MITSUBISHI ELECTRIC
Changes for the Better

พื้นฐานการเขียนโปรแกรม
PLC

หลักสูตรนี้มีไว้สำหรับผู้เข้าร่วมที่จะสร้างโปรแกรมควบคุมสำหรับ PLC เป็นครั้งแรก

Copyright ©2015 Mitsubishi Electric Corporation. All Rights Reserved.

บทนำ วัตถุประสงค์ของหลักสูตร

หลักสูตรนี้จะอธิบายเกี่ยวกับการเขียนโปรแกรม ซึ่งอาจสามารถใช้สำหรับ PLC MELSEC หนึ่งในภาษาโปรแกรมหลักคือแผนผังแลดเดอร์ (LD) หลักสูตรนี้จะครอบคลุมประเด็นสำคัญเกี่ยวกับการเขียนโปรแกรมแลดเดอร์ รวมถึงคำสั่งหลัก

บางส่วนของหลักสูตรนี้มีการอ้างอิงมาจากหลักสูตรพื้นฐานเกี่ยวกับ PLC MELSEC ขอแนะนำให้เรียนหลักสูตรพื้นฐานที่เกี่ยวของกอนเรียนหลักสูตรนี้

บทนำ โครงสร้างหลักสูตร

เนื้อหาของหลักสูตรนี้มีดังนี้

บทที่ 1 - การเขียนโปรแกรม PLC

บทนี้จะอธิบายถึงประเด็นสำคัญของการเขียนโปรแกรมแลดเดอร์

บทที่ 2 - คำสั่งบิทดีไวส์

บทนี้จะอธิบายเกี่ยวกับคำสั่งที่มีบิทดีไวส์ (ON/OFF)

บทที่ 3 - คำสั่งเวิร์ดดีไวส์

บทนี้จะอธิบายเกี่ยวกับคำสั่งที่มีเวิร์ด (ตัวเลข) ดีไวส์

บทที่ 4 - คำสั่งการบรานช์โปรแกรม

บทนี้จะอธิบายเกี่ยวกับคำสั่งที่ใช้สร้างบรานช์โปรแกรม

แบบทดสอบประเมินผล

เกณฑ์การผ่าน: ต้องได้ 60% ขึ้นไป

บทนำ วิธีการใช้งานเครื่องมือการเรียนรู้อิเล็กทรอนิกส์นี้

ไปที่หน้าถัดไป		ไปที่หน้าถัดไป
กลับไปยังหน้าที่แล้ว		กลับไปยังหน้าที่แล้ว
เลื่อนไปยังหน้าที่ต้องการ		ระบบจะแสดง "สารบัญ" ช่วยให้คุณสามารถไปยังหน้าต่างๆ ได้
ออกจากการเรียนรู้		ออกจากการเรียนรู้

บทนำ ข้อควรระวังในการใช้งาน

ข้อควรระวังด้านความปลอดภัย

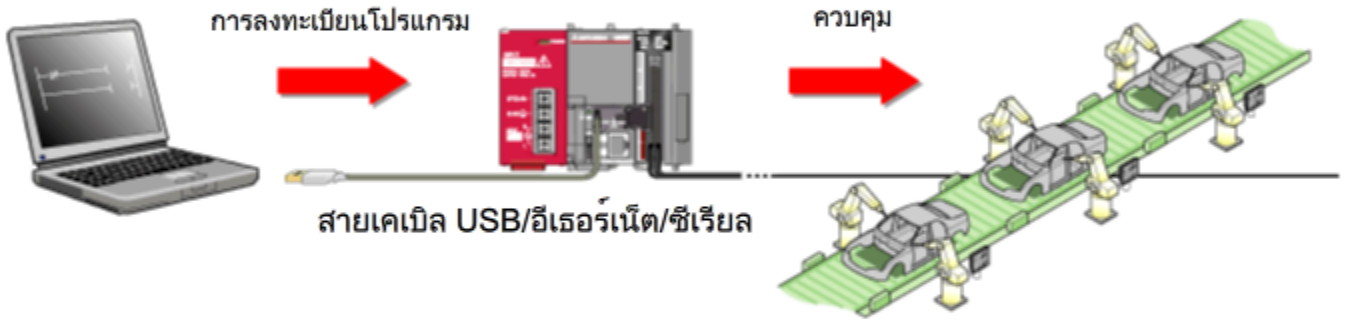
เมื่อคุณเรียนรู้ตามการใช้งานผลิตภัณฑ์จริง โปรดอ่านข้อควรระวังด้านความปลอดภัยในคู่มือที่เกี่ยวข้องอย่างละเอียด

บทที่ 1 โปรแกรมควบคุม

การทำงานที่ PLCดำเนินการ จะถูกเขียนไว้เป็นโปรแกรมควบคุม
โปรแกรมเหล่านี้จะถูกลงทะเบียนลงในโมดูล CPU ซึ่งทำการควบคุมสัญญาณอินพุทและเอาท์พุท (I/O)
ภาษาโปรแกรมที่ใช้สำหรับ PLC ได้แก่ แลดเดอร์ อินสตรักชั่นลิสต์ (IL) และซีเควนเชียลฟังก์ชั่นชาร์ท(SFC)

หลักสูตรนี้จะอธิบายถึงประเด็นสำคัญเกี่ยวกับการเขียนโปรแกรมแลดเดอร์รวมถึงคำสั่งหลัก

พีซี (ซอฟต์แวร์สำหรับแก้ไขโปรแกรมโปรแกรม)	PLC	สายการประกอบ ยานยนต์

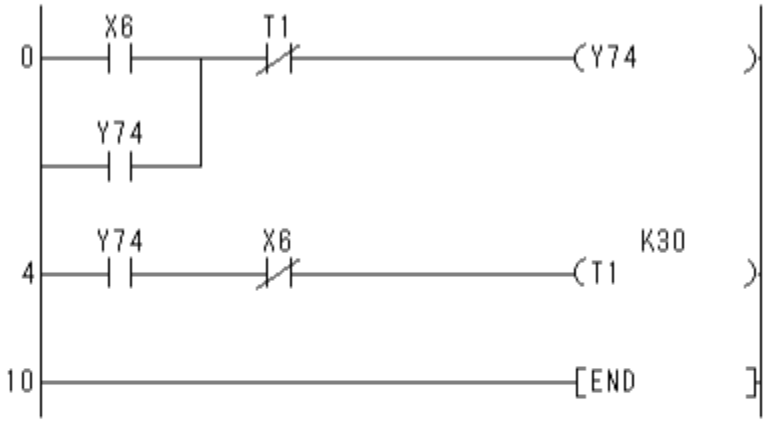


ในหลักสูตรนี้ จะมีการใช้ซอฟต์แวร์สำหรับแก้ไขโปรแกรม PLC GX Works2 หรือ GX Works3 ในการสร้างโปรแกรม
เพื่อเรียนรู้วิธีการใช้ซอฟต์แวร์สำหรับแก้ไขโปรแกรม PLC โปรดเรียนหลักสูตร "พื้นฐาน GX Works2" หรือ "Engineering Software MELSOFT GX Works3 (Ladder)" (ซอฟต์แวร์สำหรับแก้ไขโปรแกรม MELSOFT GX Works3 (แลดเดอร์))
GX Works2 จะรองรับ MELSEC-Q/L/F ซีรีส์
GX Works3 จะรองรับ MELSEC iQ-R/iQ-F ซีรีส์

1.1 ภาษาโปรแกรม

ภาษาโปรแกรมที่ใช้สำหรับ PLC ได้แก่ แลดเดอร์ อินสตรักชั่นลิสต์ (IL) และซีเควนเชียลฟังก์ชั่นชาร์ท(SFC)

โปรแกรมแลดเดอร์เป็นแผนผังโลจิกแบบกราฟิกที่อิงตามวงจรไฟฟ้า ในโปรแกรมแลดเดอร์ สัญลักษณ์ที่แทนคำสั่งจะเชื่อมต่อกับเส้น คล้ายกับแผนผังวงจร และจะสามารถจำแนกลำดับการทำงานได้อย่างง่ายดาย
นอกจากนี้ การเขียนโปรแกรมแลดเดอร์จะไม่จำเป็นต้องใช้ความรู้ด้านการเขียนโปรแกรมพิเศษ เช่น ภาษา C และ BASIC และสามารถทำความเข้าใจได้ไม่ยาก สำหรับผู้ที่มีประสบการณ์ทำงานเกี่ยวกับวงจรไฟฟ้า และรีเลย์



ตารางด้านล่างจะแสดงให้เห็นโปรแกรมเดียวกันใน IL
IL ต้องใช้ความรู้ด้านการเขียนโปรแกรมบางส่วน เพื่อแสดงการทำงานในรูปแบบของคำสั่ง

หมายเลขขั้นตอน	คำสั่ง	ดีไวส์
0	LD	X6
1	OR	Y74
2	ANI	T1
3	OUT	Y74
4	LD	Y74
5	ANI	X6
6	OUT	T1 K30
10	END	

1.2 ค่าที่ใช้ในโปรแกรม

โปรแกรมสำหรับ PLCสามารถจัดการค่าได้สองชนิด

บิท	บิทจะถูกแสดงในรูปแบบสัญญาณไฟฟ้าสองชนิด ON และ OFF ซึ่งจะสามารถแสดงผลเป็น "1" (ON) และ "0" (OFF) ได้ด้วย ค่าบิทมักถูกนำไปใช้ในการแสดงสถานะของI/O ดีไวส์ เช่น สวิตช์ และหลอดไฟ
เวิร์ด	ตัวเลขและตัวอักษร ค่าเวิร์ดมักถูกนำไปใช้เพื่อแสดงปริมาณและเวลา *หลักสูตรนี้จะอธิบายเกี่ยวกับตัวเลขเพียงอย่างเดียวเท่านั้น สำหรับรายละเอียดเกี่ยวกับตัวอักษรที่ใช้และค่าเวิร์ด โปรดดูที่ คู่มือการใช้งานผลิตภัณฑ์ที่เกี่ยวข้อง

รูปแบบตัวเลขต่อไปนี้จะถูกนำไปใช้ในการแสดง ค่า

- ค่าฐานสิบ
- ค่าฐานสอง
- ค่าฐานสิบหก
- ค่าฐานแปด

1.2.1 สัญลักษณ์ค่าฐานสิบ

ในสัญลักษณ์ค่าฐานสิบ ขนาด (ปริมาณ) ของตัวเลข จะถูกแสดงโดยใช้รูปแบบฐาน 10 จาก "0 ถึง 9"

ใน PLC MELSEC เลขค่าฐานสิบจะมีตัวอักษร "K" นำหน้า
ตัวอย่างเช่น "K153" จะหมายถึงลขค่าฐานสิบ "153"

1.2.2 สัญลักษณ์ค่าฐานสอง

ในขณะที่ ตามปกติแล้วจะมีการใช้สัญลักษณ์ค่าฐานสิบในการแสดงปริมาณและเวลา PLC และ PC(คอมพิวเตอร์) จะใช้ข้อมูลฐานสอง ซึ่งจะเป็นการทำงานร่วมกันระหว่าง "0" และ "1"
ตารางด้านล่างจะแสดงความสัมพันธ์ระหว่างค่าฐานสิบและค่าฐานสอง ไปจนถึงเลขค่าฐานสิบ "8"

ค่าฐานสิบ	ค่าฐานสอง
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
⋮	⋮

ทุกครั้งที่มีการใช้คำสั่ง 1 เวิร์ดในโปรแกรม คำสั่งจะถูกบันทึกและประมวลผลเป็นข้อมูลฐานสอง 16 บิท ผ่าน PLCตัวจริง ข้อมูลฐานสอง 16 บิทนี้ จะเท่ากับ "1 เวิร์ด"
ตัวอย่างเช่น ค่าฐานสิบ "157" จะถูกแสดงเป็น "0000000010011101" ในค่าฐานสอง

ในสัญลักษณ์ค่าฐานสิบ บิทจะถูกเขียนจากทางด้านขวา (บิททางขวาสุดจะเป็นบิทเริ่มต้น)

b15	～	b8	b7	b6	b5	b4	b3	b2	b1	b0	← ตำแหน่งบิท
0	・・・	0	1	0	0	1	1	1	0	1	← ค่าฐานสอง
2^{15}		2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	← ยกกำลัง 2
32768		256	128	64	32	16	8	4	2	1	← ค่าถ่วงน้ำหนักสำหรับค่าฐานสิบ

ในการแปลงค่าฐานสองเป็นค่าฐานสิบ ให้คูณแต่ละสถานะบิท ("0" หรือ "1") ด้วยค่าถ่วงน้ำหนักที่เหมาะสม และรวมผลคูณทั้งหมด

$$= \underline{1\times128} + 0\times64 + 0\times32 + \underline{1\times16} + 1\times8 + \underline{1\times4} + 0\times2 + \underline{1\times1}$$
$$= 128 + 16 + 8 + 4 + 1$$
$$= 157$$

จากด้านบน จะถือได้ว่าค่าฐานสองเป็นรูปแบบการนับที่อิงตามค่าถ่วงน้ำหนัก

1.2.3 สัญลักษณ์ค่าฐานสิบหก

ในสัญลักษณ์ค่าฐานสิบหก ขนาด (ปริมาณ) ของตัวเลขจะถูกแสดงโดยใช้ฐาน 16 หรือระบุด้วยตัวอักษรและตัวเลข 16 ตัว: 0 ถึง 9 และ A ถึง F แต่ละหลักในสัญลักษณ์ค่าฐานสิบหกจะเพิ่มขึ้นโดยเรียงจาก 0, 1...9, A...E จนถึง F เมื่อค่าเกินฐาน "F" ค่าดังกล่าวจะถูกย้ายไปทางซ้าย และกลายเป็น "10"
ใน PLC MELSEC เลขค่าฐานสิบหกจะมี "H" นำหน้า
ตัวอย่างเช่น "H4A9D" จะหมายถึงลขค่าฐานสิบหก "4A9D"

สัญลักษณ์ค่าฐานสองอาจยาวและยุ่งยากในการใช้งานภายในโปรแกรม และบนจอแสดงผลการตรวจสอบ
ในกรณีดังกล่าว สัญลักษณ์ค่าฐานสิบหกจะมีประโยชน์
หนึ่งหลักของค่าฐานสิบหกจะสามารถแสดงค่าฐานสองได้ 4 บิท (4 หลัก)

ภาพด้านล่างจะแสดงวิธีการแสดงค่าฐานสิบหกเป็นค่าฐานสิบ

4	3	2	1	←จำนวนหลัก
4	A	9	D	←ค่าฐานสิบหก
16^3	16^2	16^1	16^0	←ยกกำลัง 16

$$= 4 \times 16^3 + A \times 16^2 + 9 \times 16^1 + D \times 16^0$$
$$(4096)\ (10)\ (256)\quad (16)\ (13)\ (1)$$
$$= 19101$$

* ค่าฐานสิบหกหนึ่งหลักสามารถแสดงค่าฐานสองได้ 4 บิท

ค่าฐานสิบ	ค่าฐานสอง	ค่าฐานสิบหก
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	10000	10
17	10001	11
18	10010	12
•	•	•
•	•	•
•	•	•
19101	0100 1010 1001 1101	4 A 9 D

1.2.4 สัญลักษณ์ค่าฐานแปด

ในสัญลักษณ์ค่าฐานแปด ขนาด (ปริมาณ) ของตัวเลข ถูกแสดงโดยใช้รูปแบบฐาน 8 ("0 ถึง 7")
เมื่อค่าเพิ่มจาก "0", "1", "2" ถึง "7" ค่าจะถูกย้ายไปทางซ้าย และจะกลายเป็น "10"
สัญลักษณ์ค่าฐานแปดนี้จะถูกใช้สำหรับหมายเลข I/O numbers ของ MELSEC iQ-F/F ซีรีส์

ค่าฐานสิบ	ค่าฐานสอง	ค่าฐานสิบหก	ค่าฐานแปด
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	8	10
9	1001	9	11
10	1010	A	12
11	1011	B	13
12	1100	C	14
16	1101	D	15
14	1110	E	16
15	1111	F	17
16	1000	10	20
17	10001	11	21
18	10010	12	22
.	.	.	.
.	.	.	.
.	.	.	.
26	011 010	1A	32

ภาพด้านล่างจะแสดงวิธีแสดงค่าฐานแปดเป็นค่าฐานสิบ

2	1	←จำนวนหลัก
3	2	←เลขค่าฐานแปด
8^1	8^0	←ยกกำลัง 8

$= 3 \times 8^1 + 2 \times 8^0$
$= 3 \times (8) + 2 \times (1)$
$= 26$

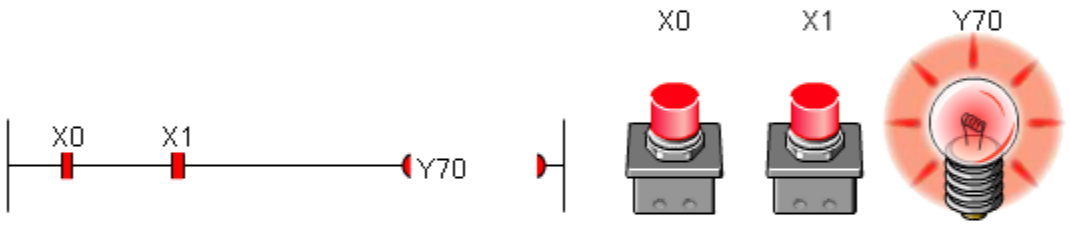
* ค่าฐานแปดหนึ่งหลักสามารถแสดงค่าฐานสองได้ 3 บิท

1.3 พื้นฐานการเขียนโปรแกรม PLC

ในการควบคุมตามลำดับ ชุดการทำงานจะถูกดำเนินการโดยอิงตามสัญญาณ ON/OFF ที่ได้รับจากอุปกรณ์ที่เชื่อมต่อกับโมดูลขาเข้า และผลการทำงานจะถูกเอาท์พุทไปยังอุปกรณ์ที่เชื่อมต่อกับโมดูลขาออก

ในการควบคุมดังกล่าว โปรแกรมควบคุมจะต้องมีเงื่อนไขอินพุทและเอาท์พุท ซึ่งจะถูกดำเนินการเมื่อเป็นไปตามเงื่อนไขอินพุท

โปรแกรมด้านล่างจะแสดงถึงการทำงานต่อไปนี้
- เมื่อสวิตช์ปุ่มกดเชื่อมต่อกับเทอร์มินอล X1 และ X2 อยู่ที่ ON ทั้งคู่ เปิด (ON) เทอร์มินอล Y70
- ผลการทำงานจะถูกเอาท์พุทไปยังเทอร์มินอล Y70 เพื่อเปิด (ON) หลอดไฟที่เชื่อมต่อ



การกดสวิตช์ X0 และ X1 ลงพร้อมกัน จะเป็นการเปิด (ON) หลอดไฟ Y70

1.4 หมายเลขและI/O ดีไวส์

โปรแกรมที่อธิบายไว้ในบทที่ 1.3 จะใช้สัญลักษณ์ตัวอักษรและตัวเลข เช่น X0, X1 และ Y70 ในการระบุI/O ดีไวส์ ตัวอักษรและตัวเลขเหล่านี้ จะถูกเรียกว่าหมายเลข I/O

บทนี้จะอธิบายเกี่ยวกับหมายเลขและI/O ดีไวส์ ซึ่งเป็น สิ่งจำเป็นสำหรับการสร้าง โปรแกรมควบคุม

MELSEC iQ-R/Q/L/iQ-F ซีรีส์ และ MELSEC-F ซีรีส์ ใช้รูปแบบที่แตกต่างกันในการแสดงหมายเลขดีไวส์ ตารางด้านล่างจะสรุปความแตกต่าง

MELSEC ซีรีส์	บิท			เวิร์ด
	X (หมายเลขอินพุท)	Y (หมายเลขเอาท์พุท)	M (รีเลย์ภายใน)	D (ดาต้ารีจิสเตอร์)
iQ-R/Q/L ซีรีส์	ค่าฐานสิบหก	ค่าฐานสิบหก	ค่าฐานสิบ	ค่าฐานสิบ
iQ-F/F ซีรีส์	ค่าฐานแปด	ค่าฐานแปด	ค่าฐานสิบ	ค่าฐานสิบ

1.4.1 หมายเลข I/O และสัญญาณ I/O (MELSEC iQ-R/Q/L ซีรีส์)

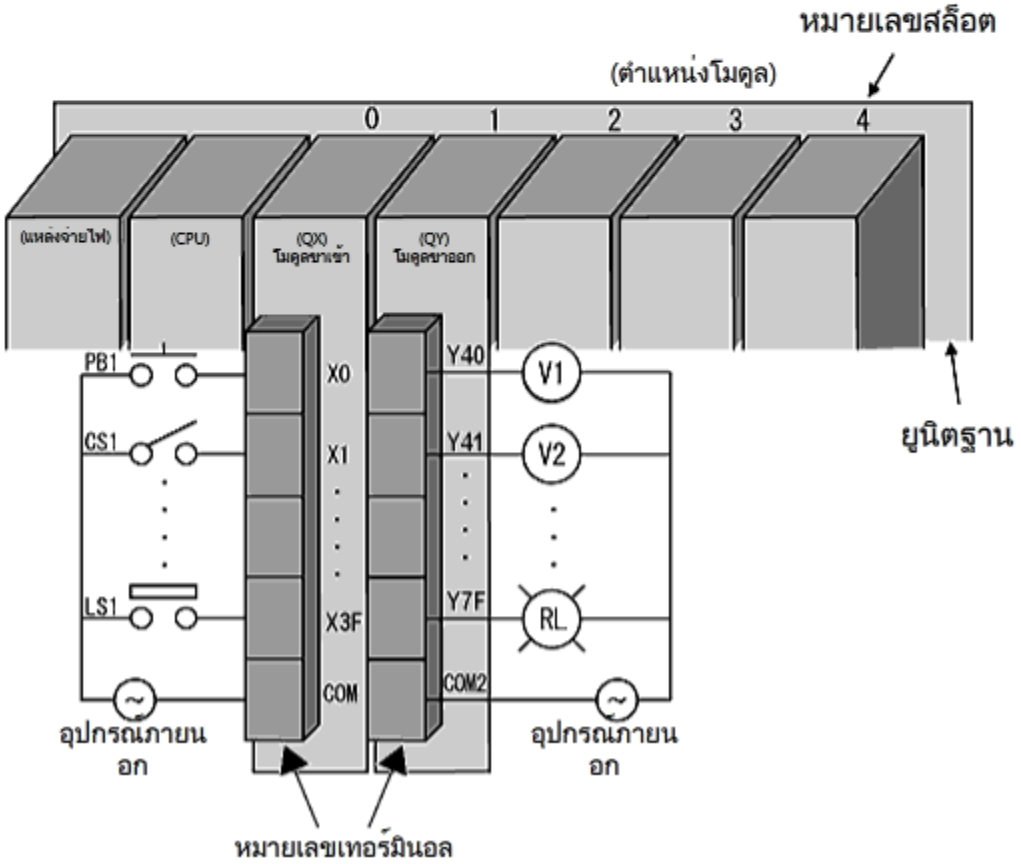
MELSEC iQ-R/Q/L ซีรีส์

หมายเลข I/O จะประกอบไปด้วยตัวอักษร ซึ่งจะระบุอินพุท (X) หรือเอาท์พุท (Y) และค่าฐานสิบหกที่แสดงถึงหมายเลขเทอร์มินอล

ในขั้นแรกหมายเลข I/O จะถูกกำหนดตามตำแหน่งการติดตั้งโมดูล

จากนั้นจะกำหนดช่วงของหมายเลข I/O ตามจำนวนจุด I/O ที่ใช้งานของโมดูล
(จำนวนจุด I/O ที่ใช้งาน จะเป็นสัดส่วนตามจำนวนของโมดูล I/O เทอร์มินอล)

ภาพด้านล่างจะแสดงวิธีการกำหนดหมายเลข I/O ให้กับโมดูลขาเข้า 64 จุด และโมดูลขาออก 64 จุด ซึ่งถูกติดตั้งลงบนสล็อตหมายเลข 0 และหมายเลข 1 ตามลำดับ



1.4.2 หมายเลข I/O และสัญญาณ I/O (MELSEC iQ-F/F ซีรีส์)

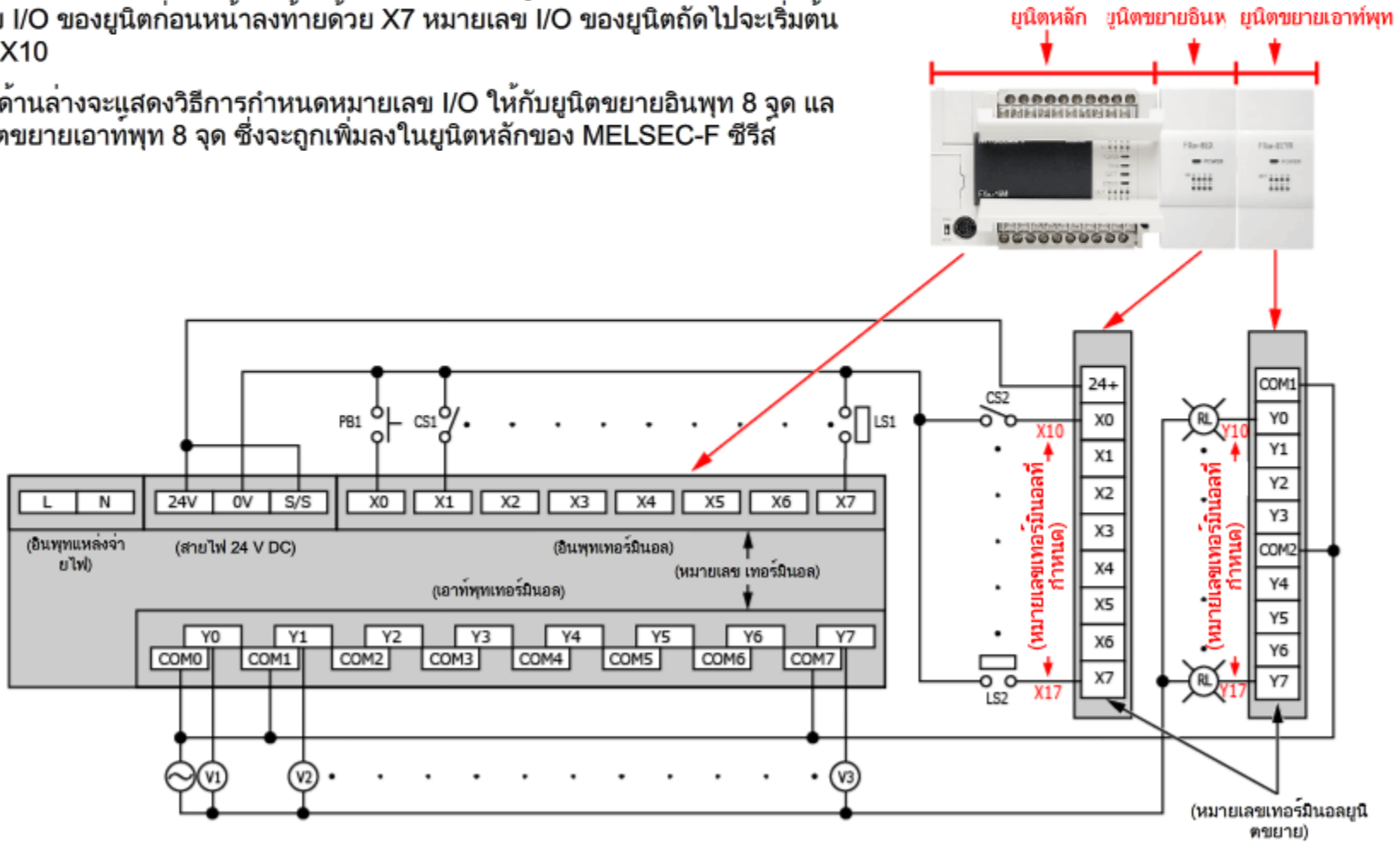
MELSEC iQ-F/F ซีรีส์

หมายเลข I/O จะประกอบไปด้วยตัวอักษร ซึ่งจะระบุอินพุท (X) หรือเอาท์พุท (Y) และค่าฐานแปดที่แสดงถึงหมายเลขเทอร์มินอล

ในขั้นแรก หมายเลข I/O จะถูกกำหนดตามหมายเลข I/O จบ (End) ของยูนิตหลัก หรือบล็อคขยาย I/O ที่อยู่ก่อนหน้า จากนั้นจะกำหนดช่วงของหมายเลข I/O ตามจำนวนจุด I/O ที่ใช้งานของยูนิต
(จำนวนของจุด I/O ที่ใช้งาน จะเป็นสัดส่วนตามจำนวนของจุด I/O ที่อยู่ในยูนิตขยาย I/O)

หลักแรกของหมายเลข I/O จะเริ่มต้นจาก "0" สำหรับยูนิตขยายใหม่ เช่น หากหมายเลข I/O ของยูนิตก่อนหน้าลงท้ายด้วย X7 หมายเลข I/O ของยูนิตถัดไปจะเริ่มต้นจาก X10

ภาพด้านล่างจะแสดงวิธีการกำหนดหมายเลข I/O ให้กับยูนิตขยายอินพุท 8 จุด และยูนิตขยายเอาท์พุท 8 จุด ซึ่งจะถูกเพิ่มลงในยูนิตหลักของ MELSEC-F ซีรีส์

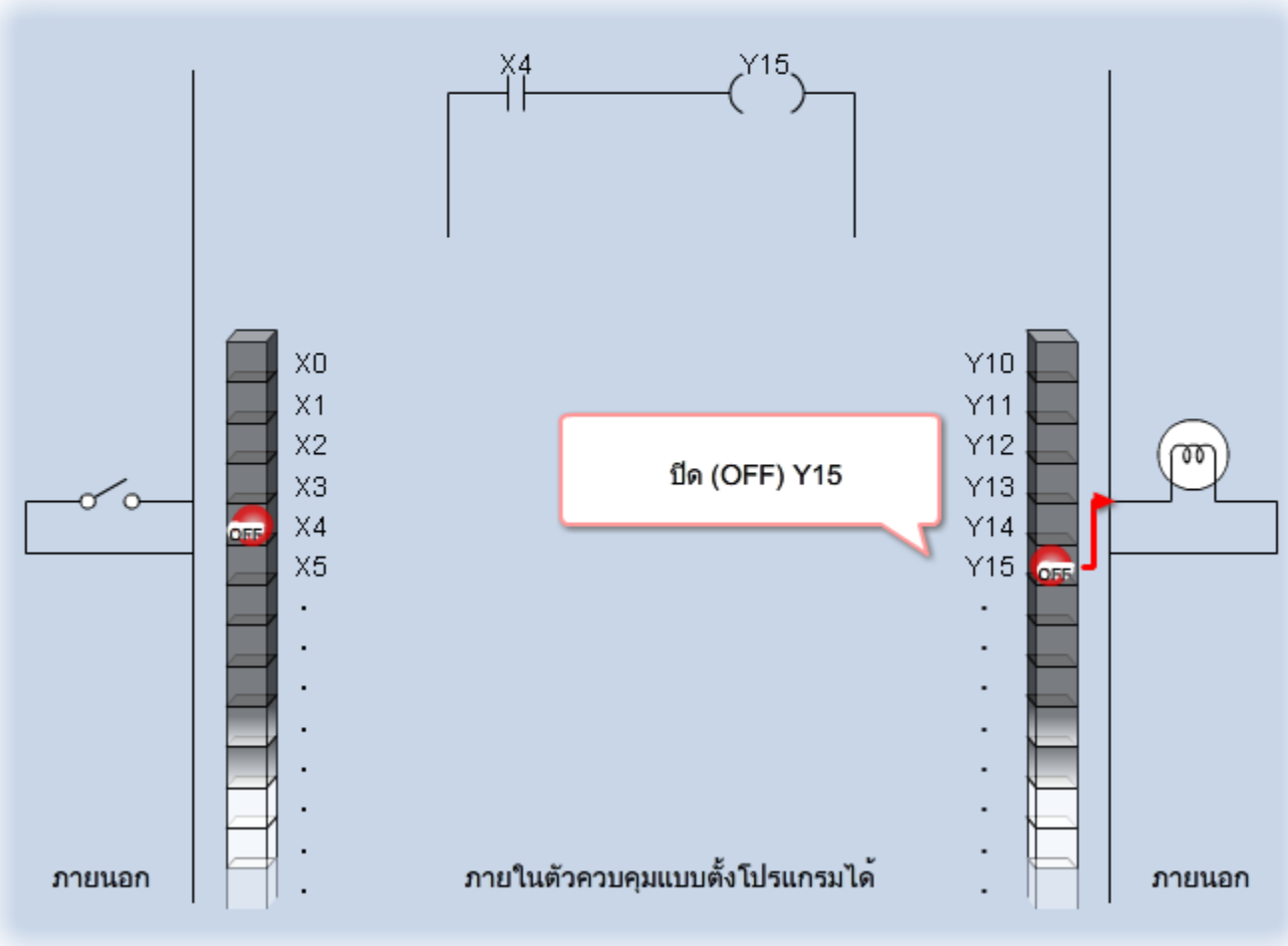


1.4.3 หมายเลขและI/O ดีไวส์

สถานะของอุปกรณ์ที่เชื่อมต่อกับยูนิต จะถูกบันทึกไว้ในพื้นที่หน่วยความจำ PLC ที่เรียกว่า "ดีไวส์"
เช่นเดียวกับเอาท์พุท เอาท์พุทดีไวส์อื่น ๆ จะทำงานตามสถานะของอุปกรณ์
ตามที่อธิบายไว้ด้านบน โปรแกรมควบคุมมักถูกดำเนินการตามสถานะของอุปกรณ์

ดีไวส์ที่จัดเก็บข้อมูลบิท (ON/OFF) เช่น อินพุท (X) และเอาท์พุท (Y) จะถูกเรียกว่า "บิทดีไวส์"

หมายเลขดีไวส์จะสอดคล้องกับหมายเลข I/O
ตัวอย่างเช่น สถานะของเทอร์มินอลที่กำหนดด้วยหมายเลข I/O X0 จะถูกบันทึกไว้ในดีไวส์ X0
เช่นเดียวกัน สถานะของดีไวส์ Y10 จะตรงกับเทอร์มินอลที่กำหนดด้วย I/O หมายเลข Y10



1.4.4 รีเลย์ภายใน

เราได้เรียนรู้ว่าบิทดีไวส์ เช่น X (อินพุท) และ Y (เอาท์พุท) จะตรงกับหมายเลขที่กำหนดให้กับโมดูล I/O เทอร์มินอลที่ติดตั้งไว้
จะมีบิทดีไวส์อีกหนึ่งกลุ่ม ซึ่งจะไม่มีความสัมพันธ์กับโมดูล I/O เทอร์มินอล และหนึ่งในนั้นจะมีชื่อว่า "รีเลย์ภายใน (M)"

รีเลย์ภายใน (M) จะถูกแสดงในรูปแบบค่าฐานสิบ แม้ว่าอินพุทดีไวส์ (X) และเอาท์พุท (Y) จะถูกแสดงเป็นค่าฐานสิบหกใน MELSEC iQ-R/Q/L ซีรีส์ และอยู่ในรูปแบบค่าฐานแปดใน MELSEC iQ-F/F ซีรีส์

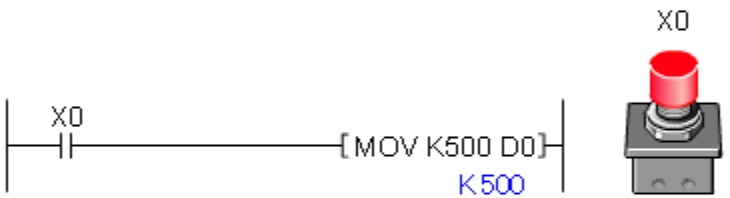
โดยส่วนใหญ่แล้ว รีเลย์ภายใน (M) จะถูกใช้ในการจัดเก็บข้อมูลบิทชั่วคราว
ตัวอย่างเช่น จะสามารถใช้รีเลย์ภายใน (M) เพื่อจัดเก็บผลลัพธ์การประมวลผลของการทำงาน เพื่อให้สามารถนำไปใช้ในอีกหนึ่งแลดเดอร์รังก์ได้

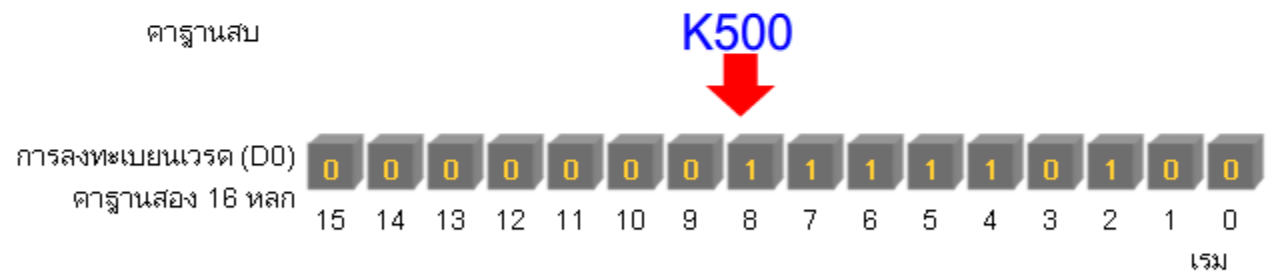
1.4.5 เวิร์ดดีไวส์

เราได้เรียนรู้ว่าดีไวส์ที่จัดเก็บข้อมูลบิท (ON/OFF) จะมีชื่อว่า "บิทดีไวส์" และดีไวส์ที่จัดเก็บเวิร์ดดีไวส์จะมีชื่อว่า "เวิร์ดดีไวส์"

"ดาต้ารีจิสเตอร์" (D) คือหนึ่งในเวิร์ดดีไวส์ที่มีการใช้งานโดยทั่วไป ดาต้ารีจิสเตอร์ (D) หนึ่งรายการ สามารถจัดเก็บข้อมูลได้ 1 เวิร์ด (16 บิท)

ภาพเคลื่อนไหวด้านล่างจะแสดงวิธีใช้ดาต้ารีจิสเตอร์ (D)
ในโปรแกรม การเปิด (ON) X0 จะเป็นการบันทึก "K500" (ค่าฐานสิบ) ลงใน D0
คำสั่ง MOV จะคัดลอกค่าลงในดีไวส์ที่กำหนด (รายละเอียดเพิ่มเติมจะอยู่ในบทที่ 3.1)
ในส่วนนี้ หมายเลขจะถูกบันทึกไว้ในดาต้ารีจิสเตอร์





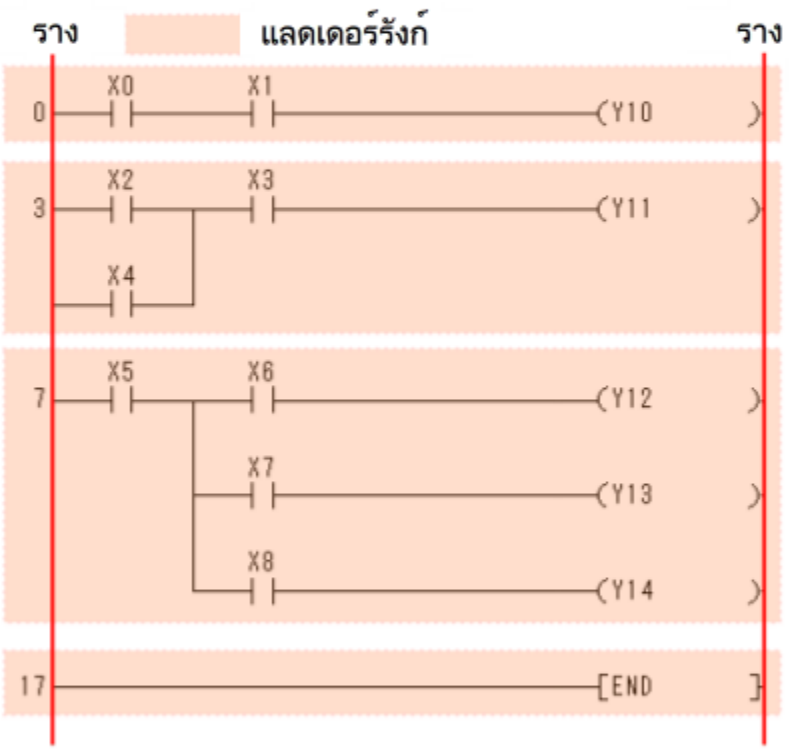
แม้ว่าจะปล่อยปุ่มและ X0 ปิด (OFF) แล้ว แต่ "K500" ที่บันทึกไว้ในการลงทะเบียนข้อมูล D0 จะยังคงอยู่

1.5 การสร้างโปรแกรมควบคุม

โปรแกรมควบคุมจะประกอบไปด้วยรางทางด้านซ้าย และขอบทางด้านขวา และสัญลักษณ์คำสั่งที่เชื่อมโยงกับเส้น

พื้นที่จากคำสั่ง -| |- ที่เชื่อมโยงรางทางด้านซ้ายเข้ากับคำสั่ง -( )- หรือ -[ ]- ซึ่งเชื่อมโยงกับรางทางด้านขวา จะถูกเรียกว่า แลดเดอร์รังก์

แลดเดอร์รังก์เหล่านี้หลาย ๆ รายการจะรวมกันเป็นโปรแกรมควบคุม ซึ่งจะจบด้วยคำสั่ง -[END]- or -[FEND]-



■ความแตกต่างระหว่างซอฟต์แวร์สำหรับแก้ไขโปรแกรม

คำสั่ง -( )- และ –[ ]- มีซอฟต์แวร์สำหรับแก้ไขโปรแกรมที่แตกต่างกัน คือ GX Works2 และ GX Works3 ในหลักสูตรนี้ คำอธิบายจะใช้คำสั่ง GX Works2

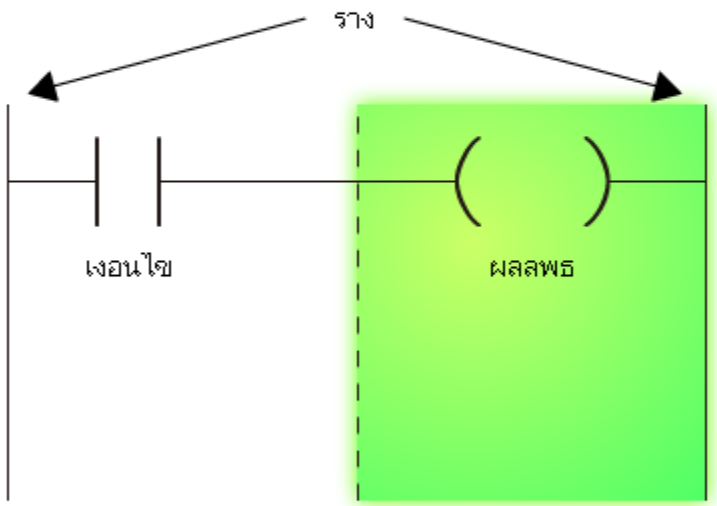
	GX Works2	GX Works3
-( )-	{Y10 }	Y10
-[ ]-	{MOV K500 D0 }	MOV K500 D0

1.5.1 สัญลักษณ์คำสั่ง

ตามที่อธิบายไว้ในบทที่ 1.3 จะต้องสั่งให้ PLCทำงานบางอย่าง เมื่อเป็นไปตามเงื่อนไขอินพุทที่กำหนดไว้ล่วงหน้า สำหรับคำสั่ง จะมีการใช้สัญลักษณ์คำสั่งเพื่อแสดงเงื่อนไขอินพุทและรายละเอียดเอาท์พุท

ในสัญลักษณ์คำสั่งมักมีหมายเลขดีไวส์
หมายเลขดีไวส์จะกำหนดพื้นที่ (ดีไวส์) ที่จัดเก็บสถานะ ซึ่งจะใช้สำหรับการพิจารณาเงื่อนไข หรือใช้เป็นผลลัพธ์เอาท์พุท

ในแลดเดอร์รังก์จะมีเงื่อนไขและผลลัพธ์เอาท์พุท เงื่อนไขจะอยู่ทางด้านซ้าย และผลลัพธ์เอาท์พุทจะอยู่ทางด้านขวา
ผลลัพธ์เอาท์พุทอาจเป็นสัญญาณ ON/OFF แบบง่าย ๆ หรือเป็นคำสั่งเฉพาะ เช่น การทำงานคำนวณหรือคัดลอก



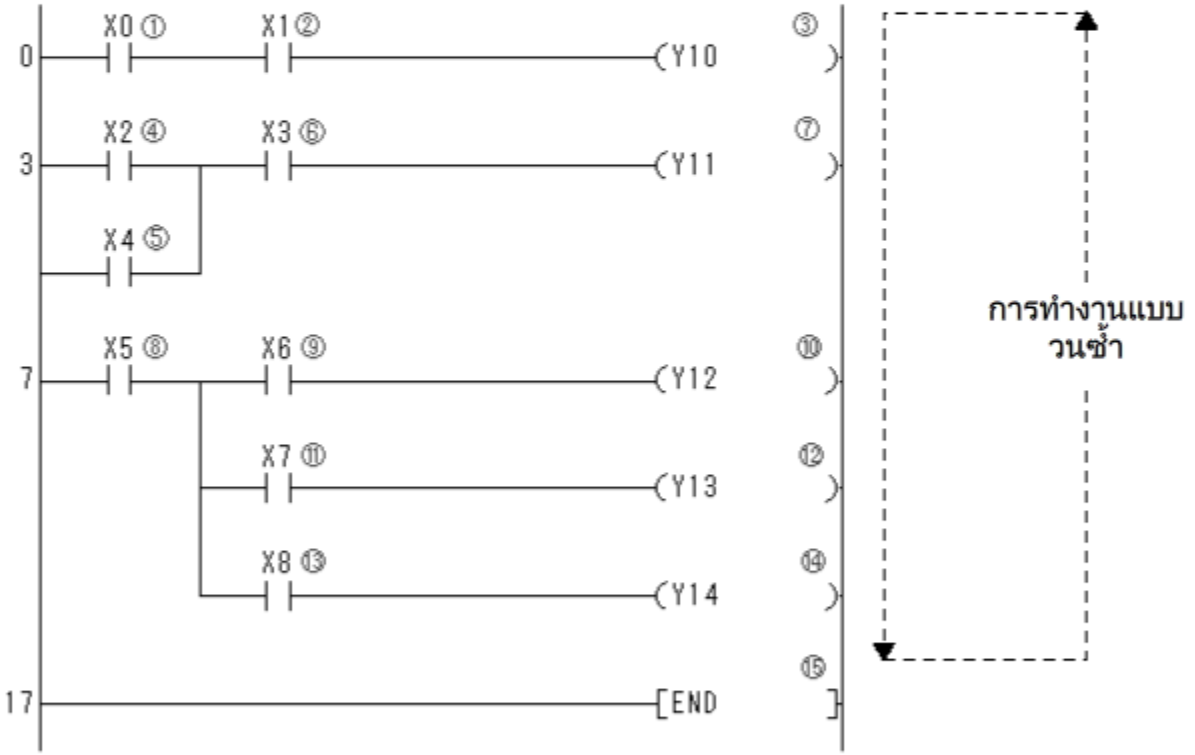
ในวงจรขั้นบันได จะมีรางขนานกันอยู่สองราง
ทางด้านซ้าย จะมีเงื่อนไขเขียนไว้
ทางด้านขวา จะมีผลลัพธ์เขียนไว้

1.6 กระบวนการทำงานของโปรแกรม

โปรแกรมจะเริ่มจากคำสั่งเริ่มต้นทางซ้ายและจบด้วยคำสั่ง -[END]-
หลังจากทำงานจนถึงคำสั่ง -[END]- การดำเนินการโปรแกรมจะเริ่มต้นอีกครั้งจากคำสั่งเริ่มต้น
การดำเนินการซ้ำนี้ มีชื่อว่า "การทำงานแบบวนซ้ำ"

การทำงานแบบวนซ้ำหนึ่งรอบ จะเรียกว่า "สแกน" และเวลาที่ใช้ในการประมวลผลการสแกนหนึ่งรอบ จะเรียกว่า "สแกนไทม์"

ภาพด้านล่างจะแสดงกระบวนการทำงานของคำสั่ง
คำสั่งจะถูกดำเนินการจากซ้ายไปขวา บนแต่ละแลดเดอร์รังก์ ตามด้วยจากบนลงล่างของแลดเดอร์รังก์ (หมายเลข 1, 2,...15 -> 1...)

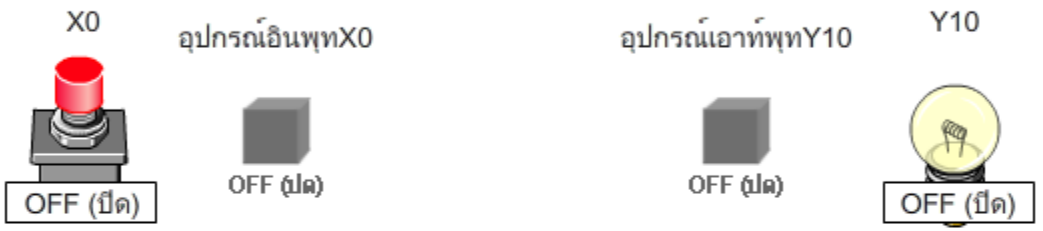


1.7 เวลารีเฟรช

ตามที่อธิบายไว้ก่อนหน้า สแกนไทม์ คือช่วงเวลาที่ต้องใช้ในการดำเนินการชุดโปรแกรมหนึ่งครั้ง สามารถแสดงสแกนไทม์ได้ในรูปแบบ:

สแกนไทม์ = เวลารีเฟรช + เวลาโปรแกรมทำงาน + เวลาประมวลผลการสิ้นสุด (END)

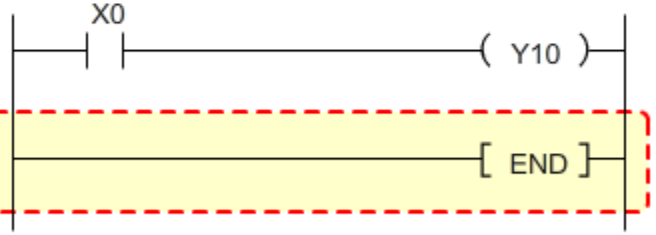
เวลารีเฟรชคือเวลาที่ต้องใช้ในการอ่านข้อมูลจากโมดูลอินพุทลงในอินพุทดีไวส์ (X) บวกเวลาที่ใช้ในการเขียนข้อมูลลงในโมดูลเอาท์พุทจากเอาท์พุทดีไวส์ (Y)



(1) รีเฟรช I/O

- ส่งสถานะ ON/OFF ของอุปกรณ์เอาท์พุทไปยังอุปกรณ์เอาท์พุทที่เชื่อมต่อ
- บันทึกสถานะ ON/OFF ที่ได้รับจากอุปกรณ์อินพุทที่เชื่อมต่อลงในอุปกรณ์อินพุท

(2) ดำเนินการโปรแกรม



(3) การประมวลผลจบ (END)

การประมวลผลจบ (END) ตามคำสั่ง END
(ในส่วนนี้จะละรายละเอียดไว้)

โปรดทราบว่าสถานะ ON/OFF ของสวิตช์จริงจะถูกอ่านหนึ่งครั้ง และบันทึกไว้ที่อินพุทดีไวส์ (X) และข้อมูลใหม่จะถูกเขียนทับค่าเดิมในระหว่างการรีเฟรชเช่นเดียวกัน ข้อมูลในเอาท์พุทดีไวส์ (Y) จะถูกเขียนลงในโมดูลขาออกทันทีที่มีการทำงานของคำสั่ง

ซึ่งหมายความว่าหากสัญญาณเปลี่ยนจาก OFF เป็น ON และกลับเป็น OFF อีกครั้ง ระบบจะไม่รับรู้ว่าสัญญาณเป็น ON อย่างไรก็ตาม สแกนไทม์จะสั้นมาก เมื่อเทียบกับความยาวของสัญญาณ มีโอกาสน้อยมากที่ PLCจะทำงานผิดพลาดในการเปลี่ยนแปลงสถานะสัญญาณ

บทที่ 2 คำสั่งบิทดีไวส์

บทนี้จะอธิบายเกี่ยวกับคำสั่งที่ใช้บิทดีไวส์ (ON/OFF)

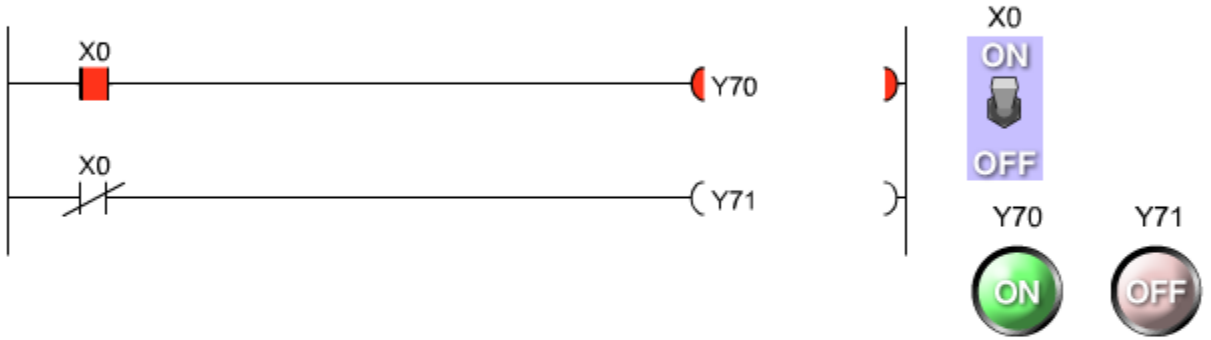
การทำงานที่ใช้บิทดีไวส์เป็นการทำงานพื้นฐานที่สุดในโปรแกรมควบคุม
อินพุทจากอินพุทดีไวส์จะถูกใช้เป็นเงื่อนไขในการควบคุมเอาท์พุทดีไวส์

2.1 เงื่อนไขอินพุทและเอาท์พุท

หน้าสัมผัสปกติเปิด (NO) และปกติปิด (NC) จะถูกใช้เป็นเงื่อนไขอินพุท
เมื่อเป็นไปตามเงื่อนไขอินพุท ระบบจะเอาท์พุทคำสั่งไปที่ขดลวด (คำสั่ง OUT)
เมื่อไม่เป็นไปตามเงื่อนไขอินพุท ระบบจะไม่เอาท์พุทคำสั่งไปที่ขดลวด
คำสั่งหน้าสัมผัส NO/NC และคำสั่ง OUT เป็นการรวมคำสั่งหลักที่ใช้ในโปรแกรมควบคุม

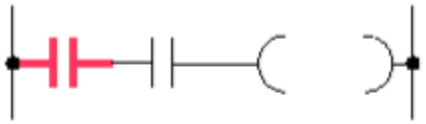
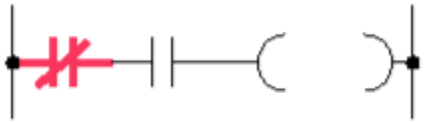
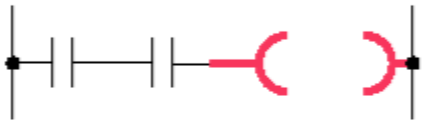
■ โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของ NO, NC และคำสั่ง OUT โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา

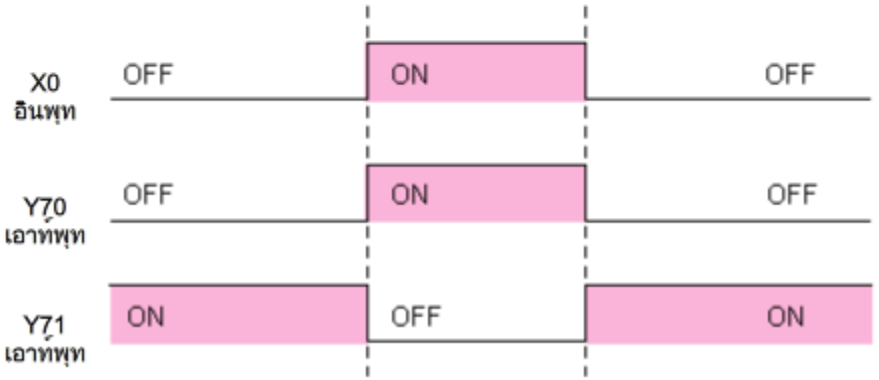


2.1 เงื่อนไขอินพุทและเอาท์พุท

■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	หน้าสัมผัส NO ดำเนินการเมื่อสถานะดีไวส์อยู่ที่ ON
	หน้าสัมผัส NC ดำเนินการเมื่อสถานะดีไวส์อยู่ที่ OFF (ตรงข้ามกับหน้าสัมผัส NO)
	เอาท์พุทขดลวด (OUT) เมื่อเป็นไปตามเงื่อนไขอินพุทที่อยู่ก่อนหน้า ข้อมูลในดีไวส์ตามค่าที่กำหนดไว้ล่วงหน้าจะถูกเอาท์พุท
	คำสั่งจบ (END) ระบุการสิ้นสุดของโปรแกรม โปรแกรมจะต้องมีคำสั่งจบ (END)

■ Time Chart

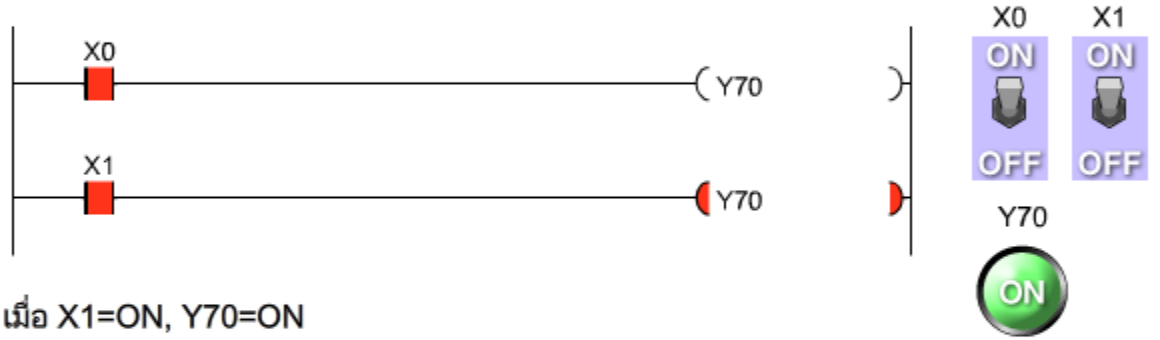


2.1.1 การใช้หมายเลขดีไวส์เดียวกันในคำสั่ง

บนแลดเดอร์รังก์จะสามารถใช้งานได้เพียงหนึ่งคำสั่ง Out ต่อหนึ่งหมายเลขดีไวส์ หากมีการใช้มากกว่าหนึ่งคำสั่ง OUT กับหมายเลขดีไวส์เดียวกัน เพียงค่าคำสั่ง OUT หลังจะมีผล ซึ่งจะทำให้คำสั่ง OUT แรกไม่มีผล

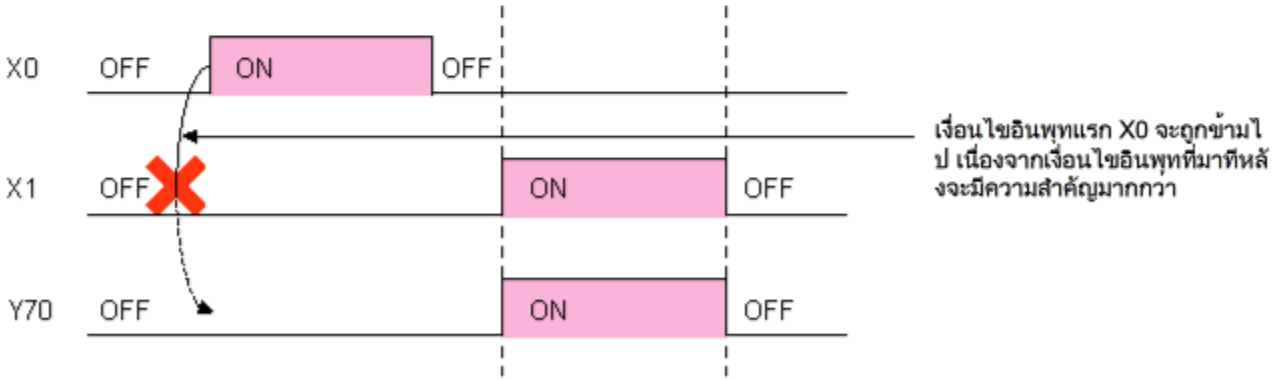
■ โปรแกรมแลดเดอร์

จำลองการทำงานของสองคำสั่งที่มีหมายเลขดีไวส์เหมือนกัน โดยการคลิกที่อินพุทสวิตช์ซึ่งปรากฏอยู่ทางด้านขวา การใช้งานชนิดนี้ (ใช้ OUT Y70 กับสองคำสั่ง) จะเรียกว่า "ขดลวดซ้ำ"



เมื่อ X1=ON, Y70=ON

■ Time Chart

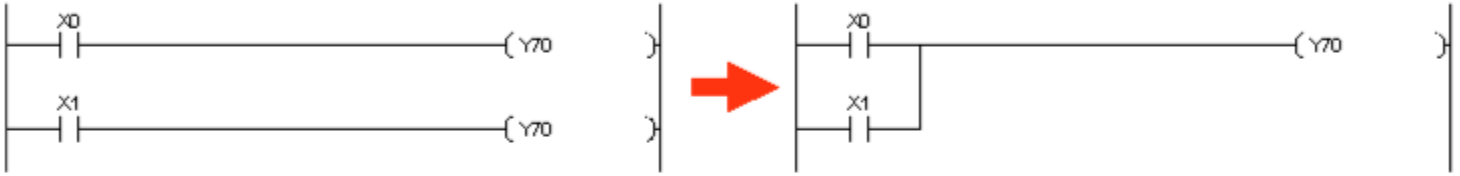


2.1.1 การใช้หมายเลขดีไวส์เดียวกันในคำสั่ง

■ ตัวอย่างการแก้ไข

ในตัวอย่างนี้ เงื่อนไขอินพุท "X1" จะมีความสำคัญมากกว่า และระบบจะข้าม "X0" ไป

เมื่อแก้ไขแลดเดอร์รังก์ให้อยู่ในรูปแบบตามในภาพ B ดีไวส์ Y70 จะถูกเปิด (ON) เมื่อเป็นไปตามหนึ่งในเงื่อนไขอินพุททั้งสอง โดยหลีกเลี่ยงความขัดแย้งระหว่างคำสั่ง OUT ทั้งสอง



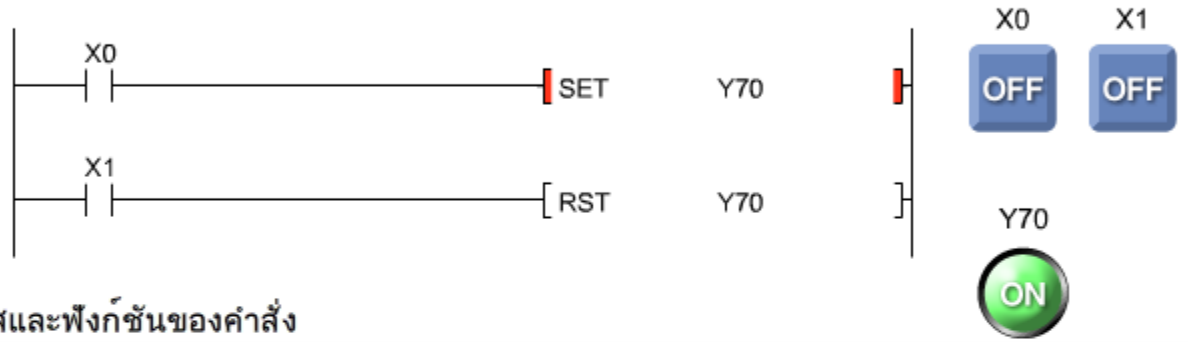
(ภาพ A) (ภาพ B)

2.2 การคงค่า/การยกเลิกเอาท์พุท

คำสั่งตั้งค่าการทำงาน (คำสั่ง SET) จะแตกต่างจากคำสั่ง OUT โดยจะเป็นการเก็บสถานะเอาท์พุทไว้ แม้ว่าจะไม่เป็นไปตามเงื่อนไขอินพุท ในการยกเลิกเอาท์พุท (OFF) สามารถดำเนินการคำสั่งรีเซ็ตค่าการทำงาน (คำสั่ง RST) ได้

■ โปรแกรมแลดเดอร์และการทำงาน

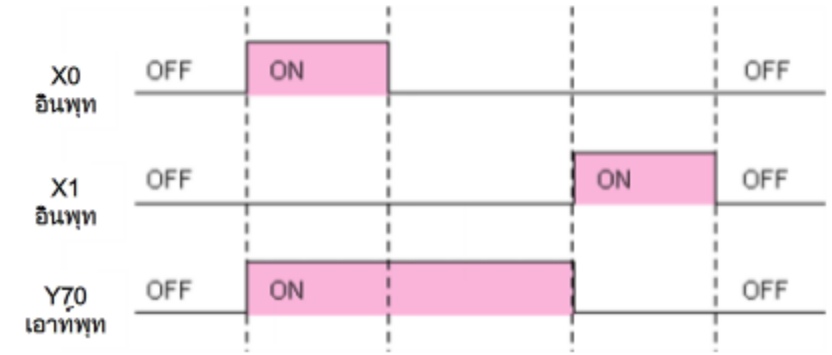
จำลองการทำงานของ SET, RST และคำสั่ง OUT โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา



■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	คำสั่งตั้งค่าการทำงาน (SET) จะเป็นการเปิด (ON) ดีไวส์ และคงค่าสถานะ ON (เอาท์พุท) ไว้ เอาท์พุทจะถูกตั้งค่าไว้แม้ว่าจะไม่เป็นไปตามเงื่อนไขอินพุท
	คำสั่งรีเซ็ตค่าการทำงาน (RSET) จะเป็นการยกเลิกสถานะ ON และยกเลิกการเอาท์พุทไปยังดีไวส์ที่กำหนด

■ Time Chart



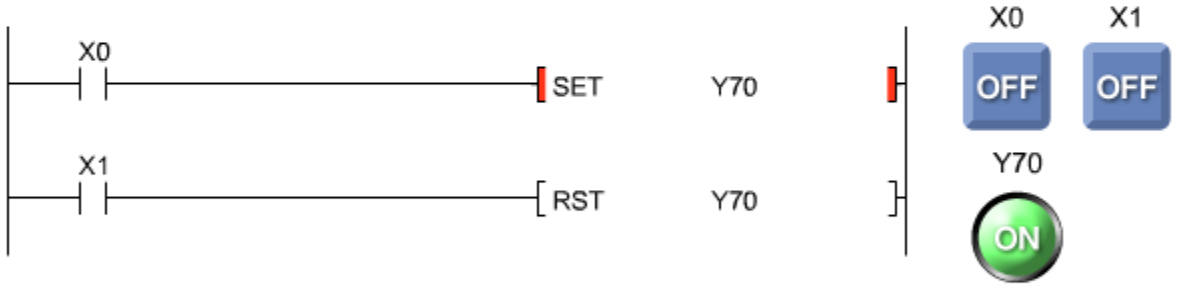
2.2.1 ความแตกต่างระหว่างคำสั่ง OUT และ SET

จำลองความแตกต่างในการทำงานระหว่างคำสั่ง OUT กับ SET โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา

■ คำสั่ง OUT

Y70 จะอยู่ที่ ON เมื่อเป็นไปตามเงื่อนไขอินพุท

■ คำสั่ง SET/RST

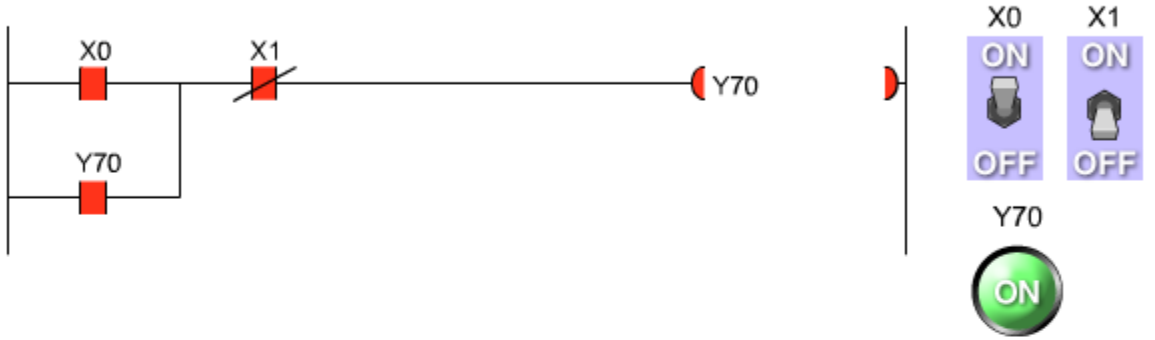


หลังจากที่เป็นไปตามเงื่อนไขอินพุท Y70 จะอยู่ที่ ON จนกว่าจะมีการทำงานของคำสั่ง RST

2.2.2 การแทนที่แลดเดอร์คงค่าด้วยคำสั่ง SET

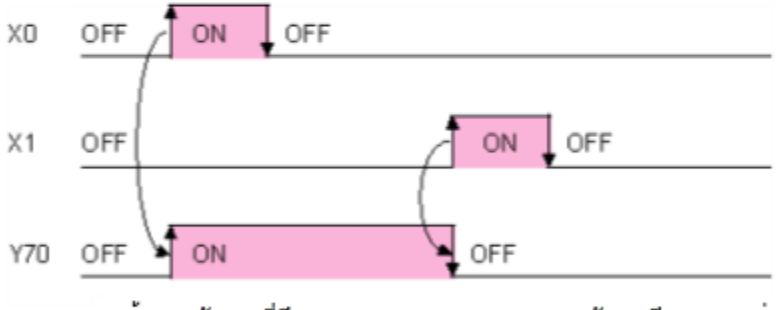
โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของแลดเดอร์คงค่า SET, RST และคำสั่ง OUT โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา



เมื่อ X0=ON และ X1=OFF, Y70=ON
Y70 = ON (คงค่า) จนกว่า X1=ON

■ Time Chart



แม้ภายหลังจากที่ปิด (OFF) X10, Y70 (ขดลวด) จะยังคงเปิด (ON) อยู่ (คงค่า)

■ การแทนที่ด้วยคำสั่ง SET

สามารถเขียนโปรแกรมแลดเดอร์ตั้งค่าใหม่เป็นโปรแกรมแลดเดอร์พร้อมคำสั่ง SET ได้
การใช้งานคำสั่ง SET จะช่วยลดความซับซ้อนให้คำสั่งโปรแกรมแลดเดอร์ได้

2.3 การเพิ่มเงื่อนไข (โลจิก AND)

ในการใช้โลจิก AND หน้าสัมผัส NO/NC จะถูกวางไว้ด้วยกันแบบอนุกรม
ภายในโลจิก AND จะถือว่าเป็นไปตามเงื่อนไข เมื่อหน้าสัมผัส NO/NC มากกว่าหนึ่งจุด ซึ่งเชื่อมต่อกันแบบอนุกรม อยู่ที่ ON

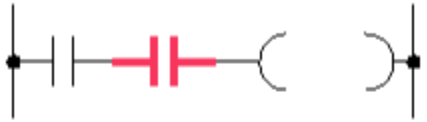
■ โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของโลจิก AND โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา

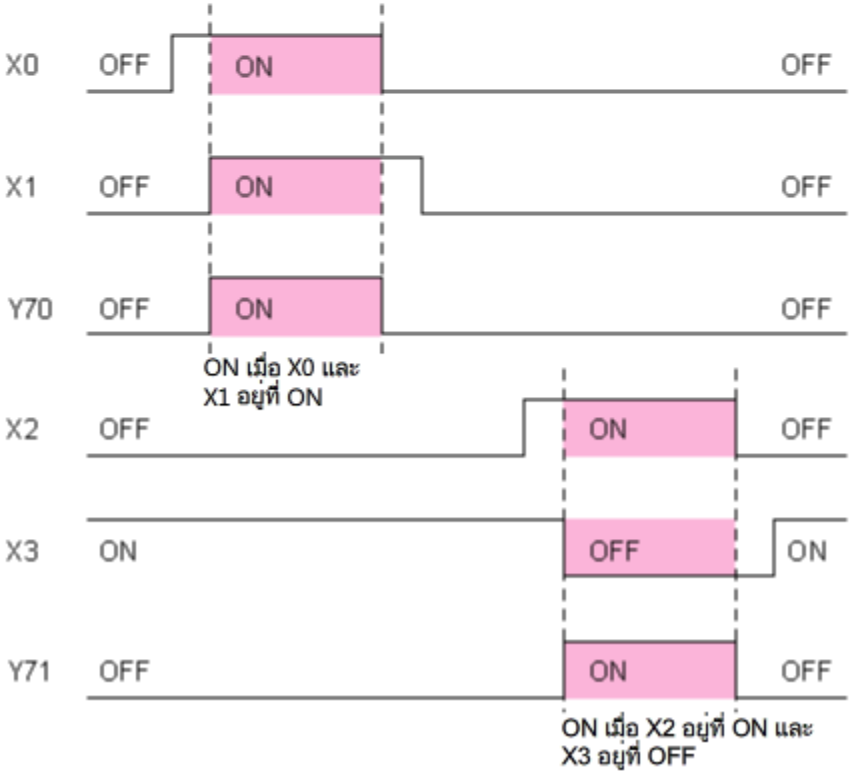
เมื่อ X0 และ X1 อยู่ที่ ON, Y70 จะอยู่ที่ ON
เมื่อ X2 อยู่ที่ ON และ X3 อยู่ที่ OFF, Y71 จะอยู่ที่ ON

2.3 การเพิ่มเงื่อนไข (โลจิก AND)

■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	การเชื่อมต่อแบบอนุกรมของหน้าสัมผัส NO หน้าสัมผัส NO จะเชื่อมต่อกันเป็นชุด (แนวนอน)
	การเชื่อมต่อแบบอนุกรมของหน้าสัมผัส NC หน้าสัมผัส NC จะเชื่อมต่อกันเป็นชุด (แนวนอน)

■ Time Chart



2.4 การเพิ่มเงื่อนไข (โลจิก OR)

ในการใช้โลจิก OR หน้าสัมผัส NO/NC จะถูกวางไว้ขนานกัน
ในโลจิก OR จะถือว่าเป็นไปตามเงื่อนไขเมื่อหนึ่งในหน้าสัมผัส NO/NC ซึ่งเชื่อมต่อขนานกัน อยู่ที่ ON

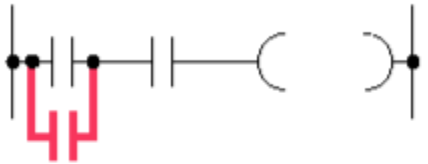
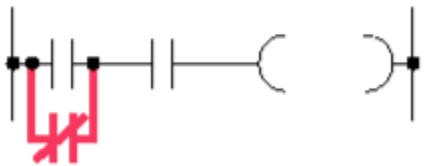
■ โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของโลจิก OR โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา

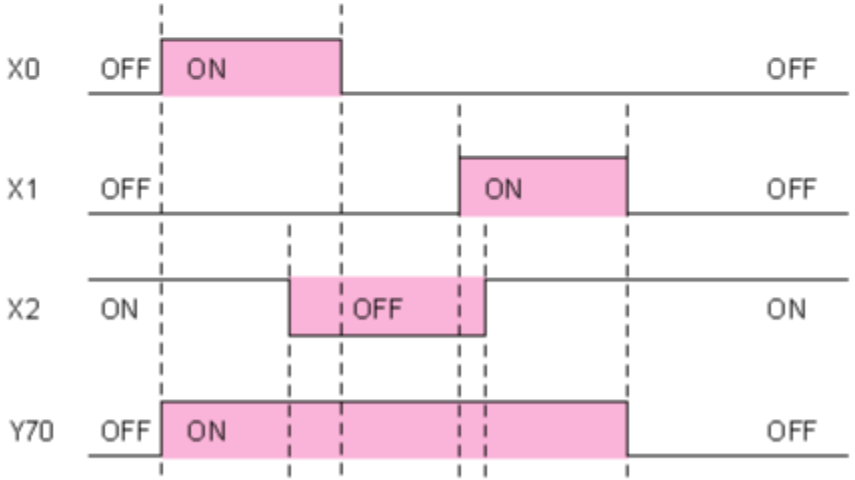
Y70 จะอยู่ที่ ON เมื่อเป็นไปตามเงื่อนไขใด ๆ ต่อไปนี้ X0 อยู่ที่ ON, X1 อยู่ที่ ON, หรือ X2 อยู่ที่ OFF

2.4 การเพิ่มเงื่อนไข (โลจิก OR)

■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	การเชื่อมต่อหน้าสัมผัส NO แบบขนาน หน้าสัมผัส NO จะถูกเชื่อมต่อกันแบบขนาน (แนวตั้ง)
	การเชื่อมต่อหน้าสัมผัส NC แบบขนาน หน้าสัมผัส NC จะถูกเชื่อมต่อกันแบบขนาน (แนวตั้ง)

■ Time Chart



2.5 เอาท์พุทในลักษณะพัลส์

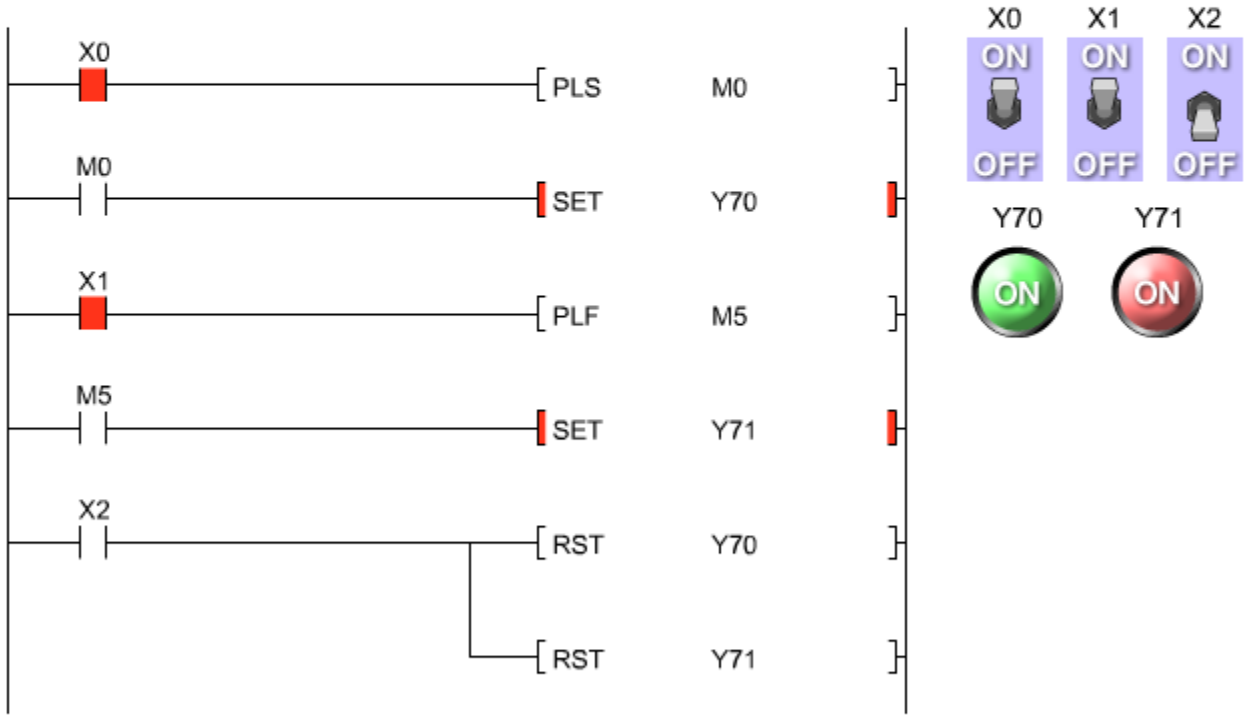
คำสั่งขอบขาขึ้น (คำสั่ง PLS) จะแตกต่างจากคำสั่ง OUT ซึ่งจะเป็นการเปิด (ON) ขดลวด ในหนึ่งรอบสแกน หลังจากที่เ
ป็นไปตามเงื่อนไขอินพุท

คำสั่งขอบขาลง (คำสั่ง PLF) จะตรงข้ามกับคำสั่ง PLS ซึ่งจะเป็นการเปิด (ON) ขดลวดในหนึ่งรอบสแกน หลังจากที่ไม่เป็
นไปตามเงื่อนไขอินพุท

ขดลวดที่ถูกเปิด (ON) ตามคำสั่ง PLS/PLF จะกลับสู่สถานะ OFF หลังจากการสแกนหนึ่งครั้ง

■ โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของคำสั่ง PLS และ PLF โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา



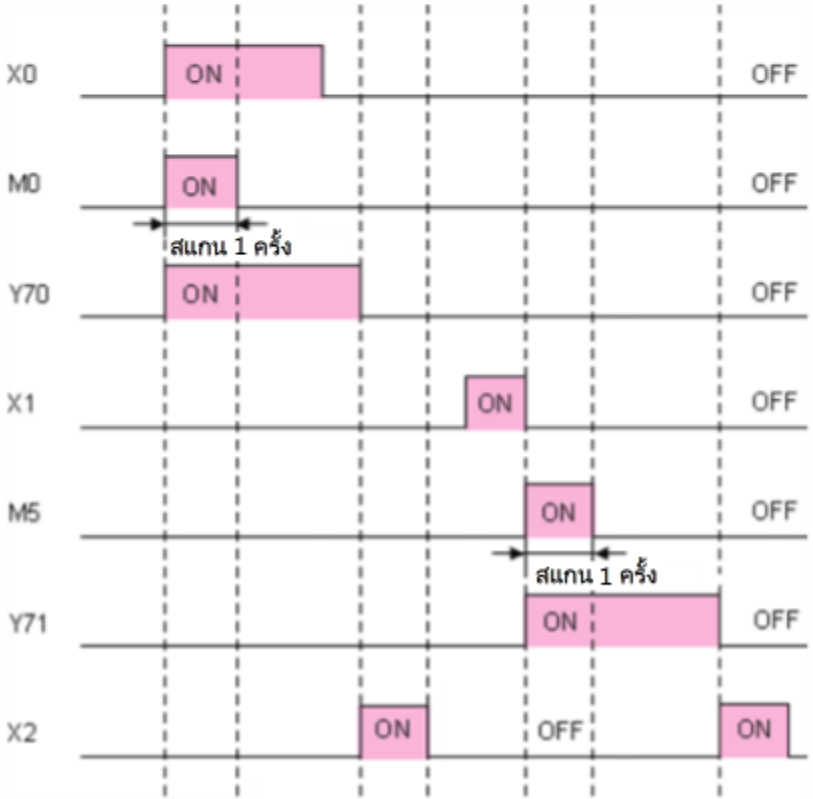
ที่ขอบขาขึ้น ของ X0 (OFF เป็น ON), M0 จะเปิด (ON) เพื่อการสแกน 1 ครั้ง
ที่ขอบขาลง ของ X1 (ON เป็น OFF), M5 จะเปิด (ON) เพื่อการสแกน 1 ครั้ง

2.5 เอาท์พุทในลักษณะพัลส์

■ รหัสและฟังก์ชันของคำสั่ง

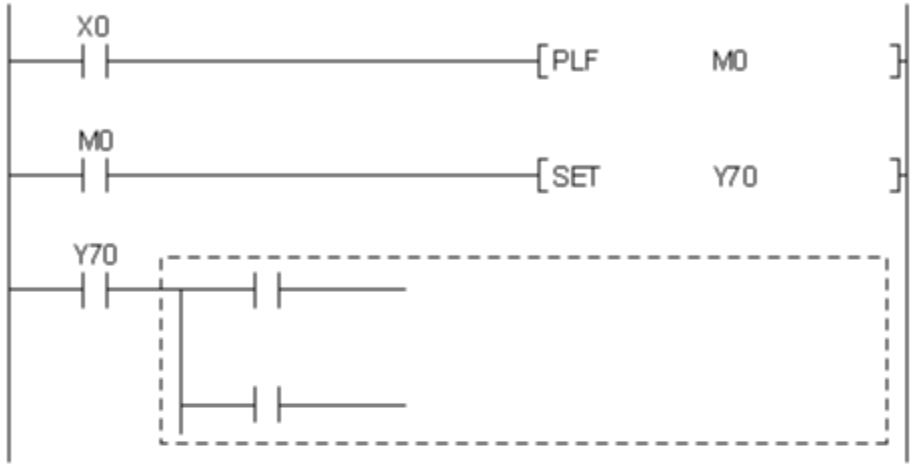
สัญลักษณ์	ฟังก์ชัน
	เอาท์พุทที่ขอบขาขึ้น (PLS) ข้อมูลจะถูกเอาท์พุทไปยังดีไวส์ที่กำหนดขณะทำการสแกนครั้งที่ 1 หลังจากที่เป็นไปตามเงื่อนไขอินพุท
	เอาท์พุทที่ขอบขาลง (PLF) ข้อมูลจะถูกเอาท์พุทไปยังดีไวส์ที่กำหนดขณะทำการสแกนครั้งที่ 1 หลังจากที่ไม่เป็นไปตามเงื่อนไขอินพุท

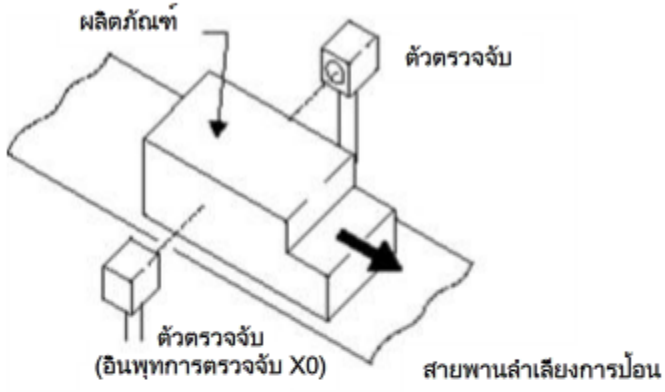
■ Time Chart



2.5.1 ตัวอย่างการใช้งานเอาท์พุทพัลส์

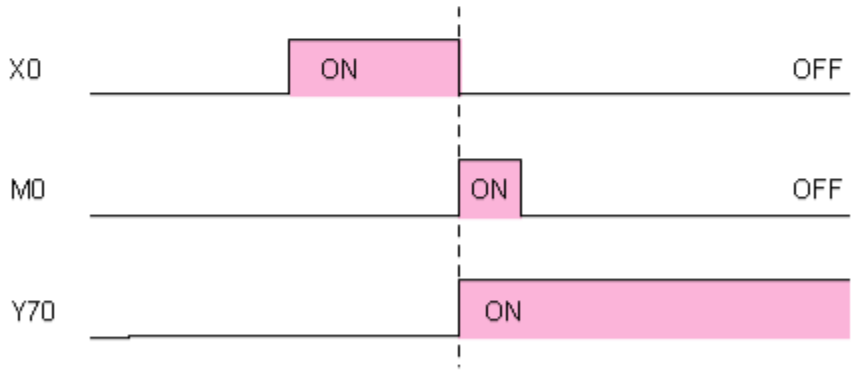
■ โปรแกรมแลดเดอร์





เอาท์พุทพัลส์จะถูกใช้ในการตรวจจับการเคลื่อนที่ผ่านของวัตถุ
เมื่อตรวจพบว่ามีผลิตภัณฑ์เคลื่อนที่ผ่าน กระบวนการถัดไปจะเริ่มต้นขึ้น

■ Time Chart



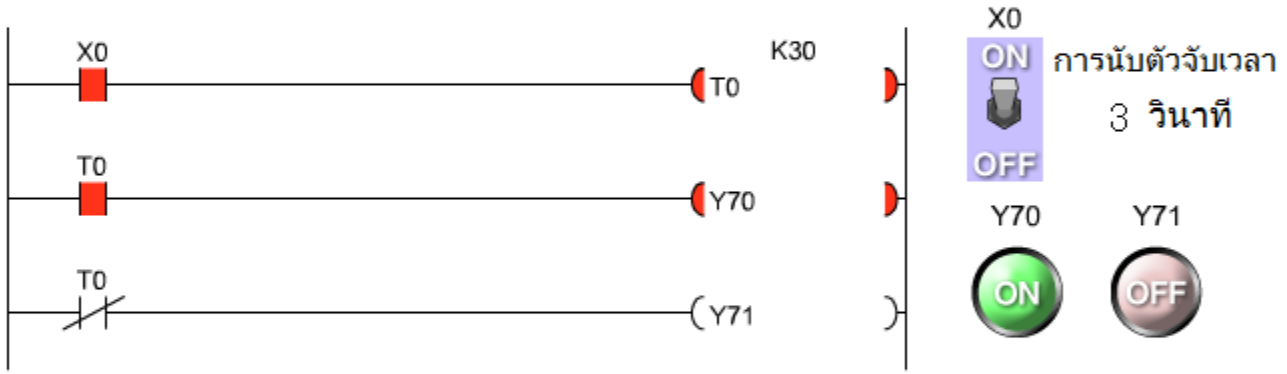
2.6 การจับเวลา

คำสั่ง OUT และดีไวส์ตัวจับเวลา (T) จะถูกใช้ในการจับเวลา
เมื่อเป็นไปตามเงื่อนไขอินพุท (ON) การจับเวลาจะเริ่มขึ้น เมื่อเวลาผ่านไปถึงค่าที่กำหนด ดีไวส์ตัวจับเวลา (T) จะถูกเปิด (ON)
หากไม่เป็นไปตามเงื่อนไขอินพุท (OFF) หรือดีไวส์ตัวจับเวลา (T) ถูกรีเซ็ตด้วยคำสั่ง RST เวลาที่ผ่านไปและเอาท์พุทจะถูกกลับไปค่าเริ่มต้น

สามารถนำสถานะของดีไวส์ตัวจับเวลา (T) ไปใช้เป็นเงื่อนไขอินพุทในส่วนอื่น ๆ ของโปรแกรมได้

■ โปรแกรมแลดเดอร์และการทำงาน

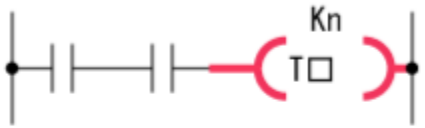
จำลองการทำงานตัวจับเวลา โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา



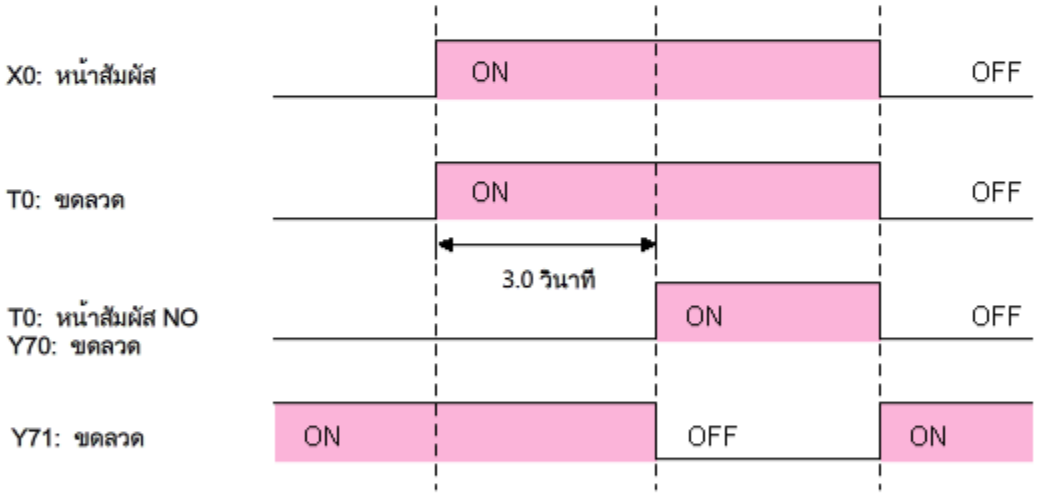
X0 จะเปิด (ON) และเมื่อผ่านไป 3 วินาที Y70 จะเปิด (ON) และ Y71 จะปิด (OFF)

2.6 การจับเวลา

■ รหัสและฟังก์ชันคำสั่ง

สัญลักษณ์	ฟังก์ชัน
 □ : หมายเลขดีไวส์	การทำงานของตัวจับเวลา ดีไวส์ตัวจับเวลา A (T) จะถูกนำไปใช้กับเอาท์พุทขดลวด (OUT) เพื่อจับเวลาการเป็นไปตามเงื่อนไข (คงอยู่ที่ ON) จะเกิดการหมดเวลาขึ้นหลังจากครบระยะเวลาที่กำหนด ควบคู่ไปกับการหมดเวลา ตัวจับเวลา (T0) จะเปิด (ON) ค่าที่ตั้งของตัวจับเวลา จะถูกระบุด้วย "Kn" (n: ค่าฐานสิบ) ตัวจับเวลามักถูกนำไปใช้เป็นตัวจับเวลาหน่วงการเปิด ซึ่งจะกำหนดช่วงเวลาหลังจากที่เป็นไปตามเงื่อนไขครบถ้วน

■ Time Chart



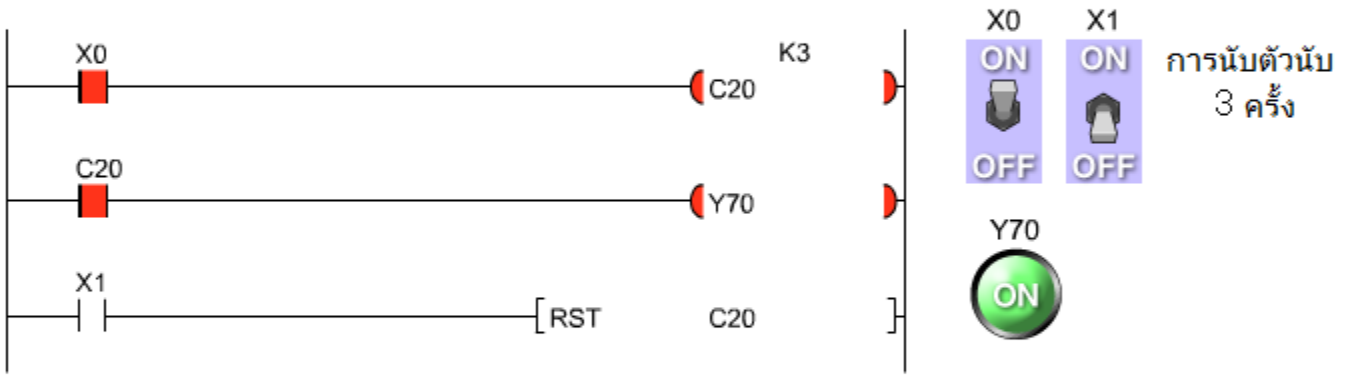
2.7 การนับ

คำสั่ง OUT และดีไวส์ตัวนับ (C) จะถูกใช้สำหรับการนับ
เมื่อเป็นไปตามเงื่อนไขอินพุท การนับจะเพิ่มขึ้น และเมื่อนับจนถึงค่าที่กำหนด ดีไวส์ตัวนับที่กำหนด (C) จะถูกเปิด (ON)
หากดีไวส์ตัวนับ (C) ถูกรีเซ็ตด้วยคำสั่ง RST การนับและสถานะดีไวส์จะถูกกลับไปค่าเริ่มต้น

สามารถนำสถานะของดีไวส์ตัวจับเวลา (C) ไปใช้เป็นเงื่อนไขอินพุทในส่วนอื่น ๆ ของโปรแกรมได้

■ โปรแกรมแลดเดอร์และการทำงาน

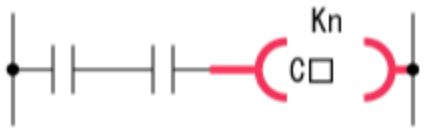
จำลองการทำงานของตัวนับ โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา



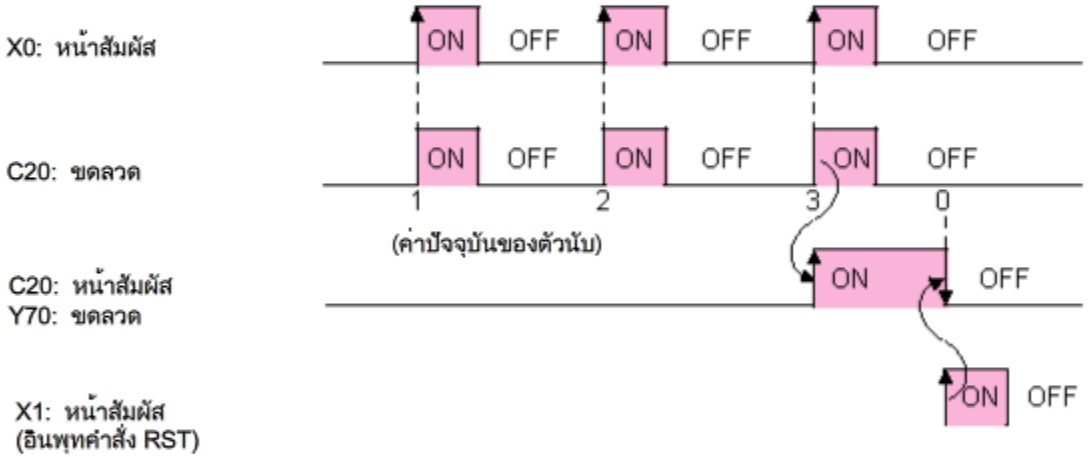
ค่าใน C20 จะเพิ่มขึ้นทุกครั้งที่ X0 เปิด (ON) เมื่อนับถึง 3 (นับออก) Y70 จะเปิด (ON)

2.7 การนับ

■ รหัสและฟังก์ชันคำสั่ง

สัญลักษณ์	ฟังก์ชัน
 □ : หมายเลขดีไวส์	ตัวนับ เมื่อใช้ร่วมกับเอาท์พุทขดลวด (OUT) ตัวนับจะทำการนับเพิ่ม (ทีละหนึ่ง) จำนวนครั้งที่เป็นไปตามเงื่อนไข การนับออกจะเกิดขึ้นเมื่อนับถึงจำนวนที่กำหนด และหน้าสัมผัสตัวนับจะเปิด (ON) ค่าที่ตั้งของตัวจับเวลา จะถูกระบุด้วย "Kn" (n: ค่าฐานสิบ)

■ Time Chart



บทที่ 3 คำสั่งเวิร์ดดีไวส์

บทนี้จะอธิบายเกี่ยวกับคำสั่งที่ใช้เวิร์ดดีไวส์

เวิร์ดดีไวส์จะมีประโยชน์ในการควบคุมเวลา การนับ และค่าที่อินพุทจากดีไวส์ภายนอก
เวิร์ดดีไวส์สามารถทำให้โปรแกรมควบคุมตอบสนองการทำงานจริงได้มากยิ่งขึ้น

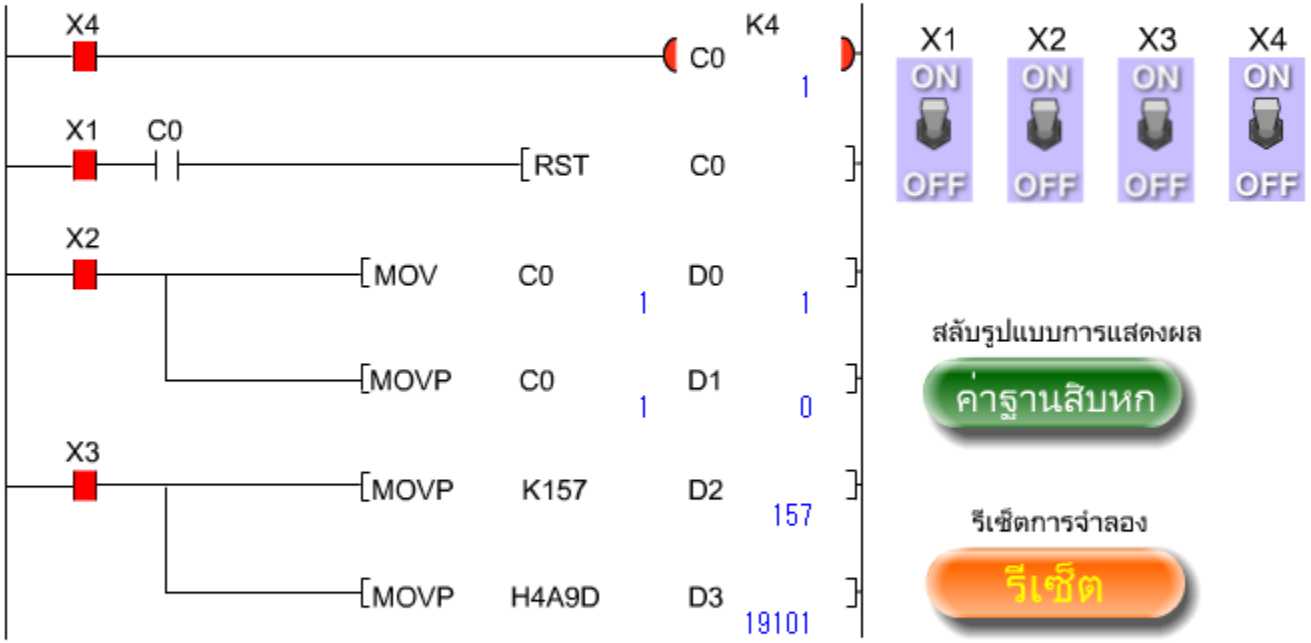
- จำลองการทำงานของโปรแกรมพื้นฐานเพื่อทำความเข้าใจการทำงานของคำสั่งหลัก
- จากการจำลอง ทำความเข้าใจบทบาทของคำสั่งและการประมวลผลที่ดำเนินการใน PLC

3.1 การย้ายข้อมูลไปยังเวิร์ดดีไวส์

คำสั่งการโอนถ่ายข้อมูล 16 บิท (MOV) จะเป็นการย้าย (คัดลอก) ข้อมูลยูนิต 1 เวิร์ด (16 บิท) ไปยังเวิร์ดดีไวส์ที่กำหนด ข้อมูลที่สามารถย้ายได้อาจเป็นค่าในดีไวส์หรือค่าที่กำหนดก็ได้ รูปแบบข้อมูลที่ย้ายได้สามารถอยู่ในค่าฐานสิบหรือค่าฐานสิบหกก็ได้

■ โปรแกรมแลดเดอร์และการทำงาน

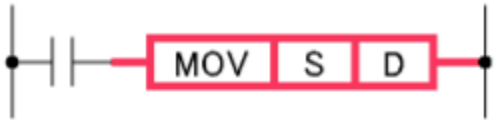
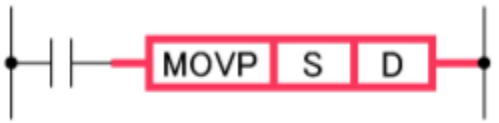
จำลองการทำงานของคำสั่งต่อไปนี้ โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา ตัวเลขสีน้ำเงินแต่ละตัว จะระบุค่า (ค่าปัจจุบัน) ที่จัดเก็บไว้ในดีไวส์



เมื่อคุณเปิด/ปิด (ON/OFF) X4 ซ้ำ ๆ ค่าปัจจุบันของ C0 จะเพิ่มขึ้น (0, 1...4->0...)

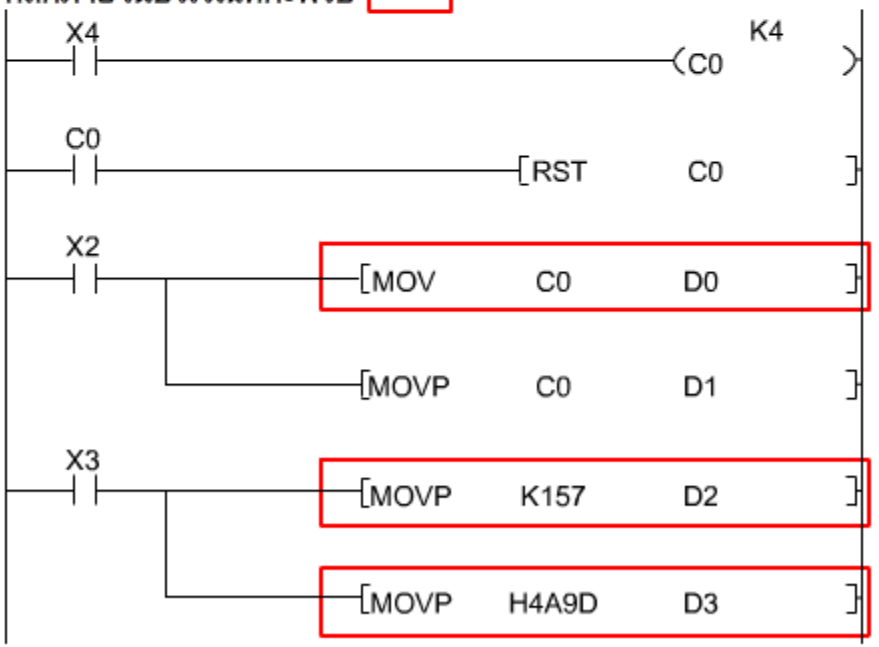
3.1 การย้ายข้อมูลไปยังเวิร์ดดีไวส์

■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	การการโอนถ่ายข้อมูล 16 บิท (MOV) เมื่อเป็นไปตามเงื่อนไขอินพุท ข้อมูลที่กำหนดไว้ในซอร์ส (S) จะถูกโอน (คัดลอก) ไปยังดีไวส์ที่กำหนดไว้ในปลายทาง (D)
	การการโอนถ่ายข้อมูล 16 บิท (พัลส์) (MOVP) ที่ขอบขาขึ้นของเงื่อนไข (OFF เป็น ON) ข้อมูลที่กำหนดไว้ในซอร์ส (S) จะถูกโอน (คัดลอก) ไปยังดีไวส์ที่กำหนดไว้ในปลายทาง (D)

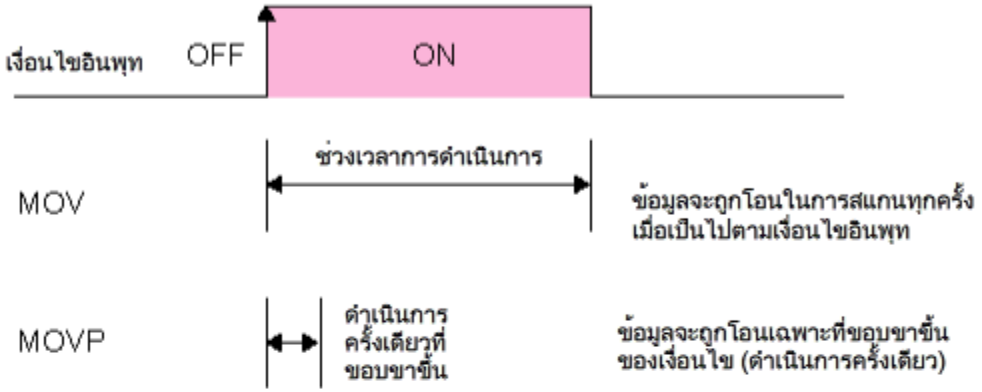
■ โปรแกรมแลดเดอร์

คลิกภายในบริเวณที่กะพริบ

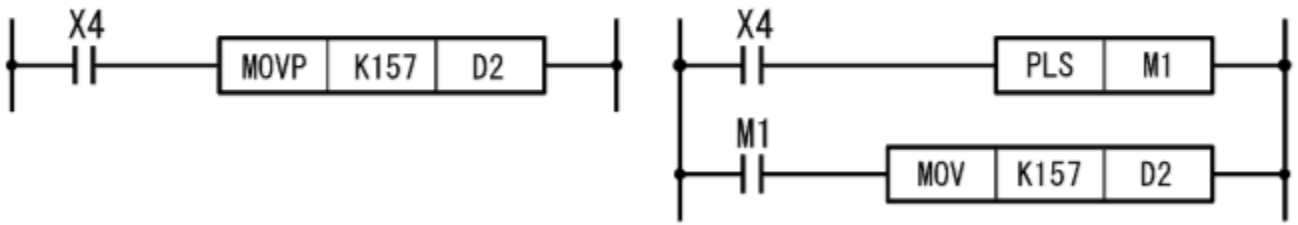


3.1.1 ความแตกต่างระหว่าง MOV และ MOVP

คำสั่ง MOV จะถูกใช้ในการอ่านข้อมูลที่มีการเปลี่ยนแปลงอย่างต่อเนื่อง ในขณะเดียวกัน คำสั่ง MOVP จะถูกใช้ในกา
รการโอนถ่ายข้อมูลหนึ่งครั้ง เช่น เพื่อตั้งค่าข้อมูล หรือเพื่ออ่านข้อมูล เมื่อเกิดข้อผิดพลาดขึ้น



ภาพด้านล่างจะแสดงสองโปรแกรมที่จะให้ผลลัพธ์ในการทำงานเดียวกัน ด้วยคำสั่ง MOV และ MOVP ในแลดเดอร์รังก์ทั้งคู่ การก
ารโอนถ่ายข้อมูลจะถูกดำเนินการเมื่อเปิด (ON) X4
เมื่อใช้คำสั่ง MOVP จะสามารถทำงานได้โดยไม่ต้องมีคำสั่ง PLS ซึ่งจะเป็นการกำหนดการดำเนินการการทำงานของขอบขาขึ้น

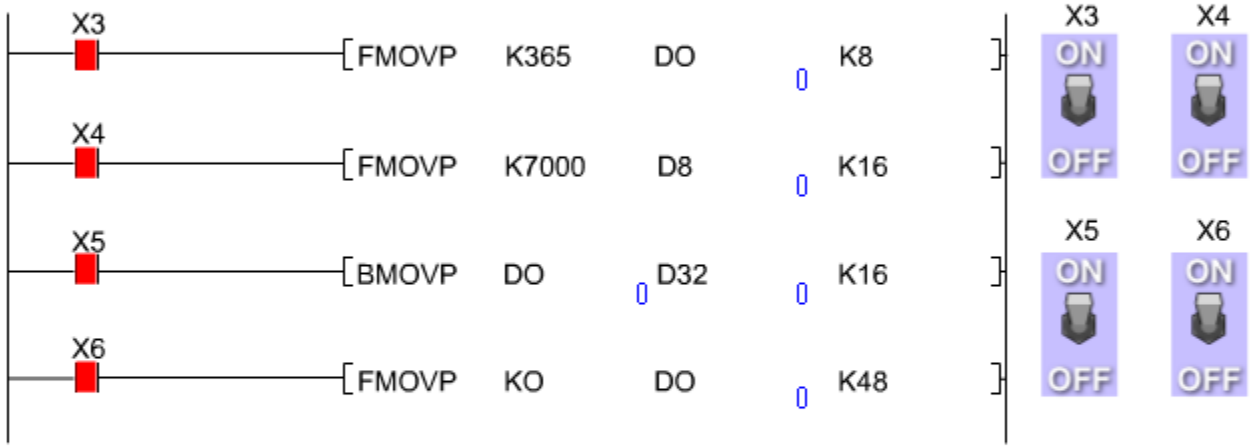


3.1.2 การย้ายข้อมูลไปยังหลาย ๆ เวิร์ดดีไวส์พร้อมกัน

คำสั่ง MOV/MOVP จะถูกนำไปใช้ในการการโอนถ่ายข้อมูลไปยังดีไวส์ ในการการโอนถ่ายข้อมูลไปยังหลาย ๆ ดีไวส์ ซึ่งมีหมายเลขต่อเนื่องกัน จะสามารถใช้ "คำสั่งโอนชุดข้อมูลที่เหมือนกัน" (FMOV) หรือ "คำสั่งโอนบล็อคข้อมูลที่เหมือนกัน" (BMOV) ได้

■ โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของคำสั่งต่อไปนี้ โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา ตัวเลขสีน้ำเงินแต่ละตัว จะระบุค่า (ค่าปัจจุบัน) ที่จัดเก็บไว้ในดีไวส์



การตรวจสอบอุปกรณ์

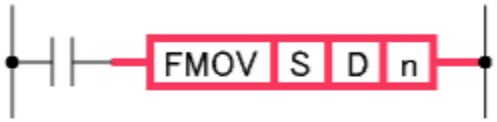
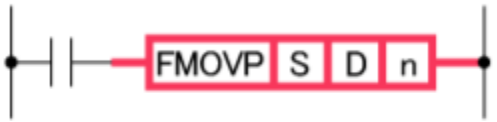
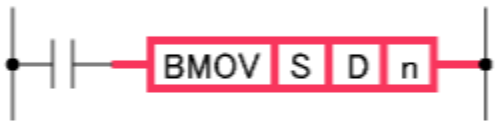
DO	0	D8	0	D32	0
D1	0	D9	0	D33	0
D7	0	D23	0	D47	0

เมื่อแต่ละสัญญาณอินพุทเปิด (ON) ข้อมูลที่กำหนดจะถูกโอนพร้อมกัน

หมายเหตุ: บน แลดเดอร์รังก์ ที่สาม ซึ่งเริ่มต้นด้วย X5 ข้อมูลจะถูกโอนตามคำสั่ง BMOV

3.1.2 การย้ายข้อมูลไปยังหลาย ๆ เวิร์ดดีไวส์พร้อมกัน

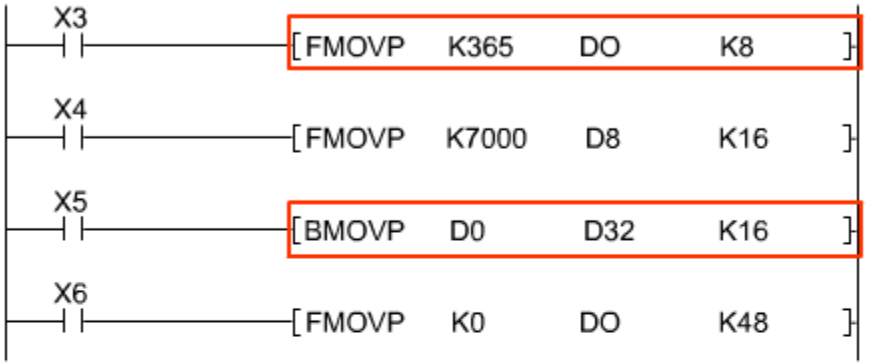
■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	การการโอนถ่ายข้อมูลที่เหมือนกันแบบชุด (FMOV) เมื่อเป็นไปตามเงื่อนไขอินพุท ข้อมูลที่กำหนดไว้ในซอร์ส (S) จะถูกโอน (คัดลอก) ไปยังดีไวส์ที่กำหนดไว้ด้วยปลายทาง (D) และจำนวน "n" ของดีไวส์ที่อยู่ถัดจาก D
	การการโอนถ่ายข้อมูลที่เหมือนกันแบบชุด (พัลส์) (FMOVP) เมื่อเป็นไปตามเงื่อนไขขอบขาขึ้น ข้อมูลที่กำหนดไว้ในซอร์ส (S) จะถูกโอน (คัดลอก) ไปยังดีไวส์ที่กำหนดไว้ด้วยปลายทาง (D) และจำนวน "n" ของดีไวส์ที่อยู่ถัดจาก D
	การโอนบล็อคข้อมูลแบบชุด (BMOV) เมื่อเป็นไปตามเงื่อนไขอินพุท ข้อมูลในดีไวส์ที่กำหนดไว้ในซอร์ส (S) และดีไวส์ที่อยู่ถัดไปในจำนวน "n" จะถูกโอนไปยังดีไวส์ที่กำหนดไว้ด้วยดีไวส์ (D) และไปยัง "n" จำนวนของดีไวส์ที่อยู่ถัดไป
	การโอนบล็อคข้อมูลแบบชุด (พัลส์) (BMOVP) เมื่อเป็นไปตามเงื่อนไขขอบขาขึ้น ข้อมูลในดีไวส์ที่กำหนดไว้ในซอร์ส (S) และดีไวส์ที่อยู่ถัดไปในจำนวน "n" จะถูกโอนไปยังดีไวส์ที่กำหนดไว้ด้วยดีไวส์ (D) และไปยัง "n" จำนวนของดีไวส์ที่อยู่ถัดไป

3.1.2 การย้ายข้อมูลไปยังหลาย ๆ เวิร์ดดีไวส์พร้อมกัน

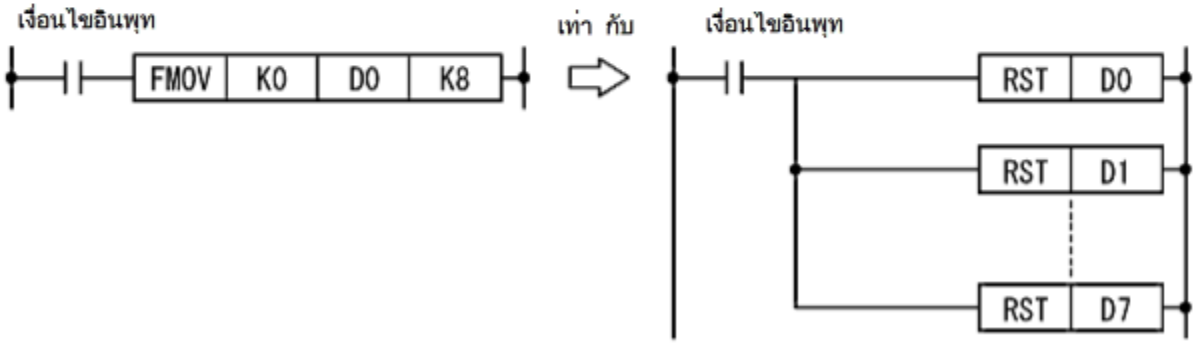
■ โปรแกรมแลดเดอร์และการทำงาน

คลิกภายในบริเวณที่กะพริบ



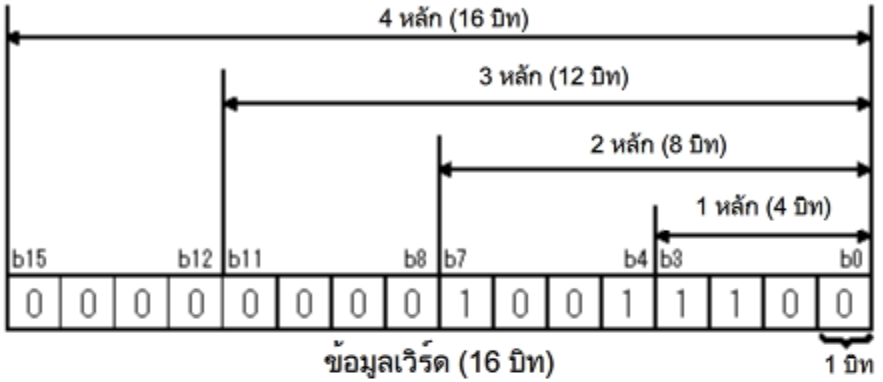
■ การใช้งานคำสั่ง FMOV และ BMOV

คำสั่ง FMOV เป็นคำสั่งที่สามารถใช้ในการลบข้อมูลจำนวนมากพร้อมกันในทีเดียวได้อย่างสะดวก



3.1.3 หลักบิทดีไวส์

สี่บิทดีไวส์จะถูกรวมเข้าเป็นหนึ่งหลักบิทดีไวส์ เพื่อควบคุมข้อมูลบิทในช่วงที่กำหนด (การการโอนถ่ายข้อมูล ฯลฯ)



■ วิธีกำหนดหลักบิทดีไวส์

หลักของบิทดีไวส์จะถูกแสดงเป็น "จำนวนหลัก" + "เลขดีไวส์เริ่มต้น" จำนวนหลักจะหาร 4 ลงตัว ตารางด้านล่างจะแสดงตัวอย่างบางส่วน
ตัวอย่างต่อไปนี้ จะเป็นตัวอย่างในขณะที่เลขดีไวส์เริ่มต้นคือ "M0"

ช่วงบิท	วิธีการกำหนด
ข้อมูล 16 บิท	K4M0 (16 บิท, M0 ถึง M15)
ข้อมูล 32 บิท	K8M0 (32 บิท, M0 ถึง M31)

หลักบิทดีไวส์ (จำนวนบิท) จะกำหนดช่วงของค่าตัวเลขที่ใช้ได้

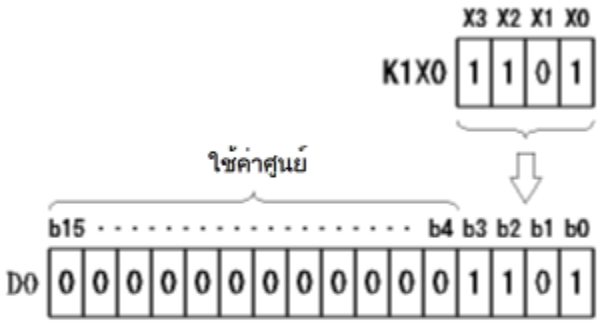
หลักบิทดีไวส์	ช่วงของค่าตัวเลขที่ใช้ได้
K1 (4 บิท)	0 ถึง 15
K2 (8 บิท)	0 ถึง 255
K3 (12 บิท)	0 ถึง 4095
K4 (16 บิท)	-32768 ถึง 32767 สามารถใช้บิทที่ 16 เป็นเครื่องหมายบวก/ลบ เพื่อแสดงค่าลบได้

3.1.3 ตัวอย่างการโอนหลักบิทดีไวส์

ใช้คำสั่งการโอนถ่ายข้อมูลในการโอนถ่าย (คัดลอก) ตัวเลขจากซอร์สดีไวส์ไปยังปลายทาง
ตัวอย่างต่อไปนี้ จะแสดงวิธีการการโอนถ่ายขอมูลที่กำหนด

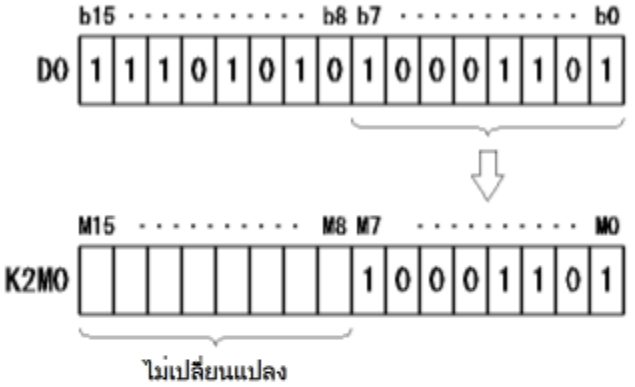
(ก) บิทดีไวส์แบบกำหนดหลัก → เวิร์ดดีไวส์

ตัวอย่าง) MOV K1X0 D0



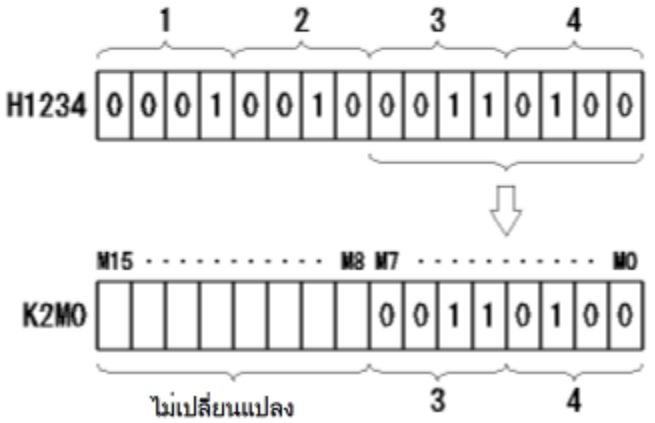
(ข) เวิร์ดดีไวส์ → บิทดีไวส์แบบกำหนดหลัก

ตัวอย่าง) MOV D0 K2M0



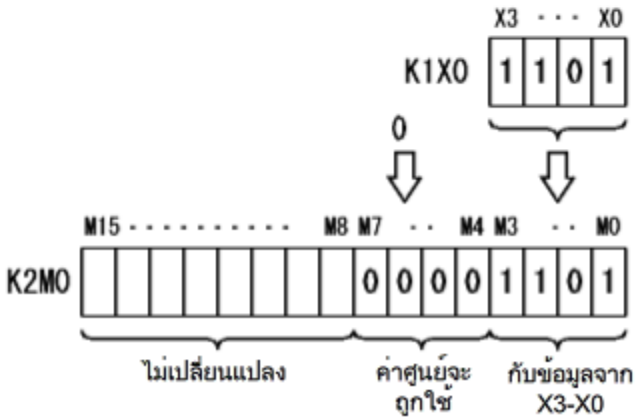
(ค) ค่าคงที่ (เลขที่กำหนดโดยตรง) → บิทดีไวส์แบบกำหนดหลัก

ตัวอย่าง) MOV H1234 K2M0



(ง) บิทดีไวส์แบบกำหนดหลัก → บิทดีไวส์แบบกำหนดหลัก

ตัวอย่าง) MOV K1X0 K2M0

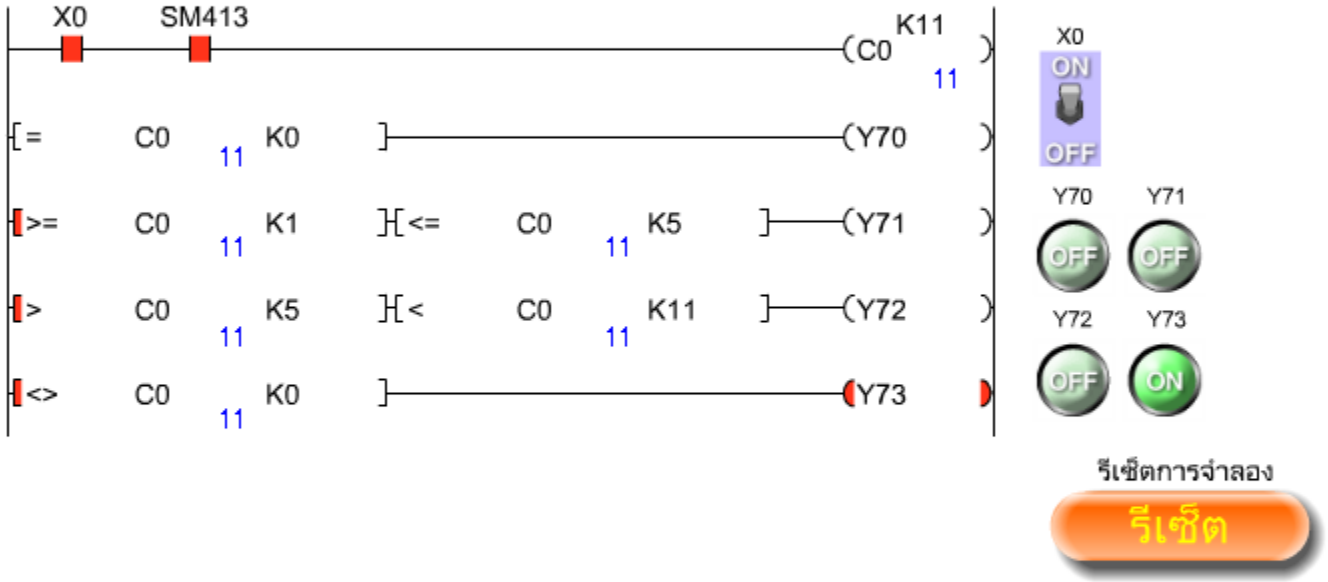


3.2 การเปรียบเทียบค่าตัวเลข

คำสั่งการทำงานเปรียบเทียบ จะใช้สำหรับการเปรียบเทียบข้อมูลเวิร์ด-ยูนิต และข้อมูลที่จัดเก็บไว้ในเวิร์ดดีไวส์
เมื่อเป็นไปตามเงื่อนไข (⊣⊢) คำสั่งถัดไปจะถูกดำเนินการ

■ โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของคำสั่งต่อไปนี้ โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา
ตัวเลขสีน้ำเงินแต่ละตัว จะระบุค่า (ค่าปัจจุบัน) ที่จัดเก็บไว้ในดีไวส์



Y70 ถึง Y73 จะเปิด/ปิด (ON/OFF) ตามค่าปัจจุบันของ C0

SM413 เป็นรีเลย์พิเศษที่จะเปิด (ON) หรือปิด (OFF) โดยเว้นช่วง 1 วินาที ตาม โมดูล CPU (จับเวลา 2 วินาที)
ขณะที่ X0 เปิด (ON), C0 จะนับเพิ่มทุก ๆ 2 วินาที

* SM413 เป็นรีเลย์พิเศษที่จะเปิด/ปิด (ON/OFF) โดยเว้นช่วง 1 วินาที (จับเวลา 2 วินาที) สามารถใช้ SM403 สำหรับ MELSEC iQ-R/Q/L/iQ-F ซีรีส์ MELSEC-F ซีรีส์ จะไม่มีรีเลย์จับเวลา 2 วินาที แต่จะมี M8011 (จับเวลา 0.01 วินาที), M8012 (จับเวลา 0.1 วินาที), M8013 (จับเวลา 1 วินาที) และ M8014 (จับเวลา 1 นาที)

3.2 การเปรียบเทียบค่าตัวเลข

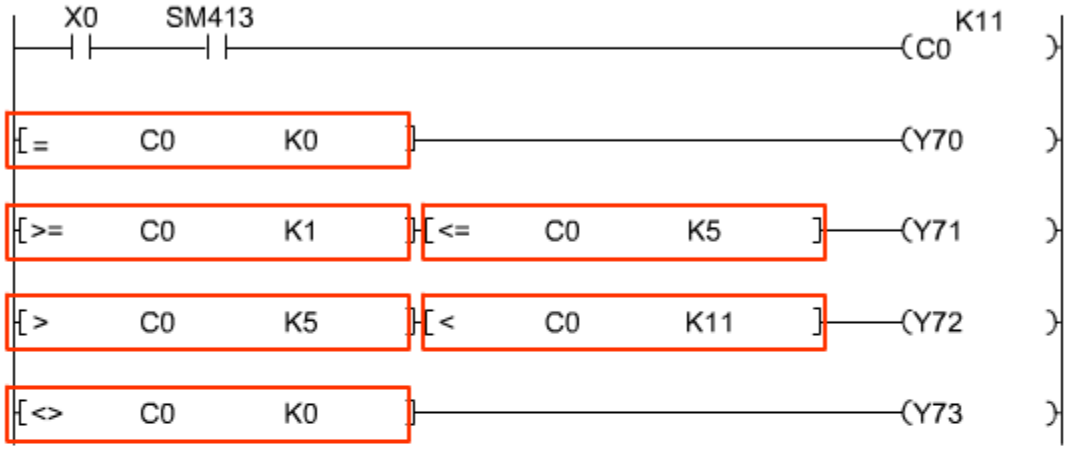
■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	เปรียบเทียบข้อมูลฐานสอง 16 บิท (=) เป็นไปตามเงื่อนไขเมื่อ SOURCE 1 เท่ากับ SOURCE2
	เปรียบเทียบข้อมูลฐานสอง 16 บิท (<) เป็นไปตามเงื่อนไขเมื่อ SOURCE 1 น้อยกว่า SOURCE2
	เปรียบเทียบข้อมูลฐานสอง 16 บิท (>) เป็นไปตามเงื่อนไขเมื่อ SOURCE 1 มากกว่า SOURCE2
	เปรียบเทียบข้อมูลฐานสอง 16 บิท (<=) เป็นไปตามเงื่อนไขเมื่อ SOURCE 1 เท่ากับหรือน้อยกว่า SOURCE2
	เปรียบเทียบข้อมูลฐานสอง 16 บิท (>=) เป็นไปตามเงื่อนไขเมื่อ SOURCE 1 เท่ากับหรือมากกว่า SOURCE2
	เปรียบเทียบข้อมูลฐานสอง 16 บิท (<>) เป็นไปตามเงื่อนไขเมื่อ SOURCE 1 ไม่เท่ากับ SOURCE2

3.2 การเปรียบเทียบค่าตัวเลข

■ โปรแกรมแลดเดอร์และการทำงาน

คลิกภายในบริเวณที่กะพริบ



SM413 เป็นรีเลย์พิเศษที่จะเปิด (ON) หรือปิด (OFF) โดยเว้นช่วง 1 วินาที ตาม โมดูล CPU (จับเวลา 2 วินาที) รีเลย์พิเศษ (SM) คือรีเลย์ดีไวส์ในโมดูล CPU รีเลย์พิเศษแต่ละตัวจะทำหน้าที่เฉพาะของตัวเอง

3.3 การดำเนินการคำนวณ

ส่วนนี้ จะอธิบายเกี่ยวกับการดำเนินการคำนวณของเวิร์ด (ตัวเลข) ดีไวส์

- ■ บวกและลบ
  การดำเนินการคำนวณด้วยเครื่องหมายบวก (+) และลบ (-)
- ■ คูณและหาร
  การดำเนินการคำนวณด้วยเครื่องหมายคูณ (*) และหาร (/)

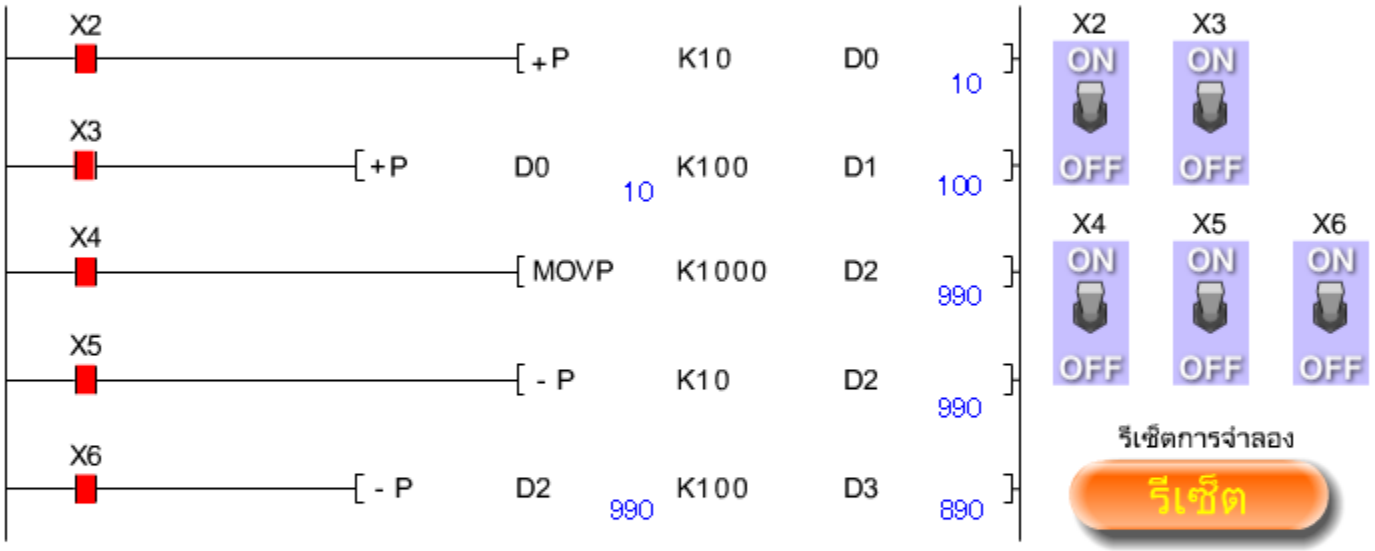
คำสั่งของ MELSEC iQ-R/Q/L/iQ-F ซีรีส์ และ MELSEC-F ซีรีส์จะแตกต่างกัน แต่แนวคิดพื้นฐานจะยังคงเหมือนกัน
ส่วนนี้ มีการอธิบายโดยอ้างอิงจากคำสั่งใน MELSEC iQ-R/Q/L /iQ-F ซีรีส์

3.3.1 บวกและลบ

แผนผังด้านล่างจะแสดงคำสั่งที่จะทำการบวกและลบ และบันทึกค่าที่ได้ลงในดีไวส์ที่กำหนด

■ โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของคำสั่งต่อไปนี้ โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา ตัวเลขสีน้ำเงินแต่ละตัว จะระบุค่า (ค่าปัจจุบัน) ที่จัดเก็บไว้ในดีไวส์

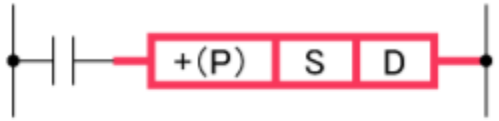
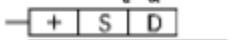
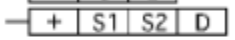
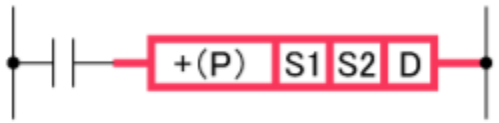
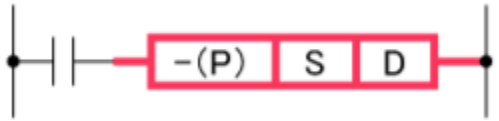
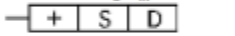
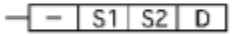
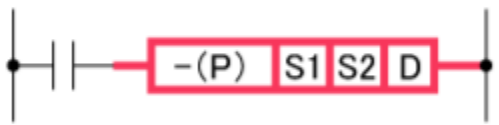


เมื่อสัญญาณอินพุทเปิด (ON) ระบบจะดำเนินการคำนวณ

* ตัวอย่างอ้างอิงจาก MELSEC iQ-R/Q/L/iQ-F ซีรีส์

3.3.1 บวกและลบ

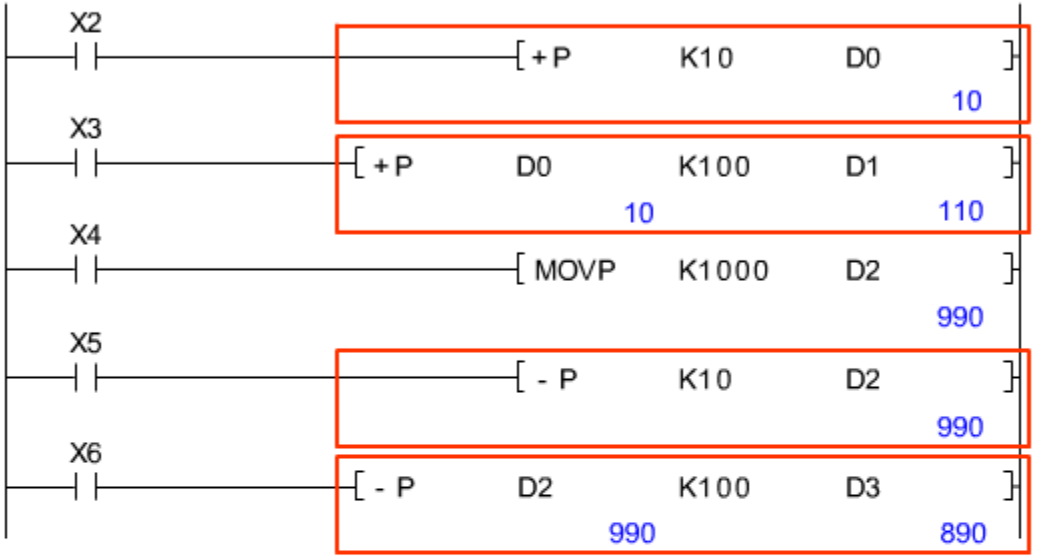
■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	การเพิ่มข้อมูลฐานสอง 16 บิท  : ดำเนินการ "D + S = D"  : ดำเนินการ "S1 + S2 = D"
	
	การลบข้อมูลฐานสอง 16 บิท  : ดำเนินการ "D - S = D"  : ดำเนินการ "S1 - S2 = D"
	

3.3.1 บวกและลบ

■ โปรแกรมแลดเดอร์และการทำงาน

คลิกภายในบริเวณที่กะพริบ

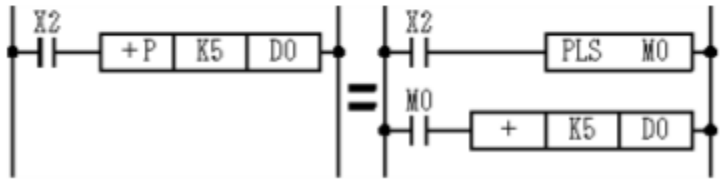


■ หมายเหตุเกี่ยวกับคำสั่งบวกและลบ

ในสถานการณ์ปกติ ให้ใช้คำสั่ง +P/−P ในการบวกและลบ

หากใช้คำสั่ง +/- ระบบจะทำการบวก/ลบซ้ำ ๆ ในขณะที่เป็นไปตามเงื่อนไขอินพุท

หากเกิดหนึ่งใน แลดเดอร์รังก์ ต่อไปนี้ ระบบจะดำเนินการบวกเพียงหนึ่งครั้งเมื่อ X2 เปิด (ON)



* ตัวอย่างอ้างอิงจาก MELSEC iQ-R/Q/L/iQ-F ซีรีส์

3.3.2 คูณและหาร

แผนผังด้านล่างจะแสดงคำสั่งที่จะทำการคูณและหาร และบันทึกค่าที่ได้ลงในดีไวส์ที่กำหนด

■ โปรแกรมแลดเดอร์และการทำงาน

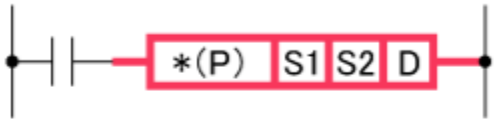
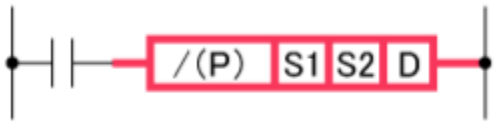
โปรดจำลองการทำงานของคำสั่งต่อไปนี้ โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา
ตัวเลขสีน้ำเงินแต่ละตัว จะระบุค่า (ค่าปัจจุบัน) ที่จัดเก็บไว้ในดีไวส์

เมื่อสัญญาณอินพุทเปิด (ON) ระบบจะดำเนินการคำนวณ

* ตัวอย่างอ้างอิงจาก MELSEC iQ-R/Q/L/iQ-F ซีรีส์

3.3.2 คูณและหาร

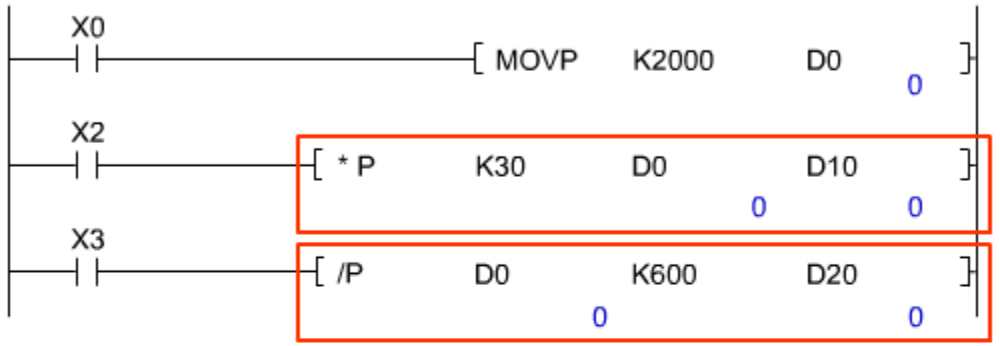
■ รหัสและฟังก์ชันของคำสั่ง

สัญลักษณ์	ฟังก์ชัน
	การคูณข้อมูลฐานสอง 16 บิท (*) ดำเนินการ "S1 x S2 = (D+1 D)" ("D+1" เป็นดีไวส์ที่อยู่ถัดจาก D หาก D เท่ากับ D100, "D+1" จะเท่ากับ D101) ผลการดำเนินการจะเป็นข้อมูล 32 บิท ซึ่งจะประกอบด้วย 2 เวิร์ดยูนิต ("D" และ "D+1")
	การหารข้อมูลฐานสอง 16 บิท ดำเนินการ "S1/S2 = (D [ผลหาร], D + 1 [เศษ])" ("D + 1" คือดีไวส์ที่อยู่ถัดจาก D หาก D เท่ากับ D100, "D + 1" จะเท่ากับ D101) ผลที่ได้จะเป็นจำนวนเต็ม

3.3.2 คูณและหาร

■ โปรแกรมแลดเดอร์และการทำงาน

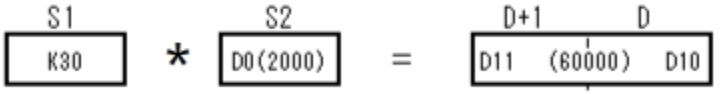
คลิกภายในบริเวณที่กะพริบ



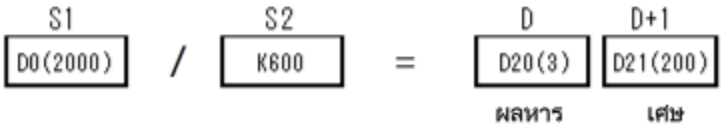
■ หมายเหตุเกี่ยวกับคำสั่งคูณและหาร

ในการทำงานของคำสั่งคูณหรือหาร จะต้องมีสองเวิร์ดดีไวส์เรียงต่อกัน (D, D+1) เป็นปลายทาง (D)

การคูณ



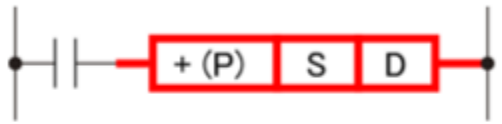
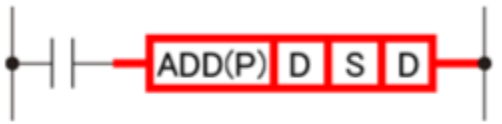
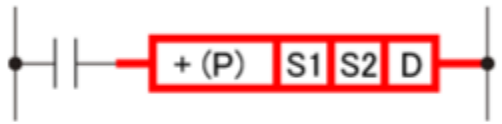
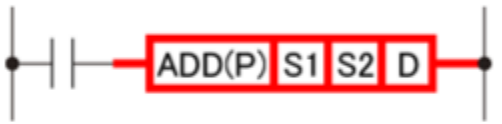
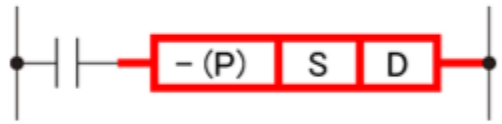
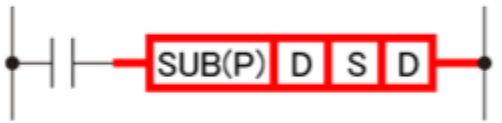
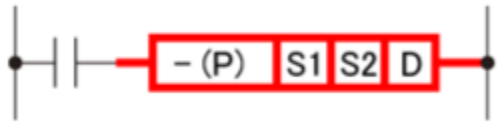
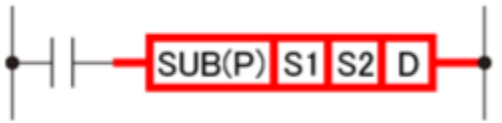
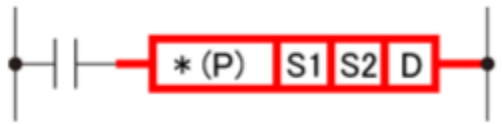
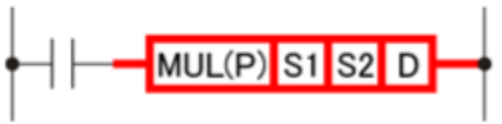
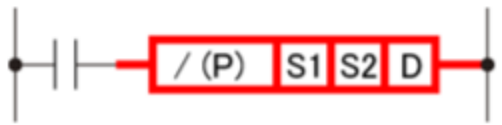
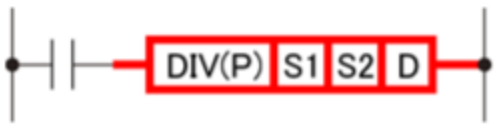
การหาร



* ตัวอย่างอ้างอิงจาก MELSEC iQ-R/Q/L/iQ-F ซีรีส์

3.3.3 ความแตกต่างระหว่าง MELSEC iQ-R/Q/L/iQ-F และ MELSEC-F

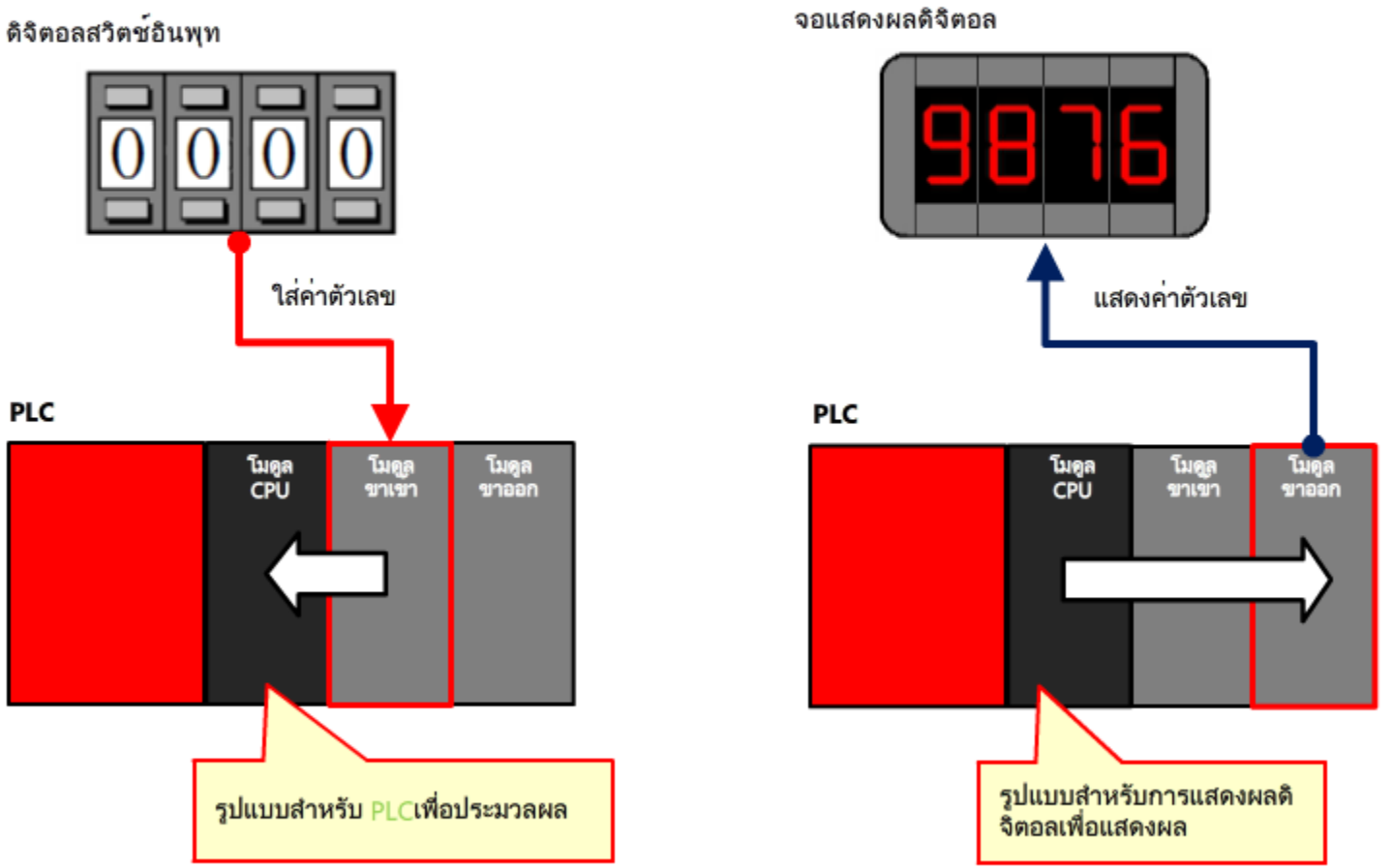
สัญลักษณ์ของ MELSEC iQ-R/Q/L/iQ-F ซีรีส์ และ MELSEC-F ซีรีส์ แตกต่างกัน
ตารางด้านล่างจะแสดงข้อแตกต่างหลัก

การดำเนินการคำนวณ	คำสั่งที่ใช้ใน MELSEC iQ-R/Q/L/iQ-F ซีรีส์	คำสั่งที่ใช้ใน MELSEC-F ซีรีส์	ความแตกต่าง
บวก (+)			MELSEC iQ-R/Q/L/iQ-F ซีรีส์: +(P) MELSEC-F ซีรีส์: ADD(P)
			
ลบ (-)			MELSEC iQ-R/Q/L/iQ-F ซีรีส์: -(P) MELSEC-F ซีรีส์: SUB(P)
			
คูณ (*)			MELSEC iQ-R/Q/L/iQ-F ซีรีส์: *(P) MELSEC-F ซีรีส์: MUL(P)
การหาร (/)			MELSEC iQ-R/Q/L/iQ-F ซีรีส์: /(P) MELSEC-F ซีรีส์: DIV(P)

3.4 การส่ง/การรับข้อมูลระหว่าง PLC และI/O ดีไวส์

ดิจิตอลสวิตช์อินพุทเป็นอินพุทดีไวส์ที่ใส่ข้อมูลให้ PLCเป็นค่าตัวเลข จอแสดงผลดิจิตอลเป็นเอาท์พุทดีไวส์ที่แสดงผลข้อมูลที่ได้รับมาจาก PLCเป็นค่าตัวเลข

ข้อมูลที่ได้รับมาจากดิจิตอลสวิตช์อินพุทต้องถูกจัดรูปแบบเพื่อให้สามารถประมวลผลได้โดย PLC ในลักษณะเดียวกัน ข้อมูลที่ส่งออกไปยังจอแสดงผลดิจิตอลต้องถูกจัดรูปแบบให้อยู่ในรูปแบบที่สามารถอ่านได้เพื่อการแสดงผลดิจิตอล



3.4.1 การรับข้อมูลจากดิจิตอลสวิตช์อินพุท

มีการใช้คำสั่ง BIN เพื่อให้ PLCรับข้อมูลจากดิจิตอลสวิตช์อินพุท

■ รหัสและฟังก์ชันของคำสั่ง

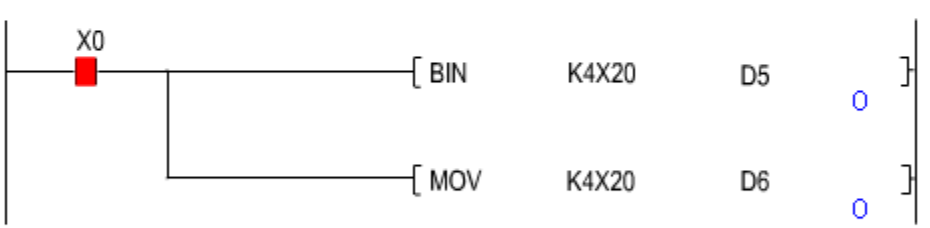
สัญลักษณ์	ฟังก์ชัน
	ข้อมูลในดีไวส์ (S) จะถูกจัดรูปแบบให้อยู่ในรูปแบบที่สามารถประมวลผลได้โดย PLC แล้วจึงจัดเก็บไว้ในดีไวส์ (D)

■ โปรแกรมแลดเดอร์และการทำงาน

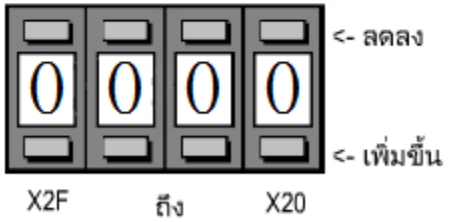
จำลองการทำงานของคำสั่งต่อไปนี้ โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา ตัวเลขสีน้ำเงินแต่ละตัว จะระบุค่า (ค่าปัจจุบัน) ที่จัดเก็บไว้ในดีไวส์

D5 เก็บข้อมูลที่ได้รับจากดิจิตอลสวิตช์อินพุท หลังจากที่ถูกจัดรูปแบบโดยคำสั่ง BIN

D6 เก็บข้อมูลที่ไม่ถูกจัดรูปแบบจากดิจิตอลสวิตช์อินพุท



สวิตช์อินพุทดิจิตอล



หากใช้คำสั่ง MOV ตัวเลขจะไม่ตรงกัน

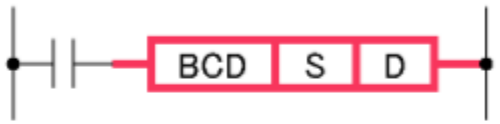
รีเซ็ตการจำลอง

รีเซ็ต

3.4.2 การแสดงผลข้อมูล PLC ไปยังจอแสดงผลดิจิตอล

ในการแสดงผลข้อมูล PLCไปยังจอแสดงผลดิจิตอล จะมีการใช้คำสั่ง BCD

■ รหัสและฟังก์ชันของคำสั่ง

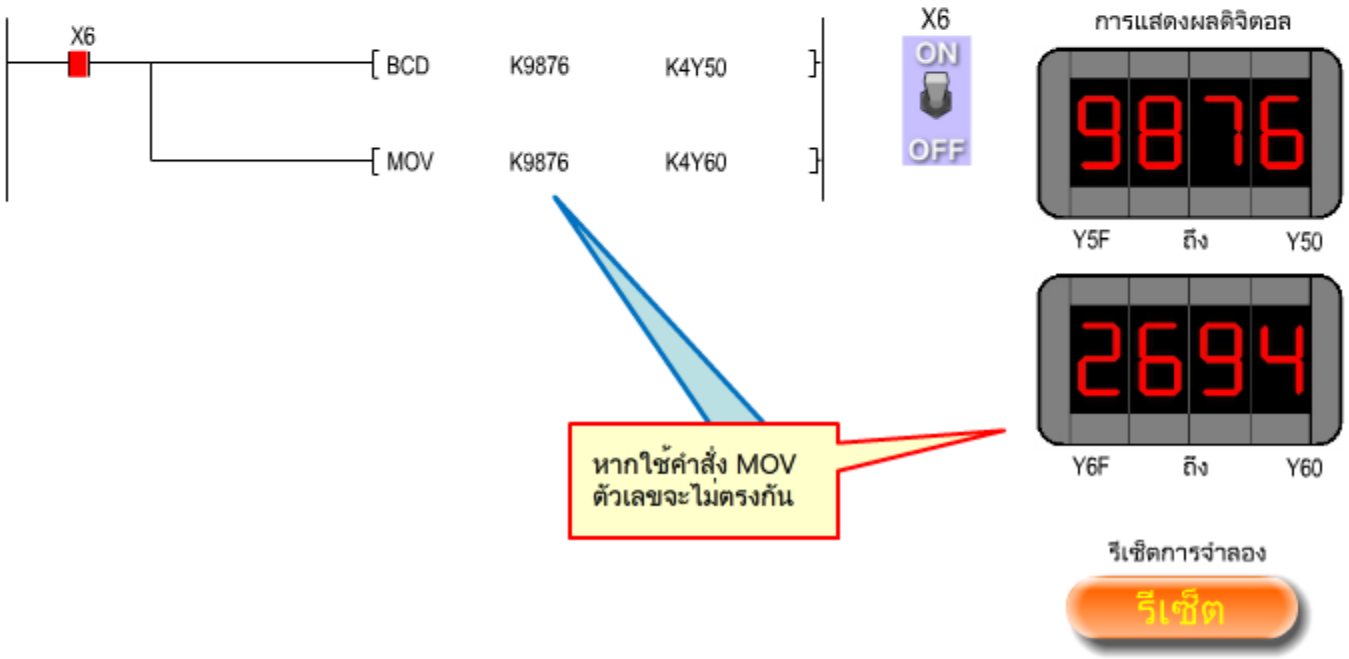
สัญลักษณ์	ฟังก์ชัน
	ข้อมูลในดีไวส์ (S) จะถูกจัดรูปแบบให้อยู่ในรูปแบบที่สามารถแสดงผลไปยังจอแสดงผลดิจิตอล แล้วจึงจัดเก็บไว้ในดีไวส์ (D)

■ โปรแกรมแลดเดอร์และการทำงาน

จำลองการทำงานของคำสั่งต่อไปนี้ โดยการคลิกที่อินพุทสวิตช์ที่ปรากฏทางด้านขวา

จอแสดงผลดิจิตอลตัวบนกำลังแสดงข้อมูลที่ถูกจัดรูปแบบโดยคำสั่ง BCD

จอแสดงผลดิจิตอลตัวล่างกำลังแสดงข้อมูลที่ไม่ถูกจัดรูปแบบ



3.5 สรุป

ในหลักสูตรนี้ คุณได้เรียนรู้เกี่ยวกับ:

- แนวคิดของการอินพุทและเอาท์พุทไปยัง/จาก PLC
- คำสั่งหลักที่ใช้ในการควบคุม PLC
- ข้อมูลที่ได้รับจาก PLC MELSEC จะถูกดำเนินการในโปรแกรมแลดเดอร์ใน PLC จากนั้นผลการดำเนินการจะถูกส่งออกภายนอกเป็นเอาท์พุท
- ความแตกต่างของรูปแบบข้อมูลบิทและเวิร์ด
- พื้นฐานเกี่ยวกับโปรแกรมควบคุม

โปรดเรียนหลักูตร "ซอฟต์แวร์สำหรับแก้ไขโปรแกรม MELSOFT GX Works3 (แลดเดอร์)" เพื่อเรียนรู้วิธีการแก้ไขและลงทะเบียนโปรแกรมไปยังโมดูล CPU MELSEC iQ-R/iQ-F ซีรีส์

โปรดเรียนหลักูตร "พื้นฐาน GX Works2" เพื่อเรียนรู้วิธีการแก้ไขและลงทะเบียนโปรแกรมไปยังโมดูล CPU MELSEC-Q/L/F iCPU ซีรีส์

ทดสอบ แบบทดสอบประเมินผล

คุณได้จบบทเรียนทั้งหมดของหลักสูตรพื้นฐานเกี่ยวกับ พื้นฐานการเขียนโปรแกรม PLC แล้ว คุณพร้อมที่จะทำแบบทดสอบท้ายหลักสูตรแล้ว หากคุณยังไม่มั่นใจเกี่ยวกับหัวข้อต่างๆ ที่จะทดสอบ โปรดทบทวนหัวข้อเหล่านั้น
คำถามในแบบทดสอบประเมินผลนี้มีทั้งหมด 11 ข้อ (54 รายการ)
คุณสามารถทำแบบทดสอบประเมินผลได้หลายครั้งตามต้องการ

วิธีการตอบคำถามในแบบทดสอบ

หลังจากเลือกคำตอบแล้ว ให้คลิกปุ่ม ตอบ คำตอบของคุณจะหายไปถ้าคุณดำเนินการต่อโดยไม่คลิกปุ่ม ตอบ (จะถือว่าคุณยังไม่ได้ตอบคำถามนั้น)

ผลคะแนน

จำนวนคำตอบที่ถูกต้อง จำนวนคำถาม เปอร์เซ็นต์คำตอบที่ถูกต้อง และผลลัพธ์ที่แสดงว่าผ่าน/ไม่ผ่านจะปรากฏบนหน้าผลคะแนน

คำตอบที่ถูกต้อง : 11

จำนวนคำถามทั้งหมด : 11

เปอร์เซ็นต์ : 100%

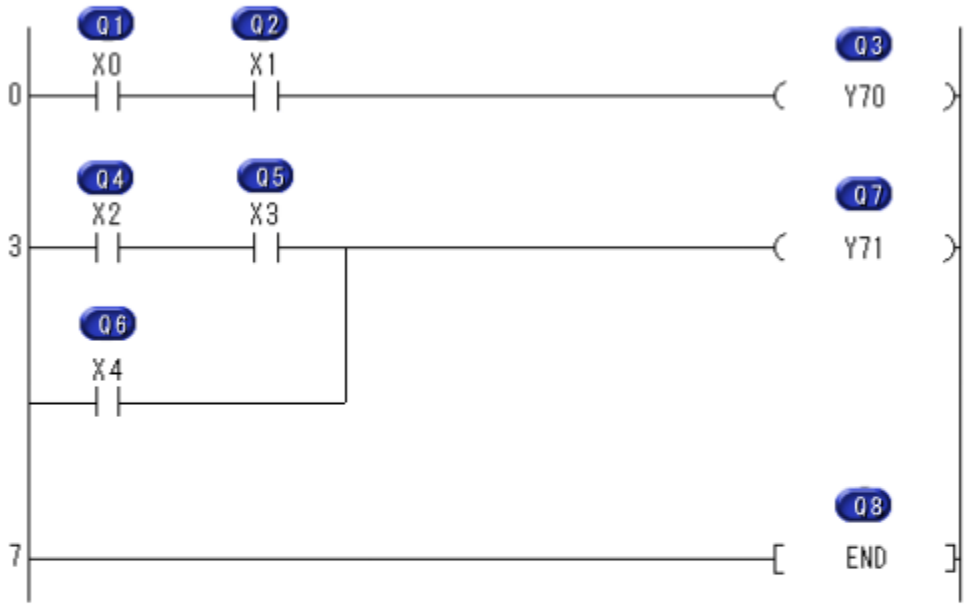
คุณต้องตอบคำถามถูกต้องเกินกว่า 60% จึงจะผ่านการทดสอบ

ดำเนินการต่อ | ทบทวน

- คลิกปุ่ม ดำเนินการต่อ เพื่อออกจากการทดสอบ
- คลิกปุ่ม ทบทวน เพื่อทบทวนการทดสอบ (ตรวจสอบคำตอบที่ถูกต้อง)
- คลิกปุ่ม ลองใหม่ เพื่อทำการทดสอบใหม่อีกครั้ง

ทดสอบ แบบทดสอบประเมินผล 1

โปรดเรียงลำดับคำสั่งต่อไปนี้ตามลำดับการประมวลผล



Q1 --Select-- Q2 --Select-- Q3 --Select-- Q4 --Select--

Q5 --Select-- Q6 --Select-- Q7 --Select-- Q8 --Select--

คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 2

ประโยคด้านล่างจะบรรยายเกี่ยวกับอุปกรณ์ I/O ภายนอก และสัญญาณ I/O ถึง/จากตัวควบคุมแบบตั้งโปรแกรมได้ โปรดทำให้ประโยคสมบูรณ์โดยการเลือกคำที่ถูกต้อง

1) หมายเลขอินพุทและเอาท์พุทของตัวควบคุมแบบตังโปรแกรมได้ Q ซีรีส์ จะเริ่มต้นจาก

(--Select--) และอยู่ในรูปแบบค่า (--Select--)

2) หมายเลขเดียวกันนี้จะถูกนำไปใช้กับสัญญาณอินพุทและเอาท์พุท ดังนั้น อินพุทจะมี

(--Select--) นำหน้า และเอาท์พุทจะมี (--Select--) นำหน้า

3) หมายเลขที่กำหนดให้อินพุทสัญญาณจากอุปกรณ์ภายนอก

จะถูกกำหนดตามเงื่อนไขต่อไปนี้:

- เมื่อมีการติดตั้ง (--Select--) ไว้บน

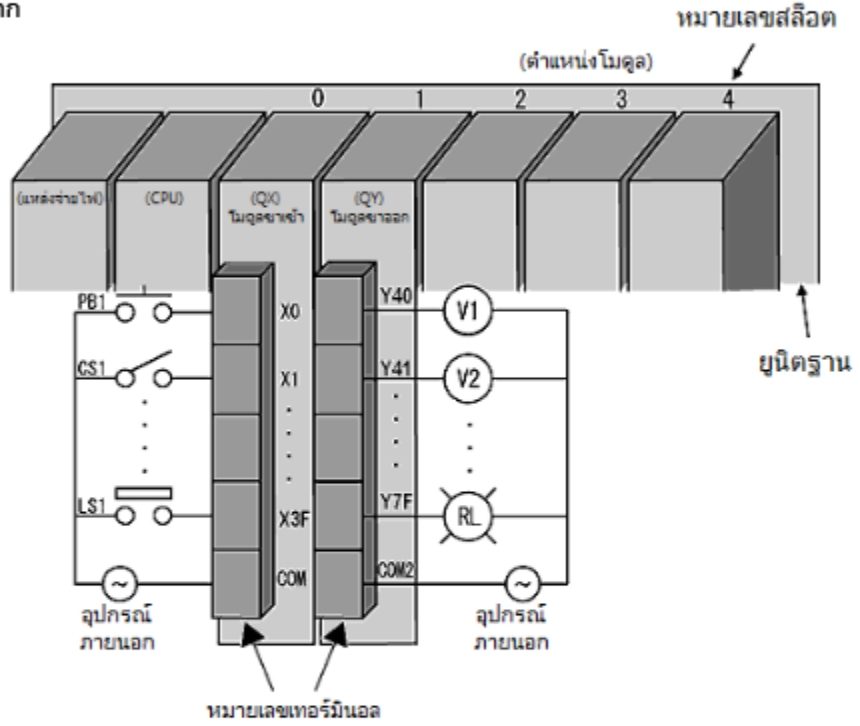
- หมายเลขเทอร์มินอล ในยูนิตฐาน

4) หมายเลขที่กำหนดให้เอาท์พุท (ขดลวด) ไปยังอุปกรณ์ภายนอก

จะถูกกำหนดตามเงื่อนไขต่อไปนี้:

-เมื่อมีการติดตั้ง (--Select--) ไว้บน

-หมายเลขเทอร์มินอล ในยูนิตฐาน



คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 3

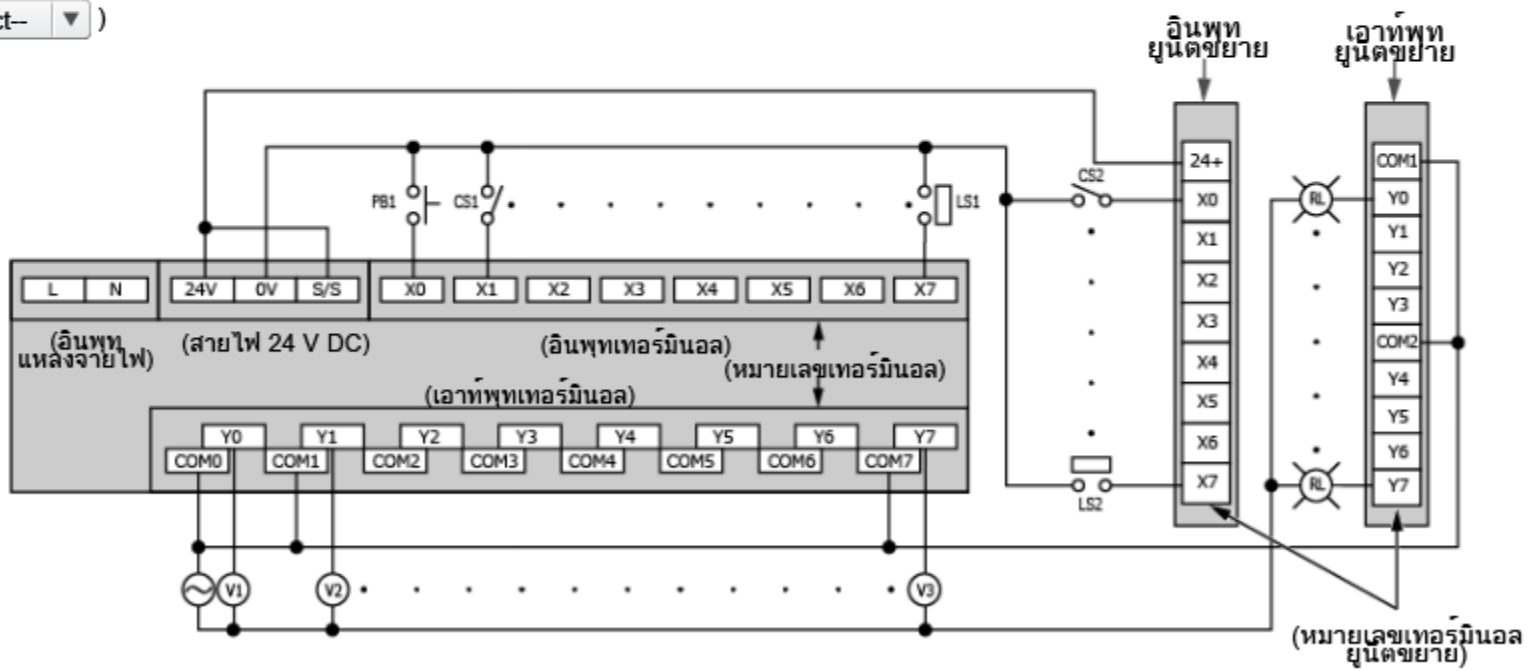
ประโยคด้านล่างจะบรรยายเกี่ยวกับอุปกรณ์ I/O ภายนอก และหมายเลข I/O
ที่กำหนดให้กับตัวควบคุมแบบตั้งโปรแกรมได้ โปรดทำให้ประโยคสมบูรณ์โดยการเลือกคำที่ถูกต้อง (MELSEC-F ซีรีส์)

1) หมายเลข I/O ของตัวควบคุมแบบตั้งโปรแกรมได้ MELSEC-F ซีรีส์ จะเริ่มต้นจาก (--Select--) และจะอยู่ในรูปแบบค่า (--Select--)

2) หมายเลขเดียวกันนี้จะถูกนำไปใช้กับสัญญาณอินพุทและเอาท์พุท ดังนั้น อินพุทจะมี (--Select--) นำหน้า และเอาท์พุทจะมี (--Select--) นำหน้า

3) หากมีการเพิ่มยูนิตขยาย I/O ยูนิตดังกล่าวจะได้รับหมายเลขที่อยู่หลังจากหมายเลขที่กำหนดให้ (--Select--) ก่อนหน้า

4) หมายเลข I/O ของยูนิตจะเริ่มต้นจากหมายเลขที่มี "0" ในหลักแรกเสมอ หากหมายเลข I/O ของยูนิตก่อนหน้าลงท้ายด้วย X17 หมายเลข I/O ของยูนิตถัดไปจะเริ่มต้นจาก (--Select--)

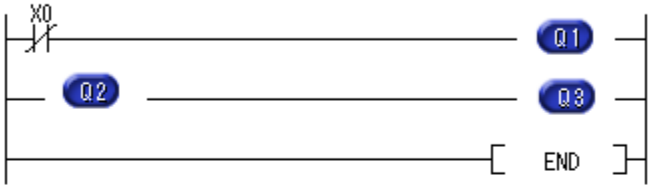


คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 4

ลากและวางคำสั่งที่เหมาะสม เพื่อให้โปรแกรมปฏิบัติงานต่อไปนี้ได้อย่างสมบูรณ์:

เมื่อสวิตช์ X0 อยู่ที่ OFF, หลอดไฟ A จะติดสว่าง (ON) (Y70 อยู่ที่ ON)
เมื่อสวิตช์อยู่ที่ ON หลอดไฟ B จะติดสว่าง (ON) (Y71 อยู่ที่ ON)



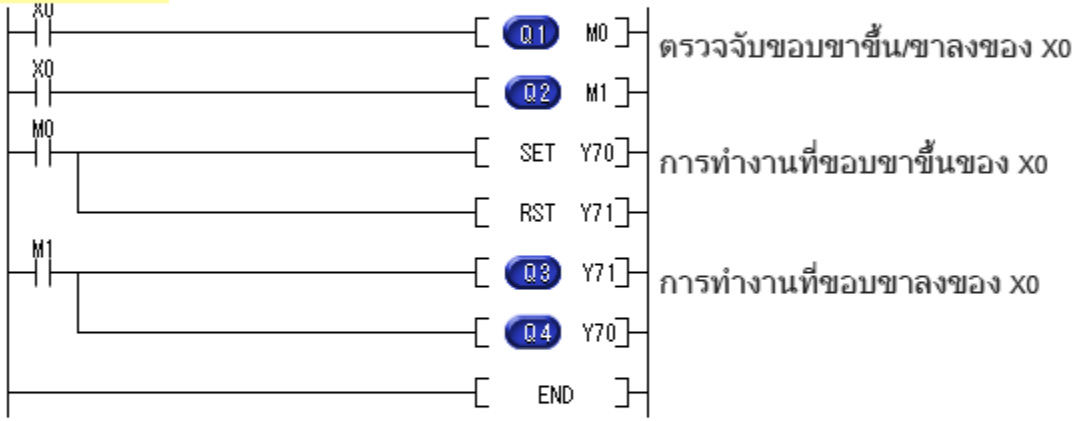
Q1 --Select-- Q2 --Select-- Q3 --Select--

คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 5

ลากและวางคำสั่งที่เหมาะสม เพื่อให้โปรแกรมปฏิบัติงานต่อไปนี้ได้อย่างสมบูรณ์:

ในระหว่างที่กำลังประมวลผลข้อมูล “สัญญาณแสดงการประมวลผล” (X0) จะอยู่ที่ ON
ที่ขอบขาขึ้นของ “สัญญาณแสดงการประมวลผล” (X0), หลอดไฟ A จะติดสว่าง (ON) (Y70 อยู่ที่ ON) และหลอดไฟ B จะดับ (OFF) (Y71 อยู่ที่ OFF)
ที่ขอบขาลงของ “สัญญาณแสดงการประมวลผล” (X0), หลอดไฟ B จะติดสว่าง (ON) (Y70 อยู่ที่ ON) และหลอดไฟ A จะดับ (OFF) (Y71 อยู่ที่ OFF)



Q1 --Select-- Q2 --Select-- Q3 --Select-- Q4 --Select--

คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 6

ลากและวางคำสั่งที่เหมาะสม เพื่อให้โปรแกรมปฏิบัติงานต่อไปนี้ได้อย่างสมบูรณ์:

หลอดไฟจะติดสว่าง (ON) ตามการ ON/OFF ของสวิตช์เริ่มการทำงาน (X0) และสวิตช์เริ่มการประมวลผล (X1)
ในเวลา 2 วินาที หลังจากที่เปิด (ON) สวิตช์ทั้งคู่ หลอดไฟ D จะติดสว่าง (ON)

[เริ่มการทำงาน (X0)]	[สวิตช์เริ่มการประมวลผล (X1)]	[หลอดไฟ]
OFF	OFF	หลอดไฟ A (Y70 อยู่ที่ ON)
ON	OFF	หลอดไฟ B (Y71 อยู่ที่ ON)
ON	ON	หลอดไฟ C (Y72 อยู่ที่ ON) และหลังจากผ่านไป 2 วินาที หลอดไฟ D (Y73 อยู่ที่ ON)

Q1 --Select-- Q2 --Select-- Q3 --Select-- Q4 --Select--

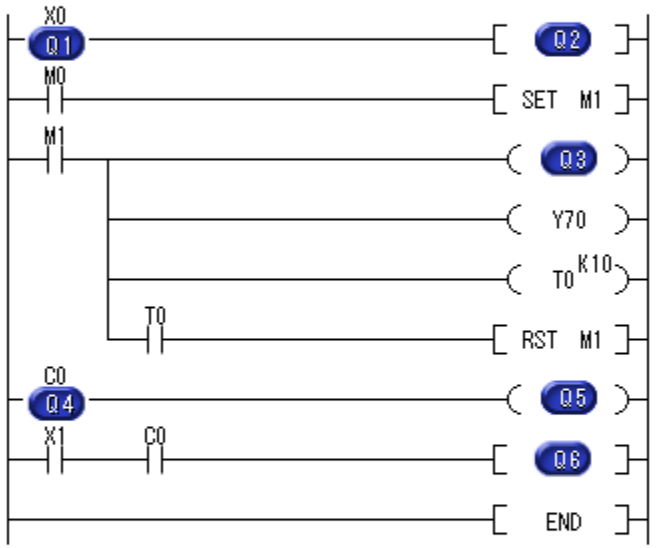
Q5 --Select-- Q6 --Select-- Q7 --Select--

คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 7

ลากและวางคำสั่งที่เหมาะสม เพื่อให้โปรแกรมปฏิบัติงานต่อไปนี้ได้อย่างสมบูรณ์:

เมื่อผลิตภัณฑ์เคลื่อนที่ผ่านบนสายพานลำเลียง สัญญาณ X0 จะเปิด (ON)
หลังจากที่ผลิตภัณฑ์ผ่านไปแล้ว (หลัง 3 วินาที) หลอดไฟ A จะติดสว่าง (ON) (Y70 จะอยู่ที่ ON เป็นเวลา 1 วินาที)
หลังจากผลิตภัณฑ์ผ่านไป 5 รายการ หลอดไฟ B จะติดสว่าง (ON) (Y71 อยู่ที่ ON)
หลังจากที่หลอดไฟ B ติดสว่าง (ON) สวิตช์ยืนยัน (X1) จะเปิด (ON)



X0 จะเปิด (ON) เมื่อตรวจพบผลิตภัณฑ์เคลื่อนที่ผ่าน

ตั้งค่าธงควบคุม

นับจำนวนผลิตภัณฑ์ที่เคลื่อนที่ผ่าน

Y70 จะเปิด (ON) เมื่อผลิตภัณฑ์เคลื่อนที่ผ่าน

ตรวจวัดระยะเวลาที่สัญญาณติด (ON) (1 วินาที)

สิ้นสุดการแจ้งการผ่านของผลิตภัณฑ์

แจ้งว่ามีผลิตภัณฑ์ผ่านไป 5 รายการ

รีเซ็ตจำนวน

Q1 --Select-- Q2 --Select-- Q3 --Select-- Q4 --Select--

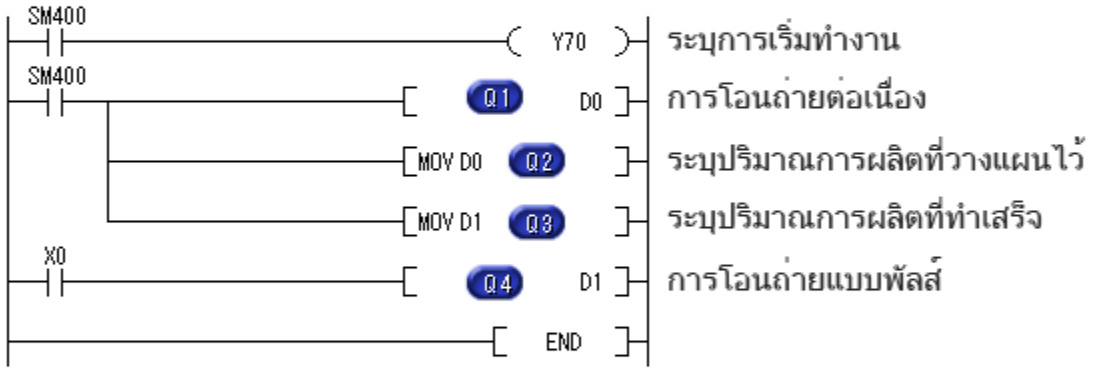
Q5 --Select-- Q6 --Select--

คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 8

ลากและวางคำสั่งที่เหมาะสม เพื่อให้โปรแกรมปฏิบัติงานต่อไปนี้ได้อย่างสมบูรณ์:

1) เมื่อเริ่มการทำงาน หลอดไฟ A จะติดสว่าง (ON) (Y70 อยู่ที่ ON)
2) มีการอินพุทปริมาณการผลิตที่วางแผนไว้โดยใช้สวิตช์ดิจิตอล (X20-X2F) เมื่อใดก็ตามที่มีการอินพุทข้อมูล ปริมาณดังกล่าวจะถูกโอนไปยังการลงทะเบียนข้อมูล D0
3) ค่าที่บันทึกไว้ในการลงทะเบียนข้อมูล (D0, D1) จะถูกโอนและอัพเดตอย่างต่อเนื่องบนจอแสดงผลดิจิตอล ตามที่แสดงด้านล่าง
   Y40-Y4F: ระบุปริมาณการผลิตที่วางแผนไว้(D0)
   Y50-Y5F: ระบุปริมาณการผลิตที่ทำเสร็จ (D1)
4) มีการใช้สวิตช์ดิจิตอล X30 ถึง X3F ในการอินพุทปริมาณการผลิตที่ทำเสร็จ เมื่อสวิตช์เสร็จสิ้นการตั้งค่า (X0) เปิด (ON) ปริมาณการผลิตที่ทำเสร็จจะถูกส่งไปยังการลงทะเบียนข้อมูล D1



* ในโปรแกรมนี้ จะใช้คำสั่ง MOV ในการโอนถ่ายข้อมูล
* ในการตรวจสอบ D0 และ D1 ให้ใช้คาฐานสิบหก

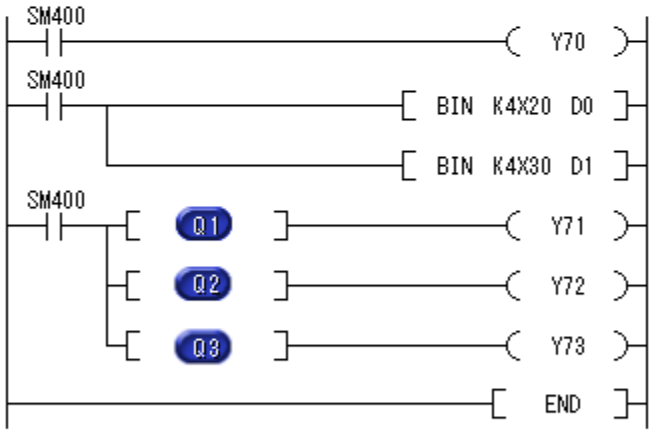
Q1 --Select-- Q2 --Select-- Q3 --Select-- Q4 --Select--

คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 9

ลากและวางคำสั่งที่เหมาะสม เพื่อให้โปรแกรมปฏิบัติงานต่อไปนี้ได้อย่างสมบูรณ์:

1) เมื่อเริ่มการทำงาน หลอดไฟ A จะติดสว่าง (ON) (Y70 อยู่ที่ ON)
2) ระบบจะดำเนินการต่อไปนี้
   - ปริมาณการผลิตที่วางแผนไว้ A ซึ่งทำการอินพุทโดยใช้สวิตช์ดิจิตอล (X20-X2F) จะถูกจัดรูปแบบและโอนไปยังการลงทะเบียนข้อมูล D0
   - ปริมาณการผลิตที่วางแผนไว้ B ซึ่งทำการอินพุทโดยใช้สวิตช์ดิจิตอล (X30-X3F) จะถูกจัดรูปแบบและโอนไปยังการลงทะเบียนข้อมูล D1
   - การลงทะเบียนข้อมูล D0 และ D1 จะถูกนำไปเปรียบเทียบระหว่างกันและกัน และจะมีการระบุผลลัพธ์ด้วยหลอดไฟ
     D0>D1: หลอดไฟ B (Y71 อยู่ที่ ON/OFF)
     D0=D1: หลอดไฟ C (Y72 อยู่ที่ ON/OFF)
     D0<D1: หลอดไฟ D (Y73 อยู่ที่ ON/OFF)



ระบุการเริ่มทำงาน

การรับข้อมูลอินพุท

เปรียบเทียบและระบุผลลัพธ์

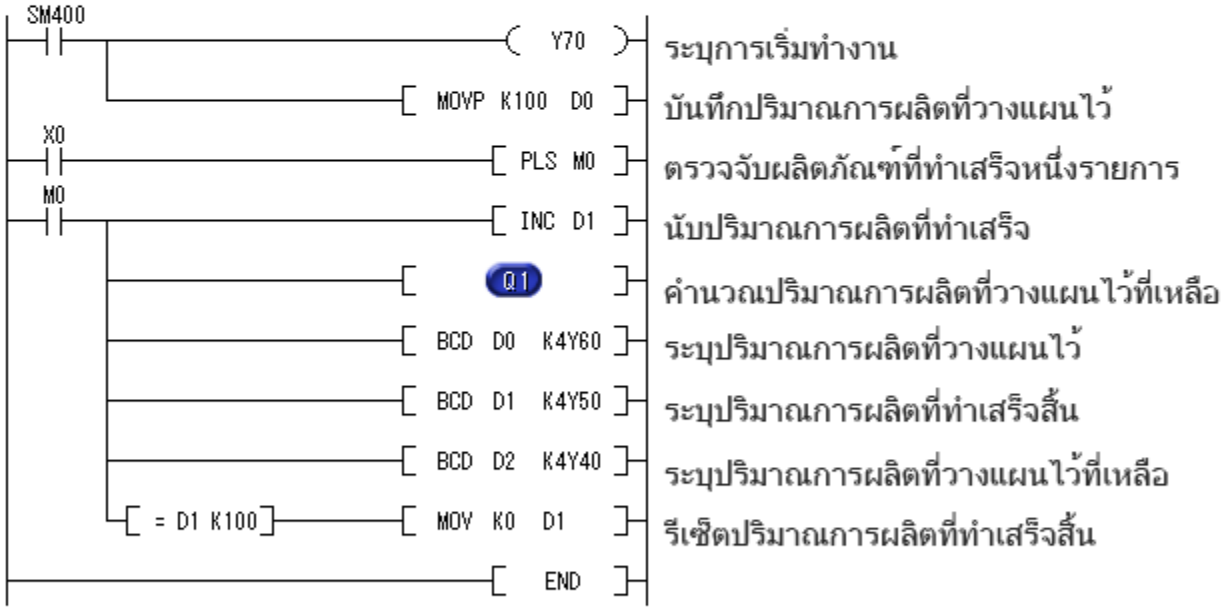
Q1 --Select-- Q2 --Select-- Q3 --Select--

คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 10

ลากและวางคำสั่งที่เหมาะสม เพื่อให้โปรแกรมปฏิบัติงานต่อไปนี้ได้อย่างสมบูรณ์:

1) เมื่อเริ่มการทำงาน หลอดไฟ A จะติดสว่าง (ON) (Y70 อยู่ที่ ON)
2) ขณะเริ่มต้น จำนวนการผลิตที่วางแผนไว้ 100 จะถูกบันทึกไว้ในการลงทะเบียนข้อมูล D0
3) ทุกครั้งที่ผลิตผลิตภัณฑ์เสร็จหนึ่งรายการ ข้อมูลต่อไปนี้จะถูกบันทึกไว้ในการลงทะเบียนข้อมูล
   D1: ปริมาณการผลิตที่ทำเสร็จสิ้น (นับที่ขอบขาขึ้นของ X0)
   D2: ปริมาณการผลิตที่วางแผนไว้ที่เหลือ (D2=D0-D1)
   จอแสดงผลดิจิตอลจะแสดงข้อมูลต่อไปนี้
   Y40-Y4F: ค่าใน D2 (ปริมาณการผลิตที่วางแผนไว้ที่เหลือ (0 ถึง 100))
   Y50-Y5F: ค่าใน D1 (ปริมาณการผลิตที่ทำเสร็จสิ้น (0 ถึง 100))
   Y60-Y6F: ค่า D0 (ปริมาณการผลิตที่วางแผนไว้ (100))



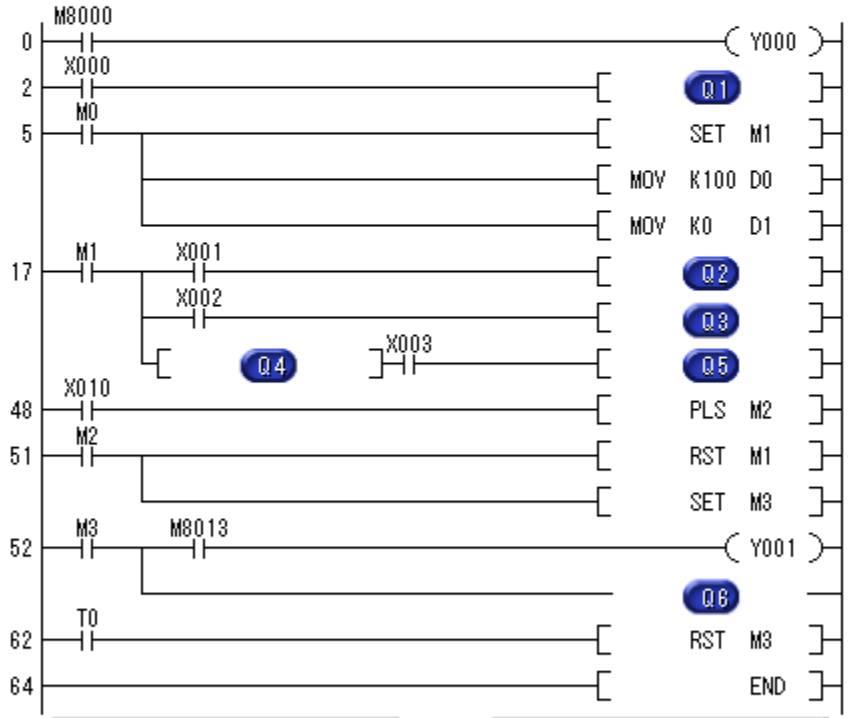
Q1 --Select--

คำตอบ ย้อนกลับ

ทดสอบ แบบทดสอบประเมินผล 11

โปรแกรมควบคุมที่ระบบด้านล่าง มีไว้สำหรับ MELSEC-F ซีรีส์ และจะมีคำสั่งและรีเลย์พิเศษ
ลากและวางคำสั่งที่เหมาะสม เพื่อให้โปรแกรมซึ่งจะทำการเทน้ำร้อนจากตัวจ่ายน้ำร้อนทำงานอย่างสมบูรณ์:

1) เมื่อเริ่มการทำงาน หลอดไฟจะติดสว่าง (ON) (Y0 อยู่ที่ ON)
2) ที่ขอบขาขึ้นของการเริ่มทำงานของตัวจ่ายน้ำร้อน (X0 เปิด (ON)) ระบบจะบันทึกค่า "100" ไว้ในปริมาณการจ่ายน้ำมาตรฐาน D0 และบันทึก "0" ไว้ในเอาท์พุทของปริมาณน้ำร้อน D1
   (รีเซ็ตขอบล)
3) เลือกปริมาณน้ำร้อนเอาท์พุท
   ที่ขอบขาขึ้นของ X1 ปริมาณเอาท์พุทมาตรฐาน D0 จะถูกบันทึกไว้ในปริมาณน้ำร้อนเอาท์พุท D1
   ที่ขอบขาขึ้นของ X2 ครึ่งหนึ่งของปริมาณเอาท์พุทมาตรฐาน D0 จะถูกบันทึกไว้ในปริมาณน้ำร้อนเอาท์พุท D1
4) หากเลือกปริมาณน้ำร้อนเอาท์พุท D1 และมีค่า 0 หรือมากกว่า ระบบจะเพิ่ม "+10" ลงในปริมาณน้ำร้อนเอาท์พุท D1 ที่ ขอบขาขึ้น X3
   จากนั้นค่าที่เพิ่มจะถูกบันทึกไว้ในปริมาณน้ำร้อนเอาท์พุท
5) ที่ขอบขาขึ้นของเอาท์พุทน้ำร้อน (X10) หลอดไฟการเทจะกะพริบเว้นช่วง 1 วินาที (Y1 จะ ON/OFF ซ้ำ) และจะนับเวลาเอาท์พุทน้ำร้อน 5 วินาที (T0)
6) หยุดการกะพริบของหลอดไฟการเท หลังจากนับเวลาเอาท์พุทน้ำร้อน (T0)



Q1 --Select-- Q2 --Select-- Q3 --Select-- Q4 --Select--

Q5 --Select-- Q6 --Select--

คำตอบ ย้อนกลับ

ทดสอบ คะแนนทดสอบ

คุณทำแบบทดสอบประเมินผลเสร็จแล้ว ผลลัพธ์ของคุณแบ่งส่วนได้ดังนี
ในการสิ้นสุดแบบทดสอบประเมินผล ให้ไปต่อยังหน้าถัดไป

คำตอบที่ถูกต้อง : 11

จำนวนคำถามทั้งหมด : 11

เปอร์เซ็นต์ : 100%

ดำเนินการต่อ ทบทวน

ขอแสดงความยินดี คุณผ่านการทดสอบแล้ว

คุณได้ผ่านหลักสูตรพื้นฐานเกี่ยวกับ พื้นฐานการเขียนโปรแกรม PLC แล้ว

ขอขอบคุณสำหรับการเรียนรู้หลักสูตรนี้

ราหวังว่าคุณจะเพลิดเพลินกับบทเรียน และข้อมูลที่คุณได้รับจากหลักสูตรนี้จะเป็นประโยชน์ในอนาคต

คุณสามารถทบทวนหลักสูตรได้หลายครั้งตามต้องการ

ทบทวน ปิด