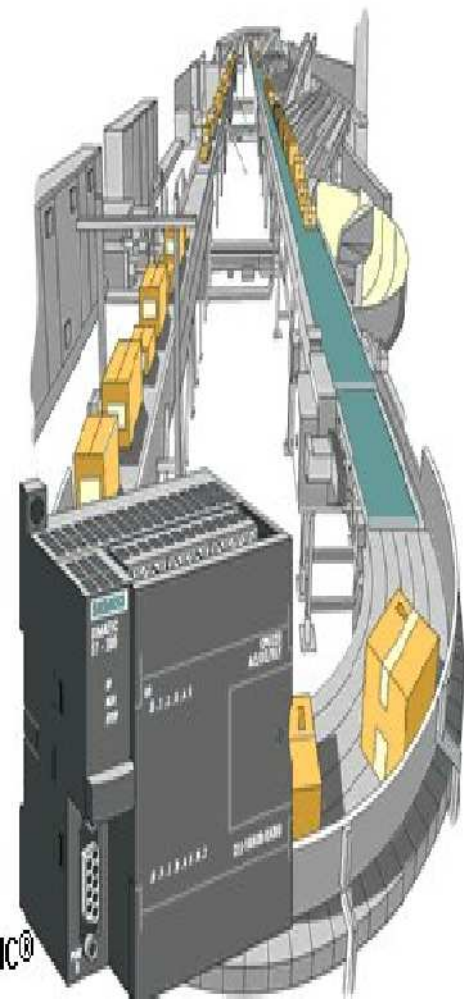


LẬP TRÌNH PLC



Mục tiêu:

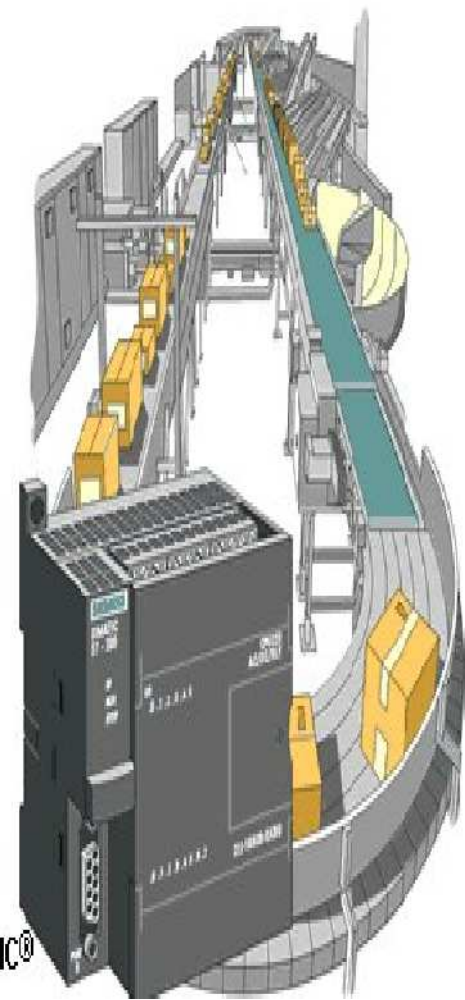
- Hiểu được cấu tạo PLC, hệ thống điều khiển sử dụng PLC
- Ghép nối được các phần tử vào/ ra với PLC
- Lập trình được cho PLC S7 – 200
- Có khả năng tự nghiên cứu để lập trình cho các loại PLC khác
- Thiết kế hệ thống điều khiển đơn giản sử dụng PLC

Nội dung:

- Tổng quan về PLC.
- Các phần tử vào/ra tín hiệu.
- Thiết bị điều khiển khả trình PLC S7-200 của SIEMENS
- Hệ lệnh của S7 – 200
- Điều khiển tuần tự

Điều kiện tiên quyết:

- Sinh viên đã học qua các học phần: Điện tử cơ bản, Kỹ thuật xung số, Điều khiển logic, Kỹ thuật Lập trình.



Chương 1: Tổng quan về PLC

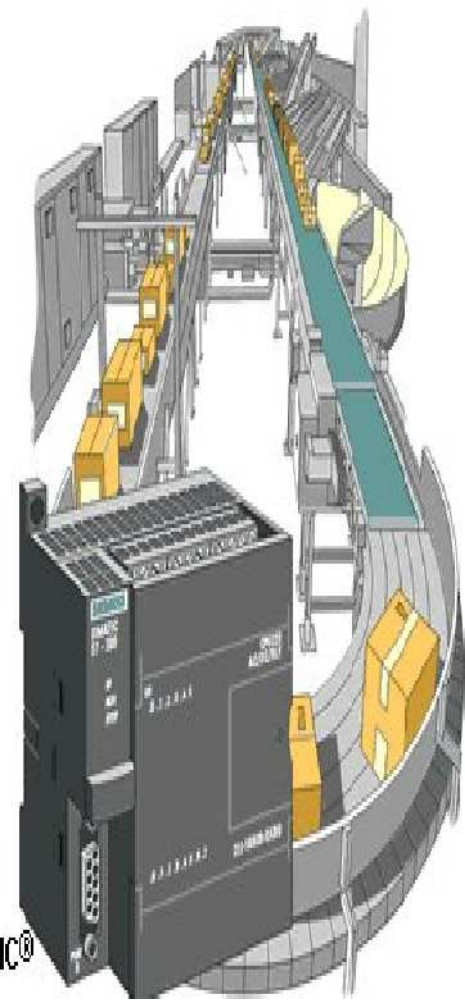
1.1. Lịch sử ra đời

1.2. PLC là gì?

1.3. Cấu tạo PLC.

Mục tiêu:

-Trang bị cho sinh viên kiến thức tổng quan về PLC từ lịch sử phát triển, cấu trúc bên trong cũng như cách thức hoạt động, ứng dụng của PLC trong lĩnh vực điều khiển tự động.



Chương 2: Các phần tử vào/ra tín hiệu. (12LT + 2 BT)

2.1. Các khái niệm cơ bản:

2.2. Các phần tử đầu vào:

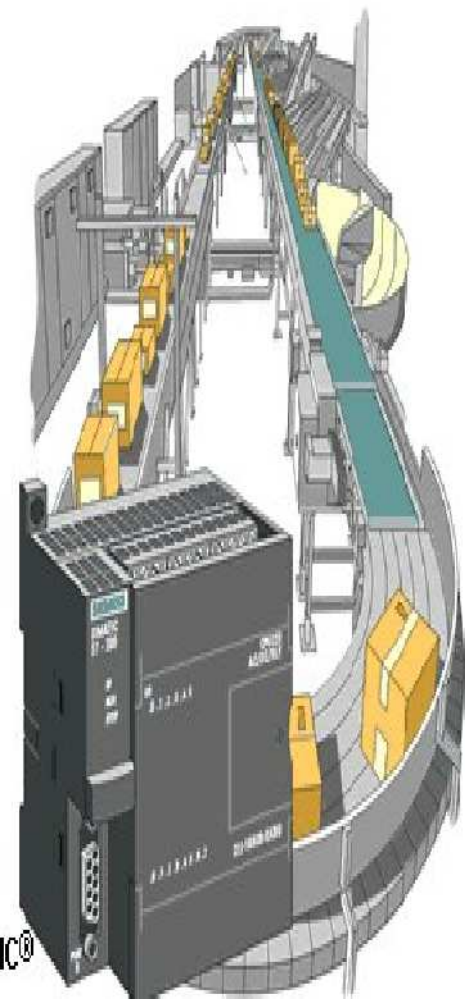
2.3. Các phần tử đầu ra:

2.4. Ghép nối phần tử vào ra với PLC

2.5. Các modul trong hệ thống điều khiển PLC

Mục tiêu:

- Trang bị cho sinh viên kiến thức các phần tử vào ra của PLC như: Cảm biến, nút nhấn, relay, contactor, valve, phitong, biến tần...
- Sinh viên kết nối được các phần tử vào ra với PLC



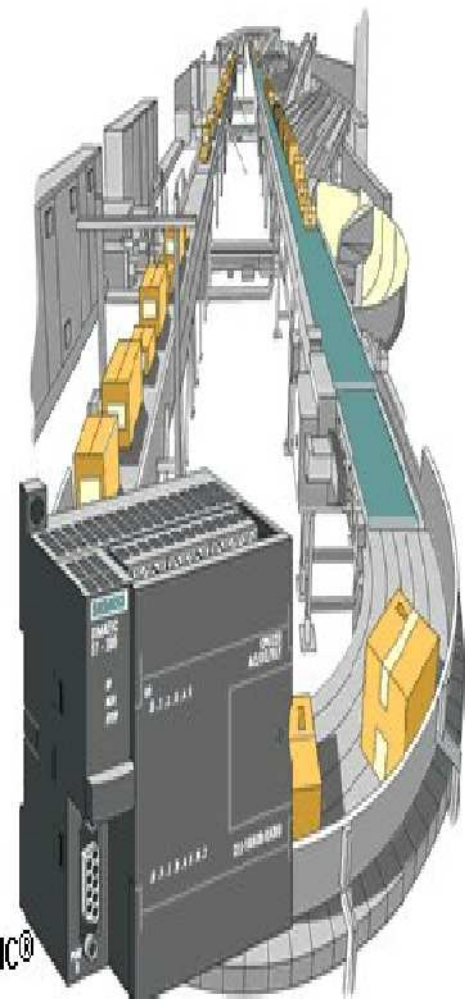
Chương 3: Thiết bị điều khiển khả trình PLC S7 200 của SIEMENS

3.1. Các thành phần của một hệ PLC S7 200.

3.2. Phần cứng của PLC S7 200

Mục tiêu:

- Trang bị cho sinh viên kiến thức ban đầu về cấu hình của PLC cũng như cổng truyền thông mà PLC đang giao tiếp với bên ngoài (Như PC, TD, PG...).
- Giới thiệu về cấu trúc bộ nhớ của PLC để sinh viên có thể dễ dàng lập trình và xử lý dữ liệu sau này.



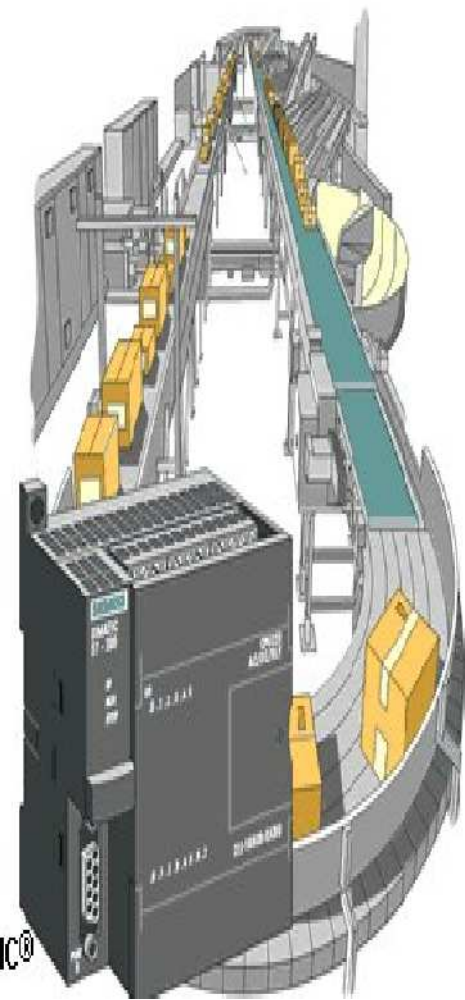
Chương 4: Hệ lệnh của S7 - 200

4.1. Phương pháp lập trình cho PLC.

4.2. Tập lệnh S7 200.

Mục tiêu:

- Trang bị cho sinh viên kiến thức về các thành phần cơ bản để lập trình và nạp chương trình cho PLC.
- Trang bị cho sinh viên các loại ngôn ngữ để lập trình cho PLC.
- Sinh viên lập trình được cho PLC S7-200 bằng các hàm có trong thư viện của phần mềm Step7 MicroWin và mở rộng để sinh viên có thể lập trình cho các loại PLC của các hãng khác.



Chương 5: Điều khiển tuần tự

5.1. Giới thiệu chung

5.2. Thiết kế chương trình

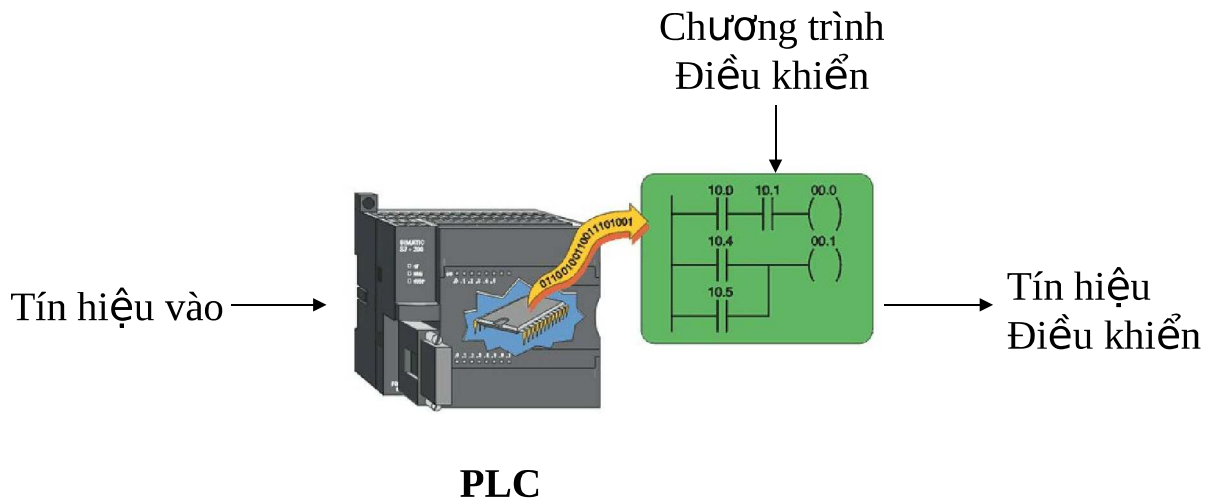
Mục tiêu:

- Trang bị cho sinh viên kiến thức về các thành phần cơ bản để lập trình và nạp chương trình cho PLC.
- Trang bị cho sinh viên các loại ngôn ngữ để lập trình cho PLC.
- Sinh viên lập trình được cho PLC S7-200 bằng các hàm có trong thư viện của phần mềm Step7 MicroWin và mở rộng để sinh viên có thể lập trình cho các loại PLC của các hãng khác.

TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

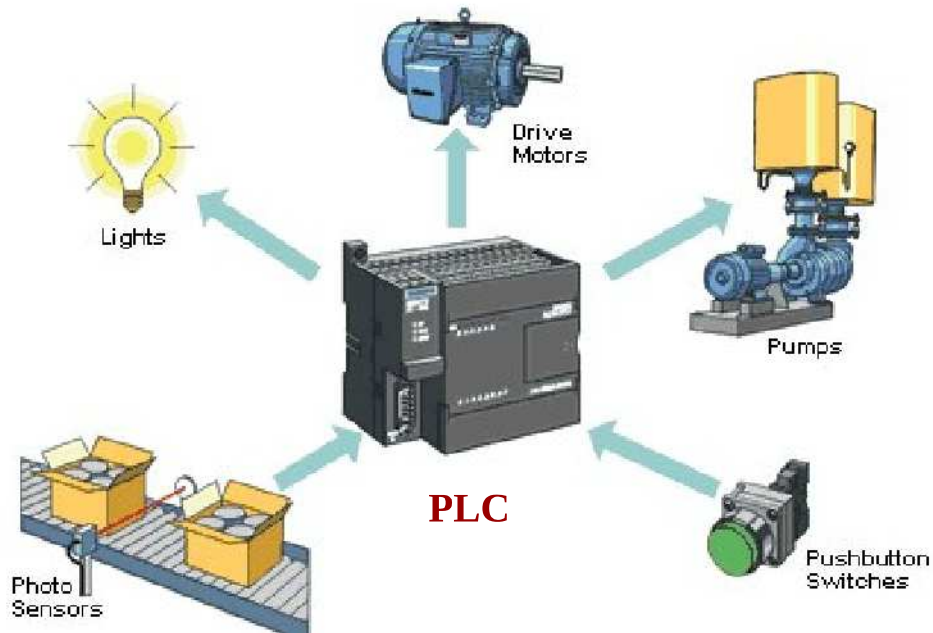
PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**



TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**



TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programable Logic Controler) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

Hệ điều khiển cơ điện

Các phần tử đầu vào



Bộ điều khiển



Phần tử chấp hành

Nút nhấn

Công tắc

Công tắc hành trình

Cảm biến

...

...

Rơ le

Công tắc tơ

Relay thời gian

bộ đếm

...

...

Động cơ

Công tắc tơ

Vale thuỷ lực, khí nén

Bộ hiển thị

...

...

TỔNG QUAN VỀ PLC

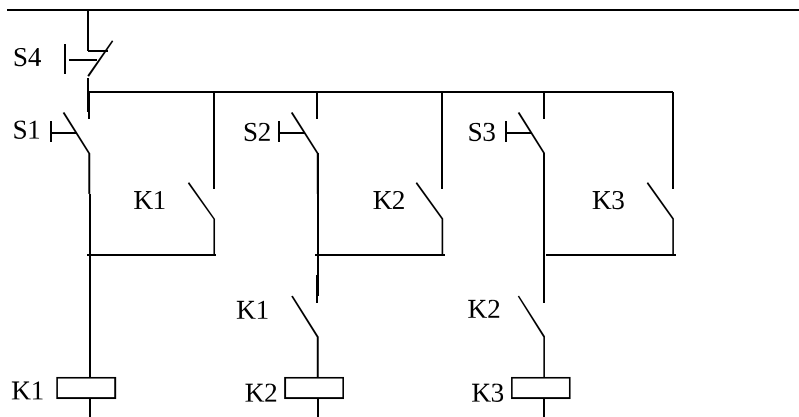
1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

Hệ điều khiển cơ

Ví dụ: Sử dụng các nút nhấn để điều khiển 3 máy bơm nước hoạt động tuần tự bằng khởi động từ.



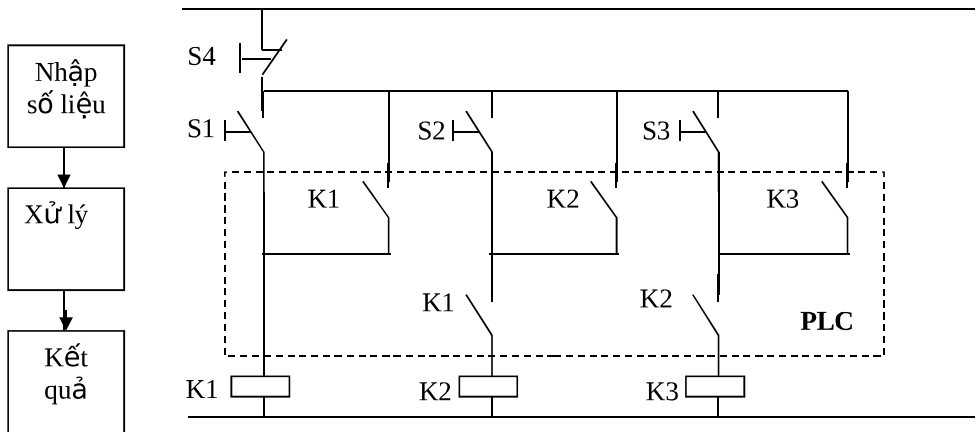
TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

*Hệ điều khiển sử dụng
PLC*



TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programable Logic Controler) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

Hệ điều khiển sử dụng PLC

Các phần tử đầu vào



Bộ điều khiển



Phần tử chấp hành

Nút nhấn

Công tắc

Công tắc hành trình

Cảm biến

...

...

PLC

Động cơ

Công tắc tơ

Vale thuỷ lực, khí nén

Bộ hiển thị

...

...

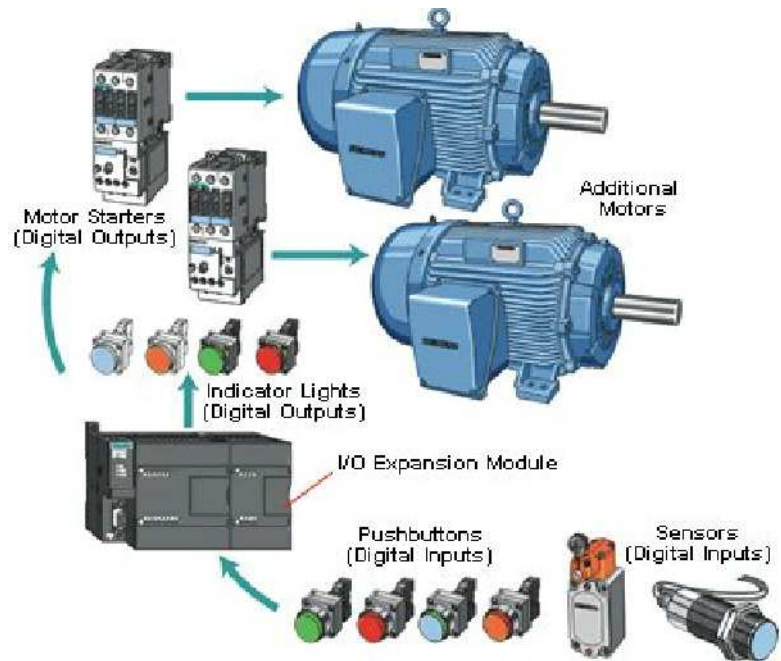
TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

Hệ điều khiển sử dụng PLC



TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programable Logic Controler) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

So sánh giữa hệ điều khiển cổ điển và PLC

Đặc điểm so sánh	Bảng điều khiển cổ điển	PLC
Phần tử điều khiển (Phần cứng)	Mục đích đặc biệt	Mục đích chung
Phạm vi điều khiển	Nhỏ và trung bình	Trung bình và lớn
Thay đổi thêm bớt	Khó	Dễ
Thời gian lắp đặt	Vài tuần, vài tháng	Vài ngày
Bảo trì, bảo dưỡng	Khó	Dễ
Độ tin cậy	Phụ thuộc nhà thiết kế và chế tạo	Cao
Hiệu quả kinh tế	Ưu điểm cho sản xuất nhỏ	Ưu điểm cho sản xuất nhỏ, trung bình và lớn

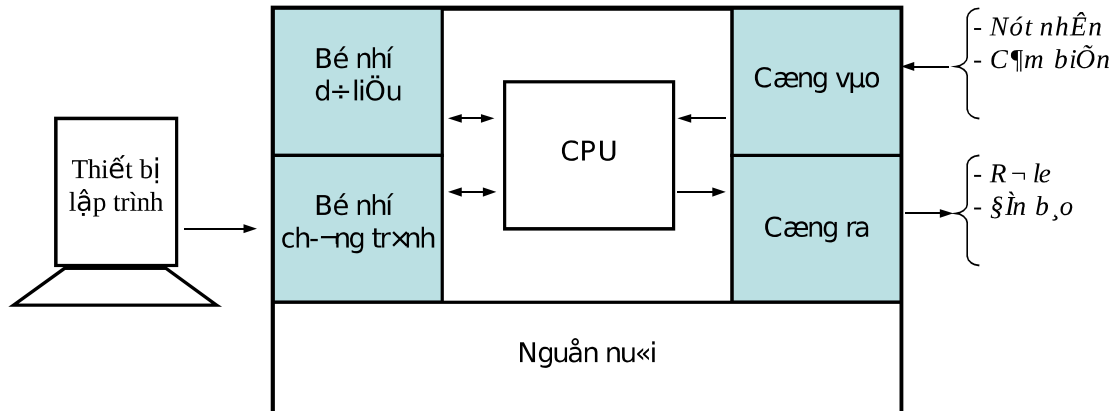
TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:



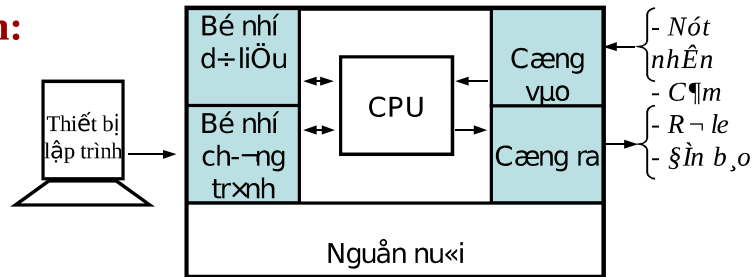
TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programmable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:



* **CPU** (Central Processing Unit) Là bé xử lý trung tâm của nhiệm vụ điều khiển và quản lý máy hoạt động bên trong PLC.

* **Bộ nhớ:**

Tất cả các loại PLC đều sở hữu 3 loại bộ nhớ sau:

Bộ nhớ ROM (Read Only Memory)

Bộ nhớ RAM (Random Access Memory)

Bộ nhớ EEPROM (Electrical Erasable Programmable ROM)

* **Thiết bị lập trình** có thể là máy tính cá nhân PC, máy lập trình chuyên dụng PG, hay máy lập trình hiển thị cầm tay TD.

* **Nguồn nuôi.** Là khối cung cấp nguồn để nuôi toàn bộ hoạt động của PLC. Nguồn nuôi này có thể được cung cấp bởi nguồn DC, AC tùy thuộc vào từng loại PLC

TỔNG QUAN VỀ PLC

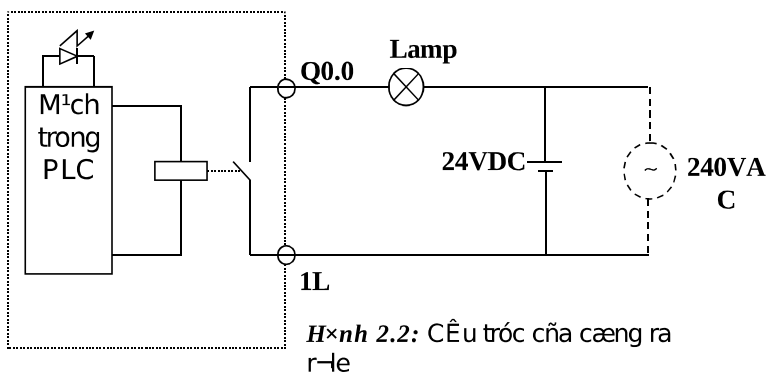
1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:

* Cấu trúc cổng vào/ra



Đặc điểm.

- Cả thó ãng c³/4t ã-íc c¶ dßng ãiÖn mét chiÒu lÉn xoay chiÒu, khi nÒi víi c, c thiÖt b¶ ngoµi kh«ng cÇn ph©n biÖt cùc tÝnh.
- §, p øng chÈm, kh«ng chÞu ã-íc tÇn sè ãng c³/4t cao.
- §ãng c³/4t ã-íc dßng t¶i kho¶ng 2A ÷ 5A tuú thuéc tÕng h·ng chÕ t³o.
- Tuæi thã thÈp (tiÖp ãiÖm r-ê chØ cho phÐp ãng c³/4t vµi chØc ngh½n lÇn).

TỔNG QUAN VỀ PLC

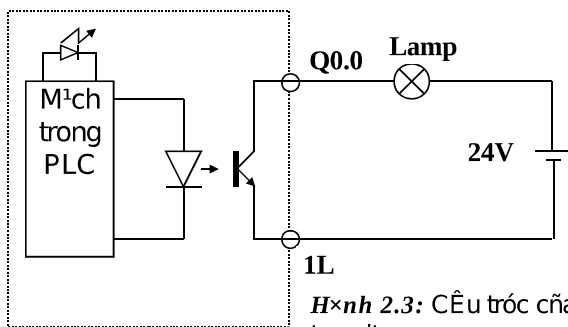
1. Khái niệm về PLC

PLC (Programmable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:

* Cấu trúc cổng vào/ra



Hình 2.3: Cấu trúc của bộ ra transistor

Đặc điểm.

- Cho tải công suất nhỏ (tối đa khoảng 50mA).
- Tốc độ phản ứng rất nhanh, chuyển mạch tần số cao.
- Tuổi thọ cao.

TỔNG QUAN VỀ PLC

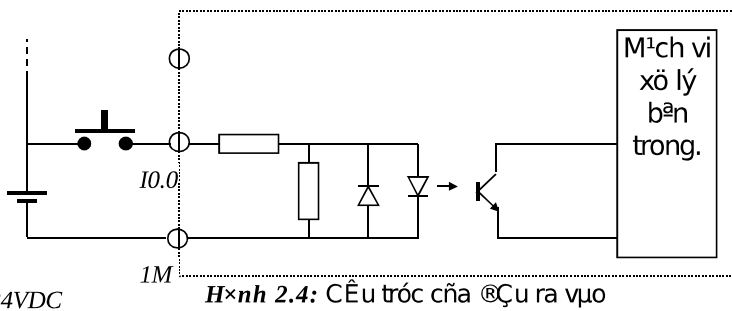
1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:

* Cấu trúc cổng vào/ra



Hình 2.4: Cấu trúc của cổng ra

Đặc điểm.

- Chở tải (tải) lớn (tải động điện) một chiều.
- Tốc độ rất nhanh, chịu tải tốt sẽ tải cao.
- Tải động điện tải khoảng 50mA.
- Tuổi thọ cao.

1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:

4. Đặc điểm PLC:

- Cấu trúc đơn giản với thiết kế có cấu trúc nhỏ gọn
- Chịu đựng được môi trường công nghiệp (rung, độ ẩm, tiếng ồn, nhiệt độ ... cao)
- Lập trình đơn giản, thực hiện được nhiều chức năng điều khiển (logic, điều khiển tương tự PID, truyền thông...)
- Ngôn ngữ lập trình động

TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

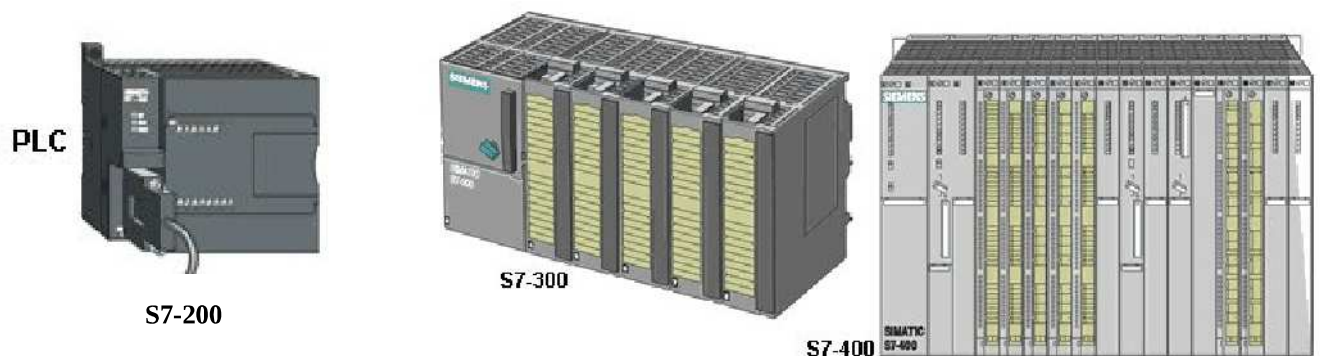
PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:

4. Đặc điểm PLC:

Các loại PLC Siemens



TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:

4. Đặc điểm PLC:

Nguồn nôi PLC

Model Description	Power Supply	Input Types	Output Types
221 DC/DC/DC	20.4-28.8 VDC	6 DC	4 DC
221 AC/DC/Relay	85-264 VAC, 47-63 Hz	6 DC	4 Relay
222 DC/DC/DC	20.4-28.8 VDC	8 DC	6 DC
222 AC/DC/Relay	85-264 VAC, 47-63 Hz	8 DC	6 Relay
224 DC/DC/DC	20.4-28.8 VDC	14 DC	10 DC
224 AC/DC/Relay	85-264 VAC, 47-63 Hz	14 DC	10 Relay
224XP DC/DC/DC	20.4-28.8 VDC	14 DC, 2 Analog	10 DC, 1 Analog
224XP AC/DC/Relay	85-264 VAC, 47-63 Hz	14 DC, 2 Analog	10 Relay, 1 Analog
226 DC/DC/DC	20.4-28.8 VDC	24 DC	16 DC
226 AC/DC/Relay	85-264 VAC, 47-63 Hz	24 DC	16 Relay

TỔNG QUAN VỀ PLC

1. Khái niệm về PLC

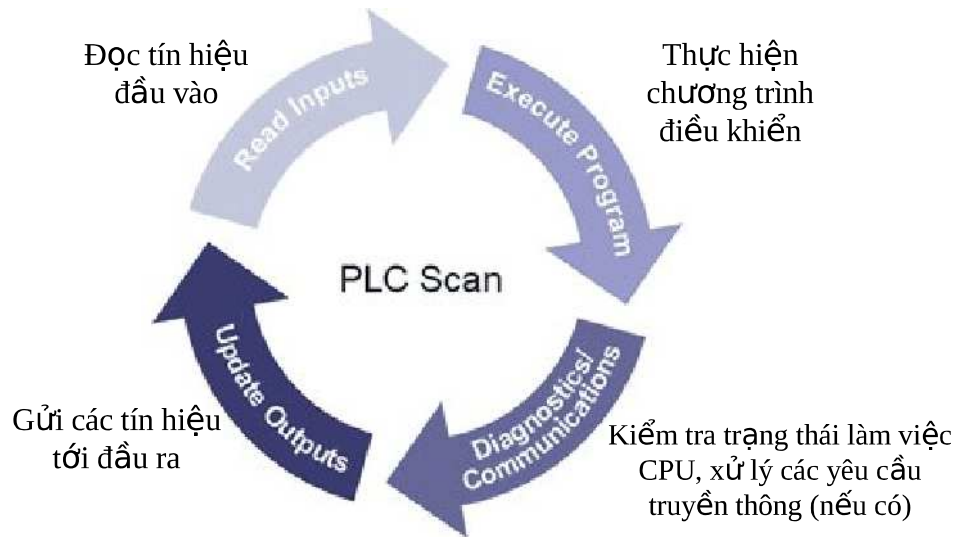
PLC (Programable Logic Controller) là một thiết bị điều khiển sử dụng một **bộ nhớ có thể lập trình**, bộ nhớ này sẽ **lưu giữ các cấu trúc lệnh** (logic, thời gian, bộ đếm, các hàm toán học...) để **thực hiện các chức năng điều khiển**

2. Cơ sở phát triển:

3. Cấu trúc PLC:

4. Đặc điểm PLC:

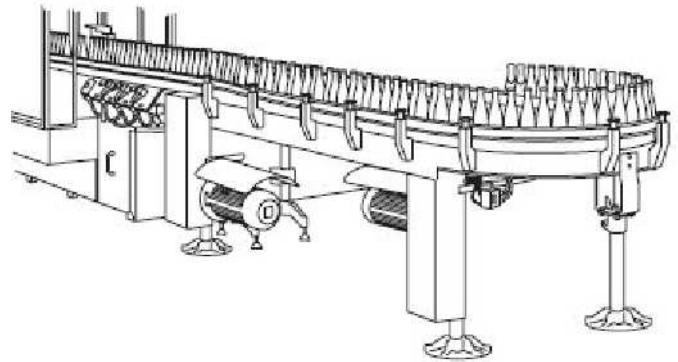
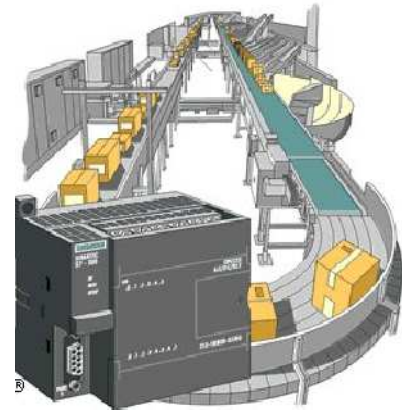
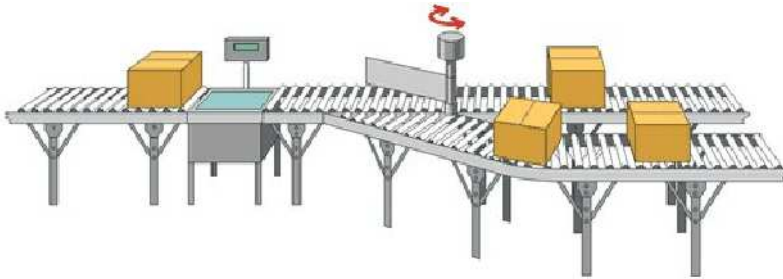
5. Vòng quét (Cycle scan):



TỔNG QUAN VỀ PLC

6. Ứng dụng PLC.

Theo tài liệu phân tích của Đức thì PLC được sử dụng để điều khiển trong nền công nghiệp Việt Nam là 70%.



1. Hệ đếm:

PLC giống như một máy tính, nó lưu trữ và xử lý tín hiệu ở hai dạng ON và OFF (1 hoặc 0), gọi là các bit nhị phân. Các tín hiệu này có thể được sử dụng là các tín hiệu độc lập, có thể được dùng để biểu diễn một giá trị số

a. Hệ thập phân:

Là hệ đếm cơ số 10:

Các con số: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Trọng số: 1, 10, 100, 1000. . .

b. Hệ nhị phân:

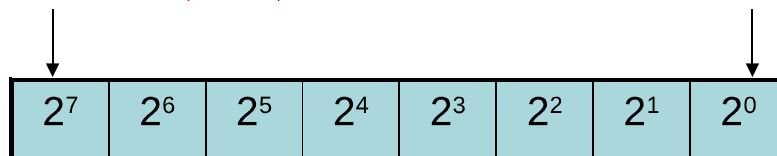
Là hệ đếm cơ số 2:

Các con số: 0, 1

Trọng số: 1, 2, 4, 8. . .

Bit có trọng số cao nhất (MSB)

Bit có trọng số thấp nhất (LSB)



1. Hệ đếm:

a. Hệ thập phân

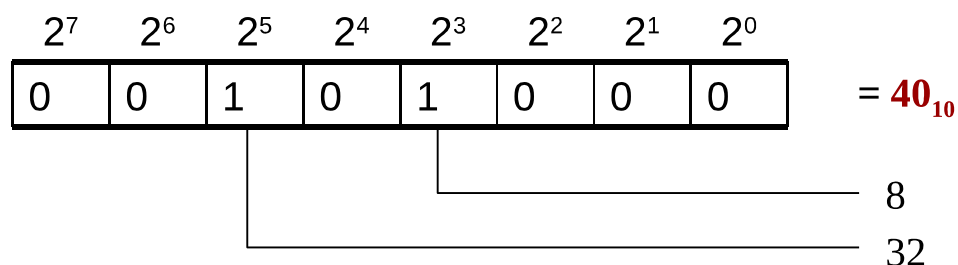
b. Hệ nhị phân

Chuyển đổi từ hệ nhị phân sang hệ thập phân

Để chuyển đổi từ hệ nhị phân sang hệ thập phân ta thực hiện theo ba bước sau:

- Tìm tất cả bit có giá trị 1 từ phải qua trái (từ LSB đến MSB)
- Viết các giá trị thập phân tương ứng cho các bit mang giá trị 1
- Công tất cả các giá trị thập phân này

Ví dụ:



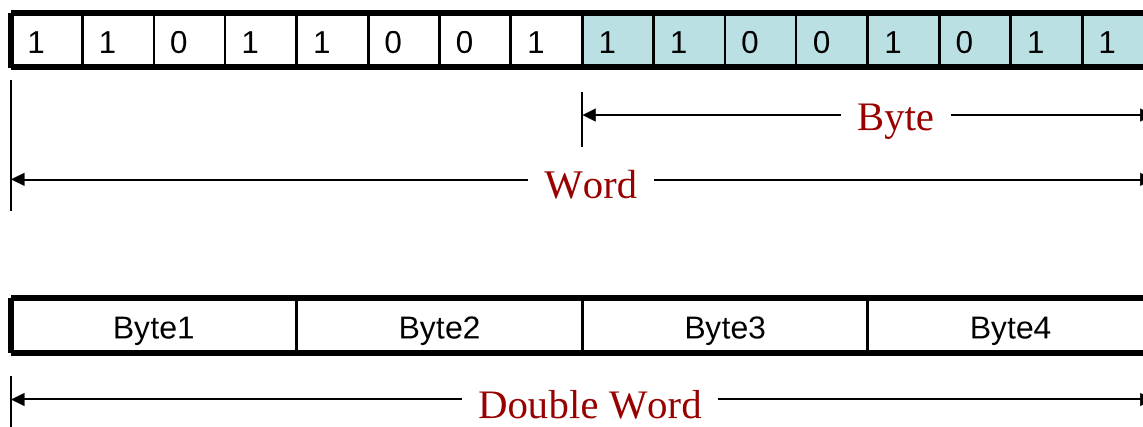
1. Hệ đếm:

a. Hệ thập phân

b. Hệ nhị phân

Bit, Byte, Word, Double Word

Mỗi con số trong hệ nhị phân biểu diễn bằng 1 bit, 8 bit được gọi là 1 byte, 2 byte là một word và 4 byte là một double word



1. Hệ đếm:

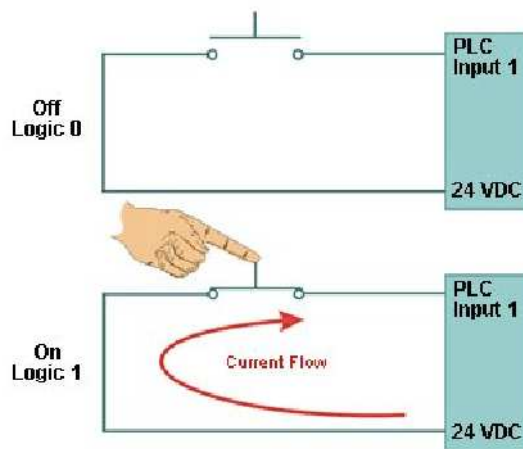
a. Hệ thập phân

b. Hệ nhị phân

Logic 0 và logic 1

Các thiết bị lập trình điều khiển chỉ có thể xử lý được tín hiệu ở dạng “ĐÓNG” hoặc “MỞ”, “ON” hoặc “OFF” (“tồn tại” hoặc “không tồn tại”). Hệ nhị phân là hệ bao gồm hai con số: 0 và 1. “0” chỉ ra rằng không có tín hiệu, tương đương với trạng thái “MỞ” của tiếp điểm, “1” nghĩa là có tín hiệu, tương đương với trạng thái “ĐÓNG” của tiếp điểm.

Ví dụ:



1. Hệ đếm:

a. Hệ thập phân

b. Hệ nhị phân

c. Số BCD(Binary-Coded Decimal):

Là một số thập phân được biểu diễn bằng 4 bit nhị phân, như vậy số nhị phân sẽ được chia thành nhóm 4 bit, mỗi bit này biểu diễn một con số thập phân tương ứng

Số thập phân	Số BCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

1. Hệ đếm:

a. Hệ thập phân

b. Hệ nhị phân

c. Số BCD(Binary-Coded Decimal):

d. Hệ Hexa:

Là hệ đếm cơ số 16:

Các con số: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

Trọng số: 1, 16, 256, 4096. . .

(Trong đó A = 10, B = 11, C = 12, D = 13, E = 14, F = 15)

Chuyển đổi giữa hệ thập phân và hệ Hexa

Ví dụ: $2B_{16}$

$$= 2 \times 16^1 + 12 \times 16^0 = 43_{10}$$

CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

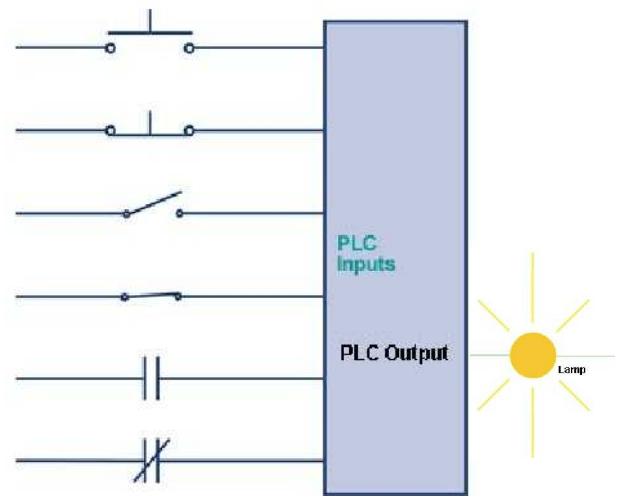
1. Các khái niệm cơ bản:

a. Đầu vào số: Còn được gọi là đầu vào tín hiệu rời rạc, là các đầu vào của PLC chỉ nhận các tín hiệu ở hai dạng “CÓ” hoặc “KHÔNG” có, các tín hiệu này có thể lấy từ nút nhấn, công tắc, cảm biến hành trình, cảm biến tiệm cận...

b. Đầu ra số: Còn được gọi là đầu ra tín hiệu rời rạc, là các đầu ra của PLC chỉ nhận các tín hiệu ở hai dạng “ĐÓNG” hoặc “MỞ”, các đầu ra này thường được nối với các cuộn dây Relay, Contactor, Đèn, Vale ... Các phần tử này sẽ được điều khiển bật tắt bởi PLC

CÓ: Logic 1
Tín hiệu ở mức cao

KHÔNG: Logic 0
Tín hiệu ở mức thấp

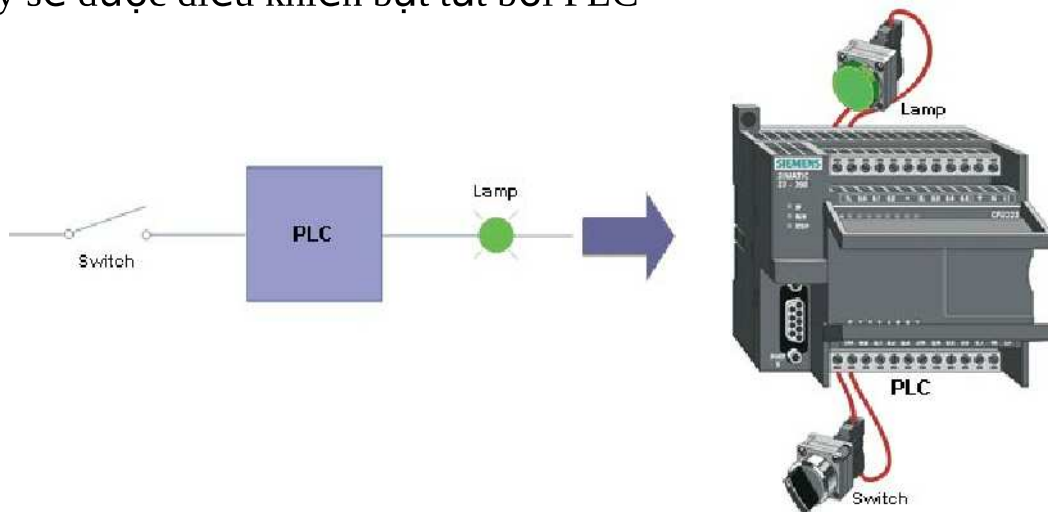


CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

a. Đầu vào số: Còn được gọi là đầu vào tín hiệu rời rạc, là các đầu vào của PLC chỉ nhận các tín hiệu ở hai dạng “CÓ” hoặc “KHÔNG” có, các tín hiệu này có thể lấy từ nút nhấn, công tắc, cảm biến hành trình, cảm biến tiệm cận...

b. Đầu ra số: Còn được gọi là đầu ra tín hiệu rời rạc, là các đầu ra của PLC chỉ nhận các tín hiệu ở hai dạng “ĐÓNG” hoặc “MỞ”, các đầu ra này thường được nối với các cuộn dây Relay, Contactor, Đèn, Vale ... Các phần tử này sẽ được điều khiển bật tắt bởi PLC

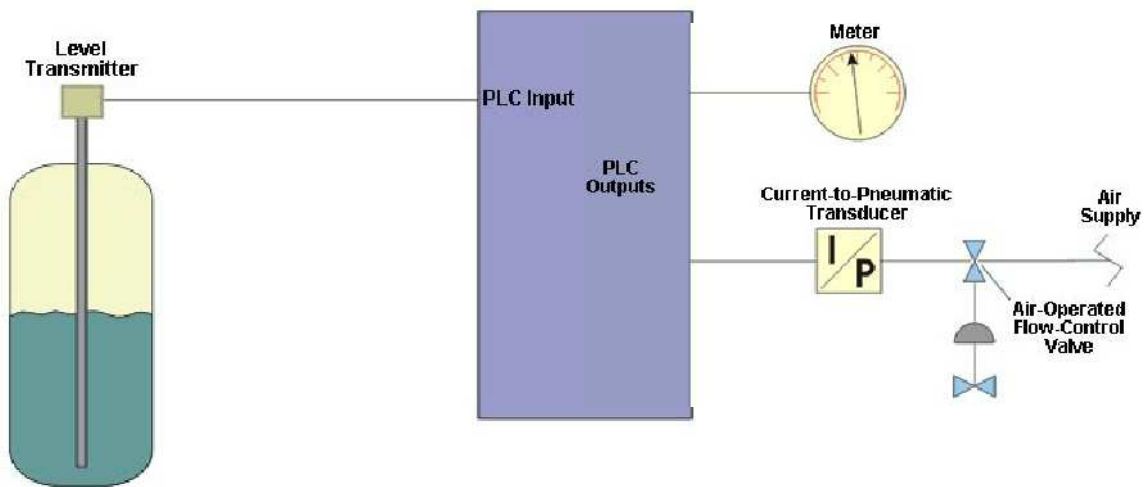


CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

a. **Đầu vào tương tự:** Là các đầu vào PLC nhận các tín hiệu biến thiên liên tục. thể hiện ở dòng điện và điện áp

b. **Đầu ra tương tự:** Là các đầu ra của PLC có tín hiệu biến thiên liên tục, có thể là điện áp hoặc dòng điện.

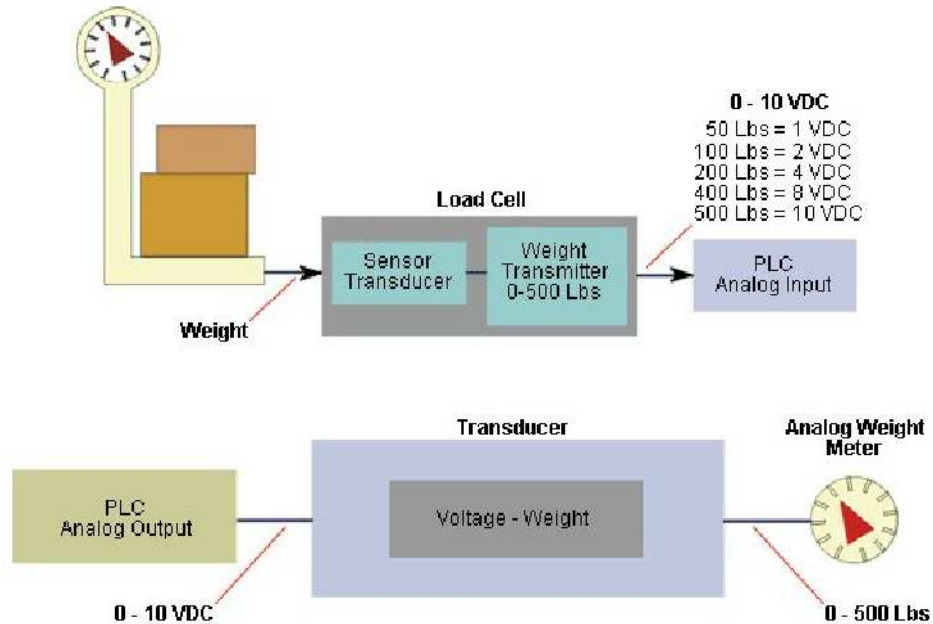


CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

a. **Đầu vào tương tự:** Là các đầu vào PLC nhận các tín hiệu biến thiên liên tục. thể hiện ở dòng điện và điện áp

b. **Đầu ra tương tự:** Là các đầu ra của PLC có tín hiệu biến thiên liên tục, có thể là điện áp hoặc dòng điện.



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

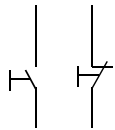
Là các phần tử chuyển đổi các tín hiệu vật lý thành tín hiệu điện đưa tới các đầu vào của PLC. Có thể là công tắc, nút nhấn, cảm biến,...

a. Nút nhấn.

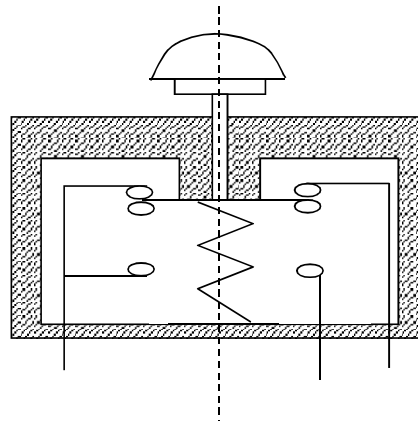


Pushbutton

Ký hiệu



Cấu tạo



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

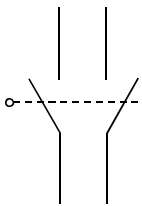
Là các phần tử chuyển đổi các tín hiệu vật lý thành tín hiệu điện đưa tới các đầu vào của PLC. Có thể là công tắc, nút nhấn, cảm biến,...

a. Nút nhấn.

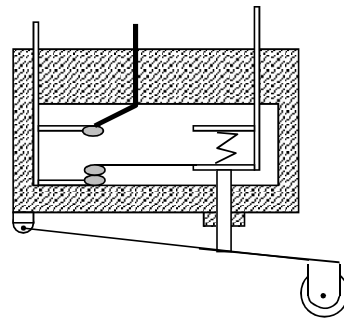
b. Công tắc hành trình



Ký hiệu



Cấu tạo



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

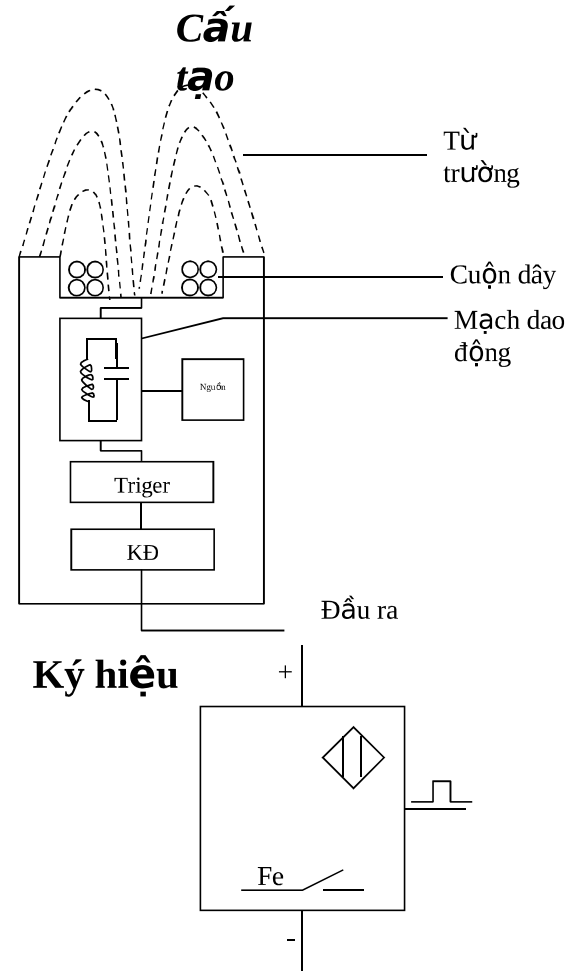
2. Các phần tử đầu vào:

c. Cảm biến cảm ứng từ



Nguyên lý hoạt động:

Khi có vật bằng kim loại đặt trong vùng đường sức của từ trường. Trong kim loại sẽ hình thành dòng điện xoáy. Như vậy năng lượng của bộ dao động sẽ giảm, dòng điện xoáy sẽ tăng khi vật cản nằm càng gần cuộn cảm ứng. Qua đó biên độ dao động của bộ dao động sẽ giảm qua mạch trigger sẽ thu được tín hiệu số, tín hiệu này sẽ được khuếch đại thành tín hiệu ra



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

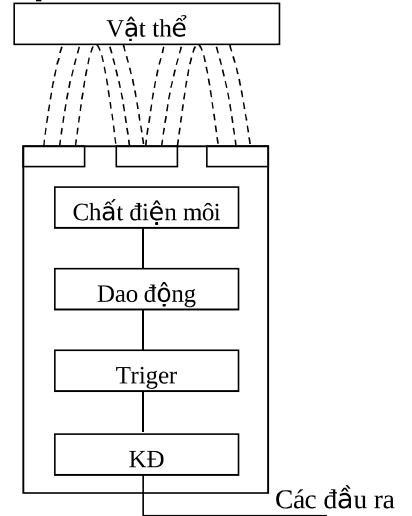
d. Cảm biến điện dung



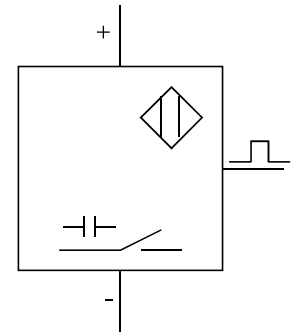
Nguyên lý hoạt động:

Nguyên tắc hoạt động dựa trên sự thay đổi điện trường. Khi có vật bằng kim loại hoặc phi kim nằm trong vùng đường sức của điện trường, điện dung của tụ điện sẽ thay đổi. Dẫn tới biên độ của tín hiệu trên mạch dao động sẽ thay đổi. Qua mạch triger, mạch khuếch đại ta sẽ thu được tín hiệu số ở đầu ra.

Cấu tạo



Ký hiệu

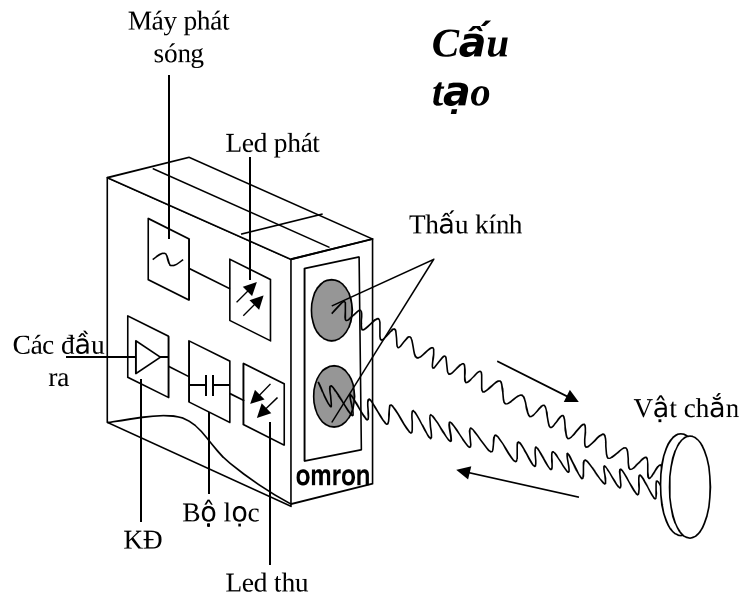


CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

e. Cảm biến quang

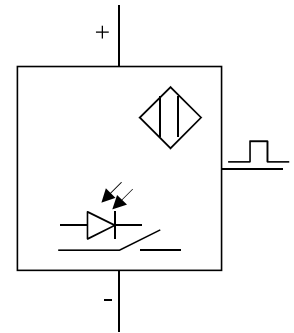


Cấu tạo

Nguyên lý hoạt động:

Gồm hai bộ phận chính là bộ phận phát và bộ phận thu. Bộ phận phát phát đi tín hiệu là ánh sáng, gặp vật chắn sẽ phản hồi trở lại bộ phận nhận, tín hiệu này sẽ được xử lý và khuếch đại thành tín hiệu ở đầu ra

Ký hiệu

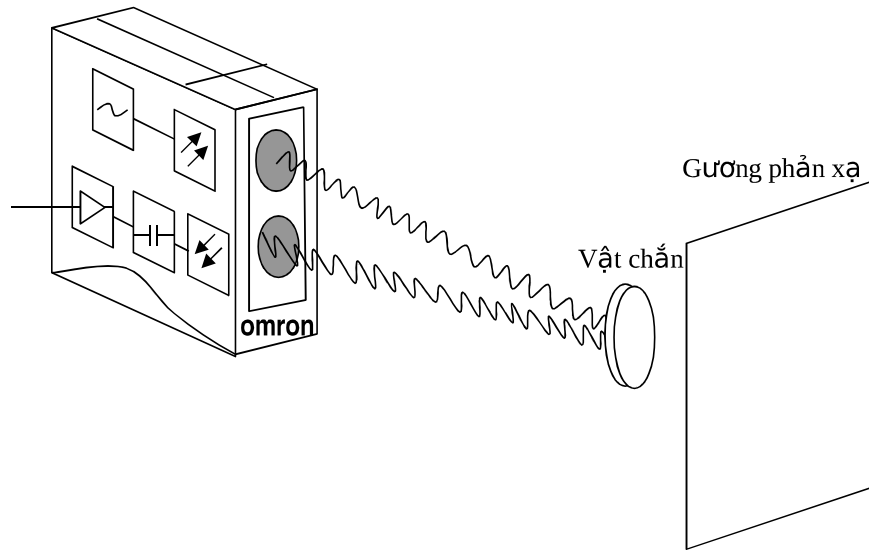


CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

e. *Cảm biến quang có gương phản xạ*

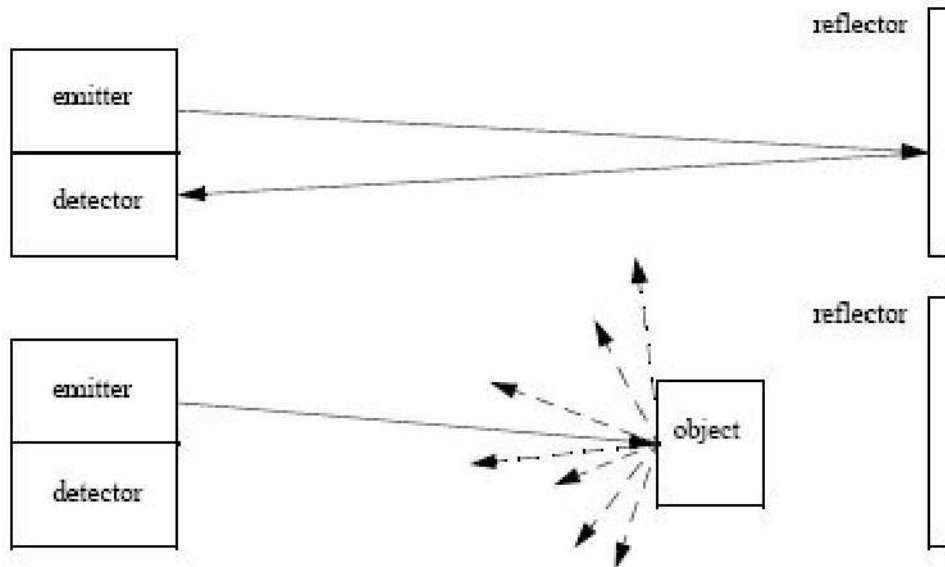


CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

e. *Cảm biến quang có gương phản xạ*



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

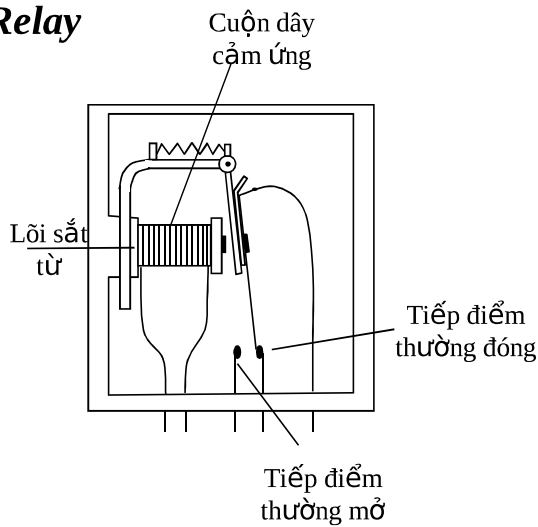
1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

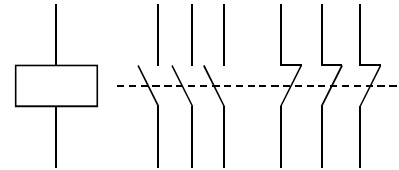
3. Các phần tử đầu ra:

Là các phần tử chuyển đổi các tín hiệu điện thành các tín hiệu vật lý, các tín hiệu điện này lấy từ đầu ra của PLC

a. Relay



Ký hiệu



Nguyên lý hoạt động:

Relay là phần tử xử lý tín hiệu. Khi có dòng điện chạy trong cuộn dây cảm ứng sẽ xuất hiện lực từ trường hút lõi sắt. Trên lõi sắt có gắn các cặp tiếp điểm để đóng mở mạch động lực và mạch điều khiển

CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

3. Các phần tử đầu ra:

a. Vale

Ký hiệu

- Chuyển đổi vị trí lòng vale

1	0	2
---	---	---

1	0
---	---

Chuyển đổi vị trí lòng vale được biểu diễn bằng các ô vuông liền nhau ký hiệu bằng các chữ cái o, a, b .. Hoặc các số 0, 1, 2 ..

Vị trí “Không” là vị trí chưa có tín hiệu điều khiển bên ngoài vào (ký hiệu là số 0, hoặc chữ b). Đối với vale 3 vị trí, vị trí “không” nằm ở giữa. Vale 2 vị trí “không” nằm ở ô vuông bên phải.

CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

3. Các phần tử đầu ra:

a. Vale

- Ký hiệu cửa nối vale:

Cửa	Ký hiệu theo ISO 1219
Cửa nối nguồn	P
Cửa nối tải	A, B, C
Cửa xả khí	R, S, T
Cửa điều khiển	X, Y.

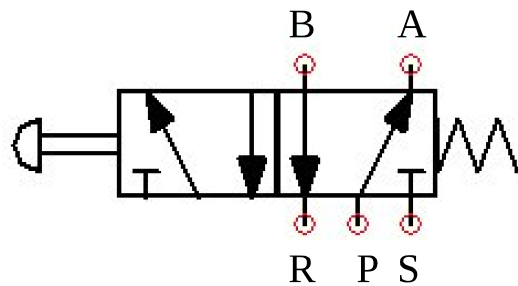
CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:
2. Các phần tử đầu vào:
3. Các phần tử đầu ra:

a. Vale

- **Hướng chuyển động của dòng khí.**

Bên trong ô vuông của mỗi vị trí là các đường thẳng có hình mũi tên biểu diễn hướng chuyển động của dòng khí qua vale. Trường hợp dòng bị chặn được biểu diễn bằng dấu gạch ngang



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

2. Các phần tử đầu vào:

3. Các phần tử đầu ra:

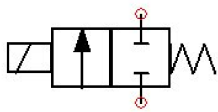
a. Vale

- Cách gọi tên vale:

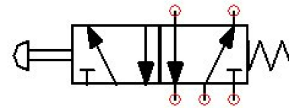
Vale + Số cửa/ Số vị trí + Tín hiệu điều khiển

Ví dụ:

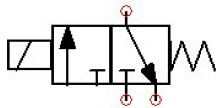
Vale 2/2 điều khiển trực tiếp bằng điện



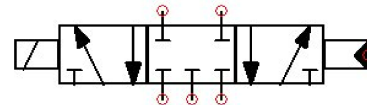
Vale 5/2 điều khiển trực tiếp bằng nút nhấn



Vale 3/2 điều khiển trực tiếp bằng điện



Vale 5/3 điều khiển gián tiếp qua vale phụ trợ



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:

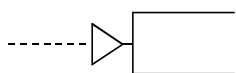
2. Các phần tử đầu vào:

3. Các phần tử đầu ra:

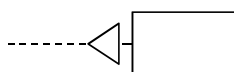
a. Vale

- Tín hiệu tác động:

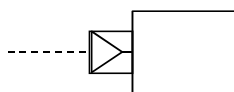
Trực tiếp bằng dòng khí nén vào



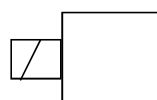
Trực tiếp bằng dòng khí nén ra



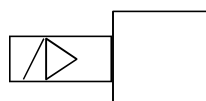
Gián tiếp bằng dòng khí nén vào qua vale phụ trợ



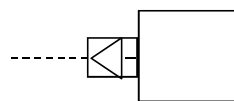
Trực tiếp bằng điện



Gián tiếp bằng điện có khí nén phụ trợ

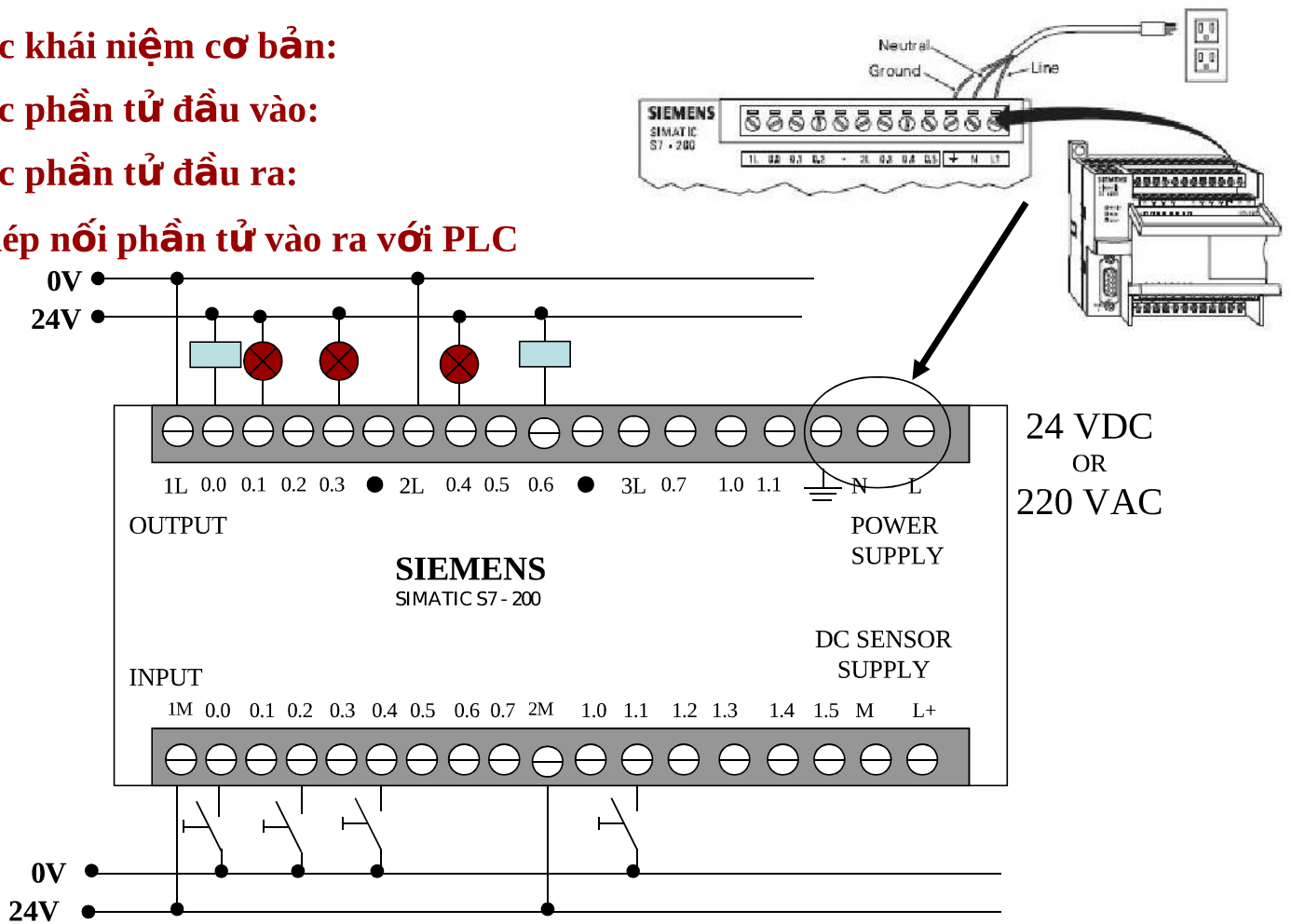


Gián tiếp bằng dòng khí nén vào qua vale phụ trợ



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:
2. Các phần tử đầu vào:
3. Các phần tử đầu ra:
4. Ghép nối phần tử vào ra với PLC



CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

1. Các khái niệm cơ bản:
2. Các phần tử đầu vào:
3. Các phần tử đầu ra:
4. Ghép nối phần tử vào ra với PLC
5. Các modul trong hệ thống điều khiển PLC

Các modul mở rộng được chia thành 5 loại chính:

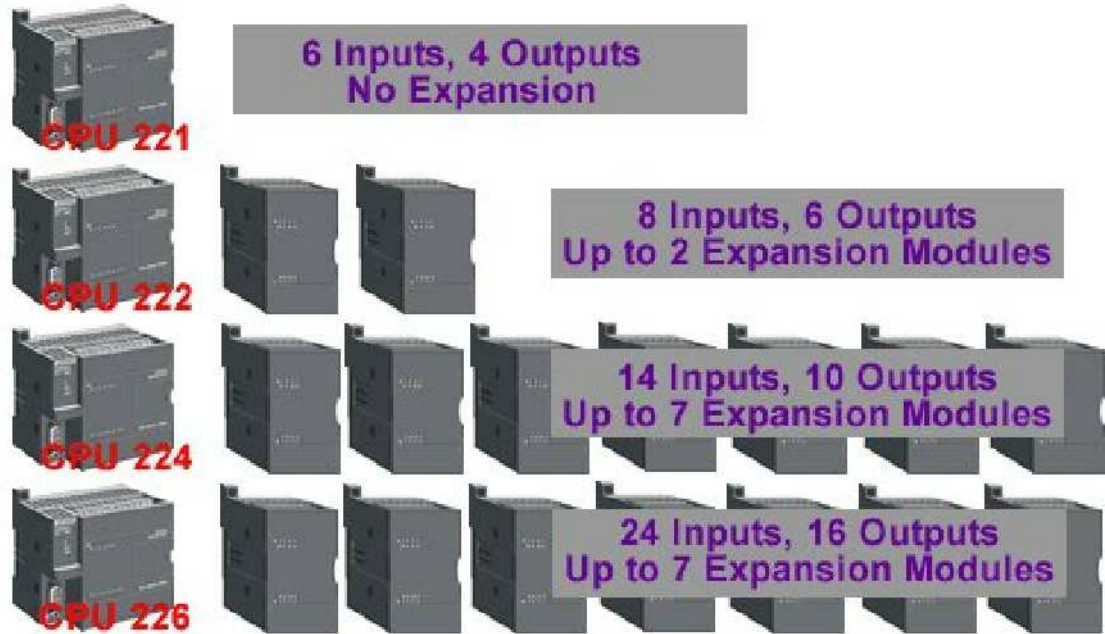
- a. PS (*power supply*): Modul nguồn nôi. Có 3 loại 2A, 5A và 10 A
- b. SM: Modul mở rộng tín hiệu vào ra.

Tùy vào từng loại CPU mà PLC có thể mở rộng với ít hay nhiều modul mở rộng tín hiệu vào ra. Có các loại modul vào ra như:

- DI (*Digital Input*): Modul mở rộng cổng vào số
- DO (*Digital Output*): Modul mở rộng cổng ra số
- DI/DO: Modul mở rộng cổng vào/ra số
- AI/AO: Modul mở rộng cổng vào ra tương tự

5. Các modul trong hệ thống điều khiển PLC

- a. PS (*power supply*): Modul nguồn nôi
- b. SM: Modul mở rộng tín hiệu vào ra.



5. Các modul trong hệ thống điều khiển PLC

a. PS (*power supply*): Modul nguồn nuôi

b. SM: Modul mở rộng tín hiệu vào ra.

c. IM(*Interface modul*): Modul ghép nối

Đây là loại modul chuyên dụng có nhiệm vụ nối từng nóm các modul mở rộng lại với nhau thành một khối và được quản lý chung bởi một CPU.

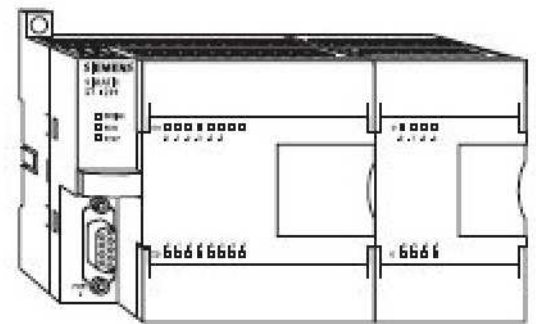
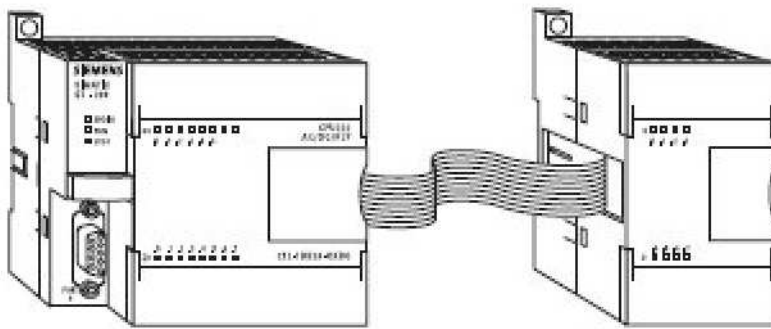
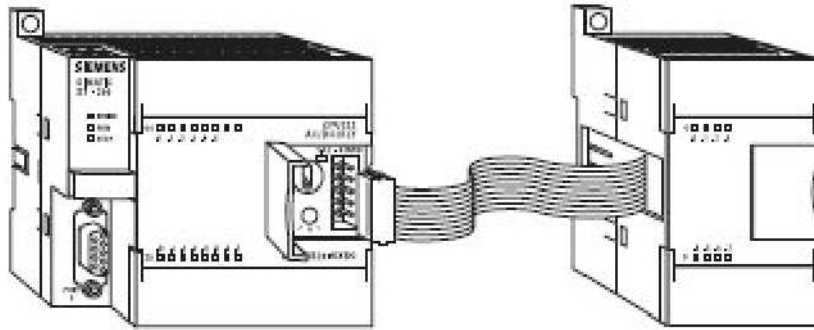
d. FM(*Function modul*): Modul có chức năng điều khiển riêng

Ví dụ modul chức năng điều khiển động cơ bước, modul điều khiển Servo, modul PID, modul điều khiển vòng kín.

d. CP(*Commuication modul*): Modul phục vụ truyền thông trong mạng giữa các PLC với nhau hoặc giữa PLC với máy tính

CÁC PHẦN TỬ VÀO/RA TÍN HIỆU

Cách ghép nối modul mở rộng



1. Các phương pháp truy nhập bộ nhớ.

Truy nhập theo bit

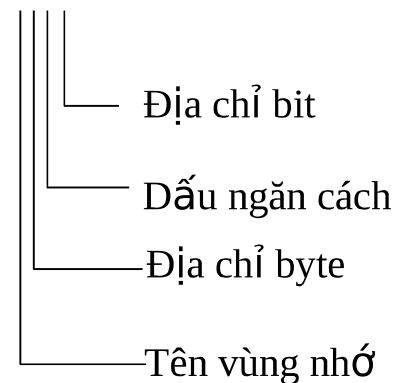
Quy cách: Tên vùng nhớ, địa chỉ byte, địa chỉ bit (ngăn cách địa chỉ byte và địa chỉ bit là dấu “.”)

Ví dụ:

- Kiểu bit:

	MSB				LSB			
	7	6	5	4	3	2	1	0
IB0								
IB1								
IB2								
IB3								
IB4								
IB5								
IB6								
IB7								

I3.4



1. Các phương pháp truy nhập bộ nhớ.

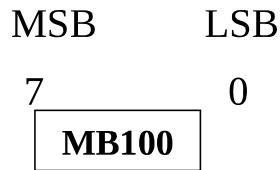
Truy nhập theo byte

Bao gồm các kiểu Byte (1 byte), Word (2 byte), Double Word (4 byte)

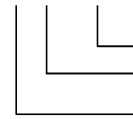
Quy cách: Tên vùng nhớ, kích thước, địa chỉ byte đầu tiên.

Ví dụ:

- Kiểu byte:

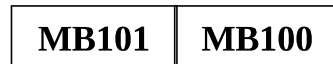


MB100

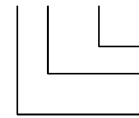


Địa chỉ byte 100
Kiểu truy nhập (byte)
Tên vùng nhớ (M)

- Kiểu word:

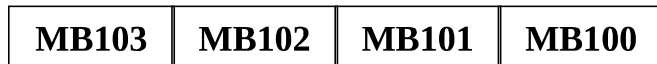


MW100

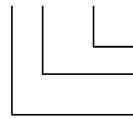


Địa chỉ byte 100
Kiểu truy nhập (word)
Tên vùng nhớ (M)

- Kiểu double word:



MD100



Địa chỉ byte 100
Kiểu truy nhập (Dword)
Tên vùng nhớ (M)

1. Các phương pháp truy nhập bộ nhớ.

2. Cấu trúc bộ nhớ.

Vùng nhớ đầu vào I

Tại thời điểm đầu tiên mỗi vòng quét PLC lấy tín hiệu từ các đầu vào và ghi các giá trị tương ứng vào vùng nhớ đầu vào

Truy nhập:

Kiểu Bit:	I[địa chỉ byte].[địa chỉ bit]	I0.1
Kiểu Byte, word, Dword	I[kích thước].[địa chỉ byte đầu tiên]	IB4, IW2, ID1

Vùng nhớ đầu ra Q

Trong quá trình thực hiện các công việc trong một vòng quét (bao gồm cả chương trình điều khiển) PLC sẽ ghi các giá trị tương ứng vào vùng nhớ này. Cuối vòng quét PLC sẽ gửi các giá trị này đến đầu ra tương ứng

Truy nhập:

Kiểu Bit:	Q[địa chỉ byte].[địa chỉ bit]	Q0.0
Kiểu Byte, word, Dword	Q[kích thước].[địa chỉ byte đầu tiên]	QB4, QW2, QD1

1. Các phương pháp truy nhập bộ nhớ.

2. Cấu trúc bộ nhớ.

Vùng nhớ M

Các ô thuộc vùng nhớ M dùng để lưu trữ trạng thái của quá trình hoạt động hoặc các thông tin điều khiển khác.

Truy nhập:

Kiểu Bit:	V[địa chỉ byte].[địa chỉ bit]	V0.1
Kiểu Byte, word, Dword	V[kích thước].[địa chỉ byte đầu tiên]	VB4, VW8, VD1

Vùng nhớ thời gian T

Mỗi bộ thời gian có hai giá trị được lưu trữ trong vùng nhớ T: Giá trị đếm thời gian hiện tại (16 bit) và giá trị timer (1bit)

Truy nhập:

T[số thứ tự bộ timer]	T37
-----------------------	-----

LẬP TRÌNH CHO PLC

1. Các thành phần cơ bản

Để lập trình cho PLC cần các thành phần cơ bản sau: PLC, thiết bị lập trình (PC, PG ...), cáp nối, phần mềm

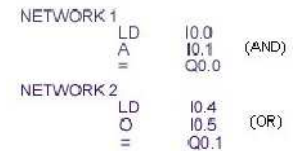
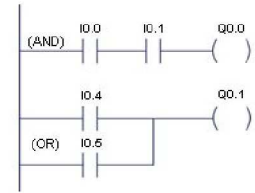
2. Các dạng ngôn ngữ lập trình cho PLC

Có 3 loại ngôn ngữ thường được sử dụng:

LAD: Còn gọi là ngôn ngữ giản đồ thang, có các thành phần giống như các thành phần trong kỹ thuật điện (tiếp điểm, cuộn dây, timer, relay..)

STL: Là một dạng thể hiện khác của các câu lệnh lập trình, một cấu trúc lệnh trong LAD có thể là một tập hợp lệnh trong STL. Chương trình ở dạng này sử dụng các câu lệnh ở dạng chữ viết giống như PASCAL, C..

FBD: Đây là ngôn ngữ viết dưới dạng liên kết của các hàm logic kỹ thuật số, loại ngôn ngữ này thích hợp cho những người quen sử dụng và thiết kế mạch điều khiển số



LẬP TRÌNH CHO PLC

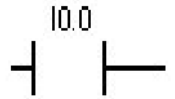
1. Các thành phần cơ bản

2. Các dạng ngôn ngữ lập trình cho PLC

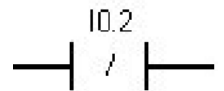
3. Một số khái niệm cơ bản

Tiếp điểm: Chương trình PLC sử dụng các bit giống như các tiếp điểm, có hai loại tiếp điểm là thường đóng (NC) và thường mở (NO):

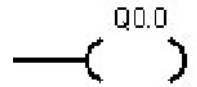
Tiếp điểm thường mở sẽ đóng lại khi bit địa chỉ của tiếp điểm này có giá trị bằng 1 và mở khi bit địa chỉ của tiếp điểm có giá trị bằng 0



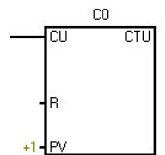
Tiếp điểm thường đóng sẽ mở khi bit địa chỉ của tiếp điểm này có giá trị bằng 1 và đóng lại khi bit địa chỉ của tiếp điểm có giá trị bằng 0



Cuộn dây: giống như cuộn dây relay, nó sẽ được kích hoạt khi tất cả tiếp điểm phía trước đóng, khi đó bit địa chỉ của phần tử này có giá trị bằng 1. Ta cũng có thể sử dụng các tiếp điểm của bit địa chỉ này tại các vị trí khác trong chương trình



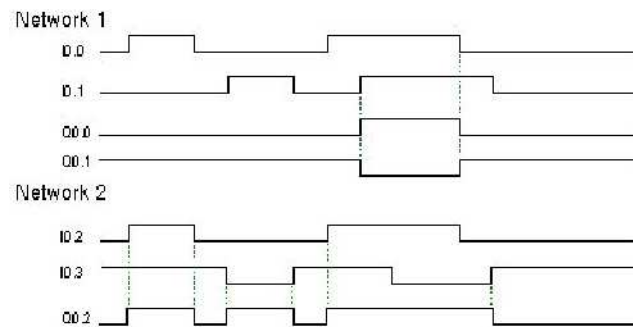
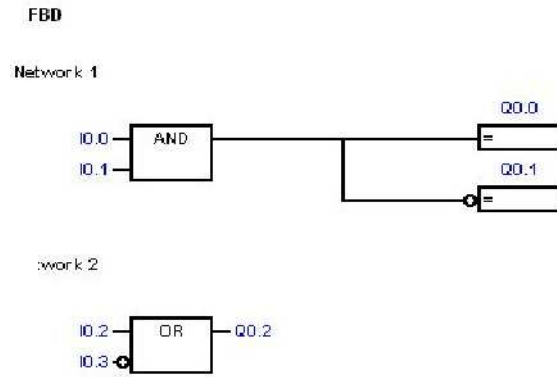
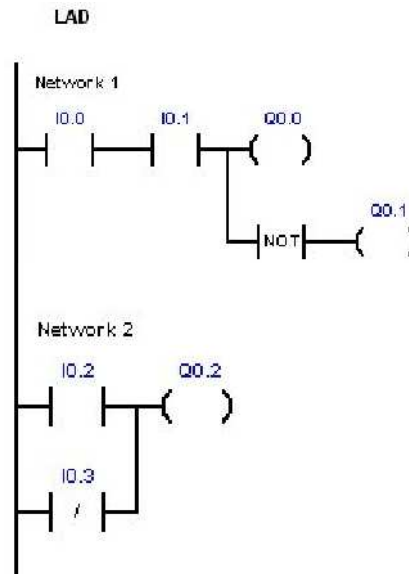
Khối: Khối trong chương trình PLC có nhiều chức năng khác nhau, một khối sẽ thực hiện chức năng này khi có tín hiệu ở đầu vào của khối, chức năng của khối có thể là bộ timer, counter..



4. Các hàm cơ bản

4.1. Hàm AND, OR, NOT

L A D	bit	
	bit	
F B D	AND	
	OR	
S T L	LD	bit
	A	bit
	O	bit
	LDN	bit
	AN	bit
	ON	bit



NETWORK 1

```
LD I0.0
A I0.1
= Q0.0
NOT I0.1
= Q0.1
```

NETWORK 2

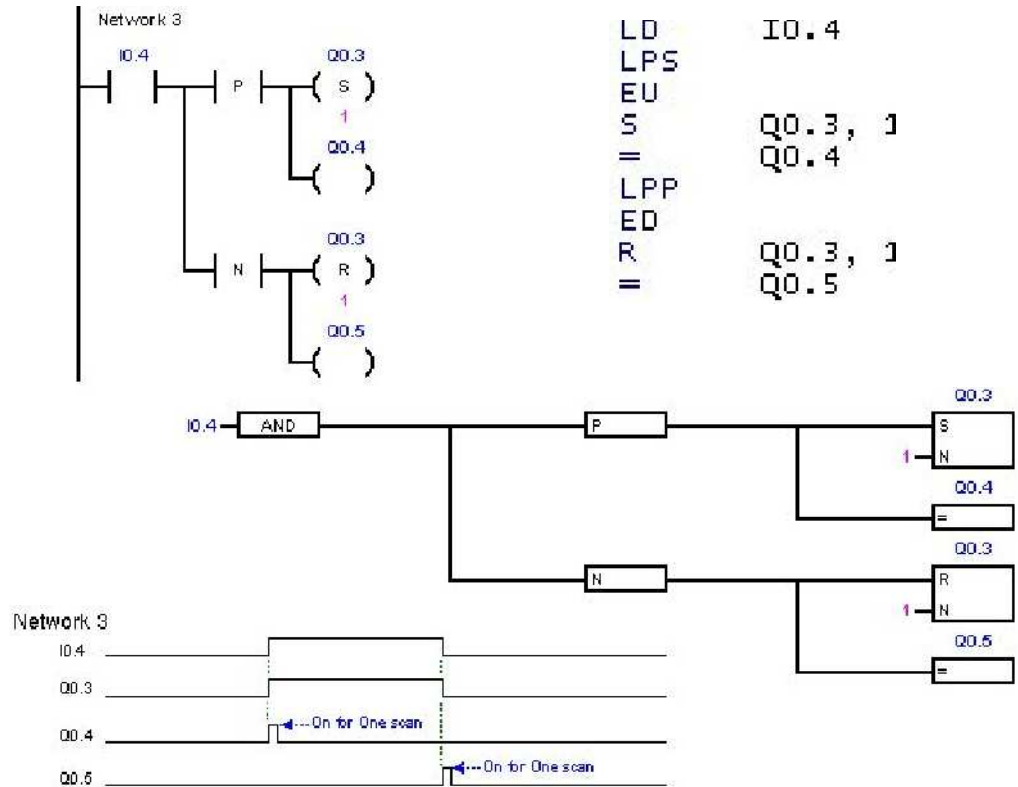
```
LD I0.2
OR I0.3
= Q0.2
```

4.2. Hàm POSITIVE và NEGATIVE

Cho một xung ở đầu ra có độ rộng một chu kỳ vòng quét tại thời điểm xuất hiện sườn lên (câu lệnh POSITIVE) và sườn xuống (câu lệnh NEGATIVE) của xung

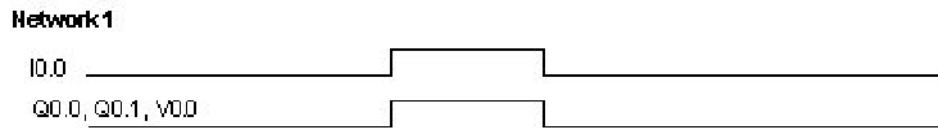
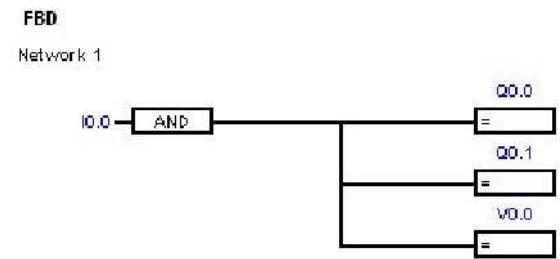
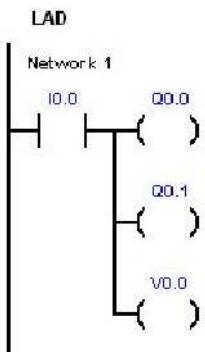
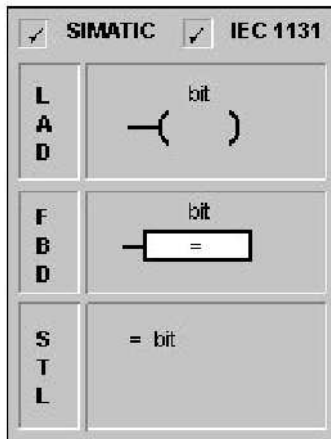
SIMATIC
 IEC 1131

L A D	
F B D	
S T L	<pre>EU ED</pre>



4.3. Lệnh đầu ra ASSIGNMENT

Gửi tín hiệu đầu ra khi có tín hiệu đầu vào

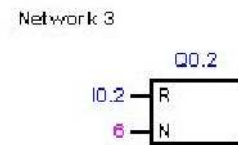
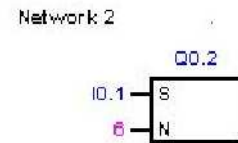
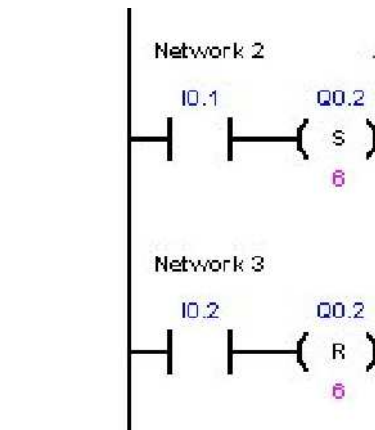
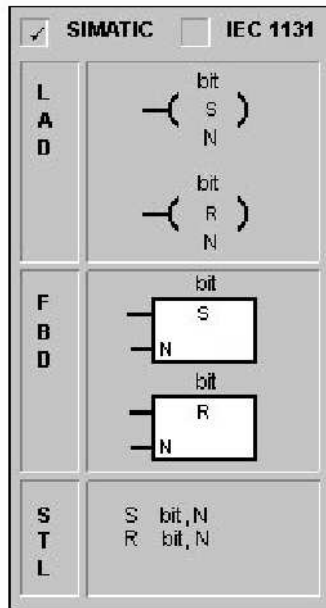


NETWORK 1

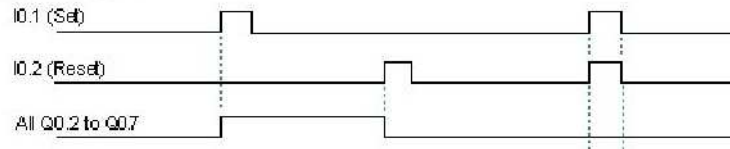
```
LD    I0.0
=     Q0.0
=     Q0.1
=     V0.0
```

4.4. Lệnh SET, RESET n bit

Ghi hoặc xoá giá trị 1 của n bit kể từ địa chỉ bit



Networks 2 and 3



NETWORK 2

```
LD I0.1
S Q0.2, 6
```

NETWORK 3

```
LD I0.2
R Q0.2, 6
```

4.5. Lệnh so sánh Byte, Word, Double word, số thực

Chúng ta có thể sử dụng các cách so sánh như: =, >=, <=, <>, <, >, cho các Byte, Word, Double word. Đầu ra bằng 1 khi so sánh đúng

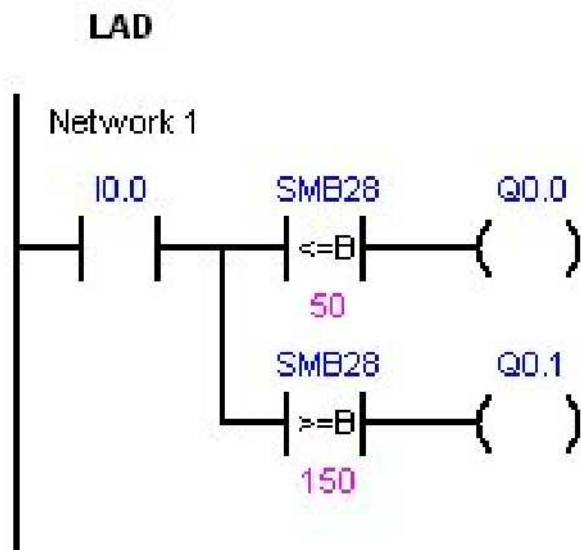
<input checked="" type="checkbox"/> SIMATIC <input type="checkbox"/> IEC 1131	
L A D	
F B D	
S T L	LDB= IN1, IN2 AB= IN1, IN2 OB= IN1, IN2 LDB<> IN1, IN2 AB<> IN1, IN2 OB<> IN1, IN2 LDB< IN1, IN2 AB< IN1, IN2 OB< IN1, IN2 LDB<= IN1, IN2 AB<= IN1, IN2 OB<= IN1, IN2 LDB> IN1, IN2 AB> IN1, IN2 OB> IN1, IN2 LDB>= IN1, IN2 AB>= IN1, IN2 OB>= IN1, IN2

<input checked="" type="checkbox"/> SIMATIC <input type="checkbox"/> IEC 1131	
L A D	
F B D	
S T L	LDW= IN1, IN2 AW= IN1, IN2 OW= IN1, IN2 LDW<> IN1, IN2 AW<> IN1, IN2 OW<> IN1, IN2 LDW< IN1, IN2 AW< IN1, IN2 OW< IN1, IN2 LDW<= IN1, IN2 AW<= IN1, IN2 OW<= IN1, IN2 LDW> IN1, IN2 AW> IN1, IN2 OW> IN1, IN2 LDW>= IN1, IN2 AW>= IN1, IN2 OW>= IN1, IN2

<input checked="" type="checkbox"/> SIMATIC <input type="checkbox"/> IEC 1131	
L A D	
F B D	
S T L	LDD= IN1, IN2 AD= IN1, IN2 OD= IN1, IN2 LDD<> IN1, IN2 AD<> IN1, IN2 OD<> IN1, IN2 LDD< IN1, IN2 AD< IN1, IN2 OD< IN1, IN2 LDD<= IN1, IN2 AD<= IN1, IN2 OD<= IN1, IN2 LDD> IN1, IN2 AD> IN1, IN2 OD> IN1, IN2 LDD>= IN1, IN2 AD>= IN1, IN2 OD>= IN1, IN2

<input checked="" type="checkbox"/> SIMATIC <input type="checkbox"/> IEC 1131	
L A D	
F B D	
S T L	LDR= IN1, IN2 AR= IN1, IN2 OR= IN1, IN2 LDR<> IN1, IN2 AR<> IN1, IN2 OR<> IN1, IN2 LDR< IN1, IN2 AR< IN1, IN2 OR< IN1, IN2 LDR<= IN1, IN2 AR<= IN1, IN2 OR<= IN1, IN2 LDR> IN1, IN2 AR> IN1, IN2 OR> IN1, IN2 LDR>= IN1, IN2 AR>= IN1, IN2 OR>= IN1, IN2

4.5. Lệnh so sánh Byte, Word, Double word, số thực



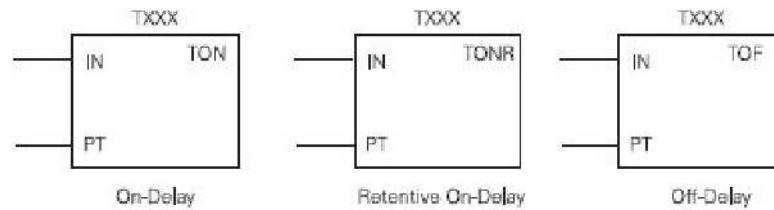
NETWORK 1

```
LD I0.0
LPS
AB<= SMB28 50
= Q0.0
LPP
AB>= SMB28 150
= Q0.1
```

4.6. Bộ timer: TON, TONR, TOF



4.6. Bộ timer: TON, TONR, TOF



- Hàm TON và TONR đếm thời gian khi có tín hiệu ở đầu vào IN, Khi giá trị đếm thời gian vượt qua giá trị đặt, bit Timer có giá trị 1.

Với bộ thời gian TON, giá trị đếm thời gian sẽ bị xoá khi mất tín hiệu ở đầu vào IN, còn với bộ thời gian TONR, giá trị đếm thời gian sẽ dừng lại tại thời điểm mất tín hiệu đầu vào IN và đếm tích lũy khi tiếp tục có tín hiệu ở đầu vào IN

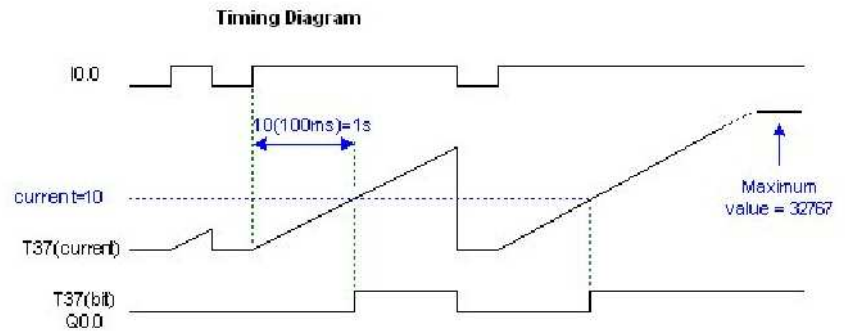
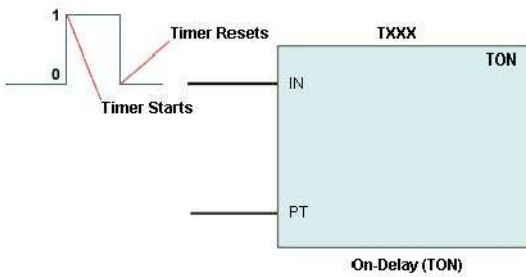
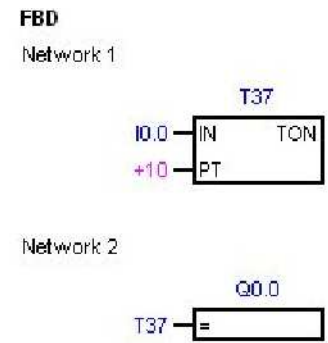
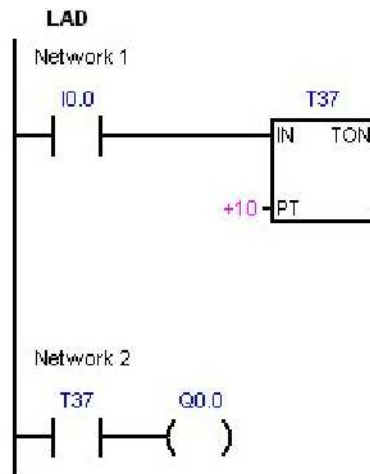
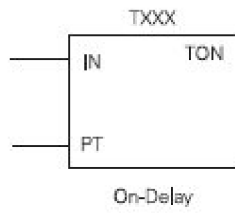
- Hàm TOF reset giá trị đếm thời gian và đặt bit Timer bằng 1 khi có tín hiệu ở đầu vào IN, khi mất tín hiệu ở đầu vào IN, bộ thời gian bắt đầu đếm thời gian và khi giá trị đếm thời gian bằng giá trị đặt bit Timer bị xoá về 0

- Tất cả các bộ đếm thời gian sẽ vẫn tiếp tục đếm thời gian khi đã đếm tới giá trị đặt, nó chỉ dừng lại khi đạt giá trị cực đại 32.767s

LẬP TRÌNH CHO PLC

4.6. Bộ timer: TON, TONR, TOF

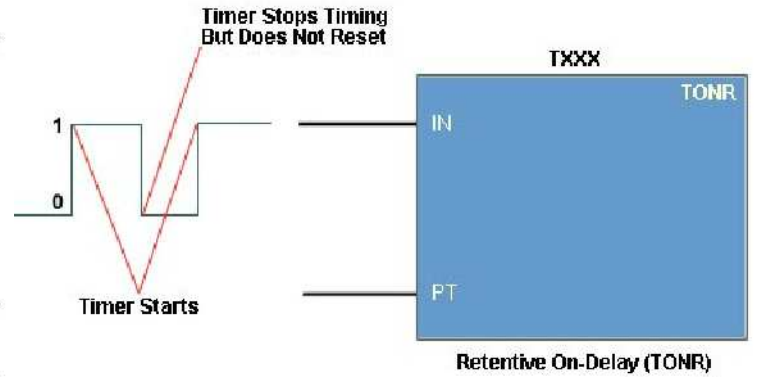
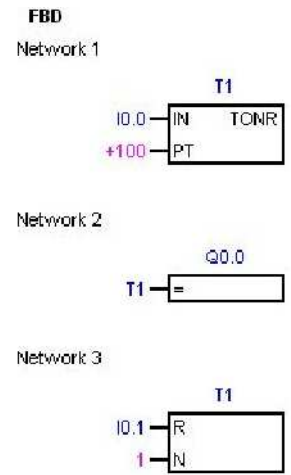
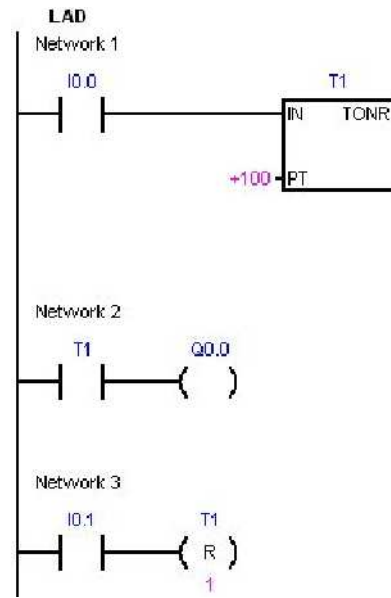
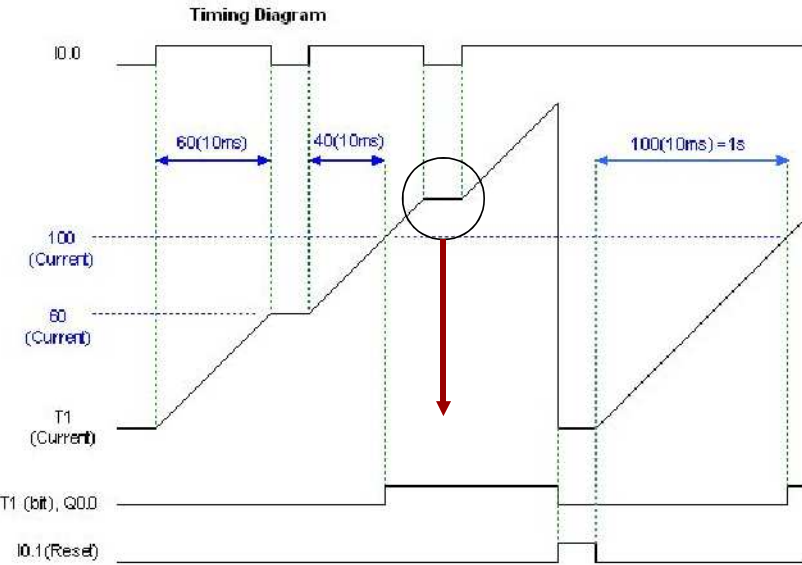
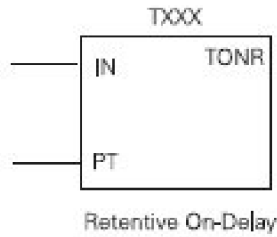
Ví dụ:



LẬP TRÌNH CHO PLC

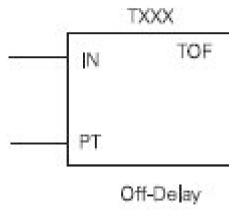
4.6. Bộ timer: TON, TONR, TOF

Ví dụ:

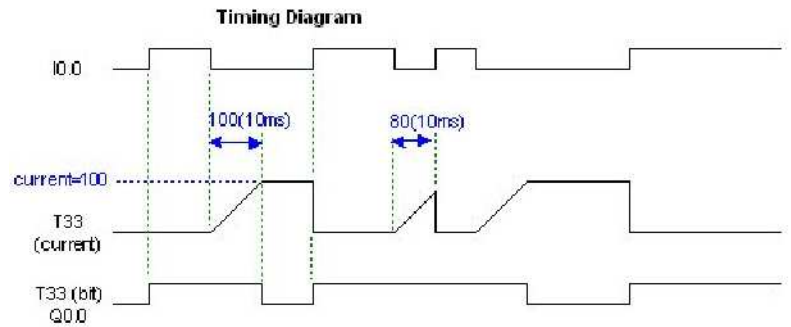
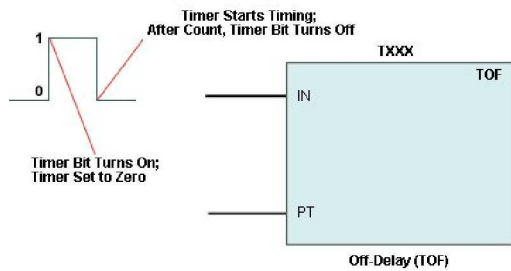
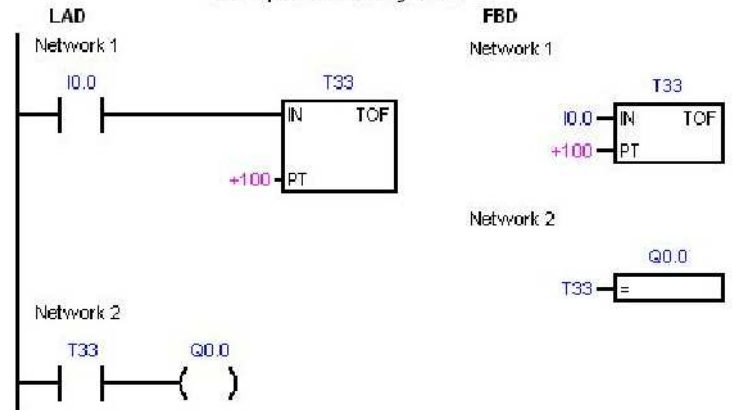


4.6. Bộ timer: TON, TONR, TOF

Ví dụ:



Example of Off-Delay Timer



4.6. Bộ timer: TON, TONR, TOF

Độ phân giải các bộ Timer

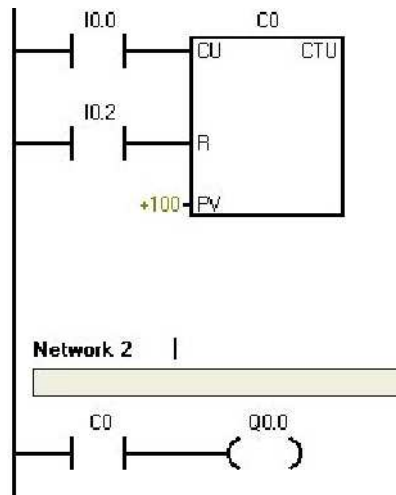
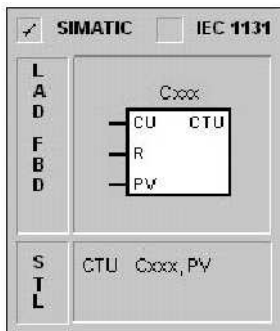
Timer Type	Resolution	Maximum Value	Timer Number
TONR	1 ms	32.767 seconds	T0, T64
	10 ms	327.67 seconds	T1-T4, T65-T68
	100 ms	3276.7 seconds	T5-T31, T69-T95
TON, TOF	1 ms	32.767 seconds	T32, T96
	10 ms	327.67 seconds	T33-T36, T97-T100
	100 ms	3276.7 seconds	T37-T63, T101-T255

Cần chú ý rằng, chúng ta không thể sử dụng cùng một địa chỉ cho hai loại bộ Thời gian, ví dụ TON T32 và TOF T32

4.7. Bộ Counter: CTU, CTUD, CTD

- Bộ đếm tiến: CTU

- Bộ đếm thuận CTU đếm tăng tại thời điểm sườn lên của tín hiệu đầu vào Cu. Khi nội dung bộ đếm bằng giá trị đặt PV, bộ đếm dừng lại và bit counter bằng 1. Bộ đếm bị reset khi có tín hiệu tại đầu vào R



Network 1

```
LD I0.0
LD I0.2
CTU C0, +100
```

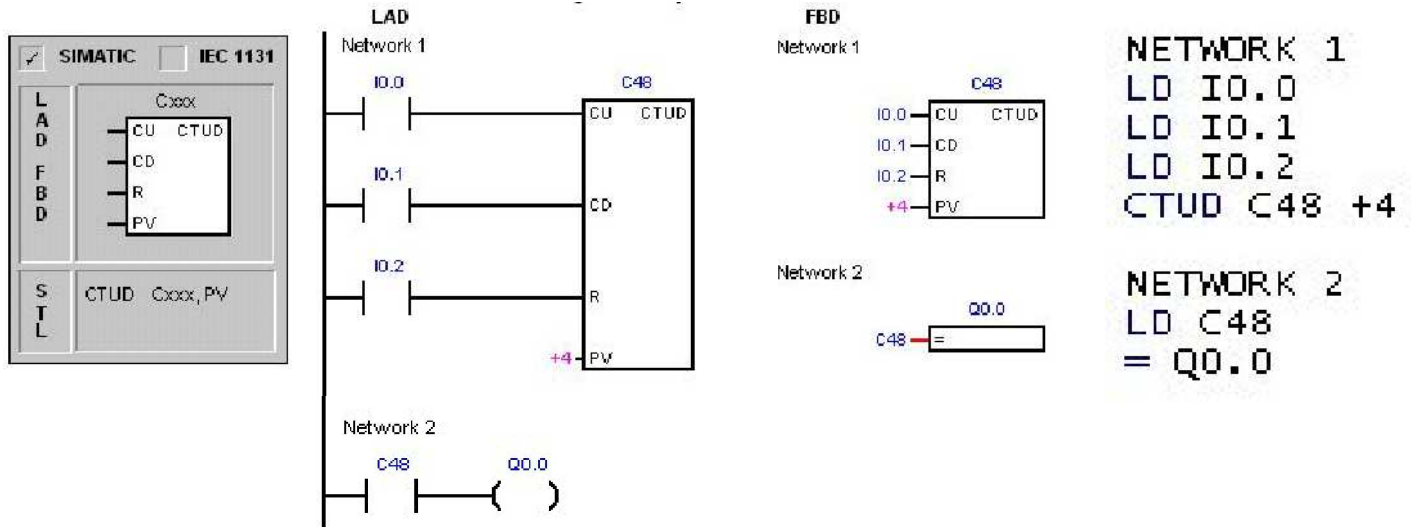
Network 2

```
LD C0
= Q0.0
```


4.7. Bộ Counter: CTU, CTUD, CTD

- Bộ đếm tiến lùi: CTUD

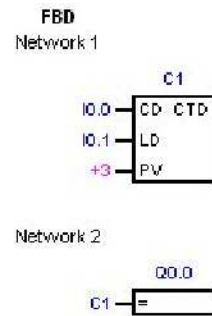
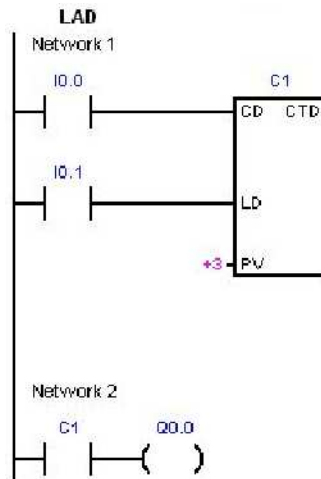
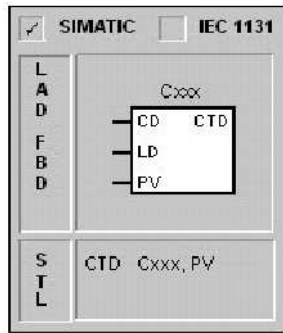
-Bộ đếm thuận nghịch CTUD đếm tăng tại thời điểm sườn lên của tín hiệu đầu vào CU và đếm nghịch tại thời điểm sườn lên của đầu vào tín hiệu CD, khi nội dung bộ đếm lớn hơn hoặc bằng giá trị đặt, bit counter bằng 1. Bộ đếm bị Reset khi có tín hiệu ở đầu vào R.



4.7. Bộ Counter: CTU, CTUD, CTD

- Bộ đếm lùi: CTD

-Bộ đếm nghịch CTD đếm nghịch tại thời điểm sườn lên của tín hiệu đầu vào CD, bộ đếm dừng lại và bit counter bằng 1 khi nội dung bộ đếm bằng 0, bộ đếm bị reset và được đặt giá trị PV khi có tín hiệu ở đầu vào LD



NETWORK 1

```
LD IO.0
LD IO.1
CTD C1 +3
```

NETWORK 2

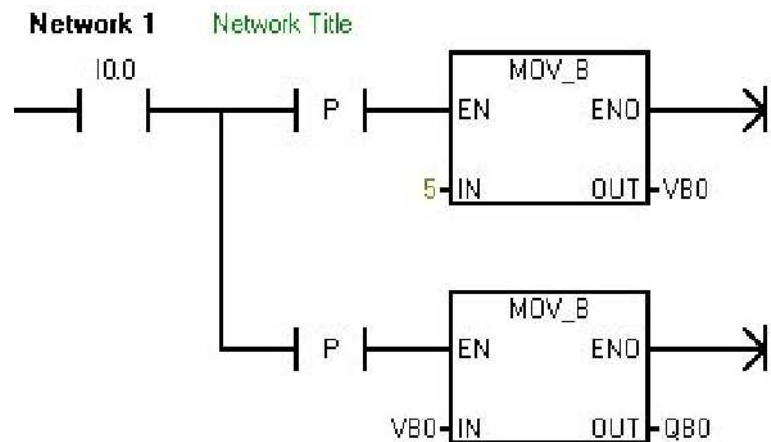
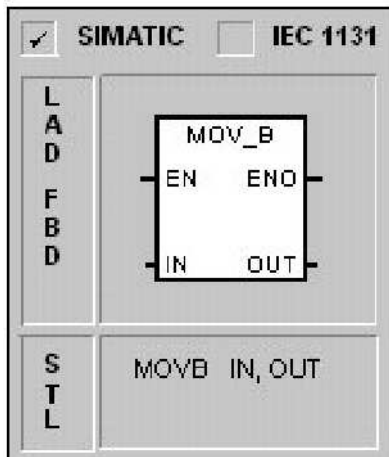
```
LD C1
= Q0.0
```

Chú ý: Nội dung bộ đếm nằm trong khoảng 0-32767

4.8. Khối chuyển dữ liệu: MOV

Thực hiện chuyển dữ liệu từ đầu vào IN tới đầu ra OUT khi có tín hiệu ở đầu vào EN, nếu việc chuyển dữ liệu thành công sẽ có tín hiệu ở đầu ENO.

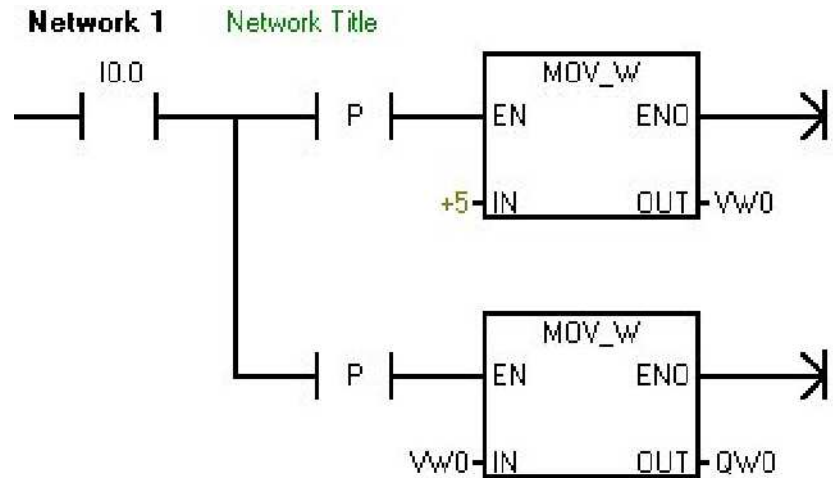
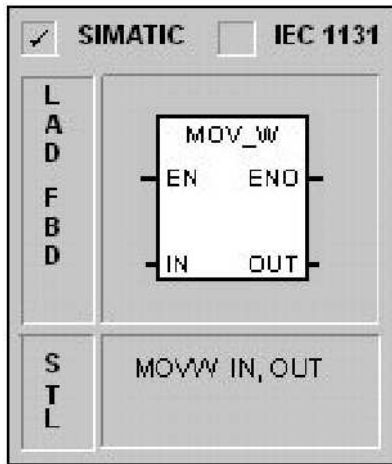
MOV_B: chuyển Byte dữ liệu



4.8. Khối chuyển dữ liệu: MOV

Thực hiện chuyển dữ liệu từ đầu vào IN tới đầu ra OUT khi có tín hiệu ở đầu vào EN, nếu việc chuyển dữ liệu thành công sẽ có tín hiệu ở đầu ENO.

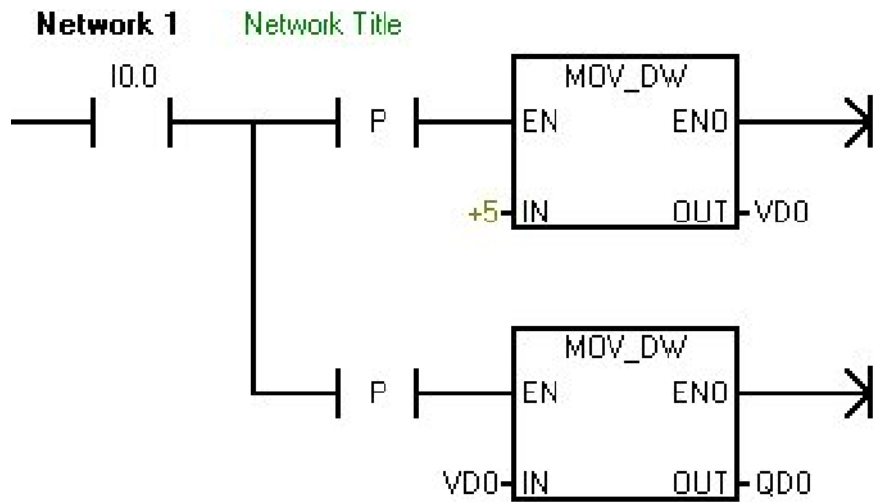
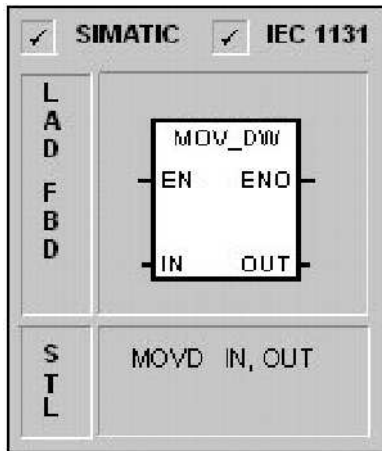
MOV_W: chuyển Word dữ liệu



4.8. Khối chuyển dữ liệu: MOV

Thực hiện chuyển dữ liệu từ đầu vào IN tới đầu ra OUT khi có tín hiệu ở đầu vào EN, nếu việc chuyển dữ liệu thành công sẽ có tín hiệu ở đầu ENO.

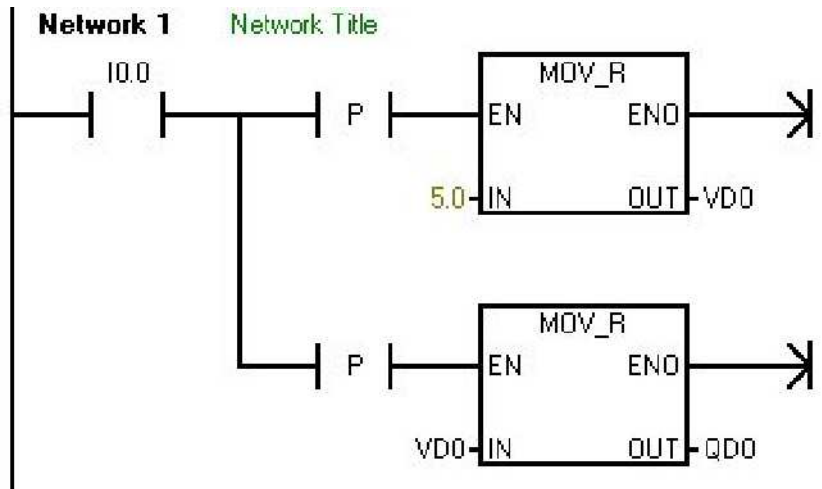
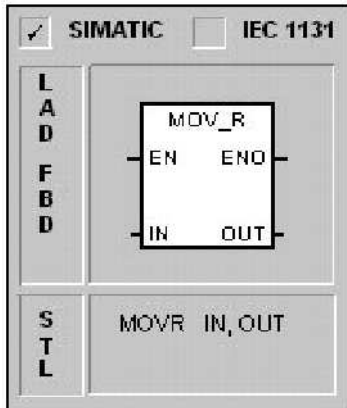
MOV_DW: chuyển Double Word dữ liệu



3.8. Khối chuyển dữ liệu: MOV

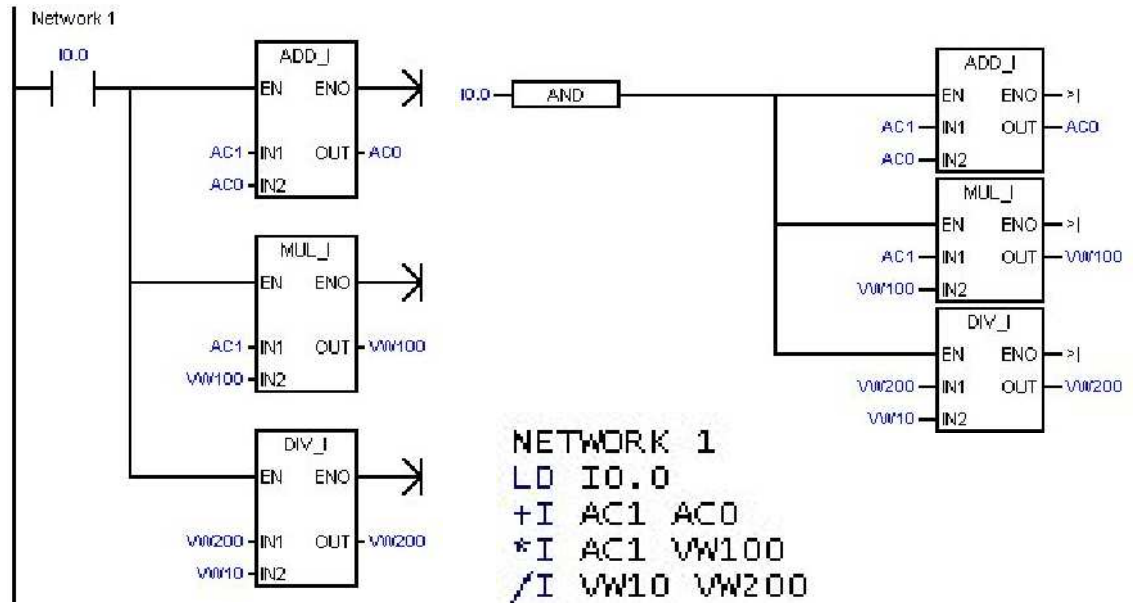
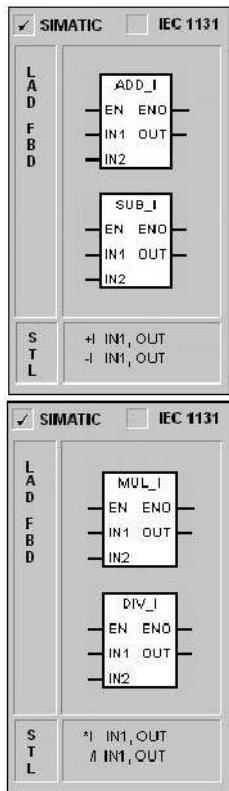
Thực hiện chuyển dữ liệu từ đầu vào IN tới đầu ra OUT khi có tín hiệu ở đầu vào EN, nếu việc chuyển dữ liệu thành công sẽ có tín hiệu ở đầu ENO.

MOV_R: chuyển số thực



4.9. Các hàm toán học: ADD, SUB, MUL, DIV

Các hàm xử lý toán học số nguyên 16 bit: *ADD_I, SUB_I, MUL_I, DIV_I*



In LAD and FBD: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

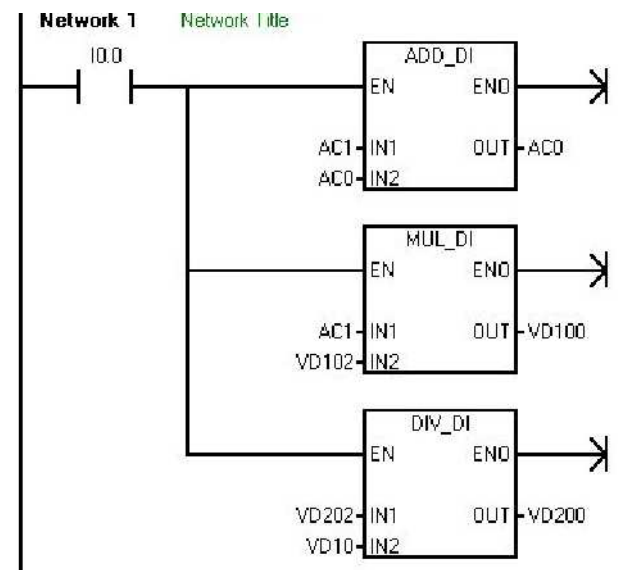
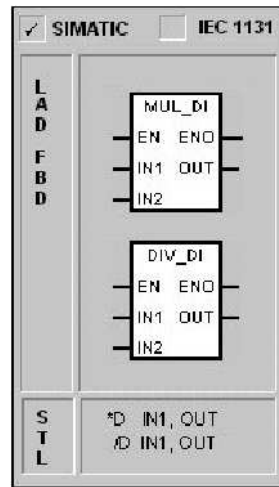
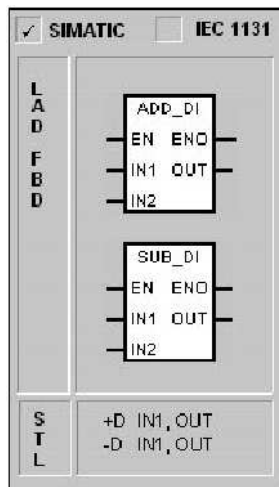
In STL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

In LAD and FBD: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

In STL: $IN1 * OUT = OUT$
 $OUT / IN1 = OUT$

4.9. Các hàm toán học: ADD, SUB, MUL, DIV

Các hàm xử lý toán học số nguyên 32 bit: *ADD_DI*, *SUB_DI*, *MUL_DI*, *DIV_DI*



In LAD and FBD: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

In STL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

In LAD and FBD: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

In STL: $IN1 * OUT = OUT$
 $OUT / IN1 = OUT$

LẬP TRÌNH CHO PLC

Chương 1: Tổng quan về PLC và logic kỹ thuật số.	6
1.1. Khái niệm PLC.....	6
1.2. Cơ sở phát triển.....	8
1.3. Cấu trúc PLC.....	14
1.4. Đặc điểm PLC:.....	19
1.5. Vòng quét (Cycle scan):.....	22
1.6. Ứng dụng PLC.....	23
1.7. Hệ đếm.....	24
1.8 Các phần tử logic cơ bản.....	30
Chương 2: Các phần tử vào/ra tín hiệu.	34
2.1. Các khái niệm cơ bản:.....	34
2.2. Các phần tử đầu vào:.....	38
2.3. Các phần tử đầu ra:.....	45
2.4. Ghép nối phần tử vào ra với PLC.....	51
2.5. Các modul trong hệ thống điều khiển PLC.....	52
Chương 3: Cấu trúc hệ thống các chương trình PLC.	