

PLC

Lập trình hiệu quả

Khóa học này mô tả việc sử dụng lập trình có cấu trúc để giúp cho việc lập trình hiệu quả hơn. Khóa học này dành cho những người đang lập trình bằng cách sử dụng sơ đồ bậc thang đơn giản.

Giới thiệu**Mục đích khóa học**

Thông qua khóa học này, bạn sẽ biết được những ưu điểm của việc lập trình có cấu trúc. Chúng tôi hy vọng khóa học này sẽ giúp bạn lập trình tốt hơn.

Khóa học sau đây là điều kiện tiên quyết trước khi tham gia khóa học này:

- Programming Basic (Kiến thức cơ bản về lập trình)

Giới thiệu**Cấu trúc khóa học**

Nội dung của khóa học này như sau.

Chương 1 - Tại sao cần lập trình có cấu trúc?

Chương này mô tả sự cần thiết của việc lập trình có cấu trúc để đem lại hiệu quả tốt hơn.

Chương 2 - Tại sao lập trình có cấu trúc lại hiệu quả

Chương này mô tả những lợi ích của việc lập trình có cấu trúc.

Bài kiểm tra cuối khóa

Điểm đạt: Bắt buộc phải đúng từ 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.

Giới thiệu **Thận trọng khi sử dụng**



Biện pháp phòng ngừa an toàn

Khi bạn học dựa trên việc 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. Khóa học này sử dụng phiên bản phần mềm sau:

- GX Works3 Phiên bản 1.032J

Chương 1 Tại sao cần lập trình có cấu trúc?



Chương này mô tả sự cần thiết của việc lập trình có cấu trúc để có kỹ thuật hiệu quả hơn.

- 1.1 Hiện trạng của ngành FA
- 1.2 Sự cần thiết đối với hiệu quả lập trình
- 1.3 Lập trình hiệu quả thông qua các chương trình có cấu trúc
- 1.4 Lập trình có cấu trúc với MELSOFT GX Works3

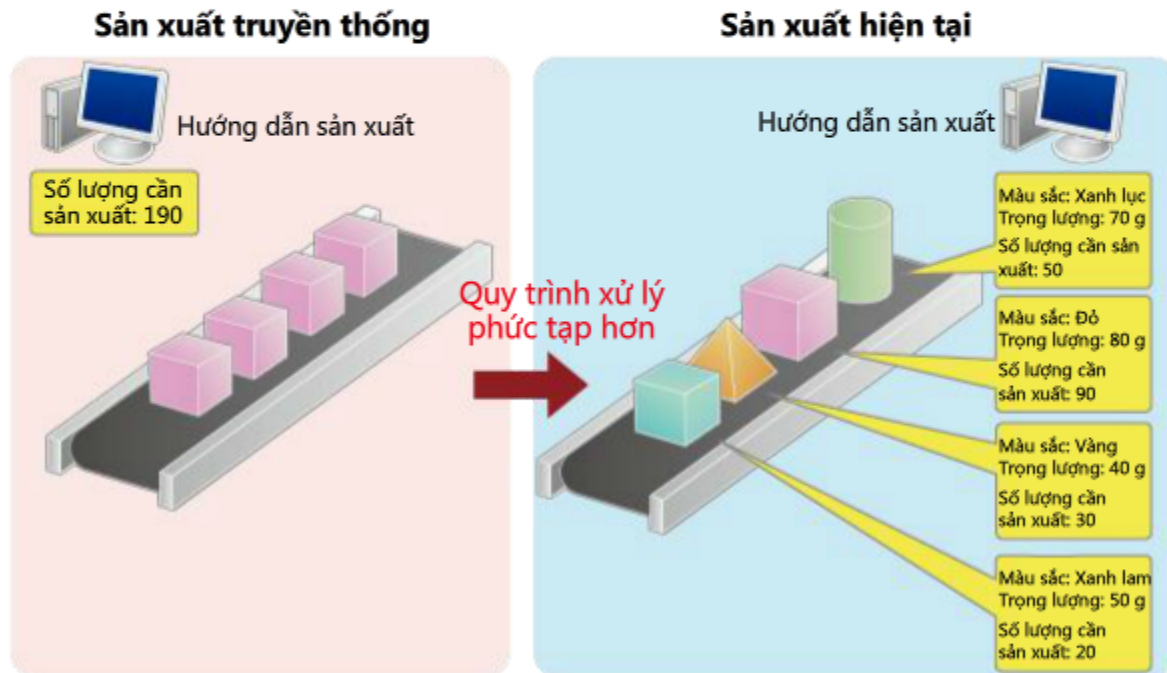
1.1

Hiện trạng của ngành FA

Trong ngành FA hiện nay, dữ liệu được xử lý bởi các thiết bị sản xuất đang gia tăng và việc xử lý ngày càng cao cấp và phức tạp hơn. Chi tiết được mô tả dưới đây.

1. Quy trình xử lý phức tạp hơn

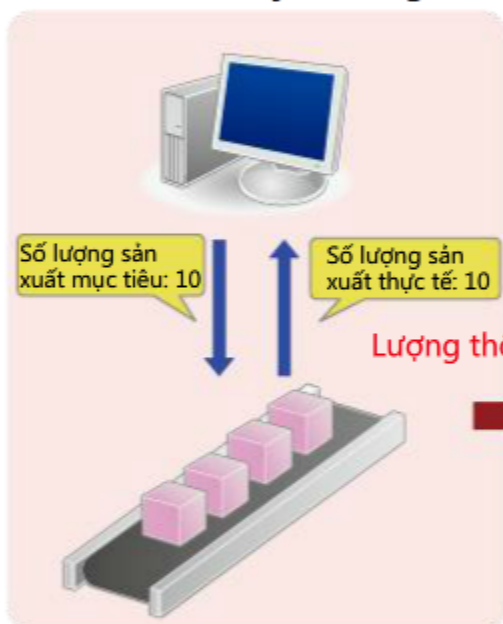
Trước đây, một sản phẩm duy nhất được sản xuất trên một dây chuyền sản xuất theo một hướng dẫn sản xuất. Tuy nhiên, việc sản xuất nhiều sản phẩm trên một dây chuyền sản xuất đang ngày càng trở thành xu hướng chính do sự đa dạng hóa sản phẩm và tinh giản hệ thống sản xuất. Do đó, việc xử lý trở nên ngày càng phức tạp.



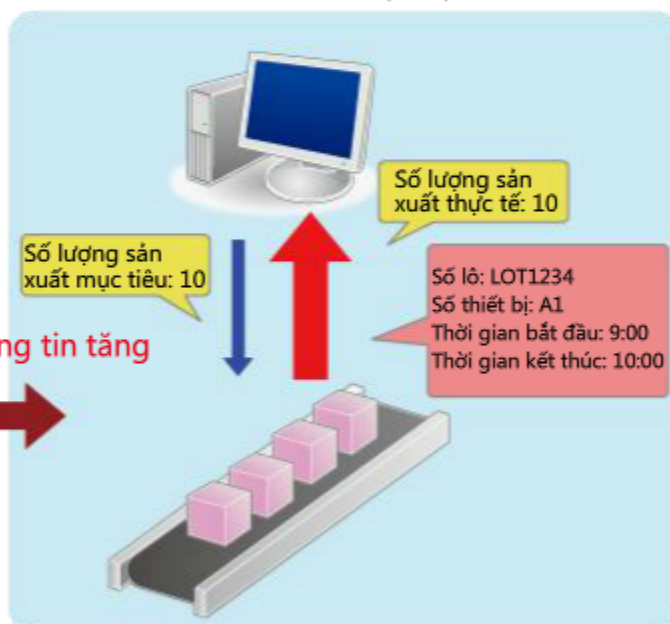
2. Tăng lượng thông tin

Trước đây, thông tin trao đổi giữa các máy tính cấp doanh nghiệp và thiết bị dây chuyền sản xuất được giới hạn trong các hướng dẫn công việc đơn giản và kết quả sản xuất. Ngày nay, các bản ghi nhật ký sản xuất được đưa vào thông tin kiểm soát sản xuất để đảm bảo khả năng truy xuất. Như vậy, lượng thông tin được xử lý tại các cơ sở sản xuất đã tăng lên.

Sản xuất truyền thống



Sản xuất hiện tại



1.1 Hiện trạng của ngành FA

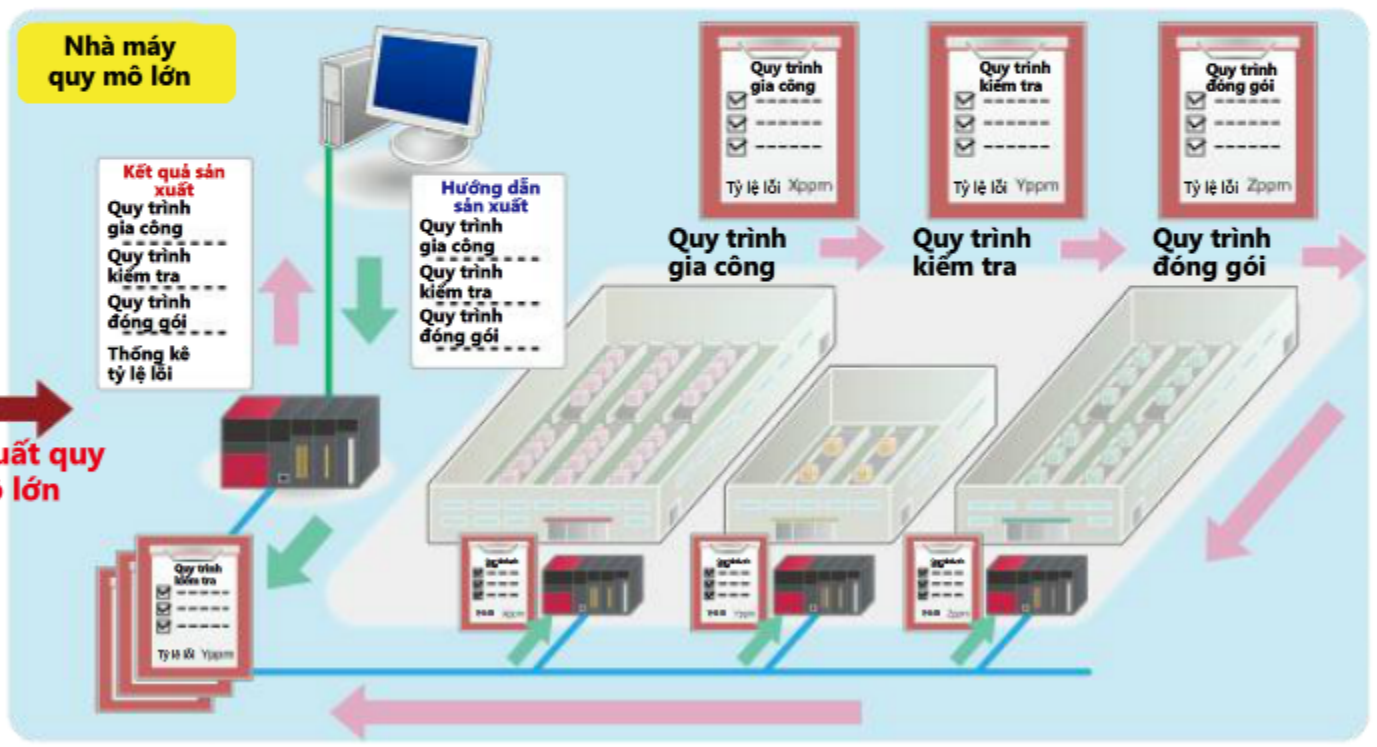
3. Cấu hình hệ thống quy mô lớn

Việc xây dựng và phát triển các nhà máy quy mô lớn ngày càng trở nên phổ biến. Tại các nhà máy quy mô lớn này, mỗi quy trình sản xuất đều được cấu hình mạng và các máy tính cấp doanh nghiệp sẽ thu thập thông tin của mỗi quy trình. Lượng thông tin được xử lý trong các môi trường này lớn hơn nhiều so với lượng thông tin trong các nhà máy quy mô nhỏ thông thường.

Sản xuất truyền thống



Sản xuất hiện tại



1.1

Hiện trạng của ngành FA

Khi thông tin được sử dụng trong môi trường FA tăng, các kiểu xử lý được thực hiện bởi bộ điều khiển có thể lập trình cũng thay đổi.

Trước đây, bộ điều khiển có thể lập trình được sử dụng chủ yếu để điều khiển trong khi sử dụng máy tính cá nhân để xử lý thông tin. Gần đây, cả dung lượng lưu trữ và tốc độ xử lý của bộ điều khiển có thể lập trình đều tăng lên. Bộ điều khiển có thể lập trình được thiết kế để sử dụng trong các môi trường FA và do đó đáng tin cậy hơn và mạnh mẽ hơn so với máy tính cá nhân. Bởi vậy, các nhà máy ngày càng có xu hướng dựa vào các bộ điều khiển này để xử lý thông tin.

Sản xuất truyền thống



Mở rộng khả năng xử lý của bộ điều khiển có thể lập trình

Điều khiển

Xử lý thông tin

Sản xuất hiện tại



Điều khiển

Xử lý thông tin

Khi bộ điều khiển có thể lập trình mở rộng, các chương trình điều khiển sẽ được sử dụng để xử lý phức tạp hơn bên cạnh việc điều khiển tuần tự và số lượng các bước cũng tăng lên. Một số ví dụ gồm có xử lý dữ liệu văn bản, thực hiện các hoạt động liên quan đến cơ sở dữ liệu và thực hiện kiểm soát mạng.

Trong các chương trình thông thường được tạo bằng sơ đồ bậc thang đơn giản, tất cả các chức năng xử lý được mô tả không có sự phân biệt rõ ràng. Các chương trình phức tạp với nhiều bước có những nhược điểm sau khi chúng được chỉnh sửa hoặc thêm tính năng mới sau này.

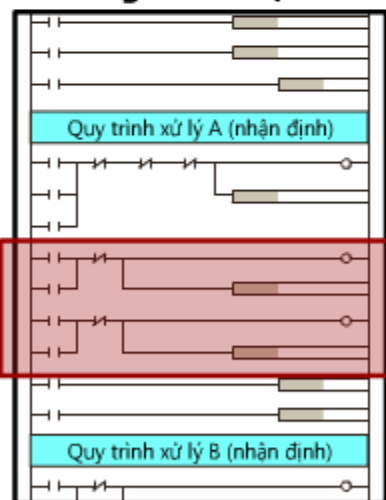
Việc sử dụng các chương trình hiện có không dễ



Xác định điểm bắt đầu hoặc kết thúc quy trình xử lý rất khó.

Phải tìm kiếm các phần có thể sử dụng được bằng cách tham khảo các ý kiến và nhận định.

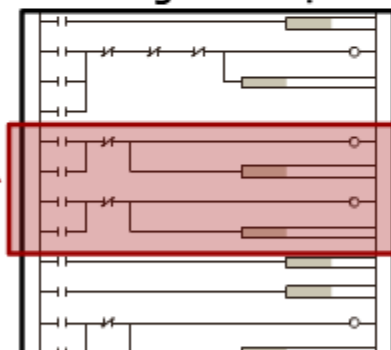
Chương trình hiện có



Bắt đầu quy trình

Kết thúc quy trình

Chương trình mới sử dụng các chương trình hiện có



Nhấp vào ở phía trên bên phải để bật lại hình động từ đầu.

Khi phát triển chương trình, cần lập kế hoạch hiệu quả như sử dụng chương trình tương tự và chuẩn hóa/chia sẻ các chương trình thường xuyên được sử dụng để giảm thời gian thiết kế để phát triển các chương trình mới.

1.3 Lập trình hiệu quả thông qua các chương trình có cấu trúc

Những hạn chế liên quan đến các chương trình bậc thang thông thường, đơn giản là chúng rất khó hiểu và khó sử dụng. Các chương trình có cấu trúc có thể cải thiện hiệu quả lập trình và loại bỏ những vấn đề này.

Hình động sau đây mô tả khái niệm về việc chia chương trình thành các phần nhỏ.

Có thể sử dụng các chương trình bằng cách chọn từng phần nhỏ.




Chương trình hiện có



Chỉ cần chọn phần chương trình

Chương trình mới sử dụng các chương trình hiện có



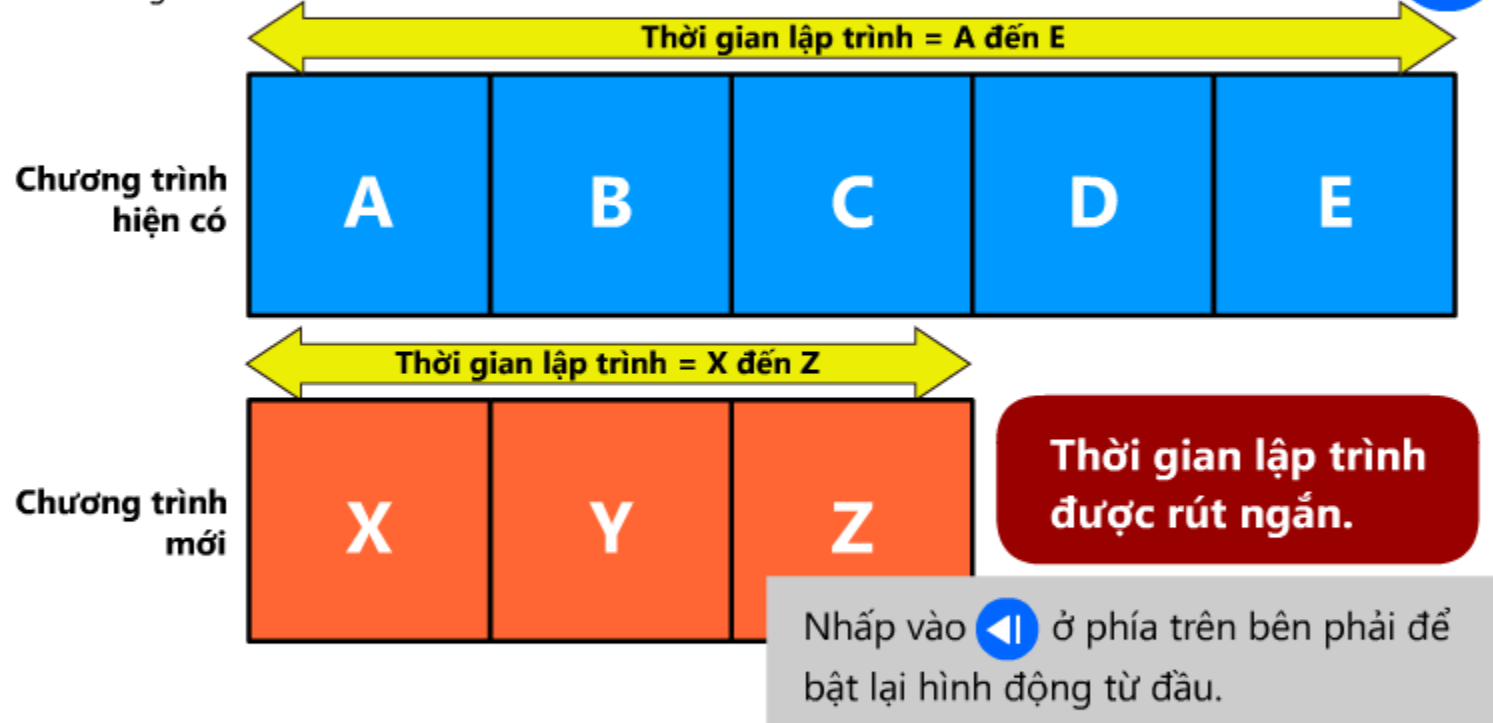
Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

Việc sử dụng chương trình sẽ cải thiện hiệu quả lập trình. Chi tiết được mô tả trong các phần tiếp theo.

1.3.1 Lợi ích của việc sử dụng chương trình

Việc sử dụng càng nhiều các chương trình hiện có sẽ càng làm giảm thời gian lập trình.

Chương trình mới bao gồm quy trình xử lý được trình bày là B và E trong chương trình hiện tại. Có thể sử dụng quy trình xử lý này, vì vậy thời gian lập trình từ X đến Z chỉ cần thiết khi tạo chương trình mới.



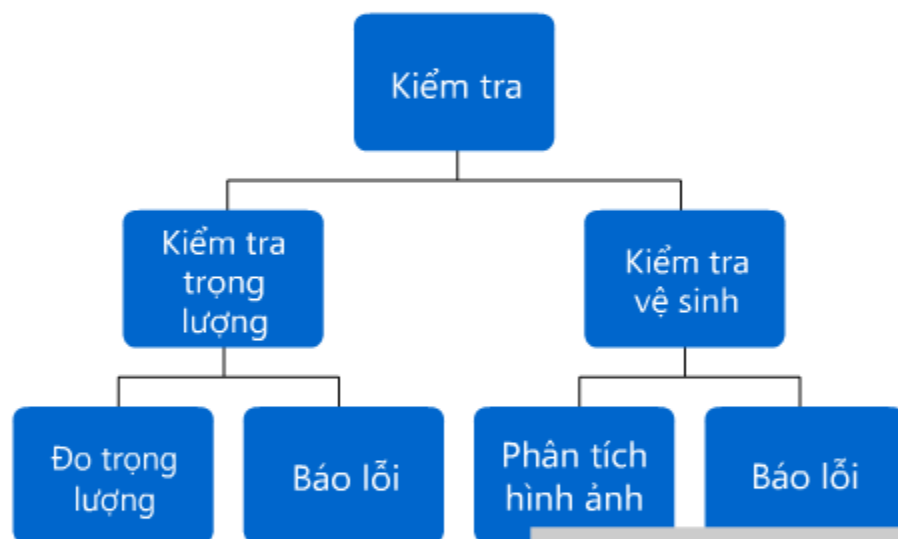
Bằng cách này, việc sử dụng các chương trình hiện có sẽ loại bỏ việc lập trình các chương trình mới, giảm thời gian lập trình tổng thể. Tối đa hóa việc sử dụng các chương trình hiện tại sẽ làm tăng đáng kể hiệu quả lập trình.


Khái niệm cơ bản phía sau việc tạo các chương trình có cấu trúc được mô tả trong các phần tiếp theo.

1.3.2 Lựa chọn hiệu quả các phần có thể sử dụng được

Hình động sau đây mô tả khái niệm về việc tổ chức phân cấp. Sử dụng tổ chức phân cấp cho phép lựa chọn các phần có thể sử dụng hiệu quả hơn.

Việc phân lớp POU theo cách này cho phép xem các chương trình trong các khối nhỏ hơn và ở cấp độ cao hơn tại cùng một thời điểm, vì vậy có thể lựa chọn các phần có thể sử dụng hiệu quả hơn.



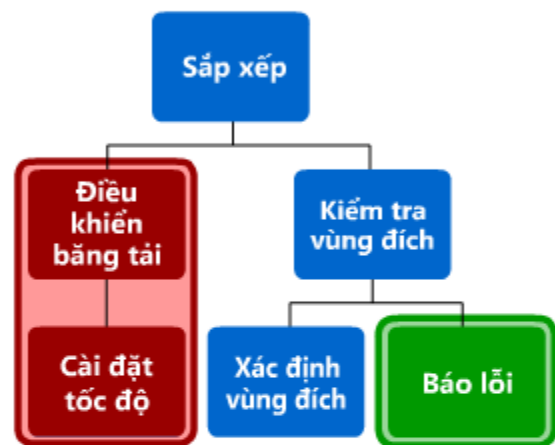
Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

1.3.2

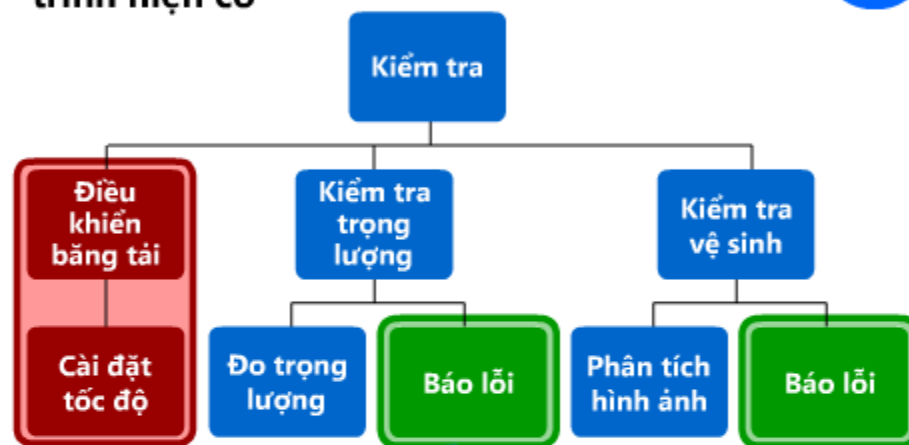
Lựa chọn hiệu quả các phần có thể sử dụng được

Ví dụ này minh họa mức độ hiệu quả của việc sử dụng các phần từ một chương trình được tổ chức phân cấp.


Chương trình hiện có



Chương trình mới sử dụng các chương trình hiện có



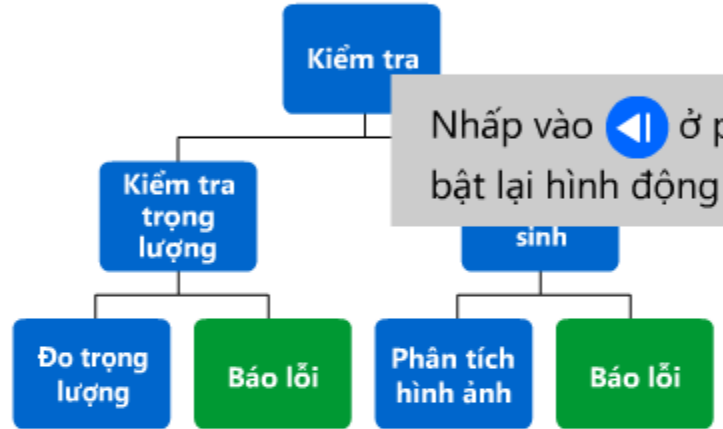
Sử dụng nhiều phần


Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

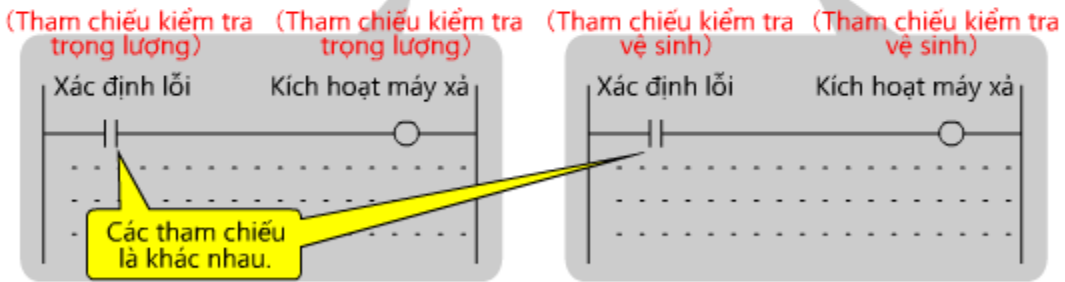
1.3.3 Sử dụng chương trình mà không cần phải phân bổ lại thiết bị

Bạn vừa tìm hiểu cách thức các chương trình có cấu trúc giúp bạn dễ dàng sử dụng các chương trình như thế nào. Với việc lập trình có cấu trúc, bạn được sử dụng thay cho thiết bị. Sử dụng nhãn cho phép lập trình viên tạo các chương trình mà không phải lo lắng về xung đột thiết bị với chương trình mới.

Không cần kiểm tra thiết bị nhiều lần.



Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.



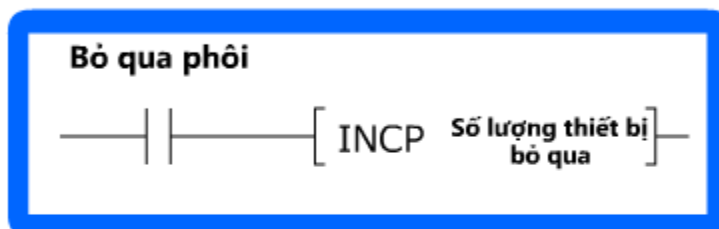
1.3.4

Lập trình sử dụng nhãn cho mục đích xử lý cụ thể


Sử dụng nhãn cho phép tạo các chương trình mà không cần phải Quan tâm các nhận xét và thiết bị.

Sử dụng nhãn cho phép tạo các chương trình có tên quy trình xử lý mà không cần phải lo lắng về thiết bị.

Sử dụng nhãn



Trực quan và dễ hiểu

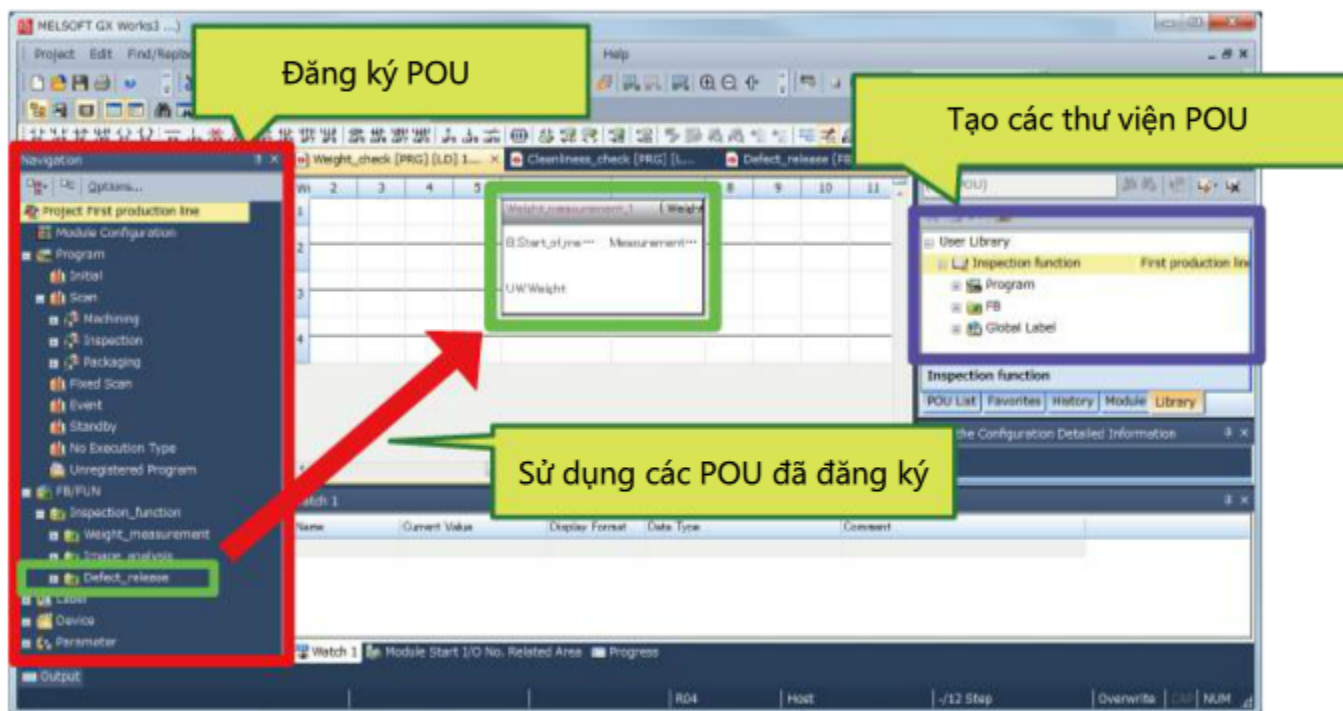
Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

1.4

Lập trình có cấu trúc với MELSOFT GX Works3

Phần mềm thiết kế MELSOFT GX Works3 hỗ trợ tạo các chương trình có cấu trúc.

MELSOFT GX Works3 giúp bạn dễ dàng sử dụng các chương trình bằng cách hiển thị các program organization unit (đơn vị tổ chức chương trình, POU) sẵn có để lựa chọn dễ dàng.



MELSOFT GX Works3

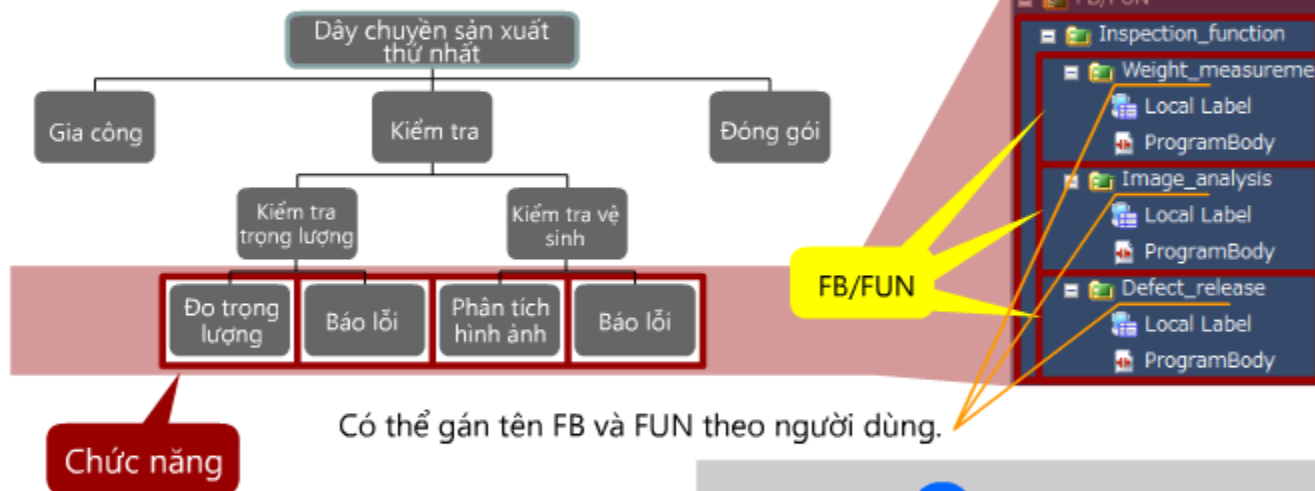
1.4.1


Việc tổ chức phân cấp các chương trình được trình bày như thế nào trong GX Works3

Hình động sau đây mô tả cách thức GX Works3 trình bày việc tổ chức phân cấp các chương trình bằng cách sử dụng một ví dụ về dây chuyền sản xuất.

Quy trình xử lý kiểm tra trọng lượng gồm có các chức năng đo trọng lượng và báo lỗi.

Các chức năng đo trọng lượng, phân tích hình ảnh và báo lỗi có trong quy trình xử lý được trình bày dưới dạng các cấp độ thấp hơn dưới cấp độ FB/FUN.



Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

1.4.2


Sử dụng các program organization unit (đơn vị tổ chức chương trình, POU)

Phần này giải thích quy trình sử dụng POU trong GX Works3. FB/FUN được chọn từ cửa sổ dự án và được sắp xếp trong chương trình.

The image shows the GX Works3 interface with the following components:

- Project Structure (Left):** A tree view showing the project hierarchy. Under 'Inspection', 'Weight_check' is selected. Under 'FB/FUN', 'Inspection_function' is expanded, and 'Weight_measurement' is selected.
- Ladder Logic Diagram (Top Center):** A hierarchical diagram showing 'Kiểm tra' (Check) branching into 'Kiểm tra trọng lượng' (Weight check) and 'Kiểm tra vệ sinh' (Cleanliness check). 'Kiểm tra trọng lượng' further branches into 'Đo trọng lượng' (Weight measurement) and 'Bảo lỗi' (Error handling). 'Kiểm tra vệ sinh' branches into 'Phân tích hình ảnh' (Image analysis) and 'Bảo lỗi' (Error handling).
- Ladder Logic Rung (Middle):** A ladder logic rung with 12 steps. A function block call 'Weight_measurement_1 (Weight_measurement)' is shown, with inputs 'B:Start_of_measurement' and 'UW:Weight', and output 'Measurement_result.UW'. The rung ends with 'END'.
- Function Block Call (Bottom):** A detailed view of the function block call showing the following logic:

Step	Logic
1	< Weight K100
2	MOV K1 Measurement_result
3	= Weight K100
4	> Weight K100

Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

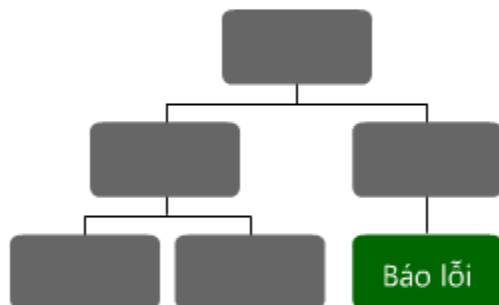
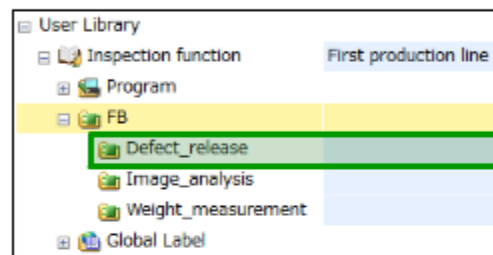
1.4.3 Sử dụng thư viện


Có thể chia sẻ POU giữa các dự án bằng cách sử dụng chức năng thư viện. Tạo thư viện các POU đã được kiểm chứng giúp đảm bảo chất lượng của chương trình.

Có thể sử dụng các POU chỉ bằng cách kéo và thả chúng vào chương trình.



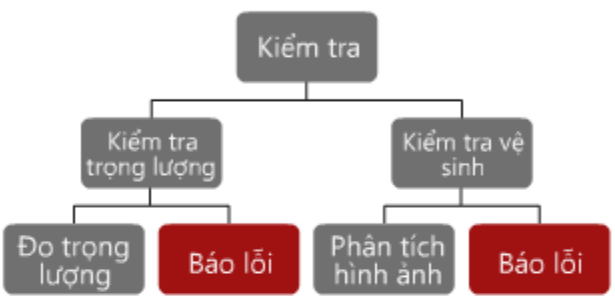
2	3	4	5	6	7
				Defect_release 2	(Defect_release)



Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.


1.4.3 Sử dụng thư viện hiệu quả

Việc tạo các thư viện POU được sử dụng thường xuyên sẽ đặc biệt hiệu quả. Trong các chương trình kiểm tra sau đây, FB/FUN "báo lỗi" là thích hợp nhất để đăng ký cho thư viện.



Các POU thường xuyên sử dụng được đăng ký.

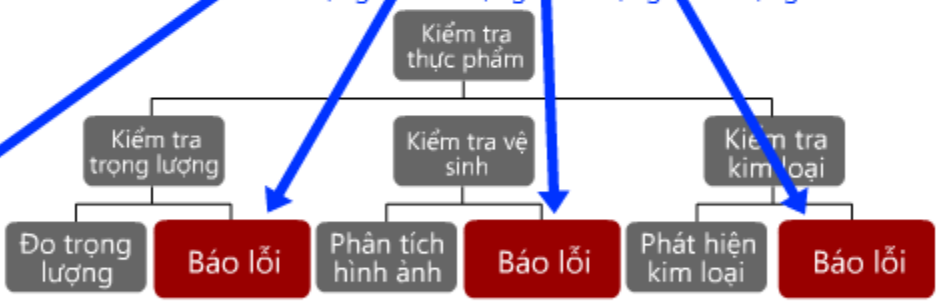


Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

Thiết bị A



Thiết bị B




Sử dụng Sử dụng Sử dụng Sử dụng

1.4.4 Sử dụng các POU được tạo sẵn

GX Works3 được cấu hình sẵn với các POU cho nhiều chương trình tiêu chuẩn. Các POU được tạo sẵn này bao gồm các FB mô-đun được tự động đăng ký phù hợp với cấu hình mô-đun và MELSOFT Library (thư viện MELSOFT).

MELSOFT Library do Mitsubishi Electric cung cấp có nhiều FB hữu ích sẵn có cho người dùng. FB giúp bạn sử dụng các mô-đun do Mitsubishi Electric sản xuất hoặc sản phẩm của các nhà sản xuất khác.



Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

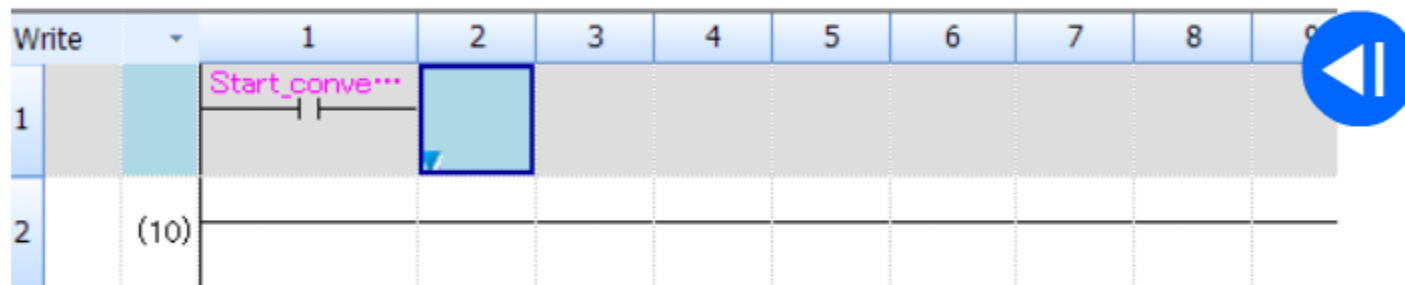
Để tìm hiểu về MELSOFT Library, vui lòng liên hệ với đại diện Mitsubishi Electric tại địa phương.


1.4.5 Nhập nhãn hiệu quả

Tính năng văn bản tiên đoán giúp bạn nhập nhãn hiệu quả hơn.

Khi bạn bắt đầu nhập nhãn, một danh sách nhãn đã đăng ký trùng với một số ký tự đầu tiên bạn đã nhập sẽ tự động xuất hiện.

Sau đó, bạn chỉ cần chọn để nhập nhãn mong muốn. Việc này cho phép bạn đặt cùng một nhãn ở nhiều vị trí mà không mắc lỗi khi nhập.



Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

Lập trình sử dụng nhãn đã đăng ký

1.5 Tổng kết

Nội dung của chương này là:

- Hiện trạng của ngành FA
- Những hạn chế của các chương trình sử dụng Bậc thang đơn giản
- Khái niệm và ưu điểm của các chương trình có cấu trúc
- Lập trình có cấu trúc với MELSOFT GX Works3

Những điểm quan trọng cần lưu ý:

Lợi ích của việc sử dụng chương trình	Sử dụng chương trình sẽ giúp giảm bớt khối lượng chương trình cần phát triển mới. Tối đa hóa việc sử dụng các chương trình hiện tại sẽ làm tăng đáng kể hiệu quả lập trình.
Chia chương trình thành các phần nhỏ	Có thể làm rõ các phần có thể sử dụng được bằng cách chia chương trình thành các phần nhỏ và gán tên có nghĩa cho các phần này.
Tổ chức phân cấp	Ngoài việc chia chương trình thành các phần nhỏ, việc sử dụng một tổ chức phân cấp cho các chương trình giúp cho việc lựa chọn các phần chương trình có thể sử dụng được trở nên dễ dàng và hiệu quả hơn.
Ưu điểm của việc sử dụng nhãn	<ul style="list-style-type: none"> • Sử dụng nhãn sẽ giúp bạn không cần phải kiểm tra xung đột thiết bị và phân bổ lại thiết bị, giúp cho việc sử dụng chương trình trở nên hiệu quả hơn • Việc lập trình sẽ trực quan hơn khi sử dụng các tên có nghĩa để trình bày quy trình xử lý cụ thể
Các chương trình có cấu trúc trong MELSOFT	<ul style="list-style-type: none"> • MELSOFT cho phép tạo các chương trình có cấu trúc trong một trình biên tập trực quan dễ hiểu • MELSOFT giúp cải thiện hiệu quả lập trình

Chương 2 Tại sao lập trình có cấu trúc lại hiệu quả

Chương này mô tả những lợi ích của việc lập trình có cấu trúc.

Quy trình lập trình sẽ được mô tả dưới dạng các giai đoạn sau.

- **Chỉnh sửa chương trình:** Tạo và chỉnh sửa chương trình
- **Thử nghiệm đánh giá:** Xác nhận rằng chương trình hoạt động chính xác
- **Tìm kiếm và sửa lỗi:** Xác định và sửa lỗi

2.1 Rút ngắn thời gian lập trình bằng cách sử dụng các chương trình

2.2 Loại bỏ quy trình phân bổ lại thiết bị

2.3 Tránh những thay đổi ngoài ý muốn

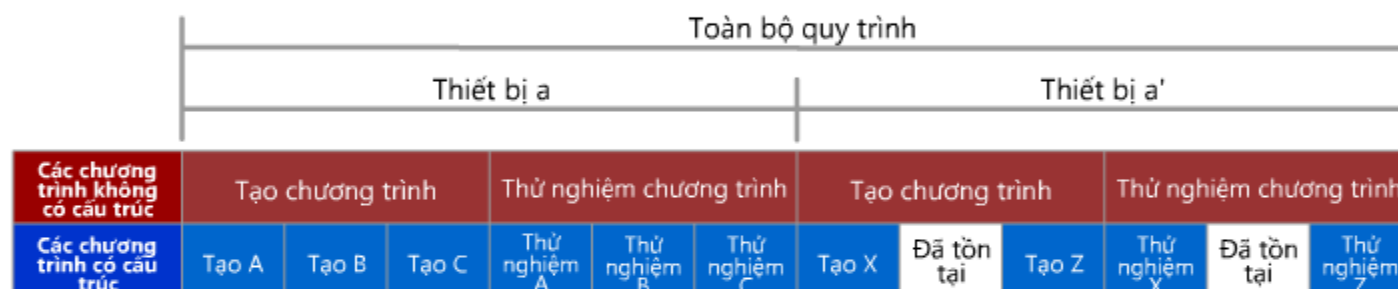
2.1 Rút ngắn thời gian lập trình bằng cách sử dụng các chương trình


Việc sử dụng các chương trình hiện có sẽ rút ngắn thời gian lập trình và cải thiện hiệu quả lập trình.

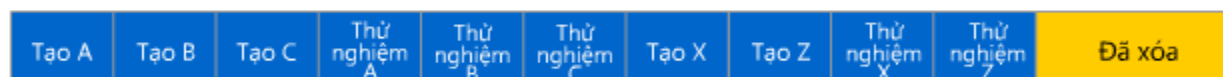
Lập trình hiệu quả bằng cách sử dụng các chương trình có cấu trúc



Quy trình lập trình



Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.



2.2

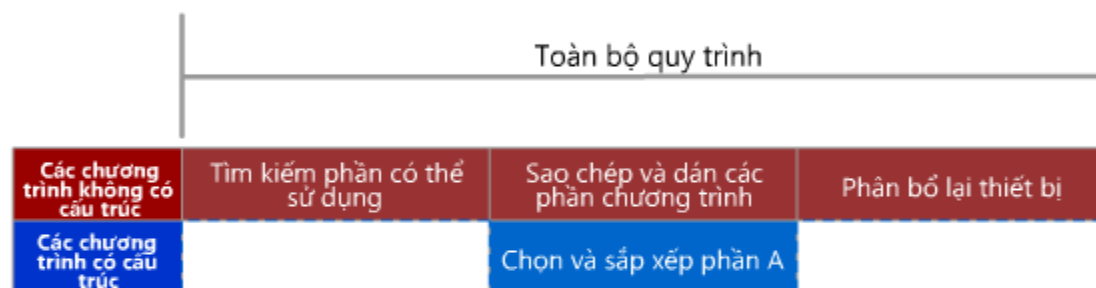
Loại bỏ quy trình phân bố lại thiết bị


Việc sử dụng các chương trình có cấu trúc sẽ giúp bạn không cần phải phân bố lại thiết bị trong chương trình mới.

Lập trình hiệu quả bằng cách sử dụng các chương trình có cấu trúc



Quy trình lập trình



Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.

Chọn và sắp xếp phần A

Đã xóa

2.3

Tránh những thay đổi ngoài ý muốn

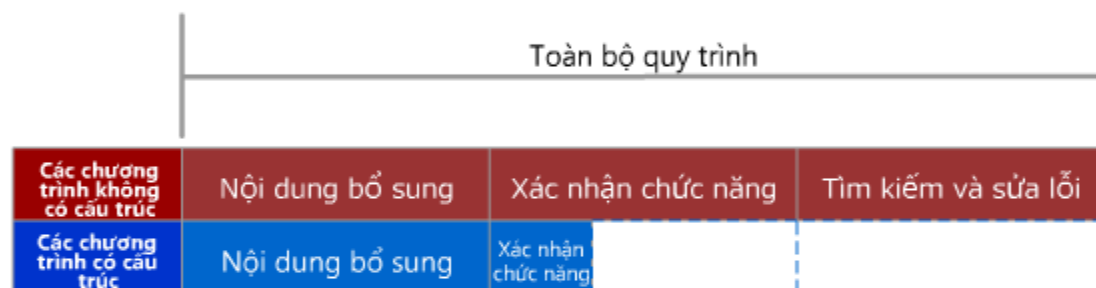



Thay đổi ngoài ý muốn có thể là do thêm chức năng cho các chương trình. Có thể tránh được những thay đổi này bằng cách sử dụng chương trình có cấu trúc.

Lập trình hiệu quả bằng cách sử dụng các chương trình có cấu trúc



Quy trình lập trình



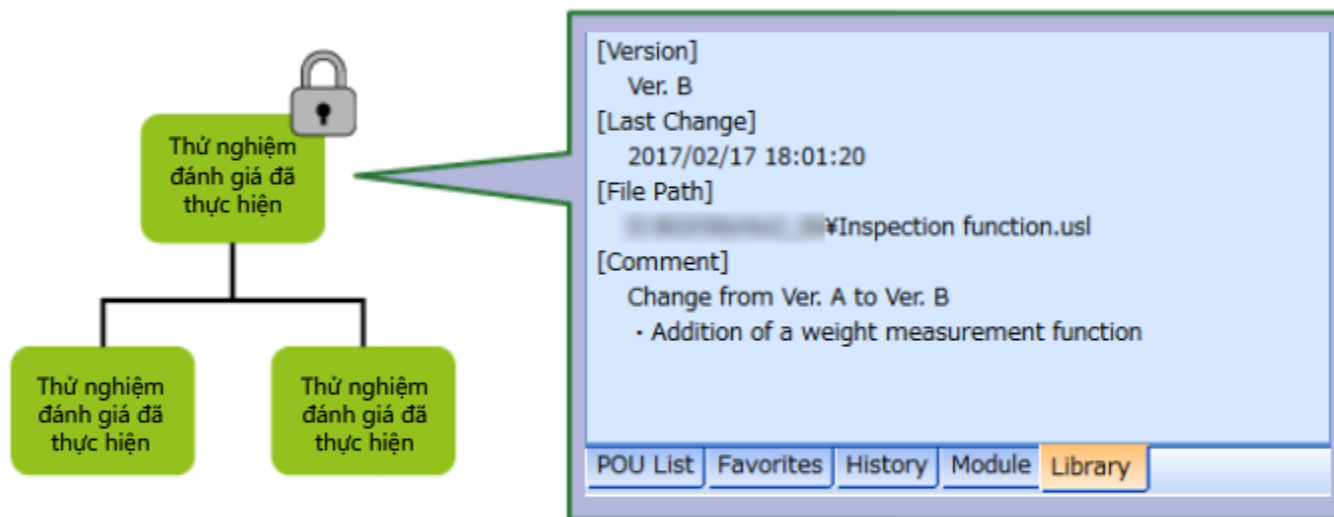
Nhấp vào  ở phía trên bên phải để bật lại hình động từ đầu.



2.3.1 Quản lý an toàn các POU

Có thể quản lý an toàn các POU bằng cách sử dụng MELSOFT GX Works3.

Có thể loại bỏ hoàn toàn những thay đổi ngoài ý muốn bằng cách khóa từng POU. Mỗi POU đều có thể có thông tin riêng bao gồm thông tin về phiên bản, ngày cập nhật và các nhận xét. Lịch sử thay đổi có thể được ghi trong các nhận xét.



Nội dung của chương này là:

- Lợi ích của việc sử dụng POU
- Lợi ích của việc sử dụng nhãn
- Lợi ích của việc sử dụng chương trình có cấu trúc

Những điểm quan trọng cần lưu ý:

Thử nghiệm đánh giá giảm	Sử dụng các chương trình có cấu trúc không chỉ giảm thời gian cần thiết để tạo chương trình, mà còn giảm thời gian cần thiết để thử nghiệm đánh giá.
Tránh những thay đổi ngoài ý muốn	Với các chương trình có cấu trúc, thay đổi được thực hiện cho từng POU, điều này giúp ngăn chặn những thay đổi ngoài ý muốn với các phần chương trình khác.
Khóa POU	Khóa POU khi hoàn thành thử nghiệm đánh giá đảm bảo việc quản lý an toàn các POU.

Khóa học này mô tả việc sử dụng lập trình có cấu trúc để giúp cho việc lập trình hiệu quả hơn. Bạn đã hoàn thành khóa học này.

Dưới đây là tổng kết những nội dung được đề cập trong khóa học này.

- Lập trình có cấu trúc là phương pháp tạo chương trình hiệu quả đang ngày càng trở nên phức tạp cùng với xu thế sản xuất của các hệ thống quy mô lớn và lượng dữ liệu tăng cao
- Các chương trình có cấu trúc dễ sử dụng hơn, giúp giảm bớt khối lượng chương trình cần phát triển mới và cải thiện hiệu quả lập trình
- Chia chương trình thành các phần nhỏ giúp phân chia các chương trình theo cách xử lý và theo chức năng. Sử dụng một tổ chức phân cấp và tên có nghĩa cho các phần của chương trình sẽ làm rõ các phần của chương trình có thể sử dụng được
- Sử dụng tổ chức phân cấp cho phép lựa chọn các phần có thể sử dụng một cách hiệu quả hơn
- Sử dụng nhãn cho phép các POU có thể sử dụng mà không phải lo lắng về xung đột thiết bị với chương trình mới
- MELSOFT GX Works3 hỗ trợ lập trình có cấu trúc và cải thiện hiệu quả lập trình
- Tạo thư viện cho phép chia sẻ/chuẩn hóa các phần chương trình được sử dụng thường xuyên, nhận biết được chất lượng đồng đều của các chương trình
- Ngoài việc sử dụng chương trình, có thể sử dụng các thư viện hiện có để giúp giảm bớt khối lượng cần phát triển mới

Khi thực sự triển khai việc lập trình có cấu trúc, hãy tham khảo các nội dung sau.

Lập trình với MELSOFT GX Works3

- Khóa học e-Learning "GX Works3 (Ladder)" (GX Works3 (Bậc thang))
- Cẩm nang hướng dẫn vận hành MELSOFT GX Works3

Cấu trúc và cú pháp chương trình

- Cẩm nang hướng dẫn lập trình cho MELSEC Dòng iQ-R

Bài kiểm tra Bài kiểm tra cuối khóa

Vì bạn đã hoàn thành tất cả các bài học của **PLC Lập trình hiệu quả**, 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 9 câu hỏi (10 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, bạn hãy chắc chắn đã nhấp vào nút **Câu trả lời**. Câu trả lời của bạn sẽ bị mất nếu bạn tiếp tục mà không nhấp vào nút Câu trả lời. (Coi như là câu hỏi chưa được câu 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/hỏng sẽ xuất hiện trên trang điểm số.

Câu trả lời đúng : **5**

Tổng số câu hỏi : **5**

Tỷ lệ phần trăm : **100%**

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

Tiếp tục

Xem lại

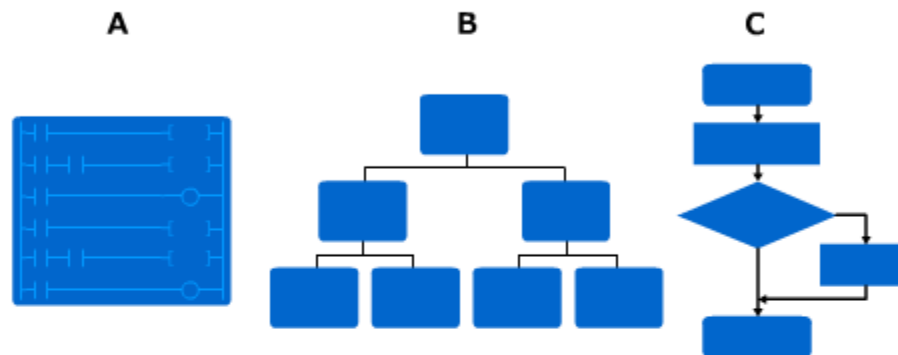
- Nhấp vào nút **Tiếp tục** để thoát khỏi bài kiểm tra.
- Nhấp vào nút **Xem lại** để xem lại bài kiểm tra. (Kiểm tra câu trả lời đúng)
- Nhấp vào nút **Thư' la,i** để làm lại bài kiểm tra một lần nữa.

Bài kiểm tra Bài kiểm tra cuối khóa 1

Các chương trình có cấu trúc

Chọn minh họa chính xác của các chương trình có cấu trúc.

- A
- B
- C



Câu trả lời

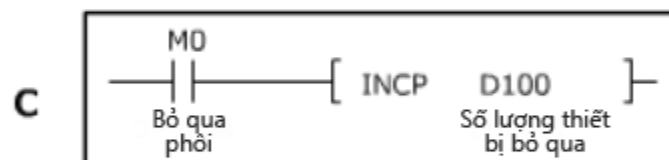
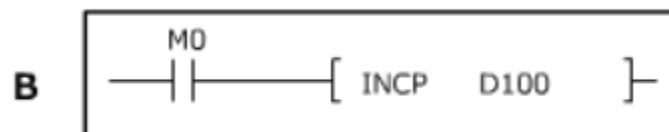
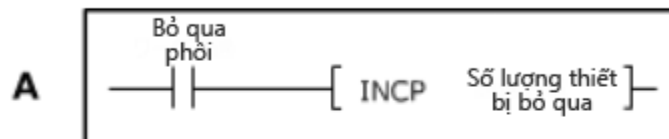
Quay lại

Bài kiểm tra **Bài kiểm tra cuối khóa 2**

Sử dụng nhãn

Chọn chương trình được tạo bằng nhãn.

- A
- B
- C



Câu trả lời

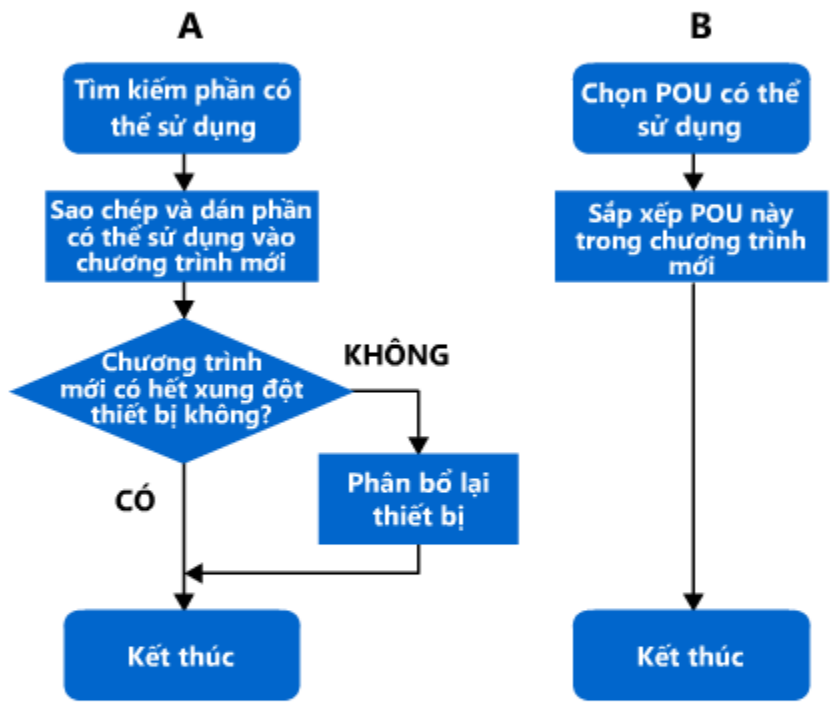
Quay lại

Bài kiểm tra Bài kiểm tra cuối khóa 3

Quy trình sử dụng chương trình

Chọn sơ đồ luồng minh họa chính xác quy trình sử dụng các chương trình có cấu trúc được tạo bằng nhãn.

- A
- B



Câu trả lời

Quay lại

Bài kiểm tra **Bài kiểm tra cuối khóa 4**

Mục đích của các chương trình có cấu trúc

Chọn mô tả chính xác về mục đích của các chương trình có cấu trúc. (Nhiều câu trả lời)

- Cải thiện tốc độ xử lý của chương trình
- Cải thiện hiệu quả lập trình
- Giảm thời gian biên dịch chương trình
- Cải thiện khả năng hiển thị của chương trình

Câu trả lời

Quay lại

Bài kiểm tra **Bài kiểm tra cuối khóa 5**



Lập trình có cấu trúc trong GX Works3

Chọn mô tả chính xác về việc tạo các chương trình có cấu trúc trong GX Works3.

- Các chương trình có cấu trúc có thể được tạo ngay sau khi cài đặt phần mềm.
- Tạo các chương trình có cấu trúc trong GX Works3 đòi hỏi phải mua một trình cắm.

Câu trả lời

Quay lại

Bài kiểm tra **Bài kiểm tra cuối khóa 6**

Chia chương trình thành các phần nhỏ.

Chọn tất cả các câu mô tả việc phân chia chương trình thành các phần nhỏ. (Nhiều câu trả lời)

- Chia chương trình thành các bước xác định trước, có số lượng bằng nhau
- Chia chương trình thành từng quy trình xử lý và chức năng
- Gán tên có nghĩa cho quy trình
- Sử dụng con trỏ để chuyển sang các phần khác
- Các đơn vị tổ chức chương trình (POU) đề cập đến kết quả phân chia chương trình thành các phần nhỏ

Câu trả lời

Quay lại

Bài kiểm tra **Bài kiểm tra cuối khóa 7****Ưu điểm của thư viện**

Chọn mô tả chính xác về ưu điểm của việc đăng ký chương trình vào các thư viện. (Nhiều câu trả lời)

- Đăng ký các POU được thường xuyên sử dụng sẽ giúp bạn sử dụng các chương trình hiệu quả
- Ngăn chặn sử dụng các POU
- Có thể chia sẻ POU giữa nhiều người
- Việc đăng ký và sử dụng lại các chương trình đã được tiêu chuẩn hóa giúp đảm bảo chất lượng chương trình nhất quán

[Câu trả lời](#)[Quay lại](#)

Bài kiểm tra **Bài kiểm tra cuối khóa 8**



Sử dụng các thư viện FB Mô-đun và MELSOFT Library

Chọn mô tả chính xác về việc sử dụng thư viện như các khối chức năng mô-đun và MELSOFT Library.

- Không cần phải xác minh hoạt động nội bộ khi sử dụng thư viện được tạo sẵn
- Phải tạo các FB Mô-đun phù hợp với kiểu mô-đun

Câu trả lời

Quay lại

Bài kiểm tra **Bài kiểm tra cuối khóa 9**

Các thư viện được cấu hình sẵn

Chọn tùy chọn trình bày thư viện do Mitsubishi Electric cung cấp.

H1 FB Mô-đun

H2 MELSOFT Library

H1

H2

[Câu trả lời](#)

[Quay lại](#)

Bài 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 tới trang tiếp theo.

Câu trả lời đúng : **9**

Tổng số câu hỏi : **9**

Tỷ lệ phần trăm : **100%**

Tiếp tục

Xem lại

Chúc mừng bạn. Bạn đã vượt qua bài kiểm tra.

Bạn đã hoàn thành khóa học **PLC Lập trình hiệu quả**.

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