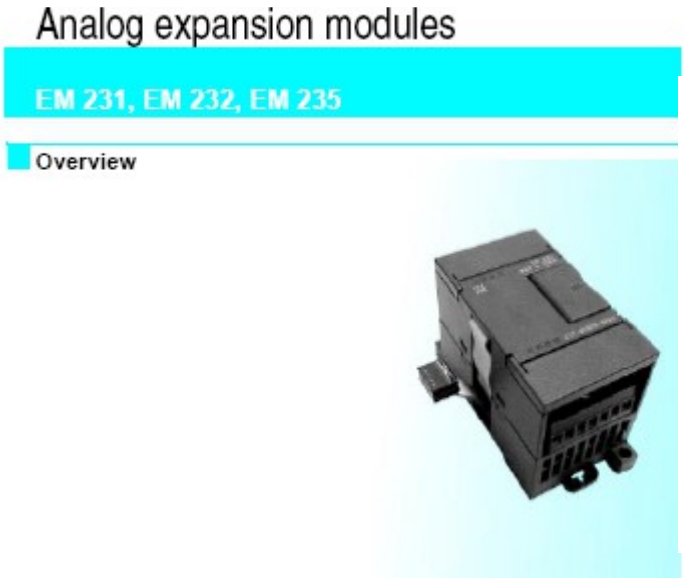
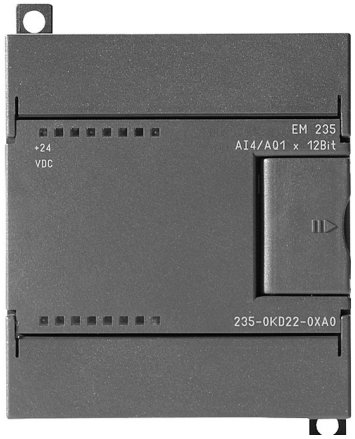
Bài 1. MODUL ANALOG





- **Đặc tính kỹ thuật:**
  - Thời gian chuyển đổi ngắn.
  - Không cần bộ khuếch đại khi kết nối với cảm biến.
  - Thực hiện được các công việc phức tạp.
- **Các thông số:**

  Số lượng ngõ vào : 3

  Số lượng ngõ ra : 1

  Tầm điện áp : 0 -10V, 0-5V, +/-5V, +/-2,5V, …

  Thông số ngõ vào : 0-10V, 0-20 mA

  Thông số ngõ ra : +/-10V, 0-20 mA

  Độ phân giải : 12 bit/V

  Kích thước : 71.2 x 80 x 62mm

  Trọng lượng : 186 g

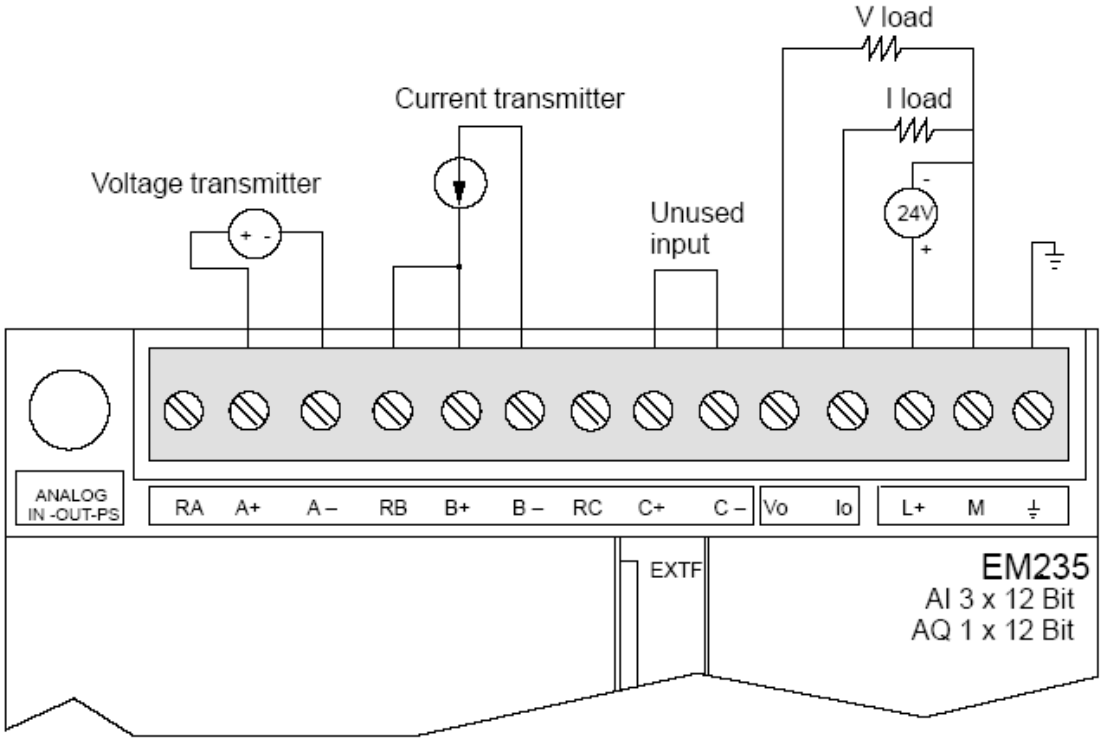
  Công suất tiêu thụ : 2 W

  Định dạng ngõ ra: có dấu : -32000 đến 32000, không dấu : 0 đến 32000
- **Kết nối:**

Modul mở rộng có các đặc tính thiết kế giống như CPU.

+ Lắp trên đường ray của thanh DIN: modul được lắp vào bên phải CPU thông qua bus (S7- 21x) hoặc cáp S7- 22x.

+ Lắp trực tiếp: thông qua cổng kết nối trên Modul.



➢ **Điều chỉnh ngõ vào:**

Việc điều chỉnh có ảnh hưởng đến trạng thái của thiết bị đo trong bộ khuếch đại do đó các kênh ngõ vào cũng bị ảnh hưởng theo. Sự thay đổi giá trị của mỗi thành phần trong từng mạch điện ngõ vào làm cho bộ chuyển đổi Analog đa thành phần có sự sai số nhỏ về giá trị đọc giữa các kênh dù được kết nối với cùng một tín hiệu ngõ vào.

Để thoả mãn được các đặc tính liệt kê trong Data Sheet, các bộ phận lọc ngõ vào phải được kích hoạt. Chọn chế độ 64 hoặc chế độ khác trong việc tính toán giá trị trung bình.

Việc điều chỉnh tuân theo các bước sau đây:

1. Tắt nguồn của Modul, chọn tầm ngõ vào thích hợp.

2. Cấp nguồn cho CPU và Modul. Để cho modul ổn định trong vòng 15 phút.

3. Sử dụng máy phát tín hiệu, nguồn áp hoặc nguồn dòng đặt tín hiệu có giá trị bằng 0 tới một trong những đầu nối của ngõ vào.

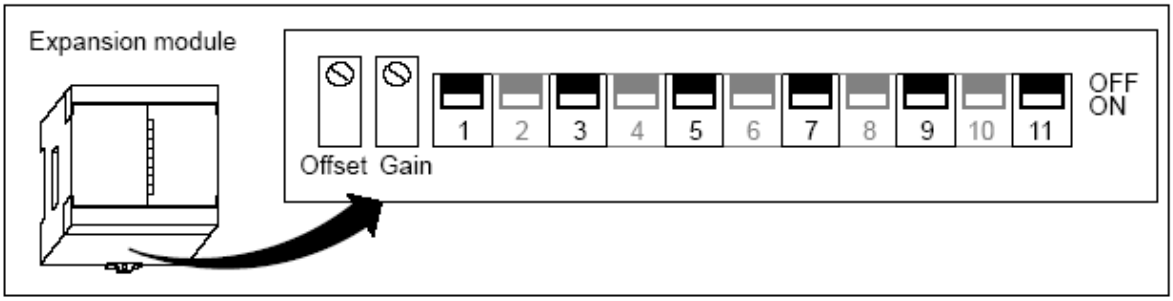
4. Đọc giá trị thu được cho CPU bằng kênh ngõ vào thích hợp.

5. Điều chỉnh OFFSET của máy đo điện thế cho đến khi bằng 0, hoặc giá trị dữ liệu dạng số mong muốn.

6. Kết nối một giá trị toàn thang tới một trong những đầu nối của ngõ vào. Đọc dữ liệu thu được cho CPU.

7. Điều chỉnh GAIN của máy đo điện thế cho đến khi bằng 32000, hoặc giá trị dữ liệu dạng số mong muốn.

8. Lặp lại sự chỉnh định OFFSET và GAIN theo yêu cầu.



- **Chỉnh định cho EM 235.**

Bảng A-4 trình bày cách chỉnh định cho EM 235 dùng các công tắc DIP. Công tắc từ 1 đến 6 dùng để chọn tầm cho ngõ vào và chọn độ phân giải.

Tất cả các ngõ vào đều phải có cùng dạng và tầm.

Table A-4 Configuration Switch Table for EM235 Analog Combination

Configuration Switch						Full-Scale Input	Resolution
1[1]	3	5	7	9	11		
ON	ON	OFF	ON	OFF	OFF	0 to 50 mV	12.5 µV
ON	ON	OFF	OFF	ON	OFF	0 to 100 mV	25 µV
ON	OFF	ON	ON	OFF	OFF	0 to 500 mV	125 µV
ON	OFF	ON	OFF	ON	OFF	0 to 1 V	250 µV
ON	OFF	OFF	ON	OFF	OFF	0 to 5 V	1.25 mV
ON	OFF	OFF	ON	OFF	OFF	0 to 20 mA[2]	5 µA
ON	OFF	OFF	OFF	ON	OFF	0 to 10 V	2.5 mV
OFF	ON	OFF	ON	OFF	OFF	±25 mV	12.5 µV
OFF	ON	OFF	OFF	ON	OFF	±50 mV	25 µV
OFF	ON	OFF	OFF	OFF	ON	±100 mV	50 µV
OFF	OFF	ON	ON	OFF	OFF	±250 mV	125 µV
OFF	OFF	ON	OFF	ON	OFF	±500 mV	250 µV
OFF	OFF	ON	OFF	OFF	ON	±1 V	500 µV
OFF	OFF	OFF	ON	OFF	OFF	±2.5 V	1.25 mV
OFF	OFF	OFF	OFF	ON	OFF	±5 V	2.5 mV
OFF	OFF	OFF	OFF	OFF	ON	±10 V	5 mV

- **Định dạng dữ liệu ngõ vào (dạng word) của EM 231 và EM 235.**

Hình A -21 giới thiệu nơi giá trị 12 bit dữ liệu được đặt trong địa chỉ Word của CPU.

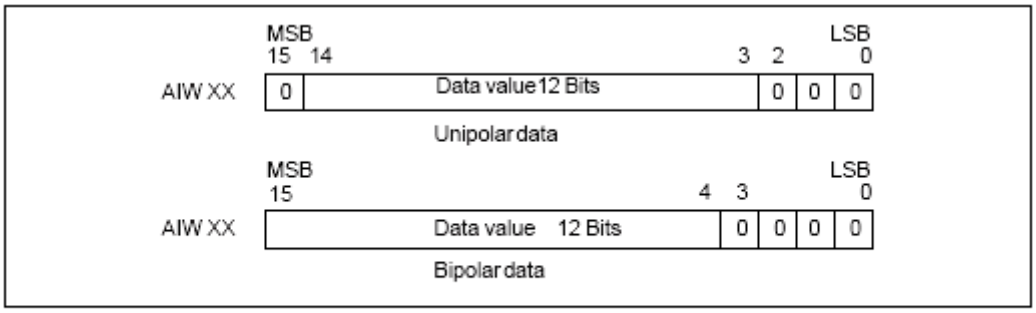


Figure A-21 Input Data Word Format for EM 231 and EM 235

- **Định dạng dữ liệu ngõ ra (dạng word) của EM 231 và EM 235.**

Hình A -23 giới thiệu nơi giá trị 12 bit dữ liệu được đặt trong địa chỉ Word của CPU.

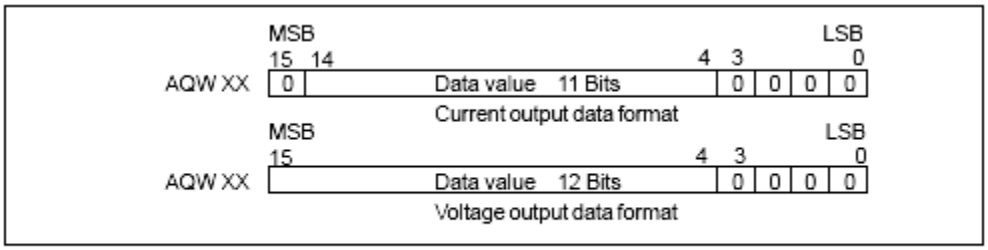
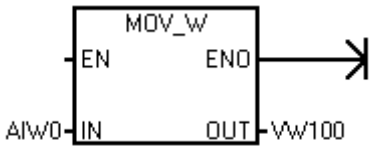
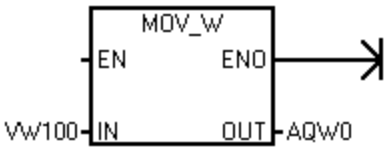


Figure A-23 Output Data Word Format for EM 232 and EM 235

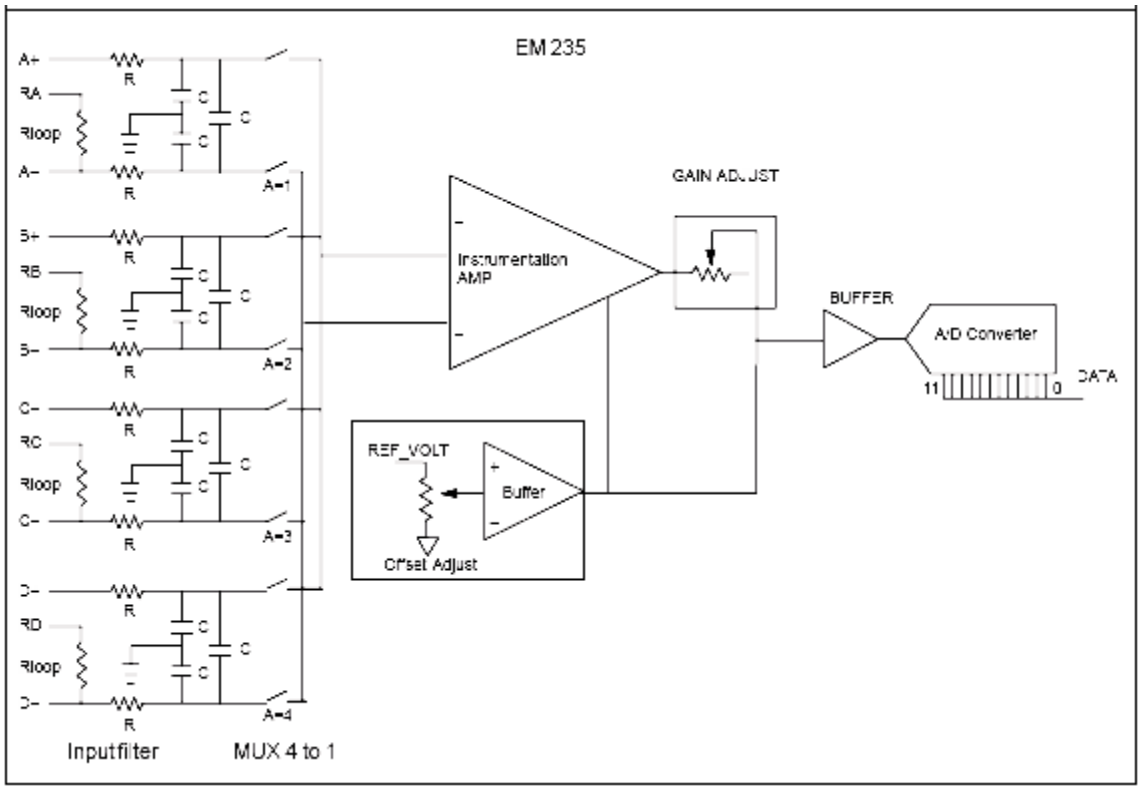
Để đọc tín hiệu tương tự ngõ vào ta sử dụng lệnh di chuyển dữ liệu:



Lệnh ghi dữ liệu tương tự ngõ ra:



- **Sơ đồ khối của EM 235**



- **Sơ đồ khối ngõ ra của EM 235.**

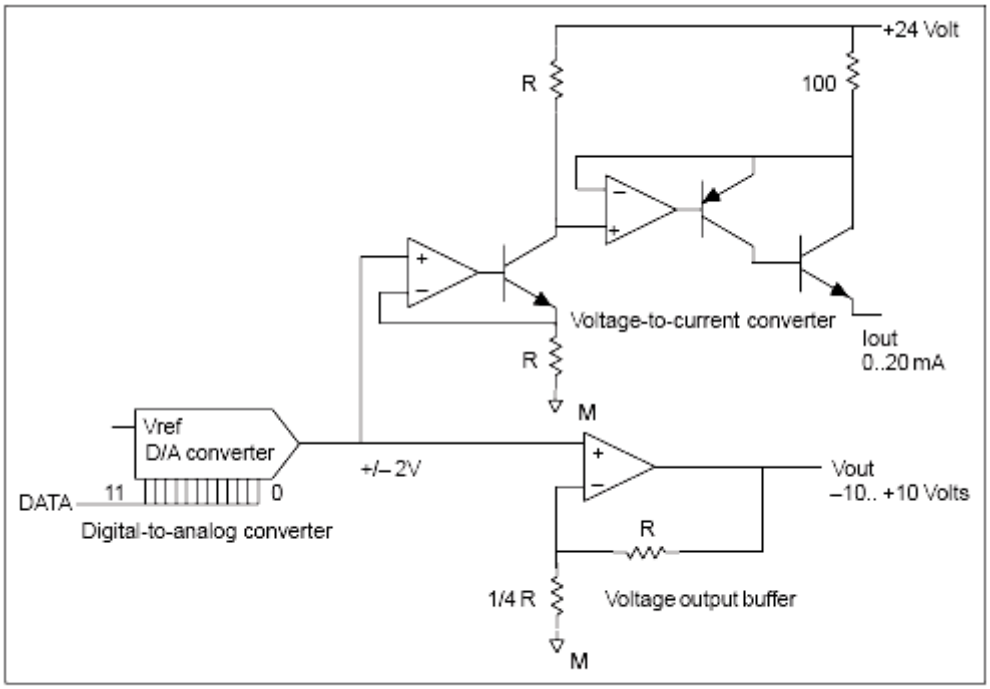


Figure A-24 EM 232 and EM 235 Output Block Diagram

Bài 2. CHƯƠNG TRÌNH CON

Chương trình con trong PLC giúp bạn chia nhỏ chương trình thành từng phần, các lệnh sử dụng trong chương trình chính của bạn sẽ quyết định việc thi hành các chương trình con. Khi chương trình chính gọi chương trình con để thực hiện, chương trình con sẽ tiến hành các lệnh của nó cho đến khi kết thức chương trình. Vì vậy hệ thống trả lại sự điều khiển cho chương trình chính tại Network mà từ đó chương trình con được gọi.

Chương trình con được sử dụng để chia chương trình của bạn thành những khối nhỏ và dễ quản lý hơn. Ưu điểm của công việc này là khi bạn tìm lỗi hoặc tiến hành sửa chữa, cải tiến những chương trình. Khi làm việc với những khối nhỏ hơn, bạn sẽ dễ dàng tìm được lỗi và sửa chữa chúng tốt hơn khi bạn làm việc với toàn bộ chương trình lớn. PLC có thể được sử dụng một cách hiệu quả hơn bằng việc gọi những khối khi cần cũng như tất cả các khối không phải thực hiện trong mỗi vòng quét của nó.

Cuối cùng thì những chương trình con có thể di chuyển được nếu chương trình con chỉ liên quan đến những thông số và bộ nhớ cục bộ của nó. Để cho chương trình con được linh động, tránh sử dụng các biến hoặc bộ nhớ toàn cục như I, Q, M, SM, AI, AQ, V, T, C, S, AC. Nếu chương trình con không gọi những tham số (IN, OUT, hoặc IN_OUT) hoặc chỉ sử dụng những biến cục bộ trong bộ nhớ L, bạn có thể xuất chương trình con này và nhập nó vào trong bất kỳ chương trình nào khác.

Để sử dụng chương trình con, bạn phải thực hiện 3 công việc sau:

- Tạo một chương trình con.
- Định nghĩa thông số của nó trong bảng các biến cục bộ của chương trình con.
- Gọi chương trình con từ một chương trình nào đó (chương trình chính, chương trình con khác hoặc chương trình ngắt).

Khi chương trình con được gọi, toàn bộ dữ liệu ngăn xếp sẽ được lưu lại và quyền điều khiển được chuyển sang cho chương trình con được gọi. Khi chương trình con hoàn thành, ngăn xếp được phục hồi lại với giá trị đã được lưu trữ và quyền điều khiển trả lại cho chương trình trước đó.

Thanh ghi tích lũy được sử dụng chung cho chương trình con và chương trình chứa nó. Hoạt động lưu trữ và phục hồi không thực hiện đối với thanh ghi tích lũy khi thực hiện chương trình con.

Chương trình con có thể chứa thông số kèm theo. Những thông số được định nghĩa trong bảng bộ nhớ cục bộ của chương trình con. Thông số phải được đặt tên (tối đa 23 ký tự), chọn kiểu biến (IN, OUT, TEMP, IN_OUT) và kiểu dữ liệu (BOOL, BYTE,…) 16 thông số có thể truyền vào hoặc lấy ra từ chương trình con.

Vùng chọn kiểu biến trong bảng biến cục bộ xác định biến nào được truyền vào chương trình con (IN), biến nào được lấy ra từ chương trình con (OUT) hoặc biến nào vừa truyền vào và sau đó lấy ra từ chương trình con (IN_OUT). Bảng sau đây mô tả các kiểu thông số của một chương trình con. Để thêm một thông số, di chuyển con

trỏ đến vùng kiểu biến mà bạn muốn thêm vào, nhấn chuột phải để nhận được bảng chọn, lựa chọn phần Insert và sau đó là chèn thêm hàng ở phía dưới (Row Below), những thông số khác của kiểu được chọn.

Kiểu thông số	Mô tả
IN	Thông số được truyền vào chương trình con, nếu thông số là một địa chỉ trực tiếp (chẳng hạn VB10), giá trị tại vị trí được xác định này sẽ được truyền vào chương trình con. Nếu thông số là một địa chỉ gián tiếp (chẳng hạn như *AC1), giá trị tại vị trí được chỉ tới sẽ được truyền vào chương trình con. Nếu thông số là một hằng số hoặc 1 địa chỉ, giá trị này cũng sẽ được truyền vào chương trình con.
IN_OUT	Giá trị của vị trí thông số được xác định sẽ được truyền vào chương trình con và kết quả từ chương trình con sẽ được trả lại cho thông số. Hằng số và địa chỉ không được phép sử dụng cho thông số IN_OUT.
OUT	Kết quả từ chương trình con sẽ được trả lại cho thông số. Hằng số và địa chỉ không được phép sử dụng cho thông số IN_OUT.
TEMP	Bất kỳ bộ nhớ cục bộ nào được sử dụng mà không thuộc các loại tham số trên có thể sử dụng như những thông số tạm thời trong chương trình con.

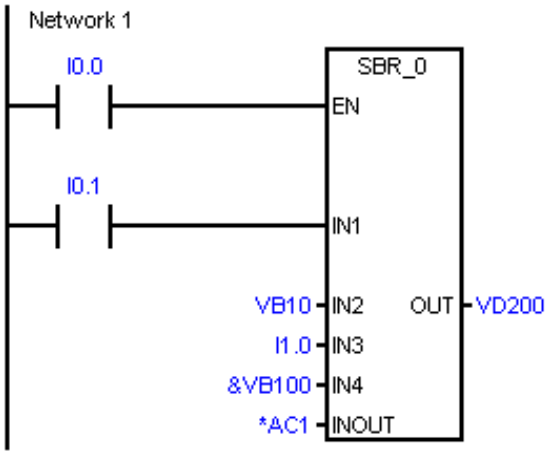
Kiểu dữ liệu trong bảng các biến cục bộ xác định kích thước và định dạng của tham số. Các kiểu tham số được trình bày trong bảng dưới đây.

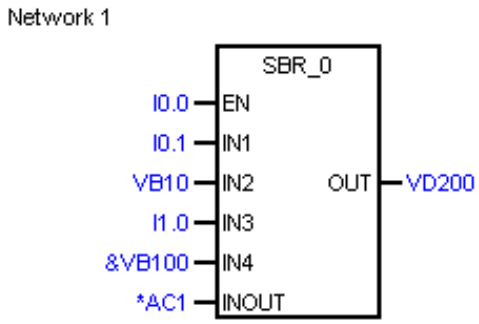
Kiểu dữ liệu	Mô tả
Power Flow	Dòng điện nguồn Boolean chỉ được cho phép đối với các ngõ vào bit. Sự khai báo này cho Step_7 biết rằng thông số ngõ vào này là kết quả của dòng điện nguồn dựa trên sự kết nối của một chỉ thị logic dạng bit. Dòng điện nguồn ngõ vào phải xuất hiện đầu tiên trong bảng biến cục bộ đước bất kỳ các loại ngõ vào khác. Chỉ thông số ngõ vào được cho phép sử dụng cách thức này. Ngõ vào cho phép (EN) và ngõ vào IN1 trong ví dụ dưới sử dụng logic Boolean.
BOOL	Kiểu dữ liệu này được sử dụng cho các bit đơn. Ngõ vào IN3 trong ví dụ là một ngõ vào BOOL.
BYTE,WORD,DWORD	Những kiểu dữ liệu này xác định một ngõ vào hoặc 1 ngõ ra không dấu lần lượt có độ lớn là 1 Byte, 2 Bytes hoặc 4 Bytes.

INT, DINT	Những kiểu dữ liệu này xác định một ngõ vào hoặc 1 ngõ ra có dấu lần lượt có độ lớn là 2 Bytes hoặc 4 Bytes.
REAL	Kiểu dữ liệu này xác định giá trị số thực 4 bytes.

Ví dụ một lệnh gọi chương trình con, bảng các biến cục bộ của chương trình con SBR_0

	Name	Var Type	Data Type	Comment
	EN	IN	BOOL	
L0.0	IN1	IN	BOOL	
LB1	IN2	IN	BYTE	
L2.0	IN3	IN	BOOL	
LD3	IN4	IN	DWORD	
		IN		
LD7	INOUT	IN_OUT	REAL	
		IN_OUT		
LD11	OUT	OUT	REAL	
		OUT		





Cách thức tạo chương trình con

Để tạo một chương trình con thực hiện theo các cách sau đây:

Lúc này cửa số chương trình sẽ chuyển từ cửa số chương trình đang thực hiện qua cửa số chương trình con và bạn có thể lập trình cho chương trình con.

- Nếu bạn muốn khai báo thông số cho chương trình con, bạn có thể sử dụng bảng các thông số cục bộ.

- Nếu bạn muốn lập trình thì sử dụng màn hình lập trình.

- Nếu bạn muốn chuyển sang một khối chương trình khác, hãy lựa chọn trên Tab chương trình mà bạn muốn sử dụng.

*** Chú ý** : Bạn không được sử dụng lệnh END để kết thúc chương trình con. Để thoát khỏi chương trình con thì lệnh RET sẽ được chương trình tự động thêm vào cuối chương trình và bạn không cần thêm lệnh để kết thúc chương trình con nữa.

Cách thức để gọi chương trình con.

Sau khi bạn đã tạo một chương trình con và đặt các tham số cho nó, bạn có thể thực hiện lệnh gọi chương trình con từ bất kỳ các khối tổ chức chương trình nào khác của bạn (Chương trình chính, chương trình con khác hoặc chương trình ngắt). Lưu ý bạn không thể gọi chương trình con từ chính nó.

Để chèn lệnh gọi chương trình con trong chương trình của bạn, hãy thực hiện theo các bước sau :

- Lựa chọn Network mà bạn muốn gọi chương trình con.

- Trong cây lệnh, mở thư mục Subroutines. Kéo và thả chương trình con bạn muốn gọi vào trong Network mà bạn đã lựa chọn.

- Nhập thông số cho chương trình con.

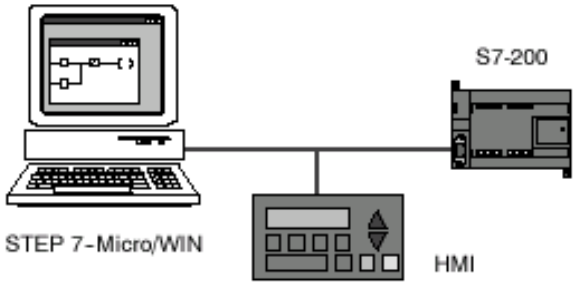
Bài 3. TRUYỀN THÔNG TRÊN MẠNG NHIỀU CHỦ

Các lệnh đọc và ghi dữ liệu từ một trạm này đến bộ nhớ của bất cứ trạm (PLC) nào khác trong mạng nhờ cổng truyền thông. Các trạm được xác định để giao lưu với nhau thông qua địa chỉ quy ước của chúng (PLC Station Address). Một mạng thông thường có thể có tới 32 trạm. Nếu sử dụng bộ lặp (Repeaters), số trạm trong mạng có thể được nâng lên con số tối đa là 125. Số dữ liệu nhiều nhất cho một lần truyền là 16 byte.

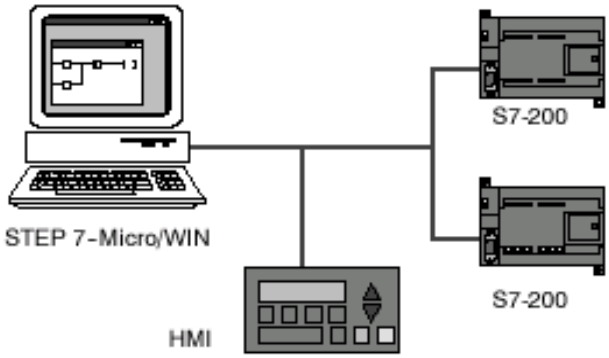
Các lệnh đọc ghi mảng dữ liệu của một trạm khác qua cổng PORT là NETR (NETwork Read: Đọc mạng) và NETW (NETwork Write: Ghi mạng). Các trạm liên kết cho việc truyền thông được xác định bằng địa chỉ quy ước của chúng. Cùng một lúc chỉ có thể sử dụng 8 lệnh đọc ghi qua mạng.

Hai lệnh NETR và NETW chỉ được sử dụng trong mạng có máy tính tham gia nếu như máy tính được nối với mạng qua cổng MPI (MultiPoint Interface). Khác với kiểu ghép nối PC/PPI với bộ chuyển đội RS232 thông qua RS485 mà ở đó thì máy tính là máy chủ duy nhất, trong mạng nối máy tính qua MPI có thể có nhiều chủ.

Trong mạng MPI tất cả các PLC với CPU từ 214 trở đi đều có thể là trạm chủ (Master) trong mạng, còn CPU 212 chỉ có thể là trạm tớ (Slave).



Máy tính và HMI là trạm chủ còn S7-200 là trạm tớ

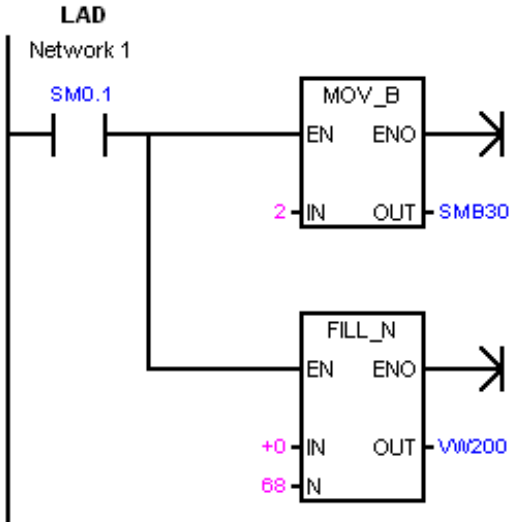


Máy tính và HMI là trạm chủ, 2 CPU S7-200 còn lại là các trạm tớ. Trạm chủ có thể yêu cầu dữ liệu từ bất kỳ trạm tớ nào.

Ngoài ra, để đảm bảo việc truyền thông được an toàn cần phải khai báo sử dụng PPI Protocol trước khi sử dụng lệnh NETR và NETW, tốt nhất là tại vòng quét đầu tiên của PLC.

PPI Protocol là một bộ xác lập kiểu truyền tin nối tiếp 11 bit cho PLC bao gồm 1 bit Start, 8 bit dữ liệu (byte), 1 bit kiểm tra chẵn lẻ và 1 bit Stop.

PPI Protocol được khai báo bằng cách ghi giá trị logic 1 vào bit 1 và logic 0 vào bit 0 của SMB30. PPI Protocol tạo khả năng truyền thông giữa nhiều máy chủ với nhiều máy tớ.



MOV_B giá trị 2 : 0000 0010 vào SMB30 để cho phép truyền thông PPI.

FILL_N giá trị 0 vào 68 ô nhớ bắt đầu từ VW200: xóa vùng bộ nhớ đệm để phát hoặc thu dữ liệu.

Kiểu truyền thông và mạng liên kết được xác định bằng nội dung quản lý bộ đệm truyền thông được chỉ thị trong lệnh bằng một toán hạng TABLE. Toán hạng:

TABLE (Byte): VB, MB, *VD, *AC.

PORT (Byte): hằng số 0.

Bộ đệm truyền thông có nhiều nhất là 23 byte được chia làm hai vùng: vùng các thông tin quản lý gồm 7 byte và vùng dữ liệu. Vùng dữ liệu có thể có tới 16 byte.

Cấu trúc của bộ đệm truyền thông, giả sử được khai báo trong trạm A để giao lưu dữ liệu với trạm B như sau:

Byte	7 6 5 4 3 2 1 0	
0	Các bit trạng thái	Xem bảng dưới
1	Địa chỉ trạm B	Địa chỉ của trạm B mà trạm A giao lưu để truyền thông.
2 3 4	Con trỏ gián tiếp chỉ vùng địa chỉ trong B	Con trỏ chỉ vào đầu mảng dữ liệu được truyền thông trong trạm B

5		
6	Độ dài mảng dữ liệu	Mảng dữ liệu truyền thông trong trạm A. nếu kiểu truyền thông với B là để đọc thì sau khi thực hiện lệnh NETR, dữ liệu đọc được của B được ghi vào đây. Ngược lại nếu kiểu truyền thông là gửi thì dữ liệu của A gửi sang B phải được ghi vào mảng này trước khi gọi lệnh NETW.
7	Dữ liệu 0	
8	Dữ liệu 1	
…	…	
22	Dữ liệu 15	

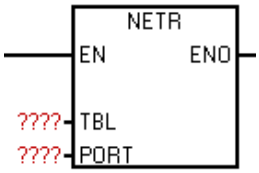
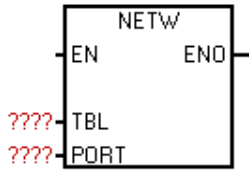
Byte đầu tiên của bộ đệm gồm 8 bit truyền thông có cấu trúc như sau:

0 3	0 không có lỗi truyền thông. 1 Lỗi quá thời gian trạm B không trả lời 2 Lỗi tín hiệu nhận (chẵn/lẽ)
1	3 Lỗi ghép nối(lặp địa chỉ hoặc lỗi phần cứng). 4 Lỗi xếp hàng, quá nhiều lệnh NETR và NETW.
2	5 Không khai báo PPI Protocol trong SM30. 6 Lỗi tham số. Bộ đệm truyền thông có tham số sai. 7 Trạm B bận.
3	8 Lỗi tổ chức. 9 Lỗi dữ liệu thông báo… A_F chưa sử dụng
4	0
5	0 Không có lỗi. 1 Có lỗi
6	0 Không làm việc. 1 Có làm việc
7	0 Chưa thực hiện xong việc truyền thông. 1 Thực hiện xong việc truyền thông

Lệnh NETR: Lệnh đọc một mảng dữ liệu của một trạm khác trong mạng qua cổng PORT. Kiểu truyền thông và trạm liên kết được xác định bằng nội dung quản lý bộ đệm truyền thông được chỉ thị trong lệnh bằng toán hạng TABLE.

Lệnh NETW: Lệnh ghi một mảng dữ liệu của một trạm khác trong mạng qua cổng PORT. Kiểu truyền thông và trạm liên kết được xác định bằng nội dung quản lý bộ đệm truyền thông được chỉ thị trong lệnh bằng toán hạng TABLE.

Cách sử dụng lệnh NETR và NETW như sau:

LAD	Toán hạng
	TBL: VB, MB,*VD,*AC PORT: hằng số 0 (với CPU từ 214 trở lên)
	TBL: VB, MB,*VD,*AC PORT: hằng số 0 (với CPU từ 214 trở lên)

Bài 4. NGẮT VÀ XỬ LÝ NGẮT

1. Giới thiệu

Các chế độ ngắt và xử lý ngắt cho phép thực hiện các quá trình tốc độ cao, phản ứng kịp thời với các sự kiện ở bên trong và bên ngoài.

Nguyên tắc cơ bản của một chế độ ngắt cũng giống như việc thực hiện lệnh gọi một chương trình con, sự khác nhau ở đây là việc gọi chương trình con được gọi một cách chủ động bằng lệnh CALL, còn xử lý ngắt được gọi bị động bằng một tín hiệu báo ngắt, hệ thống sẽ tổ chức gọi và thực hiện chương trình con tương ứng với tín hiệu báo ngắt đó, hay nói cách khác hệ thống sẽ tổ chức xử lý tín hiệu ngắt đó. Chương trình con này gọi là chương trình xử lý ngắt.

Do việc gọi chương trình xử lý ngắt bằng một tín hiệu báo ngắt mà thời điểm xuất hiện tín hiệu báo ngắt hoàn toàn bị động, bởi vậy hệ thống sẽ phải hỗ trợ thêm cho công việc xử lý ngắt như: cất giữ nội dung ngăn xếp, nội dung thanh ghi AC và các bit nhớ đặc biệt, tổ chức xếp hàng ưu tiên cho các tín hiệu báo ngắt trong trường hợp chúng chưa kịp xử lý.

Trong CPU 214 có các kiểu tín hiệu báo ngắt sau:

- Tám ngắt vào/ra theo sườn lên và sườn xuống của các cổng I0.0 I0.3.
- Hai ngắt thời gian.
- Hai ngắt truyền thông nối tiếp (nhận và truyền).
- Bảy ngắt bộ đếm tốc độ cao (CV=PV trên HSC0 và thay đổi, xóa ngoài và CV=PV trên HSC1 và HSC2).
- Hai ngắt đầu ra truyền xung là PTO0 và PTO1.

Khi có tín hiệu báo ngắt, giá trị cũ của ngăn xếp được cất đi, đỉnh của ngăn xếp nhận giá trị logic mới là 1 còn các bit khác của ngăn xếp nhận giá trị là 0, bởi vậy khi vào đầu một chương trình ngắt, lệnh có điều kiện trở thành không có điều kiện.

Các kiểu tín hiệu báo ngắt khác nhau được trình bày trong bảng sau:

Mô tả ngắt	Kiểu (mã hiệu)
Tín hiệu ngắt để nhận dữ liệu truyền thông	8
Tín hiệu ngắt báo hoàn tất việc gởi dữ liệu	9
Ngắt theo sườn lên của I0.0*	0
Ngắt theo sườn lên của I0.1	2
Ngắt theo sườn lên của I0.2	4
Ngắt theo sườn lên của I0.3	6

Ngắt theo sườn xuống của I0.0*	1
Ngắt theo sườn xuống của I0.1	3
Ngắt theo sườn xuống của I0.2	5
Ngắt theo sườn xuống của I0.3	7
Ngắt theo HSC0, khi giá trị tức thời bằng giá trị đặt trước*	12
Ngắt theo HSC1, khi giá trị tức thời bằng giá trị đặt trước	13
Ngắt theo HSC1, khi có tín hiệu báo đổi hướng đếm từ bên ngoài	14
Ngắt theo HSC1, khi có tín hiệu reset từ bên ngoài	15
Ngắt theo HSC2, khi giá trị tức thời bằng giá trị đặt trước	16
Ngắt theo HSC2, khi có tín hiệu báo đổi hướng đếm từ bên ngoài	17
Ngắt theo HSC2, khi có tín hiệu reset từ bên ngoài	18
Ngắt theo PLS0 báo hòan tất việc đếm xung	19
Ngắt theo PLS1 báo hòan tất việc đếm xung	20
Ngắt theo thời gian T0	10
Ngắt theo thời gian T1	11

* Nếu khai báo sử dụng ngắt kiểu 12 thì hai kiểu ngắt 0 và 1 đều bị vô hiệu hóa hoặc ngược lại nếu khai báo kiểu ngắt 0 hoặc 1 thì kiểu ngắt 12 sẽ bị vô hiệu hóa.

a) Thứ tự ưu tiên:

Thứ tự ưu tiên của các kiểu ngắt khác nhau đã được cứng hóa từ trước theo nguyên tắc: tín hiệu báo ngắt nào có trước thì xử lý trước. Tuy nhiên những trường hợp các tín hiệu ngắt xuất hiện cùng một lúc hoặc một tín hiệu ngắt xuất hiện trong khi hệ thống đang phải xử lý một chương trình ngắt khác thì chúng sẽ phải tuân theo quy định ưu tiên sau:

Nhóm ưu tiên	CPU 214
Ngắt truyền thông	4
Ngắt vào/ra	16

Ngắt thời gian	4

Riêng với tín hiệu ngắt truyền thông, mặc dù chưa được xử lý nhưng ký tự nhận được cùng bit kiểm tra chẵn lẻ vẫn được ghi nhớ theo đúng thứ tự ngắt.

b) Khai báo và hủy toàn cục chế độ ngắt:

Khi ở chế độ RUN toàn bộ chế độ ngắt khai báo trước sẽ tự động bị vô hiệu hóa, tất cả các chế độ ngắt có thể được kích cùng lúc bằng việc thực hiện việc kích chế độ ngắt toàn cục ENI.

Lệnh hủy toàn cục DISI sẽ vô hiệu hóa tất cả các chế độ ngắt. Tuy vậy vẫn cho phép các tín hiệu ngắt được xếp hàng chờ, nhưng không cho phép thực hiện bất cứ một chương trình xử lý ngắt nào.

c) Khai báo và hủy một chế độ ngắt:

Để khai báo một chế độ ngắt phải thực hiện hai công việc kích tín hiệu báo ngắt cho chế độ đó bằng lệnh ATCH và khai báo chương trình xử lý ngắt tương ứng bằng lệnh ngắt INT. Có thể gộp nhiều tín hiệu báo ngắt vào một chương trình xử lý ngắt, song một tín hiệu báo ngắt chỉ có thể có nhiều nhất một chương trình xử lý ngắt. Có thể hủy bỏ từng chế độ ngắt riêng biệt bằng lệnh DTCH. Lệnh này sẽ đặt một chế độ ngắt vào trạng thái không tích cực.

d) Chương trình xử lý ngắt:

Cũng giống như một chương trình con, chương trình xử lý ngắt có một nhãn riêng được đánh dấu tại đầu chương trình. Nhãn này được khai báo bằng lệnh INT. Tất cả các lệnh nằm giữa nhãn của chương trình xử lý ngắt và lệnh quay về không điều kiện RETI của chương trình xử lý ngắt đều thuộc về nội dung của chương trình xử lý ngắt.

Chương trình xử lý ngắt phải được viết một cách tối ưu về thời gian thực hiện. Chỉ nên sử dụng chương trình ngắt trong thời gian ngắn và thực hiện chương trình này thật nhanh. Khi viết một chương trình ngắt cần phải tuân theo những quy tắc sau đây:

- Càng ngắn càng tốt.

- Đặt chương trình ngắt sau chương trình chính.

- Không được sử dụng các lệnh: DISI, ENI, CALL, HDEF, FOR…NEXT và END trong chương trình ngắt.

- Cần phải kết thúc chương trình ngắt bằng lệnh quay về không điều kiện: RETI.

e) Ngắt truyền thông:

Kiểu điều khiển truyền thông bằng chương trình được gọi là kiểu điều khiển cổng tự do (Freeport Control). Khi sử dụng chế độ truyền thông này, kiểu biên bản truyền thông freeport , tốc độ truyền tín hiệu, số bit được truyền cho một ký tự,

chế độ kiểm tra (parity) phải được định nghĩa trước trong byte đặc biệt SMB30 như sau:

SMB30

p	p	d	b	b	b	m	m

SMB30	Giải thích
m	Kiểu truyền thông: 0 Point to Point (PPI) 1 Freeport 10 PPI (bình dẳng) 11 Không sử dụng
b b b	Tốc độ truyền (Baud): 000 38.400 (CPU214) hay19.200 (CPU212). 001 19.200 010 9.600 011 4.800 100 2.400 101 1.200 110 600 111 300
d	Số bit truyền: 0 8 bit 1 7 bit
p p	Kiểu kiểm tra (parity): 0 Không kiểm tra. 1 Kiểm tra chẵn (even). 10 Không kiểm tra. 11 Kiểm tra lẻ (Odd)

Chế độ ngắt truyền thông Freeport để truyền hay nhận dữ liệu có thể làm cho việc điều khiển truyền thông trong chương trình được dễ dàng hơn. Chế độ ngắt truyền thông Freeport chỉ có thể thực hiện được khi PLC ở chế độ làm việc RUN. Khai báo kiểu Freeport cho ngắt truyền thông bằng cách đặt mã nhị phân 01 vào 2 bit thấp của SMB30.

Khi PLC hoạt động ở chế độ truyền thông Freeport không thể thực hiện truyền thông giữa PLC và thiết bị lập trình.

Một vài ô nhớ đặc biệt được hệ thống sử dụng cho kiểu truyền thông Freeport. Các ô nhớ đó là:

- Byte SMB32 được dùng để ghi nhớ dữ liệu nhận vào.

- SM3.0 được sử dụng để báo lỗi kiểm tra chẵn lẻ. Nếu lỗi chẵn lẻ được phát hiện SM3.0 sẽ được đặt lên 1.

- SM4.5 được sử dụng để thông báo khi việc truyền dữ liệu đã hoàn tất.

Lệnh XMT dùng để gởi dữ liệu đi trong chế độ truyền thông Freeport. XMT cho phép gởi đi một

f) Ngắt vào/ra:

Thuộc nhóm tín hiệu báo ngắt vào/ ra là:

- Tín hiệu báo ngắt khi có sườn lên hoặc sườn xuống của cổng đầu vào.

- Tín hiệu báo ngắt của bộ đếm tốc độ cao.

- Tín hiệu báo ngắt của cổng truyền xung.

CPU 212 có khả năng tạo ra tín hiệu báo ngắt mỗi khi tại cổng vào I0.0 có sự thay đổi trạng thái từ 0 lên 1 (sườn lên) hoặc từ 1 xuống 0 (sườn xuống).

CPU 214 có khả năng tạo ra tín hiệu báo ngắt mỗi khi tại cổng vào I0.0 -:- I0.3 có sự thay đổi trạng thái từ 0 lên 1 (sườn lên) hoặc từ 1 xuống 0 (sườn xuống).

Việc sử dụng sườn lên hoặc sườn xuống làm tín hiệu báo ngắt có thể chiếm một cổng trong các ngõ vào.

g) Ngắt thời gian:

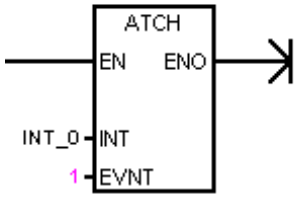
Tín hiệu báo ngắt theo thời gian được phát ra đều đặn theo chu kỳ thời gian. Chu kỳ phát tín hiệu báo ngắt theo thời gian là một số nguyên trong khoảng 5ms 255ms và được xác định bởi giá trị của SMB34, cho tín hiệu báo ngắt thời gian 0 và của SMB35 cho tín hiệu báo ngắt thời gian 1.

Tín hiệu báo ngắt theo thời gian này cho phép thực hiện gọi chương trình xử lý ngắt một cách đều đặn nên chúng thường được dùng trong việc lấy mẫu tín hiệu tại các cổng vào tương tự với tần số lấy mẫu được lập trình trước trong SMB34 hoặc SMB35.

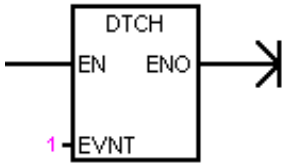
Một tín hiệu báo ngắt theo thời gian được kích đúng tại thời điểm nó được khai báo bằng lệnh ATCH. Cũng tại thời điểm đó chu kỳ phát tín hiệu báo ngắt bắt đầu với chu kỳ phát được quy định trong SMB34 hoặc SMB35. Mọi sự thay đổi về sau của nội dung 2 ô nhớ đặc biệt này không ảnh hưởng đến chu kỳ đã khai báo. Muốn thay đổi lại chu kỳ phát tín hiệu báo ngắt, bắt buộc phải khai báo lại bằng lệnh ATCH.

Sau khi được khai báo sử dụng, chế độ ngắt theo thời gian thực hiện đều đặn cho đến khi PLC rời khỏi chế độ hoạt động RUN hoặc cho đến khi ngắt thời gian bị hủy bỏ bằng lệnh DTCH.

ATCH: lệnh khai báo sử dụng một chế độ ngắt với kiểu (hay mã hiệu) được xác định bằng toán hạng EVENT(byte). Chương trình xử lý ngắt tương ứng được xác định bằng toán hạng INT(byte) trong lệnh. Sau khi khai báo tín hiệu báo ngắt của chế độ ngắt đó cũng được kích theo.



DTCH: Lệnh hủy bỏ một chế độ ngắt đã được khai báo sử dụng trước đó. Kiểu (hay mã hiệu) của nó được xác định bằng toán hạng EVENT (byte) trong lệnh.



INT: Lệnh sử dụng để khai báo một chương trình xử lý ngắt. Nhãn của chương trình được xác định trong toán hạng n của lệnh INT.

CRETI, RETI: là lệnh kết thúc có điều liện và không có điều kiện của một chương trình xử lý ngắt. Lệnh kết thúc không điều kiện thường đặt ở cuối chương trình xử lý ngắt.

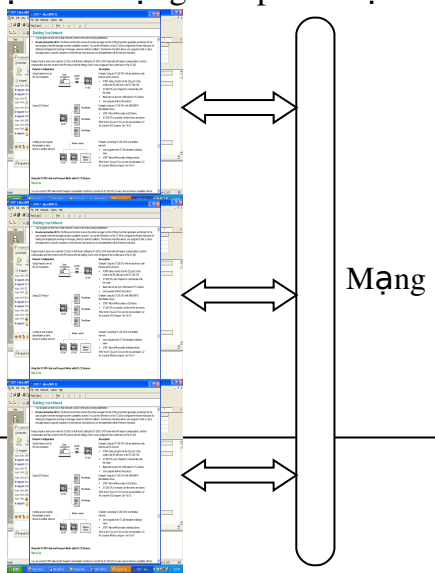
DISI: Lệnh hủy bỏ toàn cục các chế độ ngắt đã được khai báo sử dụng trước đó. Lệnh này chỉ có tác dụng treo các tín hiệu báo ngắt lên và không xử lý chúng, nhưng vẫn cho phép chúng được ghi vào hàng chờ xử lý.

ENI: Lệnh khai báo toàn cúc các chế độ ngắt hoặc kích hoạt lại tất cả các chế độ ngắt đã bị hủy trước đó bằng lệnh DISI.

2. Sử dụng ngắt truyền thông:

Khác với kiểu truyền thông MPI, khi mà một trạm chủ có thể truy cập đến tận bộ nhớ của trạm tớ thì ở kiểu truyền thông Freeport chỉ có khả năng đọc hoặc gởi dữ liệu lên mạng, mọi trạm đều có thể nhận và gởi dữ liệu trên mạng.

Kiểu trao đổi dữ liệu với mạng Freeport được mô tả như trong hình:



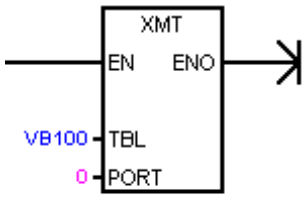
Lệnh gởi dữ liệu lên mạng có tên là XMT nhưng lại không có lệnh riêng để đọc dữ liệu từ mạng xuống. Muốn lấy dữ liệu trên mạng vào trạm phải khai báo chế độ truyền thông tương ứng trong SMB30 và sử dụng chế độ ngắt truyền thông mã hiệu 8.

a) Ghi dữ liệu lên mạng:

Lệnh XMT làm việc với cổng truyền thông nối tiếp của PLC lệnh ghi dữ liệu lên mạng. Dữ liệu gởi lên mạng phải được tổ chức thành một bảng các byte trong bộ nhớ, trong đó byte đầu tiên phải chứa độ dài dữ liệu của bảng. Số dữ liệu nhiều nhất có thể gởi là 255 byte. Hệ thống sử dụng bit SM4.5 để thông báo trạng thái quá trình gởi dữ liệu lên mạng. Trong khi gởi dữ liệu, SM4.5 có giá trị bằng 0 và sau khi byte dữ liệu cuối cùng được gởi đi SM4.5 nhận giá trị logic 1, tín hiệu ngắt 9 xuất hiện nếu trước đó nó đã được kích hoạt.

Toán hạng TABLE (VB, IB, QB, MB, SMB, *VD, *AC) của lệnh xác định mảng dữ liệu được chuyển. Toán hạng PORT 0 xác định tên cổng truyền thông được sử dụng.

Byte nhớ SMB30 xác định dạng tín hiệu nối tiếp của dữ liệu cũng như tốc độ truyền dữ liệu lên mạng. Trước khi sử dụng lệnh XMT để gởi dữ liệu bắt buộc phải khai báo kiểu truyền thông Freeport.



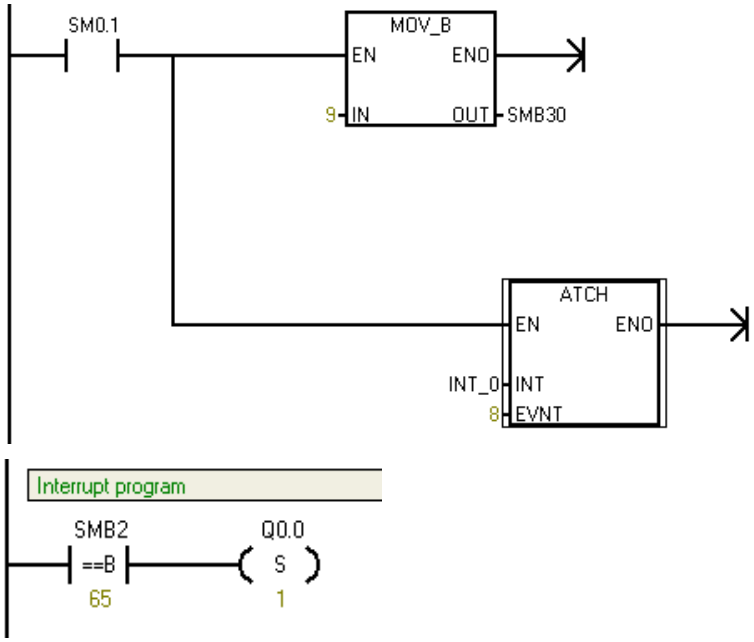
Sơ đồ tổ chức mảng dữ liệu như sau:

Độ dài mảng dữ liệu = n
Dữ liệu 1
… …
Dữ liệu n

b) Đọc dữ liệu từ mạng:

Khi chế độ ngắt truyền thông kiểu 8 đã được khai báo và chế độ truyền thông freeport phù hợp đã được xác định trong SMB30 thì mỗi khi có dữ liệu cùng kiểu trên mạng, dữ liệu sẽ được chuyển vào ô nhớ SMB2 theo từng byte. Khác với việc ghi dữ liệu lên mạng, việc nhận dữ liệu nên được tiến hành trong chế độ ngắt truyền thông. Mỗi khi nhận được một ký tự, hệ thống sẽ phát ra một tín hiệu báo ngắt có mã hiệu 8. Những ký tự nhận được sẽ được hệ thống cất giữ trong SMB2 và chuyển nó tới vùng mong muốn.

Ví dụ sau mô tả một đoạn chương trình để đọc dữ liệu từ mạng xuống.



Chương trình trên cho phép nhận dữ liệu từ trên mạng và so sánh kết quả với giá trị 65, nếu dữ liệu nhận được bằng 65 thi đóng tiếp điểm Q0.0.

Bài 5. SỬ DỤNG BỘ ĐẾM TỐC ĐỘ CAO

Bộ đếm tốc độ cao được sử dụng để theo dõi và điều khiển các quá trình có tốc độ cao mà PLC không thể khống chế được đo bị hạn chế về thời gian của vòng quét. CPU 214 có 3 bộ đếm tốc độ cao có tên là HSC0, HSC1 và HSC2.

Nguyên tắc hoạt động của các bộ đếm tốc độ cao cũng tương tự như bộ đếm thông thường khác của PLC tức là cũng đếm theo cạnh sườn lên của xung. Số đếm được được lưu trong một ô nhớ đặc biệt kiểu Double Word (DW) với ký hiệu là CV (Current Value). Khi giá trị của bộ đếm tức thời bằng với giá trị đặt trước thì bộ đếm phát ra một tín hiệu ngắt. Giá trị đặt trước là một số nguyên 32 bits cũng được lưu trong 1 ô nhớ kiểu DW.

Nếu chế độ ngắt vào/ra với bộ đếm tốc độ cao được khai báo sử dụng, các tín hiệu ngắt sau đây sẽ được phát.

- Ngắt khi PC = CV (đối với HSC0, HSC1, HSC2)

- Ngắt khi có tín hiệu báo thay đổi hướng đếm từ cổng vào (đối với HSC1 và HSC2).

- Ngắt khi có tín hiệu báo xóa (Reset) từ cổng vào (đối với HSC1 và HSC2).

Mỗi bộ đếm có nhiều chế độ làm việc khác nhau. Chọn chế độ làm việc cho một bộ đếm hoạt động bằng lệnh HDEF. Từng chế độ làm việc lại có kiểu hoạt động khác nhau. Kiểu hoạt động của mỗi bộ đếm được xác định bằng nội dung của một byte điều khiển trong vùng nhớ đặc biệt sau đó được khai báo bằng lệnh HSC.

Chỉ có thể kích bộ đếm sau khi đã khai báo chế độ làm việc và định nghĩa kiểu hoạt động cho từng chế độ trong byte điều khiển.

a) Nguyên lý hoạt động của bộ đếm tốc độ cao:

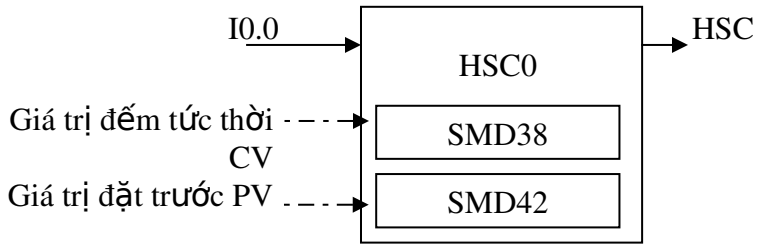
HSC0: Bộ đếm HSC0 có một cổng vào là I0.0. Nó chỉ có một chế độ làm việc duy nhất là đếm tiến và đếm lùi số các sườn lên của tín hiệu đầu vào tại cổng I0.0. Chiều đếm tiến và đếm lùi được quy định bởi trạng thái của bit SM37.3 như sau:

0 : đếm lùi theo sườn lên của I0.0.

1 : đếm tiến theo sườn lên của I0.0.

Tần số đếm cực đại cho phép của HSC0 là 2KHz.

Giá trị đếm tức thời của HSC0 là từ kép SMD38, giá trị đặt trước nằm ở từ kép SMD42. Sơ đồ nguyên lý của HSC0 được mô tả như trong hình sau:



HSC0 sử dụng byte SMB37 để xác định kiểu hoạt động như đếm tiến hay đếm lùi, cho phép hay không cho phép thay đổi giá trị đếm tức thời CV cũng như giá trị đặt trước PV và cho phép hay không cho phép kích bộ đếm. Kiểu hoạt động của HSC0 phải được định nghĩa trong SMB37 trước khi thực hiện lệnh HDEF.

Cấu trúc byte SMB37, còn được gọi là byte điều khiển của HSC0 như sau:

SM37.0	Không sử dụng
SM37.1	Không sử dụng
SM37.2	Không sử dụng
SM37.3	Chiều đếm: 0 đếm lùi, 1 : đếm lên
SM37.4	Cho phép đổi chiều đếm, 0: không cho phép, 1: cho phép
SM37.5	Cho phép sửa đổi giá trị đặt trước, 0: không cho phép, 1: cho phép
SM37.6	Cho phép sửa đổi giá trị đếm tức thời, 0: không cho phép, 1: cho phép
SM37.7	1- Cho phép kích HSC0, 0 – Cho phép hủy HSC0

Các bước khai báo sử dụng HSC0 (nên sử dụng ở vòng quét đầu tiên)

- Nạp giá trị điều khiển phù hợp cho SMB37

- Xác định chế độ làm việc cho bộ đếm bằng lệnh HDEF. Do HSC0 chỉ có một chế độ làm việc nên lệnh xác định kiểu sẽ là HDEF 0 0.

- Nạp giá trị đếm tức thời ban đầu và giá trị đặt trước vào SMD38 và SMD42.

- Khai báo sử dụng chế độ ngắt vào ra và kích tín hiệu báo ngắt HSC0 bằng lệnh ATCH.

- Kích bộ đếm bằng lệnh HSC 0.

Sau khi được kích, bộ đếm HSC0 bắt đầu làm việc và sử dụng byte SMB36 để báo trạng thái hoạt động của nó như sau:

SM36.0	Không sử dụng
SM36.1	Không sử dụng
SM36.2	Không sử dụng
SM36.3	Không sử dụng
SM36.4	Không sử dụng

SM36.5	Chiều đang đếm, 1 – đếm lên, 0 – đếm xuống.
SM36.6	Kết quả so sánh tức thời, 0 – nếu CV PV, 1 – nếu CV = PV
SM36.7	Kết quả so sánh tức thời, 0 – nếu CV PV, 1 – nếu CV > PV

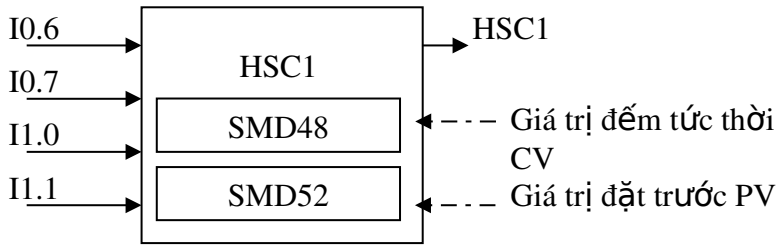
Khi sử dụng bộ đếm HSC0 cùng với chế độ ngắt vào ra, tín hiệu báo ngắt sẽ xuất hiện khi CV = PV nếu tín hiệu ngắt 12 đã được khai báo.

HSC1 (chỉ có trong CPU214): là bộ đếm linh hoạt, sử dụng bốn đầu vào I0.6, I0.7, I1.0, I1.1 với 12 chế độ làm việc khác nhau. HSC1 sử dụng từ kép SMD48 để lưu giá trị tức thời, giá trị đặt trước nằm ở từ kép SMD52, các giá trị đặt trước và tức thời là những số nguyên có dấu.

Chế độ làm việc	Mô tả
0	Đếm tiến hoặc lùi sườn lên của I0.6. Chiều đếm được quy định bởi SM47.3. Hoạt động của HSC1 hoàn toàn được điều khiển bởi hệ thống (không có tín hiệu kích và xóa từ bên ngoài).
1	Đếm tiến hoặc lùi sườn lên của I0.6. Chiều đếm được quy định bởi SM47.3. HSC1 sử dụng I1.0 làm tín hiệu xóa (reset)từ bên ngoài.
2	Đếm tiến hoặc lùi sườn lên của I0.6. Chiều đếm được quy định bởi SM47.3. HSC1 sử dụng I1.0 làm tín hiệu xóa (reset)từ bên ngoài; I1.1 làm tín hiệu khởi phát (start).
3	Đếm tiến hoặc lùi sườn lên của I0.6. Chiều đếm được quy định bởi cổng I0.7. Hoạt động của HSC1 hoàn toàn được điều khiển bởi hệ thống (không có tín hiệu kích và xóa từ bên ngoài).
4	Đếm tiến hoặc lùi sườn lên của I0.6. Chiều đếm được quy định bởi cổng I0.7. HSC1 sử dụng I1.0 làm tín hiệu xóa (reset)từ bên ngoài.
5	Đếm tiến hoặc lùi sườn lên của I0.6. Chiều đếm được quy định bởi cổng I0.7. HSC1 sử dụng I1.0 làm tín hiệu xóa (reset)từ bên ngoài; I1.1 làm tín hiệu khởi phát (start).
6	Đếm tiến theo sườn lên của I0.6 và đếm lùi theo sườn lên của I0.7. Hoạt động của HSC1 hoàn toàn được điều khiển bởi hệ thống (không có tín hiệu kích và xóa từ bên ngoài).
7	Đếm tiến theo sườn lên của I0.6 và đếm lùi theo sườn lên của I0.7. HSC1 sử dụng I1.0 làm tín hiệu xóa (reset)từ bên ngoài.
8	Đếm tiến theo sườn lên của I0.6 và đếm lùi theo sườn lên của I0.7.

	HSC1 sử dụng I1.0 làm tín hiệu xóa (reset)từ bên ngoài. I1.1 làm tín hiệu khởi phát (start).
9	Đếm số lần lệch trạng thái của hai cổng vào I0.6 và I0.7 tức là khi kết quả của phép tính logic XOR giữa I0.6 và I0.7 là 1. Hoạt động của HSC1 hoàn toàn được điều khiển bởi hệ thống (không có tín hiệu kích và xóa từ bên ngoài).
10	Đếm số lần lệch trạng thái của hai cổng vào I0.6 và I0.7 tức là khi kết quả của phép tính logic XOR giữa I0.6 và I0.7 là 1. HSC1 sử dụng I1.0 làm tín hiệu xóa (reset)từ bên ngoài.
11	Đếm số lần lệch trạng thái của hai cổng vào I0.6 và I0.7 tức là khi kết quả của phép tính logic XOR giữa I0.6 và I0.7 là 1. HSC1 sử dụng I1.0 làm tín hiệu xóa (reset)từ bên ngoài. I1.1 làm tín hiệu khởi phát (start).

Sơ đồ nguyên lý của bộ đếm HSC1



Khác với HSC0, HSC1 có 3 khả năng đếm:

- Tiến và lùi theo sườn lên của I0.6

- Tiến theo sườn lên của I0.6 và lùi theo sườn xuống của I0.7

- Tiến và lùi số lần lệch giá trị logic giữa hai cổng cào I0.6 và I0.7, tức là số lần phép tính XOR của I0.6 và I0.7 có kết quả là 1.

Chiều đếm (tiến hay lùi) trong chế độ 0, 1 và 2 được quy định bởi trạng thái của bit SM47.3 như sau:

SM47.3 = 0: đếm lùi theo sườn lên của I0.6

1: đến tiến theo sườn lên của I0.6

và trong chế độ 3, 4, 5 bởi đầu vào của I0.7:

I0.7 = 0: đếm lùi theo sườn lên của I0.6

1: đến tiến theo sườn lên của I0.6

Tần số biến đổi trạng thái cực đại cho phép của tín hiệu đầu vào tại cổng I0.6 và I0.7 là 7KHz.

HSC1 có hai tần số đếm. Trong những chế độ 0 →8, tần số đếm của HSC1 bằng tần số thay đổi trạng thái của ngõ vào do đó tần số đếm cực đại là 7KHz, riêng chế độ 9, 10 và 11, tùy theo khai báo sử dụng mà tần số này có thể bằng hoặc gấp 4 lần tần số biến thiên trạng thái kết quả của phép tính XOR giữa I0.6 và I0.7. Bởi vậy trong chế độ 9, 10, 11 tần số đếm cực đại cho phép của HSC1 sẽ là 28KHz. Hệ thống sử dụng bit SM47.2 để định nghĩa tần số đếm của HSC1 trong chế độ 9,10,11 là:

SM47.2 = 0: tần số đếm gấp 4 lần tần số thay đổi trạng thái I0.6 XOR I0.7

1: tần số đếm bằng tần số thay đổi trạng thái I0.6 XOR I0.7

Trong các chế độ 1, 2, 4, 5, 7, 8, 10 và 11, HSC1 sử dụng tín hiệu reset từ ngòai là I1.0. Bit SM47.0 được hệ thống sử dụng nhằm định nghĩa kiểu reset của HSC1 cho các chế độ này:

SM47.0 = 0: reset khi I1.0 có giá trị logic 1

1: reset khi I1.0 có giá trị logic 0

Cũng tương tự như vậy cho các chế độ 2, 5, 8 và 11, khi HSC1 sử dụng tín hiệu kích từ ngòai cổng I1.1, hệ thống sử dụng bit SM47.1 để định nghĩa kiểu kich HSC1 như sau:

SM47.1 = 0: HSC1 làm việc khi khi I1.1 có giá trị logic 1

1: HSC1 làm việc khi khi I1.1 có giá trị logic 0

HSC1 sử dụng byte SMB47 để xác định kiểu hoạt động như đếm tiến, lùi, cho phép hay không cho phép thay đổi giá trị đếm tức thời CV cũng như giá trị đặt trước PV, cho phép hay không cho phép kích bộ đếm. Kiểu hoạt động của HSC1 phải được định nghĩa trong SMB47 trướckhi thực hiện lệnh HDEF

Cấu trúc byte SMB47 như sau:

SM47.0	Kiểu reset cho tín hiệu xóa tại cổng I1.0
SM47.1	Kiểu start cho tín hiệu kích tại cổng I1.1
SM47.2	Tần số đếm của HSC1
SM47.3	Chiều đếm: 0 đếm lùi, 1 : đếm lên
SM47.4	Cho phép đổi chiều đếm, 0: không cho phép, 1: cho phép
SM47.5	Cho phép sửa đổi giá trị đặt trước, 0: không cho phép, 1: cho phép
SM47.6	Cho phép sửa đổi giá trị đếm tức thời, 0: không cho phép, 1: cho phép
SM47.7	1- cho phép kích HSC1, 0 – cho phép hủy HSC1

Các bit SM47.0, SM47.1 và SM47.2 được gán cho bộ đếm bằng lệnh HDEF, các bit còn lại gán cho bộ đếm sau lệnh HSC.

Các bước khai báo sử dụng HSC1 (nên thực hiện trong vòng quét đầu tiên):

- Nạp giá trị điều khiển phù hợp cho SMB47
- Xác định chế độ làm việc cho bộ đếm bằng lệnh HDEF
- Nạp giá trị đếm tức thời và giá trị đặt trước
- Khai báo sử dụng chế độ ngắt vào/ra và kích tín hiệu báo ngắt HSC1 bằng lệnh ATCH.
- Kích bộ đếm bằng lệnh HSC.

Khi sử dụng HSC1 cùng với chế độ ngắt vào/ra, các tín hiệu báo ngắt sau đây sẽ được phát:

- Báo ngắt khi CV = PV (ngắt 13)
- Báo ngắt khi có thay đổi chiều đếm từ I0.7 (ngắt 14)
- Báo ngắt khi HSC1 bị reset bởi I1.0 (ngắt 15)

Sau khi được kích hoạt bằng lệnh HSC, bộ đếm HSC1 sử dụng byte SMB46 để báo trạng thái hoạt động của nó như sau:

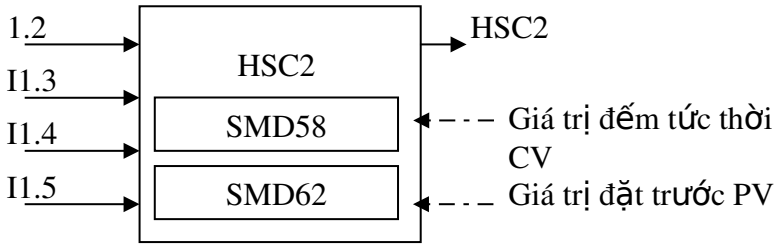
SM46.0	Không sử dụng
SM46.1	Không sử dụng
SM46.2	Không sử dụng
SM46.3	Không sử dụng
SM46.4	Không sử dụng
SM46.5	Chiều đang đếm, 1 – đếm lên, 0 – đếm xuống.
SM46.6	Kết quả so sánh tức thời, 0 – nếu CV PV, 1 – nếu CV = PV
SM46.7	Kết quả so sánh tức thời, 0 – nếu CV PV, 1 – nếu CV > PV

HSC2 (chỉ có trong CPU214): là bộ đếm linh hoạt, cũng giống như HSC1 nhưng chúng làm việc độc lập với nhau; thay vì sử dụng bốn đầu vào I0.6, I0.7, I1.0, I1.1 thì HSC2 sử dụng các đầu vào I1.2, I1.3, I1.4, I1.5 với 12 chế độ làm việc khác nhau:

Chế độ làm việc	Mô tả
0	Đếm tiến hoặc lùi sườn lên của I1.2. Chiều đếm được quy định bởi

	SM57.3. Hoạt động của HSC2 hoàn toàn được điều khiển bởi hệ thống (không có tín hiệu kích và xóa từ bên ngoài).
1	Đếm tiến hoặc lùi sườn lên của I1.2. Chiều đếm được quy định bởi SM57.3. HSC2 sử dụng I1.4 làm tín hiệu xóa (reset)từ bên ngoài.
2	Đếm tiến hoặc lùi sườn lên của I1.2. Chiều đếm được quy định bởi SM57.3. HSC2 sử dụng I1.4 làm tín hiệu xóa (reset)từ bên ngoài; I1.5 làm tín hiệu khởi phát (start).
3	Đếm tiến hoặc lùi sườn lên của I1.2. Chiều đếm được quy định bởi cổng I1.3. Hoạt động của HSC2 hoàn toàn được điều khiển bởi hệ thống (không có tín hiệu kích và xóa từ bên ngoài).
4	Đếm tiến hoặc lùi sườn lên của I1.2. Chiều đếm được quy định bởi cổng I1.3. HSC2 sử dụng I1.4 làm tín hiệu xóa (reset)từ bên ngoài.
5	Đếm tiến hoặc lùi sườn lên của I1.2. Chiều đếm được quy định bởi cổng I1.3. HSC2 sử dụng I1.4 làm tín hiệu xóa (reset)từ bên ngoài; I1.5 làm tín hiệu khởi phát (start).
6	Đếm tiến theo sườn lên của I1.2 và đếm lùi theo sườn lên của I1.3. Hoạt động của HSC2 hoàn toàn được điều khiển bởi hệ thống (không có tín hiệu kích và xóa từ bên ngoài).
7	Đếm tiến theo sườn lên của I1.2 và đếm lùi theo sườn lên của I1.3. HSC2 sử dụng I1.4 làm tín hiệu xóa (reset)từ bên ngoài.
8	Đếm tiến theo sườn lên của I1.2 và đếm lùi theo sườn lên của I1.3. HSC2 sử dụng I1.4 làm tín hiệu xóa (reset)từ bên ngoài. I1.5 làm tín hiệu khởi phát (start).
9	Đếm số lần lệch trạng thái của hai cổng vào I1.2 và I1.3 tức là khi kết quả của phép tính logic XOR giữa I1.2 và I1.3 là 1. Hoạt động của HSC2 hoàn toàn được điều khiển bởi hệ thống (không có tín hiệu kích và xóa từ bên ngoài).
10	Đếm số lần lệch trạng thái của hai cổng vào I1.2 và I1.3 tức là khi kết quả của phép tính logic XOR giữa I1.2 và I1.3 là 1. HSC2 sử dụng I1.4 làm tín hiệu xóa (reset)từ bên ngoài.
11	Đếm số lần lệch trạng thái của hai cổng vào I1.2 và I1.3 tức là khi kết quả của phép tính logic XOR giữa I1.2 và I1.3 là 1. HSC2 sử dụng I1.4 làm tín hiệu xóa (reset)từ bên ngoài. I1.5 làm tín hiệu khởi phát (start).

Sơ đồ nguyên lý của bộ đếm HSC1



Giống như HSC1, HSC2 có 3 khả năng đếm:

- Tiến và lùi theo sườn lên của I1.2

- Tiến theo sườn lên của I1.2 và lùi theo sườn xuống của I1.3

- Tiến và lùi số lần lệch giá trị logic giữa hai cổng cào I1.2 và I1.3, tức là số lần phép tính XOR của I1.2 và I1.3 có kết quả là 1.

Chiều đếm (tiến hay lùi) trong chế độ 0, 1 và 2 được quy định bởi trạng thái của bit SM57.3 như sau:

SM57.3 = 0: đếm lùi theo sườn lên của I0.6

1: đến tiến theo sườn lên của I0.6

và trong chế độ 3, 4, 5 bởi đầu vào của I0.7:

I1.3 = 0: đếm lùi theo sườn lên của I0.6

1: đến tiến theo sườn lên của I0.6

Tần số biến đổi trạng thái cực đại cho phép của tín hiệu đầu vào tại cổng I1.2 và I1.3 là 7KHz.

HSC2 có hai tần số đếm. Trong những chế độ 0 →8, tần số đếm của HSC2 bằng tần số thay đổi trạng thái của ngõ vào do đó tần số đếm cực đại là 7KHz, riêng chế độ 9,10 và 11, tùy theo khai báo sử dụng mà tần số này có thể bằng hoặc gấp 4 lần tần số biến thiên trạng thái kết quả của phép tính XOR giữa I1.2 và I1.3. Bởi vậy trong chế độ 9, 10, 11 tần số đếm cực đại cho phép của HSC2 sẽ là 28KHz. Hệ thống sử dụng bit SM57.2 để định nghĩa tần số đếm của HSC2 trong chế độ 9,10,11 là:

SM57.2 = 0: tần số đếm gấp 4 lần tần số thay đổi trạng thái I1.2 XOR I1.3

1: tần số đếm bằng tần số thay đổi trạng thái I1.2 XOR I1.3

Trong các chế độ 1, 2, 4, 5, 7, 8, 10 và 11, HSC2 sử dụng tín hiệu reset từ ngòai là I1.4. Bit SM57.0 được hệ thống sử dụng nhằm định nghĩa kiểu reset của HSC2 cho các chế độ này:

SM57.0 = 0: reset khi I1.4 có giá trị logic 1

1: reset khi I1.4 có giá trị logic 0

Cũng tương tự như vậy cho các chế độ 2, 5, 8 và 11, khi HSC2 sử dụng tín hiệu kích từ ngoài cổng I1.5, hệ thống sử dụng bit SM57.1 để định nghĩa kiểu kích HSC2

như sau:

SM57.1 = 0: HSC2 làm việc khi khi I1.5 có giá trị logic 1

1: HSC2 làm việc khi khi I1.5 có giá trị logic 0

HSC2 sử dụng byte SMB57 để xác định kiểu hoạt động như đếm tiến, lùi, cho phép hay không cho phép thay đổi giá trị đếm tức thời CV cũng như giá trị đặt trước PV, cho phép hay không cho phép kích bộ đếm. Kiểu hoạt động của HSC2 phải được định nghĩa trong SMB57 trước khi thực hiện lệnh HDEF.

Cấu trúc byte SMB57 như sau:

SM57.0	Kiểu reset cho tín hiệu xóa tại cổng I1.0
SM57.1	Kiểu start cho tín hiệu kích tại cổng I1.1
SM57.2	Tần số đếm của HSC2
SM57.3	Chiều đếm: 0 đếm lùi, 1 : đếm lên
SM57.4	Cho phép đổi chiều đếm, 0: không cho phép, 1: cho phép
SM57.5	Cho phép sửa đổi giá trị đặt trước, 0: không cho phép, 1: cho phép
SM57.6	Cho phép sửa đổi giá trị đếm tức thời, 0: không cho phép, 1: cho phép
SM57.7	1- cho phép kích HSC2, 0 – cho phép hủy HSC2

Các bit SM57.0, SM57.1 và SM57.2 được gán cho bộ đếm bằng lệnh HDEF, các bit còn lại gán cho bộ đếm sau lệnh HSC.

Các bước khai báo sử dụng HSC2 (nên thực hiện trong vòng quét đầu tiên):

- Nạp giá trị điều khiển phù hợp cho SMB57
- Xác định chế độ làm việc cho bộ đếm bằng lệnh HDEF
- Nạp giá trị đếm tức thời và giá trị đặt trước
- Khai báo sử dụng chế độ ngắt vào/ra và kích tín hiệu báo ngắt HSC2 bằng lệnh ATCH.
- Kích bộ đếm bằng lệnh HSC.

Khi sử dụng HSC2 cùng với chế độ ngắt vào/ra, các tín hiệu báo ngắt sau đây sẽ được phát:

- Báo ngắt khi CV = PV (ngắt 13)
- Báo ngắt khi có thay đổi chiều đếm từ I1.3 (ngắt 14)
- Báo ngắt khi HSC2 bị reset bởi I1.4 (ngắt 15)

sau khi được kích hoạt bằng lệnh HSC, bộ đếm HSC2 sử dụng byte SMB56 để báo trạng thái hoạt động của nó như sau:

SM56.0	Không sử dụng
SM56.1	Không sử dụng
SM56.2	Không sử dụng
SM56.3	Không sử dụng
SM56.4	Không sử dụng
SM56.5	Chiều đang đếm, 1 – đếm lên, 0 – đếm xuống.
SM56.6	Kết quả so sánh tức thời, 0 – nếu CV PV, 1 – nếu CV = PV
SM56.7	Kết quả so sánh tức thời, 0 – nếu CV PV, 1 – nếu CV > PV

b) Thủ tục khai báo sử dụng bộ đếm tốc độ cao:

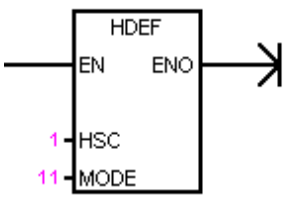
Khai báo sử dụng các bộ đếm HSC0, HSC1, HSC2 nên thực hiện tại vòng quét đầu tiên, khi mà bit SM0.1 có giá trị là 1. Thủ tục tốt nhất nên đực lập thành một chương trình con và chương trình con đó được gọi trong vòng quét đầu tiên bằng lệnh CALL.

Các công việc của chương trình con khai báo sử dụng bộ đếm tốc độ cao bao gồm:

- Nạp giá trị về kiểu hoạt động phù hợp cho byte điều khiển.
- Xác định chế độ làm việc cho bộ đếm bằng lệnh HDEF.
- Nạp giá trị đếm tức thời ban đầu và giá trị đặt trước.
- Khai báo sử dụng chế độ ngắt vào ra và kích tín hiệu báo ngắt.
- Kích bộ đếm với kiểu làm việc đã ghi trong byte điều khiển bằng lệnh HSC.

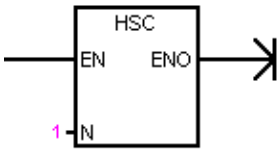
Các lệnh sử dụng như sau:

- Lệnh HDEF



Lệnh này xác định chế độ làm việc cho bộ đếm tốc độ cao. Tên của bộ đếm được xác định bằng toán hạng HSC. Chế độ làm việc được chọn là nội dung của toán hạng MODE.

- Lệnh HSC

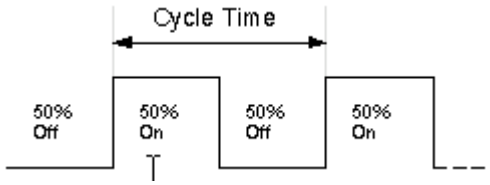


Lệnh này đặt kiểu làm việc cho bộ đếm tốc độ cao. Tên của bộ đếm được chỉ định bằng toán hạng N. Kiểu làm việc là nội dung của byte điều khiển bộ đếm.

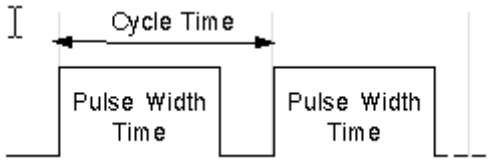
Bài 6. SỬ DỤNG HÀM PHÁT XUNG TỐC ĐỘ CAO

CPU 214 sử dụng hai cổng Q0.0 và Q0.1 để phát dãy xung tần số cao hoặc tín hiệu điều xung theo độ rộng.

PTO (Pulse Train Output) là một dãy xung vuông tuần hoàn có chu kỳ là một số nguyên trong khoảng 250 s 65.535 s hoặc từ 2 s 65.535 s. Độ rộng mỗi xung bằng ½ chu kỳ của mỗi dãy. Số xung phát cho phép nhiều nhất của dãy phát xung là 4.294.967.295 xung.



PWM (Pulse Width Modulation) là một dãy xung tuần hoàn có chu kỳ là một số nguyên trong khoảng 250 s 65.535 s hoặc từ 2 s 65.535 s. Khác với PTO, độ rộng của mỗi xung có thể quy định được là một số nguyên nằm trong khoảng 0 s 65.535 s hoặc 0ms 65.535ms. Nếu độ rộng xung được quy định lớn hơn chu kỳ của PWM thì một tín hiệu có giá trị logic bằng 1, ngược lại sẽ có một tín hiệu logic 0.



PTO và PWM được hệ thống phát ra theo cổng Q0.0 hoặc Q0.1. Các nguồn phát PTO và PWM sử dụng:

- Một byte điều khiển 8 bit
- Một từ đơn ghi chu kỳ xung
- Một từ kép ghi số xung của dãy.

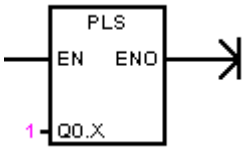
Địa chỉ những ô nhớ này như sau:

Cổng ra	Byte điều khiển	Chu kỳ	Độ rộng xung	Số xung
Q0.0	SMB67	SMW68	SMW70	SMD72
Q0.1	SMB77	SMW78	SMW80	SMD82

Các ô nhớ này phải được nạp những giá trị thích hợp trước khi thực hiện lệnh tạo xung. Lệnh PLS sẽ tạo ra một dãy xung có kiểu được quy định trong byte điều khiển với chu kỳ, độ rộng xung và số xung đã được đặt trước.

Lệnh PLS lệnh phát xung tại cổng Q0.0 hoặc Q0.1 theo cấu trúc được định nghĩa trong byte điều khiển và các ô nhớ về chu kỳ độ rộng. Cổng xung phát ra được chỉ định trong toán hạng x (0 cho Q0.0 và 1 cho Q0.1) của lệnh.

Cú pháp sử dụng lệnh PLS như sau:



Lệnh PLS sử dụng các byte điều khiển SMB67 (cho cổng Q0.0) và SMB77 (cho cổng Q0.1) theo cấu trúc như sau:

Q0.0	Q0.1	Mục đích
SM67.0	SM77.0	Đổi chu kỳ: 1 – cho phép, 0 – không cho phép
SM67.1	SM77.1	Đổi độ rộng xung: 1 – cho phép, 0 – không cho phép
SM67.2	SM77.2	Đổi số đếm cho xung PTO: 1 – cho phép, 0 – không cho phép
SM67.3	SM77.3	Đơn vị thời gian: 1 – ms, 0 - s
SM67.4	SM77.4	Không sử dụng
SM67.5	SM77.5	Không sử dụng
SM67.6	SM77.6	Chọn kiểu xung: 0 – PTO, 1 – PWM
SM67.7	SM77.7	Khai báo: 1 – kích, 0 – hủy.

Nếu như các chế độ ngắt kiểu 19 hoặc 20 được khai báo cùng với việc tạo xung thì đối với dãy xung PTO, tín hiệu báo ngắt PLS0 hoặc PLS1 sẽ xuất hiện tại thời điểm sườn xuống của xung cuối cùng (trong nửa sau của chu kỳ cuối), còn đối với dãy xung PWM tín hiệu báo ngắt xuất hiện tại thời điểm sườn lên của xung cuối cùng.

Muốn tạo những xung có nhiều dạng khác nhau thì sử dụng chương trình ngắt tương ứng để khai báo dạng xung mới.

Thủ tục khai báo sử dụng hàm phát xung tốc độ cao tốt nhất là nên được thực hiện trong vòng quét đầu và bao gồm các bước sau:

- Ghi giá trị cho byte điều khiển
- Nạp các giá trị về chu kỳ (độ rộng xung) và số xung của dãy vào những ô nhớ tương ứng
- Khai báo sử dụng chế độ ngắt (nếu cần sử dụng)
- Thực hiện lệnh PLS

Xung sẽ được phát ngay bằng sườn lên của xung đầu tiên sau khi thực hiện lệnh PLS.