

Sharp Programmable Controller

New Satellite JW300

Programiming Manual - Ladder Instruction Version



We thank you for your purchase of the SHARP programmable controller JW300. This manual (Programming Manual - Ladder Instruction Version) mainly describes software elements of the JW300 such as data memory, system memory, and instruction words. Carefully read this user's manual, hardware version and the instruction manual attached to each module so that you are able to operate JW300 properly, having thoroughly familiarized yourself with the functions of the system module and their operation method. As for the description about hardware elements such as system configuration, etc. of the JW300, see the "JW300 User's Manual - Hardware Version."

Precautions

- When you plan to use SHARP programmable controllers (hereafter referred to as "PLCs"), you are requested to design each system so that even if a fault or malfunction occurs within the PLC, it will not lead to a serious accident in your system. You should incorporate back-up measures and fail-safe features in your system that will thoroughly protect your system from malfunctions if a fault or error occurs in the PLC.
- SHARP PLCs are designed and manufactured with the idea that they will be used in general applications in ordinary industries. Therefore, they must not be used in specific applications that can affect the health or safety of the public, such as nuclear power plants and other power generating plants. Such applications require a special warranty of quality that SHARP explicitly does NOT offer for these PLCs. However, if a user will certify that he/she does not requires a special quality warranty on the PLC, and will limit the use of the PLC to non critical areas of these applications, SHARP will agree to such use.

If you are planning to use SHARP PLCs for applications that may affect the lives of human beings and property, and you need particularly high reliability performance, such as in the fields of aviation, medicine, transportation, combustion and fuel processing equipment, passenger cars, amusement park rides, and safety equipment, please contact our sales division so that we can confirm the required specifications.

Notes

- Though this manual is produced with the almost care, if you have any questions and inquiries, please feel free to contact our dealers.
- The whole or partial photocopy of this booklet is prohibited.
- Contents of this booklet may be revised for improvement without notice.

Chapter 1 Outline

Chapter 2 Data memory

Chapter 3 Program memory, Parameter memory

Chapter 4 System memory

Chapter 5 Operation of control module

Chapter 6 Block operation, logging, fault diagnostic, and PC card

Chapter 7 List of instructions

Chapter 8 Description of basic instructions

Chapter 9 Description of application instructions

Chapter 10 Application instructions (F-00 to Fx14d)

Chapter 11 Application instructions (F-15 to F-49)

Chapter 12 Application instructions (F-50 to F-79d)

Chapter 13 Application instructions (F-80 to F-173d)

Chapter 14 Application instructions (F-174 to F-403)

Table of contents

Chapter 1	Outline · · · · · · · · · · · · · · · · · · ·
Chapter 2 2-1 2-2 [1] [2] 2-3 [1] 2-4	Data memory · · · · · · · · · · · · · · · · · · ·
Chapter 3 3-1 [1] [2] 3-2 [1] [2]	Program memory, Parameter memory
Chapter 4	System memory · · · · · · · · · · · · · · · · · · ·
4-1 4-2	List of system memories 4-1 Contents of system memory 4-5
Chapter 5 5-1 [1] 5-2 [1] [2] [3] [4] [5]	Operation of control module
Chapter 6 6-1 [1] [2] [3] 6-2 [1] [2] [3] 6-3 [1] [2] [3] 6-4 [1] [2] [3]	 Block operation, logging, fault diagnostic, and PC card · · · · · · · · · 6-1 to 23 Block operation function 6-1 Set block status 6-3 Block start relay 6-4 I/O refresh 6-6 Logging function 6-7 Programming application instruction F-403 (LOG) 6-7 Setting system memory #2100 to #2165 6-8 Use examples 6-11 Fault diagnosis function 6-14 Faulty diagnosis principle 6-15 Error details of fault diagnosis 6-16 Setting details 6-18 How to use the PC card 6-19 Objective files for save and load 6-20 Setting system memory 6-21 Operation procedure to save and load to/from CF card 6-22
[4]	Error code 6-23

Chapter 7	List of instructions · · · · · · · · · · · · · · · · · · ·
7-1	List of basic instructions 7-1
7-2	List of application instructions 7-3
[1]	In order of numbers 7-3
[2]	Classification by operation 7-15
Chapter 8	Description of basic instructions · · · · · · · · · · · · · · · · · · ·
8-1	Calculation of basic instructions 8-1
8-2	Description of each basic instruction 8-2
Chapter 9	Description of application instructions · · · · · · · · · · · · · · · · · · ·
9-1	Application instruction hints and tips 9-1
[1]	Source and destination 9-1
[2]	When specifying an indirect address 9-12
9-2	Index modification function 9-17
[1]	Programming of index modification 9-17
[2]	Index modification application area 9-18
[3]	Program examples of index modification 9-19
Chapter 10	Application instructions (F-00 to Fx14d) 10-1 to 53
Chapter 11	Application instructions (F-15 to F-49) · · · · · · · · · · · · · · · · · · 11-1 to 48
Chapter 12	Application instructions (F-50 to F-79d) · · · · · · · · · · · · · · · · · · 12-1 to 50
Chapter 13	Application instructions (F-80 to F-173d) · · · · · · · · · · · · · · · · · 13-1 to 55
Chapter 14	Application instructions (F-174 to F-403) · · · · · · · · · · · · · · · · · · ·

Chapter 1 Outline

The New Satellite JW300 series are high-speed, high-performance programmable controllers for medium and large-scale control systems. These are high-level models of the JW30H series.

Features

(1) High-speed processing and large memory capacity

- High processing speeds of 33 ns for basic instructions, and 132 ns for application instructions (The overall processing speed will be approximately 20% faster than our conventional JW30H model.)
- Large, 256 K-word maximum capacity for program memory and 8 M-bytes maximum for file registers (both approximately 4 times larger than that of the JW30H).

(2) Compatible with memory cards

- Programs and parameters can be backed up on CF cards.
- Extensions to file memory, logging data, etc. can be stored on SRAM cards.

(3) Equipped with a USB port

The JW300 can exchange commands and data with PCs, at high speeds, through its USB port.

(4) Equipped with three ports for communication

There are two ports on the control module (one port on the JW-311CU/312CU) and one port on the I/O bus expansion adapter (JW-32EA), which can be used for communication. These make it easy to connect to a control terminal or image sensor camera.

(5) Structured programs / block operation

- You can separate the programs that run on the JW300 into a few blocks so that the PLC can operate various machines independently, for trial operations and other purposes.
- Each program block can be further separated into sub programs to save programming effort. This will make it possible to design some programs in parallel. These sub programs can be handled as standardized modules and can be reused.

(6) Built in faulty equipment diagnosis function

Just program relay numbers and monitor times and the PLC can monitor facilities. This feature makes for significant savings in writing ladder programs and for detecting errors.

(7) A variety of models

- Ten control module models are available. You can choose the one that best matches your system's control scale and budget.
- All of the I/O modules and special I/O modules available for the JW20H/30H series can be used with the JW300. Optional modules for the JW20H/30H also can be used if they are labeled as "compatible with the JW300."
- A Windows version of our ladder logic programming software, the JW-300SP, is available to support editing of structured programs.

The JW-15PG hand-held programmer is available to modify and monitor programs on site.

(8) Compatible with various open networks

The JW300 series is Ethernet compatible for communication, FL-net compatible for control, and DeviceNet compatible for the field. It is also AS-I compatible for sensor applications. These devices can exchange data with various layers, without any barriers.

Chapter 2 Data memory

2-1 File address

Data memory of the control module (JW-3**CU) varies with each mode.

	File address (capacity)				
JW-311CU JW-312CU	JW-321CU JW-322CU		JW-341CU JW-342CU	JW-352CU	JW-362CU
00000000(8) to 00073777(8) (30 K-bytes)	0000000(8) to 00105777(8) (35 K-bytes)		00000000 to 00177777 (64 K-byte	7(8)	
 Relay 30720 points (3.75 K-bytes) TMR/CNT contact points 1024 points (0.25 K-bytes) TMR/CNT/MD current value (2 K-bytes) Register (24 K-bytes) 	 Relay 53248 points (6.5 K-bytes) TMR/CNT contact points 2048 points (0.5 K-bytes) TMR/CNT/MD current value (4 K-bytes) Register (24 K-bytes) 	- Relay 180224 points (22 K-bytes) - TMR/CNT contact points 8192 points (2 K-bytes) - TMR/CNT/MD current value (16 K-bytes) - Register (24 K-bytes)			
File register ➡	0020000 ₍₈₎ to 00277777 ₍₈₎ (32 K-bytes)	to 00577777 ₍₈₎	to	0020000(8) to 10177777(8) (2048 K-bytes)	0020000(8) to 40177777(8) (8192 K-bytes)

• Memory map => See page 2-2 to 2-5.

Reference

- The JW30H has a file register from file 1 to 3, 10 to $2C_{(H)}$. 64K bytes of each file (16 K bytes for file 1) are independent. On the contrary, the JW300 has continuous addresses up to 8192 K-bytes. In addition, if a program of the JW30H is converted to one for the JW300 using the JW-300SP (ladder logic programming software), files 1 to 3 and 10 to $2C_{(H)}$ of the JW30H are converted to "file registers of the JW300."

2-2 Memory map

This section describes the memory map of the data memory in order of byte address (page 2-2 to 2-4) and in order of file address (page 2-5).

[1] In order of byte address

Byte address (capacity) of "relay, TMR/CNT contact points, TMR/CNT/MD current value" and "file registers" varies with model of control module (JW- 3^{**} CU). => (1) and (3).

Byte address of "registers" is common for all the models of control module. => (2).

(1) Byte address (file address) of relay, TMR/CNT contact points, TMR/CNT/MD current value.

• JW-311CU/312CU

Area	Byte address(8) (Relay number(8))	File address(8)	Capacity (bytes)
Relay *	⊐00000 to ⊐01577(000000 to 015777)	00000000 to 00001577	3840
(30720 points)	⊐02000 to ⊐07577(020000 to 075777)	00030000 to 00035577	(3.75 K)
TMR/CNT	T or C 00000 to 00777	00001600 to 00001777	256
contacts (1024 points)	T or C 01000 to 01777	00035600 to 00035777	(0.25 K)
TMR/CNT/MD	b00000 to b01777	00002000 to 00003777	2048
current value	b02000 to b03777	00026000 to 00027777	(2 K)

• JW-321CU/322CU

Area	Byte address(8) (Relay number(8))	File address(8)	Capacity (bytes)
*	⊐00000 to ⊐01577(000000 to 015777)	00000000 to 00001577	
Relay (53248 points)	⊐02000 to ⊐07577(020000 to 075777)	00030000 to 00035577	6656 (6.5 K)
	⊐10000 to ⊐15377(100000 to 153777)	00074000 to 00101377	(0.0 10)
TMR/CNT	T or C 00000 to 00777	00001600 to 00001777	
contact point (2048 points)	T or C 01000 to 01777	00035600 to 00035777	512 (0.5 K)
(2048 points)	T or C 02000 to 03777	00101400 to 00101777	
	b00000 to b01777	00002000 to 00003777	
TMR/CNT/MD	b02000 to b03777	00026000 to 00027777	4096 (4 K)
	b04000 to b07777	00102000 to 00105777	

● JW-331CU/332CU,JW-341CU/342CU,JW-352CU,JW362CU

Area	Byte address(8) (Relay number(8))	File address(8)	Capacity (bytes)
*	⊐00000 to ⊐01577(000000 to 015777)	00000000 to 00001577	
Relay (180224 points)	⊐02000 to ⊐07577(020000 to 075777)	00030000 to 00035577	22528 (22 K)
(100)	⊐10000 to ⊐54377(100000 to 543777)	00074000 to 00140377	
TMR/CNT	T or C 00000 to 00777	00001600 to 00001777	
contacts	T or C 01000 to 01777	00035600 to 00035777	2048 (2 K)
(8192 points)	T or C 02000 to 17777	00140400 to 00143777	(=)
	b00000 to b01777	00002000 to 00003777	
TMR/CNT/MD current value	b02000 to b03777	00026000 to 00027777	16384 (16 K)
	b04000 to b37777	00144000 to 00177777	

*In the relay areas, the following fixed areas are arranged.

Т	ype (inside relay area)	Byte address(8)	Relay number(8)	File address(8)	Capacity
(1)	Relay for I/O link [mode 7, 8]	⊐00100 to ⊐00177	001000 to 001777	00000100 to 00000177	512 points (64 bytes)
(2)	Special relay	⊐00730 to ⊐00737	007300 to 007377	00000730 to 00000737	64 points (8 bytes)
(3)	Relay for option module	⊐01000 to ⊐01477	010000 to 014777	00001000 to 00001477	2560 points (320 bytes)
(4)	Flag for option module	⊐01500 to ⊐01567	015000 to 015677	00001500 to 00001567	448 points (56 bytes)
(5)	Flag for I/O link	⊐01570 to ⊐01577	015700 to 015777	00001570 to 00001577	64 points (8 bytes)
(6)	Relay for I/O link [mode 1 to 6]	⊐02000 to ⊐02377	020000 to 023777	00030000 to 00030377	2048 points (256 bytes)
(7)	Relay 1 for special I/O module [basic system]	⊐03000 to ⊐03777	030000 to 037777	00031000 to 00031777	4096 points (512 bytes)
(8)	Relay for special I/O module [remote I/O slave station]	⊐04000 to ⊐04177	040000 to 041777	00032000 to 00032177	1024 points (128 bytes)
(9)	Relay 2 for special I/O module [basic system]	⊐04200 to ⊐05177	042000 to 051777	00032200 to 00033177	4096 points (512 bytes)

(2) Register byte address (file address): Common for all models

Byte address (capacity) of "register" is common for all the models (JW-3**CU).

Area	Byte address(8)	File address(8)	Capacity	(bytes)
	009000 to 009777	00004000 to 00004777	512	
	019000 to 019777	00005000 to 00005777	512	
	029000 to 029777	00006000 to 00006777	512	
	039000 to 039777	00007000 to 00007777	512	
	049000 to 049777	00010000 to 00010777	512	
	059000 to 059777	00011000 to 00011777	512	
	069000 to 069777	00012000 to 00012777	512	
	079000 to 079777	00013000 to 00013777	512	
	089000 to 089777	00014000 to 00014777	512	
	099000 to 099777	00015000 to 00015777	512	
	E0000 to E0777	00016000 to 00016777	512	
	E1000 to E1777	00017000 to 00017777	512	
	E2000 to E2777	00020000 to 00020777	512	
	E3000 to E3777	00021000 to 00021777	512	
	E4000 to E4777	00022000 to 00022777	512	
	E5000 to E5777	00023000 to 00023777	512	
	E6000 to E6777	00024000 to 00024777	512	
	E7000 to E7777	00025000 to 00025777		
	109000 to 109777	00036000 to 00036777		
	119000 to 119777	00037000 to 00037777		
	129000 to 129777	00040000 to 00040777	1	
	139000 to 139777	00041000 to 00041777		
	149000 to 149777	00042000 to 00042777		
Register	159000 to 159777	00043000 to 00043777		24K
U	169000 to 169777	00044000 to 00044777		
	179000 to 179777	00045000 to 00045777		
	189000 to 189777	00046000 to 00046777		
	199000 to 199777	00047000 to 00047777		
	209000 to 209777	00050000 to 00050777		
	219000 to 219777	00051000 to 00051777	512	
	229000 to 229777	00052000 to 00052777		
	239000 to 239777	00053000 to 00053777		
	249000 to 249777	00054000 to 00054777		
	259000 to 259777	00055000 to 00055777		
	269000 to 269777	00056000 to 00056777	512	
	279000 to 279777	00057000 to 00057777	512	
	289000 to 289777	00060000 to 00060777		
	299000 to 299777	00061000 to 00061777	512	
	309000 to 309777	00062000 to 00062777		
	319000 to 319777	00063000 to 00063777	512	
	329000 to 329777	00064000 to 00064777	512	
	339000 to 339777	00065000 to 00065777	512	
	349000 to 349777	00066000 to 00066777	512	
	359000 to 359777	00067000 to 00067777	512	
	369000 to 369777	00070000 to 0007777	512	
	379000 to 379777	00071000 to 00071777	512	
	389000 to 389777	00072000 to 00072777		
	Z000 to Z377 *		512	
		00073000 to 00073777	512	

* Z register is arranged in units of two bytes.

(3) Byte address (file address) of file register

• JW-311CU/312CU

These models do not have a file register.

• JW-321CU/322CU

Area	Byte address(8)	File address(8)	Capacity(bytes)
File register	00000000 to 00077777	00200000 to 00277777	32 K

• JW-331CU/332CU

Area	Byte address(8)	File address(8)	Capacity(bytes)
File register	00000000 to 00377777	00200000 to 00577777	128 K

• JW-341CU/342CU

Area	Area Byte address(8)		Capacity(bytes)	
File register	00000000 to 01777777	00200000 to 02177777	512 K	

• JW-352CU

Area	Byte address(8)	File address(8)	Capacity(bytes)	
File register	00000000 to 07777777	00200000 to 10177777	2048 K	

• JW-362CU

Area	Byte address(8)	File address(8)	Capacity(bytes)	
File register	00000000 to 37777777	00200000 to 40177777	8192 K	

[2] File address order arrangement

Data memory is arranged as follows in the order of file address.

File address(8)	JW-311CU/312CU	JW-321CU/322CU	JW-331CU/332CU JW-341CU/342CU JW-352CU JW-362CU	Byte address
00000000		Relay (7168 points))	⊐00000 ⊐01577
00001600		TMR/CNT contents of 00	000 to 00777 (512 points)	
00002000		Current value of TMR/CNT	/MD 00000 to 00777 (1024 bytes)	b00000
<u>00003777</u> 00004000		Register 009000 (5		<u>b01777</u> 009000
00015777 00016000		Register E0000 (4	to E7777 096 bytes)	<u>099777</u> E0000
00025777 00026000		Current value of TMR/CN		<u>E7777</u> b02000
00027777 00030000 00035577		Relay (23552 points		<u>b03777</u> ⊐02000 ⊐07577
00035600		TMR/CNT contents of 01	000 to 01777 (512 points)	
00036000		Register 019000 (!	to 199777 5120 bytes)	109000
00047777		Register 209000	to 299777 5120 bytes)	<u>199777</u> 209000
00061777 00062000		Register 309000 (4	to 389777 608 bytes)	<u>299777</u> 309000
00072777 00073000		Register Z000 to (512	Z377 bytes)	<u>389777</u> Z000
00073777		00074000		<u>Z377</u>
		Relay (22528 points) 00101377 ⊐15377 00101400 ⊂ontact of TMR/CNT 02000 to 03777(1024 points)	Relay (149504 points) 00140377 ⊐54377 00140400	
		00101777 00102000 b04000 Current value of TMR/CNT/MD 02000 to 03777 (2048 bytes) 00105777 b07777	Contents of TMR/CNT 02000 to 17777 (7168 points) 00143777 00144000 b04000	
* Δe fo	r the file register,		Current of TMR/CNT/MD 02000 to 17777 (14336 bytes)	
	age 2-4.	Eilo registor *	00177777 b37777	
		File register *	File register *	

2-3 Relay area

[1] Special relay

Sixty-four points of relay numbers 007300 to 007377 (100730 to 100737) are a special relay area.

- Use the special relays (except 007365) in the area written by a CPU. With the user program, use them as contact points and a source of application instructions.

These cannot be used as destination of the OUT and application instructions. Especially, be careful for instructions, distribution, extraction instructions, package data transfer instructions that handle data memory of two bytes or more.

- Do not use the reserved area (see the table below) for the user program.

Relay No.	Description							
007300								
to	Reserved area							
007327								
	MW flag							
007330	- This flag goes ON when the program memory is changed, and the MW lamp of the control module (JW-3**CU) blinks.							
	MW reset							
	- Use to reset the MW flag (007330).							
007331	While the MW flag is ON, turn the MW reset (007331) from OFF to ON; the MW flag goes OFF and the MW lamp of the control module goes OFF. At the same time, the MW reset goes OFF, too							
	same time, the MW reset goes OFF, too. In order to disable (turn OFF) the MW flag and MW lamp, always turn OFF the MW reset relay (007331) using the user program.							
007332								
to	Reserved area							
007337								
	Store error code							
007340	- As a result of the self-diagnosis, if an error is found, the JW300 store its error							
to 007347	code (page 5-2). - These are handled as byte address "⊐00734" of special register.							
007347	- When the JW300 recovers from the error status, the error code will be cleared							
007350								
to	Reserved area							
007353								
007354	Non carry flag							
007355	Error flag - Flag will be set according to the kind of operation when an							
007356	Carry flag application instruction is executed that may affect the flag.							
007357	Zero flag							
	0.1 second clock							
	- Used for the clock of the CNT instruction and other application instruction.							
007360	0.1 sec. (1 sec. clock => 007364)							
	007360 (0.1 second clock)							
007361	Reserved area							

To next page

Relay No.	Contents							
	Initialize pulse							
		e immediately after the run mode of the control						
	module is started. - Used to initially reset (initialize) a counter or shift register.							
	Count input							
	007362							
	Reset input							
	Shift direction input							
	Data input							
007362	SFR D000	10						
007302	Shift input 007362							
	Reset input							
		the input of rise operation command, it is necessary						
	to be used within the level operation							
	(Example)							
	ONLS							
	007362 F-01 10 000000							
	F-48 ONLR							
007363	- "ON" at fuse blown detection of JW-	262S (64-pointoutput module).						
	1-second clock							
	- Used for the clock of the CNT instru 1 sec.	ction and other application instruction.						
007364								
	(0.1 sec. clock => 007360)							
	007364 (1 second clock)							
	Setting value change switch							
		am with the support tool such as programmer to set						
	or reset a relay other than latched re 020000 to 075777).	elay area (initially conditions is 007000 to 015777,						
007365	- Program to be always ON							
	007366 007365							
		Use the normally OFF contact 007366.						
	Normally OFF contact							
007366		ed to be normally OFF (a-contact) or normally ON						
007267	(b-contact).							
007367 007370	Reserved area Memory error							
007370	CPU error							
007371	Battery error							
007372	I/O error	Self-diagnostic result						
007374	Option error	 If an error was met in a course of diagnosis, the relevant contact will be set ON. 						
007375	Special I/O module error flag	- For details, refer to page 5-2, "Self-diagnosis."						
007376	Expansion power supply error flag							
007377	Power source error							

2-4 TMR/CNT/MD data storage area

Registers having "b" in their top are areas to store current value of TMR/CNT and MD data of the MD instruction. The TMR/CNT/MD use two bytes per point. The TMR/CNT/MD numbers and "b****" are as follows.

	TMR/CNT/MD number	Data storage area]
ſ	00000	b00000, b00001	
	00001	b00002, b00003	
TMR/CNT/MD	00002	b00004, b00005	
in common	00003	b00006, b00007	
2	, ,	> >	JW-31*CU (1024 points)
L	00777	b01776, b01777	JW-32*CU
ſ	01000	b02000, b02001	(2048 points) JW-33*CU JW-34*CU
2	, , ,))	JW-352CU JW-362CU
	01777	b03776, b03777	(8192 points)
TMR/CNT in common	۶ : ۶	»	
	03777	b07776, b07777	
~		»	
Ĺ	17777	b37776, b37777	

If any of "b*****" is specified for data processing instruction (F-00 etc.), current value of the TMR/CNT can be used for calculation.

The TMR/CNT/MD have three formats.

Type 1	Type 2	Туре 3
_	TMR DTMR (BCD) UTMR (BCD)	DTMR (BIN) UTMR (BIN)
_	CNT DCNT (BCD) UCNT (BCD)	DCNT (BIN) UCNT (BIN)
MD	—	_

• Data format for type 1

	7	6	5	4	3	2	1	0	
		(x10 ¹)			(x10 ⁰)				
MD	8	4	2	1	8	4	2	1	n
		Input	inform	ation		(x1	0²)		.
	1	S1	S2	S₃	8	4	2	1	n+1

- Numerical data is handled in BCD.

- n and n+1 represent the order of addresses.

• Data format for type 2

The TMR and CNT have a setting value range of 0 to 7999. "D" means down or decrement : "U" means up or increment.

	'								_
	7	6	5	4	3	2	1	0	
	(X10 ⁰)			(X10 ⁻¹)					
TMR	8	4	2	1	8	4	2	1	n
	Reset	(X10 ²	²)		(X1	0 ¹)		 n+1
	*	4	2	1	8	4	2	1	11+1
		(X1	0 ¹)			(X1	0°)		n
CNT	8	4	2	1	8	4	2	1	
0.U	Reset	(X10 ³	3)		(X1	0²)		 n+1
	*	4	2	1	8	4	2	1	11+1
		(X10 ⁰)			(X10 ⁻¹)				n
DTMR	8	4	2	1	8	4	2	1	''
(BCD)		(X10 ²)			(X10 ¹)				 n+1
	*	4	2	1	8	4		1	11 + 1
		(X10 ⁰)			(X10 ⁻¹)				n
UTMR	8	4	2	1	8	4		1	''
(BCD)	Reset	(X10 ²	²)	(X10 ¹)				n+1
	*	4	2	1	8		2	1	+ .
		(X1	0 ¹)				0°)		n
DCNT	8		2	1	8		2	1	''
(BCD)	Reset	(X10 ³	³)		(X1	0²)		 n+1
	*	4	2	1	8	4	2	1	+ .
		(X1	0 ¹)			(X1	0°)		n
UCNT	8		2		8	4	2	1	l''
(BCD)	Reset	(X10 ³)		(X1	0²)		 n+1
	*	4	2	1	8	4	2	1]+ .

- Numerical data is handled in BCD.

- n and n+1 represent the order of addresses.

- The JW300 program distinguishes between

TMR and CNT, and between U and D.

Data format for type 3

The TMR and CNT have a set value range of 0 to 32767. "D" means down or decrement : "U" means up or increment.

•									
	7	6	5	4	3	2	1	0	
DTMR	27	2 ⁶	2 ⁵	24	2 ³	2 ²	2 ¹	2 ⁰	n
(BIN)	Reset *	2 ¹⁴	2 ¹³	2 ¹²	2 ¹¹	2 ¹⁰	2 ⁹	2 ⁸	n+1
UTMR	27	2 ⁶	2 ⁵	24	2 ³	2 ²	2 ¹	2 ⁰	n
(BIN)	Reset *	2 ¹⁴	2 ¹³	2 ¹²	211	2 ¹⁰	2 ⁹	2 ⁸	n+1
DCNT	27	2 ⁶	25	24	2 ³	2 ²	2 ¹	2 ⁰	n
(BIN)	Reset *	2 ¹⁴	2 ¹³	2 ¹²	211	2 ¹⁰	2 ⁹	2 ⁸	n+1
UCNT	27	2 ⁶	2 ⁵	24	2 ³	2 ²	2 ¹	2 ⁰	n
(BIN)	Reset *	2 ¹⁴	2 ¹³	2 ¹²	2 ¹¹	2 ¹⁰	2 ⁹	2 ⁸	n+1

- Numerical data is handled in BIN (binary).

* While the TMR/CNT are operating, these bits go "1" (ON). If they do not measure, this bit goes "1" even resetting.

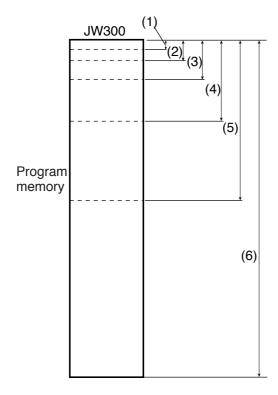
Chapter 3 Program memory, Parameter memory

3-1 Program memory

The program memory is the areas in which the user program is stored. As the JW300 begins to operate, the program is stated to read from the top address to do operation according to the program.

[1] Program addresses

In the JW300, the program capacity varies with model of control module (JW-3**CU).



(1) JW-311CU/312CU (8K words)

(2) JW-321CU/322CU (16K words)

(3) JW-331CU/332CU (32K words)

(4) JW-341CU/342CU (64K words)

- (5) JW-352CU (128K words)
- (6) JW-362CU (256K words)

⁻ The program addresses are as follows.

	Program memory	Program address					
Control module capacity		Octal	Decimal	Hexadecimal			
JW-311CU/312CU	8K words	000000 to 017777	000000 to 008191	00000 to 01FFF			
JW-321CU/322CU	16K words	000000 to 037777	000000 to 016383	00000 to 03FFF			
JW-331CU/332CU	32K words	000000 to 077777	000000 to 032767	00000 to 07FFF			
JW-341CU/342CU	64K words	000000 to 177777	000000 to 065535	00000 to 0FFFF			
JW-352CU	128K words	000000 to 377777	000000 to 131071	00000 to 1FFFF			
JW-362CU	256K words	000000 to 777777	000000 to 262143	00000 to 3FFFF			

There are 1-word, 2-word, 3-word and 4-word instructions.

	Instruction words
1-word instruction	STR, AND, etc.
2-word instruction	TMR, CNT, etc.
3-word instruction	F-00, F-01, etc.
4-word instruction	F-10, F-11, etc.

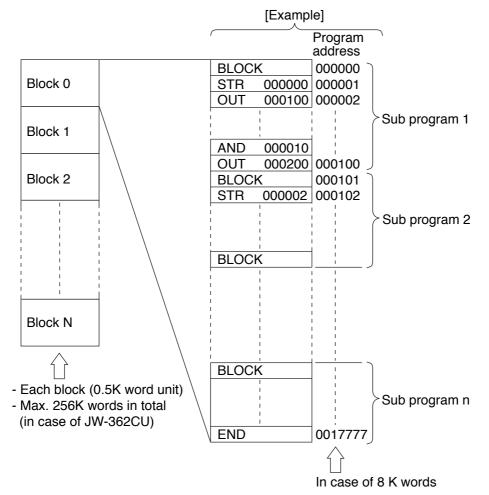
[2] Structuring program

The JW300 can store multiple programs (from now called "blocks") in one control module. The program capacity of each block is set in units of 0.5 K words using the JW-300SP (ladder logic programming software).

=> The last address of each block is written END instruction (F-40).

Each block can be programmed by separating it into multiple sub programs using the JW-300SP.

=> The top address of each sub program is written identification instruction (BLOCK).



The number of blocks and sub programs settable depend upon models of control module (JW-3**CU).

	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
No. of blocks (max.)	16	32	64	128	256	512
No. of sub programs(max.)	256	512	1024	2048	4096	8192

As for block operation, see "6-1 Block operation function."

Remarks

- The JW-15PG (hand-held programmer) cannot be used for separation into blocks and sub programs. It only can edit blocks and sub programs after they are separated using the JW-300SP.
- If a memory is cleared in units of block, the top address of a block is overwritten by BLOCK, the last address is written END instruction. Other addresses are written NOP instruction.

3-2 Parameter memory

This section describes parameter areas for special I/O modules and option modules on the JW300 system.

[1] Parameter for special I/O module

For the models of the special I/O modules listed below, set operation conditions to the special I/O parameter of the control module (JW-3**CU) using a support tools (JW-15PG, JW-300SP).

- JW-21HC, JW-22HC, JW-24AD, JW-22DA, JW-21PS, JW-21SU,
- The special I/O parameter area is determined by the special I/O module number switch (set value 0 to 7). One module uses 256 bytes.
- As for the details of the special I/O parameters, see respective user's manual of the special I/O module (JW-21HC etc.).

Setting value of module		Special I/O parameter address(8)													
No. switch	F	Rack 0		Rack 1		Rack 2	Rack 3								
0	T-00	000 to 377	T-10	000 to 377	T-20	000 to 377	T-30	000 to 377							
1	T-01	000 to 377	T-11	000 to 377	T-21	000 to 377	T-31	000 to 377							
2	T-02	000 to 377	T-12	000 to 377	T-22	000 to 377	T-32	000 to 377							
3	T-03	000 to 377	T-13	000 to 377	T-23	000 to 377	T-33	000 to 377							
4	T-04	000 to 377	T-14	000 to 377	T-24	000 to 377	T-34	000 to 377							
5	T-05	000 to 377	T-15	000 to 377	T-25	000 to 377	T-35	000 to 377							
6	T-06	000 to 377	T-16	000 to 377	T-26	000 to 377	T-36	000 to 377							
7	T-07	000 to 377	T-17	000 to 377	T-27	000 to 377	T-37	000 to 377							

Setting value		Special I/O parameter adddress(8)														
of module No. switch			Rack 5		Rack 6			Rack 7			Remote I/O slave station					
0	T-40	000 to	377	T-50	000 to	377	T-60	000	to	377	T-70	000	to 377	T-80	000	to 377
1	T-41	000 to	377	T-51	000 to	377	T-61	000	to	377	T-71	000	to 377	T-81	000	to 377
2	T-42	000 to	377	T-52	000 to	377	T-62	000	to	377	T-72	000	to 377	T-82	000	to 377
3	T-43	000 to	377	T-53	000 to	377	T-63	000	to	377	T-73	000	to 377	T-83	000	to 377
4	T-44	000 to	377	T-54	000 to	377	T-64	000	to	377	T-74	000	to 377	T-84	000	to 377
5	T-45	000 to	377	T-55	000 to	377 ס	T-65	000	to	377	T-75	000	to 377	T-85	000	to 377
6	T-46	000 to	377	T-56	000 to	377	T-66	000	to	377	T-76	000	to 377	T-86	000	to 377
7	T-47	000 to	377	T-57	000 to	377	T-67	000	to	377	T-77	000	to 377	T-87	000	to 377

Reference

- With the JW30H, the special I/O modules cannot be inserted to rack 4 to 7.

The special I/O parameter area uses 128 bytes per module.

[2] Parameter for option module

For the option module of the following models, specify operation conditions for them to the control modules (JW-3**CU) option parameter using a support tool (JW-15PG, JW-300SP).

● JW-21CM, JW-22SU, JW-20FL5/20FLT, JW-22FL5/22FLT,

JW-255CM, JW-25TCM, JW-22CM, JW-21MN

- Using the option module No. switch (set value 0 to 7), determine option parameter. This setting occupies 2 K bytes per one module.
- As for the detail of the option paramter, see each user's manual of the option module (JW-21CM etc.).

Setting value of module No. switch	Address of option parameter(8)
0	00000 to 03777
1	00000 to 03777
2	00000 to 03777
3	00000 to 03777
4	00000 to 03777
5	00000 to 03777
6	00000 to 03777
7	00000 to 03777

Reference

- With the JW30H, set parameters of the JW-255CM, JW-25TCM, JW-22CM, and JW-21MN to each module. The option parameter area uses 64 bytes per module.

Chapter 4 System memory

The system memory is used to set each function of the JW300 and monitor error detail of the JW300.

4-1 List of system memories

The system memory has a capacity of 1.5K-byte which occupy an address area from #0000 to #2777. It has battery back-up. This section describes memory numbers that are released to use for users. Other memory numbers are reserved region, and data should not be written in.

System memory address(8)	Initial value(H)		Contents	See page	* JW30H					
#0010		Seconds								
#0011		Minutes								
#0012		Hours								
#0013	Calendar clock	Date	ate Monitor the clock (BCD)							
#0014		Month	4-5							
#0015		Year								
#0016		Day of week								
#0017	00	Control								
#0030		Lower byte	Monitor the minimum value of scan time							
#0031		Upper byte	(BCD)							
#0032	00	Lower byte	Monitor the current value of every scan time	4-6						
#0033		Upper byte	(BCD)	4-0						
#0034		Lower byte Monitor the maximum value of scan time								
#0035		Upper byte								
#0046	00	Monitor the	4-7							
#0050		Lower byte		4-8	0					
#0051	00		Monitor the program's error address							
#0052			(OCT)	+0						
#0053		Upper byte								
#0054										
#0055	00	User progra	er program source sum check code							
#0056										
#0057				_						
#0060	00		am source sum check error or = 01(H), Normal = 00(H)	_						
#0064	—	Monitor boo	ot program version (sub version)							
#0065	—	Monitor boo	Monitor boot program version							
#0066			1							
#0067	_	Model code								
#0150	00	Monitor sw occurred								
#0152	00	Monitor sw occurred	itch setting of a device net module in which an error	4-9						

*O: Same function as the JW30H

 Δ : Changed function

×: JW30H does not have this function

System memory address _®	Initial value(H)	Contents	See page	* JW30H				
#0160								
#0161								
#0162								
#0163	00	Free and a regulting from solf diagnosis						
#0164	00	Error codes resulting from self-diagnosis.						
#0165								
#0166								
#0167								
#0170			7					
#0171			4-10					
#0172								
#0173								
#0174	00	Error codes of option module.		0				
#0175								
#0176								
#0178								
#0201		Sat the reporting condition of TMD	-					
	00	Set the resetting condition of TMR	-					
#0202		Set the resetting condition of CNT	-					
#0206	00	Set the operation continue/stop at fuse blown detection		-				
#0207	00	Set the operation continue/stop at option error	-					
#0211	00	Select whether to continue or stop operation when a device net module has an error.						
#0213	00	Set the top address number of comment memory using area	4-11					
#0222	02	Set the communication method used by the PG/COMM 2 port.						
#0223	00	Select the clock feature						
#0225	00	Set the 1ms timer function	4-12	×				
#0226	00	Set the constant scan time						
#0227	00	Set the 10 ms timer function						
#0230	C0	Lower byte Setting the range of a latched relay						
#0231	01	Upper byte (OCT)	4-13					
#0232		Lower byte Setting output hold address		1				
#0233	00	Upper byte (OCT)	4-14					
#0234	00		1					
#0235	00	Set the PG/COMM1 port		0				
#0236	00		4-15					
#0237	00	Set the PG/COMM2 port						
#0240	00			1				
#0241	FF							
#0242	00							
#0243	00	Set interrupt process	4-16					
#0243	00							
#0244	00			×				
	00 0A	Set extend for allowable voltage interruption time						
#0246			4-18	 				
#0247	00	Set the I/O addresses registration method						
#0250	00	Lower byte Set the range of a latched relay	4-13	0				
#0251	30	Upper byte (expansion area)		-				

System memory address	Initial value(H)		Contents	See page	* JW30H				
#0252	Value (н) 00	Lower byte Set	output holding address	page	5113011				
#0253	30	Upper byte	(expansion area)	4-14	0				
#0255	00	Set the ROM operation	, , , , , , , , , , , , , , , , , , ,		Ť				
#0256	80	Set the ROM storage	area	4-19	Δ				
#0266	00								
#0267	00	Set EA-PG port.							
#0270	FF		4.00	×					
#0271	FF	Transfer register of sy	stem memory #0000 to #2777.	4-20					
#0272	00		-						
#0277	No fixed	BCC check code			Δ				
#0300									
2	00 ост	Set device net JW-20	4-22	0					
#0377									
#0410		Specify symbol/comp							
	—	Specify symbol/comn	4-21						
#0414		(When using the ladd							
#0500									
2	00 ост	Set device net JW-20	4-22	0					
#0577									
#1600	00	Error code	The number of errors that occurred for						
#1601	30	The number of errors that occurred	each error code: No. 1						
#1602	00	Error code							
#1603	80	The number of errors that occurred	The number of errors that occurred for each error code: No. 2						
#1604			The number of errors that occurred for each error code: No. 3	4-24	×				
	00 ост								
#1675			The number of errors that occurred for each error code: No. 31						
#1676	00	Error code							
#1677	00	The number of	The number of errors that occurred for each error code: No. 32						
#10//	00	errors that occurred							

System memory address(8)	Initial value(H)	Contents	See page	* JW30H				
#2100								
#2101		— /						
#2102	00	Top address of an area that stores logging data (File address)						
#2103								
#2104	00	Number of logging data storing						
#2105	00	Time stamp format						
#2106	FF							
#2107	FF	Register address of logging specified register 1 (File address)						
#2110	FF	register address of logging specified register 1 (rile address)						
#2111	FF		4.05					
#2112	FF		4-25					
#2113	FF	Register address of logging specified register 2 (File address)						
#2114	FF	register address of logging specified register 2 (rile address)						
#2115	FF							
#2116		Register address of logging specified register 3 (File address)						
\langle	FF each	\geq						
#2151		Register address of logging specified register 9 (File address)						
#2152	FF							
#2153	FF	Register address of logging specified register 10 (File address)						
#2154	FF							
#2155	FF							
#2156				×				
2	00 each	Storage counter of next logging data						
#2161								
#2162								
2	00 each	The number of storages of logging data						
#2165								
#2220	00		4-26					
#2221	1F							
#2222								
2	00 each							
#2226		Set PC card						
#2230								
	00 each	ach						
#2236								
#2200	00	Set fault diagnosis (Enable = $01_{(H)}$, Disable = Other than $01_{(H)}$).	6-14					
#2300								
	00 each	Set device net JW-20DN (module No. 2)						
#2377			4.00					
#2400			4-23					
	00 each	each Set device net JW-20DN (module No. 3)						
#2 ⁴⁷⁷								

4-2 Contents of system memory

This section describes details of each item that is listed on the system memory table (page 4-1 to 4-4).

System memory No. (OCT)	Setting item					C	onten	ts					
#0010		Seco	onds: 0	0 to 59	(BCD)								
#0011		Minu	tes: 00	0 to 59	(BCD)								
#0012		Hours: 00 to 23 (BCD)											
#0010		Date: 01 to 31 (BCD)											
#0013				ntrol m			-	-	uishes	s 30-	-day r	nonths	
#0014				1-day n		and lea	ap yea	rs.					
#0015			: 0 Year is year. L	.eap ye	(BCD) ented ars are	by the assun	ned to	occur e	every 4	4 ye	•	n calenc	dar
#0016	Clock feature	('96, '00, and '04 are identified as leap years.) Day-of-week : 00 to 06 (BCD) Set the day of the week when you adjust the present time. It changes from 00 to 06 each time date data is incremented. Day- of-week data is not computed from the year/month/date setting. Day of week SUN MON TUE WED THU FRI SAT											
		BCD value 00 01 02 03 04 05 06 Control : Turn ON/OFF bits (D0, D3, and D7) for time adjustre monitor, 30 minutes correction, and stop/run clock.								ment/tim	ne		
		Сог	ntents	D 7	D6	D 5	D4	D3	D	2	D 1	D0	
#0017			ON DFF	Time adjust Time monitor	Not used.		30-sec. correction No		lot u	sed.	Stop clock Start clock		
		 - 30-sec. correction If 010(8) is set, 0 to 29 sec. is reset to 0 sec. ; 30 to 59 sec. is reset to 0 sec. with a carry to the minutes digit. - Time adjustment Set 001(8) to stop the clock, then write the present time to #0010 to #0016. Next, write 200(8) to set the time into the clock, whereupon bits Do and D7 are reset and the clock starts working.											
		No	#0	setting 016 and ck is st	d regist	ers 09	9770 to	0997	76 is r	not i	ipdate		g .

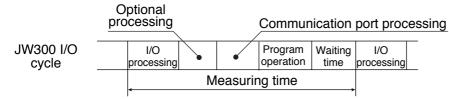
- When the value of system memory location #0223 is 00(H), time setting can be done from the JW300 program by using registers 099770 to 099777.

- The clock feature is backed up by battery.

#0030 #0031	Monitor scan time minimum value	Scan time minimum value is stored in a BCD value. [Example]: If the BCD value monitoring was 0020, the scan time minimum value is 20 ms. <u>00</u> <u>20</u> Monitored with #0030 (Lower digit) Monitored with #0031 (Upper digit)				
#0032 #0033	Monitor current value of each scan time	Scan time current value is stored in a BCD value. [Example]: If the BCD value monitoring was 0050, the current scan time is 50 ms. <u>00</u> 50 Monitored with #0032 (Lower digit) Monitored with #0033 (Upper digit)				
#0034 #0035	Monitor scan time maximum value	Scan time maximum value is stored in a BCD number. [Example]: If the BCD value monitoring was 0100, the scan time maximum value is 100 ms. <u>01</u> <u>00</u> Monitored with #0034 (Lower digit) Monitored with #0035 (Upper digit)				

The scan time measurement starts when the power is turned ON.
 When the operation is changed from run to stop (program mode), the latest scan time minimum and maximum values are stored. When the operation is changed from stop to run, the existing minimum and maximum value are cleared and newly detected values are stored.
 A scan error allowance is ±1 ms.

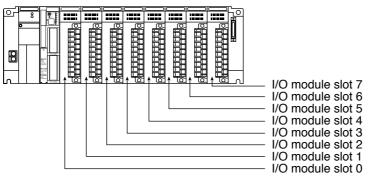
- Measurement of PC scan time is started at the beginning of I/O processing.



- The scan time is the total of the execution time from program block 0 to block N. Execution time of the waiting block is treated as 0.

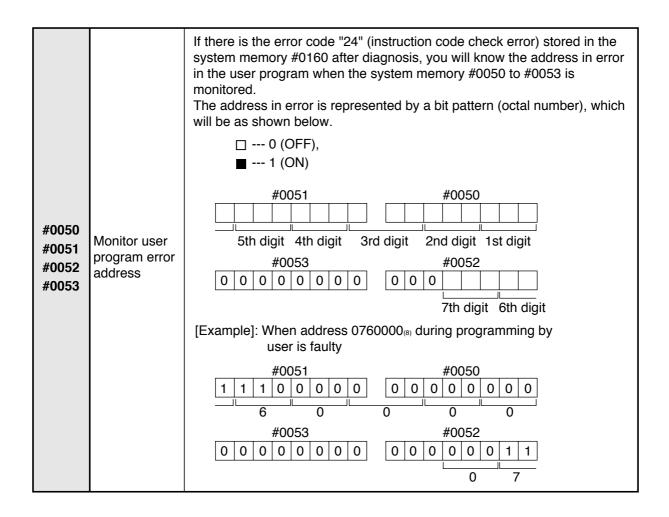
		The rack and slot errors listed below c	an be menitered. The reak							
		number and slot number where the error								
		PC I/O processing. As reference data in determining the error module.								
		 I/O data bus error Output data check error Installed module error I/O rack panel error 	Error code (BCD) 44 42 40 48							
		- Table verify error	60							
		- Module No. switch verify error	r 61 70							
		- Table registration error - Missing module error	70 71							
	Monitor the I/O	- I/O point overflow error	72							
#0046	slot number where an error	- Module No. switch set error	73							
	is detected									
		Bit arrangement of rack number, slot number								
		$Bit \rightarrow 6 5 4 3 2$	1 0							
		#0046 4 2 1 0 4	2 1							
			ot No. to 7)							
		· · · · · · · · · · · · · · · · · · ·	basic rack panel. For the expansion rack							
		Therefore, when using the I/O bus ex	xpansion adapter, the rack number is							
		•	no. Switch of expansion rack panel (JW-							
		34ZB/36ZB/38ZB) and I/O bus expansion adapter (JW-32EA). - For the slot number, the number of the I/O module slot in the basic/expansion rack panel (0 to 7) is stored.								

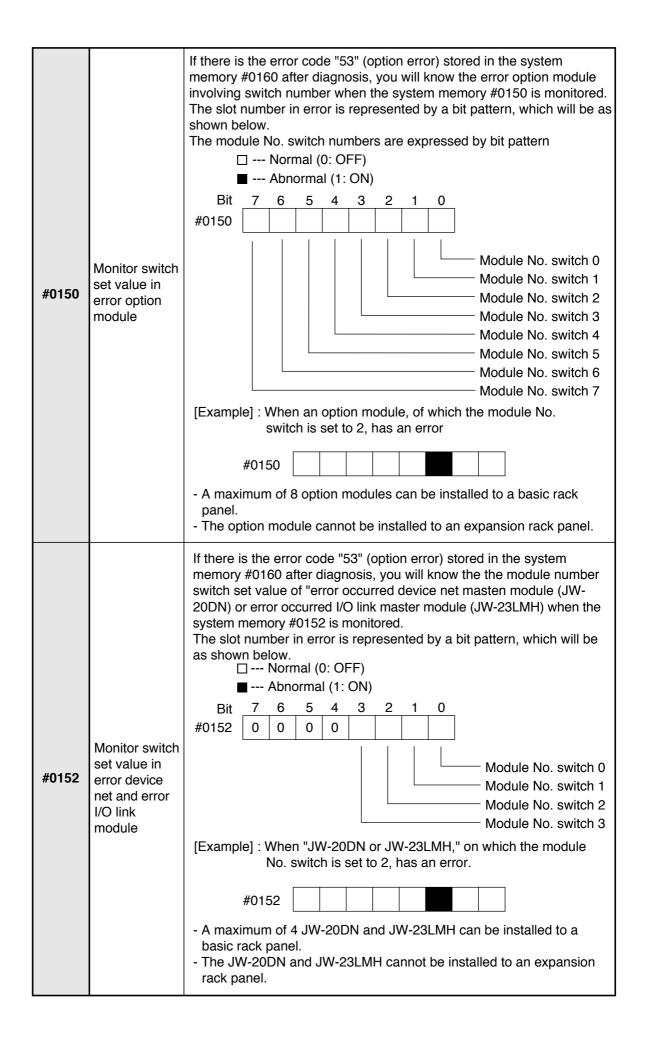
[Example] Basic rack panel: JW-318KB



Expansion rack panel: JW-38ZB

₽ ₽	[]]			······································	
						 I/O module slot 7 I/O module slot 6 I/O module slot 5 I/O module slot 4 I/O module slot 3 I/O module slot 2 I/O module slot 1 I/O module slot 0





#0160 to #0167	Diagnostic error code	 The error code will be stored when an error is encountered as a result of diagnosis. #0160 to #0167 functions as a shift register which will be able to store 8 errors. For details, refer to "Self-diagnosis." Error information including the time of error generation is stored in E7600 to E7777. Error codes remain in system memory after the cause of the error is removed. To clear the error code, write "00(H)" from support tool.
----------------------	--------------------------	---

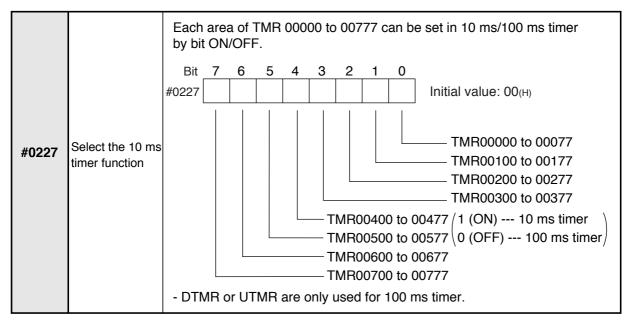
#0170 to #0177	Option module's error code	 If an error is detected by self-diagnosis on option module, the corresponding error code is stored in these locations. #0170 to #0177 function as a shift registers and can memorize occurrences of 8 errors. See each user's manual for the option module (JW-21CM etc.) for details on error code. Information which includes the time of the error occurrence is stored in E5600 to E7577 for each setting of the module No. switch (0 to 7) of the option module. All option module share system memory locations #0170 to #0177. Error code remain in system memory after the cause of the error is removed. To clear the error code, write 00(H) from support tool.
----------------------	-------------------------------	---

#0201	TMR reset condition	 Used to program the state of the TMR instruction upon power recovery. 00_(H) Reset upon power recovery. (initial condition) 01_(H) Retains the state at power down. The TMR instruction includes the DTMR (BCD), DTMR (BIN), UTMR (BCD), and UTMR (BIN) instructions.
#0202	CNT reset condition	Used to program the reset input condition for CNT instruction, application instruction (as follow). 00 _(H) Reset when ON. (initial condition) 01 _(H) Retains when OFF. - The CNT instruction includes the DCNT (BCD), DCNT (BIN), UCNT (BCD), and UCNT (BIN) instructions. - The following application instructions are available : F-60 (F/B SFR) F-160 (NSFR) F-60w (F/B SFR) F-160 (NSFR) F-60d (F/B SFR) F-261 (RCNT) F-62 (U/DC) F-62w (U/DC) F-62d (U/DC)

#0206	Setting the operation continuation/ halt at fuse blown detection	To set to continue or stop operation of the JW300 when fuse blow (failure code 49) is detected in the output module (JW-262S). 00 _(H) Operation continuation (initial setting) 08 _(H) Stop operation
-------	--	---

#0207	Set the operation continuation/ stop operation at option module error	To set to continue or stop operation of the JW300 when module error (failure code 53) is detected in the option module. When the bit of the module No. switch (0 to 7) of option module is turned ON, the operation continues. Bit 7 6 5 4 3 2 1 0 \square Stop operation (0: OFF) #0207 \square \square \square \square Stop operation (0: OFF) $\#$ 0207 \square \square \square \square \square Operation continuation (1: ON) 7 6 5 4 3 2 1 0 \leftarrow Module No. switch set value of option module Initial state is stop state(\square) in 00(H) for all bit.		
#0211	Set the operation continuation/ stop operation at device net, I/O link module error	To set to continue or stop operation of the JW300 when module error (failure code 53) is detected in the device net master module (JW-20DN), I/O link master module (JW-23LMH). When the bit of the module No. switch set value of the JW-20DN or JW-23LMH is turned ON, the operation continues. Bit 7 6 5 4 3 2 1 0 #0211 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0		
#0213	Select error storage area	Select register area to store error codes. 00 _(H) Not used 02 _(H) Register E5600 to E7777 ₍₈₎ (Initial setting)		
	Set the communi- cation method for PG/COMM2	Sets the communication method used by the PG/COMM 2 port.		
		Set value Communication method		
		00(H) RS-232C (RTS signal is ON while power is present) RS-422A (only 4-line systems at 1:1 allowed)		
#0222		 RS-232C connection only (RTS signal is OFF while sending data, and ON while not sending data) O2(H) - Use this setting if you want to control the data flow using the RTS signal. Note: The support tool cannot be used. 		
		04 _(H) RS-422A connection only (4-line system with 1: N connections is possible)		
#0223	Selects clock feature	This byte allows to control the clock feature on register. It enables time adjustment from the PC program. When "Use register" is selected, you can adjust time using the JW300's program. 00(H) Used register (initial state) 01(H) Not used Registers 099770 to 099777 are used for clock control.		

#0225	Select the 1 ms timer function	Each area of TMR 01770 to 01777 can be set in 1 ms/100 ms timer by bit ON/OFF switching. Bit 7 6 5 4 3 2 1 0 #0225 0 0 0 0 0 0 0 0 0 Initial value: 00(H)
#0226	Constant scan	JW300's scan time can set at your disposal. Setting is 01 to 99 ms in the BCD value. <u>#0226 set value</u> JW300 scan time <u>00 (BCD)</u> Minimum scan <u>01 to 99 (BCD)</u> Scan time of 01 to 99 ms Initial value is 00 (BCD). - If the program operation takes a long time than setting of scan time, the JW300 scan time is determined by the program operation time. - The JW300 scan time measures the time simultaneously with start of input and output processing. JW300 JW300 JW300 JW300 JW300 JW300 JW300 JW300 JW300 JW300 JW300 JW300 JW300



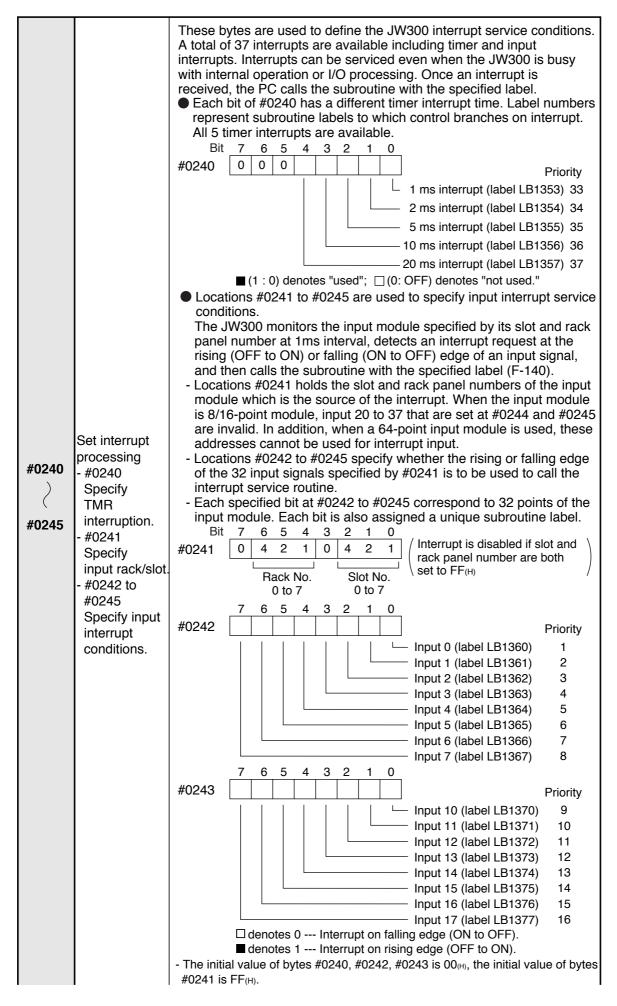
		Used to increase	e/decrease the lat	tched relay area fr	om the initial state
		(below). Set with 8 points as unit. Set numeric values by setting file address with octal.			
			Setting area (word, octal)	Byte address	Relay No.
		#0230, #0231	00000000 to 00001577	⊐00000 to ⊐01577	000000 to 015777
#0230	Set latched	#0250, #0251	00030000 to00035577	⊐02000 to ⊐07577	020000 to 075777
#0231	relay area	the final addres	ses (⊐01577, ⊐07! 77 (relay number	from the assigned 577). 100000 to 543777	-
		[Example] To set ⊐00200 and after (relay number 002000 to 015777) to the latched relay.			
			31 0 0 0 0 0 0 0 0 0 2 \$ 00000200 (Byte ad		#0230 = 020(8) #0231 = 000(8)
	Set latched	Initial value	of #0230 #0231		
		 Initial value of #0230, #0231 Area from ⊐00700 and after (relay No. 007000 to 015777) are 			
#0250		#02	latched relay. 31 0 0 0 1 1 1 1 0 7	#0230 0 0 0 0 0 0 	(Initial value) #0230 = 300(8) #0231 = 001(8)
#0250 #0251	relay area (expansion	File address 00000700 (Byte address ⊐00700)			
	relay area)	Area from ⊐0	,	•	075777, 100000
		#02	$\begin{array}{c} \text{re specified as lat} \\ 51 \\ \hline 0 0 0 0 \\ \hline 0 \\ \hline 0 \\ \hline 0 \\ \hline \end{array} \\ \begin{array}{c} 0 \\ \end{array} \\ \hline \end{array} \\ \begin{array}{c} 0 \\ \end{array} \\ \end{array} \\ \begin{array}{c} 0 \\ \end{array} \\ \begin{array}{c} 0 \\ \end{array} \\ \end{array} \\ \end{array} \\ \begin{array}{c} 0 \\ \end{array} \\ \end{array} \\ \end{array} \\ \begin{array}{c} 0 \\ \end{array} \\$	#0252	(Initial value) #0250 = 000(8) #0251 = 060(8)
		File address	s 00030000 (Byte ad	ddress: ⊐02000)	

	[Lload to get the ten address for the sutput relay which employed to
		Used to set the top address for the output relay which employed to retain the output module's output when the control module stops.
		Set with 8 points as unit. Set numeric values by setting file address
		with octal.
		Setting area (word actal) Byte address Relay No.
		(word, octal)
		#0232, #0233 00000000 to 00001577 ⊐00000 to ⊐01577 000000 to 015777 #0252, #0253 00030000 to 00035577 ⊐02000 to ⊐07577 020000 to 075777
		The output hold area occupies from the assigned byte address to the
		final addresses ($\exists 01577, \exists 07577$).
		Addresses 10000 to 154377 (relay No. 100000 to 543777) are
		always output hold area.
	Sat autout	[Example] To assign it to the output hold following ⊐00020 (Relay No. 000200 to 015777)
#0232	Set output retention	#0233 #0232
#0233	address	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
		$[\rightarrow 0 \ 0 \ 0 \ 0 \ 2 \ 0 \ [\#0233 = 000(8)]$
		File address 00000020 (Byte address: ⊐00020)
		- The output turns OFF for the area the output retention is canceled as
		the latch of the output module is reset when the JW300 is switched to the program mode or the JW300 stopped its operation as a result of
		self-diagnosis.
		000100 000011
		(Example)
		Monitor mode Program mode Monitor mode Program mode Monitor mode
		Run Stop Run Stop Run
		Input
		Output $\begin{cases} (a) \\ 000011 \end{cases}$
		(a) When 000011 is within the output retention area.
	Set output latch address (expansion relay area)	(b) When 000011 is within the output retention canceled area.
		- When the JW300 stops because of an error detected as a result of
		self-diagnosis, it may not turn OFF the output module within the area that the output retention is canceled, depending on the nature of the
		error. Output that needs to turn OFF when the JW300 meets the error
		should be connected serial to the power supply module halt output.
#0252		Initial value
#0253		Initial value of #0232, #0233 Area from 500000 to 015777 and 1000000 to 015777 and 1000000 to 015777 and 10000000 to 015777 and 100000000 to 015777 and 100000000 to 015777 and 10000000 to 015777 and 1000000000000000000000000000000000000
		Area from 300000 and after (relay No. 000000 to 015777) are
		specified as latched relay. #0233 #0232 [(Initial value]
		$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 &$
		$[\rightarrow 0 0 0 0 0 0 0 0 0 0$
		File address 00000000 (Byte address ⊐00000)
		●Initial value of #0252, #0253
		Area from ⊐02000 and after (relay No. 020000 to 075777) are
		specified as latched relay.
		#0253 #0252 (Initial value)
		#0253 = 060(8)
		└ File address 00030000 (Byte address: ⊐02000)

#0234 #0235	Set PG/COMM1 port	$\begin{array}{c c c c c c c c c c c c c c c c c c c $
		 Communication port requires to set station number (001 to 037(8)) to #0235 since the contents of communication are the same as those of the link module (JW-21CM) placed in command mode. #0235 Station number The initial values both #0234 and #0235 are 00(H).
#0236 #0237	Set PG/COMM2 port	 Specify communication condition of the PG/COMM2 port (communication port 2) to bits (Do to D6) of #0236. Bit 7 6 5 4 3 2 1 0 #0236 0 #0236 0 Transfer speed Data length Ds Stop 0 1 bit 1 2 bits D4 D3 Parity Stop bit length D2 D1 D0 Transfer speed 0 0 0 9600 bits/s 0 0 1 19200 0 1 0 38400 0 1 1 76800 1 0 0 115200 1 0 1 230400 - Communication port requires to set station number (001 to 037(e)) to #0237 since the contents of communication are the same as those of the link module (JW-21CM) placed in command mode. #0237 Station number The initial values of both #0236 and #0237 are 00(H).

Reference (Set PG/COMM port).

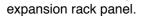
- With the JW30H, data length is "fix to 7 bits," and data transfer speed (bps) shall be select from "1200, 2400, 4800, 9600, 19200, 38400, 57600, and 115200".

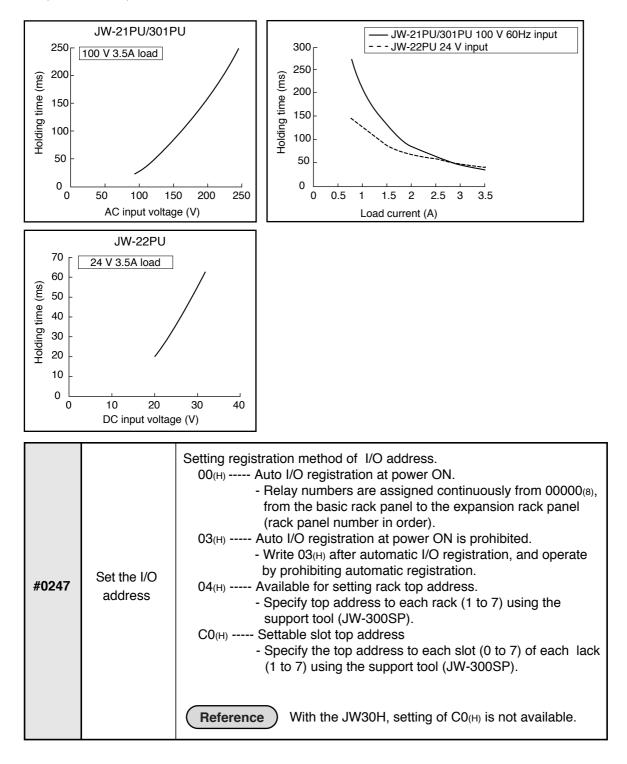


		Bit 7 6 5 4 3 2 1 0 #0244 Imput 20 Input 20 Iabel LB1400) 17 Input 21 Input 22 Iabel LB1401) 18 Input 22 Iabel LB1402) 19 Input 23 Iabel LB1403) 20 Input 23 Iabel LB1403) 20 Input 24 Iabel LB1403) 20 Input 25 Iabel LB1403) 20 Input 26 Iabel LB1403) 22 Bit 7 6 5 4 3 2 1 0 #0245 Imput 26 Iabel LB1407) 24 24 Iabel LB1407) 24 Bit 7 6 5 4 3 2 1 0 #0245 Imput 27 Iabel LB1407) 24 24 Input 30 Iabel LB1410) 25 Input 31 Iabel LB1411) 26 Input 32 Iabel LB1412) 27 Input 33 Iabel LB1413) 28
#0240 2 #0245	Set interrupt processing	 Input 34 (label LB1414) 29 Input 35 (label LB1415) 30 Input 36 (label LB1416) 31 Input 37 (label LB1417) 32 denotes 0 ··· Interrupt on falling edge (OK to OFF). denotes 1 ··· Interrupt on rising edge (OFF to ON). The initial value of bytes #0244, #0245, is 00₁₉₀. Remarks Interrupt requests are treated in much the same way as the F-142 (CALL) instruction and F-148 (CAL+) instruction. Notes for interrupts are also the same as for those instructions. Locate the subroutine programs by placing the subroutine labels following the F-40 (END) instruction in the main program. Set an interrupt program to operate within 1ms. If longer than 1ms, interruption may be ignored. Interrupts are effective if an output or special I/O module is installed in the slot and rack panel number where input interrupts are available. Pay attention to the correct module installation. Interrupts on an input module installed in the remote I/O slave module (JW-21RS) is disabled. Timer and input interrupts are ignored if no jump destination label is specified (by F-140). If more than one interrupt has simultaneously occurred, the one with the highest priority (with smaller number) is serviced first. The controller will handle interrupts that occur while exchanging data with an I/O module as follows: I/O module Executes the interrupt during data exchange. The break function is invalid while interruption is being set. The subroutine call instruction (F-142 [CALL], F-148 [CAL+1]) cannot be used within an interrupt handling program.

#0246	Setting the allowable voltage interruption time length	Set when varying the momentary power failure detection time from 0 to 255 ms. Setting value is decimal. Initial setting is 010(D) (10ms).
-------	---	---

- When varying the setting of system memory #0246 (setting of extension of momentary power failure detection time), the variable range is determined by the power source input voltage and load current. Extend the momentary power failure detection time by referring to the following graphs of holding time characteristics and output holding time characteristics.
- During momentary power failure, all actions of control module are stopped, and the operation continues after power recovery. If longer than the momentary power failure detection time setting (power failure), after recovery, the operation must be started by turning on the power again.
 Be careful sufficiently in consideration of the capacity of 5V in the basic rack panel and





		Setting	g RO	M operati	on m	ode.					
		Setting	y value	Transfer F to RAM (at p supply C	ower	Data Memo after trans	fer	trai	lode after 1sfer power upply ON		Transfer ROM to RAM by tool
		000	00	No	···/ P						Available
		021	11	Yes		Hold		Mode at r	power supply (DFF	Available
#0255	Set ROM	042	22	Yes		Clear *1			Stop		Available
	operation mode	104	44	Yes		Clear *1	1		Run		Available
				1				* Stored	data within F	RON	I is holding.
			The initial value is 00(H). See the section on "ROM operation" in the JW300 user's manual hardware version for details.								
		Set the	e stor	ring conte	ents ir	nto a RO	M to	be use	ed for operat	tion	using
		the RO	DM.								
		Set v					tents	s to RON	1[⊖:Yes, ×	: No)]
		ост	HEX	System me [#200 to #	emory 2777]	Progr	am	*2	Register *3	Fi	le register
		200	80	0		-	2		×		×
		201	81 82	00			<u>)</u>)		0 ×		× 0 *4
		203	83	00			×		0		×
		204	84	0			×		×		0 *5
		205 85 C			O ×			0			O *4 O *5
#0256	Set contents into ROM	, JW-3 JW-3 JW-3	2 RC 11CL 21CL 31CL	is 80(H). DM area o J/312CU J/322CU J/332CU J/342CU	8 16 32	gram K words K words K words K words	109 209 309	↓ 9000 to 9000 to 9000 to 9000 to	3 ROM area 099777 199777 299777 389777 E7777	a of	register
		JW-3	52CL	J	128 K words cannot			annot be	stored		
		JW-3	62CL	J	256 K words as I			SROM.			
		*4 R0	DM ai	rea of file	regis	ster		*	5 ROM area	a of	file register
		JW-3	11CL	J/312CU		None)		N	one	
					0000	0000 to C	007	7777 ₍₈₎ bytes)	00000000 t		0077777 ₍₈₎ 2 K bytes)
		JW-3	31CL	J/332CU	0000	0000 to C	007		00000000 t	0 0	
		JW-3	41CL	J/342CU	0000			77777 ₍₈₎ (bytes)	00000000 t		177777 ₍₈₎ 2 K bytes)
		JW-3	52CL	J		(25	56 K	(bytes)		(51)	2 K bytes)
		JW-3	62CL	J	0000			7777 ₍₈₎ (bytes)	00000000 t		3777777 ₍₈₎ M bytes)

		- Place com port 3) to b								f the EA-	PG p	oor	t (c	ommunication
		Bit #0266 [Data lengt	7 0 h—	6	5	4	3	2	1	0 Trans Parity Stop b		bee	d	
#0266		D6 Data		D	5	Stop	C	D4	Dз	Parity	D2	D1	D ₀	Transfer speed (bps)
#0267	Set EA-PG	0 7 bit	s	0		1 bi	t	0	0	None	0	0	0	9600
	port	1 8 bit	s	1		2 bi	ts	0	1	Odd	0	0	1	19200
								1	0	Even	0	1	0	38400
								1	1	Disable	0	1	1	76800
											1	0	0	115200
		the conten module (JV	ts c	of co 1Cl	omr M)	nun plac	icat ced	ion in c	are	the sam	e as			to 037 ₍₈₎ since of the link
		#0267				on n	-				- 00			
		- The initial	valu	les	bo	th #	026	6 ai	nd #	#0267 ar	e 00	(H) .		

#0270 #0271 #0272	Transfer system memory #0000 to #2777 to register	When to transfer system memory #0000 to #2777 to a specified register (except file registers), place the setting meaning to "transfer" to #0272, and top address to destination to #0270 to #0271 with file address. In addition, enter a "number multiplied by 4" (such as 009000, 009004) for the specified register. If other value than "number multiplied by 4" is entered, bits Do and D ₁ of #0270 are set to 0. For example, if 009001 (file address 004001 ₍₈₎) is entered, it will be changed to 009000. • #0272 • #0277 • #0271 #0270 • #0271 #0270 • #0271 #0270 • digit [Example] File address 00170000 ₍₈₎ is top address #0271 #0270 • 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 7 0 0 0 0 0 0 0 0 0 Initial value is #0270, #0271 = FF(H), #0272 = 00(H)
#0277	BCC check	The JW300 automatically computes and registers BCC check code for
#0277	BCC check	the contents of system memory actions #0200 to #0276.

#0410 to #0414	Specify use area (type) of symbol/comment	Specify top addresses and type of symbol/comment for relay, timer/counter, register, file register, and F-90 (REM) instruction that are written to the JW300 (read from the JW300) using the ladder logic programming software JW-300SP. Top address (#0410 to #0413) Specify file address of the file register to #0410 to #0413. #0411 #0410 5th digit 4th digit 3rd digit 2nd digit 1st digit #0413 #0412 0 0 0 0 0 0 0 0 Bth digit 7th digit 6th digit [Ex.] When file register 17560000 ₍₈₎ (file address 17760000 ₍₈₎) is top address. #0411 #0410 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 #0413 #0412
		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$

Set device net

#0300 to #0377	Set device net module (Module No. 0)	 Set the following when the device net master module (JW-20DN) is used with its module No. switch set to "0." When it is in master mode #0300 to #0303: Top address of the I/O table (*) #0304 to #0307: Top address of the diagnosis table (*) #0310 to #0313: Top address of the explicit message table (*) #0314 to #0317: Top address of the scan list table (*) #0321: Data length when editing the scan list #0324, #0325: ISD (communication monitor time) #0326, #0327: EPR (communication monitor time) #0330: Slave output status when the JW300 is stopped operation When it is in slave mode #0360 to #0363: Top address of the I/O table (*) #0364 to #0367: The number of I/O bytes #0370: Latch/clear the slave area when a communication error occurs. #0371, #0372: Response time to the master module
#0500 to #0577	Set device net module (Module No. 1)	 Set the following when the device net master module (JW-20DN) is used with its module No. switch set to "1." When it is in master mode #0500 to #0503: Top address of the I/O table (*) #0504 to #0507: Top address of the diagnosis table (*) #0510 to #0513: Top address of the explicit message table (*) #0514 to #0517: Top address of the scan list table (*) #0521: Data length when editing the scan list #0526, #0527: EPR (communication monitor time) #0530: Slave output status when the JW300 is stopped operation When it is in slave mode #0560 to #0563: Top address of the I/O table (*) #0560 to #0563: Top address of the I/O table (*) #0560 to #0563: Top address of the I/O table (*) #0560 to #0563: Top address of the I/O table (*) #0560 to #0563: Top address of the I/O table (*)

* When the free allocation is selected.

- For the details of the #0300 to #0377 and #0500 to #0577, see the JW-20DN user's manual. For your information, #0320, #0331 to #0377, and #0520, #0531 to #0577 are not used areas.

#2300 to #2377	Set device net module (Module No. 2)	 Set the following when the device net master module (JW-20DN) is used with its module No. switch set to "2." When it is in master mode #2300 to #2303: Top address of the I/O table (*) #2304 to #2307: Top address of the diagnosis table (*) #2310 to #2313: Top address of the explicit message table (*) #2314 to #2317: Top address of the scan list table (*) #2321: Data length when editing the scan list #2326, #2327: EPR (communication monitor time) #2330: Slave output status when the JW300 is stopped operation When it is in slave mode #2360 to #2363: Top address of the I/O table (*) #2360 to #2363: Top address of the I/O table (*) #2360 to #2363: Top address of the I/O table (*) #2370: Latch/clear the slave area when a communication error occurs. #2371, #2372: Response time to the master module
#2400 to #2477	Set device net module (Module No. 3)	 Set the following when the device net master module (JW-20DN) is used with its module No. switch set to "3." When it is in master mode #2400 to #2403: Top address of the I/O table (*) #2404 to #2407: Top address of the diagnosis table (*) #2410 to #2413: Top address of the explicit message table (*) #2414 to #2417: Top address of the scan list table (*) #2421: Data length when editing the scan list #2426, #2427: EPR (communication monitor time) #2430: Slave output status when the JW300 is stopped operation When it is in slave mode #2460 to #2463: Top address of the I/O table (*) #2460 to #2463: Top address of the I/O table (*) #2460 to #2463: Top address of the I/O table (*) #2470: Latch/clear the slave area when a communication error occurs. #2471, #2472: Response time to the master module

* When the free allocation is selected.

- For the details of the #2300 to #2377 and #2400 to #2477, see the JW-20DN user's manual. For your information, #2320, #2331 to #2377, and #2420, #2431 to #2477 are not used areas.

■ Number of error occurrence for each error code The number of errors occurred is stored for each error code (max. 32 types).

#1600, #1601	The number of errors that occurred for each error code: No. 1	#1601: T t	rror code _(H) he number of errors hat occurred 000 to 377 ₍₈₎)
#1602, #1603	The number of errors that occurred for each error code: No. 2		
#1604, #1605	" No. 3		
#1606, #1607	" No. 4		
#1610, #1611	" No. 5		
#1612, #1613	" No. 6		
#1614, #1615	" No. 7		
#1616, #1617	" No. 8		
#1620, #1621	" No. 9		
#1622, #1623	" No. 10		
#1624, #1625	" No. 11		
#1626, #1627	" No. 12		
#1630, #1631	" No. 13]	
#1632, #1633	" No. 14	Address	Content
#1634, #1635	" No. 15	n	Error code(H)
#1636, #1637	" No. 16		The number of
#1640, #1641	" No. 17	n + 1	errors that occurred
#1642, #1643	" No. 18		(000 to 377(8))
#1644, #1645	" No. 19		
#1646, #1647	" No. 20		
#1650, #1651	" No. 21		
#1652, #1653	" No. 22		
#1654, #1655	" No. 23		
#1656, #1657	" No. 24		
#1660, #1661	" No. 25		
#1662, #1663	" No. 26		
#1664 #1665	" No. 27		
#1666, #1667	" No. 28		
#1670, #1671	" No. 29		
#1672, #1673	" No. 30		
#1674, #1675	" No. 31		
#1676, #1677	" No. 32		

- No.1 to No. 32 are stored error codes in order of occurrence of errors.

Set logging

Set #2100 to #2165 for using the logging function (F-403 instruction). => See page 6-7.

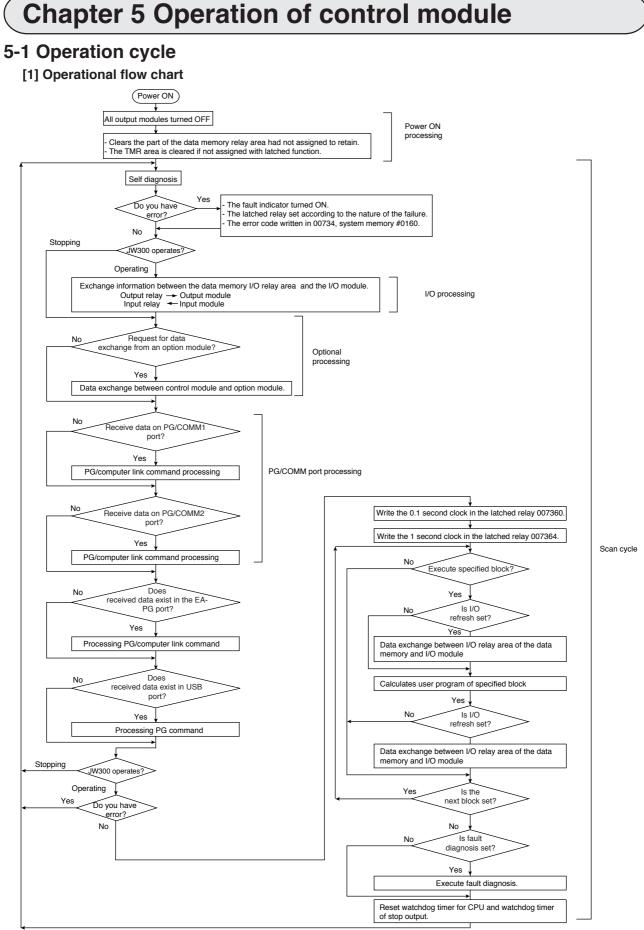
001 #2	00 10 #2103 101 1		ging function (F-403 instruction). => See page 6-7.							
		Specify top address to store logging data into file address #2100 to #2103. These also can be set to extension memory (file address 2000000000(8) and up) using a SRAM card. => See page 6-19.								
		[Ex.] When setting register 109000								
#2100 to #2103	Top address of logging data storage area	$ \begin{array}{cccccccccccccccccccccccccccccccccccc$								
	Storage area		"0100 "0100							
			$ \begin{array}{cccccccccccccccccccccccccccccccccccc$							
		- Fi	ile address of register 109000 is $00036000_{(B)}$.							
#2104	Number of storages of logging	Specify the number of storages (set value x 10) of logging data. [Ex.] when 050(D) is specified as the set value, the number of storages is 50 x 10 = 500 times.								
		Specify for	mat of time stamp that is added to logging data.							
	Time stamp format	Set value(H)	Format							
#2105		00	None							
		01	Year, month, day, hour, minute, second (add 6 bytes to logging data)							
		02	Hour, minute, and second (add 3 bytes to logging data)							
#2106 to #2111	Logging specifying register 1	Set registe setting met	or address to execute logging using file address. The thod is the same as for the #2100 to #2103.							
#2112 to #2115	Logging specifying register 2									
#2116 to #2121	Logging specifying register 3									
#2122 to #2125	Logging specifying register 4									
#2126 to #2131	Logging specifying register 5									
#2132 to #2135	Logging specifying register 6	Set logging	g specification register 2 to 10 with the same method as specification register 1 (#2106 to #2111).							
#2136 to #2141	Logging specifying register 7									
#2142 to #2145	Logging specifying register 8									
#2146 to #2151	Logging specifying register 9									
#2152 to #2155	Logging specifying register 10									

#2156 to	These function as a counter to store the next logging data.
#2161	0 to [the number of storages of logging (#2104) - 1]
#2162 to	The number of storage times of logging data
#2165	0 to FFFFFF(H)

Setting of PC card

If the JW-3*2CU is used as a control module and you will use a PC card, set #2220 to #2226 and #2230 to #2236.

#2220	Writing to a PC card is prohibited.
#2221	Select file type to store in a PC card.
#2222	Set data transfer between a PC card and the JW-3*2CU.
#2223 to #2225	Automatically save data to a PC card for each time of a specified interval.
#2226	When an error occurs for reading/writing to a PC card, the respective error code is stored.
#2230 to #2236	If the AUTO LD switch on the JW-3*2CU is OFF, specify a file name to save on a PC card when it is inserted.



- The F-80 instruction allows I/O operations when the PC is busy for internal processing.

- Program can be executed using interrupts service (#0240 to #0245) even when the PC is busy for I/O, option, or PG/COMM port processing.

5-2 Self-diagnosis

The JW30H self-diagnosis each module such as control module, and falls in the following state when abnormality occurs. Seek cause of abnormality according to this table, and take proper countermeasure => Also see "Troubleshooting" in the JW300 User's Manual Hardware Version.

				JW300		Control	Power mod	supply ule's		Error co	de (BCD)
	li	tem	Contents	operating state	Halt output	module	indic	cator	Special relay	Special register	System memory
				state		FAULT	POWER	RUN	*3	⊐00734	#0160 to 0167
			Instruction code check								24
	Memory		System memory setup check								23
			Program ROM check			ON		OFF	007370	20	25
		error	Program sum check								26
			I/O registration table check								28
			Watchdog timer			OFF		Blinks	—	00	31
	C	PU error	RAM check (R/W)						007271	30	32
			Hardware check	Stop	Open	ON	ON	OFF	007371	30	35
			I/O data bus							40	44
		During	Output data check								42
sis		refresh	Installed module check								40
Self diagnosis			I/O rack panel error								48
alt di			Number of module bytes check								45
ຶ່	error	During table	Table verify error							60	60
	ē	verify	Switch verify error						007373		61
	₽Ì		Table registration error								70
		During table	Missing module error							70	71
		registration	I/O points overflow error							70	72
			Switch set error								73
			Hardware error						007375		46
	Sp	ecial I/O	Parameter error						007375	40	47
		error	*1 *2 Fuse blown of JW-262S	Run	Close	OFF	ON	ON	007363	40	49
				Stop	Open	ON	ON	OFF			
			*1 Hardware error	Run	Close	OFF ON	ON ON	ON OFF	-		53
	Ор	tion error	Option command error	Stop Run	Open Close	OFF	ON	OFF	007374	50	54
			System protect error	Run	Close	OFF	ON	ON	1		55
F	Pov	ver supply error	Power failure or voltage drop			OFF	OFF		007377	10	13 *4
	•			Stop	Open			OFF	007376	40	43
		ttery error	Battery voltage drop or battery not inserted	Run	Close	ON	ON	ON	007372	20	22
Н	alt	output	Relay output, 100/200 V	AC, 30 VD	DC, 1A,	Turn ON	(closed) d	luring JV	/300 ope	ration.	

*1: The upper or lower state of each item may occur when the fuse is melted down in the system memory #0206 or #0207, or by the setting in the case of option abnormality.

 (Setting)
 (State)

 Continue operation
 → Upper column

 Halt
 → Lower column

*2: When external power is not additionally supplied to the JW-262S, a blown fuse error occurs. Especially be careful when to set #0206 as halt operation.

*3: The special relays 07370 to 07377 turn ON when the self-diagnosis is detected. When an error occurs, monitor the special relays; using the support tool, host communication, or data link, you can check the error details. (Though the special relays turn ON when an error occurs, the module does not process I/O during occurrence of the error, so you cannot take out data from the output.)

*4: A power supply error is stored when the power is ON, even if the module is normal.

Note: When the control module detects an error while running a self-diagnostic test, an error code will be stored. However, the condition of the operating status, output halted indicator (such as a FAULT), and the special relay will be as follows:

- If the error is released within the watchdog timer cycle time (300 ms), the machine condition will be different from the table above.

- If the error continues, exceeding the watchdog timer cycle time (300 ms), the machine condition will be as shown in the table above.

[1] Description

Error details of (1) to (28) in the previous page after the self-diagnosis are as follows:

(1) Instruction code check

Program memory code is checked when an instruction is executed. Program addresses where instruction code errors are found are stored in system memory locations #0052 and #0053.

(2) System memory setup check

Sum check is done to the system memory #0200 to #0256.

(3) Program ROM check

When operated in the ROM mode, the ROM is checked when transferring program from ROM to RAM. An error is established if a failure is found in the flash ROM.

(4) Program sum check

When the user's program is written or revised, a checksum will be created automatically. When power is supplied, the CPU can use the checksum to detect whether the content of the user program has changed. However, the checksum function cannot be used to identify which part of the program has changed. If a program checksum error occurs, you have to retransmit the system memory settings, program, and data (if required) in order to restore the system.

(5) I/O registration table check

When registering I/O modules, a checksum will be created for the stored data. When power is supplied, the CPU can use the checksum to detect whether the data has changed. If an I/O module table checksum error occurs, you have to do again the I/O module registration.

(6) Watchdog timer

When the watchdog timer in the CPU module times out, this error will occur (the status is the same as in the program mode).

(7) RAM check

Prior to going into every operational cycle, the data memory RAM is checked if read/write is possible.

(8) Hardware operation

Before going into the operation cycle, correct setup of the accumulator and stack is checked.

(9) I/O data bus

Prior to I/O processing, the I/O data bus is examined if it is in the floating state. The faulty module's slot number is stored in system memory location #0046.

(10) Output data check

Rereads and verifies, in I/O processing, the data output to the output module. If verification fails, an error occurs.

- The special I/O module does not perform this check.

(11) Installed module check

Verifies, in I/O processing, the installation state of a module registered in the I/O table when the CPU performs a data exchange with each module. If verification fails, an error occurs.

(12) I/O rack panel error

Checks that all I/O port gates built into the rack panel are closed before I/O processing. If an open port is detected, an error occurs.

(13) Module number of bytes check error

When the PC is processing a 16/32/64 point I/O module, and if the type (number of bytes) registered in the I/O table does not match the internal byte counter, this error will occur.

(14) Table verify error

Verifies, when the power is turned ON or when the mode changes (stop mode to run mode), the installation state of each module with the contents of the I/O table which has a already been registered. If verification fails, an error occurs. This error also occurs when the setting of the rack number switch on the expansion rack panel or the connection state of the I/O expansion cable changes.

(15) Switch verify error (module no. switch verify error)

Additionally verifies the module no. switch setting for the special I/O, option, Device Net or I/O link master module during I/O table verification when the power is turned ON or when the mode changes (stop mode to run mode). If verification fails, an error occurs.

(16) Table registration error

This error occurs when an initial error is generated (e. g. due to an incorrect setting of the rack number switch on the expansion rack panel or an incorrect connection of the I/O expansion cable) during I/O table registration.

(17) Missing module error

An error occurs when the contents of the table data are registered as a "state without any module installation space" during I/O table registration. It is not the state in which no module have been installed.

(18) I/O point overflow error

An error occurs if the number of installed module is too great and the number of I/O points exceeds the number of control I/O points for the control module during I/O table registration.

(19) Switch set error (module no. switch set error)

This error occurs if the settings of the module no. switches on the special I/O or option modules conflict during I/O table registration.

(20) Special I/O hardware error

This error occurs when the watchdog timer for the CPU built into the special I/O module is activated due to an error in the special I/O module itself.

(21) Special I/O parameter error

This error occurs if parameter verification fails when the control module transfers parameters to the special I/O module.

(22) Fuse blown of special I/O (JW-262S)

This error occurs when fuse of JW-262S is OFF.

When no external power is supplied to the JW-262S, a blown fuse error will also occur. Take special precautions to allow for this case when system memory location #0206 is set to "halt operation."

(23) Option hardware error

This error occurs when the watchdog timer for the CPU built into the module is activated due to an error in the option, device net, or I/O link master module.

(24) Option command error

The data exchange commands between an option module and a control module are checked. If the command is not appropriate, due to an external cause such as electrical noise, this error will occur.

(25) System protection error

When a data exchange command between an option module and control module requests a write to the system area of the control module, this error will occur.

(26) Power supply error

The JW300 ignores instantaneous power failures of 10ms or less and continues operation. But, a power failure above the limit causes the CPU to halt and the halt output is forced open. The operation is resumed automatically as soon as it recovers from the power interrupt.

- As a gradual supply voltage drop occurs, the CPU stops when it goes below 85% of the rated voltage and the halt output turned open.

But, the operation restores automatically when the normal power supply is recovered.

(27) Expansion power supply error

An error occurs when the voltage of the expansion power supply (5 VDC) drops below 4.5 V. When this error occurs, the output module is reset for the rack panel to which is mounted the power supply where the error occurred.

- If another I/O error occurs simultaneously with an expansion power supply error, it may be stored priority expansion power supply module error as error history.

(28) Battery error

Proper battery supply voltage is checked.

Choice is possible using the special relay 007372 to actuate the battery failure lamp or buzzer.

- On when in error

Nothing affects the JW300 operation as long as power is ON to the JW300, even if the battery is in a fault condition. In this event, the battery needs to be replaced with the fresh one to avoid an incidental power supply failure.

[2] Halt output

Though normally closed, the line goes open when an error is detected after diagnosis.

Connection of JW300's halt output to the system's emergency stop circuit enables to shut OFF the system when an error occurred within the programmable controller.

Note: If the error is recovered within the time of the watchdog timer (300 ms), the halt output does not turn ON. If the error is left exceeding the watchdog timer interval (300 ms), the halt output will open.

[3] Special relay

Result of diagnosis is written in the special relay area of the data memory.

When the JW300 stops after an error as a result of diagnosis, the nature of the error may be known when the special relay (007370 to 007377) is examined using the support tool.

- Because the diagnosis runs at every scan cycle, the halt output closes upon recovery from the fault and the JW300 resumes its operation. The diagnostic special relay is also reset.
- Of the special relays, only 007372 (battery error), and 007374 (option module error) can be read from output module by JW300's execution. To read the status of other relays, use the JW-21CM computer link or a support tool.

The status of special relays cannot be read with the data link either.

- The contacts of relay 007377 (power supply error) is closed for 1 scan at power ON.

[4] Error code

(1) Special register

When an error is established after the diagnosis, the error code is written in the data memory special register (byte address $\exists 00734$).

- When another error occurred apart from the current error, the error code of higher priority takes preference over other error codes.
- As soon as the error cause is removed, the error code is canceled.

(2) System memory

When an error is established after the diagnosis, the error code is written in the system memory (#0160 to #0167).

#0160 to #0167 acts as a shift register and can store eight errors at a maximum. If more than eight errors occurred, the error code first stored is canceled and so on.

			#0165	#0164	#0163	#0162	#0161	#0160	
System memory clear	00	00	00	00	00	00	00	00	
	00	00	00	00	00	00	00	24	← Instruction code error
	00	00	00	00	00	00	24	44	← I/O data bus error
24 (Lost) ←	44	xx	xx	xx	xx	xx	xx	22	← Battery error

#0167 #0166 #0165 #0164 #0163 #0162 #0161 #0160

- While major error codes are stored in the special register, minor error codes are stored in the system memory.
- The error code in the system memory would not be cleared after recovery from the error. To clear the error code, "00" must be written in the system memory (#0160 to #0167) using the support tool such as the programmer.
- If multiple errors should occur in succession, no error will be written.

(3) Registers

- Registers E7600 to E7777 hold error codes at the time of error occurrence.
- (Setting the 02(H) in the system memory #0213)

[5] ON/OFF state of the output module during error

Depending on how the system memory #0232, #0233 (000000 to 015777) and #0252, #0253 (020000 to 075777) output retention address was programmed, ON/OFF state of the output module is determined when the JW300 stops after self-diagnosis.

- Output module before the output retention address --- OFF
- Output module after the output retention address --- Retains ON/OFF state before stop.

But, the output module before the output retention address may not be turned OFF, depending on the nature of the error. The output that needs to be turned OFF at a time of JW300 error must be connected serial to the halt output of the control module. Refer to JW300's user's manual, hardware version.

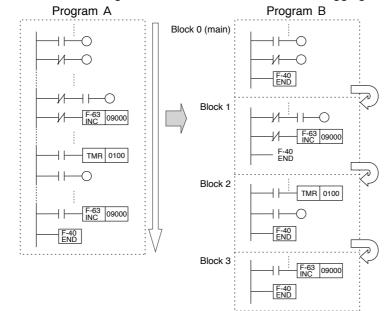
Chapter 6 Block operation, logging, fault diagnostic, and PC card

6-1 Block operation function

The JW300 has a "block operation function" that facilitates programming user programs after dividing into processing blocks (minimum 0.5 K-word) as processing units.

Using the block operation function, you can manage the programs in units of blocks, and handle each block as an independent JW300 so that you can manage the user programs easily. Further, this function is effective for shortening scan time, and convenient for debugging and test operation.





- This function divides Program A into blocks (as seen in Program B) for each functional purpose and makes management easy. You can operate and wait program for each block, so this is effective for test operation and debugging operation.
- Normally, execute only the needed blocks. For each scan, un-needed blocks are not scanned, so that total scan time can be shortened.

Processes available to execute in block units

Process	Description
Operation/wait	Using the support tool (JW-15PG, JW-300SP) and block start relay, you can run (start) and wait (halt) program operation for each block. The waiting blocks do not execute programs and remain in stop mode status.
I/O refresh	I/O refresh operation can be set and executed for each block.
Program management	Including program changes of the currently running program, each block can be handled as an independent program.

Number of blocks and sub programs

The numbers of blocks and sub-programs that can be used with the block operation vary with the model of the JW300 control modules. => See the structuring program (page 4-2).

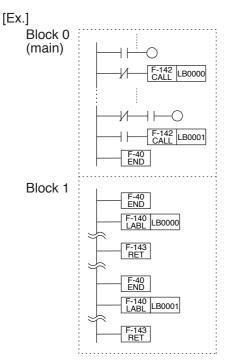
	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Number of blocks (max.)	16	32	64	128	256	512
Number of sub programs (max.)	256	512	1024	2048	4096	8192

Jump between blocks

The cursor can be jumped between blocks.

In the example below, Block 0 (main) is used as the normal user program and Block 1 as the subroutine block so that you can easily manage the blocks.

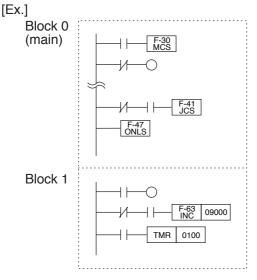
In this case, even though the block is in waiting status, the JW300 executes calculation of the jumped-to target block.



MCS and ONLS instructions

The MCS (F-30) and ONLS (F-47) instructions do not need to be reset, as these operations do not affect the next blocks.

The contents of the accumulator (ACC) and stack register (SR), which were stored by the PUSH instruction, are all initialized when starting the operation of each block.

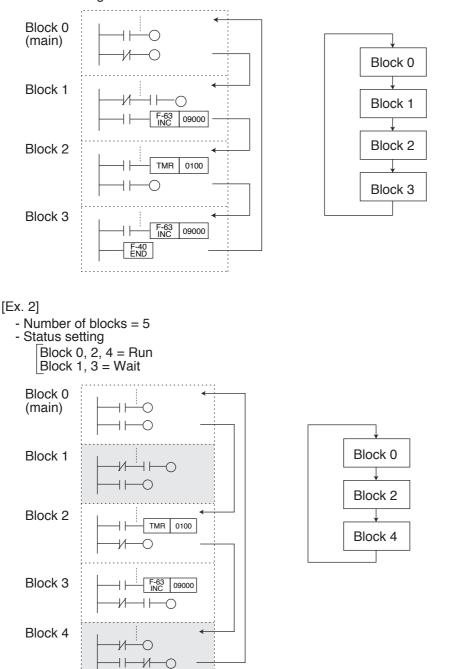


- The MCS (F-30) and ONLS (F-47) instructions do not need to be reset at Block 0, as these operations do not affect Block 1.

[1] Set block status

Each block status (run/wait) can be set using support tools (JW-15PG, JW-300SP). This is convenient for debugging of each device and partial debugging.

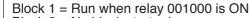
- [Ex. 1]
 - Number of blocks = 4
 - Status setting = Run all blocks



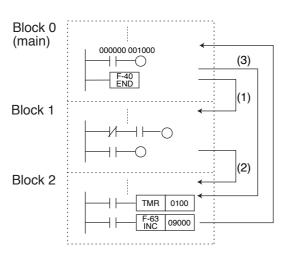
[2] Block start relay

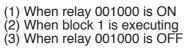
Set block start relays to each block (Block 1 and after) using the support tool (JW-15PG, JW-300SP), and you can control run (start) and wait (halt) each block using the ladder program.

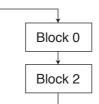
- [Ex. 1]
- Number of blocks = 3
- Setting block start relay



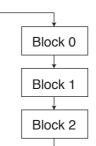
Block 2 = No block start relay







Run when relay 001000 is OFF



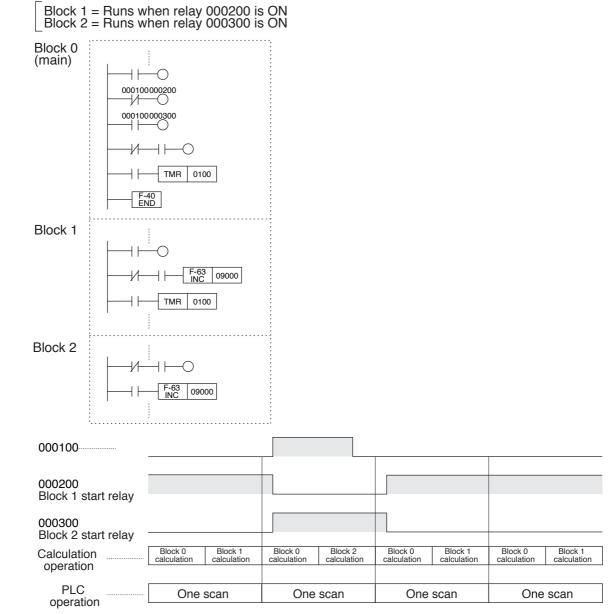
Run when relay 001000 is ON

Start relay 001000							
Calculation operation	Block 0 calculation	Block 2 calculation	Block 0 calculation	Block 1 calculation	Block 2 calculation	Block 0 calculation	Block 2 calculation
PLC	One scan			One scan		One	scan

[Ex. 2]

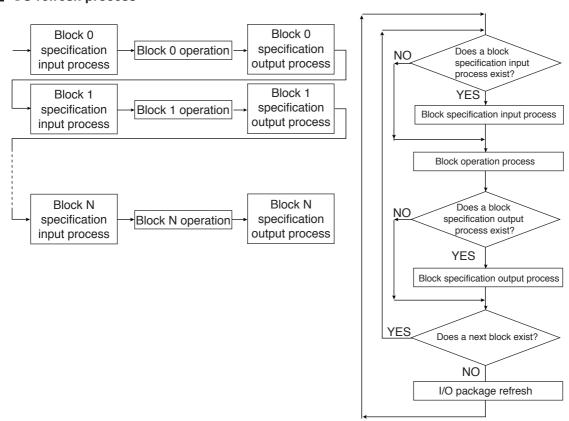
Using the block start relay, you can branch blocks. Below is an example that when relay 000200 is ON, Block 1 runs, and when relay 000300 is ON, block 2 runs (starts).

- Number of blocks = 3
- Setting block start relay



[3] I/O refresh

Specified I/O can be refreshed at each block. I/O processing for each rack/slot can be set for each block, and you can process each I/O quickly.



I/O refresh process

[Precautions]

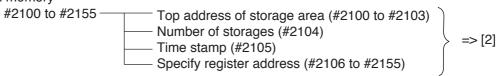
- When the I/O refresh is not set for each block, the JW300 does not I/O refresh the not yet set blocks.
- I/O refresh is not executed for blocks in waiting status.
- Do not use addresses of input modules that are used for input interrupt (system memory #0240 to #0245).
- The upper limit of the slot number varies with the rack panel used.

6-2 Logging function

Taking specified registers as the trigger condition, the data memory can be logged. This logging function is useful for analyzing devices when an error occurs, and data is collected at the rated time.

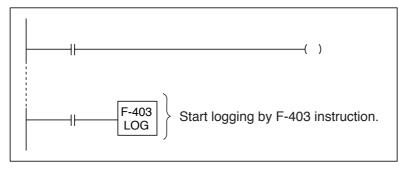
To use the logging function, you have to program application instruction F-403 (LOG) and set system memory #2100 to #2155.

- Application instruction F-403 (LOG) ---- Trigger logging => [1]
- System memory



[1] Programming application instruction F-403 (LOG)

Triggered by the application instruction F-403 (LOG) for logging, the JW300 starts logging. - Operation condition of F-403 (LOG) is the rising edge of the input signal (OFF to ON).



[2] Setting system memory #2100 to #2165

Specify top address of area to store (#2100 to #2103), number of storages (#2104), time stamp (#2105), and register address for logging (#2106 to #2155) to each system memory.

System memory No.(8)	Set item	Description				
#2100 to #2103	Top address of logging data storage area	Set top address of area to store logging data to file address #2100 to #2103. Ex. 1: When setting register 109000 $\begin{array}{c} #2101 & #2100 \\ \hline 0 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0$				
#2104	Number of storage of logging	Specify the number of storages of logging data (set value x 10). Set range is 001 to 225(D). Ex.: When the set value is 050(D), the JW300 store logging data for 50 x 10 = 500 times.				
		Select format of time stamp that is added to logging data.				
	Time stamp	Set value (H)FormatNo. of bytes used00 (Initial value)None0				
#2105	format	01 Year, month, day, hour, minute second 6				
		02 Hour, minute second 3				

System memory No.(8)	Set item	Description						
		Up to 10 register addresses for logging can be set using file address.						
		System memory number						
		Logging register 1 #2106 to #2111						
	Logging register 2 #2112 to #2115							
	Logging register 3 #2116 to #2121							
	Logging register 4 #2122 to #2125							
		Logging register 5 #2126 to #2131						
		Logging register 6 #2132 to #2135						
		Logging register 7 #2136 to #2141						
		Logging register 8 #2142 to #2145						
	Logging	Logging register 9 #2146 to #2151						
#2106 to #2155	Logging specification register	Logging register 10 #2152 to #2155						
		- Logging registers without a specified system memory number shall be						
		set to FFFFFFFF _(H) (initial value). Ex. 1: When setting register 009000 to logging register 1						
		#2107 #2106 00001000 0000000						
		-^_0-^_4-^_0-^_0-^_0-/ #2111 #2110						
		$\begin{array}{ c c c c c c c c c c c c c c c c c c c$						
		- File address of register 009000 is 00004000(8).						
		Ex. 2: When setting file register 01000000(8) to logging register 3						
		$\begin{array}{cccccccccccccccccccccccccccccccccccc$						
		$\begin{array}{c} \begin{array}{c} \hline \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ $						
		- File address of file register 01000000(8) is 01200000(8).						

System memory No.(8)	Set detail
#2156	To be a counter to store next logging data.
to #2161	0 to [Number of logging storages (#2104) -1]
#2162	Number of times to store logging data.
to #2165	0 to FFFFFFF(H)

[3] Use examples

This section describes examples of using of the logging function.

(1) Setting example of system memory (#2100 to #2155)

Below shows logging data storage examples if the following are set to system memory (#2100 to #2155) of the logging function.

Item	System memory No.	Set value	Description
Top address of logging data storage area	#2100 to #2103	0020000(8)	File register 0000000(8) => File address 0020000(8)
Number of logging storages	#2104	20(D)	200 sets (20 _(D) x 10)
Time stamp	#2105	01 (H)	Year, month, day, hour, minute, second
Logging register 1	#2106 to #2111	0000000(8)	⊐00000 => File address 0000000(8)
Logging register 2	#2112 to #2115	0000001(8)	⊐00001 => File address 0000001(8)
Logging register 3	#2116 to #2121	0000002(8)	⊐00002 => File address 0000002(8)
Logging register 4	#2122 to #2125	0000003(8)	⊐00002 => File address 0000003(8)
Logging register 5	#2126 to #2131	00004000(8)	009000 => File address 00004000(8)
Logging register 6	#2132 to #2135	00004001(8)	009001 => File address 00004001 ₍₈₎
Logging register 7	#2136 to #2141	00004002(8)	009002 => File address 00004002 ₈₎
Logging register 8	#2142 to #2145	00004003(8)	009003 => File address 00004003 ₍₈₎
Logging register 9	#2146 to #2151	00016000(8)	E0000 => File address 00016000(8)
Logging register 10	#2152 to #2155	00016001(8)	E0001 => File address 00016001 ₍₈₎

Storage example of logging data

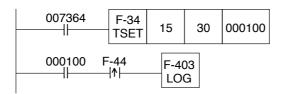
When the system memory is set as the example in the previous page, the JW300 stores logging data to area starting from file register $0000000_{(8)}$.

File register(8)	Data (Ex.)	Description	
00000000	03	Year	
00000001	11	Month	
0000002	20	Day	
0000003	13	Hour	
00000004	10	Minute	
00000005	35	Second	
0000006	02	Logging specification register 1 (data of ¬00000)	
0000007	05	Logging specification register 2 (data of ¬00001)	1st logging data
0000008	0A	Logging specification register 3 (data of ¬00002)	
0000009	A5	Logging specification register 4 (data of ¬00003)	
000000A	F1	Logging specification register 5 (data of 009000)	
0000000B	E0	Logging specification register 6 (data of 009001)	
000000C	09	Logging specification register 7 (data of 009002)	
000000D	03	Logging specification register 8 (data of 009003)	
0000000E	24	Logging specification register 9 (data of E00000)	
0000000F	57	Logging specification register 10 (data of E00001)	
00000C70	03	Year	
00000C71	11	Month	
00000C72	20	Day	
00000C73	18	Hour	
00000C74	25	Minute	
00000C75	14	Second	
00000C76	02	Logging specification register 1 (data of ⊐00000)	
00000C77	05	Logging specification register 2 (data of ⊐00001)	200th logging data
00000C78	0A	Logging specification register 3 (data of ⊐00002)	
00000C79	A5	Logging specification register 4 (data of ⊐00003)	
00000C7A	F1	Logging specification register 5 (data of 009000)	
00000C7B	E0	Logging specification register 6 (data of 009001)	
00000C7C	09	Logging specification register 7 (data of 009002)	
00000C7D	03	Logging specification register 8 (data of 009003)	
00000C7E	24	Logging specification register 9 (data of E00000)	
00000C7F	57	Logging specification register 10 (data of E00001)	

- From the 201st, logging data are overwritten on the 1st data and after.

(2) Program example of application instruction F-403 (LOG)

If the application instruction F-403 (LOG) for logging is programmed as follows, the JW300 stores data of the specified register at 15:30 as logging data.



Remarks

- If the number of logging storages (system memory #2104) is set to $000_{(D)}$, the JW300 logs data only one time.
- If an address out of the set range is specified in the logging specification register (system memory #2106 to #2155), the JW300 will not store its address data. It will store only data in the setting range.
- When the number of logging times exceeds the specified number of storages, the JW300 overwrites the 1st logging data and after. Another exceeding of the number of logging times also overwrites from the 1st logging data.
- When you will use an extension RAM using a PC card, the last address of the logging data storage area can be set up to file address 2007777777(8).
- When the logging system memory (#2100 to #2155) is changed during operation of the JW300, the set value when the application instruction is F-403 (LOG) will be effective.

6-3 Fault diagnosis function

By monitoring the execution time of each cycle, the relays can be diagnosed. Monitoring using conventional user programs can be performed by this fault diagnosis function. By using this function, you can save the user program and obtain error relay status immediately, so that maintenance ability can be improved.

To enable/disable the fault diagnosis function, set system memory #2200.

Address	Data	Description
	00(н)	Initial value
#2200	01(н)	Enable
	Other than above	Disable

Setting available number of cycles for each model (control module) shall be as follows:

	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Number of I/O cycle	128	256	512	1024	2048	4096
Number of always monitoring relays	128	256	512	1024	2048	4096
Monitoring time	0 to 32767 ms					

[1] Faulty diagnosis principle

(1) I/O cycle

		SOL (output)				
SOL (ou	tput)	-LS00 (zero position end limit sensor)				
-LS00 (zero position end limit sensor)	LS01 (target position end limit sensor)	LS01 (target position end limit sensor)				
		end innit sensor)	Start tim	e monitor	End ti	me monitor

When a machine moves from zero position end limit sensor (-LS00) to target position end limit sensor (LS01), the JW300 monitors the zero position end limit sensor (-LS00) and the target position end limit sensor (LS01) for a period after the output (SOL) turns ON until the target position end limit sensor turns ON, and detects following errors.

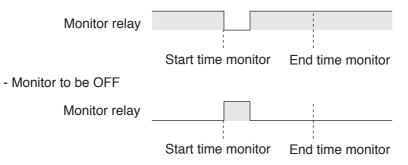
Monitor output	Machine status			
The target position end limit sensor does not turn ON.	The zero position end limit sensor is OFF and the target position end limit sensor does not change from OFF to ON.			
No output operation	The zero position end limit sensor does not change from ON to OFF. The target position end limit sensor does not change from OFF to ON.			
Zero position end limit sensor does not turn OFF.	The target position end limit sensor is ON and the zero position end limit sensor does not change from ON to OFF.			
Zero position is ON.	Zero position end limit sensor of the previous output is turned ON.			
Target position end limit sensor is OFF.	Target position end limit sensor of the previous output is turned OFF.			

(2) Always monitor relay

This is a function to always monitor a specified relay.

This function monitors whether a relay will go OFF within a specified interval, and whether a relay will go ON within a specified interval.

- Monitor to be ON

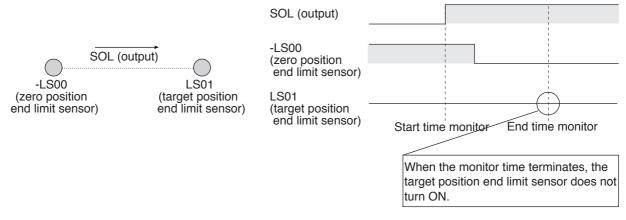


[2] Error details of fault diagnosis

(1) Error on I/O cycle

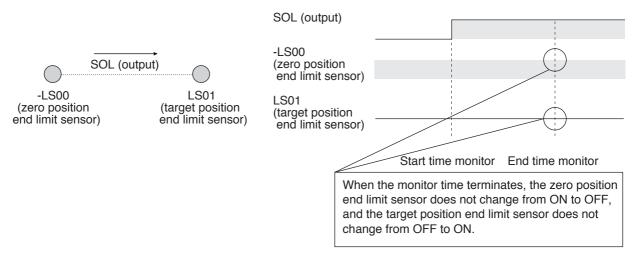
1) Does not turn ON the target position end limit sensor

The zero position end limit sensor is OFF and the target position end limit sensor does not change from OFF to ON



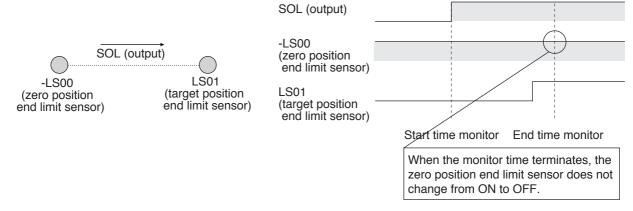
2) Does not output

The zero position end limit sensor does not change from ON to OFF, and the target position end limit sensor does not change from OFF to ON.



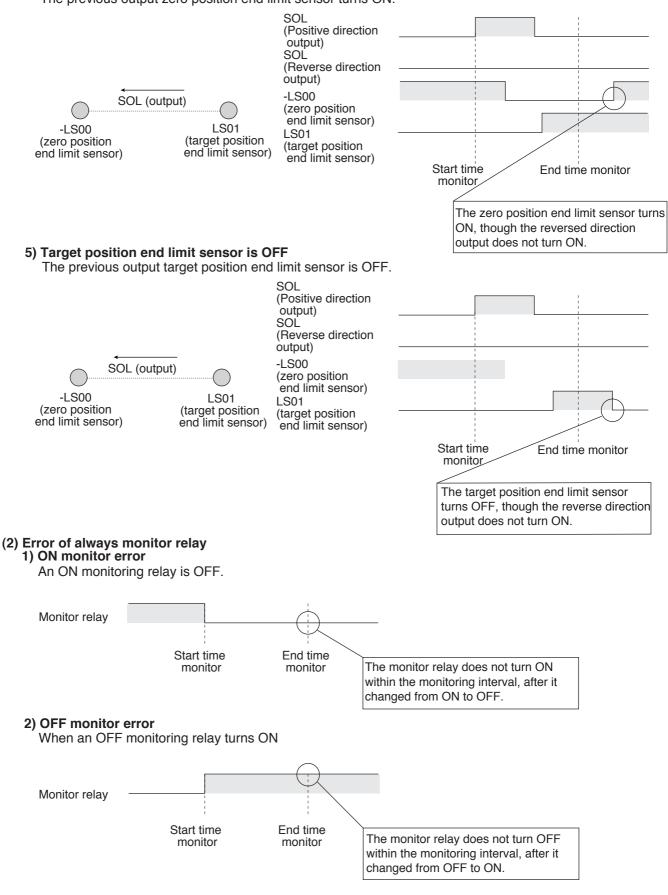
3) The zero position end limit sensor does not turn OFF.

The target position end limit sensor is ON and the zero position end limit sensor does not change from ON to OFF.



4) Zero position end limit sensor turns ON

The previous output zero position end limit sensor turns ON.



[3] Setting details

To use the fault diagnosis function, set the following items using the support tool (JW-300SP).

Set item	Description
	A relay to start the fault diagnosis function. When this is OFF, the JW300 does not diagnose faults.
Diagnosis result reset relay	When this is set to ON, clear all the diagnosis results.
History setting	Store the history of error occurrence to data or file memory.
Output relay when diagnosis result has an error	If an error is detected after fault diagnosis, this relay turns ON.

Remarks

- When the fault diagnosis is selected, total scan time will be longer. The execution time of the fault diagnosis is as follow: 24 μ s + (3 μ s x N cycles)

6-4 How to use the PC card

The JW300 control module (JW-3*2CU) is equipped with a memory card interface. Usage of the memory card is as follows:

(1) Save and load files

You can save and load user programs and data into a CF card. If there is no support tool, you can use the card for saving data on site. => See page 6-20 to 23.

(2) Memory capacity extensions

- Using a SRAM card, data memory capacity can be extended (up to 16 M bytes. => See table below).
- The card can be used to store symbol/comment data of a user program.
- Using the card, your can log in a large volume of operations and it can be used for trouble analysis and examination for shortening tact time.

Memory card interface of JW300 (JW-3*2CU)

Item	Specifications		
Interface (connector)	Connector for PC card type I/II (68 pins)		
Power supply voltage	3.3 V / 5 V power supply		
Usable memory card	CF card (need a conversion adapter) SRAM card (small size types need a conversion adapter)		
Objective data to save	 When used as save and load files User program Parameter memory System memory Data memory When used as memory extension Data memory (symbol, comment data, logging data) 		

Memory extension using a SRAM card

File address(8)	JW-312CU	JW-322CU	JW-332CU	JW-342CU	JW-352CU	JW-362CU
00000000						
00073777						
00105777						
00200000						
00277777						
00577777						
02177777						
10177777				 +		
40177777				 	 	
2000000000				·		
20077777777	*	*	*	*	*	*
(Max.)			* Ma	L ximum 16 M-b	yte SRAM car	d can be used.

[1] Objective files for save and load

To save and load a user program and data, use a CF card.

(1) CF card applicable format

FAT12	
FAT16	
VFAT	
FAT32	

- When VFAT and FAT32, long file name cannot be used.

(2) Type and extension of file

Files that can be saved/loaded into/from a CF card are as follows. The saved file has the following extension.

File type	Extension	Description
User program	ppg	Use as a set of block data and titles of block/sub program.
Block data	pbk	Data concerning blocks. When the user program is saved, the block data is created. Use as a set of user program and titles of block/sub program.
Title of block and sub program	ptl	Title data of block/sub program. It is created to save a user program. Use as a set of user program and block data.
System memory	psm	System memory data are loaded and saved. - When saving, the range is #0000 to #2777. When loading, the range is #00200 to #02777.
Parameter memory	рра	All of parameter memory data are saved/loaded.
Data memory (except file register)	pdt	All of data in the data memory (except file register) are saved/loaded.
File register	pfl	All of data of the file register are saved/loaded.

(3) File name

Set the file name with a maximum of 8 characters.

The following characters cannot be used.

[¥] [/] [:] [,] [;] [*] [?] ["] [>] [<] []

(4) Directory

The JW300 cannot use files in the directory. Make sure to enter into root directory of CF card.

(5) Files created by other model

Files that were saved by another model of control module (JW-3*2CU) cannot be used. However, after the file has been converted using the support tool (JW-300SP), it can be used again.

[2] Setting system memory Setting details for the CF card are as follows.

System memory No. (8)	Set item	Description
		Specify prohibit/allowed saving data to CF card.
#2220	Prohibit to	Bit 7 6 5 4 3 2 1 0 #2220 0 0 0 0 0 0 0 0 Initial value: 00(H)
"	save into CF card	
		Select file type to save and load to CF card. Bit 7 6 5 4 3 2 1 0
#2221	File type selection	#2221 0 0 0 Initial value: 1F(H) User program (including block data and tile of block and sub program.) System memory
		- When the JW-312CU is used, file register cannot be selected.
		For a file selected by #2221, select save (write data from JW300 to CF
		card) or load (read data from CF card to the JW300). Setting (H) Description
#2222	Select save/load	11 Save 22 Load
#2222	to/from CF card	- File name that can be handed by #2222 is "SHARP.***" only. - While the JW300 is operating, data cannot be saved/load to/from a CF
		card. - Before saving data, make sure to format the CF card.
		- After the data is saved/loaded, the set value returns to $00_{(H)}$. (Return to $00_{(H)}$ also when an error occurs)
		With this setting, the JW300 automatically saves data to the CF card at a specified interval. Set the file (type) to save in #2223. 7 6 5 4 3 2 1 0
		#2223
		Execution unit (1 to 127 days)
		Execution flag (1 (ON) Execute 0 (OFF) Does not execute
#2223 #2224	Setting auto save of CF	7 6 5 4 3 2 1 0 #2224 I I I I I I I I I I I I I I I I I I
#2225	card	Hour (0 to 23)
		7 6 5 4 3 2 1 0
		#2225 Initial value: 00(H)
		Minute (0 to 59) - File names are "AT Year Year Month Month Month. ***"
		- When $OO_{(H)}$ is set as execution unit, it will be 128 days. - Set "2224 and #2225 with BCD.
#2226	CF card error code	When an error is found during saving/loading data, an error code corresponding to the error will be stored. => See page 6-23. Even if the error is released, the error code is not cleared. To clear the data, use the support tool (JW-15PG, JW-100SP).
#2230 to #2236	Automatic save file name	When the AUTO LD switch on the JW-3*2CU is OFF and address #2230 or up is specified, the file is automatically saved when inserting the CF card. Specify file types to save on the "#2221." File name will be "@*******" Initial value: 00(H)

[3] Operation procedure to save and load to/from CF card

This chapter describes each operation procedures to save/load files to/from a CF card. Note: While the JW300 is in operation, you cannot save/load files to/from the CF card.

(1) File save procedure

To save files by system memory setting (write from the JW300 to the CF card), follow the procedures below.

[Ex.] Save user program

- 1. Insert an already formatted CF card into the PC card slot of the JW300 control module (JW-3*2CU).
- 2. Set $01_{(H)}$ to system memory #2221.
- 3. Set $11_{(H)}$ to system memory #2222.
- 4. When system memory changes to $00_{(H)}$, the save file is complete.
- => Files with file name "SHARP.ppg" "SHART.pbk" "SHART.ptl" are written to the CF card.

If there is a file having the same name in the CF card, it will be overwritten.

(2) File load procedure

To load files by system memory setting (read from the CF card to the JW300), follow the procedures below Ex.: Load user program.

- 1. Insert an already formatted CF card into the PC card slot of the JW300 control module (JW-3*2CU).
- 2. Stop operation of the JW300 control module.
- 2. Set 01(H) to system memory #2221.

3. Check that files with file name "SHARP.ppg", "SHARP.pbk" and "SHARP.ptl" exist in the CF card.

- (For handling program memory, if all of these files do not exist, the files cannot be loaded.)
- 5. Set 22(H) to system memory #2222.
- 6. When system memory #2222 changes to $OO_{(H)}$, the load is complete.

(3) File auto save procedure

With a specified time, files are saved/loaded to /from the CF card. Time interval is after D7 of system memory #2223 turns ON. Set as follows:

- Ex.: Save user program at 12:30 every 7days.
 - 1. Insert an already formatted CF card into the PC card slot of the JW300 control module (JW-3*2CU).
 - 2. Check the clock of the JW300 control module.
 - 3. Set 01(H) to system memory #2221.
 - 4. Set 12(BCD) to system memory #2224, and 30(BCD) to #2225.
 - 5. Set 87(H) to system memory #2223.
 - 6. For 7 days from step 5 above, files of file name "AT030601.ppg" "AT030601.pbk" "AT030601.ptl" are saved to the CF card. (The file name above is true when June 1st, 2003.)

(4) Auto save/load procedure

1) To load files in the CF card into the JW300 when turning ON the power

While the CF card is inserted into the JW300 control module (JW-3*2CU), if the power is input, if a file having file name "AUTOLOAD" exists in the CF card, and if the AUTO LD switch of the JW-3*2CU is turned ON, the files are loaded from the CF card to the JW300. There is no special setting. All of the files having file name "AUTOLOAD" in the CF card are loaded.

Remarks

When loading the "user program," files of "block data" and "titles of block and sub program" are needed. The user program will not load without them, unless these three files do no exist.

2) When inserting the CF card, save the files of the JW300 into the CF card

When the CF card is inserted into the JW300 control module (JW-3*2CU), and if the AUTO LD switch of the JW-3*2CU is OFF, the files of the JW300 are saved to the CF card. The files to save shall be set in system memory #2221. File names will be those set in system memory "2230 to "2236 and "@******* " If the $00_{(H)}$ is set to system memory #2221, the auto save is not executed. If a file with the same name exists in the CF card, it will be overwritten.

[4] Error code

If an error occurs on the CF card operation, any of the following codes in the table below will be stored in system memory #2226.

Error code(H)	Error detail
00	Normal complete
01	Directory name error of specified file
02	Same file or directory name exists.
03	The specified file does not exist.
05	Specified mode error.
06	File open error.
07	The specified file is already opened.
08	Device read error
09	Device write error
0A	The number of files opened at the same time exceeds the limit.
0B	Disk full.
0D	Specified parameter error.
0E	End of the file was reached.
10	Not supported.
11	Not initialized (relation with the driver and file system).
12	Not initialized (initialization of file system).
13	Is not mounted.
16	Device initialization error.
17	Device mount error.
18	Shorted cache size.
19	Write protected.
63	Others

Chapter 7 List of instructions

7-1 List of basic instructions

Mne- monics	Symbol	Words	Function	Execution condition	Zero	Fla Carry 007356		Non-carry	See page
STR		1	Starts at normally open contact and intermediate result is stored.	Jonandon	007357	00/350	007355	007354	8-2
STR NOT		1	Starts at normally closed contact and intermediate result is stored.						3
AND		1	AND						4
AND NOT		1	AND NOT						4
OR		1	OR						5
OR NOT	<u>}</u>	1	OR NOT						5
AND STR		1	AND with the intermediate result						6
OR STR		1	OR with the intermediate result						7
OUT		1	Outputs result						8
TMR	1) — TMR 2) 3)	2	Timer (decrement) 3) Setting value 1) Start input (starts with ON) 2) TMR number (00000 to 17777) 0.1 to 79.99 sec. 10.1 to 79.99 sec. 0.01 to 79.99 sec. 0.01 to 79.99 sec. 10.1 to 79.99 sec. 0.01 to 79.99 sec. 0.01 to 79.99 sec.	Start input ON					9
DTMR (BCD)	1) — DTMR (BCD) 2) 3)	3	Timer (decrement) 1) Start input (starts with ON) 2) TMR number (00000 to 17777) 3) Setting value (0.1 to 799.9 sec.)	Start input ON					9
DTMR (BIN)	1) — DTMR (BIN) 2) 3)	3	Timer (decrement) 1) Start input (starts with ON) 2) TMR number (00000 to 17777) 3) Setting value (0.1 to 3276.7 sec.)	Start input ON					9
UTMR (BCD)	1) — UTMR (BCD) 2) 3)	3	Timer (increment) 1) Start input (starts with ON) 2) TMR number (00000 to 17777) 3) Setting value (0.1 to 799.9 sec.)	Start input ON					9
UTMR (BIN)	UTMR (BIN) 2) 3)	3	Timer (increment) 1) Start input (starts with ON) 2) TMR number (00000 to 17777) 3) Setting value (0.1 to 3276.7 sec.)	Start input ON					9
CNT	1)	2	Counter (decrement)1) Counter input3) CNT number (00000 to 17777)2) Reset input4) Setting value (1 to 7999)	Counter input					12
DCNT (BCD)	1) — DCNT 2) — (BCD) 3) 4)	3	Counter (decrement)1) Counter input3) CNT number (00000 to 17777)2)Reset input4)Setting value (1 to 7999)	Counter input					12
DCNT (BIN)	1)DCNT 2)(BIN) 3) 4)	3	Counter (decrement)1) Counter input3) CNT number (00000 to 17777)2) Reset input4) Setting value (1 to 32767)	Counter input					12
UCNT (BCD)	1) UCNT 2) (BIN) 3) 4)	3	Counter (increment)1) Counter input3) CNT number (00000 to 17777)2) Reset input4) Setting value (1 to 7999)	Counter					12
UCNT (BIN)	1) UCNT 2) (BCD) 3) 4)	3	Counter (increment)1) Counter input3) CNT number (00000 to 17777)2) Reset input4) Setting value (1 to 32767)	Counter input					12
MD	1)	2	Maintenance display1), 2), 3) input information 6) MD number4) Output direct terminal5) Expansion output7) MD data (000 to 999)	Output display terminal ON					16

The JW300 has the following additional instructions (compared with the JW30H conventional model) related to differentiation, output instructions, and bit operation.

Mne- monics	Symbol	Words	Function	Execution	Zero 007357			Non-carry	See
STR POS	P	1	"a" contact point load rising edge	condition	007357	007356	007355	007354	page 8-17
STR NEG	N	1	"a" contact point load falling edge						18
AND POS	P	1	"a" contact AND rising edge						19
AND NEG	N	1	"a" contact AND falling edge						20
OR POS	P	1	"a" contact OR rising edge						21
OR NEG	N ¯	1	"a" contact OR falling edge						22
OUT POS	——(P)——	1	Rising coil						23
OUT NEG	(N)	1	Falling coil						24
OUT NOT	(/)	1	Inverse output input conditions						25
SET	(S)	1	Set at rising input signal						26
RST	——(R)——	1	Reset at rising input signal						27
PUSH		1	Makes a protected area in the internal memory for the contents of the accumulator and stack.						
POP		1	Recovers the contents of the accumulator and stack from the internal memory area.						29
MRD		1	Temporarily reads the contents of accumulator and stack in the internal memory area.						

7-2 List of application instructions [1] In order of numbers

Mne-	Symbol	Words	Function	Execution		Flag		Non corne	See
monics			Transfers data register to data register	condition	007357	007356	007355	Non-carry 007354	page
F-00	XFER 5 D	3	(1 byte) Transfers data register to data register						10-1
F-00w	XFER 3 D	3	(1 word)						1
F-00d	F-00d S D	3	Transfers data register to data register (2 words)						2
F-01	– F-01 n D BCD n	3	Transfers BCD constant (2 digits)						3
F-01w	-F-01w n D	3	Transfers BCD constant (4 digits)	ſ					3
F-01d	F-01d n D	3	Transfers BCD constant (8 digits)	ſ					4
F-02		3	Exchanges registers (1 byte)	ſ					5
F-02w	F-02w D1 D2	3	Exchanges registers (1 word)	ſ					5
F-02d	F-02d D1 D2	3	Exchanges registers (2 words)	ſ					6
F-03	— F-03 S D	3	Converts 2-digit BCD to 8-bit binary	ſ	0	0	\$	0	7
F-03w	-F-03w -BIN S D	3	Converts 4-digit BCD to 16-bit binary	۱.	0	0	\$	0	7
F-03d	-F-03d -BIN S D	3	Converts 8-digit BCD to 32-bit binary		0	0	\$	0	8
F-04	— F-04 S D	3	Converts 8-bit binary to 2-digit BCD	۱ ۲					9
F-04w	-F-04w S D	3	Converts 16-bit binary to 6-digit BCD	L L					9
F-04d	-F-04d S D	3	Converts 32-bit binary to 12-digit BCD	ſ					10
F-05	-F-05 DMPX S D	3	Demultiplexes 1-byte	ſ					11
F-05w	-F-05w S D	3	Demultiplexes 1-word	ſ					12
F-05d	-F-05d S D	3	Demultiplexes 2-word	ſ					13
F-06	-F-06 S D	3	Multiplexes 1-byte	ſ					14
F-06w	-F-06w S D	3	Multiplexes 1-word	ſ					15
F-06d	-F-06d S D	3	Multiplexes 2-word	ſ					16
F-07	-F-07 n D	3	Transfers 1-byte decimal constant	ſ					17
F-07w		3	Transfers 1-word decimal constant	ſ					17
F-07d	-F-07d n D	3	Transfers 2-word decimal constant						18
F-08	-F-08 n D	3	Transfers 1-byte octal constant	ſ					19
F-08w		3	Transfers 1-word octal constant	ſ					19
F-08d	-F-08d n D	3	Transfers 2-word octal constant	ſ					20
F-09		3	Complements 8-bit data	ſ					21
F-09w	-F-09w S D	3	Complements 16-bit data	ſ					21
F-09d	-F-09d S D	3	Complements 32-bit data	ſ					22

Mne- monics		Syml	bol		Words	Function	Execution condition	Zero 007357	Fla Carry 007356		Non-carry	See page
F-10	F-10 ADD	S1	S2	D	4	Adds register and register (BCD 2 digits)	ſ	\$	\$	\$	\$	10-23
F-10w	F-10w ADD	S1	S2	D	4	Adds register and register (BCD 4 digits)	ſ	\$	\$	\$	\$	24
F-10d	F-10d ADD	S1	S2	D	4	Adds register and register (BCD 8 digits)	ſ	\$	\$	\$	\$	25
Fc10	Fc10 ADD	S1	n	D		Adds register (BCD 2 digits) and constant (2 digits)	ſ	\$	\$	\$	\$	26
Fc10w	Fc10w ADD	S1	n	D		Adds register (BCD 4 digits) and constant (4 digits)	ſ	\$	\$	‡	\$	27
Fc10d	Fc10d ADD	S1	n	D	4	Adds register (BCD 8 digits) and constant (4 digits)	ſ	\$	\$	\$	\$	28
F-11	F-11 SUB	S1	S2	D	4	Subtracts register from register (BCD 2 digits)	ſ	\$	\$	\$	\$	29
F-11w	F-11w SUB	S1	S2	D	4	Subtracts register from register (BCD 4 digits)	ſ	\$	\$	\$	\$	30
F-11d	F-11d SUB	S1	S2	D	4	Subtracts register from register (BCD 8 digits)		\$	\$	‡	\$	31
Fc11	Fc11 SUB	S1	n	D	3	Subtracts constant (BCD 2 digits) from register (2 digits)	ſ	‡	\$	\$	\$	32
Fc11w	Fc11w SUB	S1	n	D	3	Subtracts constant (BCD 4 digits) from register (4 digits)	ſ	\$	\$	\$	\$	33
Fc11d	Fc11d SUB	S1	n	D	3	Subtracts constant (BCD 8 digits) from register (4 digits)	ſ	‡	\$	\$	\$	34
F-12	F-12 CMP	S1	S2		3	Compares register with register (1 byte)	ON	\$	‡	0	\$	35
F-12w	F-12w CMP	S1	S2		3	Compares register with register (1 word)	ON	\$	\$	0	\$	36
F-12d	F-12d CMP	S1	S2		3	Compares register with register (2 words)	ON	\$	\$	0	\$	37
Fc12	Fc12 CMP	S1	n		3	Compares register with octal constant (1 byte)	ON	\$	\$	0	\$	38
Fc12w	Fc12w CMP	S1	n]	3	Compares register with octal constant (1 word)	ON	\$	\$	0	\$	38
Fc12d	Fc12d CMP	S1	n		3	Compares register with octal constant (2 words)	ON	\$	\$	0	\$	39
Fx12	—Fx12 CMP	S1	n]	3	Compares register with hexadecimal constant (1 byte)	ON	\$	\$	0	\$	40
Fx12w	Fx12w CMP	S1	n		3	Compares register with hexadecimal constant (1 word)	ON	ł	ţ	0	ł	40
Fx12d	Fx12d CMP	S1	n]	3	Compares register with hexadecimal constant (2 words)	ON	t	t	0	1	41
F-13	F-13 AND	S	D		3	ANDs register with register (1 byte)	ſ					42
F-13w	F-13w AND	S	D		3	ANDs register with register (1 word)	ſ					42
F-13d	F-13d AND	S	D		3	ANDs register with register (2 words)	ſ					43
Fc13	Fc13 AND	n	D		3	ANDs register with octal constant (1 byte)	ſ					44
Fc13w	Fc13w AND	n	D		3	ANDs register with register (1 word)	ſ					44
Fc13d	Fc13d AND	n	D		3	ANDs register with octal constant (2 words)	ſ					45
Fx13	Fx13 AND	n	D		3	ANDs register with hexadecimal constant (1 byte)	ſ					46
Fx13w	Fx13w AND	n	D		3	ANDs register with hexadecimal constant (1 word)						46
Fx13d	Fx13d AND	n	D]	3	ANDs register with hexadecimal constant (2 words)						47

Mne- monics	Symbo	I	Words	Function	Execution		Fla Carry 007356		Non-carry	See page
F-14	-F-14 S	D	3	ORs register with register (1 byte)	f	001001	007550	001000	001004	10-48
F-14w	-F-14w S OR S	D	3	ORs register with register (1 word)	<u> </u>					48
F-14d	-F-14d S OR S	D	3	ORs register with register (2 words)	ſ					49
Fc14	Fc14 n	D	3	ORs register with octal constant (1 byte)	ſ					50
Fc14w	Fc14w n OR n	D	3	ORs register with octal constant (1 word)	ſ					50
Fc14d	Fc14d n OR n	D	3	ORs register with octal constant (2 words)	ſ					51
Fx14	-Fx14 OR n	D	3	ORs register with hexadecimal constant (1 byte)	ſ					52
Fx14w	—Fx14w n OR n	D	3	ORs register with hexadecimal constant (1 word)	ſ					52
Fx14d	Fx14d n OR n	D	3	ORs register with hexadecimal constant (2 words)	ſ					53
F-15		S2 D	4	Multiplies register by register (BCD 4 digits)	ſ	0	0	\$	0	11-1
F-15d		S2 D	4	Multiplies register by register (BCD 8 digits)	ſ	0	0	\$	0	2
Fc15	Fc15 MUL S1	n D	4	Multiplies register (BCD 4 digits) by constant (BCD 3 digits)	ſ	0	0	\$	0	3
Fc15d	Fc15d MUL S1	n D	4	Multiplies register (BCD 8 digits) by constant (BCD 4 digits)	ſ	0	0	\$	0	4
F-16		S2 D	4	Divides register (BCD 4 digits) by register (BCD 2 digits)	ſ	0	0	\$	0	5
F-16d		S2 D	4	Divides register (BCD 8 digits) by register (BCD 8 digits)	ſ	0	0	\$	0	6
Fc16	Fc16 DIV S1	n D	4	Divides register (BCD 4 digits) by constant (BCD 2 digits)	ſ	0	0	\$	0	7
Fc16d	Fc16d S1	n D	4	Divides register (BCD 8 digits) by constant (BCD 4 digits)	ſ	0	0	\$	0	8
F-17	-F-17 S	D	3	Exclusive NORs register with register (1 byte)	ſ					9
F-17w	-F-17w S	D	3	Exclusive NORs register with register (1 word)	ſ					9
F-17d	-F-17d XNR S	D	3	Exclusive NORs register with register (2 words)	ſ					10
Fc17	Fc17 n XNR n	D	3	Exclusive NORs register with octal constant (1 byte)	ſ					11
Fc17w	Fc17w n XNR n	D	3	Exclusive NORs register with octal constant (1 word)	ſ					11
Fc17d	Fc17d n XNR n	D	3	Exclusive NORs register with octal constant (2 words)	ſ					12
Fx17	- Fx17 XNR n	D	3	Exclusive NORs register with hexadecimal constant (1 byte)	ſ					13
Fx17w	Fx17w n XNR n	D	3	Exclusive NORs register with hexadecimal constant (1 word)	ſ					13
Fx17d	Fx17d n XNR n	D	3	Exclusive NORs register with hexadecimal constant (2 words)	ſ					14
F-18	-F-18 S	D	3	Exclusive ORs register with register (1 byte)	ſ					15
F-18w	-F-18w S XOR S	D	3	Exclusive ORs register with register (1 word)	ſ					15
F-18d	-F-18d XOR S	D	3	Exclusive ORs register with register (2 words)	ſ					16
Fc18	Fc18 n XOR n	D	3	Exclusive ORs register with octal constant (1 byte)	ſ					17
Fc18w	Fc18w n XOR n	D	3	Exclusive ORs register with octal constant (1 word)	ſ					17
Fc18d	Fc18d n XOR n	D	3	Exclusive ORs register with octal constant (2 words)	ſ					18

Mne- monics	Symbol	Words	Function	Execution condition		Flag 267357 669336 6673355 10 1 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1		Non-carry	See page
Fx18	- Fx18 n D	3	Exclusive ORs register with hexadecimal constant(1 byte)	F					11-19
Fx18w	- Fx18w n D	3	Exclusive ORs register with hexadecimal constant(1 word)						19
Fx18d	- Fx18d n D	3	Exclusive ORs register with hexadecimal constant(2 words)						20
F-20	$\begin{array}{c} 1) & \\ 2) & \\ 3) & \\ 4) & \\ \end{array} \begin{array}{c} (F-20) \\ 6) \\ \end{array} \begin{array}{c} 7) \\ 7) \\ \end{array}$	2	Maintenance display 1), 2), 3) Input information 4) Output direct terminal 5) Expansion output	Output display terminal ON					21
F-21	-F-21 S D	3	Obtains square root of register (BCD 8 digits)	ſ	0	0	\$	0	22
F-22	F-22 S D	3	Executes trigonometric function (SIN)	ſ	0	\$	\$	\$	22
F-23	F-23 S D	3	Executes trigonometric function (COS)	ſ	0	\$	\$	\$	23
F-24	F-24 S D	3	Executes trigonometric function (TAN)	ſ	0	\$	\$	\$	24
F-25	F-25 ASIN S D	3	Executes trigonometric function (SIN ⁻¹)	ſ	0	\$	\$	\$	25
F-26	-F-26 ACOS S D	3	Executes trigonometric function (COS ⁻¹)	ſ	0	\$	\$	\$	26
F-27	-F-27 ATAN S D	3	Executes trigonometric function (TAN ⁻¹)		0	\$	\$	\$	27
F-28	- $F-28$ S D	3	Exchanges the rectangular coordinate system (X,Y) data with polar coordinate system (γ , θ)	f	0	0	\$	0	28
F-29	-F-29 →XY S D	3	Exchanges the polar coordinate system (γ , θ) data with rectangular coordinate system(χ , γ)		0	0	\$	0	29
F-30	F-30 MCS	1	Sets master control	ON			,		30
F-31		1	Resets master control						30
F-32	-F-32 SET OUT	2	Sets coil	ON					33
F-33	-F-33 RST OUT	2	Resets coil	ON					34
F-34	-F-34 TSET N1 N2 BIT	4	Compares with current value of clock (specified relay set)	ON					36
F-35	-F-35 TRST N1 N2 BIT	4	Compares with current value of clock (specified relay reset)	ON					37
F-36	F-36 TADD S1 S2 D	4	Adds time	ſ	t	t	t	f	38
F-37		4	Subtracts time	ſ	\$	\$	\$	\$	39
F-38	-F-38 TXFR D	2	Transfers current value of clock	ſ					40
F-40		1	END instruction						41
F-41		1	Sets jump control	OFF					42
F-42	F-42 JCR	1	Resets jump control						42
F-43	F-43	1	Complements bit (ACC contents)						44
F-44	F-44	1	Differentiates at ON	ſ					45
F-45	F-45 ↓	1	Differentiates at OFF	Ţ					46
F-47	-F-47 ONLS	1	Sets level operating condition						47
F-48		1	Resets level operating condition						47
F-49		1	Conditional END	OFF					48

Mne- monics	Symbol	v	Nords	Function	Execution condition	Zero 007357	Fla	g Error 007355	Non-carry 007354	See page
F-50			3	Decodes from 4 to 16	ſ		001000			12-1
F-51			3	Encodes from 16 to 4	ſ					1
F-52	-F-52 7SEG S D		3	Decodes to 7-segment data	ſ					2
F-53			3	Converts 4-digit BCD to 16-bit binary	ſ	0	0	\$	0	3
F-54			3	Converts 16-bit binary to 6-digit BCD	ſ					3
F-55	-F-55 SWAP S D		3	Swaps upper 4 bits with lower 4 bits	ſ					4
F-56	-F-56 S D		3	Complement of 10 of 1-byte data	ſ	0	0	\$	0	5
F-56w	F-56w S D		3	Complement of 10 of 1-word data	ſ	0	0	\$	0	5
F-56d	-F-56d S D		3	Complement of 10 of 2-word data	ſ	0	0	↑	0	6
F-57	F-57 2NEG S D		3	Complement of 2 of 1-byte data	ſ					7
F-57w	-F-57w S D 2NEG S D		3	Complement of 2 of 1-word data	ſ					7
F-57d	-F-57d S D		3	Complement of 2 of 2-word data	ſ					8
F-58	F-58 ∑BIT n S	D	4	Total of ON bits	ſ					9
F-60	1) 2) — F-60 3) — SFR D 4) —		2	Shifts register bi-directionally (1 byte)1) Shift direction input3) Shift input2) Data input4) Reset input	Shift input	\$	\$	0	\$	10
F-60w	1)		2	Shifts register bi-directionally (1 word)1) Shift direction input3) Shift input2) Data input4) Reset input	Shift input	\$	\$	0	\$	12
F-60d	1) — 2) — F-60d 3) — SFR D 4) —		2	Shifts register bi-directionally (2 words)1) Shift direction input3) Shift input2)Data input4) Reset input	Shift input	\$	\$	0	\$	13
F-61	1) — F-61 2) — ASFR D		2	Shifts register asynchronously (1 byte) 1) Shift direction input 2) Shift input	Shift input ON	0	\$	0	\$	14
F-61w	1) — F-61w D 2) — ASFR D		2	Shifts register asynchronously (1 word) 1) Shift direction input 2) Shift input	Shift input ON	0	\$	0	\$	15
F-61d	1) — F-61d D 2) — ASFR D		2	Shifts register asynchronously (2 words) 1) Shift direction input 2)Shift input	Shift input ON	0	\$	0	\$	16
F-62	1) F-62 2) V/DC D		2	2-digit BCD up/down counter 1) Up/down counter direction input 2) Counter input 3) Reset input	Counter input	\$	\$	\$	\$	17
F-62w	1)		2	 4-digit BCD up/down counter 1) Up/down counter direction input 2) Counter input 3) Reset input 	Counter input	\$	\$	\$	ţ	18
F-62d	1) 2) 3) U/DC D		2	8-digit BCD up/down counter 1) Up/down counter direction input 2) Counter input 3) Reset input	Counter input	\$	\$	\$	ţ	19
F-63	F-63 D INC D		2	Increments binary counter (1 byte)		\$	\$	0	\$	20
F-63w	F-63w INC D		2	Increments binary counter (1 word)	ſ	\$	\$	0	\$	20
F-63d	F-63d D		2	Increments binary counter (2 words)	ſ	\$	\$	0	‡	21
F-64	-F-64 D DEC D		2	Decrements binary counter (1 byte)	ſ	ţ,	‡ •	0	\$	22
F-64w	– F-64w D DEC D		2	Decrements binary counter (1 word)	ſ	\$	\$	0	\$	22
F-64d			2	Decrements binary counter (2 words)	ſ	ţ	ţ	0	\$	23

Mne- monics	Symbol	Wor	ds Function	Execution condition		Fla	g 007355	Non-carry 007354	See page
F-65	-F-65 D BCDI D	2	BCD increment counter (1 byte)	ſ	\$	\$	\$	\$	12-24
F-65w	F-65w BCDI D	2	BCD increment counter (1 word)	ſ	\$	\$	\$	\$	24
F-65d		2	BCD increment counter (2 words)	ſ	\$	\$	\$	\$	25
F-66	— F-66 D BCDD D	2	BCD decrement counter (1 byte)	ſ	\$	\$	\$	\$	26
F-66w	F-66W D BCDD D	3	BCD decrement counter (1 word)	ſ	\$	\$	\$	\$	26
F-66d		3	BCD decrement counter (2 words)	ſ	\$	\$	\$	\$	27
F-67	-F-67 NSFH n D	3	Digit shift (up)	ſ					28
F-68	-F-68 NSFL n D	4	Digit shift (down)	ſ					28
F-69	-F-69 NXFR S D	4	Digit transfer	ſ					29
F-70	F-70 FILE n S C	2 4	Transfers "n" bytes in block	ſ					30
F-70w	F-70W n S C	9 4	Transfers "n" words in block	ſ					31
F-70d	F-70d n S C	9 4	Transfers "n" double word in block	ſ					32
F-71	-F-71 n D1 D	2 4	Transfers octal constant in block (1 byte)	5					33
F-71w	-F-71w n D1 D	2 4	Transfers octal constant in block (1 word)	ſ					34
F-71d	-F-71d n D1 D	2 4	Transfers octal constant block (2 words)	ſ					34
F-72	-F-72 DMPX n S D	9 4	Demultiplexes "n" bytes to file 1 register	ſ					35
F-72w	—F-72w n S C	9 4	Demultiplexes "n" words to file 1 register	ſ					36
F-72d	—F-72d n S DMPX	2 4	Demultiplexes "n" double word to file 1 register	ſ					37
F-73	-F-73 n S D	2 4	Multiplexes "n" bytes from file 1 register	ſ					38
F-73w	—F-73w n S D	9 4	Multiplexes "n" words from file 1 register	ſ					39
F-73d	—F-73d n S D	9 4	Multiplexes "n" double word from file 1 register	ſ					40
F-74	-F-74 n S D	9 4	Transfers "n" bytes	ſ					41
F-74w	-F-74w n S D	9 4	Transfers "n" words	ſ					41
F-74d	-F-74d n S C	9 4	Transfers "n" double word	ſ					42
F-76		9 4	Transfers "n" bytes in block	ſ					43
F-76w	— F-76w S1 S2 D	9 4	Transfers "n" words in block	f					44
F-76d		9 4	Transfers "n" double word in block						45
F-77		4	Creates sum check code	ſ					46
F-78		3 4	Checks data		0	0	\$	0	47
F-79		2 4	Sorts register (1 byte) data	ſ					48
F-79w	-F-79w S1 N1 N SORT S1 N1 N	2 4	Sorts register (1 word) data	ſ					49
F-79d	-F-79d S1 n1 n	2 4	Sorts register (2 words) data	-					50

Mne- monics	Symbol	Words	Function	Execution condition	Zero 007357		ag Error 007355	Non-carry 007354	See page
F-80	-F-80 IORF R,S	2	I/O refresh	ON	0	\$	\$	\$	13-1
F-82	-F-82 IORF SW	2	Special I/O refresh	ON	0	\$	\$	\$	2
F-85		4	Read from special I/O	ſ	0	\$	\$	\$	3
F-86		4	Write to special I/O	ſ	0	\$	\$	\$	3
F-90	- F-90 n REM n	2	Remark n = 0000 to 3777						4
F-91		4	Transfers BCD constant (8 digits)	ſ					5
F-97		4	Transfers a decimal constant (8 digits)	ſ					6
F-100	-F-100 ADRS S D	3	Sets indirect address	ſ					7
F-101	F-101 n file N D	4	Sets indirect address	ſ					8
F-102	F-102 n file N D	4	Reads from the register of a direct address (1 byte)	ſ					9
F-102w	-F-102w n file N D	4	Reads from the register of a direct address (1 word)	ſ					9
F-102d	F-102d n file N D	4	Reads from the register of a direct address (2 words)	ſ					10
F-103	-F-103 S n file N	4	Writes in the register of a direct address (1 byte)	ſ					11
F-103w	-F-103w MWR S n file N	4	Writes in the register of a direct address (1 word)	ſ					11
F-103d	-F-103d S n file N	4	Writes in the register of a direct address (2 words)	ſ					12
F-112		4	Compares "n" bytes (between 1-byte registers)	ON	\$	\$	0	\$	13
F-112w		3	Compares "n" words	ON	\$	\$	0	\$	14
F-112d	F-112d S1 S2 S3	3	Compares "n" double word	ON	\$	\$	0	\$	15
F-116		3	Divides register value (BCD 8 digits) by another register value (BCD 8 digits) (4 decimal places)	ſ	0	0	\$	0	16
F-130	- F-130 BIT → S1 S2	3	Multiplexes bit (indirect address)	ON	0	\$	0	0	17
F-131	— F-131 n S	2	Multiplexes bit (direct address)	ON	0	\$	0	0	17
F-132	1) - F-132 2) - S/R S D	2	Sets/resets bits (indirect address) 1) Sets/resets direction input 2) Input condition	ON					18
F-133	1) - F-133 n D 2) - S/R n D	2	Sets/resets bits (direct address) 1) Sets/resets direction input 2) Input condition	ON					18
F-140	-F-140 LABL LBn	1	Sets label LB0000 to LB1377						19
F-141	- F-141 JMP LBn	2	Jumps to label	ON					20
F-142	- F-142 CALL LBn	1	Calls labeled subroutine	ſ					22
F-143	F-143 RET	2	Returns from subroutine						22
F-144		1	Sets loop count	ſ					24
F-145	F-145NEXT	2	END of loop						24
F-146	-F-146 S FORR S	1	Sets loop count register	ſ					26
F-147	F-147 FXIT		Exits loop (conditional)	OFF					27
F-148			Calls subroutine by label	ſ					28
F-149	F-149 RETC		Retuens from subroutine (conditional)	OFF					29

Mne- monics		:	Symb	ool		Words	Function	Execution condition	Zero 007357	Flag		Non-carry 007354	See page
F-151		F-151 JMP+	LBn	S		3	Jumps to labeled program address	ON					13-30
F-153	_	F-153 → BIN	S	D		3	Converts 8-digit BCD to 32-bit binary	ſ	0	0	\$	0	31
F-154	[F-154 → BCD	S	D		3	Converts 32-bit binary to 10-digit BCD	ſ					31
F-155	_[F-155 → SEC	S	D		3	Convert hours (BCD 4 digits), minutes (BCD 2 digits) and seconds (BCD 2 digits) into seconds (BCD 8 digits)	ſ	0	0	\$	0	32
F-156	_[F-156 →HMS	S	D		3	Convert seconds (BCD 8 digits) into hours (BCD 4 digits), minutes (BCD 2 digits), and seconds (BCD 2 digits)	ſ	0	0	\$	0	33
F-160		F-160 NSFR	S1	S2	D	4	Shifts register bi-directionally (n bit)1) Shift direction input3) Shift input2) Data input4) Reset input(Shift is 1 bit) (S1)=0 to 256(S2)=0 to 7	Shift input	\$	‡	0	\$	34
Fc160		Fc160 NSFR	n₁	n ₂	D	4	Shift register bi-directionally (n bit)1) Shift direction input3) Shift input2) Data input4) Reset input(Shift is 1 bit) N1=0 to 377 N2=0 to 7	Shift input	\$	‡	0	\$	35
F-161	1) 2)	F-161 NASR	D	n		3	Shifts register asynchronously (n bytes) 1) Shift direction input 2) Shift input	Shift input ON	0	\$	0	\$	36
F-161w	1) — 2) —	F-161w NASR	D	n		3	Shifts register asynchronously (n words) 1) Shift direction input 2) Shift input	Shift input ON	0	\$	0	\$	38
F-161d	1) — 2) —	F-161d NASR	D	n		3	Shifts register asynchronously (n double words)	Shift input ON	0	\$	0	\$	39
F-163		F-163 INC2	D			2	Adds binary (+2) counter (1 byte)	ſ	\$	\$	0	\$	40
F-163w		F-163w INC2	D			2	Adds binary (+2) counter (1 word)	ſ	\$	\$	0	\$	40
F-163d		F-163d INC2	D			2	Adds binary (+2) counter (2 words)	ſ	\$	ţ	0	\$	41
F-164		F-164 DEC2	D			2	Subtracts binary (-2) counter (1 byte)	ſ	\$	\$	0	\$	42
F-164w		F-164w DEC2	D			2	Subtracts binary (-2) counter (1 word)	ſ	\$	\$	0	\$	42
F-164d	[F-164d DEC2	D			2	Subtracts binary (-2) counter (2 words)	ſ	\$	ţ	0	\$	43
F-170		F-170 INS	S	D1	D2	4	Inserts data (1 byte)	ſ	0	0	\$	0	44
F-170w	_[F-170w INS	S	D1	D2	4	Inserts data (1 word)	ſ	0	0	\$	0	45
F-170d		F-170d INS	S	D1	D2	4	Inserts data (2 words)	ſ	0	0	\$	0	46
F-171	_[F-171 DEL	S1	S2	S3	4	Deletes data (1 byte)	ſ	0	0	\$	0	47
F-171w		F-171w DEL	S1	S2	S3	4	Deletes data (1 word)	ſ	0	0	\$	0	48
F-171d	-[F-171d DEL	S1	S2	S3	4	Deletes data (2 words)	<u> </u>	0	0	\$	0	49
F-172	_	F-172 SRCH	S	D1	D2	4	Searches data (1 byte)	ſ	0	0	\$	0	50
F-172w	_[F-172w SRCH	S	D1	D2	4	Searches data (1 word)	ſ	\$	\$	0	0	51
F-172d	_	F-172d SRCH	S	D1	D2	4	Searches data (2 words)	ſ	\$	\$	0	0	52
F-173	1) — 2) —	F-173 CHNG	S	D1	D2	4	Changes data (1 byte) 1) Mode direction 2) Execution input	Execution input	\$	\$	0	0	53
F-173w	1)	F-173w CHNG	S	D1	D2	4	Changes data (1 word) 1) Mode direction 2) Execution input	Execution input	\$	‡	0	0	54
F-173d		F-173d CHNG	S	D1	D2	4	Changes data (2 words)	Execution input	\$	\$	0	0	55
F-174	_	F-174 VREV	D	n		3	Reverse order of register data (1 byte)	ſ					14-1
F-175		F-175 NSWP	D	n	-	3	Swaps upper 4 bits with lower 4 bits of registers	ſ					1

Mne-	Sym	hol	Words	Function	Execution		Flaç			See
monics			words	Reads data from register of specified	condition	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	page
F-176	- F-176 DFRD S	file N D	4	address (256 bytes)						14-2
F-177	-F-177 DFWR S	D file N	4	Writes data into register of specified address (256 bytes)						3
F-180		S2 BIT	4	Compare between register and register (1 byte)(>, with relay output)	ON	0	0	0	0	4
F-180w		S2 BIT	4	Compare between register and register (1 word)(>, with relay output)	ON	0	0	0	0	5
F-180d	F-180d CP> S1	S ₂ BIT	4	Compare between register and register (2 words)(>, with relay output)	ON	0	0	0	0	6
Fc180	— Fc180 CP> S	n BIT	4	Compare register with constant (1 byte)(>, with relay output)	ON	0	0	0	0	7
Fc180w	-Fc180w CP> S	n BIT	4	Compare register with constant (1 word)(>, with relay output)	ON	0	0	0	0	8
Fc180d	Fc180d CP> S	n BIT	4	Compare register with constant (2 words)(>, with relay output)	ON	0	0	0	0	9
F-181		S2 BIT	4	Compare between register and register (1 byte)(<, with relay output)	ON	0	0	0	0	4
F-181w		S ₂ BIT	4	Compare between register and register (1 word)(<, with relay output)	ON	0	0	0	0	5
F-181d		S2 BIT	4	Compare between register and register (2 words)(<, with relay output)	ON	0	0	0	0	6
Fc181	— Fc181 CP< S	n BIT	4	Compare register with constant (1 byte)(<, with relay output)	ON	0	0	0	0	7
Fc181w	Fc181w CP< S	n BIT	4	Compare register with constant (1 word)(<, with relay output)	ON	0	0	0	0	8
Fc181d	Fc181d CP< S	n BIT	4	Compare register with constant (2 words)(<, with relay output)	ON	0	0	0	0	9
F-182		S2 BIT	4	Compare between register and register (1 byte)(<, with relay output)	ON	0	0	0	0	4
F-182w	F-182w CP= S1	S ₂ BIT	4	Compare between register and register (1 word)(<, with relay output)	ON	0	0	0	0	5
F-182d	F-182d CP= S1	S2 BIT	4	Compare between register and register (2 words)(<, with relay output)	ON	0	0	0	0	6
Fc182	— Fc182 CP= S	n BIT	4	Compare register with constant (1 byte)(<, with relay output)	ON	0	0	0	0	7
Fc182w	Fc182w CP= S	n BIT	4	Compare register with constant (1 word)(<, with relay output)	ON	0	0	0	0	8
Fc182d	Fc182d CP= S	n BIT	4	Compare register with constant (2 words)(<, with relay output)	ON	0	0	0	0	9
F-183		S ₂ BIT	4	Compare between register and register (1 byte)(=, with relay output)	ON	0	0	0	0	4
F-183w	F-183w CP>= S1	S2 BIT	4	Compare between register and register (1 word)(=, with relay output)	ON	0	0	0	0	5
F-183d	F-183d CP>= S1	S2 BIT	4	Compare between register and register (2 words)(=, with relay output)	ON	0	0	0	0	6
Fc183	Fc183 CP>= S	n BIT	4	Compare register with constant (1 byte)(=, with relay output)	ON	0	0	0	0	7
Fc183w	Fc183w CP>= S	n BIT	4	Compare register with constant (1 word)(=, with relay output)	ON	0	0	0	0	8
Fc183d	Fc183d CP>= S	n BIT	4	Compare register with constant (2 words)(=, with relay output)	ON	0	0	0	0	9
F-184		S2 BIT	4	Compare between register and register (1 byte)(≧, with relay output)	ON	0	0	0	0	4
F-184w	F-184w CP<= S1	S2 BIT	4	Compare between register and register (1 word)(≧, with relay output)	ON	0	0	0	0	5
F-184d		S ₂ BIT	4	Compare between register and register (2 words)(≧, with relay output)	ON	0	0	0	0	6
Fc184	— Fc184 CP<= S	n BIT	4	Compare register with constant (1 byte)(≧, with relay output)	ON	0	0	0	0	7
Fc184w	Fc184w CP<= S	n BIT	4	Compare register with constant (1 word)(≧, with relay output)	ON	0	0	0	0	8
Fc184d	Fc184d CP<= S	n BIT	4	Compare register with constant $(2 \text{ words})(\geq)$, with relay output)	ON	0	0	0	0	9

Mne- monics	Symbol	Words	Function	Execution condition	Zero 007357	Fla	g Error 007355	Non-carry 007354	See page
F-185		4	Compare between register and register (1 byte)(≤, with relay output)	ON	0	0	0	0	14-4
F-185w] 4	Compare between register and register (1 word)(≶, with relay output)	ON	0	0	0	0	5
F-185d] 4	Compare between register and register (2 words)(≶, with relay output)	ON	0	0	0	0	6
Fc185	-Fc185 CP<> S n BIT] 4	Compare register with constant (1 byte)(≶, with relay output)	ON	0	0	0	0	7
Fc185w	-Fc185w CP<> S n BIT] 4	Compare register with constant $(1 \text{ word})(\leq, \text{ with relay output})$	ON	0	0	0	0	8
Fc185d	- Fc185d CP<> S n BIT] 4	Compare register with constant (2 words)(≶, with relay output)	ON	0	0	0	0	9
F-202	F-202 UN,C, file N n] 4	Open channel (with octal station number)	ON					10
F-203	F-203 UN,C, OPCH ST file N n] 4	Open channel (with hex. station number)	ON					10
F-204	F-204 n S	3	Sends data	ſ	\$	\$	‡	\$	11
F-205	— F-205 n D	3	Receives data	ſ	\$	\$	‡	\$	12
F-206	-F-206 EOP1 UN1, CH ST1 UN2] 4	Open channel 1 (set the hierarchical communication)	ON					13
F-207	-F-207 EOP2 ST2 file N n] 4	Open channel 2 (set the hierarchical communication)	ON					13
F-210] 4	Adds register and register in binary (8 bits + 8 bits)	ſ	\$	\$	0	\$	14
F-210w] 4	Adds register and register in binary (16 bits + 16 bits)	ſ	\$	\$	0	\$	15
F-210d] 4	Adds register and register in binary (32 bits + 32 bits)	ſ	\$	\$	0	\$	16
Fc210	-Fc210 S1 n D] 4	Adds register and constant in binary (8 bits + 8 bits)	ſ	\$	\$	0	\$	17
Fc210w	-Fc210w S1 n D] 4	Adds register and constant in binary (16 bits + 16 bits)	ſ	\$	\$	0	\$	17
Fc210d	-Fc210d S1 n D	4	Adds register and constant in binary (32 bits + 16 bits)	ſ	\$	\$	0	\$	18
F-211		4	Subtracts register from register in binary (8 bits - 8 bits)	ſ	\$	\$	0	\$	19
F-211w] 4	Subtracts register from register in binary (16 bits - 16 bits)	ſ	\$	\$	0	\$	20
F-211d] 4	Subtracts register from register in binary (32 bits - 32 bits)	ſ	\$	\$	0	\$	20
Fc211] 4	Subtracts constant from register in binary (8 bits - 8 bits)	ſ	\$	\$	0	\$	21
Fc211w	-Fc211w S1 n D] 4	Subtracts constant from register in binary (16 bits - 16 bits)	ſ	\$	\$	0	\$	21
Fc211d	Fc211d SUB S1 n D] 4	Subtracts constant from register in binary (32 bits - 16 bits)	ſ	\$	†	0	\$	22
F-212] 4	Window comparator (between 1-byte registers)	ON	\$	\$	\$	\$	23
F-212w] 4	Window comparator (between 1-word registers)	ON	\$	\$	\$	\$	24
F-212d] 4	Window comparator (between 2-word registers)	ON	\$	\$	\$	\$	25
Fc212	- Fc212 S1 N1 N2] 4	Window comparator (between 1-byte octal constants)	ON	\$	\$	\$	\$	26
Fc212w] 4	Window comparator (between 1-word octal constants)	ON	\$	\$	\$	\$	26
Fc212d] 4	Window comparator (between 2-word octal constants)	ON	\$	\$	\$	\$	27
Fx212] 4	Window comparator (between 1-byte hexadecimal constants)	ON	\$	\$	\$	\$	28
Fx212w] 4	Window comparator (between 1-word hexadecimal constants)	ON	\$	\$	\$	\$	28
Fx212d		4	Window comparator (between 2-word hexadecimal constants)	ON	\$	‡	\$	\$	29

Mne-	Or which	Manda	Franklan	Execution		Fla	g		See
monics	Symbol	Words	Function	condition	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	page
F-215		4	Multiplies register by register in binary (8 bits x 8 bits)	ſ	0	0	0	0	14-30
F-215w		4	Multiplies register by register in binary (16 bits x 16 bits)	ſ	0	0	0	0	30
F-215d		4	Multiplies register by register in binary (32 bits x 32 bits)	ſ	0	0	0	0	31
Fc215	— Fc215 S1 N D	4	Multiplies register by constant in binary (8 bits x 8 bits)	ſ	0	0	0	0	32
Fc215w	-Fc215w S1 N D	1	Multiplies register by constant in binary (16 bits x 16 bits)	ſ	0	0	0	0	32
Fc215d	-Fc215d S1 N D	1	Multiplies register by constant in binary (32 bits x 16 bits)		0	0	0	0	33
F-216		4	Divides register by register in binary (8 bits \div 8 bits)	ſ	0	0	\$	0	34
F-216w		4	Divides register by register in binary (15 bits \div 15 bits)	ſ	0	0	\$	0	35
F-216d		4	Divides register by register in binary (31 bits \div 31 bits)	ſ	0	0	\$	0	36
Fc216	-Fc216 S1 n D	4	Divides register by constant in binary (8 bits \div 8 bits)	ſ	0	0	\$	0	37
Fc216w	- Fc216w S1 n D	4	Divides register by constant in binary (15 bits \div 15 bits)	ſ	0	0	\$	0	38
Fc216d	-Fc216d S1 n D	1	Divides register by constant in binary (31 bits \div 15 bits)	ſ	0	0	‡	0	38
F-231	F-231 MCRN	1	Master control reset nesting						39
F-242	F-242 JCRN	4	Jump control reset nesting						40
F-252	$- \begin{bmatrix} F-252 \\ \rightarrow ASC \end{bmatrix} S n D$	4	Converts HEX code into ASCII code	ſ					41
F-253	— F-253 S n D	4	Converts ASCII code into HEX code	ſ	0	0	\$	0	42
F-260	F-260 RTMR S D BIT	4	Subtracts timer (setting value, register address)	ON	0	0	\$	0	43
Fc260	Fc260 n D BIT	4	Subtracts timer (constant, register address)	ON	0	0	\$	0	44
F-261	1) - F-261 2) - RCNT S D BIT	4	Subtracts counter (setting value, register address) 1) Counter input 2) Reset input	Counter input	0	0	\$	0	45
Fc261	1) - Fc261 n D BIT 2) - RCNT n D BIT	4	Subtracts counter (constant, register address) 1) Counter input 2)Reset input	Counter input	0	0	\$	0	46
F-263		2	Increments counter by 4 (1-byte binary)	ſ	\$	\$	0	\$	47
F-263w	F-263w D INC4 D	2	Increments counter by 4 (1-word binary)	ſ	\$	\$	0	\$	47
F-263d	F-263d INC4 D	2	Increments counter by 4 (2-word binary)		\$	\$	0	\$	48
F-264	— F-264 DEC4 D	2	Decrements counter by 4 (1-byte binary)		\$	\$	0	\$	49
F-264w	F-264W D DEC4 D	2	Decrements counter by 4 (1-word binary)	ſ	\$	\$	0	\$	49
F-264d		2	Decrements counter by 4 (2-word binary)	ſ	\$	\$	0	\$	50

Mne-	Symbol	Warda	Function	Execution		FI	ag		See
monics	Symbol	Words	Function	condition	Zero 07357	Carry 07356	Error 07355	Non-carry 07354	page
F-300		3	Transfers 1-byte data	ON					14-51
F-300w	-F-300w XFER S D	3	Transfers 1-word data	ON					51
F-300d	-F-300d XFER S D	3	Transfers 2-word data	ON					52
F-310	F-310 S1 S2 D	4	Adds registers in binary with sign (31 bits + 31 bits)	ſ	‡	‡	\$	\$	53
F-311	F-311 SSUB S1 S2 D	4	Subtracts registers in binary with sign (31 bits - 31 bits)		‡	‡	\$	\$	54
F-315		4	Multiplies registers in binary with sign (31 bits x 31 bits)	_ _	0	0	0	0	55
F-316		4	Divides registers in binary with sign (31 bits \div 31 bits)		0	0	\$	0	56
F-403	F-403 LOG	1	Logging instruction						56
NOP		1	Non-operation instruction						

[2] Classification by operation

Туре			Mnemonics	See page			Туре		Mnemonics	See page
			F-00	10-1			Register	2 digits + 2 digits	F-10	10-23
		1 byte	F-300	14-51			from	4 digits + 4 digits	F-10w	24
			F-00w	10-1		BCD	register	8 digits + 8 digits	F-10d	- 25
		1 word	F-300w	14-51		addition	-	2 digits + 2 digits	Fc10	26
			F-00d	10-2		addition	Constant	4 digits + 4 digits	Fc10w	27
		2 words	F-000				from	8 digits + 4 digits		28
		n bytes		14-52			register		Fc10d	
			F-70	12-30			Register	2 digits - 2 digits	F-11	29
		n words	F-70w	31		BCD	from	4 digits - 4 digits	F-11w	30
	_	n double word	F-70d	32		subtra-	register	8 digits - 8 digits	F-11d	31
	Register to register	n bytes	F-76	43		ction	Constant	2 digits - 2 digits	Fc11	32
	transfer	(Indirect assignment)					from	4 digits - 4 digits	Fc11w	33
		n words	F-76w	44			register	8 digits - 4 digits	Fc11d	34
		(Indirect assignment)					Register from	4 digits x 4 digits	F-15	11-1
		n double word (Indirect assignment)	F-76d	45		BCD multipli-	register	8 digits x 8 digits	F-15d	2
						cation	Constant	4 digits x 3 digits	Fc15	3
		n bytes (Same data)	F-74	41		Callon	from register	8 digits x 4 digits	Fc15d	4
		, ,					Ŭ	4 digits ÷ 2 digits	F-16	5
		n words	F-74w	41			Register	8 digits ÷ 8 digits	F-16d	6
		(Same data)				BCD	from	8 digits \div 8 digits	1 100	-
		n double word (Same data)	F-74d	42		division	register	(Decimal fraction 4 digits)	F-116	13-16
		. ,		10.0			Constant	4 digits \div 2 digits	Fc16	11-7
	BCD	2 digits	F-01	10-3			from	8 digits \div 4 digits	Fc16d	8
	constant	4 digits	F-01w	3			register			14-14
	transfer	8 digits	F-01d	4			Register	8 bits + 8 bits 16 bits + 16 bits	F-210	
		o digito	F-91	13-5	Arithmetic		from		F-210w	15
		1 byte	F-07	10-17	operation	Binary	register	32 bits + 32 bits	F-210d	16
Transfer	Decimal	1 word	F-07w	17	instructions	addition	Constant	8 bits + 8 bits	Fc210	17
nstructions	constant transfer	2 words	F-07d	18		uuuuuu	from	16 bits + 16 bits	Fc210w	17
	landici	8 digits	F-97	13-6			register	32 bits + 16 bits	Fc210d	18
		1 byte	F-08	10-19			Register from register	31 bits + 31 bits	F-310	53
		1 word	F-08w	19			(with code)	01 513 1 01 513		
	Octal	2 words	F-08d	20			Register	8 bits - 8 bits	F-211	19
	constant	n bytes	F-71	12-33			from	16 bits - 16 bits	F-211w	20
	transfer	n words		34		Binary	register	32 bits - 32 bits	F-211d	20
			F-71w			subtra-	Constant	8 bits - 8 bits	Fc211	21
		n double word	F-71d	34		ction	from	16 bits - 16 bits	Fc211w	21
		1 byte	F-05	10-11			register	32 bits - 16 bits	Fc211d	22
		1 word	F-05w	12			Register from register			E A
		2 words	F-05d	13			(with code)	31 bits - 31 bits	F-311	54
	Demultiplex	n bytes (file 1)	F-72	12-35			Register	8 bits x 8 bits	F-215	30
							from	16 bits x 16 bits	F-215w	30
		n words (file 1)	F-72w	36			register	32 bits x 32 bits	F-215d	31
						Binary	Constant	8 bits x 8 bits	Fc215	32
		n double word (file 1)	F-72d	37		division	from	16 bits x 16 bits	Fc215w	32
		d huda	F 00	10.14		annoion	register	32 bits x 16 bits	Fc215d	33
		1 byte	F-06	10-14			Register from		102100	
		1 word	F-06w	15			Register from register (with code)	31 bits x 31 bits	F-315	55
		2 words	F-06d	16				8 bits ÷8 bits	F-216	34
		n bytes (file 1)	F-73	12-38			Register		1	35
	Multiplex	-,,	170				from	$15 \text{ bits} \div 15 \text{ bits}$	F-216w	
		n words (file 1)	F-73w	39		Binary	register	$\begin{array}{c} 31 \text{ bits} \div 31 \text{ bits} \\ 8 \text{ bits} \div 8 \text{ bits} \end{array}$	F-216d	36
						multipli-	Constant		Fc216	37
		n double word (file 1)	F-73d	40		cation	from	15 bits \div 15 bits	Fc216w	37
		4 1-14-					register	31 bits \div 15 bits	Fc216d	38
	Digit transfer	4 bits	F-69	29			Register from register (with code)	31 bits \div 31 bits	F-316	56
		1 byte	F-102	13-9			(with code)		1 010	00
	Read out	1 word	F-102w	9						
	the file	2 words	F-102d	10						
				110						
		256 bytes	F-176	14-2						
		256 bytes 1 byte	F-176 F-103	14-2						
		1 byte	F-103							
	Write to file			13-11						

		Туре		Mnemonics	See page			
		Register	8 bits	F-13	10-42			
		from	16 bits	F-13w	42			
	Binary	register	32 bits	F-13d	43			
	multipli-	Register	8 bits	Fc13	44			
	cation	with octal	16 bits	Fc13w	44			
	- Callon	constant	32 bits	Fc13d	45		c	
		Register	8 bits	Fx13	46			
		with hexadecimal	16 bits	Fx13w	46			
		constant	32 bits	Fx13d	47			
		Register	8 bits	F-14	48			
		from	16 bits	F-14w	48			
		register	32 bits	F-14d	49			
	Binary division	Register	8 bits	Fc14	50			
			16 bits	Fc14w	50			
		constant	32 bits	Fc14d	51			
		Register	8 bits	Fx14	52			
_ogical		with hexadecimal	16 bits	Fx14w	52			
operation		constant	32 bits	Fx14d	53			
nstruction	Regi	Desister	8 bits	F-17	11-9			
Instruction		Register from	16 bits	F-17w	9			
		register	32 bits	F-17d	10			
			Register	8 bits	Fc17	11		
	AND	with octal	16 bits	Fc17w	11			
		constant	32 bits	Fc17d	12	Compare		
		Register	8 bits	Fx17	13	instructions	c	
		with hexadecimal	16 bits	Fx17w	13		(v	
		constant	32 bits	Fx17d	14		re	
		Register	8 bits	F-18	15		0	
		from	16 bits	F-18w	15			
		register	32 bits	F-18d	16			
		Register	8 bits	Fc18	17			
	OR	with octal	16 bits	Fc18w	17			
		constant	32 bits	Fc18d	18			
		Register	8 bits	Fx18	19			
		with	16 bits	Fx18w	19			
		hexadecimal constant	32 bits	Fx18d	20			
			8 bits	F-09	10-21			
	Com	olements	16 bits	F-09w	21			
	Complements	32 bits	F-09d	22	1	1		

	Ту	p	e		Mnemonics	See page
				1 byte	F-12	10-35
				1 word	F-12w	36
			egister	2 words	F-12d	37
			om gister	n bytes	F-112	13-13
			•	n words	F-112w	14
	Compare			n double word	F-112d	15
	Compare	D	agiatar	1 byte	Fc12	10-38
			egister th octal	1 word	Fc12w	38
		cc	onstant	2 words	Fc12d	39
		Re	egister	1 byte	Fx12	40
			th	1 word	Fx12w	40
			exadecimal Instant	2 words	Fx12d	41
				1 byte	F-180	14-4
			Register from	1 word	F-180w	5
			register	2 words	F-180d	6
		>	Pagistor	1 byte	Fc180	7
			Register with octal	1 word	Fc180w	8
			constant	2 words	Fc180d	9
				1 byte	F-181	4
			Register from	1 word	F-181w	5
			register	2 words	F-181d	6
		<		1 byte	Fc181	7
			Register with octal	1 word	Fc181w	8
noro			constant	2 words	Fc181d	9
npare	0			1 byte	F-182	4
ctions	Compare		Register from	1 word	F-182w	5
	(with		register	2 words	F-182d	6
	relay output)	=		1 byte	Fc182	7
	ouipui)		Register with octal	1 word	Fc182w	8
			constant	2 words	Fc182d	9
		-		1 byte	F-183	4
			Register from	1 word	F-183w	5
			register	2 words	F-183d	6
		≧		1 byte	Fc183	7
			Register	1 word	Fc183w	8
			with octal constant	2 words	Fc183d	9
				1 byte	F-184	4
			Register from	1 word	F-184w	5
			register	2 words	F-184d	6
		≦		1 byte	Fc184	7
			Register with octal	1 word	Fc184w	8
			constant	2 words	Fc184d	9
				2 words 1 byte	F-185	4
			Register from	1 word	F-185w	5
			register	2 words	F-185d	6
		< >		1 byte	Fc185	7
			Register with octal	1 word	Fc185w	8
			constant	2 words	Fc185d	9
		-	I	1 byte	F-212	23
			egister om	1 word	F-212 F-212w	23
			gister	2 words	F-212W	24
	Window		aista-		-	25
	compa-		egister th octal	1 byte	Fc212	20
	rator		nstant	1 word	Fc212w	20
			anister	2 words	Fc212d	
			egister ith	1 byte	Fx212	28
			exadecimal	1 word	Fx212w	28
		CC	onstant	2 words	Fx212d	29

	Туре		Mnemonics	See page		Туре		Mnemonics	See page				
		2 digits → 8 bits	F-03	10-7			1 byte	F-170	13-44				
			F-03w	7		Insert data	1 word	F-170w	45				
		4 digits →16 bits	F-53	12-3			2 words	F-170d	46				
	BCD to BIN		F-03d	10-8			1 byte	F-171	47				
		8 digits →32 bits	F-153	13.31		Delete data	1 word	F-171w	48				
		8 bits → 2 digits	F-04	10-9			2 words	F-171d	49				
			F-04w	9			1 byte	F-172	50				
		16 bits \rightarrow 6 digits	F-54	12-3		Search data	1 word	F-172w	51				
	BIN to BCD		F-04d	10-10	Data		2 words	F-172d	52				
	3	2 bits →10 digits	F-154	13-31			1 byte	F-173	53				
	I		-		processing	Change data	1 word	F-173w	54				
	Convert H	EX to ASCII	F-252	14-41	instructions		2 words	F-173d	55				
							1 byte	F-79	12-48				
	Convert AS	CII to HEX	F-253	42		Sequencing data	1 word	F-79w	49				
	Convert s	accord to					2 words	F-79d	50				
		ite · second	F-155	13-32		SIN function		F-22	11-22				
Convert		ur∙minute∙ o second	F-156	13-33		COS fund	tion	F-23	23				
instructions	Decode fr	Decode from 4 to 16		12-1		TAN func	tion	F-24	24				
	Encode from 16 to 4		F-51	1		ASIN fund	ction	F-25	25				
	Decode	7 SEG	F-52	2		ACOS fun	ction	F-26	26				
		2 digits	F-56	5		ATAN fun	otion	F-27	27				
	10's compleme	nt 4 digits	F-56w	5			F-27	21					
		8 digits	F-56d	6		Complement bit		F-43	44				
		8 bits	F-57	7		Compleme	ent Dit	1-43	44				
	2's complemer	nt 16 bits	F-57w	7		Differentiate	at ON	F-44	45				
		32 bits	F-57d	8		Differentiate		1 - + +					
	Total of	Total of ON bit				Total of ON bit		9		Differentiate	at OFF	F-45	46
	Polar coordin	ates system	F-28	11-28	Bit processing	Set co	il	F-32	33				
	Rectangular cod	ordinate system	F-29	11-29	instructions	Reset c	oil	F-33	34				
		1 byte	F-02	10-5			Indirect	F-130	13-17				
	L	1 word	F-02w	5		Multiplex bit	address	00					
	Exchange data	2 words	F-020	6			Direct	F-131	17				
Eveberg-		n bytes	F-174	14-1			address						
Exchange instructions		1 byte	F-174	12-4		Cot/report bit	Indirect address	F-132	18				
	Swap upper 4 b with lower 4 bit	its	F-175	14-1		Set/reset bit	Direct address	F-133	18				

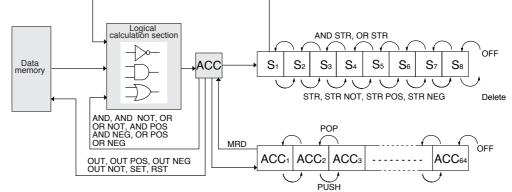
	Type 2 digits			Mnemonics	puge		Туре		Mnemonics	See page		
	BCD up/down 2 digits 4 digits		F-62	12-17		Set mast	er control	F-30	11-30			
	counter			F-62w	18							
			8 digits 2 digits	F-62d F-65	19 24		Reset ma	ster control	F-31	30		
		untor	4 digits	F-65w	24							
	Add BCD co	unter	8 digits	F-65d	25		Master contro	l reset nesting	F-231	14-39		
		~ ~ ~	2 digits	F-66	26		Catium	n control				
	Subtract B counter		4 digits	F-66w	26		Set jum	p control	F-41	11-42		
	counter		8 digits	F-66d	27	Operational	Reset iur	np control	F-42	4		
			1 byte	F-63	20	condition	1100001 jui		1-42			
		+ 1	1 word 2 words	F-63w	20 21	instructions	Jump control	reset nesting	F-242	14-40		
	Add binary		1 byte	F-63d F-163	13-40			0				
	counter	+ 2	1 word	F-163w	40		Set level oper	ating condition	F-47	11-47		
	counter	+ 2	2 words	F-163d	41							
			1 byte	F-263	14-47		Reset level ope	rating conditior	i F-48	47		
Timer/		+ 4	1 word	F-263w	47			Unconditional	F-40			
counter			2 words	F-263d	48		End	end	F-40	4		
instruc-			1 byte	F-64	12-22			Conditional	F-49	48		
tions		— 1	1 word 2 words	F-64w	22 23			end	1 10			
	Subtract		2 words 1 byte	F-64d F-164	13-42		Lab	el	F-140	13-19		
	binary	- 2	1 word	F-164w	42							
	counter	- 2	2 words	F-164d	43			Direct address	F-141	20		
			1 byte	F-264	14-49		Jump					
		- 4	1 word	F-264w	49			Indirect address	F-151	30		
			2 words	F-264d	50			Direct address	F-142	22		
	(set				tract timer				Call subroutine	Direct address	1-142	24
				F-260	43	Branch		Indirect address	F-148	28		
	E	value,		1-200	-10	instructions						
	Expansion address)		,				Return from	Unconditional return	F-143	22		
			tract timer				subroutine	Conditional				
		(sett	e, register	Fc260	44		Gubroutine	return	F-149	29		
			ress)					Direct address	F-144			
		Sub	tract counter				Set loop count		1-144	24		
		(sett		F-261	45			Indirect addres	F-146	26		
		valu	e, register	F-201	45							
	Expansion	addi	ress)				End o	f loop	F-145	24		
	counter	Sub	tract counter									
		(con	istant,	Fc261	46		Forced er	nd of loop	F-147	27		
			ster address)									
		1	8 bits	F-60	12-10							
			16 bits	F-60w	12-10							
	Reversible	chift	32 bits	F-60d	13							
	register		n bits (register address)	F-160	13-34							
Shift			n bits (constant address)	Fc160	35							
instruc-			1 byte	F-61	12-14							
tions	Aounchrom		1 word	F-61w	15							
	Asynchron		2 words	F-61d	16							
	reversible	n bytes	F-161	13-36								
				38								
	shift regis	ter	n words	F-161w								
			n words n double word pper shift)	F-161W F-161d F-67	<u>39</u> 12-28							

	Туре		Mnemonics	See page		
	Comparison with curr (specified re	elay set)	F-34	11-36		
	Comparison with curr (specified rel	F-35	37			
Clock instructions	Addition	of time	F-36	38		
	Subtractio	n of time	F-37	39		
	Transfer of currer	nt value of clock	F-38	40		
	Open channe	I instruction	F-202 F-203	14-10		
	Open channel 1 (Set hier	archy communication)	F-206	13		
Communication instructions	Open channel 2 (Set hier	archy communication)	F-207	13		
	Sending in	struction	F-204	14-11		
	Receiving i	F-205	12			
	Maintenance d	F-20	11-21			
	Square	Square root				
	Formation of sum c	heck code of data	F-77	12-46		
	Check	data	F-78	47		
	Refres	h I/O	F-80	13-1		
Other instructions		Refresh	F-82	2		
Instructions	Special I/O module	Read out	F-85	3		
		Write	F-86	3		
			F-100	7		
	Set the indire	ect address	F-101	8		
	Remark (instruction for c	F-90 (REM)	4			
	Logging f	unction	F-403	14-56		

Chapter 8. Description of basic instructions

8-1 Calculation of basic instructions

For calculation of the basic instructions, the JW300 uses the data memory section, logical calculation section, accumulator (ACC), and stack registers (S_1 to S_8).



(1) Data memory

- This is a memory specified by relay numbers (input relay, output relay, auxiliary relay, timer/counter contact point). It stores ON/OFF data.
- With the basic instructions, the JW300 reads out ON/OFF data of the data memory with the following instructions.

```
STR, STR NOT, AND, AND NOT, OR, OR NOT
```

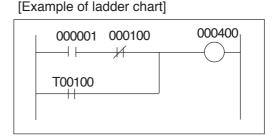
STR POS, STR NEG, AND POS, AND NEG, OR POS, OR NEG

The JW300 writes data to the data memory with the following instructions.

OUT, OUT POS, OUT NEG, OUT NOT, SET, RST, timer/counter

[Example of writing instructions]

STR	000001
AND NOT	000100
OR	T00100
OUT	000400
\land	\wedge
Instructions	Relay number



(2) Logical calculation section

According to instruction details, the JW300 carries out logical operations.

(3) Accumulator(ACC)

- This is a one-bit register that stores logical operation results.
- The register changes with the following instructions.

STR, STR NOT, AND, AND NOT, OR, OR NOT, AND STR, OR STR

STR POS, STR NEG, AND POS, AND NEG, OR POS, OR NEG, PUSH, POP

(4) Stack register (S1, S2, S3, S4, S5, S6, S7, S8)

- This is an 8-bit register in which the intermediate result is stored during processing of the serial/ parallel circuit or counter instruction, and application instruction (F-60, F-60w, F-62, F-62w etc.) that has a plural number of input conditions.
- The register changes with the following instructions.

STR, STR NOT, AND STR, OR STR, STR POS, STR NEG

(5) Clear internal memory area (ACC1, ACC2, ACC3, ----, ACC64)

- These are areas to memorize contents of the accumulator and stack.
- The areas change with the PUSH and POP instructions.

8-2 Description of each basic instruction

STR (Store)

Curren	hal	R	[Use example 1]
Sym		Use when a 1st contact point from a bus line, or a 1st contact point of a circuit block is "a" contact (normally OFF).	000000 000400 STR 000000 OUT 000400
Operatio	n details	JW300 stores the data memory content (ON/OFF data) having relay number R to the accumulator (ACC). The previously located ON/OFF data in the ACC shifts to stack register S1. The previous S1 data shift to S2. As such, other stack register data shift as S2 -> S3, S3 => S4, S4 -> S5, S5 -> S6, S6 -> S7, and S7 -> S8. The data in the S8 stack register is deleted. R ACC S1S2S3S4S5S6S7S8 Overflow	When input relay 000000 is ON, output relay 000400 goes ON. 000000 000400
	R detail		000000 000002 000401
	ACC S1	Content of R data memory	000001 000003 000004
	51 S2	Content of the ACC before the operation Content of S1 before the operation	Block A
After the operation		Content of S2 before the operation	Bus line Block B
	S4	Content of S ₃ before the operation	STR 0000001st contact point of the bus line (block A) OR 000001
	S5	Content of S4 before the operation	STR 0000021st contact point of block B
	S ₆	Content of S5 before the operation	STR 0000031st contact point of block C AND 000004
	S 7	Content of S6 before the operation	OR STR AND STR
	S8	Content of S7 before the operation	OUT 000401

	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
	000000 to 015777	000000 to 015777	000000 to	015777	
Use range of R	020000 to 075777	020000 to 075777	020000 to	075777	
	T00000 to T01777	100000 to 153777	100000 to	543777	
	C00000 to C01777	T00000 to T03777	T00000 to	o T17777	
		C00000 to C03777	C00000 to	o C17777	

STR N	ΝΟΤ	(Store not)	
Symbol Function Operation details		R ∦ 	[Use example 1] Instruction 000000 000402 STR NOT 000000
		Use when a 1st contact point from a bus line, or a 1st contact point of a circuit block is "a" contact (normally OFF).	When input relay 000000 is ON, output relay 000402
		Inverts the contents (ON/OFF data) of the data memory having relay number R, and stores them in the accumulator (ACC). The previously located ON/OFF data in the ACC shifts to stack register S1. The previous S1 data shift to S2. As such, other stack register data shift as S2 -> S3, S3 => S4, S4 -> S5, S5 -> S6, S6 -> S7, and S7 -> S8. The data in the S8 stack register is deleted. R \rightarrow ACC \rightarrow S1S2S3S4S5S6S7S8 Overflow	goes ON. 000000 000402 [Use example 2] 000000 000002 000403
	R detail	Latch	000001 000003 000004
	ACC	The value after inverting the content of R	Block ABlock C
	S 1	Content of the ACC before the operation	Bus line Block B
After the	S 2	Content of S1 before the operation	STR NOT 000000 1st contact point of the bus line (block A)
operation	S3	Content of S2 before the operation	OR 000001 STR NOT 000002 1st contact point of block B
	S 4	Content of S3 before the operation	STR NOT 000003 1st contact point of block C AND 000004
	S5	Content of S4 before the operation	OR STR
	S6	Content of S5 before the operation	AND STR OUT 000403
	S 7	Content of S6 before the operation	
	S8	Content of S7 before the operation	

	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-341CU JW-352CU JW-362CU
	000000 to 015777	000000 to 015777	000000 to 015777
Use range of R	020000 to 075777	020000 to 075777	020000 to 075777
Use range of h	T00000 to T01777	100000 to 153777	100000 to 543777
	C00000 to C01777	T00000 to T03777	T00000 to T17777
		C00000 to C03777	C00000 to C17777

AND	A) (A	ND)			
Symbol		R 	[Use example]	Instruction STR 000000	
Function		Use when the serial contact point is "a" contact (normally OFF).	- 000000 000001 000404 STR 00 AND 00 OUT 00		
Operation details		result to the ACC.	If both the input relays 000000 and 000001 are ON, output relay 000404 goes ON.		
		R AND ACC S1S2 S7S8 (Latch)	000000		
	R detail	Latch	000001		
After the operation	ACC	Value after AND operated the R content and ACC content before the operation.	000404		
	S1 to S8	Latch			

	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Use range of R	000000 to 015777	000000 to 015777		000000 to		
	020000 to 075777 T00000 to T01777	020000 to 075777 100000 to 153777		020000 to 100000 to		
	C00000 to C01777	T00000 to T03777		T00000 to		
		C00000 to C03777		C00000 to	o C17777	

AND NOT (AND NOT)

Sym	bol	R	[Use example] Instruction
Function		Use when the serial contact point is "b" contact (normally OFF).	000000000001 000405 000000 AND NOT 000001 000001 000405 000001 000405
Operation details		After inverting the content (ON/OFF data) of the data memory having relay number R, executes AND operation between the inverted data and the content of the accumulator (ACC), and store the result to the ACC. $R \rightarrow AND - ACC S_{1}S_{2} / S_{7}S_{8} / (Latch)$	When input relay 000000 is ON and 000001 is OFF, output relay 000405 goes ON.
	R detail	Latch	000001
After the ACC operation		The value after the AND operation of the inversed content of the R and ACC content before the operation	000405
	S1 to S8	Latch	
Use range of R Same as the above (AND).			

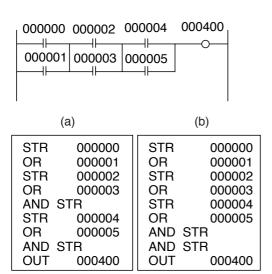
OR	(0	PR)	
Symbol Function Operation details			[Use example] Instruction
		Use when a parallel contact point is "a" contact (normal OFF).	000001 000001
		Executes OR operation between the content of the R data memory (ON/OFF data) and the content of the accumulator (ACC), and stores the result to the ACC.	If both the input relays 000000 and 000001 are ON, output relay 000406 goes ON. 000000 0000001
R detail		Latch	
After the operation	ACC	Value after OR operated the R content and ACC content before the operation.	000406
	S1 to S8	Latch	

	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Use range of R	000000 to 015777 020000 to 075777	000000 to 015777 020000 to 075777		000000 to 020000 to		
	T00000 to T01777	100000 to 153777			o 543777	
	C00000 to C01777	T00000 to T03777		T00000 to		
		C00000 to C03777		C00000 to	001////	

OR N	ОТ (О	PR NOT)	
Sym	bol	R ∦	[Use example] Instruction 000407 STR 000000
Function		Use when a parallel contact point is "b" contact (normal OFF).	000000 000407 000001 OR NOT 000001 OUT 000407
Operation details		After inverting the content (ON/OFF data) of the data memory having relay number R, executes OR operation between the inverted data and the content of the accumulator (ACC), and store the result to the ACC. $R \xrightarrow{OR} ACC S_1S_2 \xrightarrow{S_7S_8}$ (Latch)	When input relay 000000 is ON or 000001 is OFF, output relay 000407 goes ON.
After the operation S1 to S8		Latch	000001
		The value after the OR operation of the inversed content of the R and ACC content before the operation	000407
		Latch	
Use ran	ge of R	Same as the above (OR).	

AND	STR	(AND STORE)			
Func	tion	Use to connect circuit blocks to each other in series. [Use example]			
Operation detailsExecutes AND operation between stack register S1 and content (ON/OFF data) of the accumulator (ACC), and stores the result to the ACC. The previously located ON/OFF data in the ACC shifts to stack register S1. The previous S1 data shifts to S2. Then, other stack register data shift as S2 -> S3, S3 -> S4, S4 -> S5, S5 -> S6, S6 -> S7, and S7 -> S8. The data in the S8 stack register is entered as OFF data.		register S ₁ and content (ON/OFF data) of the accumulator (ACC), and stores the result to the ACC. The previously located ON/OFF data in the ACC shifts to stack register S ₁ . The previous S ₁ data shifts to S ₂ . Then, other stack register data shift as S ₂ -> S ₃ , S ₃ -> S ₄ , S ₄ -> S ₅ , S ₅ -> S ₆ , S ₆ -> S ₇ , and S ₇ -> S ₈ . The data in the S ₈ stack register is entered as OFF data.	Image: Connect block A and block B in a series. When input relay 000003 is ON, output relay 000410 goes ON.		
	ACC	AND operation result between the S1 content before calculation and ACC content	000000		
	S 1	Content of S2 before the operation	000001		
After the	S ₂	Content of S3 before the operation			
operation	S3	Content of S4 before the operation	000002		
	S4	Content of S5 before the operation	000003		
	S₅	Content of S6 before the operation			
	S6	Content of S7 before the operation	000410		
	S 7	Content of S ₈ before the operation			
	S8	OFF (0)			

Note 1: For programming the following ladder chart, two methods ("a" and "b") are available.



The same calculation result can be obtained from both (a) and (b). However, (a) uses the S_1 stack register only and (b) uses stack registers S_1 and S_2 . As the JW300 has 8 stack registers, programming (b) will restrict connection of blocks up to 9 blocks.

Note 2: The following examples (a) and (b) bring the identical operation.

(a)

	STR STR OR AND OUT	000000 000001 000002 STR 000400
--	--------------------------------	---

(b)

000001 000000 000400

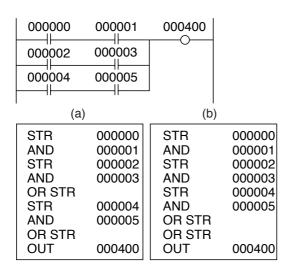
	STR	000001
000002	OR	000002
	AND	000000
	OUT	000400

The number of steps of (b) is small by one than (a).

OR S	TR (C	OR STORE)	
Function		Use to connect circuit blocks to each other in parallel.	[Use example] Block A
Operation details		Executes OR operation between stack register S ₁ and content (ON/OFF data) of the accumulator (ACC), and stores the result to the ACC. The previously located ON/OFF data in the ACC shifts to stack register S ₁ . The previous S ₁ data shifts to S ₂ . Then, other stack register data shift as S ₂ -> S ₃ , S ₃ -> S ₄ , S ₄ -> S ₅ , S ₅ -> S ₆ , S ₆ -> S ₇ , and S ₇ -> S ₈ . The data in the S ₈ stack register is entered as OFF data.	Instruction00000000000100411000002000003STR000002000003STR000002000003STR000002000003OR STRBlock BOUT00001Str00001Str000002Out000411Str000003Str000003Out000411Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str00001Str000001Str000001Str000002Str000001Str000002Str000001Str000002Str000001Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000002Str000
	ACC	OR operation result between the S1 content before calculation and ACC content	000000
	S1	Content of S2 before the operation	000001
After the	S 2	Content of S ₃ before the operation	
operation	S ₃	Content of S4 before the operation	
	S 4	Content of S₅ before the operation	000003
	S ₅	Content of S6 before the operation	
	S6	Content of S7 before the operation	000411 L L
	S 7	Content of S ₈ before the operation	
	S8	OFF (0)	

Note1: For programming the following ladder chart, tw	vo
methods ("a" and "b") are available.	

_



Note 2: The following examples (a) and (b) bring the identical operation.

(a) 000400 000000 STR 000000 ╢ \bigcirc STR 000001 000001 000002 000002 AND ╢ ┦┠ OR STR OUT 000400 (b)

STR	000001
AND	000002
OR	000000
OUT	000400
	AND OR

The number of steps of (b) is small by one than (a).

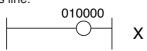
The same calculation result can be obtained from both (a) and (b). However, (a) uses the S1 stack register only and (b) uses stack registers S1 and S2. As the JW300 has 8 stack registers, programming (b) will restrict connection of blocks up to 9 blocks.

OUT (OUT)

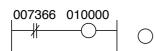
Symbol		R 	[Use example]		
Function		Use to output calculation result.	STR 000000 000413 OUT 000412		
Operation details		Write the content of the accumulator (ACC) to the data memory having relay number R.			
	R detail	ACC detail	When input relay 000000 is ON, output relay 000412, 000413, and 000414 go ON.		
After the	ACC	Latch	(With the OUT instruction, the ACC content does not change, so that continuous use of the OUT instruction		
operation	S1 to S8	Latch	is possible.)		
		JW-311CU JW-321CU JW-312CU JW-322CU	JW-331CU JW-341CU JW-352CU JW-362CU		

		JW-312CU	JW-322CU	JW-332CU	JW-341CU	JW-352CU	JW-362CU
Use rang	e range of R	000000 to 015777 020000 to 075777	000000 to 015777 020000 to 075777		000000 to 020000 to		
		02000010073777	100000 to 153777		100000 to		

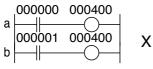
- Note 1: The special relay areas (007300 to 007377) cannot be used for output.
- Note 2: The OUT instruction cannot be started from a bus line.



For a relay you want to normally be ON, use a normally OFF contact point (007366).



Note 3: If the same relay number is used double with the OUT instruction, the error will be shown with program check using support tool JW-15PG.



Note 4: After calculation of the OUT instruction, the content of the ACC does not change, so that the following program is also available.

	000412 000413 002 000414	STR OUT AND OUT AND	000000 000412 000001 000413 000002
0000	02 000414		
		OUT	000414

TMR (Timer instruction)

The TMR instructions include five types: decrement, increment (these decrement or increment installed 0.1 second clock), handle calculated values with BCD or binary.

It also can be used by specifying a register after changing the set value to a fixed value. Using this instruction by combining it with a write instruction to registers, you can easily change set values.

(1) TMR instruction types

Instruction	Operation	Notation	TMR number *1	Setting area *2
TMR			00000 to 17777	Specify 0000 to 7999 registers
DTMR (BCD)	Decrement	BCD	00000 to 17777	Specify 0000 to 7999 registers
DTMR (BIN)		Binary	00000 to 17777	Specify 0000 to 32767 registers
UTMR (BCD)	Increment	BCD	00000 to 17777	Specify 0000 to 7999 registers
UTMR (BIN)	morement	Binary	00000 to 17777	Specify 0000 to 32767 registers

*1: The TMR number varies with the control module model used.

		JW-331CU JW-332CU		JW-352CU	JW-362CU
00000 to 01777	00000 to 03777		00000 t	o 17777	

The TMR number is shared between CNT and MD.

*2: For specifying registers, all of the byte addresses in the data memory can be specified. For specifying byte addresses, make sure to specify even address numbers using 2 bytes.

(2) Decrement TMR instructions

- Counting does not take place when the start input is OFF and the setting value keeps to be the current value, and the TMR contact is OFF.
- When the start input turns ON, the current value is decremented one step at every 0.1 second and come value=0, then the TMR contact is closed, and the condition is maintained as long as the start input is ON.

Start input	Current value	TMR contact
OFF	Setting value	OFF
ON (current value > 0)	Decremented one step at every 0.1 second	OFF
ON (current value = 0)	0	ON

(3) Increment TMR instructions

- While the start input is inactive (OFF), no count operation occurs, with the counter held at the current value=0 and the TMR contacts left open.
- When the start input is activated, the counter is incremented by one at 0.1 sec. intervals. When the setting value is reached, the TMR contacts are closed. This state is maintained as long as the start input remains active.

Start input	Current value	TMR contact
OFF	0	OFF
ON (current value < setting value)	Incremented one step at every 0.1 second	OFF
ON (current value = setting value)	Setting value	ON

(4) Symbol mark

Instruction	Symbol of TMR	Symbol of TMR contact
TMR	1)	
DTMR (BCD)	1)	
DTMR (BIN)	1)) T00001
UTMR (BCD)	1)	
UTMR (BIN)	1)	

1) Start input

Start with ON state signal

2) TMR number

00000 to 17777₍₈₎ { 00000 to 00777: Common among TMR, CNT, and MD 01000 to 17777: Common between TMR and CNT

3) Setting value

0.1 second (100ms) unit 0.01 second (10ms) unit* 0.001 second (1ms) unit* *Setting system memory #0227 can set TMR00000 to 00777 to 10 ms timer. Setting system memory #0225 can set TMR01770 to 01777 to 1 ms timer. However, the DTMR and UTMR function as 100ms only.

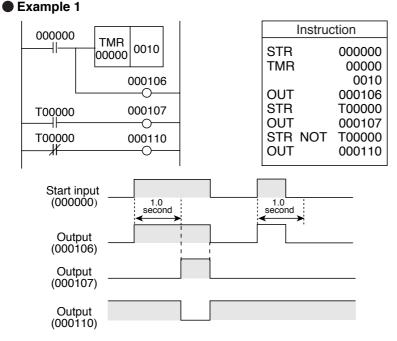
4) Precision

100 ms (setting value $^{+0}_{-0.1s}$) + scan time (second)

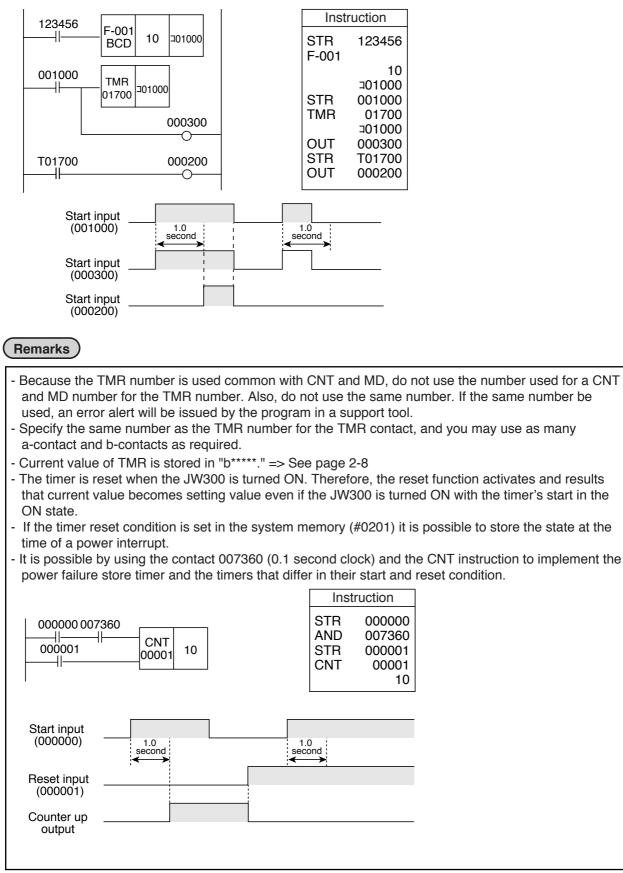
10 ms (setting value $\frac{+0}{-0.01s}$) + scan time (second)

1 ms (setting value $\frac{+0}{-0.001s}$) + scan time (second)

(5) Use example



• Example 2



CNT (Counter instruction)

The CNT instructions include five types: decrement, increment (these decrement or increment the counter at the rising edge of the counter input), handle calculated values with BCD or binary. It also can be used by specifying a register after changing the set value to a fixed value. Using this instruction by combining it with a write instruction to registers, you can easily change set values.

(1) CNT instruction types

Instruction	Operation	Notation	CNT number *1	Setting area *2
CNT			00000 to 17777	Specify 0000 to 7999 registers
DCNT (BCD)	Decrement	BCD	00000 to 17777	Specify 0000 to 7999 registers
DCNT (BIN)		Binary	00000 to 17777	Specify 0000 to 32767 registers
UCNT (BCD)	Increment	BCD	00000 to 17777	Specify 0000 to 7999 registers
UCNT (BIN)		Binary	00000 to 17777	Specify 0000 to 32767 registers

*1: The CNT number varies with the control module model used.

JW-311CU JW-312CU				JW-352CU	JW-362CU
00000 to 01777	00000 to 03777	00000 to 17777			

The CNT number is shared between CNT and MD.

*2: For specifying registers, all of the byte addresses in the data memory can be specified. For specifying byte addresses, make sure to specify even address numbers using 2 bytes.

(2) Decrement CNT instructions

- No counting takes place even if the counter input changes its state from OFF to ON as long as the reset input is ON and the setting value is retained for the current value and the CNT contact is OFF.
- When the counter changes from OFF to ON while the reset input is OFF, the current value is decremented by one. When the current value becomes 0, the CNT contact returns ON and its state is maintained until the reset input is turned OFF.

Reset input	Current value	CNT contact
ON	Setting value	OFF
OFF (current value > 0)	Decremented one step each time the counter input changes from OFF to ON	OFF
OFF (current value = 0)	0	ON

(3) Increment CNT instructions

- No counting takes place even if the counter input changes its state from OFF to ON as long as the reset input is ON and the 0 is retained for the current value and the CNT contact is OFF.
- While the reset input is inactive, the counter is incremented by one each time the count input is set to ON. When the preset value is reached, the CNT contacts are closed. This state is maintained as long as the reset input remains inactive.

Reset input	Current value	CNT contact
ON	0	OFF
OFF (current value < 0)	Incremented one step each time the counter input changes from OFF to ON	OFF
OFF (current value = 0)	Setting value	ON

(4) Symbol mark

Instruction	Symbol of CNT	Symbol of CNT contact
CNT	$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	
DCNT (BCD)	$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	
DCNT (BIN)	1) DCNT 00001 00100 2) (BIN) 00001 00100 3) 4)	2) C00001
UCNT (BCD)	1) UCNT 00001 0100 2) (BCD) 00001 0100 3) 4)	
UCNT (BIN)	1) UCNT 00001 00100 2) (BIN) 00001 00100 3) 4)	

1) Counter input

Senses OFF to ON transition

2) Reset input

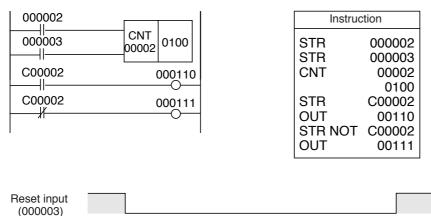
Reset with ON

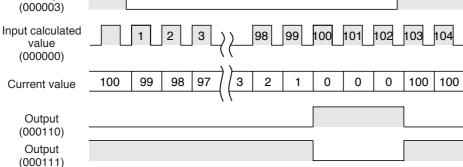
3) CNT number

00000 to 17777₍₈₎{ 00000 to 00777: Common among TMR, CNT, and MD 01000 to 17777: Common between TMR and CNT

4) Setting value

(5) Use example • Example 1



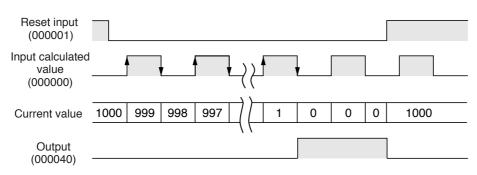


Example 2

Counters that count at rising and falling edges of count input

000000			Instruction	
000001	CNT 00100 1000		STR STR	000000 000001
000000			CNT	00100
000001	CNT 1000		STR NOT	1000 000000
C00100			STR	000001
	00040		CNT	00100 1000
I		I	STR	C00100
			OUT	000040

- These are counters that subtract with both cases when a count input changes from OFF to ON and from ON to OFF.





000001	F-001 100 =01000	Instruction
000002	BCD	STR 000001 F-001
000003	CNT 00002 701000	100 ⊐01000
C00002	000200	STR 000002 STR 000003
"	- 1	CNT 00002 ⊐01000
		STR C00002 OUT 000200

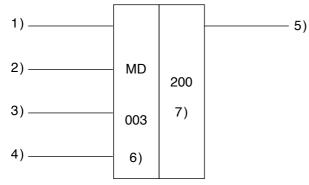
- These are counters that subtract with both cases when a count input changes from OFF to ON and from ON to OFF.

Reset input (000003)							
Input calculated value (000002)		1	2	98	99	100	101
Current value	100	99	98	2	1	0	100
Output (000200)							
Remarks							

- Because the CNT number is used common with TMR and MD, do not use the number used for a TMR and MD number for the CNT number. Also, do not use the same number. If the same number be used, an error alert will be issued by the program in a support tool.
 Also, do not use the same CNT number. If the same number be used, an error alert will be issued
- by the program in the programmer. You may ignore this warning if it is used intentionally.Specify the same number as the CNT number for the CNT contact, and you may use as many contacts and b-contacts as required.
- As successive inputs are disregarded when the counter has reached zero, the reset input must be set ON and OFF or forced to reset using a support tool, in order to start counting again.
- If both the counter input and the reset input go ON simultaneously, the reset input takes preference over the counter input.
- Current value of TMR is stored in "b*****." => See page 2-8
- When a power interrupt is encountered, the counter retains the current value. But, the current value will be reset if the reset input was programmed to turn ON at power ON. If it is necessary to retains the current value after the power interrupt, use the reset input that turns OFF at power ON.
- It would be possible to reset with OFF, if the reset condition is given in the system memory (#0202).

MD (Maintenance display)

The MD (Maintenance Display) instruction is used to monitor the operating condition of the controlled device and to trace a trouble cause in the support tool such as the programmer or send the output signal to an external source.



1) 2) 3)	Input information	This is an external output contact information used in conjunction with the MD data of 7). Relay, TMR, and CNT contact may be used.
4)	Output direct condition	This is an input used to direct whether the contact information of 1), 2), 3), and MD data of 7) be output to the data memory or relay area of the MD number specified by 6). Relay, TMR, and CNT contact may be used. Output becomes active when ON. No change occurs in the contact information and MD information even if it was turned OFF.
5)	MD expansion output	There is no need of programming the MD instruction with the condition of 4) above, when using the MD instruction continuously with the same output direct condition.
6)	MD number	 The MD instruction uses the TMR and CNT current value storage area or relay area for the storage of contact information of 1), 2), and 3) and MD data of 7). (1) To use the TMR or CNT area Similar as TMR and CNT, it should be programmed using the number of 000 to 777 and the information must be monitored on the programmer, etc. Note: It is not possible to use the same number used for the TMR and CNT. (2) When the relay area is used It should be programmed in terms of byte address ∃xxxxx. As an example, program ∃00000, 2 bytes of ∃00000 and ∃00001 will be MD area. Use of the area connected with the output module will permit external output (display) of the contact information of 1), 2), 3) and the MD data of 7).
7)	MD data	Any number of 000 to 999 may be used in the BCD code. It should be programmed in connection with the process number, relay number, external device number, etc.

STR PO	S "a" cont	act load risi	ng					
Symbol	P							
Function	 OFF to ON, this of This instruction level calculation This instruction 	When a relay number R that was specified by STR POS instruction changes from OFF to ON, this contact turns ON for one scan time. This instruction operates only for one scan even it is entered between F-47 (set level calculation condition) and F-48 (reset level calculation condition). This instruction does not operate for one scan interval soon after turning ON the power and soon after start operation.						
	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU		
Use range of R	000000 to 015777 020000 to 075777 T00000 to T01777 C00000 to C01777	000000 to 015777 020000 to 075777 100000 to 153777 T00000 to T03777 C00000 to C03777		000000 to 020000 to 100000 to T00000 to C00000 to	075777 543777 517777			
[Use exam	ple]							
	00 000001 000002	004000			000002			
000000								
000000 ├── P ── Output from the above 000001	n 1 scan time	1 s	can time		1 scan time			
000002								
004000 (output)		↓ ↓ 1 s	can time		t scan time			

STR NEG "a" contact load falling

Symbol	R N					
Function	OFF to ON, this - This instruction level calculation - This instruction	mber R that was s contact turns ON f operates only for n condition) and F- does not operate n after start operat	or one scan ti one scan eve 48 (reset leve for one scan i	me. n it is enterec el calculation	l between F-4 condition).	47 (set
	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Use range of R	000000 to 015777 020000 to 075777 T00000 to T01777 C00000 to C01777	000000 to 015777 020000 to 075777 100000 to 153777 T00000 to T03777 C00000 to C03777		000000 to 020000 to 100000 to T00000 to C00000 to	075777 543777 517777	
[Use exam 00000	00 000001 000002	004000			00000	1
000000						
000000 ├──N└── Output fror the above		1 scan time	e 1 scan	time	1 scan time	
000001						
000002						
(output)		← → 1 scan time		1	scan time	

AND POS "a" contact AND rising

Symbol	R P							
Function	 When relay number R that was specified by AND POS instruction changes from OFF to ON, this contact generates pulses of one scan time length, and executes AND operation with the ACC content, and stores its result to the ACC. This instruction operates only for one scan even it is entered between F-47 (set level calculation condition) and F-48 (reset level calculation condition). This instruction does not operate for one scan interval soon after turning ON the power and soon after start operation. 							
	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU		
Use range of R	000000 to 015777 020000 to 075777 T00000 to T01777 C00000 to C01777	JW-312CO JW-322CO JW-332CO JW-342CO 000000 to 015777 000000 to 015777 000000 to 015777 20000 to 075777 020000 to 075777 020000 to 075777 000000 to T01777 100000 to 153777 100000 to 543777						
[Use exam	ple]							
	00 000001 000002	004000		STR	000001 000001 000002 000002	2		
000000								
000001								
000002 O00002 Output from the above 004000 (output)	1 : 	scan time	1 scan t		1 scan time 1 scan time			

AND NEG

"a" contact AND falling

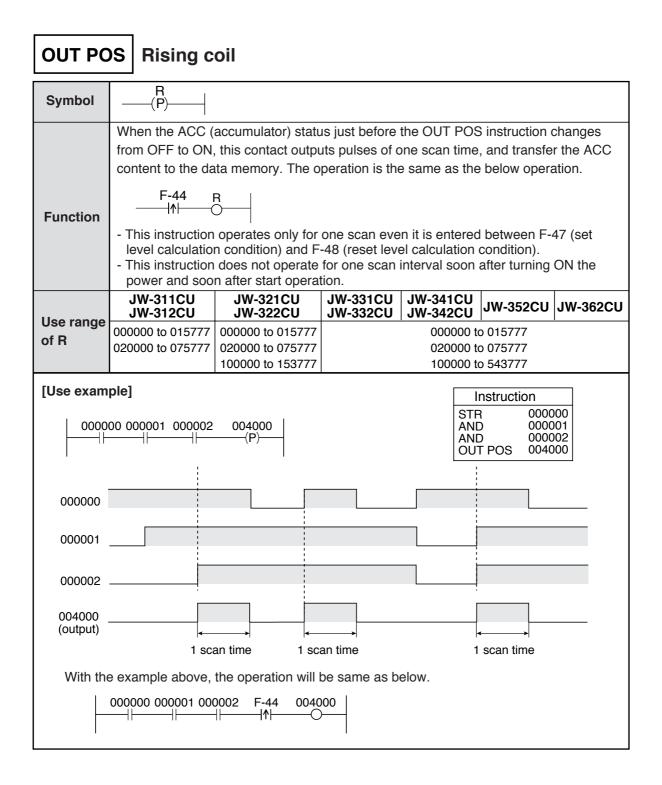
Symbol						
Function	to ON, this contact operation with the - This instruction level calculation - This instruction	er R that was spect of generates pulses ACC content, and operates only for c condition) and F-4 does not operate for after start operation	s of one scan d stores its resone scan ever 48 (reset level or one scan ir	time length, a sult to the AC i it is entered l calculation o	and executes C. between F-4 condition).	s AND 7 (set
	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Use range of R	000000 to 015777 020000 to 075777 T00000 to T01777 C00000 to C01777	000000 to 015777 020000 to 075777 100000 to 153777 T00000 to T03777 C00000 to C03777		020000 t 100000 t T00000 t	o 015777 o 075777 o 543777 o T17777 o C17777	
[Use exam	piej 000 000001 000002 ────│ ──── N ──	004000		STF	00000 ⁻ 0 NEG 000002	2
000000						
000001						
000002						
──N Output fror the above 004000 (output)	; 	1 scan time	, 1 scan	time	1 scan time	

OR PO	S "a" cont	act OR risin	g			
Symbol	R −−− <u> </u> P −−−					
Function	 When relay number R that was specified by OR POS instruction changes from OFF to ON, this contact generates pulses of one scan time length, and executes AND operation with the ACC content, and stores its result to the ACC. This instruction operates only for one scan even it is entered between F-47 (set level calculation condition) and F-48 (reset level calculation condition). This instruction does not operate for one scan interval soon after turning ON the power and soon after start operation. 					
	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Use range of R	000000 to 015777 020000 to 075777 T00000 to T01777 C00000 to C01777	000000 to 015777 020000 to 075777 100000 to 153777	00-33200	000000 tr 020000 tr 100000 tr T00000 tr	o 075777 o 543777	I
[Use exam	ple]					
0000	01	00		STF	POS 000001	
000000						
000001						
000001 Utput from the above 004000 (output)		1 scan time	「		1 scan tim	

OR NEG

a" contact OR falling

Symbol	R N ├					
Function	 When relay number R that was specified by OR NEG instruction changes from OFF to ON, this contact generates pulses of one scan time length, and executes AND operation with the ACC content, and stores its result to the ACC. This instruction operates only for one scan even it is entered between F-47 (set level calculation condition) and F-48 (reset level calculation condition). This instruction does not operate for one scan interval soon after turning ON the power and soon after start operation. 					
	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Use range	000000 to 015777	000000 to 015777		000000 to	o 015777	
of R	020000 to 075777			020000 to		
	T00000 to T01777	100000 to 153777		100000 to		
	C00000 to C01777	T00000 to T03777 C00000 to C03777		T00000 to C00000 to		
[Use exam						
	00 0040	00		STF	NEG 000001	
000000						
000001						
000001 Output from the above 004000 (output)		1 scan time			1 scan tim	→



OUT NEG Falling coil

Symbol	(N)							
		accumulator) statu	•			-		
		this contact output						
		ta memory. The o	peration is the	e same as me	e pelow opera	alion.		
	F-45 R → ↓↓							
Function	- This instruction	operates only for	ono scan ovo	n it is ontoroc	l between E-	17 (sot		
		- This instruction operates only for one scan even it is entered between F-47 (set level calculation condition) and F-48 (reset level calculation condition).						
		does not operate n after start operat		interval soon	after turning	ON the		
Use range	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU		
of R	000000 to 015777	000000 to 015777			o 015777			
	020000 to 075777	020000 to 075777 100000 to 153777			o 075777 o 543777			
[Use exam	nlel					7		
	0 000001 000002	004000		STR	nstruction 000000			
		(N)		AND	000001			
		I			NEG 004000			
000000								
000000								
000001 _								
	Г							
000002 -								
004000								
(output)			*					
		1 scan time	1 scan	time 1 s	scan time			
- With th	e example above,	the operation will	be same as b	elow.				
(000000 000001 000		00					

OUT NOT Inverse output input conditions

Symbol	R (/)					
Function		content and outpu the same as the b			ory.	
	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
Use range of R	000000 to 015777 020000 to 075777			000000 to 020000 to 100000 to	075777	
[Use exam	ple]				nstruction	
	0 000001 000002	004000		STR AND AND	00000	1
000000						
000001 _						
000002 -				ļ		
004000 (output)						
- With th	e example above,	the operation will	be same as b	elow.		
	000000 000001 00	0002 F-43 004	4000			

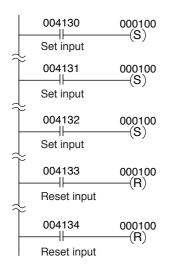
SET Set when the input signal is rising

Symbol	(S)					
Function	When the set inp	out goes ON, this in	nstruction turr	ns ON specifie	ed OUT and I	atches.
Use range	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
of R	000000 to 015777 020000 to 075777			000000 ti 020000 ti 100000 ti	075777	
[Use examp	ole]					
	00000 004000 ⊣⊢——(S)			Ir STF SET		
000000						
004000 (output)						
The OU ⁻	T004000 keeps its	anges from OFF to ON status even if OFF, OUT004000	set input 000	000 goes OF		
With the	example above, t	he operation will b	e same as be	elow.		
	000000 F-32	004000				

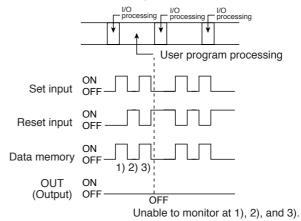
- If this instruction is located in F-30 (MCS) instruction, the OUT that was turned ON by this instruction keeps ON status even if the F-30 instruction goes OFF.
- Using this instruction, you can control one OUT on multiple circuits.
- If the OUT that is specified by this instruction locates inside a latch specified area, this OUT will be latched after recovery from a power failure. If the OUT is out of the latch specified area, it will be reset after recovery from a power failure.
- If the OUT that was specified by this instruction is in the area to latch output when the JW300 stops operation, the status before stopping the JW300 will be kept. If the OUT is out of the latch output area, it will be reset when the JW300 operation is stopped.
 - => See system memory #0232, #0233, #0252, and #0253.
- Use this instruction as a pair with RST instruction (next page).
- The SET instruction and RST instruction that are between MCS (F-30) and MCR (F-31) do not function when MCS (F-30) instruction condition is OFF.

RSIR	eset when I	input is risin	ig			
Symbol	R (R)					
	When the set inp	out goes ON, this i	nstruction turr	ns OFF specif	ied OUT and	l latches.
Function		F-33 RST 004000				
Use range	JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU
of relay number R	000000 to 015777 020000 to 075777	000000 to 015777 020000 to 075777 100000 to 153777		000000 to 020000 to 100000 to	075777	
[Use exam	ple]					
00	00000 004000 ⊣∣⊢────(R)			lr STR RST		
000000						
004000 (output)						
The OUT	004000 keeps its	changes from OFF ON status even if s OFF, OUT00400	reset input 00	00000 goes Ol		

- If the OUT that is specified by this instruction locates inside a latch specified area, this OUT will be latched after recovery from a power failure. If the OUT is out of the latch specified area, it will be reset after recovery from a power failure.
- If the OUT that was specified by this instruction is in the area to latch output when the JW300 stops operation, the status before stopping the JW300 will be kept. If the OUT is out of the latch output area, it will be reset when the JW300 operation is stopped.
 - => See system memory #0232, #0233, #0252, and #0253.
- Use this instruction as a pair with RST instruction (previous page).
- Using the SET instruction and RST instruction, you can control one OUT with multiple conditions.

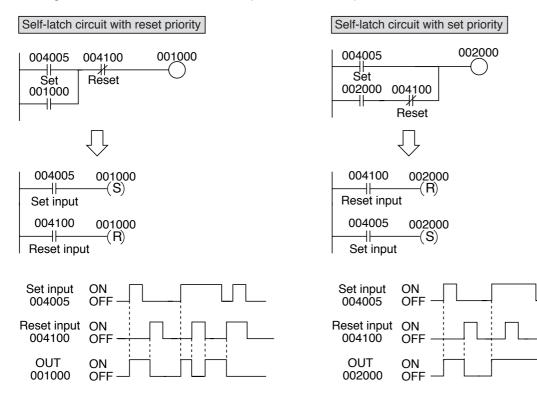


- When the SET input and RESET input are turned ON/OFF more than one time within one scan, data memory used as OUT repeats ON/OFF within one scan cycle. However, the output terminal of an output module outputs an OUT result (ON or OFF) just before I/O processing.



Even if the data memory turns ON/OFF several times while processing user program, the JW300 monitors only the result just before I/O processing.

- Using SET and RST instructions as a pair will allow simplification of self-latch circuit, etc.



PUSH	POP	MRD				
Instruction we	ord		Function			
PUSH		Stores the accumulator (ACC) and stack contents to the protected area in the internal memory area.				
POP Recovers the saved accumulator (ACC) and stack contents using PUSH instruct from the memory area. After this operation, clears the contents stored by the PU instruction.						
MRD Temporarily reads the saved accumulator (ACC) and stack contents using PU instruction from the memory area.						

- PUSH instruction shall be less than 64 in one circuit.

.

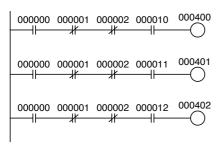
- The number of times used shall be the same for PUSH and POP. After PUSH instruction, enter POP instruction.
- MRD instruction reads only the area stored just before.

• Use example

=> Page 8-30 to 8-31

Use example 1

- When to not use PUSH, POP, and MRD instructions



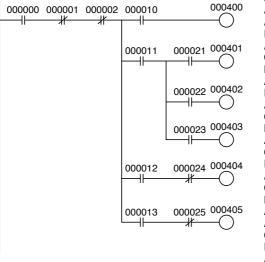
STR 000000 AND NOT 000001 AND NOT 000002 AND 000010 OUT 000400 STR 000000 AND NOT 000001 AND NOT 000002 AND 000011 OUT 000401 STR 000000 AND NOT 000001 AND NOT 000001 AND NOT 000002 OUT 000402

- When using PUSH, POP, and MRD instructions

1		STR 000000	
000000 000001 000002 000010	000400	AND NOT 000001	[PUSH]
	-0	AND NOT 000002	Make a protected area for the contents
		PUSH	of ACC and stack.
000011	000401	AND 000010	[MRD]
	$-\!\!\!\bigcirc$	OUT 000400 MBD	Reads the data in the protected area
		AND 000011	taken from the ACC and stack contents.
000012	000402	OUT 000401	
	$-\bigcirc$	POP	[POP]
		AND 000012	Recovers the data in the protected area
1		OUT 000402	taken from the ACC and stack contents
			to the previous conditions.

Instruction		Α	Accumulator ACC			-	In		mem ACC1	ory ar	rea
STR	000000	000000	+						ACCI		
AND NOT	000001	000000									
AND NOT	000002	000000	000001 #	000002 #	_						
PUSH		000000	000001 #	000002	_		` -	000000	000001	000002 #	-
AND	000010		000001 #	000002	000010	-		000000	000001 ₩	000002 #	-
OUT	000400	000000	000001 #	000002	000010	-			000001 ∦	000002 #	_
MRD			000001 #		_		\vdash	000000	000001 ∦		-
AND	000011	000000	000001 #	000002	000011	-		000000	000001 ₩	000002 #	-
OUT	000401	000000	000001 #	000002	000011	-	•	000000	000001 ∦	000002 #	-
POP			000001 #	000002	_	1					
AND	000012	000000	000001 #	000002	000012	-					
OUT	000402		000001 #	000002 #	000012	_					

Use example 2



STR 000000 AND NOT 000001 AND NOT 000002 PUSH AND 000010 OUT 000400 MRD AND NOT 000011 PUSH AND 000021 OUT 000401 MRD AND 000022 OUT 000402 POP AND 000023 OUT 000403 MRD AND 000012 AND NOT 000024 OUT 000404 POP AND 000013 AND NOT 000025 OUT 000405

Instruction	Accumulator	Internal area			
Instruction	ACC	ACC1	ACC ₂		
STR 000000					
AND NOT 000001					
AND NOT 000002	000000 00001 000002				
PUSH	000000 00001 000002				
AND 000010	000000 00001 000002 000010				
OUT 000400	000000 000001 000002 000010				
MRD	000000 000001 000002				
AND 000011	000000 00001 000002 000011 				
PUSH	000000 00001 000002 000011 	000000 000001 000002 000011			
AND 000021			000000 000001 000002		
OUT 000401			000000 000001 000002		
MRD		000000 000001 000002 000011	000000 000001 000002		
AND 000022	000000 00001 00002 000011 00022	000000 000001 000002 000011	000000 000001 000002		
OUT 000402	000000 00001 00002 000011 00022		000000 000001 000002		
POP	000000 00001 000002 000011				
AND 000023					
OUT 000403	000000 00001 00002 000011 00023				
MRD		000000 00001 000002			
AND 000012					
AND NOT000024					
OUT 000404					
POP					
AND 000013					
AND NOT 000025					
OUT 000405	000000 000001 000002 000013 00025				

Chapter 9 Description of application instructions

9-1 Application instruction hints and tips

[1] Source and destination

Data processing instruction, such as data transfer and calculation, is handled in terms of a byte or word.

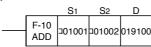
The register that contains the data before the operation is called source (S) and the register that contains the result is called destination (D).

Example 1



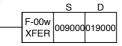
- The contents of ¬01010(S) are transferred to 009200(D).

Example 2



 The contents of ⊐01001(S₁) are added with the contents of ⊐01002(S₂) and its result is stored in 019100(D).

Example 3



- The contents of 009000(S) and 009001(S+1) are transferred to 019000(D) and 019001(D+1).

Make sure to set even address to source and destination for word processing instructions (instructions with "w") and two word processing instructions (instructions with "d").

For instructions (F-70 etc.) that operate with data memory of 2 bytes or more, and if the source (s) or destination (D) exceeds each area, the JW300 allocates the next file address as shown in the table below. => See page 2-5.

S, D	S+1, D+1		S, D	S+1, D+1
⊐01577	TMR/CNT contact area	*	E0777	E1000
-01377	(File address 00001600)			
⊐07577	TMR/CNT contact area	*	E7777	b02000
-07577	(File address 00035600)	~	109777	119000
b01777	009000		119777	129000
b03777	⊐02000			1
b37777	109000		199777	209000
009777	019000		209777	219777
019777	029000		-	
			299777	309000
099777	E0000		309777	319777
			389777	Z000

Note: Set the source and destination so that S+1 and D+1 do not enter the following areas.

1. Contact point area of TMR/CNT => See * above.

File addresses 00001600 to 00001777(8), 00035600 to 00035777(8), 00101400 to 00101777(8) (JW-32*CU), 00140400 to 00143777(8) (JW-33*CU to 362CU).

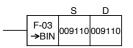
2. Out of the data memory area of each model (JW-3**CU)

[Ex. 5] File address $00074000_{(8)}$ or after when JW-31*CU is used.

3. When JW-33*CU to 362CU is used, entering from file address $00177777_{(8)}$ to $00200000_{(8)}$ (File register: FILE 1) is prohibited.

The contents of the register on the source side is not affected after the operation. Though it is possible to use the same register for the source (and destination), the contents of the source (destination, in other words) will be affected depending on the instruction used.

[Example 4]



- The contents of 009110(S) are converted into BCD 2 digits and stored in 009110(D).

The special relay area ⊐00730 to ⊐00737 is an area that is written by the CPU. Do not specify this area as the destination (D).

Use range of S and D

Use range of the source (S) and destination (D) are described in "A to C, E to H, J, and K" in the description for the application instruction. For the range of each, see below.

(1) Use range A

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
	⊐00000 to ⊐01577 ⊐02000 to ⊐07577							
	⊐10000 to ⊐15377		⊐10000 to ⊐54377					
b00000 to b01777 b02000 to b03777								
	b04000 to b07777		b04000 to	o b37777				
009000 to 099777 E00000 to E07777 109000 to 199777 209000 to 299777 309000 to 389777 Z000 to Z377 ——— *								
File register (FILE 1) ⇒	00000000 to 00077777	00000000 to 00377777	00000000 to 01777777	00000000 to 0777777	00000000 to 37777777			

* Z000 to Z377 can be used only for word instruction (F w).

When specifying an indirect address

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
	@⊐00000 to @⊐01574 @⊐02000 to @⊐07574							
	@⊐10000 to @⊐15374 @⊐10000 to @⊐54374							
@b00000 to @b01774 @b02000 to @b03774								
	@b04000 to @b07774		@b04000	to b37774				
		@00900	0 to @099774	Ļ				
		@E0000	00 to @E07774	4				
			0 to @199774					
			0 to @299774					
	@309000 to @389774							
File register @00000000 @00000000 @00000000 @000000 @000000					@00000000			
(FILE 1)	to @00077774	to @00377774	to @01777774	to @07777774	to @37777774			

[Applicable instruction words]

F-00 (S, D), F-01 (D), F-02 (D1, D2), F-03 (S, D), F-04 (S, D), etc.

(2) Use range B

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
	⊐00000 to ⊐01576 ⊐02000 to ⊐07576							
	□ 10000 to □15376 □ 10000 to □54376							
b00000 to b01776 b02000 to b03776								
	b04000 to b07776		b04000 to	o b37776				
	009000 to 099776 E00000 to E07776 109000 to 199776 209000 to 299776 309000 to 389776 Z000 to Z377 *							
File register (FILE 1) ⇒	00000000 to 00077776	00000000 to 00377776	00000000 to 01777776	00000000 to 0777776	00000000 to 37777776			

* Z000 to Z377 can be used only for word instruction (F \square w).

• When specifying an indirect address

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
	@⊐00000 to @⊐01574 @⊐02000 to @⊐07574							
	@⊐10000 to @⊐15374 @⊐10000 to @⊐54374							
	@b00000 to @b01774 @b02000 to @b03774							
	@b04000 to @b07774		@b04000	to b37774				
		@00900	0 to @099774	Ļ				
		@E0000	00 to @E07774	4				
			0 to @199774					
	@209000 to @299774							
	@309000 to @389774							
File register	@00000000	@00000000 @0000000 @0000000 @0000000						
(FILE 1) ⇒	to @00077774	to @00377774	to @01777774	to @07777774	to @37777774			

[Applicable instruction words]

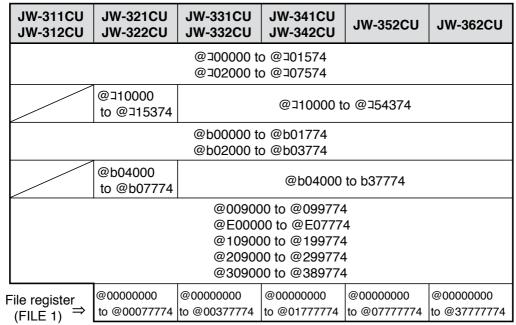
F-00w (S, D), F-01w (D), F-02w (D1, D2), F-03w (S, D), F-04w (S), etc.

(3) Use range C

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
	⊐00000 to ⊐01574 ⊐02000 to ⊐07574							
	⊐10000 to ⊐15374	710000 to 75/37/						
	b00000 to b01774 b02000 to b03774							
	b04000 to b07774		b04000 to	o b37774				
009000 to 099774 E00000 to E07774 109000 to 199774 209000 to 299774 309000 to 389774 Z000 to Z376 ——— *								
File register (FILE 1) ⇒	00000000 to 00077774	00000000 to 00377774	00000000 to 01777774	00000000 to 0777774	00000000 to 3777774			

* Z000 to Z376 can be used only for word instruction (F \square w).

When specifying an indirect address



[Applicable instruction words]

F-00d (S, D), F-02d (D1, D2), F-05w (S), F-06w (D), F-09d (S, D), etc.

(4) Use range E

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
	⊐00000 to ⊐01575 ⊐02000 to ⊐07575							
	⊐10000 to ⊐15375	110000 to 75/375						
	b00000 to b01775 b02000 to b03775							
	b04000 to b07775		b04000 te	o b37775				
009000 to 099775 E00000 to E07775 109000 to 199775 209000 to 299775 309000 to 389775 Z000 to Z376 — *								
File register (FILE 1) ⇒	00000000 to 00077775	00000000 to 00377775	00000000 to 01777775	00000000 to 0777775	00000000 to 3777775			

* Z000 to Z376 can be used only for word instruction (F \square w).

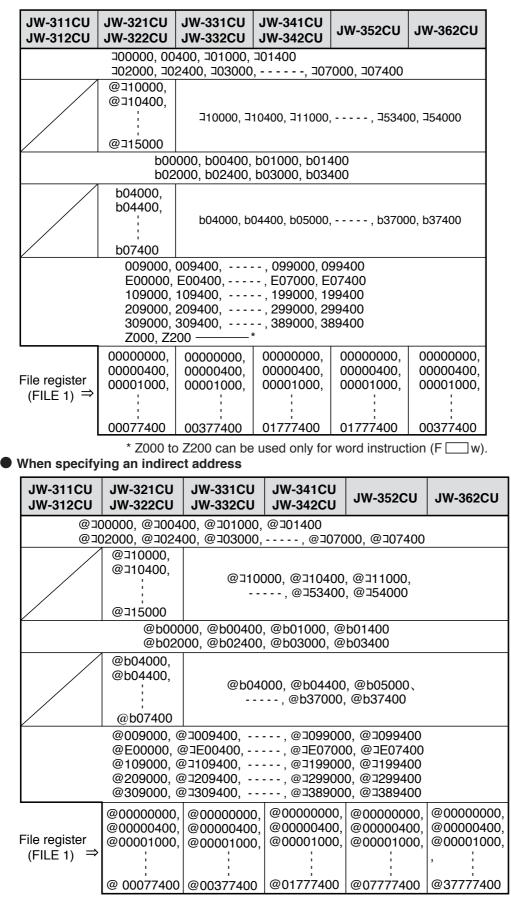
• When specifying an indirect address

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
@⊐00000 to @⊐01574 @⊐02000 to @⊐07574								
@⊐10000 to @⊐15374 @⊐10000 to @⊐54374								
@b00000 to @b01774 @b02000 to @b03774								
	@b04000 to @b07774		@b04000	to b37774				
			00 to @099774	-				
			00 to @E0777					
			0 to @199774					
	@209000 to @299774							
	@309000 to @389774							
File register	@00000000	@00000000 @0000000 @0000000 @00000000						
-	to @00077774	to @00377774	to @01777774	to @07777774	to @37777774			

[Applicable instruction words]

F-04w (S, D), F-16 (D), Fc16 (D), F-22 (S), F-23 (S), F-24 (S),etc.

(5) Use range F



[Applicable instruction words]

F-05 (D), F-05w (D), F-06 (S), F-06w (S)

(6) Use range G

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
⊐00000 to ⊐01570 ⊐02000 to ⊐07570								
	□ 10000 to □15370 □ 10000 to □54370							
b00000 to b01770 b02000 to b03770								
	b04000 to b07770		b04000 to	o b37770				
009000 to 099770 E00000 to E07770 109000 to 199770 209000 to 299770 309000 to 389770								
File register (FILE 1) ⇒	00000000 to 00077770	00000000 to 00377770	00000000 to 3777770					

• When specifying an indirect address

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
@⊐00000 to @⊐01574 @⊐02000 to @⊐07574								
	@⊐10000 to @⊐15374 @⊐10000 to @⊐54374							
@b00000 to @b01774 @b02000 to @b03774								
	@b04000 to @b07774	@b04000 to b37774						
			0 to @099774					
			0 to @E07774	-				
@ 109000 to @ 199774 @ 209000 to @ 299774								
@309000 to @389774								
File register	@00000000	@00000000 @0000000 @0000000 @00000000						
(FILE 1) ⇒	to @00077774	to @00377774	to @01777774	to @07777774	to @37777774			

[Applicable instruction words]

F-15d (D), F-16d (D), Fc16d (D), F-28 (S,D), etc.

(7) Use range H

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
⊐00000 to ⊐01572 ⊐02000 to ⊐07572								
	□ □10000 to □15372 □10000 to □54372							
b00000 to b01772 b02000 to b03772								
	b04000 to b07772		b04000 to	o b37772				
009000 to 099772 E00000 to E07772 109000 to 199772 209000 to 299772 309000 to 389772								
File register (FILE 1) ⇒	00000000 to 00077772	00000000 to 00377772	00000000 to 3777772					

• When specifying an indirect address

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
@⊐00000 to @⊐01574 @⊐02000 to @⊐07574								
	@⊐10000 to @⊐15374 @⊐10000 to @⊐54374							
@b00000 to @b01774 @b02000 to @b03774								
	@b04000 to @b07774		@b04000	to b37774				
@009000 to @099774 @E00000 to @E07774 @109000 to @199774 @209000 to @299774 @309000 to @389774								
File register (FILE 1) ⇒	@00000000 to @00077774	@00000000 to @00377774	@00000000 to @01777774	@00000000 to @07777774	@00000000 to @37777774			

[Applicable instruction words]

F-05d (S), F-116 (D), F-154(D)

(8) Use range J

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
⊐00000 to ⊐01200 ⊐02000 to ⊐07200								
	□10000 to □15000 □10000 to □54000							
b00000 to b01400 b02000 to b03400								
	b04000 to b07400		b04000 t	o b37400				
009000 to 099400 E00000 to E07400 109000 to 199400 209000 to 299400 309000 to 389400								
File register (FILE 1) ⇒	00000000 to 00077400	00000000 to 00377400	00000000 to 37777400					

• When specifying an indirect address

JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU			
@⊐00000 to @⊐01574 @⊐02000 to @⊐07574								
	@⊐10000 to @⊐15374 @⊐10000 to @⊐54374							
@b00000 to @b01774 @b02000 to @b03774								
	@b04000 to @b07774		@b04000	to b37774				
@009000 to @099774 @E00000 to @E07774 @109000 to @199774 @209000 to @299774 @309000 to @389774								
File register (FILE 1) ⇒	@00000000 to @00077774	@00000000 to @00377774	@00000000 to @01777774	@00000000 to @0777774	@00000000 to @37777774			

[Applicable instruction words]

F-176 (D)

(9) Use range K

JW-311CU JW-312CU		JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU	
000000 to 015	777 00	0000 to 015777	000000 to 015777				
020000 to 075	777 02	0000 to 075777	020000 to 075777				
	10	0000 to 153777					

[Applicable instruction words]

F-32 (OUT), F-33 (OUT), F-34 (BIT), F-35 (BIT)

[2] When specifying an indirect address

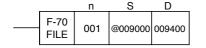
When the source (S) or destination (D) is specified using an indirect address, the file address of 3 bytes starting from the top address "file N, address n" specified in the indirect address executes operation, not the specified byte address itself. As for "file N and address n," see the next page.

File N: 00 to 80(H)

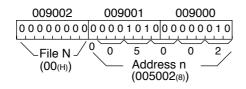
Address n: 000000 to 177777(8)

To specify an indirect address, put an "@" (at mark) in front of the byte address.

[Example]



- The content of file addresses 009000, 009001, and 009002 specified by an indirect address is transferred to 009400.



Since file address $005002_{(8)}$ of file $00_{(H)}$ in the example above is $00005002_{(8)}$ (register 019002), the result is that @009000 indicates 019002.

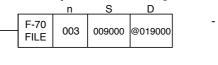
019002	009400
00011101	• 00011101

Use an even address to specify an indirect address. If an odd address is used, the address will be decrement minus one so that the same action as the odd address will be executed.

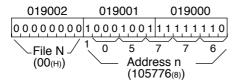
(When @009003 is set, it is treated as @009002.)

Addresses to be specified by an indirect address specification shall be within the rated data memory area of each model (JW-3**CU). If the addresses are out of the range, the JW300 does not operate.

[Prohibited example of JW-32*CU]



- The content of 3 bytes starting from 009000 shall be transferred to 3-bytes starting from a file address specified by an indirect address.



With the example above, address $105776_{(B)}$ of file $00_{(H)}$ is file address $00105776_{(B)}$. After the operation, they are transferred to 3 bytes of file address 00105776 to $00106000_{(B)}$. However, $00106000_{(B)}$ is out of the setting range (prohibited area).

Do not specify contact point areas of TMR/CNT (file address 00001600 to 00001777₍₈₎ with an indirect address.

- With the word processing instruction and double word processing instruction, specify even addresses for indirect specification file addresses. Setting an odd address will automatically subtract "1" and make it into an even address.
- With control modules JW-33*CU, JW-34*CU, JW-352CU, and JW-362CU, when the specified address exceeds the last address 17777⁽⁸⁾ of file 00^(H) they will not operate. From file 01^(H), they will continue to the top address of the next file number, and execute operation.

■ Indirect address "file N, address n"

"file N, address n" that is set by an indirect address specification and application instruction (F-100, F-101) has the following relation with the file address.

File address (a) fileN		N	_	Capacity	
File address (8)	Decimal	al Hex. N(8)		(byte)	
00000000 to 00073777			000000 to 073777	30K	1) (2) (1) (2)
00074000 to 00105777	0	00	074000 to 105777	5K 64K	
00106000 to 00177777			106000 to 177777	29K	3)
00200000 to 00277777	1	01	000000 to 077777	32K 64K	2)
00300000 to 00377777	1	01	100000 to 177777	32K	
00400000 to 00577777	2	02	000000 to 177777	64K	¥
00600000 to 00777777	3	03	000000 to 177777	64K	
01000000 to 01177777	4	04	000000 to 177777	64K	4)
01200000 to 01377777	5	05	000000 to 177777	64K	
01400000 to 01577777	6	06	000000 to 177777	64K	
01600000 to 01777777	7	07	000000 to 177777	64K	
02000000 to 02177777	8	08	000000 to 177777	64K	
02200000 to 02377777	9	09	000000 to 177777	64K	
02400000 to 02577777	10	0A	000000 to 177777	64K	
02600000 to 02777777	11	0B	000000 to 177777	64K	
03000000 to 03177777	12	0C	000000 to 177777	64K	
03200000 to 03377777	13	0D	000000 to 177777	64K	
03400000 to 03577777	14	0E	000000 to 177777	64K	5)
03600000 to 03777777	15	0F	000000 to 177777	64K	
04000000 to 04177777	16	10	000000 to 177777	64K	6)
04200000 to 04377777	17	11	000000 to 177777	64K	
04400000 to 04577777	18	12	000000 to 177777	64K	
04600000 to 04777777	19	13	000000 to 177777	64K	
05000000 to 05177777	20	14	000000 to 177777	64K	
05200000 to 05377777	21	15	000000 to 177777	64K	
05400000 to 05577777	22	16	000000 to 177777	64K	
05600000 to 05777777	23	17	000000 to 177777	64K	
06000000 to 06177777	24	18	000000 to 177777	64K	
06200000 to 06377777	25	19	000000 to 177777	64K	
06400000 to 06577777	26	1A	000000 to 177777	64K	
06600000 to 06777777	27	1B	000000 to 177777	64K	
07000000 to 07177777	28	1C	000000 to 177777	64K	
07200000 to 07377777	29	1D	000000 to 177777	64K	
07400000 to 07577777	30	1E	000000 to 177777	64K	
07600000 to 07777777	31	1F	000000 to 177777	64K	
10000000 to 10177777	32	20	000000 to 177777	64K	1 ↓

[Ex.] File address $03100000_{(8)}$ will be $n = 100000_{(8)}$ of file = $0C_{(H)}$.

1) JW-31*CU 2) JW-32*CU 3) JW-33*CU 4) JW-34*CU 5) JW-352CU 6) JW-362CU

Next page

File eddress file		fileN		Capacity			
File address (8)	Decimal	Hex.	n (8)	(byte)	From previous page		
10200000 to 10377777	33	21	000000 to 177777	64K			
10400000 to 10577777	34	22	000000 to 177777	64K			
10600000 to 10777777	35	23	000000 to 177777	64K			
11000000 to 11177777	36	24	000000 to 177777	64K			
11200000 to 11377777	37	25	000000 to 177777	64K			
11400000 to 11577777	38	26	000000 to 177777	64K			
11600000 to 11777777	39	27	000000 to 177777	64K			
12000000 to 12177777	40	28	000000 to 177777	64K			
12200000 to 12377777	41	29	000000 to 177777	64K			
12400000 to 12577777	42	2A	000000 to 177777	64K			
12600000 to 12777777	43	2B	000000 to 177777	64K			
13000000 to 13177777	44	2C	000000 to 177777	64K			
13200000 to 13377777	45	2D	000000 to 177777	64K			
13400000 to 13577777	46	2E	000000 to 177777	64K			
13600000 to 13777777	47	2F	000000 to 177777	64K			
14000000 to 14177777	48	30	000000 to 177777	64K			
14200000 to 14377777	49	31	000000 to 177777	64K			
14400000 to 14577777	50	32	000000 to 177777	64K			
14600000 to 14777777	51	33	000000 to 177777	64K			
15000000 to 15177777	52	34	000000 to 177777	64K	6)		
15200000 to 15377777	53	35	000000 to 177777	64K			
15400000 to 15577777	54	36	000000 to 177777	64K			
15600000 to 15777777	55	37	000000 to 177777	64K			
16000000 to 16177777	56	38	000000 to 177777	64K			
16200000 to 16377777	57	39	000000 to 177777	64K			
16400000 to 16577777	58	ЗA	000000 to 177777	64K			
16600000 to 16777777	59	3B	000000 to 177777	64K			
17000000 to 17177777	60	3C	000000 to 177777	64K			
17200000 to 17377777	61	3D	000000 to 177777	64K			
17400000 to 17577777	62	3E	000000 to 177777	64K			
17600000 to 17777777	63	3F	000000 to 177777	64K			
20000000 to 20177777	64	40	000000 to 177777	64K			
20200000 to 20377777	65	41	000000 to 177777	64K			
20400000 to 20577777	66	42	000000 to 177777	64K			
20600000 to 20777777	67	43	000000 to 177777	64K			
21000000 to 21177777	68	44	000000 to 177777	64K			
21200000 to 21377777	69	45	000000 to 177777	64K			
21400000 to 21577777	70	46	000000 to 177777	64K			
21500000 to 21777777	71	47	000000 to 177777	64K			
22000000 to 22177777	72	48	000000 to 177777	64K			
22200000 to 22377777	73	49	000000 to 177777	64K			
22400000 to 22577777	74	4A	000000 to 177777	64K			

Next page

File address (8)	e address @		n ₍₈₎	Capacity	
File address (8)	Decimal	Hex.	1(8)	(byte)	From previous page
			' 	· 	
22600000 to 22777777	75	4B	000000 to 177777	64K	
23000000 to 23177777	76	4C	000000 to 177777	64K	
23200000 to 23377777	77	4D	000000 to 177777	64K	
23400000 to 23577777	78	4E	000000 to 177777	64K	
23600000 to 23777777	79	4F	000000 to 177777	64K	
24000000 to 24177777	80	50	000000 to 177777	64K	
24200000 to 24377777	81	51	000000 to 177777	64K	
24400000 to 24577777	82	52	000000 to 177777	64K	
24600000 to 24777777	83	53	000000 to 177777	64K	
25000000 to 25177777	84	54	000000 to 177777	64K	
25200000 to 25377777	85	55	000000 to 177777	64K	
25400000 to 25577777	86	56	000000 to 177777	64K	
25600000 to 25777777	87	57	000000 to 177777	64K	
26000000 to 26177777	88	58	000000 to 177777	64K	
26200000 to 26377777	89	59	000000 to 177777	64K	
26400000 to 26577777	90	5A	000000 to 177777	64K	
26600000 to 26777777	91	5B	000000 to 177777	64K	
27000000 to 27177777	92	5C	000000 to 177777	64K	
27200000 to 27377777	93	5D	000000 to 177777	64K	
27400000 to 27577777	94	5E	000000 to 177777	64K	6)
27600000 to 27777777	95	5F	000000 to 177777	64K	
30000000 to 30177777	96	60	000000 to 177777	64K	
30200000 to 30377777		61	000000 to 177777	64K	
30400000 to 30577777	98	62	000000 to 177777	64K	
30600000 to 30777777	99	63	000000 to 177777	64K	
31000000 to 31177777		64	000000 to 177777	64K	
31200000 to 31377777		65	000000 to 177777	64K	
31400000 to 31577777		66	000000 to 177777	64K	
31600000 to 3177777		67	000000 to 177777	64K	
32000000 to 32177777		68	000000 to 177777	64K	
32200000 to 32377777		69	000000 to 177777	64K	
32400000 to 32577777		6A	000000 to 177777	64K	
32600000 to 32777777		6B	000000 to 177777	64K	
33000000 to 33177777		6C	000000 to 177777	64K	
33200000 to 33377777		6D	000000 to 177777	64K	
33400000 to 33577777		6E	000000 to 177777	64K	
33600000 to 33777777		6F	000000 to 177777	64K	
34000000 to 34177777		70	000000 to 177777	64K	
34200000 to 34377777		71	000000 to 177777	64K	
34400000 to 34577777		72	000000 to 177777	64K	
34600000 to 34777777		73	000000 to 177777	64K	
35000000 to 35177777	116	74	000000 to 177777	64K	

Next page

	fileN			Capacity			
File address (8)	Decimal	Hex.	n (8)	(byte)	From previous page		
	I						
35200000 to 35377777	117	75	000000 to 177777	64K			
35400000 to 35577777	118	76	000000 to 177777	64K			
35600000 to 35777777	119	77	000000 to 177777	64K			
36000000 to 36177777	120	78	000000 to 177777	64K			
36200000 to 36377777	121	79	000000 to 177777	64K			
36400000 to 36577777	122	7A	000000 to 177777	64K	6)		
36600000 to 36777777	123	7B	000000 to 177777	64K	0)		
37000000 to 37177777	124	7C	000000 to 177777	64K			
37200000 to 37377777	125	7D	000000 to 177777	64K			
37400000 to 37577777	126	7E	000000 to 177777	64K			
37600000 to 37777777	127	7F	000000 to 177777	64K			
40000000 to 40177777	128	80	000000 to 177777	64K	↓		

9-2 Index modification function

When the index modification function is applied to relays and registers that are directly specified by the basic instructions and application instructions using index register Z000 to Z377 (hereafter called to "Z_xxx"), the JW-300 operates with addresses added or subtracted by Z_xxx.

[1] Programming of index modification

For programming index modification using Z_xxx, "normal modification" and "auto modification" are available.

(1) Normal modification

Program as "relay/register, Z_xxx," and when it executes its instruction, the JW300 operates with an address adding Z_xxx to the relay or register address.

Whether addition or subtraction is determined by the upper most bit (sign). When this bit is 0, "add", and when this bit is 1, "subtract."

Detail of Z_xxx

The Z_xxx is one register and has two bytes. Set it with binary notation with sign (-77777 to $+777777_{(8)}$). The upper most bit is sign (+/-).

Detail of Z_xxx						
Binary with sign	Octal	Decimal				
0111111111111111- 011111111111111110- i						
0000000011111111-	→ +377 -	→ +255				
000000000000000000000000000000000000000	+2-	→ +2				
00000000000000000001-	+1 -	→ +1				
00000000000000000-	→ 0-	→ 0				
11111111111111111111	<u> </u>	→ -1				
1111111111111110- :	→ -2 -	→ -2				
11111110000000-	→ -400 -	→ -256				
1000000000000001-	→ -77776 -	→ -32766				
10000000000000000	→ -77777 -	→ -32767				

For negative value, take complement of 2.

¹ Sign bit (1= "-", 0 = "+")

- The "complement of 2" means a value to invert binary data (0 to 1 and 1 to 0) and add "1."

[Ex.] When 11111111 0000000,

11111111 0000000

↓ Bit inversion

00000000 11111111

↓ Add 1

0000001 0000000

 \downarrow Convert to octal notation (Sign: -)

-400(8)

(2) Auto modification

When "relay/register, Z_xxx + yy" or "relay/register, Z_xxx-yy" is programmed, after executing the same operation as the "normal modification," the JW300 adds or subtracts specified value yy (0 to 255) on the Z_xxx content (shown above).

- When yy is added to a basic instruction, the JW300 adds or subtracts an address number regardless of the accumulator (ACC) content.
- When yy is added to an application instruction, the JW300 adds or subtracts an address number only when its instruction is executed. Whether addition or subtraction is determined by the upper most bit (sign). When this bit is 0, "add", and when this bit is 1, "subtract."

[2] Index modification application area

The Z_xxx can execute index modification for the areas of data memory, TMR/CNT number and label number shown in the table below.

	Area		JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU		JW-352CU	JW-362	2CU	
*1			⊐00000 to ⊐01577(000000 to 015777)							
	Relay 2) 3)		⊐02000 to ⊐07577(020000 to 075777)						*2	
				⊐10000 to ⊐15377 (100000 to 153777)	⊐10000 to	⊐54377(100	0000 to 543	777)		
	TMR,	4)	T/C 00000 to 00777							
Data	CNT,	5)	T/C 01000 to 01777							
memory	contact	6)		T/C 02000 to 03777		T/C 0200	00 to 17777	,		
	TMR, ^{*1}		b00000 to b01777							
	CNT, MD	8)	b02000 to b03777							
	current value	9)		b04000 to b07777		b04000 te	o b37777			
	*1	10)	009000 to E07777							
	Register 11)			109000 to 389777						
TMR, CN	T number	12)	00000 to 01777 00000 to 03777 00000 to 17777							
Label r	number	13)	LB0000 to LB1777							

*1: Index modification to indirect address applicable to relay, TMR/CNT/MD current value, and register. *2: Numbers in () parentheses are relay numbers.

- Note: With the Index modification, JW300 adds and subtracts within the areas (1) to (13) of the table above.
 - When the Z_xxx added/subtracted address number is out of the range, the JW300 does not operate it.
 - With the "auto modification," if the specified value yy is repeatedly added/subtracted and if the address number will be out of the rated range, the JW300 operates within the range of the address number.

Remarks

- Do not apply the index modification to the special relays and index register Z_xxx.
- To apply the index modification to registers for application instructions having the double length calculation function, make sure to program so that the addresses after adding/subtracting shall be odd number address.
- The index modification cannot be applied to the constants of the application instructions.
- For the basic instructions having no set address such as AND STR and PUSH, the index modification cannot be applied.
- For labels that applied the index modification, if a label to jump does not exist, the JW300 does not operate.

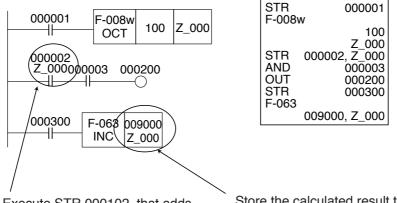
[3] Program examples of index modification

(1) Normal modification

Program "Z_xxx and relay": an address of the relay is added to or subtracted from the content of Z_xxx when executing this instruction.

Whether to add or subtract is determined by the uppermost bit (sign). When this bit is 0, it will "add." When this bit is 1, it will "subtract."

Example of addition (normal modification)



Execute STR 000102, that adds $Z_{000}(+100_{(8)})$ to relay number 000002.

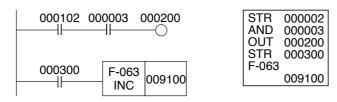
Store the calculated result to register 009100, with $-000(+100_{(8)})$ added, to register 009000.

• Detail of Z_000

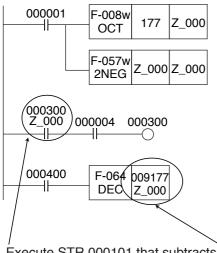
Binary value with sign	Octal
0000000 01000000	→ +100
*	

 \square Sign bit (0 = +)

- With the example above, the JW300 operates as shown below.



Example of subtraction (normal modification)

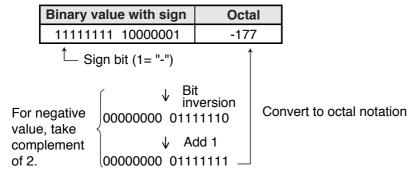


STR F-008\	000001
F-057	177 Z_000
1 0071	Z_000 Z_000
STR AND	000300, Z_000 000004
OUT STR	000300 000400
F-064	009177, Z_000

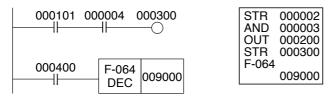
Execute STR 000101 that subtracts $Z_{-000}(-177_{(8)})$ from relay number 000300.

Store the calculated result to register 009000, with $Z_000(-177_{(8)})$ subtracted, to register 009177.

Detail of Z_000



- With the example above, the JW300 operates as shown below.

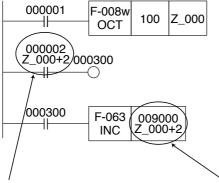


(2) In case of auto modification

Program "relays, Z_xxx + yy" or "relay, Z_xxx - yy" after operation; the specified value is added to the Z_xxx content.

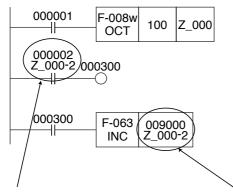
- With the basic instruction, yy is added/subtracted regardless of execute or not execute operation.
- With the application instruction, yy is added/subtracted when this instruction is executed.

Example of addition (auto)

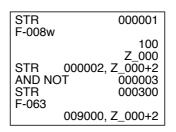


Execute STR 000102 that adds $Z_000(+100_{(8)})$ to relay number 000002. After execution of the instruction, the content of Z_000 has 2 added and becomes $+102_{(8)}$.

Example of subtraction (auto)



Execute STR 000102 that adds $Z_{000}(+100_{(8)})$ to relay number 000002. After execution of the instruction, the content of Z_000 has 2 subtracted and becomes $+076_{(8)}$.

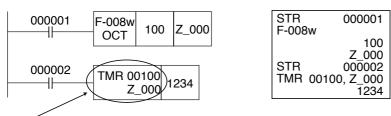


Store the calculation contents to register 09102, that has $Z_000(+102_{(8)})$ added to register 009000. After executing the instruction, the content of Z_000 has 2 added and becomes $+104_{(4)}$.[Loew1]

STR F-008w	000001
STR AND NO STR	100 Z_000 000002, Z_000-2 T 000003 000300
F-063	009000, Z_000-2

Store the calculation contents to register 009076, that has $Z_{-}000(+076_{(B)})$ added to register 009076. After executing the instruction, the content of $Z_{-}000$ has 2 subtracted and becomes $+074_{(B)}$.

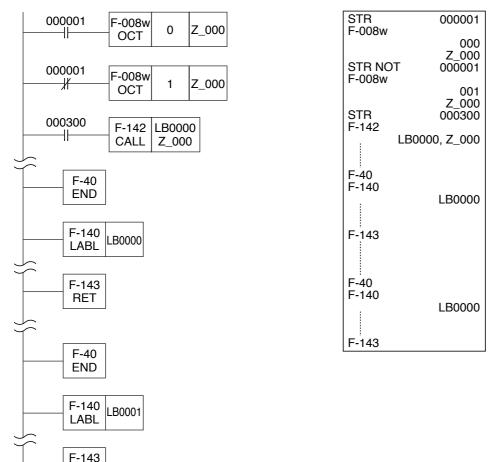
(3) Example to apply index modification to TMR/CNT number



Execute TMR number 00200 that has $Z_{000}(+100_{(8)})$ added to TMR number 00100.

RET

(4) Example to apply index modification to label number



- When the relay 000001 is ON, the JW300 shifts to subroutine of label number LB0000. When relay 000001 is OFF, it shifts to subroutine of label number LB0001.

Chapter 10 Application instructions (F-00 to Fx14d)

F-00 XFER	Tran	sfers 1-byte data			
Symbol	F-00 XFER	S D	[Explanation]	Instruction STR 004004 F-00	
Function	on Transfers the contents of the register (1 byte) to the register D.		004004 F-00 XFER 009000 ⊐00001	009000 00001	
Operation	S -> D		When the input condition 004004 changes from		
S	Use rar	ge A	OFF to ON, the contents of the register 009000 are transferred to the register ⊐00001.		
D	Use rar	ge A	009000	⊐00001	
Condition	Rising e	dge of input signal (OFF to ON)	0 1 1 1 0 1 0 1	0 1 1 1 0 1 0 1	
Contents	S	Unchanged			
after	D	Contents of register S			
operation	Flag	Unchanged			

Resembled instructions: F-00w, F-00d, F-70, F-70w, F-70d, F-74, F-74w, F-74d, F-76, F-76w, and F-76d

F-00w XFER	Tran	sfers 1-word data				
Symbol	F-00w XFER		[Explanation]	Instruction STR 004000 F-00w		
Function	Transfe S, S+1	ers the contents of the registers (1 word) to the register D.	004000 F-00w XFER 009000 -00000	009000 100000		
Operation	S, S+1 -> D, D+1		When the input condition 004000 changes from			
S	Use range B *		OFF to ON, the contents of the registers 009000 and 009001 (1 word) are transferred to the			
D	Use ran	ge B *	registers ⊐00000 and ⊐00001.			
Condition	Rising e	dge of input signal (OFF to ON)				
	S, S+1	Unchanged	0 1 1 0 1 0 1 0 1 0 1 0 1 0	0 1 1 0 0		
Contents after	D	Contents of register S		0000		
operation	D+1	Contents of register S+1	0 1 1 0 1 0 1 0 1 0 1 0 1 0 1 0	0 1 1 0 0		
	Flag	Unchanged				

* Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to use.)

-

Resembled instructions: F-00, F-00d, F-70, F-70w, F-70d, F-74, F-74w, F-74d, F-76, F-76w, and F-76d

F-00d XFER

Transfer 2-word data

Symbol	-F-00d XFER S D		[Explanation]		Instr STR F-00d	ruction 004000
Function	Transfers the contents of th S to S+3 (2 words) to the re to D+3.	•	004000 F-00d XFER	00000 =00000	F-000	009000 ⊐00000
Operation	S to S+3 -> D to D+3		When the input condition 004000 changes from OFF to ON, the contents of the registers 009000			
S	Use range C *		to 009003 (2 words) are transferred to the registers 300000 to 300003.			
D	Use range C *		C			
Condition	Rising edge of input signal (OFF to ON)	009003	009002 00 1 1 0 1 0 1 0 1 0 1		0 0 0 0 1
Contents	S to S+3 Unchanged					
after	D to D+3 Contents of registe	ers S to S+3	⊐00003 0 1 1 1 0 1 1 0 0	1 1 0 1 0 1 0 1 0 1 0 1		00000
operation	Flag Unchanged					

* Be sure to use even addresses for registers S and D.

(Odd address such as 019003 etc. are prohibited to use.)

Resembled instructions: F-00, F-00w, F-70, F-70w, F-70d, F-74, F-74w, F-74d, F-76, F-76w, and F-76d



Transfers BCD (2 digits) constant

Symbol	F-01 BCD	n D	[Explanation]	Instruction STR 004001
Function	Transfers a 2-digit BCD constant "n" to the register D.		F-01 BCD 15 009100	F-01 15 009100
Operation	n -> D		When the input condition 00400	5
n	Use range 00 to 99		OFF to ON, the BCD constant " the register 009100.	15" is transferred to
D	Use rar	nge A		
Condition	Rising edge of input signal (OFF to ON)			
Contents after	D n Flag Unchanged		009100 After transferring 0 0 0 1 0 1 0	1
operation				⊐

Resembled instructions: F-01w, F-01d, F-91

F-01w BCD	Tran	sfers BCD (4 digits)	constant		
Symbol	F-01w BCD	n D	[Explanation]	Instruction STR 004001 F-01w	
Function		ers a 4-digit BCD constant "n" egisters D, D+1.	004001 F-01W BCD 1984 019100	1984 019100	
Operation	n -> D, D+1		When the input condition 004001 changes from		
n	Use rar	nge 0000 to 9999	OFF to ON, the BCD constant "1984" is transferred to the registers 019100 and 019101.		
D	Use range B - Be sure to use even addresses for register D. (Odd address such as 019003 etc. are prohibited to use.)		The following value is containe 019100 and 019101 after the tr	•	
Condition	Rising edge of input signal (OFF to ON)		After transferring 0 0 0 1 1 0 0 1 1 0 0 0 0 1 0 0		
Contents	D, D+1	n		∕` <u>_</u> 4 <u></u> _∕	
after operation	Flag	Unchanged			

Resembled instructions: F-01, F-01d, F-91

F-01d BCD

Transfers BCD (8 digits) constant

Symbol	F-01d BCD	n D	[Explanation]	Instruction STR 004001 F-01d	
Function		rs a 8 digits BCD constant "n" egisters D to D+3.	004001 F-01d BCD 19842563 029100	19842563 029100	
Operation	Operation n -> D to D+3 n Use range 00000000 to 99999999 D Use range C - Be sure to use even addresses for register D. (Odd address such as 019003 etc. are prohibited to use.)		When the input condition 004001 changes from OFF to ON, the BCD constant "19842563" is transferred to the registers 029100 to 029103.		
n					
D			After transferring 029103 029102 0291 0 0 0 1 1 0 0 1 1 0 0 0 0 1 0 0 0 1 0 0	010101100011	
Condition	Rising e	dge of input signal (OFF to ON)	1 9 0 4 2	5 0 5	
Contents	D to D+3	n			
after operation	Flag	Unchanged			

Resembled instructions: F-01, F-01w, F-91

F-02	
XCHG	

Exchange 1-byte data between registers (eXCHanGe)

Symbol	F-02 XCHG		[Explanation]	Instruction STR 004001 F-02
Function		ntents of the register D1 are ged with the contents of the D2.	00 4001 F-02 XCHG 009000 009001	009000 009001
Operation	D1 <-> D2		When the input condition 0040 OFF to ON, the contents of the	register 009000 are
D1	Use range A		exchanged with the contents o	f the register 009001.
D2	Use rar	nge A	Before operation	After operation
Condition	Rising e	dge of input signal (OFF to ON)	009000 0 0 0 0 0 0 1 0	1000000
Contents	D1	Contents of register D ₂		<
after	D2	Contents of register D1	009001 1 0 0 0 0 0 0 0 /	
operation	Flag Unchanged			

Resembled instructions: F-02w, F-02d, F-174

F-02w XCHG

Exchange 1-word data between registers (eXCHanGe)

Symbol	F-02w		[Explanation]	Instruction STR 004000 F-02w
Function	The contents of the registers D_1 , D_{1+1} (1 word) are exchanged with the contents of the registers D_2 , D_2+1 .		004000 F-02w XCHG 009000 019000	009000 019000
Operation	D, D1+1	l <-> D2, D2+1	When the input condition 0040	<u> </u>
D1	Use range B - Be sure to use even addresses for registers D1.		OFF to ON, the contents of the and 009001 (1 word) are exch contents of the registers 01900	anged with the
D2	Use rar - Be sure D2.	nge B to use even addresses for registers	Before operation	After operation
Condition	Rising e	dge of input signal (OFF to ON)		
	D1	Contents of the register D ₂		
Contents	D1+1	Contents of register D ₂ +1	019000 1 1 0 0 1 1 0 1 019001 0 1 0 1 0 0 1 1	10011010
after	D2	Contents of the register D1		
operation	D2+1	Contents of register D1+1		
	Flag	Unchanged		

Resembled instructions: F-02, F-02d, F-174

F-02d XCHG

Exchange 2-word data between registers (eXCHanGe)

Symbol	F-02d XCHG		[Explana	ation]		Instruction STR 004000
Function	The contents of the registers D1 to D1+3 (2 words) are exchanged with the contents of the registers D2 to D2+3 (2 words).		When the input condition 00400 OFF to ON, the contents of the 009003 (2 words) are exchange		004000 of the r hange	registers 009000 to d with the contents
Operation	D1 to D1+3 <-> D2 to D2+3			gisters 019000 to	01900	, , , , , , , , , , , , , , , , , , ,
Dı	Use range C - Be sure to use even addresses for registers D1.		009000	Before operation		After operation 1 1 0 0 1 1 0 1 0 1 0 1 0 0 1 1
D2	Use ran - Be sure D2.	ge C to use even addresses for registers		1 1 0 1 0 1 0 0 0 1 0 0 0 0 1 1		0 0 1 1 0 0 1 0
Condition	Rising e	dge of input signal (OFF to ON)	019000	11001101	A	00010100
Contents	D1 to D1+3	Contents of registers D ₂ to D ₂ +3		0 1 0 1 0 0 1 1		
after	D2 to D2+3 Contents of registers D1 to D1+3		00110010		1 1 0 1 0 1 0 0	
operation	Flag	Unchanged	019003	00100001		

Resembled instructions: F-02, F-02w, and F-174



Convert 2-digit BCD to 8-bit binary

Symbol		— F-03 →BIN S D	[Explanation] Instruction STR 004006		
Function		The contents of the register S (8 bits) are assumed as a BCD code, converts into a binary equivalent, then store the result in the register D.	$ \begin{array}{c c} & F-03 \\ \hline & & \\ \hline \hline & & \\ \hline \hline & & \\ \hline \hline \\ \hline & & \\ \hline \hline \\ \hline \\$		
Operation		S ->D	bits) are assumed as a BCD code, converts into a binary equivalent, and its result is transferred to the		
S		Use range A	register 009310. The contents of the register 009300 remain		
D		Use range A	unchanged. If the contents of the register 009310 stay the same as before and the error flag is set to		
Condit	ion	Rising edge of input signal (OFF to ON)	"1." - Ex.1 009300 009310		
	S	Unchanged			
Contents after operation	D	Calculation result - Unchanged when the contents of the register S are not a BCD code.	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$		
	Flag	Contents of register S Zero 007357 Carry 007356 Error 007355 Non-carry 007354 BCD code 0 0 1 1 Not BCD code 1 1 1	- Ex.2 11000101 11000101 The contents of 009310 stay the same.		
Resembled instructions: F-03, F-03w, F-53, F-153			C - 5 - The contents of 009310 stay the same. Zero Carry Error Non-carry 007357 007356 007355 007354 0 0 1 0		

F-03w →BIN

Convert 4-digit BCD to 16-bit binary

Symbol		F-03w →BIN S D		[Explanation]	Instruction STR 004001
Function		The 2-byte BCD contents of the registers S, S+1 are converted into binary, and the result is stored in the 2-byte area of the registers D, D+1.		$\begin{array}{c c} & 004001 \\ \hline & F-03w \\ \hline & BIN \end{array} \xrightarrow{101000} 019000 \\ \hline & When the input condition 00400 \\ \hline & 0.555 \\ \hline & 0.0100 \\$	s.
Operation		S1, S+1 -> D, D+1		OFF to ON, the contents of 4-digit BCD in registers ¬01000 and ¬01001 are converted into a binary equivalent, and its result is transferred and stored in 2-byte area of registers 019000	
S		Use range B			
D)	Use range B		and 019001.	
Cond	lition	Rising edge of input signal (OFF to ON)		Before operation ———	→ After operation
	S, S+1	Unchanged		Tens Ones □01000 1 0 1 0 1 0 <	019000 0 0 0 0 0 0 0 0
Contents	D	Result (0 to 255)	Unchanged when the contents of the	9 6 Thousands Hundreds	2 ⁷ 2 ⁰
after operation	D+1	Result (256 to 9999)	registers S, S+1 are not a BCD code.		$\begin{array}{c} 019001 \\ 0 \\ 2^{15} \\ 2^{8} \end{array}$
	Flag		Carry Error Non-carry 07356 007355 007354 0 0 0 1 0 0	4 0 BCD 4096	2 ¹⁰ 2 ⁰ BIN (Binary) 2 ¹² =4096

- If the F-53 instruction is used for programming, the F-03w instruction displays the program during monitoring. Resembled instructions: F-03, F-03d, F-53, F-153

F-03d →BIN

Convert 8-digit BCD to 32-bit binary

Sym	bol	F-03d →BIN S D	[Explanation] Instruction STR 004001 F-03d			
Function		The 4-byte BCD 8-digit contents of th registers S to S+3 are converted into binary, and the result is stored in the 4-byte area of the registers D to D+3	e 1004001 F-03d 101000 029000 029000 When the input condition 004001 changes from			
Opera	ation	S to S+3 -> D to D+3	OFF to ON, the contents of 8-digit BCD in registers J01000 to J01003 are converted into			
S		Use range C	a binary equivalent, and its result is transferred and stored in 4-byte area of registers 029000			
D	1	Use range C	to 029003. Before operation — After operation			
Cond	ition	Rising edge of input signal (OFF to O	N) Tens Ones =01000 0 1 1 0 0 1 0 0 029000 0 0 0 0 0 0 0 0 0			
	S to S+3	Unchanged	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			
	D	0 to 255 Unchanged whe	Thousands Hundreds			
Contents	D+1	256 to 65280 the contents of t				
after operation	D+2	65536 to 16711680 are not a BCD	10 ⁵ -digit 10 ⁸ -digit			
oporation	D+3	16777216 to 99999999				
	Flag	$\begin{array}{c c c c c c c c c c c c c c c c c c c $				
nesem		structions: F-03, F-03w, F-53, F-153	BCD BIN (Binary) 67108864 2 ²⁶ =67108864			

F	-04
->	BCD

Convert 8-bit binary to 2-digit BCD

-> BCD				
Symbol	> E	CD S D	[Explanation]	Instruction STR 004006 F-04
Function	are a	ontents of the register S (8 bits) ssumed as a binary code, erts into a BCD equivalent, then the result in the register D.	$\begin{array}{c c} 004006 & \hline F-04 & 009320 & 009330 \\ \hline & & & \\ \hline \hline & & & \\ \hline \\ \hline$	009320 009330 06 changes from register 009320 (8
Operation	S -> [)	bits) are assumed as a binary of a BCD code, and its result is tra	ansferred to the
S	Use r	ange A	register 009330. The contents 009320 remain unchanged.	of the register
D	Use r	ange A	If the BCD number converted s the digit of hundreds will be ign	
Condition	Rising	edge of input signal (OFF to ON)		
Contents	s	Unchanged	- Ex.1 009320	009330
after	D	Result	$2^{5} + 2^{3} + 2^{2} + 2^{0}$ (32 + 8 + 4 + 1)	
operation	Flag	Unchanged	и 45	
F-04w	Co	nvert 16-bit binary to	2 ⁷ +2 ⁶ +2 ⁴ +2 ³ +2 ¹ +2 ⁰ "219 6-digit BCD	
-> BCD	00			
Symbol	F-0 > E	^{4w} S D	[Explanation]	Instruction STR 004001 F-04w
Function	The contents of 2-byte binary in the registers S, S+1 converted into a 6-digit BCD, and the result is stored in 3 bytes of the registers D, D+1, and D+2.		When the input condition 00400 OFF to ON, the contents of 2-b registers $\exists 01000$ and $\exists 01001$ a	□01000 019000 01 changes from byte binary in
Operation	S, S+1 ->D, D+1, D+2		6-digit BCD and stored in 3-byt	
S	Use r	ange B	019000 to 019002.	
D	Use r	ange E	Before operation	After operation Tens Ones
Condition	Rising	edge of input signal (OFF to ON)		000 0 1 1 0 1 0 0 0
	S, S+ ⁻	Unchanged	2 ⁷ 2 ⁰	6 8 Thousands Hundreds

- If the F-54 instruction is used for programming, the F-04w instruction displays the program during monitoring. Resembled instructions: F-04, F-04d, F-54, F-154

D

D+1

D+2

Flag

Contents

operation

after

Result (ones and tens)

Result (ten thousands)

Unchanged

Result (hundreds and thousands)

⊐01001 1 0 0 0 0 0 0 0

2¹⁵

BIN

2⁸

7 Ten thousands

3

019001 0 0 1 0 0 1 1 1

2

019002 0 0 0 0 0 0 1 1

0

6 digits BCD

032768



Convert 32-bit binary to 10-digit BCD

Symbol	F-040		[Explanation]	Instruction STR 004100
Function	the reg BCD, a	ntents of 4-byte 32-bit binary in isters S to S+3 converted into a nd the result is stored in 5 bytes egisters D to D+4.	004100 F-04d 019000 009000 -> BCD 019000 009000 When the input condition 00555 00555	n 004100 changes from
Operation	S to S-	-3 -> D to D+4	OFF to ON, the contents 019000 to 019003 (32-b	it) are converted into a
S	Use ra	nge C *	BCD and stored in area 009004.	of registers 009000 to
D	Use ra	nge H *	Before operation ——	→ After operation
Condition	Rising e	edge of input signal (OFF to ON)	019000 0 0 0 0 0 0 0 0	Tens Ones 0 1 0 0 1 0 0 0 009000
	S to S+3	Unchanged	2^7 2^0	4 8
	D	Result (ones and tens)		Thousands Hundreds
Contents	D+1	Result (hundreds and thousands)	019001 10000000	0 0 1 1 0 1 1 0 009001
after	D+2	Result (10 ⁴ and 10 ⁵ digits)	2 ¹⁵ 2 ⁸	3 6 10⁵digit 10⁴digit
operation	D+3	Result (10 ⁶ and 10 ⁷ digits)	019002 0 0 0 0 0 0 0 0 0	0 1 0 0 1 0 0 009002
	D+4	Result (10 ⁸ and 10 ⁹ digits)	2 ²³ 2 ¹⁶	4 8
	Flag	Unchanged	019003 1 0 0 0 0 0 0 0	10 ⁷ digit 10 ⁶ digit
* Be sure to use even addresses for registers S and D. Resembled instructions: F-04, F-04w, F-54, F-154			$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$

2 1

BCD

BIN

Demultiplexes 1-byte data **DMPX** (DeMultiPleXer) F-05 [Explanation] Instruction Symbol S D DMPX 004005 STR F-05 The contents of the register S+1 are 004005 F-05 □00000 ⊐00000 009000 transferred to the register of which DMPX 009000 address is implied by the register D Function (reference address) modified by the 009000 (Reference address) contents of the register S (data pointer). Data pointer (DP) ⊐00000 0 0 0 0 0 1 0 1 +Data pointer S+1 -> D+ (S) Data (005(8)) ⊐00001 Operation Data pointer (DP) 1<u>,0,1,0,1,1,0,0</u> Reference address Destination register 009005 10101100 S Use range B D Use range F The following transfer takes place when the input Condition Rising edge of input signal (OFF to ON) condition 004005 changes from OFF to ON. The data in 100000+1, that is, the data in 100001 is Unchanged S, S+1 transferred to the address 009005 which obtained after modifying the reference address (009000) by the contents Contents D Unchanged of the data pointer J00000 (005(8)). after Since the data pointer is represented by an octal number Contents of the register S+1 D+<S> operation it may assume a number between 000 to 377(8). In the above example, it would be possible to demultiplex data Flag Unchanged to address, 009000 to 009377, by varying the data pointer value in reference to the reference address.

Resembled instructions: F-05w, F-05d, F-73, F-73w

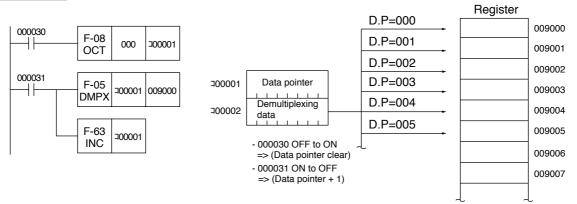
F-05

- Although it is possible to program other than the block top address to D (reference address), of that address contained block for the reference address.

[Example]	D setting	Operational reference address
	⊐00050	⊐00000
	b00210	b00000
	009105	009000
	00033210	00033000

⇒ See page "Data memory block and reference address."

Reference A programming example to change the demultiplexing address by varying the data pointer.



- (1) When 000030 is changed from OFF to ON, the octal constant 000(8) is transferred to 300001. (Data pointer 000(8))
- (2) When 000031 is changed from OFF to ON, the contents of ⊐00002 are transferred to (009000+000(8)=009000). The contents of 300001 changed to 001(8) as it is incremented plus 1 by the F-63 instruction.
- (3) When 000031 is changed from OFF to ON again, the contents of 300002 are transferred to (009000+001(8) =009001). The contents of 300001 change to 002(8) as it is incremented plus 1 by the F-63 instruction.

Thereafter, the contents of ⊐00002 are demultiplexed to registers up to 009377.

F-05wDemultiplexs 1-word dataDMPX(DeMultiPleXer)

Symbol	- F-05w S D	[Explanation] Instruction STR 010000
Function	The contents of the registers S+2, S+ are transferred to 2-byte area of the registers of which address is implied by the register D (reference address) modified by the contents of the register S (data pointer).	Deta pointer (contents are
Operation	S+2, S+3 -> D+ <s> D+<s> +1 Data pointer (DP) Reference address</s></s>	□01230 0,0,0,1,0,1,1,0 026(8) 019400 (Reference address)
S	Use range C - Be sure to use even addresses for register S. (Odd address such as ⊐00011 etc. are prohibited to use.) - Set word addresses (000 to 376(8)) to register S	$ \begin{array}{ c c c c c c } \hline \Box 01231 & \text{Not used} \\ \hline \Box 01232 & 8 & 4 \\ \hline \Box 01233 & 1 & 9 \\ \hline \end{array} \\ \hline \end{array} \\ \begin{array}{ c c c c c c c c c c c c c c c c c c c$
D	Use range F	
Condition	Rising edge of input signal (OFF to ON)	The following transfer takes place when the input condition 010000 changes from OFF to ON.
	S, S+1, S+2, S+3	- The data in $\exists 01230+2$, $\exists 01230+3$, that is, the data in $\exists 01232$ and $\exists 01233$ are transferred to the 2-
	D Unchanged	byte area from the address 019426 which
Contents after	D+ <s> Contents of the register S+</s>	
operation	D+ <s> +1 Contents of the register S+</s>	3 (026 ₍₈₎).
	Flag Unchanged	

Resembled instructions: F-05, F-05d, F-72, F-72w, F-72d

F-05d DMPX

Demultiplexs 2-word data (DeMultiPleXer)

Symbol	-F-05d S D	[Explanation] Instruction STR 010000 F-05d
Function	The contents of the registers S+2 to S+5 are transferred to 4-byte area of th registers of which address is implied by the register D (reference address) modified by the contents of the register S (data pointer).	e → DMPX → 1230 019400 → 101230 019400 → 101230 019400
Operation	S+2 to S+5 -> D+ <s> to D+ <s> +1 Data pointer (DP) Reference address</s></s>	□ □1230 □ 0 0 1 0 1 1 0 0 □ 01231 Not used □ 019400 (Reference address) □ 01231 Not used □ 019400 (Reference address)
S	Use range H - Be sure to use even addresses for register S. (Odd address such as ⊐00011 etc. are prohibited to use.) - Set word addresses (000 to 376(8)) to register	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
D	Use range F	□01235 <u>2 7</u> <u>3 6</u> 019430
Condition	Rising edge of input signal (OFF to ON)	2 7 019431
	S to S+5 Unchanged	
	D Unchanged	The following transfer takes place when the input condition 010000 changes from OFF to ON.
Contents	D+ <s> Contents of the register S+:</s>	2 - The data in ⊐01230+2 to ⊐01230+5, that is , the
after	D+ <s>+1 Contents of the register S+</s>	data in ¬01232 to ¬01235 are transferred to the 4- byte area from the address 019426 which
operation	D+ <s>+2 Contents of the register S+</s>	obtained after modifying the reference address
	D+ <s>+3 Contents of the register S+</s>	
	Flag Unchanged	(026(8)).

Resembled instructions: F-05, F-05w, F-72, F-72w, F-72d

Multiplex 1-byte data (MultiPleXer)

	<u>`</u>	,		
Symbol	F-06 MPX	S D	[Explanation]	Instruction STR 004005 F-06
Function	addres (referen content	ntents of the register whose s is implied by the register S nce address) modified by the s of the register D (data), are transferred to the register	the register whose ed by the register S ess) modified by the egister D (data insferred to the register +Data pointer (100(8)) - 2 - 2 - 100(8) 0 - 100 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
Operation	S+ <d> -> D+1 Data pointer (DP) Reference address</d>		$\begin{array}{c c c c c c c c c c c c c c c c c c c $	
S	Use range F		The following transfer takes place when the input	when the input
D	Use rar	ige B	condition 004005 changes from OFF to ON. The contents of the address 009100 is transferred to ⊐00000+1 (⊐00001) which is obtained by modifying the reference address 009000 by the contents of the data	
Condition	Rising e	dge of input signal (OFF to ON)		, , ,
	S	Unchanged	pointer ⊐00000 (100(8)). - Since the data pointer is represe	nted by an octal
Contents after	D	Unchanged (data pointer)	number it can assume a number between 000 377(8). In the above example, it would be poss multiple data to address 009000 to 009377, by the data pointer value in reference to the refer	
operation	D+1	Contents of the register S+ <d></d>		to 009377, by varying
	Flag	Unchanged	address.	

Resembled instructions: F-06w, F-06d, F-73, F-73w

- Although it is possible to program other than the block top address to S (reference address), the programmable controller assumes for its operation the top address of that address contained block for the reference address.

[Ex.]

F-06 MPX

S setting	Operational reference address	
⊐00051	⊐00000	
b00106	b00000	
009023	009000	
00031257	00031000	

 \Rightarrow See page "Data memory block and reference address."

F-06w MPX		iplex 1-word data tiPleXer)		
Symbol	F-06w MPX	S D	[Explanation] Instruction STR 010001 F-06w	
Function	whose a register by the c	yte contents of the register address is implied by the S (reference address) modified ontents of the register D (data , are transferred to the registers -3.	010001 F-06w 009000 00036 009000 100036 100036 Data pointer (contents are even address only	nts
Operation		S+ <d> +1 -> D+2, D+3 Data pointer (DP) Reference address</d>	+Data pointer (126(8))	
S	Use ran	ge F	5 6 ⊐00040	40
D	(Odd ad prohibite	ge C to use even addresses for register D. dress such as ⊐00011 etc. are ed to use.) d addresses (000 to 376(8)) to register D.	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	
Condition	Rising e	dge of input signal (OFF to ON)	condition 010001 changes from OFF to ON. - The contents of the address 009126 and 009127	
	S	Unchanged	are transferred to $\exists 00040 (\exists 00036+2)$ and $\exists 00041 (\exists 00036+3)$ which is obtained by	,
Contents	D, D+1	Unchanged	modifying the reference address 009000 by the contents of the data pointer ¬00036 (126(8)).	
after operation	D+2	Contents of the register S+ <d></d>		
operation	D+3	Contents of the register S+ <d>+1</d>		
	Flag	Unchanged		

Resembled instructions: F-06, F-06w, F-73, F-73w, F-73d

F-06d MPX		iplex 2-word data tiPleXer)						
Symbol	F-06d MPX	S D	[Explanation]			Instruction STR 010001 F-06d		
Function	whose a S (reference) contents	yte contents of the register address is implied by the register ence address) modified by the s of the register D (data pointer), sferred to the registers D+2 to	Data pointer (contents are even address only			0036 (contents		
Operation		o S+ <d>+3 -> D+2 to D5 Data pointer (DP) Reference address</d>	Reference address 009000 + Data pointer (126(9)		126(8)	0 1 0 1	011(0 ⊐00036 ⊐00037
S	Use ran	ge F		-		5	6	⊐00040
D	(Odd add prohibited	o use even addresses for register D. Iress such as ⊐00011 etc. are	009126 009127 009130	5 6 7 8 9 2		7 9 4	8 2 3	⊐00041 ⊐00042 ⊐00043
Condition	Rising e	dge of input signal (OFF to ON)	009131	4 3				
	S	Unchanged	4	-	L			
	D, D+1	Unchanged	The following t					input
Contents	D+2	Contents of the register S+ <d></d>	 condition 010001 changes from OFF to ON. The contents of the address 009126 to 009131 are 					131 are
after operation	D+3	Contents of the register S+ <d>+1</d>	transferred to ⊐00040 (⊐00036+2) to ⊐00043 (⊐00036+5) which is obtained by modifying the reference address 009000 by the contents of the					
oporation	D+4	Contents of the register S+ <d>+2</d>						
	D+5	Contents of the register S+ <d>+3</d>	data pointer		-			
	Flag	Unchanged						

Resembled instructions: F-06, F-06w, F-73, F-73w, F-73d

F-07 DCML		Transfers decimal (1 byte) constant (DeCiMaL)						
Symbol	F-07 DCML	n D	[Explanation]	Instruction STR 004004 F-07				
Function		nal constant "n" is transferred egister D.	F-07 DCML 015 009100	015 009100				
Operation	n D		When the input condition 004004 changes from OFF to ON, the decimal constant "15" is transferred to the register 009100.					
n	Use ran	ge 000 to 255						
D	Use ran	ge A	Register 009100 becomes the binary codes.	following values of				
Condition	Rising e	dge of input signal (OFF to ON)						
After	D	n (000 to 255)	009100 0 0 0 0 1 1 1 1					
operation	Flag Unchanged		2 ³ +2 ² +2 ¹ +2 ⁰ =15	5				

Resembled instructions: F-07w, F-07d, F-97

F-07w DCML		sfer decimal (1 wore ciMaL)	d) constant			
Symbol	F-07w DCML	n D	[Explanation]	Instruction STR 004001 F-07w		
Function		al constant "n" is transferred egisters D, D+1.	004001 F-07w DCML 22659 019100	22659 019100		
Operation	n-> D, D)+1	When the input condition 004001 changes from			
n	Use ran	ge 00000 to 65535	OFF to ON, the decimal constant 22659 is transferred to the registers 019100 and 019101.			
D	(Odd add	ge B to use even addresses for register D. dress such as 019003 etc. are d to use.)	After transferring 019101 019100 $2^{14}+2^{12}+2^{11}+2^{7}+2^{1}+2^{0}=22659$			
Condition	Rising edge of input signal (OFF to ON)			TZ -22039		
Contents	D, D+1	n				
after operation	Flag	Unchanged				

Resembled instructions: F-07, F-07d, F-97

F-07d DCML		sfer decimal (2 word iMaL)	d) constant			
Symbol	-F-07d DCML	n D	[Explanation]	Instruction STR 004001 F-07d		
Function		al constant "n" is transferred gisters D to D+3.	004001 F-07d DCML 2563074179 019100	2563074179 019100		
Operation	n -> D to) D+3	When the input condition 004001 changes from OFF to ON, the decimal constant 2563074179 is transferred to the registers 019100 to 019103.			
n	Use ran	ge 0000000000 to 4294967295				
D		o use even addresses for register D. ress such as 019003 etc. are		9101 019100		
Condition	Rising ec	dge of input signal (OFF to ON)	1 0 0 1 1 0 0 0 1 1 0 0 0 1 0 1 0 1 0 1			
Contents after	D to D+3	n	$ \begin{array}{c} 2^{31} + 2^{28} + 2^{27} + 2^{23} + 2^{22} + 2^{18} + 2^{16} + 2^{14} + 2^{13} + 2^{11} + 2^7 + 2^1 + 2^0 \\ = 2563074179 \end{array} $			
operation	Flag	Unchanged				

Resembled instructions: F-07, F-07w, F-97

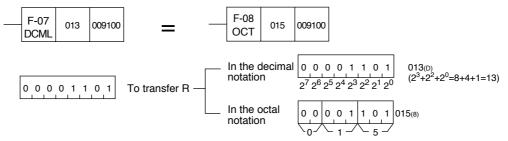
F-08	
OCT	

Transfer octal (1 byte) constant (OCTal)

Symbol	F-08 OCT	n D	[Explanation]	Instruction STR 004004	
Function		al constant "n" is transferred egister D.	004004 F-08 OCT 015 009100	F-08 015 009100	
Operation	n → D		When the input condition 004004 changes from OFF to ON, the octal constant 015 is transferred to the register 009100.		
n	Use rai	nge 000 to 377(8)			
D	Use rai	nge A	The register 009100 is in the fo representation.	llowing code	
Condition	Rising e	dge of input signal (OFF to ON)			
Contents	D	n (000 to 377(8))	009100 0 0 0 1 1	0_1	
after operation	Flag Unchanged				

Resembled instructions: F-08w, F-08d, F-71, F-71w

Reference Though F-07 (transfer decimal constant) handles a decimal number and F-08 (transfer octal constant) an octal number, the contents of the register after the transfer are represented in the binary code for both instructions.



When the F-08 instruction is used to preset the data pointer of F-05 (demultiplex) and F-06 (multiplex), you will be able to make direct recognition of the data memory address (octal).

F-08w OCT	Tran (OC)	sfer octal (1 word) c Fal)	constant		
Symbol	F-08w OCT	n D	[Explanation]	Instruction STR 004001 F-08w	
Function		l constant "n" is transferred egister D, D+1.	004001 F-08w OCT 123456 019100	123456 019100	
Operation	n -> D, I	D+1	When the input condition 0040	01 changes from	
n	Use rar	nge 000000 to 177777(8)	When the input condition 004001 changes from OFF to ON, the octal constant 123456 is transferred to the registers 019100 and 019101. After transferring 1010011110 10100101110		
D	(Odd ad	nge B to use even addresses for register D. dress such as 019003 etc. are ed to use.)			
Condition	Rising edge of input signal (OFF to ON)				
Contents after	D, D+1	n			
operation	Flag	Unchanged			

Resembled instructions: F-08, F-08d, F-71, F-71w

F-08d
OCTTransfer decimal (2 words) constant
(OCTal)

Symbol	F-08d OCT	n D	[Explanation]	Instruction STR 004001	
Function		l constant "n" is transferred egister D to D+3.	004001 F-08d OCT 12345670000 019100	F-8d 12345670000 019100	
Operation	n -> D to	o D+3	transferred to the registers 019100 to 019103. After transferring $019103 019102 019101 019100$ $0 1 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 $		
n	Use rang	ge 000000000000 to 37777777777(8)			
D	(Odd ad	ige C to use even addresses for register D. dress such as 019003 etc. are ed to use.)			
Condition	Rising e	dge of input signal (OFF to ON)			
Contents	D to D+3	n			
after operation	Flag	Unchanged			

Resembled instructions: F-08, F-08w, F-71, F-71w

F-09 INV		Complements 8-bit data (INVerter)						
Symbol	F-09 INV	S D	[Explanation]	Instruction STR 004002 F-09				
Function		ntents of the register S are d and stored in the register D.	004002 F-09 INV 009000 009003	009000 009003				
Operation	S→D		When the input condition 004002 changes from OFF to ON, the 8 bits contents of the register					
S	Use rar	ige A	009000 are inverted and its result is stored in the register 009003.					
D	Use rar	ige A	009000	009003				
Condition	Rising e	dge of input signal (OFF to ON)		10101100				
Contents	S Unchanged		The contents of the register 009000 remain					
after operation	D	Inversion of register S contents	unchanged.					
	Flag	Unchanged						

F-09w INV		plements 16-bit dat erter)	a			
Symbol	F-09w INV	S D	[Explanation]	Instruction STR 004000 F-09w		
Function	(16 bits	ntents of the registers S, S+1) are inverted and stored in the 's D, D+1.	04000 F-09w 009000 ⊐00000 INV 009000 ⊐00000	009000 700000		
Operation	S, S+1	-> D, D+1	When the input condition 004000 changes from OFF to ON, the contents of the registers 009000			
S	Use rar	ge B *	and 009001 (16 bits) are inverted and its result is stored in the registers ¬00000 and ¬00001.			
D	Use rar	ge B *				
Condition	Rising e	dge of input signal (OFF to ON)				
	S, S+1	Unchanged				
Contents after	D	Inversion of register S contents	The contents of the registers 009000 and 009001			
operation	D+1	Inversion of register S+1				
	Flag	Unchanged	remain unchanged.			

* Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to use.)

F-09d INV	Com (INVe	olements 32-bit data erter)	a			
Symbol	F-09d INV	S D	[Explanation]	Instruction STR 004000 F-09d		
Function	S+3 (32	tents of the registers S to bits) are inverted and stored gisters D to D+3.	004000 F-09d INV 009000 ⊐00000	009000 700000		
Operation	S to S+3	-> D to D+3	When the input condition 004000 changes from OFF to ON, the contents of the registers 009000 to			
S	Use rang	e C *	009003 (32 bits) are inverted and its result is stored in the registers \(\gamma\)00000 to \(\gamma\)00003.			
D	Use rang	e C *	009003 009002 0090	001 009000		
Condition	Rising ed	ge of input signal (OFF to ON)) 1 1 1 0 0 1 1 0 0 1 1 0 1 0 1 0 1 0 1			
Contents	S to S+3	Unchanged				
after	D to D+3	Inversion of registers S to S+3				
operation	Flag	Unchanged				
* Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to use.)			The contents of the registers 00 remain unchanged.	09000 to 009003		

I

ADD		(ADD)									
Symbol	I	- F-10 S1 S2 D					[Explanation]	Inst STR F-10	ruction 004000		
Function	n	The contents of the register S1 are added with the contents of the register S2 in BCD 2 digits and its result is stored in the register D.				egister	$\begin{array}{c c} & & & & & & \\ \hline & & & & & \\ \hline & & & & &$				
Operatio	n	S1+S2 -> D					are added with the contents of	the regis	ster 009010		
S1		Use range A					and its result is stored in the register 009020. The contents of registers 009000 and 009010 remain				
S2		Use range A	L L				unchanged.				
D		Use range A					- Transition of result and flags				
Conditio	n I	Rising edge	of inpu	t signal	(OFF	to ON)					
	S1	Unchange	əd				(009000) 23				
	S2	Unchange	əd								
Contents	D	 Lower 2 digits of the result. Unchanged when the contents of registers S1 and S2 are not BCD code. (Does not calculate.) 					(009020) 17 58 58 81 81 04 04 00 70 70 (23+35) (23+58) (23+81) (23+77) No subtraction Non-carry flag (23+77) (23+77) No subtraction 007354 are other than BCD) are other than BCD) 007355 (23+58) (23+77)				
after operation		Result	Zero 07357	Carry 07356	Error 07355	Non-carry 07354	Carry flag				
operation		0	1	0	0	1	Zero flag				
	Flag	1 to 99	0	0	0	1	007357		can time, max.		
		100	1	1	0	0			-, '		
		101 and above S1 and S2 are not BCD code.	0	1 0	0	0					

F-10 Add register and register (BCD 2 digits)

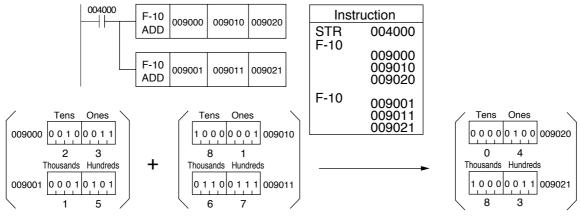
- If the contents of S1 and S2 are numbers other than BCD code, the error flag (007355) is set ON and no addition will be done.

(Example) $S_1 = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ \hline & 5 & - & - & E & (H) \end{bmatrix}$ 1110 is prohibited in the BCD mode.

Reference

In case 3 BCD digits or more have to be added, the F-10 instruction must be provided successively.

When the F-10 instruction is programmed repeatedly, the contents of the carry flag (007356) are also added after the second instruction. For the F-10 instruction that appears first in succession to the STR instruction, the contents of the carry flag (007356) are not added.



- The above example shows the case of 1523+6781=8304.

- If programmed from lower digit, the carry information will be carried on to upper digit. => See the "Double length operation."

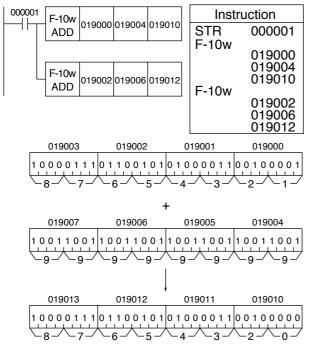
Add register and register (BCD 4 digits) F-10w (ADD)

ADD

		F-10w				[Evaluation]					
Sym	npol	ADD S1 S2	D			[Explanation]	Instr	uction 004000			
Func	tion	The contents of th are added with the registers S2, S2+1 its result is stored D+1.	conter in BCD	nts of tl 9 4 digi	he ts and	$\begin{array}{c c} & F-10w \\ \hline & & 019000 \\ \hline & & ADD \end{array} \xrightarrow{019000} 019002 019004 \end{array} \xrightarrow{019000} F-10w \\ \hline & & 019000 \\ \hline & & 019002 \\ \hline & & $					
Opera	ation	(S1, S1+1) + (S2, S2	:+1) -> I	D, D+1		registers 019000 and 019001 ar	OFF to ON, the 4 digits BCD contents of the registers 019000 and 019001 are added with the				
s	1	Use range B *				contents of the registers 019002 its result is stored in the register					
S	2	Use range B *				019005.					
D)	Use range B *				019001 0190	$\begin{array}{c} 019001 & 019000 \\ \hline 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0$				
Cond	lition	Rising edge of inpu	ıt signa	I (OFF	to ON)	6	9				
	S1,S1-	1 Unchanged									
	S2,S1-	-2 Unchanged				0 0 0 1 0 0 1 0 0 0 1 1 0	0100				
	D	D Lower 2 digits of the result Unchanged when the contents of registers			en the sters		<u>-4</u> -/				
Contents after	Contents Upper 2 digits S1, S1+1, S2 and S2+1		019005 0190								
operation	Flag	Result Zero 007357 0 1 1 to 9999 0 10000 1 10001 and above 0 Not BCD code 0	Carry 007356 0 0 1 1 0	Error 007355 0 0 0 0 0 1	Non-carry 007354 1 1 0 0 0		0 0 1 1 -3-				

* Be sure to use even addresses for registers S1, S2 and D. (Odd address such as 019003 etc. are prohibited to use.)

As double-length operation is possible for the F-10w instruction, same as the F-10 instruction, Reference write F-10w instruction in succession to add 8 digits or more BCD.



F-10d ADD		Adds ro (ADD)	egist	ter a	nd r	egist	ter (BCD 8 digits)				
Symbol		-F-10d ADD S1	S2	D			[Explanation]	Insti STR F-10d	ruction 004000		
Functior	The 8-digit BCD contents of the registers S1 to S1+3 are added by the contents of the registers S2 to S2+3 and its result is stored in the registers D to D+3.						$ \begin{array}{c c} 004000 \\ \hline F-10d \\ ADD \\ 019000 \\ 019004 \\ 019010 \\ \hline 019000 \\ 019004 \\ 019010 \\ \hline 019010 \\ \hline 019010 \\ \hline 019000 \\ 019004 \\ 019004 \\ 019010 \\ \hline 019004 \\ 019010 \\ \hline 019004 \\ 019010 \\ \hline 019004 \\ 019010 \\ \hline 019004 \\ 019010 \\ \hline 019004 \\ 019004 \\ 019010 \\ \hline 019004 \\ 019010 \\ \hline 019004 \\ 019010 \\ \hline 019010 \\$				
Operatio	n	(S1 to S1+3)	+ (S2 t	o S2+3)) -> D	to D+3	registers 019000 to 019003 are contents of the registers 01900		•		
S1		Use range C	; *				result is stored in the registers	019010 t	o 019013.		
S2		Use range C	; *								
D		Use range C	*					9001	019000		
Conditio	n	Rising edge	of inpu	t signal	(OFF	to ON)	$\begin{array}{c} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1$				
	S1 t S1+ S2 t S2+	to Unchanged					+ 019007 019006 019 000100100101010100010101 000110200000000	9005 0 1 1 0 0 1	019004 1 1 1 0 0 0 7 - 8 -		
Contents after			CD)	to S1+3 not BCI	s of regi , S₂ to S D code.	2+3 are		2011	010010		
operation	Flag	10000000 10000001	Zero 007357 1 0 1 0	Carry 007356 0 0 1	Error 007355 0 0 0 0 0	Non-carry 007354 1 1 0 0	$ \begin{array}{cccccccccccccccccccccccccccccccccccc$		019010 0 0 0 0 0 0 1 0 1 -		
		and above Not BCD code	0	0	1	0					

* Be sure to use even addresses for registers S1, S2 and D. (Odd address such as 019003 etc. are prohibited to use.)

- If the contents of registers S1 to S1+3 or S2 to S2+3 are not BCD codes, no addition occurs, with the error flag

Reference

As double-length operation is possible with the F-10d instruction, same as the F-10w instruction, write F-10d instruction in succession to subtract 16 digits or more of BCD.

Add register (BCD 2 digits) and constant (2 digits) (ADD)

Symbol	l	Fc10 S1	n	D			[Explanation] Instruction STR 004001 Fc10					
Functior	'	The content added with "n" and its ro register D.	the 2-d	igit BC	D cons		004001 Fc10 009000 85 009002 009002 When the input condition 004001 changes from					
Operatio	n	S₁+n→D					OFF to ON, the contents of the register 009000 are added with the BCD constant 85 and its result is					
S1		Use range A	A				stored in the register 009002. If operates in the same timing as the F-10 instruction.					
n		Use range (00 to 99)								
D		Use range A	4									
Conditio	n F	Rising edge	of inpu	t signal	(OFF	to ON)	$\begin{array}{c} & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ & & & \\ \end{array}$					
	S1	Unchange	ed				BCD constant 1 0 0 0 0 1 0 1 85					
Contents	D	Lower 2 d - Unchange register S (Does not	d when are no	the cont t a BCD	tents of	the	<u>_8-/_5-/</u>					
after		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354						
operation	operation 0 1 0 0 1											
	Flag	1 to 99	0	0	0	1						
	i iag	100	1	1	0	0						
		101 and above	0	1	0	0						
		S1 is not BCD code	0	0	1	0						

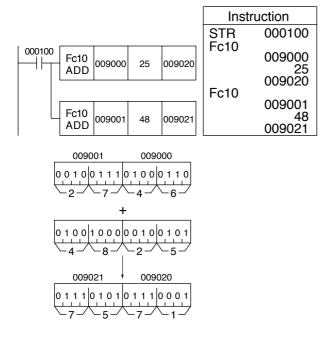
- If the contents of S1 is a number other than BCD code, the error flag (007355) is set ON and no addition will be done.
- [Example]

1010 is the code prohibited to use in the BCD code.

Reference

Fc10 ADD

Similar as the F-10 instruction, it is possible to add 3 digits or more of BCD value.



10100111

~7-

-A (H)-∕

S1

Fc10w Add register (BCD 4 digits) and constant (4 digits) ADD (ADD) [Explanation] Fc10w Symbol S1 D n ADD The BCD 4-digit contents of the 002001 Fc10w - 1 registers S1, S1+1 are added with the ADD Function 4-digit BCD constant "n" and its result is stored in the registers D, D+1.

Rising edge of input signal (OFF to ON)

code.

Carry

007356

0

0

1

0

0

Unchanged when the

contents of registers S1

and S1+1 are not BCD

(Does not calculate.)

Error

007355

0

0

0

0

Non-carry

007354

1

1

0

0

0

(S1, S1+1) +n -> D, D+1

Use range 0000 to 9999

Use range B *

Use range B *

Unchanged

Lower 2 digits

Upper 2 digits

of the result

Result

1 to 9999

10000 and above

Not BCD code

10000

of the result

When the input condition 002001 changes from OFF to ON, the 4-digit BCD contents of the registers 019500 and 019501 are added with the BCD constant 2345 and its result is stored in the registers 019600 and 019601.

2345 019600

019500

Instruction

002001

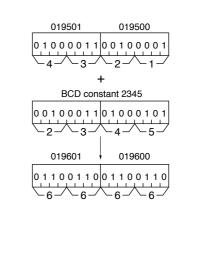
019500

019600

2345

STR

Fc10w



Be sure to use even addresses for registers S1 and D. (Odd address such as 019003 etc. are prohibited to use.)

Zero

007357

1

0

1

0

0

Reference

Operation

S₁

n

D

Condition

Contents

operation

after

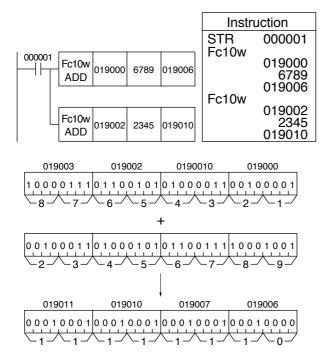
S1,S1+1

D

D+1

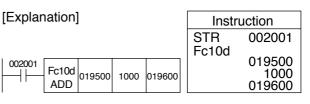
Flag

As double-length operation is possible with the Fc10w instruction, same as the Fc10 instruction, write Fc10w instruction in succession to add 8 digits or more of BCD.

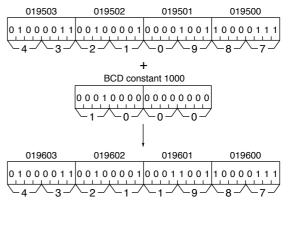


BCD register (BCD 8 digits) and constant (4 digits) Fc10d (ADD)

Symbol	-	Fc10d ADD S1	n		[Explanation]		
Functior	n r 4	The BCD 4-c egisters S1 -digit BCD c s stored in th	to S1+3 constai	esult	When the input co		
Operatio	n (S1 to S1+3)	+n -> [OFF to ON, this ins contents of register		
S1	ι	Jse range C		constant 1000, and 019600 to 019603.			
n	ι	Jse range 0	000 to	019503 01			
D	ι	Jse range C	*		0 1 0 0 0 0 1 1 0 0 1 0		
Condition	n F	Rising edge	of inpu	4 3 2			
	S to S+3	Unchange	ed	000			
Contents	D to D+3	Result (8- Unchanged S1 to S1+3 a (Does not ca	when th are not B	019603 01			
after operation	Flag	Result 0 1 to 99999999 10000000	Zero 007357 1 0 1	Carry 007356 0 0 1	Error 007355 0 0 0	Non-carry 007354 1 1 0	
		100000001 and above Not BCD code	0	1 0	0	0	



ndition of 002001 changes from struction adds the 8-digit BCD ers 019500 to 019503 to BCD d stores the result in registers

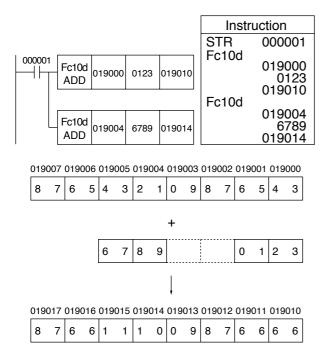


* Be sure to use even addresses for registers S1 and D. (Odd address such as 019003 etc. are prohibited to use.)

Reference

ADD

As double-length operation is possible with the Fc10d instruction, same as the Fc10 instruction, write Fc10d instruction in succession to subtract 16 digits or more of BCD.



F-11 SUB

Subtracts register from register (BCD 2 digits) (SUBtract)

Symbol		F-11 SUB S1	S2	D			[Explanation] Instruction STR 004001 F-11				
Function	n s	The contents subtracted fr register S2 ir result is store	om the n BCD	e conte 2 digits	nts of ti and its	he	004001 F-11 009030 009040 009050 SUB 009030 009040 009050 009040 When the input condition 004001 changes from				
Operation	n (S1-S2 -> D					OFF to ON, the contents of the register 009030 are subtracted by the contents of the register 009040				
S1	l	Jse range A					and its result is stored in the register 009050. The contents of registers 009030 and 009040 remain				
S2	l	Jse range A					unchanged.				
D	Use range A						- Transition of operational result and flags				
Condition	n F	Rising edge	of inpu	t signa	I (OFF	to ON)	(004001) Register (009030) 15 15 \{ 43 43 43 43 \{ E5 E5 \} 24 24				
	S1	Unchange	ed				$(009040) \xrightarrow{15 15} (11 11) (58 58) (13 13) (9C 9C$				
	S2	Unchange	ed				(009050) 40 00 00 32 32 85 85 85 85 85				
Contents	D	Unchanged registers S (Does not d	1 and S	2 are no	tents of ot BCD o	code.	(15-15) (43-11) (43-58) No subtraction No subtraction Non-carry flag (009030 not BCD). (009040 not BCD). 007354 Error flag (007355)				
after operation		Result	Zero 007357	Carry 007356		Non-carry 007354	Carry flag				
		0	1	0	0	1	Zero flag				
	Flac	1 to 99	0	0	0	1	007357 Valid until a flag affecting instruction				
		Negative value	0	1	0	0	is met in the program.				
		S1 and S2 are not BCD code	0	0	1	0					

- If (contents of S_1) < (contents of S_2) is calculated, the answer will be produced in the complement of 100. [Example]

23-85=-62 will produce the answer of 38 which is the complement of 100 of 62.

(Assume it to be 123-85=38.)

- If the contents of S1 and S2 are numbers other than BCD code, the error flag (007355) is set ON and no addition will be done. (D remains unchanged.)

[Example] S1 11000101C (H) -5 1100 is prohibited in the BCD code.

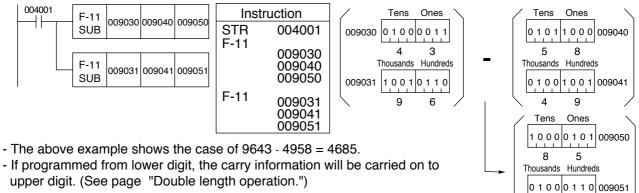
Reference

In case 3 BCD digits or more have to be subtracted, the F-11 instruction must be provided successively.

When the F-11 instruction is programmed repeatedly, the contents of the carry flag (007356) is also subtracted after the second instruction. For the F-11 instruction that appears first in succession to the STR instruction, the contents of the carry flag (007356) is not subtracted.

4

6



F-11wSubtract register from register (BCD 4 digits)SUB(SUBtract)

Symbol		F-11w SUB S1	S2	D			[Explanation] Instruction STR 004000 F-11w
Functior	FunctionThe 4-digit BCD contents of the registers S1 and S1+1 are subtracted by the contents of the registers S2, S2+1 and its result is stored in the registers D, D+1.					2,	$ \begin{array}{c} 004000\\ \hline F-11w\\ SUB\\ \hline 019000\\ \hline 019002\\ \hline 019002\\ \hline 019004\\ \hline 019002\\ \hline 019004\\ \hline 019002\\ \hline 019004\\ \hline 019002\\ \hline 019002\\ \hline 019004\\ \hline 019002\\ \hline 019004\\ \hline 019004\\ \hline 019002\\ \hline 019004\\ \hline 0190$
Operation	ation (S1, S1+1) - (S2, S2+1) -> D, D+1						registers 019000 and 019001 are subtracted by the contents of the registers 019002 and 019003 and
S1		Use range B	; *				its result is stored in the registers 019004 and 019005.
S2		Use range B	*				019001 019000
D		Use range B	} *				
Condition	n	Rising edge	of inpu	t signa	I (OFF	to ON)	-
	S1, S1+	1 Unchange	ed				019003 019002
	S2, S2+		ed				
	D						019005 019004
Contents after	D+1	Lippor 2 digite S1 S1+1 S2 and S2+1					
operation		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	
		0	1	0	0	1	
	Flag	1 10 0000	0	0	0	1	
		Negative value	-	1	0	0	
		Not BCD code	0	0	1	0	

* Be sure to use even addresses for registers S₁, S₂, and D. (Odd address such as 019003 etc. are prohibited to use.)

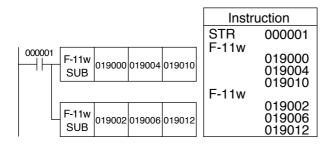
- Result will be produced as a complement of 10000, if (contents of S₁, S₁₊₁) < (contents of S₂, S₂₊₁) is carried out.

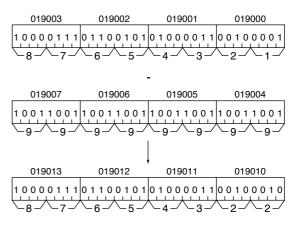
[Example]

2578-7890=-5312 will produce the answer of 4688 which is the complement of 10000 of 5312. (Assume it to be 12578-7890=4688.)

Reference

As double-length operation is possible with the F-11w instruction, same as the F-11 instruction, write F-11w instruction in succession to subtract 16 digits or more of BCD.





F-11d SUB

Subtract register from register (BCD 8 digits) (SUBtract)

Symbol	I	-F-11d SUB S1 S2	D			[Explanation] Instruction STR 004000
Functior	Function The 8-digit BCD contents of the registers S1 to S1+3 are subtracted by the contents of the registers S2 to S2+3 and its result is stored in the registers D to D+3.					$ \begin{array}{c c} & & F-11d \\ \hline & & F-11d \\ \hline & & SUB \end{array} \hline 019000 019004 019010 \end{array} \begin{array}{c} F-11d \\ & & 019000 \\ \hline & & 019004 \\ \hline & & 019010 \end{array} \end{array} $ When the input condition 004000 changes from OFF to ON, the 8-digit BCD contents of the
Operatio	n	(S1 to S1+3) - (S2 to	o S2+3)	-> D to	D+3	registers 019000 to 019003 are subtracted by the contents of the registers 019004 to 019007 and its
S1		Use range C *				result is stored in the registers 019010 to 019013.
S2		Use range C *				
D		Use range C *				019003 019002 019001 019000
Conditio	n	Rising edge of input	signal (C	OFF to C	ON)	6 - 7 - 8 - 9 - 8 - 7 - 6 -
	$\frac{S_1 \text{ to}}{S_1+3}$ $\frac{S_2 \text{ to}}{S_2+3}$					- 019007 019006 019005 019004 $0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0 0 1 0 0 0 1 1 0 0 0 1 1 0 0 0 0 1 1 - 2 - 3 - 4 - 4 - 3 - 2 - 1 -$
	Contents D to Resul D+3 (8-dig		contents	ged whe s of regis , S2 to S) code.	sters S1	
operation	Flag	Result Zero 007357 0 1 1 to 99999999 0 Negative value 0 Not BCD code 0	Carry 007356 0 0 1 0	Error 007355 0 0 0 0 1	Non-carry 007354 1 1 0 0	$ \begin{array}{c} 019013 \\ \hline 0101010101010101010101010101010101010$

* Be sure to use even addresses for registers S1, S2, and D.

(Odd address such as 019003 etc. are prohibited to use.)

- Result will be produced as a complement of 100000000, if (contents of S₁ to S₁₊₃) < (contents of S₂ to S₂₊₃) is carried out.
 - [Example]

25780000-78900000= -53120000 will produce the answer of 46880000 which is the complement of 100000000 of 53120000. (Assume it to be 125780000-78900000=46880000.)

- If the contents of S1 to S1+3 and S2 to S2+3 are numbers other than BCD code, the error flag (007355) is set ON and no addition will be done. (D to D+3 remain unchanged.)

Reference

As double-length operation is possible with the F-11d instruction, same as the F-11 instruction, write F-11d instruction in succession to subtract 16 digits or more of BCD.

				Inote	tian
				Instr	uction
				STR	000001
	019000	019010	019020	F-11d	019000 019010 019020
				F-11d	
F-11d SUB	019004	019014	019024		019004 019014 019024

· ·															
019007 019006 019005 019004 019003 019002 019001 019000															
8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3
-															
019017 019016 019015 019014 019013 019012 019011 019010															
9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
019	027	019	026	019	025	019	024	019	023	019	022	019	021	019	020
8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	4
L															

Subtracts constant (2 digits) from register (BCD 2 digits) Fc11 SUB (SUBtract)

Symbol		Fc11 SUB S1	n	D			[Explanation]	Insti STR Fc11	ruction 004001			
Functior	ו	2-digit BCD from the cor and its resul D.	ntents o	of the r	egister	S1	$\begin{array}{c c} 004001 \\ \hline \\ \hline \\ SUB \\ \hline \\ SUB \\ \hline \\ \\ 009000 \\ \hline \\ 85 \\ \hline \\ 009002 \\ \hline \\ \hline \\ 009002 \\ \hline \\ \hline \\ 009002 \\ \hline \\ \hline \\ 009000 \\ \hline \\ 85 \\ \hline \\ 009002 \\ \hline \\ \hline \\ 009000 \\ \hline \\ \hline \\ \hline \\ 009000 \\ \hline \\ \hline \\ \hline \\ 009000 \\ \hline $					
Operatio	n	S1-n -> D					the contents of the register 009					
S1		Use range A	١				stored in the register 009002. If operates in the same timing a	s the F-1	1 instruction.			
n		Use range 0	0 to 99)								
D		Use range A	١									
Conditio	n F	Rising edge	of inpu	t signa	I (OFF	to ON)			nsferring			
	S1	Unchange	ed				BCD constant 1 0 0 0 1 0 1		1 0 0 1 -5			
Contents	D	Unchanged register S1 (Does not d	are not	a BCD		the						
after operation		Result	Zero 007357	Carry 007356		Non-carry 007354						
	Flag	0	1	0	0							
	Flag	1 to 99 Negative value	0	0	0	1						
		S1 is not BCD code	0	0	1	0						

- If (contents of S_1) < n is calculated, the answer will be produced in the complement of 100. [Example]

23-85= -62 will produce the answer of 38 which is the complement of 100 of 62. (Assume it to be 123-85=38.)

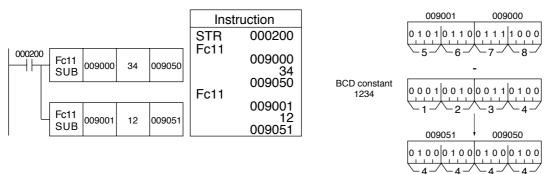
- If the contents of S1 is a number other than BCD code, the error flag (007355) is set ON and no subtraction will be done. (D remains unchanged.) [Example]

10011010 9 - A (H)-

1010 is the code prohibited to use in the BCD code.

Reference

Similar as the F-11 instruction, it is possible to subtract 3 digits or more of BCD value.



Fc11wSubtract constant (4 digits) from register (BCD 4 digits)SUB(SUBtract)

Symbo	I	-Fc11w SUB S1	n	D			[Explanation]	Instru STR Fc11w	uction 002001
Function	n	The 4-digit BC subtracted fro contents of th its result is sto D+1.	om the le regi	4-digitsters S	t BCD 61, S1+	1 and	When the input condition 00200 OFF to ON, the BCD constant 2)1 change	
Operatio	on	(S1, S1+1) -n -	-> D, [D+1			from the 4-digit BCD contents of 019500 and 019501 and its res	-	
S1		Use range B *	*				registers 019600 and 019601.		
n		Use range 00	00 to 9	9999			019501 019	500	
D		Use range B *	*				010000110010	0001	
Conditio	n	Rising edge o	of inpu	t signa	I (OFF	to ON)	-		
	S1, S1+	1 Unchanged	d				BCD constant 234		
	D Lower 2 digits of the result Unchanged when the contents of registers St								
Contents	Contents D+1 Upper 2 digits of the result and S1+1 are not BCE code.				ot BCD	019601 019	600		
operation		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		0110	
	Flag	0 1 to 9999	1 0	0	0	1		<u>-</u> 6 <u>-</u>	
		Negative value	0	1	0	0			
		Not BCD code	0	Ó	1	Ő			

 * Be sure to use even addresses for registers S1 and D.

(Odd address such as 019003 etc. are prohibited to use.)

- If (contents of S1, S1+1) < n is calculated, the answer will be produced in the complement of 10000. [Example]

4568-7890=-3322 will produce the answer of 6678 which is the complement of 10000 of 3322. (Assume it to be 14568-7890=6678.)

Reference

As double-length operation is possible for the Fc11w instruction, same as the Fc11 instruction, write Fc11w instruction in succession to subtract 9 digits or more of BCD.

				Instru	iction
				STR	000001
Fc11w SUB	019000	8888	019006	Fc11w Fc11w	019000 8888 019006
Fc11w SUB	019002	7777	019010		019002 7777 019010

019003	019002	019001	019000
10000111	01100101	01000011	00100001
01110111	01110111	10001000	10001000
019011	019010	019007	019006
0 0 0 0 1 0 0 1	10000111	01010100	00110011

Fc11d
SUBSubtract constant (4 digits) from register (BCD 8 digits)SUB(SUBtract)

Symbo	I	Fc11d SUB S1	n	D			[Explanation] Instruction S T R 002001
Functio	n	The 4-digit E subtracted fr contents of t and its result D to D+3.	om the he regi	e 8-digit sters S	BCD 1 to S1		$ \begin{array}{c} \begin{array}{c} 002001 \\ \hline \\ \hline \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ $
Operatio	on	(S1 to S1+3)	—n→[D to D+	3		from the 8-digit BCD contents of the registers 019500 to 019503 and its result is stored in the
S1		Use range C	*				registers 019600 to 019603.
n		Use range 0000 to 9999					019503 019502 019501 019500 0 1 0 0 0 0 1 1 0 0 1 0 0 0 1 0 0 0 0
D		Use range C	*				$\begin{array}{c} 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ \hline & 0 & 1 & 0 & 0 & 1 & 1 \\ \hline & 4 & - & 3 & - & 2 \\ \hline & - & 1 & - & 0 \\ \hline \end{array} $
Conditic	n	Rising edge	of inpu	ıt signa	I (OFF	to ON)	
	S1 to S1+3	Unchange	ed				
Contents	D to D+3	Operational		contents to S1+3 code.	ged whe s of regis are not l ot calcul	sters S1 BCD	019603 019602 019601 019600 0 1 0 0 0 0 1 1 0 0 1 0 0 0 0 1 1 0 0 1 1 0 0 0 1 1 1
after		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$
operation		0	1	0	0	1	
	Flag	1 to 99999999	0	0	0	1	
		Negative value	0	1	0	0	
		Not BCD code	0	0	1	0	

* Be sure to use even addresses for registers S₁ and D.

(Odd address such as 019003 etc. are prohibited to use.)

- If (contents of S₁ to S₁₊₃) < n is calculated, the answer will be produced in the complement of 100000000. [Example]

4568–7890=–3322 will produce the answer of 6678 which is the complement of 100000000 of 3322. (Assume it to be 100004568–7890=99996678.)

Reference

As double-length operation is possible with the Fc11d instruction, same as the Fc11 instruction, write Fc11d instruction in succession to subtract 16 digits or more of BCD.

	Instru	uction
	STR	000001
000001 Fc11d SUB	Fc11d	019000 1000 019010
	Fc11d	
Fc11d SUB 019004 1000 019014		019004 1000 019014

0190	019007 019006 019005 019004 019003 019002 019001 019000														
8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3
	—														
				1	0	0	0					1	0	0	0

,	0190)17	0190	016	019	015	019	014	019	013	019	012	019	011	019	010
	8	7	6	5	3	3	2	1	0	9	8	7	5	5	4	3

F-12 CMP

Compare register with register (1 byte) (CoMPare)

Symbol		- F-12 CMP S1 S2	[Explanation] Instruction S T R 004003 F -12					
Function	ı	The contents of the register S1 are compared with the contents of the register S2.	$\begin{bmatrix} 0.04003 \\ -1 \end{bmatrix} \begin{bmatrix} F-12 \\ CMP \end{bmatrix} \begin{bmatrix} 0.09000 \\ 0.09010 \end{bmatrix} \begin{bmatrix} F-12 \\ 0.09000 \\ 0.09010 \end{bmatrix}$ When the input condition 04003 is ON, the contents of					
Operation	n	S1< = >S2→Flag	the register 009000 are compared with the contents of the register 009010 and its results are set in the non-					
S1		Use range A	carry flag (007354), carry flag (007356), and the zero flag (007357). registers 009000 and 009010 remain					
S2		Use range A	unchanged after this operation.					
Condition	n	When the input signal is ON (not limited to OFF to ON change)	- Transition of register contents and flags					
	S1	Unchanged						
Contents	S2	Unchanged	009000 23 23 23 23 23 23 23 23 23 23 23 23 23					
after operation	Flaç	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	(009010 11 11 \) 23 23 \\ 58 58					
	d ir	7355) will be already "0." Istructions: F-12w, F-12d, Fc12, Fc12w, Fc12d, F-112, F-112w, F-112d	Zero flag 007357					

In case comparison is to be done only at a OFF to ON transition of the input condition, use the differentiate instruction in conjunction with the input condition.

004003 F-44	F-12	00000	009010	[Instru	uction
	CMP	009000	009010		STR	004003
					F -44	
					F -12	009000
						009010

Reference

To compare data of 2 bytes or more, it must be so programmed that comparison should start from a lower digit, as in addition (F-10) and subtraction (F-11). If the F-12 is programmed in succession, the contents of the carry flag (007356) are also compared after the second F-12 instruction. For the F-12 instruction that first follows the STR instruction, the contents of the carry flag (007356) are exempted from comparison.

				Instr	uction
004003	F 10			STR	004003
	F-12 CMP	009000	009010	F-12	
	CIVIF				009000
				F-12	009010
	F-12 CMP	009001	000011	1 12	009001
	CMP	000001	000011		009011

When programmed from a lower digit, the carry-down information is forwarded to upper digit. => See page "Double length operation."

F-12w Compare register with register (1 word) (CoMPare)

Symbol		-F-12w CMP S1	S2				[Explanation] Instruction S T R 004004					
Functior	۱	The word con S1+1 are cor contents of th	npared	with th	e word		004004 F-12w 009000 009002 F -12w 009000 When the input condition 004004 is ON, the word F - 12w 009000 009002 009002					
Operatio	n	S1, S1+1<=	>S2, S2	2+1→F	lag		contents of the registers 009000 and 009001 are compared with the word contents of the registers 009002					
S1		Use range B	*				and 09003 and its results are set in the non-carry flag (007354), carry flag (007356), and the zero flag (007357).					
S2		Use range B	*				registers 009000 to 009003 remain unchanged.					
Conditio	n	When the in (not limited t					Input (004004) Register					
	S1, S1+	Unchange	əd				(009000) 005 00					
Contents	S2, S2+	1 Unchange	əd				009002) 006 006 005 005 004 004 (octal) (009003) 123 123 124 124 125 125 (octal)					
after operation		Contents of register	Zero 007357	Carry 007356		Non-carry 007354	Non-carry flag					
operation	Flag	S1, S1+1=S2, S2+1	0	0	0	1 1 0	Carry flag (007356)					
		S1, S1+1 <s2, s2+1<br="">even address tructions: F-1</s2,>		•		Zero flag(007357)						

Fc12d, F-112, F-112w, F-112d

F-12d Compare register with register (2 words) (CoMPare)

Symt	ool	F-12d CMP S1	S2				[Explanation] Instruction S T R 004004						
Funct	ion	The word co to S1+3 are contents of t	compa	ed wit	h the w	vord	004004 F-12d 009000 009000 CMP 009000 009004 009000 When the input condition 004004 is ON, the 2 words 009004						
Opera	tion	S1 to S1+3<	= >S2 t	o S2+3	S→Flag	I	contents of the registers 009000 to 009003 are compared with the 2-word contents of the registers 009004 to						
S1		Use range C	; *				009007 and its results are set in the non-carry flag (007354), carry flag (007356), and the zero flag (007357).						
S2		Use range C	, * ,				Registers 009000 to 009007 remain unchanged after this operation.						
Condi	tion	When the in (not limited t				e)	Input (004004) Register						
	S1 to S1+3	Unchanged					(009000) 124005 124005 124005 124005						
Contents	S2 to S2+3	Unchanged					(009003) (009004) (123006 123006 124005 124005 125004 125004						
after operation		Contents of register S1 to S1+3>S2 to S2+3	Zero 007357 0	Carry 007356 0	Error 007355 0	Non-carry 007354 1	(009007) (009007) (00007) (0007) (0007) (0007) (00007) (00007) (00007) (
	-	S1 to S1+3=S2 to S2+3	1	0	0	1							
	to us	S1 to S1+3 <s2 s2+3<br="" to="">e even addres structions: F-1</s2>		w, Fc1	2, Fc12	(007356) Zero flag (007357)							

Fc12d, F-112, F-112w, F112d

Reference

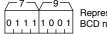
If the F-12d instruction is used in succession, 8 bytes or more data comparison may be done.

		Instru	uction
000000	F-12d 000000 000010	STR	000000
	CMP 009000 009010	F -12d	009000
		F -12d	009010
	F-12d CMP 009004 009014	20	009004 009014

Fc12
CMPCompare register with octal constant (1 byte)(CoMPare)

Symbo	bl	- Fc12 CMP S1	n				[Explanation]	Instruc T R	tion 04001			
Functio	'n	The content compared w		-			004001 Fc12 CMP	00 075		c12	09000 075	
Operatio	on	$S_1 <=>n-$	∙Flag				When the input c	ondition	004001 i	s ON, th	e	
S1		Use range A	4				contents of the register 009000 are compared with the octal constant 075 and its results are set in the					
n		Use range	000 to	377(8)			non-carry flag (007354) , carry flag (007356) , and zero flag (007357). The contents of the register					
Conditio	on	When the in (not limited t				e)	009000 remain u	nchange		•	-	
Ocurtorate	S1	Unchange	d					<u> </u>	Co	ompare		
Contents after operation	Flag	Contents of register * S1>n	Zero 007357 0	Carry 007356 0		Non-carry 007354 1	Octal constant 075	1101				
		S₁=n S₁ <n< td=""><td>1</td><td>0</td><td>0</td><td>1</td><td></td><td>Zero</td><td>Carry</td><td>Error</td><td>Non-carry</td></n<>	1	0	0	1		Zero	Carry	Error	Non-carry	
* Ennor flor	~ (00		_		0	0		007357	007356	007355	007354	
* Error flag (007355) will be already "0." Resembled instructions: F-12, F-12w, F-12d, Fc12w,							0	0	0	1		
						F-112	d					

Reference Use an octal number in writing program with the Fc12 instruction. Octal number can express any bit pattern and it dose not require annoying weight calculation. To compare with a BCD constant, convert it to an octal equivalent before writing it in the program.



Fc12w CMP Represents 79 in the BCD notation.

Represents 171 in the octal notation.

Compare register with octal constant (1 word) (CoMPare)

Symbo	I	-Fc12w CMP S1 n	[Explanation] Instruction S T R 002000				
Function	n	The word contents of the registers S1 and S1+1 are compared with an octal constant "n".	F c12w CMP 009000 012345 F c12w 009000 012345				
Operatio	n	S ₁ , S ₁ +1<=>n→Flag	When the input condition 002000 is ON, the word contents of the registers 009000 and 009001 are				
S1		Use range B *	compared with the octal constant 012345 and its results are set in the non-carry flag, carry flag, and				
n		Use range 000000 to 177777(8)	zero flag. The contents of the registers 009000, 009001 remain unchanged. Operation takes place				
Conditio	'n	When the input signal in the ON (not limited to OFF to ON change)	in the same timing as F-12w.				
	S1, S1+	1 Unchanged	0 0 0 0 1 0 0 1 0 1 1 1 0 1 1 1 1 0 0 0 0 0 0 1 0 0 1 0 1 1 1 0 1 1 1 0				
Contents after operation	Flag	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{c} \begin{array}{c} \begin{array}{c} \begin{array}{c} \\ \\ \end{array} \\ \hline \\ \end{array} \\ \hline \\ \end{array} \\ \hline \\ \\ \\ \end{array} \\ \hline \\ \\ \\ \end{array} \\ \hline \\ \\ \\ \\$				
(Odd addr	ess s	even addresses for register S ₁ . such as ⊐00011 etc. are prohibited to use.) ructions: F-12, F-12w, F-12d, Fc12,	Zero Carry Error Non-carry 007357 007356 007355 007354 0 1 0 0				

10-38

Fc12d, F-112, F-112w

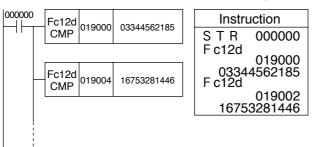
Fc12d CMP

Compare register with octal constant (2 words) (CoMPare)

Symbol	-Fc12d S1 n	[Explanation] Instruction S T R 002000 F c12d		
Function	The 2-words contents of the registers S1 to S1+3 are compared with an octal constant "n."	002000 Ec12d		
Operation	S1 to S1+3<=>n→Flag	When the input condition 002000 is ON, the word contents of the registers 009000 to 009003 are		
S1	Use range C *	compared with the octal constant 02471355171 and its results are set in the non-carry flag		
S2	Use range 000000000000000 to 37777777777(8)	(007354), carry flag (007356), and zero flag (007357). The contents of the registers 009000 to		
Condition	When the input signal in the ON (not limited to OFF to ON change)	009003 remain unchanged. Operation takes place in the same timing as F-12d.		
s	to +3 Unchanged	009003 009002 009001 009000		
Contents after operation	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	$\begin{array}{c} & & \\$		
(Odd addres	se even addresses for register S₁. s such as ⊐00011 etc. are prohibited to use.) structions: F-12, F-12w, F-12d, Fc12, Fc12w, F-112, F-112w	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$		

Reference

If the Fc12d instruction is used in succession, 8 bytes or more data comparison may be done.



Fx12
CMPCompare register with hexadecimal constant (1 byte)
(CoMPare)

Symbo	Ы	- Fx12 CMP S1	n				[Explanation]				tion 004001
Functio	'n	The contents of the register S1 are compared with a hexadecimal constant "n."				004001 Fx12 009000 3D 009000 CMP 009000 3D 3D					
Operatio	on	$S_1 <=>n-$	Flag				When the input c contents of the re				
S1		Use range A	4				the hexadecimal constant 3D and its results are set in the non-carry flag, carry flag, and zero flag. The contents of the registers 009000 remain unchanged after this operation.				
n		Use range	00 to F	F							
Conditio	on	When the in (not limited t				inge)					
	S1	Unchange	d				009000 0 1 0 1 0 0 1 1 5				
Contents after		Contents of register	Zero 007357	Carry 007356		Non-carry 007354		3-7	Co	ompare	
operation	Flag	* <u>S1>n</u> S1=n	0	0	0	1	constant (3D)	01			
	S1 <n 0="" 0<="" 1="" td=""><td></td><td>D—⁄</td><td></td><td>+</td><td></td></n>					D—⁄		+			
* Error flag (007355) will be always "0."					Zero	Carry	Error	Non-carry			
					2d, Fc1	2w,		007357	007356	007355	007354
Resembled instructions: F-12, F-12w, F-12d, Fc12w, Fc12d, Fx12, Fx12w, Fx12d,						0	0	0	1		

F-112, F-112w, F-112d

Fx12w Compare register with hexadecimal constant (1 word) (CoMPare)

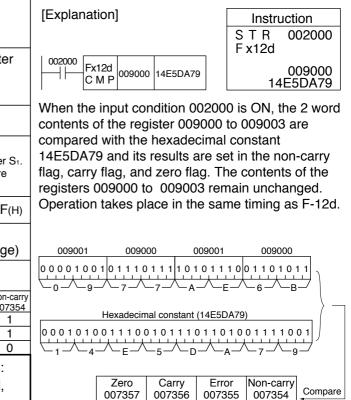
Symbol	Fx12w S1 n	[Explanation] Instruction S T R 002000 F x12w			
Function	The word contents of the register S1 and S1+1 are compared with a hexadecimal constant "n."	002000 Fx12w 009000 14E5 009000 C M P 009000 14E5 14E5			
Operation	S ₁ , S ₁ +1<=>n→Flag	When the input condition 002000 is ON, the word contents of the register 009000 and 009001 are			
S1	Use range B - Be sure to use even addresses for register S ₁ . (Odd address such as ¬00011 and etc. are prohibited to use.)	compared with the hexadecimal constant 14E5 and its results are set in the non-carry flag, carry flag, and zero flag. The contents of the registers 009000, 009001 remain unchanged. Operation takes place			
n	Use range 0000 to FFFF(H)	in the same timing as F-12w.			
Condition	When the input signal is ON (not limited to an OFF to ON change)	009001 009000			
	, +1 Unchanged	0 0 0 0 1 0 0 1 0 1 1 1 0 1 1 1 0			
Contents after operation FI	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	Hexadecimal constant (14E5) 0 0 0 1 0 1 0 0 1 1 1 0 0 1 0 1 1 4 E Zero Carry Error Non-carry			
Resembled in	structions: F-12, F-12w, F-12d, Fc12, Fc12w, Fc12d, Fx12, Fx12d, F-112, F-112w	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			

Fx12d CMP

Compare register with hexadecimal constant (2 words) (CoMPare)

Symbo)	Fx12d C M P S1 n					
Functio	n	The 2-words contents of the register S1 to S1+3 are compared with a hexadecimal constant "n."					
Operatio	on	S1, S1+3<=	=>n →	Flag			
S1		Use range C - Be sure to use even addresses for register S ₁ . (Odd address such as ⊐00011 and etc. are prohibited to use.)					
n		Use range 00000000 to FFFFFFF(H)					
Conditio	on	When the input signal is ON (not limited to an OFF to ON change)					
Quarterate	S1 to S1+3		d				
Contents after		Contents of register	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	
operation	Flag	S1 to S1+3>n	0	0	0	1	
operation	J	S1 to S1+3=n	1	0	0	1	
		S1 to S1+3 <n< td=""><td>0</td><td>1</td><td>0</td><td>0</td></n<>	0	1	0	0	

The following F instructions have similar functions: F-12, F-12w, F-12d, Fc12, Fc12w, Fc12d, Fx12, Fx12w, F-112, F-112w



Reference

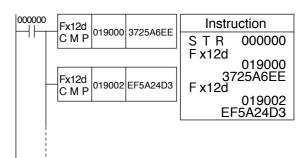
0

If the Fx12d instruction is used in succession, 8 bytes or more data comparison may be done.

1

0

0



F-13
ANDANDs register with register (1 byte)(AND)

Symbol	F-13 AND	S D	[Explanation]	Instruction S T R 004002 F -13		
Function	are ANE register	tents of the register S (8 bits) Ded with the contents of the D (8 bits) and its result is n the register D.	004002 F-13 009000 009002 009002 AND 009000 009002 009002 When the input condition 004002 changes from the input conditinges from the input conditinges			
Operation	S∩D→	D	OFF to ON, the 8-bit contents of the register 009000 are ANDed with the 8-bit contents of the register			
S	Use ran	ge A	009002 and its result is stored in the register 009002. The contents of the register 009000 remain unchanged.			
D	Use ran	ge A				
Condition	Rising e	dge of input signal (OFF to ON)	Before operation	After operation		
Contents	S	Unchanged	009000 0 1 0 1 0 0 1 1	-Ditto-009000		
after	D	Result	AND			
operation	Flag	Unchanged	009002 1 0 1 1 1 0 0 1	0 0 0 1 0 0 0 1 009002		

- AND truth table

F-13w AND

Symbol	A	В	С
	0	0	0
	1	0	0
в	0	1	0
	1	1	1

AND register with register (1 word) (AND)

Symbol	F-13w AND	S D	[Explanation]		Instruction S T R 004000		
Function	S+1 are contents	bit contents of the registers S, ANDed with the 16-bit of the registers D, D+1 and t is stored in the registers D,	When the input OFF to ON, th	009000 009002 ut condition 00400 ne 16 bits contents 009001 are ANDed	s of the registers		
Operation	S, S+1 ()D, D+1→D, D+1	contents of the registers 009002 and 009003 and its result is stored in the registers 009002 and				
S	Use ran	ge B *	009003.	009001 009000			
D	Use ran	ge B *		010110010101	1 1 0 1		
Condition	Rising e	dge of input signal (OFF to ON)	Before operation	009003 009	AND		
Contents	S, S+1	Unchanged		1 1 0 1 0 1 0 0 1 0 1 0	0 1 1 0		
after	D, D+1	Result		009003 009			
operation	Flag	Unchanged	After operation	0101000000000			
* Be sure to use even addresses for registers S and D.			The contents	of the registers 00	9000 and 009001		

(Odd address such as 019003 etc. are prohibited to use.)

F-13d AND	AN (AN	Ds register with regis ID)	ster (2 words)			
Symbol	F-1: AN		[Explanation]	Instruction S T R 004000		
Function	(32 bit of the	ontents of the registers S to S+3 s) are ANDed with the contents registers D to D+3 (32 bits) and ult is stored in the registers D to	Image: Prince of the registers Prince of the registers Image: Prince of the registers 009000 <			
Operation	S to S	+3∩D to D+3→D to D+3				
S	Use ra	inge C *	result is stored in the registers 009004 to 009007.			
D	Use ra	inge C *	Before operation 009003 009002 009	9001 009000		
Condition	Rising	edge of input signal (OFF to ON)	0 1 0 1 1 0 0 1 0 1 0 1 1 1 0 1 0 1 0 1			
Contents	S to S+3	Unchanged	AND 009007 009006 009	9005 009004		
after operation	D to D+3	Result	1 1 0 1 0 1 0 0 1 0 1 0 0 1 1 0 1 1 0 1	01001010100110		
operation	Flag	Unchanged	After operation009007009006009	9005 009004		
* Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to use.)			0 1 0 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1	000000000100		
			The contents of the registers 00 remains unchanged.	09000 to 009003		

AND register with octal constant (1 byte) Fc13 AND (AND)

Symbol	Fc13 AND	n I	D			[Explanation]	Instruction S T R 004001	
Function	the cont	constan ents of th stored in	ne regi	ster D	and its	004001 Fc13 AND 123 009002	F c13 123 009002	
Operation	n∩D→D					When the input condition 004001 changes from OFF to ON, the octal constant 123 is ANDed with		
n	Use range 000 to 377					the contents of the register 009002 and its result is stored in the register 009002.		
D	Use range A					Before operation		
Condition	Rising e	dge of in	put sig	nal (O	FF to ON)		After exercise	
Contents	D	Result				AND	After operation	
after operation	Flag	Unchar	nged			$\begin{array}{c} \text{Octal} \\ \text{constant} \\ (123) \end{array} \left(\begin{array}{c} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1$		
- AND truth table					(123)			
Symt	Symbol		В	С				
		0	0	0]			
A —	\sum	1	0	0				

Fc13w AND

в —

AND register with octal constant (1 word) (AND)

Symbol	Fc13w AND	n D	[Explanation]	Instruction S T R 004001 F c13w	
Function	the 16-l and D+	I constant "n" is ANDed with bit contents of the registers D 1 and its result is stored in the s D and D+1.	Image: When the input condition 004001 changes from			
Operation	n∩D, E)+1→D, D+1	OFF to ON, the octal constant 026562 is ANDed with the16 bits contents of the registers □00000			
n	Use rar	nge 000000 to 177777(8)	and ¬00001 and its result is stored in the registers			
D	Use rar	ige B *	Octal constant 026562			
Condition	Rising e	dge of input signal (OFF to ON)		0010110101110 02656	0 0 1 0	
Contents after	D, D+1	Result	Before operation	⊐00001 ⊐0000 1 0 0 1 1 0 0 0 1 1 0 0 1		
operation	Flag	Unchanged		=00001 =000		
		dresses for register D. 00011 etc. are prohibited to use.)	After operation			

0

1

1

1

0

1

Fc13d AND

AND register with octal constant (2 words) (AND)

	-					
Symbol	Fc1		[Explanation]	Instruction S T R 004001 F c13d		
Function	the 3 to D+	ctal constant "n" is ANDed with 2-bit contents of the registers D 3 and its result is stored in the ters D to D+3.	$\begin{array}{c c} & & & & & \\ & & & & \\ \hline & & & & \\ \hline & & & &$			
Operation	n∩D	to D+3→D to D+3	OFF to ON, the octal constant 02656237614 is ANDed with the 32 bits contents of the registers ⊐00000 to ⊐00003 and its result is stored in the registers ⊐00000 to ⊐00003.			
n	Use ra	ange 00000000000 to 3777777777(8)				
D	Use	ange C *	Octal constant (02656237614)			
Condition	Rising	edge of input signal (OFF to ON)	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$			
Contents after	D to D+3	Result	Before operation			
operation	Flag	Unchanged	<u>−0003</u> <u>−00002</u> <u>−0000</u> 1001100011000000001101			
		addresses for register D. = 300011 etc. are prohibited to use.)	After operation	AND		

0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 0 1 1 0 1 1 0 0 0 0 0 0 0

AND register with hexadecimal constant (1 byte) Fx13 (AND)

Symbol	Fx13 AND	n D	[Explanation]	Instruction S T R 004001 F x13	
Function	with the	decimal constant "n" is ANDed contents of the register D and t is stored in the register D.	004001 Fx13 AND 53 009002	53 009002	
Operation	n ∩ D-	→ D	When the input condition 004001 changes from OFF to ON, the hexadecimal constant 53 is		
n	Use rar	nge 00 to FF	ANDed with the contents of the register 009002 and its result is stored in the register 009002.		
D	Use rar	nge A	Before operation		
Condition	Rising e	dge of input signal (OFF to ON)		After operation	
Contents after	D	Result	$\left \begin{array}{c} \text{AND} \\ $		
operation	Flag	Unchanged	Hexadecimal 0 1 0 1 0 0 1 1 (53)		
- AND truth ta	ble				

Symbol	Α	В	С
	0	0	0
	1	0	0
В⊣ ∕С	0	1	0
	1	1	1

Fx13w AND

AND

AND register with hexadecimal constant (1 word) (AND)

		-			
Symbol	Fx13w	n D	[Explanation]	Instruction S T R 004001 F x13w	
Function	with the D, D+1 a	ecimal constant "n" is ANDed 16-bit contents of the registers and its result is stored in the s D, D+1.	When the input condition 0040	2D72 ⊐00000 01 changes from	
Operation	n ∩ D,	D+1→D, D+1	OFF to ON, the hexadecimal c ANDed with the 16 bits conten		
n	Use rar	nge 0000 to FFFF(H)	⊐00000 and ⊐00001 and its result is stored in the registers ⊐00000 and ⊐00001.		
D	Use rar	ige B *	Hexadecimal constant (2D	172)	
Condition	Rising e	dge of input signal (OFF to ON)		AND	
Contents	D, D+1	Result	Before operation _100001 _100000	-	
after operation	Flag	Unchanged	After operation 700001 700001 700000		
* Be sure to use even addresses for register D. (Odd address such as ⊐00011 etc. are prohibited to use.)					

Fx13d AND

AND register with hexadecimal constant (2 words) (AND)

Symbol	Fx13d A_N_D	n D		
Function	A hexadecimal constant "n" is ANDed with the 32-bit contents of the registers D to D+3 and its result is stored in the registers D to D+3.			
Operation	n ∩ D to D+3→D to D+3			
n	Use range 00000000 to FFFFFFF(H)			
D	Use range C *			
Condition	Rising edge of input signal (OFF to ON)			
Contents	D to D+3	Result		
after operation	Flag Unchanged			

[Explanation] Insti S T R F x13d A N D 16B93F8C 300000 1 When the input condition 004001 chan

Instruction S T R 004001 F x13d 16B93F8C ¬00000

When the input condition 004001 changes from OFF to ON, the hexadecimal constant 16B93F8C is ANDed with the 32 bits contents of the registers ¬00000 to ¬00003 and its result is stored in the registers ¬00000 to ¬00003.

н	Hexadecimal constant (16B93F8C)						
0010110101111001000101101011110010 							
Before operation ⊐00003	⊐00002	⊐00001	⊐00000				
10011000	11001000	01101101	10100011				
After operation	⊐00002	⊐00001	AND				
00010000	10001000	00101101	1000000				

* Be sure to use even addresses for register D.

(Odd address such as ⊐00011 etc. are prohibited to use.)

OR register with register (1 byte) F-14 OR (OR)

Symbol	F-14 OR	S D	[Explanation]	Instruction STR 004002 F-14	
Function	are ORe register	tents of the register S (8 bits) ed with the contents of the D (8 bits) and its result is n the register D.	004002 F-14 009000 009002 OR 009000 009002 009002	009000 009002 2 changes from	
Operation	SUD→I	כ	OFF to ON, the 8 bits contents of the register 009000 are ORed with the 8 bits contents of the register 009002 and its result is stored in the register 009002. The contents of the register		
S	Use ran	ge A			
D	Use ran	ge A	009000 remain unchanged.	-	
Condition	Rising e	dge of input signal (OFF to ON)	Before operation	After operation	
Contents	S	Unchanged	009000 0 1 0 1 0 0 1 1	-Ditto- 009000	
after operation	D	Result	OR		
operation	Flag	Unchanged	009002 1 0 1 1 1 0 0 1	+ 1 1 1 1 1 0 1 1 009002	

- OR truth table

F-14w OR

Symbol	A	В	С
	0	0	0
	1	0	1
В	0	1	1
-	1	1	1

OR register with register (1 word) (OR)

Symbol	F-14w OR	S D	[Explanation]		STR	uction 004000
Function	S+1 are of the re	bit contents of the registers S, ORed with the 16-bit contents egisters D, D+1 and its result is in the registers D, D+1.	When the inp	009000 009002 put condition 00400	•	
Operation	S, S+1 l	JD, D+1→D, D+1	,	he 16-bit contents 009001 are ORed	•	0
S	Use ran	ge B *	contents of the registers 009002 and 009003 ar its result is stored in the registers 009002 and			
D	Use ran	ge B *	009003.		000	
Condition	Rising e	dge of input signal (OFF to ON)		01011001010101	1101	OR
Contents	S, S+1	Unchanged	Before operation		002	
after	D, D+1	Result		1 1 0 1 0 1 0 0 1 0 1 0		
operation	Flag	Unchanged	After operation	009003 009	1 1 1 1	
Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to			The contents	s of the registers 0		d 009001

(Odd address such as 019003 etc. are prohibited to use.)

remain unchanged.

F-14d OR	OR r (OR)	egister with register	r (2 words)		
Symbol	F-14d OR	S D	[Explanation]	Instruction S T R 004000 F -14d	
Function	(32 bits) of the re	tents of the registers S to S+3 are ORed with the contents egisters D to D+3 (32 bits) and t is stored in the registers D to	When the input condition 00400 OFF to ON, the 32-bit contents 009000 to 009003 are ORed wi	009000 009004 00 changes from of the registers	
Operation	S to S+3	3∪D to D+3→D to D+3	contents of the registers 009004 to 009007 and its result is stored in the registers 009004 to 009007.		
S	Use ran	ge C *	Before operation	0001 009000	
D	Use ran	ge C *			
Condition	Rising e	dge of input signal (OFF to ON)	OR 009007 009006 005	0005 009004	
Contents	S to S+3	Unchanged	1 1 0 1 0 1 0 0 1 0 1 0 0 1 1 0 1 1 0 1	01001010110	
after	D to D+3	Result	After operation	0005 009004	
operation	Flag	Unchanged	1 1 0 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1	1 1 0 1 1 1 1 1 1 1 1 1 1	
		Idresses for registers S and D. 19003 etc. are prohibited to use.)	The contents of the registers 00 remain unchanged.	99000 to 009003	

Fc14
OROR register with octal constant (1 byte)
(OR)

Symbol	Fc14 OR	n	D			[Explanation]	STR	uction 004001
Function	content	l constar s of the r s stored in	egiste	r D and		004001 Fc14 OR 123 009002	F c14	123 009002
Operation	n∪D-	→D				When the input condition 004001 changes from OFF to ON, the octal constant 123 is ORed with		
n	Use rar	nge 000 t	o 377			the contents of the register 0090 stored in the register 00902.		
D	Use rar	nge A				Before operation		
Condition	Rising e	dge of in	put sig	gnal (C	FF to ON)		After en	ration
Contents	D	Result				>OR	After ope → 1 1 1 1 1	0 1 1 009002
after operation	Flag	Unchar	nged			Octal $(1 \sqrt{2} \sqrt{3})$ constant 0 1 0 1 0 0 1 1		
- OR truth tab	le					(123)		
Symbo	bl	A	В	С				
A		0	0	0				
	≻_c	1	0	1				
B →	/ -	0	1	1				
		1	1	1				

Fc14w
OROR register with octal constant (1 word)
(OR)

Symbol	Fc14w OR	n D	[Explanation]	Instruction S T R 004001 F c14w
Function	the 16- and D+	Il constant "n" is ORed with bit contents of the registers D 1 and its result is stored in the s D and D+1.		out condition 00400	026562 ¬00000 01 changes from
Operation	n∪D,	D+1→D, D+1	 OFF to ON, the octal constant 026562 is ORed with the 16 bits contents of the registers ¬00000 and ¬00001 and its result is stored in the registers ¬00000 and ¬00001. 		
n	Use rar	nge 000000 to 177777(8)			
D	Use rar	nge B *		Octal constant (0265	
Condition	Rising e	dge of input signal (OFF to ON)			
Contents after	D, D+1	Result	Before operation		
operation	Flag	Unchanged	A.()	⊐00001 ⊐0000	
		ddresses for register D.	After operation		

(Odd address such as J00011 etc. are prohibited to use.)

Fc14d OR

Symbol	Fc14d OR	n D	[Explanation]	Instruction S T R 004001 F c14d	
Function	the 32- to D+13	Il constant "n" is ORed with bit contents of the registers D 3 and its result is stored in the 's D to D+3.	$\begin{array}{c c} 004001 \\ \hline \\ $	02656237614 _00000 01 changes from	
Operation	n U D I	o D+3→D to D+3	OFF to ON, the octal constant 02656237614 is ORed with the 16 bits contents of the registers ⊐00000 to ⊐00003 and its result is stored in the registers ⊐00000 to ⊐00003. Octal constant (02656237614)		
n	Use rang	e 00000000000 to 3777777777(8)			
D	Use rar	ige C *			
Condition	Rising e	dge of input signal (OFF to ON)			
Contents	D to D+3	Result			
after operation	Flag	Unchanged	After operation	AND	
		dresses for register D. 00011 etc. are prohibited to use.)	<u>100003</u> <u>100002</u> <u>10000</u>		

OR register with hexadecimal constant (1 byte) (OR)

	()				
Symbol	Fx14 OR	n D	[Explanation]	Instruction S T R 004001 F x14	
Function	with the	decimal constant "n" is ORed contents of the register D result is stored in the register	004001 Fx14 53 009002 When the input condition 0040	53 009002	
Operation	n U D	→ D	OFF to ON, the hexadecimal constant 53 is ORed with the contents of the register 009002 and its result is stored in the register 009002.		
n	Use rar	nge 00 to FF			
D	Use rar	nge A	Before operation		
Condition	Rising e	dge of input signal (OFF to ON)	009002 1 0 1 0 1 0 1 0 OR	After operation	
Contents after	D	Result		→ 1 1 1 1 1 0 1 1 009002	
operation	Flag	Unchanged	constant (53)		
- OR truth tab	ماه				

- OR truth table

Fx14w OR

Fx14 OR

Symbol	Α	В	С
	0	0	0
	1	0	1
В	0	1	1
-	1	1	1

OR register with hexadecimal constant (1 word) (OR)

Symbol	Fx14w	n D	[Explanation]				
				STR 004001 Fx14w			
Function	with the	lecimal constant "n" is ORed 16-bit contents of the registers I+1 and its result is stored in the	004001 Fx14w OR 2D72 300000	2D72 ⊐00000			
	register	s D and D+1.	When the input condition 0040	01 changes from			
Operation	n U D,	D+1→D, D+1	OFF to ON, the hexadecimal constant 2D72 is ORed with the 16-bit contents of the registers ⊐00000 and ⊐00001 and its result is stored in the registers ⊐00000 and ⊐00001.				
n	Use rar	nge 0000 to FFFF(H)					
D	Use rar	nge B *	C C				
Condition	Rising e	dge of input signal (OFF to ON)					
Contents after	D, D+1	Result	D7 Before operation	-2/ OR			
operation	Flag	Unchanged					
* Be sure to use even addresses for registers D. (Odd address such as ⊐00011 etc. are prohibited to use.)			After operation _00001 _00000				

Fx14d OR

OR register with hexadecimal constant (2 words) (OR)

Symbol	Fx14d OR	n D			
Function	A hexadecimal constant "n" is ORed with the 32-bit contents of the registers D to D+3 and its result is stored in the registers D to D3.				
Operation	n ∪ D to D+3→D to D+3				
n	Use range 00000000 to FFFFFFF(H)				
D	Use range C *				
Condition	Rising edge of input signal (OFF to ON)				
Contents after	D to D+3	Result			
operation	Flag	Unchanged			

[Explar	nation	Instruction					
					STR Fx14d	004001	
004001	Fx14d OR	16B93F8C	⊐00000		16	893F8C ⊐00000	
When the input condition 004001 changes from OFE to ON, the bexadecimal constant 16B93E8C							

OFF to ON, the hexadecimal constant 16B93F8C is ORed with the 32-bit contents of the registers □00000 to □00003 and its result is stored in the registers □00000 to □00003.

F	Hexadecimal constant (16B93F8C)						
Before operation ⊐00003		⊐00001	⊐00000				
10011000	11001000	01101101	1010001	1			
After operation	⊐00002	⊐00001	⊐00000	OR			
10011110	1111001	01111111	1010111	1			

* Be sure to use even addresses for registers D. (Odd address such as ⊐00011 etc. are prohibited to use.)

Chapter 11 Application instructions (F-15 to F-49)

F-1 MU		Multiplies (MULtiply)	regis	ter b	y re	gister (BCD 4 digits)			
Sym	nbol	- F-15 S1 S2 D				[Explanation]	Instruction S T R 004001 F -15		
Func	tion	The contents of the registers S1 and S1+1 (BCD 4 digits) are multiplied by the contents of the registers S2 and S2+1 (BCD 4 digits) and its result is stored in the 4-byte area starting from the register D.				$ \begin{array}{c c} 004001 \\ \hline F-15 \\ MUL \end{array} \begin{array}{c} 009000 \\ 009010 \\ 009020 \end{array} \end{array} \begin{array}{c} 009000 \\ 009010 \\ 009020 \end{array} \\ \hline \end{array} \\ \begin{array}{c} 009000 \\ 009020 \\ 009020 \end{array} \\ \hline \end{array} \\ \begin{array}{c} 009000 \\ 009020 \\ 009020 \end{array} \\ \hline \end{array} \\ \begin{array}{c} 009000 \\ 009020 \\ 00900 \\ 009000 \\ 009000 \\ 00900 \\ 00900 \\ 009000 \\ 009000 \\ 00900 \\ 0000 \\ 009000 \\ 009000 \\ 009000 \\ 009000 \\ 00900 \\ 009000 \\ 009000 \\ 009000 \\ 00900 \\ 009000 \\ 00000 \\ 00000 \\ 00000 \\ 00000 \\ 0000 \\ 00000 \\ 0000 \\ 00000 \\ 00000 \\ 0000 \\ 0000 \\ 0000$			
Opera	ation	(S ₁ , S ₁ +1) × (S ₂ , S ₂ +1) →D, D+1, D+2, D+3				registers 009010 and 009011 and its result is stored in the 4-byte area starting from the register 009020.			
S	1	Use range B							
S	2	Use range B				Before operation Tens Ones	After operation		
D)	Use range C					0 0 009020		
Cond	lition	Rising edge of inp	ut signa	l (OFF	to ON)	Thousands Hundreds 009001 0 1	3 4 009021		
	S1, S1+	1 Unchanged				X Tens			
	S2, S2+	1 Unchanged				009010 3 4	1 2 009022		
	D	Result in ones and tens	· ·	inged if		Thousands Hundreds			
Contents	D+1	Result in hundreds and thousands S1, S1+1, S2, and		009011 1 2	0 0 009023				
after operation	D+2	D+2 Result in ten thousands and hundred thousands code.		The above example shows the operation of					
	D+3 Result in and ten m		1	not calc	culate.)	100×1234=123400.			
	Flag	Register S ₁ , Zero S ₁ +1, S ₂ , S ₂ +1 00735	Carry 007356		Non-carry 007354				
	l	BCD code 0	0	0	0				

F-15d MUL MUL (MULtiply)

Sym	nbol	- F-15d MUL S1 S2 D	[Explanation] Instruction S T R 004001 F -15d
Func	tion	The contents of the registers S1 to S1 (BCD 8 digits) are multiplied by the contents of the registers S2 to S2+3 (BCD 8 digits) and its result is stored the 8-byte area starting from the regis D.	-3 -3 004001 F-15d 009000 009004 009004 MUL 009000 009004 009020 009020 009020 When the input condition 004001 changes from
Opera	ation	(S₁ to S₁+3) × (S₂ to S₂+3)→D to D	In the o-byte area starting norm the registers
S	1	Use range C *	009020.
S	2	Use range C *	Before operationAfter operation 009000 10^1 10^0 009000 009020
C)	Use range G *	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
Cond	lition	Rising edge of input signal (OFF to OI) $009002 \begin{bmatrix} 10^5 & 10^4 \\ 3 & 4 \end{bmatrix}$ 0 1 009022
	S1 to S1-	⁻³ Unchanged	$009003 \begin{array}{c c} 10^7 & 10^6 \\ 1 & 2 \end{array} \qquad \qquad$
	S2 to S2-	³ Unchanged	\times \rightarrow $\{$
		Operation result MSB LSB Unchanged whe	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
Contents after		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	$009005 \begin{array}{ c c c c c c c c c c c c c c c c c c c$
operation	$\begin{array}{c c c c c c c c c c c c c c c c c c c $		
		$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	
		S1 to S1+3, Zero Carry Error Non- 007355 S2 to S2+3 007357 007356 007355 007	
	Flag	BCD code 0 0 0 Not BCD code 0 0 1	12340100×12340100=152278068010000.

* Be sure to use even addresses for registers S₁, S₂, and D. (Odd address such as 019003 etc. are prohibited to use.)

Fc15 MUL

Multiplies register (BCD 4 digits) by BCD constant (3 digits) (MULtiply)

Sym	npol	- Fc15 MUL S1 n D	[Explanation] Instruction S T R 004001 F c15			
Func	tion	The contents of the registers S1 and S1+1 (BCD 4 digits) are multiplied by a 3-digit BCD constant "n" and its result is stored in the 4-byte area starting from the register D.	004001 Fc15 009000			
Opera	ation	(S1, S1+1) × n→D, D+1, D+2, D+3	and 009001 are multiplied by the 3-digit BCD constant 100 and its result is stored in the 4-byte			
S	1	Use range B	area starting from the registers 009020.			
n)	Use range 000 to 999	3412×100=341200			
C)	Use range C				
Cond	lition	Rising edge of input signal (OFF to ON)	009001 0 0 1 1 0 1 0 0 → 0 0 0 1 0 0 1 0 009021			
	S1, S1-	1 Unchanged				
	D	Result in ones and tens Unchanged when	X 0 0 1 1 0 1 0 0 009022			
	D+1	Result in hundreds and thousands the contents of registers S1 and	BCD constant 100			
Contents after operation	D+2	Result in ten thousands and hundred thousands S1+1 are not BCD code.				
operation	D+3	Result in millions and ten millions (Does not calculate.)				
	Flag	S1, S1+1 Zero 007357 Carry 007356 Error 007356 Non-carry 007355 BCD code 0 0 0 0 0 0 Not BCD code 0 0 1 0 0	-			

Fc15d
MULMultiplies register (BCD 8 digits) by BCD constant (4 digits)MUL(MULtiply)

Sym	lool	-Fc15d S1 n D			[Explanation]		Instruc S T R F c15d	ction 004001	
Func	tion	The contents of the registers S1 to S1+3 (BCD 8 digits) are multiplied by a 4-digit BCD constant "n" and its result is stored in the 8-byte area starting from the register D.				$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			
Opera	ation	(S₁ to S₁+3) ×	< n→D to) D+5		constant	0100 and its result is	s stored in th	
S	1	Use range C				area start	ting from the register	s 009020.	
n	1	Use range 0000 to 9999				12343412×100		1	
D)	Use range G				009000			009020
Cond	lition	Rising edge of input signal (OFF to ON)			009001	0 0 1 1 0 1 0 0	00010010	009021	
	S1 to S1+3	Unchanged	d			009002	0 0 1 1 0 1 0 0		009022
		$D+1 \frac{10^3}{10^5} 1$	$\frac{10^0}{10^2}$ Ur $\frac{10^2}{10^4}$ the	nchanged w e contents c gisters S1 to	of				009023
Contents after operation	D to D-	D+3 10^7 1 D+4 10^9 1	$\frac{10^6}{10^8}$ are	are not BCD code. (Does not calculate.)		BCD constar		00010010	009024
	D+6 0 0 D+7 0 0			0 0		009025			
	Flag	S1 to S1+3	007357 007	Carry 007356 Error 007355 Non-carry 007354 0 0 0 0 1 0					009026
							l		009027
						4			

F-16 DIV

Divide register (BCD 4 digits) by register (BCD 2 digits) (DIVide)

		(=====)	
Sym	ibol	- F-16 S1 S2 D	[Explanation] Instruction S T R 004001
Func	tion	The contents of the registers S1 and S1+1 (BCD 4 digits) are divided by the contents of the register S2 (BCD 2 digits) and the quotient is stored in the 2-byte area starting from the register D and the remainder in the third byte area	$\begin{array}{c c c c c c c c c c c c c c c c c c c $
Opera	ation	$(S_1, S_{1+1}) \div S_2 \rightarrow D, D+1, D+2$	2-byte area starting from the registers 009020 with the remainder in the third byte area.
St	1	Use range B	Before operation → Before operation
Sz	2	Use range A	Tens Ones Tens Ones
D)	Use range E	O09000 Thousands Hundreds Thousands Hundreds Thousands Hundreds Construction
Cond	lition	Rising edge of input signal (OFF to ON)	
	S1, S1	+1 Unchanged	Tens Ones Tens Ones
	S2	Unchanged	009010 2 1 1 6 009022 Remainder
Contents	D	Quotient (in ones and tens) Unchanged when the contents of registers,	The above example shows the operation of 1234 divided by 21 equals 58 with the remainder of 16.
after operation	D+1	Quotient (in hundreds and thousands) S1, S1+1, and S2 are not BCD code or the	
	D+2	Remainder contents of S2 is 00. (Does not calculate.)	- If denominator is not greater than numerator $(S_1 < S_2, S_1+1=0)$, the quotient (contents of D, D+1) is 0 and
	Flag	S1, S1+1, S2 Zero 007357 Carry 007356 Error 007355 Non-car 007355 BCD code 0 </td <td>T_{y} remainder (contents of D+2) is numerator (contents</td>	T_{y} remainder (contents of D+2) is numerator (contents

Reference To obtain a result of 2 places under the decimal point, the following programming is suggested. Example: $1983 \div 58=34.18 \dots$ remainder 0.56

004001	F-16 DIV	009000	009010	009020		(1)	In
	F-01 BCD	00	009023			(2	<u>?)</u>	S T F F - 16
	F-02 XCH	009022	009023]		(3	3)	
	F-16 DIV	009022	009010	009015		(4	ł)	F -01
009000 <u>83</u> 009001 <u>19</u>]	009	020	34 00	Quotient	} (1)	F -02
009010 58	÷]		022 [11 00	Remainder	·	<u>?)</u>	F -16
009022 00	1		022	00 11	}	(3	3)	
009023 11] ÷—		015	18 00	Quotient) (4	+)	
009010 58]		017	56	Remainder	·)		

Instruction						
STR	004001					
F-16	009000 009010 009020					
F -01	00					
F 00	009023					
F -02 F -16	009022 009023					
1-10	009022 009010 009015					

When the input condition 004001 changes from OFF to ON, the contents of registers 009000 and 009001 are divided by the contents of the register 009010 and the quotient is stored in 009020 and 009021 and remainder is stored in 009022.
 Enter det 2000 in 009022.

(2) Enter data 00(H) in 009023.

(3) The contents of 009022 are exchanged with 009023 and a remainder is converted into thousands and hundreds.

 (4) The data in (3) is divided by the contents of 009010 again, and a quotient is stored in 009015 and 009016 with a remainder stored in 009017. The result under 2 digits of decimal place is then stored in 009015.

11-5

Divide register (BCD 8 digits) by register (BCD 8 digits) F-16d (DIVide)

Sym	bol	F-16d S1 S2 D				[Explanation] Instruction	
U U U	501					STR 004001	
Function		The contents of the registers S1 to S1+3 (BCD 8 digits) are divided by the contents of the registers S2 to S2+3 (BCD 8 digits) and the quotient is stored in the register D to D+3 and the remainder in the D+4 to D+7.				$\begin{bmatrix} 004001 \\ \hline F-16d \\ DIV \end{bmatrix} \xrightarrow{F-16d \\ 009000 \\ 009010 \\ 009020 \end{bmatrix} \xrightarrow{F-16d \\ 009000 \\ 009010 \\ 009020 \end{bmatrix}$ When the input condition 004001 changes from OFF to ON, the BCD 8 digits in registers 009000 to 009003 are divided by the BCD 4 digits in registers	
Opera	ation	$(S_1 \text{ to } S_1+3) \div S_2 \text{ to } S_2+3 \rightarrow D \text{ to } D+7$				009010 to 009013 and the quotient is stored in the registers 009020 to 009023 with the remainder in	
S1 Use range C *					the 009024 to 009027.		
Sa	2	Use range C *				Before operation After operation	
D	1	Use range G *				$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	
Cond	ition	Rising edge of input signal (OFF to ON)				2 8 Quotient	
	S1 to						
	<u>S1+3</u> S2 to S2+3				$\begin{array}{c ccccccccccccccccccccccccccccccccccc$		
Contents after	D to D+3	Quotient (8-digit BCD)	Unchanged v of registers S S2+3 are not	1 to S1+3,	and S2 to	$009010 \begin{array}{ c c c c c } 10^1 & 10^0 \\ 2 & 1 \end{array} \right) \begin{array}{ c c c c } 10^1 & 10^0 \\ 8 & 1 \end{array} 009024 \bigg)$	
operation	D+4 t D+7	0 Remainder (8-digit BCD)	contents of S (Does not ca	2 to S2+3 a		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	
	-		Zero Carry 07357 00735		Non-carry 007354	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	
	Flag	BCD code Not BCD code If S2 to S2+3 are 00000000(H)	0 0	0	0	$\begin{array}{c} 009013 \boxed{10^7 10^6} \\ 0 \end{array} \end{array} \qquad \qquad$	
	* Be sure to use even addresses for registers S ₁ , S ₂ , and D.				12345678 divided by 4321 equals 2857 with the remainder of 581.		

(Odd address such as 019003 etc. are prohibited to use.)

DIV

- If denominator is not greater than numerator (S1 to S1+3 < S2 to S2+3), the quotient (contents of D to D+3) is 0 and remainder (contents of D+4 to D+7) is numerator (contents of S1 to S1+3). For instance, 20/30 will produce the result of 0 with a remainder of 20.

Fc16 DIV

Divide register (BCD 4 digits) by BCD constant (2 digits) (DIVide)

Sym	ibol	- Fc16 S1 n D				[Explanation] Instruction S T R 004001		
Function		The contents of the registers S1 and S1+1 (BCD 4 digits) are divided by a 2- digit BCD contents "n" and the quotient is stored in the 2-byte area starting from the register D and the remainder in the third byte area.				$\begin{array}{c c c c c c c c c c c c c c c c c c c $		
Operation		$(S_1, S_1+1) \div n \rightarrow D, D+1, D+2$				and the quotient is stored in the 2-byte area starting from the register 009020 with the remainder in the third byte area.		
S1		Use range B				remainder in the third byte area.		
n		Use range 00 to 99				8765÷21=417 remainder 8		
D)	Use range E				009000 0 1 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0		
Cond	ition	Rising edge of input signal (OFF to ON)			to ON)			
	S1, S1-	+1 Unchanged						
	D	Quotient (in ones and tens		inged wh				
Contents	D+1	Quotient (in hundre and thousands)	ds S1 and	contents of registers S1 and S1+1 are not BCD code or the contents of "n" is 00. (Does not calculate.)		$\begin{array}{c c c c c c c c c c c c c c c c c c c $		
after	D+2	Remainder	-					
operation		S1, S1+1, "n" Ze		Error 007355	Non-carry 007354			
	Flag	BCD code Not BCD code If "n" is 00	0	0	0			

- If denominator is not greater than numerator (S₁ < n, S₁+1=0), the quotient (contents of D, D+1) is 0 and remainder (contents of D+2) is numerator (contents of S₁). For instance, 20/30 will produce the result of 0 with a remainder of 20.

Divide register (BCD 8 digits) by BCD constant (4 digits) Fc16d (DIVide)

Sym	npol	- Fc16d S1 n D	[Explanation] Instruction S T R 004001		
Function		The contents of the registers S1 to S1+3 (BCD 8 digits) are divided by a 4-digit BCD contents "n" and the quotient is stored in the register D to D+3 with the remainder in the registers D+4 to D+7.	$ \begin{array}{c c c c c c c c c c c c c c c c c c c $		
Opera	ation	(S1 to S1+3) \div n \rightarrow D to D+7	and the quotient is stored in the registers 009020 to 009023 with the remainder in the 009024 to 009027.		
S	1	Use range C *	Before operation \rightarrow After operation		
n	1	Use range 0000 to 9999	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$		
D)	Use range G *			
Cond	lition	Rising edge of input signal (OFF to ON)	$ \begin{bmatrix} 009001 & 10^{-} & 10^{-} \\ 009002 & 0 & 0 \\ 009002 & 6 & 5 \end{bmatrix} $		
	S1 to S1+3	Unchanged	$10^{7} 10^{6} 10^{7} 10^{4} 000032$		
	D to D+	-3 Quotient (8-digit BCD) Unchanged when the contents of registers S1 to S1+3 are not BCD code			
Contents after operation	D+4 to D+7	or the contents of "n" is	$\begin{array}{c c c c c c c c c c c c c c c c c c c $		
	Flag	S1 to S1+3, "n" Zero 007357 Carry 007356 Error 007356 Non-carry 007355 BCD code 0 <t< td=""><td>$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \xrightarrow{\text{Remainder}} \\ \begin{bmatrix} 105 & 10^4 \\ 0 & 0 \end{bmatrix} \xrightarrow{\text{009026}} \\ \begin{bmatrix} 10^7 & 10^6 \\ 0 & 0 \end{bmatrix} \xrightarrow{\text{000027}}$</td></t<>	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \xrightarrow{\text{Remainder}} \\ \begin{bmatrix} 105 & 10^4 \\ 0 & 0 \end{bmatrix} \xrightarrow{\text{009026}} \\ \begin{bmatrix} 10^7 & 10^6 \\ 0 & 0 \end{bmatrix} \xrightarrow{\text{000027}} $		
	Flag	S1 to S1+3, "n" 007357 007356 007355 007354 BCD code 0	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$		

 * Be sure to use even addresses for registers S₁ and D. (Odd address such as 019003 etc. are prohibited to use.)

DIV

The above example shows the operation of 87650000 divided by 21 equals 4173809 with the remainder of 11.

- If denominator is not greater than numerator (S1 to S1+3<n), the quotient (contents of D to D+3) is 0 and remainder (contents of D+4 to D+7) is numerator (contents of S1 to S1+3). For instance, 20/30 will produce the result

F-17 XNR

Exclusive NORs register with register (1 byte) (eXclusive NoR)

Symbol	F-17 XNR	S D				
Function	The contents of the register S are XNRed with the contents of the register D and its result is stored in the register D.					
Operation	S⊕D→D					
S	S Use range A					
D	Use range A					
Condition	Rising edge of input signal (OFF to O					
Contents	S	Unchanged				
after	D	Result				
operation	Flag	Unchanged				

	[Explanation]	Instruction
		STR 004001 F-17
	004001 F-17 XNR 009000 009001	009000 009001
	When the input condition 00400 OFF to ON, the contents of the are XNRed with the contents of 009001 and its result is stored in 009001. The contents of the reg remains unchanged.	register 009000 the register n the register
N)	Before operation	After operation
	009000 0 0 0 0 1 1 1 1 1 XNR	-Ditto009000
		- 0 1 1 1 1 1 1 0 009001

Bit matched in 009000 and 009001 (0 and 0, 1 and 1) is turned to 1 and unmatched bit (0 and 1) is turned to 0.

- Exclusive NOR truth table

F-17w XNR

Symbol	A	В	С
ĺ	0	0	1
	1	0	0
В₩	0	1	0
	1	1	1

Exclusive NORs register with register (1 word) (eXclusive NoR)

Symbol	F-17w XNR	S D	[Explanation]	Instruction S T R 004000 F -17w		
Function	S+1 are contents	bit contents of the registers S, XNRed with the 16-bit s of the registers D, D+1 and t is stored in the registers	When the input condition 0040 OFF to ON, the 16-bit contents	009000 009002 00 changes from s of the registers		
Operation	<mark>S, S+1</mark> ⊕)D, D+1→D, D+1	009000 and 009001 are XNRed with the 16-bit contents of the registers 009002 and 009003 and			
S	Use ran	ge B *	its result is stored in the registers 009002 and 009003. The contents of the registers 009000 and 009001 remain unchanged.			
D	Use ran	ge B *				
Condition	Rising e	dge of input signal (OFF to ON)				
Contents	S, S+1	Unchanged		XNR		
after	D, D+1	Result	009003 00 Before operation 1 1 0 1 0 1 0 0 1 0 1 0	9002		
operation	Flag	Unchanged	009003 00	9002		
* Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to use.)			After operation 0 1 1 1 0 0 1 0 0 0 0			

11-9

F-17d
XNRExclusive NORs register with register (2 words)(eXclusive NoR)

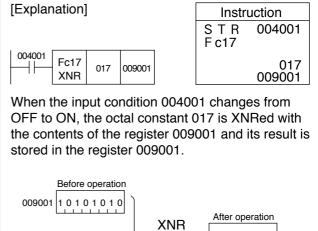
Symbol	-F-17d S D					
Function	The contents of the registers S to S+3 (32 bits) are XNRed with the contents of the registers D to D+3 (32 bits) and its result is stored in the registers D to D+3.					
Operation	$\overline{\text{S to S+3} \oplus \text{D to D+3}}$ →D to D+3					
S	Use range C *					
D	Use range C *					
Condition	Rising edge of input signal (OFF to ON)					
Contonto	S to S+3	Unchanged				
Contents after	D to D+3	Result				
operation	Flag	Flag Unchanged				

* Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to use.) [Explanation] Instruction STR 004000 F -17d 004000 F-17d 009000 009000 009004 ┥┝ XNR 009004 When the input condition 004000 changes from OFF to ON, the 32-bit contents of the registers 009000 to 009003 are XNRed with the 32-bit contents of the registers 009004 to 009007 and its result is stored in the registers 009004 to 009007. The contents of the registers 009000 to 009003 remain unchanged. Before operation 009003 009002 009001 009000 010110010101110101010110010101011101 (XNR) 009007 009006 009005 009004 11010100101010011011010100100100110 After operation 009007 009006 009005 009004 0 1 1 1 0 0 1 0 0 0 0 0 1 0 0 0 1 1 1 0 0 1 0 0 0 0 0 1 0 0

Fc17 XNR

Exclusive NORs register with octal constant (1 byte) (eXclusive NOR)

Symbol	Fc17 XNR	n D	[Explanation]		
Function	An octa the cont result is	004001 Fc17 XNR 017			
Operation	n⊕D→	D	When the input cor OFF to ON, the oct		
n	Use rar	nge 000 to 377	the contents of the stored in the registe		
D	Use rar	nge A	Before operatior		
Condition	Rising ed	dge of input signal (OFF to ON)	009001 1 0 1 0 1 0 1 0		
Contents	D	Result			
after operation	Flag	Unchanged	Octal constant (017)		



0 1 0 1 1 0 1 0 009001

- Exclusive NOR truth table

Fc17w XNR

Symbol	A	В	С
	0	0	1
	1	0	0
В₩	0	1	0
	1	1	1

Exclusive NORs register with octal constant (1 word) (eXclusive NoR)

Symbol	Fc17w XNR	n D	[Explanation] Instruction S T R 004001		
Function	the 16 t and D+	I constant "n" is XNRed with bits contents of the registers D 1 and its result is stored in the s D and D+1.	Image: marked system Image: marked system F c17w Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system Image: marked system		
Operation	n⊕D, [D+1→D, D+1			
n	Use rar	nge 000000 to 177777(8)			
D	Use rar	nge B *			
Condition	Rising ec	lge of input signal (OFF to ON)	0 0 1 0 1 1 0 1 0 1 1 1 0 0 1 0 0 2 6 6 5 6 2 XNR		
Contents	D, D+1	Result	□00001 □00000 Before operation 1 0 0 1 1 0 0 1 0 0 0 0		
after operation	Flag	Unchanged	=00001 =00000		
* Be sure to use even addresses for registers D.			After operation 0 1 0 0 1 0 1 0 0 1 0 0 1 0 1 0 1 0 4		

(Odd address such as ¬00011 etc. are prohibited to use.)

Fc17d
XNRExclusive NORs register with octal constant (2 words)(eXclusive NoR)

Symbol	Fc17d XNR	n D	[Explanation]	Instruction S T R 004001 F c17d		
Function	the 32 to D+3	al constant "n" is XNRed with bits contents of the registers D and its result is stored in the rs D to D+3.	004001 Fc17d 02656237614 00000 026562376 When the input condition 004001 changes from			
Operation	$\overline{n \oplus D \text{ to } D}$ +3 \rightarrow D to D+3		OFF to ON, the octal constant 02656237614 is XNRed with the 32-bit contents of the registers			
n	Use rai 000000	nge 000000 to 17777777777 ₍₈₎	□00000 to □00003 and its result is stored in the registers □00000 to □00003.			
D	Use rai	nge C *	Octal constant 0265	6237614		
Condition	Rising e	dge of input signal (OFF to ON)	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$			
Contents after	D to D+3	Result	Before operation	_		
operation	Flag Unchanged		Image: 100003 Image: 100002 Image: 100002 100011000 10000 10000 10000			
* Be sure to use even addresses for registers D. (Odd address such as ⊐00011 etc. are prohibited to use.)			After operation	XNR 1⊒00000		

0111000110001110101010110111010000

Fx17 XNR

Exclusive NORs register with hexadecimal constant (1 byte) (eXclusive NoR)

Symbol	Fx17 XNR	n D	[Explanation]	Instruction S T R 004001		
Function	with the	decimal constant "n" is XNRed contents of the register D and It is stored in the register D.				
Operation	n⊕D→D		When the input condition 004001 changes from OFF to ON, the hexadecimal constant 0F is			
n	Use range 00 to FF		XNRed with the contents of the register 009001 and its result is stored in the register 009001.			
D	Use range A					
Condition	Rising edge of input signal (OFF to ON)		Before operation 009001 1 0 1 0 1 0 1 0			
Contents after	D	Result		After operation		
operation	Flag	Unchanged	Hexadecimal			
- Exclusive N	OR truth	table	(OF)			

- Exclusive NOR truth table

Symbol	Α	В	С
	0	0	1
	1	0	0
В₩	0	1	0
	1	1	1

Fx17w
XNRExclusive NORs register with hexadecimal constant (1 word)
(eXclusive NoR)

Symbol			[Explanation]	Instruction S T R 004001		
Function	with the register	decimal constant "n" is XNRed 16-bit contents of the s D and D+1 and its result is n the registers D and D+1.	$\begin{bmatrix} 004001 \\ \hline Kx17w \\ X N R \end{bmatrix} \xrightarrow{2D72} 100000 \\ \hline 1000 \\ $			
Operation	$\overline{\mathbf{n} \oplus \mathbf{D}, \mathbf{D}+1} \rightarrow \mathbf{D}, \mathbf{D}+1$		OFF to ON, the hexadecimal constant 2D72 is XNRed with the 16-bit contents of the registers			
n	Use rar	nge 0000 to FFFF(H)	⊐00000 and ⊐00001 and its result is stored in the registers ⊐00000 and ⊐00001.			
D	Use range B *		Hexadecimal constant (2D72)			
Condition	Rising edge of input signal (OFF to ON)					
Contents after	D, D+1	Result	Before operation _00001 _0000	0		
operation	Flag	Unchanged	After operation	<u> </u>		
* Be sure to use even addresses for register D. (Odd address such as ⊐00011 etc. are prohibited to use.)			0 1 0 0 1 0 1 0 0 1 0 0 0	101		

Exclusive NORs register with hexadecimal constant (2 words) Fx17d XNR (eXclusive NoR)

Symbol			[Explanation]	Instruction S T R 004001 F x17d	
Function	A hexadecimal constant "n" is XNRed with the 32-bit contents of the registers D to D+3 and its result is stored in the registers D to D+3.		004001 Fx17d 16B93F8C 16B93F8C 16B93F8C When the input condition 004001 changes from 0FF to ON, the hexadecimal constant 16B93F8C is XNRed with the 32-bit contents of the registers ⊐00000 to ⊐00003 and its result is stored in the registers ⊐00000 to ⊐00003.		
Operation	$\overline{\mathbf{n} \oplus \mathbf{D} \text{ to } \mathbf{D}}$ +3→D to D+3				
n	Use range 00000000 to FFFFFFF(H)				
D	Use range C *		Hexadecimal constant (16B93	· · · · · · · · · · · · · · · · · · ·	
Condition	Rising edge of input signal (OFF to ON)		001011010111001001011 		
Contents	D to D+3 Result		Before operation ⊐00003 ⊐00002 ⊐0000 [.]	1 ⊐00000	
after operation	Flag	Unchanged	1001100011001000100110	0 0 0 1 1 0 0 1 0 0 0	
		ddresses for register D.	After operation _00003 _00002 _0000	1 <u>⊐00000</u> XNR	

Be sure to use even addresses for register D. (Odd address such as ¬00011 etc. are prohibited to use.)

F-18 XOR

Exclusive ORs register with register (1 byte) (eXclusive OR)

Symbol	F-18 XOR	S D	
Function	The contents of the register S are XORed with the contents of the register D and its result is stored in the register D.		
Operation	S⊕D→D		
S	Use range A		
D	Use range A		
Condition	Rising edge of input signal (OFF to ON)		
Contents	S	Unchanged	
after	D Result		
operation	Flag Unchanged		

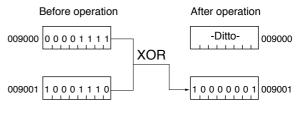
- Exclusive OR truth table

Symbol	Α	В	С
	0	0	0
	1	0	1
в₩	0	1	1
	1	1	0

[Explanation] 004001 F-18 009000 009001 When the input condition 004001 changes from

When the input condition 004001 changes from OFF to ON, the contents of the register 009000 are XORed with the contents of the register 009001 and its result is stored in the register 009001.

The contents of the register 009000 remain unchanged.



Bit matched in 009000 and 009001 (0 for 0, 1 for 1) is turned to 0 and unmatched bit (0 and 1) is turned to 1.

F-18w XOR

Exclusive ORs register with register (1 word) (eXclusive OR)

Symbol	F-18w XOR	S D	
Function	The 16-bit contents of the registers S, S+1 are XORed with the 16-bit contents of the registers D, D+1 and its result is stored in the registers D, D+1.		
Operation	S, S+1⊕D, D+1→D, D+1		
S	Use range B *		
D	Use range B *		
Condition	Rising edge of input signal (OFF to ON)		
Contents	S, S+1	Unchanged	
after	D, D+1 Result		
operation	Flag	g Unchanged	

	[Explanation]	Insti	ruction				
					STR	004000	כ
_					F -18w		
S,	004000 F-18w	009000	009002			009000	
J	XOR			l		009002	2
1	When the inp	ut con	dition	00400	0 chanc	nes from	
	OFF to ON, t				-	•	
	009000 and (•	
							4
	contents of th	•					u
	its result is st	orea ir	n the r	egiste	rs 00900	J2 and	
	009003.	009	9001	009	000		
		0101	1001	0101	1 1 0 1 -		
						XOR	
ON)		009	9003	009	002		
	Before operation	1101	0100	1010	0110 -		
		009	9003	009	002		
	After operation	1000	1101	1111	1011 🗲		

The contents of the registers 009000 and 009001

* Be sure to use even addresses for register S and D. (Odd address such as 019003 etc. are prohibited to use.)

remain unchanged.

F-18d
XORExclusive OR register with register (2 words)(eXclusive OR)

Symbol	-F-18d S D		[Explanation] Instruction S T R 004000 F -18d		
Function	The contents of the registers S to S+3 (32 bits) are XORed with the contents of the registers D to D+3 (32 bits) and its result is stored in the registers D to D+3.		$\begin{array}{c c} & & & & & \\ \hline \hline & & & \\ \hline \hline \\ \hline & & & \\ \hline \hline \\ \hline \hline & & & \\ \hline \hline \\ \hline \hline \\ \hline \hline \hline \\ \hline \\$		
Operation	S to S+3⊕D to D+3→D to D+3		contents of the registers 009004 to 009007 and its result is stored in the registers 009004 to 009007.		
S	Use range C *				
D	Use range C *		Before operation 009003 009002 009001 009000		
Condition	Rising edge of input signal (OFF to ON)		01011001010111010101010101010101011101		
Ocatonto	S to S+3	Unchanged	(XOR) 009007 009006 009005 009004		
Contents after	D to D+3	Result	After operation		
operation	Flag	Unchanged	009007 009006 009005 009004		
* Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to use.)			The contents of the registers 009000 to 009003 remain unchanged.		

Fc18 XOR

Exclusive ORs register with octal constant (1 byte) (eXclusive OR)

Symbol	Fc18 XOR	n D	[Explanation]	Instruction S T R 004001 F c18		
Function	the con	I constant "n" is XORed with tents of the register D and its stored in the register D.	004001 Fc18 017 009001 0090			
Operation	n⊕D→	D	When the input condition 004001 changes from OFF to ON, the octal constant 017 is XORed with			
n	Use rar	nge 000 to 377	the contents of the register 009 stored in the register 009001.	001 and its result is		
D	Use rar	nge A	Before operation			
Condition	Rising edge of input signal (OFF to ON)		009001 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1	After operation		
Contents after	D	Result	Octal $\begin{pmatrix} 0 \\ -1 \\ -7 \\ -7 \\ -7 \\ -7 \\ -7 \\ -7 \\ -7$	→ 10100101009001		
operation	Flag	Unchanged	constant 0 0 0 0 1 1 1 1 1 (017)			

- Exclusive OR truth table

Symbol	A	В	С
	0	0	0
	1	0	1
в₩О	0	1	1
	1	1	0

Fc18w
XORExclusive ORs register with octal constant (1 word)
(eXclusive OR)

Symbol	Fc18w XOR	n D	[Explanation]	Instruction S T R 004001 F c18w	
Function	the 16-t and D+	constant "n" is XORed with bit contents of the registers D and its result is stored in the s D and D+1.	$\begin{bmatrix} 004001 \\ \hline Fc18w \\ XOR \end{bmatrix} \xrightarrow{026562} 100000 \\ \hline 026562 \\ \hline 000000 \\ \hline 026562 \\ \hline 000000 \\ \hline 00000 \\ \hline 00000 \\ \hline 00001 \\ \hline 000001 \\ \hline 000001. \\ \hline 000001. \\ \hline 000001 \\ \hline 000001. \\ \hline 000001 \\ \hline 000000 \\ \hline 00000 \\ \hline 000000 \\ \hline 000000 \\ \hline 000$		
Operation	n⊕D, [)+1→D, D+1			
n	Use rar	ge 000000 to 177777(8)			
D	Use rar	ge 000000 B *	Octal constant 0265		
Condition	Rising ed	lge of input signal (OFF to ON)	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$		
Contents	D, D+1	Result	□00001 □000 Before operation 1 0 0 0 1 1 0 0		
after operation	Flag	Unchanged			
* Be sure to use even addresses for register D. (Odd address such as ⊐00011 etc. are prohibited to use.)			After operation 1 0 1 1 0 1 0 1 1 0 1 1	1010	

Exclusive ORs register with octal constant (2 words) Fc18d XOR (eXclusive OR)

Symbol	Fc18d XOR	n D	[Explanation]	Instruction S T R 004001 F c18d		
Function	the 32-b to D+3 a	I constant "n" is XORed with bit contents of the registers D and its result is stored in the s D to D+3.	004001 Fc18d 02656237614 02656237614 XOR 02656237614 00000 000000 When the input condition 004001 changes from 004001 000000			
Operation	n⊕Dto) D+3→D to D+3	OFF to ON, the octal constant 02656237614 is XORed with the 32-bit contents of the registers			
n	Use rar 000000	nge 00000 to 37777777777 ₍₈₎	☐ ⊐00000 to ⊐00003 and its result registers ⊐00000 to ⊐00003.	is stored in the		
D	Use rar	nge C*	Octal constant (0265	,		
Condition	Rising ed	lge of input signal (OFF to ON)	$ \begin{array}{c} 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1$			
Contents after	D to D+3	Result		1 =00000		
operation	Flag	Unchanged	10011000110000011000001101			
* Be sure to use even addresses for register D. (Odd address such as ⊐00011 etc. are prohibited to use.)			After operation	XOR 1 ⊐00000		

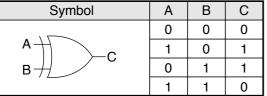
1000111001110001010101001000101111

Fx18 XOR

Exclusive ORs register with hexadecimal constant (1 byte) (eXclusive OR)

Symbol	Fx18 XOR	n D	[Explanation]	Instruction S T R 004001	
Function	with the	decimal constant "n" is XORed contents of the register D and It is stored in the register D.	When the input condition 0040	F x18 0F 009001	
Operation	n⊕D→	D	When the input condition 004001 changes from OFF to ON, the hexadecimal constant 0F is		
n	Use range 00 to FF		XORed with the contents of the and its result is stored in the re		
D	Use rar	nge A	Before operation		
Condition	Rising ed	lge of input signal (OFF to ON)	009001 1 0 1 0 1 0 1 0	After operation	
Contents	D	Result		$\rightarrow 101001000001$	
after operation	Flag	Unchanged	Hexadecimal constant (OF)		
- Exclusive OR truth table					

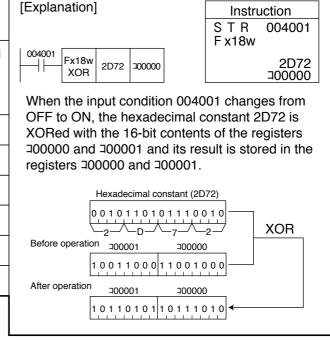
XOR



Exclusive ORs register with hexadecimal constant (1 word) Fx18w (eXclusive OR)

Symbol XOR I'' D'' FunctionA hexadecimal constant "n" is XORed with the 16-bit contents of the registers D and D+1 and its result is stored in the registers D and D+1. 004 Operation $n \oplus D, D+1 \rightarrow D, D+1$ Will OF XONUse range 0000 to FFFF(H) 10 registers DDUse range B *ConditionRising edge of input signal (OFF to ON)							
Functionwith the 16-bit contents of the registers D and D+1 and its result is stored in the registers D and D+1.With With OF With With OF With OF ParationWith With OF With OF With OF With OF OF ParationWith With OF With OF With OF OF ParationWith With OF With OF With OF ParationWith With With With OF Paration ParationWith With OF With OF Paration <td>Symbol</td> <td></td> <td>' n D</td> <td>[Exp</td>	Symbol		' n D	[Exp			
Operation $n \oplus D, D+1 \rightarrow D, D+1$ XCnUse range 0000 to FFFF(H)IODUse range B *ConditionRising edge of input signal (OFF to ON)Contents afterD, D+1ResultResult	Function	with the 16-bit contents of the registers D and D+1 and its result is					
n Use range 0000 to FFFF(H) reg D Use range B * Condition Rising edge of input signal (OFF to ON) Be Contents after D, D+1 Result	Operation	n⊕D, D)+1→D, D+1	XO			
Condition Rising edge of input signal (OFF to ON) Contents after D, D+1 Result	n	Use rar	nge 0000 to FFFF(H)	⊐00 regi			
Contents after D, D+1 Result	D	Use range B *					
after D, D+1 Result	Condition	Rising edge of input signal (OFF to ON)					
		D, D+1	Result	Bef			
		Flag Unchanged					

* Be sure to use even addresses for register D. (Odd address such as 300011 etc. are prohibited to use.)



Fx18d XOR

Exclusive ORs register with hexadecimal constant (2 words) (eXclusive OR)

Symbol	Fx18d XOR	n D	[Explanation]	Instruction S T R 004001	
Function	with the register	decimal constant "n" is XORed 32-bit contents of the s D to D+3 and its result is n the registers D to D3.	When the input condition 00400	•	
Operation	n⊕D to	D+3→D to D+3	OFF to ON, the hexadecimal constant 16B93F8C is XORed with the 32-bit contents of the registers		
n	Use rar	nge 00000000 to FFFFFFF(H)	⊐00000 to ⊐00003 and its resu registers ⊐00000 to ⊐00003.	It is stored in the	
D	Use rar	nge C *	Hexadecimal constant (16B93	,	
Condition	Rising ec	lge of input signal (OFF to ON)	0 0 0 1 0 1 1 0 1 0 1 1 1 0 0 1 0 0 1 1 1 1 	-F 8 C	
Contents	D to D+3	Result	Before operation 	100000	
after operation	Flag	Unchanged	1001100011000001100100001101		
* Be sure to use even addresses for register D. (Odd address such as ⊐00011 etc. are prohibited to use.)			Anter operation =00003 =00002 =0000 1 0 0 0 1 1 1 0 0 1 1 1 0 0 0 1 0 1 0 1		



Maintenance display

This instruction is identical to the F-20 instruction. => See MD "Maintenance display."

F-21 Obtains square root of register (BCD 8 digits) SQRT (SQuare RooT)

Sym	ibol	-F-21 S D	[Explanation]		Inst S T R F -21	ruction 000000
Func	tion	Determines the square root of the 8- digit BCD contents of registers S to S+3, and stores the result in register D. Truncates any decimal places.	When the input condition of 000000 changes from OFF to ON, this instruction obtains the			009200 nanges
Opera	ation	$\sqrt{(\text{S to S+3})} \rightarrow \text{D, D+1}$	square root of tl registers 00900	0 to 009003 ar	nd stores	
S	;	Use range C *	in registers 009]
D)	Use range B *	Before operation	2	1 3	009000
Cond	lition	Rising edge of input signal (OFF to ON)	√43214321	2	1	009002
	S to S+	3 Unchanged		4	3	009003
		Linchanged when the contents of		+		7
Contents	D, D+	Unchanged when the contents of registers S to S+3 are not BCD code.	After operation	7	3	009200
after ,		(Does not calculate.)	6573	6	5	009201
Flag		S to S+3 Zero 007357 Carry 007356 Error 007355 Non-carry 007354 BCD code 0 0 0 0 Not BCD code 0 1 0		Il decimal plac peration result		

* Be sure to use even addresses for registers S and D. (Odd address such as 019003 etc. are prohibited to use.)

Executes trigonometric function (SIN)

F-22 SIN

Sym	bol	-F-22 S D			[Explanation]	Instru S T R F -22	uction 004001		
Func	tion	Determines the SIN of the 6-digit BCD contents of registers S to S+2 and its result is stored in the registers D to D+3 (8-digit BCD).				d its	004001 F-22 009000 019000 SIN 009000 019000 019000)4001 cha	
Opera	ation	SIN (S to S+2)	→ D t	o D+3			from OFF to ON, this instruction of the 6-digit BCD contents of		
S	;	Use range E					to 009002, and stores the rest	•	
D)	Use range C					019000 to 019003.	-	009000
Cond	ition	Rising edge of	f input	signa	I (OFF	to ON)	0,1,1,0,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1	0,0,0,1,1,0,0	100001
Contents	S	Decimal fract angle. (2-digit	ion of a	an	ea of an	,	-654-	∕_3_∕_2 Decimal p	
before operation	S+1, S	Integer of on	angle.) to 999	9.99°	019003 019002 01		019000 1 0 0 1 1 1
	S to S+	2 Unchanged						<u> </u>	2-/-7-/
	D, D+	1 Decimal frac (4-digit BCD		of resu	lt.		Sign [+] Decimal poir		
Contents	D+2	Integer of re	,	2-digit	BCD)	*	The above example shows t SIN 6543.21°≑0.8927.	ne operat	ion of
after operation	D+3	A sign of result. [00(H): Positive (+) (2-digit BCD) [00(H): Negative (-)]		* D to D+3 (Contents after calcu	lation.)				
				Carry 007356		Non-carry 007354		nded off be	
	Flag	Positive (+)	0	0	0	1	to 1.0000.		
	Tiay	Negative (-)	0	1	0	0		f rogistors	S to S 2 aro
		S to S+2 are not BCD code.	0	0	1	0	 Unchanged when the contents c not BCD code. 	registers	5 10 5+2 die



Executes trigonometric function (COS)

Sym	nbol				[Explanation] Instruction S T R 004001 F -23
Fund	ction	Determines the COS of the 6-digit BCD contents of registers S to S+2 and its result is stored in the registers D to D+3 (8-digit BCD).		+2	004001 F-23 009000 019000 When the input condition of 004001 changes
Oper	ation	COS (S to S+2) \rightarrow D to	D+3		from OFF to ON, this instruction obtains the COS of the 6-digit BCD contents of registers 009000 to
S	3	Use range E			009002, and stores the result in registers 019000 to 019003.
C)	Use range C			009002 009001 009000
Cond	lition	Rising edge of input sign	al (OFF to	ON)	001000110100101011100111 -2
Contents before	S	Decimal fraction of an angle. (2-digit BCD)	Area of an	angle	Decimal point
	S+1, S-	-2 Integer of an angle. (4-digit BCD)	is 0 to 999	9.99°	019003 019002 019001 019000 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0 0 1 0 1 0 1 0 0 0 1
	S to S+	2 Unchanged			
	D, D+	1 Decimal fraction of re (4-digit BCD)	esult.		Sign [-] Decimal point
Oratoria	D+2	Integer of result. (2-d	igit BCD)	*	The above example shows the operation of COS 2345.67° \doteqdot -0.9951.
Contents after operation	D+3	A sign of result. (00(H): (2-digit BCD)	Positive (+) Negative (-)]	* D to D+3 (Contents after calculation.)
		Result Zero Carry Error Nor 007357 007356 007355 00			 The calculation result will be rounded off below the 5th digit to the right of the decimal, area of result is -1.0000
	Flag	Positive (+) 0 0 Negative (-) 0 1		1 0	to 1.0000.
		S to S+2 are not BCD code. 0 0	1	0	 Unchanged when the contents of registers S to S+2 are not BCD.

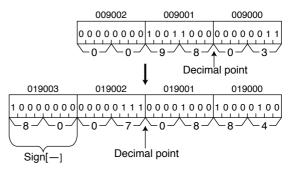
F-24 TAN

Executes trigonometric function (TAN)

Sym	bol		- F-24 S D						
Func	tion	E	Determines the TAN of the 6 digits BCD contents of registers S to S+2 and its result is stored in the registers D to D+3 (8-digit BCD).						
Opera	ation	Т	AN (S to S+2) → D	to D+3	3			
S		U	lse range E						
D		U	lse range C						
Condi	ition	R	ising edge of	input s	ignal	(OFF t	o ON)		
Contents	S		Decimal fraction of an angle. (2-digit BCD) Area of an						
before operation	S+1, S	+2	2 Integer of an angle. 0 to 9999.99 (4-digit BCD)						
	S to S-	+2	2 Unchanged						
	D, D+	1	Decimal frac (4-digit BCD)	•					
Contents	D+2		Integer of result. (2-digit BCD)				*		
after operation			A sign of res (2-digit BCD)		(H): Positi (H): Nega	ve (+) tive (-)]			
			Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		
			Positive (+)	0	0	0	1		
Flag			Negative (-)	0	1	0	0		
			More than +100 or less than -100. S to S+2 are	0	0	1	0		
			not BCD code.						

[Explar	nation	Instr	uction					
					S T R F-24	004001		
004001	F-24 TAN	009000	019000			009000 019000		
When the input condition of 004001 changes								

When the input condition of 004001 changes from OFF to ON, this instruction obtains the TAN of the 6-digit BCD contents of registers 009000 to 009002, and stores the result in registers 019000 to 019003.



- The above example shows the operation of TAN $98.03^{\circ} = -7.0884$.

* D to D+3 (Contents after calculation.)

- The calculation result will be rounded off below the 5th digit to the right of the decimal, area of result is —99.9999 to 99.9999.
- Unchanged when the contents of registers S to S+2 are not BCD.
- When operation of result is contents of S becomes more than 100.0000 or less than -100.0000, the PC turns ON an error flag, and does not operate.

F-25 ASIN

Executes trigonometric function (SIN⁻¹)

Sym	ibol	- F-25 ASIN S	D			[Explanation] Instruction S T R 004001 F -25		
Function		Determines the S BCD contents of and its result is s registers D to D+	registers tored in tl	S to S ne	S+3	004001 F-25 009000 019000 ASIN 009000 019000 019000 When the input condition of 004001 changes 009000 019000		
Opera	ation	SIN ⁻¹ (S to S+3) -	→ D to D+	-3		from OFF to ON, this instruction obtains the SIN ⁻¹ of the 8-digit BCD contents of registers 009000 to		
S		Use range C				009003, and stores the result in registers 019000 to 019003.		
D		Use range C				009003 009002 009001 009000		
Cond	ition	Rising edge of inpu	ıt signal (0	OFF to	ON)	100000000000000000000000000000000000000		
	S, S+	1 Decimal fraction of SIN ⁻¹ data. (4-digit BCD)				Sign [−]		
Contents before	S+2	Integer of SIN ⁻¹ data. (2-digit BCD)	Area of SIN ⁻¹ data is —1.0000 to 1.000					
operation	S+3	A sign of SIN ⁻¹ data. (2-digit BCD) (00(H): Positive (+) 80(H): Negative (-)	13 1.0000 10 1.0000.		1.0000.	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$		
	S to S+3	Unchanged						
	D	Decimal fraction of result. (2-digit BCD)			Sign [—]			
	D+1 D+2	(4-digit BCD)			*	The above example shows the operation of SIN ⁻¹ (-0.8735) \doteqdot -60.87° .		
Contents after operation	D+3	A sign of result. (2-digit BCD)						
		Result Ze 007 Positive (+)	ero Carry 357 007356 0 0 0 1	Error	Non-carry 007354 1 0	- The calculation result will be rounded off below the 5th digit to the right of the decimal, area of result is		
	Flag	S to S+3 are not BCD code. S to S+3 is bigger than +1 or smaller than -1.	0 0	1	0	 —90.00° to 90.00°. - Unchanged when the contents of registers S to S+3 are not BCD. 		



Executies trigonometric function (COS-1)

Symbol		F-26 ACOS S D		[Explanation] Instruction S T R 004001
Function		Determines the COS ⁻¹ of the 8 dig BCD contents of registers S to S-1 and its result is stored in the regist D to D+3 (8-digit BCD).	-3	F - 26 $F - 26$ 009000 019000 019000 When the input condition of 004001 changes from
Opera	ation	COS ⁻¹ (S to S+3) → D to D+3		OFF to ON, this instruction obtains the COS ⁻¹ of the 8-digit BCD contents of registers 009000 to
S	5	Use range C		09003, and stores the result in registers 019000 to 019003.
D)	Use range C		009003 009002 009001 009000
Cond	lition	Rising edge of input signal (OFF to 0	ON)	000000000000000000000000000000000000000
Contents before operation	5+2	Decimal fraction of COS ⁻¹ data. (4-digit BCD) Integer of COS ⁻¹ data. (2-digit BCD) A sign of COS ⁻¹ data. (2-digit BCD) A sign of COS ⁻¹ data. (00(H): Positive (+) 80(H): Negative (-)) (2-digit BCD)		Sign [+] Decimal point 019003 019002 019001 $0190000 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0$
	S to S-	-3 Unchanged Decimal fraction of result. (2-digit BCD)		Sign [+] Decimal point
	D+1, D-	+2 Integer of result. (4-digit BCD)	*	The above example shows the operation of COS^{-1} (0.5555) \Rightarrow 56.25°.
Contents after	D+3	A sign of result. (2-digit BCD) $\begin{bmatrix} 00(H): Positive (+)\\ 80(H): Negative (-) \end{bmatrix}$		* D to D+3 (Contents after calculation.)
operation	Flag	Zero Carry Error t Positive (+) 0 007357 007356 007355 Positive (+) 0 0 0 0 Negative (-) 0 1 0 0 S to S+3 are not BCD code. 0 0 1 0 S to S+3 is bigger than +1 or smaller than -1. 0 0 1 1	Non-carry 007354 1 0 0	 Area of result is 0 to 180.00° Unchanged when the contents of registers S to S+3 are not BCD.

F-27 ATAN

Executes trigonometric function (TAN⁻¹)

Sym	nbol	-F-27 ATAN S D		[Explanation] Instruction S T R 004001	
Function		Determines the TAN-1 of the 8-digit BCD contents of registers S to S+3 and its result is stored in the registe D to D+3 (8-digit BCD).	;	004001 F-27 009000 019000 ATAN 009000 019000 019000 When the input condition of 004001 changes from	
Opera	ation	TAN ⁻¹ (S to S+3) → D to D+3		OFF to ON, this instruction obtains the TAN ⁻¹ of the 8-digit BCD contents of registers 009000 to	
S	6	Use range C		009003, and stores the result in registers 019000 to 019003.	
C)	Use range C		009003 009002 009001 009000	
Cond	lition	Rising edge of input signal (OFF to O	N)	10000000100000000010011100110101	
Contents	S, S+ ⁻	(4-digit BCD)		Sign [-]	
before	S+2	TAN ⁻¹ data. (2-digit BCD) A size of TAN ⁻¹ data is -98.9999 to 98.9		\downarrow	
	S+3	A sign of TAN ⁻¹ data. (00(H): Positive (+) (80(H): Negative (-) (2-digit BCD)		019003 019002 019001 019000 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 1 0 0 0	
	S to S+	-3 Unchanged			
	D	Decimal fraction of result. (2-digit BCD)		Sign [-] Decimal point	
	D+1, D-	+2 Integer of result. (4-digit BCD)	*	The above example shows the operation of TAN ⁻¹ (-80.2735) \doteqdot -89.29° .	
Contents after	D+3	A sign of result. $\begin{bmatrix} 00(H): Positive (+) \\ 80(H): Negative (-) \end{bmatrix}$		* D to D+3 (Contents after calculation.)	
operation		Result Zero 007357 Carry 007356 Error 007355 No Positive (+) 0 0 0 0 Negative (-) 0 1 0 0	on-carry 07354 1 0	 Area of result is —89.42 to 89.42, the calculation result will be rounded off below the 3rd digit to the right of the decimal. 	
	Flag			- Unchanged when the contents of registers S to	
		S to S+3 is bigger than +99.0000 or smaller than -99.0000.	0	S+3 are not BCD.	

F-28 XY→

Exchanges the rectangular coordinate system (X, Y) data with polar coordinate system (γ , θ)

γ, θ (58309, -59.04°)

Ý

Sym	bol	$-\begin{bmatrix} F-28 \\ XY \rightarrow \end{bmatrix} S \qquad D$	[Explanation] Instruction S T R 004001 F -28
Func	tion	Exchanges the rectangular coordinat system (X, Y) of registers S to S+3 at S+4 to S+7 to polar coordinate system (γ , θ) and its result is in the registers D to D+3 and the registers D+4 to D+	e 004001 F-28 009000 019000 019000 m When the input condition of 004001 changes from 0FF to ON, the PC exchange the rectangular
Opera	ation	X (S to S+3), Y (S+4 to S+7) → γ (D to D+3), θ (D+4 to D+7)	coordinate system (X, Y) of registers 009000 to 009007 to polar coordinate system (γ , θ), and stores the result in registers 019000 to 019007.
S		Use range G	
D		Use range G	
Cond	ition	Rising edge of input signal (OFF to O	N) Sign [+] X data
	S to S+2	Integer of X data. (5-digit BCD) - Area of X data	009007 009006 009005 009004
Contents before	S+3	A sign of X data. (2-digit BCD) (00(H): Positive (+) 80(H): Negative (-)	
operation	S+4 to S+6	D Integer of Y data. (5-digit BCD) - Area of Y data	
	S+7	A sign of Y data. (00(H): Positive (+) 80(H): Negative (-) -999999 to 999	$\begin{array}{c} 99 \\ \hline \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0$
	S to S-	+7 Unchanged	γ data 019007 019006 019005 019004
	D to D+	+3 Integer of γ data. Unchanged when X, Y data isn't BCD code	019007 019006 019005 019004 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	D+4	Decimal fraction of θ data. (2-digit BCD) - Area of θ data is	$\begin{array}{c c} & & & \\ \hline & & \\ \hline & & \\ \hline & & \\ \hline \\ \hline$
Contents after operation	D+5, D+6	-179.99° to 180.00 Integer of θ data. (3-digit BCD)	
D+7		A sign of θ data. (2-digit BCD) (00(+): Positive (+) (80(+): Negative (-))	<u>30000</u> X
		Contents of register S 007357 007356 007355 0073	
	Flag	When X, Y data is BCD code. 0 0 0 0 When X, Y data 0	
		When X, Y data o 0 1 0	\neg $ $ \setminus
			-50000 \sim θ (58309 -59.04°)

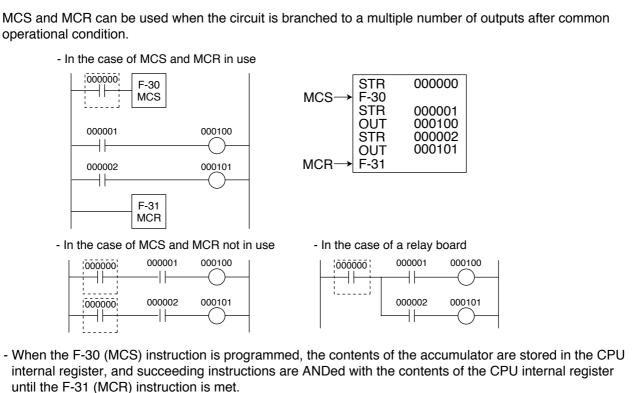


Exchanges the polar coordinate system (γ , θ) data with rectangular coordinate system (X, Y)

		rootangalai		, , - , - , - , - , - , - , -
Sym	nbol	F-29 →XY S D		[Explanation] Instruction STR 004001 F-29
Function		Exchanges the recta system (X, Y) of reg and S+4 to S+7 to p system (γ , θ) and registers D to D+3 a D+4 to D+7.	isters S to S+3 polar coordinate its result is in the	004001 F-29 009000 019000 009000 019000 When the input condition of 004001 changes from OFF to ON, the PC exchanges the polar coordinate system (γ, θ) of registers 009000 to 009007 to
Opera	ation	γ (S to S+3), θ (S+ \rightarrow X (D to D+3), Y (I		rectangular coordinate system (X, Y), and stores the result in registers 019000 to 019007.
S	;	Use range G		009003 009002 009001 009000
D)	Use range G		
Cond	lition	Rising edge of input	signal (OFF to ON)	γ data 009007 009006 009005 009004
	S to S+	+3 Integer of γ data. (8-digit BCD)		
Contents before	S+4	Decimal fraction of θ data. (2-digit BCD) $\left(\begin{array}{c} 00(H): \text{ Positive } (+)\\ 80(H): \text{ Negative } (-) \end{array}\right)$	- Area of γ data is 0 to 999999	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
operation	S+5, S+	-6 Integer of θ data. (3-digit BCD)	- Area of θ data is	10000000000000000001010100000011
	S+7	A sign of θ data. (2-digit BCD) $\begin{pmatrix} 00(H): Positive (+)\\ 80(H): Negative (-) \end{pmatrix}$	-179.99 to 180.00	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
	S to S+	7 Unchanged		0000000000000000011011010010000
	D to D+	2 Integer of X data. (5-digit BCD)		
	D+3	A sign of X data. (2-digit BCD) (00(H): Positive (+) 80(H): Negative (-))	- Unchanged when	Sign [+] Y data - The above example shows the operation of
Contents after	D+4 to D+6	D Integer of Y data. (5-digit BCD)	γ , θ data isn't BCD code	γ (6543), θ (145.67°) →X (-5403), Y (3690)
operation	D+7	A sign of Y data. (2-digit BCD) (00(H): Positive (+) 80(H): Negative (-)		γ=6543 Y=3690
		Contents of register S 00735	Carry Error Non-carry 7 007356 007355 007354]
	Flag	When γ , θ data 0	0 0 0	
		When γ , θ data 0 is not BCD	0 1 0	θ =145.67°
		a of S+4 to S+7 is le 180.01, it doesn't o		$\frac{1}{X=-5403}$

or more than 180.01, it doesn't operate.

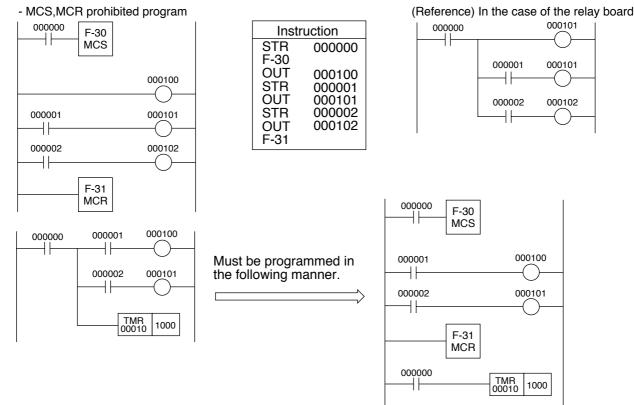
F-30
MCSSet master control
(Master Control Set)F-31
MCRReset master control
(Master Control Reset)



- The F-31 (MCR) instruction indicates the end of AND operation.

- It will help to simplify the program when the operational condition indicated in a block is complicated or many branches are set after the common operational condition.

- Do not directly connect the F-30 (MCS) derived bus line with OUT, TMR, and CNT instructions or application instruction.

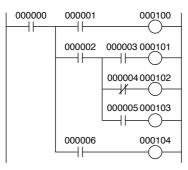


- The F-31 (MCR) instruction is an unconditional instruction.

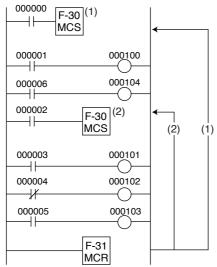
000010	E 21
	MCR

It prohibits the program like shown above.

- Another MCS may be used in between MCS and MCR.
 - In the case of the relay board



- In the case of MCS and MCR in use

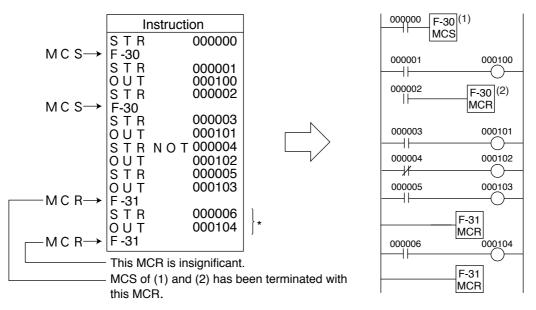


- The relay board ladder chart shown in left can be programmed in the following manner using MCS and MCR. However, there may be a need of changing the program sequence in the example (*).

Ir	nstructio	on	
STR		000000	
F -30			
STR		000001	
O U T S T R		000100	1
OUT		000008	
STR		000002)
F-30			
STR		000003	
OUT		000101	
	ΝΟΤ		
OUT		000102	
STR		000005	
F-31		000103	
יט ין			

- The F-31 (MCR) instruction indicates the termination of the preceding F-30 (MCS) instruction ; (1) and (2) in the example.

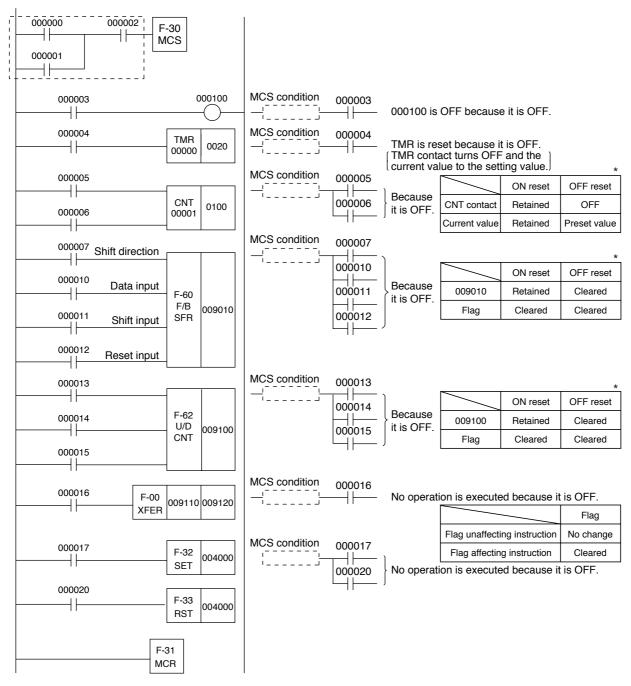
- The desired circuit would not be established if programmed in the following way.



- Although it is possible to insert MCS as many times as required between MCS and MCR (*), the range of MCS terminates with MCR marked with an asterisk (*).

When nesting like above shows, we use F-231 (MCRN) "Reset master control nesting" instruction instead of F-31 (MCR).

- When the MCS condition (enclosed with dotted line) is OFF instructions existing between MCS and MCR will be handled in the following manner.



* With a CNT, F-60, or F-62 instruction, it is possible to ON reset or OFF reset the reset condition using the system memory #0202. In the case of the OFF reset, reset is done by MCS.

Set coil

F-32 SET

Symbol	F-32 SET	OUT	[Explanation] $\begin{bmatrix} Instruction \\ S T R & 004000 \\ F - 32 \\ 000100 \end{bmatrix}$ When the set input of 004000 is ON, this instruction sets ON OUT000100. Once sets ON, OUT000100 remains active after the set input is set to OFF. When the set input of 004000 is OFF, this		
Function		es the OUT specified by F-32 ne set input is turned ON.			
Operation	Activate	e the OUT specified by F-32.			
OUT	Use ran	ge K			
Condition		et input is ON ited to OFF to ON change)	instruction does not affect the s	-	
Contents	OUT	ON	Set input ON 004000 OFF		
after operation	Flag Unchanged		OUT ON		

- When the F-32 (SET) instruction is contained in the F-30 (MCS) instruction, the OUT, once activated, remains active after the F-30 instruction has been executed.

- The F-32 (SET) instruction can control a specific OUT on more than 1 circuit.

- If the OUT specified by the F-32 (SET) instruction is within the latched relay area, it is kept the last state when the power is recovered from power failure. If the specified OUT is outside the latched relay area, it is reset at power recovery.

- If the OUT specified by the F-32 (SET) instruction is within the output hold area, it is kept the last state after the JW300's operation halts. If the specified OUT is outside the output hold area, it is reset when the JW300's operation halts. => See page "System memory #0232, #0233, #0252, and #0253."

- It is recommended that the F-32 (SET) instruction be used in conjunction with the F-33 instruction described on the next page.
- The F-32 (SET) and F-33 (RST) instructions placed between the F-30 (MCS) and F-31 (MCR) instructions are not operative if the operational condition of the F-30 (MCS) is OFF.

Resets coil

F-33 RST

Symbol	F-33 RST	OUT	[Explanation] Instruction S T R 004002		
Function		rates the OUT specified by F- ruction when the reset input is ON.	004002 Reset input F-33 RST 000110 F -33 When the reset input of 004002 is ON, this		
Operation	Deactivates the OUT specified by F-33.		instruction sets OFF OUT000110. Once sets OFF, OUT000110 remains inactive after the reset input		
OUT	Use ran	ge K	is set to OFF. When the reset input of 004002 is OFF, this instruction does not affect the state of		
Condition	When reset input is ON (not limited to OFF to ON change)		OUT000110.		
Contents after	OUT OFF		Reset input ON 004002 OFF OUT ON		
operation	Flag Unchanged		000110 OFF		

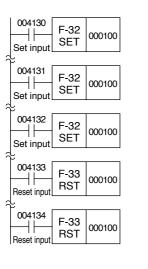
- It is the OUT specified by the F-33 (RST) instruction is within the output hold area, it is kept the last state after JW300 operation halts.

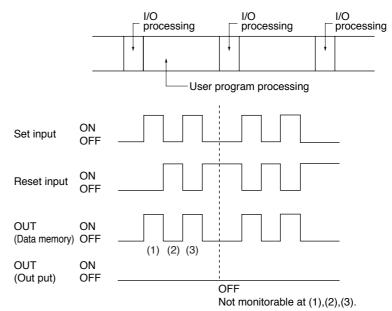
- If the specified OUT is outside the output hold area, it is reset when JW300 operation halts.

=> See page "System memory #0232, #0233, #0252, and #0253."

- The F-32 (SET) and F-33 (RST) instructions allow 1 OUT to be controlled under more than 1 condition.

- If the set and reset inputs are activated or deactivated more than 1 time in 1 scan cycle, the data memory used as the OUT is set and reset repeatedly in 1 scan cycle. In case of an output terminal of an output module, however, the PC outputs the OUT results before the I/O processing. (ON or OFF state).

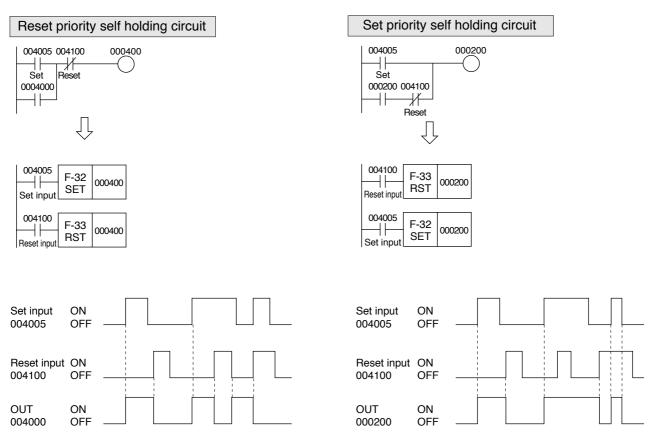




Even if the data memory turns ON/OFF several times during processing a user program, the JW300 can monitor a result just before the I/O processing.

11-34

- It is recommended that the F-32 (SET) and F-33 (RST) instruction be used in pair. It will simplify the self holding circuit.



F-34 TSET

Compares with current value of clock (specified relay set)

Symbol	F-3		[Explanation] Instruction S T R 004002 F -34		
Function	and n ₂ of the	res the constants n1 (hours) (minutes) with the current value clock and sets (turns ON) the ed BIT (relay) if they match.			
Operation	clock w	res the current value of the rith n1 and n2, and turns ON the the comparison result shows a	 specified time and the current value of the clock match, relay 000100 turns ON. Relay 000100 which is turned ON remains ON even if input condition 004002 turns OFF. If the current value of the clock and the specified time do not match, the state of relay 000100 does not change. Specified time ≷ Current value of clock → NOP 		
N1	Use rang	ge 00 to 23 (decimal)			
n2	Use rang	ge 00 to 59 (decimal)			
BIT	Use ran	ge K			
Condition		he input signal is ON ited to an OFF to ON change)	 Specified time=Current value of clock → Specified relay ON 		
Contents	n1 Unchanged				
after	n2 Unchanged				
operation	Flag	Unchanged			

- If the F-34 (TSET) instruction is within the F-30 (MCS) instruction, the relay which is turned ON remains ON even though the F-30 instruction turns OFF.
- Use of the F-34 (TSET) instruction can control 1 relay on multiple circuits.
- If the relay specified by the F-34 (TSET) instruction is within the latch specified area, it retains its state that existed before the power failure even after power is restored. If the specified Relay is outside the latch specified area, it is reset when power is recovered.
- If the relay specified by the F-34 (TSET) instruction is within the area which retains the output while the JW300 is stopped, it retains its state that existed before the stop when the JW300 is stopped. If the specified relay is outside the area which retains the output while the JW300 is stopped, it is reset when the JW300 is stopped.
- Use the F-34 (TSET) instruction as a pair with the F-35 (TRST) instruction on the next page.
- F-34 (TSET) and F-35 (TRST) instructions between the F-30 (MCS) and F-31 (MCR) instructions do not function when the operational condition for the F-30 (MCS) instruction is OFF.

F-35 TRST

Compares with current value of clock (specified relay reset)

Symbol	F-: TR	$r = n_1 n_2 B $	[Explanation] Instruction S T R 004003		
Function	and n2	ares the constants n1 (hours) (minutes) with the current of the clock and resets (turns he specified BIT (Relay) if they	$\begin{bmatrix} 004003 \\ \hline F-35 \\ TRST \\ 09 \\ 15 \\ 000110 \end{bmatrix} \begin{bmatrix} F-35 \\ 09 \\ 15 \\ 000110 \\ \hline 000110 \\ \hline 0000110 \\ \hline 000010 \\ \hline 0000010 \\ \hline 000000 \\ \hline 0000000 \\ \hline 000000 \\ \hline 0000000 \\ \hline 00000000$		
Operation	clock v	ares the current value of the vith n1 and n2, and resets the the comparison result shows h.	 clock match, relay 000110 turns OFF. The relay which is turned OFF remains OFF even if input condition 004003 turns OFF. If the current value of the clock and the specified time do not match, the state of relay 000110 does not change. Specified time ≥ Current value of clock 		
N1	Use ran	ge 00 to 23 (decimal)			
n2	Use ran	ge 00 to 59 (decimal)			
BIT	Use ran	ge K	 → NOP - Specified time=Current value of clock 		
Condition		he input signal is ON ited to an OFF to ON change)	→ Specified Relay OFF		
Contents	n1 Unchanged]		
after	n ₂	Unchanged			
operation	Flag	Unchanged			

- If the relay specified by the F-35 (TRST) instruction is within the latch specified area, it retains its state that existed before the power failure even after power is restored. If the specified relay is outside the latch specified area, it is reset when power is recovered.

- If the relay specified by the F-35 (TRST) instruction is within the area which retains the output while the JW300 is stopped, it retains its state that existed before the stop when the JW300 is stopped. If the specified relay is outside the area which retains the output while the JW300 is stopped, it is reset when the JW300 is stopped.

- Use the F-35 (TRST) instruction as a pair with the F-34 (TSET) instruction.

F-36 TADD

Adds time

Sym	bol	F-36 TADD	S1	S ₂	D		[Explanation] Instruction S T R 004001 F -36		
Function		Adds the contents of registers S_1 to S_{1+2} and the contents of S_2 to S_{2+2} for time as hours, minutes and seconds, and stores the results to D to D+2.					$ \begin{array}{c c} & & & & & \\ \hline & & & & \\ \hline & & & & \\ \hline & & & &$		
Opera	ation	(S1 to S1+	2) + (S2	to S ₂ +2) → (D t	to D+2)	registers 009100 to 009102 are added to the contents (6-digit BCD) of registers 009120 to		
St	1	Use range	θE				009122, and the results are stored to registers		
Sa	2	Use range	θE				009150 to 009152. - Ex.1		
D	1	Use range	θE				009102(hours) 009101(minutes) 009100(seconds) 0 0 0 0 0 1 1 1 0 0 1 0 0 0 1 1 0 0 0 0		
Condi	ition	Rising edg	e of inp	ut signa	l (OFF t	o ON)			
	S1 to S1+2	Uncha	inged				009122(hours) 009121(minutes) 009120(seconds) 0 0 0 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 1 0 1 0 0 0 38 minutes.		
	S2 to S2+2	Uncha	nged				28 seconds		
	D to D+2	Result					↓ 019152(hours) 019151(minutes) 019150(seconds) 18 hours,		
Contents after		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 1 1 minutes, 29 seconds		
operation		0	1	0	0	1			
	Flag	000001 to 235959	0	0	0	1	- Ex.2 000100 000101 000100 10 h		
		000000 (tomorrow)	1	1	0	0	LA.2 009102 009101 009100 13 hours, 1 3 3 6 3 7 36 minutes,		
		000000 or more	0	1	0	0	+ 37 seconds		
Excepting 0 0 1 0		009122 009121 009120 10 hours,							
- If 24 hours are exceeded and the time changes to the next day, carry flag (007356) turns ON.						1 0 5 4 1 1 54 minutes, 11 seconds 009152 009151 009150 0 hours, 0 0 3 0 4 8 30 minutes, 48 seconds Carry 1 (007356) 1 007356 1 1 1			

F-37 TSUB

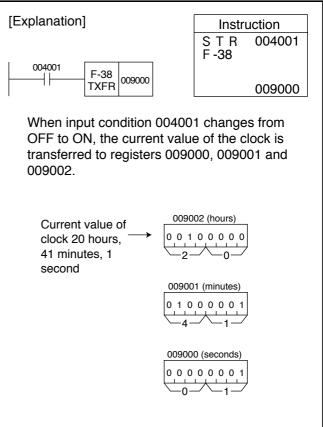
Subtracts time

Sym	bol	F-37 TSUB	S1	S ₂	D		[Explanation] Instruction S T R 004002 F -37		
Function		Subtracts the contents of registers S_2 to S_{2+2} from the contents of S_1 to S_{1+2} for time as hours, minutes and seconds, and stores the results to D to D+2.			nts of St minutes	to s and	004002 F-37 009100 009120 009150 009120 When input condition 004002 changes from OFF to ON, the contents (6-digit BCD) of 009100 009120		
Opera	ation	(S1 to S1+	2) + (S 2	to S2+2)	→ D to	D+2	registers 009100 to 009102 are subtracted from the contents (6-digit BCD) of registers 009120		
Sı		Use range	θE				to 009122, and the results are stored to		
Sa	2	Use range	θE				registers 009150 to 009152. -Ex.1_009012(hours)_009101(minutes)_009100(seconds)		
D		Use range	θE				009012(hours) 009101(minutes) 009100(seconds) 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1		
Condi	tion	Rising ed	ge of in	put sign	al (OFF	to ON)			
	S1 to S1+2	Uncha	Unchanged				0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 1 0 1		
	S2 to S2+2	Uncha	nged						
	D to D+2	Result					009152(hours) 009151(minutes) 009150(seconds) 18 hours,		
Contents after operation		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	0 0 0 0 0 1 1 0 0 1 0 0 0 1 0 0 0 1 1 0 0 1 1 0		
		0	1	0	0	1			
	Flag	000001 to 235959	0	0	0	1	-Ex.2 009102 009101 009100 13 hours,		
		Negative number	0	1	0	0	1 0 3 6 3 7 36 minutes, 		
		Excepting time	0	0	1	0	019122 019121 009120 2 1 5 4 1 1 10 hours,		
	If 0 hour is exceeded and the time changes to the previous day, carry flag (007356) turns ON.					$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			



Transfers current value of clock

Symbol	F-3 TXF		[Explan		
Function	clock (ł	Transfers the current value of the clock (hours, minutes, seconds) to registers D to D+2.			
Operation	Current	Current value of clock \rightarrow D to D+2			
D	Use ran	OFF trans 0090			
Condition	Condition Rising edge of input signal (OFF to ON)				
Contents	D to D+2	Clock data	Cı		
after operation	Flag	Unchanged	clc 41		
			se		

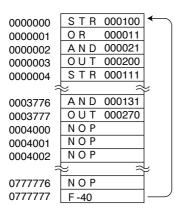


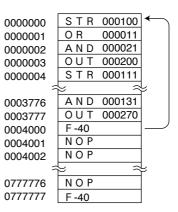


The F-40 instruction indicates the end of the program.

There is no need of writing the END instruction, except for the following cases, because it will be automatically set in the last address of the program memory when the memory is cleared after specifying the program memory capacity.

- (1) Saving scan time
- The scan time is a time the I/O processing time added with the user program execution time. The user program execution time is the total time required to execute all instructions from the program address 0000000 to the END instruction.
- The location of the END instruction automatically written after memory clear is the position of the END instruction, which is automatically written by clearing the program memory, will be 0777777 (256th word) of the main block if the control module is JW-362CU.
- Assume now if the last address is 0003777 (2048th word) when the ladder chart has written by the programmer, 0004000 to 0777776 are padded with NOP instruction with the END instruction in 0777777, so that it requires an unwanted time in order to execute NOP instructions.
- If F-40 is written in 0004000, it saves the processing time without executing those unwanted NOP instructions, and the control can proceed to a next scan cycle after termination of the user program.





(a) END (0777777) after mere memory clear

(b) F-40 (END) written in 0004000

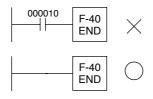
- If you are using multiple blocks, write F-40 to each block.

(2) To perform a partial program execution during trial run

By inserting the F-40 instruction at an end of a sequence block, you will be able to execute only the required portion of the program. If the result was successful, that F-40 may then be deleted.

- A multiple number of F-40 may exist when the END instruction is written in (1) and (2). In this case, the user program terminates at the first F-40. So, it would be necessary to check the location of the END instruction before going into the actual operation.

- The F-40 (END) instruction is unconditional, and it prohibits the following kind of programming.



- Although the F-40 instruction has the highest priority, it will be ignored if an END instruction exists between F-141 (JMP) and F-140 (LABL) or between F-142 (CALL) and F-140 (LABL) and that F-141 or F-142 was executed.

F-41 JCS

Set jump control (Jump Control Set)

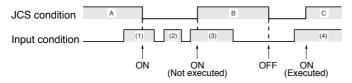


Reset jump Control (Jump Control Reset)

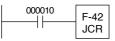
When the F-41 (JCS) condition is OFF, all instruction will not be executed, except when the F-42 (JCR) instruction that exists before the END instruction. Therefore, it does affect the contents of the data memory, even if there is an instruction that writes the result in the data memory, such as OUT, TMR, CNT, MD, and application instruction. And, it retains the state when JCS condition is ON.

	F-41 JCS			truction
000011	000101	-	→ STR F-41	000010
000012			S T R O U T S T R	000011 000101 000012
TMR00000	000102		TMR	00000 0030
000013	CNT 0010		STRT OUT STR STR CNT	M R 00000 000102 000013 000014 00001
 CNT00001 	000103		S T R C O U T S T R	0010 N T 00001 000103 000015
000015	F-63 009000 INC	-	F -63 F -42	009000
	JCR			
000010 (F-41 condition)				
000011 000101 (Output)				
000012 (Start input) 000102 (Output)	1.0 second	i ∠		
000013 (Counter input) 000014 (Reset input)	OFF			
CNT 00001 (Current value)	0001,0001		001 <mark>,0001</mark>	0001,0000
000103 (Output)	 			
000015				
009000	021 022	1		022 023

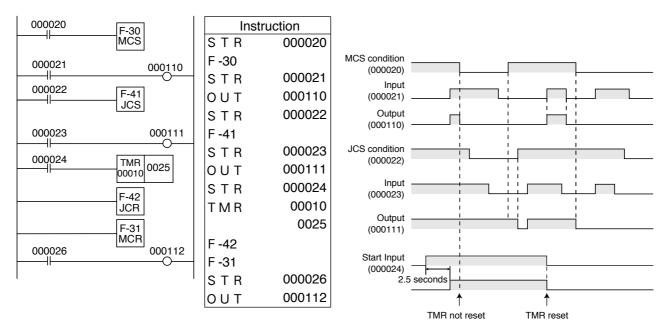
- Attention must be paid to the TMR internal clock (0.1 second clock), CNT counter input and application instruction input condition (that which the operation takes place at an OFF to ON transition in the input condition), and F-41 (JCS) condition ON/OFF timing.



- Operation takes place at a rising of (1) because the JCS condition is ON.
- Operation does not take place at a rising of (2) because the JCS condition is OFF.
- Operation does not take place at a rising of (3) because the JCS condition is OFF.
- The JCS condition turns to ON while (3) is ON, but operation dose not take place as it does not recognize that the input condition has changed from OFF to ON, because the input condition is ON with which the JCS condition of (a) changes from ON to OFF and the input condition is ON with which the JCS condition of (b) changes from OFF to ON.
- Operation does not take place at a rising of (4) because the JCS condition is OFF.
- The JCS condition becomes ON while (4) is ON. Operation takes place immediately after the JCS condition of (B) changes from OFF to ON because there was no change in that the input condition is OFF with which the JCS condition of (C) changes from ON to OFF and the input condition is ON with which the JCS condition changes from OFF to ON.
- The END instruction will be executed regardless whether the JCS condition be ON or OFF when there is a F-40 (END) instruction between F-41 (JCS) and F-42 (JCR). And the user program execution is terminated and the control proceeds to a next scan cycle.
- F-41 (JCS) and F-42 can be nested between F-30 (MCS) and F-31 (MCR). When the MCS condition becomes OFF, the instructions between JCS and JCR will be not executed regardless whether the JCS condition be ON or OFF. It is not possible to insert another F-40 and F-41 between F-41 (JCS) and F-42 (JCR). It will evoke "JCS ERROR" on the hand-held programmer during program check, if such a program was written. When nesting like above shows, we use F-242 (JCRN) "Reset jump control nesting" instruction instead of F-42 (JCR).
- The following kind programming is not possible as F-42 (JCR) is an unconditional instruction.

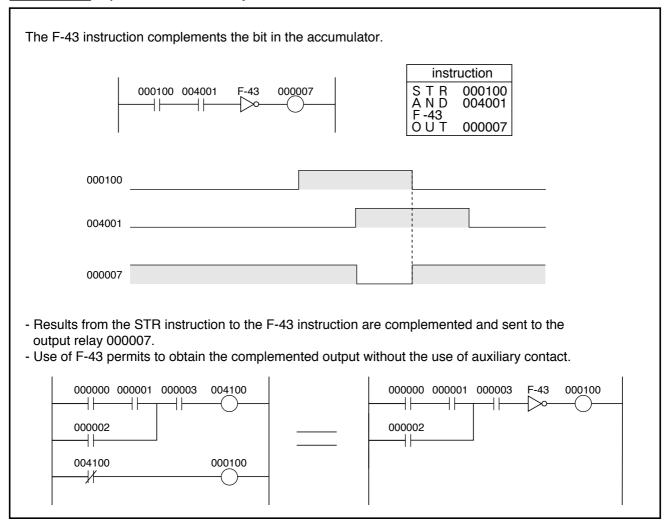


- To use an application instruction that operates at a rising edge of F-41 (JCS) and F-42 (JCR), the input condition must be different from F-41 (JCS). No operation will take place if the condition is same.
- F-41 (JCS) and F-42 (JCR) can be nested between F-30 (MCS) and F-31 (MCR). When the MCS condition becomes OFF, the instructions between JCS and JCR will be not executed regardless whether the JCS condition be ON or OFF.

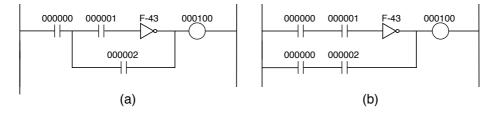


F-43 CPL

Complement bit (ComPLement)



- The F-43 instruction may be used for a single or a multiple number of contacts.
- Pay attention that the following programs (a) and (b) will not produce the same result because the F-43 instruction is the instruction that complement the contents of the accumulator.

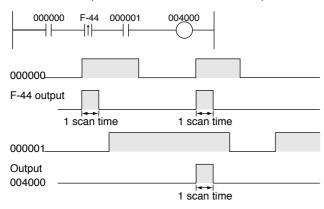




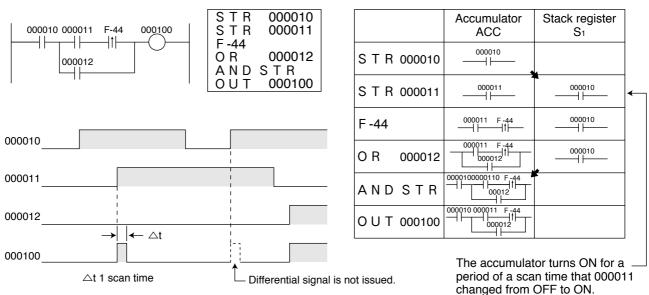
Differentiate at ON

The F-44 instruction sends 1 scan time pulse when the immediate state of the accumulator changes from OFF to ON. [Use example] Instruction 000000 000001 F-44 004000 000000 S ΤR $\neg \vdash$ -|t| ┥┝ AND 000001 F -44 004000 OUT 000000 000001 Output 004000 1scan time 1scan time

- Note that a different result is produced when the sequence of the F-44 program is changed in the above ladder chart. (Identical in the case of F-45.)



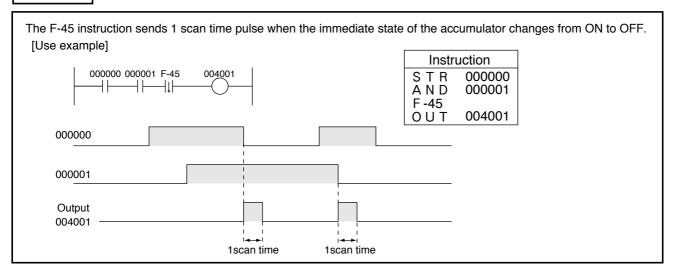
- F-44 condition may be a single or a multiple number of contacts.
- In the below example, a differential signal is not issued even if 000010 has changed from OFF to ON when 000011 is ON, because 000010 is ANDed by the AND STR instruction.



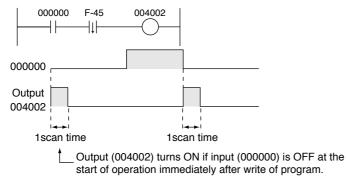
- The F-44 instruction scans only once even if inserted between an F-47 (level operation condition set) and a F-48 (level operation condition reset).

F-45 ⊣↓⊢

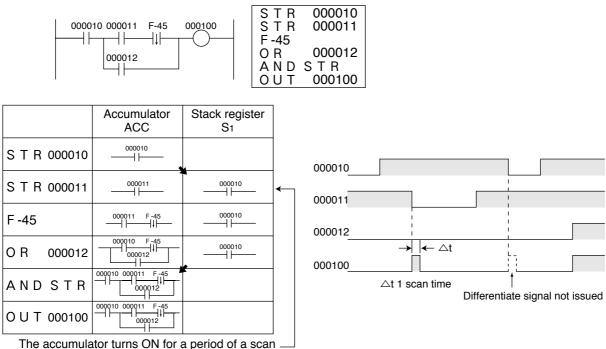
Differentiate at OFF



- You may use differential instruction (F-44, F-45) as many times as required in the program.

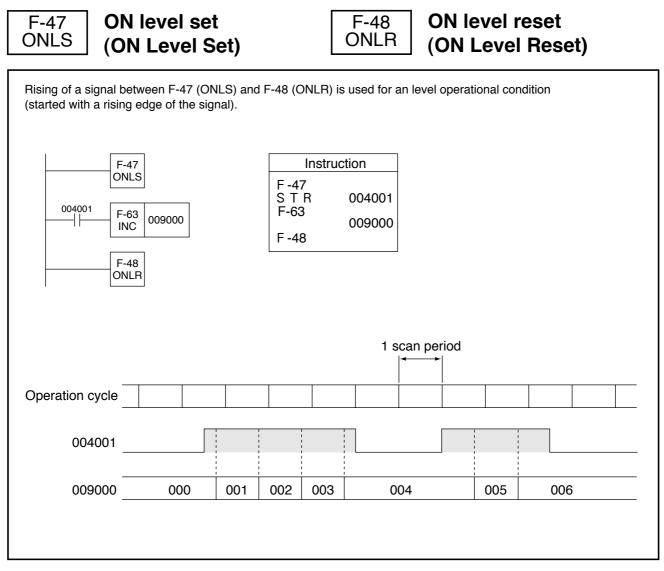


- In the below example, differentiate signal is not issued even if 000010 changed from ON to OFF when 000011 is ON, because 000010 is ANDed by the AND STR instruction.



The accumulator turns ON for a period of a scan time that 000011 changed from ON to OFF.

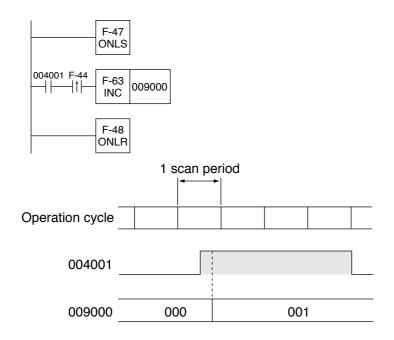
- An F-45 instruction can only be executed for a single scan time, if inserted between an F-47 instruction (level operation condition set) and an F-48 instruction (level operation condition reset).



- Another F-47 may not be nested between already existing F-47 (ONLS) and F-48 (ONLR).

- Only 1 scan will take place at a rising edge of 004001, if it contains a differentiate instruction (F-44, F-45).

[Example for F-44]

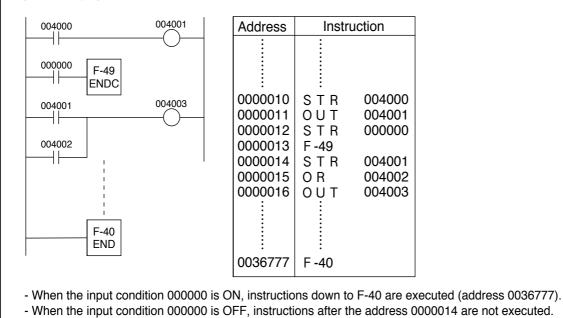


F-49 ENDC

Conditional end

When the condition of F-49 is OFF, the sequential operation terminates.

[Use example]



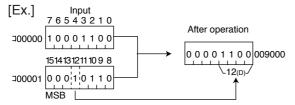
Chapter 12 Application instructions (F-50 to F-79d)

F-50 4→16	Dec	ode 4 to 16			
Symbol	F-50 4→16	S D	[Explanation]	Instruction S T R 004006 F -50	
Function	decode	ver 4 bits of the register S are d and stored in the 2-byte area egisters D and D+1.	004006 F-50 4→16 □00000 009350	⊐00000 009350	
Operation	S→D, ⊑)+1	When the input condition 004006 changes from OFF to ON, the lower 4 bits of 300000 are		
S	Use range A - Upper 4 bits of register S are ignored for calculation.		decoded and stored as 16-bit data in the 2-byte area consisting of registers 009350 and 009351. 76543210		
D	Use rar	ige B			
Condition	Rising e	dge of input signal (OFF to ON)		41312111098	
	S	Unchanged		0 0 0 0 0 0 009351	
Contents after	D	Result (0 to 7)	Only the bit position that corres 4 bits representing 0 to 15 is tu		
operation	D+1	Result (8 to 15)	other bits are turned OFF.		
	Flag	Unchanged			

F-51 16→4	Enco	ode 16 to 4			
Symbol	F-51 16→4	S D	[Explanation]	Instruction S T R 004001 F -51	
Function	1	byte data in registers S and e encoded and stored in the r D.	004001 F-51 16→4 ⊐00000 009000	⊐00000 009000	
Operation	S, S+1→D		When the input condition 004001 changes from OFF to ON, the 2-byte data consisting of registers ⊐00000 and ⊐00001 are encoded and stored in the register 009000.		
S	Use range B				
D	Use range A		Input 7 6 5 4 3 2 1 0		
Condition	ondition Rising edge of input signal (OFF to ON)			After operation	
Contents	S, S+1	Unchanged		0 0 0 0 1 1 1 0 009000	
after	D	Result	15141312111098 ⊐00001 0 1 0 0 0 0 0 0	_14 (D) ∕	
operation	Flag	Unchanged	MSB		

- After the operation, the upper 4 bits of D (009000 in the example) becomes 0 at all times.

- The MSB of the encoder input takes the highest priority.

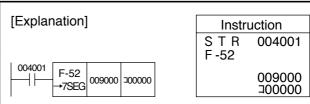




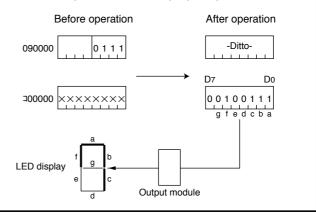
Decode to 7-segment data

Symbol	F-52 →7SEC	S D			
Function	The lower 4 bits data in register S is decoded into 7-segment display data.				
Operation	S→D				
S	Use range A				
D	Use ran	ge A *			
Condition	Rising e	dge of input signal (OFF to ON)			
Contents	S	Unchanged			
after	D	Result => "7-segment decoder chart"			
operation	Flag	Unchanged			

* Output data D0 to D6 to register D correspond to "a" to "g" of the 7-segment display. D7 is already "0."



When the input condition 004001 changes from OFF to ON, the lower 4 bits of the register 009000 are decoded into the 7-segment display data. See "7-segment decoder chart" relation between input data and display output.



7-segment decoder chart

Output data					
Input data	g fedcba	Display output			
00000000	00111111				
00000001	00000110				
00000010	01011011	2			
0000011	01001111				
00000100	01100110	Ч			
00000101	01101101	S			
00000110	01111101	6			
00000111	00100111				
00001000	01111111	8			
00001001	01101111	9			
00001010	01110111	8			
00001011	01111100	Ь			
00001100	00111001				
00001101	01011110	Ь			
00001110	01111001	8			
00001111	01110001	F			



Convert 4 digits BCD to 16 bits binary

Sym	bol	-F-53 →BIN S D]	[Explanation]	Instruction S T R 004001	
Function		4-digit BCD data of registers, S and S into a binary equiv 2 bytes in registers	+1, are converted valent and stored in			
Opera	ation	S, S+1→D, D+1		in registers ⊐01000 and ⊐019001 are converted into a binary equivalent and stored in 2-byte area		
s		Use range B		of registers 019000 and 019001.		
D		Use range B		Before operation \longrightarrow	After operation	
Cond	ition	Rising edge of input	signal (OFF to ON)	Tens Ones ⊐00000 1 0 0 1 0 1 1 0 0 0	19000 0 0 0 0 0 0 0 0 0	
	S, S+	1 Unchanged		9 6	$2^{7}2^{12}=4096^{2^{0}}$	
Contents	D	Result (0 to 255)	Unchanged when the contents of registers,	Thousands Hundreds	4000	
after operation	D+1	Result (256 to 9999)	S and S+1, are not BCD code.		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	
	Flag	S, S+1 Zero 007357 BCD code 0 Not BCD code 0	Carry Error Non-carry 007356 007355 007354 0 0 0 1 0 0		BIN(Binary)	

Note: When the F-53 instruction is used for programming, the F-03w instruction provides program monitor display.

Resembled instructions: F-03, F-03w, F-03d, F-153

Convert 16 bits binary to 6 digits BCD F-54 →BCD [Explanation] F-54 Instruction Symbol S D →BCD STR 004001 F -54 2-byte binary data in registers S, S+1 004001 F-54 ⊐00000 ⊐00000 009000 are converted into BCD 6 digits ►BCD 009000 Function equivalent and stored in 3 bytes in When the input condition 004001 changes from registers D, D+1, and D+2. OFF to ON, the 2-byte contents of binary data in Operation S, S+1→D, D+1, D+2 registers, 300000 and 300001, are converted into BCD 6 digits and stored in 3-byte area of registers S Use range B 009000 to 009002. Before operation - After operation D Use range E Tens Ones ⊐00000 0 0 0 0 0 0 0 0 0 009000 0 1 1 0 1 0 0 0 Condition Rising edge of input signal (OFF to ON) 8 27 20 6 Thousands Hundreds S, S+1 Unchanged ⊐00001 1 0 0 0 0 0 0 0 009001 00100111 Contents D Result (ones and tens) 2¹⁵ 2⁸ 2 7 after Binary D+1 Result (hundreds and thousands) Ten thousands operation 009002 00000011 D+2 Result (ten thousands) 0 3 BCD 6 digits Flag Unchanged 032768

Note: When the F-54 instruction is used for programming, the F-04w instruction provides program monitor display.

Resembled instructions: F-04, F-04w, F-04d, F-154



Swap upper 4 bits with lower 4 bits

Symbol	F-55 SWAP		[Explanation]	Instruction S T R 004001 F -55	
Function	Upper 4 bits are swapped with lower 4 bits of the register S and stored in the register D.		004001 F-55 009000 009001 009 SWAP 009000 009001 009		
Operation	S→D		When the input condition 004001 changes from OFF to ON, upper 4 bits are swapped with lower		
S	Use range A		4 bits of the register 009000 and its result is stored in the register 009001.		
D	Use range A		The contents of the register 00 remains unchanged.	9000, however,	
Condition	Rising edge of input signal (OFF to ON)		Before operation \longrightarrow	After operation	
Contents	S	Unchanged	009000 0 0 0 0 1 1 1 1	00001111	
after operation	D	Result			
oporation	Flag	Unchanged	009001	1 1 1 1 0 0 0 0	

Reference

- F-55 instruction becomes valid in the following case.

- With the F-52 (7 SEG decode) instruction, the lower 4 bits are decoded into the 7-segment display data. To display a multiple number of digits on the display, the upper 4 bits need to be swapped with the lower 4 bits, then F-52 should be executed.

Resembled instructions: F-175

F-56 NEG

> F-56w NEG

Complement of 10 of 1-byte data

Sym	Symbol - F-56 S D		[Explanation] Instruction S T R 004000			
Function		The 1 byte contents of the register S are assumed to be a 2-digit BCD code. Complement of 10 of that value is obtained and stored in the register D.	$ \begin{array}{c c} \hline 004000 \\ \hline F-53 \\ \hline NEG \end{array} \begin{array}{c} \hline 009000 \\ \hline 009100 \end{array} \end{array} \begin{array}{c} \hline F-56 \\ \hline 009000 \\ \hline 009100 \end{array} \end{array} \\ \hline When the input condition 004000 changes from \\ OFF to ON, the contents of the register 009000 \\ \hline are assumed to be 2-digit BCD code and \end{array} $			
Opera	ation	100−S→D	complement of 10 of that value is obtained and stored in the register 009100.			
S		Use range A	If the contents of the register 009000 are not a BCD code, the contents of 009100 are unchanged and the error flag is set 1. $\xrightarrow{-\text{Ex.1}} \xrightarrow{009000} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{100-58=42} \xrightarrow{009100} \xrightarrow{0091000} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} \xrightarrow{009100} 0$			
D	1	Use range A				
Cond	ition	Rising edge of input signal (OFF to ON)				
	S	Unchanged				
Contents after	D	Result - Unchanged when the contents of the register S is not a BCD code.	ZeroCarryErrorNon-carry0073570073560073550073540000			
operation Flag		Contents of register S Zero 007357 Carry 007356 Error 007355 Non-carry 007354 BCD code 0	$\begin{array}{c c} - Ex.2 & 009000 & 009100 \\ \hline 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ \hline - C & - & 3 & \end{array} \phantom{aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa$			
			ZeroCarryErrorNon-carry0073570073560073550073540010			

I

Complement of 10 of 1-word data

Sym	bol	-F-56w S D				[Explanation] Instruction S T R 004000 F -56w		
Function		The word contents of the registers S and S+1 are assumed to be a 4-digit BCD code. Complement of 10 of that value is obtained and stored in the registers D and D+1.				$\begin{array}{c c} & & & & & & & \\ \hline & & & & & \\ \hline & & & &$		
Opera	ation	10000− (S, S+1) →D, D+1				and complement of 10 of that value is obtained and stored in the registers 009100 and 009101. 009001 009000 $01001 009000$ $01001 009000$ $01001 009000$		
S		Use range B *						
D)	Use range B *						
Cond	ition	Rising edge of input signal (OFF to ON)			ON)			
	S, S+	1 Unchanged						
Contents	D	Lower digits of result		Unchanged when the contents of the				
after operation	D+1	Upper digits of result	registe not a E	register S, S+1 are not a BCD code.		10000-4908=5092		
	Flag	Contents of Zero register S, S+1 007357 BCD code		Error 007355		Zero Carry Error Non-carry 007357 007356 007355 007354 0 0 0 0		
		Not BCD code 0	0	1	0			

* Be sure to use even addresses for registers S and D. (Such as 009003 etc. are prohibited.)

F-56d NEG

Complement of 10 of 2-word data

Symbo	N	F-56d S D	[Explanation]		
Symbo	וע	NEG S D	S T R 004000		
Function		Obtain the 10's complement of the 8- digit BCD contents (2-word data) of	004000 F-53d 009000 F-56d 009000 NEG 009000 009100 004000 009100		
		registers S to S+3, and store the result in registers D to D+3.	When the input condition 004000 changes from		
			OFF to ON, the contents of the registers 009000		
Operation		10000000− (S to S+3) →D to D+3	to 009003 are assumed to be 8-digit BCD code and complement of 10 of that value is obtained		
S		Use range B *	and stored in the registers 009100 to 009103.		
D		Use range B *	009003 009002 009001 009000 0 1 0 0 1 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0		
Conditio	on	Rising edge of input signal (OFF to ON)	\downarrow		
	S to S+3	LUnchangeg	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$		
Contents after	D to D+3	- Unaffected if the contents of registers S to	10000000-49084908=50915092		
operation	Fla	Contents of registers S to S+3 Zero 007357 Carry 007356 Error 007355 Non-carry 007354 BCD code 0<	Zero Carry Error Non-carry 007357 007356 007355 007354 0 0 0 0		

 * Be sure to use even addresses for registers S and D.

(Odd addresses such as 009003 etc. are prohibited to use.)

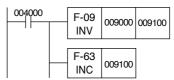


Complement of 2 of 1-byte data

Symbol	F-57 2NEG		[Explanation] Instructio S T R 004 F -57				
Function	comple	it contents of the register S are mented to 2 and its result is n the register D.	004000 F-57 009000 009100 When the input condition 0040	009000 009100			
Operation	0—S→	D	OFF to ON, the 8-bit contents of the register 009000 are complemented of 2 and stored in the register 009100.				
S	Use rar	ge A					
D	Use ran	ge A	009000	009100			
Condition	Rising e	dge of input signal (OFF to ON)		0010100			
Contents	S	Unchanged					
after	D	Result					
operation	Flag Unchanged						

- How to obtain the complement of 2.
 - (1) Complement all bits; 0 to 1 and 1 to 0.
- (2) Add 1 to the above result.

- Operation similar to the following instructions will take place for the above.



F-57w	
2NEG	

Complement of 2 of 1-word data

Symbol	-F-57w S D		[Explanation]	Instruction STR 004000		
Function	and S+	rd contents of the registers S 1 are complemented to 2 and n the registers D, D+1.	004000 F-57w 2NEG 009000 009100	F -57w 009000 009100		
Operation	0— (S,	S+1) →D, D+1	When the input condition 004000 changes from OFF to ON, the 16-bit contents of the registers 009000 and 009001 are complemented to 2 and stored in the registers 009100 and 009101.			
S	Use rar	nge B				
D	Use rar	nge B				
Condition	Rising e	dge of input signal (OFF to ON)				
	S, S+1	Unchanged				
Contents after	D	Lower digits of result				
operation	D+1	Upper digits of result	009101 009100 01110000110011			
	Flag	Unchanged				

- Operation similar to the following instructions will take place for the above.

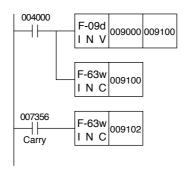




Complement of 2 of 2-word data

Symbol	F-57d 2NEG		[Explanation]	Instruction S T R 004000		
Function	content to S+3,	the 2 complement of the s (2-word data) of registers S and store the result in s D to D+3.	004000 F-57d 009000 009100 2NEG 009000 009100 004000 When the input condition of 004 004000 004000 004000	F -57d 009000 009100 4000 changes		
Operation	0— (S t	o S+3) →D to D+3	from OFF to ON, this instruction obtains the complement of 2 of the 32-bit content of registers			
S	Use rar	nge C *	009000 to 009003, and stores the result in registers 009100 to 009103.			
D	Use rar	nge C *				
Condition	Rising ec	lge of input signal (OFF to ON)	009003 009002 009001 00900			
	S to S+3	Unchanged	100011110001011011000	1 1 1 1 0 0 1 0 1 1 0 1		
Contents after	D	Lower digits of result	009103 009102 009	101 009100		
operation	D to D+3	Upper digits of result	0 1 1 1 0 0 0 0 1 1 0 1 0 0 1 0 0 1 1 1			
	Flag	Unchanged				

* Be sure to use even addresses for registers S and D. - Operation similar to the following instructions will take place for the above.



F-58 Σ BIT

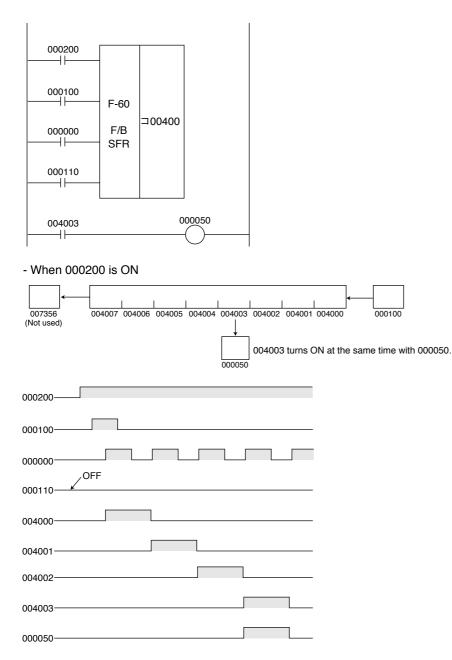
Total ON bits

Symbol	F-58 Σ BIT	n S D	[Explanation]	Instruction S T R 004002 F -58		
Function	having t	e bits in the "n" byte register he register S at its top are n the register D.	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	4 700006 009000		
Operation	ON bits-	→D	When the input condition 004002 changed from OFF to ON, the total number of active bits in a 4- byte register heralded by the register $\exists 00006$ is stored in the register 009000. $\exists 00006 \ 1 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1$			
n	Use ran	ge 0 to 7 (8 bytes for 0)				
S	Use ran	ge A				
D	Use ran	ge A				
Condition	Rising e	dge of input signal (OFF to ON)				
Contents	S, S+1,… S+n—1	Unchanged				
after operation	D	Result] - Among 32 bits in ⊐00006 to ⊐00	011, 13 bits are ON.		
operation	Flag	Unchanged				

F-60 SFR		s register bi- ward/Backw						
Symbol	(1) (2) (3) (4)	F/B D (2) Data input (3) Shift input						
Function		t data in the register i input (1).	D is shifted to u	pper or	lower bit p	ositions acc	ording to the shift	
Operation	- When th	ne shift direction input	is ON: Carry 007356 ←	- MSB	Regist	er D	LSB	
Operation	- When th	ne shift direction input	is OFF: Data input -	→ MSB	Regist	er D	LSB → Carry 007356	
D	Use ranç							
Condition	When the	e reset input (4) is OFF	, bits are shifted at	a rising	g edge (OFF	to ON) of the	ə shift input (3).	
	D	 Result is produced All bits are turned (
Contents after		Reset input (4)	Zero 007357	0	Carry 07356	Error 007355	Non-carry 007354	
operation	Flag	OFF	0 or 1	0	or 1	0	1 or 0	
		ON	0		0		0	
Shiilinduli	F/B 009000 SFR	S T R 000100 S T R 000000 S T R 004013 F-60 009000	007356 MSB - When 004012 is OFF 000100 MSB				00100 07356 Carry	
Inp	out condition	009000 (before operation 7 6 5 4 3 2 1	009000 i) (after operations) 0 7 6 5 4 3 2	tion)	Zero 007357	Carry 007356	Non-carry 007354	
	04012 0	0000000	0000000	000	•	0	•	
	00100 ○ 00000 ∫	$\bullet \circ \circ \circ \bullet \circ \bullet$			0	0		
	04013 O				<u> </u>	•	0	
	04012 •	0000000			•	0	•	
	00100 0	0 • 0 • 0 0 0			0	0		
00	₫ 00000	$\bullet \circ \circ \circ \bullet \circ \circ$	0000000	000	0	•	0	
	04013 0	• 0 0 0 0 0 0	000000	0	•	•	0	
	04012 ○ 00100 ●	• • • • • • •		$\circ \bullet \circ$	0	0	•	
	00000 ∫ 04013 ○	0000000	• • • • • • • •	000	0	•	0	
00	04012 • 00100 •	0 • 0 0 0 0 0	0 • 0 0 0 0 0		0	0	•	
	00000 ∫ 04013 ○	• • • • • • • • • • • • • • • • • • • •	0000000	$\circ \circ \bullet$	0	•	0	
	04013 •	000000	000000	000	0	0	0	
- By re	y setting the set with OF	07355) is OFF at all ⁻ e reset condition in t FF. structions: F-60w, F-	he system memo		202) for res	○ OFF et input (4),	● ON , it permits to	

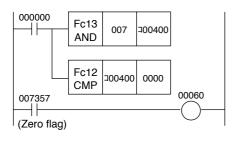
Reference

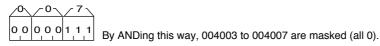
When ¬XXXXX is assigned for D, it allows to constitute a "n" bit (n<8) shift register.



- Data are shifted in 004004 to 004007.

- When all of 004000 to 004007 are 0, the zero flag turns to 1. The following program must be used to check that 004000 to 004002 are 0.





F-60w SFR		Shift register bi-directionally (1 word) (Forward/Backward Shift Register)							
Symbol	(2)	(1)(1) Shift direction indication input(2)F-60w(3)SFR(4)(2) Data input(3) Shift input(4) Reset input							
Function		bit contents of the regisition indication input (1		ifted to upper or lov	wer bit positions ac	ccording to			
	- When t	he shift direction indica	tion input (1) is ON	1:					
	Carry 007356	← MSB Re	gister D+1	Reg	ister D	LSB 🗲 Data input			
Operation	- When t	he shift direction indica	tion input (1) is OF	F:					
	Data input	t → MSB Re	gister D+1	Reg	ister D	LSB> Carry 007356			
D	Use ran	Use range B - Be sure to use even addresses for register D. (Odd addresses such as 019003 etc. are prohibited to use.)							
Condition	When th	e reset input (4) is OFF	, bits are shifted a	t a rising edge (OF	F to ON) of the shi	ift input (3).			
			vhen the reset input (4) is OFF. FF when the reset input (4) is ON.						
Contents after		Reset input (4)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354			
operation	Flag	OFF	0 or 1	0 or 1	0	1 or 0			
		ON	0	0	0	0			
[Explanation 000000 000001 F-60w 000002 SFR 000003	[Explanation] 000000 000001 000001 000002 Instruction S T R 000000 S T R 000001 S T R 000001 S T R 000001 S T R 000002 - Input conditions 000000 (1) ONShifts towards MSB 000001 (2) ONData input ON 000002 (3) OFF→ONShift direction 000003 (4) OFFNo reset function								
condition. Before operat After operati Zero	Before operation 009001 009000 $1 0 0 1 0 1 1 0$ $1 0 1 0 0 1 0 0$ 009001 009000 After operation $00 0 1 0 0 1 0 0 1$ $2ero$ $Carry$ Error 007357 007355 007354								

- By setting the reset input (4) in the system memory (#0202), it permits to reset with OFF. Resembled instructions: F-60, F-60d, F-160, Fc160

F-60d SFR		register bi-c ward/Backwa			;)				
Symbol	(1) (2) (3) (4)	(2) — F-60d (3) — F F R D (2) Data input (3) Shift input							
Function	1	32-bit contents of regi indication input (1).	sters D to D+3 tow	ard the MSB or LS	B according to the	shift			
	- When t	he shift direction indica	ation input (1) is Ol	N :	LSB				
Operation	Carry 007356	Register D+3	Register D+2 Re	egister D+1 Regi	ster D				
Operation	- When t	he shift direction indica	ation input (1) is Ol	FF :	LSB				
	Data inpu	t → Register D+3	Register D+2 Re	egister D+1 Regis	ster D → Carry 007356				
D	Use rar			ses for register D 3 etc. are prohibite					
Condition	When th	e reset input (4) is OFI	F, shift occurs at th	e rising edge (OFF	to ON) of the shift	clock input (3).			
Contents	D to D+3	 Result of operation All bits OFF when it 		vhen reset input (4) is OFF. set input (4) is ON.					
after		Reset input (4)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354			
operation	Flag	OFF	0 or 1	0 or 1	. 0 -	1 or 0			
		ON	0	0		0			
ON 0 0 0 [Explanation] Instruction Instruction Input conditions 000000 STR 000000 000000 (1) ON On Data input ON 000000 STR 0000000 STR 000000 (2) ON Data input ON 000000 STR 0000000 SOUT On ON									

- By setting the reset condition in the system memory (#0202), it permits to reset input (4) with OFF. Resembled instructions: F-60, F-60w, F-160, Fc160

F-6 ASF			gister asynd hronous Sh							
Sym	nbol	(1) — F-61 (2) — ASFR	2)ASFR D (2) Shift input							
Fund	ction		n the register D-1 cording to the shift			1) OFF) are sh	ifted to the			
Opera	ation	D−1→D - The conte	nift direction indication ents of the shift regonemist not take place	ister (D-1 or D	D+1→D)+1) are cleared	upon executio				
0)	Use range B								
Cond	lition	When the co	ntents of the regist the OFF to ON cl		done with the C	ON state of the	shift input (2)			
$\overline{}$			(1) ON		OFF		N/OFF			
		Before operation	After operation	Before operation	After operation	Before operation	After operation			
	D-1	D1	0	D1	Same as left	D1	Same as left			
Contents of	D	0	D1	0	D2	Other than 0	Same as left			
register	D+1	D2	Same as left	D2	0	D2	Same as left			
	Non-car 007354		(D1=0) (D1≠0)		2=0) 2≠0)		1			
	Error 007355			0		0				
Flag *	Carry 007356		(D1=0) (D1≠0)	,	0 (D2=0) 1 (D2≠0)		0			
	Zero 007357	7	0		0		0			
Shift di 004 Shift 004 - Shift of 09	007357 0									
со	Input Indition		Before operation 6 5 4 3 2 1 0	After operation 7 6 5 4 3 2		Carry 007356	Non-carry 007354			
	1014 ● 1015 ●	009020 0	$\bigcirc \bigcirc $		• 0 0	0	•			
	014 ● 015 ●	009017 O 009020 O	$\begin{array}{c} \circ \circ \circ \circ \bullet \circ \bullet \circ \\ \circ \circ \circ \circ \circ \circ \circ \circ \circ \\ \bullet \circ \circ \circ \circ$	0 0 0 0 0 0 0 0 0 0 0 ● 0	0 0 • 0 0	•	0			
	1014 ○ 1015 ●	009020 0	 ○ ○ ○ ● ○ ● ○ ○ ○ ○ ○ ○ ○ ○ ○ ● ○ ○ ○ ○ ○ ○ ○ ○ 	$\bullet \bullet \circ \circ \circ \circ$	00 0	•	0			
	1014 ○ 1015 ●	009020 0	 ○ ○ ○ ● ○ ● ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ 	000000	00 0	0	•			
- Th	ne error f	lag (007355) wi	II be OFF at all times	з Э.		○ OFF	• ON			

★ The carry flag (007356) turns ON only when data other than 0 are shifted from D−1 or D+1. Resembled instructions: F-61w, F-61d, F-161, F-161w, F-161d

F-61 ASF	w R	Shift regis (Asynchr	ter async onous Shi	-									
Sym	bol	(1) — F-61w (2) — ASFR	D (1) Shift (2) Shift	direction indication	n input								
Funct	tion	1 word in the real shifted to the real											
Opera	tion	D+3) will be), D+1 eration is execu	ted, the conten	D+2, D+3→D, I ts of shifted reg	D+1 isters (D-2 to D∙	1 or D+2 to						
D		Use range C (Be sure to use even addresses for register D.)											
Condi	tion	When the contents of the registers D, D+1 are 0, shift is done with the ON state of the shift input (2). (not limited to the OFF to ON change)											
		(1)	ON	(1)	OFF	(1) ON	N/OFF						
		Before operation	After operation	Before operation	After operation	Before operation	After operation						
Contents	D—2, D—1	D1	0	D1	Same as left	D1	Same as left						
of register	D, D+	1 0	D1	0	D2	Other than 0	Same as left						
_	D+2, D+3	D2	Same as left	D2	0 2=0)	D2	Same as left						
	Non-cari 007354		,		2—0) 2≠0)		1						
Flag	Error 007355				0		0						
, iag	Carry 007356	0 (D1 1 (D1			2=0) 2≠0)		0						
	Zero 007357	, ()		0		0						
- ·	000 001 001 F-6 ASF	1w 009004	nstruction R 000000 R 000001 61w 009004										
00	00000 (1) ON Shifts from 009002 and 009003. 00001 (2) ON Shift direction Contents of 009004 and 009005 are 0000(H).												
Control of colspan="2">Control of colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2">Colspan="2"Col													
Only	/ the ca	rry flag (007356)	is set ON.										

Resembled instructions: F-61, F-61d, F-161, F-161w, F-161d

F-6 ⁻ ASF		•	ster async onous Shi	-		;)							
Sym	bol	(1) F-61d ASFR D (1) Shift direction indication input (2) Shift input Shift the contents (2-words data) of registers D-4 to D-1 (when input (1) is ON) or											
Func	tion		D+7 (when inp										
Opera	ation	 D+7) will be cleared. When the contents of registers D to D+3 are other than 0, the JW300 does not operate. 											
D		Use range G (E	Be sure to use ev	ven addresses f	or register D.)								
Cond	ition	If the contents of (not limited to OF	of registers D to F to ON change)	D+3 are 0, shift	is done with the	e ON state of th	e shift input (2)						
			ON	(1) Before	OFF	(1) ON Before	I/OFF After						
		Before operation	After operation	operation	After operation	operation	operation						
Contents	D-4 to D-1	D D1	0	D1	Same as left.	D1	Same as left.						
of register	D to D+3	0	D1	0	D2	Other than 0	Same as left.						
regiotor	D+4 to D+7	D2	Same as left.	D2	0	D2	Same as left.						
	Non-carr (007354	· · ·	1=0) 1≠0)		2=0) 2≠0)		1						
	Error (007355)	0		0	(0						
Flag	Carry (007356	· · ·	1=0) 1≠0)	0 (D2 1 (D2	2=0) 2≠0)	()						
	Zero (007357)	0)	()						
0000	00 01 F-61(ASFF 000 (1) (001 (2) (1 009004 F S DN DN	Instruction T R 000000 T R 000001 -61d 009004 Shift from 0090 Shift direction										
The fol	Contents of 009004 to 009007 are 0000000(H). The following result will be produced for the above input condition.												
input condition. Before operation $009000 \\ 009000 \\ 009000 \\ 5, 6, 7, 8 \\ 009004 \\ 0, 0, 0, 0 \\ 0, 0, 0, 0 \\ 0, 0, 0, 0 \\ 0, 0, 0, 0 \\ 009002 \\ 0, 0, 0, 0 \\ 009002 \\ 009002 \\ 009003 \\ 1, 2, 3, 4 \\ 009004 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 009005 \\ 5, 6, 7, 8 \\ 009005 \\ 000005 \\ 000$													

Resembled instructions: F-61, F-61w, F-161, F-161w, F-161d

F-6 U/D			git BC Down		-		ı co	ounter			
Sym	bol (1) — F-6 2) — U/D 3) — U/D		(2) Co	/down o unter ir set inpu	put	n input				
Func	tion	2 digits) subtract	are adde are adde ed ((1) C lown cou	ed ((1)FF), a) ON) accord	or ding to	C	Up/down direction 000000 F-62 Instruction Counter input U/DC 009000 S T R 004010 004011 F-62 009000 S T R 004001			
Opera	ation	input (- When	the up/do 1) is ON: the up/do 1) is OFF	+ (D) wn di	⊦1→D irectio	n		When the reset input 0004011 is OFF, the counter is enabled, if set to ON reset mode. When the up/down counter direction input 004010 is ON, it acts as an increment counter.			
D		Use ran	ge A					When OFF, it acts as a decrement counter. If the			
Cond	ition		ng of the ON) whe			• • •		contents of 009000 were other than the BCD code, the error flag (007355) is turned ON and no counter operation takes place. (7F in the example.)			
	D	whe - Res	ult (BCD n the res ult (BCD n the res	et inp code)	ut (3)) is co	is OF ntaine	F. ed	Up/down direction (004010) Up Down			
		Up/down direction input (1)	Result	Zero 007357	Carry 007356		Non-carry 007354				
			99+1 →00	1	1	0	0	Register (009000) 97 98 98 99 99 00 00 01 01 00 00 99 (7F7F7F00 00			
Contents after		ON	00 to 98+1 →01 to 99	0	0	0	1	Non-carry flag			
operation	Flag		Not BCD code	0	0	1	0	Error flag (007355)			
			00−1 →99	0	1	0	0	Carry flag			
		OFF	01−1 →00	1	0	0	1	Zero flag			
			02 to 99−1 →01 to 98	0	0	0	1	(007357) → I ← 1 scan time, max. Valid until a flag affecting instruction is met in the program.			
			Not BCD code	0	0	1	0	-			
		Reset ir	nput (3) ON	0	0	0	0				

- If the contents of D is number other than BCD code, the error flag (007355) is set ON and no addition will be done. (D remain unchanged.)

- By setting the reset condition in the system memory (#0202), it is possible to reset the reset input (3) with OFF.

Resembled instructions: F-62w, F-62d, F-65, F-65w, F-65d, F-66, F-66w

F-62 U/D			jit BC Down				ı co	unter			
Syml	bol	(1) (2) (3) F-62 U D0 (3)		(2) Co	/down o unter in set inpu	iput	n input	[Explanation]			
Funct	tion	The 4-digit BCD contents of the register D, D+1 are added ((1) ON) or subtracted((1) OFF), according to the up/down counter direction input (1).						000002 F-62w 019000 S T R 000001 S T R S T R 000002 S T R S T R 000002 S T R O00003 S T R 000003 S T R 0000003 S T R 000000			
Opera	ation	 When the up/down direction input (1) is ON: ⟨D, D+1⟩+1→D, D+1 When the up/down direction input (1) is OFF: ⟨D, D+1⟩−1→D, D+1 						When the reset input 000003 is OFF, the counter becomes ready to count, if set to ON reset mode. When the up/down counter direction input 000001 is ON, it acts as an increment counter. When			
D		Use range B - Be sure to use even addresses for register D.						OFF, it acts as a decrement counter. If the contents of the register 019000 or 019001 are other than the BCD code, the error flag is turned ON and no counter operation takes place.			
Condi	ition	At a rising of the counter input (2) (OFF to ON) when the reset input (3) is OFF									
	D	Lower of resu	2 digits It		its are			Up/down direction DOWN (000001) UP DOWN			
-	D+1	Upper of resu	2 digits It	ON.	eset in	put (5) 15	Counter input			
		Up/down direction input (1)	Result	Zero 007357	Carry 007356		Non-carry 007354	Reset input (000003) Image: Constraint of the second			
			9999+1	1	1	0	0	Non-carry flag (007354)			
Contents after operation		ON	0000 to 9998+1	0	0	0	1	Error flag (007355)			
operation	Flag		Not BCD code	0	0	1	0	Carry flag (007356)			
			0000-1	0	1	0	0	Zero flag (007357) → I ←			
		OFF	0001-1 0002 to	1	0	0	1	1 scan time, max.			
			9999-1 Not BCD	0	0	0	1 0				
		Reset in	code put (3) ON	0	0	0	0				

- By setting the reset condition in the system memory (#0202), it is possible to reset input (3) with OFF. Resembled instructions: F-62, F-62d, F-65, F-65w, F-65d, F-66w

F-62d U/DC

8-digit BCD up/down counter (Up/Down Counter)

		V = I =	<u> </u>			/		
Sym	bol	(1) (2) (3) U/D		(2) Co	/down o unter in set inpu	put	n input	[Explanation]
Func	tion	Increments (when input (1) is ON) or decrements (when input (1) is OFF) the contents of registers D to D+3 (BCD 8 digits) according to the up/down input (1) state.						000002 F-62d 019000 S T R 000001 S T R 000002 S T R 0000003 F-62d 019000 S T R 000000 S T R 0000003 F-62d 019000 S T R 0000003 S T R <t< td=""></t<>
Opera	ation	 When the up/down direction input (1) is ON: (D to D+3)+1 → D to D+3. When the up/down direction input (1) is OFF: (D to D+3)-1 → D to D+3. 						When the reset input 000003 is OFF, the counter becomes ready to count, if set to ON reset mode. When the up/down counter direction input 000001 is ON, it acts as an increment counter. When OFF,
D		Use range C - Be sure to use even addresses for register D.						it acts as a decrement counter. If the contents of the register 019000 to 019003 are other than the BCD code, the error flag is turned ON and no counter operation takes place.
Cond	ition	Reset input (3) is OFF, and at the rising edge (OFF to ON) of the counter input (2).						
	D to D	(8 dig			rned C It (3) is			Up/down direction (000001) UP DOWN Counter input
		UP/ DOWN input (1)	Result	Zero 007357	Carry 007356		Non-carry 007354	(000002) Reset input
			999999999+1	1	1	0	0	(000003) Registers
		ON	00000000 to 99999998+1	0	0	0	1	019000 to 9999 9999 9999 0 1 0 ((9999999A 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Contents after operation			Not BCD code	0	0	1	0	Non-carry flag [] [] [] [] [_] [
	Flag		0000000-1	0	1	0	0	Error flag (007355)
		OFF	00000001-1	1	0	0	1	Carry flag (007356)
			00000002 to 999999999-1	0	0	0	1	Zero flag (007357)
			Not BCD code	0	0	1	0	1 scan time, max.
		When re is ON	eset input (3)	0	0	0	0	

- If the contents of registers D to D+3 are not BCD codes, the counter remains inoperative, and the error flag (007355) is set (the contents of D to D+3 are unaffected).

- By setting the reset condition in the system memory (#0202), it is possible to reset input (3) with OFF. Resembled instructions: F-62, F-62w, F-65, F-65w, F-65d, F-66w

Increments counter (1 byte) (INCrement)

Symbol		— F-63 D		[Explanati	on]			
Function		The contents (data) are incre		nary		F-63 INC 009		
Operation	n	$\langle D \rangle$ +1 \rightarrow D					OFF to C	e input co DN, the JV
D		Use range A					register	009030.
Condition	n F	ising edge of	input si	ignal (C	DFF to	ON)	Input (000000)	
	D	Result (bina	ary)				Register value (009030)	376 377 377
Contents after		Result (octal)	Zero 007357	Non-carry 007354	(,			
operation	Flag	377 → 000	1	1	0	0	Error flag (007355)	¥
		Other than above	0	0	0	1	Carry flag (007356)	
Resembled	instr	uctions. E ea	E 624	/ E 16	2 E 1/	2214	Zoro flog	

Resembled instructions: F-63, F-63w, F-163, F-163w, F-163d, F-263, F-263w, F-263d

- The contents of D are represented by a binary number, which can assume a number of 000 to 255 in the decimal notation or 000 to 377(8) in the octal notation.

9030 009030 ondition 000000 changes from W300 counts up the content of 000 000 001 001 002 002 003 003 004 377 OCT indicates an octal number 377. → ← 1 scan time, max. Valid until a flag affecting instruction is met in the program. Γ Zero flag (007357)

Instruction

STR F-63

000000

(octal)

F-63w INC

F-63 INC

Increments counter (1 word) (INCrement)

Symbo	ł	F-63w INC	D				[Explana	-	Instruction S T R 000002	2
Function	n	The binary c and D+1 are			•	ers D	F-63w 019000 F-63w 019000 When the input condition 000002 changes from 019000 019000 019000			
Operatio	on 🛛	⟨D, D+1⟩+1-	→D, D-	⊦1				ON, the JW300 count	•	
D		Jse range B - Be sure to us (Odd address prohibited to	e even a ses such				Input (000002) Registers	r 019000 and 019001.		-
Conditio	on F	Rising edge o	of input	signal (OFF to	ON)	(019000) (019001)	FFFD FFFE FFFF 0000	0001 0002 0003 ((Hex.)
	D	Lower dig	its of re	sult			Non-carry flag (007354)			-
Contents	D+1	Upper dig	its of re	sult			Error flag (007355)			-
after operation		Result (octal)					Carry flag (007356)			-
	Flag	177777 →000000	1	1	0	0	Zero flag (007357)			
		Other than above	0	0	0	1		1 scan time, max.		

Resembled instructions: F-63, F-63d, F-163, F-163w, F-163d, F-263, F-263w, F-263d

F-63d INC INCrements counter (2 words) (INCrement)

Symbo	I	F-63d INC	D				[Explanat	-	Instruction S T R 000002	
Function	n	The binary c to D+3 are ir			registe	ers D	$\begin{array}{c c} \hline & F-63d \\ \hline & F-63d \\ \hline & I \ N \ C \\ \hline \end{array} \end{array} \qquad \qquad$			
Operatio	n	$\langle D \text{ to } D+3 \rangle +$	1→D to	o D+3				ON, the JW300 coun	0	
D		Use range A - Be sure to use (Odd address prohibited to u	e even a ses such		•		register Input (000002) Registers (019000 to	r 019000 to 019003.	0000000000 0000000000000000000000000000	
Conditio	n	Rising edge o	of input	signal ((OFF to	ON)	019003) Non-carry flag	h	h	
	D to D+3		-				(007354) Error flag (007355)	OFF		
Contents after		Result (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		h		
operation	Flag	3777777777777 →000000000000	1	1	0	0	Zero flag	П		
		Other than above	0	0	0	1	(007357)	1 scan time	e, max.	

Resembled instructions: F-63, F-63w, F-163, F-163w, F-163d, F-263, F-263w, F-263d

F-64
DECDecrements counter (1 byte)
(DECrement)

Symbo	I	F-64 DEC	D				[Explanation] Instruction 000000 STR 000000		
Function		The content data) are de		-	er D (b	inary	F-64		
Operatio	n	⟨D⟩−1→D					When the input condition 000000 changes from OFF to ON, the JW300 counts down the content		
D		Use range A	A				of register 009000.		
Conditio	n	Rising edge	of input	signal	(OFF to	ON)	(000000) Begister value		
	D	Result (bi	inary)				(009000) 001000 000377 377 376 376 375 375 374 374 373 (001)		
Contents		Result (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	Valid until a flag affecting instruction Error flag		
after operation	Flag	001 → 000	1	0	0	1	(007355)		
	n ay	000 → 377	0	1	0	0	Carry flag		
		Other than above	0	0	0	1	Zero flag (007357)		

- The contents of D are represented by a binary number, which can assume a number of 000 to 255 in the decimal notation or 000 to 377(8) in the octal notation.

Resembled instructions: F-64w, F-64d, F-164, F-164w, F-164d, F-264, F-264w, F-264d

Decrements counter (1 word)

DEC (DECrement) [Explanation] F-64w Instruction Symbol D DEC STR 000002 F-64w The binary contents of the registers D, 000002 F-64w Function DEC 019000 D+1 are decremented. 019000 Operation $\langle D, D+1 \rangle - 1 \rightarrow D, D+1$ When the input condition 000002 changes from OFF to ON, the JW300 counts down the content D Use range B * of register 019000 and 019001. Condition Rising edge of input signal (OFF to ON) Input (000002) Registers (019000) D Lower digits of result 0002 0001 0000 FFFF FFFE FFFD FFFC (019001) Upper digits of result D+1 Non-carry flag (007354) OFF Result Non-carry Zero Carry Error Error flag (007355) Contents (octal) 007357 007356 007355 007354 after 000001 1 0 0 1 Carry flag (007356) operation →000000 Flag 000000 0 0 1 0 +177777 Zero flag (007357) \rightarrow -Other than 0 0 0 1 1 scan time, max. above

* Be sure to use even addresses for register D.

F-64w

(Odd addresses such as 019003 etc. are prohibited to use.)

Resembled instructions: F-64, F-64d, F-164, F-164w, F-164d, F-264, F-264w, F-264d

F-64d DEC

Decrements counter (2 words) (DECrement)

Sym	bol	F-64d DEC	D				[Explanation] Instruction S T R 000002
Func	tion	The binary to D+3 are			e registe	ers D	F-64d F-64d D E C 019000 019000
Opera	ation	$\langle D \text{ to } D+3 \rangle$	—1→D	to D+3	3		When the input condition 000002 changes from
D)	Use range - Be sure to us (Odd addres prohibited to	se even a ses such		•		OFF to ON, the JW300 counts down the content of register 019000 to 019003.
Cond	lition	Rising edge	of inpu	t signal	(OFF to	o ON)	Registers (019000 to
	D to D+3	D : Lower o D+3 : Uppe	•		ult		019003) Non-carry flag (007354) .OFF
Contents		Result (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	Error flag (007355)
after operation	Flag	00000000001 →00000000000000000000000000	1	0	0	1	Carry flag (007356)
	ay	0000000000 →3777777777777777777777777777	0	1	0	0	Zero flag (007357) → → ≺
		Other than above	0	0	0	1	1 scan time, max.

Resembled instructions: F-64, F-64w, F-164, F-164w, F-164d, F-264, F-264w, F-264d

F-65
BCDIBCD increment counter (1 byte)(BCD Increment)

Sym	bol	F-65 BCDI	D				[Explanation] Instruction S T R 000000			
Funct	tion	Increment t register D b		tents (E	BCD) of		F-65			
Opera	ation	⟨D⟩+1→D					When the input condition 00000 changes from			
D		Use range	A				OFF to ON, the JW300 counts up (+1) the			
Condi	ition	Rising edge	of inpu	t signal	(OFF to	ON)				
	D	Result (E - Unaffec D are not	ted if the	e conten	its of reg	gister	(000000) Register value (009000) <u>98 99 00 01 02 03 04 () 8A</u>			
Contents		Result (BCD)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	Non-carry flag (007354) → ← 1 scan time, max.			
after operation	Flag	99 to 00	1	1	0	0	Error flag OFF Valid until a flag affecting (007355) // instruction is met in the program.			
		Other than above	0	0	0	1	Carry flag (007356)			
		Not BCD code	0	0	1	0	Zero flag			

- If the contents of register D are not BCD code, the error flag (007355) is set ON and no division will be done. (D remains unchanged.)

Resembled instructions: F-62, F-62w, F-62d, F-65w, F-65d

F-65w
BCDIBCD increment counter (1 word)BCDI(BCD Increment)

Syml	ool	F-65w BCDI)				[Explanation] Instruction S T R 000002 F -65w
Funct	ion	Increment the registers D, E		•	CD) of		000002 F-65w BCDI 019000 019000
Opera	tion	⟨D, D+1⟩+1-	→ D, D+	-1			When the input condition 000002 changes from
D		Use range B - Be sure to use (Odd address prohibited to u	es such				OFF to ON, the JW300 counts up the content of register 019000 and 019001.
Condi	tion	Rising edge of	f input s	signal (OFF to	ON)	Register value (019000) 9998 9999 0000 0001 0002 0003 0004 008A
	D	Lower digits result		Unaffec contents			(019001) Non-carry flag
	D+1	Upper digits result	s of 🔤	D and D BCD co)+1 are		(007354)
Contents after		Result (BCD)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	Valid until a flag affecting OFF instruction is met in the program.
operation	Flag	9999 →0000	1	1	0	0	(007355)
		Other than above	0	0	0	1	(007356) ((
		Not BCD code	0	0	1	0	(007357)

- If the contents of registers D and D+1 are not BCD code, the error flag (007355) is set ON and no division will be done. (D, D+1 remains unchanged.)

Resembled instructions: F-62, F-62w, F-62d, F-65, F-65w

F-65d	
BCDI	

BCD increment counter (2 words) (BCD Increment)

Symbo	I	-F-65d D BCDI D					[Explanation] Instruction S T R 000002
Functio	n	Increment the contents (BCD) of registers D to D+3 by one.			,		F-65d F-65d BCDI 019000
Operatio	Operation $(D \text{ to } D+3)+1 \rightarrow D \text{ to } D+3$					When the input condition 000002 changes from	
D		o use even addresses for register D. dresses such as 019003 etc. are				OFF to ON, the JW300 counts up the content of register 019000 to 019003.	
Conditio	n	Rising edge of	f input s	signal (OFF to	ON)	(00002) Registers <u>99999998 99999999 000000000</u> (019000 to <u>99999998 99999999 000000000</u>
	D to D+3					*	`019003) Non-carry flag
Contents		Result (BCD)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	(007354) Error flag (007355)
after operation	Flag	99999999 →00000000	1	1	0	0	Carry flag (007356)
		Other than above	0	0	0	1	Zero flag (007357) → ←
		Not BCD code	0	0	1	0	1 scan time, max.

* If the contents of registers D to D+3 are not BCD code, the error flag (007355) is set ON and no division will be done. (D to D+3 remains unchanged.)

Resembled instructions: F-62, F-62w, F-62d, F-65, F-65w

F-66 BCDD

F-66w BCDD

BCD decrement counter (1 byte) (BCD Decrement)

Symbo	BCDD)				[Explanation] Instruction S T R 000000
Functio	unction Decrement the contents (BCD) of register D by one.			BCD) o	f	F-66 F-66 BCDD 009000 009000	
Operati	on	⟨D⟩−1→D					When the input condition 000000 changes from
D Use range A						OFF to ON, the JW300 counts down (-1) the	
Conditi	ion Rising edge of input signal (OFF to ON)				OFF to	ON)	content of register 019000.
	D	Result (BC - Unaffected D are not	d if the o	content	s of reg	ister	Input L <thl< th=""> <thl< th=""> <thl< th=""></thl<></thl<></thl<>
		Result	Zero 007357	Carry 007356		Non-carry 007354	ON Non-carry flag (007354) OFF → ←
Contents		01→00	1	0	0	1	1 scan time, max. Valid until a flag affecting instruction is met in the program.
operation	Flag	00→99	0	1	0	0	Error flag (007355)
		Other than above	0	0	0	1	Carry flag (007356) — — — — — — — — — — — — — — — — — — —
		Not BCD code	0	0	1	0	Zero flag (007357)

- If the contents of registers D are not BCD code, the error flag (007355) is set ON and no division will be done. (D remains unchanged.)

Resembled instructions: F-62, F-62w, F-62d, F-66w, F-66d

BCD decrement counter (1 word) (BCD Decrement)

Symbo	bl	F-66w)				[Explanation]	Instruction
		BCDD						STR 000002 F-66w
Functio	n	Decrement th registers D, I		•	BCD) o	of	F-66w 019000	019000
Operation $\langle D, D+1 \rangle - 1 -$			→D, D)+1			When the input condition 00000	
		Use range B					OFF to ON, the JW300 counts of	•
D		- Be sure to use even addresses for register D.					of register 019000 and 019001.	
		(Odd addresses such as 019003 etc. are prohibited to use.)					Input(000002) OFF	
Conditio	on	Rising edge o	finput	signal (OFF to	ON)	Register value)
	D	Lower digit	s of -	If the cor registers not BCD	D and	D+1 are	(019001)	1997 9996 9995 (/ 998A
	D+1	Upper digits result		D+1 rem	nains	Janu	(007354) OFF	→ ← 1 scan time, max. Valid until a flag affecting
Contents		Result (octal)	Zero 007357	Carry 007356		Non-carry 007354	OFF Error flag	instructions is met in the program.
after		0001 →0000	1	0	0	1	(007355)	
operation	Flag	0000 →9999	0	1	0	0	Carry flag (007356)	
		Other than above	0	0	0	1	Zero flag (007357)	
		Not BCD code	0	0	1	0		

- If the contents of registers D and D+1 are not BCD code, the error flag (07355) is set ON and no division will be done. (D, D+1 remains unchanged.)

Resembled instructions: F-62, F-62w, F-62d, F-66, F-66d

F-66d BCDD

BCD decrement counter (2 words) (BCD Decrement)

Symbo	Symbol - F-66d D						[Explanation] Instruction S T R 000002 F -66d
Functio	n	Decrement the contents (BCD) of registers D to D+3 by one.					000002 F-66d BCDD 019000 019000
Operation $\langle D \text{ to } D+3 \rangle -1 \rightarrow D \text{ to } D+3$					When the input condition 000002 changes from		
D		Use range C - Be sure to use (Odd addresse prohibited to u	es such a				OFF to ON, the JW300 counts down the content of register 019000 to 019003.
Conditio	on	Rising edge of	f input :	signal (OFF to	ON)	Input {/ //
	D to D+3					*	Registers (019000 to 019003)
		Result (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	(007354)
Contents after		00000001 →00000000	1	0	0	1	Error flag (007355) ///////////////////////////////////
operation	Flag	00000000 →99999999	0	1	0	0	(007356)
		Other than above	0	0	0	1	Zero flag (007357) \rightarrow \leftarrow {} 1 scan time, max.
		Not BCD code	0	0	1	0	

* If the contents of registers D to D+3 are not BCD code, the error flag (007355) is set ON and no division will be done. (D to D+3 remains unchanged.)

Resembled instructions: F-62, F-62w, F-62d, F-66, F-66w

Digit shift (up)

Symbol	F-67 NSFt		[Expla			Inst S T R F -67		tion 004000		
Function		e registers upper 4 bits for ytes starting with register D.	F-67 NSFH 010 009000						010	
Operation	- After the the lowe shift op	hrough D+n-1 up 4 bits each e operation, the PC stores 0 to er 4 bits of register D. After the eration, the upper 4 bits "n" egisters are cleared.	When input condition 004000 changes from OFF to ON, the 8 (010(8)) bytes of data in registers 009000 to 009007 are shifted upper 4 bits. Before operation \longrightarrow After operation						ers	
n	000 to 37	77 ₍₈₎ (256 bytes for 000(8))		009000	2 1		1 0)		
D	Use rang	je A	-	009001	4 3		3 2	_		
Condition	Rising ed	ge of input signal (OFF to ON)		009002	6 5 8 7		5 4 7 6	_	Shifted up	
Contents	D	Result (shift result)	8 bytes	009004	10 9		9 8	;	4 bits each	
after operation	Flag	Unchanged	-	009005	12 11		11 10	_	-	
		<u> -</u>		009006	14 13 16 15		13 12 15 14	_	↓ ↓	
				•		+	16 ↓ Cleared			

F-68 NSFL

F-67 NSFH

Digit shift (down)

Symbol	F-6	⁻ n)	[Expla	nation]			STR	ruction 004010
Function		ne registers to lower 4 bits for bytes starting with register D.			F-68 NSFL 010	009100	F -68	010 009100
Operation	- After the up the up which the shi	through $D + n - 1$ down 4 bits each ne operation, the PC stores 0 to per 4 bits of register $D + n - 1$ is at the beginning, store 0. After ft operation, the lower 4 bits D pers are cleared.	When input condition 004010 changes from OFF to ON, the 8 (010(8)) bytes of data in registers 009100 to 009107 are shifted to lower 4 bits. Before operation \longrightarrow After operation					
n	000 to 3	77(8) (256 bytes for 000(8))		009100	15 16 13 14		14 <u> 15</u> 12 <u> </u> 13	
D	Use ran	ge A		009102	11 12		10 11	
Condition	Rising e	8 bytes	009103	9 10 7 8	+	8 <u>9</u> 6 7	Shifted down 4 bits each	
Contents after	D]	009105 009106	5 6 3 4	+	4 <u>5</u> 2 <u>3</u>		
operation	Flag	Unchanged		009107	1 2		0 1	

F-69 NXFR

Digit transfer

Symbol	F-6		[Explanation]	Instruction S T R 004002 F -69				
Function		ers the lower 4 bits of register e lower 4 bits of register D.	004002 F-69 NXFR 009100 009130	009100 009130				
Operation	Lower 4	4 bits of S \rightarrow Lower 4bits of D	When input condition 004002 changes from OFF to ON, the contents of the lower 4 bits of register 009100 are transferred to the lower 4 bits of register 009130.					
S	Use rar	nge A						
D	Use rar	nge A						
Condition	Rising e	edge of input signal (OFF to ON)						
	S	Unchanged						
Contents after operation	D	Contents of lower 4 bits of register S ; the upper 4 bits remain unchanged	Transfer					
	Flag	Unchanged						

F-70 FILE **Transfer "n" byte in batch** (FILE)

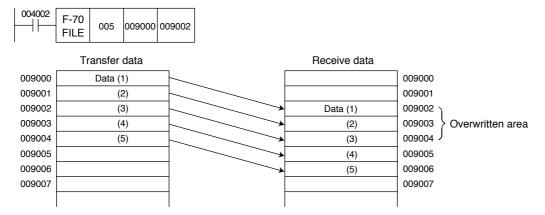
Symb	ool	F-70 n S D	[Explanation] Instruction S T R 004001						
Functi		Transfer data of S+n-1 data (n bytes) from register S to D+n-1 (n bytes) of register D.	004001 F-70 FILE 040 009000 □00000 FILE 040 009000 □00000 009000 009000 □00000						
Operation		S, S+1, ·····, S+n−2, S+n−1 →D, D+1, ·····, D+n−2, D+n−1	When the input condition 004001 changes from OFF to ON, 040(8) -byte data (32 bytes in decimal) in register 009000 through 009037 are transferred						
n		Use range 000 to 377(8) (256 bytes for 000(8))	in batch to the 32-byte area of ⊐00000 through ⊒00037.						
S		Use range A	The contents of registers, 009000 through						
D		Use range A	009037, remain unaffected.						
Condit	tion	Rising edge of input signal (OFF to ON)	Before operation After operation 009000 1 2 1 2 100000						
	S, S+n—	1 Unchanged							
Contents after operation	D D+1 :	Contents of the register S Contents of the register S+1							
	D+n—: D+n—		009037 9 9 9 3 300037						
	Flag	Unchanged							

- Do not transfer the contact point area (file address 00001600 to 00001777 etc.) of the timer/counter of file number 0.

F-70 FILE	5	019000	⊐01575	
--------------	---	--------	--------	--

If programmed in the above manner, the contents of 019003 and 019004 will be transferred to 00001600 and 00001601.

- Setting n, S, and D which may overwrite the destination, is available.



Resembled instructions: F-00, F-00w, F-00d, F-70w, F-70d, F-74, F-74w, F-74d, F-76, F-76w, F-76d

F-70v FILE		Transfers "n" words in (FILE)	batch						
Symbo	ol	F-70w n S D	[Explanation]	Instruction S T R 004000 F -70w					
Functio	on	Transfer data of S+2n-1 data (n words) from register D to D+2n-1 (n words) of register D.	004000 F-70w 040 009000 019000 FILE 040 009000 019000 019000						
Operatio	ion	S, S+1 ······, S+2n−2, S+2n−1 →D, D+1, ·····, D+2n−2, D+2n−1	When the input condition 00400 to ON, 040(8)-words data (32 wo	ords in decimal) in					
n		Use range 000 to 377(8) (256 words for 000(8))	registers 009000 to 009077 are transferred in batch to the 32-words area of 019000 to 019077. The contents of registers, 009000 to 009077, remain unaffected.						
S		Use range B - Be sure to use even addresses for registers S.							
D		Use range B - Be sure to use even addresses for registers D.		After operation 5 6 019000					
Conditio	on	Rising edge of input signal (OFF to ON)		7 8 019001					
	S, …, S+2n-	1 Unchanged		1 3 019002					
		Contents of the register S Contents of the register S+1 : -2 Contents of the register S+2n-2 -1 Contents of the register S+2n-1		2 8 019003					
	Flag	Unchanged		9 7 019076					
Resemble	ed ins	tructions: F-00, F-00w, F-00d, F-70, F-70d, F-74, F-74w, F-74d, F-76, F-76w, F-76d	009077 5 4	5 4 019077					

F-70d FILE **Transfers "n" double words in batch** (FILE)

Symb	lool		-70d n S		[Explanation]				Ir S T F -7		ion 04000	
Funct	ion	doub	sfer data of S to ble words) from re double words) of	egiste	r D to D+4n-	004000 F-70d 040 009000 019000 00900						040 09000 19000
Opera	tion		S+3, ·····, S+4 to D+3, ·····, D			When the input condition 004000 changes from OFF to ON, 040(8)-double words data (32-double						
n Use range 000 to 377(8) (256 double words for 000(8))					are tra	nsferred	in batc	egisters h to the				
S	S Use range C *1					area of 019000 to 019177. The contents of registers, 009000 to 009177,						7,
D	Use range C *2						unaffe	cted.	,			
Condit	tion	Risin	g edge of input s	ignal ((OFF to ON)		Before o	·	7	After op		
	S, … S+4n	, 1	Unchanged			009000 009001	5 7	6 8		5 7	6	019000
Contents	D to D+4 to		Contents of the Contents of the			009002 009003	1 2			1 2	3 8	019002
after operation		-4 4n-1	: Contents of the S+4n-4 to S+4		r	009004 009005		A C	B D	019004		
	Fla	ag	Unchanged			009006 009007	E 1	F 2		E 1	F 2	019006
	*1 Be sure to use even addresses for register S. *2 Be sure to use even addresses for register D.						1 1 1 1 1 1	-	L			
Reser	Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-74, F-74w,					009174 009175 009076	5 8	6		5 8	6	019174
	F-70W, F-74, F-74W, F-74d F-76 F-76W						9	7	4	9	, 7	019076

009177

5

4

5

019177

4

F-74d, F-76, F-76w, F-76d

F-71 CONS

Transfers octal constant in batch (1 byte) (CONStant)

Symbo	bl		[Explanation]	Instruction S T R 004001		
Function		An octal constant "n" is transferred in batch from the register D1 to the register D2.	004001 F-71 000 009000 009037	F -71 000 009000 009037		
Operatio	on	n→D1,, D2	When the input condition 004001 changes from OFF to ON, the octal constant 000 is transferred			
n		Use range 000 to 377(8)	in batch from registers 009000 to 009037.			
D1		Use range A	Before operation	After operation		
D2		Use range A		0 0 0 0 0 0 0 0 009000		
Conditio	on	Rising edge of input signal (OFF to ON)	$009001 \times \times \times \times \times \times \times \times \longrightarrow 0$	0 0 0 0 0 0 0 009001		
Contents after operation	D1 D1+ : D2- D2	Constant "n"		0 0 0 0 0 0 0 0 009037		
	Flag	Unchanged				

- No operation takes place if D1 or D2 is programmed that may override (1) to (9), as shown below.

Range		JW-311CU JW-312CU	JW-321CU JW-322CU	JW-331CU JW-332CU	JW-341CU JW-342CU	JW-352CU	JW-362CU				
	(1)		⊐00000 to ⊐01577								
Relay	(2)		⊐C	2000 to ⊐0757	77						
	(3)		⊐10000 to ⊐15377		⊐10000 to	⊐54377					
Value of	(4)		b00000 to b01777								
TMR,	(5)	b02000 to b03777									
CNT, MD	(6)		b04000 to b07777		b04000 to	b37777					
Degister	(7)	009000 to E07777									
Register	(8)		1	09000 to Z377	7						
FILE 1	(9)		00000000 to 00077777	00000000 to 00377777	00000000 to 01777777	00000000 to 07777777	00000000 to 37777777				

- No operation takes place if the address D_1 is greater than D_2 .

 F-71 CONS
 050
 009200
 009000
 D1(009200) > D2(009000)

Resembled instructions: F-08, F-08w, F-08d, F-71w, F-71d

F-71w CONS (CONStant) Transfers octal constant in batch (1 word)

Symb	ol		NS n D1 D2	[Explanation]	Instruction S T R 004000	
FunctionAn octal constant "n" is transferred in batch in registers D1, D1+1, to D2, D2+1.			n in registers D1, D1+1, to D2,	F-71w 01234 00900 009036 00903 00903 009036		
Operat	Operation		D1, D1+1), ······ (D2, D2+1)	When the input condition 004000 changes from OFF to ON, the octal constant 012345 is transferred in batch from registers 009000,		
n	n		range 000000 to 177777(8)			
D1		Use	range B *1	009001 to 009036, 009037.		
D2	D2 Use range B *2			After operation		
Conditi	ion	Rising	edge of input signal (OFF to ON)	009001 0 0 0 1 0 1 0 0 1 1 1 0 1 2 3	0 0 1 0 1 009000	
Contents after operation			Constant "n"			
			Unchanged	009037 0 0 0 1 0 1 0 0 1 1 1		

*1 Be sure to use even addresses for register D_1 .

F-71d CONS

*2 Be sure to use even addresses for register D₂. Resembled instructions: F-08, F-08w, F-08d, F-71, F-71d

Transfers octal constant in batch (2 words) (CONStant)

Symbol			71d n D1 D2	[Explanation]	Instruction S T R 004000	
Functi	on		ctal constant "n" is transferred in n in registers D1 to D1+3, to D2 to 8.	004000 F-71d CONS 01234567000 009000 009040	F -71d 01234567000 009000 009040	
Operat	Operation		D1 to D1+3), (D2 to D2+3)	When the input condition 004000 changes from OFF to ON, the octal constant 012345 is transferred in batch from registers 009000, 009001 to 009036, 009037. After operation		
n	n		range 00000000000 to 3777777777(8)			
D1	D1		range C *1			
D2		Use	range C *2	009003 009002 00900	1 009000	
Conditi	on	Rising	edge of input signal (OFF to ON)	$\begin{array}{c} 0 \\ 0 \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 6 \\ 6 \\ 6 \\ 6 \\ 6 \\ 6 \\ 6 \\ 6$		
Contents after operation	after		Constant "n"	009007 009006 00900 0 0 0 0 1 0 1 0 0 1 1 1 0 0 1 0 1 1 1 0 1 009043 009042 00904	1 1 0 0 0 0 0 0 0 0 0 0 1 009040	
	FI	ag	Unchanged	00001010011100101110010	1 1 0 0 0 0 0 0 0 0 0	

*1 Be sure to use even addresses for register D_1 .

*2 Be sure to use even addresses for register D₂. Resembled instructions: F-08, F-08w, F-08d, F-71, F-71w

F-72 DMPX

Demultiplex "n" bytes to file 1 register (DeMultiPleXer)

		-			-					
Sym	Ibol		72 n IPX n	S	D]	[Explanation] Instruction S T R 004000			
Function The contents of an "n" byte register group heralded by the register which is modified by the contents of S (data pointer) from the register S contained data memory block top address (reference address) are transferred to the file 1 register which address is determined by the contents of the registers D to D+2.					r which ata po ta me addre regis	ch is modified binter) from the emory block ess) are ter which	$\begin{bmatrix} F-72 & 015 \\ \hline DMPX & 015 \end{bmatrix} \xrightarrow{100413} 009011 \\ \hline F-72 & 015 \\ \hline D00413 \\ \hline 009011 \\ \hline D09011 \\ \hline F-72 \\ \hline 000413 \\ \hline 009011 \\ \hline $			
Opera	Operation $X+\langle S \rangle, \dots, X+\langle S \rangle+n-1$ $\rightarrow \langle D \text{ to } D+2 \rangle, \dots, \langle D \text{ to } D+2 \rangle+n-1$ X: Top address of the data memory block in which S is contained (reference address) $\langle S \rangle$: Data pointer				D to E data	memory block	pointer) from ⊐00400 (top address of the register ⊐00413 contained data memory block…reference address) are transferred to a 015(8) register group heralded by the file 1 register which address is determined by the contents of the registers 009011 and 009012.			
n	n Use range 000 to 377(8) (256 bytes for 000(8))						$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			
S	;	Use r	ange A				$\begin{array}{c c c c c c c c c c c c c c c c c c c $			
D)	Use r	ange E							
Cond	lition	Rising	edge of	input sig	gnal	(OFF to ON)	009012 1 1 1 1 0 1 1 0 			
	Other t file 0 R		Unchan	ged						
Contents after operation	after (D to L		Contents Contents				Reference address			
oporation	⟨D to D+	2>+n−1	Contents	s of regi	ster >	(+⟨S⟩+n−1	100400 +D.P (162)			
	Flag Unchanged						c 100562			
Resembled instructions: F-05, F-05w, F-05d, F-72w, F-72d - The reference address of the register side is ⊐00000, ⊐00400, ⊐01000···019000, 019400 which is the top address of the data memory block in which S is contained. => "Data memory block and reference address."						ide is 9400 which is ock in which S	015(e) bytes 10573204 015(e) 10573204 015(e) 015(e) 015(e) 10573220 015(e) bytes 10573220			

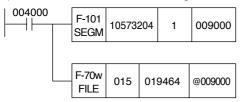
Demultiplexes "n" words to file 1 register (DeMultiPleXer)

Symb	ol	F-72 DMF		[Explanation] Instruction S T R 004000				
Functio	on	which a referen register modifie are trar addres	ntents of an "n"-word register group address is determined by the ice address (the top address of the r S included data memory block) ad by the contents of S (data pointer) insferred to the file 1 register whose s implied by the contents of the rs D to D+2.	Image: When the input condition 004000 Changes from OFF to ON, the contents of a 015(8)-words register group heralded by the register which is modified by the contents of the register 019776 (data pointer) from 019400 (the top address of the				
Operat	ion	→〈D t X: Top in ad	$\langle \dots, X+\langle S \rangle+2n-1$ to D+2 $\rangle, \dots, \langle D$ to D+2 $\rangle+2n-1$ address of the data memory block which S is contained (reference dress) ata pointer	data block in which contained the register 019776 reference address) are transferred to a 015(8)- words register group heralded by the file 1 register of which address is determined by the contents of the registers 009000 to 009002.				
n			ange 000 to 377 ₍₈₎ vords for 000 ₍₈₎)	019776 0 1 1 1 0 1 0 0 0 6 0 $$				
S		Use ra	ange B *1					
D		Use ra	ange C *2	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$				
Conditi	ion	Rising e	edge of input signal (OFF to ON)	(Be sure to use even addresses)				
		r than register	Unchanged					
	⟨D to	$D+2\rangle$	Contents of register $X+\langle S \rangle$	Reference address				
Contents after	⟨D to	D+2>+1	Contents of register $X+\langle S \rangle+1$	019400 +D.P (064) 00000000				
operation	〈D to	D+2⟩+ 2n—1	Contents of register X+ $\langle S \rangle$ +2n-1	015(8) words 019515 0573204 015/80				
F		lag	Unchanged	019515 10573204 019515 015(8) words				
*2 Be sure	e to u	se even	addresses for register S. addresses for register D. ions: F-05, F-05w, F-05d, F-72, F-72d	Register File register (File 0) (File 1)				

Reference

F-72w DMPX

Operation similar to the following instructions will take place for the above.



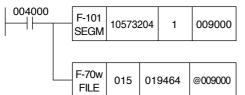
F-72d DMPX

Demultiplexes "n" double words to file 1 register (DeMultiPleXer)

Symb	ol	F-72d n S D [Explanation]	Instruction S T R 004000
Function		by the reference address (the top address of the register S included data memory block) modified by the contents of S (data pointer) are transferred to the file 1 register whose address implied by the contents of the registers D to D+2.	F -72d 015 019776 009000 on 004000 changes from is of a $015(8)$ -double words by the register which is s of the register 019776 400 (the top address of the
Operation		$X_+(S), \dots, X_+(S)_{+4n-1}$ $\rightarrow \langle D \text{ to } D_{+2} \rangle \dots \langle D \text{ to } D_{+2} \rangle_{+4n-1}$ data block in which con \dots reference address) ar	tained the register 019776 e transferred to a 015(8)- roup heralded by the file 1 ss is determined by the
n		(256 double words for $000_{(8)}$)	064 ₍₈₎ Has 019464 for the top address (Be sure to use even addresses)
S		Use range B *1	
D		Use range C *2 Dising edge of input signal (OFF to ON)	Has 10573204 for the top address
Conditi	ion	Rising edge of input signal (OFF to ON)	(Be sure to use even addresses)
		Der than 0 register Unchanged 009002 0 0 1 0 0 1 0 1 - 0 - 5 -	
	〈D to	o D+2 Contents of register X+(S) Reference address	
after operation		o D+2 \rangle +1 Contents of register X+ $\langle S \rangle$ +1 : o D+2 \rangle + 4n-1 Contents of register X+ $\langle S \rangle$ +4n-1	Demultiplexeq
FI		Flag Unchanged Unchanged	10573204
*2 Be sure	e to u	use even addresses for register S. use even addresses for register D. d instructions: F-05, F-05w, F-05d, F-72, F-72w	File register (File 1)

Reference

Operation similar to the following instructions will take place for the above.



- The reference address of the register side is \(\gamma\)00000, \(\gamma\)01000...019000, 019400 which is the top address of the data memory block in which S is contained. => "Data memory block and reference address."

F-73 MPX

Multiplex "n" byte from file 1 register (MultiPleXer)

Sym	bol	— F-73 n S D MPX n S D	[Explanation] Instruction S T R 004000
FunctionTransfer data with the following conditions. Source: n bytes data starting from a file register specified by register S to S+2. Destination: n bytes from a register displaced by D counted from the top address of the data memory block having register D.			$\begin{bmatrix} 004000 \\ H \\ H \\ H \\ H \\ MPX \end{bmatrix} \begin{bmatrix} F-73 \\ 009101 \\ 009501 \end{bmatrix} \begin{bmatrix} F-73 \\ 009101 \\ 009501 \end{bmatrix}$ $\begin{bmatrix} F-73 \\ 009101 \\ 009501 \\ 009501 \end{bmatrix}$ When the input condition 004000 changes from OFF to ON, the contents of a 015(8) bytes register group headed by the file 1 register of which
Opera	ation	$ \begin{array}{l} \langle S \text{ to } S+2\rangle, \cdots, \langle S \text{ to } S+2\rangle +n-1 \\ \rightarrow X+\langle D\rangle, \cdots, X+\langle D\rangle +n-1 \\ X: \text{ Top address of the data memory in } \\ \text{which } D \text{ is contained (reference address)} \\ \langle D\rangle: \text{ Data pointer} \end{array} $	address is determined by the contents of the registers 009101and 009103 are transferred to 015(8) bytes register group headed by the register which is modified by the contents of 009501 (data pointer) from 009400 (top address of the 009501 contained data memory block…reference address).
n		Use range 000 to 377 ₍₈₎ (256 bytes for 000)	$009501 [0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0$
S		Use range E	009101 0 1 0 0 0 1 1 0 Reference address (top address of the block in which 009501 is
D)	Use range A	
Cond	ition	Rising edge of input signal (OFF to ON)	009102 0 0 1 1 0 1 0 1 Has 00032506 for the top address
	File 1 registe	Unchanged	009103 0 0 0 0 0 0 0 0
Contents after operation	$ \begin{array}{c} X+\langle D\rangle\\ X+\langle D\rangle\\ \vdots\\ X+\langle D\rangle\\ +n- \end{array} $: Contents of file 1 register	Reference address +D.P (162) (009562
Flag		Unchanged	015(8) bytes 009576 00032506 015(8) 00032506
- The re ⊐00000 the top	ferenco 0, ⊐004 0 addre	tructions: F-06, F-06w, F-06d, F-73w, F-73d e address of the register side is 00, ⊐01000…019000, 019400 which is ss of the data memory block in which S	C009576 015(8) Register File register (File 0) (File 1)

the top address of the data memory block in which S is contained. => "Data memory block and reference address."

F-73w MPX

Multiplexes "n" words from file 1 register (MultiPleXer)

				1			
Symbo	ol		F-73w n S D	[Explanation] Instruction S T R 004001			
Function Source: n bytes data register specified by n Destination: n words displaced by D count address of the data n			ansfer data with the following conditions. urce: n bytes data starting from a file gister specified by register S to S+2. stination: n words from a register placed by D counted from the top dress of the data memory block having gister D.	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$			
Operation $\begin{cases} \langle S \text{ to } S \\ \rightarrow X_+ \langle I \\ X \colon Top \\ whi \\ add \end{cases}$			S to S+2 \rangle , \langle S to S+2 \rangle +2n-1 X+ \langle D \rangle ,X+ \langle D \rangle +2n-1 :: Top address of the data memory in which D is contained (reference address) D \rangle : Data pointer	address is determined by the contents of the registers b00000 to b00002 are transferred to an "n"- words register group headed by the register which is modified by the contents of the register ⊐01024 (data pointer) from ⊐01000 (top address of the data memory block in which contained the			
n			e range 000 to 377(8) 56 bytes for 000(8))	register $\exists 01024\cdots$ reference address). $\exists 01024 \qquad 0 \qquad 1 \qquad 1 \qquad 1 \qquad 0 \qquad 1 \qquad 0 \qquad 044_{(8)}$. Has $\exists 01044$ for the top address (Be sure to use even addresses)			
S		Us	e range C *1				
D		Us	e range B *2				
Conditi	on	Risi	ing edge of input signal (OFF to ON)				
	File regis		Unchanged	b00001 0 0 1 1 0 1 0 1 			
Contents	X+ <c X+<c< td=""><td>•</td><td>Contents of file 1 register $\langle S \text{ to } S+2 \rangle$ Contents of file 1 register $\langle S \text{ to } S+2 \rangle$+1</td><td></td></c<></c 	•	Contents of file 1 register $\langle S \text{ to } S+2 \rangle$ Contents of file 1 register $\langle S \text{ to } S+2 \rangle$ +1				
after operation	X+⟨Ľ +2n	$\begin{array}{c c} \vdots & \vdots \\ \langle D \rangle \\ 2n-1 \\ \langle S \text{ to } S+2 \rangle + 2n-1 \end{array}$		Reference address			
	Fla	ag	Unchanged	+D.P (044)			
*1 Be sure *2 Be sure - The refe ⊐00000, the top a	e to u e to u erence ⊐004 addre	se e se e e ac 400, ess o	ctions: F-06, F-06w, F-06d, F-73, F-73d even addresses for register S. even addresses for register D. ddress of the register side is ⊐01000…019000, 019400 which is of the data memory block in which S "Data memory block and reference	15(8) 101044 015(8) 101075 101075 101075 Register 00032537 Register File register (File 0) (File 1)			

⊐00000, ⊐00400, ⊐01000…019000, 019400 which is the top address of the data memory block in which S is contained. => "Data memory block and reference address."

F-73d MPX

Multiplexes "n" double words from file 1 register (MultiPleXer)

			T			
Sym	ibol	— F-73d n S D MPX n S D	[Explanation] Instruction S T R 004001			
Func	tion	Transfer data with the following conditions. Source: n double words data starting from a file register specified by register S to S+2. Destination: n double words from a register displaced by D counted from the top address of the data memory block having register D.	$\begin{bmatrix} 004001 \\ MPX \\ MPX \\ 015 \\ model{model} 01024 \end{bmatrix} \begin{bmatrix} F-73d \\ 015 \\ b00000 \\ -101024 \end{bmatrix}$ $\begin{bmatrix} F-73d \\ 015 \\ b00000 \\ -101024 \end{bmatrix}$ When the input condition 004001 changes from OFF to ON, the contents of a 015(8)-double words register group headed by the file 1 register of which address is determined by the contents of			
Opera	ation	$ \begin{array}{l} \langle S \text{ to } S+2\rangle, \cdots \langle S \text{ to } S+2\rangle +4n-1 \\ \rightarrow X+\langle D\rangle, \cdots X+\langle D\rangle +4n-1 \\ \text{X: Top address of the data memory in which D is contained (reference address) } \\ \langle D\rangle: \text{Data pointer} \end{array} $	the registers b00000 to b00002 are transferred to a 015(8)- double words register group headed by the register which is modified by the contents of the register ¬01024 (data pointer) from ¬01000 (top address of the data memory block in which contained the register ¬01024…reference			
n		Use range 000 to 377(8) (256 double words for 000(8))	address).			
S	;	Use range C *1	$ \begin{array}{ c c c c c c c c } \hline \exists 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ \hline 0 & - & 4 & - & - & 4 & - \end{array} \begin{array}{ c c c c c c c c c c c c c c c c c c c$			
D)	Use range B *2				
Cond		Rising edge of input signal (OFF to ON)				
	File 1 registe	er Unchanged	b00001 0 0 1 1 0 1 0 1 Has 00032506 for the top address (Be sure to use even addresses)			
Contents after operation	$\begin{array}{c} X+\langle D\rangle\\ X+\langle D\rangle\\ \vdots\\ X+\langle D\rangle\\ +4n \end{array}$: Contents of file 1 register	$b00002 \underbrace{[0,0,0,0,0,0,0]}_{0,0,0,0,0,0}$ Reference address			
	Flag		101000 +D.P (044)			
Resemt	oled ins	structions: F-06, F-06w, F-06d, F-73, F-73w				
*2 Be su	ure to u	ise even addresses for register S. ise even addresses for register D. e address of the register side is	J01127 J0Jexeq 00032506 015(8) double words (52 bytes)			

File register (File 1)

Register

(File 0)

 \exists 00000, \exists 00400, \exists 01000····019000, 019400 which is the top address of the data memory block in which S is contained. => "Data memory block and reference address."



Transfers "n" bytes

Symb	ol	- F-74 n S D	[Explanation] Instruction S T R 004000
Functio	on	The contents of the register S are transferred to "n" bytes registers headed by the register D.	F-74 004000 F-74 nXFR 010 009013 019416 010 009013 019416
Operation $S \rightarrow D, D+1, \dots, D+n-1$		S→D, D+1,, D+n−1	When the input condition 004000 changes from OFF to ON, the contents of the register 009013
n		Use range 000 to 377 ₍₈₎ (256 bytes for 000)	are transferred to a 010(8) bytes registers headed by the register 019416.
S		Use range A	
D		Use range A	$\begin{array}{c} 009013 1 0 0 1 0 0 1 0 \\ 1 0 0 1 0 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 1 0 \\ 1 0 0 \\ 1 0 0 1 0 \\ 1 0 0 \\ 1 0 0 1 0 \\ 1 0 $
Conditi	ion	Rising edge of input signal (OFF to ON)	
	S	Unchanged	- 010 ₍₈₎ bytes 2 2 2
Contents after operation	D D+1 : D+n-	Contents of the register S	$ \rightarrow 10010010 019424 10010010 019425 $
	Fla	g Unchanged	

Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-70d, F-74w, F-74d, F-76, F-76w, F-76d

F-74w	
nXFR	

Transfers "n" words

Symb	ol		nXFR n S D	[Explanation]	Instruction S T R 004000
Function are		are	contents of the registers S, S+1 transferred to "n" words registers ded by the registers D, D+1.	F-74w nXFR 010 009014 019416	F-74w 010 009014 019416
Operation		S, S	S+1→D, D+1…D+2n—2, D+2n—1	When the input condition 00400 OFF to ON, the contents of the	registers 009014
n			e range 000 to 377(8) 6 words for 000)	and 009015 are transferred to a registers headed by the register 019417.	
S		Use	e range B *	010417.	
D		Use	e range B *		09417 019416 6 7 8 9
Condit	ion	Risir	ng edge of input signal (OFF to ON)		6 7 8 9
	S, S	6+1	Unchanged	010 ₍₈₎ words \gtrsim	
Contents after operation	D D+1 : D+2n-2 D+2n-1		Contents of the register S Contents of the register S+1 Contents of the register S Contents of the register S+1		6 7 8 9 6 7 8 9 019435 019434
	Fla	ıg	Unchanged		

Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-74, F-74d, F-76, F-76w, F-76d

* Be sure to use even addresses for registers S and D. (Odd addresses such as 019003 etc. are prohibited to use.)

F-74d nXFR

Transfers "n" double words

Symb	ol		-74d n S D	[Explanation]	Instruction			
Functi	on	are	contents of the registers S to S+3 transferred to "n" double words sters headed by the registers D to 3.	004000 F-74d 010 009014 019416 NXFR 010 009014 019416	S T R 004000 F-74d 010 009014 019416			
Operat	Operation $\stackrel{S \text{ to } S+3}{\rightarrow D \text{ to } D+3, \dots, D+4n-4 \text{ to } D+4n-1}$			When the input condition 004000 changes from OFF to ON, the contents of the registers 009014				
			e range 000 to 377(8) 6 double words when 000(8))	to 009017 are transferred to a 010(8) double words registers headed by the registers 019416 to 019421.				
S		Use	range C *	013421.				
D		Use	range C *	009017 to 009014	019421 to 019416			
Condit	ion	Risi	ng edge of input signal (OFF to ON)		E 5 D 6 7 8 9			
	S to	S+3	Unchanged	$010_{(6)}, double words \approx \approx \approx \approx \approx \approx \approx$				
Contents after operation	D to E D+4i D+4i	n—3	Contents of the register S to S+3	(32 byte) 2	2 E 5 D 6 7 8 9 2 E 5 D 6 7 8 9 019455 to 019452			
	Fla	ıg	Unchanged					

 * Be sure to use even addresses for registers S and D.

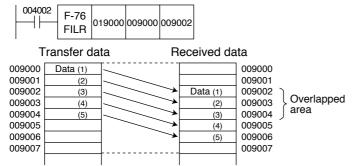
Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-70d, F-74, F-74w, F-76, F-76w, F-76d

F-76 FILF			ansfer "n" bytes in ba LR)	tch					
Symb	ol		76 S1 S2 D	[Expla	[Explanation]			Instr S T R F -76	uction 004000
Functi	on	regist - (S1)	fer data from "S ₂ to S ₂₊ (S ₁) -1" to er "D to D+(S ₁)-1" as a package. is the number of bytes specified by ster S ₁ .	009000 F-76 00000 700200 010200 300200					009000 ⊐00200 019200 nges from
Operat	ion	S2, …	·, S₂+ (S1) −1→D, …, D+ (S1) −1	OFF to	o ON, th	is instructio	n trans	sfers dat	a with the
S1		- The to 3	ange A contents of register S1 are 000(8) 77(8). If set to 000(8), 256 bytes of a are transferred.	registe with re	byte count specified by register 009000, from register ⊐00200 to the register area that begi with register 019200. - When the contents of register 009000 are 012 (D) (01				begins
S2		Use r	ange A		/	Before operation		Afte	r operation
D		Use r	ange A		⊐00200 ⊐00201 ⊐00202	$\begin{array}{c c} 0 & 1 \\ \hline 2 & 3 \\ \hline 4 & 5 \end{array}$	→ 0	19201	0 1 2 3 4 5
Condit	ion	Rising	edge of input signal (OFF to ON)		⊐00202 ⊐00203	6 7	i		6 7
	s	61	Unchanged	12 bytes <	⊐00204 ⊐00205	8 9 1 1	0	19205	8 9 1 1
	S2 to (S1)	o S2+ ─1	Unchanged		⊐00206 ⊐00207 ⊐00210	$\begin{array}{c c} 2 & 2 \\ \hline 3 & 3 \\ \hline 4 & 4 \end{array}$	0	19207	2 <u>2</u> 3 <u>3</u> 4 4
Contents after operation	D), +1, :	Contents of register S ₂ , Contents of register S ₂ +1,		⊐00210 ⊐00211 ⊐00212 _⊐00213	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	0	019211	5 <u>5</u> 6 <u>6</u> 7 7
	· ·	S1)—2 S1)—1	,		he conten	ts of register 0	09000 a		(003 (8)) : r operation
	Fla	ag	Unchanged	3 bytes	<pre></pre>				0 1
				0 Dytes	100201 200202	4 5			4 5

- Do not transfer the contact point area (file address 00001600 to 00001777(8) etc.) of the timer/counter of file number 0.



- If programmed in the above manner, the contents of 019003 and 019004 will be transferred to 00001600 and 00001601.
- Registers S₁, S₂, and D may be set to values which cause part of the source area to be overlapped with the destination area :



Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-70d, F-74w, F-74w, F-74d, F-76w, F-76d

F-76w **Transfer "n" words in batch** (FILR)

Symb	ol		Z ⁶ W S1 S2 D	[Explanation] Instruction S T R 004000 F -76w
Functi	on	regist - (S1)	fer data from "S ₂ to S ₂ +2(S ₁)-1" to er "D to D+2(S ₁)-1" as a package. is the number of words specified egister S1.	004000 F-76w ⊐00700 009000 019000 F-76w ⊐00700 009000 019000 019000 If the contents of location ⊐00700 are 040(8), this
Operat	ion		2+1, …, S2+ 2 (S1) −1 D,D+1, …, D+ 2 (S1) −1	instruction transfers 040(8) words (32 words) of data in registers 009000 to 009077 to registers
Sı		- The to 3	ange A contents of register S1 are 000 77(8). If set to 000(8), 256 words ata are transferred.	019000 to 019077. The contents of registers 009000 to 009077 are unaffected. Before operation After operation
S2		Use r	ange B *1	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$
D		Use r	ange B *2	
Condit	ion	Rising	edge of input signal (OFF to ON)	009003 2 8 2 8 019003
	S	1	Unchanged	
	S2 to (S1)) S2+ —1	Unchanged	
Contents after), +1,	Contents of register S ₂ , Contents of register S ₂₊₁ ,	009077 5 4 5 4 019077
operation		S1)—2 S1)—1	Contents of register S ₂₊₂ (S ₁) -2 Contents of register S ₂₊₂ (S ₁) -1	Transfer word count ⊐00700 040(8)
	Fla	ag	Unchanged	
*1 Be sur			en addresses for register S ₂ .	

*2 Be sure to use even addresses for register 52.

Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-70d, F-74, F-74w, F-74d, F-76, F-76d

F-76 FILF		_	ansfer "n" double wor LR)	ds in batch				
Symb	ol		76d S1 S2 D	[Explanation] Instruction S T R 004000				
Functi	on	regist - (S1)	fer data from "S ₂ to S ₂ +4(S ₁)-1" to er "D to D+4(S ₁)-1" as a package. is the number of double words that is ified by the register S ₁ .	F -76d □004000 F-76d □00700 009000 019000				
Operat	tion	S₂+ →D te	S2+3,, 4 (S1) -4 to S+4(S1)-1 o D+3,, 4(S1) -4 to D+4(S1)-1	instruction transfers 040 ₍₈₎ double words (128 bytes) of data in registers 009000 to 009177 to registers 019000 to 019177 (128 bytes). The contents of registers 009000 to 009077 are				
S1		- The to 3	ange A contents of register S1 are 000 77(8). If set to 000(8), 256 double ds of data are transferred.	unaffected. Before operation 009000 5 6 5 6				
S2		Use r	ange C *1	009001 7 8 7 8 0109001 009002 1 3 1 3 0109002				
D		Use r	ange C *2	009003 2 8 2 8 0109003				
Condit	ion	Rising	edge of input signal (OFF to ON)	009004 A B A B 0109004 009005 C D C D 0109005				
	s	51	Unchanged	009006 E F E F 0109006 009007 1 2 1 2 0109007				
		, to S₂ S1)−1	Unchanged					
Contents after operation	D+4 1	b D+3, to D+7 \vdots Contents of register S ₂ to S ₂ +3 Contents of register S ₂ +4 to S ₂ +7 \vdots Contents of register S ₂ +4 (S ₁) -4		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$				
		$\begin{array}{c} 4n-4 \\ +4n-1 \end{array} \begin{array}{c} \text{Contents of register } S_2+4 \\ \text{(S1)} & -1 \end{array}$ ag Unchanged		Transfer double words count ⊐00700 040(8)				

*1 Be sure to use even addresses for register S2.

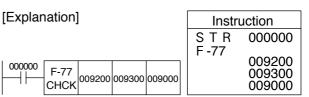
*2 Be sure to use even addresses for register D.

Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-70d, F-74, F-74w, F-74d, F-76, F-76w

Generates sum check code (CHecK Code)

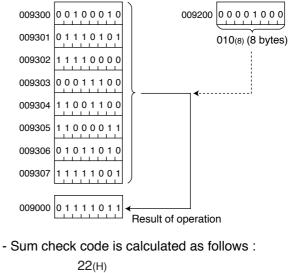
F-77 CHCK

	Symb	ol	_	F-77 CHCK S1	S2	D		[Explana	tion]	
	Function		cor (S1 the	nerates a s ntents of rec) –1 (the by contents o result in re	When the inp OFF to ON, the					
	Operat	ion	0—	Σ (S ₂ to S ₂	2 + (S 1)) —1)	→D	registers, for		
	S1		- T tc	e range A he contents 377(8). If s ata are tran	contents in registe If registe code for	er 009 r 009 regis				
	S 2		Use	e range A	and store	əd in				
	D		Use	e range A	009300	001				
	Conditi	ion	Rising edge of input signal (OFF to ON)					009301	011	
		S1 S2 to S2+		Unchanged				009302	111	
	Contents after operation			S2+ Unchanged				009303	000	
		n-	-1		009304	110				
	operation	D		Result of c	perati	on		009305	110	
		Fla	ıg	Unchange	Unchanged					
								1	1	



put condition of 000000 changes from this instruction generates a sum for register 009300 and subsequent the number of bytes specified by the egister 009200, and stores the result 09000.

9200 specifies 8 bytes : A sum check isters 009300 to 009307 is generated n register 009000:



22(H)	
75	

Obtains the complement of the

least significant 2 digits. 7BTwo's complement (100(H)-85(H))

The value of checksum code is 7B(H).

F0 1C CC C3 5A + F9 485

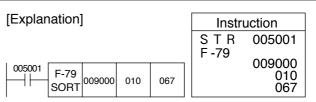
> 85 ----

F-78 CHK		Check (CHeck					
Symbo	ol	- F-78 CHK S1	S2	S3			[Explanation] Instruction S T R 000001 F -78
Functic	on	Generate a s contents of r $(S_1) - 1$ (by specified by S_1), compare check code (instruction) s set or reset t comparison - (S_1) is the n register S_1.	egister the nur the con e it with (genera stored i the flag result.	rs S2 the mber of n anothe ated by n regise accord	rough f bytes of regis er sum the F- ter S ₃ , ding to	S2+ ster 77 and the	$\begin{bmatrix} -78 & 009200 \\ 009300 \\ 009300 \\ 009000 \end{bmatrix}$ When the input condition of 000001 changes from OFF to ON, this instruction computes the sum check code for register 009300 and subsequent registers, for the number of bytes specified by the contents of register 009200, and compares it with another sum check code (generated using F-77) stored in register 009000.
Operatio	on	$ \begin{array}{c} [0-\Sigma \ (S_2 \text{ to } S_{2+} (S_1) -1)] \longleftrightarrow S_3 \\ Compared \\ \rightarrow Flag \end{array} $				əd	009300 0 0 1 0 0 0 1 0 009301 0 1 1 1 0 1 0 1 009301 0 1 1 1 0 0 0
S1		Use range A - The conter to 377(8). If data are tr	nts of r f set to	000(8),			- 009302 1 1 1 1 0 0 0 0 - 009303 0 0 0 1 1 1 0 0 - 009304 1 1 0 0 1 1 0 0
S2		Use range A					
D		Use range A					
Conditio	on	Rising edge o	of input	signal	(OFF to	ON)	
	S1	Unchange	d				After operation
	S2	Unchange	d				Checksum code is calculated. 009000
Contents after	S3	Unchange	d				Comparison Zero Carry Error Non-Carry
operation		Result of operation	Zero 007357	Carry 007356	Error 007355	Non-Carry 007354	result 007357007356b07355 007354
	Flag	No error	0	0	0	0	Matched 0 0 0
		Sumcheck error	0	0	1	0	Unmatched 1

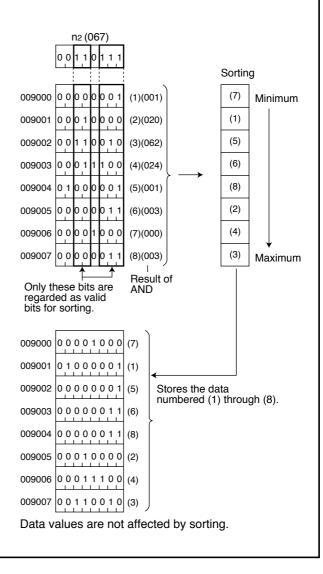
F-79 SORT

Sort register (1 byte) data

Symbo		-79 ORT	S	n1	n2			
Functio	S+n orde	1−1 er of	(1 by magr	rte eac nitude.	ch) in n2 is	sters S to ascending a mask value npared.		
Operatio	on	S < 3	S < S+1 < ···< S+n1-2 < S+n1-1					
S		Use range A						
N1			Use range 000 to 377 ₍₈₎ (256 bytes for 000)					
n2	- Va	lues	000 a		77 for	n2 have the ing all bits		
Conditio	Rising edge of input signal (OFF to ON)							
Contents after	Sto S+I	o n₁-1	Res (sor		ascen	ding o	rder)	
operation	Fl	ag	Uno	chang	jed			



When the input condition of 005001 changes from OFF to ON, this instruction ANDs the contents of registers 009000 to 009007 (8 bytes specified by n1) with 067(8) (the value specified by n2), compares and sorts the results in the ascending order of magnitude, and places them in register 009000 subsequent registers.



F-79w SORT

Sort register (1 word) data

Symbo	· -	F-79w SORT S n1 n2	[Explanation]	Instruction S T R 005001 F -79w			
Functio	n or n2	ort the details (1 word each) of gister S, S+1 to S+ $2n_1$ -2, S+ $2n_1$ -1 in der of small to large. this a mask value for the data bits to be compared.	$\begin{bmatrix} 005001 \\ F-79w \\ SORT \end{bmatrix} \xrightarrow{009000} 004 170077 \begin{bmatrix} -79w \\ 009000 \\ 170077 \\ 009000 \\ 170077 \\ 009000 \\ 004 \\ 170077 \\ 009000 \\ 004 \\ 170077 \\ 009000 \\ 000 \\$				
Operatio	on S•	S+1<···< S+ 2n1-2 · S+2n1-1	registers 009000 to 009007 (4 v n1) with 170077(8) (the contents				
S	Us	e range B *	compares and sorts the results in the ascending order of magnitude, and places them in register 009000, 009001 and subsequent registers.				
n1		e range 000 to 377 ₍₈₎ 56 words for 000)					
n2	- V h	e range 000000 to 177777(8) /alues 000000 and 177777 for n2 have the identical functions, making all bits valid.	$ \begin{array}{c} n_2 (170077) \\ 1 1 1 1 0 0 0 0 0 0 1 1 1 1 1 1 \\ 1 \sqrt{7} \sqrt{0} \sqrt{0} \sqrt{7} \sqrt{7} \end{array} $	Conting			
Conditio		sing edge of input signal (OFF to ON)	009001 009000 0 1 0 0 0 0 1 0 0 0 0 0 0 1 0 1 (1)	Sorting			
Contents after	S to S + 2n1-	1 Result (sorted in ascending order)	009003 009002				
operation	Flag	Unchanged	0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 1 (2)	(4)			
		n even address for register S. tc. are prohibited to use.)	009005 009004	→			

0000011111

0000000001

009007

009001

009003

009005

009007

000011

111111

009006

009000

009002

009004

009006

Only these bits are regarded as valid bits for sorting.

0 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 (3)

0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 (4)

0 1 0 0 0 0 1 0 0 0 0 0 1 0 1 (1)

0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 1 (2)

Data values are not affected by sorting.

(3)

(4)

(1)

(2)

Stores the data numbered (1) through (4).



Sort register (2 words) data

Symbol Sort s n no Function Sort the details (2 words each) of register S to S+3 to S+4n1 -4 to S+4n1 -4 to S+4n1 -4 to S+4n1 -1 to be compared. Sort To start of small to large, no is a mask value for the data bits to be compared. When the input condition of 005001 (Addition of 005001) (Addition of 005001 (Addition of 005000 (Addition of 00500 (Addition of 005000 (F-79d	[Explanation]				
Sort the details (2 words each) of register 51 05 ×3 to 5×41 −1 4 to 5×4n1 −1 in order of small to large, n 2 is a mask value for the data bits to be compared. Compared is a mask value for the data bits to be compared. Compared is a mask value for the data bits to be compared. Compared is a mask value for the data bits to secting order of magnitude, and places them in register 030000 to 009017 (4 double words of n2), compares and sorts the results in the ascending order of magnitude, and places them in register 030000 to 009000 to 009003 and subsequent registers. n1 Use range 000 to 377770; (256 double words for 000; all bits valid. Condition register 030000 to 009003 and subsequent register 030000 to 009003 and subsequent registers. n2 Use range 00000000000 and 37777777770; for ne have the identical functions, making all bits valid. To (27007407800) 101110000001111000000111100000001 2-477/0/0/0/7/4/0/0/7/6/0/0/0/1 To (27007407800) Sorting 000003 101110000001111000000111100000001 2-477/0/0/0/7/6/0/0/0/0/0/0/00000000000000	Symbol		S T R 005001				
Operation S to S + 3 + 4 + 1 + 4 + 10 + 3 + 4 + 11 + 4 + 10 + 3 + 4 + 11 + 4 + 10 + 3 + 4 + 11 + 4 + 10 + 3 + 10 + 3 + 1	Function	register S to S+3 to S+4n1-4 to $S+4n1-1$ in order of small to large. n2 is a mask value for the data bits to	005001 F-79d 009000 004 27007407600 004 27007407600 When the input condition of 005001 changes from OFF to ON, this instruction ANDs the contents of 009000 004 27007407600				
S Use range C -be sure to use an even address for register S: (Such as 019003 etc. are prohibited to use). of n2, compares and sorts the results in the according order of magnitude, and places them in register 009000 to 009003 and subsequent registers. m Use range 000 to 377(m) (256 double words for 000(m)) use range 0000000000 to 37777777777 for ne have the identical functions, making all bits valid. of n2, compares and sorts the results in the according order of magnitude, and places them in register 009000 to 009003 and subsequent Condition Rising edge of input signal (OFF to ON) after operation Sto Flag of no n, have and sorts the data not not not not not not not not not not	Operation	S to S+3 <····< S+4n1-4 to S+4n1-1	5				
ni (256 double words for 000(s)) nz Use range 0000000000 to 3777777777(s) - Values 0000000000 to 37777777777(s) - Values 00000000000 to 37777777777(s) - Values 00000000000 to 377777777777 Condition Rising edge of input signal (OFF to ON) Contents Statin - 1 stater ascending order operation Flag Unchanged na 2(27007407600) 10111000000111110000001111(000000101) (1) 000000000000000000000000000000000000	S	- Be sure to use an even address for register S.	of of n ₂), compares and sorts the results in the ascending order of magnitude, and places them in				
ne - Values 000000000 and 37777777777 for ne have the identical functions, making all bits valid. Condition Rising edge of input signal (OFF to ON) Contents after operation Store the data matching order operation ne (2 words each sorted in ascending order operation) contents after operation Store the data matching order operation 000007 000000 000007 000006 0000017 000001 000017 000001 000017 000001 0000017 000001 0000017 000001 0000017 000001 0000017 000001 0000017 000001 0000017 000001 0000017 000001 0000017 000001 0000017 000001 0000018 000001 00000019 000001 0000001 000001 0000017 000001 0000018 0000011 0000019 0000011 0000019 0000011 00000019 0000011	N1	e	registers.				
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	n2	- Values 0000000000 and 37777777777 for n ₂ have the identical functions, making					
$\begin{array}{c} \begin{array}{c} \begin{array}{c} \begin{array}{c} \begin{array}{c} \begin{array}{c} \begin{array}{c} \begin{array}{c} $	Condition	Rising edge of input signal (OFF to ON)					
operation Flag Unchanged n2 (27007407600) 1011100000111110000001111100000000000							
$\begin{array}{c} 1 0 1 1 1 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 1 0$		ag Unchanged					
	$\begin{array}{c} 1011110000001111110000001\\ 2 - \sqrt{7} \sqrt{6} \sqrt{6} \sqrt{6} \sqrt{6} \sqrt{6} \sqrt{6} \sqrt{6} 6$						
		· -					

Chapter 13 Application instructions (F-80 to F-173d)

I/O refresh (I/O ReFresh)

F-80 IORF

		F-80 Prover			[Explanation]						
Symb	ol	IORF RACK·SLOT					Instruction				
		г.						S T R 04000			
			<pre>kchanges </pre>	-				004000 F-80 RACK·SLOT IORF 0 2 F-80 R 0 S 2			
Functi	on	th	e I/O mod	ule spe				IORF 0 2 S 2			
			nd slot nun					When the input condition of 004000 is ON, this			
Operat	ion		put module utput modu					instruction exchanges data between the data memory and the 1st byte of the I/O module installed			
RACI	ĸ	Us	se range 0	to 7				in slot No. 2, rack No. 0.			
SLO	Г	Us	se range 0	to 7				- This instruction may be used any time repeatedly in a program. The rack, slot need not be assigned the same			
Conditi	on		hen the inp ot limited to			change	e)	 number. The I/O module subjected to I/O refreshing by this instruction is also subjected to data refresh during normal 			
	Input modu		Data mem update	ory		s error oo ta memo		I/O cycles.			
	Outp modu		Output sta update	ite	output update	state is i ed.	not	- The I/O relay which is being refreshed by this instruction cannot be forcibly set or reset from the hand-held programmer (JW-15PG) or other programming tool. (For			
Contents after			After I/O refresh	Zero 007357	Carry 007356	Error 007355	Non-Carry 007354	details of forced set/reset, see the instruction manual for the hand-held programmer.)			
operation			Bus error	0	0	1	0	- The addresses on an I/O module that are used for input			
	Flag	g	No module	0	0	0	1	interrupt (defined in system memory locations #0240 to #0243) cannot be used for I/O refreshing.			
			Transfer complete	0	1	0	0	- The upper limit of the slot number (SLOT) depends on the rack panel being used.			
			Non execution	0	0	0	0	- Non-execution occurs when the input condition is inactive. All flags are reset.			

F-82 IORF

Special I/O refresh (I/O ReFresh)

Sym	bol	F-82 IORF	SW				[Explanation]	Instruction S T R 004000
Func	tion	Exchanges data (16 bytes) and the control relay between the special I/O module specified by the SW (It is fixed number by rack no. and module no. switch) number and the PLC data memory.			al I/O is iodule	When input condition 004000 is bytes) in the special module for switch 3 and the PC's data are	rack 2, module no.	
Opera	ation	Special inp Special out					Data in special module specified by SW23 ⊐03460 to	203477
sv	v	Set up rai	igit: Racl	k no. (0) to 7)	128 points (16 bytes)	·
Condi	ition	When the (not limite				ange)		
	Specia Input modul Specia Output modul	e of data memory al Update t of output	state a during	are not	y and ou update error or rror	d		
		Result of operation	Zero 0007357	Carry 07356	Error 007355	Non-carry 007354		
Contents after operation		Special module error or bus error	0	0	1	0		
		Transfer complete	0	1	0	0		
	Flag	Missing transfer data or special module	0	0	0	1		
		During non -execution	0	0	0	0		

- This instruction can be used any number of times during program operation.

- This instruction is for the exchange of data between the special I/O module and the special I/O module relay. The number of data bytes transferred is 16.

- The special I/O module which is data refreshed with this instruction performs the data refresh operation also during an ordinary I/O cycle.
- If this instruction is executed when the data conversion for the special I/O module has not been completed, the non-carry flag (007354) turns ON.
- The error flag (007355) turns ON when a bus error is detected during I/O refresh or when a data error or bus error due to noise is detected.
- Non-execution refers to the state when the input condition is OFF. All flags turn OFF.
- Special I/O error refers to the error signal that is output from the special I/O module. Some module do not provide this signal.
- Special I/O module that is installed in remote I/O slave station (JW-21RS) can't refresh.

F-85
PRRD

Read from special I/O



Write to special I/O

Symbol			F-85 PRRD n1	SW,n ₂	D			-F-86 PRWD n1 D SW,n2		
Functior	i א פ r	nstr spec ack	ds n ₁ bytes ruction area cified by the no. and man n to D + n ₂ -	ι (n₂) of e SW (It odule no	the specting the specting the spectrum is fixed by the second sec	cial I/O r number 1) and lo	nodule by ads	Transfers the contents of n1 bytes starting with register D to the special I/O dedicated F instruction area (n2) of the special module specified by the SW (It is fixed number by rack no. and module no. switch).		
n1			range 000 to ified, the are					Use range 000 to 377(8) (When "000(8) " is specified, the area will be 256 bytes.)		
SW	ſ	∕ Up	range 00 to per digit: Ra wer digit: Mo	ck No. (0) to 7)		Use range 00 to 77 ₍₈₎ (Upper digit: Rack No. (0 to 7) Lower digit: Module No. switch (0 to 7)		
n2	5	Spe	cial I/O ded	licated F	⁻ instruc	tion area	a 0 to 3	Special I/O dedicated F instruction area 0 to 3		
D	ι	Jse	range A					Use range A		
Condition	n F	Risi	ng edge of i	input sig	nal (OF	F to ON)	Rising edge of input signal (OFF to ON)		
	n1	1	Unchange	d				Unchanged		
	SV	v	Unchange	d				Unchanged		
	na	2	Unchange	d				Unchanged		
	D to D+n1		Contents of	of n ₂				Unchanged		
			Result of operation	Zero 007357	Carry 007356	Error 007355	Non-carry 007354			
Contents after operation			No response from special module	0	0	1	0			
	Flag	g	When waiting for transfer	0	0	0	1	Same as left		
			When transfer completed	0	1	0	0			
			None of the above	0	0	0	0			

- The F-85 (PRRD) and F-86 (PRWD) instructions are used to read and write the parameter contents for the special I/O module (JW-21SU, JW-21PS, JW-21DU/22DU).

- Do not use the two instructions above when the special module is not being used. Or this may cause the malfunction.

- Each block for the special I/O dedicated F instruction area (n₂) is specified by a numeric value 0 to 3.

- Special I/O dedicated F instruction A block \rightarrow 0

- Special I/O dedicated F instruction B block \rightarrow 1

- Special I/O dedicated F instruction C block \rightarrow 2

- Special I/O dedicated F instruction D block \rightarrow 3

Each block contains 256 bytes.

RĔM	(REMark)	
Symbol		[Explanation] - Symbols and comments for JW-300SP are registered with "symbol/comment setting" by
Function	When printing ladder chart and instruction words, executed printing line comment on the ladder logic programming software (JW-300SP).	multipurpose programmer or ladder software. (Symbol: 16 half characters, Comments: 28 half characters) - When printing ladder chart, the multipurpose
Operation	NOP (This instruction does not cause the PC to perform an operation)	programmer or ladder chart prints symbol and comment contents. It does not print F-90 instruction in this case.
n	Use range 0000 to 3777(8)	If @ is registered at first character of the symbol,
Contents after operation	Data memory of flag etc. is unchanged.	it is paged and symbol comment contents are not printed. - When printing instructions, the multipurpose
		programmer or ladder chart prints each of F-90 instruction and symbol/comment contents.

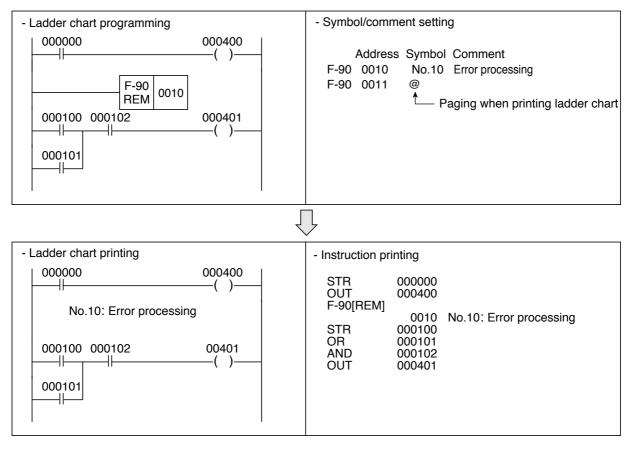
Even if @ is registered at first character of the symbol contents, paging is not executed and the

registered contents are printed.

[Example for use]

F-90

Remark



F-91 BCD8

Transfer BCD constant (8 digits)

Symb	ol	F-91 n1 n2 D	[Explanation]	Instruction S T R 004000		
Functio	on	Transfer an 8-digit BCD constant n ₁ , n ₂ (upper and lower 4 digits, respectively) to registers D to D+3.	004000 F-91 5438 9631 009000	F -91 5438 09631 09000		
Operat	ion	n1→D+3, D+2 n2→D+1, D	When the input condition of 00 from OFF to ON, this instruction			
N1		Use range 0000 to 9999 (FFFF $_{\mbox{(H)}}$ max.)	BCD constant 54389631 to registers 009000 to 009003.			
n2		Use range 0000 to 9999 (FFFF(H) max.)	After being transferred, the data will be the next value as BCD code.			
D		Use range C				
Conditi	on	Rising edge of input signal (OFF to ON)	009003 009002 009	001 009000		
Contents after operation	D to D+3		01010100001110001001			
	Flag	g Unchanged				

Resembled instructions: F-01, F01w, F-01d

F-97 BML8

Transfers a decimal constant (8 digits)

Symbo	ol		[Explanation]	Instruction S T R 004000
Functio	on	Transfer an 8-digit decimal constant n1, n2 (n1 x 10000+ n2) to registers D to D+3.	F-97 DML8 5438 9631 00900	F -97 5438 9631 009000
Operati	on	$n_1 x 10000 + n_2 \rightarrow D$ to D+3	When the input condition of (OFF to ON, This instruction	
n1		Use range 0000 to 9999	constant 54389631 to registe After being transferred, the c	ers 009000 to 009003.
n2		Use range 0000 to 9999	value as binary code.	
D		Use range C	009001	009000
Conditio	on	Rising edge of input signal (OFF to ON)		
Contents after operation	D to D+3 Flaç	(Decimal constant: 00000000 to 99999999)	009003 0_0_0_0_0_1_1_0 2 ²⁵ 2 ²⁴ 1+2+4+8+16+32+64+256+512+	009002 0 1 1 1 1 1 1 0 1 2 ²¹ 2 ²⁰ 2 ¹⁹ 2 ¹⁸ 2 ¹⁶ 2048+8192+16384+
			32768+65536+262144+524288 16777216+33554432=5438963	

- Binary code bits have the following weights:

	7	6	5	4	3	2	1	0
D	128 (2 ⁷)	64 (2 ⁶)	(2 ⁵) 32	16 (2 ⁴)	8 (2 ³)	4 (2 ²)	(2 ¹) 2	(2 ⁰)
D+1	32768 (2 ¹⁵)	16384 (2 ¹⁴)	8192 (2 ¹³)	4096 (2 ¹²)	2048 (2 ¹¹)	1024 (2 ¹⁰)	512 (2 ⁹)	256 (2 ⁸)
D+2	8388608 (2 ²³)	4194304 (2 ²²)	2097152 (2 ²¹)	1048576 (2 ²⁰)	524288 (2 ¹⁹)	262144 (2 ¹⁸)	131072 (2 ¹⁷)	65536 (2 ¹⁶)
D+3						67108864 (2 ²⁶)	33554432 (2 ²⁵)	16777216 (2 ²⁴)

F-100 ADRS		Set indirect address				
Symbo	ol	-F-100 S D	[Explanation]	Instruction S T R 004000 F -100		
Functio	on	Convert the file address of register S to file N and address n for the indirect address, and set register D to D+2.	004000 −−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−	009005 009400		
Operati	on	S→D, D+1, D+2	When input condition 004000 c ON, the JW300 converts regist	•		
S	S Use range A		address 00004005(8)) to file N and address n for indirect address, and sets them to registers 009400			
D		Use range C - Be sure to use even addresses for registers D.	to 009402. - Register after operation			
Conditi	on	Rising edge of input signal (OFF to ON)	009402 009401 009400			
	S	Unchanged	- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	L		
Contents	D	Lower address n of the register S	(00(H)) (004005(8))			
after operation	D+1	Lower address n of the register S	 Register 009005 is address 004005(8) of file 00(H). The set file N and address n will be a direct address when specifying indirect address. (@009400) 			
	D+2	P File number N of register S				
	Flag	Unchanged				

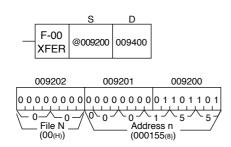
- As for "file N" and "address N" see "indirect address specification."

Reference

Indirect addressing:

Some data processing instructions allow the use of indirect addresses for the source and destination. For indirect addressing, the operation is carried out between the registers with file addresses specified by the 3-byte contents of the source and destination areas, each which begin with the source and destination registers, respectively. Indirect addressing is identified by symbol "@" prefixing the register address.

Example: This instruction transfers the contents of the registers addressed by the contents of registers 009200 to 009202 to register 009400.



In this example, file 00(H)'s file address, 000155(8), is ⊐00155 in byte address. Consequently address @009200 points to address ⊐00155.

⊐00155		009400
01101010	>	01101010

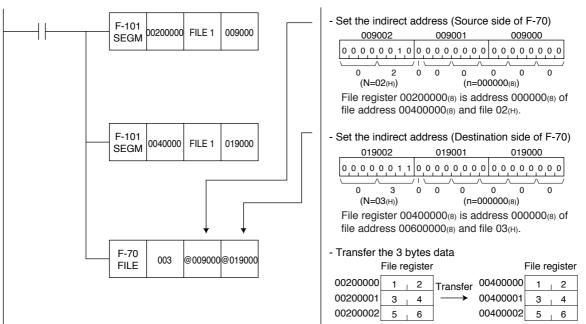
F-101 SEGM

Set indirect address

Symbo	bl	-F-101 n FILE F D	[Explanation] Instruction S T R 004500		
Functic	n	Convert file F and file address n to file N and address n that specifies indirect address, and set file N to register D+2 and address n to register D, D+1.	004500 F-101 377777 FILE 1 300536 When input condition 004500 changes from OFF to ON, the JW300 converts file register 377777(8) 6 377777		
Operatio	nn i	n (For indirect address)→D, D+1 File N (For indirect address)→D+2	(file number 1) to file N and address n for indirect address, and sets them to register ⊐00536 to		
n		Use range 00000000 to 3777777(8) (When this is the file register, it shall be a byte address)	⊐00540. - Register after operation		
F		0 (Data memory except file register) 1 (File register)	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$		
D		Use range C - Be sure to use even addresses for register D.	$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$		
Conditio	on F	Rising edge of input signal (OFF to ON)	- File register 377777(8) is address 177777(8) of file		
	D Lower digits of "n"		address 00577777(8) and file 02(H).		
Contents after	D+1	Upper digits of "n"	The file address thus set becomes an indirect address when the indirect address is set.		
operation	D+2	N	(@ 100536)		
	Flag				

- As for "file N" and "address n" see "indirect address specification."

E.g.: A program used to transmit the first 3 bytes, starting from 00200000(8), to the 3 bytes starting from 00400000(8) in file 1.



F-102 MRD

Read from the register of a direct address (1 byte)

Symbo	ol	- F-102 n FILE F D	[Explanation]	Instruction		
-		MRD		STR 005000		
Functio	on	The contents of the register in the file address "n" of the file number F are transferred to the register D.	F-102 F-102 MRD 00000536 FILE 1 019003 FILE 1 019003 FILE 1 019003			
Operati	on	n of file number $F \rightarrow D$	When the input condition 005000 changes from			
n	n Use range 00000000 to 3777777(8) (When this is the file register, it shall be a byte address)		OFF to ON, the contents of the file register 00000536(8) are transferred to the register 019003. File register			
F		0 (Data memory except file register) 1 (File register)	00000536	019003 → 0 1 1 0 1 0 1 1		
D		Use range A				
Conditio	n	Rising edge of input signal (OFF to ON)	Relation between F and r			
Conditio	511		Data memory F	Use range of "n"		
Contents	n	Unchanged	Other than file register 0	00000000 to 00177777(8)		
after	D	Contents of the register "n"	File register 1	00000000 to 37777777(8)		
operation	Flag	Unchanged	- Depending on the model of control module used (JW-3**CU), use range varies.			

Resembled instructions: F-102w, F-176

(JW-3[^]CU), use range varies.

Reads from the register of a direct address (1 word) F-102w MRD

Symbo	bl	Here and the second sec	[Explanation] [Explanation] [Instruction] S T R 005000 F-102w 00000536 FILE 1 019004 When the input condition 005000 changes from				
Functio	'n	The contents of the registers in the file addresses "n" and n+1 in the file number F are transferred to the registers D and D+1.					
Operatio	on	n of file number F, n+1→ D, D+1	OFF to ON, the 1 word registers 00000536 and				
n		Use range 0000000 to 37777776(8) (When this is the file register, it shall be a byte address) * Be sure to use even addresses for registers n. (Odd address such as 0000003 etc. are prohibited to use.)	transferred to the registers 019004 and 019005. File register 00000536 0 1 1 0 1 0 1 1 0 1 0 1 1 0 1 0 1 0 0 1 0				
F		0 (Data memory except file register) 1 (File register)					
		Use range B	Relation between F and n				
D		* Be sure to use even addresses for register D.	Data memory	F	Use range of "n"		
D		(Odd address such as 019003 etc. are prohibited to use.)	Other than file register	0	00000000 to 00177776(8)		
Conditio		, ,	File register	1	00000000 to 3777776(8)		
Conditio	n	Rising edge of input signal (OFF to ON)	- Depending on the mod				
	n	Unchanged	(JW-3**CU), use range	e varie	es.		
Contents after	D	Contents of the register "n"					
operation	D+1	Contents of the register n+1					
	Flaç	g Unchanged					

Resembled instructions: F-102, F-102d, F-176

F-102d MRD

Reads from the register of a direct address (2 words)

Symbo	bl	-F-102d n FILE F D	[Explanation]		Instruction S T R 005000 F -102d		
Functio	Function The contents of the registers in the file addresses "n" to n+3 in the file number F are transferred to the registers D to D+3.		When the input condition	00000536 FILE 1 019004			
Operatio	on	n to n+3 of file number $F \to D$ to D+3	OFF to ON, the 1 word				
n		Use range 00000000 to 37777774(8) (When this is the file register, it shall be a byte address) * Be sure to use even addresses for register n. (Odd address such as 0000003 etc. are prohibited to use.)	registers 00000536 and 00000541(8) (2 words data) are transferred to the registers 019004 and 019007. File register $00000536 \boxed{0 1 1 0 1 0 1 1} \longrightarrow 0110 1 0 1 1 019004$				
F		0 (When this is the file register, it shall be a byte address)1 (File register)	00000537 1 1 1 1 0 0 1 0 00000540 1 0 0 1 1 1 0 0 00000541 0 0 1 1 0 1 1 0		$ \rightarrow 11110010005 \rightarrow 10011100019006 \rightarrow 001101100019007 $		
D		Use range C * Be sure to use even addresses for register D. (Odd address such as 019003 etc. are prohibited to use.)	Relation between F and n Data memory F Use range of "n"				
Conditic	n	, ,	Data memory Other than file register	0	00000000 to 00177777(8)		
Conditio		Rising edge of input signal (OFF to ON)	File register	1	00000000 to 3777774(8)		
	n	Unchanged	- Depending on the mod				
	D	Contents of the register "n"	(JW-3**CU), use range	e vari	ies.		
Contents	D+1	Contents of the register n+1					
after operation	D+2	2 Contents of the register n+2					
	D+3	Contents of the register n+3					
	Flag	Unchanged					

Resembled instructions: F-102, F-102w, F-176

F-103 MWR

Write in the register of a direct address (1 byte)

Symbo	ol	-F-103 MWR S n FILE F	[Explanation]		Instruction S T R 005001 F -103	
Functic	on	The contents of the register S are transferred to the register in the file address "n" of the file number F.	005001 F-103 MWR b00001 00170000		1 00170000 FILE 1	
Operation $S \rightarrow n$ of file number F		$S \rightarrow n$ of file number F	When the input condition 005001 changes from OFF to ON, the contents of the register b00001			
S	S Use range A		are transferred to the file register 00170000(8).			
n		Use range 0000000 to 37777777(8) (When this is the file register, it shall be a byte address)	b00001		File register 00170000 → 1 0 1 1 0 0 1 0	
F		0 (Data memory except file register) 1 (File register)				
Conditio	on	Rising edge of input signal (OFF to ON)	Relation between F a	-		
	S	Unchanged	Data memory	F	Use range of "n"	
Contents			Other than file register	0	00000000 to 00177777(8)	
after	n	Contents of register S	File register	1	00000000 to 37777777(8)	
operation	Flag	g Unchanged	- Depending on the model of control module used			

Resembled instructions: F-103, F-103w, F-177

- Depending on the model of control module use (JW-3**CU), use range varies.

F-103w MWR Write in the register of a direct address (1 word)

Symb	ol	-F-103w MWR S FILE F FILE F	[Explanation]	Instruction S T R 005001 F -103w		
Function	Function The contents of the registers S and S+1 are transferred to the register in the file address "n" and n+1 of the file F.		0005001 F-103w b00000 00170000 FILE 1 00170000 MWR b00000 00170000 FILE 1 00170000 FILE When the input condition 005001 changes from 005001 changes from			
Operat	ion	S, S+1 \rightarrow n, n+1 of file number F	OFF to ON, the word on b00000 and b00001 ar		J. J	
S		Use range B - Be sure to use even addresses for register S. (Odd addresses such as 019003 etc. are prohibited to use.)	registers of 00170000			
n		Use range 00000000 to 37777776(8) (When this is the file register, it shall be a byte address) - Be sure to use even addresses for register n. (Odd addresses such as 00000003 etc. are prohibited to use.)	b00000 1 0 1 1 0 0 1 0 b00001 1 1 1 0 0 1 0 0	→ →	File register 1 0 1 1 0 0 1 0 1 1 1 0 0 1 0 00170000 1 1 1 0 0 1 0 0 00170001	
		0 (Data memory except file register)	Relation between F and n			
F		1 (File register)	Data memory	F	Use range of "n"	
Conditi	ion	Rising edge of input signal (OFF to ON)	Other than file register	0	00000000 to 00177776(8)	
	S, S+	1 Unchanged	File register	1	00000000 to 3777776(8)	
Contents after	n	Contents of the register S	- Depending on the model of control module used (JW-3**CU), use range varies.			
operation	n+1	Contents of the register S+1				
	Flag Unchanged					

Resembled instructions: F-103, F-103d, F-177

F-103d MWR

Write in the register of a direct address (2 words)

Symb	ol	Hereit Single Filler	[Explanation] Instruction S T R 005001 F -103d			
Functio	on	The contents of the registers S to S+3 are transferred to the register in the file address n to n+3 of the file number F.	005001 F-103d MWR 00000 00170000 FILE 1 B00000 FILE 1 B00000 00170000 FILE 1			
Operat	ion	S to S+3 \rightarrow n to n+3 of file number F	When the input condition 005001 changes from OFF to ON, the 2 words data contents of the			
S		Use range C - Be sure to use even addresses for register S. (Odd addresses such as 019003 etc. are prohibited to use.)	registers of 00170000 and 00170003(8) of the file register.			
n		Use range 0000000 to 37777774(8) (When this is the file register, it shall be a byte address) - Be sure to use even addresses for register n. (Odd addresses such as 00000003 etc. are prohibited to use.)	$\begin{array}{c} \text{File register} \\ \text{b00000} & 0 & 1 & 1 & 0 & 1 & 1 \\ \text{b00001} & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 1 & 1 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 1 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 1 & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 & 0 \\ \text{b00002} & 0 & 0 & 0 \\ \text{b0002} &$			
F		0 (Data memory except file register) 1 (File register)	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			
Conditi	on	Rising edge of input signal (OFF to ON)	Deletion between F and n			
	S to S	+3 Unchanged	Relation between F and n Data memory F Use range of "n"			
	n	Contents of the register S	Other than file register 0 00000000 to 00177774(8)			
Contents	n+1	Contents of the register S+1	File register 1 00000000 to 3777774(8)			
after	n+2	Contents of the register S+2	- Depending on the model of control module used			
operation	n+3	Contents of the register S+3	(JW-3**CU), use range varies.			
	Flag	Unchanged				

Resembled instructions: F-103, F-103w, F-177

F-112 NCMP

Compare "n" bytes (between 1-byte registers)

Sym	bol				[Explanation] Instruction S T R 004000 F -112				
FunctionCompare two data sets in size: compare data from register S1 to a specified number of bytes (detail of register S3) and data from register S2 to a specified number of bytes (detail of register S3).				$ \begin{array}{c c} & & & & & \\ \hline & & & & \\ \hline & & & & \\ \hline & & & &$					
Opera	ation	Result of comparison→f	ag	the 10 bytes contents of registers 009000 to					
S	1	Use range A		009011 with the magnitude of the contents of registers 009200 to 009211, and sets the result in					
S	2	Use range A			the non-carry flag , carry flag, or zero flag .				
S	S ₃ Use range A - The contents of register S ₃ are 000 to 377 ₍₈₎ . If set to 000 ₍₈₎ , 256 bytes of data are transferred.				Byte count for comparison 019300 0 0 0 0 1 0 1 0 012(8)				
Cond	lition	When the input signal is (not limited to OFF to Of		je)	009000 009200 009201 009201				
	S1 to S1+(S3)	Unchanged			009002 009202 009203 009203 009203				
Orintente	S2 to S2+(S3)	1 Unchanged			$\begin{array}{c ccccccccccccccccccccccccccccccccccc$				
Contents after operation	S3	Unchanged			009007 009207				
operation		Contents of Zero Carry register 007357 00735		Non-Carry 007354	009010 009210 009211				
		S1 to S1+ (S3) -1> S2 to S2+ (S3) -1 0 0	0	1					
	Flag	S1 to S1+ (S3) -1= S2 to S2+ (S3) -1 1 0	0	1	- 009000 to 009011>009200 to 009211→Non-carry ON - 009000 to 009011=009200 to 009211→Zero ON,				
		S1 to S1+ (S3) -1< S2 to S2+ (S3) -1 0 1	0	0	Non-carry ON - 009000 to 009011<009200 to 009211→Carry ON				
- This ir	nstructi	on is not capable of doub	le lengt	h l					

- This instruction is not capable of double length operations.

Resembled instructions: F-12, F-12w, F-12d, F-112w, F-112d, Fc12, Fc12w, Fc-12d

F-112w NCMP

Compare "n" words

Symbo	ol	-F-112w NCMP S1		[Explar	nation]			Instr S T R F -112v	uction 00200	0			
Functio	Function $ \begin{array}{c} \mbox{Compare two data sets in size:} \\ \mbox{compare data from register } S_1 \mbox{ to a} \\ \mbox{specified number of words (detail of register } S_3) \mbox{ and data from register } S_2 \\ \mbox{ to a specified number of words (detail of register } S_3). \end{array} $						$ \begin{array}{c} 002000 \\ \hline F-112w \\ \hline NCMP \end{array} 009000 0092000 019300 \\ \hline 0092000 \\ 019300 \\ \hline 0193$				0 0 (8)		
Operati	ion	Result of comparison→flag						is instructi /ords cont	tents of r	register	rs 00900	0 to	
S1		Use range B - Be sure to use	even ac	ddresse	s for reg	ister S₁.	009011 with the magnitude of the contents of registers 009200 to 009211, and sets the result in the non-carry flag, carry flag, or zero flag.						
S2		Use range B - Be sure to use	ister S₂.		-		•	r zero fla	g.				
S3		Use range A - The contents of set to 000(8), 28	377 ₍₈₎ . If nsferred.	019300 0 0 0 0 0 1 0 1 005(8) . 009000 009200									
Conditi	ion	When the inp (not limited to				e)	009	9001 9002 9003		_		009201 009202 009203	
S S	61 to 61+2(S3)-	1 Unchanged	b				ର୍ଚ୍ଚ <u>)</u> 009	9004		Compared		009204	words
S Sz	S2 to 2+2(S3)-	1 Unchanged	b				00 א 000 א	9005 9006		Log –		009205 009206	5 WO
Contents	S3	Unchanged	b					9007		F		009207	
after		Contents of Register	Zero 007357	Carry 007356	Error 007355	Non-Carry 007354		9010 9011		E		009210 009211	
		S1 to S1+2 (S3) -1> S2 to S2+2 (S3) -1	0	0	0	1) to 00901					N
	Flag	S1 to S1+2 (S3) -1= S2 to S2+2 (S3) -1	1	0	0	1	- 009000) to 00901 [.]	1=009200) to 009		o ON 1-carry O	N
		S1 to S1+2 (S3) -1< S2 to S2+2 (S3) -1	0	1	0	0	- 009000) to 00901 ⁻	1<009200) to 009		,	

- This instruction is not capable of double length operations.

Resembled instructions: F-12, F-12w, F-12d, F-112d, F-112d, Fc12w, Fc12d

F-112d NCMP

Compare "n" double words

Sym	bol							(planation] Instruction S T R 0020 F -112d	00		
Func	FunctionCompare two data sets in size: compare data from register S1 to a specified number of double words (detail of register S3) and data from register S2 to a specified number of words (detail of register S3).					$ \begin{array}{c c} & 009000 \\ \hline F-112d \\ \hline NCMP \end{array} 009000 \\ \hline 009200 \\ \hline 019300 \end{array} \end{array} $					
Operation Result of comparison \rightarrow flag						mag	2000 is ON, this instruction compares the agnitude of the 5 double words contents of				
St	S1 Use range C - Be sure to use even addresses for register S1.						registers 009000 to 009023 with the magnitude of the contents of registers 009200 to 009223, and				
Sz	2	Use range C - Be sure to use even addresses for register S ₂ .						sets the result in the non-carry flag, carry flag, or zero flag. 019300 0 0 0 0 0 1 0 1 005(8) 009000			
Sa	3	Use range A - The contents of register S ₃ are 000 to 377(8). If set to 000(8), 256 double words of data are transferred.									
Cond	ition	When the inp (not limited to				e)	double words (20 bytes)	009001 009201 009002 009202 009003 009203	double words (20 bytes)		
	S1 to S1+4(S3)-	Unchanged	t				ds (20	009003	ds (20		
	S2 to S2+4(S3)-	1 Unchanged	d				e word	009020 009220	e word		
Contents	S3	Unchanged	d l				louble	009021 009221 009222	louble		
after		Contents of register	Zero 007357	Carry 007356	Error 007355	Non-Carry 007354	5 d	009023	5 d		
-portailori		S1 to S1+4 (S3) -1> S2 to S2+4 (S3) -1	0	0	0	1	1	9000 to 009023>009200 to 009223→Non-carry (NC		
	Flag	S1 to S1+4 (S3) -1= S2 to S2+4 (S3) -1	1	0	0	1	- 009000 to 009023=009200 to 009223→Zero ON Non-carry ON				
		S1 to S1+4 (S3) -1< S2 to S2+4 (S3) -1	0	1	0	0	- 009	9000 to 009023<009200 to 009223→Carry ON			

- This instruction is not capable of double length operations.

Resembled instructions: F-12, F-12w, F-12d, F-112, F-112w, Fc12, Fc12w, Fc12d

F-116 DIV

Divides register value (BCD 8 digits) by another register value (BCD 8 digits) (DIVide) (Quotient has 8-digit integer part and 4 decimal fraction.)

Symbo	ol	-	F-116 DIV S1	S2	D					
Functio	on	re C th fr	Divide the 8 digits BCD contents of registers S_1 to S_{1+3} by the 8-digit BCD contents of registers S_2 to S_{2+3} , and store the quotient as an 8-digit integer and 4 fractional digits to the 6 bytes register area hat begins with register D.							
Operati	ion	(\$	S1 to S1+3)	÷ (S2	to S2+3	B) →D	to D+5			
S1		U	lse range C	*						
S2		U	Use range C *							
D		U	Use range H *							
Conditi	on	R	Rising edge of input signal (OFF to ON)							
	S1 to S1+3		Unchanged							
	S2 to S2+3									
	D to D+1		Quotient (4 fractional	digits)	content to S1+3	cted if th ts of reg 3 or S2 to 3 BCD co	isters S1 0 S2+3			
Contents after	D+2 1 D+5	to	Quotient (8-digit integ	ger)	the cor	ntents of re 00. (E				
operation			Contents of registers S1 to S+3 and S2 to S2+3	Zero 007357	Carry 007356	Error 007355	Non-Carry 007354			
	Flag		BCD code - Not BCD code - S2 to S2+3 value is 0	0	0	0 1	0			

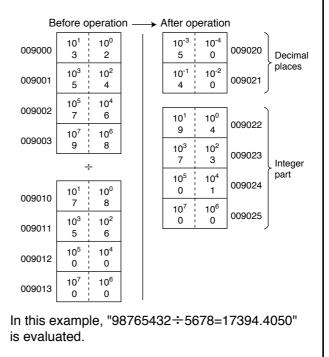
* Be sure to use even addresses for register S₁, S₂, and D.

- If the numerator is smaller than the denominator (S₁ to S₁₊₃<S₂ to S₂₊₃), the quotient (contents of D+2 to D+5) becomes 0. The 5th and subsequent decimal places in the contents of registers D and D+1 are truncated.

Resembled instructions: F-16, F-16d, Fc16d, Fc16d

[Explanation]	Instruction			
			S T R F -116	004001
004001 F-116 DIV	9000 009010	009020		009000 009010 009020

When the input condition of 004001 changes from OFF to ON, this instruction divides the 8-digit BCD contents of registers 009000 to 009003 by the 8-digit BCD contents of registers 009010 to 009013, and stores the 4 fractional digits of the quotient to registers 009020, 009021 and the 8-digit integer of the quotient to 009022 to 009025.



F-130 BIT→

Multiplexes bit (indirect address)

		-								
Symb	ol		S2				[Explanation]	Instruction S T R 004002		
$\begin{tabular}{lllllllllllllllllllllllllllllllllll$						$\begin{bmatrix} 004002 \\ H \\ $				
Operat	Operation Bit of $S_2 \langle S_1 \rangle \rightarrow Carry flag$						contents of a bit in the register 009001 determined by the lower 3 bits of the register			
S1	S ₁ Use range A						009000 are transferred to the carry flag (007356).			
S2	S2 Use range A									
Condit	ion	When the inp (not limited to	-			e)	7 6 5 4 3 2 1 0 009000 1 0 1			
	S1	Unchanged								
	S2	Unchanged					5			
Contents after		State of the specific bit	Zero 007357	Carry 007356	Error 007355	Non-Carry 007354	С	¥ arry flag		
operation	Flag	0 (OFF)	0	0	0	0	(1)	(007356)		
		1 (ON)	0	1	0		(1	507000)		

F-13 BIT-		Multiple	exes	bit	(dire	ect ac	ddress)	
Symb	ol						[Explanation] Instruction S T R 004010	
Function		The contents of a bit "n" the register S determined by the contents of the register S ₁ are transferred to the carry flag.					$\begin{bmatrix} 004010 & F-131 & 3 \\ HT \rightarrow & 3 \end{bmatrix} \begin{bmatrix} F-131 & 3 \\ 009000 \end{bmatrix}$ When the input condition 004010 is ON, the contents of a bit in the register 009000 are transferred to the	
Operation		Bit n of S \rightarrow Carry flag					carry flag (007356). 76543210 $n=3$	
n		Use range 0 to 7						
S		Use range A						
Condition		When the input signal is ON (not limited to OFF to ON change)					009000 0 1 0 1 0 0 1 0	
	S	Unchanged						
Contents after operation	Flag	State of the specific bit	Zero 007357	Carry 007356	Error 007355	Non-Carry 007354	Carry flag	
		0 (OFF)		0	0	0	(007356)	
		1 (ON)	- 0	1				

Sets/resets bit (indirect address) F-132 S/R (Set/Reset) (1) F-132 [Explanation] s D Instruction S/R (2) STR 004000 Symbol 004001 STR 004000 (1) Set / reset direction input F-132 ┥┝ F-132 004001 019000 019000 -00010 (2) Input condition S/R ⊐00010 ┥┝ A bit in the register D determined by When the input condition 004001 is ON, the bit of the contents of the register S (lower 3 □00010 determined by the lower 3 bits of the Function bits) is set or reset according to the register 019000 are set when 004000 is ON or set/reset command input of (1). reset when OFF. Operation State of (1) \rightarrow Bit of D \langle S \rangle 019000 110 S Use range A - 6 Use range A D 76543210 6 004000 is ON When the input condition (2) is ON 11110011 ⊐00010 1 0 1 1 0 0 1 1 Condition (not limited to OFF to ON change) (Bit 6 is set.) 6 S Unchanged Contents 10110011 004000 is OFF after D Only the specified bit changes (Bit 6 is reset.) operation Flag Unchanged



Sets/resets bit (direct address) (Set/Reset)

	(1)—F-1 (2)—S/		[Explanation] Instruction S T R 004002		
Symbol		et / reset direction input out condition	004002 004003 S/R 7 b01000	Š T R 004003 F-133 7 b01000	
Function		in the register D is set or reset ng to the set/reset command (1).	When the input condition of 004003 is ON, the bit 7 of the register b01000 is set when 004002 is ON or reset when OFF.		
Operation	State of	(1) → Bit D "n"	n=7 $7 6 5 4 3 2 1 0$ $b01000 0 1 0 1 1 0 1 1$ $004002 is ON$ $1 1 0 1 1 0 1 1$ $(7) 7 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 -$		
n	Use ran	ge 0 to 7			
D	Use ran	ge A			
Condition	When the input condition (2) is ON (not limited to OFF to ON change)		(Bit 7 is set.) 7 004002 is OEE		
Contents after	D Only the specified bit changes		004002 is OFF (Bit 7 is reset.)		
operation	Flag	Unchanged			

F-140 LABL	Set label (LABeL)			
Symbol	-F-140 LABL LBn			
Function	Use to set Jump destination for the F-141 (JMP) instruction. Branch destination for the F-142 (CALL) instruction. Branch destination for the F-148 (CAL+) instruction. Jump destination for the F-151 (JMP+) instruction. Branch destination for timer interrupt. Branch destination for input interrupt.			
n	Use range 0000 to 1777 ₍₈₎			
	L80002 0000005 STR 004000 009000 0000006 F-00 009000 000001 009000 004001 0000011 STR 000002 004001 0000012 A N D 004000 0000013 O U T 004001 0000014 F-140			

- Label numbers LB0000 to LB1777 may be used at any time, but the same number should not be used again within the program.

⁻ Use labels LB1353 to LB1357 for timer interrupts. For usage, see the description of system memory location #0240 and of the F-143 (return from subroutine) instruction.

⁻ Use labels LB1360 to LB1417 for input interrupts. For usage, see the description of system memory location #0241 to #0245 and of the F-143 (return from subroutine) instruction.

Jump to label

(JUMP)

F-141 JMP

Symbol	- F-141 JMP LBn						
Function	The control jumps to the label address LBn (F-140).						
n	Use range 0000 to 1777(8)						
Condition	When the input signal is ON (not to limited to OFF to ON change)						
[Explanation]	$\begin{array}{c c c c c c c c c c c c c c c c c c c $	ctions I. , the ctions					

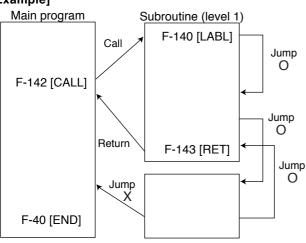
- The contents of the data memory are not affected after execution of an F-141 instruction.

- Execution may be done with F-41 (JCS) and F-42 (JCR) for the location indicated with an asterisk (*), the execution time may be saved if an F-141 (JMP) is used as it does not execute instruction down to F-140 (LABL).

- Same number may be used for the label of an F-141 instruction at any time.
- Because the control does not execute down to the jump address with an F-141 instruction, an F-40 (END instruction) will be disregarded even if there is an F-40 before the jump address.
- The jump address label (F-140) should be written in the program, in order to avoid malfunction that may occur if there is a jump address.
- Use labels LB1353 to LB1417 for timer interrupts (#0240) and for input interrupts (#0241 to #0245).
- When using the F-141 (JMP) instruction to cross multiple nested layers specified by the F-142 (CALL) and F-143 (RET) instructions.

The F-141 (JMP) instruction can only be used when the program jumps to an address inside the same subroutine, or to jump back to a start line in the same subroutine.

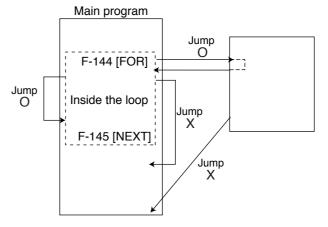
[Example]



- When using the F-141 (JMP) instruction to cross multiple nested layers specified by the F-144 (FOR) and F-145 (NEXT) instructions.

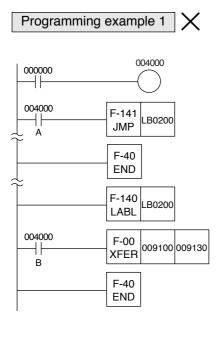
The F-141 (JMP) instruction can only be used when the program jumps to an address inside the same loop, or to jump back to a start line in the same loop.

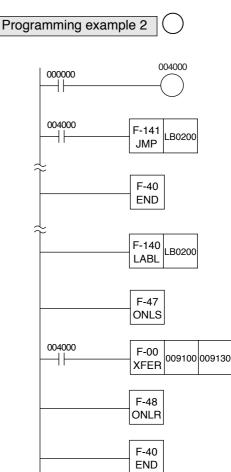




- When the following programs (ex. 1 and 2) are created using F-140 (LABL) and F-141 (JMP), F-00 (XFER) does not execute the target operation.

Operation wise, the F-141 is executed and the control jumps to the destination specified by the F-141 instruction, when the contact 004000 (A contact) turns active for both the examples. And, the F-00 is executed when a next contact 004000 (B contact) is active.





- Regarding the programming example 1, the F-141 is executed in the first cycle the contact 004000 (A contact) had turned active and the control jumps to the destination specified by the F-141. For the next contact 004000 (B contact) has been ON after the jump was made, the F-00 is executed, but, the F-00 is not executed in the second cycle after the contact 004000 (a contact) has turned ON even if it is ON, because the F-00 does not recognize the rising edge of the signal as the contents of the accumulator one scan cycle before do not match the current contents of the accumulator.
- Regarding the programming example 2, the F-00 after the jump is executed in the second cycle after the contact 004000 (A contact) has turned ON. This is because the level operation condition (F-17, F-48) is provided so as to execute the instruction after the jump at the time of ON.

Therefore, it has to be programmed like the example 2 in order to execute the program at every operational cycle after a jump.

Reference The F-00 instruction compares the contents of the previous contents of the accumulator with the current contents and executes the program when a rising edge is recognized.

- Even if the label to jump exists in a waiting status block, the JW300 executes operation of the jump destination.

Call labeled subroutine (CALL)

F-142 CALL

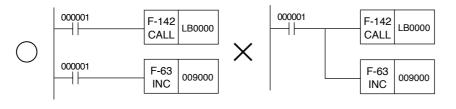


Return from subroutine (Return)

LBn F-143 RET umps to the subroutine of LBn e 0000 to 1777(8) lge of input signal (OFF to ON)		returns with an F-143 instruction.
e 0000 to 1777 ₍₈₎		returns with an F-143 instruction.
)	
lge of input signal (OFF to ON))	
Main program	Address 0000010 0000011 0000012 0000100 0000101 0000102 0000200 0000201 0000202	Instruction S T R 000000 F -142 LB0000 S T R 000001 F -142 LB0000 S T R 000002 F -142 LB0000
Subroutine program	$\begin{bmatrix} 0000300\\ 0000301\\ 0000302\\ \end{bmatrix} \begin{bmatrix} 0000315\\ 0000315\\ 0000315\\ 0000315\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 00000302\\ 000000302\\ 00000000\\ 0000000\\ 0000000\\ 0000000\\ 000000$	F -40 F -140 LB0000 F -143
	LL) and F-143 (RET) instruction	00000 0000102 000000 0000200 000000 0000201 00000 00003001 000000 00003001 00000000 0000301 000000000000000000000000000000000000

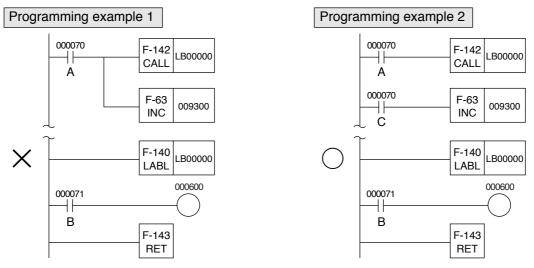
- In the above example, when the input condition 000000 changes from OFF to ON, the control jumps from the main program to the subroutine to execute the addresses 0000303 through 0000314, and returns to resume executing the main program at the address 0000013 when an F-143 instruction is met.
- When a subroutine is called, instructions within the subroutine are set to level operation with an ON state of signal.
- The following instructions may not be used within a subroutine. TMR, CNT, F-30 (MCS), F-31 (MCR), F-40 (END), F-41 (JCS), F-42 (JCR), F-44 (⊣↑⊢), F-45 (⊣↓⊢), F-47 (ONLS), F-48 (ONLR).
- The F-148 (CAL+) or F-149 (RETC) instruction can also be used for subroutine calls.
- The F-143 (RET) instruction is also used to program a subroutine for timer interrupts (set with #0240) or input interrupts (set with #0241 to #0245).
- Use labels LB1353 to LB1357 for timer interrupts.
- Use labels LB1360 to LB1417 for input interrupts.
- Main program must be created first before subroutine is created and it must be affixed with an F-40 (END) instruction at the end of the subroutine.
- The F-142 instruction cannot be used in the interrupt program.
- For the F-143 does not move to execute the F-142, but moves to execute the instruction that follows the F-142.

- A next step of an F-142 (CALL) instruction must begin with a contact input.



When the control returns from an F-143 (RET) instruction after executing an F-142 and its next instruction by a single contact point, a normal operation may not be expected because an input of a next step is dependent on the state within the subroutine.

- Either of programming examples 1 and 2 programmed using an F-142 (CALL) or F-143 (RET), has the same operation, but the F-63 (JNC) instruction of the programming example 1 will not operate with the condition for the contact 000070. For both the examples, the control jumps to subroutine specified by the F-142 has been executed when the contact 000070 (a contact) turned ON. And, It returns to the F-63 of the next step of the F-142 at the F-143, to execute the F-63 instruction.

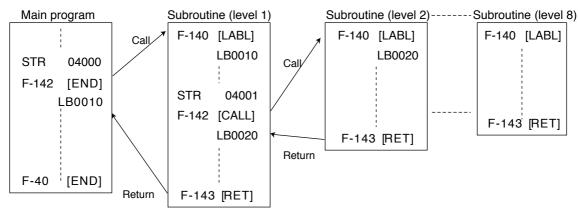


- Regarding the programming example 1, the F-142 is executed when the contact 000070 (A contact) has turned ON and the control jumps to the subroutine specified by the F-142. After executing the program steps until the F-143 is met, then moves to the step (F-63) that follows the F-142. In this event, the F-63 is executed under the condition of the contact 000071 (B contact). When the control with the F-143 returned to the step next to F-142, the F-63 may not be executed as programmed for execution as the F-63 operating condition is executed in the state of the accumulator immediately before, in this case, the state in the accumulator at the end of the subroutine.
- Regarding the programming example 2, execution does not take place under the state of the accumulator contents at the end of the subroutine after returning to the step next to the F-142, but, the execution takes place as programmed under the condition of the contact 000070 (C contact) after the return.

Therefore, it has to be as in the programming example 2 which the step immediately after the F-142 starts with a contact input.

- The F-142 (CALL) and F-143 (RET) instructions can be used to nest up to 8 subroutine calls.

[Example]

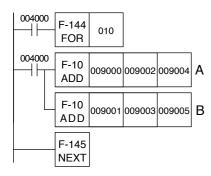


Resembled instructions: F-142, F-148, F-143, F-149

Sets loop count End of loop F-144 F-145 FOR NEXT (NEXT) (FOR) F-144 F-145 Symbol n FOR NEXT Function Repeats to execute the program "n" times between F-144 (FOR) and F-145 (NEXT). n Use range 000 to 377(8) (256 times if 000(8)) Condition Rising edge of input signal (OFF to ON) [Explanation] 000001 Instruction F-144 004 STR 000001 FOR F-144 000001 004 F-63 019000 000001 STR INC F-63 019000 F-145 F-145 NEXT 000001 When the input condition 000001 changes from OFF to ON. the F-63 (INC) instruction is repeated for four times. 019000 000(8) 004 010 014

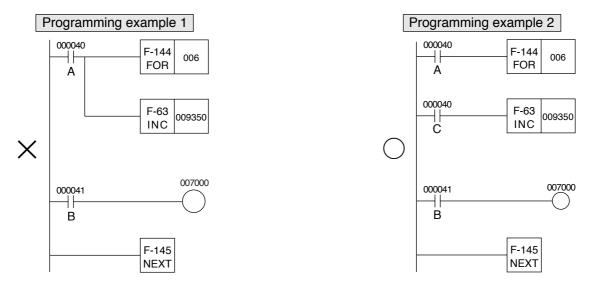
- The F-144 (FOR) must be used in conjunction with the F-145 (NEXT).

- When the F-144 (FOR) is executed, instructions between the F-144 (FOR) and F-145 (NEXT) are set ON level active.
- When the instruction is not execution, the contents of the data memory between the F-144 (FOR) and F-145 (NEXT) do not change.
- The following instruction may not be inserted between the F-144 (FOR) and F-145 (NEXT). TMR, CNT, F-30 (MCS), F-31 (MCR), F-40 (END), F-41 (JCS), F-42 (JCR), F-44 (⊣↑⊢), F-45 (⊣↓⊢), F-47 (ONLS), F-48 (ONLR), F-141 (JMP), F-144 (FOR), F-145 (NEXT)
- Instructions to be executed between the F-144 (FOR) and F-145 (NEXT) must be less as much as situation allows. The time required for the execution must be considered.
- Double-length operation is possible for the F-10 instruction. But the operation flag of the instruction of B will not affect the instruction of A.



- The same operation takes place when the follow kind of program is programmed using an F-144 (FOR) and F-145 (NEXT) instructions (examples 1 and 2). But, the F-63 instruction (INC) will not be executed for the example 1.

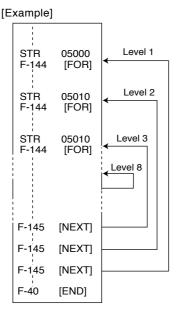
For both the examples 1 and 2, execution starts from the program steps that follows the F-144 through the F-145 for the times specified by the F-144, when the contact 000040 (A contact) has turned ON.



- Regarding the programming example 1, execution takes place from the program steps that follows the F-144 through the F-145 for the times specified by the F-144 after executing the F-144 and the F-63, when the contact 000040 (A contact) turned On, but the F-63 operates in the condition of the contact 000041 (B contact). Proper program execution may not be expected because the execution is done for the F-63 execution condition according to the previous contents of the accumulator (in the state with the contents of the accumulator stored immediately before the F-145, in this instance), when it returned to the step immediately before the F-144 with the F-145.
- Regarding the programming example 2, proper program execution is done as programmed as it is executed in the condition of the contact 000040 (C contact); not executing the program according to the accumulator contents before executing the F-145 after returning to the step immediately before the F-144 with the F-145.

Whereas, it is has to be programmed as in the example 2 for the step that immediately follows the F-144.

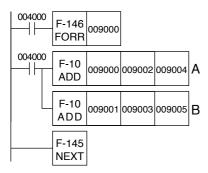
- To change the loop count, use the F-146 (FORR) instruction. To exit an execution loop, use the F-147 (EXIT) instruction.
- The F-144 (FOR) and F-145 (NEXT) instructions can be used to nest up to 8 subroutine calls.



F-146 FORR (FORR) Sets loop count register

Symbol	F-146 FORP		[Explanation]			
Function	instruction 146 (FO instruction	the execution of the series of ons in the loop between the F- RR) and F-145 (NEXT) ons the number of times d by the contents of register S.	$\begin{array}{c c c c c c c c c c c c c c c c c c c $			
S	377(8).	ge A ontents of register S is 000 to If set to 000(8), 256 times of re transferred.				
Condition	Rising edge of input signal (OFF to ON)		number of times specified by the contents of register 009000. (In the following example, the			
Contents after	S	Unchanged	register 009000 contains 4.)			
operation	Flag	Unchanged				
Resembled instructions: F-144			019000 000(8) 004	010 014		

- The F-146 (FORR) must be used in conjunction with the F-145 (NEXT).
- When the F-146 (FORR) is executed, instructions between the F-146 (FORR) and F-145 (NEXT) are set ON level active.
- When the loop is not executed at all, the contents of the data variables in the loop remain unaffected.
- To abort loop execution before the preset loop count is reached, use the F-147 (EXIT) instruction.
- The following instructions cannot be used in the loop between the F-146 (FORR) and F-145 (NEXT) instruction : TMR, CNT, F-30 (MCS), F-31 (MCR), F-40 (END), F-41 (JCS), F-42 (JCR), F-44 (⊣↑⊢), F-45 (⊣↓⊢), F-47 (ONLS), F-48 (ONLR)
- The F-146 (FORR) and F-145 (NEXT) instructions can be used to nest up to 8 subroutine calls. The using method is same as F-144 (FOR) instruction.
- Instructions to be executed between the F-146 (FORR) and F-145 (NEXT) must be less as much as situation allows. The time required for the execution must be considered.
- While the F-10 instruction is capable of double length operations, the flags for instruction "B" above do not affect instruction "A".



- For other notes, see those for the F-144 (FOR) instruction.



Symbol	F-147 E X I T								
Function	and F-145 (NEXT	ixit the execution loop that exists between (and including) the F-144 (FOR) or F-146 (FORR) nd F-145 (NEXT) instructions before the preset loop count is reached, and pass control to the instruction following the F-145 (NEXT) instruction.							
Condition	When the input si	/hen the input signal is OFF (not limited to ON to OFF change)							
[Explanation]									
000001 F-144 FORR 000001 F-63 INC 007357 F-147 EXIT Zero EXIT 7 F-145 NEXT 000001	377 019000 	Instr S T R F-144 F-63 S T R F-147 F-145 S T R O U T	uction 000001 377 019000 007357 000001 000100						
(NEXT) instr the instruction	ructions are executions following the F-	ted as NC 145 instru	OP instructi	e instructions between the F-147 (EXIT) and F-145 ions, control exits the loop, and control is passed to F-147 (EXIT) instruction does not affect the execution					

- The F-147 (EXIT) instruction must always be used in a loop between the F-144 (FOR) or F-146 (FORR) and F-145 (NEXT) instructions. Otherwise, execution may result in an error.

- F-147 may be used any number of times in a loop.

F-148 Call subroutine by Label (CAL+)

	<u> </u>	7					
Symbol	F-148 CAL+	LBn S	[Explanation]	Instru S T R F -148	uction 000010		
Function	subrouti whose v contents execution the F-14	brogram execution to the ne with the label (F-140) alue is the sum of LBn and the s of register S, and return on to the main program when 3 or F-149 instruction is ered in the subroutine.	When the input cor from OFF to ON, th subroutine having t sum of LB0100 and 009000.	ndition of 00 nis instruction the label wh	00010 cha on calls th nose value	e is the	
LBn	0000 to (LSD is a	1370(8) always treated as 0)	F-140 LB0100 C LAbel specified by the value of register 009000 of 000				
S	Use ran	ge A		LABL register 009000 of 000			
Condition	Rising e	dge of input signal (OFF to ON)	F-143 R E T				
Contents	S	Unchanged			paified by th	a valua of	
after operation	after operation Flag Unchanged		LABL LB0101	LABL LB0101 C register 009000 of 001			
Resembled ins	structions	: F-142	F-143 R E T				

- Register S can have values from 000 to 007(8). If it is set to 000(8) the subroutine call is made to label LBn (LBn+0=LBn).
- The LSD of LBn is always regarded as "0" (e. g. if LBn=0001 is specified, it is treated as LB=0000).
- The active input condition is applied to all instructions within a subroutine (not limited to OFF to ON change).
- The following instructions cannot be used in subroutines: TMR, CNT, F-30 (MCS), F-31 (MCR), F-40 (END), F-41 (JCS), F-42 (JCR), F-44 (⊣↑⊢), F-45 (⊣↓⊢), F-47 (ONLS), F-48 (ONLR)
- Be sure to write the main program before programming a subroutine, and place the F-40 (END) instruction at the last address of the main program.
- If a subroutine is given no label, the program will malfunction.
- The F-148 instruction cannot be used in interrupt program.
- The F-148 (CAL+) , F-143 (RET), and F-149 (RETC) instructions can be used to nest up to 8 subroutine calls. The using method is same as F-142 (CALL) instruction.
- See the notes for the F-142 (CALL) instruction.



Returns from subroutine (conditional)

Symbol	F-149 RETC						
Function	eturns control from the subroutine, which is called by the F-142 (CALL) or F-148 (CAL@) struction, to the main program.						
Condition	hen the input signal is OFF (not limited to ON to OFF change)						
[Explanation]							
- When the ir (RET) are e	Instruction F-140 LB0100 F-149 S T R 000002 F-143 F-149 F-143 F-143 return F-143 return F-143 nput condition of 000002 is OFF, the instructions between the F-149 (RETC) and F-143 xecuted as NOP instructions, and control is returned from F-143 to the main program. Input condition of 000002 is ON, the F-149 (RETC) instruction is no effect.						

- The F-149 instruction must always be used between the F-140 (LABL) and F-143 (RET) instructions. Otherwise, execution may result in an error.

- The F-149 instruction may be used any number of times in a subroutine.

Jump to labeled program address JMP+ (JuMP+)

F-151

Symbol	F-151 JMP+	Bn S		[Explan	ation]		STR	uction 004001
Function	program of which	address wit	ecution to the th the label (F-140) e sum of LBn and ster S.		The input conc	lition of 00		
LBn		0000 to 1370 ₍₈₎ (LSD is always treated as 0)			the label (F-1 f LB0030 and	40) of wh	ich value	is the
S	Use range A			019000.				
Condition		he input sign imited to OF	nal is ON F to ON change)		F-140 LABL LB0037		cified by th 19000 of 00	
Contents after	S	Unchanged	t		F-140 LB0031	Label spe	cified by th	e value of
operation	Flag	Unchanged	t		LABL LB0031	register 01	19000 of 00	J1(8)
Resembled ins	Resembled instructions: F-151				F-140 LABL LB0035		cified by th 19000 of 00	

- Register S can have values from 000(8) to 007(8). If it is set to 000(8) the jump is made to label LBn (LBn+0=LBn), and the JW300 shifts to label LBn (F-140) program.
- The LSD of LBn is always regarded as "0" (e. g. if LB0031 is specified, it is treated as LB0030).
- Execution of the F-151 instruction does not affect the contents of the data memory.
- The same label number may be used any number of times for the F-151 instruction.
- Since the F-151 instruction causes control to skip all instructions down to the labeled address, an F-40 (END instruction) placed before the labeled address will be ignored.
- Be sure to give a label (F-140) to the jump destination address. Otherwise the program will malfunction.
- When using the F-151 (JMP+) instruction to cross multiple nested layers specified by the F-142 (CALL) and F-143 (RET) instructions.

The F-151 (JMP+) instruction can only be used when the program jumps to an address inside the same subroutine, or to jump back to a start line in the same subroutine. The using method is same as F-141 (JMP) instruction.

- When using the F-151 (JMP+) instruction to cross multiple nested layers specified by the F-144 (FOR) and F-145 (NEXT) instructions.

The F-151 (JMP+) instruction can only be used when the program jumps to an address inside the same subroutine, or to jump back to a start line in the same subroutine. The using method is same as F-141 (JMP) instruction.

- See the notes for the F-141(JMP) instruction.



F-154

→BCD

Converts 8-digit BCD to 32-bit binary

Symbo	ol	F-153 → BIN	S D				[Explanation]	Instruction S T R 004000		
Functio	on	Convert BCI S to S+3 (4) and store the (4 bytes).	bytes)	into bir	nary co	des,	004000 F-153 009000 ⊐00000 →BIN 009000 ⊐00000 100000 When input condition 0040 to ON, the JW300 converts			
Operati	on	S, S+1, S+2,	S+3 ⊣	• D, D+	1, D+2,	, D+3	register 009000 to 009003 stores them into registers	•		
S		Use range C	*				Before operation \longrightarrow After operation			
D		Use range C *					Tens Ones			
Conditi	on	Rising edge of input signal (OFF to ON				to ON)	009000 0,1,1,0 0,1,0,0	$ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2^7 & 2^0 \end{bmatrix} $		
	S, S+1 S+2, S	1, Unshanged					6 4 Tens Ones			
	D	0 to 255					009001 1,0,0,0 1,0,0,0 8 8	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$		
	D+1	256 to 652			ed if the of S to :		o o Tens Ones	2.0 20		
Contents after	D+2	65536 to 16711680			CD cod		009002 0,0,0,1 0,0,0,0	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$		
operation	D+3	167772161 999999999	0					0 0 0 0 0 1 0 0 =00003		
		Contents of S, S+1, S+2, S+3	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	009003 0 1 1 0 0 1 1 1 6 7	2^{31} 2^{24}		
	Flag	BCD	0	0	0	- 0	BCD 67108864	BIN 2 ²⁶ =67108864		
		Not BCD code	Ŭ		1					

* Be sure to use even addresses for registers S and D. Resembled instructions: F-03, F-03w, F-03d, F-53

Converts 32-bit binary to 10-digit BCD

Symbo	ol		[Explanation]	Instruction S T R 004100			
Functio	on	Convert binary data in register S to S+3 (4 bytes: 32 bits) into BCD codes, and store them into registers D to D+4 (5 bytes).					
Operati	on	S to S+3 → D to D+4	019000 to 019003 (32 bits stores them into registers (
S		Use range C *	Before operation \longrightarrow After operation				
D		Use range H *	019000 0 0 0 0 0 0 0 0 0	Tens Ones 0 1 0 0 009000			
Conditio	on F	Rising edge of input signal (OFF to ON)	2 ⁷ 2 ⁰	4 8			
	S to S+3	Unchanged	$\begin{array}{c} 019001 \\ \hline 0 \\ 2^{15} \\ 2^{8} \end{array}$	Tens(3) Tens(2) 0,0,1,1,0,1,1,0 3,6 009001			
	D	Result (ones and tens)	2 2	5 0			
	D+1	Result (power 2 of tens and power 3 of tens)	019002	Tens(5) Tens(4) 0.1.0.0 1.0.0.0 009002			
Contents after	D+2	Result (power 4 of tens and power 5 of tens)	2 ²³ 2 ¹⁶ 019003	4 8			
operation	D+3	Result (power 6 of tens and power 7 of tens)	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	01000111 4 7			
	D+4	Result (power 8 of tens and power 9 of tens)	2 ³¹ =2147483648	Tens(9) Tens(8) 0,0,1,0 0,0,0,1 009004			
	Flag	Unchanged	BIN	² BCD ¹			

* Be sure to use even addresses for registers S and D. Resembled instructions: F-04, F-04w, F-04d, F-54, F-154

F-155 →SEC

Convert hours (BCD 4 digits), minutes (BCD 2 digits), and seconds (BCD 2 digits) into seconds (BCD 8 digits)

Symbo	ol	F-155 →SEC S	D				-	xplanation]			Instruction S T R 000005 F -155
Functic	on	Convert the registers S ((hour: lower upper 2 digit store the res bytes).	seconc 2 digits ts) to se	l), S+1(s) and s econd c	(minute S+3 (ho data, a	e), S+2 our: nd	- - - - - - - - - -				
Operati	on	\rightarrow D to D+3 (s Max. value = s	S (second), S+1 (minute), S+2, S+3 (hour \rightarrow D to D+3 (second) Max. value = 9999 hours 59 minutes 59 seconds \rightarrow 35999999 sec					to 1 second gisters 009	,	1 03 (all	
S		Use range C *						4	8	009001	Minute
D		Use range C	;*					3	6	009002	Hour
Conditio	on	Rising edge o	of input	signal (OFF to	ON)		0	1	009003)
	S to S+3		ed					0136 hours 21 seconds		sion	
Contents	D to D+3			S to S+3	ntents of r are not E econd exe	3CD or		0	1	009100	
after		Contents of registers S to S+3	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		2	5	009101	Second
operation		BCD code			0			4	9	009102	
	Flag	- Not BCD	0	0		0		0	0	009103	J
* Po ouro i		- Greater than max. value			1			00492501	seconds		

* Be sure to use even addresses for registers S and D. (Odd addresses such as 019003 etc. are prohibited to use.)

F-156 →HMS

Convert seconds (BCD 8 digits) into hours (BCD 4 digits), minutes (BCD 2 digits), and seconds (BCD 2 digits)

			•			.,			•		
Symb	ol	— F-156 →HMS S	D				[Exp	lanation]		S	Instruction STR 000010
Functio	on	Convert the digits) stored hour (BCD 4 digits), and st data, and sto D to D+3.	d in reg I digits) second	isters (, minut (BCD	S to S+ e (BCI 2 digits	-3 into D 2 S)	Whe	$\begin{array}{c} \underbrace{000010}_{\rightarrow HMS} \\ \hline 009000 \\ \hline 009100 \\ \hline 0091$			009100 0 changes from rts the second
Operati	ion	(S to S+3) → D+2 and D+3 Max. value =	S to S+3) \rightarrow D (second), D+1 (minute), +2 and D+3 (hour) lax. value = 35999999 sec. \rightarrow 999 hours 59 minutes 59 seconds se range C *						cond data,	and sto	res the result in
S		Use range C	*							-	
								5	6	009001	Second
D		Use range C	*					3	4	009002	
Conditi	on	Rising edge o	of input	signal (1	2	009003	
			n input	Signal				12345678 minutes			
	S to S+3		d					`	After conve	rsion	
Contents	D to D+3	Innoration	the c not E	D+3 are contents 3CD or e	of S to S xceed th	6+3 are ne max.		1	8	-	···· Second
after		Contents of	^{;.)} value Zero	e. (Does Carry	not calc Error	ulate.) Non-carry			l	-	
operation		registers S to S+3		007356				2	9	009102	Hour
		BCD code	0	0	0	0		3	4	009103)
	Flag	- Not BCD code - Greater than max. value	0	0	1	0		3429 hours 2 18 seconds	1 minutes		
	tous	e even addre	eene fr	or rogie	tore S	and D					

* Be sure to use even addresses for registers S and D. (Odd addresses such as 019003 etc. are prohibited to use.)

F-160 NSFF		Shift "n" bit register (N bit ShiFt Register)	
Symbo	bl	$ \begin{array}{c c} (1) & - & \\ (2) & - & \\ (3) & - & \\ (4) & - & \end{array} \begin{array}{c c} F-160 & \\ & & S1 & S2 & D \\ \hline & & & (1) \text{ Shift direction indication input} \\ (2) \text{ Data input} \\ (3) \text{ Shift input} \\ (4) \text{ Reset input} \end{array} $	[Explanation] Instruction
Functio	'n	Shift the specified bits area (register S_1) from the specified bit by register D (register S_2) to 1 bit higher or lower in the direction specified by shift direction indication input (1).	- The value of register 009000 is 016(D) bits
Operatio	on	- When the shift direction indication input (1) is ON: Bit count specified (MSB) by (S1) (LSB) by (S2) D+1 D (007356) Carry (MSB) by (S1) (LSB) by (S2) D+1 D Bit count Shifted 1 bit. Bit count Specified Bit specified by (S1) (LSB) by (S2) D+1 D D+1 D Carry (007356)	 The value of register 009001 is 003(D) bits Direction instruction input 000000 is ON Data input 0000001 is ON Reset input 000003 is OFF; in this example, the following shift operation occurs when input (000002) changes from OFF to ON. Before 019002 019001 019000 operation 10 1 1 0 1 1 1 1 0 0 0 0 1 1 0 0 1 0 0 1 1 0 0 1 0 0 1 1 0 0 0 0 1 1 0 0 0 0 1 1 0
S1		Use range A * - The contents of register S1 are 000 to 255(D). If set to 000(D), 256 bits of data are transferred.	After operation ₀₁₉₀₀₂ 019001 019000 1 0 1 1 0 0 1 1 1 1 0 0 0 0 1 1 0 0 1 0 1 0 1 1
S2		Use range A * - The contents of register S ₂ are 000 to 007(D).	Carry 007356 "ON"
D		Use range A *	
Conditic	on	When the reset input (4) is OFF, shift occurs at the rising edge (OFF to ON) of the shift input (3).	
Contents after -	Shif area		
operation	Flaç	input (4) 007357 007356 007355 007354	

* Be careful that the shift area set by S₁, S₂, and D does not enter the contact point areas of the timer/counter (file address 00001600 to 00001777₍₈₎ etc.) or the last address or more of the registers and file registers. Resembled instructions: F-60, F-60w, F-60d, Fc160

Fc160 NSFF		Shift "n" bit register (N bit ShiFt Register)				
Symbo	ol	(1) (2) (3) (4) (4) (1) Shift direction indication input (2) Data input (3) Shift input (4) Reset input	[Explanation] Instruction			
Functic	on	Shift the specified shift area (of which bit span is specified by n1 and whose LSB is specified by n2) of register D (bit n2), 1 bit in the direction specified by the shift direction indication input (1).	- The shift area is 016(D) bits			
Operatio		- When the shift direction indication input (1) is ON: Bit count specified by (n1) Carry (007356) Bit count specified Bit n2 (LSB) D+1 P D+1	 The data input is at bit 003(D) Direction instruction input 000000 is ON Data input 0000001 is ON Reset input 000003 is OFF in this example, the following shift operation occurs when input (000002) changes from OFF to ON. Before 019002 019001 019000 019000 019000 019000 After Carry Shifted 1 bit. Data input			
N1		Use range 000 to 255(D) - If set to 000(D), 256 bits of data are transferred.	operation 019002 019001 019000 1 0 1 1 0 0 1 1 1 1 0 0 0 0 1 1 0 0 1 0 1 0			
n2		Use range 000 to 007(D)	↓ Carry			
D		Use range A *	007356 ⁱⁱ ON"			
Conditio	on	When the reset input (4) is OFF, shift occurs at the rising edge (OFF to ON) of the shift input (3).				
Contents	Shif area	 (4) is OFF. - All bits OFF when reset input (4) is ON. 				
operation	Flaç	Reset input (4) Zero 007357 Carry 007356 Error 007356 Non-carry 007355 OFF 0 0 0 1 0 0 ON 0 0 0 0 0 0				

* Be careful that the shift area set by D is not enter the contact point areas of the timer/counter (file address 00001600 to 00001777(8) etc.) or the last address or more of the registers and file registers. Resembled instructions: F-60, F-60w, F-60d, F-160

F-161 NASF				al asynch nchronou		-	ter (N byt	e)
Symbo			-161 ASR D n	(1) Shift d (2) Shift i	lirection indication	on input		
Functio				ster D to D+(n-1) to a register (. ,	' shift data of ju	ist before or
		(1)	is ON. Shift or n register D to D+1 D+2	rection indicatio ccurs in the dire ward D+n. hift	•	(1) is OFF. S		he direction
Operatio	on		D+ (n-3) D+ (n-2) D+ (n-1) ▼			D-2 D-1 D		
	 Once a shift operation is executed, the contents of register (1 byte) are cleared to "0". No shift operation is performed if no register with all null data (00(H)) exists in the shift area. The shift area that is set by D and n shall not be before file address 00000000(8), in the contact point area of counter/timer (00001600 to 00001777(8) etc.), before file register 00000000(8), or after the last address of the register and file register. D Use range A n Use range 000(8) to 377(8) bytes (256 bytes for 000(8)) Shift occurs when the shift input (2) is ON (not limited to OFF to ON change) 							
D	U	Jse ra	nge A					
n	U	Jse ra	nge 000(8) to 3	377(8) bytes (25	6 bytes for 000	D(8))		
Conditio								s ON.
		Shift occurs when the shift input (2) is ON (not limited to OFE to ON change)			N/OFF			
			Before operation	After operation	Before operation	After operation	Before operation	After operation
Content just befo			DATA1	0	DATA1	Same as left	DATA1	Same as left
Contents 0	s of		0	DATA1	0	DATA2	Other than 0	Same as left
Content just afte			DATA2	Same as left	DATA2	0	DATA2	Same as left
	Non-C 0073	Carry 354	1 : D+ (n- 0 : D+ (n		,	n—1) =0 n—1) ≠0		1
Flag * (After	Erro 0073	or 155	C)		0	(0
operation)	Carr 0073	ry 556	0 : D+ (n 1 : D+ (n	/		n—1) =0 n—1) ≠0	(0
	Zero 0073	o 57	()		0	(C
[Explana	- 0 1 F-	-161 IASR 0		Instruction 5 T R 000000 5 T R 000000 5 -161 009000 006	1 000000 (1 0 000001 (2) ON ··· Shift d) ON ··· 00900		e shifted.

 * Carry flag (007356) turns ON only when data other than 0 shift to "D+(n-1)" or "D-(n-1)." Resembled instructions: F-61, F-61w, F-61d, F-161w, F-161d

 (1) When the values of 009003 and 009004 are 00(H): With the above input conditions, this instruction yields the following shift result :

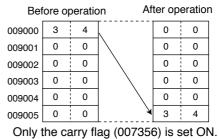
Be	efore c	operat	ion /	After o	peratio	on
009000	3	4		3	4	
009001	1	2		1	2	
009002	7	8		0	0	
009003	0	0		0	0	
009004	0	0		7	8	
009005	5	6		5	6	

Only the non-carry flag (007354) is set ON.

- To shift all 3 bytes in example (1) above, 3 scan cycles are needed:

009000	3	4	0	0
009001	1	2	0	0
009002	7	8	3	4
009003	5	6	1	2
009004	0	0	7	8
009005	0	0	5	6

(2) When the contents of 009001 to 009005 are all 00(H): A single execution of this instruction shifts the data from register D (009000) to 009005.



- If the data item that precedes another data item to be shifted is "00(H)", a single execution cycle causes more than one data item to be shifted:

3	4		0	0
0	0	\uparrow	3	4
5	6		0	0
0	0		0	0
0	0	\searrow	5	6
9	0		9	0
	0 5 0 0	0 0 5 6 0 0 0 0	0 0 5 6 0 0 0 0	0 0 5 6 0 0 0 0 5 5

F-161 NASF		directiona byte Asyr	-		-	ter (N wor	rds)					
Symbo		-161w D n	(1) Shift o (2) Shift i	direction indica nput	ation input							
Functio		In the range of "register D to D+2(n-1)+1" or "register D-2(n-1) to D+1," shift data of just before or after register (1 word) to a register (1 word) having data $0000(H)$.										
Operatio	(1) fro on - -	 When the shift direction indication input (1) is ON. Shift occurs in the direction from register D toward D+2n. 										
D	Use	range B										
n	Use	range 000 to 37	7(8) (256 words	for 000(8))								
Conditio	n Shift - Shif	occurs when the t operation is perf	shift input (2) is C formed in every s	DN (not limited to scan cycle as lor	o OFF to ON chang as the shift inp	ange) out (2) is ON.						
		(1)	ON	(1) (OFF	(1) OI	N/OFF					
		Before operation	After operation	Before operation	After operation	Before operation	After operation					
Contents of	just before 0	DATA1	0	DATA1	Same as left	DATA1	Same as left					
Conter	nts of 0	0	DATA1	0	DATA2	Other than 0	Same as left					
Contents of	just after C	DATA2	Same as left	DATA2	0	DATA2	Same as left					
	Non-Carry 007354		, D+2 (n−1) =0) , D+2 (n−1) ≠0)	$\begin{array}{c c} 1 & (D-2 & (n-1) + 1, \ D-2 & (n-1) = 0) \\ 0 & (D-2 & (n-1) + 1, \ D-2 & (n-1) \neq 0) \end{array} \begin{array}{c} 1 \end{array}$			1					
Flag * (After	Error 007355	()	0			0					
operation)	Carry 007356	0 (D+2 (n-1) +1 1 (D+2 (n-1) +1	, D+2 (n−1) =0) , D+2 (n−1) ≠0)	0 (D-2 (n-1) +1 1 (D-2 (n-1) +1	I, D−2 (n−1) =0) I, D−2 (n−1) ≠0)	0						
	Zero 007357		0		0		0					
$\begin{bmatrix} \text{Explanation} \end{bmatrix} \\ \hline \text{Instruction} \\ \text{S T R 000000} \\ \text{S T R 000001} \\ \text{F-161w} \\ 009000 \\ 005 \\ \hline \text{S T R 000001} \\ \text{F-161w} \\ 009000 \\ 005 \\ \hline \text{S T R 0000001} \\ \text{F-161w} \\ 009000 \\ 005 \\ \hline \text{S T R 0000001} \\ \text{F-161w} \\ 009000 \\ 005 \\ \hline \text{S T R 0000001} \\ \text{F-161w} \\ 009000 \\ 005 \\ \hline \text{S T R 0000001} \\ \text{F-161w} \\ 009000 \\ 005 \\ \hline \text{S T R 0000001} \\ \text{F-161w} \\ 009000 \\ 005 \\ \hline \text{S T R 0000001} \\ \hline \text{S T R 0000001} \\ \hline \text{S T R 0000001} \\ \text{S T R 0000001} \\ \hline \text{S T R 0000001} \\ \text{S T R 0000001} \\ \hline \text{S T R 000001} \\ \hline \text{S T R 000001} \\ \hline \text{S T R 0000001} \\ \hline \text{S T R 000001} \\ \hline S T R 00000$												
Or	nly the non-	carry flag (00735	4) is set ON.	C	Only the carry fla	g (007356) is se	t ON.					

* Carry flag (007356) turns ON only when data other than 0 shift to "D+2 (n-1)+1, D+2 (n-1)" or "D-2 (n-1)+1, D-2 (n-1)." Resembled instructions: F-61, F-61w, F-61d, F-161, F-161d

F-161d
NASRBi-directional asynchronous shift register (N double words)(N byte Asynchronous Shift Register)

			Jyic A	Зуі		s Shin Re	gister						
Symbo		1)— F-1 2)— NA	ASR D n (1) Shift direction indication input (2) Shift input										
Function			the range of "register D to D+4(n-1)+3" or "register D-4 (n-1) to D+3," shift data of just efore or after register (2 words) to a register (2 words) having data 0.										
		Shift o	When the shift direction input (1) is ON. - When the shift direction input (1) is OFF. Shift occurs in the direction from register D toward D+4n. - When the shift direction input (1) is OFF. Shift occurs in the direction from register D toward D+4n. - When the shift direction from register D toward D-4n.										
			D+3 D+7		D D+4 Shift	D-4(n-1)+ D-4(n-2)+		D-4(n-1) D-4(n-2) Shift					
Operatio	on	D+4(n D+4(n	-1)+3		∼ D+4(n-2) D+4(n-1)	D D+	+3	D-4 D					
		- No - The poii	 Once a shift operation is executed, the contents of registers (2 words) is cleared to "0". No shift operation is performed if no register with all null data 0 (2 words) exists in the shift area. The shift area that is set by D and n shall not be before file address 0000000(8), in the contact point area of counter/timer (00001600 to 00001777(8) etc.), before file register 00000000(8), or after the last address of the register and file register. 										
D		Use ra	inge C										
n		Use ra	ange 000	to 37	7(8) double wor	rds (256 double	e words for 000)(8))					
Conditio		Shift o	ccurs whe	n the s	shift input (2) is (ON (not limited to	o OFF to ON chang as the shift inp	ange)					
				(1)	-	,	OFF		N/OFF				
		Before After operation operation			After operation	Before operation	After operation	Before operation	After operation				
Contents of	just be	efore 0	DATA	\1	0	DATA1	Same as left	DATA1	Same as left				
Conter	nts of	0	0		DATA1	0	DATA2	Other than 0	Same as left				
Contents o	-		DATA	12	Same as left	DATA2	0	DATA2	Same as left				
	Non- 007	Carry 7354			D+4(n-1)+3 =0) $D+4(n-1)+3 \neq 0)$	$\begin{array}{c} 1 \ (D-4 \ (n-1) \ to \ D-4 \ (n-1)+3 = 0) \\ 0 \ (D-4 \ (n-1) \ to \ D-4 \ (n-1)+3 \neq 0) \end{array} \begin{array}{c} 1 \end{array}$			1				
Flag * (After		ror 355		()		0	0					
operation)	Ca 007	rry 356		,	D+4(n-1)+3 =0) $D+4(n-1)+3 \neq 0)$		D-4 (n-1)+3 =0) D-4 (n-1)+3 ≠0)	0					
	Ze 007	ero 357		()		0	0					
	[Explanation] 000000 000001 F-161w NASR 009000 005 Instruction S T R 000000 S T R 000000 F-161d 009000 005						1) ON 00900 2) ON Shift d		re shifted.				
When t the follo				to 00)9017 are 0: W	/ith the above i	nput conditions	s, this instructio	on yields				
		-	Before	opera	ation	After ope	ration						
009000		ŀ		4 5			5 6 7 8						
009004		ŀ	9 A B 0 0 0	C D 0 0									
		F		0 0			D E F 1						
009020	009014 to 009017 0												
	Only the non-carry flag (007354) is set ON.												

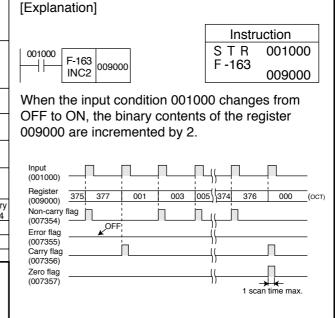
* Carry flag (007356) turns ON only when data other than 0 shift to "D+4(n-1) to D+4(n-1)+3" or "D-4(n-1) to D-4(n-1)+3." Resembled instructions: F-61, F-61w, F-61d, F-161, F-161w

F-163 INC2

Adds binary (+2) counter (1 byte)

Symb	ol	- F-163 D INC2 D							
Funct	ion	Binary contents of the register D are incremented by 2.							
Opera	tion	$\langle D \rangle$ +2 \rightarrow D							
D		Use range A							
Condit	tion	Rising edge of input signal (OFF to ON)							
Contonto	D	Result							
Contents after		Result (OCT)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354			
operation	Flag	376→000	1	1	0	0			
	· iug	377→001	0	1	0	0			
		Other than above	0	0	0	1			
Resembled instructions: F-63, F-63w, F-63d, F-163w, F-163d, F-263, F-263w,									

F-263d



F-163 INC2		Adds bi	nary	y (+2	2) co	unte	r (1 word)				
Symb	ol	F-163w D					[Explanation]				
Functi	on	Binary conter D+1 are incre		-		and	001000 F-163w 009000 S T R 001000 INC2 009000 F -163w 009000				
Operat	tion	⟨D, D+1⟩ +2 -	→ D, [D+1			When the input condition 001000 changes from OFF to ON, the 16-bit binary contents of the				
D		Use range B - Be sure to use (Odd addresse prohibited to u	es such			,	registers 009000 and 009001 are incremented by 2.				
Condit	ion	Rising edge of	f input	signal	(OFF to	ON)	Register (009000) <u>375 377 001 003 005</u> 374 376 000 (OCT)				
	D	Lower digit	ts of re	sult			(009001) <u>377 377 000 000 000</u> Non-carry flag (007354)				
Contents D+		I Upper digit	ts of re	esult			(007355)				
after operation		Result (OCT)	Zero 007357	Carry 007356		Non-carry 007354	(007356) Zero flag				
•	Flag	177776→00000	1	1	0	0	(007357)				
		177777→00001 Other than above	0	<u>1</u> 0	0	0	i scan une max.				
Decemble			.	Ť	, v		162d E 262 E 262w E 262d				

Resembled instructions: F-63, F-63w, F-63d, F-163d, F-163d, F-263, F-263w, F-263d



Adds binary (+2) counter (2 words)

Symb	ol	F-163d INC2 D					[Explanation]			
Functi	on	Binary conte D+3 are incr		•		to	001000 F-163d 009000 S T R 001000 INC2 009000 F-163d 009000			
Operat	tion	$\langle D \text{ to } D+3 \rangle$ -	-2 → D	to D+3	3		When the input condition 001000 changes from OFF to ON, the 32-bit binary contents of the			
D		Use range C - Be sure to us (Odd address prohibited to	e even a ses such				registers 009000 to 009003 are incremented by 2.			
Condit	ion	Rising edge	of input	signal	(OFF to	ON)	Register (019000 to 019003) <u>37777777774 3777777776 00000000000 0000000002</u> (OCT)			
	D to D	+3 D: Lower	digits, l	D+3: U	pper di	gits	Non-carry flag			
		Result (OCT)	Zero 007357	Carry 007356		Non-carry 007354	Error flag			
Contents after	Flag	3777777776 →0000000000	1	1	0	0	(007355) Carry flag (007356)			
operation		37777777777 →00000000001	0	1	0	0	Zero flag (007357)			
		Other than above	0	0	0	1	1 scan time max.			

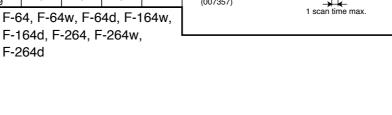
Resembled instructions: F-63, F-63w, F-63d, F-163, F-163w, F-263, F-263w, F-263d



Subtracts binary (-2) counter (1 byte)

Symbo	ol	— F-164 DEC2 D					[Explanation]
							Instruction
		Binary conte	nts of t	he reai	ister D	are	001001 STR 001001
Function	on	decremented					F-164 009020
		deorementer	1 Oy 2.				
Operat	ion	$\langle D \rangle - 2 \rightarrow D$)				When the input condition 001001 changes from
							OFF to ON, the binary contents of the register
D		Use range A	1				009020 are decremented by 2.
Conditi	on	Rising edge of	of input	signal	(OFF to	ON)	
	D	Result					(001001)
			Zero	Carry	Error	Non-carry	(009020) 006 004 002 000 376 374) 001 377 (001)
Contents		Result (OCT)	007357	007356			Non-carry flag
after	-	002→000	1	0	0	1	Error flag
operation	Flag	001→377 000→376	0	1	0	0	(007356) // // // // // // // // // // // // //
		Other than above	0	0	0	1	Zero flag (007357)
Resemble	ed ins	tructions: F-6	4, F-64	1w, F-6	4d, F-1	164w,	1 scan time max.
		F-1	64d F	-264. F	-264w	. ,	

F-264d



Subtracts binary (-2) counter (1 word) F-164w DEC2

Symbo	ol	F-164w D					[Explanation]				
						Instruction					
Functio	on	Binary conte D+1 are dec		-) and	001001 F-164w 009020 DEC2 009020 F-164w				
Operati	ion	$\langle D, D+1 \rangle -2$	2 → D,	D+1			When the input condition 001001 changes from OFF to ON, the 16-bit binary contents of the				
D		Use range E - Be sure to us (Odd address prohibited to	e even a ses such		•	0	registers 009020 and 009021 are decremented b				
Conditi	on	Rising edge o	of input	signal (OFF to	ON)	Register (009020) 006 004 002 000 376 374 001 377 (OCT) (009021) 000 000 000 377 377 (OCT)				
	D	Lower dig	its of re	esult			Non-carry flag				
	D+1	Upper dig	its of re	esult			(007354) Error flag (007355) OFF				
Contents		Result	Zero 007357	Carry 007356		Non-carry 007354	Carry flag (007356)				
after		000002→00000	1	0	0	1	Zero flag				
operation	Flaç	000001→17777 000000→17776	0	1	0	0	(007357)				
		Other than above	0	0	0	1					

Resembled instructions: F-64, F-64w, F-64d, F-164, F-164d, F-264w, F264d

F-164d DEC2

Subtracts binary (-2) counter (2 words)

Symb	ol	F-164d DEC2					[Explanation]			
Functi	on	Binary conte D+3 are dec		•		D to	001001 F-164d S T R 001001 F-164d 009020 F-164d 009020			
Operat	tion	$\langle D \text{ to } D+3 \rangle$ -	-2 → C) to D+	3		When the input condition 001001 changes from OFF to ON, the 32-bit binary contents of the			
D		Use range C - Be sure to us (Odd address prohibited to	e even a ses such				registers 009020 to 009023 are decremented by 2			
Condit	ion	Rising edge o	of input	signal (OFF to	ON)	Register (009020 to 000000000004 00000000002 00000000000 3777777776 (OCT) 009023) 00000000004 000000000000000000000000000000000000			
	D to D	+3 D : Lower	digits,	D+3 : I	Upper o	digits	Non-carry flag			
		Result (OCT)	Zero 007357	Carry 007356		Non-carry 007354	(007354)			
Contents		0000000002→ 00000000000	1	0	0	1	Carry flag			
after operation	Flag	0000000001→ 3777777777 0000000000→ 3777777776	0	1	0	0	Zero flag (007357) 1 scan time max.			
		Other than above	0	0	0	1				

Resembled instructions: F-64, F-64w, F-64d, F-164, F-164w, F-264, F-264w, F-264d

F-17 INS		Inserts of data (1 byte) (INSert)				
Symb	ool		[Explanation] Instruction S T R 004000 F -170			
Functi	ion	Insert the contents of register S into register address D_{1+} (D_{2+1}) in the shift register area whose first register address is D_{1} and of which byte count is given by the contents of register D_{2} .	 GOMMOND F-170 009000 009100 009200 Calculates when input condition 004000 changes from OFF to ON. The contents (55(H)) of registers 009000 is inserted. The top address is register 009100. 			
Operat	tion	D1 (Top address) D1+ (D2+1) (Insertion address) D1+ (D2)-1 (End address) - The shift register area that is set by D1 and D2 shall not be in the contact point area of counter/timer (file address 00001600 to 00001777(8) etc.), after the last address of the register and file register. - If the end address is not 00(H), or D2 ≤ D2+1, an error flag (007355) turns ON and JW300 does not operate. - If the contents of the last address are not 00(H), the instruction is not executed even if the contents of intermediate addresses are 00(H). Intermediate addresses 00(H) are treated as data.	- Address to insert is one added $005_{(8)}$ bytes (contents of 009201) to register 009100 . - The end address is $010_{(D)}$ byte (contents of 009200) for 009100 . The top address is included in the byte count. - When the contents of the last address are $00_{(H)}$, the data at and beyond the address of insertion are shifted down 1 byte, and the contents ($55_{(H)}$) of address 009000 are stored in the insertion address. $009000 \underbrace{55_{(H)}}_{009201} \underbrace{10_{(D)}}_{005_{(8)}} -10$ -byte from 009100 $009200 \underbrace{010_{(D)}}_{005_{(8)}} -10$ -byte from 009100 $009200 \underbrace{010_{(D)}}_{009201} \underbrace{12}_{005_{(8)}} -10$ -byte from 009100 $009100 \underbrace{12}_{000} -12$ Insert at 5th bytes from 009100 $00 009101 \underbrace{12}_{009100} -12$ $10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\ 10 \\$			
S		Use range A	Address 009111 00 009111 00 009111			
D1		Use range A Use range B - The contents of register D ₂ are 000 to 377 ₍₈₎ . 000 ₍₈₎ specifies 256 bytes. - The contents of register D ₂₊₁ are 000 to 377 ₍₈₎ . 000 ₍₈₎ specifies the top address.				
Condit		Rising edge of input signal (OFF to ON)				
	S D2,	Unchanged				
Contents after operation	D1 to D1+(D2	1 byte shifted and value of S inserted. - Does not chanded when the contents of the end address are not "00(H)" or D2 ≤ Contents of end address are not "00(H)" or D2 ≤ Contents of end address Contents of end address Contents of end address 007357 007356 007354 00				

F-17 INS		Inserts data (1 word) (INSert)
Sym	bol	F-170w S D1 D2 INS S D1 D2 [Explanation] INS S T
Func	tion	Insert the contents of register S, S+1 into register address D1+2 (D2+1), D1+2(D2+1)+1 in the shift register area whose first register address is D1 and of which word count is given by the contents of register D2. $\begin{array}{c} 004000 \\ \hline F-170w \\ \hline INS \\ 009000 \\ \hline INS \\ 009100 \\ 009200 \\ \hline O09200 \\ \hline$
Opera	ation	D1 (Top address) D1+2 (D2+1) (Insertion address) D1+2 (D2+1) (Insertion address) D1+2 (D2+1) (Insertion address) D1+2 (D2+1) (Insertion address) D1+2 (D2)-1 (Insertion data address) The shift register area that is set by D1 and D2 shall not be in the contact point area of counter/timer (file address 00001600 to 00001777(8) etc.), after the last address of the register. - If the end address is not 0000(H), or D2 \leq D2+1, an error flag (007355) turns ON and JW300 does not operate. - If the contents of the last address are not 0000(H), the instruction is not executed even if the contents of intermediate addresses are 0000(H) are treated as data. - If the contents of the last address are not 0000(H), the instruction is not executed even if the contents of intermediate addresses are 0000(H) are treated as data. - If the contents of the last address are not 0000(H), the instruction is not executed even if the contents of intermediate addresses are 0000(H) are treated as data. - If the contents of the last address are not 0000(H), the instruction is not executed even if the contents of intermediate addresses are 0000(H) are treated as data. - If the contents of the last address are not 0000(H), the instruction is not executed even if the contents of intermediate addresses are 0000(H) are treated as data. - If the contents of the last address are not 0000(H) are treated as data. - If the contents of the last address are not 0000(H) are treated as data. - If the contents of the last address are not 0000(H) are treated as data. - If the contents of the last address are not 0000(H) are treated as data. - If the contents of the last address are not 0000(H) are treated as data. - If the contents of the last address are not 0000(H) are - If the contents of the last address are not 0000(H) are - If the contents of the last address are not 0000(H) are - If the contents of the last address are - If the contents of the last address are - If the contents of
S		Use range B - Be sure to use even addresses for register S. Has many p
D	1	- Be sure to use even addresses for register D1.
Da	2	Use range B - The contents of register D ₂ are 000 to 377 ₍₈₎ . 000 ₍₈₎ specifies 256 words. - The contents of register D ₂ +1 are 000 to 377 ₍₈₎ . 000 ₍₈₎ specifies the top address.
Cond	ition	Rising edge of input signal (OFF to ON)
	S	Unchanged
	D1, D2+1	
Contents after	D1 to D1+2 (⊑ −1	address are not " $0000(H)$ " or $D_2 \le D_2+1$.
operation	Flag	$ \begin{array}{c c c c c c c c c c c c c c c c c c c $

13-45

Inserts data (2 words) (INSert) F-170d INS

Syml	bol		[Explanation] Instruction S T R 004000					
			F-170d					
		Insert the contents of register S to $S+3$ into register address $D_{1}+4$ ($D_{2}+1$) to	F-170d 009000 009100 009200 009100					
		D_1+4 (D_2+1)+3 in the shift register area						
Funct	tion	whose first register address is D1 and of which double words count is given	- Calculates when input condition 004000 changes					
		by the contents of register D ₂ .	from OFF to ON. - The contents (46320155(н)) of registers 009000 to					
		Area	009003 are inserted. The top address is register					
		D1 (Top address) D2 Double words count	009100.					
		D2+1 Insertion address	- Address to insert is one added 002(8) double words (contents of 009201) to register 009100.					
		D1+4 (D2+1) Inserted Insertion data	- The end address is 006(8) double words (contents					
		(Insertion data address) to S+3	of 009100) for 009100. The top address is included in the words count.					
			- When the contents of the end address are 0, the					
		Shifted 2 words	data at and beyond the address of insertion are shifted down 2 words, and the contents of address					
		D1+4 (D2)-1 (End address)	009000 to 009003 are stored in the insertion address.					
		- The shift register area that is set by	009000 55 (low)					
Opera	ation	D1 and D2 shall not be in the contact	$\begin{array}{c} 009001 \\ 009002 \\ 32 \end{array}$ Insertion data					
		point area of counter/timer (00001600 to 00001777(8) etc.), after	009002 32 009003 46 (high)					
		the last address of the register and	$\begin{array}{c} & & \\ \hline \hline & & \\ \hline \hline & & \\ \hline \hline & & \\ \hline & & \\ \hline \hline \\ \hline & & \\ \hline \hline \\ \hline & & \\ \hline \hline \hline \\ \hline \hline \hline \\ \hline \hline \hline \\ \hline \hline \hline \hline \\ \hline \hline \hline \hline \hline \hline \hline \\ \hline \hline$					
		file register. - If the end address is not 0000000(H),	$009201 002_{(8)} \rightarrow \text{Insert at 2nd double words from}$					
		or $D_2 \leq D_{2+1}$, an error flag (007355)	009100 Before operation \rightarrow After operation					
		turns ON and JW300 does not operate.						
		- If the contents of the end address are	Top 009101 34 34					
		not 0, the instruction is not executed even if the contents of intermediate	address 009102 00 00 009103 78 78					
		addresses are 0 at intermediate	009104 23 23					
		addresses are treated as data.	009105 34 34 009106 45 45					
s		Use range C	009107 67 67					
		- Be sure to use even addresses for register S.	009110 52 55 Insertion 009111 76 4 01					
D1		Use range C - Be sure to use even addresses for register D1.	address 009112 38 32					
		Use range B	6 009113 14 46 double 009114 37 52					
		- The contents of register D2 are 000 to	words 009115 52 / 76					
D2	2	377(8). (000(8) specifies 256 double words.) - The contents of register D ₂₊₁ are 000 to	009116 18 38 009117 45 14					
		377(8). (000(8) specifies the top address.)	009120 27 37					
Condi	tion	Rising edge of input signal (OFF to ON)	009121 68 / 52 009122 51 × 18					
	S to S+3	Unchanged	009123 39 45					
	S+3		$\begin{array}{c c c c c c c c c c c c c c c c c c c $					
	D2, D2+1	Unchanged	address 009126 00 51 009127 00 39					
Contout	D1 to	1 word shifts and value of S to S+3 contents inserted.	Shifted 2 words.					
Contents after	D1+4 (C -1	²⁾ - Unaffected when the contents of the end address are not "0" or $D_2 \leq D_{2+1}$.						
operation		Contents of Zero Carry Error Non-carry end address, etc. 007357 007356 007355 007354						
		Contents of						
	Flag							
		end address						
		not 0 or D2 ≦ D2+1						

F-17 DEL		Deletes of data (1 byte) (DELete)				
Symb	ool	- F-171 S1 S2 S3	[Explanation]	Instruction S T R 004000		
Funct	Function Delete 1 byte of data from address S1+ (S3) in the shift register area of which top register address is S1 and of which byte count is given by the contents of register S2. All the data beyond the deleted data are shifted up 1 byte position.		$\begin{bmatrix} -1/1 & 009010 & 019300 & 019301 \\ 019301 & 019301 \end{bmatrix} = \begin{bmatrix} 019300 & 019301 \\ 019301 & 019301 \end{bmatrix}$			
Operation		 S1 (Top address) S2 Byte count S1+ (S3) (Address where deletion occurs) S1+ (S2)-1 (End address) The shift register area that is set by S1+ (S2)-1 (End address) Shifted 1 byte. Shifted 1 byte. S1+ (S2)-1 (End address) Shifted 1 byte. Shifted 1 byte. S1+ (S2)-1 (End address) Shifted 1 byte. Shifted 1 byte. S1+ (S2)-1 (End address) Shifted 1 byte. Shifted 1 byte.<!--</td-->				
S1		Use range A Use range A				
S2		- The contents of register S ₂ are 000 to 377(8). (000(8) specifies 256 bytes.)				
S3		Use range A - The contents of register S ₃ are 000 to 377 ₍₈₎ . (000 ₍₈₎ specifies the top address.)				
Condit	tion	Rising edge of input signal (OFF to ON)				
	S2	Unchanged				
Contents after operation	S3 S1 to S1+(S -1	$\begin{tabular}{ c c c c c c }\hline & - Unaffected if $S_2 \leq S_3. \\ \hline Contents of $Zero$ Carry Error Non-carry $S_2, S_3 007357 007356 007355 007354 \\ \hline \end{tabular}$				
	Flag	$\begin{array}{c c c c c c c c c c c c c c c c c c c $				

F-171 DEL		Deletes data (1 word) (DELete)					
Symbo	ol		[Explanation] Instruction S T R 004000				
Functio	on	Delete 1 word of data from addresses S_{1+2} (S ₃) and S_{1+2} (S ₃) +1 in the shift register area of which top register address is S ₁ and of which word count is given by the contents of register S ₂ . All the data beyond the deleted data are shifted up 1 word position.	 Calculates when input condition 004000 changes from OFF to ON. The top address is register 009010. 				
Operation		S1 (Top address) Area S2 (Address of deletion) S1+2 (S3) (Address where deletion occurs) Data deleted S1+2 (S2)-1 (End address) Shifted 1 word. - The shift register area that is set by S2 and S3 shall not be in the contact point area of counter/timer (file addresses 00001600 to 00001777(8) etc.), after the last address of the register and file register. - If the contents of register is S2 ≦ S3, an error flag (007355) turns ON and JW300 does not operate. - The instruction is executed even if the contents of intermediate addresses are 0000(H) at intermediate addresses are treated as data.	- The address of deletion has an offset from address 009010 given by the contents (002(8) words) of register 019301. - The end address has an offset from address 009010 given by the contents (005(8) words) of register 019300. The top address is counted in the word count. - When the JW300 executes calculation, it erases the deleted address, and shifts data from the deleted address become 0000(H). 019300 $005(D) \rightarrow 5$ words from 009010 019301 $002(B) \rightarrow 5$ words from 009010 019301 $002(B) \rightarrow 5$ words from 009010 019301 $002(B) \rightarrow 5$ words from 009010 009012 $34 \qquad 34 $				
S1		Use range C - Be sure to use even addresses for register S1.					
S2		Use range A - The contents of register S ₂ are 000 to 377(8). (000(8) specifies 256 words.)					
S3		Use range A - The contents of register S ₃ are 000 to 377 ₍₈₎ . (000 ₍₈₎ specifies the top address.)					
Condition		Rising edge of input signal (OFF to ON)					
	S2	Unchanged					
Contents after operation	S3 S1 to S1+2 (S2) Flaç	Data deleted and subsequent data shifted 1 word. -1 - Unaffected if $S_2 \leq S_3$. Contents of Zero Carry Error Non-carry S2, S3 007357 007356 007355 007354					

F-171 DEL		Deletes data (2 words) (DELete)
Symbo	ol	F-171d DEL S1 S2 S3 [Explanation] Instruction S T R 004000
Functio	on	Delete 1 word of data from addresses S_{1+4} (S ₃) to S_{1+4} (S ₃) +3 in the shift register area of which top register address is S ₁ and of which double words count is given by the contents of register S ₂ . All the data beyond the deleted data are shifted up 1 word position. $F_{-171W}_{DEL}_{009010}_{019300}_{019301}$ $F_{-171W}_{DEL}_{009010}_{019300}_{019301}$ $F_{-171W}_{019300}_{019301}$ F_{-171W}_{0193
Operation		Si Address of deletion Si H4 (S3) (Address where beletion occurs) Shifted 1 word. - The shift register area that is set by S2 and S3 shall not be in the contact point area of counter/timer (file addresses 00001600 to 00001777(8) etc.), after the last address of the register and file register. - If the contents of register is $S_2 \leq S_3$, an error flag (007355) turns ON and JW300 does not operate. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 0000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The instruction is executed even if the contents of intermediate addresses are 00000000(H) at intermediate addresses are treated as data. - The end address - The instruction is executed even if - The i
S1		Use range C - Be sure to use even addresses for register S1. 009025 52 68 18 51 009026 18 15 30
S2		- The contents of register S ₂ are 000 to $377_{(8)}$. (000 ₍₈₎ specifies the top address.) $377_{(8)}$. (000 ₍₈₎ specifies the top address.) $377_{(8)}$. (000 ₍₈₎ specifies the top address.)
S3		- The contents of register S ₃ are 000 to 377(8). (000(8) specifies the top address.)
Conditi	on	Rising edge of input signal (OFF to ON)
	S2	Unchanged
	S3	Unchanged
Contents after operation	S1 to S1+4 (S2) -	Contents of Zero Carry Error Non-carry
	Flag	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$

F-172
SRCHSearches data (1 byte)
(SeaRCH)

Secretes in the range specified by the number of bytes (D2). The register S detail is data to search. Searches the data and stores the first matched address position (from D) to D2+2. The number of matches is stored in D2+1. Image: Search data data (D2+1) Operation			•								
Function Take register D1 as top address, searches in the range specified by the number of bytes (D2). The register S detail is data to search. Searches the data and stores the first matched address position (from D) to D2+2. The number of matches is stored in D2+1. Image: Contents of register 009000 is search diated to search. Stored in D2+1. This instruction of 004000 changes from OFF to C The top address is register 009000 is search diated to search. Stored in D2+1. Operation Image: Contents of register 02000 is search diated to search diated to search. Stored in D2+1. The shift register area that is set by D is and D2 shall not be in the contact point area of counter/timer (file addresses 00001600 to 00001777(a) etc.), after the last address of the register and file register. The solution of 004000 is search data S Use range A Use range A D1 Use range A Searche stored to address is 0 or 256. Use the carry flag (007366) to identify the presence of matched data. D2 Use range A D1 Use range A D2 Use range A D3 Use range A D2 Use range A D3 Unchanged D2+1 Unchange	Symb	ol	S	Dı	D 2		[Explanation]				
Operation Image: Contents after gister D2+1 are 000 to 377(e). When it is 000(e), the number of matches is 0 or 256. Use the carry flag (007366) to identify the presence of matched data. Image: Contents after gister D2+1 are 000 to 377(e). The number of bytes, taking D1 as standard, is stored. (When this is 000(e), the match data is the top address.) Image: Contents after gister D2+1 are 000 to 377(e). The number of bytes, taking D1 as standard, is stored. (When this is 000(e), the match data is the top address.) Image: Contents after gister D2+1 are 000 to 377(e). The number of bytes, taking D1 as standard, is stored. (When this is 000(e), the match data is the top address.) Image: Contents after Contents Contents Contents after Contents Contents after Conten	Functio	on	searches in number of b detail is data Searches th matched ad D ₂₊ 2. The n	the range tytes (D ₂). a to search e data and dress posi number of r	specified t The registen d stores the tion (from	oy the er S e first D1) to	 This instruction is executed wh condition of 004000 changes f The contents of register 00900 The top address is register 100 	009000 ⊐00500 019000 nen the input from OFF to ON. 00 is search data. 0500.			
S Use range A D1 Use range A Use range A 10 10 Use range A 10 Use range E The contents of register D2 are 000 to 377(8). (000(8) specifies the 256 bytes.) 10 Detail of the register D2+1 are 000 to 377(8). When it is 000(8), the number of matched data. When it is 000(8), the number of matched data. When execution is complete, the carry flag (007356) to identify the presence of matched data. When execution is complete, the carry flag (10000000, the matched data. Detail of the register D2+2 is 000 to 377(8). The number of bytes, taking D1 as standard, is stored. (When this is 000(8), the match data is the top address.) When execution is complete, the carry flag (1000000000000000000000000000000000000	Operati	ion	D1+(D2)-1 (End addres - The shift re D1 and D2 point area addresses etc.), after	egister are shall not b of counter. 00001600 the last ac	s Searc D2 Byte Count D2+1 Addres D2+2 matche a that is se e in the co /timer (file 0 to 000017 ddress of th	count of ed data ss of ed data et by ntact 777(8)	register 019000. The top addrest the byte count. - The number of matched data is register 019001. - The address (offset from D1) of data found is stored in register 009000 55(H) Search 019000 012(B) Number 019001 004(B) Number 019002 002(B) Matcher (Top address ⊐00500 11	ess is included in items is stored to of the first matched r 019002. h data er of words er of matches			
D1 Ose range A Use range E - The contents of register D2 are 000 to 377(8). (000(8) specifies the 256 bytes.) - Detail of the register D2+1 are 000 to 377(8). When it is 000(8), the number of matches is 0 or 256. Use the carry flag (007356) to identify the presence of matched data. - Detail of the register D2+2 is 000 to 377(8). The number of bytes, taking D1 as standard, is stored. (When this is 000(8), the match data is the top address.) Condition Rising edge of input signal (OFF to ON) S Unchanged D2 Unchanged D2 Unchanged D2 Unchanged D2 When the number of matches is 0 or 256, D2+1 will be 000(8).	S		Use range A				⊐00503 33				
D2 Ose range E 100507 55 - The contents of register D2 are 000 to 377(8). (000(8) specifies the 256 bytes.) - - - Detail of the register D2+1 are 000 to 377(8). When it is 000(8), the number of matches is 0 or 256. Use the carry flag (007356) to identify the presence of matched data. - When execution is complete, the carry flag (is set ON. - Detail of the register D2+2 is 000 to 377(8). The number of bytes, taking D1 as standard, is stored. (When this is 000(8), the match data is the top address.) - When execution is complete, the carry flag (is set ON. Condition Rising edge of input signal (OFF to ON) - - D2 Unchanged - D2 Unchanged D2 Unchanged D2 Unchanged D2 Unchanged D2 When the number of matches is 0 or 256, D2+1 will be 000(8).	D1		Use range A				bytes ⊐00505 55] ←			
S Unchanged D1 to D1+ (D2) -1 Unchanged D2 Unchanged D2+1, operation D2+1, D2+2 Result - When the number of matches is 0 or 256, D2+1 will be 000(8).	D2		Use range E - The contents of register D ₂ are 000 to 377 ₍₈₎ . (000 ₍₈₎ specifies the 256 bytes.) - Detail of the register D ₂ +1 are 000 to 377 ₍₈₎ . When it is 000 ₍₈₎ , the number of matches is 0 or 256. Use the carry flag (007356) to identify the presence of matched data. - Detail of the register D ₂ +2 is 000 to 377 ₍₈₎ . The number of bytes, taking D ₁ as standard, is stored. (When this is 000 ₍₈₎ , the match data is the top				- When execution is complete, th	⊷ ← ne carry flag (007356)			
D1 to D1+ (D2) -1 Unchanged D2 Unchanged D2+1, operation D2+1, D2+2 Result - When the number of matches is 0 or 256, D2+1 will be 000(8).	Condition		Rising edge of input signal (OFF to ON)								
Flag Search data Zero 007357 Carry 007356 Error 007355 Non-carry 007354 Found 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0 1 1 0 0 1<	Contents . after	D1 to D (D2) – D2 D2+1 D2+2	D1+ -1 Unchanged 12 Unchanged 12 Result - When the number of matches is 0 or 256, D2+1 will be 000(8). 14 Search data 2 007357 007356 14 0 15 Found 16 1 17 0								

F-172 SRCI	2w H	Searches data (1 word) (SeaRCH)				
Symb	ol	F-172w S D1 D2	[Explanation]	Instruction S T R 004000		
Functi	on	Taking register D_1 as the top address, searches in the range specified by the number of words (D_2). Register S and S+1 details are data to search. Store the address (offset from D_1) of the first matched data in D_2+2 and the number of matched data items in D_2+1 .	 - This instruction is executed wh condition of 004000 changes fi - The contents (3355(H)) of regis 009001 are the search data. 	rom OFF to ON.		
Operat	ion	 Search direction (Top address) D1 D1+1 D1+1 D2 Word count D2+1 D2+2 Address of matched data Address of matched data - The shift register area that is set by D1 and D2 shall not be in the contact point area of counter/timer (file addresses 00001600 to 00001777(8) etc.), after the last address of the register and file register.	$\begin{bmatrix} -00503 & 33 \\ 6 & -00504 & 44 \end{bmatrix} = \begin{bmatrix} 009000 \end{bmatrix}$	from address (006(8) words) of ess is included in tems is stored to f the first matched 019002.		
S		Use range B - Be sure to use even addresses for register S.	□00506 66 □00507 55 □00510 77 □00511 55 1 End (100512 55			
D1		Use range B - Be sure to use even addresses for register D1				
D2		 Use range E The contents of register D₂ are 000 to 377(8). (000(8) specifies the 256 words.) Detail of the register D₂₊₁ are 000 to 377(8). When it is 000(8), the number of matches is 0 or 256. Use the carry flag (007356) to identify the presence of matched data. Detail of the register D₂₊₂ is 000 to 377(8). The number of words, taking D₁ as standard, is stored. (When this is 000(8), the match data is the top address.) 	f			
Condition Rising edge of input signal (OFF to ON)						
	S, S+ D1 to					
	D1+2 ([1	D2) Unchanged				
Contents after operation	D2 D2+1					
	D2+2	Search data Zero 007357 Carry 007356 Error 007355 Non-carry 007354				
	Flag	Found 0 1 0 0 Not found 1 0 0 0 0				

F-172d
SRCHSearches data (2 words)(SeaRCH)

Symb	ool	F-172d SRCH S	D1	D2			[Explar	nation]			STR	ruction 004000
Functi	on	Taking regis searches in number of d S to S+3 de Store the ad the first mate number of m	the ran ouble v tails are dress (ched da	ge spe vords (e data f offset f ata in E	cified b D2). Re to sear from D D2+2 ar	by the egister ch. 1) of nd the	004000 F-172d 009000 □00500 019000 □00500 019000 - This instruction is executed when the input condition of 004000 changes from OFF to ON. - The contents (99223355(H)) of registers 009000 to 009003 are the search data.					
Operation		Search direction D1 (Top address) D1+4 D1+10(8) D1+2 ((D2)-1) (End address) D1+2 ((D2)-1) (End address) Search data S+3 D2 Double word count D2+1 No. of matched data Address of matched data						 The top address is register J00500. The end address has an offset from address J00500 given by the contents (006₍₈₎ double words) of register 019000. The top address is included in the double words count. The number of matched data items is stored to register 019001. The address (offset from D1) of the first matched data found is stored in register 019002. 				
		- The shift re D1 and D2 s point area addresses etc.), after register an	shall no of cour 000010 the las	ot be in iter/tim 600 to t addre	the co er (file 000017	ntact 777 ₍₈₎	addres	p = 00500 = 00501 = 00502 = 00502 = 00504	Searc 11 22 55 33 55	h directio	on It matched	address
S		Use range C - Be sure to use even addresses for register S.					⊐00505 ⊐00506	33 22				
D1		Use range C - Be sure to us		ddresse	es for reg	gister D1		⊐00507 ⊐00510 ⊐00511	<u>99</u> 44 55		55 (lower)	
D2		Use range E - The conter 377(8). (000 double wor - Detail of th 377(8). Whe of matches flag (00735 of matched - Detail of th 377(8). The taking D1 a (When this is the top a	tts of re (8) spec ds.) e regis en it is (is 0 or 6) to ic data. e regis numbe s stanc is 000(egister ifies th ter D2+ 256. U lentify the ter D2+ r of dou lard, is 8), the	D ₂ are le 256 the nur Jse the the pre 2 is 00 uble wo stored	000 to 000 to nber carry sence 0 to ords,	6 double words En addres	□00512 □00513 □00514 □00515 □00516 □00517 □00520 □00521 □00522 □00523 \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$	66 55 22 33 77 88 00 66 33 77 55 33 22 99	009001 € 019002 2 019003 € 019000 0 019001 0 019002 0 <	33 22 39 (upper) 99 (upper) 006(8) 002(8) 001(8)	Search data Number of double words Matched address Double word No. ess
Condition Rising edge of input signal (OFF to ON)			is set c	JN.								
Contents after operation	S to S+3 D1 to D1+4 (E -1 D2 D2+1 D2+2	Unchanged Unchanged Unchanged										
	Flag	Search data Found Not found	Zero 007357 0 1	Carry 007356 1 0	Error 007355 0	Non-carry 007354 0						

F-173 CHNG	Changes data (1 byte) (CHaNG)	
Symbol	(1) F-173 S D1 D2 (1) Mode input (2) CHNG S D1 D2 (2) Execution input	STR 004000
Function	Taking register D ₁ as the top address, searches in the range specified by the number of byte (D ₂). Register S details are data to search. Store the address (offset from D ₁) of the first matched data in D ₂ +2 and the number of matched data items in D ₂ +1. Data to be overwritten is the detail of register S+1. The JW300 overwrites with mode specification (1) input condition. - When mode input (1) is OFF: Only the first matched data is replaced. - When mode input (1) is ON: All matched data are replaced.	 O04000 O04001 F-173 CHNG F-173 CHNG CHNG O09000 □00500 019000 F-173 O09000 □00500 019000 F-173 O09000 □00500 019000 This instruction is executed when the input condition of 004001 changes from OFF to ON. The contents (55(H)) of register 009000 is the search data. The top address is register □00500. The end address has an offset from address □00500 given by the contents (010(D) bytes) of register 019000. The top address is included in the byte count. The number of matched data items is stored to register 019001. The address (offset from D1) of the first matched
Operation	Search direction (Top address) D1 D1+1 D1+2 D2 Number of bytes D2+1 Number of matches D2+2 Matched address - The shift register area that is set by D1 and D2 shall not be in the contact point area of counter/timer (file addresses 00001600 to 00001777(8) etc.), after the last address of the register and file register.	data found is stored in register 019002. - The contents (64(H)) of register 009001 is the replacement data. - Example: Below is an operation when 004000 is ON. 009000 55 Search data 019000 010(D) Number of bytes O09001 64 Replacement 019001 004(B) Number of matches 019002 002(B) Byte No. Before operation \rightarrow After operation first = 100500 11 = 11 = 100500 = 11 = 100500 = 100501 = 100500 = 100501 = 100500 = 100501 = 100501 = 100502 = 100503 = 10 = 100504 = 100504
S	Use range B	bytes $\exists 00505 55 \longrightarrow 64 \exists 00505 \\ \exists 00506 66 & 66 \\ \exists 00506 \\ \hline 00507 55 & \hline 00506 \\ \hline 00507 55 & \hline 00507 \\ \hline 00507 \hline 00507 \\ \hline 00507 \hline 00507 \\ \hline 00507 \hline 00507 $
D1	Use range A	□00507 <u>55</u> <u>64</u> □00507 End □00510 <u>77</u> □00510 address □00511 <u>55</u> <u>64</u> □00511
D2	 Use range E The contents of register D₂ are 000 to 377₍₈₎. (000₍₈₎ specifies the 256 bytes.) Detail of the register D₂₊₁ are 000 to 377₍₈₎. When it is 000₍₈₎, the number of matches is 0 or 256. Use the carry flag (007356) to identify the presence of matched data. Detail of the register D₂₊₂ is 000 to 377₍₈₎. The number of byte, taking D₁ as standard, is stored. (When this is 000₍₈₎, the match data is the top address.) 	- When execution is complete, the carry flag (007356) is set ON. Contents after operation
Condition	Rising edge of input signal ((2) execution input) (OFF to ON)	Flag Found 0 1 0 0 Not found 1 0

F-173wChanges data (1 word)CHNG(CHaNG)

		[Evployedian]
Symbol	(1) F-173w (2) CHNG S D1 D2 (1) Mode input (2) Execution input	[Explanation] Instruction S T R 004000
Function	Search for data matching the contents of registers S and S+1 in this register area whose top address is D1 and whose word count is given by the contents of register D2 and store the address (offset from D1) of the first matched data in D2+2 and the number of matched data items in D2+1. Replace matched data with the contents of registers S+2 and S+3 depending on the state of mode input (1): - When mode input (1) is OFF: Only the first matched data is replaced. - When mode input (1) is ON: All matched data are replaced.	 STR 004001 F-173w O09000 CHNG 009000 =00500 019000 F -173w O09000 009000 009000 009000 009000 009000 019000 This instruction is executed when the input condition of 004000 changes from OFF to ON. The contents (3355(H)) of register 009000, 009001 are the search data. The top address is register =00500. The end address has an offset from address =00500 given by the contents (005(8) words) of register 019000. The top address is included in the word count. The number of matched data items is stored to register 019001. The address (offset from D1) of the first matched
Operation	 Search direction (Top address) D1 D1+2 D1+2 D1+4 D1+2 D1+4 D2 Number of words D2+1 Number of matches D2+2 Matched address - The shift register area that is set by D1 and D2 shall not be in the contact point area of counter/timer (file addresses 00001600 to 00001777(8) etc.), after the last address of the register and file register.	data found is stored in register 019002. - The contents (1964(H)) of registers 009002 and 009003 are the replacement data. - Example: Below is an operation when 00x000 is ON. 009000 55 (lower) 009001 33 (upper) 009002 64 (lower) 009002 64 (lower) 009002 64 (lower) 19 (upper) 019002 001 (a) 0001 (b) 001 (c) 001 (c) 00500 19 (upper) 00500 11 11 11 100500 100501 22 22 22 100501 100501 22 100501 100502 55 Replacement 64 100502 19 (upper) 19 (upper) 19 (upper) 19 (upper) 19 (upper) 10 (upper) 11 (upper) 10 (upper) 10 (upper) 10 (upper) 10 (upper) 11 (upper) 10 (upper) 11 (upper) 10 (upper)
S	Use range C - Be sure to use even addresses for register S.	□00506 66 □00506 □00506 □00506 □00506 □00506 □00507 □00507 □00507 □00507 □00510
D1	Use range B - Be sure to use even addresses for register D1.	address ⊐00511 <u>55</u> ⊐00511
D2	 Use range E The contents of register D₂ are 000 to 377₍₈₎. (000₍₈₎ specifies the 256 words.) Detail of the register D₂₊₁ are 000 to 377₍₈₎. When it is 000₍₈₎, the number of matches is 0 or 256. Use the carry flag (007356) to identify the presence of matched data. Detail of the register D₂₊₂ is 000 to 377₍₈₎. The number of words, taking D₁ as standard, is stored. (When this is 000₍₈₎, the match data is the top address.) 	S to S+3 Unchanged D2 Unchanged D2 Unchanged D2+1, D2+2 Result - If the result is 256 or no matched data.D2+1 becomes 000. Person Carry 007357 Error 007356 Non-carry 007354
Condition	Rising edge of input signal ((2) execution input) (OFF to ON)	FlagFound0100Not found1000

F-173d CHNG	Changes data (2 words) (CHaNG)	
Symbol	(1) F-173d S D1 D2 (1) Mode input (2) CHNG S D1 (2) Execution input	[Explanation] Instruction S T R 004000 S T R 004001
Function	Search for data matching the contents of registers S to S+3 in this register area whose top address is D1 and whose word count is given by the contents of register D2 and store the address (offset from D1) of the first matched data in D2+2 and the number of matched data items in D2+1. Replace matched data with the contents of registers S+4 to S+7 depending on the state of mode input (1): - When mode input (1) is OFF: Only the first matched data is replaced. - When mode input (1) is ON: All matched data are replaced.	 This instruction is executed when the input condition of 004001 changes from OFF to ON. The contents (99223355(H)) of register 009000 to 009003 are the search data. The top address is ¬00500. The end address has an offset from address ¬00500 given by the contents (004(8) double words) of register 019000. The top address is included in the double words count. The number of matched data items is stored to register 019001. The address (offset from D1) of the first matched
Operation	 Search direction (Top address) D1 D1+4 D1+10(8) D1+4 D1+10(8) D2 Number of double words D2+2 Matched address The shift register area that is set by D1 and D2 shall not be in the contact point area of counter/timer (file addresses 00001600 to 00001777(8) etc.), after the last address of the register and file register. 	double words found is stored in register 019002. - The contents (19643385(H)) of registers 009004 to 009007 are the replacement data. - Example: Below is an operation when 004000 is ON. 009000 55 (lower) 009001 33 019002 22 019002 22 019002 22 019003 99 (upper) 009004 85 (lower) 009005 33 019002 001(8) 0001(8) Number of double words 0001(8) Number of matches 001(8) Double word No. Peplacement data Before operation \longrightarrow After operation $\begin{bmatrix} Top \\ address \\ 100500 \\ 100501 \\ 22 \\ 55 \end{bmatrix}$ $\begin{bmatrix} 11 \\ 22 \\ 55 \end{bmatrix}$
S	Use range C - Be sure to use even addresses for register S.	□00503 <u>33</u> □00504 <u>55</u> <u>85</u>
D1	Use range B - Be sure to use even addresses for register D1.	$\begin{array}{c c c c c c c c c c c c c c c c c c c $
D2	Use range E - The contents of register D ₂ are 000 to 377 ₍₈₎ . (000 ₍₈₎ specifies the 256 double words.) - Detail of the register D ₂ +1 are 000 to 377 ₍₈₎ . When it is 000 ₍₈₎ , the number of matches is 0 or 256. Use the carry	$\begin{array}{c c c c c c c c c c c c c c c c c c c $
	flag (007356) to identify the presence of matched data.	Sto S+7 Unchanged
	 Detail of the register D₂+2 is 000 to 377(8). The number of double words, taking D₁ as standard, is stored. (When this is 000(8), the match data is the top address.) 	D2 Unchanged Contents after operation D2+1, D2+2 - If the result is 256 or no matched data.D2+1 becomes 000(8). Result - Zero 007357 Carry 007356 Error 007355
Condition	Rising edge of input signal ((2) execution input) (OFF to ON)	FlagFound0100Not found1000

Ch	Chapter 14 Application instructions (F-174 to F-403)									
F-17 VRE		Reverse order of register data (1 byte) (Vertical REVerse)								
Symt	ool	F-174 D n VREV D n	[Explanation] Instruction S T R 004000 F -174							
Function		Within the n byte register that starts with register D, swap the contents of each register between "the lowest address to the upper most address" and "the upper most address to the lowest address."	When the input condition of 004000 changes from C ON, converts the contents of the 10 (0012(8)) in regi 009000 to 009011 in reverse order. Swap between "009000 -> 009011" and "0090011 -> 009000."				8)) in registers	to		
Operation		D D+1 , D+n-2 D+n-1 Data swap					000." After operation			
D		Use range A *	009001 009002	32 54	2)	9) 8)	01 15			
n		Use range 0000 to 1777 ₍₈₎ (1024 bytes for 0000 ₍₈₎)	009003 009004	76 98	4) 5)	7) 6)	26 37			
Condition		Rising edge of input signal (OFF to ON)	009005 009006	37 26	6) 7)	5) 4)	98 76			
Contents	D to D+n-1	Result	009007 009010	15 01	8)	3) 2)	54 32			
after operation	Flag	Unchanged	009010	23	9)]10) _	2) 	10			

* Be careful that the register areas set by D and n do not enter the contact point areas of the timer/counter (file address 00001600 to 00001777(8) etc.) and the last address or more of the registers and file registers. Resembled instructions: F-02, F-02w, F-02d

F-175
NSWPSwap upper 4 bits with lower 4 bits of register(N byte SWaP)

Sym	nbol	-F-175 D n NSWP D n	[Explanation] Instruction S T R 004000)
Function		Swap the upper 4 bits with lower 4 bits of "n" bytes of register area that begins with register D.	F-175 F-175 NSWP 009000 011 009000 011	
Opera	ation	Swap nibbles of registers D to D+n-1.	When the input condition of 004000 changes fro OFF to ON, this instruction swaps the upper nible	bles
D)	Use range A *	with lower nibbles of the 9 (0011(8))-byte content registers 009000 to 009010 (1-byte register each	
n		Use range 0000 to 1777 ₍₈₎ (1024 bytes for 0000 ₍₈₎)	Before operation After operation 009000 9 1	
Cond	lition	Rising edge of input signal (OFF to ON)	009001 8 2 2 8 009002 7 3 3 7	
Contents	D to D+n-1	Result	009003 <u>6 4</u> 009004 <u>0 5</u> <u>5 0</u>	
after operation Flag		Unchanged	009005 4 6 6 4	
* Be careful that the shift area set by D and n do not enter the contact point areas of the timer/counter		-	009006 3 7 7 3 009007 2 8 8 2	
(file address 00001600 to 00001777 $_{(8)}$ etc.) and the last address or more of the registers and file				I
registers.			Nibbles are swapped.	

Resembled instructions: F-55

F-176
DFRDRead 256 bytes from register of specified address(Direct File Read)

Sym	ibol	F-176 S FILE F D	[Explanation] Instruction S T R 004000 F -176					
Function		In file number F, transfer data (256 bytes), that was specified by register S and S+1, to the address starting from register D.	$\begin{array}{c c} \hline 004000 \\ \hline F-176 \\ \hline DFRD \end{array} \xrightarrow{100402} FILE 1 \\ \hline 019000 \end{array} \xrightarrow{100402} FILE 1 \\ \hline 019000 \\ \hline 019000 \\ \hline 019000 \\ \hline 004000 \\ \hline 019000 \\ $					
Opera	ation	Transfer 256 bytes from file F, S, S+1 (block No.).	data from file No. 1's file area that begins with block number 000020(8) (contents of register ⊐00402, ⊐00403) to the register area that begins with register					
S		Use range B - Be sure to use even addresses for register S.	019000. File address of file No. 1					
F	:	0 (Data memory except file register) 1 (File register)	100402 100403000020(8)FileBeforeAfter operationaddressoperationoperation					
C)	Use range J - Be sure to use even addresses for register D.	00010000 10 00010001 32 256 32 019001					
Cond	ition	Rising edge of input signal (OFF to ON)	⇒ → → → → → → → → → → → → → → → → → → →					
	S to S+255	Unchanged	00010375 97 97 019375 00010376 98 98 019376					
After operation	D to D+255	Result	00010376 98 98 019376 00010377 99 99 019377					
	Flag	Unchanged	- The top address of block No. 000020(8) of file No. 1 is					
Resemb	led ins	structions: F-102, F-102w, F102d	00010000(8).					

- S and S+1 are block numbers (000000 to 100000₍₈₎) separated file No. 0 (file address) and file number 1 (byte address) in 256 bytes each. => See "Data memory block number."

This instruction performs block transfer of 256 bytes each.

- Be careful for the top address that is specified by D. 256 bytes from there will be a transfer area.

F-177 DFWR

Write data into registers of specified address (256 bytes) (Direct File WRite)

Symbol		-F-177 DFWR S D FILE F	[Explanation] Instruction					
		Transfers 256 bytes of data that	S T R 004000 F -177					
Function		starts from register S to the area that was specified by register D and D+1 in file No. F.	004000 F-177 DFWR 019100 00420 FILE 1 019100 00920 FILE 1 FILE 1					
Operation		Transfers 256 bytes from S to D, D+1 (block No.) of file F.	When the input condition of 004000 changes from OFF to ON, this instruction transfers 256 bytes of					
S		Use range B - Be sure to use even addresses for register S.	data to the area that was specified by block No. 000020 ⁽⁸⁾ (register ⊐00420, ⊐00421) of file No. 1 (file register).					
D		Use range B						
F		0 (Data memory except file register)	Block No.					
•		1 (File register)	□00420 □00421 000020(8)					
Cond	ition	Rising edge of input signal (OFF to ON)						
	S to S+255	Unchanged	File Before After File address operation register					
Contents after operation	D to D+255	Result	019100 1 0 019101 3 2 3 2 0001000					
	Flag	Unchanged	256 bytes					
Resemb	led ins	structions: F-103, F-103w, F103d	019475 9 7 9 7 00010375					
		of D and D+1 use block No. (000000	019476 9 8 9 8 00010376					
addres	s) and	that is separated file No. 0 (file file No. 1 (byte address) in units of ch. => See "Data memory block	019477 9 9 9 00010377					
256 by numbe		un. => See Data memory block	- The top address of block No. 000020(8) of file No. 1 is					

bytes 00010000(8).

This instruction performs block transfer of 256 bytes _____0 each.

- For setting the block number to D, do not use block No. (3, 78, 301 to 307(8)) that contains the contact area of the TMR/CNT of file No. 0.

Compare between register and register (1 byte) (with relay output)

F-180 CP>		(ComPare >)		F-18 CP>		ComPare >=)
F-181 CP< ((ComPare <)		F-18 CP<		ComPare <=)
	F-182 CP= (re =)	F-18 CP<		ComPare <>)
Symbol			$ \begin{array}{ c c c c c c c c } \hline F-180 & S_1 & S_2 & BIT \\ \hline CP & S_1 & S_2 & BIT \\ \hline CP & S_1 & S_2 & BIT \\ \hline \hline F-182 & S_1 & S_2 & BIT \\ \hline CP & S_1 & S_2 & BIT \\ \hline \end{array} $	IT		$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$
Fun	ction					ister S1 and content of S2, the bit is
		F-180	$S_1 > S_2 \rightarrow BIT ON$		F-183	$S_1 \ge S_2 \rightarrow BIT ON$
Ope	ration	F-181	81 $S_1 \leq S_2 \rightarrow BIT ON$			$S_1 {\leq} S_2 {\rightarrow} BIT \; ON$
		F-182	$S_1 = S_2 \rightarrow BIT ON$		F-185	$S_1 \neq S_2 \rightarrow BIT ON$
5	61	Use rar	ige A	S	S ₂ Use range A	
В	IT	Use rar	e range K		Condition When the input signal is ON (not limited to OFF to ON changed)	
	S1, S2	Unchai	nged			
	BII	F-180	ON (S1>S2) OFF (S1≦S2)		F-183	ON (S1≦S2) OFF (S1 <s2)< td=""></s2)<>
Contents after operation		F-181	ON (S1 <s2) OFF (S1≧S2)</s2) 		F-184	ON (S1≦S2) OFF (S1>S2)
		F-182	ON (S1=S2) OFF (S1≠S2)		F-185	ON (S1≠S2) OFF (S1=S2)
	Flag	Zero 007357 0	Carry 007356 Error 007355 Non-carry 007354 0 007355 007354			
[Explai		009000 009	Instruction S T R 004001 F-180 009000 000200 000200	COI 009 ON Wh	ntents of 9010 are I if 00900 nen the ir	nput condition 004001 is ON, the register 009000 and register compared, and the relay 000200 is 00 > 009010. nput condition 004001 is OFF, and 009010, relay 000200 goes OFF.

Compare between register and register (1 word) (with relay output)

F-180w CP>	ComPare >)	F-183w CP>=	ComPare >=)	
F-181w CP<	ComPare <)	F-184w CP<=		
F-182w CP = (ComPare =)	F-185w CP<>	(ComPare <>)	
Symbol				
Function			nd the contents (1-word data) of registers d ON when the result is properly obtained.	
	F-180 w S ₁ , S ₁ +1>S ₂ , S ₂ +1 \rightarrow B	T ON F-1	83 w $S_1, S_1+1 \ge S_2, S_2+1 \rightarrow BIT ON$	
Operation	F-181 w S ₁ , S ₁ +1 $<$ S ₂ , S ₂ +1 \rightarrow B	T ON F-1	84 w $S_1, S_1+1 \le S_2, S_2+1 \rightarrow BIT ON$	
	F-182 w S ₁ , S ₁ +1=S ₂ , S ₂ +1 \rightarrow B	T ON F-1	85 w $S_1, S_1+1 \neq S_2, S_2+1 \rightarrow BIT ON$	
S1	Use range B	S2	Use range B	
BIT	Use range K	Condition	When the input signal is ON (not limited to OFF to ON change)	
S1, S1 + 1 S2, S2 + 1	Unchanged	·		
	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	· · · · · · · · · · · · · · · · · · ·	83w ON (S ₁ , S ₁ +1 \geq S ₂ , S ₂ +1) OFF (S ₁ , S ₁ +1 $<$ S ₂ , S ₂ +1)	
Contents after operation	$\begin{array}{c c} F-181 w & ON (S_1, S_1+1 < S_2, S_2+\\ & OFF (S_1, S_1+1 \ge S_2, S_2-1) \end{array}$	· =	84w ON (S ₁ , S ₁ +1 \leq S ₂ , S ₂ +1) OFF (S ₁ , S ₁ +1>S ₂ , S ₂ +1)	
	$ F-182w ON (S_{1},S_{1}+1=S_{2}, S_{2}+1) OFF (S_{1},S_{1}+1\neq S_{2}, S_{2}+1) $	· =	85w ON (S ₁ , S ₁ +1 \neq S ₂ , S ₂ +1) OFF (S ₁ , S ₁ +1=S ₂ , S ₂ +1)	
Flag	Zero Carry Error Non-carry 007357 007356 007355 007354 0 0 0 0	·		
Instruction c STR 004201			e input condition 004201 is ON, the of registers 009100 and 009101, and ents of registers 009110 and 009111 pared, and the relay 000100 is ON if	
004201 F-180w CP>	009100	(009100) When th (009100)	, 009101) > (009110, 009111). e input condition 04201 is OFF, and , 090101) \leq (009110, 009111), relay goes OFF.	

Compare between register and register (2 words) (with relay output)

	-180d CP> (ComPare >)			F-18 CP>		ComPare >=)
F-181d CP< (ComPare \leq)		F-18 CP<	4d	ComPare <=)
	F-182d CP = (re =)	F-18 CP<		ComPare <>)
Syr	mbol					
			- F-182d CP= S1 S2 BIT			
Fun	nction		2 + 3 are compared in magnitud			ne contents (2-words data) of registers ed ON when the result is properly
		F-180d	S_1 to $S_1+3 > S_2$ to S_2+3-	BIT ON	F-183d	S_1 to $S_1+3 \ge S_2$ to $S_2+3 \rightarrow BIT$ ON
Ope	eration	F-181d	S_1 to $S_1 + 3 < S_2$ to $S_2 + 3 - 3$	► BIT ON	F-184d	S_1 to $S_1+3 \leq S_2$ to $S_2+3 \rightarrow BIT$ ON
		F-182d	S_1 to $S_1+3 = S_2$ to $S_2+3 -$	► BIT ON	F-185d	S_1 to $S_1+3 \neq S_2$ to $S_2+3 \rightarrow BIT$ ON
Ś	S1	Use range C		5	S2	Use range C
В	ыт	Use ran	range K		Condition When the input signal is ON (not limited to OFF to ON char	
	S_1 to $S_1 + 3$ S_2 to $S_2 + 3$	Unchar	nged			
		F-180d	ON (S ₁ to S ₁ +3 > S ₂ to S OFF (S ₁ to S ₁ +3 \leq S ₂ to S	,	F-183d	ON (S ₁ to S ₁ +3 \ge S ₂ to S ₂ +3) OFF (S ₁ to S ₁ +3 < S ₂ to S ₂ +3)
Contents after operation	BIT	F-181d	ON (S ₁ to S ₁ +3 < S ₂ to S OFF (S ₁ to S ₁ +3 \ge S ₂ to S		F-184d	ON (S ₁ to S ₁ +3 \leq S ₂ to S ₂ +3) OFF (S ₁ to S ₁ +3 > S ₂ to S ₂ +3)
		F-182d	ON (S ₁ to S ₁ +3 = S ₂ to S OFF (S ₁ to S ₁ +3 \neq S ₂ to S		F-185d	ON (S ₁ to S ₁ +3 \neq S ₂ to S ₂ +3) OFF (S ₁ to S ₁ +3 = S ₂ to S ₂ +3)
	Flag	Zero 007357 0	Carry Error Non-carry 007356 007355 007354 0 0 0			
[Explanation] [b] [Explanation] [b] [b] [b] [b] [b] [b] [b] [b] [b] [b]					ents of r the cont 111 are o f (00910 n the inp	put condition 004201 is ON, the registers 0009100 and 009101, tents of registers 009110 and compared, and the relay 000100 is 00, 009103) > (009110, 009113). put condition 04201 is OFF, and 0103) \leq (009110, 009113), relay FF.

Fc180 CP>		(ComPare >)		Fc18 CP>		ComPare >=)
Fc181 CP<		ComPare <)		Fc18 CP<		ComPare <=)
Fc182 CP=		ComPa	re =)	Fc18 CP<		ComPare <>)
			-Fc180 CP> S n BIT			-Fc183 CP>= S n BIT
Syn	nbol		-Fc181 CP< S n BIT			-Fc184 CP<= S n BIT
			- Fc182 CP= S n BIT			-Fc185 CP<> S n BIT
Fun	ction	Comparing the magnitude between the content of register S and content of n, and a bit is turned ON when the result is properly obtained.				
		Fc180	$S > n \rightarrow BIT ON$		Fc183	$S \ge n \rightarrow BIT ON$
Oper	ration	Fc181	S <n bit="" on<="" td="" →=""><td></td><td>Fc184</td><td>$S \leq n \rightarrow BIT ON$</td></n>		Fc184	$S \leq n \rightarrow BIT ON$
		Fc182	S = n → BIT ON		Fc185	S ≠ n → BIT ON
	5	Use ran	ige A	n		Use range 000 to 377(8)
В	IT	Use rar	ige K	Cond	dition	When the input signal is ON (not limited to OFF to ON change)
	S	Unchar	nged			
		Fc180	ON (S>n) OFF (S≦n)		Fc183	ON (S≧n) OFF (S <n)< td=""></n)<>
After operation	BIT	Fc181	ON (S <n) OFF (S≧n)</n) 		Fc184	ON (S≦n) OFF (S>n)
		Fc182	ON (S≕n) OFF (S≠n)		Fc185	ON (S≠n) OFF (S=n)
	Flag	Zero 007357 0	Carry Error Non-carry 007356 070355 007354 0 0 0			
[Explar	-		Instruction S T R 005001 Fc180 019000	cor 01	ntents of 2 are cor	nput condition 005001 is ON, the register 019000 and octal constant npared, and the relay 000300 is ON $> 012_{(8)}$.
	Fc180 	019000 0	000300 012 000300	Wr	nen the ir	pput condition 005001 is OFF, and $\leq 012_{(8)}$, relay 000300 goes OFF.

Compare register with constant (1 byte) (with relay output)

Compare register with constant (1 word) (with relay output)

Fc18 CP2		ComPa	re >)	Fc18 CP>		ComPare >=)	
Fc18 CP<		ComPa	re <)	Fc18 CP>		ComPare <=)	
Fc18 CP		ComPa	re =)	Fc18 CP<		ComPare <>)	
Svr	nbol		$- \begin{array}{ c c c c c } \hline Fc180w & S & n & BIT \\ \hline CP > & S & n & BIT \\ \hline Fc181w & S & n & BIT \\ \hline \end{array}$	$- \frac{Fc183w}{CP \ge} S n BIT$			
			$-\frac{Fc182w}{CP=} S n BIT$	CP <= 3 11 BTT $-Fc185w$ $CP <> S n BTT$			
Fund	ction	The con n are co	tents (1-word data) of register mpared in magnitude, and a b	r S, S + 1 bit is turn	and the ed ON wi	content (1-word data) of a constant hen the result is properly obtained.	
		Fc180 w	S, S+1 > n \rightarrow BIT ON		Fc183 w S, S+1 \geq n \rightarrow BIT ON		
Oper	ration	Fc181 w	S, S+1 < n → BIT ON		Fc184 w	S, S+1 ≤ n→ BIT ON	
		Fc182 w	S, S+1 = n \rightarrow BIT ON		Fc185 w	S, S+1 ≠ n→ BIT ON	
S	6	Use ran	ge B	r	ı	Use range 000000 to $177777_{(8)}$	
BI	Т	Use ran	ge K	Condition When the input signal is ON (not limited to OFF to ON change			
	S	Unchar	nged				
		Fc180w	ON (S, S+1 > n) OFF (S, S+1 \leq n)		Fc183w	ON (S, S+1 \geq n) OFF (S, S+1 < n)	
Contents after operation	BIT	Fc181w	ON (S, S+1 < n) OFF (S, S+1 \geq n)		Fc184w	$\begin{array}{l} ON \ (S, S+1 \leq n) \\ OFF \ (S, S+1 > n) \end{array}$	
		Fc182w	ON (S, S+1 = n) OFF (S, S+1 \neq n)		Fc185w	ON (S, S+1 \neq n) OFF (S, S+1 = n)	
	Flag	Zero 007357 0	Carry Error Non-carry 007356 007355 007354 0 0 0				
[Explar	-	1	Instruction S T R 005201 Fc180w 019100	conte the o relay 0123 Wher	ents of re ctal cons 000150 45. n the inpu	at condition 005201 is ON, the gisters (019100 and 019101), and tant 012345 are compared, and the is ON if (019100, 019101) $>$ at condition 005201 is OFF, and 01) \leq 012345, relay 000150 goes	

Fc180d Fc183d (ComPare >)(ComPare >=)CP> CP > =Fc181d Fc184d (ComPare <)(ComPare <=)CP <CP>=Fc182d Fc185d (ComPare =)(ComPare <>)CP =CP <>Fc180d Fc183d S BIT S BIT n n CP> CP> Fc181d Fc184d Symbol S BIT S BIT n n CP< CP<= Fc182d Fc185d s BIT S BIT n n CP<> CP= The contents (2-words data) of register S to S + 3 and the constant n are compared in Function magnitude, and a bit is turned ON when the result is properly obtained. Fc183d Fc180d S to S+3>n \rightarrow BIT ON S to S+3≧n → BIT ON Fc181d S to S+3<n \rightarrow BIT ON S to S+3 \leq n \rightarrow BIT ON Fc184d Operation Fc182d S to S+3=n \rightarrow BIT ON S to S+3 \neq n \rightarrow BIT ON Fc185d Use range S Use range C n 00000000000000 to 377777777777778) When the input signal is ON BIT Use range K Condition (not limited to OFF to ON change) Unchanged S, S+1 ON (S to S+3 > n) ON (S to S+3 \geq n) Fc180d Fc183d OFF (S to S+3 \leq n) OFF (S to S+3 < n) ON (S to S+3 < n) ON (S to S $+3 \leq n$) After Fc181d Fc184d BIT operation OFF (S to S+3 \geq n) OFF (S to S+3 > n) ON (S to S+3 \neq n) ON (S to S+3=n) Fc182d Fc185d OFF (S to S+3=n) OFF (S to S+3 \neq n) Zero Carry 007356 Frror Non-carry 007357 007355 007354 Flag 0 0 0 0 When the input condition 005201 is ON, the [Explanation] Instruction contents of registers (019100 to 019103), and the STR 005201 octal constant 01234567777 are compared, and the Fc180d relay 000150 goes ON if (019100 to 019103) > 019100 005201 Fc180d 01234567777 000150 01234567777. 019100 01234567777 000150 CP> When the input condition 005201 is OFF, and (019100 to 019103) ≤ 01234567777, relay 000150 goes OFF.

Compare register with constant (2 words) (with relay output)



Open Channel (Signal layer: with Octal Station Number)

F-203 OPCH **Open Channel (Signal layer: with Hex Station Number)**

Symbol	- F-202 UN FILE F n - F-203 UN OPCH CH·ST FILE F n]							
Function	Specify the station for inter-PLC data communications to be performed across the satellite net (JW-22CM, JW-20CM). These instructions are used in conjunction with the F-204 (SEND) or F-205 (RCV) instructions.								
UN	Use range 0 to 7: Module number of JW-22CM.								
СН	Use range 0 to 3: Channel number for the specified module number.								
ST	Use range 000 to 377(8), 00 to FF(H): Remote station number. F-202 uses octal notation. F-203 uses hexadecimal notation.								
F	Use range 0, 1: File number of communication target	*							
n	Use range 00000000 to 77777777(8): File address (data top address) of communication target.								
Condition	When input signal is ON (not limited to OFF to ON change).								
Flag	Unchanged.								

When the communication target is the conventional model (JW30H etc.), convert these numbers to file number and file address for the JW300. => See "Indirection address assignment".

- If CH0, CH1, CH2, and CH3 are used as channel numbers, the JW-22CM with module number can be used up to 4 times in a PLC program.

- These instructions must always be used in conjunction with the F-204 (SEND) or F-205 (RCV) instruction.

F-204 SEND

Sends data

Symb	ol	F-204 n S					[Expla	anation]		STR	uction 004000	
Functio	on	Specify the top address and byte count of the sending data across the satellite net.						⁰ F-202 OPCH 2-0-	01 file 0 004000	F -202	2-0-01 file 0 004000	
Operati	ion	S to S+n−1 →	sSpec	cified st	tation						005 009000	
n		Use range 000 to 377(8) (256 bytes for 000(8))						F-204 00 SEND	5 009000			
S		Use range A						When the input condition of 004000 changes from OFF to ON, this instruction sends the contents (5-byte data) of registers 009000 to 009004 to file 0's file area beginning with address 004000 (register 009000) on satellite net station 01, via CH0 of module 2.				
Condit	ion	Rising edge of input signal (OFF to ON)				ON)			7	г	File No. 0	
	S	Unchanged					009000	001	– Module no.2 CH0	009000	001	
						009001		01 009001 5	002			
		No response from port	007357	007330	1	007354	009002	003		station 200600	003	
		Communication	0	0	0	1	009003	004 005	Response	e 009003 009004	004	
Contents after operation	Flag	Communication busy (waiting for response from remote station)	1	0	0	1	003004	Flag				
		Normal end	0	1	0	0		~9				
		Abnormal end (communication error)	0	1	1	0						
		Remote station write protected	1	1	1	0						

For the usage of this instruction, see the user's manual for the network module (JW-22CM).
1 hierarchical communication must always be used in conjunction with the F-202 or F-203,

2 hierarchical communication must always be used in conjunction with the F-206 and F-207.

Receives data

F-205 RCV

Symb	ol	F-205 n	D				[Explanation]		uction			
								S T R F -202	004003			
Functio	~ ~	Specify the count of the	•		-	e	004003 5 000	. 202	2-0-04 file 0			
Functio	JU	transmission				net.	004003 F-202 OPCH 2-0-04 file 0 004000		004000			
Operati	ion	Specified station \rightarrow D to D+n-1				1	F -205					
n	n Use range 000 to 377(8) (256 bytes for 000(8))				F-205 RCV 006 019000 019000							
D		Use range /					When the input condition of 004003 changes from OFF to ON, this instruction reads 6 bytes of data from file 0's file					
Conditi	on	Rising edge of input signal (OFF to ON)					area beginning with file address 004000 on satellite net station 04,via CH0 of module 2. The data read is stored to					
	D	Result					registers 109000 to 019005.					
		Meaning	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	Target station addressing					
		No response from port	0	0	1	0	Module no.2	04	File No. 0			
Contents after		Communication congestion	0	0	0	1	019000 01		00 00 009000 009000 009001 04 009002 07 009002 07			
operation	Flag	Communication busy (waiting for response from remote station)	1	0	0	1	019001 04 019002 07 019002 10	00 se st				
		Normal end	0	1	0	0	019003 10 019004 20		09003 10 09004 20			
		Abnormal end (communication error)	0	1	1	0	019005 30		09005 30			
							Flag					

For the usage of this instruction, see the user's manual for the network module (JW-22CM).
1 hierarchical communication must always be used in conjunction with the F-202 or F-203, 2 hierarchical communication must always be used in conjunction with the F-206 and F-207.



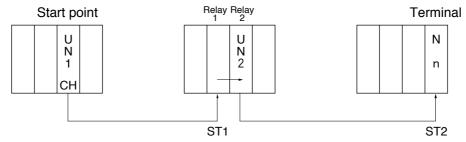


Open channel 1 (set the hierarchical communication)

Open channel 2 (set the hierarchical communication)

Symbol	- F-206 EOP1 UN1, CH ST1 UN2 - F-207 EOP2 ST2 FILE F n							
Function	Specify the station for inter-PLC data communications to be performed across the satellite net (JW-22CM, JW-20CM). The F-206 (EOP1) instructions are used in conjunction with the F-207 (EOP2),							
UN1	Use range 0 to 7: Module no. of module starts from SEND, RECEIVE instructions.							
UN2	Use range 0 to 7: Module no. of module from relay station 2 of SEND, RECEIVE instructions. Note: When the relay station 2 is JW50H/70H/100H, it becomes slot number.							
СН	Use range 0 to 3: Module no. of module starts from SEND, RECEIVE instructions.							
ST1	Use range 00 to 77(8): The relay station 1's station number of SEND, RECEIVE instructions.							
ST2	Use range 00 to 77(8): Terminal station number of SEND, RECEIVE instructions.							
F	Use range 0, 1: File number in the terminal station of SEND, RECEIVE instructions.							
n	Use range 00000000 to 77777777(8): File address in the terminal station of SEND, RECEIVE instructions.							
Condition	When input signal is ON (not limited to OFF to ON change).							
Flag	Unchanged.							

* When the communication target is the conventional model (JW30H etc.), convert these numbers to file number and file address for the JW300. => See "Indirection address assignment".

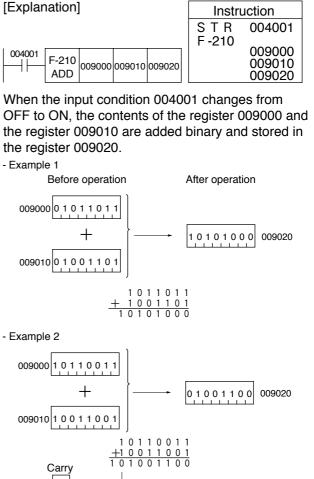


- These instructions must always be used in conjunction with the F-206, F-207, F-204 or F-206, F-207, F-205 instructions.

Add register and register in binary (8 bits + 8 bits) (ADD)

Sym	bol		S2	D			[Expl				
Func	tion	The contents in binary with S2 and its res	00400								
Opera	Operation S1+S2 → D										
S	1	Use range A	Ą				OFF the re the re				
S	2	Use range A	- Exam								
D)	Use range A	009								
Cond	lition	Rising edge of input signal (OFF to ON)									
	S1	Unchanged					009				
	S2	Unchanged									
Contents after	D	Result									
operation		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	- Exarr				
		0	1	0	0	1					
	Flag	001 to 377(8)	0	0	0	1	009				
		400(8)	1	1	0	0					
		Above 401(8)	0	1	0	0					
							1				

F-210 ADD



(007356)

F-21 AD	-	Adds register and regist (ADD)	ter in binary (16 bits + 16 bits)				
Syr	nbol	F-210w S1 S2 D	[Explanation] Instruction S T R 004001				
Fun	ction	The contents of the registers S1 and S1+1 are added in binary with the contents of the registers S2 and S2+1 and its results are stored in the registers D and D+1.	004001 F-210w 009000 009010 009020 ADD 009000 009010 009020 009020				
Ope	ration	(S1, S1+1) + (S2, S2+1) → D, D+1	When the input condition 004001 changes from OFF to ON, the contents of the registers 009000 and				
5	S 1	Use range B *	009001 are added binary with the registers 009010				
5	S 2	Use range B *	and 009011 and its results are stored in the registers 009020 and 009021.				
[C	Use range B *					
Con	dition	Rising edge of input signal (OFF to ON)					
	S1, S1+1	Unchanged	009011 009010				
	S2, S2+1	Unchanged	0010110010110010				
Contents	D	Lower digits of result					
after operation	D+1	Upper digits of result					
	Flag	Result(8) Zero 007357 Carry 007356 Error 007356 Non-carry 007356 0 1 0 0 1 000001 to 177777 0 0 0 1 200000 1 1 0 0 1 Above 200001 0 1 0 0 0					

* Be sure to use even addresses for register S_1 , S_2 , and D (Odd addresses such as 019003 are prohibited).

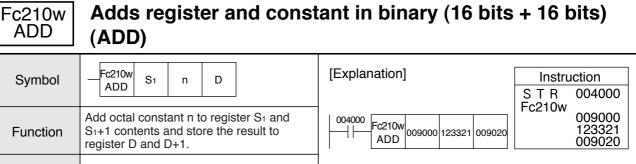
F-210d
ADDAdd register and register in binary (32 bits + 32 bits)(ADD)

Syr	nbol	-F-210d ADD S1	S2	D			[Explanation] Instruction S T R 004001 F -210d				
Fun	Function The contents of the registers S1 to S1+3 are added in binary with the contents of the registers S2 to S2+3 and its result is stored in the registers D to D+3.				s of the re	004001 F-210d ADD 009000 009010 009020					
Oper	ration	(S1 to S1+3) +	(S2 to	S2+3)) → D :	to D+3	When the input condition 004001 changes from OFF to ON, the contents of the registers 009000 to				
5	S1	Use range C *					009003 are added binary with the registers 009010				
S	S 2	Use range C *					to 009013 and its results are stored in the registers 009020 to 009023.				
[D Use range C *						009003 009002 009001 009000				
Con	dition	Rising edge of	f input s	signal	(OFF	to ON)	1 0 0 0 0 1 0 0 1 0 1 0 1 1 0 0 1 0 0 0 0 1 0 0 1 0 1 0 1 1 0 0 L				
	S1 to S1+3	Unchanged									
	S2 to S2+3	Unchanged									
Contents after	D to D+3	Result (32-bit	binary)							
operation				Carry 007356	Error 007355	Non-carry 007354					
	-	0	1	0	0	1	,				
	Flag	0 to 37777777777	0	0	0	1	009023 009022 009021 009020				
		4000000000	1	1	0	0	10110001010111101011000101011110				
		Above 40000000001	0	1	0	0					

* Be sure to use even addresses for register S₁, S₂, and D (Odd addresses such as 019003 are prohibited).

AD	D	(ADD)					,				
Syn	nbol	-Fc210 ADD S1	n D			[Explanation]	Instruction S T R 004000 Fc210				
Fun	ction	Contents of the regis the contents of an o stored in the register	ctal constar	-		004000 Fc210 ADD 009000 377 009020	009000 377 009020				
Operation $S_{1+n} \rightarrow D$						When the input condition 00400	5				
S	S1 Use range A *				OFF to ON, the contents of the register 009000 are binary added with the octal constant 377 and its						
n	n Use range 000 to 377(8)					result is stored in the register 009020.					
E)	Use range A *									
Cond	dition	Rising edge of in	out signal	(OFF t	to ON)						
	S1	Unchanged									
Contents	D	Result				Octal constant 377					
after operation		Result (Octal) Zer		Error 007355	Non-carry 007354						
		0 1	0	0	1						
	Flag	001 to 377 0	0	0	1						
		400 1 Above 401 0	1	0	0						

Fc210 Adds register and constant in binary (8 bits + 8 bits)



Operation

S₁

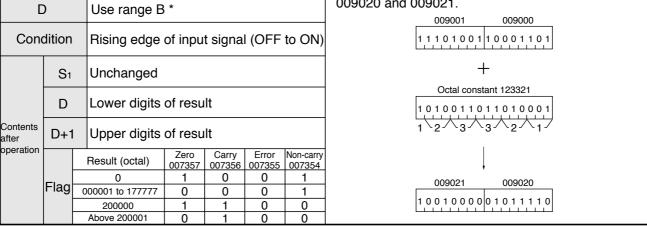
n

 $(S_1, S_1+1) + n \rightarrow D, D+1$

Use range 000000 to 177777(8)

Use range B *

When the input condition 004000 changes from OFF to ON, the contents of the registers 009000 and 009001 are added in binary with the octal constant 123321 and its results are stored in the registers 009020 and 009021.



- Be sure to use even addresses for registers S1 and D (Odd addresses such as 019003 are prohibited).

Fc210d
ADDAdds register and constant in binary (32 bits + 16 bits)(ADD)

Syn	Symbol - Fc210d S1 n D						[Explanation] Instruction			
Fun	Function Contents of the registers S1 to S1+3 are added in binary with an octal constant "n" and its result are stored in the registers D to D+3.						004000 Fc210d ADD 009000 123321 009020	Fc210d 009000 123321 009020		
Oper	ration	(S1 to S1+3)	+ n →	D to D-	+3		When the input condition 00400	00 changes from		
S	1	Use range C *					OFF to ON, the contents of the 009003 are added in binary with	5		
S	S2 Use range 000000 to 177777(8)						123321 and its result are stored in the registers			
C	D Use range C *						009020 to 009023. 009003 009002 009001 009000			
Cond	dition	Rising edge	of inpu	t signa	I (OFF	to ON)	$+ \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0$			
	S1 to S1+3	Unchanged								
Contents	D to D+3	Result (32-b	it binar	y)						
after operation		Result (Octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	ļ			
		0	1	0	0	1		001 000000		
	Flag	1 to 377777777777	0	0	0	1	009023 009022 009			
		4000000000	1	1	0	0	1 1 1 0 1 0 0 1 1 0 0 0 1 1 1 0 0 0 0 1	0 0 0 0 0 1 0 1 1 1 1 0		
		Above 4000000001	0	1	0	0		<u>· · · · · · · · ·</u>		

* Be sure to use even addresses for registers S1 and D (Odd addresses such as 019003 are prohibited).

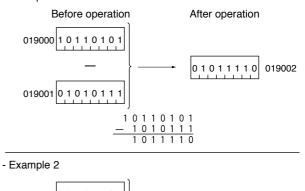
F-211 SUB

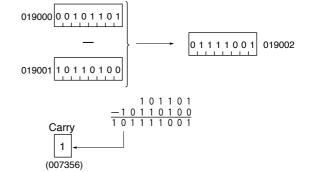
Subtracts register from register in binary (8 bits - 8 bits) (SUBtract)

Symbol		- F-211 SUB S1 S2 D								
Fun	ction	Contents of the register S1 are subtracted by the contents of the register S2 and its result is stored in the register D.								
Oper	ation	$S_1 - S_2 \rightarrow D$								
S	S1	Use range B								
S	62	Use range B								
[C	Use range B								
Conc	lition	Rising edge of input signal (OFF to ON)								
	S1	Unchanged								
Contents	S2	Unchanged								
after operation	D	Result								
		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354				
	Elaa	0	1	0	0	1				
	Flag	001 to 377(8)	0	0	0	1				
		Negative value	0	1	0	0				

[Explanation	าไ	Instruction			
	U	Instruction			
				STR	001000
				F-211	
001000					019000
	1 019000	019001	019002		019001
SUE					019002
	-	-			
When the i	nnut co	nditio	n 0010	00 chanc	les from
OFF to ON	, the co	ontent	s of the	e register	019000 are
subtracted	by the	ne register 019001			
	-			•	

are 1 and its result is stored in the register 019002. - Example 1





Syr	nbol		[Explanation] Instruction S T R 001000					
Fun	ction	Contents of the registers S1 and S+1 are subtracted from the registers S2 and S2+1 in the binary mode and its results are stored in the registers D and D+1.	F-211w SUB 019000 019002 019004 F-211w 019000 019002 019004					
Ope	ration	$(S_1, S_1+1) - (S_2, S_2+1) \rightarrow D, D+1$	When the input condition 001000 changes from					
S	S1	Use range B *	OFF to ON, the contents of the registers 019000 and 019001 are subtracted by the contents of the					
5	S2	Use range B *	registers 019002 and 019003 and its results are					
I	D	Use range B *	stored in the registers 019004 and 019005.					
Con	dition	Rising edge of input signal (OFF to ON	1010001010100010					
	S1, S1+1	Unchanged	_					
	S2, S2+1	Unchanged	019003 019002					
Contents after operation	D	Lower digits of result						
operation	D+1	Upper digits of result						
	Flag	Zero 007357 Carry 007356 Error 007355 Non-carry 007354 0 1 0 0 1						
	i lay	1 to 177777 0 0 0 1 Negative value 0 1 0 0						

- Be sure to use even addresses for registers S1, S2, and D (Odd addresses such as 019003 are prohibited).

F-211d SUB

Subtracts register from register in binary (32 bits - 32 bits) (SUBtract)

Syr	nbol	F-211d SUB S1	S2	D			[Explanation] Instruction S T R 001000 F-211d
Fun	oction	Contents of the subtracted from the binary mod registers D and	n the reg le and its	isters S2	to S2+3		001000 F-211d 019000 019004 019010 SUB 019000 019004 019010 019010
Operation $(S_1 \text{ to } S_1+3) - (S_2 \text{ to } S_2+3) \rightarrow D \text{ to } D+3$						to D+3	· · · · · · · · · · · · · · · · · · ·
5	S1	Use range C)*				OFF to ON, the contents of the registers 019000 to 019003 are subtracted by the contents of the
5	S2	Use range C)*				registers 019004 to 019007 and its results are
	D	Use range C)*				stored in the registers 019010 to 019013.
Con	dition	Rising edge	of inpu	ıt signa	l (OFF	to ON)	019003 019002 019001 019000 1 0 1 0 0 0 0 1 0 1 0 0 0 0 1 0 1 0 1 0 0 0 0 1 0 1 0 1 0 0 0 0 1 0
	S1 to S1+3	Unchanged					_
	S2 to S2+3	Unchanged					019007 019006 019005 019004
Contents after	D to D+3 Result (binary 32-bits)					0 1 1 1 0 0 0 0 1 0 1 1 1 0 0 1 0 1 1 1 0 0 0 1 0 1 1 1 0 0 1	
operation	Result (octal) Zero 007357 Carry 007356 Error 007355 Non-carry 007355		Non-carry 007354				
		0	1	0	0	1	+
	Flag	1 to 37777777777	0	0	0	1	019013 019012 019011 019010
		Negative value	0	1	0	0	

- Be sure to use even addresses for registers S1, S2, and D (Odd addresses such as 019003 are prohibited).

Fc211 SUB

Subtracts register from constant in binary (8 bits - 8 bits) (SUBtract)

Syr	nbol	Fc210 ADD S1	n	D			[Explanation]	Instru S T R Fc211	uction 004000			
Fun	ction	Contents of the the contents of stored in the re	an octal	constan			01900 Ec211					
Oper	ration	S1-n → D					When the input condition 00400	•				
S	S1 Use range A *						OFF to ON, the octal constant 123 in the binary are subtracted from the contents of the register 019000					
r	ı	Use range 0	00 to 3	77(8)			and its result is stored in the rec	gister 009	000.			
[)	Use range A	*									
Con	dition	Rising edge of input signal (OFF to ON)					Before operation					
	S1	Unchange	ed				019000 1 0 1 0 0 0 1 0	After ope	eration			
Contents	D Result			Octal constant	01001	1 1 1 009000						
after operation		Result (Octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354						
		0	1	0	0	1	1 2 0					
	Flag	1 to 377(8)	0	0	0	1						
		Negative value	0	1	0	0						

Fc211w SUB

Subtracts constant from register in binary (16 bits - 16 bits) (SUBtract)

Syr	nbol	Fc211W S1 n D	[Explanation] Instruction S T R 004000 Fc211w
Fun	ction	Contents of the register S_1 , S_{1+1} are subtracted by the octal constant n and its result is stored in the register D, D+1.	004000 Fc211w SUB 019000 123456 009000 123456 009000 123456 009000
Oper	ration	$S_1-S_2 \rightarrow D$	When the input condition 004000 changes from
S	S 1	Use range B *	OFF to ON, the contents of the octal constant 019000 and 019001 are subtracted from registers
S	S2	Use range B 000000 to 177777(8)	123456 and its results are stored in the registers 009000 and 009001.
[D	Use range B *	019001 019000
Cond	dition	Rising edge of input signal (OFF to ON)	0010110011101
	S1 to S1+1	Unchanged	— Octal constant 123456
Contents	D	Lower digits of result	$ \begin{array}{r} 1 0 1 0 0 1 1 1 0 0 1 0 1 1 1 1 0 \\ 1 2 4 3 4 5 6 \end{array} $
after operation	D+1	Upper digits of result	1 2 3 4 5 6
	Flag	Zero 007357 Carry 007356 Error 007355 Non-carry 007354 0 1 0 0 1 1 to 177777(6) 0 0 1 0 0 Negative value 0 1 0 0 0 0	009001 009000 1 0 0 0 0 1 0 1 1 0 1 0 1 1 1 0

* Be sure to use even addresses for register S_1 and D (Odd addresses such as 019003 are prohibited).

Fc211d
SUBSubtracts constant from register in binary (16 bits - 16 bits)SUB(SUBtract)

Syn	nbol	-Fc211d SUB S1 n	D			[Explanation]	Instruction S T R 004000 Fc211d				
Fun	ction	Contents of the register by the contents of the stored in the register I	register Sa								
Oper	ation	(S₁ to S₁+3) — n -	→ D to D	9+3		When the input condition 004000 changes from OFF to ON, the contents of the octal constant 019000 and 019003 are subtracted from registers 023456 and its results are stored in the registers 009000 and 009003.					
S	61	Use range C *									
S	62	Use range 00000) to 177	777(8)							
E)	Use range C *					19001 019000				
Conc	lition	Rising edge of inp	ut signa	I (OFF	to ON)	<u> 0,0,1,0,1,1,0,0 1,1,0,1,1,1,0,0 0,0,1,0,1</u>					
Contents	S1 to S1+3	Unchanged				Octal constant 023456					
after operation	D to D+3	Result (32-bit bin	ary)			$\begin{array}{c} 0.01.001.11.001\\ 0 \\ - 2 \\ - 3 \\ - 4 \\ - $	-5-^-6-				
		Result 007357 007356 007355 007354			Non-carry 007354	Ļ					
	Flag	0 1 0 0 1		009003 009002 00	009001 009000						
		1 to 3777777777(8) 0 Negative value 0	0	0	1 0	0,0,1,0,1,1,0,0,1,1,0,1,1,1,0,0,0,0,0	0010110101110				

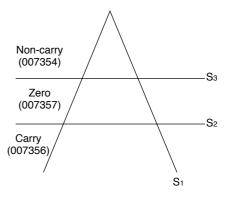
* Be sure to use even addresses for register S1 and D (Odd addresses such as 019003 are prohibited).

F-212 WNDW

Window comparator (between 1-byte registers)

Sym	ibol	F-212 WNDW St	S2	S3			[Explanation] Instruction S T R 002000 F-212					
Func	tion	Compares co contents of th stores its resu	e regist	ers S2 a	and S₃ a		002000 F-212					
Opera	ation	Compare re	sult→fl	ag			When the input condition 002000 changes from					
S	1	Use range A	4				OFF to ON, the contents of the register $\exists 00001$ are checked if $\exists 00001 < \exists 00002$, $\exists 00002 \leq \exists 00002$.					
Sa	2	Use range A	4				\leq ¬00003, and ¬00003 < ¬00001, its results are stored in the carry flag zero flag and non carry flag.					
D	D Use range A						Operation takes place only when the condition is					
Cond	ition	When the in (not limited	put sig to an C	nal is ()FF to (DN DN cha	ange)	\exists 00002 ≤ \exists 00003. Operation does not take place and the error flag is set active, if \exists 00002 ≤ \exists 00003.					
	S1	Unchanged					Input (002000)					
	S2	Unchanged					Register					
Contents after	D	Unchanged					(⊐00002) 200 200 330 (octal)					
operation		Contents of register	Zero 007357	Carry 007356		Non-carry 007354	(⊐00003) <u>300</u> <u>300</u> <u>300</u> (octal)					
		$\begin{array}{c ccccccccccccccccccccccccccccccccccc$										
	Flag	$S_2 \leq S_1 \leq S_3 0 0 0 0$					Non-carry flag					
		S 3< S1 0 0 0 1										
		S3 < S2 0 0 1 0					(007355)					
							Carry flag					

1 scan time



F-212w WNDW

Window comparator (between 1-word registers)

Sym	bol	F-212w WNDW S1	S2	S3			[Explanation] Instruction S T R 002000						
Func	tion	WNDWS1S2S3Compares contents of the registers S1 S1+1 with registers S2, S2+1 and regis S3, S3+1 and stores their result in the registers S3, S3+1 and stores their result in the registers S3, S3+1 and stores their result in the registers 					F-212w 009000 009002 009004 F-212w 009000 009002 009004 009002 009004						
Opera	ation	Compare res	sult → t	ilag			When the input condition 002000 changes from OF to ON, the word contents of the registers 009000 ar						
S	1	Use range B	*				009001 (1 word), are checked if (009000, 009001) <						
S	2	Use range B	*				$(090002, 009003)$ and $(009002, 009003) \le (009000000000000000000000000000000000$						
D)	Use range B	*				< (009000, 009001), and its results are stored in the carry flag, zero flag and non carry flag.						
Cond	lition					ange)	Operation takes place only when (009002, 009003) \leq (009004, 009005) is established and operation						
							stop with the error flag in activation if (009004, 009005) \leq (009002, 009003.)						
	S ₂ Unchanged						, , , ,						
Contents after	D	Unchanged					Input (002000)						
operation		S1 S2	007357	007356	007355	Non-carry 007354 0							
	Flag		0	0	0	0	(009002) 000 000 100 (009003) 200 200 200 200						
		0.0	0	0	0	1	(009004) 000 000 000						
			0	0	1	0	(009005) 300 300 200						
* Be sur	e to us		sses fo	r reais	ters S1	.S2	Non-carry flag						
				•		,	Error flag (007355)						
prohibi	it).						Carry flag						
							Zero flag (007357) 1 scan time						

F-212d WNDW

Window comparator (between 2-word registers)

Sym	ibol	F-212d WNDW S1	S2	S3			[Explanation] Instruction S T R 002000 F-212d
Func	tion	Compares con S1+3 with the S2+3 and stor	registe	rs S2 to	S2+3, 8	S2 to	002000 F-212d 009000 009004 009010 WNDW 009000 009004 009010 009004 009010
Opera	ation	Compare res	sult →	flag			When the input condition of 002000 is ON, this
S	1	Use range C	;*				instruction checks the 2-word contents of registers 009000 to 009003 (2 words) whether they fall in the any of (009000 to 009003) \leq (009004 to 009007) or
S	2	Use range C	, *				$(009004 \text{ to } 009007) \leq (009000 \text{ to } 009003) \leq$
D)	Use range C	; *				(009010 to 009013) or (009010 to 009013) < (009000 to 009003), and sets the carry flag
Cond	lition	When the in (not limited t	put sig o an C	nal is ()FF to (ON ON cha	ange)	accordingly. This instruction is executed only if (009004 to 009007)
	S1 to S1+3	Unchanged					\leq (009010 to 009013). If (009010 to 009013) \leq (009004 to 009007), execution is aborted, and the
	S ₂ to S ₂ +3	Unchanged					error flag is set.
Contents after	S₃ to S₃+3	Unchanged					Input (002000)
operation		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	Register 100000 200100 200100 100000 200100 300100 200100
		S1, S2, S1+1 < S2+1	1	1	0	0	
	Flag		0	0	0	0	(009004 to 009007) 200000 200000 200000 200000
		S3, S1, S3+1 < S1+1 S3, S2,	0	0	0	1	(009010 to 009013) 300000 300000 200000
		$S_{3+1}^{3,} < S_{2+1}^{2,}$	0	0	1	0	Non carry flag
	з (Odd	e even addre addresses su		-		1, S 2	(07354) Error flag (07355) Carry flag (07356)
							Zero flag (07357) 1 scan time

Fc212 WNDW

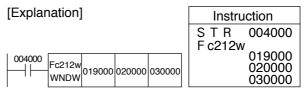
Window comparator (between 1-byte octal constants)

Sym	bol	Fc212 WNDW S	N 1	n2				[Explanation]			STR		0
Func	tion	Compares co the octal cons results in the					004000 Fc212 WNDW	019000 20	00 300	Fc212	01900 20 30	0	
Opera	ation	Result → fla	Result → flag						When the input condition 004000 changes from OFF to ON, the contents of the register 019000 are				
s	1	Use range A	Jse range A									(019000)	
n	1	Use range 000 to 377(8)						300(8), and 30 stored in the c		, .			
n	2	Use range 000 to 377(8)						Contents of 019000 (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	
0	: 4 :	When the in	put sia	nal is C	N			150	0	1	0	0	
Cond	nion	(not limited t				inge)		250	1	0	0	0	
		`				0 /		350	0	0	0	1	
	S1	Unchanged					┢			٨			
Contents after		Contents of register	Zero 007357	Carry 007356	Error 007355	Non-carry 007354				\square			
operation		S1 <n1< td=""><td>0</td><td>1</td><td>0</td><td>0</td><td></td><td>Non-o</td><td></td><td></td><td></td><td></td><td></td></n1<>	0	1	0	0		Non-o					
	Flag	n1≦S1≦n2						(007:	554)			<u> </u>	
		n2 <s1< td=""><td colspan="5">n2<s1 0="" 1<="" td=""><td>Ze</td><td></td><td></td><td></td><td></td><td></td></s1></td></s1<>	n2 <s1 0="" 1<="" td=""><td>Ze</td><td></td><td></td><td></td><td></td><td></td></s1>					Ze					
		n2 <n1< td=""><td>0</td><td>0</td><td>1</td><td>0</td><td></td><td>(007:</td><td></td><td></td><td>\square</td><td><u> </u></td><td></td></n1<>	0	0	1	0		(007:			\square	<u> </u>	
						-	Carry (00735)						

Fc212w WNDW

Window comparator (between 1-word octal constants)

Sym	ibol	Fc212w WNDW S1 n1 n2								
Func	tion	Compares contents of the registers S_1 and S_1+1 with the octal constants n_1 and n_2 , and its results are stored in the flags.								
Opera	ation	Result → flag								
S	1	Use range E	3							
n	1	Use range 000000 to 177777(8)								
n	2	Use range 0	00000	to 177	777(8)					
Cond	lition	When the in (not limited t				inge)				
	S1, S1+1	Unchanged								
Contents after		Contents of register	Zero 007357	Carry 007356	Error 007355	Non-carry 007354				
operation		S1, S1+1 < N1	0	1	0	0				
	Flag	$n_1 \leq S_1, S_{1+1} \leq n_2$	1	0	0	0				
		n2 <s 1,="" s1+1<="" td=""><td>0</td><td>0</td><td>0</td><td>1</td></s>	0	0	0	1				
		N2 < N1	0	0	1	0				



S1

When the input condition 004000 changes from OFF to ON, the word contents of the registers 019000 and 019001 (1word) are checked if (019000, 019001) \leq 020000, 020000 \leq (019000, 019001) \leq 030000, and 030000 < (019000, 019001) and its results are stored in the carry flag, zero flag and non-carry flag.

Contents of 019000.019001	Zero 007357	Carry 007356	Error 007355	Non-carry 007354
015000	0	1	0	0
025000	1	0	0	0
035000	0	0	0	1

Fc212d WNDW

Window comparator (between 2-word octal constants)

Sym	ibol	Fc212d WNDW S1	N 1	n2			[Explanation] [Explanation]					
Func	tion	Compares co to S1+3 (2-wo constants n1 stored in the	ord data and n2,	a) with	the oct	tal						
Opera	ation	Result → fla			OFF to ON, th							
S	1	Use range C	;				019000 to 019003 (2-word) are checked if (019000 to 019003) < 02000000000 , $02000000000 \leq$					
n	1	Use range o	0000000	000 to 3	777777	7777(8)	$(019000 \text{ to } 019003) \leq 03000000000, \text{ and}$ 03000000000 < (019000 to 019003) and its results					
n	2	Use range o	0000000	000 to 3	777777	7777(8)	are stored in t	•		,		
Cond	lition	When the in (not limited t				inge)	flag. Contents of 019000 to 019003	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	
	S1 to	Unchanged					01500000000	0	1	0	0	
Contents	S1+3	Contents of register	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	03500000000	0	0	0	1	
after operation		S1 to S1+3 < n1	0	1	0	0						
	Flag	$n_1 \leq S_1 t_0$ S_1+1 $\leq n_2$	1	0	0	0						
	n2 <s 0="" 1,="" 1<="" s1+1="" td=""><td></td><td></td><td></td><td></td><td></td><td></td></s>											
		N2 < N1	0	0	1	0						

Fx212
WNDWWindow comparator
(between 1-byte hexadecimal constants)

Sym	Ibol	Fx212 WNDW S1	N 1	n2			[Explanation]			STR		0
Func	tion	Compares c with the hex n2 and store	adecim	al con	stants	n₁ and		WNDW	019000 80		Fx212	01900 8 C	30 20
Opera	ation	Result → fla	g				When the input condition 004000 changes from OFF to ON, the contents of the register 019000 are					OFF	
S	1	Use range A	١				checked if (019000) $<$ 80(H), 80(H) \leq (019000) \leq C0(H), and C0(H) $<$ (019000), and its results are stored					tored	
n	1	Use range 0	0 to FF	(8)				the carry flag					
n	2	Use range 00 to FF ₍₈₎						Contents of 019000 (hexadecimal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	
Cond	lition	When the in (not limited t				inae)		70 90	0	1	0	0	
								D0	0	0	0	1	
	S1	Unchanged					_		-	-			
Contents after		Contents of registerZero 007357Carry 007356Error 007355Non-carry 007356								\wedge			
operation		S1 < n1 0 1 0 0						Non-carry					
	Flag	$n1 \leq S1 \leq n2 1 \qquad 0 \qquad 0$						(007	354)	<u> </u>		<u> </u>	
	n2 < S1 0 0 1												
		n2 < n1	0	0	1	0	(007357)n1						

Window comparator (between 1-word hexadecimal constants)

Sym	ibol	Fx212w WNDW S1	n1	n ₂						
Func	tion	S1+1 (1-word)	Compares contents of the registers S ₁ and S ₁₊₁ (1-word) with the hexadecimal constants n ₁ and n ₂ , and its results are stored in the flags							
Opera	ation	Result → fla	g							
S	1	Use range E	3							
n	1	Use range 0000 to FFFF ₍₈₎								
n	2	Use range 0	000 to	FFFF(8	3)					
Cond	lition	When the input signal is ON (not limited to an OFF to ON change)								
	S1, S1+1	Unchanged								
Contents after		Contents of register	Zero 007357	Carry 007356	Error 007355	Non-carry 007354				
operation		S1, S1+1 < n1	0	1	0	0				
	Flag	$n1 \leq S_{1}, S_{1+1} \leq n_2$	1	0	0	0				
		n2 <s 1,="" s1+1<="" td=""><td>0</td><td>0</td><td>0</td><td>1</td></s>	0	0	0	1				
		N2 < N1	0	0	1	0				

Fx212w WNDW

[Explanation]			Instru	iction
				STR Fx212w	004000
004000 Fx212v WNDW	019000	2000	3000		019000 2000 3000

S1

Carry (007356)

When the input condition 004000 changes from OFF to ON, the word contents of the registers 019000 and 019001 (1 word) are checked if (019000, 019001) \leq 2000, 2000 \leq (019000, 019001) \leq 3000, and 3000 < (019000, 019001) and its results are stored in the carry flag, zero flag and non-carry flag.

Contents of 019000,019001	Zero 007357	Carry 007356	Error 007355	Non-carry 007354
1500	0	1	0	0
2500	1	0	0	0
3500	0	0	0	1

Fx212d WNDW

Window comparator (between 2-word hexadecimal constants)

Sym	bol	Fx212d WNDW S1	n1	n ₂			[E)	(planation]			ST		
Func	tion	Compares contents of the registers S_1 to S_1+3 (2-word) with the hexadecimal constants n_1 and n_2 , and its results are stored in the flags.						004000 Fx212d 01900 2000000 3000000 WNDW 019000 20000000 30000000 3000000					
Opera	ation	Result → fla	ıg				When the input condition 004000 changes from OFF to ON, the word contents of the registers						
S	1	Use range C)					(9000 to 019) (9003) < 20)	•	,		•	00,
n	1	Use range 0	000000	00 to F	FFFFF	FFF(H)	$019003) \le 20000000, 20000000 \le (019000, 019003) \le 30000000, and 30000000 < (019000 to 019003) and its results are stored in the carry flag,$						
n	2	Use range 00000000 to FFFFFFFF(H)						zero flag and non-carry flag.					J,
Cond	ition	When the in (not limited t				ange)		Contents of 019000.019001	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	
	S1 to S1+3	Unchanged						15000000 25000000	0	1	0	007354	
Contents		Contents of register	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		35000000	0	0	0	1	
after operation		S1 to S1+3 < n1	0	1	0	0							
	Flag	$n_1 \leq S_1 t_0$ S1+3 $\leq n_2$	1	0	0	0							
		n2 <s 1="" s1+3<="" td="" to=""><td>0</td><td>0</td><td>0</td><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></s>	0	0	0	1							
		N2 < N1	0	0	1	0							

F-215 MUL Multiply register by register in binary (8 bits \times 8 bits) (MULtiply)

Sym	ibol	F-215 MUL	S1 S2	D		[Explanation]	Insti S T R F-215	ruction 004000		
Func	tion		content ar	er S₁ conter nd store the		004000 F-215 MUL 09000 009100 009200		009000 009100 009200		
Opera	ation	S1×S2→	D, D+1			When the input condition 004000 changes from OFF to ON, the contents of 009000 are multiplied by				
S	1	Use rang	e A			the contents of the register 009100 in binary and it results are stored in the registers 009200 and 009				
Sa	2	Use rang	e A			009000				
D)	Use range B					1100			
Cond	Condition Rising edge of input signal (OFF to ON)				FF to ON)	009				
	S1	Unchange	əd) I			
	S2	Unchange	əd			009201 009	200	Result of operation		
Contents after	D	Lower dig	its of resu	ult		000001001010	0100 🔸	operation		
operation	D+1	Upper dig	its of resu	ult						
	Flag	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	10110 <u>× 110</u> 1				
		0	0	0	0	10110 101100 101100	0 0 0			
						$\begin{array}{r} 1 & 0 & 1 & 1 & 0 & 0 \\ \hline 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ \hline 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \end{array}$	0 0			

F-215w MUL

Multiply register by register in binary (16 bits imes 16 bits) (MULtiply)

FunctionBinary multiply register S1, S1+1 content and register S2, S2+1 content and store the result to D to D+3. $f-215w$ 00900 009100 009200 $f-215w$ 00900 009100 009200Operation $(S_1, S_{1+1}) \times (S_2, S_{2+1}) \rightarrow D$ to D+3 $when the input condition 004000 changes fromOFF to ON, the 16-bit contents of the registers009000 and 009001(16-bit data) are multiplied bythe contents of the registers 009100 and 009100(16-bit data) in binary and its results are stored inregisters 009200 to 009203.DUse range B *DUse range C *ConditionRising edge of input signal (OFF to ON)S1Unchanged$
S1 Use range B * S2 Use range C * Ondition Rising edge of input signal (OFF to ON)
S1 Use range B * S2 Use range B * D Use range C * Condition Rising edge of input signal (OFF to ON)
S2 Use range B * (16-bit data)in binary and its results are stored i registers 009200 to 009203. D Use range C * 009001 009000 Condition Rising edge of input signal (OFF to ON) 000001 009000 S1 Use based X
D Use range C * Condition Rising edge of input signal (OFF to ON)
S2 Unchanged 0 0 1 0 0 0 1 0 1 0 0 0 1 0 1
Contents D Lower digits of result
operation D+1 Result
D+2 Result 009203 009202 009201 009200
D+3 Upper digits of result
Zero Carry Error Non-carry 007357 007356 007355 007354

 * Be sure to use even addresses for registers S1, S2 and D.

F-215d MUL

Multiplies register by register in binary (32 bits imes 32 bits) (MULtiply)

009207

In this example, $17341734(H) \times 22852285(H)$ = 0320FC45EF25FR604(H) is performed.

009206

Sym	npol	F-215d MUL	S1 S2	D		[Explanation] Instruction S T R 004000				
Func	ction		er S2 to S2-	er S₁ to S₁- +3 content a 7.		004000 F-215d 009000 MUL 00 19010 009200				
Opera	ation	(S1 to S1+3	3) $ imes$ (S2 to	o S2+3) → I	D to D+7	When the input condition 004000 changes from OF to ON, instruction multiplies the 32-bit contents of registers 009000 to 009003 by the 32-bit contents of registers 009100 to 009103, and stored the result in				
S	1	Use rang	eC*							
S	2	Use rang	e C *			registers 009200 to 009207.				
C	D Use range G *					009003 009002 009001 009000 00010111001101000001011100110100				
Conc	Condition Rising edge of input signal (OFF to ON)				FF to ON)	$\begin{bmatrix} -1 & -7 & -7 & -3 & -4 & -4 & -4 & -7 & -7 & -3 & -4 & -4 \\ \times & & & & & & & \\ & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & $				
	S1 to S1+3	Unchange	ed			<u>009103</u> 009102 009101 009100 00100010100001010000010100001010				
	S2 to S2+3	Unchange	ed							
Contents after	D to D+7	Result (64	4-bit binar	у)		009201 009200				
operation	Flag	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	$1110100000100 \\ - 6 0 4$				
		0	0	0	0	009203 009202				
		e even ad		-		1110111100101000 Result of				
prohib	•	addresses	Such as U	19003 ale		009205 009204				
						F-C-4-5-				

Fc215
MULMultiplies register by constant in binary (8 bits × 8 bits)MUL(MULtiply)

Sym	ibol	Fc215 MUL	S1 S2	D		[Explanation]	Instruction STR 01000 Fc215			
Func	tion			er S₁ conte store the re		001000 Fc215 MUL 009100 123 019000	009100 123 019000			
Opera	ation	$S_1 \times n \rightarrow$	D, D+1			When the input condition 001000 changes from OFF to ON, the contents of the register 009100 are multiplied by the octal constant 123 in binary and its results are stored in the registers 019000 and 019001.				
S	1	Use rang	e A							
S	2	Use rang	e 000 to 3	877(8)						
D)	Use rang	e B							
Cond	lition	Rising edg	ge of input	t signal (O	FF to ON)	× Octal constant 1	23			
	S1 to S1+3	Unchange	əd			0 1 0 1 0 0 1	1			
	S2 to S2+3	Lower dig	jits of resu	ult		_1/_2_/_3				
Contents after	\overline{D}_{+7} Upper digits of result					019001 019000	Result of operation			
operation	Flag	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	010011000110000100				
		0	0	0	0					

Fc215w MUL

Multiplies register by constant in binary (16 bits imes 16 bits) (MULtiply)

Symb	loc	Fc215w MUL	S1 S2	D		[Explanation] Instruction S T R 01000 Fc215w				
Funct	nction Binary multiply register S ₁ , S ₁ +1 content (16-bit data) and octal constant n and store the result to D to D+3.				content n and	001000 Fc215w 009100 006430 019000 MUL 009100 006430 019000 019000				
Opera	tion	(S1, S1+1)	\times n \rightarrow D	to D+3		When the input condition 001000 changes from OFF to ON, the 16-bit contents of the registers				
S1		Use rang	eB*			009100 and 009101 are multiplied by the octal constant 006430 in binary and its results are store				
S2		Use rang	e 000000	to 177777	7 (8)	in the registers 019000 to 019003.				
D		Use range C *				009101 009100 0 0 0 0 0 1 0 1 1 1 0 0 0 1 1 0				
Condi	Condition Rising edge of input signal (OFF to ON)				FF to ON)	Octal constant 006430				
	S1, S1+1	Unchange	əd							
Contents after	D	Lower dig	its of resu	ult						
operation	D+1	Result								
	D+2	Result								
	D+3	Upper dig	its of resu	ult		019003 019002 019001 019000				
	Flag	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 1 1 0 0 1 1 0 0 0 1 0 0 1 0 0 0 0				
	Ū	0	0	0	0					

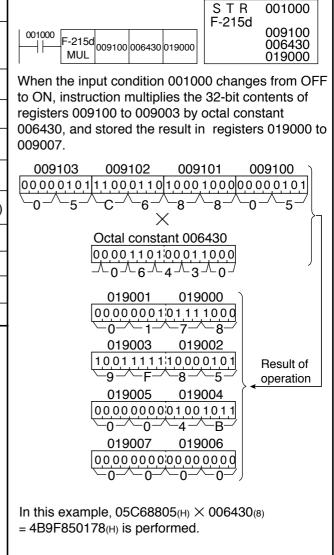
 * Be sure to use even addresses for registers S1 and D.

F-215d MUL

Multiplies register by register in binary (32 bits \times 16 bits) (MULtiply)

Sym	Ibol	F-215d MUL	S1 S2	D		[Explanation]		
Func	FunctionBinary multiply register S1 to S1+3 content (32-bit data) and octal constant n and store the result to D to D+7.							
Opera	ation	(S1 to S1+3	3) × n → I	D to D+7		When the inpute to ON, instruct		
S	1	Use rang	eC*			registers 0091 006430, and s		
S	2	Use rang	e 000000	to 177777	(8)	000430, and s 009007.		
D)	Use rang	eG*			009103 000001011		
Cond	ition	Rising edg	ge of input	t signal (Of	FF to ON)	<u>_05_</u>		
	S1 to S1+3 Unchanged							
Contents after	ifter D+7 Result (64-bit binary)							
operation	Flag	Zero 007357	Carry 007356	Error 007355	Non-carry 007354			
		0	0	0	0	0		

* Be sure to use even addresses for registers S1 and D.(Odd addresses such as 019003 are prohibited).



Instruction

F-216
DIVDivides register by register in binary (8 bits ÷ 8 bits)(DIV)(DIVide)

Sym	ibol	F-216 DIV S1	S2	D			[Explanation]	Instruction STR 010000			
Func	FunctionDivides the 8-bit contents of the register S1 by the 8-bit contents of the register S2 in binary and stores its quotient in the register D and remainder in the register D+1.				er S2 in b	010000 F-216 DIV 009000 009001 009002	F-216 009000 009001 009002				
Opera	ation	S1÷S2→ D, D+1					When the input condition 010000 changes from OFF to ON, the contents of the register 009000 are divided by the contents of the register 009001 in binary and its quotient is stored in the registers				
S	1	Use range A									
S	2	Use range A									
D)	Use range B					009002 and remainder in the re	egister 009003.			
Cond	Condition Rising edge of input signal (OFF to ON)										
	S1	Unchanged					Quotient	0 0 0 0 1 0 0 1 009002			
	S2	Unchanged									
Contents after	D	Quotient	Ur	changed	d if the co	ontents	009001 0 0 0 1 0 1 0 1 0 1 0 1 0 1	0 0 0 0 1 0 1 0 009003			
operation	D+1	Remainder	of	the regis	ster S2 is	000(8)					
		Contents of S2	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	10101) 10101) 10101 10101	<u>0 1</u> 1 1			
	Flag	000 (8)	0	0	0	0	111	0 1			
		Other than above	-		0		10	1 0			

F-21 DI	-	Divides (DIVide	-	giste	r by	regis	ster in binary (15 bits	\div 15 bits)					
Sym	nbol	F-216w DIV St	I S2	D			[Explanation]	Instruction S T R 010000 F -216w					
Func	tion	Divides the 15 +1 by the 15-b S2+1 in binary register D, D+ D+2, D+3.	it conter and stor	nts of the res its qu	e register Jotient in	S _{2,} the	010000 F-216w 019000 019002 019004 019000 DIV DIV 019000 019002 019004 019000						
Opera	ation	(S1, S1+1) - → D, D+1,	•	,			When the input condition0 10000 OFF to ON, the 15-bit contents o	•					
S	1	Use range					019000 and 019001 are divided of the registers 019002 and 0190	by the 15-bit contents					
S	2	Use range	B *				quotient is stored in the registers	•					
D)	Use range	C *				and the remainder in 019006 and						
Cond	lition	Rising edge	of inpu	ut signa	al (OFF	to ON)	1 1 0 1 1 0 0 1 1 1 0 0	100					
	S₁, S₁+1	Unchanged					÷						
	S ₂ , S ₂ +1	Unchanged					019003 01900						
	D	Lower digits of quotient (lower	r)										
	D+1	Upper digits of quotient (uppe		nchange	d if the c	ontents and S2+1	019005 01900	4					
Contents after	D+2	Lower digits of remainder (low	/er) ar	e 00000	0(8)		000000000011						
operation	D+3	Upper digits of remainder (upp		1	1	I	Quotient						
		Contents of S2, S2+1	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	019007 01900	6					
	Flag	000000 (8)	0	0	0	0	0 0 0 0 0 0 0 0 0 0 1 1 0 1 Remainder						
		Other than above		-	0		- MSB (bit 7) of the registers 019 be disregarded.	9001 and 019003 will					

 * Be sure to use even addresses for registers S1, S2 and D.

F-216d
DIVDivides register by register in binary (31 bits ÷ 31 bits)(DIVide)

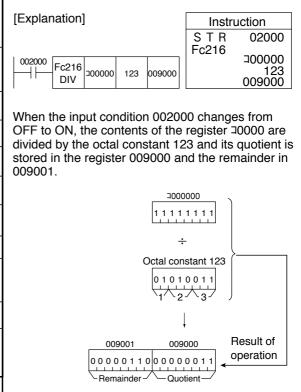
		`	/				· · · · · · · · · · · · · · · · · · ·						
Sym	npol	F-216d DIV S1	S2	D			[Explanation] Instruction S T R 010000 F-216d						
Func	ction	Divides the 31- S1+3 by the 31- S2+3 and store D+3 and remain	bit cont s its que	ents of t	he regist the regis	ers S₂ to ters D to	010000 F-216d 019000 019000 DIV 019000 019004 019010						
Opera	ation	(S1 to S1+3) → D to D+7		2 to S 2·	+3)		When the input condition 010000 changes from						
S	1	Use range () *				OFF to ON, the 31-bit contents of the register 019000 to 019003 are divided by the 31-bit contents of the register 019004 to 019007 in binary and its						
S	2	Use range () *				quotient is stored in the registers 019010 to 019013 and the remainder in 019014 to 019017.						
D)	Use range (3 *				In this example, $6CE46CE4(H) \div 042C042C(H)=1A(H)$ remainder 6C006C is performed.						
Cond	lition	Rising edge	of inpu	it signa	al (OFF	to ON)	019003 019002 019001 019000						
	S₁ to S₁+3	Unchanged											
	S ₂ to S ₂ +3	Unchanged					$\begin{array}{c} (6) \ (C) \ (E) \ (4) \ (C) \$						
Contents	D to D+3	Quotient (31-bit binary				contents 2+3 are	0 0 0 0 1 0 0 0 0 1 0 1 1 0 0 0 0 0 0 1 0 1 0 0 0 0 1 0 1 1 0 0 0 0 0 0 1 0 1 1 0 0 0 0 0 0 1 0 1 1 0 0 0 0 0 0 1 0 1 1 0						
after operation	D+4 to D+7	Remainder (31-bit binary)	0. (ot calcu		Quotient 019013 019012 019011 019010						
		Contents of S2 to S2+3	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0						
	Flag	000000 (8) Other than above	0	0	0	0	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$						
* Be sure to use even addresses for registers S1, S2, and D. (Odd addresses such as 019003 etc. are prohibited to use)							- MSB (bit 7) of the registers 019003 and 019007 will be ignored.						

prohibited to use.)

Fc216 DIV

Divides register by register in binary (8 bits \div 8 bits) (DIVide)

bol	Fc216 DIV S1	1	n	D]		[E:				
tion	the 8-bit conte and stores its	ents quo	of th tient	e registe in the re	er S2 in b	inary	00				
ation	S1÷n → D, D+1										
	Use range A										
	Use range 000 to 377(8)										
	Use range B										
ition	Rising edge of input signal (OFF to ON)										
S	Unchanged										
D	Quotient			Uncha	anged i	if					
D+1	Remainder			"n" =0	000(8).						
	Octal constant "n"			Carry 007356	Error 007355	Non-carry 007354					
Flag	000 (8)		0	0	1	0					
	Other than above		-	Ŭ	0	Ĵ					
	tion ation ition S D D+1	DOIDIVStionDivides the 8-bit content and stores its remainder in ttionS1÷n → DUse rangeUse rangeUse rangeUse rangeUse rangeUse rangeUse rangeUse rangeUse rangeUse rangeDQuotientD+1Remainder "n"Flag000 (%)	DOI DIV S1 DIV DIV S1 tion Divides the 8-bit contents and stores its quo remainder in the r ation S1÷n → D, D. Use range A Use range A Use range B Use range B ition Rising edge of S Unchanged D Quotient D+1 Remainder S Octal constant "n" OCT 000 (%)	DOI DIV S1 n Divides the 8-bit contents of th and stores its quotient remainder in the regist Divides the 8-bit contents of th and stores its quotient to enternance tion S1 ÷ n → D, D+1 Use range A Use range 000 to Use range B Use range dge of input S Unchanged D Quotient D+1 Remainder Flag OCtal constant "n" Zero o07357	DOIDIVS1nDDIVS1nDDivides the 8-bit contents of the register and stores its quotient in the register remainder in the register D+1.tionS1÷n → D, D+1Use range AUse range Q000 to 377(8)Use range 000 to 377(8)Use range BUse range BUse range ADQuotientUnchangedDQuotientUnchangedD+1Remainder"n" =0FlagOctal constant 2000 (8)Zero 000 (8)Carry 00 0	DOIDIVS1nDDIVS1nDDivides the 8-bit contents of the register S2 in b and stores its quotient in the register D remainder in the register D+1.tionS1÷n → D, D+1Use range AUse range Q000 to 377(8)Use range BUse range BItionRising edge of input signal (OFFSUnchangedDQuotientUnchanged in n" =000(8).FlagOctal constant "n"000 (8)00	DOIDIVS1nDDIVS1nDDivides the 8-bit contents of the register S2 in binary and stores its quotient in the register D and remainder in the register D+1.tionS1÷n → D, D+1Use range AUse range 000 to $377_{(8)}$ Use range BUse range BtionRising edge of input signal (OFF to ON)SUnchangedDQuotientUnchanged if "n" =000(8).FlagOctal constant 				



Fc216w DIV

Divides register by register in binary (15 bits \div 15 bits) (DIVide)

Sym	Ibol	Fc216w DIV S1	S2	D			[Explanation] Instruction S T R 020000						
Func	tion	Divides the 8-t S1+1 by the 8- binary and sto D+1 and rema	bit conto res its q	ents of th uotient in	ne registe n the reg	er S₂ in ister D,	002000 Fc216w ⊐00000 073064 009000 073064 009000						
Opera	ation	(S1, S1+1) ÷	÷n→	D, D+ ⁻	1, D+2,	D+3	When the input condition 002000 changes from OFF to ON, the 15-bit contents of the registers						
S	1	Use range E	3 *				□00000 and □00001 are divided by the octal constant						
n	I	Use range (00000	0 to 17	7777(8)		073064 in binary and its quotient is stored in the registers 009000 and 009001 and the remainder in						
D	D Use range C *						009002 and 009003. 						
Cond	Condition Rising edge of input signal (OFF to			al (OFF	to ON)	1111111111111							
	S1, S1+1	Unchanged					Octal constant 073064						
	D	Lower digits of quotient Upper digits of	_		nanged		0111011000110100 $07\sqrt{3}\sqrt{0}\sqrt{6}\sqrt{4}$						
Contents	D+1 D+2	quotient Lower digits of		"n" =	000000) (8).							
after operation	D+2 D+3	Upper digits of remainder					009001 009000						
		Octal constant "n"	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	Quotient						
	Flag	000000(8)	0	0	0								
0 0 0 Other than above 0 0					0	0	- MSB of the register ⊐00001 will be ignored.						

 * Be sure to use even addresses for registers S1 and D.

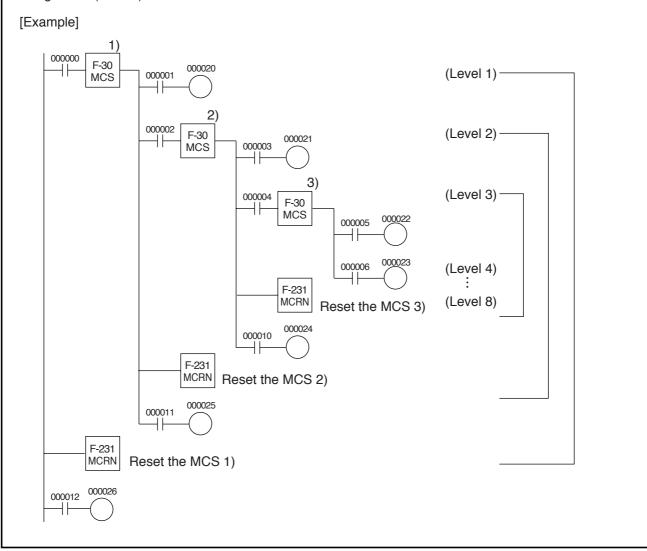
Fc216d
DIVDivides register by register in binary (31 bits ÷ 15 bits)(DIVide)

Sym	npol	Fc216d DIV St	n	D			[Explanation] Instruction S T R 002000						
Fund	tion	Divides the 31- S1+3 by the 31 S2+3 and store D+3 and remai	-bit cont s its que	tents of t	he regist	ers S₂ to ters D to	002000 Fc216d ⊐00000 □073064 009000 073064 DIV □00000 073064 009000 009000						
Opera	ation	(S1 to S1+3)) ÷ n ·	→ D to	D+7		When the input condition 002000 changes from OF to ON, the 31-bit contents of the registers 300000 t 300003 are divided by the octal constant 073064 in						
S	1	Use range	C *										
n	n Use range 000000 to 077777 ₍₈₎						 binary and its quotient is stored in the registers 009000 to 090003 and the remainder in 009004 to 009007. 						
D)	Use range G *					In this example, 7FFF5351(H) \div 73064(8)=11536(H) remainder 2059(H) is performed.						
Cond	lition	Rising edge	of inpu	ut signa	al (OFF	to ON)							
	S1 to S1+3	Unchanged					-7 - F - F - F5 - 3 - 5 - 155						
Contents	D to D+3	Quotient (31-bit binary	/)	Inchanc	ged if "n	" –0	$\begin{array}{c} \text{Octal constant 073064} \\ \hline 0 1 1 1 0 1 1 0 0 0 1 1 0 1 0 0 \\ \hline 7 \hline 3 \hline 0 \hline 6 \hline 4 \end{array}$						
after operation	D+4 to D+7	Remainder (31-bit binary				_0.	Quotient 009003 009002 009001 009000						
	Flog	Octal constant "n"	Zero 007357	Carry 007356	Error 007355	Non-carry 007354							
	Flag	000000 Other than above	0	0	0	0	Remainder 009007 009006 009005 009004						
* Be sure	e to use	e even addre	sses fo	or regis	sters S1	and D.							
						- MSB (bit 7) of the registers ⊐00003 will be ignored.							



Master control reset nesting (Master Control Reset Nesting)

Combined with F-30 (MCS), it is used when the circuit after the common operation condition is branched into plural outputs. Nesting is not enabled in F-31 (MCR), but nesting is enabled in up to 8 levels by using F-231 (MCRN).

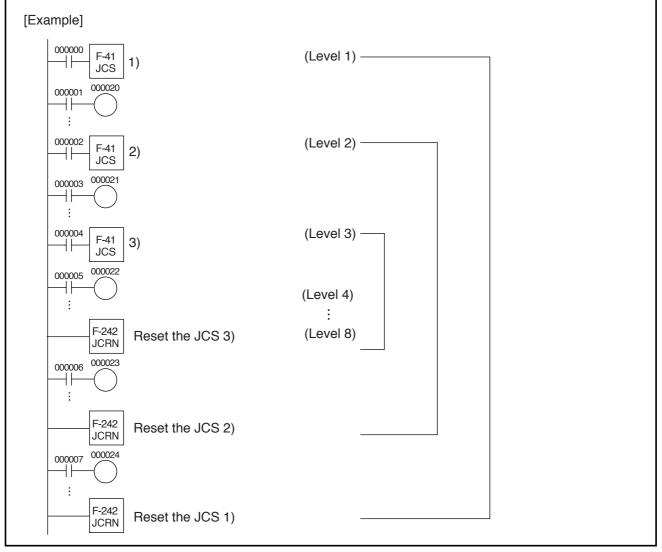


- Refer to the F-30 (MCS), F-31 (MCR).

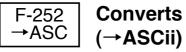
F-242 JCRN

Jump control reset nesting (Jump Control Reset Nesting)

When the condition of F-41 (JCS) is OFF, any of the commands existing in up to F-242 (JCRN) except for END command is not executed. Nesting is not enabled in F-42 (JCR), but nesting is enabled in up to 8 levels by using F-242 (JCRN).



- Refer to the F-41 (JCS), F-42 (JCR).



Converts HEX code into ASCII code

Sym	ibol									
Func	tion	In an area of n bytes starting from register S, converts hexadecimal codes to ASCII codes, and store the results in the area starting from register D. Converts from lower 4 bits of register S.								
Opera	ation	(S, S+1,, S+n-1) → ASCII → D, D+1,, D+2n-1								
S	1	Use range A								
n		Use range 0000 to 1777 ₍₈₎ *1 (When 0000 ₍₈₎ , 1024 bytes)								
D)	Use range A *1								
Cond	lition	Rising edge of input signal (OFF to ON)								
Contents	S1 to S+n-1	Unchanged								
after D to operation D+n-1 Result *2										
-porouon	Flag Unchanged									

*1 Be careful for register area to specify by n and D. The number of bytes used in register of calculation results will be double of hexadecimal code area.

*2: Make sure that the calculation result does not enter the TMR/CNT contact area (file address 00001600 to 00001777(8) etc.) of file No. 0.

[Explar	nation	Instru	uction			
					S T R F-252	000010
000010	F-252 →ASC	009200	0010	009300		009200 0010 009300

When the input condition of 000010 changes from OFF to ON, this instruction converts the 8 ($0010_{(B)}$) bytes of HEX data in registers 009200 to 009207 into their ASCII equivalents, and stores the results to the register area that begins with register 009300.

Before operation		After op	peration	
009200 1 0		3	0	009300
		3	1	009301
009201 3 2		3	2	009302
		3	3	009303
009202 5 4		3	4	009304
		3	5	009305
009203 7 6		3	6	009306
		3	7	009307
009204 9 8		3	8	009310
		3	9	009311
009205 B A	>	4	1	009312
		4	2	009313
009206 D C		4	3	009314
		4	4	009315
009207 F E	+	4	5	009316
		4	6	009317

- Relationship between hexadecimal codes and ASCII codes.

Hex code	0	1	2	3	4	5	6	7	8	9	Α	в	С	D	Е	F
ASCII code	30	31	32	33	34	35	36	37	38	39	41	42	43	44	45	46

Converts ASCII code into HEX code (→HEX)

F-253 →HEX

		` <u> </u>	<u> </u>										
Syr	nbol	– <mark>F-253</mark> S →HEX S	n	D			[Explan	ation]			In S T F-253	R	ction 000010
Fun	ction	Converts the byte register S in and stores t area that be lower 4 bits stored first. convertible the hex cod address of t register D, a	r area t to their gins wi of conv If a hey into AS e is sto he area	hat beg hex equits to t ith regis verted of code CII coo red in t a that b	gins wi quivale he regi ster D. data ar which i de is fo the last begins	th ister The e is not und, t with	When th ON, this of ASCII equivale that beg	instruct data in nts, and	condi tion c regis d stor regis	ition of 000 onverts the sters 00920 es the resul	0 0 change 017(8) by 0 to 0092 ts in the i	es fr tes (16 in regis	15) bytes nto their HEX ster area
Oper	Operation $(S, S+1, \dots, S+n-1) \rightarrow HEX \rightarrow D, D+1, \dots, D+n/2-1$ Unconvertible code $\rightarrow D+n/2-1$ (If "n" is an odd number, the last address=D+(n-1)/2)						009200 009201 009202 009203	3 3 3 3 3	0 1 2 3 4		3	0 2] 009300] 009301] 009303
	S	Use range A	. ,	/			009204 009205	3	4 5		5	4	009302
	n	Use range 0 (0000 specif	000 to ies 102	1777 ₍₈₎ 24 byte) S)		009206 009207	3	6 7		7	6	009303
	D	Use range A	<u>۱</u>				009210	3	8		9	8	009304
Con	dition	Rising edge	of inpu	t signal	(OFF	to ON)	009211 009212 009213	3	9 1 2		В	A	009305
	S to S+n-1	Unchanged					009213	4	3		D	С	009306
	D to D+n/2-1 Result If an unconvertible code is encountered, the code is stored in the last address o					code is	009215 009216	4	4 5		0	E	009307
Contents D+n/2		Unchanged if no			-]				
operation		Conversion	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	 If an une to the la 				s found, t	he c	ode is stored
	Flag	Unconvertible code Other than above	0	0	1 0	0							

* Make sure that the calculation result does not enter the TMR/CNT contact area (file address 00001600 to 00001777₍₈₎ etc.) of file No. 0.

- If "n" is an odd number, the upper 4 bits of the last data will become 0.

- Relationship between hexadecimal codes and ASCII codes.

Hex code	0	1	2	3	4	5	6	7	8	9	Α	в	С	D	Е	F
ASCII code	30	31	32	33	34	35	36	37	38	39	41	42	43	44	45	46

F-260 RTMR

Subtracts timer (setting value, register address)

Syr	mbol	-F-260 RTMR S	D BI	Т		[Explanat	tion]	Instruc S T R F-260	otion 04001					
Fun	iction	The contents of regis timer current value) a every 0.1 sec from th S and S+1 (timer set becoming 0, relay BI the input signal is he	are adde ne conte ting valu T is turn	ed by -1 nts of re ue), and ied ON,	àt gisters when and	After inpu ON, the c	260 TMR 009000 109000 0010000 It condition 004001 ch contents in register 019	anges from 9000 and 0	19001					
Ope	ration	(S.S+1)-Passage BIT (ON) fo	Ì			 (current timer value) are incremented by 1, from the contents of register 009000 and 009001 (timer setting value), every 0.1 second as long as 00400 remains ON. When the current timer value reache 								
5	S	Use range B					0, relay 010000 turns ON. This relay will stay ON							
)	Use range B					as 004001 remains ON.							
B	 IT	Use range K				long do o	009001 009000)						
Condition Starts counting on the r input signal (OFF to ON long as the input signal				d contin	ues as	Before operation	0101011001111000							
	S, S+1	Timer setting valu (BCD 4 digits, 0 to					01010110011110							
Contents before operation	D, D+1	Timer current valu (Same as content		S+1)										
	BIT	OFF					019001 019000							
	S, S+1	Unchanged						(After 0	.1 second)					
	D, D+1	Result the timer co 0000 to 9999 (BC				After operation	019001 019000		67.8					
Contents after operation	BIT	ON (timer current	value =	= 0)				0	seconds)					
operation		Contents of register D Zero 007357					- After 567.8 seconds, - Unchanged for conter	relay 010000 hts of 009000) is ON. 0, 009001					
	Flag	BCD code 0	0	0 0 0		The at	The above operation shows timer setting value							
		Not BCD code		1		is 5678 (567.8 seconds).			9 . 4.40					

- In case of using failure holding mode (refer to the system memory #0201), use latched relay area or register (after b00000 or after 009000) for register D.

- Function is same as decrement TMR instructions.

Input signal	Timer current value	Relay BIT
OFF	Timer setting value	OFF
ON (current value > 0)	Decrease-1 each 0.1 second	OFF
ON (current value = 0)	0	ON

Note: After program input, when the condition is ON and changed to the operation mode while the content of register D is 0, it must be noted that the output relay (BIT) is turned ON.



Subtracts timer (constant, register address)

Syr	nbol	Fc260 n D BIT					[Explanat	ion]	Instr S T R Fc260	uction 004001		
Fun	Function The contents of registers D and D+1 (the timer current value) are added by-1 at every 0.1 sec. from "n" (timer setting value), and when becoming 0, relay BIT is turned ON, and the input signal is held while it is ON.				Image: boot of the content o							
Oper	Operation n -Passage of time \rightarrow (D, D+1) \downarrow BIT (ON) for (D, D+1) = 0				(current timer value) are incremented by 1, from the value of n 5678 (timer setting value567.8 seconds), every 0.1 second as long as 04001			67.8 04001				
r	Timer setting value 0000 to 9999 (0 to 999,9 seconds)				remains ON. When the current timer value reaches 0, relay 010000 turns ON. This relay will stay ON as							
C)					long as 004001 remains ON.						
BI	T	Use range l	K				Before	019001 019000 Before				
Con	dition	Starts countir input signal (long as the in	ng on th OFF to	ON), an	d contin	ues as	operation		8-			
Contents before	D, D+1	Timer curre Same as co		-						0.1 second)		
operation	віт	OFF						01010110011110	8-			
	D, D+1	Result the ti 0000 to 999					After , operation	operation 019001 019000				
Contents after	ΒΙΤ	ON (timer current value = 0)			(After 567.8							
operation		Contents of register D	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		- After 567.8 seconds,	relay 0100	00 is ON.		
	Flag	BCD code	0	0	0	0						
		Not BCD code	0	0	1	0						

- In case of using failure holding mode (refer to the system memory #0201), use latched relay area or register (after b00000 or after 009000) for register D.

- Function is same as decrement TMR instructions.

Input signal	Timer current value	Relay BIT		
OFF	OFF Timer setting value			
ON (current value > 0)	Decrease-1 each 0.1 second	OFF		
ON (current value = 0)	0	ON		

Note: After program input, when the condition is ON and changed to the operation mode while the content of register D is 0, it must be noted that the output relay (BIT) is turned ON.



Subtract counter (setting value, register address)

							r				
Syr	nbol	1) — F-261 2) — RCNT	S	DE	31T		[Explanat	tion]	Instruction S T R 004000 S T R 004001		
FunctionDuring reset input 2) (OFF), contents of registers D and D+1 (counter current value) are added by-1 every time the count input 1) is changed from OFF to ON from the contents of registers S and S+1 (counter set value), and when becoming 0, the relay bit is turned ON and held. In the case of reset input 2) (ON), it follows that the counter current value = counter setting value and the relay BIT (OFF).					er currer ry time ti om OFF rs S and en becol nd held. it follows counter	004000 004001 F-261 RCNT 009000 019000 010000 During reset input 004001 (OFF), contents of registers 019000 and 019001 (counter current value) are added by -1 every time the count input 004000 i changed from OFF to ON from the contents of registers 009000 and 009001 (counter setting value) and when becoming 0, the relay 010000 is turned ON					
		(S, S+1) -cal	culation	input ti	mes → (D, D+1)		• •			
Opei	ration	ВІТ	(ON) fo	↓ pr (D, E	0+1) = 0		and held. In the case of reset input 004001 (ON), it follows that (contents of registers 019000 and 019001) = (contents of registers 009000 and				
S	6	Use range	В				009001) and the relay 010000 (OFF).				
C)	Use range B					The below 5678 time	w operation shows coun	ter setting value is		
BI	Т	Use range K					Soro unes.				
Con	dition		Calculation input 1) at reset input 2) OFF) (OFF → ON)					009001 009000 0 1 0 1 0 1 1 0 0 1 1 1 1 0 0 5 6 7 8			
	S, S+1	Counter se (BCD 4 dig		lue 00	00 to 99	999	Before operation 019001 019000				
Contents before operation	D, D+1	Counter cu (Same as c			S+1)						
	віт	OFF									
	S, S+1	Unchanged	1					019001 019000	1 (Times: 1 times)		
	D, D+1	Result the 0000 to 999									
Contents after	BIT	ON (counte	er curre	nt valu	e = 0)		After . operation	019001 019000 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 - Times: After 5678 times, relay 010000 is ON.			
operation		Contents of register D	Zero 007357	Carry 007356	Error 007355	Non-carry 007354					
	Flag	BCD code	0	0	0	0		- Unchanged for contents of 009000, 009001.			
		Not BCD code	0	0	1	0					

- Function is same as decrement CNT instructions.

Reset input 2)	Counter current value	Relay BIT
ON	Counter setting value	OFF
OFF (current value > 0)	Decrease – 1 each calculation input 1) OFF to ON	OFF
ON (current value = 0)	0	ON

Note 1: After program input, when the reset input is OFF and changed to the operation mode while the content of register D, D+1 is 0, it must be noted that the output relay (BIT) is turned ON.

Note 2: Use the latched relay area or register (after 009000) for register D. In case of using the I/O relay area or ¬xxxxx of auxiliary relay area, output relay (BIT) is ON at power OFF to ON.

Fc261 RCNT

Subtracts counter (constant, register address)

											
Syn	nbol	1) Fc261 n D BIT 2) RCNT n D BIT					[Explana	ation]	Instru S T R S T R	oo4000 004001	
Fun	ction	During reset input 2) (OFF), contents of registers D and D+1 (counter current value) are added by-1 every time the count input 1) is changed from OFF to ON from the contents of registers S and S+1 (counter set value), and when becoming 0, the relay bit is turned ON and held. In the case of reset input 2) (ON), it follows that the counter current value = counter setting value and the relay BIT (OFF).					are added by -1 every time the count input 00400 is changed from OFF to ON from value 5678 of "n"				
		n-calculati	on inpu	it times	s → (D,	D+1)	· ·	out 004001 (ON), it follo			
Oper	ration	BIT	(ON) fo	∓ ⊃r (D, D	0+1) = 0		registers 009000 and 009001) = (value 5678 of "n") and the relay 010000 (OFF).				
n	n Counter set value 0000 to 9999 (0 to 9999 times)		The below operation shows counter setting value is 5678 times.								
D)	Use range	В				5070 times.				
Bľ	Т	Use range	K					019001 019000			
Cond	dition	Calculatior (OFF) (OF	•	,	set inpu	ıt 2)	Before operation				
Contents before	D, D+1	Counter cui (Same as c						019001 019000			
operation	BIT	OFF					After $5 - 6 - 7 - 7$ (Times : 1 times)			: 1 times)	
	D, D+1	Result the t 0000 to 999					operation	019001 019000	,		
Contents after operation	BIT	ON (counte	er curre	nt valu	e = 0)				0 ₀ (Times	: 5678 times)	
operation		Contents of register D	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		- Times: After 5678 time	es, relay 010	0000 is ON.	
	Flag	BCD code	0	0	0	0					
		Not BCD code	0	0	1	0					

- Function is same as decrement CNT instructions.

Reset input 2)	Counter current value	Relay BIT
ON	Counter setting value	OFF
OFF (current value > 0)	Decrease – 1 each calculation input 1) OFF to ON	OFF
OFF (current value = 0)	0	ON

Note 1: After program input, when the reset input is OFF and changed to the operation mode while the content of register D, D+1 is 0, it must be noted that the output relay (BIT) is turned ON.

Note 2: Use the latched relay area or register (after 009000) for register D. In case of using the I/O relay area or ⊐xxxxx of auxiliary relay area, output relay (BIT) is ON at power OFF to ON.

F-263 INC4

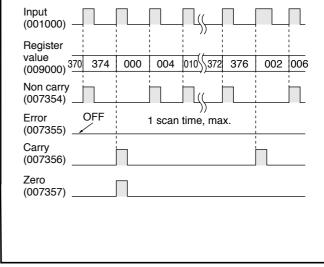
F-263w INC4

Increments counter by (+4) (1-byte binary) (INCrement)

Symbol								
Func	tion	Increment th register D by		ry conte	ents of			
Opera	ation	$\langle D \rangle$ +4 \rightarrow D						
C)	Use range A	۱.					
Condition		Rising edge of input signal (OFF to ON)						
	D	Result (bina	ry code	e)				
Contonto		Register (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		
Contents after		374 → 000	1	1	0	0		
operation	Flag	$\begin{array}{c} 375 \rightarrow 001 \\ 376 \rightarrow 002 \\ 377 \rightarrow 003 \end{array}$	0	1	0	0		
		Other than above	0	0	0	1		

Resembled instructions: F-63, F-63w, F-63d, F-163, F-163w, F-163d, F-263d, F-263w

[Explanation]	Instruction
001000 F-263	S T R 001000 F-263
F-263 INC4 009000	009000
When the input condition of 001 OFF to ON, this instruction incre contents of register 009000 by (ements the binary
Input (001000)	"_ſſſ
Register value (009000) ³⁷⁰ 3740000000	<u>}372 376 002 006</u>
Non carry (007354)	



Increments counter by (+4) (1-word binary) (INCrement)

Sym	ibol	F-263w INC4 D					[Explanation] Instruction S T R 000002			
Fund	tion	Increment the binary contents of registers D and D+1 by (+4).				isters	000002 F-263w INC4 019000 019000 019000			
Opera	ation	n $\langle D, D+1 \rangle + 4 \rightarrow D, D+1$					When the input condition of 000002 changes from			
C	D Use range B * Be sure to use even addresses for registers D. (Odd addresses such as 019003 etc. are prohibited to use.)				sses su	OFF to ON, this instruction increments the binary contents of registers 019000 and 019001 by (+4).				
Cond	Condition Rising edge of input signal (OFF to ON)				Register (009000) (009001) 177 <u>767</u> 177773 177777 000003 000007 000013 000017 (OCT) Non carry					
	D	Lower digits	of resu	ılt			(007354)OFFOFF			
	D+1	Upper digits	of resu	ılt			Carry			
Contents		Register (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	(007357)			
after		177774 → 000000	1	1	0	0	r soan ano, nax.			
operation	Flag	$\begin{array}{c} 177775 \rightarrow 000001 \\ 177776 \rightarrow 000002 \\ 177777 \rightarrow 000003 \end{array}$, v	1	0	0				
		Other than above	0	0	0	1				

Resembled instructions: F-63, F-63w, F-63d, F-163, F-163w, F-163d, F-263, F-263d

F-263d
INC4Increments counter by (+4) (2-word binary)(INCrement)

Sym	ibol		F-263d INC4 D	0				[Explan:	-	Instruction S T R 000002
Func	FunctionIncrement the binary contents of registers D and D+3 by (+4).			000002 F-263d INC4 019000 019000						
Opera	ation		$\langle D, D+3 \rangle$ +4 \rightarrow D, D+3					When the input condition of 000002 changes from OFF to ON, this instruction increments the binary		
D	I		Use range C * Be sure to use even addresses for registers D. (Odd addresses such as 019003 etc. are prohibited to use.)					Contents	s of registers 019000 to	019003 by (+4).
Cond	lition		Rising edge (OFF to ON)		t signa			Register (009000) (009001)	37777777770 3777777774	0000000000 0000000004
	D to D+3)	Result (D: lo	ower to	D+3: I	upper)		Non carry (007354) Error	0FF	Π
			Register (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354			1
			777777774	1	1	0	0	(007356) Zero		1
Contents after operation	Flag	37 → 37	2777777775 00000000001 277777776 00000000002 2777777777 0000000000	0	1	0	0	(007357)	1 scan ti	me, max.
		Ot	her than above	0	0	0	1			

Resembled instructions: F-63, F-63w, F-63d, F-163, F-163w, F-163d, F-263, F-263w

F-264 DEC4

Decrements counter by (-4) (1-byte binary) (DECrement)

Symbol		F-264 DEC4 D						
Func	tion	Increment th register D by		ry conte	ents of			
Opera	ation	$\langle D \rangle$ -4 \rightarrow D						
C)	Use range A	۱.					
Condition		Rising edge of input signal (OFF to ON)						
	D	Result (bina	ry code	e)				
		Register (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354		
Contents after		004 → 000	1	0	0	1		
operation Flag		$\begin{array}{c} 003 \to 377 \\ 002 \to 376 \\ 001 \to 375 \\ 000 \to 374 \end{array}$	0	1	0	0		
		Other than above	0	0	0	1		

[Explanation]	Instruction
000100 F-264	S T R 000100 F-264
DEC4 009000	009000
When the input condition of 000 OFF to ON, this instruction decr contents of register 009000 by (rements the binary
Input ON (000100) Register OFF value (009000) 010004 000 374 370 Non carry (007354) Error OFF 1 scan (007355) Carry (007356))
Zero (007357)	

Resembled instructions: F-64, F-64w, F-64d, F-164, F-164w, F-164d, F-264w,

F-264d



Decrements counter by (-4) (1-word binary) (DECrement)

Sym	Symbol F-264W D DEC4 D						[Explanation] Instruction S T R 000002			
Func	tion	Decrements registers D a	the bir and D+	ary co 1 by (-4	ntents 4).	of	000002 F-264w DEC4 019000 019000			
Opera	Operation $(D, D+1) - 4 \rightarrow D, D+1$									
)	Use range B * Be sure to registers D 019003 etc	use ev . (Odd	addres	SSES SL	ich as	When the input condition of 000002 changes from OFF to ON, this instruction decrements the binary contents of registers 019000 and 019001 by (-4).			
Cond	ition	Rising edge (OFF to ON)		t signa	l					
	D	Lower digits	of resu	ılt			(007354)			
	D+1	Upper digits	of resu	ılt			Carry			
Contents after		Register (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	(007357) 1 scan time, max.			
operation		000004 → 000000	1	0	0	1				
	Flag	000003 → 177777 000002 → 177776 000001 → 177775 000000 → 177774	0	1	0	0				
		Other than above	0	0	0	1				

Resembled instructions: F-64, F-64w, F-64d, F-164, F-164w, F-164d, F-264, F-264d

F-264d
DEC4Decrements counter by (-4) (2-word binary)(DECrement)

Sym	Symbol F-264d D DEC4 D						[Explanation] Instruction S T R 000002			
Function		Decrements registers D t	the bir o D+3	nary co by (-4)	ntents	of	000002 F-264d DEC4 F-264d 019000 019000			
Opera	ation	$\langle D \text{ to } D+3 \rangle$ -4	4 → D	to D+3						
D		Use range C * Be sure to registers D 019003 etc	use e\). (Odd	addres	sses si	ich as	When the input condition of 000002 changes from OFF to ON, this instruction decrements the binary contents of registers 019000 to 019003 by (-4).			
Cond	ition	Rising edge (OFF to ON)	•	t signa	I		Register (009000 to 019003) Non carry			
	D	Result (D: lo	wer to	D+3: u	pper)					
		Register (octal)	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	(007354) OFF (007355) OFF (007355)			
Contents		0000000004 → 0000000000	1	0	0	1	Carry (007356)			
	Flag	0000000003 → 37777777777, 0000000003 → 3777777777, 0000000003 → 3777777777, 0000000003 → 377777777777	0	1	0	0	Zero (007357)			
		Other than above	0	0	0	1				

Resembled instructions: F-64, F-64w, F-64d, F-164, F-164w, F-164d, F-264, F-264w



Transfer 1 byte data

Sym	bol	-F-300 S D	[Explanation]	Instruction S T R 004004		
Func	tion	Transfer register S content (1 byte data) to register D.	004004 F-300 XFER 009000⊐00001	F-300 009000 ¬00001		
Opera	ation	S → D	When the input condition 004004 is ON, the JW300 transfers register 009000 content to register $\neg 00001$. $009000 \qquad \neg 00001$ $0.1.1.1.0.1.0.1 \qquad \longrightarrow 0.1.1.1.0.1.0.1$			
S		Use range A *				
D)	Use range A *				
Cond	ition	When the input signal is ON				
	S	Unchanged				
Contents after operation	D	Contents of register S				
operation	Flag	Unchanged				

* This instruction cannot be used for the file register and indirect specification. Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-70d, F-74w, F-74w, F-74d, F-76, F-76w, F-76d

F-30 XFE	-	Transfer 1 word data				
Sym	ibol	-F-300w XFER S D	[Explanation] 004000 F-300w 009000 □00000 □ When the input condition 004000 is ON, the JW300 transfers register 009000,0090001 content (1-word data) to register □00000, □00001.			
Func	tion	Transfer register S, S+1 content (1 word data) to register D, D+1.				
Opera	ation	S, S+1 → D, D+1				
S		Use range B *				
D	1	Use range B *	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$			
Cond	ition	When the input signal is ON				
	S, S+1	Unchanged				
Contents after	D	Contents of register S				
operation	D+1	Contents of register S+1				
	Flag	Unchanged				

* This instruction cannot be used for the file register and indirect specification.

* Be sure to use even addresses for registers S and D.(Odd addresses such as 019003 are prohibited).

Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-70d, F-74, F-74w, F-74d, F-76, F-76w, F-76d



Transfer 2 word data

Sym	ibol	-F-300d S D	[Explanation]	Instruction S T R 004000 F-300d			
Fund	tion	Transfer register S to S+3 content (2 word data) to register D to D+3.	004000 F-300d XFER 009000 000000	009000 			
Opera	ation	S to S+3 \rightarrow D to D+3	When the input condition 004000 is ON, the JW300				
S	į	Use range C *	transfers register 009000 to 0090003 content (2-word data) to register 700000 to 700003. 009003 009002 009001 009000 01110010000100001				
D)	Use range C *					
Cond	lition	When the input signal is ON					
Contents	S to S+1	Unchanged	ļ				
after operation	D to D+3	Contents of register S to S+3	<u>⊐00003</u> <u>⊐00002</u> <u>⊐00</u> 01110110011010101010101	$\begin{array}{c} 0001 \exists 00000 \\ 0.1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \end{array}$			
	Flag	Unchanged					

* This instruction cannot be used for the file register and indirect specification.

* Make sure to set even addresses of 4 byte units for S and D (009000, 009004, 009010, etc.).

Resembled instructions: F-00, F-00w, F-00d, F-70, F-70w, F-70d, F-74, F-74w, F-74d, F-76, F-76w, F-76d

F-310 SADD

Add registers in binary with sign (31 bits + 31 bits) (Signed ADD)

Sym	ibol	-F-310 SADD S	51 S2	D			[Explanation] Instruction S T R 004001 F-310			
Func	tion	Contents of are added as contents of t and it's resu registers D t	s 31 bi he regi Its are	ts with sters S	sign w S2 to S2	ith the	004001 F-310 009000 009010 009020 When the input condition of 004001 changes from OFF to ON, this instruction adds the binary with sign contents of registers 009000 to 009003 to contents			
Opera	ation	(S1 to S1+3) +	- (S₂ to	S2+3) -	→ (D to) D+3)	of registers 009010 to 009013, and stores the result			
S	1	Use range C	;*				in registers 009020 to 009023. The below example shows $-16+19088743 =$ 19088727.			
S	2	Use range C	;*							
D)	Use range C	;*				Sign 009003 009002 009001 009000			
Cond	lition	Rising edge of	of input	signal	(OFF to	o ON)				
	S1 to S1+3	Unchanged					-16 (D)			
	S ₂ to S ₂₊₃	Unchanged					Sign + 009013 009012 009011 009010			
Contents after	D to D+3	Result (binar	y 31 b	its with	sign)		00000010010011010101010101111			
operation		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354				
		±0	1	0	0	1	Sign			
	Flag	Overflow	0	1	1	0	009023 009022 009021 009020			
		Underflow	0	0	1	1	0 0 0 0 0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0 0 1 0 1 0 1 0 1 0 1 1 1			
		Other than above	0	0	0	1	19088727 (D)			

 * Be sure to use even addresses for registers S1, S2 and D.

- Range of treated numerical value is -2147483648 to 2147483647(D). => (See page binary with sign.)

F-311
SSUBSubtracts registers in binary with sign (31 bits - 31 bits)(Signed SUB tract)

Sym	ibol						[Explanation] Instruction S T R 005001			
Func	tion	Contents of are subtracto with the cont to S ₂ +3 and the registers	ed as 3 tents o it's res	81 bits f the re ults are	with sig gisters	gn s S₂	$\begin{bmatrix} 005001 & F-311 & 019000 & 019010 & 019020 \\ \hline & SUB & 019000 & 019010 & 019020 \\ \hline & SUB & 019000 & 019010 & 019020 \\ \end{bmatrix} \begin{bmatrix} F-311 & 019000 & 019010 & 019010 & 019010 & 019010 \\ \hline & 019020 & 019020 & 019020 \\ \hline & 019020 & 019000 & to & 019003 & to & to & to & the result in registers & 019010 & to & 019003 & to & the result in registers & 019020 & to & 019023. \\ \hline & & 019003 & 019002 & 019001 & 0190000 & 0190000 & 0190000 & 0190000 & 0190000 & 01900000 & 0190000 & 01$			
Opera	ation	(S1 to S1+3) +	- (S₂ to	S2+3) -	→ (D to) D+3)				
S	1	Use range C	;*							
Sa	2	Use range C)*							
D)	Use range C	;*							
Cond	lition	Rising edge of	of input	signal	(OFF t	o ON)				
	S1 to S1+3	Unchanged								
	S2 to S2+3	Unchanged								
Contents after	D to D+3	Result (binar	'y 31 bi	its with	sign)					
operation		Result	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	Sign			
		±0	1	0	0	1	019023 019022 019021 019020			
	Flag	Overflow	0	1	1	0	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1			
		Underflow	0	0	1	1	-219 (D)			
		Other than above	0	0	0	1				

 * Be sure to use even addresses for registers S1, S2 and D

- Range of treated numerical value is -2147483648 to 2147483647(D). => (See page binary with sign.)

F-315 SMUL

Multiplies registers in binary with sign (31 bits imes 31 bits) (Signed MUL tiply)

		· · ·								
Sym	nbol	F-31 SMU		D		[Explanation] Instruction S T R 004201 F-315				
Function Contents of the registers S_1 to S_1+3 are multiplied as 31 bits with sign with the contents of the registers S_2 to S_2+3 and it's results are stored in the registers D to D+7.						004201 F-315 009100 009110 009120 SMUL 009100 009110 009120 009120 When the input condition of 004201 changes from OFF to ON, this instruction multiples the binary				
Opera	ation	(S1 to S1+	\cdot 3) $ imes$ (S ₂ to	o S₂+3) →	(D to D+7)	with sign contents of registers 009110 to 009113 and stores the result in registers 009120 to 009127				
S	1	Use ran	ge C *			and stores the result in registers 0.09120 to 0.09127 . The above example shows $19088743 \times (-2) =$ -38177486. Sign 0.09103 019102 019101 019100 0.0900001 0 0100011 019100				
S	2	Use ran	ge C *							
D)	Use ran	ge G *							
Cond	lition	Rising ec	dge of input	t signal (Of	FF to ON)					
	S1 to S1+3	Unchang	jed			19088743(D)				
	S ₂ to S ₂₊₃	Unchang	jed			Sign ÷				
Contents after	D to D+7	Result (t	oinary 63 b	its with sig	gn)	Image: 019113 019112 019111 019110 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1				
operation	Flag	Zero 007357	Carry 007356	Error 007355	Non-carry 007354	-2(D)				
	1.03	0	0	0	0	Ļ				
and D - Range). e of trea	ated nume	ddresses f erical value e page bin	e is -21474	483648 to	019123 019122 019121 019120 1				

F-316 SDIV

Divides registers in binary with sign (31 bits \div 31 bits) (Signed DIV ide)

									
Sym	lodr	— F-316 SDIV S1	S2	D			Instruction		
- ,		SDIV					R 005201		
Function		Contents of the are divided as the contents of S ₂ +3 and it's quiregisters D to D in the registers	31 bits the req uotient 0+3 and	with si gisters is store d the re	gn with S₂ to ed in the	$\begin{bmatrix} 005201 \\ F-316 \\ SDIV \\ 019100 \\ 019110 \\ 019120 \end{bmatrix} \begin{bmatrix} F-316 \\ 019100 \\ 019110 \\ 019120 \\ 019120 \end{bmatrix}$ When the input condition of 005201 changes from OFF to ON, this instruction divides the binary with sign contents of registers 019100 to 019103 to contents of registers 019110 to 019113, and			
Opera	ation	(S₁ to S₁+3) ÷ (\$	S2 to S2	+3) →	(D to D+7)	stores the result in registers 019120 to 01912 - Example 1			
S	1	Use range C *				The above shows -31536252 ÷ 3600 = -87 remainder -252.	760's quotient and		
S	2	Use range C *				Sign			
C)	Use range G *				019103 019102 019101 019100 1 1 1 1 1 1 1 0 0 0 0 1 1 1 1 0 1 0 0 1 0 1 1 1 0 0 0 0 1 0 0 -31536252(D)			
Cond	lition	Rising edge of in	nput sig	ınal (Of	F to ON)				
	S1 to S1+3	Unchanged				Sign ÷			
	S2 to S2+3	Unchanged				↓ 019113 019112 019111 019110 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0			
	D to D+3	Result (binary 3	1 bits	with sig	ŋn)				
Contents after	D+4 to D+7	Result (binary 3	1 bits	with sig	jn)				
operation	Flag	Contents of register Ze S2 to S2+3 007	ro Cari 357 0073	ry Erro 56 0073		• 019123 019122 0191 Quotient			
		0 0	0	1	0	Quotient 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1			
		Other than above (0	0	0	Sign	05 010101		
and D - Rang). e of tre	e even address ated numerical v 47(D). \Rightarrow (See pa	value is	-2147	483648	Remainder $111111111111111111111111111111111111$	er 252		



Logging instruction

F-403 instruction shall be used for the logging function. => See "Logging function".



SHARP MANUFACTURING SYSTEMS CORPORATION

Information about Sharp programmable controller is available at our internet homepage http://sharp-world.com/sms/

> 1.0v 12.2003 Printed in Japan (0.1I.Y.Y)