FUJI ELECTRIC

FUJI PROGRAMMABLE CONTROLLER

MICREX-F

USER'S MANUAL INSTRUCTIONS

FEH160

Preface

This User's Manual explains the program instructions, memory and system definitions required to create user programs for the MICREX-F Series PCs. Read this manual carefully to ensure correct MICREX-F Series operation.
See the corresponding User's Manual (Hardware) for details of the MICREX-F Series hardware, including system configuration and I/O specifications, and refer to the User's Manual of the program loader for information about the loader operation during programming.
This manual is an English version based on the Japanese User's Manual No. FH160a.

Safety Precautions

Before mounting, wiring, operation, maintenance and inspection of the device, be sure to read the operating instructions carefully to ensure proper operation. The operating instructions should be furnished to the maintenance supervisors of final users.

- Here, the safety precaution items are classified into "Warning" and "Caution".

⚠ **WARNING** : Incorrect handling of the device may result in death or serious injury.

⚠ **CAUTION** : Incorrect handling of the device may result in minor injury or physical damage.

- Even some items indicated by "Caution" may also result in a serious accident.

⚠ WARNING

- Never touch any part of charged circuits as terminals and exposed metal portion while the power is turned ON. It may result in an electric shock to the operator.
- Turn OFF the power before mounting, dismounting, wiring, maintaining or checking, otherwise, electric shock, erratic operation or troubles might occur.
- Place the emergency stop circuit, interlock circuit or the like for safety outside the PC. A failure of PC might break or cause problems to the machine.
- Do not connect in reverse polarity, charge (except rechargeable ones), disassemble, heat, throw in fire or short-circuit the batteries, otherwise, they might burst or take fire.

⚠ CAUTION

- Do not use one found damaged or deformed when unpacked, otherwise, failure or erratic operation might be caused.
- Do not shock the product by dropping or tipping it over, otherwise, it might be damaged or troubled.
- Follow the directions of the operating instructions when mounting the product. If mounting is improper, the product might drop or develop problems or erratic operations.
- Use the rated voltage and current mentioned in the operating instructions and manual. Use beyond the rated values might cause fire, erratic operation or failure.
- Operate (keep) in the environment specified in the operating instructions and manual. High temperature, high humidity, condensation, dust, corrosive gases, oil, organic solvents, excessive vibration or shock might cause electric shock, fire, erratic operation or failure.
- Select a wire size to suit the applied voltage and carrying current, and carry out wiring according to the operating instructions and manual. Poor wiring might cause fire.
- Contaminants, wiring chips, iron powder or other foreign matter must not enter the device when installing it, otherwise, erratic operation or failure might occur.
- Connect the ground terminal to the ground, otherwise, an erratic operation might occur.
- Periodically make sure the terminal screws and mounting screws are securely tightened.
  Operation at a loosened status might cause fire or erratic operation.
- Put the furnished connector covers on unused connectors, otherwise, failure or erratic operation might occur.
- Install the furnished terminal cover on the terminal block, otherwise, electric shock or fire might occur.
- Sufficiently make sure of safety before program change, forced output, starting, stopping or anything else during a run.
  The wrong operation might break or cause machine problems.
- Replace the fuse with a designated one, otherwise, fire or failure might occur.
- Engage the loader connector in a correct orientation, otherwise, an erratic operation might occur.
- Do not remodel or disassemble the product, otherwise, a failure might occur.
- Follow the regulations of industrial wastes when the device is to be discarded.

CONTENTS

Section 1 General

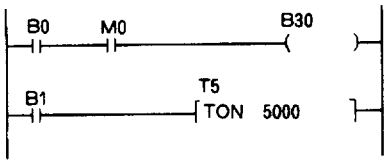
Section 1 General

1-1 How to Use This Manual

This manual explains the MICREX-F functions to help you design a system and create software for the system. Items that are related to your objective should be selected and the corresponding sections should be read.

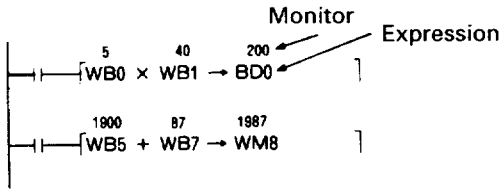
1-1-1 Designing programs

1. To program or monitor relay sequences



Programming by using ladder diagrams
Detailed explanations
- See Sections 3-2 and 3-3 for the instruction words.
- See Section 2-2 for details on memory.

2. To program or monitor computational expressions

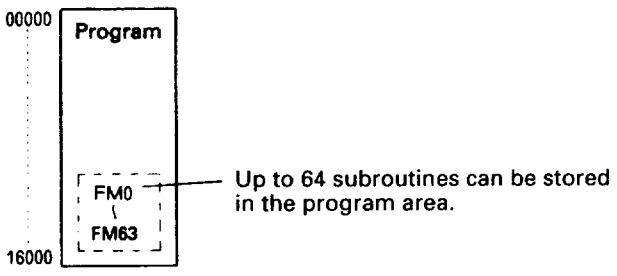


Programming by using line diagrams
Detailed explanations
- See Section 3-4 for the instruction words.
- See Section 2-2 for details on memory.

3. For example, to process a large amount of production line data

File instructions
Detailed explanations
- See Section 3-4-11 for the instruction words.
- See Section 2-2-18 for details on memory.

4. To simplify a program by using subroutines

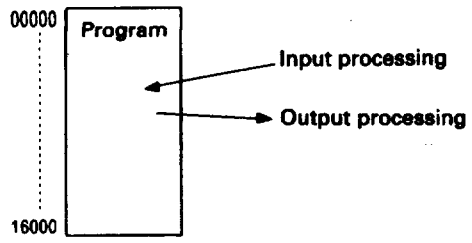


Function modules (FMs)
Detailed explanations
- See Item 2, in Section 2-4-3 for details on the FMs.
- See Item 3 to 5 in Section 3-5-2 for the instruction words.

5. To reduce the number of program steps used to regularly increment and decrement addresses

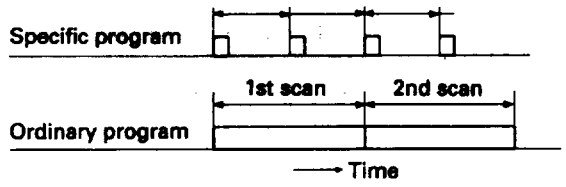
Index instructions
Detailed explanations
- See Item 3 in Section 2-4-3 for the index registers.
- See Section 3-5-4 for the instruction words.

6. To execute I/O processing asynchronously with scanning for I/O devices that require high-speed responses



Direct accessing
Detailed explanations
- See Section 2-2-19 for details on the direct access memory area.
- See Section 4-2-1 for the usage of direct accessing.
- See Section 4-1 for the system definition.

7. To execute a specific program in a fixed cycle-time regardless of scan time changes



Fixed-cycle interrupt function
Detailed explanations
- See Section 2-4-2 for details on the fixed-cycle interrupts.
- See Section 2-5-3 for the instruction words.

8. To execute high-accuracy positioning by using a high-speed counter
   To execute special processing by using external emergency signals

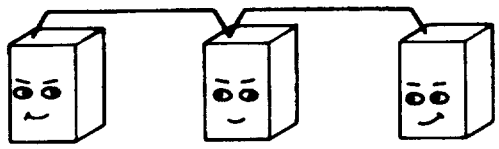
External interrupt function
Detailed explanations
- See Section 2-4-2 for details on the external interrupts.

9. To connect a general-purpose personal computer to execute data transmission with the personal computer.

Detailed explanations
- See Section 2-2-12 for details on memory.
- See Section 4-1 for the system definition.

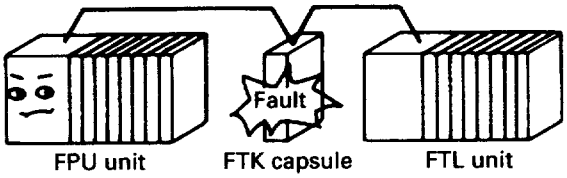
1-1-2 Designing operation mode

1. To connect multiple processors and perform inter-processor data transfer



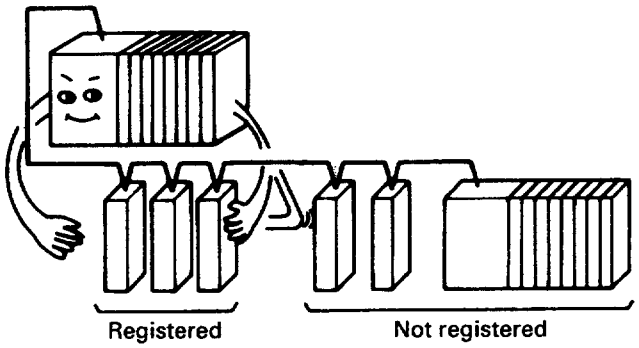
P-link function
Detailed explanations
- See the User's Manual (Communication) for the P-link.
- See Section 2-2-20 for memory.
- See Section 4-1 for the system definition.

2. To disconnect only the faulty unit and continue operation with the normal units in case of fault in the I/O units during PC operation



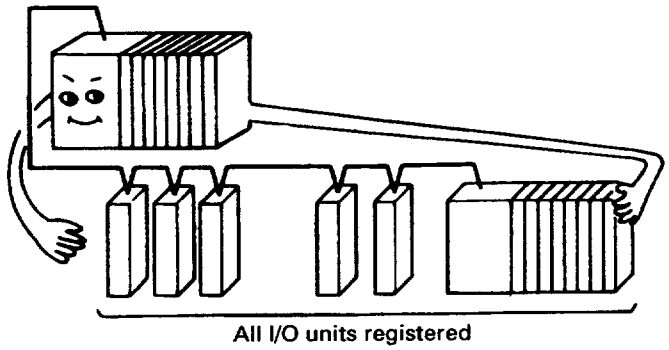
Fail-soft operation function
Detailed explanations
- See the User's Manual (Communication) for the fail-soft operation.
- See Section 4-1 for the system definition.

3. To phase in and verify the control range due to the large number of system I/O points



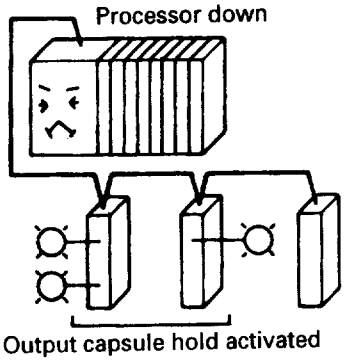
T-link registration function
Detailed explanations
- See the User's Manual (Communication) for the T-link function.
- See Section 4-1 for the system definition.

4. I/O units are distributed. The system operation should start only after all the I/O units are powered on. All the I/O units should be registered in advance so that the system operation can start on condition that power-on of all of I/O units is recognized.



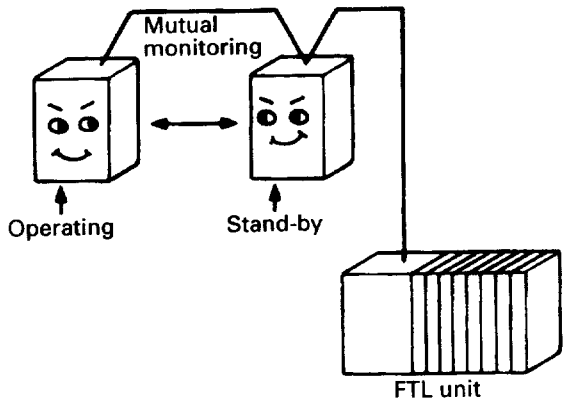
T-link registration function
Detailed explanations
- See the User's Manual (Communication) for the T-link function.
- See Section 4-1 for the system definition.

5. To retain the indicator circuit and cooling unit status when a system failure stops the processor operation



Hold registration function
Detailed explanations
- See the User's Manual (Communication) for the description of the hold registration.
- See Section 4-1 for the system definition.

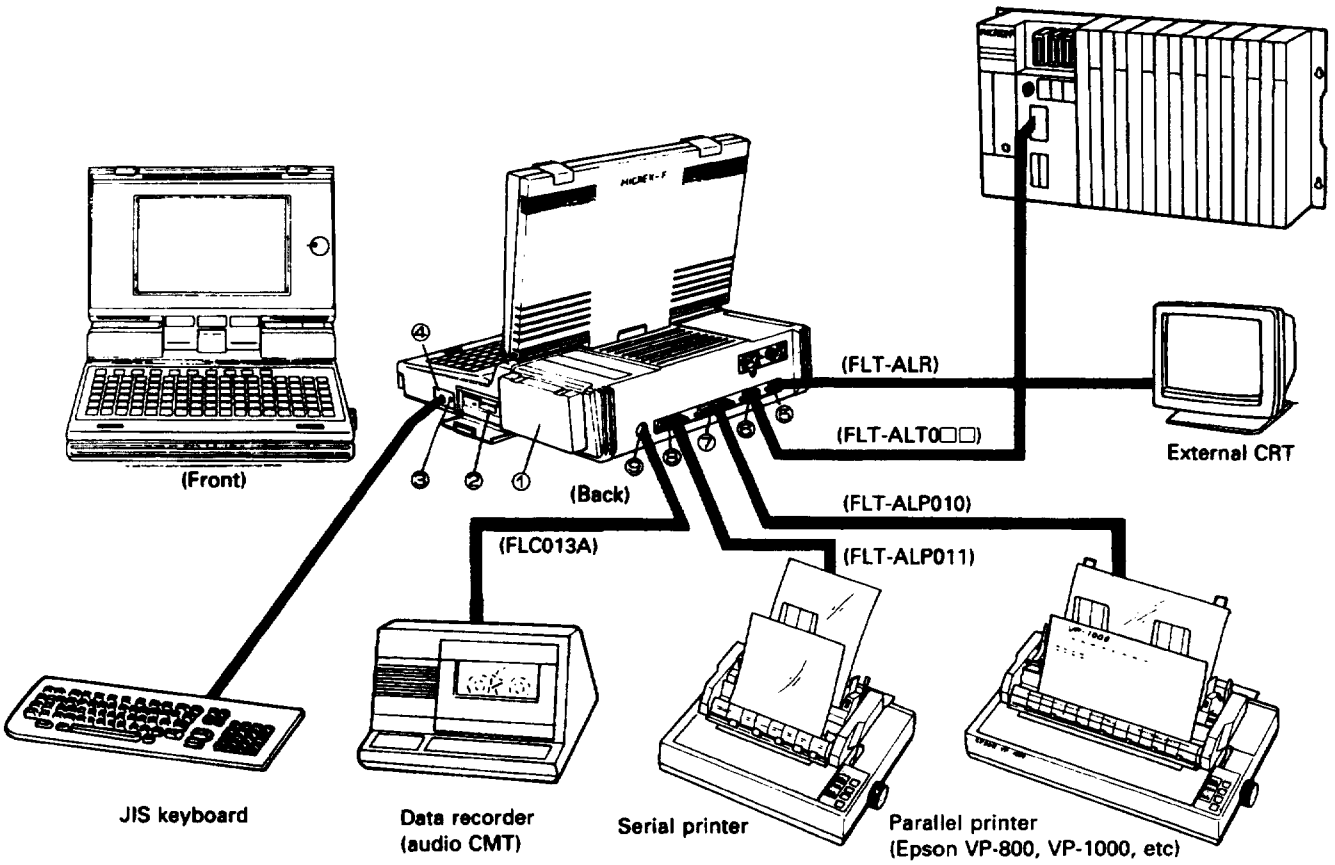
6. Continue the system operation with an auxiliary (stand-by) processor in case of trouble in the main processor itself



Duplex (Backup) function
Detailed explanations
- See the User's Manual (Communication) for the description of the duplex processor mode (backup).
- See Section 4-1 for the system definition.

1-2 Introduction to The Program Loader, LITE

The MICREX-F Series includes a multifunction program loader, LITE. This program loader enables the user to write, modify, and read user programs. It also enables the user to have an interface with printers, data recorders, and other devices to store and edit user programs. Major functions of the LITE are explained below.



① **PROM writer interface**
The LITE has a dedicated PROM writer as standard equipment. The PROM writer can be used to write and read programs in units of PROM chip or cassette.

② **3.5-inch floppy disk drive**
The LITE has two 3.5-inch disk drives (one on each side) as standard equipment. The floppy disk drives have a function to automatically switch for 2DD or 2HD floppy disk. Floppy disk drives have dustproof covers to ensure the use of the LITE in a factory.

③ **Protection key**
This key disables the LITE to write programs in processors.

④ **JIS keyboard interface**

⑤ **External CRT (RGB) interface**
The screen display on the LITE can be enlarged on an external CRT.

⑥ **T-link interface**
This interface is used to connect the LITE to an MICREX-F Series processor.

⑦ **Printer (with Centronics interface)**
Various types of printers can be connected.

⑧ **RS232C/422 interface**
A serial printer can be connected.

⑨ **Data recorder interface**
An audio CMT can be connected to store and read programs on audio cassette tape and read them.

Section 2 Specifications

Section 2 Specifications

2-1 Basic Specifications

1. F30, F50, F50H Series processors

Item		Specification						Remarks
		FPB20	FPB30	FPB40	FPB56	FPB40H	FPB56H	
Control system		Stored program, cyclic scanning						
I/O connection system		Direct I/O, remote I/O (T-link, mini T-link)						*1
I/O control system		Synchronous refreshing						
CPU		Sequence-dedicated processor, 8-bit processor						
Memory	Program section	EEPROM (standard)		IC-RAM (standard)				
	Data section	IC-RAM (standard)		IC-RAM (standard)				
Available memory		—		EPROM		EEPROM, EPROM		For user program backup *2
Programming language		Ladder diagram language (FPL common to MICREX-F Series)						F-series Programming Language
Instruction	Sequence instruction	14 types						
	Data instruction	39 types						
Instruction word length		Sequence instruction (contact): 1 step/instruction Data instruction (transfer): 3 steps/instruction						
Instruction execution time		Sequence instruction (contact): 1.4μs/instruction, Data instruction (transfer): 600μs/instruction		Sequence instruction (contact): 1.0μs/instruction, Data instruction (transfer): 750μs/instruction		Sequence instruction (contact): 0.8μs/instruction, Data instruction (transfer): 600μs/instruction		
Numeric operation data format		Signed BCD 8 digits						
Program memory capacity		2,386 steps				7,848 steps		
No. of I/O points	Basic unit + expansion unit	Max. 36 points	Max. 46 points	Max. 160 points	Max. 176 points	Max. 208 points	Max. 224 points	
	Basic unit + expansion unit + T-link system	Max. 1,600 points						
	Built-in high-speed counter	1 point (2kHz)		—		1 point (2kHz)		24V DC input processors only
	Built-in pulse input	1 point (min. pulse width 250μs or more)		—		1 point (min. pulse width 250μs or more)		
Relay and word memory	I/O relay (B)	1,600 points						
	Auxiliary relay (M)	512 points *3		512 points		512 points *3		
	Keep relay (K)	512 points *4						
	Differential relay (D)	512 points						
	Special relay (F)	480 points						
	Annunciator relay (A)	320 points						
	Timer (0.01sec.) (T)	128 points						BCD 8 digits
	Counter (C)	32 points						BCD 8 digits
	Data memory (BD)	128 words						1 word = 32 bits
	File memory (W30~)	64 words (1 word = 32 bits), 128 words (1 word = 16 bits)						Specify in the file definition instruction.
	Step control relay (S)	10,000 steps (100 steps x 100 sets)						
T-link		1 link						T-link adapter required.
Memory (IC-RAM) backup method		Backup by super capacitor 55°C: 72 hours, 25°C: 2 weeks		Backup by primary lithium battery (5 year life) 55°C: 500 days, 25°C: 5 years *5				F30 Series does not require user program memory backup due to standard equipment of EEPROM.
Self-diagnosing function		Operation time monitoring (watchdog timer; max. 2.56s.), low battery voltage detection, program syntax check, system configuration monitoring, unit fault detection						

Notes:
*1: Connection to a T-link device requires the T-link adapter (FTM050A). Connection to a mini T-link device requires the T-link adapter and T-link converter (FRC100A-G2).
*2: Use EPROM type FMC032A (equivalent to 27C256A).
Use EEPROM type FMC041A (HN58C65P-25: Hitachi).
When EEPROM is used, the program capacity is 2.3K steps.
*3: M031F is used as the pulse input detection relay, thus, the auxiliary relay has 511 I/O points.
*4: For F30 Series, K029F is used for the BER (backup error) lamp. When C0031 is programmed as an increment/decrement counter in the F30/F50H Series, K0300 to K031F are used as the current value copy area of the high-speed counter.
*5. Replace with a new battery (FTB020A) when the battery alarm lamp is lit or the period of validity expires.

2. F60 Series processors

Item			Specification	Remarks
Control system			Stored program, cyclic scanning	
I/O connection system			Remote I/O (mini T-link). Connection to a T-link device requires the T-link converter.	
I/O control system			Synchronous refreshing	
CPU			Sequence-dedicated processor, 16-bit processor	
Memory	Program section		IC-RAM/EPROM/EEPROM	
	Data section		IC-RAM	
Available memory			EPROM, EEPROM	*1
Programming language			Ladder diagram language (FPL common to MICREX-F Series)	F-series Programming Language
Instruction	Sequence instruction		19 types	Basic instructions
	Data instruction		48 types	
Instruction word length			Sequence instruction (contact): 1 step/instruction Data instruction (transfer): 3 steps/instruction	
Instruction execution time			Sequence instruction (contact): 0.8μs/instruction Data instruction (transfer): 112μs/instruction	
Numeric operation data format			Signed BCD 8 digits	
Program memory capacity			EEPROM: 2,386 steps, 5,117 steps, 10,578 steps (order specified) IC-RAM: 10,578 steps (standard) EPROM: 10,578 steps (option)	
No. of I/O points	Basic unit + expansion unit (mini T-link system)		Max. 560 points: 1 x (basic unit 56 points) + 9 x (expansion unit 56 points)	
	Basic unit + expansion unit (mini T-link system) + T-link system		Max. 1,600 points	
	Built-in high-speed counter		1 point (24V input: 2kHz, 121V input: 0.5kHz)	Supported by 12-24V DC input processors only.
	Built-in pulse input		1 point (min. pulse width: 250μs at 24V input, 1ms at 12V input)	
Relay and word memory	I/O relay (B)		1,600 points	
	Auxiliary relay (M)		2,048 points *2	
	Keep relay (K)		1,024 points *3	
	Differential relay (D)		512 points	
	Special relay (F)		480 points	
	Annunciator relay (A)		320 points	
	Timer (T)	0.01s	256 points	BCD 8 digits
		0.1s	256 points	BCD 8 digits
	Counter (C)		128 points	BCD 8 digits
	Data memory (BD)		256 words	1 word = 32 bits
	File memory (W30~)		512 words (1 word = 32 bits), 1,024 words (1 word = 16 bits)	Specify in the file definition instruction.
	Step control relay (S)		10,000 steps (100 steps x 100 sets)	
Mini T-link			1 link, no. of connectable stations: 9 units (including the T-link converter if used)	Standard
Mini T-link + T-link			1 link, no. of connectable stations: 32 units (including the T-link converter)	
Memory (IC-RAM) backup method			Backup by super capacitor 55°C: 7 days, 25°C: 30 days	
Self-diagnosing function			Operation time monitoring (watchdog timer; max. 2.56s), low capacitor voltage detection, program check, system configuration monitoring, unit fault detection	

Notes:

*1. Use EPROM type FMC032A (equivalent to 27C256A).
Use EEPROM type FMC041A (x 1) for 2.3K steps, FMC041A (x 2) for 5.1K steps, and FMC043A (x 1) for 10.5K steps.

*2. M127F is used as the pulse input detection relay, thus, the auxiliary relay has 2,048 I/O points.

*3. K61F is used for the BER (backup error) lamp.
When C127 is programmed as an increment/decrement counter, K0620 to K063F are used as the current value copy area of the high-speed counter.

Section 2 Specifications

3. F55 Series processors

Item			Specification	Remarks
Control system			Stored program, cyclic scanning, fixed-cycle interrupt, external interrupt	
I/O connection system			Direct I/O, remote I/O (T-link, mini T-link)	
I/O control system			Synchronous refreshing, direct I/O *6	
CPU			Sequence-dedicated processor, 16-bit processor	
Memory	Program section		IC-RAM	
	Data section		IC-RAM	
Available memory			EPROM, EEPROM *1	
Programming language			Ladder diagram language (FPL common to MICREX-F Series)	F-series Programming Language
Instruction	Sequence instruction		22 types	Basic instructions
	Data instruction		76 types	
Instruction word length			Sequence instruction (contact): 1 step/instruction Data instruction (transfer): 3 steps/instruction	
Instruction execution time			Sequence instruction (contact): 0.5μs/instruction Data instruction (transfer): 118μs/instruction	
Numeric operation data format			Signed BCD 8 digits	
Program memory capacity			10,578 steps	
No. of I/O points			Max. 1,600 points	
Relay and word memory	I/O relay	(B)	1,600 points + 4,800 points *2	
	Direct I/O	(W24)	256 words	1 word = 16 bits
	Auxiliary relay	(M)	4,096 points	
	Keep relay	(K)	1,024 points	
	Differential relay	(D)	1,024 points	
	Special relay	(F)	1,120 points	
	Annunciator relay	(A)	320 points	
	Timer (T)	0.01s	256 points	BCD 8 digits
		0.1s	256 points (current value area W9.0~)	BCD 8 digits
	Counter	(C)	256 points	BCD 8 digits
	Data memory	(BD)	256 words (expandable up to 4,095 words using the file memory area)	1 word = 32 bits
	File memory	(W30~)	3,840 words	Specify in the file definition instruction.
	Step control relay	(S)	10,000 steps (100 steps x 100 sets)	
T-link			1 link (optional T-link master card required)	
Memory (IC-RAM) backup method			Backup by user program memory and primary lithium battery (5 year life) 25°C: 5 years *3	
Self-diagnosing function			Operation time monitoring (watchdog timer; max. 2.56s), low battery voltage detection, program syntax check, system configuration monitoring, unit fault detection, maintenance display function *4	
User display function (W124)			Displays symbols and numbers according to the user program. *4	
Calendar function (W125)			Year, month, day, hour, minute, second (available up to year 2088)	Precision: 1s/day at 25°C

Notes

*1. Use EPROM card type NV1V-MP10 (10.5K steps) and EEPROM card type NV1V-ME10 (10.5K steps). The EEPROM card can be used for version 1001 or subsequent.

*2. 4,800 points from B1000 to B399F can be used only as an auxiliary relay.

*3. Replace with a new battery (FBT030A<NL8V-BT>) when the battery alarm lamp is lit or the period of validity expires.

*4. For the maintenance display function and user display function, see the F55 Series User's Manual (Hardware) FEH150.

*5. The maintenance display function, user display function, and calendar function are supported only by version 1001 or subsequent.

*6. When the direct I/O (direct access) mode is used, the expansion unit also enters the direct access mode. However, the interrupt card (NV1F-YP1) and high-speed counter card (NV1F-HC1) cannot be mounted on the expansion unit.

4. F80, F120H Series processors

Item			Specification		Remarks
			FPU080H	FPU120H	
Control system			Stored program, cyclic scanning, fixed-cycle interrupt, external interrupt		
I/O connection system			Direct I/O, remote I/O (T-link, mini T-link)		
I/O control system			Synchronous refreshing, direct I/O (only on processor base)		
CPU			Sequence-dedicated processor, 16-bit processor		
Memory	Program section		IC-RAM		
	Data section		IC-RAM		
Available memory			EPROM *1		
Programming language			Ladder diagram language (FPL common to MICREX-F Series)		F-series Programming Language
Instruction	Sequence instruction		22 types		Basic instructions
	Data instruction		76 types	84 types	
Instruction word length			Sequence instruction (contact): 1 step/instruction Data instruction (transfer): 3 steps/instruction		
Instruction execution time			Sequence instruction (contact): 0.8μs/instruction Data instruction (transfer): 189μs/instruction	Sequence instruction (contact): 0.8μs/instruction Data instruction (transfer): 121μs/instruction	
Numeric operation data format			Signed BCD 8 digits		
Program memory capacity			10,578 steps		
No. of I/O points			Max. 1,600 points	Max. 6,400 points	
Relay and word memory	I/O relay	(B)	1,600 points + 4,800 points *2	6,400 points (1,600 points x 4)	
	Direct I/O	(W24~)	160 words	128 words	1 word = 16 bits
	Auxiliary relay	(M)	4,069 points		
	Keep relay	(K)	1,024 points		
	Differential relay	(D)	1,024 points		
	Special relay	(F)	1,088 points	1,760 points	
	Annunciator relay	(A)	352 points		
	Timer (T)	0.01s	256 points		BCD 8 digits
		0.1s	256 points (current value area W9.0~)		BCD 8 digits
	Counter	(C)	256 points		BCD 8 digits
	Data memory	(BD)	256 words (expandable up to 4,095 words using the file memory area)		1 word = 32 bits
	File memory	(W30~)	3,840 words	5,888 words	Specify in the file definition instruction.
	Step control relay	(S)	10,000 steps (100 steps x 100 sets)		
	Index register		—	i, m j, k, l, m	
T-link			1 link (standard)	1 link (standard) + 2 links (option)	No. of connectable stations: 32 units (1 link)
Option card	T-link		—	Max. 2 channels	Total 2 channels
	P-link		—	Max. 2 channels	
	SUMINET		—	Max. 1 channel	
Memory (IC-RAM) backup method			Backup by user program memory and primary lithium battery (5 year life) 25°C: 5 years *3		
Self-diagnosing function			Operation time monitoring (watchdog timer; max. 2.56s), low battery voltage detection, program check, system configuration monitoring, unit fault detection		
Function module			—	○	
Duplex processor system			—	○	

Notes
*1. F80H: ROM socket (FMC034S) and EPROM (FMC032A: equivalent to 27C256A) are required.
F120H: ROM socket (FMC334A) and EPROM (FMC036A: equivalent to 27C100A x 2) are required.
*2. 4,800 points from B1000 to B399F can be used only as an auxiliary relay.
*3. Replace with a new battery (FBT030A<NL8V-BT>) when the battery alarm lamp is lit or the period of validity expires.

Section 2 Specifications

5. F70, F70S Series processors

Item			Specification			Remarks
			F70 (NC1P-E0)	F70S (NC1P-S0)	F70S (NC1P-S2)	
Control system			Stored program, cyclic scanning, fixed-cycle interrupt, external interrupt			
I/O connection system			Direct I/O, remote I/O (T-link, mini T-link)			
I/O control system			Synchronous refreshing, direct I/O (only on processor base)			
CPU				32-bit-dedicated processor		
Memory	Program section		IC-RAM			
	Data section		IC-RAM			
Available memory			EPROM *1			
Programming language			Ladder diagram language (FPL common to MICREX-F Series)			F-series Programming Language
Instruction	Sequence instruction		22 types			Basic instructions
	Data instruction		76 types	99 types		
Instruction word length			Sequence instruction (contact): 1 step/instruction, data instruction (transfer): 3 steps/instruction			
Instruction execution time			Sequence instruction (contact): 0.5μs/instruction Data instruction (transfer): 118μs/instruction	Sequence instruction (contact): 0.125μs/instruction Data instruction (transfer): 0.25μs/instruction		
Numeric operation data format			Signed BCD 8 digits			
Program memory capacity			10,578 steps	16,041 steps		
No. of I/O points			Max. 1,600 points	Max. 2,240 points *2	Max. 5,312 points *3	
Relay and word memory	I/O relay	(B)	1,600 points + 4,800 points*4	6,400 points + 1,792 points *5		
	Direct I/O	(W24)	160 words	160 words	128 words	1 word = 16 bits
	Auxiliary relay	(M)	4,096 points	8,192 points		
	Keep relay	(K)	1,024 points			
	Differential relay	(D)	1,024 points			
	Special relay	(F)	1,088 points	2,016 points		
	Annunciator relay	(A)	352 points	736 points		
	Timer (T) 0.01s		256 points	512 points		
	Timer (T) 0.1s		256 points	488 points		
	Counter	(C)	256 points	512 points		
	Data memory	(BD)	256 words *7			
	File memory	(W30~)	3,840 words	5,888 words		
	Step control relay	(S)	10,000 steps (100 steps x 100 sets)			
	Index register		—	i, j, k, l, m		
T-link			1 link (standard)		1 link (standard) + 2 links (optional)	No. of connectable stations: 32 units (1 link)
Memory (IC-RAM) backup method			Backup by primary lithium battery (5 year life) 25°C: 5 years *6			
Self-diagnosing function			Operation time monitoring (watchdog timer; max. 2.56s), low battery voltage detection, program check, system configuration monitoring, unit fault detection			
Function module			—	○		
Calendar function (W125)			Year, month, day, hour, minute, second (available up to year 2088)			Precision: 1s/day at 25°C

Notes
*1. Memory card NC1VMP-16 is required.
*2. 1 T-link (1,600 points) + processor-mounted base (64 points x 10)
*3. 3 T-links (1,600 points x 3) + processor-mounted base (64 points x 8)
*4. 4,800 points from B1000 to B399F can be used only as an auxiliary relay.
*5. 1,792 points from B4000 to B511F can be used only as an auxiliary relay.
*6. Replace with a new battery (FBT030A<NL8V-BT>) when the battery alarm lamp is lit or the period of validity expires.
*7. The file memory area can be used to expand data memory up to 4,095 words for the F70 Series and up to 4,096 words for the F70S Series.

6. F120S, F140S, F150S Series processors

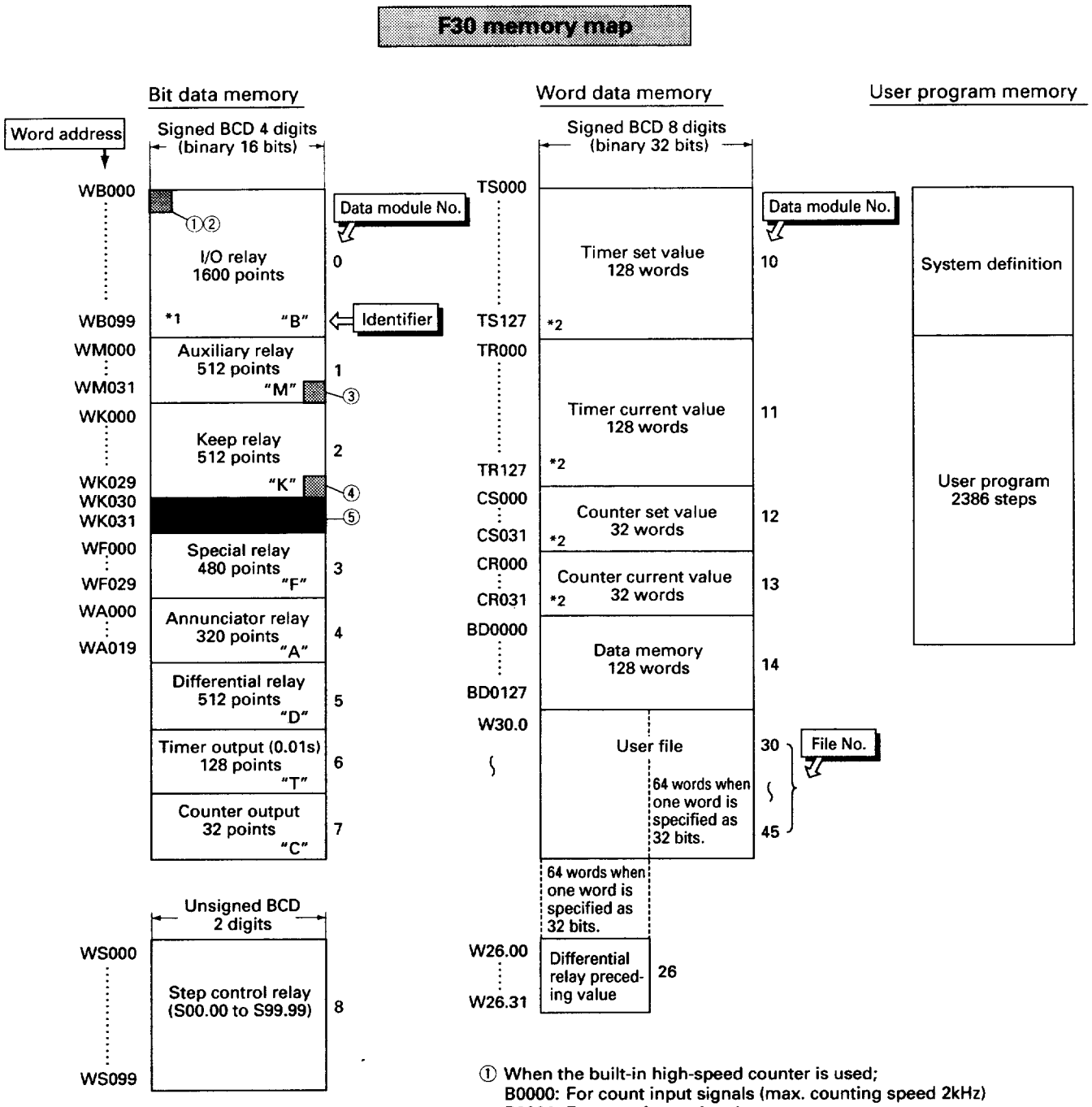
Item			Specification			Remarks
			F120S	F140S	F150S	
Control system			Stored program, cyclic scanning, fixed-cycle interrupt, external interrupt			
I/O connection system			Direct I/O, remote I/O (T-link, mini T-link)			
I/O control system			Synchronous refreshing, direct I/O (only on processor base)			
CPU			32-bit-dedicated processor			
Memory		Program section	IC-RAM			
		Data section	IC-RAM			
Available memory			EPROM *1			
Programming language			Ladder diagram language (FPL common to MICREX-F Series)			F-series Programming Language
Instruction		Sequence instruction	22 types			Basic instructions
		Data instruction	99 types			
Instruction word length			Sequence instruction (contact): 1 step/instruction, data instruction (transfer): 3 steps/instruction			
Instruction execution time			Sequence instruction (contact): 0.125μs/instruction Data instruction (transfer): 0.25μs/instruction			
Numeric operation data format			Signed BCD 8 digits			
Program memory capacity			16,041 steps	32,425 steps	65,193 steps	
No. of I/O points			Max. 6,400 points			
Relay and word memory	I/O relay	(B)	6,400 points + 1,792 points *2			
	Direct I/O	(W24~)	128 words			1 word = 16 bits
	Auxiliary relay	(M)	8,129 points			
	Keep relay	(K)	1,024 points			
	Differential relay	(D)	1,024 points			
	Special relay	(F)	2,016 points			
	Annunciator relay	(A)	736 points			
	Timer (T)	0.01s	512 points			
		0.1s	488 points			
	Counter	(C)	512 points			
	Data memory	(BD)	256 words (expandable up to 4,096 words using the file memory area)			
	File memory	(W30~)	5,888 words	102,144 words	331,520 words	
	Step control relay	(S)	10,000 steps (100 steps x 100 sets)			
	Index register		i, j, k, l, m			
T-link			1 link (standard) + 2 links (optional)			No. of connectable stations: 32 units (1 link)
Option card	T-link		Max. 2 channels	Max. 2 channels	Max. 2 channels *3	No. of mountable option cards F120S: Max. 2 F140S: Max. 2 F150S: Max. 2 F152S: Max. 4 F154S: Max. 6
	P-link		Max. 2 channels	Max. 2 channels	Max. 2 channels *4	
	SUMINET		Max. 1 channel	Max. 1 channel	Max. 1 channel	
	PE-link		Max. 2 channels	Max. 2 channels	Max. 2 channels *4	
	ME-NET		Max. 1 channel	Max. 1 channel	Max. 1 channel *5	
Memory (IC-RAM) backup method			Backup by primary lithium battery (5 year life)			
	Backup time		5 years at 25°C		4 years at 25°C	*6
Self-diagnosing function			Operation time monitoring (watchdog timer; max. 2.56s), low battery voltage detection, program check, system configuration monitoring, unit fault detection			
Function module			○			
Calendar function (W125~)			Year, month, day, hour, minute, second (available up to year 2088)			Precision: 1s/day at 25°C

Notes
*1. ROM socket (FMC334A) and EPROM (FMC036A: equivalent to 27C100A x 2) are required.
*2. 1,792 points from B4000 to B511F can be used only as an auxiliary relay.
*3. One standard link and up to 3 option links for FPU152S and FPU154S.
*4. Total No. of P-link + PE-link: Up to 2 channels
*5. Max. 2 channels for F152S and F154S
*6. Replace with a new battery (FBT030A<NL8V-BT>) when the battery alarm lamp is lit or the period of validity expires.

2-2 Memory

2-2-1 Memory maps

1. F30 Series memory map

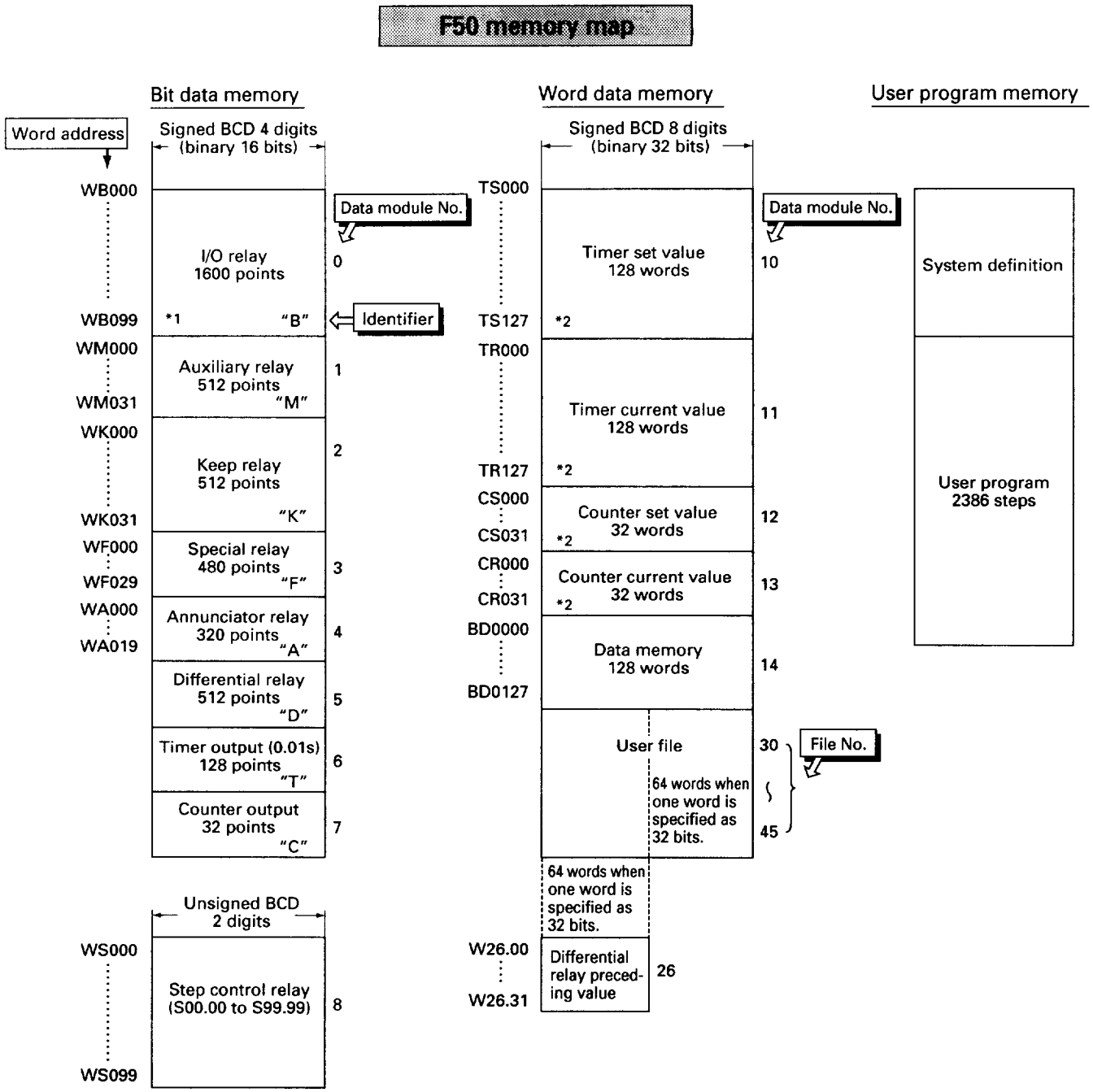


① When the built-in high-speed counter is used;
B0000: For count input signals (max. counting speed 2kHz)
B0001: For reset input signals
② When the pulse input detection relay is used;
B0001: For pulse inputs (min. pulse width 250μs)
③ M31F: Pulse input detection relay
④ K29F: Memory backup error relay
⑤ K0300 to K031F: Copy area for the built-in high-speed counter current value in CR31.

*1 The I/O relay without I/O assigned can be used as an auxiliary relay. (However, the empty input area of each unit is not available.)

*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

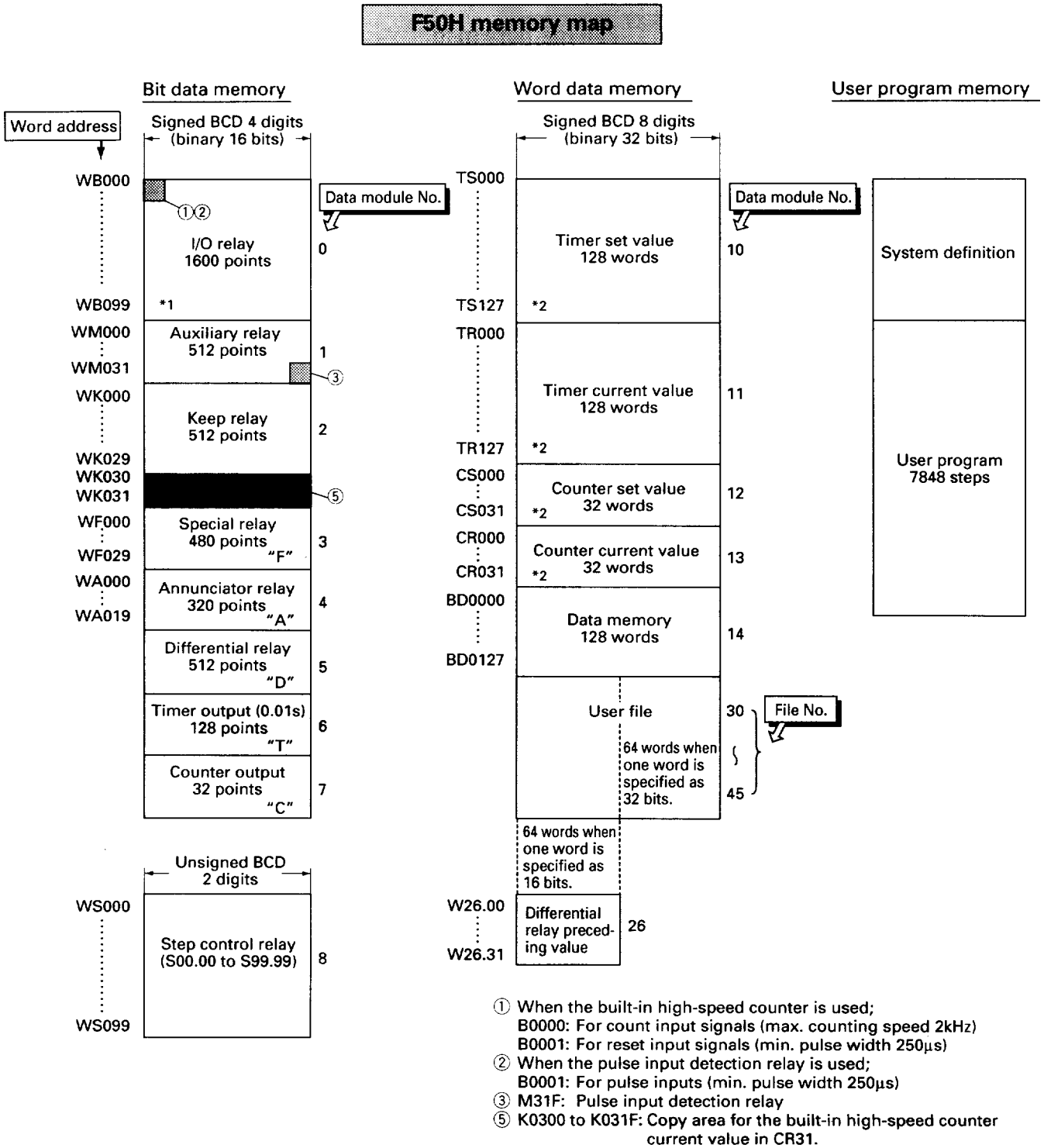
2. F50 Series memory map



*1 The I/O relay without I/O assigned can be used as an auxiliary relay. (However, the empty input area of each unit is not available.)

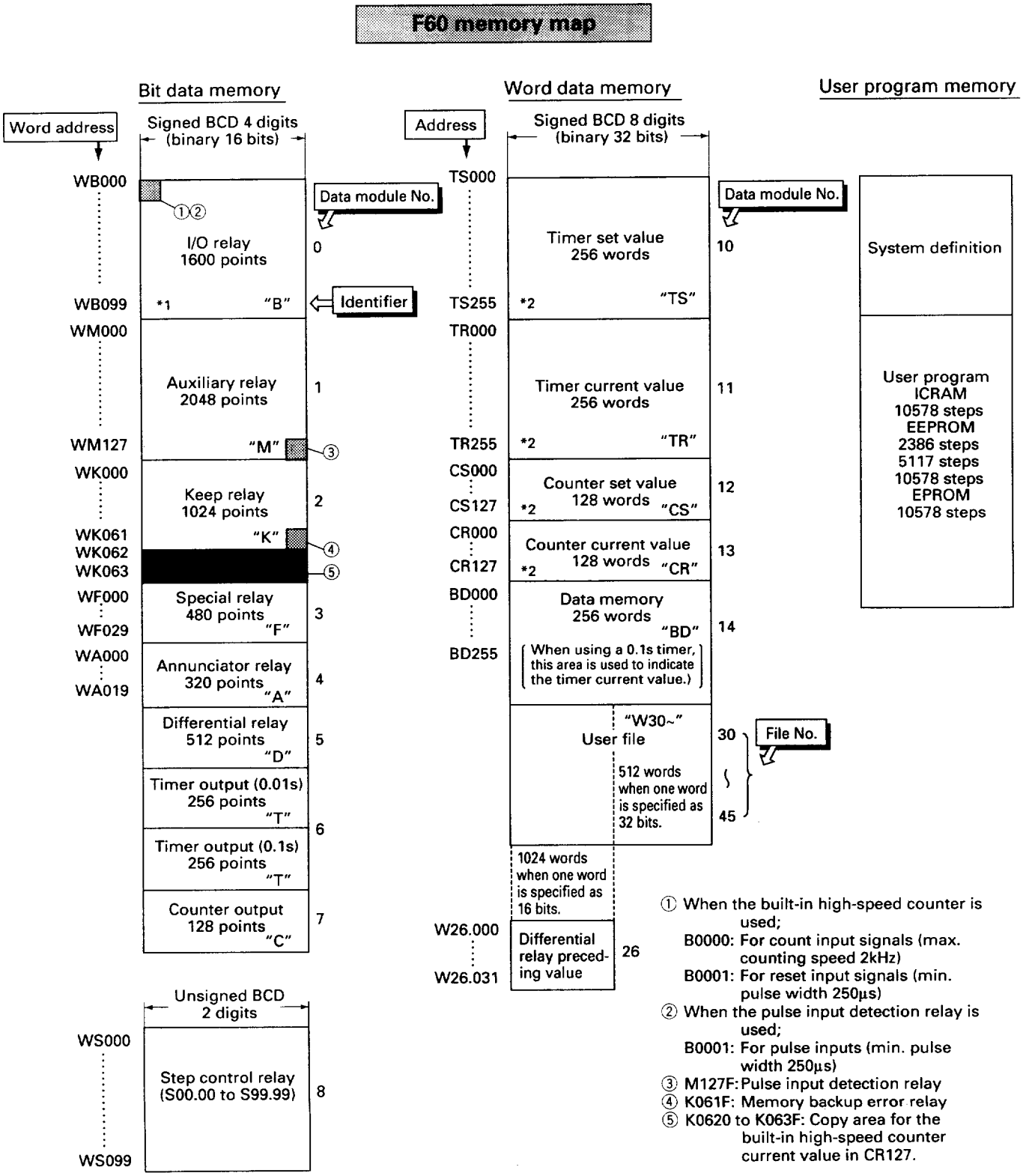
*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

3. F50H Series memory map



*1 The I/O relay without I/O assigned can be used as an auxiliary relay. (However, the empty input area of each unit is not available.)
*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.
*3 The high-speed counter function and pulse input function can be used only with the 12-24V DC input specification. For the F50H Series with the 100/200V AC input specification, C0031 cannot be programmed as a normal increment/decrement counter.

4. F60 Series memory map



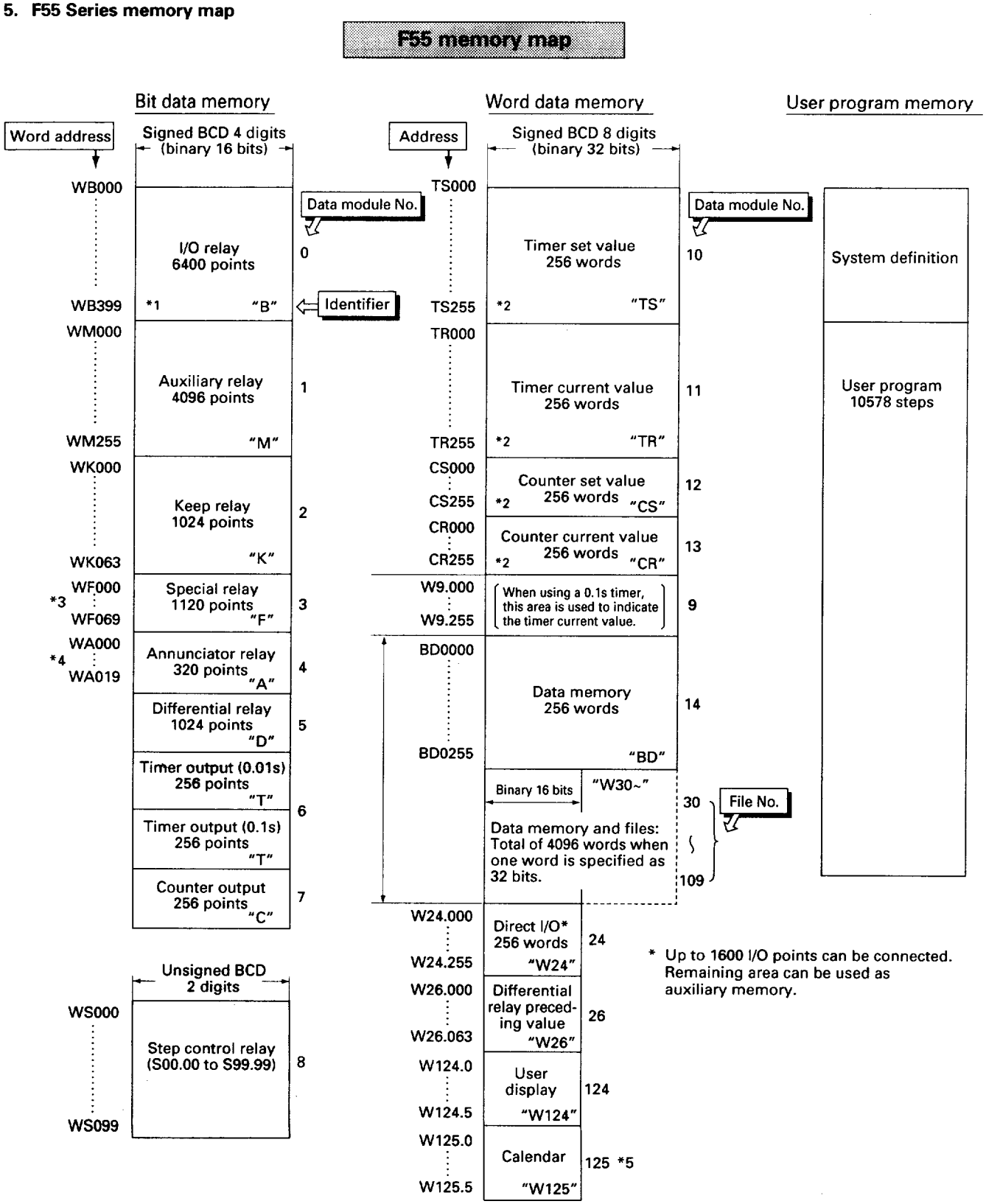
① When the built-in high-speed counter is used;
B0000: For count input signals (max. counting speed 2kHz)
B0001: For reset input signals (min. pulse width 250μs)
② When the pulse input detection relay is used;
B0001: For pulse inputs (min. pulse width 250μs)
③ M127F: Pulse input detection relay
④ K061F: Memory backup error relay
⑤ K0620 to K063F: Copy area for the built-in high-speed counter current value in CR127.

*1 The I/O relay without I/O assigned can be used as an auxiliary relay. (However, the empty input area of each unit is not available.)

*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

*3 The high-speed counter function and pulse input function can be used only with the 12-24V DC input specification. For the F60 Series with the 100/200V AC input specification, C127 cannot be programmed as a normal increment/decrement counter.

5. F55 Series memory map



* Up to 1600 I/O points can be connected. Remaining area can be used as auxiliary memory.

*1 The I/O relay without I/O assigned can be used as an auxiliary relay. (However, the empty input area of each unit is not available.)

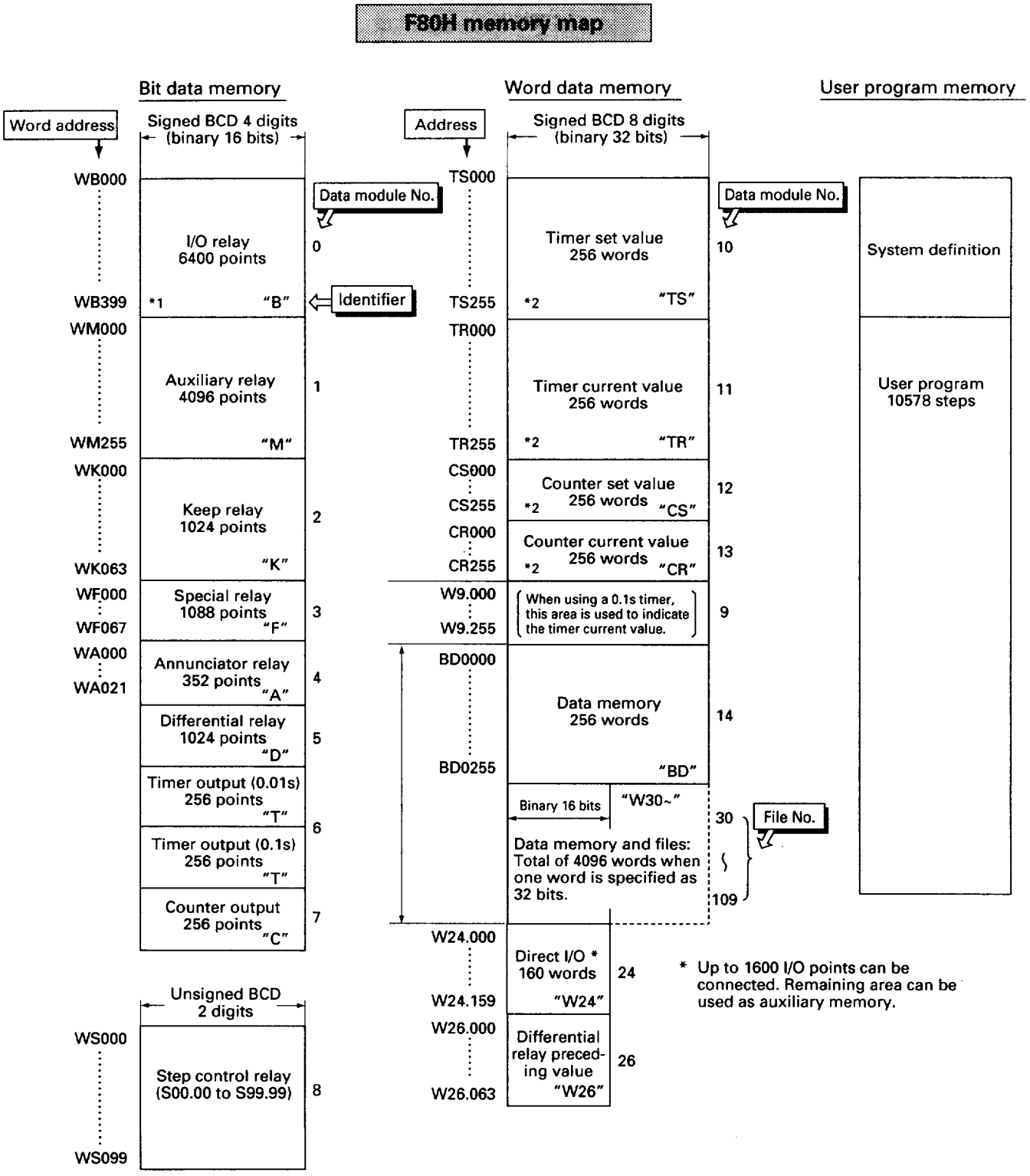
*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

*3 The F55 Series has the option configuration flag and option error flag, resulting in a total of 1120 points from WF000 to WF069.

*4 The F55 Series does not support the sampling trace and status latch functions, resulting in a total of 320 points from WA000 to WA019.

*5 The calendar function can be used only when the T-link master interface card (NV1L-TL1) is mounted on the basic unit.

6. F80H Series memory map

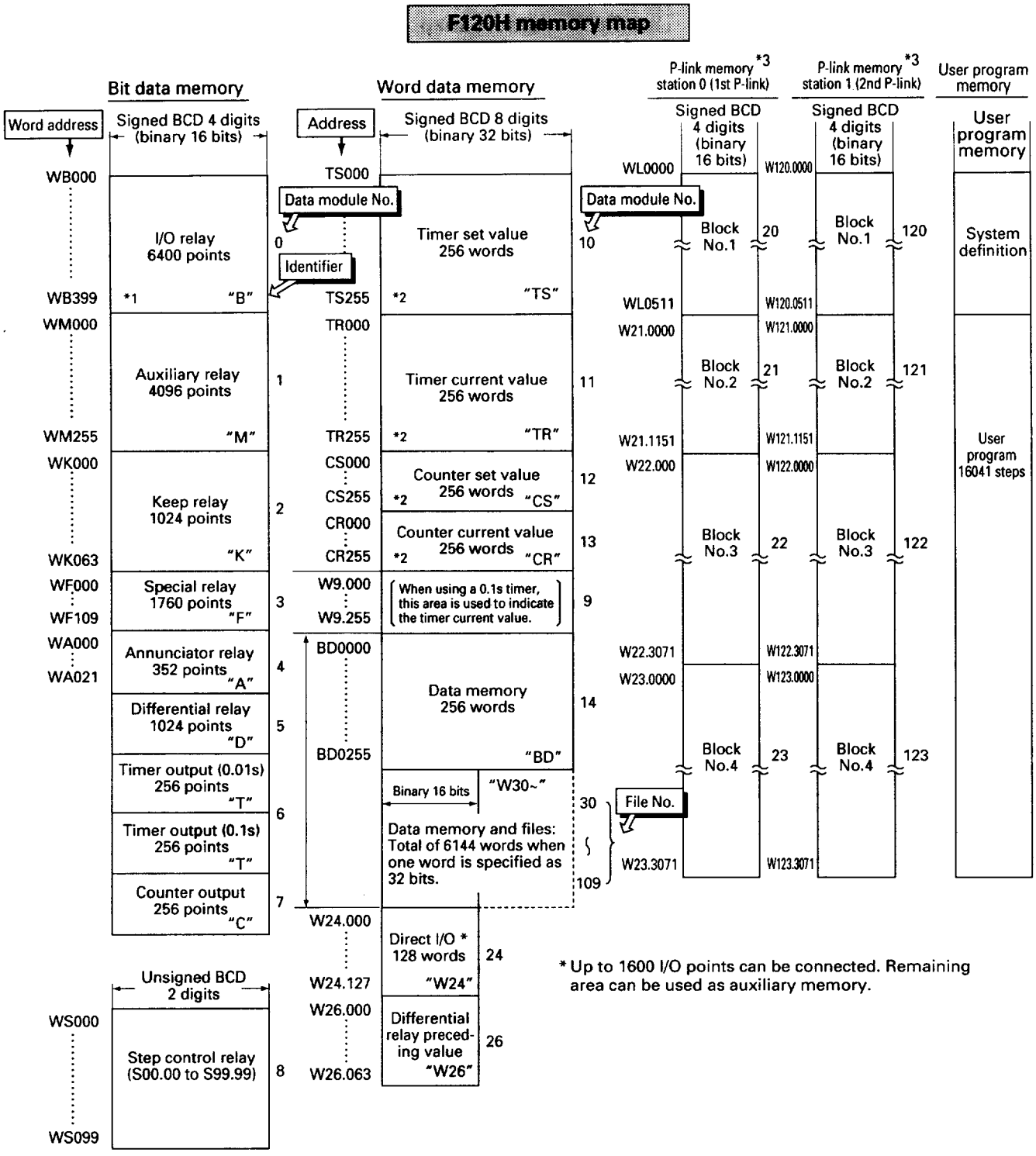


*1 The I/O relay without I/O assigned can be used as an auxiliary relay. (However, the empty input area of each unit is not available.)

*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

Section 2 Specifications

7. F120H Series memory map



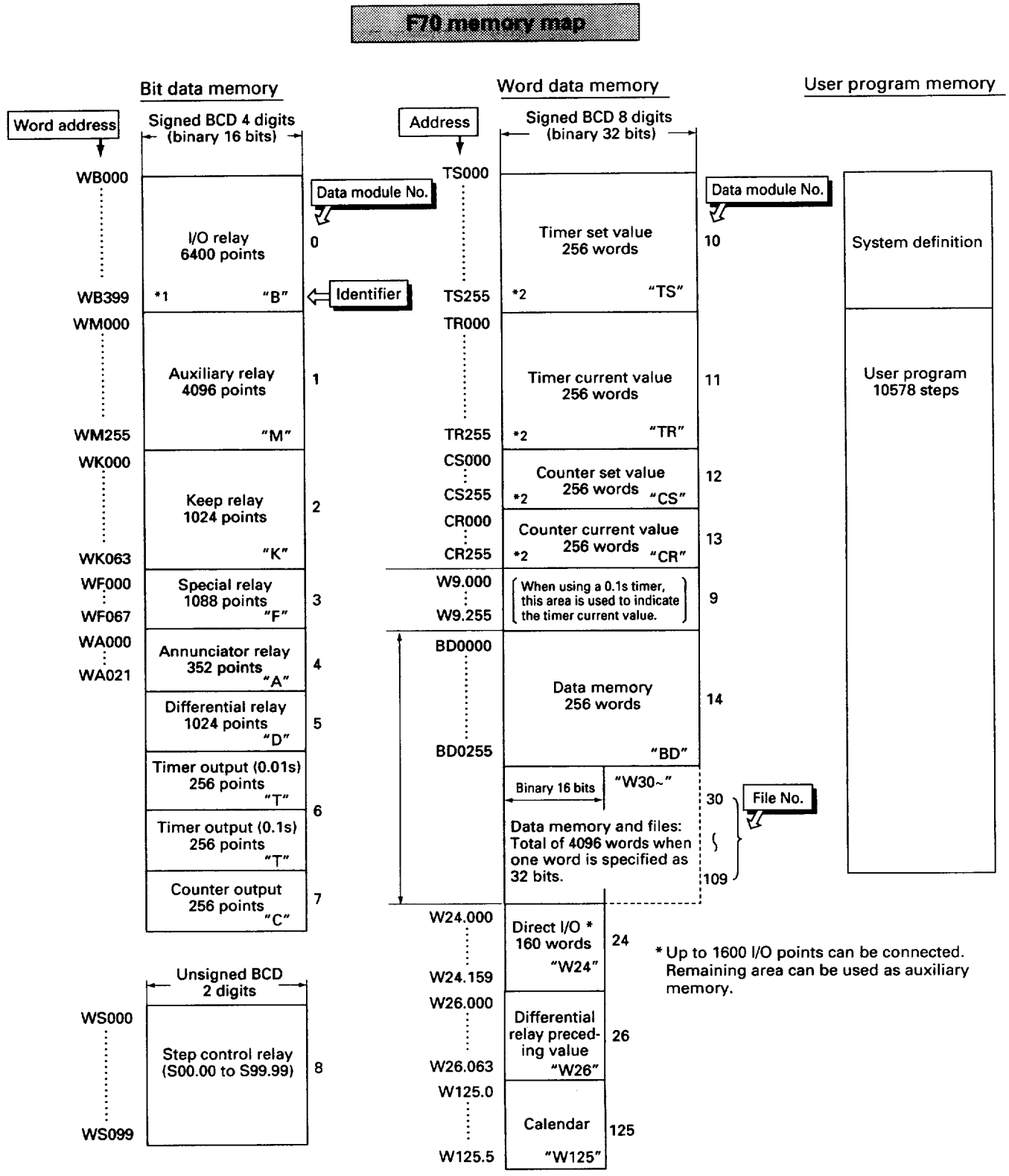
* Up to 1600 I/O points can be connected. Remaining area can be used as auxiliary memory.

*1 The I/O relay without I/O assigned can be used as an auxiliary relay. (However, the empty input area of each unit is not available.)

*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

*3 P-link memory is mounted on the P-link card. It is not available when the P-link card is not mounted.

8. F70 Series memory map

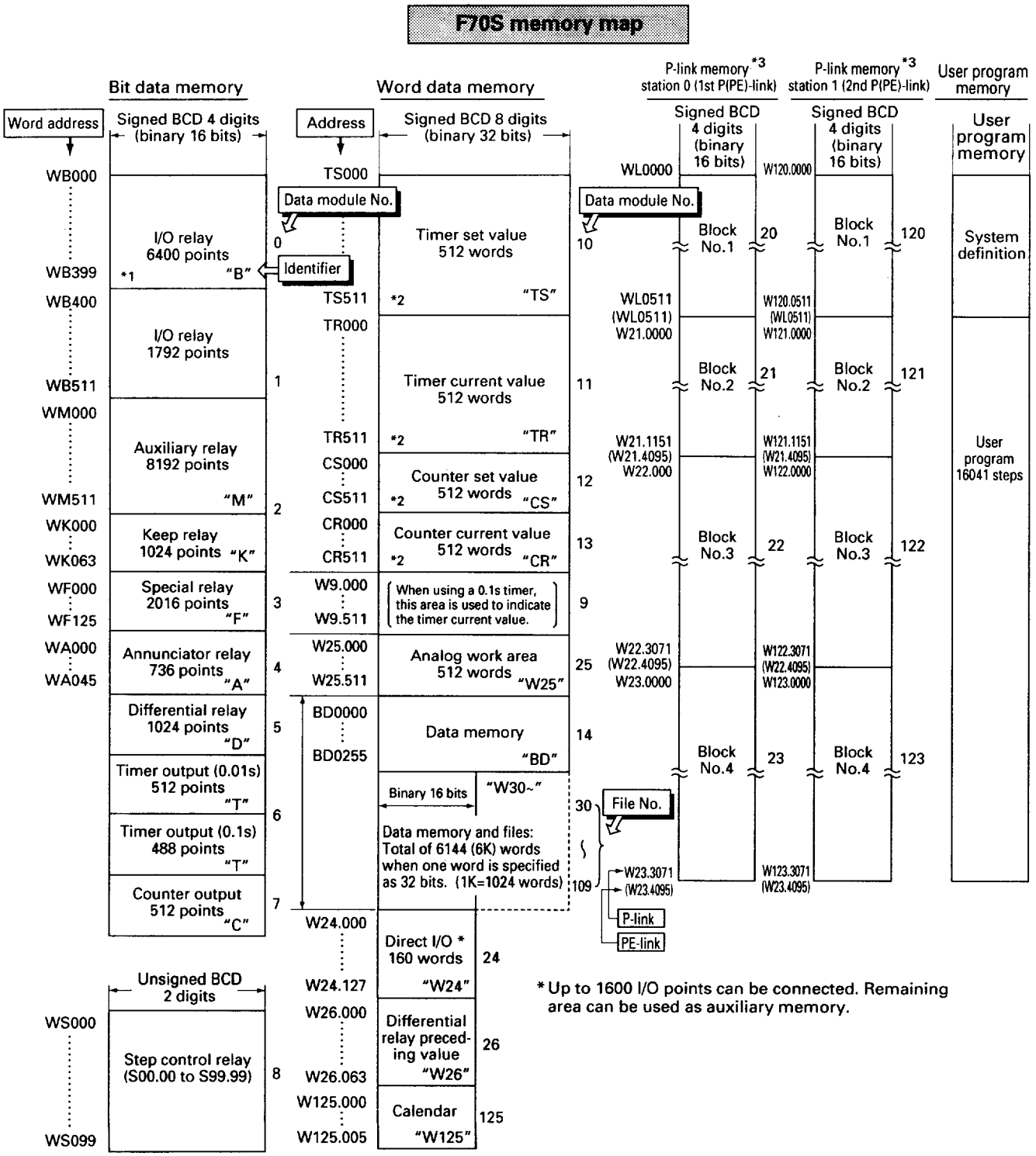


*1 The I/O relay without I/O assigned can be used as an auxiliary relay. (However, the empty input area of each unit is not available.)

*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

Section 2 Specifications

9. F70S Series memory map



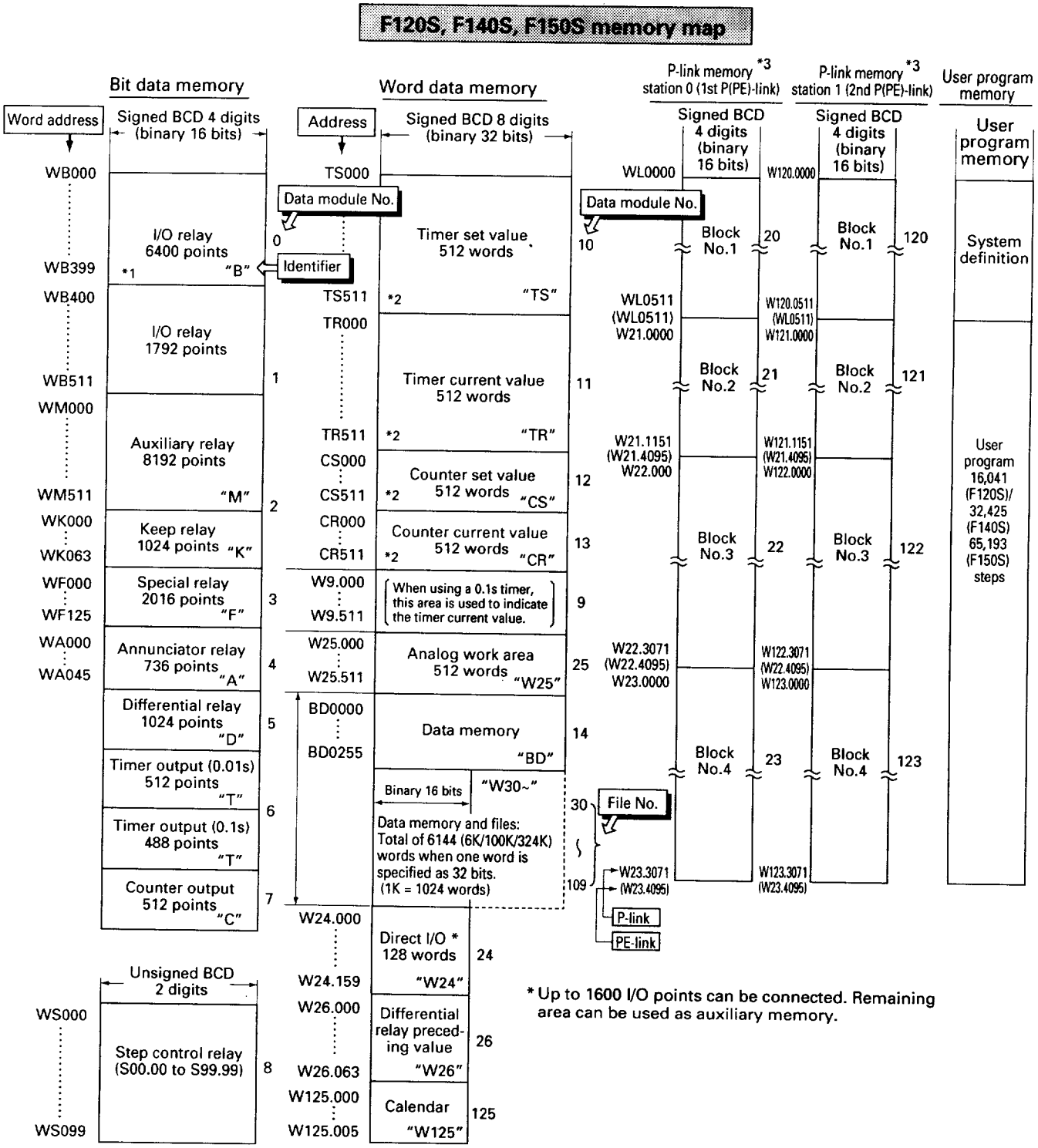
* Up to 1600 I/O points can be connected. Remaining area can be used as auxiliary memory.

*1 The I/O relay without I/O assigned can be used as an auxiliary relay. The area from B4000 to B511F can be used only as an auxiliary relay.

*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

*3 P-link memory is mounted on the P-link card. It is not available when the P-link card is not mounted.

10. F120S, F140S, F150S Series memory map



* Up to 1600 I/O points can be connected. Remaining area can be used as auxiliary memory.

*1 The I/O relay without I/O assigned can be used as an auxiliary relay. The area from B4000 to B511F can be used only as an auxiliary relay.

*2 Word data memory (TS, TR, CS, CR) not used in the program can be used as a signed BCD 8-digit data memory area.

*3 P-/PE-link memory is mounted on the P-/PE-link card. It is not available when the P-/PE-link card is not mounted.

2-2-2 Initial memory data

This section explains the initial data (values set when a user program is started at POWER-ON or by key switch or program loader operation) of memory used by the user program instructions.

1. Bit data memory

Identifier	Name		Initial data	Remarks
B	I/O relay		Latest data	
M	Auxiliary relay		OFF	
K	Keep relay		Preceding state retention	
D	Differential relay		OFF	
F	Special relay		Latest data	
A	Annunciator relay		OFF	
			Preceding state retention	When a program is started by program loader operation
L	Link relay (P-link)		Latest data	
S	Step control relay		Preceding state retention	
T	Timer	0.01s	OFF	
			Preceding state retention	Integrating timer only
		0.1s	OFF	Current value is stored in W9.
C	Counter		Preceding state retention	

2. Word data memory

Identifier	Name	Initial data	Remarks
BD	Data memory	Preceding state retention	
W30~	File memory	Preceding state retention	
WL (W120)	P-/PE-link memory	Latest data	
W21 (W121)			
W22 (W122)			
W23 (W123)			
W24	Direct access area	Latest data	
W25	Analog work area	Preceding state retention	
W26	Differential relay preceding value	Preceding state retention	
W125	Calendar	Latest data	
TS	Timer set value (0.01s)	Preceding state retention	
CS	Counter set value	Preceding state retention	
TR	Timer current value (0.01s)	OFF	When used as a timer instruction
		Preceding state retention	When used as data memory and integrating timer
CR	Counter current value	Preceding state retention	
W9	0.1s timer current value	OFF	When used as timer
		Preceding state retention	When used as data memory

Contents of initial data expressions:

Latest data

This value depends on the memory state at the beginning of the first scan operation.

(Example)

I/O relay: ON/OFF state of the input signal is the latest data.

OFF

This indicates that the memory area is initialized and set to OFF when the processor power supply is turned ON. At the beginning of the first scan operation, the user program is started with all corresponding memory areas set OFF.

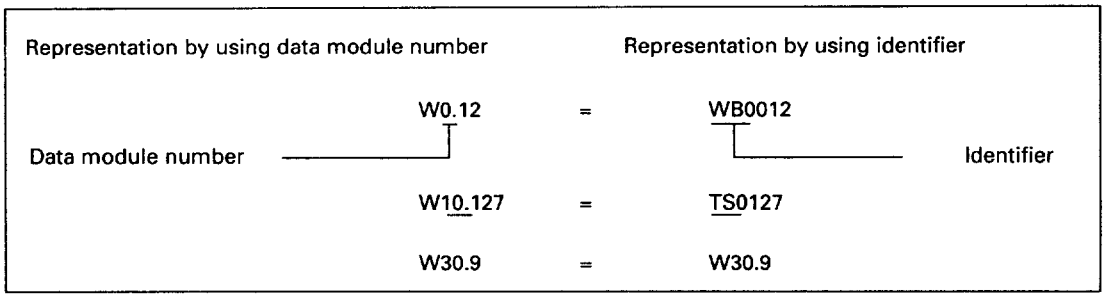
Preceding state retention

This indicates that the state is retained before the processor stopped. At the beginning of the first scan operation, the user program starts with the same ON/OFF state as set before it stopped.

2-2-3 Data module number

For the memory areas of MICREX-F PC, the file numbers (called data module numbers) listed in the following table are assigned to each area. A data-module No. is used for deletion of all data in a module, for indirect addressing by using selector (SEL) or deselector (DSEL) instructions, or for communication with a personal computer.
Word addresses of a data-module correspond to data addresses of each area (indicated by an identifier) as shown in the following example.

(Example)



Data module types

Data memory		Data module No. (file No.)
Component	Identifier	
I/O relay	B, WB	0
Auxiliary relay	M, WM	1
Keep relay	K, WK	2
Special relay	F, WF	3
Annunciator relay	A, WA	4
Differential relay	D	5
Step control relay	S, WS	8
Current value of 0.1s timer	W9.	9
Timer set value	TS	10
Timer current value	TR	11
Counter set value	CS	12

Data memory		Data module No. (file No.)
Component	Identifier	
Counter current value	CR	13
Data memory	BD	14
Direct access area	W24.	24
Differential relay preceding value	W26.	26
User file	Wxx.	30 to 109
P-link block No. 1	L, WL (W120.)	20 (120)
P-link block No. 2	W21. (W121.)	21 (121)
P-link block No. 3	W22. (W122.)	22 (122)
P-link block No. 4	W23. (W123.)	23 (123)
User indication	W124.	124 *
Calendar	W125.	125
Analog work area	W25.	25

*: F55 series only

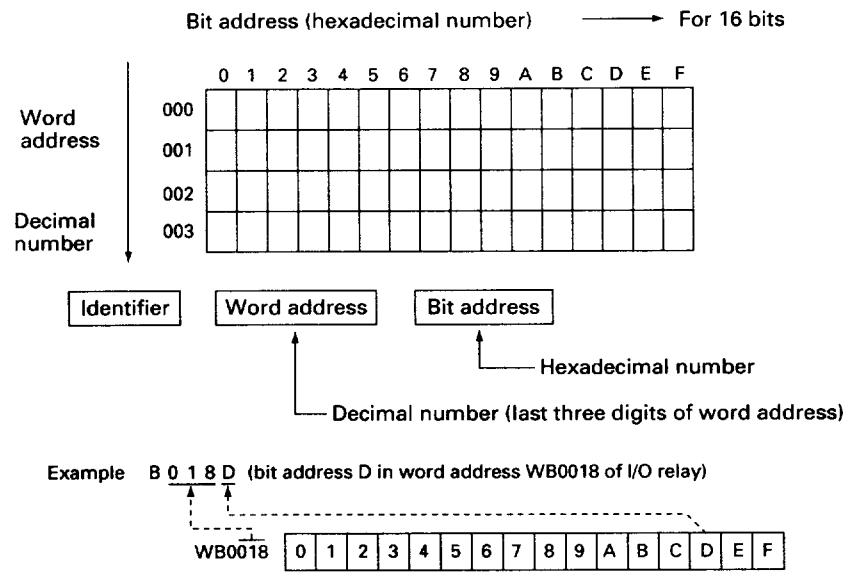
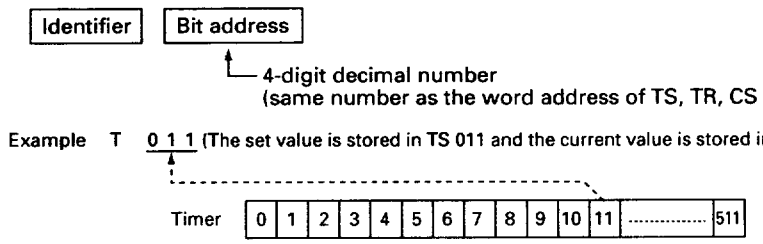
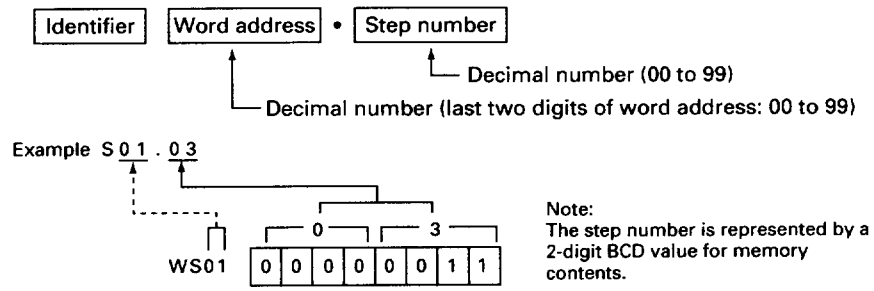
Section 2 Specifications

2-2-4 Address representation

The relays and word memory to be used by user programs must be assigned addresses. An address is represented by an identifier, word address and bit address.

The identifier is the first letter of the corresponding memory name. For example, the identifier of the keep relay is K.

1. Bit data memory

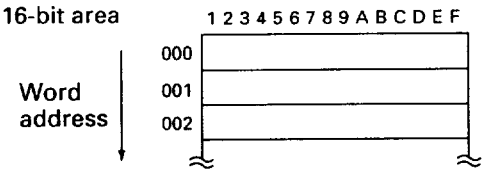
Identifier	Representation	Example of entry in loaer
B M K F A D L	A relay address is represented by an identifier, a word address and a bit address.  	B18D ("0" in the high-order digit can be omitted.)
T C	A timer or counter address is represented by an identifier and a bit address.  	T11 ("0" in the high-order digit can be omitted.)
S	The step control relay address is represented by an identifier, a word address and a step number  Note: The step number is represented by a 2-digit BCD value for memory contents.	S1.3 A "." is placed between the word address and the step number.

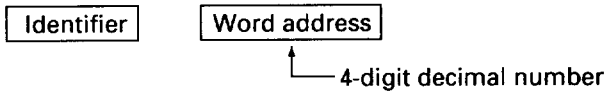
Key points

1. The relay number representation varies depending on the identifier.
2. When an address is entered in the loader, the leading "0" can be omitted.

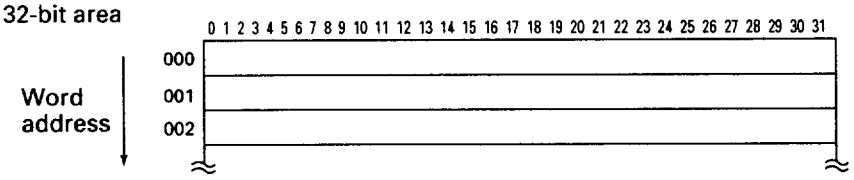
2. Word data memory

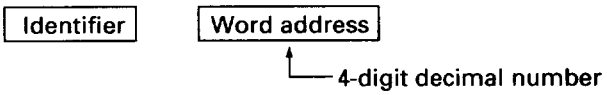
<table>
<tr><th>Identifier</th><th>Representation</th><th>Example of entry in loaer</th></tr>
<tr><td>WB
WM
WK
WF
WA
WS
WL
TS
TR
CS
CR
BD</td><td>The word data memory address is represented by an identifier and a word address.

• 16-bit area
 

Example WB002

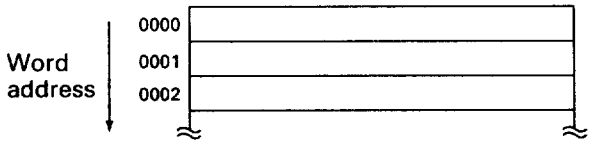
• 32-bit area
 

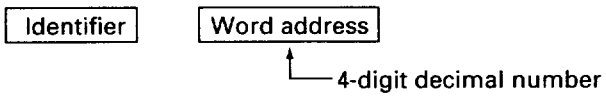
 

Example BD001</td><td>WB2

("0" in the high-order digit can be omitted.)

BD1</td></tr>
<tr><td>W9.
W21.
W22.
W23.
W24.
W25.
W26.
W30.
to
W109.
W120.
W121.
W122.
W123.
W124.
W125.</td><td>A 16-bit or 32-bit area address is represented by an identifier and a word address.

Example W21.0010
W26.0020
W30.0001
Identifier Word address</td><td>W21.10
W26.20
W30.1</td></tr>
</table>

Key points

1. Each word address is represented by a decimal number.
2. To represent a word data memory address of W9, W21 or more, the identifier and word address must be delimited by a period (.).
3. When an address is entered in the loader, the leading "0" can be omitted.

2-2-5 Index designation of memory

In F120H, F70S, F120S, F140S and F150S series, the memory can be designated index to reduce the number of the program steps and increase efficiency. The relay numbers and area designations are as follows.

1. Bit designation

Name and identifier		Relay No.	Area designations							Remarks
			B,M,K, D,F,A,L	S	T,C	i,j,k	ℓ ,m	P	Q	
i register	i	i0000 to i511F	○	—	—	○	—	○	—	• Data cannot be monitored. (Values can be monitored only at the end of scanning using the LITE) • " ℓ " and "m" can be used only for contact.
j register	j	j0000 to j511F	○	—	—	○	—	○	—	
k register	k	k0000 to k511F	○	—	—	○	—	○	—	
ℓ register	ℓ	ℓ 0000 to ℓ 0999	—	—	○	—	○	○	—	
m register	m	m0000 to m0999	—	—	○	—	○	○	—	

2. Word designation

Name and identifier		Relay No.	Area designations	Remarks
i register	Wi	Wi0000 to Wi4095	WB, WM, WK, WF, WA, WS, W9, CS, CR, TS, TR, BD, W30 to W109, WL, W21, W22 to W26, W120 to W123, W125, WP, Wi, Wj, Wk,	Data cannot be monitored. (Values can be monitored only at the end of scanning using the LITE.)
j register	Wj	Wj0000 to Wj4095		
k register	Wk	Wk0000 to Wk4095		

2-2-6 Parameter designations of memory (only for a function module FM)

In F120H, F70S, F120S, F140S and F150S series, function modules (subroutines) can be used to reduce the number of program steps and increase efficiency. The parameter relay numbers and area designations are as follows.

1. Bit designation

Name and identifier		Relay No.	Area designations							Remarks
			B,M,K, D,F,A,L	S	T,C	i,j,k	ℓ ,m	P	Q	
Parameter	P	P0000 to P0031	○	—	○	○	○	○	○	Memory cannot be used other than parameter designation. Data cannot be monitored. (Data can be monitored during step-by-step execution by the LITE)
Work area	Q	Q0000 to Q001F	—							

2. Word designation

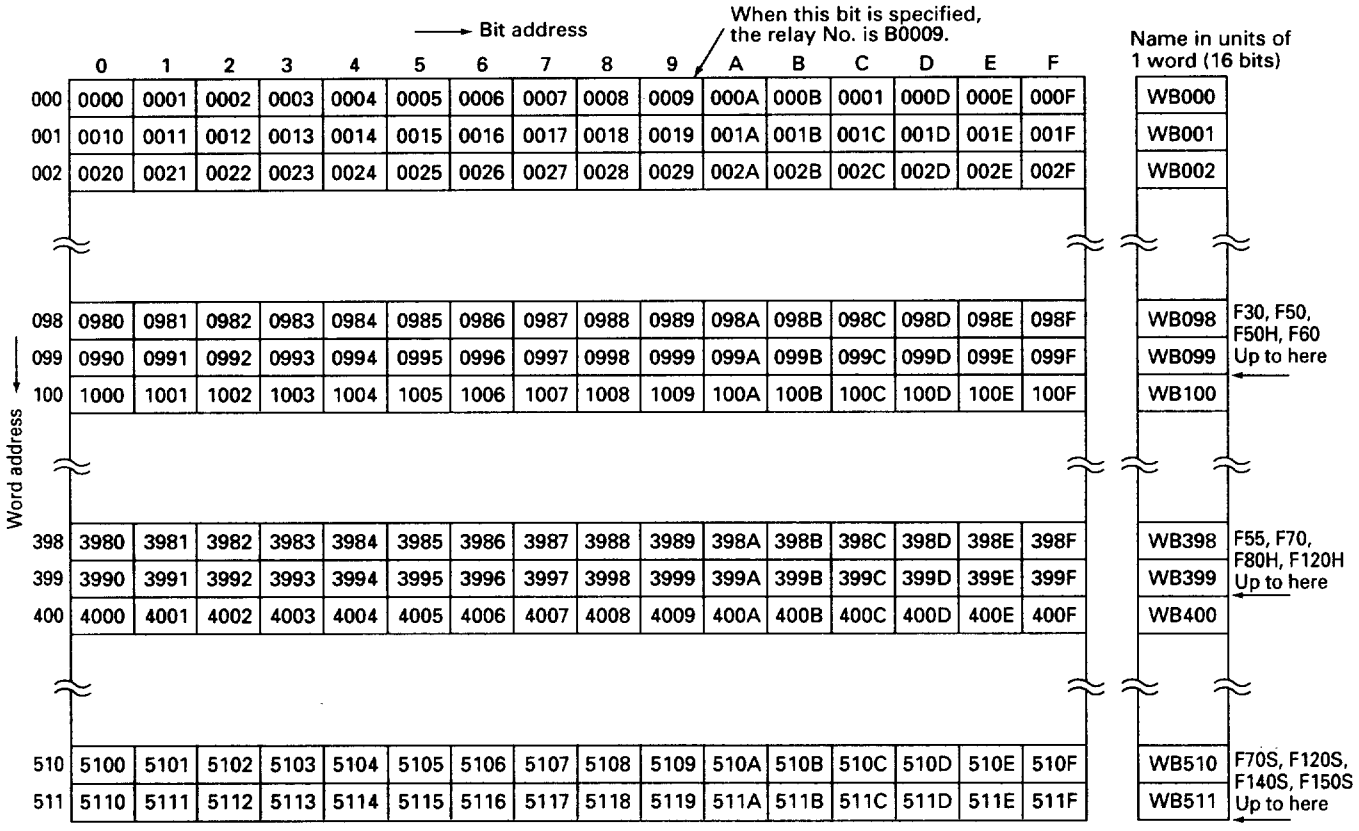
Name and identifier		Relay No.	Area designations	Remarks
Parameter	WP	WP0000 to WP0031	WB, WM, WK, WF, WA, WS, BD, TS, TR, CS, CR, W9, WL, W21 to W23, W26, W30 to W109, W120 to W123, W125, WP, WQ, Wi, Wj, Wk	Memory cannot be used other than parameter designation. Data cannot be monitored. (Data can be monitored during step-by-step execution by the LITE)
Work area	WQ	WQ0000 to WQ0031	—	

2-2-7 I/O relay areas (identifier: B or WB, data module No.: 0)

I/O relay areas are used for communication between the PC and external devices. These memory areas are used for inputting commands or data to the PC from pushbuttons, changeover switches, sensors and digital switches or for outputting results of program control to relays, solenoids and indicators.

Key points

1. The user can arbitrarily specify each memory as an input relay or output relay regardless of identifier.
2. Addresses are automatically determined according to the mounting order of I/O modules.
   The addresses of decentralized capsules connected to a T-link are set by using station number setting switches.
3. The number of NO and NC contacts is not restricted.
4. The I/O relay areas can be handled in units of bits (with one bit corresponding to a coil or contact) or in units of words (with one word consisting of 16 bits).
5. An area not assigned to an I/O unit can be used as an internal auxiliary relay area.
6. Memory range of each MICREX-F series
   - F30, F50, F50H, F60 series B0000 to B099F
   - F55, F70, F80H, F120H series B0000 to B399F
     (In F55, F70 and F80H series, B1000 to B399F can be used as an auxiliary relay only.)
   - F70S, F120S, F140S, F150S series B0000 to B511F
     (B4000 to B511F can be used as an auxiliary relay.)



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
000	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	000B	0001	000D	000E	000F	WB000
001	0010	0011	0012	0013	0014	0015	0016	0017	0018	0019	001A	001B	001C	001D	001E	001F	WB001
002	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	002A	002B	002C	002D	002E	002F	WB002
098	0980	0981	0982	0983	0984	0985	0986	0987	0988	0989	098A	098B	098C	098D	098E	098F	WB098
099	0990	0991	0992	0993	0994	0995	0996	0997	0998	0999	099A	099B	099C	099D	099E	099F	WB099
100	1000	1001	1002	1003	1004	1005	1006	1007	1008	1009	100A	100B	100C	100D	100E	100F	WB100
398	3980	3981	3982	3983	3984	3985	3986	3987	3988	3989	398A	398B	398C	398D	398E	398F	WB398
399	3990	3991	3992	3993	3994	3995	3996	3997	3998	3999	399A	399B	399C	399D	399E	399F	WB399
400	4000	4001	4002	4003	4004	4005	4006	4007	4008	4009	400A	400B	400C	400D	400E	400F	WB400
510	5100	5101	5102	5103	5104	5105	5106	5107	5108	5109	510A	510B	510C	510D	510E	510F	WB510
511	5110	5111	5112	5113	5114	5115	5116	5117	5118	5119	511A	511B	511C	511D	511E	511F	WB511

Section 2 Specifications

2-2-8 Auxiliary relay areas (identifier: M or WM, data module NO.: 1)

Auxiliary relay areas are used as internal relays of a PC and cannot be used for external output.

Key points

1. When the power supply is turned OFF or the processor stops running, the contents of auxiliary relay areas are cleared.
2. The number of NO and NC contacts used in a program is not restricted.
3. Auxiliary relay areas can be handled in units of bits or words.
4. Memory range of each MICREX-F series
   - F30, F50, F50H series M0000 to M031F
   - F60 series M0000 to M127F
   - F55, F70, F80H, F120H series M0000 to M255F
   - F70S, F120S, F140S, F150S series M0000 to M511F

Bit address → (0 1 2 3 4 5 6 7 8 9 A B C D E F)

Word address ↓

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		Name in units of 1 word (16 bits)	
000	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	000B	000C	000D	000E	000F		WM000	
001	0010	0011	0012	0013	0014	0015	0016	0017	0018	0019	001A	001B	001C	001D	001E	001F		WM001	
002	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	002A	002B	002C	002D	002E	002F		WM002	
030	0300	0301	0302	0303	0304	0305	0306	0307	0308	0309	030A	030B	030C	030D	030E	030F		WM030	F30, F50, F50H
031	0310	0311	0312	0313	0314	0315	0316	0317	0318	0319	031A	031B	031C	031D	031E	031F	*1	WM031	Up to here ←
032	0320	0321	0322	0323	0324	0325	0326	0327	0328	0329	032A	032B	032C	032D	032E	032F		WM032	
126	1260	1261	1262	1263	1264	1265	1266	1267	1268	1269	126A	126B	126C	126D	126E	126F		WM126	F60
127	1270	1271	1272	1273	1274	1275	1276	1277	1278	1279	127A	127B	127C	127D	127E	127F	*2	WM127	Up to here ←
128	1280	1281	1282	1283	1284	1285	1286	1287	1288	1289	128A	128B	128C	128D	128E	128F		WM128	
254	2540	2541	2542	2543	2544	2545	2546	2547	2548	2549	254A	254B	254C	254D	254E	254F		WM254	F55, F70, F80H, F120H
255	2550	2551	2552	2553	2554	2555	2556	2557	2558	2559	255A	255B	255C	255D	255E	255F		WM255	Up to here ←
	2560	2561	2562	2563	2564	2565	2566	2567	2568	2569	256A	256B	256C	256D	256E	256F		WM256	
510	5100	5101	5102	5103	5104	5105	5106	5107	5108	5109	510A	510B	510C	510D	510E	510F		WM510	F70S, F120S, F140S, F150S
511	5110	5111	5112	5113	5114	5115	5116	5117	5118	5119	511A	511B	511C	511D	511E	511F		WM511	Up to here ←

Notes: *1 In F30 and F50H series, M31F is a pulse input detection relay of input terminal B0001.
*2 In F60 series, M127F is a pulse input detection relay of input terminal B0001.

Note: When the ME-NET is used, the following area is used as link relays. The contents of the ME-NET link relays are retained even after the power supply is turned off (Previous value retention).
WM256 to 383: ME-NET No. 2
WM384 to 511: ME-NET No. 1

2-2-9 Keep relay areas (identifier: K or WK, data module No.: 2)

Keep relay areas are used as internal relays of a PC and cannot be used for external output. The contents of these memory areas are retained even after the power supply is turned OFF. (nonvolatile)

Key points

1. The contents of keep relay areas are retained (not cleared) even after the power supply is turned OFF or the processor stops running.
2. The number of NO and NC contacts of each relay in a program is not restricted.
3. Keep relay areas can be handled in units of bits or words.
4. Memory range of each MICREX-F series
   - F30, F50, F50H series ..... K0000 to K031F
   - F60, F55, F70, F80H, F120H, F70S, F120S, F140S, F150S series ..... K0000 to K063F

→ Bit address

Word address

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		Name in units of 1 word (16 bits)	
000	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	000B	000C	000D	000E	000F		WK000	
001	0010	0011	0012	0013	0014	0015	0016	0017	0018	0019	001A	001B	001C	001D	001E	001F		WK001	
002	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	002A	002B	002C	002D	002E	002F		WK002	
≈																		≈	
029	0290	0291	0292	0293	0294	0295	0296	0297	0298	0299	029A	029B	029C	029D	029E	029F	*1	WK029	
030	0300	0301	0302	0303	0304	0305	0306	0307	0308	0309	030A	030B	030C	030D	030E	030F		WK030	F30, F50, F50H
031	0310	0311	0312	0313	0314	0315	0316	0317	0318	0319	031A	031B	031C	031D	031E	031F	*2	WK031	Up to here
032	0320	0321	0322	0323	0324	0325	0326	0327	0328	0329	032A	032B	032C	032D	032E	032F		WK032	
≈																		≈	
061	0610	0611	0612	0613	0614	0615	0616	0617	0618	0619	061A	061B	061C	061D	061E	061F	*1	WK061	F60, F55, F70 F80H, F120H, F70S, F120S, F140S, F150S
062	0620	0621	0622	0623	0624	0625	0626	0627	0628	0629	062A	062B	062C	062D	062E	062F	*2	WK062	
063	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639	063A	063B	063C	063D	063E	063F		WK063	Up to here

Notes) *1 In F30, F60 series, K029F (F30) or K061F (F60) is used as a relay for the BER (Backup error) lamp. If a power failure time exceeds the backup time, K029F (F30) or K061F (F60) is ON. By resetting this relay, the lamp BER goes off.

*2 In F30, F50H or F60 DC input versions, when a counter C031 (C127 for F60) is used as an up-and down counter in a program, WK031 and 032 (WK062 and 063 for F60) are used as current value copy areas of built-in high-speed counter.

2-2-10 Differential relay areas (identifier: D, data module No.: 5)

Differential relays are used to detect the rising and falling edges of an input signal and are set ON during only one scan operation from the beginning of program execution steps. Each relay is used as a differential relay for rising edge (—(↑)—) or a differential relay for falling edge (—(↓)—).

Key points

1. The differential relay for rising edge is set ON at the rising edge of an input signal when the preceding value, which is stored at the corresponding address in the differential relay preceding value area (W26), is set OFF.
2. The differential relay for falling edge is set ON at the falling edge of an input signal when the preceding value, which is stored at the corresponding address in the differential relay preceding value area (W26), is set ON.
3. The number of NO and NC contacts of each relay in a program is not restricted.
4. Memory range of each MICREX-F series
   - F30, F50, F50H, F60 series ..... D0000 to D031F
   - F55, F70, F80H, F120H, F70S, F120S, F140S, F150S series ..... D0000 to D063F

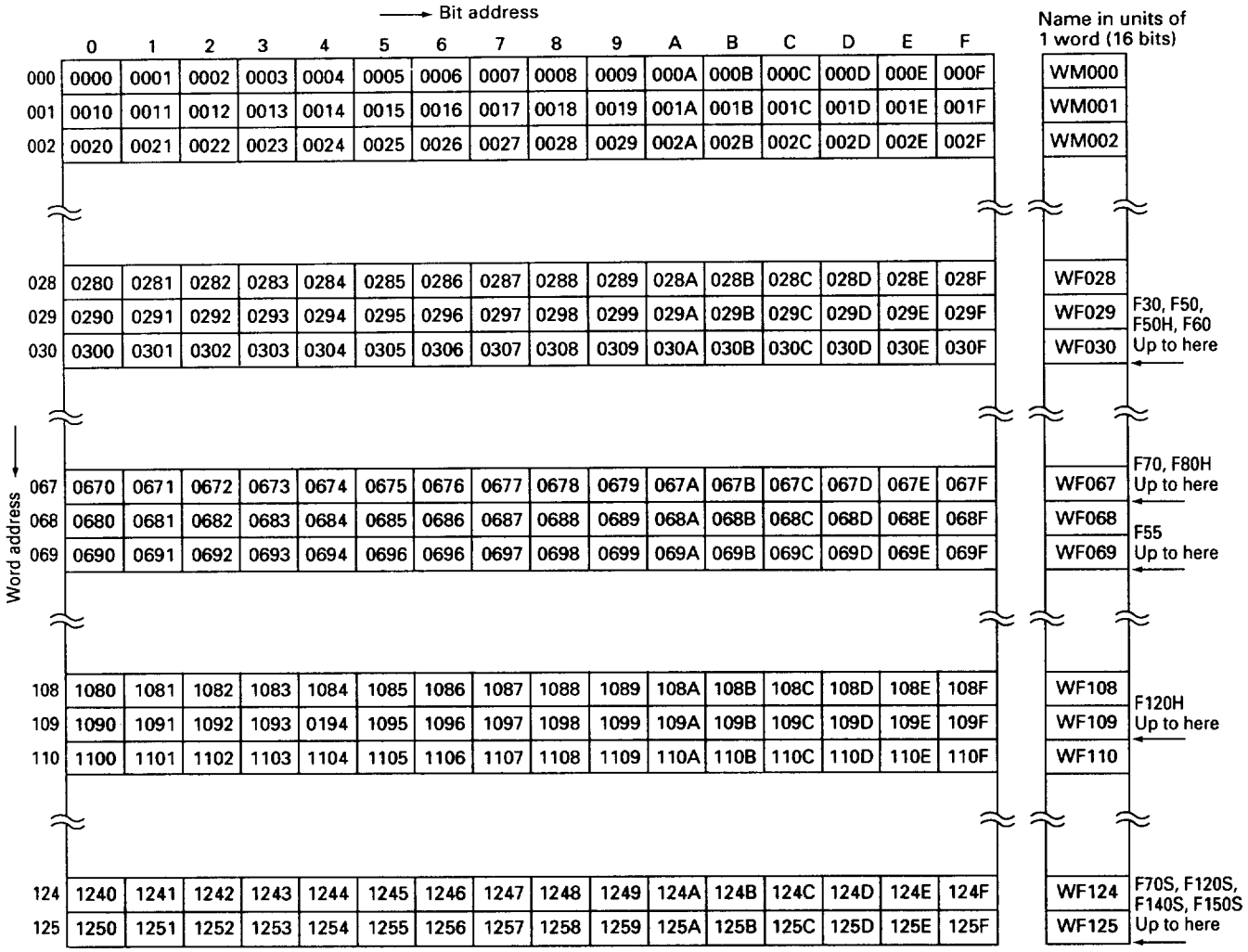
Bit address

Word address	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
000	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	000B	000C	000D	000E	000F	
001	0010	0011	0012	0013	0014	0015	0016	0017	0018	0019	001A	001B	001C	001D	001E	001F	
002	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	002A	002B	002C	002D	002E	002F	
≈																	
030	0300	0301	0302	0303	0304	0305	0306	0307	0308	0309	030A	030B	030C	030D	030E	030F	
031	0310	0311	0312	0313	0314	0315	0316	0317	0318	0319	031A	031B	031C	031D	031E	031F	F30, F50, F50H, F60 Up to here
032	0320	0321	0322	0323	0324	0325	0326	0327	0328	0329	032A	032B	032C	032D	032E	032F	
≈																	
062	0620	0621	0622	0623	0624	0625	0626	0627	0628	0629	062A	062B	062C	062D	062E	062F	
063	0630	0631	0632	0633	0634	0635	0636	0637	0638	0639	063A	063B	063C	063D	063E	063F	F55, F70, F80H, F120H, F70S, F120S F140S, F150S Up to here

2-2-11 Special relay areas (identifier: F or WF, data module NO.: 3)

Special relay areas are used for specific purposes, such as indicating the PC operation status and error status.

Special relay areas are read-only areas. (They cannot be used as coils, but only as contacts in a program).



The operation of each relay is as follows.

1. Operation status area (WF0000)

Relay No.	Name	Description	F30, F50 F50H	F60, F55 F70	F80H F120H	F70S, F120S F140S, F150S
F0000	Run	This relay is set ON during user program execution. When this relay is ON, the RUN lamp and RUN contact are ON.	○	○	○	○
F0001	Stop	This relay is set ON while the user program is stopped. When this relay is ON, the RUN lamp and RUN contact are OFF.	○	○	○	○
F0002	Fatal fault	When one of fault factor relays F0010 to F001F is ON, this relay is set ON. When this relay is ON, the ALM1 lamp and ALARM contact are ON.	○	○ *1)	○	○
F0003	Nonfatal fault	When one of nonfatal fault relays F0021 to F002F is ON, this relay is set ON (except for battery abnormalities). When this relay is ON, the ALM2 lamp and ALARM contact are ON.	○	○ *1)	○	○
F0004	Local nonfatal fault	When one of nonfatal fault factor relays F0022 to F0026, F0028 to F002C, F002E, and F002F is ON, this relay is set ON.	—	—	—	○
F0005	Other nonfatal fault	When one of nonfatal fault factor relays F0021, F0027, and F002D is ON, this relay is set ON.	—	—	—	○
F0007	Duplex master station number	If the duplex T-link master station number is 0, this relay is set OFF. If the duplex T-link master station number is 1, this relay is set ON.	—	—	○ *2)	○
F0008	Duplex system master	This relay is set ON when the processor is operating as a master processor in duplex system.	—	—	○ *2)	○
F0009	Duplex system slave	This relay is set ON when the processor is operating as a slave processor in duplex system.	—	—	○ *2)	○
F000A	Duplex mode	This relay is set ON when the duplex mode is set.	—	—	○ *2)	○
F000D	RUN/TEST	When this relay is ON, it indicates the RUN mode. When OFF, it indicates the TEST mode. (The indicated mode corresponds to the front-panel switch state).	—	—	○	○
F000E	STOP switch	In the STOP mode, this relay is set ON.	—	—	○	○
F000F	Mode selection lock	The relay is set ON when the RUN or STOP mode is set by the mode selection switch. While this relay is ON, the program loader cannot be used for control.	—	—	○	○

*1) ALARM contacts are not provided for F55 and F70 series.
*2) Supported by F120H series only.

○: Available —: Not available

2. Fatal fault factor area (WF0001)

Relay No.	Name	Description	F30, F50 F50H	F55, F60, F70, F80H	F120H, F70S, F120S, F140S, F150S
F0010	Memory error	This relay is set ON when a sum check error occurs. (It is reset by program creation, clearance or transfer.)	○	○	○
F0012	Auxiliary power supply error	This relay is set ON when an overload or short-circuit is detected in the auxiliary power supply (24V) of the processor module or basic unit.	—	○	○
F0013	Power supply fault	This relay is set ON when a processor power supply fault is detected. When power is supplied again, this relay is set OFF at transfer to the RAS copy area.	—	○	○
F0014	T-link fault	When the fail-soft operation is not specified and a T-link configuration fault occurs (F0026 is ON), this relay is set ON. This relay is set ON when a T-link LSI error occurs.	○	○	○
F0015	Option fault	This relay is set ON when a fault is detected in an option card or its setting.	—	— (○: F55)	○
F0016*	System stack error	This relay is set ON when a function module (subroutine) is called that exceeds the stack area.	—	—	○
F0017	PUSH/POP error	This relay is set ON when PUSH/POP stack overflow or underflow occurs. This relay is set ON when the PUSH operation count does not match the POP operation count.	—	—	○
F0018	User program error	This relay is set ON when the user program is incorrect. (It is reset when the user program error is corrected.)	○	○	○
F0019	WDT error	This relay is set ON when the user program execution time exceeds the watchdog timer time specified in the system definition.	○	○	○
F001A	Bus error	This relay is set ON when the fail-soft operation is not specified and a direct I/O error occurs (F0029 is ON).	○	○	○
F001B	Duplex system setting error	This relay is set ON when a processor station number is double-assigned in a duplex system.	—	—	○
F001C	I/O area double-assignment	This relay is set ON when an I/O area address is double-assigned.	○	○	○
F001D	Too many capsules on T-link	This relay is set ON when the number of I/O capsules (stations) connected to the T-link exceeds 32.	—	○	○
F001E	Too many I/O points	This relay is set ON when the number of words used exceeds the words of the I/O area.	—	○	○
F001F	Plant fault	This relay is set ON when one of relays A0000 to A000F is set ON.	○	○	○

Note*: This stack is a memory area used to temporarily save the return address from a function module (subroutine) or the address at which flag contents are to be stored.

○: Available —: Not available

Key points

1. If one of relays F0010 to F001F is set to ON, processor stops running, ALM1 lamp (ALM lamp for F30, F50, F50H) turns ON and the ALARM contact makes. F55 and F70 series have no ALARM contacts.
2. When one of the flags of fatal fault factors is set ON, the user program is not executed. Therefore, relays F0010 to F001F cannot be used by the user program.
3. Errors other than memory and user program errors can be recovered only by supplying power again. (They cannot be recovered by the start operation by the loader.)
4. The details of user program abnormality can easily be diagnosed by using the program loader LITE, D20 or program loader software.

3. Nonfatal fault factor area (WF002)

Relay No.	Name	Description	F30, F50 F50H	F60	F55, F70 F80H	F120H	F70S, F120S F140S, F150S
F0020	Battery error/ Backup error	This relay is set ON when the battery is not connected or its voltage is too low.	○	—	○	○	○
F0021	Duplex T-link disconnection	This relay is set ON when the backup processor is not connected in the duplex mode.	—	—	—	○	○
F0022	Option fault	This relay is set ON when an option card is faulty or an error is found in the system definition.	—	—	— (○: F55)	○	○
F0025	P-link (transmission) error	This relay set ON when a P-link transmission error occurs.	—	—	—	○	○
F0026	T-link configuration fault	This relay is set ON when a registered T-link station is not connected. This relay is set ON when a normally connected T-link station is disconnected. (It is set ON when one of flags in T-link faulty station area is ON.)	○	○	○	○	○
F0027	P-link configuration fault	This relay is set ON when a registered P-link station is not connected or is abnormal.	—	—	—	○	○
F0029	Direct I/O configuration fault	This relay is set ON when a registered I/O designation is faulty. This relay is set ON when an I/O unit that was operating normally is disconnected from the connector.	—	—	○	○	○
F002B	Block designation fault	This relay is set ON when a block that does not exist is started in program block selection.	—	—	—	—	○
F002E	Program slow-down	This relay is set ON when the waiting number of a fixed-cycle program (PROG50) exceeds 32. (The setting of this relay is changed during scanning.)	—	—	○	○	○
F002F	Plant fault	This relay is set ON when one of annunciator relays A0010 to A003F is set ON.	○	○	○	○	○

Note: All flags other than F002B or F002E are set ON or OFF after one scan operation is completed. ○: Available —: Not available

4. Group failure diagnosis area (WF0003)

This area is not used in F30, F50, F50H, F60, F55, F70, F80H, F120H, F70S, F120S, F140S or F150S series.

5. Operation result area (WF0004)

Relay No.	Name	Description	F30 F50, F50H	F60	F55, F70, F80H, F120, F70S, F120S, F140S, F150S
F0041	Level 1 interrupt mask	• This relay is set ON when the level 1 programs (PROG60 to PROG67) are masked by an interrupt disable instruction. It is set OFF when an interrupt enable instruction is executed.	—	—	○
F0042	Level 2 interrupt mask	• This relay is set ON when the level 2 program (PROG50) is masked by an interrupt disable instruction. It is set OFF when an interrupt enable instruction is executed.	—	—	○
F0046	File full	This relay is set ON when a file becomes full after an FFST instruction is executed. This relay is set OFF after an FIFO, FILO or FLCL instruction is executed.	—	○	○
F0047	File empty	This relay is set OFF after an FFST instruction executed. This relay is set ON when a file becomes empty after an FIFO or FILO instruction is executed. This relay is set ON after an FLCL instruction is executed.	—	○	○
F004E	Sign flag	This relay is set ON when the execution result of an arithmetic, logical operation or conversion instruction is negative. (It is set ON when the most significant bit of the stored data is 1.)	○	○	○
F004F	Zero flag	This relay is set ON when the execution result of an arithmetic, logical operation or conversion instruction is 0.	○	○	○

○: Available —: Not available

Key points

1. The contents of operation result flags F0046 to F004F are saved when an interrupt (PROG50 or PROG60 to PROG67) is generated and are restored when the interrupt is completed.

6. System clock area (WF0005)

Relay No.	Name	Description	F30, F50, F50H	F60, F55, F70, F80H, F120H	F70S, F120S, F140S, F150S
F0050	First scan	This relay is set ON only during execution of the first scan operation of the user program. (It is not set ON after test execution, conditional stop execution or step execution.)	○	○	○
F0053	0.1s clock	This relay is set ON for one scan operation in 0.1s intervals.	○	○	○
F0054	1s clock	This relay is set ON for one scan operation in 1s intervals.	○	○	○
F0056	1st PE-link	This relay is set ON when the 1st P-link is the PE-link.	—	—	○
F0057	2nd PE-link	This relay is set ON when the 2nd P-link is the PE-link.	—	—	○
F005D	Undefined expansion module No.	This relay is set ON when both input/output expansion is "specified" and an expansion module number is zero.	—	—	○
F005E	ROM operation	This relay is set ON when the ROM operation is executed.	—	— (○: F55/70)	○
F005F	Batteryless operation	This relay is set ON when a dummy connector for batteryless operation is mounted and the batteryless operation is executed.	—	— (○: F55/70)	○

Note: All WF0005 flags are set ON or OFF before the execution of one scan operation is started.

○: Available —: Not available

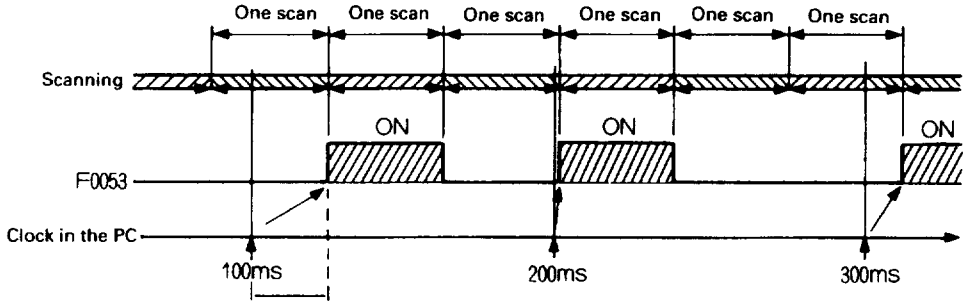
Key points

Notes on using the 0.1s clock (F0053) and 1s clock (F0054)

1. The following condition must be satisfied to use these clocks.

$$\text{Scan time} < \frac{\text{Clock time}}{2}$$

Note that because the rising edge of a clock signal is synchronized with scanning, the clock timing may be delayed by one scan operation period.

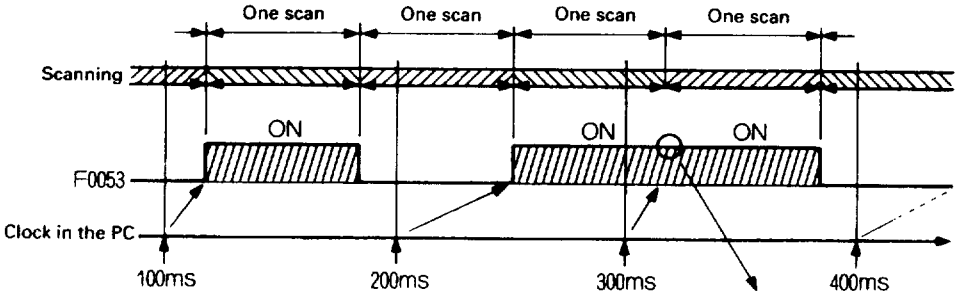


F0053 is set ON after a delay of up to one scan operation.
The minimum delay time is 0s. (F0053 is set ON at the same time as the clock in the PC.)

2. When the clocks are used under the following condition, note that an error (as shown in the figure below) may occur.

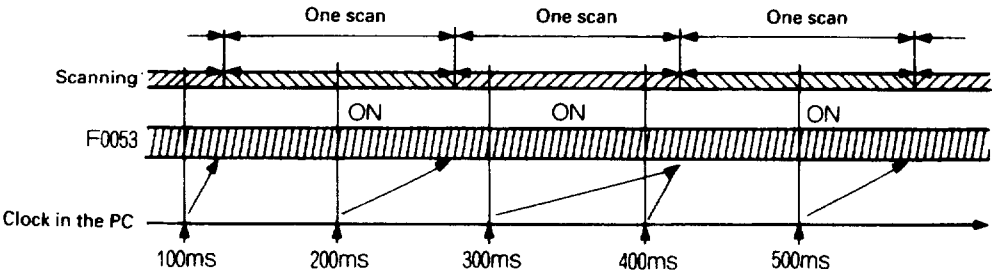
$$\frac{\text{Clock time}}{2} < \text{Scan time} < \text{Clock time}$$

Two ON signals of a clock may continue. If clocks are used as the count input signals for a counter, an error occurs because the two continuous clocks are counted as one pulse.



F0053 remains ON because the clock (300ms) in the PC is read before F0053 is set OFF.

3. When these clocks are used under the following condition, note that clock signals are continuously turned ON.



7. P-link configuration area (WF0006, WF0008) (This area is not available for F30, F50, F50H, F60, F55, F70 and F80H series.)

Relay No.	Name	Description	F70S (with P-link)	F120S, F140S, F150S (with P-link)
F0060 to F006F F0080 to F008F	1st P-link configuration (Station No. 0) to 1st P-link configuration (Station No. 15) 2nd P-link configuration (Station No. 0) to 2nd P-link configuration (Station No. 15)	These relays are set ON when data transmission via the P-link is normal and the P-link stations operate normally. (The configuration relay for a stopped P-link station is set OFF.)	○	○

8. P-link fault area (WF0007, WF0009) (This area is not available for F30, F50, F50H, F60, F55, F70 and F80H series.)

Relay No.	Name	Description	F70S (with P-link)	F120S, F140S, F150S (with P-link)
F0070 to F007F F0090 to F009F	1st P-link fault (Station No. 0) to 1st P-link fault (Station No. 15) 2nd P-link fault (Station No. 0) to 2nd P-link fault (Station No. 15)	These relays are set ON when data transmission via the P-link is abnormal. (The relay for a stopped P-link station is not set ON.)	○	○

9. T-link configuration area/T-link 0 (WF0010 to WF0019), T-link (WF0030 to WF0039), T-link 2 (WF0070 to 2F0079), and T-link 3 (WF0090 to WF0099)

Relay No.	Name	Description	F30, F50, F50H, F60	F55, F70, F80H	F120H, F70S, F120S, F140S, F150S
F0100 to F0199	T-link 0 configuration (Station No. 0) to T-link 0 configuration (Station No. 99)	These relays are set ON when data transmission via the T-link is normal and T-link stations operate normally. (F□□□A to F□□□F are not used) □□□: 010 to 019, 030 to 039, 070 to 079, 090 to 099	○	○	○
F0300 to F0399	T-link 1 configuration (Station No. 0) to T-link 1 configuration (Station No. 99)		—	—	○
F0700 to F0799	T-link 2 configuration (Station No. 0) to T-link 2 configuration (Station No. 99)		—	—	○
F0900 to F0999	T-link 3 configuration (Station No. 0) to T-link 3 configuration (Station No. 99)		—	—	○

○: Available —: Not available

10. T-link fault area/T-link 0 (WF0020 to WF0029), T-link 1 (WF0040 to WF0049), T-link 2 (WF0080 to WF0089), and T-link 3 (WF0100 to WF0199)

Relay No.	Name	Description	F30, F50, F50H, F60	F55, F70, F80H	F120H, F70S, F120S, F140S, F150S
F0200 to F0299	T-link 0 fault (Station No. 0) to T-link 0 fault (Station No. 99)	These relays are set ON when data transmission via the T-link is abnormal or a T-link station operates abnormally (faulty). (F□□□A to F□□□F are not used) □□□: 020 to 029, 040 to 049, 080 to 089, 100 to 109	○	○	○
F0400 to F0499	T-link 1 fault (Station No. 0) to T-link 1 fault (Station No. 99)		—	—	○
F0800 to F0899	T-link 3 fault (Station No. 0) to T-link 3 fault (Station No. 99)		—	—	○
F1000 to F1099	T-link 4 fault (Station No. 0) to T-link 4 fault (Station No. 99)		—	—	○

○: Available —: Not available

11. Direct I/O configuration area (WF0050) (This area is not available for F30, F50, F50H and F60.)

Relay No.	Name	Description
F0500	CPU rack configuration	This relay is ON when direct I/Os operate normally. (This relay is used only when direct I/Os operate in the direct access mode.)

12. Direct I/O fault status area (WF0060)

Relay No.	Name	Description
F0600	CPU rack error	Indicates the number of the slot of an abnormal I/O. (This relay is used only when direct I/Os operate in the direct access mode.)

13. Option configuration flag area (WF0068) (This area is not available for F30, F50, F50H, F60, F70, F80.)

Relay No.	Name	Description	F55	F120H, F70S, F120S, F140S, F150S	F152	F154
F0680	Slot 0 configuration	This relay indicates that the optional module in slot 0 is operating normally.	○	○	○	○
F0681	Slot 1 configuration	This relay indicates that the optional module in slot 1 is operating normally.	—	○	○	○
F0682	Slot 2 configuration	This relay indicates that the optional module in slot 2 is operating normally.	—	—	○	○
F0683	Slot 3 configuration	This relay indicates that the optional module in slot 3 is operating normally.	—	—	○	○
F0684	Slot 4 configuration	This relay indicates that the optional module in slot 4 is operating normally.	—	—	—	○
F0685	Slot 5 configuration	This relay indicates that the optional module in slot 5 is operating normally.	—	—	—	○

○: Available —: Not available

14. Option fault flag area (WF0069) (This area is not available for F30, F50, F50H, F60, F70, F80H.)

Relay No.	Name	Description	F55	F120H, F70S, F120S,F140S, F150S	F152S	F154S
F0690	Slot 0 error	This relay indicates that the optional module in slot 0 is abnormal.	○	○	○	○
F0691	Slot 1 error	This relay indicates that the optional module in slot 1 is abnormal.	—	○	○	○
F0692	Slot 2 error	This relay indicates that the optional module in slot 2 is abnormal.	—	—	○	○
F0693	Slot 3 error	This relay indicates that the optional module in slot 3 is abnormal.	—	—	○	○
F0694	Slot 4 error	This relay indicates that the optional module in slot 4 is abnormal.	—	—	—	○
F0695	Slot 5 error	This relay indicates that the optional module in slot 5 is abnormal.	—	—	—	○

○: Available —: Not available

15. PE-link configuration area (WF0110 to 0113, WF0118 to 0121)
(This area is not available for F30, F50, F50H, F60, F55, F70, F80H, F120H, F70S.)

Relay No.	Name	Description
F1100 to F113F	PE-link 1 configuration (Station No. 0) to PE-link 1 configuration (Station No. 3F)	These relays are set ON when data transmission via the PE-link is normal and the PE-link stations operate normally. (The configuration relay for a stopped PE-link station is set OFF.)
F1180 to F121F	PE-link 2 configuration (Station No. 0) to PE-link 2 configuration (Station No. 3F)	

16. PE-link fault area (WF0114 to 0117, WF0122 to 0125)
(This area is not available for F30, F50, F50H, F60, F55, F70, F80H, F120H, F70S.)

Relay No.	Name	Description
F1140 to F117F	PE-link 1 fault (Station No. 0) to PE-link 1 fault (Station No. 3F)	These relays are set ON when data transmission via the PE-link is abnormal. (The relay for a stopped PE-link station is not set ON.)
F1220 to F125F	PE-link 2 fault (Station No. 0) to PE-link 2 fault (Station No. 3F)	

2-2-12 Annunciator relay area (identifier: A or WA, data module No.: 4)

Annunciator relays are used in the user program for the following purposes.
These relays can be used as contacts or coils.

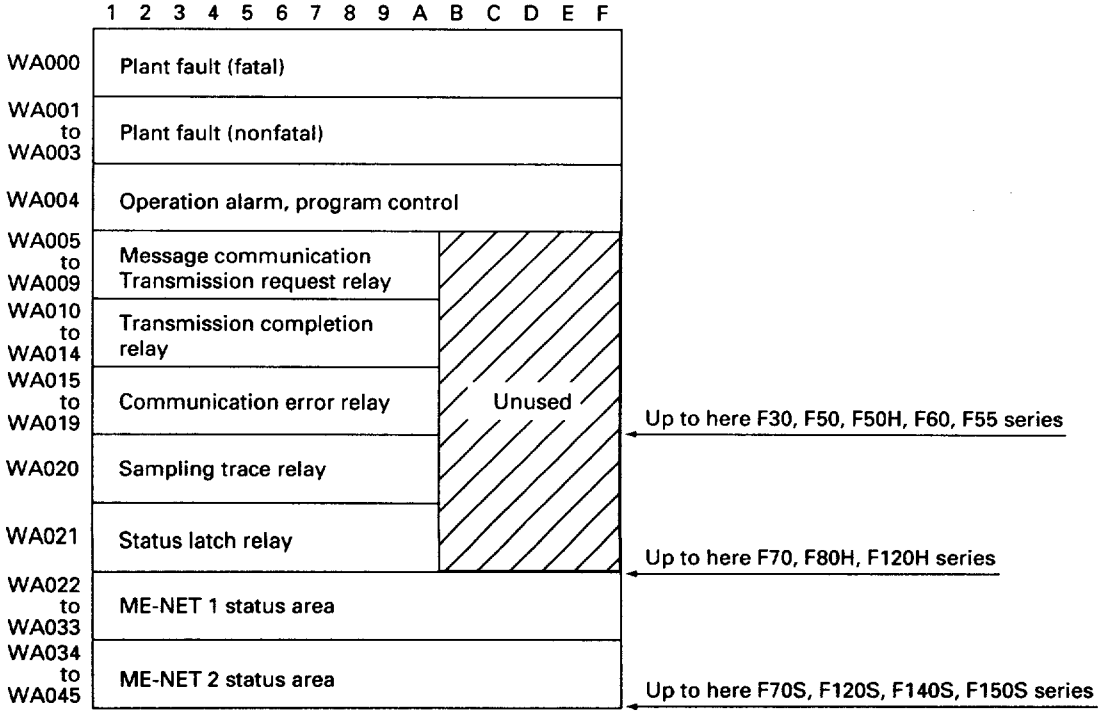
1. To stop the system when the machine or equipment malfunctions (A0000 to A000F)
2. To output an alarm while continuing system operation when the machine or equipment malfunctions (A0010 to A003F)
3. To take action according to the operation results (A0040, A0041)
4. To execute message communication (A0050 to A0199)
5. To execute a pause or stop operation (A004E, A004F)
6. To debug a user program (A0200 to A0202, A0208 to A0211, A0218)
7. To take an action according the operation status of other stations in ME-NET (A0220 to A045F)

Notes:
- The function of item 5 is only supported by F55, F70, F80H, F120H, F70S, F120S, F140S, F150 series only.
- The function of item 6 is only supported by F70, F80H, F120H, F70S, F120S, F140S, F150S series only.
- The function of item 7 is only supported by F120S, F140S, F150S.

Key points

1. The contents of these relays are cleared when the power is turned OFF. (They are not cleared by stop and start operations by the loader.)
2. If one of relays A0000 to A000F is set ON, special relay F001F is set ON. Then, relay F0002 is set ON and the processor is stopped. (Fatal fault)
3. If one of relays A0010 to A003F is set ON, special relay F002F is set ON. Then, relay F0003 is set ON and the ALARM2 (nonfatal fault) indicator of the processor lights up.
4. The numbers of NO contacts and NC contacts to be used for each relay in the program are not limited.
5. Memory range of each MICREX-F series
   - F30, F50, F50H, F60, F55 series A0000 to A019F
   - F70, F80H, F120H series A0000 to A021F
   - F70S, F120S, F140S, F150S series A0000 to A045F

Annunciator relay area functions



Note: F70 series has a memory up to WA045, but cannot conform to ME-NET.

Table 1. Processor status at plant fault occurrence
(F30, F50, F50H series)

Fault / Processor	Fatal fault (One of A0000 to A000F is ON.)	Nonfatal fault (One of A0010 to A003F is ON.)
Operation (RUN)	Stopped	Continued
Special relay (F)	F001F is ON.	F002F is ON.
External output *1		
RUN relay	OFF	ON
Fault relay	ON	ON
Indicator		
RUN	OFF	ON
ALM	ON	ON

*1 F30 series does not have an external output relay.

Table 2. Processor status at plant fault occurrence
(F60, F55, F70, F80H, F120H, F70S, F120S, F140S, F150S series)

Fault / Processor	Fatal fault (One of A0000 to A000F is ON.)	Nonfatal fault (One of A0010 to A003F is ON.)
Operation (RUN)	Stopped	Continued
Special relay (F)	F001F is ON.	F002F is ON.
External output		
RUN relay	OFF	ON
Fault relay	ON	ON *1
Indicator		
RUN	OFF	ON
ALARM1	ON	OFF
ALARM2	OFF	ON

*1 F55, F70, F70S do not have this relay.

Classification	Relay No.	Name	Description	F30, F50, F50H, F60	F55, F70, F80H, F120H F70S, F120S, F140S, F150S
Plant fault *1	A0000 to A000F	Plant fault (fatal)	When one of these relays is set ON, the current operation is stopped. (See Table 1 or 2 for the processor status.)	○	○
	A0010 to A003F	Plant fault (nonfatal)	When one of these relays is set ON, a fault signal is output. However, the current operation is continued. (See Table 1 or 2 for the processor status.)	○	○
Operation alarm	A0040	Overflow	This relay is set ON when an arithmetic operation result exceeds the storage memory area. (A result is stored as the maximum or minimum memory value.) Once an overflow occurs, the relay is not turned OFF until it is set OFF by POWER-OFF or a user program.	○	○
	A0041	Operation execution error	This relay is set ON when an operation is not executed normally. (The operation result is not stored in memory, but the previous data is retained.) Once an operation execution error occurs, this relay is not turned OFF until it is set OFF by POWER-OFF or a user program.	○	○
Program control	A004E	Pause	At the end of the scan operation during which this relay is set ON, the program is stopped but internal memory data is retained. Program execution is continued by the start operation from the loader.	—	○
	A004F	Stop	At the end of the scan operation during which this relay is set ON, the program is stopped and internal memory is initialized. Program execution is restarted from the first scan operation by the start operation from the loader.	—	○

*1 Each relay for plant fault can be used by the user program if necessary.

Classification	Relay No.	Name	Description	F30, F50, F50H, F60	F55	F70, F80H, F120H, F70S F120S, F140S, F150S
Message communication	A0050 to A0065	Transmission request	When one of these relays is set ON, the corresponding message-communication module starts sending messages.	○	○	○
	A0066 to A0099			—		
	A0100 to A0115	Transmission/ reception completion notification	One of these relays is set ON when transmission or reception by the corresponding message-communication module is completed normally. The relay is turned OFF by a user program.	○	○	○
	A0116 to A0149			—		
	A0150 to A0165	Communication error notification	One of these relays is set ON when an error occurs in the data transmitted by the corresponding message-communication module. The relay is turned OFF by a user program.	○	○	○
	A0166 to A0199			—		
Flags for debugging	A0200	Sampling trace enabled	Enables sampling trace.	—	—	○
	A0201	Sampling trace execution	Executes sampling trace.	—	—	○
	A0202	Coil trace	This relay is used when the trace method of sampling is coil trace. When this flag is ON while the sampling trace is being executed, data sampling is made.	—	—	○
	A0208	Status latch enabled	Enables status latch.	—	—	○
	A0209	Status latch execution	Executes status latch.	—	—	○
	A0210	Sampling trace under progress	Indicates that data sampling is under progress.	—	—	○
	A0211	Sampling trace completed	Indicates that data sampling is completed.	—	—	○
	A0218	Status latch completed	Indicates that the storing the status latch data into memory is completed.	—	—	○

○: Available —: Not available

• **First ME-NET status area (WA0022 to WA0033)** (This area is only supported by F120S, F140S, F150S series.)

Classification	Relay No.	Name	Description	Object processor		
				F120S	F140S	F150S
Communication monitor flag	A0220 to A022F	ME-NET station 15 to ME-NET station 0	When a station is connected to a link for data transfer, the communication monitor flag bit corresponding to the station is set ON. The bits corresponding to stations not connected to a link are set OFF.	○	○	○
	A0230 to A023F	ME-NET station 31 to ME-NET station 16				
	A0240 to A024F	ME-NET station 47 to ME-NET station 32				
	A0250 to A025F	ME-NET station 63 to ME-NET station 48				
Operating status flag 1	A0260 to A026F	ME-NET station 15 to ME-NET station 0	When a station connected to a link for data transfer is operating, the bit of the operating status flag 1 corresponding to the station is set ON. The bits corresponding to stopped stations are set OFF.	○	○	○
	A0270 to A027F	ME-NET station 31 to ME-NET station 16				
	A0280 to A028F	ME-NET station 47 to ME-NET station 32				
	A0290 to A029F	ME-NET station 63 to ME-NET station 48				
Operating status flag 2	A0300 to A030F	ME-NET station 15 to ME-NET station 0	When a station connected to a link for data transfer is operating normally, the bit of the operating status flag 2 corresponding to the station is set ON. The bits corresponding to abnormal stations are set OFF.	○	○	○
	A0310 to A031F	ME-NET station 31 to ME-NET station 16				
	A0320 to A032F	ME-NET station 47 to ME-NET station 32				
	A0330 to A033F	ME-NET station 63 to ME-NET station 48				

○: Available

• **Second ME-NET status area (WA0034 to WA0045)** (This area is only supported by F152S and F154S of F150S series.)

Classification	Relay No.	Name	Description	Object processor		
				F120S	F140S	F150S
Communication monitor flag	A0340 to A034F	ME-NET station 15 to ME-NET station 0	When a station is connected to a link for data transfer, the communication monitor flag bit corresponding to the station is set ON. The bits corresponding to stations not connected to a link are set OFF.	—	—	○
	A0350 to A035F	ME-NET station 31 to ME-NET station 16				
	A0360 to A036F	ME-NET station 47 to ME-NET station 32				
	A0370 to A037F	ME-NET station 63 to ME-NET station 48				
Operating status flag 1	A0380 to A038F	ME-NET station 15 to ME-NET station 0	When a station connected to a link for data transfer is operating, the bit of the operating status flag 1 corresponding to the station is set ON. The bits corresponding to stopped stations are set OFF.	—	—	○
	A0390 to A039F	ME-NET station 31 to ME-NET station 16				
	A0400 to A040F	ME-NET station 47 to ME-NET station 32				
	A0410 to A041F	ME-NET station 63 to ME-NET station 48				
Operating status flag 2	A0420 to A042F	ME-NET station 15 to ME-NET station 0	When a station connected to a link for data transfer is operating normally, the bit of the operating status flag 2 corresponding to the station is set ON. The bits corresponding to abnormal stations are set OFF.	—	—	○
	A0430 to A043F	ME-NET station 31 to ME-NET station 16				
	A0440 to A044F	ME-NET station 47 to ME-NET station 32				
	A0450 to A045F	ME-NET station 63 to ME-NET station 48				

○: Available —: Not available

1. Example of using pause annunciator relay (A004E)

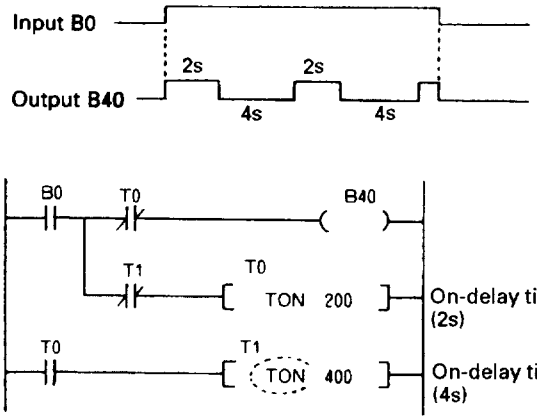
A pause annunciator relay temporarily stops program execution while retaining the output status. If the output status is to be cleared, a stop relay (A004F) must be used.

The pause annunciator relay function is used as follows.

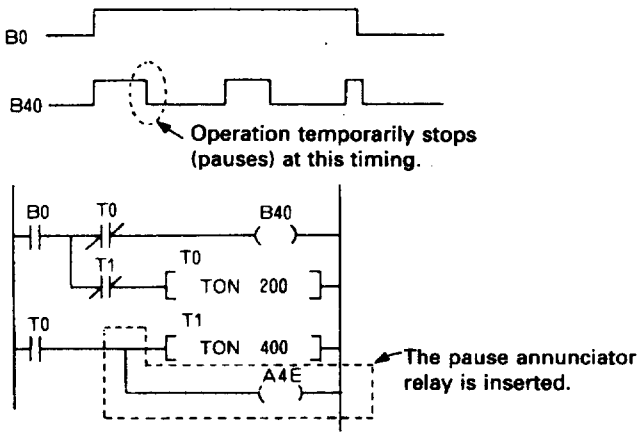
(1) Insert a pause annunciator relay in an optional circuit as required during debugging. (Multiple pause annunciator relays can be used.)
(2) Start the processor in the normal program execution mode.
(3) Set a test condition for debugging. When the condition is satisfied, the pause annunciator relay is set ON and the processor pauses (stops temporarily).
(4) Monitor the program or data by using the program loader. Program modification (including the removal of the pause annunciator relay) is possible.
(5) The program can be restarted by program loader operation START ENT and ENT).

Example

The sequence circuit and its operation are shown below.

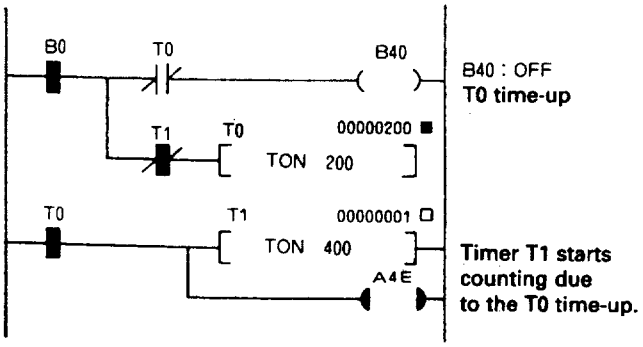


The operation of this sequence circuit is debugged. For example, this circuit operation is temporarily stopped when a timer T0 time-up occurs after output B40 has been set ON for two seconds.



Circuit monitoring by program loader LITE

- The processor stops temporarily when a timer T0 time-up occurs.



CAUTION:

When the pause annunciator relay is set ON, the processor always pauses after completing execution for the current scan. It does not pause at the circuit where the pause annunciator relay is inserted.

Restart operation

Program execution can be restarted while retaining the output status.

(Program loader LITE)

START ENT ENT

Note: • Program execution restarts at the beginning of the program.
• Program can be restarted by the program loader when the processor is in terminal (TERM) mode.

2-2-13 Timer areas (0.01-second timer) (identifier: T, identifier: TS, data module No.: 10, identifier: TR, data module No.: 11)

A 0.01-second timer consists of a timer output area (T), set value area (TS) and current value area (TR). One set of T, TS and TR having the same address functions as a timer. When the value of TR reaches the value of TS, the time-up bit (T-area) is set ON.
The TR and TS areas consist of BCD 8-digit data and up to 799,999.99 seconds (about 222 hours) can be set by timer.

Output bit address (T)	Set value address (TS)	Current value address (TR)
T000	TS000	TR000
T001	TS001	TR001
T002	TS002	TR002

Key points

1. Whether the data of T, TS and TR is to be retained when the power supply is turned OFF or the processor stops depends on the mode in which the timer is used. See the table in Item (2) of Section 2-2-2 for details.
2. If a timer area is double-used, a user program error is detected and the processor stops.
   Even when timers having different functions are used, a user program error is detected if the timer address is double-assigned.
3. The number of NO and NC contacts of a timer in a program is not restricted.
4. Data can be preset to the TR and TS areas.
5. Timer accuracy: +2 scan times to +1 scan time
6. Memory range of each MICREX-F series
   - F30, F50, F50H series ..... T000 to T127 (128 points)
   - F60, F55, F70, F80H, F120H series ..... T000 to T255 (256 points)
   - F70S, F120S, F140S, F150S series ..... T000 to T511 (512 points)

■ Timer output area (bit address)

000	001	002	003	004	005	006	007	008	009	010	011	012	013	014	015
016	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031
032	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047
≈															≈
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
≈															≈
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255
≈															≈
480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495
496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511

F30, F50, F50H Up to here

F55, F60, F70, F80H, 120H Up to here

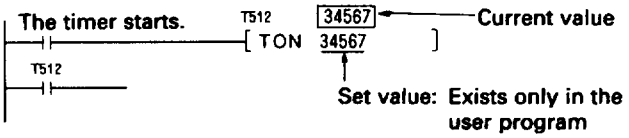
F70S, F120S, F140S, F150S Up to here

2-2-14 Timer areas (0.1-second timer) (identifier: T / identifier: W9, data module No.: 9)

A 0.1-second timer consists of a timer output area (T) and a current value area (W9). (F60 series uses a BD area as a current value area.) The timer set value exists only in the user program.
When the timer current value reaches the timer set value, the time-up bit (T area) is set ON.

Output bit address (T)	Current value address	
	(W9)	(BD) (F60 series)
T512	W00.90000	BD0000
T513	W00.90001	BD0001
		*1)

when the current value reaches 3456.7 seconds.



The timer set value consists of BCD 8-digit data and up to 7,999,999.9 seconds (about 2220 hours) can be set.

For example, T512 is set ON in the following program

Key points

1. The current value is cleared when the power supply is turned OFF or the processor stops.
2. If a timer area is double-used, a user program error is detected and the processor stops.
3. The number of NO and NC contacts of a timer in a program is not restricted.
4. Timer accuracy: +2 scan times to +1 scan time
5. Memory range of each MICREX-F series
   - This area is not available for F30, F50, F50H series, and 0.1s timer cannot be used.
   - F60, F55, F70, F80H, F120H series ..... T512 to T767 (256 points)
   - F70S, F120S, F140S, F150S series ..... T512 to T999 (488 points)

■ Timer output area (bit address)

512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527
528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543
544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559
≈															≈
752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767
≈															≈
976	977	978	979	980	981	982	983	984	985	986	987	988	989	990	991
992	993	994	995	996	997	998	999								

F60, F55, F70, F80H, F120H Up to here

F70S, F120S, F140S, F150S Up to here

*1) The area W9.488 to W9.511 can be only used to the data memory area.

2-2-15 Counter areas

identifier: C
identifier: CS, data module No.: 12
identifier: CR, data module No.: 13

A counter area consists of a counter output area (C), counter set value area (CS) and counter current value area (CR).
When the current value becomes negative, the count-up bit is also set ON.
The table on the right shows the relationships between the bit addresses of the counter output area (C) and the word addresses of the set value area (CS) and current value area (CR).
The maximum count value set by the counter is BCD 8-digit (79,999,999).

Output bit address (C)	Set value address (CS)	Current value address (CR)
C000	CS000	CR000
C001	CS001	CR002
C002	CS002	CR002

Key points

1. When the power supply is turned OFF or the processor stops, the preceding values of C, CR and CS are retained. However, if CS contains a set value, the value becomes the program-specified value.
2. If a counter area is double-written, a user program error is detected and the processor stops.
3. The number of NO and NC contacts of a counter output in a program is not restricted.
4. Data can be preset to the CR and CS areas.
5. The count-up bit of the up/down counter (UDCT) is set ON even when the current value becomes negative.
6. When C31 is used by the up/down counter instruction in F30, F50H series, or C127 is used as an up/down counter, they operate as a built-in high-speed counter. Those operation is different from that of normal up/down counter.
   For the detailed specifications and the usage, see the user's manual Hardware of each series.
7. Memory range of each MICREX-F series
   - F30, F50, F50H series
     ..... C000 to C031 (32 points)
   - F60 series
     ..... C000 to C127 (128 points)
   - F55, F70, F80H, F120H series
     ..... C000 to C255 (256 points)
   - F70S, F120S, F140S, F150S series
     ..... C000 to C511 (512 points.)

■ **Counter output area (bit address)**

000	001	002	003	004	005	006	007	008	009	010	011	012	013	014	015
016	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031
032	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047
≈															≈
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
≈															≈
239	240	241	242	243	244	245	246	247	248	249	251	252	253	254	255
≈															≈
480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495
496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511

F30, F50, F50H Up to here
F60 Up to here
F55, F70, F80H, F120H Up to here
F70S, F120S, F140S, F150S Up to here

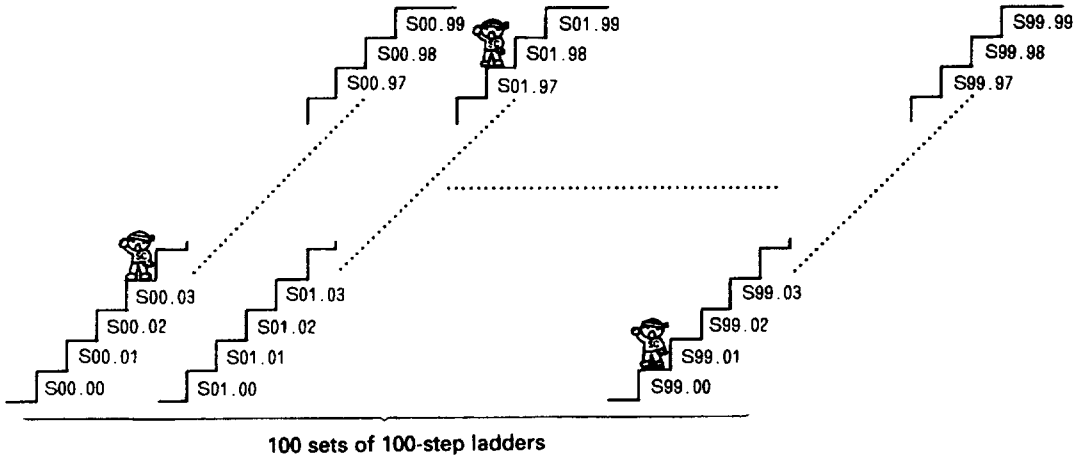
2-2-16 SC (step control) areas (identifier: S or WS, data module No.: 8)

This memory area is used to store step numbers for step control. This area contains up to 100 words of unsigned BCD 2-digit (8-bit) data that can be directly read and written by using sequence and word operation instructions. This area is nonvolatile.

Word address	Specified in units of bits	Name in units of 1 word (8 bits)
0	S00.00 to S00.99 (Total of 100 steps)	WS00
1	S01.00 to S01.99	WS01
2	S02.00 to S02.99	WS02
3	S03.00 to S03.99	WS03
4	S04.00 to S04.99	WS04
5	S05.00 to S05.99	WS05
6	S06.00 to S06.99	WS06
7	S07.00 to S07.99	WS07
8	S08.00 to S08.99	WS08
9	S09.00 to S09.99	WS09
10	S10.00 to S10.99	WS10
≈	≈	≈
96	S96.00 to S96.99	WS96
97	S97.00 to S97.99	WS97
98	S98.00 to S98.99	WS98
99	S99.00 to S99.99	WS99

Key points

1. The conceptual drawing of SC instruction and memory area is as follows.
   (1) There are a hundred sets (corresponding to word numbers 00 to 99) of 100-step ladders.
   (2) A set of 100-step ladders is used exclusively for one SC.



2. The step name format is as follows.

S00.03
- Step No. (00 to 99)
- Set No. (last two digits of word address)

3. Each step number corresponds to a BCD 2-digit value. It does not refer to any bit address as in case of such areas as B, M and K.
4. The user may regard this area as if there were 100 SC coils numbered from 00 to 99 in a single word.
5. The number of NO and NC contacts of each coil in a program is not restricted.

2-2-17 Data memory areas (identifier: BD, data module No.: 14)

The data memory area is used to store internal processor data, which can be read and written in units of words.

Key points

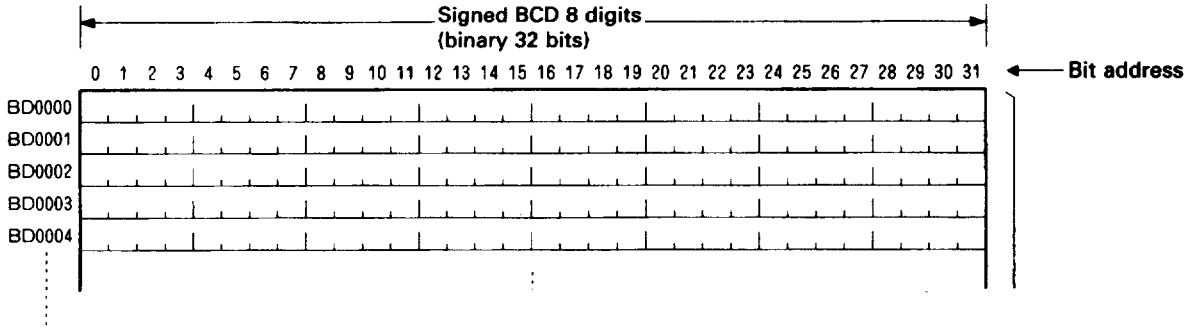
1. Even when the power supply is turned OFF or the processor stops, the preceding data are retained. (Nonvolatile area)
2. Storage data formats:
   Signed BCD 8-digit data (-79,999,999 to +79,999,999)
   Hexadecimal number (0 to FFFFFFFF)
3. If the BD memory capacity is insufficient, the following areas can be used as data memory.
   (1) Unused timer and counter set value areas (TS and CS) and current value areas (TR, CR and W9)
   (2) User file area (Data is read and written by SEL and DSEL instructions.)
   (3) Unused keep relay area (only for 16-bit data)
   (4) Unused auxiliary relay area (This area is used only for 16-bit data and volatile.)
4. In the F60 series, the BD area is used as a 0.1s timer current value area. Note that this area is not double-used when the 0.1 second timer is used.
5. In the F55, F70, F80H, F120H, F140S, F150S series, the data memory and user file share the same memory area.
   The maximum number of usable words by the total of two areas are as follows.

(1 word = 32 bits)

F55, F70, F80H	4,096 words
F120H, F70S, F120S	6,144 words
F140S	100k words (1k = 1,024)
F150S	324k words (1k = 1,024)

6. If data memory exceeding 256 words is to be used, the number of words to be used must be defined in the system definitions. (In the F30, F50, F50H, F60 series, the data memory cannot be expanded for use.)
   - F55, F70, F80H, F120H
     ..... Max. 4,095 words (up to BD4094)
   - F70S, F120S, F140S, F150S
     ..... Max. 4,096 words (up to BD4095)

7. Memory range of each MICREX-F series
   (1 word = 32 bits)
   - F30, F50, F50H series BD0000 to BD0127 (128 words)
   - F60 series
     ..... BD0000 to BD0255 (256 words)
   - F55, F70, F80H, F120H series
     ..... BD0000 to BD0255 (Expandable up to BD4094)
   - F70S, F120S, F140S, F150S series
     ..... BD0000 to BD0255 (Expandable up to BD4095)



2-2-18 User file areas (identifier: W30 or larger, data module No.: 30 or larger)

The user file area is used to store internal processor data for the following purposes.

1) To control the tracking of process information in production steps by using the FFST (FIFO and FILO store), FIFO (first-in first-out) and FILO (first-in last-out) instructions
2) To design a data table in the user program
3) To input and output data through message communication
4) To specify the I/O expansion area to transfer I/O data (one word = 16 bits and a memory size of 2048 words is used) (Supported by F70S, F120S, F140S, F150S series only.)
5) To use the PE-link expansion area (the memory on the card is used)
6) To use the ME-NET relay link (the memory on the card is used) (Supported by F120S, F140S, F150S series.)
7) To expand BD areas (the expanded area occupies the user file areas)
   1 word = 32 bits
   F55, F70, F80H, F120H series
   ..... Max. 3,839 (4,095 - 256) words are used.
   F70S, F120S, F140S, F150S series
   ..... Max. 3,840 (4,096 - 256) words are used.

Key points

1. Even when the power supply is turned OFF or the processor stops, the preceding data is retained (nonvolatile area). The status of the memory to be used as the I/O expansion area is the same as the B area status. The memory to be used as the PE-link expansion area and ME-NET link relay becomes a volatile area.
2. The size (number of words) and data format must be defined for each file number as shown below. If the area is used as the I/O expansion, PE-link expansion, or ME-NET relay link area, the size and data format are defined in the system definition. Do not use the FILE or TABL instruction to define them.

Symbol			Data format	Range of value
For D20 and LITE	For D10S	Meaning		
SI	0	Single integer	Binary 16 bits	0 to FFFF
DI	1	Double integer	Binary 32 bits	0 to FFFFFFFF
BD	2	Binary Coded Decimal	BCD 8 digits	-79,999,999 to +79,999,999

3. Memory capacity

(words)

Series	F30, F50, F50H	F60	F55, F80H, F70	F120H, F70S, F120S	F140S	F150S
Number of words (one word = 32 bits)	64	512	3840	5888	102144	331520
Number of words (one word = 16 bits)	128	1024	7680	11776	204288	663040

Notes: 1. If the BD area is expanded in system definition, the user file area is reduced.
2. Up to 4095 words can be defined by one file definition instruction. (4096 words for F70S, F120S, F140S, F150S series)

(words)

	F30, F50, F50H	F60	F55, F70, F80H	F120H	F70S, F120S, F140S, F150S
1 word = 32 bits	64	512	3840	4095	4096
1 word = 16 bits	128	1024	4095	4095	4096

3. If I/O expansion is specified, the user file area for 2048 words (one word = 16 bits) is used.
4. The data table is for read-only files, it cannot be written by programs.
   Because the data table is designed in the user program, the memory areas described in the above item 3 are not used.
5. Because the memory on the card is used as the PE-link and ME-NET expansion areas, the memory capacity in the above item 3 is not used.

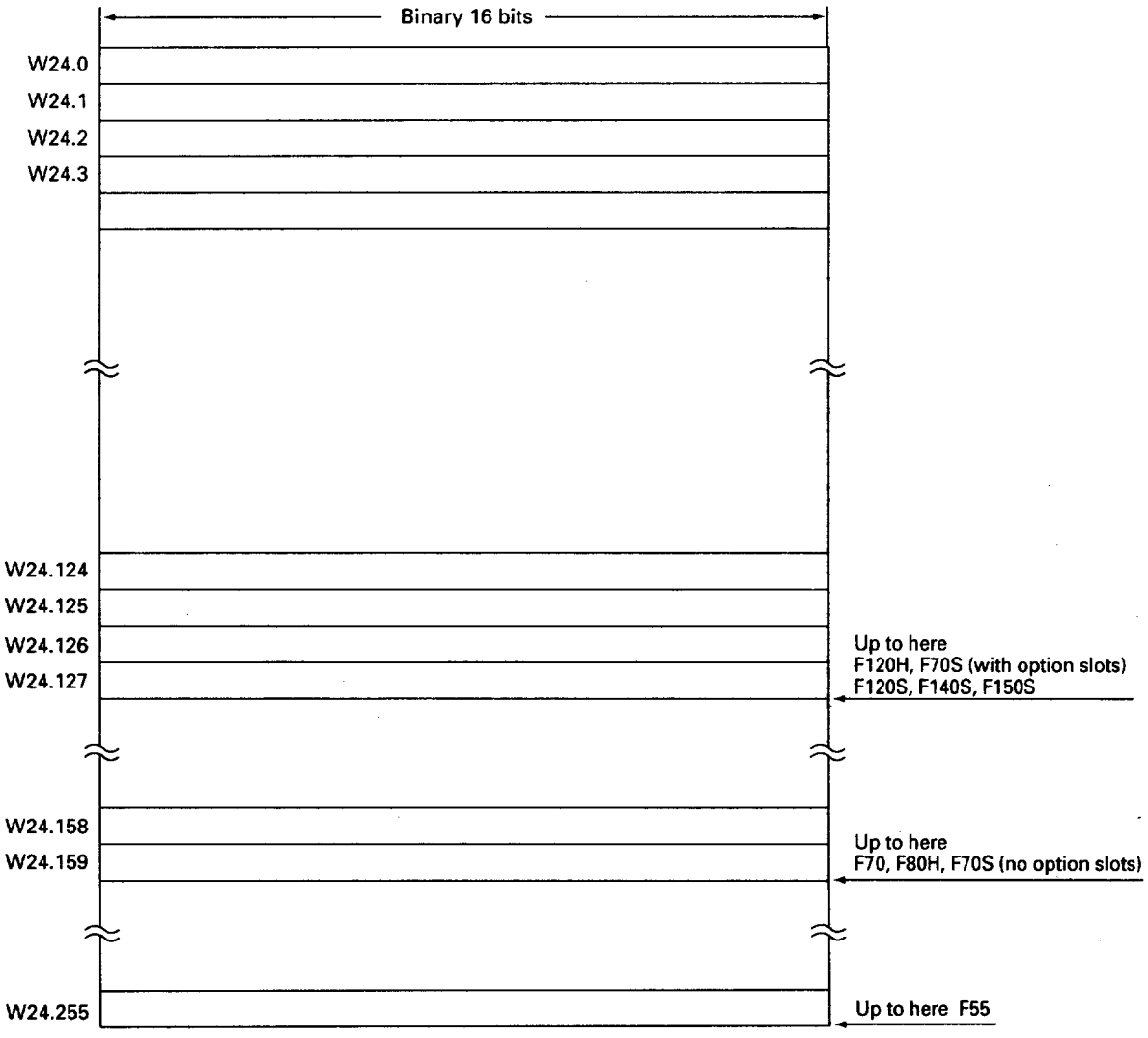
2-2-19 Direct access areas (identifier: W24, data module No.: 24)

There are two methods for processor I/O data access: (A) synchronous scanning method and (B) direct access method. The MICREX-F Series processors basically use method (A). The following PCs allow users to create programs that can incorporate both methods.
F55, F70, F80H, F120H, F70S, F120S, F140S, F150 series
The direct access area is used to execute processing using method (B) for the following purposes.

1. To shorten the I/O response delay due to program scan time
   (The direct accessing can be executed by I/O modules directly connected to the processor.)
2. To minimize the I/O response delay in positioning operations using a high-speed counter
   (The FTU500A high-speed counter module is used.)
3. To execute external interrupts (The external interrupt module is used.)

Key points

1. A 128-word (a 160-word for F80H, F70, F70S without option slots) area is reserved separately from the I/O area (B area). The user must declare to use the W24 area in the system definitions.
2. The I/O modules that use the W24 area must be mounted directly on the base unit on which the processor is mounted.
3. The W24 area cannot be accessed in units of bits, it can be accessed only in units of words.



2-2-20 P-link (PE-link) memory areas (option)

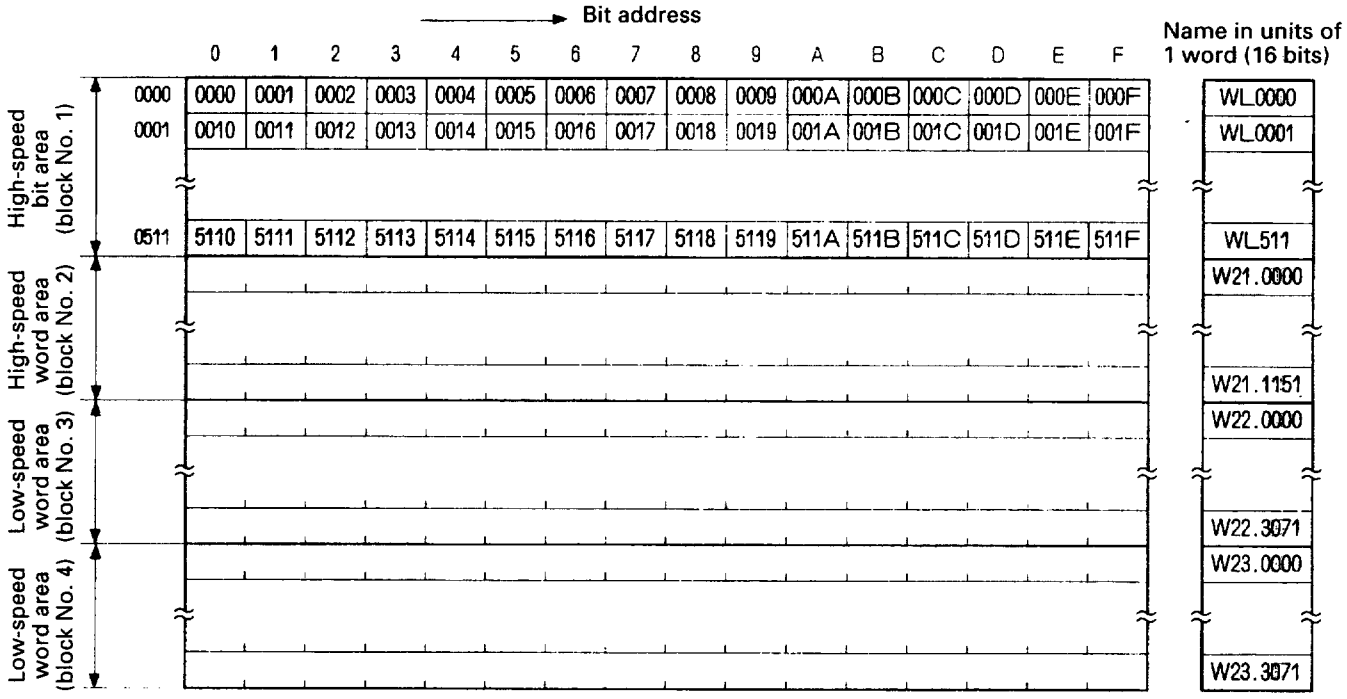
Identifier: L, WL, W21, W22, W23, W120, W121, W122, W123
Data module No. for P-link (PE-link) 0: 20, 21, 22, or 23
Data module No. for P-link (PE-link) 1: 120, 121, 122, or 123

In the processor modules with a P-link (PE-link), this memory area is used for data transfer between processors. The processor can be connected to two P-links (PE-links) by using optional P-link (PE-link) memory. When the P-link (PE-link) card is not mounted, P-link (PE-link) memory cannot be used.

Key points

1. When this area is used for data transfer between processors, the required items, including the assigned number of words, must be set in the system definitions.
2. The WL area is a bit area that can be used as a contact or coil in the user program.
3. The W21, W22, W23, W120, W121, W122, and W123 areas cannot be accessed in units of bits but can be processed only in units of words.
4. When the power supply is turned off, the contents of this area are cleared. (They are not cleared by stop and start operations of the program loader.)

P-link (PE-link) memory for P-link 1 (PE-link 1)

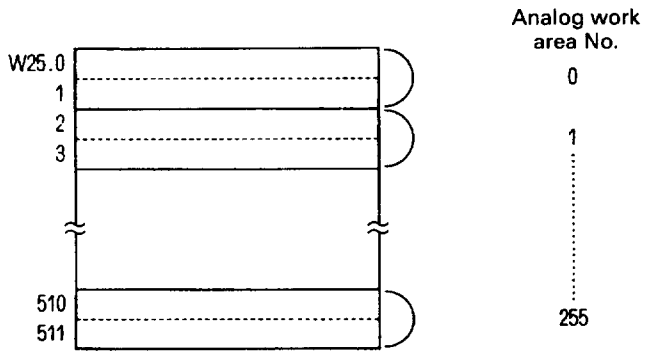


Note 1: For P-link 2 (PE-link 2), the P-link (PE-link) memory can be used as follows:
Block No. 1: W120.0000 to W120.0511
Block No. 2: W121.0000 to W121.1151
Block No. 3: W122.0000 to W122.3071
Block No. 4: W123.0000 to W123.3071

Note 2: PE-link can be used only for F120S, F140S, F150S series.

2-2-21 Analog work area

The analog work area is used for the FIL, DIF, INT, and HOLD instructions.
The analog work area W25.000 to W25.511 can be accessed. The area for two data items is used for one instruction.

Item	Description
Data module No.	W25 (0 to 511) R/W enabled *)
Data format	Binary 32 bits (Double integer)
Initial data	Preceding value is retained
Memory map	

*One work area uses two data.

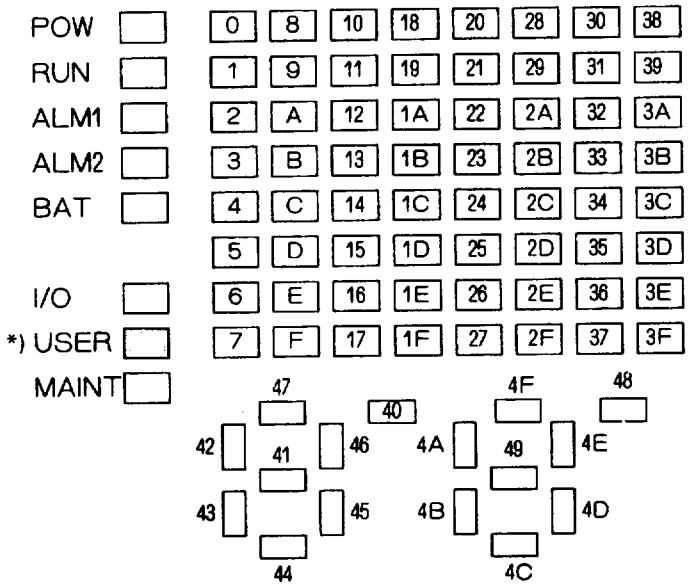
Use of work area	Instruction	FIL	DIF	INT	HOLD
	Word 1	Output data	Output data	Output data	Output data
	Word 2	Division remainder	Unused	Division remainder	Unused

*) This area is supported only by F70, F120, F140S, F150 series.

2-2-22 User display area (identifier: W124) (F55 Series only)

The 8-line x 8-line LEDs and two 7-segment LEDs on the F55 Series basic unit can be used to indicate optional information by the user program.

Base unit LED arrangement



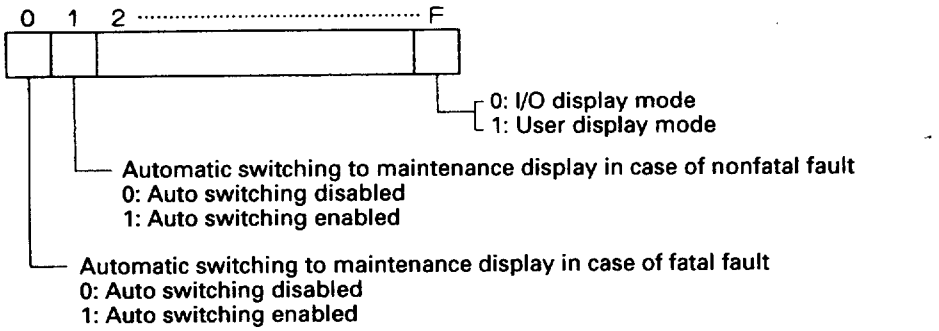
*) This LED (USER) is lit in the user display mode.

Corresponding memory area

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
W124.0	Control word															
W124.1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	D	F
W124.2	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
W124.3	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
W124.4	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
W124.5	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

Each bit
1: LED ON
0: LED OFF

Control word



2-3 I/O Address

2-3-1 Relay No. (Bit designation) and word No.

Each memory area has an individually assigned number used to designate the memory area by the user program. This area number is divided into the "relay No." used to specify a single bit (contact, coil) and the "word No." used to specify a single word of 16-bit or 32-bit data. The following areas can be specified with the relay No. and word No.

- I/O relay
- Auxiliary relay
- Keep relay
- Special relay
- Annunciator relay

These areas are represented as follows using an example of the I/O relay area.

(1) Specifying a single bit (contact, coil): Relay No.

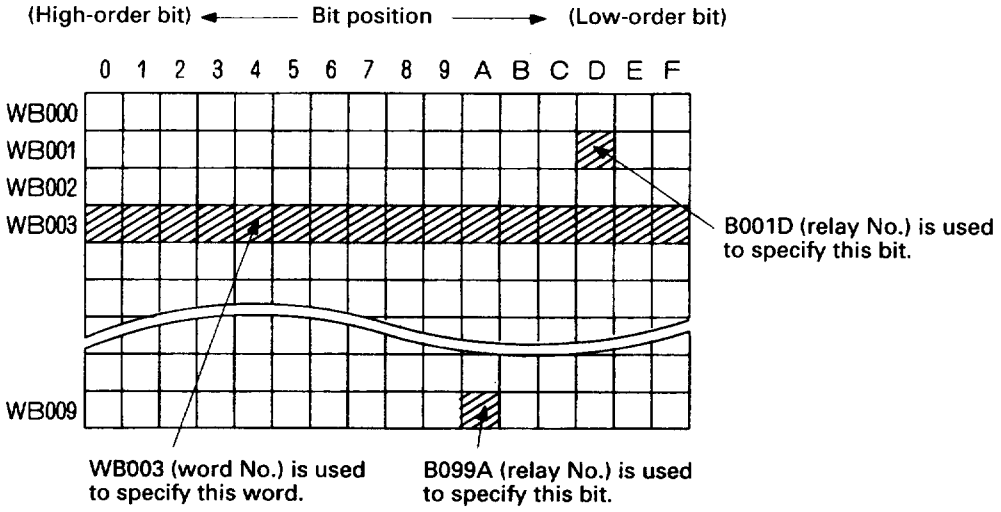
"B" indicating single-bit specification of the I/O area and the word address is placed before the bit position (hexadecimal).

Example: B001 D
Word address Bit position

(2) Specifying a single word (data): Word No.

"WB" indicating single-word specification of the I/O area is placed before the word address (decimal).

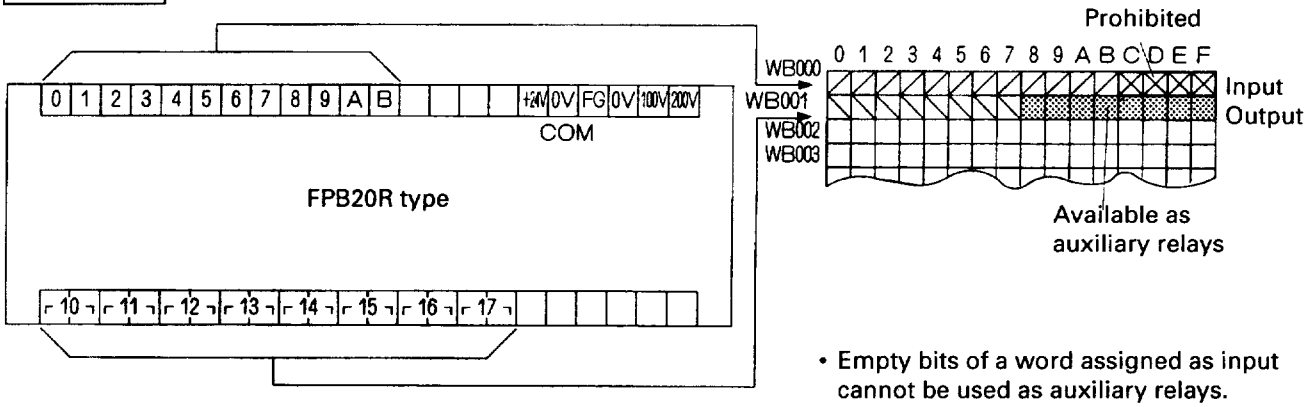
Example: WB003
Word address



2-3-2 Address assignment for the F30 Series processors

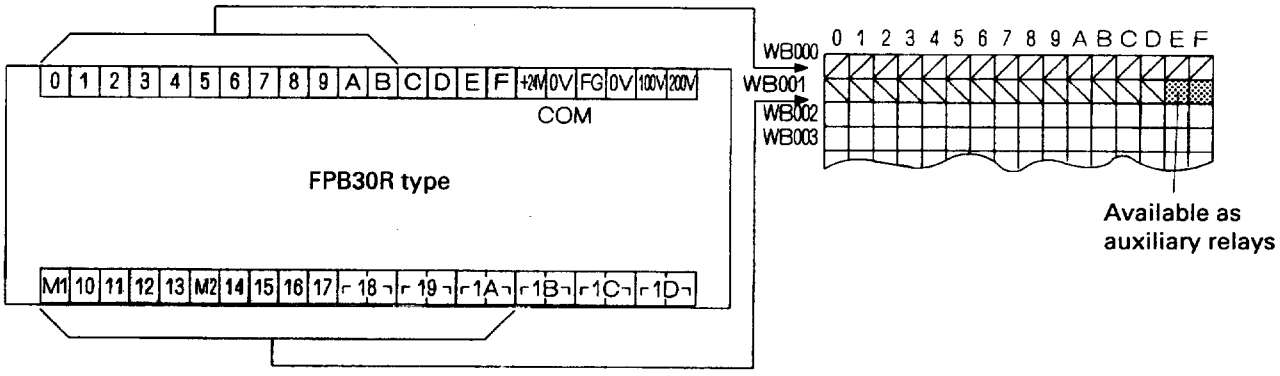
In the F30 Series processors, the basic unit I/O address exclusively uses 32 points (two words) from the leading address of the I/O area. The input section is assigned from B0000 and the output section is assigned from B0010. When expansion units are connected, the expansion unit connected to the base unit is assigned from B0020.

FPB20R-A10



- Empty bits of a word assigned as input cannot be used as auxiliary relays.
- Empty bits of a word assigned as output can be used as auxiliary relays.

FPB30R-A10



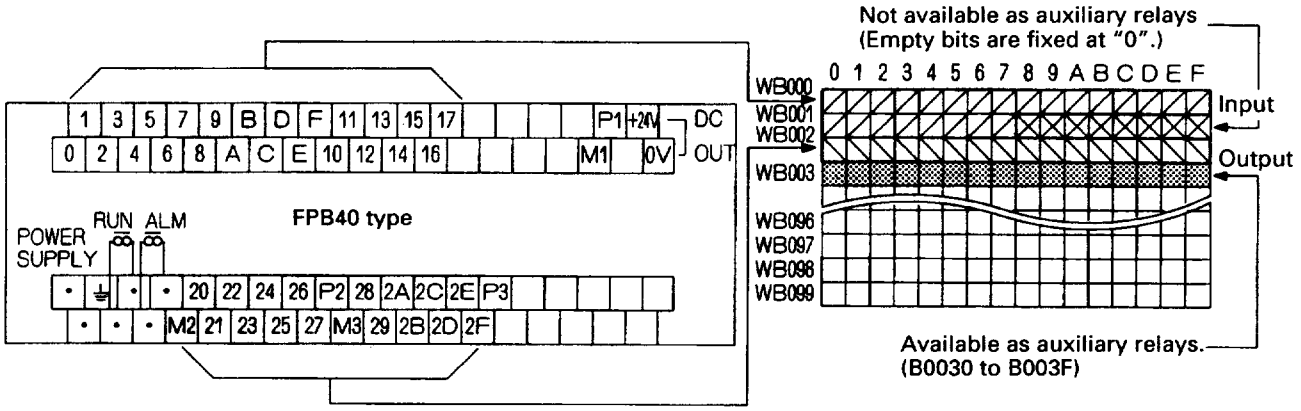
2-3-3 Address assignment for the F50/F50H Series processors

In the F50 and F50H Series processors, the basic unit I/O address exclusively uses 64 points (four words) for the 64-point unit and 48 points (three words) for the 40-point unit from the leading address of the I/O area. The input and output sections are assigned from B0000 and from B0020, respectively. The unassigned addresses can be used as auxiliary relays. (However, empty input areas are not available for the auxiliary relay.)

When expansion units are connected, the first expansion unit connected to the basic unit is assigned from B0040. The subsequent expansion units are forward-justified in units of words.

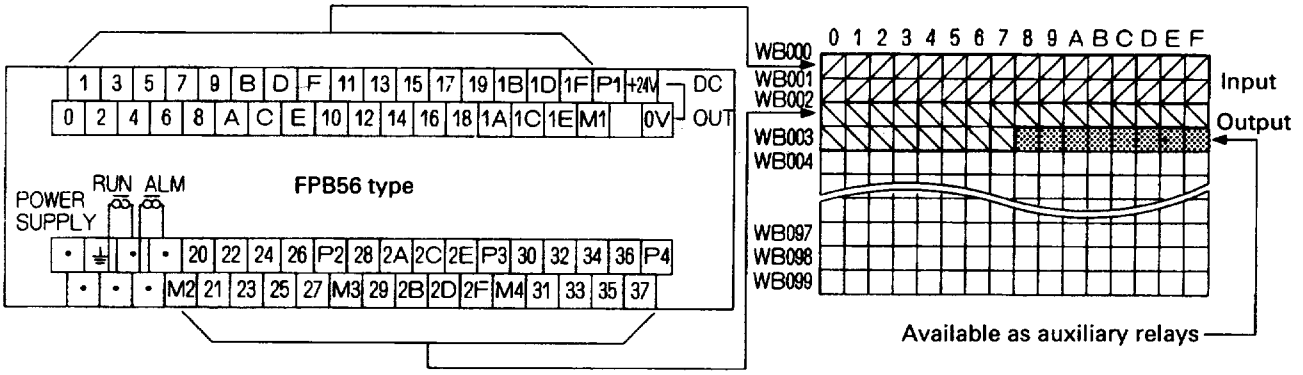
1. Example assignment in case of the basic unit only

Single unit of FPB40(H)R-A10



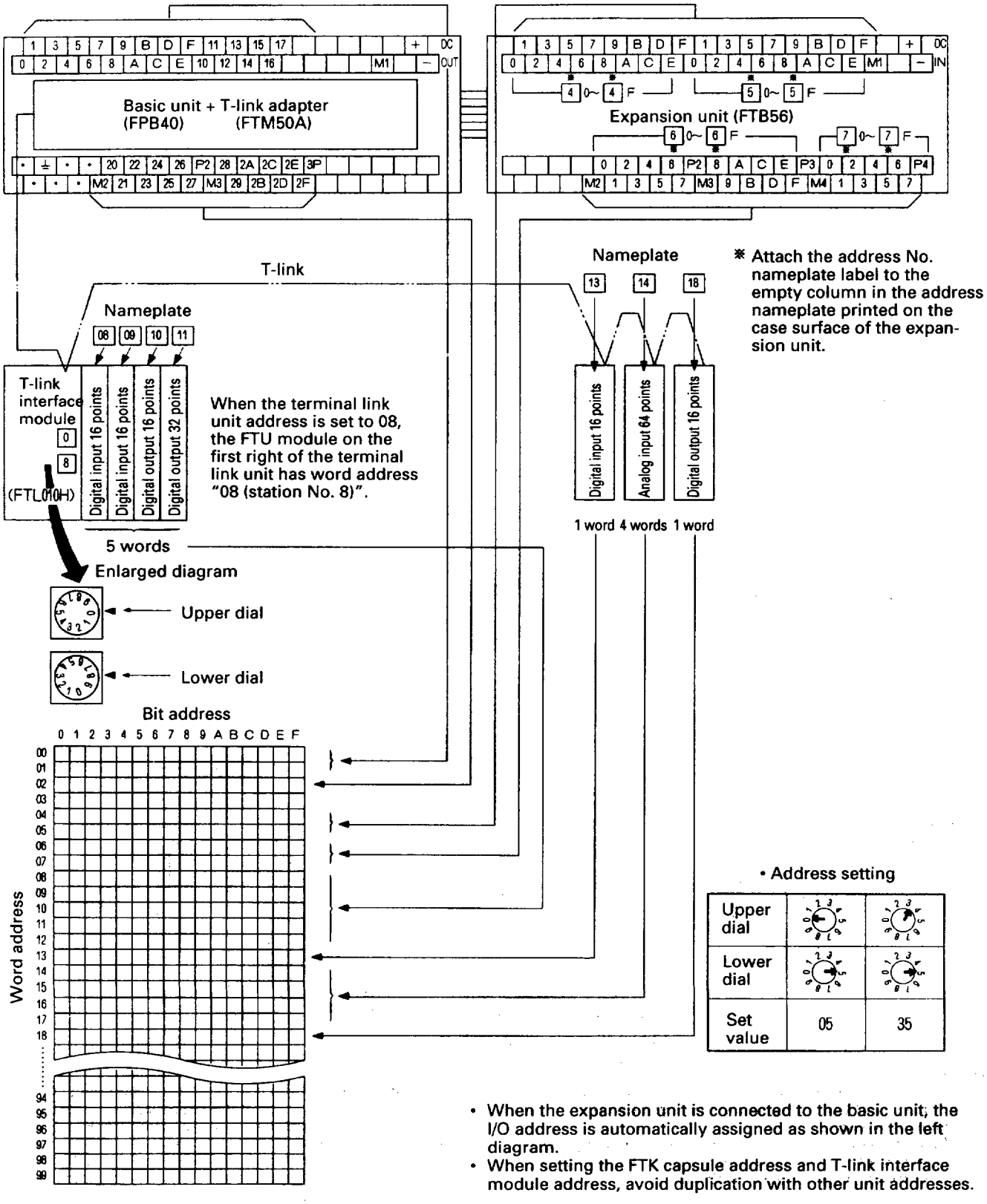
- Empty bits in the input area cannot be used as auxiliary relays.
- Empty bits in the output area can be used as auxiliary relays.

Single unit of FPB56(H)R-A10



2. Example assignment in case of using the T-link system for the F50 Series processor.

The following shows an address assignment example when the basic unit, expansion unit and T-link are used to connect the FTU module and FTK capsules. For the F30 Series, the I/O address can also be assigned in the same way as follows.



• Address setting

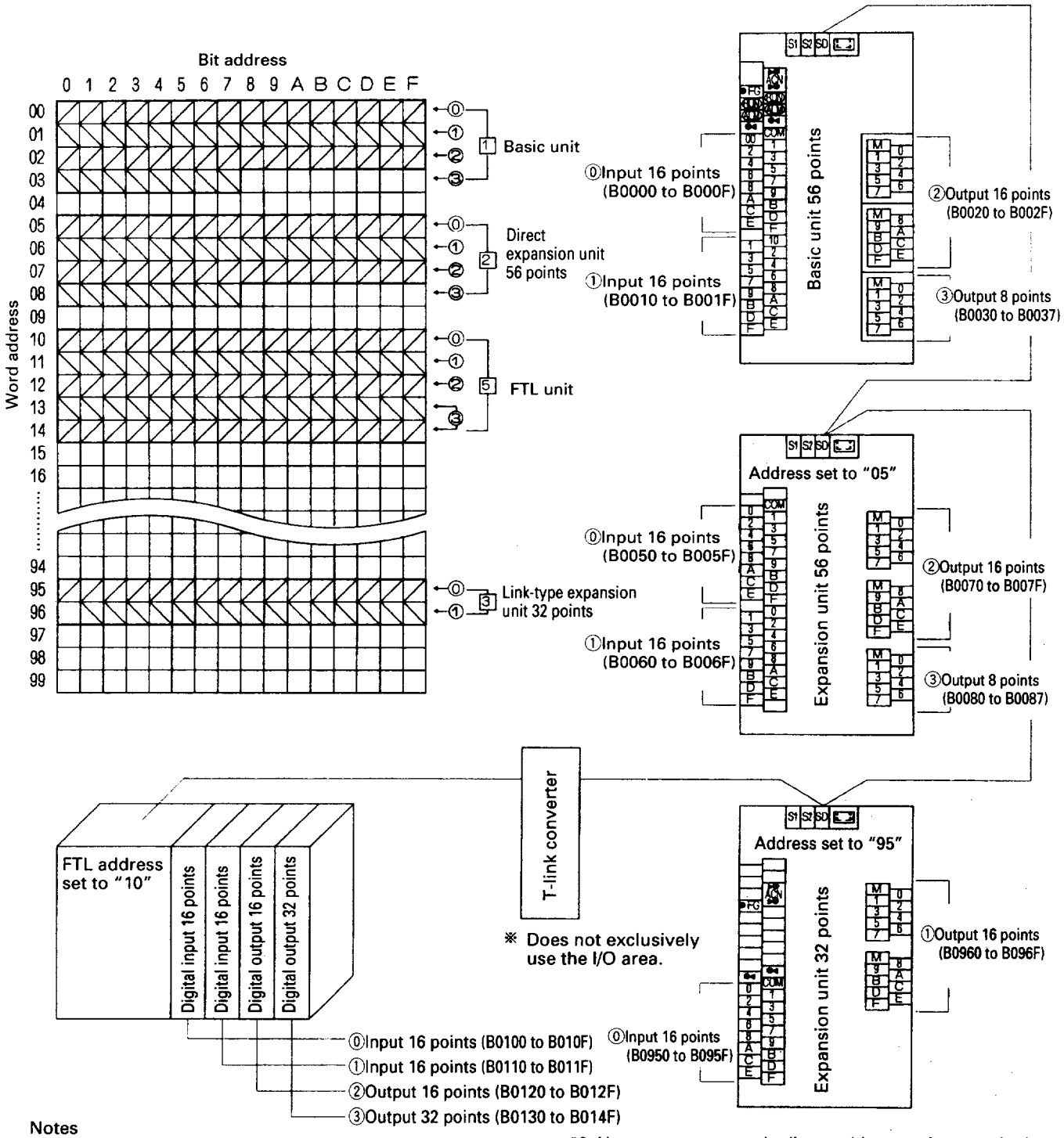
Upper dial		
Lower dial		
Set value	05	35

- When the expansion unit is connected to the basic unit, the I/O address is automatically assigned as shown in the left diagram.
- When setting the FTK capsule address and T-link interface module address, avoid duplication with other unit addresses.

2-3-4 Address assignment for the F60 Series processors

In the F60 Series processors, the basic unit I/O address exclusively uses 64 points (four words) from the leading I/O area. The input and output sections are assigned from B0000 and from B0020, respectively. The unassigned addresses can be used as auxiliary relays.

When expansion units are connected, the expansion unit address can be optionally set with the address setting dials. However, avoid duplication with the addresses of the basic unit, other expansion units and T-link/mini T-link devices.



Notes

*1 The basic unit is automatically assigned from word address "00" according to the number of words exclusively used.

*2 The expansion unit is assigned using the leading word address set by the address setting dials according to the number of words used exclusively.

*3 Use care to prevent duplicate address assignment in the I/O area with other units. If the address setting dials are set to the same address in two units, the "T-link setting error" will cause the PC to display a fatal fault and stop. In addition, when different numbers are set, the "I/O area duplication" fault will cause the PC to stop if a duplicate I/O area is assigned.

2-3-5 Address assignment for the F80H, F120H, F55, F70, F70S, F120S, F140S, F150S Series processors

The MICREX-F F80H, F120H, F55, F70, F70S, F120S, F140S, F150S Series usually have 100 words (1 word = 16 bits) of the I/O area on the T-link system, respectively. Within this area, word addresses are assigned to the I/O devices including FTU modules and FTK capsules.

When I/O area expansion is specified in the system definition *1), maximum 512 words can be assigned per T-link system (only single T-link system for the F80H, F55 and F70 Series).

1. Rules on address (T-link)

① In the F120H, F70S, F120S, F140S and F150S Series, a channel No. (link No.) 0 to 3 can be optionally set by using the T-link address setting dials. (The F80H, F55 and F70 Series processors have fixed channel No. 0.) (The I/O modules mounted on the same basic unit as the processor have the fixed channel No. 0.)

② If a connected module uses multiple words, the address of the next module is shifted by the extra number of words used.

③ A word address can be reserved between the modules on the same basic unit if a dummy module is inserted in the slot at the relevant address. (An unused slot without dummy module mounted causes a fatal fault.)

④ On a capsule-type unit, the address can be set by the address setting dials on the front panel of the unit. Any address can be set as long as it is not a duplicate address.

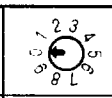
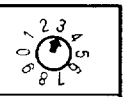
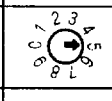
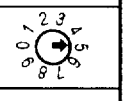
⑤ The module next on the right of FTL (expansion interface unit) has the same address as set by the address setting dials of the FTL. For subsequent modules, addresses are assigned in ascending order.

⑥ The number of I/O words is as follows:
WB area: Up to 400 words (6,400 points)
Expansion area: Up to 2,048 words *1)

⑦ The number of connectable capsules and devices are as follows:

Processor	Program loader	FTK capsule FFK capsule FTL interface T-link device
1 unit (2 units *2)	2 units per link at the same time	Max. total of 32 units (per link)

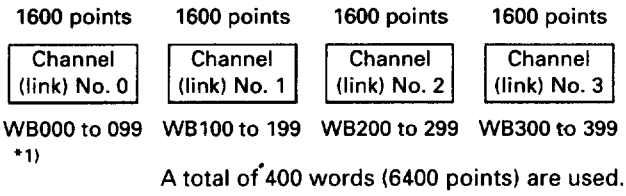
Address setting dial

Upper dial		
Lower dial		
Set value	05	35

Notes
*1) Supported only by the F70S, F120S, F140S and F150S Series.
*2) The F120H, F70S, F120S, F140S and F150S Series have the duplex processor system function.
For details, see the User's Manual (Communication) FEH161.

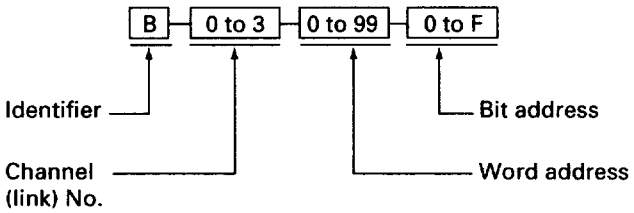
2. I/O address configuration (B areas)

On the F55, F70, F80H, F120H, F70S, F120S, F140S, F150S, B areas are commonly used as I/O areas by the Bus-link I/O modules and T-link I/O modules. (Bus-link I/O modules are the modules mounted on the processor's basic unit.) The B areas are divided into four banks, and 100 words (1600 points) of area are assigned to each channel (link).



Accordingly, channel numbers must be considered when addresses are set.
When assigning addresses to an FDL expansion I/O module or a T-link I/O module, use the address-setting dials as follows:

① Select a channel (link) No. from 0 to 3.
② Specify a word address (station No.) in the range from 0 to 99 within the selected channel.
③ To use a bit address (0 to F), set it during programming on a program loader.



*1) In the F55, F70, F80H series, only a total of 100 words (1600 points) of WB000 through WB099 can be assigned as I/O areas.

3. Example address assignment of I/O area (B area)
An example address assignment is shown below:

Address assignment for I/O modules on the base board on which the processor is mounted.
① For scan synchronization mode

		0	1	2	3	4	5	6	7
PS *	CPU	WB 0	WB 1 . 2	WB 3	WB 4	WB 5 to 8	WB 9 . 10	WB 11	WB 12
		16	32	16	16	64	32	16	16 ← No. of I/O points

* Power supply

The channel number in B area is fixed to 0. Addresses are assigned to the slots in ascending order from slot 0.

② For direct access mode

		0	1	2	3	4	5	6	7
PS *	CPU	W24.0 to 15	16 to 31	32 to 47	48 to 63	64 to 79	80 to 95	96 to 111	112 to 127

* Power supply

Fixed 16-word addresses are assigned to all slots. The range of addresses is W24.0 to W24.127.
In this case, addresses beginning with WB0 can be assigned to the T-link unit.

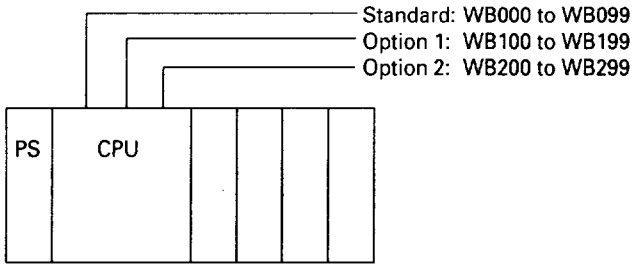
Address assignment for T-link I/O

A T-link unit has two address-setting dials to set station numbers 0 to 99. The channel number (called "link No." for T-link) is set by the link number setting dial on the T-link interface board mounted on the processor. (Channel 0 to 3)

Setting example 1:

■ Link No. settings

T-link interface board	Link No. setting dial
Standard	0 → Link No. 0
Option 1	1 → Link No. 1
Option 2	2 → Link No. 2

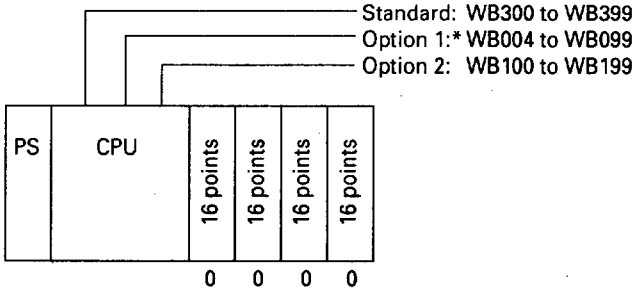


PS: Power supply

Setting example 2:

■ Link No. settings

T-link interface board	Link No. setting dial
Standard	3 → Link No. 3
Option 1	0 → Link No. 0
Option 2	1 → Link No. 1



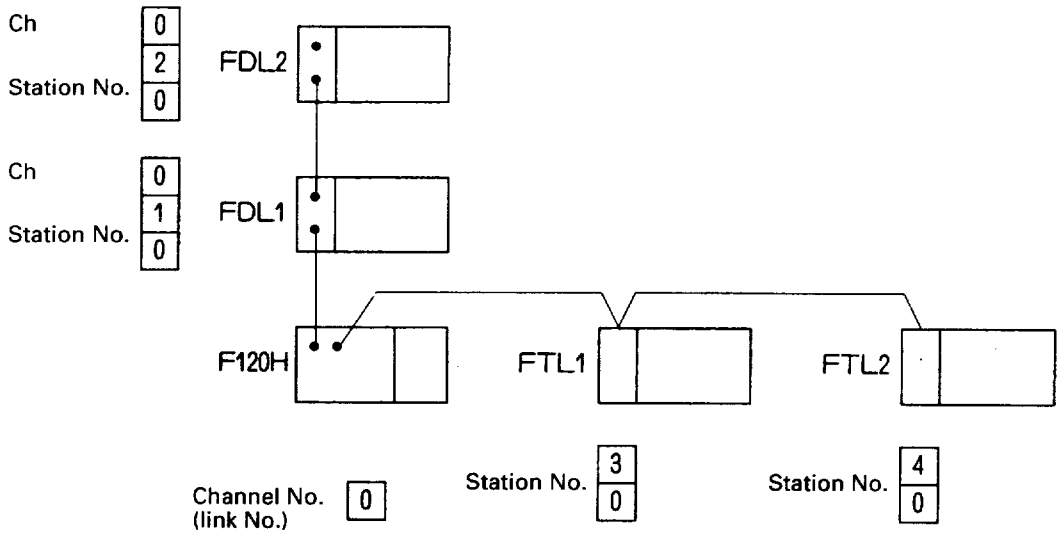
The first address for option 1 follows the last word address of the modules that are mounted on the same basic unit as the processor.

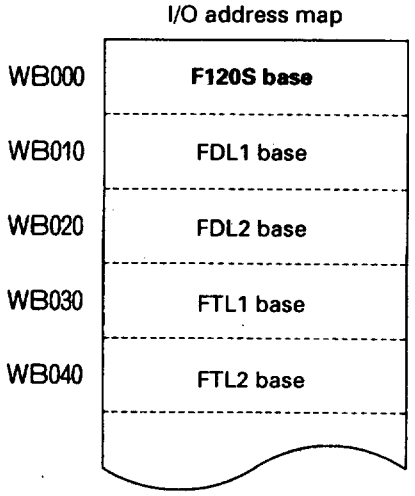
On the standard T-link interface board, the link number setting dial is set to 0 (link No. 0) before shipment.

Notes on address assignment

I/O area and addresses link numbers must not be duplicated. If there is a duplicate, a fatal fault occurs and the processor cannot operate. Addresses for a T-link unit must not be assigned over two links (channels).

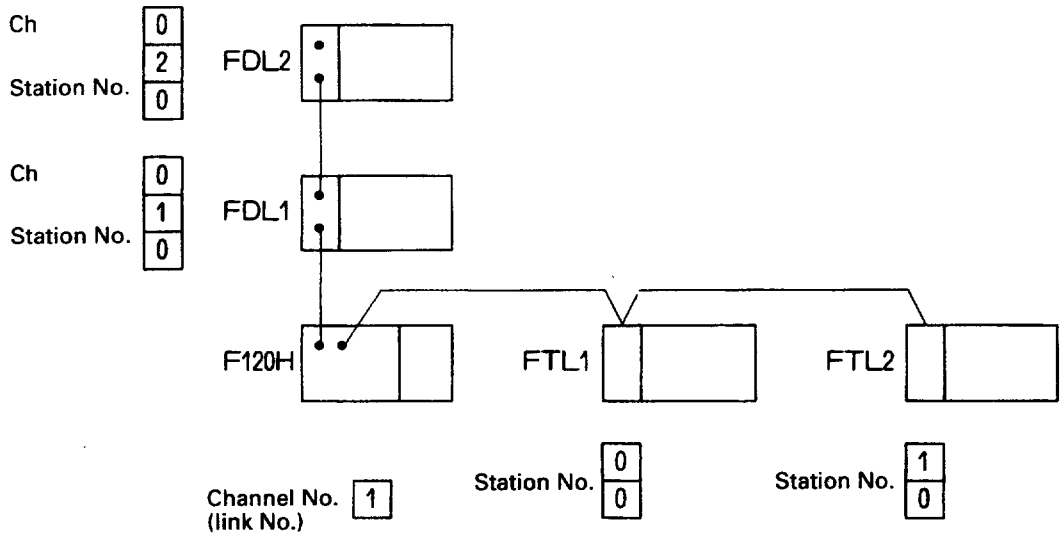
Setting example 3
Using channel No. 0 for both FDL expansion bus link I/O and standard T-link I/O *1

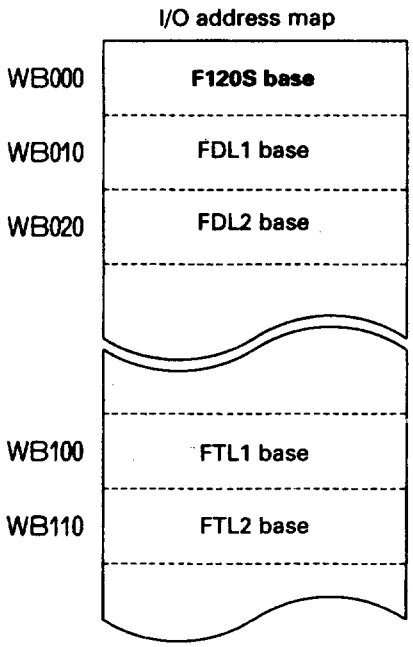




*1 FDL expansion bus link can be used only for the F120H, F120S, F140S and F150S Series.

Setting example 4
Using channel No. 0 for FDL expansion I/O and channel 1 for standard T-link I/O *1





*1 FDL expansion bus link can be used only for the F120H, F120S, F140S and F150S Series.

2-3-6 Specifying I/O expansion area

In the MICREX-F series, up to 32 T-link I/O units (total of 100 words) can usually be connected per T-link system. If up to 32 T-link I/O units with a large number of words is connected per T-link system, 100 words may be exceeded.

In the F120S, F140S, F150S, the T-link I/O units can be used by specifying an I/O expansion area for only T-link I/O units with the large number of words in the system definitions.

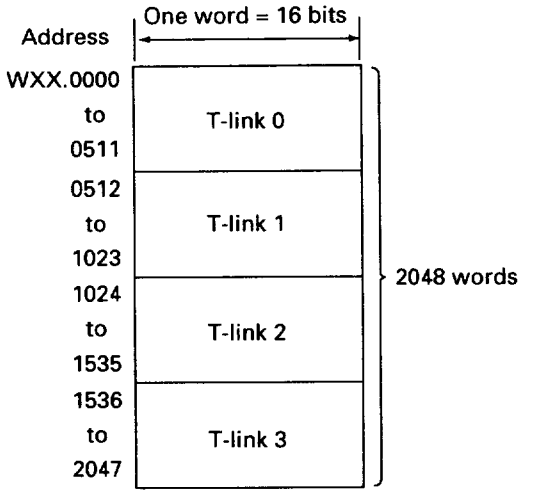
Specifying I/O expansion area

While normal I/O unit is assigned to the area B of a processor, T-link I/O unit with an I/O expansion are specified is assigned to the user file area (W30 or larger). The address and No. of words used are as follows:

Start address:
(Link No. x 512) + (T-link station number x 16)
where Link No.: 0 to 3
T-link station number: 00 to 31
(T-link station number 32 to 99 are not assigned to an I/O expansion area)

Note: Please refer to Appendix for the address and the station number.

No. of words:
One station number uses 16 words. An I/O unit that uses 64 words uses the T-link station number for four stations. For 64-word I/O, set the station number to 28 or less. For I/O modules mounted on the base board on which the processor or FTL module is mounted, each module uses 16 words.



• Notes on I/O expansion area specification

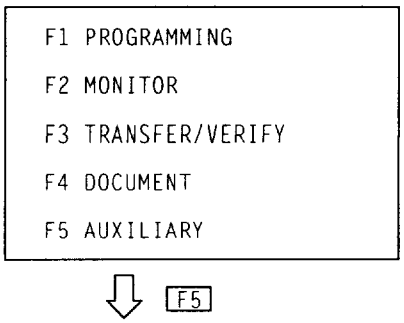
To use this function, the following 3 items should be set:
① Use of I/O expansion area
② T-link I/O station number assigned to I/O expansion area
③ Data module No. (file No.) used as I/O expansion area

All of these items are specified in the system definitions. The processor recognizes the setting of items ① and ② at POWER-ON. The processor recognizes the setting of item ③ when the system definition is changed.

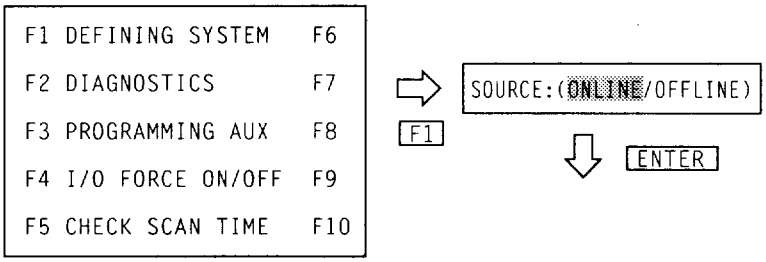
If the FILE or TABL instruction having the same data module No. as the specified data module No. (W30 or larger) exists, a user program error will cause a fatal fault and the processor stops.
If I/O expansion area is specified, a data memory (file memory) of 2,048 words is occupied (in 16 bit). The system definition can only specify the I/O area expansion and the same module number should not be specified in system definition and FILE definition.

• Steps for expanding I/O area
I/O expansion is specified in the system definition. An example of specifying I/O expansion with the program loader LITE is shown.

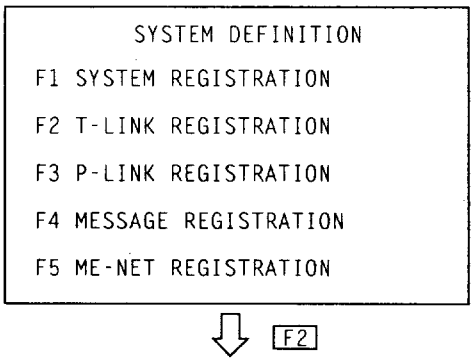
① Select [F5] "AUXILIARY" from the initial screen.



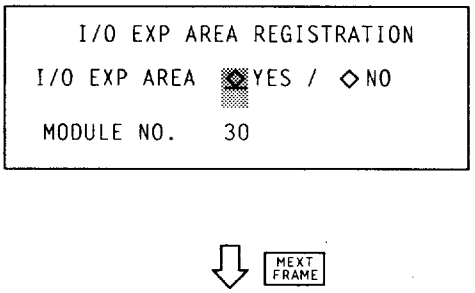
② Select [F1] "DEFINING SYSTEM," then press the [←] key to select "ONLINE" and press the [ENT] key.



③ Select [F2] "T-LINK REGISTRATION."

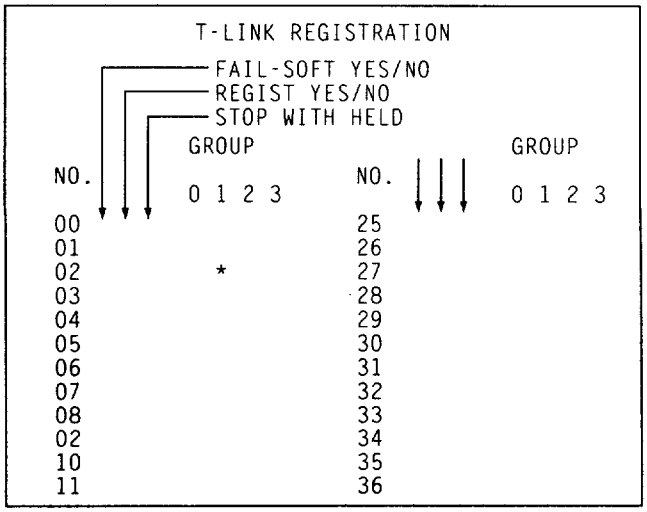


④ Press the [←] key to select "YES" for the I/O expansion area, then set the module No. (30 to 109) used as the I/O memory and press the [NEXT FRAME] key.



Note: Set the module No. so that it is different from all other expansion module Nos. (PE-link and ME-NET) in the system definition. The module No. must not be the same as the FILE No. and TABL No. in the user program.

⑤ Press the [*] key to register the station number for which I/O expansion is to be specified in group 1. After setting ends, press the [END] key, then press the [ENT] key to store the system definition.



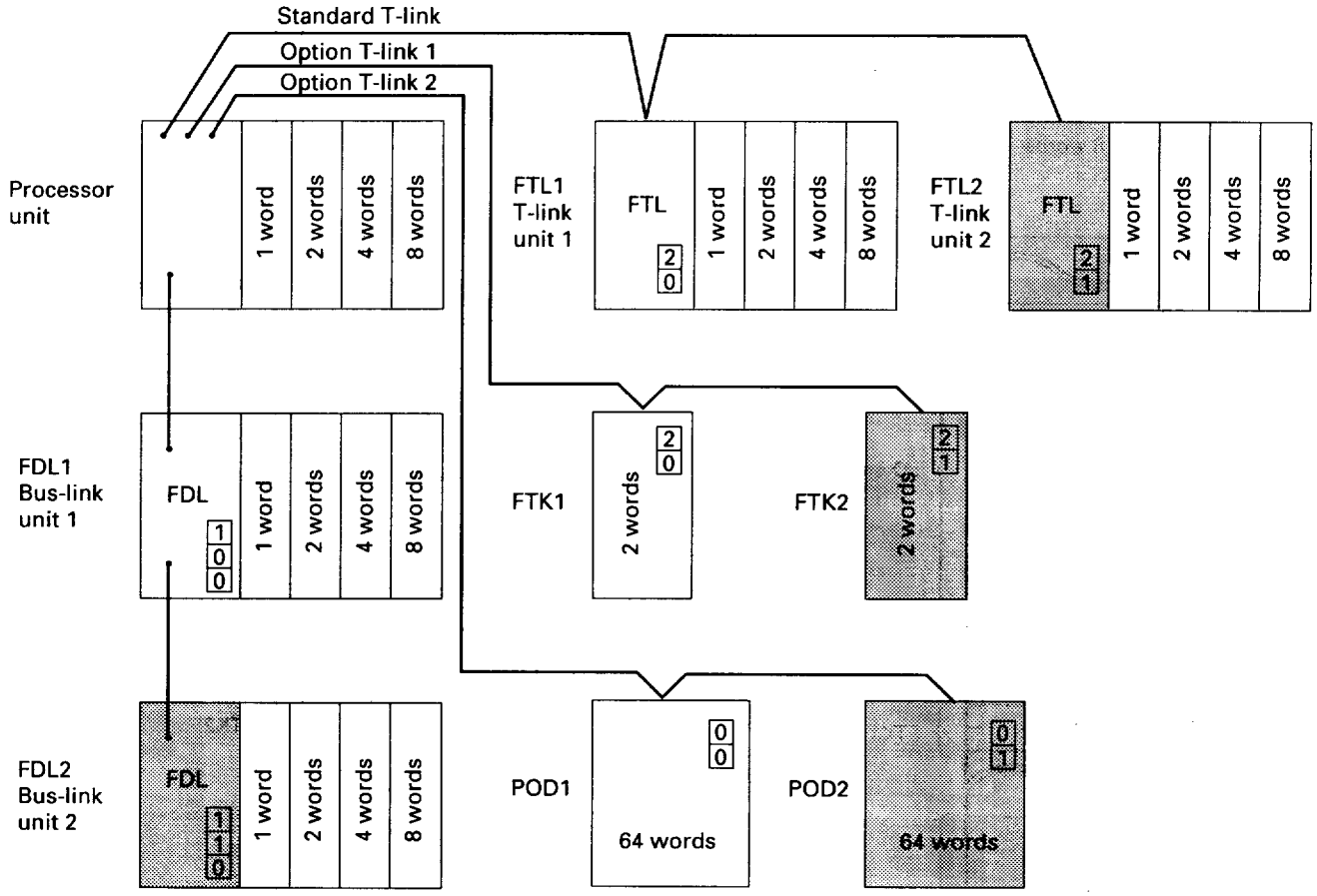
Note: After the system definition is changed, turn the processor off and then on. The processor recognizes the contents of the changed system definition when the power is turned on.

• Setting example

Setting the following numbers

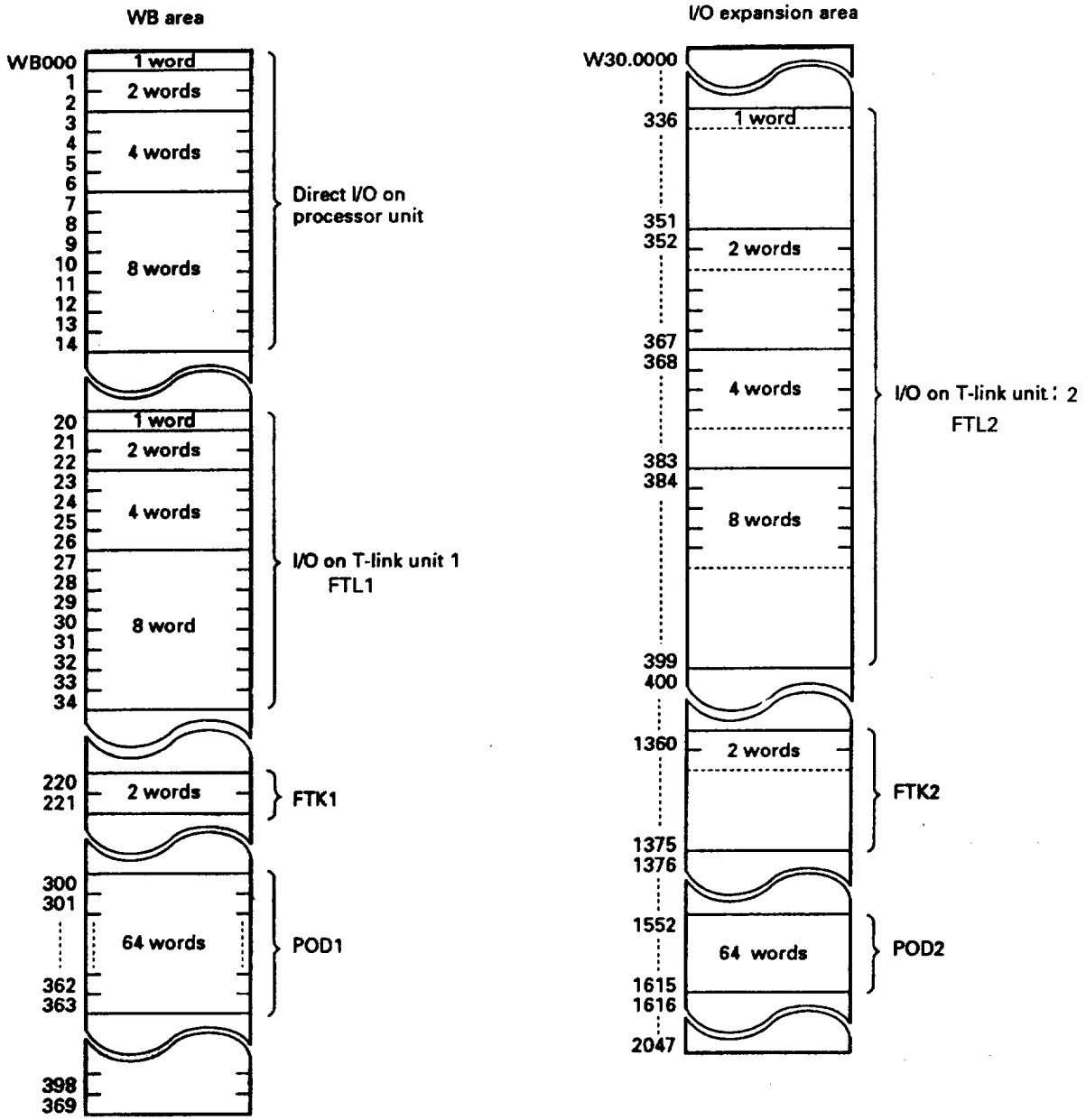
Bus-link I/O (FDL-link I/O): Link No. 1
Standard T-link I/O: Link No. 0
Option T-link 1 I/O: Link No. 2
Option T-link 2 I/O: Link No. 3

I/O expansion is specified for FDL module, FTL module, and FTK capsule and POD in the shaded area.



- See the next page for details on assignment of I/O addresses. (The I/O expansion area is specified in W30)
- Do not set the same station number to be used for T-link units (including FTK and POD), FDL units. When the set station number is the same even if the area is different, the duplicated station number causes the processor to stop due to a fatal fault.
- Data transferred with the capsule or module with I/O expansion area specified can be accessed only in units of word.

• I/O address map

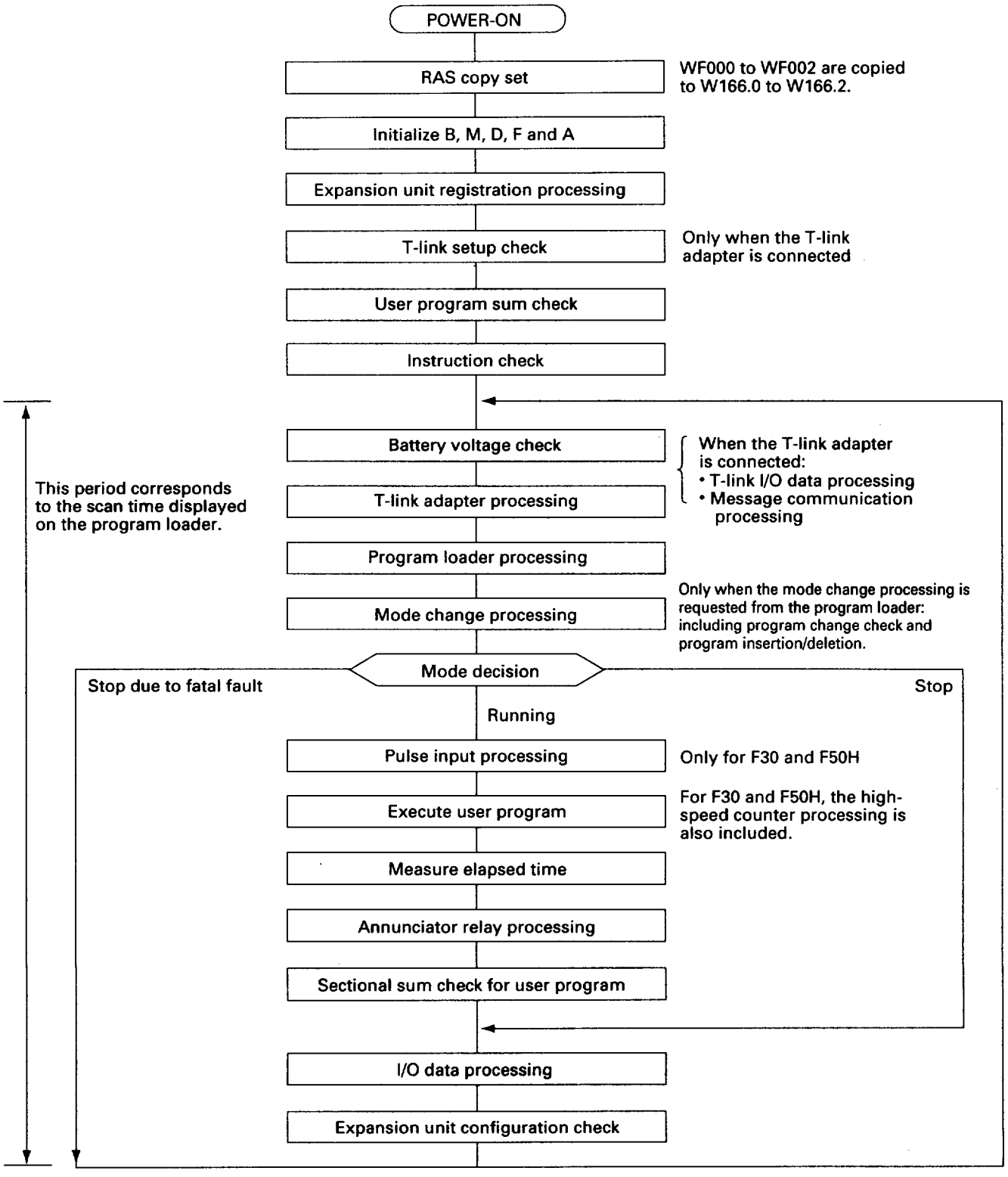


2-4 Program Processing

2-4-1 Operation flowchart

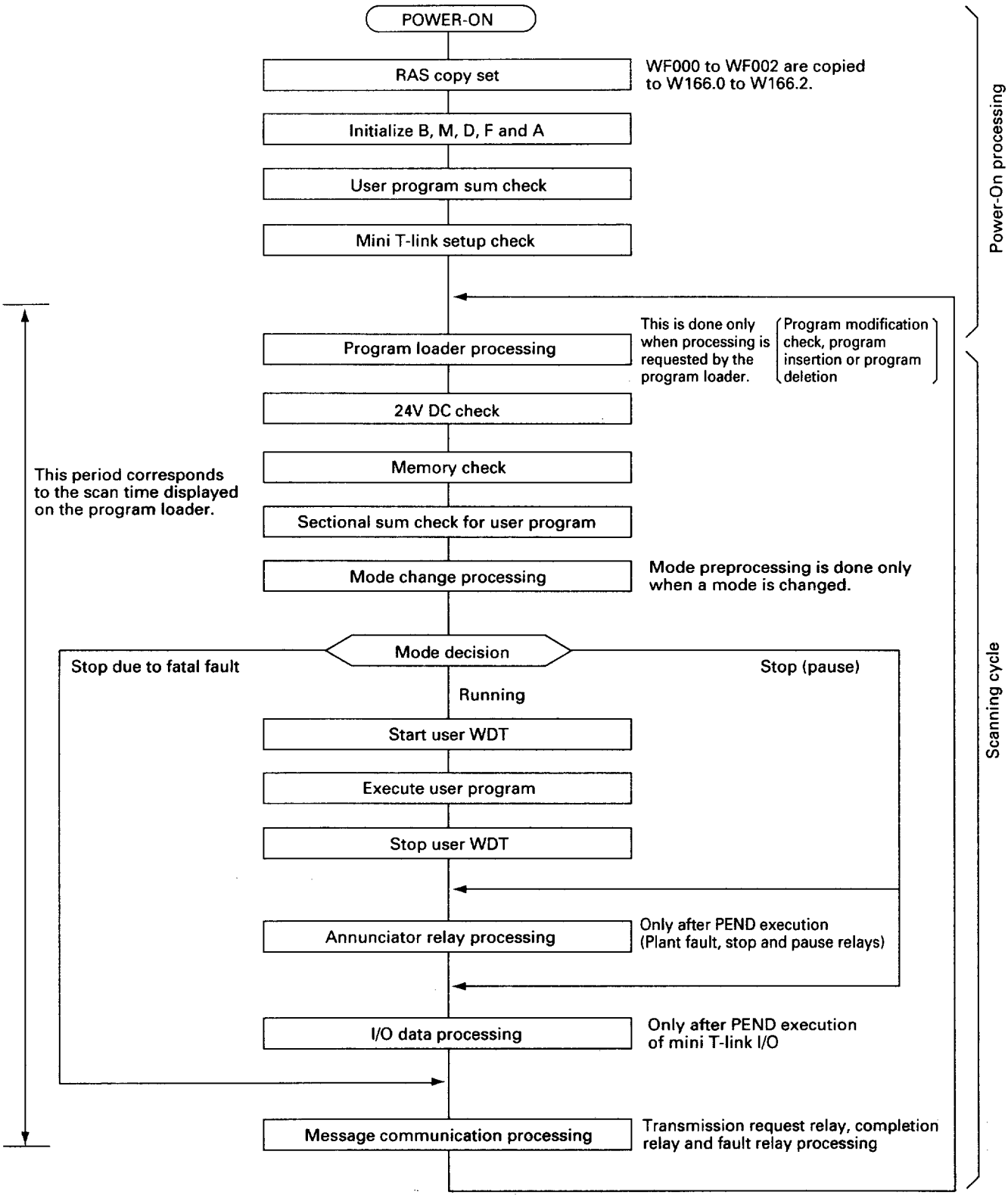
1. F30, F50 and F50H Series

The following flowchart shows the operations to be done when the basic unit power is turned on, as well as the subsequent operations.



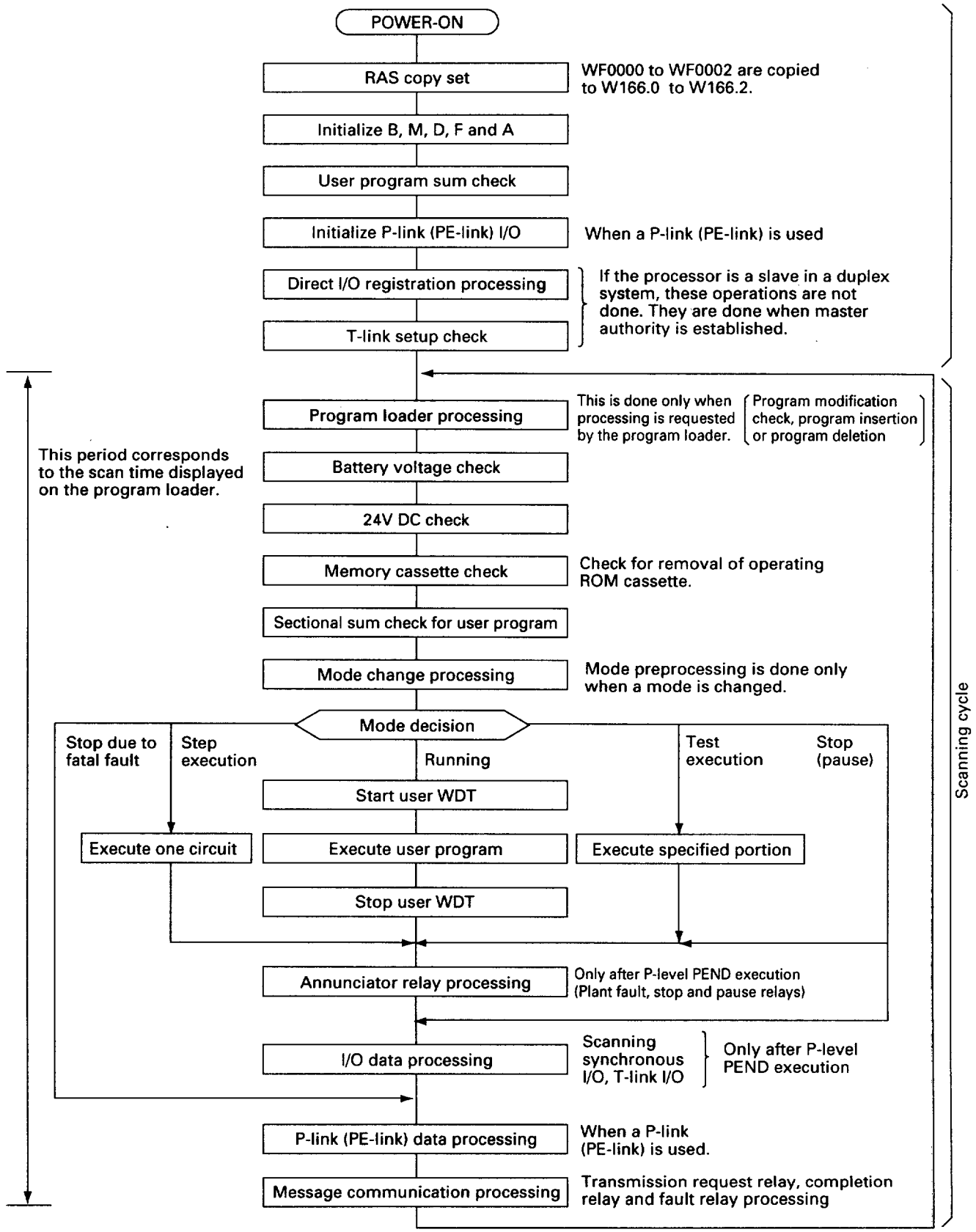
2. F60 Series

The following flowchart shows the operations to be done when the basic unit power is turned on, as well as the subsequent operations.



3. F55, F70, F80H, F120H, F70S, F120S, F140S, F150S series

The following flowchart shows the operations to be done when power is supplied to a processor module, as well as the subsequent operations.



2-4-2 Program types

Programs used for the MICREX-F series are classified into four groups. Applicable program types depend on each MICREX-F's series.

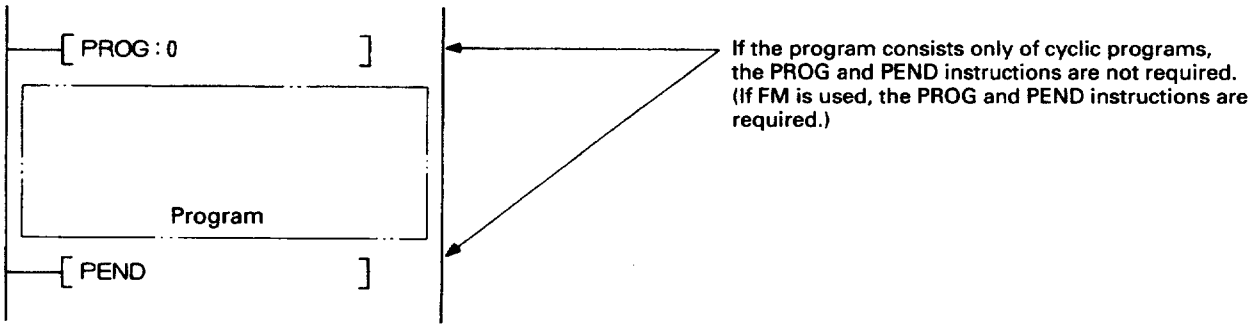
			F30, F50, F50H, F60	F55, F70, F80H F140S, F150S	F120H, F70S, F120S,
Cyclic program	PROG0	P-level	○	○	○
Fixed-cycle interrupt program	PROG50	Level 2	—	○	○
External interrupt program	PROG60 to 67	Level 1	—	○	○
Subroutine program (Function module)	FM0 to 63 *1)		—	—	○

*1) The number of function modules is 0 to 63 for F120H, F70S, F120S series; 0 to 255 for F140S, F150S series.

○: Available —: Not available

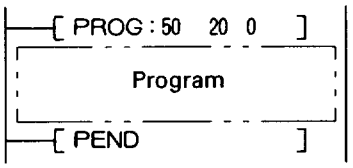
1. Cyclic program (P-level program) (program No. 0)

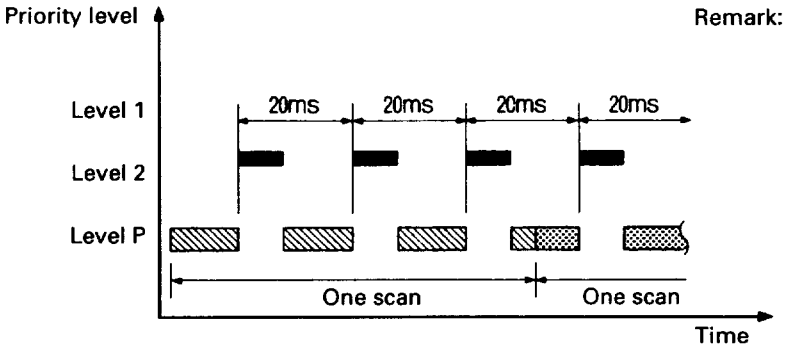
This program is always executed repeatedly. It must be incorporated into the system.



2. Fixed-cycle interrupt program (program No. 50)
Program No. 50 is a level 2 interrupt program. This program is executed for each operation cycle specified by the PROG instruction to refresh the I/O areas of the T-link I/O devices registered in group 0. (Refreshing between devices and buffers)

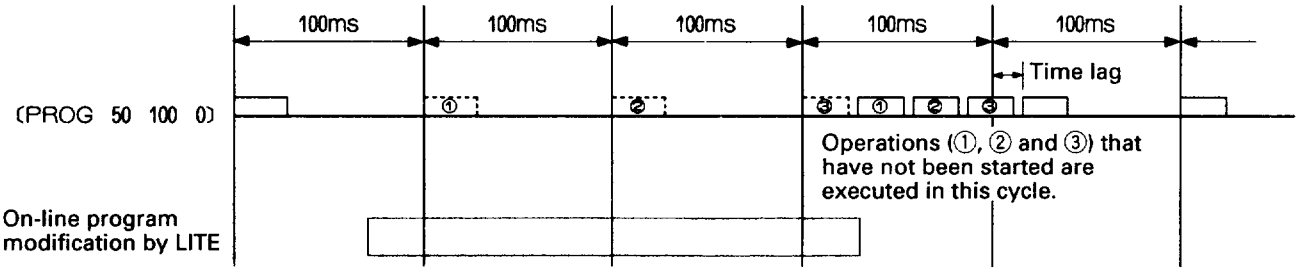
Applications:
1. If the scan time of a cyclic program is too long to follow the control-object operation, this program is used for the part that requires a high-speed response.
2. This program is used when an operation must be executed in a fixed cycle.





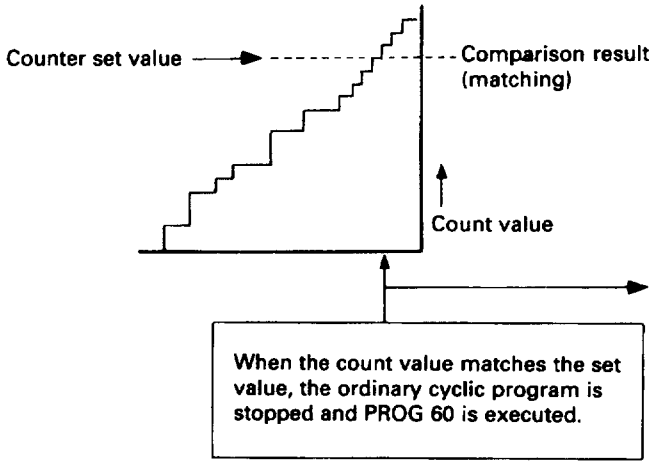
Remark: This figure shows an example of high-speed processing in a 20ms cycle.

Key points
1. I/Os registered in T-link group 0 execute I/O processing before and after execution of this program.
2. I/Os not registered in T-link group 0 execute I/O processing in synchronization with the cyclic program scanning.
3. Modules mounted on the same base board on which the processor module is mounted can be registered in T-link group 0 only when the synchronous scanning is specified for the modules. (If defined as a direct access area, the module executes I/O processing each time the program is executed for a circuit regardless of the program level.)
4. If program loader LITE operation is executed or interrupts occur while the processor is operating, the execution cycle becomes misordered as shown below. This must be noted.
5. If the number of queued interrupts exceeds 32, flag relay F002E is set to 1, and a nonfatal fault occurs.

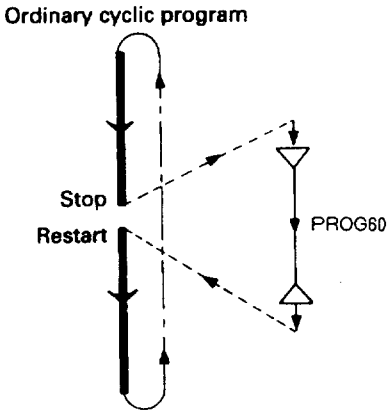


3. External interrupt programs (program No.: 60 to 67)
External interrupt programs have the highest priority level (level 1) and are executed after stopping the execution of programs (having lower levels P and 2).

Applications:
1. This program is used to execute specific processing immediately when the comparison result obtained by a high-speed counter module is output.



2. This program is used to immediately execute specific processing using external input from an external interrupt module.



Key points
An external interrupt program can be executed when the following conditions are satisfied.
1. Direct access is set in the system definitions.
2. The applicable modules are:
   - F55 series high-speed counter card (NV1F-HC1) and external interrupt module (NY1F-YP1)
   - F70, F70S series high-speed counter module (NC1F-HC1) and external interrupt module (NC1F-YP1)
   - F80H, F120H, F120S, F140S, F150S series
     High-speed counter module (FTU500A)
     External interrupt module (FTU520A)
3. An external interrupt program (PROG 60 to 67) is required.
4. The above modules must not be mounted on an expansion unit.
   They can be assigned only to a direct access area. (I/O slot on the base board on which the processor module is mounted.)

2-4-3 Program and processing

1. Interrupt program (only for the F55, F80H, F120H, F70S, F120S, F140S and F150S Series)

The MICREX-F Series assigns three priority levels, the cyclic program (also called P-level program), the fixed-cycle interrupt program and the external interrupt program. The following principles are established for executing these programs.

(1) If starting a program having high-level priority is requested while a program having lower priority is being executed, execution is temporarily stopped and the program having higher priority is executed.

(2) After an interrupt program is executed, execution of the stopped program continues from where it was interrupted.

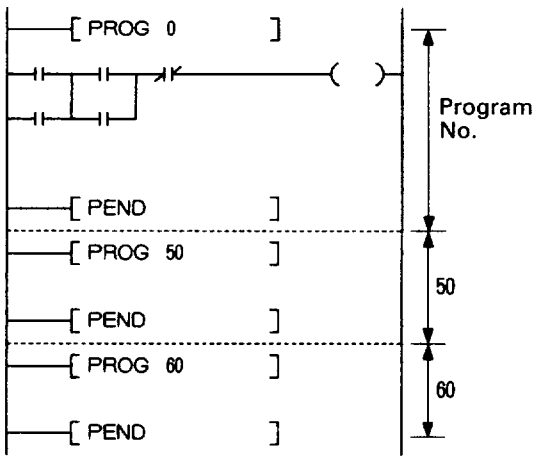
(3) If an interrupt is generated having a higher priority level than the interrupt program being executed, the current program is interrupted and the higher level interrupt program is executed.

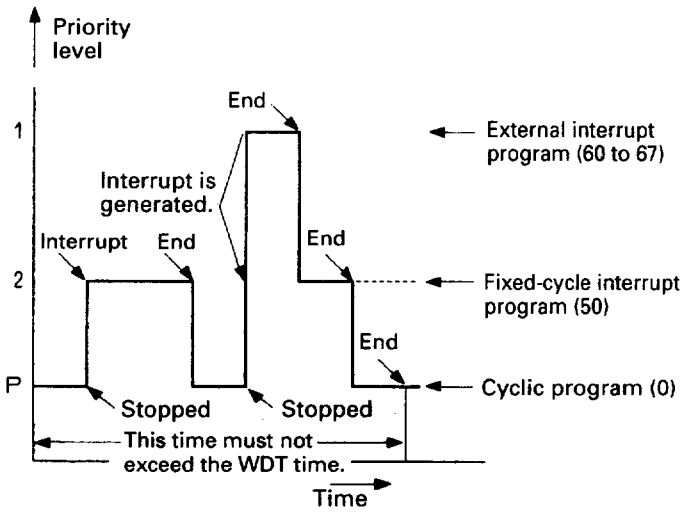
(4) If interrupts having different priorities occur at the same time, the interrupt programs are executed sequentially in order of priority.

(5) If interrupts having the same priority level occur at the same time, the interrupt programs are executed in ascending order of their program numbers.

Key Points

1. Each program must begin with a PROG instruction and end with a PEND instruction. (If only P-level programs are to be used without FMs, the PROG and PEND instructions are not required.)
2. At least one P-level program must be incorporated.
3. A subroutine (function module FM) can be called and executed by a program of any level.
4. The program scan time must not exceed the set value of WDT. (If it exceeds this value, a fatal fault occurs.)
5. Interrupt program execution is started at the end of the circuit being executed when the interrupt was requested.
6. If a new interrupt occurs that has a priority level equivalent to or lower than that of the current interrupt program being executed, the current interrupt program continues execution to its end. Then, the new interrupt program is executed.





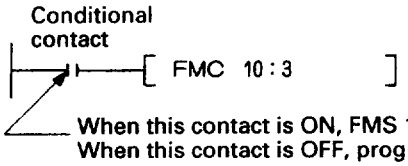
2. Function modules (FM) (Only for the F120H, F70S, F120S, F140S and F150S Series)

Some applications must frequently execute the same processing. Such programs can be created as subroutines or function modules (FM) and called as required during main program execution.

Function modules provide users with various advantages; i.e., simplified programming, easy understanding of the program, fewer program steps, improved program reliability and less time needed for debugging.

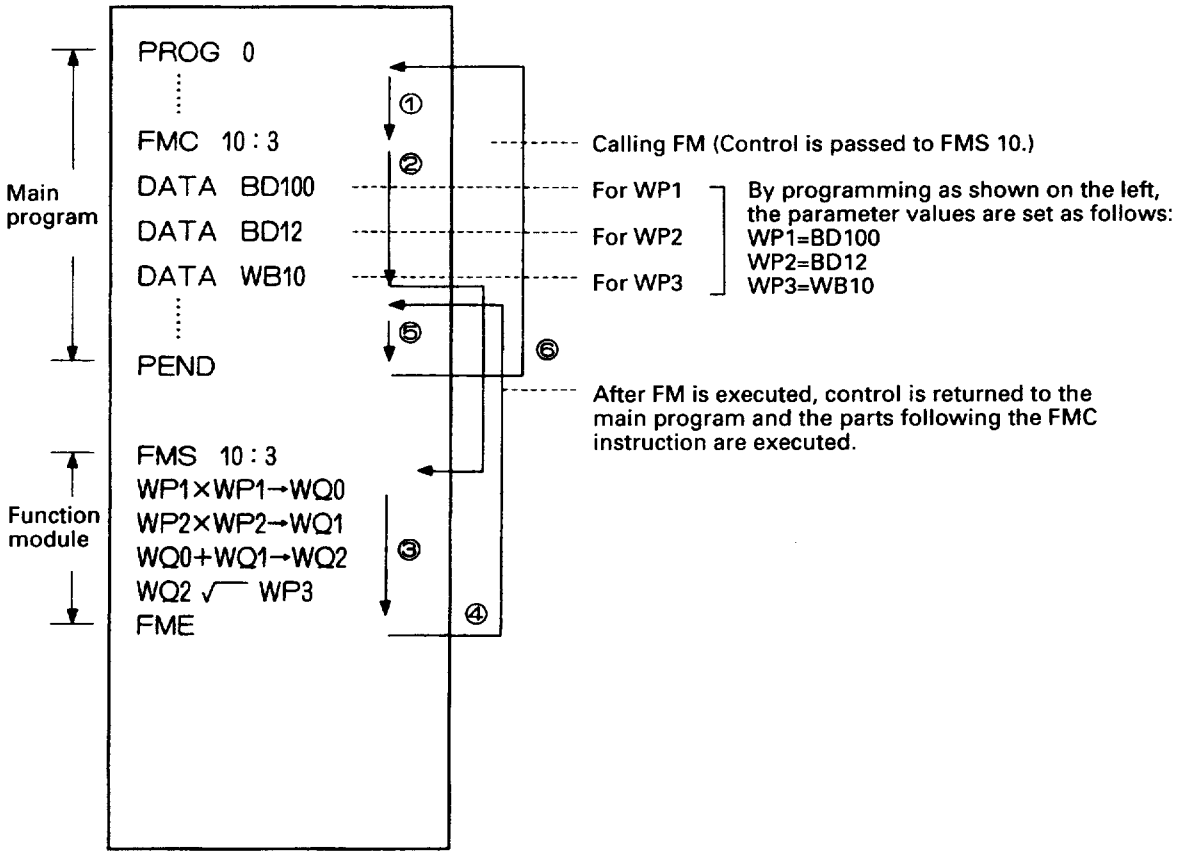
Key points

1. A function module begins with FMS n (n indicates the FM No.: 0 to 63 for the F120H, F70S and F120S Series, 0 to 255 for the F140S and F150S Series) and ends with FME.
2. A function module specified by a parameter can be called by any part of a program.
3. The caller can optionally specify a value for the parameter.
4. FMC can include a conditional contact to decide if the corresponding FM is to be executed.



When this contact is ON, FMS 10 is executed.
When this contact is OFF, program part ⑤ is executed after program part ① is executed.

Execution sequence of programs including FMC



3. Index registers (only for the F120H, F70S, F120S, F140S and F150S Series)

The two methods of specifying relay and word data addresses for the MICREX-F Series are as follows.

a. Use of fixed addresses (i.e., B0000 and WB0001)
b. Method of using variable *i* or *j* to specify an address as "*i* value + 10" or "*j* value + 20" (i.e., *i*0010 and *j*0020)

Method "a" is common to all MICREX-F Series devices. Method "b" is applicable to the F120H, F70S, F120S, F140S and F150S Series devices. Variables *i* and *j* are called index registers.

Index registers may be conveniently used for the following purposes.

(1) To execute a same pattern operation for different I/O areas
(FM or LOOP and CONT instructions must be used together.)
(2) To execute a repetitive operation (LOOP and CONT instructions must be used together.)

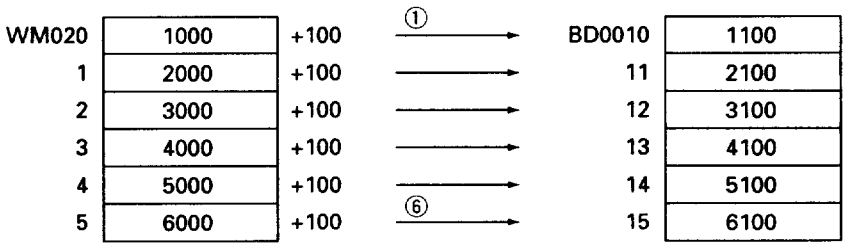
4. Loop control (LOOP and CONT) (only for F55, F70, F80H, F120, F70S, F120S, F140S and F150S series)

The LOOP and CONT instructions correspond to the FOR and NEXT statements of the BASIC programming language used for personal computers.

The program between the LOOP and CONT instructions is repeatedly executed the specified number of times during one scan.

Example of repetitive operation using index registers and LOOP and CONT instructions: *1

- The data in WM20 to WM25 is incremented by 100 (decimal number) and is sequentially stored in BD10 to BD15.



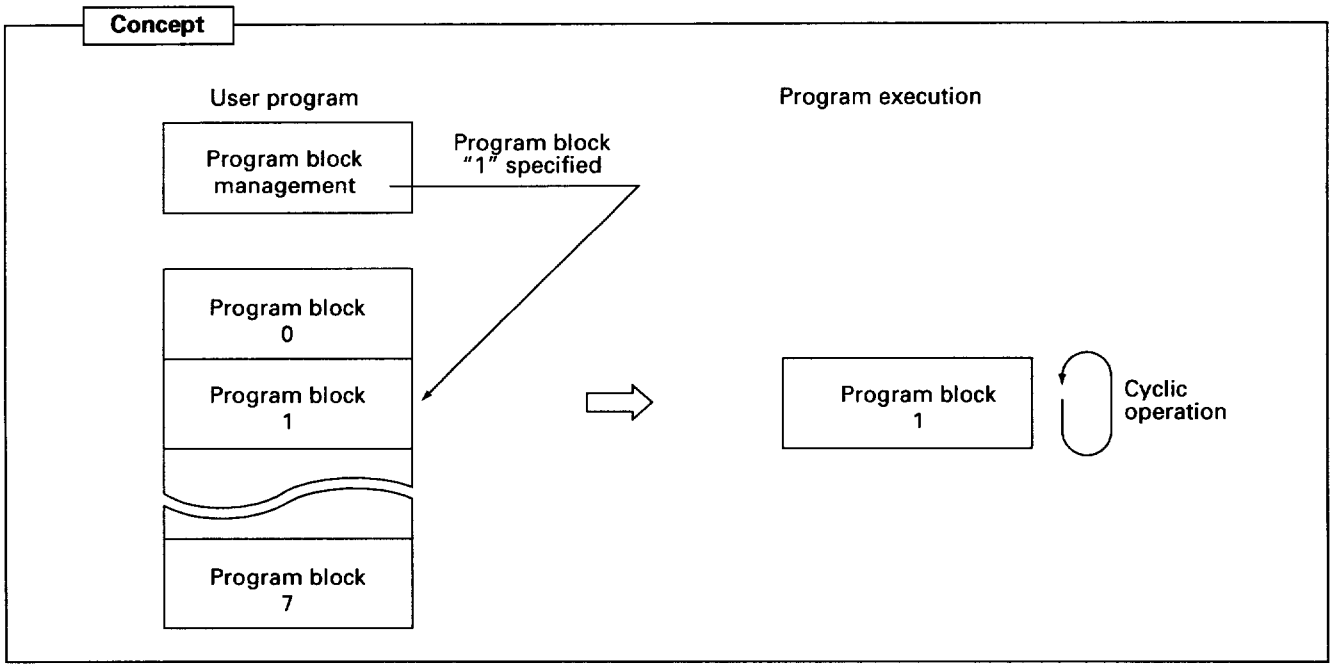

```
[ LEA  WM20:i      ] ..... WM20 data is stored in index register i.
[ LEA  BD10:j      ] ..... BD20 data is stored in index register j.
[ LOOP  4:d6       ] ..... LOOP4 is executed six times.
[ Wi0  +d100→Wj0   ] ..... Data at the address of "contents of i + 0" incremented by 100 and
                            stored at the address of "contents of j + 0"
[ i  IADD d1       ] ..... The contents of i are incremented by one and stored in register j.
[ j  IADD D1       ] ..... The contents of j are incremented by one and stored in register j.
[ CONT 4           ] ..... LOOP4 ends. If the repetition count is five or less, control is returned
                            to the step that follows the LOOP instruction.
```

Repeated six times

*1 Index register control is not available for the F55, F70 and F80H Series.

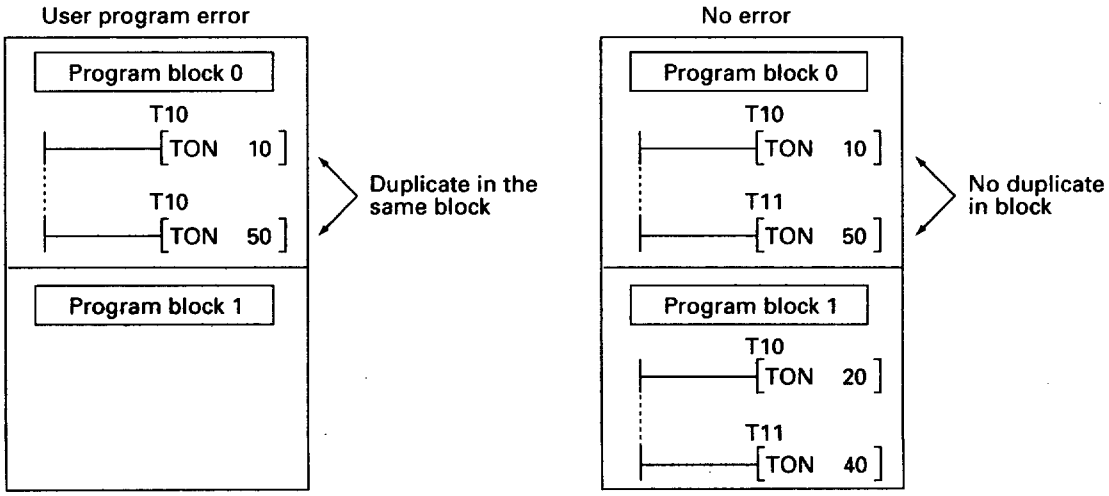
5. Program block processing

Machine tool change has conventionally required program transfer by the loader and ROM cassette replacement. Processing that executes only one of several programs as specified by the user program to save this trouble is called program block processing.



Each program block is assumed to be an independent program.

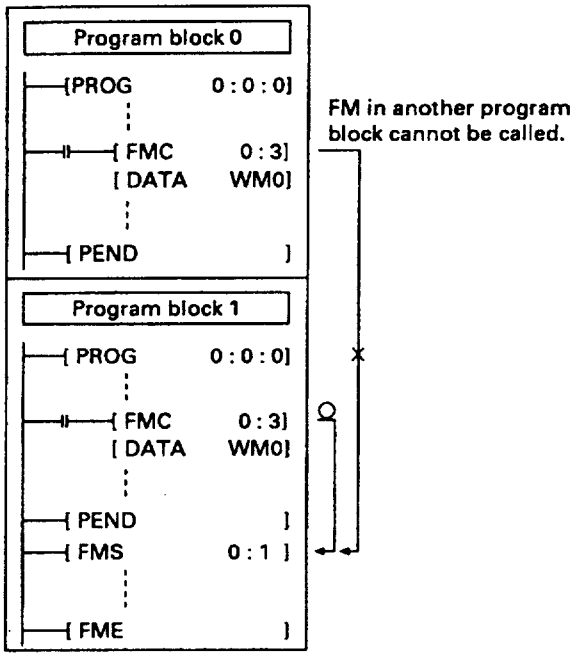
① A duplicated timer and coil in the same program block cause a user program error. A duplicated timer and coil in a different program block do not cause an error.



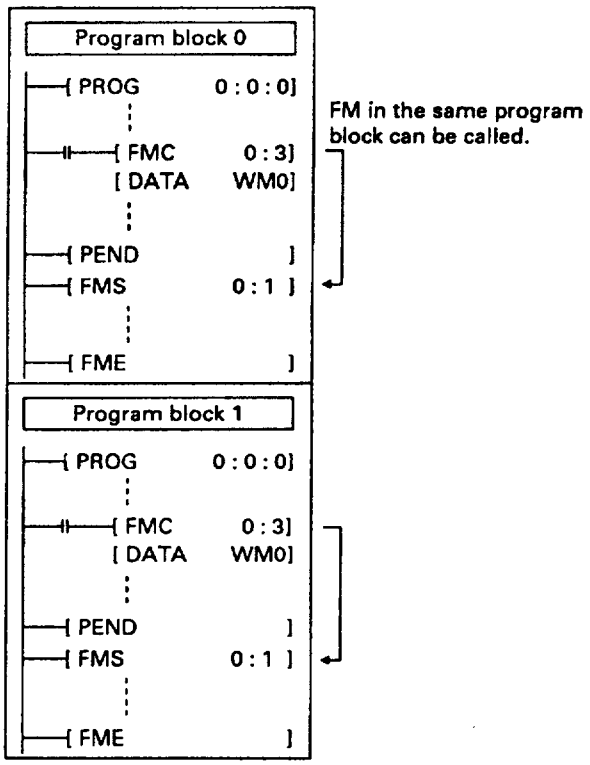
② Set fixed-cycle program (PROG50), interrupt program (PROG60 or larger), and function module (FM) for each necessary program block.

Example: Use function module (FM) 0 in program blocks 0 and 1.

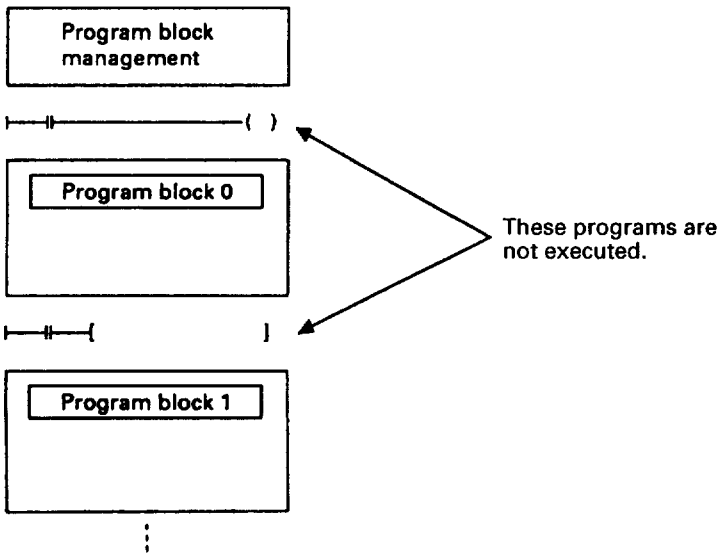
NG



OK



③ Instructions in other than the program block area are ignored.



④ The contents of the set system definition are common to program blocks.

⑤ When a program block being executed moves to another block, the block is started initially. (Scan flag 1 "50" is set ON)
Be careful to use the program block as a subroutine.

Program block description

① Program block management

```
[PROG 510:0000:0000 ] ← Means start of program block management.
[DATA       Z1      ] ← Specifies program block No.
[DATA       Z2      ] ← Specifies data memory initial operation mode.
[PROG 510:1000:0000 ] ← Means end of program block management.
```

Z_1 (program block No.)

The number of the program block to be executed is specified. The program block No. can be directly or indirectly specified in the range from 0 to 7.
If a program block No. that is outside the range or does not exit is specified, the following status occurs:

- For direct numeric value specification:
  F0018 (user program error) is set ON and the processor stops due to a fatal fault.
- For indirect specification:
  F002B (block specification error) is set ON and the processor stops due to a nonfatal fault.

Z_2 (data memory initial operation mode)

When the program block No. is changed in the program block management, the mode as to whether to retain (continue) or clear the memory contents is specified. The mode can be directly or indirectly specified.

Operation mode Z_2:

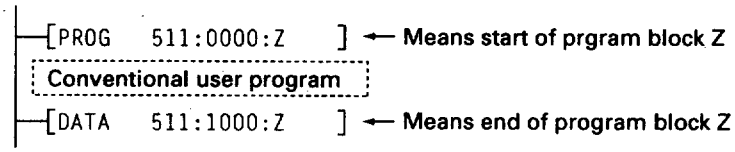
0: Memory contents are retained
1: Memory contents are cleared
Other: Memory contents are retained

For indirect specification, when the area to be cleared is specified in this mode, it is automatically changed to program block 0 after the program block is changed. (To use the area, reenter the block No. from the program in the changed program block.)

Z_2 (data memory initial operation mode)
The memory contents depend on the operation modes as follows. (The operation mode as to whether to hold or clear the preceding value area is selected.)

Data memory \ Operation mode		Operation mode 0	Operation mode 1
B area, I/O expansion area (W30 or larger) (RESET, HOLD specification)		Hold	
W24 (direct access) area (RESET, HOLD specification)		Hold	
P-link/PE-link transfer area (WL, W21 or larger)		Hold	
M, D, and A area		Hold	
F area		Hold	
K, S, W25, and W26 area		Hold	Clear
File	• X size change • Attribute (SI, DI, BD) change • Additional user file	Clear	
	• No X, Y, attribute change • Y size change • BD area change/no change	Hold (increased area: clear)	Clear
Timer (T) and counter (C)	• T/C set value area with set value	Set value	
	• Relay area of additional T/C instruction and current value area • Type change (TON to TOFF) Relay area of T/C instruction and current value area • No type change Relay area of non-cumulative T instruction and current value area	Clear	
	• No Type change Relay are of cumulative T instruction and current value area Relay area of C instruction and current value area • T/C set value area without set value • Unused T/C Relay area of unused T/C instruction, set value, and current value area	Hold	Clear

② Program block declaration



Z (program block No.)
The program block No. is directly specified in the range from 0000 to 0007.

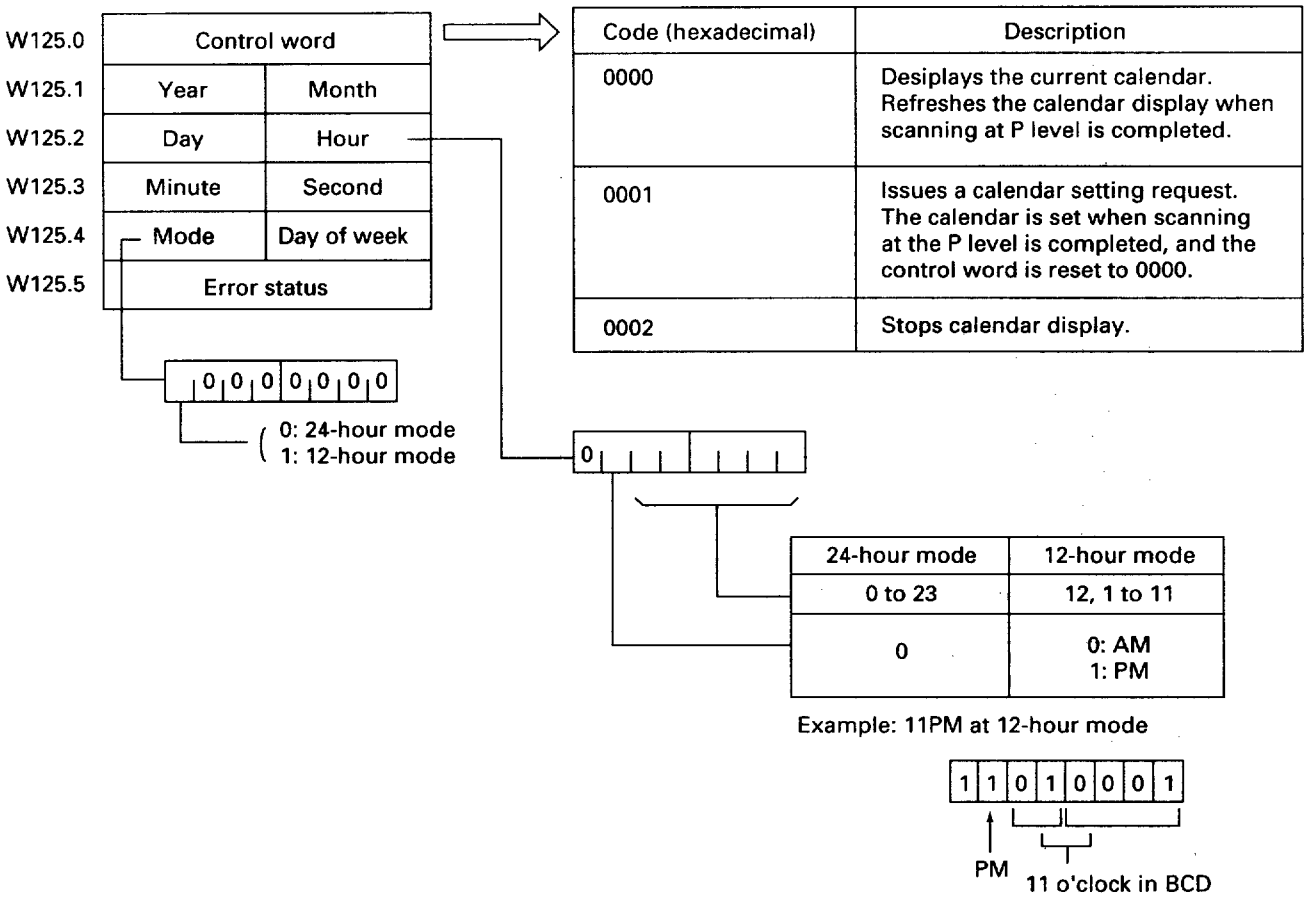
2-5 Calendar (F55, F70, F70S, F120S, F140S, F150S series only)

The calendar can be read and set by reading from or writing data to dedicated data module W125.

The calendar can be set in the range from 1989 to 2088. Leap year calculation is automatic.

1. Specification of data module W125

Module number:	W125
Attribute:	BCD 4 digits
Reset initial value:	0000 (HEX) for latest calendar display, control word, and error status
Year, month, day, hour, minute, second:	Displayed by BCD
Day of week:	0: Sunday, 1: Monday, 2: Tuesday, 3: Wednesday, 4: Thursday, 5: Friday, 6: Saturday



Code (hexadecimal)	Description
0000	Desiplays the current calendar. Refreshes the calendar display when scanning at P level is completed.
0001	Issues a calendar setting request. The calendar is set when scanning at the P level is completed, and the control word is reset to 0000.
0002	Stops calendar display.

24-hour mode	12-hour mode
0 to 23	12, 1 to 11
0	0: AM 1: PM

Example: 11PM at 12-hour mode

2. Setting calendar

■ Setting from user program

① Write data to be set for the calendar to area W125.1 or later.
② Write 0001 to W125.0 "control word."
Data is now set for calendar when scanning is completed.

■ Setting from program loader

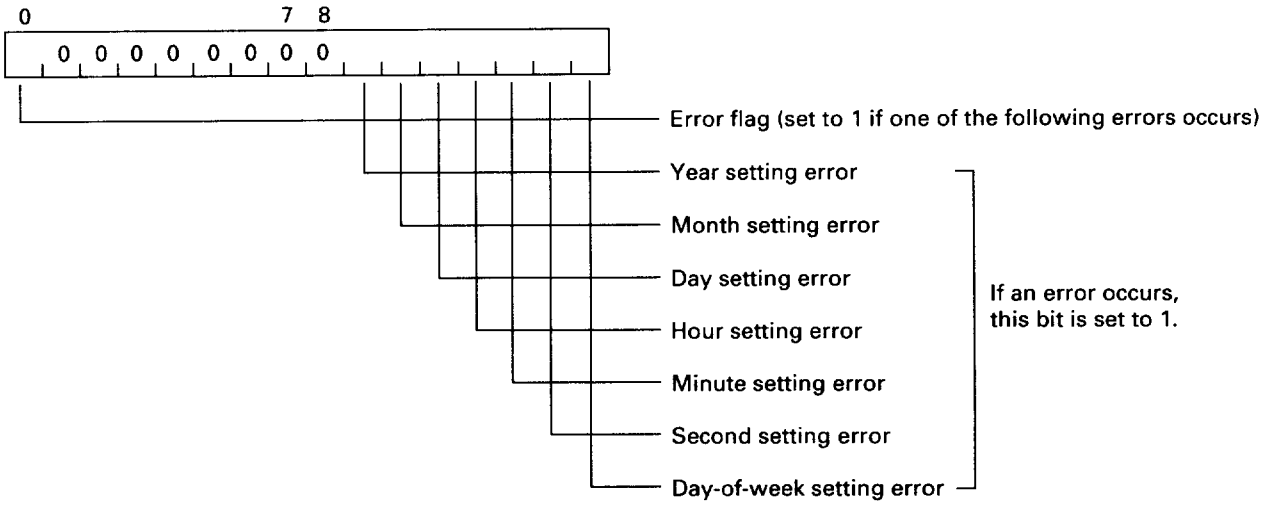
① Monitor data from W125.0
② Set 0002 in W125.0 "control word," and stop refreshing the calendar display.
③ Set data to be set for the calendar in area W125.1 or later.
④ Set 0001 in W125.0 "control word."
This sets the new data in the calendar within 100ms while PC stops or when scanning is completed while PC operates.

* Calendar can also be set by the clock setting in the auxiliary function.

3. Calendar accuracy

	Maximum daily difference		
Ambient temperature	0°C	25°C	55°C
POWER-OFF	3 seconds	1 second	4 seconds
POWER-ON	2 seconds	1 second	5 seconds

4. Error status



This error occurs due to the following causes:

① Non-BCD data is set.

② Data is set outside the setting range.

Month:	A value other than 1 to 12 is set.
Day:	A value other than 1 to 31 is set
Hour:	A value other than 0 to 23 is set in 24-hour mode. A value other than 1 to 12 is set in 12-hour mode.
Minute:	A value other than 0 to 59 is set.
Second:	A value other than 0 to 59 is set.
Day-of-week:	A value other than 0 to 6 is set.

When an error occurs, error detail is displayed on error status, the control word becomes 0002, and calendar data update stops.

The error is reset by writing normal data: the error status then becomes 0000.

Note: To use the calendar function in the F55 series, mount a T-link master interface card (NV1L-TL1) to the basic unit. The calendar function is not available unless the T-link master interface card is not mounted.

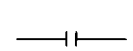
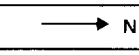
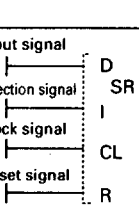
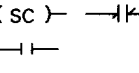
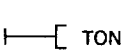
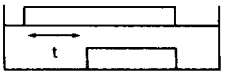
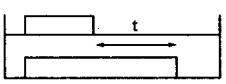
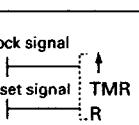
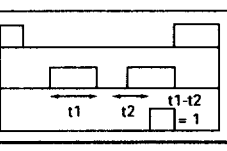
Section 3 Instructions

Section 3 Instructions

3-1 List of Instructions

This section explains the instructions that can be used for the MICREX-F series PCs and the numbers of steps of these instructions.

○ : Available — : Not available

Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Sequence	NO contact		Maximum number for a circuit Direction of column: 10 contacts + 1 coil Direction of line: 24 lines However, circuit returning is possible.	—	1	○	○	○	○	3-15
	NC contact			—	1	○	○	○	○	3-15
	Coil			—	1	○	○	○	○	3-15
	Returning			—	1 1	○	○	○	○	3-16
	Set			—	1	○	○	○	○	3-17
	Reset			—	1	○	○	○	○	3-17
	Rising edge differential			F1	1	○	○	○	○	3-19
	Falling edge differential			F2	1	○	○	○	○	3-19
	Invert			—	1	○	○	○	○	3-22
	MCS		Move of circuit buses (Master Control Set)	F6	1	—	○	○	○	3-23
	MCR		Return of the moved circuit buses (Master Control Reset)	F7	1	—	○	○	○	3-23
	Shift register		A shift register that has an arbitrary bit length of up to 511 bits. Input signals can be shifted in either direction (left or right) according to the direction signal.	F5	5	○	○	○	○	3-24
	Step control		Up to 100 groups of sequence control, having up to 100 steps, can be configured.	—	1	○	○	○	○	3-25
Timer	ON-delay timer			—	3	○	○	○	○	3-29
	OFF-delay timer			—	3	—	○	○	○	3-31
	Integrating timer			F3	4	—	○	○	○	3-33

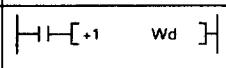
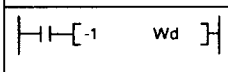
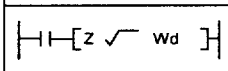
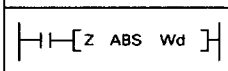
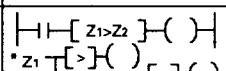
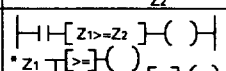
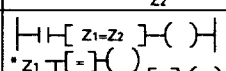
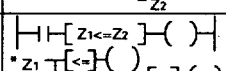
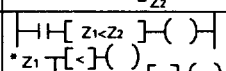
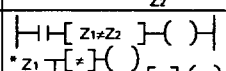
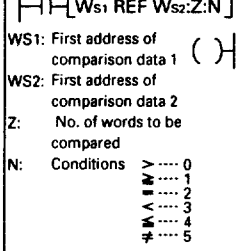
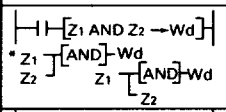
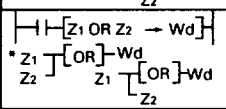
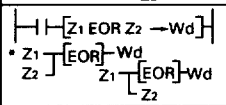
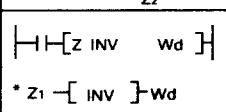
○ : Available — : Not available

Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Timer	Monostable timer	MON	Input signal / Output signal	—	3	—	○	○	○	3-35
	Monostable timer (Retriggable)	MR	Input signal / Output signal	—	3	—	○	○	○	3-37
Counter	Up counter	Count pulse / Reset signal / CT R	Reset signal / Count pulse / Count value / Output signal (Setting value)	—	4	○	○	○	○	3-40
	Down counter	Count pulse / Reset signal / CD R	Reset signal / Count pulse / Count value / Output signal (Setting value)	—	4	—	○	○	○	3-41
	Up/down counter	Add. pulse / Sub. pulse / Reset signal / UDCT R	Reset signal / Add. pulse / Sub. pulse / Count value / Output signal (Setting value)	F4	5	○	○	○	○	3-42
	Ring counter	Count signal / Reset signal / RCT R	Reset signal / Count pulse / Count value / Output signal	—	4	—	○	○	○	3-43
Arithmetic operation	Addition	[Z1+Z2 → Wd] *Z1 Z2 [+] Wd; Z1 [+] Wd Z2	Z2 is added to Z1 and the sum is stored in Wd.	F10	4	○	○	○	○	3-53
	Subtraction	[Z1-Z2 → Wd] *Z1 Z2 [−] Wd; Z1 [−] Wd Z2	Z2 is subtracted from Z1 and the remainder is stored in Wd.	F11	4	○	○	○	○	3-53
	Multiplication	[Z1xZ2 → Wd] *Z1 Z2 [x] Wd; Z1 [x] Wd Z2	Z1 is multiplied by Z2 and the product is stored in Wd.	F12	4	○	○	○	○	3-54
	Division	[Z1÷Z2 → Wd] *Z1 Z2 [÷] Wd; Z1 [÷] Wd Z2	Z1 is divided by Z2 and the quotient is stored in Wd. (Omit fractional values.)	F13	4	○	○	○	○	3-55
	Division remainder	[Z1 REM Z2 → Wd] *Z1 Z2 [REM] Wd; Z1 [REM] Wd Z2	Z1 is divided by Z2 and the remainder is stored in Wd.	F20	4	—	○	○	○	3-57
	Division (rounding to nearest whole number)	[Z1 DIVR Z2 Wd] *Z1 Z2 [DIVR] Wd; Z1 [DIVR] Wd Z2	Z1 is divided by Z2 and the quotient is stored in Wd. (rounding to nearest whole number)	F14	4	—	○	○	○	3-58
	Sign invert	[Z +/- Z → Wd] *Z [+/-] Wd	The sign (+/-) of Z is inverted and the result is stored in Wd.	F17	3	—	○	○	○	3-61

If the result exceeds the data range (7999 or ±79999999) of Wd, the overflow relay is set ON and the maximum (minimum) value is stored in Wd.

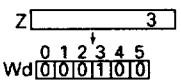
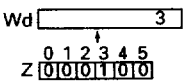
*Instruction writing using the block diagram language is supported by F120H, F70S, F120S, F140S, and F150S series.

○ : Available — : Not available

Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Arithmetic operation	Increment		Wd + 1 is stored in Wd.	F18	2	○	○	○	○	3-62
	Decrement		Wd - 1 is stored in Wd.	F19	2	○	○	○	○	3-63
	Root		The root of Z is obtained and the result is stored in Wd.	F15	3	—	—	○	○	3-59
	Absolute value		The absolute value is obtained and the result is stored in Wd.	F16	3	—	—	○	○	3-60
Comparison	>		Z_1 is compared with Z_2 and the following operation is executed depending on the result. When the conditions are satisfied, the output relay is set ON. Otherwise, the output relay remains OFF.	F30	4	○	○	○	○	3-67
	≧			F31	4	○	○	○	○	3-67
	=			F32	4	○	○	○	○	3-67
	≦			F33	4	○	○	○	○	3-68
	<			F34	4	○	○	○	○	3-68
	≠			F35	4	○	○	○	○	3-67
	File comparison		Z words having Ws_1 and Ws_2 as the first addresses, respectively, are compared and the output relay is set ON when the conditions are satisfied.	F36	6	—	—	○	○	3-70
Logical operation	AND		The AND of Z_1 and Z_2 is obtained and the result is stored in Wd.	F50	4	○	○	○	○	3-72
	OR		The OR of Z_1 and Z_2 is obtained and the result is stored in Wd.	F51	4	○	○	○	○	3-72
	Exclusive OR		The exclusive OR of Z_1 and Z_2 is obtained and the result is stored in Wd.	F52	4	○	○	○	○	3-74
	Invert		The logic of each bit of Z is inverted (from 0 to 1 or vice versa) and the result is stored in Wd.	F53	4	○	○	○	○	3-74

*Instruction writing using the block diagram language is supported by F120H, F70S, F120S, F140S, and F150S series.

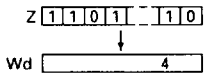
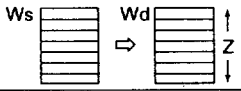
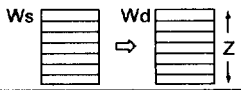
○ : Available — : Not available

Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Logical operation	Shift right logical	Ws SRL Wd:Z *Ws, Z: SRL: Wd Ws: SRL: Z, Wd Z: No. of shifted bits	The contents of Ws are shifted right by Z bits and are stored in Wd. 	F54	4	○	○	○	○	3-75
	Shift left logical	Ws SLL Wd:Z *Ws, Z: SLL: Wd Ws: SLL: Z, Wd Z: No. of shifted bits	The contents of Ws are shifted left by Z bits and are stored in Wd. 	F55	4	○	○	○	○	3-76
	Set bit	Ws SBIT Wd:Z *Ws, Z: SBIT: Wd Ws: SBIT: Z, Wd Z: No. of shifted bits	The Zth bit of Ws is set to 1 and is stored in Wd.	F56	4	—	—	○	○	3-78
	Reset bit	Ws RBIT Wd:Z *Ws, Z: RBIT: Wd Ws: RBIT: Z, Wd Z: No. of shifted bits	The Zth bit of Ws is set to 0 and is stored in Wd.	F57	4	—	—	○	○	3-79
	Test bit	Ws TBIT Wd:Z *Ws, Z: TBIT: Wd Ws: TBIT: Z, Wd Z: No. of shifted bits	The Zth bit of Ws is checked; if the data is 1, the output relay is set ON, if 0, it is set OFF.	F58	4	—	—	○	○	3-80
Conversion	Binary /BCD conversion	Z BCD Wd *Z: BCD: Wd	Binary data Z is converted into BCD and the result is stored in Wd.	F70	3	○	○	○	○	3-82
	BCD/ Binary conversion	Z BIN Wd *Z: BIN: Wd	BCD data Z is converted into binary and stored in Wd.	F71	3	○	○	○	○	3-83
	Character string	CHAR N Wd	Data which consists of N characters is transferred to the address indicated by Wd.	—	2	—	—	○	○	3-84
	ASCII/ numeric conversion	WS FIG Wd	Data in the address indicated by Ws is regarded as ASCII codes and is converted to numerical data and the result is stored in Wd.	F79	3	—	—	○	○	3-85
	Numeric/ ASCII conversion	Z ASCII Wd.N	Binary data Z is converted to ASCII codes and the result is stored in Wd.	F78	3	—	—	○	○	3-85
	Conversion to seconds	WS SEC Wd	Day, hour, minute, and seconds data at Ws is converted to seconds data and the result is stored in Wd.	F81	3	—	—	○	○	3-86
	Conversion from seconds	WS TIM Wd	Seconds data at Ws is converted to day, hour, minute, and seconds data and the result is stored in Wd.	F82	3	—	—	○	○	3-87
	Decode	Z DECO Wd *Z: DECO: Wd	The bit position indicated by Z is set to 1 and is stored in Wd. 	F72	3	○	○	○	○	3-88
	Encode	Z ENCO Wd *Z: ENCO: Wd	The most significant bit position (where 1 is set) is converted into a BCD number and is stored in Wd. 	F73	3	○	○	○	○	3-89

*Instruction writing using the block diagram language is supported by F120H, F70S, F120S, F140S, and F150S series.

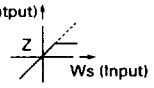
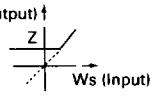
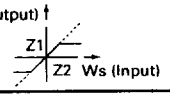
Section 3 Instructions

○ : Available — : Not available

Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Conversion	7-segment decode	Z 7SEG Wd *Z 7SEG Wd	Numerical data Z is converted into data for the 7-segment (a to g) display and is stored in Wd. Numeric value: 0,1,2,3,4,5,6,7,8,9 ,A,B,C,D,E,F 	F74	3	—	○	○	○	3-91
	Count ON-bit	Z BCNT Wd *Z BCNT Wd	The number of ON-bits "1" in Z is converted into a BCD number and is stored in Wd. 	F75	3	○	○	○	○	3-92
Trigonometric function	SIN	Z SIN Wd *Z SIN Wd	The sine of Z is calculated and the result is stored in Wd.	F90	3	—	—	—	○	3-124
	COS	Z COS Wd *Z COS Wd	The cosine of Z is calculated and the result is stored in Wd.	F91	3	—	—	—	○	3-124
	TAN	Z TAN Wd *Z TAN Wd	The tangent of Z is calculated and the result is stored in Wd.	F92	3	—	—	—	○	3-124
	SIN^{-1}	Z ASIN Wd Z ASIN Wd	The arcsine of Z is calculated and the result is stored in Wd.	F93	3	—	—	—	○	3-125
	COS^{-1}	Z ACOS Wd *Z ACOS Wd	The arccosine of Z is calculated and the result is stored in Wd.	F94	3	—	—	—	○	3-125
	TAN^{-1}	Z ATAN Wd *Z ATAN Wd	The arctangent of Z is calculated and the result is stored in Wd.	F95	3	—	—	—	○	3-125
Transfer	Data transfer	Z MOV Wd	Z is transferred to Wd.	F170	3	○	○	○	○	3-95
	Logical transfer	Z LMOV Wd *Z LMOV Wd	Z is transferred to Wd.	F180	3	○	○	○	○	3-97
	Data block transfer	Ws BT Wd:Z Z: No. of words transferred	Data in consecutive Z words is transferred. 	F171	4	○	○	○	○	3-98
	Logical block transfer	Ws LBT Wd:Z Z: No. of words transferred	Data in consecutive Z words is transferred. 	F181	4	○	○	○	○	3-99

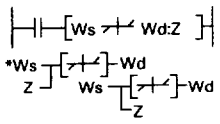
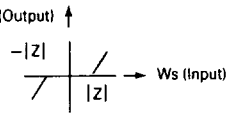
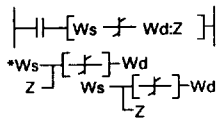
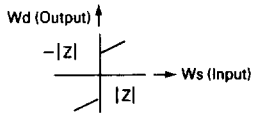
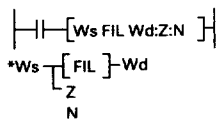
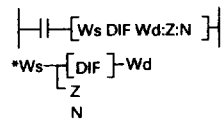
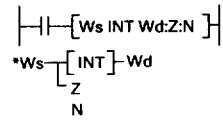
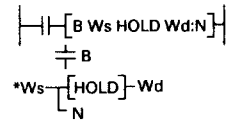
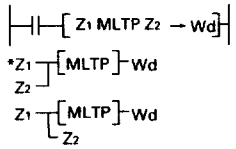
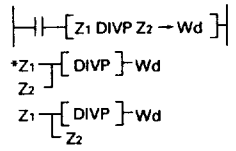
*Instruction writing using the block diagram language is supported by F120H, F70S, F120S, F140S, and F150S series.

○ : Available — : Not available

Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Transfer	Digit transfer	[Ws DT Wd:N1: N2:N3] N1: First bit position of origin N2: No. of bits to be transferred. N3: First bit position of destination	N2-bit data starting at N1th bit of data Z is transferred to the N3th bit and later of Wd register.	F172	5	○	○	○	○	3-100
	High-order digits transfer	[Z MOVU Wd]	Z (16-bit data) is transferred to the16 high-order bits in the 32-bit data area (Wd). The 16 high-order bits of the 32-bit data can be transferred to the 16-bit data area.	F173	3	—	○	○	○	3-101
	Low-order digits transfer	[Z MOVL Wd]	Z (16-bit data) is transferred to the16 low-order bits in the 32-bit data area (Wd). The 16 low-order bits of the 32-bit data can be transferred to the 16-bit data area.	F174	3	—	○	○	○	3-101
	Pattern clear	[Z1 PC Wd:Z2] Z1: Clear pattern Wd: Clear area Z2: Size	Z2 words of Wd are cleared by the Z1 pattern.	F175	4	—	—	○	○	3-103
	Search	[Z1 SRCH WS Wd:Z2]( ) Z1: Search data Ws: Search area Z2: Size Wd: Detected address	The same data as Z1 is searched for through the Z2 words of Ws and the result is output to the relay as follows. Then the detected address is stored in Wd. If present: The output relay is ON. If absent: The output relay is OFF. Detected address: 0 or larger	F176	6	—	—	○	○	3-104
	Switch	[Bs:Z1:Z2SW Wd] Bs *Z1 Z2 [SW] Wd Bs Z1 [SW] Wd Z2	The following transfer is done depending on the switching input state. Switching input ON: Z1 → Wd Switching input OFF: Z2 → Wd	F177	5	—	—	○	○	3-105
Message transmission	Message transmission	[MSGT,N1,N2,Z,Wd] [DATA Z]	Used to transmit data to equipment other than local equipment.	F182	4	—	—	○	○	3-106
	Message reception	[MSGR,N1,N2,Z,Wd] [DATA Z]	Used to receive data from equipment other than local equipment	F183	4	—	—	○	○	3-107
Analog	Upper limit	[Ws < Wd:Z] *Ws Z [<] Wd Ws [<] Wd Z	Sets the upper limit of the Ws value at Z and transfer to Wd. 	F110	4	—	○	○	○	3-112
	Lower limit	[Ws > Wd:Z] *Ws Z [>] Wd Ws [>] Wd Z	Sets the lower limit of the Ws value at Z and transfer to Wd. 	F111	4	—	○	○	○	3-113
	Upper and lower limit	[Ws ≮ Wd:Z] *Ws [≮] Wd Z1 Z2 Z1: Upper limit Z2: Lower limit	An upper limit (Z1) and a lower limit (Z2) are set for the value of Ws and a value within these limits is transferred to Wd. 	F112	5	—	—	—	○	3-114

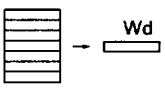
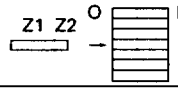
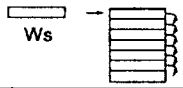
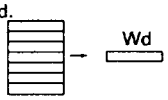
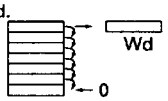
*Instruction writing using the block diagram language is supported by F120H, F70S, F120S, F140S, and F150S series.

○ : Available — : Not available

Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Analog	Dead band	 Z: Dead band width	|Z| is processed as a dead band width. If Ws > |Z|, Ws - |Z| → Wd. If Ws < -|Z|, Ws + |Z| → Wd. Otherwise, 0 → Wd. 	F113	4	—	—	—	○	3-115
	Bias	 Z: Bias value	|Z| is processed as a bias value. If Ws > 0, Ws + |Z| → Wd. If Ws < 0, Ws - |Z| → Wd. If Ws = 0, 0 → Wd. 	F114	4	—	—	—	○	3-116
	Filter	 N: Analog work area number Z: Time constant for filter/ΔT	Ws filtered and the result is output to Wd.	F115	5	—	—	—	○	3-117
	Differential	 Z: ΔT/differential time N: Analog work area number	Ws is differentiated and the result is output to Wd.	F116	5	—	—	—	○	3-118
	Integral	 Z: Integral time/ΔT N: Analog work area number	Ws is integrated and the result is output to Wd.	F117	5	—	—	—	○	3-119
	Sampling hold	 N: Analog work area number	While contact input is ON, Ws is being sampled. Just when contact input is OFF, Ws is held and output to Wd.	F118	5	—	—	—	○	3-120
	Multi-percent		Z1 is multiplied by Z2, the result is divided by 100, and the quotient is stored. The remainder is not saved.	F119	4	—	—	—	○	3-121
	Divide percent		Z1 is multiplied by 100, the result is divided by Z2, and the quotient is stored in Wd. The remainder is not saved.	F120	4	—	—	—	○	3-122

*Instruction writing using the block diagram language is supported by F120H, F70S, F120S, F140S, and F150S series.

○ : Available — : Not available

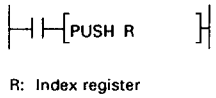
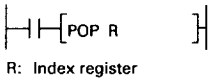
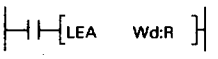
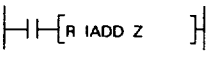
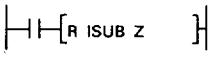
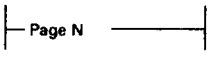
Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
File	File definition	File N1:N2:N3:X N1: File No. N2: X size N3: Y size X: Data type (BD, SI, DI)	The user file is registered and a file data area is reserved for the file data. The file must be defined before executing a file operation command.	F193	4	○	○	○	○	3-128
	Data table definition	File N1:N2:N3:X	The user file is registered and the initial data is defined. It becomes a read-only file.	F201	4	○	○	○	○	3-145
	Data	DATA	Data is set.	F235	SI:1 DI:2 BD:2	○	○	○	○	3-145
	End of data	DEND	Declares the end of data.	F202	1	○	○	○	○	3-145
	File clear	FLCL N	Resets the contents of file N to zero. FIFO and FILO files are initialized to contain no data. File N 	F194	1	○	○	○	○	3-138
	Selector	N:Z SEL Wd	The data of file N whose position is pointed by Z is transferred to Wd. File N O, Z → Wd 	F195	3	○	○	○	○	3-131
	Deselector	Z1:DSEL N:Z2	Z1 is transferred to the position in the file N pointed by Z2. Z1 Z2 → O File N 	F196	3	○	○	○	○	3-132
	File store	Z FFST N	Data of Ws is stored in file N, shifting the previous data in the file one by Ws → File N 	F190	2	—	○	○	○	3-134
	FIFO load	N FIFO Wd	The oldest data stored in file N by the FIFO store (FFST) instructions is transferred to Wd. 	F191	2	—	○	○	○	3-135
	FILO load	N FILO Wd	The latest data stored in file N by the FIFO store (FFST) instructions is transferred to Wd. 	F192	2	—	○	○	○	3-137
	File read	RFIL N1:Z1:Z2: N2 Wd N1: File No. Z1: X, Z2: No. of words Wd: Read address	The N2 words pointed by X and Y of the file No. N1 are transferred to the area beginning with first address Wd.	F197	5	—	—	○	○	3-139
	File write	Ws WFIL N1:Z2: Z2:N2 N1: File No. Z1: X, Z2: No. of words Ws: Write data address	The N2 words following the first address Ws are transferred to the area pointed by X and Y of file No. N1.	F198	5	—	—	○	○	3-140
	File information	N FINF Wd	Current file information of file No. N is stored in the storage address Wd.	F199	2	—	—	○	○	3-141

○ : Available — : Not available

Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Program declaration	Program entry	⊢[PROG N1:N2:N3]⊣ N1: Program No. N2: Fixed-cycle time N3: Start time delay	Declares initiation of program N1.	F230	3	—	—	○	○	3-149
	Program end	⊢[PEND]⊣	Declares the end of program.	F231	1	○	○	○	○	3-149
	FM call	⊢[FMC N1:N2]⊣ N1: FM No. N2: No. of parameters	Function module N1 is called.	F232	2	—	—	○ F120H series only	○	3-150
	FM start	⊢[FMS N1:N2]⊣ N1: FM No. N2: No. of work areas	Declares initiation of function module N1.	F233	2	—	—	○ F120H series only	○	3-151
	FM end	⊢[FME]⊣	Declares the end of function module.	F234	1	—	—	○ F120H series only	○	3-151
	Skip	⊢[SKIP N]⊣ N: Skip No.	All instructions placed between SKIP and SEND instructions are treated as NOP.	F250	1	—	—	○	○	3-155
	Skip end	⊢[SEND N]⊣		F251	1	—	—	○	○	3-155
	Disabled interrupt	⊢[DI Z]⊣	Interrupt by the task of interrupt level Z is disabled.	F316	2	—	—	○	○	3-153
	Enabled interrupt	⊢[EI Z]⊣	Interrupt by the task of interrupt level Z is enabled.	F317	2	—	—	○	○	3-154
	Jump	⊢[JMP N]⊣ N: Jump No.	All instructions placed between JUMP and JEND instruction are jumped and not executed. ⊢[JUMP 12]⊣ Program ⊢[JEND 12]⊣	F253	1	○	○	○	○	3-155
	Jump end	⊢[JEND N]⊣		F254	1	○	○	○	○	3-155
	Loop	⊢[LOOP N:Z]⊣ N: Loop No. Z: Repeat count	Both conditional and unconditional loop are possible.	F210	2	—	—	○	○	3-156
	Continue	⊢[CONT N]⊣		F211	1	—	—	○	○	3-156

Index register control, Page

○ : Available — : Not available

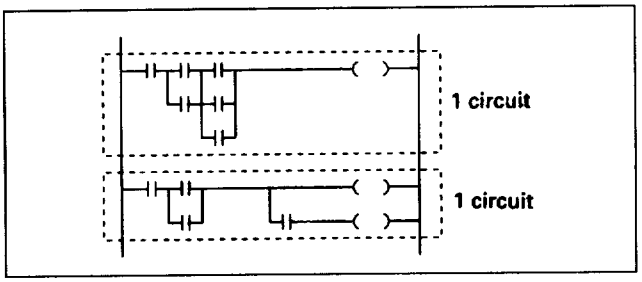
Classification	Name	Symbol	Function	-(F)- No. of D05, D10S	No. of steps	F30 F50 F50H	F60	F55 F70 F80H F120H	F70S F120S F140S F150S	See page
Index register control	Push	 R: Index register *i,j,k,l,m*	The index register pointed to by R is saved.	F310	1	—	—	F120H series only	○	3-157
	Pop	 R: Index register *i,j,k,l,m*	The index register pointed by R is restored.	F311	1	—	—	F120H series only	○	3-157
	Load effective address	 R: Index register Ws: *i,j,k,l,m* An operand by word or bit address is possible.	The execution address pointed to by operand Ws is stored in the index register.	F312	2	—	—	F120H series only	○	3-160
	Index register addition	 R: Index register *i,j,k,l,m*	R + Z → R	F313	2	—	—	F120H series only	○	3-163
	Index register subtraction	 R: Index register *i,j,k,l,m*	R - Z → R	F314	2	—	—	F120H series only	○	3-163
Page	Page	 N: 1 to 9999		F325	2	○	○	○	○	3-167
Others	Pass	———	Indicates the pass of program.	—	1	○	○	○	○	—
	Blank	----------	Indicates the blank of program.	—	1	○	○	○	○	—

3-2 Rules on Programming

3-2-1 Rules on programming using ladder diagram

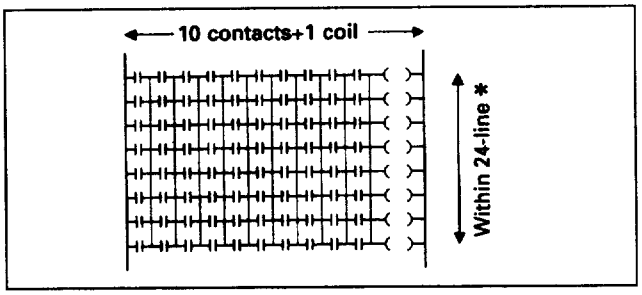
1. Units of circuit

A circuit comprises a group of contacts and coils.



2. No. of contacts and coils in a circuit

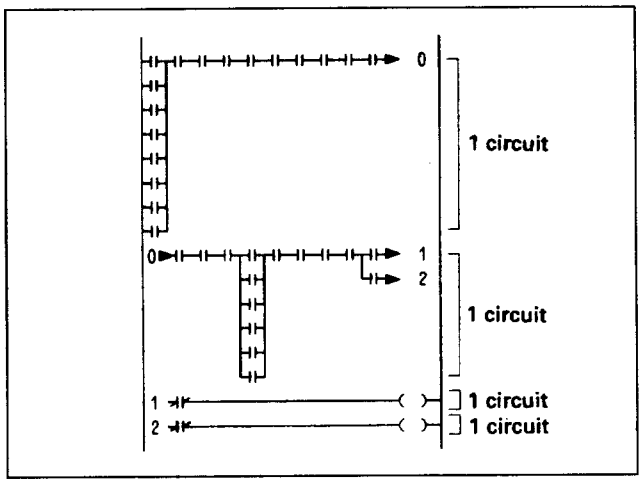
Maximum No. of contacts and coils which can be drawn in a circuit is as follows.



*Within 8 lines for a program loader other than LITE

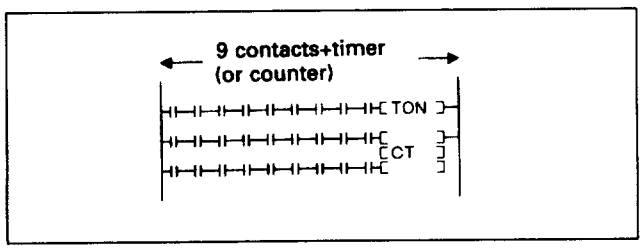
3. Returning of circuit

When drawing over 11 contacts in a line, programming can be carried out as follows.



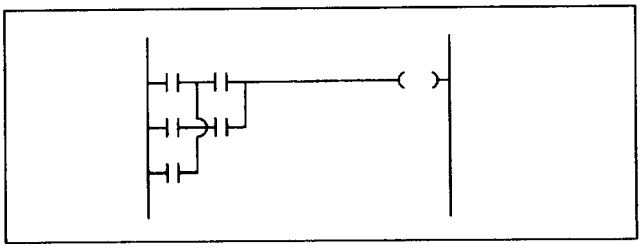
4. Timer and counter

When using a timer or counter, the number of contacts per line must not exceed 9.



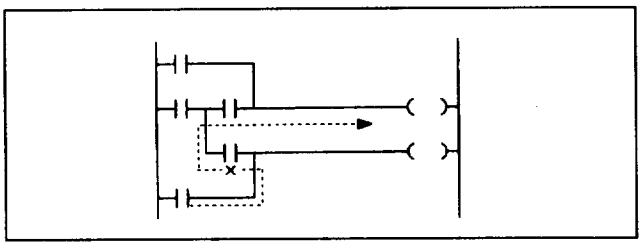
5. Crossing

Connecting lines can be drawn so that they cross each other as shown in the diagram below.



6. Sneak-circuit

Since the current flows only from left to right across the contact, the current does not flow as indicated by the dotted line in the diagram below.

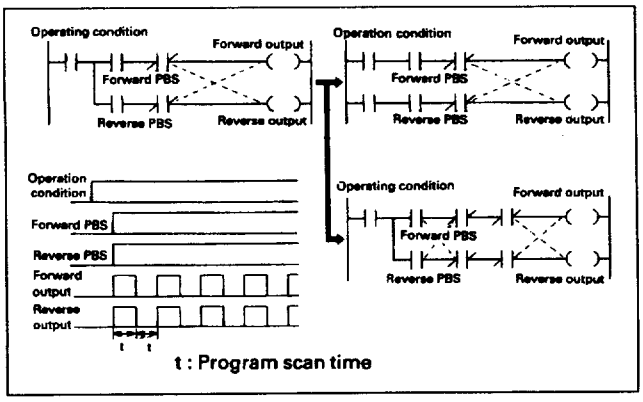


7. Execution operation in a ladder diagram

In the ladder diagram, the execution operation is carried out consecutively from the first ladder circuit. The execution operation in a circuit is carried out in a parallel operation as in the circuit with a magnetic control relay. However, a serial operation is carried out between Nos. "n" and "n+1" circuits.

8. Interlock

The execution operation in a circuit is carried out in a parallel operation. Therefore, in an interlock circuit as shown in the diagram below, when PBS's for Forward and Reverse are pushed simultaneously within one program scan time, the output synchronizes with the program scan and ON/OFF operations are repeated. In this case, to ensure interlock, it is necessary for the circuit to be divided into two, or for a NC contact to be inserted as shown in the diagram below.



3-2-2 Calculating No. of steps

The number of program steps can be calculated as the total number of steps for all instructions.

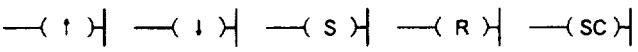
Notes on calculation

1. The ladder diagram is expressed with a combination of symbols shown in the table below. A symbol stands for a step for the user program.

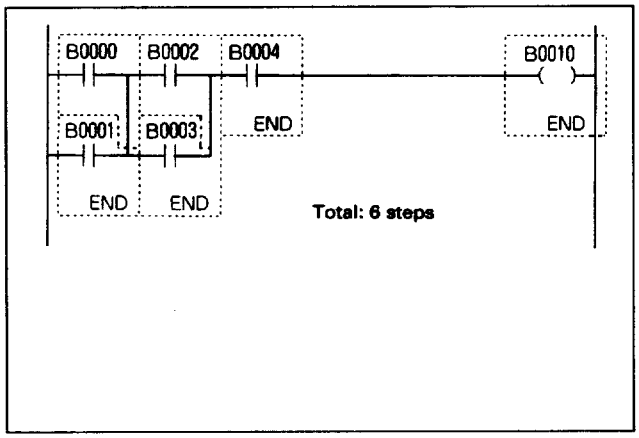
	NO contact *1	NC contact	Invert	Pass *1	Blank *1	Coil *2
Basic instruction				×N	×N	
Vertical connection				×N	×N	
Crossing				×N	×N	
Column end	END	END	END	×N END	×N END	END

Note: *1 In case same symbols continue longitudinally, they are described by an instruction.
x N in the table denotes No. of lines or No. of columns.

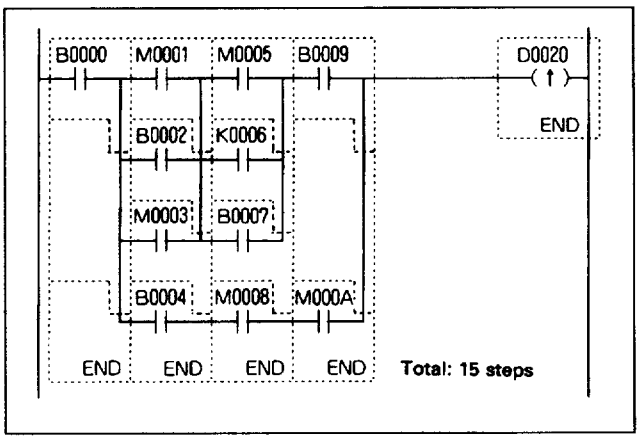
*2 Coil symbols include:



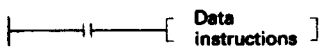
Example 1



Example 2



2. Data instructions

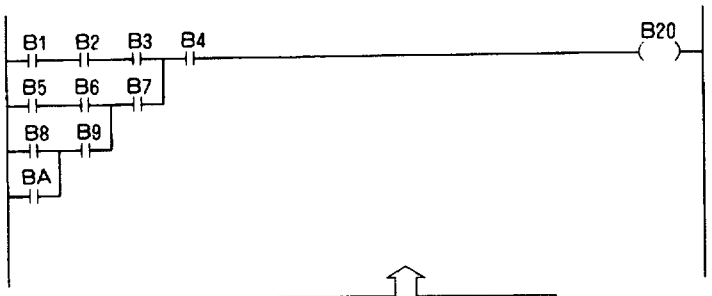


See Section 3-1 for the number of steps for the data instructions.

ONE-POINT ADVICE **Reducing number of program steps**

Relay sequences and steps

The number of steps used by user's programs differs depending on symbol positions even on sequence diagrams that function identically.

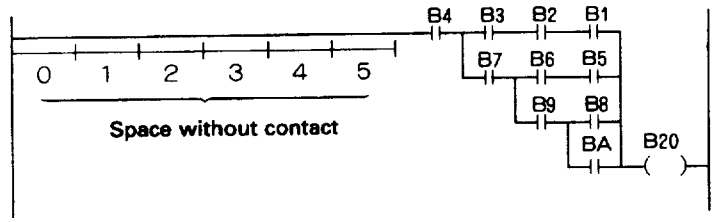


Although identical in terms of logics, the No. of steps largely varies.

11 steps

Program

Step	Command	Step	Command
0	B1	6	B9 END
1	B5	7	B3
2	B8	8	B7 END
3	BA END	9	B4 END
4	B2	10	B20
5	B6		



21 steps

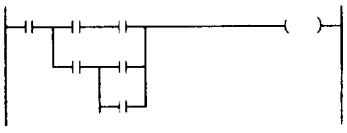
Program

Step	Command	Step	Command
0	END	11	B2
1	END	12	B6
2	END	13	B9
3	END	14	END
4	END	15	B1
5	END	16	B5
6	B4	17	B8
7	END	18	BA END
8	B3	19	Blank END
9	B7	20	B20
10	END		

Key points for reducing No. of program steps

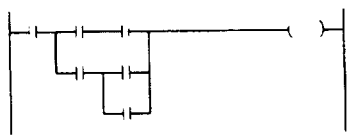
- Collect contact symbols at the left-hand side.
- Decrease No. of lines for contacts along with shift to the right-hand side.

CORRECT

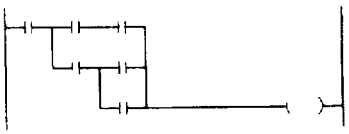


INCORRECT

- Arrange the output coil on the upper side.



CORRECT

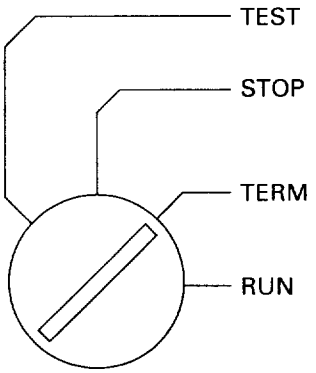


INCORRECT

3-2-3 Considerations in programming

The F80H, F120H, F70S, F120S, F140S and F150S series processors have a mode selection key switch on their front. The program loader LITE and the program loader software can be operated in either whole programming mode or local programming mode. When the program loader is in whole programming mode, programs cannot be written or modified during processor operation. When it is in the local programming mode, programs can be written or modified during processor operation. Select a mode according to the purpose.

1. Mode selection key switch



Legend	Mode	Function
RUN	Operation mode	• Mode in which the processor is in operation • The processor does not accept a stop command or program write from the program loader.
TERM	Terminal mode	• The processor accepts an operation or stop command or program write from the program loader.
STOP	Stop mode	• Mode in which the processor is in the stopped state • The processor does not accept a start command or program write from the program loader.
TEST	Test mode	• Mode used for debugging • Various kind of operations can be performed from the program loader.

2. Modes for online programming (direct write to or modification in processor)

Whether a program can be written or modified or monitoring can be performed depends on the combination of the mode set by the processor key switch and the programming mode of the LITE or program loader software. The possible combinations are shown below.

① Programming (○: possible)

Program loader / Processor		Programming mode: Whole	Programming mode: Local
Mode set by key switch	RUN	—	—
	TERM	○	○*
	STOP	—	—
	TEST	○	○*
Processor state		Stop or stop	Operation

* To store a program when the processor is in operation, press the ENT key twice.

② Monitoring (○: possible)

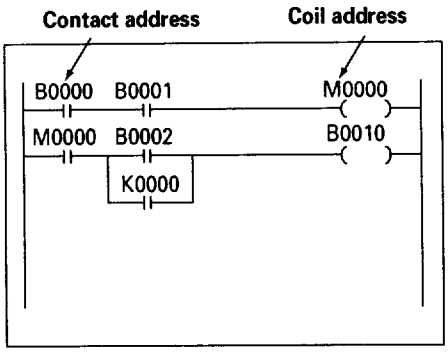
Program loader / Processor		Programming mode: Whole	Programming mode: Local
Mode set by key switch	RUN	○	○
	TERM	○	○
	STOP	○	○
	TEST	○	○
Processor state		Operation or stop	

3-3 Sequence instructions

3-3-1 Contact and coil

1. NO contact(⊣ ⊢), NC contact (⊣/⊢), Coil(⊣( )⊢)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

Instruction	NO contact, NC contact, Coil	Description	Example
Symbol	⊣ ⊢ NO Contact ⊣/⊢ NC contact ⊣( )⊢ Coil	① Restrictions on one circuit • Direction of columns: 10 contacts + 1 coil • Direction of lines: 24 lines (8 lines for F30, F50, F50H, F60, D05, D10S, D20) ② When the return instruction is used in a circuit, the restriction in ① do not apply.	
Function	① NO contact, NC contact and output coil are indicated. Combination of a symbol, identifier, and address works as a contact or coil. ② Any numbers of contact that have the same address can be specified.		

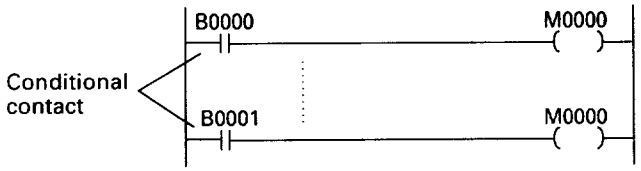
Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
⊣⊢, ⊣/⊢	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	S	Z	E	O
⊣( )⊢	○*	○	○	—	—	○	—	—	—	○**	○	○	○	—	—	○	○	—	—	↑	—

* B is not usable for input address.
** When using P or PE-link, L is also an effective identifier.

Notes on programming

Note on duplicated coil



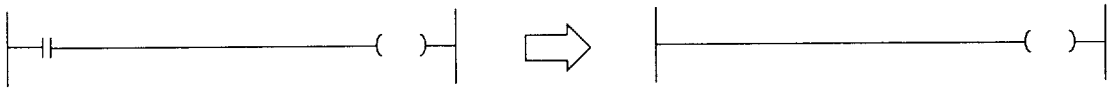
When a duplicated coil (more than one coil instructions for one and the same address) is programmed as shown on the left, the operation is as shown below.

B0000	B0001	M0000
ON	ON	ON
ON	OFF	OFF
OFF	ON	ON
OFF	OFF	OFF

A duplicated coil should be avoided though it may not cause a program error.

Merit of coil instruction

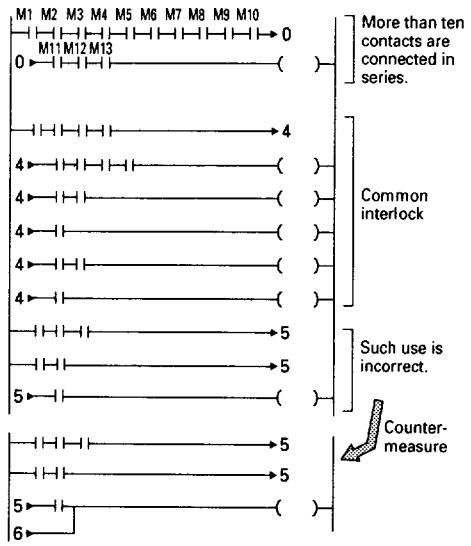
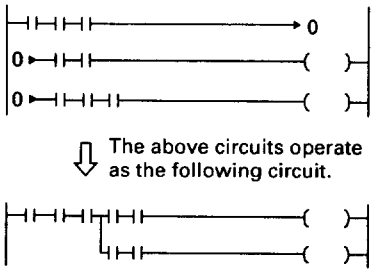
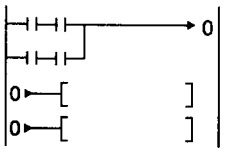
In MICREX-F series, the coil instruction can be used without using the conditional contact. Writing coil instruction which is always ON helps making a program simpler and reducing the program steps.



Conditional contact is not required if the circuit is always in execution.

2. Returning (⟶ N N ⟵)

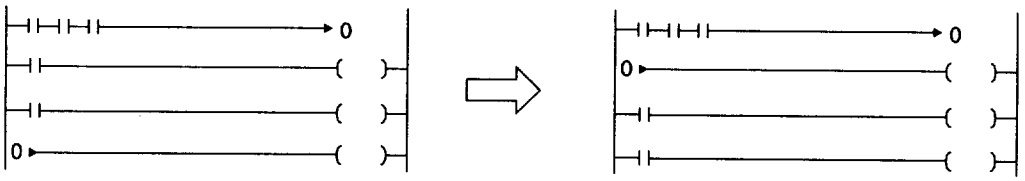
Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Returning	Description	Example
Symbol	⟶ N N ⟵ Return origin / Return destination (N: 0 to 9) (N: 0 to 23 for F55, F70, F80H, F120H, F70S, F120S, F140S, F150S	① The return origin and destination having the same number are connected. The use count of the return destination number is not restricted. ② Return numbers remain valid on different pages. ③ A return number can be used repeatedly any number of times. If the number reaches 9, subsequent numbers from 0 are used. ④ The returning instruction can also be used as a condition signal for a data instruction.	
Function	① This instruction transfers the conductive or nonconductive status of a circuit to the next circuit. PC operates on the assumption that the return origin and return destination having the same number are connected. (Example) 		

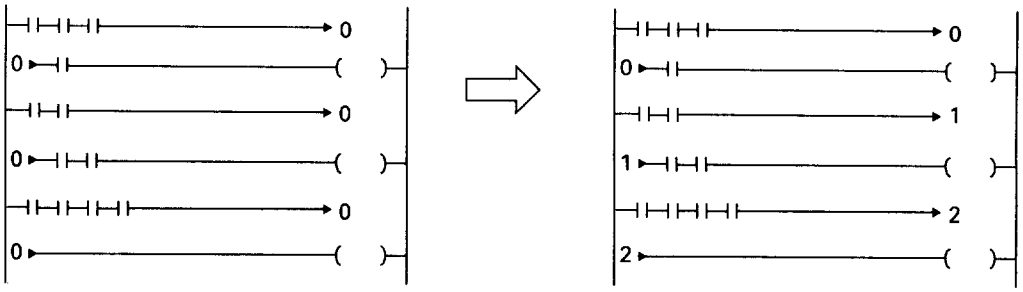
Note on programming

The returning instruction can be used for skipping circuits or repeating the same return No. only (see the line diagram on the left below). However, it is recommended to use the circuit shown on the right side as much as possible from the viewpoint of creating a more understandable program.

Example of skipping circuits

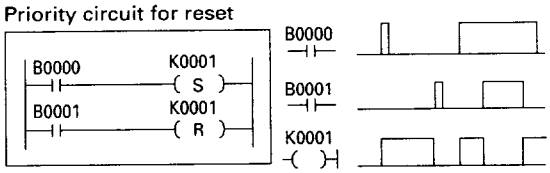
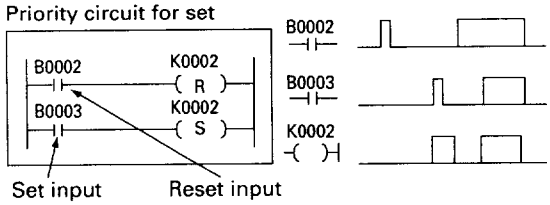


Example of repeating the same No.



3. Set (—(S)—), Reset (—(R)—)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Set, Reset	Description	Example
Symbol	—(S)— Set —(R)— Reset	① The same address must be assigned to the set coil and the reset coil as a pair. ② Any numbers of set coils and reset coils that have the same address can be specified in the program.	Priority circuit for reset  Priority circuit for set  When both the set input and the reset input are ON, the instruction which was placed later is effective.
Function	These instructions change the exciting or nonexciting states of coils and retain the changed states. When the set input becomes ON, the set coil is turned ON. Even when the set input has gone OFF, the ON state of the coil is retained. When the reset input becomes ON, the set coil is turned OFF.		

Effective identifier

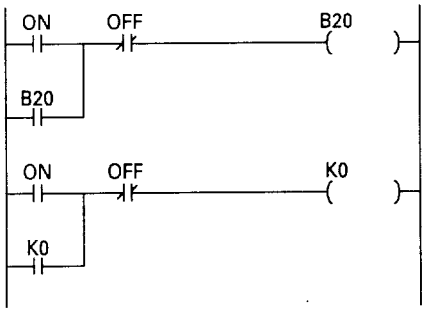
	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
—(S)—	O*	O	O	—	—	O	—	—	—	O**	O	O	O	—	—	O	O	S	Z	E	O
—(R)—	O*	O	O	—	—	O	—	—	—	O**	O	O	O	—	—	O	O	—	—	↑	—

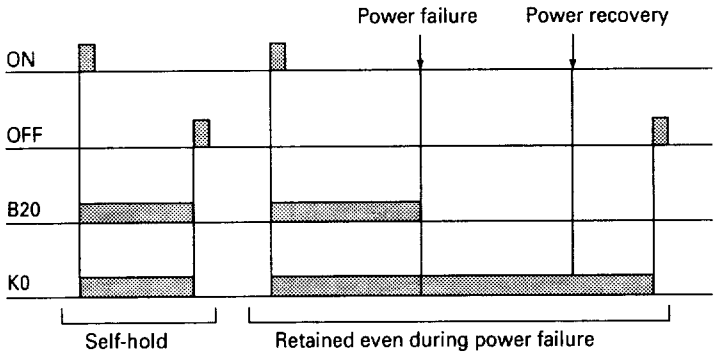
* B is not usable for input address.
** When using P or PE-link, L is also an effective identifier.

Difference between area B and area K and the set and reset operations for these areas

1. Difference between I/O relay (B) and keep relay (K)

Because the following sequences both incorporate a self-hold circuit, they provide the same operation. However, if power failure occurred while the output was ON, these circuits have different output states when power is restored.

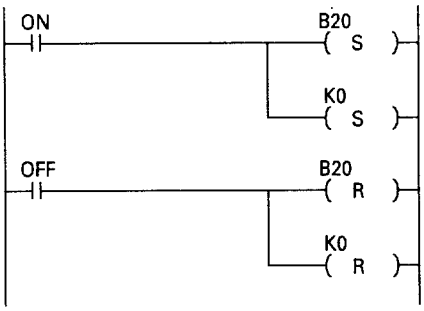


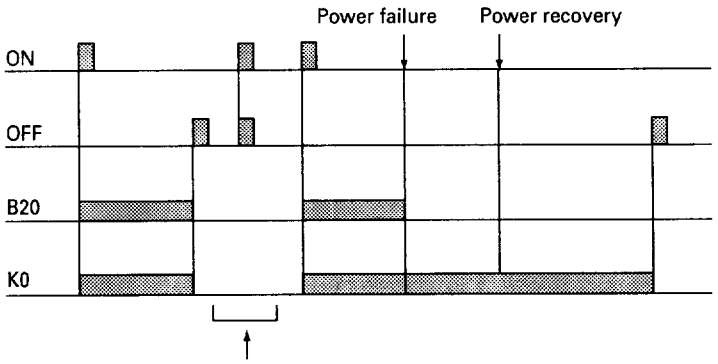


2. Difference between I/O relay (B) and keep relay (K) in operations using set and reset instructions

Because the set and reset instructions incorporate a self-hold function, the output state that is set ON is retained until "OFF" signal is given.

Note that the I/O relay (B) area and keep relay (K) area differ in operation when power is recovered.





Because a reset circuit is placed after the set circuit in the program, both K0 and B20 are reset.

4. Rising edge differential (-(↑)-), Falling edge differential (-(↓)-)

Processor (▭: Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

Instruction	Rising edge differential Falling edge differential	F1, F2	Description	Example
Symbol	-(↑)- -(↓)-		① Rising edge differential Input signal ↑ → ON, One scan The preceding value of differential relay is OFF ② Falling edge differential Input signal ↓ → ON, One scan The preceding value of differential relay is ON.	B0000 — D0001 -(↑)- B0001 — D0001 -(↓)- B0000 D0000 — One scan B0001 D0001 — One scan
Function	① When the preceding value of the differential relay is OFF, the rising edge differential relay is set ON for one scan at the rising edge of the input signal. ② When the preceding value of the differential relay is ON, the falling edge differential relay is set ON for one scan at the falling edge of the input signal.			

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
-(↑)-	—	—	—	○	—	—	—	—	—	—	○	○	○	—	—	○	—	S	Z	E	O
-(↓)-	—	—	—	○	—	—	—	—	—	—	○	○	○	—	—	○	—	—	—	↑	—

ONE-POINT ADVICE **Notes on power recovery**

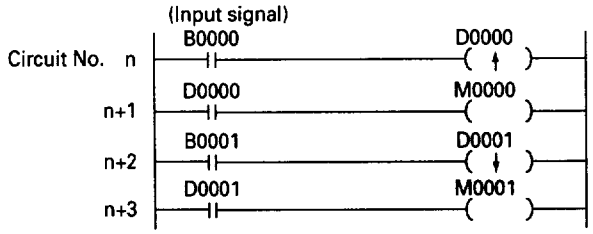
Rising edge differential and falling edge differential

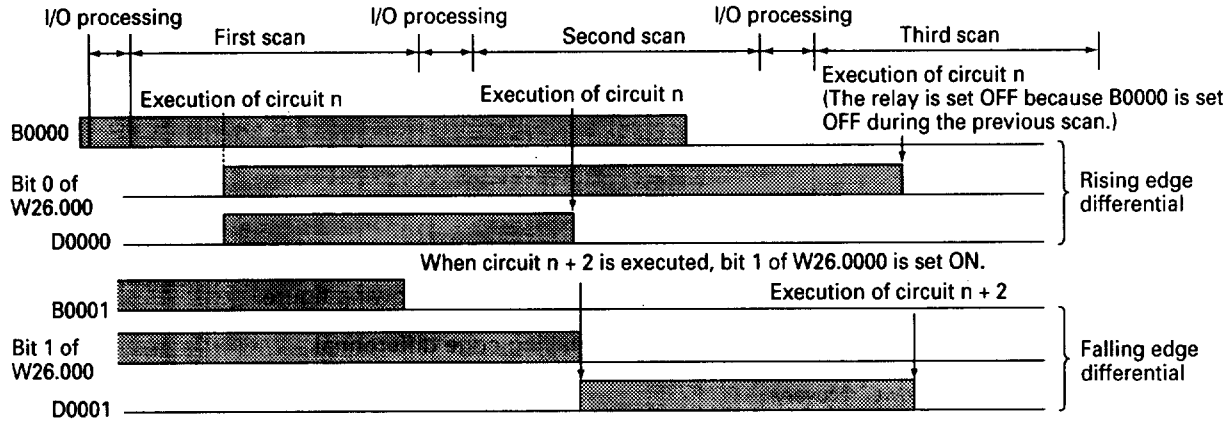
These relays detect the rising or falling edges of input signals and are set ON for only one scan after the current program step is executed. In this way, they are used as rising edge differential relays (⊣↑⊢) or falling edge differential relays (⊣↓⊢).

Notes on use

1. A rising edge differential relay is set ON only for one scan from the rising edge of an input signal when the preceding value, which is stored at the address corresponding to the preceding differential relay value area (W26), is OFF.

2. A falling edge differential relay is set ON for only one scan from the falling edge of an input signal when the preceding value, which is stored at the address corresponding to the preceding differential relay value area (W26), is ON.

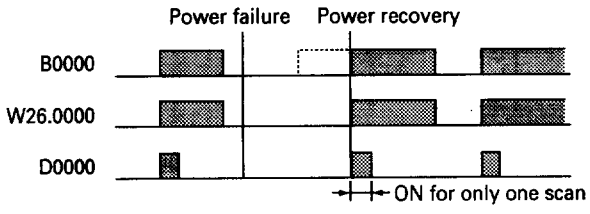




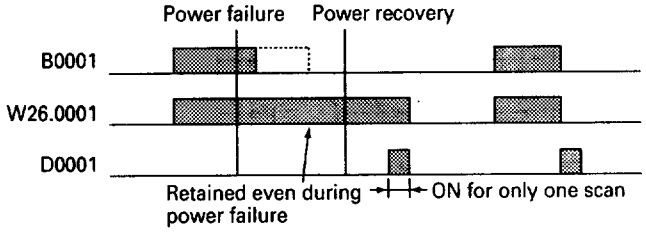
3. A differential relay stores the differential input value set during the preceding scan. Therefore, the differential relay compares the preceding value with the current input signal when a differential instruction is executed and is set ON for one scan if the comparison result does not match. The preceding value is updated immediately after the differential relay is set ON. Consequently, the differential relay cannot be set ON for more than two continuous scans.

4. Because a differential relay area is volatile, its contents are cleared when power is turned OFF. However, the preceding value area is nonvolatile. If power failure occurs while the input signal is ON (i.e., external input of B0000 is ON) and power is recovered while the input signal is OFF, a falling edge differential relay is set ON for the first scan.
If this causes a problem in operation, the processing program described on the following page must be added.

For rising edge differential



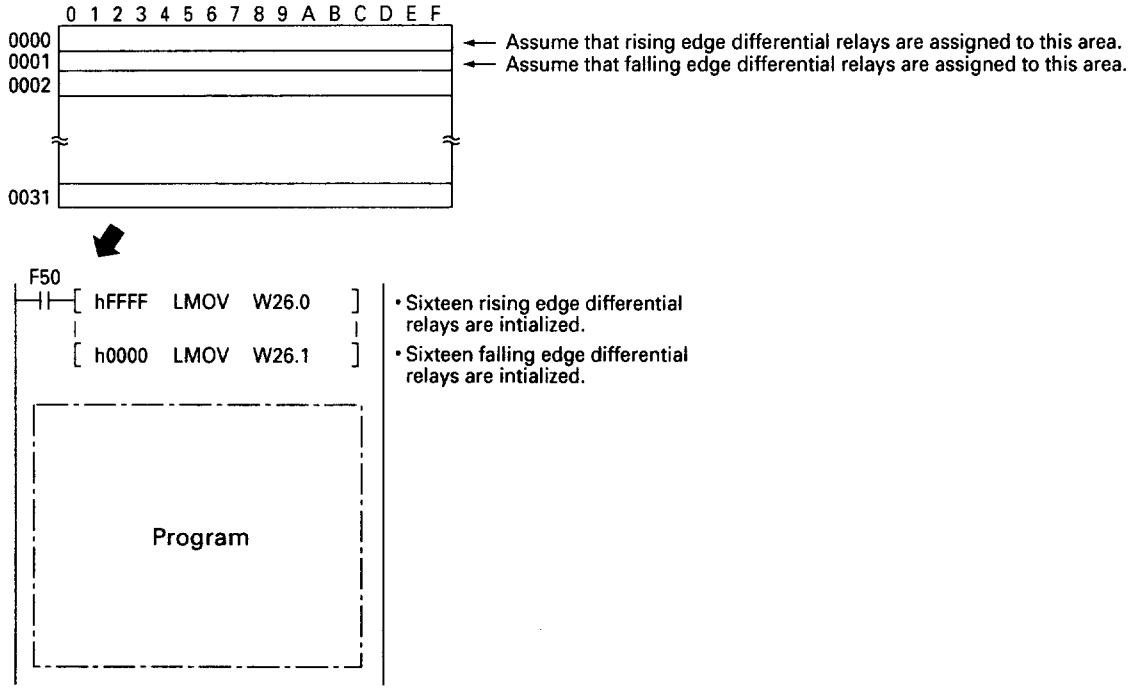
For falling edge differential



Rising edge differential and falling edge differential (continued)

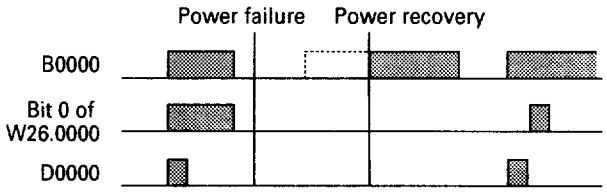
Countermeasure for Item 4. on preceding page

When designing a program, separate the areas for rising differential relays from the areas for falling differential relays in units of words. The place the following program (for initialization of differential) ahead of the program to be designed. In this way, the relays can be initialized.

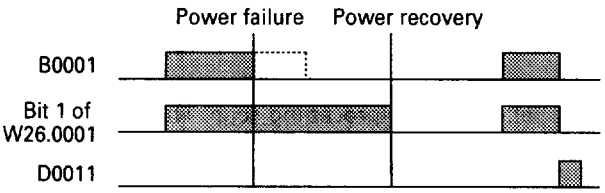


When the above initialization program is incorporated, differential relays do not operate even if the input signal changes during power failure as shown in the following figure.

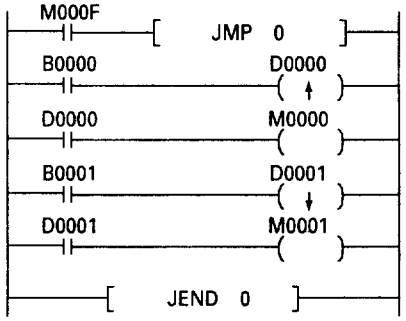
For rising edge differential

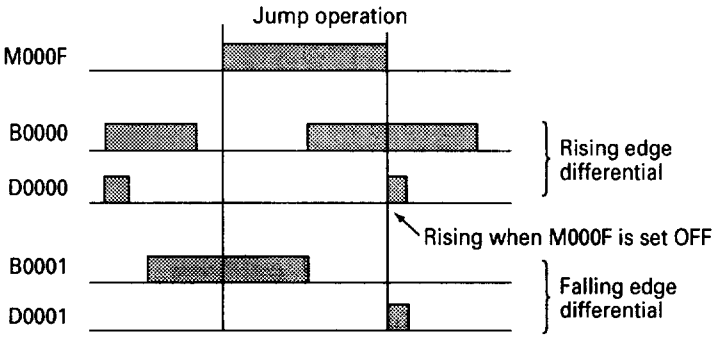


For falling edge differential



When differential relays are incorporated in a jump routine, the applicable timing chart is as follows.

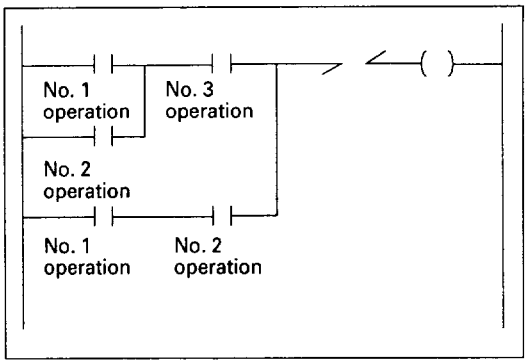




Processor (: Applicable)											Program loader				
F30	F50 F50H	F85	F60	F70	F60H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

5. Inverse (—> <—)

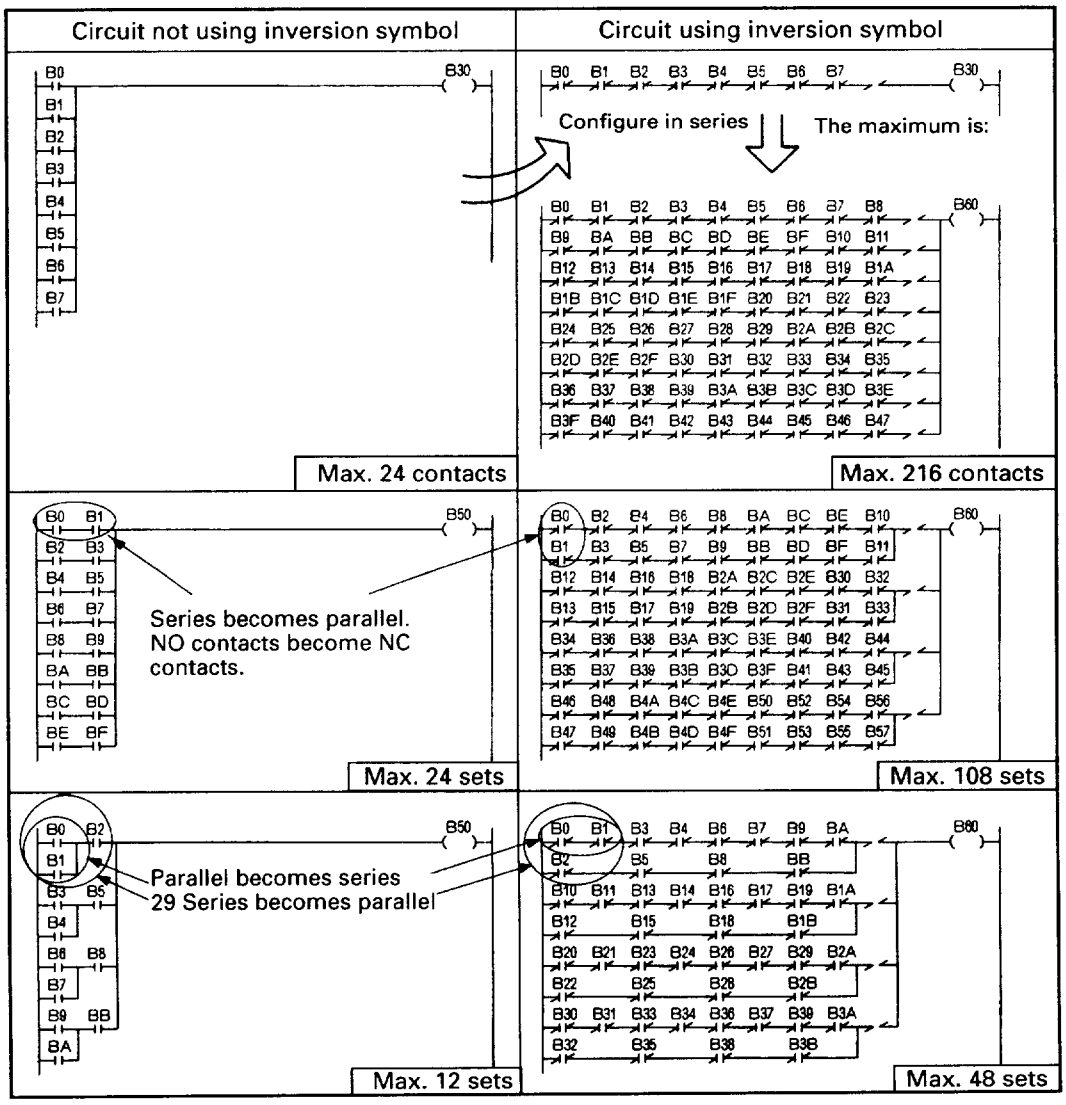
Instruction	Inverse	Description	Example
Symbol	—> <—		
Function	This instruction inverts the logical value of a logical circuit located to the left of the inversion symbol and transfers the inverted value to the circuit to the right of the inversion symbol.		



Output is ON when a signle unit is operated.

Advice on programming

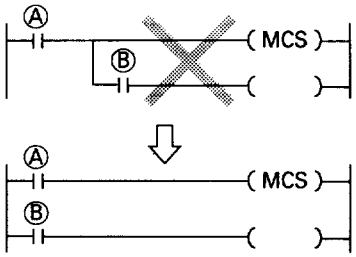
By using an invert instruction, the number of lines in the circuit can be reduced.

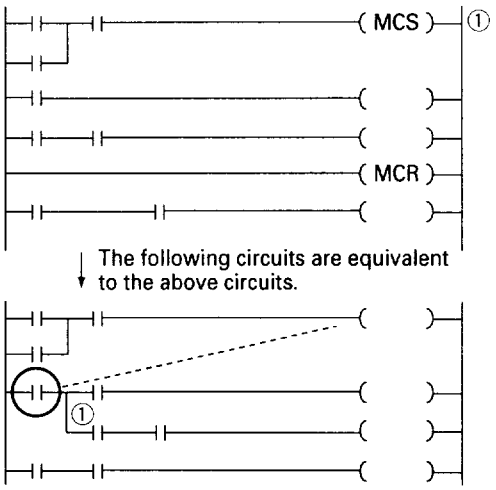


6. Master control, set (MCS), reset (MCR)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Mater control, set (MCS), reset (MCR)	F6, F7	Description	Example
Symbol	—(MCS)⊣ —(MCR)⊣		—(MCS)⊣ —(MCS)⊣ —(MCS)⊣	MCS, MCR
Function	When a program part is bound between MCS and MCR, the part is interlocked as a whole. Note: Do not make a circuit in parallel with MCS or MCR.		① The number of MCS (the number of nesting) is not restricted. Placing only a single MCR instruction resets all of the master control instructions regardless of the number of MCS. ② Reset circuits of counters are also interlocked. (They are not reset unless the MCS is ON.) ③ MCR does not require a conditional contact.	The following circuits are equivalent to the above circuits.





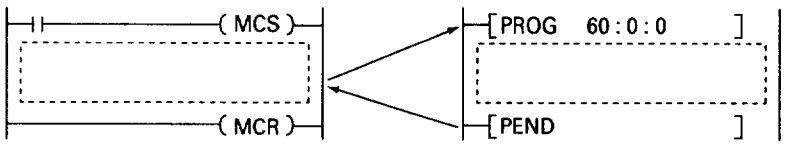
Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
—⊣⊢—	O	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	—	—	↑	—

*When using P or PE-link, L is also an effective identifier.

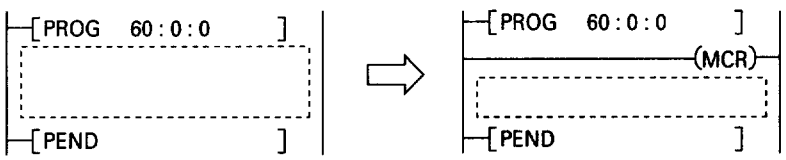
Note on programming

In a program containing an interrupt program, when interrupt is activated during execution of the program part between the MCS instruction and the MCR instruction, the condition of the master control (MCS instruction ON or OFF) is transferred to the interrupt program. Thus, if the MCS instruction is OFF, the interrupt program is not executed.



Countermeasure

The MCR instruction is placed at the leading position of the interrupt program (immediately after the Program Entry instruction (page 3-149)), so that the interrupt program will be executed regardless of the master control condition.

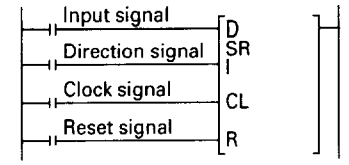


3-3-2 Shift register (SR)

Processor (▒ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120S	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

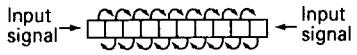
Instruction	Shift register	F5

Symbol



Function

A shift register has an arbitrary bit length of up to 511 bits. Input signals can be shifted in either direction (left or right) according to the direction signal.

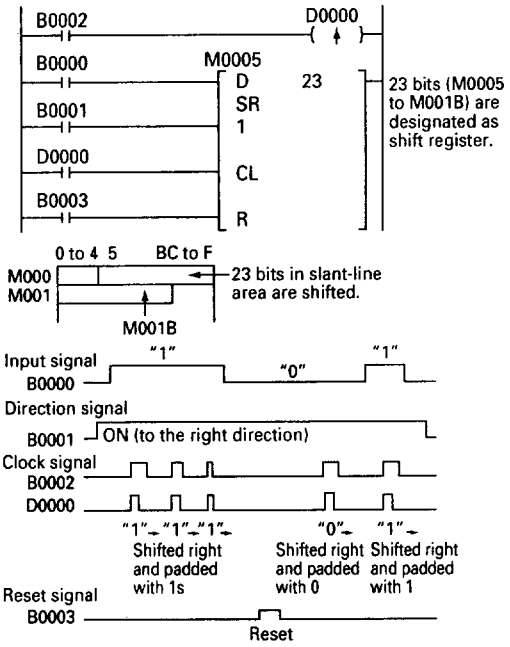


Note:
Direction signal
ON: Shift to the right
OFF: Shift to the left
Clock signal
ON: Shifting is carried out.
OFF: Shifting is not carried out.

Description

① When the reset signal is set ON, reset is recognized and contents of the shift register are all set OFF.
② The shift register instruction is restricted to the B, M, and K areas.
③ Clock must be differential signal (identifier "D").
④ The register data can be replaced with 1 or 0 by using the set or reset coil instruction.

Example



Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	S	Z	E	O
SR	○*	○	○	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

* B is not usable for input address.
** When using P or PE-link, L is also an effective identifier.

3-3-3 Step control (-(SC)-)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F65	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D08	D10S	D20	LITE	Soft. ware

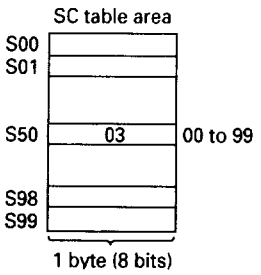
Instruction	Step control
Symbol	S -(SC)-
Function	• Up to 100 groups of sequence control, having up to 100 steps, can be configured. • Four characteristics of the stepping control relay: Self-hold function Interlock function Power-off step retentive function Subsequence priority

Description

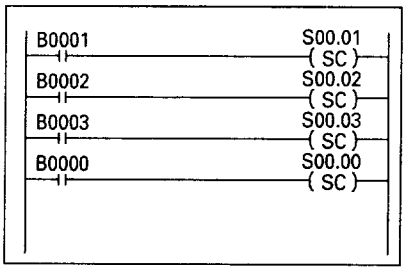
① Up to 100 group (from 00 to 99) control sequences can be programmed. Each control program can have up to 100 (from 00 to 99) program steps.

S00.00 to S99.99
- Step No.
- Sequence group No.

② The effective program step numbers of the control sequences are stored in the corresponding areas of the SC table.



Example



* "0" step shown in the above diagram is equivalent to "CLEAR" instruction of sequence control. No other specific "CLEAR" instruction is provided.

This step control circuit is designated so that the coil last turned on operates.

No.	B0001	B0002	B0003	B0000	S00.01	S00.02	S00.03	S00.00
1	ON	OFF	OFF	OFF	○			
2	ON	ON	OFF	OFF		○		
3	ON	ON	ON	OFF			○	
4	ON	ON	ON	ON				○

○: Indicates the step position.

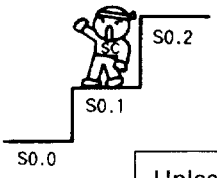
Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
-||-	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	S	Z	E	O
-(SC)-	–	–	–	–	–	–	○	–	–	–	–	–	–	–	–	–	–	–	–	–	–

Four characteristics of the stepping control relay

Characteristic 1/Self-hold function:
Each SC coil has a built-in self-hold function.

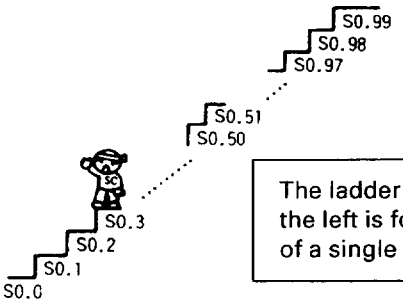
• Conceptual drawing



Unless the instruction "then go to a certain step number" is issued, control remains at the same step.

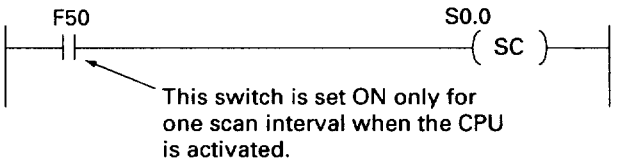
Characteristic 2/Interlock function:
Each SC coil is interlocked.

• Conceptual drawing

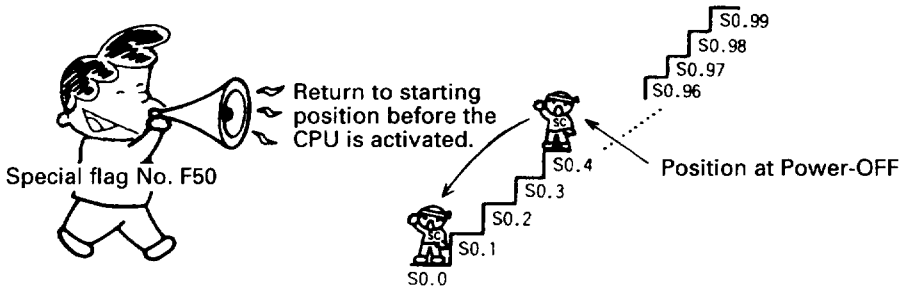


The ladder of 100 steps on the left is for exclusive use of a single SC.

Characteristic 3/Power-OFF step retentive function:
Since the S-area memory is nonvolatile, its contents are retained during a power-OFF state. Accordingly, it is necessary to add the following circuit if operations are always to be started from a specified starting position.

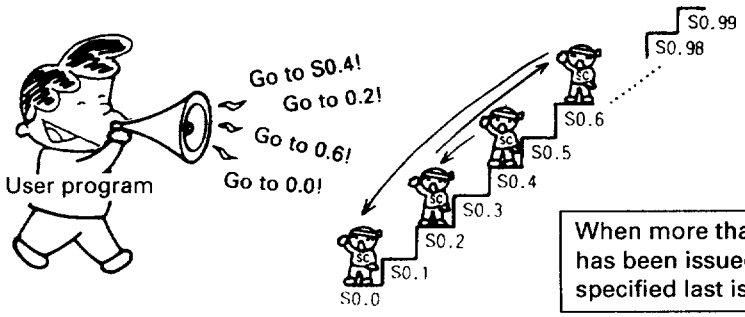


• Conceptual drawing



Characteristic 4/Subsequence priority:
Only one SC coil will be set ON, even if more than one entry appears with respect to the same word number. This fact has already been proved with regard to characteristic 2, in which case later programming steps are given higher priorities when they are output.

• Conceptual drawing



When more than one instruction has been issued, the position specified last is validated.

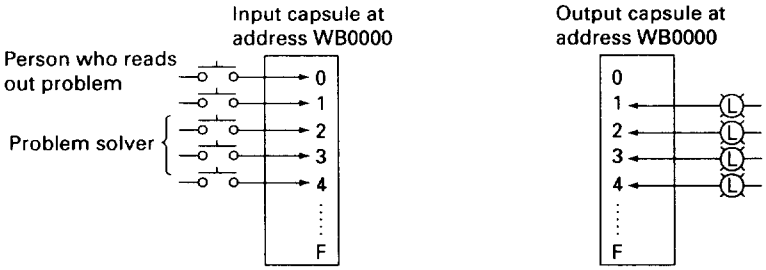
ONE-POINT ADVICE "The fastest is the winner" circuit

An example of using SC is shown below.

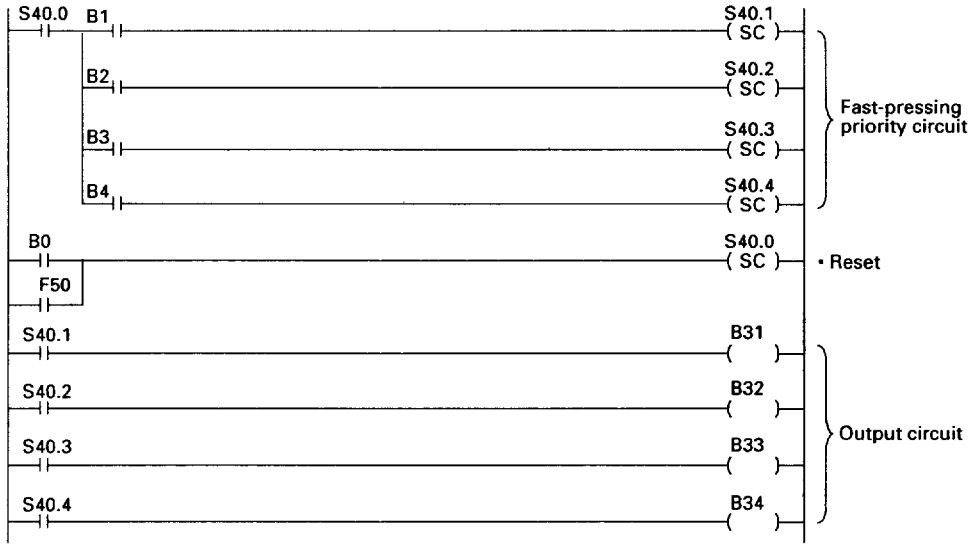
1. **Instruction**
   SC

2. **Operation**
   There are four problem solvers. A person reads out the problem, and the solvers press a button switch. An output is provided only from the person who presses the button fastest. The next problem starts when the person who reads out the problems pushes the reset button.

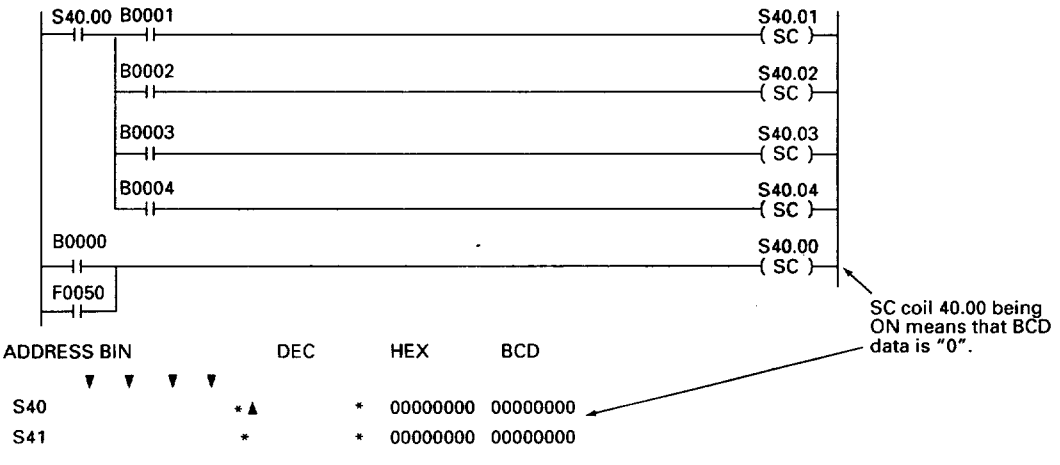
3. **System diagram**



4. **Program**



Example of monitoring with loader LITE



3-3-4 Timers

1. Timer processing

■ The following five types of timers having different functions are provided. (F30, F50, and F50H are provided with on-delay timer.)

(1) On-delay timer (time base: 0.01s or 0.1s) (F30, F50, and F50H series are provided with 0.01s timer only.)

(2) Off-delay timer (time base: 0.01s)

(3) Integrating timer (time base: 0.01s)

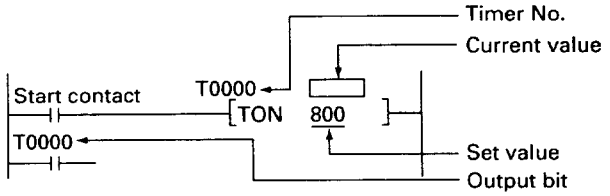
(4) Monostable timer (time base: 0.01s)

(5) Monostable timer (retriggable) (time base: 0.01s)

■ Timer operation

Timer operation is explained according to the following sequence diagram.

(1) In the program shown on the left, a set value of 800 is stored at set value data address TS000 *at the time of start-up.



(2) The current value data at address TR000 is incremented **by the value of the time base according to the start contact.

(3) When the current value becomes equal to or greater than the set value, output bit address T000 is set ON.

*at the time of start-up
- Power-ON
- When program block is changed over
- When program and system definition is changed
- When program operation is started

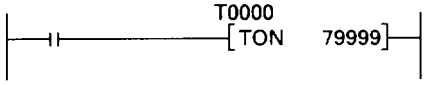
**by the value of the time base
If a timer instruction is jumped over by LOOP, JUMP, SKIP, or FM instructions within one scan, or such instructions are executed repeatedly, accurate increment is not possible.

Timer-related memory

Output bit address (T)	Set value data address (TS)	Current value data address (TR)
T000	TS000	TR000

■ One of the following two methods can be used to set timers.

(1) A numeric value is directly specified as an instruction operand and is used as the set value. (Program preset)

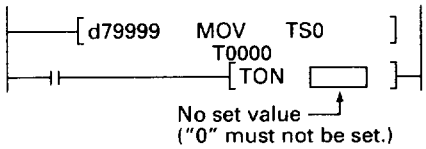


The long-time on-delay timer can be set only by using method 1).

• Notes on timer specifications

Timer functions are specified by using instructions. Timer numbers must not be double-assigned in the specifications.

(2) A transfer instruction is used to set a value in the set value area. (Preset by data)

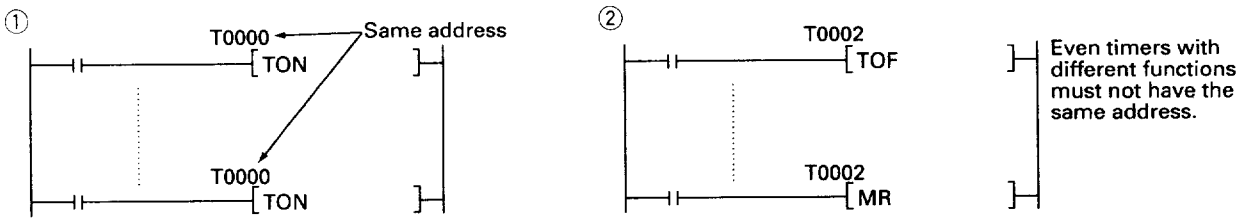


If an erroneous value is set, the timer instruction must be deleted and a new timer instruction must be written.

Key points

1. When there is no set value, the preceding preset value is retained.
2. If power is turned OFF after a value has been set, the timer count returns to the set value.
3. A program without a set value can be checked by using the program loader in the write mode.

Examples of erroneous specifications with double-assigned timer numbers



If an erroneous specification is made as shown above, correct the timer addresses and restart the processor, the alarm lamps and contacts will be set OFF.

2. On-delay timer (TON)

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	On-delay timer	Description	Example
Symbol	Input signal T□□□□ TON□□□□ Timer address Setting time	① Time base is 0.01 or 0.1s. For details, see below. ② The timer setting range is 0 to 79999999 x 0.01s or 0.1s.	B0000 T0000 TON 500 Set value: 5s. T0000 B0010 T0000 B0011
Function	① When input signal is turned ON, on-delay timer is started, and when it reaches the setting time, timer contact becomes ON. ② When input signal is turned OFF, time value becomes 0 and contact is turned OFF. Input signal / Output signal / Setting time	③ Input signal for up to 9 contacts can be set in series.	

0.01s	799,999,99s 13,333min 222hours 9.25days
0.1s	7,999,999,9s 133,330min 2,220hours 92.5days

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
⊣⊢	○	○	○	○	○	○	○	○	○	○*	○	○	○	○	○	○	○	—	—	—	—

*When using P or PE-link, L is also an effective identifier.

Time base and timer address for each MICREX-F series

Series	Time base	Address range	No. of addresses	Remarks
F30, F50, F50H	0.01s	T0 to T127	128	0.1s timer is cannot be used.
F55, F60, F70, F80H, F120H	0.01s	T0 to T255	256	*1)
	0.1s	T511 to T767	256	
F70S, F120S, F140S, F150S	0.01s	T0 to T511	512	*1)
	0.1s	T512 to T999	488	

*1) Because other timer instructions use the same address range, duplicate-use of one address should be avoided.

Note on programming

Make sure to write the timer instruction in a location where it will be executed at each scan.
If the processing is not executed due to the jump instruction or if the timer instruction is used in the interrupt program or function module (FM), the timer cannot count correctly.

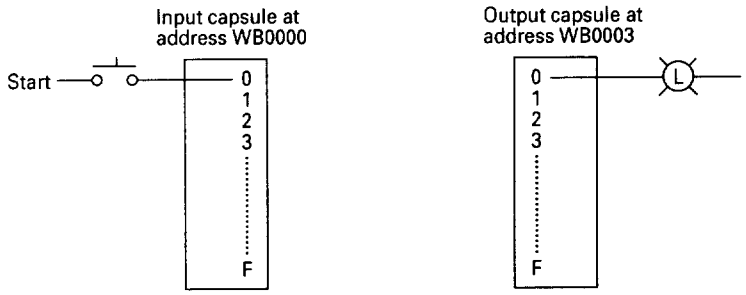
ONE-POINT ADVICE **Blinker circuit**

An example of using TON is shown below.

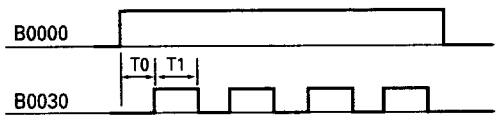
1. Instruction
TON

2. Operation
Two timers are used to provide a blinking output.

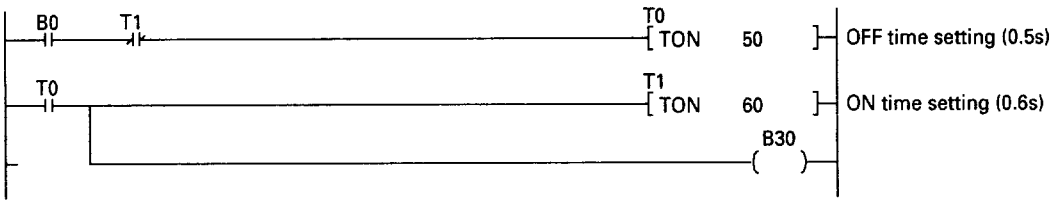
3. System diagram



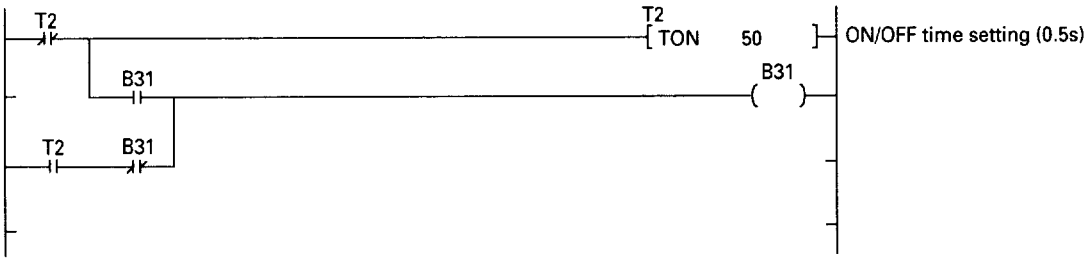
• Time chart



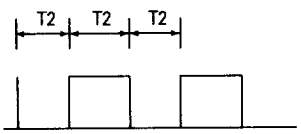
4. Program



Example with one timer



• Time chart



Section 3 Instructions

3. Off-delay timer (TOF)

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

Instruction	Off-delay timer	Description	Example
Symbol	Input signal T□□□□ TOF □□□□ Timer address Setting time	① The time base is 0.01s. ② The timer setting range is 0 to 79999999 x 0.01s. 799,999.99 s 13,333 min 222 hours 9.25 days Note: The timer base is 0.01s only. ③ The input signals for up to 9 contacts can be set in series.	B0000 T0000 TOF 500 B0010 T0000 Set value: 5s
Function	After the input signal is set OFF, the timer starts counting. When it reaches the setting time, the output signal is set OFF. Input signal Output signal Setting time		

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
⊢⊣⊢	○	○	○	○	○	○	○	○	○	○*	○	○	○	○	○	○	○	–	–	–	–

* When using P or PE-link, L is also an effective identifier.

Timer address for each MICREX-F series

Series	Address range	No. of addresses	Remarks
F55, F60, F70, F80H, F120H	T0 to T255	256	• 0.1s timer cannot be used. • Because other timer instructions use the same address range, duplicate-use of one address should be avoided.
F70S, F120S, F140S, F150S	T0 to T511	512	

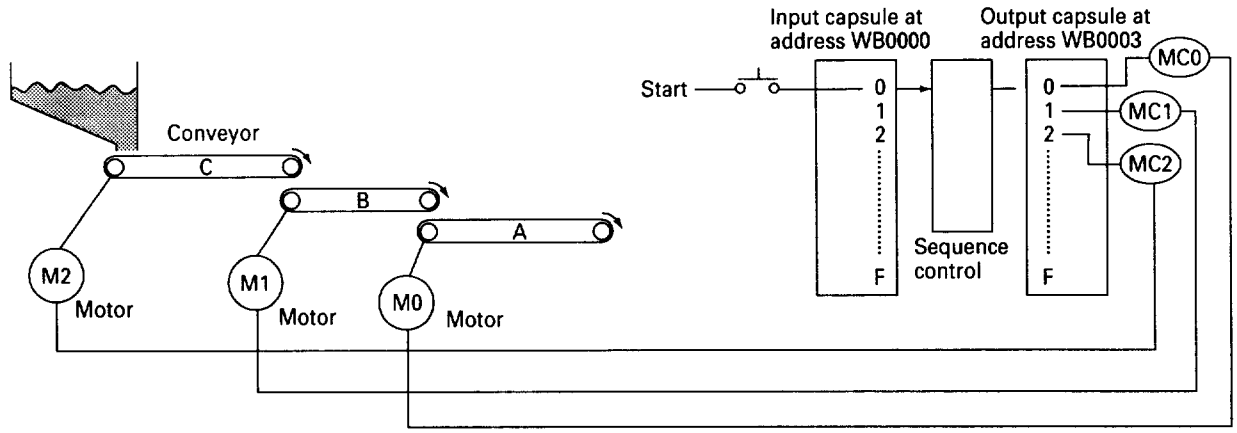
ONE-POINT ADVICE **Controlling conveyors**

An example of using TON and TOF is shown below.

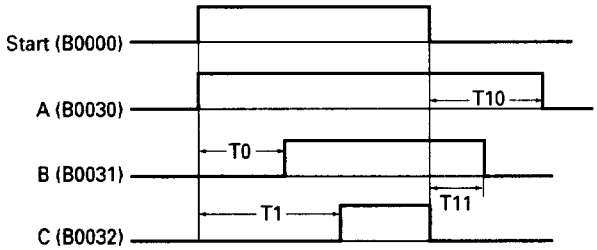
1. Instruction
TON, TOF

2. Operation
Start (A → B → C) and stop (C → B → A) of several related conveyors in proper sequence.

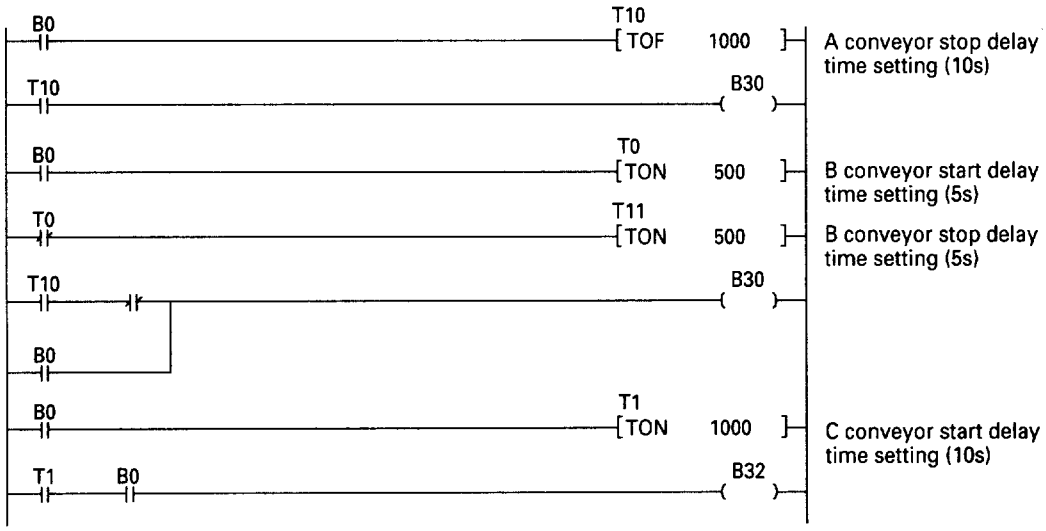
3. System diagram



• Time chart

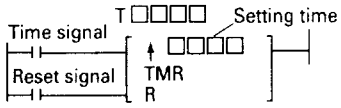
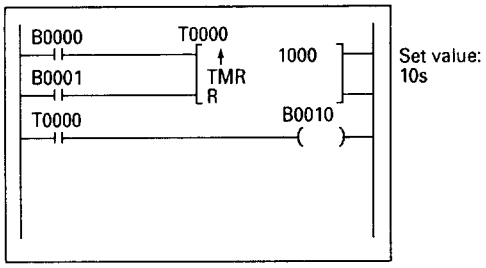
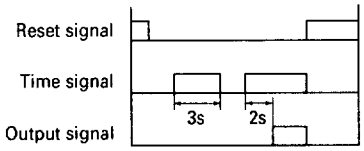


4. Program



4. Integrating timer (TMR)

Processor (▒▒: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

Instruction	Integrating timer	Description	Example
Symbol		① The time base is 0.01s. ② The timer setting range is 0 to 79999999 x 0.01s. 799,999.99 s 13,333 min 222 hours 9.25 days Note: The time base is 0.01s only. ③ The input signals for up to 9 contacts can be set in series.	
Function	① The time signal is used to count the integrated time of input. The output signal is issued when a time-up occurs. ② The time signal can be divided as shown below. 		

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
⊣⊢	○	○	○	○	○	○	○	○	○	○*	○	○	○	○	○	○	○	—	—	—	—

* When using P or PE-link, L is also an effective identifier.

Timer address for each MICREX-F series

Series	Address range	No. of addresses	Remarks
F55, F60, F70, F80H, F120H	T0 to T255	256	• 0.1s timer cannot be used. • Because other timer instructions use the same address range, duplicate-use of one address should be avoided.
F70S, F120S, F140S, F150S	T0 to T511	512	

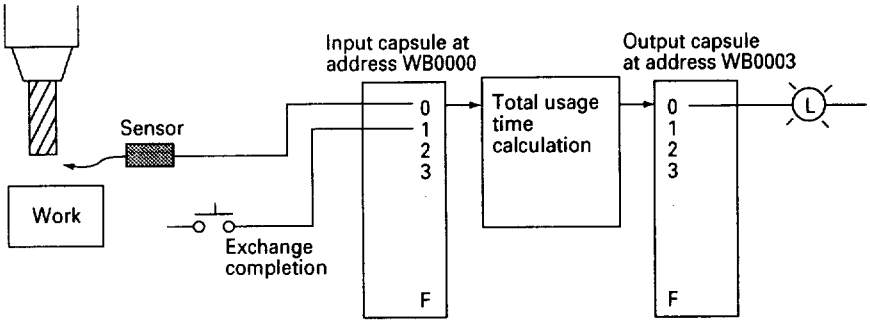
ONE-POINT ADVICE **Alarm circuit for tool life**

An example of using TMR is shown below.

1. **Instruction**
   TMR

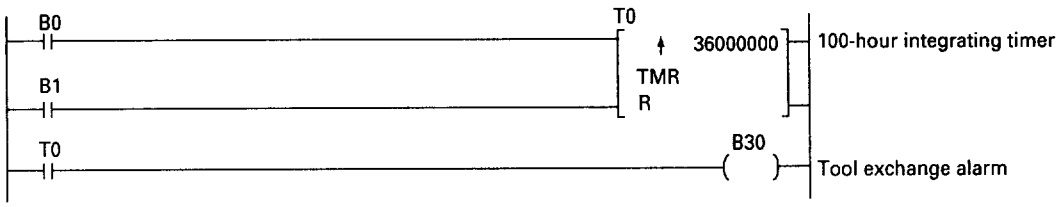
2. **Operation**
   Tool usage time at a machining center etc. is calculated, and an alarm for tool exchange is output.

3. **System diagram**



Address	Usage
B0000	Drill descent detection
B0001	Drill exchange completion
B0030	Tool life alarm
T0000	Tool life setting timer

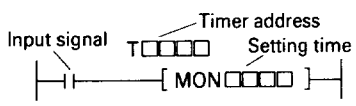
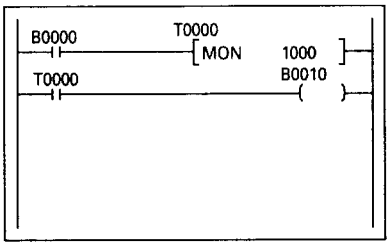
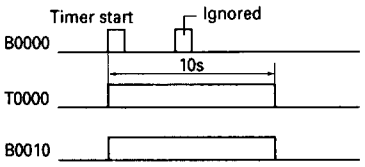
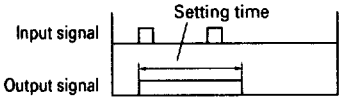
4. **Program**



Remark: Integrating timer retains the previous value even at power failure, so PC doesn't incur any problem in case of nighttime power failure.

5. Monostable timer (MON)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Monostable timer	Description	Example
Symbol		① The time base is 0.01s. ② The timer setting range is 0 to 79999999 x 0.01s 799,999.99 s 13,333 min 222 hours 9.25 days Note: The time base is 0.01s only ③ Timer address is as follows: 0000 to 0511 ④ The input signals for up to 9 contacts can be set in series.	 
Function	① The timer starts counting when the first input signal is set and continues output until the setting time is reached. ② Even if the input signal continues, the output signal is set OFF when the setting time is reached. 		

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
⊣⊢	○	○	○	○	○	○	○	○	○	○*	○	○	○	○	○	○	○	—	—	—	—

* When using P or PE-link, L is also an effective identifier.

Timer address for each MICREX-F series

Series	Address range	No. of addresses	Remarks
F55, F60, F70, F80H, F120H	T0 to T255	256	• 0.1s timer cannot be used. • Because other timer instructions use the same address range, duplicate-use of one address should be avoided.
F70S, F120S, F140S, F150S	T0 to T511	512	

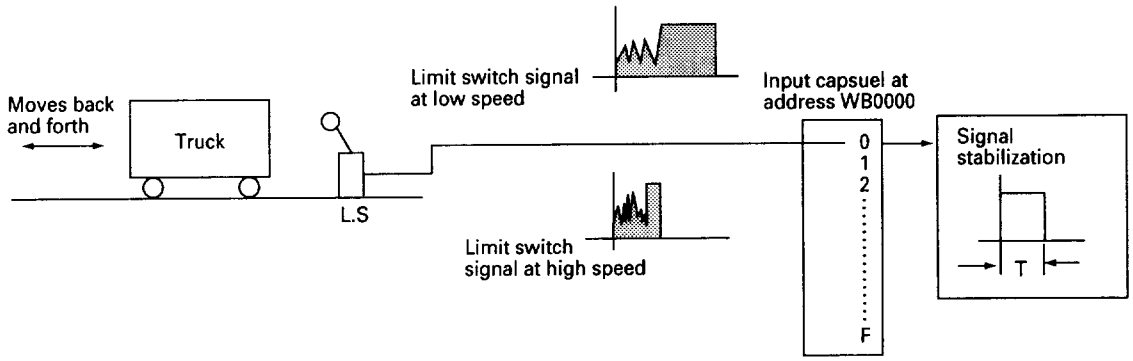
ONE-POINT ADVICE **Chattering prevention circuit**

An example of using MON is shown below.

1. **Instruction**
   MON: Monostable timer

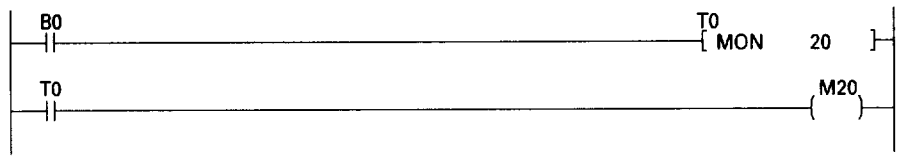
2. **Operation**
   Prevents chattering of passing signal (limit switch) of object moving at non-fixed speed, so as to provide a stable signal.

3. **System diagram**



Address	Usage
B0000	Position detection limit switch
M0020	Constant time output relay
T000	Chattering prevention timer

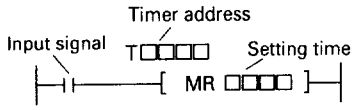
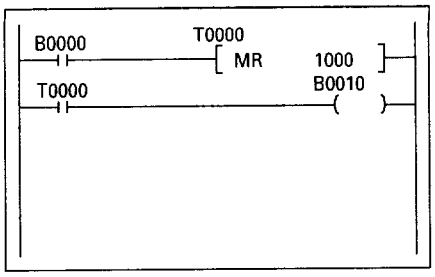
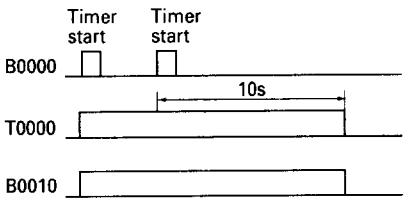
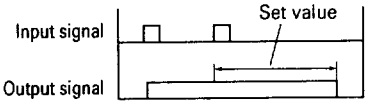
4. **Program**



- Even if B0000 produces chattering after it is instantaneously turned ON, M0020 is held for 0.2 second.

6. Monostable timer (Retriggable) (MR)

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Monostable timer	Description	Example
Symbol		① The time base is 0.01s. ② The timer setting range is 0 to 79999999 x 0.01s 799,999.99 s 13,333 min 222 hours 9.25 days Note: The time base is 0.01s only ③ Timer address is as follows: 0000 to 0511 ④ The input signals for up to 9 contacts can be set in series.	 Timer restart 
Function	① The timer starts counting at the rising edge of last input signal and continues output until the setting time is reached. ② If another input signal is set while counting is being done, the timer is restarted. 		

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
⊢⊣⊢	○	○	○	○	○	○	○	○	○	○*	○	○	○	○	○	○	○	—	—	—	—

* When using P or PE-link, L is also an effective identifier.

Timer address for each MICREX-F series

Series	Address range	No. of addresses	Remarks
F55, F60, F70, F80H, F120H	T0 to T255	256	• 0.1s timer cannot be used. • Because other timer instructions use the same address range, duplicate-use of one address should be avoided.
F70S, F120S, F140S, F150S	T0 to T511	512	

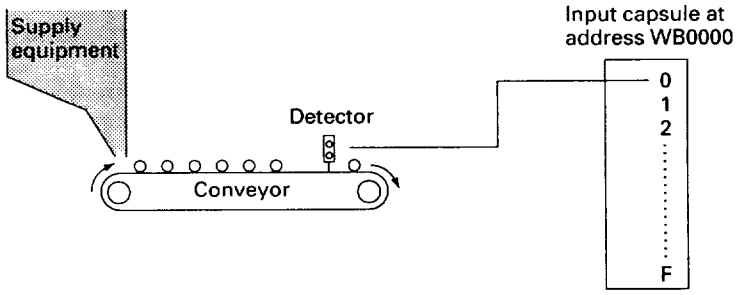
ONE-POINT ADVICE **Conveyor fault detection circuit**

An example of using MR is shown below.

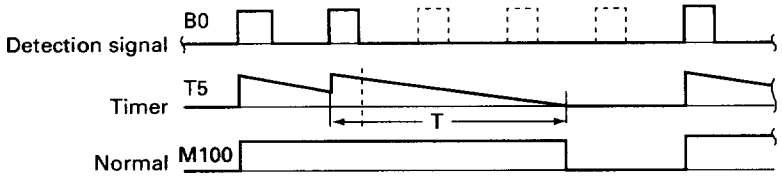
1. **Instruction**
   MR: Monostable timer (retriggable)

2. **Operation**
   Fault of conveying equipment is detected according to products flowing at a constant time.

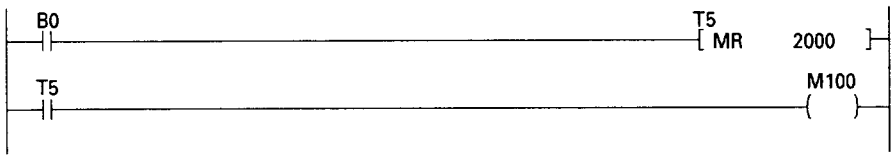
3. **System diagram**



• Time chart



4. **Program**



- If next product doesn't flow within 20 seconds, time-up is reached and M100 is turned OFF.
- Normal when ON

3-3-5 Counters

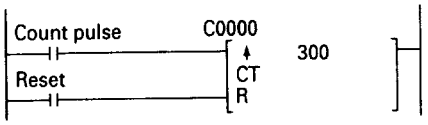
1. Counter processing

The following four types of counters having different functions are provided.

(1) Counter (Up counter)
(2) Down counter
(3) Up/down counter
(4) Ring counter

■ Counter operations

Counter operations are explained according to the following sequence diagram.



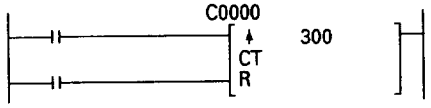
(1) In the program on the left, a set value of 300 is stored at set value data address CS000 *at the time of startup.
(2) Data at current value data address CR000 is incremented by count pulses.
(3) Output bit address C0000 is set ON when the counter current value becomes equal to or greater than the set value.
(4) When the reset signal is set, current value data address CR000 is set to 0 and output bit address C0000 is set OFF.
*at the time of startup
- Power-ON
- When program block is changed over
- When program and system definition is changed
- When program operation is started

Counter-related memory

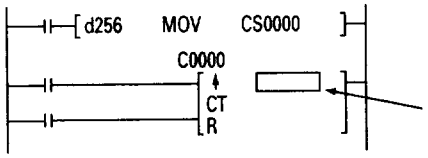
Output bit address (C)	Set value address (CS)	Current value data address (CR)
C0000	CS000	CR000

■ One of the following two methods can be used for setting.

(1) A numeric value is directly specified by an instruction operand as the set value.



(2) A transfer instruction is used to set a value in the set value area.



Key points

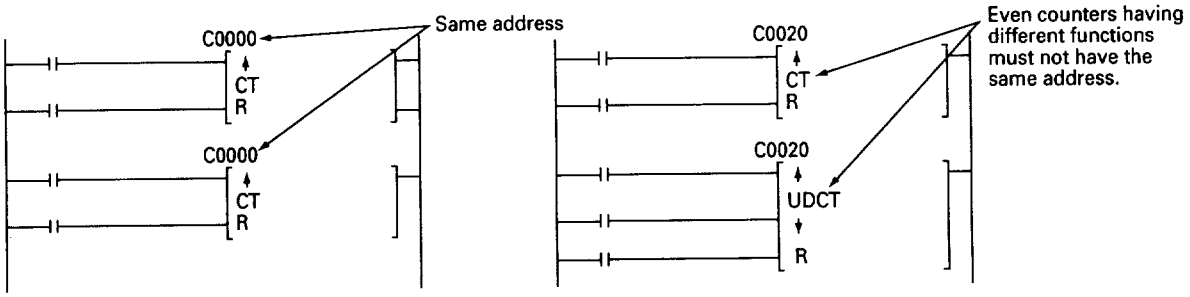
1. If there is no set value, the preceding preset value is retained.
2. If power is turned OFF while a value is being set, the counter current value is returned to the set value.
3. A program without a set value can be checked by using the program loader in the write mode.

No set value ("0" must not be set.)
If an erroneous value has been set, the counter instruction must be deleted and a new counter instruction must be written.

■ Notes on counter specifications

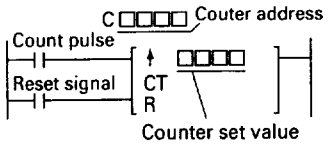
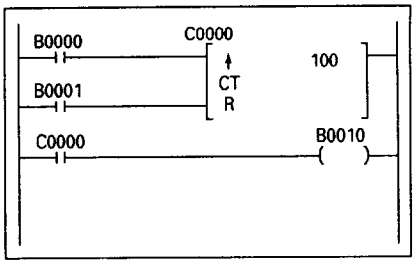
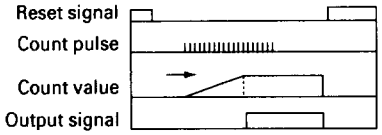
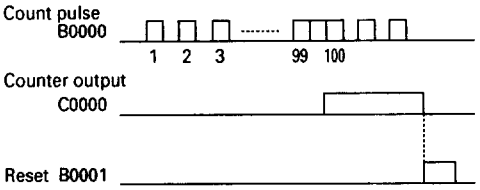
Counter types are specified by using instructions.
Counter numbers must not be double-assigned in the specifications.

Examples of erroneous specifications with double-assigned counter numbers.

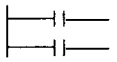


2. Counter (Up counter) (CT)

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Counter (Up counter)	Description	Example
Symbol		① The counter setting range is 0 to 79999999. ② Count-up operation is carried out until count value reaches 79999999 (max.). ③ The reset signal must be given to reset the count value at the beginning of this circuit.	
Function	① Count value is incremented by one at the rising edge of the count pulse. ② When count value becomes the same as set value, output signal is issued. 		

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	—	—	—	—

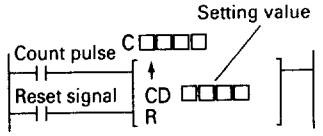
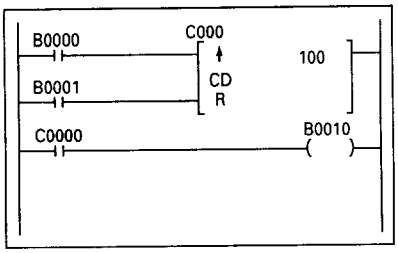
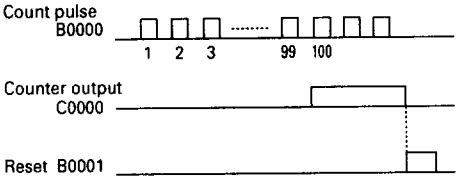
Timer address for each MICREX-F series

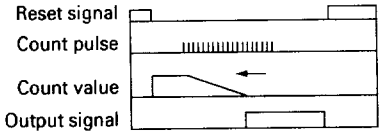
Series	Address range	No. of addresses	Remarks
F30, F50, F50H	C0 to C31	32	Because other counter instructions use the same address range, be careful to avoid double-assigning an address.
F60	C0 to C127	128	
F55, F70, F80H, F120H	C0 to C255	256	
F70S, F120S, F140S, F150S	C0 to C511	512	

Section 3 Instructions

3. Down counter (CD)

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Down counter	Description	Example
Symbol		① The counter setting range is 0 to 79999999. ② Count-down operation is carried out until count value reaches 0. ③ The reset signal must be given to reset the count value at the beginning of this circuit.	
Function	① The count value is decremented by one at the rising edge of the count pulse. ② The output signal is set when the current value reaches 0.		



Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
⊣⊢ ⊣⊢	○	○	○	○	○	○	○	○	○	○*	○	○	○	○	○	○	○	—	—	—	—

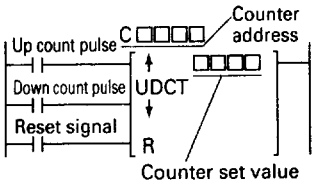
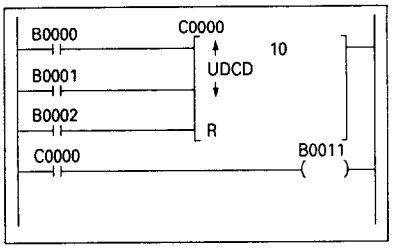
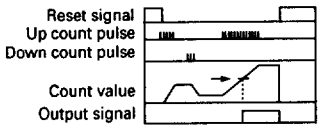
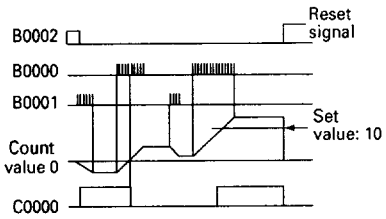
* When using P or PE-link, L is also an effective identifier.

Timer address for each MICREX-F series

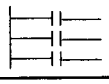
Series	Address range	No. of addresses	Remarks
F60	C0 to C127	128	Because other counter instructions use the same address range, be careful to avoid double-assigning an address.
F55, F70, F80H, F120H	C0 to C255	256	
F70S, F120S, F140S, F150S	C0 to C511	512	

Processor (▨: applicable)											Program loader			
F30	F50	F50H	F60	F80	F80H	F81	F100	F120	F120H	F200	D05	D10	D20	LITE

3. Up/Down counter (CD)

Instruction	Up/down counter	F4	Description	Example
Symbol			① The counter setting range is 0 to 79999999. ② When C31 is used in F30 or F50H series, or C127 is used in F60 series, it is automatically registered as a built-in high-speed counter. Note that it cannot be used as an ordinary up/down counter. For details on the usage of built-in high-speed counter, see the User's Manual "Hardware" of each series.	
Function	① The current value is incremented by one at the rising edge of the up-count pulse. The current value is decremented by one at the rising edge of down-count pulse. ② When the current value reaches the set value or becomes -1 or less, the output signal is set ON. 			

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
	○	○	○	○	○	○	○	○	○	○*	○	○	○	○	○	○	○	—	—	—	—

* When using P or PE-link, L is also an effective idetifier.

Timer address for each MICREX-F series

Series	Address range	No. of addresses	Remarks
F30, F50, F50H	C0 to C31	32	If the Up/down counter (UDCT) uses C31, this counter operates as a build-in high-speed counter in the F30 and F50H Series. *1)
F60	C0 to C127	128	If the Up/down counter (UDCT) uses C127, this counter operates as a build-in high-speed counter. *1)
F55, F70, F80H, F120H	C0 to C255	256	*1
F70S, F120S, F140S, F150S	C0 to C511	512	*1

*1 Use care to prevent address duplication because the same address area is shared with other counter instructions.

5. Ring counter (RCT)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Ring counter	Description	Example
Symbol	Count pulse Reset signal RCT R	① The counter setting range is 0 to 79999999.	B0000 C0000 10 B0001 RCT R C0000 B0010
Function	① The current value is incremented by one at the rising edge of count signal and the current value is reset to 0 when the first count signal is set ON after the counter is count-up. ② When the current value reaches the set value, the output signal is set ON. Reset signal Count signal Count value Output signal		B0000 Count value C0000

Effective identifier

	B	M	K	D	F	A	S	T	C	L	i	j	k	ℓ	m	P	Q	Influence flag			
																		S	Z	E	O
⊣⊢ ⊣⊢	○	○	○	○	○	○	○	○	○	○*	○	○	○	○	○	○	○	—	—	—	—

* When using P or PE-link, L is also an effective idetifier.

Timer address for each MICREX-F series

Series	Address range	No. of addresses	Remarks
F60	C0 to C127	128	Because other counter instructions use the same address range, be careful to avoid double-assigning an address.
F55, F70, F80H, F120H	C0 to C255	256	
F70S, F120S, F140S, F150S	C0 to C511	512	

3-3-6 Processing during data ON-line change (timer/counter in progress)

Explained in the table below is the operation when the set value (set value in instruction or set value area) of timer (TMR) or counter (CNT) undergoing operation is changed during processor operating (RUN) by performing ON-line change of the user program or loader.

State before change	TMR: Counting/CNT: Counting			At count up
Contents of change	Set value > Current value	Set value ≤ Current value	0 ≥ Current value	—
On-delay timer (TON)	Continues counting	Time-up is created and current value becomes equal to set value.	Current value is counted by absolute value. (Becomes positive value by first counting.)	Time-up status remains. (Current value is retained.)
Off-delay timer (TOF)	Continues counting	Time-up is created and current value becomes "0".	Current value is counted by absolute value. (Becomes positive value by first counting.)	—
Monostable timer (MON)	Continues counting	Time-up is created and current value becomes "0".	Current value is counted by absolute value. (Becomes positive value by first counting.)	—
Monostable timer (Retriggable) (MR)	Continues counting	Time-up is created and current value becomes "0".	Current value is counted by absolute value. (Becomes positive value by first counting.)	—
Integrating timer (TMR)	Continues counting	Time-up is created and current value becomes equal to set value. (However, when count input is OFF, time-up is not created. Time-up is created simultaneously with count input ON.)	Current value is counted by absolute value. (Becomes positive value by first counting.)	Time-up status remains. (Current value is retained.)
Counter (CT)	Continues counting (Counter is count-up when current value = set value)	Time-up is created by subsequent count input, and current value becomes equal to set value.	Current value is counted by absolute value. (Becomes positive value by first counting.)	By subsequent count input, counting is effected in the same way as mentioned on the left, and time-up is released unless time-up status exists.
Down counter (CD)	Continues counting (Counter is count-up when current value is zero)	Continues counting (Counter is count-up when current value is zero.)	Current value is counted by absolute value. (Becomes positive value by first counting.)	
Up and down counter (UDCT)	Continues counting (Counter is count-up when current value ≥ set value or current value < 0)	By subsequent count input, ±1 counting is effected. Counter is count-up when counted result is greater than set value.	Counting is effected by keeping negative current value. (Counter is count-up when counted result is negative.)	
Ring counter (RCT)	Continues counting (Counter is count-up when current value = set value)	Returns to current value "0" at subsequent count value.	Current value is counted by absolute value. (Becomes positive value by first counting.)	

Remarks:

1. Each set value in the above table indicates value within the set value area (TS or CS).
2. Contents of changes indicate relations between set values and current values immediately after changes are completed.
3. When contents of the set value area are changed while set values are written in TMR/CNT instructions, set values written in the instructions are loaded into the set value area at the points of operation listed below.
   - Power supply OFF → ON
   - STOP → RUN
   - Program and system definition change during RUN and STOP
   - Program block change

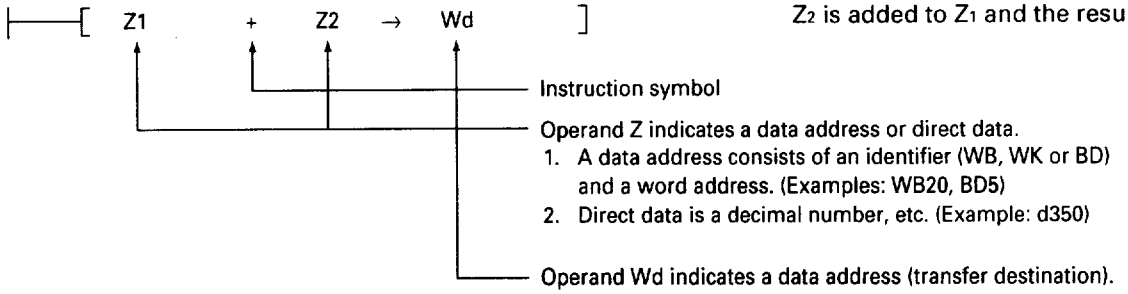
3-4 FPL data instructions

3-4-1 Method of expressing data instructions

A data instruction is expressed by an instruction symbol and operands. This section explains the expression method by using the line diagram examples.

(Example)

■ Arithmetic operation: Addition

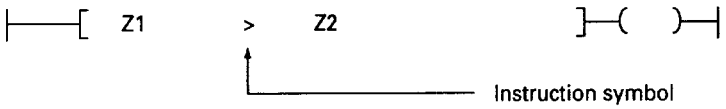


Description
Z_2 is added to Z_1 and the result is stored in Wd.

The following table lists the operands.

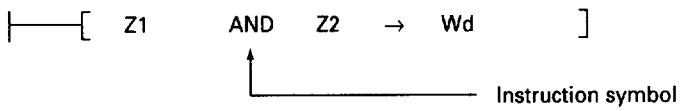
Operand	Direct number		Word address	Data module No.	Index register	Remarks
	Decimal number (BCD)	Hexadecimal number				
Zn	○	○	○			Depending on instruction
Ws			○			Transfer source
Wd			○			Transfer destination
N, N_1	○			○		Depending on instruction
R					○	Specification of index register

■ Comparison: >



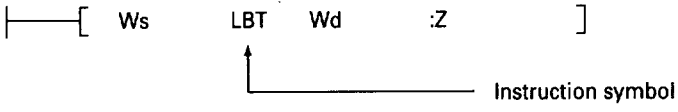
Description
When Z_1 is larger than Z_2, the output signal is set ON.

■ Logical operation: AND



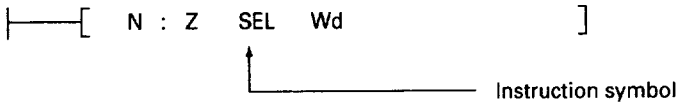
Description
The AND of Z_1 and Z_2 is obtained and the result is stored in Wd.

■ Transfer: LBT



Description
Z words of data are transferred from Ws to Wd.

■ File: SEL



Description
The data specified by Z in data module N is stored in Wd.

3-4-2 Line diagrams

1. Unconditional execution and conditional execution

The following line diagram expressions are independently used for the unconditional execution and conditional execution of operations.

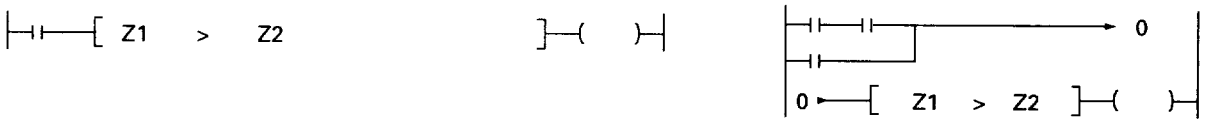
Operation	Line diagram	Remarks
Unconditional execution	⊢[ ]	An operation is executed for each scan.
Conditional execution	(When this is set ON, an operation is executed.) ⊢↑⊢[ ]	Conditional execution should be used to reduce the scan time.

2. With bit output (Example: comparison)

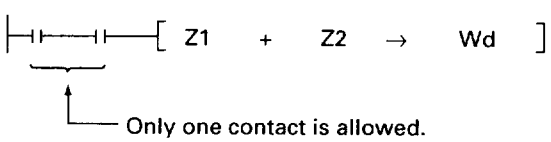
If the operational condition contact is set ON and Z1 is larger than Z2, the output coil is set ON.

3. Restrictions on line diagrams

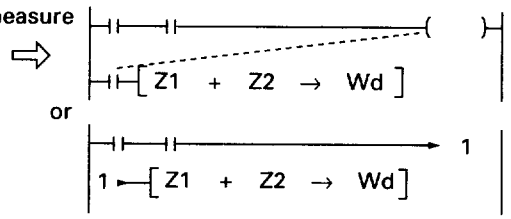
① The returning numbers 0 to 23 can be used as operational conditions.



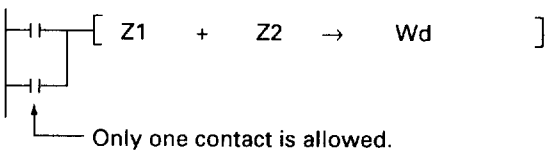
② Contacts cannot be connected in series.



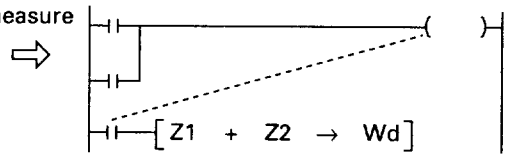
Countermeasure



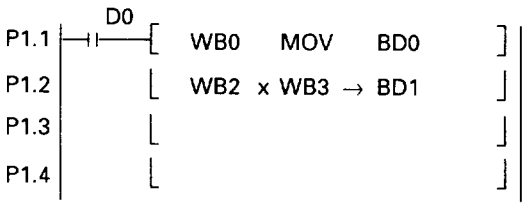
③ Contacts cannot be connected in parallel.



Countermeasure



4. How to reduce the number of program steps



This method is used when the operations of multiple line diagrams are to be executed according to a single operational condition contact. The same number of program steps as for conditional contacts can be reduced. The operation execution time can be also reduced.

Programming method

If the ARITH 2, F3 and I/R keys are pressed in that order when P1.2 is written, an operational condition contact can be eliminated.

3-4-3 Data format, operation instruction and operation flag

1. Data format

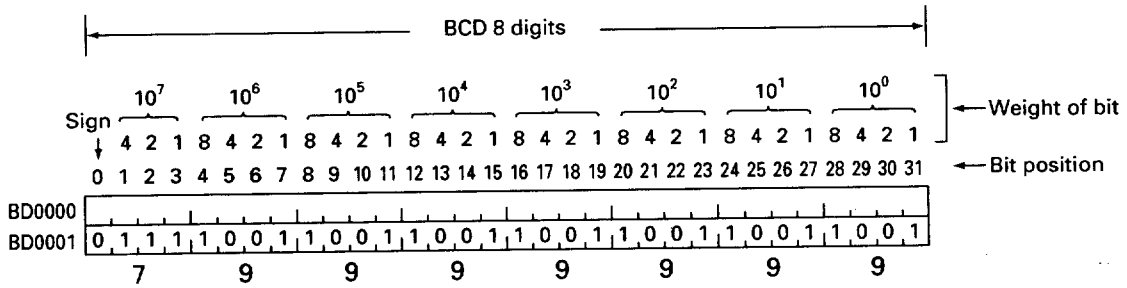
The data format is decided according to the identifier and instruction symbol used. The following table lists the data formats available for the MICREX-F Series.

Data format \ Identifier	B, M, K, D, F, A, S, T, C, L, Q	WS	WB, WM, WK, WF, WA, File, W5, WL, W21, W22, W23, W24, W26, W120, W121, W122, W123, W125	BD, TS, TR, CS, CR, W9, W25, File
Bit (ON/OFF) signal (Each bit is 1 or 0.)	○			
Unsigned BCD 2-digit data (00 to 99)		○		
Signed BCD 4-digit data (Sign bit, -7999 to 7999)			○	
Signed BCD 8-digit data (Sign bit, -79999999 to 79999999)				○ (BD file)
Signed 16-bit binary data (8000 to 7FFF)			○ (SI file)	
Signed 32-bit binary data (80000000 to 7FFFFFFF)				○ (DI file)

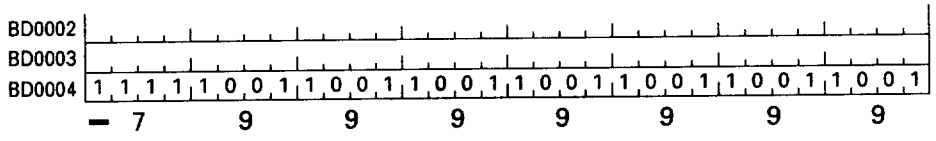
- The weight of each bit position is as follows.

Example 1: Using the BD area for signed BCD 8-digit data

The range of signed BCD 8-digit data expressed in this area is -79,999,999 to 79,999,999.



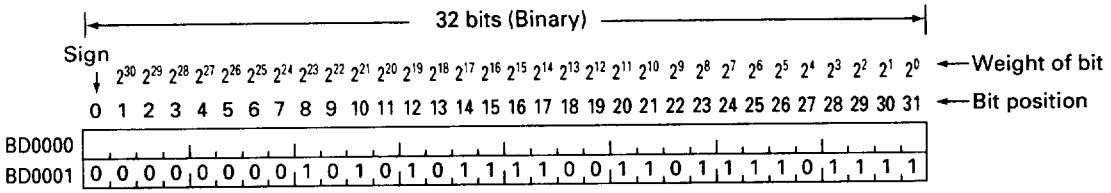
The above figure shows that the sign bit of BD0001 is 0 (indicating a positive sign) and that BD0001 is storing 79,999,999.



The above figure shows that the sign bit of BD0004 is 1 (indicating a negative sign) and that BD0004 is storing –79,999,999.

Example 2: Using the BD area for binary 32-bit data

The range of binary 32-bit data expressed in this area is h80000000 to hFFFFFFF.



The above figure shows that BD0001 is storing data h00ABCDEF.

Reference

The MICREX-F Series facilitates the construction of a hierarchical decentralized control system that includes the upper-level controller MICREX-E Series and the super minicomputer FACOM-S Series/PANAFACOM-U Series. Therefore, the MICREX-F Series incorporates an integrated method of bit weight assignment so that low-order bit positions become high-order digits.

The following table lists the numeric values that can be expressed in 8-bit, 16-bit and 32-bit areas.

	Range of data handled as BCD data				Range of data handled as binary data			
	Decimal expression	8-bit area	16-bit area	32-bit area	Conversion to decimal data	8-bit area	16-bit area	32-bit area
Range of positive numbers				Upper limit overflow	+2147483647 +2147483646			Max. 7FFFFFFF 7FFFFFFE
	+79999999 +79999998			+79999999 +79999998			Upper limit overflow	
	+8000		Upper limit overflow		+32768 +32767 +32766		Max. 7FFF 7FFE	
	+7999 +7998		+7999 +7998			Upper limit overflow		
	+100	Upper limit overflow			+256 +255 +254	Max. FF FE		
	+99 +98	99 98						
	+1 0	01 00	+0001 0000	+00000001 00000000	+1 0	01 00	0001 0000	00000001 00000000
Range of negative numbers	-1 -2 -3	Lower limit overflow	-0001 -0002 -0003	-00000001 -00000002 -00000003	-1 -2 -3	Lower limit overflow	FFFF FFFE FFFD	FFFFFFFF FFFFFFFE FFFFFFFD
	-7998 -7999		-7998 -7999					
	-8000		Lower limit overflow		-32766 -32767 -32768		8002 8001 Min. 8000	
	-79999998 -79999999			-79999998 -79999999	-32769		Lower limit overflow	
				Lower limit overflow	-2147483647 -2147483648			80000001 Min. 80000000

Key point

See Item 1 in Section 3-3-7 for details on how to express negative binary numbers.

2. Data formats and executing data instructions

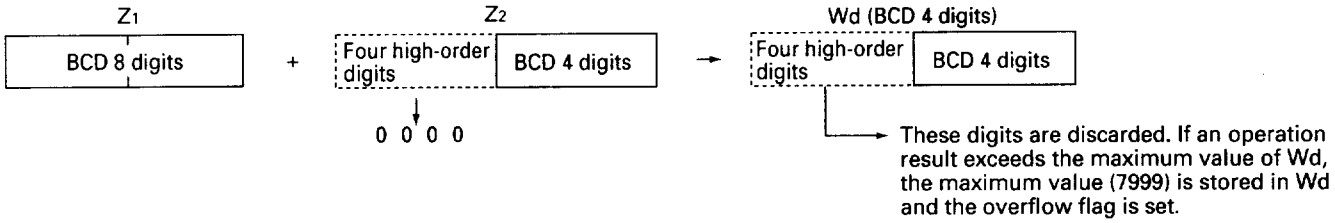
① For an area that can store BCD format data and binary format data, one of these two data formats must be specified according to an instruction symbol when instruction is executed.

② If the specified data format does not match the instruction symbol, an operation error occurs.

③ Arithmetic operations are executed using BCD 8-digit format and the results are stored in Wd format.

Source (operation data address)		Wd (result storage address)		
Z_1	Z_2	For Wd format of BCD 2 digits	For Wd format of BCD 4 digits	For Wd format of BCD 8 digits
BCD 2 digits	BCD 2 digits	BCD 2 digits	BCD 4 digits	BCD 8 digits
BCD 2 digits	BCD 4 digits			
BCD 2 digits	BCD 8 digits			
BCD 4 digits	BCD 4 digits			
BCD 4 digits	BCD 8 digits			
BCD 8 digits	BCD 4 digits			
BCD 8 digits	BCD 8 digits			

Example

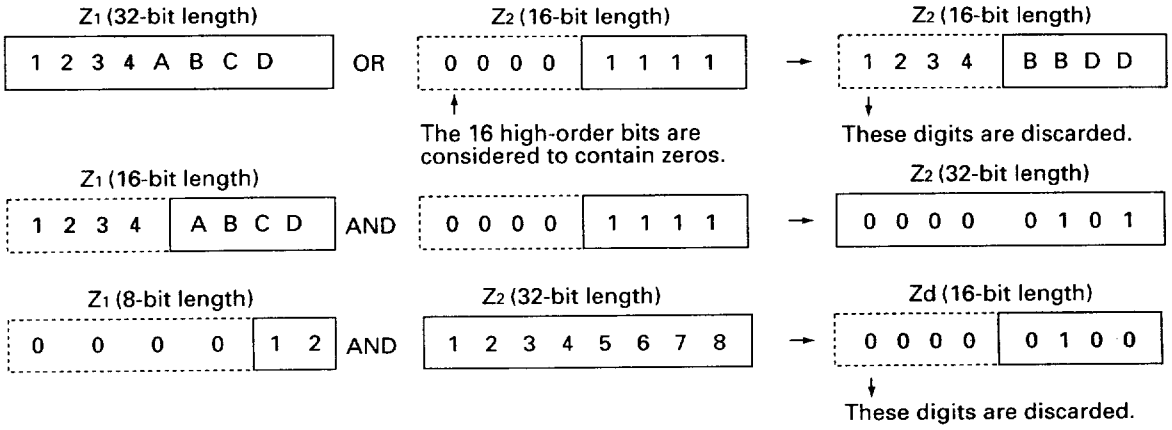


④ Logical operations are executed using 32-bit data. Only the part of the operation result that matches the data length of Wd is stored in Wd.

High-order bits of an operation result that exceeds the Wd data length are discarded.

Source (operation data address)		Wd (result storage address)		
Z_1	Z_2	For Wd 8-bit length	For Wd of 16-bit length	For Wd of 32-bit length
8-bit length	8-bit length	8-bit length	16-bit length	32-bit length
8-bit length	16-bit length			
8-bit length	32-bit length			
16-bit length	16-bit length			
16-bit length	32-bit length			
32-bit length	16-bit length			
32-bit length	32-bit length			

Example



⑤ The following table lists the available data formats for instructions and the related flags.

Instruction		Data format		S: Sign flag (F004E)	Z: Zero flag (F004F)	FE: File empty (F0047)	FF: File full (F0046)	E: Operation error (F0041)	O:Overflow (F0040)
		BCD	Binary						
Arithmetic operation		○	—	○	○	—	—	○	○
Comparison		○	—	—	—	—	—	○	—
Logical operation		○	○	○	○	—	—	○	—
Conversion	BCD	○	○	○	○	—	—	○	○
	BIN	○	○	○	○	—	—	○	○
	CHAR	—	—	—	—	—	—	○	—
	FIG	—	—	—	—	—	—	○	—
	ASCII	—	—	—	—	—	—	○	—
	SEC	—	—	—	—	—	—	○	○
	TIM	—	—	—	—	—	—	○	—
	DECO, ENCO	○	—	○	○	—	—	○	—
	7SEG	○	○	○	○	—	—	○	—
	BCNT	○	○	○	○	—	—	○	—
Transfer	MOV	○	○	—	—	—	—	○	○
	LMOV	○	○	—	—	—	—	○	—
	BT	○	○	—	—	—	—	○	○
	LBT	○	○	—	—	—	—	○	—
	DT	—	—	—	—	—	—	○	—
	MOVU, MOVL	○	○	—	—	—	—	○	—
	PC	○	○	—	—	—	—	○	—
	SRCH	○	○	—	—	—	—	○	—
	SW	○	○	—	—	—	—	○	○
	MSGT	○	○	—	—	—	—	○	—
	MSGR	○	○	—	—	—	—	○	—
File	FLCL	—	—	—	—	ON	OFF	—	—
	SEL	○	○	—	—	—	—	○	○
	DSEL	○	○	—	—	—	—	○	○
	FFST	○	○	—	—	○	○	○	—
	FIFO, FILO	—	—	—	—	○	○	○	○
	FILE	—	—	—	—	—	—	—	—
	TABL	—	—	—	—	—	—	—	—
	DATA, DEND	○	○	—	—	—	—	—	—
	RFIL	○	○	—	—	—	—	○	○
	WFIL	○	○	—	—	—	—	○	○
	FINF	—	—	—	—	—	—	○	○
Program control	PROG, PEND	—	—	—	—	—	—	—	—
	FMC	—	—	—	—	—	—	—	—
	FMS, FME	—	—	—	—	—	—	—	—
	SKIP, SEND	—	—	—	—	—	—	—	—
	DI	○	○	—	—	—	—	○	—
	EI	○	○	—	—	—	—	○	—
	JMP, JEND	—	—	—	—	—	—	—	—
	LOOP, CONT	—	—	—	—	—	—	○	—
	PUSH, POP	—	—	—	—	—	—	○	—
	LEA	—	—	—	—	—	—	○	—
	IADD	○	○	—	—	—	—	○	—
	ISUB	○	○	—	—	—	—	○	—
Analog	Bias	○	○	—	—	—	—	○	○
	FIL	○	○	—	—	—	—	○	○
	Upper limit	○	○	—	—	—	—	○	○
	Upper and lower limit	○	○	—	—	—	—	○	○
	Lower limit	○	○	—	—	—	—	○	○
	INT	○	○	—	—	—	—	○	○
	DIF	○	○	—	—	—	—	○	○
	Dead Band	○	○	—	—	—	—	○	○
	DIVP	○	—	○	○	—	—	○	○
	MLTP	○	—	○	○	—	—	○	○
	HOLD	○	○	—	—	—	—	○	○
Trigonometric function	SIN	○	○	○	○	—	—	○	○
	COS	○	○	○	○	—	—	○	○
	TAN	○	○	○	○	—	—	○	○
	ASIN	○	○	○	○	—	—	○	○
	ACOS	○	○	○	○	—	—	○	○
	ATAN	○	○	○	○	—	—	○	○

○ : Available — : Not available

3-4-4 Arithmetic operations

1. Arithmetic operations

- The 11 types of arithmetic operations (having different functions) are as follows.

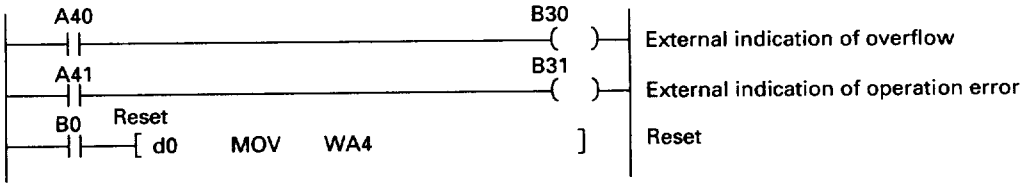
○: Available —: Not available

Instruction	Abbr.	Symbol	Function	F30, F50 F50H	F60	F55, F70, F80H, F120H, F70S, F120S, F140S, F150S
(1) Addition	+	[Z1 + Z2 → Wd]	Z2 is added to Z1 and the sum is stored in Wd.	○	○	○
(2) Subtraction	−	[Z1 − Z2 → Wd]	Z2 is subtracted from Z1 and the remainder is stored in Wd.	○	○	○
(3) Multiplication	x	[Z1 x Z2 → Wd]	Z1 is multiplied by Z2 and the product is stored in Wd.	○	○	○
(4) Division (The remainder is discarded.)	÷	[Z1 ÷ Z2 → Wd]	Z1 is divided by Z2 and the quotient is stored in Wd. (Omit fractional values.)	○	○	○
(5) Division remainder	REM	[Z1 REM Z2 → Wd]	Z1 is divided by Z2 and the remainder is stored in Wd.	—	○	○
(6) Division (rounded to the nearest whole number)	DIVR	[Z1 DIVR Z2 → Wd]	Z1 is divided by Z2 and the quotient is stored in Wd. (rounding up to nearest whole number)	—	○	○
(7) Root (rounded to the nearest whole number)	√‾	[Z √‾ Wd]	The root of Z is obtained and the result is stored in Wd.	—	—	○
(8) Absolute value	ABS	[Z ABS Wd]	The absolute value of Z is obtained and the result is stored in Wd.	—	—	○
(9) Sign invert	+/−	[Z +/− Wd]	The signs (+/−) are inverted and the result is stored in Wd.	—	○	○
(10) Increment	+1	[+1 Wd]	Wd + 1 is stored in Wd.	○	○	○
(11) Decrement	−1	[−1 Wd]	Wd − 1 is stored in Wd.	○	○	○

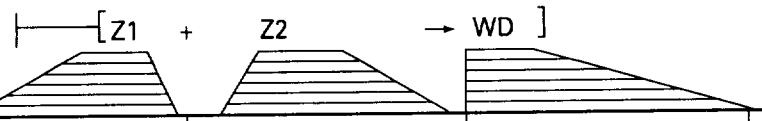
- Data formats and executing operations

(1) All operations are executed using the signed BCD 8-digit format regardless of the number of digits of source data.
(2) If the error flag (A0041) is set ON, operation execution is inhibited. (Example: The source data is not BCD data.)
(3) If the operation result exceeds the capacity of the storage destination, the maximum capacity of the destination is stored and the overflow flag (A0040) is set ON.
(4) If the operation result is a negative value, the sign flag (F004E) is set ON.
(5) If the operation result is zero, the zero flag (F004F) is set ON.
(6) The sign flag (F004E) and the zero flag (F004F) are set ON and OFF for every execution of instructions in a program.
(7) Once the overflow flag (A0040) and the operation error flag (A0041) are set ON, these flags stay ON until power is turned OFF or until these flags are reset by the user program or program loader key operation. Operation continues, ignoring the ON/OFF states of these flags.

Example: Error indication and resetting by user program



The following tables list examples of addition using various data formats and their results with related explanations.



No.	Source (Z1)		Source (Z2)		Destination (Wd)		Sign flag	Zero flag	Overflow flag	Operation error flag
	BCD 4-digit	BCD 8-digit	BCD 4-digit	BCD 8-digit	BCD 4-digit	BCD 8-digit	F004E	F004F	A0040	A0041
①	0123		4567		4690		0	0	0	0
②	0123		4567			00004690	0	0	0	0
③	4000		-7000		-3000		1	0	0	0
④	4000		-4000		0		0	1	0	0
⑤	4000		7000		7999		0	0	1	0
⑥	4000		7000			00011000	0	0	0	0
⑦	3000			12345678		12348678	0	0	0	0
⑧		23000000	6000		7999		0	0	1	0
⑨		23000000	6000			23006000	0	0	0	0
⑩		-12345678		24691356	7999		0	0	1	0
⑪		-12345678		24691356		12345678	0	0	0	0
⑫		40000000		40000000		79999999	0	0	1	0
⑬	AB45		5000		Previous data		0	0	0	1

①	123 + 4567 = 4690 Because the result storage area uses BCD 4-digit format, 4690 is stored.
②	123 + 4567 = 4690 Because the result storage area uses BCD 8-digit format, 4690 is stored and the four high-order digits are padded with zeros (00004690).
③	4000 + (-7000) = -3000 Because the result storage area uses BCD 4-digit format, -3000 is stored and the sign flag (F004E) is set to indicate a negative value.
④	4000 + (-4000) = 0 A result of 0 is stored and the zero flag (F004F) is set.
⑤	4000 + 7000 = 11000 Because the result storage area uses BCD 4-digit format, its maximum value (7999) is stored and the overflow flag (A0040) is set.
⑥	4000 + 7000 = 11000 Because the result storage area uses BCD 8-digit format, 11000 is stored and the three high-order digits are padded with zeros (00011000).
⑦	3000 + 12345678 = 12348678 Because the result storage area uses BCD 8-digit format, 12348678 is stored.
⑧	23000000 + 6000 + 23006000 Because the result storage area uses BCD 4-digit format, its maximum value (7999) is stored and the overflow flag (A0040) is set.
⑨	23000000 + 6000 = 23006000 Because the result storage area uses BCD 8-digit format, 23006000 is stored.
⑩	(-12345678) + 24691356 = 12345678 Because the result storage area uses BCD 4-digit format, its maximum value (7999) is stored and the overflow flag (A0040) is set.
⑪	(-12345678) + 24691356 = 12345678 Because the result storage area uses BCD 8-digit format, 12345678 is stored.
⑫	40000000 + 40000000 = 80000000 Because the result storage area uses BCD 8-digit format, its maximum value (79999999) is stored and the overflow flag (A0040) is set.
⑬	AB45 + 5000 = ? Because the source (Z1) is a hexadecimal number, the operation is not executed. The result storage area retains the previous value and the operation error flag (A0041) is set.

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

2. Addition (+)

Instruction	Addition	F10
Symbol	Unconditional execution ⊢[Z1 + Z2 → Wd] Conditional execution ⊢⊣[Z1 + Z2 → Wd]	
Function	① Z2 is added to Z1 and the result is stored in Wd. ② In case the result exceeds the data range of Wd (±7999 or ±79999999), the overflow flag is set ON and the maximum or minimum value is stored in Wd. ③ In case Z1 or Z2 is not BCD code, the operation error flag is set and the program operation for only that part is not executed. ④ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	

① Z1 (8 digits) + Z2 (8 digits) → Wd (8 digits)

0,0,0,0,1,2,3,4 + 0,0,0,0,5,6,7,8 ⇨ 0,0,0,0,6,9,1,2

Flag: S Z E O = 0 0 0 0

② Z1 + Z2 → Wd

4,0,0,0,0,0,0,0 + 4,0,0,0,0,0,0,0 ⇨ 7,9,9,9,9,9,9,9

Flag: S Z E O = 0 0 0 1

Z1 + Z2 → Wd

0,0,1,2,3,4,5,6 + 0,7,8,9 ⇨ 7,9,9,9

Flag: S Z E O = 0 0 0 1

③ Z1 + Z2 → Wd

4,5,6,E + 0,0,0,0,0,7,8,7 ⇨ Previous data

No arithmetic operation

Flag: S Z E O = — — 1 —

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	S	Z	E	O
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

3. Subtraction (–)

Instruction	Subtraction	F11
Symbol	Unconditional execution ⊢[Z1 – Z2 → Wd] Conditional execution ⊢⊣[Z1 – Z2 → Wd]	
Function	① Z2 is subtracted from Z1 and the result is stored in Wd. ② In case the result exceeds the data range of Wd (±7999 or ±79999999), the overflow flag is set and the maximum or minimum value is stored in Wd. ③ In case Z1 or Z2 is not BCD code, the operation error flag is set and the program operation for only that part is not executed. ④ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	

① Z1 (8 digits) – Z2 (8 digits) → Wd (8 digits)

0,0,0,0,4,0,0,0 – 0,0,0,0,4,0,0,0 ⇨ 0,0,0,0,0,0,0,0

Flag: S Z E O = 0 1 0 0

② Z1 – Z2 → Wd (Minus)

0,0,0,0,4,0,0,0 – 0,0,0,0,8,0,0,0 ⇨ – 0,0,0,0,4,0,0,0

Becomes 80004000 by hexadecimal display.

Flag: S Z E O = 1 0 0 0

③ Z1 – Z2 → Wd

4,0,0,0 – 1,0,F,0 ⇨ Previous data

No arithmetic operation

Flag: S Z E O = — — 1 —

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	S	Z	E	O
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

4. Multiplication (X)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F56	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft ware

Instruction	Multiplication	F12
Symbol	Unconditional execution ⊢[Z_1 x Z_2 → Wd] Conditional execution ⊢⊣[Z_1 x Z_2 → Wd]	
Function	① Z_1 is multiplied by Z_2, and the result is stored in Wd. ② In case the result exceeds the data range of Wd (±7999 or ±79999999), the overflow flag is set ON and the maximum or minimum value is stored in Wd. ③ In case Z_1 is not the BCD code, the operation error flag is set and the program operation for only that part is not executed. ④ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	

① Z_1 (8 digits) [0,0,0,0,0,2,2,2] x Z_2 (8 digits) [0,0,0,0,0,0,1,0] ⇨ Wd (8 digits) [0,0,0,0,0,2,2,2,0]

Flag

S	Z	E	O
0	0	0	0

② Z_1 [0,0,0,3,4,5,6,7] x Z_2 [0,0,0,0,4,0,0,0] ⇨ Minus Wd [7,9,9,9,9,9,9,9]

Flag

S	Z	E	O
1	0	0	1

Z_1 [0,0,0,0,4,0,0,0] x Z_2 [1,0,0,0] ⇨ Wd [7,9,9,9]

S	Z	E	O
0	0	0	1

③ Z_1 [0,1,E,8] x Z_2 [0,0,0,0,0,0,2,0] ⇨ Wd [Previous data]

No arithmetic operation

Flag

S	Z	E	O
—	—	1	—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z_1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	S	Z	E	O
Z_2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨: Applicable)												Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F195S	D05	D10S	D20	LITE	Software

5. Division (÷)

Instruction	Division	F13
Symbol	Unconditional execution ⊢[Z1 ÷ Z2 → Wd] Conditional execution ⊢⊣[Z1 ÷ Z2 → Wd]	
Function	① Z1 is divided by Z2, and the quotient is stored in Wd. (Omit fractional values.) ② In case the result exceeds the data range of Wd (±7999 or ±79999999), the overflow flag is set ON and the maximum or minimum value is stored in Wd. ③ In case the divisor is 0, the operation error flag is set and the program operation for only that part is not executed. ④ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	

① Z1 (8 digits) 0,0,0,0,3,9,4,0 ÷ Z2 (8 digits) 0,0,0,0,0,0,0,5 ⇨ Wd (8 digits) 0,0,0,0,0,7,8,8

Flag

S	Z	E	O
0	0	0	0

② Z1 0,7,9,9,9,9,9,9 ÷ Z2 0,0,0,5 ⇨ Wd 7,9,9,9

Flag

S	Z	E	O
0	0	0	1

③ Z1 4,5,6,7 ÷ Z2 0,0,0,0,0,0,0,F ⇨ Wd Previous data

No arithmetic operation

Flag

S	Z	E	O
—	—	1	—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

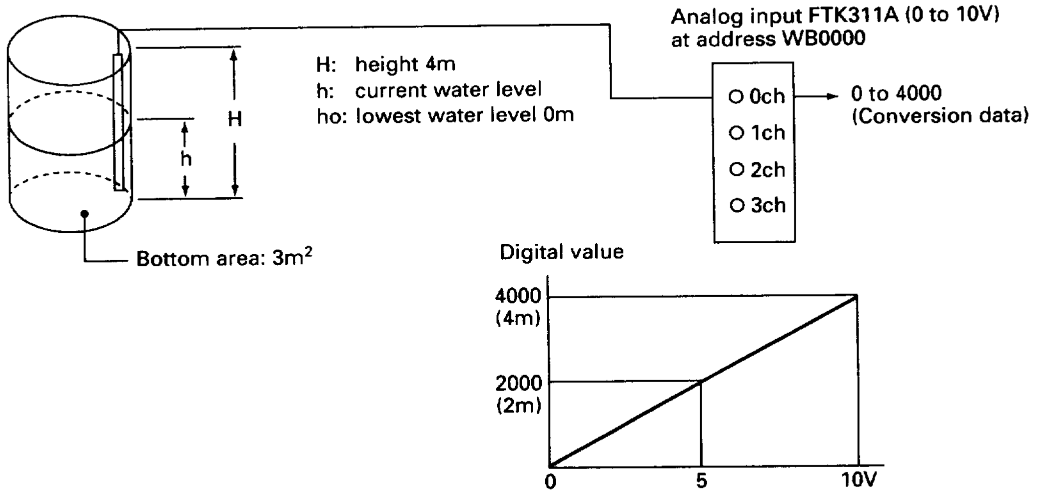
ONE-POINT ADVICE   Tank-water amount calculation circuit

An example of using multiplication and division is shown below.

1. **Instruction**
   X: multiplication

2. **Operation**
   Amount of water in tank is calculated according to input value (height) from water level gauge.

3. **System diagram**



4. **Program**

```
 F54
─┤├─┬ d3      x   WB0     ─→ BD0      ]   Bottom area (m²) x height (mm)
    └ BD0     ÷   d1000   ─→ BD1          Water volume converted to m³
```

Usually, BD0 is used to store the result.
But, BD1 is used here for ease of explanation.

- Example of program and data monitoring with loader LITE

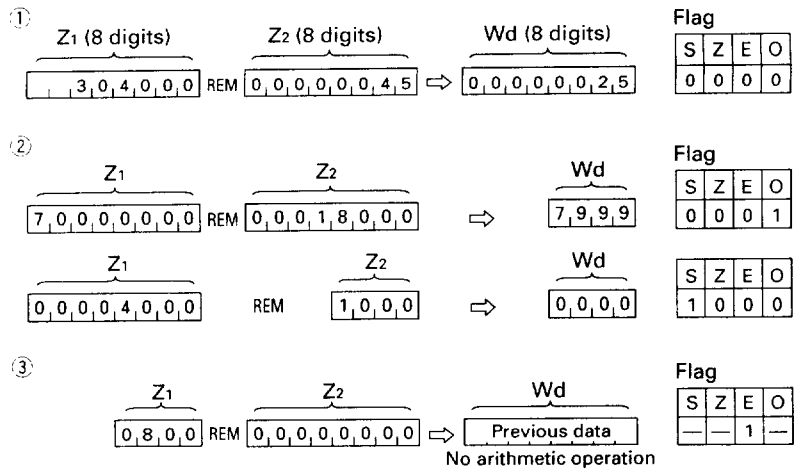
```
P0001|F0054                          00001080        00003240                   |
 0001├──┤├──[  d 00000003    x     WB0000     ──→ BD0000              ]         |
P0001|     [ 00003240                                00000003              [         |
 0002├     [ BD0000          ÷     d 00001000 ──→ BD0001              ]         |
```

```
ADDRESS  BIN             DEC        HEX       BCD
          ↓   ↓   ↓   ↓
 BD0000                    *          * 00003240  00003240  ←── Multiplication result 3240 (m²·mm)
 BD0001                    *          * 00000003  00000003  ←── 3 (m³) with m³ conversion
 BD0002                    *          * 00000000  00000000      (fraction is discarded).
 BD0003                    *          * 00000000  00000000
 BD0004                    *          * 00000000  00000000
 BD0005                    *          * 00000000  00000000
```

6. Division remainder (REM)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Division remainder	F20
Symbol	Unconditional execution ⊢[Z1 REM Z2 → Wd] Conditional execution ⊢⊣[Z1 REM Z2 → Wd]	
Function	① Z1 is divided by Z2, and the remainder is stored in Wd. ② In case the result exceeds the data range of Wd (±7999 or ±79999999), the overflow flag is set ON and the maximum or minimum value is stored in Wd. ③ In case the divisor is 0, the operation error flag is set and the program operation for only that part is not executed. ④ In F60 series, W (file) cannot be specified for operand.	



Operand and influence flag

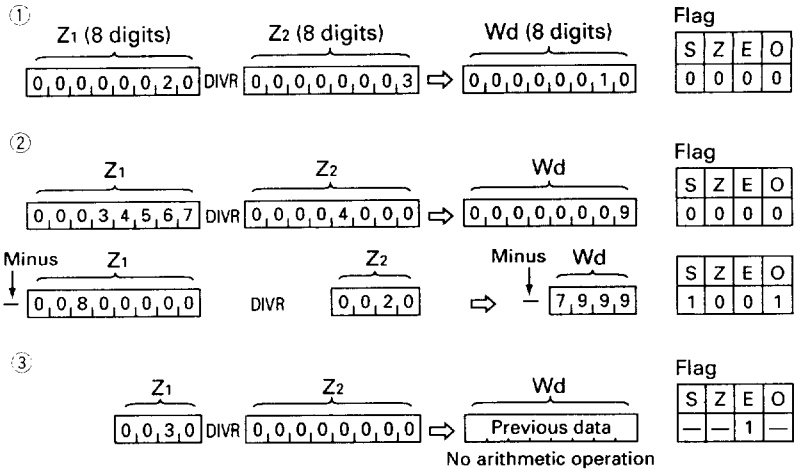
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

7. Division (Round to the nearest whole number) (DIVR)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F156S	D05	D10S	D20	LITE	Soft-ware

Instruction	Division (Round to the nearest whole number)	F14
Symbol	Unconditional execution ⊢——[Z1 DIVR Z2 → Wd] Conditional execution ⊢⊣⊢[Z1 DIVR Z2 → Wd]	
Function	① Z1 is divided by Z2, and the quotient is stored in Wd. (Rounding off to the the whole number) ② In case the result exceeds the data range of Wd (±7999 or ±79999999), the overflow flag is set and the maximum or minimum value is stored in Wd. ③ In case the divisor is 0, the logical operation error flag is set and the program operation for only that part is not executed. ④ In F60 series, W (file) cannot be specified for operand.	



Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

8. Root (√‾)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft. ware

Instruction	Root	F15	Description	Example
Symbol	Unconditional execution ⊢[Z √‾ Wd] Conditional execution ⊢⊣[Z √‾ Wd]		Input range: BD 0 to 79999999 (A negative value causes an operation error.) WB 0 to 7999 Output range: Integers only	⊢⊣[BD0000 √‾ BD0001] BD0000 [12345678] For the data shown on the left, operation result is as follows. BD0001 [3514] $\sqrt{12345678} \doteqdot 3513.6$
Function	① The root of Z is obtained and the result is stored in Wd. ② In F60 series, W (file) cannot be specified for operand.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

9. Absolute value (ABS)

Instruction	Absolute value	F16	Description	Example
Symbol	Unconditional execution ⊢—[Z ABS Wd] Conditional execution ⊢⊣[Z ABS Wd]			⊢⊣[BD0000 ABS BD0001] BD0000 -112233 For the data shown on the left, operation result is as follows. BD0001 112233
Function	① The absolute value of Z is stored in Wd. If Z is the maximum negative value, the overflow flag is set ON and its maximum positive value is stored as the operation result. ② In F60 series, W (file) cannot be specified for operand.			

Operand and influence flag

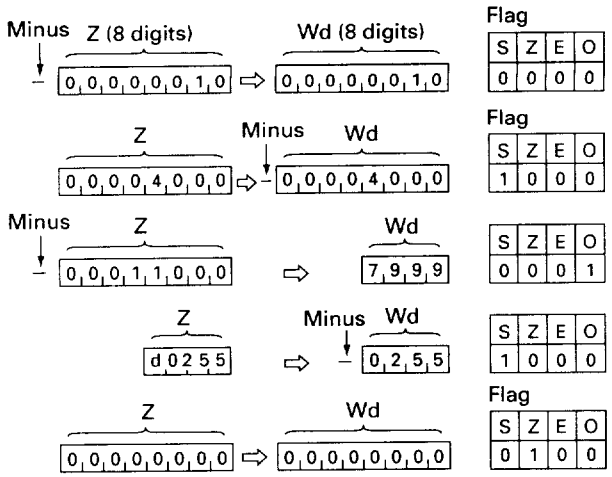
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	↕	↕	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

10. Sign invert (+/–)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Sign invert	F17
Symbol	Unconditional execution ⊢—[Z +/- Wd] Conditional execution ⊢⊣[Z +/- Wd]	
Function	① The sign (+/-) of Z is inverted and the result is stored in Wd. ② In F60 series, W (file) cannot be specified for operand.	



Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	↕	↕	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

11. Increment (+1)

Processor (▭: Applicable)												Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S		D05	D10S	D20	LITE	Soft-ware

Instruction	Increment	F18
Symbol	Unconditional execution ⊢[+1 Wd] Conditional execution ⊢⊢[+1 Wd]	
Function	① One is added to Wd and the result is stored in Wd. ② In case the result exceeds the data range of Wd (+7999 or +79999999), the overflow flag is set ON, and the maximum or minimum value is stored in Wd. ③ When a differential instruction is not used as a conditional contact, the instruction is executed per each scan. ④ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	

① Wd 0,0,0,0,0,0,9,9 + 1 ⇨ Wd 0,0,0,0,0,1,0,0

Flag: S Z E O = 0 0 0 0

② Wd 7,9,9,9,9,9,9,9 + 1 ⇨ Wd 7,9,9,9,9,9,9,9

Flag: S Z E O = 0 0 0 1

Minus Wd −0,0,0,0,0,2,0,0 + 1 ⇨ Minus Wd −0,0,0,0,0,1,9,9

Becomes 80000199 by hexadecimal display.

Flag: S Z E O = 1 0 0 0

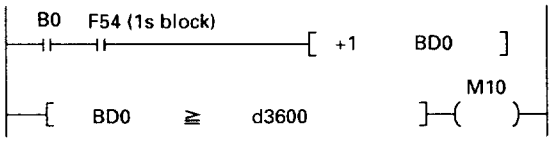
Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
																												↕	↕	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Timer circuit example using increment instruction
Increment instruction can be used in place of timer instruction.

(where time base is 1s.)

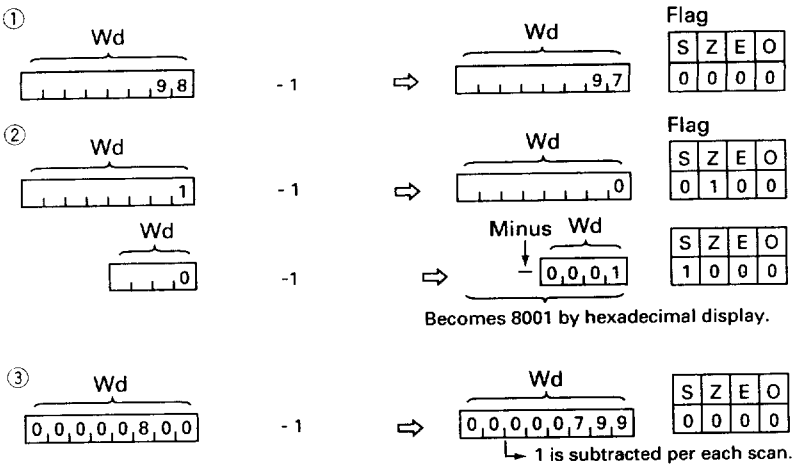


(Operation)
While B0 is ON, counting continues. M10 is set ON after 3600 seconds (1 hour) elapses.

12. Decrement (-1)

Processor (▨ : Applicable)												Program loader				
F3D	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F195S	D05	D10S	D20	LITE	Soft-ware

Instruction	Decrement	F19
Symbol	Unconditional execution ⊢[-1 Wd] Conditional execution ⊢||⊢[-1 Wd]	
Function	① One is subtracted from Wd and the result is stored in Wd. ② In case the result exceeds the data range of Wd (–7999 or –79999999), the overflow flag is set ON, and the maximum or minimum value is stored in Wd. ③ When a differential instruction is not used as a conditional contact, the instruction is executed per each scan. ④ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	



Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
																												↕	↕	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

3-4-5 Comparison

1. Comparison

- The seven types of comparison instructions (having different functions) are as follows. (File comparison instruction is not supported by F30, F50, F50H, and F60 series.)

Instruction	Symbol	Function
(1) Comparison (>)	$Z_1 > Z_2$	Z_2 is subtracted from Z_1. If the result is a positive number, the coil is set ON.
(2) Comparison (≥)	$Z_1 \geq Z_2$	$Z_2$is subtracted from Z_1. If the result is a positive number or zero, the coil is set ON.
(3) Comparison (=)	$Z_1 = Z_2$	Z_2 is subtracted from Z_1. If the result is zero, the coil is set ON.
(4) Comparison (≠)	$Z_1 \neq Z_2$	Z_2 is subtracted from Z_1. If the result is a nonzero value, the coil is set ON.
(5) Comparison (<)	$Z_1 < Z_2$	Z_2 is subtracted from Z_1. If the result is a negative number, the coil is set ON.
(6) Comparison (≤)	$Z_1 \leq Z_2$	Z_2 is subtracted from Z_1. If the result is a negative number or zero, the coil is set ON.
(7) File comparison (REF)	Ws_1 REF Ws_2: Z : N	The Z words at data address Ws_1 are compared with those at data address Ws_2. If condition N is satisfied, the coil is set ON.

- Data formats and executing comparison instructions

① A hexadecimal direct value (h□□□□) cannot be accepted as source data.

② Comparison instructions have no affect on flag relays (overflow flag, sign flag and zero flag).

③ Comparison between positive numbers (The most-significant bits are 0.)
For execution, the data having the greater absolute value is handled as the larger data value.

Example: 7FFF > 00FF

④ Comparison between negative numbers (the most-significant bits are 1.)
Because BCD data is the main object of this instruction, it is assumed that 80000000 is the largest value and FFFFFFFF is the smallest value used for execution.

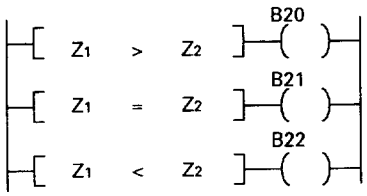
Example: 8001 > FFFF

Note:
The comparison result using BCD data is reversed in the comparison result using converted decimal data.

⑤ Comparison between a positive number and negative number
Comparison is executed, assuming that the positive number is larger than the negative number.

	Data format and range			Converted decimal value
	8-bit area	16-bit area	32-bit area	
↑ Larger			7FFFFFFF	2147483647
			7FFFFFFE	2147483646
			⋮	⋮
			00008000	32768
Range of positive numbers		7FFF	00007FFF	32767
		7FFE	00007FFE	32766
		⋮	⋮	⋮
		0100	00000100	256
↓ Smaller	FF	00FF	000000FF	255
	FE	00FE	000000FE	254
	⋮	⋮	⋮	⋮
	01	0001	00000001	
	00	0000	00000000	0
↑ Larger			80000000	-2147483648
			80000001	-2147483647
			⋮	⋮
			FFFF7FFF	-32769
Range of negative numbers		8000	FFFF8000	-32768
		8001	FFFF8001	-32767
		⋮	⋮	⋮
↓ Smaller		FFFE	FFFFFFFE	-2
		FFFF	FFFFFFFF	-1

The following table lists the results of comparisons between various data formats.



	Source (Z1)		Source (Z2)		B20 -( )-	B21 -( )-	B22 -( )-
	16-bit area	32-bit area	16-bit area	32-bit area			
①	0007		0003		ON	OFF	OFF
②	0007		008F		OFF	OFF	ON
③	00FF		00FF		OFF	ON	OFF
④	8001		F999		ON	OFF	OFF
⑤	8000		FFFF		ON	OFF	OFF
⑥		7FFFFFFF		FFFFFFFF	ON	OFF	OFF
⑦		79999999		F9999999	ON	OFF	OFF

• Explanation

①	Because 0007 is larger than 0003, B20 is set ON.
②	Because 0007 is smaller than 008F, B22 is set ON.
③	Because 00FF is equivalent to 00FF, B21 is set ON.
④	Negative numbers are compared. Because 8001 is considered larger than F999, B20 is set ON.
⑤	Negative numbers are compared. Because 8000 is considered larger than FFFF, B20 is set ON.
⑥	7FFFFFFF is a positive number and FFFFFFFF is a negative number. Because 7FFFFFFF is larger than FFFFFFFF, B20 is set ON.
⑦	79999999 is a positive number and F9999999 is a negative number. Because 79999999 is larger than F9999999, B20 is set ON.

■ Output relay following comparison instruction
Identifiers which can be used for output relays placed after comparison instructions are as follows.

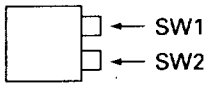
Identifier	B	M	K	D	F	A	S	T	C	i	j	k	l	m	P	Q
Availability	○*	○	○	—	—	○	—	—	—	○	○	○	—	—	○	○

○: Available — :Not available

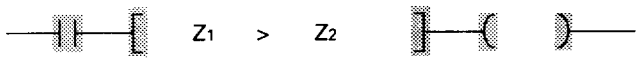
When P or PE-link is used, L can also be used.
*: B is not usable for input address.

■ The operation of the output coil of comparison instruction and operation setting switch (F70S, F120S, F140S, and F150S only)

• Operation setting switch
The operation setting switch on the processor module front determines the operation of the output coil of comparison instructions as follows:



SW1 setting
This switch is used to set the coil operation when the conditional contact is OFF in the comparison instruction with contact and coil.



Conditional contact

If the conditional contact is set OFF in the above status, the following status results.

SW1	Instruction	Output coil status
OFF (new mode)	Z1 > Z2	OFF
ON (conventional mode)	Z1 > Z2	Preceding value is held.

SW2 setting
This switch is used to set the operation of comparison instructions "+0" and "-0."

SW1	Instruction	Output coil status
OFF (new mode)	d0 = -d0	Mismatch
ON (conventional mode)	d0 = -d0	Match

Notes: 1. In the F30 to F120H Series processors, and comparison instructions operate in conventional mode. In the F70S, and F120S to F150S Series processors, comparison instructions operate in new mode.

2. Both SW1 and SW2 are factory-set to ON (Conventional mode).

2. Comparison >, ≥

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft ware

Instruction	Comparison >, ≥	F30, F31
Symbol	Unconditional execution: Z1 > Z2 (≥) — Output relay Conditional execution: Z1 > Z2 (≥) — Output relay	
Function	① The comparison between Z1 and Z2 is carried out, and the result is output to the output relay as follows. Condition is satisfied: Output relay is turned ON. Condition is not satisfied: Output relay is turned OFF. ② The highest order bit is used as a sign, and all others are compared as bit data. ③ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	

① Comparison between positive numbers

Z1 (4 digits) Z2 (8 digits)

6,6,6,6 > 0,0,0,0,5,5,5,5 ⇨ ON

Z1 Z2

0,0,0,0,6,6,6,6 > 0,0,0,0,7,F,F,F ⇨ OFF

② Comparison between negative numbers

Z1 Z2

8,0,0,0,F,F,F,F > F,F,F,F ⇨ OFF

③ Comparison between a positive number and negative number

Z1 Z2

8,9,A,B,C,D,E,F > 0,1,2,3,4,5,A,B ⇨ OFF

Negative number Positive number

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	S	Z	E	O
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

3. Comparison =, ≠

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft ware

Instruction	Comparison =, ≠	F32, F35
Symbol	Unconditional execution: Z1 = Z2 (≠) — Output relay Conditional execution: Z1 = Z2 (≠) — Output relay	
Function	① The comparison between Z1 and Z2 is carried out, and the result is output to the output relay as follows. Condition is satisfied: Output relay is turned ON. Condition is not satisfied: Output relay is turned OFF. ② The highest order bit is used as a sign, and all others are compared as bit data. ③ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	

① Comparison between positive numbers

Z1 (4 digits) Z2 (8 digits)

6,6,6,6 > 0,0,0,0,5,5,5,5 ⇨ ON

Z1 Z2

0,0,0,0,6,6,6,6 > 0,0,0,0,7,7,7,7 ⇨ OFF

② Comparison between negative numbers

Z1 Z2

9,A,B,C,D,E,F,F = 8,0,0,0,0,2,3,4 ⇨ OFF

③ Comparison between a positive number and negative number

Z1 Z2

2,3,A,B = F,F,F,F,F,F,F,F ⇨ OFF

Positive number Negative number

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	S	Z	E	O
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

4. Comparison <, ≤

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

Instruction	Comparison <, ≤	F34, F33
Symbol	Unconditional execution: Z1 < Z2 (≤) — Output relay Conditional execution: Z1 < Z2 (≤) — Output relay	
Function	① The comparison between Z1 and Z2 is carried out, and the result is output to the output relay as follows. Condition is satisfied: Output relay is turned ON. Condition is not satisfied: Output relay is turned OFF. ② The highest order bit is used as a sign, and all others are compared as bit data. ③ In F30, F50, F50H, and F60 series, W (file) cannot be specified for operand.	

① Comparison between positive numbers

Z1 (4 digits) Z2 (8 digits)

6,6,6,6 < 0,0,0,0,7,5,5,5 ⇨ ON

Z1 Z2

0,0,0,0,6,6,6,6 < 0,0,0,0,5,B,C,D ⇨ OFF

② Comparison between negative numbers

Z1 Z2

A,B,C,D,E,F,F,F < F,E,D,C,B,A,9,8 ⇨ OFF

③ Comparison between a positive number and negative number

Z1 Z2

1,2,3,4,5,6,7,8 < 9,A,B,C,D,E,F,0 ⇨ OFF

Positive number Negative number

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	S	Z	E	O
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

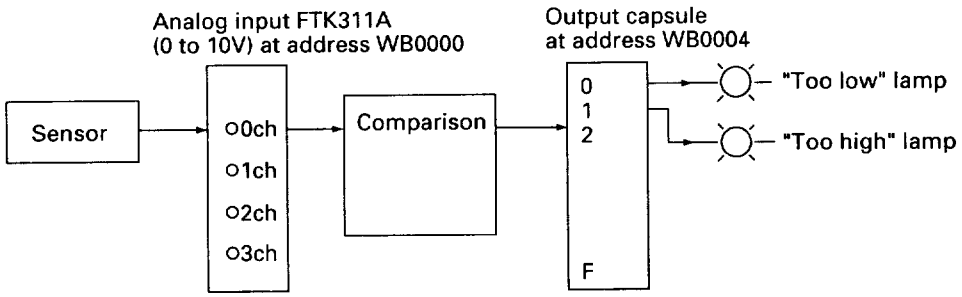
ONE-POINT ADVICE **Analog input voltage comparison circuit**

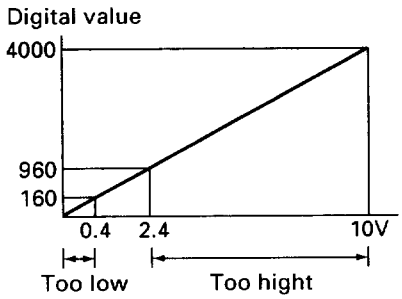
An example of using comparison instructions (<=, =>) is shown below.

1. **Instruction**
   <=, >=

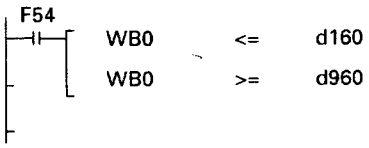
2. **Operation**
   When input voltage from the sensor is below 0.4V, the "Too low" lamp is lit, and when the voltage is above 2.4V, the "Too high" lamp is lit.

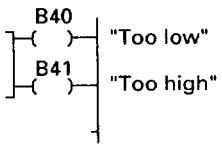
3. **System diagram**





4. **Program**





• Example of monitoring with loader LITE

```
P0001|F0054    00001012
 0001├──┤├──[ WB0000      <=   d 00000160
P0001|      |  00001012
 0002├       [ WB0000      >=   d 00000960
```

```
 B0040   |
]-(   )--|
 B0041   |
]-|   |--|
```

```
  ADDRESS  BIN                DEC         HEX       BCD
           ▼  ▼  ▼  ▼
    B000   0001000000010010   0000004114  00001012  00001012
    B001   0000000000000000   0000000000  00000000  00000000
    B002   0000000000000000   0000000000  00000000  00000000
    B003   0000000000000000   0000000000  00000000  00000000
    B004   0100000000000000   0000016384  00004000  00004000
    B005   0000000000000000   0000000000  00000000  00000000
```

B41 ("Too high" indication) is set ON because input data 1012 is larger than 960.

5. File comparison (REF)

Processor (▭: Applicable)												Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F160S	D05	D10S	D20	LITE	Soft-ware

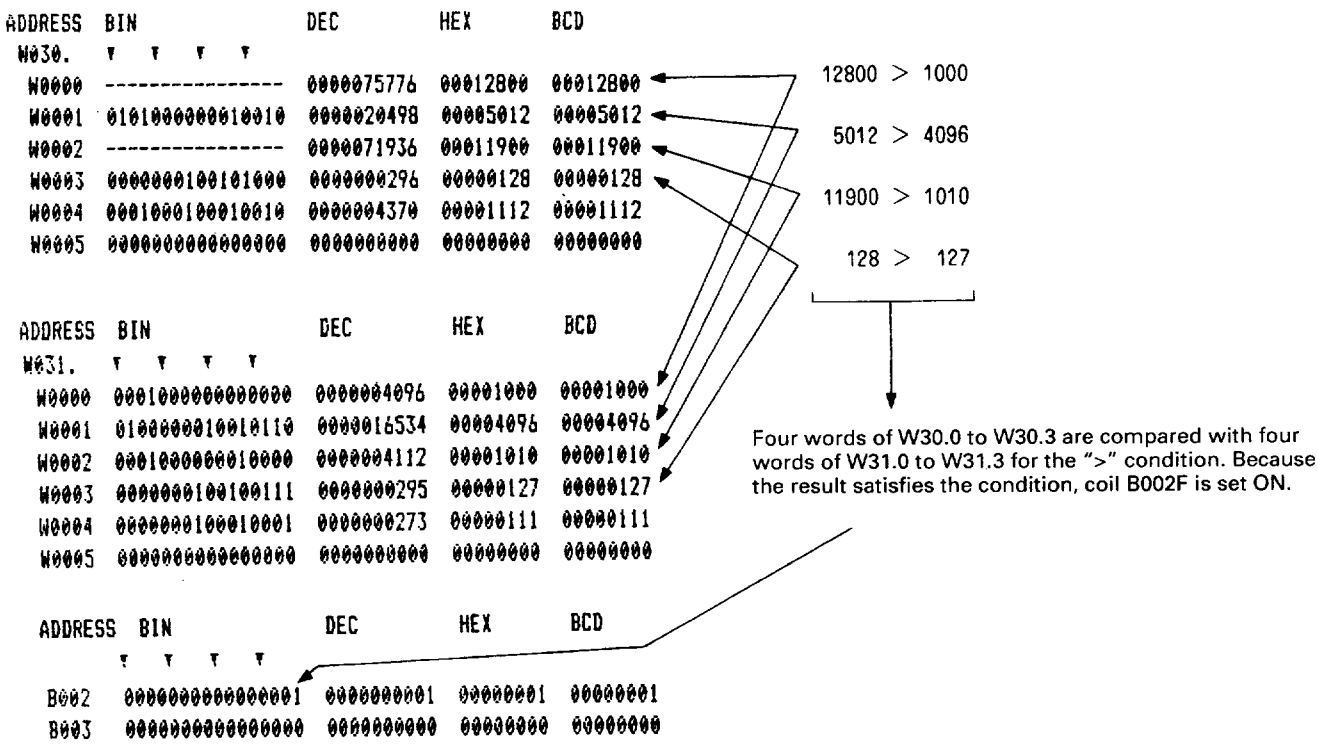
Instruction	File comparison	F36	Description	Example
Symbol	⊢⊣⊢[Ws1, REF Ws2 : Z ; N]⊢( )⊢ Ws1: First address of comparison data 1 Ws2: First address of comparison data 2 Z: Number of words to be compared N: Condition (<, >, ≦, ≧, =, ≠)		① Conditions N is specified by using numbers. N (Condition) Number > → 0 ≧ → 1 = → 2 < → 3 ≦ → 4 ≠ → 5 ② If Z is specified by using a word address, an operation error occurs and the comparison is not executed except for BCD code.	B001E B011F ⊢⊣⊢[BD0000 REF BD0010 : d5 : 0]⊢( )⊢ Condition BD0000 128000 > BD0010 1000 BD0001 5012 > BD0011 4096 BD0002 11900 > BD0012 10010 BD0003 128 > BD0013 127 BD0004 1112 > BD0014 1111 Five words of BD0000 to BD0004 are compared with five words of BD0010 to BD0014 for the ">" condition. In this example, the condition is satisfied and relay B011F is set ON.
Function	① The Z words beginning at address Ws1 are compared with those beginning at address Ws2. If the result of each comparison satisfies the condition, the output relay is set ON. Ws1 Ws2 Comparison			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Ws1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Ws2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

• Example of monitoring file comparison



3-4-6 Logical operations

1. Logical operations

• The nine types of logical operations (having different functions) are as follows.

○: Available —: Not availabe

Instruction	Abbr.	Symbol	Function	F30, F50, F50H, F60	F55, F70, F80H, F120H F70S, F120S, F140S, F150S
(1) Logical product	AND	⊣[Z1 AND Z2 → Wd]⊢	The logical product of Z1 and Z2 is obtained and the result is stored in Wd.	○	○
(2) Logical add	OR	⊣[Z1 OR Z2 → Wd]⊢	The logical add of Z1 and Z2 is obtained and the result is stored in Wd.	○	○
(3) Exclusive OR	EOR	⊣[Z1 EOR Z2 → Wd]⊢	The exclusive OR of Z1 and Z2 is obtained and the result is stored in Wd.	○	○
(4) Invert	INV	⊣[Z INV Wd]⊢	The logic of each bit of Z is inverted (0→1, 1→0) and the result is stored in Wd.	○	○
(5) Shift right logical	SRL	⊣[Ws SRL Wd : Z]⊢	The contents of Ws are shifted right by Z bits and are stored in Wd.	○	○
(6) Shift left logical	SLL	⊣[Ws SLL Wd : Z]⊢	The contents of Ws are shifted left by Z bits and are stored in Wd.	○	○
(7) Set bit	SBIT	⊣[Ws SBIT Wd : Z]⊢	The Zth bit of Ws is set to 1 and is stored in Wd.	—	○
(8) Reset bit	RBIT	⊣[Ws RBIT Wd : Z]⊢	The Zth bit of Ws is set to 0 and is stored in Wd.	—	○
(9) Test bit	TBIT	⊣[Ws TBIT Z]( )⊢	The Zth bit of Ws is checked; if the data is 1, the output relay is ON, if 0, it is OFF.	—	○

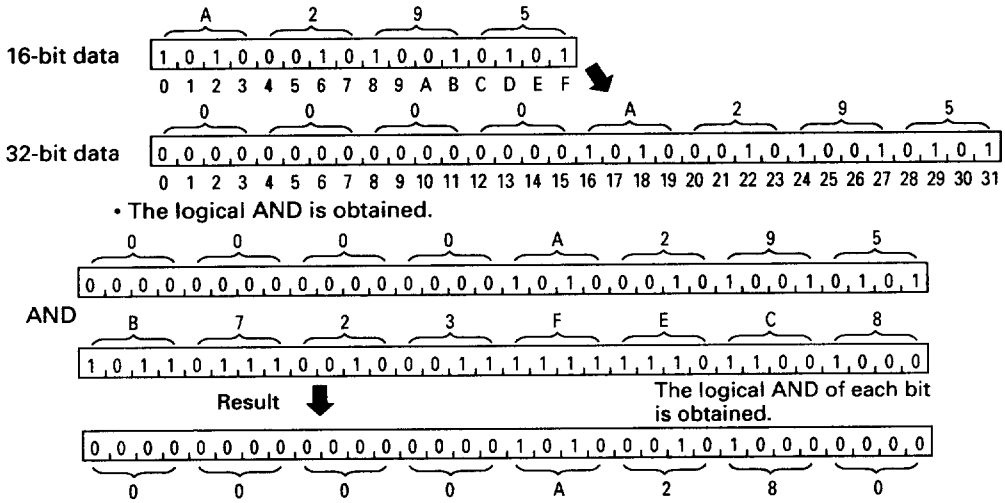
• Data formats and executing operations

① Logical operations are executed on a bit-to-bit basis.

② If two data items have different lengths, the high-order bits of the shorter data are automatically padded with zeros to make the two lengths the same.

Example: Obtaining the logical AND of 16-bit data (A295) and 32-bit data (B723FEC8)

• The 16 high-order bits are padded with zeros to make a 32-bit data.



• The logical AND is obtained.

③ If the most-significant bit of the result is 1, the sign flag (F004E) is set.

④ If all bits of the result are 0, the zero flag (F004F) is set.

■ Output relay for the test bit instruction

Identifiers usable for the output relay subsequent to the test bit instruction are as follows. When P or PE-link is used, identifier L can also be used. The operation of output relay depends on the operation setting switch (page 3-66).

Identifier	B	M	K	D	F	A	S	T	C	i	j	k	ℓ	m	P	Q
Availability	○*	○	○	—	—	○	—	—	—	○	○	○	—	—	○	○

○ : Available
— : Not available

* B is not usable for input address.

2. Logical product (AND)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	AND	F50
Symbol	Unconditional execution ⊢[Z1 AND Z2 → Wd] Conditional execution ⊢⊣[Z1 AND Z2 → Wd]	
Function	① Logical product of Z1 and Z2 is executed, and the result is stored in Wd. Logic for each bit becomes as (Input) A, B — AND — OUT (Output) ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

A	B	OUT
0	0	0
0	1	0
1	0	0
1	1	1

① Z1: 0,0,0,0,4,5,6,7 AND Z2: h,F,F ⇨ Wd: 0,0,0,0,0,0,6,7 — Flag: S=0, Z=0, E=—, O=—

② Z1: 0,0,0,0,0,0,0,0 AND Z2: 0,0,0,0,6,9,9,9 ⇨ Wd: 0,0,0,0,0,0,0,0 — Flag: S=0, Z=1, E=—, O=—

Z1: 0,0,0,0,8,0,0,0 AND Z2: 8,0,0,0 ⇨ Wd: 8,0,0,0 — Flag: S=1, Z=0, E=—, O=—

③ Z1: 4,5,6,7 AND Z2: h,0,F,F,F,F,F,F ⇨ Wd: 0,0,0,0,4,5,6,7 — Flag: S=0, Z=0, E=—, O=—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag S	Z	E	O
Z1	○	○	○	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	○	○	↕	↕	↕	–
Z2	○	○	○	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	○	○	↕	↕	↕	–
Wd	○*	○	○	–	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

3. Logical add (OR)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	OR	F51
Symbol	Unconditional execution ⊢[Z1 OR Z2 → Wd] Conditional execution ⊢⊣[Z1 OR Z2 → Wd]	
Function	① Logical product of Z1 and Z2 is executed, and the result is stored in Wd. Logic for each bit becomes as follows (Input) A, B — OR — OUT (Output) ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

A	B	OUT
0	0	0
0	1	1
1	0	1
1	1	1

① Z1: 0,0,0,0,1,2,0,0 OR Z2: 0,0,0,0,0,0,3,4 ⇨ Wd: 0,0,0,0,1,2,3,4 — Flag: S=0, Z=0, E=—, O=—

② Z1: 0,0,0,2,3,4,5,6 OR Z2: 8,0,0,0,2,0,0,0 ⇨ Wd: 8,0,0,2,3,4,5,6 — Flag: S=1, Z=0, E=—, O=—

Z1: 0,0,1,2,3,4,5,6 OR Z2: 2,0,0,0 ⇨ Wd: 3,4,5,6 — Flag: S=0, Z=0, E=—, O=—

③ Z1 (Hexadecimal): 0,F,0,F OR Z2 (Hexadecimal): 0,0,0,2,3,4,5,6 ⇨ Wd (Hexadecimal): 0,0,0,2,3,F,5,F — Flag: S=0, Z=0, E=—, O=—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag S	Z	E	O
Z1	○	○	○	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	○	○	↕	↕	↕	–
Z2	○	○	○	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	○	○	↕	↕	↕	–
Wd	○*	○	○	–	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

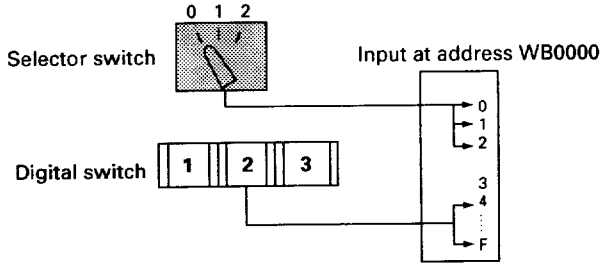
ONE-POINT ADVICE **Data masking circuit**

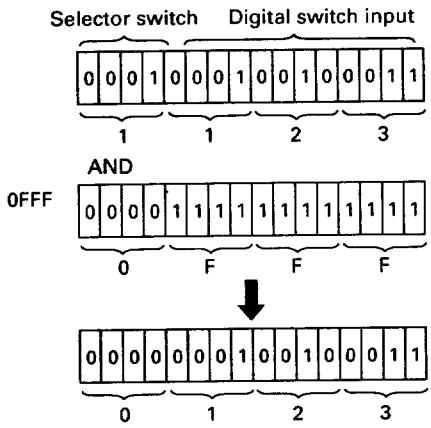
An example of using AND is shown below.

1. Instruction
AND

2. Operation
When there are inputs from both digital switch and pushbutton, only the data from digital switch is acquired.

3. System diagram

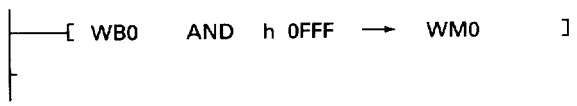




The AND instruction serves as a data filter as shown in the diagram at left. Namely, by ANDing the '0's at the unnecessary bit positions, the result becomes '0'.

⇨ This is called masking.
⇨ Be careful about digit position 1: (10^2), 2: (10^1) and 3: (10^0)

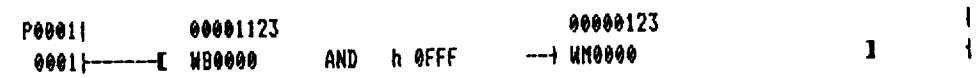
4. Program



• The logical AND of data at addresses WB0000 and h0FFF is obtained and the result is stored in WM0.

• Example of monitoring with loader LITE

① Program monitoring



② Data monitoring

ADDRESS	BIN	DEC	HEX	BCD
	▼ ▼ ▼ ▼			
B000	0001000100100011	0000004387	00001123	00001123
B001	0000000000000000	0000000000	00000000	00000000
B002	0000000000000000	0000000000	00000000	00000000
B003	0000000000000000	0000000000	00000000	00000000
M000	0000000100100011	0000000291	00000123	00000123
M001	0000000000000000	0000000000	00000000	00000000
M002	0000000000000000	0000000000	00000000	00000000
M003	0000000000000000	0000000000	00000000	00000000

The MSB 1 of 1123 is masked and output as 123.

4. Exclusive OR (EOR)

Processor (▨: Applicable)												Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Exclusive OR	F52
Symbol	Unconditional execution ⊢[Z1 EOR Z2 → Wd] Conditional execution ⊢⊣[Z1 EOR Z2 → Wd]	
Function	① Logical product of Z1 and Z2 is executed, and the result is stored in Wd. Logic for each bit becomes as follows. (Input) A, B — OUT (Output) A B OUT: 0 0 0; 0 1 1; 1 0 1; 1 1 0 ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

Z1 side 0 0 1 1 1 1 0 0 1 1 0 0 0 0 0 0 — 3 C C 0 ← Hexadecimal

Z2 side 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 — 0 F 0 F ← Hexadecimal

⇩ EOR

Wd side 0 0 1 1 0 0 1 1 1 1 0 0 1 1 1 1 — 3 3 C F ← Hexadecimal

Flag: S Z E O = 0 0 — —

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	↕	↕	↑	–
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	↕	↕	↑	–
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

5. Invert (INV)

Processor (▨: Applicable)												Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Invert	F53
Symbol	Unconditional execution ⊢[Z INV Wd] Conditional execution ⊢⊣[Z INV Wd]	
Function	① The logic for each bit in the word is inverted (0→1 and 1→0), and the result is stored in Wd. (Input) A — OUT (Output) A OUT: 1 0; 0 1 ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

① Inverseion of 16 bit data

0 1 0 1 1 1 0 1 0 1 1 0 1 0 1 0

⇩ Invert

1 0 1 0 0 0 1 0 1 0 0 1 0 1 0 1

Flag: S Z E O = 1 0 — —

② Inverseion of 32 bit data

1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0

⇩ Invert

0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1

Flag: S Z E O = 0 0 — —

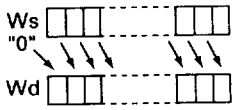
Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	↕	↕	↑	–
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

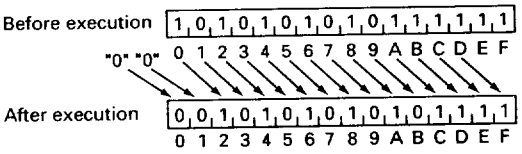
* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

6. Shift right logical (SRL)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F90H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Shift right logical	F54
Symbol	Unconditional execution ⊢[Ws SRL Wd: Z] Conditional execution ⊢⊣[Ws SRL Wd: Z]	
Function	① The contents of Ws are shifted right by Z bits and are stored in Wd. "0"s are stored in the vacant positions. Z cannot exceed the numerical ② value of the bit length of Ws, or cannot be a negative value. Z: d1 to d31 h1 to h1F  ③ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

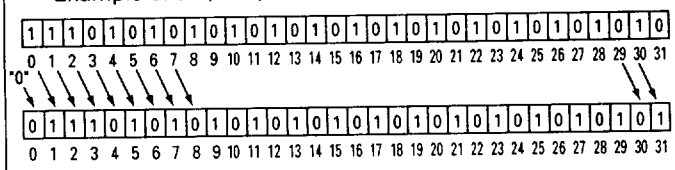
① In case of 16-bit data
Example of d2 (2 bits) shift



Flag

S	Z	E	O
0	0	—	—

② In case of 32-bit data
Example of d1 (1 bit) shift



Flag

S	Z	E	O
0	0	—	—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	↕	↕	↑	–
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

7. Shift left logical (SLL)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F356S	D05	D10S	D20	LITE	Soft-ware

Instruction	Shift left logical	F55

Symbol

Unconditional execution

```
|——[ Ws  SLL  Wd: Z ]
```

Conditional execution

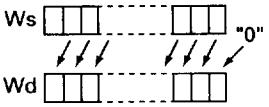
```
|—|—[ Ws  SLL  Wd: Z ]
```

Function

① The contents of Ws are shifted left by Z bits and are stored in Wd. "0"s are stored in the vacant positions.

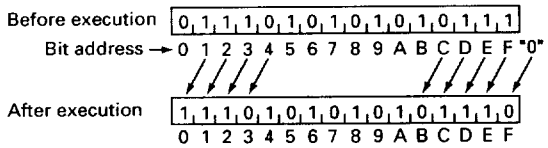
② Z cannot exceed the numerical value of the bit length of Ws, or cannot be a negative value.

Z: d1 to d31
h1 to h1F



③ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.

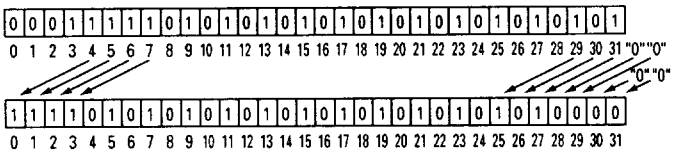
① In case of 16-bit data
Example of d1 (1 bit) shift



Flag

S	Z	E	O
1	0	—	—

② In case of 32-bit data
Example of d4 (4 bits) shift



Flag

S	Z	E	O
1	0	—	—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	○	○	○	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	—	—				
Wd	○*	○	○	—	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	—	—	↕	↕	↑	—
Z	○	○	○	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	○	○				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

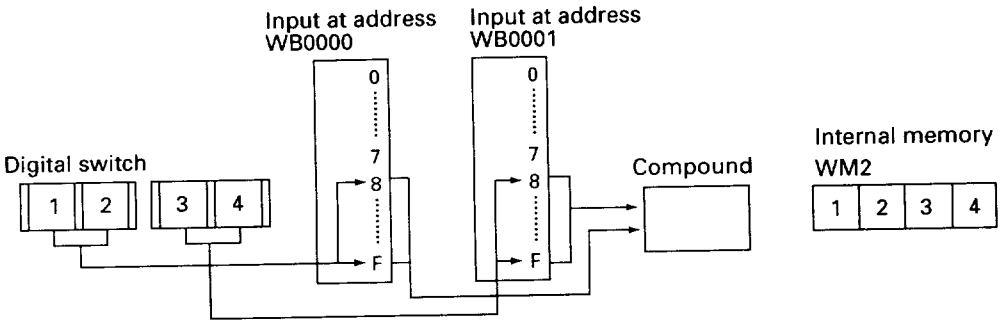
ONE-POINT ADVICE **Data compounding circuit**

An example of using SLL, AND and OR is shown below.

1. **Instruction**
   SLL, AND, OR

2. **Operation**
   Compounding of two input data (BCD 2 digits each) into one BCD 4 digit data.

3. **System diagram**



4. **Program**

```
[ WB0    SLL  WM0     :d8                ]
[ WB1    AND  h 00FF  ──→  WM1           ]
[ WM0    OR   WM1     ──→  WM2           ]
```

- Address WB0000 data is shifted 8 bits upward and stored in WM0000.
- 8 bits of address WB0001 data starting from highest position are made '0'.
- WM0000 and WM0001 are compounded.

• Example of monitoring with loader LITE

① Program monitoring

```
P0001|          00004812          00001200
 0001├──────[  WB0000     SLL   WM0000    :  d 00000008   ]
P0001|          00002134                        00000034
 0002├──────[  WB0001     AND   h 00FF      --→ WM0001    ]
P0001|          00001200          00000034        00001234
 0003├──────[  WM0000     OR    WM0001      --→ WM0002    ]
```

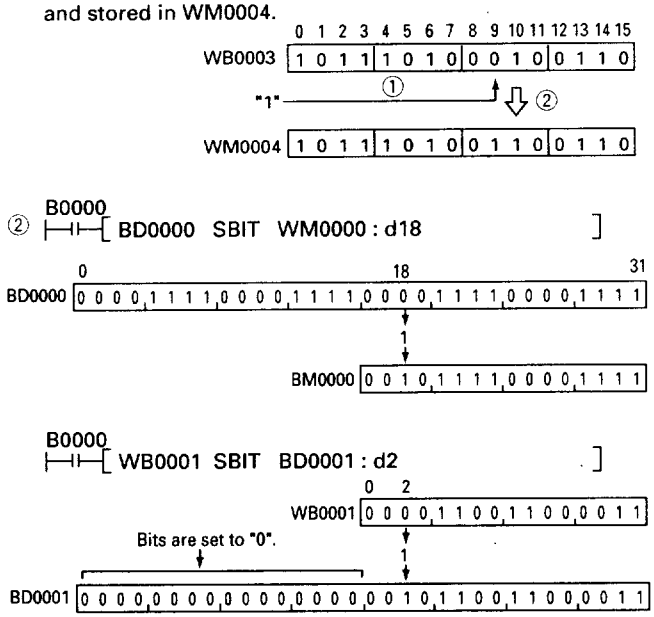
② Data monitoring

```
ADDRESS  BIN                 DEC          HEX        BCD
         ▼   ▼   ▼   ▼
  B000   0100100000010010    0000018450   00004812   00004812
  B001   0010000100110100    0000008500   00002134   00002134
  M000   0001001000000000▲   0000004608   00001200   00001200
  M001   0000000000110100    0000000052   00000034   00000034
  M002   0001001000110100    0000004660   00001234   00001234
  M003   0000000000000000    0000000000   00000000   00000000
```

Processor (▨ : Applicable)												Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

8. Set bit (SBIT)

Instruction	Set bit	F56	Description	Example
Symbol	Unconditional execution ⊢[Ws SBIT Wd: Z] Conditional execution ⊢⊣[Ws SBIT Wd: Z] Z: Bit position		① Care must be taken if the bit length of Ws and Wd is different. ② When Z is specified by using a word address, and if the contents of Z is not BCD code, an operation error occurs and the instruction is not executed.	① B001E ⊢⊣[WB0003 SBIT WM0004 : d9] When contact B001E is set ON, bit 9 of WB0003 is set to 1 and stored in WM0004. ② B0000 ⊢⊣[BD0000 SBIT WM0000 : d18] B0000 ⊢⊣[WB0001 SBIT BD0001 : d2]
Function	① The Zth bit of Ws is set to 1 and stored in Wd. However, data of Ws is not changed. ② Z cannot exceed the numerical value of the bit length of Ws, or cannot be a negative value.			



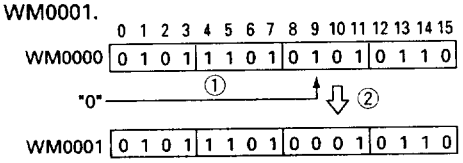
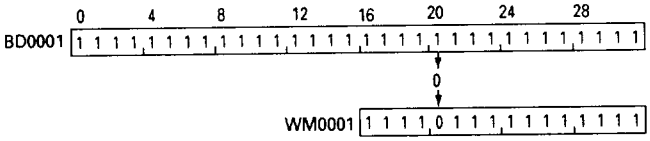
Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

9. Reset bit (RBIT)

Instruction	Reset bit	F57	Description	Example
Symbol	Unconditional execution Ws RBIT Wd: Z Conditional execution Ws RBIT Wd: Z Z: Bit position		① Care must be taken if the bit length of Ws and Wd is different. ② When Z is specified by using a word address, and if the contents of Z is not BCD code, an operation error occurs and the instruction is not executed.	B001F WM0000 RBIT WM0001 : d9 When contact B001F is set ON, bit 9 of WB0000 is set to 0 and stored in WM0001.  B0001 BD0001 RBIT WM0001 : d20 
Function	① The Zth bit of Ws is reset to 0 and stored in Wd. However, data of Ws is not changed. ② Z cannot exceed the numerical value of the bit length of Ws, or cannot be a negative value.			

Operand and influence flag

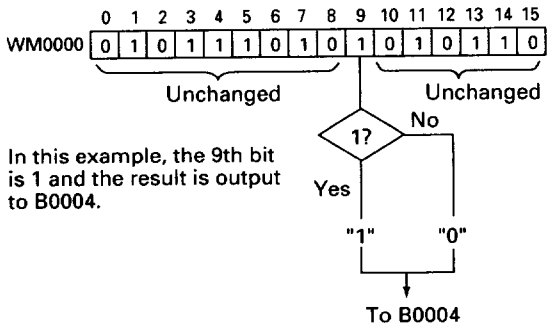
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

10. Test bit (TBIT)

Processor (: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Test bit	F58	Description	Example
Symbol	Unconditional execution Output relay Ws TBIT Z Conditional execution Output relay Ws TBIT Z Z: Bit position		① When Z is specified by using a word address, and if the contents of Z is not BCD code, an operation error occurs and the instruction is not executed.	B001E WM0000 TBIT d9 B0004 When contact B001F is set ON, bit 9 of WB0000 is set to 0 and stored in WM0001. In this example, the 9th bit is 1 and the result is output to B0004.
Function	① A check is made to find whether the Zth bit of Ws is 0 or 1. If this bit is 1, the output relay is set ON. If it is 0, the output relay is set OFF. ② Z cannot exceed the numerical value of the bit length of Ws, or cannot be a negative value.			



Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

3-4-7 Conversion

1. Conversion

- The eleven types of data conversion instructions (having different functions) are as follows.

○: Available, —: Not available

Instruction	Abbr.	Symbol	Function	F30, F50 F50H	F60	F55, F70, F80H, F120H, F70S, F120S, F140S, F150S
(1) Binary/BCD conversion	BCD	[Z BCD Wd]	Binary data Z is converted into BCD and the result is stored in Wd.	○	○	○
(2) BCD/Binary conversion	BIN	[Z BIN Wd]	BCD data Z is converted into binary and stored in Wd.	○	○	○
(3) Character string	CHAR	[CHAR N, Wd]	Data which consists of N characters is transferred to the address indicated by Wd.	—	—	○
(4) Numeric conversion	FIG	[Ws FIG Wd]	Data in the address indicated Ws is regarded as ASCII codes and is converted to numerical data and the results is stored in Wd.	—	—	○
(5) ASCII conversion	ASCII	[Z ASCII Wd, N]	Numerical data Z is converted to ASCII codes and the result is stored in Wd.	—	—	○
(6) Conversion to seconds	SEC	[Ws SEC Wd]	Day, hour, minute, and seconds data at Ws is converted to seconds data and the result is stored in Wd.	—	—	○
(7) Conversion from seconds	TIM	[Ws TIM Wd]	Seconds data at Ws is converted to day, hour, minute, and seconds data and the result is stored in Wd.	—	—	○
(8) Decode	DECO	[Z DECO Wd]	The bits indicated by Z are set to 1, other bits are set to 0, and the result is stored in Wd.	○	○	○
(9) Encode	ENCO	[Z ENCO Wd]	The most significant bit in Z that is 1 is stored in Wd.	○	○	○
(10) 7-segment decode	7SEG	[Z 7SEG Wd]	Z is converted into the activation or inactivation command for each segment of the 7-segment display and the result is stored in Wd.	—	○	○
(11) Count on bit	BCNT	[Z BCNT Wd]	The No. of bit 1's in Z is converted into the corresponding numeric value (BCD) and the result is stored in Wd.	○	○	○

- Data formats and executing operations

① If Z (source data) to be processed by a BCD/binary conversion or decode instruction is not BCD format data, the operation error flag (A0041) is set ON.

② If the most-significant bit of the operation result is 1, the sign flag (F004E) is set ON.

③ If all bits of the operation result are 0, the zero flag (F004F) is set ON.

④ Two's complement is used to express a negative binary number.

Reference:

Expressing negative binary numbers using two's complement

Two's complement is used to express a negative binary number.

For conversion into two's complement, the logic of all bits of a positive binary number are inverted and the result is incremented by one.

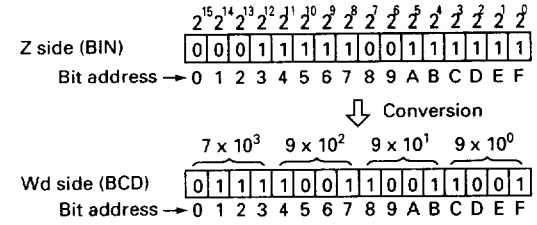
	16-bit area	32-bit area
Obtaining −1	0000 0000 0000 0001 → 1 1111 1111 1111 1110 → Invert 1111 1111 1111 1111 → +1 F F F F ← Hexadecimal notation	0000 0000 0000 0000 0000 0000 0000 0001 1111 1111 1111 1111 1111 1111 1111 1110 → Invert 1111 1111 1111 1111 1111 1111 1111 1111 → +1 F F F F F F F F
Obtaining −255	0000 0000 1111 1111 → 255 1111 1111 0000 0000 → Invert 1111 1111 0000 0001 → +1 F F 0 1 ← Hexadecimal notation	0000 0000 0000 0000 0000 0000 1111 1111 1111 1111 1111 1111 1111 1111 0000 0000 → Invert 1111 1111 1111 1111 1111 1111 0000 0001 → +1 F F F F F F F F ← Hexadecimal notation

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F160S	F150S	D05	D10S	D20	LITE	Soft-ware

2. Binary/BCD conversion (BCD)

Instruction	Binary/BCD conversion	F70
Symbol	Unconditional execution ⊢[Z BCD Wd: Z] Conditional execution ⊢⊣[Z BCD Wd: Z]	
Function	① Binary data Z is converted into BCD, and the result is stored in Wd. ② When the result exceeds the maximum value of the storage address, an overflow flag is set, and the maximum value is stored (Max. value: 79999999). ③ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

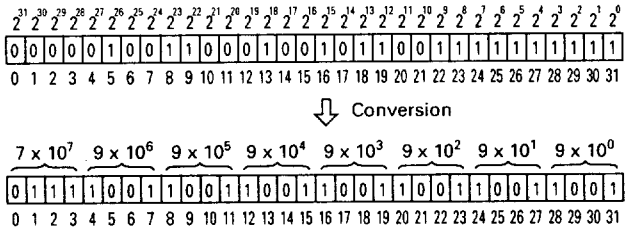
① In case of 16-bit data



Flag

S	Z	E	O
0	0	—	0

② In case of 32-bit data



Flag

S	Z	E	O
0	0	—	0

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z	○	○	○	○	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	–	○	S	Z	E	O
Wd	○*	○	○	–	○	○	○	○	○	○	○	○	○**	○	○	○	○	○	○	○	○	○	○	○	–	–	↕	↕	↑	↑

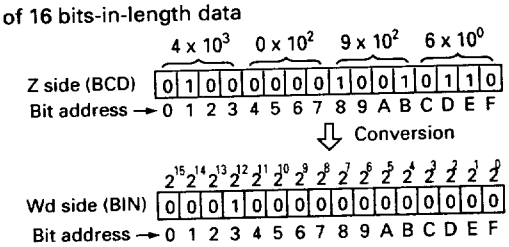
* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

3. BCD/Binary conversion (BIN)

Instruction	BCD/Binary conversion	F71
Symbol	Unconditional execution ⊢—[Z BIN Wd] Conditional execution ⊢⊣[Z BIN Wd]	
Function	① BCD data Z is converted into binary code, and the result is stored in Wd. ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

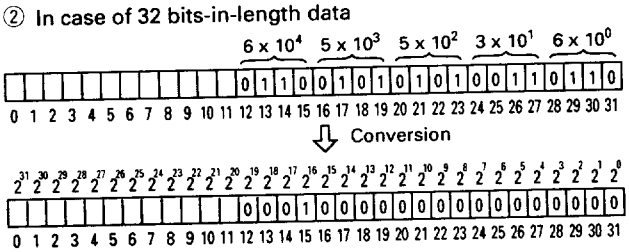
① In case of 16 bits-in-length data



Flag

S	Z	E	O
0	0	0	—

② In case of 32 bits-in-length data



Flag

S	Z	E	O
0	0	—	0

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	↕	↕	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

4. Character string (CHAR)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Character string
Symbol	Unconditional execution [CHAR N : Wd] [DATA] Conditional execution [CHAR N : Wd] [DATA]
Function	① N data items are transferred to Wd as character data. ② A DATA instruction (page 3-145) can transfer 1 to 10 characters or 1 to 5 kanji characters. ③ The No. of data items one CHAR instruction can convert is as follows. Kanji: 5 x 255 lines Character: 10 x 255 lines

Destination of character data transfer

[CHAR 2 : BD00]

No. of character data items

[DATA : 1 2 3 4 5 6 :]

[DATA : A B C D E F G H :]

BD0	31	32	33	34
BD1	35	36	41	42
BD2	43	44	45	46
BD3	47	48	00	00

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

5. Numeric conversion (FIG)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	ASCII/numeric conversion	F79	Description	Example
Symbol	Unconditional execution ⊢[Ws FIG Wd] Conditional execution ⊢⊣[Ws FIG Wd]		If the length of the result of converting the data in Ws is longer than the area Wd, an operation error occurs.	Ws 3 0 3 1 3 2 3 3 Wd (2 digits) Wd (4 digits) Wd (8 digits) 0 1 / 2 3 0 1 2 3 0 0 0 0 0 1 2 3 The storage format depends on the number of digits in the transfer destination.
Function	① The contents of Ws are assumed to be ASCII codes and converted into the corresponding numeric value, and the result is stored in Wd. ② If the result of converting the data codes in Ws is not a value that can be expressed by 0 to 9 and A to F, an operation error occurs. ③ Spaces (h20) are converted into 0s.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

6. ASCII conversion (ASCII)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Numeric/ASCII conversion	F78	Description	Example
Symbol	Unconditional execution ⊢[Z ASCII Wd, N] Conditional execution ⊢⊣[Z ASCII Wd, N]		① If the result of converting the data in Z is longer than the area Wd, an operation error occurs. ② If Z cannot be converted to ASCII code, an operation error occurs.	Z (4 digits) 0 1 2 3 (Hexadecimal number) Wd (4 digits) N=0 (No zero-suppression) N=1 (Zero-suppression) WM0 30 31 / WM1 32 33 20 21 / 22 23 Wd (8 digits) <N=0> <N=1> BD0 30 31 32 33 20 31 32 33 The storage type depends on the number of digits stored in the destination.
Function	① The data in Z is assumed to be binary data and converted into the corresponding ASCII codes, and the result is stored in Wd. ② When N=1, zero suppression is performed. ③ Z is unsigned data.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

7. Time data conversion (SEC)

Instruction	Days, hours, minutes, seconds → seconds	F81

Symbol

Unconditional execution

```
|----[ Ws  SEC  Wd ]
```

Conditional execution

```
|-|-[ Ws  SEC  Wd ]
```

Function

① The contents of Ws are assumed to be data indicating days, hours, minutes, and seconds and converted into seconds, and the result is stored in Wd.
② The data in Ws is unsigned BCD.

Description

① The area Ws must be large enough for BCD 8 digits. If Ws is an 8-bit area, 4 words are needed. If it is a 16-bit area, 2 words are needed. If it is a 32-bit area, 1 word is needed. If Ws is an 8-bit or 16-bit area, the first address must be specified for Ws. If the area specified by Ws is not enough for the required words, a user program error occurs.
② If Ws exceeds 79 days, 23 hours, 59 minutes, and 59 seconds, an operation error occurs.
③ If the result exceeds the data range for Wd, the overflow relay is turned ON and the maximum value of Wd is stored in it.

Example

Ws (2 digits)

1 8
1 2
3 9
4 0

Ws (4 digits)

1 8 1 2
3 9 4 0

→ Wd
1600780 seconds

Ws (8 digits)

BD0 | 1 8 1 2 3 9 4 0 |

The storage type depends on the number of digits in the transfer destination.

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

8. Time data conversion (TIM)

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Seconds → days, hours, minutes, seconds	F82	Description	Example
Symbol	Unconditional execution ⊢[Ws TIM Wd] Conditional execution ⊢‖[Ws TIM Wd]		① The area Wd must be large enough for BCD 8 digits. If Wd is an 8-bit area, 4 words are needed. If it is a 16-bit area, 2 words are needed. If it is a 32-bit area, 1 word is needed. If Wd is an 8-bit or 16-bit area, the first address must be specified for Wd. If the area specified by Wd is not enough for the required words, a user program error occurs. ② If Ws exceeds 6911999 seconds, an operation error occurs.	Wd (2 digits) 18 12 39 40 Wd (4 digits) Wd 1600780 seconds (18 days, 12 hours, 39 minutes, 40 seconds) 1 8 1 2 3 9 4 0 Ws (8 digits) 1 8 1 2 3 9 4 0 The storage type depends on the number of digits in the transfer destination.
Function	① The contents of Ws are assumed to be data indicating seconds and converted into days, hours, minutes, and seconds, and the result is transferred in Wd. ② The data in Ws is unsigned BCD.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–

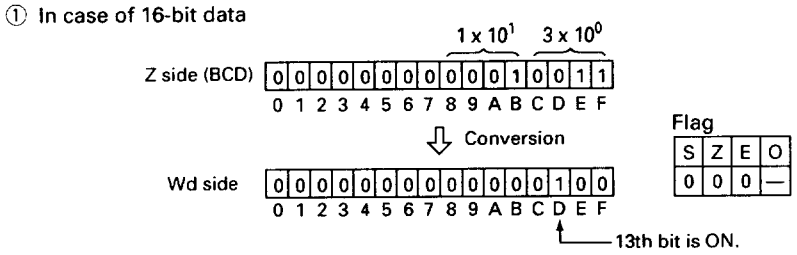
* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

9. Decode (DECO)

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F80	F70	F80H	F120H	F70S	F120S	F140S	F160S	D05	D10S	D20	LITE	Soft-ware

Instruction	Decode	F72
Symbol	Unconditional execution ⊢—[Z DECO Wd] Conditional execution ⊢⊣[Z DECO Wd]	
Function	① One is stored in one of Wd bits whose bit position is given by the source data value expressed in a BCD code. Zeros are written to all other Wd bit positions. ② If a Wd bit position corresponding to the source data in Z does not exist, an operation error flag is set. The program corresponding to the source data is not executed. ③ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

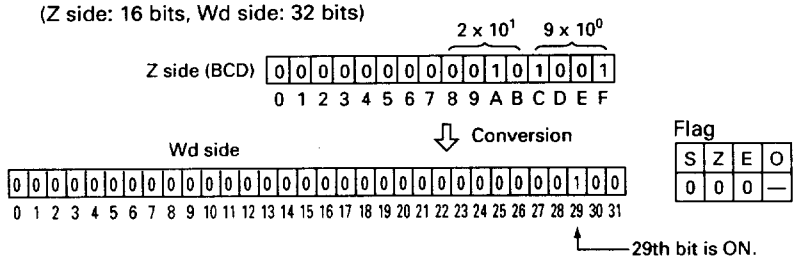
① In case of 16-bit data



Flag

S	Z	E	O
0	0	0	—

② In case of 32-bit data
(Z side: 16 bits, Wd side: 32 bits)



Flag

S	Z	E	O
0	0	0	—

Operand and influence flag

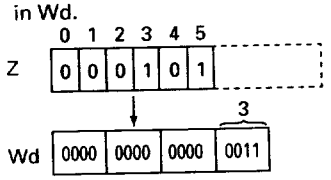
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	↕	↕	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

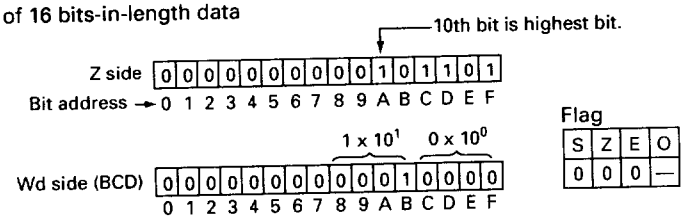
Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

10. Encode (ENCO)

Instruction	Encode	F73
Symbol	Unconditional execution ⊢[Z ENCO Wd] Conditional execution ⊢⊣[Z ENCO Wd]	
Function	① The highest bit position where "1" of source data Z is set is changed into BCD number, and it is stored in Wd. ② If all bits of Z are 0, the operation error flag (A0041) is set ON. ③ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	



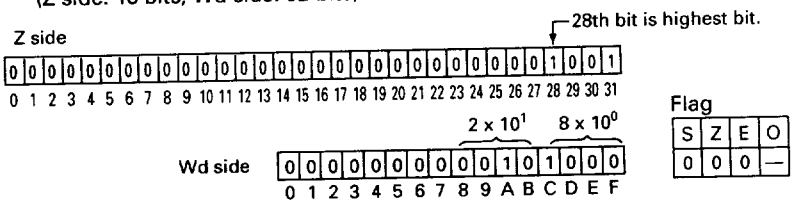
① In case of 16 bits-in-length data



Flag

S	Z	E	O
0	0	0	—

② In case of 32 bits-in-length data
(Z side: 16 bits, Wd side: 32 bits)



Flag

S	Z	E	O
0	0	0	—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				
Wd	O*	O	O	—	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	—	—	↕	↕	↑	—

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

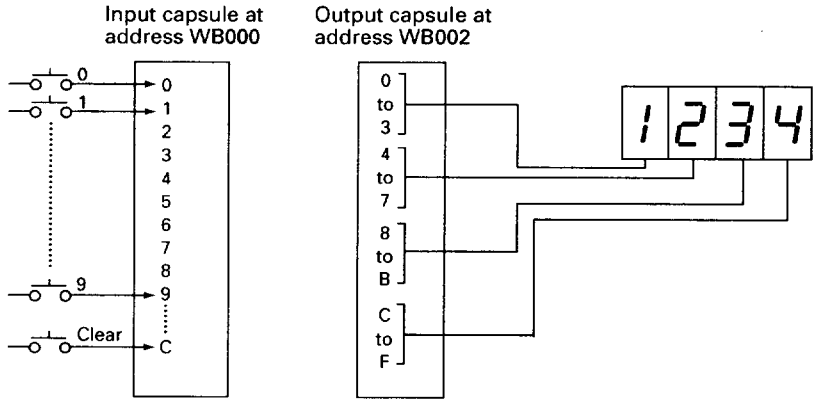
ONE-POINT ADVICE **Decimal 4-digit ten-key pad input circuit**

An example of using ENCO, SLL, OR and MOV is shown below.

1. **Instructions**
   ENCO, SLL, OR and MOV

2. **Operation**
   Data is input from a decimal 4-digit ten-key pad. Incorrect input can be cleared by pressing the CLR key.

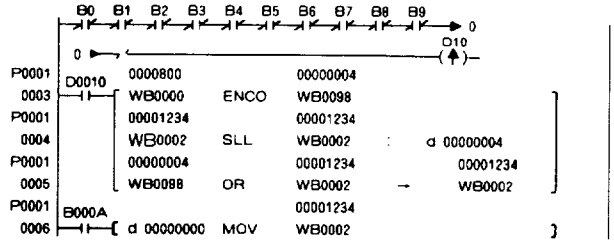
3. **System diagram**



4. **Program**

- Example of monitoring with loader LITE

① Program monitoring



② Data monitoring

Address	BIN	DEC	HEX	BCD
	▼ ▼ ▼ ▼			
B000	0000100000000000	0000002048	00000800	00000800
B001	0000000000000000	0000000000	00000000	00000000
B002	0001001000110100	0000004660	00001234	00001234
B003	0000000000000000	0000000000	00000000	00000000
B098	0000000000000100 ▲	0000000004	00000004	00000004
B099	0000000000000000	0000000000	00000000	00000000

- When one of these contacts is set ON, a differential pulse is generated.
- The numeric value input in WB000 is temporarily stored in WB098.
- The contents of WB002 are shifted four bits to the left.
- The contents of WB098 and the contents of WB002 are compounded.
- All bits of WB002 are set to 0 (resetting).

Operation procedure

① When the CLR key is pressed (B000C is ON), 0 is displayed on the 7-segment display (WB2).

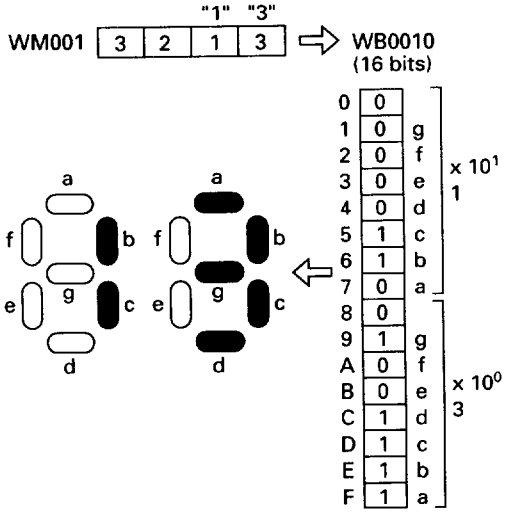
② When one of the ten keys (B0000 to B0009) is pressed, the input number is displayed on the 7-segment display.

③ When one of the ten keys is pressed, the displayed number is shifted one digit to the left and the new input number is displayed in the column 1.

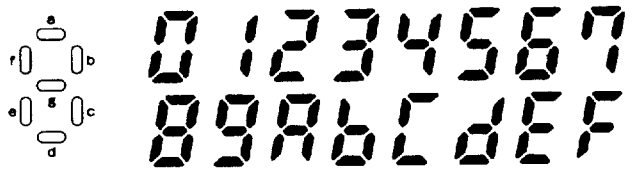
Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F50H	F120I	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

11. 7-segment decode (7SEG)

Instruction	7-segment decode	F74	Description	Example
Symbol	Unconditional execution ⊢[Z 7SEG Wd] Conditional execution ⊢⊣[Z 7SEG Wd]		① If Wd has a 16-bit length, only the eight low-order bits of Z are output and the remaining bits are discarded. ② If Wd has a 32-bit length and Z is 16-bit data, all of Z is output. ③ This instruction is used for a 7-segment display without a BCD converter.	⊢⊣[WM0001 7SEG WB0010]
Function	① The numeric data Z is converted into data set for the segments (a to g) of the 7-segment display. This data is stored in Wd. ② Characters to be displayed 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, b, C, d, E, F			



7-segment display Displayed characters



③ In F60 series, W (file) cannot be specified as operand.

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	↕	↕	↑	–
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	↕	↕	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

12. Count on bit (BCNT)

Instruction	Count on bit	F75
Symbol	Unconditional execution ⊢[Z BCNT Wd] Conditional execution ⊢⊣[Z BCNT Wd]	
Function	① Number of ON-bits 1 in Z is converted into BCD number, which is stored in Wd. Z: 1 1 0 1 All 0 1 0 Wd: 4 ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

① In case of 16-bit data

All are ON.

Z side: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Bit address → 0 1 2 3 4 5 6 7 8 9 A B C D E F

Wd side (BCD): 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 (1 × 10^1, 6 × 10^0)

Flag

S	Z	E	O
0	0	—	—

② In case of 32-bit data (Z side: 32 bits, Wd side: 16 bits)

Z side: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Wd side: 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 (2 × 10^1, 8 × 10^0)

0 1 2 3 4 5 6 7 8 9 A B C D E F

Flag

S	Z	E	O
0	0	—	—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	↕	↕	↑	—
Wd	O*	O	O	—	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	—	—				

* The input address of WB cannot be specified for Wd (operation result storage address).

** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

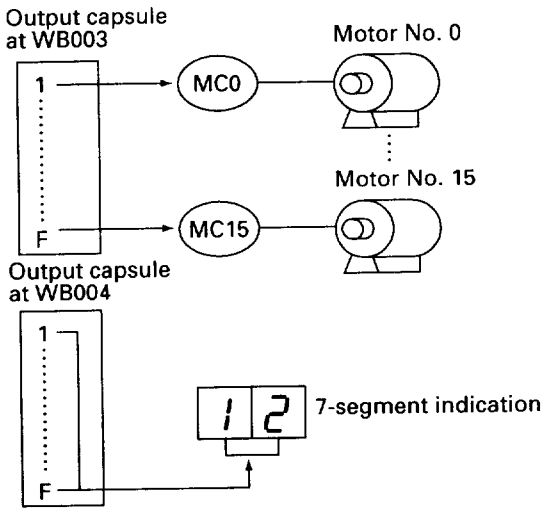
ONE-POINT ADVICE **Circuit to monitor the number of operating motors**

An example of using BCNT is shown below.

1. **Instruction**
   BCNT, 7SEG

2. **Operation**
   16 motors are used, and the No. of motors operating among them is monitored.

3. **System diagram**



4. **Program**

```
F54
 ├┤┌ WB3    BCNT WM1
 │└ WM1    7SEG WB4
 │
 ├
```

- No. of ON bits of WB3 are stored in WM1.
- WM1 data are formed into 7-segment and output to WB4.

• Example of monitoring with loader LITE

① Program monitoring

```
P0001|F0054    00007777          00000012
 0001├─┤┤─[ WB0003      BCNT  WM0001
P0001|      |  00000012          --------
 0002├       [ WM0001      7SEG  WB0004
```

② Data monitoring

```
ADDRESS  BIN                DEC         HEX       BCD
         ▼  ▼  ▼  ▼
  B003   0111011101110111▲ 0000030583  00007777  00007777
  B004   0000011001011011  0000001627  0000065B  --------
  M000   0000000000000000  0000000000  00000000  00000000
  M001   0000000000010010  0000000018  00000012  00000012
  M002   0000000000000000  0000000000  00000000  00000000
  M003   0000000000000000  0000000000  00000000  00000000
```

No. of ON bits of station No. 3 (WB003) is 12.

No. of ON bits are stored.

3-4-8 Transfer

1. Transfer

- The twelve types of transfer instructions (having different functions) are as follows.

○: Available, —: Not available

Instruction	Abbr.	Symbol	Function	F30, F50 F50H	F60	F55, F70, F80H, F120H, F70S, F120S, F140S, F150S
(1) Data transfer	MOV	⊢[Z MOV Wd]⊣	Z is transferred to Wd.	○	○	○
(2) Logical transfer	LMOV	⊢[Z LMOV Wd]⊣	Z is transferred to Wd.	○	○	○
(3) Data block transfer	BT	⊢[Ws BT Wd:Z]⊣	Data in N words are transferred.	○	○	○
(4) Logical block transfer	LBT	⊢[Ws LBT Wd:Z]⊣	Data in N continuous words are transferred.	○	○	○
(5) Digit transfer	DT	⊢[Ws DT Wd:N1:N2:N3]⊣	Data reaching from N_1+N_2 in Z is transferred to the bits following the N_3 in Wd.	○	○	○
(6) High order digit transfer	MOVU	⊢[Z MOVU Wd]⊣	16-bit data "Z" is transferred to the 16 high order bits in the 32-bit data area (Wd). The 16 high order bits of Z (32-bit data) are transferred to Wd (16-bit area).	—	○	○
(7) Low order digit transfer	MOVL	⊢[Z MOVL Wd]⊣	16-bit data "Z" is transferred to the 16 low order bits in the 32-bit data area (Wd). The 16 low order bits of Z (32-bit data) are transferred to Wd (16-bit area).	—	○	○
(8) Pattern clear	PC	⊢[Z1 PC Wd:Z2]⊣	The Z_2 words of Wd are cleared by Z_1 pattern.	—	—	○
(9) Search	SRCH	⊢[Z1 SRCH Ws:Wd:Z2]( )⊣	The data same as Z_1 is searched through the Z words of Ws and the result is output to the relay.	—	—	○
(10) Switch	SW	⊢[Bs Z1:Z2 SW Wd]⊣	The following transfer is carried out depending on switching input state. Switching input ON: $Z_1 \rightarrow$ Wd Switching input OFF: $Z_2 \rightarrow$ Wd	—	—	○
(11) Message transmission	MSGT	⊢[MSGT N1,N2,Z,Wd]⊣	The specified data is transferred to station Z indicated by N_1. The result is stored in Wd.	—	—	○
(12) Message reception	MSGR	⊢[MSGR N1,N2,Z,Wd]⊣	The specified data is transferred to station Z indicated by N_2. The result is stored in Wd.	—	—	○

- **Uses of transfer instructions**

(1) Data transfer
This instruction is used to transfer signed BCD data. If the transfer source has a longer data length than the transfer destination, data having the maximum length value is stored at the transfer destination and the overflow flag (A0040) is set ON.

(2) Logical transfer
This instruction is used to transfer binary data. If the transfer source has a longer data length than the transfer destination, only the low-order bits of data are stored at the transfer destination. In this case, the overflow flag is not set.

(3) Data block transfer
This instruction is used to transfer N words of signed BCD data. The transfer source and destination must have the same data length.

(4) Logical block transfer
This instruction is used to transfer N words of binary data. The transfer source and destination must have the same data length.

(5) Digit transfer
This instruction is used to transfer data from specified source bits.

(6) High-order digit transfer
This instruction is used to store the 16 high-order bits of 32-bit data at the transfer destination or to transfer 16-bit data to the 16 high-order bits of the 32-bit destination.

(7) Low-order digit transfer
This instruction is used to store the 16 low-order bits of 32-bit data at the transfer destination or to transfer 16-bit data to the 16 low-order bits of the 32-bit destination.

(8) Pattern clear
This instruction is used to replace N-word data with specified data.

(9) Search
This instruction is used to search for specified data out of N words of data and to output "presence" or "absence".

(10) Switch
This instruction is used to select one of two data items by using a single switch.

(11) Message transmission
This instruction is used to send data for other data modules via networks such as the T-link or P-link.

(12) Message reception
This instruction is used to receive data from other data modules via networks such as the T-link or P-link.

Identifier	B	M	K	D	F	A	S	T	C	i	j	k	ℓ	m	P	Q
Availability	○*	○	○	—	—	○	—	—	—	○	○	○	—	—	○	○

* B is not usable for input address.

2. Data transfer (MOV)

Processor (▨ : Applicable)											Program loader				
F30	F60 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Data transfer	F170
Symbol	Unconditional execution ⊢[Z MOV Wd] Conditional execution ⊢⊣[Z MOV Wd]	
Function	① Z (with a sign) is transferred to Wd. ② Z is signed BCD data. ③ Even if Z is not BCD data, it is assumed to be signed BCD data and is transferred. (Example ③ on the right) ④ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

MOV

① In case Z is BCD 4 digits and Wd is BCD 8 digits

Z (4 digits) MOV Wd (8 digits)

6,5,2,1 ➡ 0,0,0,0,6,5,1,2

Flag

S	Z	E	O
—	—	—	0

② In case Z is 32-bits long and Wd is 16-bits long.

Z_2 (8 digits) MOV Wd (4 digits)

0,0,1,2,3,4,5,6 ➡ 7,9,9,9

Flag

S	Z	E	O
—	—	—	1

* If the transfer source data is larger than "7999" and the sign flag (sign bit) is not set ON, an overflow flag is set ON as shown above and the maximum BCD value "7999" for Wd is stored.

Z MOV Wd (4 digits)

A,B,C,D,E,F,0,0 ➡ F,9,9,9

Hexadecimal (-7999 in decimal notation)

Flag

S	Z	E	O
—	—	—	1

* If the transfer source data is larger than "7999" and the sign flag (sign bit) is not set ON, an overflow flag is set ON as shown above and the maximum BCD value "-7999" for Wd is stored.

③ In case Z is 16-bits long and Wd is 32-bits long.

Z MOV Wd (8 digits)

C,D,F,F ➡ 8,0,0,0,4,D,E,F

Flag

S	Z	E	O
—	—	—	—

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

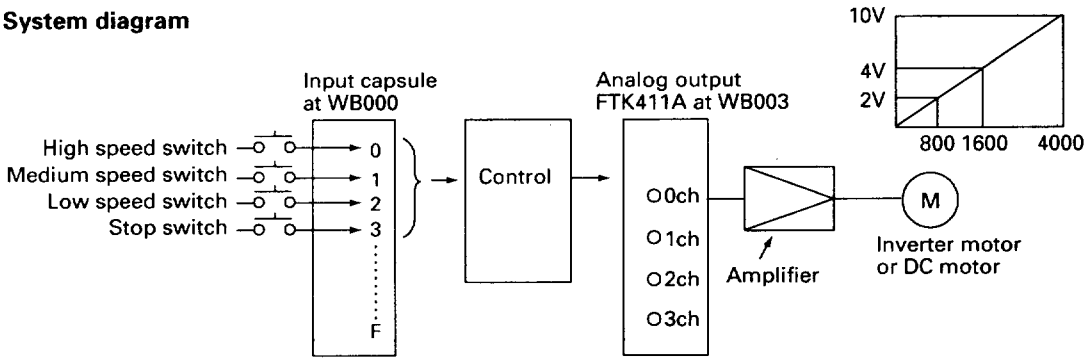
ONE-POINT ADVICE **Motor-speed control circuit**

An example of using MOV is shown below.

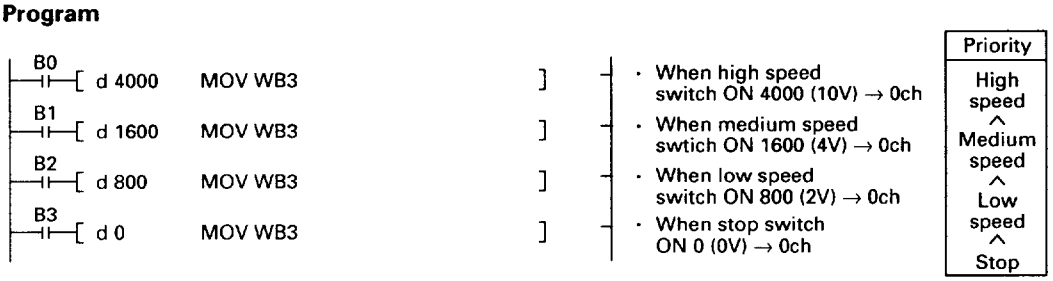
1. **Instruction**
   MOV

2. **Operation**
   Motor is rotated at 4000 rpm (10V) with high speed switch ON, and at 1600 rpm (4V) with medium speed switch ON, and at 800 rpm (2V) with low speed switch ON.

3. **System diagram**



4. **Program**



• Example of monitoring with loader LITE

① Program monitoring

```
P0001|B0000                    00001600
 0001├--┤├--[  d 00004000  MOV   WB0003  ]
P0001|B0001                    00001600
 0002├--┤├--[  d 00001600  MOV   WB0003  ]
P0001|B0002                    00001600
 0003├--┤├--[  d 00000800  MOV   WB0003  ]
P0001|B0003                    00001600
 0004├--┤├--[  d 00000000  MOV   WB0003  ]
```

② Data monitoring

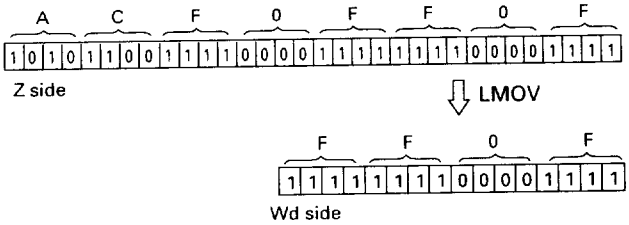
```
ADDRESS  BIN                DEC       HEX       BCD
         ▼  ▼  ▼  ▼
B000   0100000000000000  00016384  00004000  00004000  ← 1600 (4V) is output with
B001   0000000000000000  00000000  00000000  00000000    medium speed switch ON.
B002   0000000000000000  00000000  00000000  00000000
B003   0000011000000000  00005632  00001600  00001600
B004   0000000000000000  00000000  00000000  00000000
B005   0000000000000000  00000000  00000000  00000000
```

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

3. Logical transfer (LMOV)

Instruction	Logical transfer	F180
Symbol	Unconditional execution ⊢—[Z LMOV Wd] Conditional execution ⊢⊣[Z LMOV Wd]	
Function	① Z (with bit information) is transferred to Wd . ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

LMOV
(Z side: 32 bits, Wd side: 16 bits)



Flag

S	Z	E	O
—	—	—	—

* If data is transferred from 32 bit data area to 16 bit data area as shown above, the 16 low-order bits of 32-bit data is transferred to wd.

Operand and influence flag

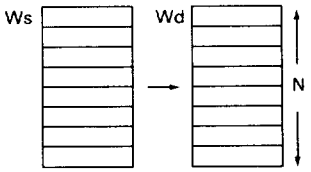
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Data block transfer (BT)

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F160S	D05	D10S	D20	LITE	Soft-ware

4. Data block transfer (BT)

Instruction	Data block transfer	F171	Description
Symbol	Unconditional execution ⊢[Ws BT Wd : Z] Conditional execution ⊢⊣[Ws BT Wd : Z] Z: No. of words to be transferred (1 to 4096)		① If the No. of remaining words of Ws and Wd are less than Z, the following occurs: If Z is directly specified, a user program error occurs. If Z is indirectly specified, an operation error occurs when the data is less than Z. For details, see below. ② The process to be taken when an overflow occurs is as same as that for MOV instruction (data transfer).
Function	① Consecutive N words of data specified by Z are transferred with a sign.  ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.		

B0060
⊢⊣[WB0000 BT WM0000 : d4]

Ws (BCD 4 digits)		Wd (BCD 4 digits)		Flag			
WB000	0,1,2,3	WM000	0,1,2,3	S	Z	E	O
WB001	5,6,7,8	WM001	5,6,7,8	—	—	0	0
WB002	0,5,0,6	WM002	0,5,0,6				
WB003	1,0,2,0	WM003	1,0,2,0				

Operand and influence flag

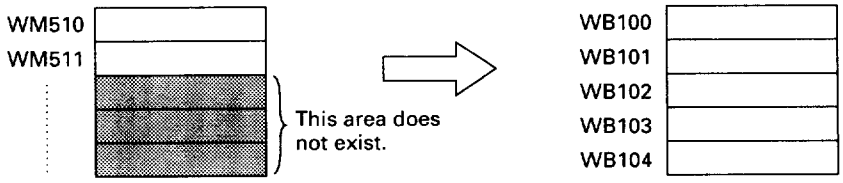
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Note on programming

In programming, when using the data block transfer (BT) and logical block transfer (LBT) instructions, pay attention to the address and word count of the transfer origin and destination.

For example, assuming that a data block is transferred from the auxiliary relay area to the keep relay area in the F120S series.



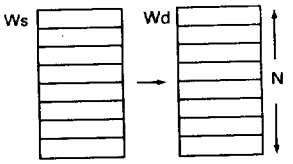
If the non-existing area is specified as shown in the above diagram, either of the following processing is performed in accordance with the programming method:

⊢[WM510 BT WB100 : d3] ⇨ If direct specification "d5" is programmed in this way, ***a user program error occurs.***

⊢[WM510 BT WB100 : WM0] ⇨ If indirect specification "WM0" is programmed, when data in WM0 exceeds d3, ***the operation execution error flag is set ON,*** disabling data transfer.

5. Logical block transfer (LBT)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F65	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Data block transfer	F181	Description
Symbol	Unconditional execution ⊢⊣⊢[Ws LBT Wd : Z] Conditional execution ⊢⊣⊢[Ws LBT Wd : Z] Z: No. of words to be transferred (1 to 4096)		① If Z is in the status below, an operation error flag (A0041) is set ON and operation is not executed. • When Z is specified by word and the contents is other than BCD code, or Z is a negative value. ② If the No. of remaining words of Ws and Wd are less than Z, a user program error occurs.
Function	① Consecutive N words of data (specified by Z) are transferred.  ② In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.		

⊢⊣⊢[WB0000 LBT BD0000 : d2]

Example of hexadecimal expression

Ws (Hexadecimal 4 digits)		Wd (Hexadecimal 8 digits)	
WB000	A 3 4 5	BD0000	0 0 0 0 A 3 4 5
WB001	C 5 6 7	BD0001	0 0 0 0 C 5 6 7

Operand and influence flag

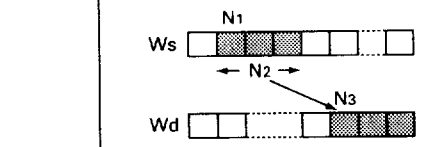
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				

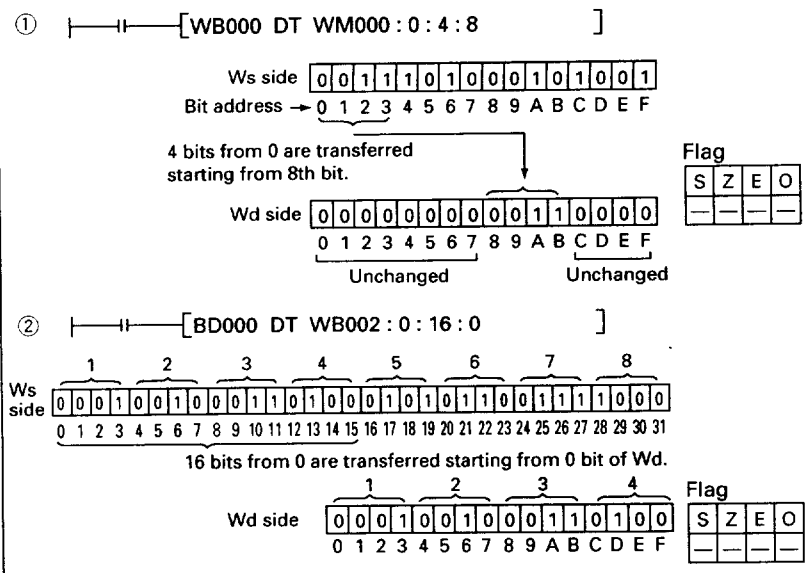
* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

6. Digit transfer (DT)

Instruction	Digit transfer	F172
Symbol	Unconditional execution ⊢[Ws DT Wd : N1 : N2 : N3] Conditional execution ⊢⊣[Ws DT Wd : N1 : N2 : N3] N1 (0 to 31):First bit position of source N2 (1 to 32):Number of bits to be transferred N3 (0 to 31):First bit position of destination	
Function	① N2 bits from bit N1 of Ws are transferred to the bit N3 and after of Wd. ② When set data for N1, N2 or N3 are incorrect, a user program error occurs on initialization sequence, and operation cannot be carried out. (ALARM lamp ON) ③ N1, N2: to be within Ws data length N2, N3: to be within Wd data length ④ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	



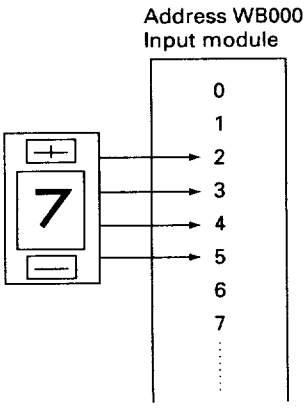


Operand and influence flag

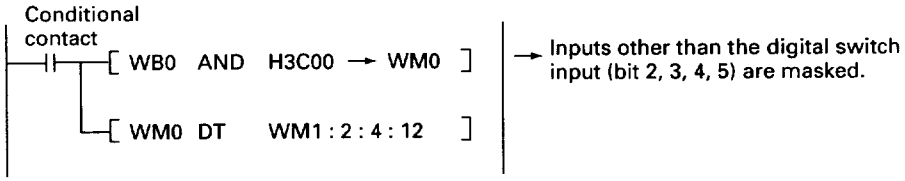
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↕	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

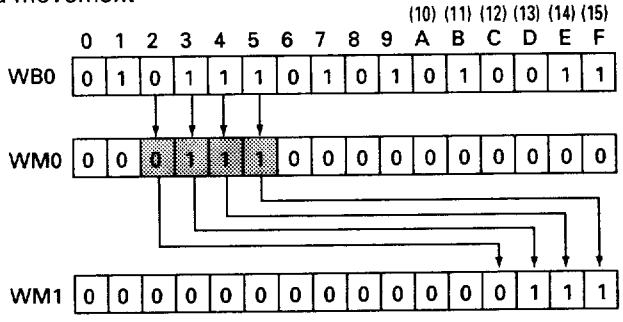
Program example



As shown on the left, the digital switch (BCD input) data which is input to bit 2, 3, 4, and 5 of input address WB0 is transferred to bit C, D, E, and F of WM1.



Data movement



Due to masking operation, information on WB0 bit 2, 3, 4, and 5 is only transferred to WM0 bit 2, 3, 4, and 5.

7. High-order digit transfer (MOVU)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	High-order digit transfer	F173	Description	Example
Symbol	Unconditional execution [Z MOVU Wd] Conditional execution [Z MOVU Wd]		① 16 bits Z Wd Unchanged ② 32 bits Z Wd 16 bits ③ The data length of Z must be one-half or double the data length of Wd.	WB000 WB001 10^7 10^6 10^5 10^4 10^3 10^2 10^1 10^0 High-order digit transfer Low-order digit transfer BD0000 [WB0000 MOVU BD0000] [WB0001 MOVL BD0000]
Function	① Z (16-bit data) is transferred to the 16 high-order bits of Wd (32-bit data area). 16 bits 32 bits ② The 16 high-order bits of Z (32-bit data) are transferred to Wd (16-bit data area). ③ In F60 series, W (file) cannot be specified as operand.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

8. Low-order digit transfer (MOVL)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Low-order digit transfer	F174	Description	Example
Symbol	Unconditional execution [Z MOVL Wd] Conditional execution [Z MOVL Wd]		① 16 bits Z Wd Unchanged ② 32 bits Z Wd 16 bits ③ The data length of Z must be one-half or double the data length of Wd.	WB0 WB1 10^7 10^6 10^5 10^4 10^3 10^2 10^1 10^0 High-order digit transfer Low-order digit transfer BD0 [WB0 MOVU BD0] [WB1 MOVL BD0]
Function	① Z (16-bit data) is transferred to the 16 high-order bits of Wd (32-bit data area). 16 bits 32 bits ② The 16 high-order bits of Z (32-bit data) are transferred to Wd (16-bit data area). ③ In F60 series, W (file) cannot be specified as operand.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

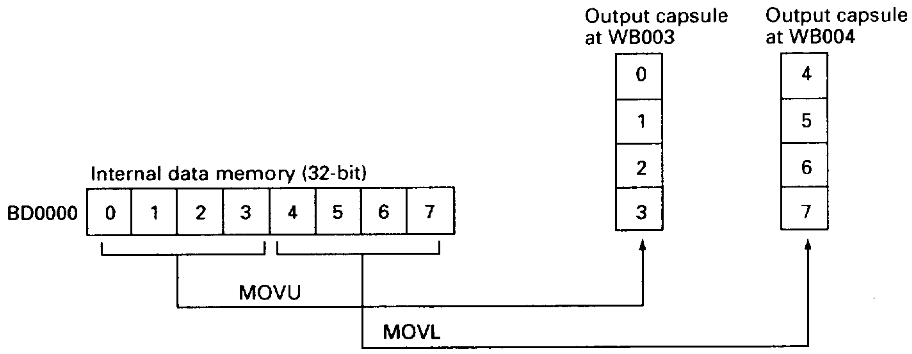
ONE-POINT ADVICE External output circuit for BCD 8-digit data

An example of using MOVU and MOVL is shown below.

1. **Instruction**
   MOVU, MOVL

2. **Operation**
   Output of BCD 8 digits data

3. **System diagram**



4. **Program**

```
  F54
 |-||--[ BD0    MOVU    WB3                              ]  -| High-order digits transfer
 |     [ BD0    MOVL    WB4                              ]  -| Low-order digits transfer
 |                                                          -|
 |                                                           |
```

• Example of monitoring with loader LITE

① Program monitoring

```
P0001|F0054    01234567          00000123
 0001|---||---[ BD0000     MOVU  WB0003              ]     -|
P0001|        | 01234567          00004567           |     -|
 0002|-       [ BD0000     MOVL  WB0004              ]     -|
                                                          -|
```

② Data monitoring

```
ADDRESS  BIN                 DEC          HEX       BCD
             ▼   ▼   ▼   ▼
 BD0000                   *             * 01234567  01234567 ←—  BD0 data are output to
 BD0001                   *             * 00000000  00000000     WB3 and WB4.
 B002    0000000000000000▲ 0000000000    00000000  00000000
 B003    0000000100100011  0000000291    00000123  00000123
 B004    0100010101100111  0000017767    00004567  00004567
 B005    0000000000000000  0000000000    00000000  00000000
```

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

9. Pattern clear (PC)

Instruction	Pattern clear	F175	Description	Example
Symbol	Unconditional execution ⊢[Z1 PC Wd : Z2] Conditional execution ⊢⊣[Z1 PC Wd : Z2] Z1: Clear patten Z2: Size Wd: Clear area first address		Clear area Z1 [ ] Wd [ ] } Z2 An operation error flag (A0041) is set ON and the program is not executed in the following cases. • When Z2 is specified by word, the contents of Z2 is not BCD code or a negative value. • The No. of remaining words of Wd is less than Z2. (If Z2 is directly specified "dXX", a user program error occurs.)	M0001 ⊢⊣[BD0001 PC BD0002 : d3] BD0001 12345 BD0002 12345 BD0003 12345 BD0004 12345 For the data shown on the left, the execution result is as follows.
Function	① The data (pattern) in Z1 is written to the Z2-word area (beginning with Wd).			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	–
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

10. Search (SRCH)

Instruction	Search	F176	Description	Example
Symbol	Unconditional execution ⊢[Z1 SRCH Wd:Ws:Z2]( )⊢ Conditional execution ⊢⊢[Z1 SRCH Wd:Ws:Z2]( )⊢ Z1: Search data Z2: Size Ws: First search address Wd: Detected address		Search area Z1 ▭ Ws … Z2 Wd ▭ Detected address	T0005 M0000 ⊢⊢[BD0120 SRCH BD0020:WM0001:d5]( )⊢
Function	① Data having the same contents as Z1 is searched for (in Z2 words) beginning with Ws and the result (presence or absence) is output to the relay. The detected address is also stored in Wd. Presence: The output relay is set ON. Absence: The output relay is set OFF. Detected address: The value to be stored is the number counted from the first search address. (The corresponding address is Ws + the contents of Wd) ② If there are multiple identical patterns, the address of the data found first is stored.		An operation error flag (A0041) is set ON and the program is not executed in the following cases. • When Z2 is specified by word, the contents of Z2 is not BCD code or a negative value. • The No. of remaining words of Wd is less than Z2. (If Z2 is directly specified "dXX", a user program error occurs.)	BD0001 1111 WM001 0002 BD0020 1200 1 0 2 1111 3 512 4 1111 (The actual address is BD0022.)

Operand and influence flag

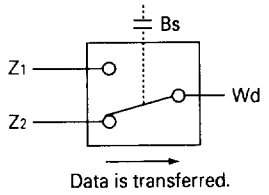
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z1, Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	–
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft. ware

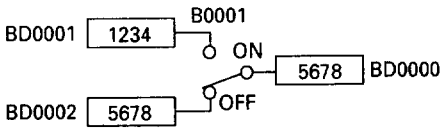
11. Switch (SW)

Instruction	Switch	F177	Description	Example
Symbol	⊢⊣[Bs : Z1 : Z2 SW Wd] Bs: Switching input		Z1 Bs (Switching input) ON Wd Z2 OFF Identifiers usable for Bs (Switching input) are shown on the table below. When P or PE-link is used, L can be also used.	⊢⊣[B0001 : BD0001 : BD0002 SW BD0000]⊢
Function	① Transfer source changes according to the switching input state. Switching input ON: Transfer from Z1 to Wd Switching input OFF: Transfer from Z2 to Wd (2) The transfer method from Z1 or Z2 to Wd is as same as that of signed transfer instruction (MOV).			



B	M	K	D	F	A	S	T
O	O	O	O	O	O	–	O
C	i	j	k	l	m	P	Q
O	O	O	O	O	O	O	O

O : Available
– : Not available



Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▭ : Applicable)												Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S		D05	D10S	D20	UTE	Soft-ware

12. Message transmission (MSGT)

Instruction	Message transmission	F182

Symbol

Unconditional execution

```
|----[MSGT N1, N2, Z, Wd ]
     [  DATA1~5          ]
```

Conditional execution

```
|-||-[MSGT N1, N2, Z, Wd ]
     [  DATA1~5          ]
```

Function

Message transmission is used to send data to the device not belonging to own station.

① The No. of words specified by DATA5 from the contents of the source area (first address) specified by DATA1, 2 are transmitted to the destination area (first address) specified by DATA3, 4.

② Information on transmission (status) is stored in Wd. After one transmission is complete, reset this status before perfoming the next transmission. Without resetting, the next transmission is disabled.

③ The maximum number of words transmitted is as follows: (1 word = 16 bits)

Direct bus	1018	words
T-link	108	words
P-link	108	words
PE-link	498	words
Direct access	1018	words
SUMINET	1002.5	words
Expansion T-link	108	words
ME-NET	108	words

Description

N_1: 1 = T-link
2 = P-link
3 = direct access
4 = SUMINET
5 = Expansion T-link
6 = ME-NET

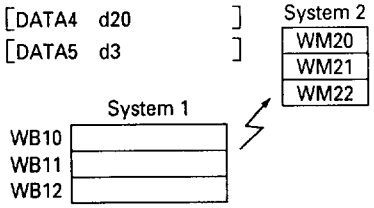
Note: Direct bus should be specified as T-link.

N_2 = Monitoring timer for use of SUMINET
Z = Station No.
Wd = Status word
DATA1 = Source data module No.
DATA2 = Source address
DATA3 = Destination data module No.
DATA4 = Destination address
DATA5 = Number of words transmitted

(DATA1 to 5 must be specified in decimal.)

Example

```
[MSGT1, 1, d30, WM0 ]
[DATA1  d0          ]
[DATA2  d10         ]
[DATA3  d1          ]
[DATA4  d20         ]
[DATA5  d3          ]
```



With the above instruction, three words are transmitted from WB10 in system 1 to WM areas in system 2 via station 30 on the T-link.

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				
Wd	O*	O	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Data module No. table

Name	Identifier	Data module No.
Input/output relay	B	0
Auxiliary relay	M	1
Keep relay	K	2
Special relay	F	3
Annunciator relay	A	4
Differential relay	D	5
Step control relay	S	8
0.1s timer current value area	W9	9
Timer setting value area (0.01s)	TS	10
Timer current value area (0.01s)	TR	11
Counter setting value area	CS	12
Counter current value area	CR	13
Data memory	BD	14

Name	Identifier	Data module No.
Direct access area	W24	24
Analog work area	W25	25
Differential relay preceding value area	W26	26
File area	W30 to W109	30 to 109
Calendar area	W125	125
P-link (Station 0) block No.1	WL	20
P-link (Station 0) block No.2	W21	21
P-link (Station 0) block No.3	W22	22
P-link (Station 0) block No.4	W23	23
P-link (Station 1) block No.1	W120	120
P-link (Station 1) block No.2	W121	121
P-link (Station 1) block No.3	W122	122
P-link (Station 1) block No.4	W123	123

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft. ware

13. Message reception (MSGR)

Instruction	Message reception	F183

Symbol

Unconditional execution

```
|--[MSGR N1:N2: Z:Wd ]
   [    DATA1~5       ]
```

Conditional execution

```
|-|-[MSGR N1:N2: Z:Wd ]
   [    DATA1~5       ]
```

Function

Message transmission is used to receive data from the device not belonging to own station.

① The No. of words specified by DATA5 from the contents of the source area (first address) specified by DATA1, 2 are transmitted to the destination area (first address) specified by DATA3, 4.

② Information on reception (status) is stored in Wd. After one reception is complete, reset this status before performing the next reception. Without resetting, the next reception is disabled.

③ The maximum number of words received is as follows: (1 word = 16 bits)

Direct bus	1018	words
T-link	108	words
P-link	108	words
PE-link	498	words
Direct access	1018	words
SUMINET	1004	words
Expansion T-link	108	words
ME-NET	108	words

Description

N_1: 1 = T-link
2 = P-link
3 = Direct access
4 = SUMINET
5 = Expansion T-link
6 = ME=NET

Note: Direct bus should be specified as T-link.

N_2 = Monitoring timer for use of SUMINET

Z = Station No.

Wd = Status word

DATA1 = Source data module No.
DATA2 = Source address
DATA3 = Reception data module No.
DATA4 = Reception address
DATA5 = Number of words received

(DATA1 to 5 must be specified in decimal)

Example

```
[MSGR1, 1, d30, WM0 ]
[DATA1  d0          ]
[DATA2  d0          ]
[DATA3  d1          ]
[DATA4  d10         ]
[DATA5  d3          ]
```

System 2: WB0, WB1, WB2

System 1: WM10, WM11, WM12

With the above instruction, three words are received from WB0 in system 2 to WM areas in system 1 via station 30 on the T-link.

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O				
Wd	O*	O	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	↑	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

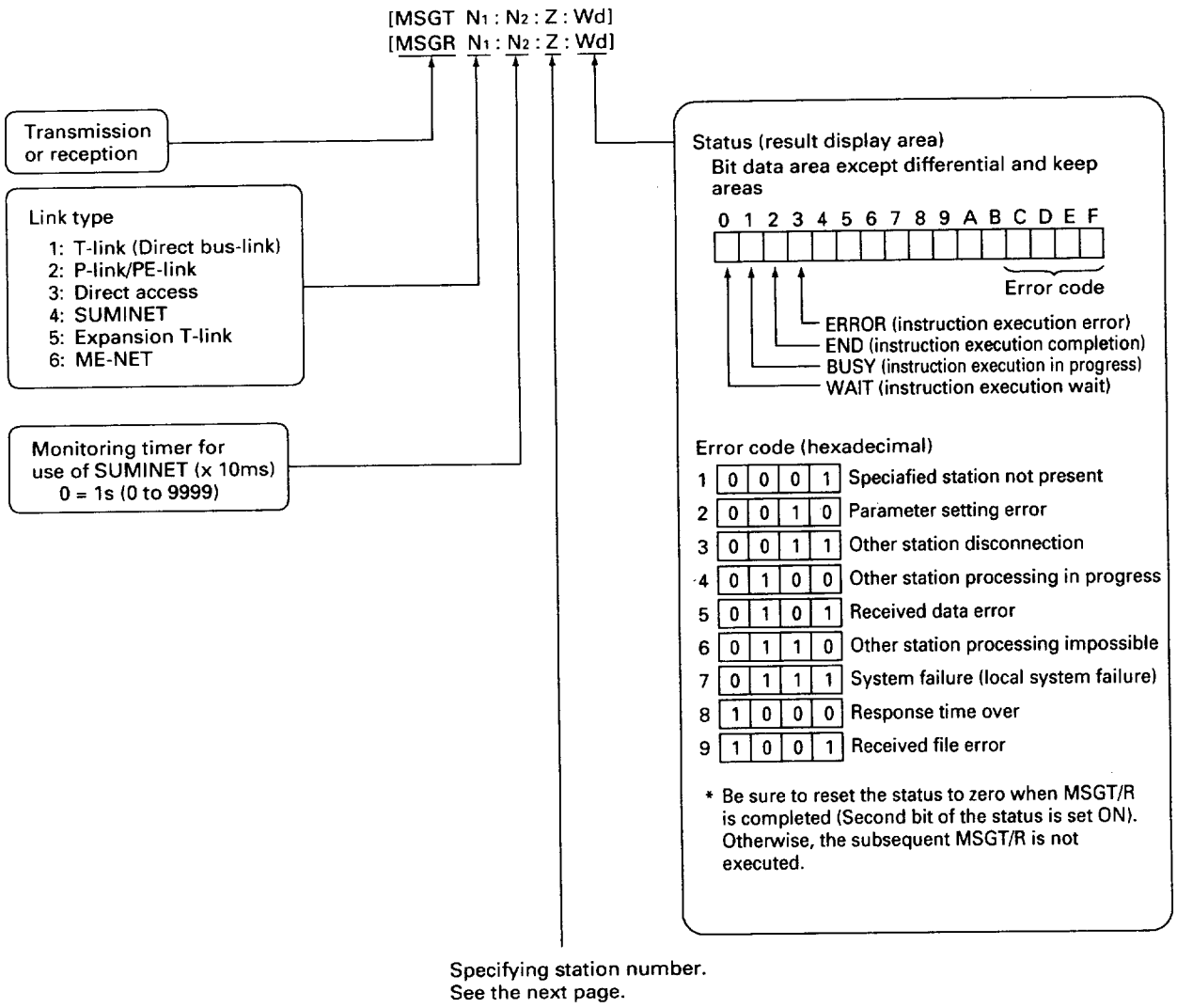
ONE-POINT ADVICE **Message transmission and reception (1)**

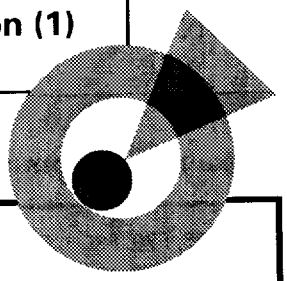
An example of communication between two units is shown below.

The F55, F70, F80H, F120H, F70S, F120S, F140S, and F150S Series enable data transfer between two units in one link with message communication instructions.

1. **Message communication instructions**
   MSGT: Transmission instruction
   MSGR: Reception instruction

2. **Meanings of instructions**





(continued)

Z: Specifying station number
Data is transmitted or received via the station with the specified number Z. (Specification depends on link types.)

Link name	Specification method	Configuration	Specifiable range
T-link (Direct bus-link)	Decimal	Channel No. | Station No. Station No.: 00 to 99 Channel No.: 00 to 03	d0000 to d0399
P-link	Hexadecimal	P-link No. | Station No. Station No.: 00 to 0F P-link No.: 00, 01	h0000 to h000F h0100 to h010F
PE-link	Hexadecimal	PE-link No. | Station No. Station No.: 00 to 3F PE-link No.: 00, 01	h0000 to h003F h0100 to h013F
Direct access	Decimal	Slot No. 0 to 9	d0 to d9
SUMINET	Hexadecimal	○○ | Network No. | Node address | Port No. Port No.: 00 Node address: 01 to 7E Network No.: 00 to 7F (0: Same network)	h00000100 to h007F7E00
Expansion T-link (Direct bus-link)	Decimal	Channel No. | Station No. Station No.: 00 to 31 Channel No.: 00 to 03	d0000 to d0031 d0100 to d0131 d0200 to d0231 d0300 to d0331
ME-NET	Hexadecimal	ME-NET No. | Station No. Station No.: 00 to 3F ME-NET No.: 00, 01	h0000 to h003F h0100 to h013F

Note: If Z indirectly specified on P-link/PE-link, SUMINET, or ME-NET, the data is recognized as decimal data. Use the binary/BCD conversion instruction to store the data.

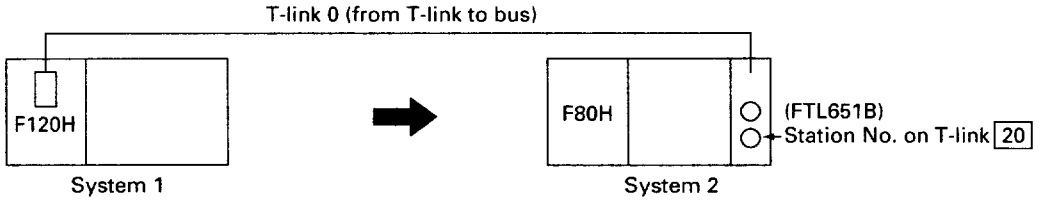
An example of communication between systems is shown below.

3. Meanings of data

[MSGT : N_1 : N_2 : Z : Wd]

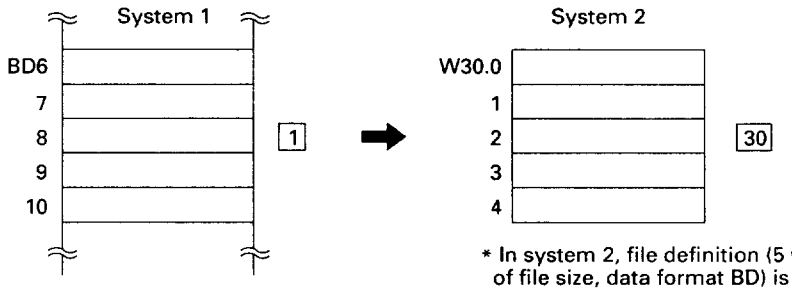
[DATA1]	→ Specifies the source data module No.	Source data
[DATA2]	→ Specifies the source relative address.	
[DATA3]	→ Specifies the destination data module No.	Destination data
[DATA4]	→ Specifies the destination relative address.	
[DATA5]	→ Specifies the No. of words transmitted. (This must follow the data format specified by DATA1, 2)	

4. Example program



Five words are transmitted from the data memory area BD6 to BD10 in system 1 to user file 30 in system 2.

```
    [ MSGT   1 : 0 : d20 : WM10 ]
    [ DATA   d14                ]   In file No. 14 (data module No. 14)
    [ DATA   d6                 ]   From BD6
    [ DATA   d30                ]   In user file 30 in system 2
    [ DATA   d0                 ]   To W30.0 through W30.4
M102[ DATA   d5                 ]   Number of words transferred: 5
    [ h0000  LMOV   WM10        ]   The data can be retransmitted or new data can be
                                    transmitted by resetting the status in WM10.
```



* In system 2, file definition (5 words or more of file size, data format BD) is necessary.

3-4-9 Analog instructions

1. Analog

- The eleven types of analog instructions (having different functions) are as follows. F30, F50, and F50H series do not support analog instructions.

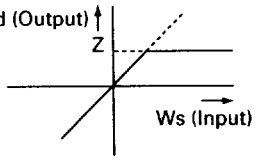
○: Available —: Not available

Instruction	Symbol	Function	F55, F60, F70, F80H, F120	F70S, F120S, F140S, F150S
(1) Upper limit 	Ws Wd : Z	A upper limit |Z| is set for the value of Ws and a value within this limit is transferred to Wd.	○	○
(2) Lower limit 	Ws Wd : Z	A lower limit |Z| is set for the value of Ws and a value within this limit is transferred to Wd.	○	○
(3) Upper and lower limit 	Ws Wd : Z_1 : Z_2	Sets the upper limit of the Ws value at Z_1 and the lower limit of the Ws at Z_2 for transfer to Wd.	—	○
(4) Dead band 	Ws Wd : Z	|Z| is processed as a dead band value. When Ws > |Z|, the result of (Ws - |Z|) is output to Wd. When Ws < -|Z|, the result of (Ws + |Z|) is output to Wd.	—	○
(5) Bias 	Ws Wd : Z	|Z| is processed as a bias value. When Ws > 0, the result of (Ws + |Z|) is output to Wd. When Ws < 0, the result of (Ws - |Z|) is output to Wd. When Ws = 0, 0 is output to Wd.	—	○
(6) Filter FIL	Ws FIL Wd : Z : N	Ws is filtered and output to Wd.		○
(7) Differential DIF	Ws DIF Wd : Z : N	Ws is differentiated and output to Wd.	—	○
(8) Integral INT	Ws INT Wd : Z : N	Ws is integrated and output to Wd.	—	○
(9) Sampling hold HOLD	Bs : Ws HOLD Wd : Z : N	While contact input is ON, Ws is being sampled. Just when contact input is OFF, Ws is held and output to Wd.	—	○
(10) Multi-percent MLTP	Z_1 MLTP Z_2 → Wd	Z_1 is multiplied by Z_2, and the result is divided by 100. The quotient is stored in Wd. The remainder is not saved.	—	○
(11) Divide-percent DIVP	Z_1 DIVP Z_2 → Wd	Z_1 is multiplied by 100, and the result is divided by Z_2. The quotient is stored in Wd. The remainder is not saved.	—	○

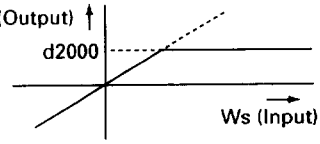
2. Upper limit (⊢F)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Upper limit	F110	Example
Symbol	⊢⊣[Ms ⊢F Wd : Z]		⊢⊣[WB0000 ⊢F WM0000 : d2000]
Function	① A upper limit (Z) is set for the value of Ws and a value within this limit is transferred to Wd.		



② In F60 series, W (file) cannot be specified as operand.
③ Ws should be BCD data.
④ The upper limit Z should be within Wd storage area.
16 bit data area:
-7,999 to 7,999
32 bit data area:
-79,999,999 to 79,999,999



① WB000 [1,500] → WM000 [1,500]
(BCD 4 digits) (BCD 4 digits)

② WB000 [2,500] → WM000 [2,000]
(BCD 4 digits) (BCD 4 digits)

Operand and influence flag

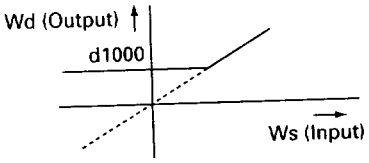
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▭ : Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

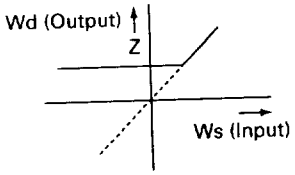
3. Lower limit (⊣/⊢)

Instruction	Lower limit	F111	Example
Symbol	⊣⊢[Ms ⊣/⊢ Wd : Z]		⊣⊢[WB0000 ⊣/⊢ WM0000 : d1000]



Function

A lower limit (Z) is set for the value of Ws and a value within this limit is transferred to Wd.



② In F60 series, W (file) cannot be specified as operand.
③ Ws should be BCD data.
④ The upper limit Z should be within Wd storage area.
16 bit data area:
-7,999 to 7,999
32 bit data area:
-79,999,999 to 79,999,999

① WB000 [1,500] → WM000 [1,500]
(BCD 4 digits) (BCD 4 digits)

② WB000 [500] → WM000 [1,000]
(BCD 4 digits) (BCD 4 digits)

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–				
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–					

*The input address of WB cannot be specified for Wd (operation result storage address).
**When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

4. Upper and lower limit (⊣/⊢)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

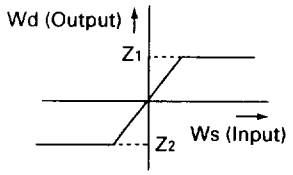
Instruction	Upper and lower limit	F112	Example

Symbol

⊢[Ms ⊣/⊢ Wd : Z1 : Z2]

Function

① An upper limit (Z_1) and a lower limit (Z_2) are set for the value of Ws and a value within these limits is transferred to Wd.

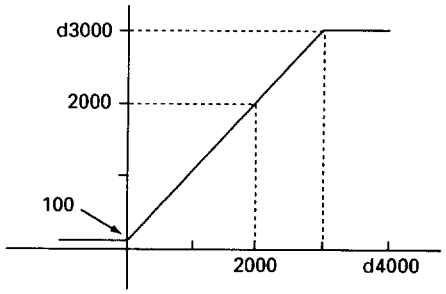


② If $Z_1 \leq Z_2$, the lower limit is used and the upper limit is ignored.
③ Ws should be BCD data.
④ The limit Z_1 and Z_2 should be within Wd storage area.
16 bit data area:
-7,999 to 7,999
32 bit data area:
-79,999,999 to 79,999,999

Example

⊢[BD0001 ⊣/⊢ BD0000 : d3000 : d100]⊢

For BD0001 [4000], BD0000 [3000].
For BD0001 [2000], BD0000 [2000].
For BD0001 [50], BD0000 [100].



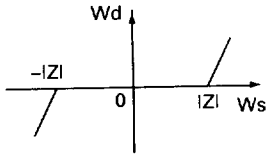
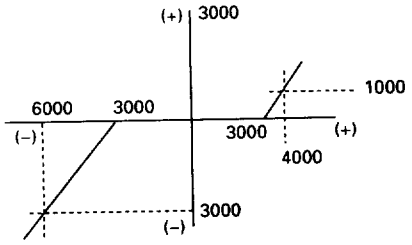
Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z1,Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

5. Dead band (⊣/⊢)

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140	F150	D05	D10S	D20	LITE	Soft-ware

Instruction	Dead band	F113	Example
Symbol	⊢[Ms ⊣/⊢ Wd : Z]		⊢[BD0001 ⊣/⊢ BD0000 : d3000]⊢
Function	① |Z| is processed as a dead band width. When Ws > |Z|, the result of (Ws - |Z|) is output to Wd. When Ws < -|Z|, the result of (Ws + |Z|) is output to Wd. In other cases, 0 is output to Wd.  ② Ws should be BCD data. ③ The dead band width Z should be within Wd storage area. 16 bit data area: 0 to 7,999 32 bit data area: 0 to 79,999,999		For BD0001 4000, BD0000 1000. For BD0001 2000, BD0000 0. For BD0001 -6000, BD0000 -3000. 

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Note on programming

When Ws indicates a 32-bit data area and Wd specifies a 16-bit data area, data may overflow depending on the Ws and Z set values.

Example :

Ws = BD0	Z	Wd = WM0
70000	d100	7999

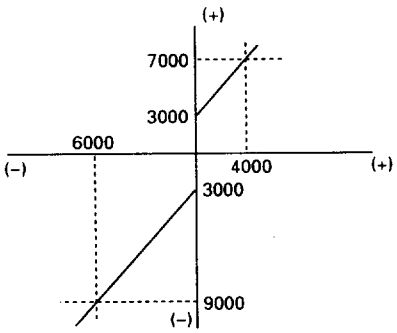
← Overflow flag A40 is set to ON and the maximum storage value 7999 is stored.

Processor (▒▒ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F190S	D05	D10S	D30	LITE	Soft-ware

6. Bias (⊣/⊢)

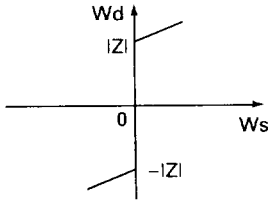
Instruction	Bias	F114	Example
Symbol	⊢⊢[Ws ⊣/⊢ Wd : Z]		⊢[BD0001 ⊣/⊢ BD0000 : d3000]⊢

For BD0001 [4000], BD0000 [7000].
For BD0001 [2000], BD0000 [0].
For BD0001 [-6000], BD0000 [-9000].
For BD0001 [79999999], BD0000 [79999999] and overflow flag A40 is set ON.



Function

① |Z| is processed as a bias value.
When Ws > 0, the result of (Ws + |Z|) is output to Wd.
When Ws < 0, the result of (Ws - |Z|) is output to Wd.
When Ws = 0, 0 is output to Wd.



② Ws should be BCD data.
③ The bias value Z should be within Wd storage area.
16 bit data area: 0 to 7,999
32 bit data area: 0 to 79,999,999
However, some Ws value may cause an overflow. The change of Ws value should be considered when setting Z.

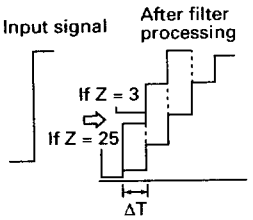
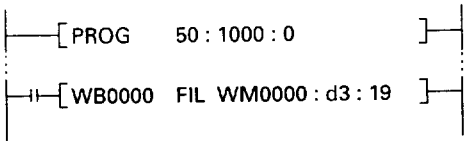
Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

7. Filter (FIL)

Instruction	Filter	F115	Description	Example
Symbol	⊢⊣⊢[Ms FIL Wd : Z : N] Z: Time constant for filter/ΔT [ΔTxZ = Time constant for filter L ΔT: Sample cycle of fixed-cycle processing] N: Analog work area number (0 to 255)		① This instruction is used with the fixed cycle program. ② The processing method is as follows: Wdn = {(Ws + Rn-1 - Wdn-1) /Z} + Wdn-1 Wdn and Wdn-1 are stored at every sample cycle of fixed cycle processing. Rn-1 indicates the remainder of the previous calculated value. ③ When time constant for filter = 30ms and fixed cycle ΔT = Δ10ms, Z becomes 3. 	Fixed-cycle level 2 interrupt processing program: 1 second (1000ms) cycle  Time constant for filter: 3 seconds
Function	① Ws is filtered and output to Wd. ② The same analog work area number N must not be used in other analog instructions. If the same number N is used, a user ③ program error will occur. If N is outside the specified range, ④ a user program error will occur. When Z<0, an operation execution error will occur.			

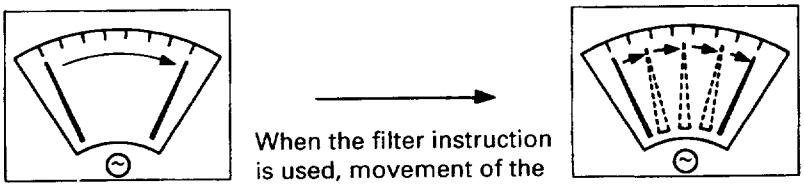
Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				

*The input address of WB cannot be specified for Wd (operation result storage address).
**When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Programming advice

The filter instruction is used to protect the analog meter from abrupt fluctuation caused by steep data changes when the PC analog output is connected to the analog meter.



When the filter instruction is used, movement of the analog meter indicator needle is smoothed.

8. Differential (DIF)

Processor (▒ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Differential	F116	Description	Example
Symbol	⊢⊣[Ws DIF Wd : Z : N] Z: ΔT/differential time [ΔT/Z = differential time ΔT: Sample cycle of fixed-cycle processing] N: Analog work area number (0 to 255)		① This instruction is used with the fixed-cycle program. ② The same analog work area number N must not be used in other analog instructions. If the same number N is used, a user program error will occur. ③ If N is outside the specified range, a user program error will occur.	Fixed-cycle level 2 interrupt processing program: 1 second (1000ms) cycle ⊢[PROG 50 : 10000 : 0]⊢ B0012 ⊢⊣[WB0000 DIF WM0000 : d5 : 127]⊢ Differential time: 0.2 second
Function	① Ws is differentiated and output to Wd. The processing method is as follows: Wd = (Wsn - Wsn-1)/Z Wsn and Wsn-1 are stored at every sampe cycle of fixed-cycle processing ② If Z < 0, an operation execution error will occur.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				

*The input address of WB cannot be specified for Wd (operation result storage address).
**When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

9. Integral (INT)

Processor (▨: Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D08S	D20	LETE	Software

Instruction	Integral	F117	Description	Example
Symbol	⊢⊣⊢[Ws INT Wd : Z : N] Z: Integral time/ΔT [ΔT x Z = Integral time ΔT: Sample cycle of fixed-cycle processing] N: Analog work area number (0 to 255)		① This instruction is used with the fixed-cycle program. ② The same analog work area number N must not be used in other analog instructions. If the same number N is used, a user program error will occur. ③ If N is outside the specified range, a user program error will occur.	Fixed-cycle level 2 interrupt processing program: 1 second (1000ms) cycle ⊢⊣⊢[PROG 50 : 1000 : 0]⊢ B0011 ⊢⊣⊢[WB0000 INT WM0000 : d10 : 99]⊢ Integral time: 10 seconds
Function	① Ws is integrated and output to Wd. The processing method is as follows: Wdn = {(Ws + Rn-1)/Z} + Wdn-1 Wdn and Wdn-1 are stored at every sample cycle of fixed-cycle processing. Rn-1 indicates the remainder of the previous calculated value. ② If Z < 0, an operation execution error will occur.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

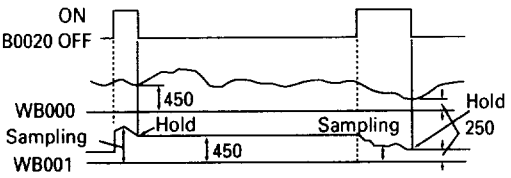
10. Sampling hold (HOLD)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

Instruction	Sampling hold	F118	Description	Example
Symbol	⊢⊣⊢[Bs : Ws HOLD Wd : N] Bs: Sample hold timing input N: Analog work area number (0 to 255)		① The following identifiers can be used for sample hold timing input (Bs).	⊢[B0020 : WB0000 HOLD WB0001 : 5]⊢
Function	① When the contact input is ON, Ws is read. While contact input is OFF, Ws is held and output to Wd.		② The same analog work area number N must not be used in other analog instructions. If the same number N is used, a user program error will occur. ③ If N is outside the specified range, a user program error will occur.	

B	M	K	D	F	A	S	T
O	O	O	O	O	O	–	O
C	i	j	k	l	m	P	Q
O	O	O	O	O	O	O	O

O: Available
–: Not available



Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

11. Multi-percent (MLTP)

Processor (▒ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft. ware

Instruction	Multi-percent	F119	Example
Symbol	⊢⊣⊢[Z1 MLTP Z2 → Wd]		M001A ⊢⊣⊢[BD0000 MLTP BD0001 → BD0002]⊢ For BD0000 = 500 and BD0001 = 125, instruction results in BD0002 = 625. $\frac{500 \times 125}{100} = 625$
Function	① Z1 is multiplied by Z2, the result is divided by 100, and the quotient is stored in Wd. The remainder is not saved. ② If the result exceeds the data range of Wd, the overflow relay is turned ON and the maximum (minimum) value is stored in Wd.		

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

12. Divide percent (DIVP)

Processor (▨: Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

Instruction	Divide percent	F120	Example
Symbol	⊢⊣⊢[Z1 DIVP Z2 → Wd]		M001D ⊢⊣⊢[BD0000 DIVP BD0001 → BD0002]⊢ For BD0000 = 625 and BD0001 = 12500, instruction results in BD0002 = 5 . $\frac{625 \times 100}{12500} = 5$
Function	① Z1 is multiplied by 100, the result is divided by Z2, and the quotient is stored in Wd. The remainder is not saved. ② If the result exceeds the data range of Wd, the overflow relay is turned ON and the maximum (minimum) value is stored in Wd.		

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	S	Z	E	O
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–				

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

3-4-10 Trigonometric functions

1. Trigonometric

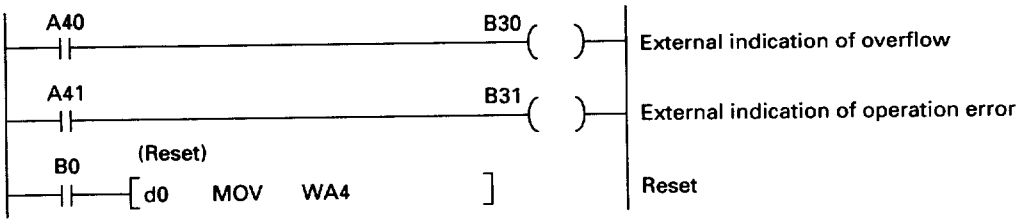
- The six types of functions (having different functions) are as follows. Trigonometric function instructions are supported by F70S, F120S, F140S, and F150S series.

Instruction	Abbreviation	Symbol	Function
(1) SIN	SIN	├[Z SIN Wd]	The sine, cosine, and tangent are calculated. Input range: ±360° step 0.1° Input unit: Value increased by 10 times Output unit: Value increased by 10000 times
(2) COS	COS	├[Z COS Wd]	
(3) TAN	TAN	├[Z TAN Wd]	
(4) ASIN	ASIN	├[Z ASIN Wd]	The arcsine, arccosine, and arctangent are calculated. Output: ±90°, 0 to 180°, ±90° accuracy 0.1° Output unit: Value increased by 10 times Input unit: Value increased by 10000 times
(5) ACOS	ACOS	├[Z ACOS Wd]	
(6) ATAN	ATAN	├[Z ATAN Wd]	

- **Data formats and operation execution**

① All operations are executed using the signed BCD 8-digit format (regardless of the amount of source data).
② If an error flag (A0041) is set ON, operation execution is inhibited. *Example: the source data does not use the BCD format)
③ If the operation result exceeds the capacity of the storage destination, only the maximum capacity of the destination is stored and the overflow flag (A0040) is set.
④ If the operation result is a negative value, the sign flag (F004E) is set ON.
⑤ If the operation result is zero, the zero flag (F004F) is set ON.
⑥ The sign flag (F004E) and zero flag (F004F) are set ON and OFF for every execution of instructions in a program.
⑦ If the overflow flag (A0040) and operation error flag (A0041) are set ON, these flags stay ON until power is turned OFF or until they are reset by the user program. These flags also stay ON until they are reset by operation from the program loader. Operation continues regardless of whether these flags are ON or OFF.

Example: Error indication and resetting by user program



2. Trigonometric functions (SIN, COS, TAN)

Processor (▒ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F195S	D05	D10S	D20	LITE	Software

Instruction	Trigonometric functions	F90, F91, F92	Description	Example
Symbol	⊢⊣⊢[Z SIN Wd] ⊢⊣⊢[Z COS Wd] ⊢⊣⊢[Z TAN Wd]		① TAN90° is output as a maximum positive value. ② If input is 10.5 degrees, input data is 105.	B001E ⊢⊣⊢[BD0000 TAN BD0010]⊢ For BD0000 [600], instruction execution results in BD0010 [17320]. BD0000 means 60°, and BD0010 means 1.7320.
Function	① The sine is calculated. ② The cosine is calculated. ③ The tangent is calculated. Input range: ±360° step 0.1° Input unit: Value increased by 10 times Output unit: Value increased by 10000 times Output accuracy: Rounded to 0.0001			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																										S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–				
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	–	–	↕	↕	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

3. Inverse trigonometric functions (ASIN, ACOS, ATAN)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Inverse trigonometric functions	F93, F94, F95	Description	Example
Symbol	⊢⊣⊢[Z ASIN Wd] ⊢⊣⊢[Z ACOS Wd] ⊢⊣⊢[Z ATAN Wd]		① ASIN and ACOS accuracy i) Input: -0.9 to 0.9 Output accuracy: 0.1° ii) Input: 0.9 to 1.0 -0.9 to -1.0 ② If input > 1 or input < -1 at ASIN and ACOS, an operation error occurs and operation is disabled.	B001E ⊢⊣⊢[BD0000 ATAN BD0010]⊢ For BD0000 of 17320, instruction execution results in BD0010 600. BD0000 means 1.7320, and BD0010 means 60°.
Function	① The arcsine is calculated. ② The arccosine is calculated. ③ The arctangent is calculated. Input unit: Value increased by 10000 times Output unit: Value increased by 10 times Output range: ±360° accuracy 0.1°			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	↕	↕	↑	↑
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↕	↕	↑	↑

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

3-4-11 Files

1. File instructions

- The types of file instructions (having different functions) are as follows.

Some of these instructions operate independently, while others operate in combination.

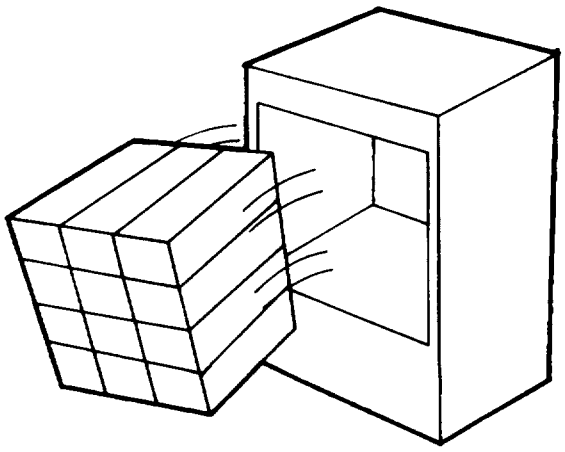
Instruction \ Application		Write/read file *1	Write/read file *2	Tracking file	Read-only file	F30 to F50H	F60	F55, F80H, F70, F120H, F120S, F140S, F150S
(1) File definition	(FILE)	●		●		Provided	Provided	Provided
(2) Selector	(SEL)	○	○	○	○	Provided	Provided	Provided
(3) Deselector	(DSEL)	○	○	○		Provided	Provided	Provided
(4) File store	(FFST)			●			Provided	Provided
(5) FIFO load	(FIFO)			○			Provided	Provided
(6) FILO load	(FILO)			○			Provided	Provided
(7) File clear	(FLCL)	○	○	○		Provided	Provided	Provided
(8) File read	(RFIL)	○		○	○			Provided
(9) File write	(WFIL)	○		○				Provided
(10) File information	(FINF)			○				Provided
(11) Data table definition	(TABL)				●	Provided	Provided	Provided
(12) Data	(DATA)				●	Provided	Provided	Provided
(13) Data end	(DEND)				●	Provided	Provided	Provided

● : Indicates an instruction required for the corresponding application.
○ : Indicates an instruction to be optionally used for the corresponding application.
*1 : Using user files
*2 : Using system files (data modules numbered 0 to 26, 120 to 123, 125)

Examples of applications (concept)

(1) FLCL

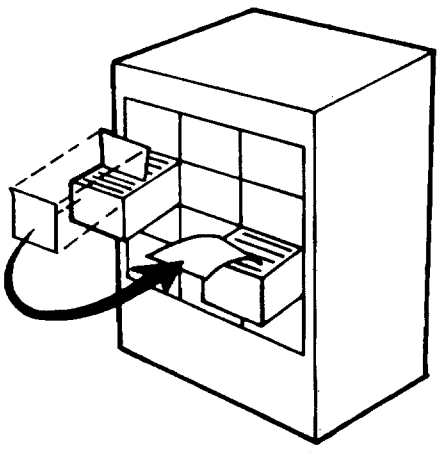
The FLCL instruction is executed.



When the FLCL instruction is executed, all information (data) is deleted (cleared to 0) from the corresponding file.

(2) SEL, FILE

Information is stored.

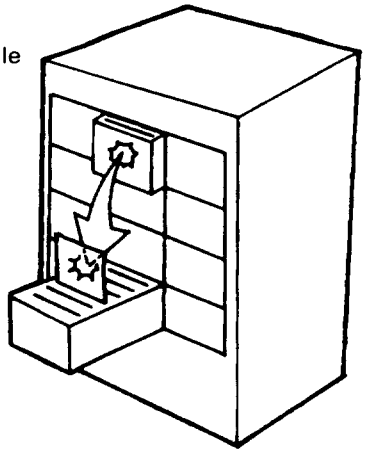


Note: Indirect specification is possible

Information on the specified page of the specified file is copied and stored at the specified word address of the specified memory.

(3) DSEL, FILE

Storage in file

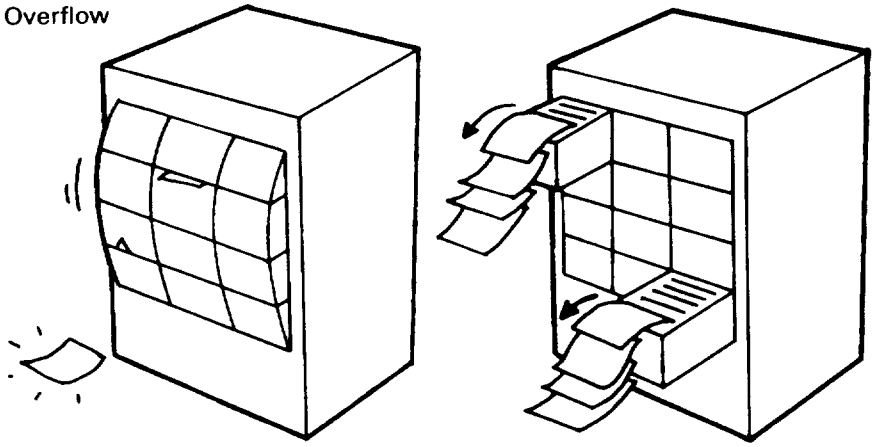


Note: Indirect specification is possible.

Data at the specified word address of specified memory is copied and stored on the specified page of specified file.

(4) FFST, FIFO or FILO

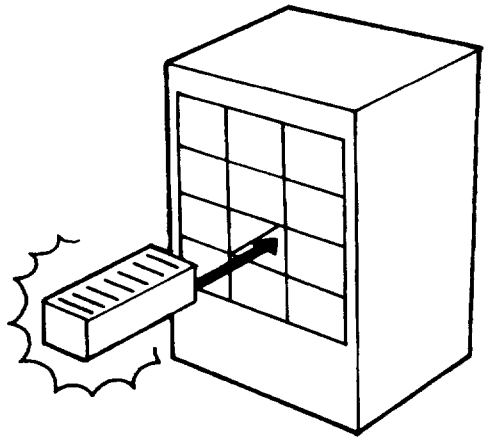
Overflow



Stored from the top, fetched from the bottom

Data is stored in the specified file. If the FIFO instruction is used, the data stored first is fetched first.
If the FILO instruction is used, the data stored last is fetched first.

(5) TABL, DATA, DEND, SEL



A user program is used to generate a file and the information to be stored in the file.
The SEL instruction is used to copy the data stored on a specified page and store it at the specified word address of the specified memory.

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft ware

2. File definition (FILE)

Instruction	File definition	F193

Symbol

⊢[FILE $N_1 : N_2 : N_3 : X$]

N_1: File No.
N_2: X size
N_3: Y size
X: Data format (SI, DI or BD)

Function

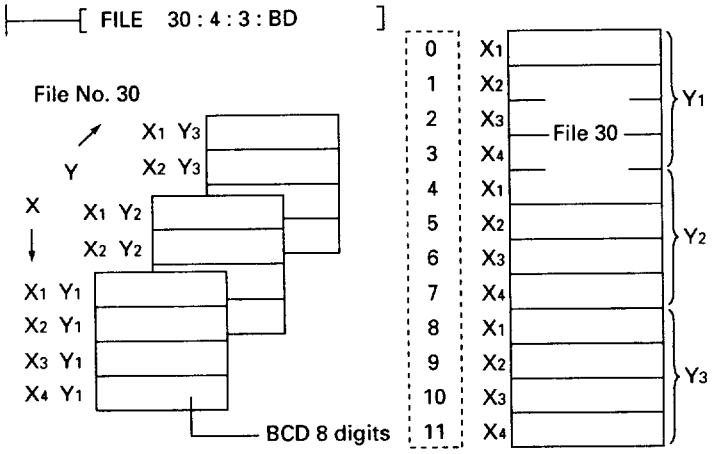
① A user file is registered and its area is assigned in the file data area.
② The FILE instruction only acquires an area when the processor is started or the program is modified. It executes no processing during operations.
③ Data format

SI	0	Binary 16-bit data
DI	1	Binary 32-bit data
BD	2	BCD 8 digits
D20, LIETE	D05, D10S	

Example

(1) Example

⊢[FILE 30 : 4 : 3 : BD]



(2) ① A file difnition may be written anywhere in the program. (If the definition is written after the PEND instruction, the scan time can be reduced.)
② Modification of file definition
- If X is modified, data is cleared.
- If only Y is modified, data is retained.
- If the data format is modified, data is cleared.

③ The word addresses to be used for the SEL and DSEL instructions are shown as enclosed in the dotted line " ⋯⋯ ".

(3) A program erro occurs in the following cases.
① When a file No. is double -assigned or an undefined file No. is specified.
② When a setting does not conform to the rules described in the left column.

Operand and influence flag	Influence flag					
	S	Z	E	O	FE	FF
	–	–	–	–	–	–

S: F004E E: A0041 FE: F0047
Z: F004F O: A0040 FF: F0046

File No. available for each MICREX-F series

F30, F50, F50H, F60	F55, F70, F80H, F120H, F70S, F120S, F140S, F150S
30 to 45	30 to 109

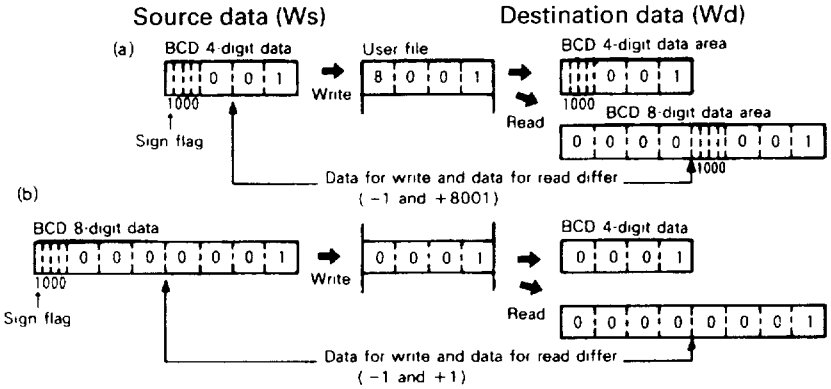
File size for each MICREX-F series

Series	N x N	
	1 word = 16 bits	1 word = 32 bits
F30, F50, F50H	128	64
F60	1024	512
F80H	4095	3840
F120H	4095	4095
F70S, F120S, F140S, F150S	4096	4096

Notes on data format, writing and reading of user files

The data format of a user file is defined as SI, DI or BD.
User file data is written and ready by using such instructions as FFST, FIFO, FILO, SEL and DSEL. The data differences in terms of the set data format and I/O area sizes are as follows.

1. When user file data format is designated as SI (16-bit long binary data)



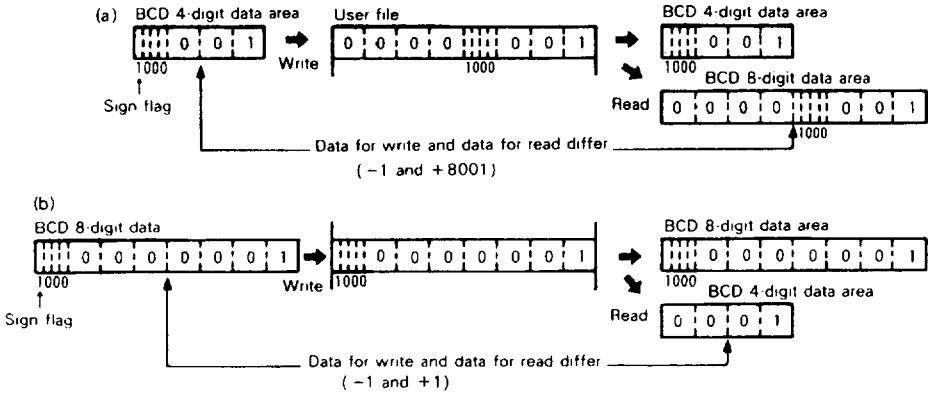
Monitoring on program loader (BCD)

Ws	Wd
-00000001	-00000001
-00000001	00008001
-00000001	00000001
-00000001	00000001

As shown in the above figure, Ws and Wd have different contents when the I/O area sizes are different.

The result of this operation becomes the same as that of the LMOV instruction.

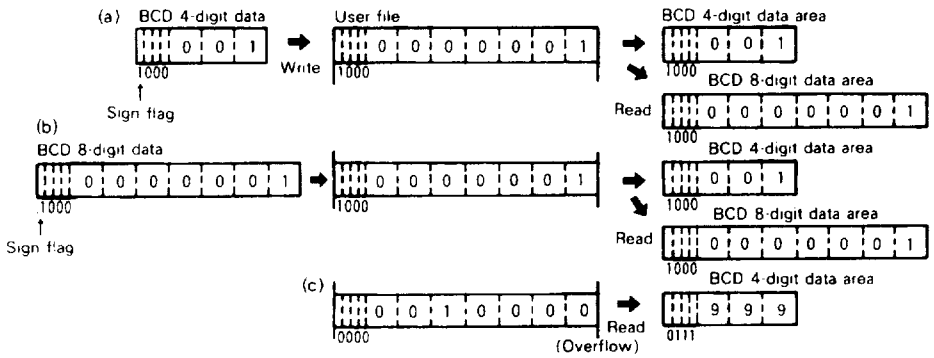
2. When user file data format is designated as DI (32-bit long binary data)



Ws	Wd
-00000001	-00000001
-00000001	00008001
-00000001	-00000001
-00000001	00000001

The result of this operation becomes the same as that of the LMOV instruction.

3. When user file data format is designated as BD (BCD 8 digits)



Ws	Wd
-00000001	-00000001
-00000001	-00000001
-00000001	-00000001
-00000001	-00000001
00010000	00007999

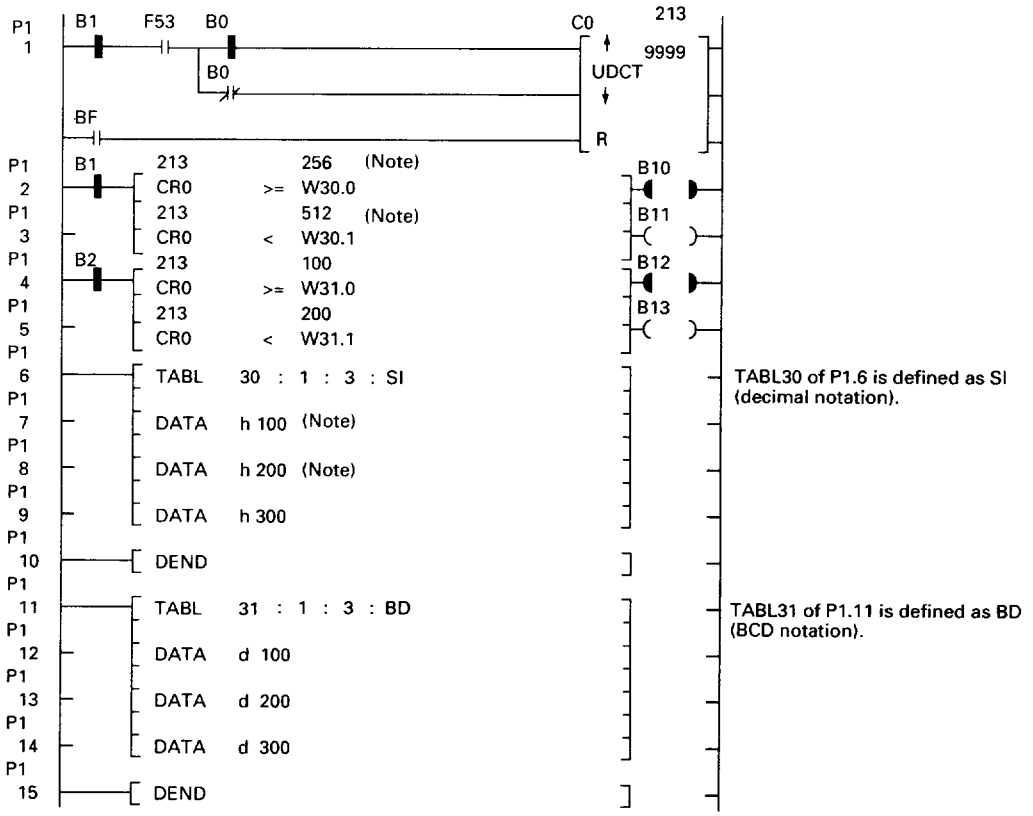
As shown in the above figure, the operation becomes the same as that of the MOV instruction. Be careful and read data to the data area having different data length.

ONE-POINT ADVICE **User file display differences depending on data format**

User file data format and its monitor display

The data format of a user file is defined as SI, DI or BD.

The comparison circuit between data of user files (30 and 31) and data of counter C0 is monitored as shown below.



Data memory

Address	Binary 0123456789ABCDEF	Decimal	Hexadecimal	BCD
W30. 0	0000000100000000	256	100	100
1	0000001000000000	512	200	200

Note:
Regardless of d (BCD) or h (Hex) being designated in the DATA statement, the data is recognized identical. (d100=h100)

However, when a ladder program is monitored, the data designated h (Hex) is displayed in decimal notation.

Processor (▭ : Applicable)												Program loader				
F30	F60 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

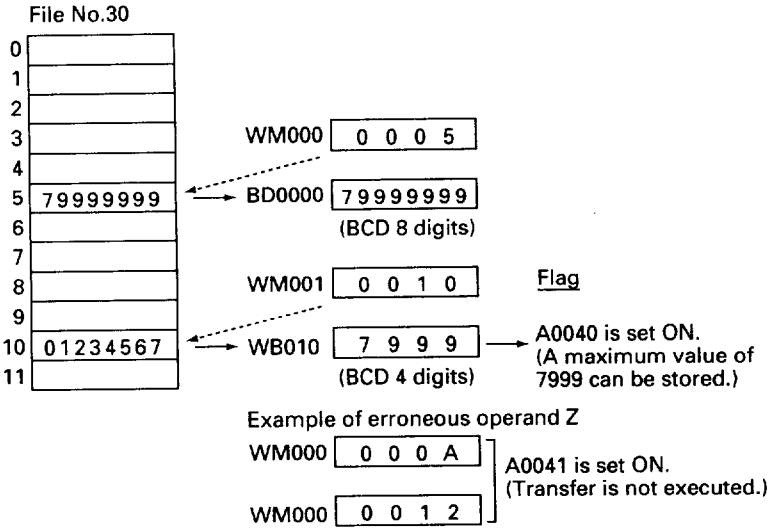
3. Selector (SEL)

Instruction	Selector	F195
Symbol	⊢[N : Z SEL Wd] N: Data module No. or file No.	
Function	① Data at the word location specified by Z in file No. N is transferred to Wd. File N: 0 to 125 O Z Wd ② The file data in SI or DI format is transferred as binary data. Data in other formats is transferred as BCD data. ③ If Z is specified as a hexadecimal direct number, it is handled as BCD data. ④ Even when the SEL instruction is executed, the file pointer is not changed. ⑤ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

Example

(1) Example

⊢[30 : WM0000 SEL BD0000]
[30 : WM0001 SEL WB0010]



(2) Conditions of setting flags
① Data Z is not BCD data. (E: A0041)
② The transfer data exceeds the maximum value of Wd. (O: A0040)
③ Data Z is larger than the maximum address of the file. (E: A0041)

(3) A program error occurs in the following cases.
① An undefined user file is specified, or specified module No. is not present.
② Z is directly specified as value of d12 or higher (as shown in the above example.)

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag					
																											S	Z	E	O	FE	FF
Z	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O						
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↑	↑	–	–

*The input address of WB cannot be specified for Wd (operation result storage address).
**When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

4. Deselector (DSEL)

Processor (▨ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F190S	D05	D10S	D20	LITE	Software

Instruction	Deselector	F196
Symbol	⊢[Z_1 DSEL N : Z_2] N: Data module No. or file No.	
Function	① Z_1 is transferred to the position specified by Z_2 in file No.N File N: 0 to 125 Z_1 → Z_2 ② The file data in SI or DI format is transferred as binary data. Data in other formats is transferred as BCD data. ③ If Z_2 is specified as a hexadecimal direct value, it is handled as BCD data. ④ Even when the DSEL instruction is executed, the file pointer is not changed. ⑤ In F30, F50, F50H, and F60 series, W (file) cannot be specified as operand.	

Example

(1) Example

⊢⊣⊢[BD0000 DSEL 30 : WM0000]

File No.30

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	7999
11	

WM0000 10 (BCD 4 digits)

BD0000 7999 (BCD 8 digits)

Example of erroneous operand Z_2

WM0000 0 0 0 A

WM0000 0 0 1 2

Flag: A0041 is set ON. (Transfer is not executed.)

(2) Conditions of setting flags
① Data Z_2 is not BCD data. (E: A0041)
② The data Z_2 is larger than the maximum word address of the file. (E: A0041)
③ The transfer data is larger than the maximum number of N. (O: A0040)

(3) A program error occurs in the following cases.
① An undefined user file is specified, or specified module No. is not present.
② Z_2 is specified as a direct value of d12 or higher (as shown in the above example.)
③ A data table No. is specifiec by N.

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag					
																											S	Z	E	O	FE	FF
Z_1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	↑	–	–
Z_2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	↑	–	–

** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

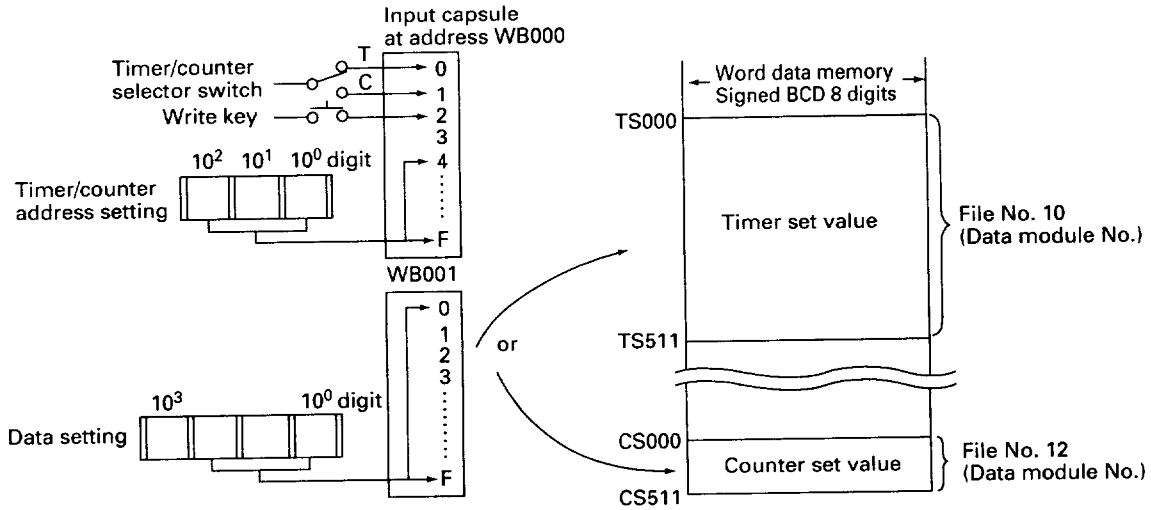
ONE-POINT ADVICE **Indirect timer/counter specification circuit**

An example of using the DSEL file instruction is shown below.

1. Instruction
AND, DSEL

2. Operation
Free setting of timer/counter via external digital switch without using a loader.

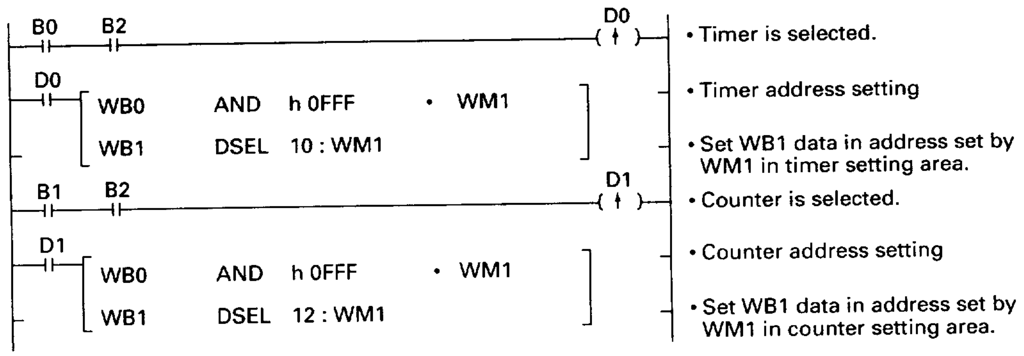
3. System diagram



Flow of setting

Select either timer or counter setting → Set timer/counter data → Set timer/counter address → Press write key and data are stored in specified address.

4. Program



Checking: After storing the above sequence, set the timer set value (TS) or counter set value (CS) while displaying the internal data with the loader.

```
ADDRESS  BIN              DEC      HEX      BCD
          ▼   ▼   ▼   ▼
 TS0000                  *        *        * 00001234  ← Data are set in TS0, 1 area.
 TS0001                  *        *        * 00002345
 TS0002                  *        *        * 00000000
 TS0003                  *        *        * 00000000
 CS0000                  *        *        * 00003456  ← Data are set in CS0, 1 area.
 CS0001                  *        *        * 00004567
 CS0002                  *        *        * 00000000
 CS0003                  *        *        * 00000000
```

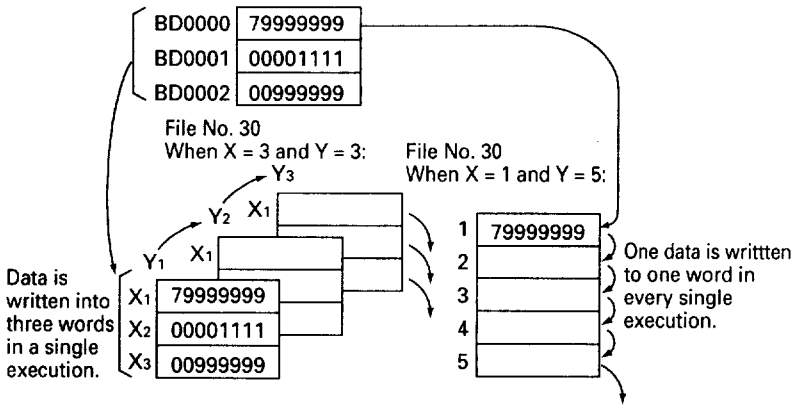
5. File store (FFST)

Processor (▒▒ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Insruction	File store	F190	Example
Symbol	⊢⊣⊢[Z FFST N] N: File No.		(1) Example ⊢⊣⊢[BD0000 FFST 30]

Function

① Data Z is stored in file N. The stored data is shifted by one digit and the file pointer is incremented by one. (When The X size is 1)
② Data that has size X beginning with Z is stored in file N. The stored data is shifted by X and the file pointer is incremented by one.
③ If the FFST instruction is executed when file is full, data is discarded in the order the data stored.
④ If a constant is specified as Z, the same constant is written to all X-size areas.
Note: The file pointer is information that indicates the position of the data written by the FFST instruction. The FIFO instruction reads the data block indicated by the file pointer.
⑤ In F60 series, W (file) cannot be specified as operand.



(2) Conditions for setting flags
① When the file pointer becomes Y, the FF (file fill) flag is set ON.
② When the FFST instruction is executed, the FE (file empty) flag is set OFF. This flag is set ON when file pointer becomes 0.

(3) A program error occurs in the following cases.
① A value of 29 or less, 110 or more, expansion module No. 30 to 109, or a data table No. is specified as N.
② The length of the memory area beginning with the Z-specified address is less than size X. (For example, WB509 is specified as Z when X=3.)

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag					
Z	O	O	O	O	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O	FE	FF
																											–	–	↑	–	↓	↑

*When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Processor (▭: Applicable)												Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

6. FIFO load (FIFO)

Instruction	FIFO	F191
Symbol	⊢⊣⊢[Z FIFO Wd] N: File No.	

Function

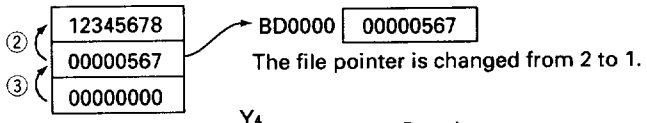
① The data having X-size indicated by the file pointer is transferred from file N to the area beginning with Wd and the file pointer is decremented by one. The file data remains unchanged after being transferred.
② If the FIFO instruction is executed when the file pointer is 0, data is not transferred and the E flag is set ON.
③ N must be a file written by using the FFST instruction.
This instruction can be used with the DSEL and WFIL instructions.
④ In F60 series, W (file) cannot be specified as operand.

Example

(1) Example

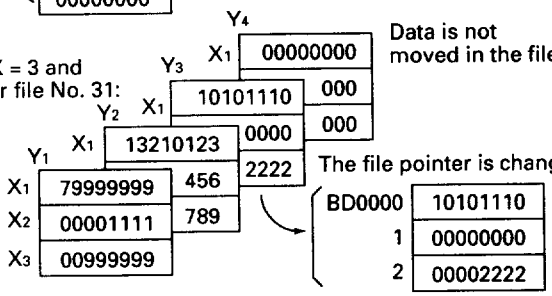
⊢⊣⊢[31 FIFO BD0000]

When X = 1 and Y = 3 for file No. 31



The file pointer is changed from 2 to 1.

When X = 3 and Y = 4 for file No. 31:



Data is not moved in the file.

The file pointer is changed from 3 to 2.

(2) Conditions for setting flags
① When the file pointer becomes 0, the FE flag is set ON.
② When the file pointer is 0 and the FIFO instruction is executed, an operation error flag is set ON.
③ When the FF flag is ON and the FIFO instruction is execute, the FF flag is set OFF.
(4) If Wd overflows, the 0 flag is set ON.

(3) A program error occurs in the following cases.
① A value of 29 or less, 110 or more, expansion module No. 30 to 109, or a data table No. is specified as N.
② The length of the memory area beginning with the Z-specified address is less than size X. (For example, WB509 is specified as Z when X=3.)

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag						
Wd	O	O	O	–	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O	FE	FF
																											–	–	↑	↑	↑	↓

*When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

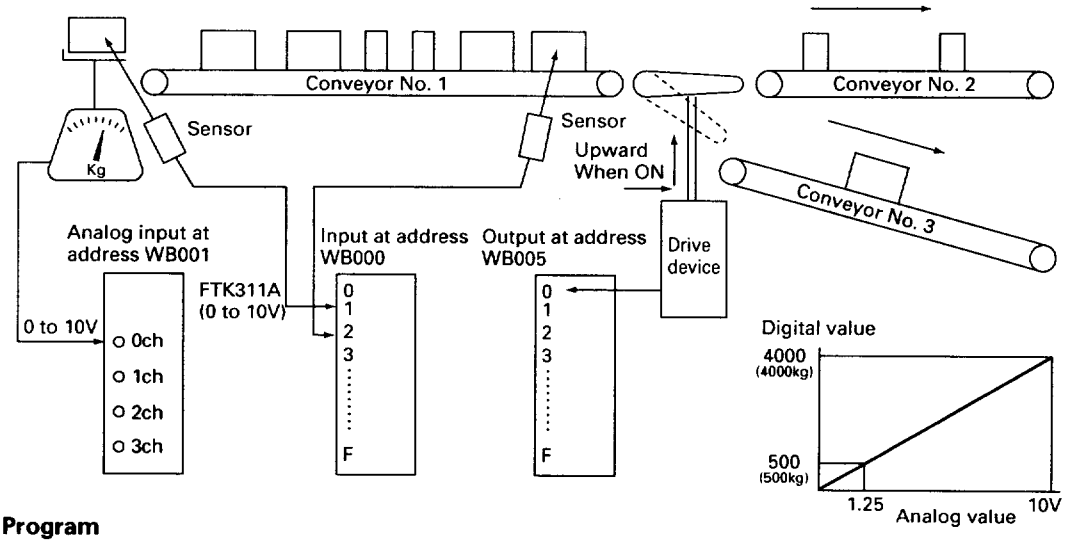
ONE-POINT ADVICE **Tracking control circuit**

An example of tracking control using file instructions is shown below.

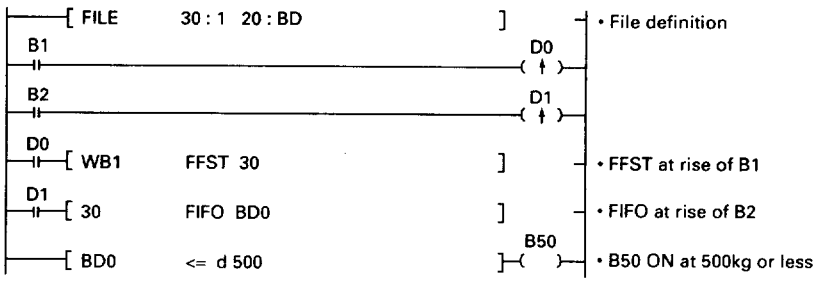
1. **Instruction**
   FILE, FFST, FIFO, <=

2. **Operation**
   Measure the weight of material at inlet of conveyor No. 1, and send material of 500 kg or less to conveyor No. 2 and material exceeding 500 kg to conveyor No. 3. Up to 20 pieces of material can be loaded at conveyor No. 1.

3. **System diagram**



4. **Program**



• Example of monitoring with loader LITE

```
P0001|        00000000
 0005|------[ BD0000      <=   d 00000500
 ADDRESS  BIN              DEC         HEX       BCD
  W030.   ▼  ▼  ▼  ▼
   W0000  0000011100000000  0000001792  00000700  00000700
   W0001  0000011100000000  0000001792  00000700  00000700
   W0002  0000011100000000  0000001792  00000700  00000700
   W0003  0000011100000000  0000001792  00000700  00000700
   W0004  0000011100000000  0000001792  00000700  00000700
   W0005  0000011100000000  0000001792  00000700  00000700
   W0006  0000011100000000  0000001792  00000700  00000700
   W0007  0000011100000000  0000001792  00000700  00000700
   W0008  0000011100000000  0000001792  00000700  00000700
   W0009  0000011100000000  0000001792  00000700  00000700
   W0010  0000001100000000  0000000768  00000300  00000300
   W0011  0000001100000000  0000000768  00000300  00000300
```

```
B0050
 ]|[   ]|--|
```

Drive device (B50) ON at detection of 300kg

Example where 700kg and 300kg material flow through together in above diagram.

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

7. FILO load

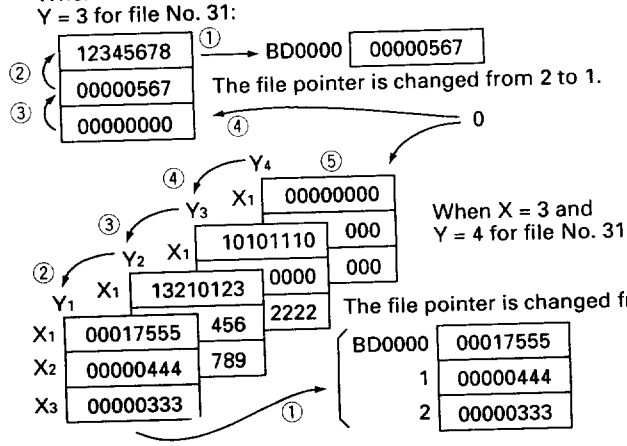
Instruction	FILO	F192
Symbol	⊢⊣[N FILO Wd] N: File No.	
Function	① Data having X-size is transferred from block Y_1 in file N to the area beginning with Wd and the file pointer is decremented by one. The data in the file is shifted by X and the resulting blanks are padded with zeros. ② If the FILO instruction is specified when the file pointer is 0, data is not transferred and the E flag is set ON. ③ N must be a file written by using the FFST instruction. This instruction can be used with the DSEL and WFIL instructions.	

Example

(1) Example

⊢⊣[31 FILO BD0000]

When X = 1 and Y = 3 for file No. 31:



(2) Conditions for setting flags
- ① When the file pointer becomes 0, the FE flag is set ON.
- ② If the FILO instruction is executed when the file pointer is 0, the E flag is set ON.
- ③ If the FILO instruction is executed when the FF flag is set ON, the FF flag is set OFF.
- ④ If Wd overflows, the zero flag is set ON.

(3) A program error occurs in the following cases.
- ① A value of 29 or less or a data table No. is specified as N.
- ② The length of the memory area beginning with address Wd is less than size X. (For example, WB0098 is specified as Wd when X = 3.)

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag						
																											S	Z	E	O	FE	FF
Wd	O	O	O	–	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	–	–	–	–	↕	↕	↕	↕

*When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

8. File clear (FLCL)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft. ware

Instruction	File clear	F194	Description	Example
Symbol	⊢⊣⊢[FLCL N] N: Data module No. or file No. Number 3 cannot be specified as N.		A user program error occurs in the following cases. ① An undefined user file is specified. ② The FLCL instruction is executed for a data table. ③ A module No. which does not exist is specified.	B000F ⊢⊣⊢[FLCL 30] File No. 30
Function	All contents of file N are cleared to zero. FIFO and FILO files are initialized to a state in which no data is stored. The file pointer is reset to 0.			

	0
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
11	0

Initialized to contain zeros.

Operand and influence flag	Influence flag					
	S	Z	E	O	FE	FF
	–	–	–	–	↑	↓

S: F004E E: A0041 FE: F0047
Z: F004F O: A0040 FF: F0046

9. File read (RFIL)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft ware

Instruction	File read	F197	Example
Symbol	⊢⊣⊢[RFIL N1 : Z1 : Z2 : N2 : Wd] N1: File No. (30 to 109) Z1: File address in X direction Z2: File address in Y direction N2: No. of words to be read Wd: First address of transfer destination		(1) Example B012B ⊢⊣⊢[RFIL 31 : d3 : d2 : 3 : BD0055]⊢
Function	① N2 words beginning with the address specified by X and Y in file N1 are transferred to the area beginning with Wd. The file pointer and file data are not changed. ② If Z1 and Z2 are specified as hexadecimal direct values, the data is handled as binary data. Otherwise, the data is handled as BCD data. ③ If specified Z1 or Z2 exceeds the file size, transfer is not executed and the E flag is set ON. ④ If the area beginning with the address specified by X and Y is less than N2 words, transfer is not executed and the E flag is set ON. ⑤ N2 words may be stored in multiple Y blocks.		X1 / X2 / X3 → Y1 X1 / X2 / X3 (01234567) → Y2 X1 (00000000) / X2 (00056789) / X3 → Y3 Three words beginning with the address specified by X3 and Y2 are transferred. BD0055 01234567 BD0056 00000000 BD0057 00056789

(2) Conditions for setting flags
① If values other than BCD data are indirectly specified for Z1 and Z2, the E flag is set ON.
② If Z1 exceeds X or Z2 exceeds Y of the file, the E flag is set ON.
③ If the length of the file area beginning with the address specified by X and Y is less than N2 words, the E flag is set ON.
④ If transfer data exceeds the capacity of Wd, the zero flag is set ON.

(3) A program error occurs in the following cases.
① A value other than a file No. (30 to 109) is specified as N1.
② Direct values area specified for Z1 and Z2 and the condition described in ② or ③ of above item (2) occurs.
③ The number of words after Wd is less than N2. (for example, WB509 is specified as Wd when N2 = 3.)

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag					
																											S	Z	E	O	FE	FF
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	↑	–	–
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	↑	–	–
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–						

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

10. File write (WFIL)

Processor (▒▒: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	File write	F198

Symbol

⊢⊣⊢[Ws WFIL N1 : Z1 : Z2 : N2]

Ws: First address of transfer origin
N1: File No. (30 to 109)
Z1: File address in X direction
Z2: File address in Y direction
N2: No. of words to be written

Function

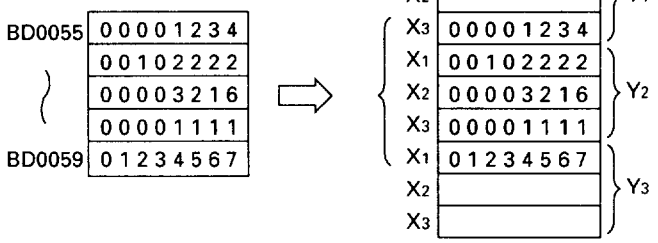
① N2 words beginning at address Ws are transferred to the area beginning at the address specified by X and Y in file N1. The file pointer is not changed.
② If Z1 and Z2 are specified as hexadecimal direct values, the data is handled as binary data. Otherwise, the data is handled as BCD data.
③ If the specified Z1 and Z2 exceed the file size, transfer is not executed and the E flag is set ON.
④ If the length of the area beginning at the address specified by X and Y is less than N2 words, transfer is not executed and the E flag is set ON.
⑤ N2 words can be stored in multiple Y blocks.

Example

(1) Example

B012B
⊢⊣⊢[BD0055 WFIL 30 : d3 : d1 : 5]⊢

Five words are transferred to the area beginning at the address specified by X3 and Y1.



(2) Conditions for setting flags
① If values other than BCD data are indirectly specified for Z1 and Z2, the E flag is set ON.
② If Z1 or Z2 exceeds the X or Y size of the file, the E flag is set ON.
③ If the length of the file area beginning at the address specified by X and Y is less than N2 words, the E flag is set ON.

(3) A program error occurs in the following cases.
① A value other than a file No. (30 to 109) is specified for N1.
② Direct values are specified for Z1 and Z2 and the condition described in ② or ③ of above item (2) occurs.
③ The number of words after Ws is less than N2. (For example, WB509 is specified as Ws when N2 = 3.)
④ A data table is specified for N1.

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag					
																											S	Z	E	O	FE	FF
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	–	–						
Z1	O	O	O	O	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	–	–	–
Z2	O	O	O	O	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	O	O						

* When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

11. File information (FINF)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	File information	F199	Description	Example
Symbol	⊢⊣[N FINF Wd] N: File No. (30 to 109) Wd: Storage address		A user program error occurs in the following cases. ① A value of 29 or less, 110 or more, an expansion module No. (30 to 109), or a data table is specified as N. ② An undefined user file is specified. ③ A module No. that does not exist is specified.	B012B ⊢⊣[30 FINF BD0020]⊢
Function	① The current file pointer of file N is stored in Wd. ② If N specifies a data table, the pointer held during file full status is stored in Wd.			n 2 1 → File pointer = 2 Areas until Y = 2 are used.

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag					
Wd	O*	O	O	–	O	O	O	O	O	O	O	O	O**	O	O	O	O	O	O	O	O	O	O	O	–	–	S	Z	E	O	FE	FF
																											–	–	↕	↕	–	–

* The input address of WB cannot be specified for Wd (operation result storage address).
** When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

ONE-POINT ADVICE **Data movement (Part 1) (FILO)**

FINF instruction and data movement (FILO)

1. Example of program

```
|
|----[ PROG  000    0000     0000                    ]
|B000B
|--||--[ 030  FINF  WM0015                          ]
|B0000                                              D0000
|--||-------------------------------------------( ↑ )--|
|B0002                                              D0002
|--||-------------------------------------------( ↑ )--|
|D0000
|--||--[ WB0001     FFST  030                       ]
|D0002
|--||--[ 030  FILO  BD0005                          ]
|
|----[ FILE  030 :  0003 :  0003 :  BD              ]
|
|A----[ PEND                                        ]
|
```

- The file pointer is transferred to WM15.
- Three-word data (WB1 to WB3) is written to file 30.
- The latest data of file 30 is transferred to BD5 to BD7 (three words).
- File definition

2. Storing data in file 30 by using FFST

Address W30.	Before storing data in file 30 BCD	First time BCD	Second time BCD	Third time BCD	Fourth or subsequent time (Reference) BCD
0000	00000000	00001811	00002422	00001844	00003333
0001	00000000	00004020	00002240	00004482	00002222
0002	00000000	00002411	00004822	00003444	00001111
0003	00000000	00000000	00001811	00002422	00001844
0004	00000000	00000000	00004020	00002240	00004482
0005	00000000	00000000	00002411	00004822	00003444
0006	00000000	00000000	00000000	00001811	00002422
0007	00000000	00000000	00000000	00004020	00002240
0008	00000000	00000000	00000000	00002411	00004822
0009	.....	.....	.....	.....	.....
M015	00000000	00000001	00000002	00000003	00000003
	Pointer number "0"	Pointer number "1"	Pointer number "2"	Pointer number "3"	The pointer number is not changed.

Y1 = W30.0000 to W30.0002; Y2 = W30.0003 to W30.0005; Y3 = W30.0006 to W30.0008. No data. Data is rewritten.

3. Reading data from file 30 by using FILO

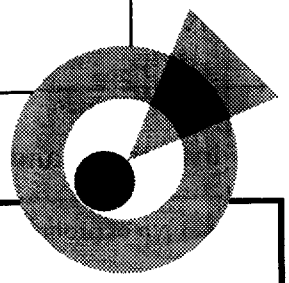
Address W30.	First reading BCD	Second time BCD
0000	00002422	00001811
0001	00002240	00004020
0002	00004822	00002411
0003	00001811	00000000
0004	00004020	00000000
0005	00002411	00000000
0006	00000000	00000000
0007	00000000	00000000
0008	00000000	00000000
0009	.....	.....
M015	00000002	00000001
	Pointer number "2"	Pointer number "1"

Y1 = W30.0000 to W30.0002; Y2 = W30.0003 to W30.0005; Y3 = W30.0006 to W30.0008.

Data movement after transfer by using FILO

Address	BCD	BCD
BD0005	00001844	00002422
BD0006	00004482	00002240
BD0007	00003444	00004822
BD0008	00000000	00000000
BD0009	00000000	00000000
BD0010	00000000	00000000
BD0011	00000000	00000000
	Y1 data	Y1 data

ONE-POINT ADVICE Data movement (Part 2) (FIFO)



FINF instruction and data movement (FIFO)

1. Example of program

```
|
|----[ PROG 000   0000    0000                    ]
|B000B
|----||--[ 030  FINF  WM0015                      ]   • The file pointer is transferred to WM15.
|B0000                                            D0000
|----||-------------------------------------------( ↑ )--|
|B0001                                            D0001
|----||-------------------------------------------( ↑ )--|
|D0000
|----||--[ WB0001    FFST  030                    ]   • Three-word data (WB1 to WB3) is written
|D0001                                                to file 30.
|----||--[ 030  FIFO  BD0000                      ]   • The oldest data of file 30 is transferred to
|                                                     BD0 to BD3 (three words).
|----[ FILE  030 :  0003 :  0003 :  BD            ]   • File definition
|
|----[ PEND                                       ]
|
```

2. Storing data in file 30 by using FFST

Before storing data in file 30		First time		Second time		Third time		Fourth or subsequent time (Reference)	
Address W30.	BCD	BCD		BCD		BCD		BCD	Data is rewritten.
0000	00000000	00001811	Y1	00002422	Y1	00001844	Y1	00003333	Y1
0001	00000000	00004020		00002240		00004482		00002222	
0002	00000000 ← No data	00002411		00004822		00003444		00001111	
0003	00000000	00000000		00001811	Y2	00002422	Y2	00001844	Y2
0004	00000000	00000000		00004020		00002240		00004482	
0005	00000000	00000000		00002411		00004822		00003444	
0006	00000000	00000000		00000000		00001811	Y3	00002422	Y3
0007	00000000	00000000		00000000		00004020		00002240	
0008	00000000	00000000		00000000		00002411		00004822	
0009	.....	.....		.....		.....		.....	
M015	00000000 ← Pointer number "0"	00000001 ← Pointer number "1"		00000002 ← Pointer number "2"		00000003 ← Pointer number "3"		00000003 ← The pointer number is not changed.	

3. Reading data from file 30 by using FIFO

First reading		Second time			Address	BCD	BCD	
Address W30.	BCD	BCD	Data movement after transfer by using FIFO		BD0000	00001811	00002422	
0000	00001844	00001844			BD0001	00004020	00002240	
0001	00004482 Y1	00004482	← Threre is one remaining Y-size block.		BD0002	00002411 Y3 data	00004822 Y2 data	
0002	00003444	00003444			BD0003	00000000	00000000	
0003	00002422	00002422			BD0004	00000000	00000000	
0004	00002240 Y2	00002240			BD0005	00000000	00000000	
0005	00004822	00004822			BD0006	0000000	00000000	
0006	00001811	00001811						
0007	00004020 Y3	00004020	Note that threre are 3 remaining items of data.					
0008	00002411	00002411						
0009	.....	.....						
M015	00000002 ← Pointer number "2"	00000001 ← Pointer number "1"						

ONE-POINT ADVICE **Data movement (Part 3) (WFIL and RFIL)**

FINF instruction and data movement (WFIL, RFIL)

1. Example of program

```
|
|----[ PROG  000    0000     0000                              ]
|B000B
|--||--[  030  FINF  WH0015                                    ]
|B0000                                                        D0000
|--||-------------------------------------------------------( ↑ )--|
|D0000
|--||--[  WB0001     FFST  030                                 ]
|
|----[ FILE  030 :  0003 :  0003 :  BD                         ]
|B0004                                                        D0003
|--||-------------------------------------------------------( ↑ )--|
|B0005                                                        D0004
|--||-------------------------------------------------------( ↑ )--|
|D0003
|--||--[  BD0005     WFIL  030 :  d 00000003 :  d 00000002:  0004  ]
|D0004
|--||--[ RFIL  030 :  d 00000002 :  d 00000003 :  0002 :  BD0050   ]
|B000E
|--||--[ FLCL  030                                              ]
|
|----[  d 00001234  MOV   BD0005                               ]
|    |                                                         |
|    [  d 00004567  MOV   BD0006                               ]
|    |                                                         |
|    [  d 00008901  MOV   BD0007                               ]
|    |                                                         |
|    [  d 00002345  MOV   BD0008                               ]
|
|----[ PEND                                                    ]
```

① BD5 to BD8 (four-word data) is written to file 30.

② Data is transferred from file 30 to BD50 to BD51 (two words).

Initial data of file 30

Address W030.	BCD	
0000	00001844	Y1
0001	00004482	
0002	00003444	
0003	00002422	Y2
0004	00002240	
0005	00004822	
0006	00001811	Y3
0007	00004020	
0008	00002411	
0009	.....	
M015	00000003 ← Pointer number "3"	

2. Data movement by using WFIL and RFIL after storing data in file 30 is shown on the overlaid reading screen.

① BD5 to BD8 (four word data) is written to file 30 by using the WFIL instruction.

	Address	BCD		
	BD0005	00001234	Data BD5 to BD8	
	BD0006	00004567		
	BD0007	00008901		
	BD0008	00002345		
W30.	W0000	00001844	Y1	
	W0001	00004482		
	W0002	00003444		
	W0003	00002422	Y2	
	W0004	00002240		
	W0005	00001234		Data BD5 to BD8 is stored at the specified position in file 30.
	W0006	00004567	Y3	
	W0007	00008901		
	W0008	00002345		
	W0009	.....		
	M015	00000003 ←	The pointer number is not changed.	

② Data is read from file 30 and transferred to BD50 to BD51 by using the RFIL instruction.

Address	BCD		
W030.			
0000	00001844	Y1	
0001	00004482		
0002	00003444		
0003	00002422	Y2	
0004	00002240		
0005	00004822		
0006	00001811	Y3	
0007	00004020		
0008	00002411		
0009	.....		
M015	00000003 ←	The pointer number is not changed.	
M016	00000000		
BD0050	00004020	The specified data is stored in file 30.	
BD0051	00002411		

12. Data table definition (TABL), data (DATA) and data end (DEND)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Data table definition	F201
	Data	F235
	Data end	F202

Symbol

```
TABLE N1 : N2 : N3 : X
DATA      N
DEND
```

Function

① A user file is registered as a data table and data is set in the data table.
- N_1: File No.
- N_2: No. of words in X direction (1 for F30, 50, 50H, 60)
- N_3: No. of words in Y direction (Although the maximum value of N_2 x N_3 is 4096, this value is restricted by the program capacity.)
- X: Data format

② Data format

SI	0	Binary 16-bit data
DI	1	Binary 32-bit data
BD	2	BCD 8 digits
Loader D20 or LITE	Loader D05 or D10S	

Example

(1) Example

```
B012B
    [ TABL   31 : 1 : 4 : BD ]  ← Data table definition
W31.0 [ DATA  d1234 ]
W31.1 [ DATA  d5678 ]   } Data setting
W31.2 [ DATA  d32468 ]
W31.3 [ DATA  d1357 ]
    [ DEND ]  ← A program must end with a DEND instruction.
```

(2) ① Because the data table is to be used exclusively for reading, the program cannot write data to this table.
② The data table can be written anywhere in the program.
③ Other instructions (including the PAGE instruction) must not be inserted in the table definition.

(3) A program error occurs in the following cases.
① The specified file No. overlaps the file definition and expansion module.
② The specified number of words does not match the number of data.
③ The specification does not conform to the rules described in the column on the left.
④ The file No. which is out of range is specified.

(4) DI specification

[DATA h□□□□□]

Five or more numeric values must be written. (If h0 is to be specified, five or more zeros must be written.)

Operand and influence flag

N_1			N (Direct numerical value)		Influence flag					
Common										
0 to 26		30 to 109	d	h	S	Z	E	O	FE	FF
–		O	O	O	–	–	–	–	–	–

S: F004E E: A0041 FE: F0047
Z: A004F O: A0040 FF: F0046

Programming advice

Data set by the data table definition (TABL) is stored in the user program area in the same way as the user program. Thus, from the viewpoint of control, essential data such as direct data can be located in ROM to create a more reliable system.

3-5 Program control instruction

3-5-1 Program control

1. Program control instruction

The following table lists the instructions used to control program execution.

○: Available, —: Not available

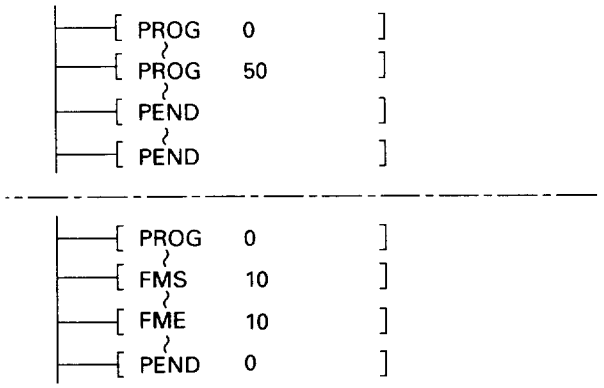
Program control instruction			Description	F30, F50 F50H	F60	F55, F70, F80H	F120H, F70S, F120S,F140S, F150S
Declaration of program	Program entry	(PROG)	The initiation of program and program type are declared.	—	—	○	○
	Program end	(PEND)	The end of program is declared.	○	○	○	○
	FM call	(FMC)	The use of subroutine programs (function modules) is declared, and the start and end of subroutine are specified.	—	—	—	○
	FM start	(FMS)					
	FM end	(FME)					
Program control	Skip	(SKIP)	All the instructions placed between the SKIP and SEND instruction are skipped and are not executed. These instructions are used to escape from LOOP.	—	—	○	○
	Skip end	(SEND)					
	Interrupt control	(DI)		—	—	○	○
	Interrupt enable	(EI)					
	Jump	(JMP)	Operation jumps from JMP to JEND instruction, and instructions placed between these instructions are not executed.	—	○	○	○
	Jump end	(JEND)					
	Loop	(LOOP)	A program routine between LOOP and CONT instruction is executed the specified number of times.	—	—	○	○
	Continuity	(CONT)					
Index register control	Push	(PUSH)	The contents of the index registers are saved before the execution of an interrupt program or subroutine program, and also restored after execution.	—	—	—	○
	Pop	(POP)					
	Load effective address	(LEA)	Indicates the address to be stored in the index register.				
	Index register addition	(IADD)	An execution address stored in the index register is modified by addition and subtraction. The result is stored again.				
	Index register subtraction	(ISUB)					
Page		(PAGE)	Indicates the start of page	○	○	○	○

2. Rules on program configuration

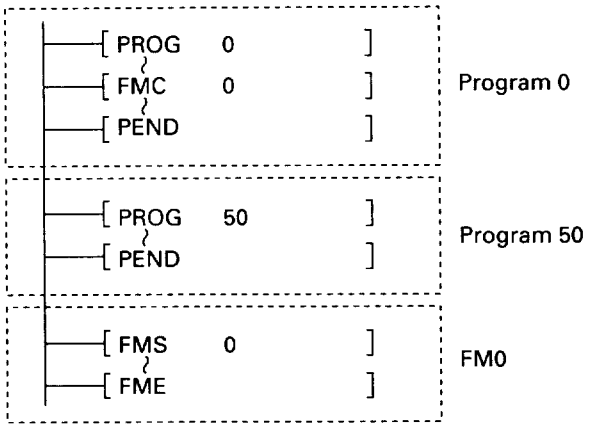
① User programs are classified into 3 types of program: P-level program (Cyclic program), Level 1 program (External interrupt program), and Level 2 program (Fixed-cycle interrupt program). *
P-level program is required for any PCs. F30, F50, F50H, and F60 series only use P-level programs.

② PROG and FMS instructions must not be inserted between PROG and PEND instructions.

Incorrect specification ✕

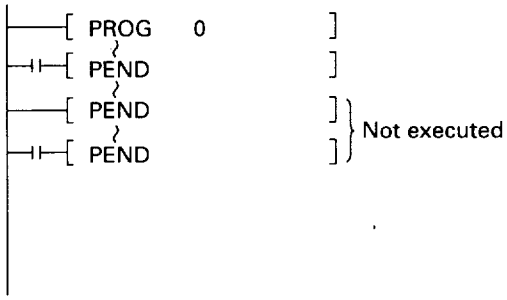


Correct specification ○

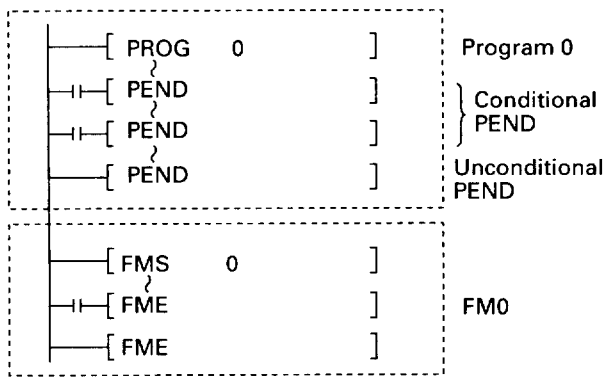


③ PEND and FME instructions can be specified many times for a specific program or function module, but the PEND and FME instructions specified in the middle of program must be conditional instructions and those specified at the end must be unconditional instructions.

Incorrect specification ✕



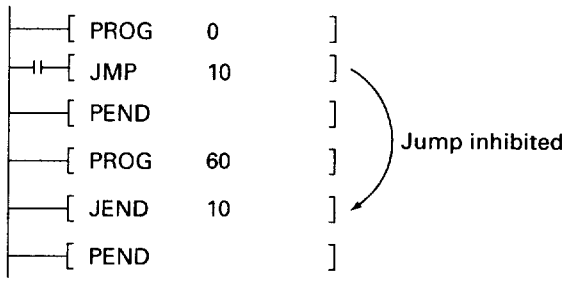
Correct specification ○



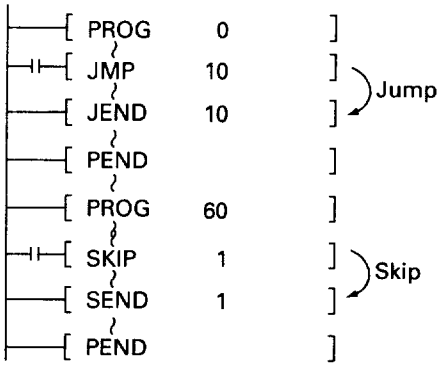
*: For details, see Subsection 2-4-2.

④ If a JMP or SKIP instruction is specified in a program, its termination (JEND or SEND) must not be specified in another program or function module.

Incorrect specification ✕

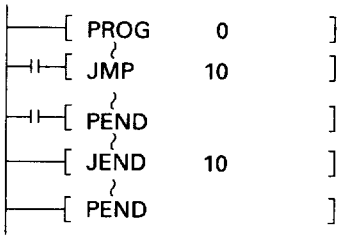


Correct specification ○



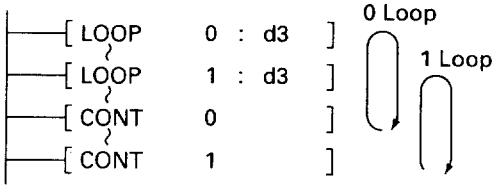
⑤ JEND and SEND can be specified after a conditional PEND instruction.

Correct specification ○

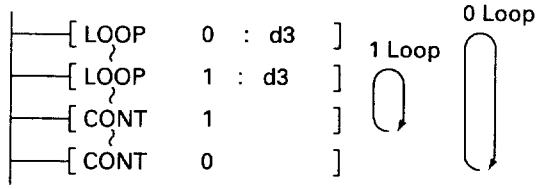


⑥ A routine specified by LOOP and CONT can be set within another LOOP and CONT routine.

Incorrect specification ✕

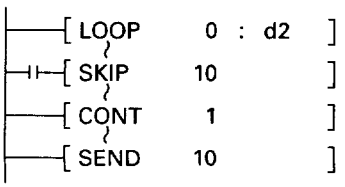


Correct specification ○

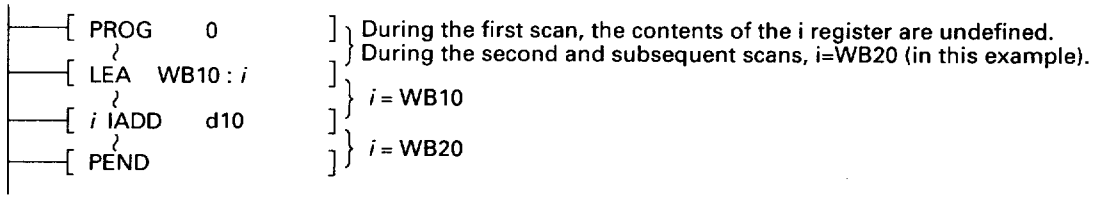


⑦ SKIP may be specified in the LOOP to CONT routine to escape from the loop.

Correct specification ○



⑧ Take note of the contents of index registers.



3-5-2 Program declaration

1. Program entry (PROG)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Program entry	F230

Symbol

⊣⊢[PROG N1 : N2 : N3]

N1: Program number
N2: Fixed-cycle time (10 to 4090ms)
N3: Start-delay time

Function

A program is defined and its start is declared.

N1
- 0: Cyclic operation program
- 50: Fixed-cycle interrupt program
- 60 to 67: External interrupt program
- 510: Block management declaration
- 511: Block definition declaration

N2 is valid only when N1=50.
(The time is set in units of 10ms.)

N3 is valid only N1=511.
(Block No. is specified.)

Note: When N1=510, or 511, N2 and N3 have the meanings on the right.

Example

(1) Example

```
[ PROG   0 : 404 : 0 ]
  ≀
[ PEND               ]
```

Time N2 must be set only when N1=50.
In other cases, N2 must be set to 0.
N3 must be set to 0.

(2) A program error occurs (ALM1 lamp lights) in the following cases.
① N2 is set to 9 or less for PROG50.
② The N1 specification does not conform to the specification rules.
③ An unconditional PEND instruction is not specified at the end of the program.

N1	N2	N3
510	0000: Start declaration of block management 1000: End declaration of block management	Not in use 0000 must be set.
511	0000: Start declaration of block definition 1000: End declaration of block definition	0000: Declaration of block No. 0 ≀ 0007: declaration of block No. 7

Note: For details on PROG510, and PROG511, see Subsection 2-58.

2. Program end (PEND)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Program end	F231

Symbol

Unconditional execution

⊢[PEND]

Conditional execution

⊣⊢[PEND]

Function

① The end of a program is declared.
② When processing reaches PEND in PROG 60 to 67, control is returned to the program that was being executed when the interrupt occurred.
③ When processing reaches PEND in PROG 50, T-link output processing is executed for the fixed-cycle group and control is returned to the program that was being executed when the interrupt occurred.
④ When processing reaches PEND in PROG 0, I/O processing is executed for the I/O device and control is returned to the beginning of the program.

Example

(1) Example

```
[ PROG   0 : 0 : 0 ]   ┐
  ≀                    │
⊣⊢[ PEND             ] │  Program "0"
  ≀                    │
[ PEND               ] ┘

[ PROG   50 : 30 : 0 ]   ┐
  ≀                      │  Program "50"
[ PEND               ]   ┘
```

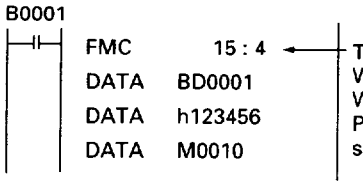
(2) ① An unconditional PEND instruction must always be specified at the end of each program. (For F70S, F120H and above series)
② PROG instruction is not required for F30, F50, F50H and F60.

Processor (▒▒: Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

3. FM call (FMC)

Instruction	FM call	F232
Symbol	Conditional execution FMC N1 : N2 DATA Z Unconditional execution FMC N1 : N2 DATA Z	
Function	① Function module N1 is called and parameters (DATA Z) are set. ② N1: Function module No. N2: No. of parameters Indicates the No. of DATA Z to be specified on the lines after the FMC instruction. • In decimal, the number of parameters is counted: 2 parameters per 1 data. * In hexadecimal, the number of parameters is: If 1 data is 4 digits..... 1 If 1 data is 8 digits..... 2 ③ An FM of the same FM No. can be called by programs of any level. ④ An FM can be called by another FM. (This is called "Nesting".) ⑤ If the No. of parameters is set 0, a program that does not use parameters for function modules is possible.	

Example



The No. of parameters (DATA No.) is 4.
WP1 = BD0001
WP2 = h123456
P4 = M0010
specification to set the above value.

(1) A program error occurs in the following cases.
① A number outside the specified range is specified. (FM No. error)
② A figure outside the specified range is specified. (FMC parameter error)
③ There is no FMS instruction with the same number as that of the specified FMC instruction. (No FMC module)
④ The No. of parameter specified by N2 does not match the No. of DATA Z. (FMC parameter error)

(2) The function module No. (N1) and the No. of parameters (N2) should be the values as below.

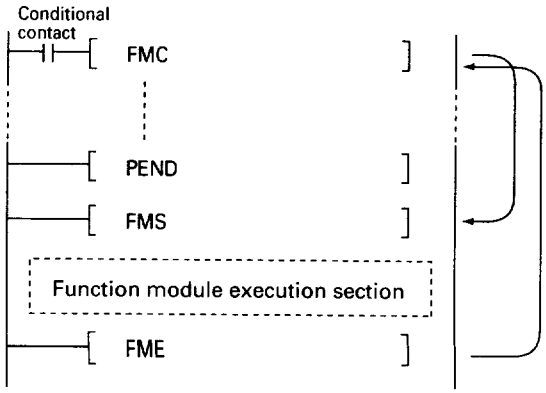
	F120H, F70S, F120S	F140S, F150S
N1	0 to 63	0 to 255
N2	0 to 31	0 to 31

Operand and influence flag

Bit	B	M	K	D	F	A	S	T	C	i	j	k	l	m	P	Q											Influence flag			
Z	O	O	O	O	O	O	—	O	O	O	O	O	O	O	O	O											S	Z	E	O
Wd	WB	WM	WK	WF	WA	WS	W9	TS	TR	CS	CR	BD	WL*	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	–	–	–	–
Z	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O				

* When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Function module operation



When the conditional contact of the function module call (FMC) is set to ON, memory and direct data specified by the DATA instruction are transferred as parameters to the function module execution section. When execution of the function module execution section is complete, the next instruction of the FMC is executed.

4. FM start (FMS)

Processor (▨: Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	FM start	F233	Description	Example
Symbol	⊢[FMS N1 : N2] N1: Function module No. (See FMC.) N2: Range of work area (0 to 31)		(1) Example ⊢[PEND] ⊢[FMS 15 : 5] Program ⊢[FME] The part of FMS to FME must be placed after PEND instruction. Work area range in this example: Word data: 6 words of WQ0 to WQ5 Relay: 32 bits of Q0 to Q1F (Overlapped with word data WQ0) (2) Work area range ① Relay and coil: 32 points of Q0 to Q1F ② Word data: 32 words of WQ0 to WQ31 (one word = 32 bits) The WQ area overlaps Q0 to Q1F and can be accessed in units of words or in units of bits. (3) A program error occurs in the following cases. ① A number outside the specified range is specified as N1. (FM No. error) ② A number outside the specified range is specified as N2. (Execution disabled) ③ There is no unconditional FME instruction at the end of FM. (No unconditional FME)	
Function	(1) The start of function module N1 is declared and work area N2 is specified. (2) Work area specification • A work area is a register that functions as an operand only during FM execution. It can be used to store word data or as a relay within the specified FM range. • When a work area is to be used only for a relay (bit specification), N2 must be 0 (32 bits of Q0 to Q1F). • When a work area is to be used only for word data (word specification), N2 must be a number (0 to 31). Example: If 10 is specified as N2, 11 words of WQ0 to WQ10 can be used. (One word = 32 bits)			

5. FM end (FME)

Processor (▨: Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	FM end	F234	Description	Example
Symbol	Unconditional execution: ⊢[FME] Conditional execution: ⊢⊣[FME]		① Multiple FME instructions can be specified for a single FMS instruction. Note that only one unconditional FME instruction can be specified. ② A function module must end with an unconditional FME instruction. ③ The FME instruction must be placed after the PEND instruction.	⊢[PEND] ⊢[FMS 15 : 2] Circuit B0000 ⊢⊣[FME] Circuit ⊢[FME] This program is executed when B0 is OFF.
Function	① The end of a function module is declared and control is passed to the instruction following the FMC instruction. ② The contents of parameters set by the FMC instruction are deleted after the FME instruction is executed. ③ The contents of the work area specified by the FMS instruction are deleted after the FME instruction is executed.			

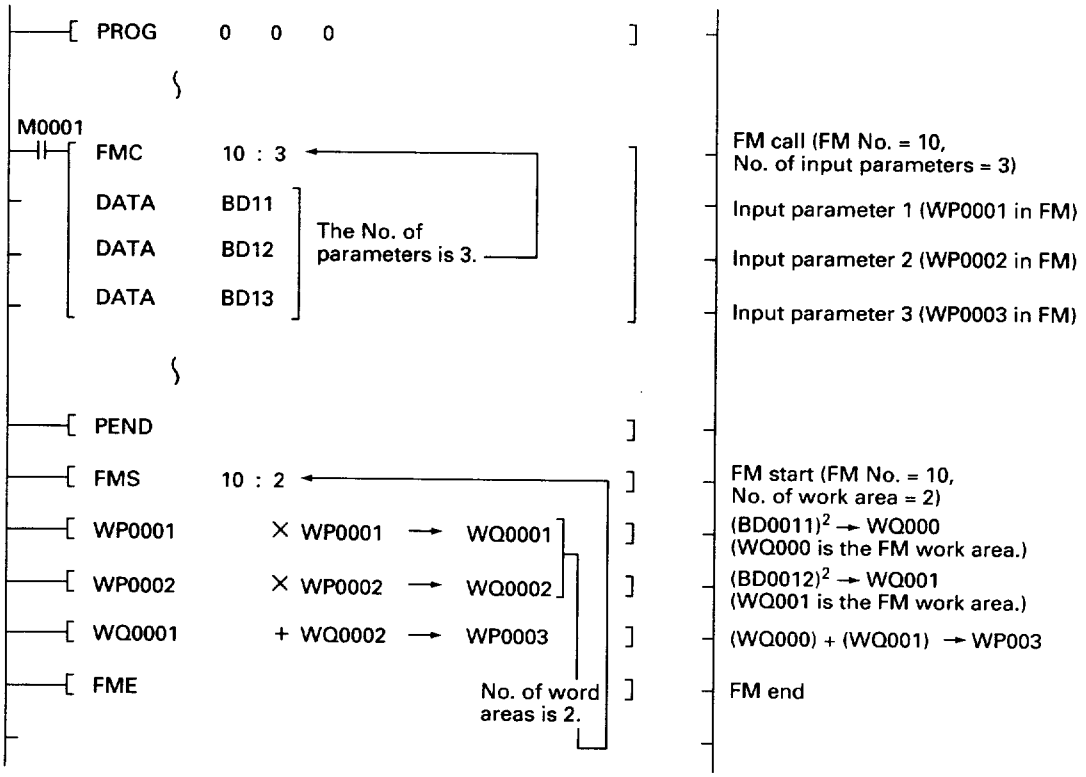
ONE-POINT ADVICE **Example of FM application**

Advantages of using function modules

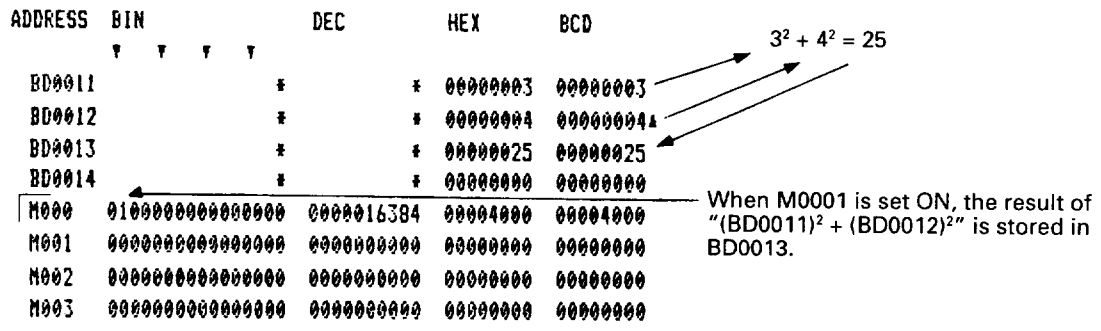
1. In programs that repeat the execution of similar processing operations, the number of program steps can be reduced by creating function modules.
2. When part of a program for which processing can be standardized is created as a function module, that part of the program can be reused. 64/256 types of programs can be registered as function modules.

An example of using function modules is shown below.

$(BD0011)^2 + (BD0012)^2 \longrightarrow BD0013$



- Example of monitoring by program loader LITE

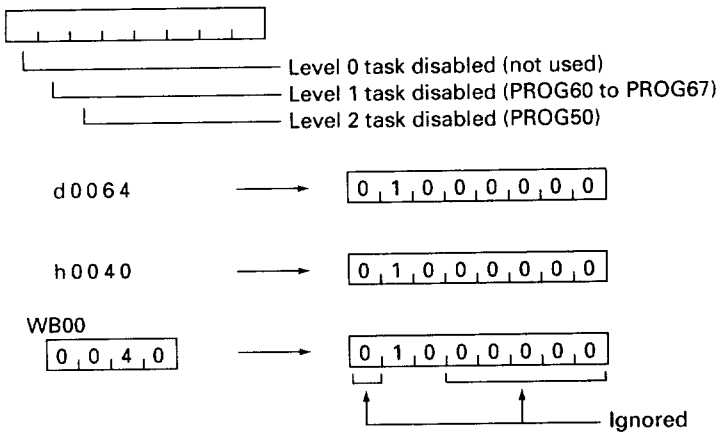


3-5-3 Program control

1. Interrupt control (DI)

Processor (▭ : Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Interrupt control	F316
Symbol	Unconditional execution ⊢[DI Z] Conditional execution ⊢⊢[DI Z]	
Function	① Interrupts by the task assigned the interrupt level indicated by Z are disabled. ② Direct decimal values are once converted into binary values. For the binary values, direct hexadecimal values, and indirect specifications, the low-order eight bits are used as disabled levels.	

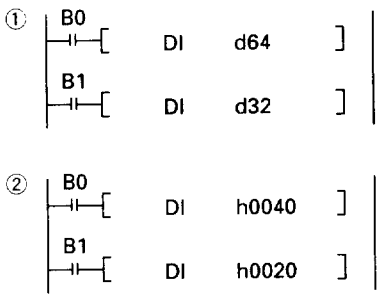


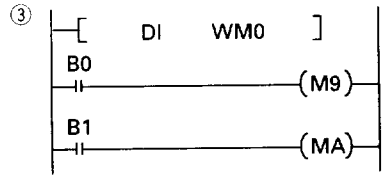
Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z	O	O	O	O	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
																											–	–	↑	–

* When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Progam example





For the programs ①, ②, and ③ commonly:
When B0 is ON, Level 1 interrupt is disabled.
When B1 is ON, Level 2 interrupt is disabled.

Level 1 = External interrupt program (PROG60 to 67)
Level 2 = Fixed-cycle interrupt progam (PROG 50)

Notes on programming

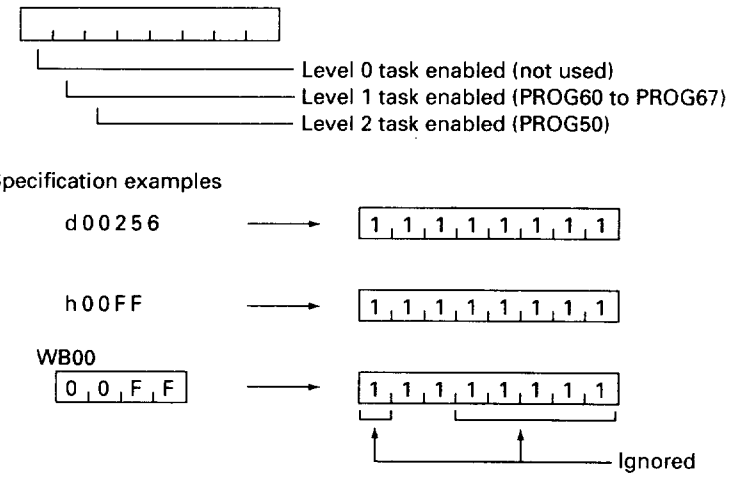
When the number of the interrupt waitings exceeds 32 while an intterrupt by Level 2 (Fixed-cycle interrupt) program is prohibited, F002E (Program sow-down flag) is set ON, and a nonfatal fault occurs.

The flag is reset when an enable flag is ON by the interrupt enable (DE) instruction. The interrupt disable (DI) instruction must be used together with the interrupt enable (DE) instruction.

2. Interrupt enable (EI)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Interrupt enable	F317
Symbol	Unconditional execution ⊢—[EI Z] Conditional execution ⊢⊢[EI Z]	
Function	① Interrupts by the task assigned the interrupt level indicated by Z are enabled. ② For direct decimal values and direct hexadecimal values, the low-order eight bits after conversion into binary equivalents are used as enabled levels. For indirect specifications, the low-order eight bits in the data after conversion into binary equivalents are used as enabled levels.	



Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z	O	O	O	O	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
																											–	–	↑	–

*When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

Note on programming

When the interrupt disable (DI) instruction and the interrupt enable (EI instruction are set ON simultaneously, the interrupt enable (EI) instruction is valid regardless of those positions in the program.

3. Skip (SKIP) and skip end (SEND)

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Skip and skip end	F250, F251	Description	Example
Symbol	Unconditional execution: SKIP N SEND N Conditional execution: SKIP N SEND N N: Skip No. (0 to 99)		① Both conditional skip and unconditional skip are available. ② Skipping in the reverse direction can be specified by placing the SEND instruction before the SKIP instruction. ③ Skip source can be placed at more than one position. However, the same SEND number cannot be double-assigned.	B0020 LOOP 55 : d12 Program B0010 SKIP 12 Program CONT 55 Program SEND 12 The skip instruction is used to exit from a LOOP/CONT loop. When contact B0001 is set ON, control skips to the specifed circuit.
Function	① When the SKIP instruction is executed, control skips to the next instruction of the SEND instruction which has the same N value as the SKIP instruction and the skipped portion of the program is not executed.			

4. Jump (JMP) and jump end (JEND)

Processor (▭: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Jump and jump end	F253, F254	Description	Example
Symbol	Unconditional execution: JMP N JEND N Conditional execution: JMP N JEND N N: Jump end No. (0 to 99)		① Conditional jump and unconditional jumps can be specified. ② When the JEND instruction is placed before the conditional JMP instruction, jumping in the reverse direction can be specified. ③ Multiple jump origins can be specified, but the same JEND No. cannot be double-assigned.	B0000 JMP 1 B0001 C000 100 B0002 CT R C0000 B0010 JEND 1 JUMP1 B0000 B0001 C0000 No conting durng jumping
Function	① When the JMP instruction is executed, the program part between the JMP instruction and JEND instruction with the same N value is jumped (by passed) and that part is not executed. JMP 12 Program JEND 12			

5. Loop (LOOP)/continue (CONT)

Processor (▭: Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F60H	F120H	F70S	F120S	F140S	F160S	D05	D10S	D20	LITE	Soft-ware

Instruction	Loop and continue	F210, F211	Descripton	Example
Symbol	Unconditional execution ⊢[CONT Z] Conditional execution ⊢⊣[LOOP N : Z] N: Loop No. (0 to 99) Z: Repetition count		① Conditional and unconditonal loops can be specified	B001F ⊢⊣[LOOP 12 : BD0020] Program ⊢[CONT 12] When contact B001F is set ON, the program is executed according to the loop count stored in BD0020.
Function	① When the loop instruction is executed, the program part between the LOOP instruction and CONT instruction having the same N values as the LOOP instruction is repeatedly executed Z times. ② If Z is 0, control skips to the CONT instruction. ③ If contact (⊣⊢) is OFF, the instructions between LOOP and CONT instruction are not executed.			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
Z	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	S	Z	E	O
																											–	–	–	–

3-5-4 Index register control

1. Push (PUSH)

Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft. ware

Instruction	Push	F310	Descripton	Example
Symbol	Unconditional execution ⊢[PUSH R] Conditional execution ⊢⊣[PUSH R] R: Index register (5 types: *i, j, k, l,* and *m*)		① PUSH and POP instructions are used as a pair. ② R specification For example, *i* must be written in a word specification (W*i*) or bit specification (*i*). ③ If the number of PUSH does not match the number of POP in a scan, or the number of PUSH exceeds 64, a PUSH/POP error occurs (F0017 is set ON) and a fatal fault results.	B0000 ⊢⊣[FMS 15 : 2]⊢ FM start declaration ⊢[PUSH *i*]⊢ The contents of index register *i* are saved. ⊢[LEA BD0120 : *i*]⊢ The effective address of BD0120 is stored in index register *i*. Program ⊢[POP *i*]⊢ The contecnts of index register *i* are restored. ⊢[FME 15]⊢ FM end declaration The above program indicates the FM porcessing assuming that the main program executes processing by using index register *i*.
Function	① The contents of the R-specified index register are saved. ② The PUSH instruction can use up to 64 registers.			

2. Pop (POP)

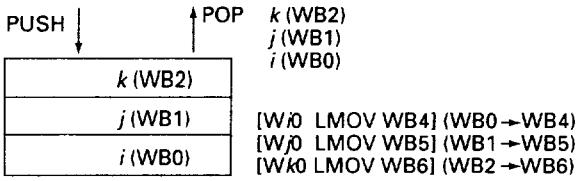
Processor (▨: Applicable)											Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft. ware

Instruction	Pop	F311	Descripton	Example
Symbol	Unconditional execution ⊢[POP R] Conditional execution ⊢⊣[POP R] R: Index register (5 types: *i, j, k, l,* and *m*)		① PUSH and POP instructions are used as a pair. ② If POP instruction is executed while PUSH instruction is not effective, a PUSH/POP error occurs (F0017 is set ON) and a fatal fault results. ③ If the index register which to be retored by POP instruction is *i, j,* or *k,* and the contents of the POP data is T, or C (bit data), an operation error (A0041) occurs. ④ If the index register which to be retored by POP instruction is *l, m,* and the contents of the POP data is other than T, or C (bit data), an operation error (A0041) occurs.	See the example column for the PUSH instruction.
Function	① The contents of the R-specified index register are restored.			

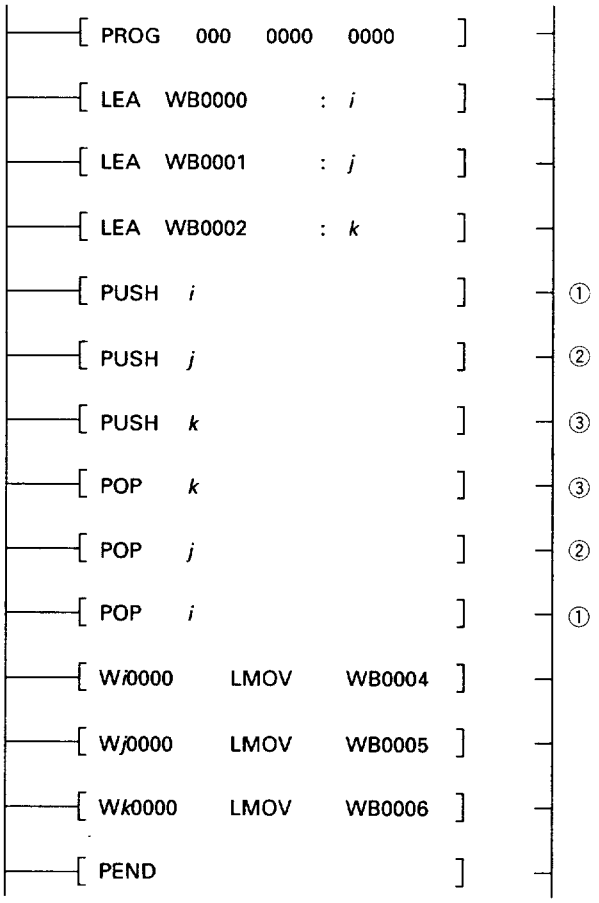
Operand and influence flag	Influence flag					
	S	Z	E	O		
	–	–	↕	–		

S: F004E O: A0040
Z: F004F E: A0041

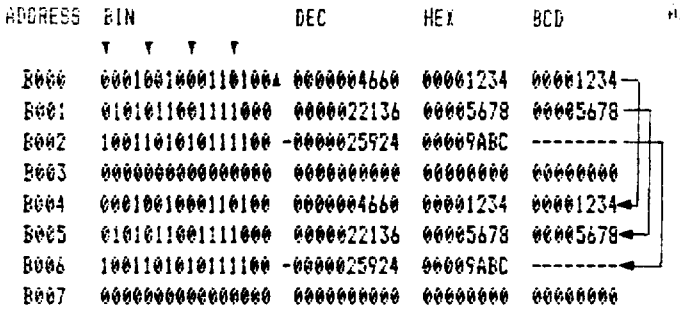
Specifying the PUSH and POP instructions



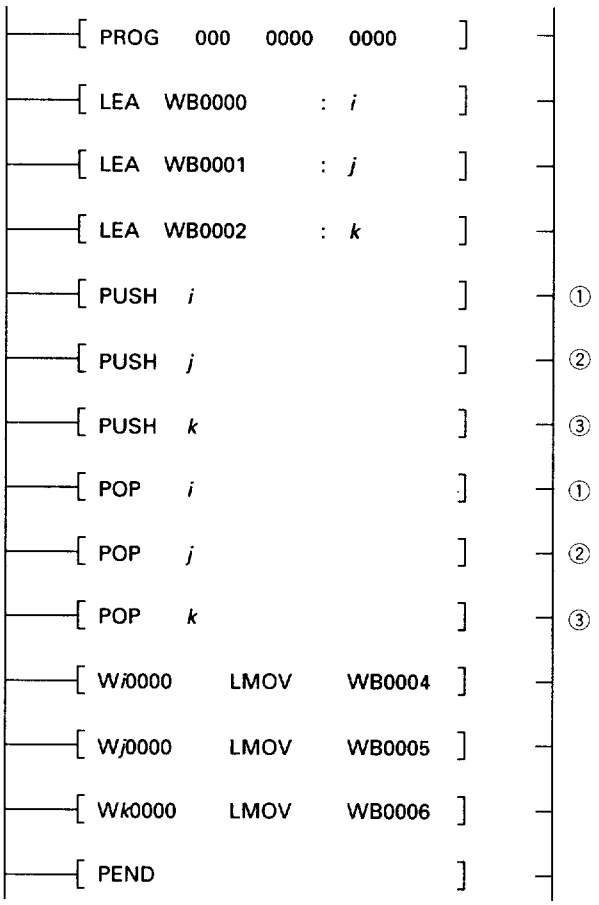
Correct:
When index registers i, j and k are specified in this order for the PUSH instruction, index registers k, j and i must be specified in that order for the POP instruction.



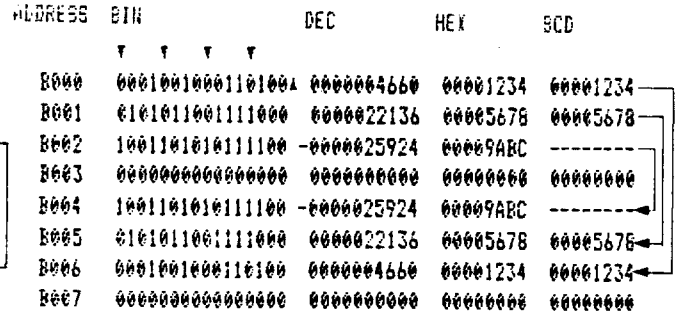
Monitoring by program loader LITE



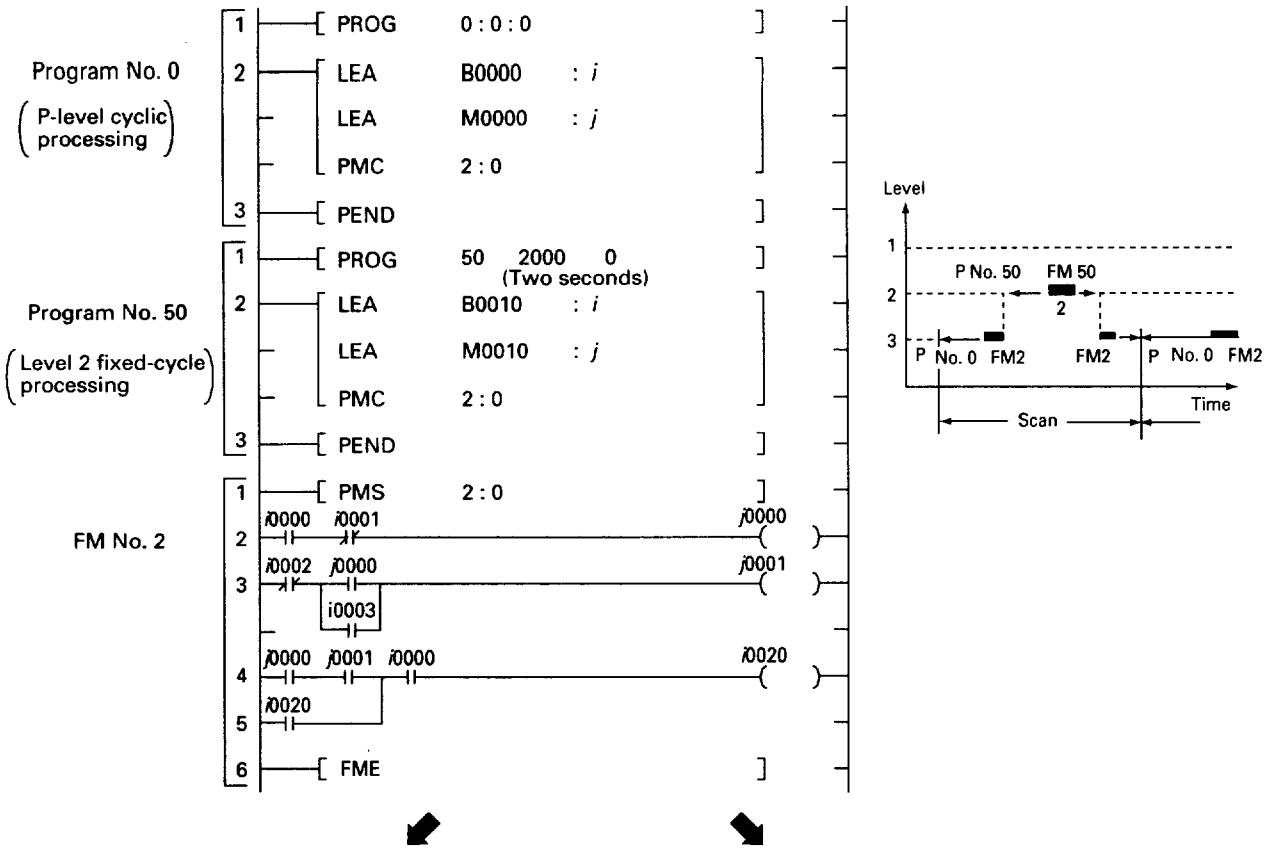
Incorrect:
If index registers i, j and k are specified in this order for the PUSH instruction and index registers i, j and k are specified in the same order for the POP instruction, the data is replaced.



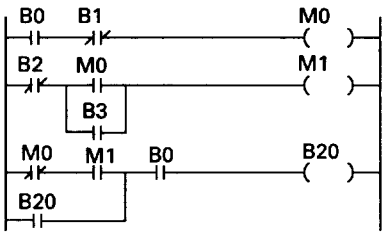
Monitoring by program loader LITE



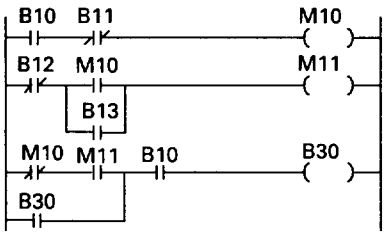
Example of using index registers and FM



Circuit operating with program No. 0



Circuit operating with program No. 50



① Data monitoring when B0 is ON

ADDRESS	BIN
B000	1000000000000000
B001	0000000000000000
B002	1000000000000000
B003	0000000000000000
B004	0000000000000000
B005	0000000000000000
B006	0000000000000000
B007	0000000000000000
M000	1100000000000000
M001	0000000000000000
M002	0000000000000000

② Data monitoring when B10 is ON

ADDRESS	BIN
B000	0000000000000000
B001	1000000000000000
B002	0000000000000000
B003	1000000000000000
B004	0000000000000000
B005	0000000000000000
B006	0000000000000000
B007	0000000000000000
M000	0000000000000000
M001	1100000000000000
M002	0000000000000000

③ Data monitoring when B0 and B10 are ON

ADDRESS	BIN
B000	1000000000000000
B001	1000000000000000
B002	1000000000000000
B003	1000000000000000
B004	0000000000000000
B005	0000000000000000
B006	0000000000000000
B007	0000000000000000
M000	1100000000000000
M001	1100000000000000

3. Load effective address (LEA)

Processor (▨ : Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Software

Instruction	Load effective address	F312	Example
Symbol	⊢⊣⊢[LEA Ws : R] R: Index register (5 types: *i, j, k, l,* and *m*) Ws: See the table below for operand.		① The object area varies depending on the type of index register. For details, see the table below. On the table below, Ws1 and Ws2 are; Ws1.....*i, j,* or *k* Ws2.....*l,* or *m* ② An idex register cannot be shared by bit addresses and word addresses. Example: ⊢[LEA B0 : *i*] *i* 0 — *i* 20 —( )— ⊢[W*i*0+BD0 → W*j*0] × An operation error occurs and the operation is not executed.
Function	① The effective address of the operand-specified are is stored at an index register.		

Operand and influence flag

Bit	B	M	K	D	F	A	S	T	C	i	j	k	l	m	P	Q	L									Influence flag			
Ws1	O	O	O	O	O	O	—	—	—	O	O	O	—	—	O	—	O									S	Z	E	O
Ws2	—	—	—	—	—	—	—	O	O	—	—	—	O	O	O	—	—									–	–	↑	–
Word	WB	WM	WK	WF	WA	WS	W9	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expansion	Wi	Wj	Wk	WP	WQ	d	h			
Ws	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	—	—	—			

Index registers

• Index registers

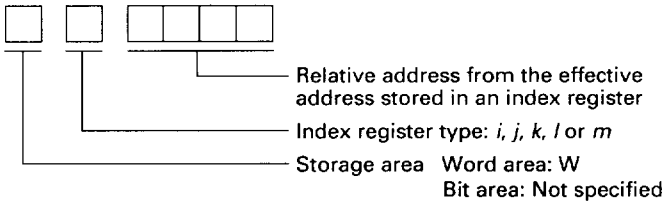
For the MICREX-F Series, a data address can be expressed as indentifier + serial number or as an effective address by using index register.

An effective address to be stored in an index register consists of an identifier and a serial number. The address can be processed by the addition and subtraction of index register contents.

Data never moves out of a data module area nor destroys other areas due to this address operation.

The five types of index registers have different storage areas.

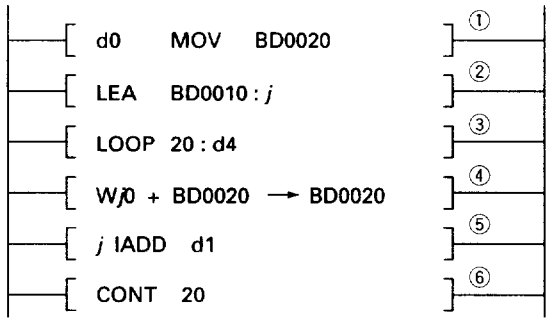
The expressions and types of index registers are as follows.



i, j, k	Word specification	0 to 4095 (decimal)
	Bit specification	0 to 511F (hexadecimal)
l, m		0 to 0999 (decimal)

• Index register operation

The operation and use of index registers are explained by using a program example in which index registers are used to add the value in area BD0010 to the value in area BD0013 and to store the result in BD0020.



Initial values in data areas

BD0010	200
11	20
12	4000
13	25
BD0020	Undefined

① "0" is stored in BD0020.

DB0020	0

② The effective address of BD0010 is stored in index register *j*.

j	BD0010

③ Processing between ③ and ⑤ is executed four times.

(A) First time The value in the area indicated by index register *j* is added to the value in BD0020 and the result is stored in BD0020.

j	BD0010	BD0020	200

(B) First time "1" is added to index register *j* and the result is stored in index register *j*.

j	BD0011	BD0020	200
(C) Second time *j*	BD0011	BD0020	220
(D) Second time *j*	BD0012	BD0020	220
⋮		⋮	
(E) Fourth time *j*	BD0013	BD0020	4245
(F) Fourth time *j*	BD0014	BD0020	4245

When the effective address stored in index register *j* is being updated, the values in BD0010 to BD0013 are added into one data and the result is stored in BD0020.

④ End of loop processing

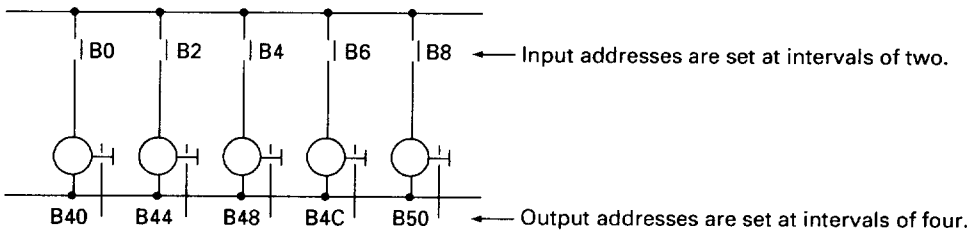
ONE-POINT ADVICE **Circuit using index registers**

Advantages of using index registers

1. An add index register can be conveniently used for the following sequence program.

(1) Input addresses have constant intervals.

(2) Output addresses have constant intervals.

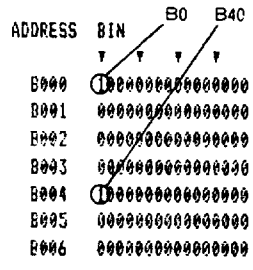


2. Programming the above sequence by using index registers.

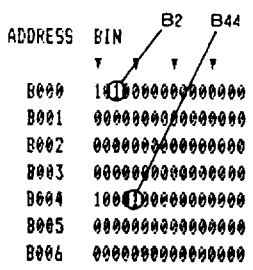
```
P0001
0001  [ PROG  000  0000  0000 ]
P0001
0002  [ LEA   B0000  : i ]          ┐ Address B0 is stored in index register
P0001                               │ i and address B40 is stored in index
0003  [ LEA   B0040  : j ]          ┘ register j.
P0001
0004  [ LOOP  001  : d 00000005 ]  ····· Execution is repeated five times in
P0001 i0000                    j0000    a loop No. 1.
0005  ─||──────────────────────( )──  ····· Sequence program to be repeated.
P0001
0006  [ i  IADD  d 00000002 ]       ┐ (2)10 is added to index register i and
P0001                               │ (4)10 is added to index register j.
0007  [ j  IADD  d 00000004 ]       ┘
P0001
0008  [ CONT 001 ]                  ····· End of loop No. 1, control is
P0001                                     returned to the instruction,
0009  [ PEND ]                           following the LOOP instruction.
```

3. Example of monitoring execution of the above program by program loader LITE

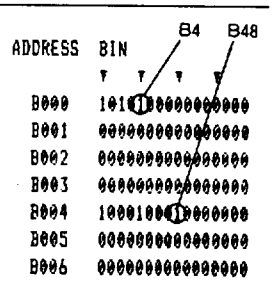
① When B0 is ON, B40 is ON.



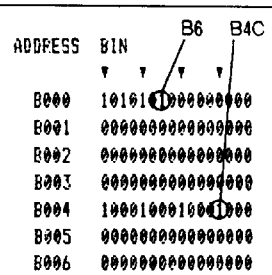
② When B2 is ON, B44 is ON.



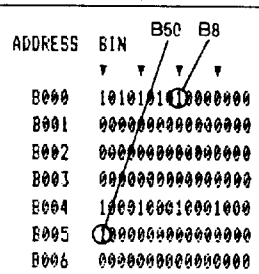
③ When B4 is ON, B48 is ON.



④ When B6 is ON, B4C is ON.



⑤ When B8 is ON, B50 is ON.



4. Add index register (IADD)

Processor (▨ : Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Add index register	F313	Descripton	Example
Symbol	⊢⊣[R IADD Z] R: Index register (5 types: *i*, *j*, *k*, *l*, and *m*)		① The object area varies depending on the type of index register. For details, see the LEA instruction on page 3-160.	B0000 ⊢⊣[LEA BD0100 : *i*] [LOOP 12 : d5] [BD0000 MOV Wi0] [*i* IADD d10] [CONT 12] The data in BD0000 is transferred to BD0100, BD0110, BD0120, BD0130, and BD0140.
Function	① The contents of Z are added to the contents of the R-specified index register and the result is stored in the index register. R + Z → R			

Operand and influence flag

	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O*	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	–

* When using P or PE-link, WL (including W21, W22, W23, W120, W121, W122, and W123) is also usable.

5. Subtract index register (ISUB)

Processor (▨ : Applicable)										Program loader				
F30	F50 F50H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft-ware

Instruction	Subtract index register	F314	Descripton	Example
Symbol	⊢⊣[R ISUB Z] R: Index register (5 types: *i*, *j*, *k*, *l*, and *m*)		① The object area varies depending on the type of index register. For details, see the LEA instruction on page 3-160.	B0000 ⊢⊣[LEA BD0100 : *i*] [LOOP 12 : d5] [BD0000 MOV Wi0] [*i* IADD d10] [CONT 12] The data in BD0000 is transferred to BD0100, BD0097, BD0094, BD0091, and BD0088.
Function	① The contents of Z are subtracted from the contents of the R-specified index register and the result is stored in the index register. R – Z → R			

Operand and influence flag

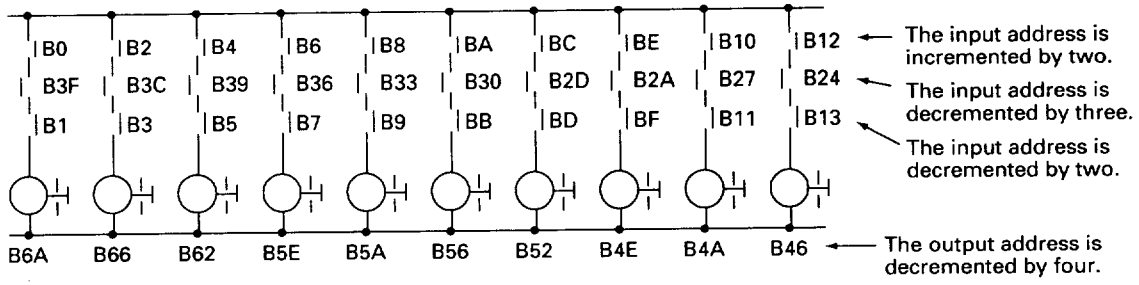
	WB	WM	WK	WF	WA	WS	W9.	TS	TR	CS	CR	BD	WL	W24	W25	W26	W30 to	W125	Expan-sion	Wi	Wj	Wk	WP	WQ	d	h	Influence flag			
																											S	Z	E	O
Z	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	–	–	↑	–

ONE-POINT ADVICE **Circuit using index registers**

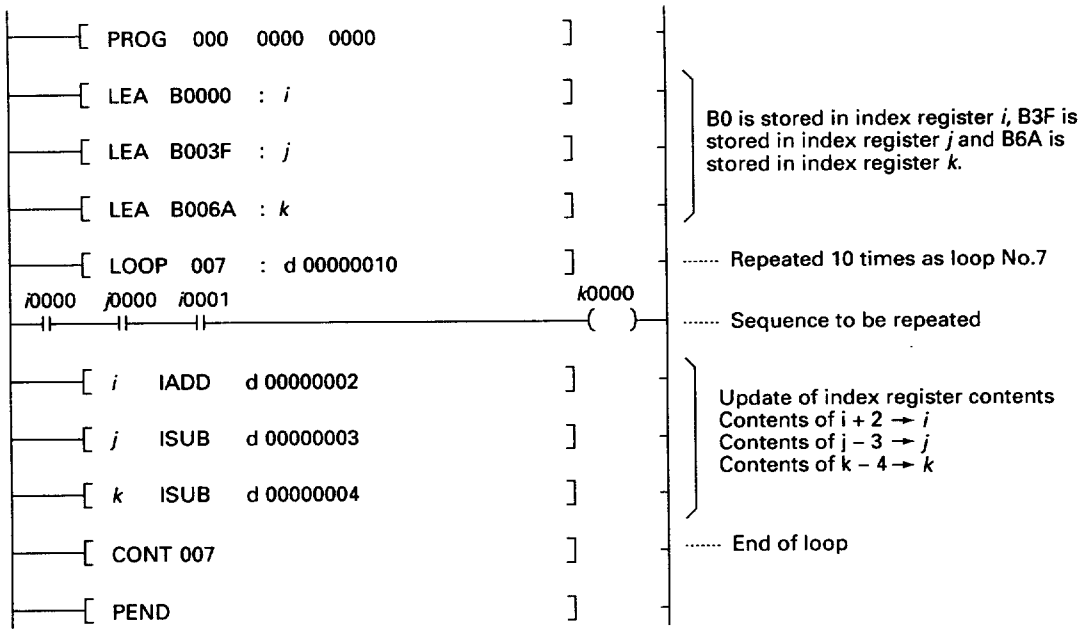
Advantages of using index registers

Example of combining add and subtract index registers

1. Sequence circuit



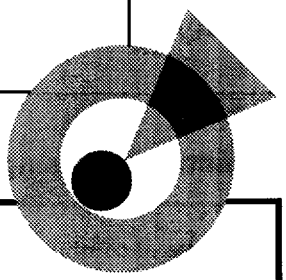
2. Programming the above sequence by using add and subtract index registers



3. Advantages of using index registers

1) Simplified programs
As shown in the above example, LOOP and CONT can be combined to repeat an operation any number of times.

2) Reduced number of program steps
Due to the advantage described above, the number of program steps can be reduced for larger programs (control).



Advatages of using index registers

Example of addition in units of words

1. Contents of addition

WB0 data (	80)	+ WM0 data (	40)	→	Stored in WK0 (	120)
WB1 data (	3000)	+ WM1 data (	67)	→	Stored in WK1 (	3067)
WB2 data (	2345)	+ WM2 data (	2345)	→	Stored in WK2 (	4690)
WB3 data (	2489)	+ WM3 data (	0)	→	Stored in WK3 (	2489)
	:		:		:	
	:		:		:	
WB19 data (	200)	+ WM19 data (	250)	→	Stored in WK19 (	450)

2. Example of program using index registers

```
[ PROG  0     0     0                    ]
[ LEA   WB0   : i                        ]
[ LEA   WM0   : j                        ]
[ LEA   WK0   : k                        ]
[ LOOP  2     : d 20                     ]
[ Wi0   +     Wj0   →   Wk0              ]
[ i     IADD  d 1                        ]
[ j     IADD  d 1                        ]
[ k     IADD  d 1                        ]
[ CONT  2                                ]
[ PEND                                   ]
```

Index register
WB0 is stored in index register *i*,
WM0 is stored in index registers *j*,
WK0 is stored in index register *k*.

..... Execution is repeated 20 times as loop No. 2.

..... Add instruction to be repeated. The monitor display is ineffective.

Index register
$(1)_{10}$ is added to index register *i*,
$(1)_{10}$ is added to index register *j*,
$(1)_{10}$ is added to index register *k*.

..... End of loop No. 2, control is returned to the instruction, following the LOOP instruction.

3. Monitoring by program loader LITE

```
ADDRESS  BIN              DEC        HEX       BCD
          ▼  ▼  ▼  ▼
  B000   0000000010000000  0000000128 00000080  00000080
  B001   0011000000000000  0000012288 00003000  00003000
  B002   0010000110100101  0000009029 00002345  00002345
  B003   0010010010001001  0000009353 00002489  00002489
  M000   0000000001000000  0000000064 00000040  00000040
  M001   0000000001100111  0000000103 00000067  00000067
  M002   0010000110100101  0000009029 00002345  00002345
  M003   0000000000000000  0000000000 00000000  00000000
  K000   0000000010010000  0000000288 00000120  00000120
  K001   0011000001100111  0000012391 00003067  00003067
  K002   0100001101001000  0000018064 00004690  00004690
  K003   0010010010001001  0000009353 00002489  00002489
  K004   0000000000000000  0000000000 00000000  00000000
  K005   0000000000000000  0000000000 00000000  00000000
```

80 + 40 = 120

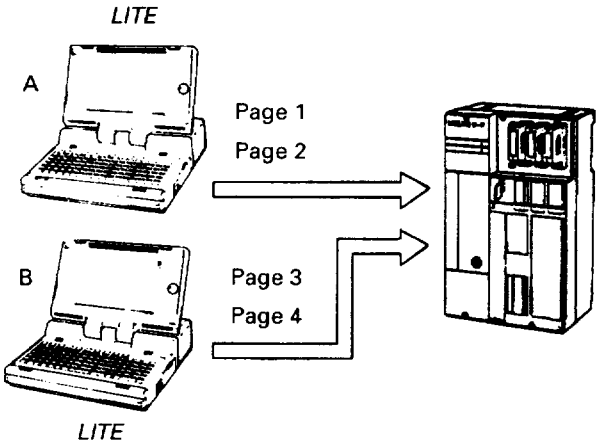
3000 + 67 = 3067

3-5-5 Page

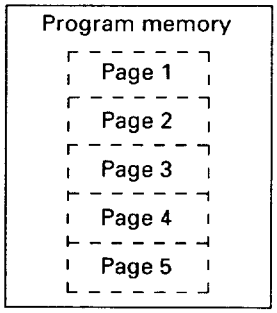
1. Page instruction

The page instruction can be effectively used to create, standardize or debug a program. An example is shown below.

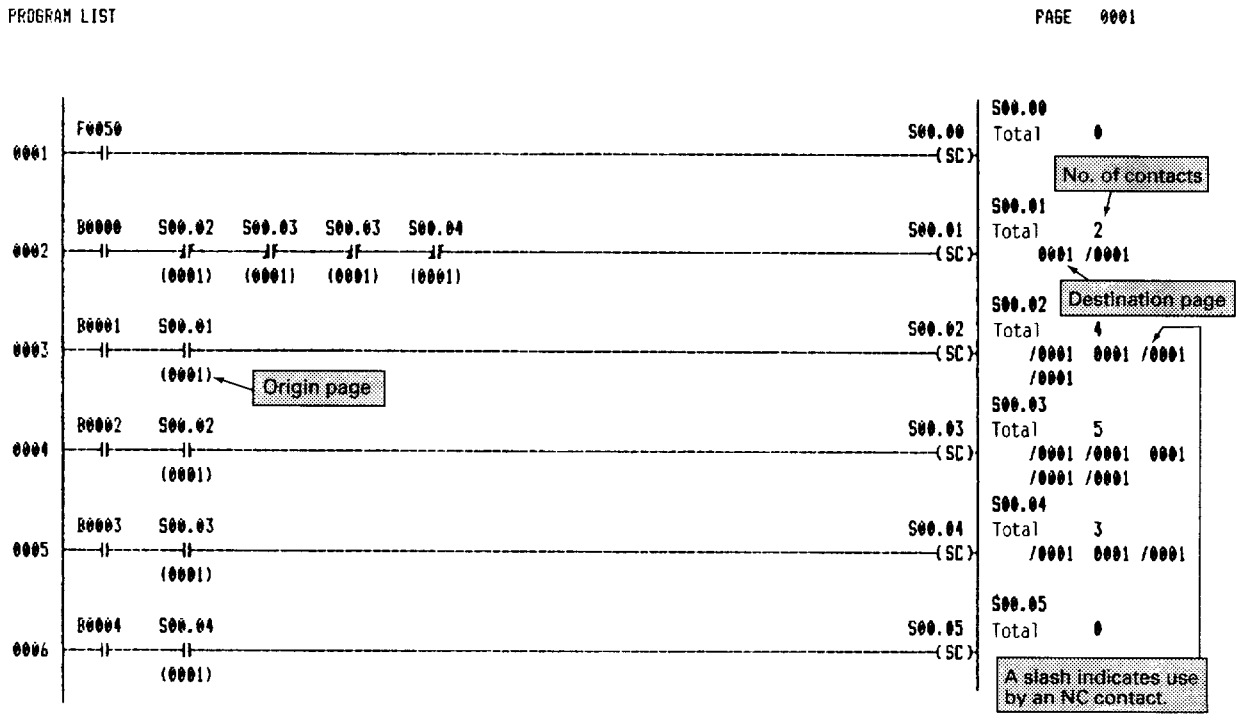
- A program containing many program steps can be created faster by using two program loaders to share the programming workload and to transfer the program to the specified processor. (In this case, page instructions must be written in the program before it is transferred.)



Program in the processor

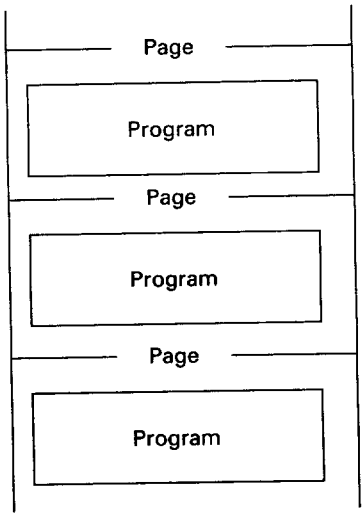


- Because the program list with cross-references can be printed with page numbers added to indicate the contact destinations and origins on the list, the program can be easily checked.



2. Page

Processor (▭ : Applicable)											Program loader				
F30	F60 F60H	F55	F60	F70	F80H	F120H	F70S	F120S	F140S	F150S	D05	D10S	D20	LITE	Soft ware

Instruction	Page	F325	Description	Example
Symbol	Unconditional execution: Page		① Even when the contents of memory are deleted, the first 1 page instruction is not deleted. ② When PAGE instruciton is executed, it has no effect on data memory. ③ Page numbers 1 to 9999 can be written. The second and subsequent pages must be programmed by using this instruction. ④ If the PAGE instruction is not written on a page, the page cannot be read. ⑤ The sequence of pages written in the program is not related to the execution sequence. Execution is done according to the order of programs. ⑥ Note that a user program error does not occur if the page number of PAGE instruction is double-assigned.	
Function	① The beginning of a page is declared.			

Section 4 System Registration and Expansion Function

Section 4 System Registration and Expansion Function

4-1 System definition registration

4-1-1 Registration items

To use the MICREX-F series in expansion mode, register the rules (system definitions) for each processor. See the explanation page in the table for details on the functions of registration items.

Classification	Registration item	Registration contents	Description	Reference page
System registration	Model code	Registering the model to be used	• Online programming does not require this registration. • The model code must be registered (changed) if offline programming is used or if a program written on a different model is transferred.	4-5
	Fail-soft operation	Selecting whether to set fail-soft operation	• The operation mode of the capsule or unit on the T-link is set. • Fail-soft operation set: Operation is continued except for an abnormal capsule. • Fail-soft operation not set: If one of the capsule in the system causes a fault, the operation of the entire system is stopped.	4-5 *
		Individual fail-soft (F55, F80H, F70S, F120S, F140S, and F150S only)	• Setting the operation mode of a specific capsule or unit • Individual fail-soft set: Even if a capsule or unit with individual fail-soft set causes a fault, operation is continued. • Individual fail-soft not set: Same as fail-soft operation of the entire system.	4-10 *
	Watchdog timer (WED) setting	Setting the WDT time	• The scan time (execution time) of the user program is monitored.	4-5
	Duplex processor (F70S, F120H, F120S, F140S, and F150S only)	Duplex set	• Set to duplex the processor.	4-6 *
		T-link station No.	• The main processor is distinguished from the standby processor.	
		Selecting mode of duplex	• Hot start (continuous) or cold start (initial) is selected. Only cold start is available for the F70S to F150S series.	
	Direct I/O registration (F55, F80H, F70S, F120S, F140S, and F150S only)	Setting direct access	• Set to execute I/O processing in scan asynchronization (direct access) mode. • Always set to use the external interrupt program.	4-7, 4-17
		Setting output hold (only in direct access mode)	• The operation mode of an output module is set. If the processor fails, output status prior to stop is held.	4-7
	Registering BD area size (F55, F80H, F70S, F120S, F140S, and F150S only)	BD area	• Registered to set the BD area of more than 256 words.	4-6

* Refer to the MICREX-F User's Manual "Communication" (FEH161).

Classification	Registration item	Registration contents	Description	Registration method
T-link registration	I/O expansion area registration	Registering I/O expansion area (F70S, F120S, F140S and F150S only)	• Registered to handle I/O data of more than 100 words in one T-link system.	4-6
	Configuration registration	Registering configuration	• Registered to limit that capsule to be operated on the T-link. (For example, at test operation)	4-9 *
		Setting output hold	• The operation mode of the output capsule is set. • If the processor fails, the preceding output status is held.	
		Registering T-link group (F55, F80H, F70, F70S, F120S, F140S, and F150S only)	• F55 to F150S series: Registered in group 0 to execute I/O processing in synchronization with the fixed-cycle interrupt program. • F120S to F150S series; Registered in group 1 to allocate the I/O expansion area.	
P-link/PE-link registration	Configuration registration (F70S, F120S, F140S, and F150S only)	Registering configuration	• Always registered to use the P-link/PE-link. • All station numbers of the processor on the P-link/PE-link are registered.	4-11 *
		Setting the local station number	• Set so that each processor station number is unique.	
		Registering P-link/ PE-link memory size	• The address and size are set for the transmission area of the local station.	
Message module registration	Use condition setting	Registering the data module number	• The data module number (file number) to be transmitted and received by message communication is set.	
		Registering the use of the message module	• The use of the message module for transmission and reception is specified.	4-14 *
		Selecting the link to be used	• The link (T-link or P-link) to be used for message communication is selected.	
		Setting station number	• The station number of the remote station is set.	
ME-NET registration	Registering configuration (F120S, F140S, and F150S only)		Registered to use the ME-NET card (optional card).	**

* Refer to the MICREX-F User's Manual "Communication" (FEH161).
** Refer to the ME-NET Operation Manual.

4-1-2 Timings at which the processor recognizes system definition

Classification	Item		Series						
			F30	F50, F50H	F60	F55, F70, F80H	F120H	F70S, F120S to F150S	Remarks
System registration	Model code		A	A	A	A	A	A	*1
	Fail-soft operation		A	A	A	A	A	A	
	WDT		A	A	A	A	A	A	
	Duplex processor	Duplex set					A	A	
		T-link					A	A	
	Direct I/O registration	Direct access				B	B	A	
		Hold				A	A	A	
	BD area expansion					A	A	A	
	Option registration						B	B	
T-link registration	Individual fail-soft		A	A	A	A	A	A	
	Configuration registration		A	A	A	B	B	A	
	Output hold		A	A	A	A	A	A	
	Group 0 registration					A	A	A	
	Group 1 registration							B	
	Expansion module No.							A	
	I/O area expansion							B	
P-link/PE-link registration	Configuration registration						B	B	
	Local station number setting						B	B	
	Broadcast transmission area registration						B	B	
	Expansion module No.							A	
Message module registration			A	A	A	A	A	A	

A: When storage by the loader is completed
B: When the power to the processor is turned off, and then turned on again (at reset)

*1: If registration is incorrect, turn on the power again, stop the processor, and then start it again.
The processor will perform the correct registration.

4-1-3 How to register system definition

When the normal system is used, system definition is not necessary. When the expansion system is used, system definition is necessary for the P-link, direct access, and protection against system failure. The steps for registering system definition with program loader LITE are explained below.

1. Basic steps for system definition

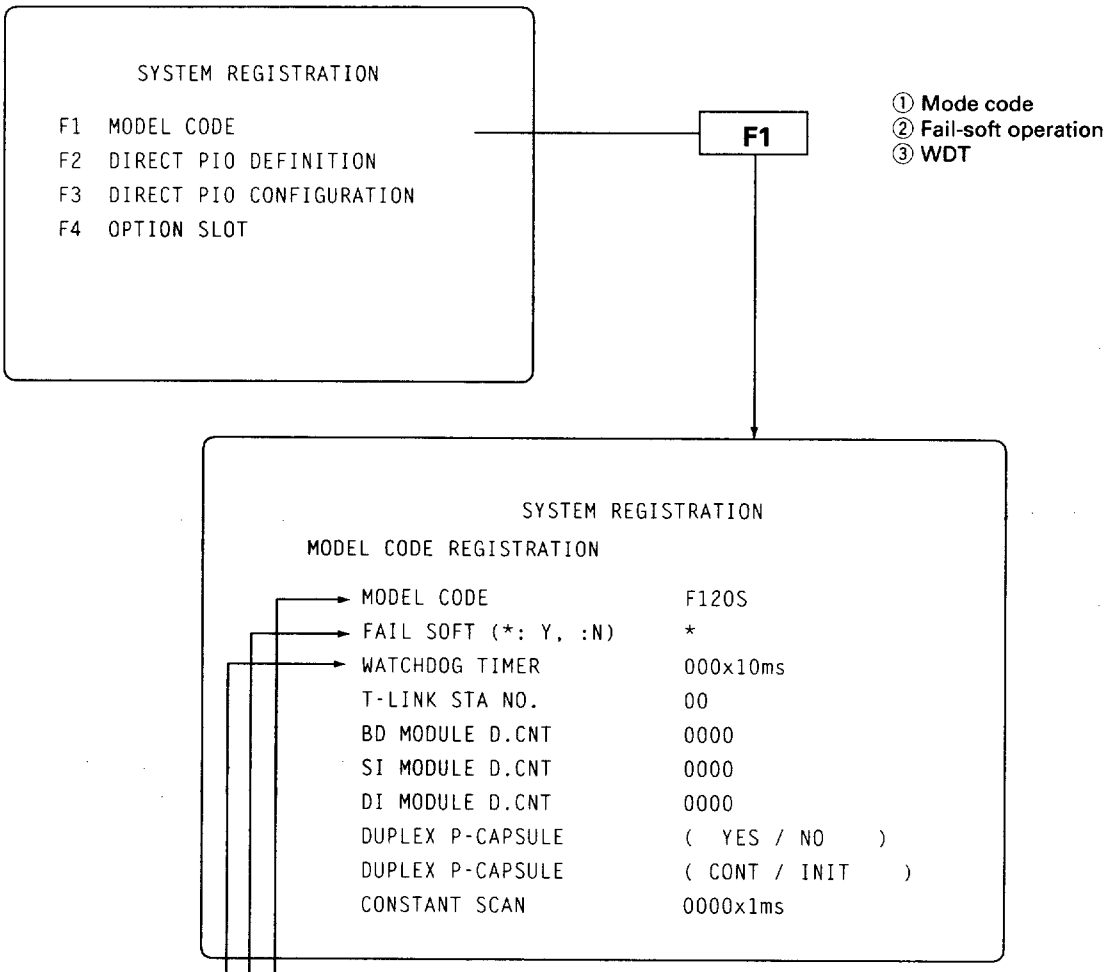
	Key operation	Display	Description
①	F5 (AUXILIARY)	F1 PROGRAMMING F2 MONITOR F3 TRANSFER/VERIFY F4 DOCUMENT F5 AUXILIARY	• LITE initial screen • D25 is the name of the software for MICREX-F.
②	F1 (SYSTEM DEFINITION)	**AUXILIARY** (1/3) F1 DEFINING SYSTEM F6 CLEARING MEMORY F2 DIAGNOSTICS F7 DEFINING COMMENT F3 PROGRAMMING AUX F8 FLOPPY F4 I/O FORCE ON/OFF F9 SET BUZZER VOL F5 CHECK SCAN TIME F10 NEXT	
③	Select online or offline, then ENT	SYSTEM DEFINITION SOURCE: ONLINE/OFFLINE	• Online: Registered directly in the processor • Offline: Registered in memory in the loader
④	Select and register to the end.	SYSTEM DEFINITION F1 SYSTEM REGISTRATION F2 T-LINK REGISTRATION F3 P-LINK REGISTRATION F4 MESSAGE REGISTRATION F5 ME-NET REGISTRATION	Select the function key from this screen to register the desired system definition.
⑤	Register the desired system. END	(See the following pages.)	
⑥	ENT	SYSTEM DEFINITION SOURCE: (ONLINE/OFFLINE)	When the message "Completion" is displayed, the system definition has been registered. For the timing of system definition recognition by the processor, see the preceding page.

2. System registration

Read the system registration selection screen to select F1 to F4.

① **Model code**
② **Fail-soft operation**
③ **WDT**

System registration selection screen



① Model code:
Use a numeric value to enter the series name of the object processor for programming in offline mode. Press arrow keys to select the desired model.

② Fail-soft operation setting:
Use a symbol to register whether fail-soft operation is set.

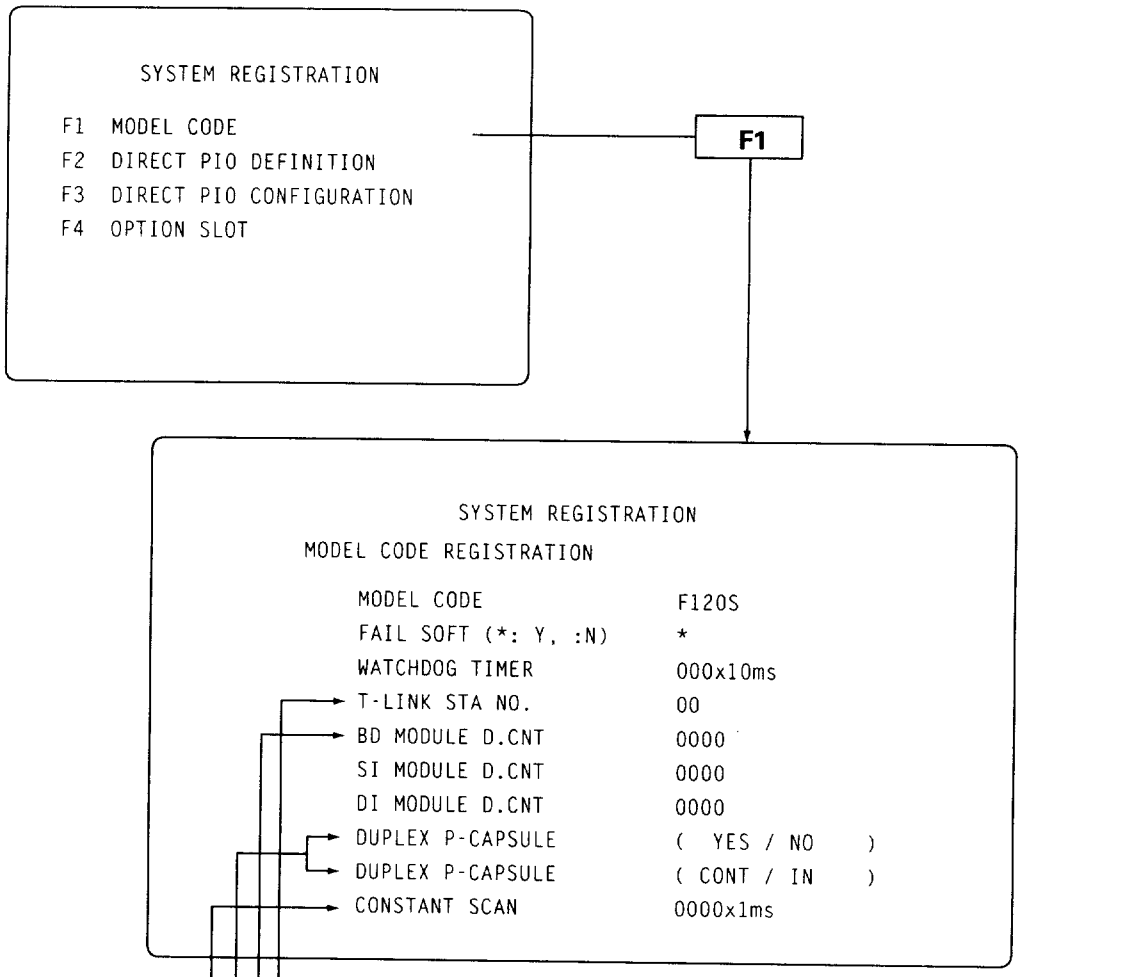
[*] : Set [SPACE] : Release

③ WDT time setting:
Use a numeric value to enter the WDT time to monitor the scan time. (The setting range is from 10 to 2560ms)

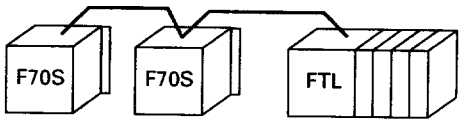
Example
1 x 10ms: 10ms
10 x 10ms: 100ms
0 x 10ms: Automatically set to 2560ms (maximum).

Note:
The watchdog timer time must be set at about 1.5 times as much as the system execution time. The timer time should be set only when it is especially required.

④ T-link station number
⑤ BD area size registration
⑥ Duplex processor availability and duplex processor mode selection



④ T-link station number:
Distinguish the main processor (set to 0) from the standby processor (set to 1) at POWER-ON.



Generally, the first processor that starts at POWER-ON becomes main. If the processor starts concurrently, the processor, module with 0 specified becomes main.

Note: The number of SI and DI module data cannot be sent in the F150S or smaller series.

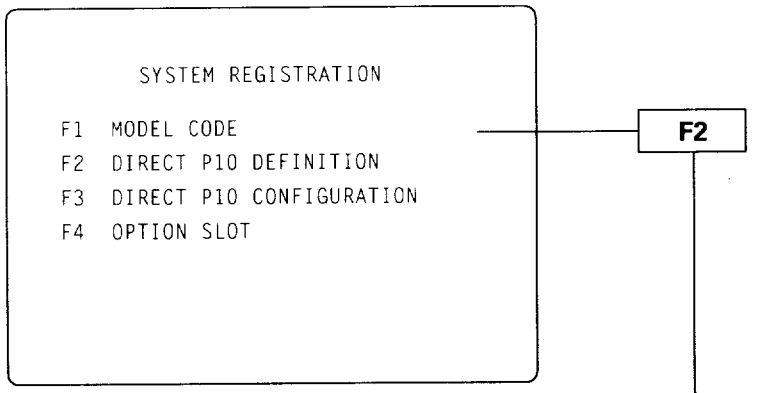
⑤ BD area (32 bits) size registration:
Registered to use a BD area of more than 256 words.
For example, set 266 to use 266 words. This is 10 words more than 256 words.

⑥ Duplex processor availability, Duplex mode selection:
Set whether the duplex processor is available and whether the data memory content remain unchanged or initialized (called initial start) when the duplex processor is switched. (For F70S, F120 to F150S, initial start only.)

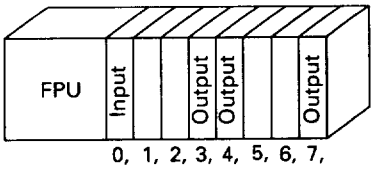
Constant scan
The constant scan cannot be set in the F150S or smaller series.

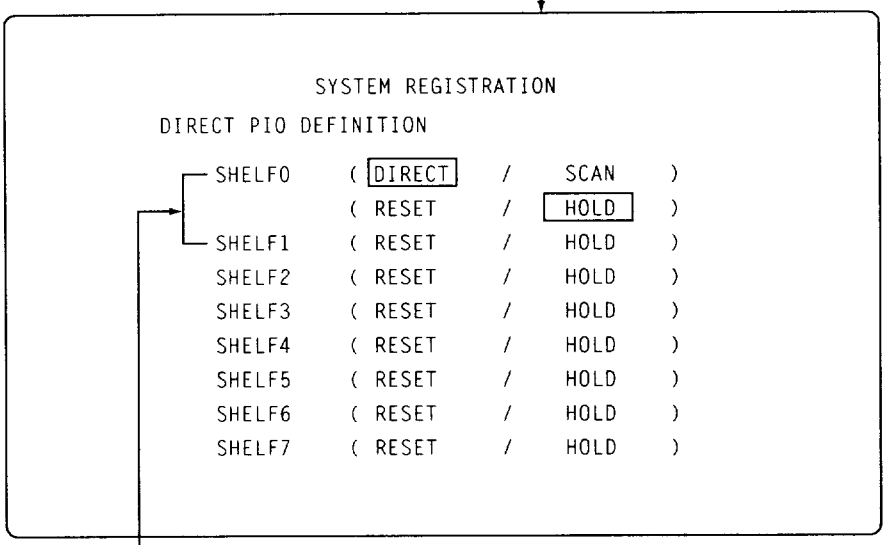
⑦ Direct I/O (processor I/O) operation mode registration

System registration selection screen



System configuration example





Direct access mode registration:
The direct access area can be used.

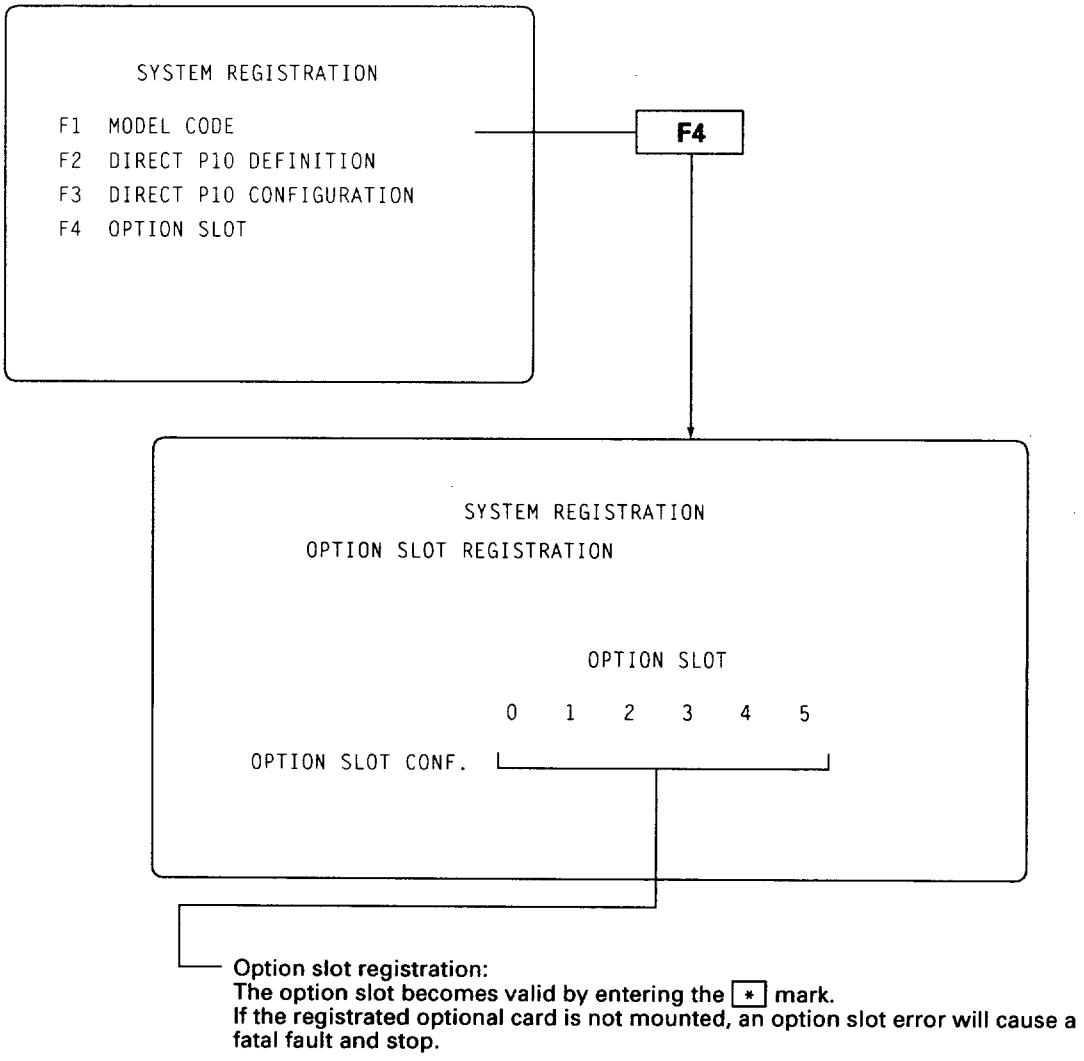
F80H, F70, F70S (No option slots)	W24.0 to W24.159
F120H, F70S (with option slots), F120S to F150S	W24.0 to W24.127
F55	W24.0 to W24.255

Direct access mode must be registered to use the module ready for interrupt processing.

- Hold (mode) registration:
  If the processor stops (fatal fault occurs and operation stops), the output status prior to stop is held.
  In the system configuration example, the output status of station numbers 3, 4, and 7 is held.
- Scan synchronization (mode), reset mode:
  In the normal system, the operation mode of the program is scan synchronization mode and reset mode for output.

⑧ **Option slot registration**
Used to manage the status in which the optional card (T-link, P-link) is mounted.

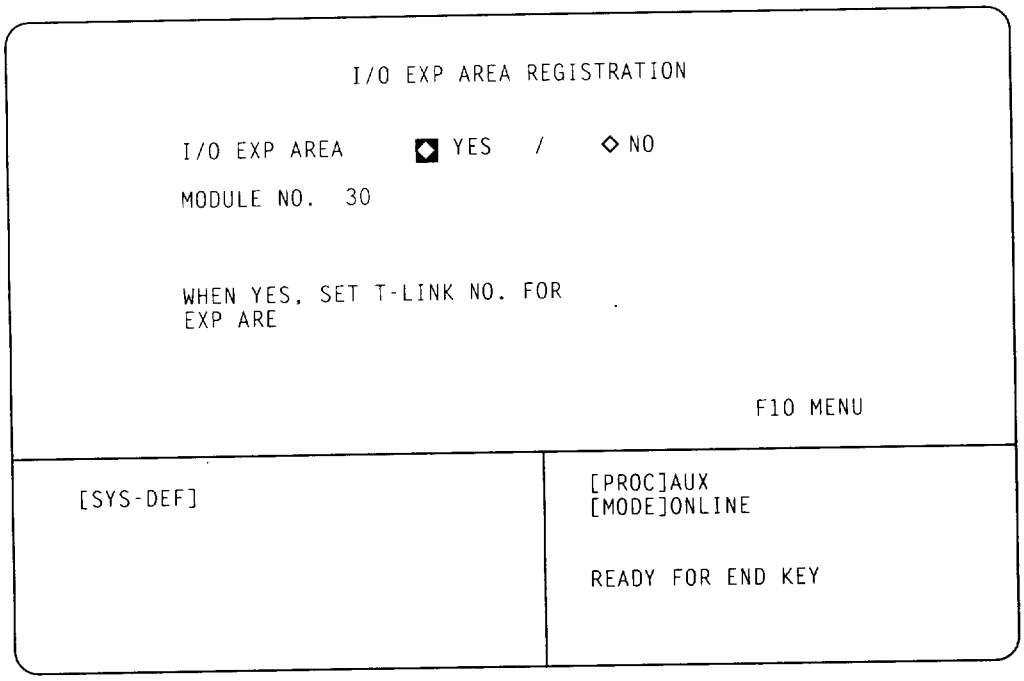
System registration selection screen



Option slot registration:
The option slot becomes valid by entering the [*] mark.
If the registrated optional card is not mounted, an option slot error will cause a fatal fault and stop.

3. T-link registration

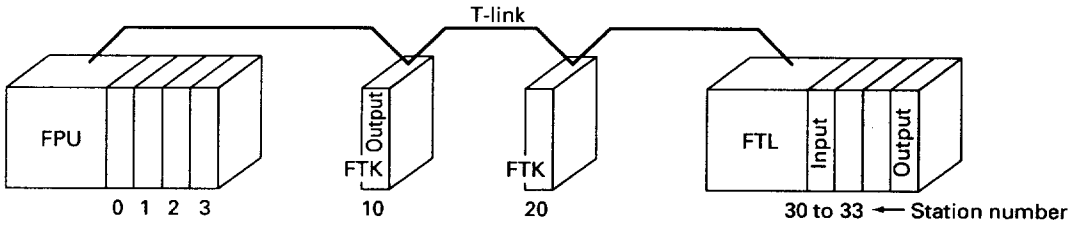
- I/O expansion area registration
  To use the I/O expansion area, specify "YES" to set the module No.



Note: Set the module No. to be used so that it is not the same as the other module No. (PE-link and ME-NET) in the system definition. The module No. must not be same as file No. and table No. in the user program. (If they are same, a user program error will occur and the processor will stop due to a fatal fault.)
I/O expansion area function is available in the F70S, F120S, F140S, and F150S series only.

① Individual fail-soft
② T-link registration
③ Hold (output hold) setting
④ T-link group registration

The following system example is used for an explanation.



Channel No.:
Read the desired link (channels 0 to 3) screen to set the channel No

```
                T-LINK REGISTRATON (1/4)   OCH
                FAIL-SOFT  YES/NO
                REGIST  YES/NO
                STOP WITH HELD
NO.  GROUP   NO.  GROUP   NO.  GROUP   NO.  GROUP
     0 1 2 3       0 1 2 3       0 1 2 3       0 1 2 3
00   * *     25               50               75
01             26               51               76
02             27               52               77
03             28               53               78
04             29               54               79
05             30               55               80
06             31               56               81
07             32               57               82
08             33               58               83
09             34               59               84
10             35               60               85
11             36               61               86
12             37               62               87
13             38               63               88
14             39               64               89
15             40               65               90
16             41               66               91
17             42               67               92
18             43               68               93
19             44               69               94
20             45               70               95
21             46               71               96
22             47               72               97
23             48               73               98
24             49               74               99
```

Note 1: To set a station number, register the first station number.

Example: When station number 0 is specified, station numbers 0, 1, 2, and 3 become valid.

Note 2: For T-link registration, be sure to set the [*] mark for the processor station number (0) to include the direct I/O. (Not necessary for F30, F50, and F50H series.)

④ T-link group registration:
- Set the [*] mark n the group 0 station number for which I/O processing is executed in synchronization with the fixed-cycle program.
- Set the [*] mark in the group 1 station number to be assigned to the I/O expansion area.
- For the F250 series, the meaning is different. Refer to the F250 series manual for details.

② T-link configuration registration:
Set the [*] mark to limit the capsule to be operated on the T-link.
In the system example, capsules with station numbers 0, 1, 2, 3, 10, 30, 31, 32, and 33 can be used. (Note 1)

① Individual fail-soft registration
The faulty capsule/module is disconnected and the normal capsule/module is operated continuously.
In the system example, modules with station numbers 0, 1, 2, and 3 become valid. (Note 1)

③ Hold registration:
Set the [*] mark in the station number of the output capsule for which the status prior to failure is to be held if a processor failure occurs. In the system example, the capsule with station No. 10 becomes valid. (Note 1)

Section 4 System Registration and Expansion Function

4. P-link/PE-link registration

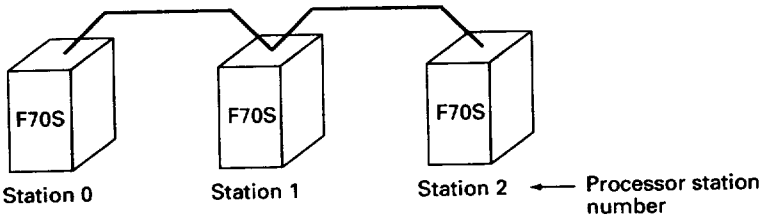
One F70S (NC1P-S2), F120S to F150S series processor can contain the P-link or PE-link with up to two channels (provided with an optional card). The P-link can be used with the PE-link but cannot be connected together.

Up to two P-link or PE-link cards (P+P, P+PE, PE+PE) can be mounted on one and the same processor module.

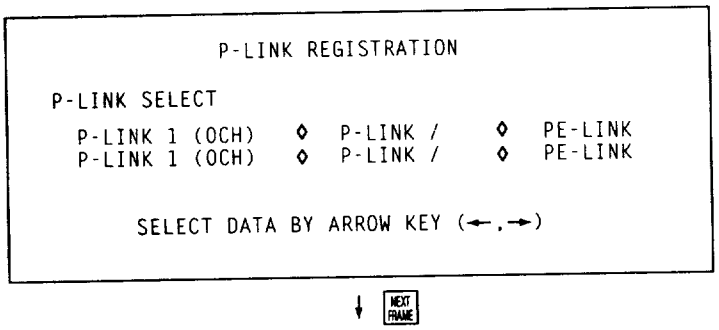
Register the following items for each channel and use the P-link or PE-link.

① **P-link/PE-link selection**
② **P-link configuration registration**
③ **Station number registration**
④ **Memory size registration**

The following system example is used for explanation.



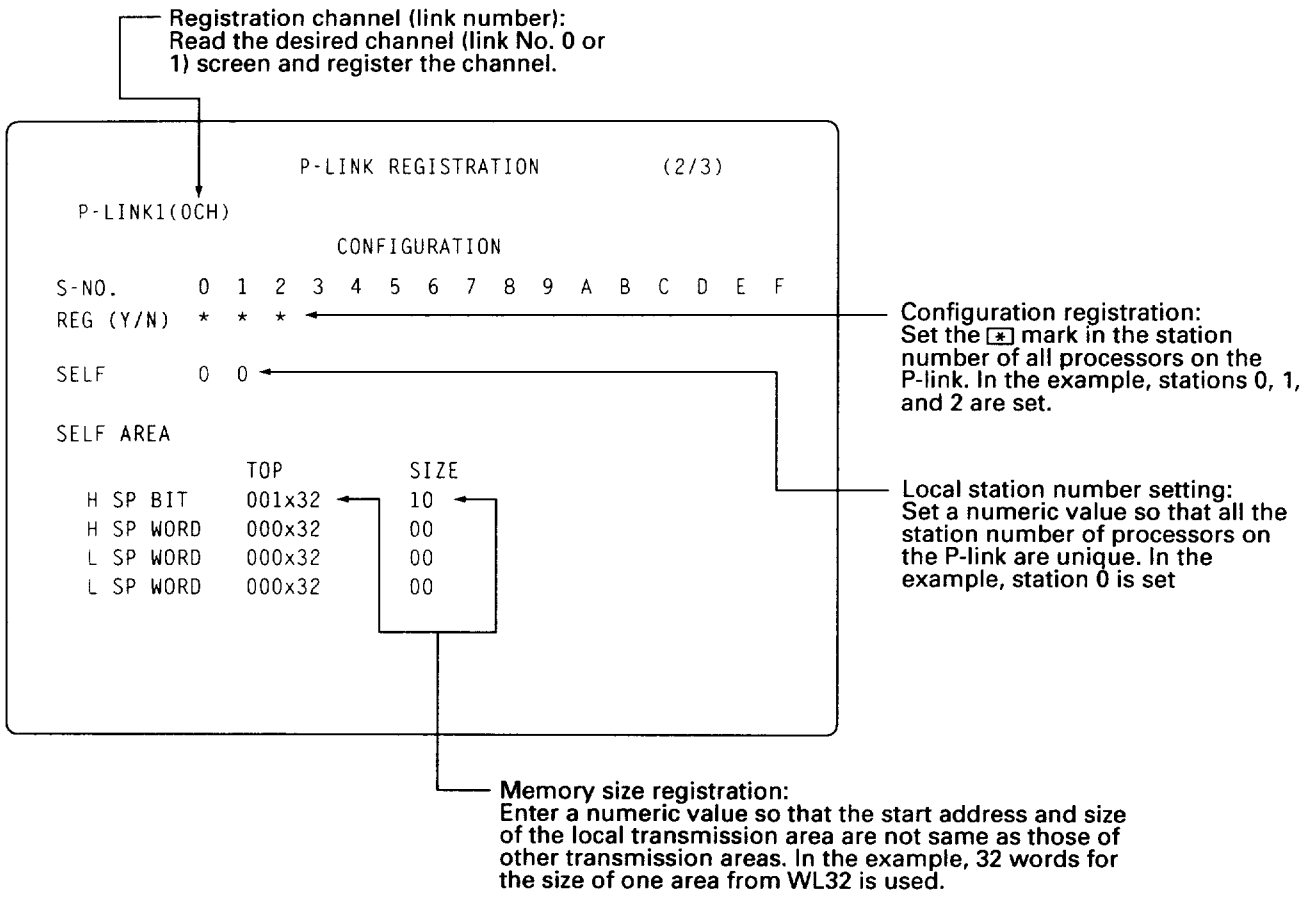
Select whether to use the P-link or PE-link.



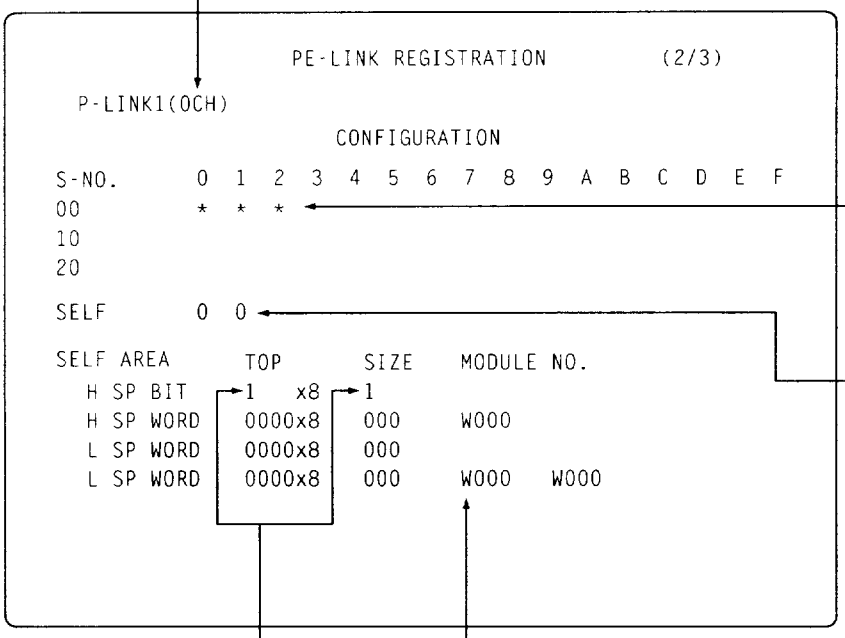
Register the configuration, station number, and memory size.

Configuration registration, station number registration, and memory size registration

• P-link



• **For PE-link**



Registration channel (link number):
Read the desired channel (link No. 0 or 1) screen and register the channel.

Configuration registration:
Set the [*] mark in the station number of all processors on the PE-link. In the example, stations 0, 1, and 2 are set.

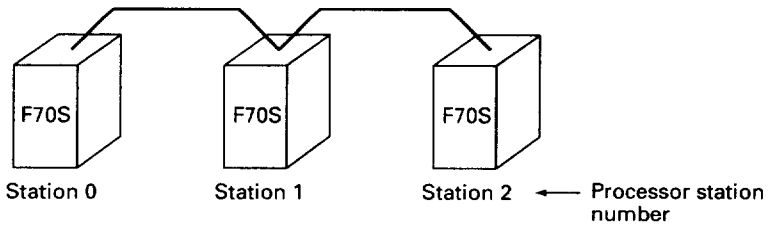
Local station number setting:
Set a numeric value so that all the station numbers of processors on the PE-link are unique. In the example, station 0 is set.

Memory size registration:
Enter a numeric value so that the start address and size of the local transmission area are not the same of those of other transmission areas. In the example, 8 words for the size of one area from WL8 is used.

Expansion module No.:
Set expansion module No. to expand memory in block 2 (high-speed word) and block 4 (low-speed two words) on the PE-link. Set 30 to 109 so that the module No. is not the same as the remote module No. (I/O expansion and ME-NET) in the system definition or file No. and table No. in the user program.
(If they are same, a user program error will occur and the processor will cause a fatal fault and stop.)
Memory of 4096 words is allocated for one module.
If memory is expanded, memory in the PE-link card is used, and user file memory is not reduced.
Because memory expansion can be set by this system definition only, the number used for the module expansion must not be used for the FILE definition.

5. Message module registration

Message module registration is used to communicate with a specific station on P-, PE- or T-link. An example of transmission from station 0 to station 2 in the following system is explained.



(1) Registering transmission station (station 0)

① **Data module number registration**
② **Message module use registration**
③ **Link selection**
④ **Remote station number setting**

MESSAGE MODULE REGISTRATION

DATA MODULE
SET (0: N. USED 1: INIT 2: SND 3: RCV)
LINK (0-3: T-LINK, 4-5: P-LINK, 6: SUMINET, 7: W24)
CAPSULE NO.
ETC
CH

NO.							NO.							NO.						
00	000	0	0	000	000	0	20	000	0	0	000	000	0	40	000	0	0	000	000	0
01	000	0	0	000	000	0	21	000	0	0	000	000	0	41	000	0	0	000	000	0
02	030	2	4	002	000	0	22	000	0	0	000	000	0	42	000	0	0	000	000	0
03	000	0	0	000	000	0	23	000	0	0	000	000	0	43	000	0	0	000	000	0
04	000	0	0	000	000	0	24	000	0	0	000	000	0	44	000	0	0	000	000	0
05	000	0	0	000	000	0	25	000	0	0	000	000	0	45	000	0	0	000	000	0
06	000	0	0	000	000	0	26	000	0	0	000	000	0	46	000	0	0	000	000	0
07	000	0	0	000	000	0	27	000	0	0	000	000	0	47	000	0	0	000	000	0
08	000	0	0	000	000	0	28	000	0	0	000	000	0	48	000	0	0	000	000	0
09	000	0	0	000	000	0	29	000	0	0	000	000	0	49	000	0	0	000	000	0
10	000	0	0	000	000	0	30	000	0	0	000	000	0							
11	000	0	0	000	000	0	31	000	0	0	000	000	0							
12	000	0	0	000	000	0	32	000	0	0	000	000	0							
13	000	0	0	000	000	0	33	000	0	0	000	000	0							
14	000	0	0	000	000	0	34	000	0	0	000	000	0							
15	000	0	0	000	000	0	35	000	0	0	000	000	0							
16	000	0	0	000	000	0	36	000	0	0	000	000	0							
17	000	0	0	000	000	0	37	000	0	0	000	000	0							
18	000	0	0	000	000	0	38	000	0	0	000	000	0							
19	000	0	0	000	000	0	39	000	0	0	000	000	0							

For station number assigned to the I/O expansion area in T-link mode, set 1.
For a station number assigned to the WB area, set 0.

Channel number:
Unused

① Data module registration:
Use a numeric value to set the data file No. In the example, file No. 30 is used.

④ Station number setting:
Use a numeric value to set the remote station number. In the example, the remote station number is 2.

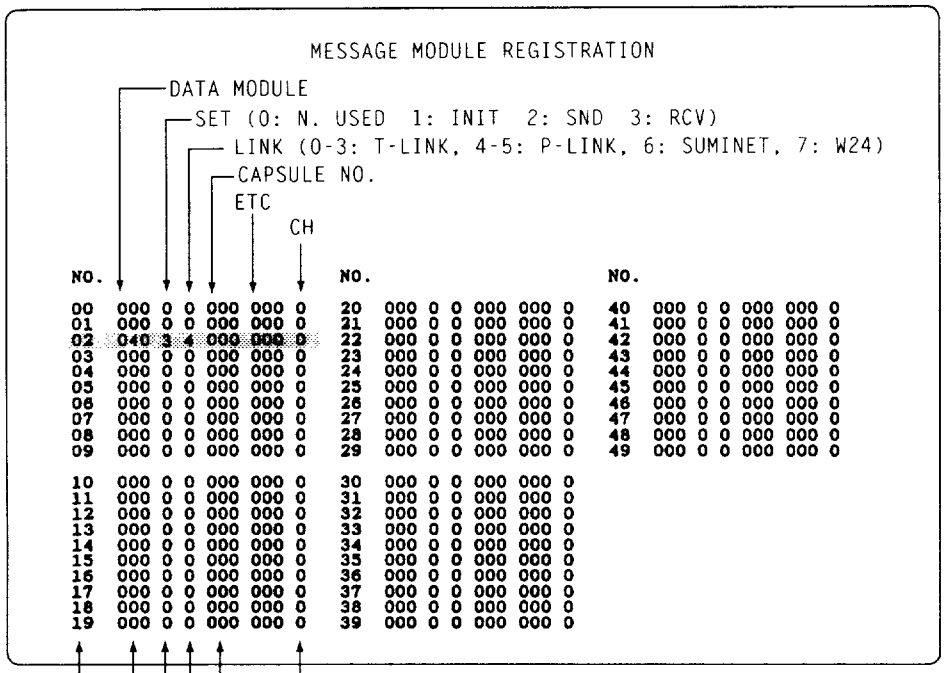
Message (communication) module No. registration:
Use a numeric value to set the message No. Set the same number for the remote module No. In the example, the module No. 2 is registered.

③ Communication link selection:
In the example, the P-link is set.

② Message module use registration:
Set a numeric value according to the use. In the example, the message module is used for transmission.

(2) Registering a reception station (station 2)

① **Data module number registration**
② **Message module use registration**
③ **Link selection**
④ **Remote station number setting**



Channel number: Unused
④ Remote station number: In the example, the remote station number is 0.
③ Communication link selection: In the example, the communication link is a P-link.
② Use: In the example, the use is for reception.
① Data module file No.: In the example, the file No. is 40.
Message (communication) module No.: in the example, the module No. is 2.

⑤ Configuration registration (P-link registration)

Register the P-link in each processor for message communication.

Registering P-link in station 0

```
                    P-LINK REGISTRATION          (2/3)

  P-LINK1(OCH)

                        CONFIGURATION

S-NO.      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
REG (Y/N)  *     *

SELF         0

SELF AREA
                    TOP          SIZE
  H SP BIT      000x32         00
  H SP WORD     000x32         00
  L SP WORD     000x32         00
  L SP WORD     000x32         00
```

Registering P-link in station 2

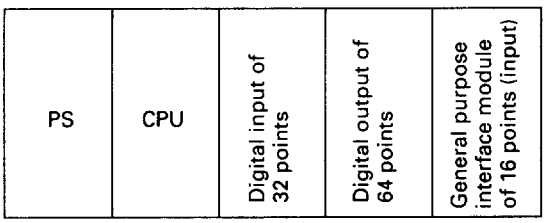
```
                    P-LINK REGISTRATION          (2/3)

  P-LINK1(OCH)

                        CONFIGURATION

S-NO.      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
REG (Y/N)  *     *

SELF         2

SELF AREA
                    TOP          SIZE
  H SP BIT      000x32         00
  H SP WORD     000x32         00
  L SP WORD     000x32         00
  L SP WORD     000x32         00
```

Note: After P-link registration is stored in each processor, perform a power reset. When the processor is turned on, it recognizes the registration contents.

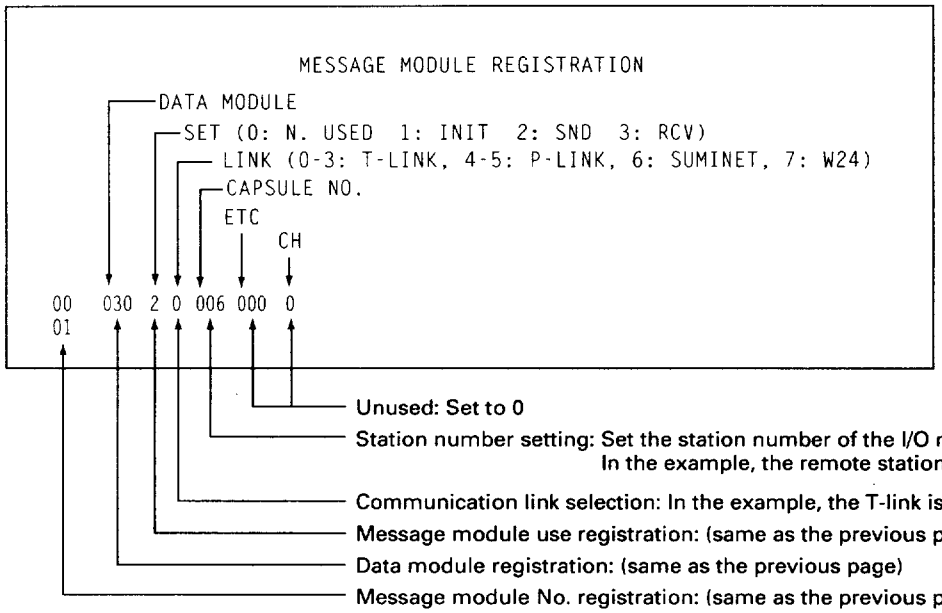
6. Message module registration (for direct I/O)

The following system configuration is used to explain the example of communicating with the direct I/O mounted on the processor unit.

The method of registration depends on whether the scan synchronization or direct access mode is used.



(1) Setting in scan mode



Unused: Set to 0

Station number setting: Set the station number of the I/O module to be communicated with.
In the example, the remote station number is 6.

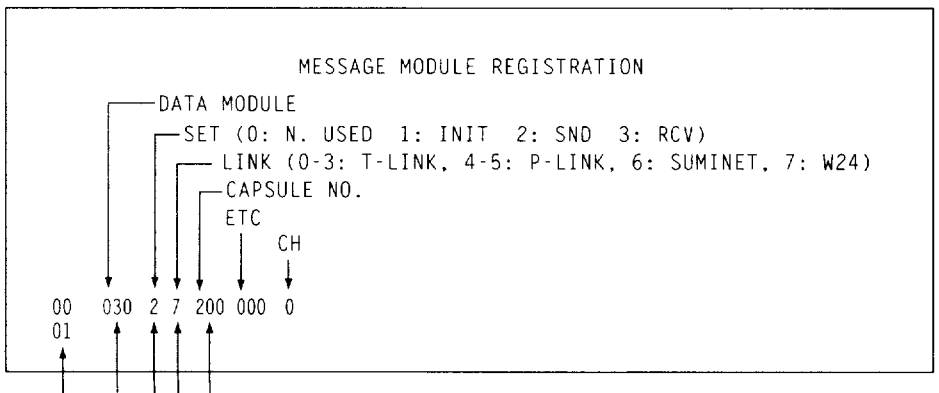
Communication link selection: In the example, the T-link is set.

Message module use registration: (same as the previous page)

Data module registration: (same as the previous page)

Message module No. registration: (same as the previous page)

(2) Setting in direct access mode



Station number setting: Set the slot number of the I/O module to be communicated with.
In the example, the slot number is set to 2.
(The slot number is counted as 0, 1, 2, ... sequentially from the left.)

Communication link selection: Set the slot number in direct access mode.

Message module registration: (same as the previous page)

Data module registration: (same as the previous page)

Message module No. registration: (same as the previous page)

4-2 Expansion functions

4-2-1 Direct access mode

I/O data for the MICREX-F is accessed by the following mode:

- Scan synchronization
- Direct access

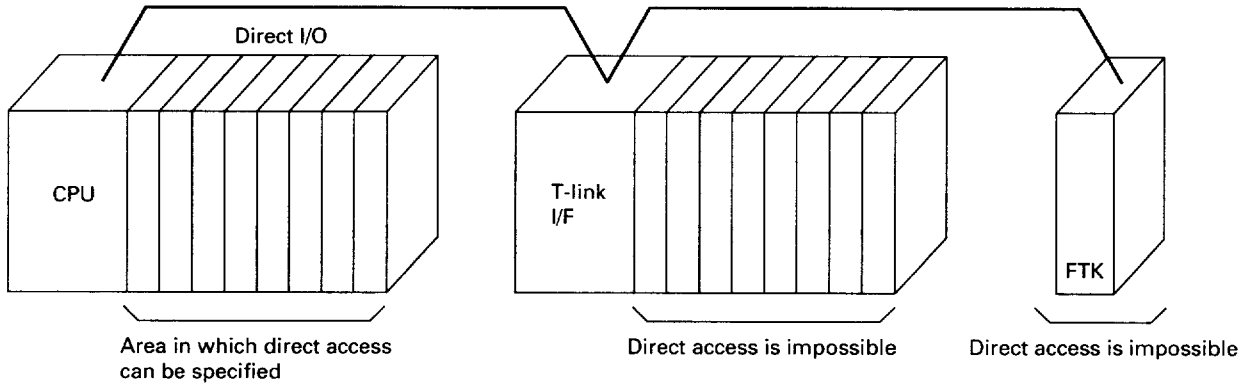
Scan synchronization is used to batch-process I/O before and after scanning. Direct access is used to access I/O data directly every time one circuit of the ladder program is executed.

The access mode for each series processor is as follows.

Series	F30, F50, F50H, F60	F55, F80H, F120H, F70, F70S, F120S to 150S
Scan synchronization	Available	Available
Direct access	Not available	Available

1. Scope of direct access

The I/O module (card) mounted on the base unit of the processor unit is called direct I/O. Direct access can be specified for this direct I/O. (Direct I/O without direct access specification can be used for conventional scan synchronization access.) Direct access cannot be specified for the I/O module on the T-link expansion unit (on which the T-link interface module is mounted) and the T-link capsule. The direct access mode cannot be used with scan synchronization mode on the same base unit.



F120H, F70S (NC1P-S2), F120S to F150S	W24.0 to W24.127
F80H, F70, F70S (NC1P-S0)	W24.0 to W24.159
F55	W24.0 to W24.255

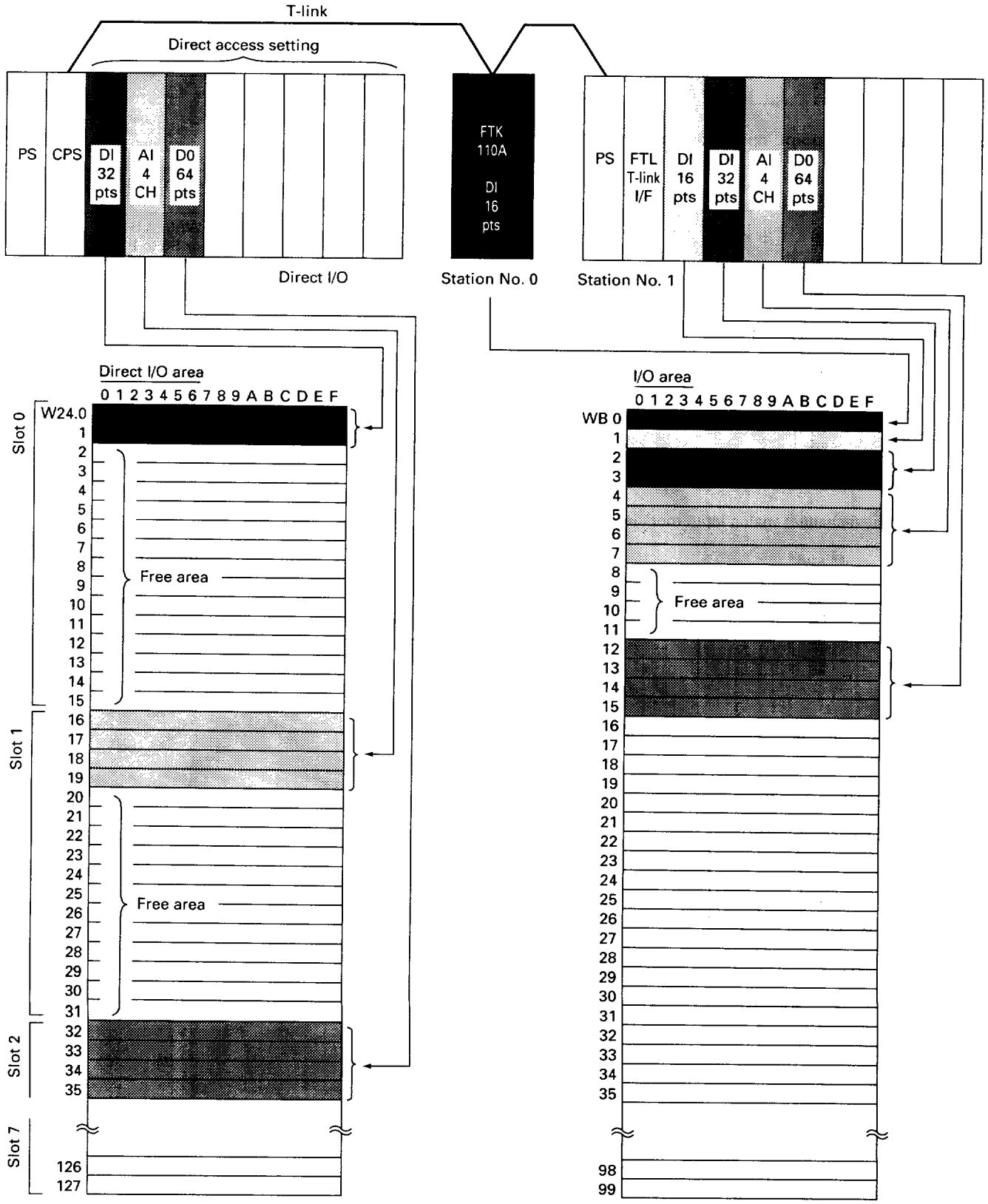
F55, F80H, F70	WB0 to WB99 can be used.
F120H, F70S, F120S to F150S	WB0 to WB399 can be used.

2. Specifying direct access area

Selection between direct access and scan synchronization is specified by system definition registration. See page 4-7, for direct I/O operation mode registration.

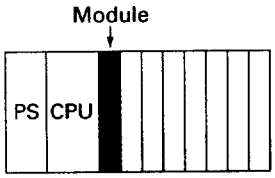
3. Address assignment for direct access area

In addition to the scan synchronization I/O area (WB area), 160 words (W24.0 to W24.159) (128 words for the processor with option slots) (16 bits) are provided as memory for direct access I/O modules. The address assignment is fixed to the slot, and 16 words are assigned to each slot. The unused area becomes a free area (see below).



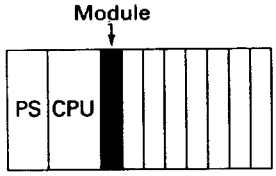
The examples of assigning direct I/O addresses are shown below.

Example 1: If the 16-point input module is mounted



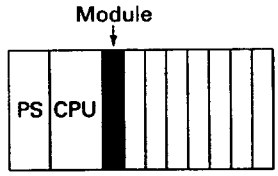
Direct access	Scan synchronization (conventional)
W24.0 (W24.1 to W24.15 are free)	WB0 or B0 to BF

Example 2: If the 32-point input module is mounted



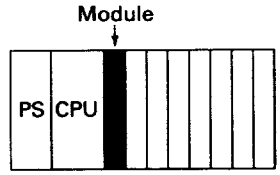
Direct access (conventional)	Scan synchronization
W24.0, W24.1 (W24.2 to W24.15 are free)	WB0, WB1, or B0 to B1F

Example 3: If the 64-point input module is mounted



Direct access (conventional)	Scan synchronization
W24.0 to W24.3 (W24.4 to W24.15 are free)	WB0 to WB3, or B0 to B3F

Example 4: If the 8-channel analog input module is mounted



Direct access (conventional)	Scan synchronization
W24.0 to W24.7 (W24.8 to W24.15 are free)	WB0 to WB7 or B0 to B7F

Address assignment for 8-channel analog input module

W24.0	0CH
W24.1	1CH
W24.2	2CH
W.24.3	3CH
W24.4	4CH
W24.5	5CH
W24.6	6CH
W24.7	7CH

4-2-2 Password

Outline

A personal identification number of four alphanumeric characters is used to protect the program.
This personal identification number is called a password.

Description

1. **Setting the password** (operation by the loader LITE)
   1) Use the loader to write a program.
   2) Press the [F5] (AUXILIARY), press [F10] (NEXT) twice, then press the [F1] (PASSWORD).
   3) The following screen appears on the loader display.

```
PASSWORD
F1    PASSWORD SET
F2    PROTECTION CANCEL
F3    PROTECTION SET
F4    PASSWORD CANCEL
```

   Press the [F1] key.

   4) Set a password in the range from 0000 to 3FFF.
   5) After the password has been set, select whether a password is available or unavailable, and press the [F3] (TRANSFER/VERIFICATION) to transfer data from the loader to processor.

Note: Reading a program having a password or deleting a password assigned to a program cannot be performed between the program loader and floppy disk or between the program loader and audio CMT.
However, a program with a password can be transferred from the loader.
The transfer destination equipment is a P capsule, floppy disk, audio CMT, or EPROM.

(2) Password error message

Error message	Action
Password out of range	Correct the password setting.
Unexecutable	Reset protection. (Enter the password assigned to the processor.)
Password is incorrect.	Enter the password assigned to the processor.
Data is not registered.	Set protection.
Password not registered	Set a password

Note: Refer to the Loader LITE User's Manual (No. LEH915) for details.

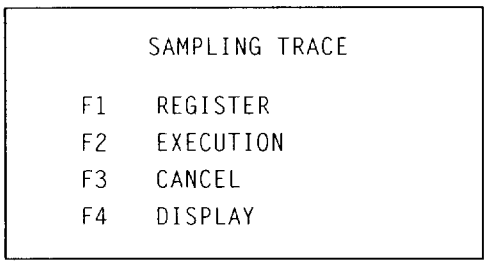
4-2-3 Sampling trace (for F80H, F120H, F70, F70S, F120S, F140S, F150S only)

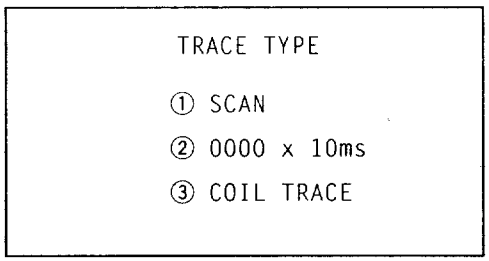
Outline

The sampling trace function enables the user to store data before and after the point specified by the program loader (D20, D25 LITE, Loader software) in sampling trace memory of the program loader. Changes in data can be traced and a history of changes in data can be checked by monitoring the memory. Data to be sampled can be registered up to eight points bit data and up to three points of word data in the loader. The sampling interval can be set for each scan or each time (10ms to 99990ms). The sampling point can be specified by key input or the annunciator relay in the program. Because changes in data before and after sampling point can be monitored, this function is useful for tracing the cause of an event.

Description

1. **Setting sampling trace** (operation by the loader LITE)
   (1) Read the sampling trace screen.
   Press the keys as follows:
   [F5] (AUXILIARY) → [F10] (NEXT) → [F5] (SAMPLING TRACE
   (2) Press the [F1] key.
   (3) Set the trace type. (Set the sampling point (time) using [*] mark and numeric values.)
   ① To sample data using scan synchronization, set the [*] marks at the cursor position _.
   ② To sample data at any time interval, set the [*] mark and the sampling time interval in the range from 10ms to 99,990ms using numeric values.
   ③ Coil trace
   (4) Move the cursor to set the trace count in the range from 0 to 999.
   (5) Move the cursor to execute a sampling trace and to register the bit address and word address.
   Up to eight bit addresses (except SC) for a contact and coil can be registered.
   Up to three word addresses (including SC) can be registered.
   (6) After data has been set, press the [END] key (STORAGE), then press the [ENT] key.





Note: Sampling trace is not provided for the following processor types: F30, F50, F50H, F60, and F55.

2. Sampling trace error message

Error message	Action
DATA NOT REGISTERED	Register the trace type, trace count, bit address, and word address in the trace designation.
INVALID INPUT	Move the cursor to the blinking area to register data. (The bit address and word address areas do not blink.)
TOO MANY REGISTERING	Move the cursor to change the number of registered data items for the bit address and word address. (The number of registered data items is 8 for a bit address and 3 for a word address.)
CANCEL IS REQUIRED	Cancel old data.
ERROR RANGE SPECIFIED	Move the cursor to the blinking area to set data registration correctly. (Bit address and word address)
DATA NOT FOUND	Set data correctly.
PROCESSOR CANNOT EXECUTE	Move the cursor to set data correctly. (Data set for trace count ranges from 0 to 999.)

Note: Refer to the loader LITE User's Manual (LEH915) for details.

4-2-4 Status latch (for F80H, F120H, F70, F70S, F120S, F140S, F150S only)

Outline

Status latch function enables the user to store data at any latch point in status latch memory in the processor.

Stored data can be monitored. A latch is executed by the annunciator relay of the user program, input switch in area B, or loader key input.

Description

1. **Setting status latch** (operation by the loader LITE)

(1) Read the status latch screen.
Press the keys as follows:
[F5] (AUXILIARY) → [F10] (NEXT) → [F6] (STATUS LATCH)

(2) Press the [F1] key.

```
STATUS LATCH

F1    REGISTER
F2    EXECUTE
F3    CANCEL
F4    DISPLAY
```

```
                STATUS LATCH REGISTER (*: ALL)

  B  *      BD   ---            W31        0 --- 40
  M  *      TS   ---            W□□□         ---
  K  *      TR0  --- 50 #1      W□□□         ---
  D  *      CS   ---            W□□□         ---
  F  *      CR * ---     #1     W□□□         ---
  A  *      Si   ---            W□□□         ---
  S  *      Di   ---            W□□□         ---
  T  *                          W□□□         ---
  C  *                          W□□□         ---
  L  *                          W□□□         ---
```

All bit areas are registered beforehand. Only these areas need not be registered and stored.

To register the data area

[*] : All-area specificatioan
Numeric input : Range specification

Registered to specify the area by the data module No. in this example, areas W31.0 to W31.40 in file No. 31 are specified.

Notes: #1 : The TR area from TR0 to TR50 is registered.
#2 : All CR areas are registered.

(3) After registration data has been entered, press the [END] key (STORAGE), then press the [ENT] key. When "COMPLETED" is displayed, this means the data has been registered.

2. Status latch error message

Error message	Action
NOT RESERVED	Register bit and word data items in display address reservation. (Up to 26 bits and words can be registered.)
TOO MANY REGISTERING	Return to the status latch screen. (When the number of registered bits and words exceeds 26)
ERROR RANGE SPECIFIED	Move the cursor to the blinking area to register the correct data.
CANCEL IS REQUIRED	Cancel old data.
DATA NOT REGISTERED	Register status latch.
DATA NOT FOUND	Register data.

Note: Refer to the Loader LITE User's Manual (No. LEH915) for details.

Appendix

Appendix 1 Scan Time

1-1 Calculation Method

1. F30, F50 and F50H Series

The approximate scan time can be calculated from the following expression:

Scan time T = ①+②+③+④+⑤+⑥+⑦+⑧ ms

Item \ Processor			F50	F30	F50H	Remarks
Instruction execution time		①	Σ (execution time of each instruction) ms			See List of Instruction Processng Speeds on page A-7.
I/O data transfer time	Basic unit I/O time	②	0.22ms	0.11ms	0.17ms	
	Expansion unit I/O time	③	0.32ms/32 points	0.20ms/16 points	0.26 ms/32 points	
	T-link	④	0.3ms/16 points			Only when the T-link is set
T-link adapter processing time		⑤	0.38ms			Only when the T-link adapter is connected
Program loader processing time		⑥	0 to 0.50ms (depending on key operation contents)			Only when the program loader is connected
Self-diagnosis time		⑦	1.11ms	0.89ms	0.92ms	
Time measurements		⑧	0.62ms	0.32ms	0.35ms	

2. F60 Series

The approximate scan time can be calculated from the following expression:

$$\text{Scan time } T = \frac{10\ (①+②+③+④+⑤+⑥)}{10-(⑦+⑧+⑨)}\ \text{ms}$$

Item \ Processor			F60	Remarks
Instruction execution time		①	Σ (execution time of each instruction) ms	See List of Instruction Processing Speeds on page A-7.
I/O data transfer time	I/O unit on mini T-link and T-link	②	0.20ms per 1 unit of I/O unit capsule	Base unit I/O is also counted as 1 unit.
	FTL010A on T-link	③	For each FTL010A unit : 0.10 x (No. of 16-point FTU modules) + 0.12 x (No. of 32-point FTU modules) + 0.15 x (No. of 64-point FTU modules) + 0.10ms	No. of 16-point conversion FTU modules is as follows: 16 points: 1 module 32 points: 2 modules 64 points: 4 modules
Function capsule (FK) processing time		④	0.3ms	Time required regardless of function capsule connection
Program loader processing time		⑤	0.5ms	Only when the program loader is connected
Self-diagnosis time		⑥	1ms	
RAS processing time at I/O transfer end	Common processing	⑦	0.2ms	Once every 10ms
	I/O capsule	⑧	0.05ms per I/O capsule	
	FTL010A	⑨	For each FTL010A unit : 0.04 + 0.13 x (No. of FTU modules) ms	

Appendix 1 Scan Time

3. F55 Series

The approximate scan time can be calculated from the following expression:

$$\text{Scan time } T = \frac{10\ (①+②+③+④+⑤+⑥+⑦)}{10-(⑧+⑨+⑩+⑪)}\ \text{ms}$$

Item \ Processor			F55	Remarks
Instruction execution time		①	Σ (execution time of each instruction) ms	See List of Instruction Processing Speeds on page A-7.
I/O data transfer time	I/O card on basic unit (processor)	②	Σ (execution time of each instruction) ms 16 points I: 0.14 ms 16 points O: 0.14 ms 32 points I: 0.22 ms 32 points O: 0.35 ms + base time 0.98 ms *	Only when I/O cards are mounted Base time is incremented by processing time of the concentrated display section.
	FTK capsule on T-link	③	0.02ms per FTK capsule	
	FTL010A, FTL010H and NC1ET on T-link	④	For each unit : 0.01 x (No. of I/O words) ms	1 word = 16 bits
Function capsule (FK) processing time		⑤	0.7ms	Time required regardless of function capsule connection
Program loader processing time		⑥	0.3ms	Only when the program loader is connected
Self-diagnosis time		⑦	2.7ms	
Fixed-cycle interrupt processing time		⑧	0.5ms	Once every 10ms
RAS processing time at I/O transfer end	Common processing	⑨	0.13ms	Once every 10ms
	FTK capsule	⑩	For each FTK capsule: 0.16ms	
	FTL010A FTL010H NC1ET		For each unit : 0.16ms + 0.01 x (No. of I/O cards) ms	

* Base time is 0.57ms when no I/O card is mounted on the basic unit.

4. F70, F80H and F120H Series

The approximate scan time can be calculated from the following expression:

$$\text{Scan time } T = \frac{10\,(①+②+③+④+⑤+⑥+⑦+⑧+⑨)}{10-(⑩+⑪+⑫+⑬)} \text{ ms}$$

Item \ Processor			F80H	F120H	Remarks
Instruction execution time		①	Σ (execution time of each instruction) ms		See List of Instruction Processing Speeds on page A-7.
I/O data transfer time	I/O module on basic unit (processor)	②	Σ (access time per module) ms 16 points I: 0.14ms 16 points O: 0.14ms 32 points I: 0.22ms 32 points O: 0.35ms 64 points I: 0.38ms 64 points O: 0.44ms + base time 0.40ms	0.01 x (No. of input words) ms + 0.02 x (No. of output words) ms + base time 0.28ms	Only when I/O modules are mounted 1 word = 16 bits
	I/O module on FDL	③	—	For each FDL unit: 0.02 x (No. of input words) ms + 0.04 x (No. of output words) ms + Base time 0.43ms	Only when FDL is connected 1 word = 16 bits
	FTK capsule on T-link	④	0.02ms per FTK capsule	0.01ms per FTK capsule	
	FTL010A, FTL010H and NC1ET on T-link	⑤	For each unit: 0.01 x (No. of I/O words) ms	For each unit 0.01 x (No. of I/O words) ms	1 word = 16 bits
Function capsule (FK) processing time		⑥	0.7ms	1.1ms	Time required regardless of function capsule connection
Program loader processing time		⑦	0.3ms	0.3ms	Only when the program loader is connected
P-link data transfer time		⑧	—	5ms	Only when the P-link is used
Self-diagnosis time		⑨	2.7ms	2.47ms	
Fixed-cycle interrupt processing time		⑩	0.5ms	0.5ms	Once every 10ms
RAS processing time at I/O transfer end	Common processing	⑪	0.13ms	0.1ms	Once every 10ms
	FTK capsule	⑫	0.16ms per FTK capsule	0.12ms per FTK capsule	
	FTL010A FTL010H NC1ET	⑬	For each unit: 0.16ms + 0.01 x (No. of FTU modules) ms	For each unit: 0.12ms + 0.01 x (No. of FTU modules) ms	

Appendix 1 Scan Time

5. F70S, F120S, F140S and F150S Series

The approximate scan time can be calculated from the following expression:

$$\text{Scan time } T = \frac{10\ (①+②+③+④+⑤+⑥+⑨+⑩+⑪)}{10-(⑦+⑧)}\ \text{ms}$$

Item \ Processor			F70S, F120S, F140S, F150S	Remarks
Instruction execution time		①	Σ (execution time of each instruction) ms	See List of Instruction Processing Speeds on page A-7.
I/O data transfer time	I/O module on basic unit (processor)	②	See the next page.	Only when I/O modules are mounted. FDL cannot be connected to F70S series.
	FTK capsule on T-link	③	See the next page.	
Function capsule (FK) processing time		④	0.128ms	Time required regardless of function capsule connection
Program loader processing time		⑤	0.170ms	Only when the program loader is connected
Self-diagnosis time		⑥	0.826ms	
Fixed-cycle interrupt processing time		⑦	0.151ms	Once every 10ms
RAS processing time at I/O transfer end		⑧	See the next page.	Once every 10ms
Option processing time	T-link	⑨	0.32ms per optional T-link card	
	P-link	⑩	See the next page.	Only when the optional card is used.
	PE-link *1	⑪	See the next page.	

*1: PE-link is not available for F70S series.

• Details on each item

② I/O data transfer time (I/O module on basic unit)
Σ (access time per module) ms + base time 0.261ms

16 points I : 20μs
16 points O: 40μ ┐
32 points I : 32μ │ I/O module on basic unit
32 points O: 50μ │ (processor base board)
64 points O: 90μ ┘

③ I/O data transfer time (capsule/module on T-link)

- Capsule/module on standard T-link
  {0.0080 x nST + 0.0013 x (WST – nST)} ms
  nST: Total number of FTK and FFK capsules on T-link, and I/O modules on expansion T-link (An input and output module is counted as two.)
  WST: Total number of I/O words of capsule on T-link (If input and output are included in one word, they are counted as two words.)
- Capsule/module on option T-link
  {0.0050 x nOT + 0.0013 x (WOT – nOT)} ms
  nOT: Total number of FTK and FFK capsules on option T-link, and I/O modules on expansion T-link (An input and output module is counted as two.)
  WOT: Total number of I/O words of capsule on option T-link (If input and output are included in one word, they are counted as two words.)

⑧ RAS processing time at transfer end
If the total number of FTK and FFK capsules on the T-link, and I/O modules on expansion T-link to be connected is n:

- if n=0 0.051ms
- if n is not 0 0.104 + 0.035 x n (ms)

⑩ Option processing time (P-link)

- For 1 optional card
  0.5 + 0.085 x No. of transmission blocks + Transfer delay (ms) (Max. 5ms)
- For 2 optional cards
  1.0 + 0.085 x No. of transmission blocks + Transfer delay (ms) (Max. 5ms)

⑪ Option processing time (PE-link)

- For 1 optional card
  0.5 + 0.004 x No. of transmission blocks (ms)
- For 2 optional cards
  0.8 + 0.004 x No. of transmission blocks (ms)

Appendix 1 Scan Time

1-2 List of Instruction Processing Speeds

Units: μs

Classfication	Instruction	F30	F50	F50H	F60	F55/F70	F80H	F120H	F70S, F120, F140S, F150S
Sequence	NO contact	1.4	1	0.8	0.8	0.5	0.8	0.8	0.125
	NC contact	1.4	1	0.8	0.8	0.5	0.8	0.8	0.125
	Coil	1.6	1.25	1	1	0.625	1	1	0.25
	Return (origin) → N	1.4	1.25	1	0.8	0.5	0.8	0.8	0.25
	Return (destination) N ←	1.6	1	0.8	0.8	0.5	0.8	0.8	0.25
	Set	1.8	1.5	1.2	1.2	0.75	1.2	1.2	0.25
	Reset	1.8	1.5	1.2	1.2	0.75	1.2	1.2	0.25
	Rising edge differential	2	1.75	1.4	1.4	0.875	1.4	1.4	0.5
	Falling edge differential	2	1.75	1.4	1.4	0.875	1.4	1.4	0.5
	Inverse	1.4	1	0.8	0.8	0.5	0.8	0.8	0.125
	MCS	—	—	—	—	0.5	0.8	0.8	0.25
	MCR	—	—	—	—	0.5	0.8	0.8	0.125
	Shift register	One-word register: 210 During clock stoppage: 158	One-word register: 262 During clock stoppage: 198	One-word register: 210 During clock stoppage: 158	One-word register: 175 During clock stoppage: 110	One-word register: 156 During clock stoppage: 119	One-word register: 250 During clock stoppage: 190	One-word register: 108 During clock stoppage: 120	One-word register: 12.125 During clock stoppage: 7.5
	Step control	Contact: 1.6 Coil: 1.6	Contact: 1.25 Coil: 1.25	Contact: 1 coil: 1	Contact: 1 Coil: 1	Contact: 0.625 Coil: 0.625	Contact: 1 Coil: 1	Contact: 1 Coil: 1	Contact: 0.125 Coil: 0.5
Timer	ON-delay timer	11/5.4	11.5/4.5	9.2/3.6	0.01s timer: 6.4/3.4 0.1s timer: 65/56	0.01s timer: 4/2.125 0.1s timer: 78/675	0.01s timer: 6.4/3.4 0.1s timer: 125/108	0.01s timer: 6.4/3.4 0.1s timer: 125/108	0.01s timer: 8.9/6.4 0.1s timer: 7.4/6
	OFF-delay timer	—	—	—	105/89	127/119	203/191	125/108	9.25/7.88
	Integrating timer	—	—	—	119/89	155/127	248/203	138/120	9.88/6.63
	Monostable timer	—	—	—	123/78	137/111	219/177	123/99	9.13/6.38
	Monostable timer (Retriggalble)	—	—	—	123/78	141/112.5	225/180	123/99	10/6.38
Counter	Up counter	12.8/6.4	13/5	10.4/4	7.8/3.8	4.9/2.4	7.8/3.8	7.8/3.8	10.25/4.88
	Down counter	—	—	—	139/85	154/136	247/217	143/120	9/4.88
	Up/down counter	256/104	320/130	256/104	185/92	176/141	282/226	178/122	14.63/9.5
	Ring counter	—	—	—	125/72	129/112.5	207/180	128/97	11.75/4.88
Arithmetic operation	Addition	820/62	1030/65	820/52	201/42	204/2.6	327.4.2	215/4.2	0.63/0
	Subtraction	920/62	1150/65	920/52	208/42	199/2.6	318/4.2	215/4.2	0.63/0
	Multiplication	7240/62	9050/65	7240/52	356/42	338/2.6	541/4.2	345/4.2	4.25/0
	Division	1600 to 13200/62	2000 to 16500/65	1600 to 13200/52	334/42	322/2.6	515/4.2	330/4.2	6.13/0
	Division remainder	—	—	—	427/42	307/26	491/4.2	348/4.2	6.25/0
	Division (rounding off to the nearest whole number)	—	—	—	347/42	331/2.6	529/4.2	338/4.2	7.13/0
	Sign invert	—	—	—	139/42	143/3.2	229/3.4	148/3.4	2/0
	Increment	400/62	500/65	400/52	149/42	153/1.6	245/2.6	156/2.6	16.63/0
	Decrement	400/62	500/65	400/52	152/42	153/1.6	245/2.6	156/2.6	116.63/0
	Root	—	—	—	—	1237/2	1980/3.4	1440.3.4	56.63/0
	Absolute value	—	—	—	—	153/2	245/3.4	158/3.4	0.5/0

Note: N: No. of words

Units: μs

Classfication	Instruction	F30	F50	F50H	F60	F55/F70	F80H	F120H	F70S,F120, F140S, F150S
Comparison	>	320/62	400/65	320/62	146/42	177/2.6	283/4.2	178/4.2	Mode 1: 0.63/0 Mode 2: 0.88/0.5
	≥	330/62	410/65	330/62	144/42	177/2.6	283/4.2	178/4.2	Mode 1: 0.63/0 Mode 2: 0.88/0.5
	=	330/62	410/65	330/62	143/42	175/2.6	280/4.2	178/4.2	Mode 1: 0.63/0 Mode 2: 0.88/0.5
	≤	330/62	410/65	330/62	144/42	177/2.6	283/4.2	178/4.2	Mode 1: 0.63/0 Mode 2: 0.88/0.5
	<	320/62	400/65	320/62	144/42	177/2.6	283/4.2	178/4.2	Mode 1: 0.63/0 Mode 2: 0.88/0.5
	≠	330/62	410/65	330/62	143/42	175/2.6	280/4.2	178/4.2	Mode 1: 0.63/0 Mode 2: 0.88/0.5
	File comparison	—	—	—	—	299+84N /3.6	478+135N /5.8	298+96N /5.8	Mode 1: 43.13+1.5N Mode 2: 43.75+1.5N
Logical operation	AND	460/62	570/65	460/52	148/42	158/2.6	253/4.2	178/4.2	0.63/0
	OR	460/62	570/65	460/52	148/42	158/2.6	254/4.2	178/4.2	0.63/0
	Exclusive OR	460/62	570/65	460/52	148/42	158/2.6	254/4.2	178/4.2	0.63/0
	Invert	350/62	440/65	350/52	115/42	125/2.4	200/3.8	148/4.2	0.5/0
	Shift right logical	540/62	670/65	670/65	257/42	171/2.6	273/4.2	262/4.2	2.13/0
	Shift left logical	540/62	670/65	540/52	257/42	171/2.6	274/4.2	262/4.2	2.13/0
	Set bit	—	—	—	—	231/2.6	370/4.2	226/4.2	14.63/0
	Reset bit	—	—	—	—	231/2.6	370.4.2	226/4.2	14.63/0
	Test bit	—	—	—	—	181/2.6	391/4.2	226/4.2	Mode 1: 20.63/0 Mode 2: 24.25/0.5
Conversion	Binary/BCD conversion	680 to 1760/62	850 to 2200/65	680 to 1760/52	193/42	158/2	252/3.4	208/3.4	0.75/0
	BCD/binary conversion	1840/62	2300/65	1840/52	181/42	155/2	248/3.4	178/3.4	0.63/0
	Character string	—	—	—	—	144+5.6N /2	230+9N /3.4	145+5.5N /3.4	11.69+0.19N /0
	ASCII/numeric conversion	—	—	—	—	211/2	325/3.4	7.8/3.8	21.63/0
	Numeric/ASCII conversion	—	—	—	—	174/2	279/3.4	143/120	19.5/0
	Conversion to seconds: day, hour, minute, second → second	—	—	—	—	301/40	482/64	178/122	7.63/0
	Conversion from seconds: second → day, hour, minute, second	—	—	—	—	339/40	543/64	128/97	13/0
	Decode	1760/62	2200/65	1760/52	158/42	179/2	286/3.4	215/4.2	2.75/0
	Encode	840/62	1050/65	840/52	189/42	176/2	281/3.4	215/4.2	43.63/0
	7-segment decode	—	—	—	139/42	143/2	229/3.4	345/4.2	5.13/0
	Count ON-bit	840/62	1050/65	840/52	218/42	262/2	324/3.4	330/4.2	19.75/0

Appendix 1 Scan Time

Units: μs

Classfication	Instruction	F30	F50	F50H	F60	F55/F70	F80H	F120H	F70S,F120, F140S, F150S
Transfer	Data transfer	600/62	750/65	600/52	112/42	118/2	189/3.4	121/3.4	0.25/0
	Logical transfer	560/62	700/65	560/52	112/42	118/2	189/3.4	121/3.4	0.25/0
	Data block transfer	1.6 to 32ms/62	2 to 40ms/65	1.6 to 32ms/52	190+2.7N /42	274+3.4N /2.6	439+5.5N /4.2	285+1.3N /4.2	10.63+0.38N /0
	Logical block transfer				187+2.7N /42	270+3.4N /2.6	432+5.5N /4.2	281+1.3N /4.2	10.63+0.38N /0/0
	Digit transfer	1.5 to 2.1ms/62	2.1 to 2.6ms/65	1.5 to 2.1ms/52	Max. 251/42	Max. 197/3	Max. 315/5	Max. 293/5	0.63/0
	High-order digit transfer	—	—	—	115/42	133/2	213/3.4	145/3.4	0.25/0
	Low-order digit transfer	—	—	—	115/42	135/2	216/3.4	145/3.4	0.25/0
	Pattern clear	—	—	—	—	.247/5.6N /2.6	395+9N /4.2	260+6N /4.2	4.44+0.19N /0
	Search	—	—	—	—	276+36N /2.6	442+58N /4.2	278+42N /4.2	Mode 1: 3.75+1.75N /0 Mode 2: 38+1.75N /0.5
	Switch	—	—	—	—	176/3	272/5	172/5	3.625/0
	Message transmission	—	—	—	—	199+5.6N /50	318+8.9N /80	590+5.3N /64	374.63 +1.38N/15
	Message reception	—	—	—	—	744+4.4N /50	1190+7.1N /80	810+5.3N /64	90.63 +1.38N/15
Analog	Upper limit	—	—	—	142/42	164/2.6	263/4.2	174/4.2	3.13/0
	Lower limit	—	—	—	142/42	166/2.6	263/4.2	174/4.2	3.5/0
	Upper and lower limit	—	—	—	—	—	—	—	5.63/0
	Dead band	—	—	—	—	—	—	—	3.13/0
	Bias	—	—	—	—	—	—	—	2.75/0
	Filter	—	—	—	—	—	—	—	7.13/0
	Differential	—	—	—	—	—	—	—	7.38/0
	Integral	—	—	—	—	—	—	—	4.88/0
	Sampling hold	—	—	—	—	—	—	—	1.88/0
	Multi-percent	—	—	—	—	—	—	—	11.63/0
	Divide-percent	—	—	—	—	—	—	—	9.25/0

Note: N: No. of words

Appendix 1 Scan Time

Units: μs

Classfication	Instruction	F30	F50	F50H	F60	F55/F70	F80H	F120H	F70S,F120, F140S, F150S
File	File definition	50	62	50	47	0.5	0.8	0.8	0
	Data table definition	50	62	50	47	0.5	0.8	0.8	0
	Data	SI: 1.4 DI, BD: 2.8	SI: 1 DI, BD: 2	SI: 0.8 DI, BD: 1.6	0	0.5	0.8	0.8	0
	End of data	50	62	50	0	0.5	0.8	0.8	0
	File clear	640~4560 /62	800~5700 /65	640~4560 /52	94+2.1N /42	87.5+0.88N /1.1	140+1.4N /1.8	169+1.3N /1.8	7.32+0.19N /0
	Selector	550/62	680/65	550/52	165/42	221/2	353/3.4	224/3.4	10.75/0
	Deselector	480/62	600/65	480/52	177/42	216/2	345/3.4	220/3.4	12.13/0
	File store	—	—	—	131+2.7N /42	350+3N /1.6	560+5N /2.6	436+2.8N /2.6	38.63 +0.38N/0
	FIFO load	—	—	—	121+42	355+3N /1.6	568+5N /2.6	450+2.2N /2.6	38.76 +0.38N/0
	FILO load	—	—	—	133+2.7N /42	387+3N /1.6	620+5N /2.6	480+2.2N /2.6	43.38 +0.38n/0
	File read	—	—	—	—	420+3N /3	672+5N /5	504+2.6N /5	62.63 +0.38N/0
	File write	—	—	—	—	427+3.3N /3	683+5.3N /5	507+2.6N /5	62+0.38N /0
	File information	—	—	—	—	154/1.6	246/2.6	153/2.6	10.63/0
Program control	Program entry	—	—	—	—	2	3.4	0.8	0.125
	Program end	42	52	42	43/42	79	126	50/118	3.38/0
	FM call	—	—	—	—	—	—	158+5.3N /118	11.25/0
	FM start	—	—	—	—	—	—	1.6	4.13 (No. of work areas: 1) 9.88 (No. of work areas:32)
	FM end	—	—	—	—	—	—	95/1.8	43.38/0
	Skip	—	—	—	—	79/1.1	126/1.8	74/1.8	2.5/0
	Skip end	—	—	—	—	0.5	0.8	0.8	0.125/0
	Disabled interrupt	—	—	—	—	141/2	226/3.4	110/3.4	4.88/0
	Enabled interrupt	—	—	—	—	141/2	226/3.4	110/3.4	23.63/0
	Jump	163/62	110/65	163/52	66/42	71/1.1	113/1.8	64/1.8	2.5/0
	Jump end	50	62	50	47	0.5	0.8	0.8	0.125
	Loop	—	—	—	—	151/1.6	241/2.6	173/2.6	18.25/5.13
	Continue	—	—	—	—	88	141	88	6.25
	Push	—	—	—	—	—	—	82/118	7.75/0
	Pop	—	—	—	—	—	—	82/118	9.13/0
	Load effective address	—	—	—	—	—	—	111/2.6	20.25/0
	Index register addition	—	—	—	—	—	—	180/2.6	15.75/0
	Index register subtraction	—	—	—	—	—	—	180/2.6	15.75/0
Others	Page	2.8	2	1.6	1.6	1	1.6	1.6	0
	Pass	1.4	1	0.8	0.8	0.5	0.8	0.8	0.125
	Blank	1.4	1	0.8	0.8	0.5	0.8	0.8	0.125

Note 1: N: No. of words
Note 2: For the F70S, F120S, F140S and F150S Series, the execution time of a conditional contact in the data instruction is 0.357 (μs). The data instruction between master control instructions, 0.25 (μs) is incremented regardless of the conditional contact existence.

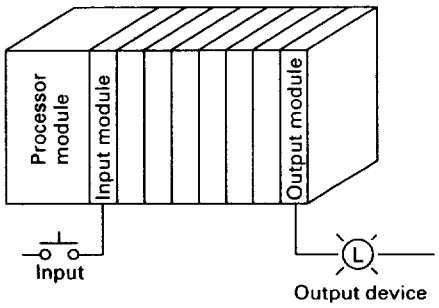
Appendix 2 I/O Response

2-1 For Direct I/O Modules

The I/O module mounted on the base unit of the processor module is called direct I/O. When program processing is in the scan synchronization mode, processing speed is the highest with this direct I/O system.

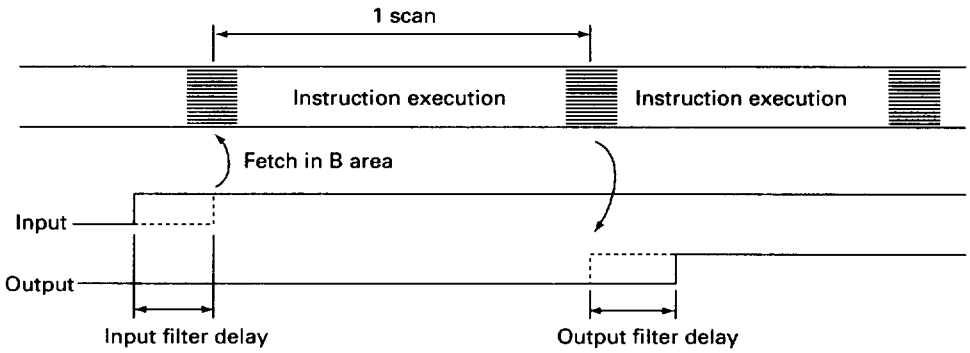
Note: In the F60 series, this is also true of I/Os connected by a mini T-link.

Configuration example



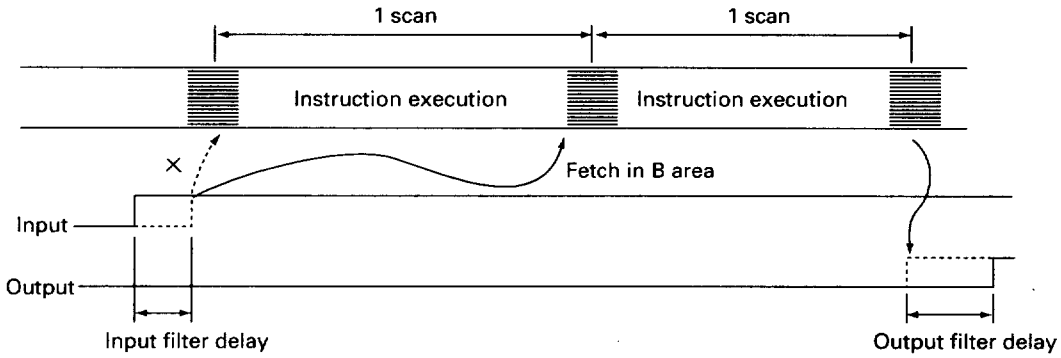
Circuit example

Minimum I/O response time



Minimum I/O response time = Input filter delay + 1 scan time + Output response delay

Maximum I/O response time



Maximum I/O response time = Input filter dealy + (1 scan time) x 2 + Output response delay

Example of calculating I/O response time

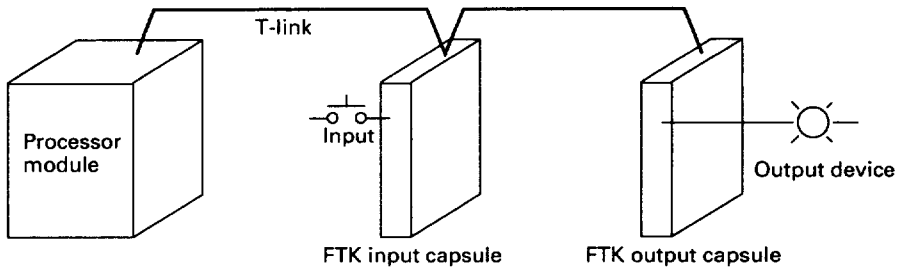
Input module OFF → ON response time = 3ms
Output module OF F → ON response time = 1ms
Scan time: 20ms

- Minimum I/O response time = 3ms +20ms x 1 + 1ms
  = 24ms
- Maximum I/O response time = 3ms + 20ms x 2 + 1ms
  = 44ms

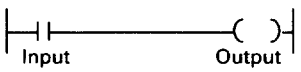
2-2 For FTK I/O Capsules

For the FTK I/O capsules installed on the T-link, the response time includes sampling delay on the T-link and processor buffer delay.

Configuration example

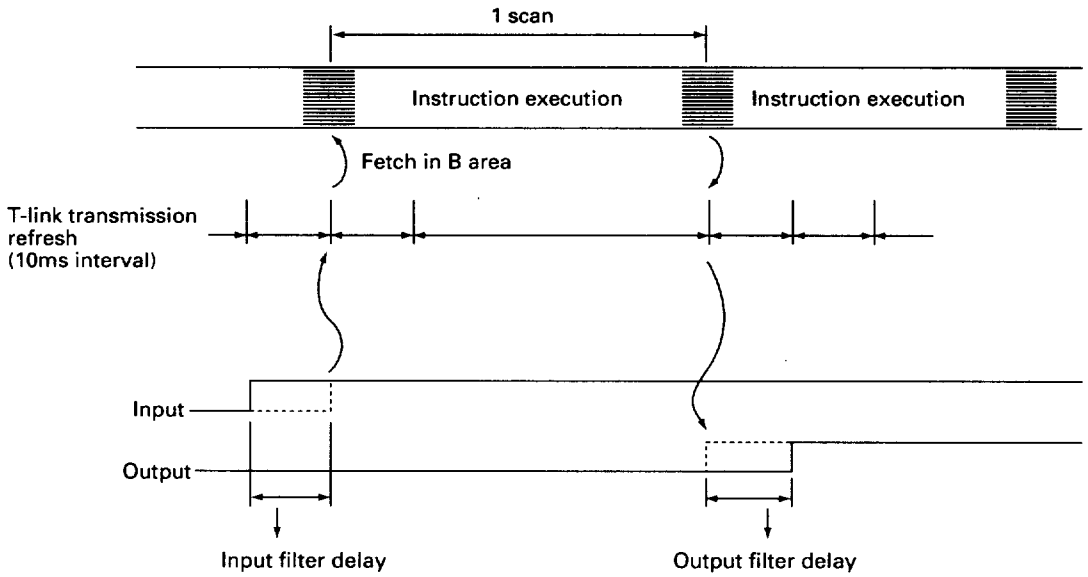


Circuit example



The input is an input capsule connected to the T-link and the output is an output capsule connected to the T-link

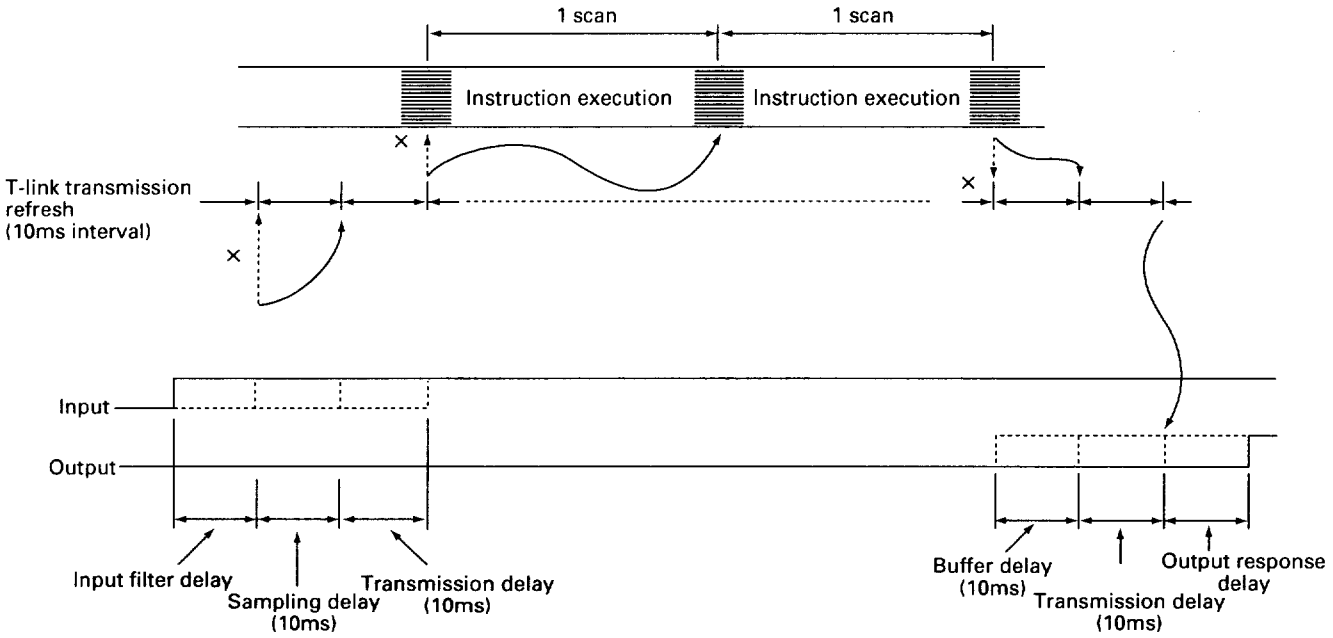
Minimum I/O response time



Minimum I/O response time = Input filter delay + 1 scan time + Output response delay

Appendix 2 I/O Response

Maximum I/O response time



Maximum I/O response time
= Input filter delay + ***Sampling delay*** + ***Transmission delay*** + (1 scan time) x 2
+ ***Buffer delay*** + ***Transmission delay*** + Output response delay
= Input filter delay + (1 scan time) x 2 + Output response delay + ***40ms***

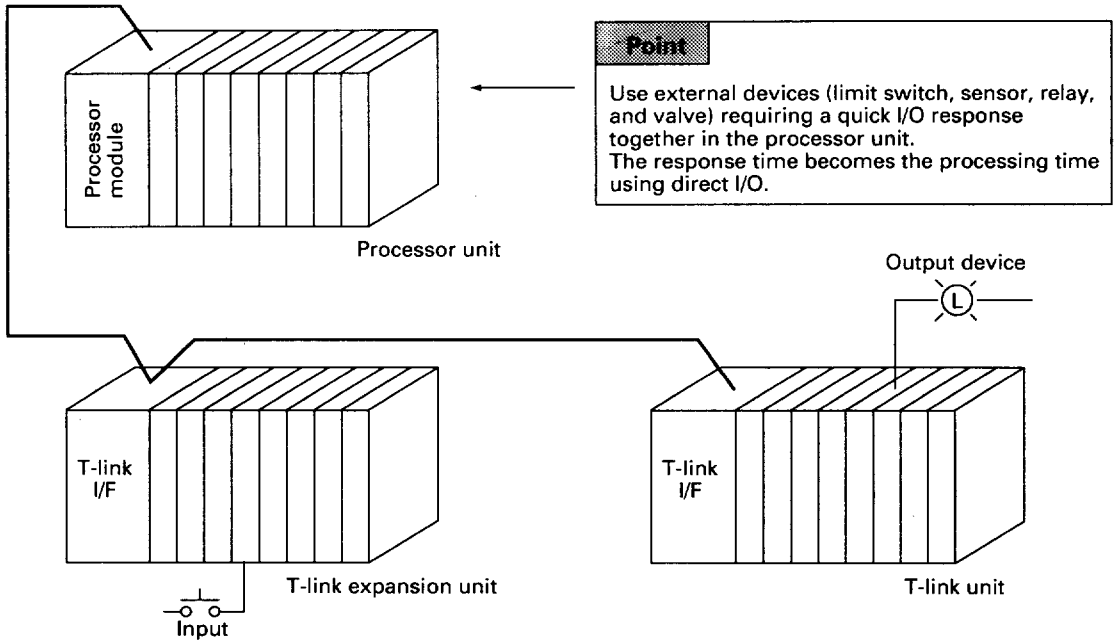
Note: Sampling delay + transmission delay + buffer delay + transmission delay = 40ms

Example of calculating I/O response time
Input capsule FTK113A: OFF→ON response time = 3ms
Output capsule FTK211A: OFF→ON response time = 1ms
Scan time: 20ms
- Minimum I/O response time = 3ms + 20ms + 1ms = 24ms
- Maximum I/O response time = 3ms + 20ms x 2 + 1ms + 40ms = 84ms

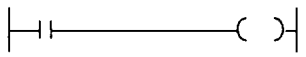
2-3 For T-link Expansion Unit

For the modules on the T-link expansion unit, in addition to the conditions in Section 2-2 on page A-12, the buffer delay for the T-link interface module must be considered.

Configuration example

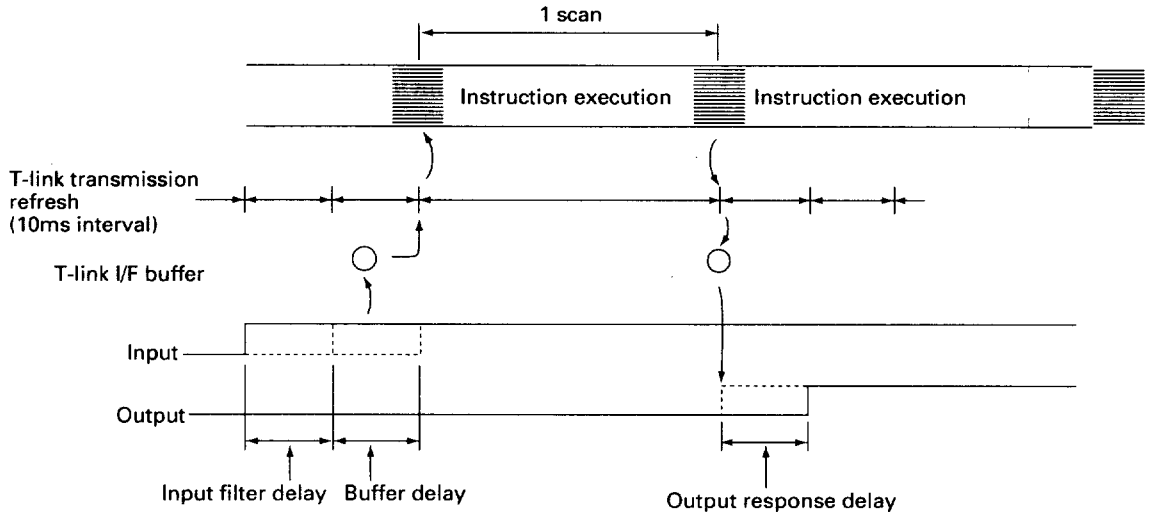


Circuit example



Input and output are input and output modules mounted on the T-link expansion unit.

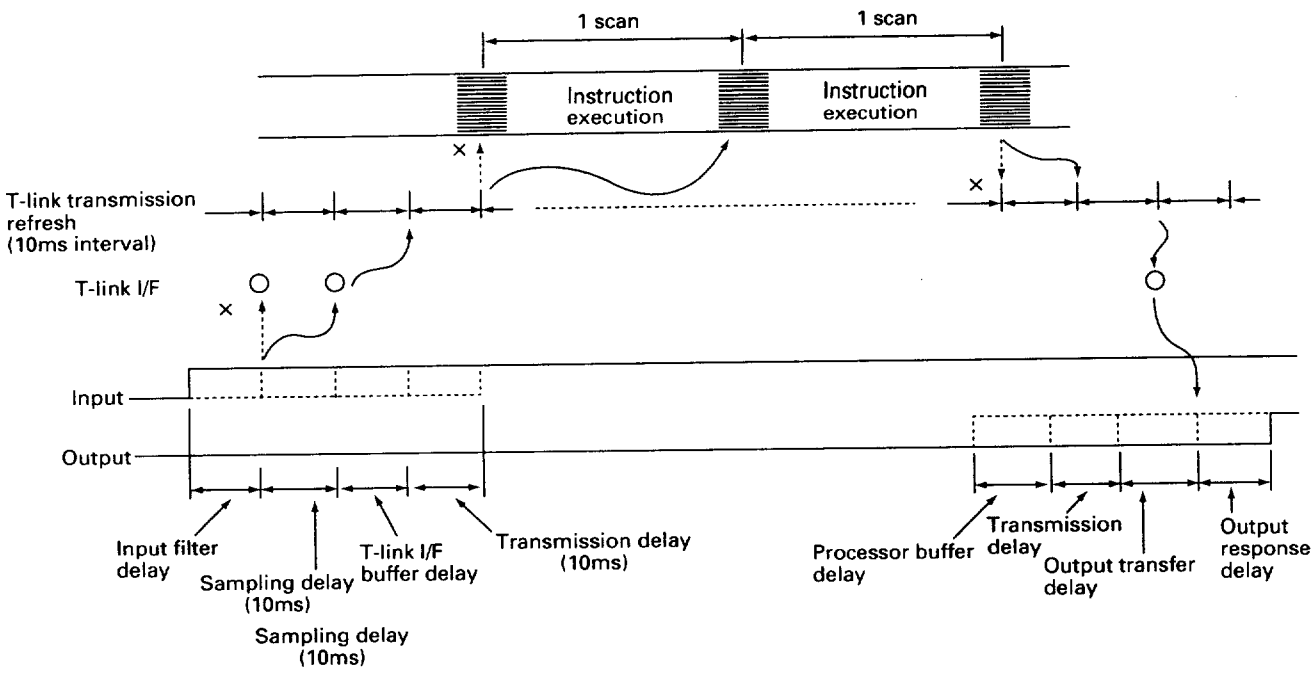
Minimum I/O response time



Minimum I/O response time = Input filter delay + Buffer delay + 1 scan time + Output response delay
= Input filter delay + 1 scan time + Output response delay + 10ms

Appendix 2 I/O Response

Maximum I/O response time



Maximum I/O response time
= Input filter delay + ***Sampling delay*** + ***T-link I/F buffer delay*** + ***Transmission delay*** + (1 scan time) x 2
(I/O module ⟶ T-link I/F) (T-link I/F ⟶ CPU)
+ ***Processor buffer delay*** + ***Transmission delay*** + ***Output transfer delay*** + Output response delay
(CPU ⟶ T-link I/F) (T-link I/F ⟶ I/O module)
= Input filter delay + (1 scan time) x 2 + Output response delay + ***60ms***

Example of calculating I/O response time

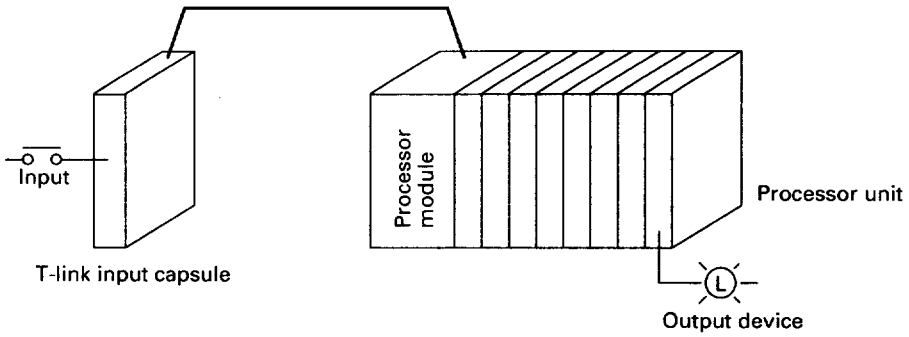
Input module NC1X1604:
OFF→ON response time = 3ms
Output module NC1Y16T01P2:
OFF→ON response time = 1ms

Scan time: 20ms (F70 series)
- Minimum I/O response time
  = 3ms + 20ms + 1ms + 10ms = 34ms
- Maximum I/O response time
  = 3ms + 20ms x 2 + 1ms + 60ms = 104ms

2-4 For basic unit processor unit and FTK capsules

An example of inputting data from the FTK input capsule and outputting data from the output module on the basic unit (processor unit) is shown below.

Configuration example



Circuit example

Input is the input capsule connected to the T-link and output is the output module mounted on the basic unit (processor unit).

Minimum I/O response time

Minimum I/O response time = ***Input filter delay*** + 1 scan time + ***Output response delay***

- ***Input filter delay*** — (Delay caused by input capsule)
- ***Output response delay*** — (Delay caused by output capsule)

Maximum I/O response time

Maximum I/O response time = Input filter delay + ***Sampling delay*** + ***Transmission delay***

- ***Sampling delay*** + ***Transmission delay*** — (Delay caused by input capsule)

+ (1 scan time) x 2 + ***Output response delay***

- ***Output response delay*** — (Delay caused by output module)

= Input filter delay + (1 scan time) x 2 + Output response delay + ***20ms***

Example of calculating I/O response time

Input capsule FTK113A:
OFF→ON response time = 3ms
Output module NC1Y16T01P2:
OFF→ON response time = 1ms
Scan time: 20ms

- Minimum I/O response time
  = 3ms + 20ms + 1ms = 24ms
- Maximum I/O response time
  = 3ms + 20ms x 2 + 1ms + 20ms = 64ms

■ **I/O expansion address correspondence**

Channel 0		Channel 1		Channel 2		Channel 3	
T-link Station No.	Expansion Address	T-link Station No.	Expansion Address	T-link Station No.	Expansion Address	T-link Station No.	Expansion Address
00	WXX.0000-	00	WXX.0512-	00	WXX.1024-	00	WXX.1536-
01	WXX.0016-	01	WXX.0528-	01	WXX.1040-	01	WXX.1552-
02	WXX.0032-	02	WXX.0544-	02	WXX.1056-	02	WXX.1568-
03	WXX.0048-	03	WXX.0560-	03	WXX.1072-	03	WXX.1584-
04	WXX.0064-	04	WXX.0576-	04	WXX.1088-	04	WXX.1600-
05	WXX.0080-	05	WXX.0592-	05	WXX.1104-	05	WXX.1616-
06	WXX.0096-	06	WXX.0608-	06	WXX.1120-	06	WXX.1632-
07	WXX.0112-	07	WXX.0624-	07	WXX.1136-	07	WXX.1648-
08	WXX.0128-	08	WXX.0640-	08	WXX.1152-	08	WXX.1664-
09	WXX.0144-	09	WXX.0656-	09	WXX.1168-	09	WXX.1680-
10	WXX.0160-	10	WXX.0672-	10	WXX.1184-	10	WXX.1696-
11	WXX.0176-	11	WXX.0688-	11	WXX.1200-	11	WXX.1712-
12	WXX.0192-	12	WXX.0704-	12	WXX.1216-	12	WXX.1728-
13	WXX.0208-	13	WXX.0720-	13	WXX.1232-	13	WXX.1744-
14	WXX.0224-	14	WXX.0736-	14	WXX.1248-	14	WXX.1760-
15	WXX.0240-	15	WXX.0752-	15	WXX.1264-	15	WXX.1776-
16	WXX.0256-	16	WXX.0768-	16	WXX.1280-	16	WXX.1792-
17	WXX.0272-	17	WXX.0784-	17	WXX.1296-	17	WXX.1808-
18	WXX.0288-	18	WXX.0800-	18	WXX.1312-	18	WXX.1824-
19	WXX.0304-	19	WXX.0816-	19	WXX.1328-	19	WXX.1840-
20	WXX.0320-	20	WXX.0832-	20	WXX.1344-	20	WXX.1856-
21	WXX.0336-	21	WXX.0848-	21	WXX.1360-	21	WXX.1872-
22	WXX.0352-	22	WXX.0864-	22	WXX.1376-	22	WXX.1888-
23	WXX.0368-	23	WXX.0880-	23	WXX.1392-	23	WXX.1904-
24	WXX.0384-	24	WXX.0896-	24	WXX.1408-	24	WXX.1920-
25	WXX.0400-	25	WXX.0912-	25	WXX.1424-	25	WXX.1936-
26	WXX.0416-	26	WXX.0928-	26	WXX.1440-	26	WXX.1952-
27	WXX.0432-	27	WXX.0944-	27	WXX.1456-	27	WXX.1968-
28	WXX.0448-	28	WXX.0960-	28	WXX.1472-	28	WXX.1984-
29	WXX.0464-	29	WXX.0976-	29	WXX.1488-	29	WXX.2000-
30	WXX.0480-	30	WXX.0992-	30	WXX.1504-	30	WXX.2016-
31	WXX.0496-	31	WXX.1008-	31	WXX.1520-	31	WXX.2032-

Fuji Electric Co., Ltd.

ED & C and Drive Systems Group

12-1, Yurakucho 1-chome, Chiyoda-ku, Tokyo, 100 Japan
Phone: 3-3211-1424
Fax: 3-3211-7983
Telex: J22331 FUJIELE A or FUJIELE B

Printed on recycled paper

Information in this manual is subject to change without notice.

Printed in Japan 1997-12/ F 5 FIS