

BỘ XÂY DỰNG

GIÁO TRÌNH LẬP TRÌNH PLC



NHÀ XUẤT BẢN XÂY DỰNG



BỘ XÂY DỰNG

GIÁO TRÌNH
LẬP TRÌNH
PLC

(Tái bản)

NHÀ XUẤT BẢN XÂY DỰNG
HÀ NỘI - 2013

LỜI NÓI ĐẦU

Sự tiến bộ trong công nghệ điện tử - tin học ngày nay thực sự là một cuộc cách mạng công nghệ, đã được ứng dụng vào lĩnh vực điều khiển tự động, đặc biệt là kỹ thuật vi xử lý. Hiện nay người ta đã sản xuất những thiết bị có kết cấu rất nhỏ gọn dạng máy tính mà bên trong có chứa bộ vi xử lý có thể lập trình được. Đó chính là thiết bị điều khiển lập trình (**Programmable Logic Controller**) viết tắt là **PLC** với nhiều ưu điểm nổi bật so với các bộ điều khiển cổ điển khác và được ứng dụng rộng rãi vào lĩnh vực điều khiển tự động.

Cuốn giáo trình này trình bày các kiến thức, kỹ năng liên quan đến bộ điều khiển PLC được trình bày theo chương trình khung Môđun **Lập trình PLC** Hệ Cao đẳng nghề Cơ điện tử. Giáo trình được trình bày tích hợp giữa lý thuyết và thực hành, giúp học viên có điều kiện nghiên cứu các kiến thức về lập trình logic và một số bài tập ứng dụng cơ bản, thực tế. Trên cơ sở đó rèn luyện các kiến thức, kỹ năng cơ bản để sau khi ra trường các học viên có khả năng áp dụng vào công việc thực tế.

Do hạn chế thời gian nên cuốn giáo trình không tránh khỏi những thiếu sót, chúng tôi xin được cảm ơn ý kiến đóng góp của các thầy cô giáo và các bạn đồng nghiệp... nhằm xây dựng cuốn giáo trình ngày càng hoàn thiện.

Nhóm tác giả

CHƯƠNG TRÌNH MÔĐUN ĐÀO TẠO LẬP TRÌNH PLC

Mã số môđun: MD25

Thời gian môđun: 90h; (Lý thuyết: 30h; Thực hành: 60h)

I. VỊ TRÍ, TÍNH CHẤT CỦA MÔĐUN

- Trước khi học môđun này cần hoàn thành các môđun và môn học cơ sở.
- Là môđun đào tạo bắt buộc.

II. MỤC TIÊU MÔĐUN

Sau khi hoàn tất môđun này, học viên có năng lực:

- Trình bày được nguyên lý hệ điều khiển lập trình PLC.
- Phân tích được phản ứng, nguyên tắc hoạt động của phần mềm hệ điều khiển lập trình PLC.
- Phương pháp kết nối dây giữa PLC và thiết bị ngoại vi.
- Thực hiện được một số bài toán đơn giản trong công nghiệp.
- Phân tích một số chương trình, phát hiện sai hỏng.

III. NỘI DUNG MÔĐUN

1. Nội dung tổng quát và phân bố thời gian

Số TT	Tên các bài trong môđun	Thời gian			
		Tổng số	Lý thuyết	Thực hành	Kiểm tra
1	Đại cương về điều khiển lập trình	15	5	10	
2	Các phép toán nhị phân của PLC	20	6	13	2
3	Các phép toán số của PLC	15	4	11	1
4	Lắp đặt mô hình điều khiển bằng PLC	40	15	28	4
	Cộng:	90	30	53	7

Ghi chú: Thời gian kiểm tra được tích hợp giữa lý thuyết và thực hành và được tính bằng giờ thực hành.

IV. ĐIỀU KIỆN THỰC HIỆN MÔĐUN

1. Vật liệu

- Dây nối, đầu cốt.
- Bàn gá, ống luồn dây.
- Nguồn điện các loại.

2. Dụng cụ thiết bị

- Các mô hình thực hành PLC S7-200 CPU 224.
- Board cắm linh kiện.
- Máy tính, máy chiếu.
- Mô hình thiết bị công nghiệp dùng PLC.

V. PHƯƠNG PHÁP VÀ NỘI DUNG ĐÁNH GIÁ

Áp dụng hình thức kiểm tra tích hợp giữa lý thuyết với thực hành. Các nội dung trọng tâm cần kiểm tra là:

1. Lý thuyết

- Các lệnh logic cơ bản, lệnh điều khiển bộ thời gian, bộ đếm.
- Các lệnh toán học, đồng hồ thời gian thực.
- Kết nối đầu vào/ra tên bản thí nghiệm.

2. Thực hành

- Lập trình các bài tập ứng dụng các lệnh logic.
- Lập trình được các bài tập ứng dụng trên mô hình.
- Sử dụng thành thạo phần mềm Microwin 4.0; S7-200 Simulator.

VI. HƯỚNG DẪN THỰC HIỆN MÔĐUN

1. Phạm vi áp dụng chương trình

Chương trình môđun này được sử dụng để giảng dạy cho trình độ Cao đẳng nghề.

2. Hướng dẫn một số điểm chính về phương pháp giảng dạy môđun

Trước khi giảng dạy, giáo viên cần căn cứ vào nội dung của từng bài học để chuẩn bị đầy đủ các điều kiện cần thiết nhằm đảm bảo chất lượng giảng dạy.

Nên áp dụng phương pháp đàm thoại để học viên ghi nhớ kỹ hơn.

Khi giải bài tập, làm các bài thực hành... Giáo viên hướng dẫn, thao tác mẫu và sửa sai tại chỗ cho học viên.

Nên sử dụng các mô hình, học cụ mô phỏng để minh họa các bài tập ứng dụng các hệ truyền động điện, các loại thiết bị điều khiển.

Bài 1

ĐẠI CƯƠNG VỀ ĐIỀU KHIỂN LẬP TRÌNH

MỤC TIÊU CỦA BÀI:

- Trình bày được cấu trúc và nhiệm vụ các khối chức năng của PLC.
- Thực hiện được sự kết nối giữa PLC và thiết bị ngoại vi.
- Mô tả được cấu trúc chương trình PLC.

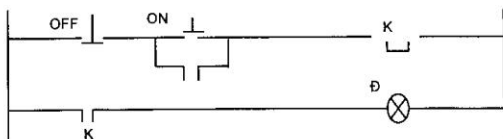
NỘI DUNG CỦA BÀI:

1. TỔNG QUAN VỀ ĐIỀU KHIỂN

1.1. Điều khiển nối cứng và điều khiển lập trình

1.1.1. Điều khiển nối cứng

Trong các bộ điều khiển nối cứng, các thành phần chuyển mạch như các role, contactor, các công tắc, đèn báo, động cơ, v.v... được nối cố định với nhau. Toàn bộ chức năng điều khiển, cách tiến hành chương trình được xác định qua cách thức nối các role, công tắc,... với nhau theo sơ đồ thiết kế. Khi muốn thay đổi lại hệ thống thì phải nối dây lại cho hệ thống điều khiển, nên đối với hệ thống phức tạp thì việc làm này đòi hỏi tốn nhiều thời gian và chi phí nên hiệu quả đem lại không cao.



Hình 1.1. Bộ điều khiển nối cứng đơn giản

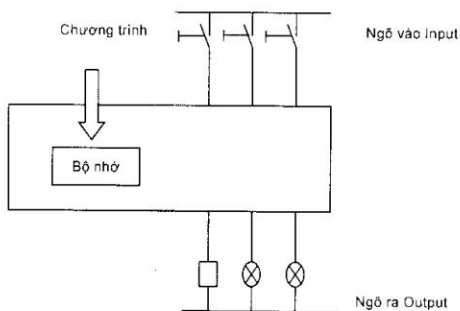
1.1.2. Điều khiển lập trình

Trong công nghiệp, các công nghệ khoa học kỹ thuật được ứng dụng vào sản xuất nên nhu cầu tự động hóa ngày càng tăng, đòi hỏi kỹ thuật điều khiển phải đáp ứng đủ các yêu cầu:

- Dễ dàng thay đổi chức năng điều khiển dựa trên các thiết bị cũ.
- Thiết bị điều khiển dễ dàng làm việc với các dữ liệu, số liệu.
- Kích thước vật lý gọn gàng, dễ bảo quản, dễ sửa chữa.
- Hoàn toàn tin cậy trong môi trường công nghiệp.

Hệ thống điều khiển để đáp ứng được các yêu cầu trên phải sử dụng bộ vi xử lý, bộ điều khiển lập trình, điều khiển qua các cổng giao tiếp với máy tính.

Bộ điều khiển logic khả trình PLC (*Programmable Logic Controller*) là loại thiết bị cho phép thực hiện linh hoạt các thuật toán điều khiển thông qua các ngôn ngữ lập trình. Với chương trình điều khiển của PLC đã tạo cho nó trở thành một bộ điều khiển số nhỏ gọn, dễ dàng thay đổi thuật toán, các số liệu và trao đổi thông tin với môi trường xung quanh.



Hình 1.2. Bộ điều khiển logic khả trình

Do những đặc điểm nổi bật của PLC trong điều khiển, nên ngày nay nó được sử dụng rất rộng rãi trong các giải pháp tự động hoá trong công nghiệp ở rất nhiều lĩnh vực:

- Điều khiển thang máy, thiết bị nâng, hạ hàng.
- Điều khiển các quy trình sản xuất: đóng gói bao bì, xi măng, bia v.v...
- Tự động hoá các hệ thống dịch vụ: trạm xăng, trạm rửa xe ô tô, máy bơm nước, máy bán nước tự động v.v...
- Tự động hoá các máy công cụ: lò sấy, xi mạ v.v...

Tuy nhiên không phải bất cứ hệ thống điều khiển nào cũng sử dụng PLC mà tùy vào yêu cầu cụ thể và so sánh về yếu tố kinh tế mà ta chọn phương án điều khiển thích hợp.

1.2. Cấu trúc của một PLC

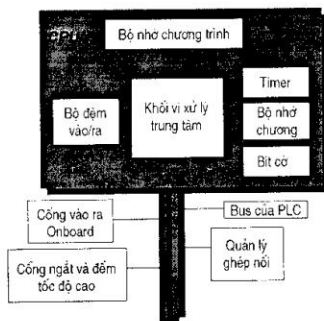
PLC gồm ba khối chức năng cơ bản, đó là:

- Bộ xử lý.
- Bộ nhớ.
- Khối vào/ra.

Trạng thái ngõ vào của PLC được phát hiện và lưu vào bộ nhớ đệm. PLC thực hiện các lệnh logic trên các trạng thái của chúng và thông qua trạng thái ngõ ra được cập nhật và lưu vào bộ nhớ đệm; sau đó, trạng thái ngõ ra trong bộ nhớ đệm được dùng để đóng mở các tiếp điểm kích hoạt các thiết bị công tác. Như vậy, sự hoạt động các thiết bị được điều khiển hoàn toàn tự động theo chương trình trong bộ nhớ. Chương trình được nạp vào PLC qua thiết bị lập trình chuyên dùng

1.2.1. Bộ xử lý

Đơn vị xử lý trung tâm (CPU- Central processing Unit) điều khiển và quản lý tất cả hoạt động bên trong PLC. Việc trao đổi thông tin giữa CPU, bộ nhớ và khối vào/ra được thực hiện thông qua hệ thống bus dưới sự điều khiển của CPU.



Hình 1.3. Cấu trúc bộ PLC

1.2.2. Bộ nhớ

Bộ nhớ (Memory) có nhiệm vụ lưu chương trình điều khiển được lập bởi người dùng và các dữ liệu khác như cờ, thanh ghi tạm, trạng thái đầu vào, lệnh điều khiển đầu ra, ... Nội dung của bộ nhớ được mã hoá dưới dạng mã nhị phân.

Mỗi lệnh trong chương trình có một vị trí riêng trong bộ nhớ, tất cả mọi vị trí trong bộ nhớ đều được "đánh số". Những số này chính là địa chỉ ô nhớ trong bộ nhớ.

1.2.3. Khối vào ra

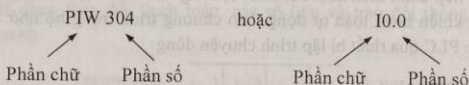
Mọi hoạt động xử lý tín hiệu bên trong PLC có mức điện áp 5V-DC và 15V-DC (điện áp cho TTL và CMOS), trong khi tín hiệu bên ngoài có thể lớn hơn nhiều thường 24V-DC ÷ 240V-DC với dòng lớn.

2. THIẾT BỊ LẬP TRÌNH S7-200

2.1. Địa chỉ các ngõ vào ra

- Địa chỉ ô nhớ trong S7-200 bao gồm hai phần: Phần chữ và phần số.

Ví dụ:



- Bộ PLC S7-200 CPU 224 có:

14 đầu vào: I0.0 ÷ I0.7 và I1.0 ÷ I1.5

10 đầu ra: Q0.0 ÷ Q0.7 và Q1.0 ÷ Q1.1

2.2. Phần chữ chỉ vị trí và kích thước của ô nhớ

- M: Chỉ ô nhớ trong miền các biến cờ có kích thước là 1 bit
- MB: Chỉ ô nhớ trong miền các biến cờ có kích thước là 1 byte (8 bit).
- MW: Chỉ ô nhớ trong miền các biến cờ có kích thước là 2 byte (16 bit).
- MD: Chỉ ô nhớ trong miền các biến cờ có kích thước là 4 byte (32 bit).
- I: Chỉ ô nhớ có kích thước là 1 bit trong miền bộ đệm ngõ vào số.
- IB: Chỉ ô nhớ có kích thước là 1 byte trong miền bộ đệm ngõ vào số.
- IW: Chỉ ô nhớ có kích thước là 2 byte (1 từ) trong miền bộ đệm ngõ vào số.
- ID: Chỉ ô nhớ có kích thước là 4 byte (2 từ) trong miền bộ đệm ngõ vào số.
- Q: Chỉ ô nhớ có kích thước là 1 bit trong miền bộ đệm ngõ ra số.
- QB: Chỉ ô nhớ có kích thước là 1 byte trong miền bộ đệm ngõ ra số.
- QW: Chỉ ô nhớ có kích thước là 2 byte trong miền bộ đệm ngõ ra số.
- QD: Chỉ ô nhớ có kích thước là 4 byte trong miền bộ đệm ngõ ra số.
- T: Chỉ ô nhớ trong miền nhớ của bộ thời gian (timer).
- C: Chỉ ô nhớ trong miền nhớ của bộ đếm (counter).

2.3. Phần số chỉ địa chỉ của byte hoặc bit trong miền nhớ đã xác định

Nếu ô nhớ đã được xác định thông qua phần chữ là có kích thước 1 bit thì phần số sẽ là địa chỉ của byte và số thứ tự của bit trong byte đó, được tách với nhau bằng dấu chấm. Ví dụ:

I 0.0: Chỉ bit 0 của byte 0 trong miền nhớ bộ đếm ngõ vào số PII.

Q 4.1: Chỉ bit 1 của byte 4 của miền nhớ bộ đếm ngõ ra số PIQ.

M 10.5: Chỉ bit 5 của byte 10 trong miền các biến cờ M.

Trong trường hợp ô nhớ đã được xác định là byte, từ hoặc từ kép thì phần số sẽ là địa chỉ của byte đầu tiên trong mảng byte của ô nhớ đó.

Ví dụ: DIB 15: Chỉ ô nhớ có kích thước 1 byte (byte 15) trong khối DB đã được mở bằng lệnh OPN DI.

DIW 18: Chỉ ô nhớ có kích thước 1 từ gồm 2 byte 18 và 19 trong khối DB đã được mở bằng lệnh OPN DB.

DB2.DBW15: Chỉ ô nhớ có kích thước 2 byte 15 và 16 trong khối dữ liệu DB2.

M105: Chỉ ô nhớ có kích thước 2 từ gồm 4 byte 105, 106, 107, 108 trong miền nhớ các biến cờ M.

Bài tập thực hành 1:

Nhận biết các thành phần vào/ra của bộ PLC S7-200 CPU 224.

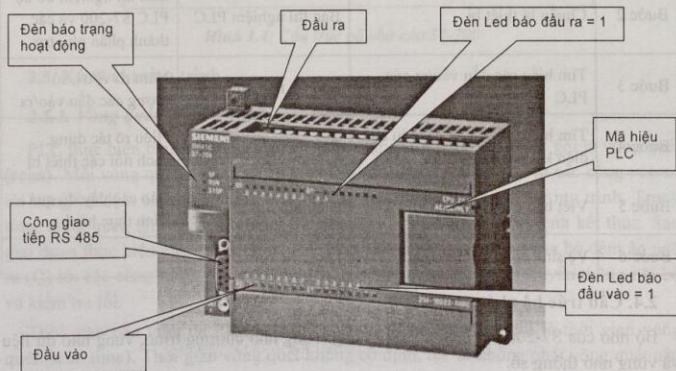
*** Mục tiêu bài tập:**

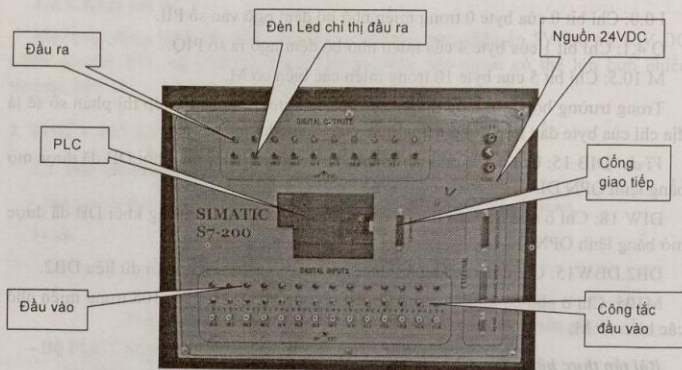
Sau khi học xong bài học, sinh viên có khả năng:

- Nhận biết được các đầu vào, ra của bộ PLC.
- Nhận biết, phân biệt được các thành phần đầu vào/ra của bộ PLC.

*** Nội dung:**

- a) Các đầu vào/ra của PLC.
- b) Khí cụ điện nổi đầu vào/ra của PLC.





** Quy trình thực hiện:*

STT	Nội dung công việc	Thiết bị, dụng cụ	Yêu cầu
Bước 1	Nghiên cứu cấu trúc bộ PLC	Sơ đồ	Nắm rõ cấu trúc, các thành phần vào/ra của PLC
Bước 2	Chuẩn bị thiết bị	Bàn thí nghiệm PLC	Bàn thí nghiệm có bộ PLC S7-200 và các thành phần phụ trợ
Bước 3	Tìm hiểu các đầu vào/ra của PLC	Bàn thí nghiệm PLC	Nắm rõ vị trí, số lượng các đầu vào/ra
Bước 4	Tìm hiểu, phân biệt các khí cụ điện kết nối đầu vào/ra	Bàn thí nghiệm PLC	Hiểu rõ tác dụng, cách nối các thiết bị
Bước 5	Viết báo cáo	Giấy bút	Báo cáo đầy đủ quá trình thực hành
Bước 6	Vệ sinh công nghiệp	Dụng cụ vệ sinh	

2.4. Cấu trúc bộ nhớ của S7-200

Bộ nhớ của S7-200 được chia làm 3 vùng: vùng nhớ chương trình, vùng nhớ dữ liệu và vùng nhớ thông số.

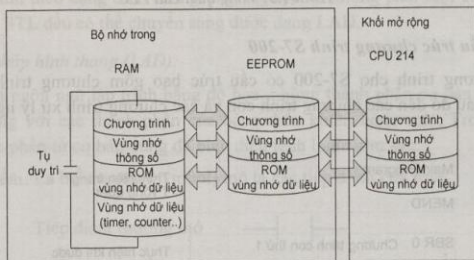
* Vùng nhớ chương trình

Vùng nhớ chương trình chứa các chỉ thị điều khiển vi xử lý để thực hiện yêu cầu điều khiển, chương trình ứng dụng sau khi soạn thảo được nạp vào ROM và vẫn tồn tại khi mất điện.

* **Vùng nhớ thông số:** Gồm các ô nhớ chứa các thông số cài đặt, mật khẩu, địa chỉ thiết bị điều khiển và các thông tin về các vùng trống có thể sử dụng. Nội dung của vùng nhớ này được chứa trong ROM giống như vùng chương trình.

* Vùng nhớ dữ liệu

Vùng nhớ dữ liệu là nơi làm việc, vùng này gồm các địa chỉ để lưu trữ các phép tính, lưu trữ tạm thời các kết quả trung gian, và chứa các hằng số được sử dụng trong các chỉ dẫn hoặc các thông số điều chỉnh khác. Ngoài ra trong vùng này còn có các phần tử và đối tượng như: Bộ định thời, bộ đếm, các bộ đếm tốc độ cao và các ngõ vào/ra analog.



Hình 1.4: Cấu trúc bộ nhớ của S7-200

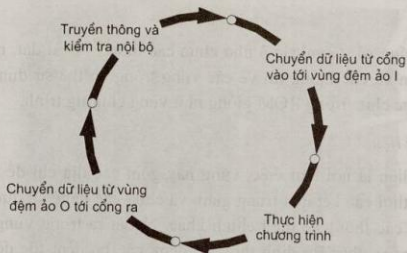
2.5. Xử lý chương trình

2.5.1. Vòng quét chương trình

PLC thực hiện chương trình theo chu trình lặp. Mỗi vòng lặp được gọi là vòng quét (scan). Mỗi vòng quét được bắt đầu bằng giai đoạn chuyển dữ liệu từ các cổng vào số tới vùng bộ đếm ảo ngõ vào (I), tiếp theo là giai đoạn thực hiện chương trình. Trong từng dòng quét, chương trình được thực hiện từ lệnh đầu tiên đến lệnh kết thúc. Sau giai đoạn thực hiện chương trình là giai đoạn chuyển các nội dung của bộ đếm ảo ngõ ra (Q) tới các cổng ra số. Vòng quét được kết thúc bằng giai đoạn truyền thông nội bộ và kiểm tra lỗi.

Thời gian cần thiết để PLC thực hiện được một vòng quét gọi là thời gian vòng quét (scan time). Thời gian vòng quét không cố định, tức là không phải vòng quét nào

cũng được thực hiện trong một khoảng thời gian như nhau. Có vòng quét thực hiện lâu, có vòng quét thực hiện nhanh tùy thuộc vào số lệnh trong chương trình được thực hiện, vào khối lượng dữ liệu truyền thông... trong vòng quét đó.



Hình 1.5: Vòng quét của PLC

2.5.2. Cấu trúc chương trình S7-200

Các chương trình cho S7-200 có cấu trúc bao gồm chương trình chính (main program), sau đó đến các chương trình con và các chương trình xử lý ngắt được chỉ ra sau đây:

Main Program
:
MEND

Thực hiện trong
1 vòng quét

SBR 0 Chương trình con thứ 1
:
RET

Thực hiện khi được
chương trình chính gọi

SBR n Chương trình con thứ n + 1
:
RET

INT 0 Chương trình xử lý ngắt thứ 1
:
RETI

Thực hiện khi có
tín hiệu báo ngắt

INT n Chương trình xử lý ngắt thứ n + 1
:
RETI

- Chương trình chính được kết thúc bằng lệnh kết thúc chương trình (MEND).
- Chương trình con là một bộ phận của chương trình. Các chương trình con phải được viết sau lệnh kết thúc chương trình chính, đó là lệnh MEND.

- Các chương trình xử lý ngắt là một bộ phận của chương trình. Nếu cần sử dụng chương trình xử lý ngắt phải được viết sau lệnh kết thúc chương trình chính MEND.

Các chương trình con được nhóm lại thành một nhóm ngay sau chương trình chính. Sau đó tiếp tục đến các chương trình xử lý ngắt. Bằng cách viết như vậy, cấu trúc chương trình được rõ ràng và thuận tiện hơn trong việc đọc chương trình sau này. Có thể tự do trộn lẫn các chương trình con và chương trình xử lý ngắt đăng sau chương trình chính.

2.5.3. Phương pháp lập trình

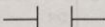
Cách lập trình cho S7-200 nói riêng và cho các PLC nói chung dựa trên hai phương pháp cơ bản. Phương pháp hình thang (**Ladder**, viết tắt là **LAD**) và phương pháp liệt kê lệnh (**Statement list**, viết tắt là **STL**).

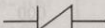
Nếu có một chương trình viết dưới dạng LAD, thiết bị lập trình sẽ tự động tạo ra một chương trình theo dạng STL tương ứng. Ngược lại không phải mọi chương trình viết dưới dạng STL đều có thể chuyển sang được dạng LAD.

* Phương pháp hình thang (LAD):

LAD là một ngôn ngữ lập trình bằng đồ họa, những thành phần cơ bản dùng trong LAD tương ứng với các thành phần của bảng điều khiển bằng role. Trong chương trình LAD, các phần tử cơ bản dùng để biểu diễn lệnh logic như sau:

- ♦ Tiếp điểm: Là biểu tượng (symbol) mô tả các tiếp điểm của role:

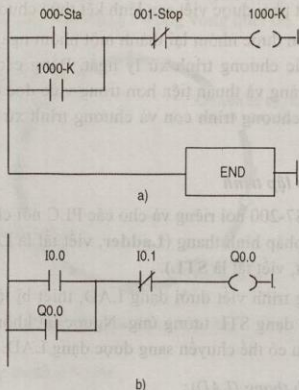
Tiếp điểm thường mở 

Tiếp điểm thường đóng 

- ♦ Cuộn dây (coil): Là biểu tượng  mô tả role được mắc theo chiều dòng điện cung cấp cho role.
- ♦ Hộp (box): Là biểu tượng mô tả các hàm khác nhau, nó làm việc khi có dòng điện chạy đến hộp. Những dạng hàm thường được biểu diễn bằng hộp là các bộ thời gian (timer), bộ đếm (counter) và các hàm toán học. Cuộn dây và các hộp phải mắc đúng chiều dòng điện.

Mạng LAD: Là đường nối các phần tử thành một mạch hoàn thiện, đi từ đường nguồn bên trái sang đường nguồn bên phải. Đường nguồn bên trái là dây pha, đường nguồn bên phải là dây trung hòa và cũng là đường trở về nguồn cung cấp (thường không được thể hiện khi dùng chương trình tiện dụng STEPT - MICRO/DOS hoặc STEPT - MICRO/WIN. Dòng điện chạy từ trái qua tiếp điểm đến đóng các cuộn dây hoặc các hộp trở về bên phải nguồn.

Ví dụ:



Hình 1.6. Lập trình theo dạng LAD

a) Chương trình lập trình cho PLC CPM1; b) Chương trình lập trình PLC S7-200.

* **Phương pháp liệt kê lệnh (STL):** Là phương pháp thể hiện chương trình dưới dạng tập hợp các câu lệnh. Mỗi câu lệnh trong chương trình, kể cả những lệnh hình thức biểu diễn một chức năng của PLC.

Ví dụ:

LD	000
OR	1000
AND NOT	001
OUT	1000
END (01)	

2.6. Cài đặt và sử dụng phần mềm Step 7 Microwin 4.0

2.6.1. Những yêu cầu với máy tính

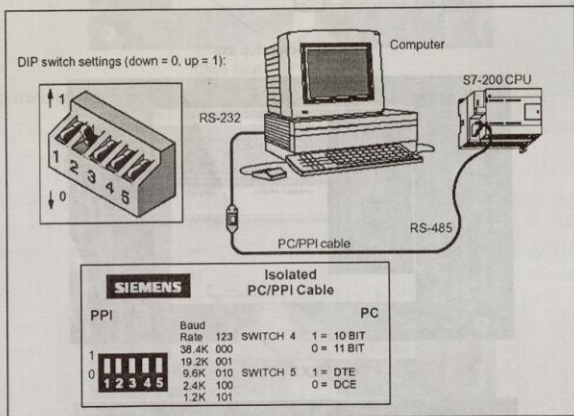
Máy tính cá nhân PC, muốn cài đặt được phần mềm STEP 7-Micro/Win phải thỏa mãn những yêu cầu sau đây:

- 640 Kbyte RAM (ít nhất phải có 500 Kbyte bộ nhớ còn trống).
- Màn hình 24 dòng, 80 cột ở chế độ văn bản.
- Còn khoảng 2 Mbyte trống trong ổ đĩa cứng.

- Có hệ điều hành MS-DOS ver. 5.0 hoặc cao hơn.
- Bộ chuyển đổi RS232-RS485 phục vụ ghép nối truyền thông trực tiếp giữa máy tính và PLC (truyền thông online).

Truyền thông giữa STEP 7-Micro/Win với S7-200 CPU qua cổng truyền thông ở phía đáy của PLC. Sử dụng cáp có bộ chuyển đổi RS232-RS485, được gọi là cáp PC/PPI, để nối máy tính với PLC tạo thành mạch truyền thông trực tiếp.

Cắm 1 đầu của cáp PC/PPI với cổng truyền thông 9 chân của PLC, còn đầu kia với cổng truyền thông nối tiếp RS-232C của máy PC. Nếu máy PC có cổng truyền thông nối tiếp RS232 với 25 chân, thì phải ghép nối qua bộ chuyển đổi 25 chân/9 chân để có thể ghép nối với cáp truyền thông PC/PPI.

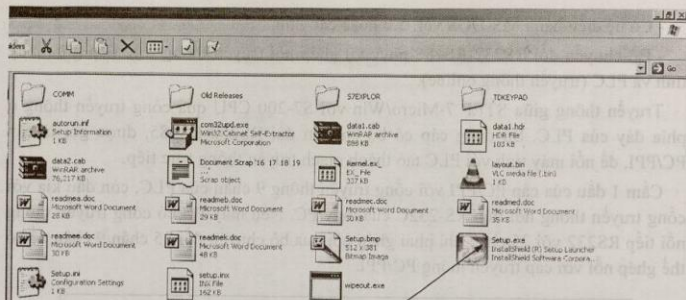


Hình 1.7. Kết nối máy tính với PLC qua cổng truyền thông PPI dùng cáp PC/PPI

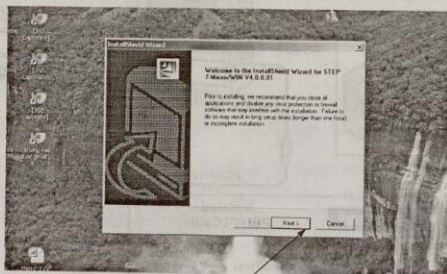
2.6.2. Cài đặt và sử dụng phần mềm Step 7 Microwin 4.0

Sau khi kiểm tra bộ nhớ, nếu thấy ổ đĩa cứng hoàn toàn có đủ khả năng để cài phần mềm STEP 7-Micro/Win vào ổ cứng, thì lần lượt tiến hành các bước:

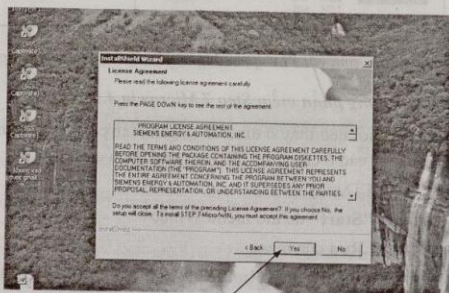
1. Chèn đĩa CD vào ổ CD máy tính.
2. Kích chuột vào nút **"Start"** để mở menu Window.
3. Kích chuột vào mục **Run** của menu.
4. Cài đặt từ ổ CD: Trong hộp thoại **Run**, gõ **e:\setup** và kích OK hoặc ENTER.
5. Cài từ ổ cứng: click file **Setup.exe**.



Click chọn setup.exe



Chọn NEXT



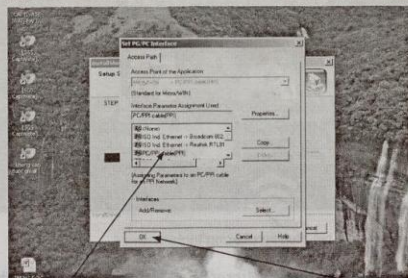
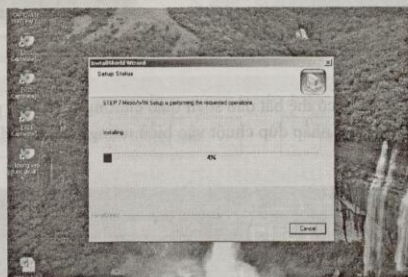
Chọn Yes để chấp nhận các điều khoản của phần mềm

Nhấn Brower... để lựa chọn nơi cài đặt phần mềm rồi chọn Next để bắt đầu quá trình cài đặt:



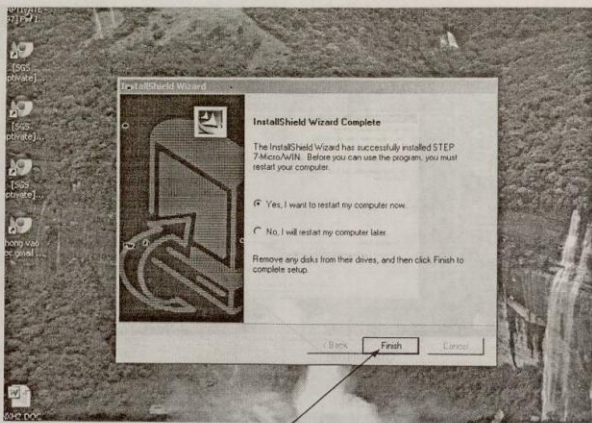
Chọn Next

Quá trình cài đặt đang diễn ra:



Chọn kết nối giữa PLC và máy tính bằng Cable PC/PPi

Xong chọn OK



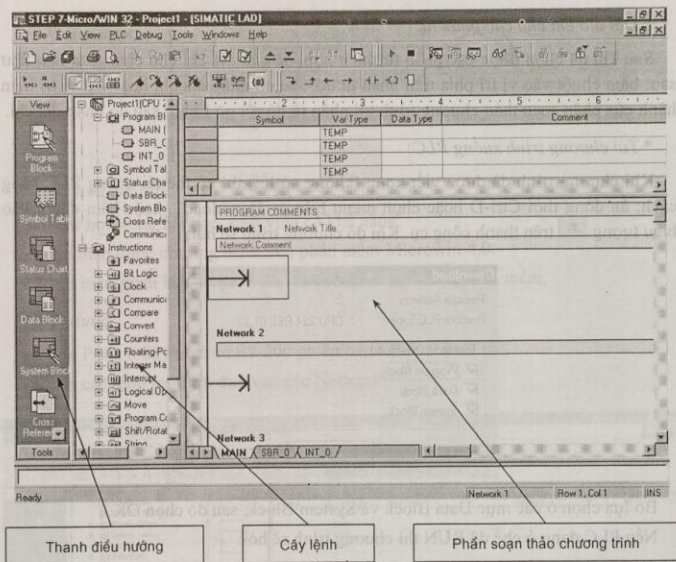
Click Finish để khởi động lại máy tính kết thúc quá trình cài đặt

Sau khi đã cài đặt xong có thể bắt đầu soạn thảo chương trình nhờ phần mềm STEP 7-Micro/Win 32 bằng cách nhấp đúp chuột vào biểu tượng STEP 7-Micro/Win 32 trên màn hình.



Click vào biểu tượng chương trình

Sau khi mở chương trình Step7-Micro/Win để tạo Project mới thì giao diện của chương trình như sau:



Ở thanh điều hướng. Có thể dùng các biểu tượng trên thanh công cụ này để mở các phần tử của dự án.

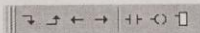
*** Mở phần soạn thảo chương trình:**

Nếu trên màn hình chưa xuất hiện phần soạn thảo chương trình thì làm như sau: Nháy chuột vào biểu tượng *Program Block* để mở phần soạn thảo chương trình.

*** Đưa các phần tử vào chương trình:**

Có thể dùng cây lệnh để chèn các phần tử dưới dạng LAD vào chương trình bằng cách: kéo và thả các phần tử từ cây lệnh vào chương trình. Nếu nháy kép vào một phần tử thì phần tử đó sẽ được chèn vào vị trí hiện hành của phần soạn thảo chương trình.

Cũng có thể chọn phần tử bằng cách: chọn từ thanh công cụ *instruction*:



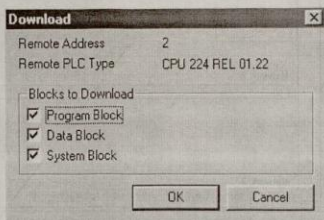
Nếu thanh công cụ này chưa xuất hiện thì có thể làm như sau: Chọn menu View → Toolbars → Instruction.

*** Đặt địa chỉ cho các phần tử:**

Sau khi đã chọn được các phần tử thích hợp, để gán địa chỉ cho chúng, ta làm như sau: bấm chuột vào vị trí phía trên phần tử đó, sau đó nhập địa chỉ hoặc đặt vị trí hiện hành của phần soạn thảo chương trình tới phần tử đó rồi gõ Enter, sau đó nhập địa chỉ.

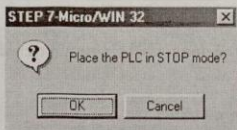
*** Tải chương trình xuống PLC:**

Khi chương trình đã được viết xong (không có lỗi) thì có thể tải xuống PLC bằng cách: ấn đồng thời Ctrl-D hoặc chọn menu File → Download... hoặc bấm chuột vào biểu tượng  trên thanh công cụ. Khi đó chương trình sẽ hỏi:

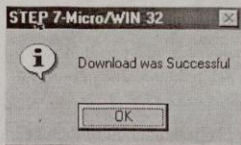


Bỏ lựa chọn ở các mục Data Block và System Block, sau đó chọn OK.

Nếu PLC đang ở chế độ RUN thì chương trình sẽ hỏi:

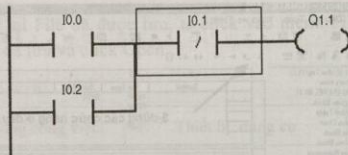


Chọn OK để chương trình tiếp tục. Sau khi đã tải xong, chương trình sẽ thông báo:



Bài tập thực hành 2:

Soạn thảo chương trình sau trên phần mềm Microwin 4.0 và lưu lại thành file: tapsoanthao.mwp.



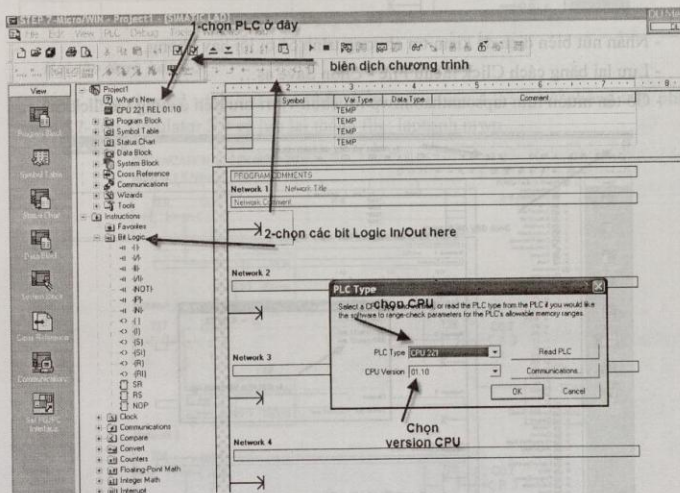
*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

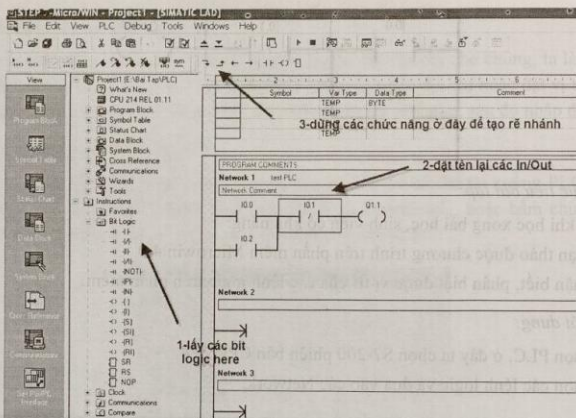
- Soạn thảo được chương trình trên phần mềm Microwin 4.0.
- Nhận biết, phân biệt được vị trí của các lệnh logic trên phần mềm.

*** Nội dung:**

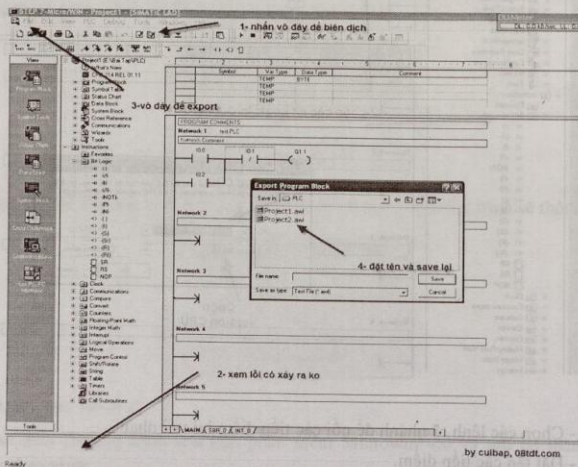
- Chọn PLC, ở đây ta chọn S7-200 phiên bản CPU 224.
- Chọn các lệnh logic và đưa vào các Network.



- Chọn các lệnh rẽ nhánh để nối các tiếp điểm lại với nhau.
- Đặt tên các tiếp điểm.



- Nhấn nút biên dịch để kiểm tra chương trình có bị lỗi hay không.
- Lưu lại bằng cách Click menu File - Chọn Save as...
- Gõ tên muốn lưu: tapsoanthao.mwp rồi chọn vị trí lưu trên ổ cứng và click Save.



- Khi muốn mở lại File đã được lưu, ta click vào menu File, chọn Open, rồi tìm đường dẫn đến file đã lưu và click Open.

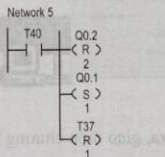
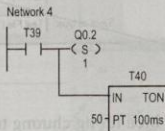
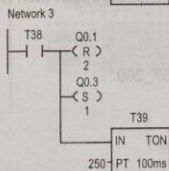
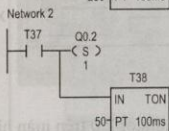
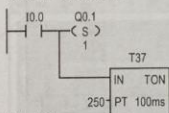
** Quy trình thực hiện:*

STT	Nội dung công việc	Thiết bị, dụng cụ	Yêu cầu
Bước 1	Tìm hiểu chương trình	Chương trình cần soạn thảo	Nắm rõ cấu trúc, các lệnh cần sử dụng
Bước 2	Chuẩn bị thiết bị	Máy vi tính, phần mềm Microwin 4.0	Máy tính và phần mềm hoạt động tốt
Bước 3	Soạn thảo chương trình	Máy vi tính, phần mềm Microwin 4.0	Chương trình soạn thảo đúng, đủ
Bước 4	Lưu lại chương trình	Máy vi tính, phần mềm Microwin 4.0	Lưu đúng tên file, nhớ vị trí lưu
Bước 5	Mở lại chương trình	Máy vi tính, phần mềm Microwin 4.0	Chương trình mở lại đúng, không báo lỗi
Bước 6	Vệ sinh công nghiệp	Dụng cụ vệ sinh	

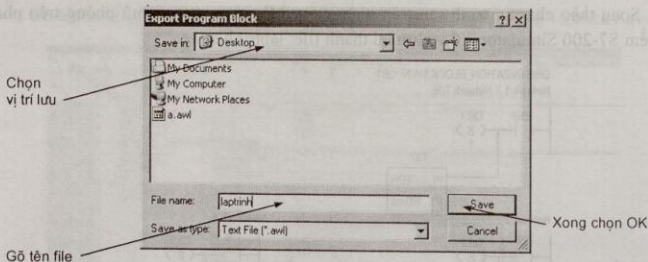
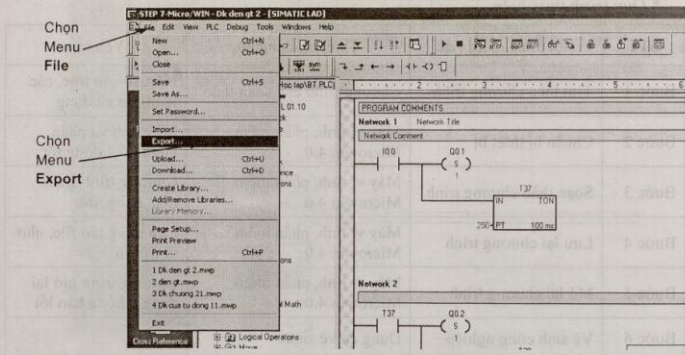
Bài tập thực hành 3:

Soạn thảo chương trình sau trên phần mềm Microwin 4.0 và mô phỏng trên phần mềm S7-200 Simulator 2.0 và lưu lại thành file: laptrinh.mwp

ORGANIZATION BLOCK MAIN: OB1
Network 1 // Network Title



Sau khi soạn thảo xong chương trình trên phần mềm Microwin 4.0, ta tiến hành xuất file vừa soạn thảo thành định dạng *.awl để tiến hành mô phỏng.



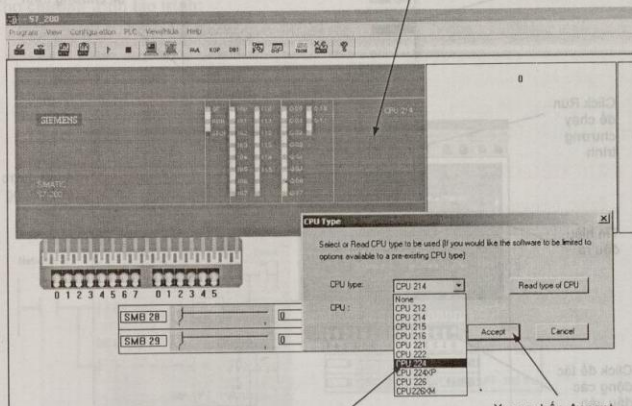
Ta click mở biểu tượng chương trình S7-200 Simulator 2.0 trên màn hình:



S7_200.EXE
Aplicación MFC S7_200

Sau khi mở ra, giao diện chương trình như sau:

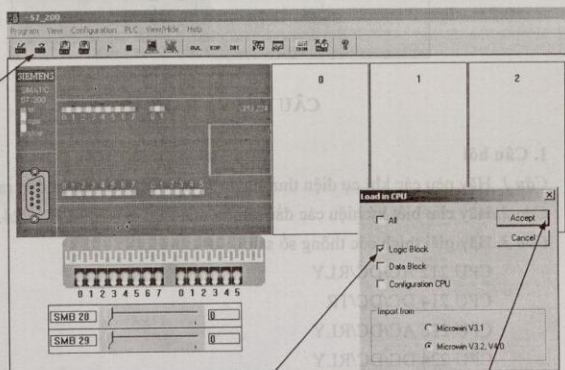
Click vào vùng này để mở bảng chọn CPU



Chọn kiểu CPU 224

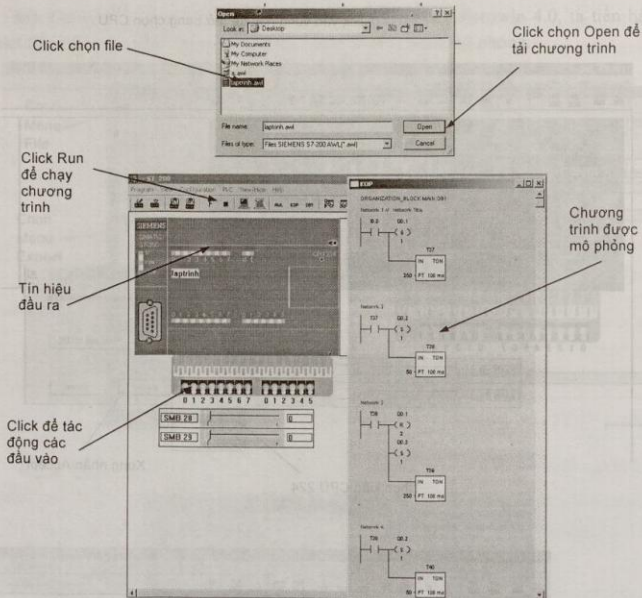
Xong nhấn Accept

Click biểu tượng Load Program



Click chọn Logic Block

Click Accept



CÂU HỎI VÀ BÀI TẬP

1. Câu hỏi

Câu 1. Hãy nêu các khí cụ điện thường được nối với đầu vào và đầu ra của PLC.

Câu 2. Hãy cho biết ký hiệu các đầu vào/ra của PLC S7-200 CPU 224.

Câu 3. Hãy giải thích các thông số sau:

CPU 212 AC/DC/RLY

CPU 214 DC/DC/TR

CPU 226 AC/DC/RLY

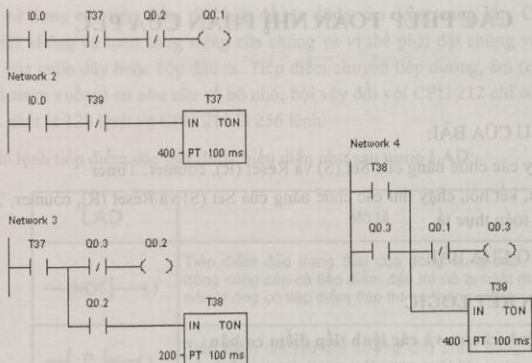
CPU 224 DC/DC/RLY

Câu 4: Hãy so sánh sự giống và khác nhau giữa việc điều khiển động cơ bằng bộ PLC và bằng công tắc tơ.

Câu 5: Hãy cho biết vị trí các đầu vào/ra của PLC S7-200 CPU 224 trên vùng nhớ?

2. Bài tập

Cho chương trình lập trình:



Yêu cầu:

- Soạn thảo chương trình trên phần mềm Microwin 4.0.
- Mô phỏng hoạt động của chương trình trên phần mềm S7-200 Simulator 2.0.

Bài 2

CÁC PHÉP TOÁN NHỊ PHÂN CỦA PLC

MỤC TIÊU CỦA BÀI:

- Trình bày các chức năng của Set (S) và Reset (R), counter, Timer
- Lập trình, kết nối, chạy thử các chức năng của Set (S) và Reset (R), counter, Timer trong các bài toán thực tế.

NỘI DUNG CỦA BÀI:

1. CÁC LIÊN KẾT LOGIC

1.1. Các lệnh vào/ra và các lệnh tiếp điểm cơ bản

1.1.1. Lệnh vào/ra

* Load (LD):

Lệnh LD nạp giá trị logic của một tiếp điểm vào trong bit đầu tiên của ngăn xếp. Các giá trị cũ còn lại trong ngăn xếp bị đẩy lùi xuống một bit.

Network 1



* Load Not (LDN):

Lệnh LDN nạp giá trị một logic nghịch đảo của một tiếp điểm vào trong bit đầu tiên của ngăn xếp. Các giá trị cũ còn lại trong ngăn xếp bị đẩy lùi xuống một bit.



* OUTPUT (=):

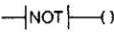
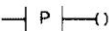
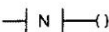
Lệnh sao chép nội dung của bit đầu tiên trong ngăn xếp vào bit được chỉ định trong lệnh. Nội dung của ngăn xếp không bị thay đổi.



1.1.2. Các lệnh tiếp điểm đặc biệt

Có thể dùng các lệnh tiếp điểm đặc biệt để phát hiện sự chuyển tiếp trạng thái của xung (sườn xung) và đảo lại trạng thái dòng điện cung cấp (giá trị của định ngăn xếp). LAD sử dụng các tiếp điểm đặc biệt để tác động vào dòng cung cấp. Các tiếp điểm đặc biệt không có toán hạng riêng của chúng và vì thế phải đặt chúng vào vị trí phía trước của cuộn dây hoặc hộp đầu ra. Tiếp điểm chuyển tiếp dương, âm (các lệnh sườn lên và sườn xuống) có nhu cầu về bộ nhớ, bởi vậy đối với CPU 212 chỉ có thể sử dụng nhiều nhất là 128 lệnh và CPU 214 là 256 lệnh.

Các lệnh tiếp điểm đặc biệt được biểu diễn như sau trong LAD:

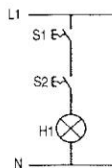
LAD	Mô tả
	Tiếp điểm đảo trạng thái của dòng cung cấp. Nếu dòng cung cấp có tiếp điểm đảo thì nó bị ngắt mạch, nếu không có tiếp điểm đảo thì nó thông mạch.
	Tiếp điểm chuyển đổi dương cho phép dòng cung cấp thông mạch trong một vòng quét khi sườn xung điều khiển chuyển từ 0 lên 1
	Tiếp điểm chuyển đổi âm cho phép dòng cung cấp thông mạch trong một vòng quét khi sườn xung điều khiển chuyển từ 1 lên 0

1.2. Các lệnh liên kết logic cơ bản

1.2.1. Lệnh AND (A)

Lệnh A phối hợp giá trị logic của một tiếp điểm n với giá trị bit đầu tiên của ngăn xếp. Kết quả phép tính được đặt lại vào bit đầu tiên trong ngăn xếp. Giá trị của các bit còn lại trong ngăn xếp không bị thay đổi.

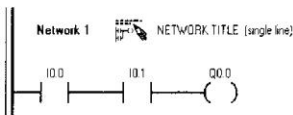
Ví dụ:



Hình 2.1. Lệnh And

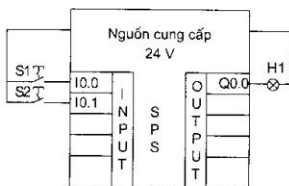
Sơ đồ mạch điện trên mô tả một liên kết AND. Đèn H1 chỉ sáng khi tất cả các công tắc được đóng lại. Khi 1 công tắc hở mạch thì đèn H1 cũng bị cắt mạch. Liên kết AND có trạng thái 1 khi tất cả các ngõ vào có trạng thái 1.

Chương trình dạng LAD:



Xác lập vào/ra		
Kí hiệu	Địa chỉ	Chú thích
S1	I0.0	Công tắc thường mở
S2	I0.1	Công tắc thường mở
H1	Q0.0	Đèn báo

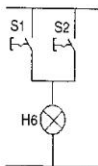
Sơ đồ kết nối PLC:



1.2.2. Lệnh OR (O)

Lệnh OR phối hợp giá trị logic của một tiếp điểm n với giá trị bit đầu tiên của ngăn xếp. Kết quả của phép tính được đặt lại vào bit đầu tiên trong ngăn xếp. Giá trị của các bit còn lại trong ngăn xếp không bị thay đổi.

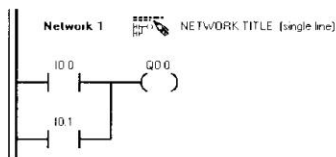
Ví dụ:



Hình 2.2. Lệnh OR

Sơ đồ mạch điện trên mô tả một liên kết OR. Đèn H6 sẽ sáng nếu một hoặc tất cả các công tắc đều đóng mạch. Ngõ ra của liên kết OR luôn luôn có trạng thái 1 nếu ít nhất một trong các ngõ vào có trạng thái 1.

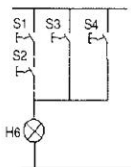
Chương trình dạng LAD:



Xác lập vào/ra		
Kí hiệu	Địa chỉ	Chú thích
S1	I0.0	Công tắc thường mở
S2	I0.1	Công tắc thường mở
H6	Q0.0	Đèn báo

1.3. Liên kết các cổng logic cơ bản

1.3.1. Mạch hỗn hợp AND trước OR dạng 1



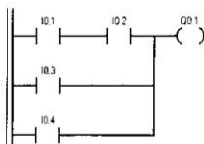
Hình 2.3. Sơ đồ mạch hỗn hợp AND trước OR dạng 1

Xác lập vào/ra		
Kí hiệu	Địa chỉ	Chú thích
S1	I0.1	Công tắc thường mở
S2	I0.2	Công tắc thường mở
S3	I0.3	Công tắc thường mở
S4	I0.4	Công tắc thường mở
H6	Q0.0	Đèn báo

Mô tả:

Ngõ ra Q0.0 chỉ có tín hiệu bằng 1, nếu các ngõ vào I0.1 và I0.2 của cổng AND đều có tín hiệu bằng 1 hoặc các ngõ vào I0.3 hoặc I0.4 có tín hiệu bằng 1.

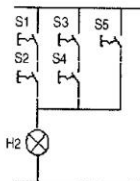
Chương trình dạng LAD:



Chương trình dạng STL:

```
NETWORK 1      //NETWORK TITLE (single line)
//
//NETWORK COMMENTS
//
LD      I0.1
A       I0.2
O       I0.3
O       I0.4
=       Q0.1
```

1.3.2. Mạch hỗn hợp AND trước OR dạng 2



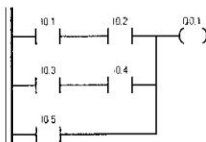
Hình 2.4. Sơ đồ mạch hỗn hợp AND trước OR dạng 2

Xác lập vào/ra		
Kí hiệu	Địa chỉ	Chú thích
S1	I0.1	Công tắc thường mở
S2	I0.2	Công tắc thường mở
S3	I0.3	Công tắc thường mở
S4	I0.4	Công tắc thường mở
S5	I0.5	Công tắc thường mở
H2	Q0.1	Đèn báo

Mô tả:

Ngõ ra Q0.1 chỉ có tín hiệu bằng 1, nếu có ít nhất một ngõ vào của các cổng AND hoặc ngõ vào I0.5 có tín hiệu 1.

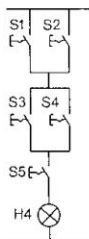
Chương trình dạng LAD:



Chương trình dạng STL:

```
NETWORK 1      //NETWORK TITLE (single line)
//
//NETWORK COMMENTS
//
LD    I0.1
A     I0.2
LD    I0.3
A     I0.4
OLD
O     I0.5
```

1.3.3. Mạch hỗn hợp OR trước AND dạng 1



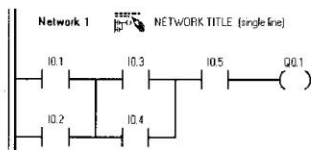
Hình 2.5: Sơ đồ mạch hỗn hợp OR trước AND dạng 1

Xác lập vào/ra		
Kí hiệu	Địa chỉ	Chú thích
S1	I0.1	Công tắc thường mở
S2	I0.2	Công tắc thường mở
S3	I0.3	Công tắc thường mở
S4	I0.4	Công tắc thường mở
S5	I0.5	Công tắc thường mở
H4	Q0.1	Đèn báo

Mô tả:

Ngõ ra Q0.1 chỉ có tín hiệu bằng 1, nếu 2 ngõ vào của cổng OR và ngõ vào I0.5 có tín hiệu bằng 1.

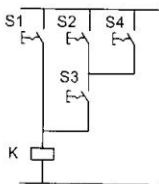
Chương trình dạng LAD:



Chương trình dạng STL:

```
NETWORK 1 //NETWORK TITLE (single line)
//
//NETWORK COMMENTS
//
LD I0.1
O I0.2
LD I0.3
O I0.4
ALO
A I0.5
* Q0.1
```

1.3.4. Mạch hỗn hợp OR trước AND dạng 2



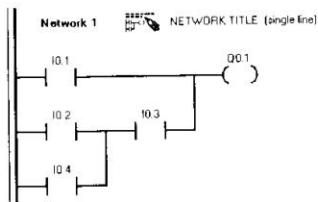
Hình 2.6. Sơ đồ mạch hỗn hợp OR trước AND dạng 2

Xác lập vào/ra		
Kí hiệu	Địa chỉ	Chú thích
S1	I0.1	Công tắc thường mở
S2	I0.2	Công tắc thường mở
S3	I0.3	Công tắc thường mở
S4	I0.4	Công tắc thường mở
K1	Q0.1	Cuộn dây ngắt điện từ

Mô tả:

Ngõ ra Q0.1 chỉ có tín hiệu bằng 1, nếu ngõ vào I0.1 hoặc ngõ vào I0.3 và ngõ vào I0.2 hoặc I0.4 có tín hiệu bằng 1.

Chương trình dạng LAD:

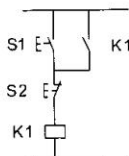


Chương trình dạng STL:

```
NETWORK 1 //NETWORK TITLE (single line)
//
//NETWORK COMMENTS
//
LD I0.1
LD I0.2
O I0.4
A I0.3
OLD
= Q0.1
```

Bài tập thực hành 1:

Lập trình điều khiển công tắc tơ K1 đóng mở trên PLC S7-200 theo sơ đồ sau:



* Mục tiêu bài tập:

Sau khi học xong bài học, sinh viên có khả năng:

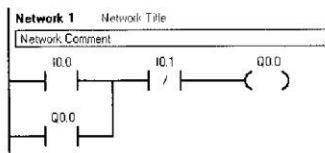
- Lập trình được chương trình điều khiển công tắc tơ K1 trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

* Nội dung:

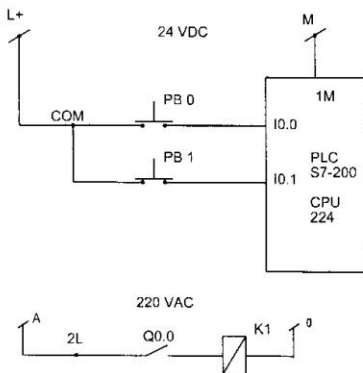
a) Quy ước các đầu vào/ra:

STT	Tiếp điểm logic	Chức năng
1	I0.0	Nút bấm thường mở S1
2	I0.1	Nút bấm thường đóng S2
3	Q0.0	Công tắc tơ K1

b) Chương trình lập trình:



c) Sơ đồ nối phần cứng:

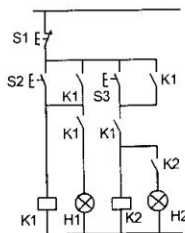


** Quy trình thực hiện:*

STT	Nội dung công việc	Phương pháp thực hiện	Yêu cầu
1	Tìm hiểu yêu cầu bài toán	Quan sát sơ đồ, lắng nghe, tư duy phân tích yêu cầu bài toán	Nắm rõ những công việc cần phải làm
2	Quy ước các đầu vào/ra	Từ yêu cầu bài toán, đưa ra số lượng và quy ước các đầu vào/ra cần thiết	Các đầu vào/ra đầy đủ, quy ước chính xác, hợp lý
3	Lựa chọn lệnh	Theo quy ước đầu vào/ra đưa ra các lệnh tương ứng phù hợp	Các lệnh đưa ra chính xác, hợp lý
4	Viết chương trình lập trình	Căn cứ yêu cầu bài toán, quy ước đầu vào/ra, các lệnh cần thiết để viết chương trình	Chương trình đúng với yêu cầu bài toán
5	Chuẩn bị thiết bị, dụng cụ, dây nối	Căn cứ quy ước đầu vào/ra, chuẩn bị dây nối, kiểm tra bàn thực tập	Bàn thí nghiệm được cấp nguồn, đầy đủ dây nối
6	Nối dây phản cứng, nạp chương trình vào PLC	Căn cứ quy ước đầu vào/ra nối dây trên bàn thực tập, tải chương trình vào PLC từ phần mềm MicroWin 4.0	- Đầu vào/ra được nối chính xác - PLC không báo lỗi
7	Chạy chương trình	Tác động đầu vào, kiểm tra kết quả đầu ra	Kết quả đúng với yêu cầu bài toán
8	Vệ sinh công nghiệp	Dụng cụ vệ sinh	Thiết bị dụng cụ gọn gàng, sạch sẽ

Bài tập thực hành 2:

Lập trình điều khiển công tắc tơ K1, K2 và bóng đèn H1, H2 đóng mở trên PLC S7-200 theo sơ đồ sau:



** Mục tiêu bài tập:*

Sau khi học xong bài học, sinh viên có khả năng:

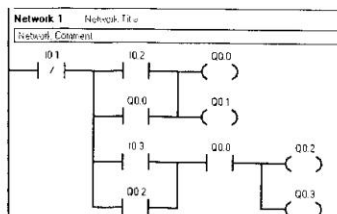
- Lập trình được chương trình điều khiển các công tắc tơ và bóng đèn trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

** Nội dung:*

a) Quy ước các đầu vào/ra:

STT	Tiếp điểm logic	Chức năng
1	I0.1	Nút bấm thường đóng S1
2	I0.2	Nút bấm thường mở S2
3	I0.3	Nút bấm thường mở S3
4	Q0.0	Công tắc tơ K1
5	Q0.1	Đèn H1
6	Q0.2	Công tắc tơ K2
7	Q0.3	Đèn H2

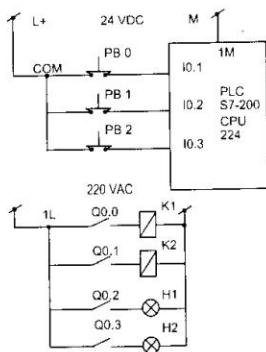
b) Chương trình lập trình:



Viết ở dạng STL:

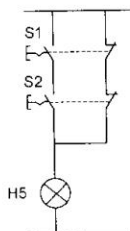
Network 1	Network Title
Network Comment	
LDN	I0.1
LPS	
LD	I0.2
O	Q0.0
ALD	
=	Q0.0
=	Q0.1
LPP	
LD	I0.2
O	I0.3
A	Q0.0
ALD	
=	Q0.2
=	Q0.3

c) Nối dây phần cứng:



Bài tập thực hành 3:

Cho sơ đồ điều khiển bóng đèn H5 như sau:



Yêu cầu:

- Viết chương trình điều khiển bóng đèn H5 đóng mở trên PLC S7-200.
- Nạp chương trình vào PLC và chạy thử.

2. CÁC LỆNH GHI/XÓA GIÁ TRỊ TIẾP ĐIỂM

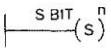
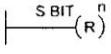
2.1. Lệnh SET (S) và RESET (R) trong S7-200

2.1.1. Cấu trúc lệnh

- Lệnh Set (S) có tác dụng đặt (đóng) và duy trì trạng thái đóng của cuộn dây đầu ra hoặc đặt 1 vào bit của bộ nhớ. Điểm tác động được xác định là: bắt đầu tại địa chỉ xác định trong lệnh và số điểm được tác động là từ 1 đến 255 điểm.

- Lệnh Reset (R) có tác dụng mở và duy trì trạng thái mở của cuộn dây đầu ra hoặc xoá các bit bộ nhớ về 0. Điểm tác động được xác định là: bắt đầu tại địa chỉ xác định trong lệnh và số điểm được tác động là từ 1 đến 255 điểm.

Mô tả trong LAD:

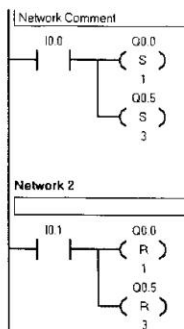
LAD	Mô tả	Toán hạng
	Đóng một mảng gồm n các tiếp điểm kể từ S_BIT	S_BIT I, Q, M, SM, C, V (bit)
	Ngắt một mảng gồm n các tiếp điểm kể từ S_BIT. Nếu S_BIT lại chỉ vào Timer hoặc Counter thì lệnh sẽ xóa bit đầu ra của Timer, Counter đó.	n IB, QB, MB, SMB, VB, AC, hằng số, *VB, *AC

Mô tả trong STL:

STL	Mô tả	Toán hạng
S S_BIT n	Ghi giá trị logic vào một mảng gồm n bit kể từ địa chỉ S_BIT	S-BIT: I, Q, M, SM, C, V (bit)
R S_BIT n	Xóa một mảng gồm n bit kể từ địa chỉ S_BIT. Nếu S_BIT lại chỉ vào Timer hoặc Counter thì lệnh sẽ xóa bit đầu ra của Timer, Counter đó	n: IB, QB, MB, SMB, VB, AC, hằng số, *VB, *AC

Ví dụ:

Viết trong LAD:



Viết trong STL:

Network Comment		
LD	I0.0	
S	Q0.0	1
S	Q0.5	3

Network 2		
LD	I0.1	
R	Q0.0	1
R	Q0.5	3

- Khi bấm I0.0, đầu ra Q0.0 được set lên bằng 1; đầu ra Q0.5, Q0.6, Q0.7 cũng bằng 1 và bằng 1 khi nhả tay khỏi nút bấm.

- Khi bấm I0.1, đầu ra Q0.0 được reset xuống bằng 0; đầu ra Q0.5, Q0.6, Q0.7 cũng bằng 0 và bằng 0 khi nhả tay khỏi nút bấm.

2.1.2. Bài tập

Bài tập thực hành 4:

Lập trình điều khiển 5 bóng đèn trên PLC S7-200 theo yêu cầu sau:

- Ấn nút 1 thì đèn 1 và đèn 5 sáng
- Ấn nút 2 thì đèn 2 và đèn 4 sáng
- Ấn nút 3 thì đèn 3 sáng
- Ấn nút 4 thì 5 đèn tắt.

* Mục tiêu bài tập:

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển các bóng đèn trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bản thí nghiệm.

* Nội dung:

a) Quy ước các đầu vào/ra:

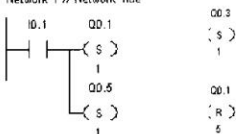
STT	Tiếp điểm logic	Chức năng
1	I0.1	Nút bấm 1
2	I0.2	Nút bấm 2
3	I0.3	Nút bấm 3
4	I0.4	Nút bấm 4
5	Q0.1	Đèn 1
6	Q0.2	Đèn 2
7	Q0.3	Đèn 3
8	Q0.4	Đèn 4
9	Q0.5	Đèn 5

b) Chương trình lập trình:

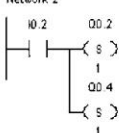
- Dạng I.AD:

ORGANIZATION_BLOCK MAIN.OB1

Network 1 // Network Title



Network 2



- Dạng STL:

Network 1 Network Title

Network Comment

LD IO 1
S Q0 1, 1
S Q0 5, 1

Network 2

LD IO 2
S Q0 2, 1
S Q0 4, 1

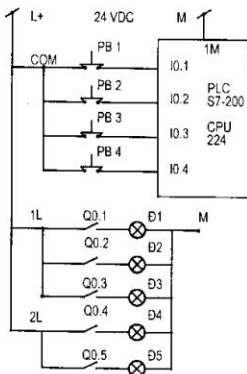
Network 3

LD IO 3
S Q0 3, 1

Network 4

LD IO 4
R Q0 1, 5

c) Nối dây phản ứng:



2.2. Timer

2.2.1. Khái niệm

- Timer là bộ tạo thời gian trễ giữa tín hiệu vào và tín hiệu ra nên trong bộ điều khiển vẫn gọi là khâu trễ.

- Bộ điều khiển lập trình S7-200 có 256 timer (với CPU 224) được chia làm 3 loại khác nhau:

- + Timer tạo thời gian trễ không có nhớ (ON - Delay Timer) kí hiệu là TON.

- + Timer tạo thời gian trễ có nhớ (Retentive - ON - Delay Timer) kí hiệu là TONR.

- + Timer tạo thời gian trễ mở (OFF - Delay Timer) kí hiệu là TOF.

- Cả hai loại Timer đều có 3 loại với 3 độ phân giải thời gian khác nhau là: 1ms; 10ms; 100ms.

Thời gian trễ τ được tạo ra chính là tích của độ phân giải của Timer được chọn và giá trị đặt trước cho Timer PT.

Thời gian trễ: $\tau = \text{Độ phân giải} \times \text{PT}$

- Reset timer là công việc đưa 1 timer về trạng thái ban đầu. Khi reset một bộ Timer, T-word và T-bit của nó đồng thời được xóa và có giá trị bằng 0, như vậy giá trị đếm tức thời được đặt về 0 và tín hiệu đầu ra cũng có logic bằng 0. Có thể reset bất cứ bộ Timer nào của S7-200 bằng lệnh R. Có hai phương pháp để reset một Timer TON:

- Xóa tín hiệu đầu vào.

- Dùng lệnh reset.

Dùng lệnh Reset (R) là phương pháp duy nhất để reset các bộ Timer kiểu TONR.

** Cập nhật Timer có độ phân giải là 1ms:*

- CPU của S7-200 có các bộ Timer có độ phân giải 1ms cho phép PLC cập nhật và thay đổi giá trị đếm tức thời trong T-word mỗi 1ms một lần. Các bộ Timer có độ phân giải thấp này có khả năng điều khiển chính xác các thao tác.

Ngay sau khi bộ Timer với độ phân giải 1ms được kích, việc cập nhật để thay đổi giá trị đếm tức thời trong T-word hoàn toàn tự động. Chỉ nên đặt giá trị rất nhỏ cho PT của bộ Timer có độ phân giải 1ms. Tần số cập nhật để thay đổi giá trị đếm tức thời và T-bit của một bộ Timer có độ phân giải 1ms không phụ thuộc vào vòng quét (scan) của bộ điều khiển và vòng quét của chương trình đang chạy. Giá trị đếm tức thời và T-bit của bộ Timer này có thể được cập nhật vào bất kỳ thời điểm nào trong vòng quét và được cập nhật nhiều lần trong một vòng quét nếu thời gian vòng quét lớn hơn 1ms.

Thực hiện lệnh R đối với một Timer có độ phân giải 1ms đang ở trạng thái làm việc có nghĩa là đưa Timer đó về trạng thái ban đầu, giá trị đếm tức thời của Timer được đưa về 0 và T-bit nhận giá trị logic 0.

** Cập nhật Timer có độ phân giải là 10ms*

CPU của S7-200 có các bộ Timer với độ phân giải 10ms. Sau khi đã được kích, việc cập nhật T-word và T-bit để thay đổi giá trị đếm tức thời và trạng thái logic đầu ra của các bộ Timer này không phụ thuộc vào chương trình và được tiến hành hoàn toàn tự động mỗi vòng quét một lần và tại thời điểm đầu vòng quét.

Thực hiện lệnh R đối với một bộ Timer có độ phân giải là 10ms đang ở trạng thái làm việc là đưa Timer về trạng thái ban đầu và xoá T-word và T-bit của Timer.

** Cập nhật Timer có độ phân giải là 100ms*

Hầu hết các bộ Timer của S7-200 là các bộ Timer có độ phân giải là 10ms. Giá trị để lưu trữ trong bộ Timer 100ms được tính tại mỗi đầu vòng quét và thời gian để tính sẽ là khoảng thời gian từ đầu vòng quét trước đó.

Việc cập nhật để thay đổi giá trị đếm tức thời của Timer chỉ được tiến hành ngay tại thời điểm có lệnh khai báo cho Timer trong chương trình. Bởi vậy quá trình cập nhật giá trị đếm tức thời không phải là quá trình tự động và không nhất thiết phải thực hiện một lần trong mỗi vòng quét ngay cả khi Timer đã được kích.

2.2.2. Bộ thời gian TON

Bộ thời gian trễ mở: sẽ đếm thời gian khi đầu vào cho phép (IN) đóng. Khi giá trị hiện thời của bộ thời gian bằng hoặc lớn hơn giá trị thời gian đặt PT (Preset Time) thì bit của bộ thời gian được đặt bằng 1.

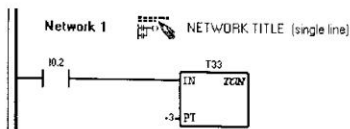
Giá trị hiện thời của bộ thời gian TON bị xoá khi đầu vào IN mở ra.

Bộ thời gian TON vẫn tiếp tục đếm thời gian sau khi đã đếm tới giá trị thời gian đặt và nó dừng đếm khi đạt tới giá trị lớn nhất: 32.767 giây.

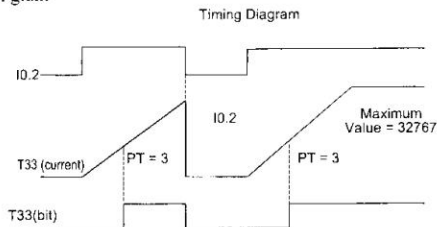
PLC S7-200 cung cấp 3 độ phân giải: 1ms, 10ms, 100ms. Cụ thể như sau:

Loại bộ thời gian	Độ phân giải	Giá trị lớn nhất	Số của bộ thời gian
TON	1ms	32.767s (0.546')	T32, T96
	10ms	327.67s (5.46')	T33 ÷ T36, T97 ÷ T100
	100ms	3276.7s (54.6')	T37 ÷ T63, T101 ÷ T255

Ví dụ: Một bộ thời gian TON T33 với độ phân giải 10ms với giá trị đặt là 3 thì sẽ đếm được khoảng thời gian là $3 \times 10\text{ms} = 30\text{ms}$ sau khi được phép hoạt động.



Giản đồ thời gian:



2.2.3. Bộ thời gian TONR

Bộ thời gian trễ đóng có nhớ: sẽ đếm thời gian khi đầu vào cho phép (IN) đóng. Nếu đầu vào cho phép (IN) mở thì TONR vẫn giữ giá trị hiện thời của nó. Khi giá trị hiện thời của bộ thời gian bằng hoặc lớn hơn giá trị thời gian đặt PT (Preset Time) thì bit của bộ thời gian được đặt bằng 1.

Có thể dùng bộ thời gian TONR để chứa một số khoảng thời gian khi đầu vào tắt hay mở. Dùng lệnh Reset (R) để xóa giá trị hiện thời của bộ thời gian TONR.

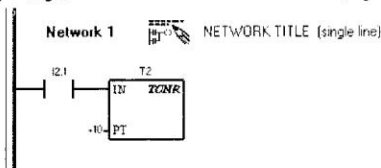
Bộ thời gian TONR vẫn tiếp tục đếm thời gian sau khi đã đếm tới giá trị thời gian đặt và nó dừng đếm khi đạt tới giá trị lớn nhất: 32.767.

Bộ thời gian TONR đếm các khoảng thời gian khác nhau. Do đó, độ phân giải (hay thời gian cơ sở) sẽ quyết định khoảng thời gian mà bộ thời gian sẽ đếm.

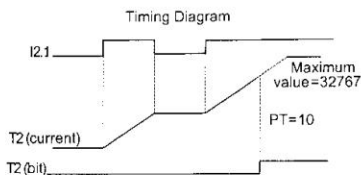
PLC S7-200 cung cấp 3 độ phân giải cho bộ thời gian TONR: 1ms, 10ms, 100ms. Cụ thể như sau:

Loại bộ thời gian	Độ phân giải	Giá trị lớn nhất	Số của bộ thời gian
TONR	1ms	32.767s (0.546')	T0, T64
	10ms	327.67s (5.46')	T1÷T4, T65÷T68
	100ms	3276.7s (54.6')	T5÷T31, T69÷T95

Ví dụ: Một bộ thời gian TONR T2 với độ phân giải 10ms, với giá trị đặt là 10 thì sẽ đếm được khoảng thời gian là $10 \times 10\text{ms} = 100\text{ms}$ sau khi được phép hoạt động.



Giản đồ thời gian:

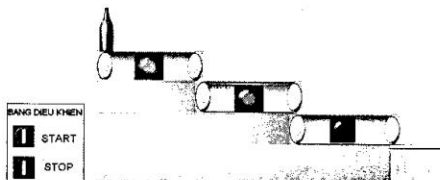


Bài tập thực hành 5:

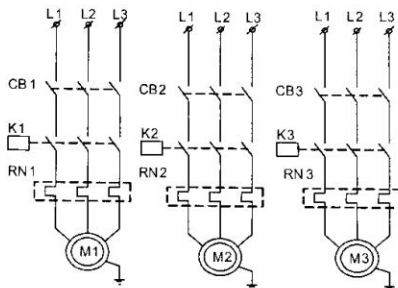
Ứng dụng bộ thời gian Timer điều khiển băng tải.

* Mô tả đề bài:

- Có 1 hệ thống băng tải gồm 3 động cơ.
- Khi khởi động nhấn START thì động cơ 1 chạy, sau 3 giây thì tự động động cơ 2 chạy, tiếp theo 3 giây kể từ động cơ 3 chạy.
- Tương ứng mỗi động cơ chạy thì có đèn sáng.
- Khi nhấn nút STOP (dừng) thì động cơ thứ 1 dừng trước, sau 5 giây thì tự động động cơ thứ 2 dừng và sau 5 giây thì tự động động cơ 3 dừng hẳn.



Sơ đồ mạch động lực:



Yêu cầu:

- Viết chương trình lập trình trên phần mềm Microwin 4.0.
- Vẽ sơ đồ kết nối phần cứng PLC.
- Nạp chương trình vào PLC và kiểm tra kết quả.

* Mục tiêu bài tập:

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển băng tải trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bản thí nghiệm.

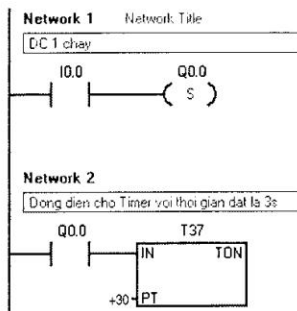
* Nội dung:

a) Quy ước các đầu vào/ra:

STT	Tiếp điểm logic	Giải thích
1	I0.0	Nút nhấn khởi động START
2	I0.1	Nút nhấn dừng STOP
3	Q0.0	Ngõ ra điều khiển động cơ kéo băng tải 1
4	Q0.1	Ngõ ra điều khiển động cơ kéo băng tải 2
5	Q0.2	Ngõ ra điều khiển động cơ kéo băng tải 3.

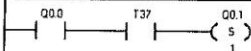
b) Chương trình lập trình:

- Viết trong LAD:



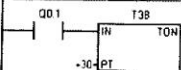
Network 3

DC 2 chạy sau 3s



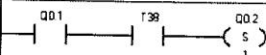
Network 4

Dừng đèn cho T38 và thời gian cài đặt là 3s



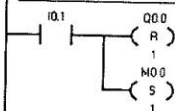
Network 5

DC 3 chạy



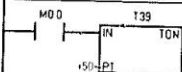
Network 6

Dừng DC 1



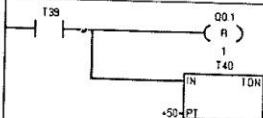
Network 7

Cài đặt T39 và thời gian đặt là 5s



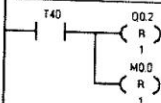
Network 8

Dừng DC 2



Network 9

Dừng DC 3



- Viết trong STL:

Network 1 Network 1.1

DC 1 chạy

ID I0.0
S Q0.0.1

Network 2

Đồng bộ cho Timer với tần số đặt là 3s

ID Q0.0
TON T37, +3s

Network 3

DC 1 chạy tại 3s

ID Q0.0
A T37
S Q0.1.1

Network 4

Đồng bộ cho T38 với thời gian đặt là 3s

ID Q0.1
TON T38, +3s

Network 5

DC 3 chạy

ID Q0.1
A T38
S Q0.2.1

Network 6

Dùng DC 1

ID I0.1
R Q0.0.1
S M0.0.1

Network 7

Card 139 với thời gian đặt là 3s

ID M0.0
TON T39, +3s

Network 8

Dùng DC 2

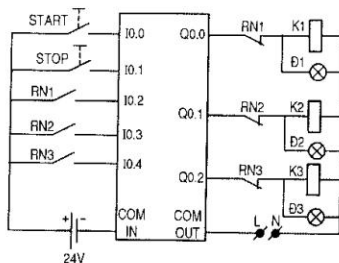
ID T39
R Q0.1.1
TON T40, +3s

Network 9

Dùng DC 3

ID T40
R Q0.2.1
R M0.0.1

c) Sơ đồ kết nối PLC:

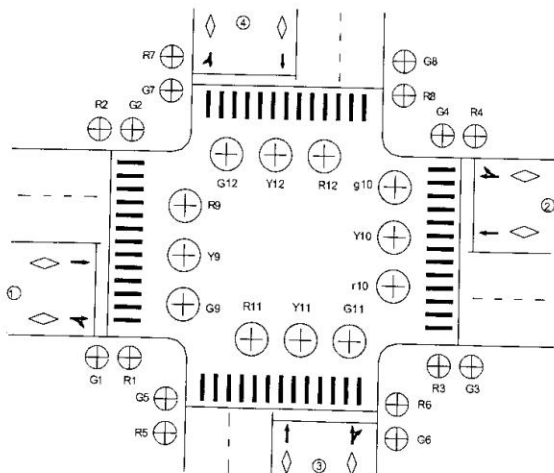


Bài tập thực hành 6:

Ứng dụng bộ thời gian Timer điều khiển hệ thống đèn tín hiệu giao thông.

* Mô tả đề bài:

Nút giao thông là nơi giao nhau của các tuyến đường: giữa các đường ô tô, giữa đường ô tô với đường sắt, giữa đường ô tô với đường thành phố và giữa các đường phố trong đô thị.



Chu kỳ đèn tín hiệu bao gồm:

Thời gian đèn xanh hướng	1 - 2.
Thời gian đèn vàng hướng	1 - 2.
Thời gian đèn đỏ hướng	1 - 2.
Thời gian đèn xanh hướng	3 - 4.
Thời gian đèn vàng hướng	3 - 4.
Thời gian đèn đỏ hướng	3 - 4.

Nguyên lý hoạt động của hệ thống:

- Khi ấn nút khởi động hệ thống (Start) thì các đèn xanh G9, G10 sáng, để báo rằng các phương tiện giao thông được đi theo các hướng 1-2, 2-1 và được rẽ trái và rẽ phải. Khi đó, các đèn đỏ R11, R12 sáng để báo rằng các phương tiện ở hướng này phải dừng lại; đèn đỏ dành cho người đi bộ R1, R2, R3, R4 sáng để báo rằng người đi bộ không được đi sang đường, ngang qua hướng 1-2; đèn xanh dành cho người đi bộ G5, G6, G7, G8 sáng để báo rằng người đi bộ được đi sang đường, ngang qua hướng 3-4.

- Sau 15 giây, các đèn xanh G9, G10 tắt, các đèn vàng Y9, Y10 sáng trong thời gian 3 giây để báo rằng các phương tiện đi theo hướng 1-2 chuẩn bị dừng lại. Đồng thời các đèn xanh dành cho người đi bộ G5, G6, G7, G8 tắt và các đèn đỏ R5, R6, R7, R8 sáng, để báo rằng người đi bộ không được đi sang đường, ngang qua hướng 3-4.

- Sau 3 giây, hiện tượng của hướng 1-2 lặp lại ở hướng 3-4.

Yêu cầu:

- Viết chương trình lập trình trên phần mềm Microwin 4.0
- Vẽ sơ đồ kết nối phần cứng PLC
- Nạp chương trình vào PLC và kiểm tra kết quả

*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

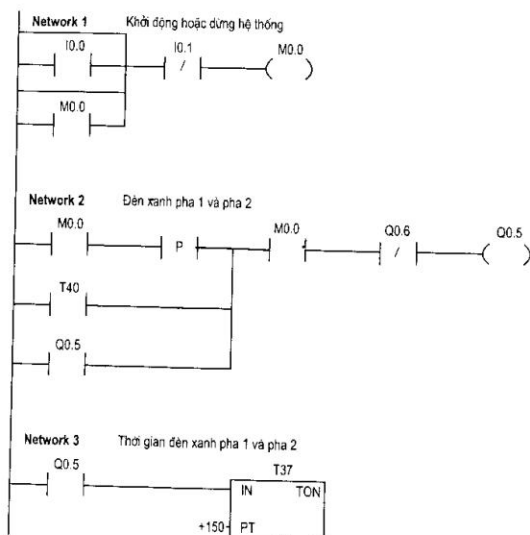
- Lập trình được chương trình điều khiển hệ thống đèn giao thông trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

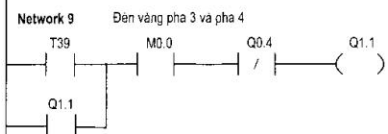
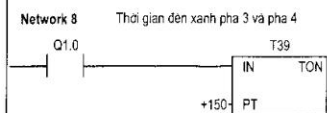
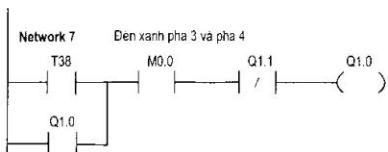
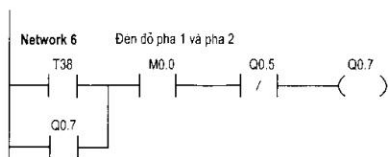
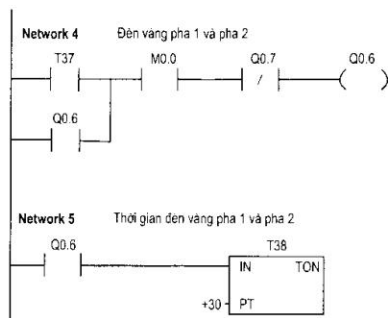
*** Nội dung:**

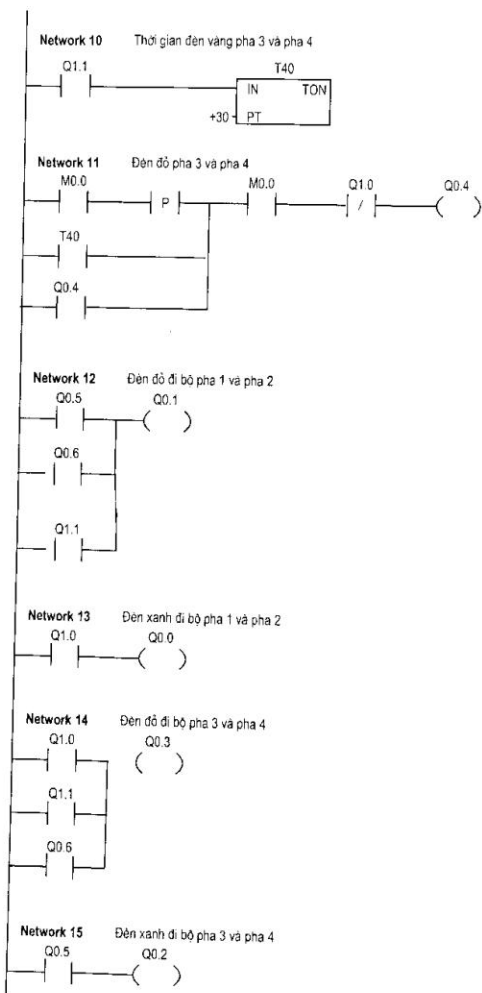
a) Quy ước các đầu vào/ra:

STT	Tiếp điểm logic	Chức năng
1	I0.0	Nút nhấn khởi động START
2	I0.1	Nút nhấn dừng STOP
3	Q0.0	Điều khiển G1, G2, G3, G4
4	Q0.1	Điều khiển R1, R2, R3, R4
5	Q0.2	Điều khiển G5, G6, G7, G8
6	Q0.3	Điều khiển R5, R6, R7, R8
7	Q0.4	Điều khiển R11, R12
8	Q0.5	Điều khiển G9, G10
9	Q0.6	Điều khiển Y9, Y10
10	Q0.7	Điều khiển Y9, R10
11	Q1.0	Điều khiển G11, G12
12	Q1.1	Điều khiển Y11, Y12

b) Chương trình lập trình:

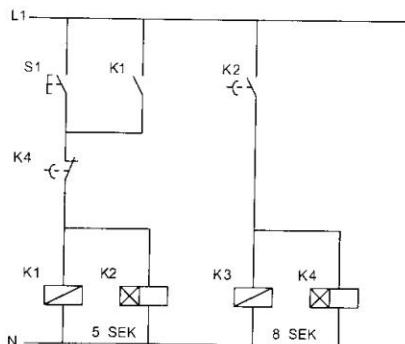






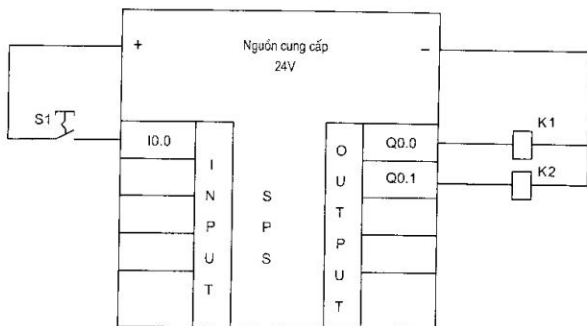
Bài tập thực hành 7:

Ứng dụng bộ thời gian Timer điều khiển đóng mạch tuần tự theo sơ đồ:



* Mô tả đề bài:

Nhấn S1, K1 có điện và tự duy trì - đồng thời rơle K2 có điện. Sau 1 khoảng thời gian 5s, tiếp điểm K2 của rơle thời gian đóng mạch cho K3, K4. Sau 1 khoảng thời gian 8s tiếp điểm K4 hở ra và toàn bộ mạch trở về trạng thái ban đầu.

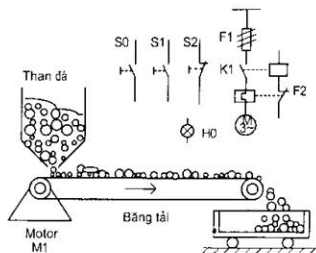


* Yêu cầu:

- Viết chương trình lập trình trên phần mềm Microwin 4.0.
- Nạp chương trình vào PLC và kiểm tra kết quả.

Bài tập thực hành 8:

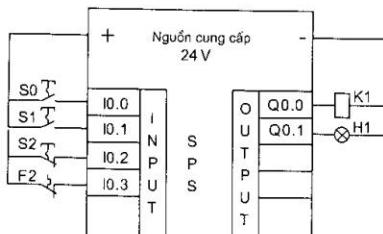
Ứng dụng bộ thời gian Timer điều khiển động cơ kéo băng tải theo sơ đồ:



* Mô tả hoạt động:

- Công tắc S0 dùng để khởi động cho thiết bị và đèn H0 chỉ báo chế độ làm việc.
- Khi nhấn S1 động cơ M1 khởi động kéo băng tải và than đá trong thùng chứa được vận chuyển theo băng tải. Khi nhấn S2 thì băng tải dừng, nó sẽ được cắt khỏi nguồn qua bộ bảo vệ quá dòng F2.
- Sau 10s xe chở than đá lại về vị trí cũ, quá trình bắt đầu lại từ đầu.

Xác định ngõ vào/ra		
Ký hiệu	Địa chỉ	Chú thích
S0	I0.0	Công tắc thường mở
S1	I0.1	Nút nhấn thường mở
S2	I0.2	Nút nhấn thường đóng
F2	I0.3	Tiếp điểm thường đóng
K1	Q0.0	Khởi động từ
H0	Q0.1	Đèn báo



** Yêu cầu:*

- Viết chương trình lập trình trên phần mềm Microwin 4.0.
- Nạp chương trình vào PLC và kiểm tra kết quả.

2.3. Bộ đếm (Counter)

2.3.1. Khái niệm

- Counter là bộ đếm thực hiện chức năng đếm sườn xung trong S7-200. Các bộ đếm của S7-200 được chia làm hai loại:

- Bộ đếm lên **CTU (Count Up)**.
- Bộ đếm lên và đếm xuống **CTUD (Counter Up/Down)**.

Số lượng bộ đếm của S7-200 CPU 224 là 256, được ký hiệu từ C0 ÷ C255.

2.3.2. Bộ đếm tiến Counter up (CTU)

- Khi đầu vào đếm tiến (CU: Count Up) nhận được chuyển tiếp từ tắt sang mở (OFF - ON) thì giá trị hiện thời của bộ đếm sẽ được tăng lên 1.

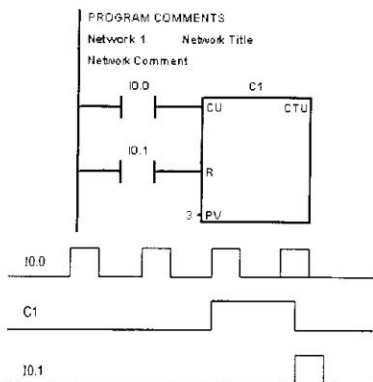
- Khi giá trị hiện thời của bộ đếm lớn hơn hoặc bằng giá trị đặt (PV: Preset Value) thì bit của bộ đếm (Counter bit) được đặt giá trị logic 1.

- Bộ đếm bị xóa về 0 khi đầu vào Reset có giá trị logic 1.

- Bộ đếm dừng đếm khi bộ đếm đạt tới giá trị cực đại (32.767)

- Ký hiệu từ: C0 ÷ C47; C80 ÷ C127.

Ví dụ:



Khi đầu vào I0.0 thay đổi trạng thái từ 0 lên 1 thì bộ đếm C1 bắt đầu đếm, I0.0 thay đổi được 3 lần thì C0 tác động, bit đầu ra C1 từ 0 lên 1. Để reset C1 ta bấm nút I0.1

2.3.3. Bộ đếm tiến/lùi Counter up down (CTUD)

- Bộ đếm tiến/lùi CTUD sẽ thực hiện việc đếm tiến khi đầu vào CU (Count Up) nhận được chuyển tiếp từ 0 lên 1 và đếm lùi khi đầu vào đếm lùi (CD: Count Down) nhận được chuyển tiếp từ 0 lên 1.

- Số sườn xung đếm được ghi vào thanh ghi 2 byte (Word) của bộ đếm gọi là thanh ghi C_Word (CV).

- Giá trị đặt PV (Preset Value) được so sánh với giá trị hiện thời mỗi khi lệnh của bộ đếm được thực hiện.

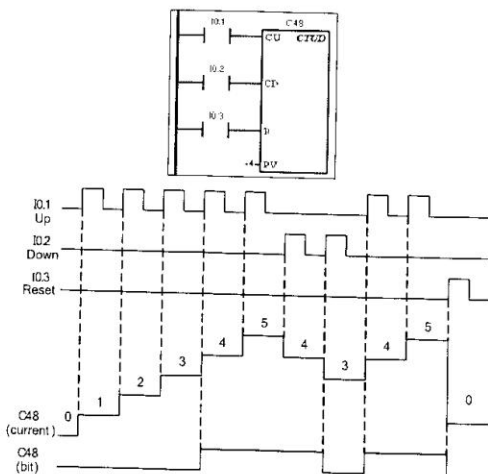
- Khi đạt tới giá trị cực đại (32.767), lệnh đếm tiến tiếp theo sẽ đưa giá trị hiện thời của bộ đếm về giá trị cực tiểu (-32767). Ngược lại, khi đã đạt tới giá trị cực tiểu, lệnh đếm lùi tiếp theo sẽ đưa giá trị cực đại vào giá trị hiện thời của bộ đếm.

- Khi giá trị hiện thời của bộ đếm lớn hơn hoặc bằng giá trị đặt PV (Preset Value) thì bit của bộ nhớ được đặt bằng 1. Nếu không, bit của bộ nhớ được xóa về 0.

- Bộ đếm bị xóa khi đầu vào Reset có giá trị logic 1.

- Ký hiệu từ: C48 ÷ C79.

Ví dụ:



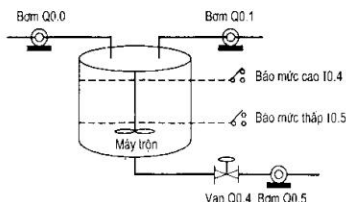
Khi đầu vào I0.1 thay đổi trạng thái từ 0 lên 1 thì bộ đếm C48 bắt đầu đếm tiến, I0.1 thay đổi được 4 lần thì C48 tác động, bit đầu ra C48 từ 0 lên 1. Khi đầu vào I0.2 thay đổi trạng thái từ 0 lên 1 thì bộ đếm C48 bắt đầu lùi, I0.2 thay đổi được 2 lần thì giá trị đếm tức thời bằng 3 nên bit đầu ra C48 từ 1 trở về 0. Để reset C48 ta bấm nút I0.3.

Bài tập thực hành 9:

Ứng dụng bộ đếm điều khiển hệ thống bình trộn sơn.

**** Mô tả đề bài:***

- Có hai đường ống để đưa hai loại sơn màu khác nhau làm cơ sở cho việc tạo ra màu sơn như mong muốn.
- Hai cảm biến được sử dụng để báo mức trong bình: báo mức cao I0.4 và báo mức thấp I0.5.
- Một thiết bị trộn được điều khiển bởi động cơ trộn (Q0.2).
- Loại sơn thứ nhất được đưa vào bình bằng máy bơm được điều khiển qua Q0.0. Loại sơn thứ hai được đưa vào bình bằng máy bơm thứ hai được điều khiển qua Q0.1.



- Sau khi hai dung dịch trong bình đạt mức cực đại ($I0.4 = 1$) thì dừng hai máy bơm và bắt đầu quá trình trộn, quá trình này được điều khiển bởi động cơ trộn (Q0.2) và thời gian trộn cần thiết là 5s.
- Sau khi trộn xong, sản phẩm được đưa ra để rót vào các hộp đựng sơn qua van (Q0.4) và máy bơm (Q0.5).
- Nếu số lần trộn đã đủ 10 lần thì dừng hệ thống lại.

**** Yêu cầu:***

- Viết chương trình lập trình trên phần mềm Microwin 4.0.
- Vẽ sơ đồ kết nối phần cứng PLC.
- Nạp chương trình vào PLC và kiểm tra kết quả.

**** Mục tiêu bài tập:***

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển bình trộn sơn trên phần mềm Microwin 4.0 theo yêu cầu bài toán.

- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

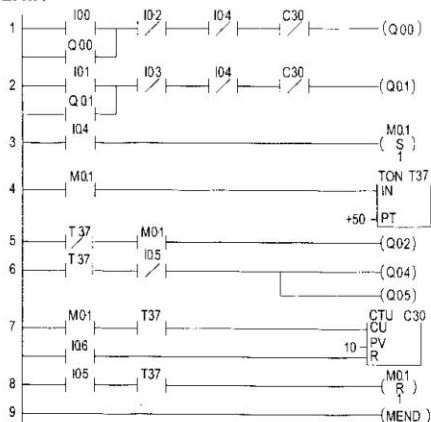
** Nội dung:*

a) Quy ước các đầu vào/ra:

STT	Tiếp điểm logic	Giải thích
1	I0.0	Nút nhấn khởi động bơm sơn loại 1
2	I0.1	Nút nhấn khởi động bơm sơn loại 2
3	I0.2	Nút nhấn dừng bơm sơn loại 1
4	I0.3	Nút nhấn dừng bơm sơn loại 2
5	I0.4	Cảm biến báo mức thấp
6	I0.5	Cảm biến báo mức cao
7	Q0.0	Máy bơm 1
8	Q0.1	Máy bơm 2
9	Q0.2	Máy trộn
10	Q0.4	Van tháo sơn ra ngoài
11	Q0.5	Máy bơm sơn ra ngoài

b) Chương trình lập trình:

- Viết trong LAD:



- Viết trong STL:

Network 1 Network Title

Network Comment	
LD	I0.0
O	Q0.0
AN	I0.2
AN	I0.4
AN	C30
=	Q0.0

Network 2

LD	I0.0
O	Q0.1
AN	I0.3
AN	I0.4
AN	C30
=	Q0.1

Network 3

LD	I0.4
S	M0.1 1

Network 4

LD	M0.1
TON	T37 + 50

Network 5

LDN	T37
A	M0.1
=	Q0.2

Network 6

LD	T37
AN	I0.5
=	Q0.4
=	Q0.5

Network 7

LD	M0.1
A	T37
LD	I0.6
CTU	C30 +10

Network 8

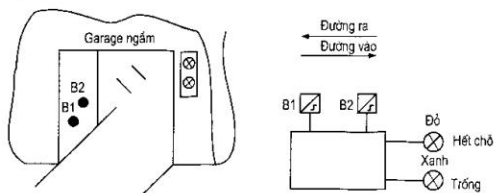
LD	I0.5
A	T37
R	M0.1 1

Bài tập thực hành 10:

Ứng dụng bộ đếm điều khiển đóng mở cửa gara xe ô tô.

* Mô tả đề bài:

- Một garage ngầm có 20 chỗ đậu xe. Ở ngõ vào có 2 đèn báo: đèn đỏ đã báo hiệu garage đã hết chỗ, đèn xanh báo hiệu garage còn trống.
 - Đường vào và đường ra chỉ cho phép một xe chạy.
- Sơ đồ công nghệ được cho như hình vẽ sau:



Xác định ngõ vào/ra		
Ký hiệu	Địa chỉ	Mô tả
B1	I0.0	CB quang 1, xe vào
B2	I0.1	CB quang 2, xe ra
H1	Q0.0	Đèn báo đỏ
H2	Q0.1	Đèn báo xanh

* Yêu cầu:

- Viết chương trình lập trình trên phần mềm Microwin 4.0.
- Vẽ sơ đồ kết nối phần cứng PLC.
- Nạp chương trình vào PLC và kiểm tra kết quả.

Bài tập thực hành 11:

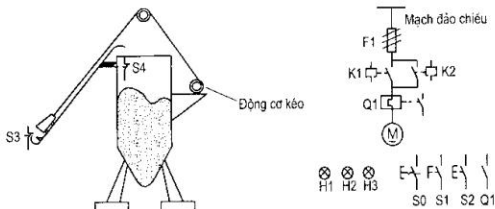
Ứng dụng bộ đếm điều khiển động cơ hệ thống kéo hàng.

* Mô tả đề bài:

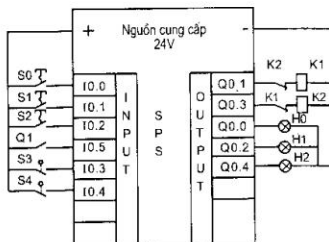
- Công tắc S0 dùng để khởi động cho thiết bị, đèn báo H0 chỉ báo chế độ làm việc.
- Khi nhấn S1, động cơ M1 có điện và xe di chuyển lên phía trên – đèn H1 chớp với tần số 1Hz.
- Khi xe lên tới vị trí trên cùng dừng phải công tắc hành trình S4 thì động cơ bị ngắt mạch.

- Nhấn S2 động cơ có điện trở lại và di chuyển xe xuống phía dưới – đèn báo H2 chớp với tần số 1Hz. Khi xe đến vị trí cuối cùng động phải công tắc hành trình S3 thì động cơ bị ngắt mạch.

- Khi động cơ M1 có sự cố đèn H0 chớp với tần số 1Hz. Quá trình mới được khởi động lại như ban đầu. Sau 10 lần thì dừng lại



Xác định ngõ vào/ra		
Ký hiệu	Địa chỉ	Chú thích
S0	I0.0	Công tắc thường mở
S1	I0.1	Nút nhấn thường mở
S2	I0.2	Nút nhấn thường đóng
S3	I0.3	Công tắc hành trình
S4	I0.4	Công tắc hành trình
Q1	I0.5	Bảo vệ động cơ
K1	Q0.1	Khởi động từ chạy phải
K2	Q0.3	Khởi động từ chạy trái
H0	Q0.0	Đèn báo
H1	Q0.2	Đèn báo
H2	Q0.4	Đèn báo



* *Yêu cầu:*

- Viết chương trình lập trình trên phần mềm Microwin 4.0
- Nạp chương trình vào PLC và kiểm tra kết quả.

CÂU HỎI VÀ BÀI TẬP

1. Câu hỏi

Câu 1. Hãy nêu các cách tạo thời gian trễ 60 phút trong PLC S7-200.

Câu 2. Nêu giới hạn lớn nhất của bộ đếm tiến CTU và đếm tiến/lùi CTUD trong PLC S7-200 CPU 224.

Câu 3. So sánh sự khác nhau giữa bộ thời gian trễ có nhớ TONR và không có nhớ TON.

Câu 4. Hãy nêu cách tạo thời gian trễ bằng cách kết hợp bộ thời gian và bộ đếm. Ví dụ: với tạo thời gian trễ 500 phút.

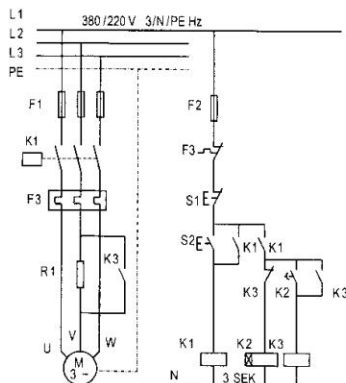
2. Bài tập

Cho các sơ đồ điều khiển động cơ như sau:

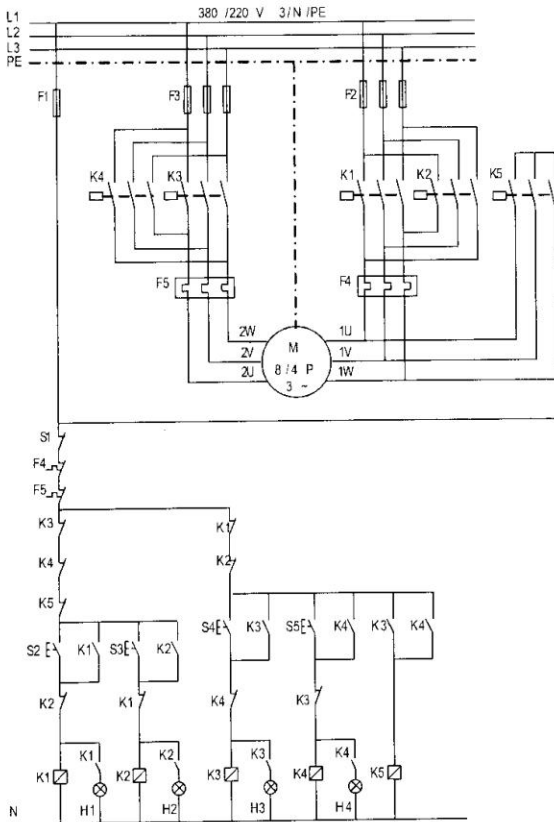
Yêu cầu:

- Nêu nguyên lý hoạt động của sơ đồ.
- Quy ước các đầu vào/ra.
- Viết chương trình ở dạng LAD và STL.
- Nạp vào PLC và chạy thử.

a)



b)



Bài 3

CÁC PHÉP TOÁN SỐ CỦA PLC

MỤC TIÊU CỦA BÀI:

- Trình bày được các phép toán so sánh, các phép toán số.
- Lập trình, kết nối, chạy thử các phép toán so sánh, phép toán số, trong các bài toán thực tế.

NỘI DUNG CỦA BÀI:

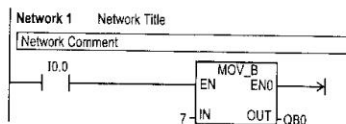
1. CHỨC NĂNG TRUYỀN DẪN

1.1. Truyền byte, Word, Doubleword

1.1.1. Truyền byte

- Lệnh truyền byte: sẽ thực hiện copy dữ liệu byte tại ngõ vào IN và truyền tới byte tại ngõ ra OUT.

Ví dụ:



Trong STL: LD I0.0

MOVB 7, QB0

- Byte đầu vào có giá trị thập phân là 7, tương đương mã nhị phân là 00000111; khi bấm nút I0.0 thì lệnh truyền byte được thực hiện, sẽ truyền đến byte đầu ra Q0.

- Sau khi thực hiện lệnh, đầu ra Q0 = 11100000; tương đương:

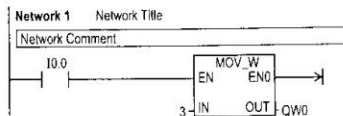
$$Q0.0 = Q0.1 = Q0.2 = 1$$

$$Q0.3 = Q0.4 = Q0.5 = Q0.6 = Q0.7 = 0$$

1.1.2. Truyền Word

- Lệnh truyền Word: copy dữ liệu Word tại ngõ vào IN và truyền tới Word tại ngõ ra OUT.

Ví dụ:



Trong STL: LD I0.0

MOVW 7, QW0

- Word đầu vào có giá trị thập phân là 3, tương đương nhị phân 00000011; khi bấm nút I0.0 thì lệnh truyền Word được thực hiện, sẽ truyền đến Word đầu ra Q0.

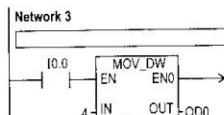
- Sau khi thực hiện lệnh, đầu ra Q0 = 00000011; tương đương

$Q1.0 = Q1.1 = 1$

$Q1.2 = Q1.3 = Q1.4 = Q1.5 = Q1.6 = Q1.7 = 0$

1.1.3. Truyền Doubleword

- Lệnh truyền Doubleword: Tương tự như lệnh truyền Word; nhưng dữ liệu ở đây được truyền là 1 từ kép (Doubleword). Khi lệnh được thực hiện sẽ copy dữ liệu tại ngõ vào IN và truyền tới tại ngõ ra OUT.



Trong STL: LD I0.0

MOVD 7, QD0

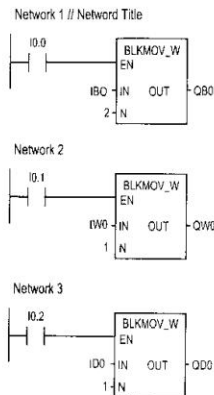
Bảng các tham số và toán hạng:

Move...	Inputs/Outputs	Operands	Data Types
Byte	IN	VB, IB, QB, MB, SB, SMB, LB, AC, Constant, *VD, *AC, *LD	BYTE
	OUT	VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD	BYTE
Word	IN	VW, IW, QW, MW, SW, SMW, LW, T, C, AIW, Constant, AC, *VD, *AC, *LD	WORD, INT
	OUT	VW, T, C, IW, QW, SW, MW, SMW, LW, AC, AQW, *VD, *AC, *LD	WORD, INT
Double Word	IN	VD, ID, QD, MD, SD, SMD, LD, HC, &VB, &IB, &QB, &MB, &SB, &T, &C, AC, Constant, *VD, *AC, *LD	DWORD, DINT
	OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	DWORD, DINT
Real	IN	VD, ID, QD, MD, SD, SMD, LD, AC, Constant, *VD, *AC, *LD	REAL
	OUT	VD, ID, QD, MD, SD, SMD, LD, AC, *VD, *AC, *LD	REAL

1.2. Truyền một vùng nhớ dữ liệu

- Phép truyền Block Move Byte, Block Move Word, Block Move Doubleword sẽ thực hiện truyền một số lượng byte (N) có địa chỉ byte đầu tại ngõ vào IN sang vùng nhớ có địa chỉ đầu tại ngõ ra OUT. N là số lượng byte có giới hạn từ 1 đến 255.

Ví dụ:



Trong STL:

Network 1	Network Title
Network Comment	
LD	I0.0
BMB	IB0, QB0, 2
Network 2	
LD	I0.1
BMW	IW0, QW0, 1
Network 3	
LD	I0.2
BMD	ID0, QD0, 1

- Khi bấm I0.0, lệnh truyền vùng dữ liệu Block Move Byte thực hiện truyền 2 byte kể từ byte đầu vào IB0 đến byte đầu ra QB0.

- Khi bấm I0.1, lệnh truyền vùng dữ liệu Block Move Word thực hiện truyền 1 word kể từ word đầu vào IW0 đến word đầu ra QW0.

- Khi bấm I0.2, lệnh truyền vùng dữ liệu Block Move Doubleword thực hiện truyền 1 doubleword kể từ doubleword đầu vào ID0 đến doubleword đầu ra QD0.

Bài tập thực hành 1:

Lập trình điều khiển 8 bóng đèn nháy trên PLC S7-200 theo yêu cầu sau:

- Ấn nút khởi động thì đèn 1, 3, 5, 7 sáng.
- Sau 10s thì đèn 2, 4, 6, 8 sáng; sau 5s nữa thì 8 đèn đều tắt.
- Trong quá trình hoạt động có thể ấn nút dừng bất kỳ lúc nào để dừng.

*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển các bóng đèn nháy trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

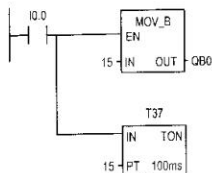
*** Nội dung:**

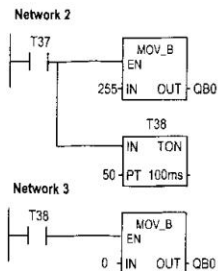
a) Quy ước các đầu vào/ra:

STT	Tiếp điểm logic	Chức năng
1	I0.0	Nút bấm thường mở
2	Q0.0	Đèn 1
3	Q0.1	Đèn 2
4	Q0.2	Đèn 3
5	Q0.3	Đèn 4
6	Q0.4	Đèn 5
7	Q0.5	Đèn 6
8	Q0.6	Đèn 7
9	Q0.7	Đèn 8

b) Chương trình lập trình:

Network 1 // Network Title





Trong STL:

Network 1 **Network**

Network Comment		
LD	I0.0	
MOVB	15.	QB0
TON	T37.	100

Network 2

LD	T37	
MOVB	255.	QB0
TON	T38.	50

Network 3

LD	T38	
MOVB	0.	QB0

2. CHỨC NĂNG SO SÁNH, DỊCH CHUYỂN, CHUYỂN ĐỔI

2.1. Chức năng so sánh

- Các phép so sánh có thể sử dụng là so sánh =, <>, >, >=, <, <= và chỉ có thể áp dụng cho byte, số nguyên I, số nguyên kép DI và số thực R.
- Dữ liệu tại ngõ vào IN1 được so sánh với dữ liệu tại ngõ vào IN2.
- Trong soạn thảo LAD thì tiếp điểm sẽ ON khi thoả mãn điều kiện so sánh.

2.1.1. So sánh byte

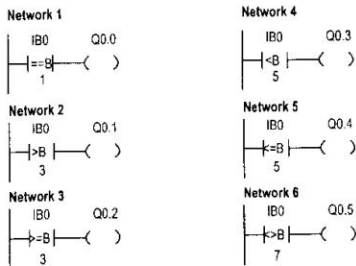
- Lệnh so sánh byte dùng để so sánh 02 giá trị dạng byte được định địa chỉ bởi hai toán hạng ở đầu vào của lệnh: [IN1] và [IN2].
- Có tất cả 06 phép so sánh có thể được thực hiện:
 - + So sánh bằng $[IN1] = [IN2]$

- + So sánh lớn hơn hoặc bằng $[IN1] \geq [IN2]$
- + So sánh nhỏ hơn hoặc bằng $[IN1] \leq [IN2]$
- + So sánh lớn hơn $[IN1] > [IN2]$
- + So sánh nhỏ hơn $[IN1] < [IN2]$
- + So sánh khác $[IN1] \neq [IN2]$.

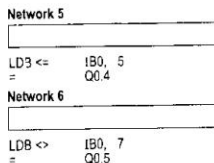
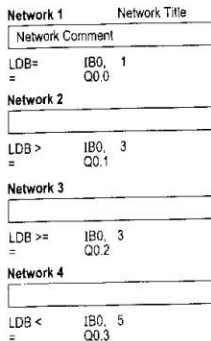
- Các byte được đem so sánh là những giá trị không dấu.

- Trong LAD, lệnh này có dạng một công tắc và công tắc đó đóng (ON) khi điều kiện đem so sánh có giá trị đúng. Trong FBD, đầu ra sẽ có giá trị 1 nếu điều kiện đem so sánh là đúng. Trong STL, lệnh được thực hiện sẽ ghi giá trị 1 vào đỉnh ngăn xếp (với những lệnh Load) hoặc thực hiện phép toán logic AND hay OR (tùy theo lệnh cụ thể) giá trị 1 với đỉnh ngăn xếp nếu điều kiện so sánh đúng.

Ví dụ:



Trong STL:



- Network 1: Lệnh so sánh bằng, nếu $IB0 = 1$ thì đầu ra $Q0.0 = 1$

Bit	I0.7	I0.6	I0.5	I0.4	I0.3	I0.2	I0.1	I0.0
Value	0	0	0	0	0	0	0	1

- Network 2: Lệnh so sánh lớn hơn, nếu $IB0 > 3$ thì đầu ra $Q0.1 = 1$

Bit	I0.7	I0.6	I0.5	I0.4	I0.3	I0.2	I0.1	I0.0
Value	0	0	1	0	0	0	1	1

- Network 3: Lệnh so sánh lớn hơn hoặc bằng, nếu $IB0 \geq 3$ thì đầu ra $Q0.2 = 1$

Bit	I0.7	I0.6	I0.5	I0.4	I0.3	I0.2	I0.1	I0.0
Value	0	0	0	0	1	0	1	1

- Network 4: Lệnh so sánh nhỏ hơn, nếu $IB0 < 5$ thì đầu ra $Q0.3 = 1$

Bit	I0.7	I0.6	I0.5	I0.4	I0.3	I0.2	I0.1	I0.0
Value	0	0	0	0	0	0	0	1

- Network 5: Lệnh so sánh nhỏ hơn hoặc bằng, nếu $IB0 \leq 5$ thì đầu ra $Q0.4 = 1$

Bit	I0.7	I0.6	I0.5	I0.4	I0.3	I0.2	I0.1	I0.0
Value	0	0	0	0	0	1	0	1

- Network 6: Lệnh so sánh khác nhau, nếu $IB0 \neq 7$ thì đầu ra $Q0.5 = 1$

Bit	I0.7	I0.6	I0.5	I0.4	I0.3	I0.2	I0.1	I0.0
Value	0	0	0	0	1	0	1	1

2.1.2. So sánh số nguyên I

- Lệnh so sánh Integer dùng để so sánh 02 giá trị dạng I được định địa chỉ bởi hai toán hạng ở đầu vào của lệnh: $[IN1]$ và $[IN2]$.

- Có tất cả 06 phép so sánh có thể được thực hiện:

- + So sánh bằng $[IN1] = [IN2]$
- + So sánh lớn hơn hoặc bằng $[IN1] \geq [IN2]$
- + So sánh nhỏ hơn hoặc bằng $[IN1] \leq [IN2]$
- + So sánh lớn hơn $[IN1] > [IN2]$
- + So sánh nhỏ hơn $[IN1] < [IN2]$
- + So sánh khác $[IN1] \neq [IN2]$.

- Các byte được đem so sánh là những giá trị nằm trong khoảng -32767 đến +32767.

- Trong LAD, lệnh này có dạng một công tắc và công tắc đó đóng (ON) khi điều kiện đem so sánh có giá trị đúng. Trong FBD, đầu ra sẽ có giá trị 1 nếu điều kiện đem so sánh là đúng. Trong STL, lệnh được thực hiện sẽ ghi giá trị 1 vào đỉnh ngăn xếp (với những lệnh Load) hoặc thực hiện phép toán logic AND hay OR (tùy theo lệnh cụ thể) giá trị 1 với đỉnh ngăn xếp nếu điều kiện so sánh đúng.

Ví dụ:

- Network 1: Khởi tạo bộ thời gian T37, giá trị tức thời của bộ Timer là số thực sẽ được so sánh.

- Network 2: Lệnh so sánh bằng, nếu $T37 = 100$ thì đầu ra $Q0.0 = 1$

- Network 3: Lệnh so sánh lớn hơn, nếu $T37 > 200$ thì đầu ra $Q0.1 = 1$

- Network 4: Lệnh so sánh lớn hơn hoặc bằng, nếu $T37 \geq 200$ thì đầu ra $Q0.2 = 1$

- Network 5: Lệnh so sánh nhỏ hơn, nếu $T37 < 300$ thì đầu ra $Q0.3 = 1$

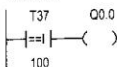
- Network 6: Lệnh so sánh nhỏ hơn hoặc bằng, nếu $T37 \leq 300$ thì đầu ra $Q0.4 = 1$

- Network 7: Lệnh so sánh khác nhau, nếu $T37 \neq 500$ thì đầu ra $Q0.5 = 1$

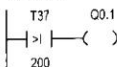
Network 1



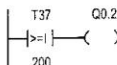
Network 2



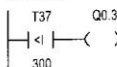
Network 3



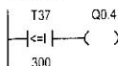
Network 4



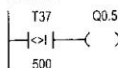
Network 5



Network 6



Network 7



Viết trong STL:

Network 1		Network
Network Comment		
LD	I0.0	
TON	T37,	500
Network 2		
LDW =	T37,	100
=	Q0.0	
Network 3		
LDW >	T37,	200
=	Q0.1	
Network 4		
LDW >=	T37,	200
=	Q0.2	
Network 5		
LDW <	T37,	300
=	Q0.3	
Network 6		
LDW <=	T37,	300
=	Q0.4	
Network 7		
LDW <>	T37,	500
=	Q0.5	

2.1.3. So sánh từ kép D

- Lệnh so sánh Double Word dùng để so sánh 02 giá trị dạng D được định địa chỉ bởi hai toán hạng ở đầu vào của lệnh: [IN1] và [IN2].

- Có tất cả 06 phép so sánh có thể được thực hiện:

- + So sánh bằng [IN1] == [IN2]
- + So sánh lớn hơn hoặc bằng [IN1] >= [IN2]
- + So sánh nhỏ hơn hoặc bằng [IN1] <= [IN2]
- + So sánh lớn hơn [IN1] > [IN2]
- + So sánh nhỏ hơn [IN1] < [IN2]
- + So sánh khác [IN1] <> [IN2].

- Các từ kép D được đem so sánh là những giá trị nằm trong khoảng:

-2.147.483.647 đến + 2.147.483.647

- Trong LAD, lệnh này có dạng một công tắc và công tắc đó đóng (ON) khi điều kiện đem so sánh có giá trị đúng. Trong FBD, đầu ra sẽ có giá trị 1 nếu điều kiện đem so sánh là đúng. Trong STL, lệnh được thực hiện sẽ ghi giá trị 1 vào đỉnh ngăn xếp (với những lệnh Load) hoặc thực hiện phép toán logic AND hay OR (tùy theo lệnh cụ thể) giá trị 1 với đỉnh ngăn xếp nếu điều kiện so sánh đúng.

2.1.4. So sánh số nguyên thực R

- Lệnh so sánh số thực Real dùng để so sánh 02 giá trị dạng R được định địa chỉ bởi hai toán hạng ở đầu vào của lệnh: [IN1] và [IN2].

- Có tất cả 06 phép so sánh có thể được thực hiện:

+ So sánh bằng [IN1] == [IN2]

+ So sánh lớn hơn hoặc bằng [IN1] >= [IN2]

+ So sánh nhỏ hơn hoặc bằng [IN1] <= [IN2]

+ So sánh lớn hơn [IN1] > [IN2]

+ So sánh nhỏ hơn [IN1] < [IN2]

+ So sánh khác [IN1] <> [IN2].

- Các R được đem so sánh là những giá trị có dấu theo kiểu dấu phẩy động.

- Trong LAD, lệnh này có dạng một công tắc và công tắc đó đóng (ON) khi điều kiện đem so sánh có giá trị đúng. Trong FBD, đầu ra sẽ có giá trị 1 nếu điều kiện đem so sánh là đúng. Trong STL, lệnh được thực hiện sẽ ghi giá trị 1 vào đỉnh ngăn xếp (với những lệnh Load) hoặc thực hiện phép toán logic AND hay OR (tùy theo lệnh cụ thể) giá trị 1 với đỉnh ngăn xếp nếu điều kiện so sánh đúng.

Bài tập thực hành 2:

Lập trình điều khiển 4 bóng đèn trên PLC S7-200 theo yêu cầu sau:

- Ấn nút khởi động để khởi động chương trình.

- Sau 5s thì đèn 1 sáng; sau 10s thì đèn 2 sáng; sau 15s thì đèn 3 sáng; sau 20s thì đèn 4 sáng. Khi nào thời gian quá 10 phút thì 4 đèn đều tắt.

- Trong quá trình hoạt động có thể ấn nút dừng bất kỳ lúc nào để dừng.

*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển các bóng đèn trên phần mềm Microwin 4.0 theo yêu cầu bài toán.

- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

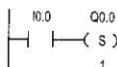
* Nội dung:

a) Quy ước các đầu vào/ra:

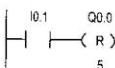
STT	Tiếp điểm logic	Chức năng
1	I0.0	Nút bấm khởi động
2	I0.1	Nút bấm dừng
3	Q0.1	Đèn 1
4	Q0.2	Đèn 2
5	Q0.3	Đèn 3
6	Q0.4	Đèn 4

b) Chương trình lập trình:

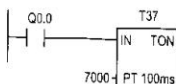
Network 1 // Network Title



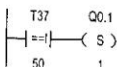
Network 2



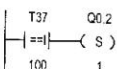
Network 3

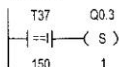
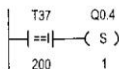
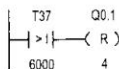


Network 4



Network 5



Network 6**Network 7****Network 8**

Viết trong STL:

Network 1		Network Title	
Network Comment			
LD	I0.0		
S	Q0.0	1	
Network 2			
LD	I0.1		
R	Q0.0	5	
Network 3			
LD	Q0.0		
TON	T37,	7000	
Network 4			
LDW =	T37,	50	
S	Q0.1	1	
Network 5			
LDW =	T37,	100	
S	Q0.2,	1	
Network 6			
LDW =	T37,	150	
S	Q0.3,	1	
Network 7			
LDW =	T37,	200	
S	Q0.4,	1	
Network 8			
LDW >	T37,	6000	
S	Q0.1,	4	

2.2. Chức năng dịch chuyển

- Các lệnh dịch, mỗi khi được thực hiện, sẽ dịch các bit trong khối dữ liệu ở đầu vào IN sang trái hoặc sang phải với số lần dịch được xác định ở đầu vào N và đưa kết quả tới đầu ra OUT.

- Các lệnh dịch sẽ đưa các số 0 vào vị trí trống khi một bit được dịch ra ngoài.

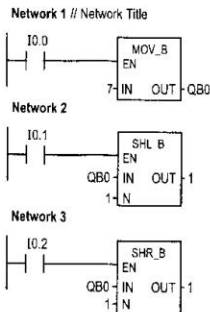
- Các lệnh dịch có thể dịch được các dạng dữ liệu là byte, từ, từ kép (byte, Word, Doubleword).

2.2.1. Lệnh dịch byte

- Lệnh dịch byte gồm có lệnh dịch phải byte SHR_B và lệnh dịch trái byte SHL_B

- Các lệnh SHR_B và SHL_B sẽ dịch dữ liệu tại byte ngõ vào IN sang phải hoặc sang trái với số vị trí dịch được nhập tại N, kết quả được chứa vào byte ngõ ra OUT. Ở các lệnh SHIFT thì tại vị trí các bit bị dịch sẽ lấp đầy bằng số 0. Số vị trí bit cần dịch được nhập tại ngõ N ≤ 8 . Việc ảnh hưởng đến các Bit nhớ đặc biệt (SM1.0 và SM1.1).

Ví dụ:



Trong STL:

Network 1	Network Title
Network Comment	
LD	I0.0
MOVB	7, QB0
Network 2	
LD	I0.1
SLB	QB.0, 1
Network 3	
LD	I0.2
SRB	QB0, 1

- Trong chương trình trên, Network 1 có nhiệm vụ tạo 1 giá trị ban đầu cho byte đầu ra Q0 là 111.

- Khi ấn nút I0.1 thì dữ liệu byte Q0 sẽ được dịch sang trái 1 vị trí.

- Khi ấn nút I0.2 thì dữ liệu byte Q0 sẽ được dịch sang phải 1 vị trí

Bit	I0.7	I0.6	I0.5	I0.4	I0.3	I0.2	I0.1	I0.0
Ban đầu	0	0	0	0	0	1	1	1
SHL_B	0	0	0	0	1	1	1	0
SHR_B	0	0	0	0	0	1	1	1

2.2.2. Lệnh dịch Word

- Lệnh dịch Word gồm có lệnh dịch phải Word SHR_W và lệnh dịch trái Word SHL_W

- Các lệnh SHR_W và SHL_W sẽ dịch dữ liệu tại Word ngõ vào IN sang phải hoặc sang trái với số vị trí dịch được nhập tại N, kết quả được chứa vào Word có địa chỉ tại ngõ ra OUT. Tại vị trí các Bit bị dịch sẽ lấp đầy bằng số 0. Số vị trí Bit cần dịch được nhập tại ngõ N ≤ 16 . Việc ảnh hưởng đến các Bit nhớ đặc biệt (SM1.0 và SM1.1).

2.2.3. Lệnh dịch Doubleword

- Lệnh dịch Doubleword gồm có lệnh dịch phải SHR_DW và dịch trái SHL_DW.

- Các lệnh SHR_DW và SHL_DW sẽ dịch dữ liệu tại Doubleword ngõ vào IN sang phải hoặc sang trái với số vị trí dịch được nhập tại N, kết quả được chứa vào Doubleword có địa chỉ tại ngõ ra OUT. Tại vị trí các bit bị dịch sẽ được lấp đầy bằng số 0. Số vị trí Bit cần dịch được nhập tại ngõ N ≤ 32 . Việc ảnh hưởng đến các Bit nhớ đặc biệt (SM1.0 và SM1.1).

Bài tập thực hành 3:

Lập trình điều khiển 8 bóng đèn trên PLC S7-200 theo yêu cầu sau:

- Ấn nút khởi động để khởi động chương trình.
- Các bóng đèn sáng/tắt xen kẽ nhau, sau 5s thì đổi trạng thái.
- Quá trình lặp lại tương tự tuần hoàn.
- Trong quá trình hoạt động có thể ấn nút dừng bất kỳ lúc nào để dừng.

* Mục tiêu bài tập:

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển các bóng đèn trên phần mềm Microwin 4.0 theo yêu cầu bài toán.

- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

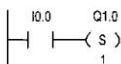
* Nội dung:

a) Quy ước các đầu vào/ra:

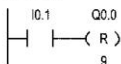
STT	Tiếp điểm logic	Chức năng
1	I0.0	Nút bấm khởi động
2	I0.1	Nút bấm dừng
3	Q0.0	Đèn 1
4	Q0.1	Đèn 2
5	Q0.2	Đèn 3
6	Q0.3	Đèn 4
7	Q0.4	Đèn 5
8	Q0.5	Đèn 6
9	Q0.6	Đèn 7
10	Q0.7	Đèn 8

b) Chương trình lập trình:

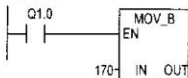
Network 1 // Network Title



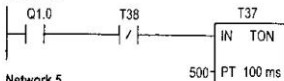
Network 2



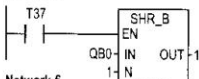
Network 3



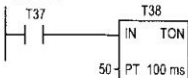
Network 4



Network 5



Network 6



Viết trong STL:

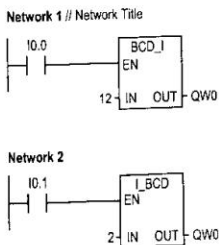
Network 1		Network Title
Network Comment		
LD	I0.0	
S	Q1.0,	1
Network 2		
LD	I0.1	
R	Q0.0,	9
Network 3		
LD	Q1.0	
MOVB	170,	QB0
Network 4		
LD	Q1.0,	
AN	T38	
TON	T37,	50
Network 5		
LD	T37	
SRB	QB0,	1
Network 6		
LD	T37	
TON	T38,	50

2.3. Chức năng chuyển đổi

2.3.1. Chuyển đổi số BCD_I và I_BCD

- Lệnh chuyển đổi số BCD sang số Integer (BCD_I) sẽ thực hiện việc chuyển đổi số BCD tại ngõ vào IN sang giá trị số nguyên I và chứa kết quả vào địa chỉ xác định tại ngõ ra OUT. Giá trị có thể nhập tại ngõ IN từ 0 đến 9999BCD.

Ví dụ:



Network 1		Network Title
Network Comment		
LD	I0.0	
MOVW	12,	QW0
BCDI	QW0	
Network 2		
LD	I0.1	
MOVW	2,	QW0
IBCD	QW0	

- Lệnh chuyển đổi số I sang số BCD sẽ thực hiện việc chuyển đổi số I tại ngõ vào IN sang giá trị số BCD và chứa kết quả vào địa chỉ xác định tại ngõ ra OUT. Giá trị có thể nhập tại ngõ IN từ 0 đến 9999 Integer.

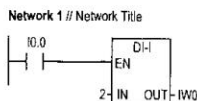
Khi xảy ra lỗi chuyển đổi thì trạng thái ENO = 0.

2.3.2. Chuyển đổi số nguyên kép DI sang số thực R và số nguyên I

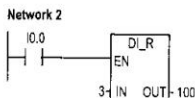
- Lệnh chuyển đổi một số nguyên kép DI 32 Bit sang một số thực R và đặt kết quả vào địa chỉ được xác định tại ngõ ra OUT.

- Lệnh chuyển đổi số DI_I chuyển đổi giá trị số DI tại ngõ vào IN sang một giá trị số nguyên I và đặt kết quả tại ngõ OUT, nếu phép chuyển đổi bị tràn (kết quả lớn hơn khả năng chứa của ngõ OUT) thì ngõ ra không thay đổi và trạng thái ENO = 0.

Ví dụ:



Network 1	Network Title
Network Comment	
LD	I0.0
DTI	2, IW0



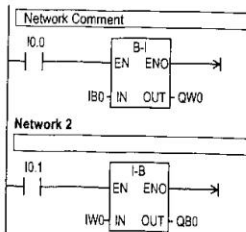
Network 2	Network Title
Network Comment	
LD	I0.0
ITD	3, I00

2.3.3. Chuyển đổi byte sang số nguyên I và số nguyên I sang byte

- Lệnh chuyển đổi B_I chuyển đổi dữ liệu chứa trong byte có địa chỉ tại ngõ IN sang giá trị số nguyên, kết quả chứa vào biến xác định tại ngõ ra OUT.

- Lệnh chuyển đổi I_B chuyển đổi giá trị chứa trong word có địa chỉ tại ngõ IN sang giá trị byte, kết quả chứa vào biến xác định tại ngõ ra OUT. Các số nguyên có thể chuyển đổi là từ 0 đến 255.

Ví dụ:



Network 1	Network Title
Network Comment	
LD	I0.0
BTI	IB0, QW0

Network 2	Network Title
Network Comment	
LD	I0.1
ITB	IW0, QB0

- Network 1 sẽ chuyển đổi dữ liệu ở byte I0 sang số thực I lưu ở địa chỉ đầu ra Q0.
- Network 2 sẽ chuyển đổi dữ liệu ở số thực I lưu ở địa chỉ đầu vào IW0 sang byte đầu ra QB0.

2.4. Chức năng toán học

Trong LAD, bốn hộp toán (math box) thực hiện các phép tính cộng, trừ 16-bit và 32-bit. Hộp nhân (multiply box) nhân hai số nguyên 16-bit và kết quả là một số nguyên 32-bit. Hộp chia (divide box) chia hai số 16-bit, thương là một số 16-bit và dư cũng 16-bit được nạp vào từ ngay trước. Nếu lập trình bằng LAD, có thể tiết kiệm ô nhớ bằng cách sử dụng đầu vào IN1 đồng thời cũng là đầu ra OUT.

Trong STL, lệnh thực hiện bốn phép tính số học được quy định cho toán hạng 16-bit và 32-bit. Hộp nhân thực hiện phép nhân hai số nguyên 16-bit và tích số là một số nguyên 32-bit. Lệnh chia thực hiện phép chia một số nguyên 16-bit với 16-bit cuối của một số nguyên 32-bit. Kết quả là một giá trị từ kép (32-bit) trong đó từ thấp (từ bit 0 đến bit 15) là thương số và từ cao (từ bit 16 đến 31 bit) là số dư của phép tính.

2.4.1. Phép cộng và phép trừ

- Phép cộng số nguyên 16 bit:

ADD_I (LAD)
+ I (STL)

Lệnh thực hiện phép cộng các số nguyên 16-bit IN1 và IN2. Trong LAD kết quả là một số nguyên 16-bit được ghi vào OUT, tức là:

$$IN1 + IN2 = OUT$$

Trong STL, kết quả cũng là một giá trị 16-bit nhưng được ghi lại vào IN2, tức là:

$$IN1 + IN2 = IN2$$

- Phép trừ số nguyên 16 bit:

SUB_I (LAD)
- I (STL)

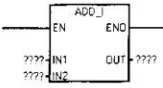
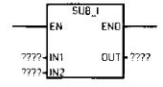
Lệnh thực hiện phép trừ các số nguyên 16-bit IN1 và IN2. Trong LAD kết quả là một số nguyên 16-bit và được ghi vào OUT, tức là:

$$IN1 - IN2 = OUT$$

Trong STL, kết quả cũng là một giá trị 16-bit nhưng được ghi lại vào IN2, tức là:

$$IN1 - IN2 = IN2$$

Cú pháp dùng lệnh cộng, trừ hai số nguyên 16-bit trong LAD và STL như sau:

LAD	STL	Toán hạng
	$+ I \text{ IN1 IN2}$	IN1, IN2 VW, T, C, QW, MW, SMW, (word) AC, AIW, hằng số, *VD, *AC
	$- I \text{ IN1 IN2}$	OUT VW, T, C, QW, MW, SMW, (word) AC, *VD, *AC

- Phép cộng số nguyên 32 bit:

ADD_DI (LAD)

+ D (STL)

Lệnh thực hiện phép cộng các số nguyên 32-bit IN1 và IN2. Trong LAD kết quả là một số nguyên 32-bit được ghi vào OUT, tức là $IN1 + IN2 = OUT$.

Trong STL, kết quả cũng là một giá trị 32-bit nhưng được ghi lại vào IN2, tức là $IN1 + IN2 = IN2$.

- Phép trừ số nguyên 32 bit:

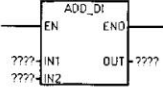
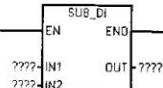
SUB_DI (LAD)

- D (STL)

Lệnh thực hiện phép trừ các số nguyên 32-bit IN1 và IN2. Trong LAD kết quả là một số nguyên 32-bit và được ghi vào OUT, tức là $IN1 - IN2 = OUT$.

Trong STL, kết quả cũng là một giá trị 32-bit nhưng được ghi lại vào IN2, tức là $IN1 - IN2 = IN2$.

Cú pháp dùng lệnh cộng, trừ hai số nguyên 32-bit trong LAD và STL như sau:

LAD	STL	Toán hạng
	$+ D \text{ IN1 IN2}$	IN1, IN2 VW, T, C, QW, MW, SMW, (từ kép) AC, AIW, hằng số, *VD, *AC
	$- D \text{ IN1 IN2}$	OUT VW, T, C, QW, MW, SMW, (từ kép) AC, *VD, *AC

- Phép cộng số thực:

ADD_R (LAD)

+ R (STL)

Lệnh thực hiện phép cộng các số thực 32-bit IN1 và IN2. Trong LAD kết quả là một số thực 32-bit được ghi vào OUT, tức là $IN1 + IN2 = OUT$.

Trong STL, kết quả cũng là một giá trị thực 32-bit nhưng được ghi lại vào IN2, tức là $IN1 + IN2 = IN2$.

- Phép trừ số thực:

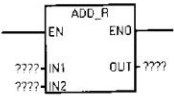
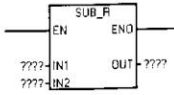
SUB_R (LAD)

- R (STL)

Lệnh thực hiện phép trừ các số thực 32-bit IN1 và IN2. Trong LAD kết quả là một số thực 32-bit và được ghi vào OUT, tức là $IN1 - IN2 = OUT$.

Trong STL, kết quả cũng là một giá trị thực 32-bit nhưng được ghi lại vào IN2, tức là $IN1 - IN2 = IN2$.

Cú pháp dùng lệnh cộng, trừ hai số thực 32-bit trong LAD và STL như sau:

LAD	STL	Toán hạng
	+ R IN1 IN2	IN1, IN2 VD, ID, QD, MD, SMD, (từ kép) AC, HC, hằng số, *VD, *AC OUT VD, ID, QD, SMD, AC (từ kép) *VD, *AC
	- R IN1 IN2	

2.4.2. Phép nhân và phép chia

- Phép nhân MUL:

Trong LAD: Lệnh thực hiện phép nhân hai số nguyên 16-bit IN1 và IN2 và cho ra kết quả 32-bit chứa trong từ kép OUT (4 byte).

Trong STL: Lệnh thực hiện phép nhân giữa số nguyên 16-bit n1 và số nguyên chứa trong từ thấp (từ bit 0 đến bit 15) của toán hạng 32-bit n2 (4 byte). Kết quả 32-bit được ghi lại vào n2.

Cú pháp dùng lệnh nhân hai số nguyên trong LAD và STL như sau:

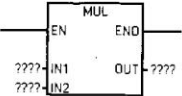
LAD	STL	Toán hạng
	MUL n1 n2	n1, IN1, IN2 VW, T, C, IW, QW, MW (word) AC, AIW, hằng số, *VD, *AC n2, OUT VD, ID, QD, SMD, AC (từ kép) *VD, *AC

- Phép chia DIV

Trong LAD: Lệnh thực hiện phép chia số nguyên 16-bit IN1 cho số nguyên 16-bit IN2. Kết quả 32-bit chứa trong từ kép OUT (4 byte) gồm thương số ghi trong mảng 16-bit từ bit 0 đến bit 15 (từ thấp) và phần dư cũng 16 bit ghi trong mảng từ bit 16 đến bit 31 (từ cao).

Trong STL: Lệnh thực hiện phép chia số nguyên 16-bit n1 cho số nguyên 16-bit nằm trong từ thấp (từ bit 0 đến bit 15) của toán hạng 32-bit n2 (4 byte). Kết quả 32-bit được ghi lại vào n2 bao gồm thương số ghi trong mảng 16-bit từ bit 0 đến bit 15 (từ thấp) và phần dư ghi trong mảng 16 bit từ bit 16 đến bit 31 (từ cao).

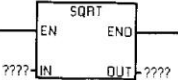
Cú pháp dùng lệnh chia hai số nguyên trong LAD và STL như sau:

LAD	STL	Toán hạng
	DIV n1 n2	n1, IN1, IN2, VW, T, C, IW, QW, MW (word) AC, AIW, hằng số, *VD, *AC n2, OUT VD, ID, QD, SMD, AC (từ kép) *VD, *AC

2.4.3. Phép lấy căn bậc 2 SQRT

Lệnh thực hiện phép lấy căn bậc hai của số thực 32-bit IN. Kết quả cũng là một số 32-bit được ghi vào từ kép OUT (4 byte).

Cú pháp dùng lệnh chia hai số thực trong LAD và STL:

LAD	STL	Toán hạng
	SQRT IN OUT	IN VD, ID, QD, MD, SMD, (từ kép) AC, HC, hằng số, *VD, *AC OUT VD, ID, QD, SMD, AC (từ kép) *VD, *AC

Bài tập thực hành 4:

* Mô tả đề bài:

- Khi ấn I0.0 thì giá trị tại IW2 + 5 và hiển thị ra QW6.
- Khi nhấn I0.1 thì giá trị IW2 - 2 và hiển thị ra QW6.
- Khi nhấn I0.2 thì giá trị IW2 $\times$ 4, hiển thị ra QW6.
- Nhấn I0.3 thì giá trị tại IW2 : 3 và hiển thị ra QW6.

* Yêu cầu:

- Viết chương trình lập trình trên phần mềm Microwin 4.0.
- Vẽ sơ đồ kết nối phần cứng PLC.
- Nạp chương trình vào PLC và kiểm tra kết quả.

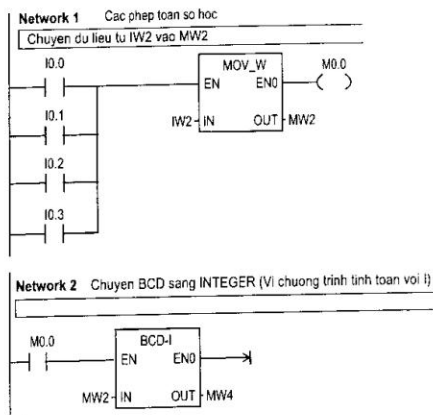
* Mục tiêu bài tập:

Sau khi học xong bài học, sinh viên có khả năng:

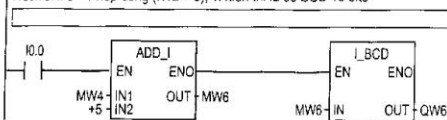
- Lập trình được chương trình các phép toán số trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

* Nội dung:

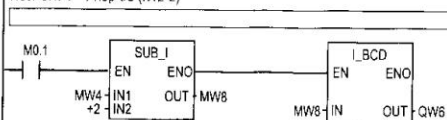
Chương trình được viết trong LAD như sau:



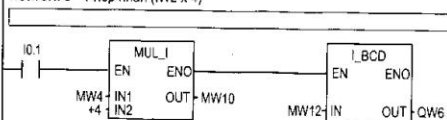
Network 3 Phep cong (IW2 + 5), vi hien thi la so BCD 16 bits



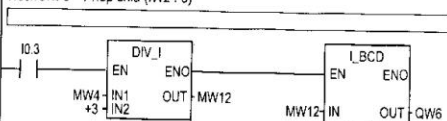
Network 4 Phep tru (IW2-2)



Network 5 Phep nhan (IW2 x 4)



Network 6 Phep chia (IW2 : 3)



Chương trình được viết trong STL như sau:

Network 1 Cac phep toan so hoc

Chuyen du lieu tu IW2 vao MW2

```
LD      I0.0
O       I0.1
O       I0.2
O       I0.3
MOVW    IW2, MW2
AENO
=       M0.0
```

Network 2 Chuyen BCD sang INTEGER
(Vi chương trình tính toán với I)

```
LD      M0.0
MOVW    MW2, MW4
BCD1    MW4
```

Network 3 Phép cộng (IW2 + 5)
vi hiện thị la so BCD 16 bits

LD	I0.0	
MOVW	MW4,	MW6
AENO		
+ I	+5,	MW6
AENO		
MOVW	MW6,	QW6
IBCD	QW6	

Network 4 Phép trừ (IW2 - 2)

LD	I0.1	
MOVW	MW4,	MW8
AENO		
- I	+2,	MW8
AENO		
MOVW	MW8,	QW6
IBCD	QW6	

Network 5 Phép nhân (IW2 x 4)

LD	I0.2	
MOVW	MW4,	MW10
AENO		
* I	+4,	MW10
AENO		
MOVW	MW12,	QW6
IBCD	QW6	

Network 6 Phép chia (IW2 : 3)

LD	I0.3	
MOVW	MW4,	MW2
AENO		
/ I	+3,	MW2
AENO		
MOVW	MW12,	QW6
IBCD	QW6	

3. ĐỒNG HỒ THỜI GIAN THỰC

- Đồng hồ thời gian thực là lệnh dùng để đọc, ghi các giá trị về thời gian ở trong máy vi tính

- Những giá trị đọc được hoặc ghi được với đồng hồ thời gian thực là các giá trị về ngày, tháng, năm và các giá trị về giờ, phút, giây

- Các dữ liệu đọc, ghi với đồng hồ thời gian thực trong LAD và trong STL có độ dài 1 byte và phải được mã hóa theo kiểu số nhị thập phân thân BCD (thí dụ 16#95 cho năm 95). Chúng nằm trong bộ đệm gồm 8 byte liền nhau theo thứ tự:

Byte 0	Năm (0 ÷ 99)	
Byte 1	Tháng (0 ÷ 12)	
Byte 2	Ngày (0 ÷ 31)	
Byte 3	Giờ (0 ÷ 23)	
Byte 4	Phút (0 ÷ 59)	
Byte 5	Giây (0 ÷ 59)	
Byte 6	0	
Byte 7	0	Ngày trong tuần

- Các dữ liệu hợp lệ:

Năm (yy)	Tháng (mm)	Ngày (dd)	Giờ (hh)	Phút (mm)	Giây (ss)
0 ÷ 99	1 ÷ 12	1 ÷ 31	0 ÷ 23	0 ÷ 59	0 ÷ 59

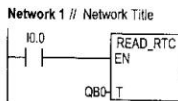
- Riêng giá trị về ngày trong tuần là một số tương ứng với 4 bit thấp trong byte thứ 7:

Chủ nhật	Thứ hai	Thứ ba	Thứ tư	Thứ năm	Thứ sáu	Thứ bảy
1	2	3	4	5	6	7

- Lệnh đọc thời gian thực READ_RTC (LAD): đọc nội dung của đồng hồ thời gian thực vào bộ đệm 8 byte được chỉ thị trong lệnh bằng toán hạng T.

- Lệnh ghi thời gian thực SET_RTC (LAD): ghi nội dung của bộ đệm 8 byte được chỉ thị trong lệnh bằng toán hạng T vào đồng hồ thời gian thực.

Ví dụ:



Network 1	Network Title
Network Comment	
LD	I0.0
TOOR	QBO

- Khi bấm nút I0.0, lệnh đọc thời gian thực sẽ được thực hiện, giá trị thời gian thực được lưu tại 8 byte, bắt đầu từ byte Q0 đến Q7 theo dạng số BCD:

+ Byte 0 chỉ giá trị năm có giá trị: 11

Bit	Q0.7	Q0.6	Q0.5	Q0.4	Q0.3	Q0.2	Q0.1	Q0.0
Value	0	0	0	1	0	0	0	1

+ Byte 1 chỉ giá trị tháng có giá trị: 10

Bit	Q1.7	Q1.6	Q1.5	Q1.4	Q1.3	Q1.2	Q1.1	Q1.0
Value	0	0	0	1	0	0	0	0

+ Byte 2 chỉ giá trị ngày có giá trị: 27

Bit	Q2.7	Q2.6	Q2.5	Q2.4	Q2.3	Q2.2	Q2.1	Q2.0
Value	0	0	1	0	0	1	1	1

+ Byte 3 chỉ giá trị giờ có giá trị: 10

Bit	Q3.7	Q3.6	Q3.5	Q3.4	Q3.3	Q3.2	Q3.1	Q3.0
Value	0	0	0	1	0	0	0	0

+ Byte 4 chỉ giá trị phút có giá trị: 18

Bit	Q4.7	Q4.6	Q4.5	Q4.4	Q4.3	Q4.2	Q4.1	Q4.0
Value	0	0	0	1	1	0	0	1

+ Byte 5 chỉ giá trị giây có giá trị: 27

Bit	Q5.7	Q5.6	Q5.5	Q5.4	Q5.3	Q5.2	Q5.1	Q5.0
Value	0	0	1	0	0	1	1	1

+ Byte 7 chỉ giá trị thứ trong tuần có giá trị: 4, tức là chỉ thứ năm

Bit	Q7.7	Q7.6	Q7.5	Q7.4	Q7.3	Q7.2	Q7.1	Q7.0
Value	0	0	0	0	0	1	0	0

Vậy giá trị thời gian thực tại thời điểm thực thi lệnh đọc thời gian thực là:

10 giờ 18 phút 27 giây ngày 27 tháng 10 năm 2011

Bài tập thực hành 5:

Ứng dụng đồng hồ thời gian thực trong PLC S7-200 để báo giờ làm việc và nghỉ.

**** Mô tả đề bài:***

Một xí nghiệp được trang bị một hệ thống tự động báo giờ làm việc và tan tầm như sau:

- Giờ làm việc: Đúng 7:00 giờ sáng, thì một hồi còi vang lên kéo dài trong 1 phút báo hiệu giờ làm việc bắt đầu.

- Giờ nghỉ: Đúng 17:00 giờ, thì một hồi còi vang lên kéo dài trong 1 phút báo hiệu giờ làm việc kết thúc.

**** Yêu cầu:***

- Viết chương trình lập trình trên phần mềm Microwin 4.0.

- Vẽ sơ đồ kết nối phần cứng PLC.

- Nạp chương trình vào PLC và kiểm tra kết quả.

**** Mục tiêu bài tập:***

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển còi báo hiệu giờ làm việc và giờ nghỉ trên phần mềm Microwin 4.0 theo yêu cầu bài toán.

- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

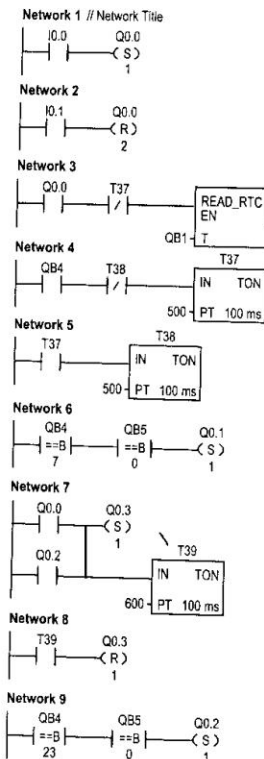
**** Nội dung:***

a) Quy ước các đầu vào/ra:

STT	Tiếp điểm logic	Chức năng
1	I0.0	Nút bấm khởi động
2	I0.1	Nút bấm dừng
3	Q0.0	Tiếp điểm trung gian
4	Q0.1	Tiếp điểm trung gian khi 7h
5	Q0.2	Tiếp điểm trung gian khi 17h
6	Q0.3	Còi báo hiệu

b) Chương trình lập trình:

Viết trong STL:



Network 1		Network Title	
Network Comment			
LD	I0.1		
S	Q0.0,	1	
Network 2			
LD	I0.1		
R	Q0.0,	2	
Network 3			
LD	Q0.0		
AN	T37		
TODR	QB1		
Network 4			
LDB <>	QB4,	24	
AN	T38		
TON	T37,	5	
Network 5			
LD	T37		
TON	T38,	5	
Network 6			
LDB =	QB4,	7	
AB=	QB5,	0	
S	Q0.1,	1	
Network 7			
LD	Q0.1		
O	Q0.2		
S	Q0.3,	1	
TON	T39,	600	
Network 8			
LD	T39		
R	Q0.3,	1	
Network 9			
LDB =	QB4,	23	
AB =	QB5,	0	
S	Q0.2,	1	

Bài tập thực hành 6:

Điều khiển chuông báo giờ học trong nhà trường (Trung học phổ thông).

** Mô tả đề bài:*

- 7h sáng thì bắt đầu ấn nút khởi động hệ thống, chuông kêu trong 5s rồi tắt báo hiệu vào lớp.
- Sau 45 phút thì khởi động chuông trong 5s báo hiệu hết tiết 1, ra chơi.
- Sau 5 phút ra chơi thì chuông lại kêu trong 5s báo vào lớp, vào tiết 2.
- Hết tiết 2 quá trình lại diễn ra tương tự.
- Hết tiết 6 chuông kêu 10s báo hiệu hết buổi học.

*** Yêu cầu:**

- Viết chương trình lập trình trên phần mềm Microwin 4.0.
- Vẽ sơ đồ kết nối phần cứng PLC.
- Nạp chương trình vào PLC và kiểm tra kết quả.

*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển chuông báo hiệu giờ học và giờ nghỉ trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

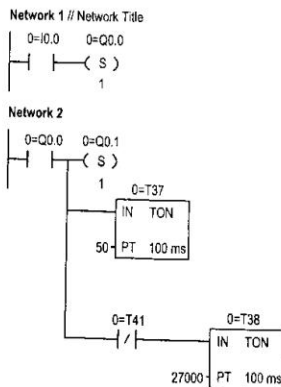
*** Nội dung:**

a) Quy ước các đầu vào/ra:

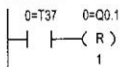
STT	Tiếp điểm logic	Chức năng
1	I0.0	Nút bấm khởi động
2	I0.1	Nút bấm dừng
3	Q0.0	Tiếp điểm trung gian
4	Q0.1	Chuông báo hiệu

b) Chương trình lập trình:

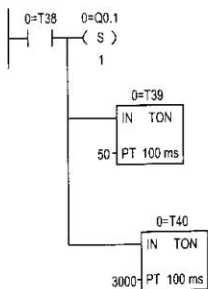
- Dùng lệnh Set, Reset:



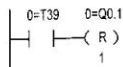
Network 3



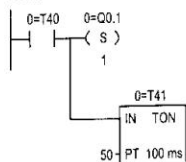
Network 4



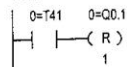
Network 5



Network 6

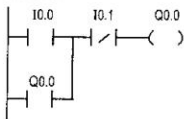


Network 7

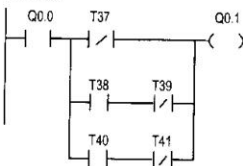


- Dùng các tiếp điểm logic:

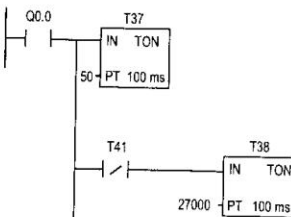
Network 1



Network 2



Network 3



Viết trong STL:

Network 1	Network Title
Network Comment	
LD I0.0	
S Q0.0, 1	

Network 2	Network Title
Network Comment	
LD Q0.0	
S Q0.1, 1	
TON T37, 50	
AN T41	
TON T38, 27000	

Network 3	Network Title
Network Comment	
LD T37	
R Q0.1, 1	

Network 4

LD	T38	
S	Q0.1,	1
TON	T39,	50
TON	T40,	3000

Network 5

LD	T39	
R	Q0.1,	1

Network 6

LD	T40	
S	Q0.1,	1
TON	T41,	50

Network 7

LD	T41	
R	Q0.1,	1

Bài tập thực hành 7:

Điều khiển chuông báo giờ học trong nhà trường (Dạy nghề).

*** Mô tả đề bài:**

- 13h chiều thì bắt đầu ấn nút khởi động hệ thống, chuông kêu trong 5s rồi tắt báo hiệu vào lớp.

- Sau 90 phút thì khởi động chuông trong 5s báo hiệu hết tiết 1, 2, ra chơi.
- Sau 10 phút ra chơi thì chuông lại kêu trong 5s báo vào lớp, vào tiết 3, 4.
- Hết tiết 3, 4 quá trình lại diễn ra tương tự vào tiết 5, 6.
- Hết tiết 6 chuông kêu 10s báo hiệu hết buổi học.

*** Yêu cầu:**

- Viết chương trình lập trình trên phần mềm Microwin 4.0.
- Vẽ sơ đồ kết nối phần cứng PLC.
- Nạp chương trình vào PLC và kiểm tra kết quả.

*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển chuông báo hiệu giờ học và giờ nghỉ trên phần mềm Microwin 4.0 theo yêu cầu bài toán.

- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

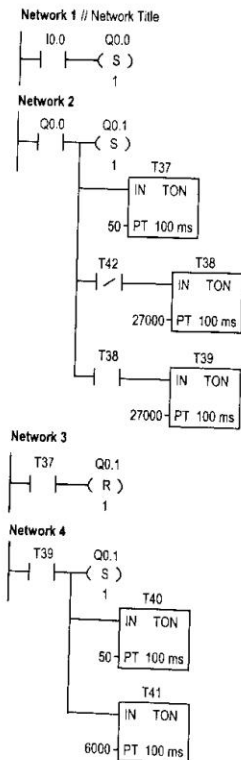
*** Nội dung:**

- a) Quy ước các đầu vào/ra:

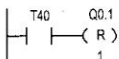
STT	Tiếp điểm logic	Chức năng
1	I0.0	Nút bấm khởi động
2	I0.1	Nút bấm dừng
3	Q0.0	Tiếp điểm trung gian
4	Q0.1	Chuông báo hiệu

b) Chương trình lập trình:

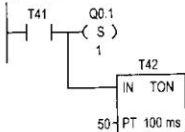
- Dùng lệnh Set, Reset:



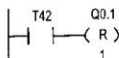
Network 5



Network 6

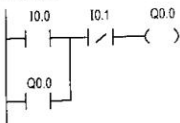


Network 7

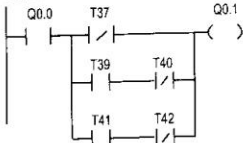


- Dùng các tiếp điểm logic:

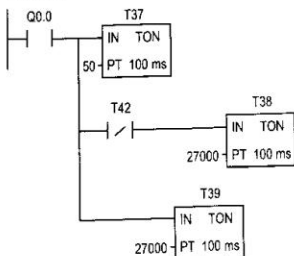
Network 1

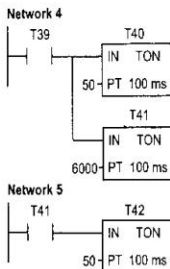


Network 2



Network 3





Viết trong STL:

Network 1		Network Title	
Network Comment			
LD	I0.0		
S	Q0.0,	1	
Network 2			
LD	Q0.0		
LPS			
S	Q0.1	1	
TON	T37,	50	
AN	T42		
TON	T38,	27000	
LPP			
A	T38		
TON	T39,	27000	
Network 3			
LD	T37		
R	Q0.1,	1	
Network 4			
LD	T39		
S	Q0.1,	1	
TON	T40,	50	
TON	T41,	6000	
Network 5			
LD	T40		
R	Q0.1,	1	
Network 6			
LD	T41		
S	Q0.1,	1	
TON	T42,	50	
Network 7			
LD	T42		
R	Q0.1,	1	

CÂU HỎI VÀ BÀI TẬP

1. Câu hỏi

Câu 1. So sánh sự khác nhau giữa các lệnh dịch chuyển, so sánh b và số nguyên trong PLC S7-200.

Câu 2. Nêu các toán hạng của phép biến đổi I-BCD và BCD-I trong PLC S7-200 CPU 224.

Câu 3. Nêu ưu nhược điểm của việc truyền 1 vùng dữ liệu so với việc truyền 1 từ đơn Word và từ kép DW.

Câu 4. Em hãy đọc giá trị thời gian thực trong PLC và thực hiện thiết lập lại giá trị thời gian cho PLC là: 20 giờ, 10 phút ngày 20 tháng 10 năm 2020.

2. Bài tập:

Bài 1. Một công ty được trang bị một hệ thống tự động báo giờ làm việc và chấm công như sau:

- Hệ thống tự động được kích hoạt từ 6h sáng.
- Mỗi nhân viên đi qua sẽ quét thẻ, cảm biến sẽ tự động ghi nhớ và đếm số lượng người vào đúng giờ.
- Từ 7h trở đi thì có người quét thẻ đi vào chuông sẽ kêu một hồi 5s báo hiệu.
- Đến 12h thì lại bắt đầu đếm số người đi ra.

Bài 2. Có 1 mô hình điều khiển đóng mở cửa tự động như sau:

- Có 2 sensor I0.2, I0.3 cảm biến phát hiện có người tới gần, cửa tự động mở, mở ra hết cỡ gặp công tắc hành trình I0.4 thì dừng lại trong 5s.
- Sau đó đóng lại gặp công tắc hành trình I0.5 thì dừng lại, đang đóng mà có người tới thì lại tự động mở ra.
- Nếu có vật cản thì I0.6 sẽ báo và cửa dừng lại, hết vật cản thì lại đóng tiếp.

Yêu cầu:

- Quy ước các đầu vào/ra.
- Viết chương trình ở dạng LAD và STL.
- Nạp vào PLC và chạy thử.

Bài 4

LẮP ĐẶT MÔ HÌNH ĐIỀU KHIỂN BẰNG PLC

MỤC TIÊU:

- Phân tích quy trình công nghệ của một số mạch máy sản xuất.
- Lập trình được một số mạch ứng dụng trong thực tế.
- Nạp chương trình, vận hành và kiểm tra mạch hoạt động theo yêu cầu thực tế.

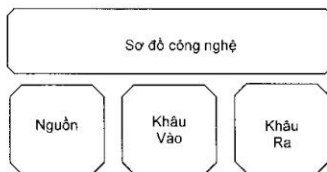
NỘI DUNG CỦA BÀI:

1. GIỚI THIỆU

Nâng cao chất lượng trong giảng dạy kỹ thuật là việc luôn luôn gắn liền giữa lý thuyết và thực hành. Hiện nay, trong thực hành của sinh viên, thì các đồ dùng dạy học với dáng vẻ trang nhã và gọn gàng, bảo đảm an toàn và giúp cho sinh viên có cái nhìn khái quát về những ứng dụng trong thực tế có một ý nghĩa quan trọng. Nó không những giúp cho sinh viên có hứng thú trong học tập mà còn giúp cho sinh viên có thêm những sáng kiến mới, cũng như cách thức tổ chức trong thực tế.

Từ lý do đó, chúng tôi đã đưa ra một số mô hình đáp ứng được những yêu cầu trên. Đây là các mô hình dùng cho môn học PLC từ cơ bản đến nâng cao, lấy từ các ứng dụng trong thực tế được mô phỏng quy trình công nghệ bằng các đèn LED. Mỗi mô hình có thể có nhiều bài tập ứng dụng khác nhau.

Mỗi mô hình có kích thước A4 (210 × 297mm) rất phù hợp trong việc lắp ghép trên các bảng tạo thành một môđun hoàn chỉnh. Cấu trúc chung bề mặt của mô hình được phân bố như sau:



Hình 4.1: Cách kết nối với mô hình.

Mục đích của việc phân thành từng cụm riêng là giúp cho sinh viên tránh được những nhầm lẫn đáng tiếc trong quá trình thực hành, đồng thời cũng giúp cho sinh viên tiếp thu thêm về cách thức tổ chức trong thực hành.

Các mô hình thực tập gồm có:

- Mô hình thang máy xây dựng.
- Mô hình điều khiển động cơ sao - tam giác.
- Mô hình xe chuyển nguyên liệu.
- Mô hình đo chiều dài và sắp xếp vật liệu.
- Mô hình thiết bị nâng hàng hóa.
- Mô hình thiết bị đóng vỏ vào chai.
- Mô hình thiết bị trộn hóa chất.

Các mô hình này đã được sắp xếp theo thứ tự và có các bài tập kèm theo.

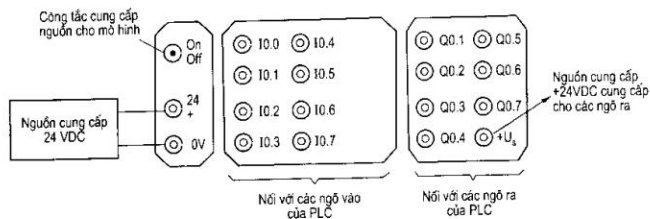
Toàn bộ các mô hình sử dụng điện áp 24VDC. Điện áp này có thể được lấy từ nguồn riêng hoặc nguồn 24VDC có sẵn cung cấp cho PLC. Đối với các PLC có ngõ ra là relay thì trên mô hình có thiết kế sẵn nguồn U_s dùng làm nguồn cung cấp cho các ngõ ra này.

Các mô hình cũng có thể được ứng dụng cho các bộ điều khiển lập trình loại nhỏ như: LOGO! của hãng Siemens, EASY của hãng Moeller, ZEN của Omron...

Tùy theo nội dung bài học mà có thể chọn mô hình thích hợp cho bài tập ứng dụng. Một mô hình có thể sử dụng với nhiều bài tập ứng dụng khác nhau. Lấy ví dụ: *mô hình thang máy xây dựng có thể được sử dụng trong các bài học như ứng dụng điều khiển theo tổ hợp logic, điều khiển với các lệnh ghi/xóa tiếp điểm, sử dụng timer, sử dụng counter, và ứng dụng trong điều khiển trình tự.*

2. CÁCH KẾT NỐI DÂY

Cách nối dây từ PLC đến mô hình được cho như hình vẽ:

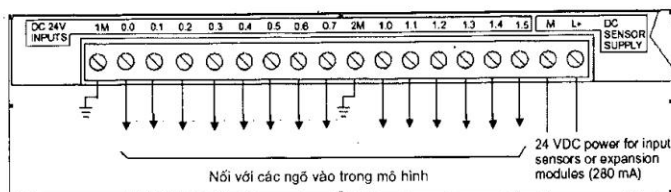


Hình 4.2: Cách kết nối với mô hình.

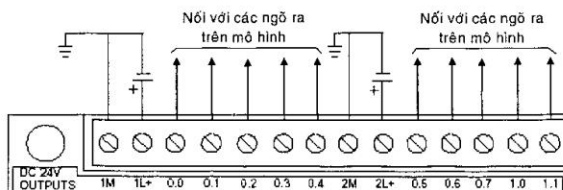
Để kết nối được với PLC, yêu cầu các môđun vào/ra của PLC như sau:

- Sử dụng nguồn áp 24VDC (ổn áp).
- Nguồn cung cấp cho môđun vào/ra phải được kết nối.
- Nếu các ngõ ra là rơle và chưa có nguồn cung cấp thì đấu chung một đầu lại rồi nối với nguồn $+U_s$ ở trên mô hình (hoặc nguồn +24VDC ngoài), còn các đầu còn lại của rơle nối với ngõ ra trên mô hình.

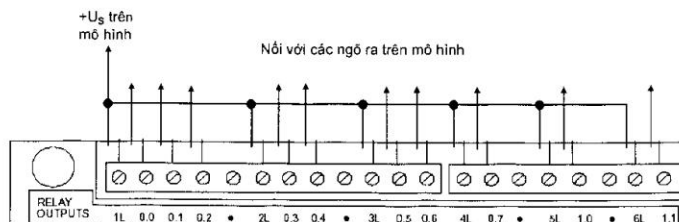
Các kết nối có thể thực hiện như ví dụ sau:



Hình 4.3: Cách nối với các ngõ vào trong mô hình



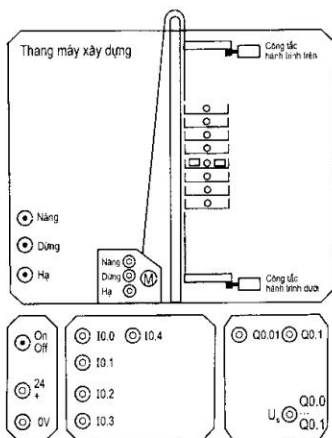
Hình 4.4: Cách nối các ngõ ra 24VDC của PLC với các ngõ ra trong mô hình.



Hình 4.5: Cách nối các ngõ ra rơle của PLC với các ngõ ra trong mô hình.

3. CÁC MÔ HÌNH VÀ BÀI TẬP ỨNG DỤNG

3.1. Mô hình thang máy xây dựng



a) Mô tả

Mô tả quy trình công nghệ của một thang máy xây dựng. Sự chuyển động của thang được biểu diễn dưới dạng LED. Tín hiệu các công tắc giới hạn được tạo ra tự động.

Mô hình này được ứng dụng trong phần bài tập cơ bản trong môn học PLC (ứng dụng các cổng logic, timer, counter). Ngoài ra cũng có thể được áp dụng cho phần lập trình nâng cao (điều khiển trình tự).

b) Bảng ký hiệu

Ký hiệu	Địa chỉ	Chú thích
Nâng	I0.0	Nút nhấn nâng, thường hở
Hạ	I0.1	Nút nhấn hạ, thường hở
Dừng	I0.2	Nút nhấn dừng, thường đóng
GH_Trên	I0.3	Công tắc hành trình trên, thường đóng
GH_Dưới	I0.4	Công tắc hành trình dưới, thường đóng
K1	Q0.0	Cuộn dây khởi động từ K1, nâng gàu
K2	Q0.1	Cuộn dây khởi động từ K2, hạ gàu

c) Bài tập

Các bài tập này được giải với theo phần mềm STEP 7 Microwin 4.0.

Bài tập thực hành 1: Ứng dụng công logic, các lệnh ghi/xóa tiếp điểm.

*** Mô tả đề bài:**

- Khi nhấn nút nhấn nâng, gàu sẽ được nâng lên đến công tắc giới hạn trên thì dừng lại.
- Khi nhấn nút nhấn hạ, gàu sẽ hạ xuống đến công tắc giới hạn dưới thì dừng lại.
- Khi đang nâng hoặc hạ, nếu nhấn nút nhấn dừng thì gàu dừng lại.

*** Yêu cầu:**

- Hãy viết chương trình theo mô tả theo hai cách: ứng dụng công logic và sử dụng các lệnh ghi xóa tiếp điểm.

- Kiểm tra hoạt động bằng mô hình.

*** Mục tiêu bài tập:**

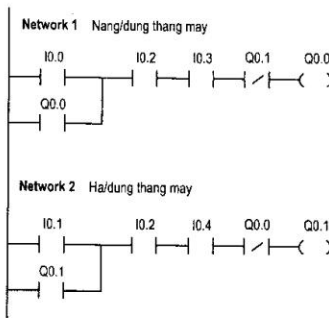
Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển thang máy xây dựng trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bản thí nghiệm.

*** Nội dung:**

Cách 1: Ứng dụng công logic.

- Chương trình viết ở dạng LAD:



- Chương trình được viết ở STL:

```

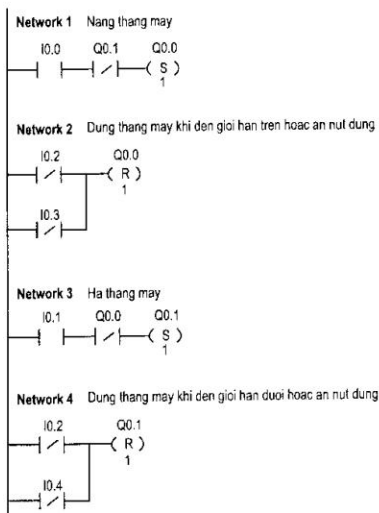
NETWORK 1          //Nâng/Dừng thang máy
//
LD      IO.0
O       Q0.0
A       IO.2
A       IO.3
AN      Q0.1
=       Q0.0

NETWORK 2          //Hạ/Dừng thang máy
//
LD      IO.1
O       Q0.1
A       IO.2
A       IO.4
AN      Q0.0
=       Q0.1

```

Cách 2: Sử dụng lệnh Set/reset:

Chương trình viết ở LAD:



Chương trình viết ở STL:

```
NETWORK 1          //Nâng thang máy
//
LD      IO.0
AN      Q0.1
S        Q0.0, 1
```

```
NETWORK 2          //Dừng thang máy khi đến giới hạn trên hoặc ấn nút dừng
//
LDN     IO.2
ON      IO.3
R        Q0.0, 1
```

```
NETWORK 3          //Hạ thang máy
//
LD      IO.1
AN      Q0.0
S        Q0.1, 1
```

```
NETWORK 4          //Dừng thang máy khi đến giới hạn dưới hoặc ấn nút dừng
//
LDN     IO.2
ON      IO.4
R        Q0.1, 1
```

Bài tập thực hành 2: Ứng dụng Timer.

**** Mô tả đề bài:***

a) Khi ấn nút nâng thì gàu được nâng lên, đến giới hạn trên thì dừng lại 5s, sau đó tự động hạ xuống. Đến giới hạn dưới thì dừng.

- Trong quá trình nâng lên hoặc hạ xuống cũng có thể dừng.

b) Khi ấn nút nâng thì gàu được nâng lên, đến giới hạn trên thì dừng lại 5s, sau đó tự động hạ xuống đến giới hạn dưới thì dừng lại 10s, sau đó tự động nâng lên.

- Thang cũng có thể nâng lên khi chưa hết 10s chờ tự động mà có người ấn nút nâng.

**** Yêu cầu:***

- Viết chương trình điều khiển thang máy xây dựng trên phần mềm Microwin 4.0.

- Kiểm tra hoạt động bằng mô hình.

**** Mục tiêu bài tập:***

Sau khi học xong bài học, sinh viên có khả năng:

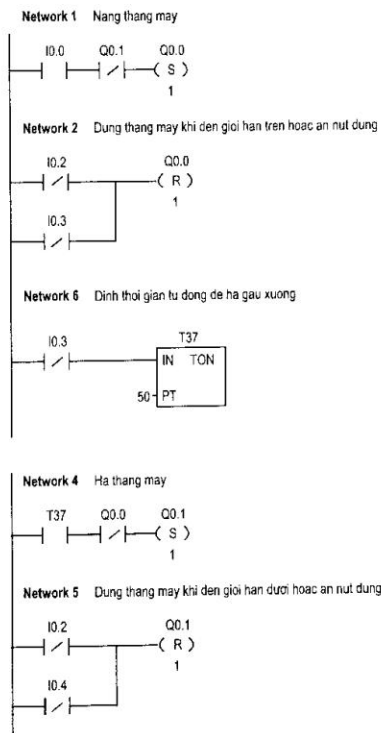
- Lập trình được chương trình điều khiển thang máy xây dựng trên phần mềm Microwin 4.0 theo yêu cầu bài toán.

- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

* Nội dung:

Câu a):

- Chương trình viết ở LAD:



- Chương trình viết ở STL:

```

NETWORK 1          //Nâng thang máy
//
LD      I0.0
AN      Q0.1
S       Q0.0, 1

NETWORK 2          //Dừng thang máy khi đến giới hạn trên hoặc an nút dừng
//
LDN     I0.2
ON      I0.3
R       Q0.0, 1

NETWORK 3          //Định thời gian tự động để hạ gầu xuống
//
LDN     I0.3
TON     T37, +50

NETWORK 4          //Hạ thang máy
//
LD      T37
AN      Q0.0
S       Q0.1, 1

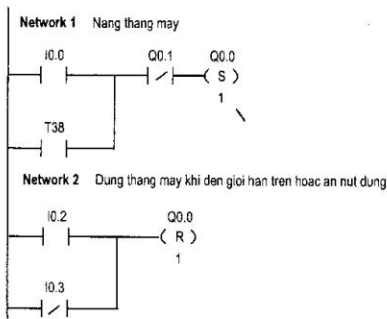
NETWORK 5          //Dừng thang máy khi đến giới hạn dưới hoặc an nút dừng
//
LDN     I0.2
ON      I0.4
R       Q0.1, 1

```

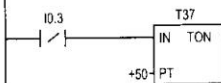
Ghi chú: Nếu bài toán có cho yêu cầu khi ấn nút hạ thì gầu cũng hạ, lúc này cần thêm một tiếp điểm "NO" của I0.1 song song với tiếp điểm "NO" T37 ở Network 4.

Câu b):

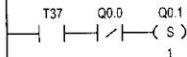
- Chương trình viết ở LAD:



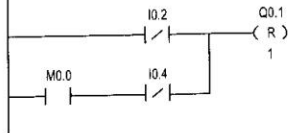
Network 3 Dinh thời gian tự động để hạ gầu xuống



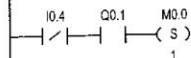
Network 4 Hạ thang máy



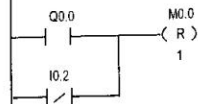
Network 5 Dừng thang máy khi đến giới hạn dưới hoặc ấn nút dừng



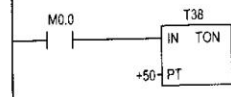
Network 6 Nhớ trạng thái thang máy hạ xuống đến giới hạn dưới



Network 7 Xóa trạng thái nhớ thang máy ở giới hạn dưới



Network 8 Dinh thời gian tự động nâng lên



Chương trình viết ở STL:

```
NETWORK 1      //Nâng thang máy
//
LD      I0.0
O       T38
AN      Q0.1
S       Q0.0, 1

NETWORK 2      //Dừng thang máy khi đến giới hạn trên hoặc an nút dừng
//
LDN     I0.2
ON      I0.3
R       Q0.0, 1

NETWORK 3      //Định thời gian tự động để hạ gầu xuống
//
LDN     I0.3
TON     T37, +50

NETWORK 4      //Hạ thang máy
//
LD      T37
AN      Q0.0
S       Q0.1, 1

NETWORK 5      //Dừng thang máy khi đến giới hạn dưới hoặc an nút dừng
//
LDN     I0.2
LD      M0.0
AN      I0.4
OLD
R       Q0.1, 1

NETWORK 6      //Nhớ trạng thái thang máy hạ xuống đến giới hạn dưới
//
LDN     I0.4
A       Q0.1
S       M0.0, 1

NETWORK 7      //Xóa trạng thái nhớ thang máy ở giới hạn dưới
//
LD      Q0.0
ON      I0.2
R       M0.0, 1

NETWORK 8      //Định thời gian tự động nâng lên
//
LD      M0.0
TON     T38, +50
```

Ghi chú: Nếu bài toán có cho yêu cầu khi ấn nút hạ thì gầu cũng hạ, lúc này cần thêm một tiếp điểm "NO" của I0.1 song song với tiếp điểm "NO" T37 ở Network 4.

Bài tập thực hành 3: Ứng dụng bộ đếm.

*** Mô tả đề bài:**

- Khi ấn nút nâng thì gầu được nâng lên, đến giới hạn trên thì dừng lại 5s, sau đó tự động hạ xuống đến giới hạn dưới thì dừng lại 10s, sau đó tự động nâng lên.
- Khi gầu nâng lên được 10 lần thì không nâng lên nữa và sau đó hạ xuống trở về vị trí cơ bản và quá trình lặp lại.
- Trong quá trình đang nâng hoặc hạ cũng có thể dừng gầu.

*** Yêu cầu:**

- Viết chương trình điều khiển thang máy xây dựng trên phần mềm Microwin 4.0.
- Kiểm tra hoạt động bằng mô hình.

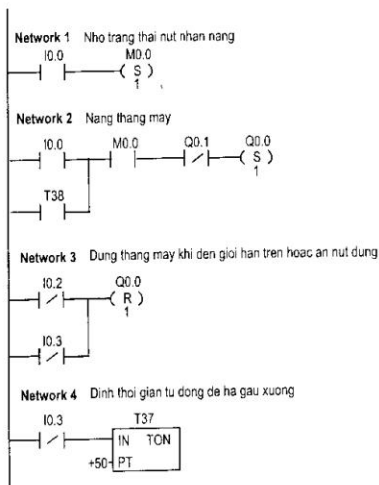
*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

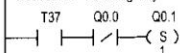
- Lập trình được chương trình điều khiển thang máy xây dựng trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

*** Nội dung:**

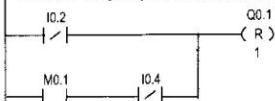
- Chương trình viết ở LAD:



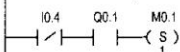
Network 5 Hạ thang máy



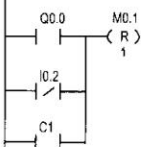
Network 6 Dừng thang máy khi đến giới hạn dưới hoặc an nút dừng



Network 7 Nhỏ trạng thái thang máy hạ xuống đến giới hạn dưới



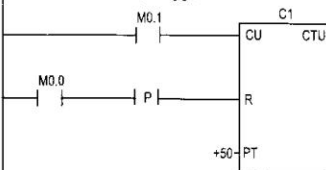
Network 8 Xóa trạng thái nhỏ thang máy ở giới hạn dưới



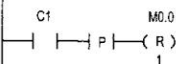
Network 9 Định thời gian tự động nâng lên



Network 10 Đếm số lần nâng gầu



Network 11 Không cho phép gầu tự động nâng lên khi đã nâng đủ 10 lần



Ghi chú: Nếu bài toán có cho yêu cầu khi ấn nút hạ thì gàu cũng hạ, lúc này cần thêm một tiếp điểm "NO" của I0.1 song song với tiếp điểm "NO" T37 ở Network 5.

- Chương trình viết ở STL:

```
NETWORK 1      //Nho trạng thái nút nhấn nâng
//
LD      I0.0
S      M0.0, 1

NETWORK 2      //Nâng thang máy
//
LD      I0.0
O      T38
A      M0.0
AN      Q0.1
S      Q0.0, 1

NETWORK 3      //Dung thang máy khi đến giới hạn trên hoặc ấn nút dừng
//
LDN     I0.2
ON      I0.3
R      Q0.0, 1

NETWORK 4      //Định thời gian tự động hạ gàu xuống
//
LDN     I0.3
TON     T37, +50

NETWORK 5      //Hạ thang máy
//
LD      T37
AN      Q0.0
S      Q0.1, 1

NETWORK 6      //Dung thang máy khi đến giới hạn dưới hoặc ấn nút dừng
//
LDN     I0.2
LD      M0.1
AN      I0.4
OLD
R      Q0.1, 1

NETWORK 7      //Nho trạng thái thang máy hạ xuống đến giới hạn dưới
//
LDN     I0.4
A      Q0.1
S      M0.1, 1
```

```

NETWORK 8      //Xoa trạng thái nhỏ thang máy ở giới hạn dưới
//
LD      Q0.0
ON      I0.2
O        C1
R      M0.1, 1

NETWORK 9      //Dinh thời gian tự động nâng lên
//
LD      M0.1
TON     T38, +S0

NETWORK 10     //Đếm số lần nâng gàu
//
LD      M0.1
LD      M0.0
EU
CTU     C1, +10

NETWORK 11     //Không cho phép gàu tự động nâng lên khi đã nâng đủ 10 lần
//
LD      C1
EU
R      M0.0, 1

```

3.2. Mô hình điều khiển động cơ Y-Δ

3.2.1. Mô tả

- Mô hình này mô phỏng một động cơ mở máy Y/Δ. Sự chuyển động của rotor và sự đóng cắt của các contactor được mô tả bằng LED.

- Mô hình này được ứng dụng trong phần bài tập cơ bản của môn học PLC (ứng dụng PLC trong điều khiển động cơ, các cổng logic, timer).

- Có thể dựa vào mô hình bảng tải để lập thành các bài tập khác nhau như: Mở máy động cơ Y/Δ, đảo chiều quay động cơ, điều khiển động cơ ở các tốc độ khác nhau, bồn sấy.

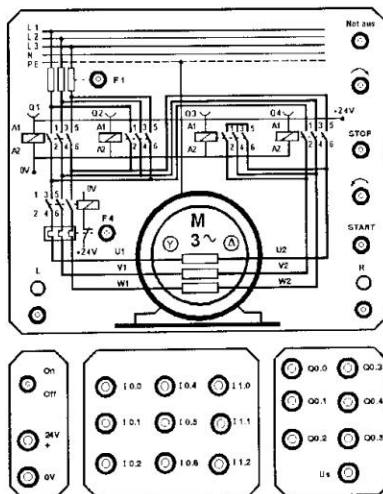
- Cách thức nối dây cho mô hình như ở mục 2.

- Cách vận hành mô hình:

Sau khi đã nối dây mô hình với PLC xong, thực hiện viết chương trình theo bài tập đưa ra [có thể tự kiểm tra các ngõ vào/ra bằng phần mềm (đối với S7-200 dùng bảng Status chart)] và sau đó thực hiện mô phỏng với mô hình.

Khi động cơ quay phải (trái), các đèn LED sẽ chuyển động theo chiều phải (trái) theo các trạng thái tương ứng: Nếu đèn chuyển động nhanh báo động cơ quay tốc độ nhanh và ngược lại. Các contactor khi được đóng điện được báo bởi các đèn báo đặt ở

ký hiệu cuộn dây. Trong động cơ có đặt 3 dây đèn báo tượng trưng cho 3 cuộn dây của động cơ. Nếu động cơ có điện thì các dây đèn này sáng.



3.2.2. Bảng ký hiệu

Ký hiệu	Địa chỉ	Chú thích
Start	I0.0	Nút nhấn khởi động, thường hở
Stop	I0.1	Nút nhấn dừng, thường đóng
F1	I0.2	CB 3 pha, công tắc
F4	I0.4	Relay nhiệt, công tắc
Right	I0.5	Chọn chiều quay phải, thường hở
Left	I0.6	Chọn chiều quay trái, thường hở
H_Right	I1.0	Quay phải tốc độ nhanh, thường hở
H_Left	I1.1	Quay trái tốc độ nhanh, thường hở
Not aus	I1.2	Dừng khẩn cấp, công tắc
Q1	Q0.0	Khởi động từ K1, quay phải
Q2	Q0.1	Khởi động từ K2, quay trái
Q3	Q0.2	Khởi động từ K3, chạy Y
Q4	Q0.3	Khởi động từ K4, chạy D
R	Q0.4	Đèn báo quay phải
L	Q0.5	Đèn báo quay trái

3.2.3. Bài tập

Bài tập thực hành 1: Ứng dụng các cổng logic, lệnh ghi xóa tiếp điểm.

*** Mô tả đề bài:**

- Khi ấn nút khởi động "Start" (I0.0), các contactor Q1 và Q3 đóng lại. Động cơ chạy ở chế độ Y. Sau đó nếu nút nhấn "Right" (I0.5) được ấn thì contactor Q3 tắt và contactor Q4 có điện. Động cơ chạy ở chế độ D.

- Động cơ được cung cấp điện bởi CB 3 pha, và được bảo vệ bởi role nhiệt và dừng bằng nút nhấn "Stop".

*** Yêu cầu:**

- Viết chương trình điều khiển động cơ Y/ Δ trên phần mềm Microwin 4.0

- Kiểm tra hoạt động bằng mô hình.

*** Mục tiêu bài tập:**

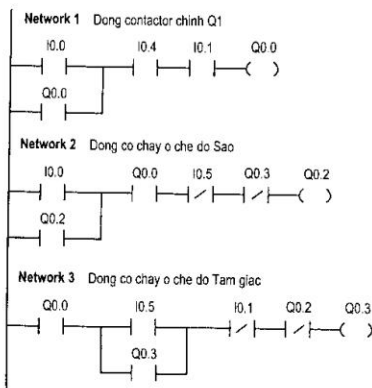
Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều khiển động cơ Y/ Δ trên phần mềm Microwin 4.0 theo yêu cầu bài toán.

- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.

*** Nội dung:**

- Chương trình viết ở LAD:



- Chương trình viết ở STL:

```
NETWORK 1          //Đóng contactor chính Q1
//
LD      I0.0
O       Q0.0
A       I0.4
A       I0.1
=       Q0.0

NETWORK 2          //Đóng cơ chạy ở chế độ Sao
//
LD      I0.0
O       Q0.2
A       Q0.0
AN      I0.5
AN      Q0.3
=       Q0.2

NETWORK 3          //Đóng cơ chạy ở chế độ Tam giác
//
LD      Q0.0
LD      I0.5
O       Q0.3
ALD
A       I0.1
AN      Q0.2
=       Q0.3
```

Bài tập 2:

** Mô tả đề bài:*

- Trước tiên chọn chiều quay của động cơ bằng các nút nhấn "Right" hoặc "Left". Khi ấn các nút này tương ứng các contactor Q1 hoặc Q2 đóng lại và các đèn báo quay phải "R" hoặc quay trái "L" sáng.

- Sau đó ấn nút khởi động "Start" thì động cơ sẽ quay ở tốc độ thấp trước (contactor Q3 có điện). Bây giờ có thể cho động cơ quay ở tốc độ cao hơn bằng cách ấn các nút tương ứng với các chiều quay "H-Right" hoặc "H-Left". Khi ấn các nút này thì contactor Q4 có điện.

- Động cơ được bảo vệ quá nhiệt và có dừng bằng nút nhấn "Stop".

- Không cho phép đảo chiều trực tiếp, cũng như việc chuyển đổi tốc độ chỉ có thể từ thấp sang cao.

** Yêu cầu:*

- Viết chương trình điều khiển tốc độ và đảo chiều quay động cơ trên phần mềm Microwin 4.0.

- Kiểm tra hoạt động bằng mô hình.

** Mục tiêu bài tập:*

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình điều tốc độ và đảo chiều quay động cơ trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bản thí nghiệm

Bài tập 3: Ứng dụng Timer điều khiển mở máy động cơ Y/ Δ .

** Mô tả đề bài:*

- Sau khi CB 3 pha được đóng, ấn nút khởi động "Start" thì động cơ hoạt động ở chế độ Y. Sau thời gian 10s thì động cơ được chuyển sang hoạt động ở chế độ tam giác.
- Động cơ quay theo chiều phải.
- Khi ấn nút "Right" hoặc "Left" thì động cơ sẽ hoạt động ở chế độ sao/tam giác với chiều quay là chiều đã chọn. Tương ứng các đèn báo quay phải hoặc trái sáng.

** Yêu cầu:*

- Viết chương trình mở máy động cơ Y/ Δ trên phần mềm Microwin 4.0
- Kiểm tra hoạt động bằng mô hình.

** Mục tiêu bài tập:*

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình mở máy động cơ Y/ Δ trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bản thí nghiệm.

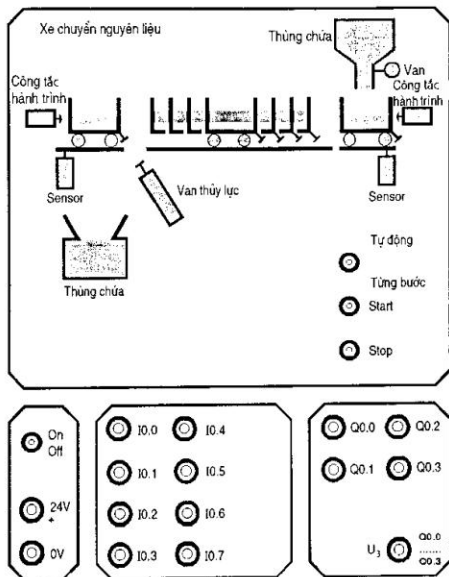
3.3. Mô hình xe chuyển nguyên liệu

3.3.1. Mô tả

Mô phỏng một xe vận chuyển nguyên liệu từ nơi này đến nơi khác với việc lấy nguyên liệu từ bồn chứa và xả nguyên liệu vào bồn chứa khác bằng các LED với nhiều màu sắc khác nhau. Cũng như các cảm biến và công tắc hành trình đều tạo ra tự động.

- Ứng dụng trong PLC cơ bản: Điều khiển tổ hợp logic.
- Ứng dụng trong PLC nâng cao: Điều khiển trình tự.
- Cách thức nối dây cho mô hình như ở mục 2.
- Cách vận hành mô hình:

Sau khi đã nối dây mô hình với PLC xong, thực hiện viết chương trình theo bài tập đưa ra [có thể tự kiểm tra các ngõ vào ra bằng phần mềm (đối với S7-200 dùng bảng Status chart)] và sau đó thực hiện mô phỏng với mô hình.



Nguyên liệu trong bồn chứa khi đang được xả hoặc được rót vào biểu thị bởi những dòng LED chạy, các đèn LED chạy đuổi tượng trưng cho xe đang di chuyển, van thủy lực được biểu thị bởi đèn sáng dần và tắt dần.

Tùy theo yêu cầu bài tập mà có thể lập trình xe chạy ở chế độ tự động, chế độ tay, hoặc hoạt động ở cả hai chế độ.

3.3.2. Bảng ký hiệu

Ký hiệu	Địa chỉ	Chú thích
Start	I0.0	Khởi động hệ thống, thường hở.
End 1	I0.1	Cổng tắc hành trình ở trạm xả, thường đóng
Fill 1	I0.2	Cảm biến báo xe rỗng, thường đóng.
End 2	I0.3	Cổng tắc hành trình trạm nạp, thường đóng.
Fill 2	I0.4	Cảm biến báo đầy, thường hở.

Ký hiệu	Địa chỉ	Chú thích
Stop	I0.5	Dừng, thường đóng.
Step	I0.6	Chế độ bước, thường hở.
Auto	I0.7	Chế độ tự động, thường hở.
Dir_A	Q0.0	Xe chạy về hướng A
Dir_B	Q0.1	Xe chạy về hướng B
Y1	Q0.2	Van xả nguyên liệu
Y2	Q0.3	Van thủy lực

3.3.3. Bài tập

Bài tập thực hành:

* Mô tả đề bài:

- Xe vận chuyển nguyên liệu có thể thực hiện qua công tắc chọn chế độ:
- + Chế độ tự động: I0.6.
- + Chế độ bước: I0.7.
- Vị trí cơ bản: Xe ở vị trí công tắc hành trình End 2 (I0.3 và xe chưa được làm đầy).
- Chế độ tự động:

Khi xe ở vị trí cơ bản và công tắc chọn chế độ đặt ở chế độ tự động, khi nhấn nút khởi động (I0.0) thì van xả Y1 mở, vật liệu được đổ vào xe, cảm biến Fill 2 dùng để nhận biết xe đã được đổ đầy. Khi xe đầy thì van xả Y1 mất điện và xe chạy về hướng B. Sau thời gian ổn định 5s, xe dừng lại tại B (trạm nhận nguyên liệu) khi chạm công tắc hành trình S2. Xanh thủy lực của thiết bị xả được điều khiển và tắt hẳn trên xe được mở, vật liệu được rót vào bồn chứa. Khi xe xả hết vật liệu cảm biến S4 phát ra tín hiệu 1, pittông thủy lực của thiết bị xả mất điện, tắt hẳn trở về vị trí cũ, xe dừng 5 giây sau đó chạy về hướng A. Chu kỳ hoạt động được lặp lại.

Nếu trong chu kỳ hoạt động mà nút "dừng" được ấn thì quá trình vẫn tiếp tục cho đến khi xe trở về vị trí cơ bản (xe rỗng và ở trạm nhận nguyên liệu) và dừng hẳn.

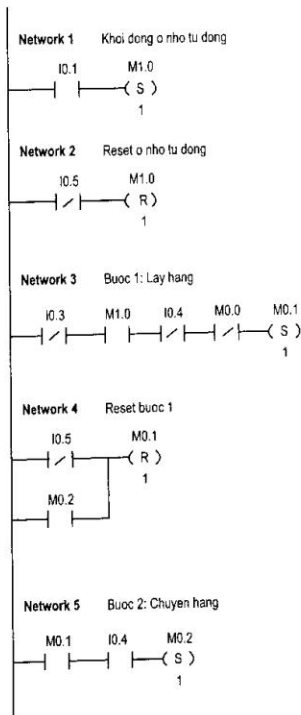
- Chế độ bước:

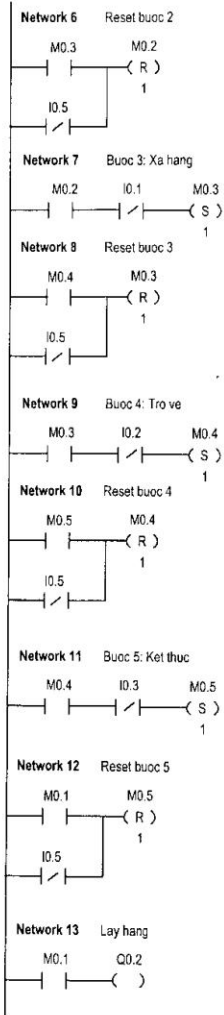
Ở mỗi bước thực hiện phải thông qua nút nhấn "start".

Ví dụ: khi ấn "start" xe đứng vị trí van xả được mở, khi xe đầy thì S3 tác động, van xả đóng lại. Nếu tiếp tục ấn "start" thì xe chạy về hướng B.

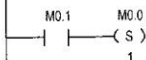
* *Yêu cầu:*

- Viết chương trình điều khiển xe chuyển nguyên liệu này và sau đó kết nối với thiết bị 7 mô phỏng để kiểm tra hoạt động ở hai dạng: Điều khiển dùng tổ hợp logic và điều khiển trình tự.
- Kiểm tra hoạt động bằng mô hình.
- Chương trình được viết ở LAD:

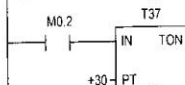




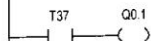
Network 14 Set o nha khoi dong



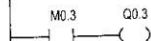
Network 15 Dinh thoi gian on dinh sau khi hang da do day xe



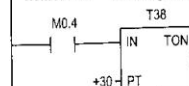
Network 16 Xe chuyen ve tram xa



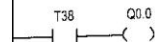
Network 17 Xa hang



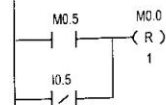
Network 18 Dinh thoi gian on dinh sau khi xa het hang



Network 19 Di chuyen xe ve tram nhan hang



Network 20 Reset o nha khoi dong



- Chương trình được viết ở dạng STL:

```
NETWORK 1          //Khởi động o nhỏ tu động
//
LD      IO.0
S       M1.0, 1

NETWORK 2          //Reset o nhỏ tu động
//
LDN     IO.5
R       M1.0, 1

NETWORK 3          //Buoc 1: Lay hang
//
LDN     IO.3
A       M1.0
AN      IO.4
AN      MO.0
S       MO.1, 1

NETWORK 4          //Reset buoc 1
//
LDN     IO.5
O       MO.2
R       MO.1, 1

NETWORK 5          //Buoc 2: Chuyen hang
//
LD      MO.1
A       IO.4
S       MO.2, 1

NETWORK 6          //Reset buoc 2
//
LD      MO.3
ON      IO.5
R       MO.2, 1

NETWORK 7          //Buoc 3: Xa hang
//
LD      MO.2
AN      IO.1
S       MO.3, 1

NETWORK 8          //Reset buoc 3
//
LD      MO.4
ON      IO.5
R       MO.3, 1
```

```

NETWORK 9      //Buoc 4: Tro ve
//
LD      MO.3
A       IO.2
S       MO.4, 1

NETWORK 10     //Reset buoc 4
//
LD      MO.5
ON      IO.5
R       MO.4, 1

NETWORK 11     //Buoc 5: Ket thuc
//
LD      MO.4
AN      IO.3
S       MO.5, 1

NETWORK 12     //Reset buoc 5
//
LD      MO.1
ON      IO.5
R       MO.5, 1

NETWORK 13     //Lay hang
//
LD      MO.1
=       Q0.2

NETWORK 14     //Set o nho khoi dong
//
LD      MO.1
S       MO.0, 1

NETWORK 15     //Dinh thoi gian on dinh sau khi hang da do day xe
//
LD      MO.2
TON     T37, +30

NETWORK 16     //Xe chuyen ve tram xa
//
LD      T37
=       Q0.1

```

```

NETWORK 17      //Xe hang
//
LD      M0.3
=       Q0.3

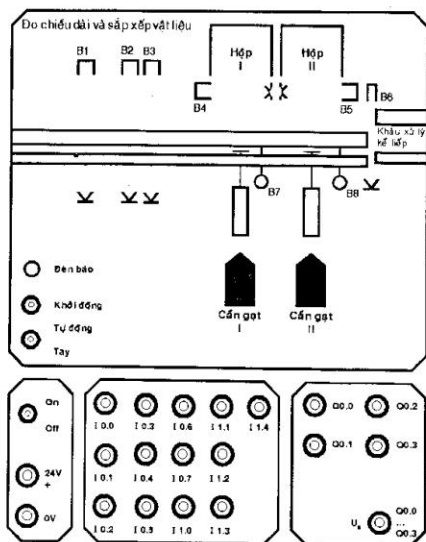
NETWORK 18      //Dinh thoi gian on dinh sau khi xa het hang
//
LD      M0.4
TON     T38, +30

NETWORK 19      //Di chuyen xe ve tram nhan hang
//
LD      T38
=       Q0.0

NETWORK 20      //Reset o nho khoi dong
//
LD      M0.5
ON      I0.5
R       M0.0, 1

```

3.4. Mô hình đo chiều dài và sắp xếp vật liệu



3.4.1. Mô tả

Mô hình mô phỏng một hệ thống băng tải vận chuyển gỗ và sắp xếp các loại gỗ có chiều dài ngắn khác nhau vào các thùng chứa bằng các cần gạt khí nén. Hàng LED lớn ở trên tượng trưng cho các đoạn gỗ di chuyển trên băng tải, hàng nhỏ ở dưới là băng tải. Gỗ trong thùng chứa được sắp xếp thành hàng.

- Ứng dụng trong PLC cơ bản: Điều khiển tổ hợp logic.

- Ứng dụng trong PLC nâng cao: Điều khiển trình tự.

3.4.2. Cách vận hành mô hình

Sau khi đã nối dây mô hình với PLC xong, thực hiện viết chương trình theo bài tập đưa ra [có thể tự kiểm tra các ngõ vào/ra bằng phần mềm (đối với S7-200 dùng bảng Status chart)] và sau đó thực hiện mô phỏng với mô hình.

Chỉ có thể đặt gỗ được trên băng tải nếu băng tải hoạt động. Việc đặt những thanh gỗ dài ngắn khác nhau được tạo ra bằng cách ấn nút "Khởi động" lâu hay ngắn.

Khi những thanh gỗ này đi qua cảm biến quang thì các cảm biến này sẽ thay đổi trạng thái. Các cần gạt nếu được kích hoạt thì các thanh gỗ ngay vị trí của nó sẽ biến mất và sau đó một thanh LED trong hộp sáng lên cho biết gỗ đã vào trong hộp. Tùy theo bài tập đặt ra mà có thể phát hiện được người thực hành viết chương trình sai hay đúng.

3.4.3. Bảng ký hiệu

Ký hiệu	Địa chỉ	Chú thích
Khởi động	I0.0	Khởi động hệ thống, thường hở.
B1	I0.1	Cảm biến quang, thường đóng
B2	I0.2	Cảm biến quang, thường đóng
B3	I0.3	Cảm biến quang, thường đóng
B4	I0.4	Cảm biến quang, thường đóng
B5	I0.5	Cảm biến quang, thường đóng
B6	I0.6	Cảm biến quang, thường đóng
B7	I0.7	Công tắc hành trình, thường đóng
B8	I1.0	Công tắc hành trình, thường đóng
S I	I1.1	Nút nhấn đưa gỗ vào thùng I
S II	I1.2	Nút nhấn đưa gỗ vào thùng II
Tự động	I1.3	Công tắc chọn chế độ tự động
Tay	I1.4	Công tắc chọn chế độ tay
Băng tải	Q0.0	Băng tải vận chuyển gỗ
Cần gạt I	Q0.1	Cần gạt đưa gỗ vào thùng I
Cần gạt II	Q0.2	Cần gạt đưa gỗ vào thùng II
Đèn báo	Q0.3	Đèn báo băng tải sẵn sàng nhận gỗ

Bài tập thực hành:

*** Mô tả đề bài:**

Mô hình đo chiều dài và sắp xếp vật liệu được dùng để mô phỏng việc sắp xếp các thanh gỗ có chiều dài ngắn khác nhau trên băng tải vào các thùng khác nhau.

Hệ thống có thể hoạt động ở hai chế độ: tự động và bằng tay.

- Chế độ tự động:

Khi "đèn báo" sáng báo hiệu hệ thống sẵn sàng làm việc. Bằng cách nhấn nút "Khởi động" thì "đèn báo" tắt và một tín hiệu khởi động được tạo ra. Các thanh gỗ đơn được đặt lên băng tải (bằng nút ấn khởi động) và băng tải chuyển động.

Chiều dài của thanh gỗ được nhận biết bằng các cảm biến quang B1, B2, B3. Điều này có nghĩa:

- + Cảm biến B1 tác động tương ứng gỗ ngắn.
- + Cảm biến B1 và B2 tác động tương ứng gỗ trung bình.
- + Cảm biến B2, B2 và B3 tác động tương ứng gỗ dài.

Khi gỗ ngắn đến cảm biến B7 thì "Tay gạt 1" sẽ đẩy thanh gỗ này vào thùng 1. Khi gỗ trung bình đến cảm biến B8 thì "Tay gạt 2" sẽ đẩy thanh gỗ này vào thùng 2. Gỗ dài thì được di chuyển tiếp tục đến khâu xử lý kế tiếp. Tay gạt 1 và tay gạt 2 được sử dụng bằng khí nén, được điều khiển khoảng 1 giây và sau đó trở về vị trí cơ bản của nó.

Sau khi sự sắp xếp thành công (tương ứng với các cảm biến quang B4, B5 và B6) thì thiết bị tự động phát tín hiệu khởi động tiếp theo và băng tải lại vận chuyển gỗ tiếp tục.

- Chế độ bằng tay:

Ở chế độ tay mỗi thanh gỗ được xử lý xong thì yêu cầu khởi động lại hệ thống bằng tay. Tín hiệu khởi động chỉ được phép xử lý nếu việc điều khiển trước đây được báo bằng đèn. Ngay sau khi sắp xếp thành công thì đèn báo lại sáng.

Tay gạt 1 và 2 được điều khiển bằng tay từ nút nhấn điều khiển.

Ghi chú: Đây chỉ là một khâu sắp xếp gỗ, và gỗ được đặt vào băng tải nhờ nút nhấn khởi động. Điều này có nghĩa nút nhấn khởi động vừa đóng vai trò khởi động vừa là nơi cung cấp gỗ cho băng tải.

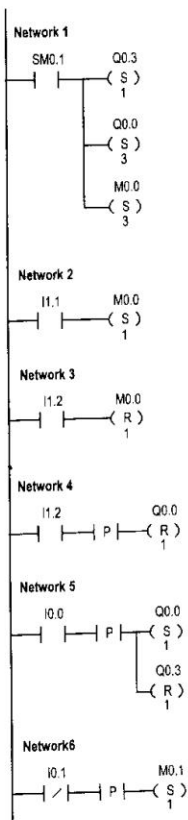
*** Yêu cầu:**

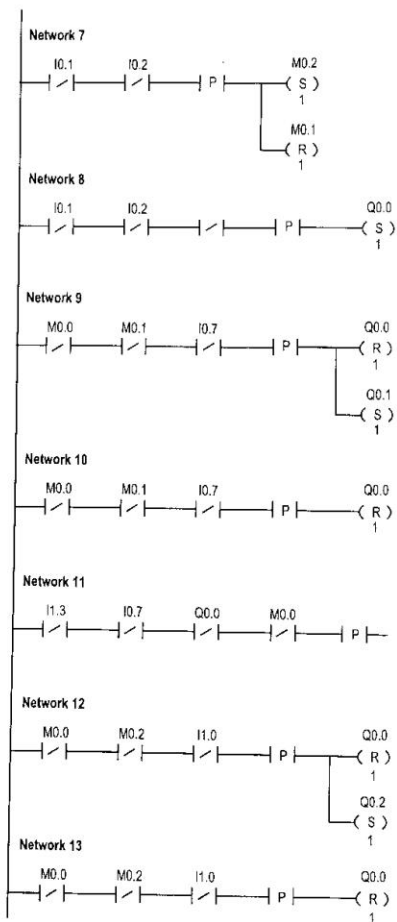
- Hãy viết chương trình điều khiển hệ thống này và kiểm tra bằng mô hình ở hai cách:
 - a) Dùng tổ hợp logic.
 - b) Dùng phương pháp điều khiển trình tự.
- Kết nối kiểm tra trên bản thí nghiệm.

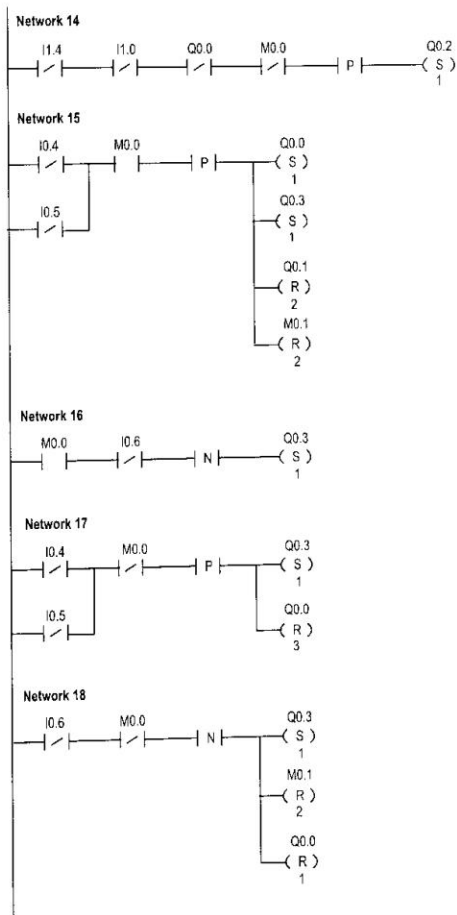
** Mục tiêu bài tập:*

Sau khi học xong bài học, sinh viên có khả năng:

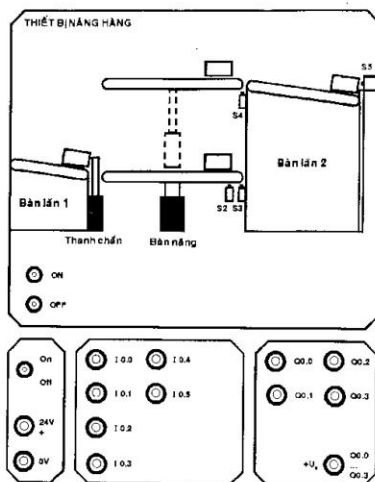
- Lập trình được chương trình trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được các đầu vào/ra cho PLC trên bàn thí nghiệm.
- Chương trình viết ở LAD:







3.5. Mô hình thiết bị nâng hàng



3.5.1. Mô tả

Mô phỏng một hệ thống nâng hàng bằng các đèn LED với nhiều màu sắc khác nhau. Hàng hóa từ bàn lần thấp được đưa lên cao sang bàn lần 2 nhờ vào bàn nâng. Hệ thống này thường thấy trong việc sắp xếp hàng hóa trong kho hoặc đưa hàng hoá vào các khoang chứa hàng của máy bay.

Ứng dụng:

- PLC cơ bản: Điều khiển tổ hợp logic.
- PLC nâng cao: Điều khiển trình tự.
- Kết nối dây như trình bày ở mục 2.

3.5.2. Cách vận hành mô hình

Sau khi đã nối dây mô hình với PLC xong, thực hiện viết chương trình theo bài tập đưa ra [có thể tự kiểm tra các ngõ vào/ra bằng phần mềm (đối với S7-200 dùng bảng Status chart)] và sau đó thực hiện mô phỏng với mô hình.

Vật thể trên bàn lần 1 xem như do một nơi khác chuyển đến. Cứ sau khi vật thể được đưa sang bàn nâng thì một vật thể trên bàn lần 1 lại xuất hiện. Vật thể trên bàn

nâng được đưa sang đầu bên kia của bàn nâng nhờ vào băng tải trên bàn nâng (tượng trưng bởi LED chạy xuôi). Khi vật thể được đưa sang bàn lăn 2 thì nó sẽ lăn đến cuối bàn và dừng lại 1s, sau đó tự biến mất.

3.5.3. Bảng ký hiệu

Ký hiệu	Địa chỉ	Chú thích
ON	I0.0	Khởi động hệ thống, thường hở
OFF	I0.1	Dừng hệ thống, thường đóng
S2	I0.2	Bảo hàng ở vị trí cuối bàn nâng, thường đóng
S3	I0.3	Giới hạn dưới bàn nâng, thường đóng
S4	I0.4	Giới hạn trên bàn nâng, thường đóng
S5	I0.5	Bảo hàng ở cuối bàn lăn 2
Thanh chắn	Q0.0	Chặn hàng hóa ở bàn nâng 1
Băng tải	Q0.1	Băng tải chuyển hàng
K1	Q0.2	Nâng hàng hoá lên
K2	Q0.3	Hạ bàn nâng xuống

Bài tập thực hành:

* Mô tả đề bài:

- Hàng hóa được đặt sẵn trên bàn lăn 1. Bàn nâng ở vị trí giới hạn dưới thì khi ấn nút khởi động "ON", băng tải trên bàn nâng hoạt động, đồng thời thanh chắn hạ xuống (sử dụng khí nén) khoảng 2s để hàng hóa được đưa sang bàn nâng. Sau đó thanh chắn trở về vị trí cũ.

- Khi hàng hóa đến vị trí cuối bàn nâng (S2), thì băng tải dừng. Khởi động từ K1 của động cơ M1 có điện kéo bàn nâng lên. Khi đến giới hạn trên thì bàn nâng dừng lại. Băng tải bắt đầu chuyển động đưa hàng sang bàn lăn 2.

- Khi hàng đến công tắc hành trình S5 thì băng tải dừng. Khởi động từ K2 của động cơ M1 có điện hạ bàn nâng xuống, đến giới hạn dưới thì dừng.

- Quá trình mới lại bắt đầu cho đến khi nào nhấn nút dừng "OFF".

* Yêu cầu:

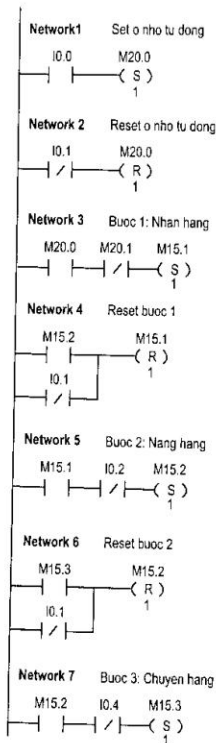
- Viết chương trình điều khiển thiết bị nâng trên và kiểm tra bằng mô hình theo hai cách:

- + Viết theo tổ hợp Logic.
- + Viết theo điều khiển trình tự.
- Kết nối kiểm tra trên bản thí nghiệm.

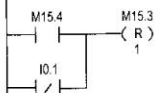
*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

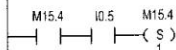
- Lập trình được chương trình trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được phần cứng trên bản thí nghiệm.
- Chương trình viết ở LAD:



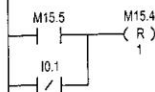
Network 8 Reset bước 3



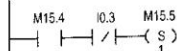
Network 9 Bước 4: Hạ bàn nâng



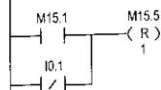
Network 10 Reset bước 5



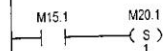
Network 11 Bước 5: Kết thúc



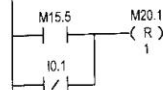
Network 12 Reset bước 5



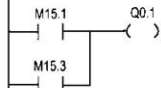
Network 13 Set ở nhà khởi động



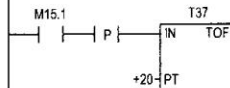
Network 14 Reset ở nhà khởi động



Network 15 Khởi động băng tải trên bàn nâng



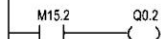
Network 16 Định thời gian thành chân có điện 2s



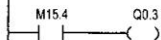
Network 17 Đóng điện cho thành chân



Network 18 Nâng hàng hoá



Network 19 Hạ bàn nâng



- Chương trình viết ở STL:

```

NETWORK 1      //Set o nhó tu động
//
LD      IO.0
S       M20.0, 1

NETWORK 2      //Reset o nhó tu động
//
LDN     IO.1
R       M20.0, 1

NETWORK 3      //Bước 1: Nhận hàng
//
LD      M20.0
AN      M20.1
S       M15.1, 1
  
```

```

NETWORK 4          //Reset buoc 1
//
LD      M15.2
ON      IO.1
R      M15.1, 1

NETWORK 5          //Buoc 2: Nang hang
//
LD      M15.1
AN      IO.2
S      M15.2, 1

NETWORK 6          //Reset buoc 2
//
LD      M15.3
ON      IO.1
R      M15.2, 1

NETWORK 7          //Buoc 3: Chuyen hang
//
LD      M15.2
AN      IO.4
S      M15.3, 1

NETWORK 8          //Reset buoc 3
//
LD      M15.4
ON      IO.1
R      M15.3, 1

NETWORK 9          //Buoc 4: Ha ban nang
//
LD      M15.4
A      IO.5
S      M15.4, 1

NETWORK 10         //Reset buoc 5
//
LD      M15.5
ON      IO.1
R      M15.4, 1

```

```

NETWORK 11      //Buoc 5: Ket thuc
//
LD      M15.4
AN      IO.3
S       M15.5, 1

NETWORK 12      //Reset buoc 5
//
LD      M15.1
ON      IO.1
R       M15.5, 1

NETWORK 13      //Set o nho khai dong
//
LD      M15.1
S       M20.1, 1

NETWORK 14      //Reset o nho khai dong
//
LD      M15.5
ON      IO.1
R       M20.1, 1

NETWORK 15      //Khoi dong bang tai tren ban nang
//
LD      M15.1
O       M15.3
=       Q0.1

NETWORK 16      //Dinh thoi gian thanh chan co dien 2s
//
LD      M15.1
EU
TOF     T37, +20

NETWORK 17      //Dong dien cho thanh chan
//
LD      T37
=       Q0.0

NETWORK 18      //Nang hang hoa
//
LD      M15.2
=       Q0.2

NETWORK 19      //Ha ban nang
//
LD      M15.4
=       Q0.3

```

3.6. Mô hình thiết bị đóng vỏ nước chai

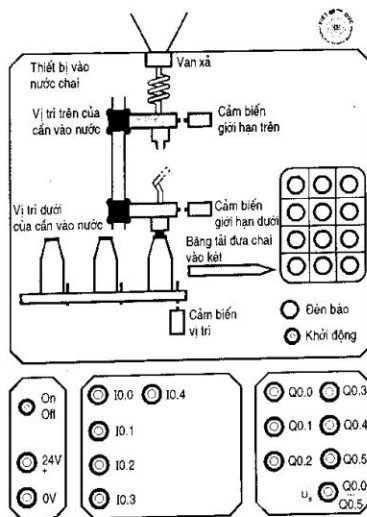
3.6.1. Mô tả

Mô phỏng một thiết bị đóng vỏ nước chai có các cảm biến, công tắc hành trình và sự chuyển động bằng các LED.

Ứng dụng:

- PLC cơ bản: Điều khiển tổ hợp logic.
- PLC nâng cao: Điều khiển trình tự.

Cách kết nối dây như sau:.



3.6.2. Cách vận hành mô hình

Sau khi đã nối dây mô hình với PLC xong, thực hiện viết chương trình theo bài tập đưa ra [có thể tự kiểm tra các ngõ vào/ra bằng phần mềm (đối với S7-200 dùng bảng Status chart)] và sau đó thực hiện mô phỏng với mô hình.

Hai chai bia bên phải được xem là chai rỗng. Chai ở vị trí thứ ba được xem như chai đã được đưa đến đúng vị trí. Nước trong chai dâng lên được mô phỏng bằng đèn LED sáng dần.

Tùy vào sự sáng dân này mà có thể định thời gian làm đầy chai. Khi chai đã được đổ đầy nước và nếu băng tải vận chuyển chai hoạt động thì chai đầy sẽ tự động được chuyển sang băng tải đưa chai vào kết. Một tín hiệu sẽ phát ra nếu chai đã đứng ở vị trí trong kết. Khi kết đạt đến 12 thì nó không thể tự Reset được. Để có thể xóa các LED trong kết này phải ấn nút "Khởi động".

Để hoạt động giống thực tế thì khi cần vào nước đến miệng chai phải dừng lại 1s để ổn định.

3.6.3. Bảng ký hiệu

Ký hiệu	Địa chỉ	Chú thích
S1	I0.0	Giới hạn trên của cần vào nước, thường đóng
S2	I0.1	Giới hạn dưới của cần vào nước, thường đóng
S3	I0.2	Cảm biến vị trí chai, thường hở
S4	I0.3	Khởi động hệ thống, thường hở
S5	I0.4	Chai đứng vị trí trong kết, thường hở
K1	Q0.0	Van xả nước
K2	Q0.1	Hạ cần vào nước xuống
K3	Q0.2	Nâng cần vào nước lên
K4	Q0.3	Băng tải vận chuyển chai rỗng
K5	Q0.4	Đèn báo kết đầy

Bài tập thực hành:

* Mô tả đề bài

- Trước khi vận hành thiết bị vô nước chai thì các chai rỗng phải được đặt lên băng tải. Nếu sau đó nút nhấn khởi động (I0.3) được tác động, thì băng tải sẽ vận chuyển chai rỗng với thời gian trì hoãn ban đầu là 1s.

- Băng tải dừng lại khi có một chai đến cảm biến vị trí (I0.2).

- Bây giờ cần vô nước sẽ hạ từ trên xuống, khi đến giới hạn dưới (I0.1) thì dừng lại, sau đó 1s van xả sẽ được mở đổ nước vào chai, van xả sẽ đóng lại khi chai đầy thời gian làm đầy kéo dài khoảng 3s.

- Sau khi van xả đóng lại 1s, cần vô nước được nâng lên, đến giới hạn trên (I0.0) thì dừng lại. Sau đó 1s, băng tải vận chuyển chai rỗng lại tiếp tục và quá trình cứ thế lặp lại.

- Chai đã đổ đầy nước được đưa sang băng tải đưa chai vào kết khi băng tải chai rỗng hoạt động, khi chai đứng vị trí trong kết thì có một tín hiệu phát ra (I0.4).

- Quá trình được lập đi lập lại cho đến khi số lượng chai trong két đủ 12 chiếc, lúc này đèn báo sáng lên và hệ thống dừng lại. Quá trình mới lại bắt đầu khi nút nhấn "khởi động" được tác động.

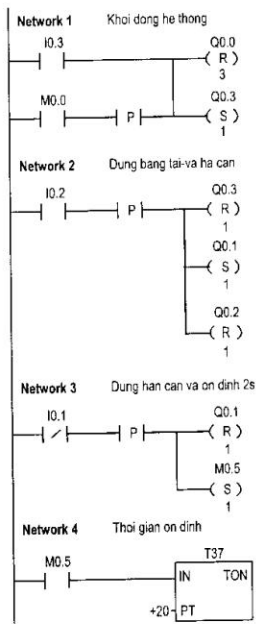
*** Yêu cầu:**

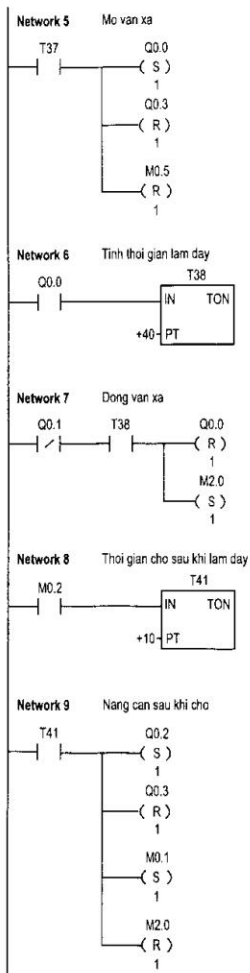
- Viết chương trình điều khiển trên phần mềm Microwin 4.0.
- Kết nối kiểm tra trên bàn thí nghiệm.

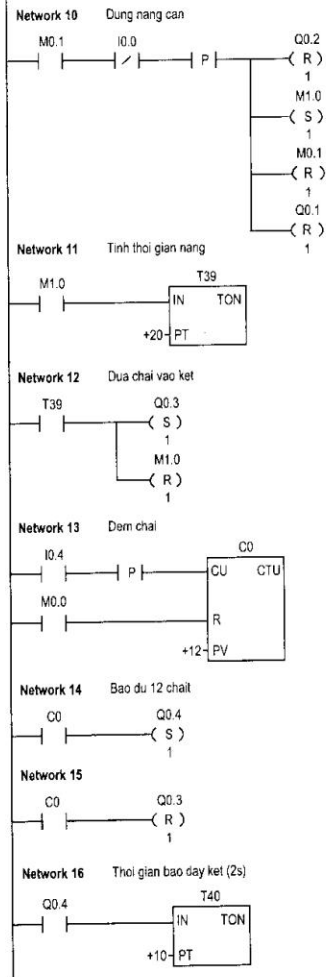
*** Mục tiêu bài tập:**

Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình trên phần mềm Microwin 4.0 theo yêu cầu bài toán
- Kết nối được phần cứng trên bàn thí nghiệm
- Chương trình viết ở LAD:







- Chương trình được viết ở STL:

```
NETWORK 1          //khởi động hệ thống
// Khi nhận lệnh (NO) thì băng tải sẽ hoạt động
LD      I0.3
LD      M0.0
EU
OLD
R        Q0.0, 3
S        Q0.3, 1

NETWORK 2          //dừng băng tải-và hệ cần
//
LD      I0.2
EU
R        Q0.3, 1
S        Q0.1, 1
R        Q0.2, 1

NETWORK 3          //Dừng hệ cần và ổn định 2s
//
LDN     I0.1
EU
R        Q0.1, 1
S        M0.5, 1

NETWORK 4          //thời gian ổn định
//
LD      M0.5
TON     T37, +20

NETWORK 5          //Mở van xả
//
LD      T37
S        Q0.0, 1
R        Q0.3, 1
R        M0.5, 1

NETWORK 6          //Tính thời gian làm đầy
//
LD      Q0.0
TON     T38, +40

NETWORK 7          //Đóng van xả
//
LDN     Q0.1
A        T38
R        Q0.0, 1
S        M2.0, 1

NETWORK 8          //Thời gian chờ sau khi làm đầy
//
LD      M2.0
TON     T41, +10
```

```

NETWORK 9          //Nang can sau khi cho
//
LD      T41
S       Q0.2, 1
R       Q0.3, 1
S       M0.1, 1
R       M2.0, 1

NETWORK 10         //Dung nang can
//
LD      M0.1
AN      ID.0
EU
R       Q0.2, 1
S       M1.0, 1
R       M0.1, 1
R       Q0.1, 1

NETWORK 11         //Tinh thoi gian nang
//
LD      M1.0
TON     T39, +20

NETWORK 12         //Dua chai vao ket
//
LD      T39
S       Q0.3, 1
R       M1.0, 1

NETWORK 13         //dem chai
//
LD      ID.4
EU
LD      M0.0
CTU     C0, +12

NETWORK 14         //bao du 12 chai
//
LD      C0
S       Q0.4, 1

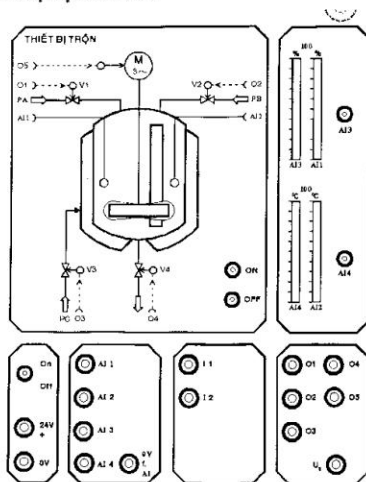
NETWORK 15
LD      C0
R       Q0.3, 1

NETWORK 16         //thoi gian bao day ket (2s)
//
LD      Q0.4
TON     T40, +10

NETWORK 17         //Lap lai chu ky
//
LD      T40
=       M0.0
R       Q0.4, 1

```

3.7. Mô hình thiết bị trộn hóa chất



3.7.1. Mô tả

Mô phỏng một thiết bị trộn định lượng, sử dụng các cảm biến analog để đo lượng chất lỏng chứa trong bình và đo nhiệt độ trong bình. Nhiệt độ và lượng chất lỏng trong bình có thể được điều chỉnh trước bằng các phím chỉnh định bên ngoài.

Ứng dụng trong PLC nâng cao (điều khiển tuần tự, xử lý tín hiệu analog, các phép toán, bộ điều chỉnh 2 điểm...).

3.7.2. Cách vận hành mô hình

Đây là mô hình thiết bị trộn dùng các cảm biến analog. Mô hình này được thiết kế cho các PLC có gắn các *môđun ngõ vào analog có ngõ vào là điện áp (0-10V)*. Tùy theo loại PLC và cũng tùy theo các môđun ngõ vào analog hiện có mà chúng ta có thể đưa ra một số bài tập phù hợp.

Cảm biến đo nhiệt độ ở mô hình này được thiết kế phụ thuộc vào V3. Có nghĩa là nếu V3 được cung cấp điện thì nhiệt độ tăng dần từ khoảng 5°C lên và đến cực đại là 100°C. Nếu V3 mất điện thì nhiệt độ tự động giảm xuống từ từ.

Đối với cảm biến đo lượng chất lỏng, nó chỉ đo được tại mỗi thời điểm và chỉ với một bơm chất lỏng. Tức là nếu như bơm A hoạt động thì bơm B phải ngưng và ngược lại.

3.7.3. Bảng ký hiệu

Ký hiệu	Địa chỉ	Chú thích
AI1	AIW0	Cảm biến nhận biết lượng chất lỏng trong bình
AI2	AIW2	Cảm biến nhận biết nhiệt độ trong bình
AI3	AIW4	Đặt chỉnh trước lượng chất lỏng cần thiết
AI4	AIW6	Đặt chỉnh trước nhiệt độ cần thiết
ON	I0.0	Khởi động hệ thống, thường hở
OFF	I0.1	Dừng hệ thống, thường đóng
V1	Q0.0	Bơm chất lỏng A
V2	Q0.1	Bơm chất lỏng B
V3	Q0.2	Cung cấp nhiệt cho bình trộn
V4	Q0.3	Van xả chất lỏng
M1	Q0.4	Quạt trộn hóa chất

Bài tập thực hành:

*** Mô tả đề bài:**

- Có hai loại chất lỏng A và B cần được trộn với nhau theo tỷ lệ 1/3. Nhiệt độ cần thiết để trộn hai chất này được đặt chỉnh ở AI4. Lượng chất lỏng muốn trộn được đặt chỉnh ở AI3.

- Khi ấn nút khởi động "ON" thì bơm B hoạt động trước. Sau khi chất lỏng B đạt đến mức cần thiết thì dừng lại (được nhận biết bởi cảm biến AI1). Nhiệt độ trong bồn trộn được tăng dần lên đến giá trị đặt (sai số cho phép là $\pm 10\%$, được nhận biết bởi cảm biến AI2) thì bơm chất lỏng A hoạt động, đồng thời quạt trong bồn trộn cũng quay.

- Khi chất lỏng A đã đổ đúng lượng cho phép thì bơm A dừng. Quạt trộn tiếp tục quay trong khoảng thời gian 10s nữa thì dừng lại. Sau đó van xả tự động mở ra để xả chất lỏng đã trộn vào bồn chứa.

- Quá trình mới lại bắt đầu, nếu nút nhấn "ON" được ấn. Hệ thống có thể dừng nếu nhấn nút dừng "OFF".

*** Yêu cầu:**

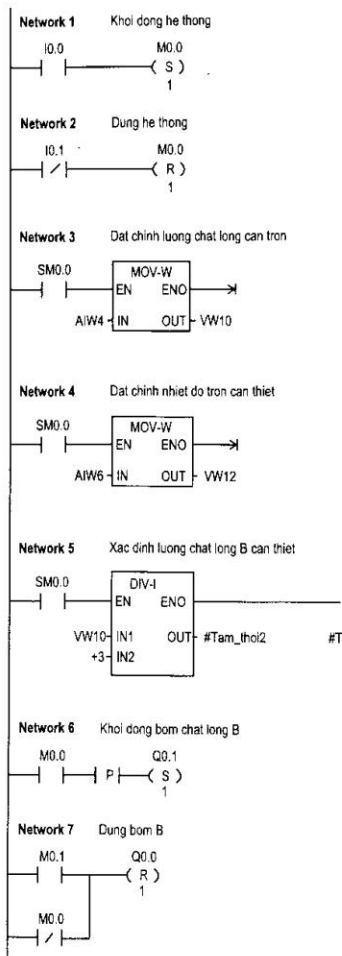
- Viết chương trình điều khiển trên phần mềm Microwin 4.0.
- Kết nối kiểm tra trên bàn thí nghiệm.

*** Mục tiêu bài tập:**

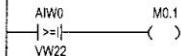
Sau khi học xong bài học, sinh viên có khả năng:

- Lập trình được chương trình trên phần mềm Microwin 4.0 theo yêu cầu bài toán.
- Kết nối được phần cứng trên bàn thí nghiệm.

- Chương trình viết ở LAD:



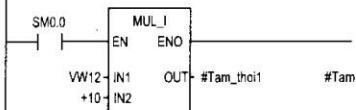
Network 8 Xác định lượng chất lỏng B đã đủ mức cần thiết



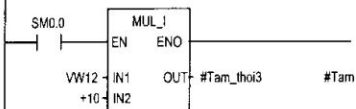
Network 9 Xác định lượng chất lỏng A đã đủ mức cần thiết



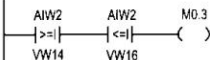
Network 10 Giới hạn dưới nhiệt độ đạt



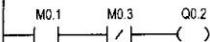
Network 11 Giới hạn trên nhiệt độ đạt



Network 12 Xác định xem nhiệt độ bồn trộn đã đạt đến giá trị đạt?



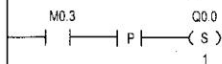
Network 13 Cấp nhiệt cho bồn trộn



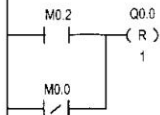
Network 14 Bật quạt trộn



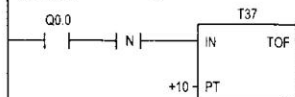
Network 15 Bơm A hoạt động



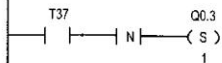
Network 16 Dừng bơm A khi lượng chất lỏng đã đủ



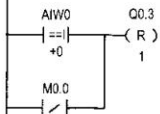
Network 17 Định thời gian trộn



Network 18 Xả chất lỏng đã trộn



Network 19 Dừng van xả khi chất lỏng trong thùng đã xả hết



- Chương trình viết ở STL:

```

NETWORK 1      //Khởi động hệ thống
//
LD      IO.0
S       MO.0, 1

NETWORK 2      //Dừng hệ thống
//
LDN     IO.1
R       MO.0, 1

NETWORK 3      //Đạt chính lượng chất lỏng cần trộn
//
LD      SMO.0
MOVW    AIW4, VW10

NETWORK 4      //Đạt chính nhiệt độ trộn cần thiết
//
LD      SMO.0
MOVW    AIW6, VW12

NETWORK 5      //Xác định lượng chất lỏng B cần thiết
//
LD      SMO.0
MOVW    VW10, #Tam_thoi2
AENO
/I      +3, #Tam_thoi2
AENO
MOVW    #Tam_thoi2, VW22
*1      +2, VW22

NETWORK 6      //Khởi động bơm chất lỏng B
//
LD      MO.0
EU
S       Q0.1, 1

NETWORK 7      //Dừng bơm B
//
LD      MO.1
ON      MO.0
R       Q0.0, 1

NETWORK 8      //Xác định lượng chất lỏng B đã đủ mức cần thiết
//
LDU>=   AIW0, VW22
=       MO.1

```

```

NETWORK 9          //Xac dinh luong chat long A da du muc can thiet
//
LDW>= AIW0, VU10
=      MO.2

NETWORK 10         //Gioi han duoi nhiet do dat
//
LD      SMO.0
MOVW   VU12, #Tam_thoi1
AENO
*I      +10, #Tam_thoi1
AENO
MOVW   #Tam_thoi1, VU14
*I      +9, VU14

NETWORK 11         //Gioi han tren nhiet do dat
//
LD      SMO.0
MOVW   VU12, #Tam_thoi3
AENO
*I      +10, #Tam_thoi3
AENO
MOVW   #Tam_thoi3, VU16
*I      +9, VU16

NETWORK 12         //Xac dinh xem nhiet do bon tron da dat den gia tri dat?
//
LDW>= AIW2, VU14
AW<= AIW2, VU16
=      MO.3

NETWORK 13         //Cap nhiet cho bon tron
//
LD      MO.1
AN      MO.3
=      QO.2

NETWORK 14         //Bat quat tron
//
LD      MO.3
S      QO.4, 1

NETWORK 15         //Bom A hoat dong
//
LD      MO.3
EU
S      QO.0, 1

```

```

NETWORK 16      //Dung Bom A khi lượng chất lỏng đã đủ
//
LD      MO.2
ON      MO.0
R        Q0.0, 1

NETWORK 17      //Đinh thời gian tron
//
LD      Q0.0
ED
TOF      T37, +10

NETWORK 18      //Xả chất lỏng đã tron
//
LD      T37
ED
S        Q0.3, 1

NETWORK 19      //Dòng van xả khi chất lỏng trong thùng đã xả hết
//
LDW=     AIW0, +0
ON      MO.0
R        Q0.3, 1

```

CÂU HỎI VÀ BÀI TẬP

1. Câu hỏi

Câu 1. Nêu cách kết nối đầu vào và đầu ra trên các mô hình thí nghiệm dùng trong PLC S7-200.

Câu 2. Nêu tác dụng của các cảm biến trong mô hình đo chiều dài vật liệu

Câu 3. Trong mô hình thang máy, khi công tắc hành trình giới hạn trên và dưới khi lắp bị đổi chỗ cho nhau thì khi vận hành hiện tượng gì sẽ xảy ra? Cách khắc phục trên chương trình lập trình?

Câu 4. Trong mô hình điều khiển động cơ chạy Y/ Δ , đầu vào I0.4 có thể sử dụng dạng tiếp điểm thường đóng hay không?

Câu 5. Trong mô hình điều khiển xe chuyển nguyên liệu, đầu vào I0.5 có thể sử dụng dạng tiếp điểm thường mở hay không?

2. Bài tập:

Bài 1. Viết chương trình điều khiển dạng Set, reset ở mô hình đo chiều dài vật liệu và mô hình thiết bị nâng hàng.

Bài 2. Cho bốn trộn có động cơ hoạt động như sau:

- Khi ấn nút "Start", thì động cơ quay phải ở tốc độ thấp trong thời gian 10s → dừng 5s → quay trái 10s → dừng 5s.

- Quá trình cứ thế lặp lại. Sau khoảng 20 lần lặp thì động cơ dừng 10s và sau đó quay phải ở tốc độ thấp, được 5s thì chuyển sang quay ở tốc độ cao khoảng 30s thì dừng hẳn.

- Bồn trộn có thể dừng bằng cách ấn nút dừng "Stop".

* Yêu cầu:

- Quy ước các đầu vào/ra.

- Viết chương trình ở dạng LAD và STL.

- Nạp vào PLC và chạy thử.

TÀI LIỆU THAM KHẢO

1. Tăng Văn Mùi, Nguyễn Tiến Dũng. *Điều khiển logic lập trình PLC*. NXB Thống kê, 2006.
2. Trần Thế San, TS. Nguyễn Ngọc Phương. *Hướng dẫn thiết kế mạch và lập trình PLC*. NXB Đà Nẵng.
3. Nguyễn Thu Thiên, Mai Xuân Vũ. *Sổ tay hướng dẫn lập trình PLC*. NXB Trẻ.
4. Nguyễn Đoàn Phước, Phan Xuân Minh. *Tự động hóa với Simatic S7-200*. NXB Nông Nghiệp.
5. *Kỹ thuật điều khiển lập trình*. Trung Tâm Việt – Đức.

MỤC LỤC

	Trang
<i>Lời nói đầu</i>	3
Chương trình mô đun đào tạo lập trình PLC	4
Bài 1: ĐẠI CƯƠNG VỀ ĐIỀU KHIỂN LẬP TRÌNH	
1. Tổng quan về điều khiển	7
1.1. Điều khiển nối cứng và điều khiển lập trình	7
1.2. Cấu trúc của một PLC	9
2. Thiết bị lập trình S7-200	10
2.1. Địa chỉ các ngõ vào/ra	10
2.2. Phần chữ chỉ vị trí và kích thước của ô nhớ	10
2.3. Phần số chỉ địa chỉ của byte hoặc bit trong miền nhớ đã xác định	10
2.4. Cấu trúc bộ nhớ của S7-200	12
2.5. Xử lý chương trình	13
2.6. Cài đặt và sử dụng phần mềm Step 7 Microwin 4.0	16
Câu hỏi và bài tập	28
Bài 2: CÁC PHÉP TOÁN NHỊ PHÂN CỦA PLC	
1. Các liên kết logic	30
1.1. Các lệnh vào/ra và các lệnh tiếp điểm cơ bản	30
1.2. Các lệnh liên kết logic cơ bản	31
1.3. Liên kết các công logic cơ bản	33
2. Các lệnh ghi/xóa giá trị tiếp điểm	41
2.1. Lệnh SET (S) và RESET (R) trong S7-200	41
2.2. Timer	45
2.3. Bộ đếm (Counter)	59
Câu hỏi và bài tập	66
Bài 3: CÁC PHÉP TOÁN SỐ CỦA PLC	
1. Chức năng truyền dẫn	68
1.1. Truyền Byte, Word, Doubleword	68
1.2. Truyền 1 vùng nhớ dữ liệu	70

2. Chức năng so sánh, dịch chuyển, chuyển đổi	72
2.1. Chức năng so sánh	72
2.2. Chức năng dịch chuyển	80
2.3. Chức năng chuyển đổi	83
2.4. Chức năng toán học	85
3. Đồng hồ thời gian thực	91
Câu hỏi và bài tập	103
Bài 4: LẬP DẠT MÔ HÌNH ĐIỀU KHIỂN BẰNG PLC	
1. Giới thiệu	104
2. Cách kết nối dây	105
3. Các mô hình và bài tập ứng dụng	107
3.1. Mô hình thang máy xây dựng	107
3.2. Mô hình điều khiển động cơ Y- Δ	118
3.3. Mô hình xe chuyển nguyên liệu	122
3.4. Mô hình đo chiều dài và sắp xếp vật liệu	130
3.5. Mô hình thiết bị nâng hàng	136
3.6. Mô hình thiết bị đóng vỏ nước chai	143
3.7. Mô hình thiết bị trộn hóa chất	150
Câu hỏi và bài tập	157
TÀI LIỆU THAM KHẢO	159

GIÁO TRÌNH LẬP TRÌNH PLC

(Tái bản)

Chịu trách nhiệm xuất bản :

TRỊNH XUÂN SƠN

Biên tập : ĐINH THỊ PHƯỢNG

Chế bản : TRẦN KIM ANH

Sửa bản in : ĐINH THỊ PHƯỢNG

Trình bày bìa : VŨ BÌNH MINH

In 300 cuốn khổ 19 x 27cm tại Xưởng in Nhà xuất bản Xây dựng. Giấy chấp nhận đăng ký kế hoạch xuất bản số 36-2013/CXB/1108-158/XD ngày 5-1-2013. Quyết định xuất bản số 185-2013/QĐXB ngày 21-8-2013. In xong nộp lưu chiểu tháng 8-2013.