

PROGRAMMABLE CONTROLLERS  
MELSEC-F

เรียนรู้ผ่านรีเลย์ซีเควนซ์...

# Your First PLC

สำหรับผู้เริ่มต้นใช้งาน





## ● ข้อควรระมัดระวังเกี่ยวกับเรื่องความปลอดภัย ● (กรุณาอ่านก่อนเริ่มการอบรม)

ควรอ่านคู่มือเล่มนี้ขณะออกแบบระบบ พร้อมกับต้องคำนึงถึงเรื่องความปลอดภัยอย่างเคร่งครัด  
อีกทั้งก่อนที่จะมีการฝึกอบรมควรระมัดระวังและปฏิบัติตามขั้นตอนเหล่านี้ให้ได้อย่างถูกต้อง

### (ข้อควรระมัดระวังก่อนเริ่มทำการอบรม)

#### ⚠ อันตราย

- เพื่อป้องกันอันตรายจากไฟฟ้าดูด จึงไม่ควรสัมผัสขั้วไฟ ขณะที่กระแสไฟฟ้าไหลอยู่
- เมื่อมีการเปิดฝาเครื่อง ควรที่จะปิดสวิตช์ หรือควรตรวจสอบเรื่องของความปลอดภัยให้ถี่ถ้วนก่อนเริ่มปฏิบัติงาน
- ไม่ควรใช้มือยื่นเข้าไปบริเวณจุดที่มีอุปกรณ์กำลังเคลื่อนที่

#### ⚠ ข้อควรระมัดระวัง

- กรุณาอบรมตามคำแนะนำของผู้สอน
- สำหรับปลั๊กไฟแบบมีสายดิน กรุณาใช้กับปลั๊กสามขา  
กรณีที่มีการใช้หัวต่อจาก 3 หัวไปเป็น 2 หัว ควรเสียบสายดินของปลั๊ก เข้ากับขั้วต่อของหัวต่อด้วย
- ไม่ควรกระทำการถอดยูนิต หรือเปลี่ยนสายไฟโดยพลการ อาจเป็นสาเหตุทำให้เกิดความเสียหาย  
การทำงานที่ผิดพลาด อุบัติเหตุ หรือเพลิงไหม้ได้
- เมื่อมีการประกอบหรือถอดแยกยูนิตจำเป็นต้องปิดเครื่องก่อนทุกครั้ง  
ขณะที่กระแสไฟไหลผ่านอาจจะเป็นสาเหตุที่ทำให้เกิดการเสียหายของยูนิต หรืออาจเกิดไฟดูดได้
- เมื่อเกิดกลิ่น หรือเสียงที่ผิดปกติขึ้นที่เครื่อง (อาทิเช่น X/Y Table) ให้ทำการปิดสวิตช์ทันที
- เมื่อพบว่ามีส่วนผิดปกติเกิดขึ้น ให้เรียกผู้ฝึกสอนทันที

ระวัง :

ชุดฝึกการต่อสายไฟ, PLC Training Kit ใช้กับไฟ AC100V

กรุณาอย่าใช้กับไฟเกิน AC100V เด็ดขาด เพราะจะเป็นสาเหตุทำให้อุปกรณ์เสียหาย  
และเป็นต้นเหตุของไฟไหม้ได้ เวลาใช้งานกรุณาเลือกใช้เครื่องแปลงไฟที่เหมาะสม

## คำแนะนำเบื้องต้น

คู่มือฉบับนี้ มีไว้สำหรับแนะนำผู้ที่กำลังจะใช้เครื่อง PLC เป็นครั้งแรก ซึ่งมีทั้งเนื้อหาความรู้ขั้นพื้นฐาน สำหรับการควบคุมเครื่อง PLC รวมถึงการแนะนำกรณีตัวอย่างง่าย ๆ

อีกทั้งเนื้อหาในคู่มือฉบับนี้ มีการใช้เครื่อง Micro PLC รุ่น FX3G-14MR/ES เป็นสื่อในการสอน และรุ่น FX-I/O-DEMO2 สำหรับใช้เป็นเครื่องในการฝึกสอน

เอกสารที่เกี่ยวข้องจะมีการระบุไว้ตามด้านล่าง

ชื่อเรียกเอกสาร	หมายเลขเอกสาร	เนื้อหา
■ FX3G Series main unit		
FX3G Series Hardware Manual	JY997D33401	สเปกของ Input - Output สำหรับรุ่น FX3G รวมถึงสายไฟและการติดตั้งคู่มือสำหรับ User รุ่น FX3G ที่มีความสมบูรณ์มากกว่า ฉบับ Hardware
FX3G Series User's Manual [Hardware Edition]	JY997D31301	สเปกของ Input - Output สำหรับตัวเครื่อง FX3G PLC รวมถึงสายไฟ การติดตั้ง และการซ่อมบำรุงที่เกี่ยวข้องกับ Hardware โดยมีเนื้อหาที่ละเอียดแยกตามหัวข้อ
■ Programming		
FX3s/FX3G/FX3GC/FX3U/FX3UC Series Programming Manual [Basic & Applied Instruction Edition]	JY997D16601	มีการอธิบายคำสั่งขั้นพื้นฐานของ PLC รุ่น FX3s-FX3G-FX3GC-FX3U-FX3UC • รวมถึงคำสั่งการใช้งาน คำอธิบายสำหรับ Device แต่ละประเภท และหัวข้อต่างๆ ที่เกี่ยวกับ Programming ของ ซีเควนซ์
■ Handy Programming Panel		
FX-30P Installation Manual	JY997D34201	เนื้อหาบางส่วนนำมาจากคู่มือการทำงานเครื่อง FX-30P โดยเกี่ยวกับสเปกและการติดตั้ง FX-30P
FX-30P Operation Manual	JY997D34401	อธิบายหัวข้อรายละเอียดเกี่ยวกับ Handy Programming Panel รุ่น FX-30P

### เครื่องหมายการค้า

- Windows, Windows 7, Windows 8 เป็นเครื่องหมายการค้า หรือเครื่องหมายการค้าของ Microsoft Corporation ของประเทศสหรัฐอเมริกา และในประเทศอื่นๆ
- ชื่อบริษัท และชื่อสินค้าของบริษัทอื่นในเอกสารชุดนี้ล้วนเป็นเครื่องหมายการค้า และเป็นเครื่องหมายการค้าของบริษัทนั้นๆ

หนังสือเล่มนี้ มิได้จัดทำขึ้นเพื่อการรับประกันการจดสิทธิในทรัพย์สินอุตสาหกรรมหรืออำนาจเชิงสิทธิ อีกทั้งการอนุญาตในอำนาจการนำไปปฏิบัติต่างๆ แต่อย่างใด

อีกทั้งปัญหาต่างๆ ที่อาจเกิดขึ้นโดยข้อบัญญัติทางสิทธิในทรัพย์สินอุตสาหกรรมอันเนื่องมาจากสาเหตุที่ได้มีการนำเนื้อหาที่ระบุอยู่ในหนังสือเล่มนี้ไปใช้ทางบริษัทจะไม่รับผิดชอบใดๆ ทั้งสิ้น

# สารบัญ

บทที่ 1 การควบคุมซีควেনซ์ (Sequence Control) คืออะไร	1-1
1.1 การควบคุมซีควেনซ์ (Sequence Control) คืออะไร	1-2
1.1.1 ซีควেনซ์ (Sequence) มีความหมายอย่างไร	1-2
1.1.2 ยกตัวอย่างจากสิ่งที่อยู่ใกล้ตัวเรา	1-2
1.2 อุปกรณ์ที่เกี่ยวข้องกับการควบคุมซีควেনซ์ (Sequence Control)	1-4
1.2.1 โครงสร้างของการควบคุมซีควেনซ์ (Sequence Control)	1-4
1.3 สิ่งจำเป็นในการควบคุมซีควেনซ์ (Sequence Control)	1-6
1.3.1 ทำความเข้าใจเกี่ยวกับภาพรวมของการควบคุมซีควেনซ์ (Sequence Control)	1-6
1.3.2 มาเรียนรู้คำศัพท์ใหม่กัน	1-7
1.4 ผูกการเดินสายไฟภายใต้เงื่อนไขของซีควেনซ์กันเถอะ	1-10
1.4.1 หลังจากที่ได้ทำการเรียนรู้คำศัพท์ใหม่	1-10
1.4.2 ศึกษาจากตัวอย่างอื่นๆ	1-13
1.5 มาจำสัญลักษณ์เกี่ยวกับซีควেনซ์กันเถอะ	1-15
1.5.1 ตารางตรวจสอบสัญลักษณ์หลักของซีควেনซ์	1-15
บทที่ 2 PLC คืออะไร	2-1
2.1 PLC คืออะไร	2-2
2.1.1 PLC มีไว้ทำอะไร	2-2
2.2 องค์ประกอบของ PLC	2-3
2.2.1 การควบคุมซีควেনซ์ (Sequence Control) ทำได้อย่างไร	2-3
2.2.2 หากมีการรวมระหว่าง รีเลย์ (Relay) กับ Timer	2-4
2.2.3 แผนผังซีควেনซ์ (Sequence diagram) และซีควেনซ์โปรแกรม (Sequence program)	2-6
2.3 การเดินสายไฟและโปรแกรม	2-9
2.3.1 การเดินสายไฟของ PLC และโปรแกรมมีลักษณะอย่างไร	2-9
2.4 ข้อดีของการใช้งาน PLC	2-10
2.4.1 ประโยชน์ที่จะได้รับจากการใช้งาน PLC	2-10
2.4.2 เปรียบเทียบกับการควบคุมรีเลย์ (Relay control)	2-11
บทที่ 3 การใช้งานโปรแกรม GX Works2	3-1
3.1 ความรู้พื้นฐานเพื่อการใช้งานโปรแกรม	3-2
3.1.1 โครงสร้างหน้าจอ	3-2
3.1.2 Workspace และ Project	3-5
3.2 การเรียกใช้งานโปรแกรม (Starting Gx Works2) และการสร้าง Project ใหม่ (New project)	3-6
3.2.1 เรียกใช้งานโปรแกรม (Starting GX Works2)	3-6
3.2.2 การสร้าง Project ใหม่ (New project)	3-7
3.3 การสร้างโปรแกรมหรือวงจร (Circuit)	3-9
3.3.1 การสร้างโปรแกรมหรือวงจร (Circuit) โดยใช้ฟังก์ชันคีย์ (Function Key)	3-9
3.3.2 การสร้างวงจร (Circuit) โดยใช้ Toolbar buttons	3-12
3.4 การเขียน Program ลงใน PLC	3-15
3.4.1 การเชื่อมต่อกับ PLC	3-15
3.4.2 "Transfer Setup" ใน GX Work2 และ "Writing programs"	3-16
3.4.3 การ Monitoring เพื่อดูการทำงานของ PLC	3-18

1

2

3

4

5

ผนวก

1

ผนวก

2

ผนวก

3

ผนวก

4

3.5	การแก้ไขโปรแกรม	3-20
3.5.1	การแก้ไขโปรแกรม	3-20
3.5.2	การแทรก/ลบ Rows	3-26
3.5.3	การ Cut & Copy (Pasting) ladder	3-28
3.6	การบันทึก Program ที่จัดทำเสร็จแล้ว	3-30
3.6.1	Save, Save as	3-30
3.6.2	การ Read project	3-31
3.7	Debug program	3-32
3.7.1	Ladder monitor	3-32
3.7.2	Device registration monitor	3-33
3.7.3	Device batch monitor (การเรียกดู Device ทั้งหมด)	3-34
3.7.4	Device test	3-35
3.7.5	การ Writing program ขณะ PLC กำลัง RUN	3-38
3.8	การใส่ Comment	3-39
3.8.1	ประเภทของ Comment	3-39
3.8.2	วิธีการสร้าง Device Comment	3-40
3.8.3	การจัดสร้าง Statements	3-42
3.8.4	การจัดสร้าง Notes	3-42

บทที่ 4 คำสั่งของซีควেনซ์ (Sequence instructions)	4-1
---	-----

4.1	คำสั่ง (Instruction)	4-2
4.1.1	Commands และ Programs	4-2
4.1.2	โครงสร้างของโปรแกรม	4-3
4.1.3	หน้าสัมผัสปกติเปิด a (N.O. Contact), หน้าสัมผัสปกติปิด b (N.C. Contact), Out instruction, End instruction	4-4
4.1.4	Series connection	4-5
4.1.5	Parallel connection	4-6
4.1.6	Series และ Parallel connection	4-7
4.1.7	SET instruction, RST instruction	4-8
4.1.8	Latch circuit (วงจรคงสภาพตัวเอง)	4-9
4.2	Timer circuit	4-10
4.3	Counter circuit	4-11
4.4	ลำดับ Program (Order of program)	4-12

บทที่ 5 การฝึกการเขียนโปรแกรม	5-1
-------------------------------	-----

5.1	ตัวอย่างการนำไปใช้งาน 1 (การควบคุมบันไดเลื่อน)	5-2
5.2	ตัวอย่างการนำไปใช้งาน 2 (เครื่องชั่ง)	5-4
5.3	ตัวอย่างการนำไปใช้งาน 3 (การควบคุมพัดลม)	5-6
5.4	ตัวอย่างการนำไปใช้งาน 4 (การควบคุมของเครื่อง Press)	5-8
5.5	ตัวอย่างการนำไปใช้งาน 5 (Timing Chart)	5-10
5.6	เฉลยคำตอบ	5-12

ภาคผนวก 1 GX Developer	1-1
ผนวก 1.1 ความรู้พื้นฐานเพื่อการใช้งาน GX Developer	1-2
ผนวก 1.1.1 โครงสร้างหน้าจอของ GX Developer.	1-2
ผนวก 1.1.2 เกี่ยวกับ "Project"	1-5
ผนวก 1.2 การเรียกใช้งาน GX Developer และการจัดทำ Project ใหม่	1-6
ผนวก 1.2.1 การเรียกใช้งาน GX Developer.	1-6
ผนวก 1.2.2 การจัดทำ Project ใหม่ (New Project)	1-7
ผนวก 1.3 การสร้างโปรแกรมหรือวงจร (Circuit)	1-9
ผนวก 1.3.1 การสร้างโปรแกรมหรือวงจร (Circuit) โดยใช้ฟังก์ชันคีย์ (Function Key)	1-9
ผนวก 1.3.2 จัดสร้าง Program โดยการใช้ Toolbar	1-12
ผนวก 1.4 การเขียน Program ลง PLC	1-15
ผนวก 1.4.1 การเชื่อมต่อกับ PLC.	1-15
ผนวก 1.4.2 "Transfer Setup" ใน GX Developer.	1-16
ผนวก 1.4.3 การเขียน Program.	1-17
ผนวก 1.4.4 การ Monitoring การทำงานของ PLC.	1-18
ผนวก 1.5 การแก้ไข Program (Editing program)	1-20
ผนวก 1.5.1 แก้ไข Program	1-20
ผนวก 1.5.2 การแทรก/ลบ Rows	1-26
ผนวก 1.5.3 การ cut & copy (Pasting) Ladder	1-28
ผนวก 1.6 การบันทึก Program ที่จัดทำเสร็จแล้ว	1-30
ผนวก 1.6.1 การ save, Save as	1-30
ผนวก 1.6.2 การ Save as.	1-31
ผนวก 1.6.3 การ Reading Project	1-32
ผนวก 1.7 การทำงานที่จำเป็นในการ Debug Program	1-33
ผนวก 1.7.1 Ladder Monitor.	1-33
ผนวก 1.7.2 Device registration monitor	1-35
ผนวก 1.7.3 Device batch monitor (การเรียกดู Device ทั้งหมด)	1-37
ผนวก 1.7.4 Device Test	1-38
ผนวก 1.7.5 การเขียน Program ขณะ Run Program.	1-40
ผนวก 1.8 การใส่ Comment	1-41
ผนวก 1.8.1 ประเภทของ Comment	1-41
ผนวก 1.8.2 วิธีการสร้าง Device comment	1-42
ผนวก 1.8.3 ดำเนินการจัดสร้าง Statement	1-44
ผนวก 1.8.4 การจัดสร้าง Note	1-44
ผนวก 1.9 การจัดสร้าง List Program	1-45
ผนวก 1.9.1 การแสดงหน้าจอ List edit	1-45
ผนวก 1.9.2 วิธีการป้อนคำสั่ง (Instructions).	1-45
ผนวก 1.9.3 การตรวจสอบรายละเอียดในการ Input List.	1-47
ภาคผนวก 2 แนะนำตัวอย่างการใช้งาน List programs	2-1
ผนวก 2.1 List programs.	2-2
ภาคผนวก 3 การใช้งาน Handy programming panel	3-1
ผนวก 3.1 การใช้ Handy programming panel (HPP)	3-2
ภาคผนวก 4 ผังการเชื่อมต่อ I/O ใน Training kit (Wiring diagram)	4-1
ผนวก 4.1 I/O Wiring diagram สำหรับ Training machine	4-2

# MEMO



## บทที่ 1

# การควบคุมซีควเอนซ์ (Sequence Control) คืออะไร

### มาทำความรู้จักกับ การควบคุมซีควเอนซ์ (Sequence Control)

“PLC หรือซีควเอนเซอร์ (Sequencer)” ที่จะแนะนำให้ทุกท่านรู้จักหลังจากนี้นั้น เป็นอุปกรณ์ที่ใช้ “การควบคุมซีควเอนซ์ (Sequence Control)” ถ้าอย่างนั้น “การควบคุมซีควเอนซ์ (Sequence Control)” คืออะไรกันแน่ ปกติเราจะไม่ค่อยได้ยินชื่อนี้ผ่านหูสักเท่าไร แต่ในความเป็นจริงจะมีการใช้งานผ่านอุปกรณ์ต่างๆ รอบตัวเรา มากมาย ซึ่งไม่ว่าใครก็น่าจะเคยสัมผัสกันมาแล้ว

ยกตัวอย่างเช่น เครื่องซักผ้า ชนิด Full automatic ที่มีระบบ “การควบคุมซีควเอนซ์ (Sequence Control)” อย่างเต็มรูปแบบ

เนื้อหาในบทนี้ จะมีการยกตัวอย่างจากอุปกรณ์ต่างๆ ที่ปรากฏอยู่รอบตัวเรา โดยที่หัวข้ออยู่ที่ “การควบคุมซีควเอนซ์ (Sequence Control)” เพื่อให้ทุกท่านทำความเข้าใจว่า “การควบคุมซีควเอนซ์ (Sequence Control)” ที่ว่านี้คืออะไร

# 1.1 การควบคุมซีควেনซ์ (Sequence Control) คืออะไร

## 1.1.1 ซีควেনซ์ (Sequence) มีความหมายว่าอย่างไร

“การควบคุมซีควেনซ์ (Sequence Control)”...

โดยทั่วไปแล้วเราจะไม่ค่อยคุ้นเคยกับคำนี้สักเท่าไร แต่ในความเป็นจริงแล้วรอบๆ ตัวเราจะมีการใช้คำนี้อยู่อย่างกว้างขวาง รวมถึงรูปร่างลักษณะของอุปกรณ์ตัวนี้เราก็น่าจะเคยเห็นหรือเคยสัมผัสกันอยู่บ้าง

ก่อนอื่น ถ้าเราลองค้นหาความหมายในพจนานุกรมคำว่า “ซีควেনซ์ (Sequence)” จะได้ความหมายว่า

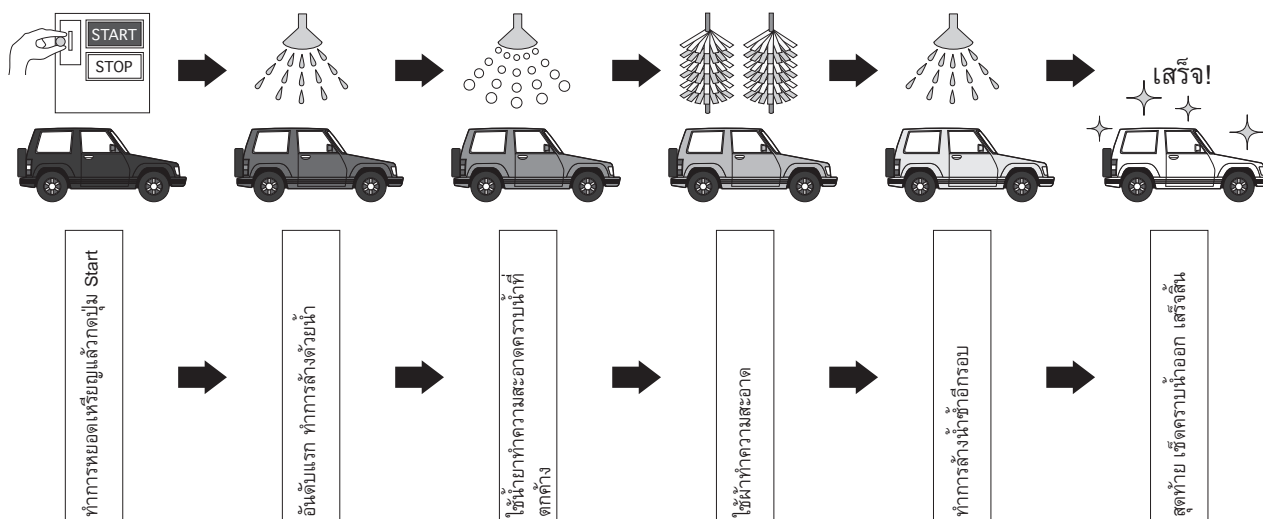
- ① การเกิดอย่างต่อเนื่อง, ต่อเนื่องกัน, การเกิดขึ้นหลายครั้งติดกัน
- ② ดำเนินต่อ, ความต่อเนื่อง
- ③ ลำดับ, คิว
- ④ เหตุการณ์ที่เกิดขึ้นตามมาอาทิเช่น, ผลกระทบ, ผลลัพธ์...

เพราะฉะนั้นคำว่า ซีควেনซ์ (Sequence) คือ สิ่งที่เกิดต่อเนื่อง หรือลำดับของการเกิดขึ้นของปรากฏการณ์

คำว่า “การควบคุมซีควেনซ์ (Sequence Control)” มาจากคำว่า “ซีควেনซ์ (Sequence)” ซึ่งมีความหมายว่า การทำงานที่ปฏิบัติตามลำดับขั้นตอนที่ได้มีการกำหนดไว้ล่วงหน้า อีกทั้งคำว่า “การควบคุม (Control)” หมายความว่า การทำงานที่ต้องเป็นไปตามที่วางแผน

## 1.1.2 ยกตัวอย่างจากสิ่งที่อยู่ใกล้ตัวเรา...

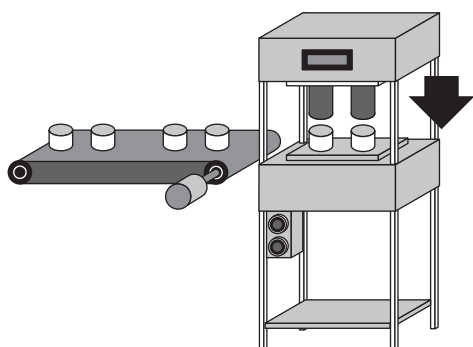
เครื่องล้างรถอัตโนมัติที่เราเห็นกันบ่อยๆ ตามปั้มเติมน้ำมัน



จากการแนะนำตัวอย่างเครื่องจักรในหน้าที่ผ่านมา ซึ่งถึงแม้ว่าจะเป็นลำดับขั้นตอนที่ไม่ยุ่งยากแต่ก็เป็นแนวคิดของ ซีควেনซ์ (Sequence) ซึ่งการควบคุมซีควেনซ์ (Sequence Control) นั่นก็คือ การควบคุมเครื่องให้ทำงานเป็นระบบอัตโนมัติเพื่อให้การทำงานทุกครั้งออกมาถูกต้อง

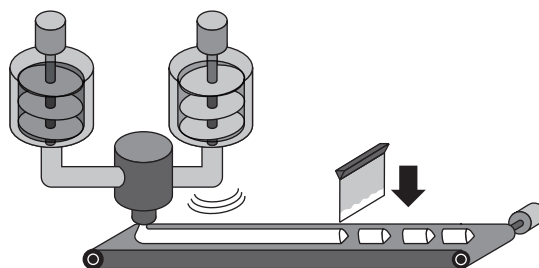
ในอีกทางหนึ่ง การควบคุมซีควেনซ์ (Sequence Control) นั้น ถูกใช้กับงานในหลายๆ ด้าน หลายๆ สาขา ซึ่งกลายเป็นแนวคิดที่ขาดอีกต่อไปไม่ได้

#### ■ FA (Factory automation) ในโรงงานเครื่องจักร



ใช้กับการควบคุม สายพาน (Conveyor) หรือการควบคุมในเครื่องจักร Machining และเครื่องจักร Assembly

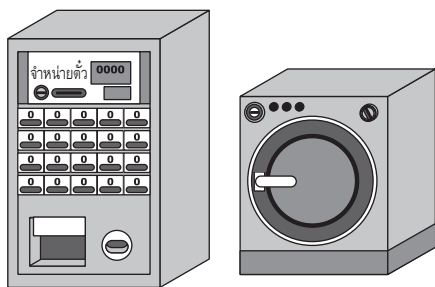
#### ■ เครื่องจักรผลิตอาหาร และการแปรรูป



ใช้ในการทำงานประเภท วัตถุดิบ, Injection, Heat-up, Cutting, Packing โดยมีการควบคุมผ่านเครื่องจักรในแต่ละประเภท

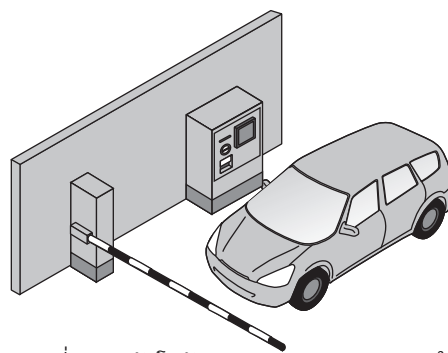
## มีการนำไปใช้อย่างแพร่หลายในหลายสาขา

#### ■ อุปกรณ์การทำงานในแต่ละประเภท



- เครื่องซักผ้าขนาดใหญ่
- เครื่องออกตัวอัตโนมัติ
- ตู้เย็น เครื่องแช่แข็ง หรืออุปกรณ์เสริมต่างๆ • นำไปใช้ในการควบคุมระบบอัตโนมัติต่างๆ

#### ■ นำไปใช้ในการควบคุมระบบอัตโนมัติต่างๆ



- การควบคุมที่จอดรถอัตโนมัติ
- การควบคุมการย้ายของบน Conveyor
- การควบคุมการเปิดปิดของประตูกันน้ำ
- การควบคุมภายในโรงเรือนเพาะปลูก
- การควบคุมสัญญาณไฟจราจร หรือป้ายไฟฟ้า
- การควบคุมประตูอัตโนมัติ

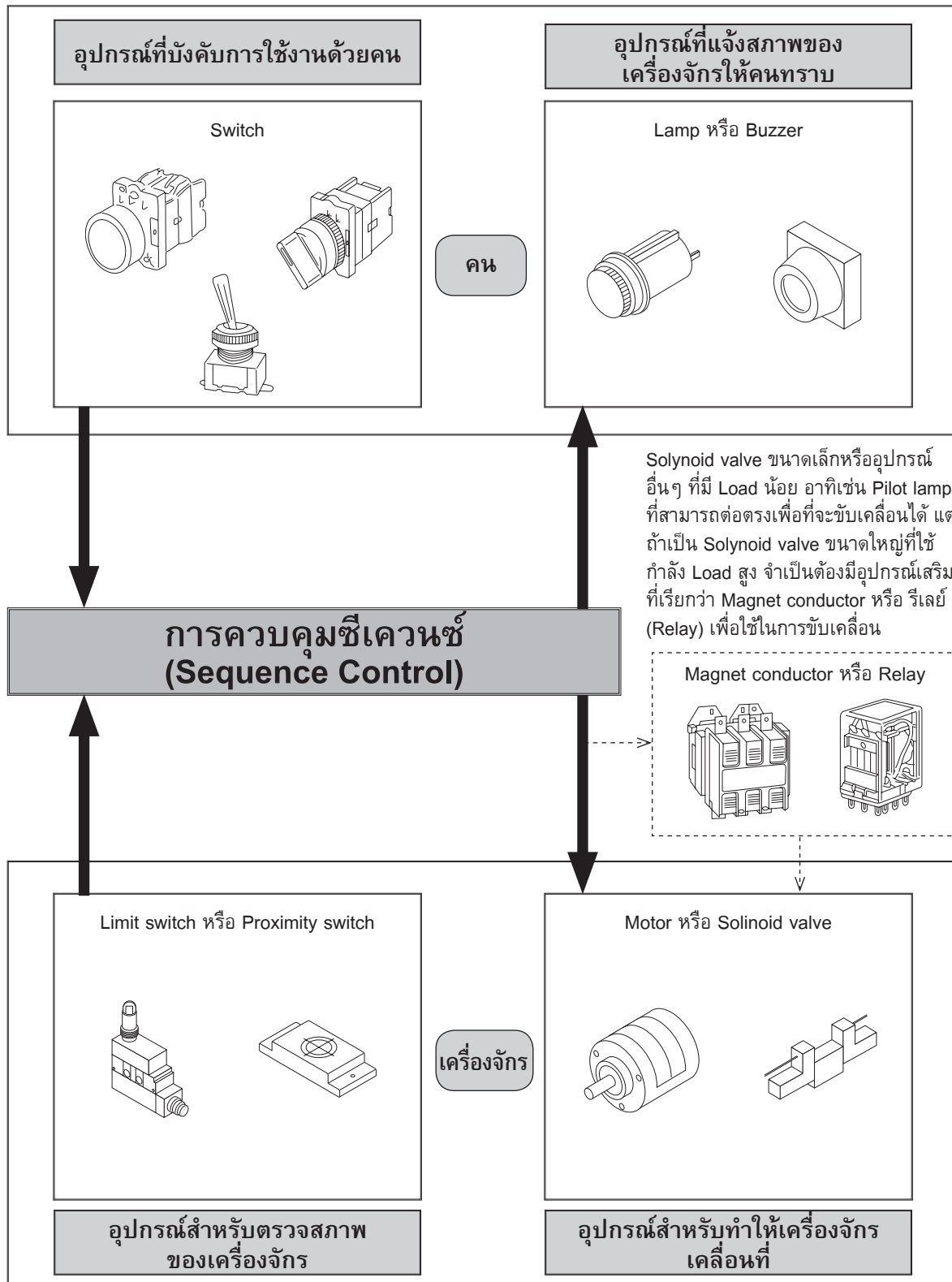
การควบคุมซีควেনซ์ (Sequence Control) ไม่ใช่เรื่องที่ยากเพราะเป็นสิ่งที่อยู่รอบๆ ตัวเรา

## 1.2 อุปกรณ์ที่เกี่ยวข้องกับการควบคุมซีควেনซ์ (Sequence Control)

### 1.2.1 โครงสร้างการควบคุมซีควেনซ์ (Sequence Control)

ในกรณีที่ต้องใช้การควบคุมเป็นแบบ PLC จำเป็นต้องมีอุปกรณ์ดังต่อไปนี้

อุปกรณ์ชนิดนี้แบ่งการทำงานออกเป็นหัวข้อใหญ่ๆ ได้ดังนี้ “อุปกรณ์ที่บังคับการใช้งานด้วยคน”, “อุปกรณ์ที่แจ้งสภาพของเครื่องจักรให้คนทราบ” “อุปกรณ์สำหรับตรวจสอบสภาพของเครื่องจักร”, “อุปกรณ์สำหรับทำให้เครื่องจักรเคลื่อนที่”



แบบดังต่อไปนี้นี้เป็นเพียงตัวอย่าง ซึ่งในความเป็นจริงยังมีอุปกรณ์อีกหลายแบบ

การควบคุมซีควেনซ์ (Sequence Control) เป็นการกำหนดลำดับการทำงานของอุปกรณ์เพื่อให้อุปกรณ์ทำงานตามคำสั่ง ซึ่งการทำงานของอุปกรณ์นี้จะมีเงื่อนไขประกอบไปด้วย “อุปกรณ์ที่บังคับการใช้งานด้วยคน” “อุปกรณ์สำหรับตรวจสอบสภาพของเครื่องจักร” แต่สำหรับเงื่อนไขในการที่จะเดินเครื่องนั้น จะประกอบไปด้วย “อุปกรณ์ที่แจ้งสภาพของเครื่องจักรให้คนทราบ” “อุปกรณ์สำหรับทำให้เครื่องจักรเคลื่อนที่”

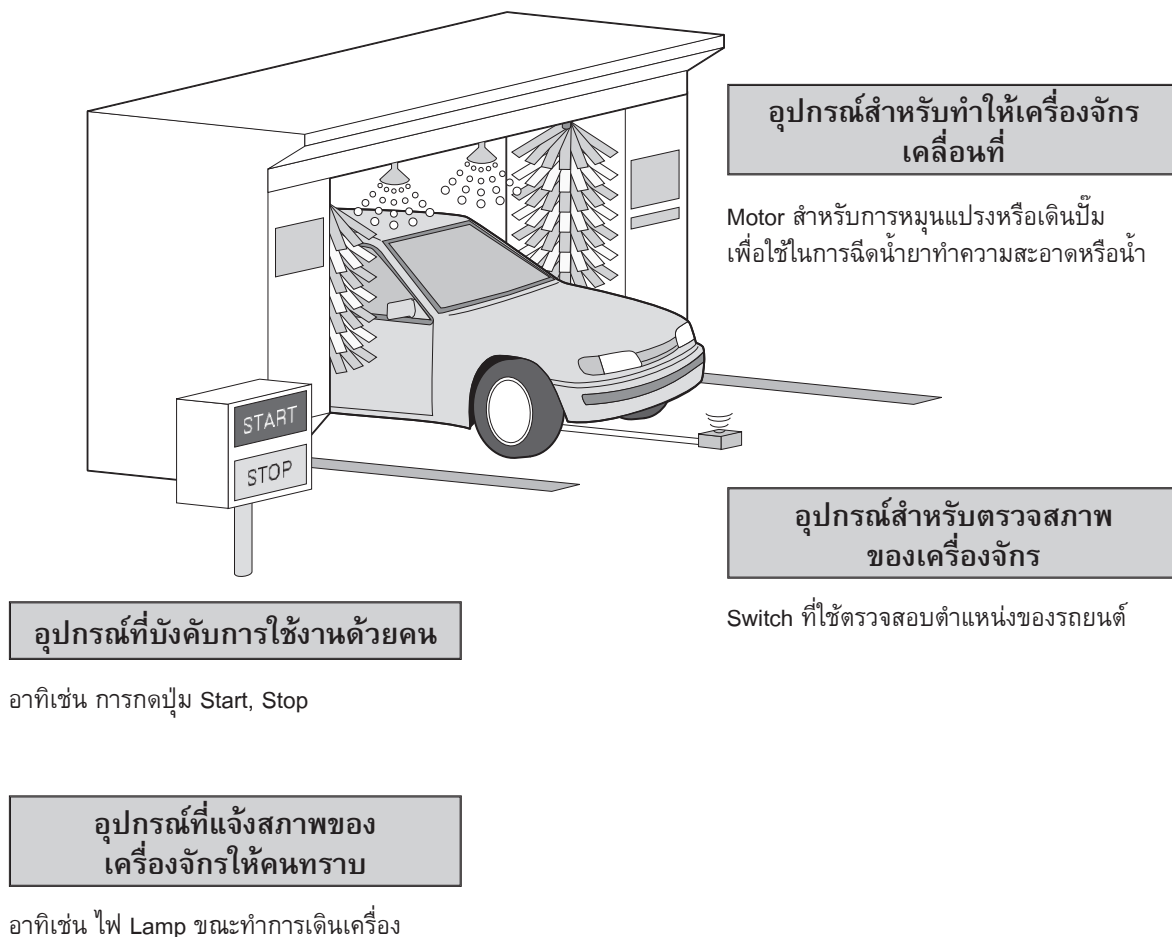
#### ตู้ปฏิบัติการ (Operation panel)

เป็นตู้ Panel ที่มีการติดตั้งอุปกรณ์ที่ใช้คนในการควบคุม (อาทิเช่น ปุ่ม Button switch, Selector switch เป็นต้น) หรืออุปกรณ์ที่ใช้ในการแจ้งสถานะของเครื่องจักร (อาทิเช่น Lamp หรือ Digital display machine เป็นต้น)

#### ตู้ควบคุม (Control panel)

เป็นตู้ Panel ที่มีการติดตั้งอุปกรณ์สำหรับควบคุมการทำงานของเครื่องจักร อาทิเช่น Magnetic contactor, รีเลย์ (Relay) หรือ PLC เป็นต้น

ตัวอย่าง ภายในเครื่องล้างรถอัตโนมัติจะประกอบไปด้วยอุปกรณ์ต่างๆ ที่ใช้ในการควบคุมซีควেনซ์ (Sequence Control)



## 1.3 สิ่งที่เป็นในการควบคุมซีควেনซ์ (Sequence Control)

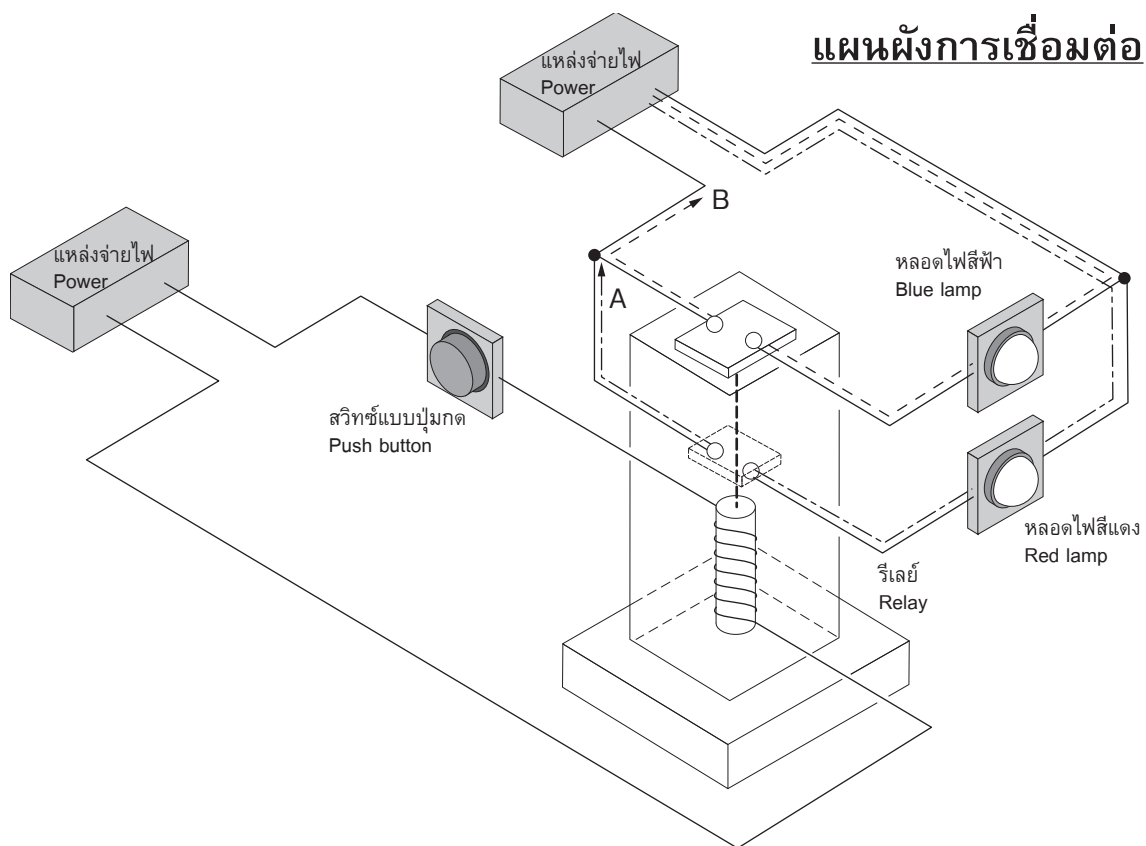
### 1.3.1 ทำความเข้าใจเกี่ยวกับภาพรวมของการควบคุมซีควেনซ์ (Sequence Control)

ลองมาคิดเกี่ยวกับการควบคุมซีควেনซ์ (Sequence Control) โดยอาศัยผังการเชื่อมต่อดังต่อไปนี้ อีกทั้งตรงนี้จะอธิบายคำศัพท์ที่ควรจำในการศึกษาการควบคุมซีควেনซ์ (Sequence Control) ด้วย

**ตัวอย่างที่ 1** : ปุ่ม Push button, ไฟ Lamp (ฟ้า, แดง), มีการใช้รีเลย์ (Relay) ในการเดินสายไฟในวงจรไฟฟ้า

#### ○ รายละเอียดของ การควบคุมซีควেনซ์ (Sequence Control)

- ① สถานะของอุปกรณ์ถ้าไม่ได้ทำการกด Push button เส้น B ----- จะเป็นเส้นที่กระแสไฟไหลผ่าน ไฟ Lamp สีฟ้าจะติด
  - ② เมื่อมีการกด Push button เส้น A —·—·—·— จะเป็นเส้นที่กระแสไฟไหลผ่าน ไฟ Lamp สีแดงจะติด
  - ③ เมื่อมีการปล่อยปุ่ม Push button ไฟสีฟ้าจะติดเหมือนเช่นข้อ ①
- ①~③ การทำงานของทั้งสามข้อนี้เป็นส่วนหนึ่งของ การควบคุมซีควেনซ์ (Sequence Control)



## 1.3.2 มาเรียนรู้คำศัพท์ใหม่กัน

### ● เกี่ยวกับหน้าสัมผัส (Contact) ประเภทต่างๆ

หน้าสัมผัส (Contact) มีหน้าที่ในการนำหรือตัดกระแส เมื่อมีการเปิด-ปิด

สิ่งที่จะเป็นพื้นฐานของหน้าสัมผัส (Contact) คือ “หน้าสัมผัสปกติเปิด a (“Normally open” contact หรือ N.O. Contacts)” และ “หน้าสัมผัสปกติปิด b (“Normally closed” contact หรือ N.C. Contacts)” ซึ่งหน้าสัมผัส (Contact) นั้นจะปรากฏอยู่ใน Switch หรือ รีเลย์ (Relay), Timer, Counter

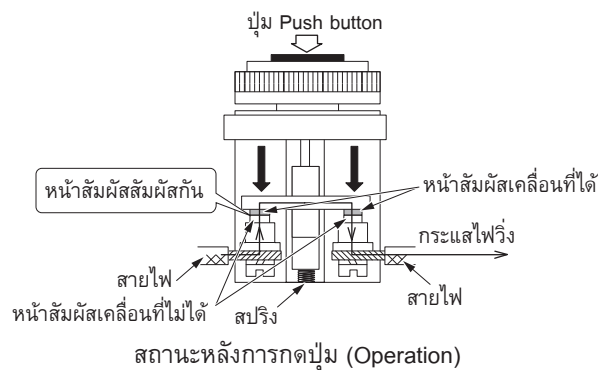
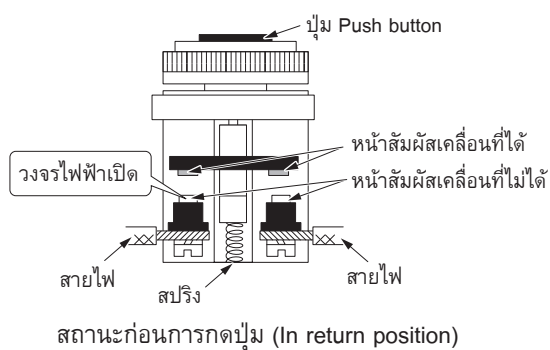
#### หน้าสัมผัสปกติเปิด a (N.O. Contacts)

หน้าสัมผัสปกติเปิด a (N.O. Contacts) จะเป็นชื่อเรียกของหน้าสัมผัส (Contact) ที่มีสถานะเปิด และเมื่อมีคำสั่ง หรือมีการกระตุ้น หน้าสัมผัส (Contact) ก็จะทำการปิดลง

※1 คำสั่ง หมายความว่า การควบคุมให้เกิดการทำงานหรือการเปลี่ยนแปลง ถ้าเป็น Push button การกดปุ่มก็คือ Command นั่นเอง

การทำงาน ในกรณีที่ปุ่ม Button switch

กรณีที่ปุ่มกดปุ่ม Button switch ตัว หน้าสัมผัส (Contact) จะเปิด และจะปิดเมื่อมีการกดปุ่ม

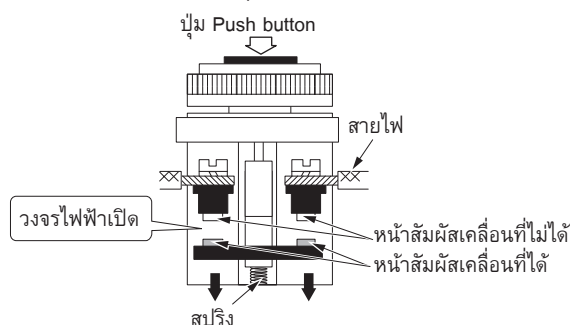
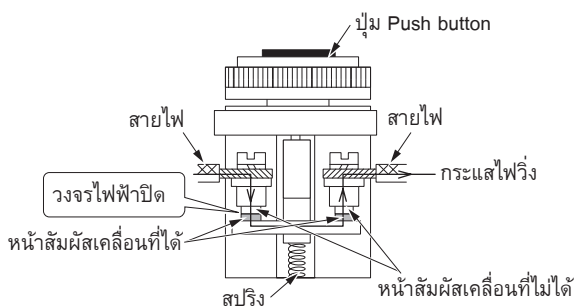


#### หน้าสัมผัสปกติปิด b (N.C. Contacts)

หน้าสัมผัสปกติปิด b (N.C. Contacts) คือ หน้าสัมผัส (Contact) ที่ปกติจะเป็นสถานะปิด เมื่อได้รับคำสั่ง หรือมีการกระตุ้น หน้าสัมผัส (Contact) จะทำการเปิด

การทำงาน ในกรณีที่ปุ่ม Button switch

ในกรณีที่ปุ่มกดปุ่ม Button switch ตัว หน้าสัมผัส (Contact) จะปิด และจะเปิดเมื่อมีการกดปุ่ม



### อ้างอิง

#### ที่มาของชื่อ หน้าสัมผัสปกติเปิด a (N.O. Contacts), หน้าสัมผัสปกติปิด b (N.C. Contacts)

ชื่อหน้าสัมผัสปกติเปิด a (N.O. Contacts), หน้าสัมผัสปกติปิด b (N.C. Contacts) มาจากไหน

หน้าสัมผัสปกติเปิด a (N.O. Contact) : Arbeit contact ... หน้าสัมผัสที่ทำงาน (ภาษาเยอรมัน)

หน้าสัมผัสปกติปิด b (N.C. Contact) : Break contact ... หน้าสัมผัสที่หยุด

กล่าวคือ การดึงเอา “อักษรตัวแรก” มาใช้นั่นเอง

บางครั้งก็เรียก หน้าสัมผัสปกติเปิด a ว่า “หน้าสัมผัสปกติเปิด” ภาษาอังกฤษ “N.O. Contacts (Normally-open contacts)” และเรียก หน้าสัมผัสปกติปิด b ว่า “หน้าสัมผัสปกติปิด” ภาษาอังกฤษ “N.C. Contacts (Normally-closed contacts)”

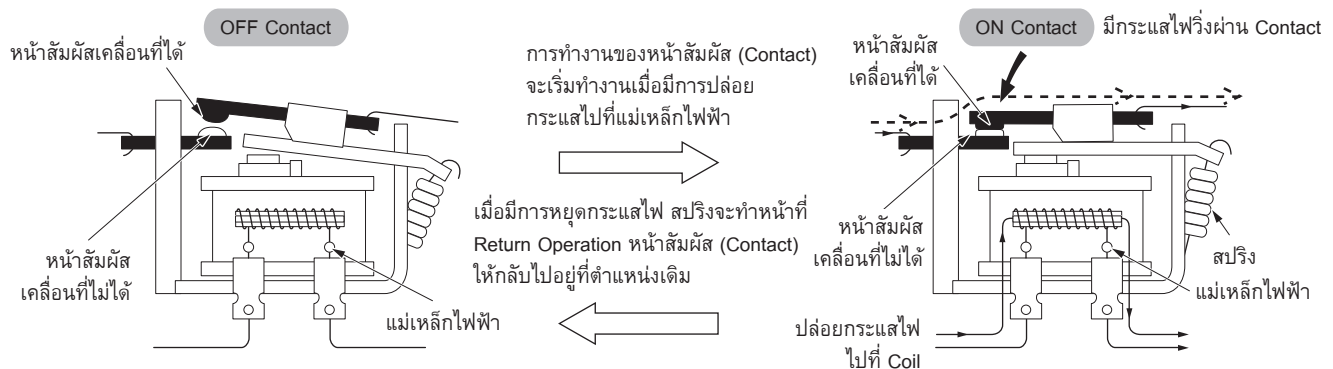
## ● เกี่ยวกับรีเลย์ (Relay)

รีเลย์ (Relay) เป็นชื่อเรียกอุปกรณ์ไฟฟ้าชนิดหนึ่งซึ่งมีประโยชน์ในการถ่ายทอดกระแสไฟฟ้า

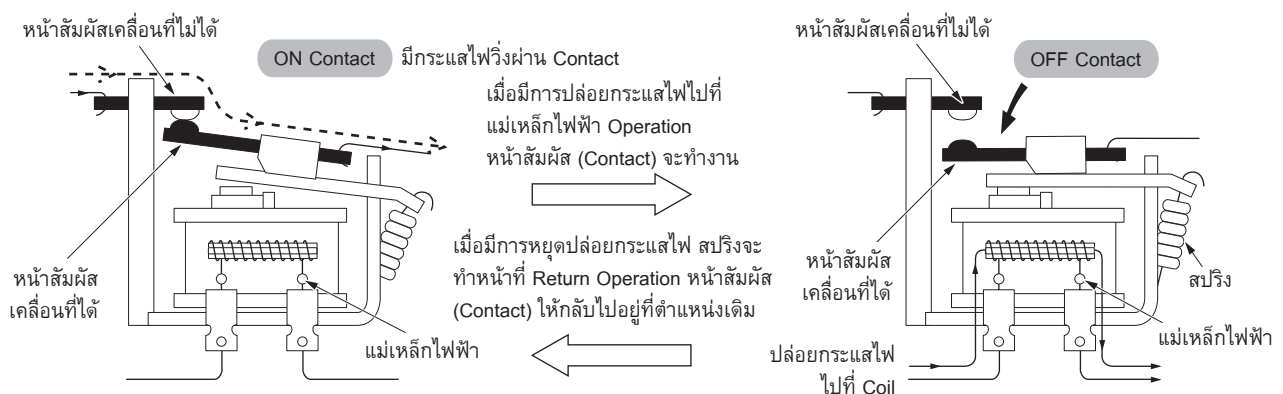
ในความเป็นจริงภายใน Relay จะมีแม่เหล็กไฟฟ้า เมื่อแม่เหล็กไฟฟ้ามีการทำงานจะทำการดูด Armature ขึ้น เพื่อทำการเปิด-ปิด หน้าสัมผัส (Contact)

※1 หน้าสัมผัส (Contact) ... หน้าสัมผัส (Contact) เป็นชื่อเรียกบริเวณที่มีการสัมผัสผ่านการเปิด-ปิด ที่มาจากการปล่อยหรือตัดกระแสไฟ นอกเหนือจาก รีเลย์ (Relay) จะมี Switch, Timer, Counter ก็จะมีหน้าสัมผัส (Contact) เช่นเดียวกัน อีกทั้งภายในหน้าสัมผัส (Contact) จะมีทั้งหน้าสัมผัสปกติเปิด a (N.O. Contacts), หน้าสัมผัสปกติปิด b (N.C. Contacts)

### หน้าสัมผัสปกติเปิด a (N.O. Contacts)



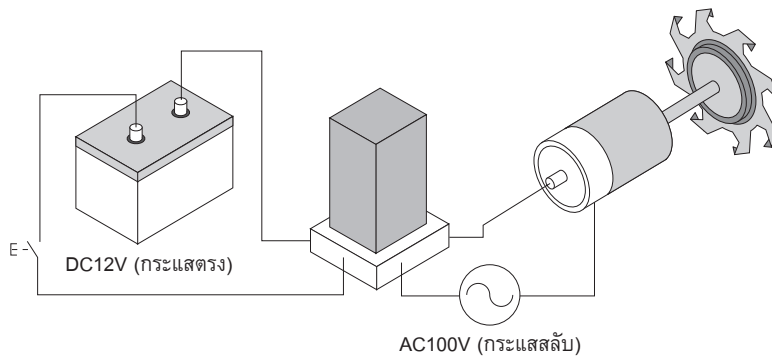
### หน้าสัมผัสปกติปิด b (N.C. Contacts)



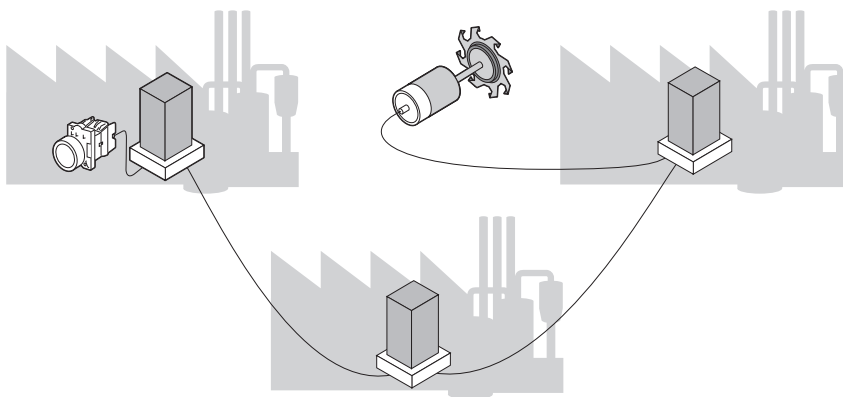


☆ ทำไม รีเลย์ (Relay) ถึงจำเป็น ?

- ① สัญญาณของ รีเลย์ (Relay) เพียงเล็กน้อยก็สามารถที่จะเดิน Motor หรือ Lamp ได้
- ② มีการควบคุมกระแสตรงเพื่อใช้เดินมอเตอร์กระแสสลับ หรือควบคุมกระแสไฟที่มีความแตกต่างกันได้ ดังรูป



- ③ สามารถเดินเครื่อง Motor หรือ Lamp ที่อยู่ในระยะห่างได้
- ④ สามารถควบคุมแบบซับซ้อนได้ถ้ามีการใช้ รีเลย์ (Relay) เป็นจำนวนมาก



# 1.4 ฝึกการเดินสายไฟภายใต้เงื่อนไขของซีเควนซ์กันเถอะ

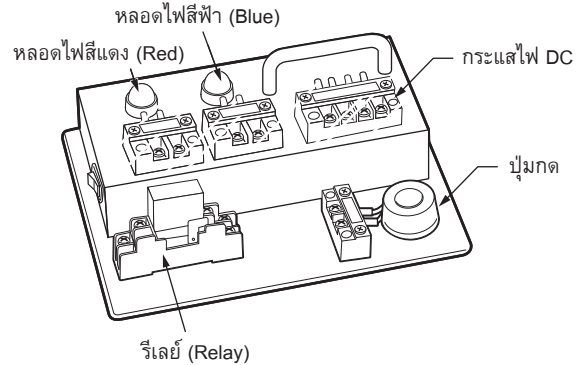
## 1.4.1 หลังจากที่ได้ทำการเรียนรู้ศัพท์ใหม่

หลังจากที่ได้ทำการเรียนรู้เรื่องหน้าสัมผัสรีเลย์ (Relay contact) แล้ว มาลองซ้อมจริงในหน้า 1-6 ตัวอย่างที่ 1

◎ รายละเอียดโดยย่อของการเดินสายรีเลย์ Relay wiring demonstration

### ส่วนประกอบ

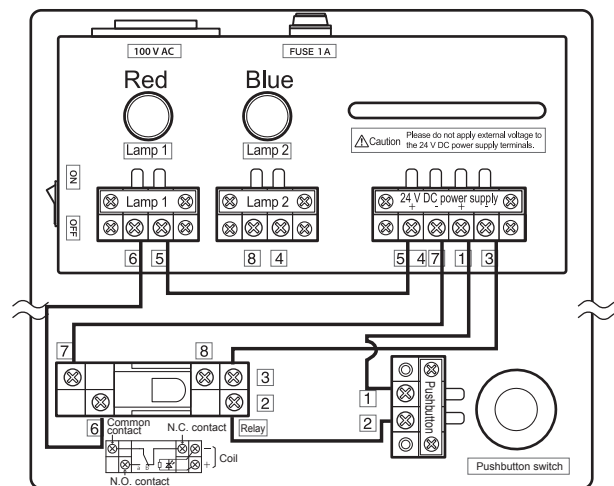
เคเบิลสายไฟ. . . . . 1 เส้น  
ปุ่มกด . . . . . 1 อัน  
หลอดไฟสีแดง (Red) . . . 1 อัน  
หลอดไฟสีฟ้า (Blue) . . . 1 อัน  
รีเลย์ (Relay) . . . . . 1 อัน  
สายลีด  
สีน้ำตาล (Brown), สีแดง (Red), สีส้ม (Orange),  
สีเหลือง (Yellow), สีเขียว (Green), สีฟ้า (Blue),  
สีม่วง (Purple), สีเทา (Gray) . . . . . อย่างละ 1 เส้น



### ● มาเดินสายไฟกัน

1. ตรวจสอบว่ากระแสไฟตัดหรือไม่
2. รูปที่ 1 แสดงถึงการเดินสายไฟ Red lamp  
ระหว่าง ขั้วต่อ (Terminal) เบอร์ 5 [สีเขียว (Green)]  
ระหว่าง ขั้วต่อ (Terminal) เบอร์ 6 [สีฟ้า (Blue)]  
ระหว่าง ขั้วต่อ (Terminal) เบอร์ 1 [สีน้ำตาล (Brown)]  
ระหว่าง ขั้วต่อ (Terminal) เบอร์ 2 [สีแดง (Red)]  
ระหว่าง ขั้วต่อ (Terminal) เบอร์ 3 [สีส้ม (Orange)]  
ระหว่าง ขั้วต่อ (Terminal) เบอร์ 7 [สีม่วง (Purple)]  
ใช้ Lead wires ในการ Connect
3. รูปที่ 2 แสดงถึงการเดินสายไฟ Blue lamp  
ระหว่าง ขั้วต่อ (Terminal) เบอร์ 4 [สีเหลือง (Yellow)]  
ระหว่าง ขั้วต่อ (Terminal) เบอร์ 8 [สีเทา (Gray)]  
ใช้ Lead wires ในการ Connect

รูปที่ 1 หลอดไฟสีแดง (Red)

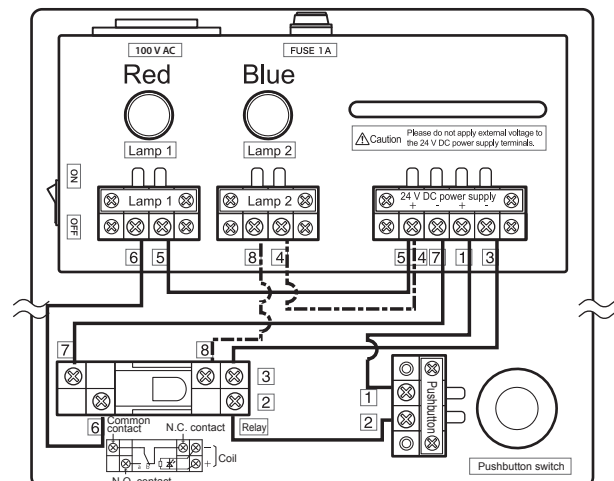


※1 แผนผังการเดินสายไฟ (Wiring diagram)

รูปที่ 1 หลอดไฟสีฟ้า (Blue)

ข้อระมัดระวัง) จากการตั้งกล่าว ขั้วต่อ (Terminal) เบอร์ 4 กับ 5  
ต้องเป็น Common ขั้วต่อ (Terminal)

※1 แผนผังการเดินสายไฟคืออะไร...  
เป็นแผนผังที่แสดงถึงอุปกรณ์ไฟฟ้าชนิดต่างๆ ที่ใช้บริเวณ  
วงจรไฟฟ้า หรือใช้ในการ Connect กับวงจรไฟฟ้า อีกทั้ง  
แผนผังดังกล่าวสามารถตรวจสอบโครงสร้างของอุปกรณ์  
ไฟฟ้าหรือการ Wiring ได้อย่างชัดเจนจึงสะดวกสบายต่อ  
การจัดทำอุปกรณ์ หรือสะดวกต่อการ Maintenance



## ● มาตรวจสอบการทำงานกัน

จงบารายละเอียดของ การควบคุมซีควেনซ์ (Sequence Control) ในหน้า 1-6 และนำมาเรียบเรียงใหม่โดยใช้คำศัพท์ที่ตนเรียนรู

- ① เมื่อมีการปล่อยกระแสเข้า หน้าสัมผัสปกติเปิด a (N.O. Contacts) กระแสจะทำการวิ่งผ่าน Route B และไฟที่หลอดไฟสีฟ้า (Blue lamp) จะติด
- ② หลังจากนั้น ถ้าได้กดปุ่ม Push button switch (Push button switch ก็คือ หน้าสัมผัสปกติเปิด a (N.O. Contacts)) จากการทำงานของ รีเลย์ (Relay) จะทำให้หน้าสัมผัสปกติเปิด a (N.O. Contacts) ปิด จึงทำให้กระแสไฟวิ่งผ่าน Route A ไฟสีแดงจะติด
- ③ เมื่อมีการปล่อยปุ่ม Push button จะทำให้ไฟสีฟ้าติดอีกครั้ง

### อธิบายการทำงานของ ซีควেনซ์ (Sequence) ผ่าน Flow chart

Flow chart คืออะไร

การควบคุมซีควেনซ์ (Sequence Control) คือ การออกแบบโครงสร้างวงจร

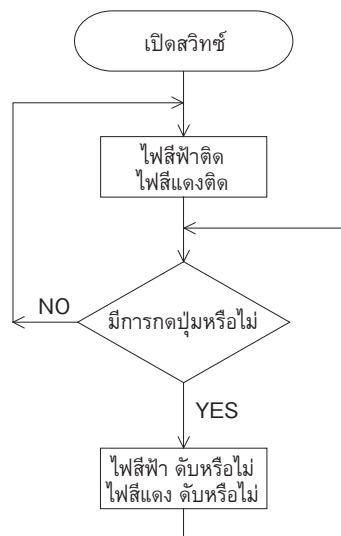
ไฟฟ้าโดยนำอุปกรณ์ต่างๆ มาผสมกัน

ดังนั้นอุปกรณ์ที่มีการประกอบเข้าไประวมถึงรายละเอียดและลำดับการทำงาน

เมื่อนำมาเขียนเป็นรายละเอียดจะทำให้เกิดความเข้าใจโดยรวมได้ยาก

สำหรับ Flow chart นั้นจะใช้กรอบสี่เหลี่ยมและสัญลักษณ์ลูกศรเป็นเครื่องหมาย

เพื่อสามารถที่จะสื่อให้เกิดความเข้าใจได้ง่าย



### การแสดง ซีควেনซ์ (Sequence) จาก Time chart

Time chart คืออะไร

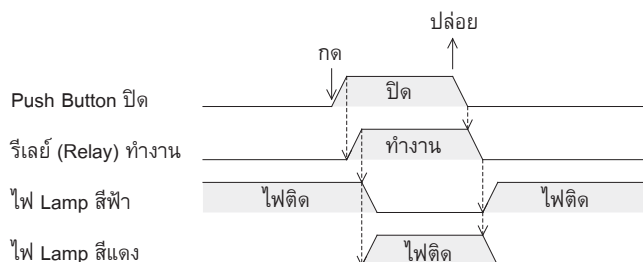
มีการสรุปให้เข้าใจง่ายในเรื่องการปรับเวลาให้ตรงกับลำดับการเดินเครื่อง

ระบุอุปกรณ์ที่ใช้ในการควบคุมที่แน่นอน และมีการระบุการเปลี่ยนแปลงของเวลาที่แน่นอน

สังเกตที่สัญลักษณ์เส้นประ จะทราบว่าอุปกรณ์ไหนที่มีการทำงานที่เกี่ยวข้องบ้าง

อาจมีบางกรณี que แสดงเป็นสัญลักษณ์เพื่อให้มีความเข้าใจง่าย หรือ

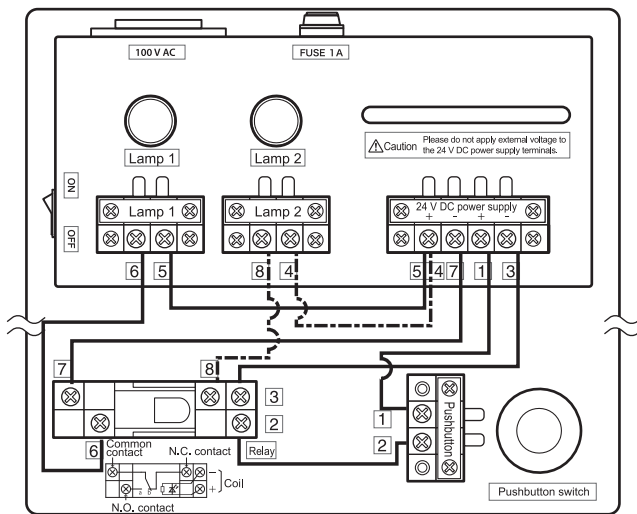
อาจจะสังเกตรายละเอียดของ การควบคุมซีควเอนซ์ (Sequence Control) จาก Flow chart หรือ Time chart



## แผนผังการเดินสายไฟ (Wiring diagram) และแผนผังซีควเอนซ์ (Sequence diagram)

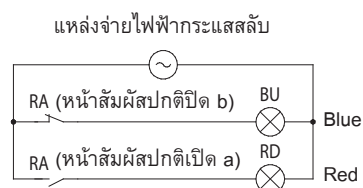
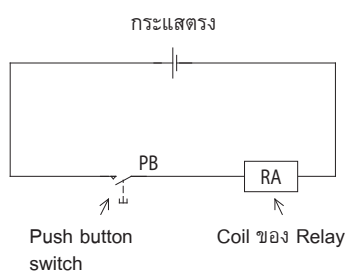
แผนผังการเดินสายไฟ (Wiring diagram) หรือ โครงสร้างของเครื่องจักรนั้น ถ้าหากสำหรับงานเดินสายไฟแล้วถือว่าเป็นภาพที่เข้าใจได้ง่าย แต่ถ้านำงานวงจรไฟฟ้าที่มีความซับซ้อน อาจทำให้เข้าใจถึงลำดับการทำงานได้ยาก และเพื่อที่จะทำให้เกิดความเข้าใจได้มากยิ่งขึ้น เราลองมาจัดเป็นรูปแบบของ PLC drawing

### การเดินสายไฟ (Wiring diagram)



หมายเหตุ) ด้วยเครื่องในการฝึกสอนรุ่น FX-I/O-DEMO2 ทั้งรีเลย์ และหลอดไฟจะทำงานด้วยแหล่งจ่ายไฟฟ้ากระแสตรง ขนาด 24 V อย่างไรก็ตาม โดยทั่วไปรีเลย์จะทำงานด้วยแหล่งจ่ายไฟฟ้ากระแสตรงขนาด 24 V และหลอดไฟจะทำงานด้วยแหล่งจ่ายไฟฟ้ากระแสสลับขนาด 100 V ซึ่งแสดงอยู่ในรูปด้านล่างนี้

### แผนผังซีควเอนซ์ (Sequence diagram)



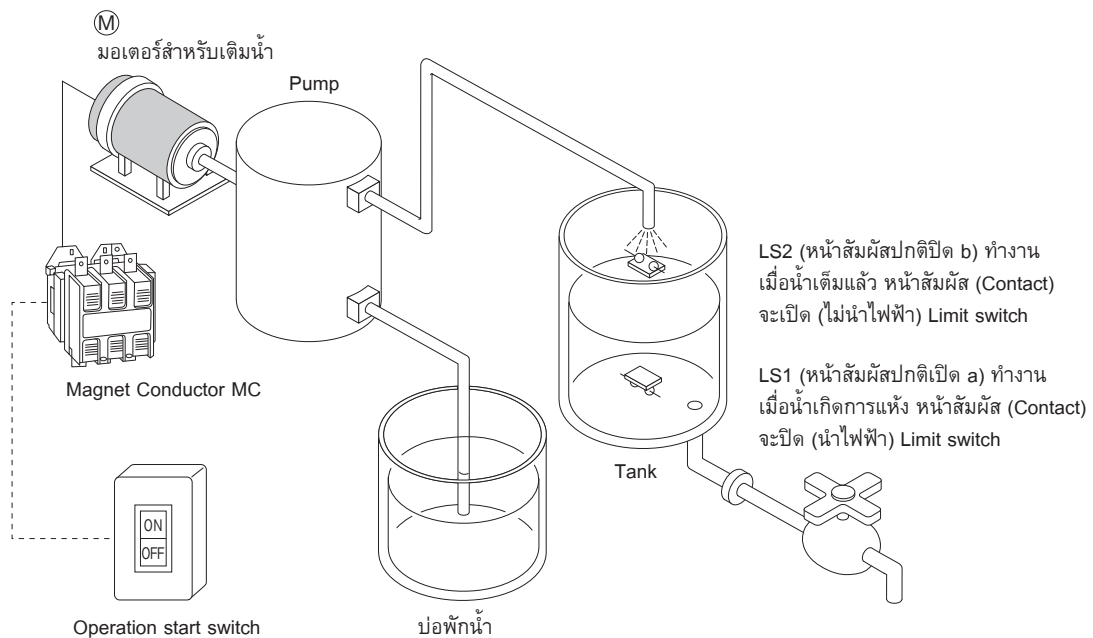
แผนผังซีควเอนซ์ (Sequence diagrams) จะถูกจัดทำขึ้นโดยเรียงตามลำดับของการทำงานที่มาจากวงจรไฟฟ้าจำนวนมาก และมีการจัดเรียงเนื้อหาแผนผังการเชื่อมต่อ เพื่อให้การทำงานเป็นไปอย่างเข้าใจง่าย อีกทั้งวิธีการจัดทำ Diagrams ในรูปแบบนี้ยังมีการกำหนดให้เป็นวิธีมาตรฐาน เมื่อบุคคลที่สามเข้ามาเปิดดูก็จะสามารถทำความเข้าใจได้ง่าย

## 1.4.2 ศึกษาจากตัวอย่างอื่น ๆ

**ตัวอย่างที่ 2** : ควบคุมระดับน้ำใน Tank

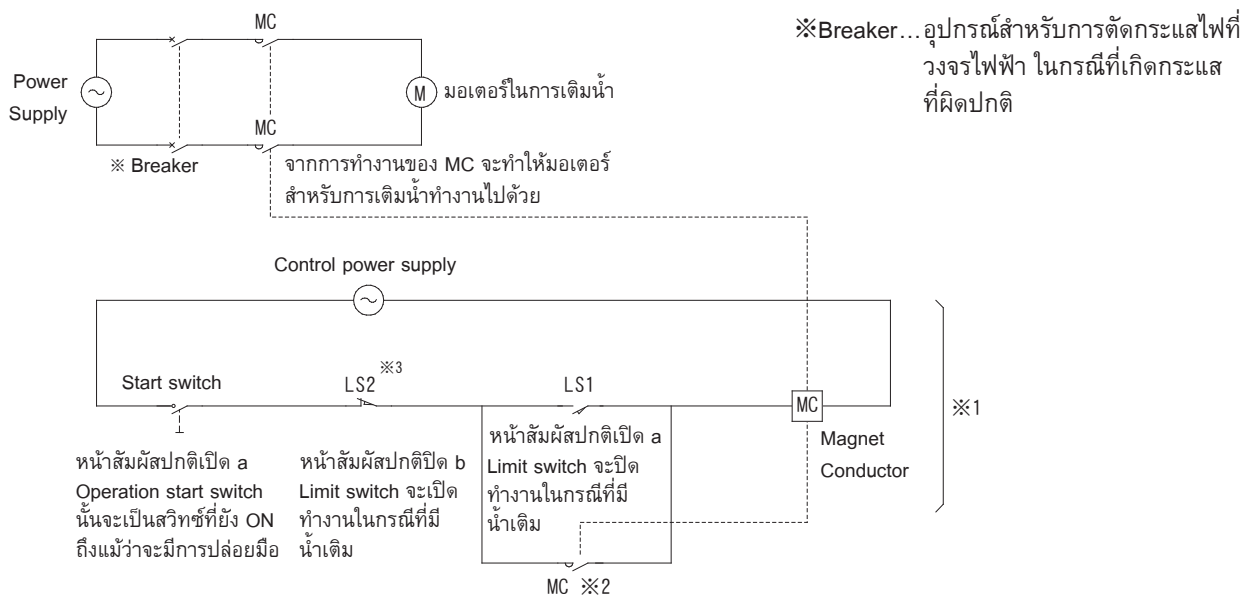
### ◎ รายละเอียดการควบคุมซีควেনซ์ (Sequence Control)

- ① เมื่อมีการปิดสวิตช์การ Start operation เกิดขึ้น เมื่อน้ำเกิดการแห้ง Limit switch LS1 จะทำการปิด Magnet conductor MC จะเริ่มทำงานเพื่อที่จะให้มอเตอร์ที่ใช้ในการเติมน้ำเริ่มการทำงาน ถึงแม้ว่าน้ำจะอยู่ในระดับปานกลาง Magnet conductor MC จะทำงานเพื่อรักษาระดับของน้ำ
- ② ถ้าน้ำเต็ม Limit switch LS2 จะเปิด การทำงานของ Magnet conductor MC เพื่อการรักษาระดับจะถูกยกเลิก อีกทั้งมอเตอร์ที่ใช้สำหรับการเติมน้ำก็จะหยุดไปด้วย
- ③ เมื่อน้ำถึงระดับที่จำเป็นต้องเติมน้ำ มอเตอร์ที่ใช้สำหรับการเติมน้ำจะทำการเติมน้ำแบบอัตโนมัติอีกครั้ง



จากการทำงานของมอเตอร์ผ่านการควบคุมชุดนี้ เมื่อน้ำเกิดการแห้งจนถึงระดับปานกลางมอเตอร์จะทำงาน เมื่อน้ำเกิดเต็มจนถึงระดับปานกลางมอเตอร์จะหยุดการทำงาน ถึงแม้จะอยู่ในระดับปานกลางเหมือนกันแต่สถานะการทำงานจะต่างกัน ซึ่งเราเรียกการทำงานดังกล่าวว่า Hysteresis operation (การทำงานตามประวัติ) จากวิธีนี้เราสามารถลดปริมาณความถี่ในการเดินและการหยุดของมอเตอร์ได้

## ● แผนผังซีควเอนซ์ (Sequence diagram)



※1 ซีควเอนซ์ (Sequence) ในบริเวณนี้จะถูกเปลี่ยนไปยัง PLC ตามปกติ

※2 เกี่ยวกับ Self-holding circuit (วงจรคงสภาพตัวเอง)

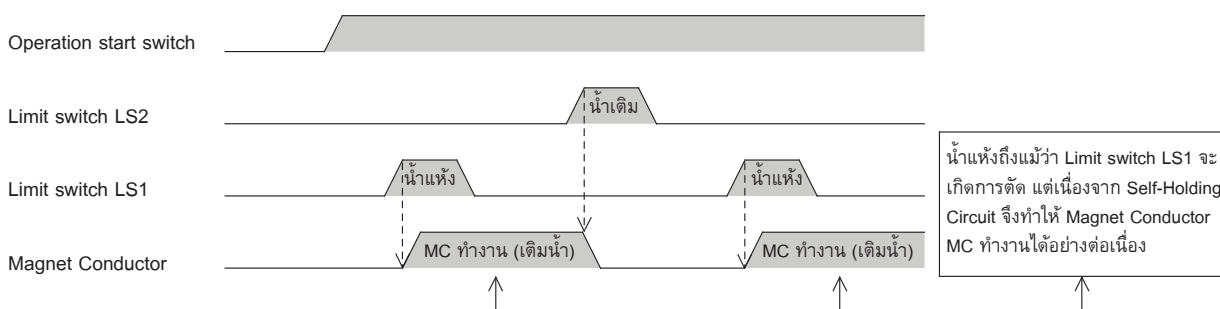
Self-holding circuit (วงจรคงสภาพตัวเอง) คืออะไร อุปกรณ์ที่ต้องรับคำสั่งจากภายนอกอย่างเช่น รีเลย์ (Relay), Magnet conductor MC

จากหน้าสัมผัส (Contact) ที่มีอยู่ในตัวรีเลย์ (Relay), Magnet conductor MC จะทำการ Bypass และสร้างวงจรการทำงานขึ้น อีกทั้ง Self-holding circuit จะแบ่งเป็นแบบปุ่มกดซึ่งจะมีการตัดเมื่อทำการปล่อยปุ่มหรือแบบที่มีหน่วยความจำ ซึ่งจะสามารถทำให้เกิดการทำงานแบบต่อเนื่องได้

ในกรณีนี้จะมีการต่อ Limit switch LS1 แบบขนานต่อเข้ากับ หน้าสัมผัส (Contact) ของ Magnet conductor MC เพื่อใช้ในการขับเคลื่อนมอเตอร์ ซึ่งเราเรียกววงจรแบบนี้ว่า Self-holding circuit

※3 สำหรับ Limit switch LS2 นั้นมีหน้าที่ในการตัดวงจรของ ※2 Self-holding circuit หรือหยุดการทำงานของมอเตอร์ในการเติมน้ำ และเมื่อมีการต่อเข้ากับ หน้าสัมผัสปกติปิด b (N.C. Contacts) หรือ Limit switch LS2 เมื่อเกิดปัญหาหน้าสัมผัสเกิดผิดปกติ ภายในวงจร หรือสายไฟเกิดขาด Self-holding circuit จะเกิดการตัด ทำให้อมอเตอร์สำหรับเติมน้ำเกิดการหยุด

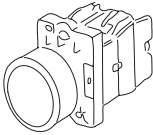




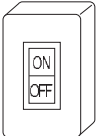




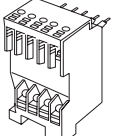





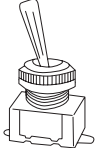




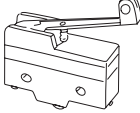












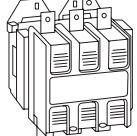




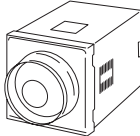




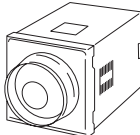




## การทำงานของวงจรคงสภาพตัวเอง (Time chat)



# 1.5 มาจำสัญลักษณ์ที่เกี่ยวข้องกับซีเควนซ์กันเถอะ

## 1.5.1 ตารางตรวจสอบสัญลักษณ์หลักของซีเควนซ์

(JIS C 0617)

แยกตามหน้าสัมผัส (Contacts) แยกตามผลิตภัณฑ์		หน้าสัมผัสปกติเปิด a (ปกติเปิดเสมอ)		หน้าสัมผัสปกติปิด b (ปกติปิดเสมอ)		ที่มาของการขับเคลื่อน (Drive Source)	
		เขียนแนวนอน	เขียนแนวตั้ง	เขียนแนวนอน	เขียนแนวตั้ง		
Push button switch (Automatic Restoration)						Manual	
Push button switch (Residual Movements)						Manual	
Thermo Relay (OCR)						Heater 	Movements
Switch (General)						Manual	
Limit switch (Machine control)						 Dock	 Cam
รีเลย์ (Relay)						 Electrode Coil	
Electrode Contactor (Magnet Conductor)							
Timer on Delay							
Timer off Delay							

# MEMO



# PLC มันเป็นอย่างนี้เอง!!!

## บทที่ 2

### PLC คืออะไร

---

2

#### PLC คืออะไร...

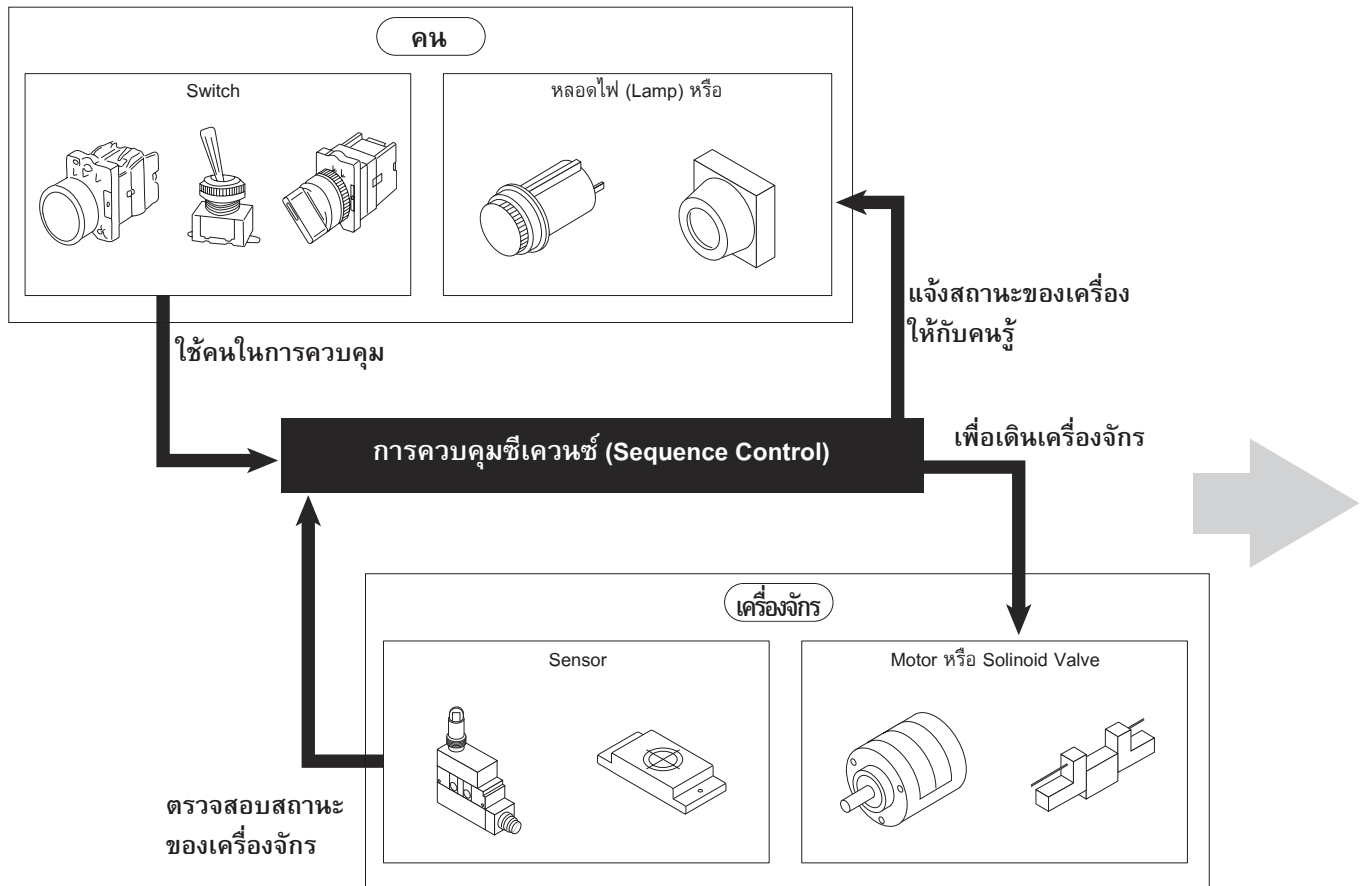
PLC มีชื่อเรียกอีกชื่อหนึ่งว่า Programmable Controller (PLC) หรือ การควบคุมซีควেনซ์ Sequence Control (SC) ซึ่งมีคำนิยามเกี่ยวกับอุปกรณ์นี้ว่า “คือ สิ่ง que เอาไว้ควบคุมอุปกรณ์เครื่องจักรต่างๆ โดยผ่านสิ่งที่เรียกว่า Input-Output ซึ่งเป็นอุปกรณ์อิเล็กทรอนิกส์ที่ภายในประกอบไปด้วยหน่วยความจำเพื่อจดจำคำสั่งที่สามารถสั่งงานได้ด้วยโปรแกรม (Programmable)”

#### ที่จริงแล้ว...

มาถึงตอนนี้ ขอให้คิดว่า มันคือ อุปกรณ์ที่ทำให้ “การควบคุมซีควেনซ์ (Sequence Control)” ที่เคยกระทำโดยการเดินสายไฟ รีเลย์ (Relay) และ Timer ให้เกิดขึ้นได้จริงด้วย “โปรแกรม” ง่าย ๆ

## 2.1 PLC คืออะไร

### 2.1.1 PLC มีไว้ทำอะไร



ภายในการควบคุมซีควেনซ์ (Sequence Control) จะมีสัญญาณคำสั่งเกี่ยวกับ "ใช้คนในการควบคุม" "ตรวจสอบสถานะของเครื่องจักร" หรือ อาจจะเรียกว่าสัญญาณเกี่ยวกับเงื่อนไข อีกทั้งมีการเรียก "แจ้งสถานะของเครื่องจักรให้คนรู้" "ทำการขับเคลื่อนเครื่องจักร"

ซีควেনเซอร์ (Sequencer) หรือ PLC คือ "การควบคุมซีควেনซ์ (Sequence Control)" ที่อยู่ในรูปตัวนับ มีหน้าที่ในการควบคุมอุปกรณ์เหล่านั้น ซึ่งหน้าที่ของ PLC หรือซีควেনเซอร์ (Sequencer) นั้นเป็นไปตามชื่อ นั่นก็คือการควบคุมซีควেনซ์ (Sequence Control) นั่นเอง

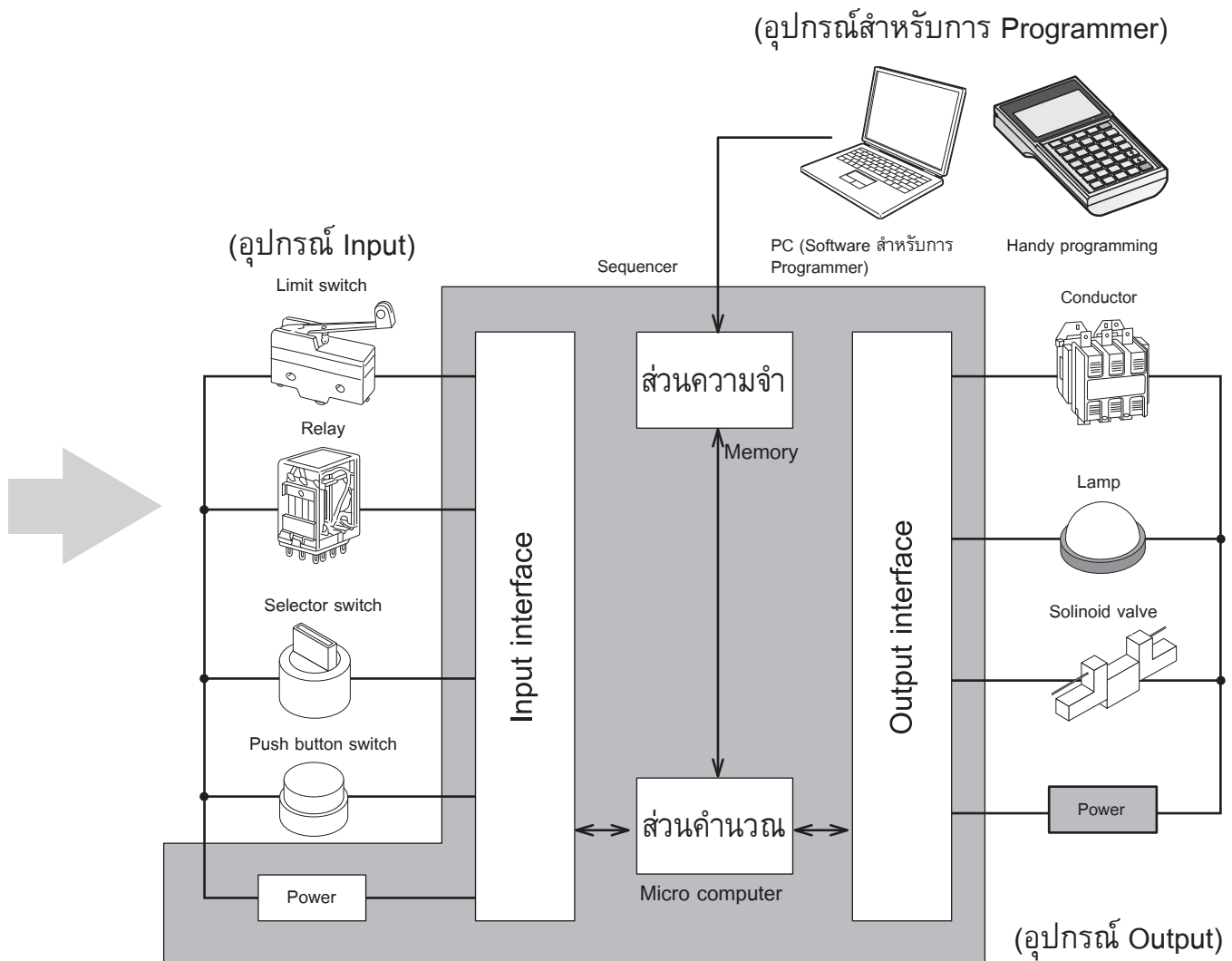
#### อ้างอิง

#### ซีควেনเซอร์ (Sequencer) เป็นคำศัพท์ที่มีตชบชช อีเล็คทริกสร้างขึ้นหรือ?

โดยทั่วไปไม่มีการใช้ชื่อ "ซีควเอนเซอร์ (Sequencer)" กันอย่างกว้างขวาง ในปัจจุบันสมาคมผู้ผลิตเครื่องใช้ไฟฟ้าประจำประเทศญี่ปุ่น (JEMA) มีชื่อเรียกอย่างเป็นทางการว่า Programmable Controller (PLC) ซึ่งจะเรียก PLC หรือ Sequencer ก็ได้ แต่ในประเทศไทยนิยมเรียกกันว่า PLC และในเอกสารชุดนี้จะใช้คำว่า PLC เป็นหลัก แต่เดิมชื่อ "Sequencer" มีประวัติการใช้งานมาก่อนคำว่า PLC ในบางส่วน ซึ่งหลังจากที่ Mitsubishi Electric ได้วางจำหน่าย PLC ในชื่อ Sequencer โดยที่รุ่นที่เป็นตัวแทนหลักจะเป็น K Series และ F Series

## 2.2 องค์ประกอบของ PLC

### 2.2.1 การควบคุมซีควেনซ์ (Sequence Control) ทำได้อย่างไร



PLC จะมีการเชื่อมต่อกับ Load สำหรับการขับเคลื่อนโดยมีเงื่อนไขของสัญญาณหรือคำสั่งที่เป็นสัญญาณ ในแต่ละอุปกรณ์ตามเนื้อหาในหน้าซ้ายมือ อีกทั้งสิ่งที่จะเชื่อมต่อกับฝั่ง Input คือ "อุปกรณ์ Input" และสิ่งที่จะเชื่อมต่อกับฝั่ง Output เรียกว่า "อุปกรณ์ Output"

ซึ่งอุปกรณ์ Input-Output จะมีการเชื่อมต่อระหว่าง PLC กับ ขั้วต่อ (Terminal) ในแต่ละตัว

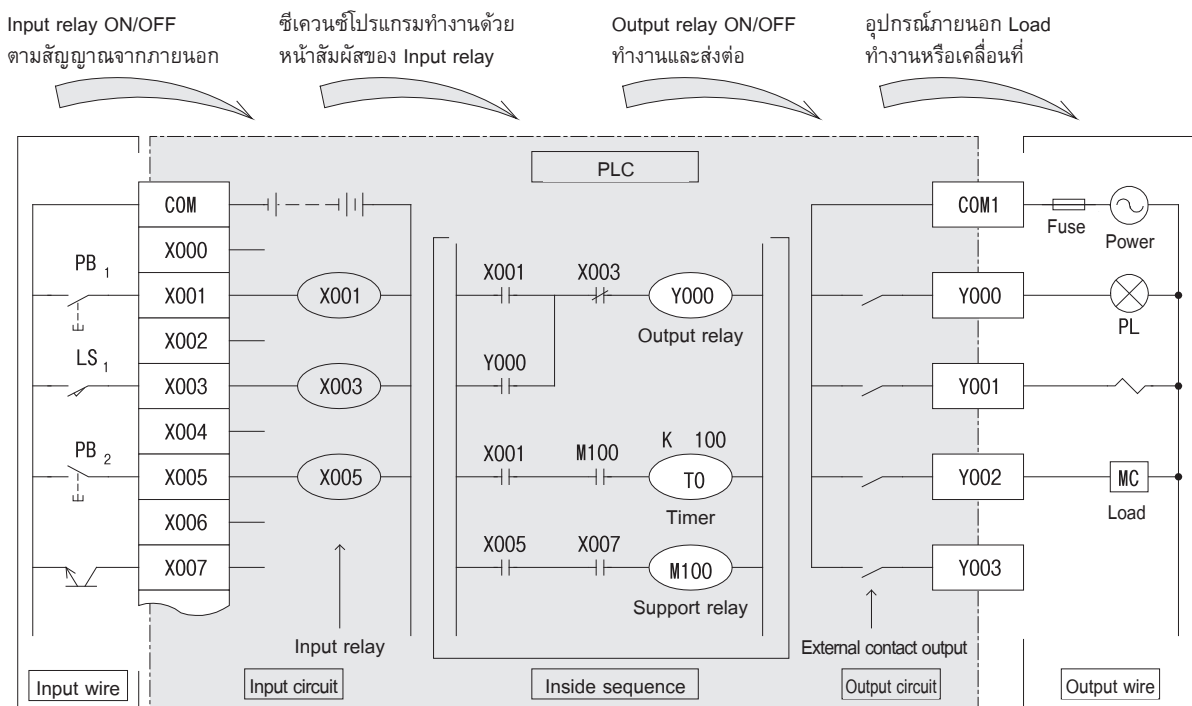
สำหรับ PLC สามารถเชื่อมต่อกับอุปกรณ์ Input-Output ในแต่ละตัว การเชื่อมต่อเพื่อที่จะทำการควบคุมซีควেনซ์ (Sequence Control) นั้น จะมีขั้นตอนอิเล็กทรอนิกส์ภายใน PLC

การเชื่อมต่อกับภายใน PLC จะมีการใช้คำศัพท์เฉพาะ (คำสั่ง) สำหรับ Sequence

ซึ่งอุปกรณ์ที่มีการนำคำสั่งทั้งหมดมาประกอบกันคือ ซีควেনซ์โปรแกรม (Sequence program) โดยที่เราจะมีการควบคุมซีควেনซ์ (Sequence Control) ผ่าน Program นี้

เพราะฉะนั้นไม่จำเป็นที่จะต้องเดินสายไฟภายนอก

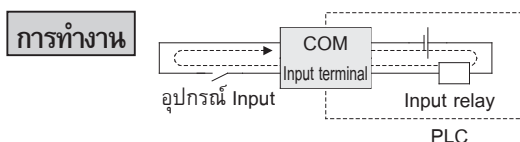
## 2.2.2 หากมีการรวมระหว่าง รีเลย์ (Relay) กับ Timer



อุปกรณ์ Input อุปกรณ์ Output ซีควเอนซ์โปรแกรม (Sequence program) มีโครงสร้างตามรูปภาพด้านบน  
 อุปกรณ์ Input เชื่อมต่อกับ Input relay ของ PLC และอุปกรณ์ Output ทำการควบคุมผ่านหน้าสัมผัสสำหรับ Output ภายนอก (External output)

### ● Input relay

Input relay มีหน้าที่ในการแปลงสัญญาณที่รับจากอุปกรณ์ภายนอกแล้วส่งไปยัง PLC ตามรูปภาพด้านบน อุปกรณ์ Input สามารถทำงานได้เพียงแค่เชื่อมต่อระหว่าง Input terminal กับ COM terminal อีกทั้ง หน้าสัมผัสของ Relay โดยทั่วไป ถึงแม้จะเป็นจำนวนมากแต่มีแค่ไม่กี่หน้าสัมผัสเท่านั้น แต่ภายในซีควเอนซ์โปรแกรม มีจำนวนหน้าสัมผัสอยู่นับไม่ถ้วน

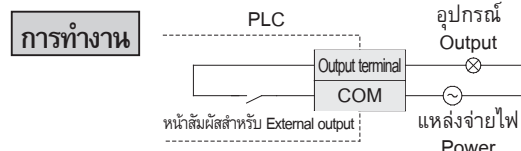


ภายใน PLC จะมี Power สำหรับ Input relay อยู่ เมื่อหน้าสัมผัส (Contact) ของอุปกรณ์ Input มีการนำกระแส ไฟจะทำการไหลเหมือนกับเส้นประ ซึ่งจะทำให้ Input relay สามารถขับเคลื่อนได้

ตามที่ได้อธิบายไปในเบื้องต้นระหว่าง PLC กับอุปกรณ์ Input นั้น ไม่สำคัญว่าต้องเป็นหน้าสัมผัสปกติเปิด a (N.O. Contact) หรือ หน้าสัมผัสปกติปิด b (N.C. Contact) เพราะว่าถ้ามีการนำกระแสเกิดขึ้นระหว่าง COM กับ Input terminal ซึ่งหน้าสัมผัสที่ปิดจะเป็นหน้าสัมผัสปกติเปิด a (N.O. Contact) และถ้าหน้าสัมผัสเปิดจะเป็นหน้าสัมผัสปกติปิด b (N.C. Contact)

### ● หน้าสัมผัสสำหรับ Output

หน้าสัมผัสสำหรับ Output นั้น คือหน้าสัมผัสสำหรับขับเคลื่อนโหลด (Load) จากภายนอก โดย Input relay ที่ขับเคลื่อนด้วยซีควเอนซ์โปรแกรม  
 หน้าสัมผัสสำหรับ External output นั้นสามารถเชื่อมต่อกับอุปกรณ์ที่มีกระแสไฟต่างกัน อย่างเช่น COM ที่มีหน่วยเป็น AC (กระแสสลับ) / DC (กระแสตรง)



เมื่อ Output Relay ที่อยู่ในซีควเอนซ์โปรแกรม (Sequence program) เกิดการขับเคลื่อน หน้าสัมผัสสำหรับ External output จะปิด  
 โดยที่กระแสไฟที่จะใช้ขับเคลื่อนอุปกรณ์ Output นั้น ต้องจัดเตรียมจากภายนอกของ PLC

อุปกรณ์ Input-Output ที่ต่อเข้ากับ Input และ Output terminal นั้น มีการแบ่งอุปกรณ์ (Device) ในแต่ละขั้วต่อ (Terminal) (หรืออาจจะเป็นหมายเลข In-Output) เพื่อเทียบ Input relay กับ Output relay ของซีควเอนซ์โปรแกรม (Sequence program) ภายในอุปกรณ์ (Device) จะมีหมายเลขในแต่ละขั้วต่อ (Terminal) นอกเหนือจากนั้นภายใน Sequence จะประกอบไปด้วย Timer และ Counter

- อุปกรณ์ (Device) ... ภายในอุปกรณ์ (Device) จะมีสัญลักษณ์ของอุปกรณ์ (Device) เพื่อการแสดงหน้าที่ของอุปกรณ์ (Device) และประกอบไปด้วยหมายเลขอุปกรณ์ (Device) ที่มีโครงสร้างแยกเป็นอย่างละตัว (ซึ่งบางที่เราจะเรียกอุปกรณ์ (Device) ว่า Element number)

**Input relay** : X000 ~ X177 (128 จุด) \* 1

มีหน้าที่เป็น Window เพื่อเปิดรับสัญญาณจาก Input switch ที่อยู่ภายนอกของ Sequence จะแทนสัญลักษณ์อุปกรณ์ (Device) เป็น X ภายในประกอบไปด้วย Input relay ที่ตอบสนองจำนวนของ Input (จำนวนขั้วต่อ Terminal)

**Output relay** : Y000 ~ Y177 (128 จุด) \* 1

มีหน้าที่เป็น Window เพื่อขับเคลื่อน Load ที่อยู่ภายนอกของ PLC จะแทนสัญลักษณ์อุปกรณ์ (Device) เป็น Y ภายในประกอบไปด้วย หน้าสัมผัส (Contact) สำหรับ Output ที่ตอบสนองจำนวนของ Output (จำนวนขั้วต่อ Terminal)

**Auxiliary relay** : M0 ~ M7679 (7680 จุด) \* 1

เป็น Auxiliary relay รีเลย์เสริมที่ประกอบอยู่ใน PLC (บางที่เรียกว่ารีเลย์ภายใน หรือ Inside relay)

- Input relay, Output relay, Auxiliary relay, Timer, Counter จำนวนของอุปกรณ์เหล่านี้ ที่สามารถใช้ได้นั้นต่างกันไปขึ้นอยู่กับประเภทของ PLC

\*1 : เป็นขอบเขตการทำงานของอุปกรณ์ (Device) รวมถึงจำนวนจุดของ PLC รุ่น FX3G Series

**Timer** : T0 ~ T319 (320 จุด) \* 1

Timer เป็นอุปกรณ์ที่อยู่ภายใน Sequence มีหน้าที่ในการจับเวลา และประกอบไปด้วย Coil กับ หน้าสัมผัส (Contact) เมื่อถึงเวลาที่กำหนด หน้าสัมผัส (Contact) จะทำการปิด

**Counter** : C0 ~ C199 (200 จุด) \* 1

Counter เป็นอุปกรณ์ที่อยู่ภายใน PLC มีหน้าที่ในการนับจำนวน เมื่อถึงจำนวนที่กำหนดหน้าสัมผัส (Contact) จะปิด

## อ้างอิง

### เลขฐาน 10, เลขฐาน 8, เลขฐาน 16

อุปกรณ์ (Device) จะมีทั้งเลขฐาน 10 หรือนอกเหนือจากนั้นจะมีเลขฐาน 8 หรือ เลขฐาน 16 ตามตารางอีกด้วย

	Input relay, Output relay	Auxiliary relay, Timers, Counters
Micro PLC FX Series	เลขฐาน 8	เลขฐาน 10
General PLC Q/QnA/A Series	เลขฐาน 16	เลขฐาน 10

**เลขฐาน 10 คืออะไร** โดยทั่วไปจะใช้ 0 ~ 9, 10 ~ 19, 20 ~ 29, ... จะเป็นวิธีการคำนวณที่มีการปัดหน่วยที่เป็น 10 ขึ้น

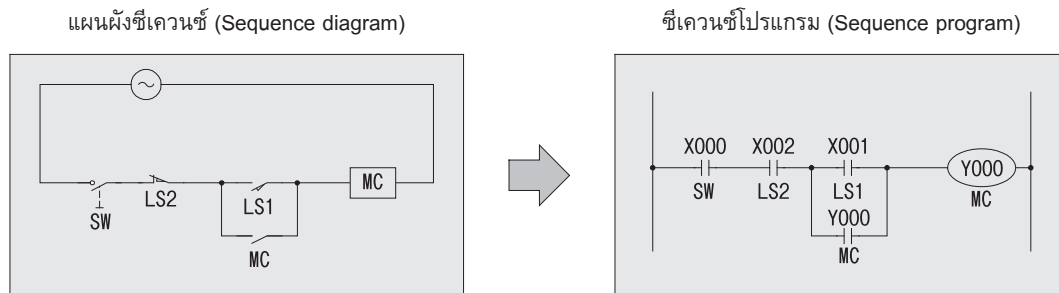
**เลขฐาน 8 คืออะไร** 0 ~ 7, 10 ~ 17, 20 ~ 27, ... จะเป็นวิธีการคำนวณที่มีการปัดหน่วยที่เป็น 8 ขึ้น

**เลขฐาน 16 คืออะไร** 0 ~ 9, 0A, 0B, 0C, 0D, 0E, 0F, 10 ~ 19, 1A, 1B, 1C, 1D, 1E, 1F, ... จะเป็นวิธีการคำนวณที่มีการปัดหน่วยที่เป็น 16 ขึ้น

## 2.2.3 แผนผังซีเควนซ์ (Sequence diagram) และซีเควนซ์โปรแกรม (Sequence program)

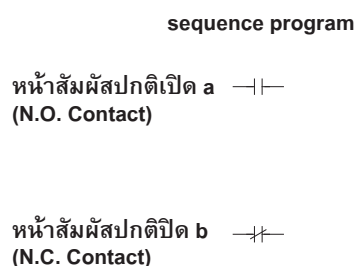
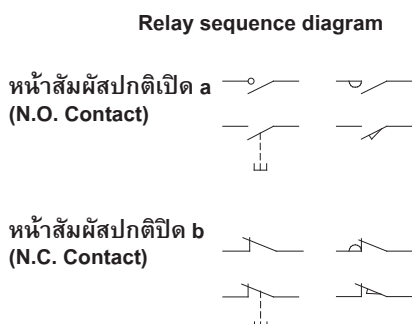
ซีเควนซ์โปรแกรม (Sequence program) ใน PLC นั้น มีอุปกรณ์ Input ที่มีสายต่อจากภายนอกเข้าไปในแต่ละขั้วต่อ (Terminal) และมีอุปกรณ์ Output ที่เปรียบเสมือนวงจรไฟฟ้าเป็นการควบคุมซีเควนซ์ (Sequence Control) ..... สำหรับรายละเอียดที่เกี่ยวกับคำสั่งตรวจสอบได้ในบทที่ 4

ต่อไปจะเป็นการอธิบายเกี่ยวกับแผนผังซีเควนซ์ (Sequence diagram) และซีเควนซ์โปรแกรม (Sequence program) ตัวอย่างหน้า 1-13 เมื่อมองการควบคุมระดับน้ำในแทงก์เป็นซีเควนซ์โปรแกรม (Sequence program)



พอมีการมองสลับกันแล้ว มีจุดไหนที่แตกต่างกันบ้างจะขออธิบายดังต่อไปนี้

- ① วิธีการแสดง หน้าสัมผัสปกติเปิด a (N.O. Contact) หน้าสัมผัสปกติปิด b (N.C. Contact)



- ② ไม่สามารถแสดงออกเป็นแบบวงจรไฟฟ้าได้
- ③ สำหรับซีเควนซ์โปรแกรม (Sequence program) จะอธิบายโดยใช้อุปกรณ์ (Device) (Element number) ตามที่อธิบายไว้ในก่อนหน้า

### ข้อสำคัญ

- ④ ตามแผนผังซีเควนซ์ (Sequence diagram) แล้ว LS2 ก็คือ หน้าสัมผัสปกติปิด b (N.C. Contact) และซีเควนซ์โปรแกรม (Sequence program) คือ หน้าสัมผัสปกติเปิด a (N.O. Contact)

เริ่มแรกจากแผนผังซีเควนซ์ (Sequence diagram) ของ LS2 ด้านบนมีหน้าที่ตัดวงจรสภาพตัวเอง Self-hold circuit สำหรับ MC ซึ่งเป็นอุปกรณ์ที่ใช้เพื่อการหยุด MC โดยในปกติแล้วจะอยู่ในสถานะนำกระแสอยู่ในวงจรไฟฟ้าอยู่ตลอดเวลา

## อ้างอิง

### การทำงานของ “หน้าสัมผัสปกติเปิด a (N.O. Contact)” “หน้าสัมผัสปกติปิด b (N.C. Contact)” ในซีควেনซ์โปรแกรม (Sequence program)

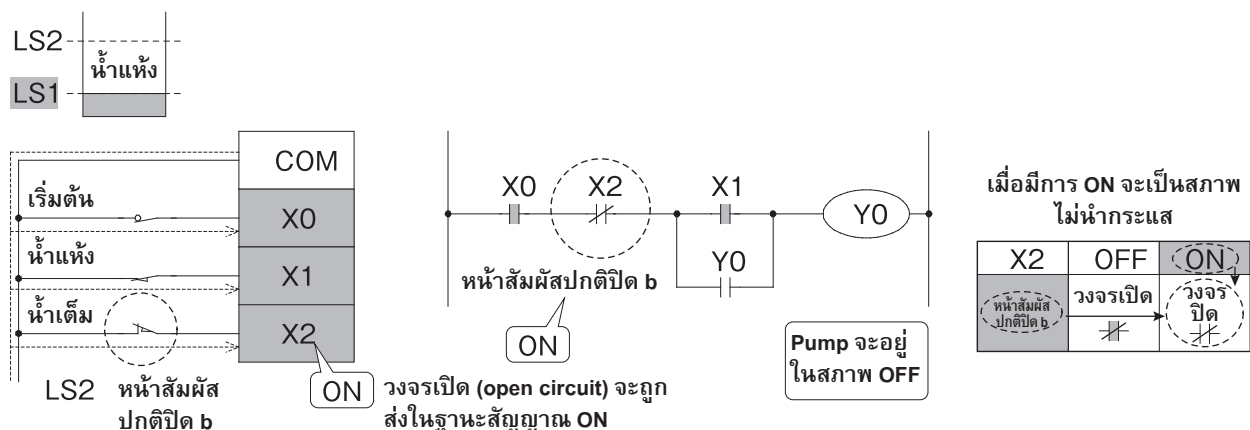
การทำงานของ “หน้าสัมผัสปกติเปิด a (N.O. Contact)” “หน้าสัมผัสปกติปิด b (N.C. Contact)” ผ่าน Sequen program นั้น ถ้า Input relay ทำงานเหมือนกับหน้า 2-4 จะเกิดเงื่อนไขขึ้นตามเนื้อหาด้านล่าง

- เมื่ออุปกรณ์ Input มี “สถานะไม่มีไฟฟ้าไหลผ่าน” เข้าไปใน PLC ก็จะทำให้ “วงจรภายในโปรแกรมเปิด” ก็จะเป็น “หน้าสัมผัสปกติเปิด a (N.O. Contact)” และในทางกลับกันจะกลายเป็น “หน้าสัมผัสปกติปิด b (N.C. Contact)” เมื่อ “วงจรภายในโปรแกรมเปิด”
- เมื่ออุปกรณ์ Input มี “สถานะมีไฟฟ้าไหลผ่าน” เข้าไปใน PLC ก็จะทำให้ “วงจรภายในโปรแกรมเปิด” ก็จะเป็น “หน้าสัมผัสปกติเปิด a (N.O. Contact)” และในทางกลับกันจะกลายเป็น “หน้าสัมผัสปกติปิด b (N.C. Contact)” เมื่อ “วงจรภายในโปรแกรมเปิด”

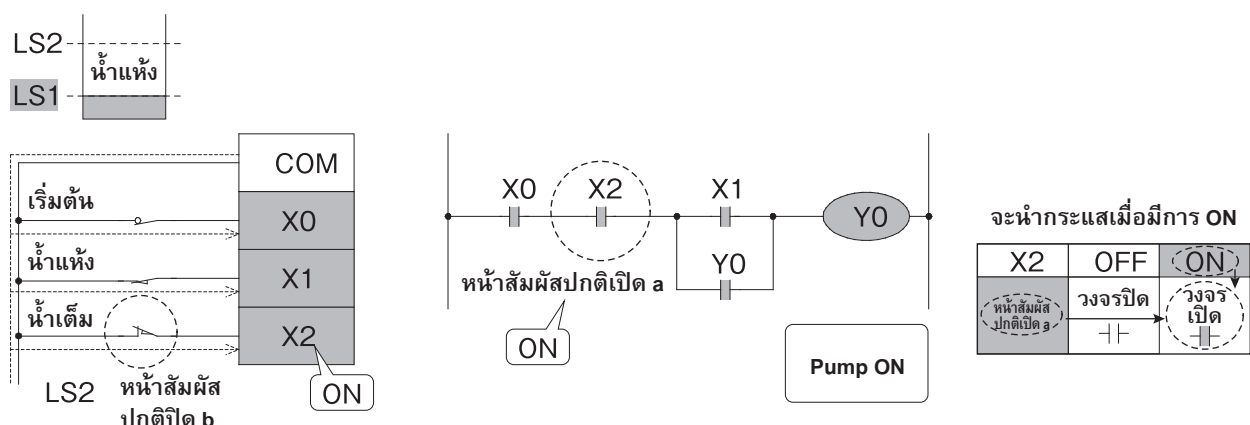
เพราะฉะนั้นเพื่อให้สถานะในการทำงานเหมือนกันกับ Sequence diagram ในหน้า 1-14 นั้น จำเป็นต้องกำหนดสัญญาณของ LS2 ที่อยู่ใน Sequencer program ให้เป็น “หน้าสัมผัสปกติเปิด a (N.O. Contact)”  
รายละเอียดให้อ้างอิงจากรูปภาพ (1) ~ (2) ในด้านล่าง

#### (1) สาเหตุที่ใช้ LS2 (X2) เป็นหน้าสัมผัสปกติเปิด a (N.O. Contact)

① ในกรณีที่กำหนดจุดหน้าสัมผัส (Contact) ให้เป็นหน้าสัมผัสปกติปิด b (N.C. Contact) เหมือนกันกับ Sequence diagram



② ในกรณีที่มีการกำหนดจุดหน้าสัมผัส (Contact) ให้เป็นหน้าสัมผัสปกติเปิด a (N.O. Contact)

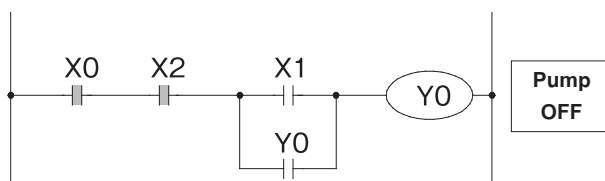
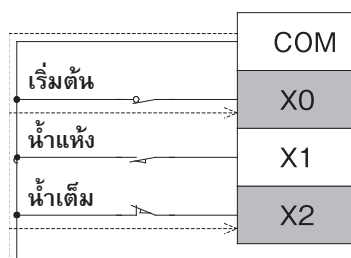
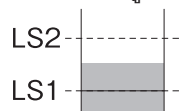


#### ● แนวคิดในเรื่องโปรแกรม

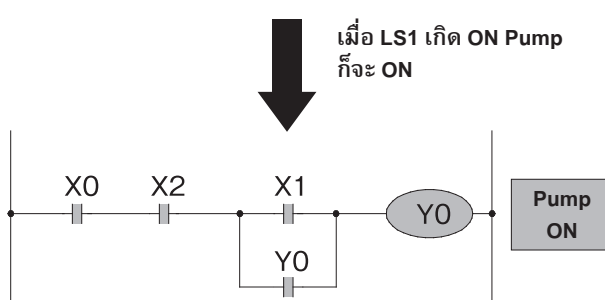
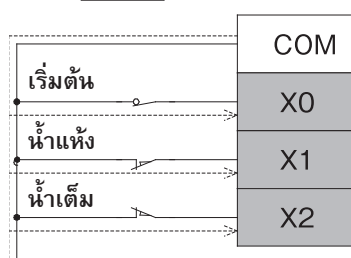
เมื่อมีสัญญาณจากภายนอกเข้ามาที่ Input ... หน้าสัมผัสปกติเปิด a (N.O. Contact) จะเป็นตัวนำกระแสหน้าสัมผัสปกติปิด b (N.C. Contact) จะเป็นตัวไม่นำกระแส

## (2) Flow การควบคุมระดับน้ำใน Tank

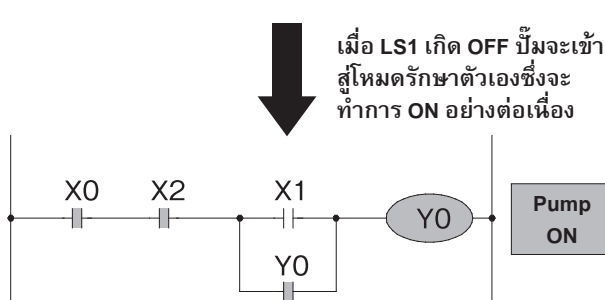
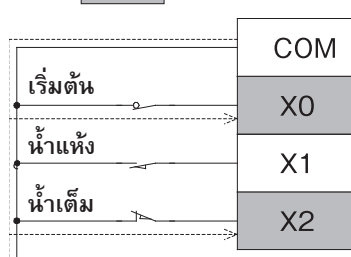
- เมื่อระดับน้ำอยู่ตรงกลางปั๊มจะ OFF



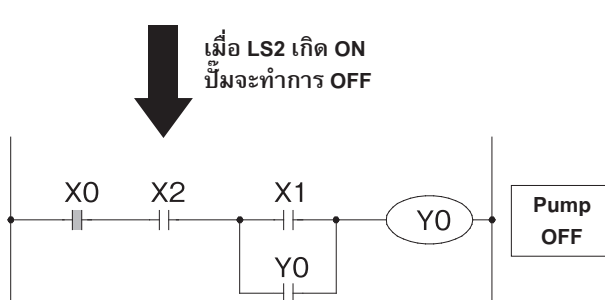
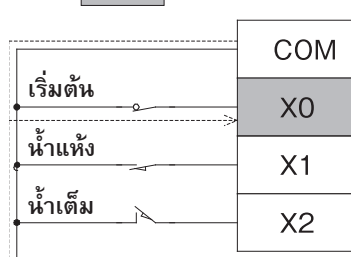
- เมื่อระดับน้ำใน Tank อยู่ในระดับแห้ง



- เมื่อปั๊ม ON แสดงว่าน้ำกำลังเต็ม



- เมื่อระดับน้ำใน Tank เต็ม

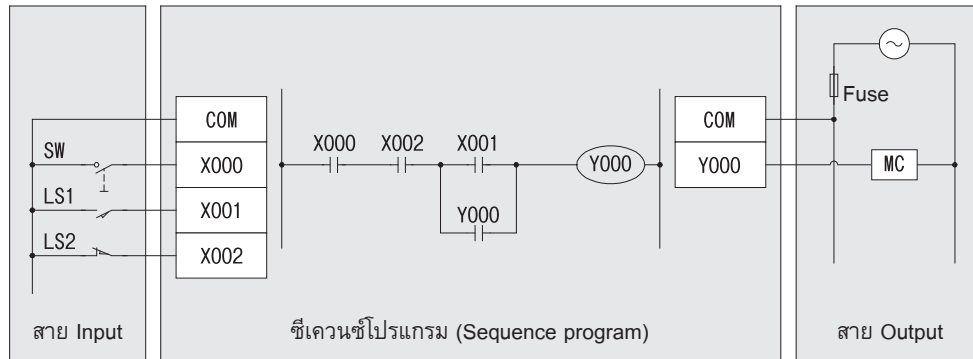




## 2.3 การเดินสายไฟและโปรแกรม

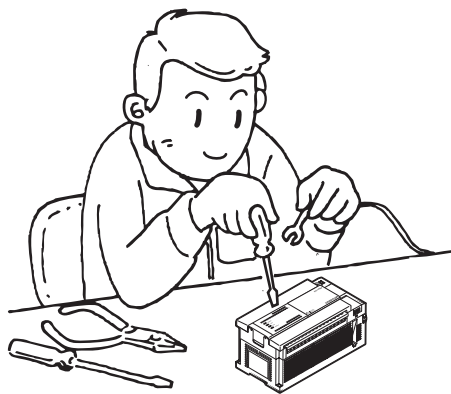
### 2.3.1 การเดินสายไฟของ PLC และโปรแกรมมีลักษณะอย่างไร

รูปด้านล่างระบุให้เห็นรายละเอียดของแผนผังวงจรไฟฟ้าในหน้า 1-14

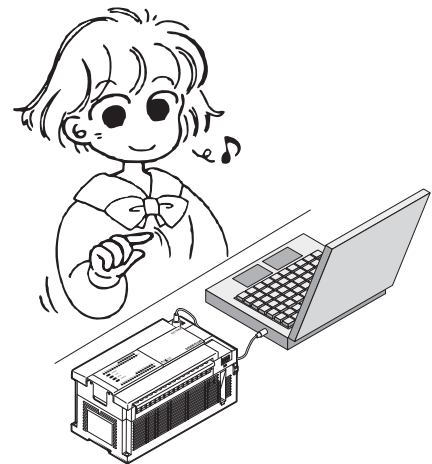


สายไฟสำหรับ PLC สามารถทำงานแยกระหว่างสาย Input-Output กับสายไฟภายในได้

สำหรับสาย Input-Output จำเป็นต้องใช้ประแจหรือไขควงในการปฏิบัติงานเหมือนเดิม



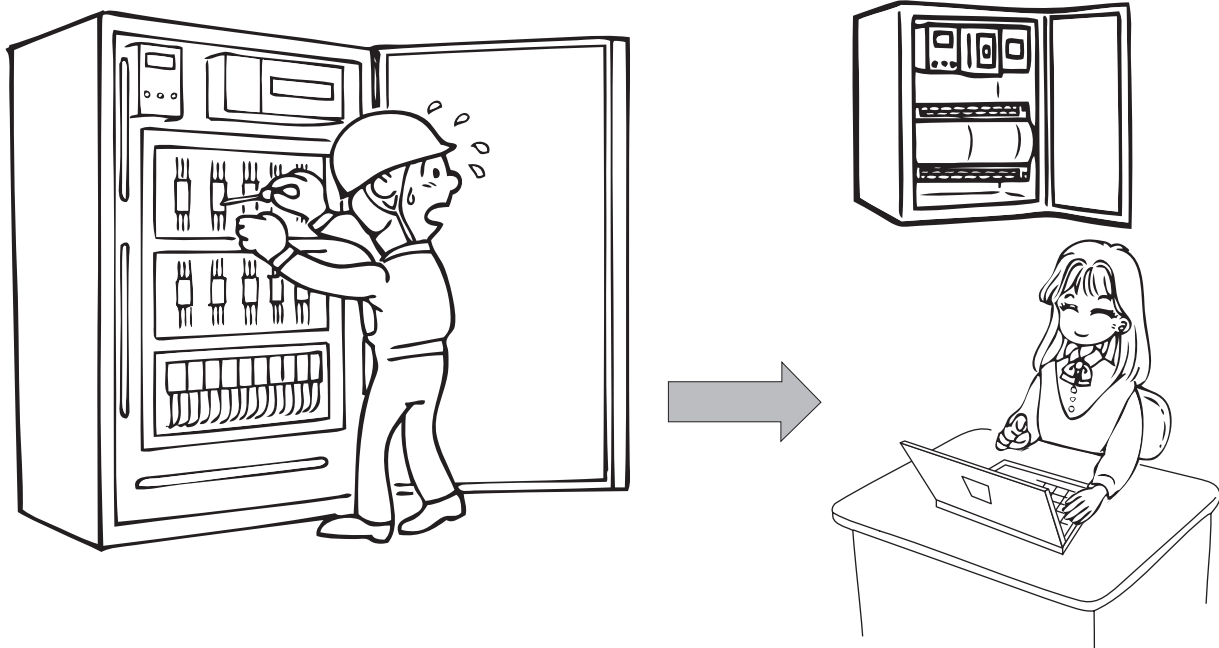
สำหรับสายไฟภายในที่มีความซับซ้อนนั้น (Sequence program) สามารถใช้ Control keyboard สำหรับ PC หรือ Programming panel ในการควบคุมได้อย่างสะดวกง่ายดาย



อีกทั้งการเชื่อมโยงระหว่าง Input terminal กับ Input relay coil หรือการเชื่อมโยงระหว่าง External contact output กับ Output terminal นั้นจะสามารถใช้งานได้ตั้งแต่มีการผลิตที่โรงงาน

## 2.4 ข้อดีของการใช้งาน PLC

### 2.4.1 ประโยชน์ที่จะได้รับจากการใช้งาน PLC



1

#### ประหยัดค่าใช้จ่าย

สำหรับตู้ควบคุม (Control panel) ที่มีการใช้ Relay หรือ Timer มากกว่า 10 ตัว การแทนที่ด้วยการใช้ PLC จะถือว่าได้เปรียบกว่าในด้านเศรษฐกิจ

2

#### ช่วยทุ่นแรงในด้านการออกแบบ

ช่วยทำให้ขั้นตอนการออกแบบซีเคานซ์และการออกแบบการเดินสายไฟสำหรับอุปกรณ์ต่างๆ ทำได้ง่ายขึ้น อีกทั้งง่ายและสะดวกในการทำทดสอบการทำงานของเครื่อง ช่วยลดขั้นตอนในการออกแบบกว่าในอดีต

3

#### ลดขั้นตอนการทำงาน

ลดขั้นตอนจากการต้องเตรียมชิ้นส่วนประกอบ สามารถเตรียมเครื่องจักรพร้อมกับการเตรียมตู้ควบคุมได้ในเวลาเดียวกัน และมีความยืดหยุ่นในการเปลี่ยนรายละเอียดงาน ช่วยทำให้การเดินสายไฟทำได้ง่ายขึ้น จึงทำให้สามารถลดขั้นตอนในการทำงานได้หลายขั้นตอน

4

#### ขนาดกะทัดรัดและเป็นมาตรฐานเดียวกัน

เมื่อเทียบกับการติดตั้งแผงรีเลย์แล้วถือว่ามีความกะทัดรัดกว่า อีกทั้งยังสามารถสร้างมาตรฐานด้วยการนำเอาโปรแกรมที่เขียนขึ้นกลับมาใช้ใหม่ได้เรื่อยๆ

5

#### เพิ่มความเชื่อมั่น

ลดปัญหาที่เกิดจาก Relay หรือ Timer เพียงแค่มีการตรวจเช็คในครั้งแรก แล้วจากนั้นก็ใช้งานได้อย่างมั่นใจ

6

#### พัฒนาความสามารถด้านการบำรุงรักษา

บำรุงรักษาง่ายและรวดเร็วด้วยฟังก์ชันการตรวจสอบความเสียหาย หรือตรวจสอบอายุการใช้งานของชิ้นส่วนที่มีระยะเวลาการใช้งานสั้นๆ ได้ (Self-diagnostic functions)

## 2.4.2 เปรียบเทียบกับการควบคุมรีเลย์ (Relay control)

หัวข้อ \ วิธีการ		การควบคุมรีเลย์ (Relay control)	การควบคุม PLC (PLC control)
1	ความสามารถ (Function)	ถ้าใช้ Relay หลายตัวก็สามารถควบคุมแบบการทำงานที่ซับซ้อนได้	สามารถควบคุมการทำงานที่ซับซ้อนได้มากมายด้วยโปรแกรม
2	การเปลี่ยนแปลงเนื้อหาการควบคุม	ไม่มีวิธีอื่นนอกเหนือจากการเปลี่ยนการเดินสายไฟ	สามารถทำได้โดยอิสระ เพียงแค่เปลี่ยนโปรแกรม
3	ความเชื่อมั่น	ถ้ามีการใช้งานตามปกติจะไม่มีปัญหา แต่จะมีจำกัดในเรื่องของหน้าสัมผัสไม่ดีและเรื่องของอายุการใช้งาน	ชิ้นส่วนที่สำคัญใช้วัสดุทั้งหมดเป็นกึ่งตัวนำไฟฟ้า จึงได้รับความไว้วางใจสูง
4	คุณสมบัติโดยทั่วไป	อุปกรณ์ที่ทำเสร็จนั้นไม่สามารถนำไปใช้อย่างอื่นได้	ขึ้นอยู่กับโปรแกรมสามารถควบคุมประเภทไหนก็ได้
5	การขยายเครื่องจักร (Expandability)	จำเป็นต้องมีการเพิ่มหรือมีการสร้างใหม่ซึ่งมีความยากลำบาก	สามารถขยายได้ตามอิสระจนเต็มขีดความสามารถ
6	ความง่ายง่ายในการดูแลรักษา	จำเป็นต้องมีการตรวจสอบเป็นระยะ และเปลี่ยนชิ้นงานตามอายุการใช้งาน	สามารถทำการซ่อมบำรุงได้เพียงแค่การเปลี่ยนยูนิท
7	ความหลากหลายด้านฟังก์ชัน	ทำได้เฉพาะการควบคุมรีเลย์ (Relay control)	นอกเหนือจาก Sequence program แล้ว ยังสามารถควบคุมอื่นๆ เช่น อนาล็อก (Analog) หรือ การกำหนดตำแหน่ง (Positioning) ได้อีกด้วย
8	ขนาดเครื่องจักร	มีขนาดใหญ่ทั่วๆ ไป	ถึงแม้จะเป็นการควบคุมระดับสูงและซับซ้อน ก็ไม่ทำให้เครื่องจักรมีขนาดใหญ่ขึ้นกว่าเดิม
9	ระยะเวลาในการออกแบบและการจัดทำ	ใช้เวลานาน ไม่ว่าจะเป็นการจัดทำ Drawing จำนวนมาก การจัดหาชิ้นงานและการทดสอบประกอบ	ถึงจะเป็นการควบคุมที่ซับซ้อน ก็สามารถทำการออกแบบได้อย่างง่ายดาย ไม่สร้างขั้นตอนในการจัดทำ

อ้างอิง
<p><b>ความเป็นมาของ PLC อย่างย่อ</b></p> <p>เริ่มจากการที่บริษัท GM (General Motors) ในอเมริกามีความประสงค์ที่จะพัฒนาค้นคว้าเกี่ยวกับ PLC จึงทำให้ PLC ได้ถูกคิดค้นขึ้นในปี 1968 และในปี 1969 อเมริกาได้เริ่มวางจำหน่ายในตลาด สำหรับในประเทศญี่ปุ่นได้ให้กำเนิดรุ่นแรกในประเทศขึ้นในปี 1970 แต่สำหรับรุ่นทั่วไปนั้นจะเกิดขึ้นหลังจากปี 1976 บริษัท Mitsubishi Electric เป็นผู้เปิดตลาดโดยการวางจำหน่ายรุ่นทั่วไปตั้งแต่ปี 1977 โดยผลิตภัณฑ์เป็นแบบ One-board จนติดตลาดและเป็นที่รู้จักกันทั่วไป หลังจากนั้นในปี 1980 ได้มีการวางจำหน่ายรุ่น K-Series ที่มีการติดตั้งฟังก์ชันการคำนวณและจัดการเรื่องตัวเลข (Numeric value processing functions) และในปี 1981 ถือเป็นจุดเริ่มในการวางจำหน่าย Micro PLC F-Series ที่มีการติดตั้งโปรแกรม (Built-in programmer) และเข้าสู่ยุคที่เริ่มใช้ PLC อย่างแพร่หลายในปัจจุบัน</p>

# MEMO

## บทที่ 3

### การใช้งานโปรแกรม GX Works 2

ขอแค่ใช้คอมพิวเตอร์เป็น ซีเควนซ์ (Sequence) ก็เป็นเรื่องง่าย ๆ...

ในการจะสร้างหรือแก้ไขซีเควนซ์โปรแกรม (Sequence program) สามารถดำเนินการได้อย่างง่ายดาย เหมือนการวาดรูปง่าย ๆ ด้วยซอฟต์แวร์สำหรับติดตั้งในคอมพิวเตอร์ GX Works 2 เพียงแค่เรียนรู้พื้นฐานการใช้งานเบื้องต้น ที่เหลือก็แค่การฝึกให้เกิดความชำนาญ เป็นซอฟต์แวร์ (Software) ที่มีฟังก์ชันที่ใช้งานง่ายและหลากหลาย ก่อนอื่นเรามารู้และทำความรู้จักกับการใช้งานที่จำเป็นตามลำดับเพื่อให้เกิดความเชี่ยวชาญ และชำนาญในการใช้งานกันเถอะ

การเรียกใช้งานและการปรับปรุงแก้ไขก็สามารถดำเนินการได้อย่างราบรื่น ...

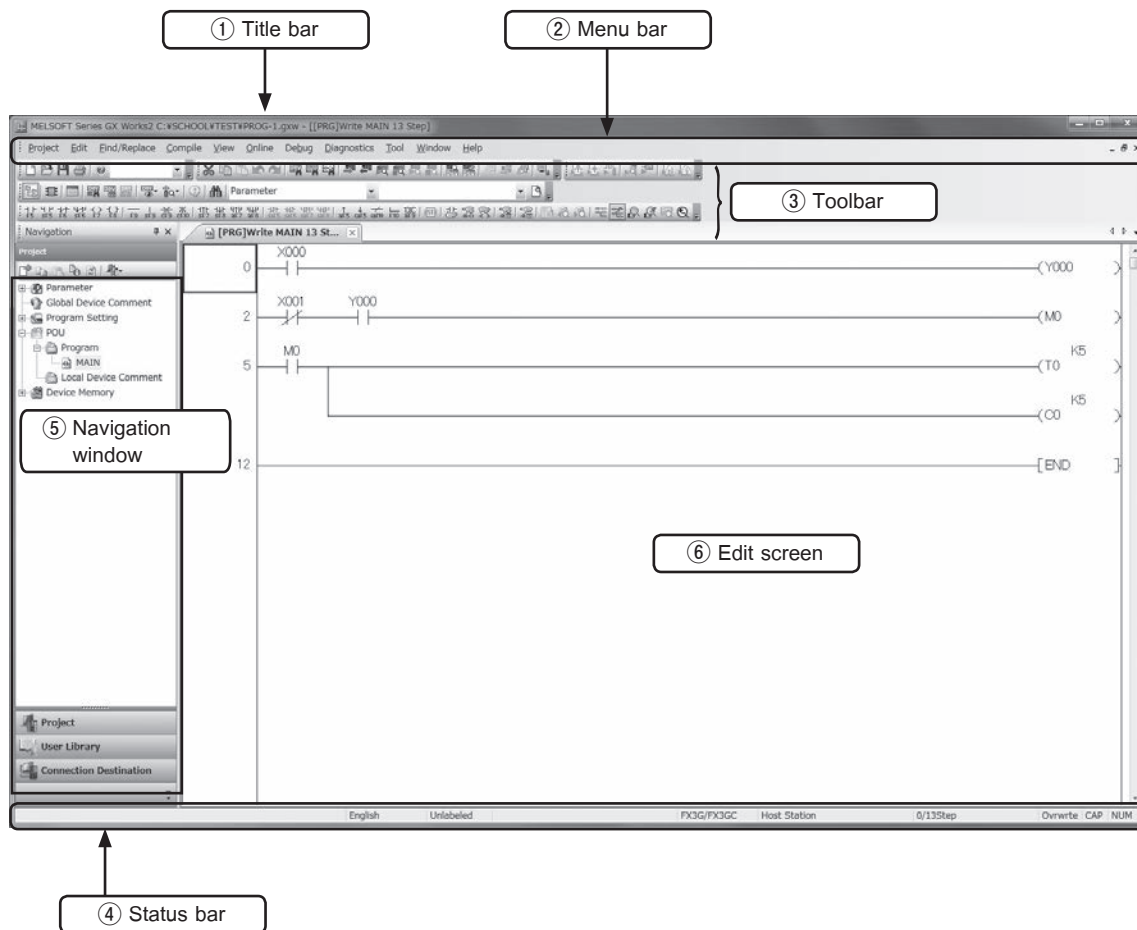
การสร้างโปรแกรม (Program) กับการแก้จุดบกพร่อง (Debug) เป็นของคู่กัน เนื่องจากสามารถทำการแสดงผล (Monitoring) สภาพการทำงานของ Program และ PLC ได้จากจอคอมพิวเตอร์จึงสะดวกในการตรวจสอบการทำงานและแก้ไขให้เป็นไปตามที่ต้องการได้อย่างรวดเร็ว

โปรแกรม (Program) ดูกง่ายไม่ซับซ้อน ...

เพื่อให้สามารถดูซีเควนซ์โปรแกรม (Sequence program) ได้ง่ายขึ้น ภายใน GX Works 2 จึงมีฟังก์ชัน “Comment input function” เพียงแค่กรอก Comment เอาไว้ก็จะทำให้เข้าใจโปรแกรมได้ง่ายขึ้นและช่วยเพิ่มประสิทธิภาพในการจัดทำซีเควนซ์โปรแกรม (Sequence program) และ Debug ได้

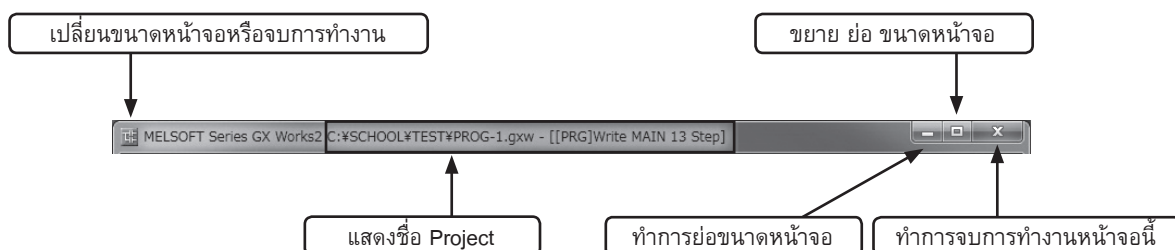
## 3.1 ความรู้พื้นฐานเพื่อการใช้งานโปรแกรม

### 3.1.1 โครงสร้างหน้าจอ

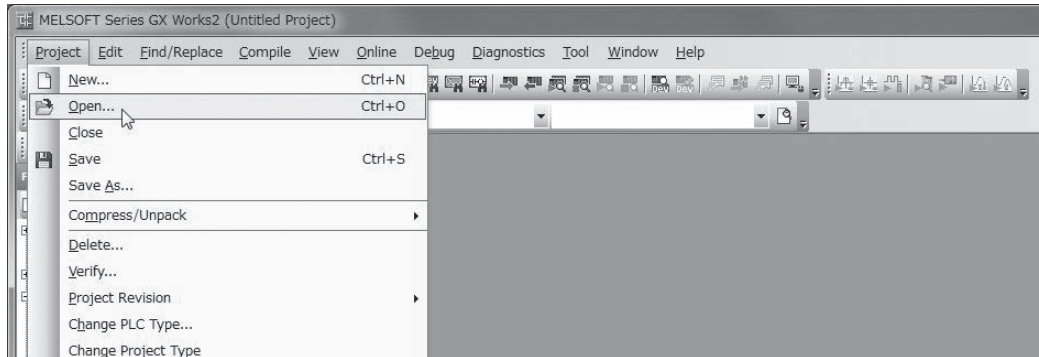


#### ① Title bar

จะแสดงชื่อ Project ที่กำลังเปิดทำงานอยู่ และ Icon การทำงานของ Windows

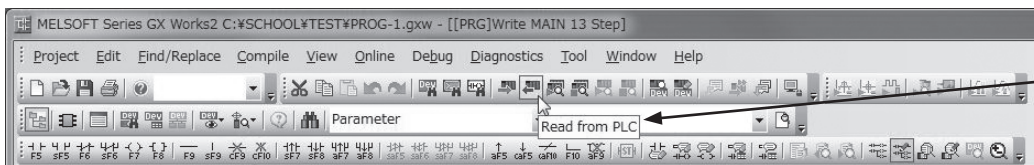


## ② Menu bar



กดเลือกเมนู แล้วจะมีหน้าจอเมนูแสดงขึ้นมาด้านล่าง

## ③ Toolbar\*



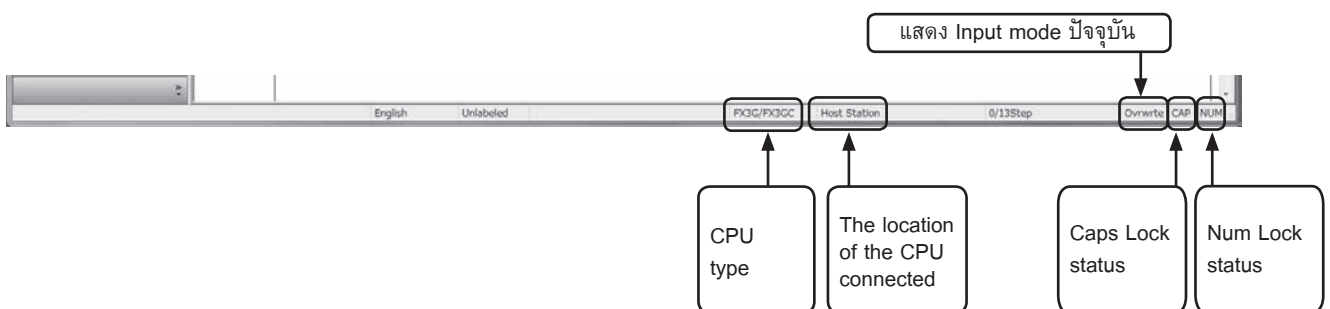
เมื่อนำเมาส์ไปวาง  
จะปรากฏข้อความ  
อธิบายรายละเอียด  
ของฟังก์ชันนั้นๆ

\* : รายละเอียดของ Toolbar เนื่องจากสามารถทำการเคลื่อนย้ายหรือเอาออกได้ ทำให้หัวข้อที่แสดงและการจัดวางแตกต่างกันไป

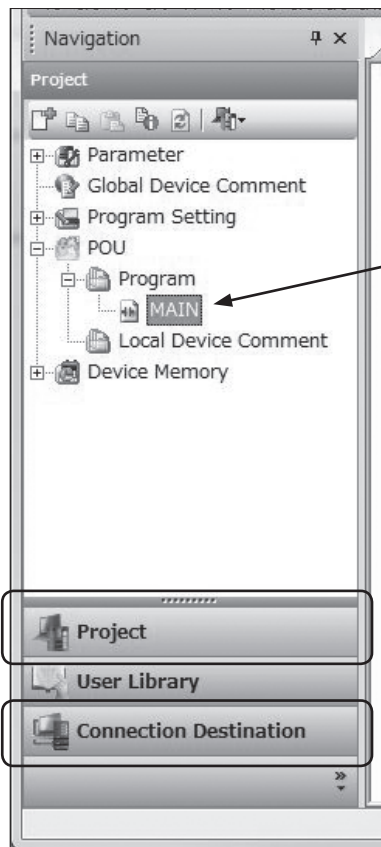
หัวข้อไหนที่มีการใช้งานบ่อยๆ จะเอามาวางเป็น Icon button สามารถคลิกเลือกได้โดยตรง ซึ่งจะเหมือนการเลือกเครื่องมือจาก Menu bar

## ④ Status bar

ทำการแสดงสภาพการทำงานและการตั้งค่า



⑤ Navigation window

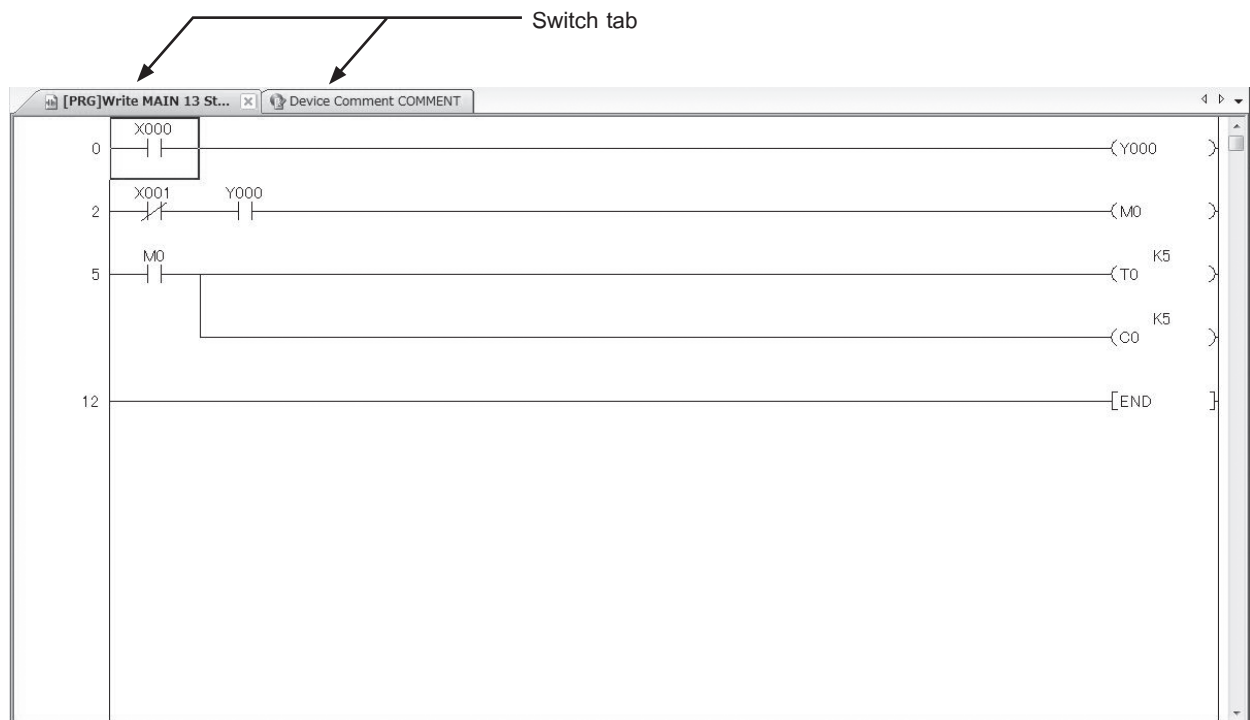


คลิก [POU] → [Program] →  
[MAIN] จะปรากฏแผนผังวงจร (Circuit diagram)  
ขึ้นมา

เมื่อใช้เมาส์คลิกจะปรากฏรายละเอียดของ Tools  
บน "Project"

เมื่อใช้เมาส์คลิกจะปรากฏรายละเอียดของเครื่อง  
ที่คอมพิวเตอร์ Connection Destination ที่ด้านบน  
(PLC Connection)

⑥ Edit screen





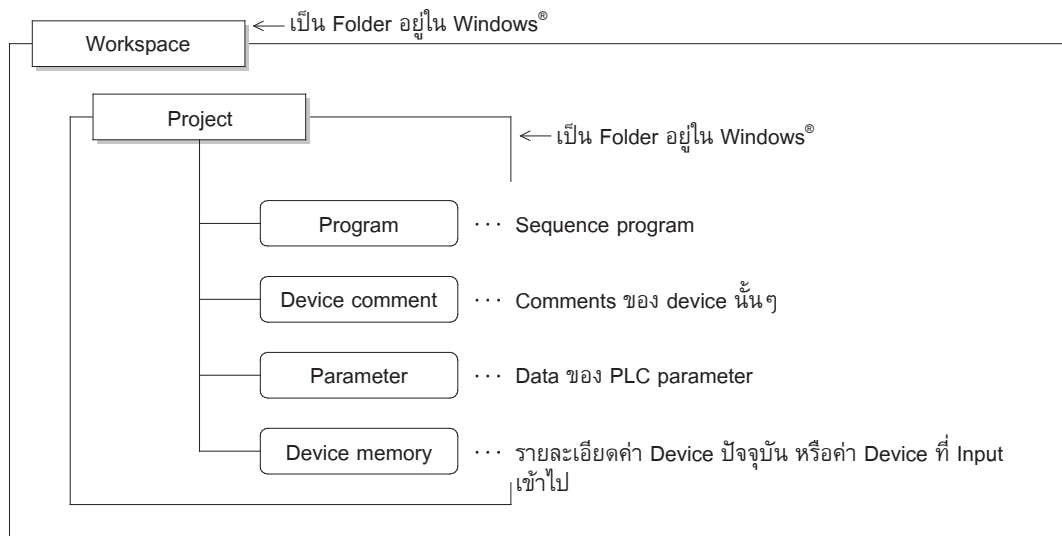
### 3.1.2 Workspace และ Project

- Workspace

Workspace คือตัวควบคุม Project หลายๆ ตัว ใน GX Works2 จะควบคุมให้มีชื่อ Program 1 ชื่อ ในการกำหนด Workspace ขอให้กำหนดเป็น เช่น Windows® Explorer ห้ามทำการเปลี่ยนแปลง

- “Project” ประกอบด้วย Program, Device comment, Parameter, Device memory

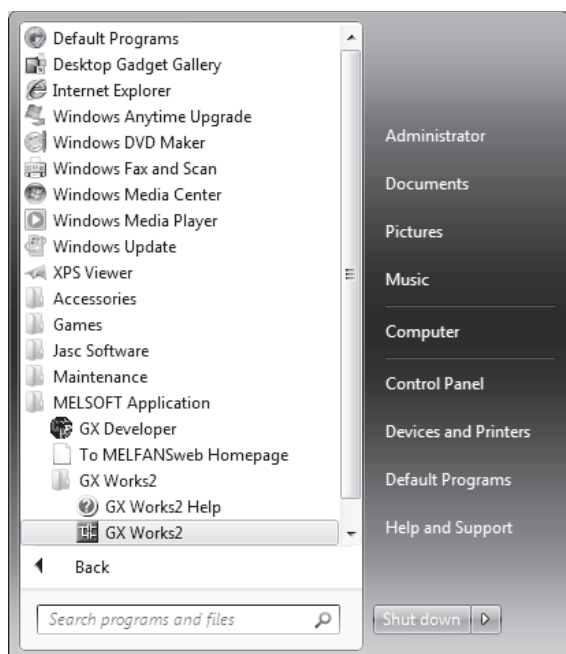
ใน GX Works2 ข้อมูลทั้งหมดนี้ถูกรวมไว้ด้วยกันเรียกว่า “Project” และถูก Save ลงใน Folder โดยมีชื่อ Workspace



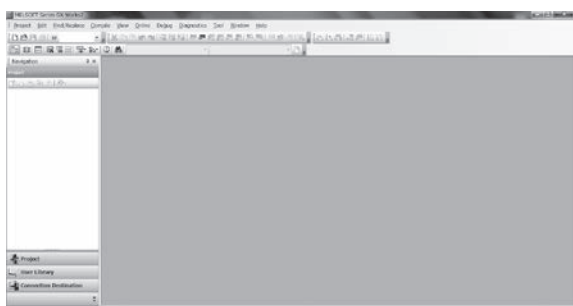
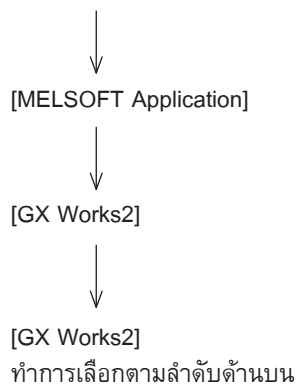
กรณีบันทึกแบบ Workspace

## 3.2 การเรียกใช้งานโปรแกรม (Starting GX Works2) และการสร้าง Project ใหม่ (New project)

### 3.2.1 เรียกใช้งานโปรแกรม (Starting GX Works2)

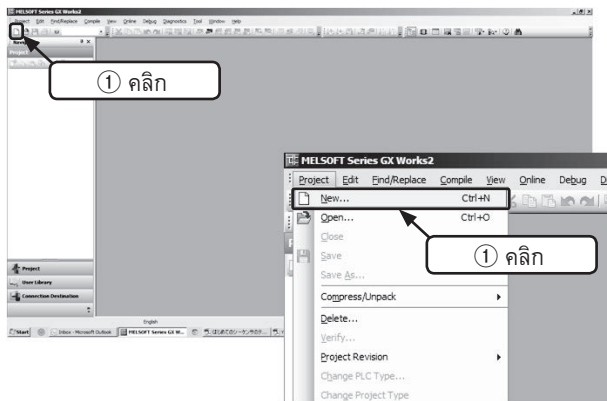



① เลือก [Start] ในด้านล่างซ้ายหน้าจอ Windows แล้วเลือกตามขั้นตอนด้านล่าง [Program]

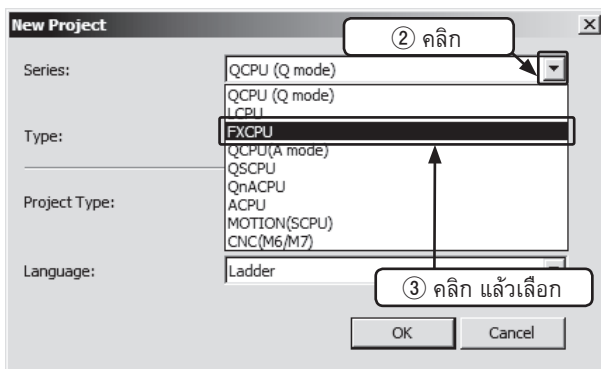


② Software เริ่มทำงาน

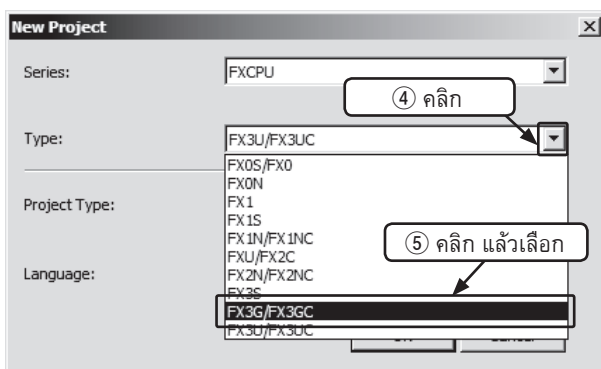
### 3.2.2 การสร้าง Project ใหม่ (New project)



- ① ทำการเลือก  จาก Toolbar หรือเลือก [Project] → [New Project] ( **Ctrl** + **N** ) จากเมนู

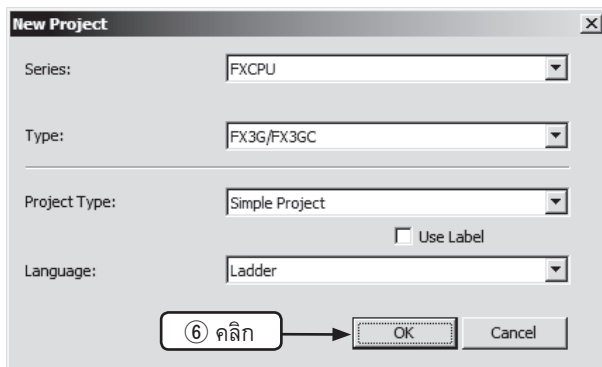


- ② คลิกปุ่ม [▼] ตรง "PLC Series"  
③ เลือก "FXCPU"



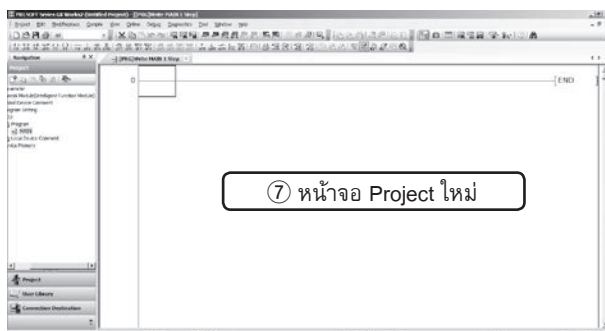
- ④ คลิกปุ่ม [▼] ตรง "PLC Type"  
⑤ เลือก "FX3G/FX3GC"





⑥ คลิก

(ระวัง) “Project type” ให้เลือก “Simple Project”  
 “Use Label” ตรวจสอบช่องว่างว่างอยู่หรือไม่  
 “Program Language” เลือกเป็น “Ladder”

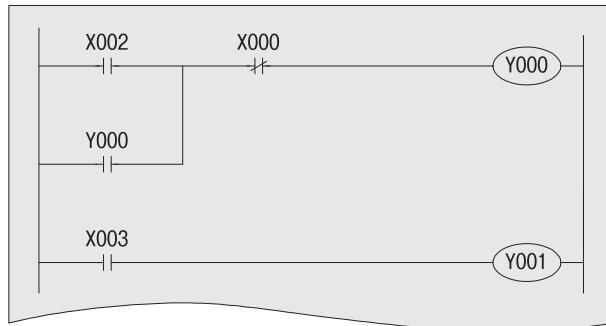


⑦ หน้าจอ Project ใหม่จะปรากฏขึ้นอยู่ในสภาพพร้อมใช้งาน  
 สามารถ Input program ได้

### 3.3 การสร้างโปรแกรมหรือวงจร (Circuit)

#### 3.3.1 การสร้างโปรแกรมหรือวงจร (Circuit) โดยใช้ฟังก์ชันคีย์ (Function keys)

[Circuit ที่สร้างขึ้น]



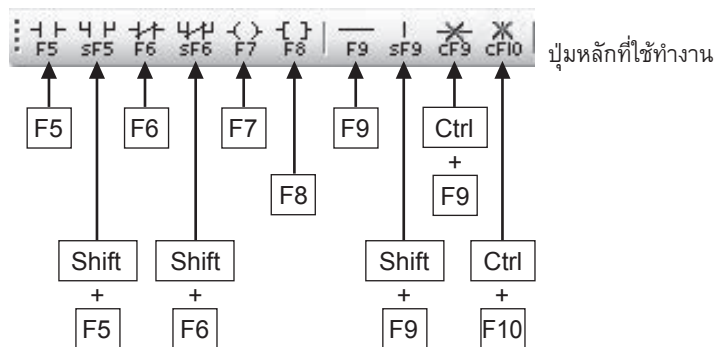
#### Point

ในคู่มือฉบับนี้จะแสดงหมายเลข Input relay (X), Output relay (Y) ด้วยตัวเลข 3 หลัก "X000", "Y000"

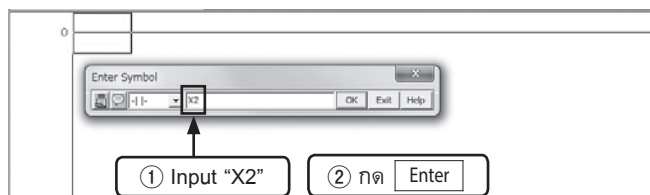
ในกรณีที่ Input จากคอมพิวเตอร์สามารถย่อได้ เป็น "X0", "Y0"

#### Point

- Function keys และสัญลักษณ์ Circuit จะปรากฏที่แผง Toolbar ด้านบนหน้าจอ



- เวลาใส่หน้าสัมผัสปกติเปิดเสมอ a (NO) และ Coil (C), (R), (S) เท่านั้น ที่สามารถย่อ function keys ได้
- ตัวหนังสือที่ใช้ในการ Input ทั้งหมดเป็นตัวอักษรภาษาอังกฤษพิมพ์เล็ก

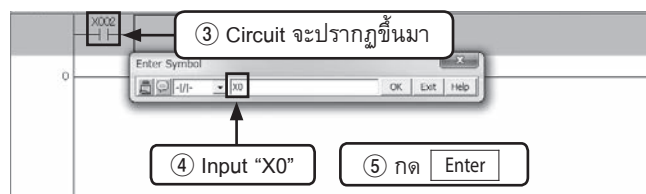


- ① กดที่ **F5** (⇧F5)  
แล้ว Input "X2"

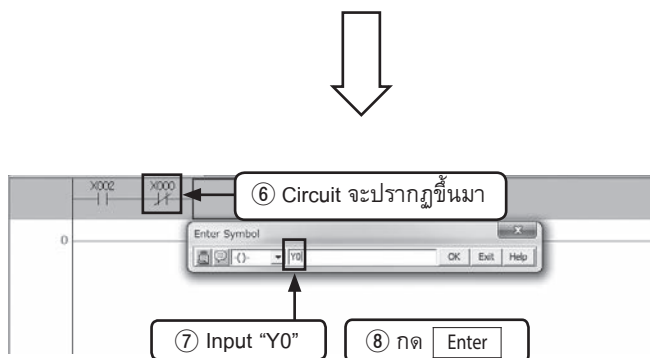


กด **ESC** หรือ **[Exit]**  
เพื่อทำการยกเลิก

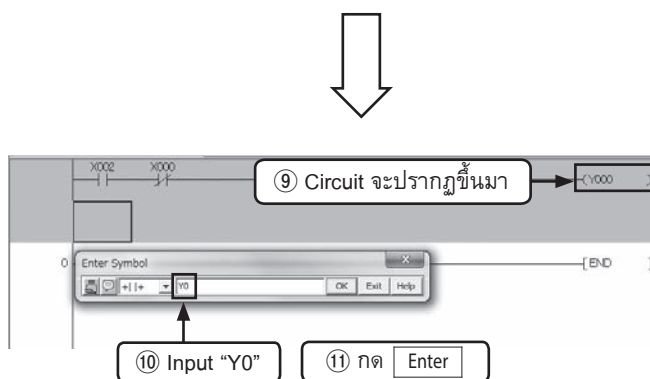
- ② ยืนยันโดยกด **Enter** หรือกด **[OK]**



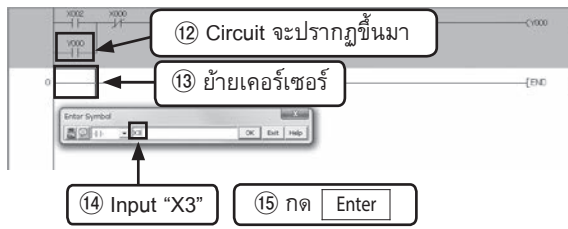
- ③ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา (⇧F5)
- ④ ทำการกด **F6** (⇧F6)  
แล้ว Input "X0"
- ⑤ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**



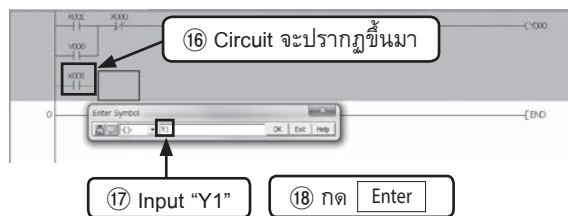
- ⑥ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา (⇧F6)
- ⑦ ทำการกด **F7** (⇧F7)  
แล้ว Input "Y0"
- ⑧ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**



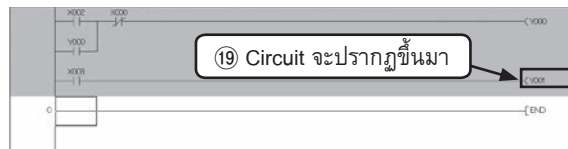
- ⑨ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา (⇧F7)
- ⑩ ทำการกด **Shift + F5** (⇧F5)  
แล้ว Input "Y0"
- ⑪ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**



- ⑫ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา (  $\neg$  )
- ⑬ เคลื่อนเคอร์เซอร์ไปไว้ที่หัวแถวของบรรทัดถัดไป
- ⑭ ทำการกด **F5** (  $\neg$  )  
แล้ว Input "X3"
- ⑮ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**



- ⑯ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา (  $\neg$  )
- ⑰ ทำการกด **F7** (  $\neg$  )  
แล้ว Input "Y1"
- ⑱ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**



- ⑲ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา (  $\neg$  )




**F4** (Build)



## ⑳ Compile Operation (สำคัญ)

ทำการกด "Compile" เพื่อทำการยืนยัน Circuit ที่ยังไม่ได้รับการยืนยัน (ส่วนที่แสดงเป็นสีเทา)

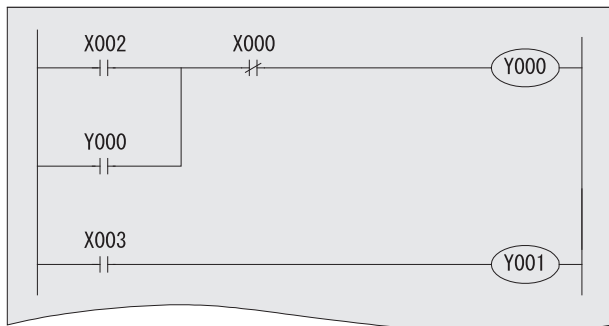
ขอให้เลือกดำเนินการตามขั้นตอนที่ระบุด้านล่าง

- กด **F4** (Build)
- กด  จาก Toolbar
- ทำการเลือก **[Conversion/Compile] → [Compile]** จากเมนู

ทำการยืนยัน Circuit เพื่อลบส่วนที่ปรากฏเป็นสีเทา  
กรณีที่เกิด Error เคอร์เซอร์จะเคลื่อนไปในตำแหน่งที่  
Circuit เป็น NG ขอให้ทำการแก้ไขให้ถูกต้อง

### 3.3.2 การสร้างวงจร (Circuit) โดยใช้ Toolbar buttons

[Circuit ที่จะสร้าง]

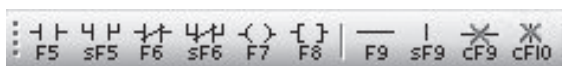


#### Point

ในคู่มือฉบับนี้จะแสดงสัญลักษณ์ Input relay (X) Output relay (Y) ด้วยตัวเลข 3 หลัก "X000", "Y000"  
ในกรณีที่ Input จากคอมพิวเตอร์สามารถย่อเป็น "X0", "Y0" ได้

#### Point

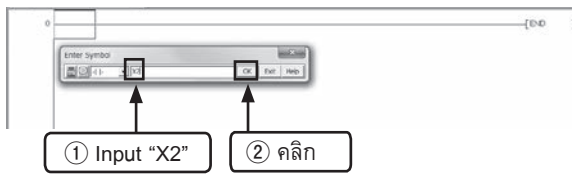
- ทำการคลิกเมาส์เลือกปุ่มเครื่องมือ และ Input สัญลักษณ์ของ Circuit ลงไป




ปุ่มหลักๆ ที่ใช้งาน

- ตัวอักษรสำหรับ Input เป็นตัวอักษรภาษาอังกฤษพิมพ์เล็ก



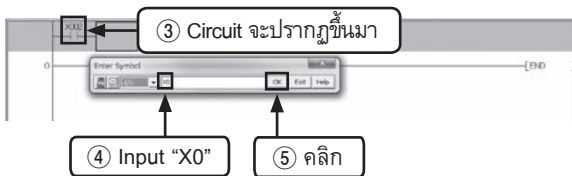


- ① คลิกเลือกที่  จาก Toolbar button แล้ว Input "X2"




กด **ESC** หรือ **[Exit]** เพื่อยกเลิก

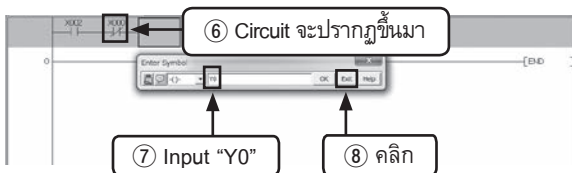
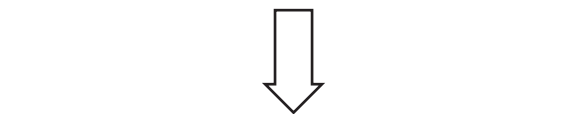
- ② ทำการยืนยันโดยกด **Enter** หรือ **[OK]**



- ③ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา ( $X_2$ )

- ④ ทำการกด  จาก Toolbar button แล้ว Input "X0"

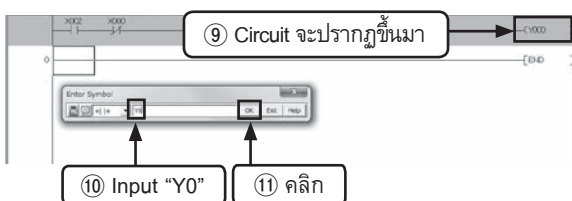
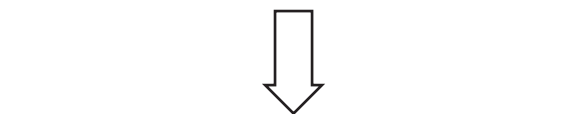
- ⑤ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**



- ⑥ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา ( $X_0$ )

- ⑦ ทำการกด  จาก Toolbar button แล้ว Input "Y0"

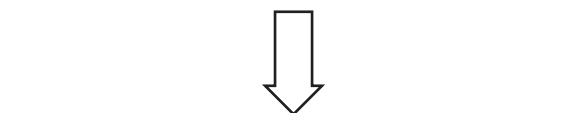
- ⑧ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**

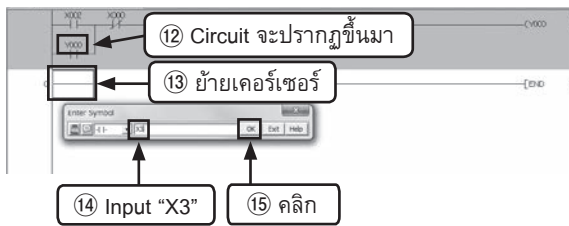


- ⑨ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา ( $-Y_0-$ )

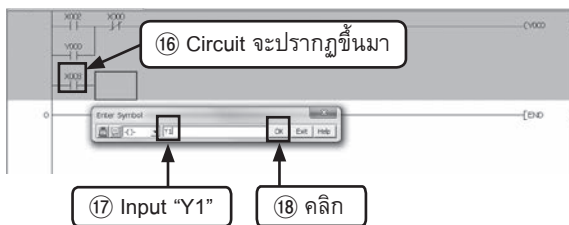
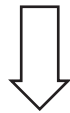
- ⑩ ทำการกด  จาก Toolbar button แล้ว Input "Y0"

- ⑪ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**

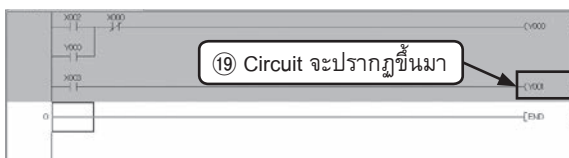




- ⑫ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา ( $Y_0$ )
- ⑬ ทำการย้ายตำแหน่งเคอร์เซอร์ไปที่หัวแถวของแถวถัดไป
- ⑭ ทำการกด  $F5$  จาก Toolbar button แล้ว Input "X3"
- ⑮ ทำการยืนยันโดยกด  หรือ



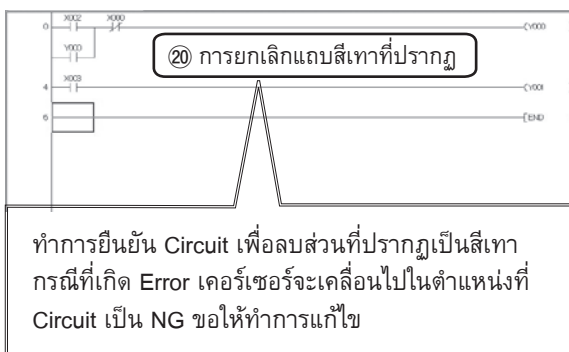
- ⑯ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา ( $X_3$ )
- ⑰ ทำการกด  $F7$  จาก Toolbar button แล้ว Input "Y1"
- ⑱ ทำการยืนยันโดยกด  หรือ



- ⑲ จะปรากฏ Circuit ที่เรา Input ไว้แล้วขึ้นมา ( $-(Y_1)-$ )




(Build)



## ⑳ Compile Operation (สำคัญ)

ทำการกด "Compile" เพื่อทำการยืนยัน Circuit ที่ยังไม่ได้รับการยืนยัน (ส่วนที่แสดงเป็นสีเทา)

ขอให้เลือกดำเนินการตามขั้นตอนที่ระบุด้านล่าง

- กด  (Build)
- กด  จาก Toolbar
- ทำการเลือก [Conversion/Compile] → [Compile] จากเมนู

## 3.4 การเขียน Program ลงใน PLC

Sequence program ที่จัดทำขึ้นมา จะถูกเขียนลงใน FX PLC

### 3.4.1 การเชื่อมต่อกับ PLC

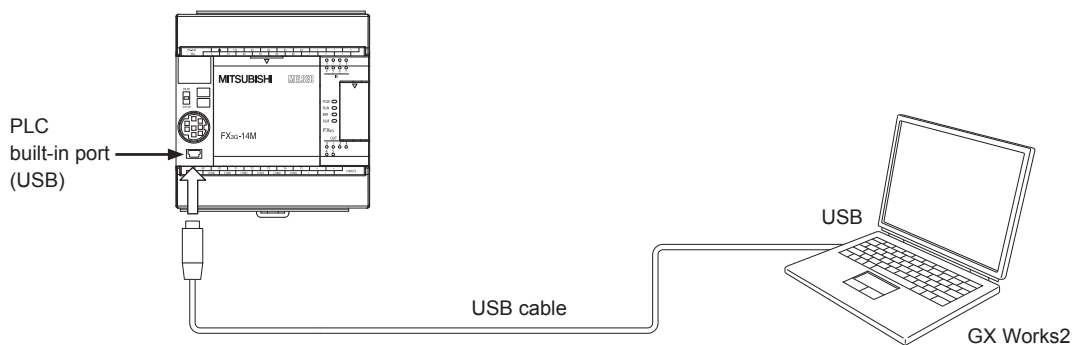
การเชื่อมต่อกับ FX3G PLC (เชื่อมต่อโดยใช้สาย USB cable)

[การเตรียมด้านคอมพิวเตอร์]

จำเป็นต้อง Install USB driver ลงในคอมพิวเตอร์เพื่อทำการต่อสาย USB ระหว่างคอมพิวเตอร์กับ FX3G PLC

ขั้นตอนการ Install USB driver ให้ปฏิบัติตามคู่มือใน “GX Works2 Operating Manual (บท Common)”

[รูปแบบการเชื่อมต่อ]



#### Point

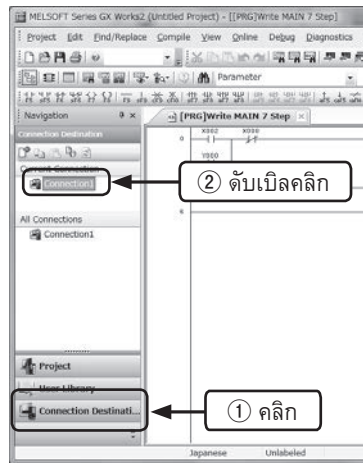
#### วิธีการติดตั้งไดรเวอร์ USB

กรณีใช้งาน <Windows® 7, Windows® 8>

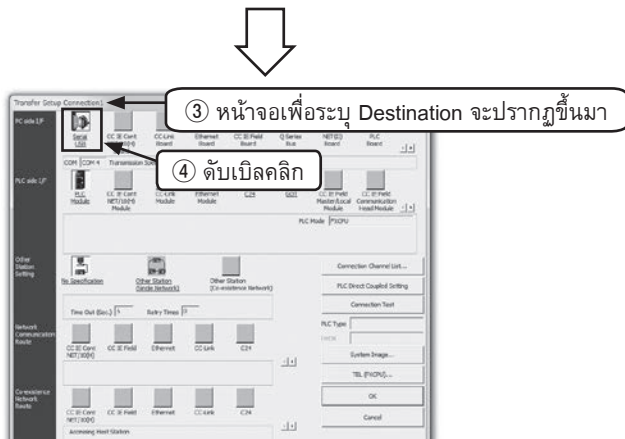
- ① ติดสายเคเบิ้ลระหว่างคอมพิวเตอร์ และ PLC เข้าด้วยกัน จากนั้นเปิด PLC
  - ② เลือก “Device Manager” ใน “System and Security” จากแท็บ “Control Panel” จากนั้นคลิกขวาที่ “Other devices” แล้วเลือก “Update Driver Software...”
  - ③ เมื่อหน้าจอของ “Update Driver Software” แสดงขึ้นให้เลือก “Browse my computer for driver software” ต่อไปจะปรากฏ “Easysocket ¥USBDrivers” โฟลเดอร์ที่มีการติดตั้ง GX Works2 ในกรณีผลิตภัณฑ์ MELSOFT ถูกติดตั้งไว้หลายตัว ให้อ้างอิงปลายทางการติดตั้งของผลิตภัณฑ์ตั้งแต่ครั้งแรก
- \* กรณีใช้ “FX3U-USB-BD”, “FX-USB-AW” เชื่อมกับคอมพิวเตอร์ เนื่องจากวิธีการติดตั้งมีความแตกต่างตามแต่ละคอมพิวเตอร์ที่ใช้งาน ฉะนั้น กรุณาตรวจสอบละเอียดใน “GX Works2 Operating Manual (Common)”

### 3.4.2 “Transfer Setup” ใน GX Work2 และ “Writing programs”

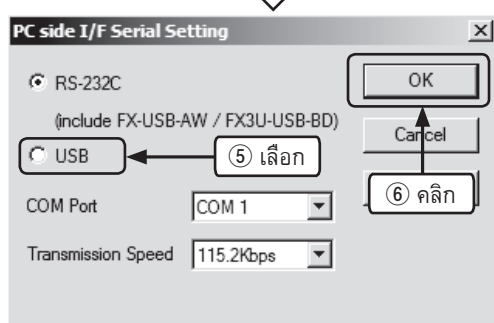
ทำการ Set up GX Works2 เพื่อเชื่อมต่อกับ PLC



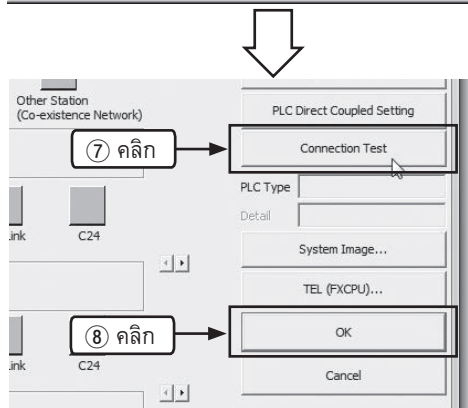
- ① คลิกที่ “Connection Destination” ในหน้าจอ Navigation
- ② ดับเบิลคลิกที่ “Connection 1”



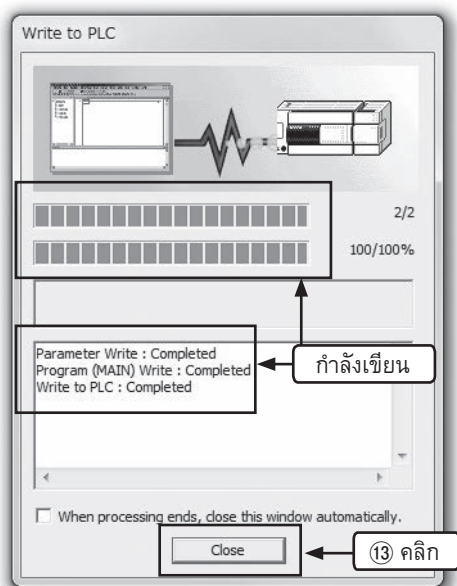
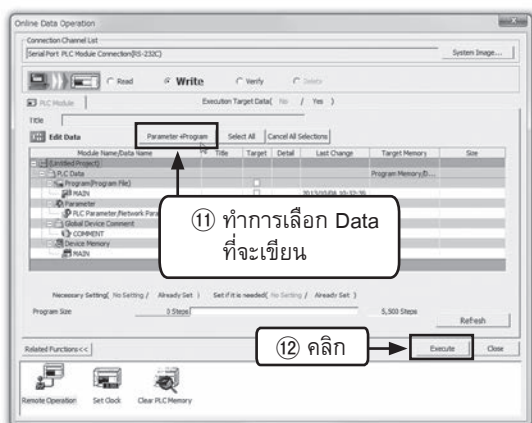
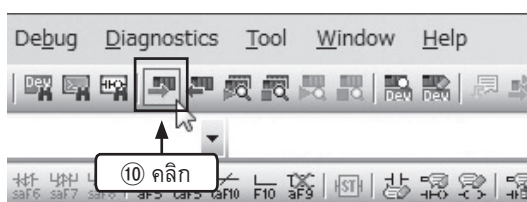
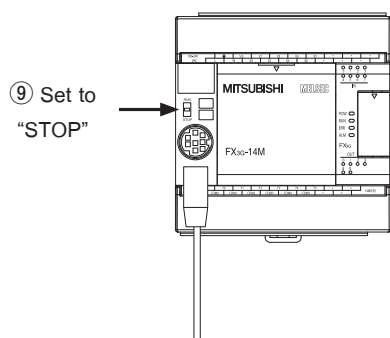
- ③ หน้าจอเพื่อระบุ Destination จะปรากฏขึ้นมา
- ④ ดับเบิลคลิกที่ Icon 



- ⑤ ทำการ Set up communication port ด้านคอมพิวเตอร์
  - ทำการเลือก “USB” เมื่อต้องการเชื่อมต่อไปยัง Built-in USB port ใน FX3G PLC
- ⑥ จากนั้นคลิก [OK]



- ⑦ คลิกปุ่ม [Connection Test] เพื่อทำการตรวจสอบการรับส่งสัญญาณไปที่ PLC
- ⑧ คลิก [OK] หลังตรวจสอบเสร็จ เพื่อยืนยันรายละเอียดที่ทำการตั้งค่าไป



⑨ ที่ PLC ทำการสับสวิตช์ “RUN/STOP” ไปที่ “STOP”

[หัวข้อเพิ่มเติม : Function auto RUN/STOP จาก Programming software]

- กรณีทำการเขียนโดยที่ PLC อยู่ในสถานะ RUN หลังจากดำเนินการตามข้อ ⑫ แล้ว จะมีข้อความว่า “หลัง Remote STOP จะดำเนินการเขียน PLC หรือไม่”

กรณีที่ต้องการเขียนให้คลิก [Yes (Y)]

- เมื่อเขียนเสร็จแล้ว จะมีข้อความว่า “PLC อยู่ในสถานะ STOP จะ REMOTE ไป RUN หรือไม่”

เพื่อให้ PLC RUN ให้คลิก [Yes (Y)]

⑩ เลือก  จาก Toolbar หรือเลือก [Online] → [Write to PLC] จากเมนู

⑪ คลิก [Parameter + Program]

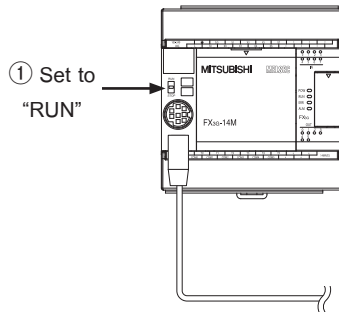
⑫ คลิก [Execute]

(อ้างอิงหัวข้อเพิ่มเติมข้อ ⑨)

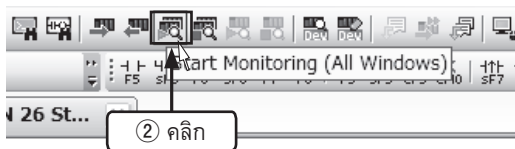
จะมีกล่องข้อความปรากฏขึ้นเพื่อแสดงสถานะกำลัง Write

⑬ หลังจากเสร็จสิ้นกระบวนการคลิก [Close]


### 3.4.3 การ Monitoring เพื่อดูการทำงานของ PLC



① ที่ PLC ทำการสับสวิตช์ “RUN/STOP” ไปที่ “RUN”

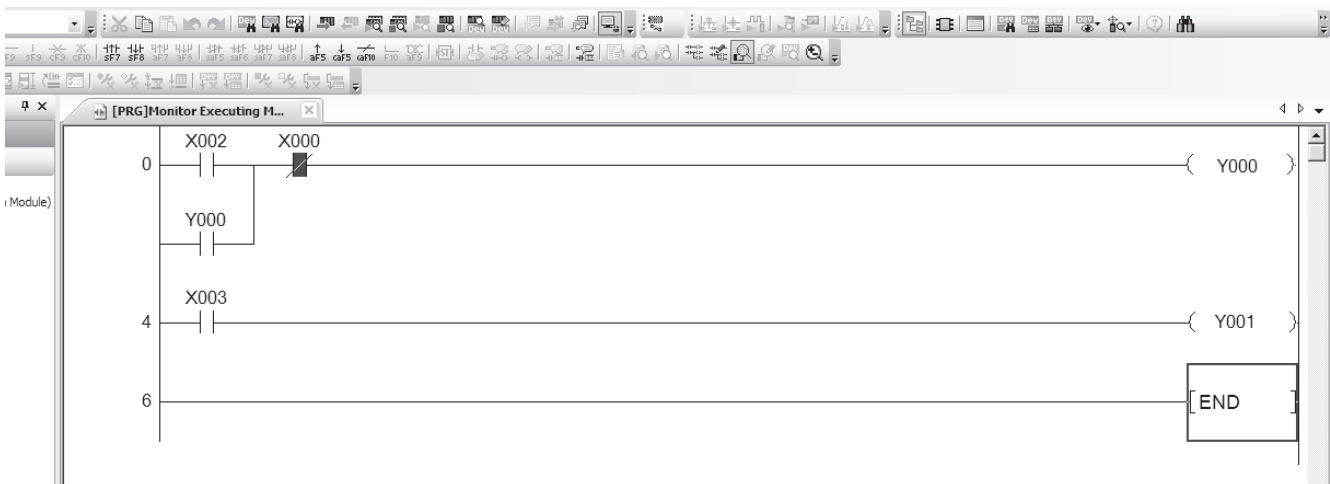


② ดำเนินการอย่างใดอย่างหนึ่งตามตัวเลือกด้านล่าง

- กด **F3** (Monitor Mode/Start Monitoring)
- กด  จาก Toolbar
- เลือก [Online] → [Monitor] → [Start Monitoring (All Windows)] จากเมนู



#### ตรวจสอบการทำงานโดยการ Monitor Mode



- 1) ทำการตรวจสอบว่าเมื่อ [Switch X000 “OFF”] แล้ว [Switch X002 “ON”], [Output Y000 “ON”] หรือไม่
- 2) ทำการตรวจสอบว่าแม้ [Switch X002 “OFF”] อย่างไร [Output Y000 ก็ยัง “ON”] อยู่หรือไม่
- 3) ทำการตรวจสอบว่าเมื่อ [Switch X000 “ON”] แล้ว [Output Y000 “OFF”] หรือไม่
- 4) ทำการตรวจสอบว่าเมื่อ [Switch X003 “ON/OFF”] สลับกัน [Output Y001 “ON/OFF”] สลับกันด้วย

#### Point

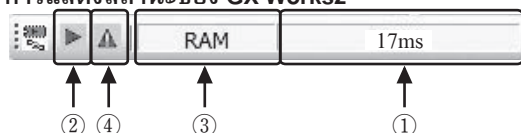
#### การ Edit หรือการหยุดการ Monitoring

- ทำการหยุดการ Monitor โดยคลิกที่ Icon  (Stop Monitoring) หรือเลือก [Online] → [Monitor] → [Stop Monitoring]
- ในการ Edit circuit สามารถทำได้ทั้งตอนที่กำลัง Monitor อยู่ หรือหยุดการ Monitor ไปแล้ว

## อ้างอิง

### (1) Dialog ที่แสดง Monitor status

#### การแสดงผลสถานะของ GX Works2



#### ① Scan time

ทำการแสดง Scan time สูงสุดใน Sequence program

#### ② PLC status

ทำการแสดงสถานะของ PLC

เมื่อ PLC RUN จะแสดงเป็น “▶” เมื่อ PLC STOP จะแสดงเป็น “■”

#### ③ Memory status

ทำการแสดงรายละเอียด Memory ของ PLC

#### ④ ERR status (PLC Diagnostics)

เมื่อทำการคลิกจะสามารถตรวจสอบรายละเอียดได้

### (2) วิธีดูการแสดงผล Status ของ Ladder monitor

#### ① คำสั่งหน้าสัมผัส (Contact Instruction)

Input contact Type	X0 : OFF	X0 : ON
หน้าสัมผัสปกติเปิด a	X000 ├─┤ หน้าสัมผัสเปิด (Contact open)	X000 ├─■─┤ หน้าสัมผัสปิด (Contact close)
หน้าสัมผัสปกติปิด b	X000 ├─■─┤ หน้าสัมผัสปิด (Contact close)	X000 ├─┤ หน้าสัมผัสเปิด (Contact open)

#### ② คำสั่งเอาท์ (OUT Instruction)

Driving status Type	ไม่ทำงาน/ไม่เคลื่อนไหว	ทำงาน/เคลื่อนไหว
OUT instruction	─(Y000)─┤	─┤Y000┤─┤
Set instruction, etc.	─[ SET M0 ]─┤	─┤SET M0┤─┤

คำสั่ง RST (RST Instruction) คือ การแสดง ON/OFF ของ Device ที่จะ Reset

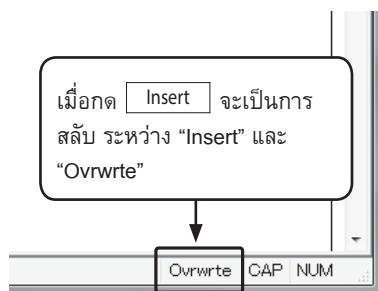
Device status Type	Reset device OFF	Reset device ON
RST instruction	─┤RST M0┤─┤	─[ RST M0 ]─┤

## 3.5 การแก้ไขโปรแกรม

### 3.5.1 การแก้ไขโปรแกรม

#### Point

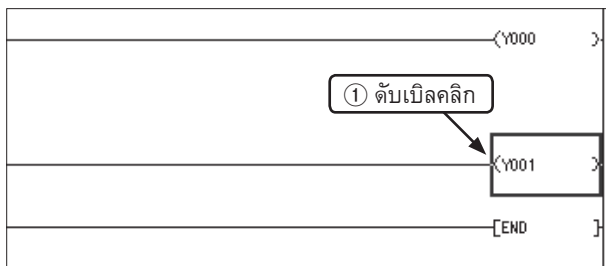
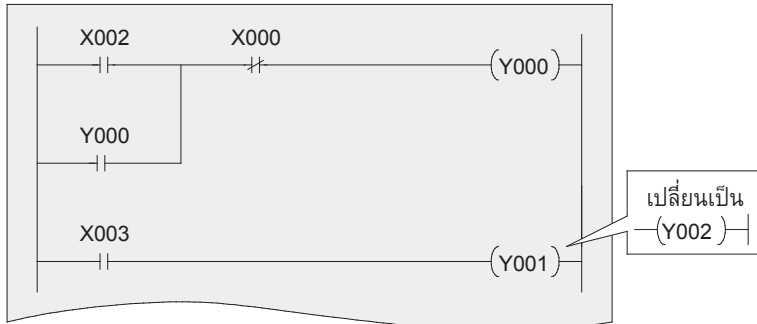
- ตัวอักษรที่จะ Input ต้องเป็นตัวอักษรภาษาอังกฤษพิมพ์เล็ก ตัวอักษรภาษาอังกฤษพิมพ์ใหญ่ไม่สามารถ Input ได้
- การสลับระหว่าง “Ovrwrte” และ “Insert”
  - กรณีที่จะทำการแก้ไข Program ที่มีอยู่แล้ว ขอให้เลือก “Ovrwrte” mode
  - เมื่อ Set mode เป็น “Insert” จะเป็นการ Input เป็น Program ใหม่



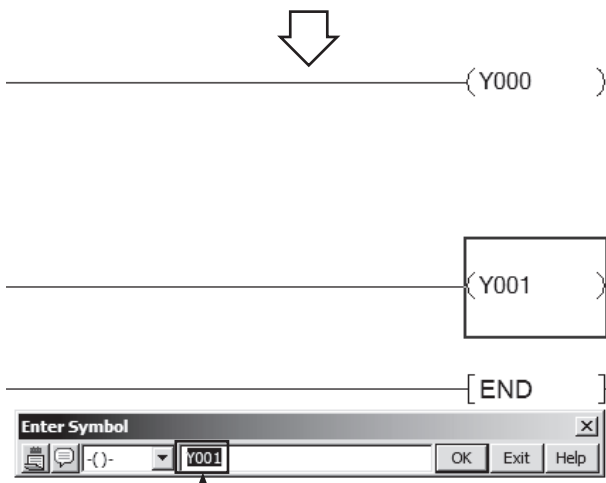


# ① การเปลี่ยนหมายเลข OUT coils และ contacts

[Program ที่จะแก้ไข]

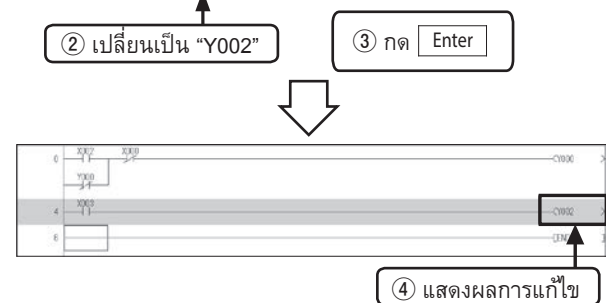


① ทำการดับเบิลคลิกในส่วนที่จะทำงานแก้ไข

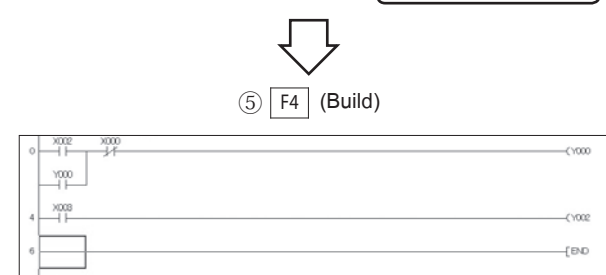


② เปลี่ยน "Y001" → "Y002"

③ กด  หรือ  เพื่อยืนยัน



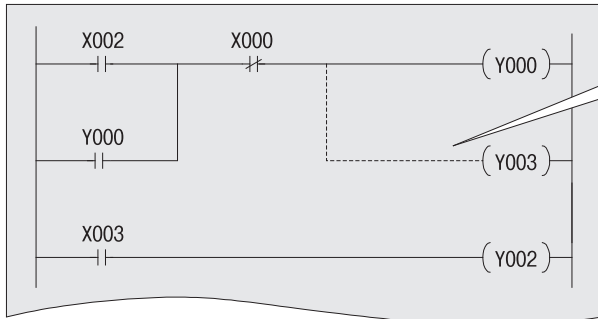
④ จะปรากฏ Program หลังการแก้ไขขึ้นแสดงในกรอบสี่เทา



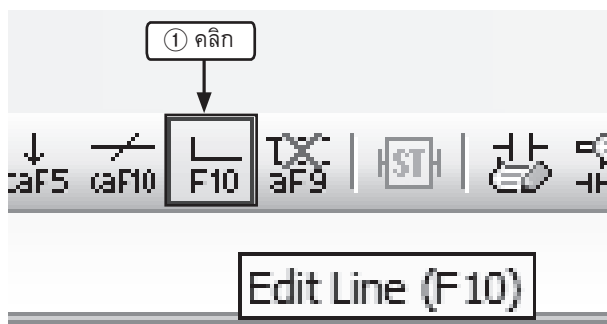
⑤ กด  (Build) เพื่อยืนยัน

## ② เพิ่ม Lines

[Program ที่ จะทำการเพิ่ม Lines]

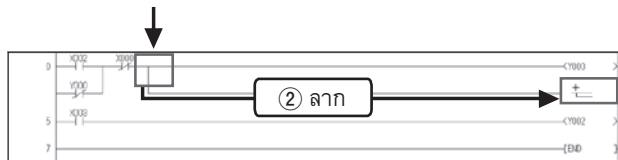


เพิ่ม Line แนวตั้ง  
หรือแนวนอน และ  
จัดทำ OUT coils

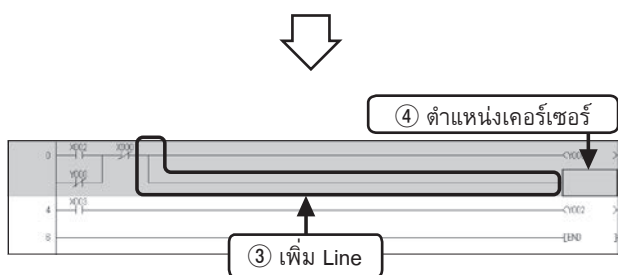


① คลิก Toolbar (F10)

วางเคอร์เซอร์ที่ด้านบนขวา  
ในตำแหน่งที่ต้องการเพิ่ม

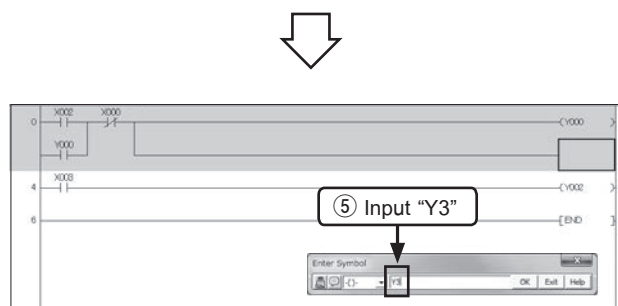


② เมื่อนำเคอร์เซอร์ไปวางด้านขวาบนของตำแหน่งที่ต้องการ  
เพิ่ม Line แล้ว ให้ทำการลากเส้น



③ Line จะเพิ่มขึ้นตามตำแหน่งที่ลากเคอร์เซอร์

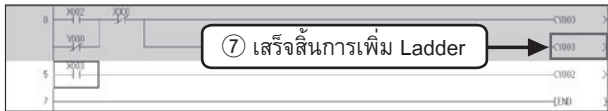
④ วางเคอร์เซอร์ในตำแหน่งที่ต้องการเพิ่ม OUT coil แล้วคลิก จาก Toolbar



⑤ Input "Y3" เข้าไป

⑥ ทำการยืนยันโดยกด  หรือ

⑥ กด

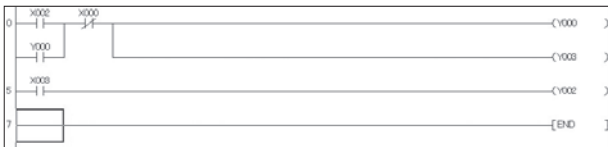


⑦ เมื่อทำการเพิ่ม Ladder แล้ว จะแสดงเป็นบล็อกสี่เท่า



⑧ **F4** (Build)

⑧ ทำการกด **F4** (Build) เพื่อยืนยันการเปลี่ยนแปลง



- เมื่อดำเนินการเสร็จแล้ว ทำการคลิก  จาก Toolbar อีกครั้ง

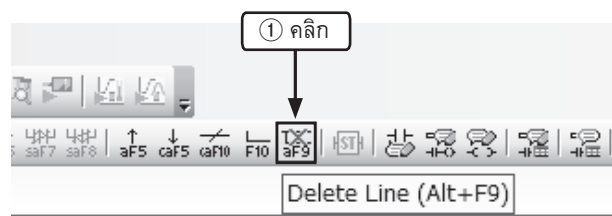
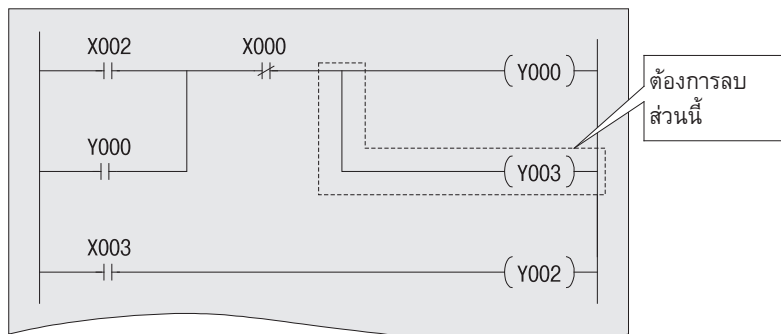
## อ้างอิง

### การเพิ่มหรือลบ Lines โดยการใช่ Keys

- GX Works2 สามารถทำการเพิ่มหรือลบ Lines ได้โดยใช้ **Ctrl** +  ,  ,  , 

### ③ การลบ Lines

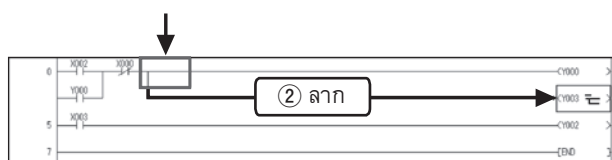
[Program ที่ต้องการลบ Lines]



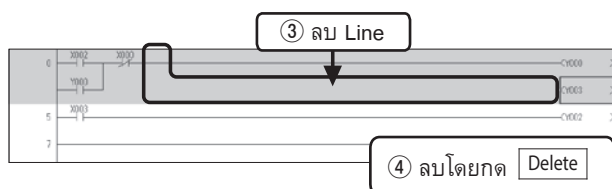
① คลิก  ( Alt + F9 ) จาก Toolbar



วางเคอร์เซอร์ที่ด้านบนขวา  
ในตำแหน่งที่ต้องการเพิ่ม



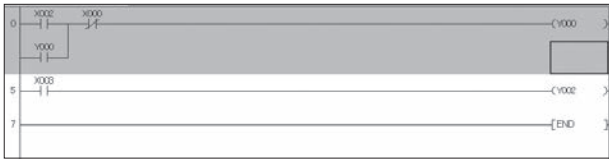
② ทำการลากเคอร์เซอร์จากด้านขวาบนของตำแหน่งที่ต้องการลบเส้น ลากจนสุดตำแหน่ง



③ ทำการลบ Lines

④ OUT coils ทำการลบโดยกด Delete



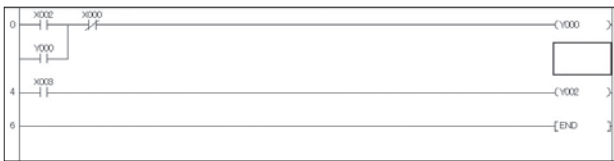



⑤ Program ที่ต้องการลบจะปรากฏเป็นกรอบสีเทา



⑥ **F4** (Build)

⑥ กด **F4** (Build) เพื่อเปลี่ยนรายละเอียด



- กรณีที่ต้องการจบการดำเนินการให้กด  จาก Toolbar อีกครั้ง

## อ้างอิง

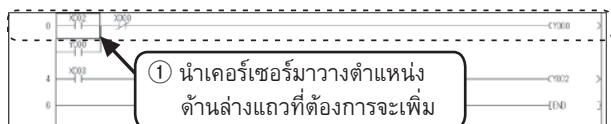
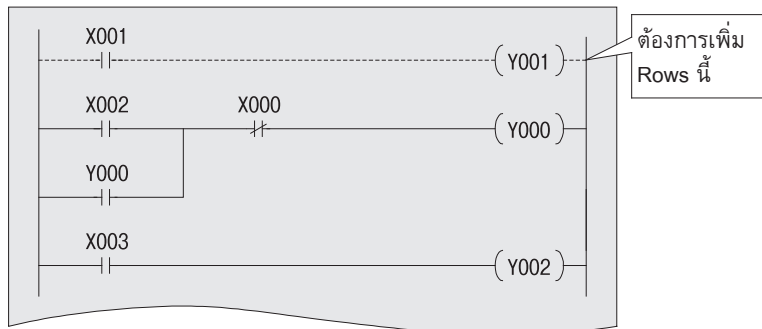
### การเพิ่มหรือลบ Lines โดยใช้ Keys

- GX Works2 สามารถทำการเพิ่มหรือลบ Lines ได้โดยใช้ **Ctrl** +  ,  ,  , 

## 3.5.2 การแทรก/ลบ Rows

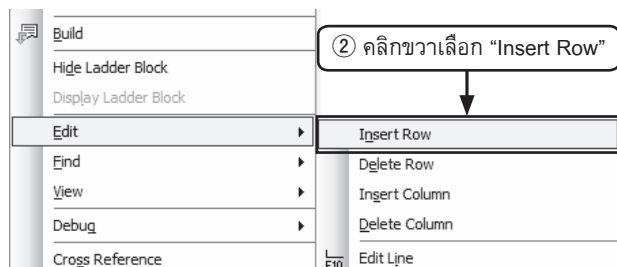
### ① การแทรก Rows

[Program ที่ต้องการแทรก Rows]



การแทรก Rows ทำการแทรกด้านบนแถวที่เคอร์เซอร์วางอยู่

① นำเคอร์เซอร์วางแถวล่างของตำแหน่งที่ต้องการแทรก

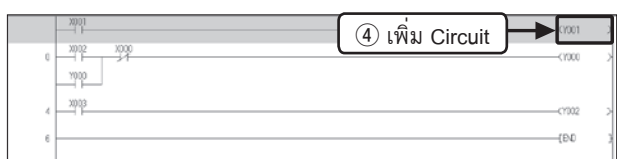


② ทำการคลิกขวาในตำแหน่งใดก็ได้เพื่อเลือก

- [Edit] → [Insert Row]



③ เสร็จสิ้นการแทรก Rows



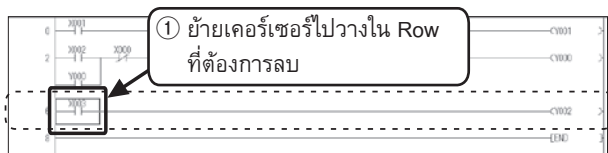
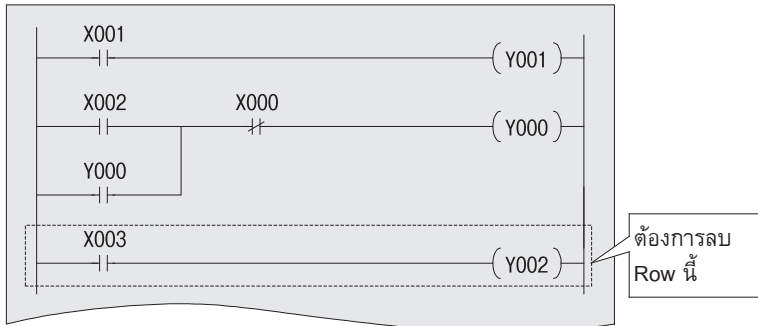
④ ทำการเพิ่ม Circuit ลงในแถวที่แทรกเข้าไปใหม่



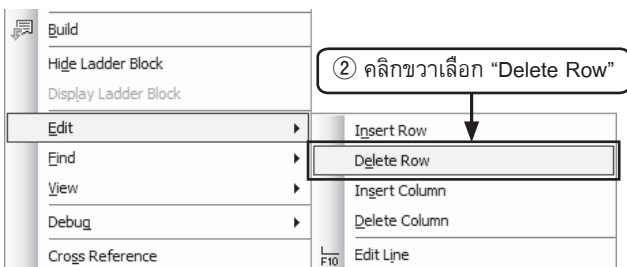
⑤ ทำการกด **Build** (F4) เพื่อยืนยันการเปลี่ยนแปลง

## ② การลบ Rows

[Program ที่ต้องการลบ Rows]

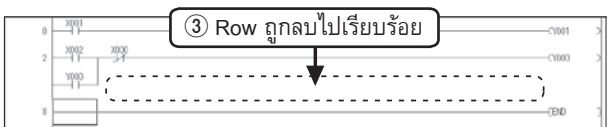


① ย้ายเคอร์เซอร์ไปวางใน Row ที่ต้องการลบ



② วางเมาส์ในตำแหน่งใดก็ได้แล้วคลิกขวเลือก

- [Edit] → [Delete Row]



③ Row ถูกลบไปเรียบร้อยแล้ว

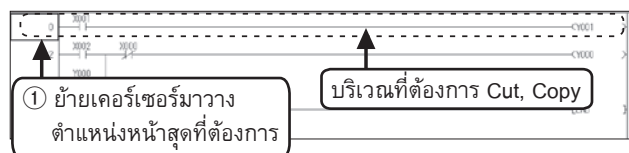
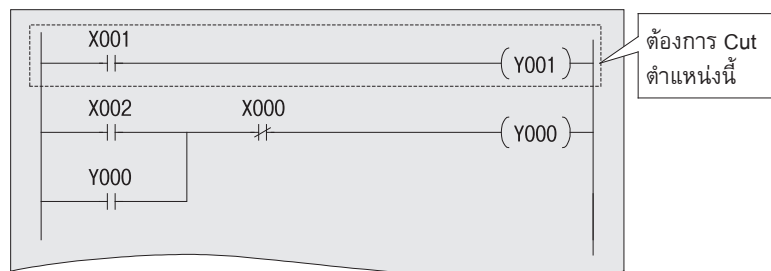
### Point

กรณี que การลบ Rows จะไม่มีการแสดงเป็นกรอบสีเทา ให้กด [Build] (F4) เพื่อทำการยืนยันการดำเนินการ

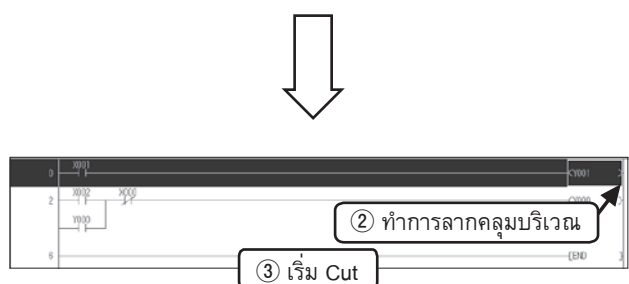
### 3.5.3 การ Cut & Copy (Pasting) ladder

#### ① การ Cut


[Program ที่ต้องการ Edit]

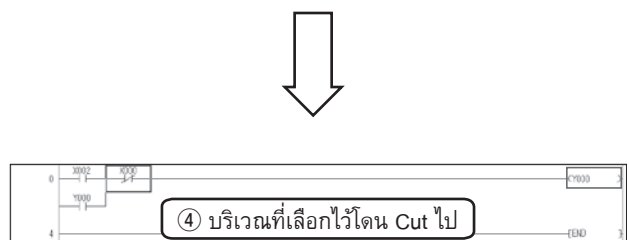


① ย้ายเคอร์เซอร์มาวางหน้า Circuit ที่ต้องการ Cut



② ทำการลากคลุมทุกตำแหน่ง

③ Cut โดยเลือก  จาก Toolbar หรือเลือก [Edit] → [Cut] (Ctrl + X) แล้วทำการ Cut



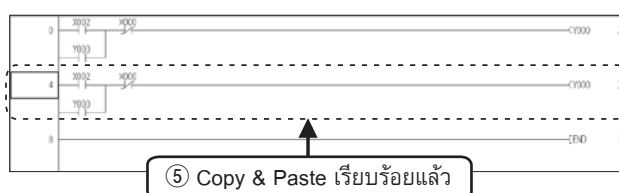
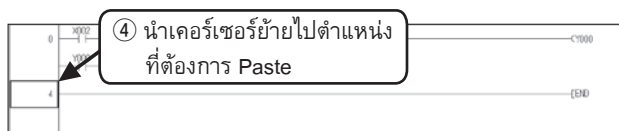
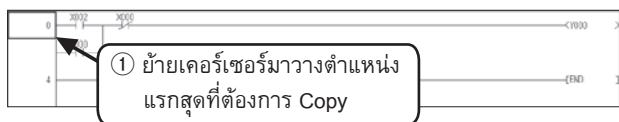
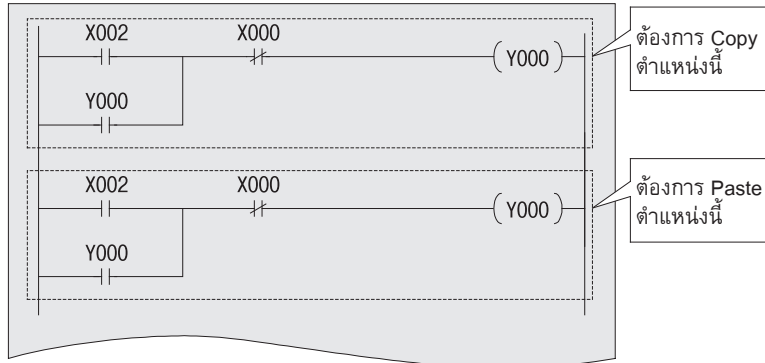
④ บริเวณที่เลือกไว้จะถูกรับ Cut

กรณี Cut บางส่วนของ Circuit จะเหลือแสดงเป็นแถบสีเทาอยู่ หลังจากแก้ Circuit ไว้ กด [F4] (Build) เพื่อยืนยันการดำเนินการ



## ② Copy (Pasting)


[Copy (Pasting)]



นำ Program ที่ "Cut" แล้วมาวางแทนหน้า

① ทำการย้ายเคอร์เซอร์มาวางไว้ที่แถวหน้าของ Circuit ที่จะทำการ Copy

② ทำการลากคลุมตำแหน่งจบ

③ ทำการเลือก  ที่ Toolbar หรือเลือก [Edit] → [Copy] (Ctrl + C) จากเมนูแล้วทำการ Cut


④ ย้ายเคอร์เซอร์มาวางในตำแหน่งที่ต้องการ Paste


### Point

การกดเปลี่ยน  Insert

"Ovwrite" mode: การ Paste โดยการเขียนทับในตำแหน่งที่เคอร์เซอร์วาง

"Insert" mode : ทำการ Insert ข้างบนตำแหน่งที่วางเคอร์เซอร์

⑤ ทำการเลือก  จาก Toolbar หรือเลือก [Edit] → [Paste] (Ctrl + V) จากเมนู

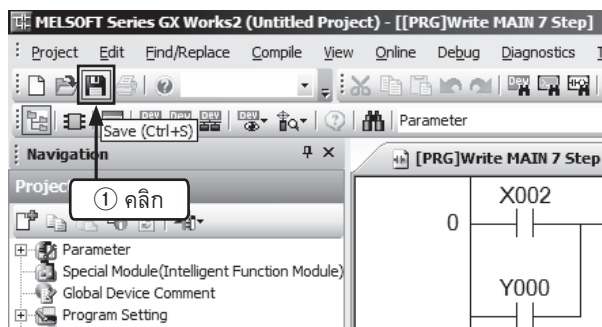
กรณีต้องการ Paste เฉพาะบางส่วนของ Program จะมีการแสดงเป็นกรอบสีเทา ให้กด  (F4) (Build) เพื่อเป็นการยืนยัน


## 3.6 การบันทึก Program ที่จัดทำเสร็จแล้ว

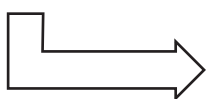
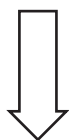
### 3.6.1 Save, Save as

#### Point

กรณีที่ Circuit ยังไม่ได้รับการยืนยัน ให้กด **Build** (F4)



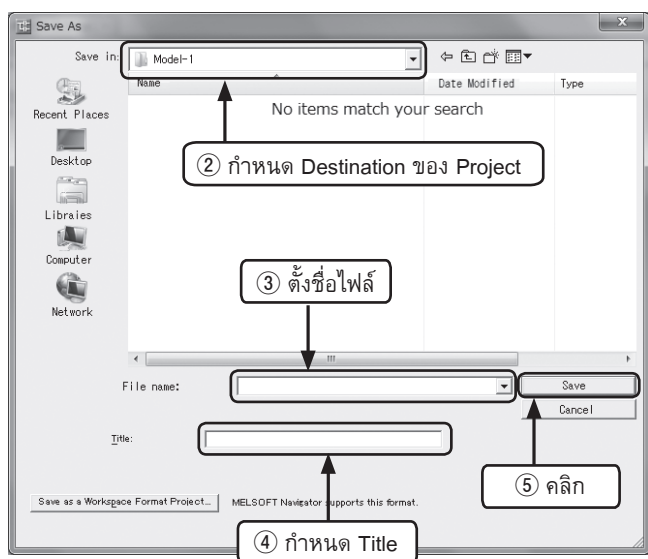
- ① ทำการเลือก  จาก Toolbar หรือเลือก [Project] → [Save] ( **Ctrl** + **S** ) หรือ [Save As] จากเมนู



(กรณีที่ Overwrite)

**Save Program** เสร็จเรียบร้อย

(เฉพาะกรณี Save as เท่านั้น)




- ② กำหนด Destination ของ Project
- ③ ตั้งชื่อไฟล์
- ④ กำหนด Title ที่จะแสดงเนื้อหาของ Program (ตั้งได้ตามต้องการ)
- ⑤ คลิก **Save**

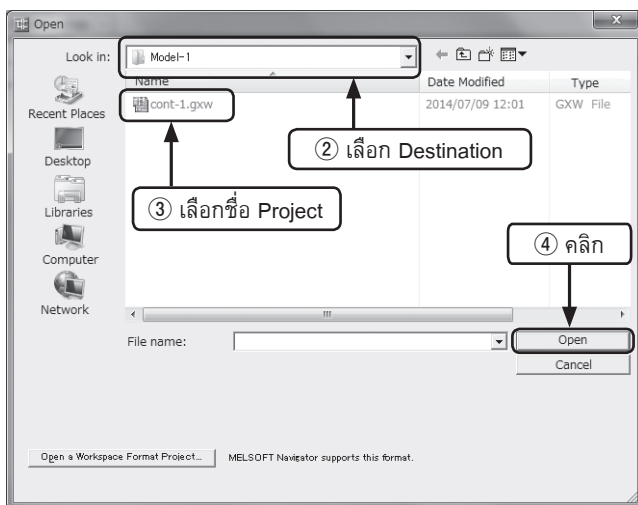
### 3.6.2 การ Read project

#### อ้างอิง

กรณีเปิด Project อื่นๆ ค้างไว้ ในตอนที่ทำการ Read file ใหม่ Project นั้นจะถูกปิดไป  
ถ้า Project ที่จะถูกปิดนั้นยังไม่ได้ทำการ Convert หรือ Save จะมีข้อความแจ้งเตือนขึ้นมา



- ① เลือก  จาก Toolbar หรือเลือก [Project] → [Open] (  +  ) จากเมนู



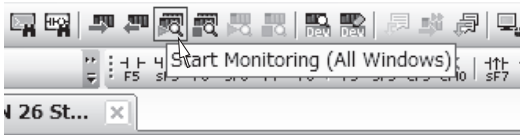
- ② เลือก Destination ของ Project  
③ เลือก Project ที่จะเปิด  
④ คลิก  เพื่อเปิด Project


## 3.7 Debug program

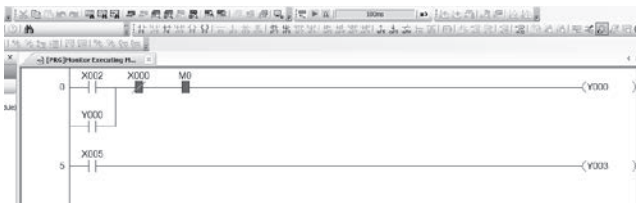
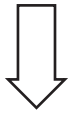
เรื่องการอ่าน Program หรือการเชื่อมต่อ PLC ขอให้อ้างอิง “ภาคผนวก 1.4 การเขียน Program ใน PLC”

### 3.7.1 Ladder monitor

ทำการ Monitor สภาพการทำงานของ Coils หรือสภาพการทำงานหน้าสัมผัส (Contact) ในแต่ละจุดในขณะที่ Program กำลังแสดงอยู่ (รายละเอียดในการแสดงขอให้อ้างอิง “การ Monitoring การทำงานของ Program”)




- ① เลือก  จาก Toolbar หรือเลือก [Online] → [Monitor] → [Start Monitoring (All Windows)] จากเมนู



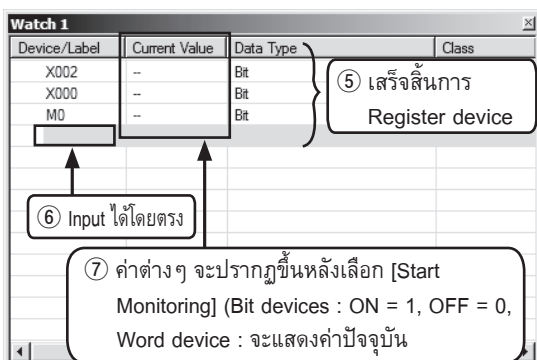
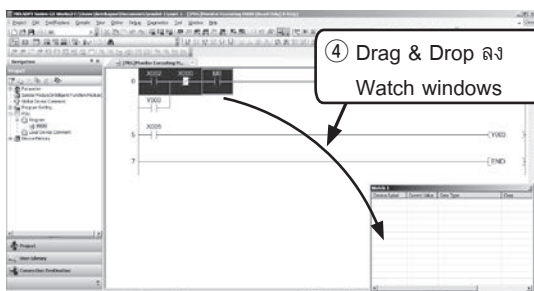
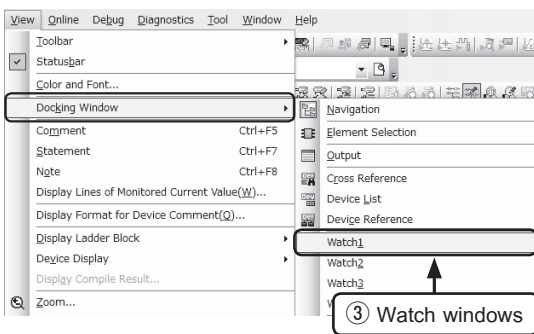
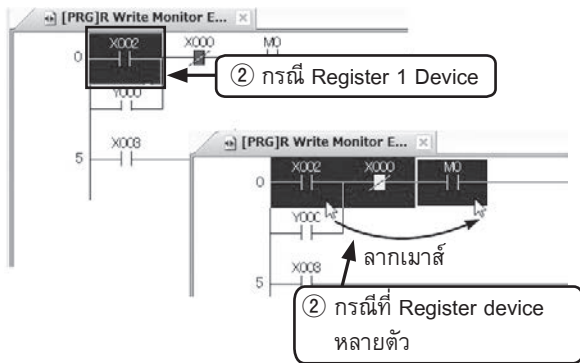
- ② ที่ Ladder monitor จะแสดงค่าปัจจุบันของ Word device (Timer, Counter, Data register) และสถานะการ ON/OFF ของ Ladder

#### อ้างอิง

- ใน GX Works2 สามารถหยุดการ Monitor ได้โดยการเลือก  (Stop Monitoring) จาก Toolbar หรือเลือก [Online] → [Monitor] → [Stop Monitoring] จากเมนู ส่วนการ Edit program สามารถดำเนินการได้ทั้งในขณะที่กำลัง Monitor หรือหยุดการ Monitor แล้ว

### 3.7.2 Device registration monitor

เป็นการกำหนดขอบเขตของ Program หรือการกำหนด Device 1 Device ในหน้าจอ Ladder monitor แล้วทำการ Register device ส่วนนั้นๆ ลงใน Watch windows



- ① ไปที่หน้าจอ Monitor mode (อ้างอิงภาคผนวก 3.7.1 “Ladder Monitor”)
- ②
  - กรณีที่ต้องการ Register 1 Device ให้ใช้เมาส์คลิกเลือกที่ Device
  - กรณีที่ต้องการ Register device หลายตัว ให้ใช้เมาส์ลากคลุมพื้นที่ที่ต้องการ

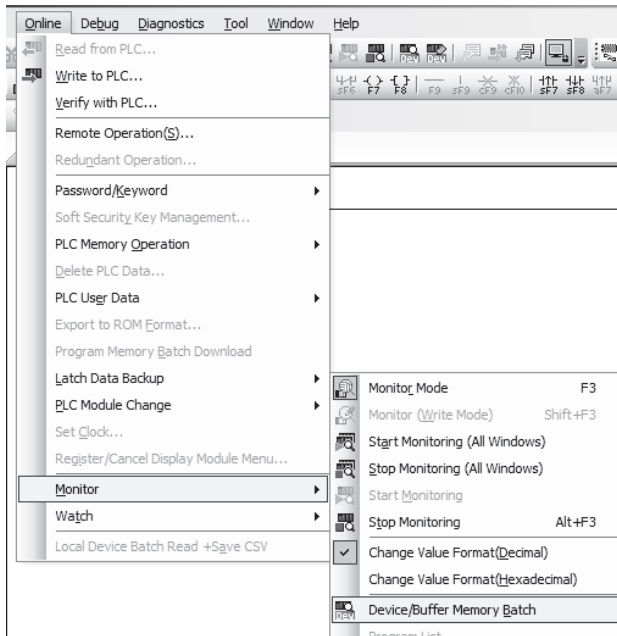
- ③ เลือก [View] → [Docking Window] → [Watch1] และ Watch windows จะแสดงออกมา

- ④ Device ที่เลือกไว้แล้ว ให้ทำการ Drag และ Drop ลงใน Watch windows

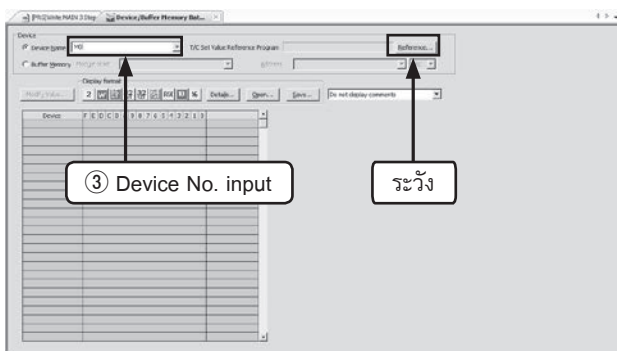
- ⑤ Device ที่เลือกไว้จะถูก Register ลงใน Watch window
- ⑥ สามารถ Input device names ได้โดยตรงลงในช่องว่าง “Device/Label” เช่น “X0”, “M0”, “D0”
- ⑦ เลือก [Online] → [Monitor] → [Start Monitoring] จากเมนู

### 3.7.3 Device batch monitor (การเรียกดู Device ทั้งหมด)

การระบุหมายเลข Device และการ Monitor device ต่อเนื่อง

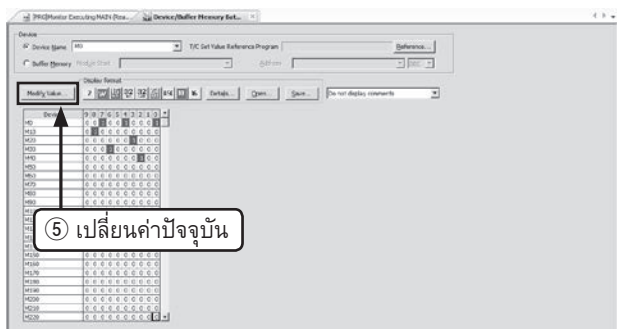


- ① ไปที่หน้าจอ Monitor mode (อ้างอิงภาคผนวก 3.7.1 “Ladder Monitor”)
- ② ทำการเลือก [Online] → [Monitor] → [Device/Buffer Memory Batch] จากเมนู



- ③ ทำการ Input หมายเลขแรกของ Device ที่จะทำการ Monitor ใน Window “Device Memory Batch” กด

(ระว่าง) กรณีที่กำหนด Timer, Counter ให้คลิก [Reference] แล้วเลือก Program “MAIN”

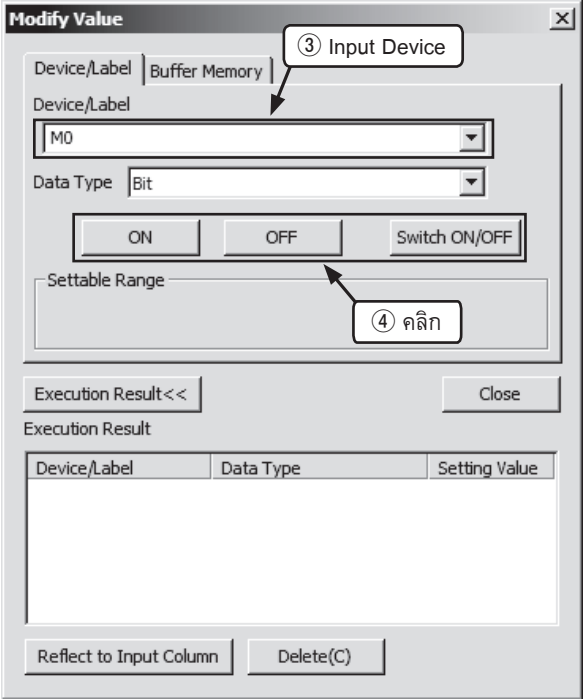
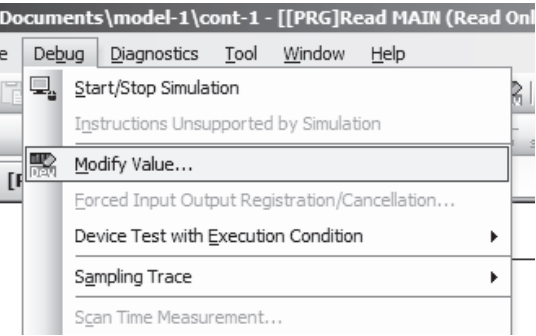


- ④ จะปรากฏสถานะการทำงานขึ้นตามการทำงานของ Device
  - Bit devices (X, Y, M, S) : ON = 1, OFF = 0
  - Timers และ Counters: สถานะ ON/OFF ค่าที่กำหนดไว้และค่าปัจจุบันของ Contact Coil
  - Data registers: ค่าปัจจุบัน
- ⑤ หลังจากใช้เมาส์เลือกส่วนที่ต้องการให้ปรากฏใน Monitor data ให้ทำการคลิกที่ [Modify Value] จะสามารถเปลี่ยนค่าปัจจุบันของ Word device หรือ สามารถ Set forcibly ON/OFF

### 3.7.4 Device test

#### ① Force ON/OFF (การบังคับเปิด/ปิด)

ทำการ Force ON/OFF bit device ของ PLC (เช่น M, Y, T, C) (จะไม่ Force ON/OFF ใน X)  
หาก Force ON/OFF Input (X) ในขณะที่ PLC กำลัง RUN อยู่ จะทำการ ON/OFF เฉพาะเมื่อครบ 1 Cycle เท่านั้น  
ในกรณีที่จะตรวจสอบการทำงานของ Output (Y) ต้องให้ PLC อยู่ในสถานะ STOP ก่อน



① ไปที่หน้าจอ Monitor mode (อ้างอิงภาคผนวก 3.7.1 “Ladder Monitor”)

② เลือก [Debug] → [Modify Value] จากเมนู

③ ทำการ Input หมายเลข Device ที่จะทำการ Force ON/OFF

- ④
- [Force ON] : Device ON
  - [Force OFF] : Device OFF
  - [Force ON/OFF Invert] : Device จะ ON/OFF สลับกัน ทุกครั้งที่กด

#### อ้างอิง

#### Force ON/OFF (Ladder monitor window)

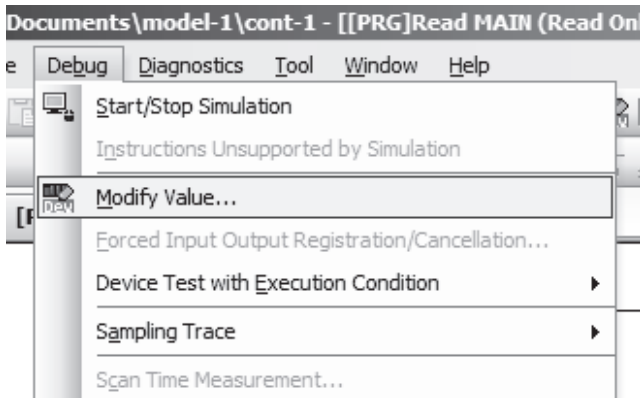
- ทำการกด [Shift] key ค้าง ใน Bit device ไหนก็ได้ (Contact, Coil) บน Ladder monitor window แล้วดับเบิลคลิก จะสามารถเลือก Force ON/OFF device ได้

## ② การเปลี่ยนแปลงค่าปัจจุบันของ Word device

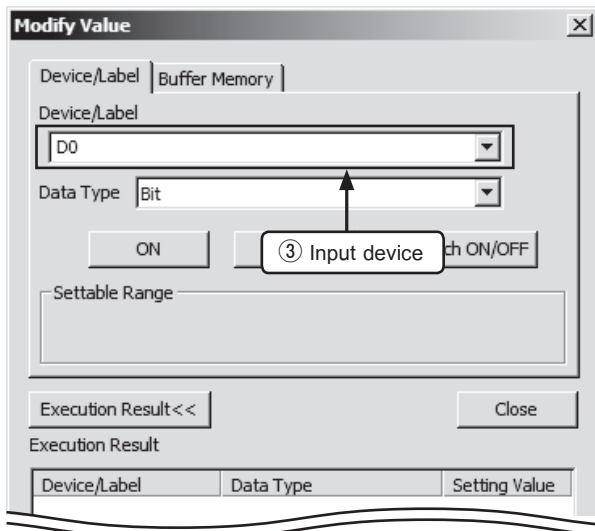
การเปลี่ยนค่าปัจจุบันของ Word device ของ PLC (เช่น T, C, D เป็นต้น) เป็นค่าที่ต้องการ

① ไปที่หน้าจอ Monitor (อ้างอิงภาคผนวก 3.7.1 “Ladder Monitor”)

② ทำการเลือก [Debug] → [Modify Value] จากเมนู

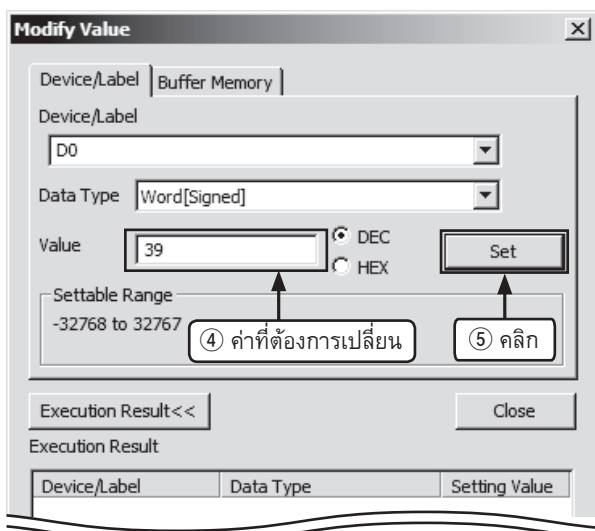


③ ทำการป้อนหมายเลข Device ที่ต้องการเปลี่ยนแปลงแล้วกด Enter



④ ทำการป้อนค่าที่ต้องการเปลี่ยน

⑤ คลิก [Set]



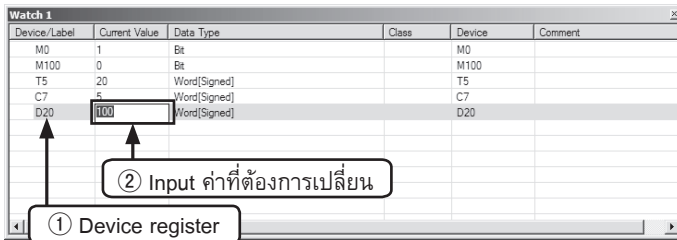


### ③ การเปลี่ยนค่าปัจจุบันและการ Force ON/OFF โดยใช้ Watch window

เป็นการใช้ Watch window ทำการ Force ON/OFF bit device หรือการเปลี่ยนค่าปัจจุบันของ Word device (T, C, D เป็นต้น)

เมื่อ Force ON/OFF input (X) ในตอนที่ PLC ทำงาน ก็จะมี ON/OFF ลงใน 1 scan cycle time

เมื่อต้องการตรวจสอบการทำงานของ Output (Y) ต้องให้ PLC อยู่ในสถานะ STOP ก่อน



- ① Input หมายเลข Device ในช่อง “Device/Label” แล้ว Register ลง Watch windows

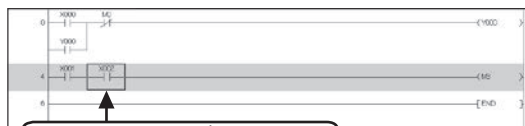
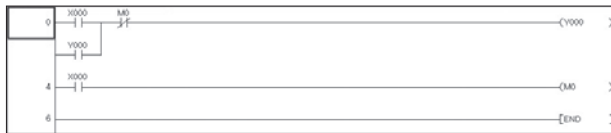
(เพิ่มเติม) หลังจาก Register แล้วเลือก [Online] → [Watch] → [Start Watching]

- ② Input ค่าที่ต้องการเปลี่ยนลงในช่อง “Current Value” แล้วกด

- Bit devices (เช่น X,Y)  
: Input ค่าเป็น ON “1”, OFF “0”
- Word devices (เช่น T, C, D)  
: Input ค่าที่อยู่ในขอบเขตที่สามารถ Input ได้

### 3.7.5 การ Writing program ขณะ PLC กำลัง RUN

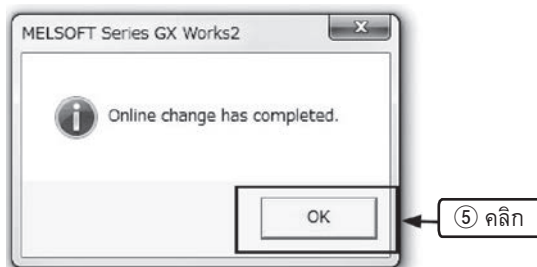
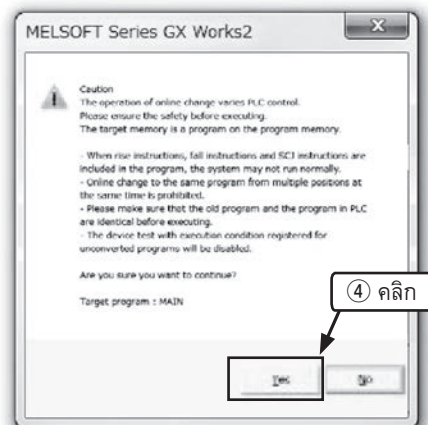
การเขียนบางส่วนของ Program ที่ต้องการแก้ไขลงใน PLC ในขณะที่ PLC กำลัง RUN เนื่องจากไม่ได้เป็นการ Write program ทั้งหมดลงไป จึงสามารถดำเนินการ Write ได้ในเวลาอันรวดเร็ว



Online program  
chang



[Shift] + [F4]



① อธิบายโดยใช้ตัวอย่างการเพิ่มหน้าสัมผัส (Contact) เข้าไปใน Circuit ตามภาพทางซ้ายมือ

② เพิ่มหน้าสัมผัส (Add Contact)  
กรอบสี่เหลี่ยมบริเวณวงจรจะปรากฏขึ้นมา

③ กด [Shift] + [F4] หรือเลือก [Convert/Compile] → [Online Program Change]

④ คลิก [Yes] เมื่อมีข้อความแจ้งเตือนเกี่ยวกับระบบความปลอดภัยของ PLC ขณะกำลัง RUN

⑤ จะปรากฏข้อความแจ้ง "Online change has completed" ขึ้นมาคลิก [OK]

#### Caution

- กรณีที่ Program ก่อนการแก้ไขใน PLC และ ใน PLC เป็นคนละตัวกันจะไม่สามารถดำเนินการได้ กรณีที่ไม่แน่ใจว่าเป็นตัวเดียวกันหรือไม่ ขอให้ทำการเปรียบเทียบดูก่อนหรือ ส่งไปที่ [Write to PLC]

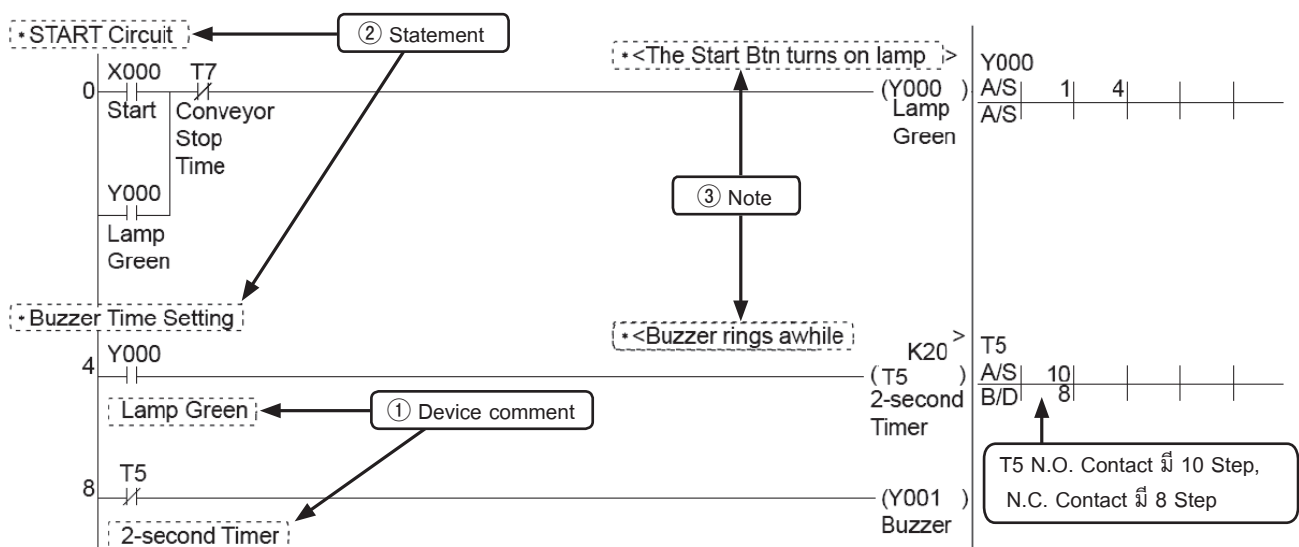
## 3.8 การใส่ Comment

### 3.8.1 ประเภทของ Comment

สามารถใส่ Comment ได้ 3 ประเภทตามด้านล่าง

ประเภท	วัตถุประสงค์	จำนวนตัวอักษร (ตัวพิมพ์ใหญ่)	หมายเหตุ
① Device comment	Comment แสดงบทบาทและหน้าที่การใช้งานของแต่ละ Device	16 (ที่ PLC มากสุด 8 ตัวอักษร)	กรณี Write ลง PLC จำเป็นต้องทำการ Set ค่า "Comments capacity" และจำเป็นต้อง Set "Comment range setting" ที่จะ write
② Statement	Comment แสดงบทบาทและหน้าที่การใช้งานของ Ladder blocks	32	เป็น Comment เฉพาะด้าน Computer software (ไม่ Download ลง PLC)
③ Note	Comment แสดงบทบาทและหน้าที่การใช้งานต่อ Output instructions	16	เป็น Comment เฉพาะด้าน Computer software (ไม่ Download ลง PLC)

[ตัวอย่าง Comment]



#### Point

#### วิธีการแสดง Comment

- ทำการเลือก [View] → [Display comment] จากเมนู แล้ว Comment จะปรากฏ
- เมื่อต้องการยกเลิกการแสดง Comment ให้ดำเนินการตามขั้นตอนด้านบนอีกครั้ง

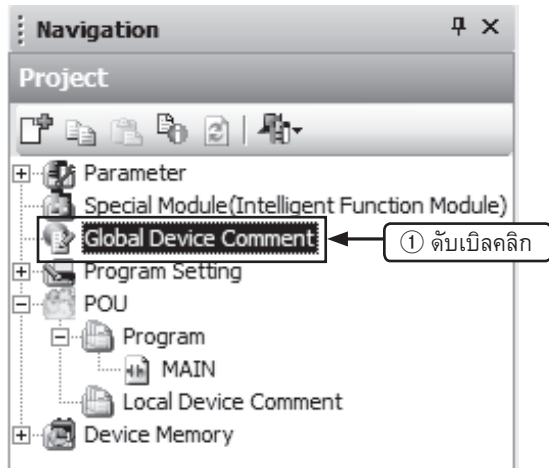
#### Point

#### Global device comment และ Local device comment ใน GX Works2

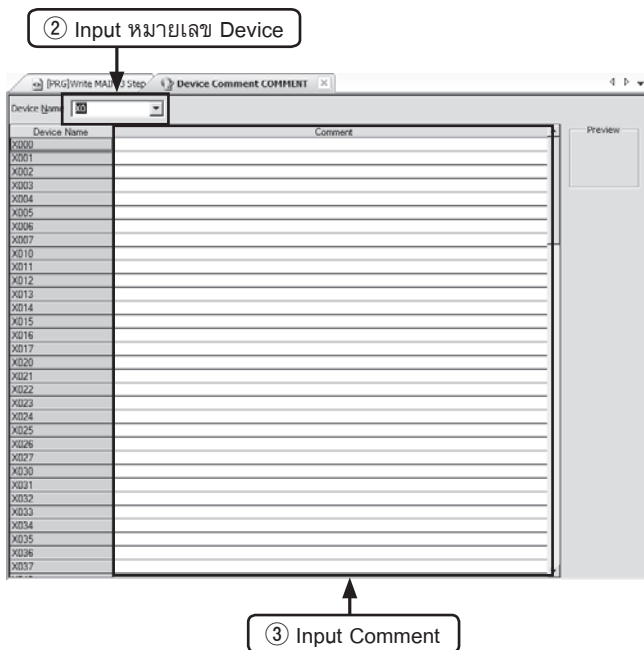
- Global device comment : เป็น Comment ที่สามารถ Input ลงใน PLC
- Local device comment : เมื่อทำการ Set local device comment ใหม่ จะสามารถจัดทำ Comment ต่างหากแยกกับ Global device comment  
และใน GX Developer สามารถใช้งาน Comment ที่แยกต่าง Program เช่น QnH, QnU แต่ใน FX Series ไม่สามารถ Write ลง PLC ได้  
การใช้งานปกติแนะนำให้เลือก "Global device comment"

## 3.8.2 วิธีการสร้าง Device comment

### ① วิธีการ Input device จาก List



① ทำการคลิก [Global Device Comment] จาก Project list




② ทำการ Input ตัวเลขต้นของ Device ที่ต้องการจัดทำ Comment ลงใน "Device Name" แล้วคลิก

③ ทำการ Input comments ลงในช่อง "Comment"

- กรณีที่ต้องการ Input comment ของ Device อื่นๆ ให้ทำการใส่หมายเลขของ Device นั้นตามขั้นตอนข้อ ②

### ② วิธีการป้อน Comment โดยใช้ Enter symbol



① คลิก  จาก Toolbar และ ดับเบิลคลิกหมายเลข Ladder ที่ต้องการ Input comment

② ทำการ Input comment ลงในหน้าจอ "Enter symbol" แล้วคลิก [OK]

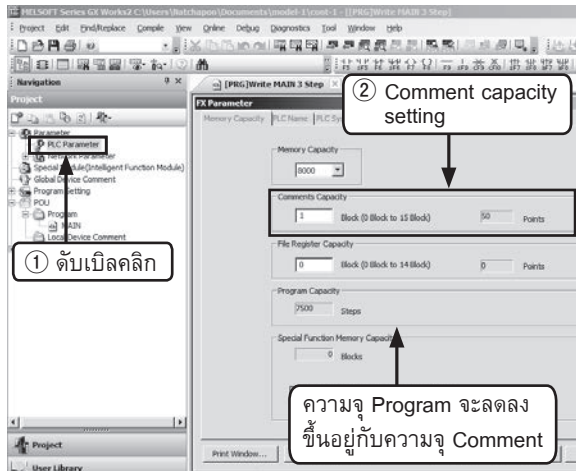
- เมื่อดำเนินการเสร็จแล้วให้คลิก  อีกครั้ง

## Point

### การ Set การ Write device comments ใน PLC

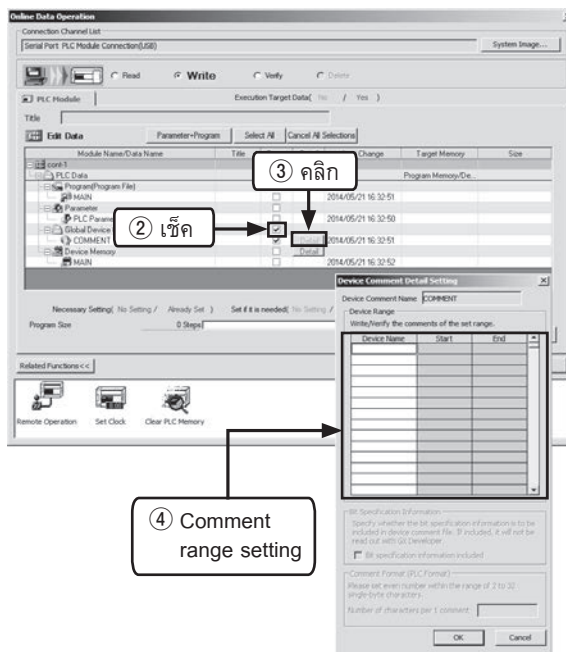
เพื่อทำการ Write comment ใน PLC จำเป็นต้อง Set “Parameter setting” และ “Comment range setting”

#### 1) การ Set parameter



- ① เลือก [Parameter] → [PLC parameter]
- ② ทำการ Set “จำนวน Block” ใน “Comment capacity”  
ใน 1 Block สามารถบรรจุได้ 50 Comment  
โดย Program memory ที่ 500 Step

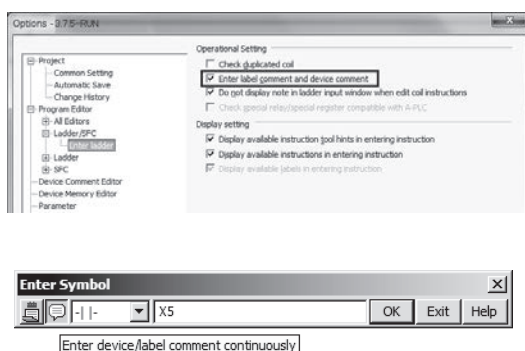
#### 2) Comment range setting



- ① เลือก [Online] → [Write to PLC]
- ② ทำการเช็คเครื่องหมายที่ “Global device comment”
- ③ คลิก [Details]
- ④ ทำการ Set ประเภท Device และขอบเขตที่จะ Write ที่ PLC ลงใน Setup comment range dialog

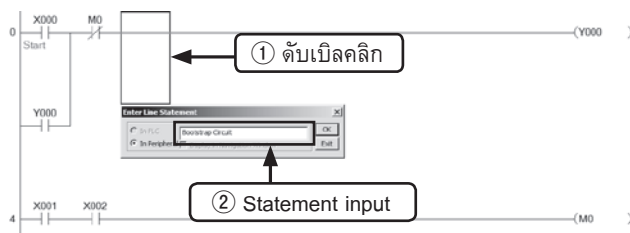
## อ้างอิง


### วิธีการป้อน Comment เมื่อจัดสร้าง Program (ตามวิธีการข้อ 2)



- เลือก [Tools] → [Options] จากนั้นเลือก [Program Editor] → [Ladder/SFC] → [Enter ladder] และคลิกใส่เครื่องหมายถูกใน [Enter label comment and device comment] หากกำหนดค่าตรงนี้จะทำให้นหน้าต่าง [Enter comment] ปรากฏขึ้นในขั้นตอนการกรอก Ladder เวลาเขียนโปรแกรม
- คลิก Icon ทางซ้ายมือเพื่อกรอก Comments

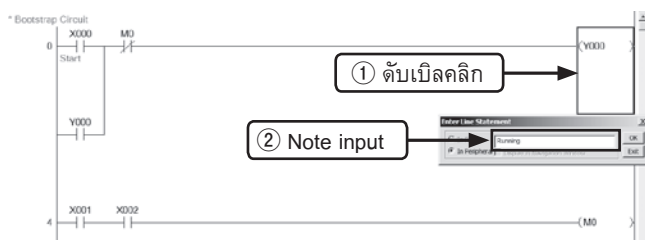
### 3.8.3 การจัดสร้าง Statements




- ① คลิก  จาก Toolbar ทำการดับเบิลคลิกตรงส่วนไหนก็ได้ของ Ladder block ที่ต้องการจะ Input statement
- ② ทำการ Input statement ที่หน้าจอ "Enter line statement" แล้วคลิก [OK]

• ในกรณีที่ดำเนินการเสร็จแล้ว ให้กด  อีกครั้ง

### 3.8.4 การจัดสร้าง Notes



- ① คลิก  จาก Toolbar ทำการดับเบิลคลิกตรง Output instruction ที่ต้องการจะ Input note
- ② ทำการ Input note ที่หน้าจอ "Enter Note" แล้วคลิก [OK]

• ในกรณีที่ดำเนินการเสร็จแล้ว ให้กด  อีกครั้ง

# คำสั่ง (Instructions) นั้นหรือไม่ยากหรอก เรื่องจิบ ๆ!

## บทที่ 4

### คำสั่งของซีควেনซ์ (Sequence instructions)

จนมาถึงเนื้อหาในบทนี้เราได้เรียนรู้เกี่ยวกับ...

ใน PLC นั้นประกอบไปด้วย Relay, Timer, Counter ต่างๆ มากมาย และข้างใน Sequence เหล่านั้นยังเชื่อมต่อกัน โดยทางด้าน Programming กับอุปกรณ์ต่างๆ ในแต่ละการทำงาน และการเดินสายไฟ สำหรับการเดินสายไฟจำเป็นต้องมีกฎเกณฑ์ เพื่อรองรับกับลักษณะการต่อ Coils ลักษณะของหน้าสัมผัส (Contact) ประเภทของ Coils ที่เหมาะสม ในส่วนนี้จะเรียกว่า “คำสั่ง” (Instruction) นั้นเอง รูปแบบของคำสั่ง (Instructions) ที่จัดสร้างขึ้นประกอบด้วย ‘ภาษาคำสั่ง (Instruction word) + Device number’ หรือ ที่ทำงานด้วย ภาษาคำสั่งเดี่ยวๆ (Individual instruction words)

ในบทนี้จะขออธิบายเกี่ยวกับคำสั่ง (Instruction) คำสั่งพื้นฐานต่างๆ เพื่อใช้จัดสร้าง Sequence program ขอให้ทำความเข้าใจของ แต่ละคำสั่ง (Instruction) ต่างๆ ให้ได้ด้วยนะ

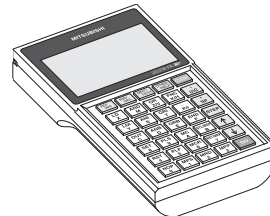
## 4.1 คำสั่ง (Instruction)

### 4.1.1 Commands และ Programs

อุปกรณ์ที่เกี่ยวข้องกับการ Input ซึ่ควนซ์โปรแกรม (Programing devices) มีแบบที่ไ้เขียนวงจรโปรแกรมในหน้าจคอมพิวเตอร์และแบบที่ไ้ Input โปรแกรมด้วยภาษาคำสั่ง (Instruction words) (อุปกรณ์สำหรับทำ List program) เหล่านี้เพียงแค่วิธีการ Input โปรแกรมต่างกัน เท่านั้น ส่วนตัวลักษณะโปรแกรมมีความคล้ายคลึงกัน



สำหรับ Input ด้วย Ladder diagrams  
(Programming software สำหรับคอมพิวเตอร์)



สำหรับ Input ด้วย Instruction words  
(เช่น Handy programming panel)

ตารางด้านล่างแสดง Instruction อำนวยความสะดวกให้ PLC และแสดง Ladder instruction แต่ละอัน

เครื่องหมาย, ชื่อเรียก	ความหมาย	Ladder
<b>LD</b> Load	หน้าสัมผัสปกติเปิด a (N.O. Contact) Bus connection	
<b>LDI</b> Load inverse	หน้าสัมผัสปกติปิด b (N.C. Contact) Bus connection	
<b>AND</b> And	หน้าสัมผัสปกติเปิด a (N.O. Contact) Series connection	
<b>ANI</b> And inverse	หน้าสัมผัสปกติปิด b (N.C. Contact) Series connection	
<b>OR</b> Or	หน้าสัมผัสปกติเปิด a (N.O. Contact) Parallel connection	
<b>ORI</b> Or inverse	หน้าสัมผัสปกติปิด b (N.C. Contact) Parallel connection	
<b>ANB</b> And block	Series connection between block	
<b>ORB</b> Or block	Parallel connection between block	
<b>OUT</b> Out	Coil drive instruction	
<b>SET</b> Set	Latch operation, Coil instruction	
<b>RST</b> Reset	Cancel latch operation, Coil instruction	
<b>NOP</b> NOP	No operationin	Delete program or space
<b>END</b> End	End of program	End of program    Return to step 0

● แสดงจุดหน้าสัมผัส (Contact) ใน Ladder จะเชื่อมต่อกันหรือไม่ขึ้นอยู่กับสถานะการ ON/OFF ของ Input relay, Output relay, Auxiliary relay, Timer, Counter  
○ แสดงการทำงานของ Coil



## 4.1.2 โครงสร้างของโปรแกรม

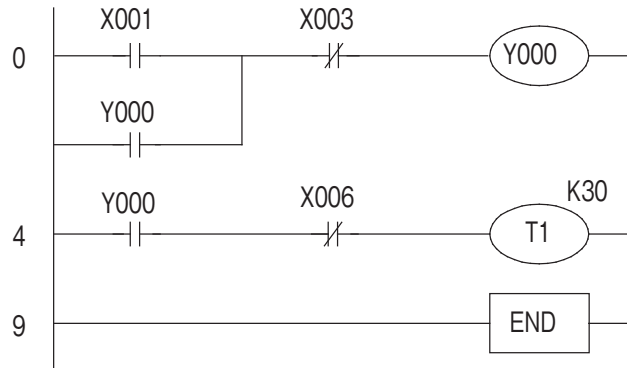
Internal sequence สำหรับ Sequence control จะต้องจัดทำ Circuit diagram (Ladder list) หรือ Instruction list เพื่อเป็น Sequence program

**Instruction list (Program list)**

Step No.	คำสั่ง (Instruction)	
	Instruction code	Device (Number) (Operand)
0	LD	X001
1	OR	Y000
2	ANI	X003
3	OUT	Y000
4	LD	Y000
5	ANI	X006
6	OUT	T1 K30
9	END	

Repeat operation

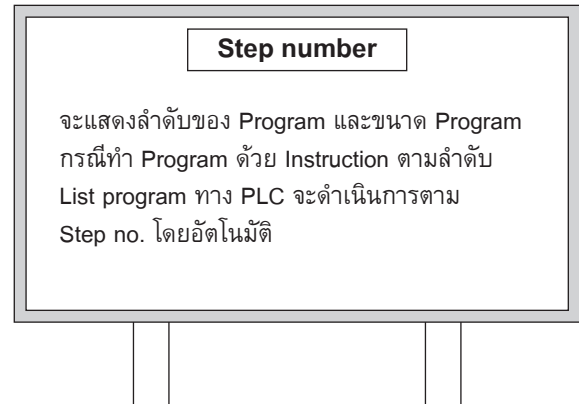
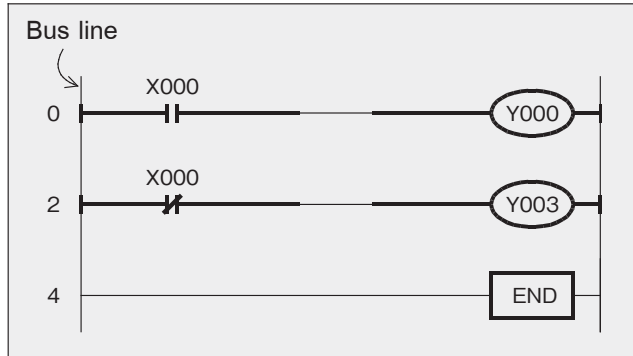
**Circuit diagram (Ladder diagram)**



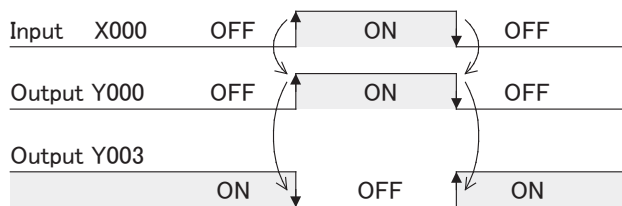
- Program จะสร้างจากการประกอบกันของโค้ดคำสั่ง (Instruction codes) และ Device number (Operand) จำนวนมาก แต่ละคำสั่ง (Instruction) จะมีการติดหมายเลขตามลำดับ ในที่นี้จะเรียกว่า Step no. (Step no. จะถูกควบคุมโดยอัตโนมัติ)
- [Instruction] จะเท่ากับ [Instruction code] + [Device number] แต่อาจมีบาง Instruction ที่มีเฉพาะ Instruction code ไม่มี Device หรือ นำเฉพาะ Instruction code มาแสดงเป็น Instruction อย่างง่าย ๆ
- Max step ขีดจำกัดที่สามารถจัดทำ Program ได้จะขึ้นอยู่กับ [Program memory capacity] ของ PLC ที่ใช้อยู่ ในที่นี้จะขอเรียกว่า PLC program capacity  
ยกตัวอย่าง Capacity ใน Program memory ของแต่ละรุ่น FX1s PLC [Step 2000], FX3G [Step 32000], FX3u [Step 64000]
- PLC repeatedly จาก Step 0 ถึง End step ในที่นี้จะเรียกว่า Cyclic operation ส่วนเวลาที่ใช้ใน 1 รอบการทำงานจะเรียกว่า Scan time Scan time นั้นจะแตกต่างกันขึ้นอยู่กับเนื้อหา Program หรือ ลำดับการ Cyclic operation จริง จะแตกต่างกันประมาณ ms ~ หน่วย 10 ms
- PLC program ที่จัดทำมาจาก Circuit diagram (Ladder diagram) จะใช้ความจุจาก Program memory ใน PLC ในรูปแบบ "Instruction list (Program list)"  
Programming software ที่ใช้กับคอมพิวเตอร์ จะทำการ Conversion ระหว่าง "Instruction list (Program list)", Circuit diagram (Ladder diagram)

### 4.1.3 หน้าสัมผัสปกติเปิด a (N.O. Contact), หน้าสัมผัสปกติปิด b (N.C. Contact), Out instruction, End instruction

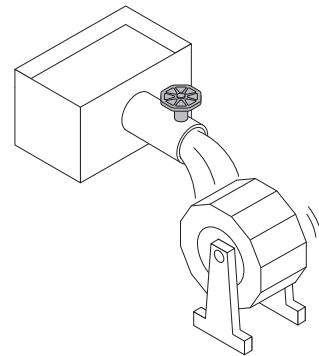
#### Circuit program



#### จาก Program ด้านบน...



Input X000 ON Output Y000 ON  
Input X000 OFF Output Y003 ON



#### อ้างอิง

#### การ Programming โดย Instruction list

**LD**

**Load**  
คำสั่งสำหรับ Bus connection  
สำหรับหน้าสัมผัสปกติเปิด a  
(N.O. Contact)

**LDI**

**Load inverse**  
คำสั่งสำหรับ Bus connection  
สำหรับหน้าสัมผัสปกติปิด b  
(N.C. Contact)

**OUT**

**Out**  
คำสั่งเคลื่อนที่ Coil

**END**

**End**  
คำสั่งใช้สำหรับเมื่อสิ้นสุดโปรแกรม

#### List Program

Step	Instruction
0	LD X000
1	OUT Y000
2	LDI X000
3	OUT Y003
4	END

หน้าสัมผัสปกติเปิด a (N.O. Contact) ที่ใช้ในช่วงแรกของ Bus line ใช้ Instruction LD (Load) ส่วนหน้าสัมผัสปกติปิด b (N.C. Contact) ใช้ LDI (Load inverse) Contact instruction เช่น LD หรือ LDI สามารถใช้ Device input relay X, Output relay Y, Timer T, Counter C, Auxiliary relay M ได้ Coil drive instruction เช่น OUT จะใช้ Device อื่นๆ ได้ ยกเว้น Input relay X

## 4

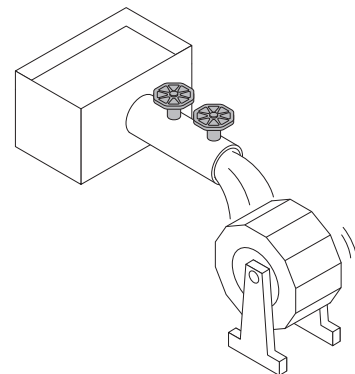
```

graph LR
    subgraph Network [Network 0]
        X000[NO X000] --- X001[NO X001]
        X001 --- X002[NO X002]
        X002 --- Y000[(Y000)]
        X001 --- Branch(( ))
        Branch --- X003[NO X003]
        X003 --- Y003[(Y003)]
    end
    subgraph Network6 [Network 6]
        END[END]
    end

```

The timing diagram illustrates the following sequence of events:

- Input X000** transitions from OFF to ON.
- Output Y000** transitions from OFF to ON.
- Input X001** transitions from OFF to ON.
- Input X002** transitions from OFF to ON.
- Output Y003** transitions from OFF to ON.
- Input X002** transitions from ON to OFF.
- Output Y000** transitions from ON to OFF.
- Input X001** transitions from ON to OFF.
- Input X000** transitions from ON to OFF.
- Output Y003** transitions from ON to OFF.
- Input X002** transitions from OFF to ON.
- Output Y000** transitions from OFF to ON.
- Input X003** transitions from OFF to ON.
- Output Y003** transitions from OFF to ON.
- Input X003** transitions from ON to OFF.
- Output Y003** transitions from ON to OFF.



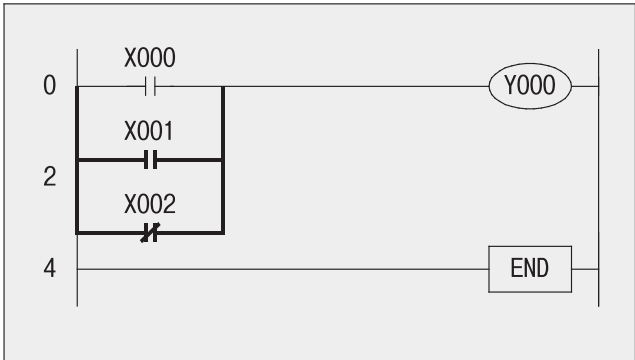
เมื่อ Input X000 ON, X001 ON, X002 OFF, X003 ON จะทำให้ Output Y003 ON

## Programming โดย Instruction list

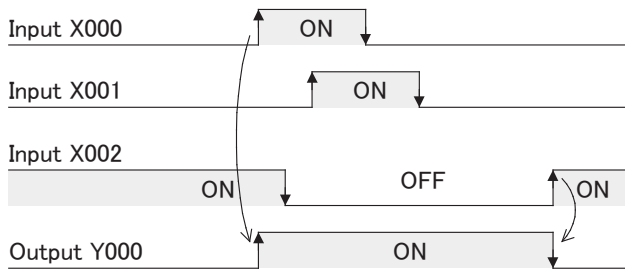
Y000, Y003 จะต้องกำลังทำงานอยู่ และ Series contact จะต้องเชื่อมต่อกันทั้งหมด

### 4.1.5 Parallel connection

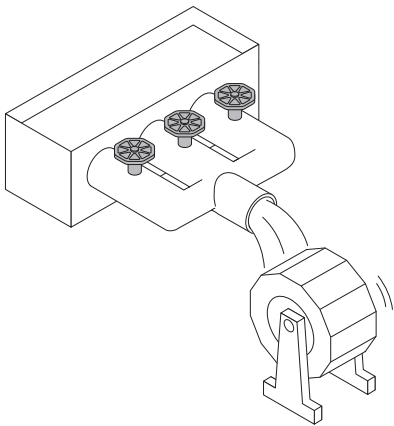
#### Circuit program



#### จาก Program ด้านบน...



ถ้า Input X000 ON, X001 ON, X002 OFF อย่างใดอย่างหนึ่งเป็นตามเงื่อนไขที่กล่าวมา จะทำให้ Output Y000 ON



#### อ้างอิง

#### Programming โดย Instruction list

**OR**

Or  
คำสั่ง Parallel connection instruction  
สำหรับหน้าสัมผัสปกติปิด a  
(N.O. Contact)

**ORI**

Or inverse  
คำสั่ง Parallel connection instruction  
สำหรับหน้าสัมผัสปกติเปิด b  
(N.C. Contact)

#### List program

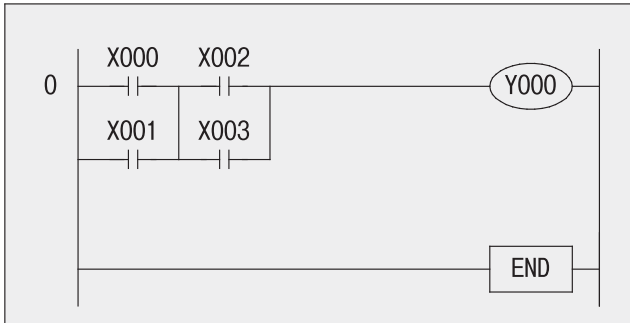
Step	Instruction
0	LD X000
1	OR X001
2	ORI X002
3	OUT Y000
4	END

OR (OR) instruction จะใช้สำหรับ N.O. Contact ส่วน ORI (OR inverse) instruction จะใช้สำหรับ N.C. Contact ในการ connect กับ parallel ตาม LD, LDI instruction ส่วน Output Y000 ด้านบน เมื่อมีการ connect เพียง 1 contact ก็จะทำงานเมื่อมีการ conduction

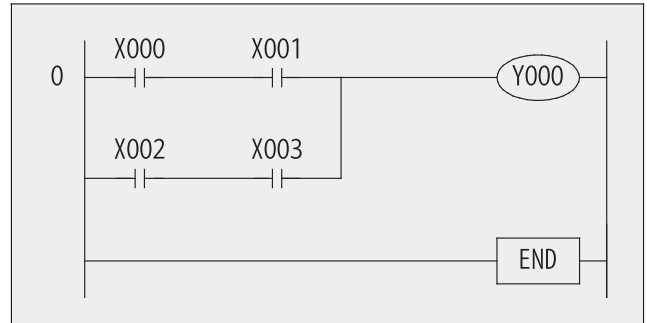
## 4.1.6 Series และ Parallel connection

### Circuit program

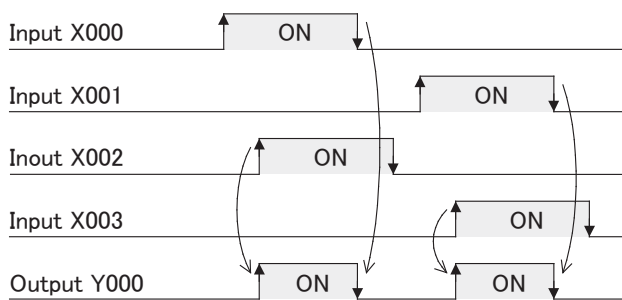
ตัวอย่าง Program ①



ตัวอย่าง Program ②

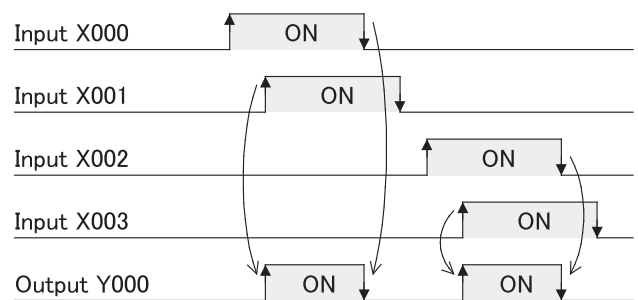


จาก Program ด้านบน...



Input X000 หรือ Input X001 และ X002 หรือ X003 อันใดอันหนึ่ง ON Y000 จะเริ่มทำงาน

จาก Program ด้านบน...



X000 และ X001 ON พร้อมกัน หรือ X002 และ X003 ON พร้อมกัน Output Y000 จะทำงาน

4

### อ้างอิง

### Programming โดย Instruction list

ตัวอย่าง Program ①

#### List program

Step	Instruction
0	LD X000
1	OR X001
2	LD X002
3	OR X003
4	ANB
5	OUT Y000
6	END

ตัวอย่าง Program ②

#### List program

Step	Instruction
0	LD X000
1	AND X001
2	LD X002
3	AND X003
4	ORB
5	OUT Y000
6	END

รายละเอียด ANB, ORB กรุณาดูในภาคผนวก

## 4.1.7 SET instruction, RST instruction

# SET

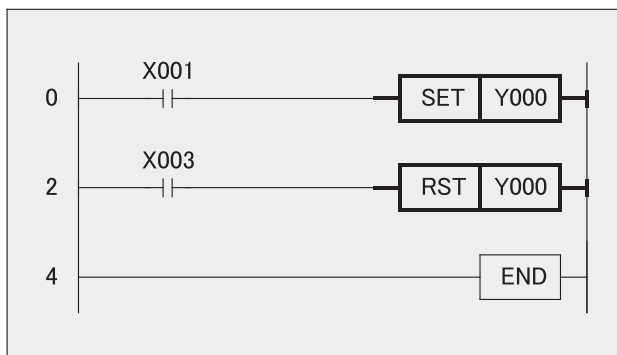
**SET**  
คำสั่ง Output  
คงสภาพการทำงาน

# RST

**Reset**  
คำสั่ง Cancel  
การคงสภาพการทำงาน

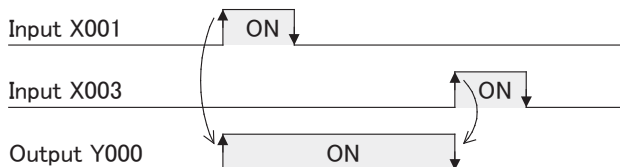
SET/RST instruction จะใช้กับ เซ็น Output relay Y,  
Auxiliary relay M  
ในส่วนอื่นๆ เช่น Counter หรือ Timer ก็ใช้ RST instruction

### Circuit program



เป็นคำสั่งเพื่อขับเคลื่อน Coil เหมือนกับ OUT instruction  
สำหรับ OUT instruction เมื่อ Contact drive coil เปลี่ยนจาก  
ON เป็น OFF, OUT instruction ที่สั่ง Coil ก็จะเปลี่ยนเป็น OFF  
ด้วย เมื่อใช้ Instruction SET ถึง Contact จะเปลี่ยนจาก ON เป็น  
OFF coil ก็ยังจะ Stand by ON อยู่อย่างนั้น  
ใช้คำสั่ง RST (Reset) เพื่อทำการเปลี่ยนการทำงานของ Coil ที่สั่ง  
ด้วย SET instruction จาก ON เป็น OFF

### จาก Program ด้านบน...



เมื่อ Input X001 ON Output Y000 จะ ON ด้วย หลังจากนั้น  
ถึง Input X001 ON จะ OFF Y000 ก็จะ Stand by ON อยู่อย่างนั้น  
เมื่อ Input X003 ON Output Y000 จึงจะ OFF

### ● วิธีการ Input

[ ] SET Y0  
F8 Space

[ ] RST Y0  
F8 Space

### อ้างอิง

### Programming โดย Instruction list

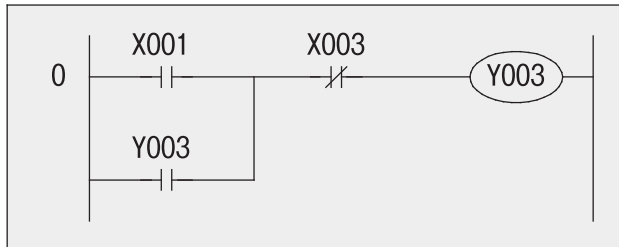
#### List program

Step	Command
0	LD X001
1	SET Y000
2	LD X003
3	RST Y000
4	END

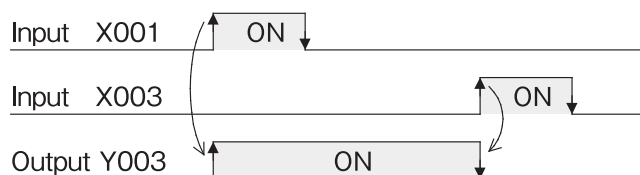
#### 4.1.8 Latch circuit (วงจรคงสภาพตัวเอง)

สามารถทำการ Hold หรือ Release output ได้โดย การ Programming self-hold sequencer

##### Circuit program



##### จาก Program ด้านบน...



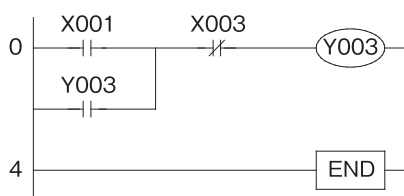
- เมื่อ X001 ON, X003 OFF Y003 จะ ON
- ถึง X001 จะ OFF แต่ Y003 จะยังทำงานต่อไป (Self-hold) เมื่อ X003 ON จึงจะเปลี่ยนเป็น OFF

4

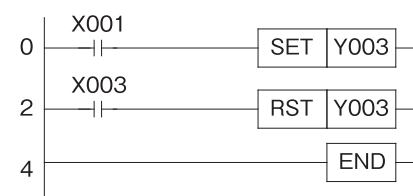
##### อ้างอิง

##### Circuit ทั้งคู่ทำการแบบเดียวกัน

###### [ Output latch circuit ]



###### [ Output SET/RST circuit ]



การทำงานเหมือนกัน

เมื่อใช้ SET instruction ถึงแม้ว่า Contact coil drive จะ OFF แต่ Output จะถูก Hold ON ไว้  
ดังนั้น Output เดียวกันจะสามารถ Programming ก็ครั้งก็ได้ โดยไม่ใช้ Double coil และยังสามารถใช้งานได้อย่างดีในการควบคุม Output จากจุดต่างๆ ของ Program

##### อ้างอิง

##### Programming โดย Instruction list

###### [ Output latch circuit ]

Step	Instruction
0	LD X001
1	OR Y003
2	ANI X003
3	OUT Y003
4	END

###### [ Output SET/RST circuit ]

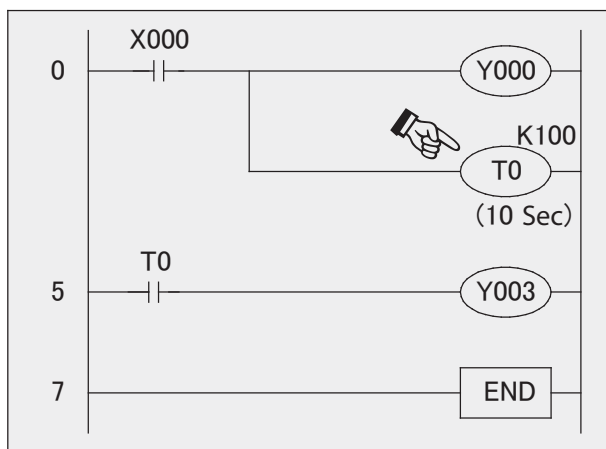
Step	Instruction
0	LD X001
1	SET Y003
2	LD X003
3	RST Y003
4	END

## 4.2 Timer circuit

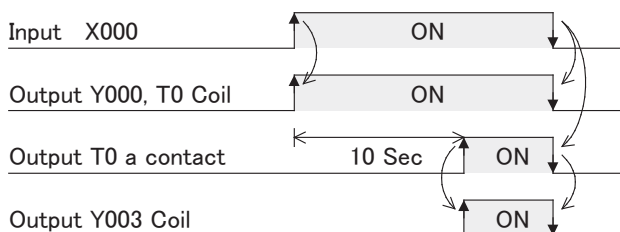
### Timers

Timers นั้นมีหลากหลายประเภท ในที่นี้จะขออธิบายถึงวิธีการใช้งานของ Digital timer ของ Micro PLC

### Circuit program



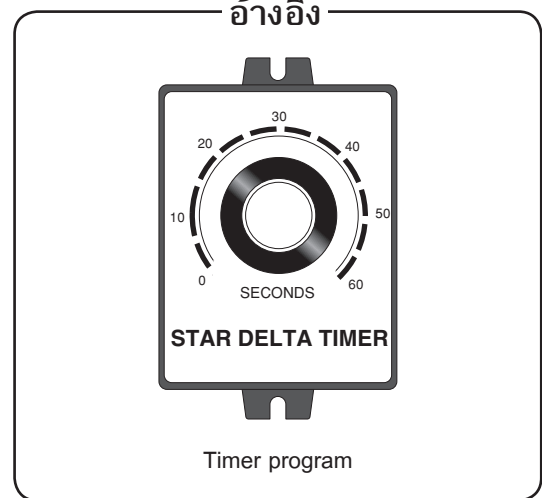
### จาก Program ด้านบน...



`( ) T0 K100`  
`F7` Space

\*1 :100 ms (0.1 sec) หน่วยของ Timer

### อ้างอิง



Timer program

- Timer contact หลังจาก Coil ทำงาน จะเริ่มทำงาน หลังจาก Deley time ที่กำหนดไว้ (On delay timer) เราจะเรียกเวลาที่กำหนดไว้ว่า Set value จะแสดงสัญลักษณ์เป็น K ค่า K ที่จะ Set นั้น สามารถ Set ได้ตั้งแต่ 1 - 32,767 ยกตัวอย่าง K100 Timer จะเท่ากับ 10 sec <sup>\*1</sup>
- เมื่อ X000 OFF ในขณะที่ Timer ทำงาน ค่าปัจจุบันของ Timer จะกลับไป 0 Timer contact ก็ OFF
- วิธีการ Input

### อ้างอิง

### Programming โดย Instruction list

#### List program

Step	Instruction
0	LD X000
1	OUT Y000
2	OUT T0 K100

Step	Instruction
5	LD T0
6	OUT Y003
7	END

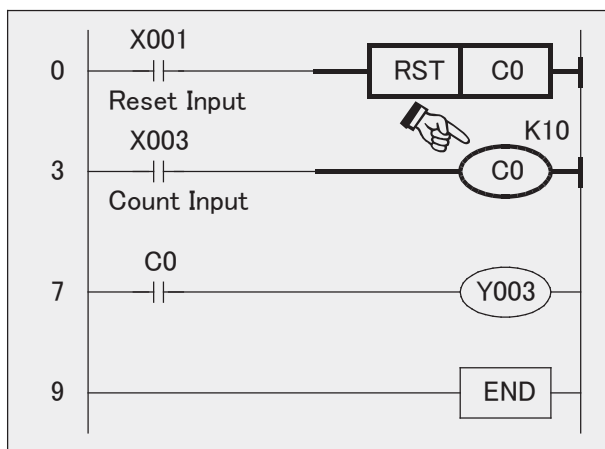


## 4.3 Counter circuit

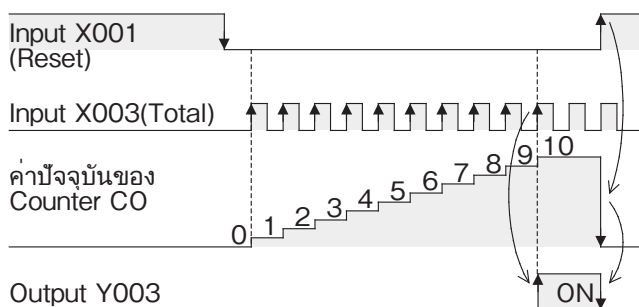
### Counters

Counter นั้นมีหลากหลายประเภท ในที่นี้จะขออธิบายถึงวิธีการใช้งาน Program ของ Counters ทั่วไปของ Micro PLC

#### Circuit program

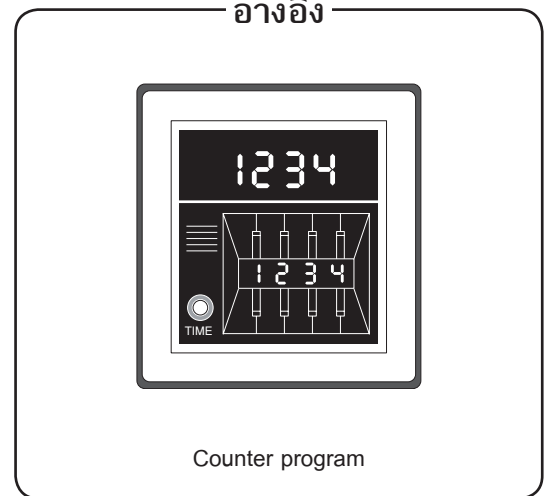


#### จาก Program ด้านบน...



( ) C0 K10  
F7 Space

อ้างอิง



Counter program

- สำหรับ Counters เมื่อ Contact (X003) เปลี่ยนจาก OFF → ON จำนวนจะถูกนับ  
Contact (X003) เราจะเรียกว่า Count input ส่วนตัวเลขที่ถูกนับโดย Counters เราจะเรียกว่า "Present value"  
ส่วน Counter contact เมื่อค่าปัจจุบันวิ่งไปถึง ค่าที่กำหนดไว้ก็จะเริ่มทำงาน  
ค่าที่กำหนดนั้น สามารถกำหนดได้ ตั้งแต่ 1 - 32,767
- หลังจาก Count up แล้ว ค่าปัจจุบันของ Counter จะไม่มีการเปลี่ยนแปลง Output contact ก็จะทำงานเหมือนเดิม
- เมื่อ Reset input X001 ON แล้ว ค่าปัจจุบัน ของ Counter จะเป็น 0 Counter contact ก็จะ OFF
- วิธีการ Input

4

อ้างอิง

#### Programming โดย Instruction

##### List program

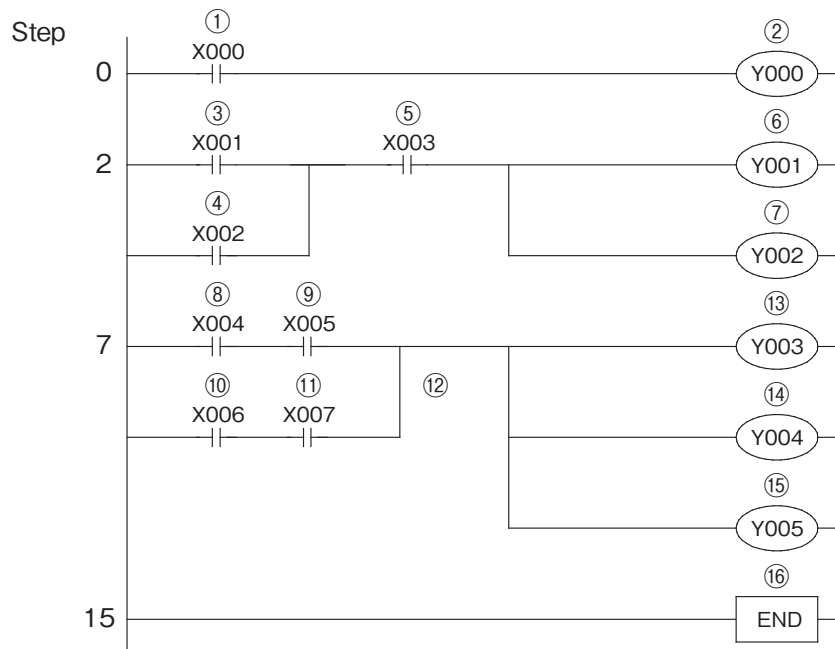
Step	Instruction
0	LD X001
1	RST C0
3	LD X003
4	OUT C0 K10

Step	Instruction
7	LD C0
8	OUT Y003
9	END

## 4.4 ลำดับ Program (Order of program)

### ลำดับ Program คือ

Circuit diagram จะเรียงลำดับจากซ้ายไปขวา จากบนลงล่าง



Program ด้านบนนี้ จะเรียงลำดับ Program ① - ⑯

อ้างอิง		Programming โดย Instruction list	
ลำดับ	Step	Instruction	
①	0	LD	X000
②	1	OUT	Y000
③	2	LD	X001
④	3	OR	X002
⑤	4	AND	X003
⑥	5	OUT	Y001
⑦	6	OUT	Y002
⑧	7	LD	X004
⑨	8	AND	X005
⑩	9	LD	X006
⑪	10	AND	X007
⑫	11	ORB	
⑬	12	OUT	Y003
⑭	13	OUT	Y004
⑮	14	OUT	Y005
⑯	15	END	

## บทที่ 5

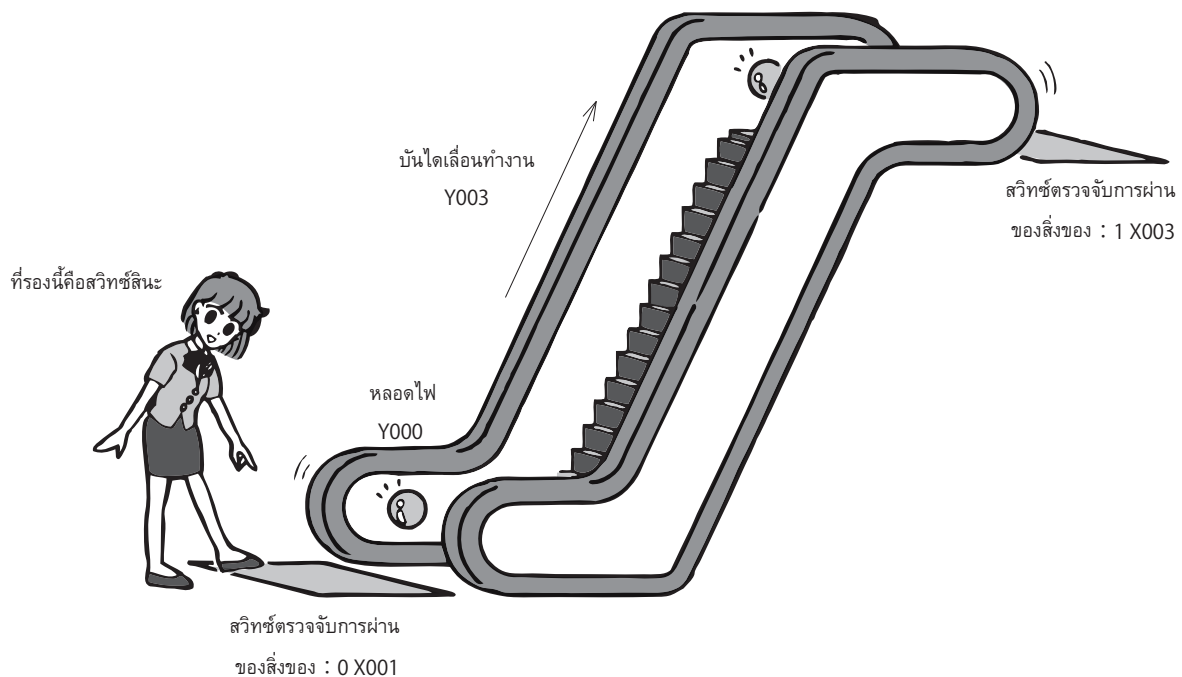
### การฝึกการเขียนโปรแกรม

---

เอาละเราหยุดเรื่องของทฤษฎีไว้แค่นี้ก่อนดีกว่า...

ในบทนี้จะเน้นการสอนเรื่องการใช้งานทั่วไป เช่น Programming และการ Monitoring โดยยึดตามตัวอย่างง่ายๆ ดังต่อไปนี้

## 5.1 ตัวอย่างการนำไปใช้งาน 1 (การควบคุมบันไดเลื่อน)



ลองคิดวิธีการทำ Program PLC โดยใช้ระบบบันไดเลื่อนนี้เป็นตัวอย่างกัน

### 《I/O assignment (การกำหนด I/O)》

Input	
X001	สวิทช์ตรวจจบการผ่านของสิ่งของ : 0
X003	สวิทช์ตรวจจบการผ่านของสิ่งของ : 1

Output	
Y000	หลอดไฟแสงส่องสว่าง
Y003	Output ให้บันไดเลื่อนทำงาน

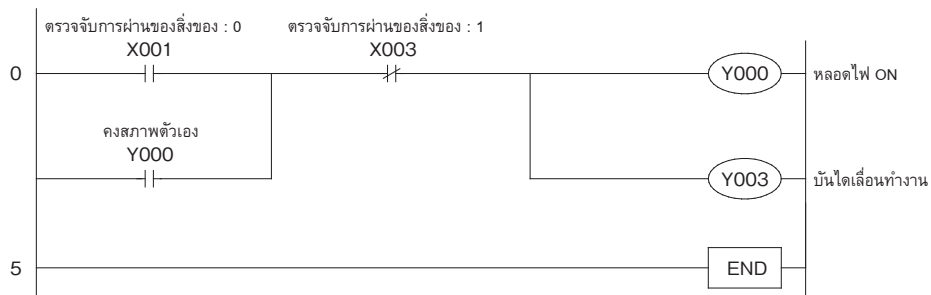
### 《ระบบการควบคุม》

- ① บันไดเลื่อนจะไม่ทำงานจนกว่าจะมีคนเข้ามาใกล้
- ② เมื่อมีคนเข้ามาใกล้ สวิทช์ตรวจจบการผ่านของสิ่งของ : 0 จะ ON จากนั้นไฟจะติด (เฉพาะบันไดเลื่อนขึ้นเท่านั้น)
- ③ เมื่อคนขึ้นบันไดเลื่อนจนสุดทางเลื่อน สวิทช์ตรวจจบการผ่านของสิ่งของ : 1 จะ ON ทั้งไฟและบันไดเลื่อนจะหยุดทำงาน

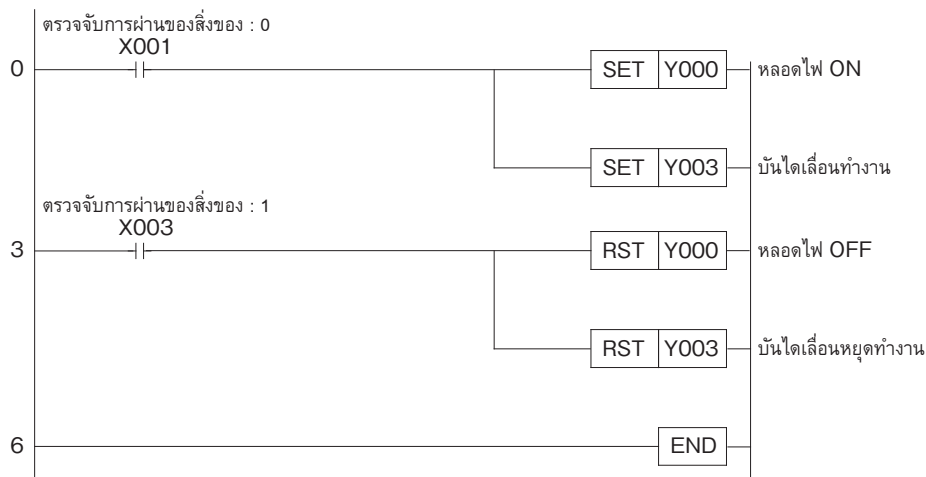
จากด้านบน ขอให้คิดว่าสวิทช์ตรวจจบการผ่านของสิ่งของทั้ง 0 และ 1 คือ สวิทช์แบบ Non-hold type (แบบไม่คงสภาพตัวเอง)  
ส่วนคนที่ใช้บันไดเลื่อน ก็ได้ขึ้นบันไดเลื่อนแบบต่อเนื่อง

## สามารถสร้างโปรแกรม (Program) ได้ดังนี้

### 《Circuit program : ตัวอย่างที่ 1》



### 《Circuit program : ตัวอย่างที่ 2》

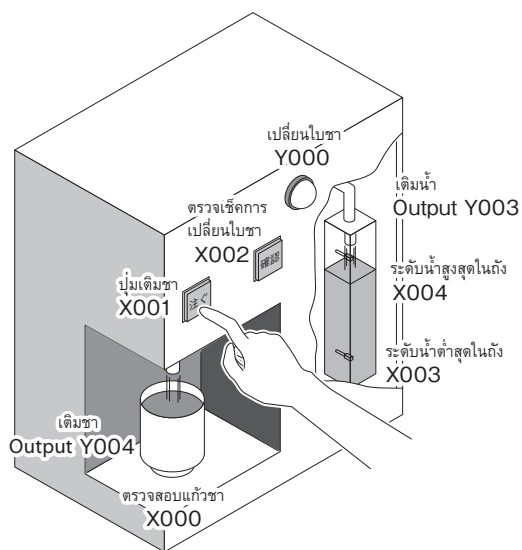


### 《ต่อไปมาลองตรวจสอบขั้นตอนการทำงานของ Program》

เราลองนำ Program ด้านบน มาเขียนลงใน PLC แล้วลองตรวจสอบขั้นตอนการทำงานของ Program

เมื่อทำการ ON Input X001, Output Y000, Y003 จะทำงาน หลังจากนั้น Input X003 ON, Output Y000, Y003

## 5.2 ตัวอย่างการนำไปใช้งาน 2 (เครื่องชงชา)



เรามาคิด Program PLC โดยใช้เครื่องชงชาเป็นตัวอย่างกัน

### 《 I/O assignment (การกำหนด I/O) 》

Input	
X000	ตรวจสอบแก้วชา
X001	ปุ่มกดเติมน้ำ
X002	ปุ่มตรวจสอบใบชา
X003	ระดับน้ำต่ำสุดในถัง
X004	ระดับน้ำสูงสุดในถัง

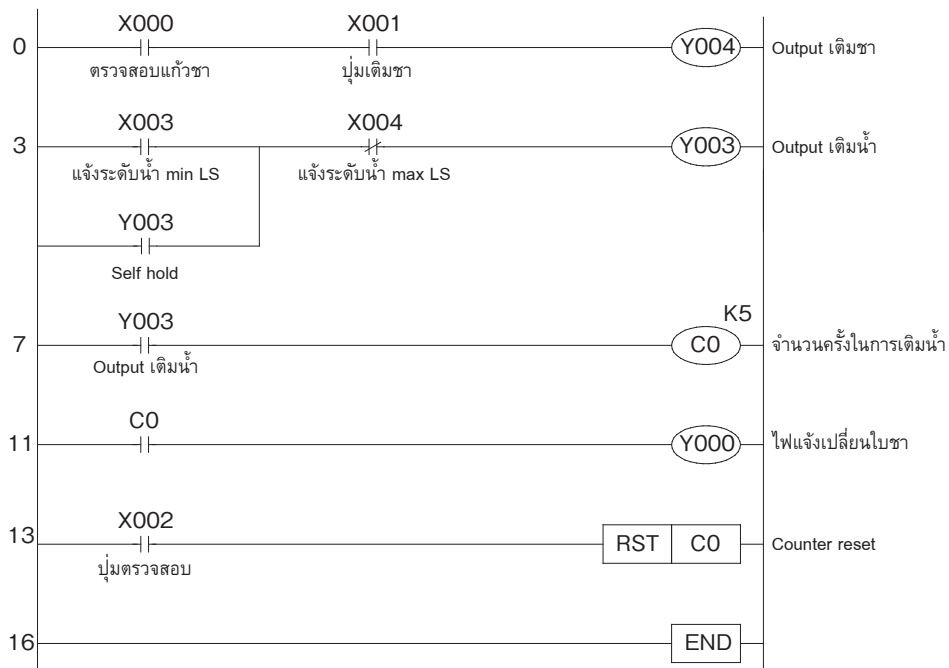
Output	
Y000	ไฟแสดงสถานะการเปลี่ยนใบชา
Y003	Output การเติมน้ำ
Y004	Output การเติมน้ำ

### 《ระบบการควบคุม》

- ① เมื่อ X000 การตรวจสอบแก้วชา ON อยู่ แล้วเรากดปุ่มเติมน้ำ X001 (X001 ON), Output การเติมน้ำ Y004 จะทำงาน น้ำร้อนจะไหลลงในแก้ว น้ำร้อนจะไหลเมื่อเรากดปุ่มเท่านั้น เมื่อเราเอามือออก น้ำร้อนจะหยุดไหล และถ้า X000 เป็น OFF (หรือตรวจสอบไม่พบแก้วชา) ถึงเราจะกดปุ่มเติมน้ำ X001 น้ำร้อนก็จะไม่ไหล
- ② เมื่อน้ำในถังลดลง ตัวแจ้งระดับน้ำ LS X003 จะ ON, Output ตัวเติมน้ำ Y003 จะทำงาน เมื่อ Output ตัวเติมน้ำ Y003 ทำการเติมน้ำจนตัวแจ้งระดับน้ำ LS X004 ON, Output ตัวเติมน้ำ Y003 จะหยุดทำงาน
- ③ เมื่อตัวเติมน้ำทำงาน 5 ครั้ง ไฟแจ้งการเปลี่ยนใบชาจะทำงาน
- ④ เมื่อกดปุ่มตรวจสอบ ไฟแจ้งการเปลี่ยนใบชาจะดับ

## สามารถสร้างโปรแกรม (Program) ได้ดังนี้

### 《Circuit program》

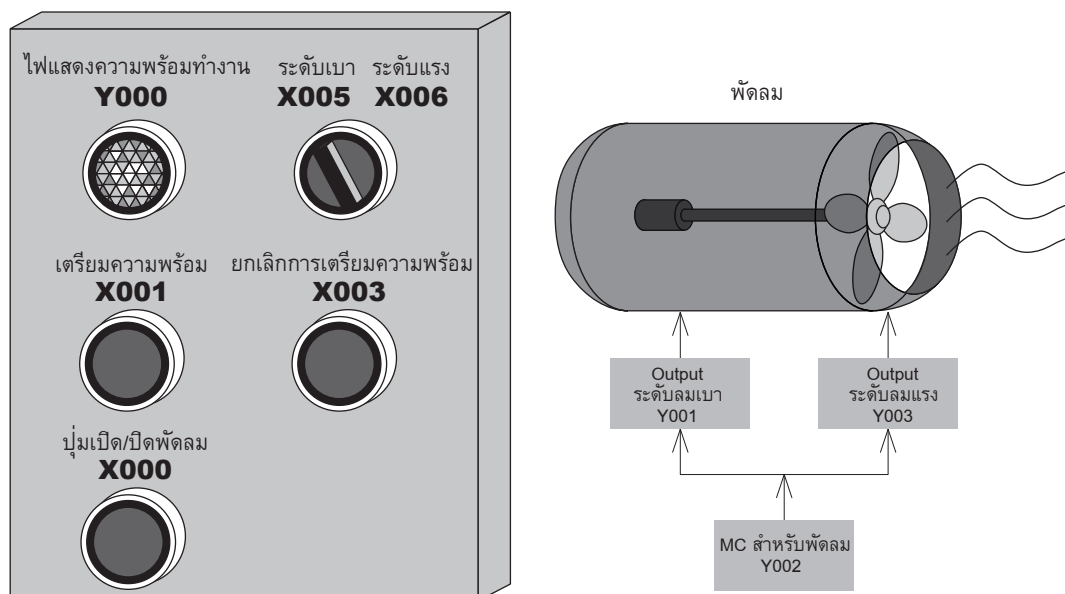


### 《ต่อไปมาลองตรวจสอบขั้นตอนการทำงานของ Program》

เราลองนำ Program ด้านบน มา Input ลงใน PLC แล้วลองตรวจสอบขั้นตอนการทำงานของ Program

- ① เมื่อ Input X000 หรือ X001 ON, Output Y004 จะทำงาน
- ② เมื่อ Input X003 ON, Output Y003 จะทำงาน เมื่อ Input X004 ทำงาน Output Y003 จะหยุดทำงาน
- ③ ทุกครั้งที่ Output Y003 ทำงาน Counter C0 จะเพิ่มขึ้น 1 ครั้ง เมื่อ Counter ครบ 5 ครั้ง Y000 จะทำงาน
- ④ เมื่อ Input X002 ON, Counter C0 จะเป็น 0 แล้ว Output Y000 จะหยุดทำงาน

## 5.3 ตัวอย่างการนำไปใช้งาน 3 (การควบคุมพัดลม)



เรามาลองคิด Program PLC โดยใช้ระบบควบคุมพัดลมเป็นตัวอย่าง

### 《I/O assignment (การกำหนด I/O)》

Input	
X000	ปุ่มเปิด/ปิดพัดลม
X001	สวิตช์เตรียมความพร้อม
X003	สวิตช์ยกเลิกการเตรียมความพร้อม
X005	เลือกกระดัดเบา
X006	เลือกกระดัดแรง

Output	
Y000	ไฟแสดงความพร้อม
Y001	Output ลมระดับเบา
Y002	MC สำหรับพัดลม
Y003	Output ลมระดับแรง

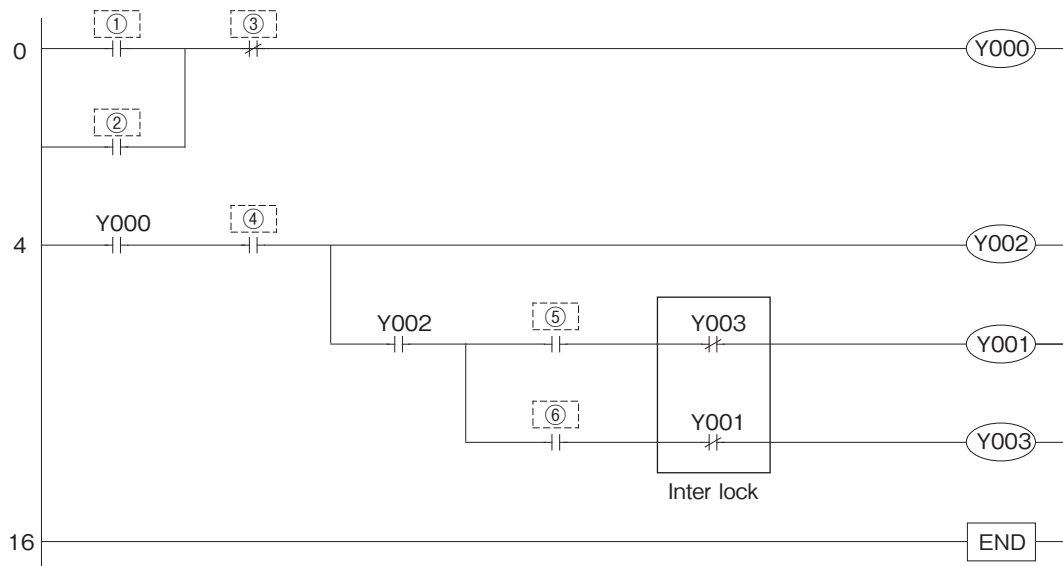
### 《ระบบการควบคุม》

- ① เมื่อสวิตช์เตรียมความพร้อม (X001) ON ไฟแสดงความพร้อม (Y000) จะติด ON, Self-hold จะทำงาน และเมื่อสวิตช์ยกเลิกการเตรียมความพร้อม (X003) ON ไฟแสดงความพร้อม (Y000) จะดับ OFF, Self-hold เลิกทำงาน
- ② ไฟแสดงความพร้อมติด (Y000) ON พัดลมจะเริ่ม/หยุด การทำงานตาม MC สำหรับพัดลม ซึ่งสั่งการเปิด/ปิดการทำงานตามสวิตช์เปิด/ปิดพัดลม (X000) ส่วนแรงลมจะเลือกจากสวิตช์ (X005/X006)
  - กรณีที่ X005 ON : Output แรงลมระดับเบาจะ ON (Y001)
  - กรณีที่ X006 ON : Output แรงลมระดับแรงจะ ON (Y003)



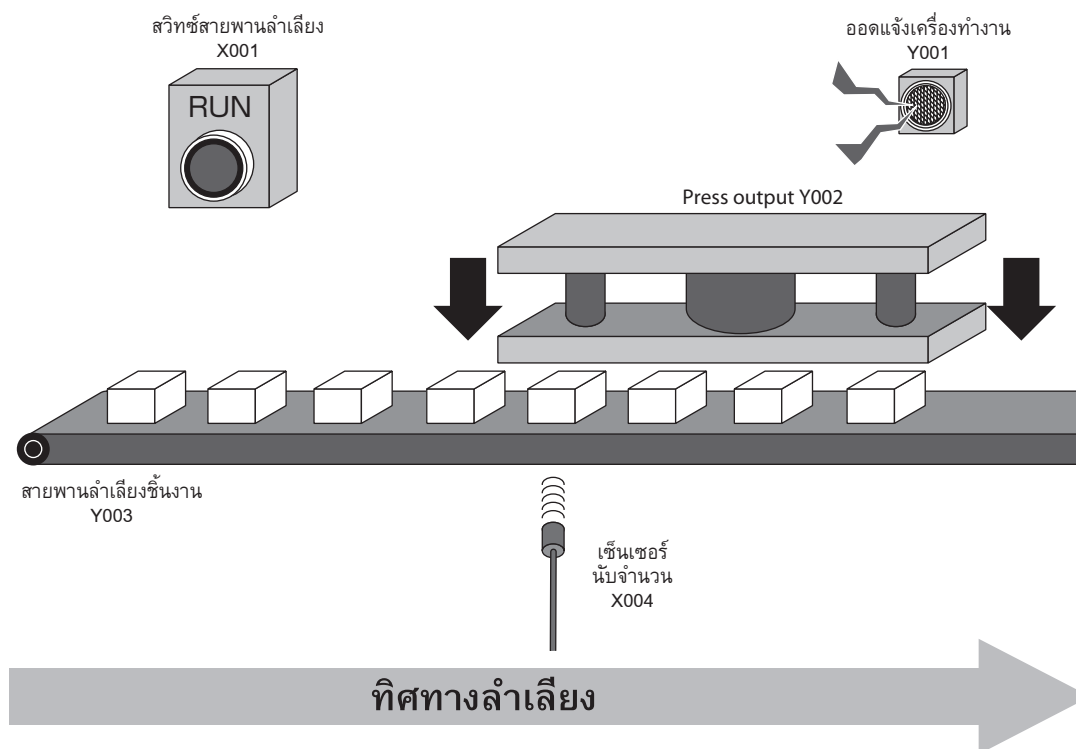
## 《Circuit program》

กรุณาใส่ Device ที่หายไปลงในช่องว่าง และทำให้โปรแกรมสมบูรณ์



- เผลย ดูท้ายบท
- List program กรุณาดูในภาคผนวก

## 5.4 ตัวอย่างการนำไปใช้งาน 4 (การควบคุมของเครื่อง Press)



เรามาลองคิด Program PLC โดยใช้ระบบควบคุมเครื่อง Press เป็นตัวอย่าง

### 《I/O assignments (การกำหนด I/O)》

Input	
X001	สวิตช์สายพานลำเลียง
X004	เซ็นเซอร์นับจำนวน

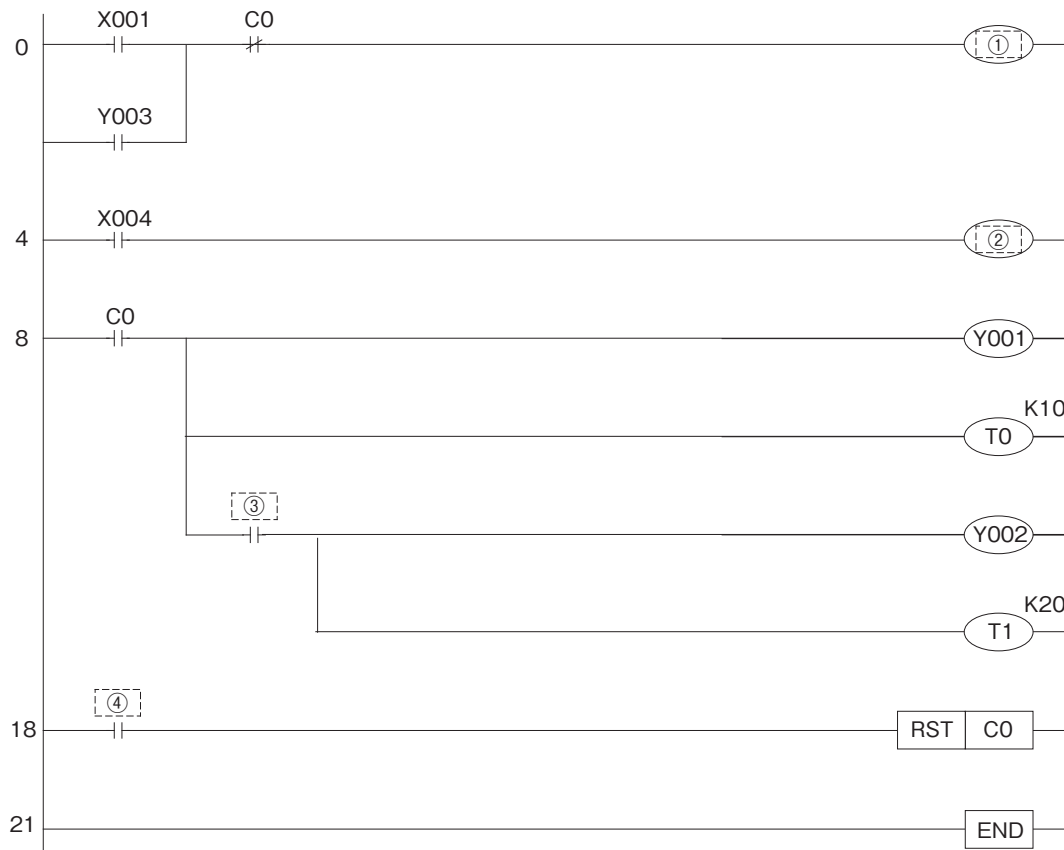
Output	
Y001	ออกแจ้งเครื่องเริ่มทำงาน
Y002	Press output
Y003	สายพานลำเลียงชิ้นงาน

### 《ระบบการควบคุม》

- ① สายพานลำเลียงชิ้นงานจะ ON (Y003) เมื่อสวิตช์สายพานลำเลียง (X001) ON อยู่ และจะทำการส่งงานไปยังตำแหน่ง Press เมื่อเซ็นเซอร์ (X004) ทำการตรวจจับงานงานที่ผลิตเสร็จแล้วครบ 4 ตัว สายพานสำหรับส่งงาน (Y003) จะ OFF และหยุดทำงาน
- ② หลังจากสายพานหยุดทำงาน ออกแจ้งเครื่องเริ่มทำงาน (Y001) จะตั้งประมาณ 1 วินาที และหลังจากนั้นจะเริ่มทำการ Press (Y002)
- ③ เมื่อ Press (Y002) ดำเนินการเสร็จภายใน 2 วินาที ออก (Y001) ก็จะ OFF ในเวลาเดียวกัน
- ④ หลังจากดำเนินการ Press เสร็จแล้ว จำนวนตัวงานที่ Press จะถูก Reset และจะดำเนินการขั้นตอน 1-5 ใหม่อีกครั้ง

## 《Circuit program》

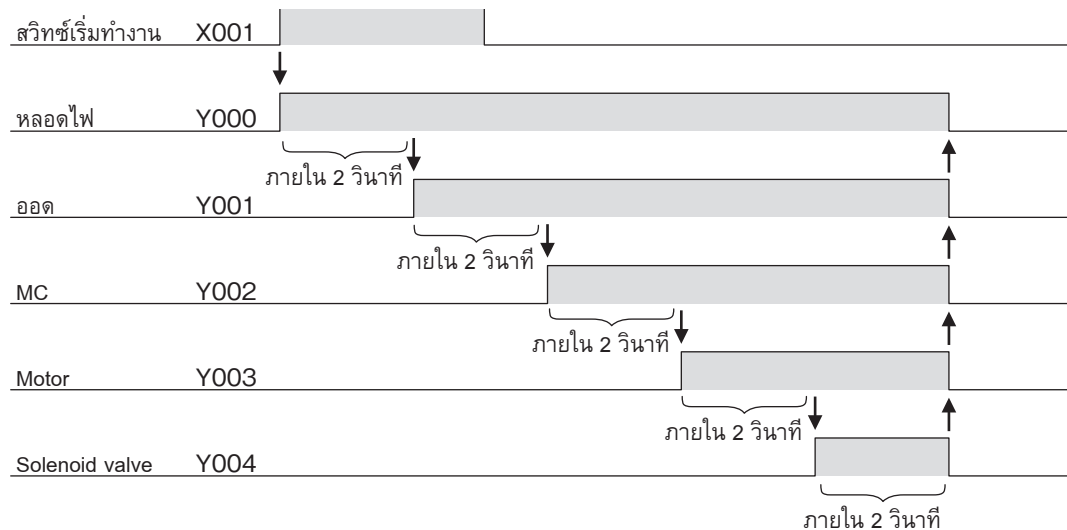
กรุณาใส่ Device ที่หายไปลงในช่องว่าง และทำให้โปรแกรมสมบูรณ์



- เผลย ดูท้ายบท
- List program กรุณาดูในภาคผนวก

## 5.5 ตัวอย่างการนำไปใช้งาน 5 (Timing chart)

มาลองคิด Program PLC โดยยัด Timing chart เป็นตัวอย่างดู



### 《I/O assignment (การกำหนด I/O)》

Input	
X001	สวิทช์เริ่มทำงาน

Output	
Y000	หลอดไฟ
Y001	ออกหรือกริ่งเตือน
Y002	MC
Y003	Motor
Y004	Solenoid valve

### 《ระบบการควบคุม》

เมื่อสวิทช์เริ่มงาน (X001) ON หลอดไฟ (Y000) ON ในเวลาเดียวกัน Timer (T0) จะเริ่มจับเวลา ออก (Y001) จะดังเมื่อครบเงื่อนไขที่ตั้งไว้คือ เวลา Timer เกินจาก 2 วินาที และ Timer (T1) จะเริ่มจับเวลาพร้อมกัน หลังจากนั้น MC (Y002), Motor (Y003), Solenoid valve (Y004) จะ ON ตามลำดับ ลดหลั่นกัน 2 วินาที เมื่อ Solenoid valve ลำดับสุดท้าย ON ครบ 2 วินาที Output ทั้งหมดจะ OFF

## 5

5



- 5

## 5.6 เฉลยคำตอบ

ตัวอย่างการติดตั้ง 3	
①	X001
②	Y000
③	X003
④	X000
⑤	X005
⑥	X006

ตัวอย่างการติดตั้ง 4	
①	Y003
②	C0 K4
③	T0
④	T1

ตัวอย่างการติดตั้ง 5	
①	Y000
②	Y001
③	T1 K20
④	Y002
⑤	T2 K20
⑥	Y003
⑦	T3 K20
⑧	Y004
⑨	T4 K20

# เรามาลองสร้างโปรแกรมโดยใช้คอมพิวเตอรืกันเถอะ!!!

## ภาคผนวก 1

### GX Developer

---

ขอแค่ใช้คอมพิวเตอรืเป็น ซีเควนซ์ (Sequence) ก็เป็นเรื่องง่าย ๆ...

ในการจะสร้างหรือแก้ไขซีเควนซ์โปรแกรม (Sequence Program) สามารถดำเนินการได้อย่างง่ายดาย เหมือนการวาดรูปง่าย ๆ ด้วยซอฟต์แวร์สำหรับติดตั้งในคอมพิวเตอรื GX Developer เพียงแค่เรียนรู้พื้นฐานการใช้งานเบื้องต้น ที่เหลือก็แค่การฝึกให้เกิดความชำนาญ เป็นซอฟต์แวร์ (Software) ที่มีฟังก์ชันที่ใช้งานง่ายและหลากหลาย ก่อนอื่นเรามาเรียนรู้และทำความรู้จักกับการใช้งานที่จำเป็นตามลำดับเพื่อให้เกิดความเชี่ยวชาญ และชำนาญในการใช้งานกันเถอะ

การเรียกใช้งานและการปรับปรุงแก้ไขก็สามารถดำเนินการได้อย่างราบรื่น...

การสร้างโปรแกรม (Program) กับการ Debug (แก้จุดบกพร่อง) เป็นของคู่กัน เนื่องจากสามารถทำการแสดงผล (Monitoring) สภาพการทำงานของ Program และ PLC ได้จากจอคอมพิวเตอรื จึงสะดวกในการตรวจสอบการทำงานและแก้ไขให้เป็นไปตามที่ต้องการได้อย่างรวดเร็ว

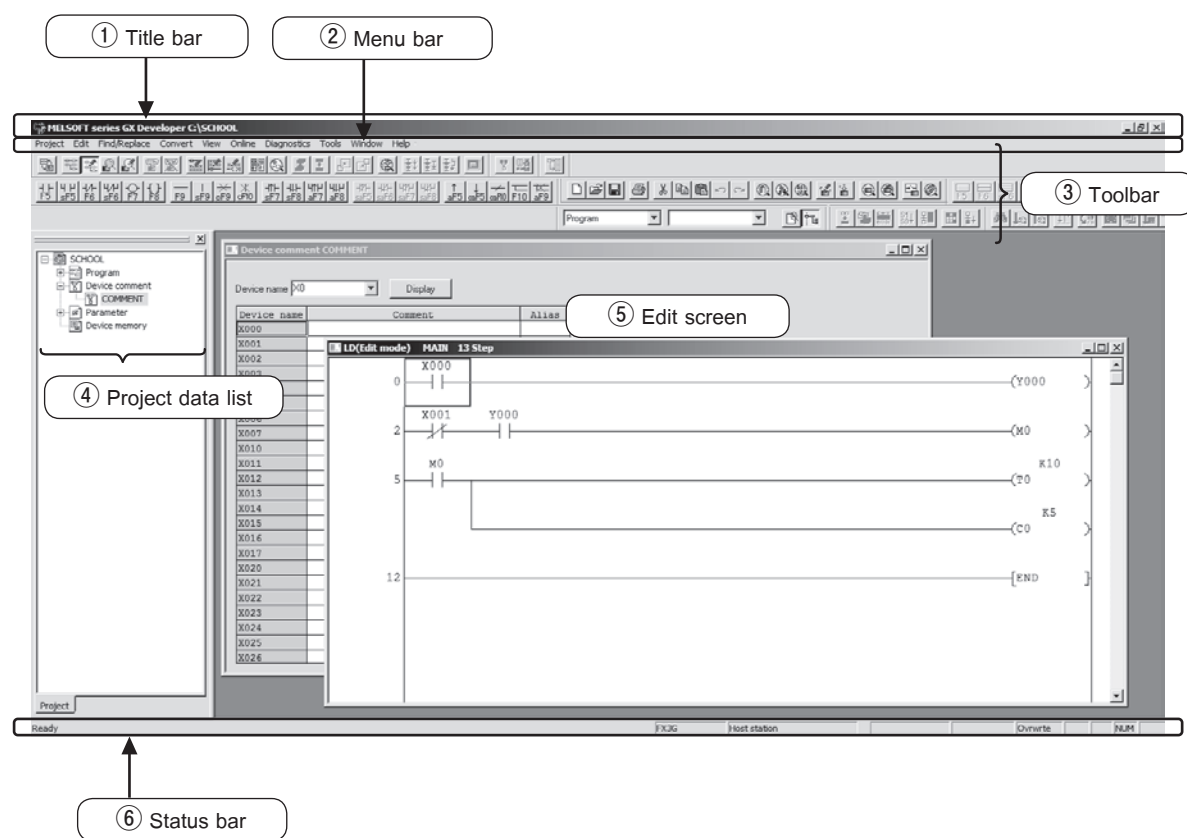
โปรแกรม (Program) ดูกง่ายไม่ซับซ้อน...

เพื่อให้สามารถดูซีเควนซ์โปรแกรม (Sequence Program) ได้ง่ายขึ้น ภายใน GX Developer จึงมีฟังก์ชัน “Comment input function”

เพียงแค่กรอก Comment เอาไว้ก็จะทำให้เข้าใจโปรแกรมได้ง่ายขึ้น และช่วยเพิ่มประสิทธิภาพในการจัดทำซีเควนซ์โปรแกรม (Sequence Program) และ Debug (แก้จุดบกพร่อง) ได้

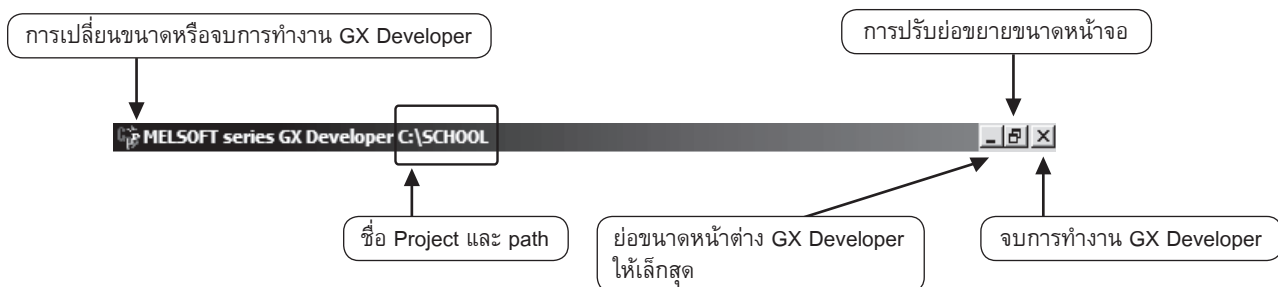
# ภาคผนวก 1.1 ความรู้พื้นฐานเพื่อการใช้งาน GX Developer

## ภาคผนวก 1.1.1 โครงสร้างหน้าจอของ GX Developer



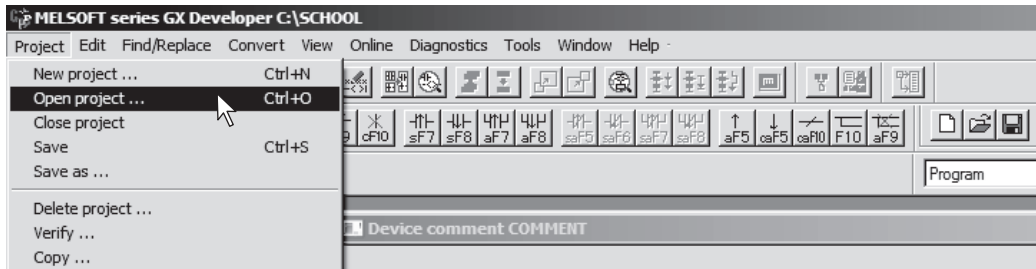
### ① Title bar

จะแสดงชื่อ Project ที่กำลังเปิดทำงานอยู่ และ Icon การทำงานของ Windows



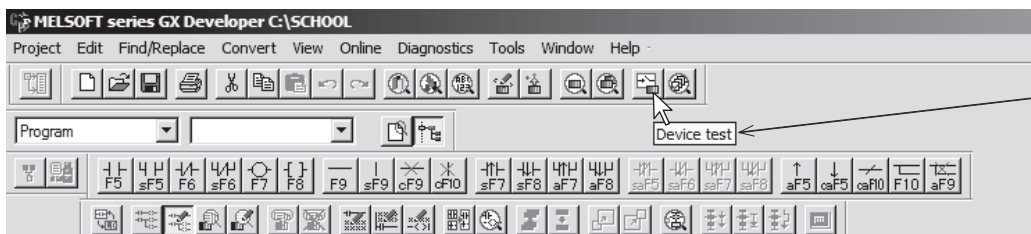


## ② Menu bar



กดเลือกเมนู แล้วจะมีหน้าจอเมนู Drop down แสดงขึ้นมา

## ③ Toolbar

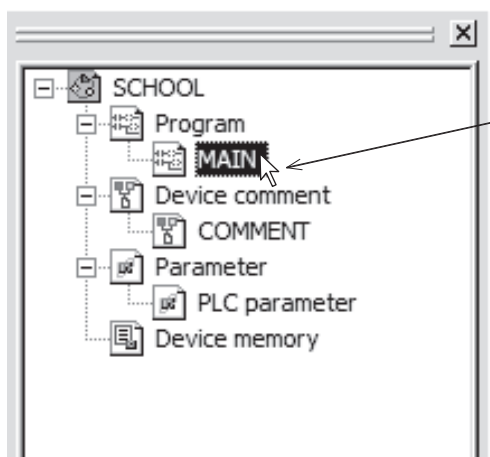


เมื่อนำเมาส์ไปวาง  
จะปรากฏข้อความ  
อธิบายรายละเอียด  
ของฟังก์ชันนั้นๆ

\* : รายละเอียดของ Toolbar เนื่องจากสามารถทำการเคลื่อนย้ายหรือเอาออกได้ ทำให้หัวข้อที่แสดงและการจัดวางแตกต่างกันไป

หัวข้อไหนที่มีการใช้งานบ่อยๆ จะเอามาวางเป็น Icon buttons สามารถคลิกเลือกได้โดยตรง ซึ่งจะเสมือนการเลือก  
เครื่องมือจาก Menu bar

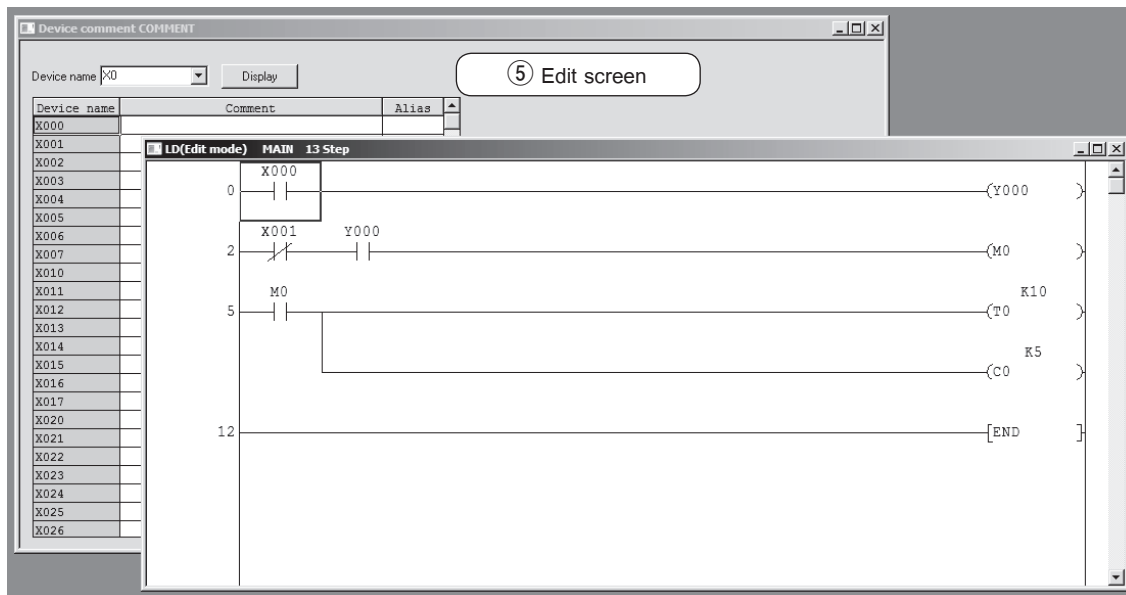
## ④ Project data list



สามารถเลือกชื่อที่ปรากฏอยู่  
ได้โดยตรง โดยการคลิกเมาส์

หน้าจอในการสร้าง Ladder creating หรือหน้าจอ Set Parameter เป็นต้น จะแสดงโดย Tree structure

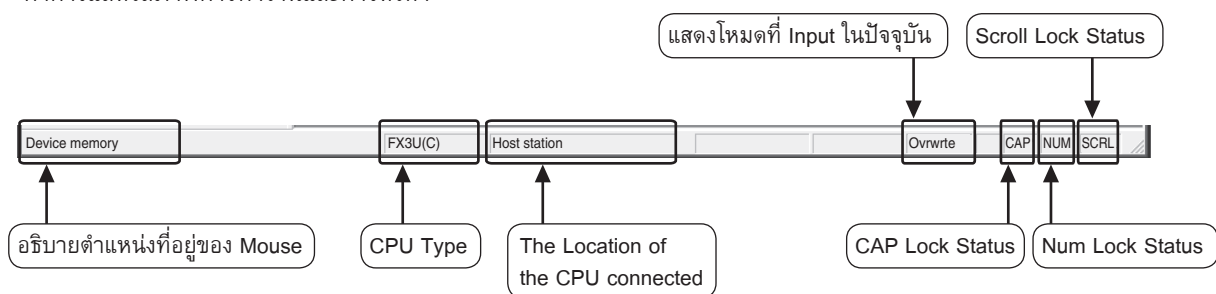
## ⑤ Edit screen



หน้าจอในการสร้าง Ladder creating หรือหน้าจอ monitor จะแสดงหลายรูปแบบในหน้าจอ

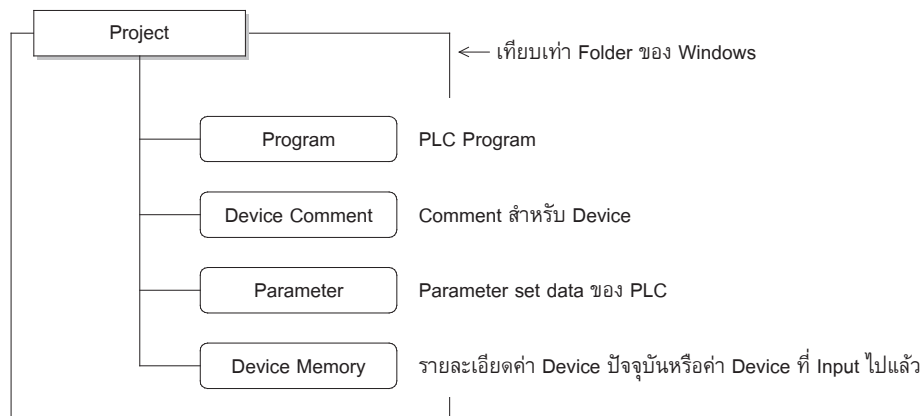
## ⑥ Status bar

ทำการแสดงสภาพการทำงานและการตั้งค่า



## ภาคผนวก 1.1.2 เกี่ยวกับ “Project”

“Project” หมายถึง สิ่งที่รวม Program, Device comment, Parameter, Device Memory เข้าด้วยกัน  
GX Developer จะเรียกรวบรวม Data ว่า Project ซึ่งจะเก็บไว้ใน Folder Windows

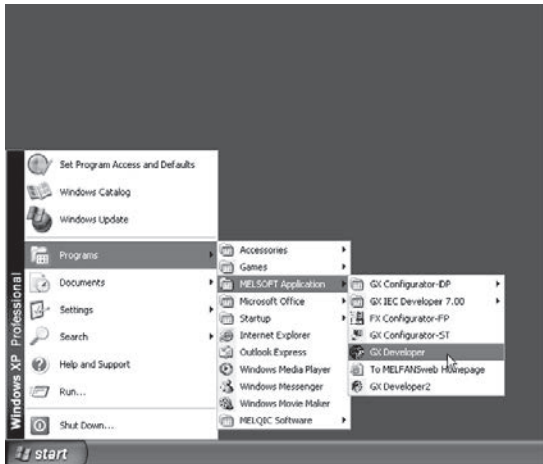


### ● การ Edit หลาย Project

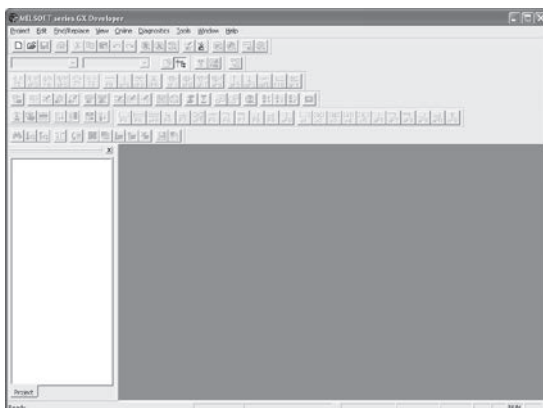
ในกรณีที่ Edit Project มากกว่า 2 Project ด้วย GX Developer ให้เปิด GX Developer แยกขึ้นมาอีกต่างหาก

## ภาคผนวก 1.2 การเรียกใช้งาน GX Developer และการจัดทำ Project ใหม่

### ภาพผนวก 1.2.1 การเรียกใช้งาน GX Developer

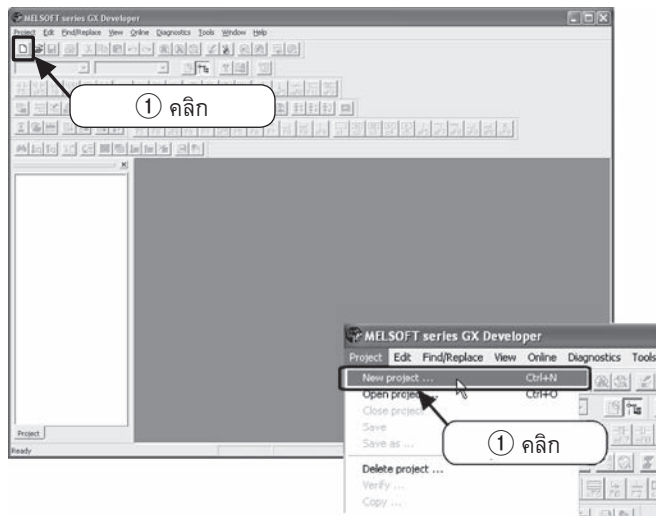



- ① กดปุ่ม **Start** Start Windows เลือก [Programs]  
↓  
[MELSOFT Application]  
↓  
เลือก [GX Developer]

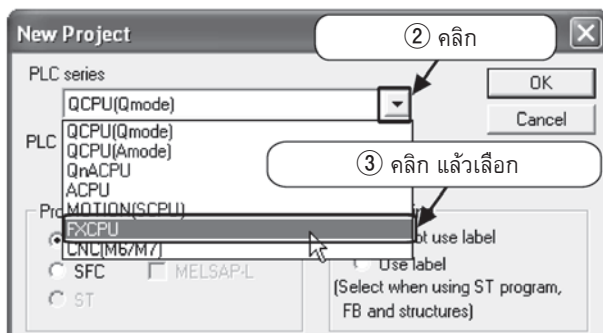


- ② GX Developer เริ่มทำงาน

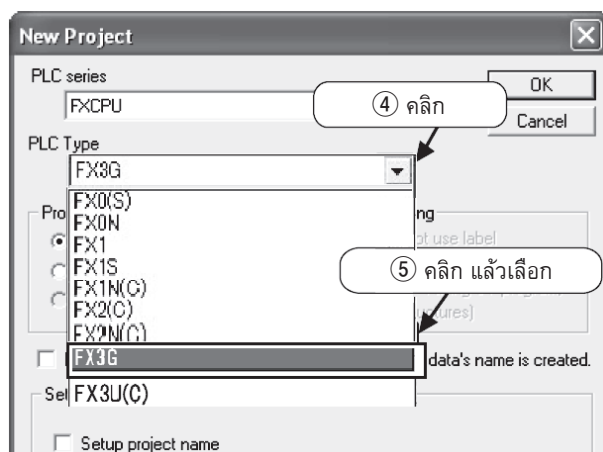
## ภาคผนวก 1.2.2 การจัดทำ Project ใหม่ (New Project)



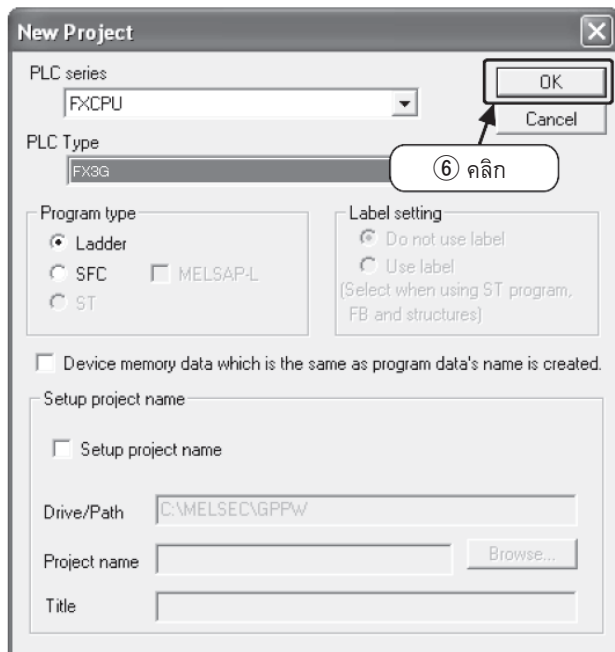
- ① เลือก  จาก Toolbar หรือเลือก [Project] → [New Project] ( **Ctrl** + **N** ) จากเมนู



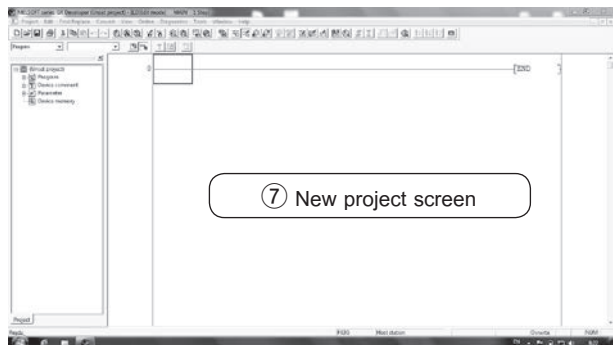
- ② คลิกปุ่ม [▼] ของ PLC Series  
③ เลือก "FXCPU"



- ④ คลิกปุ่ม [▼] ของ PLC type  
⑤ เลือก "FX3G"  
(ข้อควรระวัง) ทำการเลือกให้ตรงกับ รุ่น PLC ที่ท่านใช้งาน  
อยู่ปัจจุบัน



⑥ คลิก OK

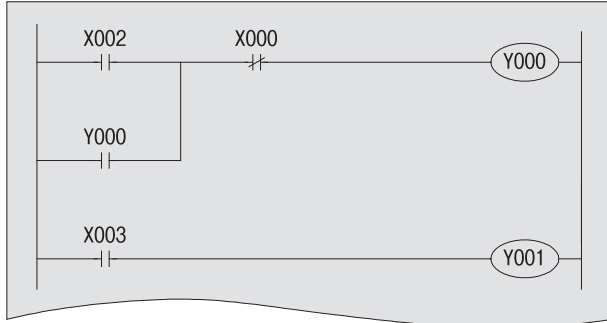


⑦ หน้าจอ Project ใหม่จะปรากฏสภาพพร้อมที่สามารถทำการ Input Project

## ภาคผนวก 1.3 การสร้างโปรแกรมหรือวงจร (Circuit)

### ภาคผนวก 1.3.1 การสร้างโปรแกรมหรือวงจร (Circuit) โดยใช้ฟังก์ชันคีย์ (Function Key)

[Program ที่จะจัดสร้าง]



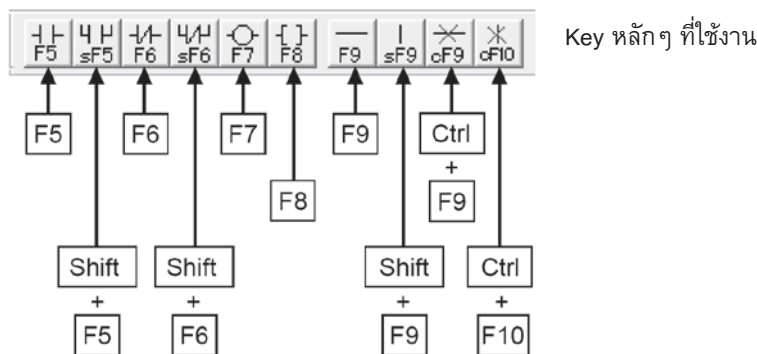
#### Point

ในเอกสารฉบับนี้ จะแสดงหมายเลข Input relay (X), Output Relay (Y) เป็นสามหลัก "X000", "Y000"

เมื่อ Input จากคอมพิวเตอร์ จะสามารถย่อได้เป็น "X0", "Y1"

#### Point

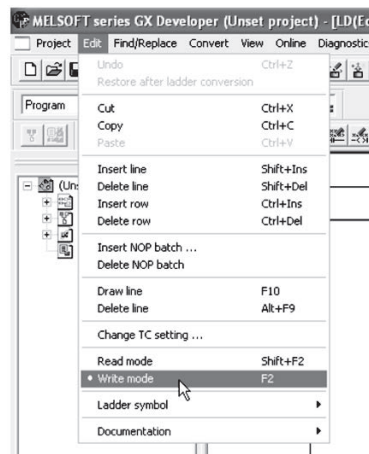
- Function Key และ ladder elements จะถูกแสดงที่ ปุ่มบน Toolbar



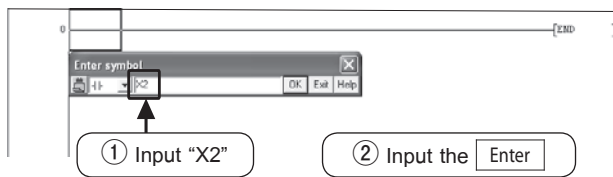
- ในตอนที่จัดสร้าง Program ขอให้ตรวจสอบว่าเลือกโหมด "Write mode" เรียบร้อยแล้ว  
การเลือกจาก Toolbar



ทำการเลือกจากเมนู ([Edit] → [Write mode])



- การป้อนตัวอักษรขอให้ป้อนเป็นตัวพิมพ์เล็ก

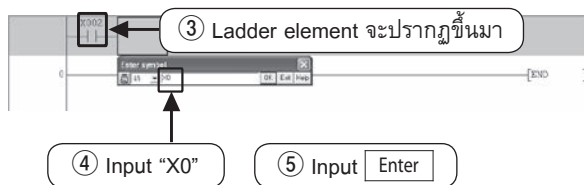


- ① กดปุ่ม **F5** (**⇧F5**)  
Input "X2"

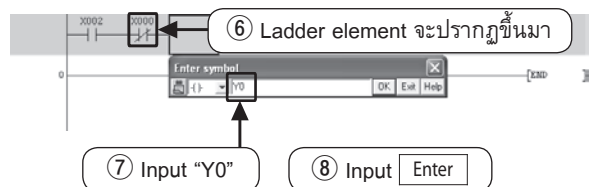


กด **ESC** หรือ **[Exit]**  
เพื่อ Cancel

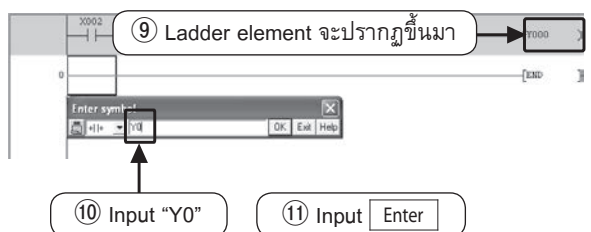
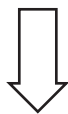
- ② ทำการยืนยันโดยกด **Enter** หรือ **[OK]**



- ③ Ladder ที่ input แล้ว (**⇧F5**) จะปรากฏขึ้นมา
- ④ กด **F6** (**⇧F6**) หรือ input "X0"
- ⑤ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**

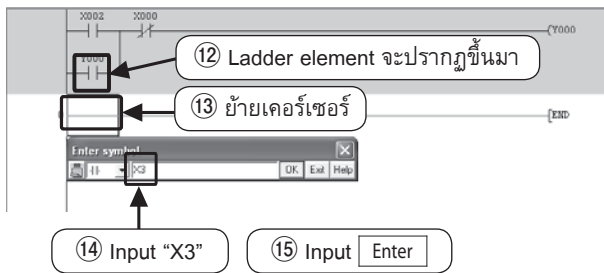


- ⑥ Ladder ที่ input แล้ว (**⇧F6**) จะปรากฏขึ้นมา
- ⑦ กด **F7** (**⇧F7**) หรือ input "Y0"
- ⑧ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**

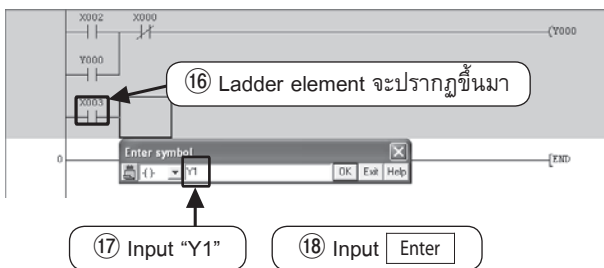


- ⑨ Ladder ที่ input แล้ว (**⇧F7**) จะปรากฏขึ้นมา
- ⑩ กด **Shift** + **F5** (**⇧F5**) หรือ input "Y0"
- ⑪ ทำการยืนยันโดยกด **Enter** หรือ **[OK]**

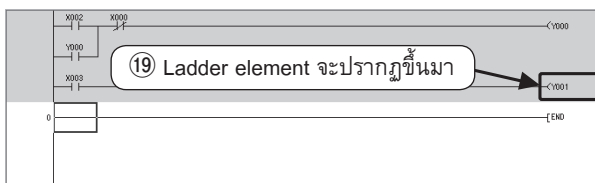




- ⑫ Ladder ที่ input แล้ว (Y0) จะปรากฏขึ้นมา
- ⑬ ทำการเคลื่อนเคอร์เซอร์มาไว้หน้าสุด
- ⑭ กด F5 (—|—) หรือ input "X3"
- ⑮ ทำการยืนยันโดยกด Enter หรือ [OK]



- ⑯ Ladder ที่ input แล้ว (X3) จะปรากฏขึ้นมา
- ⑰ กด F7 (—(—) หรือ input "Y1"
- ⑱ ทำการยืนยันโดยกด Enter หรือ [OK]



- ⑲ Ladder ที่ input แล้ว (Y1) จะปรากฏขึ้นมา!!
- เสร็จสิ้นการดำเนินการจัดสร้าง Program!!

F4 (Convert)



- ⑳ การแปลง Program (สำคัญ)

ทำการแปลง Program เพื่อให้ตำแหน่งที่ยังไม่ถูกยืนยัน (บริเวณสีเทา) สามารถเริ่มทำงานได้

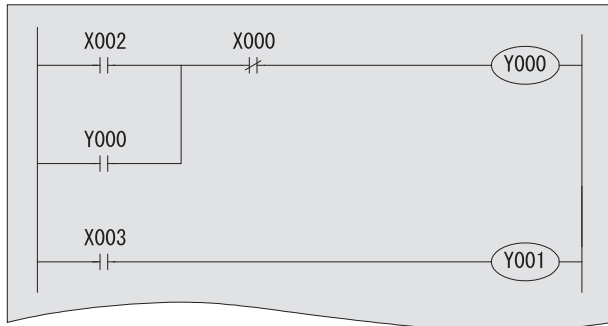
กดเลือก F4 (Convert)

หรือเลือก Toolbar  หรือเลือก [Convert] → [Convert] จากเมนู

ทำการลบสีเทาออกและยืนยัน Program  
กรณีที่เกิด Error เคอร์เซอร์จะเคลื่อนตัวไปตำแหน่งที่เกิดปัญหาขึ้น  
ขอให้ตามไปแก้ไข

## ภาคผนวก 1.3.2 จัดสร้าง Program โดยการใช้ Toolbar

[Program ที่จะจัดสร้าง]

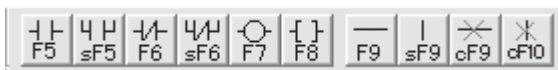


### Point

ในเอกสารฉบับนี้ จะแสดงหมายเลข Input relay (X), Output Relay (Y) เป็นสามหลัก "X000", "Y000" เมื่อ Input จากคอมพิวเตอร์ จะสามารถย่อได้เป็น "X0", "Y1"

### Point

- Function Key และ ladder elements จะถูกแสดงที่ ปุ่มบน Toolbar

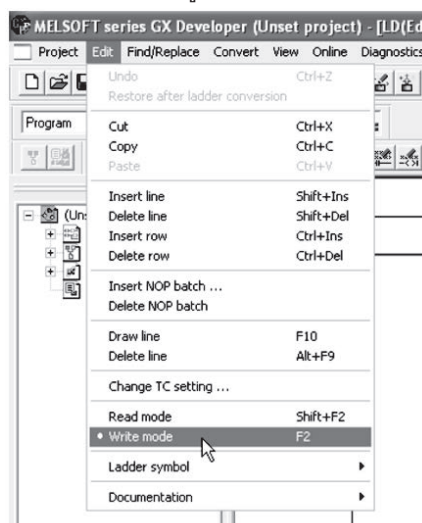


Key หลักๆ ที่ใช้งาน

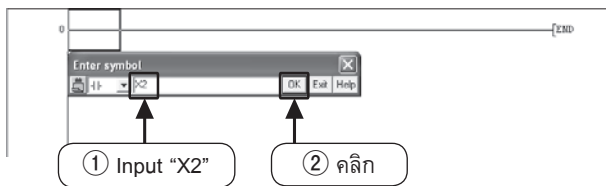
- ในตอนที่จัดสร้าง Program ขอให้ตรวจสอบว่าเลือกโหมด "Write mode" เรียบร้อยแล้ว การเลือกจาก Toolbar



ทำการเลือกจากเมนู ([Edit] → [Write mode])



- การ Input ตัวอักษร จะ Input ด้วยอักษรภาษาอังกฤษตัวพิมพ์เล็ก

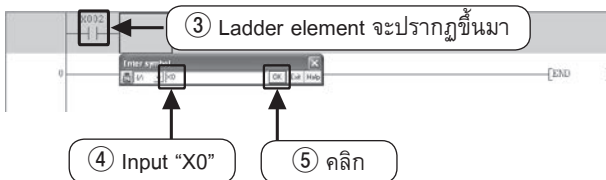


- ① คลิกเลือก  $\overline{X2}$  จาก Toolbar และ Input "X2"



กด ESC หรือ [Exit]  
เพื่อ Cancel

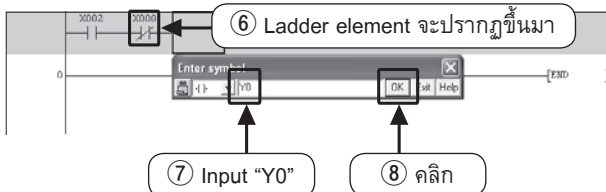
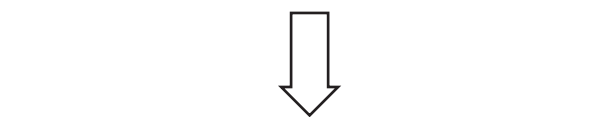
- ② ทำการยืนยันโดยกด Enter หรือ [OK]



- ③ Ladder ที่ input แล้ว ( $\overline{X2}$ ) จะปรากฏขึ้นมา

- ④ เลือก  $\overline{X0}$  จาก Toolbar แล้ว input "X0"

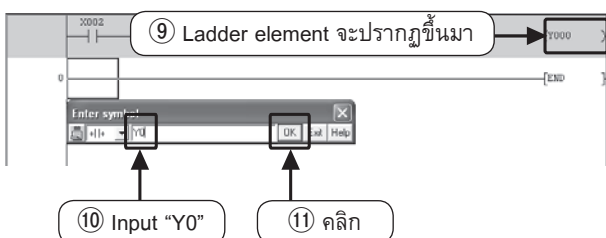
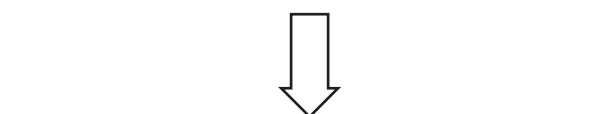
- ⑤ ทำการยืนยันโดยกด Enter หรือ [OK]



- ⑥ Ladder ที่ input แล้ว ( $\overline{X0}$ ) จะปรากฏขึ้นมา

- ⑦ เลือก  $\overline{Y0}$  จาก Toolbar แล้ว input "Y0"

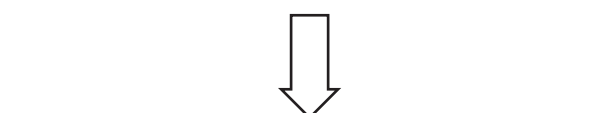
- ⑧ ทำการยืนยันโดยกด Enter หรือ [OK]

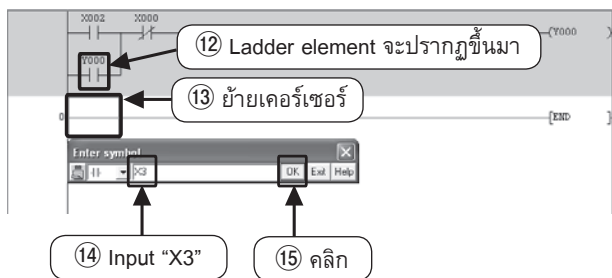


- ⑨ Ladder ที่ input แล้ว ( $\overline{Y0}$ ) จะปรากฏขึ้นมา

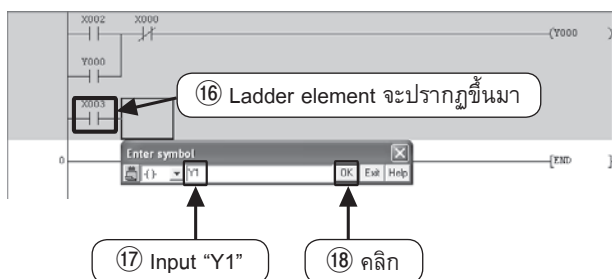
- ⑩ เลือก  $\overline{Y0}$  จาก Toolbar และ input "Y0"

- ⑪ ทำการยืนยันโดยกด Enter หรือ [OK]

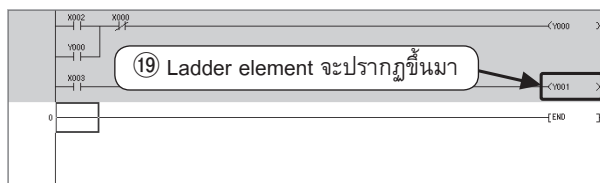




- ⑫ Ladder ที่ input แล้ว (  $\text{---|---}$  ) จะปรากฏขึ้นมา
- ⑬ ทำการเคลื่อนเคอร์เซอร์มาไว้หน้าสุด
- ⑭ เลือก  $\text{---|---}$  จาก Toolbar และ input "X3"
- ⑮ ทำการยืนยันโดยกด  หรือ



- ⑯ Ladder ที่ input แล้ว (  $\text{---|---}$  ) จะปรากฏขึ้นมา
- ⑰ เลือก  $\text{---|---}$  จาก Toolbar และ input "Y1"
- ⑱ ทำการยืนยันโดยกด  หรือ



- ⑲ Ladder ที่ input แล้ว (  $\text{---|---}$  ) จะปรากฏขึ้นมา!!
- เสร็จสิ้นการดำเนินการจัดสร้าง Program!!




(Convert)



- ⑳ การแปลง Program (สำคัญ)  
ทำการแปลง Program เพื่อให้ตำแหน่งที่ยังไม่ถูกยืนยัน (บริเวณสีเทา) สามารถเริ่มทำงานได้

กดเลือก  (Convert)

หรือเลือก Toolbar  หรือเลือก [Convert] → [Convert] จากเมนู

ทำการลบสีเทาออกและยืนยัน Program  
กรณีที่เกิด Error เคอร์เซอร์จะเคลื่อนตัวไปตำแหน่งที่เกิดปัญหาขึ้น  
ขอให้ตามไปแก้ไข

## ภาคผนวก 1.4 การเขียน Program ลง PLC

Sequence Program ที่จัดทำขึ้นจะเขียนใส่ใน FX PLC

### ภาคผนวก 1.4.1 การเชื่อมต่อกับ PLC

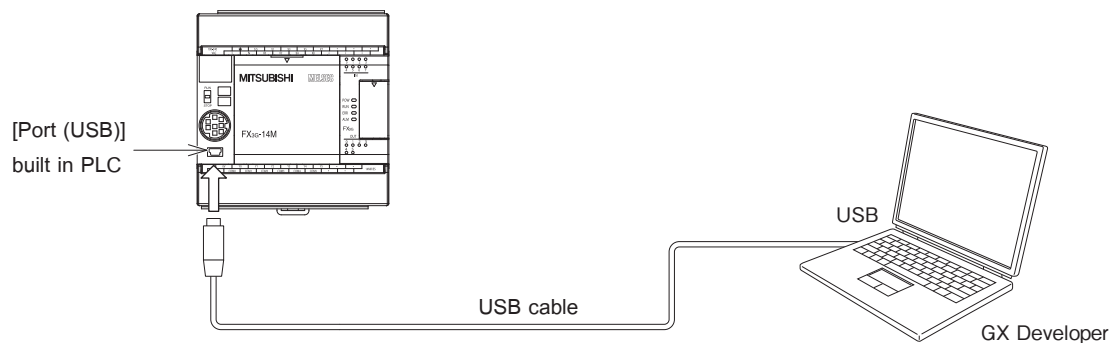
การต่อกับ FX3G PLC (เชื่อมต่อโดยใช้สาย USB)

[ขั้นตอนการเตรียมการด้านคอมพิวเตอร์]

จำเป็นต้อง Install USB Driver ลงในคอมพิวเตอร์เพื่อทำการต่อสาย USB ระหว่างคอมพิวเตอร์กับ FX3G PLC

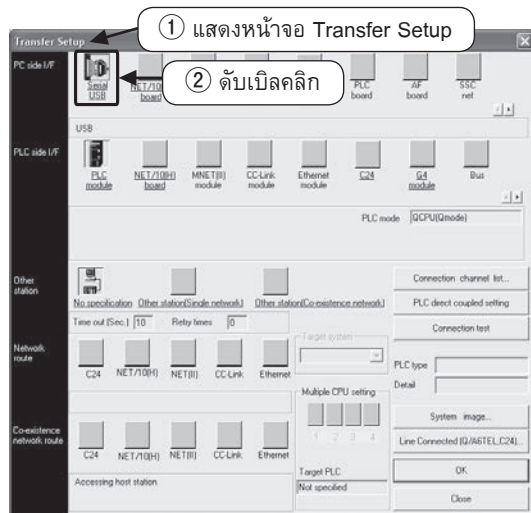
ขั้นตอนการ Install Driver ให้ปฏิบัติตามคู่มือใน [GX Developer Operation Manual (บท Start up)]

(รูปแบบการเชื่อมต่อ)




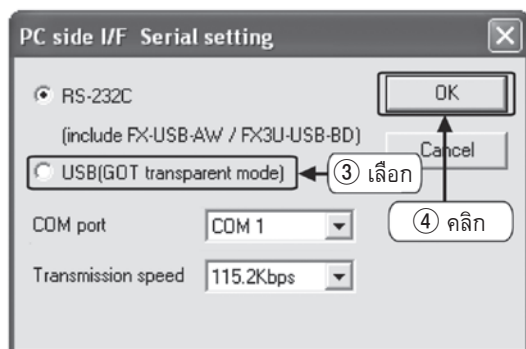
## ภาคผนวก 1.4.2 “Transfer Setup” ใน GX Developer

ดำเนินการ Setting GX Developer เพื่อเชื่อมต่อกับ PLC



① จากเมนูเลือก [Online] → [Transfer Setup]

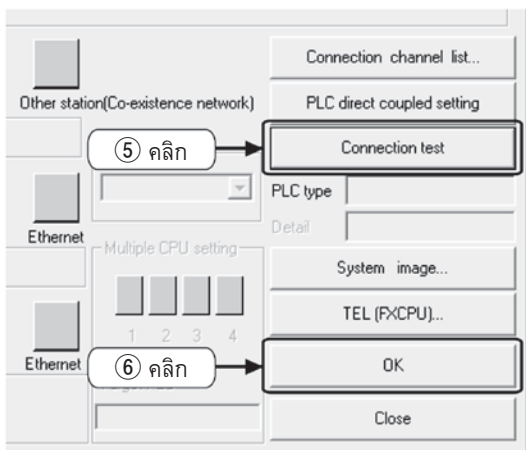
② ดับเบิลคลิกที่ Icon 



③ ทำการ Set Communication port ของคอมพิวเตอร์

- ทำการเลือก [USB (GOT transparent mode)] เมื่อจะทำการเชื่อมต่อไปที่ FX3G PLC USB port (Built in)

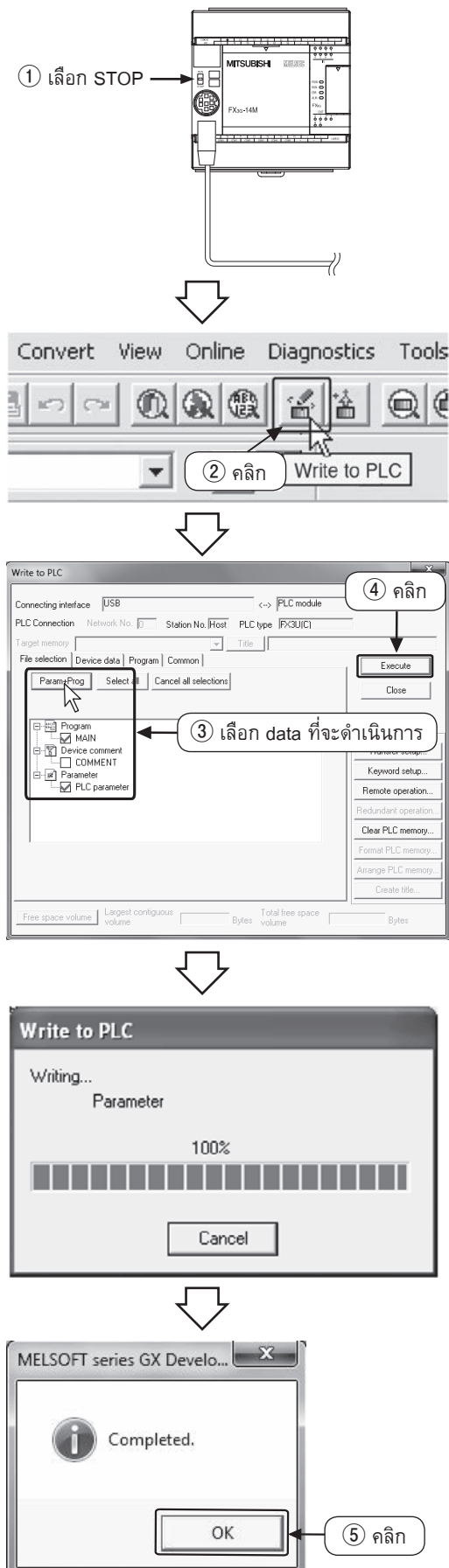
④ หลังจากเลือกเสร็จ คลิก OK




⑤ คลิก [Connection Test] เพื่อตรวจสอบการ communicate กับ PLC

⑥ หลังตรวจสอบเรียบร้อยแล้วคลิก [OK] เพื่อทำการยืนยันเนื้อหาที่ Set ไป

## ภาคผนวก 1.4.3 การเขียน Program



① ทำการสลับตรงสวิตช์ [RUN/STOP] ของ PLC เป็น [STOP]

② ทำการเลือก  จาก Toolbar หรือจากเมนู เลือก [Online] → [Write to PLC]

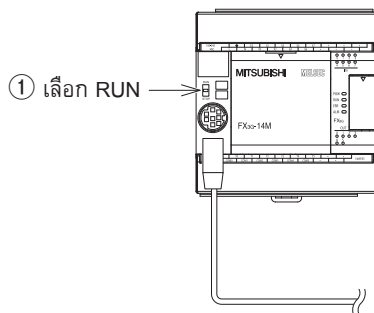
③ คลิก [Param + Prog.]

④ คลิก [Execute]

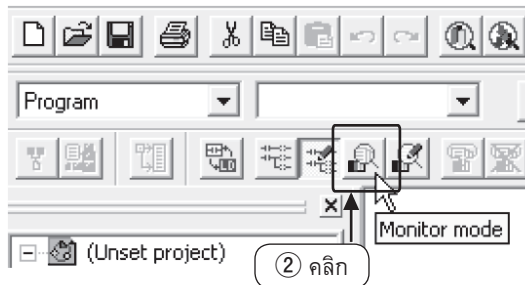
จะปรากฏ Dialog box แสดงสถานะการ Write

⑤ เมื่อดำเนินการเสร็จสิ้นคลิก [OK]

## ภาคผนวก 1.4.4 การ Monitoring การทำงานของ PLC



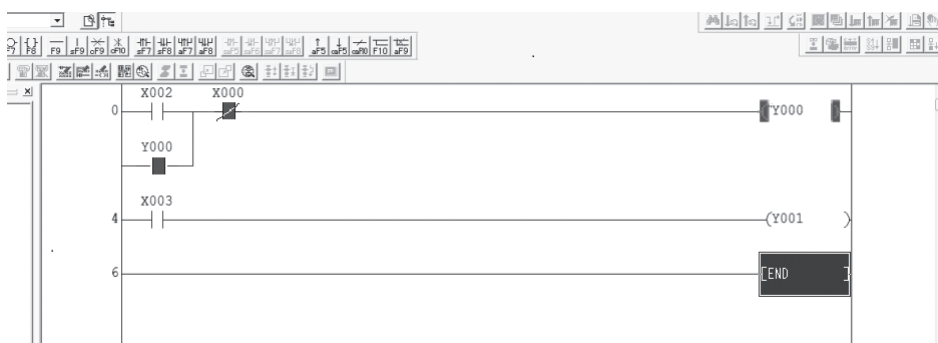
① ทำการสลับตรงสวิตช์ [RUN/STOP] ของ PLC เลือก [RUN]



② ทำการเลือก  จาก Toolbar หรือจากเมนู เลือก [Online] → [Monitor] → [Monitor Mode]



ตรวจสอบการทำงานได้จาก Monitor Mode

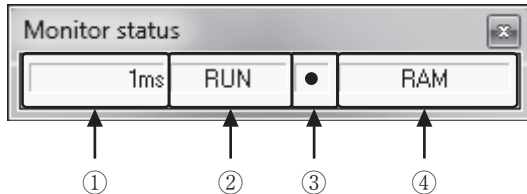


- ① เมื่อ [switch X000 "OFF"] แล้วทำการ [switch X002 "ON"] ให้ทำการตรวจสอบว่า [Output Y000 "ON"] หรือไม่
- ② ทำการตรวจสอบว่าเมื่อ [switch X002 "OFF"] [Output Y000 "ON"] หรือไม่
- ③ ทำการตรวจสอบว่าเมื่อ [switch X000 "ON"] [Output Y000 "OFF"] หรือไม่
- ④ ทำการตรวจสอบว่าเมื่อทำการสลับ [switch X003 "ON/OFF"] [Output Y001 "ON/OFF"] หรือไม่



## อ้างอิง

### (1) แสดง Monitor Status Dialog



#### ① Scan time

แสดง Scan time สูงสุดของ PLC Program

#### ② PLC Status

แสดง PLC Status

#### ③ Execute status display

ตำแหน่งที่ทำงาน (Execute) จะกะพริบ

#### ④ Memory type

แสดง Memory type ของ PLC

### (2) วิธีการแสดง Status ของ Ladder Monitor

#### ① คำสั่งหน้าสัมผัส (Contact Instruction)

Type \ Input contact	X0 : OFF	X0 : ON
หน้าสัมผัสปกติเปิด a (N.O.Contact)	X000 — — หน้าสัมผัสเปิด (Contact open)	X000 — /— หน้าสัมผัสปิด (Contact close)
หน้าสัมผัสปกติปิด b (N.C.Contact)	X000 — /— หน้าสัมผัสเปิด (Contact close)	X000 — — หน้าสัมผัสปิด (Contact open)

#### ② คำสั่งเอาต์ (Out Instruction)

Type \ Drive status	ไม่ทำงาน/ไม่เคลื่อนไหว	ทำงาน/เคลื่อนไหว
—( )— OUT Instruction	—(Y000)—	—(Y000)—
—[ ]— SET Instruction. Etc.	—[SET M0]—	—[SET M0]—

คำสั่ง RST (RST Instruction) คือ การแสดง ON/OFF ของ Device ที่จะ Reset

Type \ Drive status	Reset Device OFF	Reset Device ON
—[ ]— RST instruction	—[RST M0]—	—[RST M0]—

## ภาคผนวก 1.5 การแก้ไข Program (Editing program)

### ภาคผนวก 1.5.1 แก้ไข Program

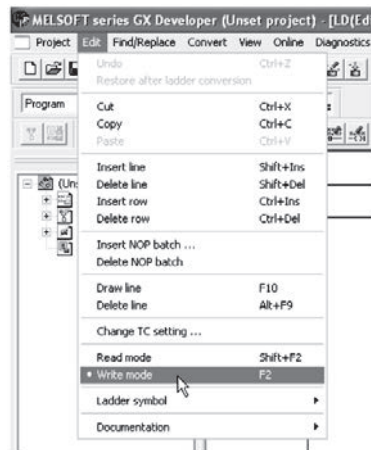
#### Point

- เมื่อจะทำการแก้ไข Program ให้เลือก "Write mode"

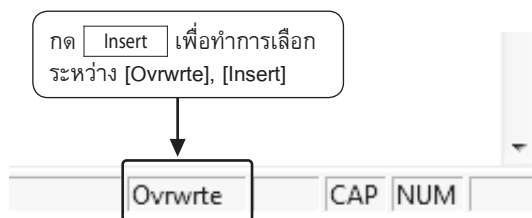
ทำการเลือกจาก Toolbar



ทำการเลือกจากเมนู ([Edit] → [Write mode])

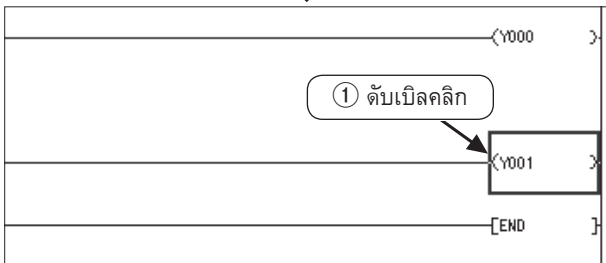
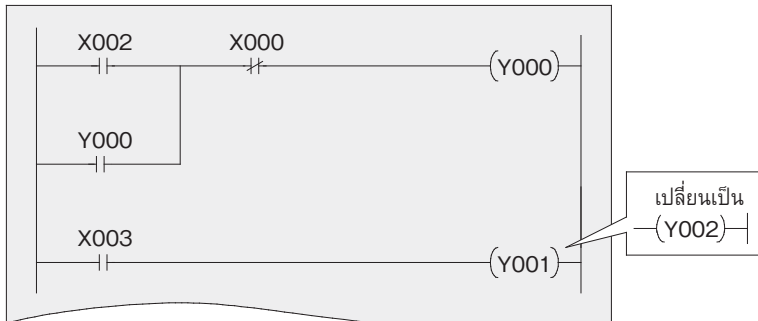


- ตัวหนังสือที่ใช้ ขอให้เป็นตัวพิมพ์เล็กทั้งหมด ตัวพิมพ์ใหญ่ไม่สามารถใช้งานได้
- การเลือกระหว่าง [Ovrwrite], [Insert]
  - กรณีที่ทำการแก้ไข Program ที่มีอยู่เดิม ขอให้เลือก [Ovrwrite]
  - เมื่อทำการเลือกโหมด [Insert] จะเป็นการเพิ่ม Program ใหม่ แยกเป็นอีก Program

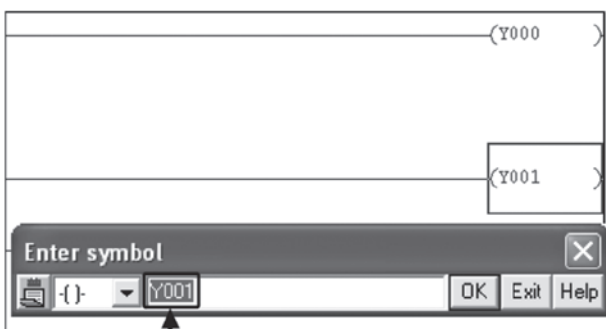


# ① ทำการเปลี่ยนหมายเลขของ OUT Coils หรือ Contact

[Program ที่จะแก้]

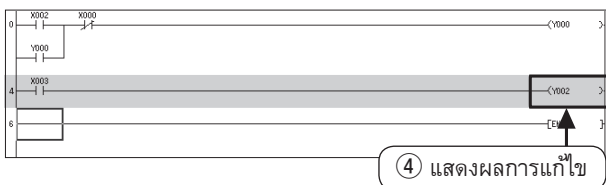


① ดับเบิลคลิกตรงตำแหน่งที่ต้องการแก้ไข



② ทำการเปลี่ยน “Y001” → “Y002”

③ กด Enter

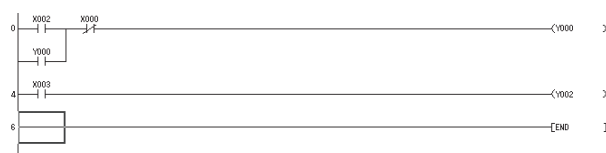


③ ทำการยืนยันโดยกด Enter หรือ [OK]

④ ผลการแก้ไขจะถูกแสดงในกรอบสี่เทา



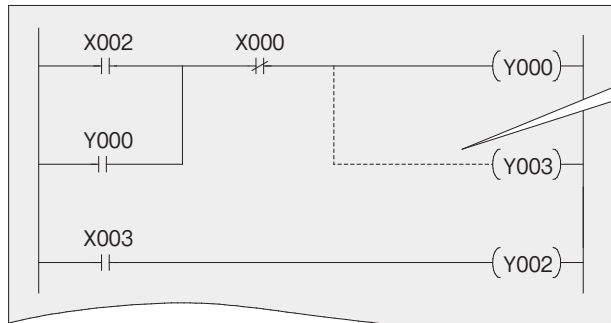
F4 (Convert)



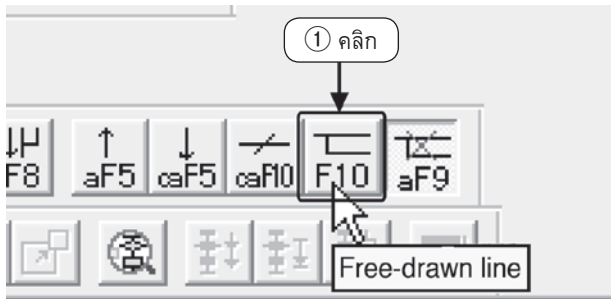
⑤ กด F4 (Convert) เพื่อทำการยืนยันเพื่อทำการเปลี่ยนแปลง

## ② การเพิ่ม Lines

[Program ที่จะทำการเพิ่ม Lines]



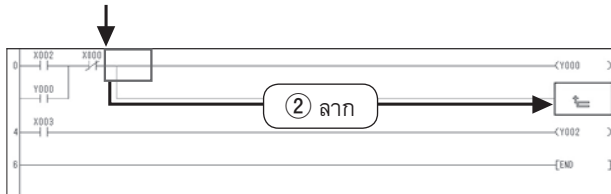
ทำการเพิ่ม Line แนวตั้ง/แนวนอน แล้วทำการสร้าง OUT Coil



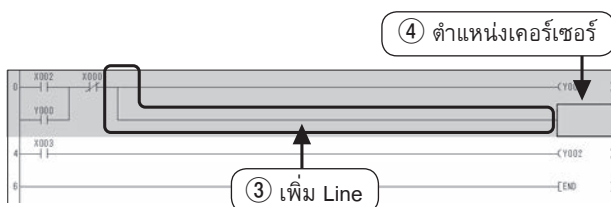
① คลิก (F10) จาก Toolbar



นำเคอร์เซอร์ไปวางด้านขวาบนตำแหน่งเริ่มลากเส้น Line

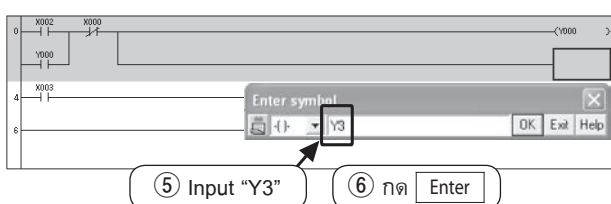


② เมื่อนำเคอร์เซอร์ไปวางด้านขวาบนของตำแหน่งที่ต้องการเพิ่ม Line แล้ว ให้ทำการลากเส้น



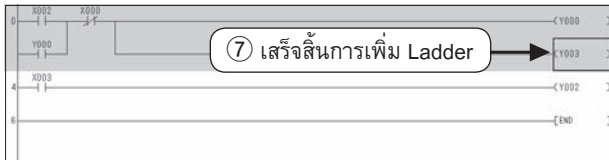
③ Line จะเพิ่มขึ้นตามตำแหน่งที่ลากเคอร์เซอร์

④ วางเคอร์เซอร์ในตำแหน่งที่ต้องการเพิ่ม OUT Coil แล้วคลิก จาก Toolbar



⑤ Input "Y3" เข้าไป

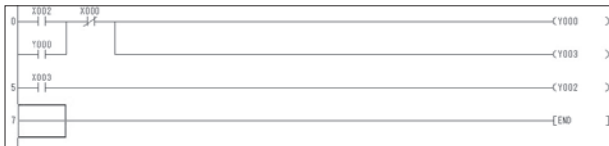
⑥ ทำการยืนยันโดยกด  หรือ



⑦ เมื่อทำการเพิ่ม Ladder แล้ว จะแสดงเป็นบล็อกสีเทา



⑧ **F4** (Convert)

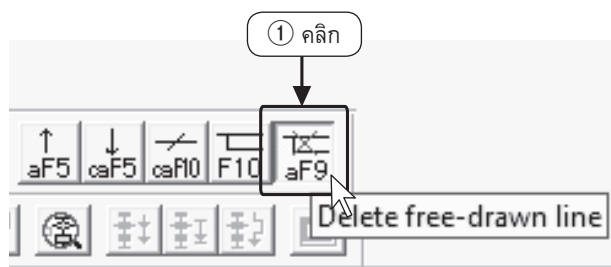
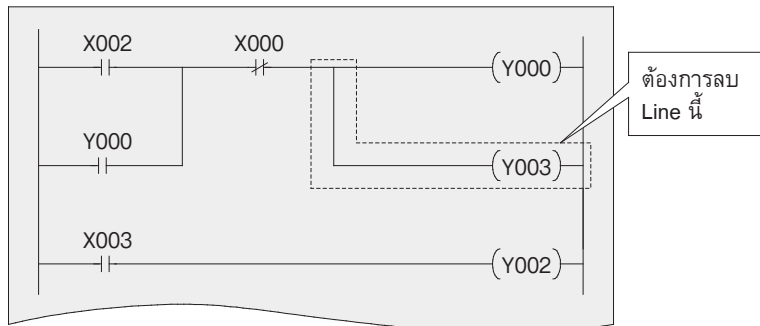



⑧ ทำการกด **F4** (Convert) เพื่อยืนยันการเปลี่ยนแปลง

- เมื่อดำเนินการเสร็จแล้ว ทำการคลิก  จาก Toolbar อีกครั้ง

### ③ การลบ Lines

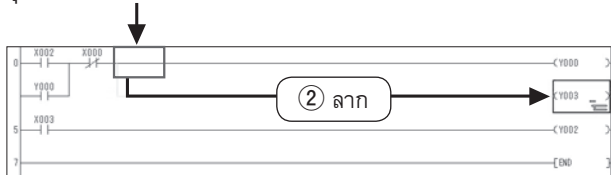
[Program ที่ต้องการลบ Lines]



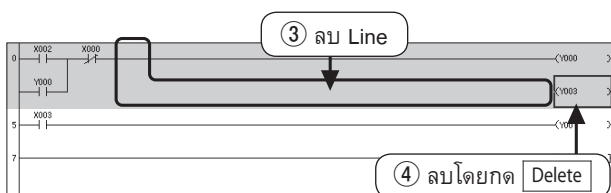
① คลิก  ( Alt + F9 ) จาก Toolbar



วางเคอร์เซอร์ที่ตำแหน่งขวามือ  
จุดเริ่มของ Line ที่ต้องการจะลบ

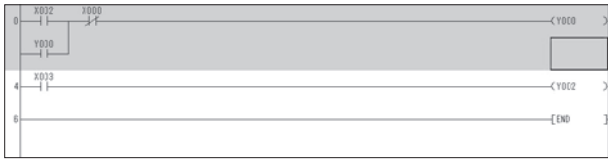


② ทำการลากเคอร์เซอร์จากด้านขวามือ Line ที่ต้องการยกเลิก  
ไปถึงตำแหน่งสุดท้ายของ Line

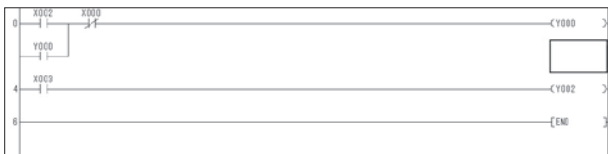


③ ทำการลบ Line  
④ OUT Coils ทำการลบโดยกด Delete





⑥ F4 (Convert)



⑤ Line ที่ต้องการจะลบจะแสดงขึ้นมาเป็นกรอบสี่เทา

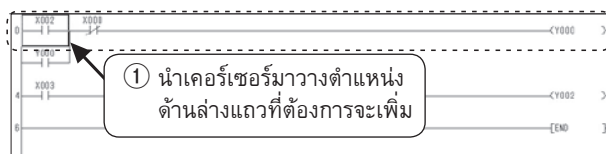
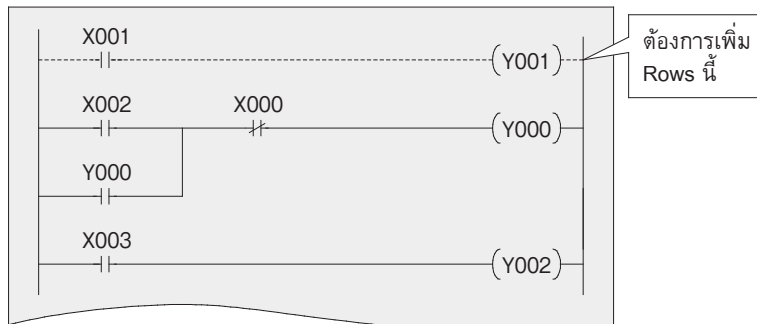
⑥ กด F4 (Convert) ทำการยืนยันการเปลี่ยนแปลง

- เมื่อดำเนินการเสร็จแล้ว ทำการคลิก  จาก Toolbar อีกครั้ง

## ภาคผนวก 1.5.2 การแทรก/ลบ Rows

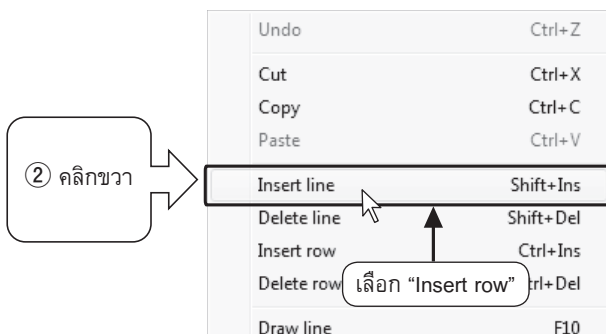
### ① การแทรก Rows

[Program ที่ต้องการแทรก Rows]

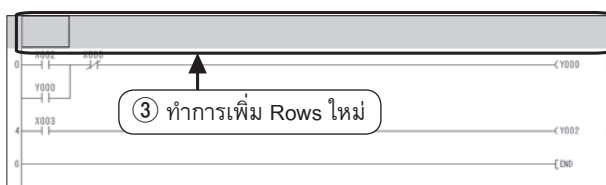


การแทรก Rows ทำการแทรกด้านบนแถวที่เคอร์เซอร์วางอยู่

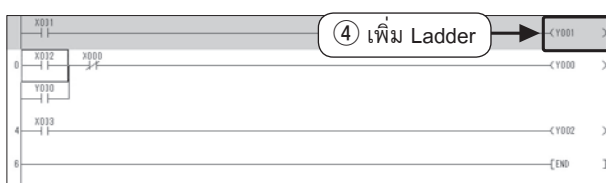
① นำเคอร์เซอร์วางแถวล่างของแถวที่ต้องการแทรก



② ทำการคลิกขวาในตำแหน่งใดก็ได้ เพื่อเลือก [Insert row]



③ เสร็จสิ้นการแทรก Rows



④ ทำการเพิ่ม Ladder ลงในแถวที่แทรกเข้าไปใหม่

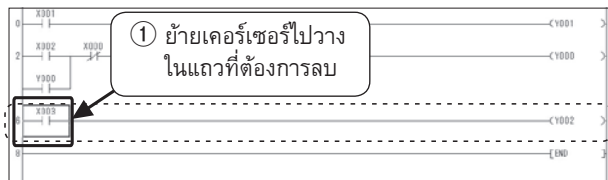
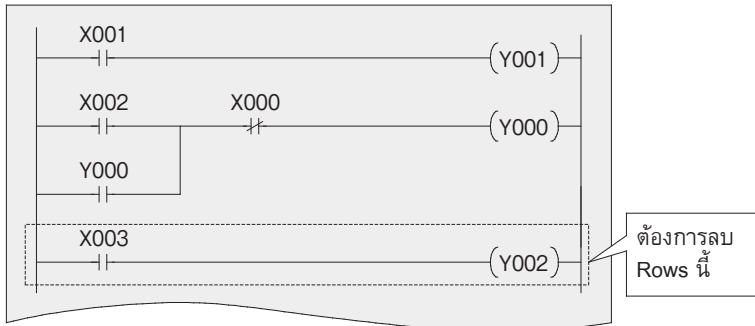


⑤ ทำการกด Convert (F4) เพื่อยืนยันการเปลี่ยนแปลง

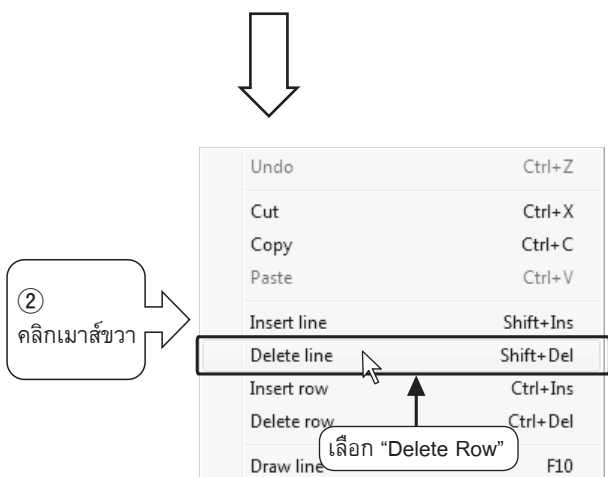


## ② การลบ Rows

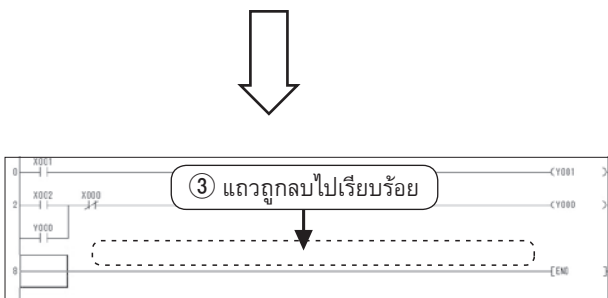
[Program ที่ต้องการลบ Rows]



① ย้ายคอน์เซอร์ไปวางใน Rows ที่ต้องการลบ



② วางเมาส์ในตำแหน่งใดก็ได้แล้วคลิกขวา เลือก [Delete Row]



③ แถวถูกลบไปเรียบร้อยแล้ว

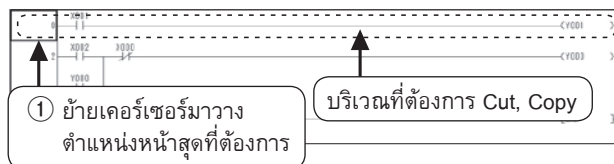
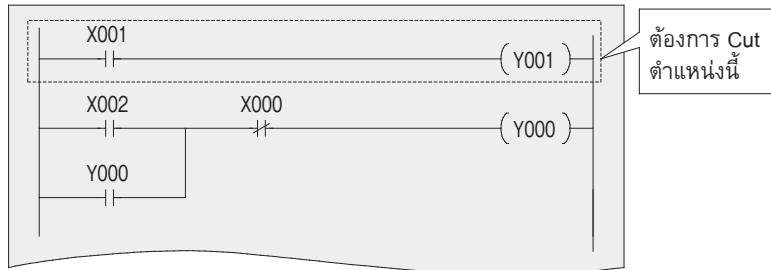
### Point

กรณีที่ต้องการลบ Rows จะไม่มีการแสดงเป็นกรอบสีเทา ให้กด **Convert** F4 เพื่อทำการยืนยันการดำเนินการ

## ภาคผนวก 1.5.3 การ Cut & Copy (Pasting) Ladder

### ① การ Cut


[Program ที่ต้องการ Edit]

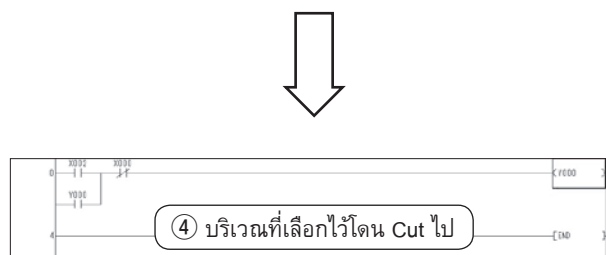


① ย้ายเคอร์เซอร์มาวางหน้า Ladder ที่ต้องการ Cut



② ทำการลากคลุมทุกตำแหน่ง

③ Cut โดยเลือก  จาก Toolbar หรือเลือก [Edit] → [cut] (Ctrl + X) แล้วทำการ Cut

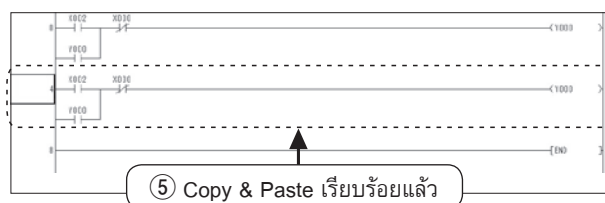
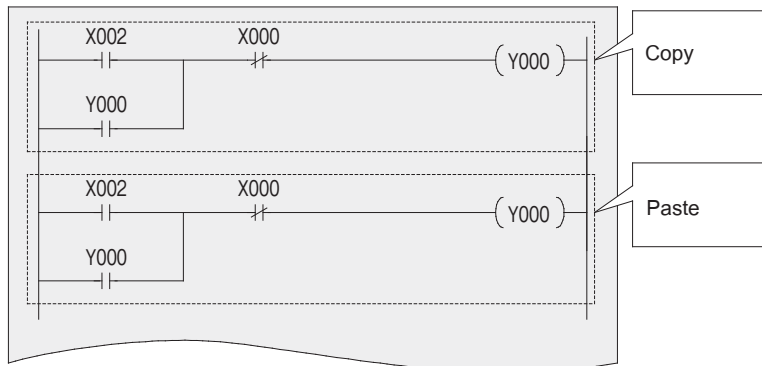


④ บริเวณที่เลือกไว้จะถูก Cut

กรณี Cut บางส่วนของ Ladder จะเหลือแสดงเป็นแถบสีเทาอยู่ หลังจากแก้ Ladder ไว้ กด **F4** (Convert) เพื่อยืนยันการดำเนินการ

## ② Copy (Pasting)


ส่วนของโปรแกรมที่ต้องการ Copy (Pasting)



นำ Program ที่ "cut" แล้วมาวางแถวหน้า

① ทำการย้ายเคอร์เซอร์มาวางไว้ที่แถวหน้าของ Ladder ที่ จะทำการ Copy

② ทำการลากคลุมตำแหน่งจบ

③ ทำการเลือก  ที่ Toolbar หรือเลือกเมนู [Edit] → [Copy] (Ctrl + C)


④ ย้ายเคอร์เซอร์มาวางในตำแหน่งที่ต้องการ Paste


### Point

การกดเปลี่ยน 

[Ovrwrte] mode : การ Paste โดยการเขียนทับ ในตำแหน่งที่เคอร์เซอร์วาง

[Insert] mode : ทำการ Insert ข้างบนตำแหน่งที่วางเคอร์เซอร์

⑤ ทำการเลือก  จาก Toolbar หรือเลือก [Edit] → [Paste] (Ctrl + V) จากเมนู

กรณีต้องการ Paste เฉพาะบางส่วนของ Program จะมี การแสดงเป็นกรอบสีเทา ให้กด  (Convert) เพื่อเป็นการ ยืนยัน


# ภาคผนวก 1.6 การบันทึก Program ที่จัดทำเสร็จแล้ว

## ภาคผนวก 1.6.1 การ Save, Save as

### Point

กรณีที่ Ladder ยังไม่ได้รับการยืนยัน ให้กด F4

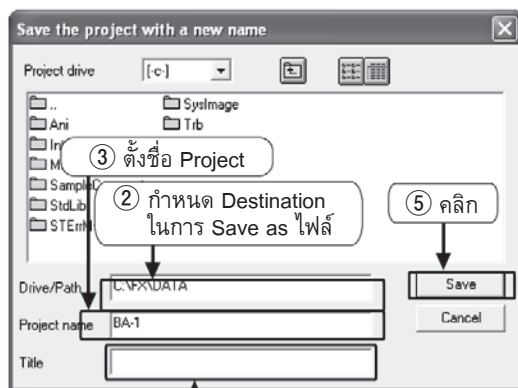


- 1 ทำการเลือก  จาก Toolbar หรือเลือก [Project] → [Save] จากเมนู ( + )

(เฉพาะกรณี Save as เท่านั้น)

(กรณี Overwrite)

เสร็จเรียบร้อยแล้ว



- 2 กำหนด Destination ของ Project
- 3 ตั้งชื่อ Project
- 4 สามารถกำหนด Title ได้ เช่น รายละเอียดของ Project (option)
- 5 คลิก

4 ใส่ Title ของไฟล์



- 6 จะมี Dialog box ปรากฏขึ้นมา คลิก

กรณี Save ใน แผ่น Floppy disk ความจุอาจไม่พอ ให้ทำการ Save ใน Hard disk แล้วค่อยทำการย้ายจาก Hard disk ไปที่ Floppy disk

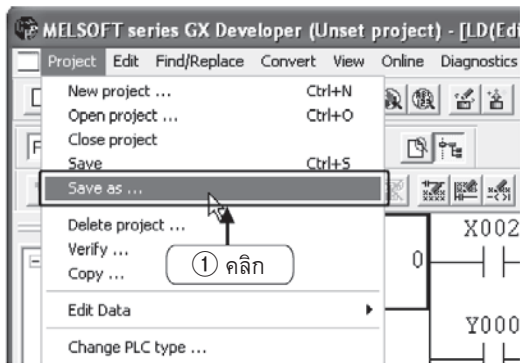
### อ้างอิง

- Project name ไม่สามารถใช้สัญลักษณ์ด้านล่างได้  
/, \, >, <, \*, ?, ", ", |, :, ;, (, ;, \ สามารถใช้ได้เฉพาะ Drive ที่กำหนดเท่านั้น) และพยางค์สุดท้ายของ Project name ห้ามใช้ . (period)
- กรณีทำการตั้งชื่อ Project มากกว่า 8 ตัวอักษรโดยใช้ GX Developer (SW6D5-GPPW เป็นต้นไป) แล้ว Load ด้วย Version GX Developer (ก่อน Version SW2D5-GPPW) ตัวอักษรหลังจากหลักที่ 8 จะไม่แสดง
- จำนวนตัวอักษร Project path + Project name ตัวพิมพ์เล็กไม่เกิน 150 ตัว (ตัวพิมพ์ใหญ่ไม่เกิน 75 ตัว)
- ตัวอักษรของ Title ตัวพิมพ์เล็กไม่เกิน 32 ตัว (ตัวพิมพ์ใหญ่ไม่เกิน 16 ตัว)
- กรณีที่ Project path + Project name มีการเว้นวรรคด้วย ถึงจะดับเบิลคลิก File GPPW.gpj, \*.gps file ใน Explorer GX Developer ก็จะไม่ทำงานปกติ
- กรณีที่ Project path + Project name มีการเว้นวรรคด้วย ให้ทำการเปิด Project ดังนี้ หลังจาก Run ด้วย Explorer → [Project] → [Project open]

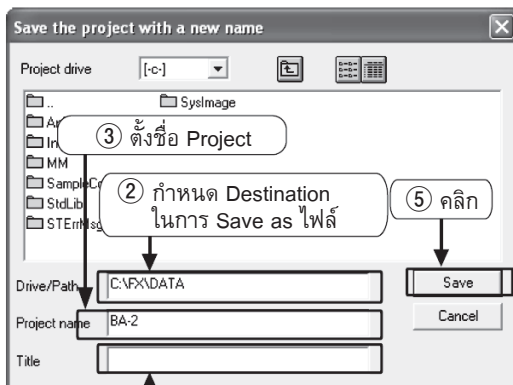
## ภาคผนวก 1.6.2 การ Save as

### Point

กรณีที่ Ladder ยังไม่ได้รับการยืนยัน ให้กด F4



① เลือก [Project] → [Project save as] จากเมนู



- ② ทำการกำหนด Destination ของ Project
- ③ กำหนดชื่อ Project
- ④ สามารถกำหนด Title ได้ เช่น รายละเอียดของ Project (option)
- ⑤ คลิก

④ ใส่ Title ของไฟล์



⑥ จะมี Dialog box ปรากฏขึ้นมา คลิก

วิธีการตั้งชื่อ Drive/Path name และ Project name อ้างอิงหน้าก่อนหน้า

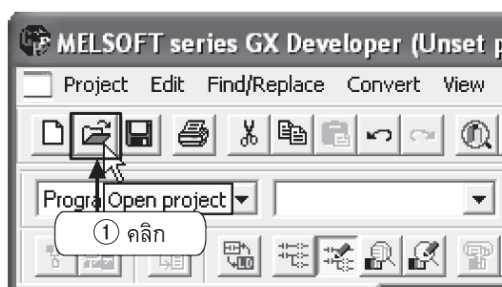
กรณีที่ Save ใน แผ่น Floppy disk ความจุ อาจไม่พอ ให้ทำการ Save ใน Hard disk แล้วค่อยทำการย้ายจาก Hard disk ไปที่ Floppy disk


## ภาคผนวก 1.6.3 การ Reading Project

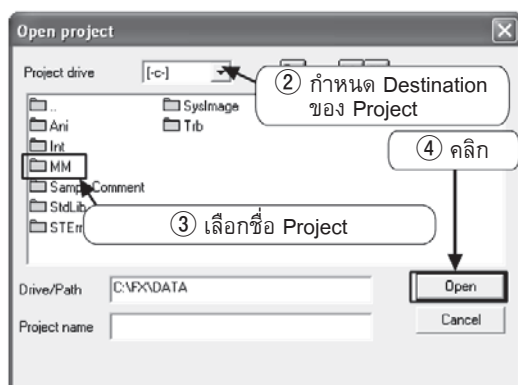
### อ้างอิง

กรณี Reading Project โดยที่เปิดอีก Project ตั้งอยู่ Project นั้นจะถูกปิด

กรณีที่ Project นั้น ยังไม่ได้ Convert ladder หรือยังไม่ได้ Save จะมีข้อความแจ้งเตือนปรากฏขึ้นมา



- ① เลือก  จาก Toolbar หรือเลือก [Project] → [Open project] ( **Ctrl** + **O** ) จากเมนู



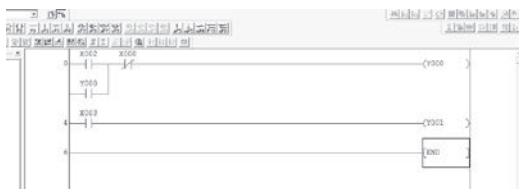
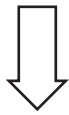
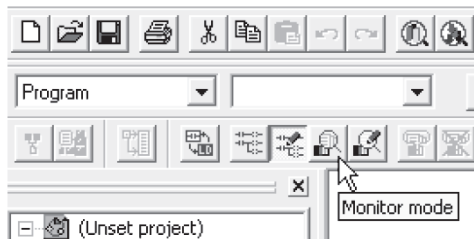
- ② ทำการเลือก Destination ของ Project
- ③ เลือก Read Project
- ④ คลิก **Open** เพื่อเลือก Read Project

# ภาคผนวก 1.7 การทำงานที่จำเป็นในการ Debug Program


เรื่องการอ่าน Program หรือการเชื่อมต่อ PLC ขอให้อ้างอิง [ภาคผนวก 1.4 การเขียน Program ใน PLC]

## ภาคผนวก 1.7.1 Laddet Monitor

ทำการ Monitor สภาพการทำงานของ Coils หรือสภาพการทำงานของ Contact ในแต่ละจุดในขณะที่ Program กำลังแสดงอยู่

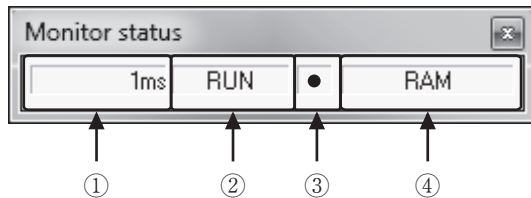


- ① เลือก  จาก Toolbar หรือเลือก [Online] → [Monitor] → [Monitor mode]

- ② ที่ Ladder Monitor จะแสดงค่าปัจจุบัน ของ word device (timer, counter, data register) และสถานะการ ON/OFF ของ Ladder
- ③ เมื่อต้องการจบการทำงานของ Ladder monitor ให้คลิกขวา บนหน้าจอ เลือก [Stop monitor]
- ④ เมื่อต้องการแก้ไขหรือเขียน Program เลือก  จาก Toolbar หรือเลือก [Edit] → [Write mode] จากเมนู

## อ้างอิง

### (1) แสดง Monitor Status Dialog



#### ① Scan time

แสดง Scan time สูงสุดของ PLC Program

#### ② PLC Status

แสดง PLC Status

#### ③ Execute status display

ตำแหน่งที่ทำงาน (Execute) จะกะพริบ

#### ④ Memory type

แสดง Memory type ของ PLC

### (2) วิธีดูการแสดง Status ของ Ladder Monitor

#### ① คำสั่งหน้าสัมผัส (Contact Instruction)

Type \ Input contact	X0 : OFF	X0 : ON
หน้าสัมผัสปกติเปิด a (N.O.Contact)	X000 — — หน้าสัมผัสเปิด (Contact open)	X000 —■— หน้าสัมผัสปิด (Contact close)
หน้าสัมผัสปกติปิด b (N.C.Contact)	X000 —■— หน้าสัมผัสปิด (Contact close)	X000 — — หน้าสัมผัสเปิด (Contact open)

#### ② คำสั่งเอาต์ (Out Instruction)

Type \ Drive status	ไม่ทำงาน/ไม่เคลื่อนไหว	ทำงาน/เคลื่อนไหว
OUT Instruction —( )—	—(Y000)—	—■(Y000)—
SET Instruction. Etc. —[ ]—	—[SET M0]—	—■[SET M0]—

คำสั่ง RST (RST Instruction) คือ การแสดง ON/OFF ของ Device ที่จะ Reset

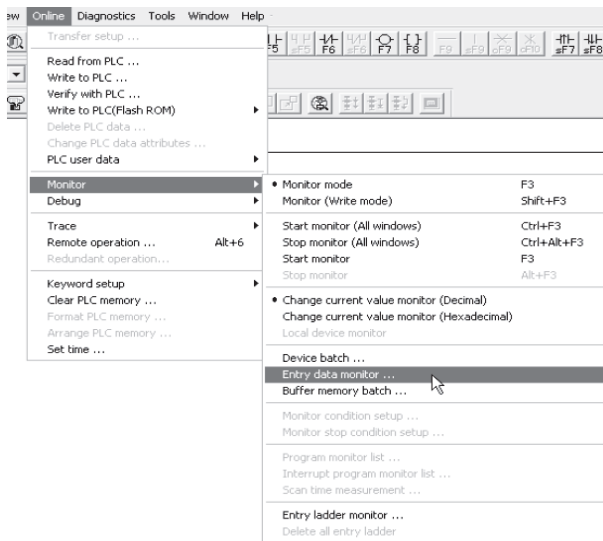
Type \ Drive status	Reset Device OFF	Reset Device ON
RST instruction —[ ]—	—■[RST M0]—	—[RST M0]—



## ภาคผนวก 1.7.2 Device registration monitor

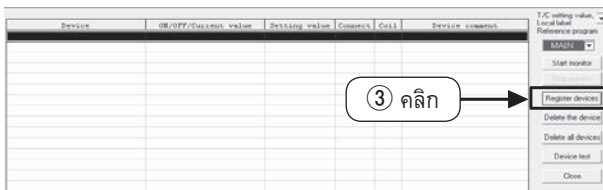
### ① การ Register Device อื่น ๆ

เรื่องการ Register Device อื่น ๆ (Optional Device) ในหน้าจอ Monitor และการ Monitor เฉพาะ Parts ที่ต้องการ

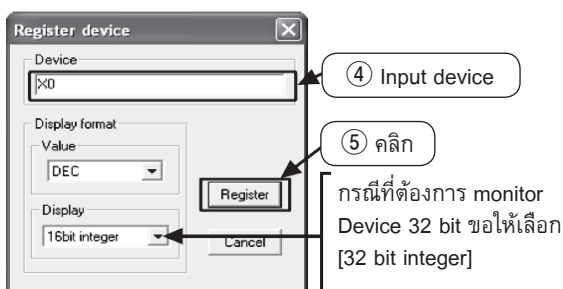


① เปิดหน้าจอ Ladder monitor (อ้างอิงภาคผนวก 1.7.1)

② ทำการเลือก [Online] → [Monitor] → [Entry data monitor] จากเมนูหรือทำการคลิกขวบนหน้าต่าง Program แล้วเลือก [Entry data monitor]



③ คลิก [Register devices] ในหน้าจอ [Entry data monitor]



④ ทำการ Input device number ในหน้าจอ Register device

⑤ คลิก [Register]

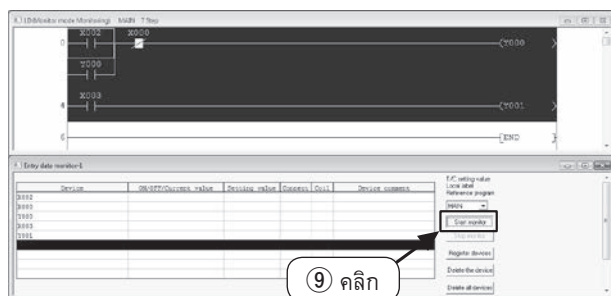
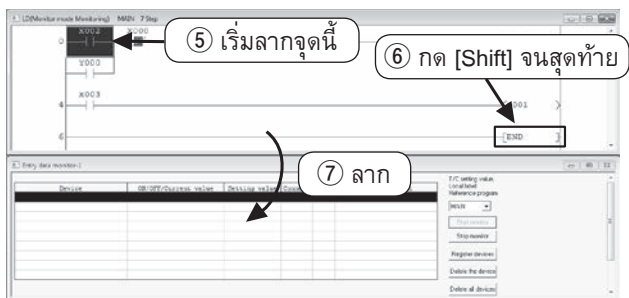
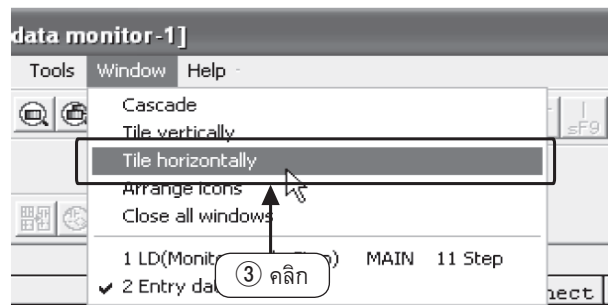


⑥ Device จะถูกบันทึกลงใน Monitor window

⑦ เมื่อคลิก [Start monitor] จะปรากฏ สถานะ ON/OFF ของ Coils และค่าตัวเลข รวมถึงจุดสถานะตามการทำงานของ Device นั้น

## ② การ Register Device ระหว่างที่ Ladder monitor กำลังแสดง

กำหนดขอบเขตการแสดงผลโดยเลือกจากหน้าจอ Ladder monitor และทำการ Register device ที่อยู่ในขอบเขตนั้น



① ไปที่หน้าจอ Monitor mode (อ้างอิงภาคผนวก 1.7.1)

② ทำการเลือก [Online] → [Monitor] → [Entry data monitor] หรือคลิกขวาที่หน้าจอ Ladder เลือก [Entry data monitor] (อ้างอิงหน้าก่อนหน้า)

③ ทำการเลือก [Windows] → [Tile horizontally] จะปรากฏ [Ladder monitor windows] และ [Entry data monitor windows] ขึ้นเรียงกัน (ทำให้ [Entry data monitor windows] อยู่ในสถานะ Stop monitoring)

④ [Ladder monitor windows] และ [Entry data monitor windows] จะปรากฏจากบนลงล่าง

⑤ คลิกที่จุดเริ่มของ Ladder

⑥ กด [Shift] ค้างและคลิกที่จุดสุดท้าย เพื่อทำการเลือกขอบเขต

⑦ ทำการลาก ขอบเขตที่เราเลือกไว้ ลงมาใน [Entry data monitor windows] โดยกด [Ctrl]

⑧ ทำการ Register Device ใน Monitor windows

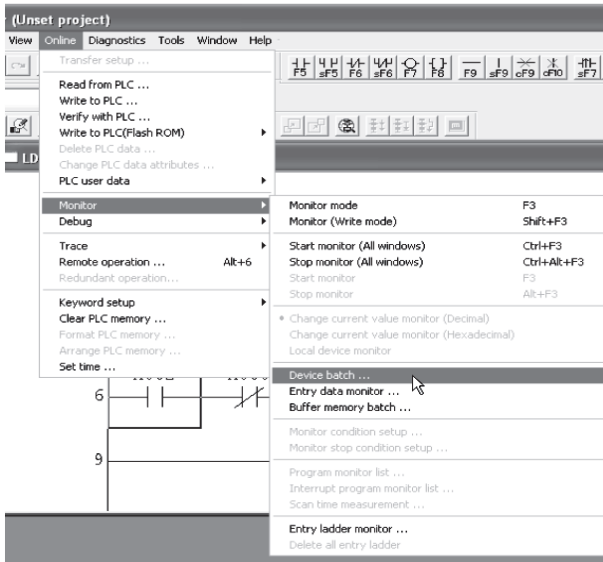
⑨ เมื่อคลิก [Start monitor] จะปรากฏ ตำแหน่ง Coils status ON/OFF และค่า ตามการทำงานของ Device

## ภาคผนวก 1.7.3 Device batch monitor (การเรียกดู Device ทั้งหมด)

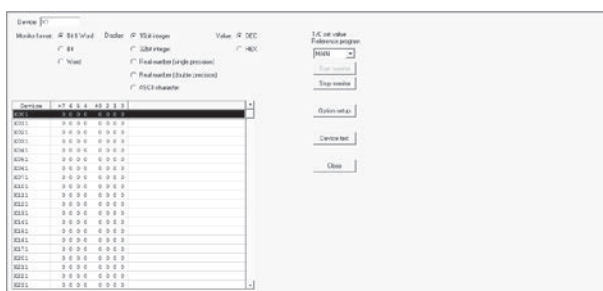
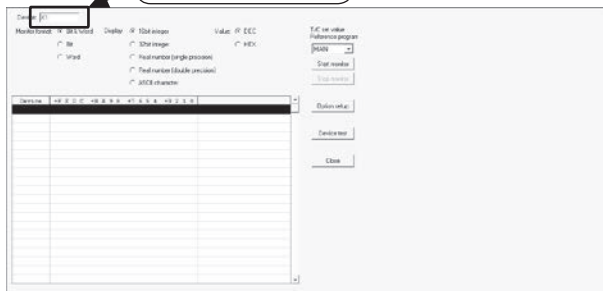
ทำการระบุตัว Device แรก และทำการ Monitor Device ต่อเนื่อง

① ไปที่หน้าจอ Monitor mode (อ้างอิงภาคผนวก 1.7.1)

② ทำการเลือก [Online] → [Monitor] → [Device batch] หรือคลิกขวาที่หน้าจอ Ladder เลือก [Device batch]



③ Device input



③ ทำการใส่หมายเลขแรกของ Device ที่จะ Monitor ใน Windows [Device batch monitor] กด [Enter] หรือคลิก [Start monitor]

④ ปรากฏ Coils Status ON/OFF และค่า ตามการทำงานของ Device

ผนวก  
1

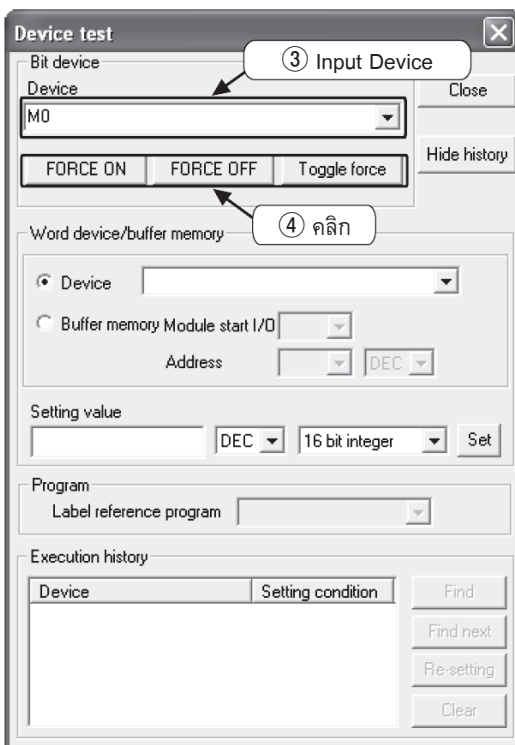
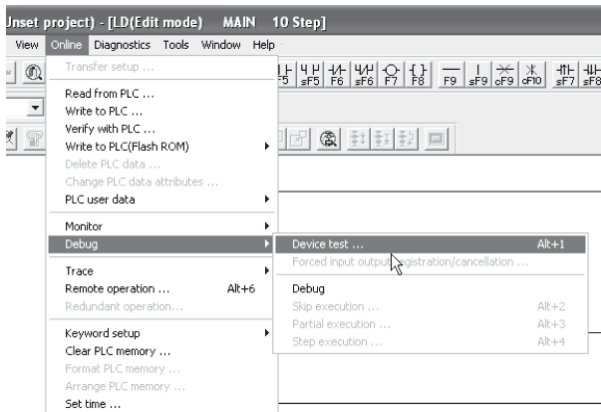
## ภาคผนวก 1.7.4 Device Test

### ① Force ON/OFF (การบังคับเปิด/ปิด)

ทำการ Force ON/OFF Bit Device ของ PLC (เช่น M, Y, T, C) (จะไม่ Force ON/OFF ใน X) ในตอนที่ PLC กำลัง RUN อยู่ จะทำการ ON/OFF เฉพาะเมื่อครบ 1 Cycle เท่านั้น จะต้องให้ Sequence Program ทำงานให้เสร็จสิ้นก่อน ในกรณีที่จะดำเนินการอื่นๆ อย่างเช่น ตรวจสอบ Output ต้องให้ PLC อยู่ในสถานะ STOP ก่อน

① ไปที่หน้าจอ Monitor mode (อ้างอิงภาคผนวก 1.7.1)

② ทำการเลือก [Online] → [Debug] → [Device Test] หรือคลิกขวาที่หน้าจอ Program เลือก [Device test]



③ ทำการ Input หมายเลข Device ที่จะทำการ Force ON/OFF

- ④
- [Force ON] : Device ON
  - [Force OFF] : Device OFF
  - [Toggle force] : Device จะ ON/OFF สลับกันทุกครั้งที่เกิด

#### อ้างอิง

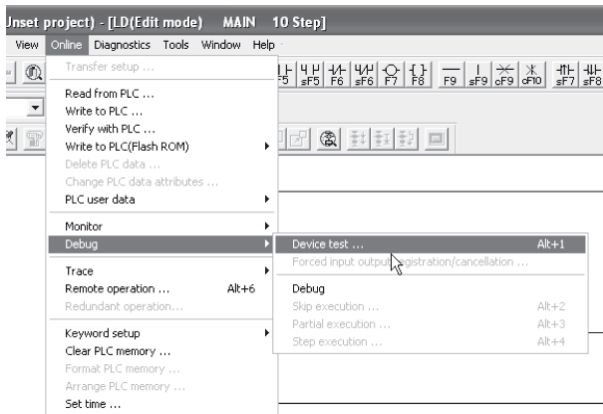
#### Force ON/OFF (Ladder Monitor Windows)

ทำการกด [Shift] ค้าง ใน Bit Device ไหนก็ได้ (Contact, Coil) บน Ladder monitor windows แล้วดับเบิลคลิก จะสามารถเลือก force ON/OFF Device ได้

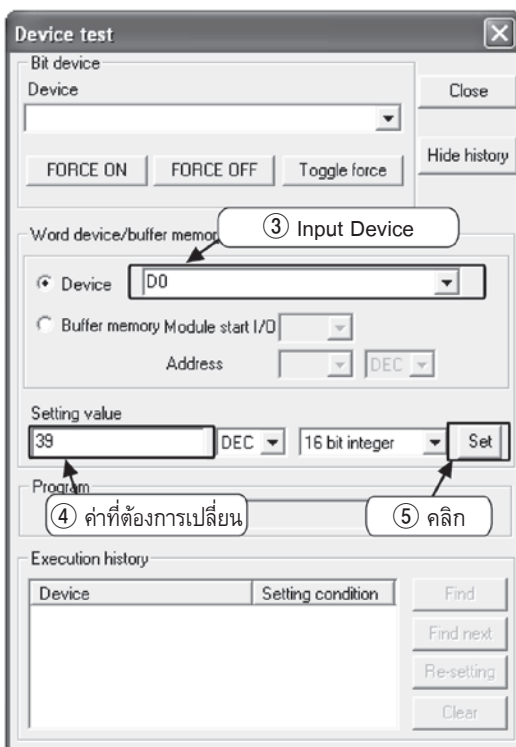
## ② การเปลี่ยนแปลงค่าปัจจุบันของ Word Device

การเปลี่ยนค่าปัจจุบันของ Word Device ของ PLC (เช่น T, C, D เป็นต้น) เป็นค่าที่ต้องการ

① ไปที่หน้าจอ Monitor (อ้างอิงภาคผนวก 1.7.1)



② ทำการเลือก [Online] → [Debug] → [Device Test] หรือคลิกขวาที่หน้าจอ Program เลือก [Device test]



③ ทำการป้อนหมายเลข Device ที่ต้องการเปลี่ยนแปลง


④ ทำการป้อนค่าที่ต้องการเปลี่ยน

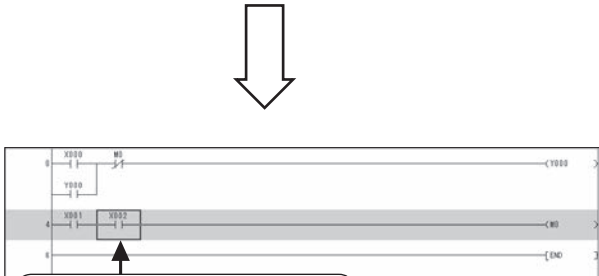
⑤ คลิก [Set]

## ภาคผนวก 1.7.5 การเขียน Program ขณะ Run Program

ทำการเขียน Program ลงใน PLC เฉพาะ Program ส่วนที่แก้ไข ขณะ PLC กำลังทำงาน  
ทำการลดระยะเวลาการเขียน program โดยไม่ต้อง Run Program ทั้งหมด



- ① อธิบายการเพิ่ม หน้าสัมผัส (Contact) ลงที่ Ladder ตาม  
ด้านซ้าย เปลี่ยน Program ให้อยู่ใน Write mode (  )



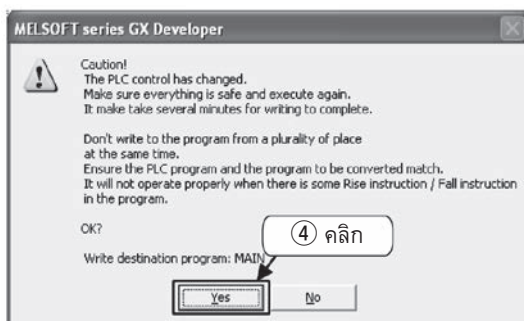
- ② Add contact (เพิ่มหน้าสัมผัส)

- ② ทำการเพิ่มหน้าสัมผัส (Add Contact)  
กรอบ Program จะปรากฏเป็นสีเทา

Write during RUN

[Shift] + [F4]

- ③ เลือก [Shift] + [F4] หรือเลือก [Convert] → [Convert (Write during RUN)] จากเมนู



- ④ คลิก

- ④ เพื่อความปลอดภัยจากการเปลี่ยนการควบคุม PLC ด้วย  
การเปลี่ยน Program จะมี Message แจ้งเตือน ให้คลิก [Yes]



- ⑤ คลิก

- ⑤ เมื่อมี Message แจ้งขึ้นมากว่า "RUN write processing has  
completed" ให้คลิก [OK]

### ข้อควรระวัง

กรณีที่ Program ใน GX Developer กับ Program ใน PLC ก่อนการแก้ไข ไม่ตรงกัน จะไม่สามารถ write ได้ กรณีที่  
ไม่แน่ใจว่า Program ตรงกันหรือไม่ให้ทำการเปรียบเทียบก่อน หรือ Transfer ไปที่ [Write to PLC]

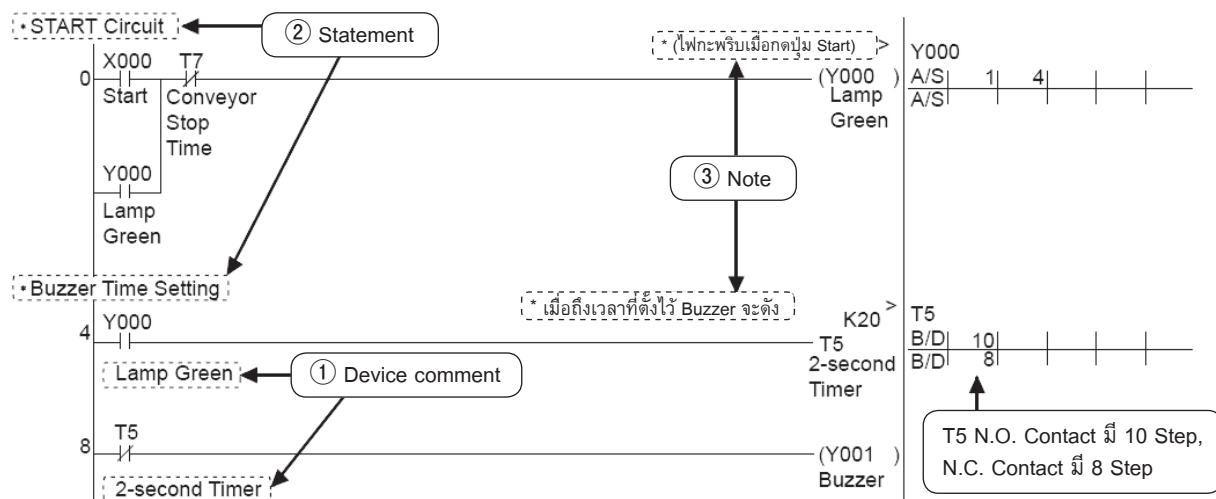
# ภาคผนวก 1.8 การใส่ Comment

## ภาคผนวก 1.8.1 ประเภทของ Comment

สามารถใส่ Comment ได้ 3 ประเภทตามด้านล่าง

ประเภท	วัตถุประสงค์	จำนวนตัวอักษร (ตัวพิมพ์ใหญ่)	หมายเหตุ
① Device comment	Comment แสดง บทบาทและหน้าที่ การใช้งานของแต่ละ Device	16 (ที่ PLC มากสุด 8 ตัวอักษร)	กรณี write ลง PLC จำเป็นต้องทำการ Set ค่า [Comment Capacity] และจำเป็นต้อง Set [Comment range setting] ที่จะ write
② Statement	Comment แสดง บทบาทและหน้าที่ การใช้งานของ Ladder block	32	เป็น Comment เฉพาะด้าน Computer software (ไม่ Download ลง PLC)
③ Note	Comment แสดงบทบาทและหน้าที่ การใช้งานต่อ Output instruction	16	เป็น Comment เฉพาะด้าน Computer software (ไม่ Download ลง PLC)

[ตัวอย่าง Comment]

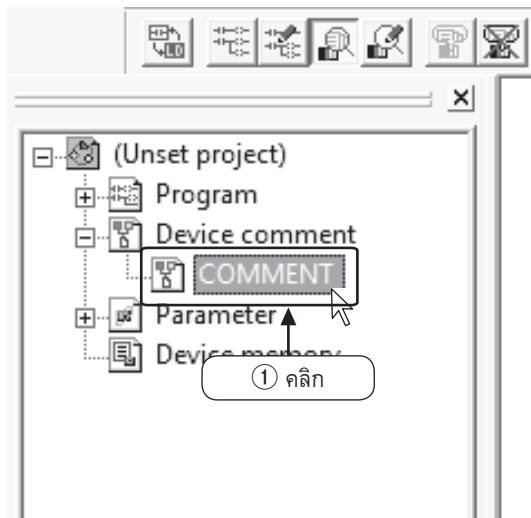


ผนวก  
1

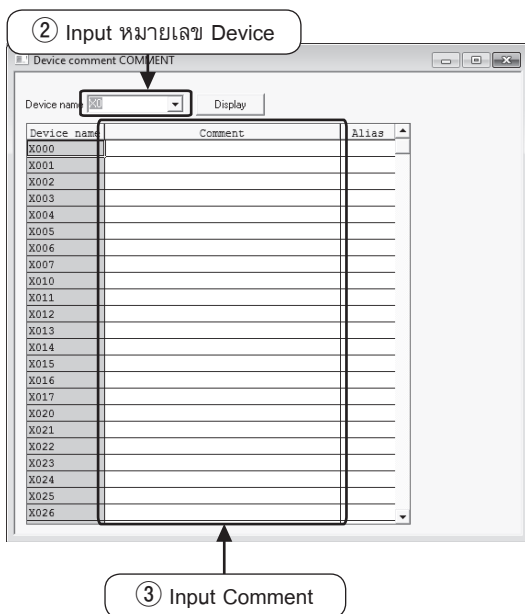
Point	วิธีการแสดง Comment
	<ul style="list-style-type: none"> <li>• ทำการเลือก [Display] → [Display comment] จากเมนู แล้ว Comment จะปรากฏ</li> <li>• เมื่อต้องการยกเลิกการแสดง Comment ให้ดำเนินการตามขั้นตอนด้านบนอีกครั้ง</li> </ul>

## ภาคผนวก 1.8.2 วิธีการสร้าง Device comment

### ① วิธีการ Input Device จาก List



- ① ทำการคลิก [Device Comment] → [COMMENT] จาก Project list



- ② ทำการ Input ตัวเลขต้นของ Device ที่ต้องการจัดทำ Comment ลงใน "Device name" แล้วคลิก [Display]

- ③ ทำการ Input Comment ลงในช่อง [Comment]

- กรณีที่ต้องการ Input comment ของ Device อื่นๆ ให้ทำการใส่หมายเลขของ Device นั้นตามขั้นตอนข้อ ② อีกครั้ง

### ② วิธีการป้อน Comment โดยใช้ Enter Symbol



- ① คลิก [Enter device comment] จาก Toolbar และ ดับเบิลคลิกหมายเลข Ladder ที่ต้องการ Input Comment

- ② วิธีการป้อน Comment โดยใช้ Enter Symbol แล้วคลิก [OK]

- เมื่อดำเนินการเสร็จแล้วให้คลิก [Enter] อีกครั้ง



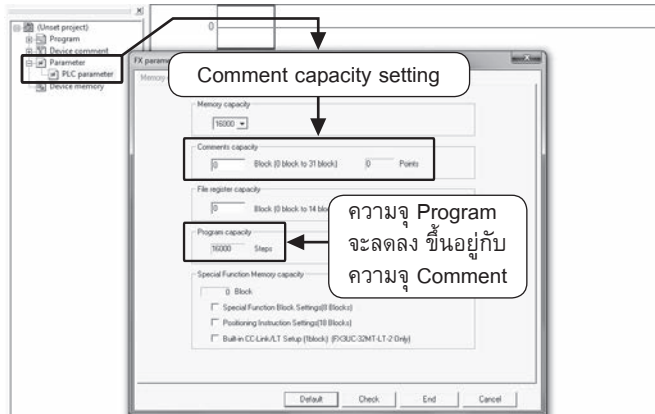
## Point

### การ Set การ Write Device Comment ใน PLC

เพื่อทำการ write Comment ใน PLC จำเป็นต้อง Set “Parameter Setting” และ “Comment range setting”

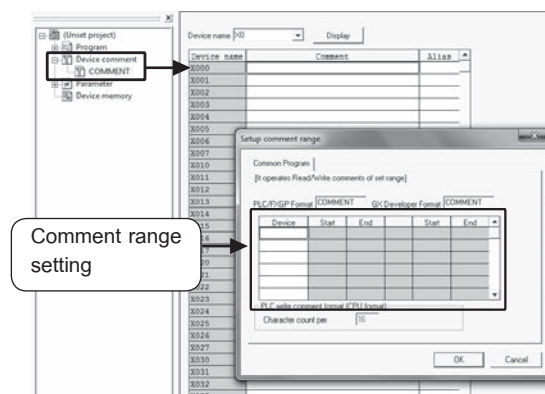
#### ① การ Set parameter

- เลือก [Parameter] → [PLC Parameter]
- ทำการ Set “จำนวน Block” ใน [Comment Capacity]  
ในหนึ่ง Block สามารถบรรจุได้ 50 Comment โดย Program memory ที่ 500 Step



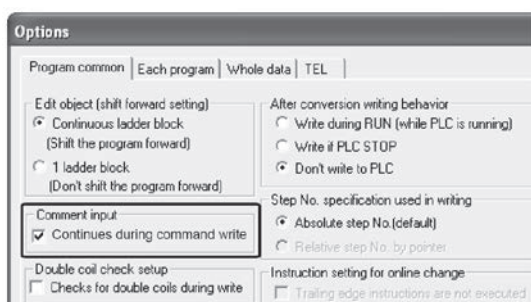
#### ② Comment range setting

- เลือก [Device Comment] → [COMMENT] จะปรากฏหน้าจอให้ Input Comment
- เลือก [Edit] → [Setup Comment range] จากเมนู
- ทำการ Set ประเภท Device และขอบเขตที่จะ Write ที่ PLC ลงใน Setup comment range dialog



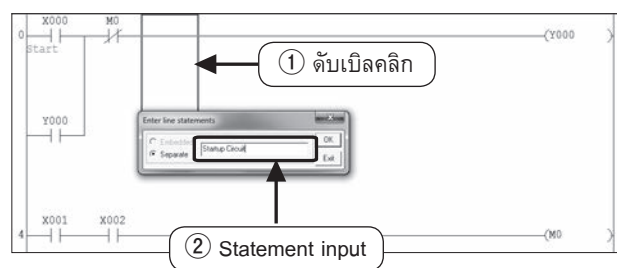
## อ้างอิง


### วิธีการป้อน Comment เมื่อจัดสร้าง Program



ทำการเลือก [Tools] → [Options] จากเมนู ทำการใส่เครื่องหมายถูก “Continues during command write” ในช่อง [Comment input] ใน [Program common] เมื่อดำเนินการ Set ตามนี้แล้ว จะปรากฏหน้าจอข้อ ② “Enter symbol” หลังจาก Input Program ในขั้นตอนการจัดสร้าง Program

## ภาคผนวก 1.8.3 ดำเนินการจัดสร้าง Statement

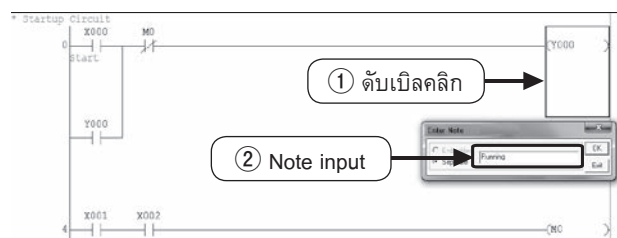



① คลิก  จาก Toolbar ทำการดับเบิลคลิกตรงส่วนไหนก็ได้ของ Ladder block ที่ต้องการจะ input Statement

② ทำการ Input statement ที่หน้าจอ [Enter Line Statement] แล้วคลิก [OK]

- เมื่อดำเนินการเสร็จแล้ว ทำการคลิก  จาก Toolbar อีกครั้ง

## ภาคผนวก 1.8.4 การจัดสร้าง Note



① คลิก  จาก Toolbar ทำการดับเบิลคลิกตรง Output instruction ที่ต้องการจะ Input Note

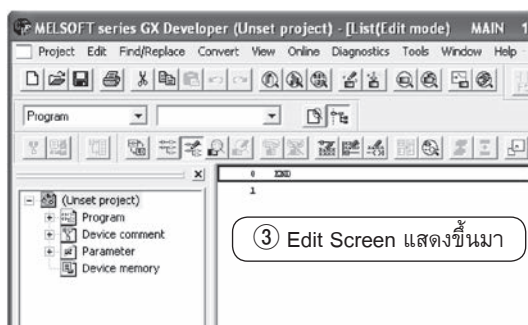
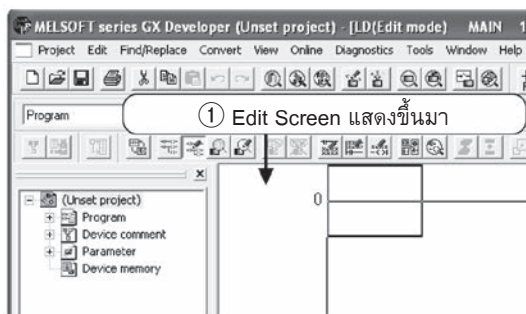
② ทำการ Input Note หน้าจอ "Enter Note" แล้วคลิก [OK]

- เมื่อดำเนินการเสร็จแล้ว ทำการคลิก  จาก Toolbar อีกครั้ง

## ภาคผนวก 1.9 การจัดสร้าง List Program

GX Developer สามารถจัดสร้าง Program โดยใช้ List ได้

### ภาคผนวก 1.9.1 การแสดงหน้าจอ List edit



① จัดสร้าง New project (อ้างอิงภาคผนวก 1.2.2) หรือเลือกเปิด Program จาก Project เดิม

② เลือก  จาก Toolbar หรือเลือก [View] → [Instruction list] จากเมนู

③ จะปรากฏหน้าจอ List edit ขึ้น

กรณีต้องการกลับมาที่หน้าจอ Ladder ให้กด  เลือก [View] → [Ladder] จากเมนู

### ภาคผนวก 1.9.2 วิธีการป้อนคำสั่ง (Instructions)

Start up monitor

0	END
1	

After input list

0	LD	X000	
1	OUT	Y000	
2	LDI	X001	
3	AND	Y000	
4	OUT	M0	
5	LD	M0	
6	OUT	T0	K10
9	OUT	C0	K5
12			

① ทำการ Input ตาม Step ตั้งแต่ Step 0 ในตัวเลขที่ปรากฏของแต่ละ Step เมื่อ Input เข้าไปแล้ว จะทำการเพิ่มให้อัตโนมัติ (วิธีการ Input อ้างอิงหน้าถัดไป)

● วิธีการป้อนคำสั่งพื้นฐานและคำสั่งประยุกต์ (Input Basic instruction และ applied instruction)

Instruction กับหมายเลข Device หรือ Operand ทำการเว้นวรรคด้วยการใส่ "Space"

[ตัวอย่าง Basic instruction]

LD	X0	Enter	}	Connection and OUT Instructions
OUT	Y0	Enter		
LDI	X0	Enter		
AND	Y0	Enter		
OUT	M0	Enter		
LD	M0	Enter	}	Coil instructions for the timer and counter
OUT	T0 K10	Enter		
OUT	C0 K5	Enter		

[ตัวอย่างคำสั่งประยุกต์ (Applied instruction)]

MOV	K1	D0	<input type="text" value="Enter"/>	
CMP	K20	D3	M10	<input type="text" value="Enter"/>

## อ้างอิง


### การใช้งาน Key ในการ Input และ Edit

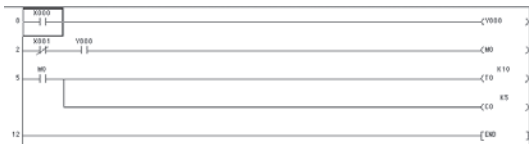
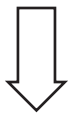
- เมื่อกด  จะเป็นการสลับ Mode [Ovrwrte] / [Insert]
- สามารถทำการ Delete 1 instruction ได้ด้วย
- เมื่อคลิกขวาที่เมาส์ สามารถทำการ [Insert line] / [Delete line]

## ภาคผนวก 1.9.3 การตรวจสอบรายละเอียดในการ Input List

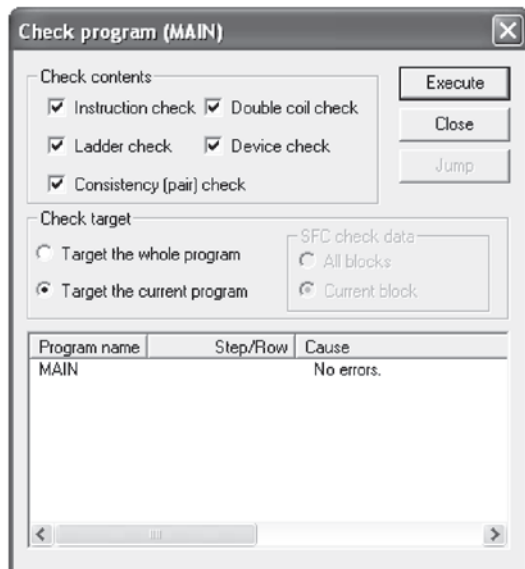
ทำการตรวจสอบ Program ที่จัดสร้างขึ้นว่ามี Error หรือไม่ โดยใช้ Input List ที่หน้าจอ Ladder

0	LD	X000	
1	OUT	Y000	
2	LDI	X001	
3	AND	Y000	
4	OUT	M0	
5	LD	M0	
6	OUT	T0	K10
9	OUT	C0	K5
12	END		
13			

- ① เลือก  จาก Toolbar หรือเลือก [View] → [Ladder] จากเมนู



- ② ขอให้ทำการตรวจสอบ Program ที่จัดสร้างขึ้น ที่ List ที่ Input ไป



- ③ เมื่อทำการ Execute Program check ที่เลือกจากเมนู [Tools] → [Check program] จะสามารถตรวจสอบได้ว่ามี Error หรือไม่ หรือเกิด Error ที่ Step ไหน

# MEMO

## ภาคผนวก 2

### แนะนำตัวอย่างการใช้งาน List programs

## ภาคผนวก 2.1 List programs

### ตัวอย่างที่ 1

#### 《List program : ตัวอย่างที่ 1》

Step	Command
0	LD X001
1	OR Y000
2	ANI X003
3	OUT Y000
4	OUT Y003
5	END

#### 《List program : ตัวอย่างที่ 2》

Step	Command
0	LD X001
1	SET Y000
2	SET Y003
3	LD X003
4	RST Y000
5	RST Y003
6	END

### ตัวอย่างที่ 3

#### 《List program》

Step	Command
0	LD X001
1	OR Y000
2	ANI X003
3	OUT Y000
4	LD Y000
5	AND X000
6	OUT Y002
7	AND Y002
8	MPS
9	AND X005
10	ANI Y003
11	OUT Y001
12	MPP
13	AND X006
14	ANI Y001
15	OUT Y003
16	END

### ตัวอย่างที่ 2

#### 《List program》

Step	Command
0	LD X000
1	AND X001
2	OUT Y001
3	LD X003
4	OR Y003
5	ANI X004
6	OUT Y003
7	LD Y003
8	OUT C0 K5
11	LD C0
12	OUT Y000
13	LD X002
14	RST C0
16	END

### ตัวอย่างที่ 4

#### 《List program》

Step	Command
0	LD X001
1	OR Y003
2	ANI C0
3	OUT Y003
4	LD X004
5	OUT C0 K4
8	LD C0
9	OUT Y001
10	OUT T0 K10
13	AND T0
14	OUT Y002
15	OUT T1 K20
18	LD T1
19	RST C0
21	END



## ตัวอย่างที่ 5

### 《List program》

Step	Command
0	LD X001
1	OR Y000
2	ANI T4
3	OUT Y000
4	OUT T0 K20
7	LD T0
8	OUT Y001
9	OUT T1 K20
12	LD T1
13	OUT Y002
14	OUT T2 K20
17	LD T2
18	OUT Y003
19	OUT T3 K20
22	LD T3
23	OUT Y004
24	OUT T4 K20
27	END

# MEMO

## ภาคผนวก 3

# การใช้งาน Handy programming panel

---

กรณีใช้ Handy programming panel (HPP)...

กรณีที่ใช้ HPP จะสามารถทำการ Edit program list ได้อย่างง่ายดาย  
ในการจัดสร้าง Program หรือ Debug เครื่อง สามารถใช้ HPP ทำการเปลี่ยน Program อย่างง่ายๆ หรือ Set timer / Counter ได้โดยเป็นส่วนที่ Computer ทั่วไปไม่สามารถเปลี่ยนแปลงได้

และยังดำเนินการตรวจสอบ รายละเอียด Errors ได้...

ใช้ Monitor function, Program check function ของ HPP จะสามารถตรวจสอบรายละเอียด Error ได้อย่างสะดวก

สามารถ Storing programs และ Transferring programs ไปที่ PLCs...

HPP รุ่น FX-30P สามารถจุ Sequence programs ได้สูงสุด 15 programs (กรณี Step programs เกิน 32000 steps ได้สูงสุด 7 sequence programs) ในการ Update ของ PLC ที่แตกต่างกันหากไม่จำเป็นต้องใช้ PC และกรณีการเขียน Program เดียวกันลงใน PLC หลายๆตัว ก็ไม่จำเป็นต้องใช้ PC

ใช้กับเครื่องจักรสำหรับต่างประเทศก็ได้...

HPP รุ่น FX-30P เนื่องจากรองรับการเปลี่ยนภาษาทั้งญี่ปุ่น, อังกฤษ, จีน จึงสะดวกในการใช้งานในต่างประเทศ

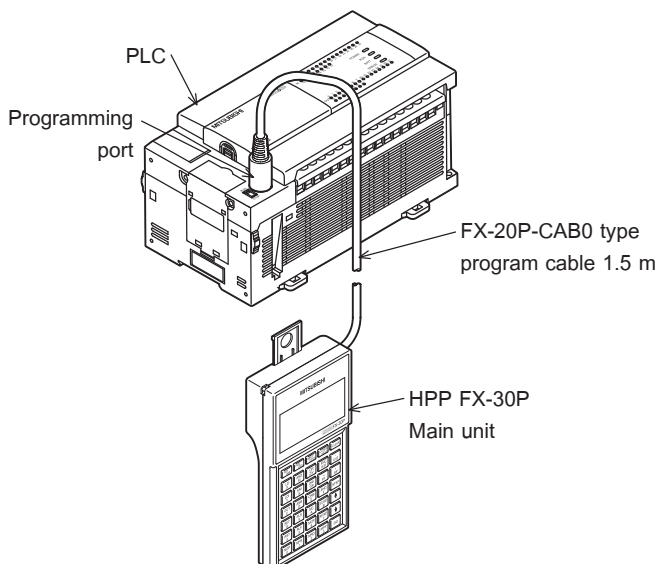
## ภาคผนวก 3.1 การใช้ Handy programming panel (HPP)

กรณีที่ผู้ใช้ HPP จะสามารถทำการ Edit programs list ได้อย่างง่ายดาย

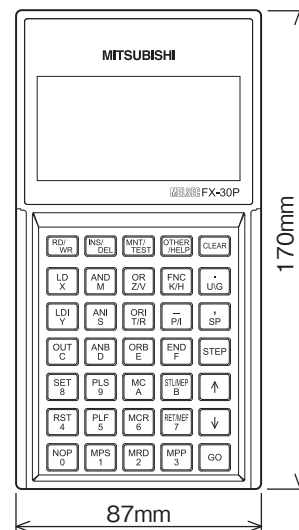
ในการจัดสร้าง Programs หรือ Debug เครื่อง สามารถใช้ HPP ทำการเปลี่ยน Programs อย่างง่ายๆ หรือ Set timers/counters ได้โดยเป็นส่วนที่ Computer ทั่วไปไม่สามารถเปลี่ยนแปลงได้

- ① Functions ที่สะดวกในการ Set up เครื่องจักร
  - ทำการ Edit programs โดยการใช้ Instruction list
  - Devices monitoring, Force ON/OFF
- ② Functions ที่สะดวกในการใช้งาน เมื่อเกิด Error ขึ้น
  - Program check function
  - PC (PLC) diagnostic function
- ③ Functions ที่สะดวกสำหรับการ Maintenance
  - Functions ที่สามารถแก้ไข Parameter timer, Counter
  - Functions ในการ Transfer sequence programs ที่ Update แล้ว (กรณีเป็น Programs capacity เท่ากับหรือต่ำกว่า 32000 steps สามารถอยู่ใน HPP ได้มากกว่า 15 sequence programs)
- ④ Functions ที่สะดวกในการใช้งานกับเครื่องจักรส่งออกต่างประเทศ
  - ข้อความสามารถรองรับได้หลายภาษา เช่น อังกฤษ, จีน, ญี่ปุ่น

[ตัวอย่างการเชื่อมต่อ]



[FX-30P Dimensions]



น้ำหนัก : 0.3 kg

## [ตาราง FX-30P Function]

### ■ Online mode (Functions การเชื่อมต่อโดยตรงไปยัง Memory ของ PLC)

Function		Description
Programming	Read	การ Read sequence program (จาก Program memory ใน PLC → FX-30P Monitor )
	Write	การ Writing sequence program (โดย Key input ลงใน FX-30P → Program memory ใน PLC)
	Insert	ทำการ Insert instruction ลงใน Sequence program (โดย Key input ลงใน FX-30P → Program memory ใน PLC)
	Delete	ทำการยกเลิก Instruction จาก Sequence program (โดย Key input ลงใน FX-30P → Program memory ใน PLC)
Monitor		ทำการ Read สถานะการทำงาน (Program memory → FX-30P screen)
Test		การ Force writing device (โดย Key input ลงใน FX-30P → Program memory)
อื่นๆ	การสลับ Offline	การสลับเปลี่ยน Offline mode
	PLC Diagnostic	ทำการ PLC diagnostic
	Memory cassette transfer	ทำการส่ง Memory cassette
	Parameter	ทำการ Set parameter
	Key word	ทำการ Set key word
	Device conversion	ทำการ Device conversion
	Latch clear	ทำ Latch clear
	Device batch monitor	ทำ Device batch monitor
	BFM batch monitor	ทำการ Monitor buffer memory
	Baud rate	ทำการเปลี่ยน Baud rate
	PLC memory clear	ทำการ Clear memory ใน PLC
	Remote RUN/STOP	ทำการเปลี่ยน Mode RUN/STOP ใน PLC
	PLC clock setting	ทำการ Set PLC timer
	HPP set	ทำ HPP set

### ■ Offline mode (Functions การ Access เข้า RAM Memory ใน FX-30P)

Function		Description
Programming	Read	การ Read sequence program (จาก Built in RAM ของ FX-30P → FX-30P Screen )
	Write	การ Writing sequence program (โดย Key input ลงใน FX-30P → Built in RAM ของ FX-30P)
	Insert	ทำการ Insert instruction ลงใน Sequence program (โดย Key input ลงใน FX-30P → Built in RAM ของ FX-30P)
	Delete	ทำการยกเลิก Instruction จาก Sequence program (โดย Key input ลงใน FX-30P → Built in RAM ของ FX-30P)
อื่นๆ	การสลับ Online	การสลับเปลี่ยน Offline mode
	Program check	ทำการ Check program
	HPP-FX transfer	ทำการ Transfer ระหว่าง 30P RAM - FX PLC
	HPP-PC transfer*	ทำการ Transfer ระหว่าง 30P RAM - Computer
	Parameter	ทำการ Set parameter
	Device conversion	ทำการ Device conversion
	PLC type	ทำการเปลี่ยน PLC Type
	HPP memory clear	ทำการ Clear memory ใน 30P
	Program management	ทำการ Manage program ใน 30P RAM หรือ Flash memory (15 Block)
	HPP setting	ทำ HPP set

\*: การ Transfer ระหว่าง HPP-PC รองรับ Version ตั้งแต่ F/W version 1.10 ขึ้นไป

### ■ Set HPP (Functions ในการ Set FX-30P main unit)

Function	Description
Language	สามารถเลือกภาษาในการแสดงคำสั่งได้ English, Japanese, Chinese
Buzzer sound level	สามารถปรับระดับเสียง Buzzer ได้
LCD contrast	สามารถปรับ LCD contrast ได้
Backlight brightness	สามารถปรับระดับ Backlight ของ LCD brightness ได้
Screen saver	สามารถ Set screen save ได้
HPP protect	สามารถ Set protect ให้ Program ใน 30P ได้
HPP initialization	สามารถ Set 30P ให้กลับสู่สภาพเหมือนตอน Ship ออกจากโรงงานได้
HPP F/W update	สามารถทำการ Update firmware 30P ได้

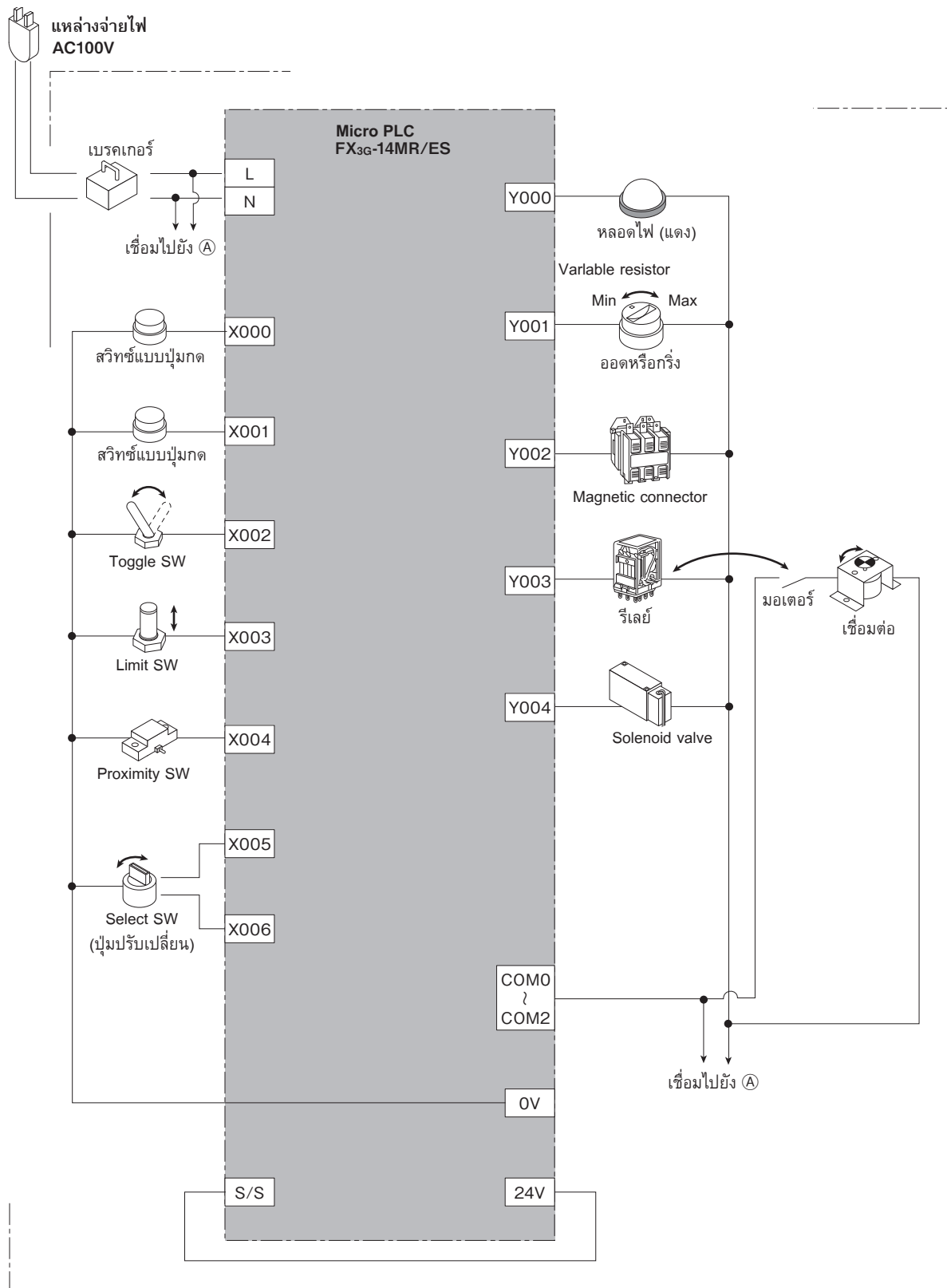
# MEMO

## ภาคผนวก 4

### ผังการเชื่อมต่อ I/O ใน Training kit (Wiring diagram)

---

## ภาคผนวก 4.1 I/O Wiring diagram สำหรับ Training machine



ลักษณะการเชื่อมต่อของ  
FX-I/O-DEMO2



# MEMO

## ประวัติการแก้ไข

วันที่จัดทำ	ลำดับ	รายละเอียด
สิงหาคม 2557	A	จัดพิมพ์ครั้งแรก
พฤษภาคม 2558	B	แก้ไขเนื้อหาการติดต่อสอบถาม
พฤษภาคม 2559	C	การเพิ่มและการแก้ไขรายละเอียดอื่น ๆ



# MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN

สำนักงานจำหน่ายประจำประเทศไทย  
บริษัท มิตซูบิชิ อิเล็กทริก แพลทอร์รี่ ออโตเมชัน (ประเทศไทย) จำกัด  
เลขที่ 896/19 และ 20 อาคารเอสวี ซิตี ออฟฟิศทาวเวอร์ 1 ชั้น 12  
ถนนพระราม 3 แขวงบางโพงพาง เขตยานนาวา กรุงเทพมหานคร 10210  
โทร : 02-682-6522-31 แฟกซ์ : 02-682-6020  
URL : [http://www.MitsubishiElectric.com/fa/th\\_th](http://www.MitsubishiElectric.com/fa/th_th)