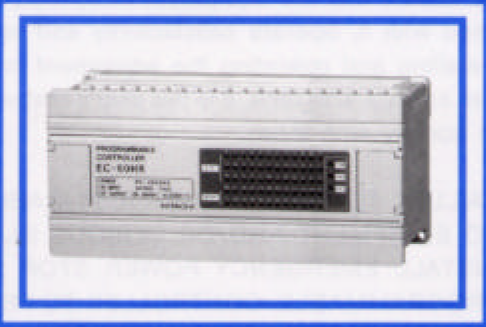
HITACHI

OPERATION MANUAL

EC SERIES BOARD TYPE

[EC - ××HRP]
[EC - D××HRP]
[EC - ××HR]
[EC - D××HR]



HITACHI PROGRAMMABLE CONTROLLER

NJI 124B(X)

WARNING

To ensure that the equipment described by this manual. As well as all equipment connected to and used with it, operate satisfactorily and safety, all applicable local and national codes that apply to installing and operating the equipment must be followed. Since codes can vary geographically and can change with time, it is the user’s responsibility to determine which standard and codes apply, and to comply with them.

FAILURE TO COMPLY WITH APPLICABLE CODES AND STANDARDS CAN RESULT IN DAMAGE TO EQUIPMENT AND/OR SERIOUS INJURY TO PERSONNEL.

INSTALL EMARGENCY POWER STOP SWITCH, WHICH OPERATES INDEPENDENTLY OF THE PROGRAMMABLE CONTROLLER TO PROTECT THE EQUIPMENT AND/OR PERSONNEL IN CASE OF THE CONTROLLER MALFUNCTION.

Personnel who are to install and operate the equipment should carefully study this manual and any other referred to by it prior to installation and/or operation of the equipment. Hitachi Industrial Equipment Systems Co., Ltd. constantly strives to improve its products, and the equipment and the manual(s) that describe it may be different from those already in your possession.

If you have any questions regarding the installation and operation of the equipment, or if more information is desired, contact your local Authorized Distributor or Hitachi Industrial Equipment Systems Co., Ltd.

IMPORTANT

THIS EQUIPMENT GENERATES, USES, AND CAN RADIATE RADIO FREQUENCY ENERGY AND, IF NOT INSTALLED AND USED IN ACCORDANCE WITH THE INSTRUCTION MANUAL, MAY CAUSE INTERFERENCE TO RADIO COMMUNICATIONS. AS TEMPORARILY PERMITTED BY REGULATION, IT HAS NOT BEEN TESTED FOR COMPLIANCE WITH THE LIMITS FOR CLASS A COMPUTING DEVICES PURSUANT TO SUBPART J OF PART 15 OF FCC ROULES, WHICH ARE DESIGNED TO PROVIDE PEASONABLE PROTECTION AGAINST SUCH INTERFERENCE.

OPERATION OF THIS EQUIPMENT IN A RESIDENTIAL AREA IS LIKELY TO CAUSE INTERFERENCE IN WHICH CASE THE USER, AT HIS OWN EXPENSE, WILL BE REQUIRED TO TAKE WHATEVER MEASURES MAY BE REQUIRED TO CORRECT THE INTERFERENCE.

LIMITED WARRANTY AND IMITATION OF LIABILITY

Hitachi Industrial Equipment Systems Co., Ltd. (Hitachi) warrants to the original purchaser that the programmable logic controller (PLC) manufactured by Hitachi is free from defects in material and workmanship under normal use and service. The obligation of Hitachi under this warranty shall be limited to the repair or exchange of any part or parts which may prove defective under normal use and service within eighteen (18) months from the date of manufacture or twelve (12) months from the date of installation by the original purchaser which ever occurs first, such defect to be disclosed to the satisfaction of Hitachi after examination by Hitachi of the allegedly defective part or parts. This warranty in expressly in lieu of all other warranties expressed or implied including the warranties of merchantability and fitness for use and of all other obligations or liabilities and Hitachi neither assumes, nor authorizes any other person to assume for Hitachi, any other liability in connection with the sale of this PLC. This warranty shall not apply to this PLC or any part hereof which has been subject to accident, negligence, alternation, abuse, or misuse. Hitachi makes no warranty whatsoever in respect to accessories or parts not supplied by Hitachi. The term “original purchaser”, as used in this warranty, shall be deemed to mean that person for whom the PLC in originally installed.

In no event, whether as a result of breach of contract, warranty, tort (including negligence) or otherwise, shall Hitachi or its suppliers be liable for any special, consequential, incidental or penal damages including but not limited to, loss or profit or revenues, loss of use of the products or any associated equipment, damage to associated equipment, cost of capital, cost of substitute products, facilities, services or replacement power, down time costs, or claims of original purchaser’s customers for such damages.

To obtain warranty service, return the product to your distributor, or send it with a description of the problem, proof of purchase, post paid, insured, and in a suitable package to:

Quality Assurance Dept.
Hitachi Industrial Equipment Systems Co., Ltd.
46-1 Ooaza-Tomioka Nakajo-machi
Kitakanbara-gun, Niigata-ken
959-2608 JAPAN

The Information and/or drawing set forth in this document and all right in and to inventions disclosed herein and patent which might be granted thereon disclosing or employing and the materials, methods, techniques or apparatus described herein are the exclusive property of Hitachi Industrial Equipment Systems Co., Ltd .

No copies of the information or drawings shall be made without the prior constant of Hitachi Industrial Equipment Systems Co., Ltd.

Hitachi Industrial Equipment Systems Co., Ltd. provides customer assistance in varied technical areas. Since Hitachi does not possess full access to data concerning all of the uses and applications of customer's products, responsibility is assumed by Hitachi neither for customer product design nor for any infringement of patents or rights of others, which may result from Hitachi assistance.

The specifications and descriptions contained in this manual were accurate at the time they were approved for printing. Since Hitachi Industrial Equipment Systems Co., Ltd. Incorporated constantly strives to improve all its products, we reserve the right to make changes to equipment and/or manual at any time without notice and without incurring any obligation other than as noted in this manual.

Hitachi Industrial Equipment Systems Co., Ltd. assumes no responsibility for errors that may appear in this manual.

As the product works with user program, and Hitachi Industrial Equipment Systems Co., Ltd. cannot test all combination of user program components, it is assumed that a bug or bugs may happen unintentionally. If it is happened: please inform the fact to Hitachi Industrial Equipment Systems Co., Ltd. or its representative. Hitachi will try to find the reason as much as possible and inform the countermeasure when obtained.

Nevertheless Hitachi Industrial Equipment Systems Co., Ltd. intends to make products with enough reliability, the product has possibility to be damaged at any time. Therefore personnel who are to install and operate the equipment have to prepare with the countermeasure such as power off switch can be operated independently of the controller. Otherwise, it can result in damage to equipment and/or serious injury to personnel.

TABLE OF CONTENTS

OPERATION

MANUAL

EC SERIES

System Configuration	**Unit Specifications**	Name of Unit Components	Specifications

Table 1-1 Unit Specifications (1/3)

Item	Model Name	Main specifications				Remarks
		input		output		
Power source voltage 24VDC Positive logic	EC-D20HRP	DC input	12	Relay output	8	
	EC-D28HRP		16		12	
	EC-D40HRP		24		16	
	EC-D60HRP		36		24	
	ECL-D20HRP		12		8	high functional type
	ECL-D40HRP		24		16	high functional type
	ECL-D60HRP		36		24	high functional type
Power source voltage 100~240VAC Positive logic	EC-20HRP		12		8	
	EC-28HRP		16		12	
	EC-40HRP		24		16	
	EC-60HRP		36		24	
	ECL-20HRP		12		8	high functional type

Table 1-1 Unit Specifications (1/3)

Item	Model Name	Main specifications				Remarks
		input		output		
Power source voltage 24VDC Positive logic	EC-D20HRP	DC input	12	Relay output	8	
	EC-D28HRP		16		12	
	EC-D40HRP		24		16	
	EC-D60HRP		36		24	
	ECL-D20HRP		12		8	high functional type
	ECL-D40HRP		24		16	high functional type
	ECL-D60HRP		36		24	high functional type
Power source voltage 100~240VAC Positive logic	EC-20HRP		12		8	
	EC-28HRP		16		12	
	EC-40HRP		24		16	
	EC-60HRP		36		24	
	ECL-20HRP		12		8	high functional type

Table 1-1 Unit Specifications (3/3)

Item		Model Name	Main Specifications	Remarks
Cable	For programmer extension	CNPG-15	Between EC／ECL and programmer	1.5m
Peripheral equipment	Standard programmer	PGMJ	Audio cassette interface	Instruction word
	Universal programmer	PGMJ-R2	With audio cassette interface, and RS-232C serial port.	
	Software package for personal computer programming	ACTSIP-E ACTGRAPH + E	Ladder software for IBM PC Grafcet software for IBM PC	Sold by ACTRON AB.
		EB,EM PROGRAMMER	Ladder software for IBM PC	Sold by LOGITEK S.A.
Memory pack		MPE-1E	EEPROM 925 words	Used only for storing and copying program using PGM J-R2
		MPE-2E	EEPROM 1949 word's	

CAUTION

MPM-1E, MPM-2E, MPM-2R, MPE-2R are not available for using storing and copying program with PGMJ-R2.

Table 1-1 Unit Specifications (3/3)

Item		Model Name	Main Specifications	Remarks
Cable	For programmer extension	CNPG-15	Between EC／ECL and programmer	1.5m
Peripheral equipment	Standard programmer	PGMJ	Audio cassette interface	Instruction word
	Universal programmer	PGMJ-R2	With audio cassette interface, and RS-232C serial port.	
	Software package for personal computer programming	ACTSIP-E ACTGRAPH + E	Ladder software for IBM PC Grafcet software for IBM PC	Sold by ACTRON AB.
		EB,EM PROGRAMMER	Ladder software for IBM PC	Sold by LOGITEK S.A.
Memory pack		MPE-1E	EEPROM 925 words	Used only for storing and copying program using PGM J-R2
		MPE-2E	EEPROM 1949 words	

CAUTION

MPM-1E, MPM-2E, MPM-2R, MPE-2R are not available for using storing and copying program with PGMJ-R2.

Fig. 1-2 shows the name of each part provided on the standard type of EC series, in the case of the EC-60HRP. The remaining EC-20, 28, 40 are different only in the number of I/O points.

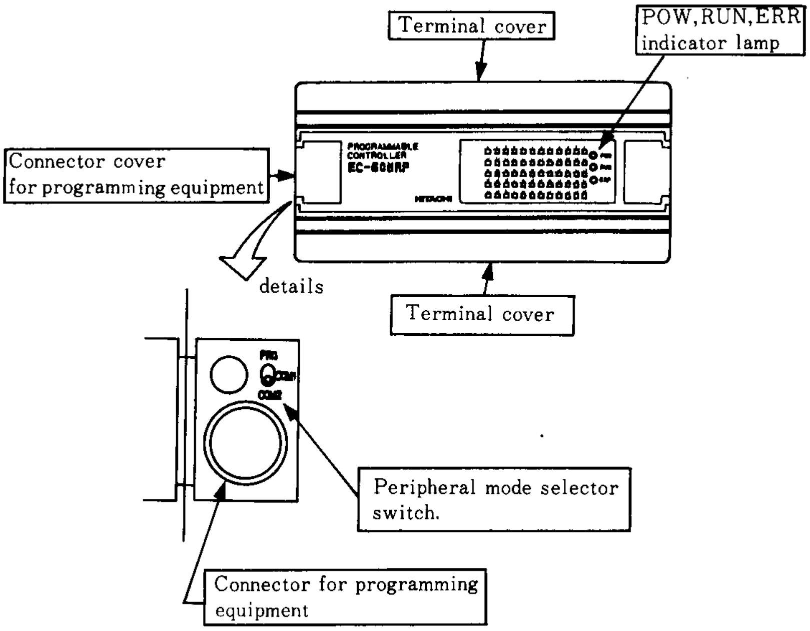

Terminal cover
POW,RUN,ERR indicator lamp
Connector cover for programming equipment
PROGRAMMABLE CONTROLLER
EC-60HRP
HITACHI
details
Terminal cover
PRG
COM1
COM2
Peripheral mode selector switch.
Connector for programming equipment

Figure 1-2 Names of Standard type Parts (EC type)

Table 1-2 Basic Specifications (1/2)

Item			EC	ECL (NOTE 7)
Control specifications	Control system		Stored program cyclic system	
	Processing speed		1.5μs/basic command (NOTE 1)	
	Program capacity		1949 words	
	Kind of memory		EEPROM	
Processing functions	Basic instruction		12 kinds (ORG, STR, AND, OR, ORSTR, ANDSTR, OUT, etc.)	
	Application instruction		21 kinds (edge detection, step, master control, jump, etc.)	
	Arithmetic instruction		36 kinds (word load, word out, 4 rule calculations, logic calculation, comparison, carry, etc.)	
Input/output processing specifications	No. of external input/output points		60 points max	
	No. of internal outputs	Retentive at power failure	256 points (128 words)	
		Non-retentive at power failure	256 points (128 words)	
		Special function	12 points+4 words	
	Timer/ counter	Counting system	Addition	
		No. of points	96 points	
		Timer preset value	0.01 to 9.99 sec, 0.1 to 99.9 sec, 1 to 999 sec (NOTE 2)	
		Counter preset value	1 to 999 times (NOTE 3)	
	High speed counter		Addition/subtraction 1 point, 2-phase, 10 kHz, BCD 8 digits (NOTE 4)	
	External interruption input		1 point (NOTE 5)	
	Operation control input		1 point (exclusive terminal: start signal)	

Table 1-2 Basic Specifications (2/2)

Item	EC	ECL
Peripherals	Standard programmer: PGMJ (audio cassette interface) Universal programmer: PGMJ-R2 (audio cassette interface)	
Serial interface	RS-232C	
Memory protection at power failure	Capacitor backup for 2 weeks (at 25℃) (NOTE 6)	

NOTES:
1 The OUT T/C instruction is excluded.
2 T0 to T9 are presettable from 0.1 to 999.9 sec.
3 C0 to C9 are presettable from 1 to 9999 times.
4 Depending on program, inputs X0 to X2 serve as high speed counter inputs. Although change to single phase is possible.
5 Depending on program, the input X3 functions to input interruption.
6 Since EEPROM is used for program memory, there is no need for backup. Only internal output and timer/counter current value are to be protected when memory protection against power failure is provided.
7 As concerns additional functions of ECL, refer separate volume for ECL.

Table 1-3 General Specifications

Item		Specifications
Line voltage	Rated voltage	100V AC to 240V AC (without switching) (AC type)／24VDC(DC type)
	Allowable fluctuation	85V AC to 264V AC (AC type)／19.2~30VDC(DC type)
Frequency	Rated frequency	50/60 Hz(AC type)
	Allowable fluctuation	47 to 63 Hz(AC type)
Allowable momentary power failure		20 ms or shorter
Power consumption		See Table 1-4.
Dielectric strength		1,500V AC for 1 min. between input/output terminals (including power terminal) and ground terminal (NOTE 1)
Insulation resistance		20MΩ or more for 1 min, between input/output terminals (including power terminal) and ground terminal when measured with 500V DC megger (NOTE 1)
Operating temperature		0 to 55℃
Storage temperature		−10 to 75℃ (Retention of the memory contents is assured only within the operating temperature range.)
Operating humidity		20 to 90% RH (non-condensing)
Storage humidity		10 to 90% RH (non-condensing)
Vibration resistance		Frequency 16.7Hz,multi-amplitude 3mm in X,Y and Z directions.
Shock resistance		Conforms to JIS C0912 on condition that shock of 10G is applied twice in each of X, Y and Z directions.
Noise resistance		Noise voltage 1,500 Vp-p, pulse with 1 μs (Measurement by our company method with noise simulator)
Environment		Must be free from corrosive gas and dust
Altitude		2,000m or less
Grounding		100Ω max. (exclusive grounding required)
Weight		See Table 1-4

NOTE: 1 A varistor for suppressing lightning surge is connected to the power supply terminal. Therefore, the external connector (between Z terminal to FG terminal) must be separated when testing dielectric strength or insulation resistance of the power supply terminal.

Table 1-4 Power/Current Consumption, Weight and Dimensions (NOTE 2)

Item		EC-20 ECL-20	EC-D20 ECL-D20	EC-28	EC-D28	EC-40 ECL-40	EC-D40 ECL-D40	EC-60 ECL-60	EC-D60 ECL-D60
Power consumption (VA) at 220V AC		18VA or less	——	20VA or less	——	23VA or less	——	28VA or less	——
Current consumption (mA) at 24V DC		——	400mA or less	——	500mA or less	——	600mA or less	——	700mA or less
Weight (kg)		0.6	0.5	0.6	0.5	0.6	0.5	0.6	0.5
External dimension (mm)	W	170		170		170		170	
	H	75		75		75		75	
	D	80		80		80		80	

NOTE: 2 For mounting dimensions, refer to CHAPTER 5.

Table 1-5 DC Input Specifications (NOTE) (1/3)

Item		Specifications	
System		Non-voltage contact or PNP transistor (open collector) (HRP type) Non-voltage contact or NPN transistor (open collector) (HR type)	
Nominal voltage		24V DC (built-in power supply can be used)	
Input current		Approx. 7mA/24V DC (at an impedance of about 3.4 kΩ)	
Operating voltage	ON	ON voltage: 19V or more (ON resistance: 300Ω or less)	
	OFF	OFF voltage: 7V or less (OFF resistance: 200kΩ or more)	
Max, input delay time	OFF to ON	5ms ± 2.5ms	Lag time of input No. 0-3 is selectable by software.
	ON to OFF	5ms ± 2.5ms	
Polarity		Common terminal…(−) or (+) corresponding to input logic	
Isolation method		Photocoupler	
Sensor power supply		470mA−(7mA×No. of ON input)−(6mA×No. of ON output)	

Table 1-5 DC Input Specifications (NOTE) (2/3)

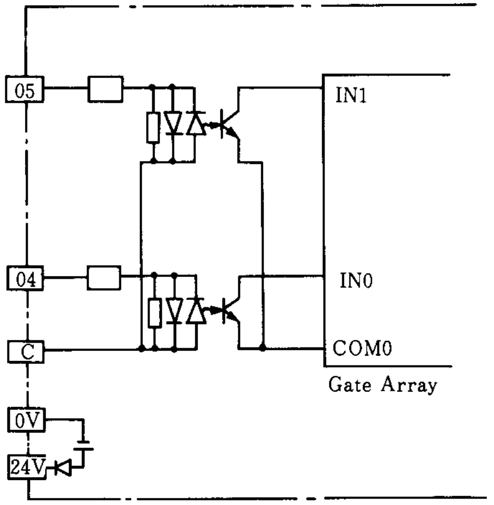
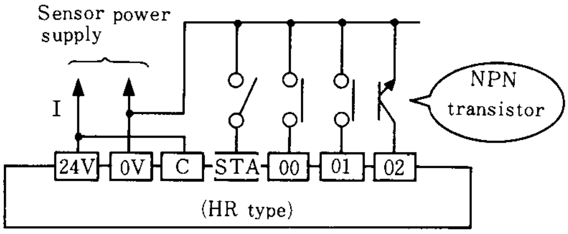
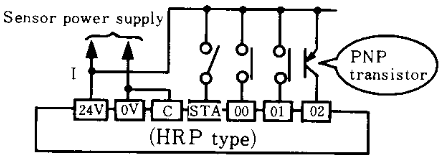
Item	Specifications
Circuit diagram (a part)	

Table 1-5 DC Input Specifications (NOTE) (3/3)

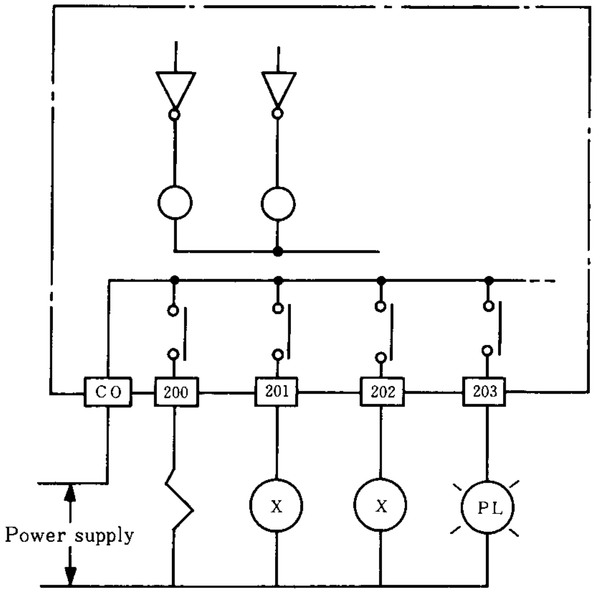
Item	Specifications
Instance of external wiring	 Sensor power supply I NPN transistor 24V 0V C STA 00 01 02 (HR type)  Sensor power supply I PNP transistor 24V 0V C STA 00 01 02 (HRP type) Current flown through sensor I = 470mA − (7mA× no. of ON input) − (6mA× no. of ON output)

NOTE: Operation control input (STA terminal) has the same specifications as above.

Table 1-6 Relay Output Specifications (1/2)

Item		Specifications
System		Relay contact output
Rated output voltage		100V AC / 200V AC, 24V DC
Max, load current in total	1 circuit	1A (cosϕ = 1.0)
	2 circuits(1 common)	2A (cosϕ = 1.0) (1 circuit 1A (cosϕ = 1.0))
	4 circuits(1 common)	4A (cosϕ = 1.0) (1 circuit 1A (cosϕ = 1.0))
	6 circuits(1 common)	4A (cosϕ = 1.0) (1 circuit 1A (cosϕ = 1.0))
Min, load current		1mA (5V DC)
Leakage current		Negligible
Max, output delay time	OFF to ON	10ms
	ON to OFF	10ms
Polarity		None
Isolation method		Relay
Lifetime	Electrical	○200k times or more at 120V AC and 1A resistive load ○1000k times or more with Hitachi magnetic contactor (14VA, 50Hz at power ON, 6VA after power ON)
	Mechanical	10,000k times or more

Table 1-6 Relay Output Specifications (2/2)

Item	Specifications
Circuit diagram and external wiring	 CO 200 201 202 203 X X PL Power supply

2. Input / output and numbers

External Inputs (X) External outputs (Y)	Internal outputs (M)	Timer (T)	Counter (C)

I/O number assignment

Table 2-1 shows how the external inputs (X), external outputs (Y), internal outputs (M) and timers/counters (T/C) are assigned.

Table 2-1 Assignment of I/O Numbers (1/2)

I/O classification	I/O No.		Remarks
External input (36 points)	X0~X11	EC(L)-20	○Decimal numbers. Numbers 16 to 19, 36 to 39 are omitted. ○Each number has a data capacity of 8 bits. But only the uppermost-digit bit b7 is actually reflected on the external terminal. Example Y200 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | ↑ Bit to be reflected on external terminal
	X12~X15	EC-28	
	X20~X27	EC(L)-40	
	X28~X35 X40~X43	EC(L)-60	
External output (24 points)	Y200~Y207	EC(L)-20	○All numbers are usable indifferently to EC(L) types. Due to reasons of hardware configuration, the numbers unused for the external terminals are handled as follows. External input···Can be programmed, but remains turned off during operation. External output···Can be programmed and provides the same function as the internal output(M).
	Y208~Y211	EC-28	
	Y212~Y215	EC(L)-40	
	Y220~Y227	EC(L)-60	

Table 2-1 Assignment of I/O Numbers (2/2)

I/O classification			I/O No.	Remarks
Internal output	Non-retentive memory at power failure (256 points)		M400~M655	○Decimal numbers ○Each number has a data capacity of 8 bits. Example M400 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | M401 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 | ○The bit handling instruction determines ON/OFF status of b7. ○The word handling instruction handles 8-bit data of M400 and that of M401, 16 bits in total, when No. 400 is designated.
	Retentive memory at power failure (256 points)		M700~M955	
	Special function (16 points)		M960~M991	Detailed in Table **2-3**.
Timers and counters (96 points in total)		Coil contact	TC 0~TC 95	○Decimal numbers ○Timer and counter share the same number. ○Up timer and up counter, respectively ○100 is added to timer/counter number (2-digit) for representing a current value, and 200 is added for indicating preset value. ○States of coil and contact are shown by bit data ○Current value and preset value are 4-digit BCD.
		Current value	TC100~TC195	
		Preset value	TC200~TC295	

Table 2-2 lists each range of constant and argument used in instructions such as AJMP, INT or MODE.

Table 2-2 Each Range of Constant and Argument

I/O classification		Range	Remarks
Constant	Word constant	0000H~FFFFH	The hexadecimal code H is not suffixed at the time of program entry. (Example) FUN 0. 1234(AR ← 1234H)
	Byte constant	0~255	Only FUN50 uses this expression. (Example) FUN50 123(AR ← 123)
Argument		0~63	Used as an argument of FUN08(AJMP), FUN09(AJEND), FUN93(INT), FUN96(HC) and FUN97(MODE) instructions. (Example) FUN08 63(AJMP63)

Table 2-3 Function of Special Internal Output (1/3)

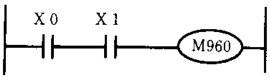
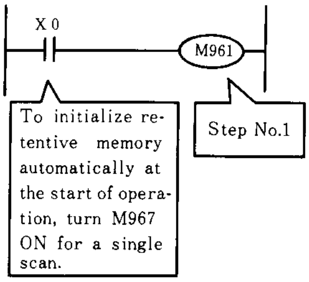
No.	Operation	Function
M960	All outputs OFF	When M960 is switched ON by the program, all output signals go OFF. **Example**  ○Suppose that an error program is written. (The X0 and X1 are not closed simultaneously during normal operation.) As a result, M960 is switched ON. In this status, the PC judges that there is a system error and it switches all output signals OFF. However, program operation does not stop. ○Eliminate the cause of the error and turn on power supply again.
M961	Initializing retentive memory	**Example**  X0 M961 To initialize retentive memory automatically at the start of operation, turn M967 ON for a single scan. Step No.1 ○In the system in this figure, retentive memory is or is not initialized depending on whether X0 is ON OFF at the start of operation. X0 = ON···Retentive memory is initialized when power is switched ON. X0 = OFF···Retentive memory is not initialized when power is switched ON. ○Retentive memory is initialized only at the start of operation. During operation, it is not initialized even if M961 is switched ON. ○M961 coil operates only when it is written in step 1. It is invalid when it is written in any other step.

Table 2-3 Function of Special Internal Output (2/3)

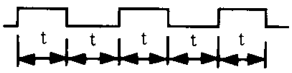
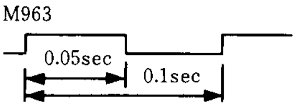
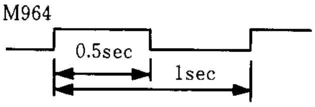
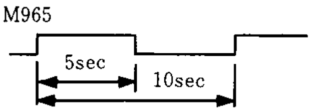
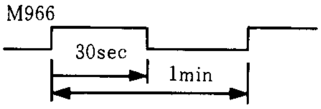
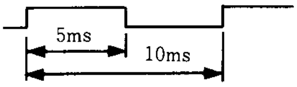
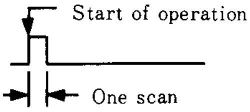
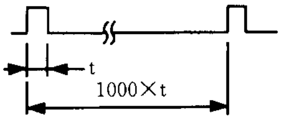
No.	Operation	Function
M962	Cyclic oscillation	 t : Period of one scan (scan time) Signal goes ON/OFF alternately for each scan.
M963	0.1 sec clock	 
M964	1 sec clock	
M965	10 sec clock	 
M966	1 min clock	
M969	10 ms clock	 In master station(St.NO.0)of ECL, this clock (10 ms clock)is not correct.
M967	ON for a single scan after start of operation	 To initialize all volatile memories at the start of operation, use M967 in combination with M961. To initialize memory individually, use M967 alone.
M968	1000-scan cycle	 t : Scan time ON once every 1000 scans. Used for measuring scan time.

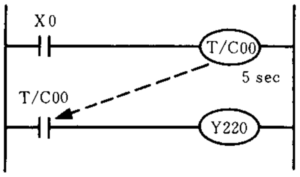
Table 2-3 Function of Special Internal Output (3/3)

No.	Operation	Function
WM970	System error factor	If system error occurs (when ERR lamp comes on), an error code within 0 to 65535 is displayed. The code cannot be cleared by turning on power supply again.
WM980	Syntax error factor	If syntax error is detected in the check specified by a peripheral or in the check before start of operation, an error code within 0 to 65535 is displayed. The code cannot be cleared by turning on power supply again.
WM982	Scan time	The latest scan time is indicated in steps of 10 ms, though the first scan is shown as 65535 ms. Indication contains an error of ±10 ms. Unit is millisecond (ms). (Indicated as 0, 10, 20 ms ...)
WM984	Max. scan time	Of scan times after the start of operation, the maximum time is displayed in steps of 10 ms, though the first scan is shown as 0 ms. Indication contains an error of ±10 ms. Unit is millisecond (ms). (Indicated as 0, 10, 20 ms ...)
M990	Normally ON	Always ON irrespective of run/stop status.
M991	ON during run	ON during run and OFF during stop

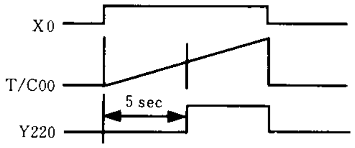
M972 through M977 are used by the system.
M978, M979 and M986 through M989 are for functional expansion and unused (undefined) by the system.

Timers : T/C00 to T/C95 (96 points shared with counter)

Example



Code		Remarks
ORG	X0	(1) Write a decimal point (.) before the preset value. (2) Write the I/O classification code "T/C"
OUT	T/C00.5	
ORG	T/C00	
OUT	Y220	



Time lapse is fetched in incremental mode.

The timer contact turns on when the current time value reaches the preset value.

[Explanation]

1. Kinds of timer

(1) On-delay timers are used. In the above sequence, the timer coil T/C00 is excited when input X0 turns ON. After 5 sec, the timer contact turns ON.
There are many timers with "a" and "b" contacts. They are used for generating a sequence in the PC.

(2) The same data area is shared by timers and counters, a total of 96 (T/C00through T/C95). A number used for a counter cannot be used for a timer.

2. Key input of timer

For specifying a timer coil using the programmer, enter the timer number (1 or 2 digits), decimal point (.) as a separator and the preset value in this order.

$$\boxed{\text{OUT}}\boxed{\text{T/C}}\boxed{\text{Timer No.}}\boxed{\cdot}\boxed{\text{Preset value}}$$

Decimal point for separating timer number and preset value

I/O classification number must not be omitted.

3. Time base

The timers have two time bases : 0.01 and 0.1 sec. Time base is automatically selected according to the key-in method.

Time base	Key-in method	Preset value range
0.1 sec	$\boxed{\text{OUT}}\boxed{\text{T/C}}\boxed{\text{Timer No.}}\boxed{\cdot}\underbrace{\boxed{6}\boxed{3}\boxed{\cdot}\boxed{5}}_{63.5\text{ sec}}$	T/C 0~ 9 ··· 0.1~999.9sec T/C10~95··· {0.1~ 99.9sec 1 ~999 sec
	$\boxed{\text{OUT}}\boxed{\text{T/C}}\boxed{\text{Timer No.}}\boxed{\cdot}\underbrace{\boxed{7}\boxed{7}\boxed{0}}_{770.0\text{ sec}}$	
0.01 sec	$\boxed{\text{OUT}}\boxed{\text{T/C}}\boxed{\text{Timer No.}}\boxed{\cdot}\underbrace{\boxed{0}\boxed{\cdot}\boxed{5}\boxed{5}}_{0.55\text{ sec}}$	T/C0~95 ··· 0.01~9.99 (settable only in 3 digits)

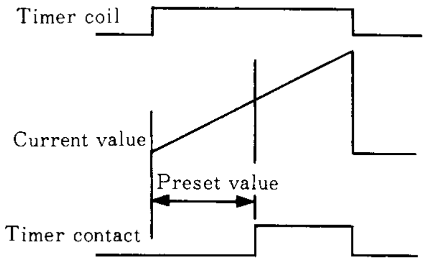
4. Preset value

Up to 10 timers/counters (T/C0 to T/C9) can be set using 4 digits. (The timer adopting 0.01 sec time base is excluded.)

Up to 86 timers/counters (T/C10 to T/C95) can be set using 3 digits.

5. Current value

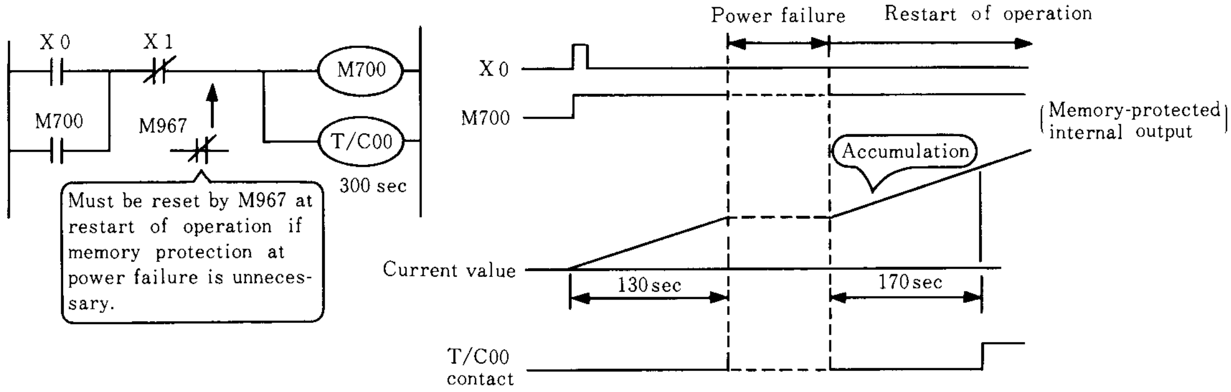
Each timer operates in the incremental mode. It starts timing when the timer coil is energized. When the current value reaches the preset value, the timer contact closes. When the timer coil is deenergized, the current value is reset to 0.



NOTE

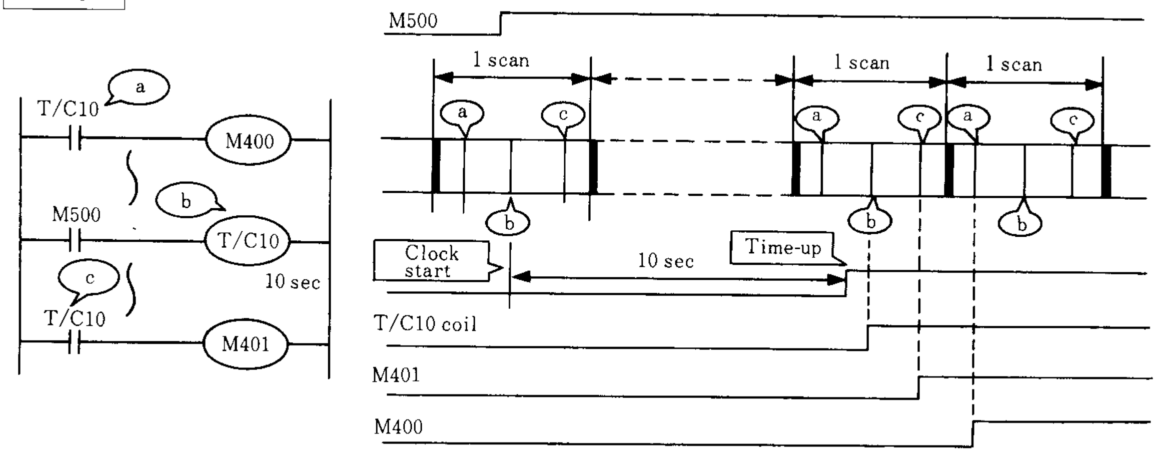
The current value of each timer is retained in memory even if power failure occurs or when power supply is turned off. When combining the timer with the retentive internal output, an accumulation timer can be composed.

Example


X 0
X 1
M700
M967
M700
T/C00
300 sec
Must be reset by M967 at restart of operation if memory protection at power failure is unnecessary.
Power failure
Restart of operation
X 0
M700
Memory-protected internal output
Accumulation
Current value
130sec
170sec
T/C00 contact

6. Contact operation timing chart and accuracy

Example



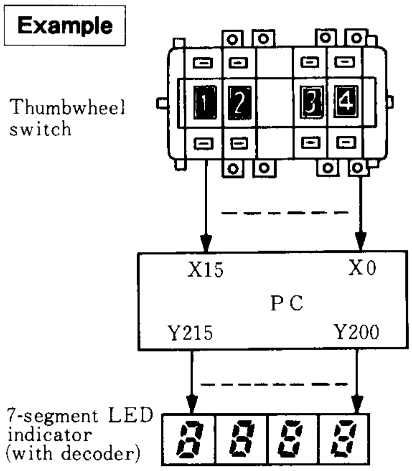
The clock starts when the timer coil is energized (time point (b)). When the coil instruction is executed after time-up., the output contact closes.

Condition	Timer starts by other than external input signal.		Timer starts by external input signal.	
	Timer contact (a) before coil	Timer contact (b) after coil	Timer contact (a) before coil	Timer contact (b) after coil
Timer accuracy	+2 scans	+1 scan	Input fetch delay (4 ms single scan filter) +2 scans	Same as at left +1 scan

Total timer accuracy
Preset time +2 scans − time base

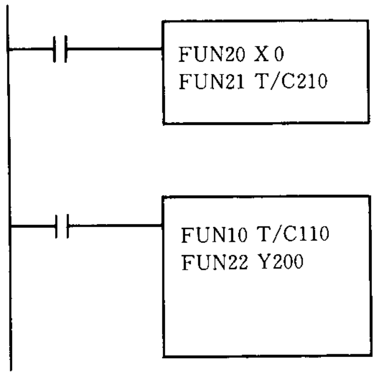
7. Handling timer preset value and current value in arithmetic instructions in application

Example



The preset value of a timer can be changed by using the thumbwheel switch, and the current value of a timer can be read on the 7-segment LED indicator. An example of program is shown below. The table below lists the number assignment when using the timer preset value and current value in the arithmetic operation.

Segment	Assignment No.	Remarks
Curent value	T/C100~T/C195	Add 100 to timer coils T/C00 to T/C95.
Preset value	T/C200~T/C295	Add 200 to timer coils T/C00 to T/C95.



X0 through X15(thumbwheel switch data) → AR
AR → timer T/C preset value
} The timer preset value can be changed by using the thumbwheel switch.

Current value of T/C10 timer → AR
AR → Y200 through 215
} The timer current value is read on the indicator.

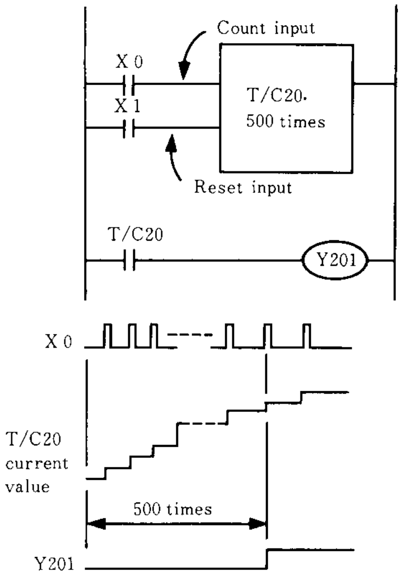
AR : Arithmetic Register

The timer preset value and current value are data to be processed in blocks of 16 bits as shown below.

Segment	Kind of timer	Data to be processed by arithmetic instruction
Preset value and current value	0.1 sec timer	b_{15} b_0 $\underbrace{\boxed{2}\boxed{6}\boxed{4}\boxed{5}}$ Indicates 264.5 sec. ……BCD 4 digits The least significant digit represents 0.1sec order.
	0.01 sec timer	b_{15} b_0 $\underbrace{\boxed{F}\boxed{0}\boxed{5}\boxed{5}}$ Indicates 0.55 sec. ……BCD 3 digits The most significant digit stands for "F" (0.01sec timer). The least significant digit represents 0.01sec order.

Counter:T/C00 through T/C95 (selectable to function as timer or counter, total of 96)

Example



The current value is incremented.

The counter contacts close when the current value reaches the preset value.

Code		Remarks
ORG	X0	(1) The reset input is given in the form of STR instruction. (2) Place a decimal point (.) before the preset value. (3) Use the I/O classification code "T/C."
STR	X1	
OUT	T/C20.500	
ORG	T/C20	
OUT	Y201	

[Explanation]

1. Kind of counter

(1) An up-counter is used. In the above sequence, the counter T/C20 counts ON/OFF cycles of input X0.When the count reaches 500, the counter contacts close.

The counters can be provided with any number of "a" and "b" contacts. They are used for generating sequences in the PC.

(2) Timers and counters share the same data area. There are 96 timers/counters in total (T/C00 through T/C95).

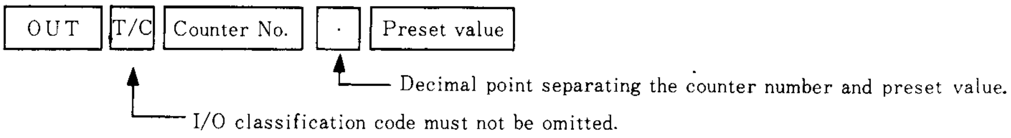
Once a T/C number is assigned to a timer, it cannot be reused for a counter.

(3) When the reset input turns ON, the counter is reset and the current value returns to 0.

2. Counter key input

(1) Program the count input and reset input in this order. Reset input must be programmed by an STR instruction. Reset input discriminates the counter from the timer.

(2) A counter preset value can be entered in the same way as for a timer.


OUT
T/C
Counter No.
·
Preset value
Decimal point separating the counter number and preset value.
I/O classification code must not be omitted.

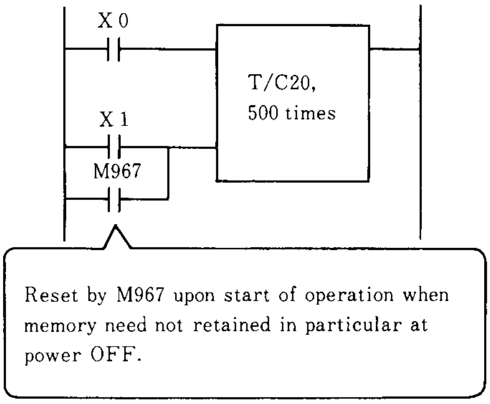
3. Preset value

Up to 10 timers/counters (T/C0 to T/C9) can be set using 4 digits.
Up to 86 timers/counters (T/C10 to T/C95) can be set using 3 digits.

4. Current value

The current value of each counter is incremented by 1 (one) whenever the count input turns from OFF to ON. The counter contacts close when the current value reaches the preset value.
When the reset input turns ON, the current value is reset to 0.
The current value of the counter is retained in memory even if power is turned OFF.

Example


X0
X1
M967
T/C20,
500 times
Reset by M967 upon start of operation when memory need not retained in particular at power OFF.

If the retentive data is unnecessary, use the special internal output M967, which turns on a single scan at start of operation. Program as shown at left.

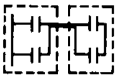
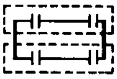
5. Handling the counter preset value and current value in the arithmetic instructions

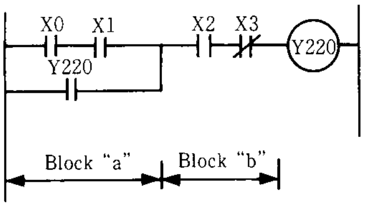
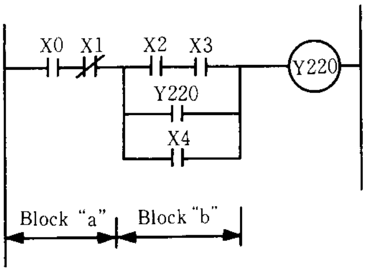
When using a combination of counter preset value and current value in arithmetic instructions, the current value should be equal to the counter coil number (T/C0 through 95) incremented by 100, namely T/C100 to T/C195. The preset value should be equal to the coil number incremented by 200, namely T/C200 to T/C295.
The counter preset value and current value are 16-bit data (4-digit BCD value) and processed as shown in the table below.

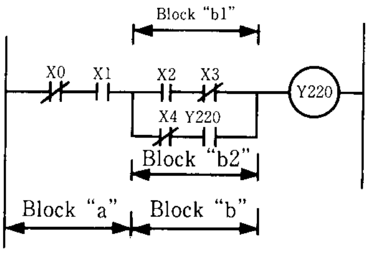
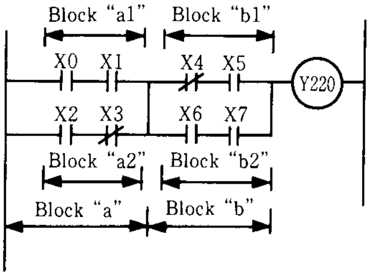
Item	Assignment No.	Data for processing of arithmetic instruction
Current value	T/C100 through T/C195 (equal to counter coil numbers T/C0 to T/C95 incremented by 100)	b_{15} b_0 | 3 | 4 | 5 | 6 | ······4-digit BCD Indicates 3456 times.
Preset value	T/C200 through T/C295 (equal to counter coil numbers T/C0 to T/C95 incremented by 200)	

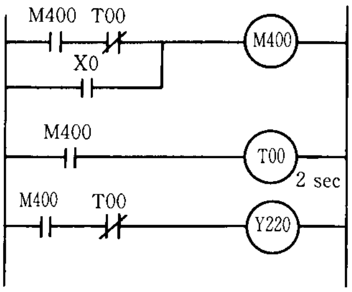
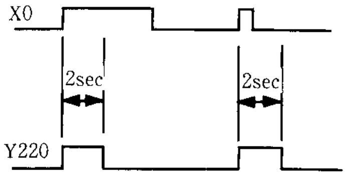
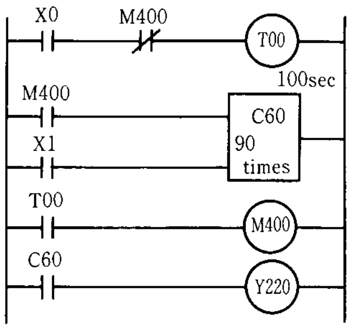
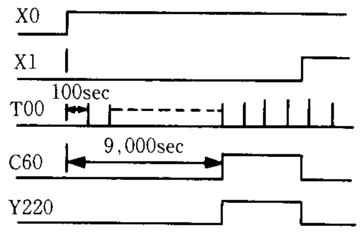
3. Programming

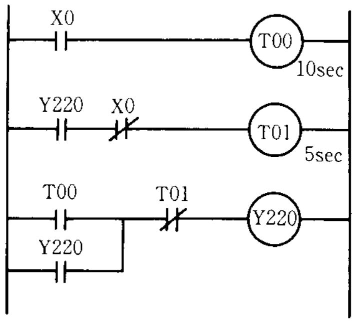
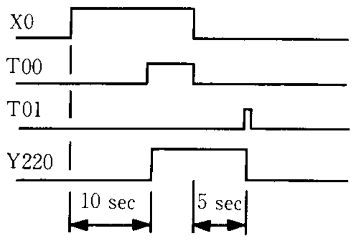
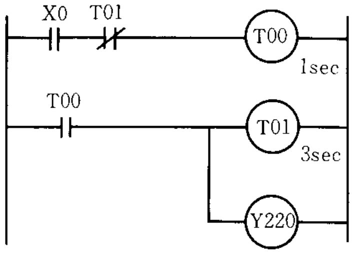
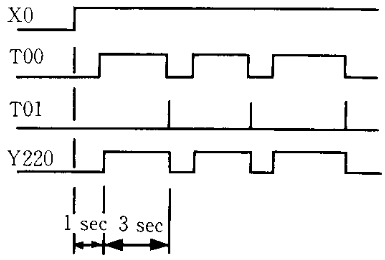
Table 3-1 Basic Instructions

Instruction	Symbol	Function	Component used	No.of words	Change in register AR	ER	C	Acc
ORG	├┤├─	Connection of normally open contacts ("a"contacts) to bus	X,Y,M T/C 0~T/C 95	1	·	·	·	↕
ORG NOT	├┤/├─	Connection of normally closed contacts ("b"contacts) to bus		1	·	·	·	↕
STR	├┤├─	Start of branching normally open contacts ("a"contacts)	X,Y,M T/C 0~T/C95	1	·	·	·	↕
STR NOT	├┤/├─	Start of branching normally closed contacts ("b"contacts)		1	·	·	·	↕
AND	─┤├─	Serial connection of normally open contacts ("a"contacts)	X,Y,M T/C 0~T/C95	1	·	·	·	↕
AND NOT	─┤/├─	Serial connection of normally closed contacts ("b"contacts)		1	·	·	·	↕
OR	─┤├┘	Parallel connection of normally open contacts ("a"contacts)	X,Y,M T/C 0~T/C95	1	·	·	·	↕
OR NOT	─┤/├┘	Parallel connection of normally closed contacts ("b"contacts)		1	·	·	·	↕
AND STR		Serial connection of logic block	None	1	·	·	·	↕
OR STR		Parallel connection of logic block		1	·	·	·	↕
OUT	─○┤	Output of calculation result	Y,M T/C0~T/C95 (with preset value)	1	·	·	·	·
OUT NOT	─⊘┤	Inverted output of calculation result	Y,M	1	·	·	·	·

Circuit	Configuration	Program Instruction Code	Program Data	Explanation
Parallel-to-serial circuit	 X0 X1 X2 X3 Y220 Y220 Block "a" Block "b"	ORG	X 0	First the parallel circuit of block "a" and then the serial circuit of block "b" are programmed.
		AND	X 1	a
		OR	Y220	
		AND	X 2	
		AND NOT	X 3	b
		OUT	Y220	
Serial-to-parallel circuit	 X0 X1 X2 X3 Y220 Y220 X4 Block "a" Block "b"	ORG	X 0	a The circuit is divided into blocks "a" and "b" which are programmed separately.
		AND NOT	X 1	
		STR	X 2	
		AND	X 3	
		OR	Y220	b
		OR	X 4	
		AND STR		a · b Blocks "a" and "b" are combined by AND STR.
		OUT	Y220	

Circuit	Configuration	Program		Explanation	
		Instruction Code	Data		
Serial-to-parallel circuit	 Block “b1” X0 X1 X2 X3 Y220 X4 Y220 Block “b2” Block “a” Block “b”	ORG NOT	X 0	a	Block “a” is programmed.
		AND	X 1		
		STR	X 2	b1	Block “b1” is programmed.
		AND NOT	X 3		
		STR NOT	X 4	b2	Block “b2” is programmed.
		AND	Y220		Blocks “b1” and “b2” are combined by OR STR.
		OR STR		b1+b2	
		AND STR		a · b	Blocks “a” and “b” are combined by AND STR.
		OUT	Y220		
Serial connection of parallel circuits	 Block “a1” Block “b1” X0 X1 X4 X5 Y220 X2 X3 X6 X7 Block “a2” Block “b2” Block “a” Block “b”	ORG	X 0	a1	First block “a1” and then block “a2” are programmed.
		AND	X 1		
		STR	X 2	a2	
		AND NOT	X 3		These blocks are combined by OR STR.
		OR STR		a1+b2	
		STR NOT	X 4	b1	
		AND	X 5		Block “b1” and “b2” are programmed in the same way as above.
		STR	X 6	b2	
		AND	X 7		
		OR STR		b1+b2	
		AND STR		a · b	Blocks “a” and “b” are combined by AND STR.
		OUT	Y220		

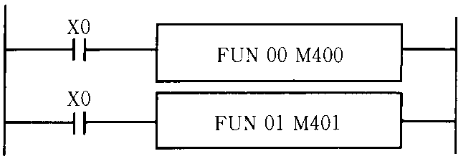
Circuit		Configuration	Program		Explanation
			Instruction Code	Data	
Timer/counter application circuit	One-shot circuit	 M400 T00 M400 X0 M400 T00 2 sec M400 T00 Y220	ORG	M400	
			AND NOT T/C	00	
			OR	X0	
			OUT	M400	
			ORG	M400	
			OUT T/C	00.002	
			ORG	M400	
			AND NOT T/C	00	
			OUT	Y220	
	Timer and counter circuits	 X0 M400 T00 100sec M400 C60 90 times X1 T00 M400 C60 Y220	ORG	X0	
			AND NOT	M400	
			OUT T/C	00.100	
			ORG	M400	
			STR	X1	
			OUT T/C	60.090	
			ORG T/C	00	
			OUT	M400	
			ORG T/C	60	
			OUT	Y220	

Circuit		Configuration	Program		Explanation
			Instruction Code	Data	
Timer/counter application circuit	ON/OFF delay circuit		ORG OUT T/C ORG AND NOT OUT T/C ORG T/C OR AND NOT T/C OUT	X0 00.010 Y220 X0 01.005 00 Y220 01 Y220	
	Flicker circuit		ORG AND NOT T/C OUT T/C ORG T/C OUT T/C OUT	X0 01 00.001 00 00.003 Y220	

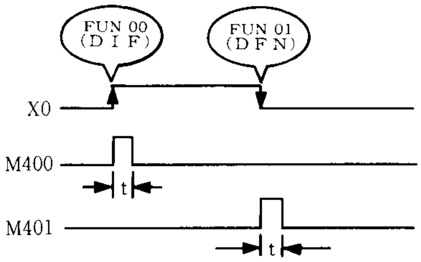
Application Instructions (I)

Edge	Set, Reset and Step Process	Master control	Jump	UP/down counter	Latch	Shift register	NOP and end

Instruction	Symbol	Name	Function	Component	No. of words	Change in register			
						AR	ER	C	Acc
FUN00	DIF	Rising edge	Detects rising edge (_↑‾) of signal.	M	1	·	·	·	·
FUN01	DFN	Trailing edge	Detects trailing edge (‾↓_) of signal.	M	1	·	·	·	·



Code		Remarks
ORG	X0	
FUN00	M400	Detects rising edge.
ORG	X0	
FUN01	M401	Detects trailing edge.

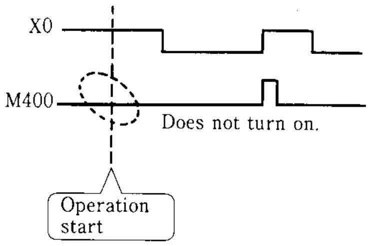


t : one(1) scan time

[Explanation]

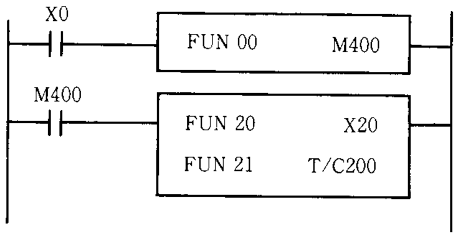
1. The FUN00 (DIF) is used to detect the rising edge of an input signal (status change from LOW to HIGH), and the FUN01 (DFN) is used to detect the trailing edge of the signal (status change from HIGH to LOW). These instructions are programmed in combination with an internal output (M) so that the specified internal output (M) turns on only for 1 scan time when the edge is detected. Any number of FUN00 and FUN01 instructions can be used (so far as internal output permits).

2. The edge detect instructions are executed according to the input change after operation start.



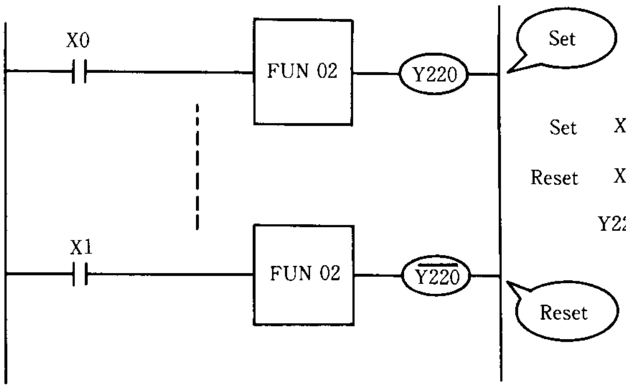
Rising edge is not detected and M400 does not turn on when X0 is already turned on at start of operation.

3. The edge detect function is effective for word LOAD and COMPARE instructions,which are executed only when input conditions change (For instance,it is used to start—up conditions of arithmetic instructions as the ladder diagram shown left).

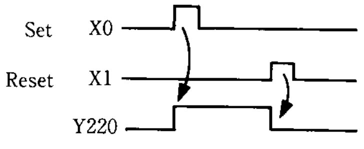


Instruction	Symbol	Name	Function	Component	No. of words	Change in register			
						AR	ER	C	Acc
FUN02	IF	If	Set/reset	None	1	·	·	·	·
FUN03	IFR	If reset	Step process		1	·	·	·	·

Set/reset



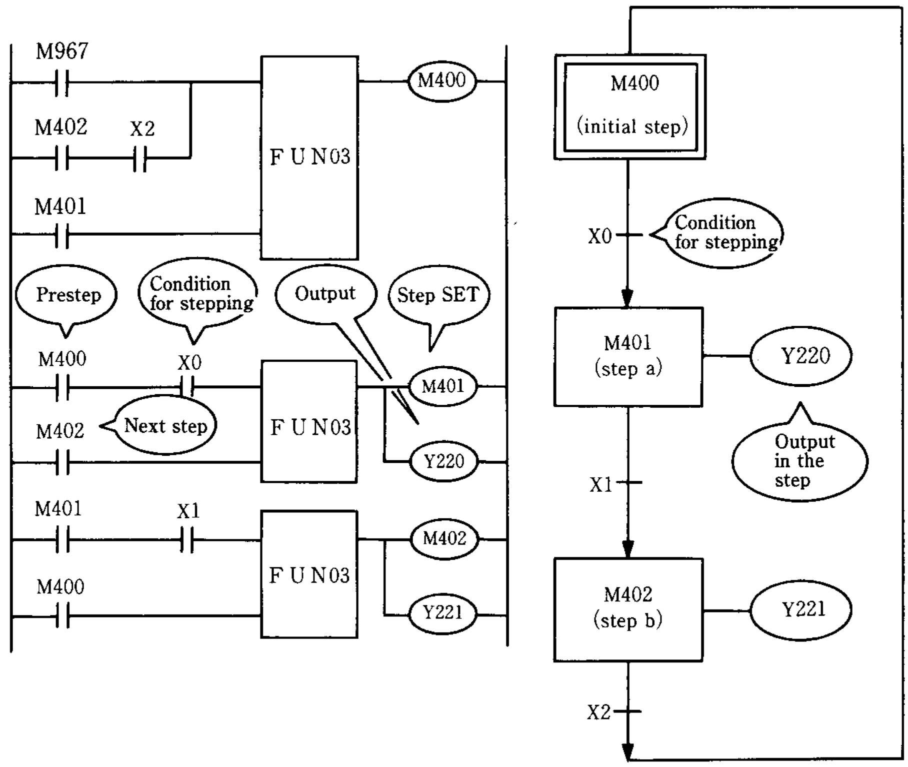
● Timing chart



● Coding

```
ORG        X0
FUN 02
OUT        Y220

ORG        X1
FUN 02
OUT NOT    Y220
```

Step processing



● Status transition diagram

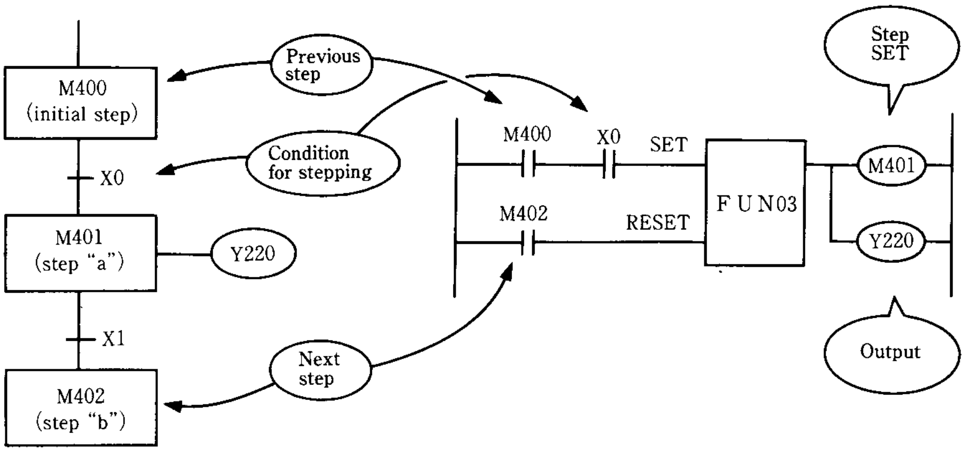
● Coding

```
ORG    M967
STR    M402
AND      X2
OR STR
STR    M401
FUN  03
OUT    M400
ORG    M400
AND      X0
STR    M402
FUN  03
OUT    M401
OUT    Y220

ORG    M401
AND      X1
STR    M400
FUN  03
OUT    M402
OUT    Y221
```

[Explanation]

1. Instructions FUN02 and OUT are combined and used as the SET instruction.
   Instructions FUN02 and OUT NOT are combined and used as the RESET instruction.
   ○ON status is held under SET input and OFF status is held under RESET input.
   ○Any other program may be inserted between SET coil and RESET coil. The program written last is given the highest priority.
   ○A keep relay can be composed when combining a FUN02 instruction with the memory-protected internal output.

2. FUN03 is the step process (sequential control) instruction. Set input and reset input are provided. A step process program can be created in the regular format using the status transition diagram.



Status transition diagram

[Explanation of operation]

(1) If step condition X0 is set to ON in the initial step (M400),step "a" (M401) turns ON and Y220 is output.

(2) Y220 continues its output even when step condition X0 is set to OFF.

(3) When step condition X1 is set to ON,step "b" turns ON and Y220 is set to OFF.

(4) Even when step condition X1 is set to ON in the initial step (M400) ,step "b" (M402) won't turn ON. All steps are executed in correct sequence.

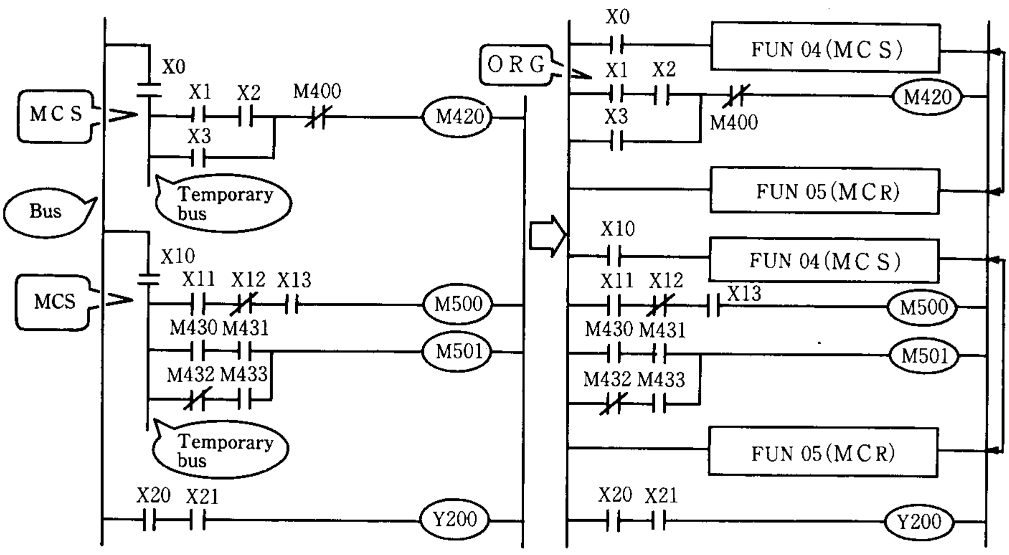
[Programming method]

(1) To program a FUN03 SET input,the internal output (M400) specifying the previous step is ANDed with the condition for stepping (X0).

(2) For FUN03 RESET input,the internal output (M402) specifying the next step is programmed.

(3) After FUN03,the internal output (M401) specifying the current step and output (Y220) are programmed.

Instruction	Symbol	Name	Function	Component	No. of words	Change in register			
						AR	ER	C	Acc
FUN04	MCS	Master control	Sets common serial contacts.	None	1	·	·	·	·
FUN05	MCR		Resets common serial contacts.		1	·	·	·	·

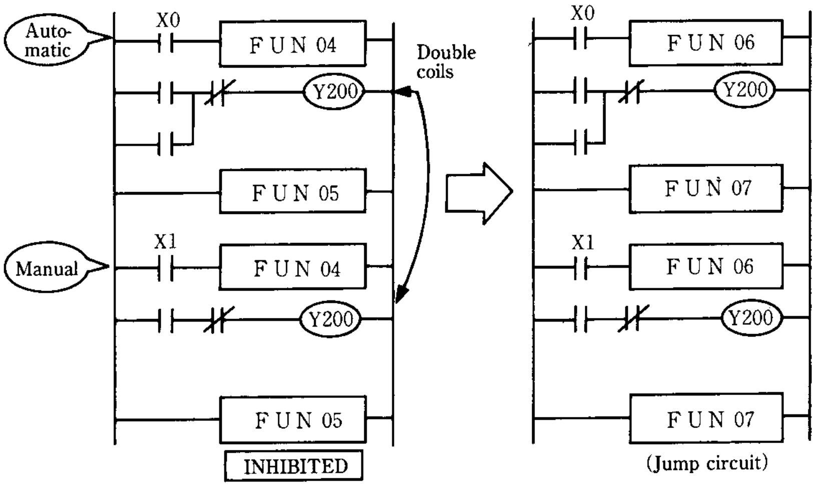


Coding

```
ORG          X0
FUN04
ORG          X1
AND          X2
OR           X3
AND NOT    M400
OUT        M420
FUN05
ORG         X10
FUN04
ORG         X11
AND NOT     X12
AND         X13
OUT        M500
ORG        M430
AND        M431
STR NOT    M432
AND        M433
OR STR
OUT        M501
FUN05
ORG         X20
AND         X21
OUT        Y200
```

[Explanation]

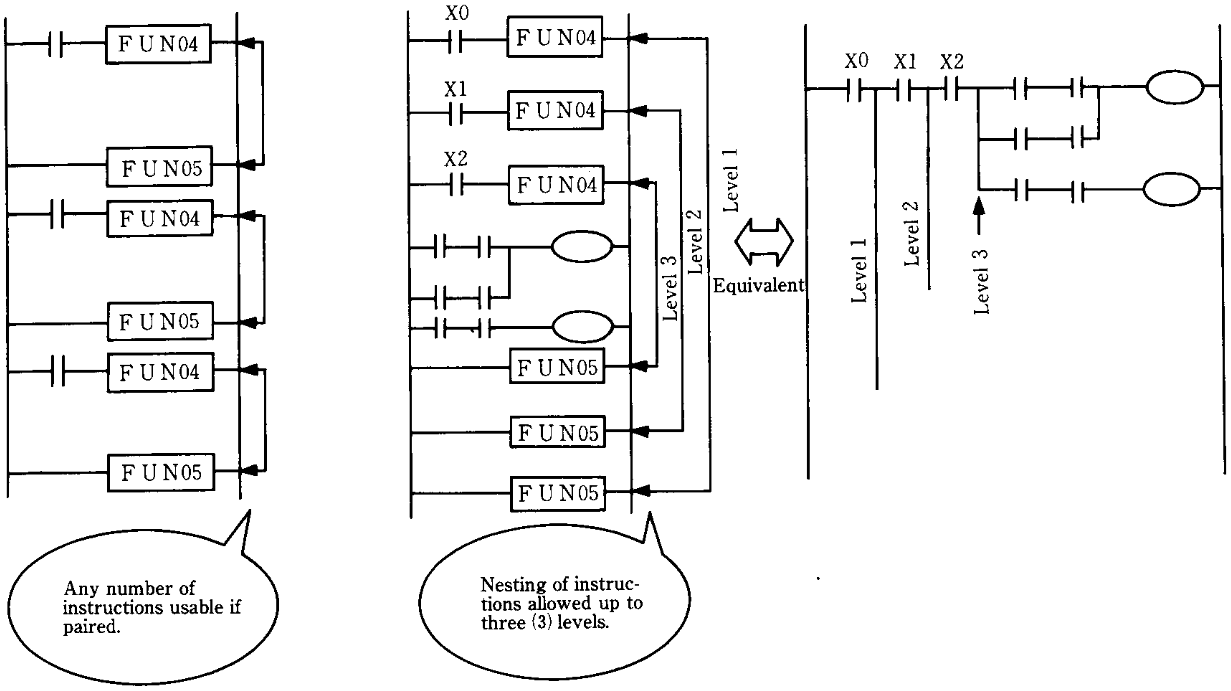
1. The FUN04 (MCS) and FUN05 (MCR) instructions are used for setting and resetting the common serial contacts,respectively. They must always be used as a pair. Otherwise a syntax error occurs.
2. The FUN04 instruction must be followed by an ORG (or ORG NOT) instruction.
3. When the master control contact is OFF,the subsequent output coil is set to OFF. In the example above,M420 is unconditionally OFF if input X0 is OFF.



NOTE

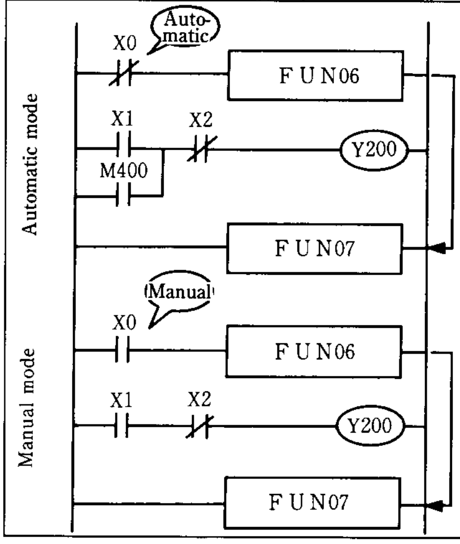
For swithover between manual and automatic controls according to a master control instruction, take care not to use double coils. If double coils must be used, specify the FUN06 or FUN07 instruction. This avoids error occurrence in syntax check. Refer page 46 to use FUN06 and FUN07.

4. Any number of master control instructions can be used if they are paired unless nesting.

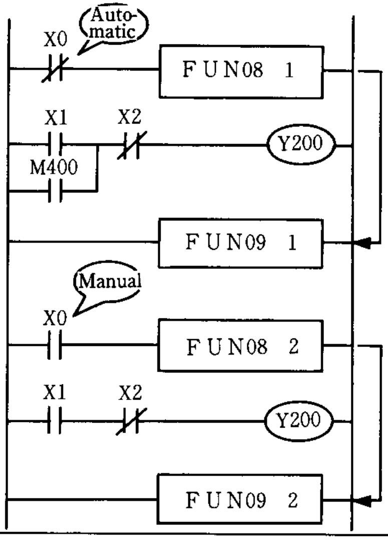


Instructions can be nested up to three (3) levels. At four (4) levels or more,syntax error will occur.

Instruction	Symbol	Name	Function	Component	No. of words	Change in register			
						AR	ER	C	Acc
FUN06	JMP	Jump without addressing	Skips program till JEND.	None	1	·	·	·	·
FUN07	JEND				1	·	·	·	·
FUN08	AJMP	Jump with addressing	Jumps to AJEND of corresponding address number.	Address No. (0 to 63)	2	·	·	·	·
FUN09	AJEND				2	·	·	·	·



```
ORG NOT   X0
FUN06

ORG       X1
OR      M400
AND NOT   X2
OUT     Y200

FUN07

ORG       X0
FUN06

ORG       X1
AND NOT   X2
OUT     Y200

FUN07
```

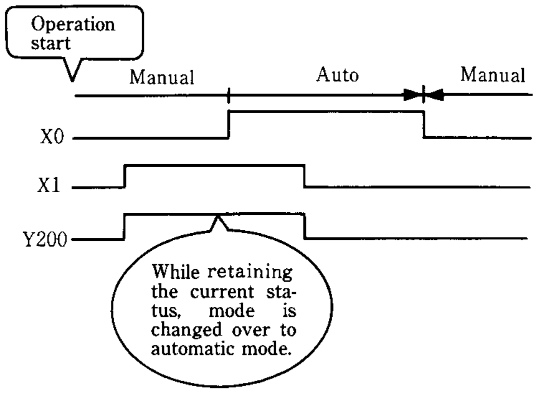


```
ORG NOT   X0
FUN08      1

ORG       X1
OR      M400
AND NOT   X2
OUT     Y200

FUN09      1

ORG       X0
FUN08      2

ORG       X1
AND NOT   X2
OUT     Y200

FUN09      2
```

[Explanation]

1. The FUN06 and FUN07 instructions specify jump without addressing, while the FUN08 and FUN09 instructions specify jump with addressing. These instructions all cause control to jump to JUMP END when the jump condition is set to ON.

2. When the jump conditions are satisfied, the program lines located between the current address and destination address are not executed. The output is held at the status before the jump.By using this function, a manual/auto switching circuit can be composed as illustrated above. If the same output coil is programmed between the jump circuits, a syntax error(double coil error E.)occurs, but operation can continue.



NOTE

If the jump conditions are satisfied, the timer in the jump circuit stops operating.
It restarts when the jump conditions are reset.

3. A jump instruction cannot be used between master control instructions.

4. The table below lists differences between the FUN06/07 instructions and FUN08/09 instructions. Scan time can be shortened by using the functional combination of FUN08/09.

Table 3–2 : Differences between FUN06/07 and FUN08/09 (1/2)

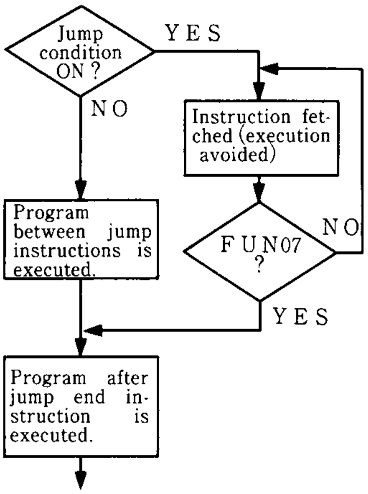
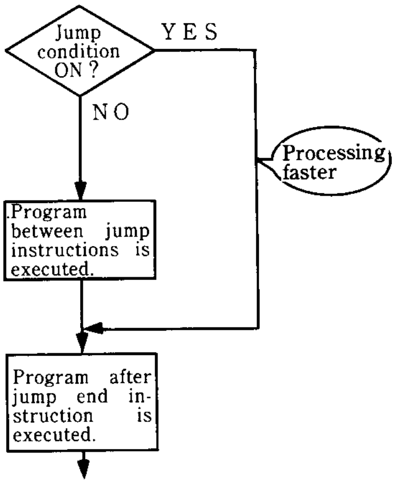
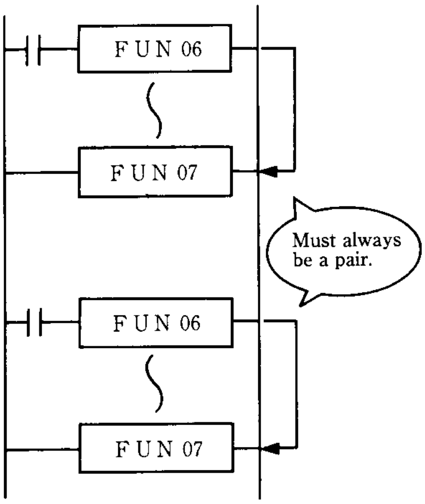
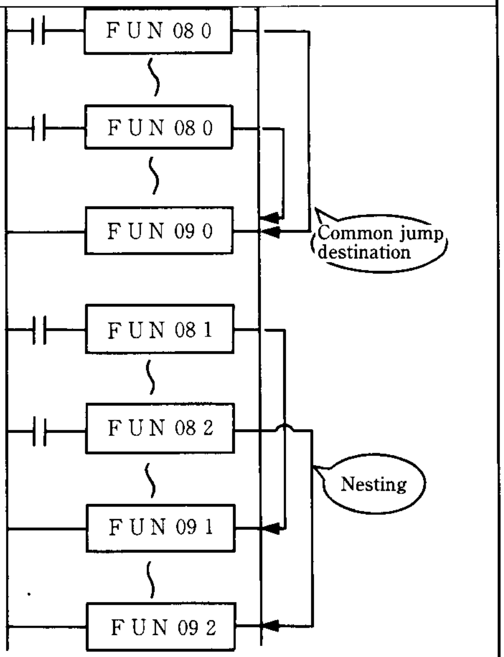
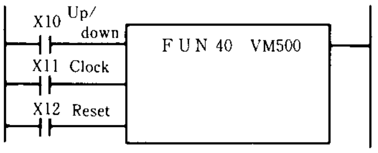
Item	FUN 06, FUN 07	FUN 08, FUN 09
Method of processing instruction	 Jump condition ON ? YES NO Instruction fetched (execution avoided) Program between jump instructions is executed. FUN07 ? NO YES Program after jump end instruction is executed.	 Jump condition ON ? YES NO Processing faster Program between jump instructions is executed. Program after jump end instruction is executed.

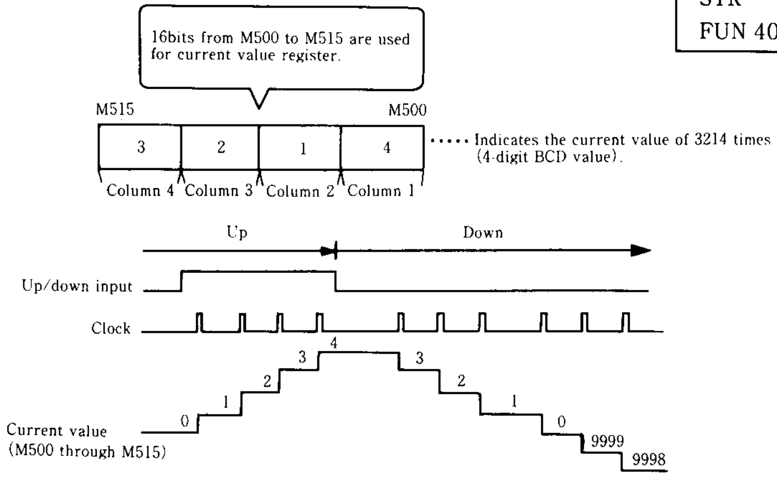
Table 3–2 : Differences between FUN06/07 and FUN08/09 (2/2)

Item	FUN 06, FUN 07	FUN 08, FUN 09
Jump method	 FUN 06 FUN 07 Must always be a pair. FUN 06 FUN 07 1. These instructions must always be used as a pair. If not paired, a syntax error will occur. 2. Nesting is unallowable.	 FUN 08 0 FUN 08 0 FUN 09 0 Common jump destination FUN 08 1 FUN 08 2 Nesting FUN 09 1 FUN 09 2 1. Jump from multiple FUN08 instructions to a single FUN09 instruction is allowed. 2. Nesting is allowed at different addresses 3. Jump to a preceding step is also possible.

Instruction	Symbol	Name	Function	Component used	No. of words	Change in register			
						AR	ER	C	Acc
FUN40	UDC	Up/down counter	Up/down counter	VM	1	•	•	•	•

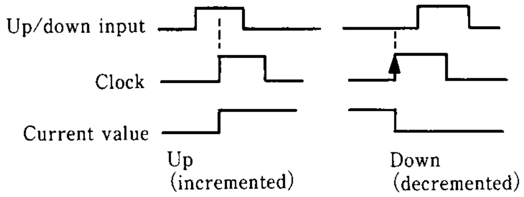


Code		Remarks
ORG	X10	
STR	X11	Use STR.
STR	X12	
FUN 40	VM500	UDC


16bits from M500 to M515 are used for current value register.
M515
M500
3
2
1
4
••••• Indicates the current value of 3214 times (4-digit BCD value).
Column 4
Column 3
Column 2
Column 1
Up
Down
Up/down input
Clock
Current value (M500 through M515)
0
1
2
3
4
3
2
1
0
9999
9998

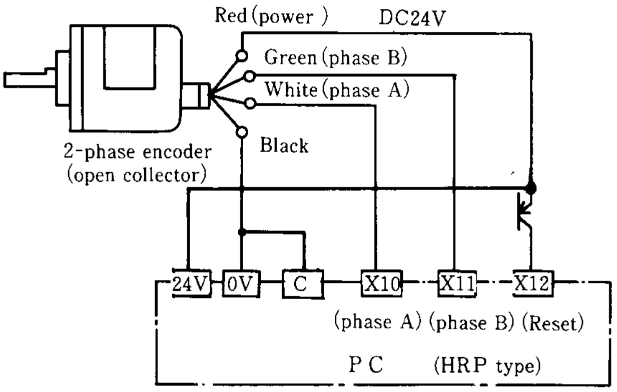
[Explanation]

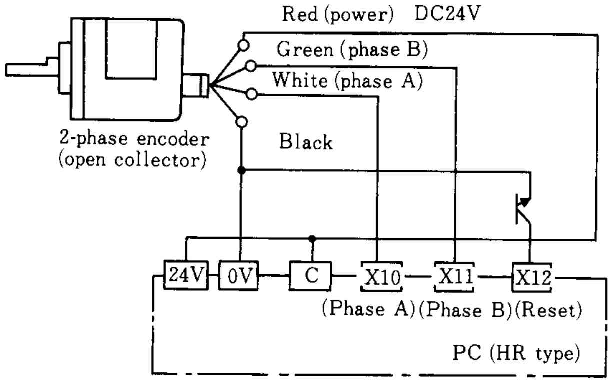
1. FUN40(UDC)is the up/down counter instruction. It is to be programmed in combination with an internal output(VM).
2. 16 bits starting from the coil number specified by that instruction(M500 through M515 in the example shown above)are used as the current value register of up/down counter.The current value is presented in BCD 4 digits.
3. The up/down input, clock input and reset input are programmed in that order.
The current value changes at the rising edge of the clock(from OFF to ON).Either UP or DOWN condition is selected according to the ON or OFF status of up/down input as shown below.



4. When the reset input is set to ON, the current value is reset to zero.
5. An example of PC in shown below. Connection to encoder

Connection to encoder





An incremental type encoder with open collector output can be connected. The figure at left exemplifies connection to the Omron E6A-CW4C.

NOTE

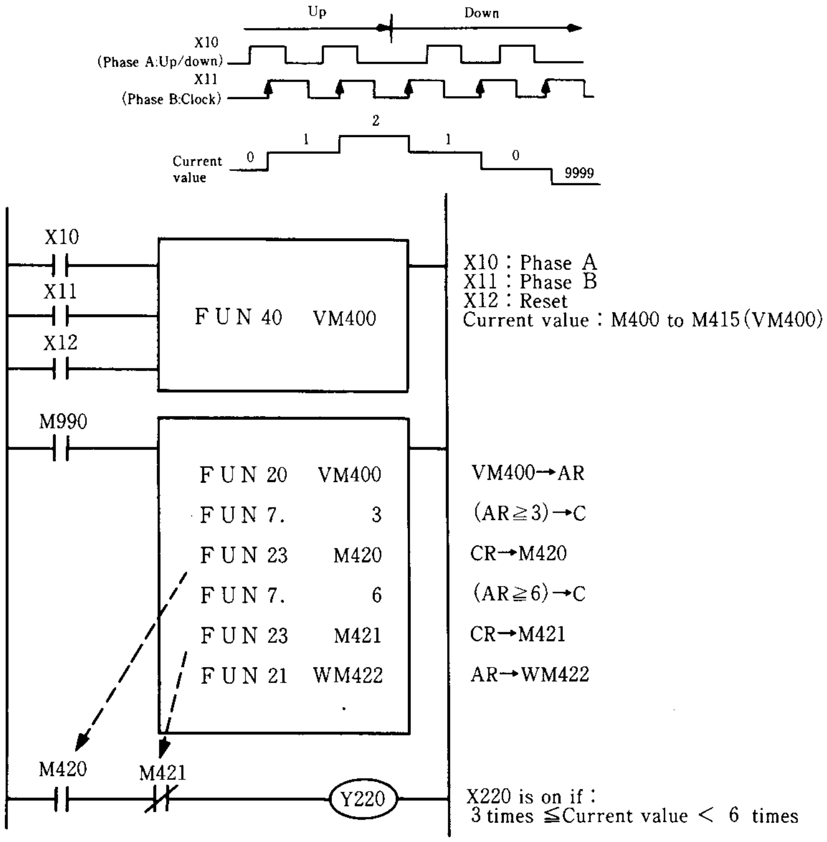
A pulse shorter than the scan time cannot be measured because the FUN 40 (UDC) instruction measures the pulse count by using the software built in the PC. When measuring pulses of a frequency exceeding approximately 50 Hz, it is requested to use the high speed counter instruction (FUN96) detailed later.

[Application Example]

The figure at right shows a sequence under connection with the above-mentioned encoder, in which the Y220 external output is turned ON only when the current value of the up/down counter is between 3 to 6.

[Monitor]

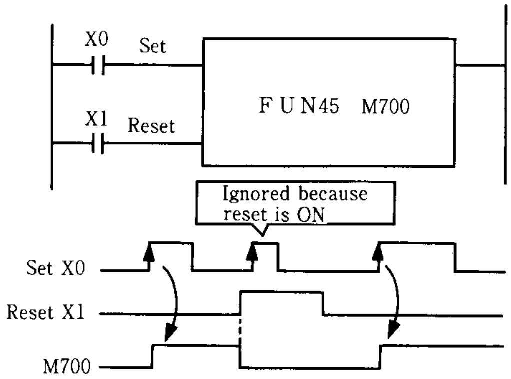
Key-in sequence for monitoring is shown right.



※The current value can be checked by monitoring WM422.

Key-in sequence CLR OUT 4 2 2 MON MON MON

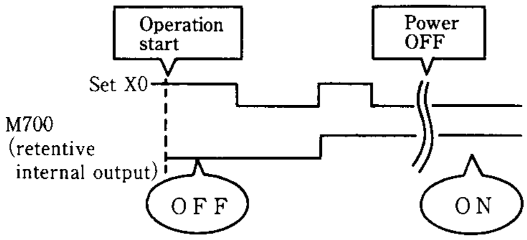
Instruction	Symbol	Name	Function	Component	No. of words	Change in register			
						AR	ER	C	Acc
FUN45	LATCH	Latch	Reset priority latch	M	1	·	·	·	·



Code		Remarks
ORG	X0	Use STR for Reset input.
STR	X1	
FUN45	M700	

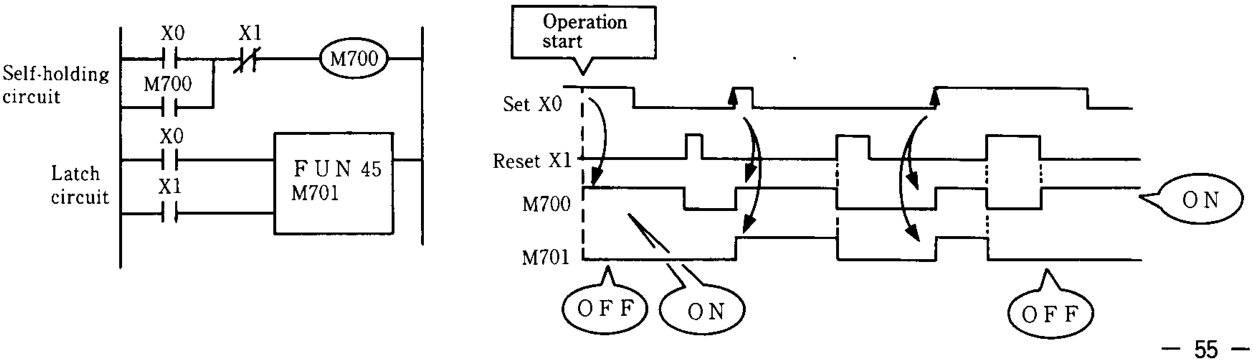
[Explanation]

1. FUN45 (LATCH) is an edge triggered latch instruction with the reset priority signal. It should be programmed in combination with an internal output(M).
2. The ON status is set at the rising edge of the set input signal (from OFF to ON). The OFF status is set when the reset input goes ON. When the reset input is ON, the set input is rejected. If the set input and reset input go ON simultaneously, the reset input takes priority.
3. The FUN45 instruction can be combined with a retentive memory internal output(M) to produce the function of a keep relay.

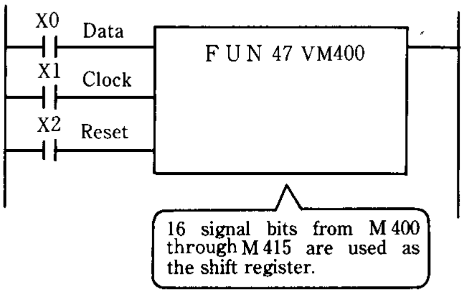


In the above sequence, the status of M700 at occurrence of power interruption is retained till its recovery because M700 is a retentive internal output.

4. The self-holding circuit operates at a specific level (ON or OFF status), but the latch is operated at the signal edge. This causes the difference shown below.

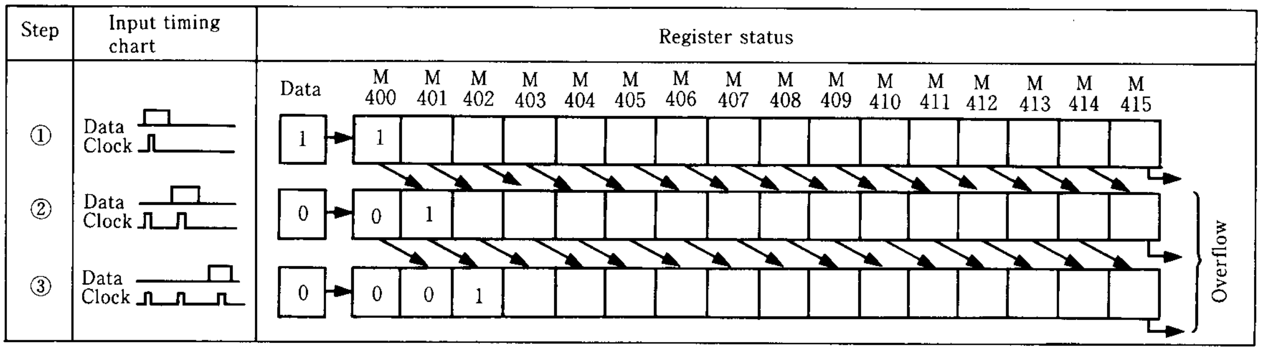


Instruction	Symbol	Name	Function	Component	No. of words	Change in register			
						AR	ER	C	Acc
FUN47	SFR	Shift register	16-bit shift register	VM	1	•	•	•	•



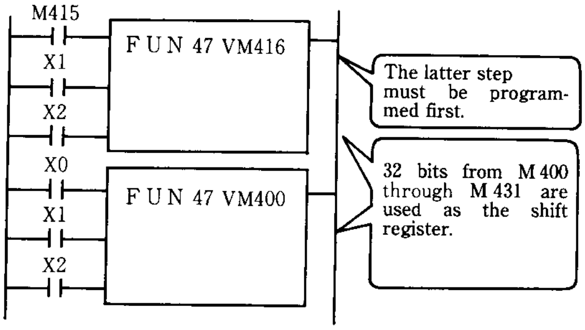
16 signal bits from M 400 through M 415 are used as the shift register.

Code		Remarks
ORG	X0	
STR	X1	Use STR.
STR	X2	
FUN 47	VM400	



[Explanation]

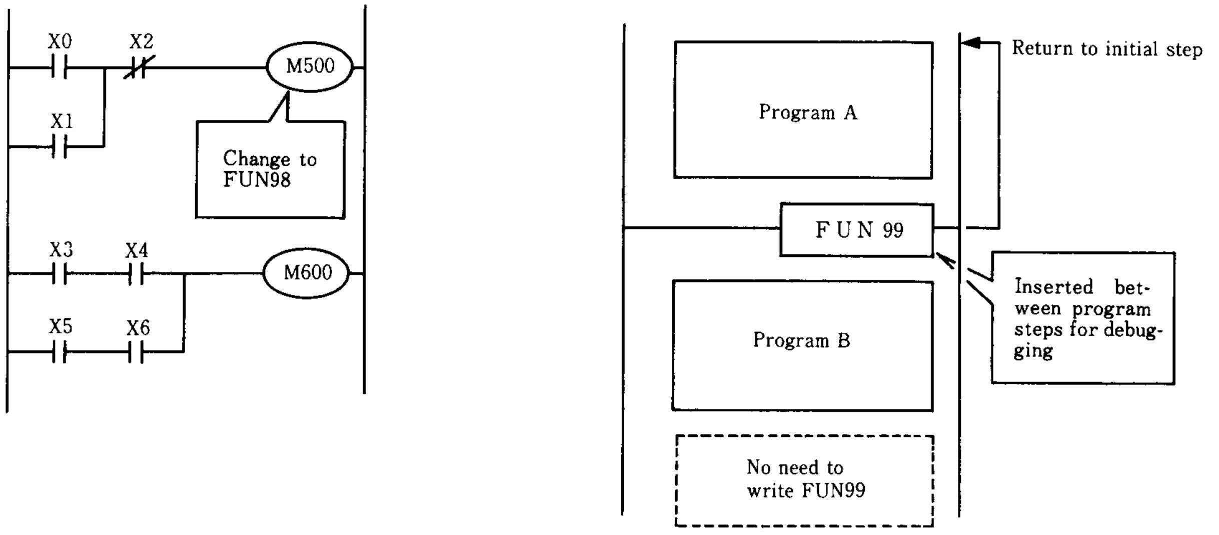
1. FUN47(SFR) is the shift register instruction. It is to be programmed in combination with internal output(VM).
2. 16 bits(M400 through M415 in the example shown above), from the coil number specified by the FUN47 instruction and higher, are used as the register.
3. When the clock input rises(from OFF to ON), concurrent data input(ON/OFF status)is written in the least significant bit of the register(M400 in this example). The ON/OFF status of each register is shifted to the next high-order bit synchronized with the rise of clock input.
4. Data of the most significant bit(M415 in this example)may overflow as a result of shift operation. When connecting two(2)or more shift registers, the latter step(with a larger I/O number)must be programmed first in order to prevent data being lost due to overflow.



The latter step must be programmed first.

32 bits from M 400 through M 431 are used as the shift register.

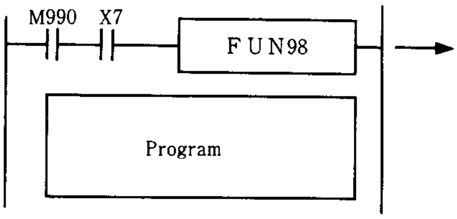
Instruction	Symbol	Name	Function	Component	No. of words	Change in register			
						AR	ER	C	Acc
FUN98	NOP	NOP	No operation	None	1	•	•	•	•
FUN99	END	END	Return to initial step	None	1	-	-	-	-



[Explanation]

1. FUN98 is NOP instruction. This instruction does not cause any execution in its step and hands over control to the next step.When this instruction is written instead of OUT instruction, output is avoided in the relevant circuit.

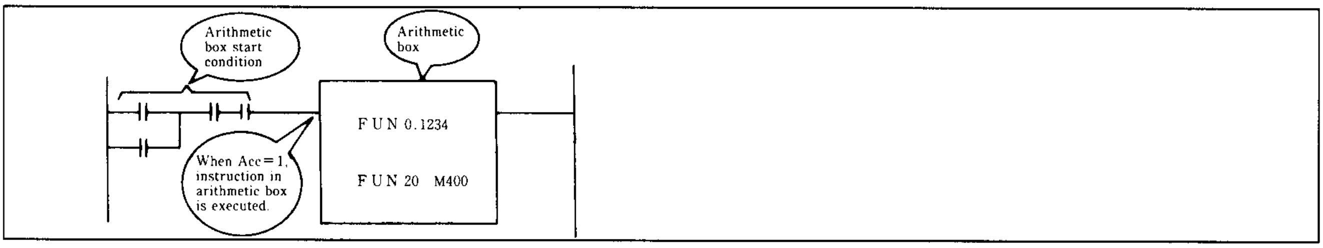
2. In the EM series, FUN98 serves as a start instruction. Although an EM program can be run in the EC series, its start circuit becomes meaningless(because operation is controlled via the exclusive start terminal in the EC series).



Functions as a start circuit in EM series(run/stop according to ON/OFF status of X7).
Functions as NOP(no operation)in EC series.

3. The FUN99(END)instruction is not required usually. However, it is recommended to insert this instruction for separating programs at the time of test run since operation can be checked more easily. Program is executed from step 000 to FUN99 instruction. Once operation has been confirmed, delete the FUN99 instructions.
After completely clearing a program, all user memorys are written with the FUN99 instruction(though indication is not provided).
Since the FUN99 instruction is assumed in an area not yet programmed, there is no need for writing that instruction at the end of a program.

]

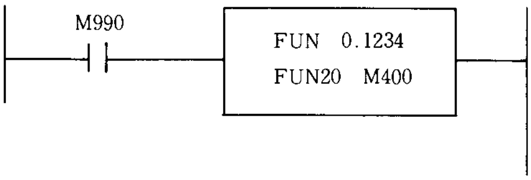

Arithmetic box start condition
Arithmetic box
F U N 0.1234
F U N 20 M400
When Acc=1, instruction in arithmetic box is executed.

[Explanation]

1. Arithmetic instruction is assumed to be contained in the arithmetic box, and consecutive arithmetic instructions are put in the same arithmetic box.
   Before each arithmetic box, start condition is to be provided. When the start condition is satisfied(Acc=1), arithmetic instruction in the arithmetic box is executed. This won't occur if the start condition is not satisfied(Acc = 0).

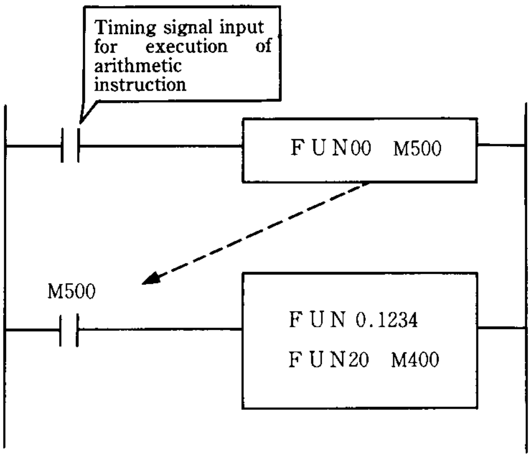
2. For an arithmetic instruction to be executed every scan, it is recommended to use the special internal output M990 for the start condition since ON status is always secured.

Example

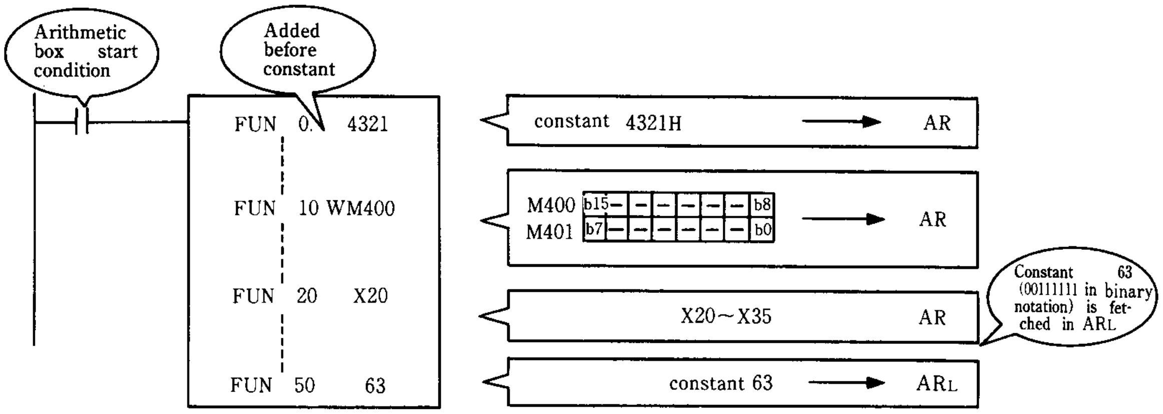


3. For an arithmetic instruction to be executed for only one scan at a certain timing, it is recommended to use the edge instruction as the start condition.

Example

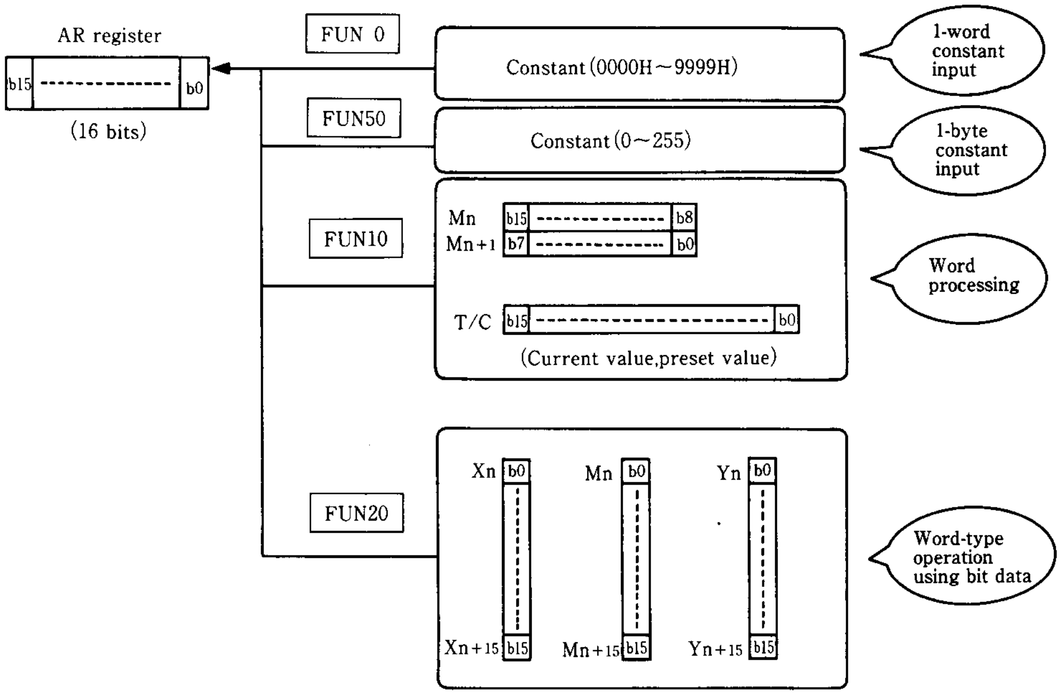


Instruc-tion	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN 0	LOADI	Load	Constant→AR	Constant (0000H~9999H)	2	↕	·	·	·
FUN10	LOADW		I/O→AR	WX,WY,WM,T/C100~295	2	↕	·	·	·
FUN20	LOADB		I/O→AR	VX,VY,VM,T/C0~95	2	↕	·	·	·
FUN50	LDBYTI		1 byte constant→AR$_L$	Constant (0~255)	2	↕	·	·	·



[Explanation]

1. Load instruction loads the word data (16 bits) or byte data (8 bits) to be processed into the AR register.Four kinds of load instructions FUN0 (LOADI),FUN10 (LOADW),FUN20 (LOADB) and FUN50 (LDBYTI) are selectively usable to suit the component.



(1) The FUN0 instruction loads a one-word constant (0000H to 9999H) into the AR register. The constant must be preceded by a decimal point (.) when keying in.

(2) The FUN10 instruction loads one-word I/O data into the AR register.

① Internal outputs are used for both bit and byte data (8-bit data for each number). 8 bit data of the specified internal output (Mn) and the next internal output (Mn+1),16-bit data in total,are loaded into the AR register.

MSB

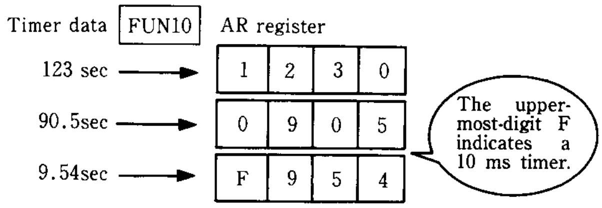
M400	b_{15}	–	–	–	–	–	–	b_8
M401	b_7	–	–	–	–	–	–	b_0

LSB

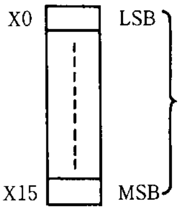
One 16-bit word consisting of data from M400 and M401 is used.

② The timer/counter current values (T/C100 to T/C195) and preset values (T/C200 to T/C295) are 4-digit BCD data (16 bits). The counter preset value and current value are loaded into the AR register without change. However,the timer value is processed as shown below before loaded into the AR register.

Example

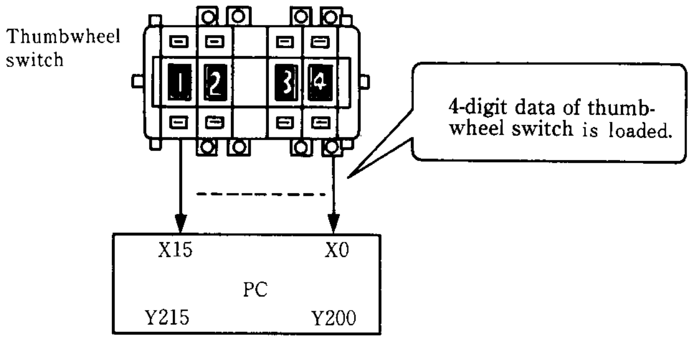


(3) The FUN20 instruction loads 16 I/O data simultaneously into the AR register. 16 data from the specified number and upward are loaded into the AR register.



16 data bits from X0 through X15 are used.

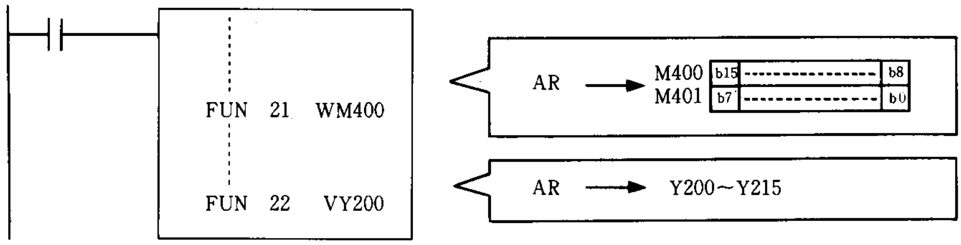
Example



The FUN20 instruction is used to load data from the thumbwheel switch in this instance.

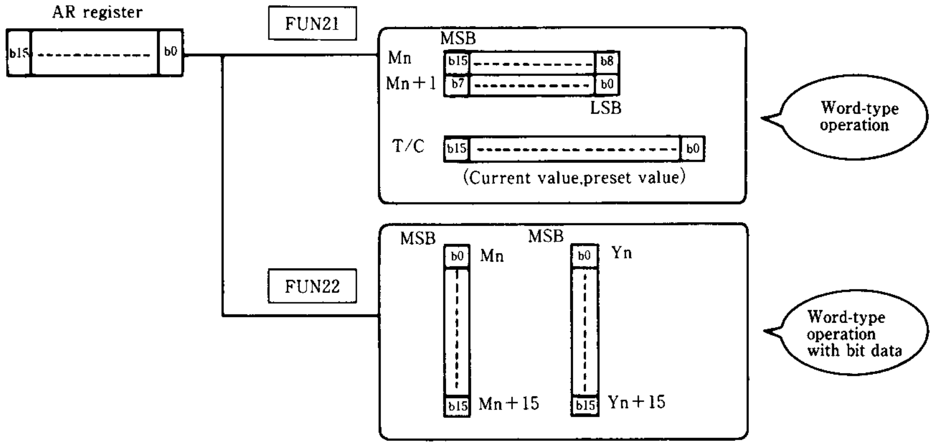
(4) The FUN50 instruction loads a desired bit pattern into the AR' register.
The E—series programmers (PGMJ and PGMJ-R2) do not have keys A through F reguired for hexadecimal notation. However,when a decimal constant (0 to 255) is specified by the FUN50 instruction,it is handled as a one-byte data (00H to FFH) and loaded into the lower 8 bits (AR_L) of the AR register. In this case,the upper 8 bits (AR_H) of the same register remain unchanged. When used in combination with the FUN82 instruction,the FUN50 instruction is capable of loading a desired bit pattern into upper 8 bits (AR_H).

Instruc-tion	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN21	OUTW	Out	AR→I/O	WY,WM,T/C100~295	2	·	·	·	·
FUN22	OUTB	Out	AR→I/O	VY,VM	2	·	·	·	·



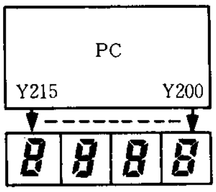
[Explanation]

1. OUT instruction outputs data in the AR register to the destination component. Two kinds of OUT instructions below are selectively usable so as to meet the component.



(1) The FUN21 (OUTW) instruction outputs data in the AR register to the 16-bit area made up of the specified internal output (Mn) and the next internal output (Mn + 1). This instruction is also used for outputting AR register data to current value (T/C100 through T/C195) or preset value (T/C200 to T/C295) of timer/counter.

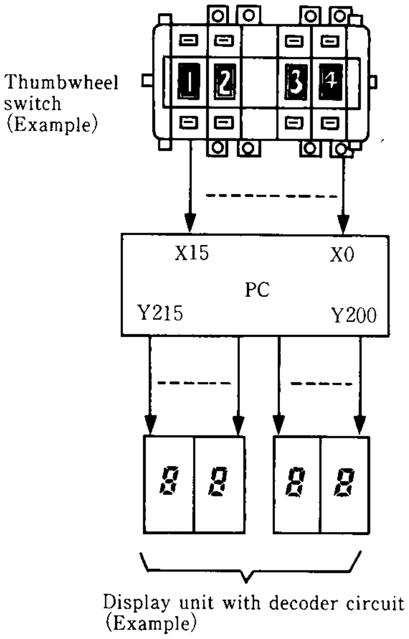
(2) The FUN22 (OUTB) instruction is used to output AR register data to the numerical display (7-segment LED).



[Application example of LOAD and OUT Instructions]

1. Explanation of operations

Example


Thumbwheel switch (Example)
1
2
3
4
X15
X0
PC
Y215
Y200
Display unit with decoder circuit (Example)

Truth table of thumbwheel switch

	Switch terminal	8	4	2	1
PC terminal		X 3	X 2	X 1	X 0
		X 7	X 6	X 5	X 4
		X11	X10	X 9	X 8
Digit		X15	X14	X13	X12
Thumb-wheel switch dial	0				
	1				●
	2			●	
	3			●	●
	4		●		
	5		●		●
	6		●	●	
	7		●	●	●
	8	●			
	9	●			●

Indicates terminal wiring.

● ON

Truth table of display unit with decoder circuit

	Display terminal	D	C	B	A
PC terminal		Y203	Y202	Y201	Y200
		Y207	Y206	Y205	Y204
		Y211	Y210	Y209	Y208
Digit		Y215	Y214	Y213	Y212
Numerical display unit	0				
	1				●
	2			●	
	3			●	●
	4		●		
	5		●		●
	6		●	●	
	7		●	●	●
	8	●			
	9	●			●

Indicates terminal wiring.

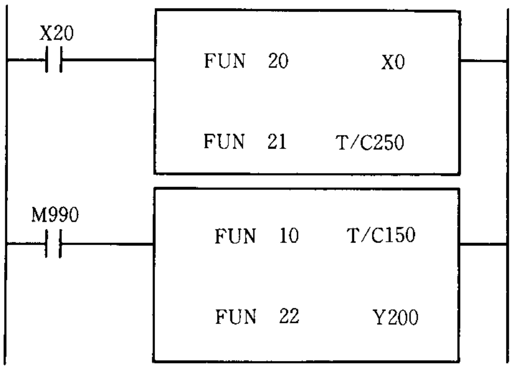
● ON

(1) Preset value of the counter in PC is set when X20 turns ON with a 4-digit BCD thumbwheel switch connected to the PC external input terminal.

(2) Current value of the counter in PC is output to the 7-segment display unit. This unit is provided with a decoder circuit.

2. Sequence

Example



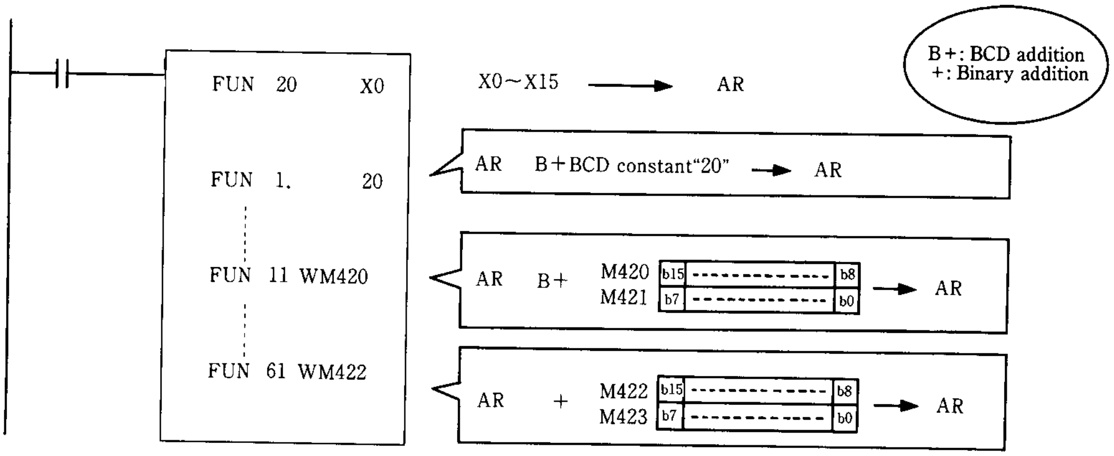
X0~X15→AR

AR→Preset value of T/C50(T/C250)

Current value of T/C50(T/C150)→AR

AR→ Y200~Y215

Instruc-tion	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN1	ADDI	BCD Add	AR B+ constant→AR	Constant (0000H~9999H)	2	↕	·	↕	·
FUN11	ADD	BCD Add	AR B+ I/O→AR	WX,WY,WM,T/C100~295	2	↕	·	↕	·
FUN61	ADDBNR	Binary Add	AR + I/O→AR	WX,WY,WM,T/C100~295	2	↕	·	↕	·


FUN 20 X0
FUN 1. 20
FUN 11 WM420
FUN 61 WM422
X0~X15 → AR
B+: BCD addition
+: Binary addition
AR B+BCD constant"20" → AR
AR B+ M420 b15 b8 M421 b7 b0 → AR
AR + M422 b15 b8 M423 b7 b0 → AR

[Explanation]

1. ADD instructions add AR register data to component data and load the sum to the AR register. The FUN1 instruction is used for BCD addition with a constant and the FUN11 instruction is used for BCD addition with I/O. For binary addition with I/O,the FUN61 instruction is employed. When the sum is within 4 digits,the carry C turns OFF.
   When the sum is more than four (4) digits,each instruction is handled as shown below.

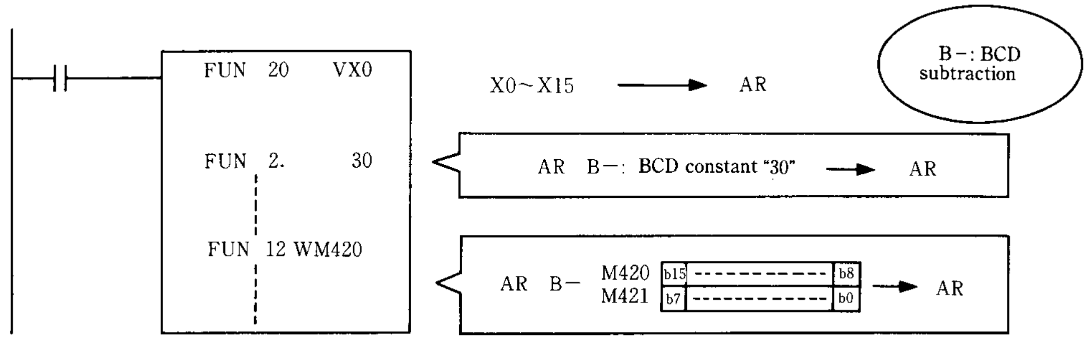
Condition	Instruction	A R	C	Remarks
Sum has exceeded 4 digits.	FUN 1	Remains unchanged	1	Carry C indicates occurrence of error.
	FUN 11			
	FUN 61	Sum loaded	1	Carry C indicates occurrence of a carry.

2. If a non-BCD data is handled by the FUN1 or FUN11 instruction,neither AR register data nor carry C data is assured. The table below lists example programs for different components.

Classifi-cation	Component		Program	Explanation
BCD addition	Constant		FUN 1. 4321	AR B+BCD constant 4321→AR
	Internal output		FUN 11 (WM)500	AR B+ WM500 →AR
	Timer/ counter	Current value	FUN 11 T/C150	AR B+T/C50 current value→AR
		Preset value	FUN 11 T/C250	AR B+T/C50 preset value→AR
Binary addition	Internal output		FUN 61 (WM)422	AR + WM422 →AR

Note (WM) is to be omitted for key in.

Instruc-tion	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN2	SUBI	BCD Subtract	AR B-constant→AR	Constant (0000H~9999H)	2	↕	·	↕	·
FUN12	SUB		AR B-I/O→AR	WX,WY,WM,T/C100~295	2	↕	·	↕	·
FUN62	SUBBNR	Binary Subtract	AR-I/O→AR	WX,WY,WM,T/C100~295	2	↕	·	↕	·


FUN 20 VX0
FUN 2. 30
FUN 12 WM420
X0~X15 → AR
B-: BCD subtraction
AR B-: BCD constant "30" → AR
AR B- M420 b15 b8 M421 b7 b0 → AR

[Explanation]

1. SUB instructions subtract component data from AR register data and load the difference to the AR register.The FUN2 instruction is used for BCD subtraction with a constant and the FUN12 instruction is used for BCD subtraction with I/O.
   For binary subtraction with I/O,the FUN62 instruction is employed. When the difference is 0 or positive,the carry C turns OFF.
   If the difference is negative,each instruction is handled as listed below.

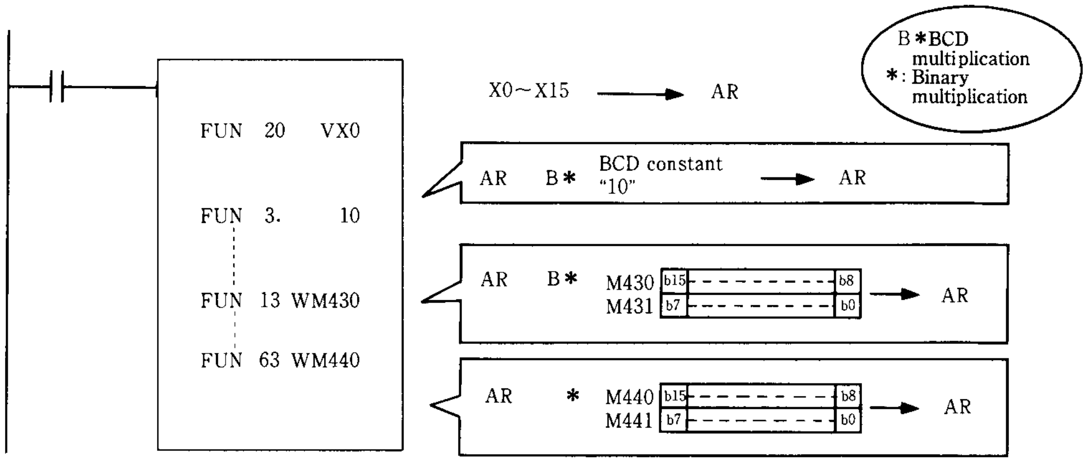
Condition	Instruction	A R	C	Remarks
Difference is negative.	FUN 2	Remains unchanged	1	Carry C indicates occurrence of error.
	FUN 12			
	FUN 62	Difference loaded	1	Carry C indicates decrement to next lower digit.

2. If a non-BCD data is handled by the FUN2 or FUN12 instruction,neither AR register data nor carry C data is reliable. The table below lists example programs for different components.

Classifi-cation	Component		Program	Explanation
BCD subtraction	Constant		FUN 2. 4321	AR B-BCD constant 4321→AR
	Internal output		FUN 12 (WM)500	AR B-WM 500 →AR
	Timer/ counter	Current value	FUN 12 T/C150	AR B-T/C50 current value→AR
		Preset value	FUN 12 T/C250	AR B-T/C50 preset value →AR
Binary subtraction	Internal output		FUN 62 (WM)510	AR - WM 510 →AR

Note (WM) is to be omitted for Key in.

Instruc-tion	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN3	MULI	BCD Multiply	AR B∗ constant→AR	Constant (0000H~9999H)	2	↕	·	↕	·
FUN13	MUL		AR B∗ I/O→AR	WX,WY,WM,T/C100~295	2	↕	·	↕	·
FUN63	MULBNR	Binary Multiply	AR ∗I/O→AR	WX,WY,WM,T/C100~295	2	↕	↕	↕	·



[Explanation]

1. MUL instructions multiply AR register data with component data and load the product to the AR register.The FUN3 instruction is used for BCD multiplication with a constant and the FUN13 instruction is used for BCD multiplication with I/O. For binary multiplication with I/O,the FUN63 instruction is employed. When the product reaches 5 digits,it is reflected on the ER register.
If the product is within 4 digits,the carry C turns OFF.

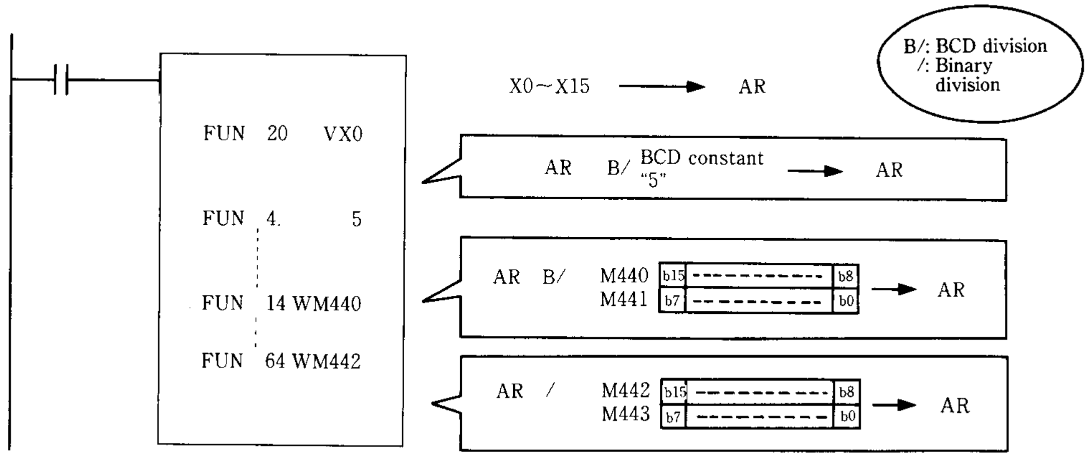
Condition	Instruction	A R	E R	C	Remarks
Product exceeds 4 digits.	F U N 3	Remains unchanged	Remains unchanged	1	Carry C indicates occurrence of error.
	F U N 13				
	F U N 63	Product loaded	Product in 5 digits or more loaded	1	Carry C indicates the product reaches 5 digits.

2. If non-BCD constant is handled by the FUN3 or FUN13 instruction,neither AR register data nor carry C data is reliable. The table below lists example programs for different components.

Classification	Component		Program	Explanation
BCD multipli-cation	Constant		F U N 3 4321	AR B∗ BCD constant 4321 →AR
	Internal output		F U N 13 (WM)500	AR B∗ WM500 →AR
	Timer/ counter	Current value	F U N 13 T/C150	AR B∗ T/C50 current value →AR
		Preset value	F U N 13 T/C250	AR B∗ T/C50 preset value →AR
Binary multiplication	Internal output		F U N 63 (WM)510	AR ∗ WM510 →AR

Note (WM) is to be omitted for key in.

Instruction	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN4	DIVI	BCD Divide	AR B/constant→AR	Constant (0000H~9999H)	2	↕	·	↕	·
FUN14	DIV		AB B/ I/O→AR	WX,WY,WM,T/C100~295	2	↕	·	↕	·
FUN64	DIVBNR	Binary Divide	AR / I/O→AR	WX,WY,WM,T/C100~295	2	↕	↕	↕	·


B/: BCD division
/: Binary division
FUN 20 VX0
FUN 4. 5
FUN 14 WM440
FUN 64 WM442
X0~X15 → AR
AR B/ BCD constant "5" → AR
AR B/ M440 b15 b8 M441 b7 b0 → AR
AR / M442 b15 b8 M443 b7 b0 → AR

[Explanation]

1. DIV instructions divide AR register data by component data and load the quotient to the AR register. The FUN4 instruction is used for BCD division by a constant and FUN14 is used for BCD division by I/O. For binary division by I/O, the FUN64 instruction is employed and the remainder is reflected on the ER register. Unless divided by 0, the carry C turns OFF. Each DIV instruction is handled as listed below in cases of usual division and 0 division.

Condition	Instruction	A R	E R	C	Remarks
Usual division	FUN 4	Quotient	Remains unchanged.	0	Remainder is neglected.
	FUN 14				
	FUN 64	Quotient	Remainder	0	Remainder is loaded in ER
÷ 0	FUN 4	Remains unchanged.	Remains unchanged.	1	Carry C indicates occurrence of error.
	FUN 14				
	FUN 64				

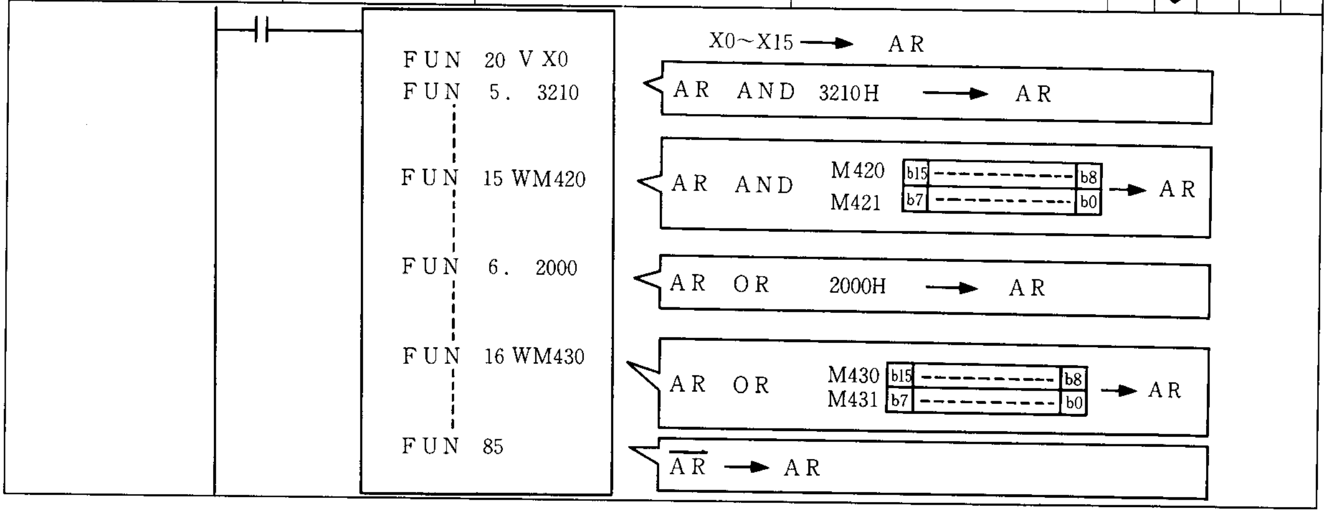
2. If non-BCD constant is handled in the FUN4 or FUN14 instruction, neither AR register data nor carry C data is reliable.

The table below lists example programs for different programs.

Classification	Component		Program	Explanation
BCD division	Constant		FUN 4. 5	AR B/ BCD constant 5→AR
	Internal output		FUN 14 (WM)500	AR B/ WM500 →AR
	Timer/counter	Current value	FUN 14 T/C150	AR B/ T/C50 current value →AR
		Preset value	FUN 14 T/C250	AR B/ T/C50 preset value →AR
Binary division	Internal output		FUN 64 (WM)510	AR / WM510 →AR

Note (WM) is to be omitted for key in.

Instruction	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN5	ANDI	Logical AND	AR AND constant →AR	Constant (0000H~9999H)	2	↕	·	·	·
FUN15	AND		AR AND I/O→AR	WX, WY, WM, T/C100~295	2	↕	·	·	·
FUN6	ORI	Logical OR	AR OR cônstant →AR	Constant (0000H~9999H)	2	↕	·	·	·
FUN16	OR		AR OR I/O → AR	WX, WY, WM, T/C100~295	2	↕	·	·	·
FUN85	WNOT	Logical NOT	$\overline{AR}$→AR	None	1	↕	·	·	·


```
FUN 20 V X0         X0~X15 ──► AR
FUN 5. 3210         AR AND 3210H ──► AR
FUN 15 WM420        AR AND M420 [b15 ---- b8] ──► AR
                           M421 [b7  ---- b0]
FUN 6. 2000         AR OR 2000H ──► AR
FUN 16 WM430        AR OR M430 [b15 ---- b8] ──► AR
                          M431 [b7  ---- b0]
FUN 85              AR(bar) ──► AR
```

[Explanation]

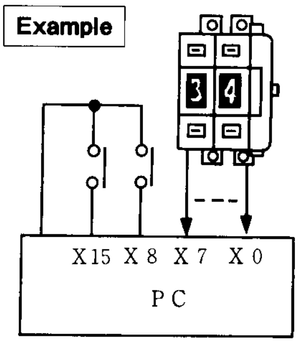
1. FUN5 (ANDI) and FUN15 (AND) instructions perform logical AND operation between AR register data and component data.

   FUN6 (ORI) and FUN16 (OR) instructions perform logical OR operation between AR register data and component data.

   FUN85 (WNOT) instruction performs logical NOT operation with regard to the AR register.

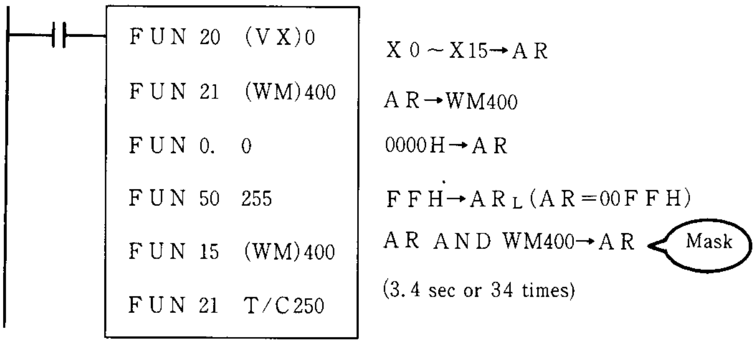
2. Example of logical AND

Example

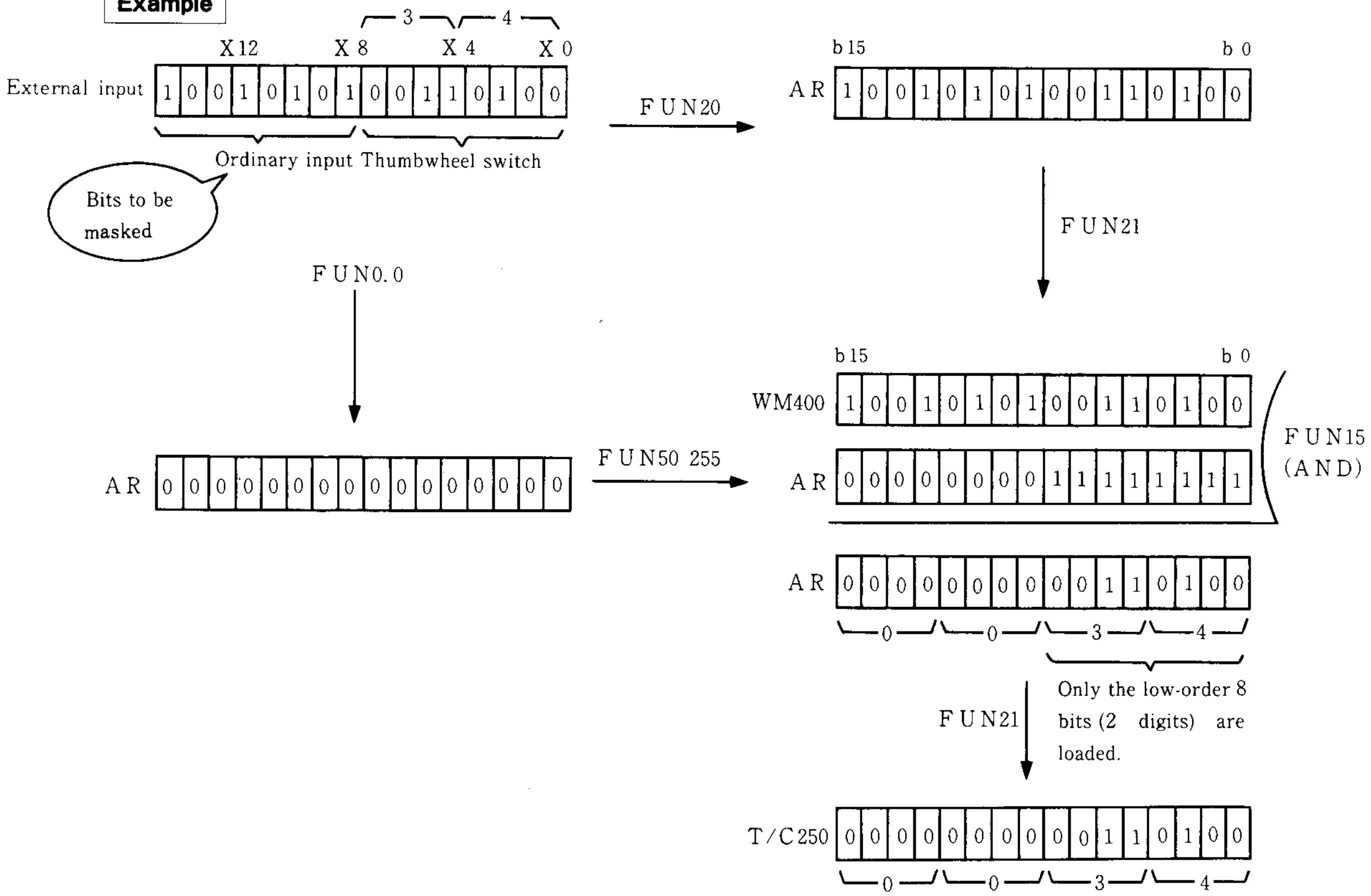


Note (VX)or(WM)is to be omitted for key in

When loading the 2-digit thumbwheel switch data into X0 through X7 to be followed by loading of ordinary input signals in X8 through X15, the switch data is also loaded into X8 through X15 unintentionally. This is because the FUN20 (LOADB) instruction operates on a data word of 16 bits long. To mask X8 through X15, use the FUN15 instruction.


```
FUN 20  (VX)0       X0~X15→AR
FUN 21  (WM)400     AR→WM400
FUN 0.  0           0000H→AR
FUN 50  255         FFH→AR_L (AR=00FFH)
FUN 15  (WM)400     AR AND WM400→AR  <Mask
FUN 21  T/C250      (3.4 sec or 34 times)
```

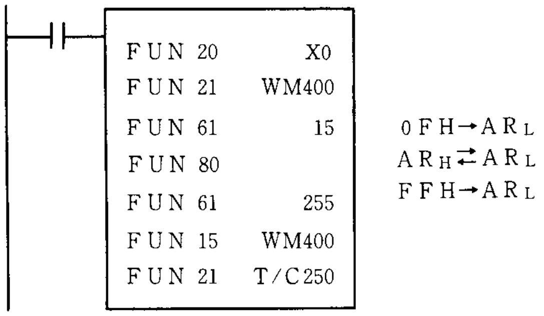
Example


3
4
X12
X8
X4
X0
External input
1 0 0 1 0 1 0 1 0 0 1 1 0 1 0 0
Ordinary input
Thumbwheel switch
Bits to be masked
FUN20
AR 1 0 0 1 0 1 0 1 0 0 1 1 0 1 0 0
b15
b0
FUN21
FUN0.0
WM400 1 0 0 1 0 1 0 1 0 0 1 1 0 1 0 0
AR 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
FUN50 255
AR 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1
FUN15
(AND)
AR 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0
0
0
3
4
FUN21
Only the low-order 8 bits (2 digits) are loaded.
T/C250 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0
0
0
3
4

The data flow is shown above. Only 2-digit data of the thumbwheel switch is effective. The T/C50 preset value can be set and X8 through X15 are usable for ordinary input signals.

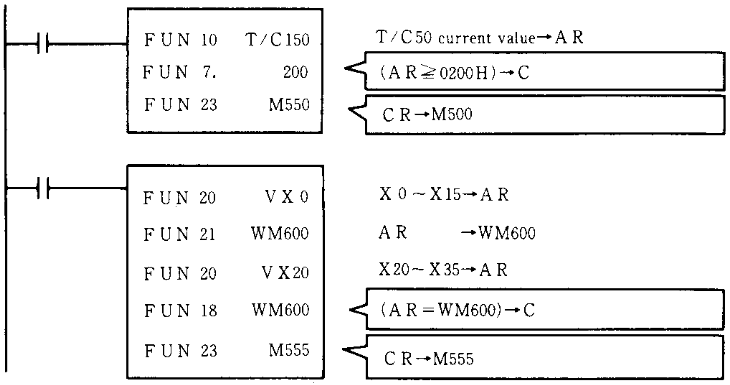
3. The diagram below shows a sequence in which only the 12 bits (3 digits) are effective with the high-order 4 bits masked.

Example



```
FUN 20      X0
FUN 21  WM400
FUN 61      15      0FH→ARL
FUN 80              ARH⇄ARL
FUN 61     255      FFH→ARL
FUN 15  WM400
FUN 21  T/C250
```

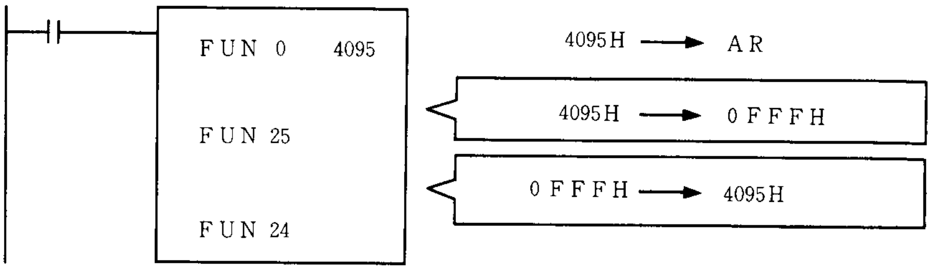
Instruc-tion	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN7	CPEHI	Compare (≧)	AR ≧ constant 1→C AR < constant 0→C	Constant (0000H~9999H)	2	·	·	↕	·
FUN17	CPEH		AR ≧ I/O 1→C AR < I/O 0→C	WX, WY, WM, T/C100~295	2	·	·	↕	·
FUN8	CPEI	Compare (=)	AR = constant 1→C AR ≠ constant 0→C	Constant (0000H~9999H)	2	·	·	↕	·
FUN18	CPE		AR = I/O 1→C AR ≠ I/O 0→C	WX, WY, WM, T/C100~295	2	·	·	↕	·
FUN9	CPLI	Compare (<)	AR < constant 1→C AR ≧ constant 0→C	Constant (0000H~9999H)	2	·	·	↕	·
FUN19	CPL		AR < I/O 1→C AR ≧ I/O 0→C	WX, WY, WM, T/C100~295	2	·	·	↕	·
FUN23	OUC	Carry output	CR → I/O (low-order 7 bits set to 0)	Y, M	2	·	·	·	·



[Explanation]

1. Compare instructions are classified into 3 types; ≧, = and <. Each type consists of 2 kinds of instruction. So six kinds of compare instructions in total are selectable to suit the component. AR register and component data are compared as binary numbers without sign. If the result of comparison is true, carry C is set to ON. If it is false, carry C is set to OFF.
   The result remains unchanged when assuming comparison is made between BCD values.
2. FUN7 (CPEHI), FUN8 (CPEI) and FUN9 (CPLI) are instructions to compare AR register data with constants 0000H to 9999H.
3. FUN17 (CPEH), FUN18 (CPE) and FUN19 (CPL) are instructions to compare AR register data with external input, external output, internal output, timer/counter current value and preset value. Component data need not be BCD data (0000H through 9999H).
4. The FUN23 (OUC) instruction outputs the carry register CR to internal output (M) or external output (Y). The low-order 7 bits of carry register CR are always at 0.

Instruction	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN24	BCD	BCD Convert	AR $\xrightarrow{\text{BCD Convert}}$ AR	None	1	↕	·	↕	·
FUN25	BNR	Binary Convert	AR $\xrightarrow{\text{Binary Convert}}$ AR	None	1	↕	·	↕	·



[Explanation]

1. The FUN24 (BCD) instruction converts the binary data in the AR register into BCD data. If the result of conversion is digits or less, carry C turns OFF.

Example

```
       0        F        F        F
AR  0 0 0 0  1 1 1 1  1 1 1 1  1 1 1 1   (Binary)

              ⇩ FUN24

       4        0        9        5
AR  0 1 0 0  0 0 0 0  1 0 0 1  0 1 0 1   (BCD)
```

NOTE: If the result of conversion overflows 4 digits, the AR register data is not converted (the contents of register remain unchanged) and carry C turns ON.

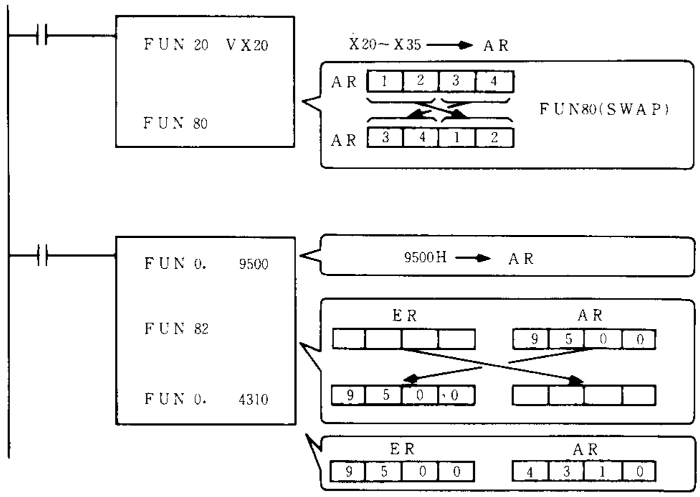
2. The FUN25 (BNR) instruction converts the BCD data in the AR register into binary data. When the AR register contains BCD data before conversion, carry C turns OFF.

Example

```
       0        2        5        5
AR  0 0 0 0  0 0 1 0  0 1 0 1  0 1 0 1

              ⇩ FUN25

       0        0        F        F
AR  0 0 0 0  0 0 0 0  1 1 1 1  1 1 1 1
```

Note: Before conversion, each digit of the AR register must be a value in the range of 0 to 9. If the AR register data is within A to F, it will not be converted (the contents of register remain unchanged) and the value of carry C will become unreliable.

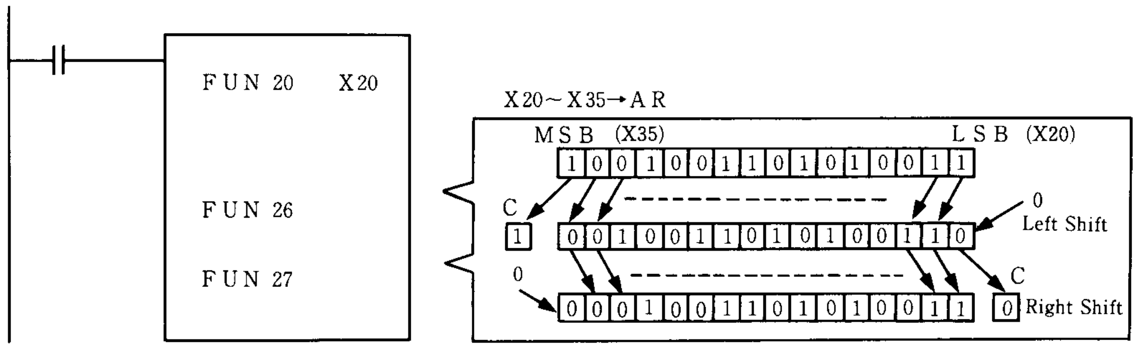
Instruc-tion	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN80	SWAP	AR_H, AR_L Exchange	$AR_H \rightleftarrows AR_L$	None	1	↕	·	·	·
FUN82	XCG	AR, ER Exchange	$AR \rightleftarrows ER$	None	1	↕	↕	·	·



[Explanation]

1. The FUN80 (SWAP) instruction exchanges the upper byte (b8 through b15) and the lower byte (b0 through b7) of the AR register.
2. The FUN82 (XCG) instruction exchanges the AR register and ER register. The FUN82 instruction is used for setting data in the ER register.

Instruc-tion	Abbreviation	Name	Function	Component	No. of words	Register			
						AR	ER	CR	Acc
FUN26	LSFR	Left Shift	C ← AR ←0	None	1	↕	·	↕	·
FUN27	RSFR	Right Shift	0 → AR →C	None	1	↕	·	↕	·

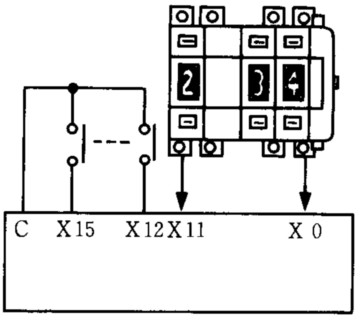


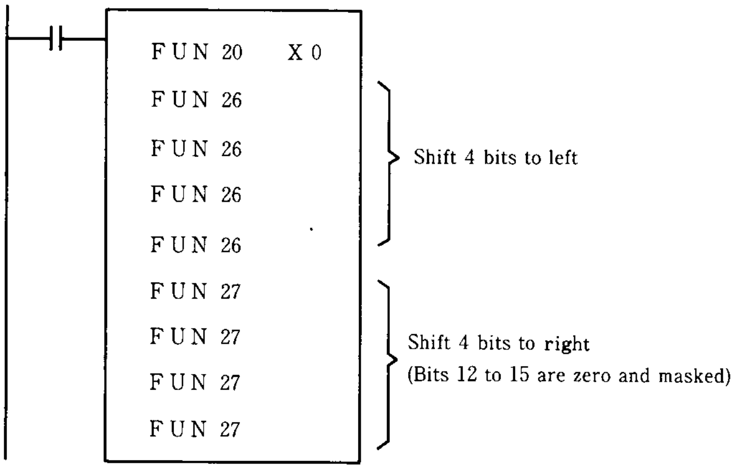
[Explanation]

1. The FUN26 (LSFR) instruction shifts AR register data 1 bit to the left. Upon shift, the least significant bit is padded with zero and the overflow bit is set to carry C.
2. The FUN27 (RSFR) instruction shifts AR register data 1 bit to the right. Upon shift, the most significant bit is padded with zero and the overflow bit is set to carry C.
3. Example of shift instructions.

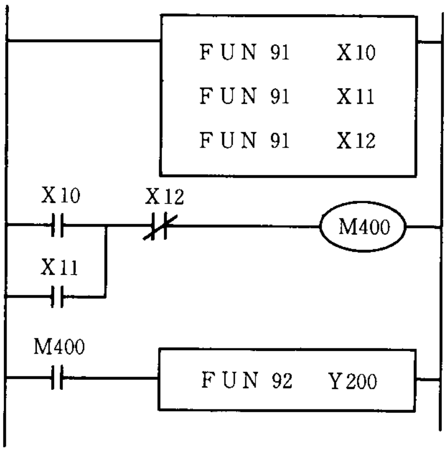
Example

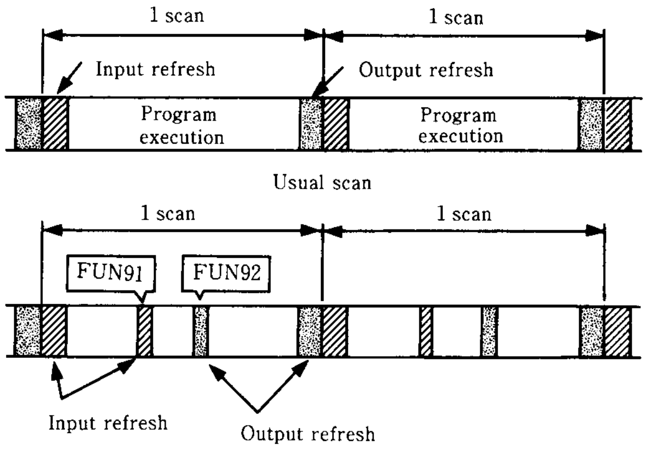
In the sequence below, only X0 through X11 data is loaded and X12 through X15 are used for ordinary input (that is, X12 through X15 are masked).





Instruction	Symbol	Meaning	Function	Component	No. of words	Change in register			
						AR	ER	CR	Acc
FUN91	REFX	I/O refresh	Inputs specified I/O.	X	1	·	·	·	·
FUN92	REFY		Outputs specified I/O.	Y	1	·	·	·	·



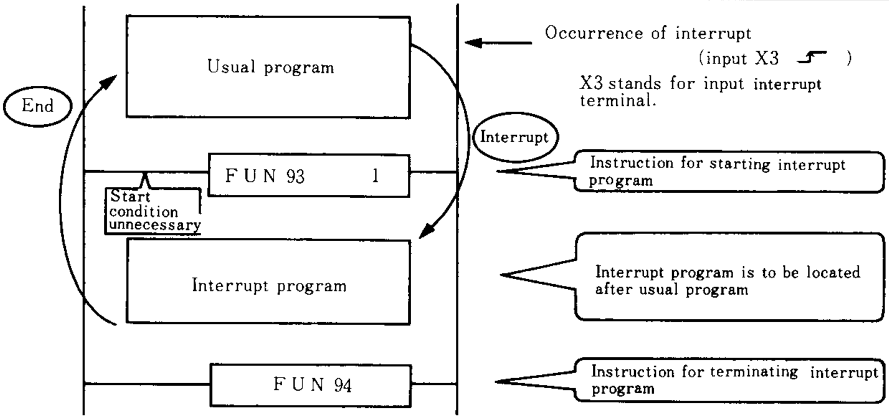


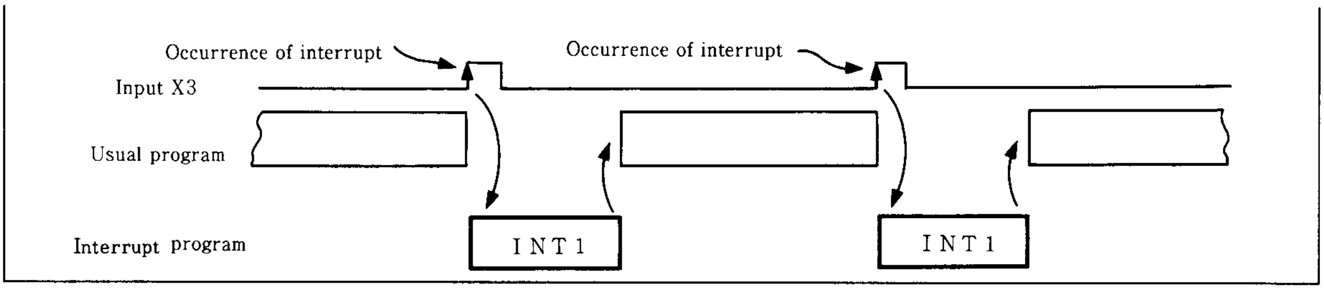
I/O refresh according to FUN91/FUN92 instructions
(Specified input and output can be refreshed in the middle of scan.)

[Explanation]

1. FUN91 (REFX) is input refresh instruction. It rewrites data memory of the specified input number in the course of scan (upon its execution). This instruction does not have a start condition.
2. FUN92 (REFY) is output refresh instruction. It rewrites the specified output number and its data memory the same as in the current Acc register during scan (upon its execution).
3. Input signals shorter than scan time can be acquired by uniform allocation of the refresh instruction at several locations in the entire program.

Instruction	Symbol	Name	Function		Component	No. of words	Change in register			
							AR	ER	C	Acc
FUN93	INT	Declares interrupt	Argument 0	Declares interrupt with high speed counter.	Arguments 0 to 63	2	-	-	-	-
			Argument 1	Declares input interrupt.			-	-	-	-
			Argument 2	Declares interrupt at fixed intervals of 10ms.			-	-	-	-
FUN94	RTI	Return from interrupt	Return from interrupt		None	1	Value before occurrence of interrupt			


Usual program
End
F U N 93 1
Start condition unnecessary
Interrupt program
F U N 94
Interrupt
Occurrence of interrupt (input X3 ⌐)
X3 stands for input interrupt terminal.
Instruction for starting interrupt program
Interrupt program is to be located after usual program
Instruction for terminating interrupt program



[Explanation]

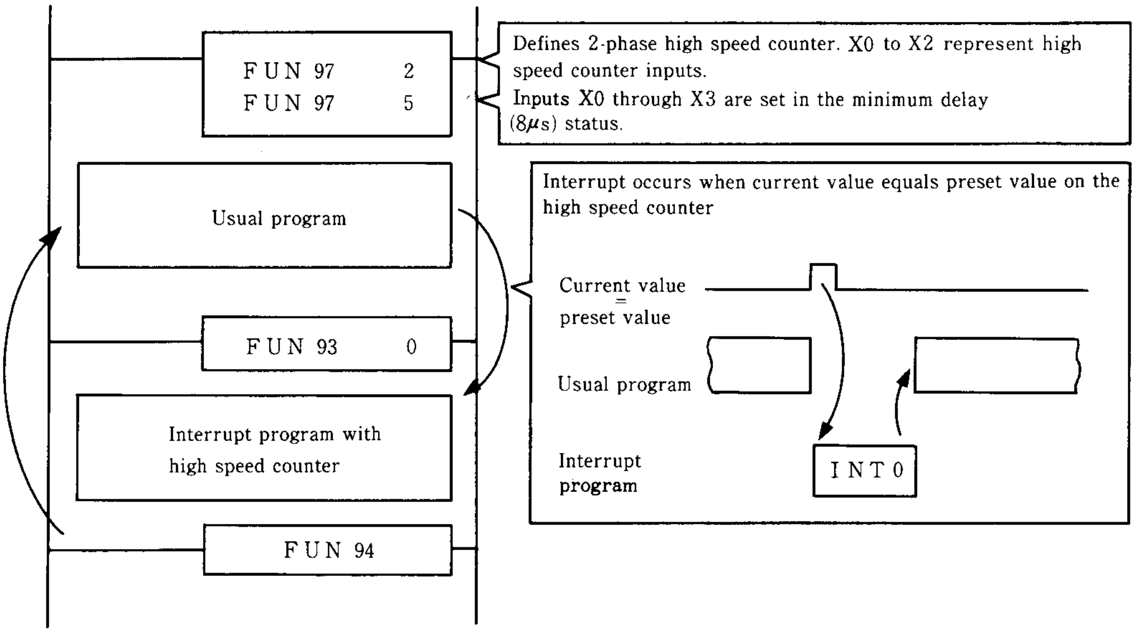
1. An interrupt program is to be located next to a usual program. These programs are to be separated by the FUN93 (INT) instruction. FUN99 (END) is not used. The end of interrupt program must always be the FUN94 (RTI) instruction. Neither FUN93 nor FUN94 requires start condition.
2. There are three kinds of interrupt listed below.

Kind	Factor of interrupt occurrence
INT0	Current value of high speed counter equals preset value.
INT1	According to edge signal of input X3 (either rising or trailing edge selectable)
INT2	At fixed intervals of 10 ms

(1) INT0 (interrupt with high speed counter)

The EC series incorporates a BCD 8-digit high speed counter. This counter is defined by the FUN97 (MODE) instruction. (For details,refer to "high speed counter" explained later.)

An interrupt program is executed when the current value equals the preset value on the high speed counter on condition that the counter is defined and that the program is written sandwiched between the FUN93 0 (INT0) and FUN94 (RTI) instructions.


FUN 97 2
FUN 97 5
Defines 2-phase high speed counter. X0 to X2 represent high speed counter inputs.
Inputs X0 through X3 are set in the minimum delay (8μs) status.
Usual program
FUN 93 0
Interrupt program with high speed counter
FUN 94
Interrupt occurs when current value equals preset value on the high speed counter
Current value = preset value
Usual program
Interrupt program
INT0

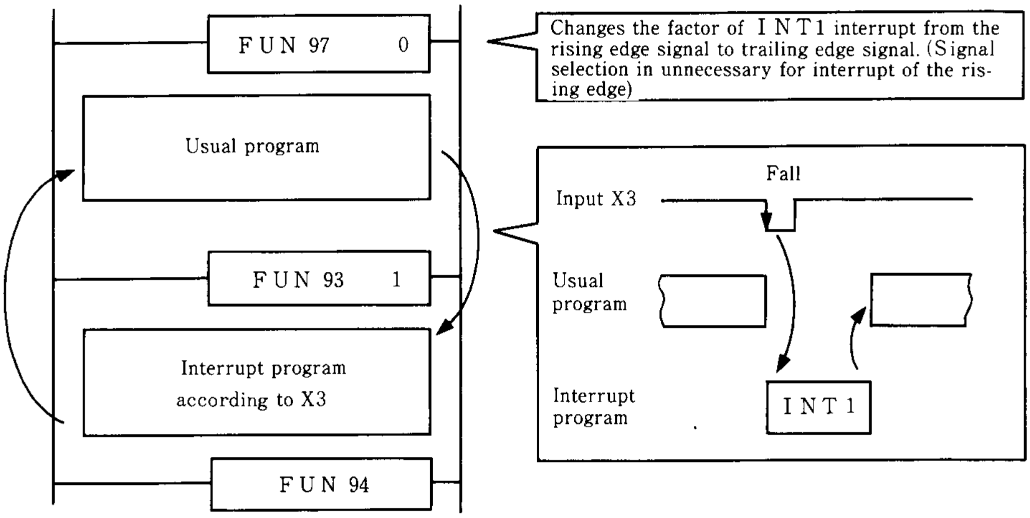
(2) INT1 (input X3 interrupt)

① Interrupt at rising edge (_|¯) of input X3

In case an interrupt program is written between the FUN93 1 (INT1) and FUN94 (RTI) instructions after a usual program, the program is executed at the rising edge of external input X3.

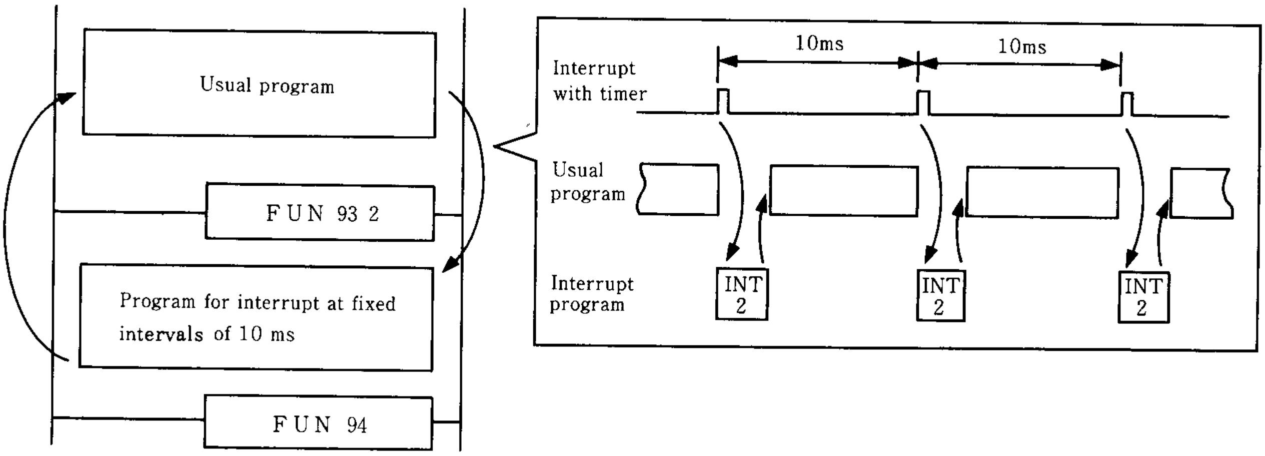
② Interrupt at trailing edge (¯|_) of input X3

Interrupt at trailing edge defined by the FUN97 (MODE) instruction. Interrupt program is written as in the interrupt at the rising edge. The program is executed at the trailing edge of external input X3.


FUN 97 0
Changes the factor of INT1 interrupt from the rising edge signal to trailing edge signal. (Signal selection in unnecessary for interrupt of the rising edge)
Usual program
FUN 93 1
Interrupt program according to X3
FUN 94
Fall
Input X3
Usual program
Interrupt program
INT1

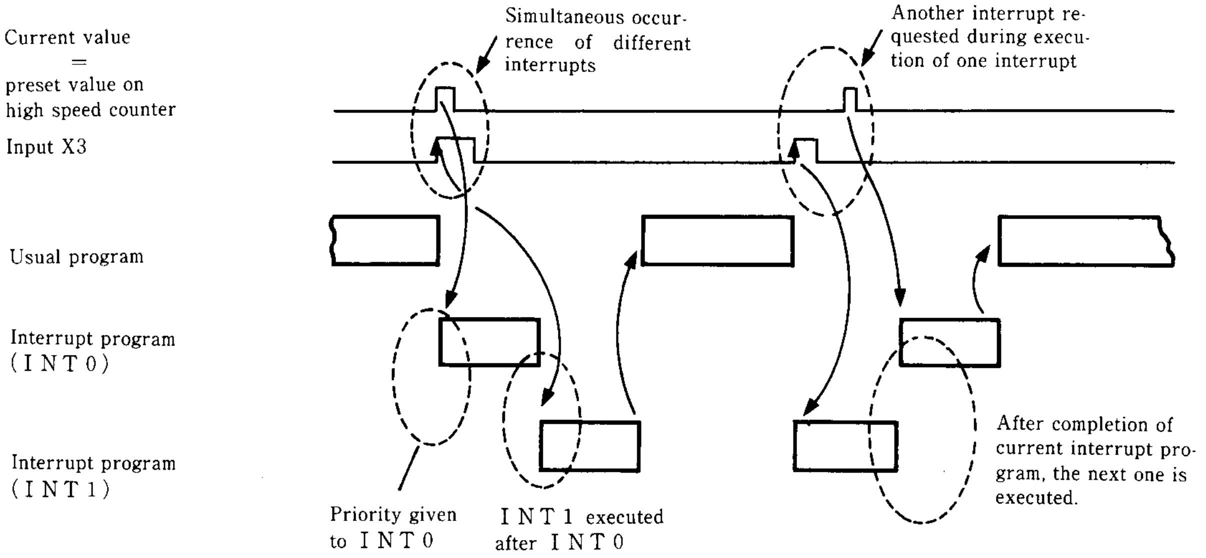
(3) (INT2) (interrupt at fixed intervals of 10ms)

Interrupt program is executed every 10ms when it is written between the FUN93 2 (INT2) and FUN94 (RTI) instructions after a usual program.



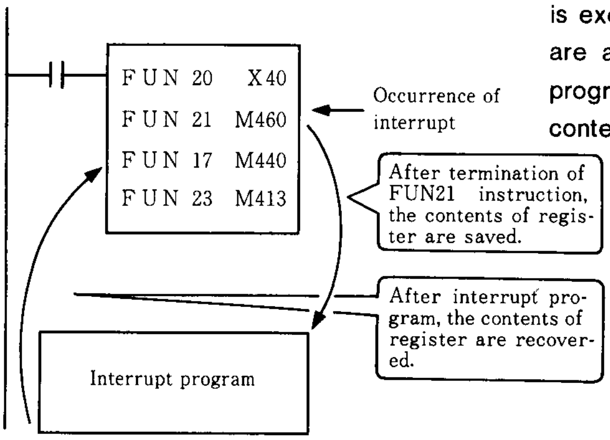
3. For interrupt,priority is given to INT0,INT1 and INT2 in this order. If different interrupts are applied simultaneously, the one given a higher priority is executed first. (Request for the other interrupt is stored in memory.)

   In case another interrupt is requested during execution of any interrupt, the second interrupt program is executed after completion of the first one. If the interrupt factor of the second program is the same as the first one, the second program will not be executed.



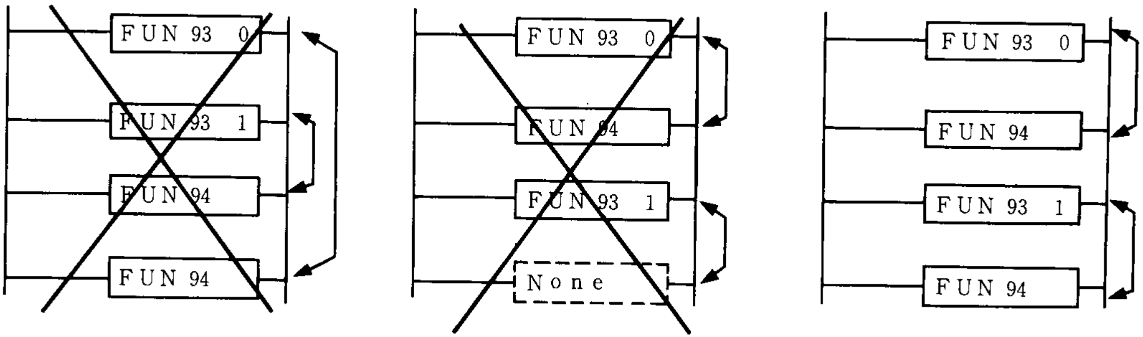
Current value = preset value on high speed counter
Input X3
Usual program
Interrupt program (INT0)
Interrupt program (INT1)
Simultaneous occurrence of different interrupts
Another interrupt requested during execution of one interrupt
Priority given to INT0
INT1 executed after INT0
After completion of current interrupt program, the next one is executed.

4. When applying an interrupt,the instruction under execution is terminated and the relevant interrupt program is executed once. On this occasion, the contents of register are automatically saved. After termination of the interrupt program, a usual program before interrupt returns and the contents of register are recovered.



FUN 20 X40
FUN 21 M460
FUN 17 M440
FUN 23 M413
Occurrence of interrupt
After termination of FUN21 instruction, the contents of register are saved.
After interrupt program, the contents of register are recovered.
Interrupt program

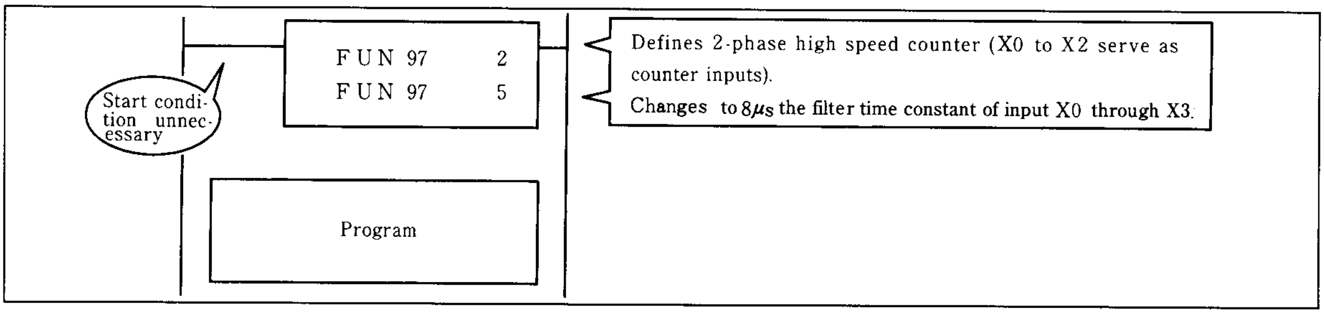
5. Interrupt program cannot be nested. FUN93 and FUN94 must always be used as a pair.



good

I/O Refresh	Interrupt	Mode Set	High Speed Counter

Instruction	Symbol	Name	Function		Component	No. of words	Change in register			
							AR	ER	C	Acc
FUN97	MODE	Mode Set	Argument 0	Changes factor of input interrupt to trailing edge.	Argument 0 to 63	2	·	·	·	·
			Argument 1	Defines single-phase high speed counter. X0～X2:inputs			·	·	·	·
			Argument 2	Defines 2-phase high speed counter. X0～X2:inputs			·	·	·	·
			Argument 3	Changes to 8μs the filter time constant of high speed counter or input interrupt.			·	·	·	·
			Argument 4	Change inputs X0～X3 to 16ms except interrupt and high speed counter.			·	·	·	·
			Argument 5	Changes to 8μs the filter time constant of input X0 through X3.			·	·	·	·
			Argument 6	Changes to 16ms the filter time constant of high speed counter or input interrupt.			·	·	·	·
			Argument 7	Changes to 8ms the filter time constant of high speed counter or input interrupt.			·	·	·	·
			Argument 8	Changes to 2ms the filter time constant of high speed counter or input interrupt.			·	·	·	·
			Argument 9	Changes to 1ms the filter time constant of high speed counter or input interrupt.			·	·	·	·
			Argument 10	Changes to 8μs the filter time constant of high speed counter or input interrupt.			·	·	·	·
			Argument 11	Change inputs X0～X3 to 16ms except interrupt and high speed counter.			·	·	·	·
			Argument 12	Change inputs X0～X3 to 8ms except interrupt and high speed counter.			·	·	·	·
			Argument 13	Change inputs X0～X3 to 2ms except interrupt and high speed counter.			·	·	·	·
			Argument 14	Change inputs X0～X3 to 1ms except interrupt and high speed counter.			·	·	·	·
			Argument 15	Change inputs X0～X3 to 8μs except interrupt and high speed counter.			·	·	·	·


Start condition unnecessary
FUN 97 2
FUN 97 5
Defines 2-phase high speed counter (X0 to X2 serve as counter inputs).
Changes to 8μs the filter time constant of input X0 through X3.
Program

[Explanation]

1. The FUN97 (MODE) instruction defines operation mode. It is valid anywhere it is written in a program.Start condition is unnecessary. A parameter undefined in the FUN97 instruction remains at its initial value.(For instance, the filter time constant will remain at the initial value 4 ms unless otherwise specified.)
2. Mode is selectable among 16 kinds below.

(1) MODE 0 (defines interrupt at trailing edge of input X3)

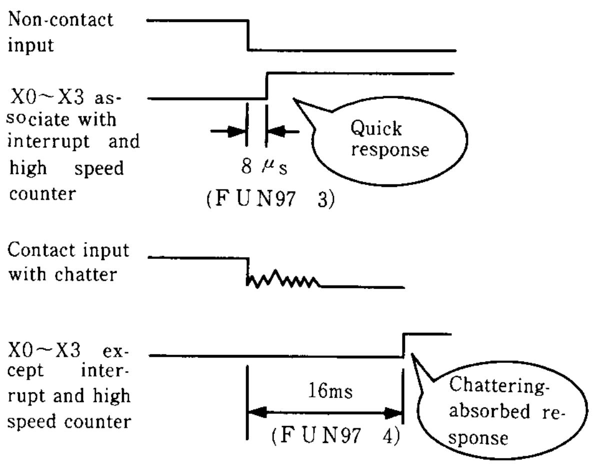
Input X3 serves to input interrupt. It is initially set so that interrupt is input at its rising edge (_┌‾).Input X3 is changeable so as to input interrupt at the trailing edge (‾┐_)by writing the FUN97 0 (MODE 0) function.

(2) MODE 1, MODE 2 (define high speed counter)

The FUN97 1 (MODE 1) and FUN97 2 (MODE 2) instructions define the kinds of high speed counter.

Instruction	Argument	Operation of high speed counter			
		No. of phases	X 0	X 1	X 2
FUN 97	1	1 (single phase)	_┌‾ Incremented at	‾┐_ Decremented at	M (marker)
	2	2	Phase A	Phase B	M (marker)

(3) MODE 3, MODE 4 (define filter time constant of inputs)


Non-contact input
X0~X3 associate with interrupt and high speed counter
Quick response
8 μs
(FUN97 3)
Contact input with chatter
X0~X3 except interrupt and high speed counter
16ms
(FUN97 4)
Chattering-absorbed response

① The FUN97 3 (MODE 3) instruction changes the filter time constant (input delay time) of X0~X2 defined as inputs of high speed counter and X3 defined as interrupt input to 8 μs.

This instruction ensures quick response to chattering-free and non-contact input such as encoder signal. It is used for a high speed counter which operates at 10kHz or less.

② The FUN97 4 (MODE 4) instruction changes the filter time constant of inputs X0 through X3 to 16 ms. In case of chattering signal such as from a high power contact, input delay time is extended for absorbing the chattering to stabilize response. But, it does not effect for input specified as high speed counter or interrupt.

(4) MODE 5 (Changes to 8 μs the filter time constant of inputs X0~X3)

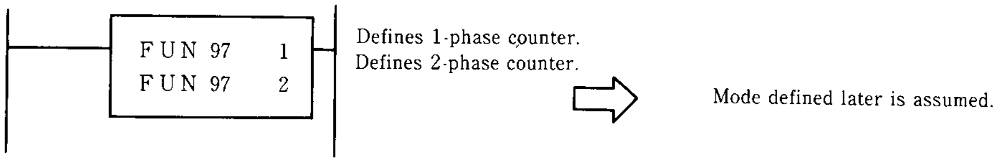
The FUN97 5 (MODE 5) instruction is used for high speed counters within a range of 2 to 10kHz. It changes to 8 μs the filter time constant of inputs X0 to X3 (sets inputs X0 to X3 minimum delay mode)

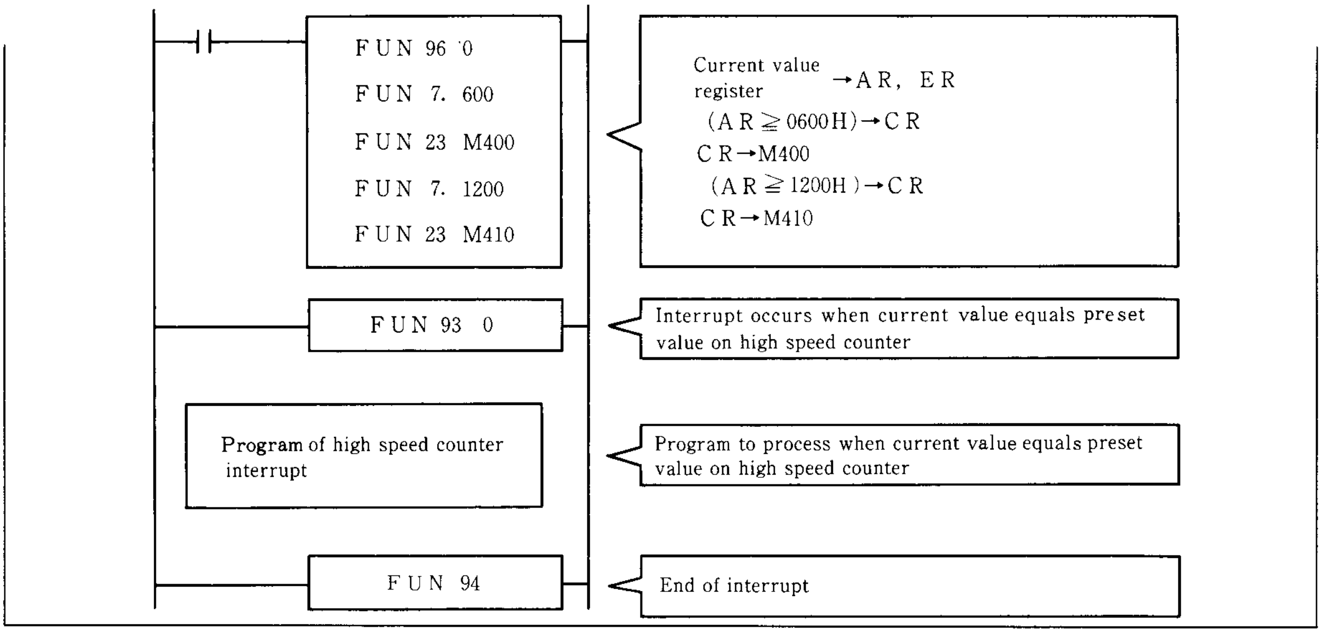
(5) MODE6,MODE7,MODE8,MODE9,MODE10(Change the filter time constant)

The FUN97 6,7,8,9,10(mode6,7,8,9,10)instruction changes the filter time constant(input delay time) of inputs X0 to X2 defined as inputs of high speed counter and X3 defined as interrupt input to 16ms,8ms,2ms,1ms,8 μs.

(6) MODE11,MODE12,MODE13,MODE14,MODE15(Change the filter time constant)
The FUN97 11,12,13,14,15(mode11,12,13,14,15)instruction changes the filter time constant(input delay time) of inputs X0 through X3 to 16ms,8ms,2ms,1ms,8 μs But it dose not effect for input specified as high speed counter or interrupt.

3. Mode set instruction is checked at start of operation and it is handled the same as in the FUN98 (NOP) instruction.
4. If different modes are set in a single block, the mode defined later is adopted.




FUN 96 0
FUN 7. 600
FUN 23 M400
FUN 7. 1200
FUN 23 M410
Current value register →AR, ER
(AR ≧ 0600H)→CR
CR→M400
(AR ≧ 1200H)→CR
CR→M410
FUN 93 0
Interrupt occurs when current value equals preset value on high speed counter
Program of high speed counter interrupt
Program to process when current value equals preset value on high speed counter
FUN 94
End of interrupt

[Explanation]

1. When using a high speed counter, it must be defined as either a 1-phase counter by the FUN97 1 (MODE 1) instruction or a 2-phase counter by the FUN97 2 (MODE 2).

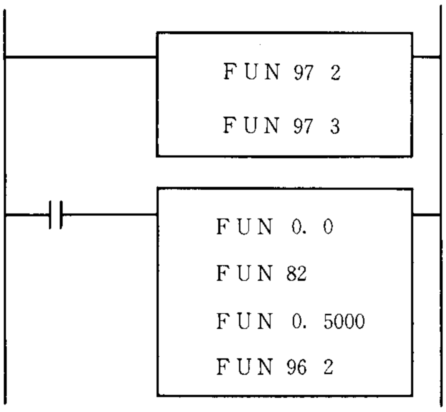
(1) 1-phase counter ...X0 or X1 functions as a counter input, and X2 as a marker (M).

① Current value increments at the rising edge (_┌¯) of X0.

② Current value decrements at the trailing edge (¯┐_) of X1.

③ X2 serves as a marker (M). At its rising edge (_┌¯), current value is reset (to 0). Because marker (M) is not level input, count is made even when it stays at ON.

Instruction	Symbol	Argument	Function	No. of words	Register status		
					AR	ER	C
FUN96	HC	0	Current value register → AR, ER	2	↕	↕	·
		1	AR, ER → Current value register		·	·	·
		2	AR, ER → Preset value register		·	·	·



Defines 2-phase high speed counter

Sets to the filter time constant of counter inputs X0 through X2 8 μs.

0000H→AR
AR↔(ER is set with 0000H)
5000H→AR
ER, AR(00005000H)→Preset value register

① When phase-B input lags behind phase-A input, current value increments at the rising edge of phase-A input.

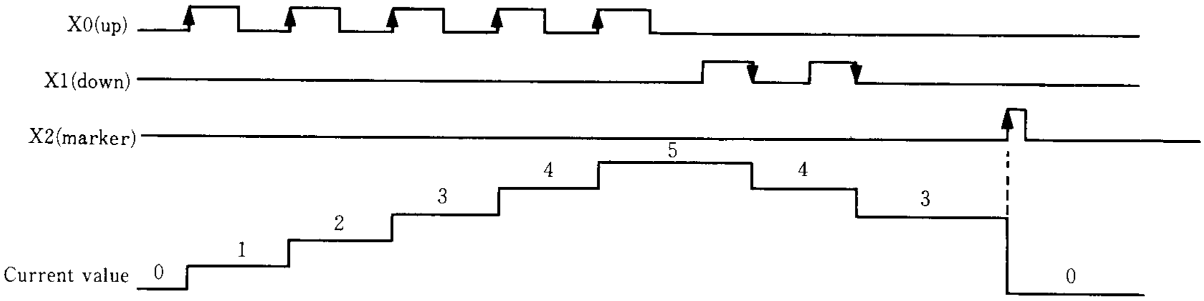
② When phase-B input advances ahead of phase-A input, current value decrements at the trailing edge of phase-A input.

③ The current value does not change for 1 pulse during changeover count up and count down.

④ The current value is reset (to 0) at the rising edge of marker. Because marker (M) is not level input, count is made even when it is at ON.

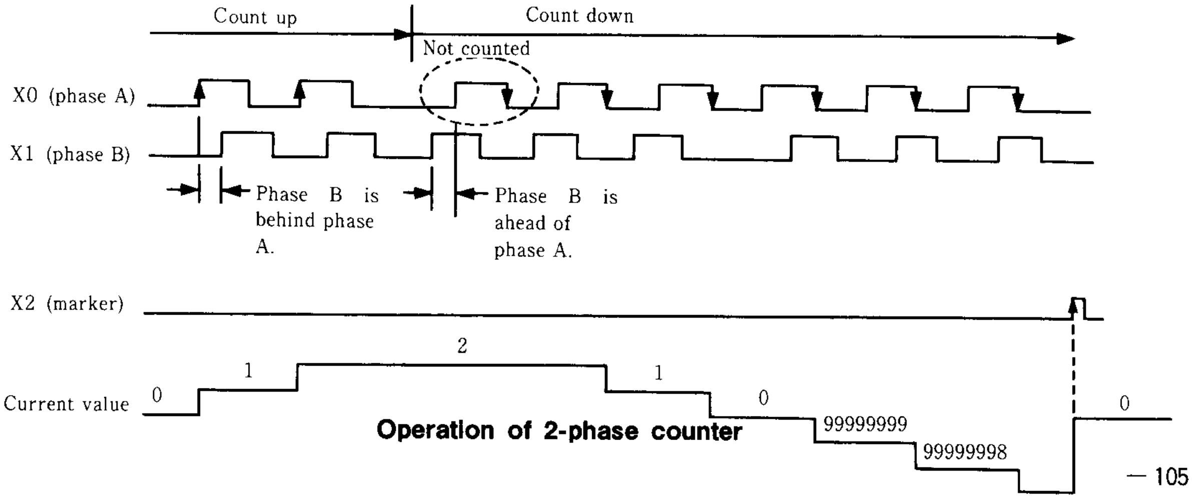
[Note]

Both current value register and preset value register are handled as BCD value. When monitored in a program, they are indicated correctly with the hexadecimal monitor (H affixed).



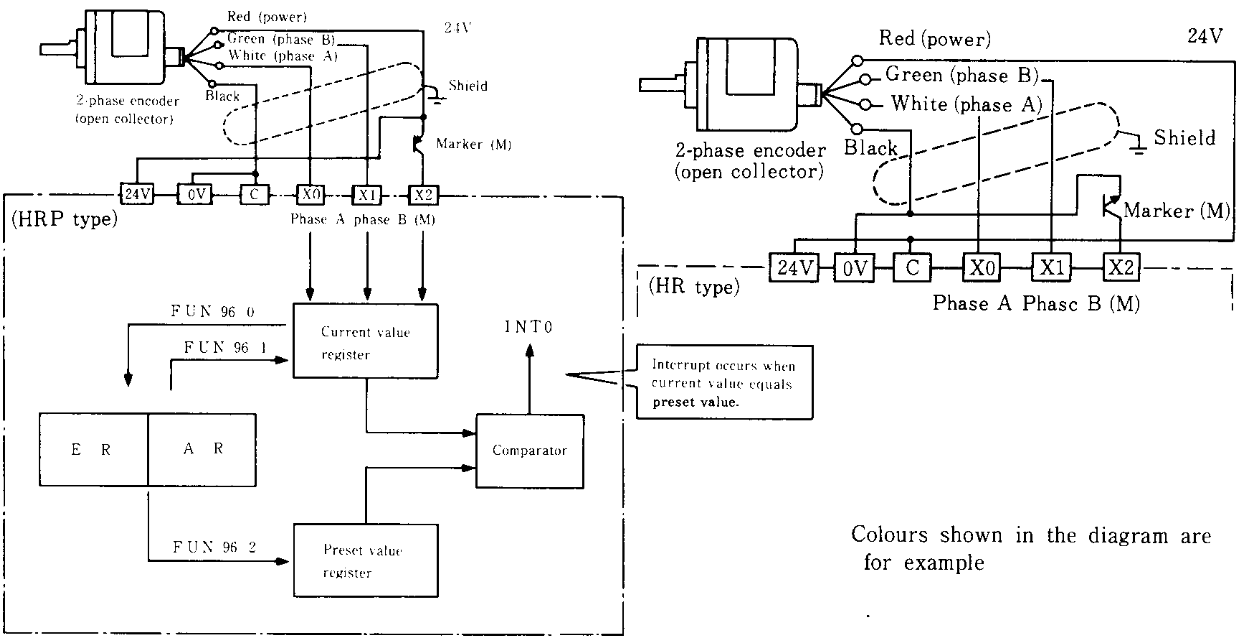
Operation of 1-phase counter

(2) 2-phase counter ...X0 functions as a phase-A input,X1 as a phase-B, and X2 as a marker (M). Whether to count up to or down is determined according to the phase difference of pulses entered to phases A and B.



Operation of 2-phase counter

EXAMPLE



Colours shown in the diagram are for example

NOTE

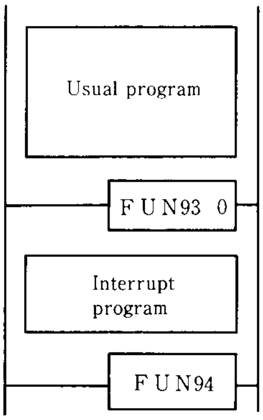
Input wires from an encoder must be separated from other input/output lines and shielded. Unless shielded,miscount might take place due to noise.

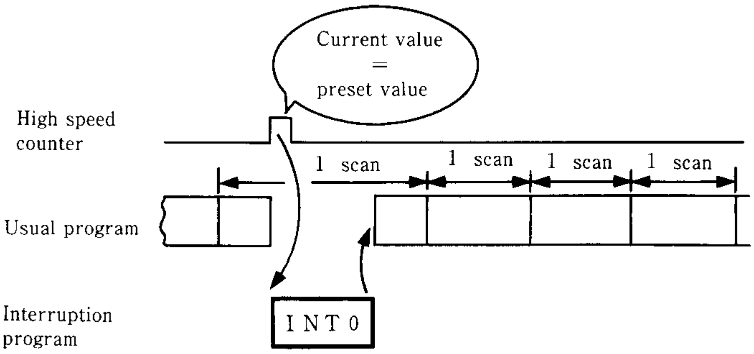
2. Definition of filter time constant in to be specified if necessary.
   The filter time constant (input delay time) of inputs X0 through X2 is initially set at 4 ms. It is to be altered when high speed pulses need be followed. For this purpose, the FUN97 (MODE) instruction is used. Listed below are frequencies countable with each time constant.

Instruction	Argument	Function	1-phase counter		2-phase counter
			Increment	Decrement	
FUN97	9	Changes to 1ms the filter time constant of high speed counter or input interrupt.	0.5kHz	0.5kHz	0.5kHz
	10	Changes to 8 μs the filter time constant of high speed counter or input interrupt.	10kHz	10kHz	10kHz
Initial value		4ms	100Hz	100Hz	100Hz

3. Block diagram of high speed counter and external wiring with encoder is shown as an example below.
   An incremental type encoder with open collector output can be connected.
   This figure exemplifies connection with the Omron E6A-CW4C.
   In case of a 1-phase encoder, the green wire (phase-B) must not be connected.

5. Interrupt (INT 0) occurs when current value equals preset value. Interrupt program is executed when it is written.





4. Detail of counter registers is explained below.

(1) Current value register

① BCD 8 digits (00000000H to 99999999H)
Operation is as explained on page 105.

② The FUN96 0 (MODE 0) is used to read the current value register.
Low-order 4 digits of current value are in AR.
High-order 4 digits of current value are in ER.

③ The FUN96 1 (MODE 1) is used to write in the current value register. If data is written in other than BCD,a wrong value might be read out. To prevent this,only BCD data must be written.

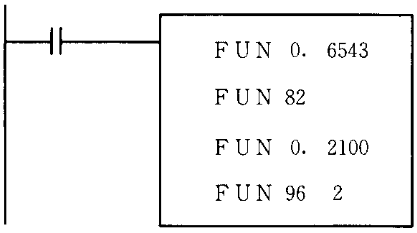
④ Current value will not be retained on occurrence of power failure. It is initialized to 00000000H when operation restarts.

(2) Preset value register

① BCD 8 digits (00000000H to 99999999H)

② The FUN96 2 is used to read the preset value register. Procedures for setting constants are shown below.

EXAMPLE



```
FUN 0. 6543
FUN 82
FUN 0. 2100
FUN 96  2
```

Constant 6543H → AR(high-order 4 digits set to AR)
AR ←→ ER(constant 6543H set to ER)
Constant 2100H → AR(low-order 4 digits set to AR)
ER,AR,(65432100H) → preset value register

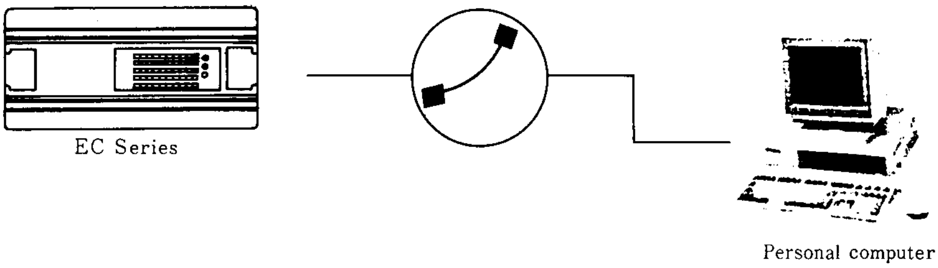
③ Preset value will not be retained on occurrence of power failure. It is initialized when operation restarts. Initial value is 80000000H.

4. Peripheral equipment

Personal computer interface

Personal computer interface function

○ Configuration

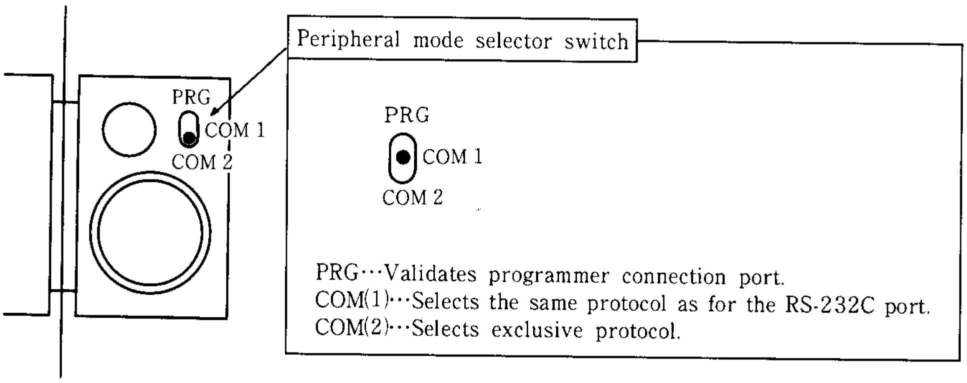


[Explanation]

1. Since the EC Series comprises the RS-232C interface,it is directly connectable to a personal computer.

2. Direct connection to personal computer

For connection to a personal computer,turn the peripheral mode selector switch to COM(1) or COM(2). At the PRG position,the programmer connection port is validated. At COM (1) or COM (2) position, the RS-232C port is validated, namely the programmer connection port is unusable.



PRG···Validates programmer connection port.
COM(1)···Selects the same protocol as for the RS-232C port.
COM(2)···Selects exclusive protocol.

Note

The status of the peripheral mode selector switch is determined just when turning on power supply. Alteration of switch setting during energization is ineffective. For mode change,power supply must be turned off.

(1) At the COM(1) position,programming and monitoring are possible by use of the personal computer programming software E-LDR.

(2) At the COM(2) position,the exclusive protocol is selected.
Refer to the EC protocol manual separately issued.

NOTICE

The limitations of software package E-LADDER that is applied to EC/ECL are shown below.
1.Use E-LADDER V5.xx,and select EB mode.
2.If use E-LADDER V4.xx,select EM mode within MS-DOS V3.xx.
(It is impossible to use instructions over specifications of EM.)

MS-DOS is a trademark of Microsoft Corporation.

4. Cable connection between EC series and personal computers are illustrated below.

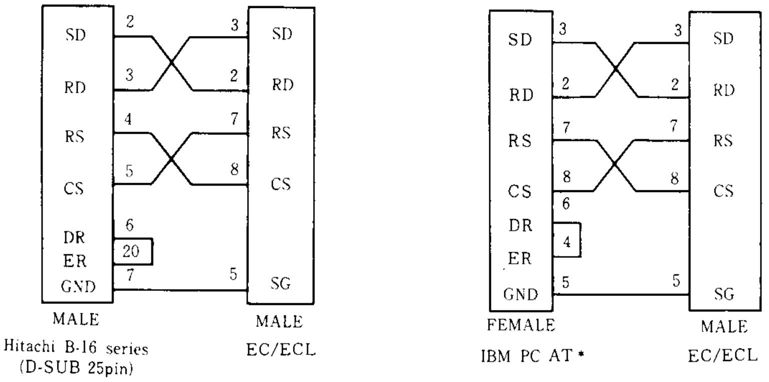


Figure 4-1 Cable connection between EC/ECL and the computer

*IBM PC AT is a product of Internal Business Machines

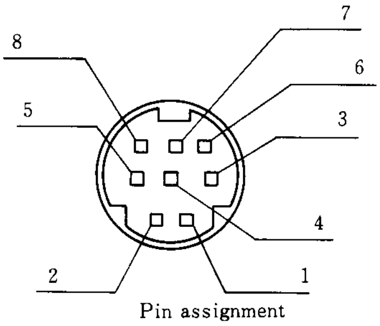
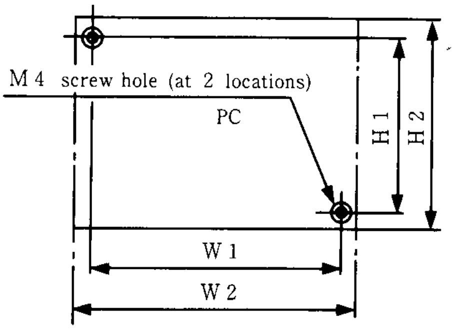


Figure 4-2 Pin assignment of EC/ECL (FEMALE)

[External dimensions and mounting dimensions]



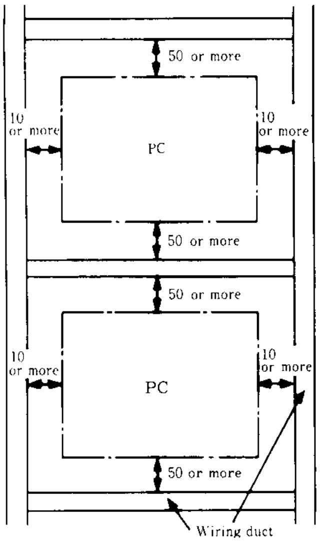
dimensions in mm

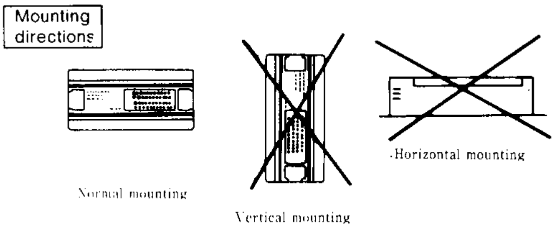
Item \ Model	EC (ECL)
W1	162
W2	170
H1	72
H2	80

1. Provide a space of 50 mm or more at the top and bottom of each unit for facilitating ventilation and maintenance. Also secure a free space at the left and right for ventilation.
2. During installation,pay strict attention not to let fragments due to drilling or wiring fall into the programmable controller.
3. Avoid installation right above equipment which radiates much heat (such as a heater,transformer or large-capacity resistor).
4. Secure a distance of 200 mm or more from a high tension cable (3,000 V min.) or power cable.

[Installation interval]

dimensions in mm

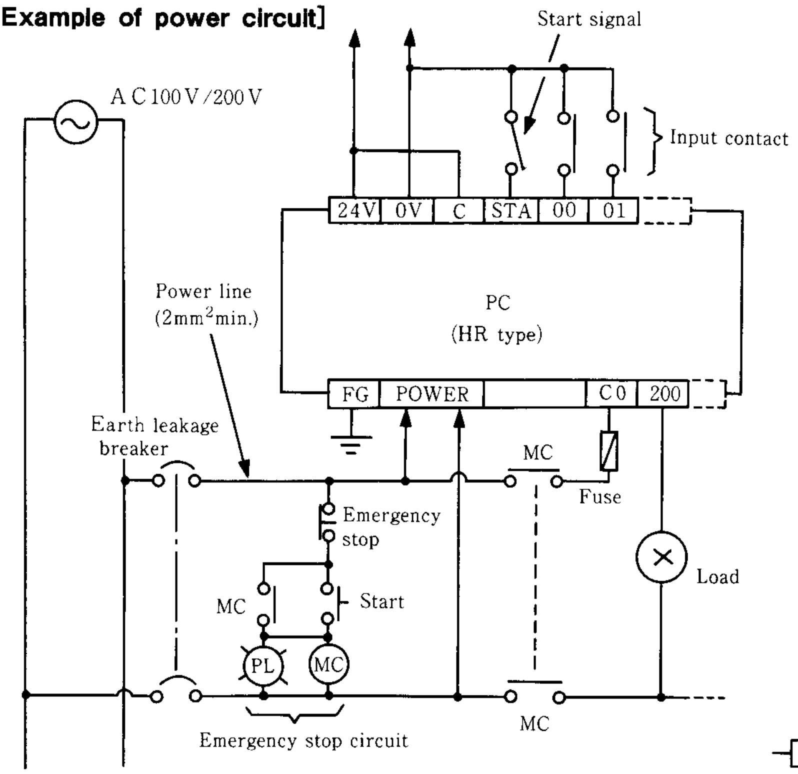




[Installation environment]

1. Avoid locations which receive direct sunlight, and which are subjected to condensation or are exposed to wind and rain.
2. Installation is unallowable at locations where there is dust,oil smoke,conductive dust or corrosive gas.
3. Do not install the programmable controller at locations at which vibration or shock will be directly applied.

[Example of power circuit]

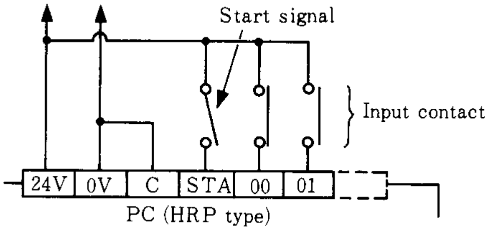


1. Line voltage

   This instrument operates on either 100 V or 200 V AC system. Connect a desired voltage within 93.5 to 250 V AC to the POWER terminal.

2. Emergency stop circuit

   It is recommended to apply external load to the PC via a relay for load power supply which can be opened/closed by a pushbutton. This allows load to be turned off for emergency stop independently of PC operation.



3. Use a power cable of 2 mm² or more to prevent occurrence of voltage drop.

[Power specifications]

Item \ Model		EC(L)-20	EC-28	EC(L)-40	EC(L)-60
Line voltage(V)		110/120/200/220/240VAC(85～264VAC),50/60Hz			
Power consumption (VA)	Without programmer	14VA or less	16VA or less	19VA or less	24VA or less
	With programmer	18VA or less	20VA or less	23VA or less	28VA or less
Sensor current via 24 V terminal		Current flown through sensor I = 470mA −(7mA × no.of ON inputs) −(6mA × no.of ON outputs)			
Permissible momentary power failure		20ms			

[Grounding]

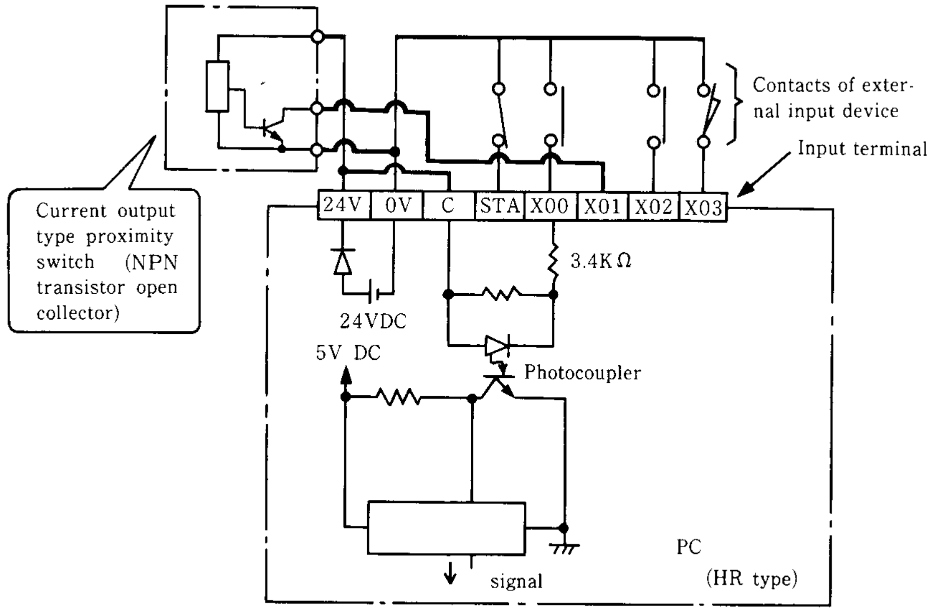
Connect the grounding terminal (FG terminal) to make 100 Ω or less using a cable of 2 mm² or more.Restrict the length of grounding cable within 20m

(1) Grounding can be shared with an instrument panel or relay panel.

(2) Common grounding must be avoided with equipment which may generate high-level noise such as a high-frequency furnace,large-scale power panel (beyond a few KW),thyristor converter and electric welding machine.

[Example of Inputcircuit]

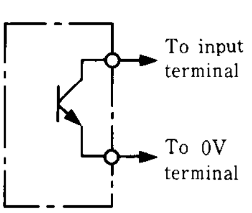
(HR type)


Current output type proximity switch (NPN transistor open collector)
Contacts of external input device
Input terminal
24V
0V
C
STA
X00
X01
X02
X03
3.4KΩ
24VDC
5V DC
Photocoupler
signal
PC
(HR type)

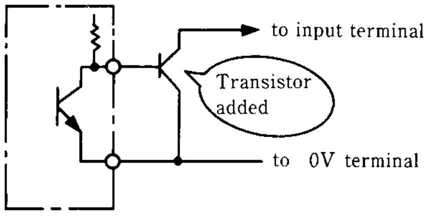
The example shows input circuit for negative logic.

[Explanation] (HR type)

1. The EC series incorporates the power supply (24 V DC) for external inputs. When each input terminal (X00,X01...) is short-circuited with the 0V terminal (0V),input is turned on. As a result,a current of about 7 mA flows from the PC to the external input contacts.
2. Sensors such as proximity switch and photoelectric switch are directly connectable when they are of current output type (NPN transistor open collector output).
Sensors of voltage output type must be connected to the input terminal via a transistor.

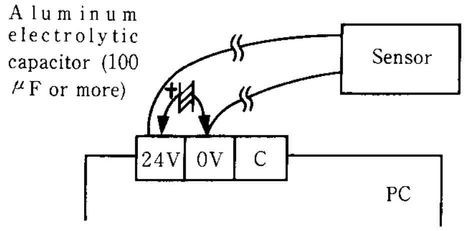


Current output type (directly connectable)



Voltage output type (requiring addition of transistor)

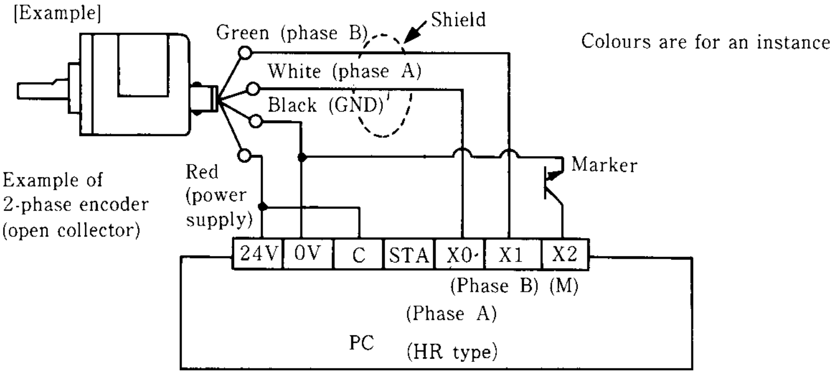
(HR type)



3. Sensor power supply is to be used as shown below.
   (1) Current can be supplied to the sensor via the 24V terminal. Its current value I is represented by:
   I = 470mA – (7mA × no.of ON inputs)
   – (6mA × no.of ON outputs)
   (2) Since ripple of the 24 V DC power supply is within 200 mVp-p,it does not pose a problem on use for ordinary sensors. However,if a sensor misoperates because it has an extremely high sensitivity or its wiring is too long,it is effective to connect an aluminum electrolytic capacitor of 100 μF or more (at 50 V min.) between the 24 V terminal and zero volt terminal.

(HR type)

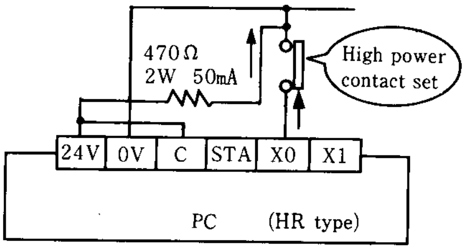
4. Although the instrument is sensitive to an input current within 2 to 6 mA,secure 7 mA for reliable ON operation and 1 mA or less for reliable OFF operation.
5. Input response time is explained below.
   (1) The primary and secondary circuits of input are isolated by means of a photocoupler. The secondary circuit is provided with a filter. This is for minimizing the chance of misoperation due to chattering of external input contacts or due to entrance of noise through the input line. The filter delays response for about 4 ms.
   (2) As for 4 input terminals X0 through X3,response time is changeable by user program.
6. High speed counter input and interruption input are explained below.
   The input terminals X0 through X2 are usable as high speed counter input terminals when programmed by user. The input terminal X3 is usable as an interruption input terminal also when programmed by user. **The signal wires of the high speed counter must be separated completely from other I/O wires or must be shielded.**



(HR type)

7. Prevention of poor contact of high power contact is explained below.

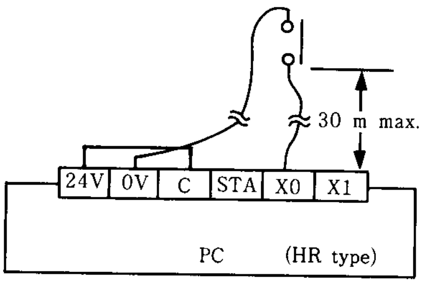
Example



When external contacts are closed, a current of about 7 mA flows through them. Therefore,use contacts which do not incur poor contact at that current level. If you must employ a high power contact set,an adequate current must be supplied to the contacts via a resistor as shown at left in order to prevent poor contact.

8. Length of input wiring is to be limited no more than 30 m.

Example



Input wiring must be 30 m max. If wiring beyond 30 m is inevitable,the input wire and output wire must be separated completely.

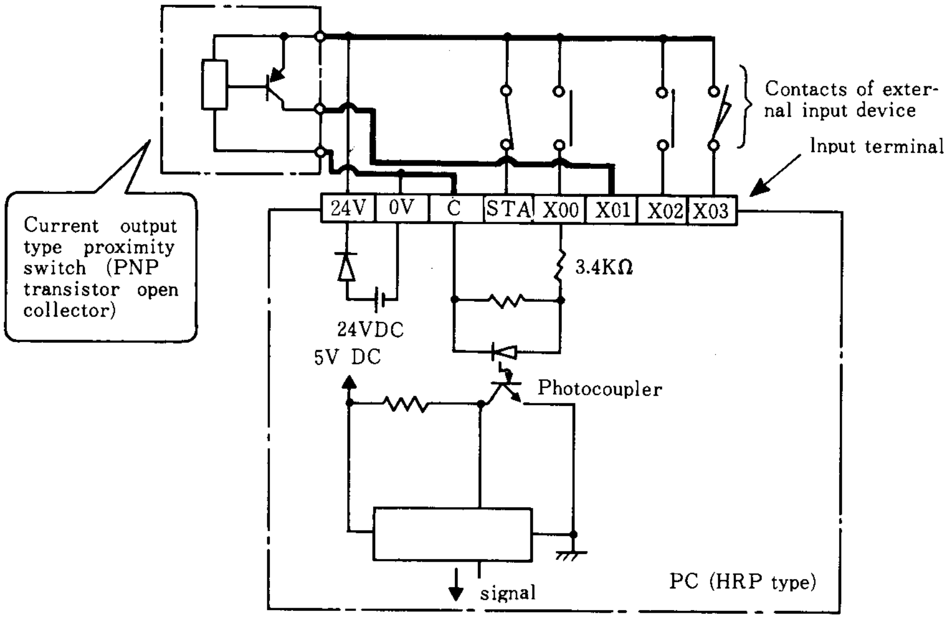
Even in this case,wiring length must not exceed 100 m.

Be sure to keep the circuit resistance within 300 Ω at worst.

9. Ripple must be 10% max. Neither full-wave nor half-wave rectification power supply is allowed to be used for external input.

[Example of Inputcircuit]

(HRP type)

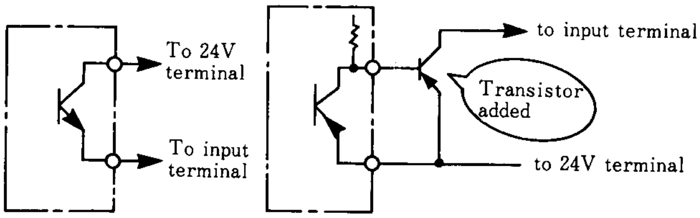

Contacts of external input device
Input terminal
Current output type proximity switch (PNP transistor open collector)
24V
0V
C
STA
X00
X01
X02
X03
3.4KΩ
24VDC
5V DC
Photocoupler
signal
PC (HRP type)

The example shows input circuit for positive logic.

[Explanation] (HRP type)

1. The EC series incorporates the power supply (24 V DC) for external inputs. When each input terminal (X00,X01...) is short-circuited with the common terminal (C),input is turned on. As a result,a current of about 7 mA flows from the PC to the external input contacts.
2. Sensors such as proximity switch and photoelectric switch are directly connectable when they are of current output type (PNP transistor open collector output).

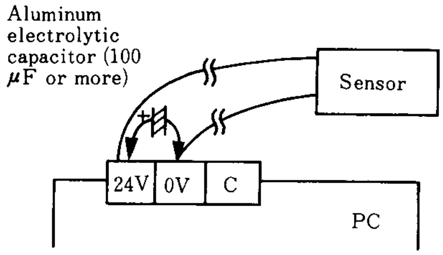
   Sensors of voltage output type must be connected to the input terminal via a transistor.



Current output type (directly connectable)

Voltage output type (requiring addition of transistor)

(HRP type)



3. Sensor power supply is to be used as shown below.

   (1) Current can be supplied to the sensor via the 24V terminal. Its current value I is represented by:

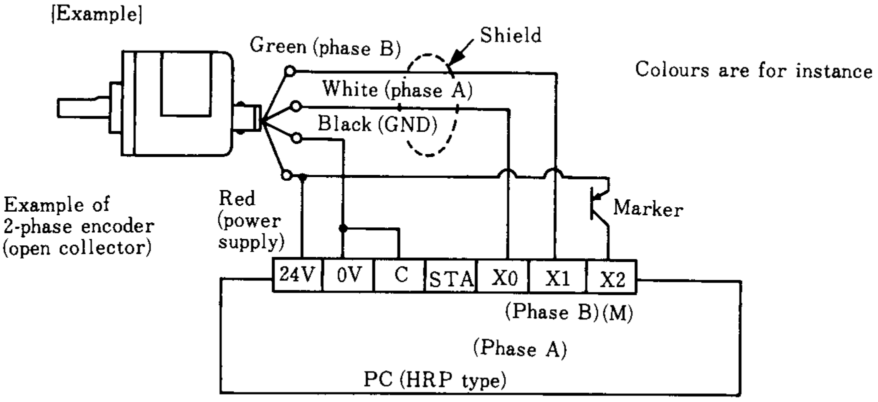
   $$I = 470\text{mA} - (7\text{mA} \times \text{no.of ON inputs}) - (6\text{mA} \times \text{no.of ON outputs})$$

   (2) Since ripple of the 24 V DC power supply is within 200 mVp-p,it does not pose a problem on use for ordinary sensors. However,if a sensor misoperates because it has an extremely high sensitivity or its wiring is too long,it is effective to connect an aluminum electrolytic capacitor of 100 μF or more (at 50 V min.) between the 24 V terminal and zero volt terminal.

(HRP type)

4. Although the instrument is sensitive to an input current within 2 to 6 mA,secure 7 mA for reliable ON operation and 1 mA or less for reliable OFF operation.
5. Input response time is explained below.
   (1) The primary and secondary circuits of input are isolated by means of a photocoupler. The secondary circuit is provided with a filter. This is for minimizing the chance of misoperation due to chattering of external input contacts or due to entrance of noise through the input line. The filter delays response for about 4 ms.
   (2) As for 4 input terminals X0 through X3,response time is changeable by user program.
6. High speed counter input and interruption input are explained below.
   The input terminals X0 through X2 are usable as high speed counter input terminals when programmed by user. The input terminal X3 is usable as an interruption input terminal also when programmed by user. **The signal wires of the high speed counter must be separated completely from other I/O wires or must be shielded.**

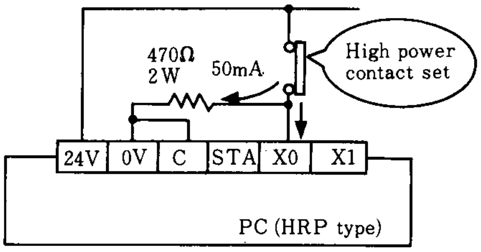
[Example]



(HRP type)

7. Prevention of poor contact of high power contact is explained below.

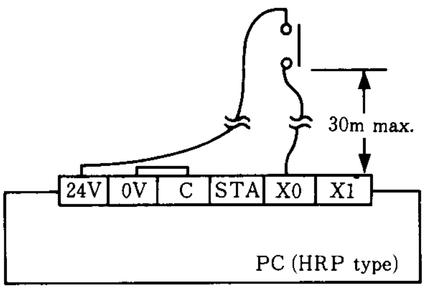
Example



When external contacts are closed, a current of about 7 mA flows through them. Therefore,use contacts which do not incur poor contact at that current level. If you must employ a high power contact set,an adequate current must be supplied to the contacts via a resistor as shown at left in order to prevent poor contact.

8. Length of input wiring is to be limited no more than 30 m.

Example



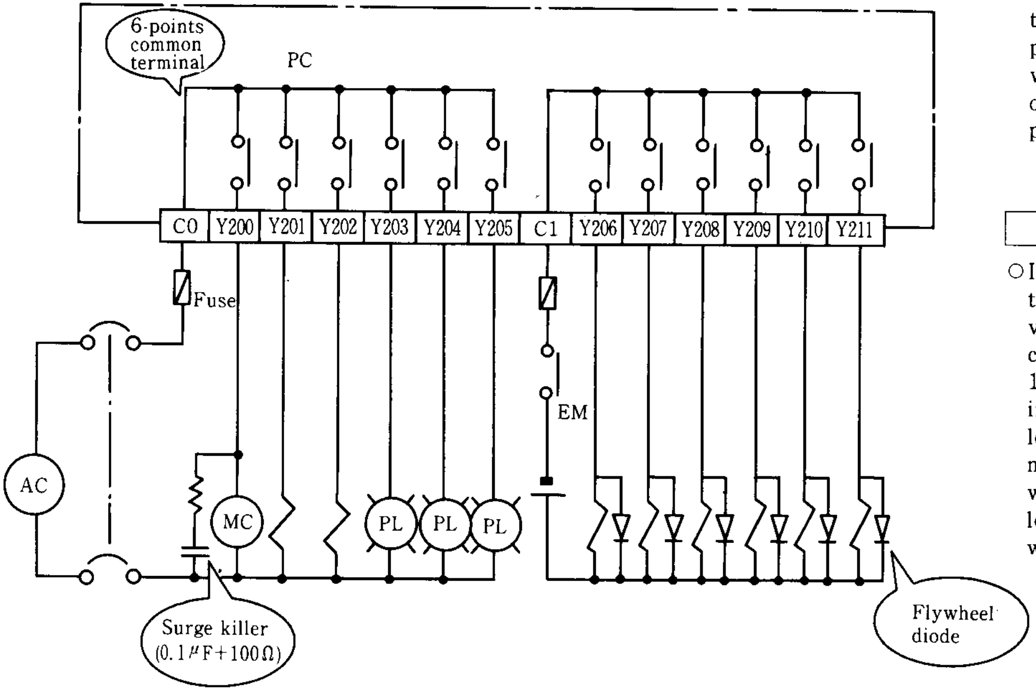
Input wiring must be 30 m max. If wiring beyond 30 m is inevitable,the input wire and output wire must be separated completely.

Even in this case,wiring length must not exceed 100 m.

Be sure to keep the circuit resistance within 300 Ω at worst.

9. Ripple must be 10% max. Neither full-wave nor half-wave rectification power supply is allowed to be used for external input.

[Example of output circuit]



Emergency stop circuit

○ It is recommended to apply external load to the PC via a load power relay (EM) which can be opened/closed by a push-button.

Inductive load

○ In case of an inductive load (contactor, valve,etc.) whose coil capacity exceeds 10 VA (after turning on), a surge killer must be connected in parallel with the load.For DC load,connect a flywheel diode.

[Explanation]

1. Output terminal

   Common terminals(C0,C1 and so on)are isolated from each other, so they are used in different circuit blocks.

   When using in the same circuit block, the common terminals need be short-circuited.

2. Lifetime of relay

   The lifetime of relay will be significantly shortened when it is used for breaking a rush overcurrent or directly driving a capacitor load.

3. No fuse is incorporated in the output circuit.

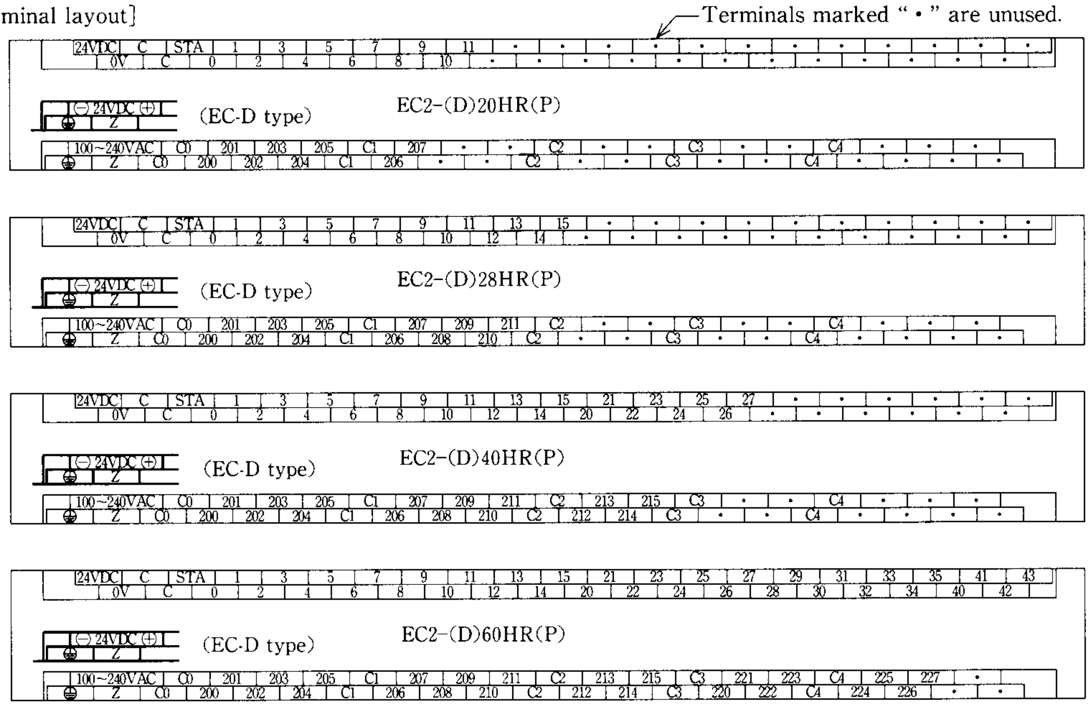
   Therefore, each common line must be provided with a fuse of 6 A for preventing the PC output circuit from being snapped and so on due to a short- circuited load occured by wrong operation.

CAUTION

In case of an inductive load whose coil capacity exceeds 9 VA after turned on, a surge killer (such as a 0.1 μF capacitor + 100 Ω resistor combination) must be connected in parallel with the load. For DC load, connect a flywheel diode.

Mounting	Power wiring	I/O wiring	**Terminal Layout**

[Terminal layout]


Terminals marked " • " are unused.
24VDC | C | STA | 1 | 3 | 5 | 7 | 9 | 11 | • | • | • | • | • | • | • | • | • | • | • | •
0V | C | 0 | 2 | 4 | 6 | 8 | 10 | • | • | • | • | • | • | • | • | • | • | • | •
⊖ 24VDC ⊕ | ⏚ | Z
(EC-D type)
EC2-(D)20HR(P)
100~240VAC | C0 | 201 | 203 | 205 | C1 | 207 | • | • | C2 | • | • | C3 | • | • | C4 | • | • | •
⏚ | Z | C0 | 200 | 202 | 204 | C1 | 206 | • | • | C2 | • | • | C3 | • | • | C4 | • | • | • | •
24VDC | C | STA | 1 | 3 | 5 | 7 | 9 | 11 | 13 | 15 | • | • | • | • | • | • | • | • | • | •
0V | C | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | • | • | • | • | • | • | • | • | • | •
⊖ 24VDC ⊕ | ⏚ | Z
(EC-D type)
EC2-(D)28HR(P)
100~240VAC | C0 | 201 | 203 | 205 | C1 | 207 | 209 | 211 | C2 | • | • | C3 | • | • | C4 | • | • | •
⏚ | Z | C0 | 200 | 202 | 204 | C1 | 206 | 208 | 210 | C2 | • | • | C3 | • | • | C4 | • | • | • | •
24VDC | C | STA | 1 | 3 | 5 | 7 | 9 | 11 | 13 | 15 | 21 | 23 | 25 | 27 | • | • | • | • | • | •
0V | C | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 20 | 22 | 24 | 26 | • | • | • | • | • | •
⊖ 24VDC ⊕ | ⏚ | Z
(EC-D type)
EC2-(D)40HR(P)
100~240VAC | C0 | 201 | 203 | 205 | C1 | 207 | 209 | 211 | C2 | 213 | 215 | C3 | • | • | C4 | • | • | •
⏚ | Z | C0 | 200 | 202 | 204 | C1 | 206 | 208 | 210 | C2 | 212 | 214 | C3 | • | • | C4 | • | • | • | •
24VDC | C | STA | 1 | 3 | 5 | 7 | 9 | 11 | 13 | 15 | 21 | 23 | 25 | 27 | 29 | 31 | 33 | 35 | 41 | 43
0V | C | 0 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 20 | 22 | 24 | 26 | 28 | 30 | 32 | 34 | 40 | 42
⊖ 24VDC ⊕ | ⏚ | Z
(EC-D type)
EC2-(D)60HR(P)
100~240VAC | C0 | 201 | 203 | 205 | C1 | 207 | 209 | 211 | C2 | 213 | 215 | C3 | 221 | 223 | C4 | 225 | 227 | •
⏚ | Z | C0 | 200 | 202 | 204 | C1 | 206 | 208 | 210 | C2 | 212 | 214 | C3 | 220 | 222 | C4 | 224 | 226 | • | •

6. Maintenance

Periodic Check	Troubleshooting	Error Display and How to Deal with Error

Periodic Check

Table 6-1 Display in normal status

Status \ Display		Lamp indication		
		POW	RUN	ERR
Normal status	Operation	☼	☼	●
	Stop	☼	●	●

☼ : Lit

● : Extinguished

Other Checkup Items

(1) Abnormal temperature rise due to heat source or direct sunlight
(2) Entrance of dust, chips or wiring scraps into panel
(3) Loosening of wire and terminal connections

[Explanation]

1. The EC series incorporates neither battery nor consumable whose life reaches its end in a short period of time. However, attention must be paid to the service life of the output relay in case it is activated frequently.
2. The aluminum electrolytic capacitor used in the power supply unit also has a limited lifetime.
   In this capacitor, a chemical reaction is taking place. And its lifetime changes widely at different ambient temperatures. Electrolytic capacitor is generally subordinate to the "Arrhenius's equation (double effect rule with change of 10℃)." This signifies that its lifetime is reduced by half with a temperature rise of 10℃ and lengthened to a twofold value with a temperature fall of 10℃.
   The EC series has been designed so as to have a standard lifetime of about 10 years at an in-frame temperature of 40℃. For a longer service life, installation should ensure an adequate ventilation and appropriate ambient temperature.
3. Never use lacquer thinner or the like because such a substance may cause the cover surface to be dissolved or discolored.

Table 6-2 Troubleshooting (1/3)

No.	Phenomenon	Check item	Check result	Remedy
1	POW lamp does not light when turning on power supply.	Check line voltage.	Abnormal	Correct to normal line
			Normal	Exchange the product.
2	Operation does not start though STA input turns on.	connect STA and 24V terminals to check on.	Abnormal	Correct external wiring around start switch.
			Normal	Exchange the product.
		Check programmer switch.	Set at PROG.	Set to TEST or RUN.
		Conduct syntax check by keying in ($\boxed{\text{CLR}}$ $\boxed{\text{SRC}}$)	Error detected.	Correct program.
			Error not detected.	Exchange the product.

Table 6-2 Troubleshooting (2/3)

No.	Phenomenon	Check item	Check result	Remedy
3	During operation, RUN lamp went off and operation stopped. (Or RUN lamp went off shortly after start of operation.)	Check if ERR lamp is lit.	Lit	Eliminate noise source and recheck program. Then restart operation. (If error recurs even after eliminating noise source, the product must be exchanged with a new one.
		Check if shorter program can be run.	Can be run.	Shorten scan time somehow because it is longer than 100 ms.
			Cannot be run.	Exchange the product.
4	Input lamp stays OFF.	Connect the relevant input terminal and 24V terminal to check if the lamp lights up.	Lights up.	Correct external wiring or replace external input device.
			Does not light up.	Utilize unassigned input terminal or exchange the product.
5	Input lamp won't go off.	Open-circuit the relevant input terminal and check if the lamp goes off.	Goes off.	Correct external wiring or exchange external input device.
			Does not go off.	Utilize unassigned input terminal or exchange the product.

Table 6-2 Troubleshooting (3/3)

No.	Phenomenon	Check item	Check result	Remedy
6	Output lamp will not come on or go off.	Monitor the relevant output with programmer and confirm that the lamp status matches the monitored contents.	Matches	Correct program.
			Does not match.	Utilize unassigned output terminal or exchange the product.
7	Output lamp does not meet load ON/OFF status.	Check for conductivity across relevant output terminal and C terminal (with the aid of tester).	Output lamp matches conductive status.	Correct external wiring or exchange external output device
			Output lamp does not match conductive status.	Utilize unassigned output terminal or exchange the product. (If the contacts of internal relay are fused because of excessively large load current, an intermediate relay is required.)

[Explanation]

1. If a trouble occurs on the system under normal operation, we must judge first as to whether the trouble is attributable to the EC series or other section.
   Check and take measure as per the table above.

Table 6-2 Troubleshooting (2/3)

No.	Phenomenon	Check item	Check result	Remedy
3	During operation, RUN lamp went off and operation stopped. (Or RUN lamp went off shortly after start of operation.)	Check if ERR lamp is lit.	Lit	Eliminate noise source and recheck program. Then restart operation. (If error recurs even after eliminating noise source, the product must be exchanged with a new one.
		Check if shorter program can be run.	Can be run.	Shorten scan time somehow because it is longer than 100 ms.
			Cannot be run.	Exchange the product.
4	Input lamp stays OFF.	Connect the relevant input terminal and 24V terminal to check if the lamp lights up.	Lights up.	Correct external wiring or replace external input device.
			Does not light up.	Utilize unassigned input terminal or exchange the product.
5	Input lamp won't go off.	Open-circuit the relevant input terminal and check if the lamp goes off.	Goes off.	Correct external wiring or exchange external input device.
			Does not go off.	Utilize unassigned input terminal or exchange the product.

Table 6-2 Troubleshooting (3/3)

No.	Phenomenon	Check item	Check result	Remedy
6	Output lamp will not come on or go off.	Monitor the relevant output with programmer and confirm that the lamp status matches the monitored contents.	Matches	Correct program.
			Does not match.	Utilize unassigned output terminal or exchange the product.
7	Output lamp does not meet load ON/OFF status.	Check for conductivity across relevant output terminal and C terminal (with the aid of tester).	Output lamp matches conductive status.	Correct external wiring or exchange external output device
			Output lamp does not match conductive status.	Utilize unassigned output terminal or exchange the product. (If the contacts of internal relay are fused because of excessively large load current, an intermediate relay is required.)

[Explanation]

1. If a trouble occurs on the system under normal operation, we must judge first as to whether the trouble is attributable to the EC series or other section.
   Check and take measure as per the table above.

Table 6-3 Syntax and Sum-Check Error Codes

Syntax/sum-check error code (decimal)	Error display on PGMJ	Error display on PGMJ-R2	Description
0	Blank	Blank	No error
1	E	E	Combination of instruction words does not match the syntax rule.
2	E	E	The structure of main routine or interrupt processing routine is abnormal.
3	E	E	There are multiple INT instructions of the relevant number.
4	E	E	The FUN06-FUN07 structure is abnormal.
5	E	E	The FUN08-FUN09 structure is abnormal.
6	⊔	uE	STR level is under that specified for instruction word.
7	⊓	oE	STR level is over that specified for instruction word.
8	⊔	uE	The level of master control is under that specified for instruction word.
9	⊓	oE	The level of master control is over that specified for instruction word.
10	E	E	IF or IFR is duplicated. An impermissible instruction(OUT T/C) is written after IF or IFR.
11	E	E	The I/O number, constant, etc. of instruction word is not within the specified range.
12	E	E	This double coil is impermissible.
13	E	dE	Occurrence of double coil though operation is allowed. (Alarm is issued.)
20	F	fE	Program cannot be interpreted because an undefined operation code or operand is used. Or the user memory area is not formatted correctly.
30	E	E	User program is judged to be abnormal according to the result of sum check.

[Explanation]

1. Syntax check of program is carried out just before operation or by keying in $\boxed{\text{CLR}}$ $\boxed{\text{SRC}}$
2. If error is detected in syntax check, syntax error code can be checked by keying in $\boxed{\text{CLR}}$ $\boxed{9}$ $\boxed{8}$ $\boxed{0}$ $\boxed{\text{MON}}$ $\boxed{\text{MON}}$

Table 6-4 System Error Code

System error code (decimal)	Description
10	Trap interruption has occurred.
11	Stack pointer abnormality is detected.
12	Contradiction to logic is detected.
13	Improbable interruption has occurred.
14	NMI interruption has occurred.
20	Data has not been written successfully in the user program memory.
21	Sum-check error is detected in system ROM.
22	System RAM Read/Write check error.
30	Undefined PCS instruction word is fetched.
31	PCS stack pointer abnormality is detected.
32	Sum-check error has occurred in user program during operation.
40	Received signal has overflowed the buffer.(Communication port)

[Explanation]

1. If the ERR lamp comes on, system error code can be checked by keying in $\boxed{\text{CLR}}$ $\boxed{9}$ $\boxed{7}$ $\boxed{0}$ $\boxed{\text{MON}}$ $\boxed{\text{MON}}$ after turning on power supply again.