



11

Special Function Instructions

In this chapter, we mainly introduce PWM pulse width modulation, frequency detect, precise time, interruption etc;

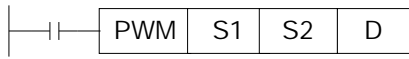
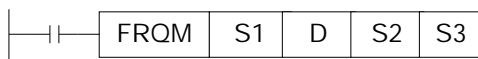
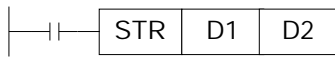





11-1 . PWM Pulse Width Modulation

11-2 . Frequency Detect

11-3 . Precise Time

11-4 . Interruption

Instructions List

| Mnemonic | Function | Circuit and soft components | Chapter |
|--|--|---|---------|
| Pulse Width Modulation, Frequency Detection | | | |
| PWM | Output pulse with the specified occupied ratio and frequency |  | 11-1 |
| FRQM | Frequency Detection |  | 11-2 |
| Time | | | |
| STR | Precise Time |  | 11-3 |
| STRR | Read Precise Time Register |  | 11-3 |
| STRS | Stop Precise Time |  | 11-3 |
| Interruption | | | |
| EI | Enable Interruption |  | 11-4-1 |
| DI | Disable Interruption |  | 11-4-1 |
| IRET | Interruption Return |  | 11-4-1 |

11-1 . PWM Pulse Width Modulation

1、 Instruction's Summary

Instruction to realize PWM pulse width modulation

| PWM pulse width modulation [PWM] | | | |
|----------------------------------|----------------------|----------------------|-------------------------|
| 16 bits instruction | PWM | 32 bits instruction | - |
| execution condition | normally ON/OFF coil | suitable models | XC1、 XC2、 XC3、 XC5、 XCM |
| hardware requirement | - | software requirement | - |

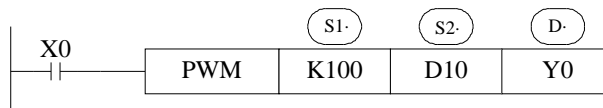
2、 Operands

| Operands | Function | Type |
|----------|--|--------------|
| S1 | specify the occupy ratio value or soft component's ID number | 16 bits, BIN |
| S2 | specify the output frequency or soft component's ID number | 16 bits, BIN |
| D | specify the pulse output port | bit |

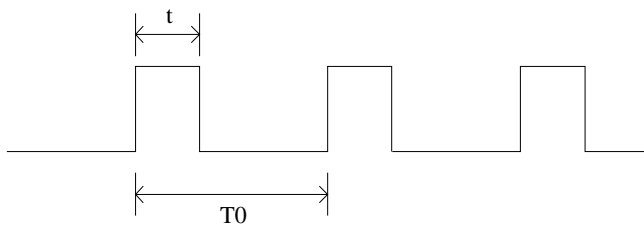
3、 Suitable Soft Components

| | | | | | | | | | | | | | |
|------|----------|--------|----|----|----|----|----|-----|----|----------|--------|----|----|
| Word | Operands | System | | | | | | | | Constant | Module | | |
| | | D | FD | ED | TD | CD | DX | DY | DM | DS | K/H | ID | QD |
| | S1 | | | | | | | | | | | | |
| | S2 | | | | | | | | | | | | |
| Bit | | | | | | | | | | | | | |
| | Operands | System | | | | | | | | | | | |
| | | X | Y | M | S | T | C | Dnm | | | | | |
| | D | | | | | | | | | | | | |

Function and Action



- | The occupy ratio **n**: 1~255
- | Output pulse **f**: 0~72KHz
- | Pulse is output at Y000 or Y001 (Please use transistor output)
- | The output occupy/empty ratio of PMW = $n / 256 \times 100\%$
- | PWM output use the unit of 0.1Hz, so when set (S2) frequency, the set value is 10 times of the actual frequency (i.e. 10f). E.g. : to set the frequency as 72KHz, then set value in (S2) is 720000.
- | When X000 is ON, output PWM wave ; when X000 is OFF, stop output. PMW output doesn't have pulse accumulation.



In the left graph: $T0=1/f$
 $T/T0=n/256$

11-2 . Frequency Testing

1、 Instruction's Summary

Instruction to realize frequency testing

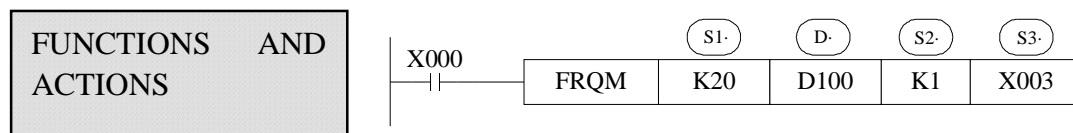
| frequency testing [FRQM] | | | |
|--------------------------|----------------------|----------------------|---------------------|
| 16 bits instruction | FRQM | 32 bits instruction | - |
| execution condition | normally ON/OFF coil | suitable models | XC1、XC2、XC3、XC5、XCM |
| hardware requirement | - | software requirement | - |

2、 Operands

| Operands | Function | Type |
|----------|---|--------------|
| S1 | Specify the sampling pulse number or soft component's ID number | 16 bits, BIN |
| S2 | Specify the frequency division choice's number | 16 bits, BIN |
| S3 | Specify the pulse input port | bit |
| D | specify the tested result's soft component's number | 16 bits, BIN |

3、Suitable Soft Components

| | | | | | | | | | | | | | |
|------|----------|--------|----|----|----|----|----|-----|----|----------|--------|----|----|
| Word | Operands | System | | | | | | | | Constant | Module | | |
| | | D | FD | ED | TD | CD | DX | DY | DM | DS | K/H | ID | QD |
| | S1 | | | | | | | | | | | | |
| | S2 | | | | | | | | | | | | |
| | D | | | | | | | | | | | | |
| Bit | Operands | System | | | | | | | | | | | |
| | | X | Y | M | S | T | C | Dnm | | | | | |
| | S3 | | | | | | | | | | | | |



- | S1: sampling pulse number: the number to calculate the pulse frequency
 - | D: tested result, the unit is Hz.
 - | S2: Frequency division choice. It can be K1 or K2;
- When the frequency division is K1, the range is: no less than 9Hz, precision range: 9~18KHz.
- When the frequency division is K2, the range: no less than 300Hz, precision range: 300~400KHz.
- | In frequency testing, if choose frequency division as K2, the frequency testing precision is higher than frequency division K1.
 - | When X000 is ON, FRQM will test 20 pulse cycles from X003 every scan cycle. Calculate the frequency's value and save into D100. Test repeatedly. If the tested frequency's value is smaller than the test bound, then return the test value as 0.

The pulse output to X number:

| Model | | X Number |
|------------|-----------------------|--------------|
| XC2 series | 14/16/24/32/48/60 I/O | X1、 X6、 X7 |
| XC3 series | 14 I/O | X2、 X3 |
| | 24/32 I/O | X1、 X11、 X12 |
| | 48/60 I/O、 XC3-19AR-E | X4、 X5 |
| XC5 series | 24/32 I/O | X3 |
| | 48/60 I/O | X1、 X11、 X12 |
| XCM series | 24/32 I/O | X3 |

11-3 . Precise Time

1、 Instruction List

Read and stop precise time when execute precise time;

| precise time [STR] | | | |
|--------------------------|-----------------|-----------------------|-------------------------|
| 16 bits instruction | - | 32 bits instruction | STR |
| execution condition | edge activation | suitable models | XC1、 XC2、 XC3、 XC5、 XCM |
| hardware requirement | - | software requirements | - |
| read precise time [STRR] | | | |
| 16 bits instruction | - | 32 bits instruction | STRR |
| execution condition | edge activation | suitable models | XC1、 XC2、 XC3、 XC5、 XCM |
| hardware requirement | V3.0e and above | software requirements | - |
| stop precise time [STRS] | | | |
| 16 bits instruction | - | 32 bits instruction | STRS |
| execution condition | edge activation | suitable models | XC1、 XC2、 XC3、 XC5、 XCM |
| hardware requirement | V3.0e and above | software requirements | - |

2、 Operands

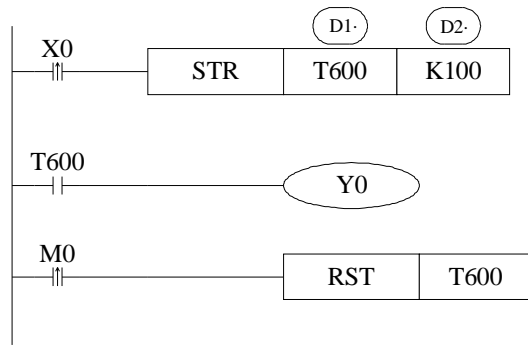
| Operands | Function | Type |
|----------|---|--------------|
| D | Timer's Number | bit |
| D1 | Timer's Number | bit |
| D2 | specify timer's value or soft component's ID number | 16 bits, BIN |

3、 Suitable Soft Components

| | | | | | | | | | | | | | |
|------|----------|--------|----|----|----|----|----|-----|----|----------|--------|----|----|
| Word | operands | system | | | | | | | | constant | module | | |
| | | D | FD | ED | TD | CD | DX | DY | DM | DS | K/H | ID | QD |
| | D2 | | | | | | | | | | | | |
| Bit | operands | system | | | | | | | | | | | |
| | | X | Y | M | S | T | C | Dnm | | | | | |
| | D | | | | | | | | | | | | |
| | D1 | | | | | | | | | | | | |

FUNCTIONS AND ACTIONS

《Precise Time》

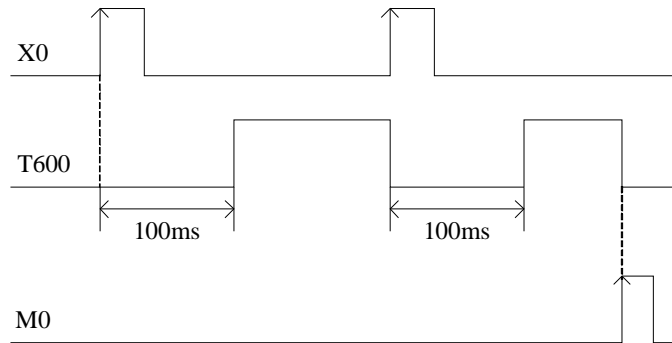


D1: Timer's number. Range: T600~T618 (T600、 T602、 T604...T618, the number should be even)

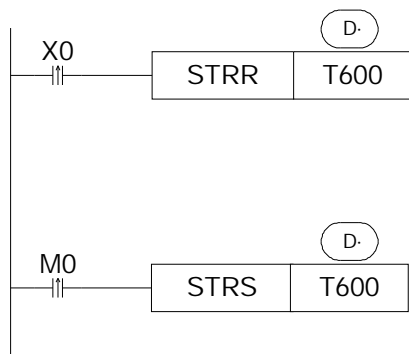
D2: Time Value

- | The precise timer works in form of 1ms
- | The precise timer is 32 bits, the count range is 0~+2,147,483,647.
- | When X000 turns from OFF to ON, timer T600 starts to time, when time accumulation reaches 100ms, set T600; if X000 again turns from OFF to ON, timer T600 turns from ON to OFF, restart to time, when time accumulation reaches 100ms, T600 again reset. See graph below:
- | When run STR instruction, reset the timer, then start to time;

See time graph below:



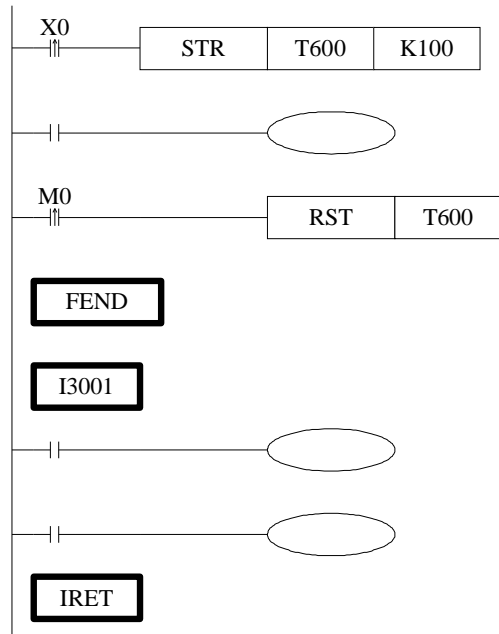
《read the precise time》、《stop precise time》



- | When X000 changes from OFF to ON, move the current precise time value into TD600 immediately, no relate to the scan cycle;
- | When M000 changes from OFF to ON, execute STRS instruction immediately, stop precise time and refresh the count value in TD600. No relate to the scan cycle;

Precious Time Interruption

- | When the precise time reaches the count value, generate a correspond interruption tag, execute some interruption subroutines.
- | Start the precise time in precise time interruption;
- | Every precise timer has its own interruption tag, see table below:



When X000 changes from OFF to be ON, timer T600 starts to time. When time accumulates to 100ms, set T600; meantime, generate an interruption, the program jumps to interruption tag I3001 and execute the subroutine.

Interruption Tag correspond with the Timer

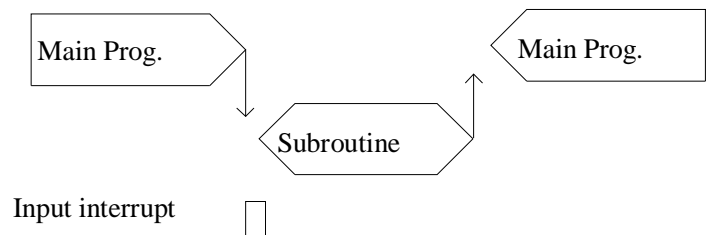
| Timer's Nr. | Interruption Tag |
|-------------|------------------|
| T600 | I3001 |
| T602 | I3002 |
| T604 | I3003 |
| T606 | I3004 |
| T608 | I3005 |
| T610 | I3006 |
| T612 | I3007 |
| T614 | I3008 |
| T616 | I3009 |
| T618 | I3010 |

11-4 . Interruption

XC series PLC are equipped with interruption function. The interruption function includes external interruption and time interruption. Via interruption function we can dispose some special programs. This function is not effected by the scan cycle.

11-4-1 . External Interruption

The input terminals X can be used to input external interruption. Each input terminal corresponds with one external interruption. The input's rising/falling edge can activate the interruption. The interruption subroutine is written behind the main program (behind FEND). After interruption generates, the main program stops running immediately, turn to run the correspond subroutine. After subroutine running ends, continue to execute the main program.



External Interruption's Port Definition

XC3-14

| Input Terminal | Pointer Nr. | | Disable the interruption instruction |
|----------------|---------------------|----------------------|--------------------------------------|
| | Rising Interruption | Falling Interruption | |
| X7 | I0000 | I0001 | M8050 |

XC2 series, XC3-24/32, XC5-48/60

| Input Terminal | Pointer Nr. | | Disable the interruption instruction |
|----------------|---------------------|----------------------|--------------------------------------|
| | Rising Interruption | Falling Interruption | |
| X2 | I0000 | I0001 | M8050 |
| X5 | I0100 | I0101 | M8051 |
| X10 | I0200 | I0201 | M8052 |

XC3-48/60, XC3-19AR-E

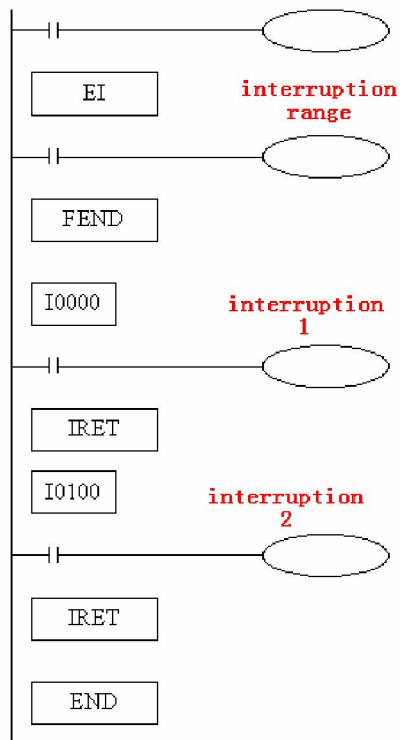
| Input Terminal | Pointer Nr. | | Disable the interruption instruction |
|----------------|---------------------|----------------------|--------------------------------------|
| | Rising Interruption | Falling Interruption | |
| X10 | I0000 | I0001 | M8050 |
| X7 | I0100 | I0101 | M8051 |
| X6 | I0200 | I0201 | M8052 |

XC5-24/32, XCM-24/32-

| Input Terminal | Pointer Nr. | | Disable the interruption instruction |
|----------------|---------------------|----------------------|--------------------------------------|
| | Rising Interruption | Falling Interruption | |
| X2 | I0000 | I0001 | M8050 |
| X5 | I0100 | I0101 | M8051 |
| X10 | I0200 | I0201 | M8052 |
| X11 | I0300 | I0301 | M8053 |
| X12 | I0400 | I0401 | M8054 |

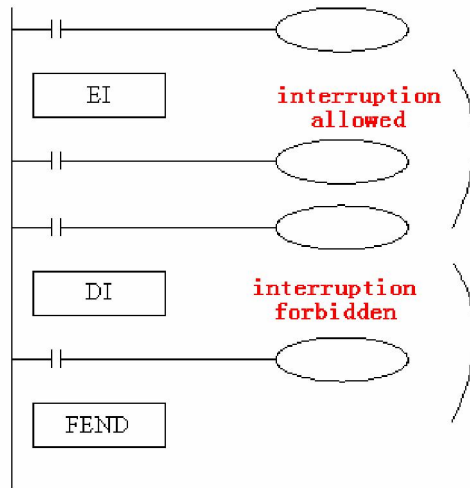
Interruption Instruction

Enable Interruption [EI], Disable Interruption [DI], Interruption Return [IRET]



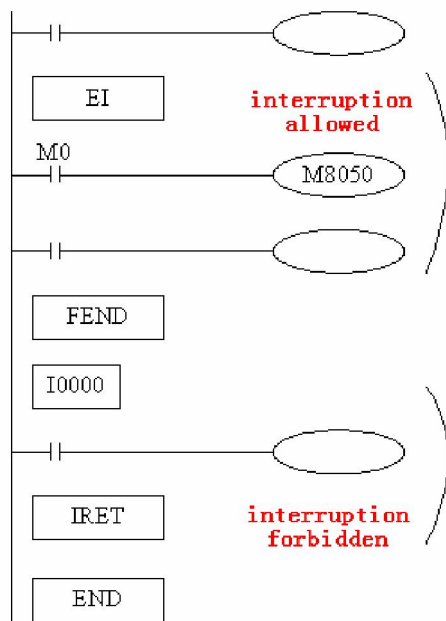
- | If use EI instruction to allow interruption, then when scanning the program, if interruption input changes from OFF to be ON, then execute subroutine 、 , return to the original main program;
- | Interruption pointer (I****) should be behind FEND instruction;
- | PLC is default to allow interruption

Interruption's Range Limitation



- | Via program with DI instruction, set interruption forbidden area;
- | Allow interruption input between EI~DI
- | If interruption forbidden is not required, please program only with EI, program with DI is not required.

Disable The Interruption

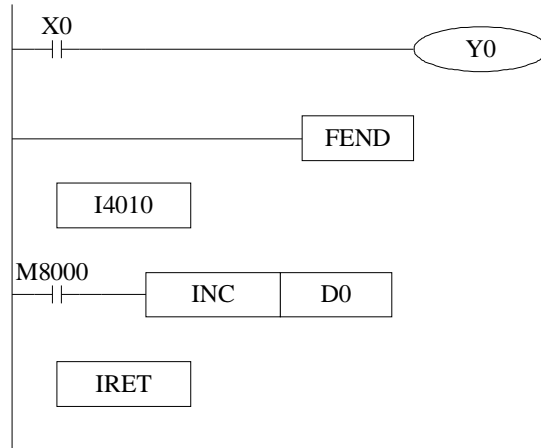


- | Every input interruption is equipped with special relay (M8050~M8052) to disable interruption;
- | In the left program, if use M0 to set M8050 "ON", then disable the interruption input at channel 0.

11-4-2 . Time Interruption

FUNCTIONS AND ACTIONS

In the condition of main program's execution cycle long, if you need to handle a special program; or during the sequential scanning, a special program needs to be executed at every certain time, time interruption function is required. This function is not affected by PLC's scan cycle, every Nm, execute time interruption subroutine.



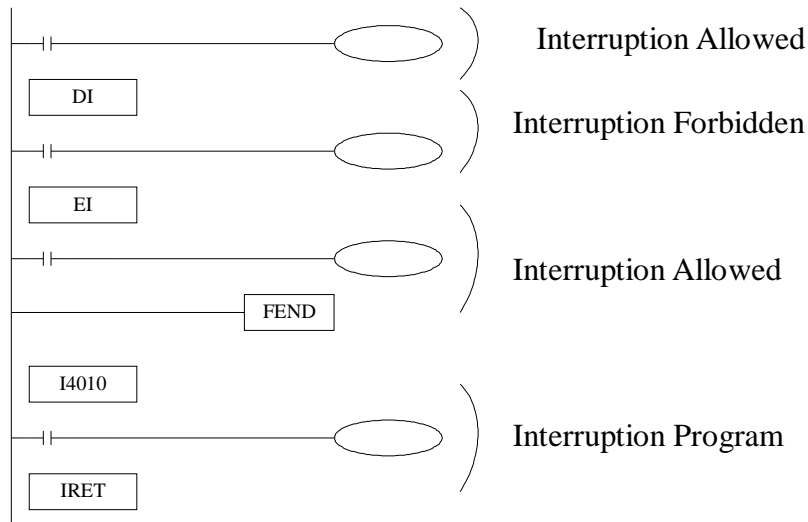
- | Time interruption is default in open status, time interruption subroutine is similar with other interruption subroutine, it should be written behind the main program, starts with I40xx, ends with IRET.
- | There are 10CH time interruptions. The represent method is I40**~I49** (“**” means time interruption’s time, unit is ms. For example, I4010 means run one channel time interruption every 10ms.

Interruption Nr.

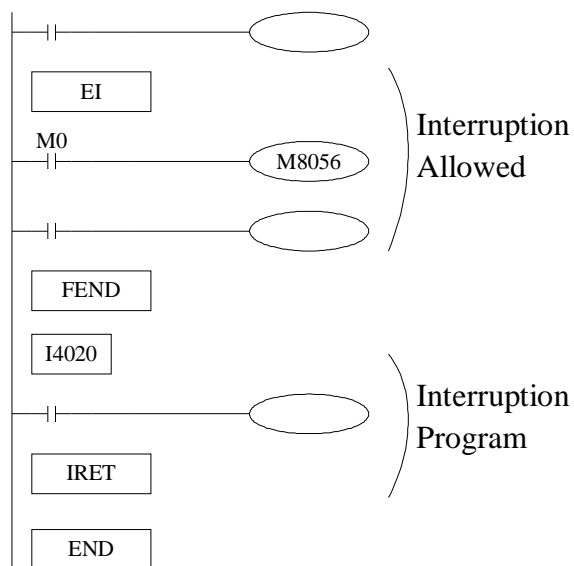
| Interruption Nr. | Interruption Forbidden Instruction | Description |
|------------------|------------------------------------|---|
| I40** | M8056 | “**” represents time interruption’s time, range from 1 to 99, unit is ms. |
| I41** | M8057 | |
| I42** | M8058 | |
| I43** | - | |
| I44** | - | |
| I45** | - | |
| I46** | - | |
| I47** | - | |
| I48** | - | |
| I49** | - | |

Interruption range's limitation

- Normally time interruption is in “allow” status
- With EI, DI can set interruption's allow or forbidden area. As in the above graph, all time interruptions are forbidden between DI~EI, and allowed beyond DI~EI.



Interruption Forbidden



- The first 3CH interruptions are equipped with special relays (M8056~M8059) to forbid interrupt

- In the left example program, if use M0 to enable M8056 “ON”, the forbid 0CH's time interruption.

