною станцією, а 00 A0 (десятькове 160) — це кількість елементів. Відповідь підпорядкованої станції: 01 81 02 C1 91

Відповідь підпорядкованої станції містить неприпустиму адресу даних, оскільки існує лише 156 Y елементів, визначених для адреси протоколу 100, але $160 > 156$, тому 160 є неприпустимим.

4. Надсилання з головної станції: 01 04 00 02 00 0A D1 CD

Головна станція надсилає кадр із кодом функції 04

Відповідь підпорядкованої станції: 01 84 01 82 C0

Відповідь підпорядкованої станції містить неприпустимі функціональні коди.

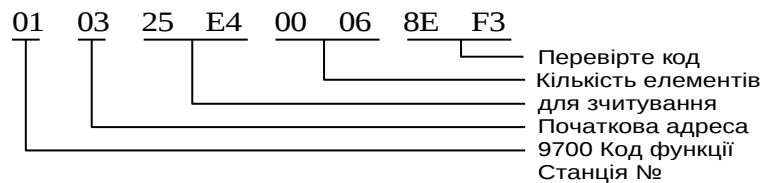
□ Примітка

1. Елемент X не підтримує операцію запису (тобто запис елемента X є недійсним). Щодо деталей про властивості запису SM та SDelements, зверніться до Додатка А "Спеціальне допоміжне реле" та Додатка В "Спеціальний регістр даних" (якщо елемент не підлягає запису, операція запису є недійсною).
2. Адреса підпорядкованої станції — 01, останні два байти є контрольними кодами CRC, а другий байт — функціональним кодом.

10.4.8 Обробка подвійного слова елемента

Поточне значення лічильника елемента C є елементом типу слово або подвійне слово. Значення від C200 до C255 є елементами типу подвійне слово, які зчитуються та записуються за допомогою функціональних кодів (03 та 16) регістрів читання/запису. Адреса кожних двох регістрів відповідає одному елементу типу подвійне слово C, і регістри можна зчитувати або записувати лише парами.

Наприклад, зчитування RTU кадру з трьох C подвійних слів від C200 до C202.



У повернутих даних дві адреси 9700 та 9701 вказують на вміст C200. 9700 — це MSB, а 9701 — це LSB.

Під час зчитування елемента подвійного слова, якщо початкова адреса для зчитування не є парним числом, система відповідає кодом помилки недопустимої адреси. Якщо кількість регістрів, що зчитуються, не є парним числом, система відповідає кодом помилки недопустимих даних.

Для прикладу:

Надсилання з головної станції: 01 03 25 E5 00 04 5E F2

Початкова адреса для зчитування головною станцією: 4 елементи слів 25 E5 (десятькове 9701)

Відповідь підпорядкованої станції: 01 83 02 C0 F1

Відповідь підпорядкованої станції: недопустима адреса

даних Відправлення з головної станції: 01 03 25 E4 00

05 CE F2

Початкова адреса для зчитування головною станцією: 5 елементів слів з 25 E5

Відповідь підпорядкованої станції: 01 83 03 01 31

Підпорядкована станція повертає неприпустимі дані.

10.4.9 Обробка LONG INT даних

Один LONG INT дані можна зберегти у двох D елементах. Наприклад, якщо LONG INT дані збережено у D3 та D4, VEICHI PLC вважає, що MSB зберігаються у D3, а LSB зберігаються у D4. Коли головна станція зчитує LONG INT дані через Modbus, 32-бітні дані мають бути перегруповані відповідно до принципу зберігання LONG INT у VEICHI PLC після зчитування даних. Принцип зберігання FLOAT такий самий, як і принцип зберігання LONG INT.

10.4.10 Діагностичний код функції

Код діагностичної функції використовується для тестування зв'язку між головною та підпорядкованою станціями, а також для перевірки різних внутрішніх умов помилок підпорядкованої станції. Підтримувані коди діагностичних підфункцій наведено в наступній таблиці:

Функція код	Підфункція код код	Назва підфункції code	Функція код	Підфункція код код	Назва підфункції code
08	00	Поверніть запит дані	08	12	Поверніть шинний зв'язок лічильник помилок
08	01	Перезапустіть опцію зв'язку	08	13	Поверніть шину винятковою лічильник помилок
08	04	Примусовий режим тільки прослуховування режим	08	14	Поверніть повідомлення підлеглого рахувати
08	10	Очистити лічильник	08	15	Поверніть номер підпорядкованої станції кількість count
08	11	Поверніть повідомлення шини лічильник	08	18	Поверніть шинний символ перевищення ліміту

Застосовується до VC2/VC3 серії

10.4.11 Код помилки

Коли інструкція, надіслана з головної станції, перебуває у стані нормальної відповіді, підпорядкована станція повертає дані або статистичне значення в полі даних. Але у стані аномальної відповіді сервер повертає коди помилок у полі даних. Коди помилок наведено в наступній таблиці.

Код помилки	Значення коду помилки
0x01	Код помилки code
0x02	Регістр помилок адреса
0x03	Помилка даних

Крім того, якщо підпорядкована станція отримує дані за наступних умов, повідомлення не повертається:

- (1) Коли виникають помилки у широкомовному кадрі, такі як помилка даних, помилка адреси, тощо
- (2) Коли відбувається переповнення символів, повідомлення не повертається. Наприклад, кадр RTU має розмір понад 256 байтів;
- (3) У режимі передачі RTU відбувається тайм-аут інтервалу між символами, що рівнозначно отриманню помилкового кадру, і жодне повідомлення не повертається;
- (4) Коли підпорядкована станція перебуває в режимі прослуховування, жодне повідомлення не повертається.
- (5) Підпорядкована станція отримала ASCII кадр помилки, включаючи помилку кінця кадру та помилку діапазону символів.

□ Примітка

Станцію зчитування налаштовано з обов'язковими елементами. Зчитане значення — це значення, що виконується програмою, яке може не збігатися з обов'язковим значенням.

10.4.12 Налаштування параметра Modbus налаштування

- Налаштування комунікаційного порту в системному блоці
На інтерфейсі комунікаційного порту є два послідовні порти (PORT0 та PORT1). PORT0 підтримує лише ведений пристрій Modbus, тоді як PORT1 та PORT 2 підтримують як головний, так і ведений пристрої Modbus.
- Налаштування параметрів протоколу зв'язку Modbus параметрів
На інтерфейсі операнда Modbus є кнопка значення за замовчуванням. Значення за замовчуванням — це налаштування зв'язку, рекомендоване протоколом зв'язку Modbus. Детальну інформацію про пункти налаштування параметрів можна знайти в наведеній нижче таблиці.

Елемент	Налаштування вмісту
Станція №	0–247
Швидкість передачі даних	115200, 57600, 38400, 19200, 9600, 4800, 2400, 1200
Біт даних	Встановлення на 7 або 8 біт: 7 біт для режиму ASCII, та 8 біт для режиму RTU режиму
Перевірка парності	Встановлення на none, odd та even
Зупинка біта	Встановлення на 1 або 2: 1 для перевірки на парність або непарність, та 2 для відсутності.
Modbus майстер/підпорядкован	Можна встановити як головну або підпорядковану станцію: PORT1 можна встановити як головну/підпорядковану станцію, і PORT0 можна

ий	може бути встановлено лише як підпорядкована станція
Режим передачі	Вибір RTU або ASCII режиму

Елемент	Налаштування вмісту
Час очікування для головний режим	Час очікування відповіді підпорядкованого пристрою головною станцією перевищує попередньо встановлений час
Примітка: Після встановлення операнда та його завантаження в системний блок, він стає дійсним лише після однієї операції.	

10.4.13 Інструкція Modbus інструкція

Коли PLC використовується як головна станція Modbus, кадр даних Modbus може бути надісланий/отриманий через інструкцію Modbus, надану системою. Детальнішу інформацію про використання інструкції Modbus див. у розділі 6.12.1 "Modbus: Інструкція зв'язку головної станції".

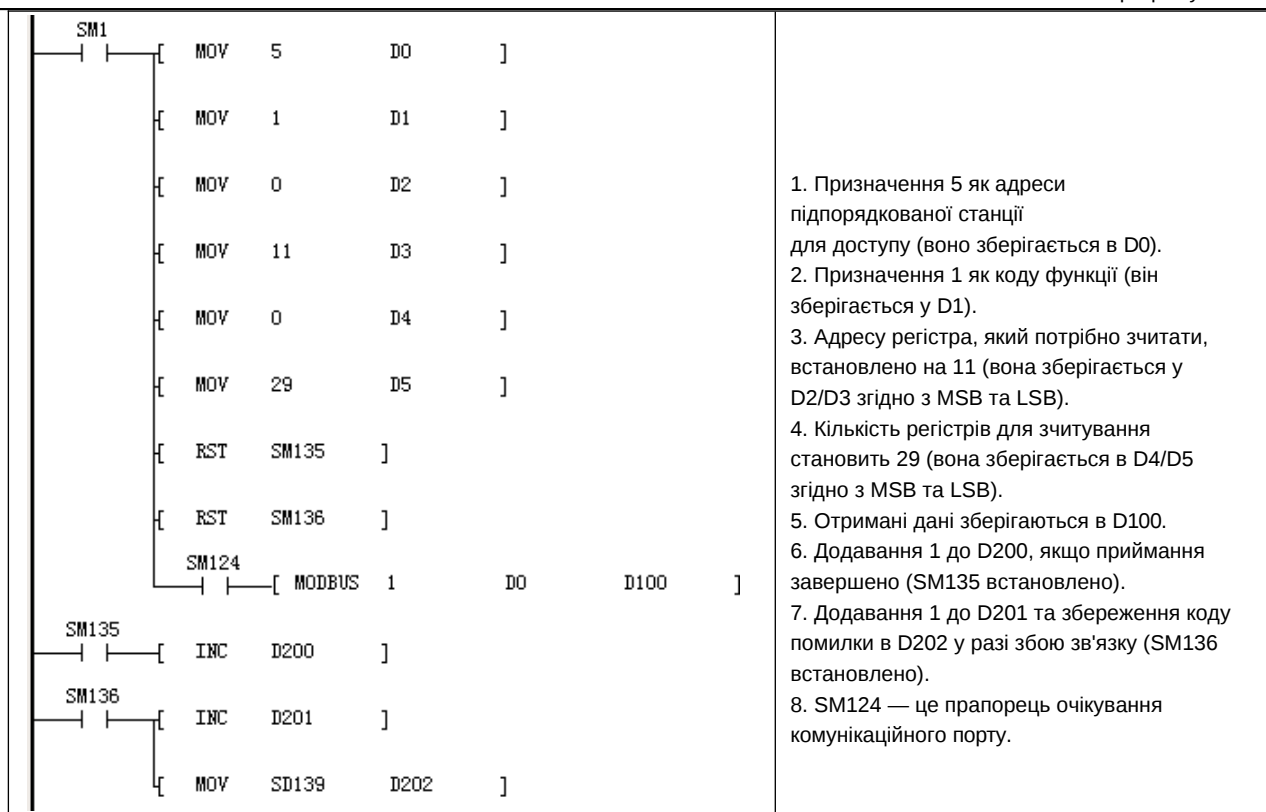
Якщо PLC налаштовано як головну станцію, у головному режимі під час встановлення параметра Modbus у системному блоці є пункт тайм-ауту. Щоб забезпечити правильність отриманих даних, час тайм-ауту має бути довшим за цикл сканування підпорядкованої станції Modbus і мати розумний запас. Наприклад, якщо VC1 серії PLC використовується як підпорядкована станція, а цикл сканування VC1 серії PLCs становить 300ms, тайм-аут головного режиму головної станції має перевищувати 300ms. Доцільно встановити тайм-аут на 350ms.

Прикладна програма

Приклад 1: Коли VC1 серії PLC є ведучим (master) та веденим (slave) пристроєм Modbus, він зчитує стан бітового елемента станції 5#. Протокольна адреса веденої станції, що зчитується ведучою станцією, — це бітові значення в діапазоні від 11 до 39. Припускаючи, що зчитані дані виглядають наступним чином, місце зберігання отриманих даних починається з D100, D100 зберігає адресу, D101 зберігає код функції, D102 зберігає кількість регістрів, а одиниці, починаючи з D103, зберігають зчитані значення бітових елементів.

42	41	40	39	38	37	36	35	34	33	32	31	30	29	28	27
X	X	X	0	0	1	1	0	0	0	0	1	0	1	1	0
D106								D105							
26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11
1	1	0	0	0	1	1	0	1	1	1	0	0	0	0	1
D104								D103							

Коли кількість зчитаних регістрів не є кратною 8, недостатні старші біти заповнюються нулями. У прикладі три старші біти (40, 41 та 42) регістру D106 заповнені нулями.



□ Примітка

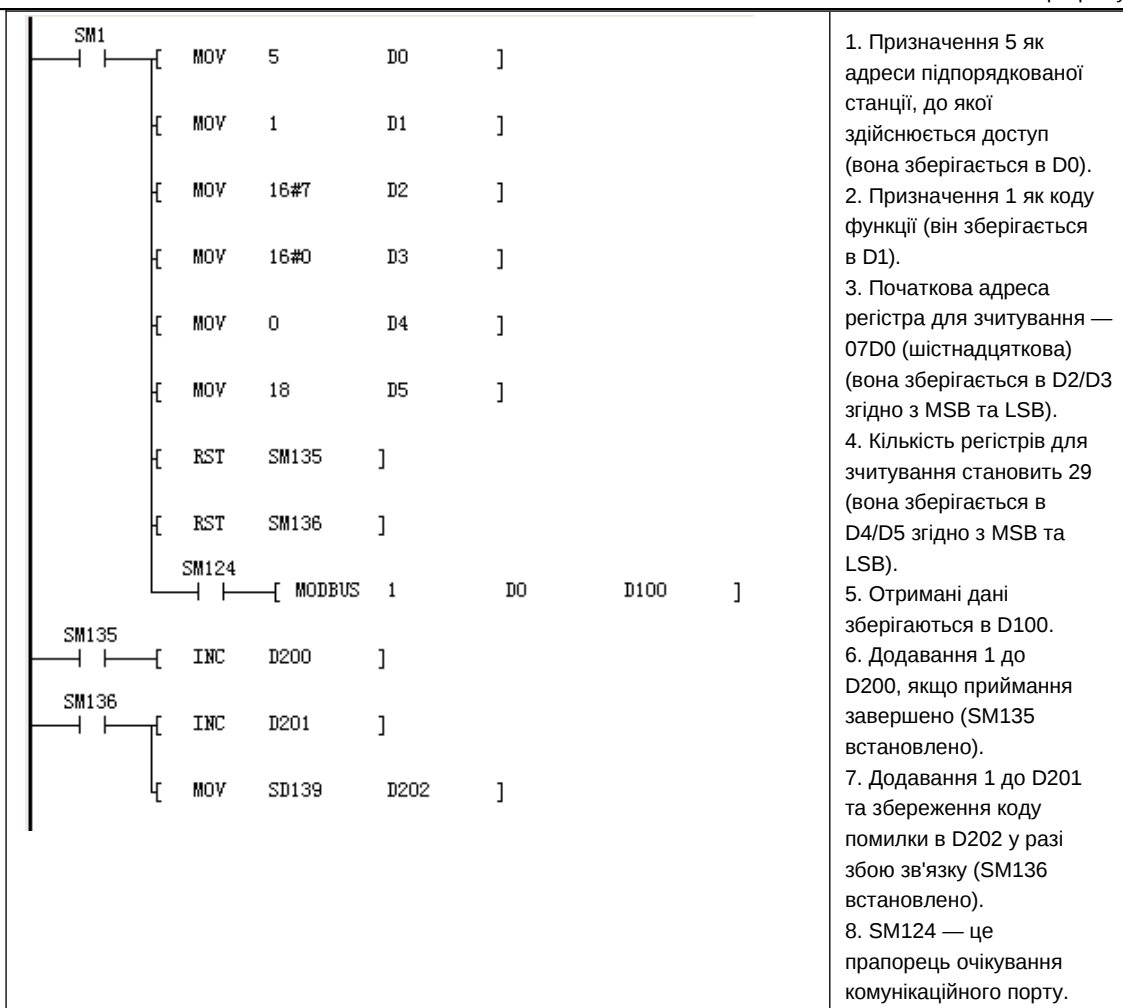
1. Коли логічна адреса використовується для адресації бітових елементів серії VC1 PLCs, логічна адреса 1 є адресою протоколу 0. У наведеному вище прикладі, щоб зчитати значення 11–39 бітових елементів (адреса протоколу) у підпорядкованій станції, логічна адреса має починатися з 12.

2. Збій цього зв'язку не впливає на наступний зв'язок, тобто, якщо в одній користувацькій програмі є дві інструкції Modbus XMT, перший зв'язок зазнає невдачі та має код помилки, це не впливає на передачу даних другої інструкції Modbus. Таким чином, у прикладі код помилки в SD139 розміщується в D202, тому код можна спостерігати через D202.

3. Для передавання повідомлення підпорядкованою станцією, якщо головна станція перебуває в режимі «тільки прослуховування», дані не повертаються, і система встановлює прапорець помилки. Тому, під час використання мережі Modbus серії VC1, якщо VC1 використовується як головна станція, користувач повинен чітко знати, яка підпорядкована станція PLC перебуває в режимі «тільки прослуховування», щоб гарантувати, що збій зв'язку не спричинений режимом «тільки прослуховування» підпорядкованої станції.

Приклад 2: Коли VC1 серії PLC є провідним та підпорядкованим пристроєм Modbus, він зчитує стани бітових елементів 2000–2017 протокольної адреси станції 5#.

Зчитані дані наведені нижче. Отриманий кадр починається з D100, D100 зберігає адресу, D101 зберігає код функції, D102 зберігає кількість регістрів, а елементи, починаючи з D103, зберігають зчитані значення бітових елементів.



Приклад 3: Коли VC1 серії PLC є головним пристроєм Modbus та підпорядкованою станцією, він зчитує значення елементів слів 40–43 протокольної адреси станції 5#.

MSB від елемент 40	LSB від елемент 40	MSB від елемент 41	LSB з елемент 41	MSB від елемент 42	LSB від елемент 42	MSB від елемент 43	LSB від елемент 43
D103	D104	D105	D106	D107	D108	D109	D110

	1. Призначення 5 як адреси підпорядкованої станції для доступу (воно зберігається в D0). 2. Призначення 3 як коду функції (він зберігається у D1). 3. Початкова адреса регістра для зчитування дорівнює 40 (вона зберігається в D2/D3 згідно з MSB та LSB). 4. Кількість регістрів для зчитування становить 4. 5. Отримані дані зберігаються в D100. 6. Додавання 1 до D200, якщо приймання завершено (SM135 встановлено). 7. Додавання 1 до D201 та збереження коду помилки у D202 у разі збою зв'язку (SM136 встановлено). 8. SM124 — це прапорець очікування комунікаційного порту.

10.5 N:N протокол зв'язку протокол

10.5.1 N:N вступ

N:N — це мікро-PLC мережа, розроблена Suzhou Veichi Electric Co.,Ltd. Фізичний рівень N:N використовує RS485, тому PLCs можна підключити безпосередньо через PORT1 або підключити через PORT0 за допомогою перетворювача RS-232/RS-485 до мережі N:N. PLCs, підключені до N:N, можуть автоматично обмінюватися значеннями між певними елементами D та M, що робить доступ до інших елементів PLC у мережі таким же простим і зручним, як доступ до власного елемента. У N:N доступ до даних між PLCs є повністю еквівалентним (комунікаційна мережа N:N).

Зручно налаштовувати N:N. Більшість параметрів N:N потрібно налаштовувати лише для №0 PLC. Крім того, N:N підтримує онлайн-модифікацію мережевих параметрів і здатна автоматично виявляти нові додані PLCs у мережі. Якщо будь-який PLC від'єднується від мережі, інші PLCs продовжують обмінюватися даними. Вона також здатна контролювати стан зв'язку всієї мережі через відповідні елементи SM будь-якого PLC у N:N.

10.5.2 N:N форма передачі даних мережею form

Існує два типи повідомлень у N:N токени, що видаються головною станцією, та широкомовні повідомлення від кожного PLC до власних даних.

Токен рівномірно видається головною станцією. Головна станція спочатку утримує токен. Після трансляції даних токен циклічно та послідовно видається кожній підпорядкованій станції. Тільки та підпорядкована станція, яка отримує токен, може здійснювати трансляцію іншим PLCs (включаючи головну станцію).

На рисунках 10-1–10-5 показано основні процеси мережевого зв'язку. На рисунку 1# позначено як головну станцію. Слід зауважити, що зазвичай 0# є головною станцією за замовчуванням, тоді як 1# є резервною головною станцією (у разі виникнення помилки зв'язку головної станції або збою живлення, відбувається перемикання на неї).

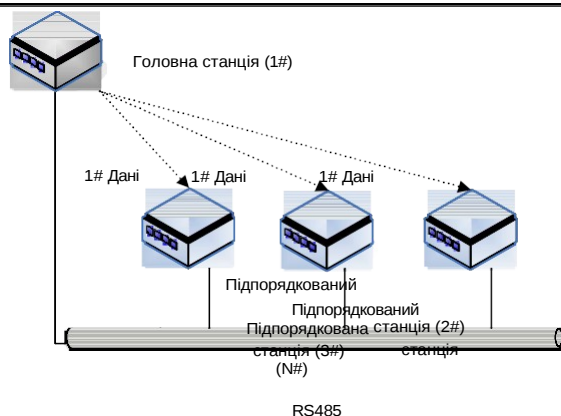


Рисунок 10-1 Головна станція здійснює широкомовну розсилку

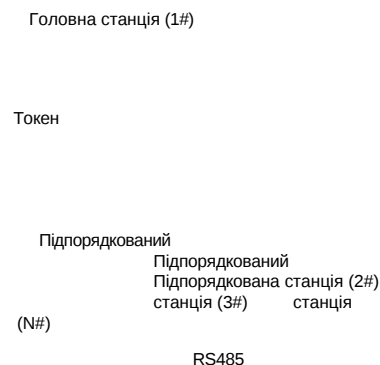


Рисунок 10-2 Головна станція передає маркер на підпорядковану станцію 2#

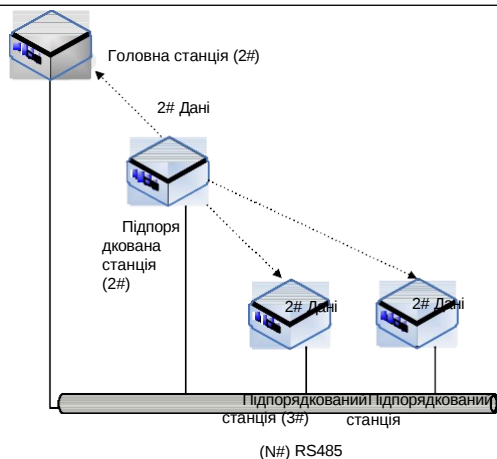


Рисунок 10-3 2 # підпорядкована станція здійснює широкомовну розсилку

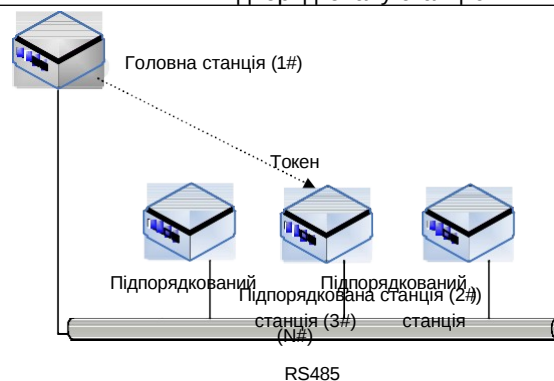


Рисунок 10-4 Головна станція передає маркер на підпорядковану станцію 3#

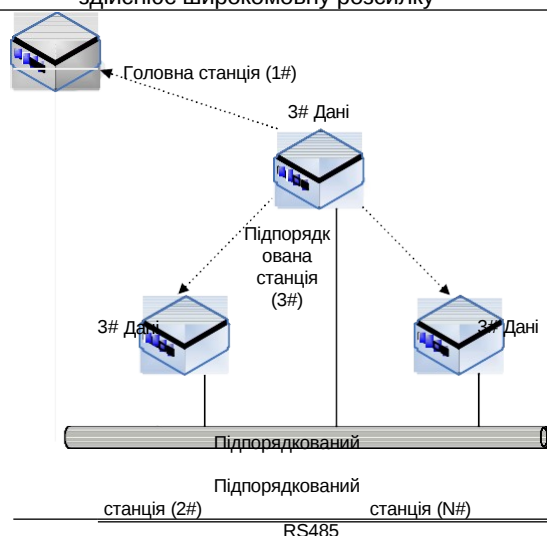


Рисунок 10-5 3 # підпорядкована станція здійснює широкомовну передачу

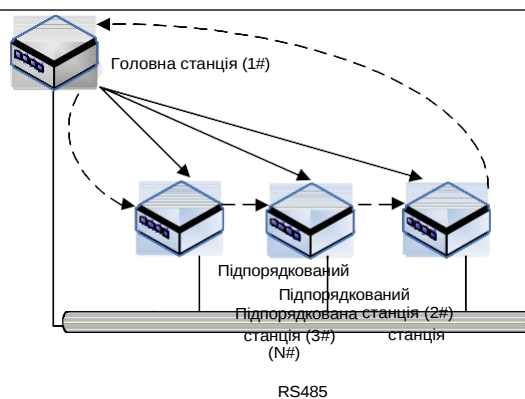


Рисунок 10-6 Проблема з токенами та потік заомовлень

На рисунку 10-6 показано порядок потоку маркерів. Жирна лінія вказує на фактичний процес видачі маркера, а пунктирна лінія вказує на порядок станції, у якій маркер утримується та транслюється. Вам слід зауважити, що маркер не передається однією підпорядкованою станцією (наприклад, 2#PLC) іншій підпорядкованій станції (наприклад, 3#PLC), а маркер спочатку видається головною станцією для 2#PLC, а потім видається головною станцією для 3#PLC.

10.5.3 N:N мережева структура

N:N підтримує два типи мереж: одношарову мережу та багатошарову мережу, як показано на наступних рисунках.

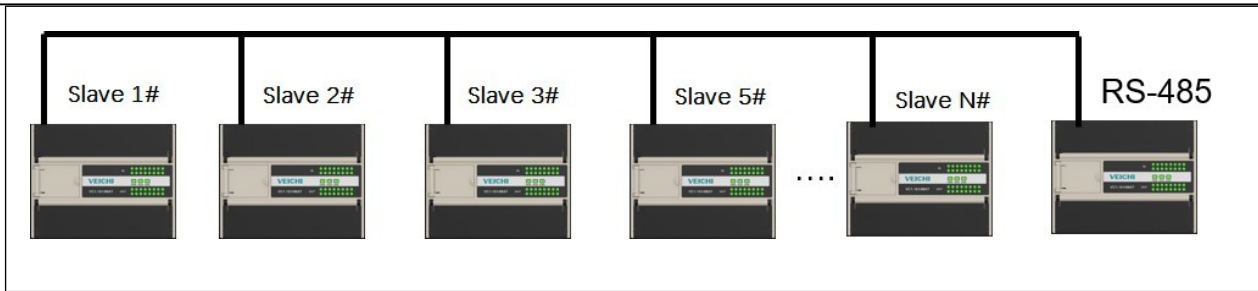


Рисунок 10-7 N:N одношарова мережа

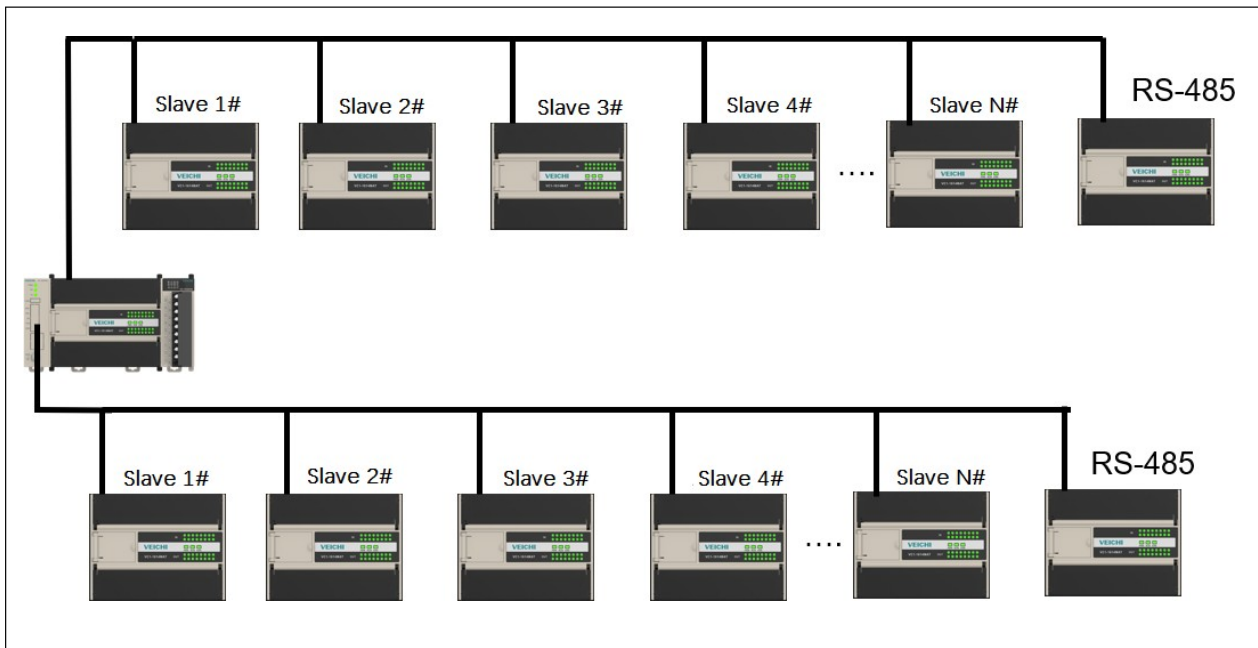
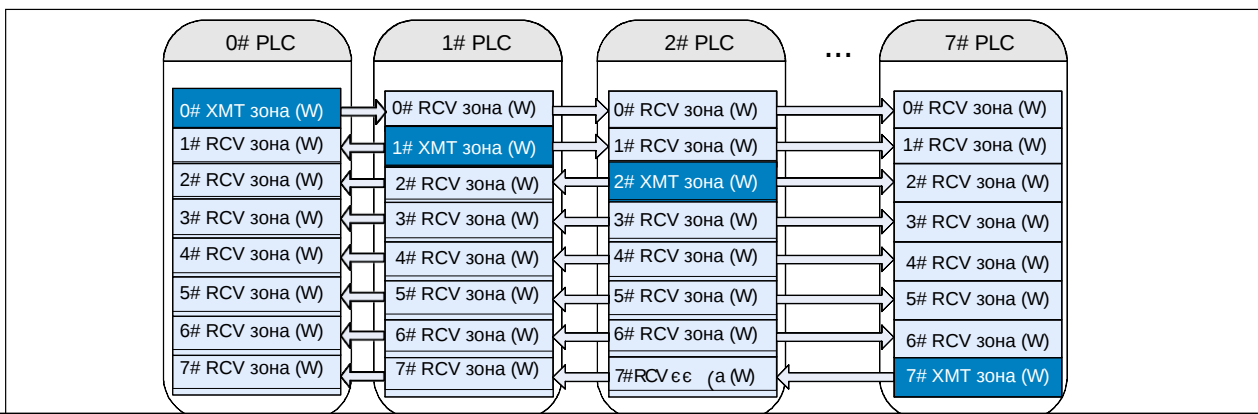


Рисунок 10-8 N:N багатошарова мережа

В одношаровій мережі кожен PLC підключений до N:N лише через один порт. У багатошаровій мережі має бути підключений PLC (проміжний вузол) між шарами, а два комунікаційні порти PLC відповідно підключені до різних шарів. Одношарова мережа може підтримувати максимум 32 PLCs, а кожен шар багатошарової мережі може підтримувати максимум 16 PLCs.

10.5.4 N:N режим оновлення mode

The PLCs connected to N:N can automatically exchange between parts of D elements and M elements in the network. The quantity and No. of these D and M elements are fixed, and these elements are called "Elements Sharing Area". If the PLC uses N:N, the value of the Elements Sharing Area keeps refreshing automatically, so as to keep the value consistency of the Elements Sharing Area for each PLC in the network.



Як показано на рисунку вище, кожен PLC, підключений до N:N, має область запису даних у зоні спільного використання елементів. N:N автоматично надсилає інформацію (значення визначених елементів D та M) з області запису даних до інших PLCs, отримує інформацію від інших PLCs та зберігає її в області читання даних.

Кількість елементів у зоні спільного використання елементів є фіксованою (загалом можна спільно використовувати 64 D компоненти та 512 M компонентів), і ці елементи розподіляються між більш ніж одним PLC. Тому, чим менше PLCs підключено до мережі, тим більше елементів можна розподілити на кожен PLC. Цей взаємозв'язок визначено таблицею режиму оновлення N:N.

- Розподіл елемента D на N:N одношарова мережа

Розподіл елементів D у надсилання зони	Режим 1	Режим 2	Режим 3	Режим 4	Режим 5
D7700–D7701	#0	#0	#0	#0	#0
D7702–D7703	#1				
D7704–D7705	#2				
D7706–D7707	#3	#1			
D7708–D7709	#4				
D7710–D7711	#5				
D7712–D7713	#6	#3	#1		
D7714–D7715	#7				
D7716–D7717	#8				
D7718–D7719	#9	#2		#1	
D7720–D7721	#10				
D7722–D7723	#11				
D7724–D7725	#12	#6	#3		
D7726–D7727	#13				
D7728–D7729	#14				
D7730–D7731	#15	#7		#2	
D7732–D7733	#16				
D7734–D7735	#17				
D7736–D7737	#18	#9	#5		
D7738–D7739	#19				
D7740–D7741	#20				
D7742–D7743	#21	#10		#3	
D7744–D7745	#22				
D7746–D7747	#23				
D7748–D7749	#24	#12	#6		
D7750–D7751	#25				
D7752–D7753	#26				
D7754–D7755	#27	#13		#7	
D7756–D7757	#28				
D7758–D7759	#29				
D7760–D7761	#30	#14	#3		
D7762–D7763	#31				
		#15			#7

Для прикладу:

(1) У режимі 1 елементи D, розподілені до зони передавання, призначеної станцією 0#, є D7700–D7701. Станція 0# PLC може записувати значення в D7700 та D7701, а інші станції (1#–31#) можуть безпосередньо зчитувати значення D7700 та D7701.

(2) У режимі 2 елементи D, розподілені до зони передавання, призначеної станцією 0#, є D7700–D7703. Станція 0# PLC може записувати значення в D7700, D7701, D7702, D7703, а інші станції (1#–15#) можуть безпосередньо зчитувати значення D7700–D7704.

- Розподіл елемента M на N:N одношарова мережа

Розподіл елементів M у надсилання зони	Режим 1	Режим 2	Режим 3	Режим 4	Режим 5
M1400–M1415	#0	#0			
M1416–M1431	#1				

M1432–M1447	#2	#1	#0	#0	#0
Розподіл М елементи в зоні відправленн я	Режим 1	Режим 2	Режим 3	Режим 4	Режим 5
M1448–M1463	#3				
M1464–M1479	#4				
M1480–M1495	#5	#2	#1		
M1496–M1511	#6	#3			
M1512–M1527	#7				
M1528–M1543	#8	#4	#2		
M1544–M1559	#9	#5			
M1560–M1575	#10			#1	
M1576–M1591	#11	#6			
M1592–M1607	#12		#3		#0
M1608–M1623	#13	#7			
M1624–M1639	#14				
M1640–M1655	#15	#8	#4		
M1656–M1671	#16	#9		#2	
M1672–M1687	#17	#10			
M1688–M1703	#18	#11	#5		
M1704–M1719	#19				
M1720–M1735	#20	#12	#6		
M1736–M1751	#21	#13			
M1752–M1767	#22	#14	#7		
M1768–M1783	#23				
M1784–M1799	#24	#15			
M1800–M1815	#25				
M1816–M1831	#26				
M1832–M1847	#27				
M1848–M1863	#28				
M1864–M1879	#29				
M1880–M1895	#30				
M1896–M1911	#31				

Для прикладу:

(1) У режимі 1 елементи М, розподілені до зони передачі, призначеної станцією 0#, — це M1400–M1415. Станція 0# PLC може записувати значення в M1400–M1415, а інші станції (1#–31#) можуть безпосередньо зчитувати значення M1400–M1415.

(2) У режимі 2 елементи М, розподілені до зони передавання, призначеної станцією 0#, — це M1400–M1431. Станція 0# PLC може записувати значення в M1400–M1431, а інші станції (1#–31#) можуть безпосередньо зчитувати значення M1400–M1431.

● Розподіл елемента D на N:N багатодіагональній мережі (шар 0)

Розподіл D елементи в зоні передавання	Режим 6	Режим 7	Режим 8	Режим 9
D7700–D7701	#0	#0	#0	
D7702–D7703	#1			
D7704–D7705	#2	#1		
D7706–D7707	#3			
D7708–D7709	#4	#2		
D7710–D7711	#5	#3	#1	
D7712–D7713	#6			
D7714–D7715	#7	#4		
D7716–D7717	#8			
D7718–D7719	#9	#5	#2	
D7720–D7721	#10			
D7722–D7723	#11			

D7724–D7725	#12	#6	#3	#1
D7726–D7727	#13			
D7728–D7729	#14	#7		
D7730–D7731	#15			

Наприклад:

У режимі 6 елементи D, розподілені до зони передавання, призначеної станції 0#, є D7700–D7701. Станція 0# PLC може записувати значення в D7700–D7701, а інші станції (1#–15#) можуть безпосередньо зчитувати значення D7700–D7701.

- Розподіл елементів D на N: багат шарова мережа N (шар 1)

Розподіл елементів D під час надсилання зона	Режим 10	Режим 11	Режим 12	Режим 13
D7732–D7733	#0	#0	#0	#0
D7734–D7735	#1			
D7736–D7737	#2	#1		
D7738–D7739	#3			
D7740–D7741	#4	#2	#1	
D7742–D7743	#5			
D7744–D7745	#6	#3		
D7746–D7747	#7			
D7748–D7749	#8	#4	#2	#1
D7750–D7751	#9			
D7752–D7753	#10	#5		
D7754–D7755	#11			
D7756–D7757	#12	#6	#3	
D7758–D7759	#13			
D7760–D7761	#14	#7		
D7762–D7763	#15			

Наприклад:

У режимі 10 елементи D, розподілені до зони передавання, призначеної станцією 0# (рівень 1), — це D7732–D7733. Станція 0# PLC може записувати значення в D7732–D7733, а інші станції (1#–15#) можуть безпосередньо зчитувати значення D7732–D7733.

- Розподіл елементів M на N: багат шарова мережа N (шар 0)

Розподіл М елементів при надсиланні зона	Режим 6	Режим 7	Режим 8	Режим 9
M1400–M1415	#0	#0	#0	#0
M1416–M1431	#1			
M1432–M1447	#2	#1		
M1448–M1463	#3			
M1464–M1479	#4	#2	#1	
M1480–M1495	#5			
M1496–M1511	#6	#3		
M1512–M1527	#7			
M1528–M1543	#8	#4	#2	#1
M1544–M1559	#9			
M1560–M1575	#10	#5		
M1576–M1591	#11			
M1592–M1607	#12	#6	#3	
M1608–M1623	#13			
M1624–M1639	#14	#7		
M1640–M1655	#15			

Наприклад:

У режимі 6 елементи M, розподілені до зони передавання, призначеної станцією 0# (рівень 0), — це M1400–M1415. Станція 0# PLC може записувати значення в M1400–M1415, а інші станції (1#–15#) можуть безпосередньо зчитувати значення M1400–M1415.

- Розподіл елементів M на N: багат шарова мережа N (шар 1)

Розподіл М елементи в зоні передавання	Режим 10	Режим 11	Режим 12	Режим 13
M1656–M1671	#0	#0	#0	#0
M1672–M1687	#1			
M1688–M1703	#2	#1		
M1704–M1719	#3			
M1720–M1735	#4	#2	#1	
M1736–M1751	#5			
M1752–M1767	#6	#3		
M1768–M1783	#7			
M1784–M1799	#8	#4	#2	#1
M1800–M1815	#9			
M1816–M1831	#10	#5		
M1832–M1847	#11			
M1848–M1863	#12	#6	#3	
M1864–M1879	#13			
M1880–M1895	#14	#7		
M1896–M1911	#15			

Наприклад:

У режимі 10 елементи M, розподілені до зони передачі, призначеної станцією 0# (рівень 1), — це M1656–M1671. Станція 0# PLC може записувати значення в M1656–M1671, а інші станції (1#–15#) можуть безпосередньо зчитувати значення M1656–M1671.

□ Примітка

Після того, як PLC налаштовано з протоколом зв'язку N:N, D7700–D7763 та M1400–M1911 використовуються як загальні ресурси для обміну мережевими даними. Вам необхідно звернути увагу на ці елементи під час їх використання у програмі!

10.5.5 Покращений режим оновлення

Для підтримки спільного використання більшої кількості елементів, мікро-PLCs серії VC пропонують режими 14–18. Ці режими застосовні лише до одношарової структури або ситуацій, де є багато спільних компонентів. Компоненти M та компоненти D були розширені на початковій основі (M1400–M1911 та D7500–D7755).

Області компонентів M (512) наведено у наступній таблиці:

Розподіл M елементів	Режим 14	Режим 15	Режим 16	Режим 17	Режим 18
M1400-M1415	#0	#0	#0	#0	#0
M1416-M1431	#1				
M1432-M1447	#2				
M1448-M1463	#3	#1			
M1464-M1479	#4				
M1480-M1495	#5				
M1496-M1511	#6	#3			
M1512-M1527	#7				
M1528-M1543	#8	#4	#2	#1	
M1544-M1559	#9				
M1560-M1575	#10	#5			
M1576-M1591	#11				
M1592-M1607	#12				
M1608-M1623	#13				
M1624-M1639	#14	#7			
M1640-M1655	#15				
M1656-M1671	#16		#8		
M1672-M1687	#17				
M1688-M1703	#18				

M1704-M1719	#19	#9		#2	#1
M1720-M1735	#20	#10	#5		
M1736-M1751	#21				
M1752-M1767	#22	#11			
M1768-M1783	#23				

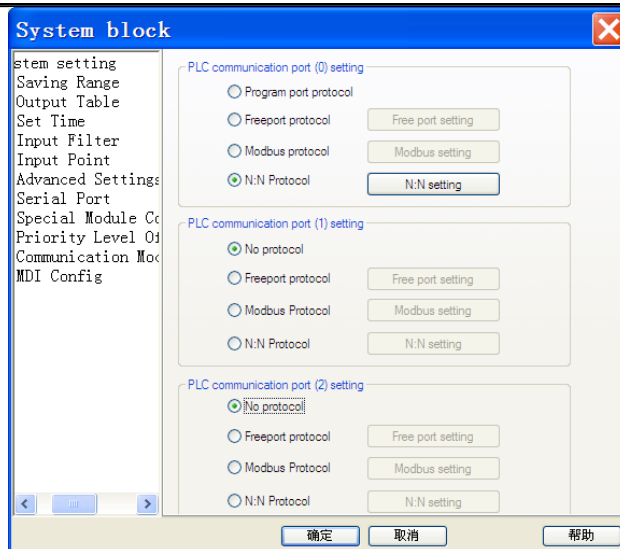
M1784-M1799	#24	#12	#6	#3	
M1800-M1815	#25				
M1816-M1831	#26	#13			
M1832-M1847	#27				
M1848-M1863	#28	#14	#7		
M1864-M1879	#29				
M1880-M1895	#30	#15			
M1896-M1911	#31				

Області елементів D (256) показані в наступній таблиці:

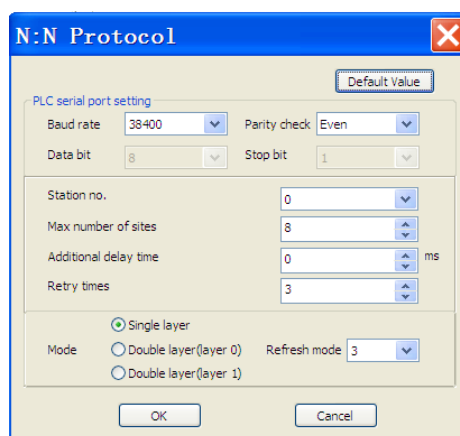
Розподіл D елементів	Режим 14	Режим 15	Режим 16	Режим 17	Режим 18	
D7500–D7507	#0	#0	#0	#0	#0	
D7508–D7515	#1					
D7516–D7523	#2	#1				
D7524–D7531	#3					
D7532–D7539	#4	#2	#1			
D7540–D7547	#5					
D7548–D7555	#6	#3				
D7556–D7563	#7					
D7564–D7571	#8	#4	#1	#1		
D7572–D7579	#9					
D7580–D7587	#10	#5				
D7588–D7595	#11					
D7596–D7603	#12	#6				#3
D7604–D7611	#13					
D7612–D7619	#14	#7				
D7620–D7627	#15					
D7628–D7635	#16	#8	#4		#2	#1
D7636–D7643	#17					
D7644–D7651	#18	#9				
D7652–D7659	#19					
D7660–D7667	#20	#10	#5			
D7668–D7675	#21					
D7676–D7683	#22	#11				
D7684–D7691	#23					
D7692–D7699	#24	#12	#6	#3	#1	
D7700–D7707	#25					
D7708–D7715	#26	#13				
D7716–D7723	#27					
D7724–D7731	#28	#14	#7			
D7732–D7739	#29					
D7740–D7747	#30	#15				
D7748–D7755	#31					

10.5.6 N:N налаштування параметра параметра

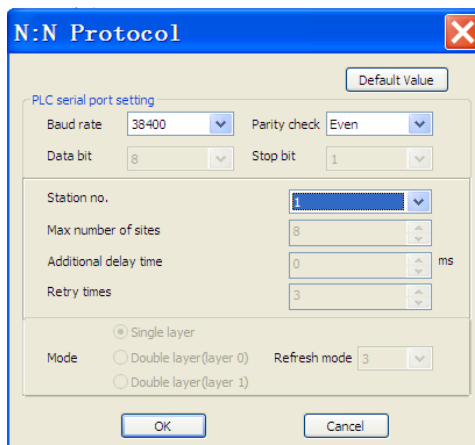
Виберіть пункт **Комунікаційний порт** у блоці Система та виберіть протокол **N:N** у налаштуванні **PLC комунікаційного порту 1**, щоб активувати відповідну кнопку налаштування **N:N**, як показано нижче.



Натисніть кнопку **N:N налаштування**, щоб увійти в інтерфейс налаштування **N:N протоколу**, як показано нижче.



Як показано на попередньому рисунку, параметри N:N встановлюються через системний блок. **Номер станції** має встановлюватися, починаючи з 0#, і задаватися по порядку. Декілька PLCs не можуть мати однаковий номер станції. Станція 0# використовується для запуску та налаштування всієї мережі. Налаштування **Максимальної кількості станцій**, **Додаткового часу затримки**, **Кількості спроб**, **Режиму** можна виконати через станцію 0#. Для станцій з іншими номерами, окрім того, що швидкість передачі даних та перевірка парності мають збігатися з параметрами станції 0#, потрібно лише встановити власний номер станції, як показано на наступному рисунку.



Максимальна кількість станцій інспекції стосується загальної кількості PLCs, що використовуються в мережі. Якщо ви використовуєте загалом 6 PLCs, вам потрібно встановити максимальну кількість станцій інспекції на 6, а номер станції для цих 6 PLCs встановити від 0 до 5. Якщо ви хочете додати ще дві PLCs до мережі без будь-якого переривання її роботи, ви можете встановити максимальну кількість станцій інспекції на 8, а PLC станції, що будуть додані в

майбутньому, на 6 та 7 respectively. Коли 6 та 7 підключаються до мережі, вони автоматично виявляються N:N протягом однієї секунди та включаються в обмін даними з 0-5.

10.6 Декілька стратегій керування

10.6.1 Визначення головної станції

Станція 0# за замовчуванням є головною станцією. Тільки станція 0# може ініціалізувати та запустити всю мережу. Налаштування щодо N:N, такі як режим оновлення, додатковий час затримки та кількість спроб, потрібно конфігурувати лише через станцію 0#. Коли станція 0# змінює відповідну конфігурацію в режимі онлайн, резервна головна станція бере на себе керування мережею під час процесу завантаження системних блоків. Коли станція 0# завершує завантаження системного блоку, резервна головна станція передає місце головної станції станції 0#.

Стратегія головної станції в мережі: станція з мінімальним номером станції діє як головна станція.

10.6.2 Максимальна кількість інспекційних станцій

Під час встановлення макс. кількості станцій інспекції рекомендується встановити макс. кількість станцій інспекції рівною загальній кількості PLCs, що входять до фактичної мережі, і встановити номери станцій, починаючи з 0# по порядку. Коли макс. кількість станцій інспекції встановлено на N#, мережа керує лише станціями в діапазоні від 0# до N-1#. Зокрема, якщо макс. кількість станцій інспекції, встановлена користувачем, є неправильною, тобто коли максимальна кількість станцій інспекції менша за кількість PLCs, що входять до фактичної мережі 485, PLCs з номерами станцій, які більші або дорівнюють макс. кількості станцій інспекції, не можуть транслювати свої дані, але можуть приймати ширококомовні дані від PLCs з номерами станцій, які менші за макс. кількість станцій інспекції.

10.6.3 Мультимайстер та підпорядкований (M: N)

А Мережу зі структур з кількома ведучими та кількома веденими пристроями можна побудувати за допомогою N:N. Значення "ведучий" та "ведений" тут таке: "ведучий" вказує на PLC, який може не лише записувати власні елементи M та D, а й зчитувати елементи M та D інших станцій; "ведений" вказує на PLC, який може лише зчитувати елементи M та D інших станцій. За встановленої максимальної кількості станцій опитування (ця кількість також залежить від режиму оновлення), PLC, номер станції якого менший за кількість станцій опитування, може використовуватися як "ведучий", тоді як PLC, номер станції якого більший за кількість станцій опитування, може використовуватися лише як "ведений". Ведена станція може лише зчитувати відповідні елементи M та D ведучої станції. Ці елементи M та D мають відповідний зв'язок з кожною ведучою станцією згідно з режимом оновлення у ведучій станції. Ви можете звернутися до таблиці спільних елементів M та D N:N. Для веденої станції у цих таблицях немає відповідних елементів M та D.

10.6.4 Приклади використання N:N

Усього є 5 PLCs, а номер станції варіюється від 0# до 4#. Ви можете вибрати 3 для режиму оновлення 3. Якщо ви хочете зберегти суму D100 у 0#PLC та D305 у 2#PLC у D500 4#PLC, ви можете запрограмувати це наступним чином:

Програмування 0#: **MOV D100 D7700**

Програмування 2#: **MOV D305 D7716**

Програмування 4#: **ADD D7700 D7716 D500**

Опис: Цей приклад демонструє N:N одношарову мережу. У мережі є 5 PLC станцій, а режим оновлення встановлено на 3. Кожна станція може бути розподілена з 8 D елементами та 64 M елементами. D елементи, розподілені на станцію 0#, знаходяться в діапазоні від D7700 до D7707, ті, що розподілені на станцію 2#, знаходяться в діапазоні від D7716 до D7723, а ті, що розподілені на станцію 4#, знаходяться в діапазоні від D7732 до D7739. Збереження значення D100 станції 0# у спільній зоні запису D7700, розподілений мережею, та значення D305 станції 2# у спільній зоні запису D7716, розподілений мережею. Виконання операції додавання D7700 та D7716 у 4#PLC та збереження суми в локальному елементі D500.

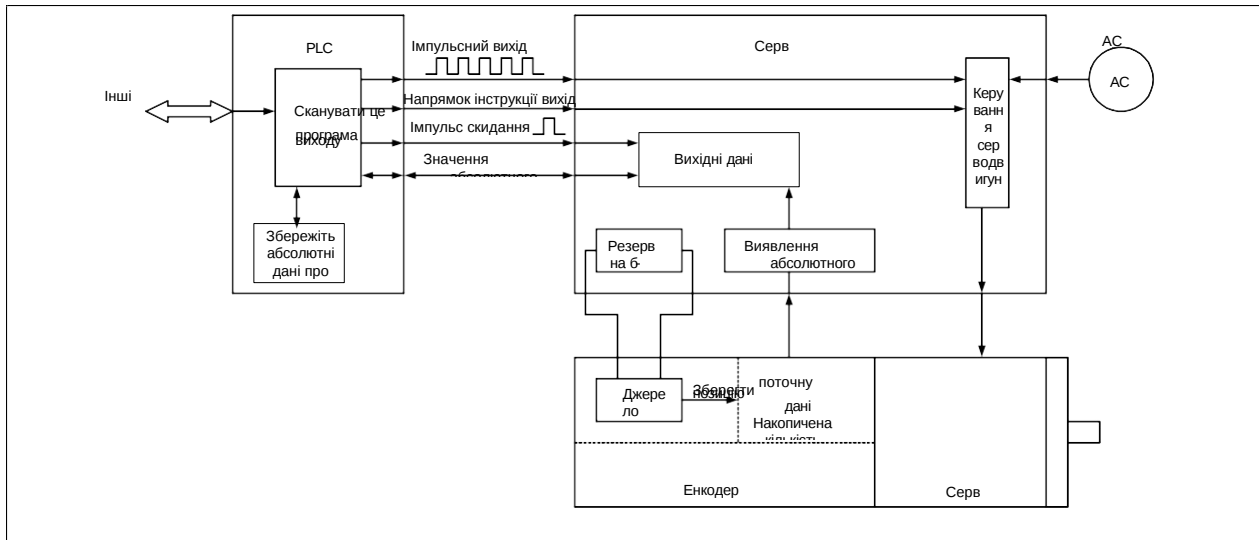
Розділ 11 Посібник з функції позиціонування guide

Розділ 11 Посібник з функції позиціонування guide.....	316
11.1 Система позиційного керування system.....	317
11.1.1 Абсолютна позиція системи.....	317
11.1.2 Система позиційного керування system.....	318
11.1.3 Процес позиціонування керування.....	319
11.2 Огляд функцій позиціонування серії VC PLC.....	319
11.3 Примітка щодо використання інструкцій позиціонування.....	322
11.4 Спеціальні елементи, пов'язані з інструкціями позиціонування.....	323
11.4.1 Вихідні осі серії VC PLC.....	323
11.4.2 Вихідні осі серії VC1 PLC.....	324
11.5 Екземпляр застосунку.....	324
11.5.1 Метод конфігурації PLS оболонки інструкції.....	324

11.1 Система позиційного керування system

11.1.1 Абсолютна позиція системи

The absolute position system obtains the absolute position data of the servo motor on the stroke by detecting the current position data of the servo motor encoder and total number of running turns. According to this principle, an absolute coordinate system can be established on the mechanical stroke. The following figure is a functional block diagram of an absolute position



system.

Рисунок 11-1 Функціональна схема системи абсолютного позиціонування

На наведеному вище рисунку, на відміну від звичайного інкрементального енкодера, поточні дані про положення та накопичені оберти енкодера системи абсолютного позиціонування можуть зберігатися. Ці дані можуть підтримуватися за допомогою живлення від резервної батареї. Навіть у разі збою живлення сервопривод може отримати поточні дані про абсолютне положення після повторного ввімкнення живлення.

Після увімкнення PLC дані про абсолютне положення можна отримати від сервопривода через зв'язок або інші спеціальні методи, після чого визначається координата положення ходу. PLC використовує інструкцію позиціонування для керування сервоприводом та двигуном, завдяки чому досягається точне позиціонування на ходу та автоматично оновлюються дані про абсолютне положення. Таким чином можна сформувати робочу систему, що базується на координатах абсолютного положення.

The following diagram is a simplified mechanical example of an absolute position system built on the VC series PLC positioning instructions.

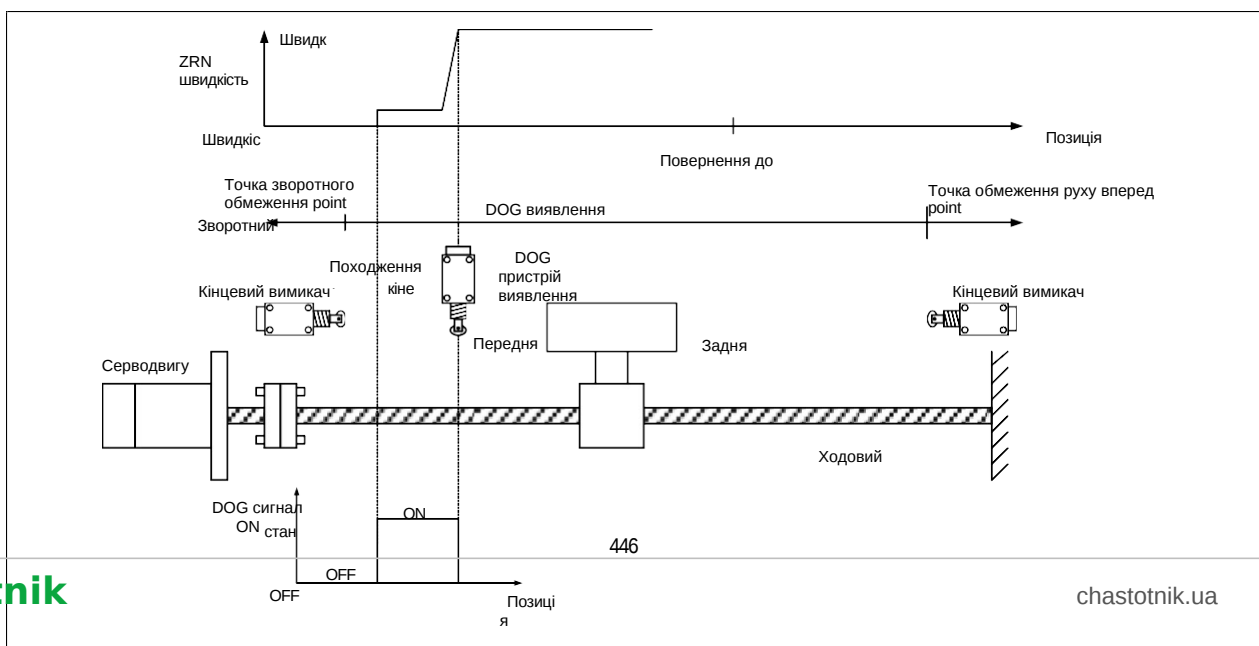


Рисунок 11-2 Схема механічного прикладу системи абсолютного позиціонування системи

Система приводить у рух ходовий гвинт за допомогою серводвигуна для переміщення столу. Положення столу в межах ходу визначається абсолютним енкодером. Під час процесу повернення у нульове положення, коли пристрій сигналу DOG виявляє передній край столу (встановлено), серводвигун сповільнюється до швидкості повзання; коли пристрій сигналу DOG виявляє, що задній край столу (скидання), це є сигналом досягнення початкового положення, і PLC зупиняє високошвидкісний імпульсний вихід. Кінцевий вимикач прямого ходу та кінцевий вимикач зворотного ходу повинні бути встановлені. Оскільки інструкція ZRN не здатна автоматично шукати сигнал DOG, необхідно розпочати операцію ZRN далі переднього краю пристрою виявлення DOG. Ви можете виконати операцію покрокового переміщення для ручного регулювання положення столу за допомогою проектування та програмування.

11.1.2 Система позиційного керування system

Систему керування позиціонування можна розділити на систему керування з розімкненим контуром, систему керування з напівзамкненим контуром та систему керування із замкненим контуром відповідно до різних режимів керування.

Система керування з розімкненим контуром — це система керування, у якій регульовальна система не приймає керування за зворотним зв'язком і лише керує виходом, також відома як система керування без зворотного зв'язку. Система керування з розімкненим контуром здебільшого складається з контролера, крокового привода, та крокового двигуна. Контролер надсилає імпульсну інструкцію на кроковий привод, а потім кроковий двигун приводить у рух стіл на певну відстань. Цей тип системи є відносно простим, стабільним у роботі та легким для опанування, проте він не може виявляти помилки, як і не може їх виправляти. Його точність керування та здатність до придушення перешкод є відносно низькими, і він чутливий до змін параметрів системи. Тому він зазвичай використовується лише в застосунках, де зовнішній вплив не береться до уваги, або інерція мала, або точність не є високою.

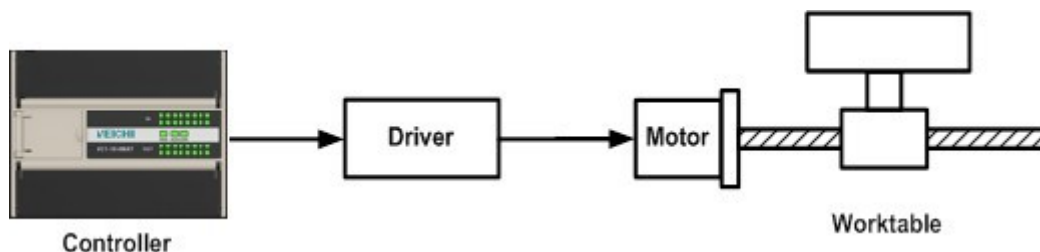


Рисунок 11-3 Схема системи керування у розімкненому контурі діаграма

Система керування із замкненим контуром — це система автоматичного керування, у якій замкнений контур складається із прямого каналу передачі сигналу та каналу зворотного зв'язку, також відома як система керування зі зворотним зв'язком. Система керування із замкненим контуром зазвичай складається з контролера, сервопривода, серводвигуна, детектора тощо. Система автоматично визначає фактичне переміщення столу та передає його назад до контролера для керування із замкненим контуром. Цей тип системи має високу точність позиціонування, проте система є складною, важкою в налагодженні та обслуговуванні, а її ціна є відносно високою. Вона переважно використовується для високоточних застосувань та великих CNC верстатів.

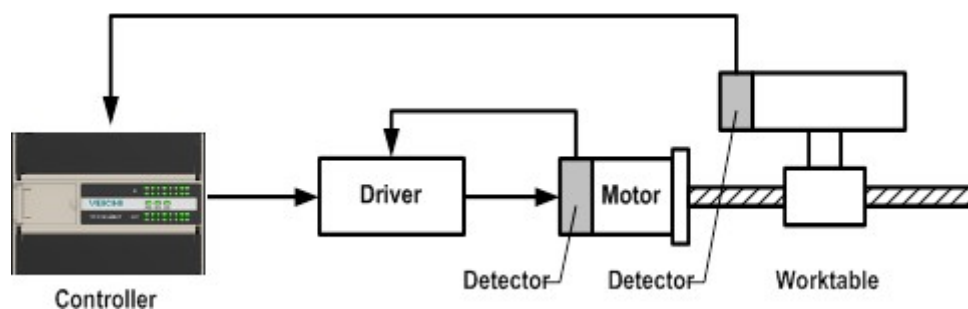


Рисунок 11-4 Схема системи керування із замкненим контуром systemdiagram

Напівзамкнена система керування працює подібно до замкненої системи керування, за винятком того, що детектор встановлено не на столі, а на валу серводвигуна. Точність, швидкість та динамічні характеристики цієї системи перевершують систему керування з розімкненим контуром. Її складність і вартість нижчі, ніж у замкненої системи керування. Вони переважно використовуються для застосувань середньої точності та більшості малих і середніх CNC верстатів.

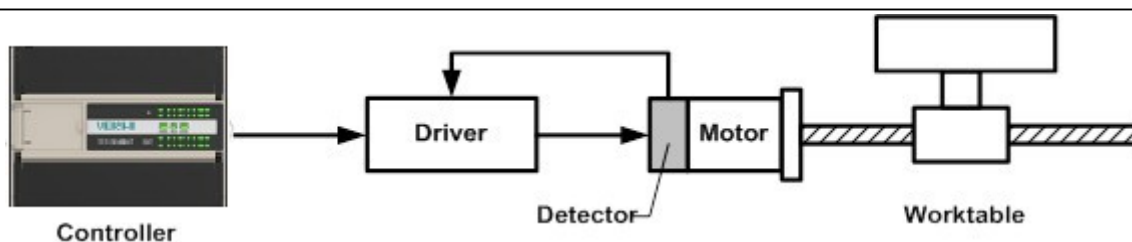


Рисунок 11-5Semi-схема системи керування із замкненим контуром зворотного зв'язку

11.1.3 Процес позиціонування керування

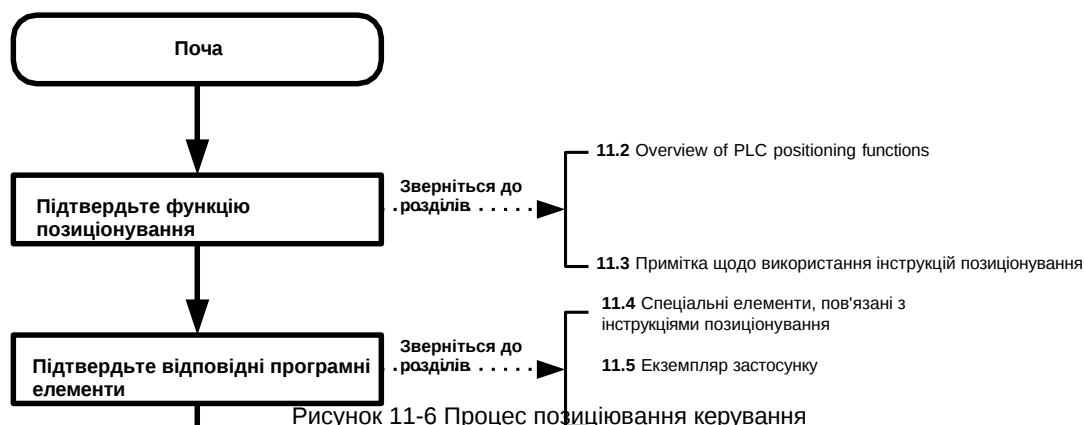


Рисунок 11-6 Процес позиціонування керування

11.2 Огляд функцій позиціонування серії VC PLC

Функції позиціонування, що підтримуються серією VC PLCs, включають просте імпульсне вихідне позиціонування, двовісну лінійну та дугову інтерполяцію траєкторії, а також синхронне керування рухом між осями, що може широко використовуватися в системах керування позиціонуванням для керування кроковими та сервоприводами різних брендів. Дані про абсолютну позицію можна отримати за допомогою засобів, наданих відповідним сервоприводом.

Таблиця 11-1 Огляд функцій позиціонування головного модуля серії VC PLC

Ім'я	VC3(Планування)	VC2(планування)	VC1	VC1S(планування)
Керовані осі	8-вісь	3-осьовий	3-осьовий	2-осьовий
Макс. вихідна частота	200 кГц	100 кГц	100 кГц	100 кГц
Режим імпульсного виходу mode	Відкрити колектор	Відкрити колектор	Відкрити колектор	Відкрити колектор
Тип імпульсного виходу типу	Імпульс+напрямок, позитивні та негативні імпульси	Імпульс+напрямок	Імпульс+напрямок	Імпульс+напрямок
Прискорення та сповільнення обробки	Трапецеїдальне прискорення/сповільнення, трикутне прискорення та вимкнення сповільнення	Трапецеїдальне прискорення/сповільнення	Трапецеїдальне прискорення/сповільнення	Трапецеїдальне прискорення/сповільнення

Функція інтерполяції	2-осьовий лінійний інтерполяція та кругова інтерполяція	—	—	—
Функція синхронізації	Синхронізація положення, електронний редуктор	—	—	—
Абсолютний позиція виявлення	ABS інструкція прочитати	—	—	—
Діапазон позиціонування	-2 147 483 648—+2 147 483 647 (імпульс)			

*Примітка: Тільки інструкція інтерполяції підтримує режими виходу додатних та від'ємних імпульсів modes.

Під час підключення до сервоприводу необхідно встановити вхідний сигнал підсилювача сервоприводу в режим від'ємної логіки. Форма імпульсного виходу визначається наступним чином:

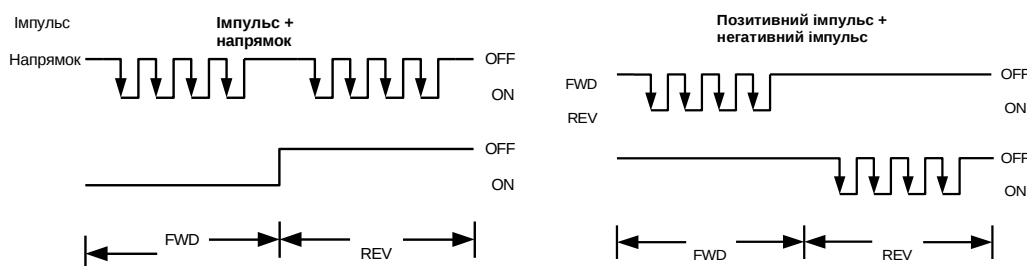
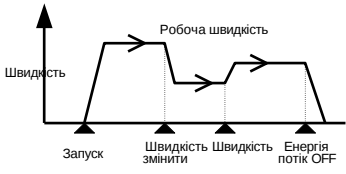
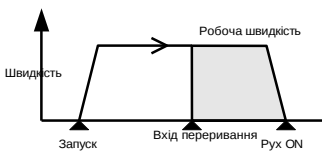
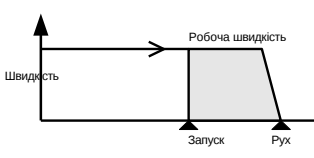
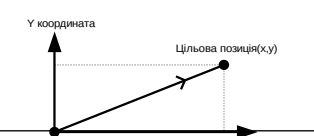


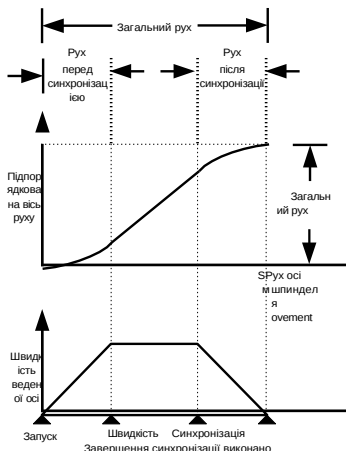
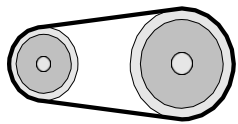
Рисунок 11-7 Визначення форми імпульсного виходу

Примітка: Високошвидкісну I/O інструкцію також можна використовувати як імпульсний вихід, але лише для вихідного керування імпульсним сигналом, і вона не керує сигналом напрямку. Коли використовуються ці високошвидкісні I/O інструкції, відповідні елементи позиції SD піддаються накопичувальній обробці у прямому напрямку. Коли ви хочете керувати сервоприводом під час руху вперед, вам потрібно запрограмувати та встановити сигнал напрямку сервоприводу на ON, а для руху назад встановити на OFF.

Таблиця 11-2 Перелік функцій позиціонування головних модулів серії VC PLC

Ім'я	Дія	Вміст	VC3	VC2	VC1	VC1S
DSZR	Швидкість Швидкість повзання Нуль: ON DOG: ON Запуск Повернення до початкової швидкості	Він діє відповідно до заданої швидкості повернення до нуля і може автоматично шукати сигнал DOG. Сповільнюється до швидкості повзання після виявлення DOG (датчик DOG — це ON). Зупиняється при надходженні нульового сигналу, ZRN завершено.	•	•	•	
ZRN	Швидкість Швидкість повзання DOG: OFF DOG: ON Запуск Повернення до нульової швидкості	Він діє відповідно до заданої швидкості повернення до нуля та сповільнюється до швидкості повзання, як тільки виявлено DOG (датчик DOG — це ON). Зупинка відбувається, коли датчик DOG — це OFF, і ZRN завершено.	•	•	•	•
DRVI	Швидкість Робоча швидкість Запуск Цільова позиція Рух	Він діє відповідно до встановленої швидкості роботи, зупиняється у цільовій позиції та використовує відносні	•	•	•	•

		координати для позиції.				
Ім'я	Дія	Вміст	VC3	VC2	VC1	VC1S
DRVA		Він діє відповідно до налаштування швидкість роботи, зупиняється на цільова позиція, і використовує абсолютні координати для the позиція	•	•	•	•
PLSV		Він діє відповідно до налаштування швидкість роботи. Якщо робота швидкість змінюється, він працює на тій новій швидкості; якщо потік енергії flow стає недійсним, імпульс привод зупиняється. Коли там є прискорення/сповільнення, це прискорення/сповільнення є виконується, коли швидкість є змінено.	•	•	•	•
ABS		Зчитування поточного абсолютного дані про положення від сервоприводу				
DVIT		Він діє відповідно до налаштування робота на швидкості. Якщо вхід переривання є ON, він виконує вказану кількість циклів імпульси, а потім сповільнюється до STOP.			•	
STOPDV		Коли операція позиціонування є виконується, якщо це інструкцію запущено, вона виконує вказану кількість імпульсів та потім сповільнюється до зупинки.	•	•		
CW		Він переміщується до цільової позиції вздовж колової траєкторії у напрямку за годинниковою стрілкою на тому задана лінійна швидкість.	•			
CCW		Він рухається до цільової позиції вздовж колової траєкторії у напрямку проти годинникової стрілки на зазначена лінійна швидкість.	•			
LIN		Він переміщується до цільової позиції вздовж лінійної траєкторії згідно з зазначеним	•			

Початкова точка		X координата	векторна швидкість.				
Ім'я	Дія		Вміст	VC3	VC2	VC1	VC1S
MOVELIN K			Підпорядкована вісь рухається разом із віссю шпинделя та підтримує з нею синхронність у межах заданого діапазону положення, забезпечуючи керування прискоренням/сповільненням під час перехідного процесу до та після синхронізації.				
GEARBO X			Він керує підпорядкованою віссю для руху разом із віссю шпинделя відповідно до певного коефіцієнта електронного редуктора.				

Інструкція позиціонування та високошвидкісна інструкція видають керований імпульс на високошвидкісний порт відповідно до налаштувань, і вихід імпульсу не залежить від циклу сканування користувачської програми. Детальнішу інформацію щодо використання цих інструкцій див. у розділі 6.10 "Високошвидкісна I/O інструкція". У програмі використання інструкцій позиціонування або високошвидкісних інструкцій для різних вихідних портів дозволяє отримати незалежний високошвидкісний імпульсний вихід на відповідному вихідному порті.

11.3 Примітка щодо використання інструкцій позиціонування

Коли інструкція позиціонування або високошвидкісна інструкція є дійсною (включаючи завершення виводу), інші операції на тому самому порту є недійсними. Інші інструкції мають коректний вивід лише тоді, коли інструкція високошвидкісного імпульсного виводу є недійсною.

Якщо на одному порту є декілька інструкцій позиціонування або високошвидкісних інструкцій, перша дійсна інструкція займає вихідний порт, а наступні дійсні інструкції вихідний порт не займають.

- Транзисторний вихід
Серія VC PLC з транзисторним виходом є обов'язковою.
- Вимоги до інструкцій позиціонування при програмуванні
Інструкції позиціонування можна використовувати багаторазово у програмі, але потрібно звернути увагу на:
 1. Ви не можете одночасно використовувати привод та інші інструкції позиціонування або високошвидкісного імпульсного виходу на одній і тій самій точці високошвидкісного імпульсного виходу. А Точка високошвидкісного імпульсного виходу може керуватися лише однією інструкцією позиціонування (або високошвидкісною інструкцією) в будь-який момент часу.
 2. Коли потік енергії інструкції позиціонування розривається, інструкцію можна знову активувати лише тоді, коли потік енергії буде відновлено після одного або кількох PLC циклів сканування.
- Основні моменти щодо одночасного використання інструкцій високошвидкісного керування та позиціонування одночасно
З огляду на реалізацію функції, рекомендується використовувати інструкцію позиціонування для заміни цих інструкцій високошвидкісного імпульсного виходу (PLSY, PLSR та PLS), щоб завершити автоматичне оновлення елементів SD абсолютного положення.

Елементи SD абсолютного положення можуть використовуватися для збереження та оновлення поточного

абсолютного положення після використання інструкції позиціонування. Автоматичне збільшення та зменшення значення елемента SD абсолютного положення визначається на основі значення зміни накопичення вихідних імпульсів елемента SD та напрямку руху під час виклику інструкції позиціонування, тому вони пов'язані. Не виконуйте операцію запису в елемент SD накопичення вихідних імпульсів під час використання інструкції позиціонування. В іншому випадку дані елемента SD абсолютного положення можуть бути спотворені.

Якщо необхідно одночасно використовувати інструкції позиціонування та інші інструкції виводу високошвидкісних імпульсів (PLSY, PLSR та PLS), вам потрібно написати програму PLC так, щоб дані елементів абсолютного положення SD у регістрі абсолютного положення могли бути коректно оновлені.

- Обмежувальні умови для фактичної вихідної частоти позиційної інструкції

Коли виконується інструкція позиціонування, мінімальна частота фактичного вихідного імпульсу обмежується наступною формулою:

$$F_{\min\_acc} = \sqrt{\frac{F_{\max} \times}{500}}$$

У наведеній вище формулі, $F_{\max}$ вказує на макс. швидкість; T вказує на час розгону/гальмування, а одиницею вимірювання є мс.

Результат обчислення $F_{\min\_acc}$ є мінімальним значенням ліміту вихідної частоти value.

Якщо вихідна частота, вказана в інструкції з позиціонування, дорівнює F , існує три випадки фактичної вихідної частоти, як наведено нижче.

- Коли F менше базової частоти або більше макс. частоти $F_{\max}$, фактично немає виходу.
- Коли F менше ніж $F_{\min\_acc}$, а фактичний вихід становить $F_{\min\_acc}$.
- Коли F більше або дорівнює $F_{\min\_acc}$, і менше або дорівнює $F_{\max}$, вихід дорівнює F .

11.4 Спеціальні елементи, пов'язані з інструкціями позиціонування

11.4.1 Вихідні осі серії VC PLC

Визначення та призначення вихідних осей приводу серії VC1 PLCs наведено в таблиці нижче.

Таблиця 11-3 Визначення виходу VC1-1616MAT осі

Вихід вісь	Підтримується режим	Визначення вихідних точок		Режим виходу визначення
0	Імпульс + напрямок	Імпульс	Y0	
		Напрямок	Будь-які вихідні точки, окрім Y0	
1	Імпульс + напрямок	Імпульс	Y1	
		Напрямок	Будь-які вихідні точки, крім Y1	
2	Імпульс + напрямок	Імпульс	Y2	
		Напрямок	Будь-які вихідні точки, крім Y2	

□ Примітка

При використанні будь-якої з вихідних осей для підключення сервоприводу необхідно враховувати сполучення вихідних точок. Усі вихідні позиціонування по одній осі можуть використовувати лише режим "імпульс + напрямок". Тільки інструкція інтерполяції може використовувати режим "FWD + REV". Одночасно може виконуватися лише одна група інструкцій інтерполяції. Одна група інструкцій інтерполяції займає 4 порти: Y0, Y1, Y2, Y3, серед яких Y0 та Y1 є X-осьми, а Y2 та Y3 є Y-осьми. Коли є подвійні імпульси, Y0 — це позитивний імпульс, а Y1 — негативний імпульс, тоді як Y2 — це позитивний імпульс, а Y3 — негативний імпульс. У режимі "імпульс + напрямок" необхідно врахувати, що при виборі

сигналу напрямку вихідна точка не може використовуватися для інших цілей одночасно. Наприклад, ви не можете визначати ту саму вихідну точку, що й для імпульсу або сигналу напрямку інших вихідних осей.

11.4.2 Вихідні осі серії VC1 PLC

Визначення та призначення вихідних осей серії VC1 PLCs наведено в таблиці нижче.

Таблиця 11-5 Визначення виходу серії VC1 осі

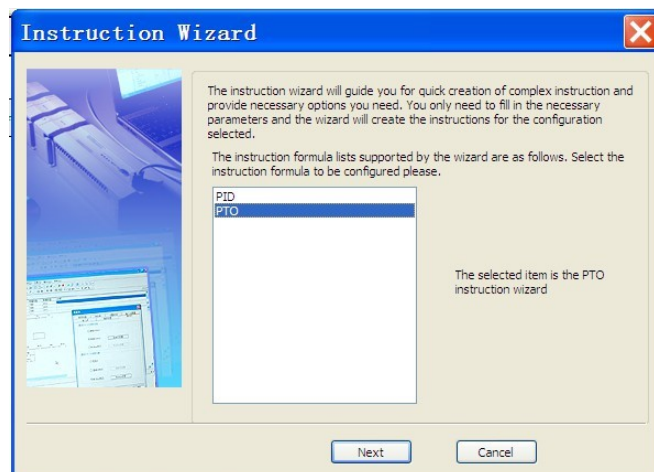
Вихід вісь	Підтримується режим	Визначення вихідних точок		Режим виходу визначення
0	Імпульс + напрямок	Імпульс	Y0	Імпульс + напрямок Імпульс Напрямок OFF ON FWD REV
		Напрямок	Будь-які вихідні точки, окрім Y0	
1	Імпульс + напрямок	Імпульс	Y1	
		Напрямок	Будь-які вихідні точки, крім Y1	
2	Імпульс + напрямок	Імпульс	Y2	
		Напрямок	Будь-які вихідні точки, крім Y2	

11.5 Екземпляр застосунку

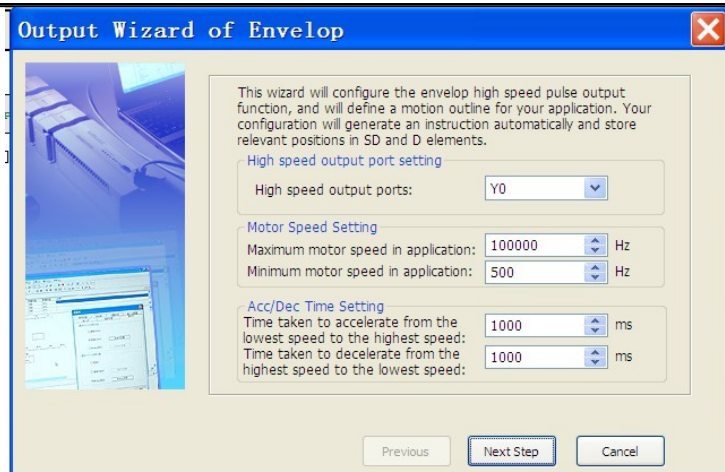
11.5.1 Метод конфігурації PLS оболонки інструкції

- Метод конфігурації PLS інструкції envelope

Інструкція PLS генерується за допомогою майстра інструкцій PTO. В Auto Studio ви можете вибрати **Інструмент - >Майстер інструкцій...**, щоб налаштувати інструкцію PLS. Виберіть PTO, як показано нижче.



Натисніть **Далі**, щоб увійти в інтерфейс конфігурації, де можна налаштувати точки високошвидкісного імпульсного виходу, макс. та мін. частоту високошвидкісного імпульсного виходу, а також час ACC/DEC, як показано на наступному рисунку.



Під час часу розгону/гальмування швидкість розгону всіх сегментів кривої обвідної є постійною. Наприклад, якщо налаштування такі, як показано вище, то час розгону, необхідний для прискорення двигуна від 20000 Гц до 50000 Гц, становить:

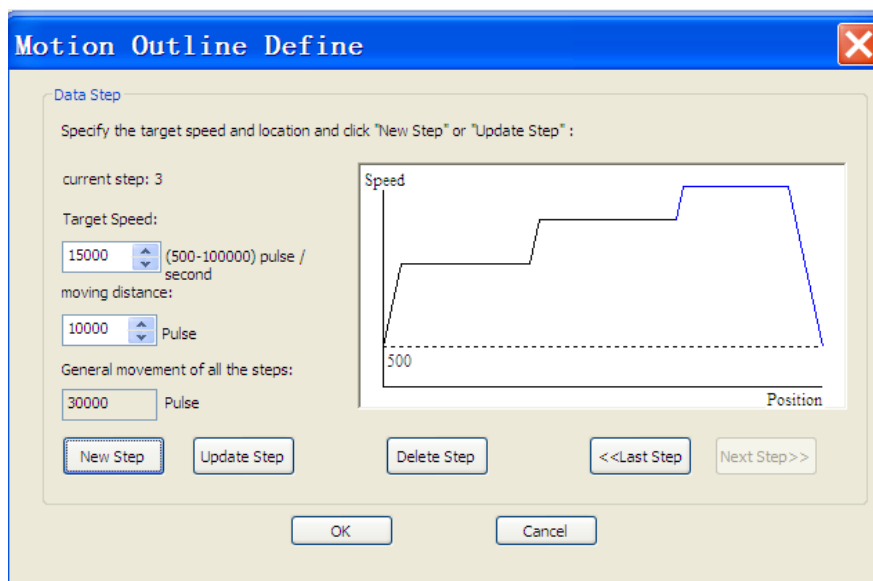
$$1000 \times (50000 - 20000) \div (100000 - 5000) = 316 \text{ (мс)} = 0.316 \text{ (с)}$$

Протягом часу прискорення загальну кількість вихідних імпульсів можна розрахувати за допомогою формули обчислення площі трапеції:

$$(20000 + 50000) \times 0.316 \div 2 = 11060 \text{ (кількість імпульсів)}$$

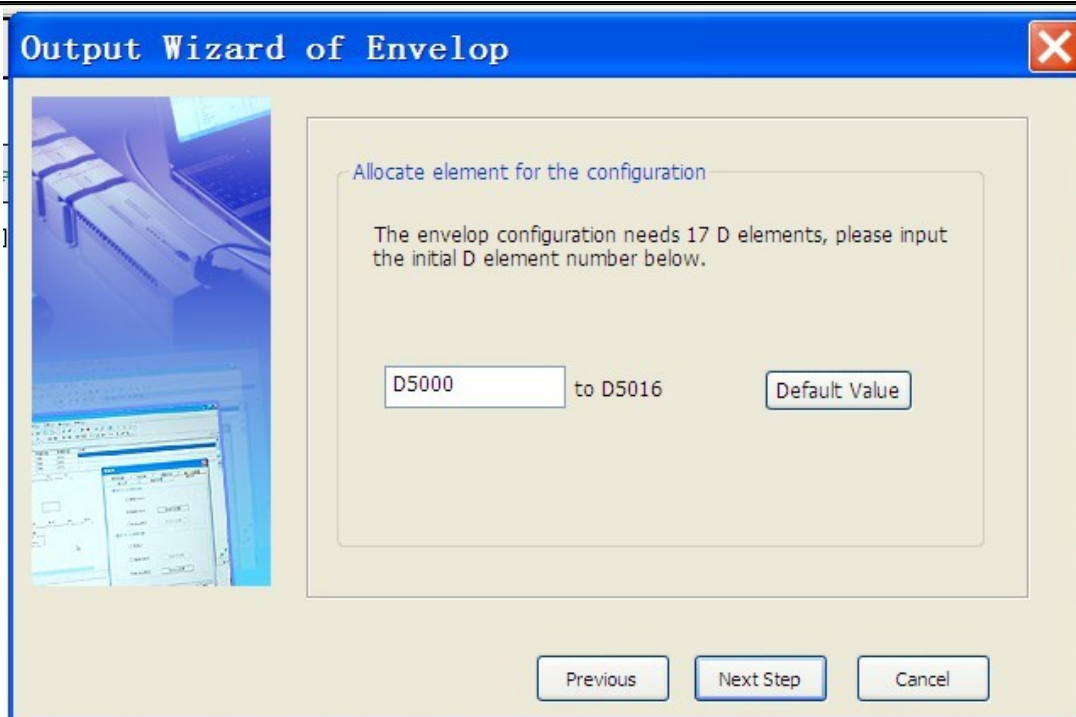
Тому, якщо у вас є вимога щодо часу або кількості імпульсів під час прискорення/сповільнення, вам необхідно виконати відповідні розрахунки перед встановленням макс. швидкості, мін. швидкості та часу прискорення/сповільнення.

Після натискання **Наступний Крок** на рисунку вище, введіть **Контур Руху Визначення** на рисунку нижче. Ви можете спочатку ввести цільову швидкість і відстань переміщення для першого кроку, а потім натиснути **Новий Крок**. Після цього ви можете ввести цільову швидкість і відстань переміщення для другого кроку, натиснути **Новий**



Крок і так далі, а наприкінці натиснути **ОК**.

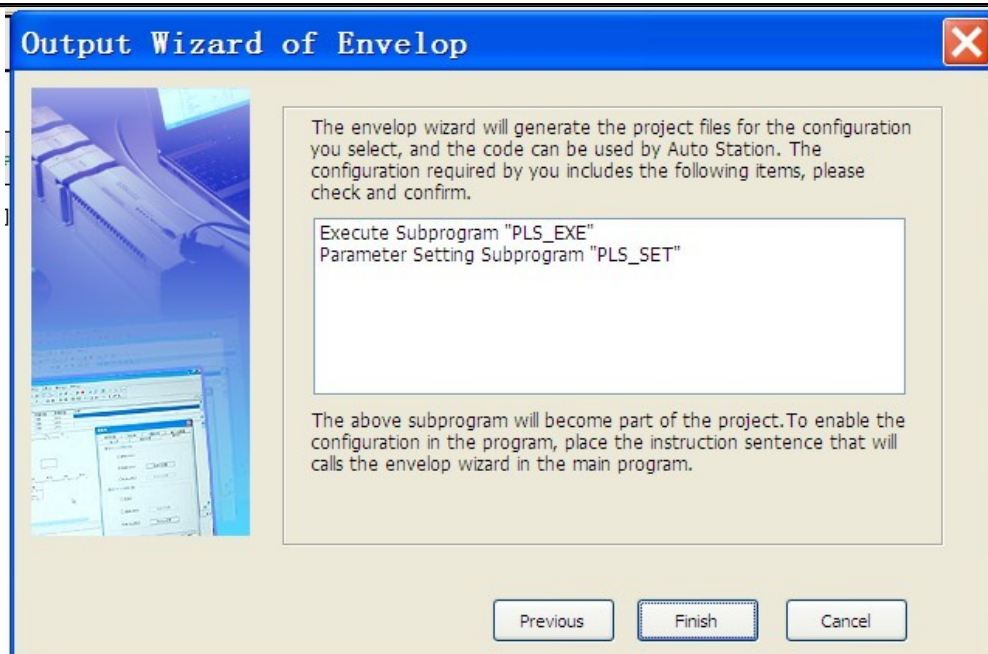
Вищевказана конфігурація зберігається в елементах D, і ви можете вибрати, в яких елементах D зберігати ці налаштування, як показано нижче.



Майстер створить дві підпрограми, одна з яких є підпрограмою налаштування параметрів, а інша — PLS підпрограмою виконання, як показано нижче. Вам потрібно переконаватися, що підпрограма налаштування параметрів правильно викликається та виконується (відповідні D елементи призначені) під час програмування в основній програмі, а потім викликати підпрограму виконання.



На цьому етапі всі налаштування завершено, як показано на рисунку нижче. Натисніть **Завершити**, щоб завершити РТО конфігурацію.



Додаток А Спеціальне допоміжне реле

Усі спеціальні допоміжні реле ініціалізуються, коли PLCs змінюється з STOP на RUN. Спеціальні допоміжні реле, які були встановлені в системних налаштуваннях, встановлюються на попередньо задане значення в системному блоці після ініціалізації.

□ Примітка

Зарезервовані елементи SD та SM не вказані в таблиці. Зарезервовані елементи SM за замовчуванням доступні для читання та запису (R/W).

1. PLC робочий стан прапорець

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM0	Моніторинг RUN біт	Цей біт завжди ON у RUN, і OFF у STOP.	R	✓	✓	✓	✓	✓
SM1	Біт імпульсу початку вого пуску	Цей біт встановлюється у високий рівень, коли користувачка програма переходить з STOP на RUN, і встановлюється у низький рівень після цикл сканування	R	✓	✓	✓	✓	✓
SM2	Увімкнення живлення-ON прапорець біта	Цей біт встановлюється ON після того, як на систему подано живлення ON, і встановить OFF після циклу сканування	R	✓	✓	✓	✓	✓
SM3	Системна помилка	Цей біт встановлюється, коли виникає системна помилка після увімкнення живлення або після того, як PLC зміниться з STOP на RUN, або скидання, якщо не виникає системна помилка	R	✓	✓	✓	✓	✓
SM4	Напруга батареї низька	Цей біт встановлюється, коли напруга акумулятора занижка низький, або скидання, якщо напругу батареї виявлено вищою за 2.4 V.	R		✓	✓	✓	✓
SM7	Режим роботи без	Якщо цей біт встановлено в 1 (налаштовується лише через системний блок), система не повідомляє про помилки, спричинені втратою даних	R		✓	✓	✓	✓

	аккумулятор	резервної батареї або втратою таблиць примусового керування через резервну батарею відмова						
SM8	Константа режим сканування mode	Встановлення цього біта, і час сканування є сталим (налаштовується лише через системний блок)	R	✓	✓	✓	✓	✓
SM9	Режим запуску вхідної точки	Встановлення цього біта, і PLC може змінюватися з STOP на RUN, коли призначена точка входу X має стан ON (налаштовується лише через система блок)	R	✓	✓	✓	✓	✓

2. Біт роботи годинника

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM10	10 мс тактовий сигнал	Тактова осциляція з циклом 10 мс (реверсування кожного півперіоду. Перший півперіод дорівнює 0 протягом цього робота користувачської програми)	R	✓	✓	✓	✓	✓
SM11	100 мс тактовий сигнал	Тактова осциляція з циклом 100 мс (реверсування кожного півциклу. Перший півцикл дорівнює 0 протягом робота користувачської програми)	R	✓	✓	✓	✓	✓
SM12	1 с годинник	Тактова осциляція з циклом 1 с (реверсування кожні пів циклу. Перший пів цикл дорівнює 0 під час роботи користувачської програми)	R	✓	✓	✓	✓	✓
SM13	1 мін тактова частота	Тактова осциляція з циклом 1 хв (реверсування кожні пів циклу. Перша половина циклу дорівнює 0 протягом робота користувачської програми)	R	✓	✓	✓	✓	✓
SM14	1 година годинника	Тактова осциляція з циклом 1 година (реверсування кожні пів циклу. Перший пів цикл дорівнює 0 протягом робота користувачської програми)	R	✓	✓	✓	✓	✓
SM15	Сканування циклічних коливань увімкнено біт	Цей біт інвертується один раз за кожен цикл сканування (перший цикл дорівнює 0 під час виконання користувачської програми)	R	✓	✓	✓	✓	✓

3. Помилка виконання програми користувача error

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM20	Помилка виконання інструкції	Цей біт встановлюється у разі помилки виконання інструкції, а відповідний код типу помилки записується у SD20. Цей біт скидається після програми інструкцію виконано.	R	✓	✓	✓	✓	✓
SM21	Елемент інструкції номер підрядковий переповнення	Цей біт встановлюється у разі помилки виконання інструкції, а відповідний код типу помилки записується у SD20.	R	✓	✓	✓	✓	✓

SM22	Неприпустимий параметр інструкції	Цей біт встановлюється у разі помилки виконання інструкції, а відповідний код типу помилки записується у SD20. Цей біт скидається після програми інструкцію виконано.	R	✓	✓	✓	✓	✓

4. Керування перериванням

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM25	Передній фронт входу X0 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM26	Передній фронт входу X1 переривання прапорець біта	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM27	Передній фронт входу X2 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM28	Вхід X3 по передньому фронту переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM29	Передній фронт входу X4 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM30	Передній фронт входу X5 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM31	Передній фронт входу X6 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM32	Передній фронт входу X7 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM33	Вхід X0 по спадному фронту переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM34	Задній фронт входу X1 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM35	Вхід X2 по спадному фронту переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM36	Задній фронт входу X3 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM37	Вхід X4 по спадаючому фронту переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM38	Вхід X5 по спадному фронту переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM39	Задній фронт входу X6 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM40	Задній фронт входу X7 переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM41	SOM 0 кадр передавання прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM42	SOM 0 переривання прийому кадру interrupt увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W		✓	✓	✓	✓
SM43	SOM 1 кадр передачі прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W		✓	✓	✓	✓

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5

SM44	COM 1 переривання прийому кадру переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W		✓	✓	✓	✓
SM45	COM 2 кадр передачі прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W		✓	✓	✓	✓
SM46	COM 2 переривання прийому кадру interrupt увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W		✓		✓	
SM47	Прапорець дозволу переривання за таймером 0 біт	Увімкнути, коли встановлено як 1	R/W	✓	✓		✓	
SM48	Прапорець дозволу переривання за таймером 1 біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	
SM49	Прапорець дозволу переривання за таймером 2 біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	
SM50	PTO (Y0) вихід завершення прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM51	PTO (Y1) вихід завершення прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM52	PTO (Y2) вихід завершення прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W	✓	✓	✓	✓	✓
SM53	PTO (Y3) вихід завершення прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W			✓	✓	✓
SM54	PTO (Y4) вихід завершення прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W				✓	
SM55	PTO (Y5) вихід завершення прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W			✓	✓	
SM56	PTO (Y6) вихід завершення прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W				✓	
SM57	PTO (Y7) вихід завершення прапорець дозволу переривань біт	Увімкнути, коли встановлено як 1	R/W				✓	
SM58	Високошвидкісний лічильник переривання увімкнути прапорець біт	Увімкнути, коли встановлено як 1	R/W				✓	
SM59	Інтерполяція завершення прапорець дозволу переривання 1 біт	Увімкнути, коли встановлено як 1	R/W				✓	
SM60	Інтерполяція завершення прапорець дозволу переривання 2 біт	Увімкнути, коли встановлено як 1	R/W				✓	
SM61	Інтерполяція завершення прапорець дозволу переривання 3 біт	Увімкнути, коли встановлено як 1	R/W				✓	
SM62	Інтерполяція завершення прапорець дозволу переривання 4 біт	Увімкнути, коли встановлено як 1	R/W				✓	
SM63	Увімкніть біт прапорця для переривання 0 на основі положення в інструкції з позиціонування	Увімкнути, коли встановлено як 1	R/W				✓	
SM64	Увімкнуті біт прапорця для позиціонування переривання 1 на основі інструкції позиціонування	Увімкнути, коли встановлено як 1	R/W				✓	
SM65	Увімкнуті біт прапорця для позиціонування переривання 2 на основі в інструкції позиціонування	Увімкнути, коли встановлено як 1	R/W				✓	
SM66	Увімкніть біт прапорця для переривання 3 на основі положення в	Увімкнути, коли встановлено як 1	R/W				✓	

	інструкція з позиціонування							
SM67	Увімкніть біт прапорця для переривання за положенням 4 у інструкції з позиціонування	Увімкнути, коли встановлено як 1	R/W				✓	
SM68	Увімкніть біт прапорця для переривання 5 за положенням у інструкції з позиціонування	Увімкнути, коли встановлено як 1	R/W					
SM69	Увімкнути біт прапорця для позиціонування переривання 6 на основі в інструкції позиціонування	Увімкнути, коли встановлено як 1	R/W					
SM70	Увімкніть біт прапорця для переривання за позицією 7 у інструкції з позиціонування	Увімкнути, коли встановлено як 1	R/W				✓	

5. Периферійна інструкція

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM78	Вибір режиму друку	Коли цей біт встановлено в 1, є 1-16 символів. Коли він встановлений в 0, їх зазвичай 8 символи	R/W			✓	✓	✓
SM79	Інструкція периферійного пристрою друку	Коли цей біт встановлено в 1, то друк триває	R			✓	✓	✓

6. Operation прапорець біт

Адреса	Ім'я	Функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM80	Біт прапорця нуля	Цей біт встановлюється, коли відповідний обчислений результат дорівнює нулю. Ви можна очистити цей біт вручну	R/W	✓	✓	✓	✓	✓
SM81	Прапорець перенесення/переповнення біт	Цей біт встановлюється, коли відповідний обчислений результат є перенесенням. Ви можна очистити цей біт вручну	R/W	✓	✓	✓	✓	✓
SM82	Прапорець позики біт	Цей біт встановлюється, коли пов'язаний Розрахований результат є позикою. Ви можете скинути цей біт вручну	R/W	✓	✓	✓	✓	✓

7. DHST/DHSP Завершення порівняння таблиці прапорець

Адреса	Ім'я	Функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM83	Порівняння таблиць прапорець	Встановить на ON, коли таблиця запис завершено.	R/W	✓	✓	✓	✓	✓

8. ASCII інструкція перетворення коду прапорець

Адреса	Ім'я	Функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM85	ASCII режим зберігання інструкцій прапорець	0: MSB/LSB кожного слова зберігає один код ASCII 1: LSB кожного слова зберігає одне ASCII код	R/W	✓	✓	✓	✓	✓

9. Біт моніторингу захоплення імпульсів bit

Адреса	Ім'я	Функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM90	Захоплення імпульсів X0 capture біт моніторингу біт	(1) Усі елементи в таблиці очищуються, коли PLC змінюється з STOP на RUN; (2) Захоплення імпульсів є дійсним, коли інструкції HCNT та SPD виконуються на одному вхідному порту. Детальніше див. у розділі 6.10.9 "SPD: інструкція вимірювання частоти" та розділі 6.10.1 "HCNT: інструкція високошвидкісного лічильника привода".	R/W	✓	✓			
SM91	Захоплення імпульсів X0 capture біт моніторингу біт		R/W	✓	✓			
SM92	Захоплення імпульсів X0 capture біт моніторингу біт		R/W	✓	✓			
SM93	Захоплення імпульсів X0 capture біт моніторингу біт		R/W	✓	✓			
SM94	Захоплення імпульсів X0 capture біт моніторингу біт		R/W	✓	✓			
SM95	Захоплення імпульсів X0 capture біт моніторингу біт		R/W	✓	✓			
SM96	Захоплення імпульсів X0 capture біт моніторингу біт		R/W	✓	✓			
SM97	Захоплення імпульсів X0 capture біт моніторингу біт		R/W	✓	✓			

Примітка:

1. Усі елементи в таблиці очищуються, коли PLC змінюється з STOP на RUN. Захоплення імпульсів є дійсним, коли інструкції HCNT та SPD виконуються на одному й тому самому вхідному порту. Детальнішу інформацію див. у розділі 6.10.9 "SPD: інструкція вимірювання частоти" та розділі 6.10.1 "HCNT: інструкція приводу високошвидкісного лічильника".

10. Чотиричастотний

Адреса	Ім'я	функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM100	Перемикання між одним/чотирма разами фазного входу АВ для X0 та X1	Очищення при від STOP —> RUN;	R/W	✓	✓	✓	✓	✓
SM101	Перемикання між одним/чотирма разами від АВ фазний вхід X2 та X3		R/W			✓	✓	✓
SM102	Перемикання між одним/чотирма разами від АВ фазний вхід X3 та X4		R/W	✓	✓	✓	✓	✓
SM103	Перемикання між одним/чотирма разами від АВ фазний вхід X4 та X5		R/W			✓	✓	✓

11. Вільний порт (PORT0)

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM110	PORT0 передача увімкнути прапорець	Цей біт встановлюється, коли використовується інструкція ХМТ, і скидається після завершення передачі. Ви можете вручну скинути цей біт, щоб призупинити поточну передачу на PORT0. Передача може бути продовжена коли потік енергії увімкнено знову	R/W	✓	✓	✓	✓	✓
SM111	PORT0 отримання прапорця	Цей біт встановлюється, коли використовується інструкція RCV, і скидається після завершення передачі. Ви можете вручну скинути цей біт, щоб	R/W	✓	✓	✓	✓	✓

	дозволу біта	призупинити поточну передачу на PORT0. Передача може бути продовжена коли потік енергії увімкнено знову						
SM112	PORT0 завершени передачі біта прапорець біта	Цей біт встановлюється після того, як передачу завершено	R/W	✓	✓	✓	✓	✓
SM113	PORT0 отримання бітовий прапорець завершення	Цей біт встановлюється після того, як приймання завершено	R/W	✓	✓	✓	✓	✓
SM114	PORT0 очікування прапорець біта	Цей біт встановлюється, коли порт неактивний	R	✓	✓	✓	✓	✓

□ Примітка

SM110-SM114 — це прапорці для станів приймання, завершення та очікування в усіх протоколах зв'язку, що підтримуються PORT0. Наприклад, PORT0 серії VC1 PLCs підтримує протокол Modbus. Незалежно від того, який протокол використовується, функції M110-SM114 залишаються незмінними.

12. Вільний порт (PORT1)

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM120	PORT1 біт дозволу передачі	Цей біт встановлюється, коли використовується інструкція XMT, і скидається після завершення передачі. Ви можете вручну скинути цей біт, щоб призупинити поточну передачу в PORT1. Передача може продовжитися, коли ця потік енергії ON знову	R/W	✓	✓	✓	✓	✓

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM121	PORT1 отримання біта прапора дозволу	Цей біт встановлюється, коли використовується інструкція RCV, і скидається після завершення передачі. Ви можете вручну скинути цей біт, щоб призупинити поточну передачу на PORT1. Передача може продовжитися, коли ця потік енергії ON знову	R/W	✓	✓	✓	✓	✓
SM122	PORT1 прапор завершення передачі біт	Цей біт встановлюється після того, як передачу завершено	R/W	✓	✓	✓	✓	✓
SM123	PORT1 прапор завершення отримання біт	Цей біт встановлюється після того, як приймання завершено	R/W	✓	✓	✓	✓	✓
SM124	PORT1 стан очікування прапорець біт	Цей біт встановлюється, коли порт неактивний	R	✓	✓	✓	✓	✓
SM125	PORT1 MODBUS зв'язок завершення	Цей біт встановлюється, коли обмін даними завершено	R		✓	✓	✓	✓

SM126	PORT1 MODBUS зв'язок помилка	Цей біт встановлюється при виникненні помилки зв'язку	R		✓	✓	✓	✓

□ Примітка

SM120-SM124 — це прапорці для станів отримання, завершення та очікування в усіх протоколах зв'язку, що підтримуються PORT1. Наприклад, PORT1 серії VC1 PLCs підтримує N:N, Modbus та протоколи Free-port. Незалежно від того, який протокол використовується, функції M120-SM124 залишаються незмінними.

13. Розширений вільний порт (PORT2)

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM130	PORT2 біт дозволу передачі	Цей біт встановлюється, коли використовується інструкція ХМТ, і скидається після завершення передачі. Ви можете вручну скинути цей біт, щоб призупинити поточну передачу на PORT2. Передачу можна продовжити коли потік енергії увімкнено знову	R/W		✓	✓	✓	✓
SM131	PORT1 біт прапорця дозволу отримання	Цей біт встановлюється, коли використовується інструкція RCV, і скидається після завершення передачі. Ви можете вручну скинути цей біт, щоб призупинити поточну передачу на PORT2. Передача може бути продовжена коли потік енергії увімкнено знову	R/W		✓	✓	✓	✓
SM132	PORT2 біт прапорця завершення передачі	Цей біт встановлюється після того, як передачу завершено	R/W		✓	✓	✓	✓
SM133	PORT2 отримання прапорець завершення біт	Цей біт встановлюється після того, як приймання завершено	R/W		✓	✓	✓	✓
SM134	PORT2 прапорець очікування біт	Цей біт встановлюється, коли порт неактивний	R		✓	✓	✓	✓
SM135	PORT2 MODBUS завершення зв'язку	Цей біт встановлюється, коли обмін даними завершено	R		✓	✓	✓	✓
SM136	PORT2 MODBUS помилка зв'язку	Цей біт встановлюється при виникненні помилки зв'язку	R		✓	✓	✓	✓

□ Примітка

SM130-SM134 — це прапорці для станів отримання, завершення та очікування в усіх протоколах зв'язку, що підтримуються PORT2.

14. N:N зв'язок

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM140	Помилка зв'язку станції 0 прапорець		R	✓	✓	✓	✓	✓
SM141	Помилка зв'язку станції 1 прапорець		R	✓	✓	✓	✓	✓
SM142	Помилка зв'язку станції 2 прапорець		R	✓	✓	✓	✓	✓
SM143	Помилка зв'язку станції 3 прапорець		R	✓	✓	✓	✓	✓
SM144	Помилка зв'язку станції 4 прапорець		R	✓	✓	✓	✓	✓
SM145	Помилка зв'язку станції 5 прапорець		R	✓	✓	✓	✓	✓

SM146	Помилка зв'язку станції 6 прапорець		R	✓	✓	✓	✓	✓
SM147	Помилка зв'язку станції 7 прапорець		R	✓	✓	✓	✓	✓
SM148	Помилка зв'язку станції 8 прапорець		R	✓	✓	✓	✓	✓
SM149	Помилка зв'язку станції 9 прапорець		R	✓	✓	✓	✓	✓
SM150	Помилка зв'язку станції 10 прапорець		R	✓	✓	✓	✓	✓
SM151	Помилка зв'язку станції 11 прапорець		R	✓	✓	✓	✓	✓
SM152	Помилка зв'язку станції 12 прапорець		R	✓	✓	✓	✓	✓
SM153	Помилка зв'язку станції 13 прапорець		R	✓	✓	✓	✓	✓
SM154	Помилка зв'язку станції 14 прапорець		R	✓	✓	✓	✓	✓
SM155	Помилка зв'язку станції 15 прапорець		R	✓	✓	✓	✓	✓
SM156	Помилка зв'язку станції 16 прапорець		R	✓	✓	✓	✓	✓
SM157	Помилка зв'язку станції 17 прапорець		R	✓	✓	✓	✓	✓
SM158	Помилка зв'язку станції 18 прапорець		R	✓	✓	✓	✓	✓
SM159	Помилка зв'язку станції 19 прапорець		R	✓	✓	✓	✓	✓
SM160	Помилка зв'язку станції 20 прапорець		R	✓	✓	✓	✓	✓
SM161	Помилка зв'язку станції 21 прапорець		R	✓	✓	✓	✓	✓
SM162	Помилка зв'язку станції 22 прапорець		R	✓	✓	✓	✓	✓
SM163	Помилка зв'язку станції 23 прапорець		R	✓	✓	✓	✓	✓
SM164	Помилка зв'язку станції 24 прапорець		R	✓	✓	✓	✓	✓
SM165	Помилка зв'язку станції 25 прапорець		R	✓	✓	✓	✓	✓
SM166	Помилка зв'язку станції 26 прапорець		R	✓	✓	✓	✓	✓
SM167	Помилка зв'язку станції 27 прапорець		R	✓	✓	✓	✓	✓
SM168	Помилка зв'язку станції 28 прапорець		R	✓	✓	✓	✓	✓
SM169	Помилка зв'язку станції 29 прапорець		R	✓	✓	✓	✓	✓
SM170	Помилка зв'язку станції 30 прапорець		R	✓	✓	✓	✓	✓
SM171	Помилка зв'язку станції 31 прапорець		R	✓	✓	✓	✓	✓

15. Помилка системної шини прапор

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM190	Біт прапорця помилки шини головного модуля	1. Цей біт скидається після правильного ввімкнення живлення 2. Цей біт скидається, коли PLC змінюється з STOP на RUN 3. Цей біт скидається, коли завантажується нова програма 4. Цей біт спричиняє STOP системи STOP	R	✓	✓	✓	✓	✓
SM191	Біт прапорця помилки шини загального модуля	1. Цей біт встановлюється, і система генерує сигнал тривоги, коли виникає загальна помилка роботи шини модуля 2. Цей біт скидається автоматично, коли (це) системна помилка видалена	R	✓	✓	✓	✓	✓
		1. Цей біт встановлюється, і система генерує сигнал тривоги, коли виникає						

SM192	Спеціальний модуль біт прапорця помилки шини	помилка роботи шини спеціального модуля 2. Цей біт скидається автоматично, коли (це) системна помилка видалена	R	✓	✓	✓	✓	✓

16. Прапорець помилки годинника реального часу

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM193	R/W помилка годинника реального часу clock error	Цей біт встановлюється у разі помилки годинника реального часу. Цей біт автоматично скидається, якщо системна помилка видалена	R		✓	✓	✓	✓

17. Увімкнути прапорець інтегрованого аналогового каналу

Адреса №	Відповідна адреса лічильника ні.	Функція	R/W	VC1S	VC1	VC2	VC3	VC2
SM200	C200	Коли SM2 __ є ON, його відповідний C2__ стає а зменшити лічильник Коли SM2 __ є OFF, його відповідний C2__ стає одним збільшити лічильник	R/W	✓	✓	✓	✓	✓
SM201	C201		R/W	✓	✓	✓	✓	✓
SM202	C202		R/W	✓	✓	✓	✓	✓
SM203	C203		R/W	✓	✓	✓	✓	✓
SM204	C204		R/W	✓	✓	✓	✓	✓
SM205	C205		R/W	✓	✓	✓	✓	✓
SM206	C206		R/W	✓	✓	✓	✓	✓
SM207	C207		R/W	✓	✓	✓	✓	✓
SM208	C208		R/W	✓	✓	✓	✓	✓
SM209	C209		R/W	✓	✓	✓	✓	✓
SM210	C210		R/W	✓	✓	✓	✓	✓
SM211	C211		R/W	✓	✓	✓	✓	✓
SM212	C212		R/W	✓	✓	✓	✓	✓
SM213	C213		R/W	✓	✓	✓	✓	✓
SM214	C214		R/W	✓	✓	✓	✓	✓
SM215	C215		R/W	✓	✓	✓	✓	✓
SM216	C216		R/W	✓	✓	✓	✓	✓
SM217	C217		R/W	✓	✓	✓	✓	✓
SM218	C218		R/W	✓	✓	✓	✓	✓
SM219	C219		R/W	✓	✓	✓	✓	✓
SM220	C220		R/W	✓	✓	✓	✓	✓
SM221	C221		R/W	✓	✓	✓	✓	✓
SM222	C222		R/W	✓	✓	✓	✓	✓
SM223	C223		R/W	✓	✓	✓	✓	✓
SM224	C224		R/W	✓	✓	✓	✓	✓
SM225	C225		R/W	✓	✓	✓	✓	✓
SM226	C226	Коли SM2 __ є ON, його відповідний C2__ стає а зменшити лічильник Коли SM2 __ є OFF, його відповідний C2__ стає одним збільшити лічильник	R/W	✓	✓	✓	✓	✓
SM227	C227		R/W	✓	✓	✓	✓	✓
SM228	C228		R/W	✓	✓	✓	✓	✓
SM229	C229		R/W	✓	✓	✓	✓	✓
SM230	C230		R/W	✓	✓	✓	✓	✓
SM231	C231		R/W	✓	✓	✓	✓	✓
SM232	C232		R/W	✓	✓	✓	✓	✓
SM233	C233		R/W	✓	✓	✓	✓	✓
SM234	C234		R/W	✓	✓	✓	✓	✓
SM235	C235		R/W	✓	✓	✓	✓	✓

18. Напрямок рахунку та моніторинг високошвидкісних лічильників

Диференція	Адреса ні.	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5

Дія									
Однофазний один точковий лічильник вхід	SM236	C236	ON та низький OFF з SM2 _ відповідають рахунку вгору та вниз лічильника відповідно. Коли C2 __ однофазного двонаправленого лічильника імпульсів та двофазного лічильника імпульсів знаходиться в	R/W	✓	✓	✓	✓	✓
	SM237	C237		R/W	✓	✓	✓	✓	✓
	SM238	C238		R/W	✓	✓	✓	✓	✓
	SM239	C239		R/W	✓	✓	✓	✓	✓
	SM240	C240		R/W	✓	✓	✓	✓	✓
	SM241	C241		R/W	✓	✓	✓	✓	✓

Диференціальна дія	Адреса ні.	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5
	SM242	C242	режим зниження, відповідний йому SM2 __ змінюється на ON. Коли він перебуває в режимі збільшення, його відповідне SM2 __ __ змінюється на OFFI.	R/W	✓	✓	✓	✓	✓
	SM243	C243		R/W	✓	✓	✓	✓	✓
	SM244	C244		R/W	✓	✓	✓	✓	✓
	SM245	C245		R/W	✓	✓	✓	✓	✓
	SM246	C246		R/W	✓	✓	✓	✓	✓
	SM247	C247		R/W	✓	✓	✓	✓	✓
Одно-фаза двонаправлений усі рахунки вхід	SM248	C248	Зареєструвати вміст ON та OFF для SM2 __ __ відповідають рахунку вгору та зниження лічильника відповідно	R	✓	✓	✓	✓	✓
	SM249	C249		R	✓	✓	✓	✓	✓
	SM250	C250		R	✓	✓	✓	✓	✓
	SM251	C251		R	✓	✓	✓	✓	✓
	SM252	C252		R	✓	✓	✓	✓	✓
	SM253	C253		R	✓	✓	✓	✓	✓
	SM254	C254		R	✓	✓	✓	✓	✓
	SM255	C255		R	✓	✓	✓	✓	✓
Дво-фаза рахувати вхід	SM256	C256		R	✓	✓	✓	✓	✓
	SM257	C257		R	✓	✓	✓	✓	✓
	SM258	C258		R	✓	✓	✓	✓	✓
	SM259	C259		R	✓	✓	✓	✓	✓
	SM260	C260		R	✓	✓	✓	✓	✓
	SM261	C261		R	✓	✓	✓	✓	✓
	SM262	C262		R	✓	✓	✓	✓	✓
	SM263	C263		R	✓	✓	✓	✓	✓

19. Високошвидкісний вихід та інструкція позиціонування

1) Прапорець кореляції Y0 біт

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM270	Імпульсний вихід Y0 зупинка інструкція	Коли цей біт ON, імпульс Y0 є вимкнено	R/W		✓	✓	✓	
SM271	Імпульсний вихід Y0 моніторинг (зайнято/готово)	Використовуйте для моніторингу стану Y0. Зайнято — ON, готово — OFF.	R/W		✓	✓	✓	
SM272	PWM вихід мікросекунд	Коли цей біт ON, PWM інструкція виходів Y0 у мікросекундах.	R/W		✓	✓		
SM273	Вихід імпульсу переривання приводу дійсний	Коли цей біт ON, програма переривання та підпрограма можуть викликати інструкцію PLSY Y0, а виклик у головній програмі буде безперервно та багаторазово приводитися в дію за допомогою енергії потік	R/W		✓	✓	✓	
SM274	Огинаюча виконується у циклі	Коли цей біт має значення ON, виконується цикл інструкцій PLS для	R/W		✓	✓	✓	

		Y0 повторно						
SM275	PLSV прогресивне частотне перетворення	Коли цей біт ON , частота команди Y0 PSLV змінюється поступово, забезпечуючи прискорення та функція сповільнення.	R/W		✓		✓	
SM276	Увімкнення очищення функція	Функція виходу сигналу CLR Y0 привода інструкція ZRN є дійсною.	R/W	✓	✓	✓	✓	
SM277	Елемент, визначений сигналом очищення, є дійсним	Y елемент Y (N), що відповідає значенню N у SD175, використовується для індикації сигналу скидання. Якщо не вказано інше, Y0 за замовчуванням є Y10, що застосовується до DSZR та ZRN.	R	✓	✓	✓	✓	
SM278	ZRN напрямом	Y0 застосовується до DSZR	R		✓	✓	✓	
SM279	FWD обмеження	Y0 застосовується до DSZR/DVIT	R		✓	✓	✓	

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM280	REV межа	Y0 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM281	Логічна інверсія the DOG сигнал	Y0 застосовується до DSZR	R		✓	✓	✓	
SM282	Логічна інверсія the нульовий сигнал	Y0 застосовується до DSZR	R		✓	✓	✓	
SM283	Логічна інверсія сигналу переривання	Y0 застосовується до DVIT, але не застосовується до входу переривання користувача інструкція	R/W		✓	✓		
SM284	позиціонування інструкція — це керування приводом	Y0 застосовується до DSZR/DVIT	R/W		✓	✓	✓	
SM285	Переривання користувача вхід інструкція	Y0 застосовується до DVIT	R/W		✓			

2) Прапорець кореляції Y1 біт

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM290	Імпульсний вихід Y1 зупинка інструкція	Коли цей біт ON, імпульс Y1 є вимкнено	R/W		✓	✓	✓	
SM291	Імпульсний вихід Y1 моніторинг (зайнято/готово)	Використовуйте для моніторингу стану Y1. Зайнято — ON, готово — OFF.	R/W		✓	✓	✓	
SM292	PWM вихід мікросекунд	Коли цей біт ON, інструкція PWM для Y1 видає на виході мікросекунди	R/W		✓	✓		
SM293	Вихід імпульсу переривання привода дійсний	Коли цей біт дорівнює ON, програма переривання та підпрограма можуть викликати інструкцію PLSY для Y1, а виклик у головній програмі буде безперервно та багаторазово керуватися за допомогою енергії потік	R/W		✓	✓	✓	
SM294	Огинаюча виконується у циклі	Коли цей біт дорівнює ON , то PLS цикл інструкцій Y1 виконується повторно	R/W		✓	✓	✓	
SM295	PLSV прогресивне частотне перетворення	Коли цей біт ON , частота команди Y1 PSLV змінюється поступово, забезпечуючи прискорення та функція сповільнення.	R/W		✓		✓	

SM296	Увімкнення очищення функція	Функція виходу сигналу CLR Y1 для інструкція ZRN є дійсною.	R/W	✓	✓	✓	✓	
SM297	Елемент, визначений сигналом очищення, є дійсним	Y елемент Y (N), що відповідає значенню N у SD195, використовується для індикації сигналу скидання. Якщо не вказано інше, Y1 за замовчуванням є Y11, що застосовується до DSZR та ZRN.	R	✓	✓	✓	✓	
SM298	ZRN напрямом	Y1 застосовується до DSZR	R		✓	✓	✓	
SM299	FWD обмеження	Y1 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM300	REV межа	Y1 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM301	Логічна інверсія the DOG сигнал	Y1 застосовується до DSZR	R		✓	✓	✓	
SM302	Логічна інверсія the нульовий сигнал	Y1 застосовується до DSZR	R		✓	✓	✓	
SM303	Логічна інверсія сигналу переривання	Y1 застосовується до DVIT, але не застосовується до входу переривання користувача інструкція	R/W		✓	✓		
SM304	позиціонування інструкція — це керування приводом	Y1 застосовується до DSZR/DVIT	R/W		✓	✓	✓	
SM305	Переривання користувача вхід інструкція	Y1 застосовується до DVIT	R/W		✓			

3) Прапорець кореляції Y2 біт

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM310	Імпульсний вихід Y2 зупинка інструкція	Коли цей біт ON, імпульс Y2 є вимкнено	R/W		✓	✓	✓	
SM311	Імпульсний вихід Y2 моніторинг (зайнято/готово)	Використовуйте для моніторингу стану Y2. Зайнято — ON, готово — OFF.	R/W		✓	✓	✓	
SM312	PWM вихід мікросекунд	Коли цей біт ON, інструкція PWM виходу Y2 у мікросекунди	R/W		✓	✓		
SM313	Вихід імпульсу переривання привода дійсний	Коли цей біт дорівнює ON, програма переривання та підпрограма можуть викликати інструкцію PLSY Y2, а виклик у головній програмі буде безперервно та багаторазово приводитися в дію за допомогою енергії потік	R/W		✓	✓	✓	
SM314	Огинаюча виконується у циклі	Коли цей біт ON, PLS цикл інструкцій Y2 виконується повторно	R/W		✓	✓	✓	
SM315	PLSV прогресивне частотне перетворення	Коли цей біт ON, частота команди Y2 PSLV змінюється поступово, забезпечуючи прискорення та функція сповільнення.	R/W		✓		✓	
SM316	Увімкнення очищення функція	Функція виходу сигналу CLR Y2 для інструкція ZRN є дійсною.	R/W	✓	✓	✓	✓	
SM317	Елемент, визначений сигналом очищення, є	Y елемент Y (N), що відповідає значенню N у SD215, використовується для індикації сигналу скидання. Якщо не вказано інше, Y2 за замовчуванням є Y12,	R	✓	✓	✓	✓	

	дійсним	що застосовується YдоY DSZR та ZRN.						
SM318	ZRN напрямок	Y2 застосовується до DSZR	R		✓	✓	✓	
SM319	FWD обмеження	Y2 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM320	REV межа	Y2 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM321	Логічна інверсія the DOG сигнал	Y2 застосовується до DSZR	R		✓	✓	✓	
SM322	Логічна інверсія the нульовий сигнал	Y2 застосовується до DSZR	R		✓	✓	✓	
SM323	Логічна інверсія сигналу переривання	Y2 застосовується до DVIT, але не застосовується до входу переривання користувача інструкція	R/W		✓	✓		
SM324	позиціонування інструкція — це керування приводом	Y2 застосовується до DSZR/DVIT	R/W		✓	✓	✓	
SM325	Переривання користувача вхід інструкція	Y2 застосовується до DVIT	R/W		✓			

4) Прапорець кореляції Y3 біт

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM330	Імпульсний вихід Y3 зупинка інструкція	Коли цей біт ON, імпульс Y3 є вимкнено	R/W		✓	✓	✓	
SM331	Імпульсний вихід Y3 моніторинг (зайнято/готово)	Використовуйте для моніторингу стану Y3. Зайнято — ON, готово — OFF.	R/W		✓	✓	✓	
SM332	PWM вихід мікросекунд	Коли цей біт ON, PWM інструкція виходів Y3 у мікросекундах.	R/W		✓	✓		
SM333	Вихід імпульсу переривання приводу дійсний	Коли цей біт ON, програма переривання та підпрограма можуть викликати інструкцію PLSY Y3, а виклик в основна програма буде безперервно	R/W		✓	✓	✓	

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
		і багаторазово приводиться в дію за допомогою енергії потік						
SM334	Огинаюча виконується у циклі	Коли цей біт дорівнює ON, то PLS цикл інструкцій Y3 виконується повторно	R/W		✓	✓	✓	
SM335	PLSV прогресивне частотне перетворення	Коли цей біт ON, частота команди Y3 PSLV змінюється поступово, забезпечуючи прискорення та функція сповільнення.	R/W		✓		✓	
SM336	Увімкнення очищення функція	Функція виходу сигналу CLR Y3 привода інструкція ZRN є дійсною.	R/W	✓	✓	✓	✓	
SM337	Елемент, визначений сигналом очищення, є дійсним	Y елемент Y (N), що відповідає значенню N у SD235, використовується для індикації сигналу очищення. Якщо не вказано інше, Y3 за замовчуванням є Y13, що застосовується до DSZR та ZRN.	R	✓	✓	✓	✓	
SM338	ZRN напрямок	Y3 застосовується до DSZR	R		✓	✓	✓	
SM339	FWD обмеження	Y3 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM340	REV межа	Y3 застосовується до DSZR/DVIT	R		✓	✓	✓	

SM341	Логічна інверсія the DOG сигнал	Y3 застосовується до DSZR	R		✓	✓	✓	
SM342	Логічна інверсія the нульовий сигнал	Y3 застосовується до DSZR	R		✓	✓	✓	
SM343	Логічна інверсія сигналу переривання	Y3 застосовується до DVIT, але не застосовується до входу переривання користувача інструкція	R/W		✓	✓		
SM344	позиціонування інструкція — це керування приводом	Y3 застосовується до DSZR/DVIT	R/W		✓	✓	✓	
SM345	Переривання користувача вхід інструкція	Y3 застосовується до DVIT	R/W		✓			

5) Прапорець кореляції Y4 біт

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM350	Імпульсний вихід Y4 зупинка інструкція	Коли цей біт ON, імпульс Y4 є вимкнено	R/W		✓	✓	✓	
SM351	Імпульсний вихід Y4 моніторинг (зайнято/готово)	Використовуйте для моніторингу стану Y4. Зайнято — ON, готово — OFF.	R/W		✓	✓	✓	
SM352	PWM вихід мікросекунд	Коли цей біт ON, інструкція PWM Y4 видає на вихід мікросекунди	R/W		✓	✓		
SM353	Вихід імпульсу переривання привода дійсний	Коли цей біт ON, програма переривання та підпрограма можуть викликати інструкцію PLSY Y4, а виклик у головній програмі буде постійно та багаторазово приводитися в дію за допомогою енергії потік	R/W		✓	✓	✓	
SM354	Огинаюча виконується у циклі	Коли цей біт дорівнює ON, виконується цикл інструкції PLS для Y4 повторно	R/W		✓	✓	✓	
SM355	PLSV прогресивне частотне перетворення	Коли цей біт ON, частота команди PSLV Y4 змінюється поступово, забезпечуючи прискорення та функція сповільнення.	R/W		✓		✓	
SM356	Увімкнення очищення функція	Функція виходу сигналу CLR Y4 для інструкція ZRN є дійсною.	R/W	✓	✓	✓	✓	
SM357	Елемент, визначений чистий сигнал є дійсним	Y елемент Y (N), що відповідає значенню N у SD255, використовується для індикації чіткий сигнал. Якщо не вказано, Y4 є	R	✓	✓	✓	✓	

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
		Y14 за замовчуванням, що застосовується до DSZR та ZRN.						
SM358	ZRN напрям	Y4 застосовується до DSZR	R		✓	✓	✓	
SM359	FWD обмеження	Y4 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM360	REV межа	Y4 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM361	Логічна інверсія the DOG сигнал	Y4 застосовується до DSZR	R		✓	✓	✓	
SM362	Логічна інверсія the нульовий сигнал	Y4 застосовується до DSZR	R		✓	✓	✓	
	Логічна інверсія	Y4 застосовується до DVIT, але не						

SM363	сигналу переривання	застосовується до входу переривання користувача інструкція	R/W		✓	✓		
SM364	позиціонування інструкція — це керування приводом	Y4 застосовується до DSZR/DVIT	R/W		✓	✓	✓	
SM365	Переривання користувача вхід інструкція	Y4 застосовується до DVIT	R/W		✓			

6) Y5 Прапорець кореляції біт

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM370	Імпульсний вихід Y5 зупинка інструкція	Коли цей біт ON, імпульс Y5 є вимкнено	R/W		✓	✓	✓	
SM371	Імпульсний вихід Y5 моніторинг (зайнятий/готовий)	Використовуйте для моніторингу стану Y5. Зайнято — ON, готово — OFF.	R/W		✓	✓	✓	
SM372	PWM вихід мікросекунд	Коли цей біт ON, PWM інструкція виходів Y5 у мікросекундах.	R/W		✓	✓		
SM373	Вихід імпульсу переривання привода дійсний	Коли цей біт має значення ON, програма переривання та підпрограма можуть викликати інструкцію PLSY для Y5, а виклик у головній програмі буде безперервно та багаторазово приводитися в дію за допомогою енергії потік	R/W		✓	✓	✓	
SM374	Огинаюча виконується у циклі	Коли цей біт дорівнює ON, виконується інструкція циклу PLS для Y5 повторно	R/W		✓	✓	✓	
SM375	PLSV прогресивне частотне перетворення	Коли цей біт ON, Y5 PSLV частота завдання змінюється поступово, забезпечуючи прискорення та функція сповільнення.	R/W		✓		✓	
SM376	Увімкнення очищення функція	Функція виходу сигналу CLR Y5 для інструкція ZRN є дійсною.	R/W	✓	✓	✓	✓	
SM377	Елемент, визначений сигналом очищення, є дійсним	Y елемент Y (N), що відповідає значенню N у SD275, використовується для індикації сигналу скидання. Якщо не вказано інше, Y5 за замовчуванням є Y15, що застосовується до DSZR та ZRN.	R	✓	✓	✓	✓	
SM378	ZRN напрямом	Y5 застосовується до DSZR	R		✓	✓	✓	
SM379	FWD обмеження	Y5 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM380	REV межа	Y5 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM381	Логічна інверсія the DOG сигнал	Y5 застосовується до DSZR	R		✓	✓	✓	
SM382	Логічна інверсія the нульовий сигнал	Y5 застосовується до DSZR	R		✓	✓	✓	
SM383	Логічна інверсія сигналу переривання	Y5 застосовується до DVIT, але не застосовується до входу переривання користувача інструкція	R/W		✓	✓		
SM384	позиціонування інструкція — це керування приводом	Y5 застосовується до DSZR/DVIT	R/W		✓	✓	✓	

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM385	Переривання користувача вхід інструкція	Y5 застосовується до DVIT	R/W		✓			

7) Прапорець кореляції Y6 біт

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM390	Імпульсний вихід Y6 зупинка інструкція	Коли цей біт ON, імпульс Y6 є вимкнено	R/W		✓	✓	✓	
SM391	Імпульсний вихід Y6 моніторинг (зайнято/готово)	Використовуйте для моніторингу стану Y6. Зайнято — ON, готово — OFF.	R/W		✓	✓	✓	
SM392	PWM вихід мікросекунд	Коли цей біт ON, інструкція PWM Y6 виводить у мікросекунди	R/W		✓	✓		
SM393	Вихід імпульсу переривання привода дійсний	Коли цей біт ON, програма переривання та підпрограма можуть викликати інструкцію PLSY Y6, а виклик у головній програмі буде безперервно та багаторазово приводитися в дію за допомогою енергії потік	R/W		✓	✓	✓	
SM394	Огинаюча виконується у циклі	Коли цей біт ON, PLS цикл інструкцій Y6 виконується багаторазово	R/W		✓	✓	✓	
SM395	PLSV прогресивне частотне перетворення	Коли цей біт ON, частота команди Y6 PSLV змінюється поступово, забезпечуючи прискорення та функція сповільнення.	R/W		✓		✓	
SM396	Увімкнення очищення функція	Функція виходу сигналу CLR Y6 для інструкція ZRN є дійсною.	R/W	✓	✓	✓	✓	
SM397	Елемент, визначений сигналом очищення, є дійсним	Y елемент Y (N), що відповідає значенню N у SD295, використовується для індикації сигналу очищення. Якщо не вказано інше, Y6 за замовчуванням є Y16, що застосовується YдоY DSZR та ZRN.	R	✓	✓	✓	✓	
SM398	ZRN напрямом	Y6 застосовується до DSZR	R		✓	✓	✓	
SM399	FWD обмеження	Y6 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM400	REV межа	Y6 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM401	Логічна інверсія the DOG сигнал	Y6 застосовується до DSZR	R		✓	✓	✓	
SM402	Логічна інверсія the нульовий сигнал	Y6 застосовується до DSZR	R		✓	✓	✓	
SM403	Логічна інверсія сигналу переривання	Y6 застосовується до DVIT, але не застосовується до входу переривання користувача інструкція	R/W		✓	✓		
SM404	позиціонування інструкція — це керування приводом	Y6 застосовується до DSZR/DVIT	R/W		✓	✓	✓	
SM405	Переривання користувача вхід інструкція	Y6 застосовується до DVIT	R/W		✓			

8) Прапорець кореляції Y7 біт

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM410	Імпульсний вихід Y7 зупинка інструкція	Коли цей біт ON, імпульс Y7 є вимкнено	R/W		✓	✓	✓	
SM411	Імпульсний вихід Y7 моніторинг (зайнято/готово)	Використовуйте для моніторингу стану Y7. Зайнято — ON, готово — OFF.	R/W		✓	✓	✓	
SM412	PWM вихід мікросекунд	Коли цей біт ON, PWM інструкція виходів Y7 у мікросекундах.	R/W		✓	✓		

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM413	Вихід імпульсу переривання привода дійсний	Коли цей біт ON, програма переривання та підпрограма можуть викликати інструкцію PLSY Y7, а виклик у головній програмі буде постійно та багаторазово активуватися з енергією потік	R/W		✓	✓	✓	
SM414	Огинаюча виконується у циклі	Коли цей біт ON, PLS цикл інструкцій Y7 виконується повторно	R/W		✓	✓	✓	
SM415	PLSV прогресивне частотне перетворення	Коли цей біт ON, Y7 PSLV частота завдання змінюється поступово, забезпечуючи прискорення та функція сповільнення.	R/W		✓		✓	
SM416	Увімкнення очищення функція	Функція виходу сигналу CLR Y7 для інструкція ZRN є дійсною.	R/W	✓	✓	✓	✓	
SM417	Елемент, визначений сигналом очищення, є дійсним	Y елемент Y (N), що відповідає значенню N у SD315, використовується для індикації сигналу скидання. Якщо не вказано інше, Y7 за замовчуванням є Y17, що застосовується до DSZR та ZRN.	R	✓	✓	✓	✓	
SM418	ZRN напрямом	Y7 застосовується до DSZR	R		✓	✓	✓	
SM419	FWD обмеження	Y7 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM420	REV межа	Y7 застосовується до DSZR/DVIT	R		✓	✓	✓	
SM421	Логічна інверсія the DOG сигнал	Y7 застосовується до DSZR	R		✓	✓	✓	
SM422	Логічна інверсія the нульовий сигнал	Y7 застосовується до DSZR	R		✓	✓	✓	
SM423	Логічна інверсія сигналу переривання	Y7 застосовується до DVIT, але не застосовується до входу переривання користувача інструкція	R/W		✓	✓		
SM424	позиціонування інструкція — це керування приводом	Y7 застосовується до DSZR/DVIT	R/W		✓	✓	✓	
SM425	Переривання користувача вхід інструкція	Y7 застосовується до DVIT	R/W		✓			

20. Вихід за часом інструкція

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM430	Вихід таймера годинника 1	Для використання згідно з DUTY інструкцією	R/W			✓	✓	✓
SM431	Вихід таймера годинника 2	Для використання згідно з DUTY інструкцією	R/W			✓	✓	✓
SM432	Вихід таймера годинника 3	Для використання згідно з DUTY інструкцією	R/W			✓	✓	✓
SM433	Вихід таймера	Для використання згідно з DUTY	R/W			✓	✓	✓

	годинника 4	інструкцією						
SM434	Вихід таймера годинника 5	Для використання згідно з DUTY інструкцією	R/W			✓	✓	✓

21. Сигнал тривоги

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM435	Сигнал тривоги є дійсним	Після встановлення SM400 у ON, цей після SM401 та SD401 робота	R/W			✓	✓	✓
SM436	Дія сигнальної тривоги	Встановлення SM401 у ON, коли є будь-який дія у стані S900-S999	R/W			✓	✓	✓

22. CANopen інструкція

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SM440	CANopen інструкція завершено		R/ W			✓	✓	✓
SM441	CANopen інструкція є помилковим		R/W			✓	✓	✓

SM442	CANopen інструкція є в роботі		R			✓	✓	✓

Додаток В Спеціальний регістр даних регістр

□ Примітка

1. Зарезервовані елементи SD та SM не вказані в таблиці. Зарезервовані елементи SD за замовчуванням доступні лише для читання.

1. PLC робочий стан дані

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD00	PLC тип	100 вказує на VC1S 112 вказує на VC1, 40 вказує на VC2, 30 вказує на VC3 50 вказує на VC5	R	✓	✓	✓	✓	✓	
SD01	Версія №	Наприклад, 100 вказує на 1.00	R	✓	✓	✓	✓	✓	
SD02	Користувач ємність програми	Наприклад, 8 вказує на програму з 8 тис. кроків	R	✓	✓	✓	✓	✓	
SD03	Помилка системи код	Збереження помилки системи, що виникла коди	R	✓	✓	✓	✓	✓	
SD04	Акумулятор значен ня напруги	Приймаючи 0.1 V за одиницю, а 3.6V вказує на 36	R		✓	✓	✓	✓	
SD07	Номер подовження подовження I/O модулі		R	✓	✓	✓	✓	✓	
SD08	Номер особливого значення		R	✓	✓	✓	✓	✓	

	модулі								
SD09	Налаштування точок вводу для керування роботою. Десяткове (X0 відображається як 0, а X10 відображається як 8. Максимум: 15) (Налаштовується через системний блок лише)		R	✓	✓	✓	✓		✓
SD10	Кількість головних модулів I/O точки	MSB: вхід; LSB: вихід	R	✓	✓	✓	✓	✓	
SD11	Кількість модулів розширення I/O точки	MSB: вхід; LSB: вихід	R	✓	✓	✓	✓	✓	
SD12	Кількість аналогових входів головного модуля I/O точки	MSB: вхід; LSB: вихід	R	✓		✓	✓	✓	

2. Код помилки експлуатації FIFO зона

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD20	Зарезервована помилка операції код 0	У порядку надходження зарезервовано п'ять останніх кодів помилок роботи. SD20 завжди зберігає останні коди помилок.	R	✓	✓	✓	✓	✓	
SD21	Зарезервовано код помилки роботи 1		R	✓	✓	✓	✓	✓	
SD22	Зарезервована помилка операції код 2		R	✓	✓	✓	✓	✓	
SD23	Зарезервована помилка операції код 3		R	✓	✓	✓	✓	✓	
SD24	Зарезервовано код помилки роботи 4		R	✓	✓	✓	✓	✓	

3. FROM/TO помилка

Адреса	Ім'я	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD25	Нумерація спеціального модуля неправильна .	R			✓	✓	✓	Початковий значення: 255
SD26	Нумерація чипа I/O неправильна (починається з 0) коли I/O оновлюється	R			✓	✓	✓	Початковий значення: 255

4. Сканувати час

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD30	Струм сканувати значення	Поточний час сканування (одиниця: 0.1ms)	R	✓	✓	✓	✓	✓	
SD31	Мін. сканування час	Мін. час сканування (одиниця: 0.1ms)	R	✓	✓	✓	✓	✓	

SD32	Макс. сканування час	Макс. час сканування (одиниця: 0.1ms)	R	✓	✓	✓	✓	✓	
SD33	Встановити значення часу сканування константи	Початкове значення: 0ms (налаштовується лише через системний блок), одиниця виміру: 1 мс. Коли час постійного циклу сканування довший за встановлене значення тайм-ауту моніторингу користувача, виникає аварійний сигнал тайм-ауту програми користувача. Коли цикл сканування програми користувача довший за час постійного сканування, режим постійного сканування цього циклу автоматично стає недійсним і аварійний сигнал не виникає. SD33 розглядається як 1000 мс, коли він є встановити більше ніж 1000 мс.	R	✓	✓	✓	✓	✓	0–1000мс
SD34	Встановити значення параметра тайм-аут програми	Початкове значення: 200 мс (налаштовується лише через системний блок). SD34 вважається рівним 100, якщо встановлено значення менше 100. SD34 вважається 1000, коли його встановлено менше ніж 1000.	R	✓	✓	✓	✓	✓	100–1000мс

□ Примітка

- Допуск похибки SD30, SD31 та SD32 становить 1ms.
- Рекомендовано встановлювати час виконання користувацької програми (SD34) принаймні на 5ms більшим за постійний час циклу опитування (SD33). В іншому разі, коли значення SD33 наближається до SD34, система схильна видавати помилку перевищення часу виконання користувацької програми через вплив роботи системи та користувацької програми.

5. Константа фільтрації вхідного сигналу налаштування

Адреса	Ім'я	Дія та функція	R/W	VC1S	VC1	VC2	VC3	VC5
SD35	Фільтрація вхідного сигналу константа	(налаштовується лише через систему блок)	R/W	✓	✓			
SD36	Фільтрація константа	(налаштовується лише через систему блок)	R/W	✓	✓			
SD37	Фільтрація вхідного сигналу константа	(налаштовується лише через систему блок)	R/W	✓	✓			
SD38	Фільтрація вхідного сигналу константа	(налаштовується лише через систему блок)	R/W	✓	✓			
SD39	Фільтрація константа	(налаштовується лише через систему блок)	R/W	✓	✓			
SD40	Фільтрація вхідного сигналу константа	(налаштовується лише через систему блок)	R/W	✓	✓			
SD41	Фільтрування константа	(налаштовується лише через систему блок)	R/W	✓	✓			
SD42	Фільтрація вхідного сигналу	(налаштовується лише через систему блок)	R/W	✓	✓			

константа

6. Цикл переривання за таймером cycle

Адреса	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD47	Цикл переривання за таймером 0	Той переривання є не спрацьовує, коли значення є не в межах 1–32767	R/W	✓	✓	✓	✓	✓	1–32767 мс
SD48	Цикл переривання за таймером 1	Той переривання є не спрацьовує, коли значення не знаходиться в межах 1–32767	R/W	✓	✓	✓	✓	✓	1–32767 мс
SD49	Цикл переривання за таймером 2	Той переривання є не спрацьовує, коли значення не знаходиться в межах 1–32767	R/W	✓	✓	✓	✓	✓	1–32767 мс

Помилка $\pm 1\text{ms}$ може виникнути, коли система обробляє переривання користувача за таймером. Для забезпечення нормальної роботи переривання за таймером рекомендується встановити цикл переривань за таймером більшим або рівним 5ms.

7. Годинник реального часу clock

Адреса	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD60	Рік	Для годинника реального часу	R	✓	✓	✓	✓	✓	2000–2099
SD61	Місяць	Для годинника реального часу	R	✓	✓	✓	✓	✓	1–12 місяці
SD62	День	Для годинника реального часу	R	✓	✓	✓	✓	✓	1–31 день
SD63	Година	Для годинника реального часу	R	✓	✓	✓	✓	✓	0–23 години
SD64	Хвилиноа	Для годинника реального часу	R	✓	✓	✓	✓	✓	0–59 хвилин
SD65	Другий	Для годинника реального часу	R	✓	✓	✓	✓	✓	0–59 секунд
SD66	Тиждень	Для годинника реального часу	R	✓	✓	✓	✓	✓	0 (неділя)–6 (Субота)

Ви можете встановити ці елементи лише за допомогою інструкції TWR або через головний комп'ютер

8. Використання DHSP та DHST інструкцій

Адреса	Ім'я	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD86	Порівняння таблиць MSB та DHSP вихід дані	R/W	✓	✓	✓	✓	✓	
SD87	Порівняння таблиць LSB та DHSP вихід дані	R/W	✓	✓	✓	✓	✓	
SD88	MSB з DHST або DHSP таблиця порівняльні дані	R/W	✓	✓	✓	✓	✓	
SD89	LSB з DHST або DHSP таблиця порівняльні дані	R/W	✓	✓	✓	✓	✓	
SD90	Запис № таблиці, що виконується	R/W	✓	✓	✓	✓	✓	

9. Керування та стан приймання через вільний порт (PORT0)

Адреса	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
		b2, b1, b0 000: швидкість передачі 38 400 бод 001: швидкість передачі 19 200 бод 010: швидкість							
	SD100.0–SD100.2								

SD110	Вільний порт 0 режим стан слово	Швидкість передачі baud оцінити	передачі 9600 бод 011: швидкість передачі 4800 бод 100: швидкість передачі даних 2400 бод rate 101: швидкість передачі 1200 бод rate 110: швидкість передачі 57 600 бод 111: швидкість передачі даних 115 200 бод rate	R	✓	✓	✓	✓	✓		
		SD100.3 Зупинка біта	0: 1 стоп біт 1: 2 стопові біти								
		SD100.4 Пере вірка парн ості	0: парність; 1: непарна парність								
		SD100.5 Парність перевірити увімкнення	0: без перевірки парності; 1: перевірка парності								
		SD100.6 Символ біт даних	Біт даних кожного символу 0: 8-бітний символ 1: 7-бітний символ								
		SD100.7 Вільний-порт отримання пуск символ режим	1: має специфічний пуск символ 0: без специфічного пуску символ								
		SD100.8 Вільний-порт отримання кінець символ режим	1: має специфічний кінець символ 0: без специфічного кінця символ								
		SD100.9 Вільний-порт міжсимвольний символ тайм-аут увімкнення	1: має дійсний міжсимвольний таймаут 0: не дійсний міжсимвольний тайм-аут								
		SD100.10 Вільний-порт міжрамковий тайм-аут увімкнення	1: має тайм-аут між кадрами 0: немає тайм-ауту між кадрами								
		SD100.11	Зарезервовано								
		SD100.12 MSB/LSB дійсний	0: LSB слова елемента є дійсним 1: MSB/LSB того елемент слова є								

		дійсним							
		SD100.13 – SD100.15	Зарезервовано						
SD101	Початок символу		R/W	✓	✓	✓	✓	✓	✓
SD102	Кінцевий символ		R/W	✓	✓	✓	✓	✓	✓

Адреса	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD103	Тайм-аут між символами	За замовчуванням: 0 мс (тайм-аут між символами) є пропущено)	R/W	✓	✓	✓	✓	✓	✓
SD104	Тайм-аут кадру	За замовчуванням: 0 мс (тайм-аут кадру становить пропущено)	R/W	✓	✓	✓	✓	✓	✓
SD105	Код повідомлення про завершення отримання	Біт 0: встановлюється, коли прийом завершено Біт 1: встановлюється, коли вказане кінцеве слово отримано Біт 2: встановлюється при отриманні макс. кількості символів Біт 3: встановлюється при міжсимвольному тайм-ауті Біт 4: встановлюється при тайм-ауті кадру Біт 5: встановлюється при помилці перевірки парності Біти 6-15: зарезервовано	R	✓	✓	✓	✓	✓	✓
SD106	Наразі отримано символи		R	✓	✓	✓	✓	✓	✓
SD107	Загальна кількість наразі отриманих символів		R	✓	✓	✓	✓	✓	✓
SD108	Наразі передані символи		R	✓	✓	✓	✓	✓	✓
SD109	PORT 0 Станція номер налаштування		R/W	✓	✓	✓	✓	✓	✓
SD110	PORT 0 налаштування максимального часу очікування (після відправлення та перед отриманням) ESBUS додатковий затримка		R/W	✓	✓	✓	✓	✓	✓
SD111	PORT 0 повторна спроба		R/W	✓	✓	✓	✓	✓	✓

	номер								
SD112	N:N режим оновлення мережі (зарезервовано для PORT 0)		R/W	✓	✓	✓	✓	✓	✓
SD113	Код помилки для Modbus-майстра станція (PORT 0)		R	✓	✓	✓	✓	✓	✓

10. Керування прийомом через вільний порт та стан (PORT1)

Адреса	Ім'я		Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD120	Вільний порт 1 словове стану привода	SD120.0 – SD120.2 Швидкість передачі даних порту	b2, b1, b0 000: швидкість передачі 38 400 бод 001: швидкість передачі 19 200 бод 010: швидкість передачі 9600 бод 011: швидкість передачі 4800 бод 100: швидкість передачі даних 2400 бод rate 101: швидкість передачі 1200 бод rate 110: швидкість передачі 57 600 бод 111: 115 200 бод оцінити	R/W	✓	✓	✓	✓	✓	

Адреса	Ім'я		Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
		SD120.3 Зупинка біта	0: 1 стоп біт 1: 2 стопові біти							
		SD120.4 Перевірка парності	0: парність parity 1: непарний контроль							
		SD120.5 Увімкнення перевірки парності	0: без перевірки парності 1: з перевіркою парності							
		SD120.6 Дані символів біт	Біт даних кожного символу 0: 8-бітний символ 1: 7-бітний символ							
		SD120.7 Режим прийому стартового символу вільного порту	1: має специфічний початковий символ 0: без специфічного початкового символу							
		SD120.8								

		Режим завершального символу прийому вільного порту	1: має специфічний кінцевий символ 0: немає визначеного кінцевого символу							
		SD120.9 Міжсимвольний тайм-аут вільного порту увімкнення	1: має дійсний тайм-аут між символами 0: немає дійсного тайм-ауту між символами							
		SD120.10 Тайм-аут між кадрами вільного порту увімкнення	1: має міжфреймовий тайм-аут 0: немає міжфреймового тайм-ауту							
		SD120.11	Зарезервовано							
		SD120.12 MSB/LSB в дійсний	0: LSB слова елемента є дійсним 1: MSB/LSB цього елемент слова є дійсним							
		SD120.13–SD120.15	Зарезервовано							
SD121	Початок символу			R/W	✓	✓	✓	✓	✓	
SD122	Кінцевий символ			R/W	✓	✓	✓	✓	✓	
SD123	Тайм-аут між символами		За замовчуванням: 0 мс (тайм-аут між символами пропущено)	R/W	✓	✓	✓	✓	✓	
SD124	Час очікування кадру		За замовчуванням: 0 мс (тайм-аут кадру пропущено)	R/W	✓	✓	✓	✓	✓	

Адреса	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD125	Код повідомлення про завершення отримання	Біт 0: встановлюється, коли прийом завершується Біт 1: встановлюється, коли вказане кінцеве слово отримано Біт 2: встановлюється при отриманні макс. кількості символів Біт 3: встановлюється при міжсимвольному тайм-ауті Біт 4:	R	✓	✓	✓	✓	✓	

		встановлюється при тайм-ауті кадру Біт 5: встановлюється після того, як помилка перевірки парності Біти 6-15: зарезервовано							
SD126	Наразі отримано символи		R	✓	✓	✓	✓	✓	
SD127	Загальна кількість наразі отриманих символи		R	✓	✓	✓	✓	✓	
SD128	Наразі передані символи		R	✓	✓	✓	✓	✓	
SD129	PORT 1 Станція номер налаштування		RW	✓	✓	✓	✓	✓	✓
SD130	PORT1 максимальне налаштування часу очікування (після відправлення та перед отриманням) ESBUS додатковий затримка		RW	✓	✓	✓	✓	✓	✓
SD131	PORT 1 повторна спроба номер		RW	✓	✓	✓	✓	✓	✓
SD132	N:N режим оновлення мережі (зарезервовано для PORT 1)		RW	✓	✓	✓	✓	✓	✓
SD133	Код помилки для Modbus-майстра станція (PORT 1)		R	✓	✓	✓	✓	✓	✓
SD134	Код помилки для Modbus майстер станція (PORT 1)	Значення дорівнює 0, якщо помилка відсутня в зв'язок	R	✓	✓	✓	✓	✓	✓

11. Керування прийомом через вільний порт та стан (PORT2)

Адреса	Ім'я		Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD140	Вільний-порт 2	словосполука	SD140.0 – Швидкість передачі даних	RW	✓	✓	✓	✓	✓	
			b2, b1, b0 000: швидкість передачі 38 400 бод 001: швидкість передачі 19 200 бод 010: швидкість передачі 9600 бод 011: швидкість передачі 4800 бод 100: швидкість передачі даних 2400							

	прив оду	порту	бод rate 101: швидкість передачі 1200 бод rate 110: швидкість передачі 57 600 бод 111: 115 200 бод оцінити							

Адреса	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
	SD140.3 Зупинка біта	0: 1 стоп біт 1: 2 стопові біти							
	SD140.4 Пере вірка парн ості	0: парність parity 1: непарний контроль							
	SD140.5 Увімкне ння перевірк и на парність	0: без перевірки парності 1: перевірка парності							
	SD140.6 Дані символів біт	Біт даних кожного символу 0: 8-бітний символ 1: 7-бітний символ							
	SD140.7 Режим прийому стартово го символу вільного порту	1: має специфічний початковий символ 0: без специфічного початкового символу							
	SD140.8 Режим кінцевого символу прийому вільного порту mode	1: має специфічний кінцевий символ 0: немає визначеного кінцевого символу							
	SD140.9 Міксимв ольний тайм-аут вільного порту увімкненн я	1: має дійсний тайм- аут між символами 0: немає дійсного тайм-ауту між символами							
	SD140. 10 Тайм-аут між кадрами вільного порту е увімкненн я	1: має тайм-аут між кадрами 0: немає тайм- ауту між кадрами							
	SD140.1 1	Зарезервовано							

		SD140.1 2 MSB/LS В дійсний	0: LSB слова елемента є дійсним 1: MSB/LSB того елемент слова є дійсним							
		SD140.1 3– SD140.1 5	Зарезервовано							
SD141	Початок символу			R/W	✓	✓	✓	✓	✓	
SD142	Кінцевий символ			R/W	✓	✓	✓	✓	✓	
SD143	Тайм-аут між символами		За замовчуванням: 0 мс (тайм-аут між символами пропущено)	R/W	✓	✓	✓	✓	✓	
SD144	Час очікування кадру		За замовчуванням: 0 мс (тайм-аут кадру пропущено)	R/W	✓	✓	✓	✓	✓	

Адреса	Ім'я	Зареєструвати вміст	R/W	VC1S	VC1	VC2	VC3	VC5	Діапазон
SD145	Код повідомлення про завершення отримання	Біт 0: встановлюється, коли прийом завершено Біт 1: встановлюється, коли вказане кінцеве слово отримано Біт 2: встановлюється при отриманні макс. кількості символів Біт 3: встановлюється при міжсимвольному тайм-ауті Біт 4: встановлюється при тайм-ауті кадру Біт 5: встановлюється після того, як помилка перевірки парності Біти 6-15: зарезервовано	R	✓	✓	✓	✓	✓	
SD146	Наразі отриман о символи		R	✓	✓	✓	✓	✓	
SD147	Загальна кількість наразі отриманих символи		R	✓	✓	✓	✓	✓	
SD148	Наразі передані символи		R	✓	✓	✓	✓	✓	
SD149	PORT 2 Станція номер налаштування		R/W	✓	✓	✓	✓	✓	✓
SD150	PORT 2 максимальне налаштування часу очікування (після відправлення та		R/W	✓	✓	✓	✓	✓	✓

	перед отриманням) ECBUS додаткове затримка								
SD151	PORT 2 повторна спроба номер		RW	✓	✓	✓	✓	✓	✓
SD152	N:N режим оновлення мережі (зарезервовано для PORT 2)		RW	✓	✓	✓	✓	✓	✓
SD153	Код помилки для Modbus-майстра станція (PORT 2)		R	✓	✓	✓	✓	✓	✓
SD154	Код помилки для Modbus майстер станція (PORT 2)	Значення дорівнює 0, якщо помилка відсутня в зв'язок	R	✓	✓	✓	✓	✓	✓

12. Високошвидкісний імпульсний вихід та позиціонування

1) Кореляція Y0 регістр

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD160	Сумарна кількість імпульсів Y0 на виході.	0	R/W	✓	✓	✓	✓	
SD161								
SD162	Поточна позиція інструкції позиціонування виходу Y0.	0	R/W	✓	✓	✓	✓	
SD163								
SD164		0	R	✓	✓	✓	✓	✓

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD165	Поточна частота виходу Y0 виходу інструкція з позиціонування.							
SD166	Макс. швидкість, при якій ZRN, PLSV, DRVI, DRVA, DSZR та DVIT інструкції виконуються (10- 100000) (Y0)	100000	R/W	✓	✓	✓	✓	
SD167								
SD168	Базова швидкість, коли інструкції ZRN, PLSV, DRVI, DRVA, DSZR та DVIT є виконано (1/10 від макс. швидкості) (Y0)	5000	R/W	✓	✓	✓	✓	
SD169	Час розгону/гальмування для прискорення від базової швидкості (SD204) до макс. швидкості (SD202, SD203), коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються (Y0) (50ms — 5000ms)	1000	R/W	✓	✓	✓	✓	
SD170	Час сповільнення для зниження швидкості від макс. швидкості до базової швидкості, коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються Y0 (50 мс —	1000	R/W		✓			

	5000 мс)							
SD171	Швидкість повзання Y0 застосовується до DSZR	1000	R/W	✓	✓	✓	✓	
SD172	Швидкість повернення до нуля Y0 застосовується до DSZR	50000	R/W	✓	✓	✓	✓	
SD173								
SD174	Кількість сегментів, на які PLS інструкція виводу наразі виконується (Y0 застосовно)	0	R/W	✓	✓	✓	✓	
SD175	Y0 визначено як програмний елемент якого чіткий сигнал	0	R/W	✓	✓	✓	✓	
SD176	DVIT сигнал переривання програмний елемент специфікація (застосовна Y0)	0	R/W	✓	✓	✓	✓	

2) Кореляція Y1 регістр

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD180	Сумарна кількість імпульсів виходу Y1.	0	R/W	✓	✓	✓	✓	
SD181								
SD182	Поточна позиція інструкції позиціонування виходу Y1.	0	R/W	✓	✓	✓	✓	
SD183								
SD184	Поточна частота інструкції позиціонування виходу Y1.	0	R	✓	✓	✓	✓	✓
SD185								
SD186	Макс. швидкість, при якій ZRN, PLSV, DRVI, DRVA, DSZR та DVIT	100000	R/W	✓	✓	✓	✓	
SD187								
SD188	Базова швидкість, коли інструкції ZRN, PLSV, DRVI, DRVA, DSZR та DVIT є виконано (1/10 від макс. швидкості) (Y1)	5000	R/W	✓	✓	✓	✓	
SD189	Час розгону/гальмування для прискорення від базової швидкості (SD204) до макс. швидкості (SD202, SD203), коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються (Y1) (50ms — 5000ms)	1000	R/W	✓	✓	✓	✓	

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD190	Час сповільнення для зниження швидкості від макс. до базової, коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються Y1 (50 мс — 5000 мс)	1000	R/W		✓			
SD191	Швидкість повзання Y1 застосовується до DSZR	1000	R/W	✓	✓	✓	✓	
SD192	Швидкість повернення до нуля Y1 застосовна до DSZR	50000	R/W	✓	✓	✓	✓	
SD193								
SD194	Кількість сегментів, на які PLS інструкція виводу наразі виконується (Y1 застосовно)	0	R/W	✓	✓	✓	✓	
SD195	Y1 визначено як програмний елемент якого чіткий сигнал	0	R/W	✓	✓	✓	✓	

SD196	DVIT сигнал переривання програмний елемент специфікація (застосовна Y1)	0	R/W	✓	✓	✓	✓	

3) Кореляція Y2 регістр

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD200	Сумарна кількість імпульсів Y2 на виході.	0	R/W	✓	✓	✓	✓	
SD201								
SD202	Поточна позиція інструкції позиціонування виходу Y2.	0	R/W	✓	✓	✓	✓	
SD203								
SD204	Поточна частота інструкції позиціонування виходу Y2.	0	R	✓	✓	✓	✓	✓
SD205								
SD206	Макс. швидкість, при якій ZRN, PLSV, DRVI, DRVA, DSZR та DVIT інструкції виконуються (10- 100000) (Y2)	100000	R/W	✓	✓	✓	✓	
SD207								
SD208	Базова швидкість при ZRN, PLSV, DRVI, Інструкції DRVA, DSZR та DVIT виконуються (1/10 від макс. швидкості) (Y2)	5000	R/W	✓	✓	✓	✓	
SD209	Час розгону/гальмування для прискорення від базової швидкості (SD204) до макс. швидкості (SD202, SD203), коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються (Y2) (50ms — 5000ms)	1000	R/W	✓	✓	✓	✓	
SD210	Час сповільнення для зниження швидкості від макс. швидкості до базової швидкості, коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються Y2 (50 мс — 5000 мс)	1000	R/W		✓			
SD211	Швидкість повзання Y2 застосовується до DSZR	1000	R/W	✓	✓	✓	✓	
SD212	Швидкість повернення до нуля Y2 застосовується до DSZR	50000	R/W	✓	✓	✓	✓	
SD213								
SD214	Кількість сегментів, для яких інструкція виводу PLS наразі виконується (Y2 застосовно)	0	R/W	✓	✓	✓	✓	
SD215	Y2 визначено як програмний елемент якого чіткий сигнал	0	R/W	✓	✓	✓	✓	

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD216	DVIT сигнал переривання програмний елемент специфікація (застосовна Y2)	0	R/W	✓	✓	✓	✓	

4) Кореляція Y3 регістр

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD220								

SD221	Сумарна кількість імпульсів виходу Y3.	0	R/W	✓	✓	✓	✓	
SD222	Поточна позиція інструкції позиціонування виходу Y3.	0	R/W	✓	✓	✓	✓	
SD223								
SD224	Поточна частота інструкції позиціонування виходу Y3.	0	R	✓	✓	✓	✓	✓
SD225								
SD226	Макс. швидкість, при якій ZRN, PLSV, DRVI, DRVA, DSZR та DVIT інструкції виконуються (10- 100000) (Y3)	100000	R/W	✓	✓	✓	✓	
SD227								
SD228	Базова швидкість, коли інструкції ZRN, PLSV, DRVI, DRVA, DSZR та DVIT є виконано (1/10 від макс. швидкості) (Y3)	5000	R/W	✓	✓	✓	✓	
SD229	Час розгону/гальмування для прискорення від базової швидкості (SD204) до макс. швидкості (SD202, SD203), коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються (Y3) (50ms — 5000ms)	1000	R/W	✓	✓	✓	✓	
SD230	Час сповільнення для зниження швидкості від максимальної до базової, коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються Y3 (50 мс — 5000 мс)	1000	R/W		✓			
SD231	Швидкість повзання Y3 застосовується до DSZR	1000	R/W	✓	✓	✓	✓	
SD232	Швидкість повернення до нуля Y3 застосовується до DSZR	50000	R/W	✓	✓	✓	✓	
SD233								
SD234	Кількість сегментів, для яких інструкція виводу PLS наразі виконується (Y3 застосовно)	0	R/W	✓	✓	✓	✓	
SD235	Y3 визначено як програмний елемент якого чіткий сигнал	0	R/W	✓	✓	✓	✓	
SD236	DVIT сигнал переривання програмний елемент специфікація (застосовно Y3)	0	R/W	✓	✓	✓	✓	

5) Y4 Кореляція регістру

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD240	Сумарна кількість імпульсів виходу Y4.	0	R/W	✓	✓	✓	✓	
SD241								
SD242	Поточна позиція інструкції позиціонування виходу Y4.	0	R/W	✓	✓	✓	✓	
SD243								
SD244	Поточна частота вихідної команди позиціонування Y4.	0	R	✓	✓	✓	✓	✓
SD245								

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD246	Макс. швидкість, при якій ZRN, PLSV,							

SD247	DRVI, DRVA, DSZR та DVIT інструкції виконуються (10- 100000) (Y4)	100000	R/W	✓	✓	✓	✓	
SD248	Базова швидкість, коли інструкції ZRN, PLSV, DRVI, DRVA, DSZR та DVIT є виконано (1/10 від макс. швидкості) (Y4)	5000	R/W	✓	✓	✓	✓	
SD249	Час розгону/гальмування для прискорення від базової швидкості (SD204) до макс. швидкості (SD202, SD203), коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються (Y4) (50ms — 5000ms)	1000	R/W	✓	✓	✓	✓	
SD250	Час сповільнення для зниження швидкості від макс. швидкості до базової швидкості, коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються Y4 (50 мс — 5000 мс)	1000	R/W		✓			
SD251	Швидкість повзання Y4 застосовується до DSZR	1000	R/W	✓	✓	✓	✓	
SD252	Швидкість повернення до нуля Y4 застосовується до DSZR	50000	R/W	✓	✓	✓	✓	
SD253								
SD254	Кількість сегментів, для яких інструкція виводу PLS наразі виконується (Y4 застосовно)	0	R/W	✓	✓	✓	✓	
SD255	Y4 визначено як програмний елемент якого чіткий сигнал	0	R/W	✓	✓	✓	✓	
SD256	DVIT сигнал переривання програмний елемент специфікація (застосовна Y4)	0	R/W	✓	✓	✓	✓	

6) Кореляція Y5 регістр

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD260	Сумарна кількість імпульсів виходу Y5.	0	R/W	✓	✓	✓	✓	
SD261								
SD262	Поточна позиція інструкції позиціонування виходу Y5.	0	R/W	✓	✓	✓	✓	
SD263								
SD264	Поточна частота інструкції позиціонування виходу Y5.	0	R	✓	✓	✓	✓	✓
SD265								
SD266	Макс. швидкість, при якій ZRN, PLSV, DRVI, DRVA, DSZR та DVIT інструкції виконуються (10- 100000) (Y5)	100000	R/W	✓	✓	✓	✓	
SD267								
SD268	Базова швидкість, коли інструкції ZRN, PLSV, DRVI, DRVA, DSZR та DVIT є виконано (1/10 від макс. швидкості) (Y5)	5000	R/W	✓	✓	✓	✓	
SD269	Час розгону/гальмування для прискорення від базової швидкості (SD204) до макс. швидкості (SD202, SD203), коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються (Y5) (50ms — 5000ms)	1000	R/W	✓	✓	✓	✓	

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD270	Час сповільнення для зниження швидкості від макс. швидкості до базової швидкості, коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються Y5 (50 мс — 5000 мс)	1000	R/W		✓			
SD271	Швидкість повзання Y5 застосовується до DSZR	1000	R/W	✓	✓	✓	✓	
SD272	Швидкість повернення до нуля Y5 застосовується до DSZR	50000	R/W	✓	✓	✓	✓	
SD273								
SD274	Кількість сегментів, на які PLS інструкція виводу наразі виконується (застосовно Y5)	0	R/W	✓	✓	✓	✓	
SD275	Y5 визначено як програмний елемент якого чіткий сигнал	0	R/W	✓	✓	✓	✓	
SD276	DVIT сигнал переривання програмний елемент специфікація (застосовна Y5)	0	R/W	✓	✓	✓	✓	

7) Y6 Кореляція регістру

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD280	Сумарна кількість імпульсів на виході Y6.	0	R/W	✓	✓	✓	✓	
SD281								
SD282	Поточна позиція інструкції позиціонування виходу Y6.	0	R/W	✓	✓	✓	✓	
SD283								
SD284	Поточна частота вихідної команди позиціонування Y6.	0	R	✓	✓	✓	✓	✓
SD285								
SD286	Макс. швидкість, при якій ZRN, PLSV, DRVI, DRVA, DSZR та DVIT інструкції виконуються (10- 100000) (Y6)	100000	R/W	✓	✓	✓	✓	
SD287								
SD288	Базова швидкість при ZRN, PLSV, DRVI, Інструкції DRVA, DSZR та DVIT виконуються (1/10 від макс. швидкості) (Y6)	5000	R/W	✓	✓	✓	✓	
SD289	Час розгону/гальмування для прискорення від базової швидкості (SD204) до макс. швидкості (SD202, SD203), коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються (Y6) (50ms — 5000ms)	1000	R/W	✓	✓	✓	✓	
SD290	Час сповільнення для зниження швидкості від макс. швидкості до базової швидкості, коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються Y6 (50 мс — 5000 мс)	1000	R/W		✓			
SD291	Швидкість повзання Y6 застосовується до DSZR	1000	R/W	✓	✓	✓	✓	
SD292								

SD293	Швидкість повернення до нуля Y6 застосовується до DSZR	50000	R/W	✓	✓	✓	✓	
SD294	Кількість сегментів, для яких інструкція виводу PLS наразі виконується (Y6 застосовно)	0	R/W	✓	✓	✓	✓	
SD295	Y6 визначено як програмний елемент якого чіткий сигнал	0	R/W	✓	✓	✓	✓	

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD296	DVIT сигнал переривання програмний елемент специфікація (застосовно Y6)	0	R/W	✓	✓	✓	✓	

8) Y7 Кореляція реєстр

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD300	Сумарна кількість імпульсів Y7 виходу.	0	R/W	✓	✓	✓	✓	
SD301								
SD302	Поточна позиція інструкції позиціонування виходу Y7.	0	R/W	✓	✓	✓	✓	
SD303								
SD304	Поточна частота вихідної команди позиціонування Y7.	0	R	✓	✓	✓	✓	✓
SD305								
SD306	Макс. швидкість, при якій ZRN, PLSV, DRVI, DRVA, DSZR та DVIT інструкції виконуються (10- 100000) (Y7)	100000	R/W	✓	✓	✓	✓	
SD307								
SD308	Базова швидкість, коли інструкції ZRN, PLSV, DRVI, DRVA, DSZR та DVIT є виконано (1/10 від макс. швидкості) (Y7)	5000	R/W	✓	✓	✓	✓	
SD309	Час розгону/гальмування для прискорення від базової швидкості (SD204) до макс. швидкості (SD202, SD203), коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються (Y7) (50ms – 5000ms)	1000	R/W	✓	✓	✓	✓	
SD310	Час сповільнення для зниження швидкості від макс. швидкості до базової швидкості, коли ZRN, DRVI, DRVA, DSZR та DVIT інструкції виконуються Y7 (50 мс – 5000 мс)	1000	R/W		✓			
SD311	Швидкість повзання Y7 застосовується до DSZR	1000	R/W	✓	✓	✓	✓	
SD312	Швидкість повернення до нуля Y7 застосовується до DSZR	50000	R/W	✓	✓	✓	✓	
SD313								
SD314	Кількість сегментів, для яких інструкція виводу PLS наразі виконується (Y7 застосовно)	0	R/W	✓	✓	✓	✓	
SD315	Y7 визначено як програмний елемент якого чіткий сигнал	0	R/W	✓	✓	✓	✓	
SD316	DVIT сигнал переривання програмний елемент специфікація (застосовна Y7)	0	R/W	✓	✓	✓	✓	

13. Вихід за часом інструкція

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD330	Час сканування, що використовується таймером годинника 1		R/W			✓	✓	✓
SD331	Час сканування, що використовується таймером годинника 2		R/W			✓	✓	✓
SD332	Час сканування, що використовується таймером годинника вивід 3		R/W			✓	✓	✓
SD333	Час сканування, що використовується таймером годинника 4		R/W			✓	✓	✓
SD334	Час сканування, що використовується таймером годинника вивід 5		R/W			✓	✓	✓

14. Інструкція сигналізації аварії

Адреса	Дія та функція	Початкове значення	R/W	VC1S	VC1	VC2	VC3	VC5
SD339	Зберігаючи мінімальну кількість дій у Сигнал тривоги S900-S999 інструкція	0	R/W				✓	✓

Додаток С Зарезервовані елементи

Опис	VC1S/VC1/VC2/VC3/VC5	
N:N зона спільного доступу до мережі	D7700	D7763
N:N розширений режим (режим 14-18) зона мережевого спільного доступу зона	D7500	D7755
N:N зона спільного доступу до мережі	M1400	M1911

Додаток D Коди помилок зв'язку Modbus коди

Аномальний код	Значення коду несправності
0x01	Неприпустимий код функції
0x02	Неприпустима адреса регістра
0x03	Номер даних помилка
0x10	Тайм-аут зв'язку, а саме час зв'язку перевищує максимальний час зв'язку, встановлений користувачем
0x11	Отримання кадру даних помилка
0x12	Помилка параметра, налаштування параметра (режим або ведучий/ведений) помилка
0x13	Номер станції збігається з тим, що встановлено інструкцією, і виникла помилка помилка
0x14	Переповнення адреси елемента (кількість отриманих або надісланих даних перевищує обсяг пам'яті елемента)
0x15	Виконання інструкції не вдалося
0x16	Отримана адреса підпорядкованого пристрою не збігається із запитаною, а елемент детального коду помилки містить інформацію про отриману адресу підлеглого пристрою
0x17	Отриманий код функції не збігається із запитаним, а елемент детального коду помилки зберігає його отримано функціональний код
0x18	Помилка кадру інформації: початкова адреса елемента не збігається, а елемент детального коду помилки зберігає the отримано початок елемента адреса
0x19	Отримана довжина даних не збігається з довжиною, визначеною протоколом, або кількість елементів перевищує вказану макс. число, визначене цією функцією code.
0x20	CRC/LRC помилка парності парності
0x21	Зарезервовано
0x22	Початкову адресу елемента параметра інструкції встановлено неправильно

0x23	Параметр інструкції встановлює невідтримуваний код функції або недопустимий код функції
0x24	Кількість елементів параметрів інструкції встановлено неправильно
0x25	Зарезервовано
0x26	Параметр не можна змінювати під час роботи.
0x27	Параметр захищено паролем

Додаток Е Коды системних помилок кодів

Код помилки	Значення	Тип помилки	Опис
0	Помилка не виникла		
1–9	Зарезервовано		
10	SRAM помилка	Помилка системи	Користувальська програма зупиняється ERR індикатор вмикається. Щоб усунути цю помилку, необхідно вимкнути PLC та перевірити його апаратне забезпечення
11	FLASH помилка	Помилка системи	Користувальська програма зупиняється ERR індикатор вмикається. Щоб усунути цю помилку, необхідно вимкнути PLC та перевірити його апаратне забезпечення
12	Помилка комунікаційного порту error	Помилка системи	Користувальська програма зупиняється ERR індикатор вмикається. Щоб усунути цю помилку, необхідно вимкнути PLC та перевірити його апаратне забезпечення
13	Помилка годинника реального часу	Помилка системи	Користувальська програма зупиняється Індикатор ERR вмикається. Щоб усунути цю помилку, необхідно вимкнути живлення PLC та перевірити апаратне забезпечення.
14	I2C помилка	Помилка системи	Користувальська програма зупиняється Індикатор ERR вмикається. Щоб усунути цю помилку, необхідно вимкнути живлення PLC та перевірити апаратне забезпечення.
20	Серйозна локальна I/O помилка	Помилка системи	Користувальська програма зупиняється Індикатор ERR вмикається. Щоб усунути цю помилку, необхідно вимкнути живлення PLC та перевірити апаратне забезпечення.
21	Серйозна помилка I/O розширення	Помилка системи	ERR індикатор блимає Сигнал тривоги скидається автоматично після усунення помилки
22	Серйозна помилка спеціального модуля error	Помилка системи	ERR індикатор блимає Сигнал тривоги скидається автоматично після усунення помилки
23	Помилка оновлення годинника реального часу (неправильний час зчитується під час той система оновлення)	Помилка системи	ERR індикатор блимає Сигнал тривоги скидається автоматично після усунення помилки
24	EEPROM читання/читання помилка операції	Помилка системи	ERR індикатор блимає Сигнал тривоги скидається автоматично після усунення помилки
25	Локальний аналоговий сигнал помилка	Помилка системи	ERR індикатор блимає Сигнал тривоги скидається автоматично після усунення помилки
26	Помилка конфігурації спеціального модуля	Помилка системи	ERR індикатор блимає Сигнал тривоги скидається автоматично

	системи		після усунення помилки
40	Файл програми користувача помилка	Помилка системи	Програма користувача зупиняється, і індикатор ERR вмикається ON Щоб усунути цю помилку, необхідно завантажити створити нову програму або відформатувати диск
41	Файл конфігурації системи помилка	Помилка системи	Програма користувача зупиняється ERR індикатор вмикається Щоб усунути цю помилку, необхідно завантажити нові файли конфігурації системи або форматування диска

Код помилки	Значення	Тип помилки	Опис
42	Помилка файла блоку даних	Помилка системи	Програма користувача зупиняється, і ERR індикатор вмикається ON Щоб усунути цю помилку, необхідно завантажити новий файл блоку даних або форматування диска
43	Втрата даних з живленням від акумулятора помилка	Помилка системи	Користувачка програма продовжує виконуватися, а індикатор ERR блимає Щоб усунути цю помилку, необхідно очистити його елементи, відформатуйте диск або скиньте.
44	Примусова помилка втрати таблиці error	Помилка системи	Користувачка програма продовжує виконуватися, а індикатор ERR блимає Щоб усунути цю помилку, необхідно очистити його елементи, примусово, відформатувати диск або скинути.
45	Файл інформації користувача помилка	Помилка системи	Програма користувача зупиняється, і ERR індикатор вмикається ON Щоб усунути цю помилку, необхідно завантажити нові файли програми та блоку даних або відформатувати його диск
46–59	Зарезервовано		
60	Компіляція програми користувача помилка	Помилка виконання	Програма користувача зупиняється, і ERR індикатор вмикається ON.
61	Робота користувачкої програми тайм-аут	Помилка виконання	Програма користувача зупиняється, і ERR індикатор вмикається ON.
62	Неприпустима програма користувача помилка виконання інструкції error	Помилка виконання	Програма користувача зупиняється, і ERR індикатор вмикається ON.
63	Неприпустимий тип елемента з інструкції операнд	Помилка виконання	Програма користувача зупиняється, і ERR індикатор вмикається ON.
64	Неприпустимий операнд інструкції значення	Помилка виконання	Користувачка програма продовжує виконуватися, і ERR індикатор залишається вимкненим. Відповідний код помилки відображається в SD20
65	Поза межами інструкції елемент операнда діапазон	Помилка виконання	
66	Підпрограма стек переповнення	Помилка виконання	
67	Запит на переривання користувачем запит переповнення черги	Помилка виконання	

68	Неприпустимий перехід за міткою або виклик підпрограми	Помилка виконання	
69	Ділення на 0 помилка	Помилка виконання	
70	Помилка визначення стека, що використовується	Помилка виконання	Коли розмір стека або кількість елементів стека менші за нуль, або кількість елементів стека перевищує ліміт стека розмір
71	Зарезервовано	Помилка виконання	
72	Невизначена підпрограма користувача або переривання підпрограма	Помилка виконання	
73	Недійсний спеціальний модуль адреса	Помилка виконання	
74	Виникла помилка під час оцініть спеціальний модуль	Помилка виконання	
75	I/O негайне оновлення помилка	Помилка виконання	
76	Помилка налаштування годинника	Помилка виконання	
77	PLSR інструкція параметр помилка	Помилка виконання	
78	BFM буфер оцінених спеціальний модуль виходить за межі діапазону	Помилка виконання	

Додаток F Протоколи зв'язку Modbus протоколи

1. Огляд протоколу зв'язку Modbus protocol

Мікро-VC приводи серії PLCS оснащені двома або трьома комунікаційними портами: PORT0 (також як порт програмування) підтримує підлеглу станцію Modbus, PORT1 та PORT2 підтримують головну та підлеглу станції Modbus (налаштовуються через фонове програмне забезпечення).

1. Використовуючи інтерфейс RS485 або RS232, з 3-лінійною системою RS232 як фізичним інтерфейсом.
2. Підтримка RTU режимів
3. Будучи підпорядкованою станцією Modbus з адресою в діапазоні від 1 до 31.
4. Підтримка режиму ширококомовлення. Широкомовлення є ефективним для кодів запису та підфункцій діагностики.
5. Підтримка швидкостей передачі даних, включаючи 38 400 біт/с, 19 200 біт/с, 9 600 біт/с, 4 800 біт/с, 2 400 біт/с та 1 200 біт/с. (За замовчуванням: 19200, 8 біт, 1 стоповий біт та перевірка на парність)
6. Поле допоміжних даних 2 × 252 байти (режим ASCII) або 252 байти (режим RTU).

2. Підтримувані коди функцій Modbus та адресація елементів режим

Підпорядкована станція підтримує функціональні коди 01, 02, 03, 05, 06, 08, 15 та 16 (десяткові).

- Зв'язок між кодом функції читання/запису елемента та елементом

Функціональний код	Назва функції код	Адреса даних Modicon addresss	Тип керованого елемента	Примітка
01	Зчитати котушку	0 ^{Примітка 1} :xxxx	Y, X, M, SM, S, T та C	Біт зчитування
02	Зчитати дискретний вхід	1 ^{Примітка 2} :xxxx	X	Біт зчитування
03	Зчитати регістр	4 ^{Примітка 3} :xxxx ^{Примітка 4}	D, SD, Z, T та C	Слово зчитати
05	Запишіть одну котушку	0:xxxx	Y, M, SM, S, T та C	Біт запису
06	однорядковий зареєструвати	4:xxxx	D, SD, Z, T та C	Слово записати
15	Запишіть декілька катушок	0:xxxx	Y, M, SM, S, T та C	Біт запису
16	декілька регістри	4:xxxx	D, SD, Z, T та C	Слово записати

Примітки:

- 0 вказує на "катушку".
- 1 вказує на "дискретний вхід".
- 4 вказує на "регістр".
- xxxx вказує діапазон «1–9999». Кожен тип має незалежний діапазон логічних адрес від 1 до 9999 (адреса протоколу починається з 0).
- 0, 1 та 4 не мають фізичного значення і не задіяні у фактичній адресації.
- Ви не повинні записувати дані в елемент X за допомогою функціональних кодів 05 та 15; в іншому випадку система не надає зворотний зв'язок про помилку, якщо записаний операнд і дані правильні, але система не виконує жодної операції за цією інструкцією.

- Зв'язок між елементом PLC та адресою протоколу зв'язку Modbus

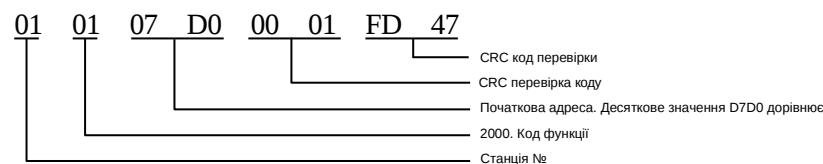
Елемент немає	Тип	Фізичний елемент	Протокол адреса	Підтримується функція code	Примітка
Y	Біт	Y0–Y777 (вісімковий код) 512 точок у загальній кількості	0000–0511	01, 05 та 15	Стан виходу, код елемента: Y0–Y7 та Y10–Y17

X	Біт	X0–X777 (вісімковий код) 512 точок у загальній кількості	1200–01711	01, 05 та 15 02	Стан входу. Він підтримує два типи адрес, а код елемента — так само, як вище
M	Біт	M0–M2047	2000–4047	01, 05 та 15	

Елемент Т немає	Тип	Фізичний елемент	Протокол адреса	Підтримується функція code	Примітка
		M2048–M10240	12000–20191		
SM	Біт	SM0–SM255 SM256–SM1023	4400–4655 30000–30767	01, 05 та 15	
S	Біт	S0–S1023 S1024–S4095	6000–7023 31000–34071	01, 05 та 15	
T	Біт	T0–T255 T256–T511	8000–8255 11000–11255	01, 05 та 15	Стан Т елемента
C	Біт	C0–C255 C256–C511	9200–9455 10000–10511	01, 05 та 15	Стан С елемента
D	Слово	D0–D7999	0000–7999	03, 06 та 16	
SD	Слово	SD0–SD255 SD256–SD1023	8000–8255 12000–12767	03, 06 та 16	
Z	Слово	Z0–Z15	8500–8515	03, 06 та 16	
T	Слово	T0–T255 T256–T511	9000–9255 11000–11255	03, 06 та 16	Поточне значення Т елемента
C	Слово	C0–C199	9500–9699	03, 06 та 16	Поточне значення елемента С (INT)
C	Подвійний слово	C200–C255	9700–9811	03 та 16	Поточне значення елемента С (DINT)
C	Подвійний слово	C256–C306	10000–10101	03 та 16	Поточне значення елемента С (DINT)
R	Слово	R0–R32767	13000–45767	03, 06 та 16	

Примітка:

Протокольна адреса — це адреса, що використовується під час передачі даних і відповідає логічній адресі даних Modicon. Протокольна адреса починається з 0, тоді як логічна адреса даних Modicon починається з 1, тобто протокольна адреса + 1 = логічна адреса даних Modicon. Наприклад, якщо протокольна адреса M0 дорівнює 2000, то відповідна їй логічна адреса даних Modicon — 0:2001. Фактично, зчитування та запис M0 виконуються через протокольну адресу, наприклад: кадр зчитування елемента M0 (надісланий з головної станції).



- Аномальна відповідь опис:

Аномальний код	Визначення
0x01	Код помилки code
0x02	Регістр помилок адреса
0x03	Помилка даних

- Примітки

- Елементи X та Y використовують вісімкову систему. Всього існує 256 точок від X0 до X377, 256 точок від Y0 до Y377, з комбінаціями Y0–Y7, Y10–Y17, Y20–Y27 тощо.
- Для X елемента доступні два методи адресації. Один — це протокольна адреса 1200–1455 з відповідними кодами функцій 01, 05 та 15; інший — це протокольна адреса 0–255 з кодом функції 02.
- Обробка 32-бітних елементів: елемент С є лічильником. Він має свій стан та поточне значення. C200–C255 — це 32-бітні елементи, але кожен елемент С у цьому діапазоні займає дві протокольні адреси під час

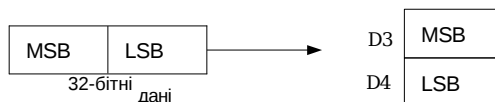
протокольної адресації. Для

Протокольна адреса C200 — 9700–9701. Під час зчитування елементів через Modbus, як початкова протокольна адреса, так і кількість елементів, що зчитуються, є парними числами.

4. Для більшості елементів SM та SD реальне значення не може бути записане через Modbus, проте підпорядкована станція PLC все одно повертає "OK" для індикації завершення операції запису, що є допустимим.

5. Крім того, протокол зв'язку Modbus серії VC підтримує читання та запис елементів подвійного слова, змінної LONGINT та чисел з рухомою комою. У PLCs серії VC 32-бітні дані зберігаються зі старшими бітами за старшою адресою. Для

наприклад, 32-бітні дані зберігаються у двох D елементах (D3 та D4), з MSB у D3 та LSB у D4, як показано на наступному рисунку: (Зверніться до опису для конкретного прикладу)



3. Код функції Modbus опис

3.1 Зчитування стану котушки (0x01)

А максимум 256 бітових елементів можна зчитати у серії VC PLCs.

1. Запит корпусу

Адреса	Код функції (01H)	Почати адресу		Кількість елементів		
		H	L	H	L	

2. Відповідь кадру

Якщо адреса зчитування не кратна 8, біти, що залишилися, заповнюються 0 (починаючи зі старших бітів).

Адреса s	Код функції (01H)	Кількість зчитаних елементів (n) (Кількість байтів)	Зчитати дані №1		Зчитати дані № n/p	Перевірити код (CRC або LRC)

3.2 (0x02)Зчитування стану дискретного входу (0x02)

У серії VC PLC це спеціально стосується елемента X. Код функції підтримує лише функцію читання елемента X з максимальною кількістю зчитувань 256.

1. Запит корпусу

Адреса	Функціональний код (02H)	Почати адресу		Кількість елементів		Перевірте код (CRC або LRC)
		H	L	H	L	

2. Відповідь кадру

Якщо адреса зчитування не є кратною 8, решта бітів заповнюються 0 (починаючи зі старших бітів).

Адреса	Функція код (02H)	Кількість зчитаних елементів (n) (Кількість байтів)	Зчитайте дані №1		Зчитати дані № n/p	Перевірте код (CRC або LRC)

3.3 Читання регістрів зберігання (0x03)

Зчитування регістрів зберігання означає зчитування значення регістра даних (слова) на підпорядкованій станції, з можливістю зчитування максимум 125 регістрів. Широкомовна передача не підтримується.

1. Запит корпусу

Адреса	Функціональний код	Почати адресу	Кількість елементів	Перевірте код

	(03H)	H	L	H	L	(CRC або LRC)

2. Відповідь кадру

Адреса	Функція код (03H)	Кількість елементів для зчитування (n) (Кількість байти)	Зчитайте дані №1			Зчитати дані № п/п		Перевірте код (CRC або LRC)
			H	L		H	L	

3.4 Примусовий запис (запис) окремої котушки (0x05)

Примусовий запис (запис) окремої котушки записує значення бітового елемента на підпорядковану станцію та підтримує широкомовлення, тобто запис того самого елемента на всі підпорядковані станції. Підтримується максимум 1 бітовий елемент.

1. Запит корпусу

Адреса	Функціональний код (05H)	Почати адресу		Записаний елемент значення		Перевірте код (CRC або LRC)
		H	L	H	L	

Примітка: Записане значення елемента дорівнює 0xFF00(ON,1) або 0x0000(OFF,0).

2. Відповідь кадру

Кадр відповіді є повторенням кадру запиту.

Адреса	Функціональний код (05H)	Почати адресу		Записаний елемент значення		Перевірте код (CRC або LRC)
		H	L	H	L	

3.5 Попереднє встановлення (запис) одного регістра (0x06)

Попереднє встановлення (запис) одного регістра (0x06) записує значення елемента слова на підлеглу станцію та підтримує широкомовну передачу, тобто запис одного й того самого елемента на всі підлеглі станції. Підтримується максимум 1 елемент слова.

1. Запит корпусу

Адреса	Функціональний код (06H)	Почати адресу		Записаний елемент значення		Перевірте код (CRC або LRC)
		H	L	H	L	

2. Відповідь кадру

Кадр відповіді є повторенням кадру запиту.

Адреса	Функціональний код (06H)	Почати адресу		Записаний елемент значення		Перевірте код (CRC або LRC)
		H	L	H	L	

3.6 Повернення діагностичної перевірки (0x08)

Діагностичні регістри та інформацію про помилки зв'язку можна отримати за допомогою діагностичної перевірки.

Діагностичний код	Значення
0x00	Повернення запиту кадру
0x 01	Перезапуск комунікаційної опції
0x 04	Режим «тільки прослуховування» підпорядкованої станції
0x0a	Очищення лічильників та діагностичних регістрів
0x0b	Повернення повідомлення шини лічильника
0x0c	Повернення помилки шини CRC лічильник
0x0d	Повернення лічильника помилок підпорядкованого пристрою
0x0e	Повернення повідомлення підпорядкованого пристрою лічильник
0x0f	Повернення кількості відповідей від підпорядкованого пристрою відсутнє
0x12	Повернення лічильника переповнення символів шини

Опис кадру підфункціонального коду є наступним.

- Повернення запиту кадр(0x00):

1. Запит корпусу

Адреса	Функціональний код (0x08H)	Функціональний код		Будь-який символ		Перевірте код (CRC або LRC)
		(0x00H)	(0x00H)	H	L	

2. Відповідь кадру

Повернення кадру запиту без змін.

Адреса	Функціональний код (0x08H)	Функціональний код		Будь-який символ		Перевірте код (CRC або LRC)
		(0x00H)	(0x00H)	H	L	

- Перезапуск опції зв'язку (0x01):

Після отримання кадру, PLC вихід із режиму лише прослуховування (широкомовний кадр підтримується).

1. Запит корпусу

Поле нормальних даних становить 0x00 00 або 0xff 00.

Адреса	Функціональний код (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		0x00H	0x01H	H	L	

2. Відповідь кадру

Адреса	Функціональний код (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		0x00H	0x01H	H	L	

- Режим «тільки прослуховування» підпорядкованої станції(0x04):

Підпорядкована станція переходить у режим «тільки прослуховування». Жодна інструкція не виконується і не потребує відповіді. Підпорядкована станція може розпізнати лише інструкцію опції перезавантаження зв'язку і переходить у режим онлайн після отримання цієї інструкції (широкомовний кадр підтримується).

1. Request frame

Адреса	Код функції (0x08H)	Функціональний код коду		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x04H)	0x00H	0x00H	

2. Кадр відповіді

Без повернення.

- Очищення лічильників та діагностичних регістрів

(0x0A): Очищення всіх лічильників (підтримується

широкомовний кадр).

1. Запит корпусу

Адреса	Код функції (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0AH)	0x00H	0x00H	

2. Відповідь кадру

Адреса	Код функції (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0AH)	0x00H	0x00H	

- Повернення повідомлення шини лічильник(0x0B):

Запис загальної кількості повідомлень до всіх головних станцій від підпорядкованих станцій з моменту останнього запуску, скидання та ввімкнення живлення лічильників, що виключає повідомлення про CRC помилку.

1. Запит корпусу

Адреса	Код функції (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0BH)	0x00H	0x00H	

2. Відповідь кадру

Адреса	Функціональний код (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0BH)	H	L	

- CRC лічильник помилок (0x0C):

Запис кількості помилок CRC, отриманих підпорядкованою станцією з моменту останнього запуску, скидання та ввімкнення живлення лічильників.

1. Запит корпусу

Адреса	Код функції (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0CH)	0x00 H	0x00 H	

2. Відповідь кадру

Адреса	Функціональний код (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0CH)	H	L	

- Повернення кількості помилок підпорядкованого пристрою (0x0D):

Запис кількості помилок, виявлених підпорядкованою станцією з моменту останнього запуску, скидання та ввімкнення живлення лічильників, що включає помилки, виявлені у ширококомовному повідомленні.

1. Запит корпусу

Адреса	Код функції (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0DH)	0x00 H	0x00 H	

2. Відповідь кадру

Адреса	Функціональний код (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0DH)	H	L	

- Повернення повідомлення підпорядкованого пристрою count(0x0E):

Запис кількості повідомлень адресації, отриманих підпорядкованою станцією з моменту останнього запуску, очищення та ввімкнення живлення лічильників.

1. Запит корпусу

Адреса	Код функції (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0EH)	0x00 H	0x00 H	

2. Відповідь кадру

Адреса	Функціональний код (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0EH)	H	L	

- Повернення кількості відсутності відповіді від підлеглого пристрою (0x0F):

Запис кількості повідомлень, що не повернулися на підпорядковану станцію з моменту останнього запуску, очищення та ввімкнення живлення лічильників.

1. Запит корпусу

Адреса	Код функції (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0FH)	0x00 H	0x00 H	

2. Відповідь кадру

Адреса	Функціональний код (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x0FH)	H	L	

- Повернення лічильника переповнення символів шини (0x12)

Запис кількості повідомлень, які неможливо обробити через переповнення символів з моменту останнього запуску, очищення та ввімкнення живлення лічильників.

1. Запит корпусу

Адреса	Код функції (0x08H)	Функціональний код	Поле даних	Перевірте код (CRC або LRC)

Адреса	Функціональний код (0x08H)	Функціональний код		Поле даних		Перевірте код (CRC або LRC)
		(0x00H)	(0x12H)	H	L	

Адреса	Код функції (OFN)	Початок адреса		Кількість від елементи		Кількість байтів (n)	Значення записаного елемента №1		Значення елемента, що записується №N	Перевірте код (CRC або LRC)
		H	L	H	L					

B7	B6	B5	B4	B3	B2	B1	B0

Додат и тиск	Функціональний код	Почати адресу		Кількість елементів		Перевірте код
	(OFN)	H	L	H	L	(CRC або LRC)

Адреса	Функціональний код (0x10H)	Початкова адреса		Кількість елементів		Кількість байтів (n)	Значення письмового елемента №1			Значення письмового елемента №N		Перевірте код (CRC або LRC)
		H	L	H	L		H	L		H	L	

Додати натиснути	Функціональний код (0x10H)	Почати адресу		Кількість елементів		Перевірте код (CRC або LRC)
		H	L	H	L	

Наприклад: для елемента Y діапазон адрес протоколу становить 0000–0255 (Y0–Y377):

- Якщо початкова адреса зчитування дорівнює 1 і зчитується 256 елементів, повертається помилка адреси (код помилки 02), оскільки існує лише 255 Y елементів, що починаються з 1.
- Якщо початкова адреса зчитування дорівнює 0 і зчитується 257 елементів, повертається помилка даних (код помилки 03), оскільки фактична визначена кількість Y елементів становить лише 256.
- Якщо початкова адреса зчитування дорівнює 0 і зчитується 256 елементів, стани 256 елементів повертаються.

Іншими словами, кількість зчитаних елементів повинна бути в межах фактично визначеного діапазону. Це справедливо для читання/запису бітових/словних елементів.

4. Приклад керування зв'язком Modbus

Підпорядкована станція Modbus відповідає на повідомлення від головної станції відповідно до отриманого повідомлення локальної адресації, а не надсилає повідомлення активно. Підпорядкована станція підтримує лише коди функцій Modbus 01, 02, 03, 05, 06, 08, 15 та 16, а інші коди є "неприпустимими кодами функцій" (за винятком широкомовного кадру).

● Читання та запис елементів

Усі підтримувані коди функцій, за винятком 08, використовуються для читання та запису елементів. У принципі, в одному кадрі можна зчитати максимум 2000 бітових елементів та 125 словних елементів, а записати максимум 1968 бітових елементів та 120 словних елементів. Однак фактичні адреси протоколу для різних типів елементів відрізняються та є розривними (наприклад, адреса протоколу Y377 — 255, тоді як адреса протоколу X0 — 1200). Тому під час читання або запису елемента можна зчитувати лише елементи одного типу, а максимальна кількість елементів, що зчитуються/записуються, залежить від кількості фактично визначених елементів. Наприклад, під час читання Y елементів (Y0 – Y377) діапазон адрес протоколу становить від 0 до 255, відповідна логічна адреса даних Modicon — від 1 до 256, а максимальна кількість Y елементів, які можна зчитати, становить 256. Приклади наведено нижче:

1. Передача з головної станції: 0101 000001003D 9A

01 – адреса; 01 – код функції; 00 00 – початкова адреса; 01 00 – кількість елементів для зчитування; 3D 9A – перевірка Відповіді підпорядкованої станції: надання коректної відповіді

2. Надсилання з головної станції: 01 01 00 00 01 01 FC 5A

Головна станція зчитує 01 01 (257) елементів, що перевищує визначену кількість Y елементів.

Відповідь підпорядкованої станції: 01 81 03 00 51

Відповідь підпорядкованої станції містить неприпустиму адресу, оскільки 257>256, а 256 є максимально дозволеною кількістю Y елементів.

3. Надсилання з головної станції: 01 01 00 64 00 A0 7D AD

00 64 (десятькове: 100) — це початкова адреса для зчитування головною станцією, а 00 A0 (десятькове: 160) — це кількість елементів.

Відповідь підпорядкованої станції: 01 81 02 C1 91

Відповідь підпорядкованої станції містить неприпустиму адресу, оскільки адреса протоколу 300 не має визначення бітових елементів.

Відповідь підпорядкованої станції містить неприпустиму адресу, оскільки існує лише 156 Y елементів, визначених починаючи зі 100, а 160 Y елементів перевищують вказану кількість.

4. Надсилання з головної станції: 01 01 01 2C 00 0A 7C 38

Головна станція зчитала 10 бітових елементів 01 2C

(десятькове: 300). Відповідь підпорядкованої станції: 01 81 02

C1 91

Відповідь підпорядкованої станції містить неприпустиму адресу, оскільки адреса протоколу 300 не має визначення бітових елементів.

5. Надсилання з головної станції: 01 04 00 02 00 0A D1 CD

Головна станція надсилає кадр із кодом функції 04

Відповідь підпорядкованої станції: 01 84 01 82 C0

Відповідь підпорядкованої станції містить неприпустимий функціональний код.

6. Надсилання з головної станції: 01 02 00 00 00 0A F8 0D

Головна станція зчитує 10 (X0–X9) вхідних елементів (X елементів) від початкової адреси.

Відповідь підпорядкованої станції: 01 02 02 00 00 B9 B8

Підпорядкована станція відповідає коректною інформацією, яка має 02 байти, а вміст становить 00 00.

7. Надсилання з головної станції: 01 01 04 B0 00 0A BC DA

Головна станція зчитує 10 бітових елементів (X0–X9), починаючи з 04 B0 (десятькове:

1200). Відповідь підпорядкованої станції: 01 01 02 00 00 B9 FC

Підпорядкована станція відповідає 02 байтами, і вміст становить 00 00.

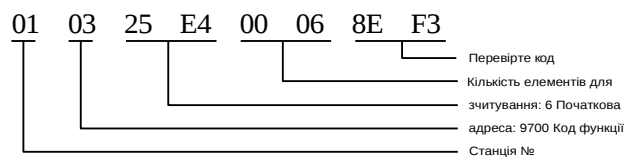
□ Примітка

1. Адреса підпорядкованої станції — 01, останні два байти є контрольними кодами CRC, а другий байт — функціональним кодом.
2. X елемент не підтримує операцію запису.

● Обробка подвійних слівелементів

1. Поточне значення лічильника елемента C є елементом типу слово або подвійне слово. Значення від C200 до C263 є елементами типу подвійне слово, які зчитуються та записуються через функціональні коди (03 та 16) регістрів зчитування/запису. Адреса кожних двох регістрів відповідає одному елементу типу подвійне слово C, і регістри можуть бути зчитані або записані лише парами.

Наприклад, зчитування RTU кадру з трьох C подвійних слів від C200 до C202.



У повернутих даних дві адреси 9700 та 9701 вказують на вміст C200. 9700 — це MSB, а 9701 — це LSB.

2. Під час зчитування елемента подвійного слова, якщо початкова адреса для зчитування не є парним числом, система відповідає кодом помилки недопустимої адреси. Наприклад:

Надсилання з головної станції: 0103 25E500 045E F2

Початкова адреса для зчитування, надіслана головною станцією, становить 25 E5 (чотири елементи слів, десяткове число: 9701). Відповідь підпорядкованої станції: 01 8302 C0 F1

Відповідь підпорядкованої станції: Неприпустима адреса даних

3. Якщо кількість зчитаних регістрів не є парним числом, система відповідає кодом помилки некоректних даних.

Наприклад: Надсилання з головної станції: 01 03 25 E4 00 05 CE F2

Початкова адреса для зчитування головною станцією — 25 E4 (5 елементів даних). Відповідь підпорядкованої станції: 01 83 03 01 31

Підпорядкована станція повертає неприпустимі дані.

● Обробка LONG INT даних

Згідно з методом зберігання VC серії PLCs, одні LONG INT дані можуть бути збережені у двох D елементах. Наприклад, якщо LONG INT дані збережені в D3 та D4, VEICHI PLC вважають, що MSB зберігаються в D3, а LSB зберігаються в D4. Коли головна станція зчитує LONG INT дані через Modbus, 32-бітні дані повинні бути перегруповані згідно з принципом LONG INT storage VEICHI PLC після зчитування даних.

Принцип зберігання FLOAT такий самий, як і принцип зберігання LONG INT.

5. Опис трансляції

Підпорядкована станція підтримує широкомовлення, але не всі коди функцій. Підпорядкована станція підтримує коди функцій 01, 02, 03, 05, 06, 08, 15 та 16 (десяткові), серед яких 01, 02 та 03 можуть зчитувати елемент, але не підтримують широкомовлення, відповідь не отримується після

надсилання широкомовного повідомлення; 05, 06, 15 та 16 — це елементи запису, що підтримують широкомовну передачу, відповідь після надсилання широкомовного повідомлення не отримується, але підпорядкована станція обробляє отримані дані; 08 — це код діагностичної функції, він не підтримує широкомовну передачу, за винятком його підфункцій 0x01, 0x04 та 0x0A (шістнадцяткові).

Додаток G ASCII таблиця кодування символів коду

ASCII HEX code		3 найважливіші біти							
		0	1	2	3	4	5	6	7
Молодші 4 біти	0	NUL	DLE	SPACE	0	@	P	`	п
	1	SOH	DC1	!	1	A	Q	a	q
	2	STX	DC2	"	2	B	R	б	г
	3	ETX	DC3	#	3	C	S	с	с
	4	EOT	DC4	\$	4	D	T	д	т
	5	ENQ	NAK	%	5	E	U	е	у
	6	ACK	SYN	&	6	F	V	ф	в
	7	BEL	ETB	'	7	G	W	г	ш
	8	BS	CAN	(	8	H	X	h	x
	9	HT	EM	)	9	I	Y	я	т
	A	LF	SUB	*	:	J	Z	й	z
	B	VT	ESC	+	;	K	[	к	{
	C	FF	FS	,	<	L		л	
	D	CR	GS	-	=	M	]	м	}
	E	SO	RS	.	>	N	^	н	—
	F	SI	US	/	?	O	—	о	DEL

Додаток H Показчик порядку інструкцій таблиця

Інструкція	Функція інструкції	Крок довжина	Прапорці впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
ACOS	Інструкція для отримання числа з рухомою комою числа	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	115
ADD	Інструкція додавання цілих чисел instruction	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	98
ANB	Блок потоку енергії та інструкція	1		✓	✓	✓	✓	✓	61
AND	NO контакт та інструкція	1		✓	✓	✓	✓	✓	59
AND<	Порівняння цілих чисел AND< інструкція	5		✓	✓	✓	✓	✓	200
AND<=	Порівняння цілих чисел AND<= інструкція	5		✓	✓	✓	✓	✓	200
AND<>	Порівняння цілих чисел AND<> інструкція	5		✓	✓	✓	✓	✓	200
AND=	Порівняння цілих чисел AND= інструкція	5		✓	✓	✓	✓	✓	200
AND>	Порівняння цілих чисел AND> інструкція	5		✓	✓	✓	✓	✓	200
AND>=	Порівняння цілих чисел AND>= інструкція	5		✓	✓	✓	✓	✓	200
ANDD<	Порівняння довгих цілих чисел AND< інструкція	7		✓	✓	✓	✓	✓	203
ANDD<=	Порівняння довгих цілих чисел AND<= інструкція	7		✓	✓	✓	✓	✓	203

A		чисел AND<= інструкція								
	ANDD<>	Порівняння довгих цілих чисел AND<> інструкція	7		✓	✓	✓	✓	✓	203
	ANDD=	Порівняння довгих цілих чисел AND= інструкція	7		✓	✓	✓	✓	✓	203
	ANDD>	Порівняння довгих цілих чисел AND> інструкція	7		✓	✓	✓	✓	✓	203
	ANDD>=	Порівняння довгих цілих чисел AND>= інструкція	7		✓	✓	✓	✓	✓	203
	ANDR<	Число з рухомою комою порівняння AND> інструкція	7		✓	✓	✓	✓	✓	206
	ANDR<=	Число з рухомою комою порівняння AND<= інструкція	7		✓	✓	✓	✓	✓	206
	ANDR<>	Число з рухомою комою порівняння AND<> інструкція	7		✓	✓	✓	✓	✓	206
	ANDR=	Число з рухомою комою порівняння AND= інструкція	7		✓	✓	✓	✓	✓	206
	ANDR>	Число з рухомою комою порівняння AND> інструкція	7		✓	✓	✓	✓	✓	206
	ANDR>=	Число з рухомою комою порівняння AND>= інструкція	7		✓	✓	✓	✓	✓	206
	ANI	NC контакт та інструкція	1		✓	✓	✓	✓	✓	60
	ANR	Інструкція скидання сигналу тривоги	1	Нуль, перенос та позика				✓	✓	249
	ANS	Інструкція налаштування сигналу тривоги	7	Нуль, перенос та позика				✓	✓	248
	ASC	ASCII код перетворення інструкція	19	Нуль, перенос та позика	✓	✓	✓	✓	✓	123
	ASIN	Інструкція для отримання ASIN числа з рухомою комою числа	7	Нуль, перенос та позика				✓	✓	115
	ATI	Інструкція для перетворення одного ASCII код у 16-бітні шістнадцяткові дані	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	124
	ATAN	Інструкція для отримання ATAN числа з рухомою комою числа	7	Нуль, перенос та позика				✓	✓	115
	ALT	Альтернативна інструкція	11	Нуль, перенос та позика	✓	✓	✓	✓	✓	184

Інструкція		Функція інструкції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	ABSD	Абсолютне керування кулачком інструкція	9	Нуль, перенос та позика			✓	✓	✓	181
B	BAND	Бітовий контакт слова AND інструкція	5		✓	✓	✓	✓	✓	196
	BANI	Інструкція ANI контакту біта слова	5		✓	✓	✓	✓	✓	197
	BCD	Інструкція для перетворення а слово до 16-бітного BCD коду	5	Нуль, перенесення та позика	✓	✓	✓	✓	✓	119
	BIN	BIN: Інструкція для перетворення а 16-бітний BCD код у слово	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	120
	BITS	Інструкція з підрахунку на біт в слово	5		✓	✓	✓	✓	✓	194
	BKADD	Операція масового додавання даних operation інструкція	9	Нуль, перенос та позика				✓	✓	210
	BKCMP=, >, <, <=, >=	Інструкція порівняння масивів даних	9					✓	✓	211
	BKSUB	Операція віднімання масиву даних operation інструкція	9	Нуль, перенос та позика				✓	✓	211
	BLD	Бітовий контакт слова LD інструкція	5		✓	✓	✓	✓	✓	195
	BLDI	Бітовий контакт слова LDI інструкція	5		✓	✓	✓	✓	✓	196
	BMOV	Передача блоку даних інструкція	7		✓	✓	✓	✓	✓	92
	BON	Інструкція для оцінювання за бітом в слово	7					✓	✓	195
	BOR	Бітовий контакт слова OR інструкція	5		✓	✓	✓	✓	✓	197
	BORI	Бітовий контакт слова ORI інструкція	5		✓	✓	✓	✓	✓	198
	BOUT	Бітовий вихід котушки інструкція	5		✓	✓	✓	✓	✓	198
B	BRST	Інструкція скидання бітової котушки інструкція	5		✓	✓	✓	✓	✓	199
	BSET	Інструкція встановлення біта котушки слова instruction	5		✓	✓	✓	✓	✓	198
	BTOW	Комбінація даних у байтових одиницях комбінація інструкція	7	Нуль, перенос та позика				✓	✓	245
C	CALL	Інструкція для виклику користувача підпрограма			✓	✓	✓	✓	✓	90
	CCITT	CCITT перевірте інструкцію	7		✓	✓	✓	✓	✓	191
	CCW	Траєкторія дуги проти годинникової стрілки траєкторія інтерполяція	12	Нуль, перенесення та позика				✓	✓	238
	CFEND	Інструкція для умовного повернення головної програми користувача	1		✓	✓	✓	✓	✓	88
	CIRET	Інструкція для умовного повернення користувацького переривання програми	1		✓	✓	✓	✓	✓	89
	CJ	Умовний перехід інструкція	3		✓	✓	✓	✓	✓	88
	COS	Інструкція для отримання COS з число з рухомою комою number	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	112

Інструкція		Функція інструкції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	DBCD	Інструкція для перетворення а подвійне слово до 32-бітного BCD коду	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	120
	DBIN	Інструкція з перетворення 32-бітовий BCD код у подвійне слово	7		✓	✓	✓	✓	✓	121
	DBITS	Інструкція з підрахунку на біт в подвійне слово	6		✓	✓	✓	✓	✓	195
	DCMP<	Дата < порівняння інструкція	7			✓	✓	✓	✓	153
	DCMP<=	Порівняння дати <= інструкція	7			✓	✓	✓	✓	153
	DCMP<>	Порівняння дати <> інструкція	7			✓	✓	✓	✓	153
	DCMP=	Дата = інструкція порівняння	7			✓	✓	✓	✓	153
	DCMP>	Дата > порівняння інструкція	7			✓	✓	✓	✓	153
	DCMP>=	Порівняння Дата >= інструкція	7			✓	✓	✓	✓	153
	DCNT	32-бітне інкрементування та інструкція декрементного підрахунку	7		✓	✓	✓	✓	✓	77
	DDEC	Довге ціле число мінус один інструкція	4	Нуль, перенос та позика	✓	✓	✓	✓	✓	105
	DDIV	Інструкція ділення довгих цілих чисел instruction	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	103
	DEC	Ціле число мінус один інструкція	3	Нуль, перенос та позика	✓	✓	✓	✓	✓	101
	DEG	Інструкція для кута в радіанах, вираженого числом з рухомою комою перетворення	7	Нуль, перенос та позика				✓	✓	116
	DECO	Декодування інструкції	5		✓	✓	✓	✓	✓	193
	DFLT	Інструкція з перетворення довгого цілого числа у число з рухомою комою номер	7	Нуль, перенос та позика			✓	✓	✓	118
	DFMOV	Подвійне слово блоку даних заповнення інструкція	9		✓	✓	✓	✓	✓	93
	DFROM	Інструкція для зчитування подвійного слова з регістра буфера спеціального модуля	10				✓	✓	✓	141
	DGBIN	Інструкція для перетворення 32-код Грея біта у подвійне слово	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	122
	DGRY	Інструкція з перетворення подвійного слова у 32-бітний код Грея код	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	121
	DHSCI	Інструкція для запуску переривання на основі порівняння високошвидкісного підрахунку	10		✓	✓	✓	✓	✓	157
	DHSCR	Інструкція для скидання налаштувань високошвидкісний підрахунок порівняння	10		✓	✓	✓	✓	✓	159
	DHSCS	Інструкція з налаштування високого-підрахунок швидкості порівняння	10		✓	✓	✓	✓	✓	156
	DHSP	Інструкція для імпульсного виходу на основі	10	513	✓	✓	✓	✓	✓	164

Інструкція		Функція інструкції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	DINT	Інструкція для перетворення числа з рухомою комою до довгого цілого числа	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	119
D	DIS	Інструкція з відокремлення 4 біти 16-бітових даних	7	Нуль, перенос та позика				✓	✓	247
	DIV	Інструкція цілочисельного ділення instruction	7		✓	✓	✓	✓	✓	99
	DMOV	Передача даних подвійним словом transmission instruction	7		✓	✓	✓	✓	✓	91
	DMUL	Множення довгих цілих чисел інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	103
	DNEG	Заперечення довгого цілого числа negation instruction	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	106
	DRCL	Інструкція для 32-бітного циклічного зсуву вліво з прапорцем переносу біт	9	Переносити	✓	✓	✓	✓	✓	136
	DRCR	Інструкція для 32-бітного циклічного зсуву праворуч з прапорцем переносу біт	9	Переносити	✓	✓	✓	✓	✓	135
	DROL	32-бітна циклічна операція зсуву вліво instruction	9	Переносити	✓	✓	✓	✓	✓	135
	DROR	32-бітна циклічна операція зсуву праворуч instruction	9	Переносити	✓	✓	✓	✓	✓	134
	DRVA	Керування абсолютним положенням control instruction	11	Нуль, перенос та позика	✓	✓	✓	✓	✓	230
	DRVI	Відносне положення керування instruction	11	Нуль, перенос та позика	✓	✓	✓	✓	✓	229
	DSHL	32-бітна інструкція логічного зсуву вліво	9		✓	✓	✓	✓	✓	138
	DSHR	32-бітна інструкція зсуву праворуч instruction	9		✓	✓	✓	✓	✓	138
	DSQT	Інструкція для вилучення цього квадратний корінь з довгого цілого числа	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	104
	DSUB	Віднімання довгих цілих чисел instruction	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	102
	DSUM	Довге ціле число накопичення instruction	9	Нуль, перенос та позика	✓	✓	✓	✓	✓	108
	DTI	Інструкція для перетворення довгого ціле число до цілого числа	6	Нуль, перенос та позика	✓	✓	✓	✓	✓	117
	DTO	Інструкція для написання подвійного слова з регістра буфера спеціального модуля	10				✓	✓	✓	143
	DUTY	Інструкція для генерування за часом імпульси	7					✓	✓	250
	DVABS	Інструкція для отримання the абсолютне значення довгого цілого числа	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	105
	DWAND	Подвійне слово AND instruction	10		✓	✓	✓	✓	✓	130
	DWINV	Подвійне слово заперечення instruction	10		✓	✓	✓	✓	✓	131
	DWOR	Подвійне слово OR instruction	10		✓	✓	✓	✓	✓	130
	DWXOR	Подвійне слово XOR instruction	10		✓	✓	✓	✓	✓	131
	DXCH	Інструкція обміну подвійних слів	7		✓	✓	✓	✓	✓	94

Інструкція		Функція інструкції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	EXP	Інструкція для отримання the натуральний степінь числа з рухомою комою	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	114
F	FIFO	Інструкція першим прийшов — першим обслужений	7		✓	✓	✓	✓	✓	95
	FLT	Інструкція для перетворення одного ціле число у число з рухомою комою number	6	Нуль, перенос та позика	✓	✓	✓	✓	✓	118
	FMOV	Інструкція заповнення блоку даних	7		✓	✓	✓	✓	✓	92
	FOR	Цикл інструкції	3		✓	✓	✓	✓	✓	85
	FROM	Інструкція для читання слів зі спеціального модуля буфера регістра	9				✓	✓	✓	141
G	GBIN	Інструкція для перетворення 16-бітовий код Грея у слово	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	122
	GRY	Інструкція для перетворення а слово до 16-бітового коду Грея code	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	121
H	HACKLE	Вихідний сигнал пилкоподібної форми сигналу інструкція	12		✓	✓	✓	✓	✓	179
	HCNT	Інструкція з керування височільник швидкості	7		✓	✓	✓	✓	✓	156
	HOUR	Інструкція хронографа	8		✓	✓	✓	✓	✓	152
	HTOS	Інструкція для перетворення година-хвилина-другі дані до секунд	5					✓	✓	155
I	INC	Ціле число плюс один інструкція	3	Нуль, перенос та позика	✓	✓	✓	✓	✓	100
	INTR	Інструкція з ініціалізації привода розширення регістр	5	Нуль, перенос та позика						223
	INT	Інструкція з перетворення числа з рухомою комою на одне ціле число	6	Нуль, перенос та позика	✓	✓	✓	✓	✓	118
	INV	Потік енергії заперечення інструкція	1		✓	✓	✓	✓	✓	157
	ITA	Інструкція для перетворення 16-бітові шістнадцяткові дані до ASCII коду	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	124
	ITD	Інструкція для перетворення одного ціле число у довге ціле число	6	Нуль, перенос та позика	✓	✓	✓	✓	✓	117
L	LBL	Визначення мітки переходу інструкція	3		✓	✓	✓	✓	✓	86
	LCNV	Інструкція з конвертації проекту	9	Нуль, перенос та позика			✓	✓	✓	125
	LD	NO контакт інструкція	1		✓	✓	✓	✓	✓	199
	LD<	Порівняння цілих чисел LD<інструкція	5		✓	✓	✓	✓	✓	199
	LD<=	Порівняння цілих чисел LD<=інструкція	5		✓	✓	✓	✓	✓	199
	LD<>	Порівняння цілих чисел LD<>інструкція	5		✓	✓	✓	✓	✓	199
	LD=	Порівняння цілих чисел LD=інструкція	5	515	✓	✓	✓	✓	✓	199

Інструкція		Функція інструкції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	LDD=	Порівняння довгих цілих чисел LD=інструкція	7		✓	✓	✓	✓	✓	202
	LDD>	Порівняння довгих цілих чисел LD>інструкція	7		✓	✓	✓	✓	✓	202
	LDD>=	Порівняння довгих цілих чисел LD>=інструкція	7		✓	✓	✓	✓	✓	202
	LDI	NC контакт інструкція	1		✓	✓	✓	✓	✓	59
	LDR<	Число з рухомою комою порівняння LD<інструкція	7		✓	✓	✓	✓	✓	205
	LDR<=	Число з рухомою комою порівняння LD<=інструкція	7		✓	✓	✓	✓	✓	205
	LDR<>	Число з рухомою комою порівняння LD<>інструкція	7		✓	✓	✓	✓	✓	205
	LDR=	Число з рухомою комою порівняння LD=інструкція	7		✓	✓	✓	✓	✓	205
	LDR>	Число з рухомою комою порівняння LD>інструкція	7		✓	✓	✓	✓	✓	205
	LDR>=	Число з рухомою комою порівняння LD>=інструкція	7		✓	✓	✓	✓	✓	205
	LIFO	Інструкція останнім прийшов — першим вийшов	7		✓	✓	✓	✓	✓	96
	LIMIT	Верхня/нижня межа керування інструкція	9	Нуль, перенос та позика				✓	✓	212
	LIN	Лінійна траєкторія інтерполяції інструкція	12	Нуль, перенос та позика				✓	✓	234
	LN	Інструкція для отримання the натуральний логарифм числа з рухомою комою	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	113
	LOADR	Інструкція для зчитування даних з файл розширення register	5	Нуль, перенос та позика						221
	LOG	Інструкція для отримання десяткового логарифма числа число з рухомою комою число	7	Нуль, перенос та позика				✓	✓	116
	LOGR	Інструкція для входу в систему розширення регістр	11	Нуль, перенос та позика						224
M	LRC	LRC перевірте інструкцію	7		✓	✓	✓	✓	✓	192
	MC	Основна інструкція керування	3		✓	✓	✓	✓	✓	70
	MCR	Інструкція скидання головного керування	1		✓	✓	✓	✓	✓	70
	MEAN	Значення інструкції	7	Нуль, перенос та позика				✓	✓	244
	Modbus	Головна станція Modbus станція інструкція зв'язку	8		✓	✓	✓	✓	✓	184
	MOV	Передача даних слова інструкція	5		✓	✓	✓	✓	✓	91
	MOVLINK	Інструкція з синхронного керування інструкція	17	Нуль, перенос та позика						238
	MPP	Стек потоку енергії виходу виштовхнути інструкція	1		✓	✓	✓	✓	✓	63
	MPS	Потік вихідної енергії натиснути інструкція	1		✓	✓	✓	✓	✓	62

Інструкція		Функція інструкції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	OR<	Порівняння цілих чисел OR<інструкція	5		✓	✓	✓	✓	✓	201
	OR<=	Порівняння цілих чисел OR<=інструкція	5		✓	✓	✓	✓	✓	201
	OR<>	Порівняння цілих чисел OR<>інструкція	5		✓	✓	✓	✓	✓	201
	OR=	Порівняння цілих чисел OR=інструкція	5		✓	✓	✓	✓	✓	201
	OR>	Порівняння цілих чисел OR>інструкція	5		✓	✓	✓	✓	✓	201
	OR>=	Порівняння цілих чисел OR>=інструкція	5		✓	✓	✓	✓	✓	201
	ORB	Блок потоку енергії або інструкція	1		✓	✓	✓	✓	✓	62
	ORD<	Порівняння довгих цілих чисел ORD<інструкція	7		✓	✓	✓	✓	✓	204
	ORD<=	Порівняння довгих цілих чисел ORD<=інструкція	7		✓	✓	✓	✓	✓	204
	ORD<>	Порівняння довгих цілих чисел ORD<>інструкція	7		✓	✓	✓	✓	✓	204
	ORD=	Порівняння довгих цілих чисел ORD=інструкція	7		✓	✓	✓	✓	✓	204
	ORD>	Порівняння довгих цілих чисел ORD>інструкція	7		✓	✓	✓	✓	✓	204
	ORD>=	Порівняння довгих цілих чисел ORD>=інструкція	7		✓	✓	✓	✓	✓	204
	ORI	NO контакт або інструкція	1		✓	✓	✓	✓	✓	61
	ORR<	Число з рухомою комою порівняння OR>інструкція	7		✓	✓	✓	✓	✓	207
	ORR<=	Число з рухомою комою порівняння OR<=інструкція	7		✓	✓	✓	✓	✓	207
	ORR<>	Число з рухомою комою порівняння OR<>інструкція	7		✓	✓	✓	✓	✓	207
	ORR=	Число з рухомою комою порівняння OR=інструкція	7		✓	✓	✓	✓	✓	207
	ORR>	Число з рухомою комою порівняння OR>інструкція	7		✓	✓	✓	✓	✓	207
	ORR>=	Число з рухомою комою порівняння OR>=інструкція	7		✓	✓	✓	✓	✓	207
P	OUT	Вихід котушки інструкція	1		✓	✓	✓	✓	✓	61
	OUT Sxx	SFC перехід стану інструкція	3		✓	✓	✓	✓	✓	72
	PID	PID інструкція інструкція	9		✓	✓	✓	✓	✓	174
	PLS	Імпульсний вихід лінії огибаючої інструкція	7			✓	✓	✓	✓	171
	PLSR	Інструкція для виходу імпульсного лічильника з прискорення/сповільнення	10		✓	✓	✓	✓	✓	168
	PLSV	Імпульсний вихід змінної швидкості інструкція	8	Нуль, перенос та позика	✓	✓	✓	✓	✓	228
	PLSY	Високошвидкісний імпульсний вихід інструкція	9		✓	✓	✓	✓	✓	167
	POWER	Інструкція щодо піднесення до степеня числа	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	113

Інструкція		Функція інструкції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	RADD	Додавання чисел з рухомою комою інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	108
	RAMP	Вихід сигналу рампи інструкція	12		✓	✓	✓	✓	✓	178
	RCL	Інструкція для 16-бітного циклічного зсуву вліво з прапорцем перенесення біт	7	Переносити	✓	✓	✓	✓	✓	134
	RCR	Інструкція для 16-бітного циклічного зсуву праворуч з прапорцем перенесення біт	7	Переносити	✓	✓	✓	✓	✓	133
	RCV	Приймання інструкції через вільний порт	7		✓	✓	✓	✓	✓	186
	RDIV	Ділення чисел з рухомою комою ділення інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	110
	REF	Інструкція для негайного виконання оновлення I/O	5		✓	✓	✓	✓	✓	144
	REFF	Інструкція з налаштування входу фільтрування константи	3		✓	✓	✓	✓	✓	144
	RET	SFC сегмент програми кінець інструкція	1		✓	✓	✓	✓	✓	73
	RLCNV	Проект з рухомою комою project інструкція з перетворення	12	Нуль, перенос та позика			✓	✓	✓	126
	RMOV	Дані з рухомою комою data інструкція з передачі інструкції	7		✓	✓	✓	✓	✓	92
	RMUL	Число з рухомою комою інструкція множення	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	109
	RND	Інструкція для генерування випадкові числа	3	Нуль				✓	✓	249
	RNEG	Заперечення числа з рухомою комою інструкція	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	111
	ROL	16-бітна циклічна операція зсуву вліво інструкція	7	Переносити	✓	✓	✓	✓	✓	132
	ROR	16-бітна циклічна операція зсуву праворуч інструкція	7	Переносити	✓	✓	✓	✓	✓	132
	RSQT	Інструкція для видобування квадратного кореня з числа з рухомою комою номер	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	110
	RST	Інструкція скидання котушки	1		✓	✓	✓	✓	✓	69
	RST Sxx	SFC скидання стану інструкція	3		✓	✓	✓	✓	✓	72
	RSUB	Число з рухомою комою інструкція віднімання	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	109
	RSUM	Число з рухомою комою інструкція накопичення	9	Нуль, перенос та позика	✓	✓	✓	✓	✓	114
	RVABS	Інструкція для отримання the абсолютне значення числа з рухомою комою	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	111
	RCMP	Інструкція з налаштування порівняння чисел з рухомою комою для ON	9				✓	✓	✓	210
	SAVER	Інструкція для запису даних у an файл розширення реєстру	7	Нуль, перенос та позика						222
	SCL	Інструкція з налаштування координат інструкція	7	Нуль, перенос та позика				✓	✓	214
	SEG	Інструкція для перетворення a	5	Нуль, перенос та 518 позика	✓	✓	✓	✓	✓	123

слово до 7-сегментного коду

Інструкція		Функція інструкції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
V	SPD VABS	Вимірювання частоти інструкція для отримання the абсолютне значення цілого числа	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	166
	SQT	Інструкція для вилучення the частоти для розчитування значення	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	100
	VRRD	Інструкція для зчитування значення	5	Нуль, перенос та позика						143
	STL	SFC стан запитання метра	3		✓	✓	✓	✓	✓	71
W	WAND	Слово AND інструкція	7		✓	✓	✓	✓	✓	127
	WOF	Інструкція з перетворення інструкція для скидання ерорів до однієї мс секунди великої програми	5		✓	✓	✓	✓	✓	185
	WINV STOP	Слово INV для функції користувача	5		✓	✓	✓	✓	✓	130
	WOR	Слово OR інструкція програма	7		✓	✓	✓	✓	✓	129
	WSEL STRADD	Інструкція зсуву слова вліво Комбінація рядків інструкція	9	Нуль, перенос та позика	✓	✓	✓	✓	✓	216
	WVNST R	Зсув рядка символів праворуч Почну рядка інструкція	9	Нуль, перенос та позика	✓	✓	✓	✓	✓	96
	WTOB STRLEFT	Розділення даних за байтами Інструкція для зчитування рядка інструкція	7	Нуль, перенос та позика				✓	✓	244
	WXOR	Слово XOR інструкція	7		✓	✓	✓	✓	✓	129
	XLEN	Інструкція для виявлення того довжина	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	247
	XMT	Надсилання інструкції через вільний порт для читання будь-якого	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	185
Z	STRZMR	Інструкція керування зоною символи рядка	7	Нуль, перенос та позика				✓	✓	219
	ZRN	Інструкція повернення до нуля	11	Нуль, перенос та позика	✓	✓	✓	✓	✓	227
	STRMIDW	Інструкція для заміни будь-якого	7	Нуль, перенос та позика				✓	✓	219
	ZRST	Інструкція зі скидання бітів у 0 символи рядка	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	193
	STRMOV	Інструкція з передачі рядка у пакетному режимі	5	Нуль, перенос та позика				✓	✓	221
	ZSET	Інструкція зі скидання бітів в пакет	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	193
Примітка:		Інструкція для зчитування рядка	7	Нуль, перенос та позика				✓	✓	217
T	SUB	Інструкція віднімання цілих чисел	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	99
	SUM	Цілочисельне накопичення інструкція	8	Нуль, перенос та позика	✓	✓	✓	✓	✓	106
	SWAP	MSB/LSB інструкція інструкція	3		✓	✓	✓	✓	✓	93
	TADD	Інструкція додавання годинника	7	Нуль та перенос		✓	✓	✓	✓	150
	TAN	Інструкція для отримання TAN з число з рухомою комою	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	112
	TCMP<	Інструкція порівняння часу	7			✓	✓	✓	✓	154
	TCMP<=	Порівняння Time >= інструкція	7			✓	✓	✓	✓	154
	TCMP<>	Порівняння часу <> інструкція	7			✓	✓	✓	✓	154
	TCMP=	Час = інструкція порівняння	7			✓	✓	✓	✓	154
	TCMP>	Час > інструкція порівняння	7			✓	✓	✓	✓	154
	TCMP>=	Порівняння Time >= інструкція	7			✓	✓	✓	✓	154
	TKY	Інструкція введення цифрових значень	7					✓	✓	146
	TMON	Неперезапускний моностабільний мультивібратор інструкція синхронізації	5		✓	✓	✓	✓	✓	75
	TO	Інструкція для запису слів із буфера спеціального модуля зареєструвати	9				✓	✓	✓	142
	TOF	Час затримки вимкнення інструкція	5		✓	✓	✓	✓	✓	74
	TON	Час затримки увімкнення інструкція	5		✓	✓	✓	✓	✓	73
	TONR	Затримка увімкнення типу	5	519	✓	✓	✓	✓	✓	73

Додаток I Показчик класифікації інструкцій таблиця

Інструкція		Інструкція функції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
Основні інструкції	LD	NO контакт інструкція	1		✓	✓	✓	✓	✓	59
	LDI	NC контакт інструкція	1		✓	✓	✓	✓	✓	59
	AND	NO контакт та інструкція	1		✓	✓	✓	✓	✓	59
	ANI	NC контакт та інструкція	1		✓	✓	✓	✓	✓	60
	OR	NO контакт або інструкція	1		✓	✓	✓	✓	✓	60
	ORI	NC контакт або інструкція	1		✓	✓	✓	✓	✓	61
	OUT	Вихід котушки інструкція	1		✓	✓	✓	✓	✓	61
	SET	Набір котушок інструкція	1		✓	✓	✓	✓	✓	69
	RST	Інструкція скидання котушки	1		✓	✓	✓	✓	✓	69
	ANB	Потік енергії блок та інструкція	1		✓	✓	✓	✓	✓	61
	ORB	Блок потоку енергії або інструкція	1		✓	✓	✓	✓	✓	62
	INV	Потік енергії заперечення інструкція	1		✓	✓	✓	✓	✓	69
	NOP	Без експлуатації інструкція	1		✓	✓	✓	✓	✓	70
	MPS	Вихідний потік енергії натисніть інструкцію	1		✓	✓	✓	✓	✓	62
	MRD	Інструкція для зчитування вершини стека потоку вихідної енергії значення	1		✓	✓	✓	✓	✓	63
	MPP	Потік вихідної енергії flow виштовхування зі стека інструкція	1		✓	✓	✓	✓	✓	63
	MC	Головне керування інструкція	3		✓	✓	✓	✓	✓	70
	MCR	Головне керування скидання інструкція	1		✓	✓	✓	✓	✓	70
	EU	Виявлення переднього фронту сигналу інструкція	2		✓	✓	✓	✓	✓	64
	ED	Задній фронт виявлення інструкція	2		✓	✓	✓	✓	✓	64
	TON	Затримка увімкнення часу інструкція	5		✓	✓	✓	✓	✓	73
	TOF	Затримка вимкнення часу інструкція	5		✓	✓	✓	✓	✓	74

	TMON	Неперезапускний моностабільна часова інструкція	5		✓	✓	✓	✓	✓	75
	TONR	Увімкнення ON за типом пам'яті час затримки інструкція	5		✓	✓	✓	✓	✓	73
	CTU	16-бітне інкрементування лічильник інструкція	5		✓	✓	✓	✓	✓	76
	CTR	16-бітний циклічний рахунок інструкція	5		✓	✓	✓	✓	✓	76

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	DCNT	32-бітне інкрементування та інструкція декрементного підрахунку	7		✓	✓	✓	✓	✓	77
Інструкції керування потоком виконання	LBL	Мітка переходу визначення інструкція	3		✓	✓	✓	✓	✓	86
	CJ	Умовний перехід інструкція	3		✓	✓	✓	✓	✓	88
	CALL	Інструкція для виклику користувацька підпрограма			✓	✓	✓	✓	✓	90
	CSRET	Інструкція для умовного повернення підпрограма	1		✓	✓	✓	✓	✓	90
	CFEND	Інструкція для умовного повернення головна програма користувача	1		✓	✓	✓	✓	✓	88
	CIRET	Інструкція для умовного повернення переривання користувача програма	1		✓	✓	✓	✓	✓	89
	FOR	Цикл інструкції	3		✓	✓	✓	✓	✓	85
	NEXT	Цикл повернення інструкція	1		✓	✓	✓	✓	✓	85
	WDT	Інструкція для скидання сторожового таймера користувацької програми	1		✓	✓	✓	✓	✓	88
	STOP	Інструкція для зупинки користувача програма	1		✓	✓	✓	✓	✓	89
	EI	Увімкнути переривання інструкція	1		✓	✓	✓	✓	✓	89
	DI	Вимкнути переривання інструкція	1		✓	✓	✓	✓	✓	89
SFC інструкції	STL	SFC стан завантаження інструкція	3		✓	✓	✓	✓	✓	71
	SET Sxx	SFC перехід стану інструкція	3		✓	✓	✓	✓	✓	72
	OUT Sxx	SFC стан переходу інструкція	3		✓	✓	✓	✓	✓	72
	RST Sxx	SFC стан скидання інструкція	3		✓	✓	✓	✓	✓	72
	RET	SFC сегмент програми кінець інструкція	1		✓	✓	✓	✓	✓	73
	MOV	Передача даних слова	5		✓	✓	✓	✓	✓	91

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	DXCH	Подвійне слово заміна інструкція	7		✓	✓	✓	✓	✓	94
	FMOV	Блок даних заповнення інструкція	7		✓	✓	✓	✓	✓	92
	DFMOV	Блок даних подвійний заповнення слова інструкція	9		✓	✓	✓	✓	✓	93
	WSFR	Зсув рядка символів праворуч інструкція	9	Нуль, перенос та позика	✓	✓	✓	✓	✓	96
	WSFL	Зсув рядка слів ліворуч інструкція	9	Нуль, перенос та позика	✓	✓	✓	✓	✓	97
	PUSH	Інструкція передачі даних	7		✓	✓	✓	✓	✓	94
	FIFO	Першим прийшов — першим вийшов інструкція	7		✓	✓	✓	✓	✓	95
	LIFO	Останнім прийшов — першим пішов інструкція	7		✓	✓	✓	✓	✓	96
Інструкції цілочисельних арифметичних операцій з довгим цілим числом	ADD	Цілочисельне додавання інструкція	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	98
	DADD	Додавання довгих цілих чисел addition інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	102
	SUB	Віднімання цілих чисел інструкція	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	99
	DSUB	Віднімання довгих цілих чисел інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	102
	INC	Ціле число плюс один інструкція	3	Нуль, перенос та позика	✓	✓	✓	✓	✓	100
	DINC	Довге ціле число плюс один інструкція	4	Нуль, перенос та позика	✓	✓	✓	✓	✓	104
	DEC	Ціле число мінус один інструкція	3	Нуль, перенос та позика	✓	✓	✓	✓	✓	101
	DDEC	Довге ціле число мінус одна інструкція	4	Нуль, перенос та позика	✓	✓	✓	✓	✓	105
	MUL	Цілочисельне множення інструкція	8	Нуль, перенос та позика	✓	✓	✓	✓	✓	99
	DMUL	Множення довгих цілих чисел multiplication інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	103
	DIV	Цілочисельне ділення інструкція	7		✓	✓	✓	✓	✓	99
	DDIV	Ділення довгих цілих чисел інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	103
	VABS	Інструкція з отримання абсолютного значення ціле число	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	101

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
Інструкції для виконання арифметичних операцій з рухомою комою	SUM	Цілочисельне накопичення інструкція	8	Нуль, перенос та позика	✓	✓	✓	✓	✓	106
	DSUM	Довге ціле число накопичення інструкція	9	Нуль, перенос та позика	✓	✓	✓	✓	✓	108
	RADD	Додавання чисел з плаваючою комою інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	108
	RSUB	Віднімання чисел з рухомою комою інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	109
	RMUL	Множення чисел з рухомою комою інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	109
	RDIV	Ділення чисел з рухомою комою інструкція	10	Нуль, перенос та позика	✓	✓	✓	✓	✓	110
	RVABS	Інструкція для отримання абсолютного значення а число з рухомою комою число	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	111
	RNEG	Рухома-кома заперечення числа інструкція	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	111
	RSQT	Інструкція для видобування квадратного кореня з числа з рухомою комою точка номер	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	110
	SIN	Інструкція для отримання SIN числа з рухомою комою	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	111
	COS	Інструкція для отримання COS числа з рухомою комою число	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	112
	TAN	Інструкція для отримання TAN числа з рухомою комою	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	112
	LN	Інструкція для отримання натурального логарифма числа а число з рухомою комою число	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	113
	EXP	Інструкція для отримання степеня натурального числа а число з рухомою комою число	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	114
	POWER	Інструкція щодо піднесення до степеня а	10	Нуль, перенос та позика 524	✓	✓	✓	✓	✓	113

Інструкція		Інструкція функції	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	RAD	Інструкція для числа з рухомою комою кут-радіан перетворення	7	Нуль, перенос та позика			✓	✓	✓	116
	DEG	Інструкція для числа з рухомою комою кут у радіанах перетворення	7	Нуль, перенос та позика			✓	✓	✓	116
	LOG	Інструкція для отримання десяткового логарифма числа з рухомою комою номер	7	Нуль, перенос та позика			✓	✓	✓	116
Інструкції логічних операцій зі словами/подвійними словами	WAND	Слово AND інструкція	7		✓	✓	✓	✓	✓	127
	DWAND	Подвійне слово AND інструкція	10		✓	✓	✓	✓	✓	130
	WOR	Слово OR інструкція	7		✓	✓	✓	✓	✓	129
	DWOR	Подвійне слово OR інструкція	10		✓	✓	✓	✓	✓	130
	WXOR	Слово XOR інструкція	7		✓	✓	✓	✓	✓	130
	DWXOR	Подвійне слово XOR інструкція	10		✓	✓	✓	✓	✓	131
	WINV	Слово INV інструкція	5		✓	✓	✓	✓	✓	130
	DWINV	Подвійне слово заперечення інструкції	7		✓	✓	✓	✓	✓	131
Інструкції з побітового зсуву та циклічного зсуву instructions	ROR	16-бітний циклічний зсув праворуч інструкція	7	Переносити	✓	✓	✓	✓	✓	132
	DROR	32-бітне циклічне зсування праворуч інструкція	9	Переносити	✓	✓	✓	✓	✓	134
	ROL	16-бітний циклічний зсув вліво інструкція	7	Переносити	✓	✓	✓	✓	✓	132
	DROL	32-бітний циклічний зсув вліво інструкція	9	Переносити	✓	✓	✓	✓	✓	135
	RCR	Інструкція для 16-бітного циклічного зсуву праворуч через прапор переносу прапорець біта	7	Переносити	✓	✓	✓	✓	✓	133
	DRCR	Інструкція для 32-бітного циклічного зсуву праворуч через прапор переносу	9	Переносити	✓	✓	✓	✓	✓	135
	RCL	Інструкція для 16-бітного циклічного зсуву ліворуч через прапорець переносу	7	Переносити	✓	✓	✓	✓	✓	134
	DRCL	Інструкція для циклічного зсуву вліво на 32 біти з переносом прапорець біта	9	Переносити	✓	✓	✓	✓	✓	136

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
Покращені інструкції обробки бітів інструкцій	ENCO	Кодування інструкції	5		✓	✓	✓	✓	✓	194
	BITS	Інструкція з підрахунку бітів у слово	5		✓	✓	✓	✓	✓	194
	DBITS	Інструкція для підрахунку встановлених бітів у подвійному слові	6		✓	✓	✓	✓	✓	195
	ZRST	Інструкція для скидання бітів у 0 у пакеті	5		✓	✓	✓	✓	✓	193
	ZSET	Інструкція для скидання бітів у пакеті	5		✓	✓	✓	✓	✓	193
	BON	Інструкція для оцінювання біт ON у слові	7					✓	✓	195
Високошвидкісні I/O інструкції	HCNT	Високошвидкісний лічильник інструкція привода	7		✓	✓	✓	✓	✓	156
	DHSCS	Набір порівняння високошвидкісного підрахунку інструкція	10		✓	✓	✓	✓	✓	156
	DHSCR	Високошвидкісний рахунок скидання порівняння інструкція	10		✓	✓	✓	✓	✓	159
	DHSCI	Високошвидкісний рахунок інструкція тригера переривання порівняння	10		✓	✓	✓	✓	✓	157
	DHSZ	Діапазон високошвидкісного підрахунку порівняння інструкція	13		✓	✓	✓	✓	✓	161
	DHST	Таблиця високошвидкісного підрахунку порівняння інструкція	10		✓	✓	✓	✓	✓	162
	DHSP	Інструкція для імпульсного виходу на основі високошвидкісного підрахунку таблиця порівняння	10		✓	✓	✓	✓	✓	164
	SPD	Частота інструкція з вимірювання	7		✓	✓	✓	✓	✓	166
	PLSY	Високошвидкісний імпульс інструкція	9		✓	✓	✓	✓	✓	167
	PLSR	Інструкція для виходу імпульсного лічильника з прискоренням/уповільненням	10		✓	✓	✓	✓	✓	168

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	DABSD	Подвійне слово кулачок абсолютне керування інструкція	11	Нуль, перенос та позика			✓	✓	✓	183
	ALT	Альтернативний вихід інструкція	3	Нуль, перенос та позика	✓	✓	✓	✓	✓	184
Периферійні інструкції	FROM	Інструкція для зчитування слів зі спеціального модуля буфера зареєструвати	9				✓	✓	✓	141
	DFROM	Інструкція для зчитування подвійних слів із буфера спеціального модуля зареєструвати	10				✓	✓	✓	141
	TO	Інструкція для запису слів із буфера спеціального модуля зареєструвати	9				✓	✓	✓	142
	DTO	Інструкція для запису подвійних слів зі спеціального модуля буфера зареєструвати	10				✓	✓	✓	143
	VRRD	Інструкція з читання значення аналогового сигналу потенціометр	5							143
	REFF	Інструкція з налаштування фільтрація вхідного сигналу константа	3		✓	✓	✓	✓	✓	144
	REF	Інструкція для негайного виконання оновлення I/O	5		✓	✓	✓	✓	✓	144
	EROMWR	EEPROM записати інструкція	7							145
	PR	Друк інструкції	5				✓	✓	✓	145
	TKY	Цифрова клавіша вводу інструкція	7				✓	✓	✓	146
	ZRN	Нуль повернення інструкція	11	Нуль, перенос та позика	✓	✓	✓	✓	✓	227
	PLSV	Змінна швидкість імпульсу інструкція	8	Нуль, перенос та позика	✓	✓	✓	✓	✓	228
	DRVI	Відносне положення інструкція керування	11	Нуль, перенос та позика	✓	✓	✓	✓	✓	229
	DRVA	Абсолютна позиція інструкція керування	11	Нуль, перенос та позика	✓	✓	✓	✓	✓	230

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	MOVLINK	Синхронне керування інструкція	17	Нуль, перенос та позика						238
Інструкції годинника реального часу ns	TRD	Зчитування годинника реального часу read інструкція	3			✓	✓	✓	✓	148
	TWR	Годинник реального часу запис інструкція	3			✓	✓	✓	✓	149
	TADD	Годинник додавання інструкція	7	Нуль та перенос		✓	✓	✓	✓	150
	TSUB	Віднімання годинника інструкція	7	Нуль та позика		✓	✓	✓	✓	151
	HOURL	Хронограф інструкція	8			✓	✓	✓	✓	152
	HTOS	Інструкція з перетворення даних у форматі години-хвилини-секунди до секунд	5				✓	✓	✓	155
	STOH	Інструкція з перетворення секунд у години-хвилини-другі дані	5				✓	✓	✓	155
Інструкції щодо контактів для порівняння	LD=	Порівняння цілих чисел LD= інструкція	5		✓	✓	✓	✓	✓	199
	LDD=	Порівняння довгих цілих чисел LD= інструкція	7		✓	✓	✓	✓	✓	202
	LDR=	Порівняння чисел з рухомою комою LD= інструкція	7		✓	✓	✓	✓	✓	205
	LD>	Порівняння цілих чисел LD> інструкція	5		✓	✓	✓	✓	✓	199
	LDD>	Порівняння довгих цілих чисел LD> інструкція	7		✓	✓	✓	✓	✓	202
	LDR>	Порівняння чисел з рухомою комою LD> інструкція	7		✓	✓	✓	✓	✓	205
	LD>=	Порівняння цілих чисел LD>= інструкція	5		✓	✓	✓	✓	✓	199
	LDD>=	Довге ціле число порівняння LD>= інструкція	7		✓	✓	✓	✓	✓	202
Інструкції щодо контактів	LD<	Порівняння цілих чисел LD<інструкція	5		✓	✓	✓	✓	✓	199
	LDD<	Порівняння довгих цілих чисел LD< інструкція	7		✓	✓	✓	✓	✓	202
	LDR<	Рухома-кома інструкція LD<	7		✓	✓	✓	✓	✓	205

контактів

порівняння чисел

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	LDD<>	Довге ціле число порівняння LD<> інструкція	7		✓	✓	✓	✓	✓	202
	LDR<>	Порівняння чисел з рухомою комою LD інструкція	7		✓	✓	✓	✓	✓	205
	AND=	Порівняння цілих чисел AND= інструкція	5		✓	✓	✓	✓	✓	200
	ANDD=	Порівняння довгих цілих чисел AND= інструкція	7		✓	✓	✓	✓	✓	203
	ANDR=	Порівняння чисел з рухомою комою AND= інструкція	7		✓	✓	✓	✓	✓	206
	AND>	Порівняння цілих чисел AND> інструкція	5		✓	✓	✓	✓	✓	200
	ANDD>	Порівняння довгих цілих чисел AND> інструкція	7		✓	✓	✓	✓	✓	203
	ANDR>	Рухома-кома інструкція порівняння чисел AND>	7		✓	✓	✓	✓	✓	206
	AND>=	Порівняння цілих чисел AND>= інструкція	5		✓	✓	✓	✓	✓	200
	ANDD>=	Довге ціле число порівняння AND>= інструкція	7		✓	✓	✓	✓	✓	203
	ANDR>=	Рухома-кома інструкція порівняння чисел AND>=	7		✓	✓	✓	✓	✓	206
	AND<	Порівняння цілих чисел AND< інструкція	5		✓	✓	✓	✓	✓	200
	ANDD<	Порівняння довгих цілих чисел AND< інструкція	7		✓	✓	✓	✓	✓	203
	ANDR<	Рухома-кома інструкція порівняння чисел AND<	7		✓	✓	✓	✓	✓	206
	AND<=	Порівняння цілих чисел AND<= інструкція	5		✓	✓	✓	✓	✓	200

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
Інструкції щодо контактів для порівняння	AND<>	Порівняння цілих чисел AND<> інструкція	5		✓	✓	✓	✓	✓	200
	ANDD<>	Порівняння довгих цілих чисел AND<> інструкція	7		✓	✓	✓	✓	✓	203
	ANDR<>	Порівняння чисел з рухомою комою AND<> інструкція	7		✓	✓	✓	✓	✓	206
	OR=	Порівняння цілих чисел OR= інструкція	5		✓	✓	✓	✓	✓	201
	ORD=	Довге ціле число порівняння OR= інструкція	7		✓	✓	✓	✓	✓	204
	ORR=	Рухома-кома порівняння чисел OR= інструкція	7		✓	✓	✓	✓	✓	207
	OR>	Порівняння цілих чисел OR> інструкція	5		✓	✓	✓	✓	✓	201
	ORD>	Довге ціле число порівняння OR> інструкція	7		✓	✓	✓	✓	✓	204
	ORR>	Порівняння чисел з рухомою комою OR> інструкція	7		✓	✓	✓	✓	✓	207
	OR>=	Порівняння цілих чисел OR>= інструкція	5		✓	✓	✓	✓	✓	201
	ORD>=	Довге ціле число порівняння OR>= інструкція	7		✓	✓	✓	✓	✓	204
	ORR>=	Порівняння чисел з рухомою комою OR>= інструкція	7		✓	✓	✓	✓	✓	207
	OR<	Порівняння цілих чисел ORінструкція	5		✓	✓	✓	✓	✓	201
	ORD<	Порівняння довгих цілих чисел OR< інструкція	7		✓	✓	✓	✓	✓	204
	ORR<	Рухома-кома інструкція порівняння чисел OR>	7		✓	✓	✓	✓	✓	207
	OR<=	Порівняння цілих чисел OR<= інструкція	5		✓	✓	✓	✓	✓	201
	ORD<=	Порівняння довгих цілих чисел OR<= інструкція	7		✓	✓	✓	✓	✓	204
	ORR<=	Рухома-кома інструкція порівняння чисел OR<=	7				✓	✓	✓	207
	OR<>	Порівняння цілих чисел OR<> інструкція	5		✓	✓	✓	✓	✓	201
		Порівняння								

Інструкція		функція	Крок довжина	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
	LCMP	Інструкція з налаштування порівняння довгого цілого числа з ON	9				✓	✓	✓	209
	RCMP	Інструкція для налаштування числа з плаваючою комою number порівняння з ON	9				✓	✓	✓	210
Інструкції з перетворення значень	ITD	Інструкція для перетворення цілого числа на довге ціле число	6	Нуль, перенос та позика	✓	✓	✓	✓	✓	117
	DTI	Інструкція для перетворення довгого цілого числа на ціле число	6	Нуль, перенос та позика	✓	✓	✓	✓	✓	117
	FLT	Інструкція з перетворення цілого числа у число з плаваючою комою номер	6	Нуль, перенос та позика	✓	✓	✓	✓	✓	118
	DFLT	Інструкція щодо перетворення цілого числа великого розміру у число з рухомою комою точка номер	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	118
	INT	Інструкція з перетворення числа з рухомою комою на ціле число	6	Нуль, перенос та позика	✓	✓	✓	✓	✓	118
	DINT	Інструкція для перетворення числа з рухомою комою на довге ціле ціле число	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	119
	BCD	Інструкція з перетворення слова на 16-бітний BCD код	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	119
	DBCD	Інструкція з перетворення подвійного слова на 32-бітне BCD код	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	120
	BIN	Інструкція для перетворення 16-бітного коду BCD на слово	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	120
	DBIN	Інструкція з перетворення 32-бітного коду BCD на подвійне слово	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	121
	GRY	Інструкція з перетворення слова на 16-бітний код Грея	5	Нуль, перенос та позика	✓	✓	✓	✓	✓	121
	DGRY	Інструкція з перетворення подвійного слова у	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	121

Інструкція		функція	Крок довжи на	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторін ка
	ASC	ASCII код Інструкція з перетворення	19	Нуль, перенос та позика	✓	✓	✓	✓	✓	123
	ITA	Інструкція з перетворення 16- бітових шістнадцяткових даних на ASCII код	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	124
	ATI	Інструкція з перетворення коду ASCII у 16-бітний шістнадцятковий дані	7	Нуль, перенос та позика	✓	✓	✓	✓	✓	124
	LCNV	Проект перетворення інструкція	9	Нуль, перенос та позика			✓	✓	✓	125
	RLCNV	Перетворення проєкту з рухомою комою conversion інструкція	12	Нуль, перенос та позика			✓	✓	✓	126
Слово інструкції з контактув ання	BLD	Бітовий контакт слова LD інструкція	5		✓	✓	✓	✓	✓	195
	BLDI	Бітовий контакт слова LDI інструкція	5		✓	✓	✓	✓	✓	196
	BAND	Бітовий контакт слова AND інструкція	5		✓	✓	✓	✓	✓	196
	BANI	Бітовий контакт слова ANI інструкція	5		✓	✓	✓	✓	✓	197
	BOR	Бітовий контакт слова OR інструкція	5		✓	✓	✓	✓	✓	197
	BORI	Бітовий контакт слова ORI інструкція	5		✓	✓	✓	✓	✓	198
	BSET	Біт слова котушки встановити інструкція	5		✓	✓	✓	✓	✓	198
	BRST	Скидання біта котушки reset інструкція	5		✓	✓	✓	✓	✓	199
	BOUT	Слово біт котушка вихід інструкція	5		✓	✓	✓	✓	✓	198
Інструкції з комунікаці її	Modbus	Головна станція зв'язку інструкція	8		✓	✓	✓	✓	✓	184
	XMT	Вільний порт передавання інструкція	7		✓	✓	✓	✓	✓	185
	RCV	Прийом у вільному порту receiving інструкція	7		✓	✓	✓	✓	✓	186
	MODRW	Modbus читання/запис інструкція	14		✓	✓	✓	✓	✓	187
Перевірт е інструк ції	CCITT	CCITT перевірити інструкція	7		✓	✓	✓	✓	✓	191
	CRC16	CRC16 перевірити інструкція	7		✓	✓	✓	✓	✓	191
	LRC	LRC перевірка інструкція	7		✓	✓	✓	✓	✓	192

Інструкція		функція	Крок довжини	Прапорець впливу біт	VC1S	VC1	VC2	VC3	VC5	Сторінка
Інструкції з порівняння часу	DCMP<>	Інструкція для порівняння часу зчитування будь-яких символів рядка	7	Нуль, перенос та позика		✓	✓	✓	✓	153
	STRMIDR	Інструкція для заміни будь-якого символу рядка	7	Нуль, перенос та позика		✓	✓	✓	✓	219
	TCMP=	Інструкція для порівняння часу зчитування будь-яких символів рядка	7	Нуль, перенос та позика		✓	✓	✓	✓	154
	TCMP>	Інструкція для порівняння часу зчитування будь-яких символів рядка	7	Нуль, перенос та позика		✓	✓	✓	✓	154
	STRMIDW	Інструкція для заміни будь-якого символу рядка	7	Нуль, перенос та позика		✓	✓	✓	✓	219
	TCMP<	Інструкція для порівняння часу зчитування будь-яких символів рядка	7	Нуль, перенос та позика		✓	✓	✓	✓	154
	TCMPSTR	Інструкція для порівняння часу зчитування будь-яких символів рядка	7	Нуль, перенос та позика		✓	✓	✓	✓	250
Регістр файла інструкції з обробки даних	TCMP<	Інструкція для порівняння часу зчитування будь-яких символів рядка	7	Нуль, перенос та позика		✓	✓	✓	✓	154
	LOADR	Інструкція для завантаження даних із файлу	5	Нуль, перенос та позика		✓	✓	✓	✓	221
	MEAN	Інструкція для розширення файлу	7	Нуль, перенос та позика				✓	✓	244
	WTOB	Інструкція для розширення файлу	7	Нуль, перенос та позика				✓	✓	244
	SAVER	Інструкція для розширення файлу	7	Нуль, перенос та позика				✓	✓	222
	BTOW	Інструкція для розширення файлу	7	Нуль, перенос та позика				✓	✓	245
	INITR	Інструкція для ініціалізації реєстру	5	Нуль, перенос та позика				✓	✓	223
	UNGR	Інструкція для розширення реєстру	7	Нуль, перенос та позика				✓	✓	246
	DIS	Інструкція для розширення реєстру	7	Нуль, перенос та позика				✓	✓	247
	ITER	Інструкція для розширення реєстру	7	Нуль, перенос та позика				✓	✓	225
Інша інструкція	ANS	Інструкція для розширення реєстру	7	Нуль, перенос та позика				✓	✓	248
	RND	Інструкція для розширення реєстру	3	Нуль, перенос та позика				✓	✓	249
	ANR	Інструкція для розширення реєстру	1	Нуль, перенос та позика				✓	✓	249
Інструкції з обробки масивів даних	DUTY	Інструкція для розширення реєстру	7	Нуль, перенос та позика				✓	✓	250
	BRADD	Інструкція для розширення реєстру	9	Нуль, перенос та позика				✓	✓	210
	BKSUB	Інструкція для розширення реєстру	9	Нуль, перенос та позика				✓	✓	211
Технічний паспорт інструкції	BKCMPE, >, <, <=, >=	Інструкція порівняння масивів даних	9	Нуль, перенос та позика				✓	✓	211
	LIMIT	Верхня/нижня межа інструкції з керування	9	Нуль, перенос та позика				✓	✓	212
	DBAND	Керування зоною нечутливості інструкції	9	Нуль, перенос та позика				✓	✓	213
	ZONE	Керування зоною інструкції	9	Нуль, перенос та позика				✓	✓	213
	SCL	Налаштування координат інструкції	7	Нуль, перенос та позика				✓	✓	214
Інструкції	SER	Пошук даних інструкції	9	Нуль, перенос та позика				✓	✓	215
	STRADD	комбінація інструкції	7	Нуль, перенос та позика				✓	✓	216

виявлення довжини