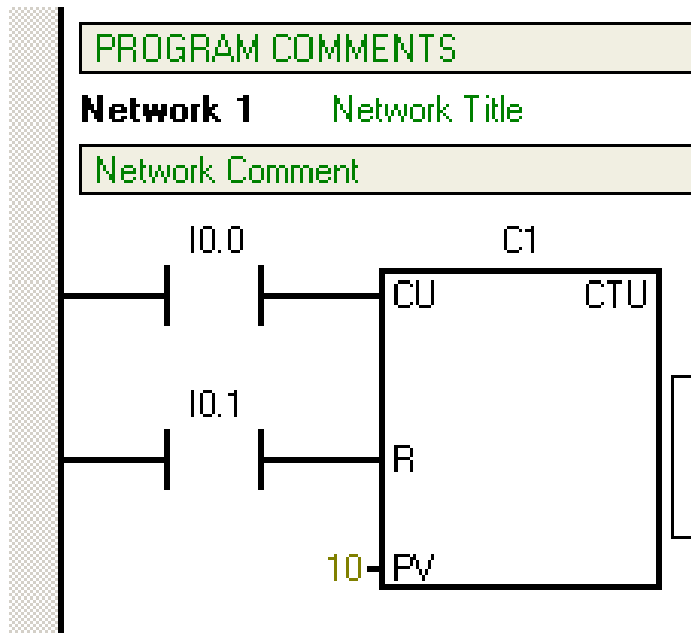


tập lệnh counter

Đếm lên



Cxxx Constant (C0 – C255) WORD

CU (LAD) Power Flow BOOL

CU (FBD) I, Q, M, SM, T, C, V, S, L, Power Flow BOOL

R (LAD) Power Flow BOOL

R (FBD) I, Q, M, SM, T, C, V, S, L, Power Flow BOOL

PV VW, IW, QW, MW, SMW, LW, AIW, AC, T, C, Constant, *VD, *AC, *LD, SW INT

Khi có cạnh lên ở ngõ vào CU, nội dung bộ đếm tăng lên 1. Khi nội dung lớn hơn PV thì Counter ON.

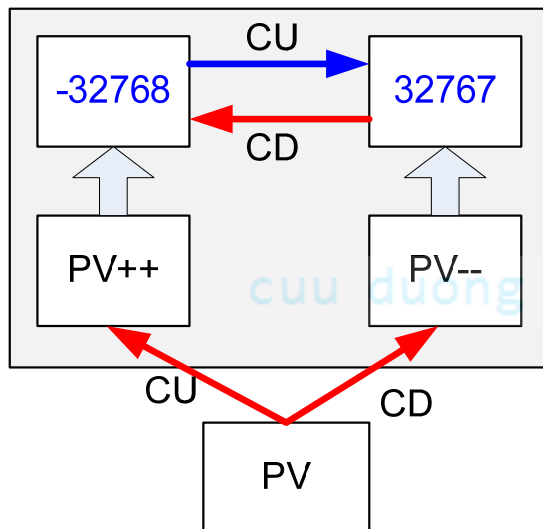
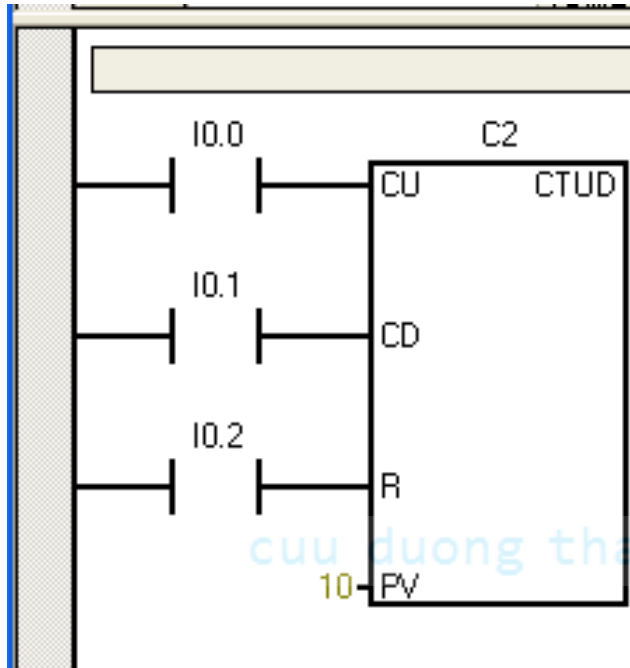
Khi ngõ vào R (reset) tác động, nội dung của bộ đếm sẽ về 0.

Nội dung bộ đếm sẽ tăng đến max (32767) nếu không có reset.

Counter: 0-255

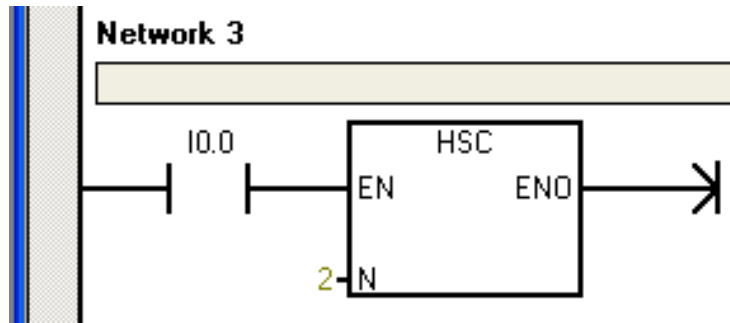
Chú ý: không dung chung 1 counter với các chức năng khác nhau CTU, CTD, CTUD trong một chương trình.

Đếm lên, xuống



Nội dung bộ đếm sẽ được giữ nguyên khi đạt PV

Lệnh đếm tốc độ cao (High Speed Counter – HSC)



N = 1, 2, 3, 4, 5

HSC có thể hoạt động ở 12 chế độ khác nhau.

- CPU 221 and CPU 222 support 4 High-Speed Counters (HSC0, HSC3, HSC4, and HSC5)
- CPU 221 and CPU 222 do not support HSC1 and HSC2.
- CPU 224, CPU224XP, and CPU 226 support 6 High-Speed Counters (HSC0 to HSC5)

Thiết lập hoạt động cho HSC bằng HDEF (High speed counter definition)

Các bước thiết lập hoạt động cho HSC

- Định nghĩa counter và chế độ hoạt động
- Thiết lập byte điều khiển
- thiết lập giá trị ban đầu của counter (starting value)
- thiết lập giá trị cuối của counter (preset value)
- Đăng ký ngắt và cho phép ngắt hoạt động
- Kích hoạt hoạt động của counter

Thiết lập ngõ vào cho HSC

Mode	Description	Inputs			
	HSC0	I0.0	I0.1	I0.2	
	HSC1	I0.6	I0.7	I1.0	I1.1
	HSC2	I1.2	I1.3	I1.4	I1.5
	HSC3	I0.1			
	HSC4	I0.3	I0.4	I0.5	
	HSC5	I0.4			
0	Single-phase counter with internal direction control	Clock			
1		Clock		Reset	
2		Clock		Reset	Start
3	Single-phase counter with external direction control	Clock	Direction		
4		Clock	Direction	Reset	
5		Clock	Direction	Reset	Start
6	Two-phase counter with 2 clock inputs	Clock Up	Clock Down		
7		Clock Up	Clock Down	Reset	
8		Clock Up	Clock Down	Reset	Start
9	A/B phase quadrature counter	Clock A	Clock B		
10		Clock A	Clock B	Reset	
11		Clock A	Clock B	Reset	Start
12	Only HSC0 and HSC3 support mode12. HSC0 counts the number of pulses going out of Q0.0. HSC3 counts the number of pulses going out of Q0.1.				

cuu duong than cong . com

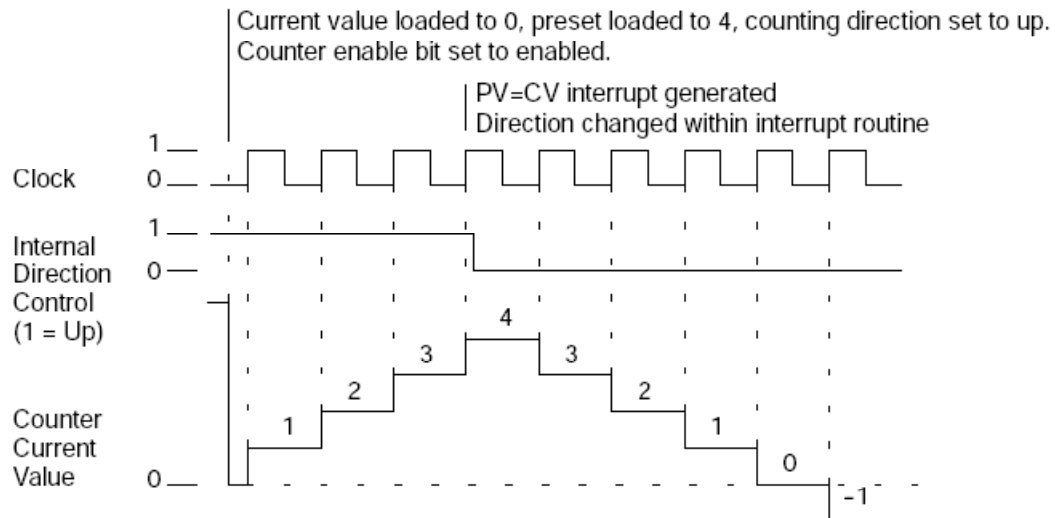


Figure 6-22 Operation Example of Modes 0, 1, or 2

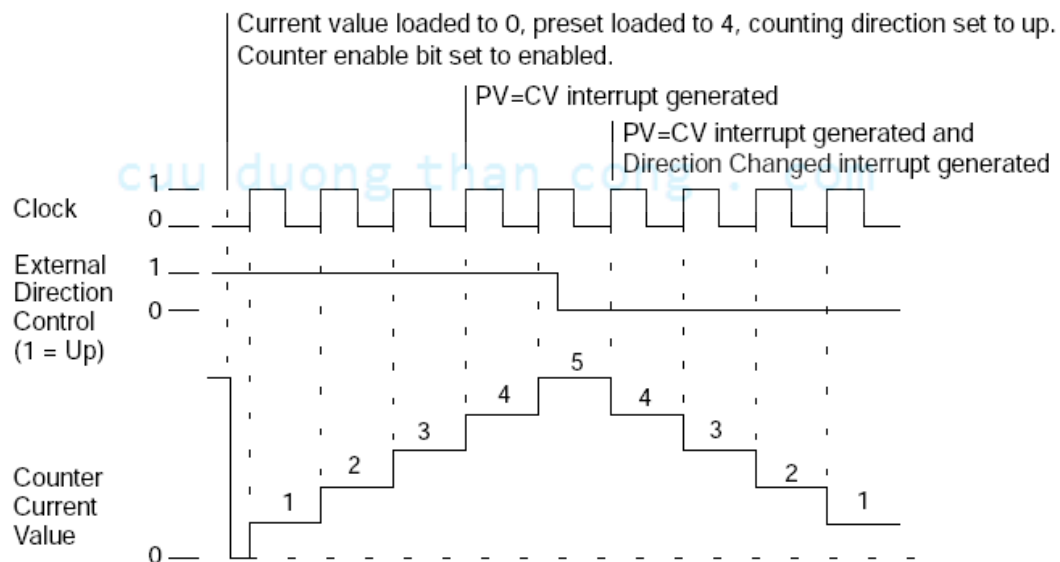


Figure 6-23 Operation Example of Modes 3, 4, or 5

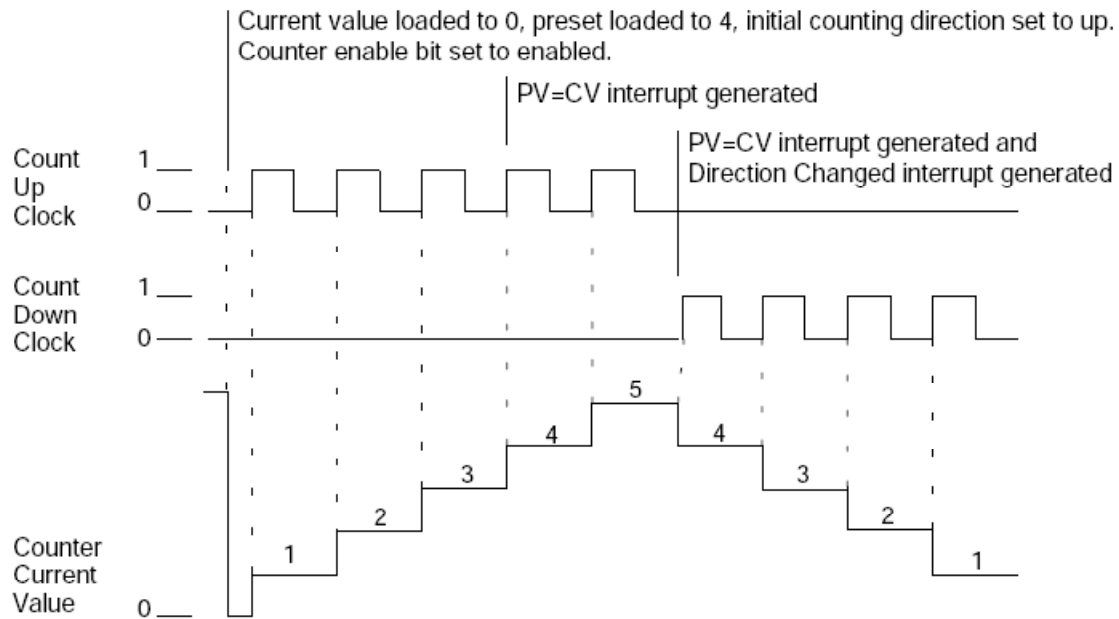


Figure 6-24 Operation Example of Modes 6, 7, or 8

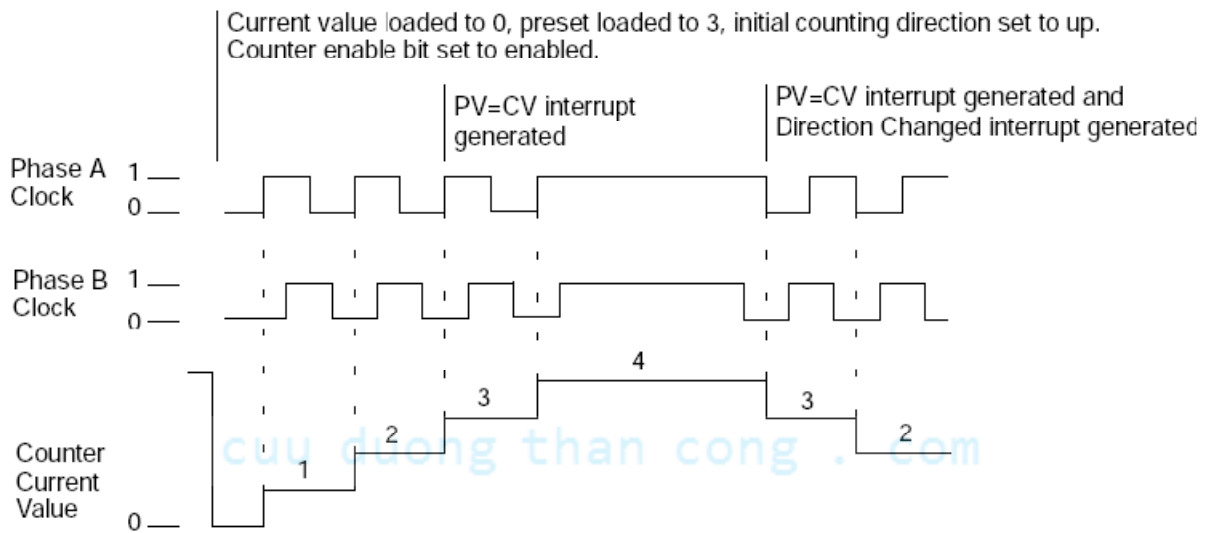


Figure 6-25 Operation Example of Modes 9, 10, or 11 (Quadrature 1x Mode)

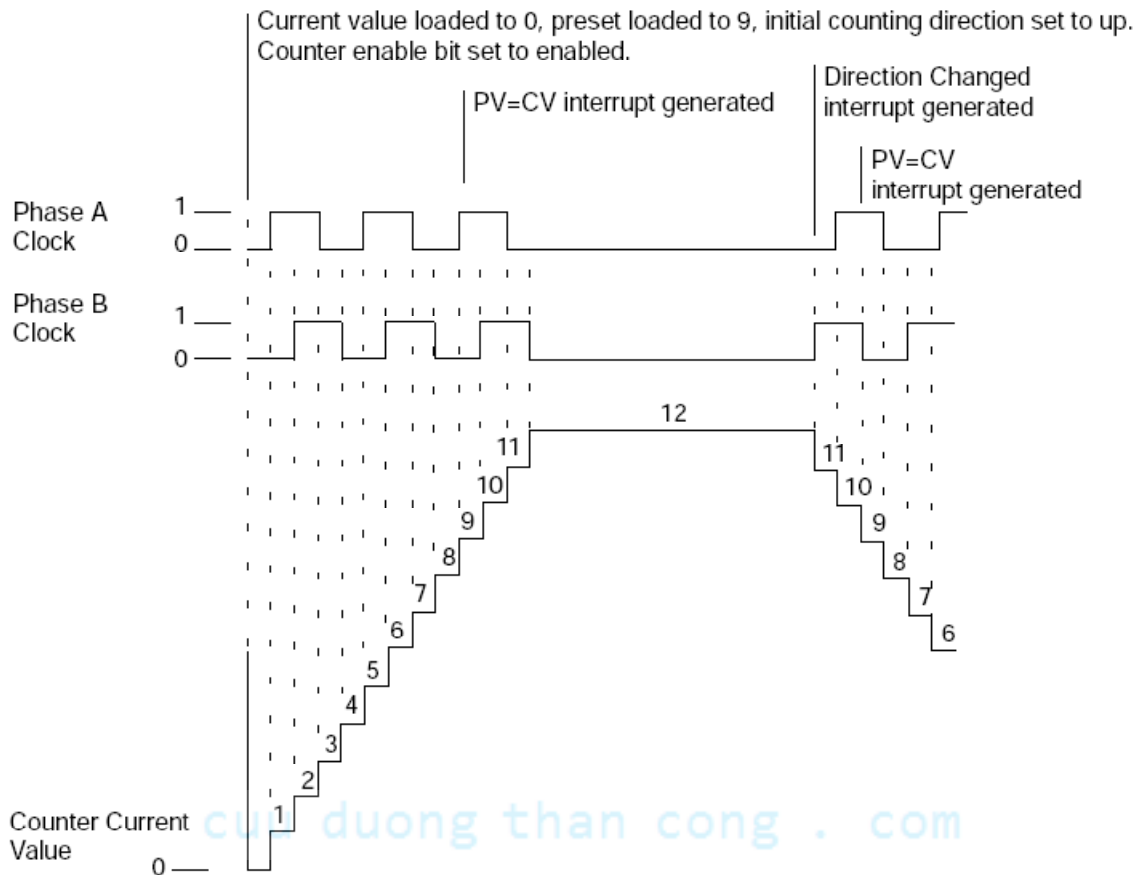


Figure 6-26 Operation Example of Modes 9, 10, or 11 (Quadrature 4x Mode)

Hoạt động start và reset

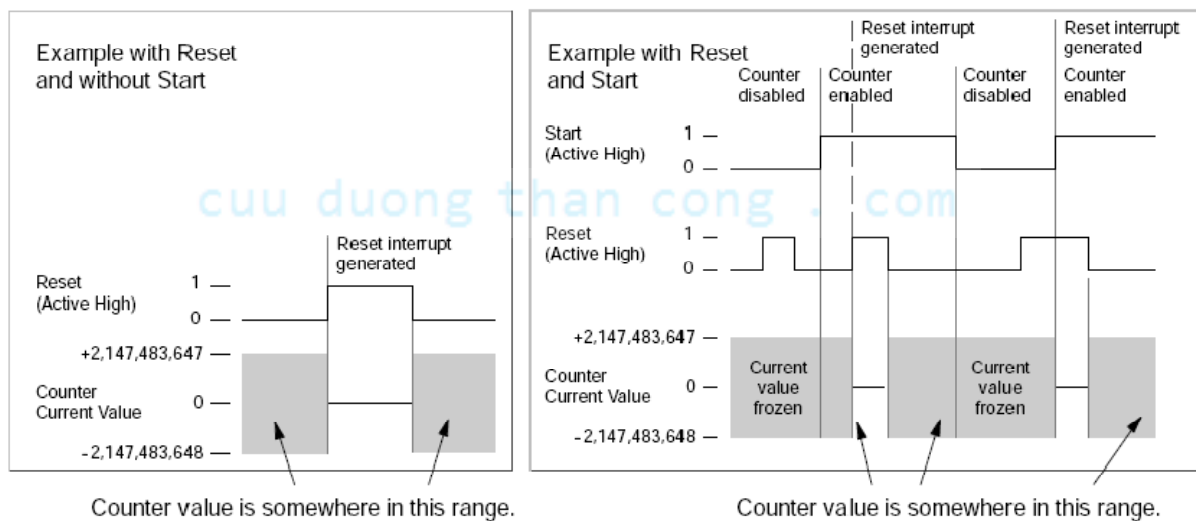


Figure 6-27 Operation Examples Using Reset with and without Start

Table 6-27 Active Level for Reset, Start, and 1x/4x Control Bits

HSC0	HSC1	HSC2	HSC4	Description (used only when HDEF is executed)
SM37.0	SM47.0	SM57.0	SM147.0	Active level control bit for Reset ¹ : 0 = Reset is active high 1 = Reset is active low
---	SM47.1	SM57.1	---	Active level control bit for Start ¹ : 0 = Start is active high 1 = Start is active low
SM37.2	SM47.2	SM57.2	SM147.2	Counting rate selection for quadrature counters: 0 = 4X counting rate 1 = 1X counting rate

¹ The default setting of the reset input and the start input are active high, and the quadrature counting rate is 4x (or four times the input clock frequency).

Tip

You must set these three control bits to the desired state before the HDEF instruction is executed. Otherwise, the counter takes on the default configuration for the counter mode selected.

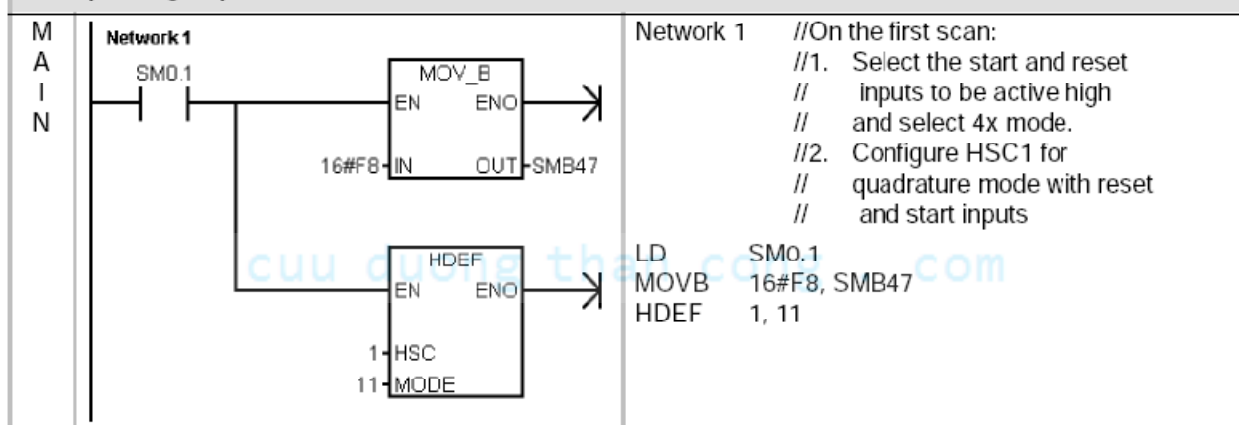
Once the HDEF instruction has been executed, you cannot change the counter setup unless you first place the S7-200 in STOP mode.

Chú ý

Phải thiết lập 3 bit điều khiển trước khi thực thi lệnh HDEF. Nếu không counter sẽ hoạt động ở chế độ mặc định.

Khi lệnh HDEF được thực thi, ta không thể thiết lập lại cho counter trừ khi PLC vào chế độ stop.

Example: High-Speed Counter Definition Instruction



Thiết lập byte điều khiển

Sau khi định nghĩa counter và chế độ của counter, ta có thể thiết lập các thông số động của counter, mỗi counter có thể điều khiển các hoạt động sau:

- Cho phép hoặc vô hiệu hoá
- Thay đổi chiều (chế độ 1, 2, 3), hay chiều ban đầu của các chế độ còn lại
- Lấy giá trị đầu
- Lấy giá trị cuối

Table 6-28 Control Bits for HSC0, HSC1, HSC2, HSC3, HSC4, and HSC5

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	Description
SM37.3	SM47.3	SM57.3	SM137.3	SM147.3	SM157.3	Counting direction control bit: 0 = Count down 1 = Count up
SM37.4	SM47.4	SM57.4	SM137.4	SM147.4	SM157.4	Write the counting direction to the HSC: 0 = No update 1 = Update direction
SM37.5	SM47.5	SM57.5	SM137.5	SM147.5	SM157.5	Write the new preset value to the HSC: 0 = No update 1 = Update preset
SM37.6	SM47.6	SM57.6	SM137.6	SM147.6	SM157.6	Write the new current value to the HSC: 0 = No update 1 = Update current value
SM37.7	SM47.7	SM57.7	SM137.7	SM147.7	SM157.7	Enable the HSC: 0 = Disable the HSC 1 = Enable the HSC

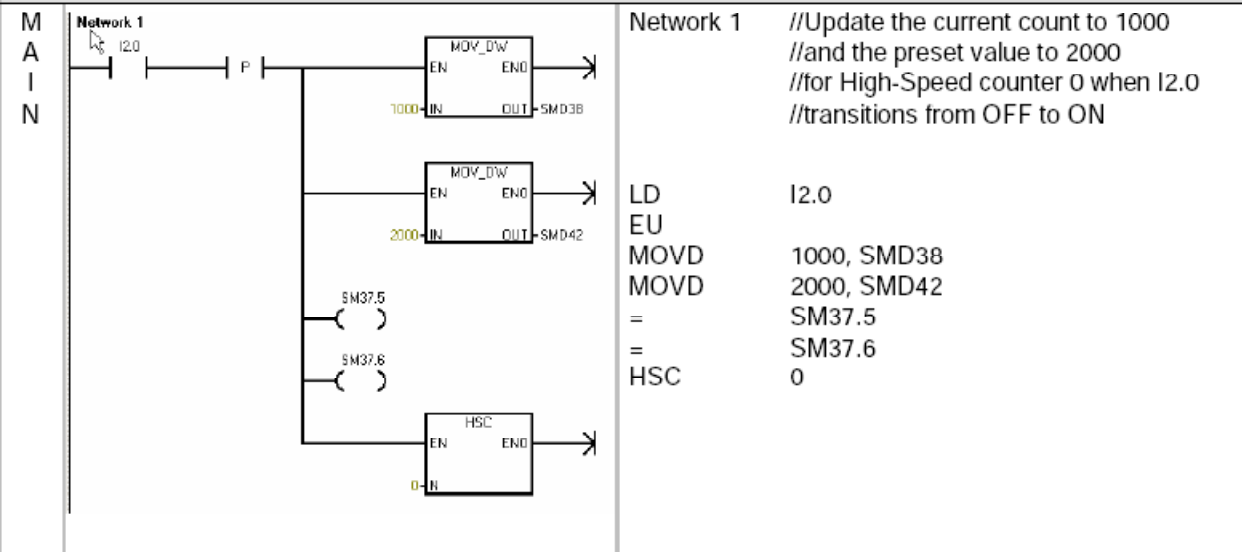
Example: Reading and Saving the Current Count

M A I N			<p>Network 1 //Save the value of //High Speed Counter 0 //into VD200 when I3.0 //transitions from OFF to ON.</p>
	<p>LD I3.0 EU MOVD HC0, VD200</p>		

Tip

Changes to the control byte and the SM locations for new current value and new preset value will not affect the high-speed counter until the corresponding HSC instruction is executed.

Example: Updating the Current and Preset Values



cuu duong than cong . com

Đọc nội dung bộ đếm

Table 6-29 Current Values of HSC0, HSC1, HSC2, HSC3, HSC4, and HSC5

Value to be Read	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5
Current value (CV)	HC0	HC1	HC2	HC3	HC4	HC5

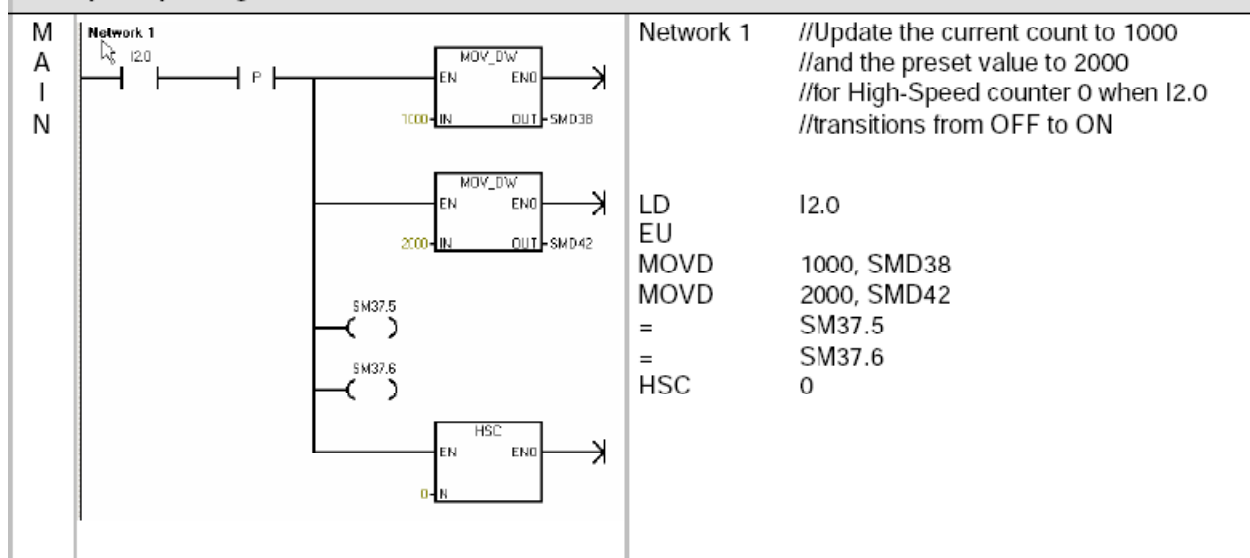
Nạp giá trị cho bộ đếm

Table 6-30 New Current and New Preset Values of HSC0, HSC1, HSC2, HSC3, HSC4, and HSC5

Value to be Loaded	HSC0	HSC1	HSC2	HSC3	HSC4	HSC5
New current value (new CV)	SMD38	SMD48	SMD58	SMD138	SMD148	SMD158
New preset value (new PV)	SMD42	SMD52	SMD62	SMD142	SMD152	SMD162

Những thay đổi của byte điều khiển và nội dung bộ đếm không tác động vào HSC cho đến khi lệnh HSC được thực thi.

Example: Updating the Current and Preset Values



Assigning Interrupts

All counter modes support an interrupt event when the current value of the HSC is equal to the loaded preset value. Counter modes that use an external reset input support an interrupt on activation of the external reset. All counter modes except modes 0, 1, and 2 support an interrupt on a change in counting direction. Each of these interrupt conditions can be enabled or disabled separately. For a complete discussion on the use of interrupts, see the section on Communications and Interrupt instructions.

Notice

A fatal error can occur if you attempt either to load a new current value or to disable and then re-enable the high-speed counter from within the external reset interrupt routine.

Status Byte

A status byte for each high-speed counter provides status memory bits that indicate the current counting direction and whether the current value is greater or equal to the preset value. Table 6-31 defines these status bits for each high-speed counter.

Thiết lập Ngắt

tất cả các chế độ của counter đều hỗ trợ ngắt sự kiện khi giá trị hiện thời của HSC bằng giá trị PV. Các chế độ counter sử dụng ngõ vào reset ngoài hỗ trợ ngắt bằng cách kích hoạt ngõ reset ngoài. Tất cả các chế độ của counter trừ các mode 0, 1, 2 đều hỗ trợ ngắt đảo chiều. Các điều kiện ngắt có thể được cho phép hoặc cấm một cách riêng biệt.

Byte trạng thái;

Byte trạng thái cho mỗi HSC cho biết các bit trạng thái xác định chiều quay cũng như giá trị hiện thời có lớn hơn hoặc bằng PV.

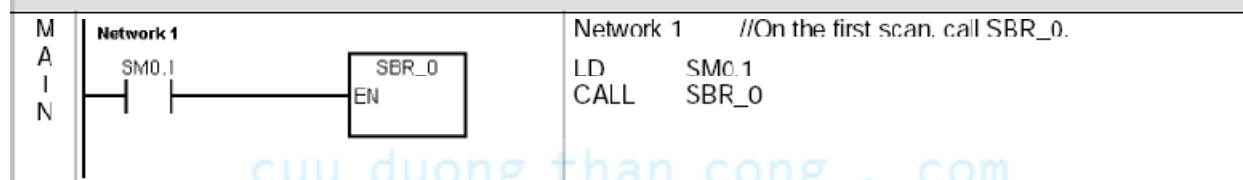
Tip

Status bits are valid only while the high-speed counter interrupt routine is being executed. The purpose of monitoring the state of the high-speed counter is to enable interrupts for the events that are of consequence to the operation being performed.

Table 6-31 Status Bits for HSC0, HSC1, HSC2, HSC3, HSC4, and HSC5

HSC0	HSC1	HSC2	HSC3	HSC4	HSC5	Description
SM36.0	SM46.0	SM56.0	SM136.0	SM146.0	SM156.0	Not used
SM36.1	SM46.1	SM56.1	SM136.1	SM146.1	SM156.1	Not used
SM36.2	SM46.2	SM56.2	SM136.2	SM146.2	SM156.2	Not used
SM36.3	SM46.3	SM56.3	SM136.3	SM146.3	SM156.3	Not used
SM36.4	SM46.4	SM56.4	SM136.4	SM146.4	SM156.4	Not used
SM36.5	SM46.5	SM56.5	SM136.5	SM146.5	SM156.5	Current counting direction status bit: 0 = Counting down 1 = Counting up
SM36.6	SM46.6	SM56.6	SM136.6	SM146.6	SM156.6	Current value equals preset value status bit: 0 = Not equal 1 = Equal
SM36.7	SM46.7	SM56.7	SM136.7	SM146.7	SM156.7	Current value greater than preset value status bit: 0 = Less than or equal 1 = Greater than

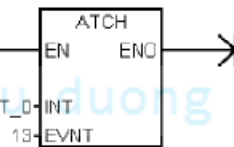
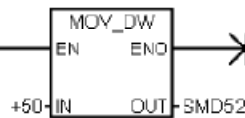
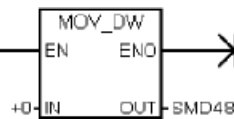
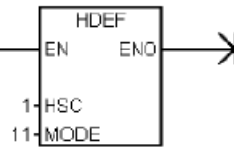
Example: High-Speed Counter Instruction



S
B
R
0

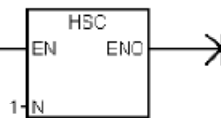
Network 1

SM0.1



cuu duong than cong . com

(ENI)



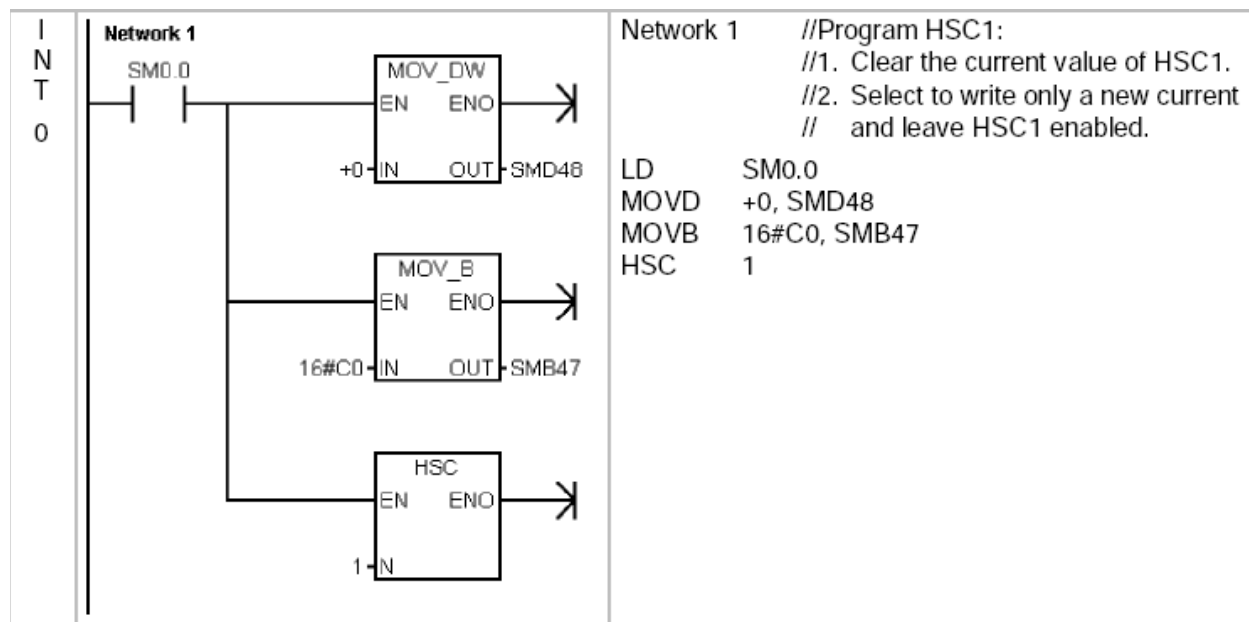
Network 1

//On the first scan, configure HSC1:

//1. Enable the counter.
// - Write a new current value.
// - Write a new preset value.
// - Set the initial direction to count up.
// - Select the start and reset inputs
// to be active high.
// - Select 4x mode.
//2. Configure HSC1 for quadrature mode
// with reset and start inputs.
//3. Clear the current value of HSC1.
//4. Set the HSC1 preset value to 50.
//5. When HSC1 current value = preset value,
// attach event 13 to interrupt routine INT_0.
//6. Global interrupt enable.
//7. Program HSC1.

LD SM0.1
MOVB 16#F8, SMB47
HDEF 1, 11
MOVD +0, SMD48
MOVD +50, SMD52
ATCH INT_0, 13
ENI
HSC 1

cuu duong than cong . com

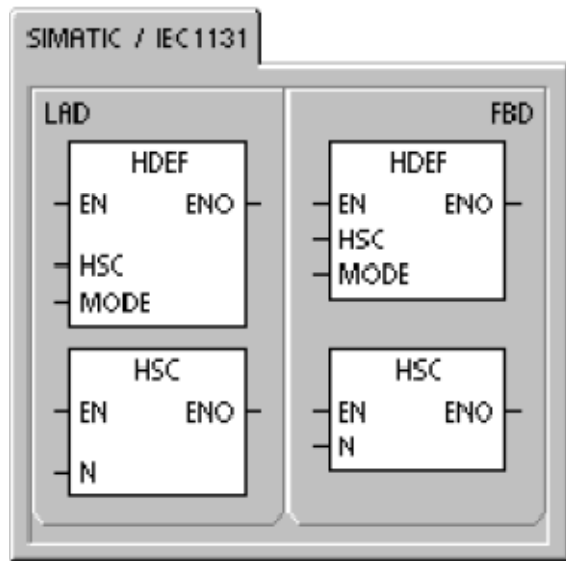


Độc tài liệu s7-200 manual trang 127 – 131

cuu duong than cong . com

cuu duong than cong . com

High-speed counter definition (HDEF) Định chế độ hoạt động cho HSC



HSC Constant (0,1,2,3,4, or 5) **BYTE**

MODE Constant (0,1,2,3,4,5,6,7,8,9,10, or 11) **BYTE**

Tip

CPU 221 and CPU 222 support four high-speed counters: HSC0, HSC3, HSC4, and HSC5. These CPUs do not support HSC1 and HSC2.

CPU 224, CPU 224XP, and CPU 226 support six high-speed counters: HSC0 to HSC5.

NETWORK 1

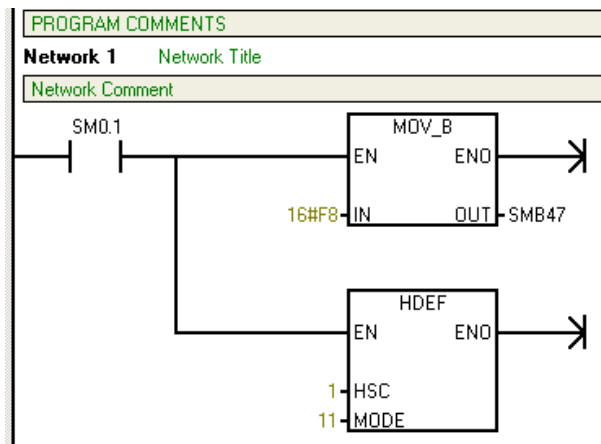
LD SM0.1 // On the first scan

MOVB 16#F8 SMB47 // Select the start and reset inputs to be active high

// Select 4x mode

HDEF 1 11 // HSC1 configured for quadrature mode

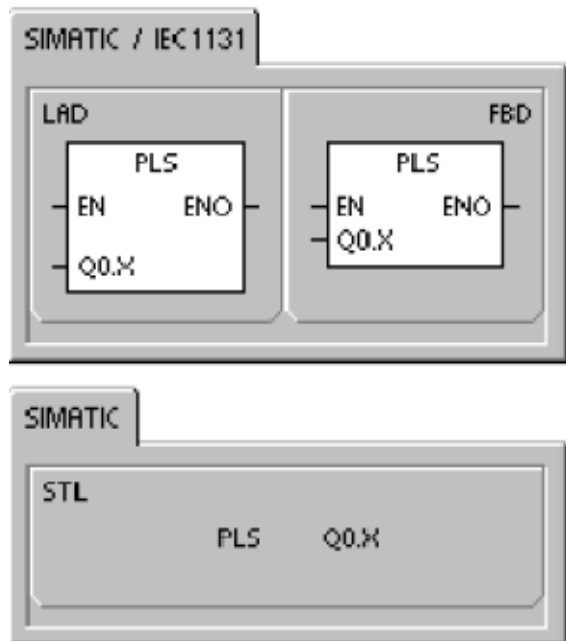
// with reset and start inputs



cuu duong than cong . com

cuu duong than cong . com

PULSE OUTPUT INSTRUCTION (PLS)



PLS được sử dụng để tạo các chức năng PTO (Pulse Train Output) và PWM (Pulse Width Modulation) tốc độ cao ở các ngõ ra Q0.0 và Q0.1

PTO cung cấp ngõ ra xung vuông 50% với chu kỳ khác nhau và số lượng xung mong muốn.

PWM cung cấp xung ngõ ra có độ rộng thay đổi và chu kỳ thay đổi.

S7-200 có 2 bộ PWM/PTO để tạo xung PWM hay PTO. Một bộ tạo xung cho Q0.0 và một bộ tạo xung cho Q0.1.

Các ô nhớ SM được sử dụng để lưu trữ dữ liệu cho các bộ tạo xung, bao gồm: byte điều khiển 8bit, bộ đếm xung 32 bit không dấu, chu kỳ và độ rộng xung 16 bit không dấu.

Table 6-32 Valid Operands for Pulse Output Instruction

Inputs/Outputs	Data Types	Operands
Q0.X	WORD	Constant: 0 (= Q0.0) or 1 (= Q0.1)

Tạo xung tần số cao và điều rộng xung

Lệnh PLS phát dãy xung vuông tần số thay đổi, số xung thay đổi PTO hay dãy xung điều rộng PWM có chu kỳ thay đổi ở ngõ ra Q0.0 hay Q0.1. Dùng SMB67/SMB77 cài đặt cấu hình phát xung

Q0.0	Q0.1	Control Bits	
SM67.0	SM77.0	PTO/PWM update cycle time value	0 = no update; 1 = update cycle time
SM67.1	SM77.1	PWM update pulse width time value	0 = no update; 1 = update pulse width
SM67.2	SM77.2	PTO update pulse count value	0 = no update; 1 = update pulse count
SM67.3	SM77.3	PTO/PWM time base select	0 = 1 μ s/tick; 1 = 1ms/tick
SM67.4	SM77.4	PWM update method:	0 = asynchronous update; 1 = synchronous update
SM67.5	SM77.5	PTO operation:	0 = single segment operation; 1 = multiple segment operation
SM67.6	SM77.6	PTO/PWM mode select	0 = selects PTO; 1 = selects PWM
SM67.7	SM77.7	PTO/PWM enable	0 = disables PTO/PWM; 1 = enables PTO/PWM

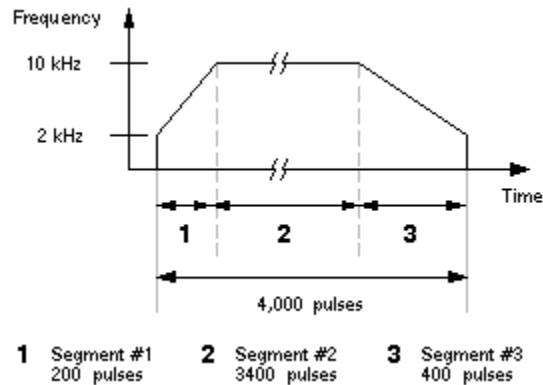
Dùng các ô nhớ 16 bit chứa chu kỳ và bề rộng xung, ô nhớ 32 bit chứa số lượng xung. Trước khi dùng lệnh PLS phải xóa Q0.0 và Q0.1, lệnh PLS còn được dùng mỗi khi thay đổi thông số dãy xung.

Q0.0	Q0.1	Other PTO/PWM Registers
SMV68	SMV78	PTO/PWM cycle time value (range: 2 to 65535)
SMV70	SMV80	PWM pulse width value (range: 0 to 65535)
SMD72	SMD82	PTO pulse count value (range: 1 to 4294967295)
SMB166	SMB176	Number of segment in progress (used only in multiple segment PTO operation)
SMV168	SMV178	Starting location of profile table, expressed as a byte offset from V0 (used only in multiple segment PTO operation)
SMB170	SMB180	Linear profile status byte
SMB171	SMB181	Linear profile result register
SMB172	SMB182	Manual mode frequency register

Mode PWM tạo xung có chu kỳ từ 10 microsec đến 65535 microsec, 2 ms đến 65535 ms, bề rộng xung từ 0 microsec (msec) đến 65535 microsec (msec), Sự thay đổi đặc tính xung thực hiện đồng bộ ở cuối mỗi chu kỳ xung, trong trường hợp thay đổi cả đơn vị thời gian thì thực hiện thay đổi ngay tức khắc gọi là cập nhật bất đồng bộ

Mode PTO (pulse train output) phát xung vuông có chu kỳ thay đổi 10 microsec đến 65535 microsec, 2 ms đến 65535 ms, số lượng xung từ 1 đến 4.294.967.295. Trong suốt dãy xung, chu kỳ có thể thay đổi hay giữ nguyên. Ở chế độ single segment pipelining, giả sử chia thời gian thành nhiều khoảng, ta qui định số lượng xung và chu kỳ, chứa vào ô nhớ SMV68/78 và SMD72/82, sau đó thực hiện lệnh PLS; sử dụng ngắt khi đã phát đủ số lượng xung để thực hiện lệnh PLS với số lượng xung và chu kỳ mới.

Trường hợp multisegment pipelining, ta lập trình trước biểu đồ thay đổi trong một bảng gọi là profile table. Ví dụ sau điều khiển động cơ bước với biểu đồ vận tốc gồm ba đoạn tăng tốc, vận tốc không đổi và giảm tốc



Tần số xung ban đầu là 2KHz, tần số cao nhất là 10 KHz tương ứng chu kỳ là 500ms và 100ms. Trong giai đoạn tăng tốc phải phát 200 xung với chu kỳ giảm từ 500ms xuống 100ms, còn giai đoạn giảm tốc, chu kỳ tăng từ 100ms lên 500ms và phải phát 400 xung, vậy ta phải xác định

gia số chu kỳ = (chu kỳ cuối – chu kỳ đầu)/số xung là số nguyên

đoạn tăng tốc gsck là -2, đoạn giữa là 0, và đoạn cuối là 1

Các thông số của mỗi đoạn chứa trong một bảng ô nhớ, giả sử địa chỉ ban đầu là VB100

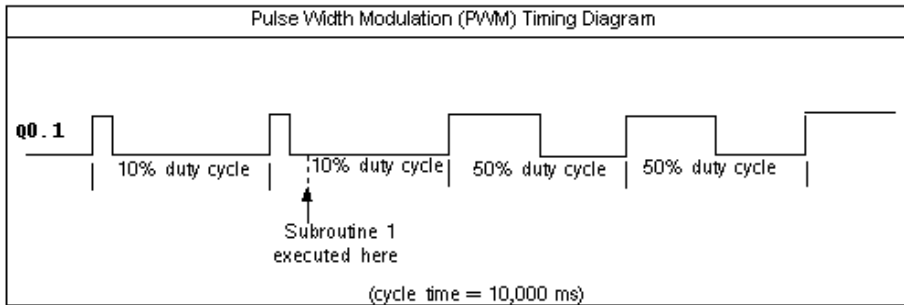
VB100	3	Số đoạn	
VW101	500	Chu kỳ đầu	Đoạn 1
VW103	-2	Gia số chu kỳ	
VD105	200	Số xung	
VW109	100	Chu kỳ đầu	Đoạn 2
VW111	0	Gia số chu kỳ	
VD113	3400	Số xung	
VW117	100	Chu kỳ đầu	Đoạn 3
VW119	1	Gia số chu kỳ	
VD121	400	Số xung	

Để điều khiển trơn, chu kỳ cuối đoạn trước nên bằng chu kỳ đầu đoạn sau

Các bit sau cho biết trạng thái của PTO

Q0.0	Q0.1	Status Bits
SM66.4	SM76.4	PTO profile aborted due to delta calculation error 0 = no error; 1 = aborted
SM66.5	SM76.5	PTO profile aborted due to user command 0 = no abort; 1 = aborted
SM66.6	SM76.6	PTO pipeline overflow/underflow 0 = no overflow; 1 = overflow/underflow
SM66.7	SM76.7	PTO idle 0 = in progress; 1 = PTO idle

Ví dụ: tạo xung PWM



// Main, Reset Q0.1 , gọi SBR_0

NETWORK 1

LD SM0.1

R Q0.1 1

CALL SBR_0

NETWORK 2 // Khi M0.0 On thay đổi bề rộng xung 50%

LD M0.0

EU

CALL SBR_1

//SBR0

NETWORK 1

LD SM0.0

MOVB 16#DB, SMB77 // Byte điều khiển, PWM, đơn vị ms, cập nhật đồng bộ

MOVW +10000, SMW78 // Chu kỳ 10,000 ms

MOVW +1000, SMW80 // Bề rộng xung 1,000 ms

PLS 1 // Phát xung ở Q0.1

MOVB 16#DA, SMB77 // Nạp byte điều khiển cho lần sau

//SBR 1

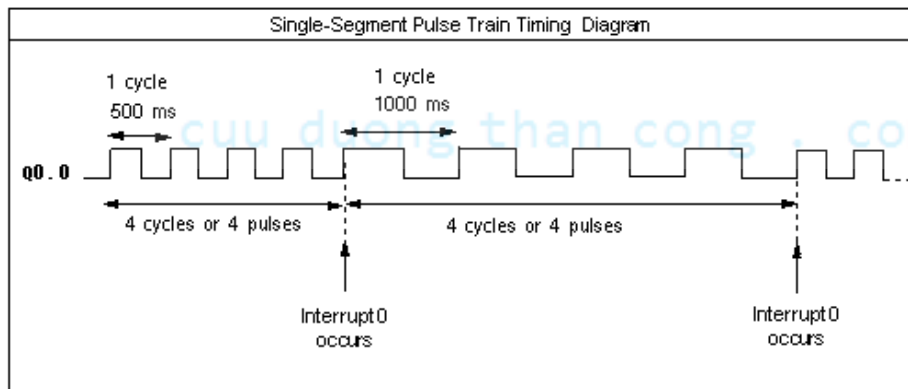
NETWORK 1

LD SM0.0

MOVW +5000, SMW80 // Bề rộng xung 5000 ms

PLS 1 // Phát xung

Ví dụ: tạo xung PTO một đoạn



// Main, phát dãy 4 xung PTO chu kỳ thay phiên 500ms rồi 1000ms

NETWORK 1

```

LD SM0.1
R Q0.0 1
CALL SBR_0
// SBR 0
NETWORK 1
LD SM0.0
MOVB 16#8D, SMB67 // Byte điều khiển, PTO, một đoạn, đơn vị ms:
MOVW +500, SMW68 // Chu kỳ 500ms.
MOVD +4, SMD72 // Phát 4 xung.
ATCH INT_0, 19 // Gọi ngắt 19 khi phát xong
ENI
PLS 0 // phát xung ở Q0.0
MOVB 16#89, SMB67 // Nạp byte điều khiển cho đoạn sau

```

// Interrupt 0

```

NETWORK 1
// Nếu chu kỳ là 500 ms: đổi chu kỳ là 1000 ms và phát 4 xung
LDW= SMW68 +500
MOVW +1000 SMW68
PLS 0
CRETI

```

NETWORK 2

```

// Nếu chu kỳ là 1000 ms: đổi chu kỳ là 500 ms và phát 4 xung
LDW= SMW68 +1000
MOVW +500 SMW68
PLS 0

```

Ví dụ: phát xung PTO nhiều đoạn

// Main

NETWORK 1

```

LD SM0.1
R Q0.0 1
CALL SBR_0

```

// Subroutine 0

NETWORK 1

```

// Nạp bản profile
LD SM0.0
MOVB 3 VB500 // Số đoạn là 3
// Segment 1:
MOVW +500 VW501 // Chu kỳ ban đầu 500 ms
MOVW -2 VW503 // Giá số chu kỳ -2 ms
MOVD +200 VD505 // Số xung 200
// Segment 2:
MOVW +100 VW509 // Chu kỳ đầu 100 ms
MOVW +0 VW511 // Giá số 0 ms
MOVD +3400 VD513 // Số đoạn 3400
// Segment 3:

```

MOVW +100 VW517 // Chu kỳ đầu 100 ms

MOVW +1 VW519 // Gia số 1 ms

MOVD +400 VD521 // Số xung là 400

NETWORK 2

LD SM0.0

MOVB 16#A8 SMB67 // Byte điều khiển, PTO, nhiều đoạn, đơn vị ms,

MOVW +500 SMW168 // Bảng bắt đầu ở VB500.

ATCH INT_0 19 // Gọi ngắt 0 khi kết thúc

ENI

PLS 0 // Phát xung ở Q0.0

MOVB 16#89 SMB67 // Nạp byte điều khiển cho lần sau

//INT0, khi phát đủ xung, cho Q0.5 on

NETWORK 1

LD SM0.0

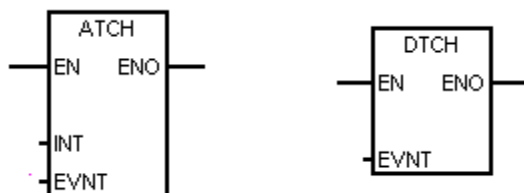
= Q0.5

cuu duong than cong . com

cuu duong than cong . com

LỆNH NGẮT

Có thể có tối đa 128 chương trình phục vụ ngắt, viết sau chương trình con, đóng khung bằng INT_n RETI. Cấm/ cho phép ngắt bằng lệnh ENI/DISI. INT_n được gọi đến khi xảy ra sự kiện EVNT. Có tất cả 27 sự kiện có thể gây ra ngắt. Sự kiện được liên kết với INT_n thông qua lệnh gắn ATCH và tháo DTCH.



Bảng 9.5: Các sự kiện ngắt và ưu tiên

Số Sự kiện	Miêu tả	Ưu tiên	Ưu tiên trong nhóm
8	Nhận ký tự ở Port 0	Cao nhất	1
9	Truyền xong Port 0		1
23	Nhận xong bản tin ở Port 0		1
24	Nhận xong bản tin ở Port 1		2
25	Nhận ký tự ở Port 1		2
26	Truyền xong Port 1		2
0	Ngắt ở cạnh lên của I0.0	Giữa	1
2	Ngắt ở cạnh lên của I0.1		2
4	Ngắt ở cạnh lên của I0.2		3
6	Ngắt ở cạnh lên của I0.3		4
1	Ngắt ở cạnh xuống của I0.0		5
3	Ngắt ở cạnh xuống của I0.1		6
5	Ngắt ở cạnh xuống của I0.2		7
7	Ngắt ở cạnh xuống của I0.3		8
12	Đếm vận tốc cao HSC0: trị đo bằng trị đặt		1
13	Đếm vận tốc cao HSC1: trị đo bằng trị đặt		9
14	HSC1 đổi hướng đếm		10
15	Xóa ngoài HSC1		11
16	Đếm vận tốc cao HSC2: trị đo bằng trị đặt		12
17	HSC2 đổi hướng đếm		13
18	Xóa ngoài HSC2		14
32	HSC3: trị đo bằng trị đặt		19
29	HSC4: trị đo bằng trị đặt		20
30	HSC4 đổi hướng đếm		21
31	Xóa ngoài HSC4		22

33	HSC5: trị đo bằng trị đặt		23
19	Đếm xung PLS0 xong		15
20	Đếm xung PLS1 xong		16
10	Ngắt thời gian 0		1
11	Ngắt thời gian 1		2
21	Ngắt timer T32	Thấp nhất	3
22	Ngắt timer T96		4

Tùy theo loại CPU có một số sự kiện không được hỗ trợ. Sau đây ta sẽ phân tích một số ngắt chính.

9.28.1 Ngắt thời gian

Ngắt thời gian 0/1 xảy ra theo chu kỳ ấn định (tối đa 255ms) bởi nội dung của SMB34/SMB35 (đơn vị ms) thường dùng để đọc hay xuất tín hiệu analog. Ngắt timer T32/T96 xảy ra khi timer T32 hoặc T96 hoàn tất thời gian trễ đã đặt.

Ví dụ:

NETWORK 1 // Subroutine 0

LD SM0.0

MOVB 100, SMB34 // Chu kỳ ngắt 100ms

ATCH INT_0 10 // Liên kết sự kiện số 10 với INT_0

ENI // Cho phép ngắt toàn cục

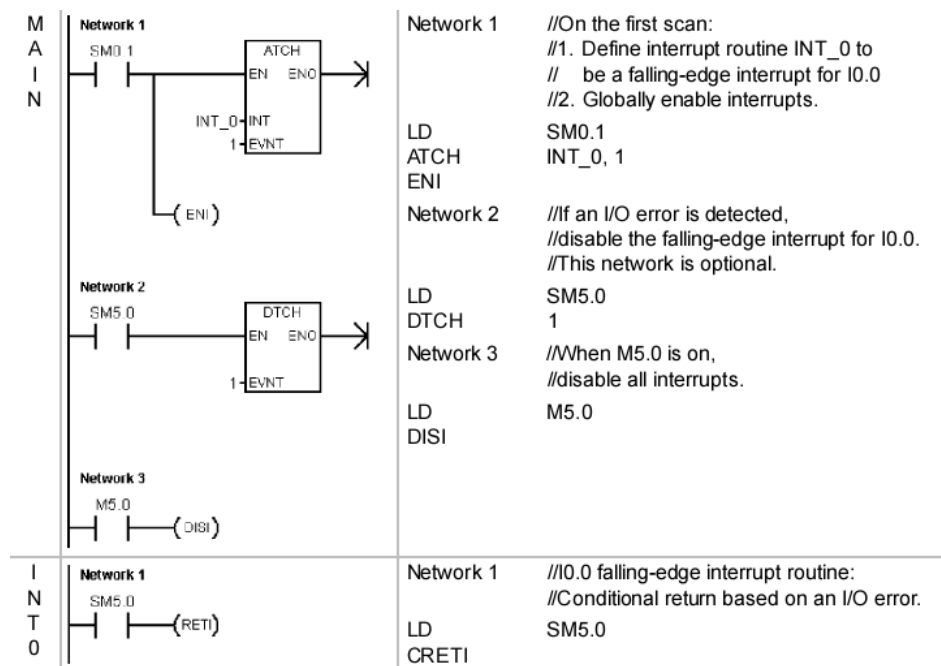
NETWORK 1 // Interrupt 0

// Đọc giá trị AIW4 mỗi 100ms

LD SM0.0

MOVW AIW4, VW100

9.28.2 Ngắt ngõ vào

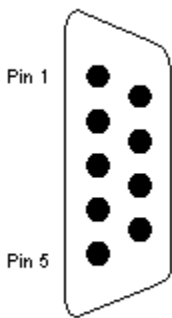


cuu duong than cong . com

cuu duong than cong . com

TRUYỀN THÔNG

PLC có một hoặc hai cổng (Port 0, Port 1) dùng để truyền thông nối tiếp RS485.

	Pin Number	Signal
	1	Shield
	2	24 V Return
	2	RS-485 Signal B
	4	Request-to-Send
	5	5 V Return
	6	+5 V
	7	+24 V
	8	RS-485 Signal A
	9	Not applicable

Thông qua cổng nối tiếp có thể ghép S7-200 với các thiết bị khác như S7-200, S7-300, HMI, PC. Muốn sử dụng cổng nối tiếp theo giao thức người dùng, ta phải đặt cổng ở chế độ freeport bằng cách đặt nội dung cho SMB30 (port 0) và SMB130 (port 1) theo bảng sau:

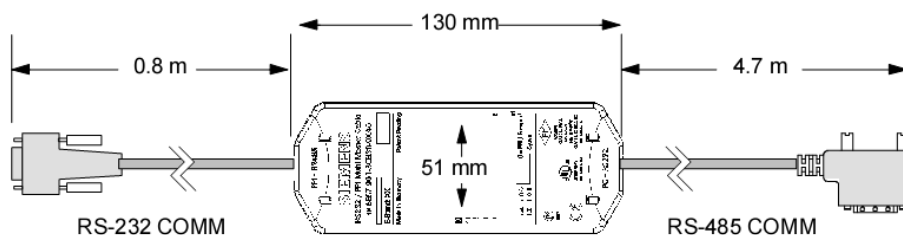
Bảng 9.6

SMB 30, SMB 130	Miêu tả
Bit 7, 6	00, 10: không parity
	01: parity chẵn
	11: parity lẻ
Bit 5	0: data 8 bit
	1: data 7 bit
Bit 4, 3, 2	000: 38400 bps (CPU212 19200)
	001: 19200
	010: 9600
	011: 4800
	100: 2400
	101: 1200
	110: 115200 (CPU212 600)
	111: 57600 (CPU212 300)
Bit 1, 0	00, 11: PPI slave (dùng cho mạng)
	01: Freeport
	10: PPI master (dùng cho mạng)

Như vậy với giao thức free port, 9600 baud ta nạp 9 vào SMB30 và 5 với 19200 baud

Ngoài chế độ truyền thông freeport mà giao thức do người dùng qui định, PLC còn dùng chế độ PPI master /slave, giao thức truyền thông đã qui định sẵn bởi phần mềm trong PLC.

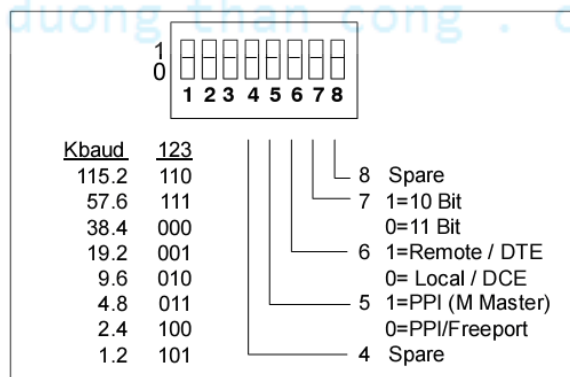
Kết nối PLC với thiết bị khác thông qua cable kết nối PPI Multi Master, có hai loại RS232 và USB, hay cable PPI/PC



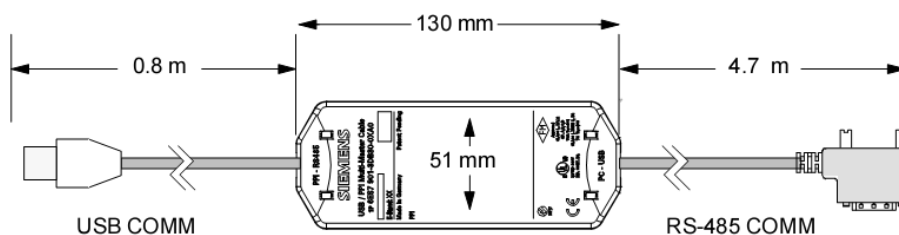
RS-485 Connector Pin-out		RS-232 Local Connector Pin-out	
Pin Number	Signal Description	Pin Number	Signal Description
1	No connect	1	Data Carrier Detect (DCD) (not used)
2	24 V Return (RS-485 logic ground)	2	Receive Data (RD) (output from PC/PPI cable)
3	Signal B (Rx/D/TxD+)	3	Transmit Data (TD) (input to PC/PPI cable)
4	RTS (TTL level)	4	Data Terminal Ready (DTR) ¹
5	No connect	5	Ground (RS-232 logic ground)
6	No connect	6	Data Set Ready (DSR) ¹
7	24 V Supply	7	Request To Send (RTS) (not used)
8	Signal A (Rx/D/TxD-)	8	Clear To Send (CTS) (not used)
9	Protocol select	9	Ring Indicator (RI) (not used)

¹ Pins 4 and 6 are connected internally.

Khi nối cáp với máy tính, đặt switch 7 vị trí 0, switch 6 vị trí local (0), còn khi nối với modem, đặt switch 6 vị trí remote (1), máy tính là Master nên chọn switch 5 vị trí 1, nếu chọn vận tốc truyền 9600 thì vị trí các switch là 01001000

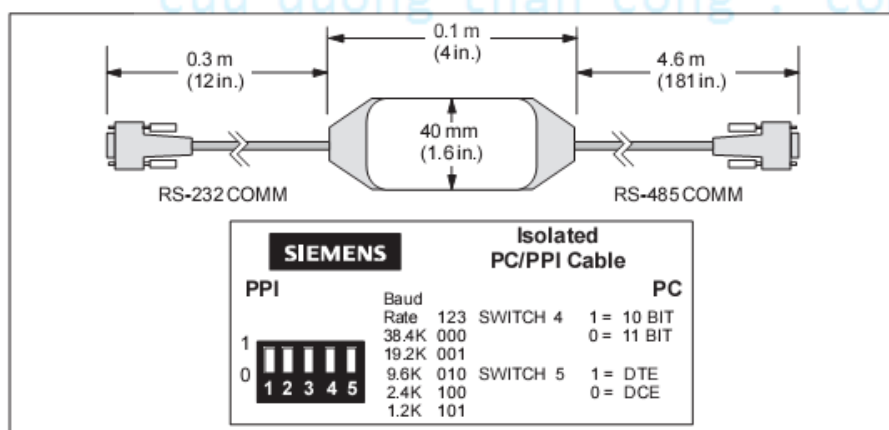


LED	Color	Description
Tx	Green	RS-232 transmit indicator
Rx	Green	RS-232 receive indicator
PPI	Green	RS-485 transmit indicator

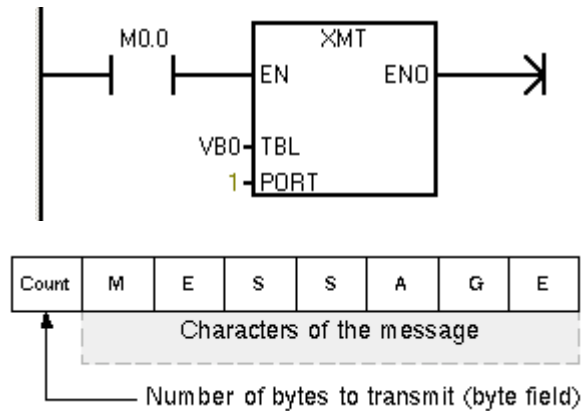


LED	Color	Description
Tx	Green	USB transmit indicator
Rx	Green	USB receive indicator
PPI	Green	RS-485 transmit indicator

RS-485 Connector Pin-out		USB Connector Pin-out	
Pin Number	Signal Description	Pin Number	Signal Description
1	No connect	1	USB - DataP
2	24 V Return (RS-485 logic ground)	2	USB - DataM
3	Signal B (Rx/D/TxD+)	3	USB 5V
4	RTS (TTL level)	4	USB logic ground
5	No connect		
6	No connect		
7	24 V Supply		
8	Signal A (Rx/D/TxD-)		
9	Protocol select (low = 10 bit)		



Tương tự PLC Omron, ở chế độ freeport có thể truyền và nhận data bằng hai lệnh XMT và RCV. Khi dùng lệnh XMT, dữ liệu truyền lấy từ bảng có địa chỉ đầu ở TABLE, còn dữ liệu nhận chứa vào TABLE, chiều dài bảng tối đa 255. Byte đầu của bảng cho biết chiều dài dữ liệu. Khi ký tự cuối của bảng được gửi sẽ báo sự kiện ngắt 9 hay 26 hay tác động SM4.5 (port 0), SM4.6 (port 1), hai bit này ON khi đã thực hiện xong việc truyền và OFF khi đang truyền.



Sự kiện ngắt 8 (Port 0), 25 (Port 1) sẽ xảy ra khi thu một ký tự, lúc này SMB2 chứa ký tự vừa nhận còn SM3.0 chứa kết quả kiểm tra parity (0: không có lỗi parity, 1: có lỗi parity).

Chú ý là không cùng lúc nhận và gửi dữ liệu qua port, giữa hai chế độ cần có khoảng nghỉ.

Ví dụ: Truyền dùng lệnh XMT và nhận ký tự dùng SMB2, máy tính gửi xuống ký tự A thì Q0.0 ON, gửi ký tự B thì Q0.0 OFF, khi I0.0 có cạnh lên thì PLC gửi lên ký tự A, I0.1 có cạnh lên thì PLC gửi lên ký tự B, máy tính nhận được A thì đèn Start sáng, máy tính nhận được B thì đèn Stop sáng, Có thời gian nghỉ 5ms từ khi PLC nhận dữ liệu đến khi gửi lên

//Chương trình chính

//Network 1

LD SM0.1

MOVB 9, SMB30

ATCH INT_0, 8

ENI

MOVB 'B', VB103

MOVB 'A', VB101

MOVB 1, VB102

MOVB 1, VB100

// Network2

LDN M1.0

A I0.0

EU

XMT VB100, 0

S Q0.1, 1

// Network3

LDN M1.0

A I0.1

EU

XMT VB102, 0

R Q0.1, 1

//Chương trình ngắt0

//Network1

LD SM0.0

LPS

S M1.0,1

AB= SMB2, 'A'

S Q0.0, 1

LPP

AB= SMB2, 'B'

R Q0.0, 1

//Network2

LD M1.0

MOVB 5, SMB34

ATCH INT1, 10

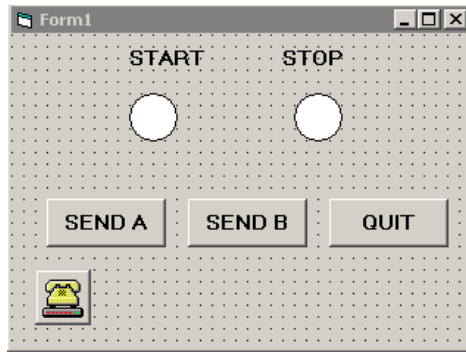
//Chương trình ngắt1

LD SM0.0

R M1.0, 1

DTCH 10

Chương trình VB



```
Private Sub cmdQuit_Click()

    MSComm1.PortOpen = False

    End

End Sub
```

```
Private Sub cmdSendA_Click()

    MSComm1.Output = "A"

End Sub
```

```
Private Sub cmdSendB_Click()

    MSComm1.Output = "B"

End Sub
```

```
Private Sub Form_Load()

    If MSComm1.PortOpen Then MSComm1.PortOpen = False

    With MSComm1

        CommPort = 3

        Settings = "9600,n,8,1"

        InputLen = 0

        InputMode = 0

        RThreshold = 1

        PortOpen = True

    End With

End Sub
```

```

End With

With Shape1
    . FillColor = vbWhite
    . Shape = 3
    . FillStyle = 0
End With

With Shape2
    . FillColor = vbWhite
    . Shape = 3
    . FillStyle = 0
End With
End Sub

```

```

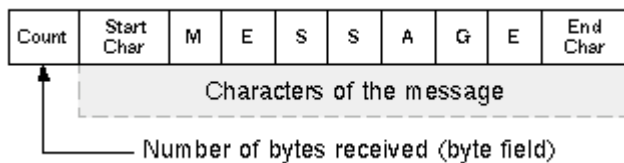
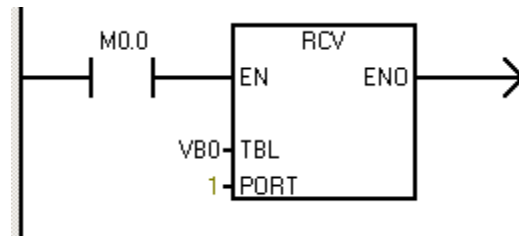
Private Sub MSComm1_OnComm()
    Dim data

    If MSComm1.CommEvent = comEvReceive Then
        data = MSComm1.Input

        Select Case data
            Case "A"
                Shape1.FillColor = vbBlue
                Shape2.FillColor = vbWhite
            Case "B"
                Shape1.FillColor = vbWhite
                Shape2.FillColor = vbRed
        End Select
    End If
End Sub

```

Lệnh RCV dùng để nhận một bản tin chứa vào một bảng, ta phải qui định các điều kiện khởi đầu và kết thúc cho bản tin và dùng các ô nhớ SMB86..SMB94 (Port 0), SMB186..SMB194 (Port 1) hỗ trợ, Khi thu xong bằng lệnh RCV sẽ tác động sự kiện ngắt 23 Port 0 (24 Port 1) . Chú ý là hai lệnh XMT và RCV không thể sử dụng đồng thời, nếu hai lệnh thực hiện cùng lúc thì lệnh sau không thực hiện và ENO là 0.



S7-200			
Symbol Name	SM Address		
	Port 0	Port 1	
P0_Start_Char	SMB88		Start of message character.
P1_Start_Char		SMB188	
P0_End_Char	SMB89		End of message character.
P1_End_Char		SMB189	
P0_Idle_Time	SMV90		Word data: Idle line time period given in milliseconds. The first character received after the idle line time has expired is the start of a new message.
P1_Idle_Time		SMV190	
P0_Timeout	SMV92		Word data: Inter-character/message timer timeout value given in milliseconds. If the time period is exceeded, the receive message is terminated.
P1_Timeout		SMV192	
P0_Max_Char	SMB94		Maximum number of characters to be received (1 to 255 bytes).
P1_Max_Char		SMB194	

Note: This range must be set to the expected maximum buffer size, even if the character count message termination is not used.

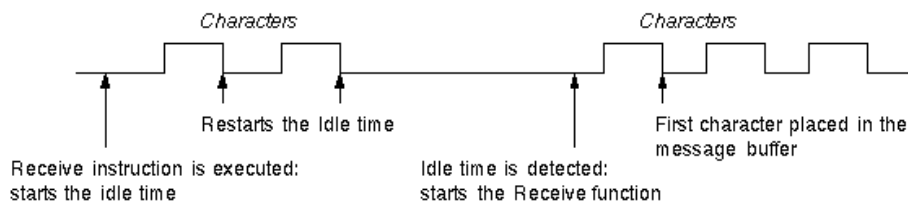
			Receive Message Control Byte								
			Bit Format	MSB				LSB			
				7						0	
	Port 0	Port 1		en	sc	ec	il	c/m	Tm r	bk	0
P0_Ctrl_Rcv	SMB87										
P1_Ctrl_Rcv		SMB187									
P0_Ctrl_Rcv_7	SM87.7		en:	0							= receive message function is disabled
P1_Ctrl_Rcv_7		SM187.7		1							= receive message function is enabled
P0_Ctrl_Rcv_6	SM87.6		sc:		0						= ignore SMB88 or SMB188
P1_Ctrl_Rcv_6		SM187.6			1						= use the value of SMB88 or SMB188 to detect start of message
P0_Ctrl_Rcv_5	SM87.5		ec:			0					= ignore SMB89 or SMB189
P1_Ctrl_Rcv_5		SM187.5				1					= use the value of SMB89 or SMB189 to detect end of message
P0_Ctrl_Rcv_4	SM87.4		il:				0				= ignore SMW90 or SMB190
P1_Ctrl_Rcv_4		SM187.4					1				= use the value of SMW190 to detect an idle line condition
P0_Ctrl_Rcv_3	SM87.3		c/m:					0			= timer is an inter-character timer
P1_Ctrl_Rcv_3		SM187.3						1			= timer is a message timer
P0_Ctrl_Rcv_2	SM87.2		tmr:						0		= ignore SMW92 or SMW192
P1_Ctrl_Rcv_2		SM187.2							1		= terminate receive if the time period in SMW92 or SMW192 is exceeded
P0_Ctrl_Rcv_1	SM87.1		bk:							0	= ignore break conditions
P1_Ctrl_Rcv_1		SM187.1								1	= use break condition as start of message detection

S7-200 Symbol Name	SM Address		Bit Format	Receive Message Status Byte							
	Port 0	Port 1		MSB				LSB			
P0_Stat_Rcv	SMB86			7					0		
P1_Stat_Rcv		SMB186		n	r	e	0	t	c	P	
P0_Stat_Rcv_7	SM86.7		n:	1							
P1_Stat_Rcv_7		SM186.7									
P0_Stat_Rcv_6	SM86.6		r:	1							
P1_Stat_Rcv_6		SM186.6									
P0_Stat_Rcv_5	SM86.5		e:	1							
P1_Stat_Rcv_5		SM186.5									
P0_Stat_Rcv_2	SM86.2		t:					1			
P1_Stat_Rcv_2		SM186.2									
P0_Stat_Rcv_1	SM86.1		c:					1			
P1_Stat_Rcv_1		SM186.1									
P0_Stat_Rcv_0	SM86.0		p:					1			
P1_Stat_Rcv_0		SM186.0									

Chú ý là thay đổi cấu hình chỉ hiệu quả khi thực hiện lệnh RCV. Khi lệnh RCV đã gây ngắt thì ngừng hoạt động

Để nhận biết bắt đầu bản tin có thể dùng nhiều cách:

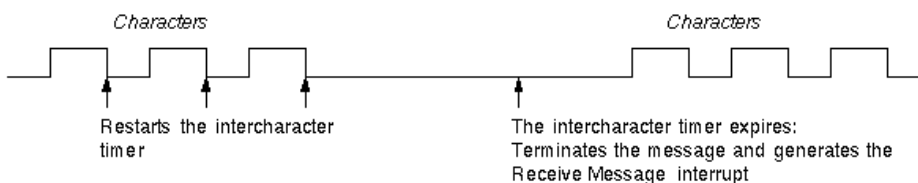
- Qui định ký tự bắt đầu trong ô nhớ SMB88 (SMB188), trong byte điều khiển cho bit sc SM87.6 (SM187.6) logic 1 (thường dùng khi truyền ký tự)
- Phát giác đường dây rỗi: qui định thời gian đường dây rỗi giữa hai lần phát bản tin, thường là hơn ba lần thời gian truyền một byte, trong ô nhớ SMW90 (SMW190) đơn vị là ms. Khi lệnh RCV được thực hiện PLC sẽ tính thời gian đường dây rỗi, các byte nhận trước thời gian này sẽ làm khởi động lại bộ tính thời gian này, sau thời gian rỗi qui định mới chấp nhận các byte đưa vào bảng (thường dùng khi truyền nhị phân) trong byte điều khiển cho bit il SM87.4 (SM187.4) lên 1



- Phát giác tín hiệu break: đường truyền ở điều kiện break khi nó ở mức zero hơn thời gian truyền một byte, lúc đó PLC sẽ nhận các ký tự sau điều kiện break, bit 1 (bk) của byte điều khiển đặt lên 1

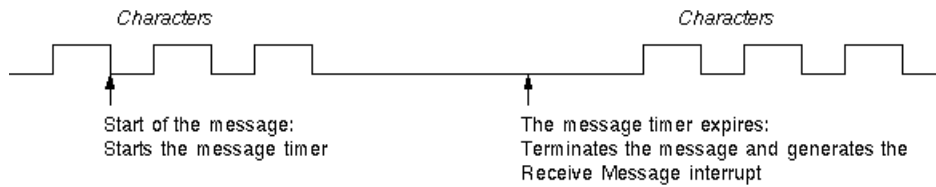
Để nhận biết kết thúc bản tin gây ra ngắt có thể dùng:

- Ký tự cuối trong ô nhớ SMB89 (SMB189), bit 5 (ec) byte điều khiển logic 1
- Timer giữa ký tự: nếu thời gian giữa hai ký tự lớn hơn thời gian trong ô nhớ SMW92 (SMW192) đơn vị ms thì coi như kết thúc truyền bản tin, bit 2 (tmr) của byte điều khiển mức 1, bit 3 (c/m) mức 0



- Timer bản tin: qui định thời gian truyền một bản tin trong ô nhớ SMW92 (SMW192) , sau thời gian này coi như kết thúc nhận, bit 2 (tmr) của byte điều

kiểm mức 1, bit 3 (c/m) mức 1



- Số ký tự tối đa: qui định truyền tối đa bao nhiêu ký tự SMB94 (SMB194), nếu nhận quá gây ra ngắt
- Sai parity
- Do kết thúc thu, đặt bit 7 (en) của byte điều khiển về 0

Ví dụ: nhận bản tin và truyền trở lại bản tin đã nhận dùng lệnh RCV và XMT

NETWORK 1 // Main Program

```
LD SM0.1 //On the first scan,
MOVB 16#09 SMB30 //Initialize Freeport:
// - Select 9600 baud
// - Select 8 data bits
// - Select no parity
MOVB 16#B0 SMB87 //Initialize RCV message control byte:
// RCV enabled, detect end of message character, detect idle line condition as message start condition
MOVB 16#0A SMB89 //Set end of message character to hex 0A (line feed)
MOVW +5 SMW90 //Set idle line timeout to 5 ms.
MOVB 100 SMB94 //Set maximum number of characters to 100.
ATCH INT_0 23 //Attach interrupt 0 to the receive complete event.
ATCH INT_2 9 //Attach interrupt 2 to the transmit complete event.
ENI
RCV VB100 0 //Enable receive box with buffer at VB100 for port 0
```

NETWORK 1 // Interrupt 0

```
//Receive complete interrupt routine
LDB= SMB86 16#20 //If receive status shows receive of end character,
MOVB 10 SMB34 //then attach a 10 ms timer
ATCH INT_1 10 //to trigger a transmit
CRETI //and return.
NOT
RCV VB100 0 //If the receive completed for any other reason, then start a new receive.
```

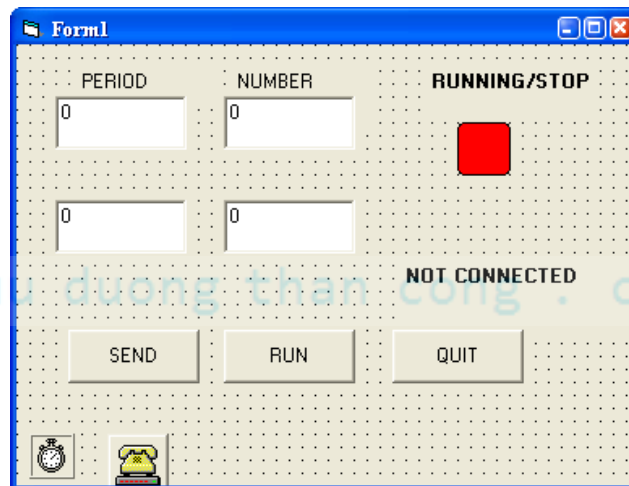
NETWORK 1 // Interrupt 1

```
//10 ms timer interrupt
LD SM0.0
DTCH 10 //Detach timer interrupt
XMT VB100 0 //Transmit message back to user on port 0
```

NETWORK 1 // Interrupt 2

```
LD SM0.0
RCV VB100 0
```

Ví dụ: Viết chương trình PLC nhận dữ liệu nối tiếp dạng mã ASCII, baudrate 19200. Lệnh thứ nhất ký tự “A” sau đó là **sáu byte**, sẽ được PLC cất vào **ba ô nhớ** từ VB0, dùng làm chu kỳ xung và số lượng xung cho lệnh phát xung PTO ở Q0.0. Lệnh thứ hai “BC” cho M0.0 ON.. Khi nhận được dữ liệu PLC gửi trả lại thông tin đã nhận, nếu M0.0 chuyển trạng thái từ OFF sang ON thì PLC cho M0.0 OFF và phát xung, phát đủ xung gửi lên máy tính ký tự ‘@’. Viết chương trình VB gửi và nhận dữ liệu với PLC như hình dưới. Trong ô txtPeriod gõ hai số hex, ô txtNumber gõ bốn số hex, bấm cmdSend gửi lệnh thứ nhất, bấm nút cmdRun gửi lệnh thứ hai. Khi PLC gửi trả lại lệnh thứ nhất hiển thị sáu byte trong hai ô text dạng số nguyên, khi PLC gửi trả lại lệnh thứ hai cho đèn shpRun màu xanh, khi PLC gửi “@” cho đèn này màu đỏ. Khi máy tính gửi lệnh, nếu sau 0.1s không nhận được trả lời từ PLC thì báo nhãn lblError “NOT CONNECTED”



```
Private Declare Function Beep Lib "kernel32" (ByVal dwFreq As Long, ByVal dwDuration As Long) As Long
```

```
Dim datain
```

```
Dim received
```

```
Dim timeout
```

```
Private Sub cmdQuit_Click()
```

```
    MSComm1.PortOpen = False
```

```
End
```

```
End Sub
```

```
Private Sub cmdRun_Click()  
    lblError.Visible = False  
    datain = ""  
    Shape1.FillColor = vbRed  
    MSComm1.Output = "BC"  
    Timer1.Enabled = True  
    received = False  
End Sub
```

```
Private Sub cmdSend_Click()  
    lblError.Visible = False  
    datain = ""  
    MSComm1.InBufferCount = 0  
    Text3.Text = ""  
    Text4.Text = ""  
    datain = ""  
    MSComm1.Output = "A" & Text1.Text & Text2.Text  
    cmdRun.Enabled = True  
    Timer1.Enabled = True  
    received = False  
End Sub
```

```
Private Sub Form_Load()  
    With MSComm1  
        .CommPort = 3  
        .Settings = "19200,n,8,1"  
        .RThreshold = 1  
        .InputLen = 0
```

```

.PortOpen = True

End With

lblError.Visible = False

Timer1.Interval = 500

Timer1.Enabled = False

Text1.Text = F0

Text2.Text = 0056

cmdRun.Enabled = False

received = False

datain = ""

End Sub

Private Sub MSComm1_OnComm()

    If MSComm1.InBufferCount > 0 Then
        received = True

        datain = datain & MSComm1.Input

    End If

    Text5.Text = datain

    If (Mid(datain, 1, 2) = "BC") Then

        Shape1.FillColor = vbBlue

        cmdRun.Enabled = False

        datain = ""

    End If

    If (Mid(datain, 1, 1) = "A") And Len(datain) > 6 Then

        Text3.Text = Mid(datain, 2, 2)

        Text4.Text = Mid(datain, 4, 4)

        datain = ""

    End If

    If Mid(datain, 1, 1) = "@" Then

```

```

        Shape1.FillColor = vbRed

        cmdRun.Enabled = True

        Beep 2000, 500

        datain = ""

    End If

End Sub

```

```

Private Sub Timer1_Timer()

    If Not (received) Then

        lblError.Visible = True

        Beep 1000, 200

        timeout = True

    End If

    Timer1.Enabled = False

End Sub

```

Chương trình PLC

Chương trình chính MAIN

Network 1	AENO
LD SM0.1	MOVB 1, VB20
LPS	AENO
MOVB 5, SMB30	MOVB '@, VB21
ATCH INT1, 19	LRD
AENO	MOVB 0, SMB68
ATCH INT0, 9	AENO
AENO	MOVW 50, SMW90
ATCH INT2, 23	AENO
LRD	MOVW 20, SMW92

ENI	LRD
	MOVD 0, VD10
MOVB 2#10011101, SMB67	AENO
AENO	MOVD 0, VD14
MOVB 2#10010100, SMB87	AENO
AENO	MOVW 0, SMW72
MOVB 100, SMB94	LPP
LRD	RCV VB10, 0

Ngắt Thu INT4

Network 1
LD SM0.0
XMT VB10, 0
= M0.1
= Q0.1

Network 2
LDB= VB10, 7
ATH VB12, VB0, 6
MOVB VB0, SMB69
MOVW VW1, SMW74

Network 3
LDB= VB10, 2
R Q0.0, 1
PLS 0
S M0.0, 1

Ngắt Truyền INT0

Network 1
LDN M0.0
RCV VB10, 0

Ngắt PTO INT1

Network 1 /
LD SM0.0
R M0.0, 1
XMT VB20, 0

cuu duong than cong . com

cuu duong than cong . com