



# PLC ขั้นสูง - GX Works2

หลักสูตรนี้ให้ความรู้เกี่ยวกับฟังก์ชันต่างๆ ในการปรับปรุงสภาพแวดล้อมในการพัฒนาโปรแกรมในไซตการออกแบบ ที่ประสบปัญหาเกี่ยวกับ "ผลผลิต" "คุณภาพ" "การจัดการโครงการ" และ "มาตรการด้านความปลอดภัย" หลักสูตรนี้จัดทำขึ้นสำหรับโปรแกรมเมอร์ที่เขียนการทำงานตามลำดับที่ใช้งาน MELSOFT GX Works2 อยู่แล้ว

»  
บทนำ

## วัตถุประสงค์ของหลักสูตร

หลักสูตรนี้ให้ความรู้เกี่ยวกับฟังก์ชันต่างๆ ในการปรับปรุงสภาพแวดล้อมในการพัฒนาโปรแกรมในไซต์การออกแบบ ที่ประสบปัญหาเกี่ยวกับ "ผลผลิต" "คุณภาพ" "การจัดการโครงการ" และ "มาตรการด้านความปลอดภัย" หลักสูตรนี้จัดทำขึ้นสำหรับโปรแกรมเมอร์ที่เขียนการทำงานตามลำดับที่ใช้งาน MELSOFT GX Works2 อยู่แล้ว

## หน้า

## โครงสร้างของหลักสูตร

เนื้อหาของหลักสูตรนี้มีดังนี้  
เราขอแนะนำให้คุณเริ่มต้นจากบทที่ 1

บทที่ 1 - การปรับปรุงสภาพแวดล้อมการพัฒนาโปรแกรมโดยใช้ GX Works2

คุณสามารถเรียนรู้ถึงปัญหาที่พบในไซต์การออกแบบและสภาพแวดล้อมในการพัฒนาโปรแกรมที่จำเป็นต้องแก้ไข

บทที่ 2 - การตั้งโปรแกรม

คุณสามารถเรียนรู้เกี่ยวกับฟังก์ชันต่างๆ ที่ใช้ในการตั้งโปรแกรม

บทที่ 3 - การแก้จุดบกพร่อง

คุณสามารถเรียนรู้เกี่ยวกับฟังก์ชันต่างๆ ที่ใช้ในการแก้จุดบกพร่อง

บทที่ 4 - การจัดการโครงการและมาตรการด้านความปลอดภัย

คุณสามารถเรียนรู้เกี่ยวกับฟังก์ชันต่างๆ ในการจัดการโครงการ และมาตรการด้านความปลอดภัยในขั้นตอนของการพัฒนาและบำรุงรักษาหลังจากเริ่มใช้งานระบบ

แบบทดสอบประเมินผล

คะแนนที่ผ่านหลักสูตร: 60% ขึ้นไป

»  
บทนำ

## วิธีการใช้งานเครื่องมือการเรียนรู้อิเล็กทรอนิกส์นี้

ไปที่หน้าถัดไป		ไปที่หน้าถัดไป
กลับไปยังหน้าที่แล้ว		กลับไปยังหน้าที่แล้ว
เลื่อนไปยังหน้าที่ต้องการ		ระบบจะแสดง "สารบัญ" ช่วยให้คุณสามารถไปยังหน้าต่างๆ ได้
ออกจากการเรียนรู้		ออกจากการเรียนรู้ ออกจากการเรียนรู้ระบบจะปิดหน้าต่าง เช่น หน้าจอ "เนื้อหา" และการเรียนรู้

»  
บทนำ

## ข้อควรระวังในการใช้งาน

## ข้อควรระวังด้านความปลอดภัย

เมื่อคุณเรียนรู้โดยการใช้งานผลิตภัณฑ์จริง โปรดอ่านข้อควรระวังต่างๆ ในคู่มือการใช้งานอย่างละเอียดให้เข้าใจ

## ข้อควรระวังในหลักสูตรนี้

- หน้าจอที่แสดงของซอฟต์แวร์เวอร์ชันที่คุณใช้อาจจะแตกต่างจากในหลักสูตรนี้

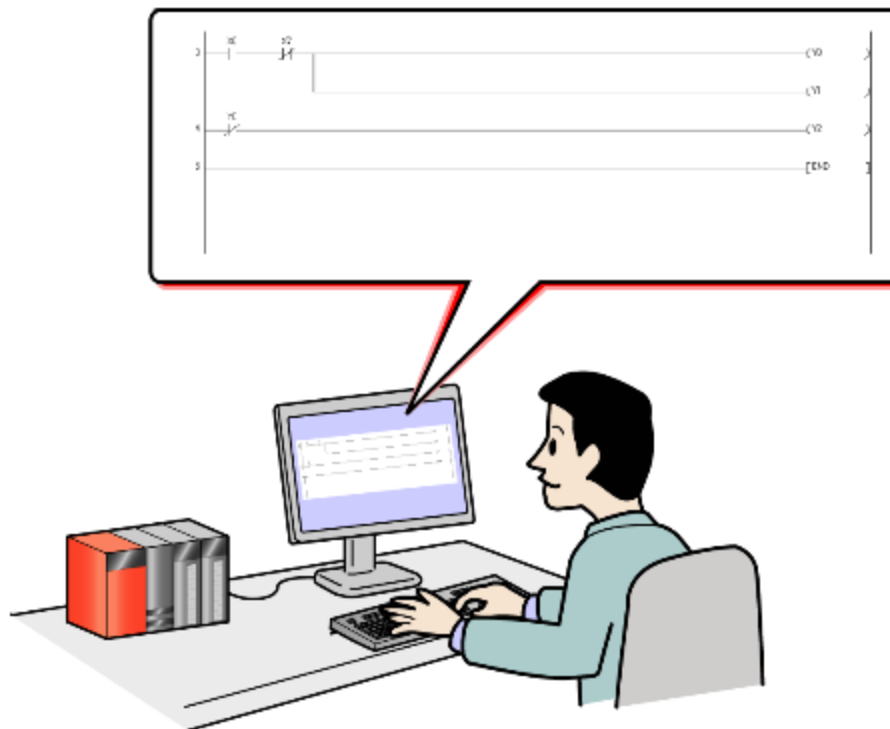
# บทที่ 1 การปรับปรุงสภาพแวดล้อมการพัฒนาโปรแกรมโดยใช้ GX Works2

## ขั้นตอนการเรียนรู้ในบทที่ 1

หลักสูตรนี้จัดทำขึ้นสำหรับโปรแกรมเมอร์ที่ทำงานด้านการพัฒนาโปรแกรมเชิงลำดับ คุณจะได้เรียนรู้วิธีการใช้ฟังก์ชันต่างๆ อย่างมีประสิทธิภาพใน MELSOFT GX Works2 เพื่อแก้ไขปัญหาที่พบในไซต์การออกแบบที่เกี่ยวข้องกับ "ผลผลิต" "คุณภาพ" "การจัดการโครงการ" และ "ความปลอดภัย"

ในบทที่ 1 คุณจะได้เรียนรู้เกี่ยวกับปัญหาที่พบในไซต์การออกแบบ และสภาพแวดล้อมในการพัฒนาโปรแกรมที่จำเป็นต้องแก้ไข

- 1.1 ปัญหาที่พบในไซต์การออกแบบ
  - 1.1.1 การปรับปรุงผลผลิต
  - 1.1.2 การปรับปรุงคุณภาพ
  - 1.1.3 การจัดการโครงการ
  - 1.1.4 มาตรฐานด้านความปลอดภัย
- 1.2 กระบวนการเรียนรู้



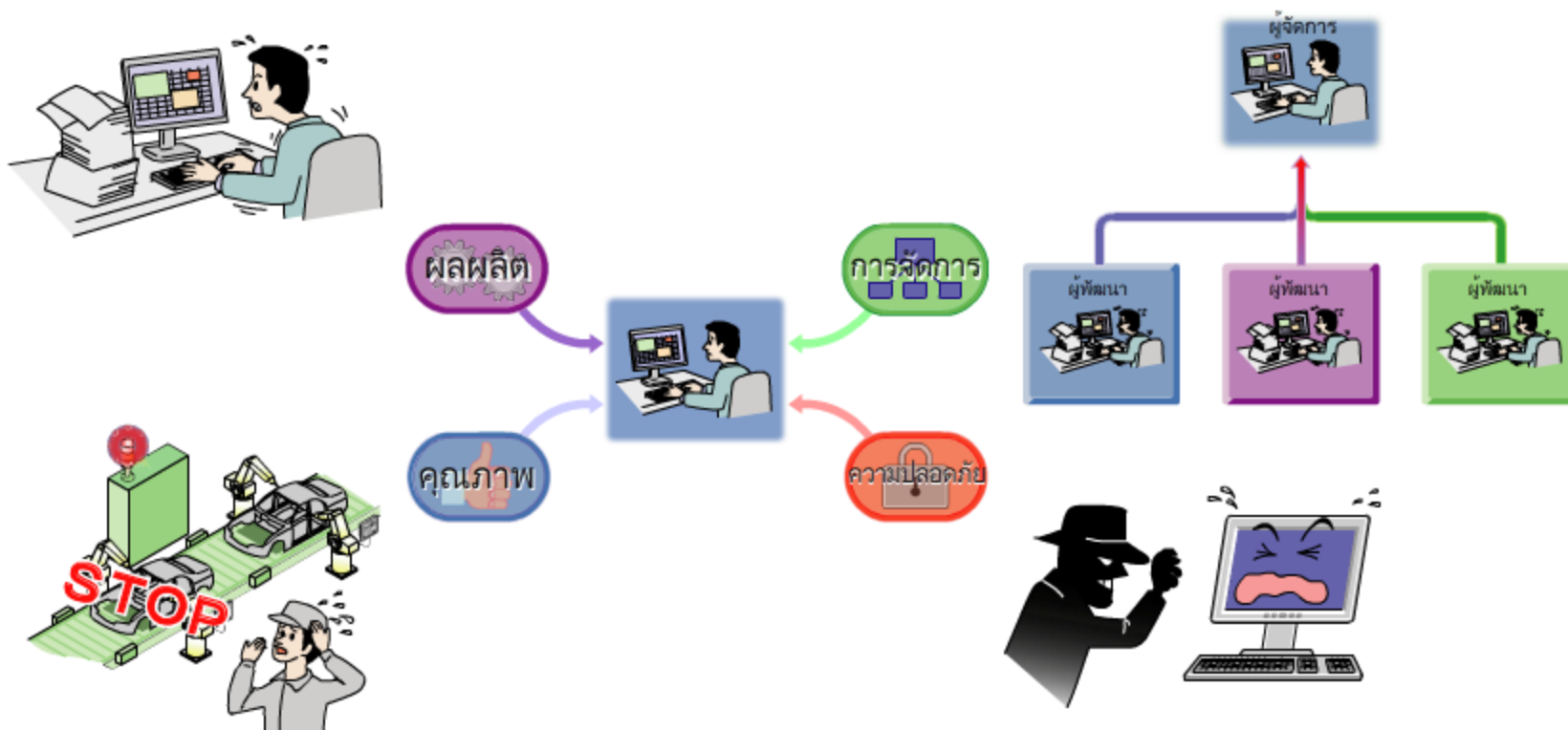
## 1.1

## ปัญหาที่พบในไซตการออกแบบ

ขณะที่ไซตการออกแบบจำเป็นต้องหาวิธีการเพิ่มผลผลิตเพื่อลดต้นทุนในการพัฒนาโปรแกรม พวกเขาต้องตรวจสอบให้มั่นใจว่าโปรแกรมที่พัฒนาขึ้นมีคุณภาพสูงอย่างแท้จริง

นอกจากนี้ไซตการออกแบบยังต้องรับผิดชอบการจัดการโครงการที่นำไปสู่การพัฒนาทีม และทำให้มั่นใจว่าสามารถกู้คืนได้รวดเร็วในกรณีที่มีปัญหาเกิดขึ้น

มาตรการด้านความปลอดภัยเป็นอีกสิ่งหนึ่งที่จำเป็น เนื่องจากโปรแกรมเชิงลำดับมีทั้งความรู้และข้อมูลที่สำคัญรวมอยู่ด้วย



## 1.1.1 การปรับปรุงผลผลิต

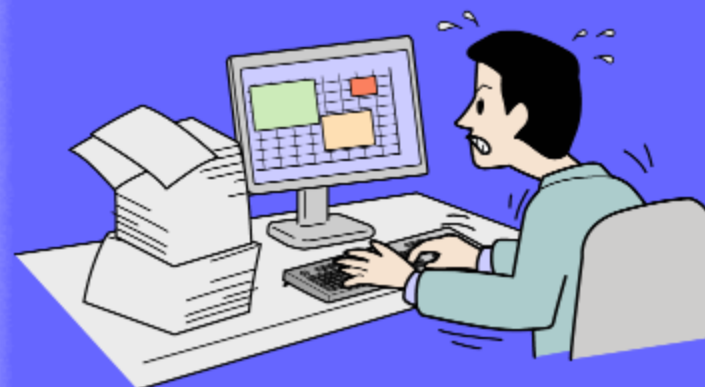
ไซตการออกแบบจำเป็นต้องพัฒนาโปรแกรมเชิงลำดับเพื่อจัดการกับการประมวลผลที่ซับซ้อนและมีสเกลใหญ่  
อย่างไรก็ดี ต้นทุนของการพัฒนาเพิ่มขึ้นตามระยะเวลาของการพัฒนา  
ดังนั้น คุณจึงถูกขอให้เพิ่มผลผลิตในการพัฒนาโปรแกรมเพื่อที่จะลดต้นทุนการพัฒนา

GX Works2 มีฟังก์ชันต่อไปนี้ที่สามารถช่วยแก้ไขปัญหาค้นให้คุณได้:

- ลานเบล
- บูล็อคฟังก์ชัน
- ค่าเริ่มต้นอุปกรณ์และหน่วยความจำอุปกรณ์\*
- ข้อความที่มีโครงสร้างในบรรทัด
- นำเข้าจากหมายเหตุตัวอย่าง

\* MELSEC-F ซีรีส์ ไม่สามารถกำหนดค่าอุปกรณ์เริ่มต้นได้

ผลผลิต





## 1.1.2 การปรับปรุงคุณภาพ

จุดบกพร่องในโปรแกรมเชิงลำดับอาจเป็นสาเหตุให้ระบบหยุดทำงาน เกิดความผิดปกติ หรือเกิดอุบัติเหตุ ซึ่งนำไปสู่การหยุดการผลิตและส่งผลกระทบต่อสูญเสียกำไรและความเชื่อถือได้ในกระบวนการผลิต คุณถูกขอให้พัฒนาโปรแกรมที่มีคุณภาพที่ไม่มีจุดบกพร่องใดๆ

GX Works2 มีฟังก์ชันต่อไปนี้ที่สามารถช่วยแก้ไขปัญหาค้นหาให้คุณได้:

- ลาเบล
- บูลีนคฟังก์ชัน
- ค่าเริ่มต้นอุปกรณ์และหน่วยความจำอุปกรณ์\*
- หมายเหตุ
- นำเข้าจากหมายเหตุตัวอย่าง
- ดู
- ข้อมูลอ้างอิง
- การติดตามการสุ่มตัวอย่าง\*
- การทดสอบอุปกรณ์ตามเงื่อนไขการทำงาน\*
- ฟังก์ชันการดำเนินการขั้นตอน\*
- การตั้งค่าระบบ I/O



\* MELSEC-F ซีรีส์ ไม่สามารถใช้ในการตั้งค่าเริ่มต้นของอุปกรณ์ การติดตามการสุ่มตัวอย่าง การทดสอบอุปกรณ์ตามเงื่อนไขการทำงาน และฟังก์ชันการดำเนินการขั้นตอน

## 1.1.3 การจัดการโครงการ

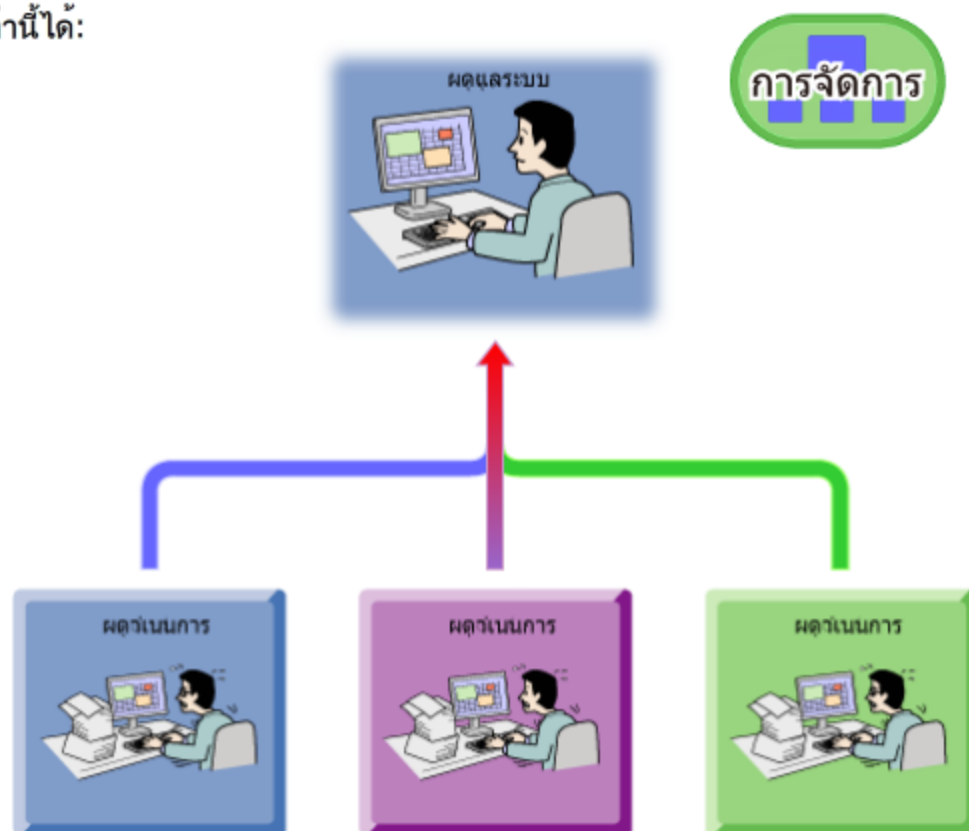
โปรแกรมเชิงลำดับสูงเลใหญ่มักได้รับการพัฒนาโดยทีมโปรแกรมเมอร์

จากมุมมองด้านสิทธิการจัดการของแต่ละบุคคลและการเก็บข้อมูลเป็นความลับ จำเป็นต้องกำหนดข้อจำกัดสำหรับช่วงข้อมูลที่เข้าถึงได้และฟังก์ชันที่สามารถใช้งานได้

นอกจากนี้ เพื่อให้มั่นใจว่าสามารถกู้คืนข้อมูลได้รวดเร็วในกรณีที่เกิดการสูญหายในโปรแกรม เนื่องจากความล้มเหลวตัวควบคุมแบบโปรแกรม คุณจึงถูกขอให้ใช้การจัดการเวอร์ชันของโปรแกรมและการสำรองข้อมูลโปรแกรมตามระยะเวลา

GX Works2 มีฟังก์ชันต่อไปนี้ที่สามารถช่วยแก้ไขปัญหเหล่านี้ได้:

- ความปลอดภัย
- ประวัติการแก้ไข
- ตรวจสอบด้วย PC



## 1.1.4 มาตรการด้านความปลอดภัย

โปรแกรมเชิงลำดับมีทั้งความรู้และข้อมูลเชิงกลยุทธ์ที่สำคัญรวมอยู่ด้วย การรั่วไหลของข้อมูลวิธีการและข้อมูลจากโปรแกรมออกสู่ภายนอกอาจส่งผลกระทบต่อธุรกิจ การแก้ไขโปรแกรมโดยไม่ได้รับอนุญาตอาจนำไปสู่ปัญหาการผลิต เช่น ระบบหยุดทำงาน จำเป็นต้องมีมาตรการด้านความปลอดภัยที่เหมาะสมในการป้องกันปัญหาเหล่านี้

GX Works2 มีฟังก์ชันต่อไปนี้ที่สามารถช่วยแก้ไขปัญหาดังกล่าวได้:

- ความปลอดภัย



## 1.2 กระบวนการเรียนรู้

ในหลักสูตรนี้ คุณจะได้เรียนรู้เกี่ยวกับฟังก์ชันต่างๆ ใน GX Works2 โดยทำตามกระบวนการพัฒนาระบบจริง

(1) การตั้งโปรแกรม ..... บทที่ 2



(2) การแก้จุดบกพร่อง ..... บทที่ 3



(3) การจัดการโครงการและมาตรการด้านความปลอดภัย ..... บทที่ 4

### <คำอธิบายไอคอนต่างๆ>

ไอคอนที่แสดงในหน้าต่างๆ ในบทที่ 2, 3 และ 4 แทนฟังก์ชันการปรับปรุงที่เกี่ยวข้องดังแสดงด้านล่าง



ฟังก์ชันที่ใช้ปรับปรุงผลผลิตของโปรแกรม



ฟังก์ชันที่ใช้ปรับปรุงคุณภาพของโปรแกรม



ฟังก์ชันที่ใช้สำหรับการจัดการโครงการ



ฟังก์ชันที่ใช้สำหรับมาตรการด้านความปลอดภัย

## บทที่ 2 การตั้งโปรแกรม

### ขั้นตอนการเรียนรู้ในบทที่ 2

ในบทที่ 2 คุณจะเรียนรู้เกี่ยวกับฟังก์ชันต่างๆ ที่ใช้สำหรับการตั้งโปรแกรม GX Works2 มีฟังก์ชันต่างๆ มากมายเพื่อให้มั่นใจว่าการตั้งโปรแกรมมีประสิทธิภาพและปรับปรุงคุณภาพของโปรแกรม

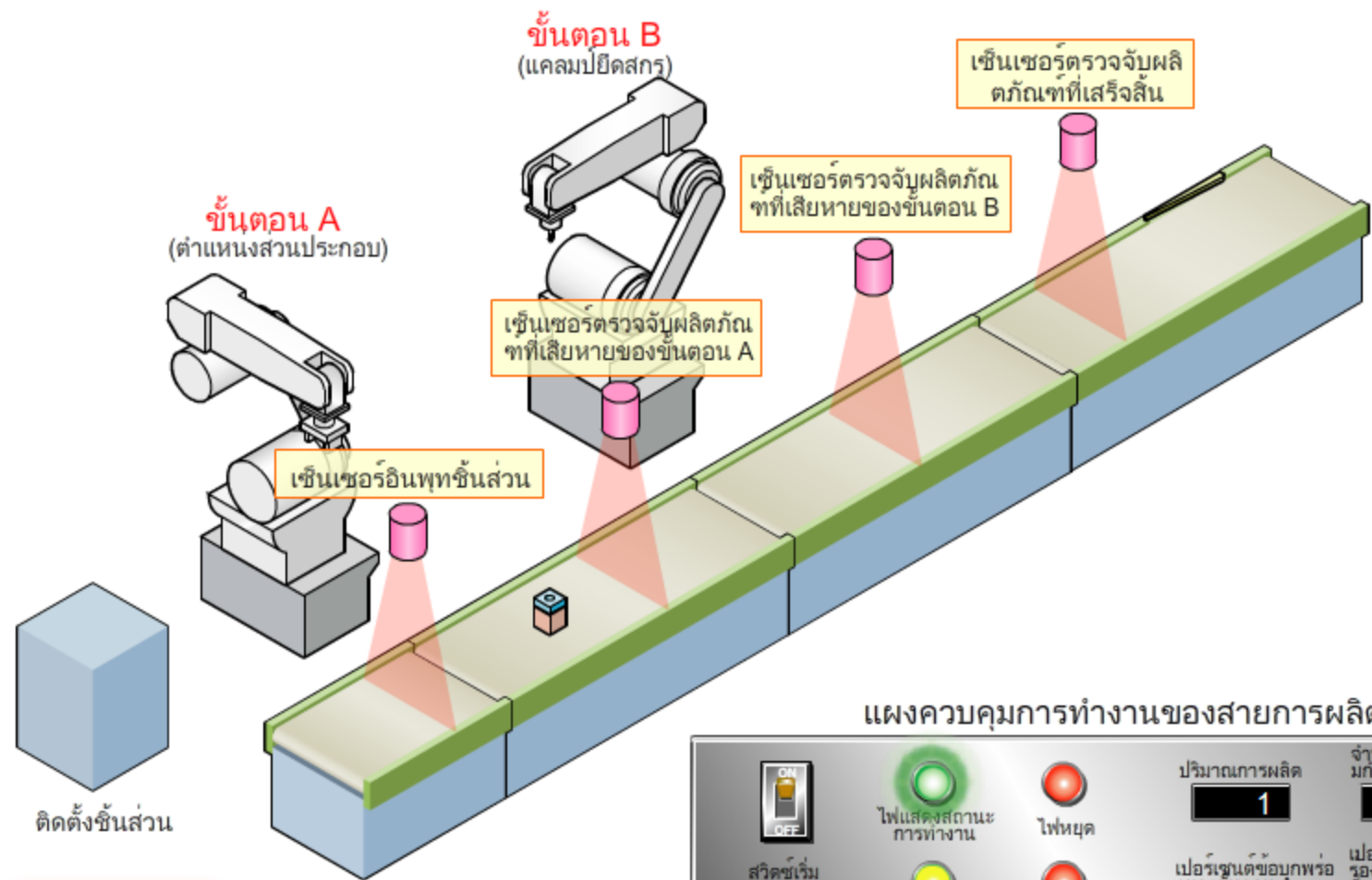
- 2.1 การเปลี่ยนชื่ออุปกรณ์เป็นชื่อที่เกี่ยวข้องกับการใช้งาน
  - 2.1.1 ชนิดลาเบล
  - 2.1.2 ชนิดการใช้งานลาเบลและค่าที่บันทึก
  - 2.1.3 การลงทะเบียนลาเบล
  - 2.1.4 การกำหนดลาเบลให้กับอุปกรณ์โดยอัตโนมัติ
- 2.2 การจัดการแลตเตอร์บล็อกที่ซับซ้อนเป็นบล็อกฟังก์ชันเพื่อความแตกต่าง
  - 2.2.1 การสร้างและวางบล็อกฟังก์ชัน
  - 2.2.2 การใช้ไลบรารีบล็อกฟังก์ชัน
- 2.3 การเปลี่ยนค่าเริ่มต้นของอุปกรณ์โดยไม่ต้องแก้ไขโปรแกรม
- 2.4 การปรับโปรแกรมแลตเตอร์ให้ง่ายขึ้น
- 2.5 การสร้างโปรแกรมที่เข้าใจง่ายและอ่านได้ง่าย
  - 2.5.1 การเขียนหมายเหตุในแต่ละแลตเตอร์บล็อก
  - 2.5.2 การเขียนหมายเหตุในแต่ละเอาต์พุต (คอยล์ คำแนะนำการใช้งาน)
- 2.6 การปรับให้อ่านโปรแกรมที่มีรีเลย์/รีจิสเตอร์พิเศษได้ง่ายขึ้น

# บทที่ 2 การแนะนำระบบที่ใช้สำหรับการเรียนรู้ในหลักสูตรนี้

ผลผลิต

คุณภาพ

หลักสูตรนี้ใช้ระบบการตั้งสมมติฐานต่อไปนี้เพื่อช่วยให้คุณได้เรียนรู้วิธีการใช้งานฟังก์ชันต่างๆ ใน GX Works2:



- ▶ การผลิตตามปกติ
- ▶ ความผิดพลาดเปอร์เซ็นต์ของช่องของขั้นตอน A
- ▶ ความผิดพลาดเปอร์เซ็นต์ของช่องของขั้นตอน B

เมื่อกดปุ่ม "Back to Top" (กลับไปทางด้านบน) จะกลับไปทำการเลือกการทำ **กลับไปทางด้านบน**

## แผงควบคุมการทำงานของสายการผลิต

<input type="checkbox"/> ON <input type="checkbox"/> OFF สวิตช์เริ่ม	<input checked="" type="checkbox"/> ไฟแสดงสถานะการทำงาน	<input type="checkbox"/> ไฟหยุด	ปริมาณการผลิต <b>1</b>	จำนวนการผลิตตามกำหนดเวลา <b>20</b>
<input type="checkbox"/> ON <input type="checkbox"/> OFF รีเซ็ตปริมาณการผลิต	<input checked="" type="checkbox"/> ปริมาณการผลิตที่ ได้รับ	<input type="checkbox"/> ไฟแสดงความผิดพลาดเปอร์เซ็นต์ของช่องของขั้นตอน A	เปอร์เซ็นต์ของช่องของขั้นตอน A ที่ยอมรับได้ <b>0</b>	เปอร์เซ็นต์ของช่องของขั้นตอน A ที่ยอมรับได้ <b>10</b>
	<input type="checkbox"/> ไฟแสดงความผิดพลาดเปอร์เซ็นต์ของช่องของขั้นตอน B	<input type="checkbox"/> ไฟแสดงความผิดพลาดเปอร์เซ็นต์ของช่องของขั้นตอน B	เปอร์เซ็นต์ของช่องของขั้นตอน B ที่ยอมรับได้ <b>0</b>	เปอร์เซ็นต์ของช่องของขั้นตอน B ที่ยอมรับได้ <b>5</b>

## 2.1

### การเปลี่ยนชื่ออุปกรณ์เป็นชื่อที่เกี่ยวข้องกับการใช้งาน

**อุปกรณ์** ที่ใช้ในโปรแกรมเชิงลำดับจะถูกกำหนดชื่อ ซึ่งจะประกอบด้วยตัวอักษรตามด้วยตัวเลข เช่น "M0" หรือ "D5" ดังนั้น ชื่ออุปกรณ์จะไม่เกี่ยวข้องกับการใช้งานอุปกรณ์

โปรแกรมสเกลใหญ่ใช้อุปกรณ์มากมายหลายชนิด ซึ่งหมายความว่าในระหว่างการตั้งโปรแกรม คุณต้องตรวจสอบเอกสารการออกแบบระบบอยู่ตลอดเวลาเพื่อกำหนดการใช้งานอุปกรณ์ วิธีนี้ส่งผลให้ประสิทธิภาพการทำงานลดลง และส่งผลกระทบต่อคุณภาพของโปรแกรมเนื่องจากมีข้อผิดพลาดเกิดขึ้นในการเลือกอุปกรณ์

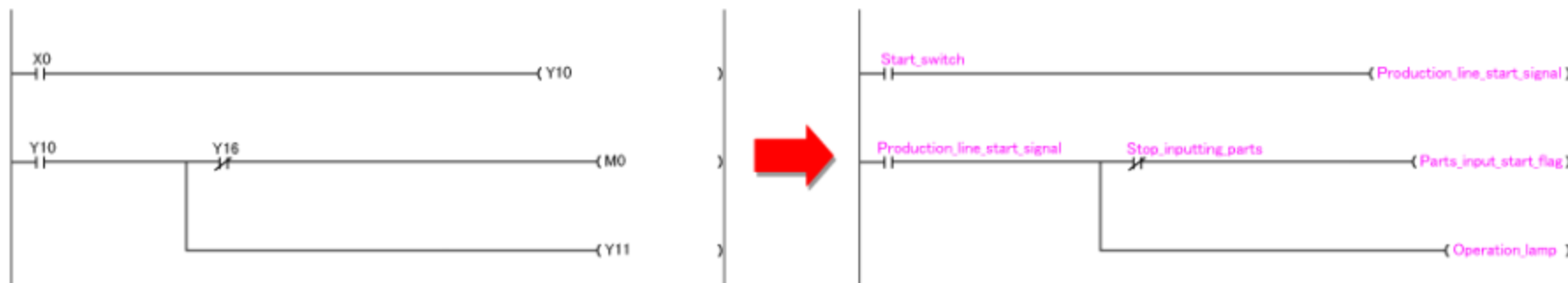
**"ลาเบล"** สามารถใช้แก้ไขปัญหานี้ได้

คุณสามารถใช้ชื่อ (ลาเบล) ที่ระบุการใช้งานจริง เช่น **"สัญญาณเริ่มสายการผลิต"** แทนการใช้ชื่ออุปกรณ์ สำหรับชื่อชนิดนี้ สามารถใช้อักษรภาษาญี่ปุ่น (ฮิรางานะ กะตากานะ และคันจิ) นอกเหนือจากอักขระตัวเลขผสมตัวอักษร

<ข้อความสั่งสำหรับการตั้งค่า **"M0"** ที่มีสัญญาณเริ่มสายการผลิตไปที่ **ON** ตามคำสั่ง **SET**>

อุปกรณ์	—[ SET M0 ]—
ลาเบล	—[ SET Production line start signal ]— (ตั้งค่าสัญญาณเริ่มสายการผลิต)

การใช้งานลาเบลชนิดนี้จะช่วยให้สร้างโปรแกรมที่อ่านได้ง่ายยิ่งขึ้น ปรับปรุงประสิทธิภาพการพัฒนาโปรแกรม และป้องกันการเกิดข้อผิดพลาดกับอินพุตอุปกรณ์



## 2.1.1 ชนิดลาเบล

ผลผลิต

คุณภาพ

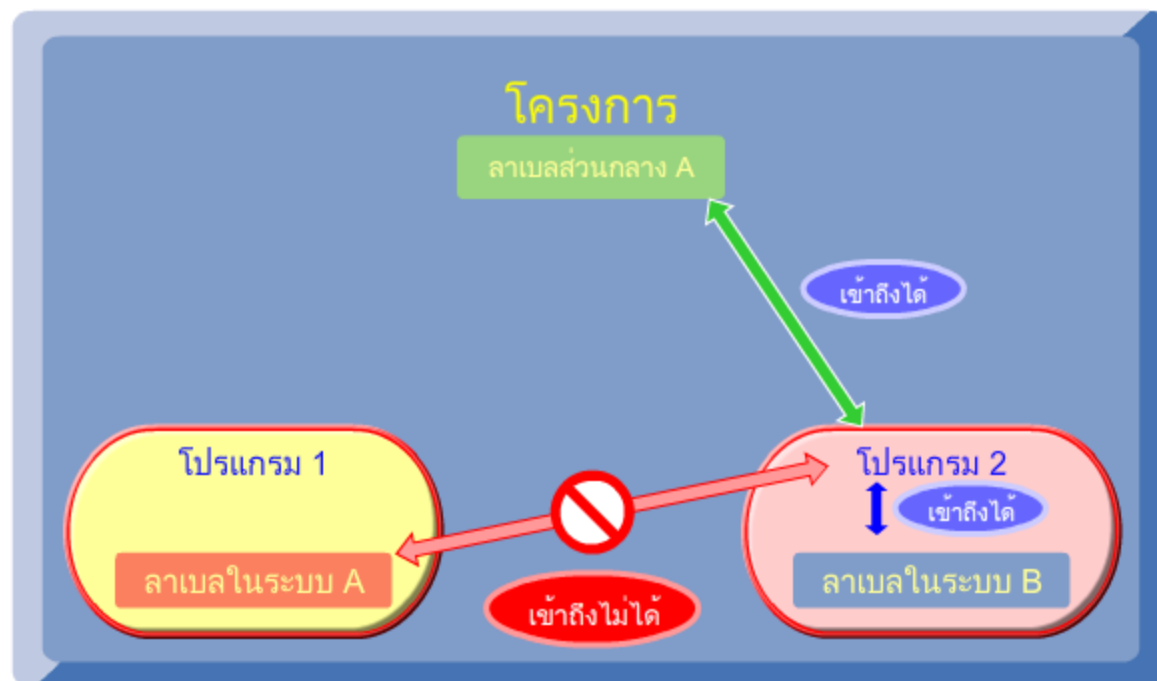
มีลาเบลสองชนิดดังนี้: "ลาเบลส่วนกลาง" และ "ลาเบลในระบบ"

### <ลาเบลส่วนกลาง>

ลาเบลส่วนกลาง ใช้สำหรับโครงการทั้งหมดและสามารถเข้าถึงได้จากทุกโปรแกรมในโครงการนั้น

### <ลาเบลในระบบ>

ลาเบลในระบบใช้สำหรับโปรแกรมเฉพาะและสามารถเข้าถึงได้โดยโปรแกรมที่ลงทะเบียนลาเบลนั้นเท่านั้น



มีสองปุ่มคือ "Program 1" (โปรแกรม 1) และ "Program 2" (โปรแกรม 2) กดโปรแกรมใดโปรแกรมหนึ่งเพื่อดูว่าสามารถเข้าถึงลาเบลทั้งสองชนิดได้หรือไม่



## 2.1.2

## ชนิดการใช้งานลาเบลและค่าที่บันทึก

ผลผลิต

คุณภาพ

เมื่อลงทะเบียนลาเบล ให้ระบุการใช้งานลาเบลและชนิดของค่าที่สามารถบันทึกได้โดยใช้ **"ชั้น"** และ **"ชนิดข้อมูล"**

## &lt; ชั้น &gt;

ชั้นระบุช่วงการใช้และการใช้งานลาเบล

ชั้นที่สามารถเลือกได้หลากหลายขึ้นอยู่กับชนิดของลาเบล

ชั้น	วัตถุประสงค์ในการใช้	พื้นที่การตั้งค่าลาเบล		
		ลาเบล ส่วนกลาง	ภายใน โปรแกรม ลาเบล	บล็อกฟังก์ชัน ลาเบลในระบบ
VAR_GLOBAL	ลาเบลทั่วไปที่สามารถใช้ได้โดยโปรแกรมและบล็อกฟังก์ชันในโครงการ	○	×	×
VAR_GLOBAL_ CONSTANT	ลาเบลทั่วไปที่มีค่าคงที่ที่สามารถใช้ได้โดยโปรแกรมและบล็อกฟังก์ชันในโครงการ	○	×	×
VAR	ลาเบลที่สามารถใช้โดยโปรแกรมและบล็อกฟังก์ชันที่ต้องการใช้ลาเบล	×	○	○
VAR_CONSTANT	ลาเบลที่มีค่าคงที่ที่สามารถใช้โดยโปรแกรมและบล็อกฟังก์ชันที่ต้องการใช้ลาเบล	×	○	○
VAR_RETAIN	ลาเบล (ประเภทแลตซ์) ที่สามารถใช้โดยโปรแกรมและบล็อกฟังก์ชันที่ต้องการใช้ลาเบล	×	○	○
VAR_INPUT	ลาเบลที่ใช้สำหรับอินพุทบล็อกฟังก์ชันที่ต้องการใช้ลาเบล * ไม่สามารถเปลี่ยนแปลงค่าในส่วนประกอบของ โปรแกรมได้	×	×	○
VAR_OUTPUT	ลาเบลใช้สำหรับเอาท์พุทของบล็อกฟังก์ชันที่ต้องการใช้ลาเบล	×	×	○
VAR_IN_OUT	ลาเบลที่ใช้สำหรับอินพุท และเอาท์พุทของบล็อกฟังก์ชันที่ต้องการใช้ลาเบล * สามารถเปลี่ยนแปลงค่าในส่วนประกอบของโปรแกรมได้	×	×	○

## 2.1.2

## ชนิดการใช้งานลาเบลและค่าที่บันทึก

ผลผลิต

คุณภาพ

## &lt;ชนิดข้อมูล&gt;

ชนิดข้อมูลหมายถึง ชนิดค่าที่บันทึกไว้ในลาเบล

ชนิดข้อมูลที่กำหนดให้กับลาเบล ระบุชนิดและช่วงค่าที่สามารถบันทึกในลาเบลและอุปกรณ์ที่เกี่ยวข้องได้

ชนิดข้อมูลที่สามารถใช้ได้กับโปรแกรมแลตเตอร์แสดงไว้ด้านล่าง

ชนิดข้อมูล	คำอธิบาย	ความยาวบิต	ช่วงค่าที่บันทึก
บิต	บันทึก ON หรือ OFF ตรงกับอุปกรณ์ "M"	1 บิต	1: ON, 0: OFF (เปิด)
ค่า (สัญลักษณ์)	บันทึกจำนวนเต็มที่ไม่มีเศษส่วน ตรงกับอุปกรณ์ "D"	16 บิต	-32768 ถึง 32767
ค่าคู่ (สัญลักษณ์)		32 บิต	-2147483648 ถึง 2147483647
FLOAT (single precision)	บันทึกจำนวนจริงรวมเศษส่วน ตรงกับอุปกรณ์ "D"	16 บิต	$-2^{128}$ ถึง $-2^{-126}$ , 0, $2^{-126}$ ถึง $2^{128}$
FLOAT (double precision)		32 บิต	$-2^{1024}$ ถึง $-2^{-1022}$ , 0, $2^{-1022}$ ถึง $2^{1024}$
สตริง	บันทึกชุดตัวอักษร ตรงกับอุปกรณ์ "D"	ตัวแปร	สูงสุด 255 ตัวอักษร
ตัวจับเวลา	เปิดเมื่อถึงเวลาที่ระบุ ตรงกับตัวจับเวลา "T"	-	-
ตัวจับเวลาสะสม	เปิดเมื่อถึงเวลาที่ระบุ ตรงกับอุปกรณ์ตัวจับเวลาสะสม "ST"	-	-
ตัวนับ	เปิดเมื่อถึงจำนวนนับที่ระบุ ตรงกับอุปกรณ์ตัวนับ "C"	-	-
ตัวชี้	บันทึกตำแหน่งเริ่มชั่วคราว ตรงกับอุปกรณ์ตัวชี้ "P"	-	-

# 2.1.3

## การลงทะเบียนลาเบล



MELSOFT Series GX Works2 C:\Sequential Programs\e-learning\Robot control - [Local Label Setting MAIN [PRG] ]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation icons: File, Edit, View, Run, Stop, etc.

Navigation

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
  - Global1
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool
- Structured Data Type
- Local Device Commen

- Device Memory
- Device Initial Value

	Class	Label Name	Data Type	Constant
1	VAR	Production_qty_ attained	Bit	...
2				...
3				...
4				...
5				...
6				...
7				...
8				...
9				...
10				...
11				...
12				...
13				...
14				...
15				...
16				...
17				...
18				...
19				...
20				...
21				...
22				...
23				...
24				...
25				...
26				...



## 2.1.4

## การกำหนดลาเบลให้กับอุปกรณ์โดยอัตโนมัติ

ผลลัพธ์

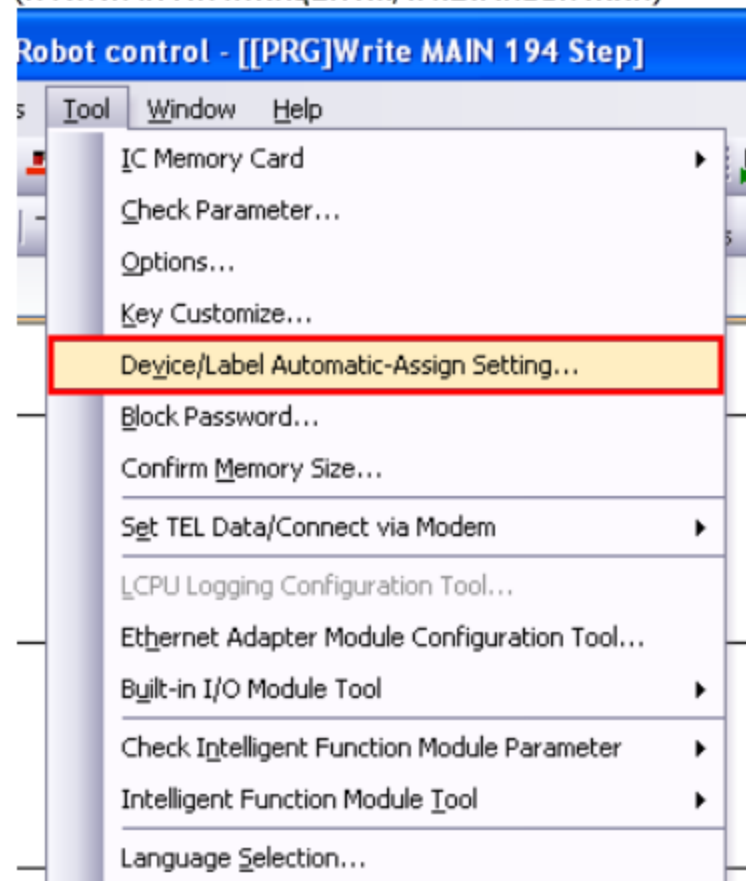
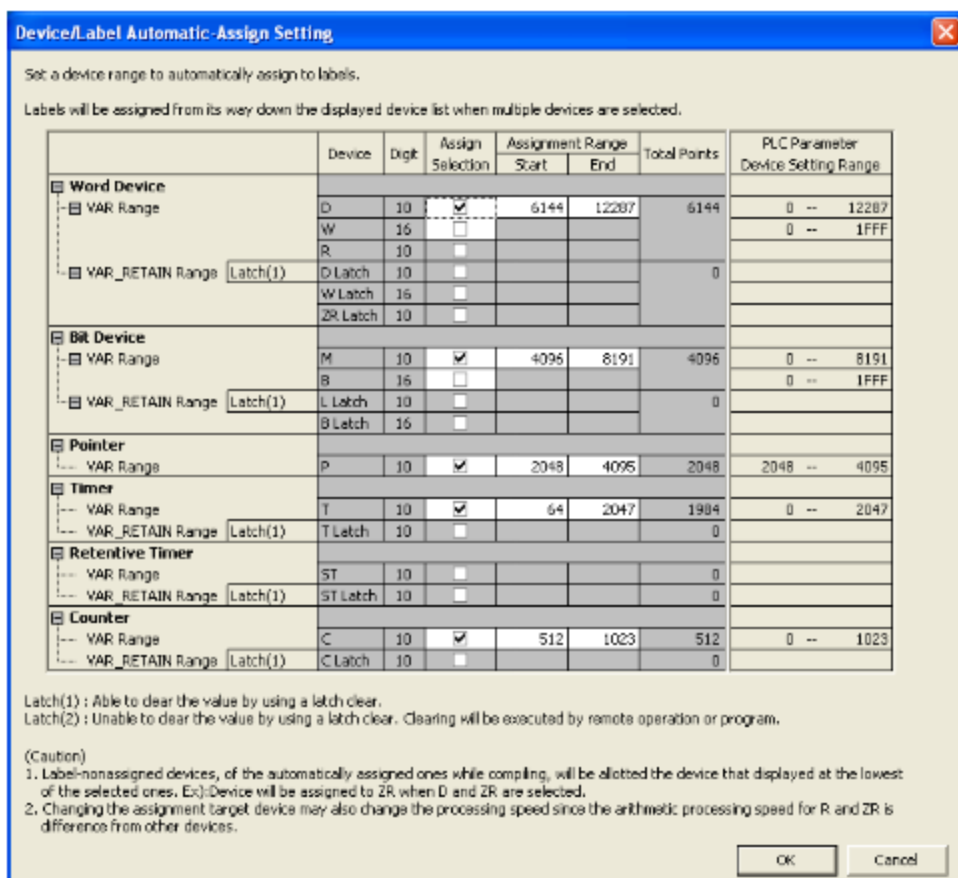
คุณภาพ

การแปลงโปรแกรมโดยอัตโนมัติจะกำหนดอุปกรณ์ที่เหมาะสมให้กับลาเบลตามชั้นและชนิดข้อมูลเมื่อใช้ลาเบล ไม่จำเป็นต้องทราบว่ากำหนดให้กับอุปกรณ์ใด

ใช้ **"Device/Label Automatic-Assign Setting"** (การตั้งค่าการกำหนดอุปกรณ์/ลาเบลโดยอัตโนมัติ) ในการเปลี่ยนช่วงอุปกรณ์ที่กำหนดให้กับลาเบล

\* หน้าจอนี้เป็นหน้าต่างการตั้งค่าอุปกรณ์การกำหนดโดยอัตโนมัติสำหรับ MELSEC-Q และ MELSEC-L ซีรีส์ หน้าจออาจแตกต่างกันไปจากใน MELSEC-F ซีรีส์

<การเริ่ม **"Device/Label Automatic-Assign Setting"** (การตั้งค่าการกำหนดอุปกรณ์/ลาเบลโดยอัตโนมัติ)>  
จากเมนู GX Works2 เลือก **[Tool]** (เครื่องมือ) – **[Device/Label Automatic-Assign Setting]** (การตั้งค่าการกำหนดอุปกรณ์/ลาเบลโดยอัตโนมัติ)



## 2.2

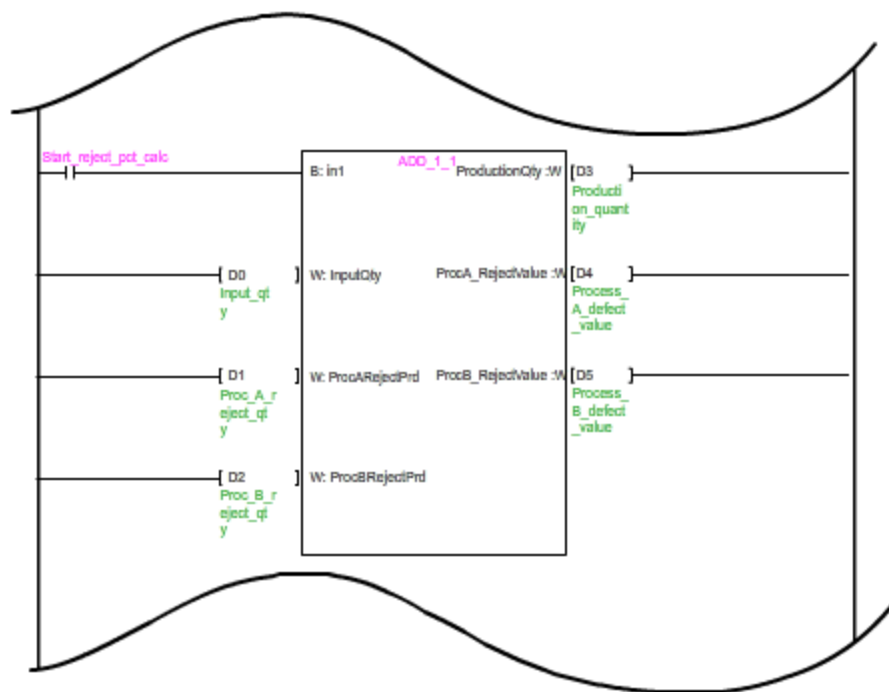
## การจัดการแลตเตอร์บล็อกที่ใช้บ่อยเป็นบล็อกฟังก์ชันเพื่อความแตกต่าง

ผลผลิต

คุณภาพ

ในโปรแกรมสเกลใหญ่ บางแลตเตอร์บล็อกอาจถูกใช้ซ้ำ  
หรือบางแลตเตอร์บล็อกอาจถูกใช้ในโปรแกรมอื่นๆ  
ประสิทธิภาพการทำงานจะไม่เพิ่มขึ้นหากคุณต้องป้อนแลตเตอร์บล็อกเดียวกันซ้ำๆ ทุกครั้งที่ต้องการ  
นอกจากนี้ หากพบข้อบกพร่องในแลตเตอร์บล็อกทั่วไป จะต้องค้นหาและแก้ไขแต่ละแลตเตอร์บล็อก ซึ่งทำให้สิ้นเปลืองเวลาอย่างมาก

ใช้ "บล็อกฟังก์ชัน (FB)" ในการแก้ไขปัญหเหล่านี้  
สามารถจัดการแลตเตอร์บล็อกที่ใช้ซ้ำเป็นบล็อกฟังก์ชัน ซึ่งสามารถใช้ได้ในทุกโปรแกรม ซึ่งจะช่วยให้เพิ่มประสิทธิภาพการทำงานอย่างมาก  
หากพบข้อบกพร่องในบล็อกทั่วไป คุณสามารถแก้ไขได้ง่ายๆ ที่บล็อกฟังก์ชัน  
ไม่เพียงช่วยประหยัดเวลา แต่ยังป้องกันความเสี่ยงที่อาจแก้ไขบล็อกไม่ครบทั้งหมด



กดปุ่มเพื่อดูภาพเคลื่อนไหวตั้งแต่ต้น



## 2.2.1

## การสร้างและวางบล็อกฟังก์ชัน

ผลผลิต

คุณภาพ

เนื่องจากบล็อกฟังก์ชันจะถูกใช้ในโปรแกรมอื่นๆ อุปกรณ์จริง (เช่น X, Y และ D) จึงไม่สามารถใช้ได้กับบล็อกฟังก์ชัน อุปกรณ์นี้ต้องใช้ **ลาเบล** แทนตามที่คุณได้เรียนรู้ในส่วนที่ 2.1

กระบวนการจัดการแลดเดอรบล็อกเป็นบล็อกฟังก์ชันมีแสดงเป็นภาพเคลื่อนไหว

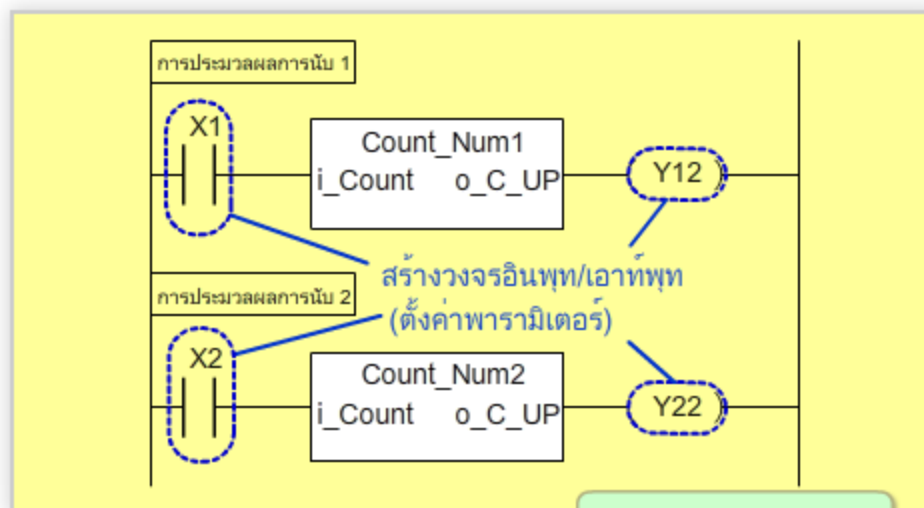
1. เตรียมโปรแกรมที่จะจัดการในบล็อกฟังก์ชัน

2. แบ่งโปรแกรมเป็นอนพุทและเอาพุท และเปลี่ยนอุปกรณ์ภายในโดยลาเบลภายในเพื่อสร้างบล็อกฟังก์ชัน

3. วางบล็อกฟังก์ชันที่สร้างแล้วในโปรแกรมโดยใช้ลากแล้ววาง

4. กำหนดพารามิเตอร์ในลาเบลอนพุทและเอาพุทของบล็อกฟังก์ชันที่คณวางในโปรแกรม

5. สร้างวงจรอนพุทส่งพารามิเตอร์ไปยังลาเบลอนพุท และวงจรเอาพุทที่พารามิเตอร์จากลาเบลเอาพุท ก่อนและหลังบล็อกฟังก์ชัน



กดปุ่มเพื่อย้อนกลับไปยังลำดับแรก

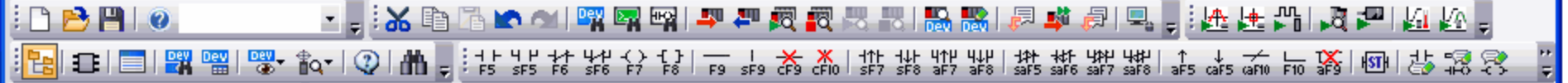


# 2.2.1 การสร้างและวางบล็อกฟังก์ชัน



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN (188)Step \*]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

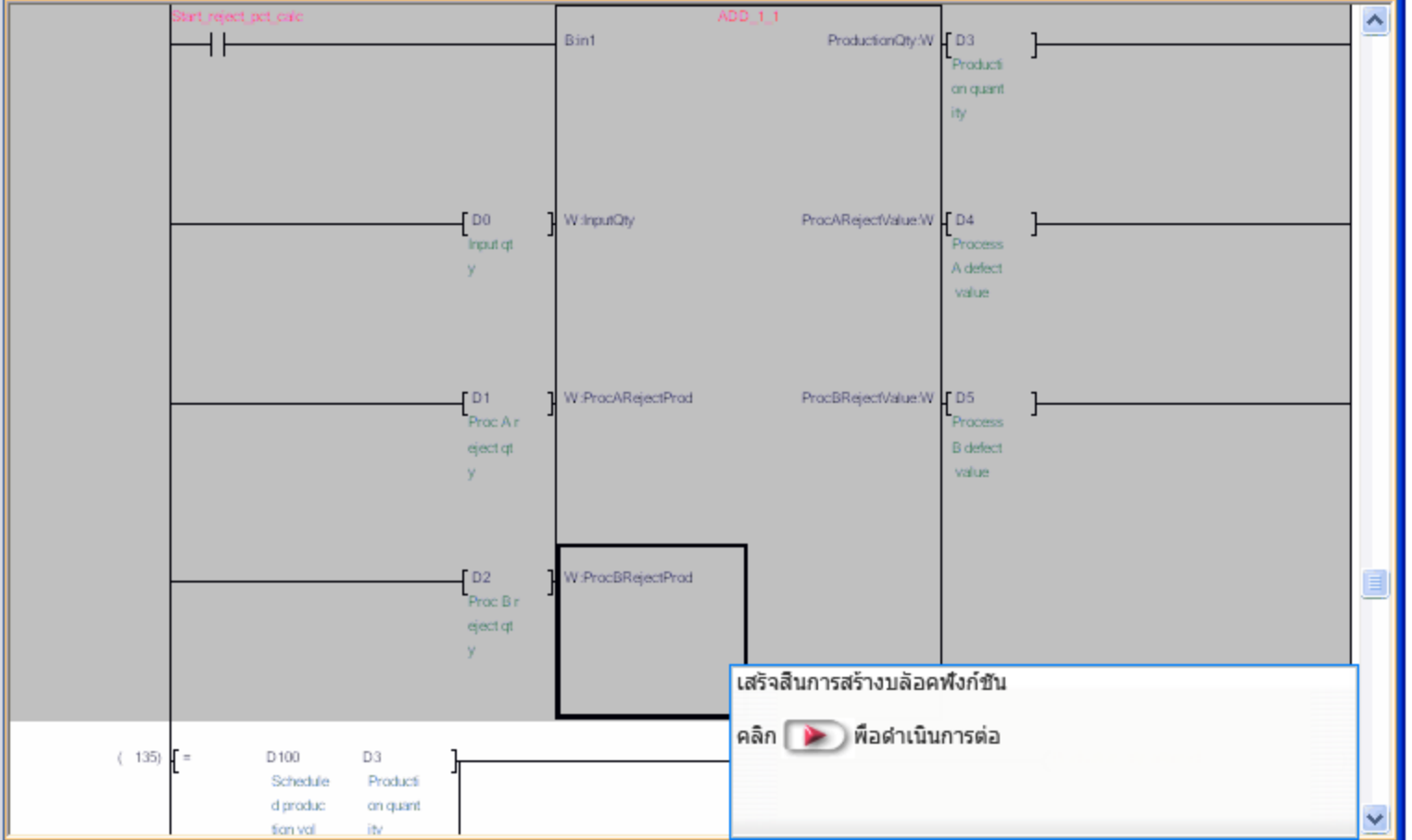


**Navigation**

[PRG]Write MAIN (188)Ste... Global Label Setting Global1 Local Label Setting MAIN [PRG] Function/FB Lat

**Project**

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
  - Global1
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
  - FB\_Pool
    - ADD\_1
      - Program
        - Local Label
  - Structured Data Type



เสร็จสิ้นการสร้างบล็อกฟังก์ชัน

คลิก เพื่อดำเนินการต่อ



## 2.2.2

## การใช้ไลบรารีบล็อกฟังก์ชัน

ผลผลิต

คุณภาพ

ขณะนี้คุณสามารถเรียนรู้วิธีการสร้างบล็อกฟังก์ชัน

Mitsubishi Electric ได้จัดการโปรแกรมควบคุมสำหรับโมดูลต่างๆ เช่น CPUs, โมดูลอินพุท/เอาต์พุตอะนาล็อก เครื่องขยาย และโมดูลการกำหนดตำแหน่ง เป็นบล็อกฟังก์ชัน ซึ่งใช้งานได้ฟรีเรียกว่า **"ไลบรารี FB"**

เมื่อใช้ไลบรารี FB คุณไม่จำเป็นต้องพัฒนาโปรแกรมควบคุมโมดูลซึ่งเคยเป็นความรับผิดชอบของผู้ใช้ในอดีตอีกต่อไป แม้แต่โมดูลที่คุณไม่คุ้นเคยก็สามารถใช้งานได้ง่ายๆ

ไลบรารี FB สามารถดาวน์โหลดได้จากหน้า **"MELSOFT Library Download"** ในเว็บไซต์ **MITSUBISHI ELECTRIC FA**

&lt;ก่อน&gt;

&lt;ถ้าคุณใช้ไลบรารี FB&gt;



\* ไม่มีไลบรารี FB สำหรับ MELSEC-F ซีรีส์  
รุ่นที่สนับสนุนมีแสดงอยู่ในหน้าดาวน์โหลด โปรดตรวจสอบรุ่นที่สนับสนุนก่อนดาวน์โหลด



## 2.3

## การเปลี่ยนค่าเริ่มต้นของอุปกรณ์โดยไม่ต้องแก้ไขโปรแกรม

ผลลัพธ์

คุณภาพ

โดยปกติ ค่าเริ่มต้นหรือค่าคงที่ของอุปกรณ์จะถูกตั้งค่าโดยใช้คำสั่ง MOV ก่อนประมวลผลโปรแกรมหลัก ในกรณีนี้ จะต้องแก้ไขโปรแกรมโดยตรงทุกครั้งที่มีการเปลี่ยนแปลงการทำงานของโปรแกรมตามการใช้งานระบบ วิธีนี้ไม่เพียงสิ้นเปลืองเวลามาก แต่ยังเสี่ยงต่อการแก้ไขที่ล้มเหลวหรือไม่สามารถแก้ไขได้

ใช้ **"ค่าเริ่มต้นอุปกรณ์"** ในการแก้ไขปัญหาล่าช้า

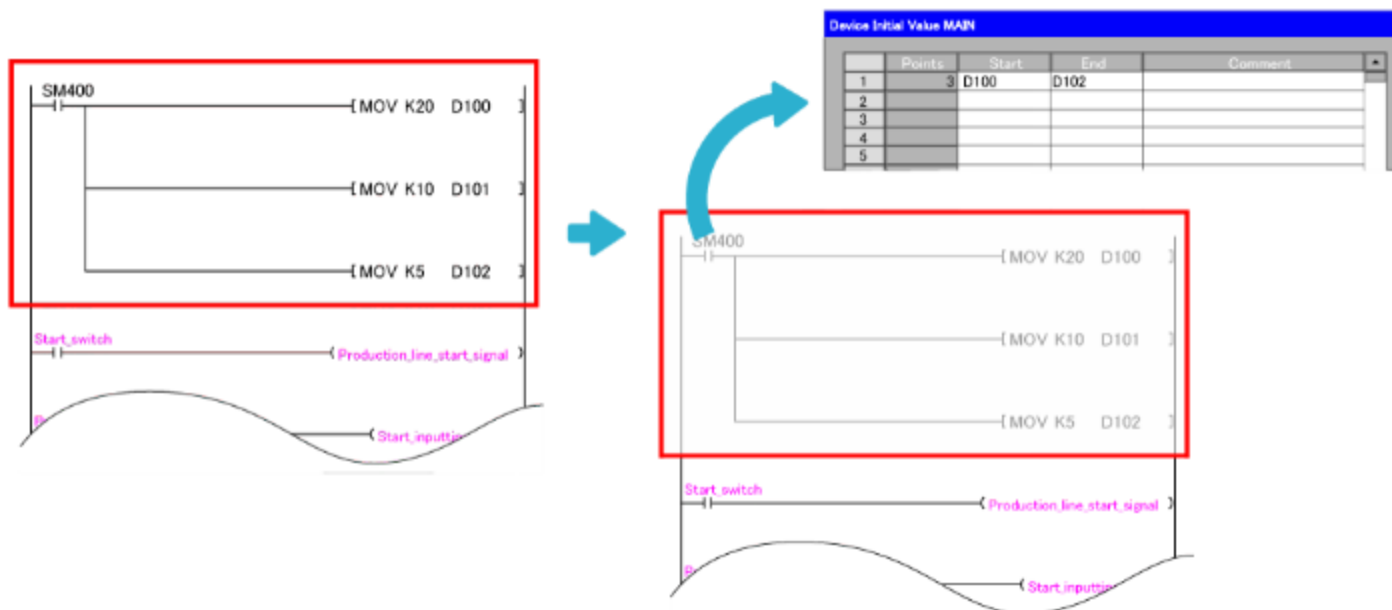
เมื่อใช้ฟังก์ชัน GX Works2 จะช่วยให้อุ่นใจว่ามีการบริหารจัดการค่าเริ่มต้นอุปกรณ์ที่ถูกต้อง และไม่จำเป็นต้องแก้ไขโปรแกรม ให้คุณสร้างโปรแกรมได้อย่างมีประสิทธิภาพมากขึ้น

นอกจากนี้ ฟังก์ชันนี้ยังช่วยให้คุณไม่ต้องใช้โปรแกรมการตั้งค่าเริ่มต้นของอุปกรณ์ ซึ่งจะช่วยลดจำนวนโปรแกรม (การใช้หน่วยความจำ) และลดความเสี่ยงต่อการล้มเหลว

สำหรับค่าเริ่มต้นของอุปกรณ์ ให้ระบุช่วงอุปกรณ์เพื่อตั้งค่าเริ่มต้น

ค่าเริ่มต้นจริงจะถูกบันทึกไว้ใน **หน่วยความจำอุปกรณ์** และกำหนดให้กับช่วงอุปกรณ์ที่ระบุ

ในการเตรียมพื้นที่หน่วยความจำอุปกรณ์ตามการใช้งานระบบ สามารถเปลี่ยนแปลงค่าเริ่มต้นของอุปกรณ์ได้โดยเปลี่ยนพื้นที่หน่วยความจำอุปกรณ์ที่จะกำหนด



# 2.3 การเปลี่ยนค่าเริ่มต้นของอุปกรณ์โดยไม่ต้องแก้ไขโปรแกรม



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN (166)Step \*]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation [PRG]Write MAIN (166)Ste... Global Label Setting Global1 Local Label Setting MAIN [PRG]

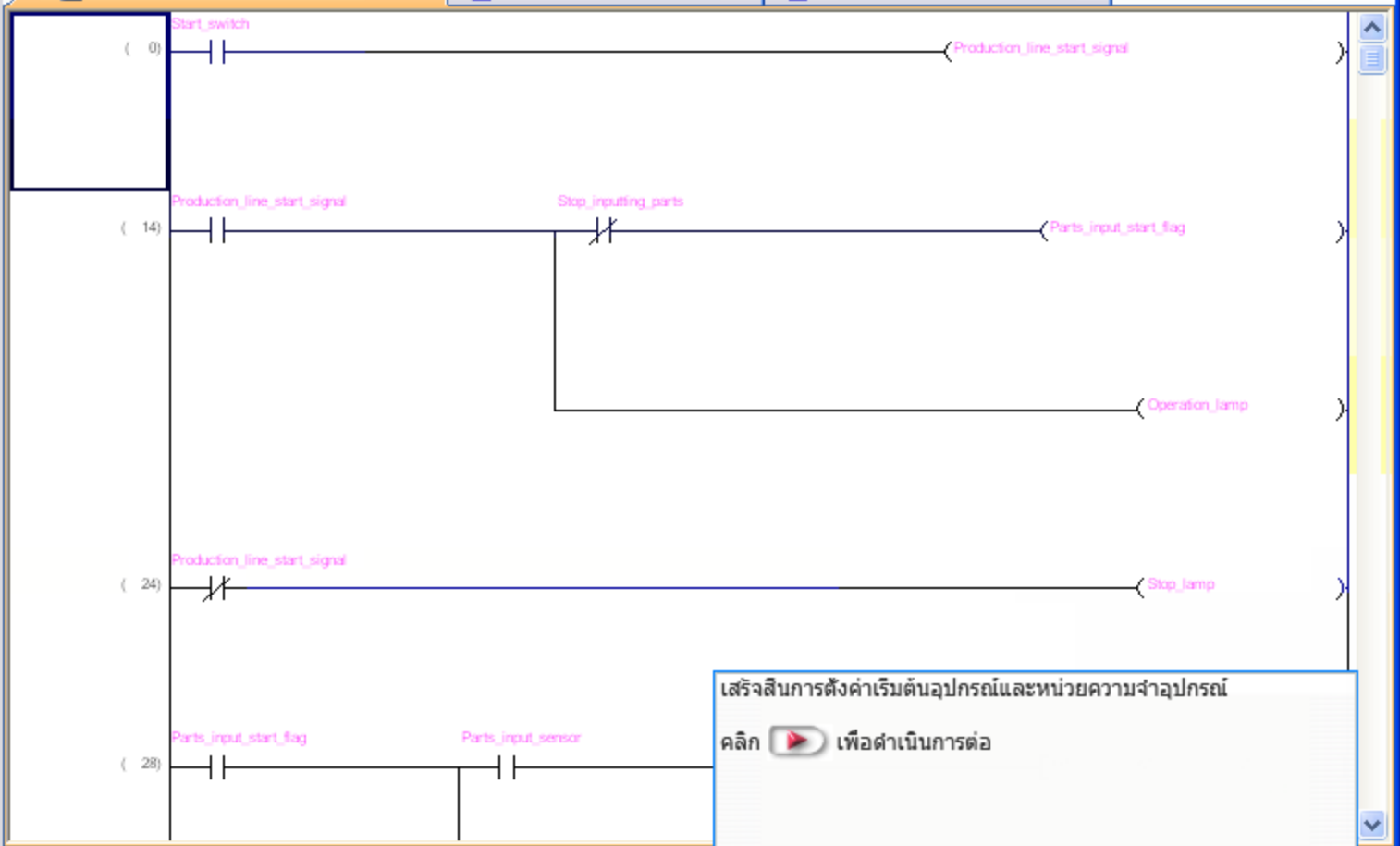
**Project**

- Global1
  - Program Setting
  - POU
    - Program
      - MAIN
        - Program
          - Local Label
  - FB\_Pool
  - Structured Data Type
  - Local Device Comment
- Device Memory
  - MAIN
  - ProdQty1
- Device Initial Value
  - MAIN

**Project**

User Library

Connection Destination



เสร็จสิ้นการตั้งค่าเริ่มต้นอุปกรณ์และหน่วยความจำอุปกรณ์

คลิก เพื่อดำเนินการต่อ

## 2.4

## การปรับโปรแกรมแลตเตอร์ให้ง่ายขึ้น

ผลผลิต

คุณภาพ

ในโปรแกรมแลตเตอร์สเกลใหญ่ การเชื่อมต่ออุปกรณ์ คำสั่ง และบรรทัดกฎ อาจมีความซับซ้อนทำให้ยากที่จะควบคุมการประมวลผลของโปรแกรมได้

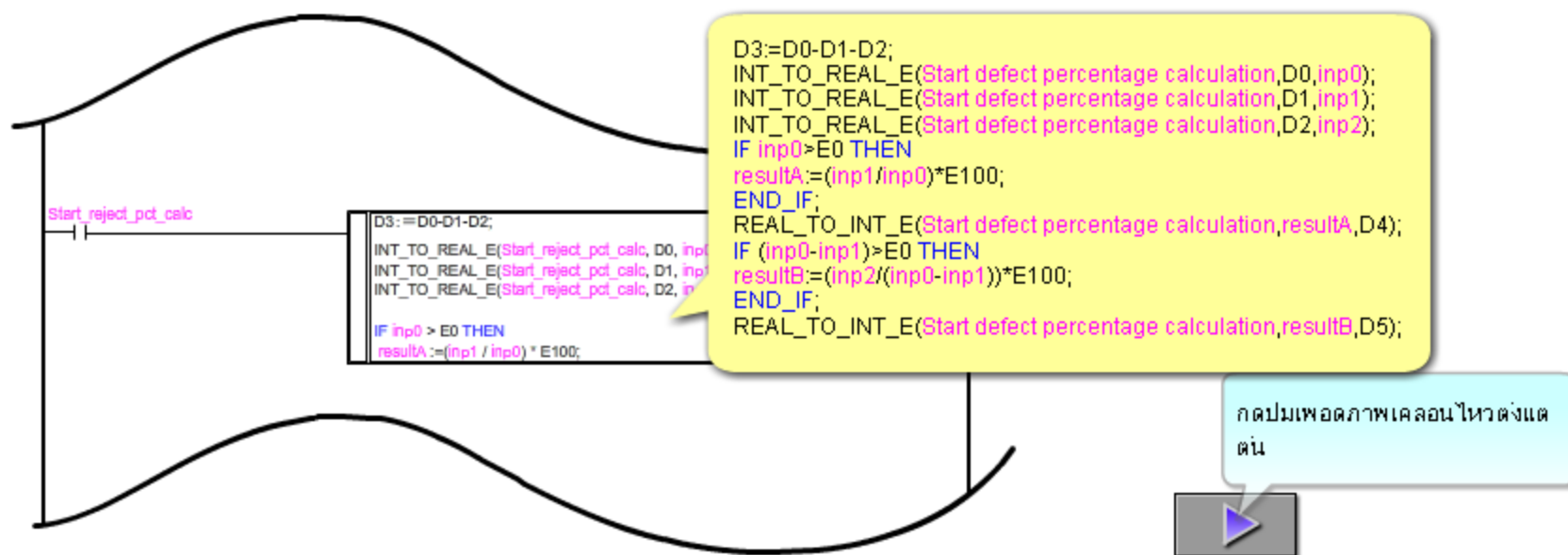
โดยเฉพาะอย่างยิ่ง การคำนวณตัวเลข เช่น สูตรที่จบในบรรทัดเดียว จะต้องตั้งโปรแกรมโดยใช้คำสั่งแบบรวม

ใช้ "ข้อความที่มีโครงสร้างในบรรทัด" ในการแก้ไขปัญหาเหล่านี้

โปรแกรมแลตเตอร์ ใช้แทนโปรแกรมที่ส่วนหนึ่งเขียนขึ้นในภาษา **ข้อความที่มีโครงสร้าง (Structure Text - ST)**

ภาษา ST เป็นภาษาโปรแกรมควบคุมการทำงานตามลำดับ คล้ายกับภาษา C ที่ใช้ในการเขียนโปรแกรมซอฟต์แวร์คอมพิวเตอร์ สามารถเขียนการคำนวณตัวเลขได้โดยใช้สูตรต่างๆ ดังนั้นแม้แต่โปรแกรมเมอร์ที่ไม่คุ้นเคยกับภาษา C ก็สามารถใช้อินไลน์ structured text ได้

ภาพต่อไปนี้แสดงตัวอย่างส่วนของโปรแกรมแลตเตอร์ในระบบที่ถูกแทนที่ด้วย inline structured text คุณจะเห็นว่าโปรแกรมที่มีความซับซ้อนขณะนี้สามารถเข้าใจได้ง่าย



## 2.4

## การปรับโปรแกรมแลตเตอร์ให้ง่ายขึ้น

ผลลัพธ์

คุณภาพ

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 114 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



Navigation

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Global1
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
  - FB\_Pool
  - Structured Data Type
  - Local Device Commen
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

[PRG]Write MAIN 114 Step

Global Label Setting Global1

Local Label Setting MAIN [PRG]



English

Simple

Q03UDE

Host Station

(83 N)

## 2.5

## การสร้างโปรแกรมที่เข้าใจง่ายและอ่านได้ง่าย

ผลผลิต

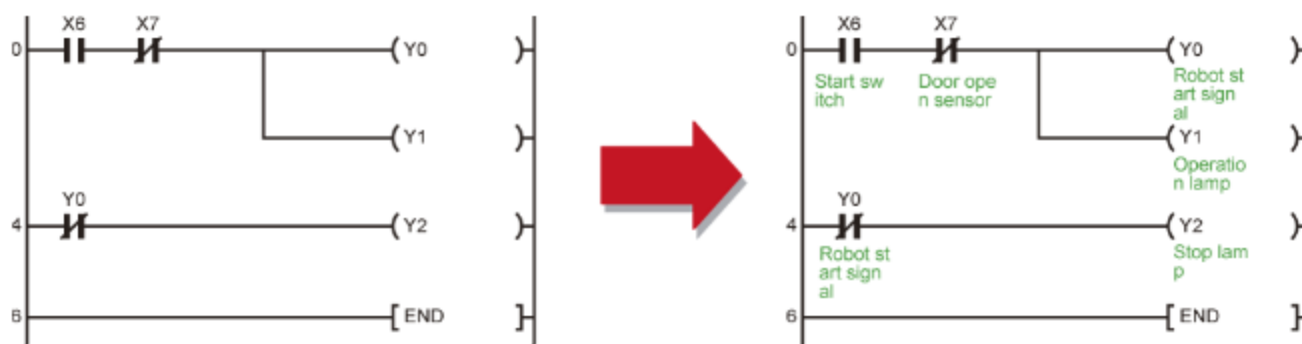
คุณภาพ

คุณอาจพบว่าเป็นเรื่องยากที่จะเข้าใจรายละเอียดการควบคุมในโปรแกรมสเกลใหญ่ จากเพียงแค่มองดูที่โปรแกรม ซึ่งนำไปสู่ปัญหาต่อไปนี้:

- คุณสร้างข้อผิดพลาดในโปรแกรม (เช่น อินพุตคำสั่งหรืออุปกรณ์ที่ไม่ถูกต้อง)
- คุณไม่สามารถค้นหาสาเหตุของข้อผิดพลาดในโปรแกรมได้
- ผู้ที่ต้องเขียนโปรแกรมต่อจากคุณไม่เข้าใจรายละเอียดการควบคุม

ใช้ **"ความเห็น"** ในการแก้ไขปัญหเหล่านี้

สามารถแนบบันทึกต่างๆ เช่น ข้อมูลการควบคุม และชื่ออุปกรณ์ กับโปรแกรมเพื่อให้เข้าใจรายละเอียดการควบคุมได้ง่ายขึ้น และนำไปอธิบายเหตุการณ์ทุกครั้งที่ได้เพื่อสร้างโปรแกรมที่เข้าใจได้ง่ายทั้งสำหรับตนเองและผู้อื่น



GX Works2 ให้คุณป้อนหมายเหตุต่อไปนี้

ชนิดหมายเหตุ	ช่วงหมายเหตุ
หมายเหตุอุปกรณ์	สามารถแนบหมายเหตุกับอุปกรณ์ หมายเหตุนี้ระบุการใช้แต่ละอุปกรณ์และชนิดอุปกรณ์ I/O ที่เชื่อมต่อ
ข้อความสั่ง	สามารถแนบหมายเหตุกับแลดเดอร์บล็อกได้ หมายเหตุนี้ช่วยให้การประมวลผลเข้าใจได้ง่ายขึ้น
บันทึก	สามารถแนบหมายเหตุกับ Coil/คำสั่งแนะนำการใช้งานในโปรแกรม หมายเหตุนี้ช่วยให้เนื้อหาของ Coil (เอาท์พุท) และคำสั่งแนะนำการใช้งานเข้าใจได้ง่าย

# 2.5.1 การเขียนหมายเหตุในแต่ละแลดเดอร์บล็อก

MELSOFT Series GX Works2 C:\Sequential Programs\e-learning\Robot control - [[PRG]Write MAIN 145 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation [PRG]Write MAIN 145 Step

Scheduled prod qty setting

( 0 ) SM400 Always ON N

- [MOV K20 D100 Schedule d produc tion vol ume]
- [MOV K10 D101 Process A defect thresho ld]
- [MOV K5 D102 Process B defect thresho ld]

ข้อความในบรรทัดถูกป้อนที่จุดเริ่มต้นของแลดเดอร์บล็อก

Prod line start

( 7 ) X0 Start sw itch (Y10 Product)

เสร็จสิ้นการตั้งค่าข้อความในบรรทัด  
คลิก เพื่อดำเนินการต่อ

English Simple Q03UDE Host Station (7/ N)

# 2.5.2

## การเขียนหมายเหตุในแต่ละเอาต์พุต (คอยล์ คำแนะนำการใช้งาน)

ผลผลิต

คุณภาพ

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 181 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation [PRG]Write MAIN 181 Step

Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
  - FB\_Pool
  - Structured Data Types
  - Local Device Comment
- Device Memory
- Device Initial Value

Prod line start

( 22 ) X0 Start switch

( 34 ) Y10 Product on line flag

( 34 ) Y16 Stop flag

(Y10 Product on line flag)

(M0 Parts in out start flag)

(Y11)

MOV K10 D101 Process A defect threshold

MOV K5 D102 Process B defect threshold

หมายเหตุถูกป้อนในตำแหน่ง Coil (M0) แล้ว

<Start inputing parts >

M0 Parts in out start flag

เสร็จสิ้นการตั้งค่าหมายเหตุ

คลิก เพื่อดำเนินการต่อ

English Simple Q03UDE Host Station (37 M...

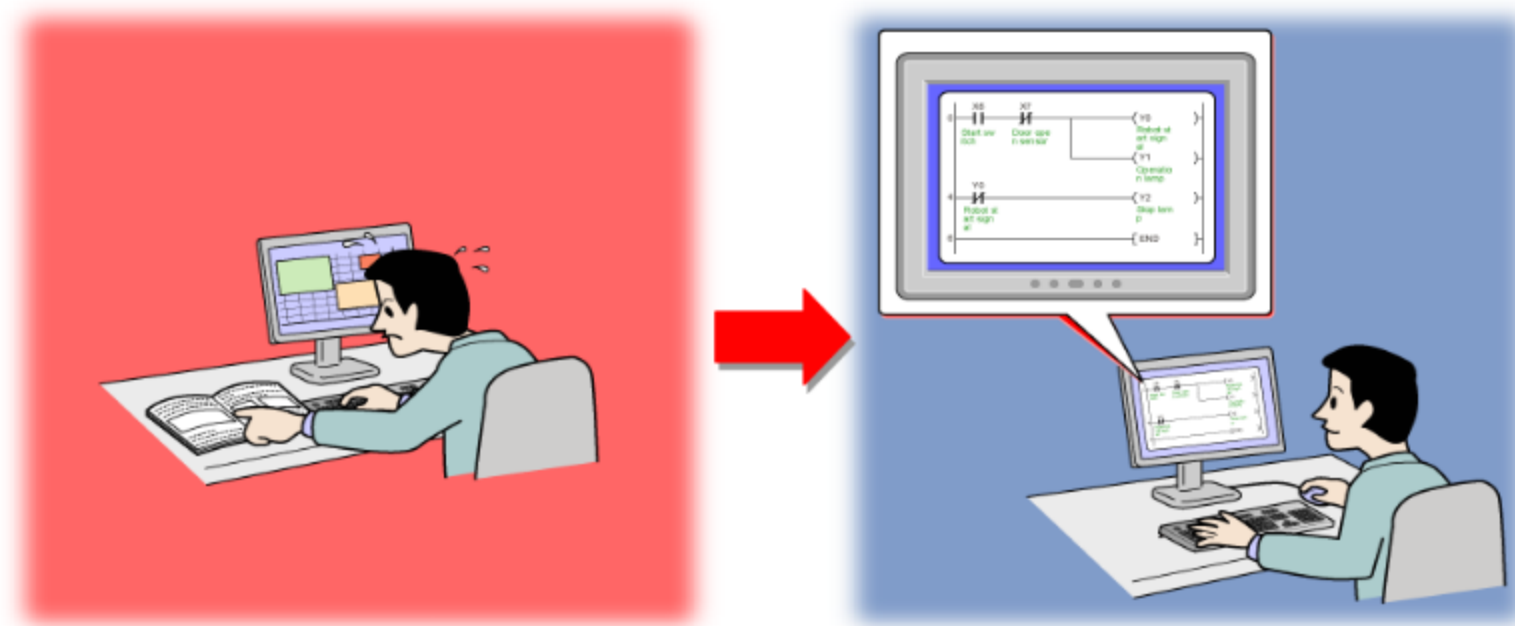
## 2.6

## การปรับให้อ่านโปรแกรมที่มีรีเลย์/รีจิสเตอร์พิเศษได้ง่ายขึ้น

หากมีการใช้รีเลย์พิเศษ รีจิสเตอร์พิเศษ และ/หรืออุปกรณ์โมดูลฟังก์ชันอัจฉริยะในโปรแกรม อาจเป็นเรื่องยากที่จะเข้าใจการใช้งานและฟังก์ชันทั้งหมดของอุปกรณ์นี้ คุณจำเป็นต้องอ่านโปรแกรมพร้อมกับมีคู่มืออยู่ในมือตลอดเวลา ถึงแม้ว่าโปรแกรมจะอ่านได้ง่ายขึ้นเมื่อแนบหมายเหตุกับแต่ละอุปกรณ์ แต่ยังคงจำเป็นต้องใช้เวลาและความพยายามอย่างมากในการแนบหมายเหตุหากมีการใช้อุปกรณ์จำนวนมาก

ใช้ **"ความเห็นตัวอย่าง"** ในการแก้ไขปัญหาเหล่านี้

GX Works2 มีตัวอย่างหมายเหตุที่อธิบายการใช้งานและฟังก์ชันต่างๆ ของรีเลย์พิเศษ รีจิสเตอร์พิเศษ และอุปกรณ์โมดูลฟังก์ชันอัจฉริยะ การใช้ตัวอย่างหมายเหตุเหล่านี้จะช่วยให้อ่านโปรแกรมได้ง่ายขึ้นเพื่อให้อ่านโปรแกรมได้ง่ายขึ้น สามารถแก้ไขตัวอย่างหมายเหตุได้ถ้าต้องการ





## 2.6

## การปรับให้อ่านโปรแกรมที่มีรีเลย์/รีจิสเตอร์พิเศษได้ง่ายขึ้น

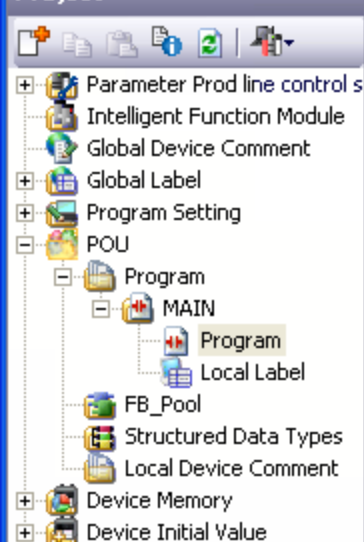
MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



Navigation

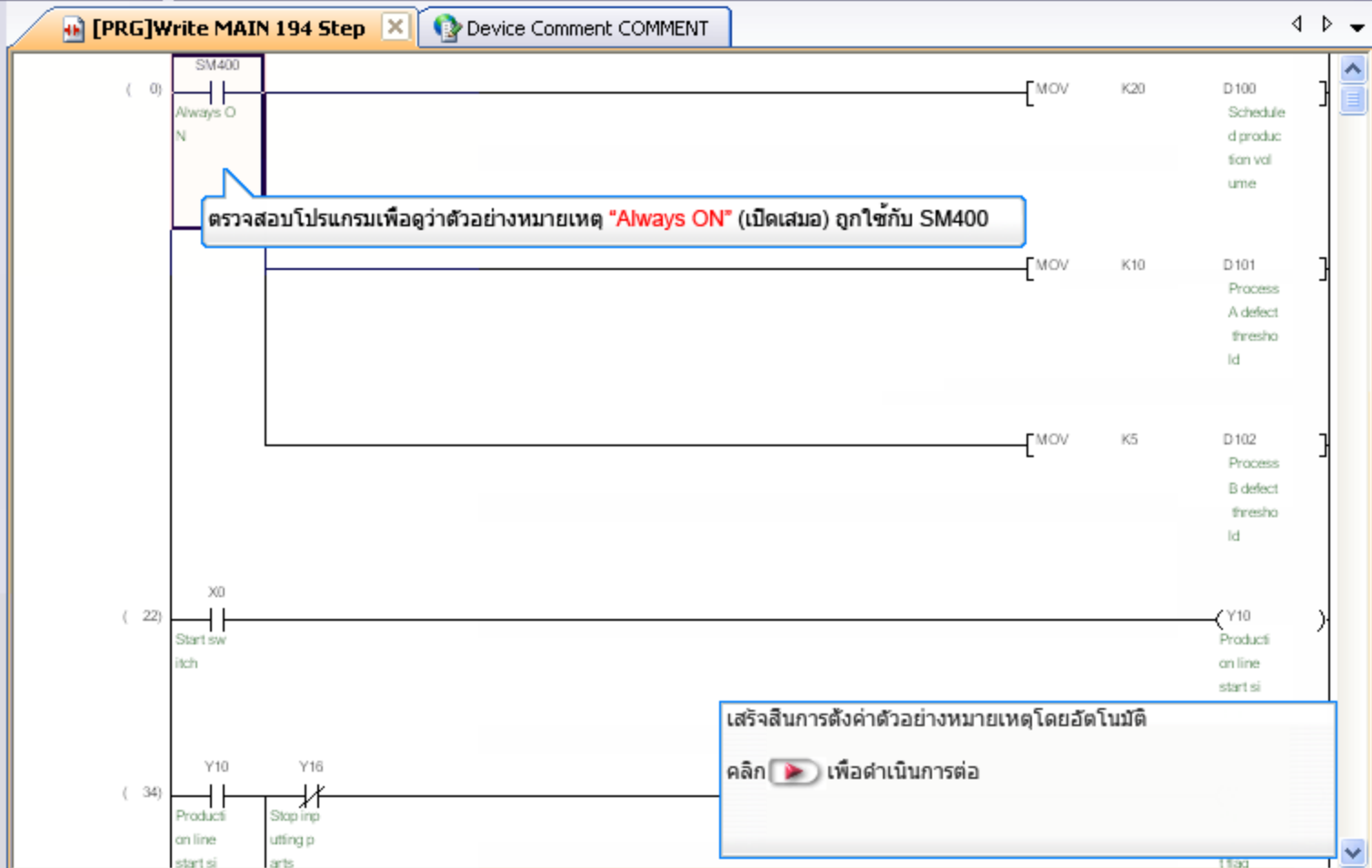
Project



Project

User Library

Connection Destination



English

Simple

Q03UDE

Host Station

(15 N)



## บทที่ 3 การแก้จุดบกพร่อง

### ขั้นตอนการเรียนรู้ในบทที่ 3

ในบทที่ 3 คุณจะเรียนรู้เกี่ยวกับฟังก์ชันต่างๆ ที่ใช้สำหรับการแก้จุดบกพร่อง GX Works2 มีเครื่องมือตรวจสอบและแก้จุดบกพร่องต่างๆ มากมายสำหรับแก้ไขข้อผิดพลาด (จุดบกพร่อง) สร้างโปรแกรมคุณภาพที่ไม่มีความผิดพลาดโดยใช้เครื่องมือตรวจสอบและแก้จุดบกพร่อง

- 3.1 การตรวจสอบเฉพาะอุปกรณ์เป้าหมายและลาเบลเท่านั้น
- 3.2 การตรวจสอบสถานะการใช้อุปกรณ์และลาเบล
- 3.3 การเก็บรวบรวมข้อมูลเกี่ยวกับการเปลี่ยนแปลงค่าอุปกรณ์ตามเวลา
- 3.4 การเปลี่ยนแปลงค่าอุปกรณ์โดยไม่ต้องแก้ไขโปรแกรม
- 3.5 การดำเนินการแก้จุดบกพร่องโปรแกรมที่ละขั้นตอน
  - 3.5.1 การใช้ฟังก์ชัน Step Execution (การดำเนินการขั้นตอน)
- 3.6 การจำลองการทำงานของอุปกรณ์ภายนอก
  - 3.6.1 การป้อนค่าอุปกรณ์สำหรับตั้งค่าระบบ I/O
  - 3.6.2 การใช้รูปแบบแผนภูมิเวลาสำหรับตั้งค่าระบบ I/O

### 3.1

## การตรวจสอบเฉพาะอุปกรณ์เป้าหมายและลาเบลเท่านั้น

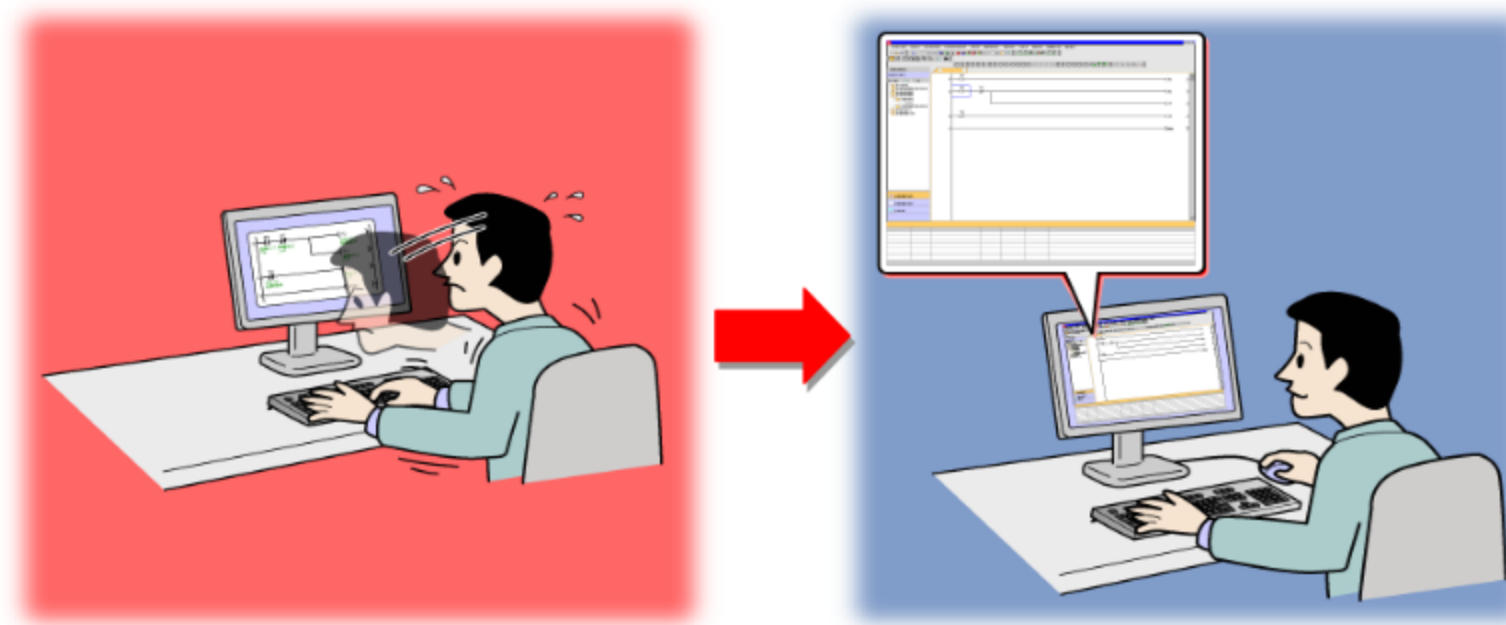
คุณภาพ

โปรแกรมหนึ่งใช้คำสั่งและอุปกรณ์มากมาย

นอกจากนี้ ความยาวในแนวดิ่งของโปรแกรมายังหมายถึงเฉพาะส่วนหนึ่งที่จะแสดงขึ้นเมื่อปรากฏบนหน้าจอคอมพิวเตอร์ PC ดังนั้น การตรวจสอบมอนิเตอร์แลตเตอร์เพียงอย่างเดียวไม่สามารถตรวจสอบการทำงานได้ทั้งหมด

ใช้ "ดู" ในการแก้ไขปัญหาลำนี้

ฟังก์ชันนี้สามารถใช้ตรวจสอบเฉพาะอุปกรณ์และลาเบลที่คุณได้ระบุไว้ล่วงหน้าเท่านั้น สามารถสร้าง Watches ได้ตั้งแต่สองรายการขึ้นไปเพื่อมั่นใจว่าจะมีการตรวจสอบแต่ละช่วง



## 3.1

## การตรวจสอบเฉพาะอุปกรณ์เป้าหมายและลาเบลเท่านั้น

คุณภาพ

MELSOFT Series GX Works2 C:\Sequential Programs\e-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



Navigation [PRG]Monitor Executing M...

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
    - Program

Prod qty calc

( 74) M1

Start re  
ject pct  
calc

D1 D2 D6

GX Simulator 2

Tool Options

Switch

RESET  RUN

LED

MODE

RUN

ERR.

USER

Watch 1 ค่าปัจจุบันของอุปกรณ์ที่ลงทะเบียนกับ Watch 1 จะถูกตรวจสอบ

Device/Label	Current Value	Data Type	Class	Device	Comment
X0	0	Bit		X0	Start switch
X1	0	Bit		X1	Parts input sensor
X2	0	Bit		X2	
X3	0	Bit		X3	

เสร็จสิ้นการลงทะเบียนและการตรวจสอบ Watch

คลิก เพื่อดำเนินการต่อ

Watch 1 Watch 2

English

Simple

Q03UDE

Simulation

(84 N/3



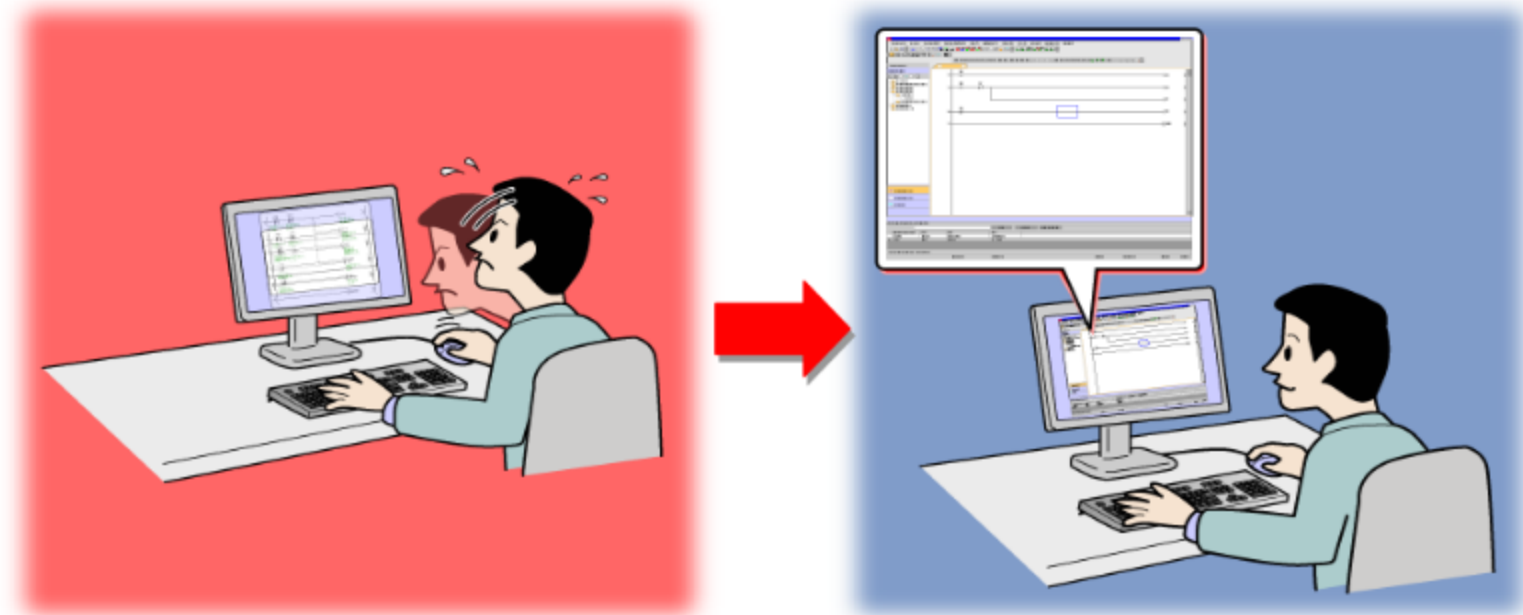
## 3.2

## การตรวจสอบสถานะการใช้อุปกรณ์และลาเบล

คุณภาพ

โปรแกรมหนึ่งใช้อุปกรณ์และลาเบลเดียวกันในหลายตำแหน่ง  
คุณอาจต้องการตรวจสอบสถานะการใช้อุปกรณ์และลาเบลเหล่านี้โดยเปรียบเทียบระหว่างตำแหน่งที่ใช้งาน

ใช้ "ข้อมูลอ้างอิง" ในการทำงานนี้  
ฟังก์ชันนี้แสดงตำแหน่งอุปกรณ์และลาเบลที่ตรงกับเงื่อนไขการค้นหา ซึ่งคุณสามารถเปรียบเทียบและตรวจสอบการใช้งานที่ไม่ถูกต้องได้



## 3.2

## การตรวจสอบสถานะการใช้อุปกรณ์และลาเบล

คุณภาพ

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



Navigation

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
- Program
  - MAIN

Project

Cross Reference

Cross Reference Information Condition Setting

Device/Label D1

Find

Print...

Print Preview...

Device/Label	Device	Instruction	Ladder Symbol	Position	Data Name
Filtering Condition	Filtering Con...	Filtering Condit...		Filtering Condition	
D1	D1	+P	-[ ]-	Step No.63	เสร็จสิ้นการใช้ข้อมูลอ้างอิง
D1	D1	+	-[ ]-	Step No.84	คลิก ▶ เพื่อดำเนินการต่อ
D1	D1	FLT	-[ ]-	Step No.92	

5: device/cross reference information of label "D1"

English

Simple

Q03UDE

Host Station

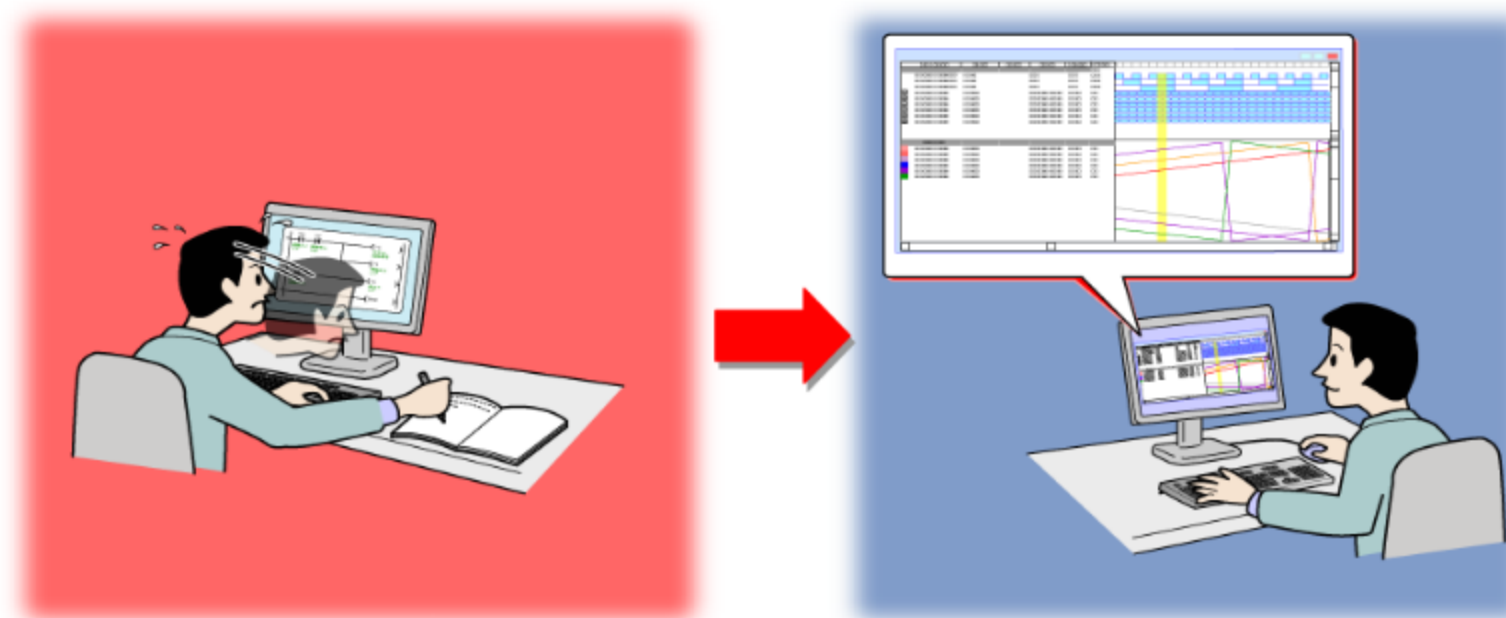
(84 M...



### 3.3 การเก็บรวบรวมข้อมูลเกี่ยวกับการเปลี่ยนแปลงค่าอุปกรณ์ตามเวลา คุณภาพ

คุณอาจต้องการยืนยันว่าการเปลี่ยนแปลงค่าอุปกรณ์และลาเบลอยู่ภายในช่วงการออกแบบ หรือทำการแก้ไขปัญหาโดยตรวจสอบการเปลี่ยนแปลงที่เกิดขึ้นในกรณีที่ล้มเหลว

ใช้ "การติดตามการสุ่มตัวอย่าง" ในกรณีนี้ (เฉพาะสำหรับ MELSEC-Q และ MELSEC-L ซีรีส์)  
ฟังก์ชันนี้สามารถใช้ตรวจสอบและบันทึกการเปลี่ยนแปลงค่าอุปกรณ์และลาเบลตามเวลา สามารถบันทึกกระเบื้องลงไฟล์ได้



# 3.3 การเก็บรวบรวมข้อมูลเกี่ยวกับการเปลี่ยนแปลงค่าอุปกรณ์ตามเวลา

คุณภาพ

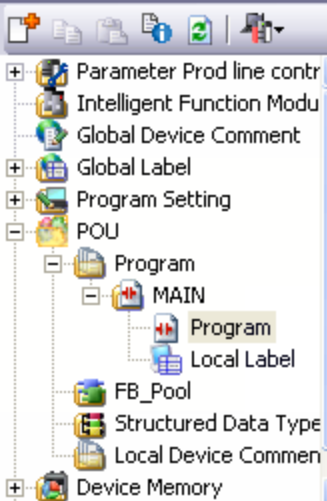
MELSOFT Series GX Works2 C:\Sequential Programs\e-learning\Robot control - [Sampling Trace]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



Navigation

Project



Project

User Library

Connection Destination

[PRG]Monitor Executing MAIN (Rea... Sampling Trace

Device/Label	Device	Comment	Data Type
M0	M0	Parts input st	Bit
M1	M1	Start reject pc	Bit
M2	M2	Production qt	Bit
<input checked="" type="checkbox"/> D3	D3	Production qt	Word(Signed)
<input checked="" type="checkbox"/> D4	D4	Process A de	Word(Signed)
<input checked="" type="checkbox"/> D5	D5	Process B de	Word(Signed)
X0	X0	Start switch	Bit
X1	X1	Parts input se	Bit
X2	X2	Proc A defect	Bit
X3	X3	Proc B defect	Bit

Trend Graph	Device	Comment	Data Type
<input checked="" type="checkbox"/>	D3	Production qt	Word(Signed)
<input checked="" type="checkbox"/>	D4	Process A de	Word(Signed)
<input checked="" type="checkbox"/>	D5	Process B de	Word(Signed)

เสร็จสิ้นการตั้งค่าการติดตามการสุ่มตัวอย่าง

คลิก เพื่อดำเนินการต่อ

English

Simple

Q03UDE

Simulation

NL



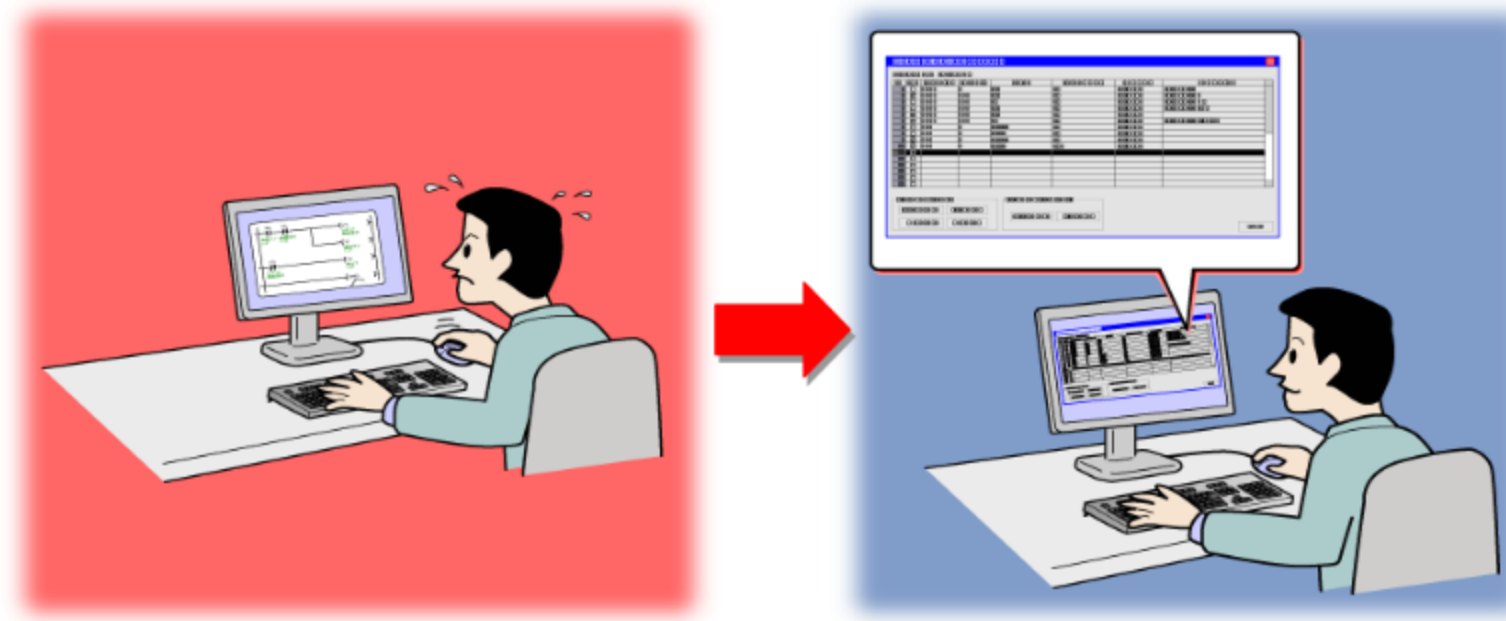
## 3.4

# การเปลี่ยนแปลงค่าอุปกรณ์โดยไม่ต้องแก้ไขโปรแกรม

คุณภาพ

เมื่อทำการแก้จุดบกพร่อง คุณอาจต้องการบังคับเปลี่ยนแปลงค่าอุปกรณ์เพื่อตรวจสอบความแตกต่างในการทำงานของโปรแกรม อย่างไรก็ตาม นั่นหมายความว่า คุณต้องแก้ไขโปรแกรมทุกครั้งที่เปลี่ยนแปลงค่าอุปกรณ์ ซึ่งต้องใช้เวลาและความพยายามอย่างมาก นอกจากนี้ หากคุณลืมตั้งค่าโปรแกรมที่แก้ไขกลับไปเป็นการตั้งค่าเดิม อาจทำให้ล้มเหลวได้

ใช้ "การทดสอบอุปกรณ์ตามเงื่อนไขการดำเนินการ" ในกรณีนี้ (เฉพาะสำหรับ MELSEC-Q และ MELSEC-L ซีรีส์) ฟังก์ชันนี้จะเปลี่ยนแปลงค่าอุปกรณ์เมื่อดำเนินการตามขั้นตอนที่ระบุไว้ล่วงหน้าโดยไม่ต้องแก้ไขโปรแกรม



# 3.4

## การเปลี่ยนแปลงค่าอุปกรณ์โดยไม่ต้องแก้ไขโปรแกรม



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation [PRG]Monitor Executing M... Sampling Trace

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
  - FB\_Pool

GX Simulator 2

Tool Options

Switch

RESET  STOP  RUN

LED

MODE

RUN

ERR.

USER

( 133 ) [ = D100 20 Schedule d produc tion vol ume D3 0 Producti on quant ity ]

( 147 ) [ > M2 Producti on qty a stained ]

[ INT D20 0.000 Process B defect value D5 0 Process B defect value ]

[ D4 0 Process A defect value D101 10 Process A defect thresho ld ]

[ D5 0 Process B defect value D102 5 Process B defect thresho ld ]

[ M2 Producti on qty a stained ]

[ Y13 Producti on qty a stained lamp ]

[ Y14 Producti on qty a stained lamp ]

เสร็จสิ้นการทดสอบอุปกรณ์ตามเงื่อนไขการดำเนินการ  
คลิก ▶ เพื่อดำเนินการต่อ

English Simple Q03UDE Simulation (13 M...

## 3.5

## การดำเนินการแก้จุดบกพร่องโปรแกรมทีละขั้นตอน

คุณภาพ

ในระหว่างการแก้จุดบกพร่อง คุณอาจต้องการยืนยันการดำเนินการตามคำสั่งในแต่ละขั้นตอน หรือตรวจสอบการเปลี่ยนแปลงค่าอุปกรณ์ ใดๆ ก็ดี การแก้จุดบกพร่องทีละขั้นตอนอาจทำได้ยากเนื่องจากความเร็วในการทำงานของโปรแกรม (เวลาการสแกน)

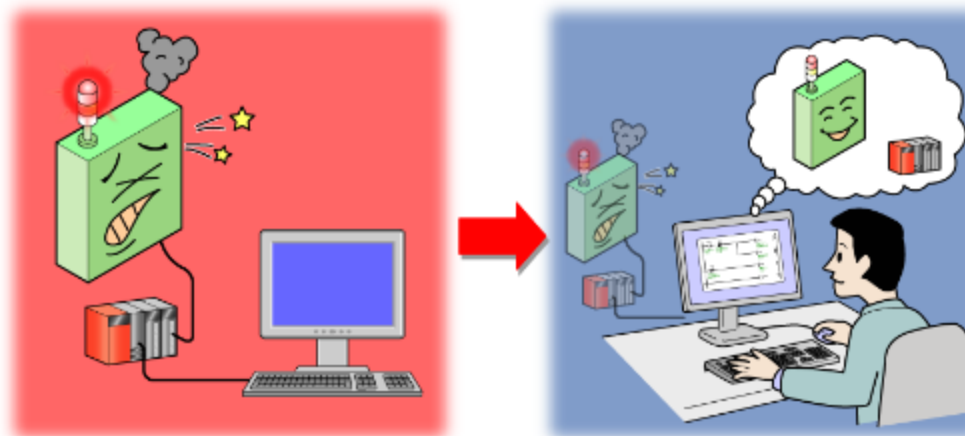
ใช้ "ฟังก์ชันการดำเนินการขั้นตอน" ในการแก้ไขปัญหานี้ (เฉพาะสำหรับ MELSEC-Q และ MELSEC-L ซีรีส์)

ฟังก์ชันนี้เปิดใช้การทำงานของโปรแกรมทีละขั้นตอน และใช้การแก้จุดบกพร่องแบบทีละขั้นตอน

ใช้ฟังก์ชันการดำเนินการขั้นตอนร่วมกับฟังก์ชันการจำลองการทำงาน (ฟังก์ชันการดำเนินการขั้นตอนไม่สามารถใช้ในการแก้จุดบกพร่องบน PLC จริงได้)

ฟังก์ชันต่อไปนี้อาจใช้ได้กับฟังก์ชันการดำเนินการขั้นตอน

ฟังก์ชัน	คำอธิบาย
การหยุดการทำงาน	ฟังก์ชันนี้จะใช้โปรแกรมจนกว่าจะตรงตามเงื่อนไขการหยุดที่ระบุ การทำงานของโปรแกรมจะหยุดเมื่อตรงตามเงื่อนไขการหยุด ใช้จุดพักและอุปกรณ์พักเพื่อระบุเงื่อนไขการหยุด
การดำเนินการขั้นตอน	ฟังก์ชันนี้จะใช้โปรแกรมทีละขั้นตอน
การทำงานบางส่วน	ฟังก์ชันนี้จะใช้โปรแกรมจากตำแหน่งที่ระบุเท่านั้น



## 3.5.1

## การใช้ฟังก์ชัน Step Execution (การดำเนินการขั้นตอน)

คุณภาพ

ระบุ **จุดพัก** และ **อุปกรณ์พัก** เป็นตำแหน่งเริ่มการแก้จุดบกพร่องและเงื่อนไขการเริ่มต้นตามลำดับ นอกจากนี้คุณสามารถระบุ **ช่วงที่ข้าม** ที่คุณต้องการหลีกเลี่ยงการทำงานของโปรแกรมชั่วคราว (**เฉพาะสำหรับ MELSEC-Q และ MELSEC-L ซีรีส์**)

เมื่อตรงตามเงื่อนไขการหยุดหลังจากเริ่ม **ดำเนินการพัก** การทำงานของโปรแกรมจะถูกขัดจังหวะ หลังจากนั้น ขณะที่โปรแกรมทำงานที่ละขั้นตอนด้วยฟังก์ชันการดำเนินการขั้นตอน ให้ตรวจสอบการเปลี่ยนแปลงค่าอุปกรณ์เพื่อค้นหาความผิดปกติ

## &lt;จุดพัก&gt;

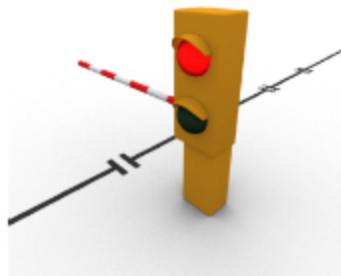
กำหนดจุดพักที่คุณต้องการขัดจังหวะการทำงานของโปรแกรม  
ระบุจุดนี้ในหน่วยขั้นตอน  
สามารถกำหนดจุดพักได้สูงสุด 64 จุดในโครงการทั้งหมด

## &lt;อุปกรณ์พัก&gt;

ตั้งค่าอุปกรณ์พักตามการทำงานของโปรแกรมที่ถูกขัดจังหวะ เมื่อค่าอุปกรณ์หรือลาเบลตรงตามเงื่อนไขที่ระบุ  
สามารถตั้งค่าอุปกรณ์ได้สูงสุด 16 บิต และ/หรือค่า

## &lt;ช่วงที่ข้าม&gt;

กำหนดช่วงที่ไม่ต้องใช้งานโปรแกรม ในหน่วยแลดเดอร์บล็อก ในระหว่างการดำเนินการขั้นตอน สามารถระบุช่วงได้สูงสุด 16 ช่วงในโครงการทั้งหมด



## &lt;การใช้ช่วงที่ข้าม&gt;

จุดที่ล้มเหลวจะลดลงเมื่อใช้ **ช่วงที่ข้าม**  
ดำเนินการหยุดโดยใช่และไม่ใช่การตั้งค่าช่วงที่ข้าม  
ถ้าเกิดความล้มเหลวเฉพาะเมื่อยกเลิกการตั้งค่าช่วงที่ข้าม แสดงว่าช่วงมีความผิดปกติเกิดขึ้น



# 3.5.1 การใช้ฟังก์ชัน Step Execution (การดำเนินการขั้นตอน)

MELSOFT Series GX Works2 C:\Sequential Programs\e-learning\Robot control - [[PRG]Read MAIN (Read Only) 194 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

File Edit View Tools

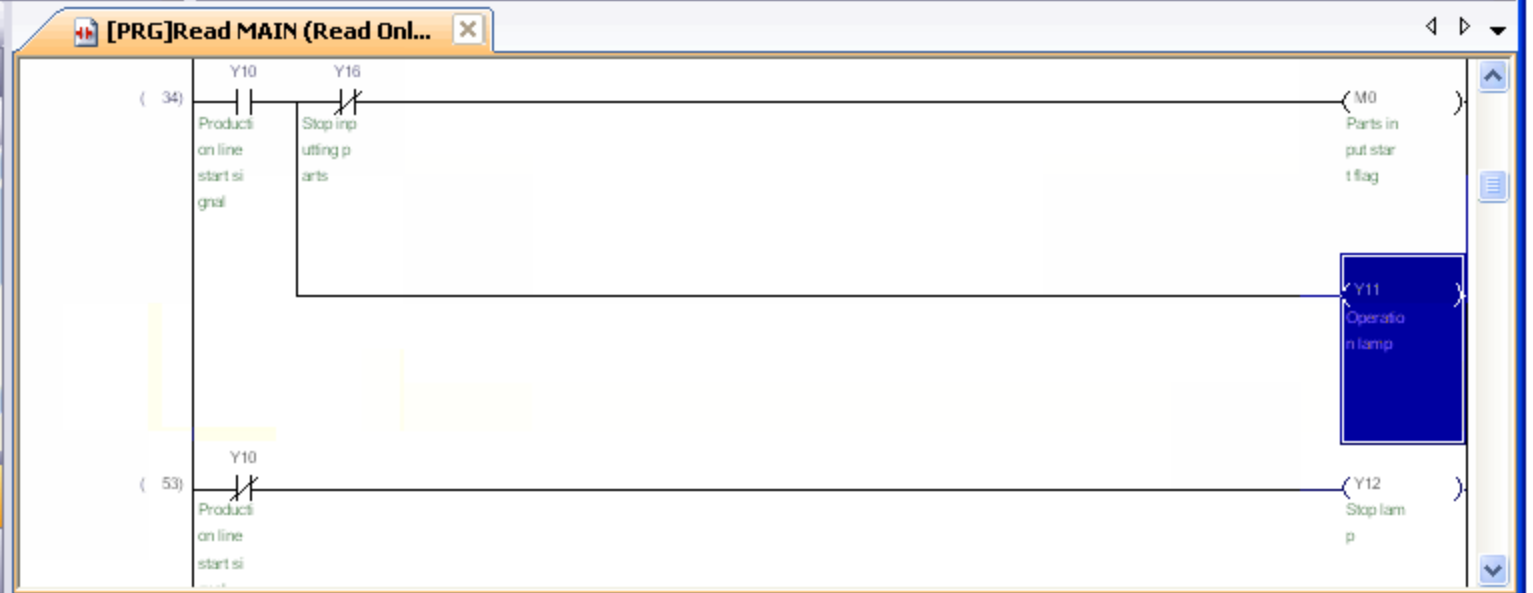
F5 SF5 F6 SF6 F7 F8 F9 SF9 CF9 CF10 SF7 SF8 3F7 3F8 saf5 saf6 saf7 saf8 af5 caF5 caF10 F10 3F5

Navigation

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
- Program
  - MAIN

Project



Break Device

New Condition Cancel Cancel All

Combination

Judge each break device (OR condition)  Judge all break devices (AND condition)

Enable/Disable	Comparative Source (Device/Label)	Condition	Comparative Target (Value/Device/Label)	Comparative Type

การกดจุดควบคุมพร้อมเสร็จสิ้นแล้ว เสร็จสิ้นฟังก์ชันการดำเนินการขั้นตอน  
คลิก ▶ เพื่อดำเนินการต่อ

Skip Range Break Point Break Device

English Simple Q03UDE Host Station (52 N)

## 3.6

## การจำลองการทำงานของอุปกรณ์ภายนอก

คุณภาพ

การแก้จุดบกพร่องร่วมกับการใช้งานอุปกรณ์ภายนอกไม่สามารถทำได้ในสภาพแวดล้อมการพัฒนาโปรแกรม ที่ไม่สามารถใช้งานอุปกรณ์ภายนอก เช่น ฟังก์ชันการจำลองการทำงาน เพื่อแก้ไขปัญหานี้ จึงได้มีการเพิ่มโปรแกรมแก้จุดบกพร่องที่จำลองการทำงานของอุปกรณ์ภายนอกแบบเดิมๆ เข้ามาอย่างไรก็ตาม การสร้างโปรแกรมจำลองการทำงานไม่เพียงใช้เวลาและความพยายามอย่างมาก แต่ยังจำเป็นต้องแก้ไขโปรแกรมเมื่อมีการเปลี่ยนแปลงการทำงานด้วย

ใช้ **"การตั้งค่าระบบ I/O"** ในการแก้ไขปัญหานี้

ฟังก์ชันนี้สามารถจำลองการทำงานของอุปกรณ์ภายนอกโดยไม่ใช้โปรแกรมแก้จุดบกพร่อง การทำงานของอุปกรณ์ภายนอกสามารถตั้งค่าหรือเปลี่ยนแปลงได้ง่ายในหน้าต่างการตั้งค่า ดังนั้น การสร้าง/การแก้ไขโปรแกรมแบบเดิมจึงไม่จำเป็นอีกต่อไป

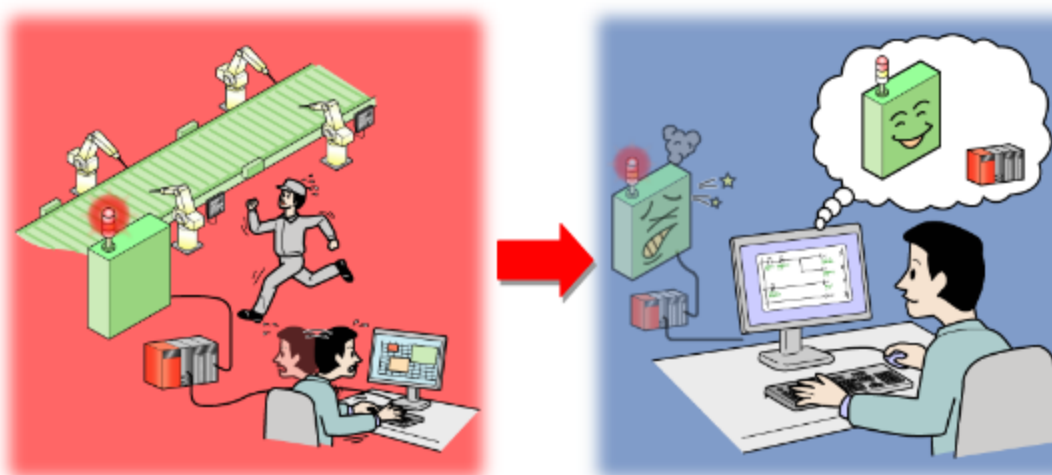
การทำงานของอุปกรณ์ภายนอกสามารถตั้งค่าได้สองวิธีดังนี้:

**<การตั้งค่าอุปกรณ์>**

ค่าอุปกรณ์ที่ระบุสามารถเปลี่ยนแปลงได้เมื่อถึงเวลาที่กำหนดไว้โดยตัวจับเวลาหลังจากตรงตามเงื่อนไขที่ระบุ

**<การตั้งค่าในรูปแบบแผนภูมิเวลา>**

การเปลี่ยนอุปกรณ์ที่ถูกตั้งค่าในรูปแบบแผนภูมิเวลาที่กำหนดสามารถกำหนดได้เมื่อตรงตามเงื่อนไขที่ระบุ



# 3.6.1

## การป้อนค่าอุปกรณ์สำหรับตั้งค้าระบบ I/O



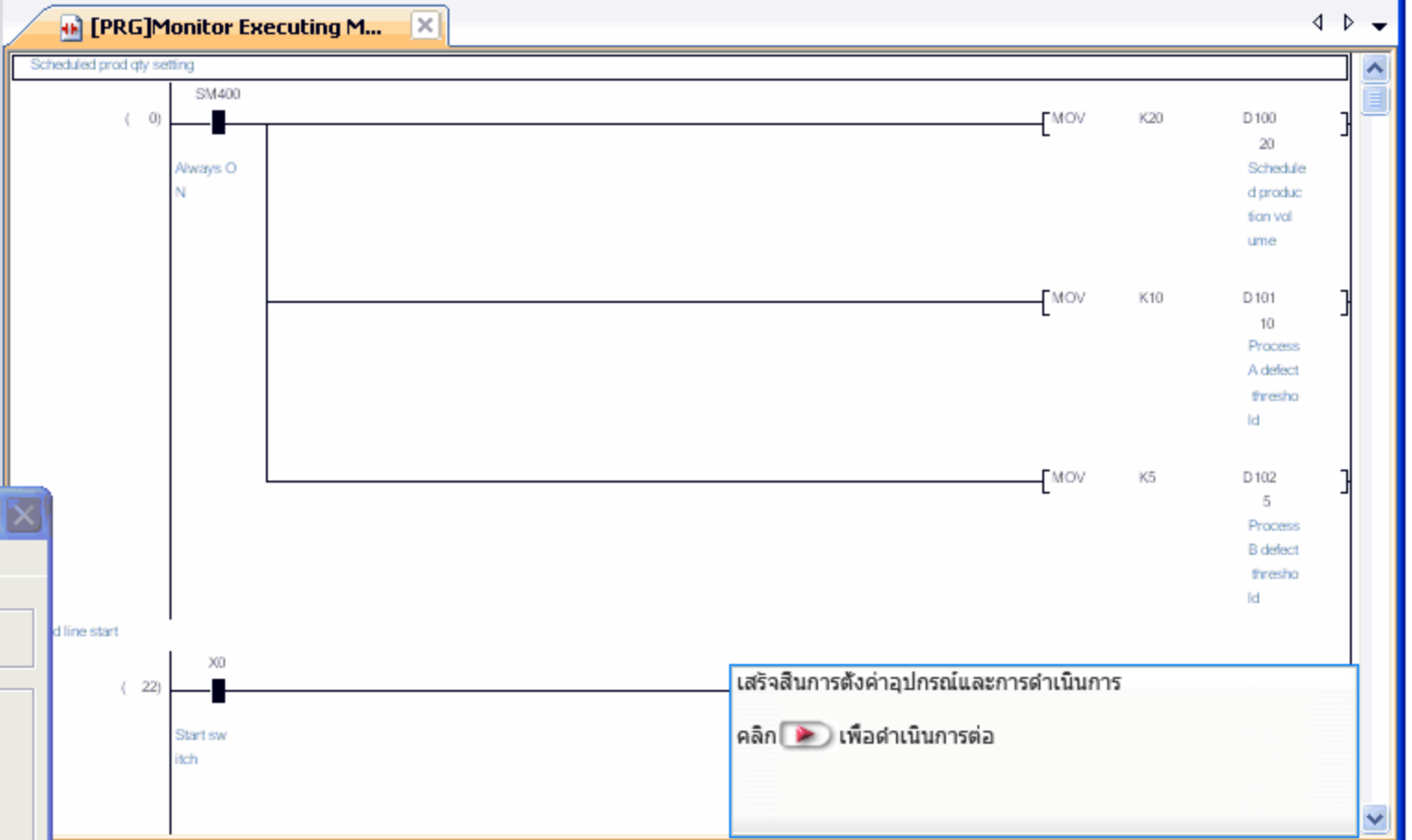
MELSOFT Series GX Works2 C:\Sequential Programs\e-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Toolbar: File operations, navigation, and execution controls.

Navigation Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
    - Program
    - Local Label
- FB\_Pool
- Structured Data Type



GX Simulator 2

Tool Options

Switch:  RESET  STOP  RUN

LED:
 

- MODE
- RUN
- ERR.
- USER

เสร็จสิ้นการตั้งค่าอุปกรณ์และการดำเนินการ  
 คลิก เพื่อดำเนินการต่อ

Simulation controls: Play, Stop, Home, and a slider.

## 3.6.2

## การใช้รูปแบบแผนภูมิเวลาสำหรับตั้งค่าระบบ I/O

คุณภาพ

MELSOFT Series GX Works2 C:\Sequential Programs\e-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step]

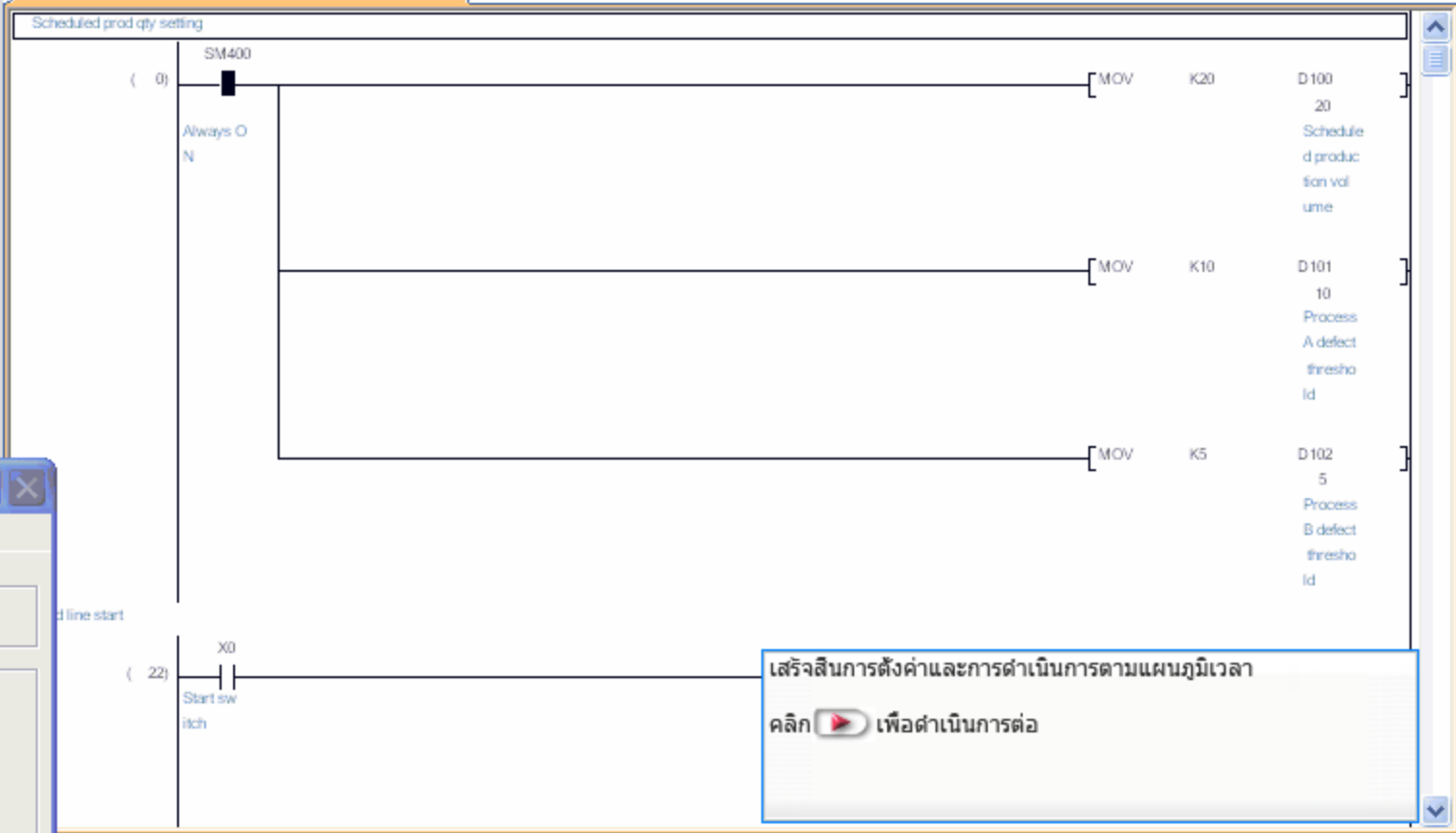
Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



Navigation [PRG]Monitor Executing M...

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool
- Structured Data Type



GX Simulator 2

Tool Options

Switch

RESET  RUN

LED

MODE	<span style="color: green;">■</span>
RUN	<span style="color: green;">■</span>
ERR.	<span style="color: gray;">■</span>
USER	<span style="color: gray;">■</span>

เสร็จสิ้นการตั้งค่าและการดำเนินการตามแผนภูมิเวลา

คลิก เพื่อดำเนินการต่อ

Simulation (0/ N)





## บทที่ 4 การจัดการโครงการและมาตรการด้านความปลอดภัย

### ขั้นตอนการเรียนรู้ในบทที่ 4

ในบทที่ 4 คุณจะเรียนรู้เกี่ยวกับฟังก์ชันที่ใช้สำหรับการจัดการโครงการและมาตรการด้านความปลอดภัย

- 4.1 การป้องกันการรั่วไหลของข้อมูลวิธีการและการแก้ไขโปรแกรมโดยไม่ได้รับอนุญาต
  - 4.1.1 การจำกัดข้อมูลที่เข้าถึงได้ของแต่ละผู้ใช้
- 4.2 การสำรองข้อมูลโครงการและการจัดการเวอร์ชัน
- 4.3 การเปรียบเทียบโปรแกรมที่บันทึกไว้กับตัวควบคุมแบบโปรแกรมและคอมพิวเตอร์ส่วนบุคคล

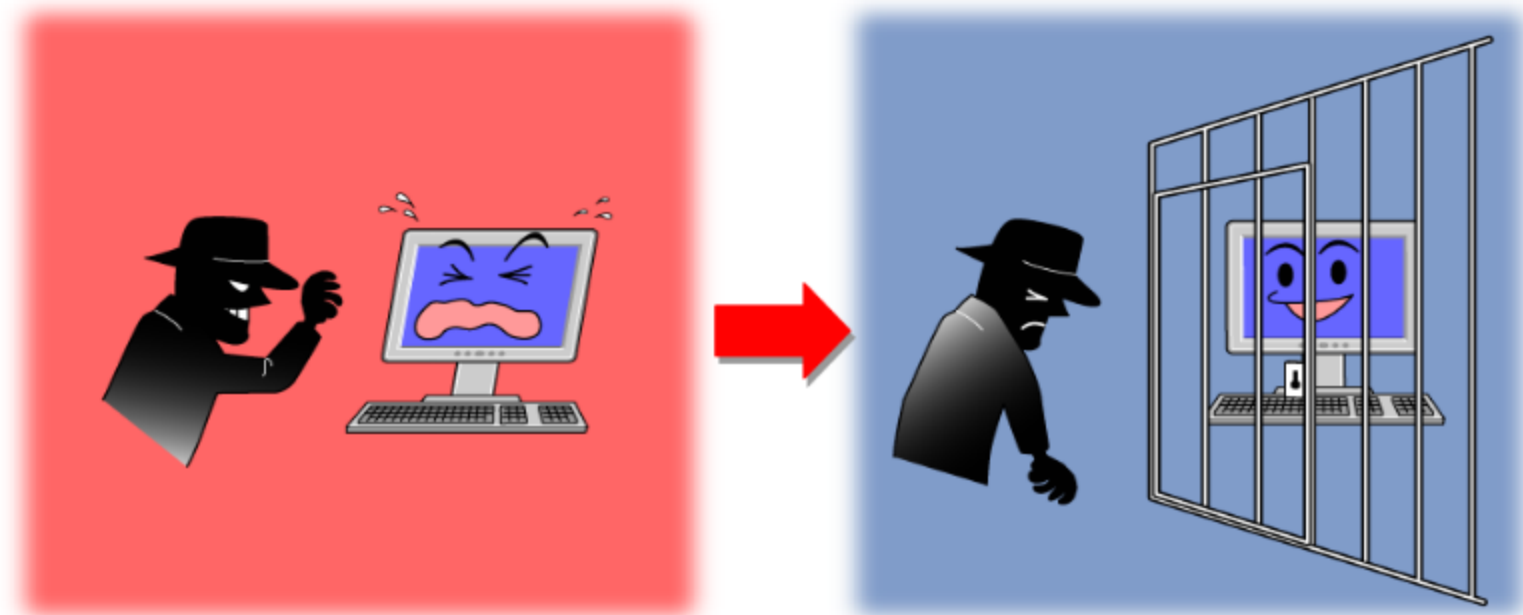
## 4.1

## การป้องกันการรั่วไหลของข้อมูลวิธีการและการแก้ไขโปรแกรมโดยไม่ได้รับอนุญาต



โปรแกรมเชิงลำดับมีทั้งความรู้และข้อมูลเชิงกลยุทธ์ที่สำคัญรวมอยู่ด้วย การรั่วไหลของข้อมูลวิธีการและข้อมูลจากโปรแกรมออกสู่ภายนอกอาจส่งผลกระทบต่อธุรกิจ การแก้ไขโปรแกรมโดยไม่ได้รับอนุญาตอาจนำไปสู่ปัญหาการผลิต เช่น ระบบหยุดทำงาน

ใช้ **"ความปลอดภัย"** ในการแก้ไขปัญหานี้  
ฟังก์ชันนี้จำกัดผู้ใช้ที่สามารถเข้าใช้งานแต่ละโครงการที่มีการป้องกันด้วยรหัสผ่าน นอกจากนี้ยังจำกัดช่วงข้อมูลหรือฟังก์ชันที่แต่ละผู้ใช้สามารถเข้าถึงหรือใช้งานได้ ดังนั้น ฟังก์ชันนี้จึงป้องกันไม่ให้ผู้ใช้ที่ไม่ได้รับอนุญาตสามารถเรียกดูหรือแก้ไขโปรแกรมได้



## 4.1.1

## การจำกัดข้อมูลที่เข้าถึงได้ของแต่ละผู้ใช้



โปรแกรมเชิงลำดับสเกลใหญ่มักพัฒนาโดยโปรแกรมเมอร์ตั้งแต่สองคนขึ้นไปทำงานร่วมกัน ในกรณีของการพัฒนาแบบทีม ช่วงข้อมูลที่เข้าถึงได้และฟังก์ชันที่มีอยู่จะต้องได้รับการจัดการตามช่วงงานและระดับทักษะของแต่ละโปรแกรมเมอร์ และการเก็บข้อมูลที่เป็นความลับที่จัดการโดยแต่ละโปรแกรมเมอร์ การจัดการการเข้าใช้งานสามารถทำได้โดยการตั้งค่า**ระดับการเข้าถึงที่ปลอดภัย**

## &lt;ระดับการเข้าถึง&gt;

สามารถกำหนดสิทธิ์ในการทำงานสำหรับข้อมูลในโครงการให้กับผู้ใช้แต่ละคนได้ สามารถกำหนดระดับการเข้าถึงได้ 5 ระดับต่อไปนี้

ระดับการเข้าถึง		สิทธิ์ในการทำงาน	
↑ สูง	ผู้ดูแลระบบ	ระดับผู้ดูแลระบบ	อนุญาตให้ใช้ทุกฟังก์ชัน
	ผู้พัฒนา (ระดับ 3)	ระดับผู้พัฒนา	จำกัดการตั้งค่าความปลอดภัย การเข้าใช้งานข้อมูล และการทำงานบางอย่าง
	ผู้พัฒนา (ระดับ 2)		
	ผู้พัฒนา (ระดับ 1)		
↓ ต่ำ	ผู้ใช้	ระดับผู้ดำเนินการ	สามารถเรียกดูข้อมูลโครงการได้เท่านั้น ไม่สามารถอ่านจาก PLC CPU ได้

# 4.1.1

## การจำกัดข้อมูลที่เข้าถึงได้ของแต่ละผู้ใช้



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Read MAIN (Read Only) 194 Step]

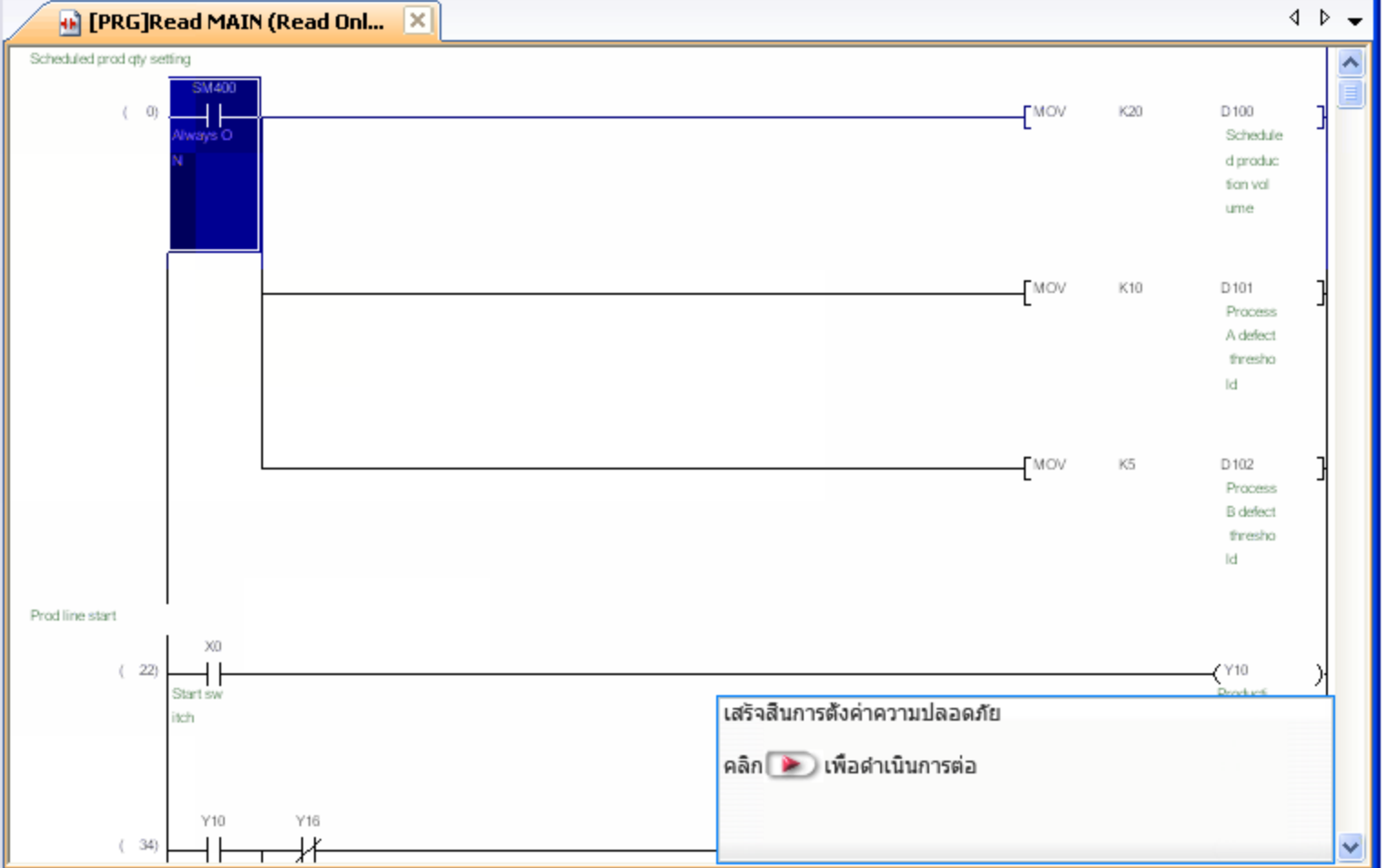
Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation icons: Save, Undo, Redo, Copy, Paste, etc.

Navigation

Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
  - FB\_Pool
  - Structured Data Types
  - Local Device Comment
- Device Memory
- Device Initial Value



เสร็จสิ้นการตั้งค่าความปลอดภัย

คลิก เพื่อดำเนินการต่อ

Navigation icons: Stop, Play, Previous, Next

## 4.2

## การสำรองข้อมูลโครงการและการจัดการเวอร์ชัน

การจัดการ

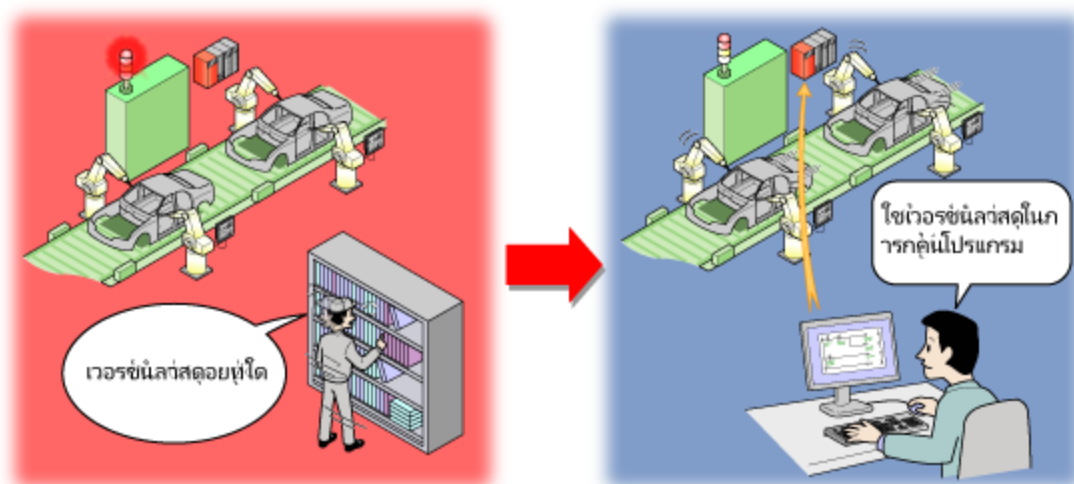
โปรแกรมที่สำคัญอาจสูญหายเนื่องจากความล้มเหลวของ CPU ตัวควบคุมแบบตั้งโปรแกรมได้ หากไม่มีโปรแกรมสำรองข้อมูล อาจมีความหวังเพียงน้อยนิดที่จะกู้คืนข้อมูลได้อย่างรวดเร็ว แม้ว่าจะมีโปรแกรมสำรองข้อมูล แต่อาจเป็นเรื่องยากที่จะทราบได้ว่าเป็นเวอร์ชันเดียวกับโปรแกรมที่สูญหายหรือไม่ เว้นแต่ว่าจะมีการจัดการเวอร์ชันที่ถูกต้อง

ในการเตรียมการสำหรับเหตุการณ์ที่ไม่คาดคิด จำเป็นต้องทำการสำรองข้อมูลตามระยะเวลาและมีการจัดการเวอร์ชันที่เหมาะสม

ใช้ "เปลี่ยนประวัติ" ในการแก้ไขปัญหานี้

ฟังก์ชันนี้สามารถบันทึกได้สูงสุด 100 ชุดประวัติการเปลี่ยนแปลง (หมายเลขประวัติ วันที่/เวลา ผู้ใช้ ชื่อ หมายเหตุ) ของโครงการ ข้อมูลโครงการเมื่อทำการบันทึกจะได้รับการสำรองข้อมูลในขณะเดียวกัน

การจัดการเวอร์ชันโดยใช้ฟังก์ชันประวัติการเปลี่ยนแปลง จะช่วยให้คุณสามารถกู้คืนโปรแกรมที่สูญหาย ตรวจสอบเวอร์ชันของโปรแกรม และสามารถกู้คืนอย่างรวดเร็วในกรณีที่เกิดเหตุการณ์ที่ไม่คาดคิด



# 4.2 การสำรองข้อมูลโครงการและการจัดการเวอร์ชัน



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation [PRG]Write MAIN 194 Step Verify Result [Revision Verify]

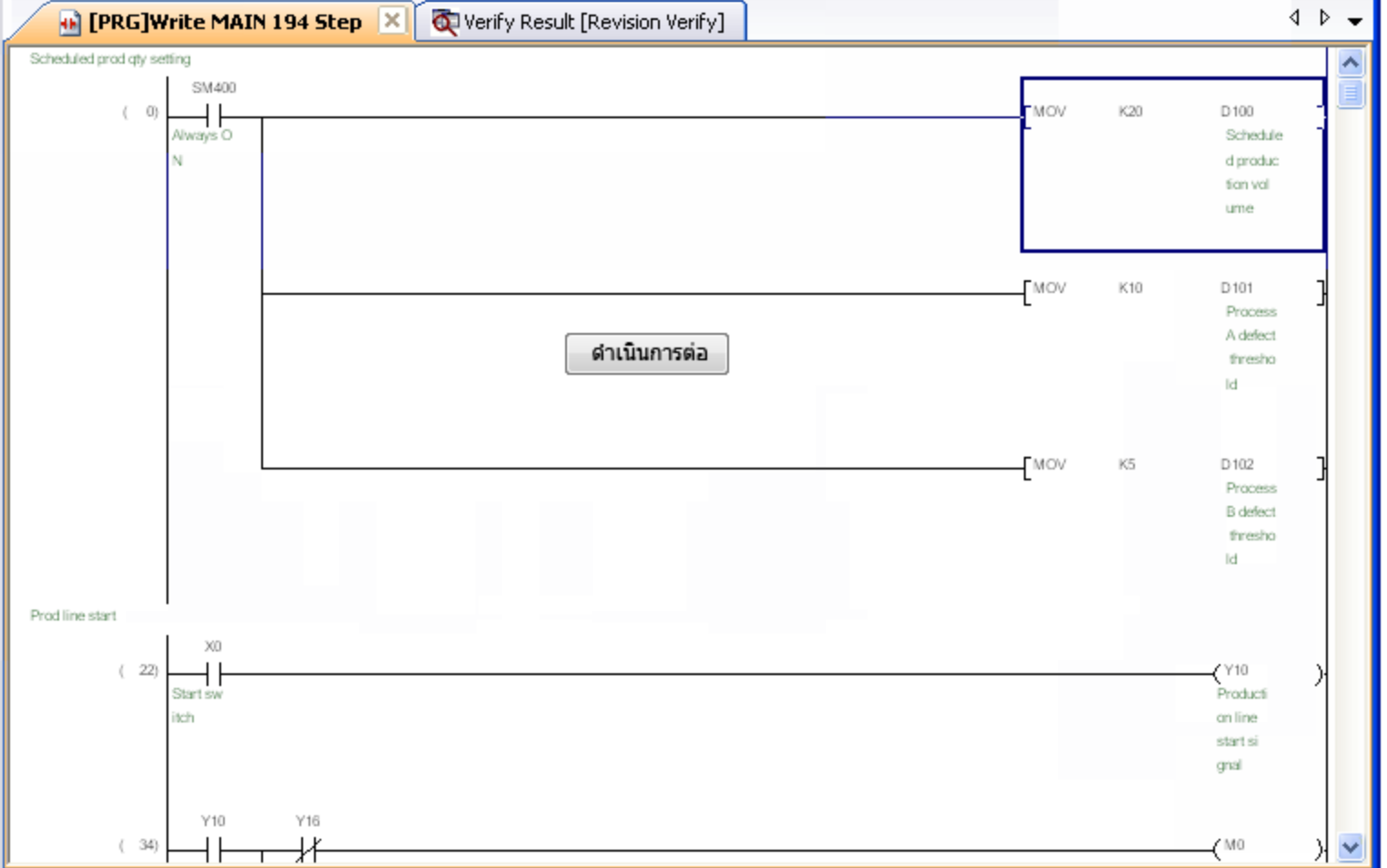
**Project**

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool
- Structured Data Types
- Local Device Comment
- Device Memory
- Device Initial Value

**Project**

User Library

Connection Destination



ดำเนินการต่อ

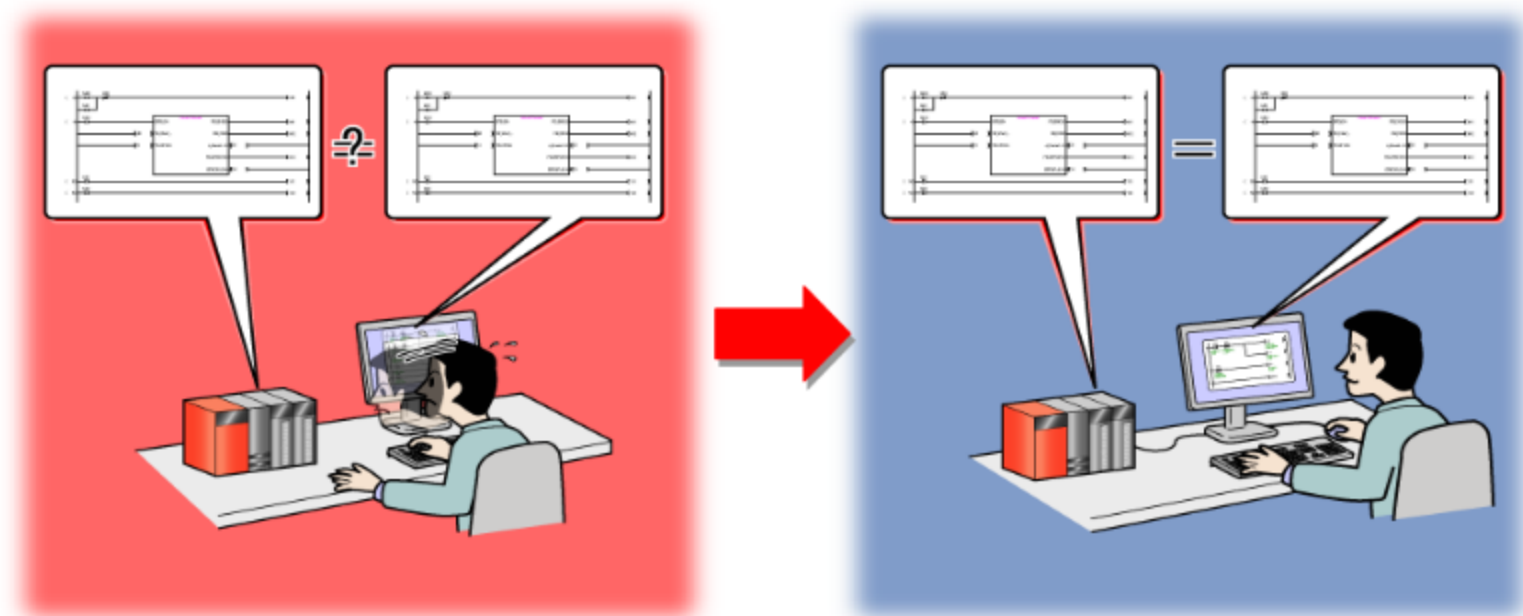
## 4.3

## การเปรียบเทียบโปรแกรมที่บันทึกไว้กับตัวควบคุมแบบโปรแกรมและคอมพิวเตอร์ส่วนบุคคล

การจัดการ

โดยปกติ โปรแกรมจะถูกบันทึกไว้ใน PC ในสภาพแวดล้อมการพัฒนาโปรแกรม และถูกเขียนไว้ใน PLC ด้วย ทั้งสองโปรแกรมนี้อาจไม่เหมือนกันเสมอไป การตรวจสอบด้วยสายตาเพื่อดูว่าเหมือนกันหรือไม่อาจส่งผลให้เกิดข้อผิดพลาด

ใช้ **"ตรวจสอบด้วย PLC"** ในการแก้ไขปัญหานี้  
ฟังก์ชันนี้สามารถตรวจสอบได้ว่าโปรแกรมที่เปิดโดย GX Works2 ตรงกับโปรแกรมที่เขียนไว้ใน PLC



## 4.3

## การเปรียบเทียบโปรแกรมที่บันทึกไว้กับตัวควบคุมแบบโปรแกรมและคอมพิวเตอร์ส่วนบุคคล

การจัดการ

MELSOFT Series GX Works2 C:\Sequential Programs\e-learning\Robot control - [[PRG]Write MAIN 194 Step]

Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help



Navigation [PRG]Write MAIN 194 Step Verify Result [PLC Verify]

Project

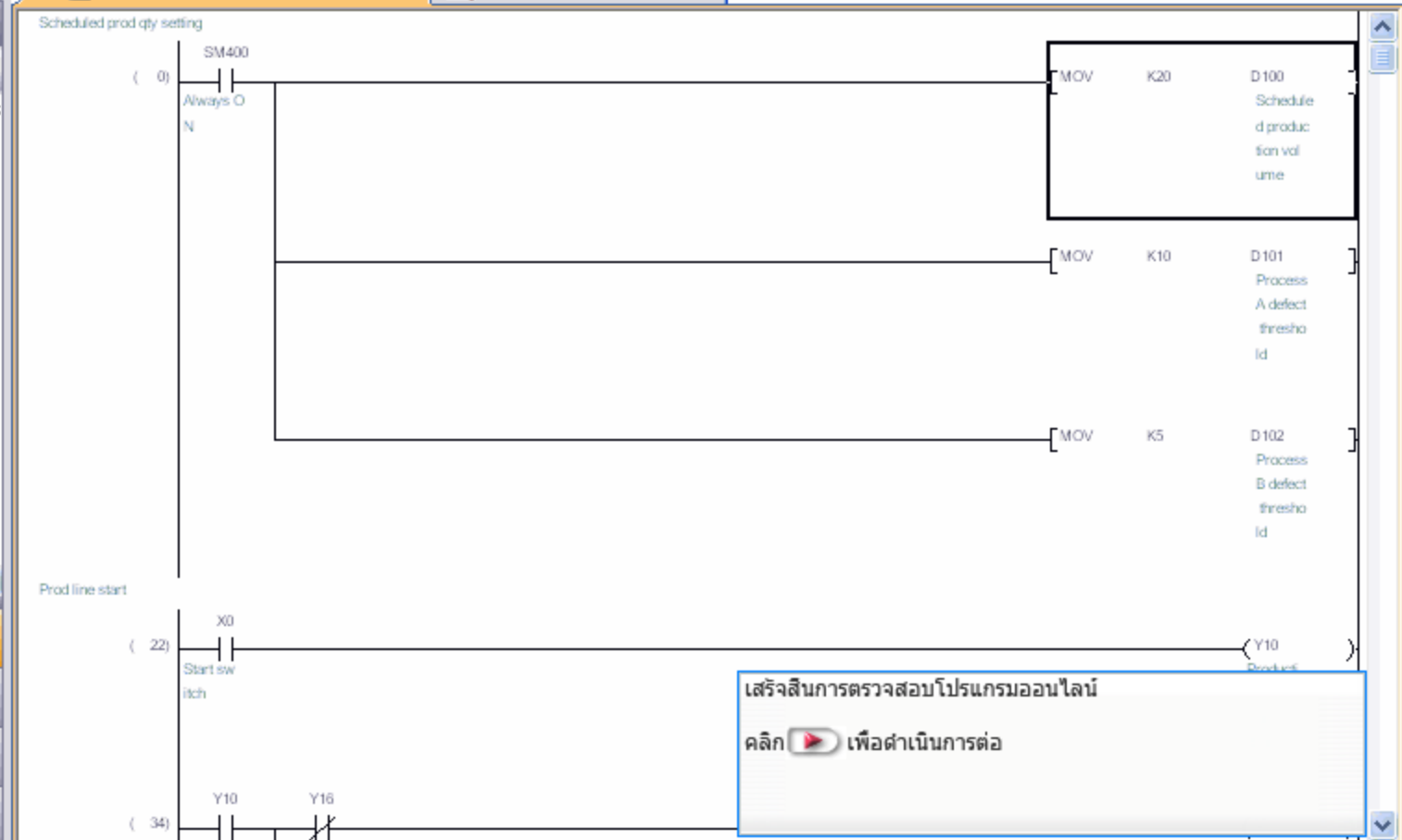
- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
  - Program
    - MAIN
      - Program
        - Local Label
- FB\_Pool
- Structured Data Types
- Local Device Comment

- Device Memory
- Device Initial Value

Project

User Library

Connection Destination



English

Simple

Q03UDE

Host Station

(16 N...





## ทดสอบ แบบทดสอบประเมินผล

ขณะนี้คุณสามารถเรียนรู้ครบทุกบทเรียนในหลักสูตร PLC ขั้นสูง - GX Works2 สร็จสิ้นแล้ว และพร้อมที่จะทำการทดสอบประเมินความรู้ของคุณ หากคุณยังไม่มั่นใจเกี่ยวกับหัวข้อต่างๆ ที่จะทดสอบ โปรดทบทวนหัวข้อเหล่านั้น  
**มีคำถามทั้งสิ้น 8 คำถาม (8 ข้อ) ในการทดสอบประเมินผล**  
 คุณสามารถทำแบบทดสอบประเมินผลได้หลายครั้งตามต้องการ

### วิธีการตอบคำถามในแบบทดสอบ

หลังจากเลือกคำตอบแล้ว ให้คลิกปุ่ม **ตอบ** คำตอบของคุณจะหายไป คำตอบของคุณจะหายไป ถ้าคุณดำเนินการต่อโดยไม่คลิกปุ่ม **ตอบ** (โดยจะถือว่าคำตอบยังไม่ได้ตอบคำถามนั้น)

### ผลคะแนน

จำนวนคำตอบที่ถูกต้อง จำนวนคำถาม เปอร์เซ็นต์คำตอบที่ถูกต้อง และผลลัพธ์ที่แสดงว่าผ่าน/ไม่ผ่านจะปรากฏบนหน้าผลคะแนน

คำตอบที่ถูกต้อง: **2**

จำนวนคำถามทั้งหมด: **9**

เปอร์เซ็นต์: **22%**

คุณต้องตอบคำถามถูกต้องเกินกว่า **60%** จึงจะผ่านการทดสอบ

ดำเนินการต่อ

ทบทวน

ลองใหม่

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

## ทดสอบ แบบทดสอบประเมินผล 1

ฟังก์ชันใดต่อไปนี้จะทำให้การตั้งโปรแกรมมีประสิทธิภาพ  
โดยใช้แลตเดอรับล็อคที่ใช้บ่อยเป็นส่วนประกอบที่ใช้ร่วมกันได้ (เลือกหนึ่งข้อ)

- ข้อความที่มีโครงสร้างในบรรทัด
- ลาเบล
- บล๊อคฟังก์ชัน

ตอบ

ย้อนกลับ

## ทดสอบ แบบทดสอบประเมินผล 2

ฟังก์ชันใดต่อไปนี้อาจสร้างโปรแกรมที่อ่านได้ง่าย  
โดยเปลี่ยนชื่ออุปกรณ์เป็นชื่อที่เกี่ยวข้องกับการใช้งาน (เลือกหนึ่งข้อ)

- หมายเหตุอุปกรณ์
- ลาเบล
- บันทึก

ตอบ

ย้อนกลับ

## ทดสอบ แบบทดสอบประเมินผล 3

ฟังก์ชันใดต่อไปนี้อ่านโปรแกรมที่อ่านได้ง่าย  
โดยให้ข้อมูลเกี่ยวกับการประมวลผลแต่ละแลดเดอร์บล็อก (เลือกหนึ่งข้อ)

- หมายเหตุอุปกรณ์
- บรรทัดข้อความสั่ง
- บันทึก

ตอบ

ย้อนกลับ

## ทดสอบ แบบทดสอบประเมินผล 4

ข้อใดต่อไปนี้เป็นคำอธิบายที่ถูกต้องของฟังก์ชัน "Verify with PLC" (ตรวจสอบด้วย PLC) (เลือกหนึ่งข้อ)

- เปรียบเทียบโปรแกรมที่กำลังแก้ไขกับโปรแกรมที่บันทึกไว้ในประวัติการเปลี่ยนแปลง
- เปรียบเทียบโปรแกรมที่กำลังแก้ไขกับโปรแกรมที่บันทึกไว้ในพีซีที่เลือกไว้
- เปรียบเทียบโปรแกรมที่กำลังแก้ไขกับโปรแกรมที่เขียนไว้ใน PLC CPU

ตอบ

ย้อนกลับ

## ทดสอบ แบบทดสอบประเมินผล 5

ข้อใดต่อไปนี้เป็นคำอธิบายที่ถูกต้องของฟังก์ชัน "การตั้งค่าระบบ I/O" (เลือกหนึ่งข้อ)

- จำลองการทำงานของอุปกรณ์ I/O ภายนอกบนคอมพิวเตอร์ส่วนบุคคลในระหว่างการแก้จุดบกพร่อง
- ควบคุมการทำงานของอุปกรณ์ I/O ภายนอกระยะไกลจากคอมพิวเตอร์ส่วนบุคคลในระหว่างการแก้จุดบกพร่อง
- จำลองการทำงานของ PLC CPU บนคอมพิวเตอร์ส่วนบุคคลในระหว่างการแก้จุดบกพร่อง

ตอบ

ย้อนกลับ

## ทดสอบ แบบทดสอบประเมินผล 6

ข้อใดต่อไปนี้เป็นคำอธิบายที่ถูกต้องของฟังก์ชัน "Change history" (เปลี่ยนแปลงประวัติ) (เลือกหนึ่งข้อ)

- บันทึกการทำงานของ GX Works2 ทีละขั้นตอนเพื่อให้กู้คืนได้ในภายหลัง
- บันทึกข้อมูลประวัติและสำรองข้อมูลโครงการเพื่อเปิดใช้การตรวจสอบและการกู้คืนในภายหลัง

ตอบ

ย้อนกลับ

## ทดสอบ แบบทดสอบประเมินผล 7

ฟังก์ชันใดต่อไปนี้อาจใช้ได้ในการแก้จุดบกพร่องเพื่อเปลี่ยนแปลงเฉพาะค่าอุปกรณ์โดยไม่ต้องแก้ไขโปรแกรม (เลือกหนึ่งข้อ)

- การหยุดการทำงาน
- การตั้งค่าระบบ I/O
- ทดสอบอุปกรณ์ตามเงื่อนไขการทำงาน

ตอบ

ย้อนกลับ



## ทดสอบ แบบทดสอบประเมินผล 8

สมมติว่าโครงการมีสองโปรแกรมคือ A และ B และคุณใช้ฟังก์ชัน "ลาเบล"  
ลาเบลชนิดใดต่อไปนี้สามารถเข้าถึงได้โดยโปรแกรม B (เลือกสองข้อ)

- ลาเบลส่วนกลาง
- ลาเบลในระบบของโปรแกรม A
- ลาเบลในระบบของโปรแกรม B

ตอบ

ย้อนกลับ

## ทดสอบ คะแนนการทดสอบ

คุณทำแบบทดสอบประเมินผลเสร็จเรียบร้อยแล้ว ผลคะแนนของคุณเป็นดังต่อไปนี้  
หากต้องการจบแบบทดสอบประเมินผล ให้ไปยังหน้าถัดไป

คำตอบที่ถูกต้อง: 0

จำนวนคำถามทั้งหมด: 8

เปอร์เซ็นต์: 0%

ดำเนินการต่อ

ทบทวน

ลองใหม่

# คุณไม่ผ่านการทดสอบ

คุณสำเร็จหลักสูตร PLC ขั้นสูง - GX Works2 แล้ว

ขอขอบคุณสำหรับการเรียนรู้หลักสูตรนี้

เราหวังว่าคุณจะเพลิดเพลินกับบทเรียน และข้อมูลที่คุณได้รับจากหลักสูตรนี้จะ  
เป็นประโยชน์ในอนาคต

คุณสามารถทบทวนหลักสูตรได้หลายครั้งตามต้องการ

ทบทวน

ปิด