

&gt;&gt;



PLC

# Khóa học nâng cao về GX Work2

Khóa học này cung cấp kiến thức về các chức năng để cải thiện môi trường phát triển của nơi thiết kế gấp các vấn đề về "năng suất", "chất lượng", "quản lý dự án" và "biện pháp bảo mật". Khóa học dành cho nhà lập trình chương trình PLC đã sử dụng MELSOFT GX Works2.

&gt;&gt;

## Giới thiệu Mục đích khóa học

Khóa học này cung cấp kiến thức về các chức năng để cải thiện môi trường phát triển của nơi thiết kế gắp các vấn đề về "năng suất", "chất lượng", "quản lý dự án" và "biện pháp bảo mật". Khóa học dành cho nhà lập trình chương trình PLC đã sử dụng MELSOFT GX Works2.

## » **Giới thiệu Cấu trúc khóa học**

Nội dung của khóa học này như sau.

Chúng tôi khuyến cáo bạn nên bắt đầu từ Chương 1.

### **Chương 1 - Cải thiện môi trường phát triển bằng GX Works2**

Bạn sẽ tìm hiểu về các vấn đề mà nơi thiết kế gặp phải và môi trường phát triển cần thiết để giải quyết các vấn đề đó.

### **Chương 2 - Lập trình**

Bạn sẽ tìm hiểu về các chức năng được dùng để lập trình.

### **Chương 3 - Sửa lỗi**

Bạn sẽ tìm hiểu về các chức năng được dùng để sửa lỗi.

### **Chương 4 - Quản lý dự án và Biện pháp bảo mật**

Bạn sẽ tìm hiểu về các chức năng dành cho quản lý dự án và biện pháp bảo mật ở giai đoạn phát triển và bảo trì sau khi bắt đầu vận hành hệ thống.

### **Bài kiểm tra cuối khóa**

Điểm đạt: 60% trở lên

## Giới thiệu | Làm thế nào sử dụng Công cụ e-Learning

Đến trang tiếp theo		Đến trang tiếp theo.
Trở lại trang trước		Trở lại trang trước.
Di chuyển đến trang mong muốn		"Mục lục" sẽ được hiển thị, cho phép bạn điều hướng đến trang mong muốn.
Thoát khỏi bài học		Thoát khỏi bài học. Cửa sổ chẳng hạn như màn hình "Nội dung" và bài học sẽ được đóng lại.

&gt;&gt;

## Giới thiệu Thận trọng khi sử dụng

### Biện pháp phòng ngừa an toàn

Khi bạn học tập bằng cách sử dụng các sản phẩm thực tế, hãy đọc kỹ các biện pháp phòng ngừa an toàn trong hướng dẫn sử dụng tương ứng.

### Biện pháp phòng ngừa trong khóa học này

- Màn hình hiển thị của phiên bản phần mềm mà bạn sử dụng có thể khác với các màn hình trong khóa học này.

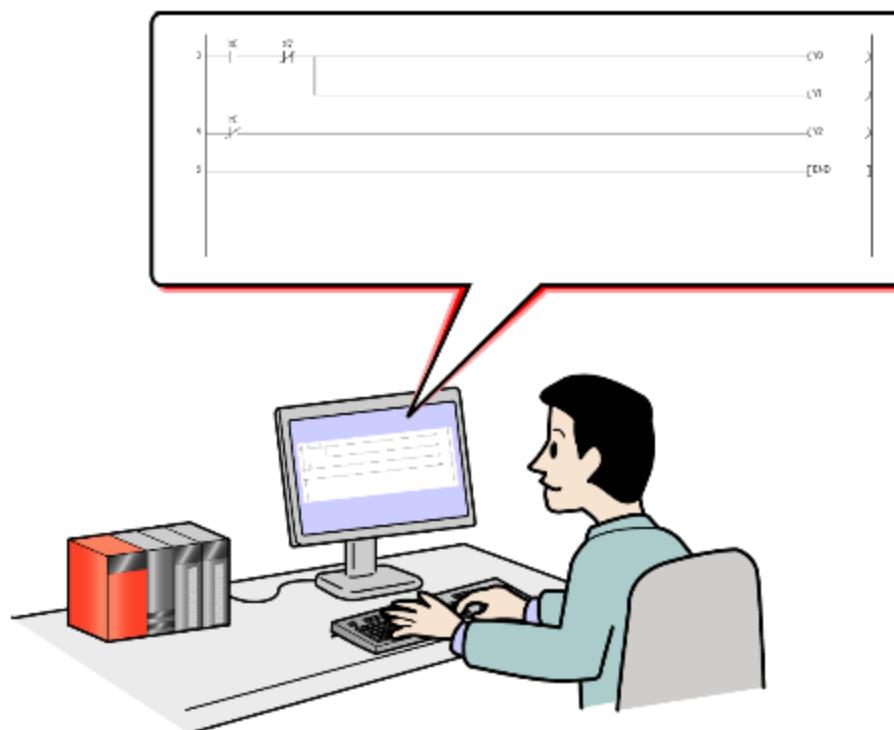
## Chương 1 Cải thiện môi trường phát triển bằng GX Works2

### Các bước đào tạo trong Chương 1

Khóa học này dành cho nhà lập trình phát triển các chương trình PLC. Bạn sẽ tìm hiểu cách sử dụng các chức năng hữu ích trong **MELSOFT GX Works2** để giải quyết các vấn đề tại nơi thiết kế liên quan đến "**năng suất**," "**chất lượng**," "**quản lý dự án**" và "**bảo mật**."

Trong Chương 1, bạn sẽ tìm hiểu về các vấn đề mà nơi thiết kế gặp phải và môi trường phát triển cần thiết để giải quyết các vấn đề đó.

- 1.1 Các vấn đề nơi thiết kế gặp phải
  - 1.1.1 Cải thiện năng suất
  - 1.1.2 Cải thiện chất lượng
  - 1.1.3 Quản lý dự án
  - 1.1.4 Biện pháp bảo mật
- 1.2 Quy trình học



## 1.1

## Các vấn đề nơi thiết kế gấp phải

Mặc dù nơi thiết kế được yêu cầu cải thiện năng suất để giảm chi phí phát triển nhưng cũng cần đảm bảo rằng các chương trình được phát triển có chất lượng cao.

Đồng thời, nơi thiết kế phải thực hiện quản lý dự án cho phép phát triển nhóm và đảm bảo khôi phục nhanh trong trường hợp sự cố.

Các biện pháp bảo mật cũng là yêu cầu bắt buộc vì các chương trình PLC có chứa nhiều kiến thức và dữ liệu quan trọng.



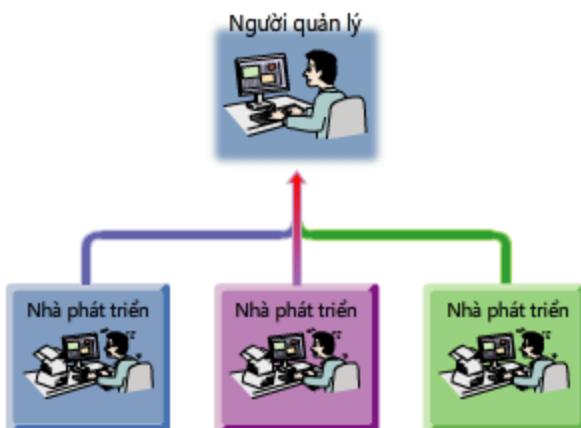
Năng suất

Chất lượng



Quản lý

Bảo mật



### 1.1.1

## Cải thiện năng suất

Nơi thiết kế cần phải phát triển các chương trình PLC để thực hiện xử lý trên quy mô lớn, phức tạp.

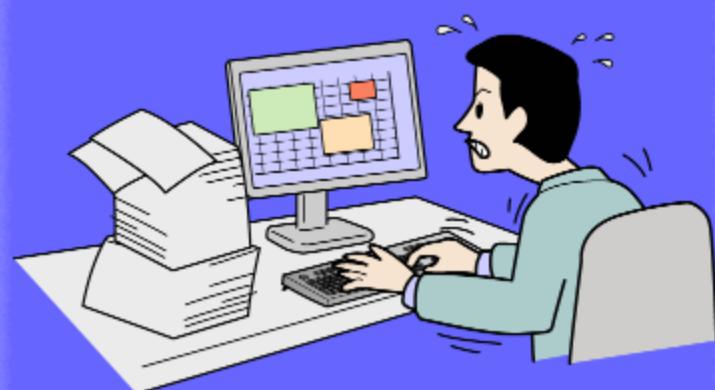
Tuy nhiên, chi phí phát triển sẽ tăng lên cùng với độ dài của thời gian phát triển.

Do đó, bạn cần phải cải thiện năng suất phát triển chương trình để giảm chi phí phát triển.

GX Works2 cung cấp các chức năng sau để giải quyết vấn đề này:

- Nhãn
- Khối chức năng
- Giá trị ban đầu của thiết bị và bộ nhớ thiết bị\*
- Văn bản nội tuyến có cấu trúc
- Nhập vào từ chủ thích mẫu

\* Sê-ri MELSEC-F không thể cấu hình các giá trị ban c



**1.1.2****Cải thiện chất lượng**

Một lỗi trong chương trình PLC cũng có thể dẫn đến dừng hệ thống, lỗi hoặc tai nạn, gây ngưng trệ sản xuất, tổn thất lợi nhuận và độ tin cậy.

Bạn cần phải phát triển các chương trình có chất lượng và hoàn toàn không có lỗi.

GX Works2 cung cấp các chức năng sau để giải quyết vấn đề này:

- Nhãn
- Khối chức năng
- Giá trị ban đầu của thiết bị và bộ nhớ thiết bị\*
- Chú thích
- Nhập vào từ chú thích mẫu
- Theo dõi
- Tham chiếu chéo
- Theo vết lấy mẫu\*
- Kiểm tra thiết bị theo điều kiện thực hiện\*
- Chức năng thực hiện từng bước\*
- Thiết lập hệ thống I/O



\* Sê-ri MELSEC-F không thể sử dụng thiết lập giá trị ban đầu của thiết bị, theo vết lấy mẫu, kiểm tra thiết bị theo điều kiện thực hành và chức năng thực hiện từng bước.

### 1.1.3 Quản lý dự án

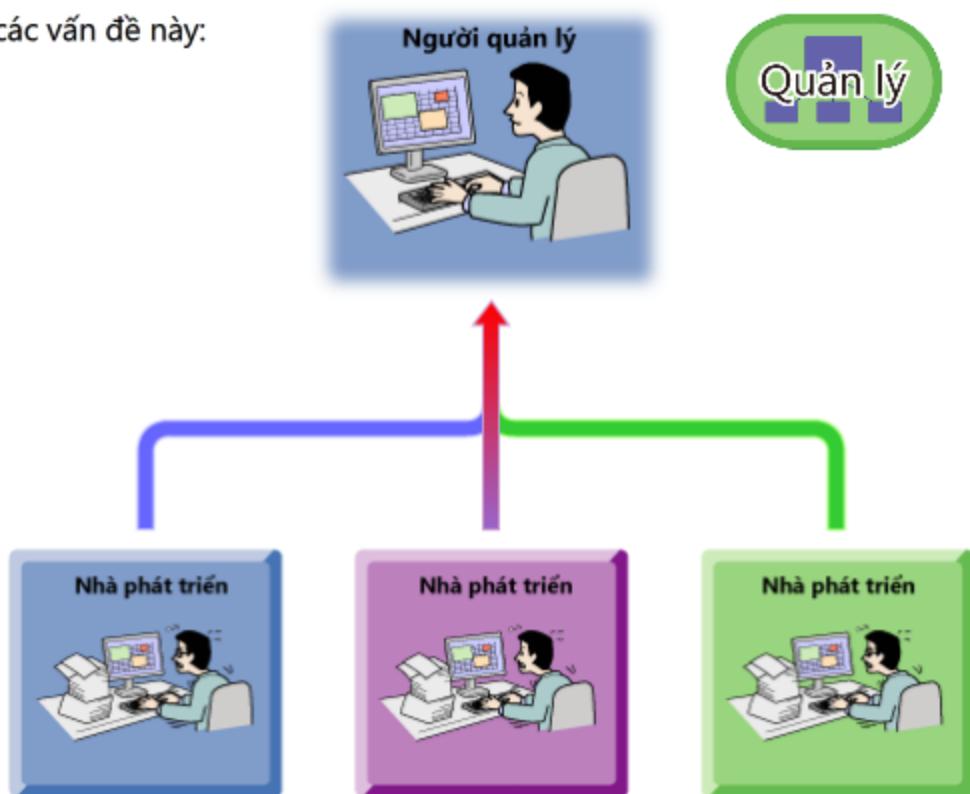
Chương trình PLC quy mô lớn thường được phát triển bởi một nhóm các nhà lập trình.

Từ quan điểm của cơ quan quản lý riêng lẻ và yêu cầu bảo mật dữ liệu, cần phải thiết lập giới hạn phạm vi truy cập dữ liệu và sử dụng chức năng.

Ngoài ra, để đảm bảo khôi phục nhanh trong trường hợp mất chương trình do lỗi bộ điều khiển khả trình, bạn cần phải thực hiện quản lý phiên bản chương trình và sao lưu chương trình định kỳ.

GX Works2 cung cấp các chức năng sau để giải quyết các vấn đề này:

- Bảo mật
- Lịch sử sửa đổi
- Xác thực với máy tính



## 1.1.4

## Biện pháp bảo mật

Chương trình PLC có các kiến thức và dữ liệu quan trọng về mặt chiến lược.

Việc **rò rỉ** bí quyết và dữ liệu từ chương trình ra bên ngoài có thể có ảnh hưởng hủy hoại đến doanh nghiệp.

**Sửa đổi trái phép** chương trình có thể dẫn đến các sự cố sản xuất như làm dừng hệ thống.

Các biện pháp bảo mật phù hợp phải được thực hiện để ngăn chặn các sự cố này.

GX Works2 cung cấp chức năng sau để giải quyết các vấn đề này:

- Bảo mật



**1.2****Quy trình học**

Trong khóa học này, bạn sẽ tìm hiểu về các chức năng của GX Works2 bằng cách thực hiện theo quy trình dành cho phát triển hệ thống thực tế.

(1) Lập trình ..... Chương 2



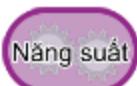
(2) Sửa lỗi ..... Chương 3



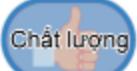
(3) Quản lý dự án và biện pháp bảo mật ..... Chương 4

**<Giải thích các biểu tượng>**

Các biểu tượng hiển thị trên các trang trong Chương 2, 3 và 4 tương ứng với các chức năng cải thiện tương ứng, như trình bày dưới đây.



Chức năng dùng để cải thiện năng suất chương trình



Chức năng dùng để cải thiện chất lượng chương trình



Chức năng dùng để quản lý dự án



Chức năng dùng cho các biện pháp bảo mật

## Chương 2 Lập trình

### Các bước đào tạo trong Chương 2

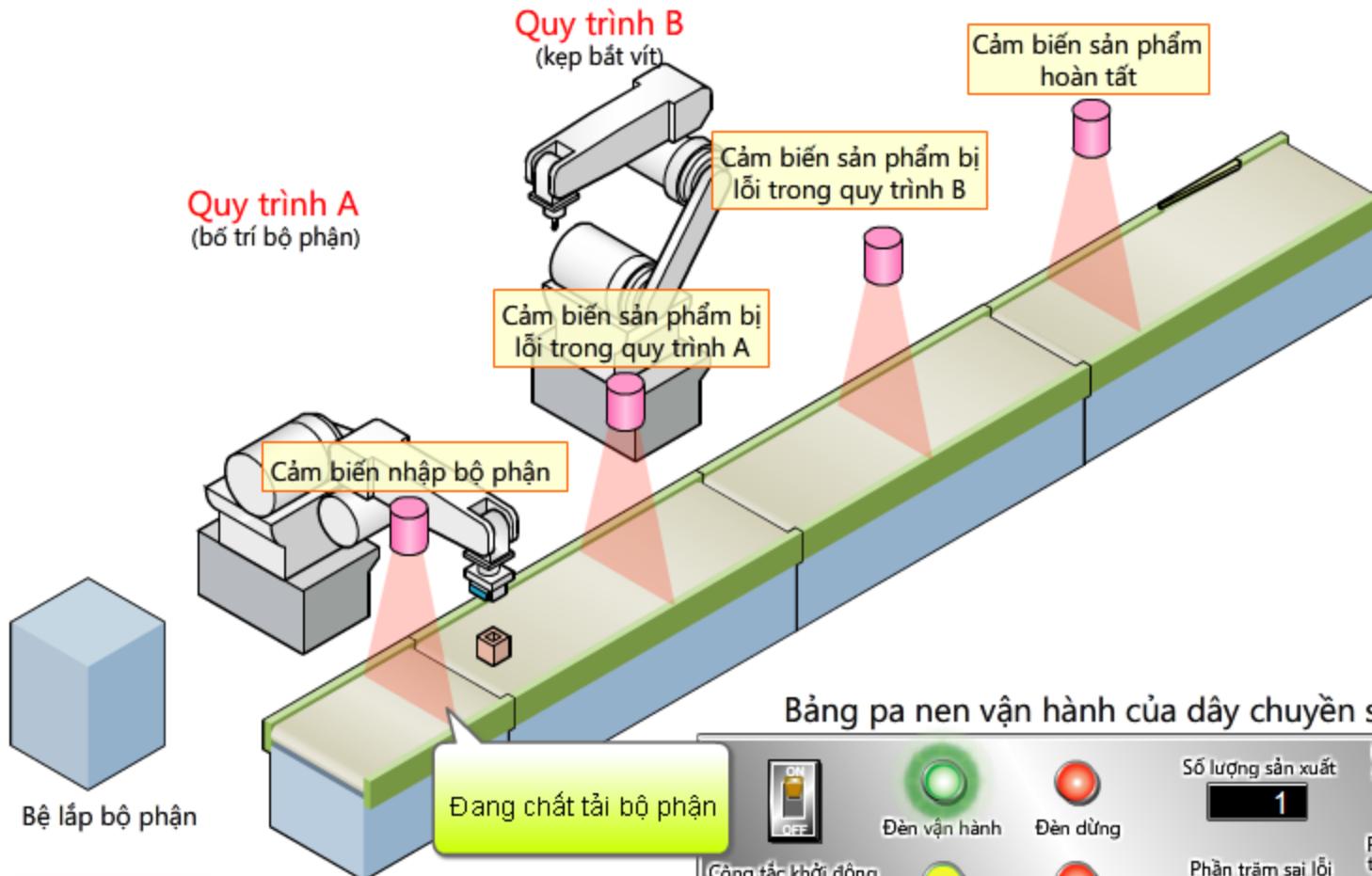
Trong Chương 2, bạn sẽ tìm hiểu về các chức năng được sử dụng để lập trình.

GX Works2 cung cấp nhiều chức năng khác nhau để đảm bảo lập trình hiệu quả và cải thiện chất lượng của các chương trình.

- 2.1 Thay tên thiết bị bằng tên có liên quan đến ứng dụng
  - 2.1.1 Loại nhãn
  - 2.1.2 Ứng dụng và giá trị được lưu của các loại nhãn
  - 2.1.3 Đăng ký nhãn
  - 2.1.4 Tự động phân giao nhãn cho thiết bị
- 2.2 Sắp xếp các khối lập trình PLC dạng thang (ladder) dùng nhiều lần dưới dạng các khối chức năng để chuyển hướng
  - 2.2.1 Tạo và đặt các khối chức năng
  - 2.2.2 Sử dụng thư viện khối chức năng
- 2.3 Thay đổi giá trị ban đầu của thiết bị mà không cần sửa chương trình
- 2.4 Đơn giản hóa chương trình ladder
- 2.5 Tạo chương trình dễ hiểu và dễ đọc
  - 2.5.1 Ghi chú thích cho từng khối lập trình PLC dạng thang (ladder)
  - 2.5.2 Ghi chú thích cho từng đầu ra (cuộn cảm, chỉ lệnh ứng dụng)
- 2.6 Đọc các chương trình có rơ le/dăng ký đặc biệt dễ dàng hơn

## Chương 2 Giới thiệu về hệ thống dùng cho hoạt động học tập trong khóa học này

Khóa học này sử dụng hệ thống giả định dưới đây để giúp bạn tìm hiểu cách sử dụng các chức năng của GX Works2:



Sản xuất bình thường

Lỗi phần trăm sai lỗi  
trong quy trình A

Lỗi phần trăm sai  
lỗi trong quy trình B

Nhấn vào nút "Quay lại phần đầu" để  
quay lại lựa chọn vận hành đầu tiên.

Quay lại phần đầu

Công tắc khởi động		Đèn vận hành		Số lượng sản xuất	<input type="text" value="1"/>	Khoi luong san xuat theo lich	<input type="text" value="20"/>
Đã đạt số lượng sản xuất		Đèn lỗi phần trăm sai lỗi trong quy trình A		Phần trăm sai lỗi trong quy trình A được phép	<input type="text" value="0"/>	Phần trăm sai lỗi trong quy trình A được phép	<input type="text" value="10"/>
Cài lại số lượng sản xuất		Đèn lỗi phần trăm sai lỗi trong quy trình B		Phần trăm sai lỗi trong quy trình B được phép	<input type="text" value="50"/>	Phần trăm sai lỗi trong quy trình B được phép	<input type="text" value="5"/>

## 2.1

## Thay tên thiết bị bằng tên có liên quan đến ứng dụng

Nâng suất

Chất lượng

**Thiết bị** dùng trong chương trình PLC được đặt tên có một chữ cái, theo sau là một số, ví dụ "M0" hoặc "D5".

Do đó, tên thiết bị không đưa ra bất kỳ gợi ý nào về ứng dụng của thiết bị.

Một chương trình quy mô lớn sử dụng nhiều loại thiết bị khác nhau, tức là trong khi lập trình, bạn phải liên tục kiểm tra tài liệu thiết kế hệ thống để xác định ứng dụng của thiết bị. Việc này làm giảm hiệu quả làm việc và ảnh hưởng không tốt đến chất lượng chương trình do các sai sót trong việc chọn thiết bị.

Có thể sử dụng "**Nhãn**" để giải quyết các vấn đề này.

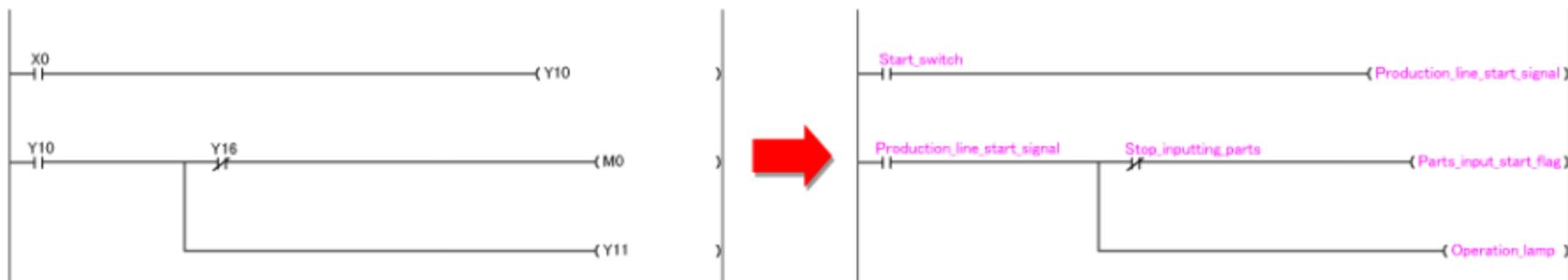
Thay vì sử dụng tên thiết bị, bạn có thể sử dụng tên (nhãn) biểu thị ứng dụng thực tế, ví dụ: "**Production line start signal**" (Tín hiệu khởi động dây chuyền sản xuất).

Đối với loại tên này, có thể sử dụng các ký tự tiếng Nhật (hiragana, katakana và kanji) ngoài các ký tự chữ và số.

### <Hướng dẫn thiết lập "M0" có tín hiệu khởi động dây chuyền sản xuất về ON (BẬT) bằng chỉ lệnh SET (CÀI)>

Thiết bị	—[ SET M0 ]—
Nhãn	—[ SET Production line start signal ]—

Sử dụng loại nhãn này rất hiệu quả cho việc tạo chương trình dễ đọc, cải thiện hiệu quả phát triển chương trình và ngăn được lỗi đầu vào của thiết bị.



## 2.1.1

## Loại nhãn

Năng suất

Chất lượng

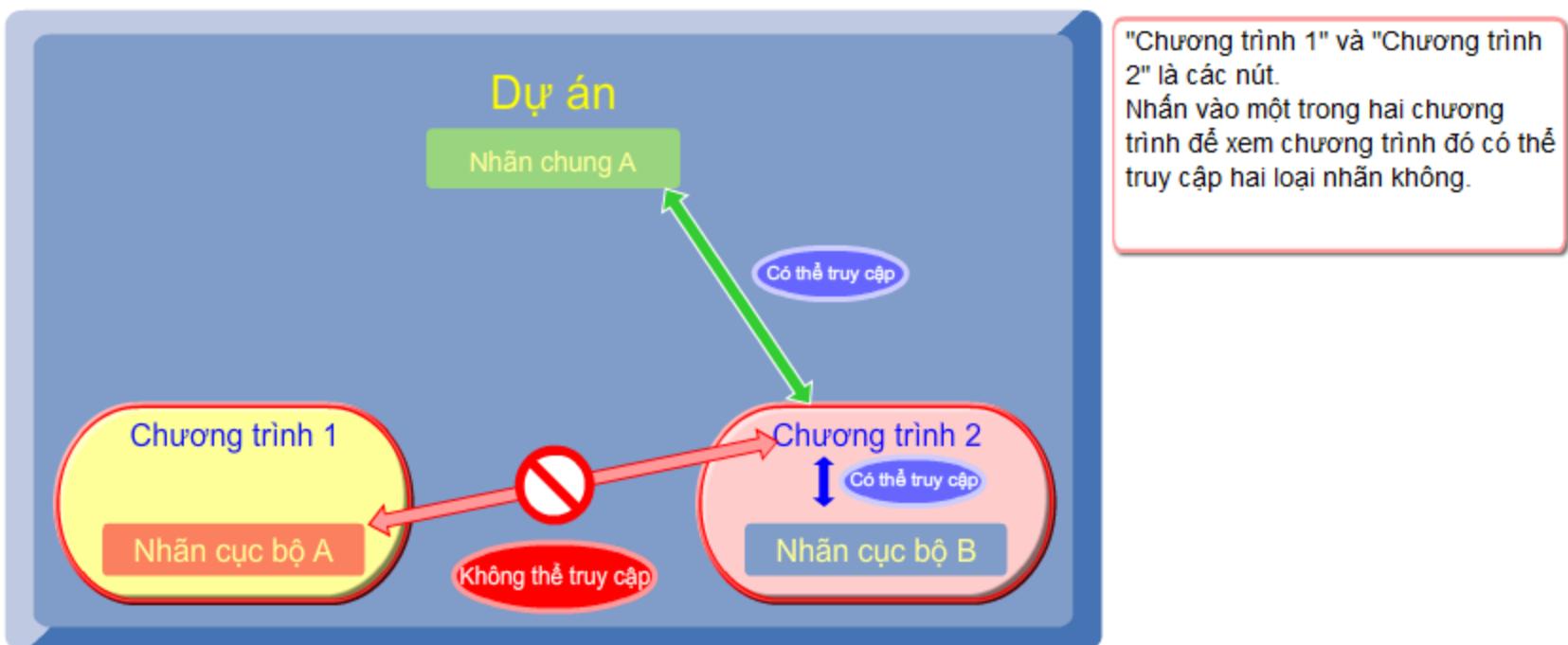
Có hai loại nhãn: "**nhãn chung**" và "**nhãn cục bộ**".

#### <Nhãn chung>

Nhãn chung được sử dụng cho toàn bộ dự án và bất kỳ chương trình nào trong dự án đó cũng có thể truy cập được nhãn này.

#### <Nhãn cục bộ>

Nhãn cục bộ được sử dụng trong một chương trình cụ thể và chỉ chương trình mà nhãn đó được đăng ký mới có thể truy cập được nhãn.



## 2.1.2

## Ứng dụng và giá trị được lưu của các loại nhãn

 Năng suất Chất lượng

Khi đăng ký một nhãn, hãy chỉ rõ ứng dụng của nhãn và loại giá trị có thể lưu bằng cách sử dụng "**Lớp**" và "**Kiểu dữ liệu**".

## &lt;Lớp&gt;

Lớp biểu thị phạm vi sử dụng và ứng dụng của nhãn.  
Có thể chọn các lớp khác nhau tùy thuộc vào loại nhãn.

Lớp	Mục đích sử dụng	Khu vực thiết lập nhãn		
		Nhanchung	Nhãn cục bộ của chương trình	Nhãn cục bộ của khối chức năng
VAR_GLOBAL	Nhãn dùng chung mà các chương trình và khối chức năng trong một dự án có thể sử dụng	<input type="radio"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VAR_GLOBAL_CONSTANT	Nhãn dùng chung có một hằng số mà các chương trình và khối chức năng trong một dự án có thể sử dụng	<input type="radio"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VAR	Nhãn được dùng bởi các chương trình và khối chức năng mà theo đó nhãn được thiết lập	<input checked="" type="checkbox"/>	<input type="radio"/>	<input type="radio"/>
VAR_CONSTANT	Nhãn có một hằng số được dùng bởi các chương trình và khối chức năng mà theo đó nhãn được thiết lập	<input checked="" type="checkbox"/>	<input type="radio"/>	<input type="radio"/>
VAR_RETAIN	Nhãn có một hằng số được dùng bởi các chương trình và khối chức năng mà theo đó nhãn được thiết lập	<input checked="" type="checkbox"/>	<input type="radio"/>	<input type="radio"/>
VAR_INPUT	Nhãn dùng cho đầu vào của một khối chức năng mà theo đó nhãn được thiết lập  <i>* Không thể thay đổi được giá trị này trong một bộ phận của chương trình.</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>
VAR_OUTPUT	Nhãn dùng cho đầu ra của một khối chức năng mà theo đó nhãn được thiết lập	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>
VAR_IN_OUT	Nhãn dùng cho đầu vào và đầu ra của một khối chức năng mà theo đó nhãn được thiết lập  <i>* Có thể thay đổi giá trị này trong một bộ phận của chương trình.</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>

**2.1.2****Ứng dụng và giá trị được lưu của các loại nhãn**

Năng suất

Chất lượng

**<Kiểu dữ liệu>**

Kiểu dữ liệu đề cập đến loại giá trị được lưu trong nhãn.

Kiểu dữ liệu được chỉ định cho nhãn biểu thị loại và phạm vi giá trị có thể lưu trong nhãn và thiết bị tương ứng.

Dưới đây là các kiểu dữ liệu dùng được với chương trình ladder.

Kiểu dữ liệu	Mô tả	Chiều dài bit	Phạm vi giá trị được lưu
Bit	ON hoặc OFF đã được lưu. Tương ứng với thiết bị "M"	1 bit	1: ON, 0: OFF
Tử (đã ký)	Một số nguyên không có phân số đã được lưu.	16 bit	-32768 đến 32767
Tử kép (đã ký)	Một số thực có phân số đã được lưu. Tương ứng với thiết bị "D"	32 bit	-2147483648 đến 2147483647
FLOAT (độ chính xác đơn)	Một số thực có phân số đã được lưu.	16 bit	$-2^{128}$ đến $-2^{-126}$ , 0, $2^{-126}$ đến $2^{128}$
FLOAT (độ chính xác kép)	Tương ứng với thiết bị "D"	32 bit	$-2^{1024}$ đến $-2^{-1022}$ , 0, $2^{-1022}$ đến $2^{1024}$
Chuỗi	Một chuỗi ký tự đã được lưu. Tương ứng với thiết bị "D"	Biến thiên	Tối đa 255 ký tự
Bộ hẹn giờ	BẬT khi đến thời gian được chỉ định. Tương ứng với thiết bị hẹn giờ "T"	-	-
Bộ hẹn giờ có nhớ	BẬT khi đến thời gian được chỉ định. Tương ứng với thiết bị hẹn giờ có nhớ "ST"	-	-
Bộ đếm	BẬT khi đến số đếm được chỉ định. Tương ứng với thiết bị đếm "C"	-	-
Con trỏ	Vị trí bắt đầu vòng lặp chương trình con đã được lưu. Tương ứng với thiết bị trỏ "P"	-	-

cp 1-GX\_Works2\_Advanced\_fod00146\_vie

2.1.3 Đăng ký nhãn

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

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

Navigation

[PRG]Write MAIN (194)Step \* Global Label Setting Global1 Local Label Setting MAIN [P...]

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

Hoàn thành thiết lập cho đăng ký nhãn.  
Nhập để tiếp tục.

English Simple Q03UDE Host Station Line N/A

## 2.1.4

# Tự động phân giao nhãn cho thiết bị

Năng suất

Chất lượng

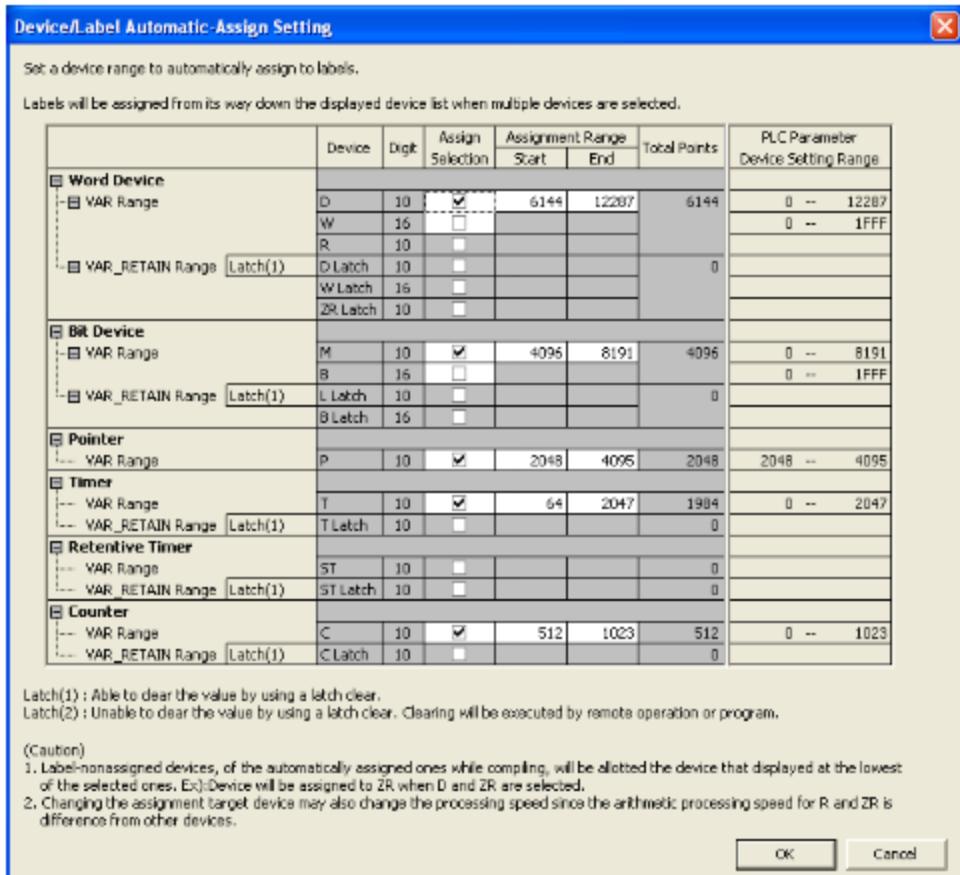
Việc chuyển đổi tự động một chương trình sẽ phân giao một thiết bị phù hợp cho một nhãn theo lớp và kiểu dữ liệu.  
Khi đang sử dụng nhãn, không cần thiết phải biết thiết bị nào đã được phân giao.

Sử dụng "**Device/Label Automatic-Assign Setting**" (Thiết lập phân giao tự động thiết bị/nhãn) để thay đổi phạm vi của các thiết bị được phân giao cho nhãn.

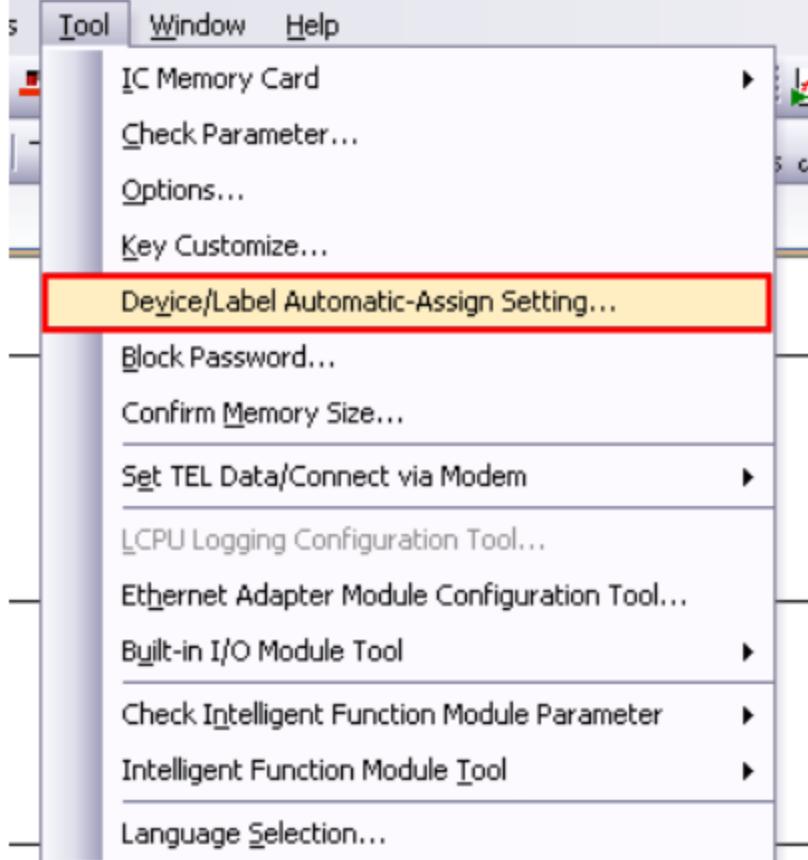
\* Màn hình này là cửa sổ thiết lập thiết bị phân giao tự động cho sê-ri MELSEC-Q và MELSEC-L. Màn hình này có thể khác đối với sê-ri MELSEC-F.

<Khởi động "Device/Label Automatic-Assign Setting" (Thiết lập phân giao tự động thiết bị/nhãn)>

Từ menu GX Works2, chọn [Tool] (Công cụ) – [Device/Label Automatic-Assign Setting] (Thiết lập phân giao tự động thiết bị/nhãn).



## Robot control - [[PRG]Write MAIN 194 Step]



## 2.2

## Sắp xếp các khối lập trình PLC dạng thang (ladder) dùng nhiều lần dưới dạng các khối chức năng để chuyển hướng

Năng suất

Chất lượng

Trong một chương trình quy mô lớn, một số khối lập trình PLC dạng thang (ladder) có thể được sử dụng nhiều lần.

Hoặc một số khối lập trình PLC dạng thang (ladder) có thể được sử dụng trong các chương trình khác nhau.

Hiệu quả làm việc sẽ không thể cải thiện nếu bạn phải nhập cùng một khối lập trình PLC dạng thang (ladder) mỗi lần cần đến.

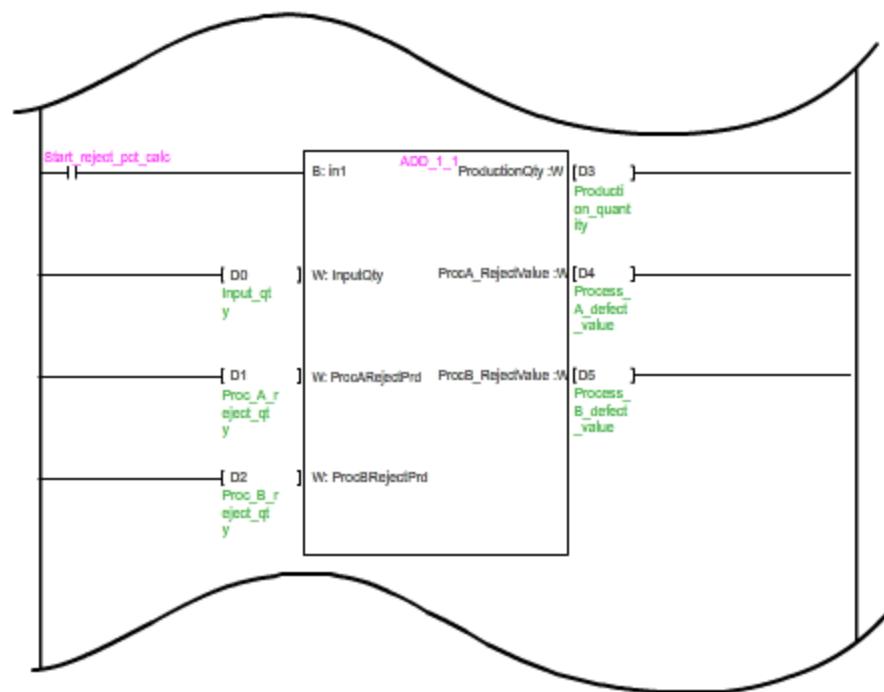
Ngoài ra, nếu phát hiện lỗi trong khối lập trình PLC dạng thang (ladder) dùng chung thì phải xác định và sửa từng khối. Việc làm này mất rất nhiều thời gian.

Sử dụng "**Function Block (FB)**" (Khối chức năng) để giải quyết các vấn đề này.

Một khối lập trình PLC dạng thang (ladder) dùng nhiều lần có thể được **sắp xếp làm khối chức năng**, có thể dùng trong mọi chương trình. Điều này giúp cải thiện đáng kể hiệu quả làm việc.

Nếu phát hiện lỗi trong một khối dùng chung, tất cả những gì bạn cần làm là sửa khối chức năng.

Việc làm này không chỉ tiết kiệm thời gian mà còn ngăn chặn rủi ro không sửa một trong các khối.



Nhấn vào nút để xem hình động từ đầu.



## 2.2.1

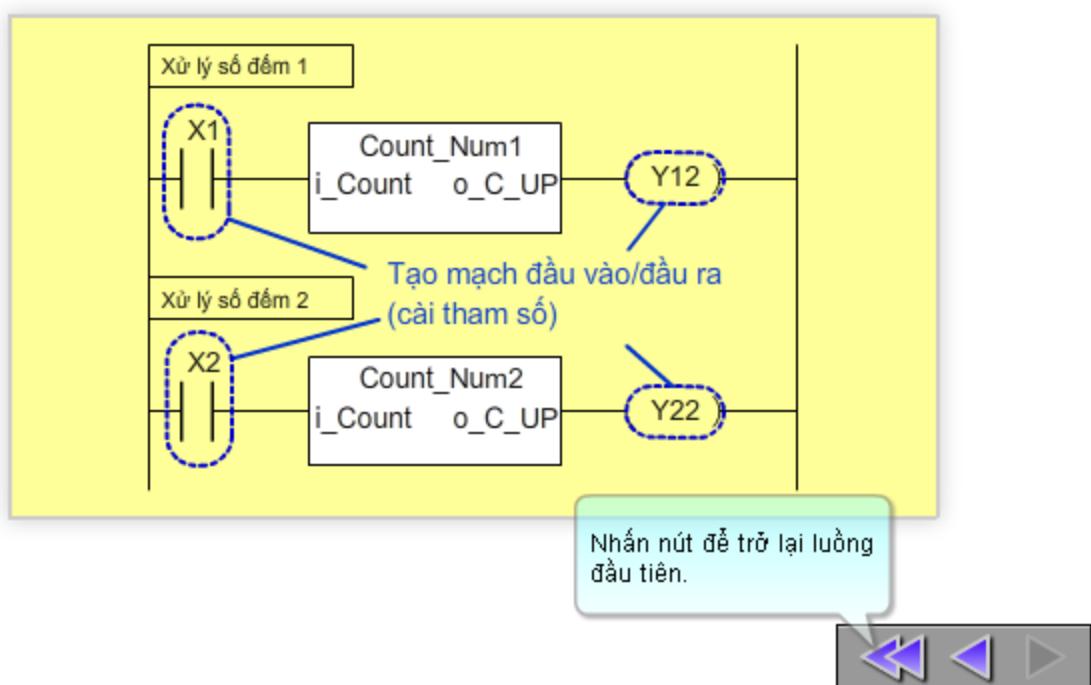
## Tạo và đặt các khối chức năng

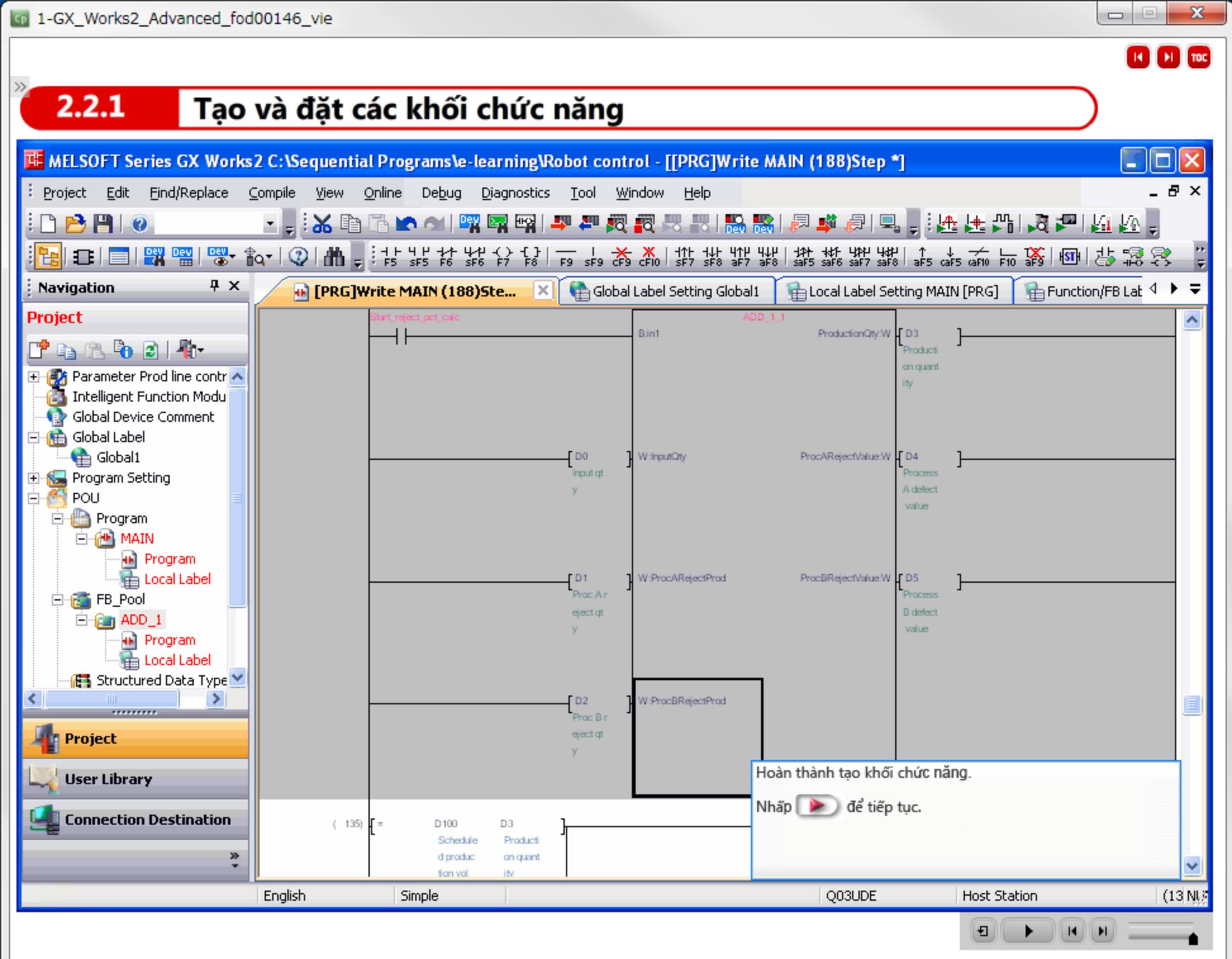
Năng suất

Chất lượng

Vì khối chức năng sẽ được sử dụng trong các chương trình khác nhau nên không thể sử dụng các thiết bị thực (như X, Y và D) trong khối chức năng. Các thiết bị này phải được thay thế bằng các **nhãn** như bạn đã tìm hiểu trong Mục 2.1. Quy trình sắp xếp các khối lập trình PLC dạng thang (ladder) thành một khối chức năng được minh họa bằng hình động.

1. Chuẩn bị một chương trình cần sắp xếp vào khối chức năng.
2. Chia chương trình thành đầu vào và đầu ra, đồng thời thay thế các thiết bị bên trong theo nhãn bên trong để tạo khối chức năng.
3. Đặt khối chức năng đã tạo vào chương trình bằng cách kéo và thả.
4. Chỉ định tham số cho các nhãn đầu vào và đầu ra của khối chức năng mà bạn đã đặt trong chương trình.
5. Tạo mạch đầu vào dẫn chuyển tham số đến nhãn đầu vào và mạch đầu ra nhận tham số từ nhãn đầu ra, trước và sau khối chức năng.





**2.2.2****Sử dụng thư viện khối chức năng**

Năng suất

Chất lượng

Bây giờ, bạn đã biết cách tạo một khối chức năng.

Mitsubishi Electric đã sắp xếp các chương trình điều khiển của nhiều mô đun khác nhau như CPU, mô đun đầu vào/đầu ra analog, mạng và mô đun định vị trí dưới dạng các khối chức năng, được cung cấp miễn phí tại "**Thư viện FB**".

Sử dụng thư viện FB giúp loại bỏ nhu cầu phát triển các chương trình điều khiển mô đun mà trước đây người dùng thường phải đảm nhiệm. thậm chí cả các mô đun mà bạn không quen thuộc cũng có thể được đưa vào một cách dễ dàng.

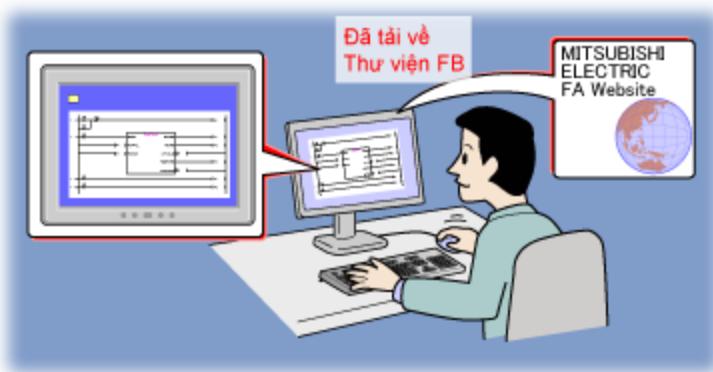
Bạn có thể tải về thư viện FB từ trang "**MELSOFT Library Download**" (Tải về thư viện MELSOFT) trên **MITSUBISHI ELECTRIC FA Website**.

\* Nhấp vào URL sau đây để truy cập trang tải về.\*

[http://wwwf2.mitsubishielectric.co.jp/melfansweb/plceng/download/library/index\\_e.htm](http://wwwf2.mitsubishielectric.co.jp/melfansweb/plceng/download/library/index_e.htm)

<Trước khi dùng>

<Nếu bạn sử dụng thư viện FB>



\* Không có thư viện FB được cung cấp cho sê-ri MELSEC-F.

Các sản phẩm được hỗ trợ được liệt kê trên trang tải về. Vui lòng kiểm tra sản phẩm được hỗ trợ trước khi tải về.

## 2.3 Thay đổi giá trị ban đầu của thiết bị mà không cần sửa chương trình

Năng suất

Chất lượng

Giá trị ban đầu hoặc hằng số của một thiết bị thường được cài bằng cách sử dụng chỉ lệnh MOV trước khi xử lý chương trình chính.

Trong trường hợp này, chương trình phải được sửa trực tiếp mỗi lần thay đổi hoạt động của chương trình theo ứng dụng của hệ thống.

Phương pháp này không chỉ tốn thời gian mà còn có rủi ro xuất hiện lỗi khi sửa hoặc không sửa được.

Sử dụng "**Giá trị ban đầu của thiết bị**" để giải quyết các vấn đề này.

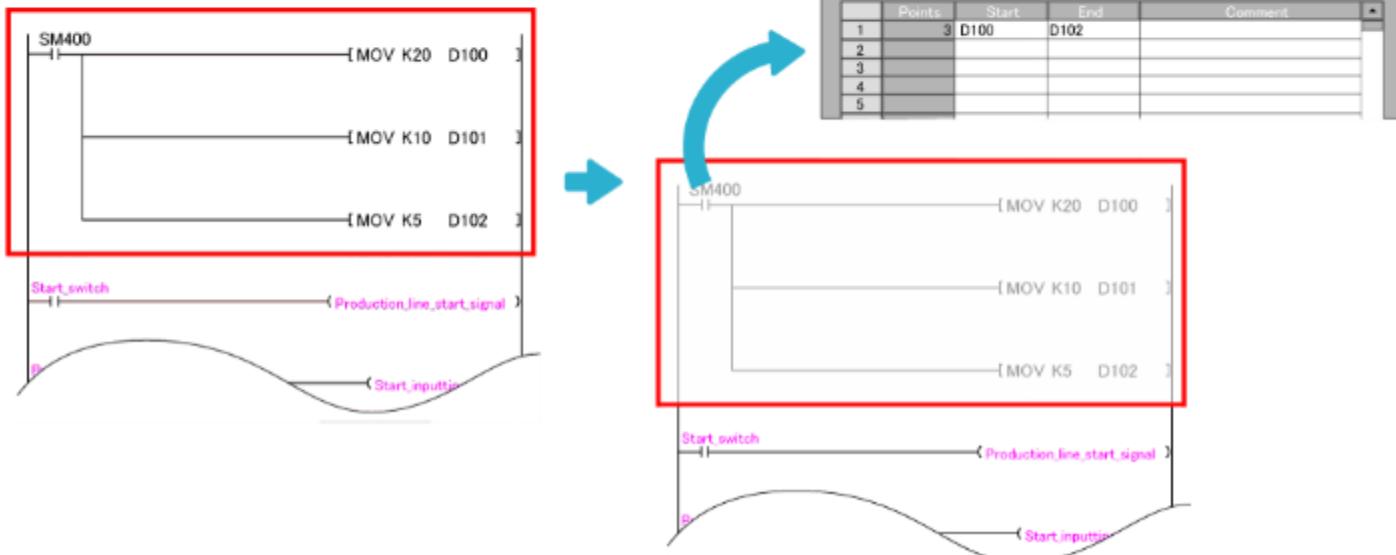
Sử dụng chức năng của GX Works2 đảm bảo việc quản lý hợp lý các giá trị ban đầu của thiết bị và loại bỏ nhu cầu sửa chương trình, nhờ vậy cho phép bạn tạo ra các chương trình hiệu quả hơn nhiều.

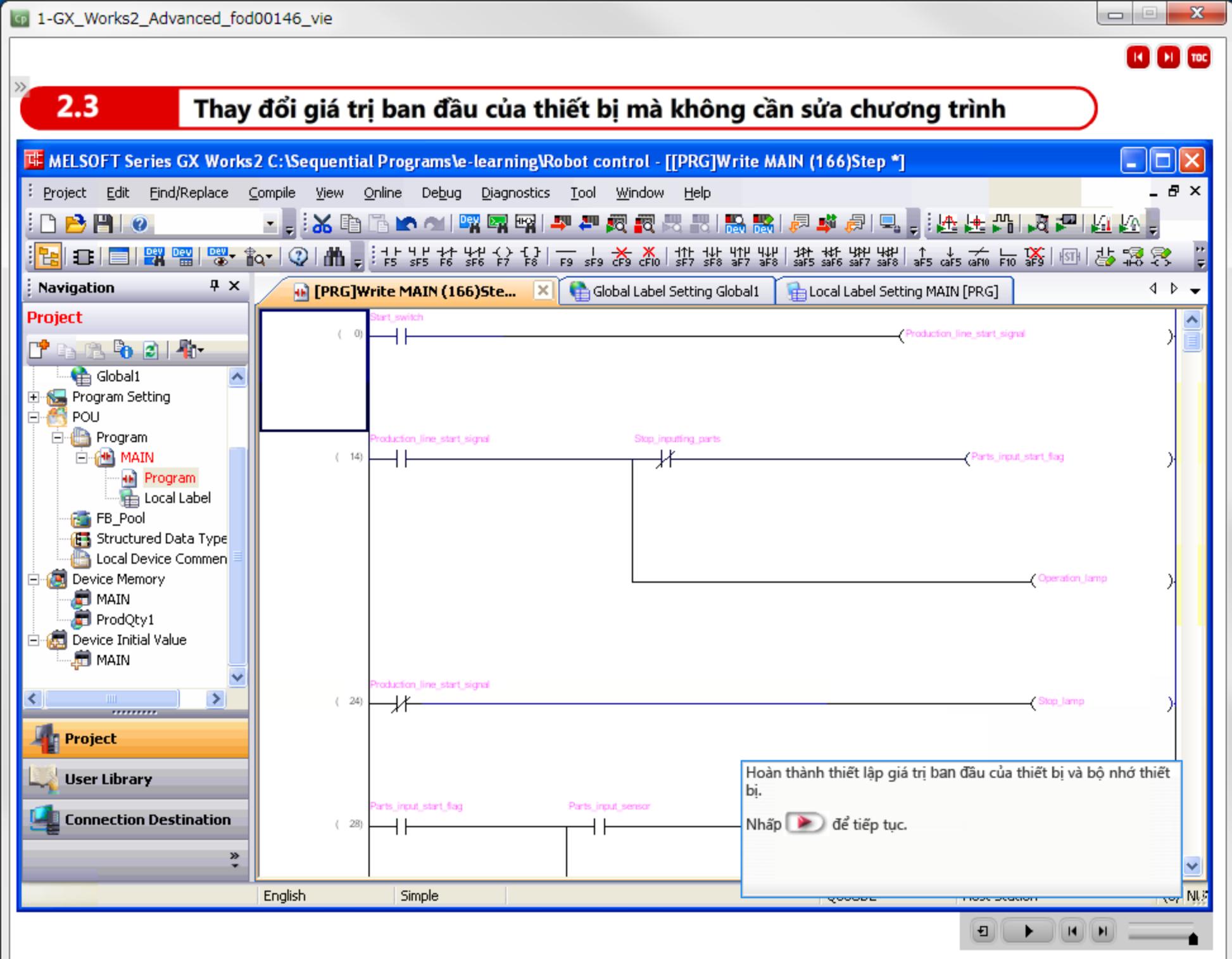
Ngoài ra, chức năng này còn loại bỏ nhu cầu sử dụng chương trình thiết lập giá trị thiết bị ban đầu, giúp giảm khối lượng chương trình (mức sử dụng bộ nhớ) và giảm rủi ro bị lỗi.

Đối với giá trị ban đầu của thiết bị, hãy chỉ định phạm vi của các thiết bị để thiết lập giá trị ban đầu.

Giá trị ban đầu thực tế được lưu trong **bộ nhớ thiết bị** và được phân giao cho phạm vi thiết bị được chỉ định.

Với các vùng bộ nhớ thiết bị được chuẩn bị theo ứng dụng của hệ thống, bạn chỉ cần thay đổi vùng bộ nhớ thiết bị cần phân giao để thay đổi giá trị ban đầu của thiết bị.





## 2.4

## Đơn giản hóa chương trình ladder

Năng suất

Chất lượng

Trong chương trình ladder quy mô lớn, việc kết nối các thiết bị, chỉ lệnh và dòng theo quy luật có thể trở nên khá phức tạp, khiến việc nắm bắt hoạt động xử lý của chương trình trở nên khó khăn.

Đặc biệt, các phép tính số học, như công thức được hoàn thành trên một dòng đơn, phải được lập trình bằng cách kết hợp các chỉ lệnh.

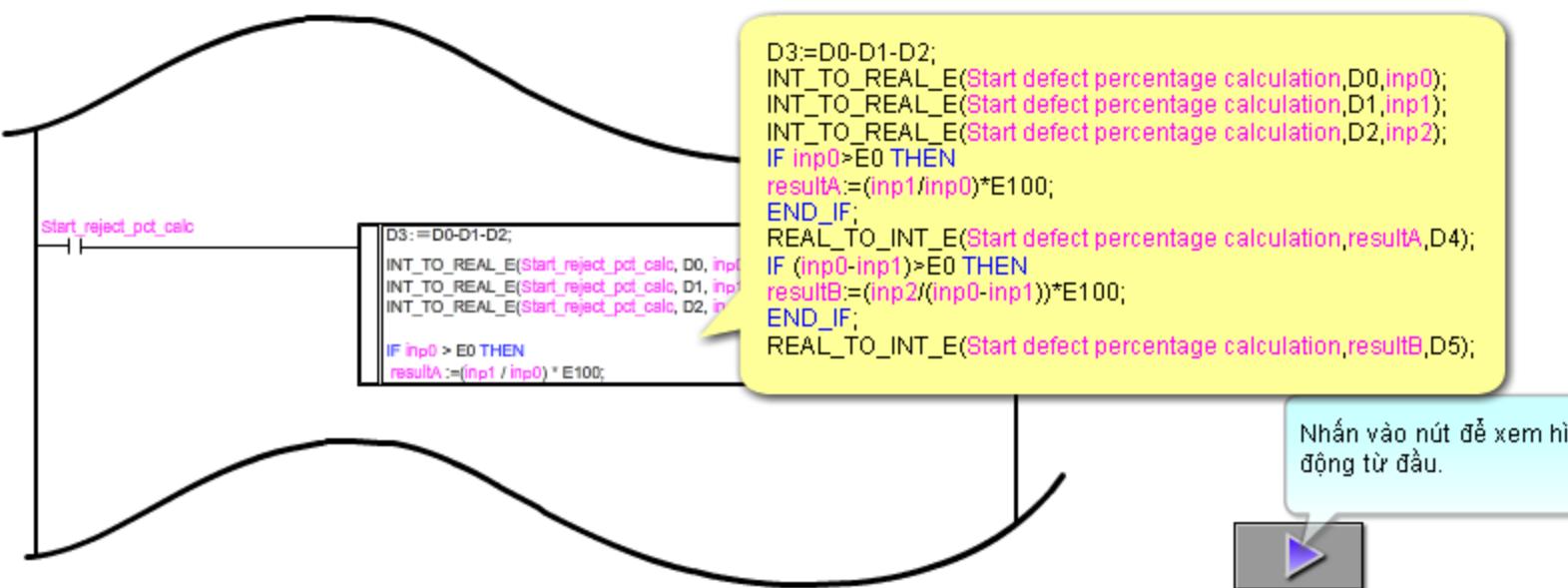
Sử dụng "**Inline Structured Text**" (Văn bản nội tuyến có cấu trúc) để giải quyết các vấn đề này.

Chương trình ladder được thay bằng một chương trình được biên soạn một phần bằng **ngôn ngữ văn bản cấu trúc (ST)**.

Ngôn ngữ ST là ngôn ngữ lập trình kiểm soát tuần tự tương tự như ngôn ngữ C dùng cho lập trình phần mềm máy tính.

Có thể biên soạn các phép tính số học bằng các công thức, vì vậy ngay cả những nhà lập trình không quen với C cũng có thể sử dụng văn bản nội tuyến có cấu trúc.

Hình sau đây mô tả một ví dụ mà trong đó một phần của chương trình ladder cho hệ thống được thay bằng văn bản nội tuyến có cấu trúc. Bạn có thể thấy chương trình ladder phức tạp giờ đây đã dễ hiểu hơn.



## 2.4

## Đơn giản hóa chương trình ladder

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

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

Navigation [PRG]Write MAIN 114 Step Global Label Setting Global Local Label Setting MAIN [PRG]

**Project**

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

**Project**

**User Library**

**Connection Destination**

Start\_reject\_pct\_calc ( 75 )

```

INT_TO_REAL_E(Start_reject_pct_calc,D2,inp2);
IF inp0>E0 THEN
resultA:=(inp1/inp0)*E100;
END_IF;
REAL_TO_INT_E(Start_reject_pct_calc,resultA,D4);
IF (inp0>inp1)>E0 THEN
resultB:=(inp2/(inp0-inp1))*E100;
END_IF;
REAL_TO_INT_E(Start_reject_pct_calc,resultB,D5);
  
```

( 87 ) = D100 D3 Production quantity

( 94 ) > D4 Process A defect value

( 94 ) > D5 D101 Process A defect threshold

( 94 ) > D102 Proc\_A\_reject\_pct\_error\_lamp

Production\_qty\_attained

Production\_qty\_attained\_lamp

Hoàn thành thiết lập văn bản nội tuyến có cấu trúc.  
Nhập để tiếp tục.

English Simple Q03UDE Host Station (83 NV)

## 2.5

## Tạo chương trình dễ hiểu và dễ đọc



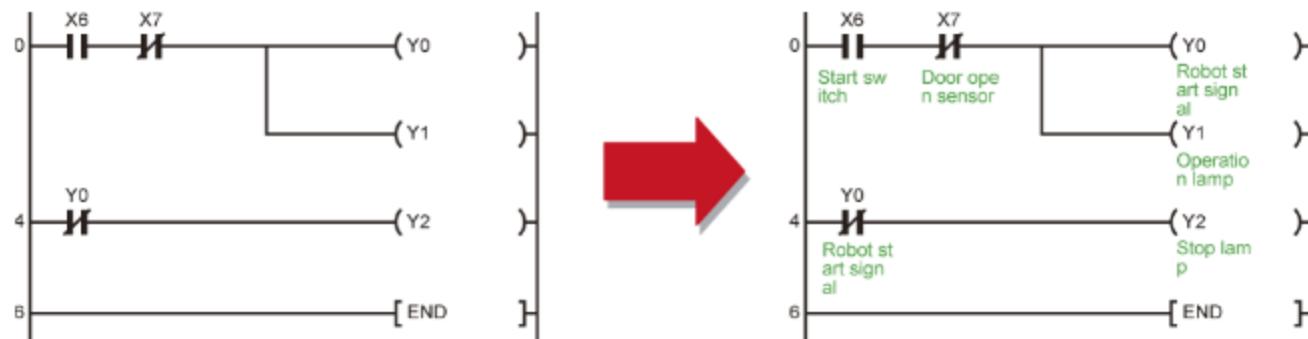
Bạn có thể thấy khó để hiểu được các chi tiết điều khiển trong một chương trình quy mô lớn khi mới chỉ nhìn vào chương trình. Có thể dẫn đến các vấn đề sau đây:

- Bạn làm cho chương trình báo lỗi (như nhập sai chỉ lệnh hoặc thiết bị).
- Bạn không thể tìm ra nguyên nhân khiến chương trình báo lỗi.
- Người đảm nhiệm việc lập trình không hiểu được chi tiết điều khiển.

Sử dụng "**Chú thích**" để giải quyết các vấn đề này.

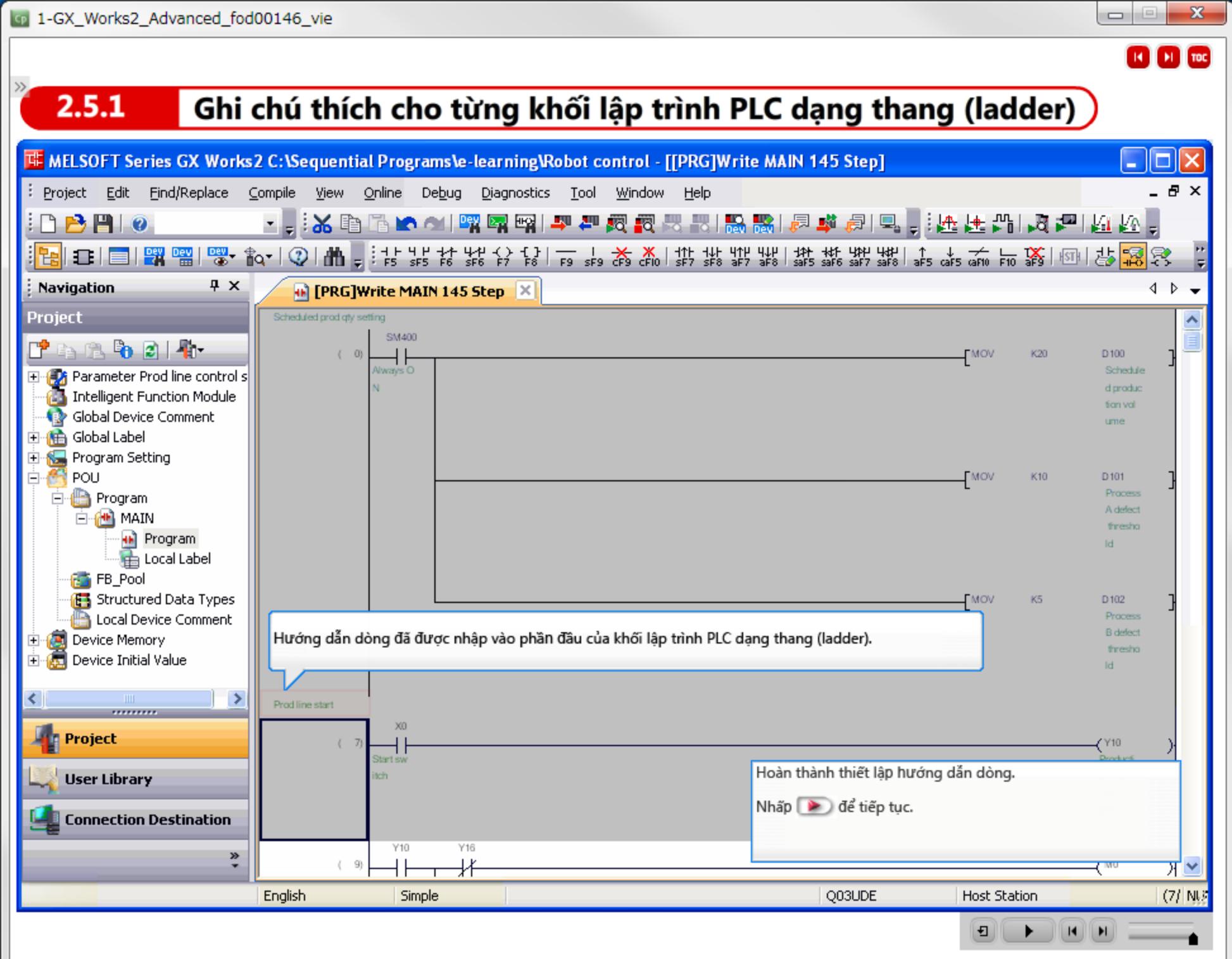
Các bản ghi nhớ như thông tin điều khiển và tên thiết bị có thể được gắn với chương trình để bạn có thể dễ dàng hiểu được chi tiết điều khiển.

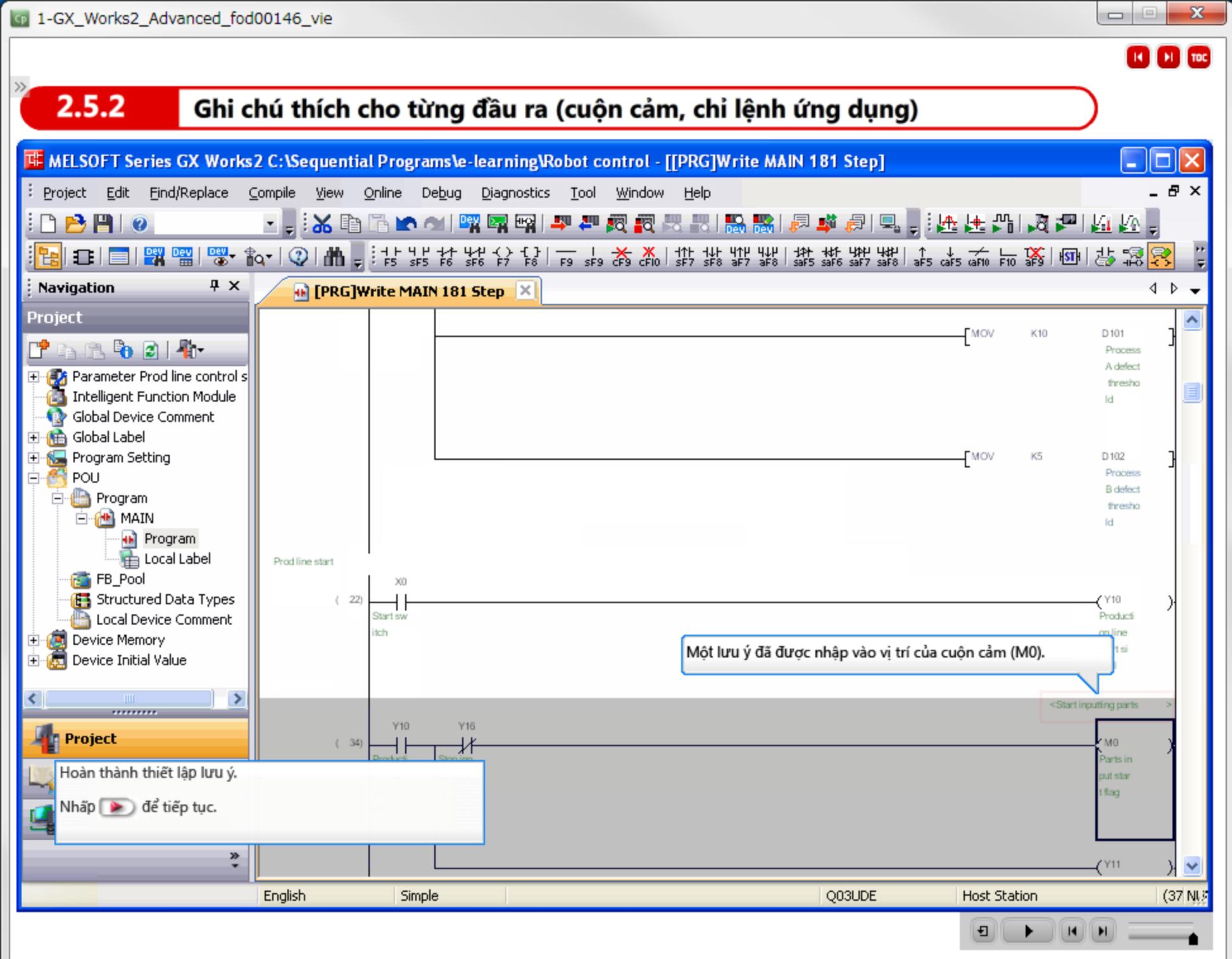
Những chú thích này phải được nhập ở bất kỳ đâu có thể để tạo các chương trình dễ hiểu không chỉ đối với bạn mà cho cả những người khác.



GX Works2 cho phép nhập các chú thích sau.

Loại chú thích	Phạm vi chú thích
Ghi chú thiết bị	Có thể gắn chú thích với một thiết bị. Chú thích này biểu thị ứng dụng của từng thiết bị và loại thiết bị I/O được kết nối.
Hướng dẫn	Có thể gắn chú thích với một khối lập trình PLC dạng thang (ladder). Chú thích này giúp luồng xử lý trở nên dễ hiểu.
Lưu ý	Chú thích có thể được gắn với một cuộn cảm/chỉ lệnh ứng dụng trong chương trình. Chú thích này giúp nội dung về cuộn cảm (đầu ra) và chỉ lệnh ứng dụng trở nên dễ hiểu.





## 2.6

## Đọc các chương trình có rơ le/đăng ký đặc biệt dễ dàng hơn

 Nâng suất Chất lượng

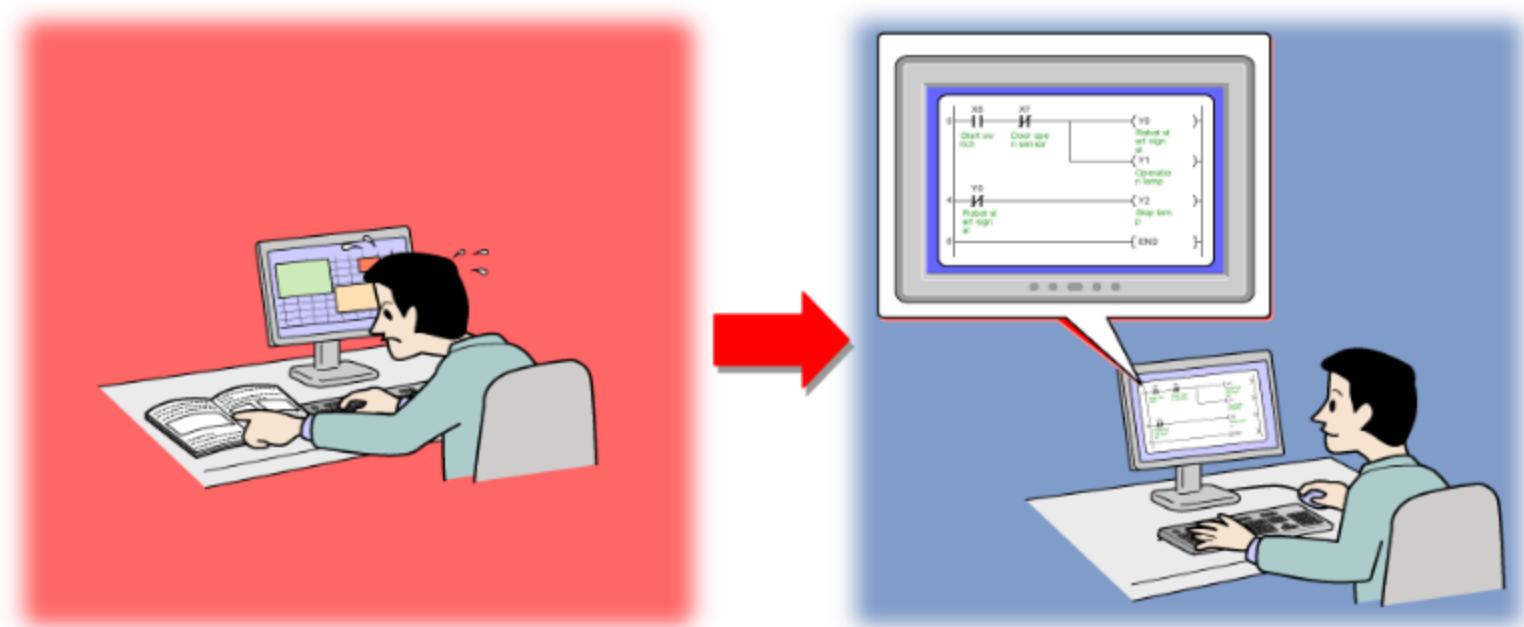
Nếu các rơ le đặc biệt, đăng ký đặc biệt và/hoặc tài nguyên của mô đun chức năng thông minh được sử dụng trong một chương trình thì người dùng khó có thể hiểu được tất cả các ứng dụng và chức năng của các thiết bị đó. Để đọc chương trình, bạn phải cần đến hướng dẫn sử dụng.

Mặc dù chương trình có thể dễ đọc hơn nếu có chú thích đi kèm với từng thiết bị nhưng điều này sẽ đòi hỏi rất nhiều thời gian và công sức nếu bạn phải đính kèm chú thích cho nhiều thiết bị.

Sử dụng "**Chú thích mẫu**" để giải quyết các vấn đề này.

GX Works2 cung cấp chú thích mẫu mô tả ứng dụng và chức năng của các rơ le đặc biệt, đăng ký đặc biệt và tài nguyên của mô đun chức năng thông minh.

Sử dụng các chú thích mẫu này giúp việc đính kèm chú thích vào thiết bị trở nên dễ dàng hơn để chương trình dễ đọc hơn. Có thể sửa đổi các chú thích mẫu khi cần thiết.



## 2.6

## Đọc các chương trình có rơ le/đăng ký đặc biệt dễ dàng hơn

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



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



## Navigation

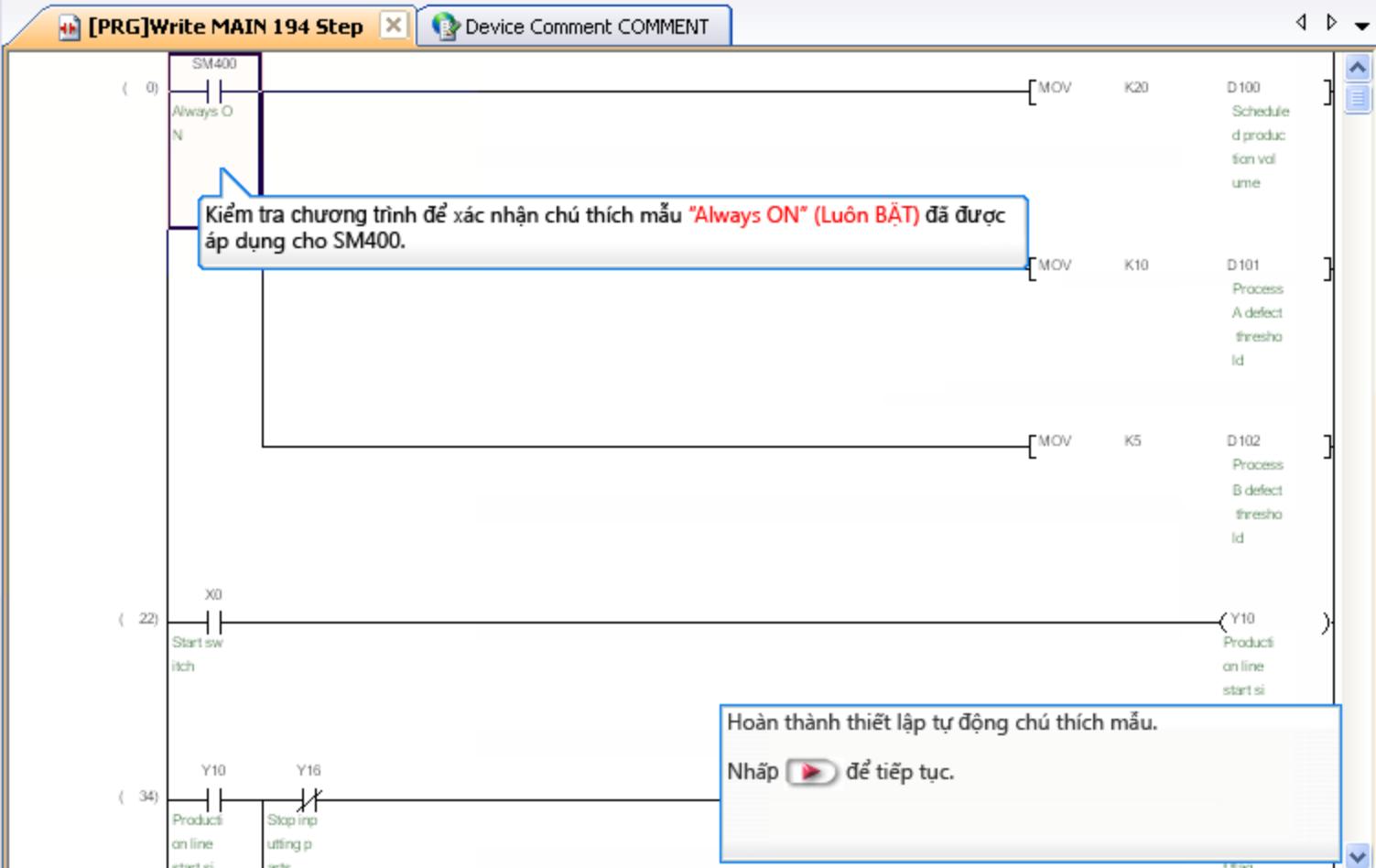
## Project

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

## Project

## User Library

## Connection Destination



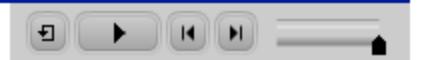
English

Simple

Q03UDE

Host Station

(15 N)



## Chương 3 Sửa lỗi

### Các bước đào tạo trong Chương 3

Trong Chương 3, bạn sẽ tìm hiểu về các chức năng được sử dụng để sửa lỗi.

GX Works2 cung cấp nhiều công cụ theo dõi và sửa lỗi khác nhau để sửa lỗi.

Tạo các chương trình chất lượng, không có lỗi sử dụng các công cụ theo dõi và sửa lỗi này.

- 3.1 Chỉ theo dõi thiết bị và nhãn đích
- 3.2 Kiểm tra trạng thái sử dụng thiết bị và nhãn
- 3.3 Thu thập thông tin về hoạt động thay đổi giá trị thiết bị theo thời gian
- 3.4 Thay đổi giá trị thiết bị mà không cần sửa chương trình
- 3.5 Sửa lỗi vận hành chương trình theo từng bước
  - 3.5.1 Sử dụng chức năng thực hiện từng bước
- 3.6 Mô phỏng hoạt động của thiết bị ngoại vi
  - 3.6.1 Nhập giá trị thiết bị để thiết lập hệ thống I/O
  - 3.6.2 Sử dụng định dạng biểu đồ thời gian để thiết lập hệ thống I/O

## 3.1

## Chỉ theo dõi thiết bị và nhãn đích



Một chương trình sử dụng nhiều chỉ lệnh và thiết bị.

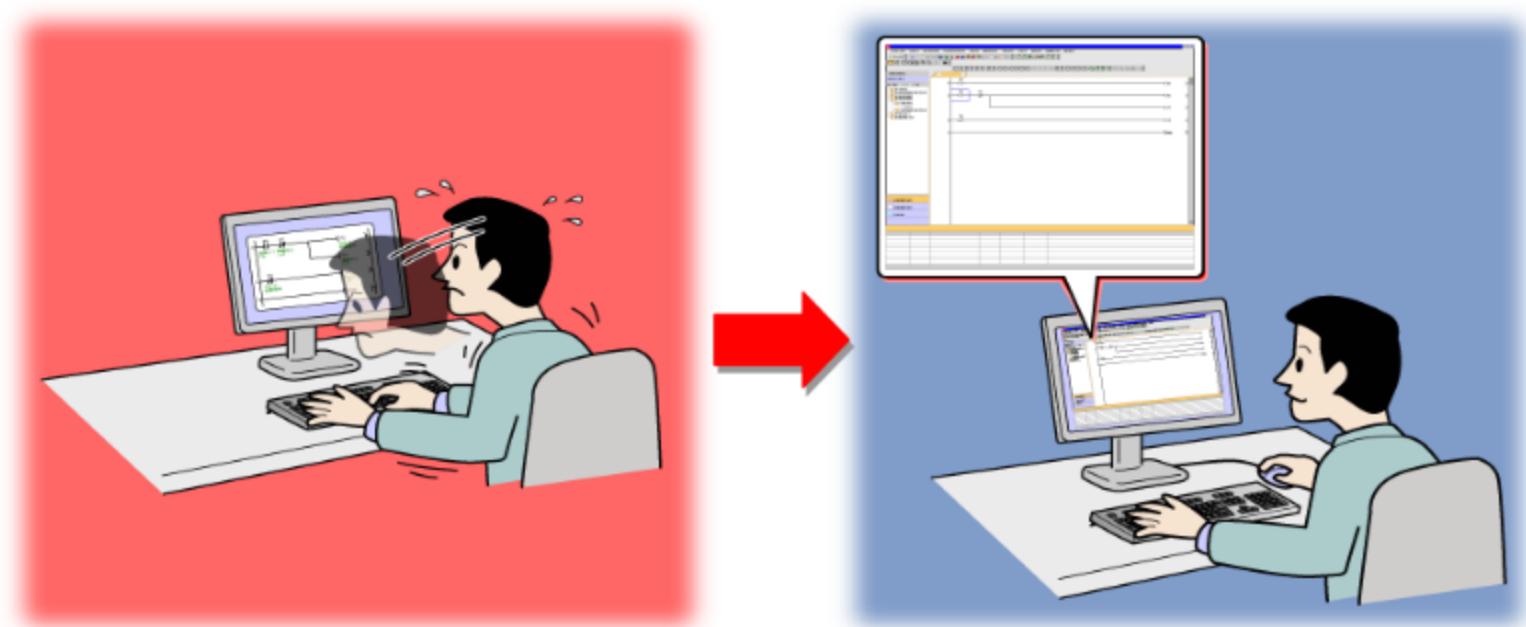
Ngoài ra, chiều dài dọc dài của chương trình có nghĩa là chỉ có một phần của chương trình được hiển thị ở một thời điểm trên màn hình giám sát của máy tính.

Do đó, chỉ riêng màn hình ladder không thể theo dõi được toàn bộ hoạt động.

Sử dụng "**Theo dõi**" để giải quyết các vấn đề này.

Có thể dùng chức năng này để chỉ theo dõi các thiết bị và nhãn mà bạn đã chỉ định trước.

Có thể tạo hai Theo dõi trở lên để đảm bảo từng phạm vi đều được theo dõi.



cp 1-GX\_Works2\_Advanced\_fod00146\_vie

3.1 Chỉ theo dõi thiết bị và nhãn đích

MELSOFT Series GX Works2 C:\Sequential Programs\le-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...

Prod qty calc

M1

( 74)

Start re ject pct calc

D1 D2 D6

0 0 0

GX Simulator 2

Tool Options

Switch

RESET  RUN

LED

MODE RUN

ERR. USER

Các giá trị hiện tại của thiết bị đã đăng ký với Watch 1 (Theo dõi 1) được theo dõi.

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	

Watch 1 Watch 2

English Simple

Q03UDE Simulation (84 N)

Hoàn thành đăng ký và theo dõi Watch (Theo dõi).

Nhập  để tiếp tục.

## 3.2

## Kiểm tra trạng thái sử dụng thiết bị và nhãn

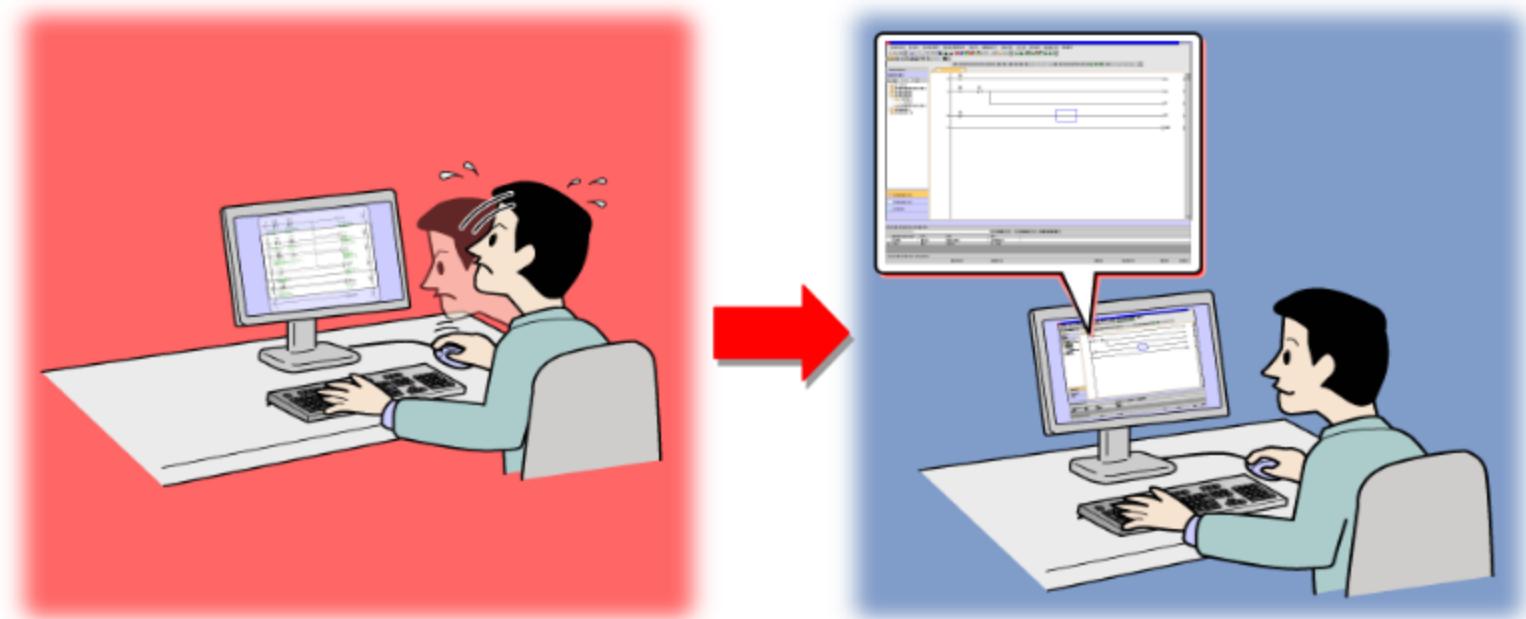


Một chương trình sử dụng các thiết bị và nhãn giống nhau tại nhiều vị trí khác nhau.

Bạn có thể muốn kiểm tra trạng thái sử dụng của các thiết bị và nhãn này bằng cách so sánh chúng giữa các vị trí.

Sử dụng "**Tham chiếu chéo**" để thực hiện việc này.

Chức năng này liệt kê vị trí của các thiết bị và nhãn khớp với điều kiện tìm kiếm, nhờ đó bạn có thể so sánh chúng và kiểm tra việc sử dụng không đúng.



cp 1-GX\_Works2\_Advanced\_fod00146\_vie

3.2 Kiểm tra trạng thái sử dụng thiết bị và nhãn

MELSOFT Series GX Works2 C:\Sequential Programs\le-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

Prod qty calc

( 74) M1 Start re-  
ject pct calc

D1 Proc A r  
eject qt y

D2 Proc B r  
eject qt y

D6 Total de  
fective  
products

D6 Total de  
fective  
products

D3 Product  
on quant  
ity

Con trả chương trình đã di chuyển đến vị trí của bước Số 74.

FLT

D0 Input qt y

D10 Converted input

Project

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

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	Hoàn thành sử dụng Cross Reference (Tham chiếu chéo).
D1	D1	+P	-[-]	Step No.63	Nhập  để tiếp tục.
D1	D1	+	-[-]	Step No.84	
D1	D1	FLT	-[-]	Step No.92	

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

English Simple Q03UDE Host Station (84 NL)

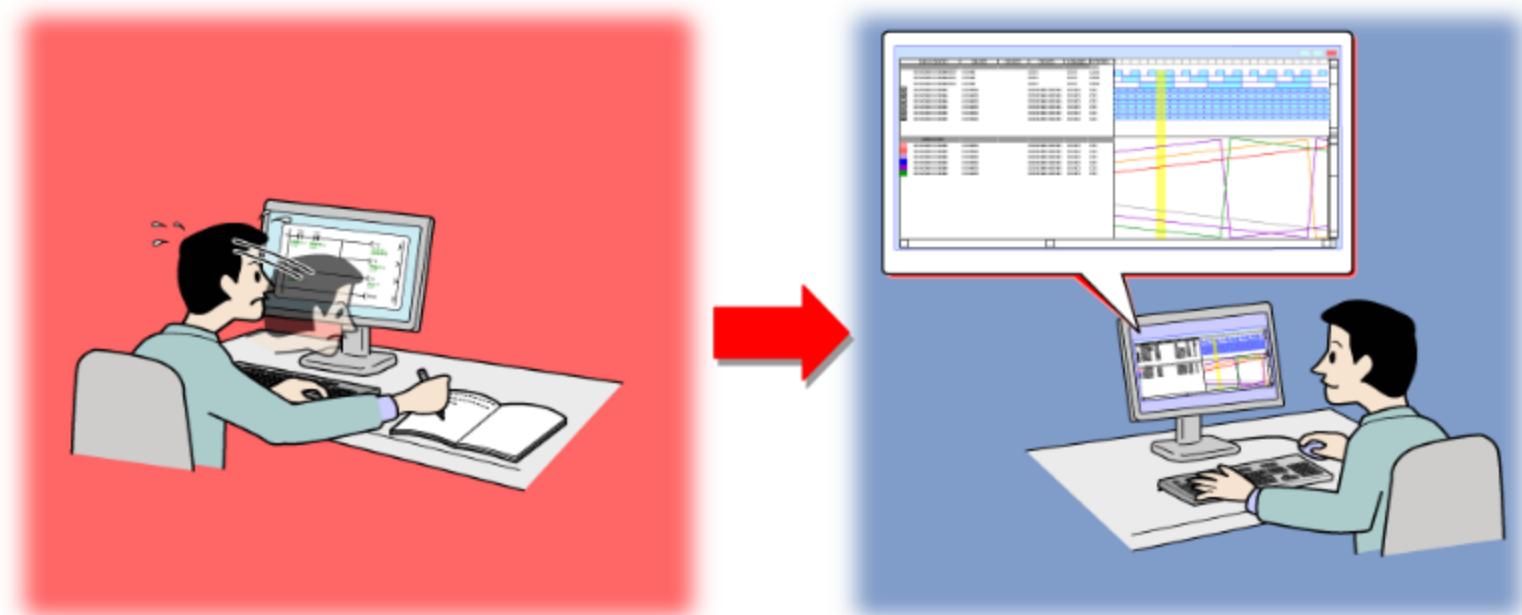
### 3.3 Thu thập thông tin về hoạt động thay đổi giá trị thiết bị theo thời gian



Bạn có thể muốn xác nhận việc thay đổi giá trị thiết bị và nhãn có nằm trong phạm vi thiết kế hay không hoặc xử lý sự cố bằng cách kiểm tra các thay đổi đã xảy ra trong sự kiện lỗi đó.

Sử dụng "**Theo vết lấy mẫu**" trong trường hợp này. (Chỉ dành cho sê-ri MELSEC-Q và MELSEC-L)

Có thể sử dụng chức năng này để theo dõi và ghi lại các thay đổi đối với giá trị thiết bị và nhãn theo thời gian. Có thể lưu bản ghi vào các tập tin.



cp 1-GX\_Works2\_Advanced\_fod00146\_vie

3.3 Thu thập thông tin về hoạt động thay đổi giá trị thiết bị theo thời gian

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

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

Navigation

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

Device/Label	Device	Comment	Data Type
M0	M0	Parts input sta	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

D3	D3	Production qt	Word[Signed]
D4	D4	Process A de	Word[Signed]
D5	D5	Process B de	Word[Signed]

Hoàn thành thiết lập theo vết lấy mẫu.  
Nhập  để tiếp tục.

English Simple Q03UDE Simulation N13

## 3.4

## Thay đổi giá trị thiết bị mà không cần sửa chương trình



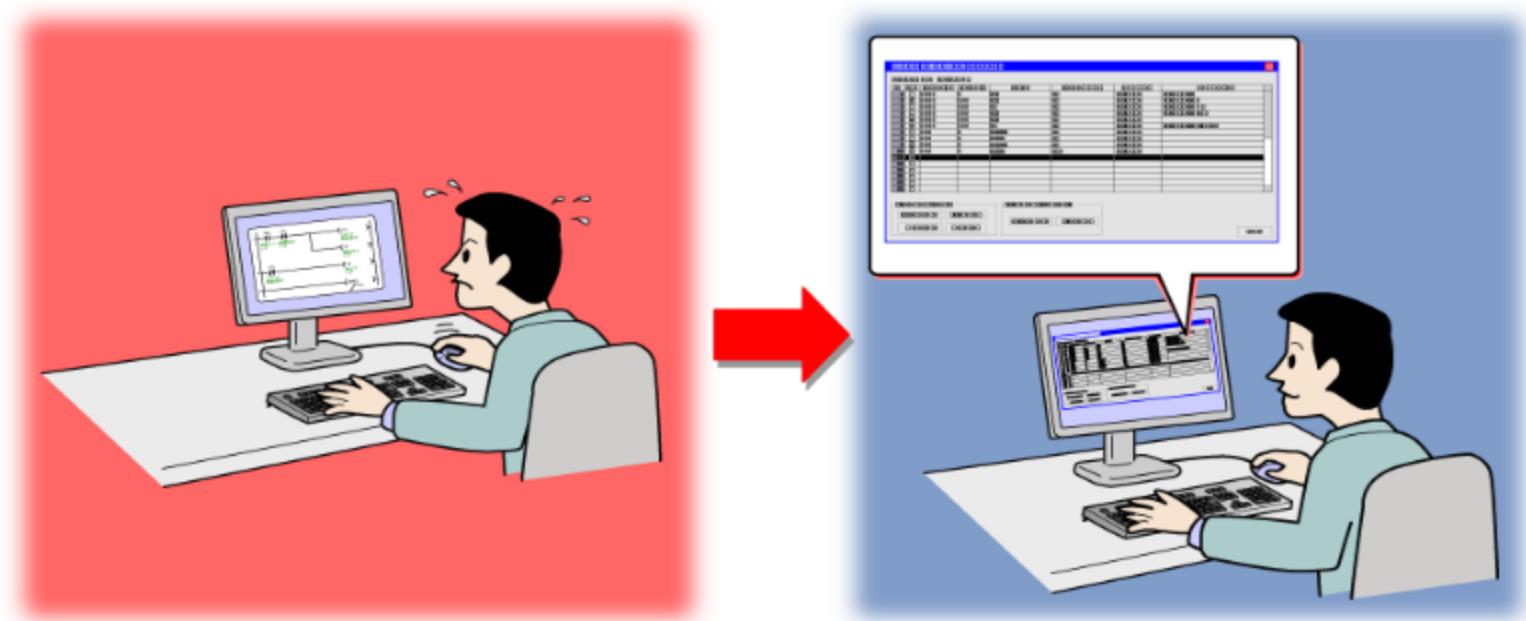
Khi thực hiện sửa lỗi, bạn có thể muốn thay đổi cưỡng bức các giá trị thiết bị để kiểm tra sự khác nhau trong vận hành chương trình.

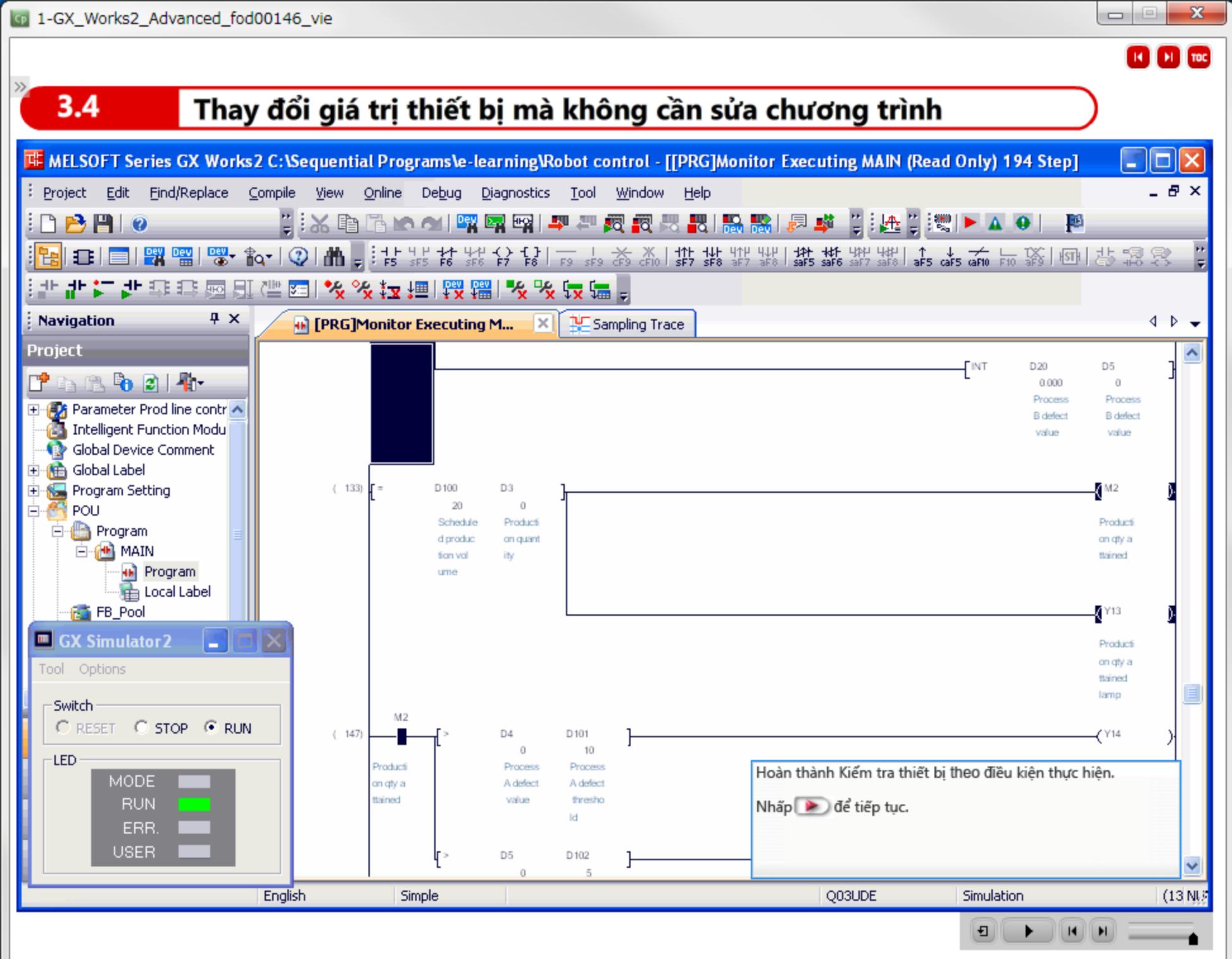
Tuy nhiên, điều này có nghĩa là bạn phải sửa đổi chương trình mỗi lần một giá trị thiết bị được thay đổi. Việc làm này sẽ mất khá nhiều thời gian và công sức.

Ngoài ra, nếu bạn quên cài chương trình đã sửa đổi về thiết lập ban đầu, bạn có thể gây ra lỗi khác.

Sử dụng "**Kiểm tra thiết bị theo điều kiện thực hiện**" trong trường hợp này. (Chỉ dành cho sê-ri MELSEC-Q và MELSEC-L)

Chức năng này sẽ thay đổi các giá trị thiết bị khi thực hiện số bước được chỉ định trước, mà không cần bạn phải sửa đổi chương trình.





## 3.5

## Sửa lỗi vận hành chương trình theo từng bước



Khi đang sửa lỗi, bạn có thể muốn xác nhận việc thực hiện chỉ lệnh trong từng bước hoặc kiểm tra các thay đổi đối với giá trị thiết bị. Tuy nhiên, sửa lỗi từng bước có thể khó thực hiện do tốc độ thực hiện chương trình (thời gian quét) nhanh.

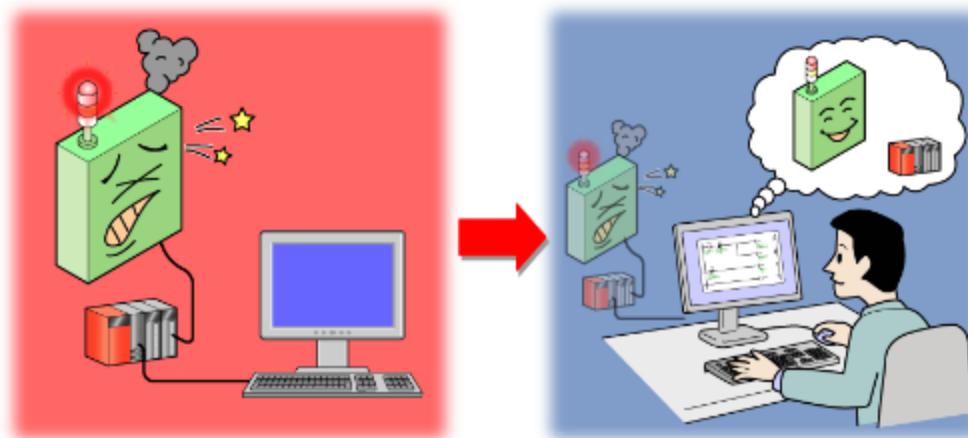
Sử dụng "**Chức năng thực hiện từng bước**" để giải quyết vấn đề này. (**Chỉ dành cho sê-ri MELSEC-Q và MELSEC-L**)

Chức năng này sẽ cho phép thực hiện chương trình từng bước một và do đó thực hiện sửa lỗi theo từng bước.

Sử dụng chức năng thực hiện từng bước cùng với **chức năng giả lập**. (Không thể sử dụng chức năng thực hiện từng bước để sửa lỗi trên một PLC thực).

Chức năng thực hiện từng bước có thể sử dụng các chức năng sau.

Chức năng	Mô tả
Thực hiện ngắt	Chức năng này thực hiện chương trình cho đến khi các điều kiện ngắt đã chỉ định được thỏa mãn. Dừng thực hiện chương trình khi các điều kiện ngắt được thỏa mãn. Sử dụng một điểm ngắt và thiết bị ngắt để chỉ định điều kiện ngắt.
Step execution (Thực hiện từng bước)	Chức năng này thực hiện chương trình theo từng bước.
Partial execution (Thực hiện một phần)	Chức năng này chỉ thực hiện chương trình từ vị trí đã chỉ định.



**3.5.1****Sử dụng chức năng thực hiện từng bước**

Chỉ định một **điểm ngắt** và **thiết bị ngắt** tương ứng làm vị trí bắt đầu sửa lỗi và điều kiện bắt đầu sửa lỗi. Bạn cũng có thể chỉ định **phạm vi bỏ qua** mà trong đó bạn muốn tạm thời tránh thực hiện chương trình. (Chi dành cho sê-ri MELSEC-Q và MELSEC-L)

Khi các điều kiện ngắt đã được thỏa mãn sau khi bắt đầu **thực hiện ngắt**, việc thực hiện chương trình sẽ bị gián đoạn. Do đó, trong khi thực hiện vận hành chương trình theo từng bước với **chức năng thực hiện từng bước**, hãy kiểm tra các thay đổi giá trị thiết bị để xác định lỗi.

**<Điểm ngắt>**

Cài điểm ngắt khi bạn muốn gián đoạn việc thực hiện chương trình.

Chỉ rõ điều này trong các đơn vị bước.

Có thể cài tối đa 64 điểm ngắt trong toàn bộ dự án.

**<Thiết bị ngắt>**

Cài thiết bị ngắt dựa trên thực hiện chương trình bạn muốn gián đoạn khi giá trị thiết bị

hoặc nhãn thỏa mãn điều kiện đã chỉ định.

Có thể cài tối đa 16 thiết bị bit và/hoặc từ.

**<Phạm vi bỏ qua>**

Cài phạm vi mà trong đó chương trình sẽ không được thực hiện, theo đơn vị của các khối lập trình PLC dạng thang (ladder), trong khi thực hiện từng bước.

Có thể chỉ định tối đa 16 phạm vi trong toàn bộ dự án.

**<Ứng dụng phạm vi bỏ qua>**

Có thể thu hẹp điểm lỗi bằng cách sử dụng **phạm vi bỏ qua**.

Thực hiện nghỉ khi cài và không cài phạm vi bỏ qua.

Nếu lỗi chỉ xảy ra khi thiết lập phạm vi bỏ qua được đưa ra, điều này thể hiện phạm vi đó có lỗi.

cp 1-GX\_Works2\_Advanced\_fod00146\_vie

3.5.1 Sử dụng chức năng thực hiện từng bước

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

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

Navigation

[PRG]Read MAIN (Read Only)

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
				Sửa lỗi đã hoàn tất. Hoàn thành chức năng thực hiện từng bước.
				Nhập  để tiếp tục.

Skip Range Break Point Break Device English Simple Q03UDE Host Station (52 NV)

## 3.6

## Mô phỏng hoạt động của thiết bị ngoại vi



Sửa lỗi có liên quan đến hoạt động của thiết bị ngoại vi là điều không khả thi trong môi trường phát triển không thể sử dụng thiết bị ngoại vi, như chức năng giả lập.

Để giải quyết vấn đề này, một chương trình sửa lỗi mô phỏng hoạt động của một thiết bị ngoại vi thường được thêm vào. Tuy nhiên, việc tạo chương trình giả lập không chỉ mất nhiều thời gian và công sức mà còn khiến bạn phải sửa đổi chương trình khi thay đổi hoạt động.

Sử dụng "**Thiết lập hệ thống I/O**" để giải quyết vấn đề này.

Chức năng này có thể mô phỏng hoạt động của một thiết bị ngoại vi mà không cần sử dụng chương trình sửa lỗi.

Có thể dễ dàng cài hoặc thay đổi hoạt động của thiết bị ngoại vi trong cửa sổ thiết lập. Do đó, không cần phải tạo/sửa đổi chương trình thông thường.

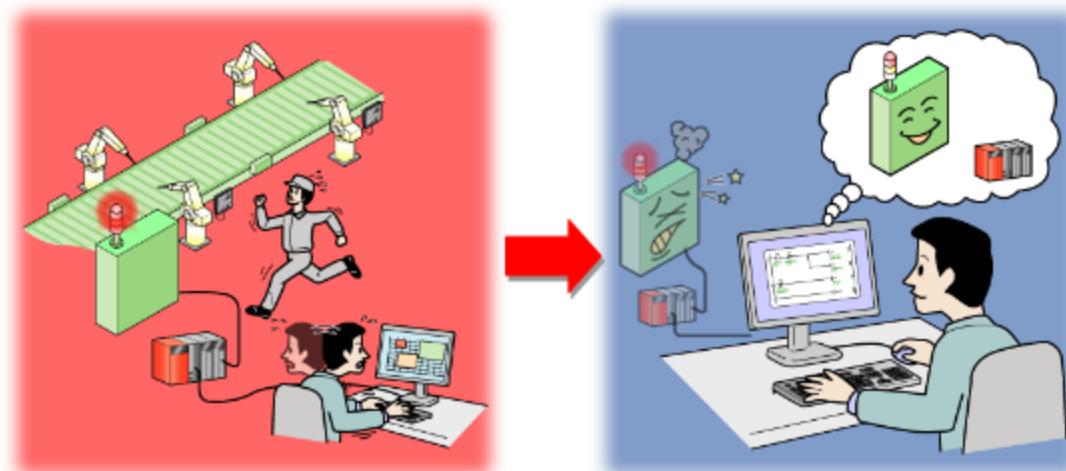
Có thể cài hoạt động của thiết bị ngoại vi theo hai cách sau đây:

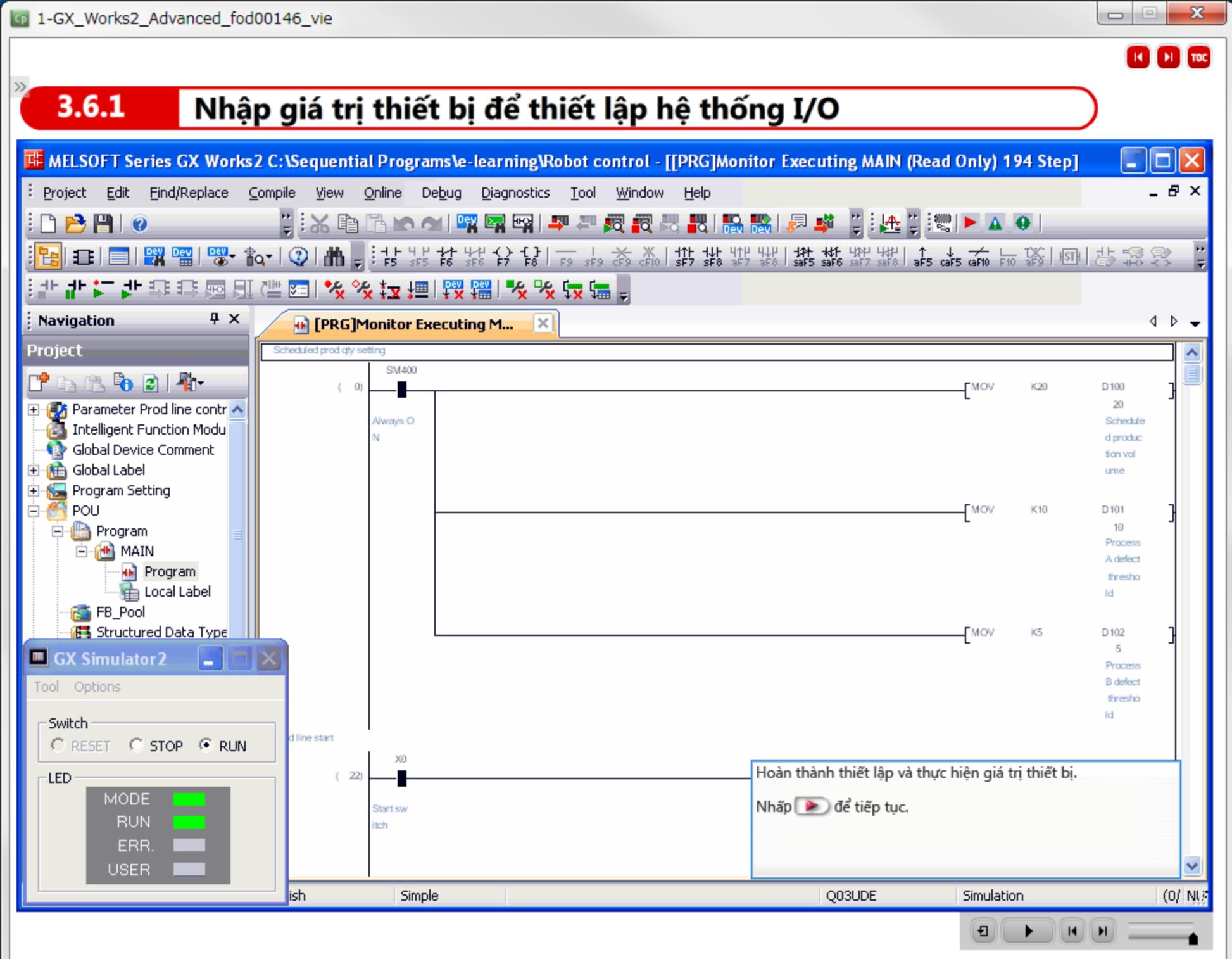
### <Thiết lập giá trị thiết bị>

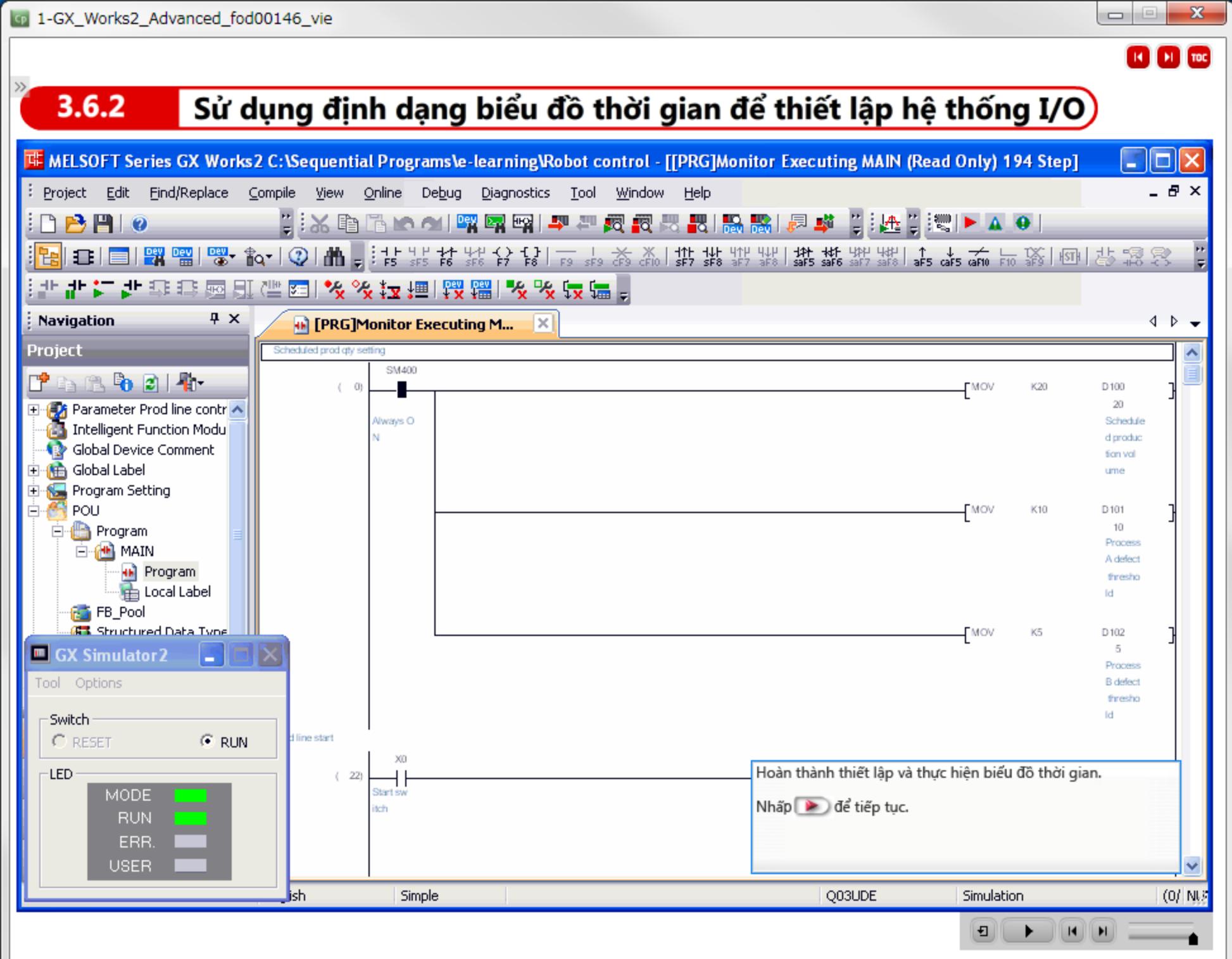
Có thể thay đổi giá trị thiết bị được chỉ định tại thời điểm cài bộ hẹn giờ sau khi thỏa mãn các điều kiện đã chỉ định.

### <Thiết lập ở định dạng biểu đồ thời gian>

Có thể điều khiển thay đổi thiết bị đã được cài ở định dạng biểu đồ thời gian đã chỉ định khi các điều kiện đã chỉ định được thỏa mãn.







## Chương 4 Quản lý dự án và Biện pháp bảo mật

### Các bước đào tạo trong Chương 4

Trong Chương 4, bạn sẽ tìm hiểu về các chức năng dành cho quản lý dự án và các biện pháp bảo mật.

- 4.1 Ngăn rò rỉ bí quyết và sửa đổi trái phép các chương trình
  - 4.1.1 Giới hạn dữ liệu có thể truy cập cho từng người dùng
- 4.2 Sao lưu dự án và quản lý phiên bản
- 4.3 So sánh các chương trình được lưu vào bộ điều khiển khả trình và máy tính cá nhân

## 4.1

## Ngăn rò rỉ bí quyết và sửa đổi trái phép các chương trình



Chương trình PLC có các bí quyết và dữ liệu quan trọng về mặt chiến lược.

Việc rò rỉ bí quyết và dữ liệu từ chương trình ra bên ngoài có thể có ảnh hưởng hủy hoại đến doanh nghiệp.

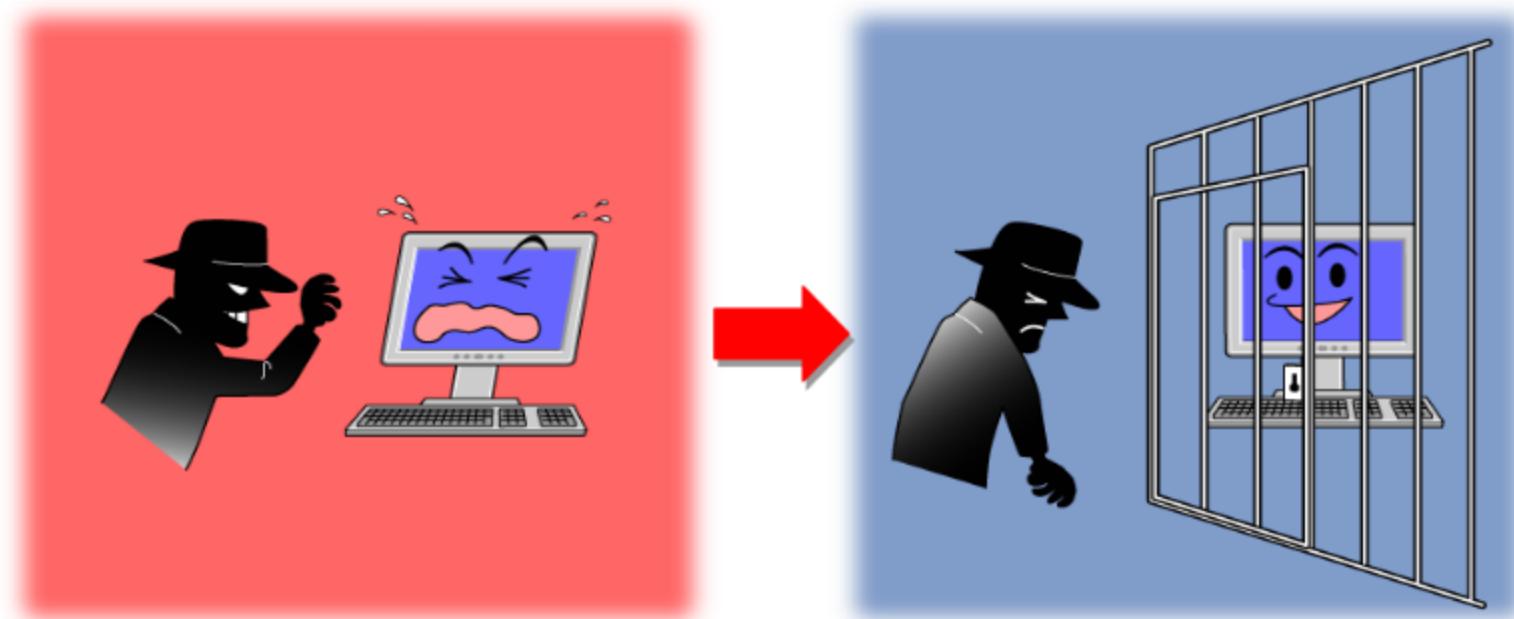
Sửa đổi trái phép chương trình có thể dẫn đến các sự cố sản xuất như làm dừng hệ thống.

Sử dụng "**Bảo mật**" để giải quyết các vấn đề này.

Chức năng này sẽ giới hạn số người dùng có thể truy cập từng dự án với sự bảo vệ bằng mật khẩu.

Chức năng này cũng có thể giới hạn phạm vi dữ liệu hoặc chức năng mà từng người dùng có thể truy cập hoặc vận hành.

Do đó, chức năng này sẽ ngăn chặn người dùng trái phép duyệt hoặc chỉnh sửa các chương trình.



**4.1.1****Giới hạn dữ liệu có thể truy cập cho từng người dùng**

Một chương trình PLC quy mô lớn thường là sản phẩm hợp tác của hai lập trình viên trở lên.

Trong trường hợp phát triển nhóm, phạm vi dữ liệu truy cập được và chức năng sẵn có phải được quản lý hợp lý theo phạm vi công việc và mức kỹ năng của từng lập trình viên và tính bảo mật của dữ liệu mà mỗi lập trình viên xử lý.

Có thể thực hiện công tác quản lý truy cập này bằng cách thiết lập **các cấp độ truy cập** bảo mật.

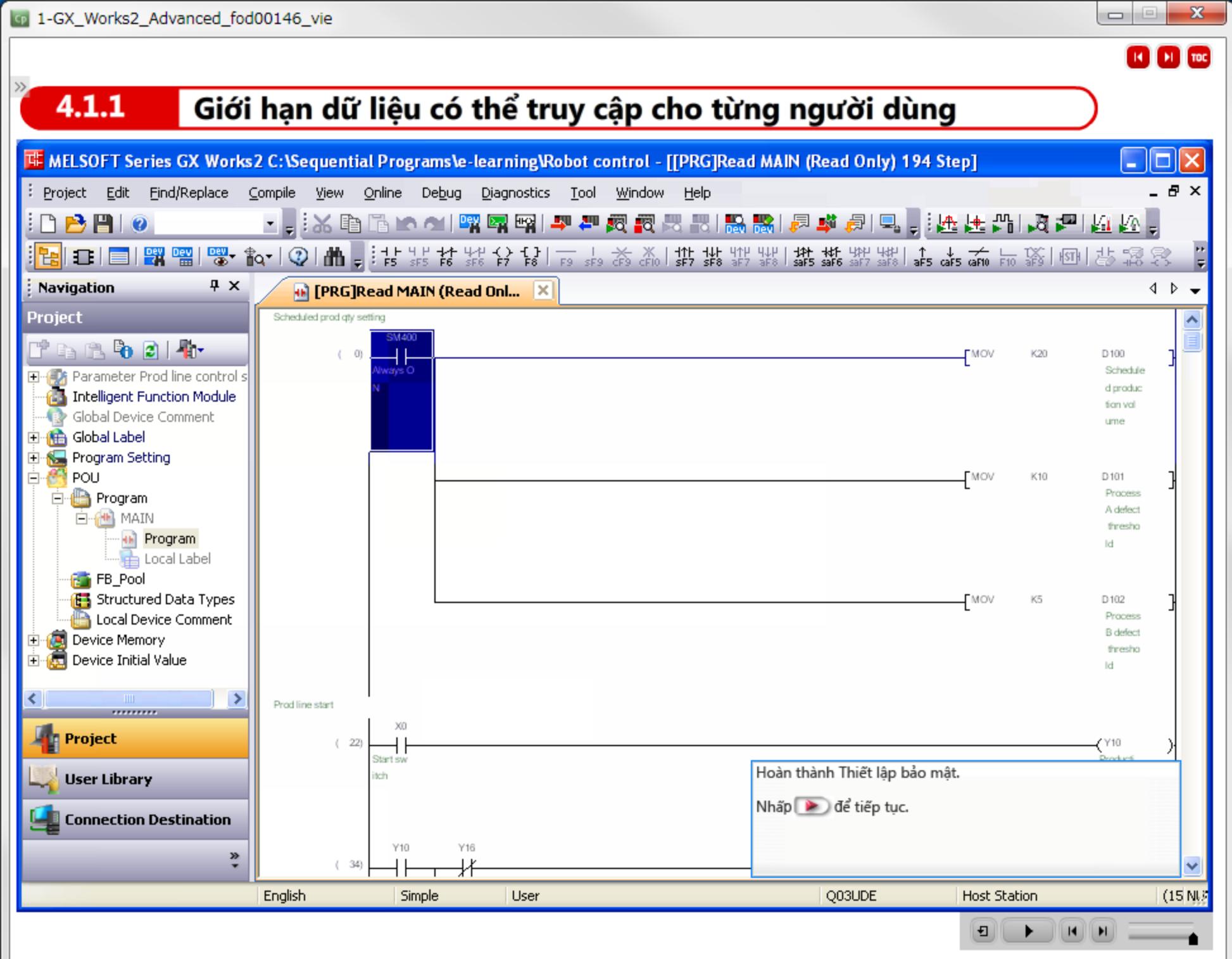
**<Cấp độ truy cập>**

Có thể cài đặc quyền vận hành cho dữ liệu có trong dự án cho từng người dùng.

Có thể cài năm cấp độ truy cập sau đây.

Cấp độ truy cập		Quyền vận hành	
Cao	Administrators (Quản trị viên)	Cấp quản trị viên	Được phép sử dụng tất cả các chức năng.
	Developers (Level3) (Nhà phát triển(Cấp độ3))	Cấp nhà phát triển	Thiết lập bảo mật, truy cập dữ liệu và một số vận hành bị hạn chế.
	Developers (Level2) (Nhà phát triển(Cấp độ2))		
	Developers (Level1) (Nhà phát triển(Cấp độ1))		
Thấp	Users (Người dùng)	Cấp người vận hành	Chỉ duyệt được dữ liệu dự án. Không đọc được từ CPU PLC.





## 4.2

## Sao lưu dự án và quản lý phiên bản



Một chương trình quan trọng có thể bị mất do lỗi CPU của PLC.

Nếu không có chương trình sao lưu, sẽ có rất ít hi vọng khôi phục nhanh.

Ngay cả khi có chương trình sao lưu, việc xác định được phiên bản giống với chương trình đã mất cũng không phải là đơn giản nếu công tác quản lý phiên bản không được đảm bảo.

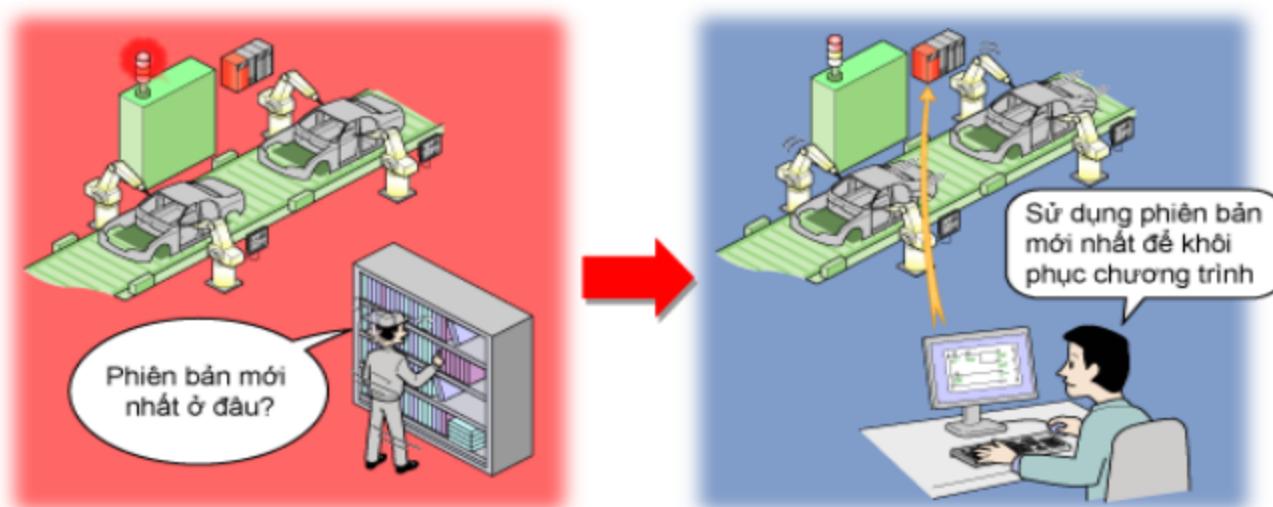
Để chuẩn bị cho các tình huống ngoài mong đợi, cần phải thực hiện sao lưu định kỳ và đảm bảo công tác quản lý phiên bản.

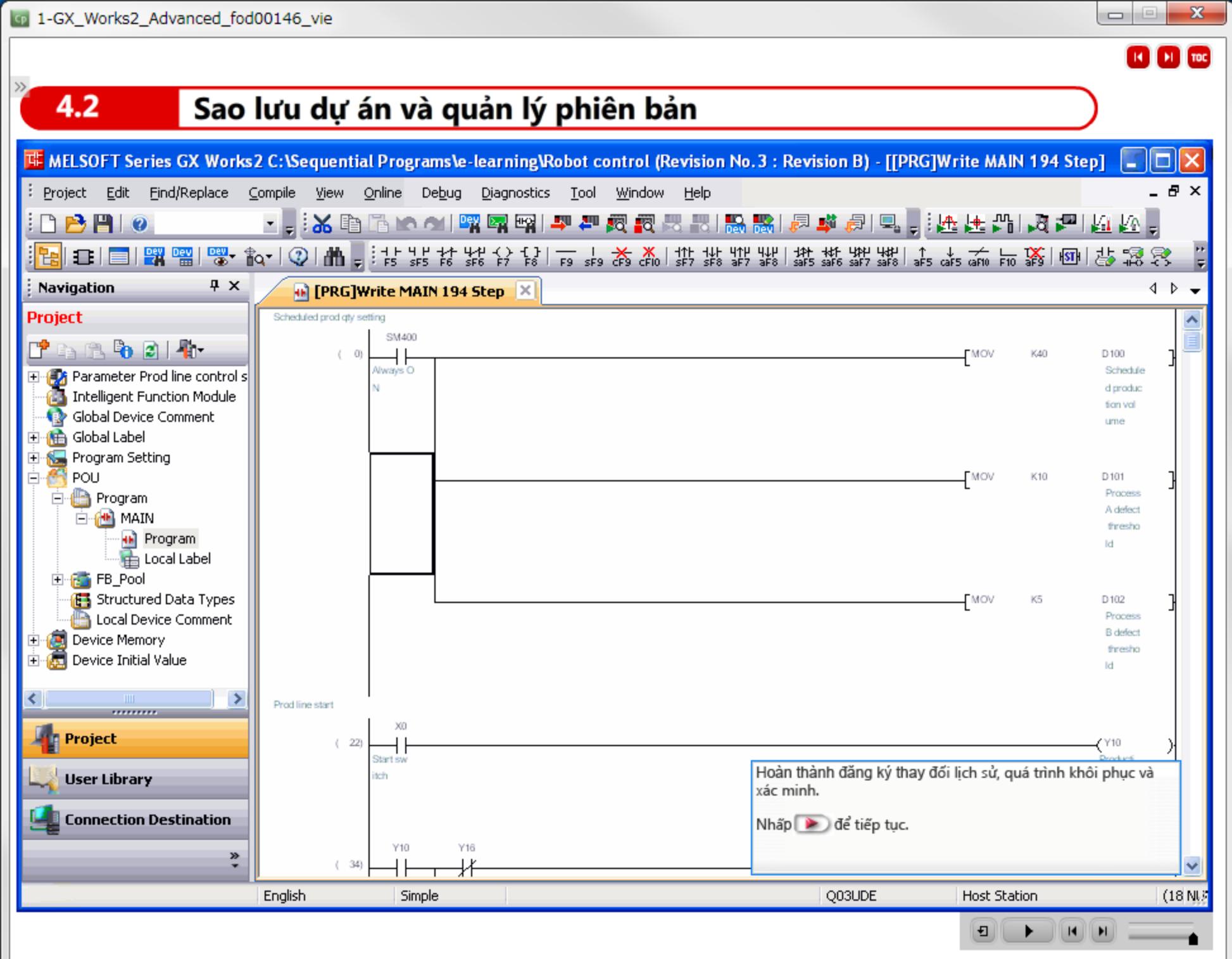
Sử dụng "**Lịch sử thay đổi**" để giải quyết các vấn đề này.

Chức năng này có thể ghi lại tối đa 100 bộ lịch sử thay đổi (số, ngày/giờ, người dùng, tiêu đề, chú thích trong lịch sử) của dự án.

Dữ liệu dự án tại thời điểm ghi cũng được sao lưu cùng thời điểm đó.

Quản lý phiên bản được bảo đảm bằng chức năng lịch sử thay đổi sẽ cho phép bạn khôi phục các chương trình bị mất, xác thực các phiên bản của chương trình và do đó thực hiện khôi phục nhanh trong các trường hợp không mong đợi.





## 4.3 So sánh các chương trình được lưu vào bộ điều khiển khả trình và máy tính cá nhân

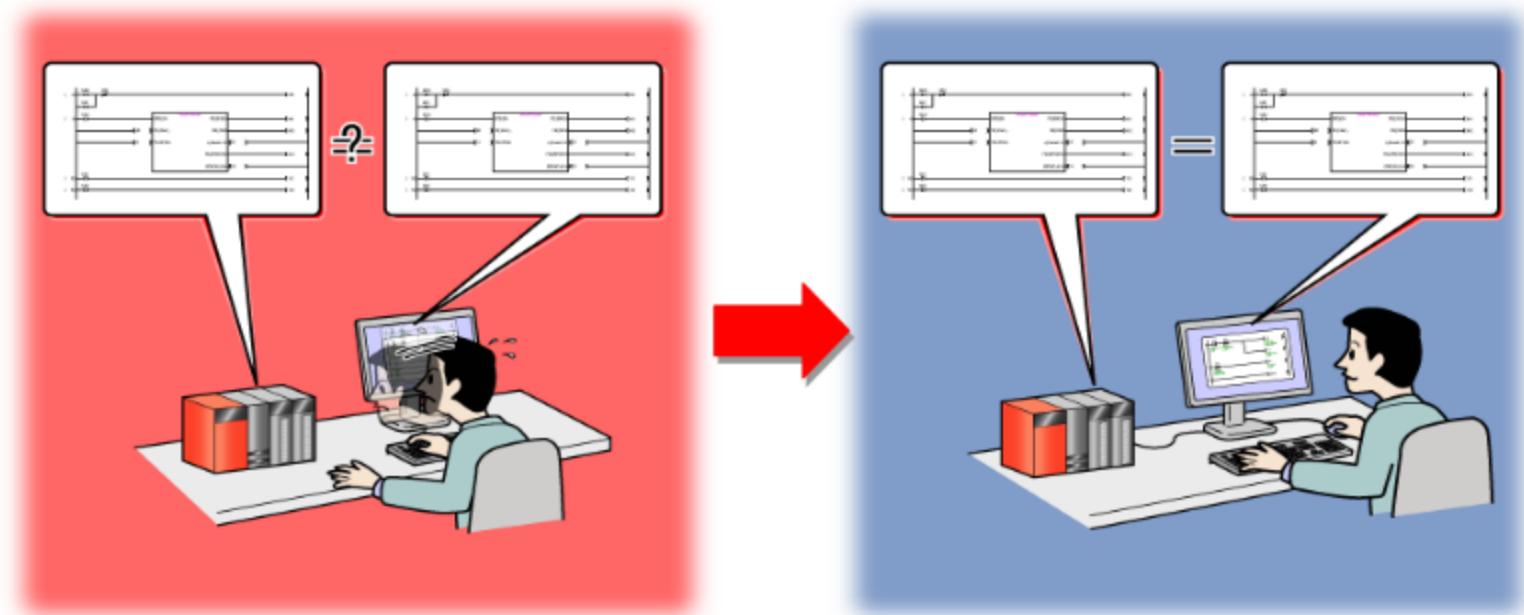


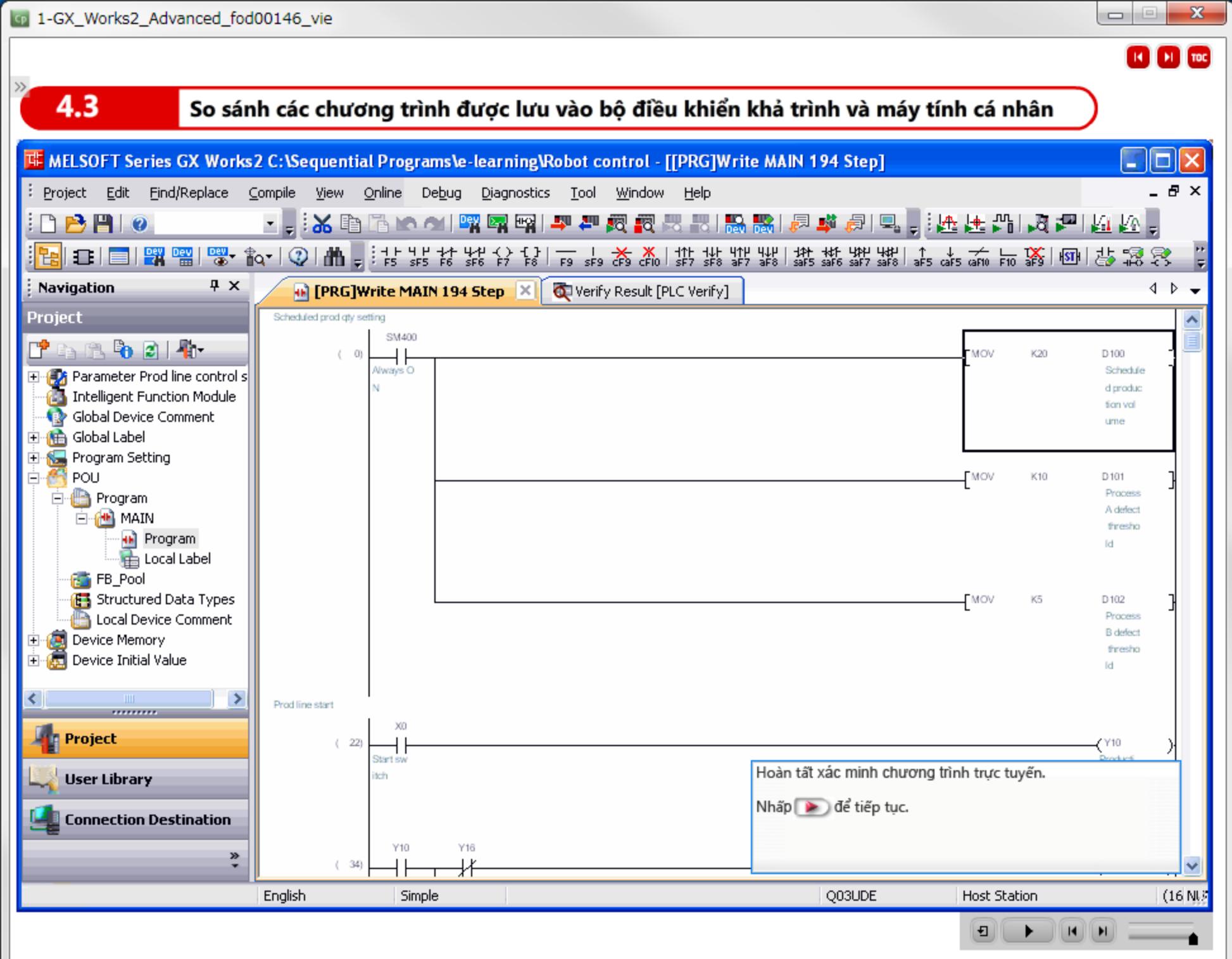
Thông thường, các chương trình được lưu vào một máy tính trong môi trường phát triển và cũng được ghi vào PLC. Hai chương trình này không phải lúc nào cũng giống nhau.

Chỉ thực hiện kiểm tra trực quan để xem hai chương trình có giống nhau hay không có thể dẫn đến lỗi.

Sử dụng "**Xác thực với PLC**" để giải quyết vấn đề này.

Chức năng này có thể xác thực rằng chương trình mở bằng GX Works2 có khớp với chương trình được ghi vào PLC hay không.





## Kiểm tra **Bài kiểm tra cuối khóa**

Bây giờ, bạn đã hoàn thành tất cả các bài học của Khóa học về PLC **Khóa học nâng cao về GX Work2**, bạn đã sẵn sàng tham gia bài kiểm tra cuối khóa. Nếu bạn không rõ về bất cứ chủ đề nào được trình bày, vui lòng nhân cơ hội này xem xét lại các chủ đề đó.

**Có tổng cộng 8 câu hỏi (8 mục) trong Bài kiểm tra cuối khóa này.**

Bạn có thể làm bài kiểm tra cuối khóa nhiều lần tùy thích.

### Làm thế nào ghi điểm bài kiểm tra

Sau khi chọn câu trả lời, hãy chắc chắn đã nhấp vào nút **Trả lời**. Câu trả lời của bạn sẽ bị mất nếu bạn tiếp tục mà không nhấp vào nút Trả lời. (Coi như là câu hỏi chưa được trả lời).

### Kết quả điểm số

Số lượng câu trả lời đúng, số lượng câu hỏi, tỷ lệ câu trả lời đúng, và kết quả đạt/hỗng sẽ xuất hiện trên trang điểm số.

Câu trả lời đúng: 2

Tổng số câu hỏi: 9

Tỷ lệ phần trăm: 22%

Để vượt qua bài kiểm tra, bạn phải trả lời đúng **60%** các câu hỏi.

**Tiếp tục**

**Xem lại**

**Thư lại**

- Nhấp vào nút **Tiếp tục** để thoát khỏi bài kiểm tra.
- Nhấp vào nút **Xem lại** để xem lại bài kiểm tra. (Kiểm tra câu trả lời đúng)
- Nhấp vào nút **Thư lại** để làm lại bài kiểm tra một lần nữa.

&gt;&gt;

## Kiểm tra Bài kiểm tra cuối khóa 1

Chức năng nào sau đây cho phép lập trình hiệu quả bằng cách sử dụng các khối lập trình PLC dạng thang (ladder) nhiều lần dưới dạng các bộ phận có thể chia sẻ? (Chọn một đáp án).

- Văn bản nội tuyến có cấu trúc
- Nhãn
- Khối chức năng

[Trả lời](#)[Quay lại](#)

## Kiểm tra Bài kiểm tra cuối khóa 2

Chức năng nào sau đây có thể tạo các chương trình dễ đọc bằng cách thay đổi tên thiết bị thành tên có liên quan đến ứng dụng của thiết bị đó? (Chọn một đáp án).

- Ghi chú thiết bị
- Nhãn
- Lưu ý

[Trả lời](#)[Quay lại](#)

## Kiểm tra Bài kiểm tra cuối khóa 3

Chức năng nào sau đây có thể tạo các chương trình để đọc bằng cách cung cấp thông tin về việc xử lý từng khôi lập trình PLC dạng thang (ladder)? (Chọn một đáp án).

- Ghi chú thiết bị
- Hướng dẫn dòng
- Lưu ý

[Trả lời](#)[Quay lại](#)

## Kiểm tra Bài kiểm tra cuối khóa 4

Câu nào sau đây giải thích đúng về chức năng “Verify with PLC” (Xác thực với PLC)? (Chọn một đáp án).

- So sánh chương trình được chỉnh sửa với một chương trình được ghi trong lịch sử thay đổi.
- So sánh chương trình được chỉnh sửa với một chương trình đã chọn được lưu vào máy tính.
- So sánh chương trình được chỉnh sửa với một chương trình được ghi vào CPU PLC.

Trả lời

Quay lại

## Kiểm tra Bài kiểm tra cuối khóa 5

Câu nào sau đây giải thích đúng về chức năng “I/O System Setting” (Thiết lập hệ thống I/O)? (Chọn một đáp án).

- Mô phỏng hoạt động của thiết bị I/O ngoại vi trên máy tính cá nhân trong khi sửa lỗi.
- Điều khiển từ xa hoạt động của thiết bị I/O ngoại vi từ máy tính cá nhân trong khi sửa lỗi.
- Mô phỏng hoạt động của CPU PLC trên máy tính cá nhân trong khi sửa lỗi.

Trả lời

Quay lại

## Kiểm tra Bài kiểm tra cuối khóa 6

Câu nào sau đây giải thích đúng về chức năng “Change history” (Lịch sử thay đổi)? (Chọn một đáp án).

- Ghi lại hoạt động của GX Works2 theo từng bước để có thể khôi phục thoải mái sau này.
- Ghi lại thông tin lịch sử và sao lưu dự án để cho phép xác thực và khôi phục sau này.

Trả lời

Quay lại

## Kiểm tra Bài kiểm tra cuối khóa 7

Chức năng nào sau đây có thể được sử dụng trong sửa lỗi để chỉ thay đổi giá trị thiết bị mà không sửa đổi chương trình? (Chọn một đáp án).

- Thực hiện ngắt
- Thiết lập hệ thống I/O
- Kiểm tra thiết bị theo điều kiện thực hiện

[Trả lời](#)[Quay lại](#)

## Kiểm tra Bài kiểm tra cuối khóa 8

Giả sử dự án có hai chương trình, A và B, và bạn sử dụng chức năng “label” (nhãn).  
Chương trình B có thể truy cập loại nhãn nào sau đây? (Chọn hai đáp án).

- Nhãn chung
- Nhãn cục bộ của chương trình A
- Nhãn cục bộ của chương trình B

[Trả lời](#)[Quay lại](#)

## Kiểm tra **Điểm kiểm tra**

Bạn đã hoàn thành Bài kiểm tra cuối khóa. Kết quả của bạn như sau.

Để kết thúc Bài kiểm tra cuối khóa, hãy tiếp tục với trang tiếp theo.

Câu trả lời đúng : **0**

Tổng số câu hỏi : **8**

Phần trăm : **0%**

[Tiếp tục](#)

[Xem lại](#)

[Thư lại](#)

**Bạn đã trượt bài kiểm tra.**

&gt;&gt;

Bạn đã hoàn thành Khóa học **PLC Khóa học nâng cao về GX Work2.**

Cảm ơn bạn đã tham gia khóa học này.

Chúng tôi hy vọng bạn thích các bài học và những thông tin bạn có  
được trong khóa học này sẽ hữu ích trong tương lai.

Bạn có thể xem lại khóa học này nhiều lần tùy ý.

Xem lại

Đóng