	หน่วยที่ 9 คำสั่ง Move	สอนครั้งที่ 10
	รหัสวิชา 30127-2005 วิชาโปรแกรมเมเบิลคอนโทรลเลอร์	จำนวน 5 ชั่วโมง

สาระการเรียนรู้

- 1) คำสั่ง Move (MOV)
- 2) คำสั่ง Move Pulse (MOVP)

แนวคิดสำคัญ

ในการออกแบบโปรแกรมเพื่อควบคุมกระบวนการทำงานนั้น บางครั้งการออกแบบโดยใช้คำสั่งพื้นฐานเพียงอย่างเดียวไม่เพียงพอ จำเป็นต้องเรียนรู้โดยใช้คำสั่งอื่นเพิ่มเติม คำสั่งเคลื่อนย้ายข้อมูลเป็นอีกคำสั่งหนึ่งที่ผู้ใช้ควรศึกษา เพื่อเพิ่มทักษะ ความชำนาญในการออกแบบโปรแกรมให้สามารถออกแบบโปรแกรมด้วยโปรแกรมที่มีความหลากหลายมากขึ้น

จุดประสงค์การเรียนรู้

จุดประสงค์ทั่วไป

- 1) เพื่อให้มีความรู้ เข้าใจเกี่ยวกับโปรแกรมคำสั่ง Move
- 2) เพื่อให้มีทักษะเกี่ยวกับโปรแกรมคำสั่ง Move
- 3) เพื่อให้ผู้เรียนเป็นผู้มีคุณธรรมจริยธรรมและคุณลักษณะอันพึงประสงค์สอดคล้องกับ จรรยาบรรณวิชาชีพ

จุดประสงค์เชิงพฤติกรรม

ด้านความรู้

- 1) อธิบายหลักการทำงานของโปรแกรมที่ใช้คำสั่ง Move (MOV) ได้ถูกต้อง
- 2) อธิบายหลักการทำงานของโปรแกรมที่ใช้คำสั่ง Move Pulse (MOVP) ได้ถูกต้อง

ด้านทักษะ

- 1) เขียนโปรแกรมโดยใช้คำสั่ง Move (MOV) ได้ถูกต้อง
- 2) เขียนโปรแกรมโดยใช้คำสั่ง Move Pulse (MOVP) ได้ถูกต้อง
- 3) ออกแบบโปรแกรมโดยใช้กลุ่มคำสั่ง Move ได้ถูกต้อง
- 4) ออกแบบโปรแกรมโดยใช้กลุ่มคำสั่ง Move Pulse ได้ถูกต้อง

ด้านคุณธรรม จริยธรรม และคุณลักษณะที่พึงประสงค์

- 1) มีคุณธรรม จริยธรรมและคุณลักษณะที่พึงประสงค์สอดคล้องกับจรรยาบรรณวิชาชีพ

สมรรถนะประจำหน่วย

- 1) แสดงความรู้ในการเขียนโปรแกรมโดยใช้คำสั่ง Move (MOV)
- 2) แสดงความรู้ในการเขียนโปรแกรมโดยใช้คำสั่ง Move Pulse (MOVP)
- 3) ใช้คำสั่ง Move (MOV) และ Move Pulse (MOVP) ออกแบบโปรแกรมควบคุม

คำแนะนำ

หน่วยที่ 9 กลุ่มคำสั่ง Move ใช้ร่วมกับใบงานที่

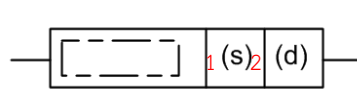
หน่วยที่ 9

คำสั่ง Move

คำสั่ง Move เป็นคำสั่งที่ใช้ ในการเคลื่อนย้ายข้อมูล โดยใช้งานตามขนาดของข้อมูล เป็นการเคลื่อนย้ายข้อมูลจากต้นทาง (Source) ไปยังข้อมูลปลายทาง (Destination) โดยไม่เปลี่ยนแปลงค่าของข้อมูล คำสั่ง Move แบ่งออกได้ 2 ชนิดคือ Move (MOV) และ Move Pulse (MOVP)

9.1 คำสั่ง Move (MOV)

เป็นคำสั่งถ่ายโอนหรือเคลื่อนย้ายข้อมูล ขนาด 16 บิต จากข้อมูลต้นทาง (Source) ไปยัง ข้อมูลปลายทาง (Destination) โดยไม่มีการเปลี่ยนแปลงค่าใดๆ โดยจะส่งข้อมูลจากต้นทางไปยังปลายทางตลอดเวลา เมื่อมีสัญญาณเข้ามาที่คำสั่ง Move



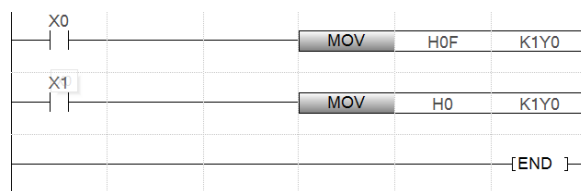
รูปที่ 9.1 แสดงคำสั่ง MOVE

คำอธิบาย

ส่วนที่ 1 (s) คือข้อมูลต้นทาง (Source) ที่ต้องการถ่ายโอนหรือหมายเลขอุปกรณ์จัดเก็บข้อมูล

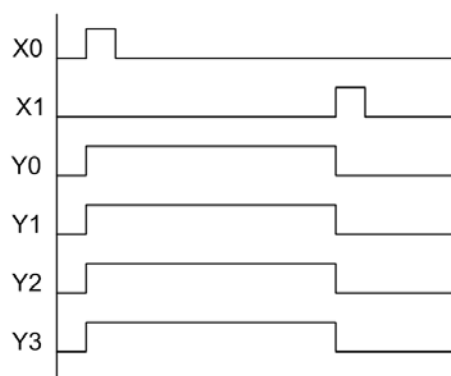
ส่วนที่ 2 (d) คือส่วนข้อมูลปลายทาง (Destination) ที่ทำหน้าที่รับข้อมูล

ตัวอย่างที่ 9.1 การใช้งานคำสั่ง MOVE (MOV)



รูปที่ 9.2 แสดงคำสั่ง MOVE

Timing Diagram



รูปที่ 9.3 Timing Diagram

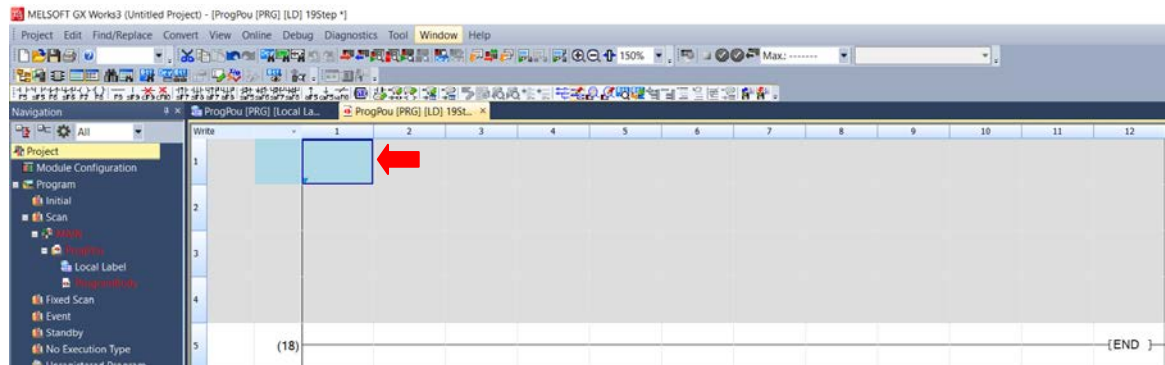
การทำงานของโปรแกรม

เมื่อโยกสวิตช์ X0 จะทำให้คำสั่ง MOVE ทำงานเคลื่อนย้ายข้อมูลจากต้นทางไปยังปลายทาง โดยข้อมูลต้นทางเป็นข้อมูลเลขฐานสิบหก เขียนด้วยตัวอักษรตัว HF (H คือ Hexadecimal คือเลขฐานสิบหก) เมื่อพิจารณาในเรื่องการแปลงเลขฐาน 16 ขนาด 1 บิตจะเท่ากับเลขฐานสองขนาด 4 บิต เมื่อ

ส่งข้อมูลเป็นตัวอักษร F ก็เท่ากับส่งข้อมูลเลขฐานสองค่าเป็น 1111 เมื่อข้อมูลต้นทางถูกส่งไปยังปลายทาง ค่าปลายทางซึ่งเป็นค่าคงที่ K1Y0 ซึ่ง K1 หมายถึงข้อมูลขนาด 4 บิต โดยเริ่มต้นบิตแรกกำหนดเป็น Y0 เมื่อนับไป 4 บิต ก็เท่ากับว่ากำหนดที่บิต Y3 – Y0 เมื่อนำข้อมูลต้นทางซึ่งมีค่าเท่ากับ 1111 ส่งออกที่เอาต์พุต Y3 – Y0 ก็จะทำให้เอาต์พุต Y3 – Y0 ทำงาน และเมื่อโยกสวิตช์ X1 จะเคลื่อนย้ายข้อมูลเลขฐานสิบหก เขียนด้วยตัวอักษรตัว H0 คือส่งข้อมูลออกเป็น 0 เมื่อข้อมูลต้นทางถูกส่งไปยังปลายทาง ก็เท่ากับว่าส่งข้อมูล 0 ออกที่เอาต์พุต Y3 – Y0 ก็จะทำให้เอาต์พุต Y3 – Y0 หยุดทำงาน ดังรูปที่ 9.2

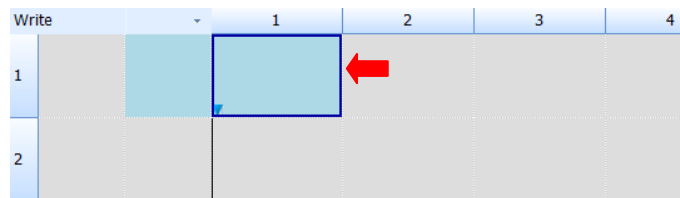
วิธีการเขียนโปรแกรม

1. เปิดโปรแกรม GX Works3 ทำการเชื่อมต่อสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับ PLC จากนั้น เปิดพื้นที่ในการเขียนโปรแกรม ดังรูปที่ 9.4



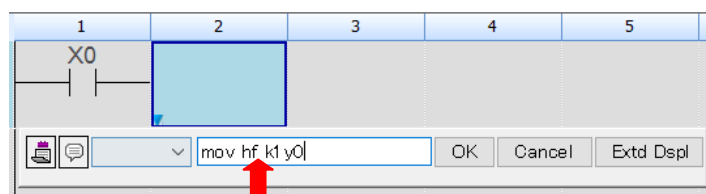
รูปที่ 9.4 เปิดพื้นที่ในการเขียนโปรแกรม

2. คลิกตำแหน่งแรก เพื่อจะเริ่มต้นเขียนโปรแกรม ดังรูปที่ 9.5



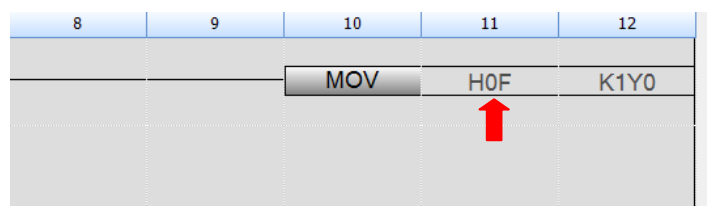
รูปที่ 9.5 คลิกตำแหน่งแรก

3. เขียนโปรแกรมตามโปรแกรมตัวอย่างที่ 9.1 จนถึงคำสั่ง Move พิมพ์ MOV เคาะเว้น 1 ครั้ง พิมพ์ HF เคาะเว้น 1 ครั้ง พิมพ์ K1Y0 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 9.6



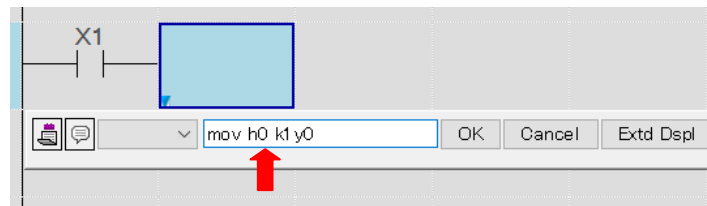
รูปที่ 9.6 พิมพ์ MOV เคาะเว้น 1 ครั้ง พิมพ์ HF เคาะเว้น 1 ครั้ง พิมพ์ K1Y0

4. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 9.7



รูปที่ 9.7 กดปุ่ม Enter

5. เขียนโปรแกรมตามโปรแกรมตัวอย่างที่ 9.1 จนถึงคำสั่ง Move พิมพ์ MOV เคาะเว้น 1 ครั้ง พิมพ์ H0 เคาะเว้น 1 ครั้งพิมพ์ K1Y0 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 9.8



รูปที่ 9.8 พิมพ์ MOV เคาะเว้น 1 ครั้งพิมพ์ H0 เคาะเว้น 1 ครั้งพิมพ์ K1Y0

6. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 9.9

8	9	10	11	12
		MOV	H0F	K1Y0
		MOV	H0	K1Y0

รูปที่ 9.9 กดปุ่ม Enter

7. เขียนโปรแกรมจนครบตามโปรแกรมตัวอย่างที่ 9.1 จากนั้นโหลดโปรแกรมลงใน PLC (Write to PLC) และทดลองการทำงานของโปรแกรมตามขั้นตอน ดังรูปที่ 9.10

Read Mtr		1	2	3	4	5	6	7	8	9	10	11	12
1	(0)	X0									MOV	H0F	K1Y0
2	(8)	X1									MOV	H0	K1Y0
3	(16)												[END]

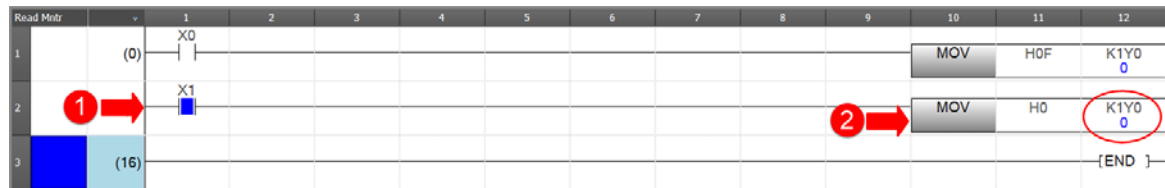
รูปที่ 9.9 ทดลองการทำงานของโปรแกรมตามขั้นตอน

8. ทดลองการทำงานของโปรแกรม โดยการโยกสวิตช์ X0 (1) จะทำให้คำสั่ง MOVE ทำงาน เคลื่อนย้ายข้อมูล H0F ไปที่ K1Y0 ส่งผลทำให้เอาต์พุต Y0 – Y3 ทำงาน (2) และเมื่อ Y0 – Y3 ทำงาน ค่าของข้อมูลจะมีค่าเท่ากับ 15 ดังรูปที่ 9.10

Read Mtr		1	2	3	4	5	6	7	8	9	10	11	12
1	1	X0									MOV	H0F	K1Y0
2	(8)	X1									MOV	H0	K1Y0
3	(16)												[END]

รูปที่ 9.10 ทดลองการทำงานของโปรแกรม โดยการโยกสวิตช์ X0

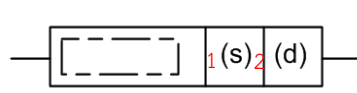
9. ทดลองการทำงานของโปรแกรม โดยการโยกสวิตช์ X1 (1) จะทำให้คำสั่ง MOVE ทำงาน เคลื่อนย้ายข้อมูล H0 ไปที่ K1Y0 ส่งผลทำให้ เอาต์พุต Y0 – Y3 หยุดการทำงาน (2) เมื่อ Y0 – Y3 หยุดการทำงาน ค่าของข้อมูลจะมีค่าเท่ากับ 0 ดังรูปที่ 9.11



รูปที่ 9.11 ทดลองการทำงานของโปรแกรม โดยการโยกสวิตช์ X1

คำสั่ง Move Pulse (MOVP)

เป็นคำสั่งถ่ายโอนหรือเคลื่อนย้ายข้อมูล ขนาด 16 บิต จากข้อมูลต้นทาง (Source) ไปยัง ข้อมูลปลายทาง (Destination) โดยไม่มีการเปลี่ยนแปลงค่าใดๆ โดยจะส่งข้อมูลจากต้นทางไปยังปลายทาง เพียงครั้งเดียว (1 Scan) เมื่อมีสัญญาณเข้ามาที่คำสั่ง Move Pulse



รูปที่ 9.12 แสดงคำสั่ง Move Pulse

คำอธิบาย

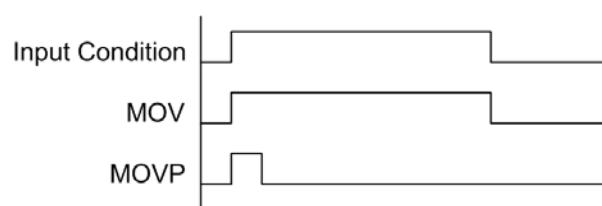
ส่วนที่ 1 (s) คือส่วนของข้อมูลต้นทาง (Source) ที่ต้องการส่งถ่ายหรือเคลื่อนย้ายข้อมูล

ส่วนที่ 2 (d) คือส่วนข้อมูลปลายทาง (Destination) ที่ทำหน้าที่รับข้อมูล

ข้อแตกต่างระหว่างคำสั่ง MOVE (MOV) และ MOVE PULSE (MOVP)

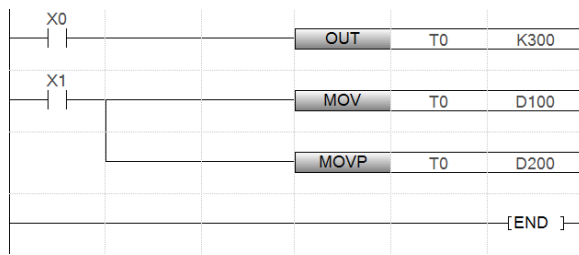
ในการใช้งานคำสั่ง MOVE เมื่อมีสัญญาณ ON เข้ามาที่คำสั่ง MOVE คำสั่ง MOVE จะทำงานและทำงานตลอดเวลาตามจังหวะสัญญาณ ON ที่เข้ามาที่คำสั่ง MOVE และหากไม่มีสัญญาณ (OFF) คำสั่ง MOVE ก็จะหยุดทำงานไปด้วยเช่นกัน แต่ในกรณีของคำสั่ง MOVE PULSE เมื่อมีสัญญาณ ON เข้ามาที่ คำสั่ง MOVE PULSE คำสั่ง MOVE PULSE จะทำงานเพียงครั้งเดียว (1 Pulse) และจะหยุดการทำงาน ทันทีถึงแม้ว่าจะยังมีสัญญาณ ON เข้ามาที่คำสั่ง MOVE PULSE ก็ตาม และจะยอมทำงานอีกครั้งก็ต่อเมื่อ สัญญาณเดิม OFF และมีสัญญาณใหม่ ON เข้ามาอีกครั้ง จึงจะยอมปล่อยสัญญาณออกอีกครั้งหนึ่ง ดังรูป ที่ 9.13

Timing Diagram



รูปที่ 9.13 Timing Diagram

ตัวอย่างที่ 9.2 การใช้งานคำสั่ง Move Pulse (MOVP)



รูปที่ 9.13 Ladder Diagram คำสั่ง Move Pulse

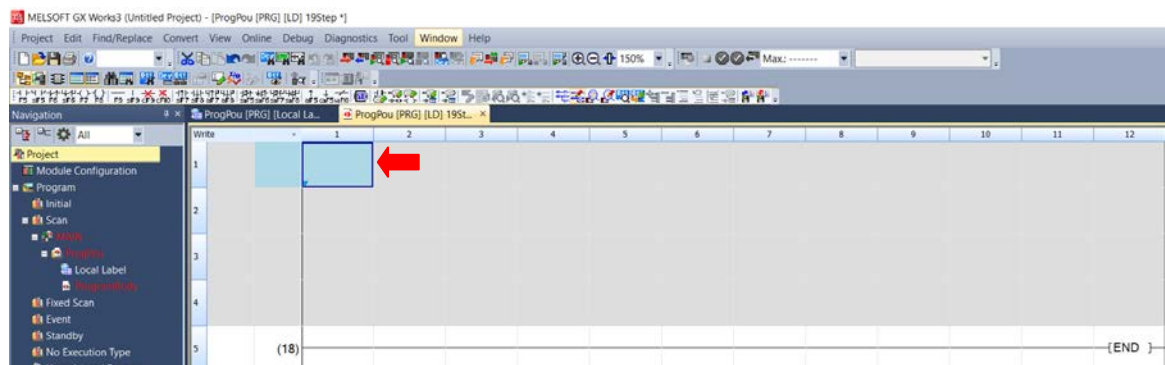
การทำงานของโปรแกรม

จากตัวอย่าง เป็นการใช้งานคำสั่ง MOV และ MOVP พร้อมกัน เพื่อให้เห็นถึงข้อแตกต่างของทั้งสองคำสั่ง โดยใช้สวิตช์ X0 ควบคุมคำสั่ง Timer T0 ตั้งค่าเวลาเท่ากับ 30.0 วินาทีและใช้สวิตช์ X1 ควบคุมคำสั่ง MOV โดยต้นทางรับสัญญาณค่าเวลาจาก Timer T0 และปลายทางส่งไปยังตัวแปรสำหรับเก็บข้อมูล D100 และควบคุม

คำสั่ง MOVP โดยต้นทางรับสัญญาณค่าเวลาจาก Timer T0 และปลายทางส่งไปยังตัวแปรสำหรับเก็บข้อมูล D200 เมื่อโยกสวิตช์ X0 Timer T0 จะเริ่มนับเวลา จากนั้นให้โยกสวิตช์ X1 คำสั่ง MOV และ MOVP จะทำงาน โดยคำสั่ง MOV จะรับค่าสัญญาณจาก Timer T0 ส่งไปยังตัวแปร D100 ตลอดเวลาเพราะเมื่อมีสัญญาณเข้ามาที่คำสั่ง MOV คำสั่ง MOV จะทำงานตลอด แต่ในส่วนของคำสั่ง MOVP จะรับค่าสัญญาณจาก Timer T0 เพียง 1 ครั้ง (1 Scan) เท่านั้น ไม่มีการเปลี่ยนแปลงค่าตามค่าเวลาของ Timer T0 ถึงแม้ว่าจะมีสัญญาณเข้ามาที่คำสั่ง MOVP ตลอดเวลาก็ตามดังรูปที่ 9.13

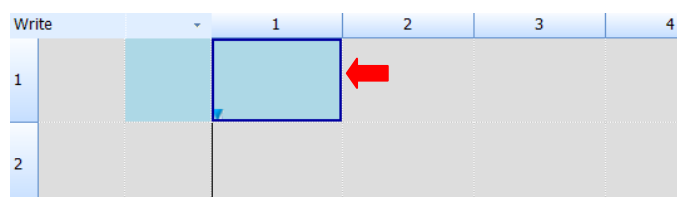
วิธีการเขียนโปรแกรม

1. เปิดโปรแกรม GX Works3 ทำการเชื่อมต่อสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับ PLC จากนั้นเปิดพื้นที่ในการเขียนโปรแกรม ดังรูปที่ 9.14



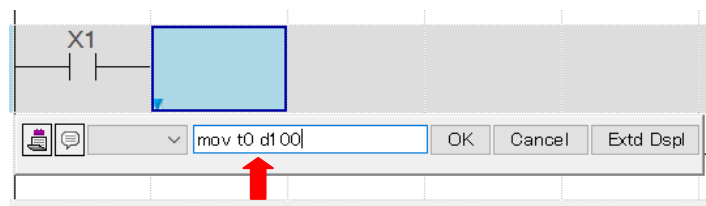
รูปที่ 9.14 เปิดพื้นที่ในการเขียนโปรแกรม

2. คลิกตำแหน่งแรก เพื่อจะเริ่มต้นเขียนโปรแกรม ดังรูปที่ 9.15



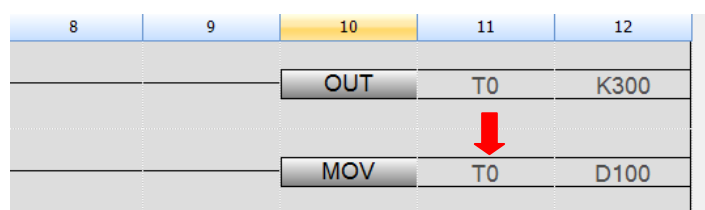
รูปที่ 9.15 คลิกตำแหน่งแรก

3. เขียนโปรแกรมตามโปรแกรมตัวอย่างที่ 9.2 จนถึงคำสั่ง Move พิมพ์ MOV เคาะเว้น 1 ครั้ง พิมพ์ T0 เคาะเว้น 1 ครั้ง พิมพ์ D100 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 9.16



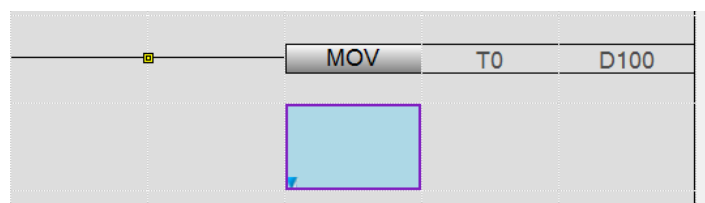
รูปที่ 9.16 พิมพ์ MOV เคาะเว้น 1 ครั้ง พิมพ์ T0 เคาะเว้น 1 ครั้ง พิมพ์ D100

4. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่เขียนโปรแกรม ดังรูปที่ 9.17



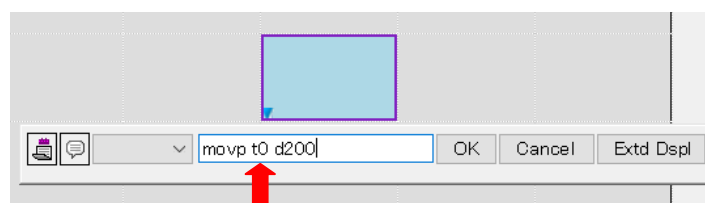
รูปที่ 9.17 กดปุ่ม Enter

5. คลิกที่ตำแหน่งถัดไป ดังรูปที่ 9.18



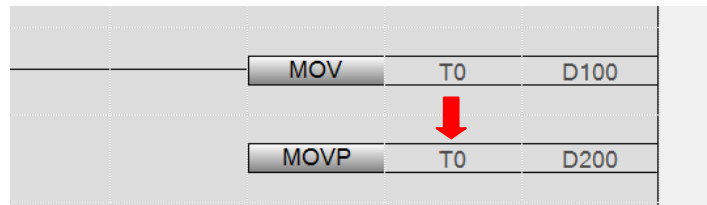
รูปที่ 9.17 กดปุ่ม Enter

6. พิมพ์ MOVP เคาะเว้น 1 ครั้ง พิมพ์ T0 เคาะเว้น 1 ครั้ง พิมพ์ D200 จะปรากฏหน้าต่างต่างขึ้นมา ดังรูปที่ 9.18



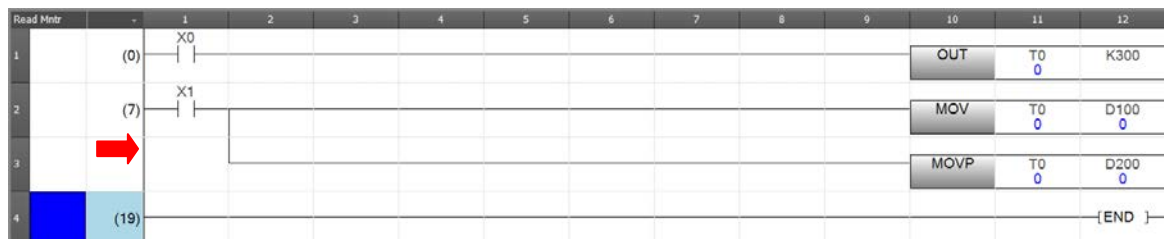
รูปที่ 9.18 พิมพ์ MOVP เคาะเว้น 1 ครั้ง พิมพ์ T0 เคาะเว้น 1 ครั้ง พิมพ์ D200

7. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 9.19



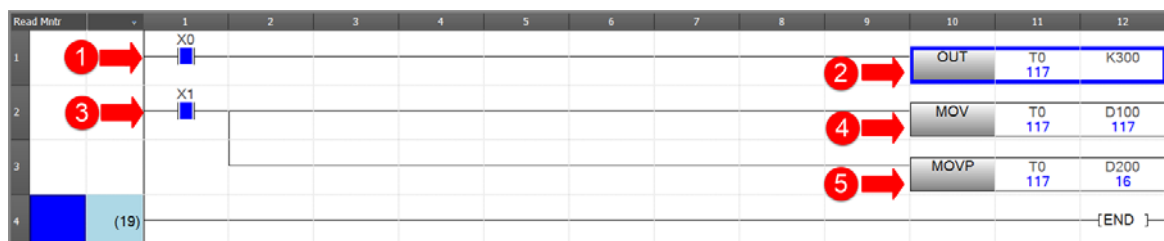
รูปที่ 9.19 กดปุ่ม Enter

8. เขียนโปรแกรมจนครบตามโปรแกรมตัวอย่างที่ 9.2 จากนั้นโหลดโปรแกรมลงใน PLC (Write to PLC) และทดลองการทำงานของโปรแกรมตามขั้นตอน ดังรูปที่ 9.20



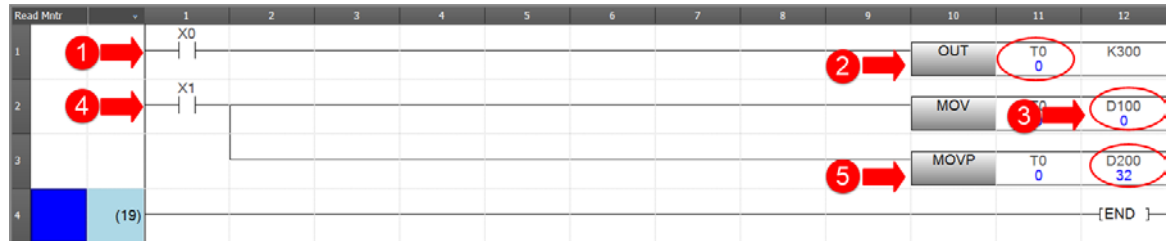
รูปที่ 9.20 ทดลองการทำงานของโปรแกรมตามขั้นตอน

9. ทดลองการทำงานของโปรแกรม โดยการโยกสวิตช์ X0 (1) จะทำให้คำสั่ง TIMER (T0) ทำงาน ค่าเวลาของ TIMER (T0) จะเริ่มนับค่า (2) จากนั้นให้โยกสวิตช์ X1 (3) คำสั่ง Move จะทำการเคลื่อนย้ายข้อมูลคือค่าเวลาของ TIMER (T0) ไปยังตัวแปรสำหรับเก็บข้อมูล D100 ตลอดเวลา (4) ส่วนคำสั่ง Move Pulse จะทำการเคลื่อนย้ายข้อมูลคือค่าเวลาของ TIMER (T0) ไปยังตัวแปรสำหรับเก็บข้อมูล D200 เพียงครั้งเดียวในช่วงจังหวะที่ X1 On ครั้งแรก (5) ดังรูปที่ 9.21



รูปที่ 9.21 ทดลองการทำงานของโปรแกรม โดยการโยกสวิตช์ X0 และ X1

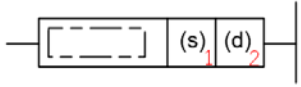
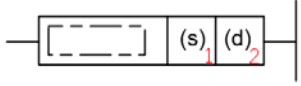
10. โยกสวิตช์ X0 ลง (1) จะทำให้ค่าเวลาของ TIMER (T0) ถูก Reset เป็น 0 (2) และค่าของข้อมูลในตัวแปร D100 จะถูกเปลี่ยนเป็น 0 ด้วยเช่นกัน (3) และเมื่อโยกสวิตช์ X1 ลง (4) ค่าของข้อมูลในตัวแปร D200 จะยังคงค้างข้อมูลเดิม เนื่องจากคำสั่ง Move Pulse ทำงานครั้งแรกเพียงครั้งเดียว ค่าของข้อมูล TIMER (T0) ที่เปลี่ยนเป็น 0 จึงไม่ถูกเคลื่อนย้ายไปที่ D200 ดังรูปที่ 9.22



รูปที่ 9.22 ทดลองการทำงานของโปรแกรม โดยการโยกสวิตช์ X1

สรุป

คำสั่ง Move เป็นคำสั่งที่ใช้ ในการ เคลื่อนย้ายข้อมูล โดยใช้งานตามขนาดของข้อมูล เป็นการเคลื่อนย้ายข้อมูลจากทางด้านอินพุตไปยังด้านเอาต์พุตโดยไม่เปลี่ยนแปลงค่าของข้อมูล คำสั่ง Move แบ่งตามขนาดของข้อมูล

คำสั่ง	สัญลักษณ์	การทำงาน
Move (MOV)		เป็นคำสั่งในการส่งถ่ายหรือเคลื่อนย้ายข้อมูลขนาด 16 บิต จากข้อมูลต้นทาง (Source) ไปยังข้อมูลปลายทาง (Destination) โดยไม่มีการเปลี่ยนแปลงค่าใดๆ โดยจะส่งข้อมูลจากต้นทางไปยังปลายทางตลอดเวลา เมื่อมีสัญญาณ ON เข้ามาที่คำสั่ง Move
Move Pulse (MOVP)		เป็นคำสั่งในการส่งถ่ายหรือเคลื่อนย้ายข้อมูลขนาด 16 บิต จากข้อมูลต้นทาง (Source) ไปยังข้อมูลปลายทาง (Destination) โดยไม่มีการเปลี่ยนแปลงค่าใดๆ โดยจะส่งข้อมูลจากต้นทางไปยังปลายทางเพียงครั้งเดียว (1 Scan) เมื่อมีสัญญาณ ON เข้ามาที่คำสั่ง Move Pulse