	หน่วยที่ 6 กลุ่มคำสั่งพื้นฐาน	สอนครั้งที่ 6
	รหัสวิชา 30127-2005 วิชาโปรแกรมเมเบิลคอนโทรลเลอร์	จำนวน 5 ชั่วโมง

สาระการเรียนรู้

- 1) คำสั่ง Normally Open (NO) คำสั่ง Normally Closed (NC) และ คำสั่ง Output (Output)
- 2) คำสั่ง Set (SET) คำสั่ง Reset (RST) และคำสั่ง Zone Reset (ZRST)
- 3) คำสั่ง Turning On a specified Device for One Scan at the Rising Edge of an Input Condition (PLS) และคำสั่ง Turning On a specified Device for One Scan at the Falling Edge of an Input Condition (PLF)
- 4) คำสั่ง Master Control Set (MC) และ Master Control Reset (MCR)

แนวคิดสำคัญ

กลุ่มคำสั่งพื้นฐาน เป็นกลุ่มคำสั่งเริ่มต้นที่ผู้ใช้งานจะต้องเรียนรู้ทำความเข้าใจ ตั้งแต่การทำงานด้วยหน้าสัมผัสแบบปกติปิดและแบบปกติเปิด การนำหน้าสัมผัสไปควบคุมเอาต์พุต ขั้นตอนการทำงานของโปรแกรม รวมถึงการใช้คำสั่งอื่นๆ ภายในกลุ่มคำสั่งพื้นฐาน เพื่อให้สามารถเข้าใจถึงการทำงานของคำสั่งและวิธีการเขียนโปรแกรม สำหรับใช้เป็นพื้นฐานในการเขียนโปรแกรมต่อไป

จุดประสงค์การเรียนรู้

จุดประสงค์ทั่วไป

- 1) เพื่อให้มีความรู้ เข้าใจเกี่ยวกับการโปรแกรมกลุ่มคำสั่งพื้นฐาน
- 2) เพื่อให้มีทักษะเกี่ยวกับโปรแกรมกลุ่มคำสั่งพื้นฐาน
- 3) เพื่อให้ผู้เรียนเป็นผู้มีคุณธรรมจริยธรรมและคุณลักษณะอันพึงประสงค์สอดคล้องกับ

จรรยาบรรณวิชาชีพ

จุดประสงค์เชิงพฤติกรรม

ด้านความรู้

- 1) อธิบายหลักการทำงานของโปรแกรมที่ใช้คำสั่ง Normally Open (NO) คำสั่ง Normally Closed (NC) และ คำสั่ง Output (Output) ได้ถูกต้อง
- 2) อธิบายหลักการทำงานของโปรแกรมที่ใช้คำสั่ง Set (SET) คำสั่ง Reset (RST) และคำสั่ง Zone Reset (ZRST) ได้ถูกต้อง
- 3) อธิบายหลักการทำงานของโปรแกรมที่ใช้คำสั่ง Turning On a specified Device for One Scan at the Rising Edge of an Input Condition (PLS) และ คำสั่ง Turning On a specified Device for One Scan at the Falling Edge of an Input Condition (PLF) ได้ถูกต้อง
- 4) อธิบายหลักการทำงานของโปรแกรมที่ใช้คำสั่ง Master Control Set (MC) และ Master Control Reset (MCR) ได้ถูกต้อง

ด้านทักษะ

- 1) เขียนโปรแกรมโดยใช้คำสั่ง Normally Open (NO) คำสั่ง Normally Closed (NC) และ คำสั่ง Output (Output) ได้ถูกต้อง
- 2) เขียนโปรแกรมโดยใช้คำสั่ง Set (SET) คำสั่ง Reset (RST) และคำสั่ง Zone Reset (ZRST) ได้ถูกต้อง
- 3) เขียนโปรแกรมโดยใช้คำสั่ง Turning On a specified Device for One Scan at the Rising Edge of an Input Condition (PLS) และ คำสั่ง Turning On a specified Device for One Scan at the Falling Edge of an Input Condition (PLF) ได้ถูกต้อง
- 4) เขียนโปรแกรมโดยใช้คำสั่ง Master Control Set (MC) และ Master Control Reset (MCR) ได้ถูกต้อง

ด้านคุณธรรม จริยธรรม และคุณลักษณะที่พึงประสงค์

- 1) มีคุณธรรม จริยธรรมและคุณลักษณะที่พึงประสงค์สอดคล้องกับจรรยาบรรณวิชาชีพ

สมรรถนะประจำหน่วย

- 1) แสดงความรู้ในการเขียนโปรแกรมโดยใช้คำสั่ง Normally Open (NO) คำสั่ง Normally Closed (NC) และ คำสั่ง Output (Output)
- 2) แสดงความรู้ในการเขียนโปรแกรมโดยใช้คำสั่ง Set (SET) คำสั่ง Reset (RST) และคำสั่ง Zone Reset (ZRST) ได้ถูกต้อง
- 3) แสดงความรู้ในการเขียนโปรแกรมโดยใช้คำสั่ง Turning On a specified Device for One Scan at the Rising Edge of an Input Condition (PLS) และ คำสั่ง Turning On a specified Device for One Scan at the Falling Edge of an Input Condition (PLF)
- 4) แสดงความรู้ในการเขียนโปรแกรมโดยใช้คำสั่ง Master Control Set (MC) และ Master Control Reset (MCR)

คำแนะนำ




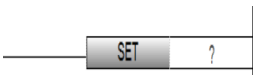
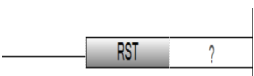
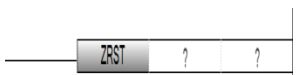

หน่วยที่ 6 กลุ่มคำสั่งพื้นฐานใช้ร่วมกับใบงานที่

หน่วยที่ 6


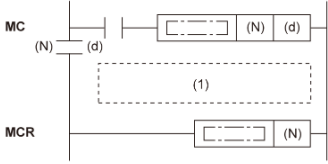
กลุ่มคำสั่งพื้นฐาน

การเขียนโปรแกรม PLC จะมีกลุ่มคำสั่งพื้นฐานให้เลือกใช้สำหรับการเขียนโปรแกรมเบื้องต้น โดยกลุ่มคำสั่งพื้นฐานจะมีชื่อคำสั่งและสัญลักษณ์รวมถึงความหมายของคำสั่ง ดังตารางที่ 6.1

ตารางที่ 6.1 สัญลักษณ์ของกลุ่มคำสั่งพื้นฐาน

ชื่อคำสั่งและสัญลักษณ์	ความหมาย : การทำงาน
 Normally Open (NO)	ความหมาย เป็นคำสั่งหน้าสัมผัสสปกติเปิด การทำงาน หน้าสัมผัสจะเปลี่ยนสถานะจากปกติเปิดเป็นสถานะปิดเมื่อมีการเปลี่ยนสัญญาณอินพุตจาก 0 เป็น 1 หรือจาก Off เป็น On
 Normally Close (NC)	ความหมาย เป็นคำสั่งหน้าสัมผัสสปกติปิด การทำงาน หน้าสัมผัสจะเปลี่ยนสถานะจากปกติปิดเป็นสถานะเปิดเมื่อมีการเปลี่ยนสัญญาณอินพุตจาก 0 เป็น 1 หรือจาก Off เป็น On
 Output	ความหมาย เป็นคำสั่งการแสดงผลของ Output การทำงาน เอาต์พุตจะทำงานเมื่อมีสัญญาณอินพุตเข้าที่เอาต์พุตคอยล์
 Set (SET)	ความหมาย เป็นคำสั่งที่ใช้กระทำการ Set ค่าที่ต้องการบิต (bit) ให้มีสถานะ On การทำงาน เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง Set จะทำให้คำสั่ง Set ทำงานไปกระทำการ Set bit ที่ตั้งค่าไว้ ให้มีสถานะ On
 Reset (RST)	ความหมาย เป็นคำสั่งที่ใช้กระทำการ Reset ค่าที่ต้องการบิต (bit) ให้มีสถานะ Off การทำงาน เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง Reset จะทำให้คำสั่ง Reset ทำงาน ไป Reset bit ที่ตั้งค่าไว้ ให้มีสถานะ Off
 Zone Reset (ZRST)	ความหมาย เป็นคำสั่งที่ใช้กระทำการ Reset แบบกลุ่ม กับค่าที่ต้องการบิต (bit) ให้มีสถานะ Off การทำงาน เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง Zone Reset จะทำให้คำสั่ง Zone Reset ทำงาน ไป Reset กลุ่ม bit ที่ตั้งค่าไว้ ให้มีสถานะ Off
 PLS	ความหมาย เป็นคำสั่งที่หน้าสัมผัสทำงาน (On) ด้วยระยะเวลา 1 Scan time เมื่อเปลี่ยนสัญญาณจาก Off เป็น On (ขอบขาขึ้น) การทำงาน หน้าสัมผัสทำงาน (On) เป็นระยะเวลา 1 Scantime เมื่อเปลี่ยนสัญญาณอินพุตจาก 0 เป็น 1 หรือจาก Off เป็น On (หรือเรียกว่า ช่วงขอบขาขึ้น)

ตารางที่ 6.1 สัญลักษณ์ของกลุ่มคำสั่งพื้นฐาน (ต่อ)

ชื่อคำสั่งและสัญลักษณ์	ความหมาย : การทำงาน
 <p>PLF</p>	<p>ความหมาย เป็นคำสั่งที่หน้าสัมผัสทำงาน (On) ด้วยระยะเวลา 1 Scantime เมื่อเปลี่ยนสัญญาณจาก On เป็น Off (ขอบขาลง)</p> <p>การทำงาน หน้าสัมผัสจะทำงาน On เป็นระยะเวลา 1 Scan time เมื่อเปลี่ยนสัญญาณอินพุตจาก 1 เป็น 0 หรือจาก On เป็น Off (หรือเรียกว่า ช่วงขอบขาลง)</p>
	<p>ความหมาย เป็นคำสั่งควบคุมการทำงานของโปรแกรมซึ่งอยู่ระหว่าง คำสั่ง MC และ MCR โดยขึ้นต้นด้วยคำสั่ง MC และลงท้ายด้วย MCR</p> <p>การทำงาน เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง MC จะทำให้โปรแกรมซึ่งอยู่ระหว่าง MC และ MCR ทำงานเป็นปกติ แต่หากไม่มีสัญญาณเข้าที่คำสั่ง MC จะทำให้โปรแกรมซึ่งอยู่ระหว่าง MC และ MCR ไม่สามารถทำงานได้ จะค้างสภาวะการทำงานเดิมก่อนหน้า</p>

6.1 คำสั่ง Normally Open (NO)

คำสั่ง Normally Open (NO) หรือคำสั่งหน้าสัมผัสปกติเปิด โดยหน้าสัมผัสจะเปลี่ยนสภาวะจากปกติเปิดเป็นสภาวะปิด เมื่อมีการเปลี่ยนสัญญาณอินพุตจาก 0 เป็น 1 ดังรูปที่ 6.1 (ก)

6.2 คำสั่ง Normally Closed (NC)

คำสั่ง Normally Closed (NC) หรือคำสั่งหน้าสัมผัสปกติปิด โดยหน้าสัมผัสจะเปลี่ยนสภาวะจากปกติปิดเป็นสภาวะเปิด เมื่อมีการเปลี่ยนสัญญาณอินพุตจาก 0 เป็น 1 ดังรูปที่ 6.1 (ข)



(ก) คำสั่ง Normally Open

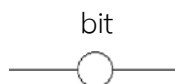


(ข) คำสั่ง Normally Closed

รูปที่ 6.1 Normally Open และคำสั่ง Normally Closed

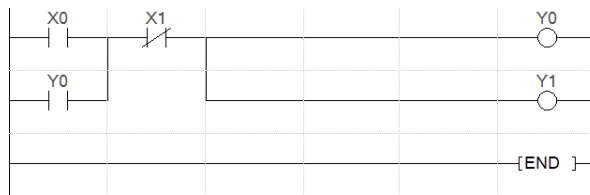
6.3 คำสั่งเอาต์พุต (Output)

คำสั่งเอาต์พุต (Output) เป็นคำสั่งการแสดงผลของเอาต์พุต ดังรูปที่ 6.2



รูปที่ 6.2 คำสั่ง Output

ตัวอย่างที่ 6.1 การใช้คำสั่ง Normally Open (NO) คำสั่ง Normally Closed (NC) และ คำสั่ง Output (Output)



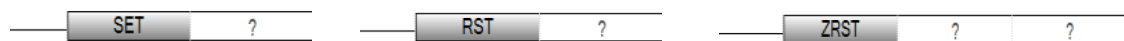
รูปที่ 6.3 โปรแกรมตัวอย่างโดยใช้คำสั่ง Normally Open (NO) คำสั่ง Normally Closed (NC) และ คำสั่ง Output (Output)

การทำงานของโปรแกรม

เป็นการใช้คำสั่งหน้าสัมผัสปกติเปิดและคำสั่งหน้าสัมผัสปกติปิดควบคุมการทำงานของเอาต์พุต โดยมีการทำงานของโปรแกรม คือ เมื่ออินพุต X0 ทำงาน จะทำให้คำสั่งเอาต์พุต Y0 และ Y1 ทำงานและเมื่ออินพุต X1 ทำงาน จะทำให้คำสั่งเอาต์พุต Y0 และ Y1 หยุดทำงาน ดังรูปที่ 6.3

6.4 คำสั่ง Set (SET) คำสั่ง Reset (RST) และคำสั่ง Zone Reset (ZRST)

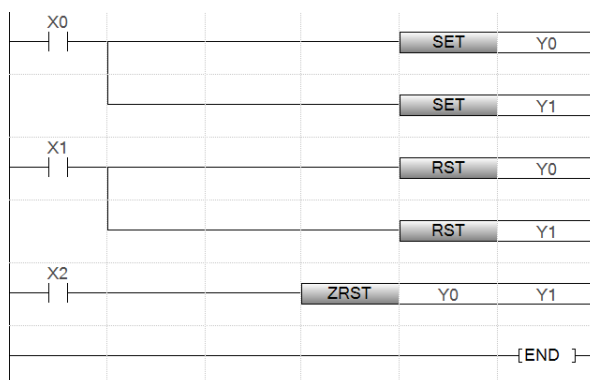
คำสั่ง Set เป็นคำสั่งที่ทำให้ตำแหน่งบิต (bit) ที่ถูกระบุทำงาน (Lock) ดังรูปที่ 6.4 (ก) และคำสั่ง Reset เป็นคำสั่งที่ทำให้ตำแหน่งบิตที่ถูกระบุหยุดทำงาน (Unlock) ดังรูปที่ 6.4 (ข) และคำสั่ง Zone Reset เป็นคำสั่งให้ตำแหน่งบิตที่ถูกระบุแบบกลุ่มหยุดทำงาน (Zone Unlock) ตามรายละเอียดที่ระบุในบิตเริ่มต้นและบิตสิ้นสุด ดังรูปที่ 6.4 (ค)



(ก) คำสั่ง Set (SET) (ข) คำสั่ง Reset (RST) (ค) คำสั่ง Zone Reset (ZRST)

รูปที่ 6.4 คำสั่ง Set (SET) คำสั่ง Reset (RST) และคำสั่ง Zone Reset (ZRST)

ตัวอย่างที่ 6.2 การใช้คำสั่ง Set (SET) คำสั่ง Reset (RST) และคำสั่ง Zone Reset (ZRST)



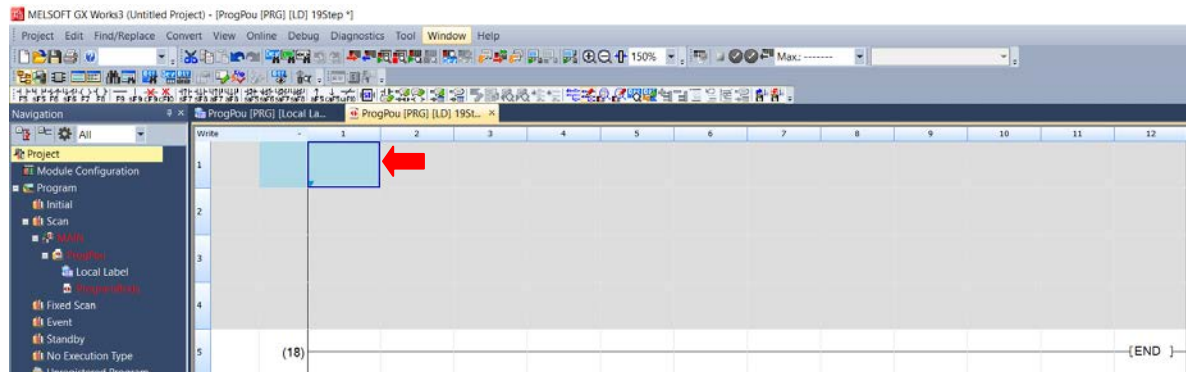
รูปที่ 6.5 ตัวอย่างการใช้คำสั่ง Set (SET) คำสั่ง Reset (RST) และคำสั่ง Zone Reset (ZRST)

การทำงานของโปรแกรม

เป็นการใช้คำสั่ง Set Reset และ Zone Reset เพื่อควบคุมการทำงานของเอาต์พุต โดยมีการทำงานคือ เมื่ออินพุต X0 ทำงาน จะทำให้คำสั่ง Set ทำงาน ส่งผลให้เอาต์พุตตำแหน่ง Y0 และ Y1 ทำงาน เมื่ออินพุต X1 ทำงาน จะทำให้คำสั่ง Reset ทำงาน ส่งผลให้เอาต์พุตตำแหน่ง Y0 และ Y1 หยุดทำงาน หรือเมื่ออินพุต X2 ทำงาน จะทำให้คำสั่ง Zone Reset ทำงาน ส่งผลให้เอาต์พุตตำแหน่ง Y0 และ Y1 หยุดทำงานเช่นเดียวกับเมื่ออินพุต X1 ทำงาน ดังรูปที่ 6.5

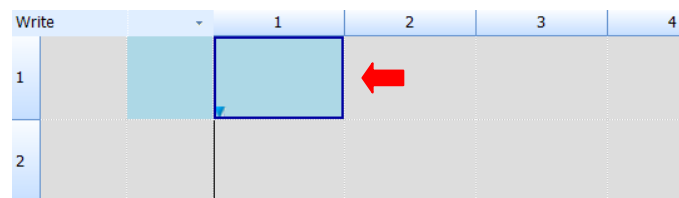
วิธีการเขียนโปรแกรม

1. เปิดโปรแกรม GX Works3 ทำการเชื่อมต่อสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับ PLC จากนั้นเปิดพื้นที่ในการเขียนโปรแกรกดังรูปที่ 6.6



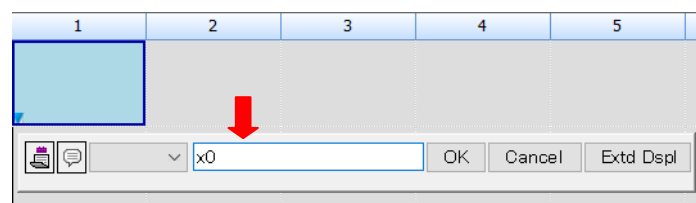
รูปที่ 6.6 เปิดโปรแกรม GX Works3 ทำการเชื่อมต่อสื่อสารข้อมูล

2. คลิกตำแหน่งแรก เพื่อจะเริ่มต้นเขียนโปรแกรม ดังรูปที่ 6.7



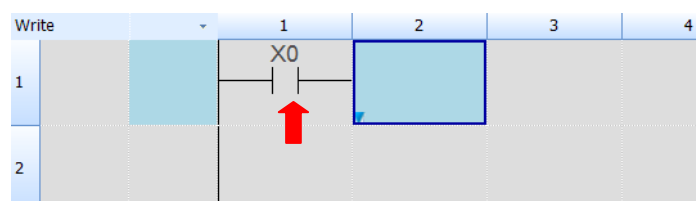
รูปที่ 6.7 คลิกตำแหน่งแรก เพื่อเริ่มต้นเขียนโปรแกรม

3. พิมพ์ X0 จะปรากฏหน้าต่างต่างขึ้นมา ดังรูปที่ 6.8



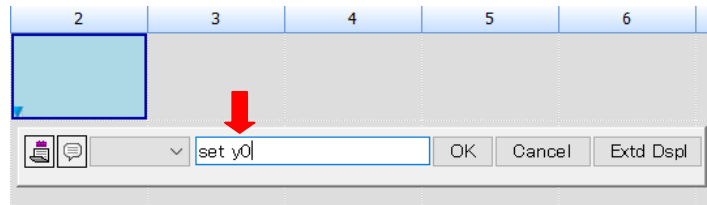
รูปที่ 6.8 พิมพ์ X0

4. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.9



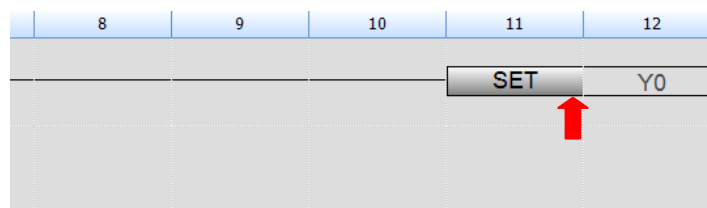
รูปที่ 6.9 กดปุ่ม Enter คำสั่งจะเข้าไปในพื้นที่การเขียนโปรแกรม

5. พิมพ์ SET เคาะเว้น 1 ครั้ง พิมพ์ Y0 จะปรากฏหน้าต่างขึ้นมา ดังรูป 6.10



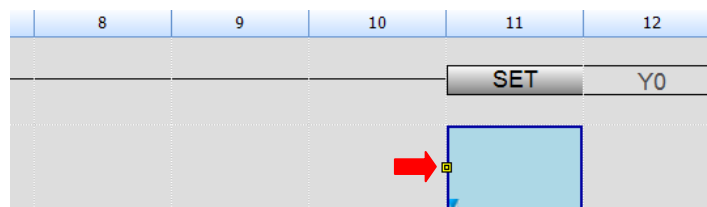
รูปที่ 6.10 พิมพ์ SET เคาะเว้น 1 ครั้ง พิมพ์ Y0

6. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.11



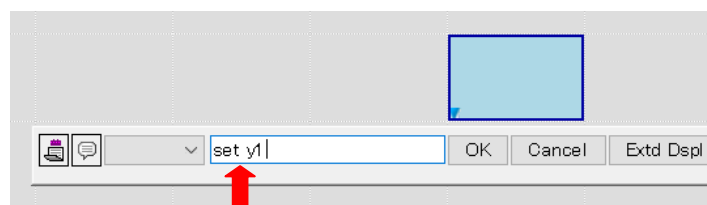
รูปที่ 6.11 กดปุ่ม Enter คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม

7. คลิกตำแหน่งต่อไป เพื่อจะเขียนโปรแกรมต่อ ดังรูปที่ 6.12



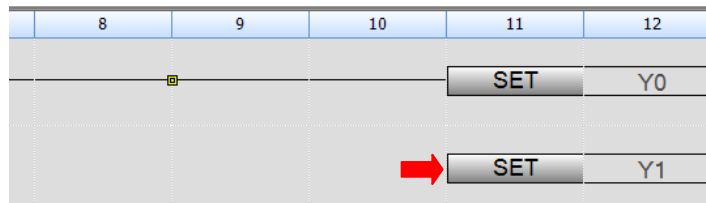
รูปที่ 6.12 คลิกตำแหน่งต่อไป

8. พิมพ์ SET เคาะเว้น 1 ครั้ง พิมพ์ Y1 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 6.13



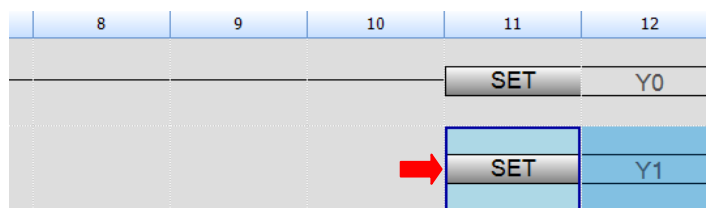
รูปที่ 6.13 พิมพ์ SET เคาะเว้น 1 ครั้ง พิมพ์ Y1

9. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.14



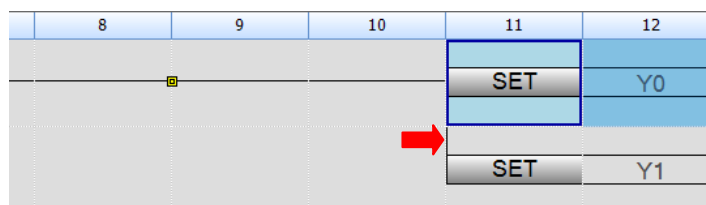
รูปที่ 6.14 กดปุ่ม Enter คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม

10. คลิกที่คำสั่ง SET Y1 เพื่อจะเชื่อมต่อเส้น ดังรูปที่ 6.15



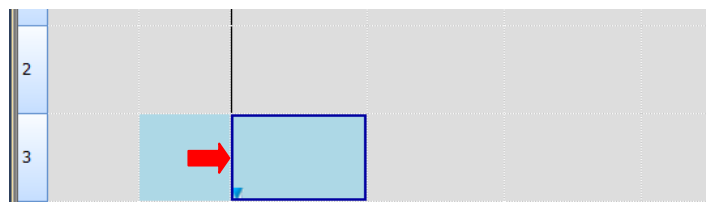
รูปที่ 6.15 คลิกที่คำสั่ง SET Y1 เพื่อเชื่อมต่อเส้น

11. กดปุ่ม Ctrl ค้างไว้และตามด้วยหัวลูกศรชี้ขึ้น (↑) จะได้การเชื่อมต่อเส้นตามโปรแกรม ดังรูปที่ 6.16



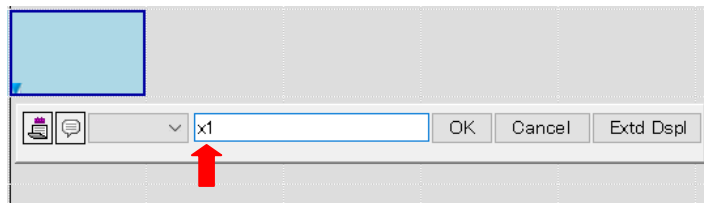
รูปที่ 6.16 กดปุ่ม Ctrl ค้างไว้และตามด้วยหัวลูกศรชี้ขึ้น (↑)

12. คลิกที่ตำแหน่งต่อไป ดังรูปที่ 6.17



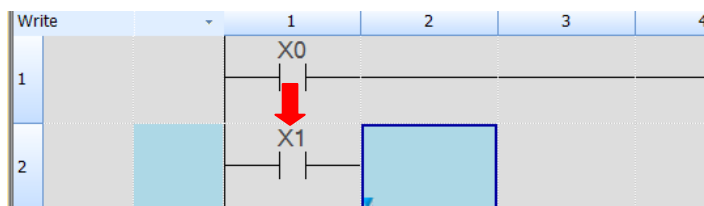
รูปที่ 6.17 คลิกที่ตำแหน่งต่อไป

13. พิมพ์ X1 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 6.18



รูปที่ 6.18 พิมพ์ X1

14. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.19



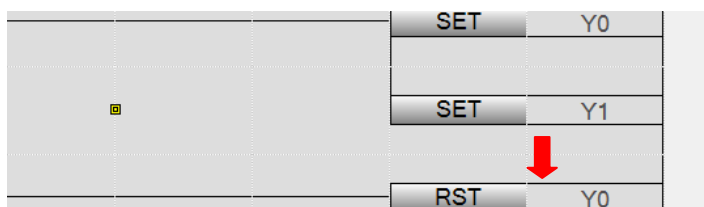
รูปที่ 6.19 กดปุ่ม Enter คำสั่งจะเข้าไปในพื้นที่การเขียนโปรแกรม

15. พิมพ์ RST เคาะเว้น 1 ครั้ง พิมพ์ Y0 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 6.20



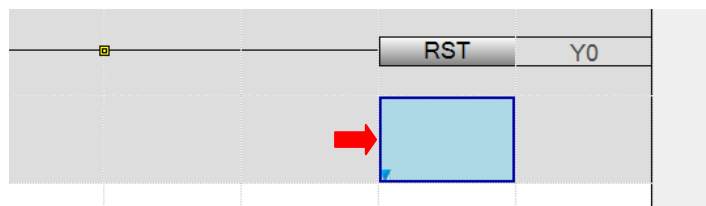
รูปที่ 6.20 พิมพ์ RST เคาะเว้น 1 ครั้ง พิมพ์ Y0

16. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.21



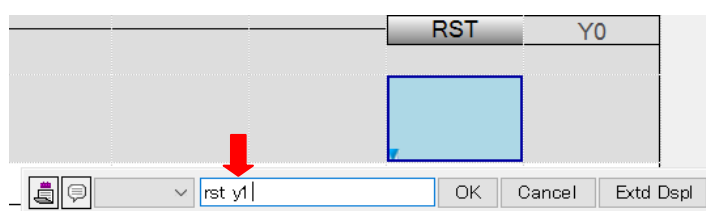
รูปที่ 6.21 กดปุ่ม Enter คำสั่งจะเข้าไปในพื้นที่การเขียนโปรแกรม

17. คลิกตำแหน่งต่อไป เพื่อจะเขียนโปรแกรมต่อ ดังรูปที่ 6.22



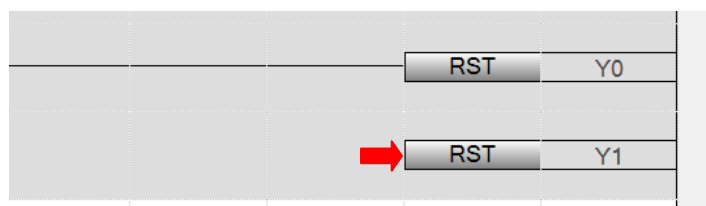
รูปที่ 6.22 คลิกตำแหน่งต่อไป

18. พิมพ์ RST เคาะเว้น 1 ครั้ง พิมพ์ Y1 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 6.23



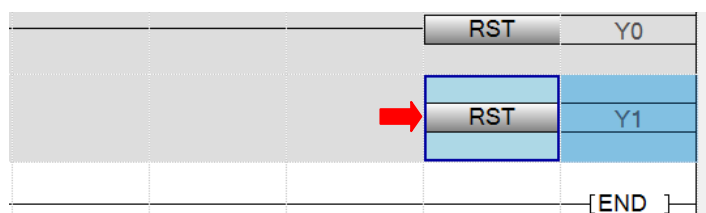
รูปที่ 6.23 พิมพ์ RST เคาะเว้น 1 ครั้ง พิมพ์ Y1

19. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.24



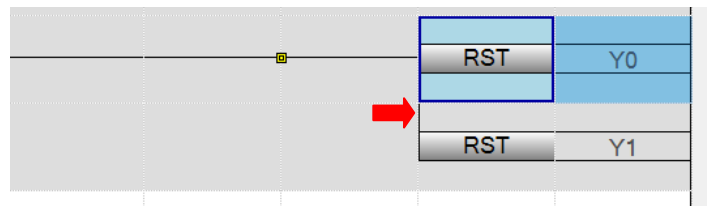
รูปที่ 6.24 กดปุ่ม Enter คำสั่งจะเข้าไปในพื้นที่การเขียนโปรแกรม

20. คลิกที่คำสั่ง RST Y1 เพื่อจะเชื่อมต่อเส้น ดังรูปที่ 6.25



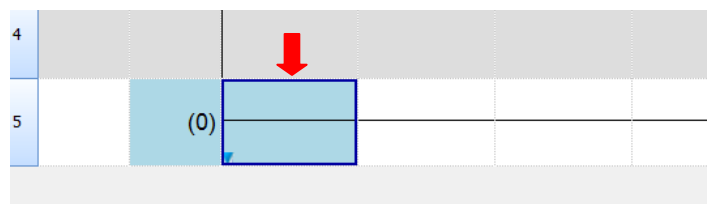
รูปที่ 6.25 คลิกที่คำสั่ง RST Y1

21. กดปุ่ม Ctrl ค้างไว้และตามด้วยหัวลูกศรชี้ขึ้น (↑) จะได้รับการเชื่อมต่อเส้นตามโปรแกรม ดังรูปที่ 6.26



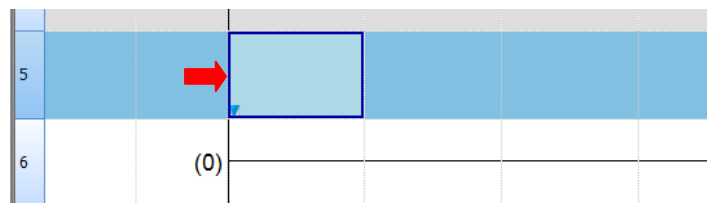
รูปที่ 6.26 กดปุ่ม Ctrl ค้างไว้และตามด้วยหัวลูกศรชี้ขึ้น (↑)

22. คลิกตำแหน่งต่อไป เพื่อจะเลื่อนบรรทัดของโปรแกรมลง สำหรับขยายพื้นที่ในการเขียนโปรแกรมต่อ ดังรูปที่ 6.27



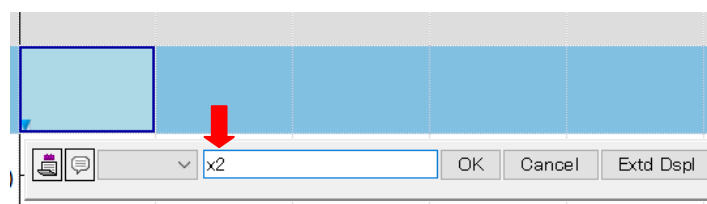
รูปที่ 6.27 คลิกตำแหน่งต่อไป เพื่อเลื่อนบรรทัดของโปรแกรมลง

23. กดปุ่ม Ctrl ค้างไว้และตามด้วยปุ่ม Insert 1 ครั้ง บรรทัดของโปรแกรมจะขยับลง เพื่อขยายพื้นที่ในการเขียนโปรแกรม ดังรูปที่ 6.28



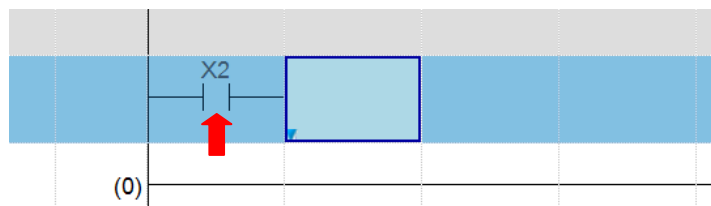
รูปที่ 6.28 กดปุ่ม Ctrl ค้างไว้และตามด้วยปุ่ม Insert 1 ครั้ง

24. พิมพ์ X2 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 6.29



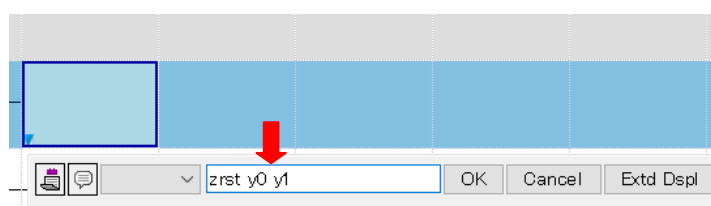
รูปที่ 6.29 พิมพ์ X2

25. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.30



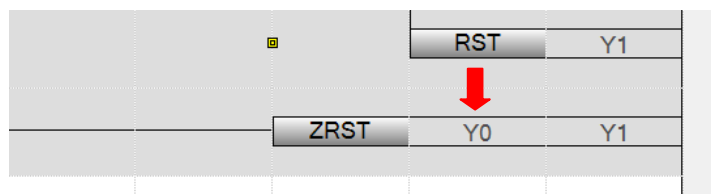
รูปที่ 6.30 กดปุ่ม Enter คำสั่งจะเข้าไปในพื้นที่การเขียนโปรแกรม

26. พิมพ์ ZRST เคาะเว้น 1 ครั้งพิมพ์ Y0 เคาะเว้น 1 ครั้งพิมพ์ Y1 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 6.31



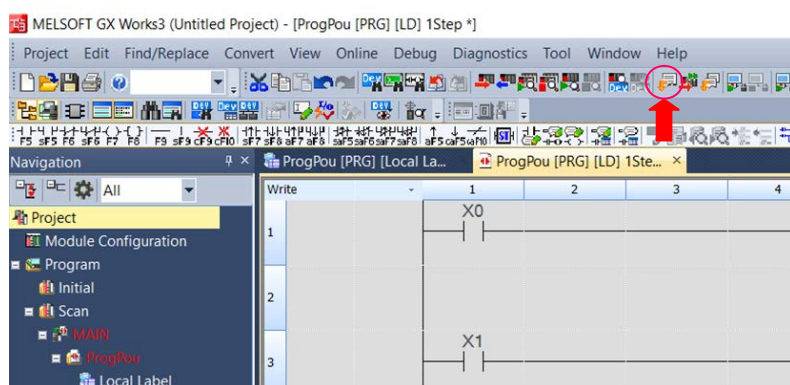
รูปที่ 6.31 พิมพ์ ZRST เคาะเว้น 1 ครั้งพิมพ์ Y0 เคาะเว้น 1 ครั้งพิมพ์ Y1

27. กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.32



รูปที่ 6.32 กดปุ่ม Enter คำสั่งจะเข้าไปในพื้นที่การเขียนโปรแกรม

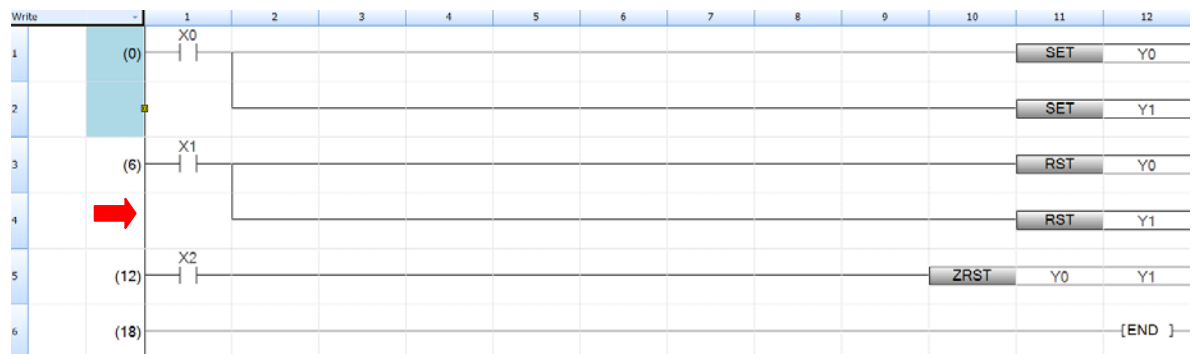
28. เมื่อเขียนโปรแกรมเสร็จเรียบร้อยแล้วตามตัวอย่างที่กำหนด ขั้นตอนต่อไปคือการตรวจสอบโปรแกรม (Convert หรือ F4) คลิกที่ปุ่ม Convert หรือ กด F4 ดังรูปที่ 6.33



= Convert

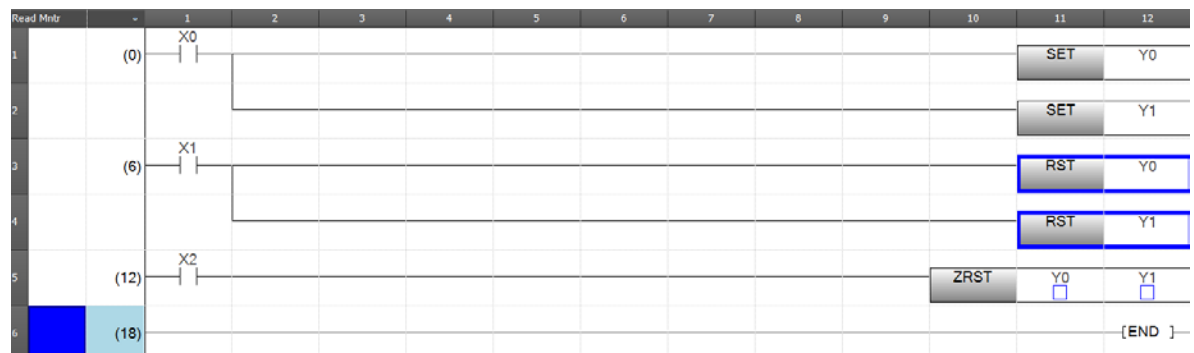
รูปที่ 6.33 คลิกที่ปุ่ม Convert

29. รวบรวมขั้นตอนการตรวจสอบ จนกระทั่งครบ 100 % พื้นที่เขียนโปรแกรม จะเปลี่ยนจากสีเทา เป็น สีขาว ดังรูปที่ 6.34



รูปที่ 6.34 รวบรวมขั้นตอนการตรวจสอบ

30. จากนั้นให้โหลดโปรแกรมลงใน PLC (Write to PLC) และทดลองการทำงานของโปรแกรม ตามลำดับ ดังรูปที่ 6.35



รูปที่ 6.35 โหลดโปรแกรมลงใน PLC

6.5 คำสั่ง Turning On a specified Device for One Scan at the Rising Edge of an Input Condition (PLS) และ คำสั่ง Turning On a specified Device for One Scan at the Falling Edge of an Input Condition (PLF)

คำสั่ง Turning On a specified Device for One Scan at the Rising Edge of an Input Condition (PLS) เป็นคำสั่งที่มีผลต่อตำแหน่งที่ต่อร่วมกับคำสั่ง PLS ด้วยระยะเวลาเพียง 1 Scantime เมื่อได้รับสัญญาณสถานะ On หรือช่วงขอบขาขึ้นจากคอนแทคที่ควบคุมคำสั่ง PLS ดังรูปที่ 6.36 (ก)

คำสั่ง คำสั่ง Turning On a specified Device for One Scan at the Falling Edge of an Input Condition (PLF) เป็นคำสั่งที่มีผลต่อตำแหน่งที่ต่อร่วมกับคำสั่ง PLF ด้วยระยะเวลาเพียง 1 Scantime เมื่อได้รับสัญญาณสถานะ Off หรือช่วงขอบขาลงจากคอนแทคที่ควบคุมคำสั่ง PLF ดังรูปที่ 6.36 (ข)



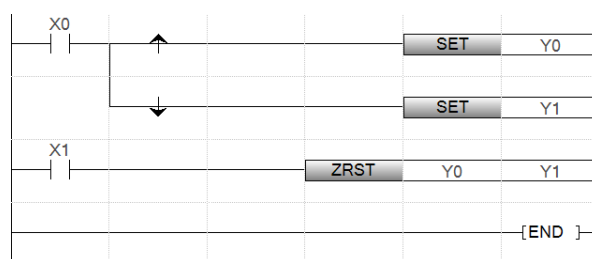
(ก) คำสั่ง PLS



(ข) คำสั่ง PLF

รูปที่ 6.36 คำสั่ง PLS และคำสั่ง PLF

ตัวอย่างที่ 6.3 การใช้คำสั่ง PLS และคำสั่ง PLF



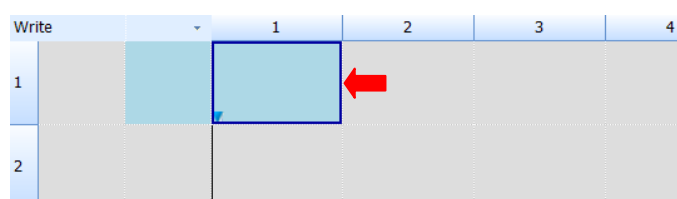
รูปที่ 6.37 โปรแกรมตัวอย่างโดยใช้คำสั่ง PLS และ PLF

การทำงานของโปรแกรม

เมื่ออินพุต X0 ทำงานหรือมีสถานะ On (ช่วงขอบขาขึ้น) จะทำให้คำสั่ง PLS ทำงาน ส่งผลให้อาต์พุต Y0 ทำงาน เมื่ออินพุต X0 หยุดทำงานหรือมีสถานะ Off (ช่วงขอบขาลง) จะทำให้เอาต์พุต Y1 ทำงาน และเมื่ออินพุต X1 ทำงาน จะทำให้คำสั่ง Zone Reset ทำงาน ส่งผลให้อาต์พุตตำแหน่ง Y0 และ Y1 หยุดทำงาน ดังรูปที่ 6.37

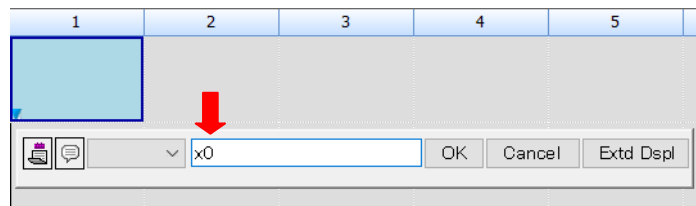
วิธีออกแบบโปรแกรม

- 1) เปิดโปรแกรม GX Works3 เพื่อติดต่อสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับ PLC
- 2) เริ่มต้นเขียนโปรแกรม โดยคลิกตำแหน่งแรก ดังรูปที่ 6.38



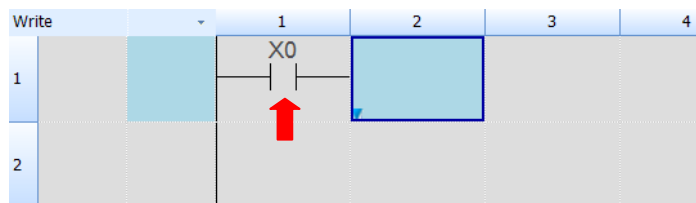
รูปที่ 6.38 คลิกตำแหน่งแรก

3) พิมพ์ X0 จะปรากฏหน้าต่างขึ้นมา ดังรูปที่ 6.39



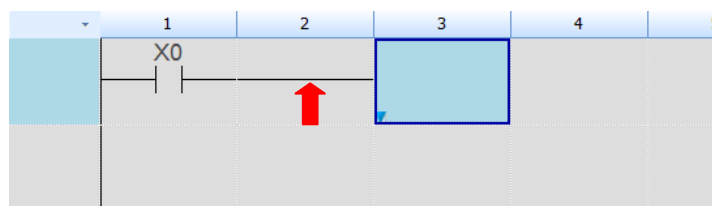
รูปที่ 6.39 พิมพ์ X0

4) กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม ดังรูปที่ 6.40



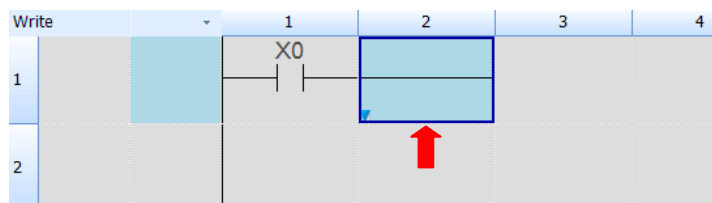
รูปที่ 6.40 กดปุ่ม Enter คำสั่งจะเข้าไปในพื้นที่การเขียนโปรแกรม

5) เชื่อมต่อเส้นไปทางขวา โดยการกดปุ่ม Ctrl ค้างไว้และตามด้วยหัวลูกศรชี้ขึ้น (→) จะได้รับการเชื่อมต่อเส้นตามโปรแกรม ดังรูปที่ 6.41



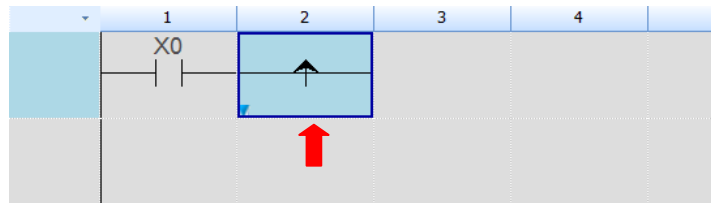
รูปที่ 6.41 เชื่อมต่อเส้นไปทางขวา

6) คลิกตำแหน่งเส้น หรือใช้เครื่องหมายลูกศรชี้ทางซ้าย (←) เพื่อเลื่อนคอร์เซอร์กลับมาที่ตำแหน่ง ดังรูปที่ 6.42



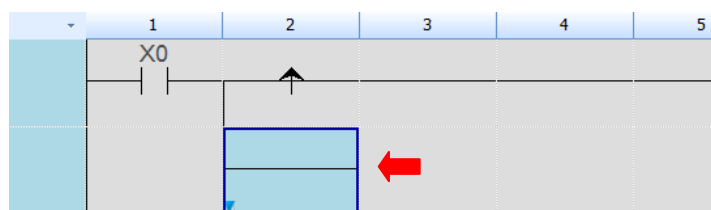
รูปที่ 6.42 เลื่อนคอร์เซอร์กลับมาที่ตำแหน่ง

7) กดปุ่ม Alt ค้างไว้และตามด้วยปุ่ม / จะได้คำสั่ง PLS ตามโปรแกรม ดังรูปที่ 6.43



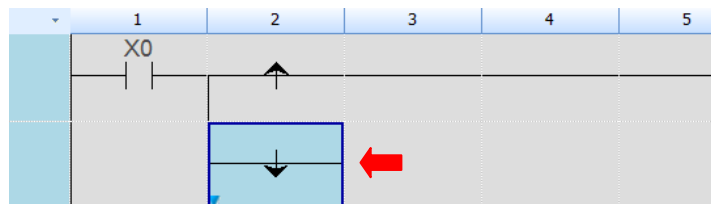
รูปที่ 6.43 กดปุ่ม Alt ค้างไว้และตามด้วยปุ่ม /

8) เขียนโปรแกรมต่อไปตามโปรแกรมตัวอย่าง จนถึงตำแหน่งที่จะเขียนคำสั่ง PLF ดังรูปที่ 6.44



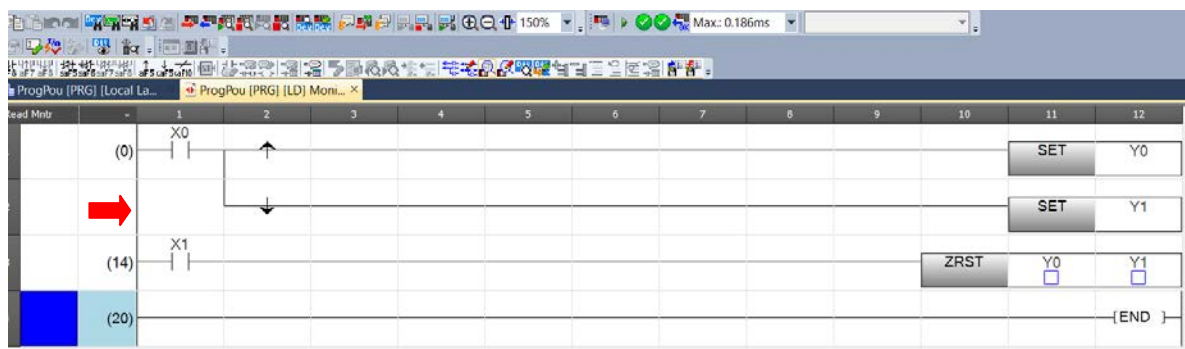
รูปที่ 6.44 เขียนโปรแกรมต่อ จนถึงคำสั่ง PLF

9) กดปุ่ม Alt ค้างไว้และตามด้วยกดปุ่ม / 2 ครั้ง จะได้คำสั่ง PLF ตามโปรแกรม ดังรูปที่ 6.45



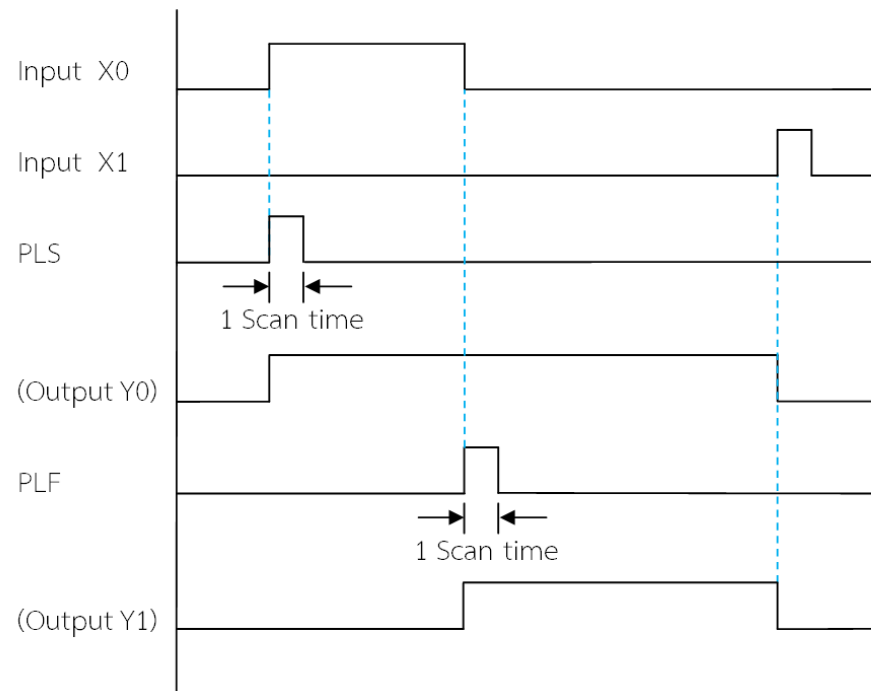
รูปที่ 6.45 กดปุ่ม Alt ค้างไว้และตามด้วยกดปุ่ม / 2 ครั้ง

10) เขียนโปรแกรมจนครบตามโปรแกรมตัวอย่างที่ 6.3 จากนั้นโหลดโปรแกรมลงใน PLC (Write to PLC) และทดลองการทำงานของโปรแกรมตามขั้นตอนดังรูปที่ 6.46



รูปที่ 6.46 เขียนโปรแกรมจนครบตามโปรแกรมตัวอย่างและทดลองการทำงาน

ตามที่กล่าวมาแล้ว คำสั่ง PLS และ PLF มีผลการทำงานเพียง Scantime เดียว โดยคำสั่ง PLS จะมีผลช่วงขอบขาขึ้นและ PLF จะมีผลช่วงขอบขาลงของอินพุตที่ ควบคุมคำสั่ง PLS และ PLF ดังรูปที่ 6.47

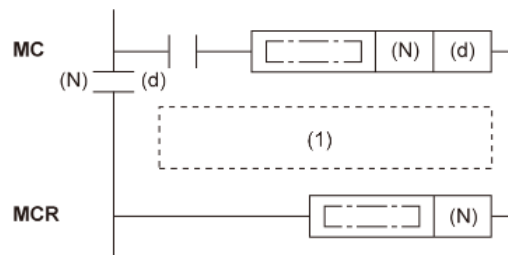


รูปที่ 6.47 Timing Diagram ของ PLS และ PLF

6.6 คำสั่ง Master Control Set (MC) และ Master Control Reset (MCR)

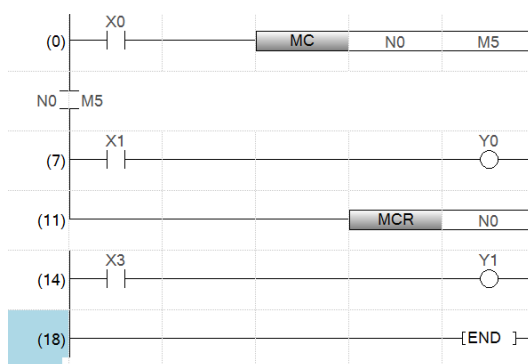
เป็นคำสั่ง ควบคุมการทำงานของโปรแกรมซึ่งอยู่ระหว่างคำสั่ง MC และ MCR ให้ทำงานหรือไม่ทำงาน โดยการใช้งานจะต้องขึ้นต้นด้วยคำสั่ง MC และลงท้ายด้วยคำสั่ง MCR เสมอ คำสั่ง MC จะมีค่าที่ต้องกำหนด 2 ค่าคือ ค่า N และค่า d โดยค่า N (Nesting) หมายถึงจำนวนรังหรือจำนวนชุดที่สามารถใช้งานด้วยคำสั่ง MC และ MCR ได้ และค่า d คือค่าของบิตหรือตำแหน่งใดๆ ที่ต้องการให้ทำงานเมื่อคำสั่ง MC ทำงาน และคำสั่ง MCR มีค่าที่ต้องกำหนด 1 ค่า คือ ค่า N (Nesting) โดยการกำหนดค่า N หากต้องการให้คำสั่ง MC และ MCR เป็นรังหรือชุดเดียวกัน จะต้องระบุค่า N หมายเลขเดียวกัน โดยค่า N สามารถกำหนดค่าได้ตั้งแต่ N0 – N14 และค่า d สามารถนำบิตหรือตำแหน่งของรีเลย์ช่วยมาใช้งาน

การทำงานของคำสั่งคือ เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง MC จะทำให้โปรแกรมซึ่งอยู่ระหว่าง MC และ MCR ทำงานเป็นปกติ แต่หากไม่มีสัญญาณจากอินพุตเข้า ที่คำสั่ง MC จะทำให้โปรแกรมซึ่งอยู่ระหว่าง MC และ MCR จะไม่สามารถทำงานได้ จะค้างสภาวะการทำงานเดิมก่อนหน้า



รูปที่ 6.48 สัญลักษณ์คำสั่ง MC และ MCR

ตัวอย่างที่ 6.4 การใช้คำสั่ง MC และ MCR



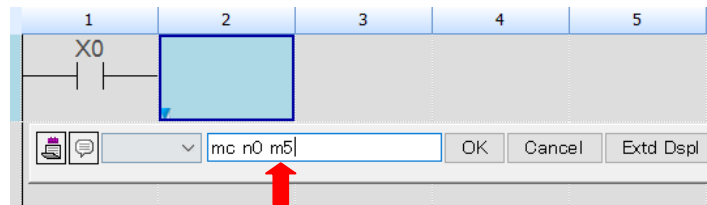
รูปที่ 6.49 ตัวอย่างการใช้คำสั่ง MC และ MCR

การทำงานของโปรแกรม

เมื่อคำสั่ง MC และ MCR ยังไม่ทำงาน (X0 มีสถานะ Off) โปรแกรม ที่อยู่ระหว่างคำสั่ง MC และ MCR จะไม่ทำงาน (X1 มีสถานะ On แต่ Output Y0 ไม่ทำงาน) แต่เมื่อคำสั่ง MC และ MCR ทำงาน (X0 มีสถานะ On และคงสภาวะ On ไว้) จะทำให้โปรแกรมซึ่งอยู่ระหว่างคำสั่ง MC และ MCR ทำงานเป็นปกติ X1 มีสถานะ On Output Y0 จะทำงานเป็นปกติ) สังเกต โปรแกรม MC และ MCR จะถูกควบคุมแยกขาดออกจากโปรแกรมปกติ การทำงานของ X3 สามารถทำงานได้เป็นปกติ เนื่องจากอยู่นอกเหนือการควบคุมของคำสั่ง MC และ MCR ดังรูปที่ 6.49

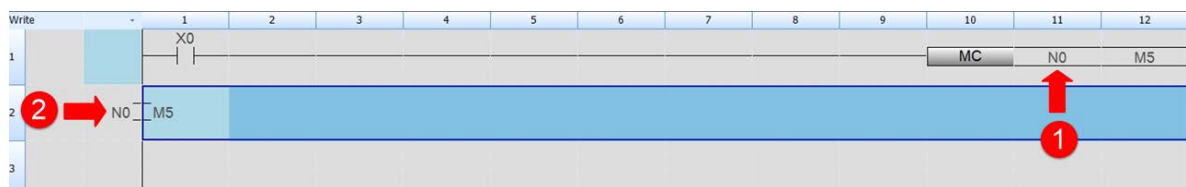
วิธีออกแบบโปรแกรม

- 1) เปิดโปรแกรม GX Works3 เพื่อติดต่อสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับ PLC
- 2) เขียนโปรแกรมตามโปรแกรมตัวอย่างที่ 6.4 จนถึงคำสั่ง MC พิมพ์ MC เคาะเว้น 1 ครั้งพิมพ์ NO เคาะเว้น 1 ครั้งพิมพ์ M5 จะปรากฏหน้าต่างขึ้นมามีดังรูปที่ 6.50



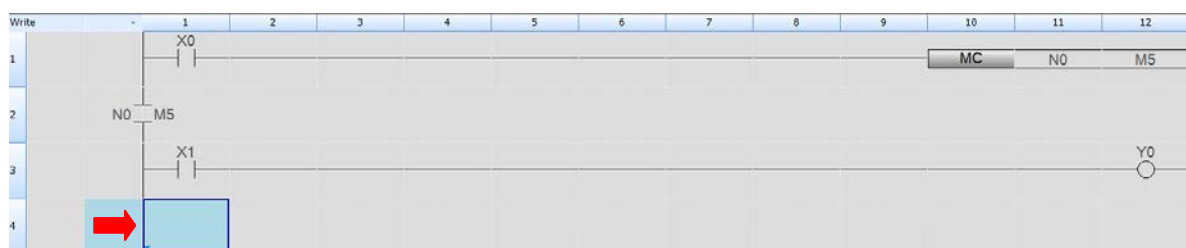
รูปที่ 6.50 พิมพ์ MC เคาะเว้น 1 ครั้งพิมพ์ NO เคาะเว้น 1 ครั้งพิมพ์ M5

- 3) กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม (1) และโปรแกรมจะสร้างหน้าคอนแทคปกติเปิด NO M5 ขึ้นมา (2) ดังรูปที่ 6.51



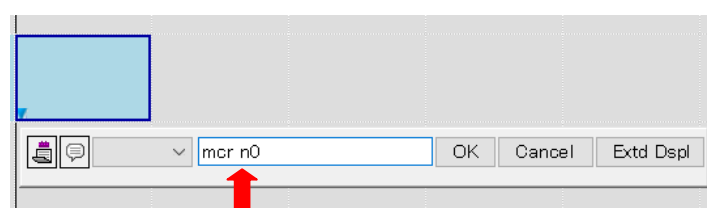
รูปที่ 6.51 กดปุ่ม Enter

- 4) คลิกตำแหน่งต่อไปและเขียนโปรแกรมจนถึงคำสั่ง MCR ดังรูปที่ 6.52



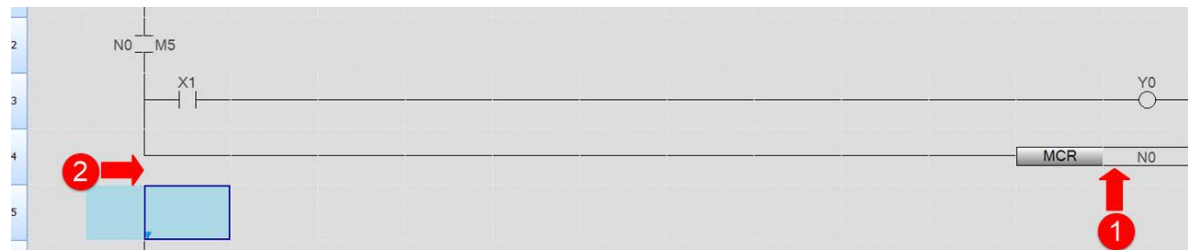
รูปที่ 6.52 คลิกตำแหน่งต่อไปและเขียนโปรแกรมจนถึงคำสั่ง MCR

- 5) พิมพ์ MCR เคาะเว้น 1 ครั้งพิมพ์ NO จะปรากฏหน้าต่างขึ้นมามีดังรูปที่ 6.53



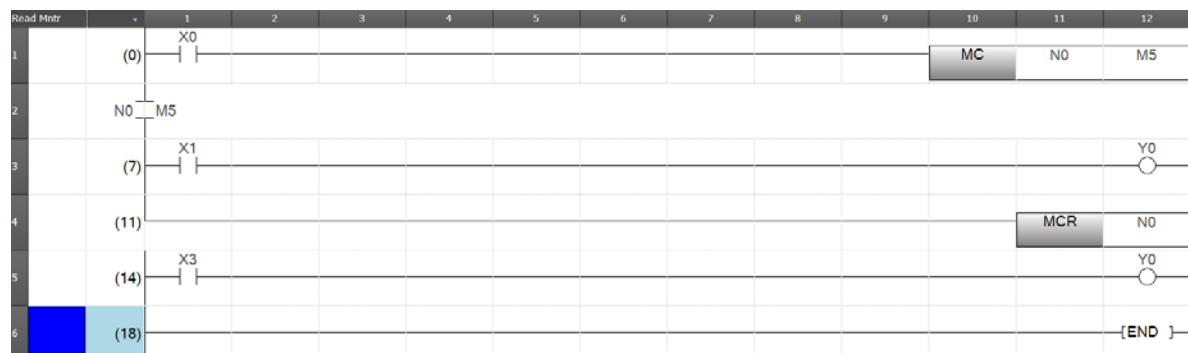
รูปที่ 6.53 พิมพ์ MCR เคาะเว้น 1 ครั้งพิมพ์ NO

6) กดปุ่ม Enter จะได้คำสั่งเข้าไปในพื้นที่การเขียนโปรแกรม (1) และสังเกตโปรแกรม MC และ MCR จะถูกควบคุมแยกขาดออกจากโปรแกรมปกติ (2) ดังรูปที่ 6.54



รูปที่ 6.54 กดปุ่ม Enter คำสั่งจะเข้าไปในพื้นที่การเขียนโปรแกรม


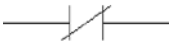

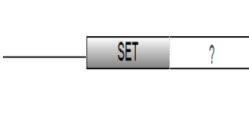
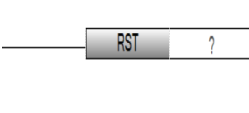
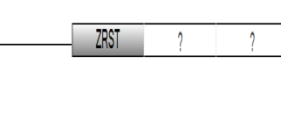

7) เขียนโปรแกรมจนครบตามโปรแกรมตัวอย่างที่ 6.4 จากนั้นโหลดโปรแกรมลงใน PLC (Write to PLC) และทดลองการทำงานของโปรแกรมตามขั้นตอนดังรูปที่ 6.55


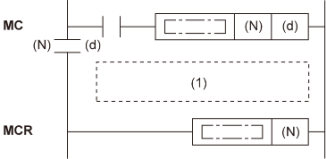


รูปที่ 6.55 เขียนโปรแกรมจนครบตามโปรแกรมตัวอย่างและทดลองการทำงาน

สรุป

การเขียนโปรแกรม PLC จะมีกลุ่มคำสั่งพื้นฐานให้เลือกใช้สำหรับการเขียนโปรแกรมเบื้องต้น โดยกลุ่มคำสั่งพื้นฐานจะมีชื่อคำสั่งและสัญลักษณ์รวมถึงความหมายของคำสั่ง สามารถสรุปได้ดังนี้

ชื่อคำสั่งและสัญลักษณ์	ความหมาย : การทำงาน
 Normally Open (NO)	ความหมาย เป็นคำสั่งหน้าสัมผัสสปกติเปิด การทำงาน หน้าสัมผัสจะเปลี่ยนสถานะจากปกติเปิดเป็นสถานะปิด เมื่อมีการเปลี่ยนสัญญาณอินพุตจาก 0 เป็น 1 หรือจาก Off เป็น On
 Normally Close (NC)	ความหมาย เป็นคำสั่งหน้าสัมผัสสปกติปิด การทำงาน หน้าสัมผัสจะเปลี่ยนสถานะจากปกติปิดเป็นสถานะเปิด เมื่อมีการเปลี่ยนสัญญาณอินพุตจาก 0 เป็น 1 หรือจาก Off เป็น On
 Output	ความหมาย เป็นคำสั่งการแสดงผลของ Output การทำงาน เอาต์พุตจะทำงานเมื่อมีสัญญาณอินพุตเข้าที่เอาต์พุตคอยล์
 Set (SET)	ความหมาย เป็นคำสั่งที่ใช้กระทำการ Set ค่าที่ต้องการบิต (bit) ให้มีสถานะ On การทำงาน เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง Set จะทำให้คำสั่ง Set ทำงานไปกระทำการ Set bit ที่ตั้งค่าไว้ให้มีสถานะ On
 Reset (RST)	ความหมาย เป็นคำสั่งที่ใช้กระทำการ Reset ค่าที่ต้องการบิต (bit) ให้มีสถานะ Off การทำงาน เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง Reset จะทำให้คำสั่ง Reset ทำงานไป Reset bit ที่ตั้งค่าไว้ให้มีสถานะ Off
 Zone Reset (ZRST)	ความหมาย เป็นคำสั่งที่ใช้กระทำการ Reset แบบกลุ่ม กับค่าที่ต้องการบิต (bit) ให้มีสถานะ Off การทำงาน เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง Zone Reset จะทำให้คำสั่ง Zone Reset ทำงานไป Reset กลุ่ม bit ที่ตั้งค่าไว้ให้มีสถานะ Off
 PLS	ความหมาย เป็นคำสั่งที่หน้าสัมผัสทำงาน (On) ด้วยระยะเวลา 1 Scantime เมื่อเปลี่ยนสัญญาณจาก Off เป็น On (ขอบขาขึ้น) การทำงาน หน้าสัมผัสทำงาน (On) เป็นระยะเวลา 1 Scantime เมื่อเปลี่ยนสัญญาณอินพุตจาก 0 เป็น 1 หรือจาก Off เป็น On (หรือเรียกว่า ช่วงขอบขาขึ้น)

ชื่อคำสั่งและสัญลักษณ์	ความหมาย : การทำงาน
 <p style="text-align: center;">PLF</p>	<p>ความหมาย เป็นคำสั่งที่หน้าสัมผัสทำงาน (On) ด้วยระยะเวลา 1 Scantime เมื่อเปลี่ยนสัญญาณจาก On เป็น Off (ขอบขาลง)</p> <p>การทำงาน หน้าสัมผัสจะทำงาน On เป็นระยะเวลา 1 Scantime เมื่อเปลี่ยนสัญญาณอินพุตจาก 1 เป็น 0 หรือจาก On เป็น Off (หรือเรียกว่า ช่วงขอบขาลง)</p>
	<p>ความหมาย เป็นคำสั่งควบคุมการทำงานของโปรแกรมซึ่งอยู่ระหว่างคำสั่ง MC และ MCR โดยขึ้นต้นด้วยคำสั่ง MC และลงท้ายด้วย MCR</p> <p>การทำงาน เมื่อมีสัญญาณจากอินพุตเข้าที่คำสั่ง MC จะทำให้โปรแกรมซึ่งอยู่ระหว่าง MC และ MCR ทำงานเป็นปกติ แต่หากไม่มีสัญญาณเข้าที่คำสั่ง MC จะทำให้โปรแกรมซึ่งอยู่ระหว่าง MC และ MCR ไม่สามารถทำงานได้ จะค้างสภาวะการทำงานเดิมก่อนหน้า</p>