廃 番

YAMATAKE

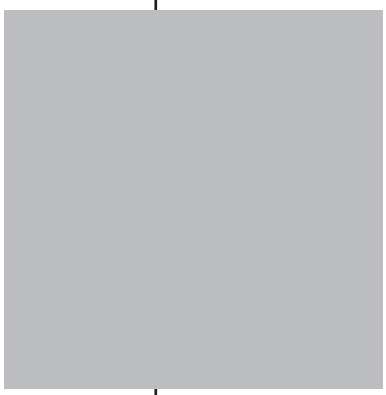
No. CP-UM-1563E

MACHINE CONTROLLER

MX200 / MX100 / MX50 / MX30 / MX20

User's Manual

Programming Instruction Words



Thank you for purchasing the MX200 / MX100 / MX50 / MX30 / MX20.
This manual should be read by those who write programs for the MX200 / MX100 / MX50 / MX30 / MX20.
It provides a basic knowledge for writing programs and a description of the instruction words that are used in creating programs.
See the "Basic Programming Manual No.CP-UM-1562E" for the information on programming procedures.
Be sure to keep this manual near by for handy reference.

Yamatake Corporation

廃 番

RESTRICTIONS ON USE

This product has been designed, developed and manufactured for general-purpose application in machinery and equipment. Accordingly, when used in the applications outlined below, special care should be taken to implement a fail-safe and/or redundant design concept as well as a periodic maintenance program.

- Safety devices for plant worker protection
- Start/stop control devices for transportation and material handling machines
- Aeronautical/aerospace machines
- Control devices for nuclear reactors

Never use this product in applications where human safety may be put at risk.

REQUEST

Make sure that this Instruction Manual is handed over to the user before the product is used.

Copying or duplicating this Instruction Manual in part or in whole is forbidden. The information and specifications in this Instruction Manual are subject to change without notice.

Considerable effort has been made to ensure that this Instruction Manual is free from inaccuracies and omissions.
If you should find any inaccuracies or omissions, please contact Yamatake Corporation.

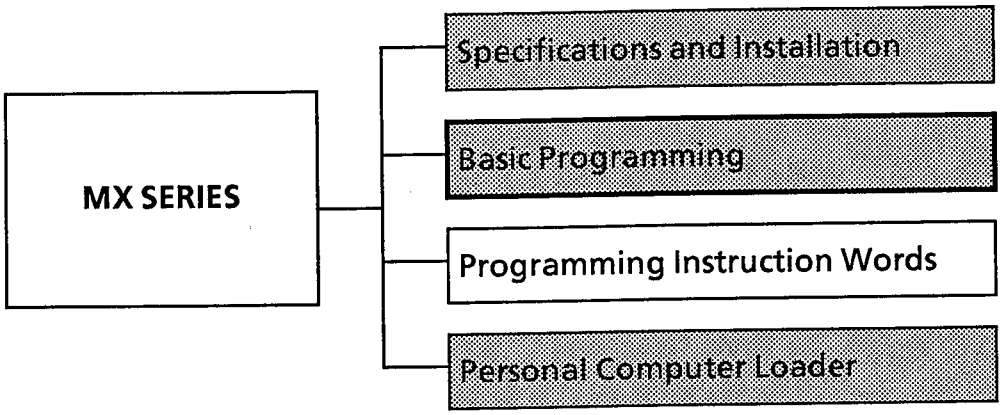
In no event is Yamatake Corporation liable to anyone for any indirect, special or consequential damages as a result of using this product.

廃 番

廃 番

The Role of This Manual

This user's manual describes instruction words programming procedures for the MX200 Super Machine Controller, the MX100 Machine Controller, the MX50 Board Machine Controller, the MX30 One-board Machine controller and the MX20 Compact-board Machine controller.
These procedures are the same for the whole MX series, the MX200, MX100, MX50, MX30 and MX20.
All the manuals that accompany the MX200 Super Machine Controller are listed below and followed by a brief outline.



Specifications and Installation **CP-UM-1604E**

This manual is for first-time users of the MX200 Super Machine Controller, hardware designers in charge of integrating it in control panels and maintenance personnel.
This manual is an introduction to the MX200 controller and the line of products that can be used with it. It details procedures for installing and wiring, maintenance inspections, troubleshooting and hardware specifications.

Basic Programming **CP-UM-1562E**

This manual contains the basics of program writing and programming techniques and provides information on the register and memory structure of the MX200, MX100, MX50, MX30, and MX20.

Personal Computer Loader **CP-UM-1602E**

This manual is required reading for those who write programs for the MX200 Super Machine Controller.
This manual comes with a loader software package supplied on floppy disks. The loader software package supports the MX200, MX100, MX50,MX30 and MX20 and runs on a PC-9801 series personal computer. The manual describes how to create an "execution system disk" and how to operate the loader.

廃 番

Configuration of the User's Manual

This user's manual consists of five chapters, in which the respective items are described as shown below.

Chapter 1 TYPES OF MX SERIES INSTRUCTIONS

This chapter describes the type of instruction words that are provided.

Chapter 2 RANGE OF USABLE DEVICES

This chapter gives the area of devices that can be used with the MX200, MX100, MX50, MX30 and MX20 controllers.

Chapter 3 LIST OF INSTRUCTION WORDS

This chapter describes the meaning of terms used in chapters 4 and 5.

Chapter 4 LIST OF INSTRUCTION WORDS FOR THE MX SERIRS

This chapter provides the symbols and meaning of each instruction word, the devices that can be used with them and their execution conditions.

Chapter 5 DESCRIPTION OF INSTRUCTION WORDS

Ladder circuit examples, descriptions and notes for each instruction word are provided.

廃 番

Contents

廃 番

Chapter 1 TYPES OF MX SERIES INSTRUCTIONS

The types of instructions used by the MX series are listed in the table below.

Instructions		Description	FUN NO
Basic instruction		One-bit instruction for "a" and "b" contacts, output coils, timer and counter	
Compare instruction		"a" contact instruction used to compare word data and double-word data items.	
Application instructions	Transfer instruction	Instruction for data transfer, distribution and extraction	00 to 08
	BCD arithmetic instruction	Arithmetic operation instruction for BCD data	10 to 19
	Semi-basic instruction	Instruction for pulse output, hold output, inversion and application timer and counter	20 to 28
	Branch instruction	Instruction for circuit connections such as master control and jump	30 to 44
	Conversion instruction	Instruction for BCD, BIN and other conversions	50 to 59
	Shift instruction	Instruction for bit shifts in data	60 to 66
	BIN arithmetic instruction	Instruction for arithmetic operations of BCD data	70 to 79
	Logical operation instruction	Instruction for logic data operations	80 to 83
	Special instruction	Timer and counter data read and write operations and I/O refresh	90 to 99
*Expanded application instructions	Register bit processing instruction	Instruction for processing word devices in bit increments	100 to 108
	Expanded transfer instruction	Instructions for writing data tables and performing word I/O module operations	110 to 117
	ASCII communication instruction	Instruction for starting continuous read/write operations and defining communication nodes	120 to 122
	Character string processing instruction	Instruction for character string transfer, compare, concatenation and extraction	130 to 138
	Character string processing instruction	Instruction for converting and retrieving character strings	140 to 148
	Expanded auxiliary instruction	Instruction for reading internal flags and operation error data	150 to 151
	Expanded special instruction	Instruction for calculating SUM	160
	Library instruction	Instruction for starting the library	180

* The expanded instructions can be used only by the MX200 and MX50 controller.

廃 番

Chapter 2 RANGE OF USABLE DEVICES

The following pages lists usable devices.

Area	Name of area	Controller model	Number of points	Address range
External relays	Input relay	MX200	320	X000 to X19F
		MX100	160	X000 to X09F
		MX50	320	X000 to X19F
			256	X000 to X15F
		MX30	64	X000 to X03F
		MX20	12	X000 to X00B
	Input link for OP link	MX200	160	X200 to X29F
		MX100	160	X200 to X29F
		MX50	160	X200 to X29F
	Input relay for remote I/O	MX30	160	X200 to X29F
	Output relay	MX200	320	Y000 to Y19F
		MX100	160	Y000 to Y09F
		MX50	320	Y000 to Y19F
			256	Y000 to Y15F
		MX30	64	Y000 to Y03F
		MX20	12	Y000 to Y00B
	Output relay for remote I/O	MX200	160	Y200 to Y29F
		MX100	160	Y200 to Y29F
		MX50	160	Y200 to Y29F
	Output relay for OP link	MX30	160	Y200 to Y29F
Internal relays	Auxiliary relay	MX200	1440	M000 to M89F
		MX100	800	M000 to M49F
		MX50	1440	M000 to M89F
		MX30	800	M000 to M49F
		MX20	800	M000 to M49F
	Latched relay	MX200	800	L000 to L49F
		MX100	800	L000 to L49F
		MX50	800	L000 to L49F
		MX30	800	L000 to L49F
		MX20	800	L000 to L49F
	Special relay	MX200	160	M900 to M99F
		MX100	160	M900 to M94F, M980 to M99F
		MX50	160	M900 to M99F
		MX30	160	M900 to M94F, M980 to M99F
		MX20	160	M900 to M94F, M980 to M99F
Timer/counter		MX200	256	T/C000 to T/C255
		MX100	200	T/C000 to T/C199
		MX50	256	T/C000 to T/C255
		MX30	200	T/C000 to T/C199
		MX20	200	T/C000 to T/C199

Description
Assigns DI modules.
In case of without high speed counter model.
In case of with high speed counter model.(X160 to X19F applied to HSC)
Assigns DI modules.
Assigns OP I/O, ST series devices.
Assigns remote CBL inputs.
Assigns DO modules.
In case of without high speed counter model.
In case of with high speed counter model.(Y160 to Y19F applied to HSC)
Assigns DO modules.
Assigns OP I/O, ST and other devices.
Assigns remote CBL outputs.
Area for strong bit data which is cleared when the power is turned off.
Battery backed up area for strong bit data.
Indicates alarm and error conditions.
Indicates when the time and counter reaches a set value. The set value is shown when the time and counter are used with the WTCR and RTCS instructions, while the current value is shown when they are used with the WTCA and RTCA instructions.

Area	Name of area	Controller model	Number of points	Address range
Resisters	Input resister	MX200	20W	R0500 to R0519
		MX100	20W	R0500 to R0519
		MX50	20W	R0500 to R0519
			12W	R0508 to R0519
		MX30	—	—
		MX20	—	—
	Output resister	MX200	20W	R0600 to R0619
		MX100	20W	R0600 to R0619
		MX50	20W	R0600 to R0619
			12W	R0608 to R0619
		MX30	—	—
		MX20	—	—
	Data resister	MX200	4500W	R0000 to R0499, R1000 to R4999
		MX100	500	R0000 to R0499
		MX50	4500W	R0000 to R0499, R1000 to R4999
		MX30	500	R0000 to R0499
		MX20	500	R0000 to R0499
	Link resister	MX200	4000W	P0000 to P3999
		MX100	—	—
		MX50	4000W	P0000 to P3999
		MX30	—	—
		MX20	—	—
	Special resister	MX200	100W	R0900 to R0999
		MX100	40W	R0900 to R0939
		MX50	100W	R0900 to R0999
		MX30	40W	R0900 to R0939
		MX20	40W	R0900 to R0939
Data table		MX200	9000	#0000 to #9998
		MX100	6000	#0000 to #5999
		MX50	9000	#0000 to #9998
		MX30	6000	#0000 to #5999
		MX20	6000	#0000 to #5999

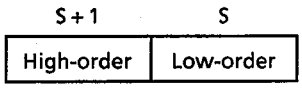
Description
Assigns special function modules.
In case of without high speed counter model.
In case of with high speed counter model.(R0500 to R0507 applied to HSC)
—
Assigns special function modules.
In case of without high speed counter model.
In case of with high speed counter model.(R0600 to R0607 applied to HSC)
Assigns special function modules.
Battery backed up area for storing word data. (R0600 to R0619 on the MX30 and MX20 can be used as a data resister)
Data resister for CBL communications.
Stores all types of setting data.
Area for string word data. This area is stored in the program memory of MX100, MX30, MX20 and connot be overwritten when the program is executed.

廃 番

Chapter 3 LIST OF INSTRUCTION WORDS

① Category : Instruction category
② FUN No. : The number assigned to each instruction
③ Instruction Word : Instruction mnemonic
④ Symbol : Symbol used for each instruction
⑤ Description : Indicates the process performed by the instruction

- S and D indicate 16-bit data (depends on the bit count, see 7. below).
- (S + 1, S) and (D + 1, D) indicate 32-bit data (depends on the bit count, see 7. below). In the case of 32-bit data, S + 1 and D + 1 are high-order 16-bit data and S and D are low-order 16-bit data.



For example, when S is R0000, then (S + 1, S) indicate R0001 and R0000,as shown below.

⑥ Register (device specifications)
Z : I/O relay
S : Source of operation
D : Destination of operation
n : Constant
⑦ Bit count : Indicates the bit count of the above registers.
⑧ Usable devices : The symbol "○" in the column of the following devices indicates that they can store the given instruction in the above registers.
X : Input relay
Y : Output relay
M : Auxiliary relay
L : Hold relay
Special M : Special relay
T/C : Timer/Counter
R : Data register
P : Link register
Input R : Input register
Output R : Output register
Special R : Special register R
Special P : Special register P (for CBL communication specifications)
: BCD specifications or hexadecimal constants
K : Decimal constants

⑨ Flags : "○" indicates that a flags can be modified by an instruction.

900·C	:	Carry flag(M900)
901·OV	:	Overflow flag(M901)
902·U	:	Underflow flag(M902)
90E	:	Operation error flag(M90E)

⑩ Execution conditions : indicates the condition of the execution of each instruction.

Always executed	:	Executed regardless of input
Input ON	:	Executed in each scan during input ON
Input OFF	:	Executed in each scan during input OFF
Rise	:	Executed in a single scan when the input is changed from OFF to ON.
Fall	:	Executed in a single scan when the input is changed from ON to OFF.

⑪ Pulse setting : "○" indicates that the execution condition can be set to rise.

⑫ Indirect specification : "○" indicates that indirect specification is possible.

⑬ Byte count : Indicates the byte count of each instruction.

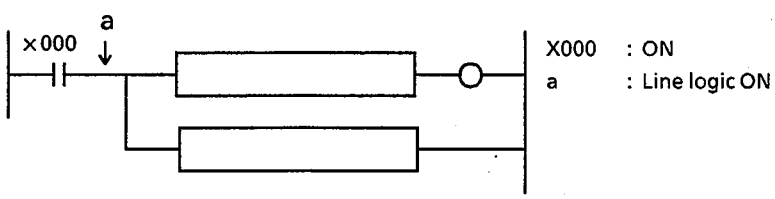
⑭ See page : Reference to pages where further information is provided.

⑮ Handling of word or multi-word data

: Data which can be handled as words or multi-words are indicated as X**0, Y**0, M**0 or L**0.

⑯ Line logic : Indicates logic conditions at arbitrary points in the ladder program.

[Example]

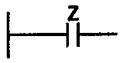
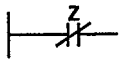
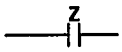
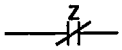
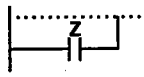
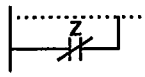
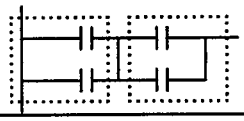
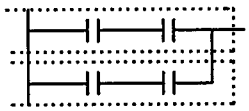
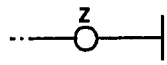
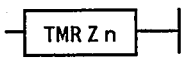
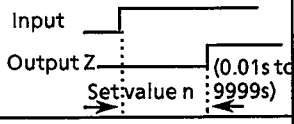
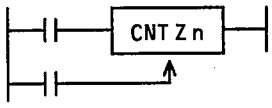
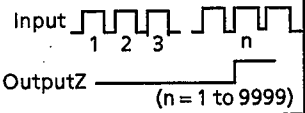
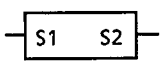
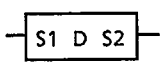


廃 番

Chapter 4 LIST OF INSTRUCTION WORDS FOR THE MX SERIRS

A list of instruction words is given on the following pages.

List of MX series instruction words

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Basic instructions	–	LD		Starts contact "a" operation	Z	1
	–	LD NOT		Starts contact "b" operation	Z	1
	–	AND		Series contact "a" connection	Z	1
	–	AND NOT		Series contact "b" connection	Z	1
	–	OR		Parallel contact "a" connection	Z	1
	–	OR NOT		Parallel contact "b" connection	Z	1
	–	ANB		Inter-block series connection		
	–	ORB		Inter-block parallel connection		
	–	OUT		Output of operation result	Z	1
	–	TMR		On delay timer operation 	Z n	1
	–	CNT		Addition counter 	Z n	1
Compare instructions	–	Compare symbols (18 types)			S1 S2	16 16
	–	Double-word compare symbols (18 types)			S1 S2	32 32

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○	○	○													Always executed	-	-	2	5-2
○	○	○	○	○	○													Always executed	-	-	2	5-4
○	○	○	○	○	○													Always executed	-	-	2	5-5
○	○	○	○	○	○													Always executed	-	-	2	5-6
○	○	○	○	○	○													Always executed	-	-	2	5-7
○	○	○	○	○	○													Always executed	-	-	2	5-8
																		Always executed	-	-	2	5-9
																		Always executed	-	-	2	5-10
	○	○	○	○														Always executed	-	-	2	5-12
					○													Always executed	-	-	4	5-14
												○										
					○													Rise count input	-	-	4	5-16
												○										
○	○	○	○	○		○	○	○	○	○	○	○	○					Always executed	-	-	7	5-18
○	○	○	○	○		○	○	○	○	○	○	○	○									
○	○	○	○	○		○	○	○	○	○	○	○	○					Always executed	-	-	11	5-20
○	○	○	○	○		○	○	○	○	○	○	○	○									

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Transfer instructions	00	MOV	F-00 MOV S:D	Moves 16-bit data (S)→(D)	S	16
					D	16
	01	BMOV	F-01 BMOV S_1,S_2:D	Moves 16-bit data block (S1) to (S2)→(D) to	S1	16
					S2	xn
					D	xn
	02	XCH	F-02 XCH D1,D2	Exchanges 16-bit data (D1)←→(D2)	D1	16
					D2	16
	03	M4BT	F-03 M4BT S:D	Moves 4-bit data (S)→(D)	S	4
					D	4
	04	M8BT	F-04 M8BT S:D	Moves 8-bit data (S)→(D)	S	8
					D	8
	05	DMPX	F-05 DMPX S:D_1 ℓ D_2	Distributes 16-bit data (D2: offset) (S)→(D1 + (D2))	S	16
					D1	16
					D2	16
	06	MPX	F-06 MPX S_1 ℓ D_2:D	Extracts 16-bit data (S2: offset) (S1 + (S2))→(D)	S1	16
					S2	16
					D	16
	07	DCPY	F-07 DCPY S:D_1,D_2	Transfers 16-bit data batches (copy) (S)→(D1) to (D2)	S	16
					D1	16
					D2	16
	08	STB	F-08 STB n:S,D	Moves strobed data to destination D starting from S to the "n-th" byte. (n = 1 to 99)	n	
					S	16
					D	9

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○	○		○	○	○	○	○	○	○	○					Input ON	○	○	7	5-22
	○	○	○	○		○	○		○													
○	○	○	○			○	○	○	○									Input ON	○	○	9	5-24
○	○	○	○			○	○	○	○													
	○	○	○			○	○		○													
	○	○	○			○	○		○									Input ON	○	○	7	5-26
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○									Input ON	○	-	7	5-27
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○									Input ON	○	-	7	5-28
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○					Input ON	○	-	9	5-29
	○	○	○			○	○		○								○					
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○									Input ON	○	-	9	5-30
○	○	○	○			○	○	○	○								○					
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○					Input ON	○	-	9	5-31
	○	○	○			○	○		○													
	○	○	○			○	○		○													
												○						Rise	-	-	9	5-32
○	○	○	○			○	○	○	○													
	○																					

Cate-gory	FUN	Instruction word	Symbol	Description	Argu-ment	Bit count
BCD arith-metic inst-ruc-tions	10	BCD +	F-10 BCD+ $S_1+S_2=D$	BCD addition (S1) + (S2)→(D)	S1	16
					S2	16
					D	16
	11	BCD –	F-11 BCD– $S_1-S_2=D$	BCD Subtraction (S1) – (S2)→(D)	S1	16
					S2	16
					D	16
	12	BCD*	F-12 BCD* $S_1*S_2=D$	BCD multiplication (S1) × (S2)→(D)	S1	16
					S2	16
					D	32
	13	BCD/	F-13 BCD/ S_1/S_2 D	BCD division (S1) ÷ (S2)→(D)	S1	16
					S2	16
					D	32
	14	BCDI	F-14 BCDI D	Addition of 1 to BCD data (D) + 1→(D)	D	16
	15	BCDD	F-15 BCDD D	Subtraction of 1 from BCD data (D) – 1→(D)	D	16
	16	DBC +	F-16 DBC+ $S_1+S_2=D$	Double-word BCD addition (S1 + 1,S1) + (S2 + 1,S2)→(D + 1,D)	S1	32
					S2	32
					D	32
	17	DBC –	F-17 DBC– $S_1-S_2=D$	Double-word BCD subtraction (S1 + 1,S1) – (S2 + 1,S2)→(D + 1,D)	S1	32
					S2	32
					D	32
	18	DBC*	F-18 DBC* $S_1*S_2=D$	Double-word BCD multiplication (S1 + 1,S1) × (S2 + 1,S2)→(D + 1,D)	S1	32
					S2	32
					D	64
	19	DBC/	F-19 DBC/ $S_1/S_2=D$	Double-word BCD division (S1 + 1,S1) ÷ (S2 + 1,S2)→(D + 1,D)	S1	32
					S2	32
					D	64

Usable device														Flags				Execu-tion condi-tions	Pulse speci-fica-tion	Indi-rect speci-fica-tion	Byte count	See page
X	Y	M	L	Spe-cial M	T /C	R	P	Input R	Out-put R	Spe-cial R	Spe-cial P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○			○			○		○	Input ON	○	○	9	5-34
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○				○	○	Input ON	○	○	9	5-35
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○					○	Input ON	○	-	9	5-36
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○					○	Input ON	○	-	9	5-37
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													
	○	○	○			○	○		○						○		○	Input ON	○	-	5	5-38
	○	○	○			○	○		○							○	○	Input ON	○	-	5	5-39
○	○	○	○			○	○	○	○			○			○		○	Input ON	○	-	13	5-40
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○				○	○	Input ON	○	-	13	5-42
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○					○	Input ON	○	-	13	5-44
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○					○	Input ON	○	-	13	5-46
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Semi-basic inst-ruc-tions	20	PLS↑	F-20 PLS↑ Z	Rise pulse — Rise pulse Input: Z:	Z	1
	21	PLF↓	F-21 PLF↓ Z	Fall pulse — Fall pulse Input: Z:	Z	1
	22	INV	F-22 INV	INV ON → OFF OFF → ON	–	–
	23	SET	F-23 SET Z	Contact set	Z	1
	24	RST	F-24 RST D	Device reset	D	1 16
	25	STM	F-25 STM Z n	Accumulation timer Input: t1 t2 t3 ←n = t1 + t2 + t3→ Timer output:		
	26	UDC	F-26 UDC Z n	Addition/subtraction counter Directional input 1:addition、Directional input 0:subtraction (n: 4 digits)	Z n	1
	27	DCNT	F-27 DCNT Z n	Double-word counter Input: 1 2 3 n Counter output: (n: 8 digits)	Z n	1
	28	DUDC	F-28 DUDC 2 n	Double-word addition/subtraction counter Directional input 1:addition、Directional input 0:subtraction (n: 8 digits)	Z n	1

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
	○	○	○															Rise	-	-	5	5-48
	○	○	○															Fall	-	-	5	5-49
																		Input ON	-	-	2	5-50
	○	○	○	○														Input ON	- See note1	-	5	5-51
	○	○	○	○		○	○		○									Input ON	- See note1	-	5	5-52
					○													Always exe-cuted	-	-	7	5-54
												○										
					○													Rise (count input)	-	-	7	5-56
												○										
					○													Rise (count input)	-	-	9	5-58
												○										
					○													Rise (count input),	-	-	9	5-60
												○										

Note 1 Only with MX200

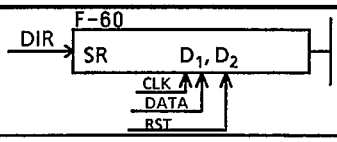
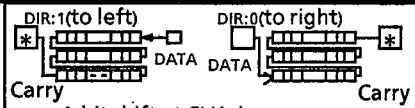
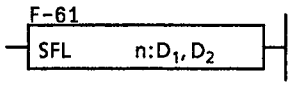
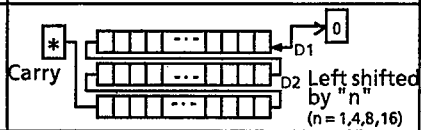
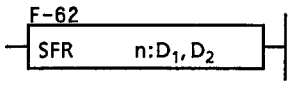
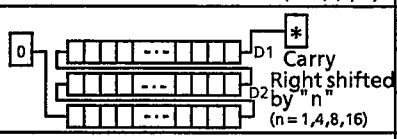
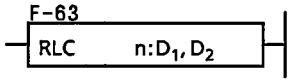
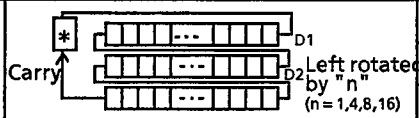
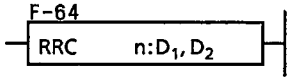
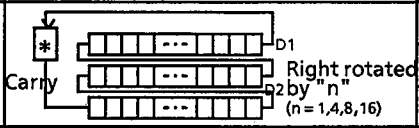
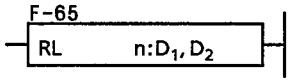
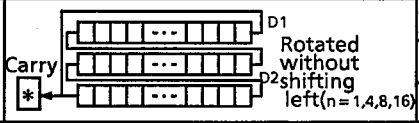
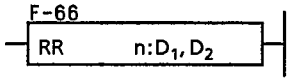
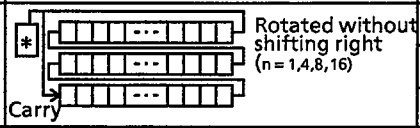
Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Branch instructions	30	MC	F-30 # MC n	Starts master control	n	
	31	MCR	F-31 # MCR n	Releases master control	n	
	32	JMP	F-32 # JMP n	Jumps to specified JPE	n	
	33	JPE	F-33 JPE n	Jump destination of JMP with the same "n" number	n	
	34	LPS	LPS ( )	Stores LPS input line logic temporarily		
	35	LRD	LRD ( )	Reads line logic stored in LPS.		
	36	LPP	LPP ( )	Clears line logic stored in LPS.		
	40	END	F-40 END	Ends program.		
	41	RBP	F-41 RBP	Jumps to the beginning of the program unconditionally.		
	42	CALL	F-42 CALL n	Goes to the subroutine with the specified number "n".	n	
	43	SBR	F-43 SBR n	Starts the subroutine program.	n	
	44	RET	F-44 RET n	Ends the subroutine program.	n	

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T /C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
												○						Input OFF	-	-	5	5-62
												○						Always executed	-	-	5	5-64
												○						Input ON	○	-	5	5-66
												○						Always executed	-	-	5	5-69
																		Always executed	-	-	2	5-70
																		Always executed	-	-	2	5-72
																		Always executed	-	-	2	5-73
																		Always executed	-	-	2	5-74
																		Always executed	-	-	2	5-75
												○						Input ON	○	-	5	5-76
												○						Always executed	-	-	5	5-77
												○						Always executed	-	-	5	5-78

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Conversion instruction	50	→BCD	F-50 →BCD S:D	BCD conversion (S) → (D) BIN(0 to 9999)	S	16
					D	16
	51	→BIN	F-51 →BIN S:D	BIN conversion (S) → (D) BCD(0 to 9999)	S	16
					D	16
	52	CPL	F-52 CPL D	BIT conversion (D) → (D)	D	16
	53	4→16	F-53 4→16 S:D	(S) Decode (D) ↓1 is set; 4 bits(BIN); 00 .. 01000 16 bits	S	4
					D	16
	54	16→4	F-54 16→4 S:D	(S) ↓Address of 1 000001 .... Encode (D); 16 bits; 4 bits(BIN)	S	16
					D	4
	55	7SEG	F-55 7SEG S:D	(S) 7SEG conversion; 4 bits(BIN); 7 bits	S	4
					D	7
	56	TCMP	F-56 TCMP S?T:D	(S) Compare (matching: 1)(T) 16 bits; (D) 16 bits	S	16
					T	
					D	16
	57	→DBC	F-57 →DBC S:D	BCD conversion (S+1,S) → (D+1,D) BIN(0 to 99999999)	S	32
					D	32
	58	→DBI	F-58 →DBI S:D	BIN conversion (S+1,S) → (D+1,D) BCD(0 to 99999999)	S	32
					D	32
	59	→ASC	F-59 →ASC S:D	Converts 4-digit hexadecimal data to ASCII code data (S) 16 bits; (D); (D+1)	S	16
					D	32

廃 番

Usable device														Flags				Execu-tion condi-tions	Pulse speci-fica-tion	Indi-rect speci-fica-tion	Byte count	See page
X	Y	M	L	Spe-cial M	T /C	R	P	Input R	Out-put R	Spe-cial R	Spe-cial P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○								○	Input ON	○	-	7	5-79
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○								○	Input ON	○	-	7	5-80
	○	○	○			○	○		○													
	○	○	○			○	○		○									Input ON	○	-	5	5-81
○	○	○	○			○	○	○	○									Input ON	○	-	7	5-82
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○								○	Input ON	○	-	7	5-84
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○									Input ON	○	-	7	5-86
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○					Input ON	○	-	9	5-88
		○	○			○	○		○													
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○								○	Input ON	○	-	7	5-90
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○								○	Input ON	○	-	7	5-91
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○	○	○	○	○					Input ON	○	-	7	5-92
	○	○	○			○	○		○													

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Shift instructions	60	SR			D1 D2	16 16
	61	SFL			n D1 D2	 16 16
	62	SFR			n D1 D2	 16 16
	63	RLC			n D1 D2	 16 16
	64	RRC			n D1 D2	 16 16
	65	RL			n D1 D2	 16 16
	66	RR			n D1 D2	 16 16

廃 番

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T /C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
	○	○	○			○	○		○					○				Rise (clock input)	-	-	7	5-94
	○	○	○			○	○		○													
												○		○				Input ON	○	-	9	5-96
	○	○	○			○	○		○													
	○	○	○			○	○		○													
												○		○				Input ON	○	-	9	5-98
	○	○	○			○	○		○													
	○	○	○			○	○		○													
												○		○				Input ON	○	-	9	5-100
	○	○	○			○	○		○													
	○	○	○			○	○		○													
												○		○				Input ON	○	-	9	5-102
	○	○	○			○	○		○													
	○	○	○			○	○		○													
												○		○				Input ON	○	-	9	5-104
	○	○	○			○	○		○													
	○	○	○			○	○		○													
												○		○				Input ON	○	-	9	5-106
	○	○	○			○	○		○													
	○	○	○			○	○		○													

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
BIN arithmetic instructions	70	BIN +	F-70 BIN + $S_1+S_2=D$	(S1) + (S2)→D	S1	16
					S2	16
					D	16
	71	BIN −	F-71 BIN − $S_1-S_2=D$	(S1) − (S2)→(D)	S1	16
					S2	16
					D	16
	72	BIN*	F-72 BIN* $S_1*S_2=D$	(S1) × (S2)→(D)	S1	16
					S2	16
					D	32
	73	BIN/	F-73 BIN/ $S_1/S_2=D$	(S1) ÷ (S2)→(D)	S1	16
					S2	16
					D	32
	74	BINI	F-74 BIN D	(D) + 1→(D)	D	16
	75	BIND	F-75 BIND D	(D) − 1→(D)	D	16
	76	DBI +	F-76 DBI + $S_1+S_2=D$	(S1 + 1,S1) + (S2 + 1,S2)→(D + 1,D)	S1	32
					S2	32
					D	32
	77	DBI −	F-77 DBI − $S_1-S_2=D$	(S1 + 1,S1) − (S2 + 1,S2)→(D + 1,D)	S1	32
					S2	32
					D	32
	78	DBI*	F-78 DBI* $S_1*S_2=D$	(S1 + 1,S1) × (S2 + 1,S2)→ (D + 3,D + 2,D + 1,D)	S1	32
					S2	32
					D	64
	79	DBI/	F-79 DBI/ $S_1/S_2=D$	(S1 + 1,S1) ÷ (S2 + 1,S2)→ Quotient(D + 1,D) remainder(D + 3,D + 2)	S1	32
					S2	32
					D	64

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○		○			Input ON	○	○	9	5-108
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○			○		Input ON	○	○	9	5-110
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○					Input ON	○	-	9	5-112
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○				○	Input ON	○	-	9	5-114
	○	○	○			○	○		○													
	○	○	○			○	○		○						○			Input ON	○	-	5	5-116
	○	○	○			○	○		○							○		Input ON	○	-	5	5-117
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○		○			Input ON	○	-	13	5-118
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○			○		Input ON	○	-	13	5-119
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○					Input ON	○	-	13	5-120
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○				○	Input ON	○	-	13	5-121
	○	○	○			○	○		○													

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Logical operation instructions	80	WAND	F-80 WAND S_1&S_2 = D	(S1) ∧ (S2)→(D)	S1	16
					S2	16
					D	16
	81	WOR	F-81 WOR S_1 | S_2 = D	(S1) ∨ (S2)→(D)	S1	16
					S2	16
					D	16
	82	WXOR	F-82 WXOR S_1 ∧ S_2 = D	(S1) ⊻ (S2)→(D)	S1	16
					S2	16
					D	16
	83	WXNR	F-83 WXNR S_1 ! S_2 = D	(S1) $\overline{⊻}$ (S2)→(D)	S1	16
					S2	16
					D	16

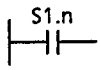
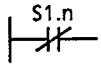
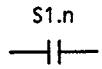
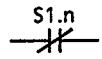
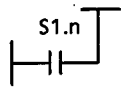
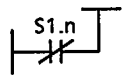
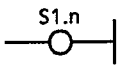
廃 番

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T /C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○			○	○					Input ON	○	○	9	5-122
○	○	○	○			○	○	○	○			○	○									
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○					Input ON	○	○	9	5-123
○	○	○	○			○	○	○	○			○	○									
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○					Input ON	○	○	9	5-124
○	○	○	○			○	○	○	○			○	○									
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○					Input ON	○	-	9	5-125
○	○	○	○			○	○	○	○			○	○									
	○	○	○			○	○		○													

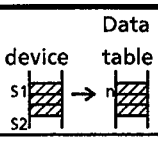
Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Special operation instructions	90	WTCS	F-90 WTCS S: D	Writes timer and counter settings.	S	16
					D	16
	91	RTCS	F-91 RTCS S: D	Reads timer and counter settings.	S	16
					D	16
	92	WTCA	F-92 WTCA S: D	Writes current timer and counter value.	S	16
					D	16
	93	RTCA	F-93 RTCA S: D	Reads current timer and counter value.	S	16
					D	16
	94	STOP	F-94 STOP	Stops program operation.		
	95	WDT	F-95 WDT n	Watchdog timer setting n × 100msec(n = 1 to 20)	n	
	96	ISS	F-96 ISS	Input refresh (X, R0500 to)		
	97	IOS	F-97 IOS	Input ,output refresh (X, R0500 to , Y, R0600 to)		
	98	DTBL Note 2	F-40 END F-98 DTBL n	Data table "n" word settings (n = 1 to 6000) Data table, 16 bits, n	n	
	99	RTBL	F-99 RTBL S_1, S_2:D	Data table read Data table, S1 a, S2 b, D:, b, Word	S1 S2 D	S1 S2 D

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○			○					○	Input ON	○	-	7	5-126
					○																	
					○													Input ON	○	-	7	5-127
	○	○	○			○	○		○													
○	○	○	○			○	○	○	○			○					○	Input ON	○	-	7	5-128
					○																	
					○													Input ON	○	-	7	5-129
	○	○	○			○	○		○													
																		Input ON	-	-	2	5-130
												○						Input ON	-	-	5	5-131
																		Input ON	○	-	2	5-132
																		Input ON	○	-	2	5-133
												○						Always executed	-	-	5	5-134
○	○	○	○			○	○	○	○			○					○	Input ON	○	-	9	5-136
○	○	○	○			○	○	○	○			○										
	○	○	○			○	○		○													

Note 2 Only with MX100 / 30

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Register bit processing instruction	100	RLD		Starts contact "a" operation on bit "n" in register S1.	S1	16
					n	
	101	RLDN		Starts contact "b" operation on bit "n" in register S1.	S1	16
					n	
	102	RAND		Series contact "a" connection of bit "n" in register S1.	S1	16
					n	
	103	RADN		Series contact "b" connection of bit "n" in register S1.	S1	16
					n	
	104	ROR		Parallel contact "a" connection of bit "n" in register S1.	S1	16
					n	
	105	RORN		Parallel contact "b" connection of bit "n" in register S1.	S1	16
					n	
	108	ROUT		Outputs result of bit operation of bit "n" in register S1.	S1	16
					n	

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
						○	○	○	○	○	○							Always executed	-	-	7	5-138
												○										
						○	○	○	○	○	○							Always executed	-	-	7	5-139
												○										
						○	○	○	○	○	○							Always executed	-	-	7	5-140
												○										
						○	○	○	○	○	○							Always executed	-	-	7	5-141
												○										
							○	○	○	○	○							Always executed	-	-	7	5-142
												○										
						○	○	○	○	○	○							Always executed	-	-	7	5-143
												○										
						○	○	○	○	○	○							Always executed	-	-	7	5-144
												○										

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Expanded transfer instructions	110	WTBL	F-110 WTBL S1,S2:n	Writes 16-bit data (blocks) to data table 	S1 S2 n	16 16 16
	115	RSIN	F-115 RSIN A n D	Reads input data no. "n" in slot A.	A n D	16 16 16
	116	RSOT	F-116 RSOT A n D	Reads output data no. "n" in slot A.	A n D	16 16 16
	117	WSOT	F-117 WSOT A n S	Writes input data no. "n" in slot A.	A n S	16 16 16

廃 番

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○	○	○						○	Input ON	○	○	9	5-146
○	○	○	○			○	○	○	○	○	○											
○	○	○	○			○	○	○	○	○	○	○										
○	○	○	○			○	○	○	○			○	○				○	Input ON	○	○	9	5-148
○	○	○	○			○	○	○	○			○	○									
		○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○				○	Input ON	○	○	9	5-150
○	○	○	○			○	○	○	○			○	○									
		○	○			○	○		○													
○	○	○	○			○	○	○	○			○	○				○	Input ON	○	○	9	5-152
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○									

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
ASCII communication instructions	120	SNOD	F-120 SNOD ART	Sets node addresses, number of retries and time-out values for ASCII communications.	A	16
					R	
					T	
	121	SXRS	F-121 SXRS AWD	Starts continuous word read ASCII communications.	A	16
					W	16
					D	16
	122	SXWS	F-122 SXWS AWD	Starts continuous word write ASCII communications.	A	16
					W	16
					D	16

廃 番

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○			○										
												○	○				○	Input ON	○	-	9	5-154
												○										
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○				○	Input ON	○	○	9	5-156
						○																
○	○	○	○			○	○	○	○			○	○									
○	○	○	○			○	○	○	○			○	○				○	Input ON	○	○	9	5-158
						○																

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Character string processing instructions	130	SCPY	F-130 SCPY S D	Copies character string data.	S	16
					D	16
	131	SCAT	F-131 SCAT S1 S2	Concatenates character string data.	S1	16
					S2	16
	132	SCMP	F-132 SCMP S1 S2	Compares character string data.	S1	16
					S2	16
	133	SNCP	F-133 SNCP S1 S2 n	Compares the number of characters in character string data.	S1	16
					S2	16
					n	16
	134	SLEN	F-134 SLEN S D	Checks length of character string data.	S	16
					D	16
	135	SMID	F-135 SMID S W D	Extracts character strings.	S	16
					W	16
					D	16
	136	ATOI	F-136 ATOI S n D	Converts ASCII data to decimal data.	S	16
					n	16
					D	16
	137	ITOA	F-137 ITOA S n D	Converts decimal data to ASCII data.	S	16
					n	16
					D	16
	138	ITSA	F-138 ITSA S n D	Converts decimal data to signed ASCII data.	S	16
					n	16
					D	16

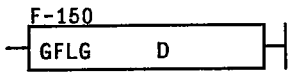
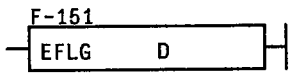
廃 番

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T /C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
						○						○					○	Input ON	○	○	7	5-160
						○																
						○											○	Input ON	○	○	7	5-162
						○						○										
						○						○					○	Input ON	○	○	7	5-163
						○						○										
						○						○					○	Input ON	○	○	9	5-164
						○						○										
	○	○	○			○	○		○			○	○									
						○						○					○	Input ON	○	○	7	5-166
	○	○	○			○																
						○						○					○	Input ON	○	○	9	5-168
						○						○										
						○																
						○											○	Input ON	○	○	9	5-170
						○						○	○									
						○																
						○											○	Input ON	○	○	9	5-172
						○						○	○									
						○																
						○											○	Input ON	○	○	9	5-174
						○						○	○									
						○																

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Character string processing instructions	140	ATOB	F-140 ATOB S n D	Converts ASCII hexadecimal character strings to hexadecimals.	S	16
					n	16
					D	16
	141	BTOA	F-141 BTOA S n D	Converts hexadecimal data to ASCII hexadecimal character strings.	S	16
					n	16
					D	16
	142	CBTA	F-142 CBTA S n D	Converts hexadecimal data to ASCII hexadecimal character strings and concatenates such data.	S	16
					n	16
					D	16
	144	SRCH	F-144 SRCH S1 S2 D	Retrieves character strings.	S1	16
					S2	16
					D	16
	145	CMID	F-145 CMID S W D	Extracts and concatenates character strings.	S	16
					W	16
					D	16
	147	CITA	F-147 CITA S n D	Converts decimal data to ASCII character strings and concatenates such data.	S	16
					n	16
					D	16
	148	CISA	F-148 CISA S n D	Converts decimal data to signed ASCII character strings and concatenates such data.	S	16
					n	16
					D	16

廃 番

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
						○												Input ON	○	○	9	5-176
						○					○	○					○					
						○																
						○												Input ON	○	○	9	5-178
						○					○	○					○					
						○																
						○												Input ON	○	○	9	5-179
						○					○	○					○					
						○																
																		Input ON				
						○					○							Input ON	○	○	9	5-180
						○					○						○					
	○	○	○			○																
						○					○							Input ON	○	○	9	5-182
						○					○						○					
						○																
																		Input ON				
						○												Input ON	○	○	9	5-184
						○					○	○					○					
						○																
						○												Input ON	○	○	9	5-185
						○					○	○					○					
						○																

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Expanded auxiliary instructions	150	GFLG		Reads internal flags.	D	16
	151	EFLG		Reads detailed operation error information.	D	16

廃 番

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T /C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
	○	○	○			○	○		○									Input ON	○	○	5	5-186
	○	○	○			○	○		○									Input ON	○	○	5	5-188

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Expanded special instruction	160	SUM	F-160 SUM S1 S2 D	Bit sum coefficient	S1	16
					S2	16
					D	16

廃 番

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○									Input ON	○	-	9	5-190
○	○	○	○			○	○	○	○													
	○	○	○			○	○		○													

Category	FUN	Instruction word	Symbol	Description	Argument	Bit count
Library instruction	180	LIB	F-180 LIB n A1 A2	Executes libraries.	n	16
					A1	16
					A2	16

Usable device														Flags				Execution conditions	Pulse specification	Indirect specification	Byte count	See page
X	Y	M	L	Special M	T/C	R	P	Input R	Output R	Special R	Special P	#	K	900 C	901 OV	902 U	90E					
○	○	○	○			○	○	○	○		○	○					○	Input ON	○	-	9	5-191
○	○	○	○			○	○	○	○		○	○										
○	○	○	○			○	○	○	○		○	○										

廃 番

Chapter 5 DESCRIPTION OF INSTRUCTION WORDS

■ Guide to description of instruction words

Instruction symbol
Instruction category
Function number
Name of instruction
Condition for execution of instruction operation
Type of execution condition o when pulse specification is enabled.
When indirect specification is enabled.

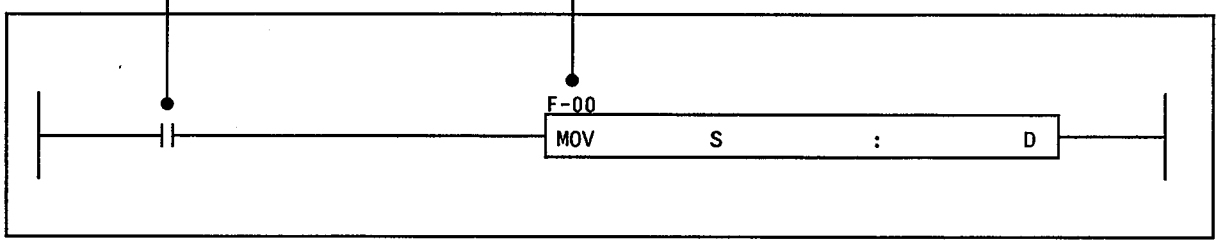
Transfer instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
MOV	FUN 00	Transfer	7Byte		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link Output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to l490	M900 to M970	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	M900 to M910	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	R0900 to R0939	#0000 to #FFFF	K-32768 to K-32767
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Argument (device specification)
Name of device area
Parameter value

Indicates an instruction that requires a contact as an execution condition.
Indication of instruction in ladder (as it is displayed on the screen of the personal computer loader).



● Function description : Describes the function of an instruction word.

● Description : Gives detailed description of how an instruction word is to be used.

◇Notes◇ : Provides important additional information.

[NOTE]

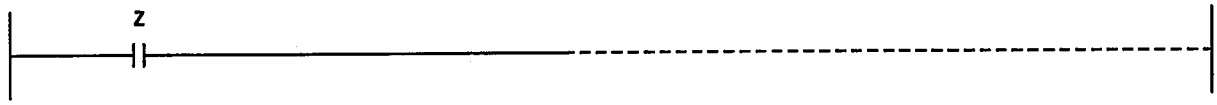
The argument area and parameter range given for each instruction is that of MX200 and MX50. The parameter ranges for the MX100, MX30, and MX20 differ, see Chapter 2 RANGE OF USABLE DEVICES for details.

Instruction									
Instruction		FUNCTION No.	Instruction name	Execution condition	Always executed	Pulse specification	-	Indirect specification	-
LD		FUN	Load	Number of bytes used	2 bytes	Flags	Shift carry M900	-	Underflow M902	-
							Overflow M901	-	Operation error	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	X000 to X19F	X200 to X29F	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example

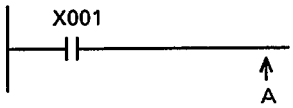


● Function description

This instruction reads the status of Z in the line logic and starts a logical operation.

● Description

When Z is X001, the line logic goes ON when input X001 is ON and goes OFF when X001 is OFF.



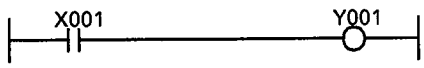
The relation between input X001 and line logic A is shown inthe table below.

Truth table

Z	A
0	0
1	1

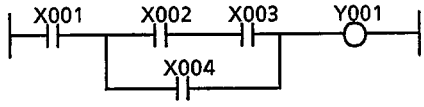
The LD instruction is used in the following four cases.

① At the head of a circuit block



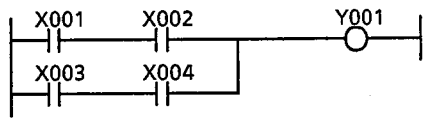

```
LD   X001
OUT  Y001
```

② At the head of circuit blocks connected by ANB



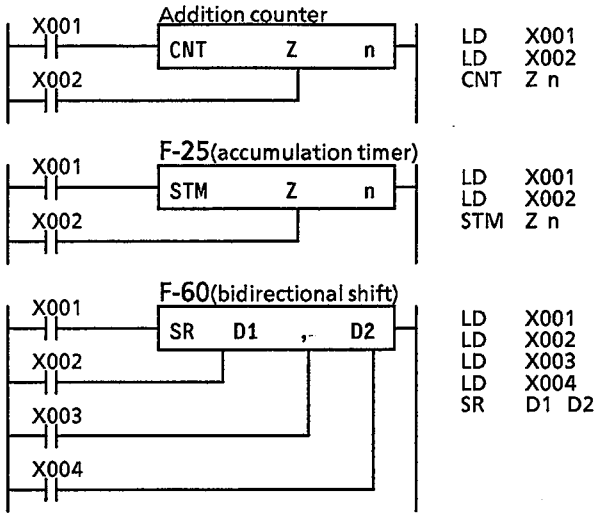

```
LD   X001
LD   X002
AND  X003
OR   X004
ANB
OUT  Y001
```

③ At the head of circuit blocks connected by ORB




```
LD   X001
AND  X002
LD   X003
AND  X004
ORB
OUT  Y001
```

④ In a multi-input instruction

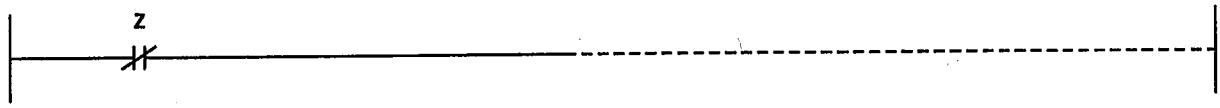


Instruction									
			Execution condition	Always executed	Pulse specification	-	Indirect specification	-	
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-	
LD NOT	FUN	Load not	2 bytes		Overflow M901	-	Operation error M90E	-	

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	X000 to X19F	X200 to X29F	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example

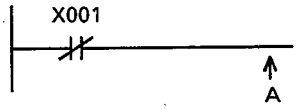


● Function description

This instruction inverts Z status, reads line logic and starts a logical NOT operation.

● Description

When Z is X001, the line logic goes OFF when input X001 is ON and goes on when X001 is OFF.
The line logic status is equivalent to the inversion of the input.



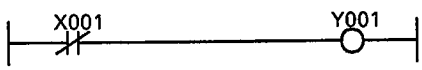
The relation between input X001 and line logic A is shown in the table below.

Truth table

Z	A
0	1
1	0

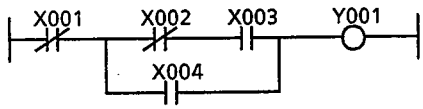
The LD instruction is used in the following three cases.

① At the head of a circuit block



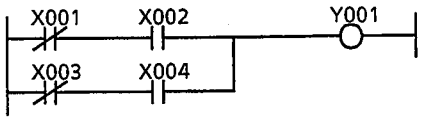

```
LD NOT  X001
OUT     Y001
```

② At the head of circuit blocks connected by ANB




```
LD NOT  X001
LD NOT  X002
AND     X003
OR      X004
ANB
OUT     Y001
```

③ At the head of circuit blocks connected by ORB




```
LD NOT  X001
AND     X002
LD NOT  X003
AND     X004
ORB
OUT     Y001
```

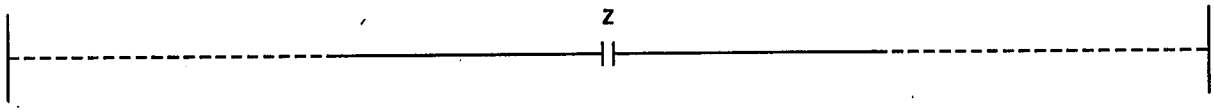
廃 番

Basic instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
AND	FUN	And	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Auxiliary M	Input register	Output register
Z	X000 to X19F	X200 to X29F	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example

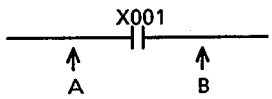


● Function description

This instruction performs an AND operation between line logic and Z.

● Description

When Z is X001, a logic and operation is performed on line logic status prior to X001 and X001.



The relation between line logic A and line logic B, resulting from input X001, is shown in the table below.

Truth table

A	Z	B
0	0	0
0	1	0
1	0	0
1	1	1

◇Note◇

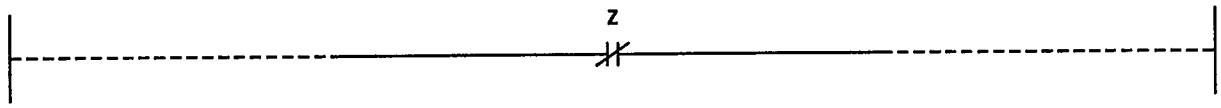
The number of serial contacts that can be created with LD (NOT) and this instruction is limited by program capacity only.
However, a total of 177 can be displayed in ladder notation.

Basic instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
AND NOT	FUN	And not	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	X000 to X19F	X200 to X29F	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example

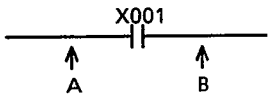


● Function description

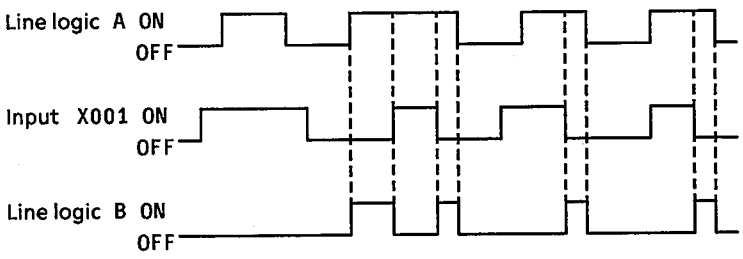
This instruction performs an AND NOT operation between line logic and Z.

● Description

When Z is X001, a logic and not operation is performed on line logic status prior to X001 and X001.



The relation between line logic A and line logic B, resulting from input X001, is shown in the table below.



Truth table

A	Z	B
0	0	0
0	1	0
1	0	1
1	1	0

◇Note◇ The number of serial contacts that can be created with LD (NOT) and this instruction is limited by program capacity only.
However, a total of 177 can be displayed in ladder notation.

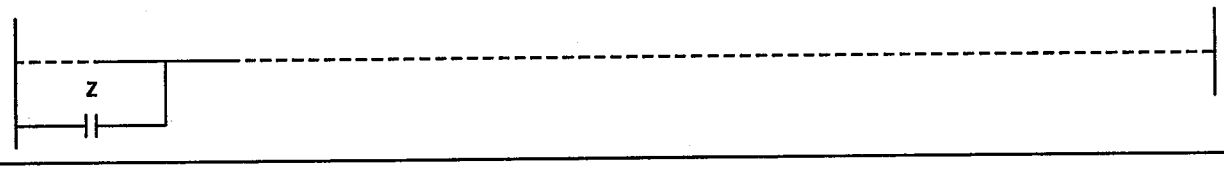
廃 番

Basic instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
OR	FUN	Or	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	X000 to X19F	X200 to X29F	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example

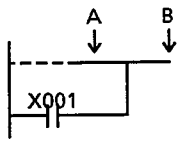


● Function description

This instruction performs an OR operation between Z status and line logic. One contact is used for parallel connection.

● Function description

When Z is X001, line logic B connected to X001 goes on when X001 is on regardless of whether line logic A is on or not.
When X001 is OFF, line logic B and A are identical.

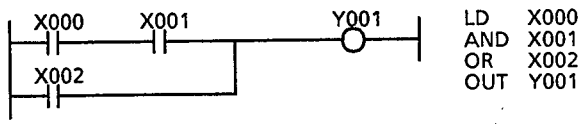


The relation between line logic A and line logic B which depends on input X001 is shown in the table below.

Truth table

A	Z	B
0	0	0
0	1	1
1	0	1
1	1	1

[Example]



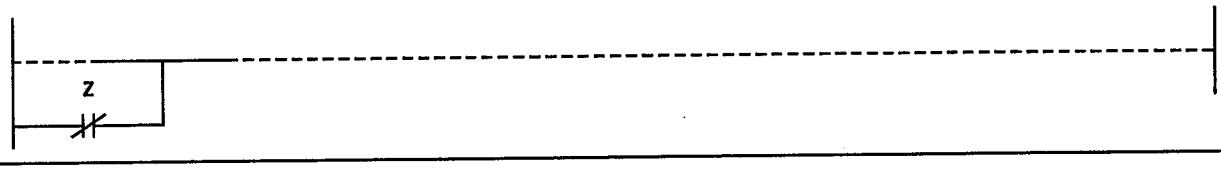
◇Note◇ No more than 22 OR instructions can be used sequentially in ladder notation.

Basic instruction				Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name		Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
OR NOT	FUN	Or not		2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	X000 to X19F	X200 to X29F	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example

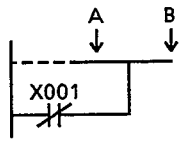


● Function description

This instruction performs an or operation between the inverse value of Z status and the line logic.

● Description

When Z is X001, line logic B connected to X001 becomes identical to line logic A when X001 is on. When X001 is off, line logic B goes on regardless of whether line logic A is on or not.



The relation between line logic A and line logic B which depends on input X001 is shown in the table below.

Truth table

A	Z	B
0	0	1
0	1	0
1	0	1
1	1	1

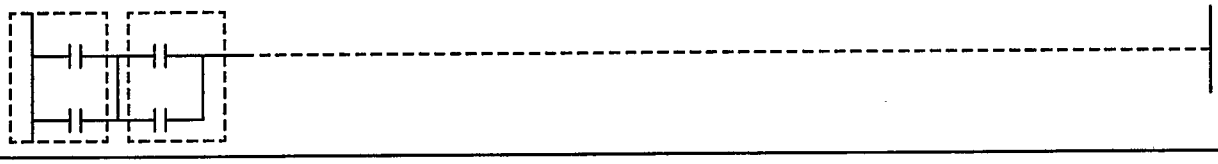
◇Note◇ No more than 22 OR NOT instructions can be used sequentially in ladder notation.

Basic instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ANB	FUN	And block	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

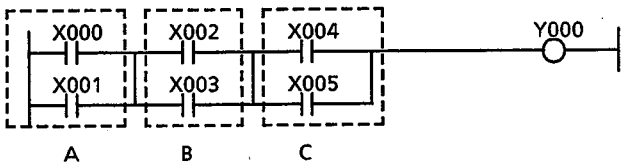
Ladder circuit example



● Function description

This instruction performs an AND BLOCK operation between logic circuit blocks.
There are two methods for configuring a circuit using mnemonics.

【 Example 】



Description example 1:
block sequential description

```
LD   X000 ┐
OR   X001 ┘A
LD   X002 ┐
OR   X003 ┘B
ANB
LD   X004 ┐
OR   X005 ┘C
ANB
OUT  Y000
```

Description example 2:
block contiguous description

```
LD   X000 ┐
OR   X001 ┘A
LD   X002 ┐
OR   X003 ┘B
LD   X004 ┐
OR   X005 ┘C
ANB
ANB
OUT  Y000
```

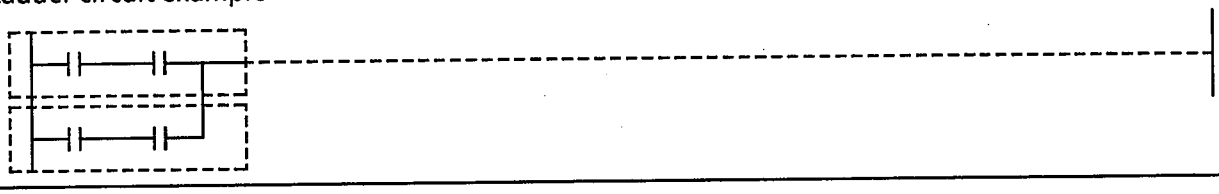
◇Note◇ The only limit to the number of ANB instructions that can be used in example 1 is program capacity. In example 2 no more than 8 ANB instructions can be used.

Basic instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ORB	FUN	Or block	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

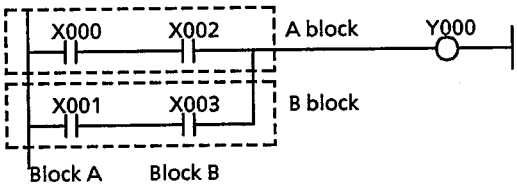
Ladder circuit example



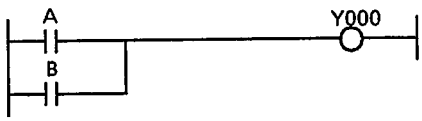
● Function description

This instruction performs an or operation between two logic circuit blocks.

● Description



Logic blocks A and B can be expressed as shown below.



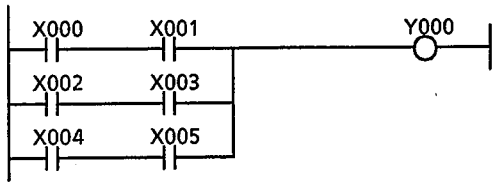
The ORB instruction executes a parallel coupling (OR) of these two blocks. Since ORB couples two or more circuit blocks, the OR or OR NOT instructions are used when B contains only one contact.

[Mnemonic description]

```
LD    X000 ┐
AND   X002 ┘A
LD    X001 ┐
AND   X003 ┘B
ORB
OUT   Y000
```

The two methods (see below) can be used for configuring circuits using mnemonics.

【 Example 】



Description example 1

```
LD    X000
AND   X001
LD    X002
AND   X003
ORB
LD    X004
ANB   X005
ORB
OUT   Y000
```

Description example 2

```
LD    X000
ANB   X001
LD    X002
ANB   X003
LD    X004
ANB   X005
ORB
ORB
OUT   Y000
```

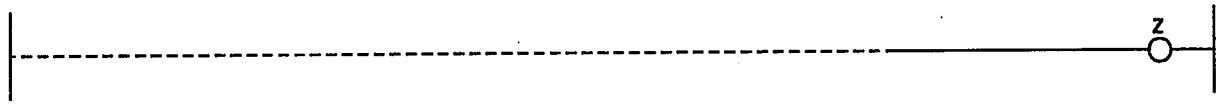
◇Note◇ The only limit to the number of ORB instructions that can be used in example 1 is program capacity. In example 2, no more than 8 ORB instructions can be used.

Basic instruction				Execution condition	Always executed	Pulse specificatio	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-	
OUT	FUN	Out	2 bytes		Overflow M901	-	Operation error M90E	-	

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example

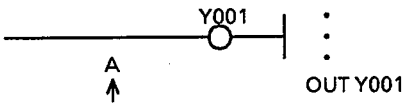


● Function description

This instruction outputs line logic status to relay Z.

● Description

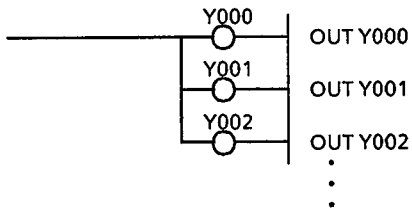
When Z is Y001, Y001 goes on when line logic status goes ON and goes OFF when line logic status goes OFF.



The relation between output Y001 and line logic A is shown below.

Truth table

A	Y000
0	0
1	1



◇Notes◇

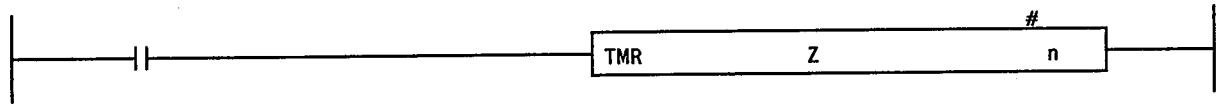
- The number of output stages is not limited by program capacity.
- However, ladder notation limits the use of output stages to 22.An alarm is output if a relay is used more than once in a program (double coil alarm: 40).

Basic instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
TMR	FUN	Timer	4 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-
n	-	-	-	-	#00.01 to #9999	-

Ladder circuit example

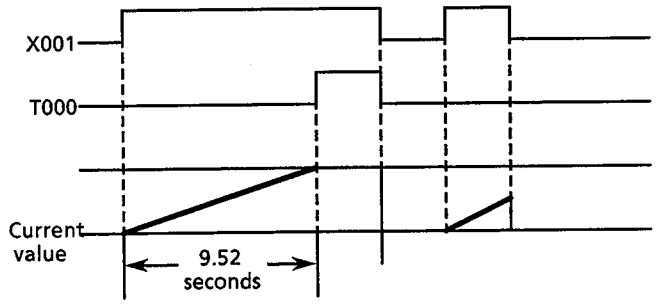


● Function description

This instruction turns ON the timer relay when the timer is activated by the timer input going ON and set value "n" is reached.
When the timer input goes OFF, the timer stops, the current timer value is reset and the timer output also goes OFF.

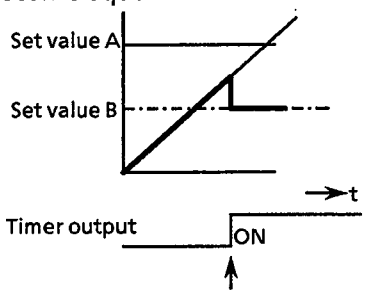
● Description

Three types of units (10 ms, 100 ms and 1 s) can be set depending on the position of the decimal point.
The following operation is performed when the timer input is X001, the timer is T000 and the timer setting is 9.52 s.



◇Notes◇

- The timer and counter are shared devices so the same number cannot be used for both.
- Since the timer is the master clock (10 ms, 100 ms, 1 s) in the MX controllers, the smallest unit of the set value is also a measure of its deviation.
  For example, if the 10 ms timer is used to set 1.00 s, the deviation is also 10 ms. Likewise, if the 1 s timer is used, the deviation is 1 s.
  Consequently, when the setting is three digits or more, a timer with 4 digits should be used for higher accuracy (i.e. a 999.0 s timer should be used for setting 999 s).
- The accuracy of timer output varies -0 to +2 scans due to the timing of the timer input.
- Changes made in RUN mode
  When the set time is less than or equal to current time, the timer is triggered, the output is set to ON and the current value and set value become equal.



Set value A changes to set value B

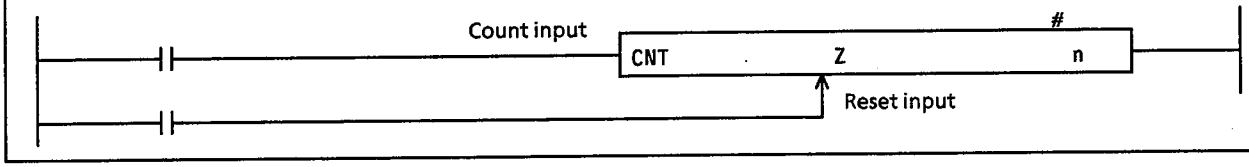
- An error is generated (error code :30) when the same timer number is used more than once.
- If a jump is made past the timer instruction, the timer is not triggered.
- Timer values do not match actual operation after a break.
- The current value is cleared when the input condition is OFF, modes are changed and the power is turned OFF.
- Timer accuracy can be set to two decimal digits.

Basic instruction			Execution condition	Rise *	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
CNT	FUN	Addition counter	4 bytes		Overflow M901	-	Operation error	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-
n	-	-	-	-	#0001 to #9999	-

Ladder circuit example

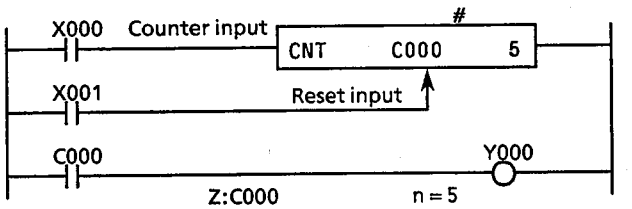


● Function description

The counter starts counting when the count input rises (goes from OFF to ON).When the set value is reached, it is output to the counter relay.
When the reset input is on, the counter is reset to 0 (current value) and the count output is turned OFF. The counter then stays at 0 until the reset input goes OFF.

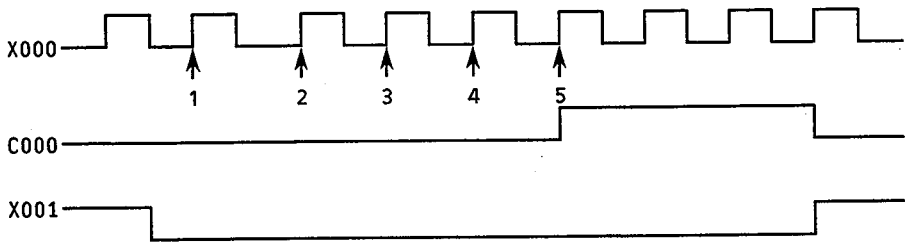
* The counter value is output also at times other than when the count input rises.

● Description




```
LD   X000
LD   X001
CNT  C000
  #  0005
LD   C000
OUT  Y000
```

The timer operates as shown below when the count input is X000, the counter is C000, the set value is 5 times and the reset input is X001.

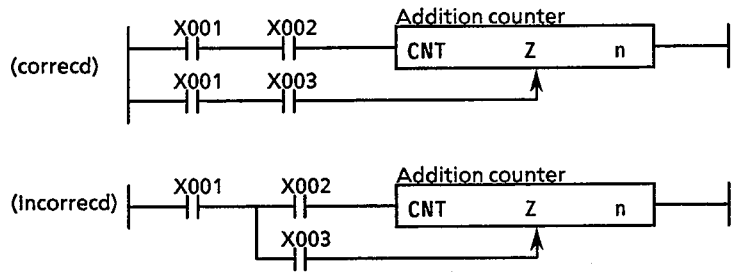


◇Notes◇

- If an output instruction (OUT, MOV, etc.) are input after CNT, a program error (error code :21) is caused.

```
LD    X000
LD    X001
CNT   Z
  #   0005
OUT   Y000  ←Program error
```

- The counter does not operate when the counter input is ON. It goes on when the count input rises.
- The current value of the counter is not cleared when the input condition is OFF, but when the reset input is on.
- The timer and counter are shared devices so the same number cannot be used for both.
- A program error (error code :21) is generated when a reset input is not set.
- The current counter setting is not cleared by mode changes or powering down.
- The reset input is entered using an LD or equivalent instruction (see below).



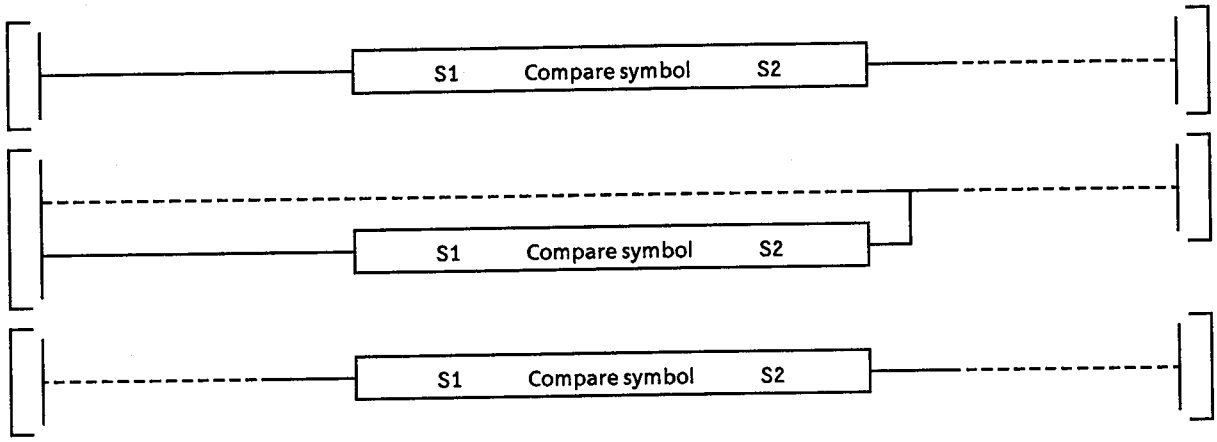
- The counter goes ON when the current value equals a set value. The counter output does not go ON when the set value is changed during counting or the current value is larger than the set value. It goes ON only when the count input rises. The rising edge of the count output causes the set value ON the counter to become the current value and an output is made.
- When there is a change from STOP to RUN mode and the counter input is ON, the counter is incremented by 1.

Instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
(18 types)	FUN	Compare	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	M900 to M970	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	M900 to M970	R0500 to R0519	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 R3800 to P3999	#0000 to #FFFF	K0 to K65535
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	#0000 to #FFFF	K0 to K65535

Ladder circuit example



Compare symbol: =、≠、<、≦、≧、>

● Function description

This instruction compares the unsigned 16-bit data of one word data S1 and S2.When the result of the comparison is true, it equals the value of contact "a" when on.

● Description

There are six possible results of a comparison.

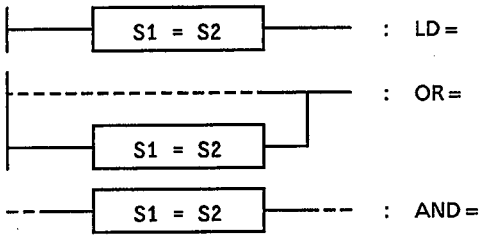
	Conducting status ON
=、S1 = S2	True
≠、S1≠S2	True
<、S1<S2	True
≦、S1≦S2	True
≧、S1≧S2	True
>、S1>S2	True

This instruction is handled in the same way as a normal input relay. The combination of 6 compare symbols result in 18 types of instructions

LD=	AND=	OR=
LD≠	AND≠	OR≠
LD<	AND<	OR<
LD≦	AND≦	OR≦
LD≧	AND≧	OR≧
LD>	AND>	OR>

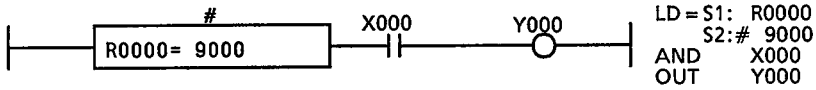
18 types

For example, there are three types of = instruction.



【 Example 】

- When register R0000 matches the value #9000, an AND operation is performed on this register and contact X000. The result of the operation is passed to Y000.



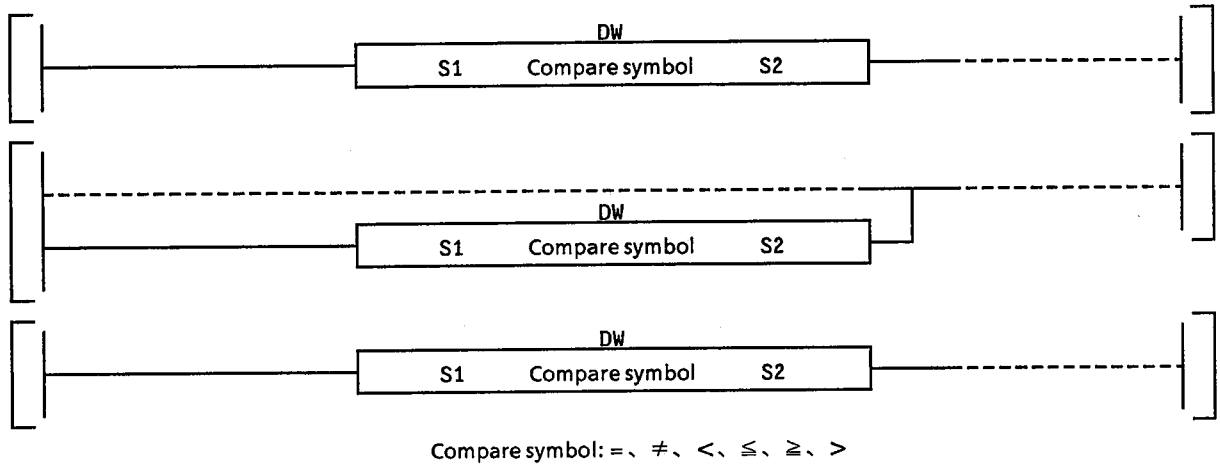
Instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
(18 types)	FUN	Double-word compare	11 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	M900 to M960	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	M900 to M960	R0500 to R0518	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	R0900 to R0998 R3800 to P3998	#00000000 to #FFFFFFFF	K0 to K4294967295
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	R0900 to R0998 P3800 to P3998	#00000000 to #FFFFFFFF	K0 to K4294967295

Ladder circuit example



Compare symbol: =、≠、<、≦、≧、>

● Function description

This instruction compares the 32-bit unsigned data of two word data S1 + 1 (high-order), S1 (low-order) and two word data S2 + 1 (high-order) and S2 (low-order). When the result of the comparison is true, it equals the value of contact "a" when on.

● Description

There are six possible results of a comparison.

		Conducting status ON
=、S1+1, S1	= S2+1, S2	True
≠、S1+1, S1	≠ S2+1, S2	True
<、S1+1, S1	< S2+1, S2	True
≦、S1+1, S1	≦ S2+1, S2	True
≧、S1+1, S1	≧ S2+1, S2	True
>、S1+1, S1	> S2+1, S2	True

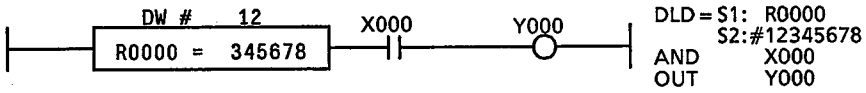
This instruction is used in combination with the LD, AND and ORinstructions. The combination of 6 compare symbols result in 18 types of instructions

DLD=	DAND=	DOR=
DLD≠	DAND≠	DOR≠
DLD<	DAND<	DOR<
DLD≦	DAND≦	DOR≦
DLD≧	DAND≧	DOR≧
DLD>	DAND>	DOR>

18 types

【 Example 】

When register R0001 (high-order) and R0000 (low-order) match the value #12345678, an AND operation is performed on this register and contact X000. The result of the operation is passed to Y000.

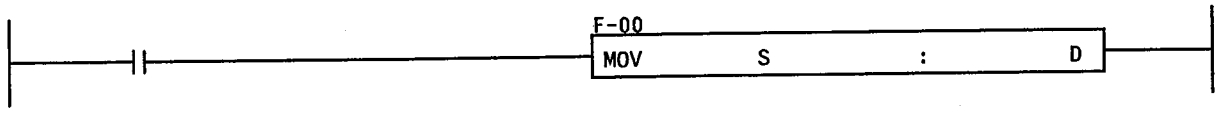


Transfer instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
MOV	FUN 00	Data transfer	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	M900 to M970	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	M900 to M970	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	-	-

Ladder circuit example

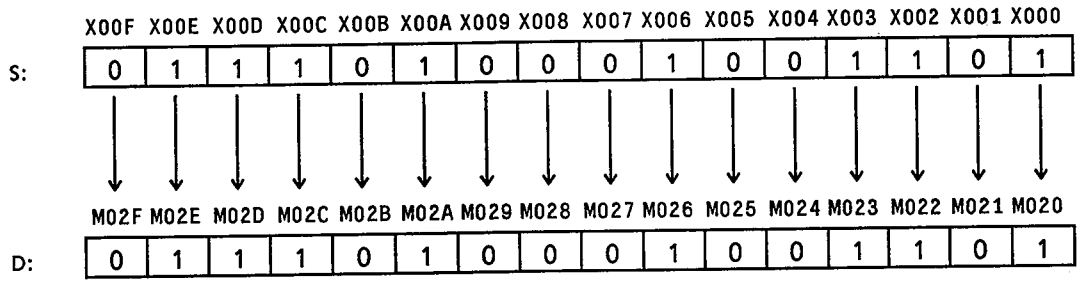


● Function description

This instruction transfers one word data S to destination D.
The value of S is not changed in the operation.

● Description

A word S starting from X000 is transferred to D from M020 as shown below.



◇Note◇ A timer or a counter cannot be used at the destination of the transfer. Use the set value write instruction (FUN90) and set value read instruction (FUN91) of the timer and the counter.

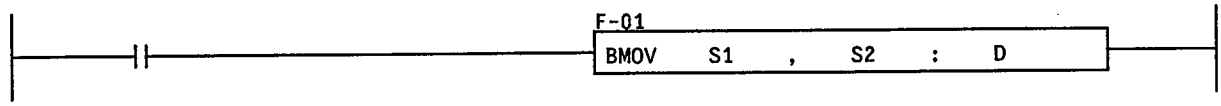
Transfer instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BMOV	FUN 01	Block transfer	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

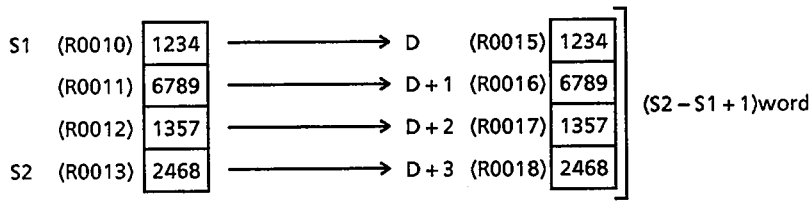


● Function description

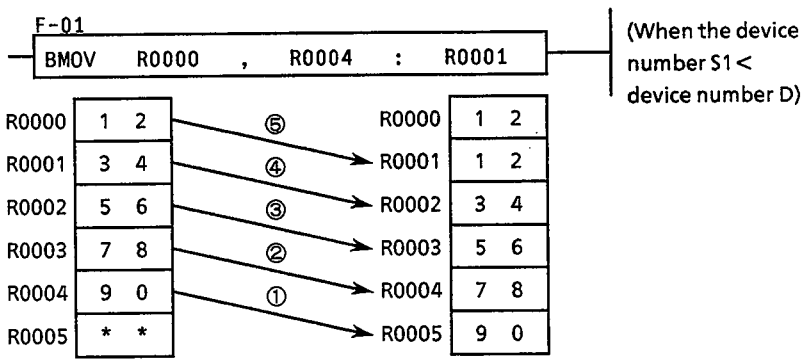
This instruction moves S1 and S2 word units to destination D.

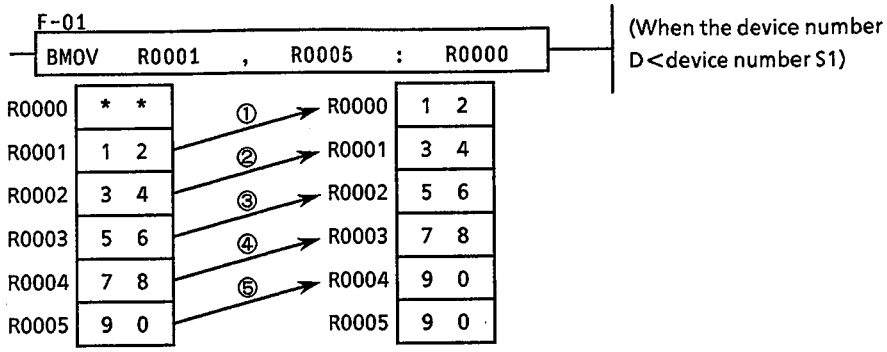
● Description

When S1 is in R0010, S2 in R013 and D in R015, the following transfer is performed.



When the device specified as S is the same as that specified as D, the following process is performed.





Since the process is performed in the order 1 - 5 shown above, there is no risk that the data is corrupted.
When D is a contact area, moves are performed in word units. For example, when D is M000 and a 2-word move is transferred, the data is moved to M000 to M00F and M010 to M01F.

◇Notes◇

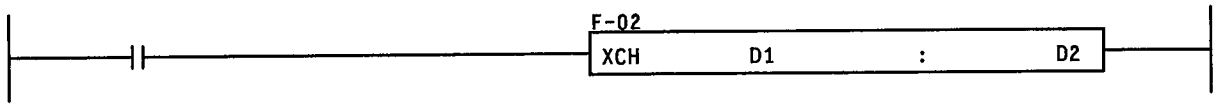
- S1>S2 a parameter error is generated.
- S1 and S2 must belong to the same area.
- A D setting that exceeds the D area causes a parameter error.
- Use of indirect specification
  When S1>S2 or a D setting exceeds the D area, an operation error (M90E, M90F) is generated.

Transfer instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
XCH	FUN 02	Data exchange	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

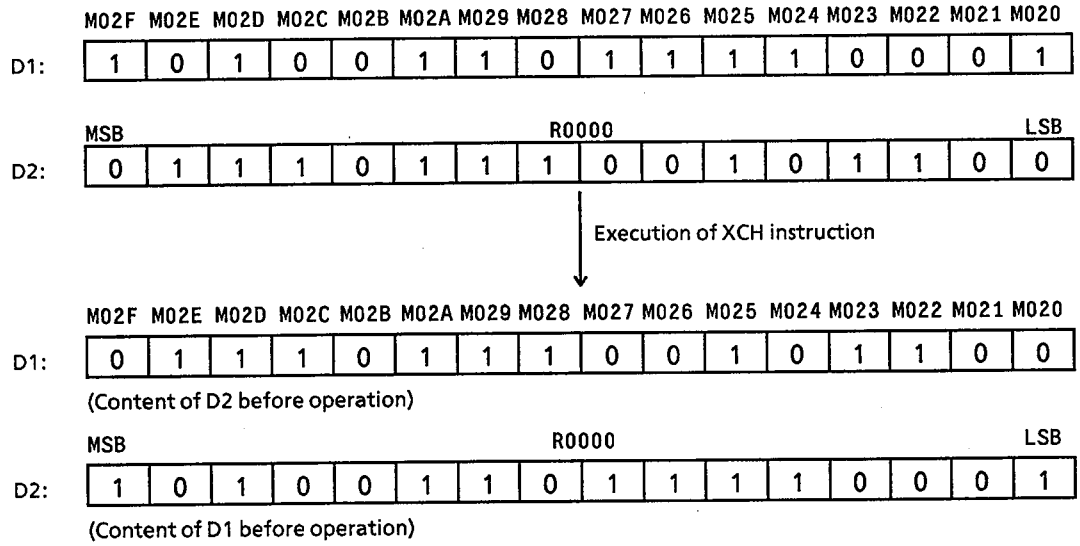


● Function description

This instruction exchanges the content of word data D1 and D2.

● Description

When D1 starts from M020 and D2 from R0000, the following operation is performed.



廃番

Transfer instruction

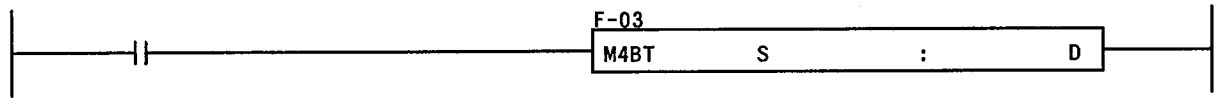
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
M4BT	FUN 03	4-bit transfer	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X19C	X200 to X29C	Y000 to Y19C	Y200 to Y29C	M000 to M89C	L000 to L49C	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y19C	Y200 to Y29C	M000 to M89C	L000 to L49C	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

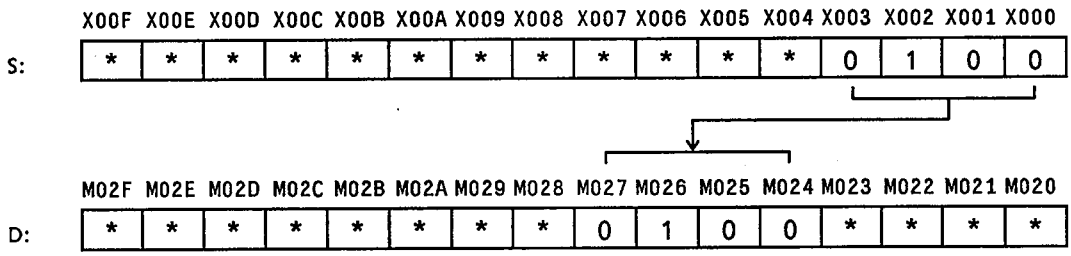


● Function description

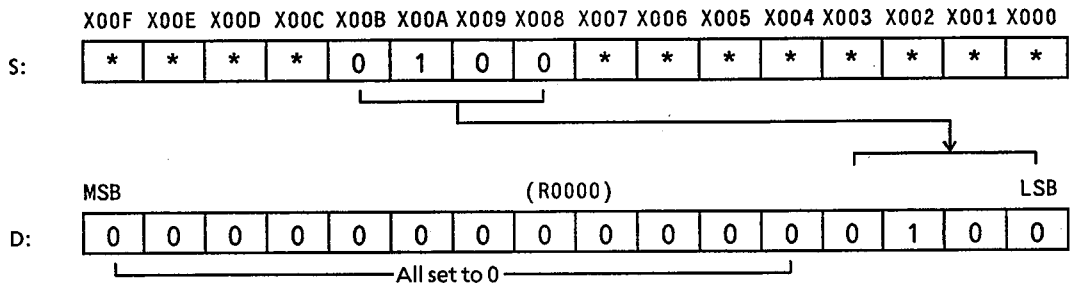
This instruction transfers 4-bit data from S to destination D. When S or D is a relay area (X, Y, M or L), the transfer data can be specified with its least significant digit set to 0, 4, 8 or C. When S or D is a register, the low-order 4 bits are transferred and the high-order 12 bits are set to 0.

● Description

i) When S starts from X000 and D from M024, the following operation is performed.



ii) When S starts from X008 and D from R0001, the following operation is performed.



Transfer instruction

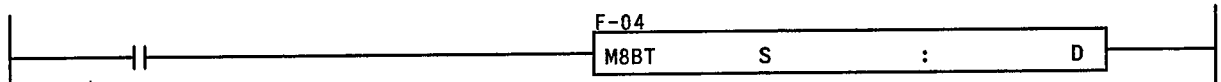
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags				
M8BT	FUN 04	8-bit transfer	7 bytes		Shift carry M900	-	Underflow M902	-
					Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X198	X200 to X298	Y000 to Y198	Y200 to Y298	M000 to M898	L000 to L498	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y198	Y200 to Y298	M000 to M898	L000 to L498	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

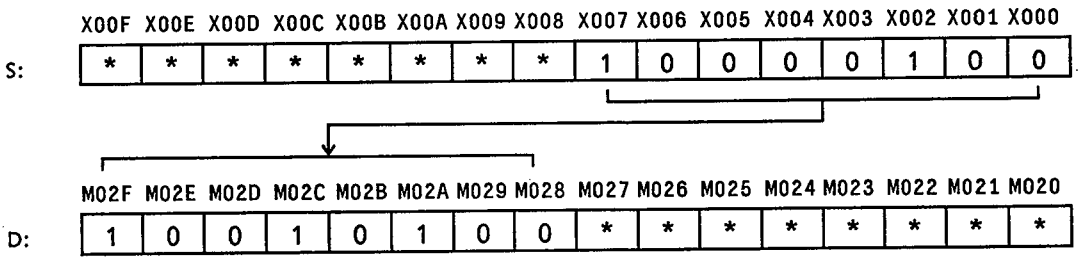


● Function description

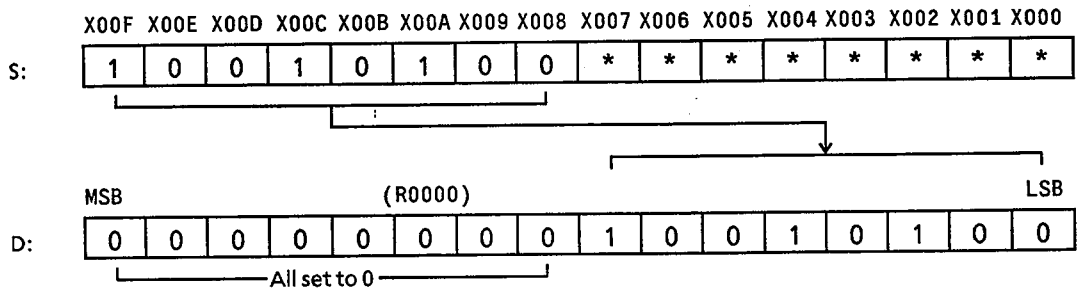
This instruction transfers 8-bit data from S to destination D. When S or D is a relay area (X, Y, M or L), the transfer data can be specified with its least significant digit set to 0 or 8. When S or D is a register (R), the low-order 8 bits are transferred and the high-order 8 bits are set to 0.

● Description

i) When S starts from X000 and D from M028, the following operation is performed.



ii) When S starts from X008 and D from R0001, the following operation is performed.

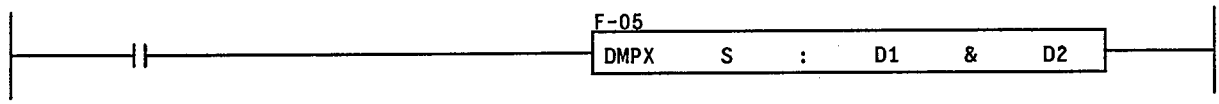


Transfer instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DMPX	FUN 05	Data distribution	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D1	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

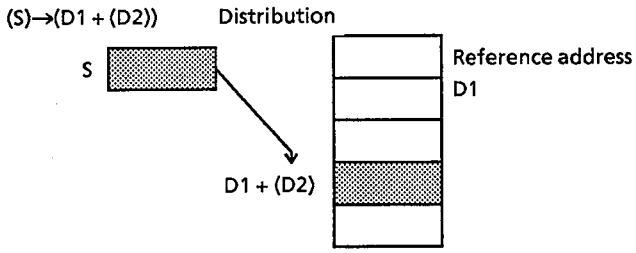
	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



● Function description

This instruction stores word data of distribution data S in an address which is the result of adding the content (BCD value) of offset data D2 to reference address D1.



● Description

When D1 starts from M000, the reference address, and the offset data D2 starting from R0000 is 008, the content of S starting from X000 is transferred to M080.

S :X000
D1 :M000
D2 :R0000

Distribution

X000 1001001001001010 → M080 1001001001001010

(R0000):08 BCD value

◇Notes◇

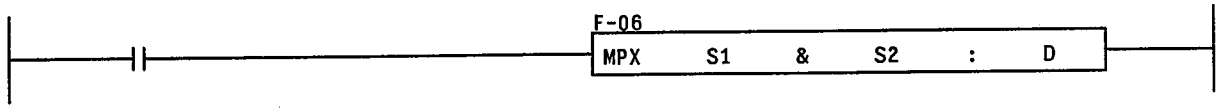
- When D2 does not contain a BCD value, an operation error (M90E or M90F) is generated and the operation is aborted.
- The D1 + (D2) address must be in the same area as D1. If it exceeds this area, an operation error (M90E or M90F) is generated and the operation is aborted. (For example, if D1 is an output address, D1 + (D2) is also an output address.)
- When D1 is not in a register area, the address/word is 10 so the value added to D1 is offset (D2) x 10 (address/word).

Transfer instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
MPX	FUN 06	Data extraction	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X200 to X290	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

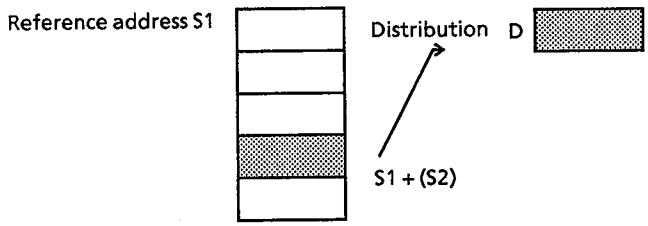
	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



● Function description

This instruction stores the address data which is obtained by adding the content (BCD value) of offset data S2 to reference address S1 at extraction address D.



● Description

When S1 starts from M000, the reference address, and offset data S2 starting from M010 is #008, the content of S starting from X000 is transferred to D starting from R0060.

S1 :M000
S2 :M010
D :R0060
(S2) :08

M080 1010101010101010 → R0060 1010101010101010

◇Notes◇

- When the content of S2 is not a BCD value, an operation error (M90E or M90F) is generated and the operation is aborted.
- The S1 + (S2) address must be in the same area as S1. If it exceeds this area, an operation error (M90E or M90F) is generated and the operation is aborted. (For example, if S1 is an output address, S1 + (S2) is also an output address.)
- When S1 is not in a register area, the address/word is 10 so the value added to S1 is offset (D2) x 10 (address/word).

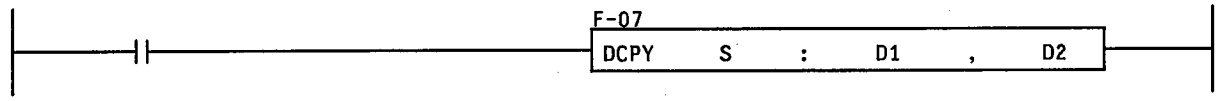
廃 番

Transfer instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DCPY	FUN 07	Data batch copy	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D1	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

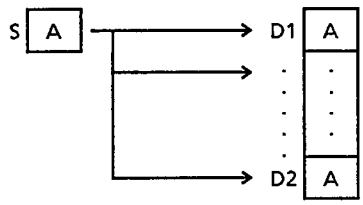
	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



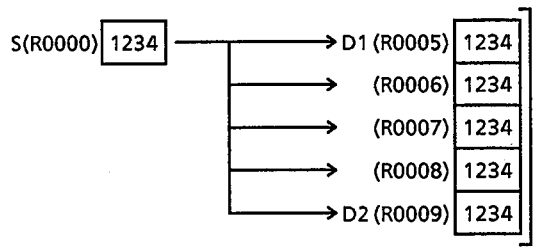
● Function description

This instruction transfers constants or word data at address S, which stores the data to be transferred, (from the starting address at D1 to D2) to the starting address D1 to D2 at the destination.



● Description

This indicates that data #1234 at S:R0000 is transferred to D1:R0005 to D2:R0009.



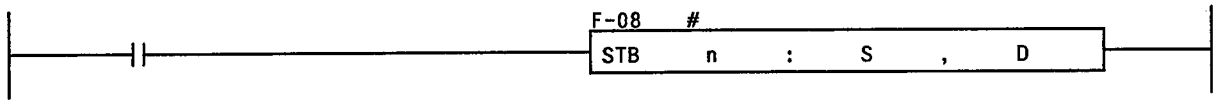
◇Note◇ D1＜D2

Transfer instruction			Execution condition	Rise	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Instruction name	Flags	Shift carry M900	-	Underflow M902	-
STB	FUN 08	Strobed 8-bit transfer	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y090	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#0001 to #0099	-
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	-	-	-	-	-

Ladder circuit example



● Function description

This instruction outputs data to D as strobed data starting from S to the "n-th" byte (8-bit units). One data unit is output every 3 scans in the following order: low-order S bytes, high-order S bytes, low-order S+1 bytes, etc. Data is output to Y0*0 to Y0*7 while strobed data is output to Y0*8 at destination D. Data in Y0*9 to Y0*F is not affected.

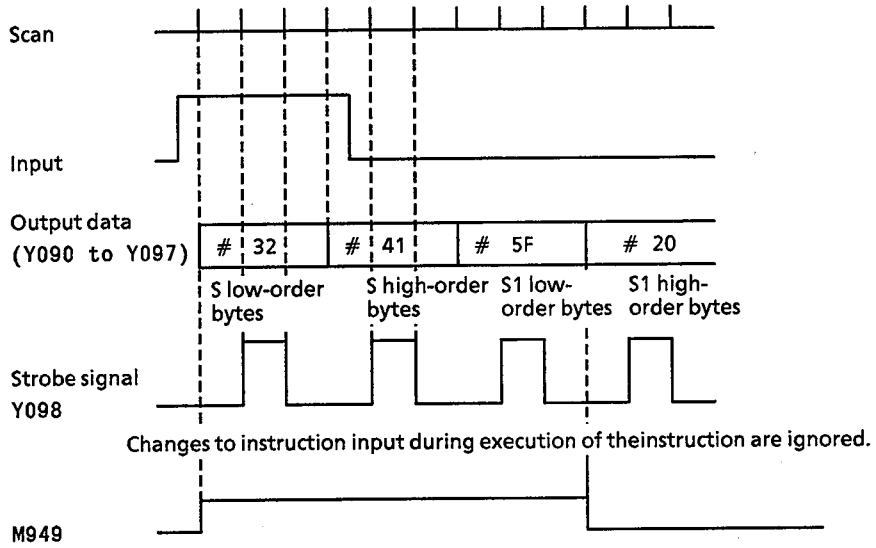
Special relay (Execution of STB instruction in progress) M940 to M949 corresponding to output Y000 to Y098 goes on when this instruction is executing.When the input rises, all data up to the "n-th" item is executed and changes made during execution are ignored. "n" is specified as BCD data.

● Description

When "n" is 3, S is in R0000 and D in Y090, the following process is performed.

	F 87	0
S: R0000	41H	32H
R0001	20H	5FH

The data is output in the following order: S low-order bytes (32H), S high-order bytes (41H), S + 1 low-order bytes (5FH), etc.



◇Note◇

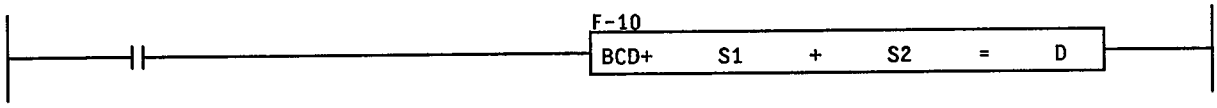
An STB instruction is processed as shown below.
1st scan Y0*8←OFF
Y0*0 to Y0*7←DATA
2nd scan Y0*8←ON
3rd scan Y0*8←OFF
Thus if the content of Y0*0 to Y0*7 is changed using the MOV instruction after the 1st scan, the data cannot be recovered.

BCD arithmetic instruction				Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-	
BCD +	FUN 10	BCD addition	9 bytes		Overflow M901	○	Operation error M90E	-	

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

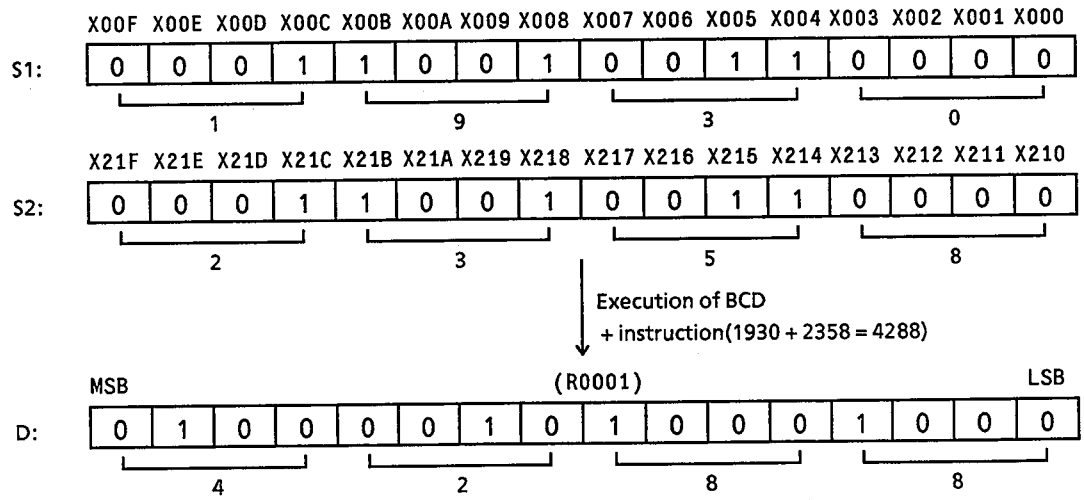


● Function description

This instruction performs a BCD addition of one word 4-digit BCD data S1 and S2 and passes the result to D.
The content of S1 and S2 are not changed by the operation.
S1 contains the summand and S2 the addend.

● Description

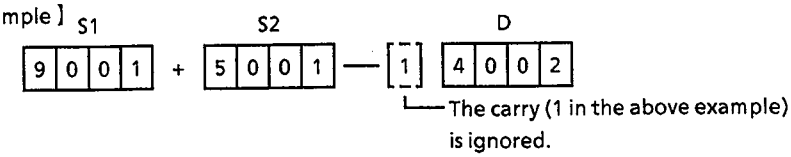
When S1 starts from X000, S2 from X210 and D from R0001, the following operation is performed.



◇Notes◇

- When S1 and S2 digits are not BCD values (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.
- When the result of adding S1 and S2 is 5 digits or larger (exceeding 9999), an overflow flag (M901) is set.

[Example]

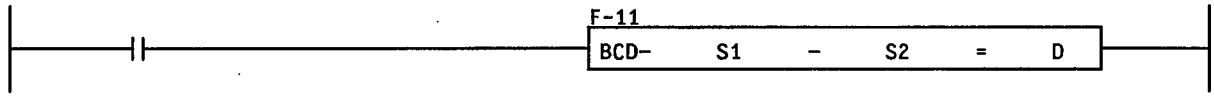


BCD arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	○
BCD –	FUN 11	BCD subtraction	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

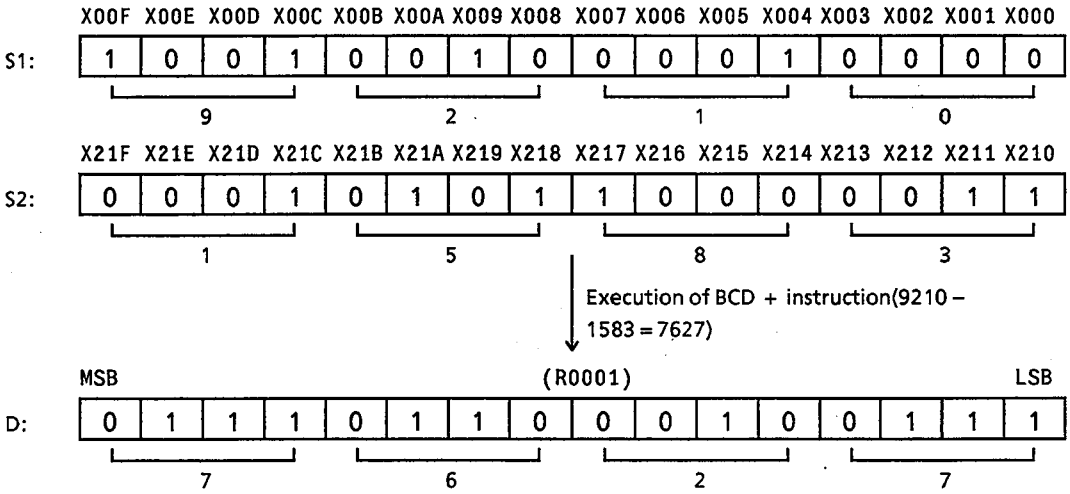


● Function description

This instruction subtracts (BCD subtraction) S2 from S1 (both one word 4-digit BCD data) and passes the result to D.
The content of S1 and S2 are not changed by the operation.
S1 contains the minuend and S2 the subtrahend.

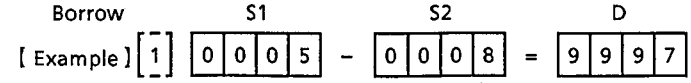
● Description

When S1 starts from X000, S2 from X210 and D from R0001, the data is processed as shown below.



◇Notes◇

- When S1 and S2 digits are not BCD values (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.
- When the result of subtracting S2 from S1 is a negative value, an underflow flag (M902) is set. Since the result of D is borrowed, S1 is assumed to be 10000 and S2 is subtracted from it.

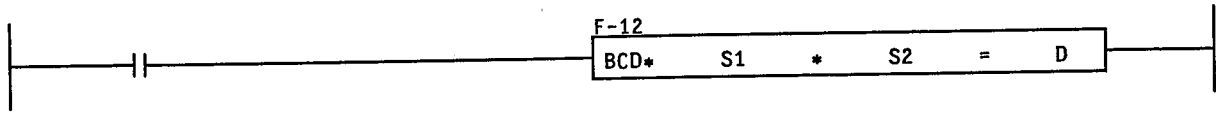


BCD arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BCD*	FUN 12	BCD multiplication	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

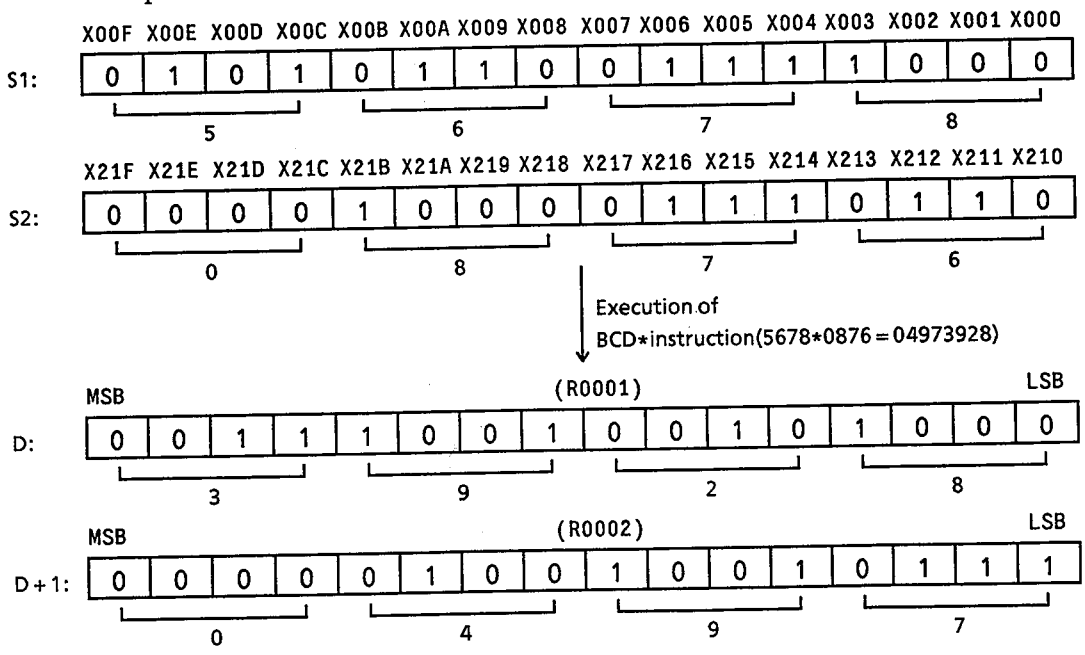


● Function description

This instruction performs a BCD multiplication of S1 and S2 one word 4-digit BCD data and passes the high-order 4 digits of the result to D+1 and the low-order 4 digits to D.
The content of S1 and S2 is not changed by the operation. S1 contains the multiplicand and S2 the multiplier data.

● Description

When S1 starts from X000, S2 from X210 and D from R0001, the data is processed as shown below.



◇Note◇ When S1 and S2 digits are not BCD values (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.

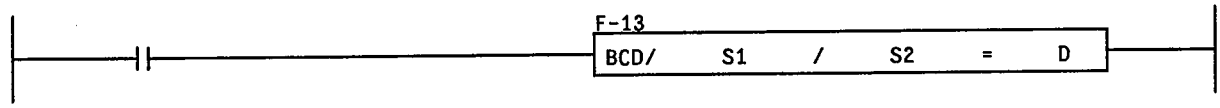
廃 番

BCD arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BCD/	FUN 13	BCD division	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

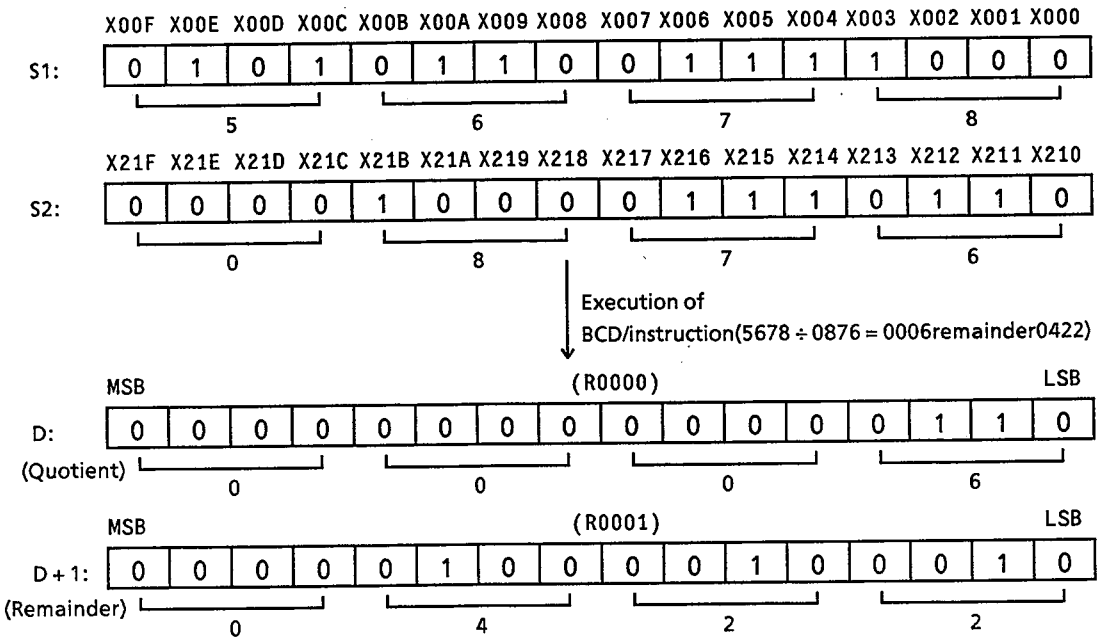


● Function description

This instruction divides the 4-digit BCD data (1 word is 16 bits) in S1 by data in S2 and passes the resulting quotient to D and the remainder to D+1. S1 contains the dividend and S2 contains the divisor.

● Description

When S1 starts from X000, S2 from X210 and D from R0000, the following operation is performed.



◇Notes◇

- When S1 and S2 digits are not BCD values (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.
- When the divisor S2 is 0, an operation error is generated.

BCD arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BCDI	FUN 14	BCD increment	5 bytes		Overflow M901	○	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

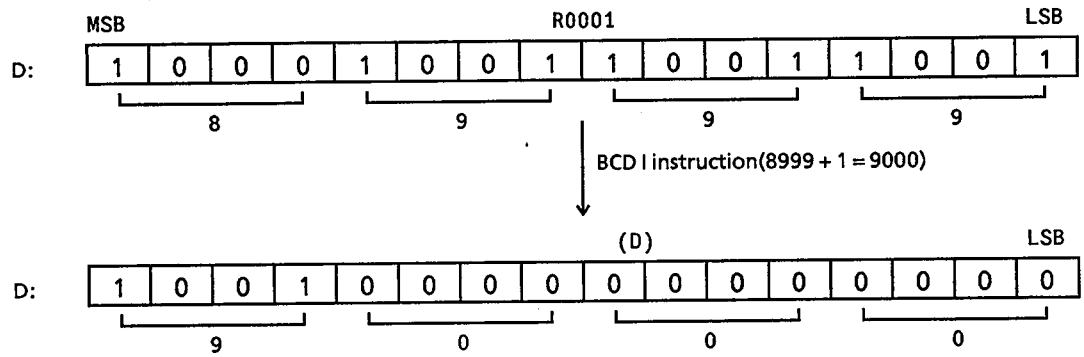
Ladder circuit example

● Function description

This instruction increments 4-digit BCD word data D by one(+1).

● Description

When D is in R0001, the following operation is performed.



◇Notes◇

- When a line in D is not a BCD value (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.
- When the content of D is 9999 and the result of the operation generates 10000, the overflow flag (M901) is set and the content of D is 0000.

BCD arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	○
BCDD	FUN 15	BCD decrement	5 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

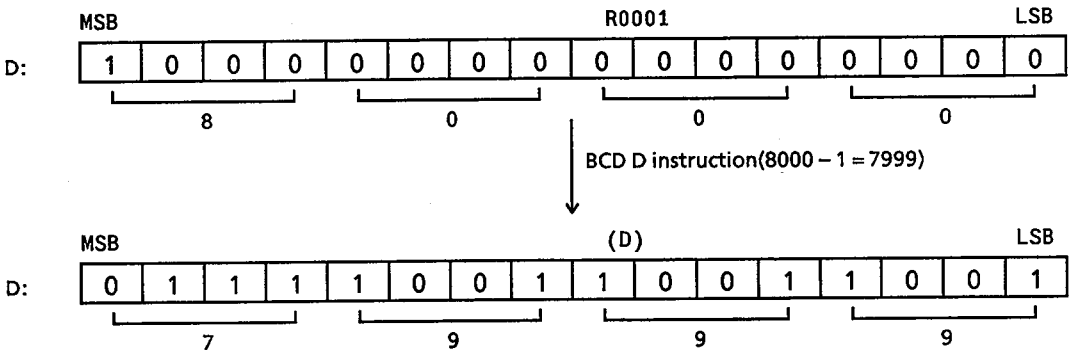
Ladder circuit example

● Function description

This instruction decrements 4-digit BCD word data D by one(-1).

● Description

When D is in R0001, the following operation is performed.



◇Notes◇

- When a line in D is not a BCD value (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.
- When the content of D is 0000 and the result of the operation generates -0001, the underflow flag (M902) is set and the content of D is 9999.

BBCD arithmetic instruction

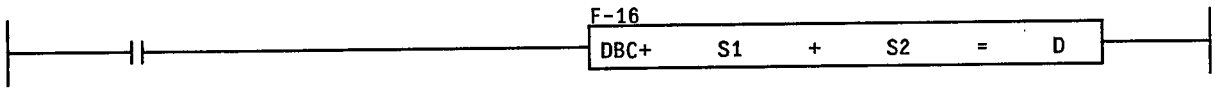
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DBC+	FUN 16	Double-word BCD addition	13 bytes		Overflow M901	○	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #99999999	-
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #99999999	-
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

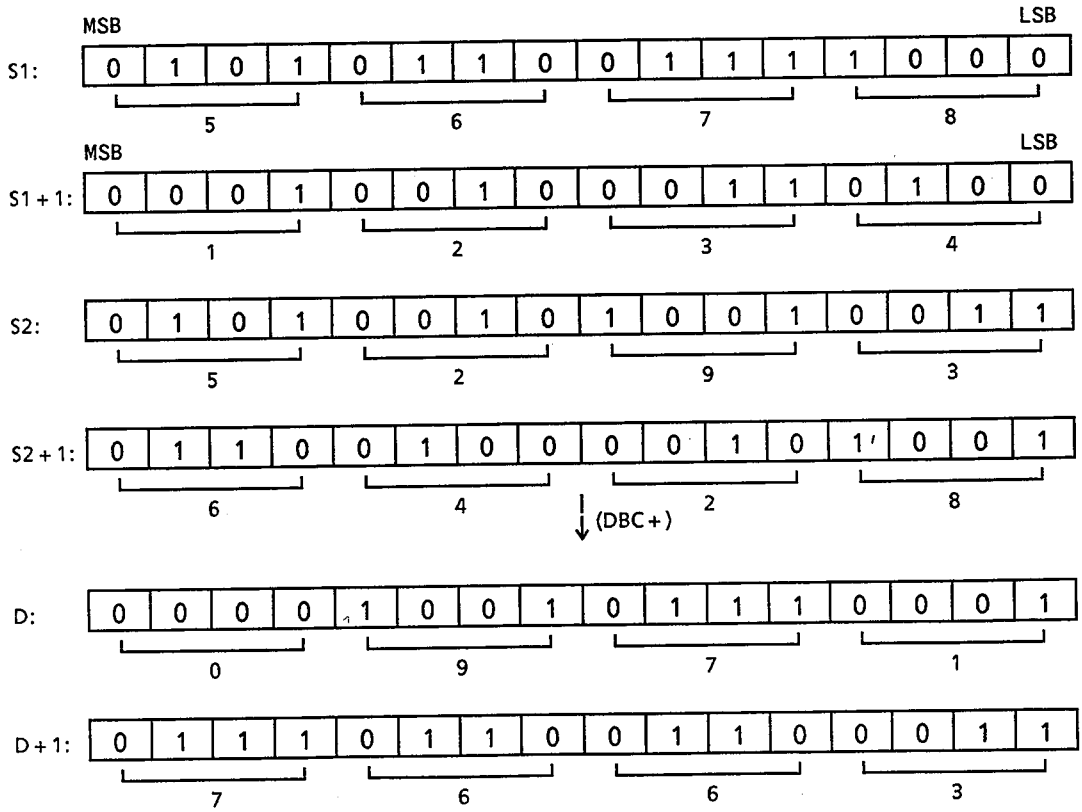


● Function description

This instruction adds 8-digit BCD data in two words consisting of S1 + 1 (high-order) and S1 (low-order) to two words consisting of S2 + 1 (high-order) and S2 (low-order). The resulting 32-bit data string is passed to D+1 (high-order) and D (low-order). The content of S1 + 1, S1, S2 + 1 and S2 is not affected by the operation.S1 + 1 and S1 contain the summand and S2 + 1 and S2 contain the addend.

● Description

When S1 +1 and S1 are 12345678 and S2 + 1 and S2 are 64285293, the result of the operation which is placed in D becomes as follows.



◇Notes◇

- When S1 + 1, S1, S2 + 1 and S2 digits are not BCD values (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.
- When the result of adding S1 + 1, S1, S2 + 1 and S2 is 9 digits, an overflow flag (M901) is set.

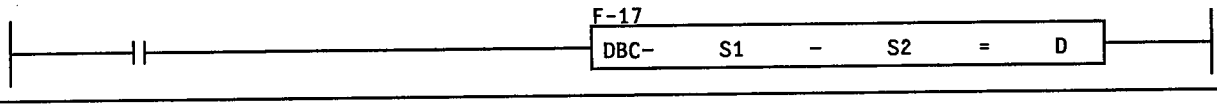
[Example] 99999999 + 10000000 = 109999999, D + 1; D: 09999999
The result is M901: ON.

BCD arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	○
DBC-	FUN 17	Double-word BCD subtraction	13 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #99999999	-
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #99999999	-
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

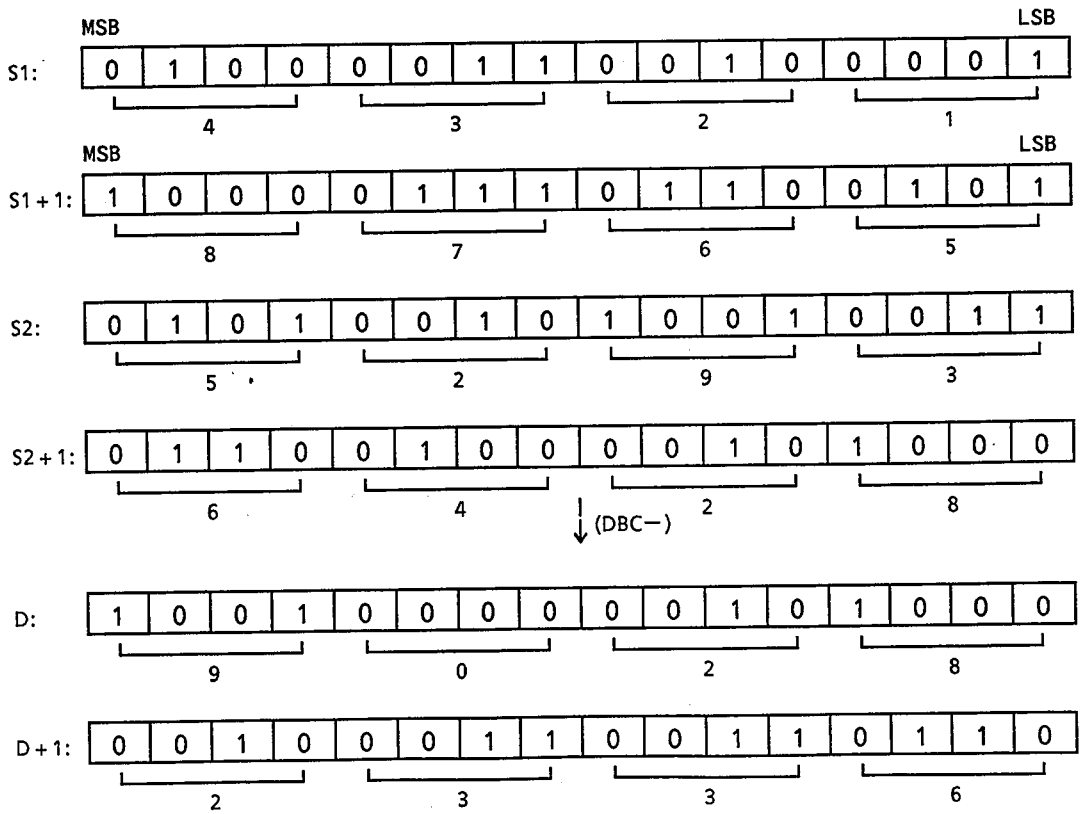


● Function description

This instruction subtracts two word 8-digit BCD data S2 + 1 (high-order) and S2 (low-order) from S1 + 1 (high-order) and S1 (low-order). The result (32-bit data) is passed to D +1 (high-order) and D (low-order). The content of S1 + 1, S1, S2 and S2 + 1 is not affected by the operation.S1 + 1 and S1 contain the minuend and S2 + 1 and S2 the subtrahend.

● Description

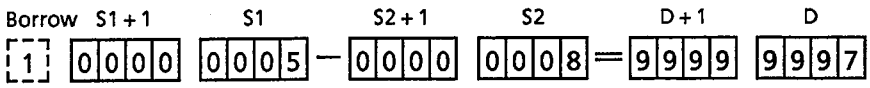
When S1 +1 and S1 are 87654321 and S2 + 1 and S2 are 64285293, the result of the operation which is placed in D becomes as follows.



◇Notes◇

- When S1 + 1, S1, S2 + 1 and S2 digits are not BCD values (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.
- When the result of subtracting S2 + 1, S2 from S1 + 1 and S1 is a negative value, an underflow flag (M902) is set. Since the result in D is the remainder of 100000000 (S1 + 1 and S1) minus S2 + 1 and S2.

【 Example 】



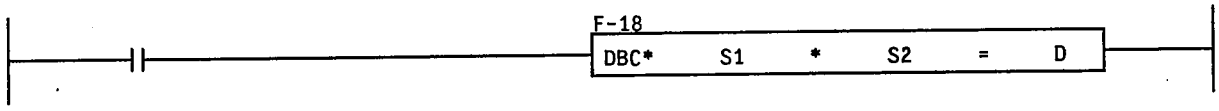
廃番

BCD arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DBC*	FUN 18	Double-word BCD multiplication	13 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y160	Y200 to Y260	M000 to M860	L000 to L460	-	-	R0600 to R0616

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #99999999	-
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #99999999	-
D	-	R0000 to R0496 R1000 to R4996	P0000 to P3796	-	-	-

Ladder circuit example

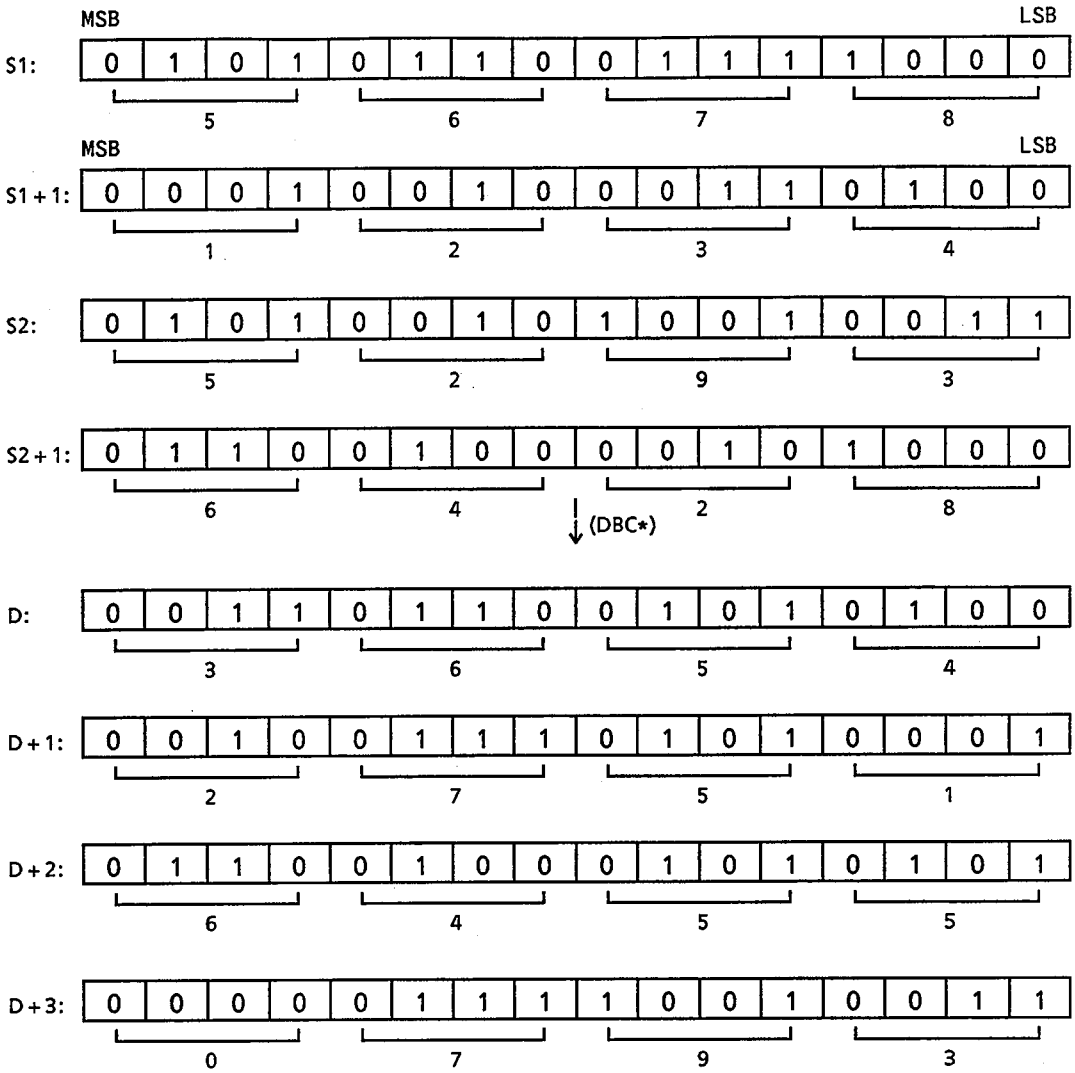


● Function description

This instruction multiplies 8-digit BCD data in two words consisting of S1 + 1 (high-order) and S1 (low-order) with two words consisting of S2 + 1 (high-order) and S2 (low-order) . The 8 digit result is passed to D + 3 (high-order), D + 2, D + 1 and D (low-order). The content of S1 + 1, S1, S2 and S2 + 1 is not affected by the operation.S1 + 1 and S1 contain the multiplicand and S2 + 1 and S2 the multiplier.

● Description

When S1 +1 and S1 are 12345678 and S2 + 1 and S2 are 64285293, the result of the operation which is placed in D becomes as follows.



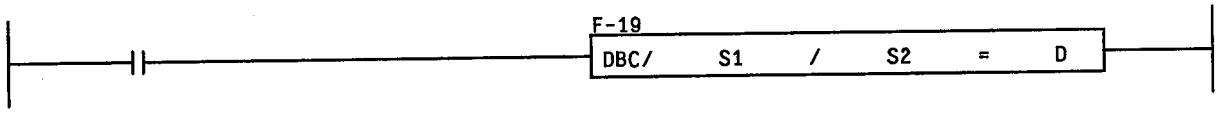
◇Note◇ When S1 + 1, S1, S2 + 1 and S2 digits are not BCD values (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.

BCD arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DBC/	FUN 19	Double-word BCD division	13 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y160	Y200 to Y260	M000 to M860	L000 to L460	-	-	R0600 to R0616

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #99999999	-
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #99999999	-
D	-	R0000 to R0496 R1000 to R4996	P0000 to P3796	-	-	-

Ladder circuit example

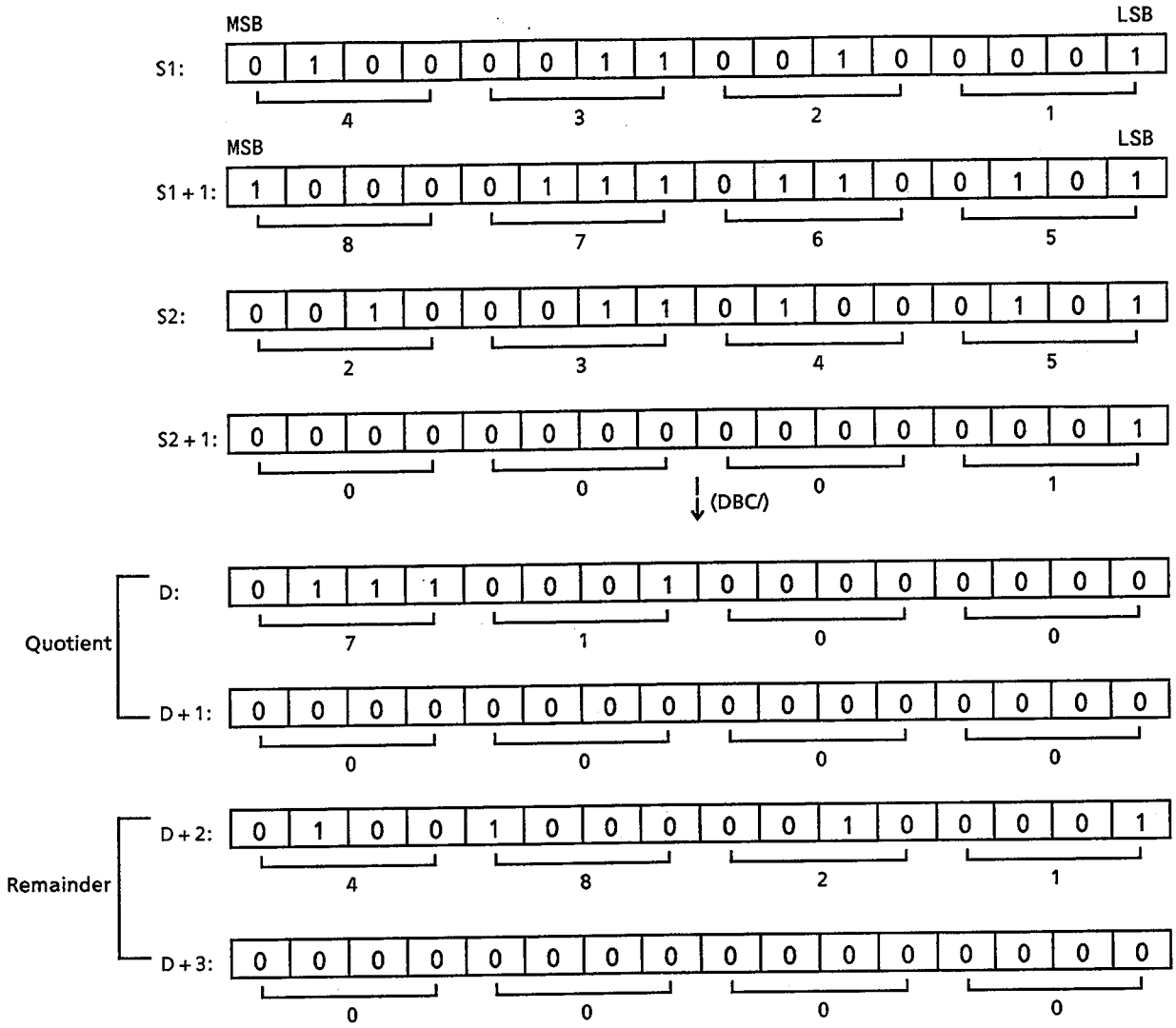


● Function description

This instruction divides the two word 8-digit BCD data in S1 + 1 (high-order) and S1 (low-order) by data in S2 + 1 (high-order) and S2 (low-order). The resulting quotient is passed to D + 1 (high-order) and D (low-order) while the remainder is passed to D+3 (high-order) and D + 2 (low-order).S1 + 1 and S1 contain the dividend and S2 + 1 and S2 contain the divisor.

● Description

When S1 + 1 and S1 is 87654321 and S2 + 1 and S2 are 00012345, the following operation is performed.



◇Notes◇

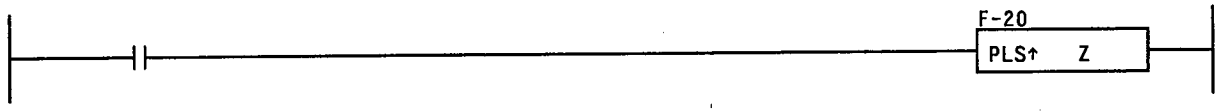
- When S1 + 1, S1, S2 + 1 and S2 digits are not BCD values (0 to 9), an operation error (M90E and M90F) is generated and the operation is aborted.
- When the divisor S2 is 0, an operation error is generated.

Semi-basic instruction			Execution condition	Rise *	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
PLS ↑	FUN 20	Rise pulse	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	-	-	-	-	-	-

Ladder circuit example

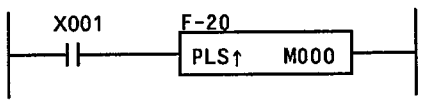


● Function description

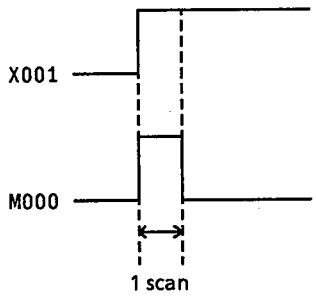
This instruction turns on output Z for one scan when the input rises (goes from OFF to ON).

* Z is off when the input is not rising.

● Description



When the input is X001 and output Z is M000, the following operation is performed.



◇Note◇ When the input is on and the system changes from the STOP mode to the RUN mode, output Z is turned on for the duration of one scan.

Semi-basic instruction			Execution condition	Fall *	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
PLF ↓	FUN 21	Falling pulse	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	-	-	-	-	-	-

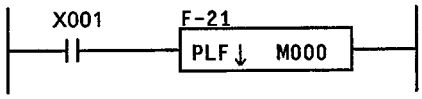
Ladder circuit example

● Function description

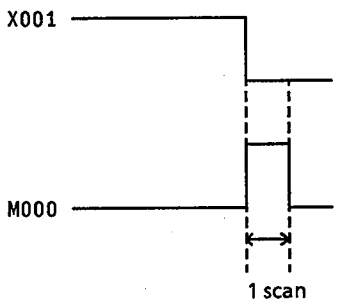
This instruction turns on output Z for one scan when the input falls (goes from ON to OFF).

* Z is off when the input is not falling.

● Description



When the input is X001 and output Z is M000, the following operation is performed.



◇Note◇ When the system goes from the STOP mode to the RUN mode, output Z stays OFF if the input is OFF.

Semi-basic instruction			Execution condition	Input ON	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
INV	FUN 22	Inverse line logic	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

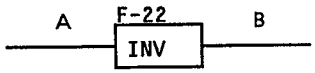
	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

Ladder circuit example

● Function description

This instruction inverts the line logic immediately before the instruction.

● Description



The relation between line logic A and line logic B is inverted as shown below.

Line logic A	ON(1)	OFF(0)
	↓	↓
Line logic B	OFF(0)	ON(1)

廃 番

Semi-basic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SET	FUN 23	Set	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	-

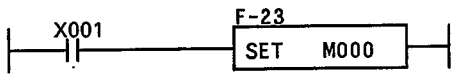
	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	-	-	-	-	-

Ladder circuit example

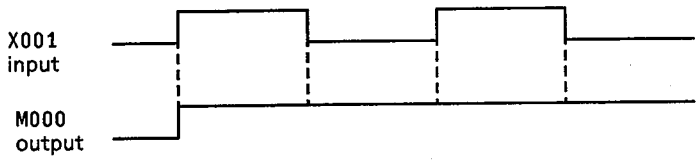
● Function description

This instruction turns ON the specified Z relay when the line logic is ON and stays ON even if the line logic is turned OFF after this.

● Description



When the input is X001 and output Z is M000, the following operation is performed.



A SET instruction is terminated after execution with an RST (reset) or an OUT instruction.

◇Note◇ Pulse settings cannot be made on the MX100, MX30 and MX20.

Semi-basic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RST	FUN 24	Reset	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y19F	Y200 to Y29F	M000 to M89F	L000 to L49F	M900 to M97F	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

● Function description

This instruction resets the specified device when the line logic is ON.

● Description

The table below shows the status of the device after the instruction is executed.

Device D	Status
Y, M, L	Relay is set to OFF.
R	The content is set to 0.

The reset device remains in reset status even when the line logic is OFF.
The bit device can be turned ON after the execution of an RST instruction using the SET instruction or the OUT instruction.

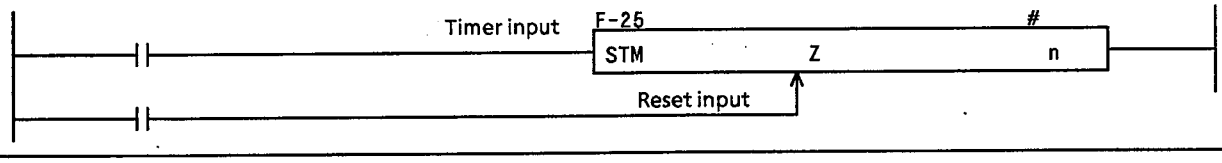
◇Note◇ Pulse settings cannot be made on the MX100, MX30 and MX20.

Semi-basic instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
STM	FUN 25	Accumulation timer	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-
n	-	-	-	-	#00.01 to #9999	-

Ladder circuit example

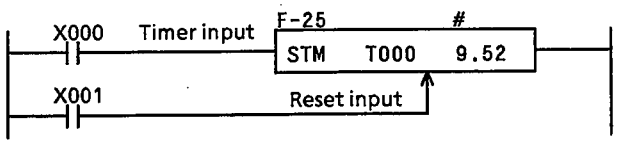


● Function description

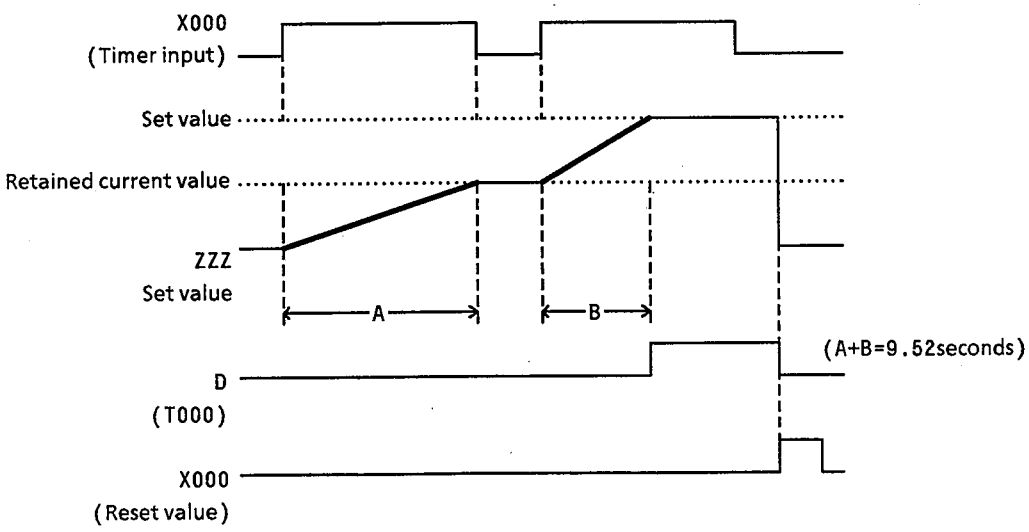
This instruction performs the same operation as the timer instruction (TMR), but the current value is not cleared, when the timer input is OFF. When the timer input is turned ON again, an addition operation is performed.
When the reset input goes ON, the current value of the timer is set to 0 and the timer is set to OFF.

● Description

The timer can be set to values between 0.01 seconds to 9999 seconds. Three types of units (10 ms, 100 ms and 1 s) can be set depending on the position of the decimal point.



The following operation is performed when the timer input is X000, the reset input is X001 and the timer is T000.

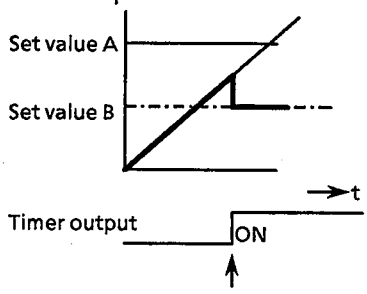


◇Notes◇

- The timer and counter are shared devices so the same number cannot be used for both (error code: 30).
- If an output instruction (OUT, MOV, etc.) are input after STM, a program error is caused.

```
LD    X000
LD    X001
STM   T000
  #   9.52
OUT   Y000  ← Program error
```

- Since the timer is the master clock (10 ms, 100 ms, 1 s) in the MX controllers, the smallest unit of the set value is also a measure of its deviation.
  For example, if the 10 ms timer is used to set 1.00 s, the deviation is also 10 ms. Likewise, if the 1 s timer is used, the deviation is 1 s.
  Consequently, when the setting is three digits or more, a 4-digit timer should be used for higher accuracy (i.e. a 999.0 s timer should be used for setting 999 s).
- The accuracy of timer output varies -0 to + 2 scans due to the timing of the timer input.
- Changes made in RUN mode
  When the set time is less than or equal to current time, the timer is activated, the output is set to ON and the current value and set value become equal.



Set value A→set value B

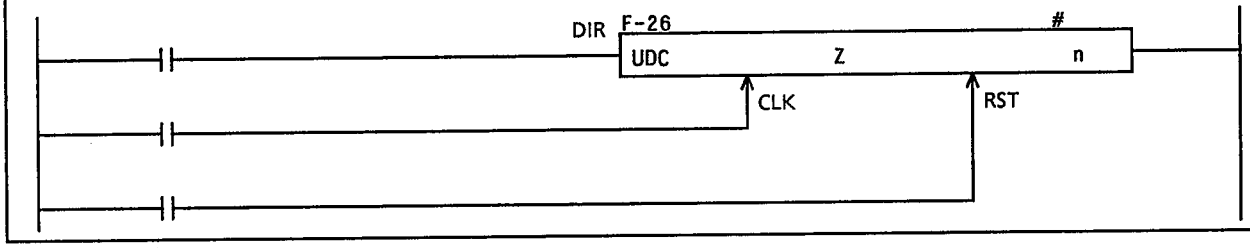
- An error is generated (error code :30) when the same timer number is used more than once.
- If a jump is made past the timer instruction, the timer is not triggered.
- Timer values do not match actual operation after a break.
- The current value is cleared when the input condition is OFF, modes are changed and the power is turned OFF.
- Timer accuracy can be set to two decimal digits.
- The deviation of the timer value is the sum of the number of times the timer input has been turned ON.

Semi-basic instruction			Execution condition	Rise	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
UDC	FUN 26	Up-down counter	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C255	-	-	-	-	-
n	-	-	-	-	#0000 to #9999	-

Ladder circuit example



● Function description

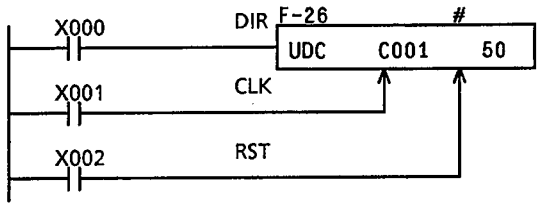
This instruction causes the current value of the counter specified as Z to be incremented by 1 or decremented by 1 when the directional input (DIR) and clock (CLK) pulses rise (goes from OFF to ON).

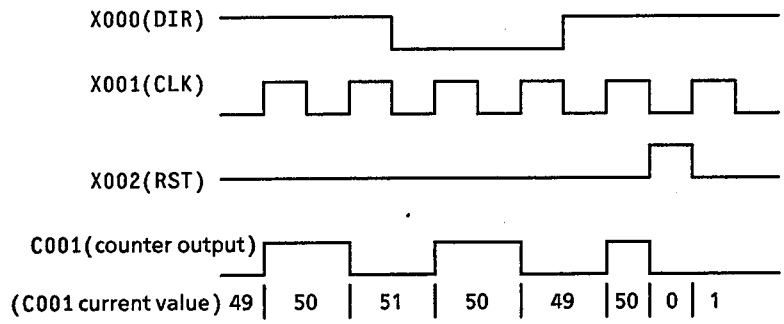
● Description

I/O	Status				
DIR	1	0	-	-	-
CLK	↑	↑	-	0	1
RST	0	0	1	0	0
Output	(See note 1.)		OFF	Status held	
Current value	1 added	1subtr acted	0	Status held	

◇Note◇ When the set value matches data in the register, the counter is activated, otherwise it is OFF.

When the directional input (DIR) is X000, the clock input (CLK) is X001, the reset input (RST) is X002, device (counter) is C:001 and setting value "n" is #50, the following operation is performed.





When the content of the register is a BCD value in the range 0 to 9999, adding 1 to a value of 9999 results in 0 and subtracting 1 from 0 results in 9999.

◇Notes◇

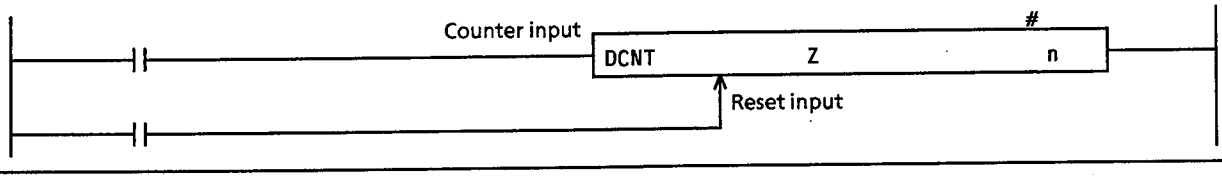
- The timer and counter are shared devices so the same number cannot be used for both (error code: 30).
- An output instruction (such as OUT or MOVE) cannot be placed after a UDC instruction. Such an input generates a program error in the program check (error code: 21).
- Failure to make a reset entry generates a program error in the program check (error code: 21).
- Mode changes and turning off the power does not clear the current value in the counter.
- When the controller goes from the STOP mode to the RUN mode while the clock (CLK) is ON, the current value is incremented or decremented by 1.

Semi-basic instruction			Execution condition	Rise	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DCNT	FUN 27	Double-word addition counter	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C254	-	-	-	-	-
n	-	-	-	-	#00000001 to #99999999	-

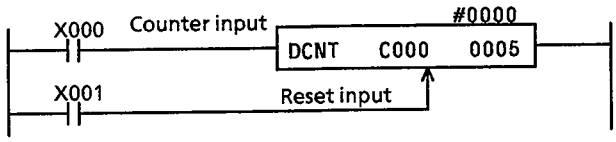
Ladder circuit example



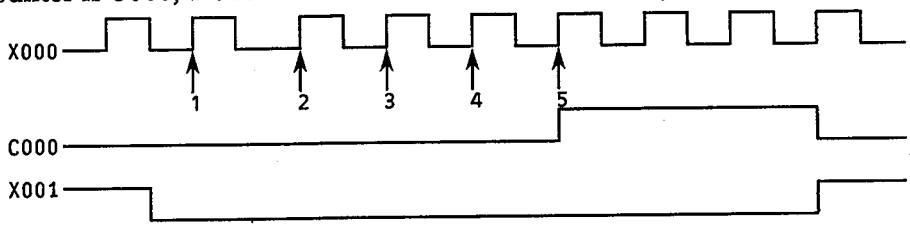
● Function description

This instruction turns on the 8-digit BCD counter when the count input rises (goes from OFF to ON). When a predetermined value is reached, the count is output.The counter is reset when the reset input is ON. Then the current value is reset to 0 and the counter output goes OFF. The counter stays in initialized status until the reset input goes OFF.

● Description



The following operation is performed when the counter input is X000, the counter is C000, the set value is 5 times and the reset input is X001.



◇Notes◇

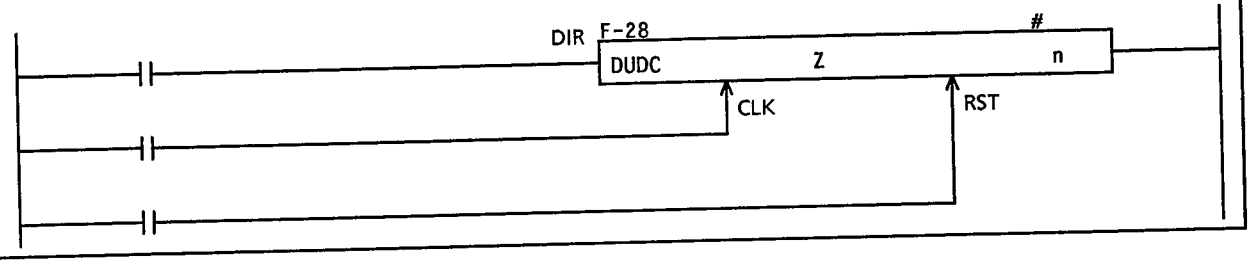
- The double-word counter instruction uses two contiguous counter areas. As a result, when DCNT uses C001, C002 cannot be used.
- The current value is not cleared when the input condition is OFF. However, it is cleared by a reset input.
- The timer and counter are shared devices so the same number cannot be used for both.
- Current values are not reset by mode transitions or power OFF conditions.
- The counter goes on when the current value = set value.
  The counter output does not go on when the set value is changed during counting or the current value is larger than the set value. It goes on only when the count input rises. The rising edge of the count output causes the set value on the counter to become the current value and an output is made.
- When there is a change from STOP to RUN mode and the counter input is on, the counter is incremented by 1.

Semi-basic instruction			Execution condition	Rise	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DUDC	FUN 28	Double-word up-down counter	9 bytes		Overflow M901	-	Underflow M902	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
Z	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
Z	T/C000 to T/C254	-	-	-	-	-
n	-	-	-	-	#00000000 to #99999999	-

Ladder circuit example



● Function description

This instruction is an 8-digit BCD reversible counter instruction which increments or decrements the current value of the counter specified as Z by 1 when the directional input (DIR) and clock (CLK) pulses rise (OFF→ON).

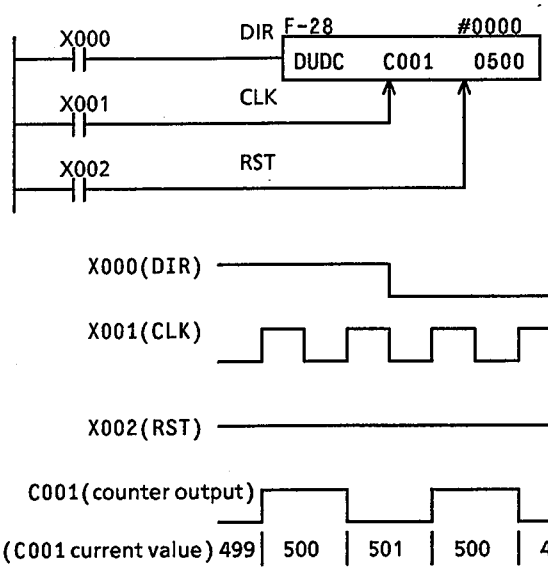
● Description

I/O	Status				
DIR	1	0	-	-	-
CLK	↑	↑	-	0	1
RST	0	0	1	0	0
Output	(See note 1.)		OFF	Status held	
Current value	1 added	1subtr acted	0	Status held	

1:ON
0:OFF
-:Undefined

◇Note◇ When the set value matches data in the register, the counter is activated. Otherwise it is OFF.

When the directional input (DIR) is X000, the clock input (CLK) is X001, the reset input (RST) is X002, device (counter) is C:001 and setting value "n" is #50, the following operation is performed.



When the content of the register is a BCD value in the range 0 to 9999, adding 1 to a value of 9999 results in 0 and subtracting 1 from 0 results in 9999.

◇Notes◇

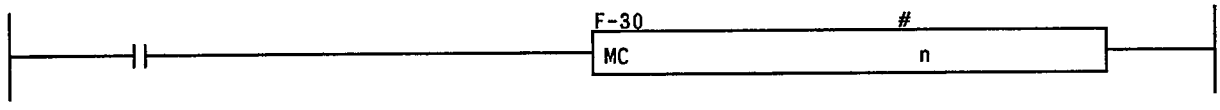
- The double-word counter instruction uses two contiguous counter areas. As a result, when C001 is used (for example, UDC C001 #500), the adjacent C002 area cannot be used.
- The timer and counter are shared devices so the same number cannot be used for both.
- An output instruction (such as OUT or MOVE) cannot be placed after a DUDC instruction. Such an input generates a program error in the program check.
- Current values are not reset by mode transitions or power OFF conditions.
- Failure to make a reset entry generates a program error (error code: 21).
- Mode changes and turning OFF the power does not clear the current value in the counter.
- When the controller goes from the STOP mode to the RUN mode while the clock (CLK) is ON, the current value is incremented or decremented by 1.

Branch instruction			Execution condition	Input ON	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
MC	FUN 30	Master control set	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#0 to #7	-

Ladder circuit example



● Function description

This instruction turns ON the master control contact when the input condition is set to ON. It uses the MCR (FUN31) with the same number to reset the master control contact.

● Description

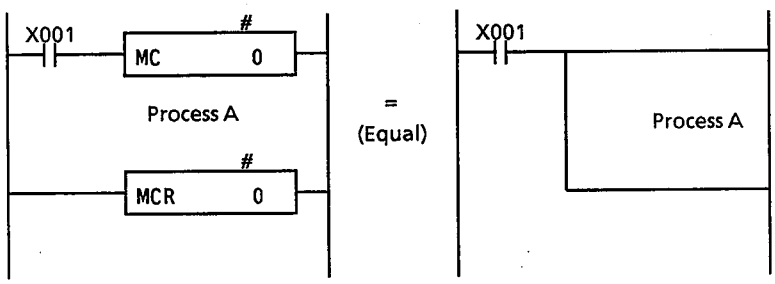
When the input condition is ON, the program is executed between MC and MCR and the result of the execution is output.
The processing time does not change when the input condition is OFF.
Processing between MC and MCR for different instructions is as follows

a) OUT instruction…… All outputs are set to OFF.

b) TMR(timer)instruction…… The output contact is OFF and current values are reset.

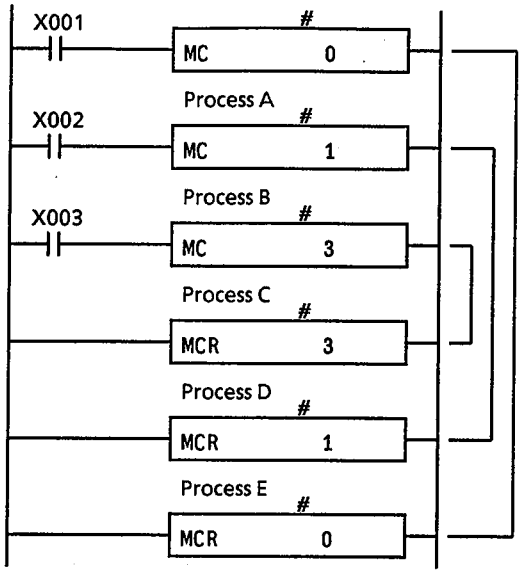
c) CNT(counter)instruction…… The output contact is OFF, but current values are retained.

d) STM(accumulation timer)instruction…… The output contact is OFF, but current values are retained.



- When a nested structure cannot be used, the same number can be used.
- Unlike the JMP instruction, the scan time does not change.

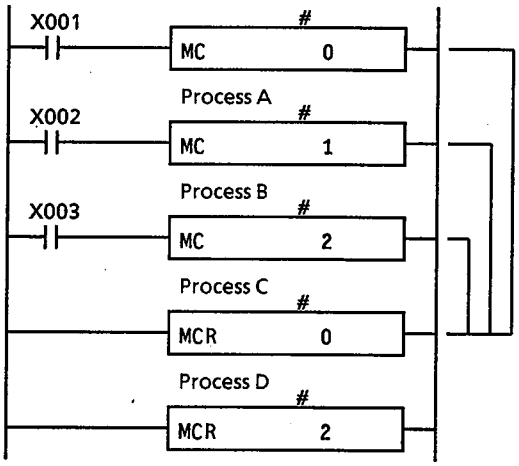
(Sequential setting and resetting)



- As shown in the figure, low numbers are used for MC-MCR.
- Numbers may be omitted.

X001	X002	X003	Programs executed
ON	OFF	OFF	A→E
ON	ON	OFF	A→B→D→E
ON	ON	ON	A→B→C→D→E

(Sequential setting and batch resetting)



- An MC with a number higher than that specified by an MCR is reset by a batch process. For this reason, MCR #2 does not function.

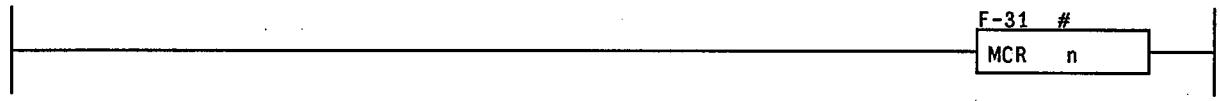
X001	X002	X003	Programs executed
ON	OFF	OFF	A
ON	ON	OFF	A→B
ON	ON	ON	A→B→C

Branch instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
MCR	FUN 31	Master control reset	5 bytes		verflow M901	-	Underflow M902	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#0 to #7	-

Ladder circuit program



● Function description

This instruction resets the master controller. (It indicates the end of the range of master control.)

● Description

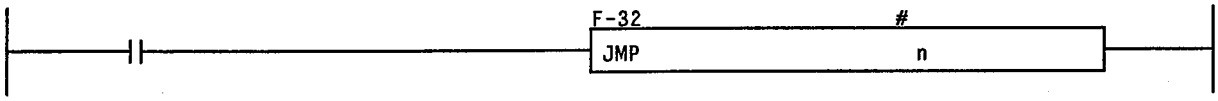
See MC (FUN30) for details.

Branch instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
JMP	FUN 32	Jump	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#000 to #499	-

Ladder circuit program

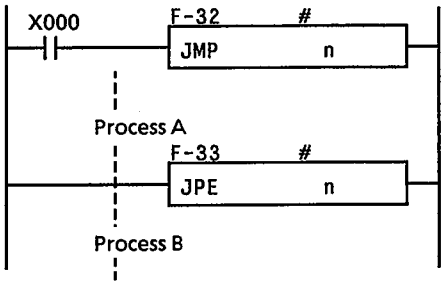


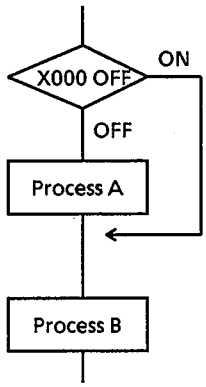
● Function description

This instruction causes the program to jump to the pointer location with the specified "n" number, when the input condition goes on. The destination of the jump is the JPE (FUN33) with the same number. The number "n" is a figure between #00 to 99 for MX100 and MX30.

● Description

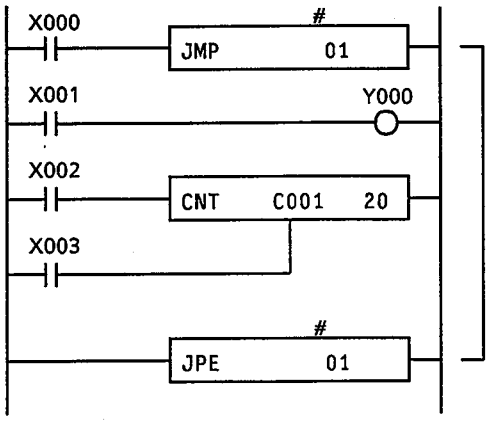
The following shows the basic operation of this instruction.





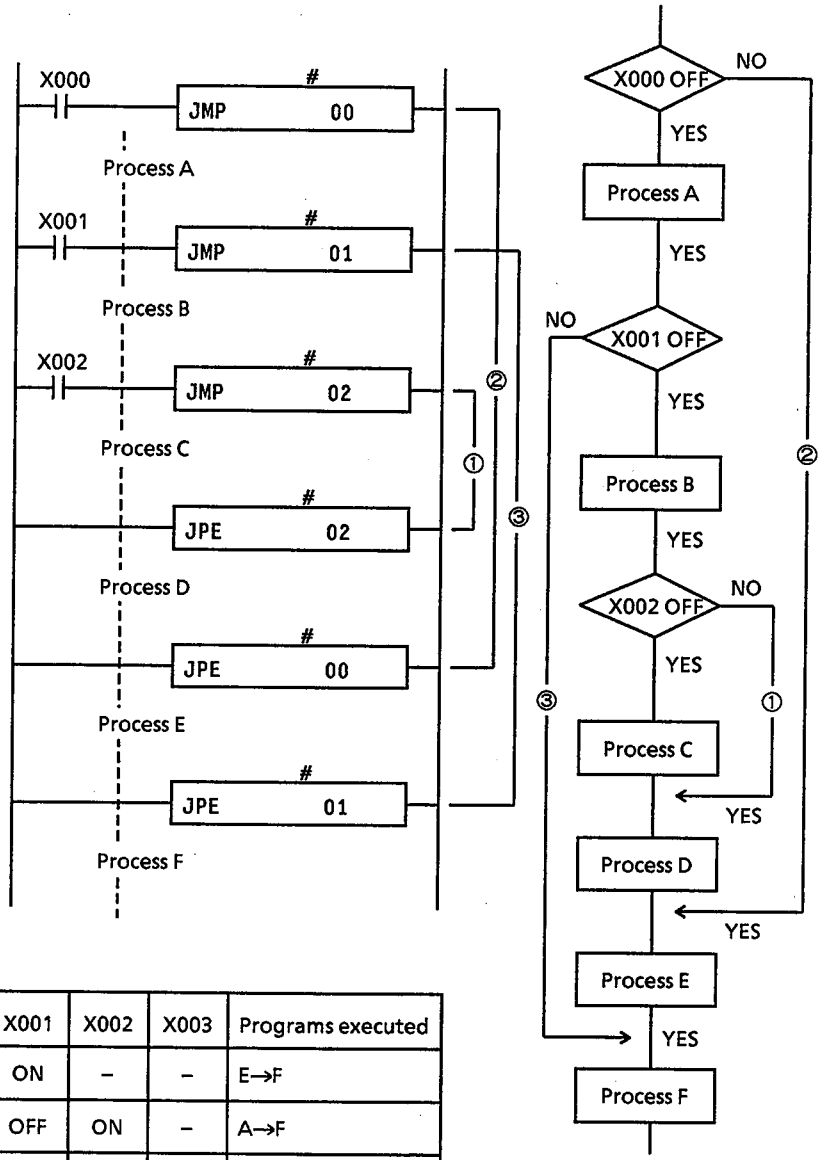
When X000 goes ON, process A is not executed (which shortens processing time), but processes B and beyond are executed. The output status of instructions and the current value of the timer counter of process A is retained.

[Example]



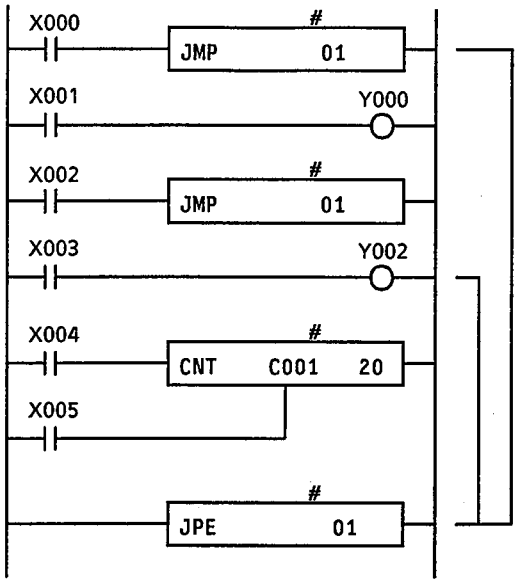
X000	Y000	C001
ON	Hold	Hold
OFF	Execu-tion	Execu-tion

JMP and JPE instruction are used in pairs (having the same number) in a nested structure. The figure below shows the nesting structure.



X001	X002	X003	Programs executed
ON	–	–	E→F
OFF	ON	–	A→F
OFF	OFF	ON	A→B→D→E→F

A JPE instruction can be paired with two or more JMP instructions with the same number. The figure below shows how such a structure functions.



X000	X002	Y000	Y001	C001
ON	ON	Hold	Hold	Hold
ON	OFF	Hold	Hold	Hold
OFF	ON	Execu-tion	Hold	Hold
OFF	OFF	Execu-tion	Execu-tion	Execu-tion

◇Notes◇

- The following conditions lead to program errors.
  ① When there is no JPE with the same number as the JMP instruction (error code: 24).
  ② When there are two or more JPE instructions with the same number (error code: 31).
  ③ When there is an input condition before a JPE instruction (error code: 21).
  ④ When there is an output instruction directly after a JPE instruction (error code: 21).
  ⑤ JPE which is the destination of a JMP instruction cannot be specified before the JMP instruction (error code: 2) [on MX100 and MX30].On the MX200, specifying a JPE instruction before the JMP instruction generates an alarm. (alarm code: 41).
  ⑥ An alarm is generated when only the JPE instruction is written (alarm code: 41).
- The portion of the program that was jumped over, is not executed. If this portion contains pulse instructions and pulse specifications, rising and falling edge processing is not performed. This may lead to unexpected behavior which requires consideration.

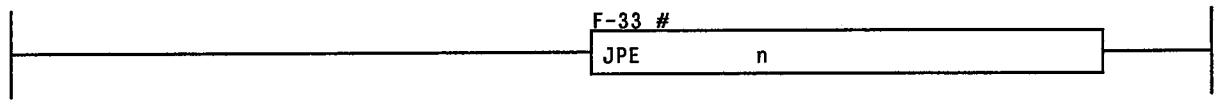
廃 番

Branch instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
JPE	FUN 33	Jump end	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#000 to #499	-

Ladder circuit example



● Function description

This instruction indicates the destination of a JMP instruction with the same "n" number.

- The number "n" is in the range #00 to #99 for MX100, MX30 and MX20.

● Description

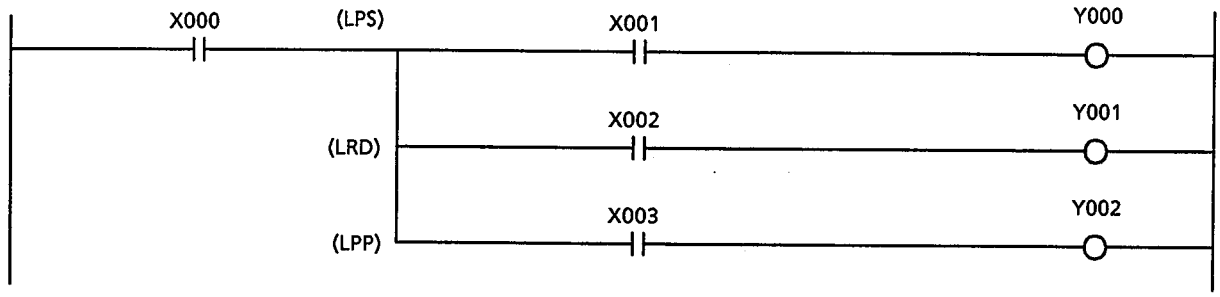
See JMP (FUN32) for details.

Branch instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
LPS	FUN 34	Line logic push	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

Ladder circuit example



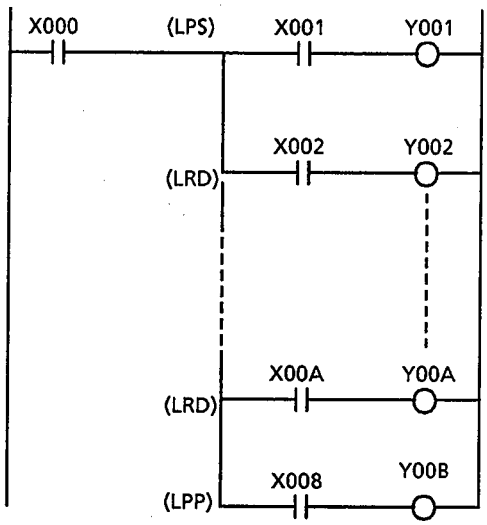
● Function description

This instruction causes the input line logic to be temporarily stored.

● Description

This instruction is used when several output circuits are created on the same line. The LPS, LRD and LPP instructions are used to indicate the starting point, relay point and end point of the branch, respectively.

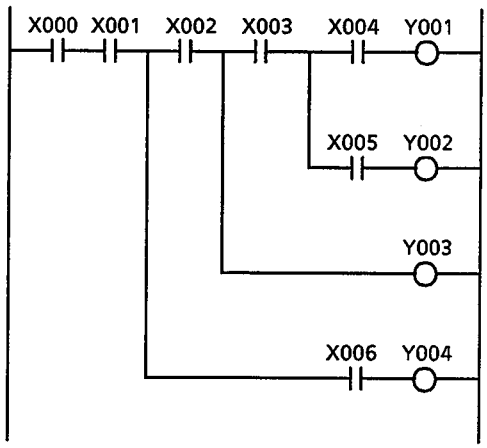
◇Note◇ LPS (FUN34), LRD (FUN35) and LPP (FUN36) are not displayed in the ladder circuit display of a personal computer loader.




```
Mnemonic
    LD    X000
    LPS
    AND   X001
    OUT   Y001
    LRD
    AND   X002
    OUT   Y002
    |
    LRD
    AND   X00A
    OUT   Y00A
    LPP
    AND   X00B
    OUT   Y00B
```

[Example]

(Ladder display)



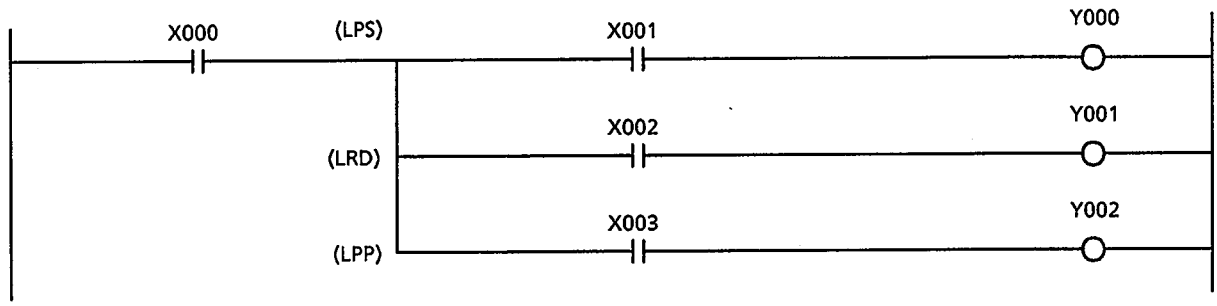

```
(Mnemonic)
    LD    X000
    AND   X001
    LPS
    AND   X002
    LPS
    AND   X003
    LPS
    AND   X004
    OUT   Y001
    LPP
    AND   X005
    OUT   Y002
    LPP
    OUT   Y003
    LPP
    AND   X006
    OUT   Y004
```

Branch instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
LRD	FUN 35	Line logic read	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

Ladder circuit example



● Function description

This instruction reads the line logic stored by LPS.

● Description

This instruction is used when several output circuits are created on the same line. The LRD instruction is a relay point for the branch and LPS.

◇Note◇ LPS (FUN34), LRD (FUN35) and LPP (FUN36) are not displayed in the ladder circuit display of a personal computer loader.

Branch instruction

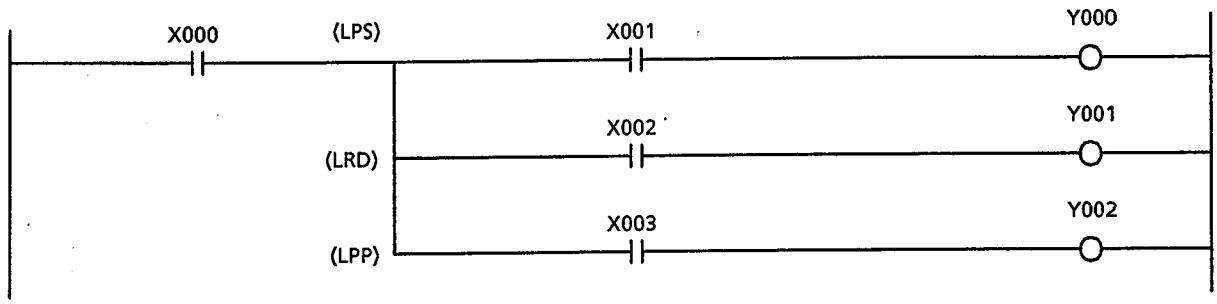
Execution condition	Always executed	Pulse specification	-	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
LPP	FUN 36	Line logic pop	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

Ladder circuit example



● Function description

This instruction releases the read operation of the line logic memory.

● Description

This instruction is used when several output circuits are created on the same line. The LPP instruction indicates the end point of the branch starting from LPS.

廃番

Branch instruction

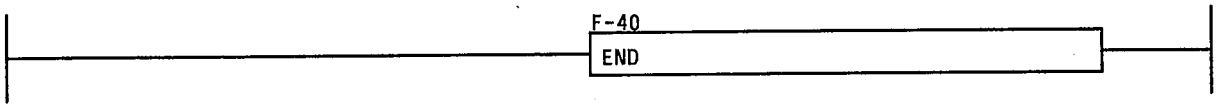
Execution condition	Always executed	Pulse specification	-	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
END	FUN 40	Program end	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

Ladder circuit example



● Function description

This instruction indicates the end of the program. Program execution stops after this instruction.

● Description

The END instruction is placed at the end of the program and an alarm is generated if it is missing (alarm code: 43).

Branch instruction

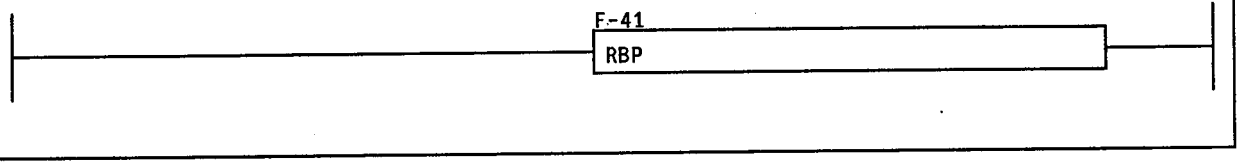
Execution condition	Always executed	Pulse specification	-	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RBP	FUN 41	Initial jump	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

Ladder circuit example

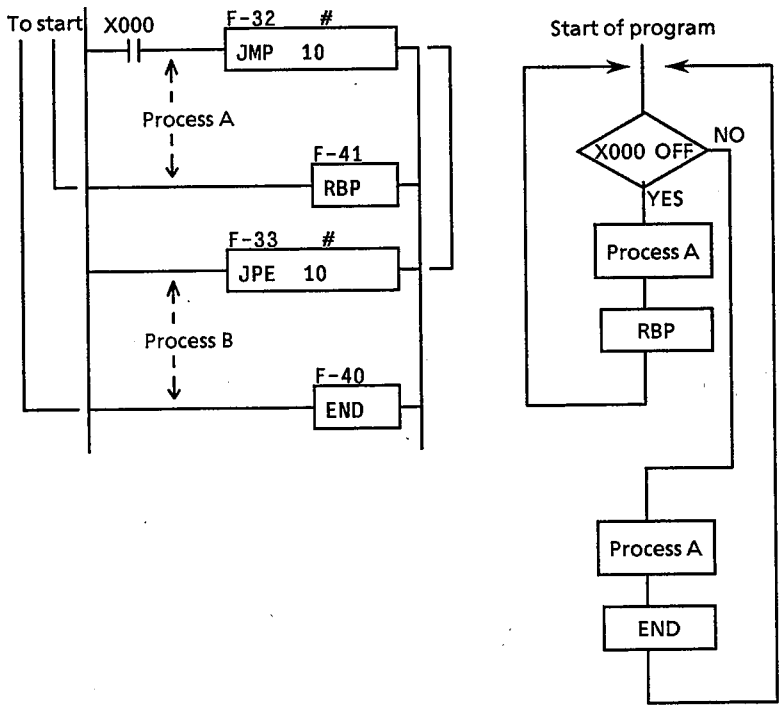


● Function description

This instruction terminates the program unconditionally. In combination with subroutine calls or jump instructions, it can be used to shorten programs or set conditions.

● Description

The following process can be performed in combination with the JMP or JPE instructions.



◇Note◇ This instruction cannot be used instead of the END (FUN40) instruction. A program must have an END instruction at the end.

廃番

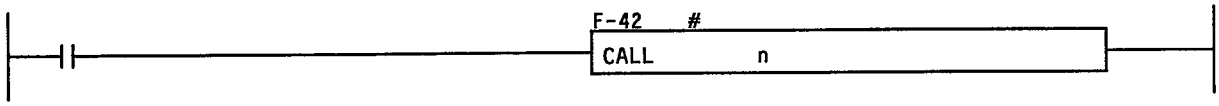
Branch instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
CALL	FUN 42	Subroutine call	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#000 to #499	-

Ladder circuit example

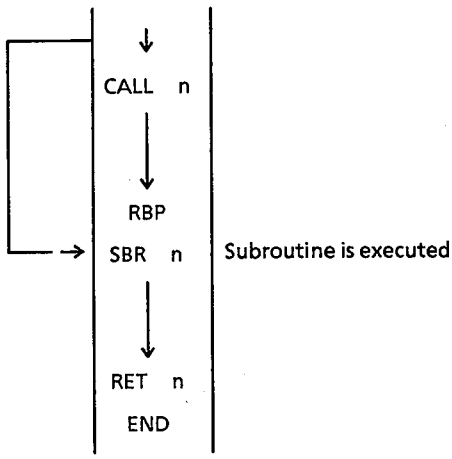


● Function description

This instruction calls subroutine (SBR) with number "n" when the input condition is on. The number "n" is in the range #00 to #99 for MX100 and MX30.

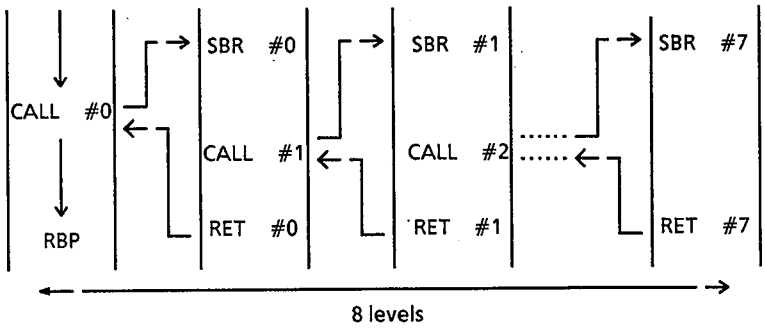
● Description

CALL n calls SBR n with the same number and the program in SBR n and RET n is executed.



The subroutine can consist of up to 8 levels of nesting.

◇Note◇ A program cannot have two subroutines with the same number.



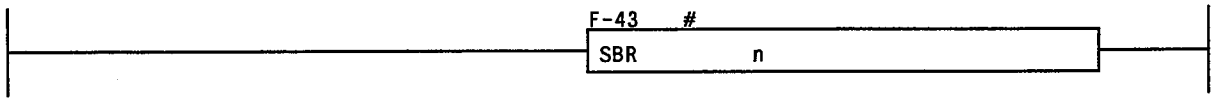
廃 番

Branch instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SBR	FUN 43	Subroutine start	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

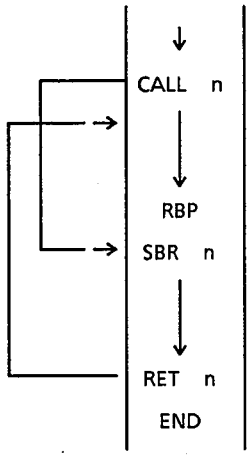
	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#000 to #499	-

Ladder circuit example



● Function description

This instruction indicates the start of subroutine with the number indicated by CALL.



Subroutine n is executed

The number "n" is in the range #00 to #99 for MX100, MX30 and MX20.

● Description

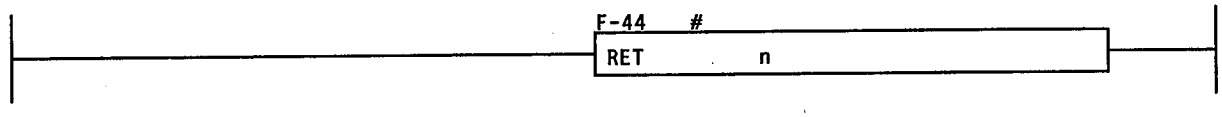
- See CALL (FUN42) for details.
- Place this instruction after RBP and RET (see error: 22).

Branch instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RET	FUN 44	Subroutine return	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#000 to #499	-

Ladder circuit example



● Function description

This instruction returns control to the instruction following the called CALL instruction.
The number "n" is in the range #00 to #99 for MX100 and MX30.

● Description

- See CALL (FUN42) for details.
- If two subroutines with the same number are executed in succession, an error is generated.

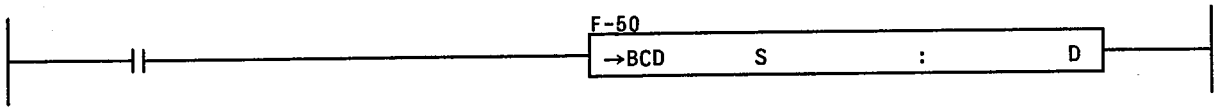
廃番

Conversion instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
→BCD	FUN 50	BIN→BCD conversion	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

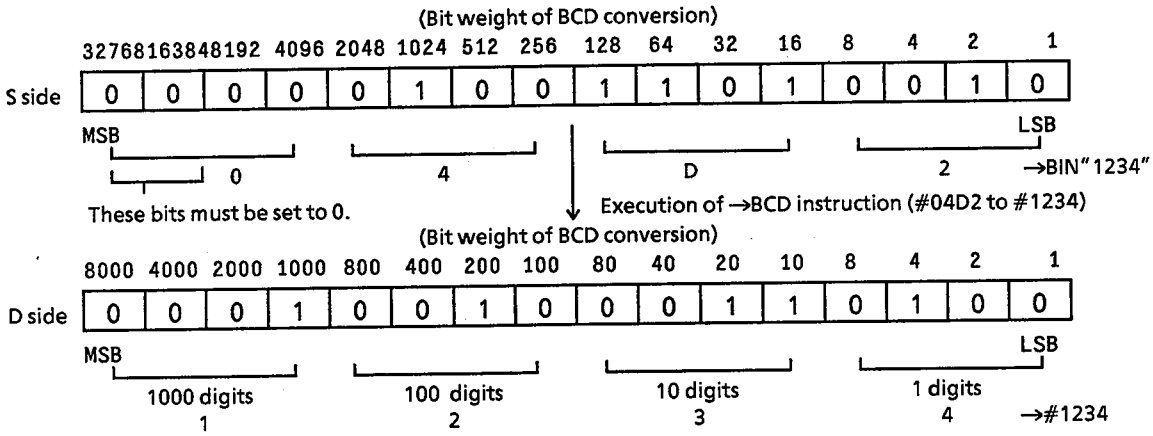


● Function description

This instruction converts one word of S binary (BIN) data to 4-digit BCD data and passes the result to D.

● Description

When S starts from #04D2, the following operation is performed.



◇Note◇ Since this is a conversion to BCD data, BIN data (exceeding #270F) at S that produces a D value that is larger than 9999 will generate an operation error (M90E and M90F go on) causing the operation to be aborted.

Conversion instruction

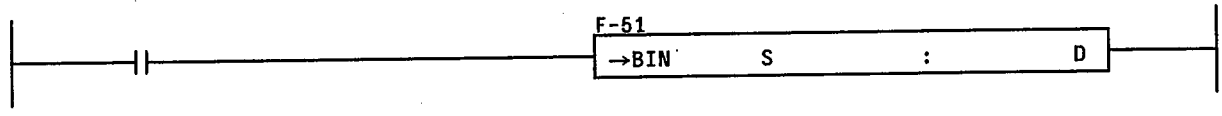
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
→BIN	FUN 51	BCD→BIN conversion	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

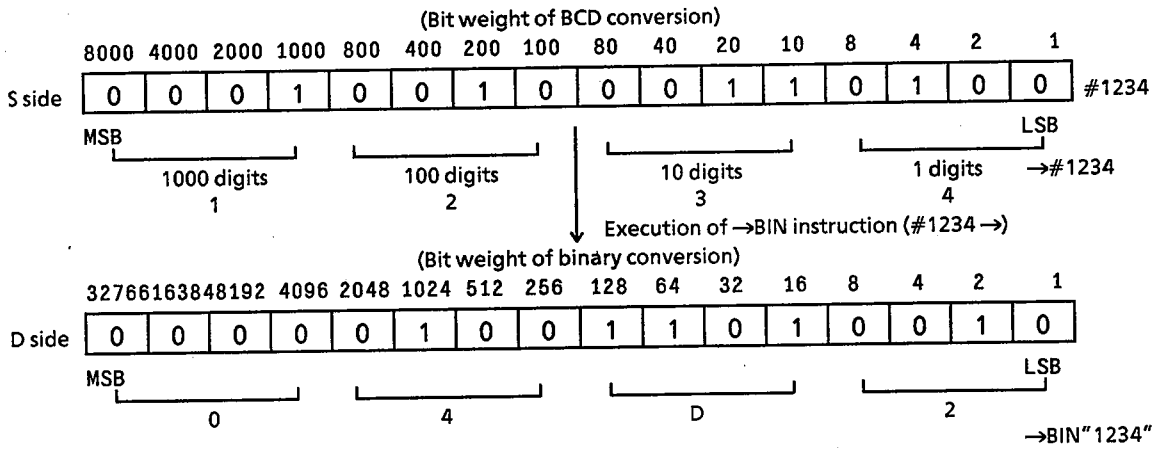


● Function description

This instruction converts 4-digit BCD data at S to binary (BIN) data which is passed to D.

● Description

When S starts from #1234, the following operation is performed.



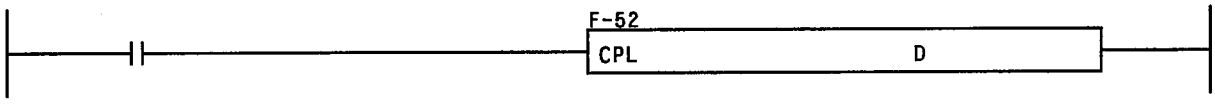
◇Note◇ When a digit at S is something other than a BCD value (0 to 9), an operation error (M90E and M90F go on) is generated and the operation is aborted.

Conversion instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
CPL	FUN 52	Data inversion	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

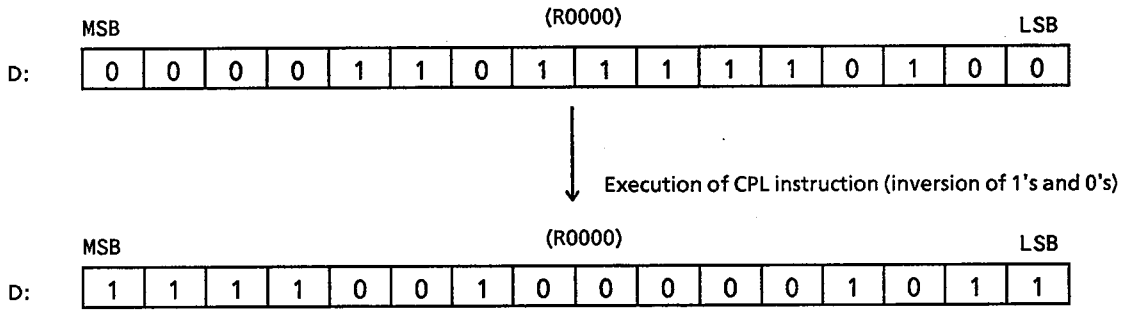


● Function description

This instruction performs a logic inversion of each bit of one-word data at D.

● Description

When D is in R0000, the following operation is performed.



Conversion instruction

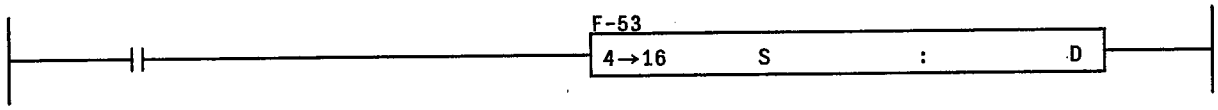
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
4→16	FUN 53	4 → 16 decode	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X19C	X200 to X29C	Y000 to Y19C	Y200 to Y29C	M000 to M89C	L000 to L49C	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



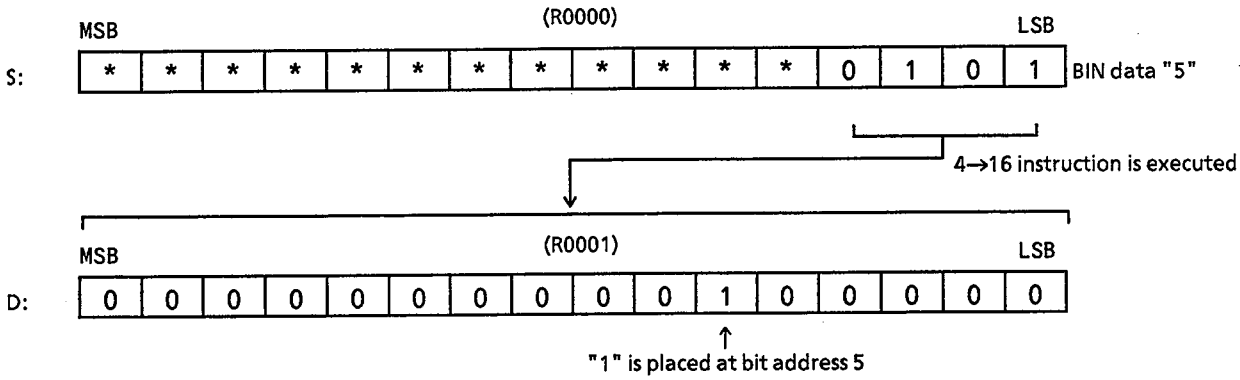
● Function description

This instruction decodes 4-bit data to 16-bit data which is passed to D.

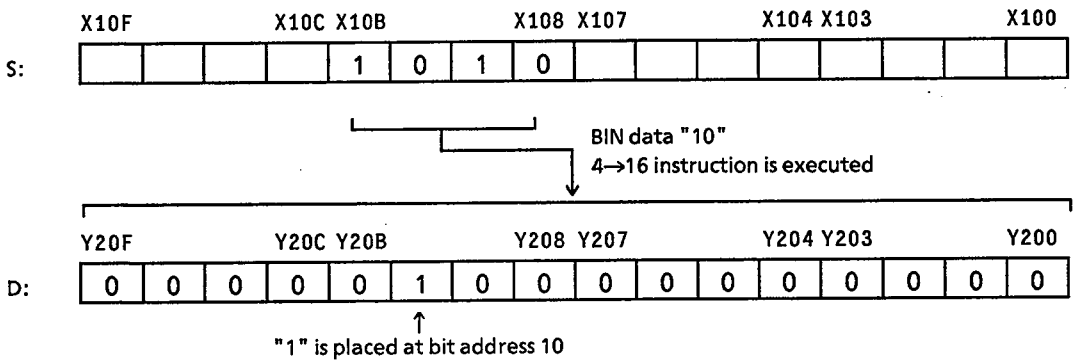
● Description

When S is a relay area, the least significant digit of the data to be converted can be specified as 0, 4, 8 or C.
When S is a register, the low-order 4-bits are decoded.

i) When S is in R0000 and D in R0001, the following operation is performed.



ii) When S starts from X108 and D starts from Y200, the following operation is performed.



Conversion instruction

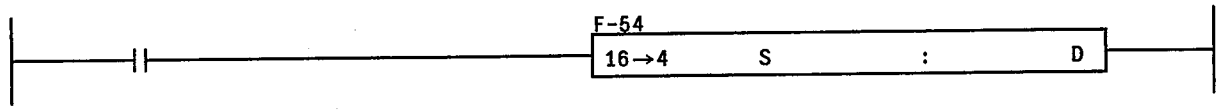
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags				
16→4	FUN 54	16 →4 encode	7 bytes		Shift carry M900	-	Underflow M902	-
					Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to 19C	Y200 to Y29C	M000 to M89C	L000 to L49C	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



● Function description

This instruction encodes word data at S to 4-bit data which is passed to D.

● Description

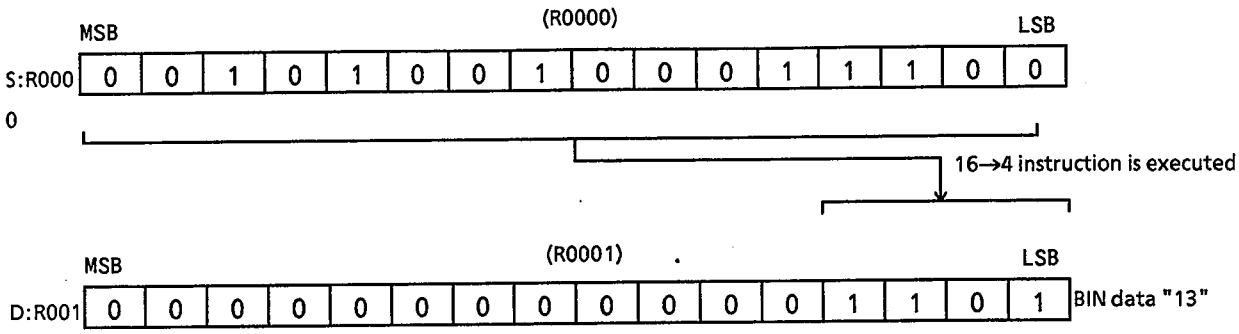
When D is a relay area, the least significant digit of the data to be converted can be specified as 0, 4, 8 or C.
When D is a register, the encoded data is stored in the low-order 4-bits while all the high-order 12 bits are set to 0.

When there are several 0's in the data to be encoded, the first "1" in the high-order (MSB) bits is used while the other bits are ignored.

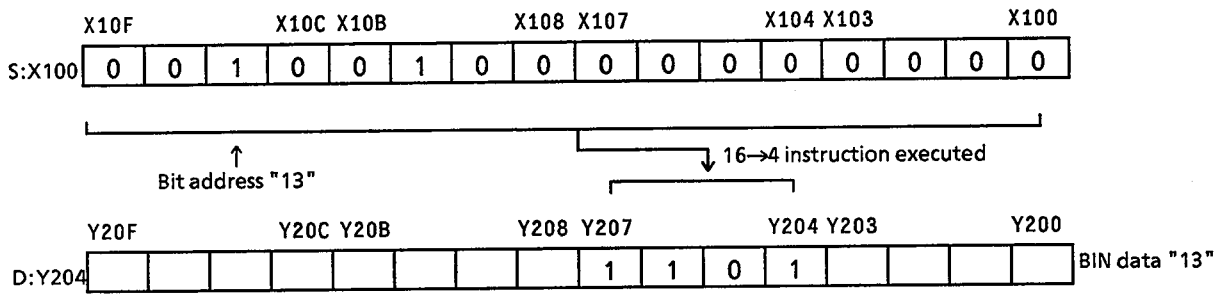
[Example]

S: 0010 1001 0001 1100 → 0010 0000 0000 0000

i) When S is in R0000 and D in R0001, the following operation is performed.



ii) When S starts from X100 and D from Y204, the following operation is performed.



◇Note◇ When S data is all 0's, an operation error (M90E, M90F) is generated and the operation is aborted.

Conversion instruction

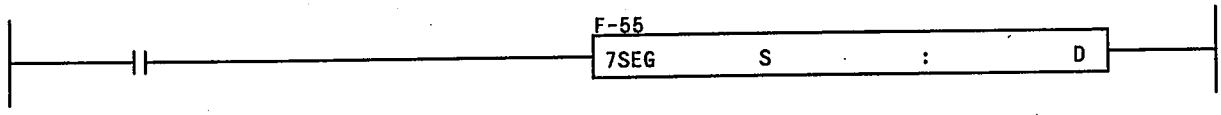
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags				
7SEG	FUN 55	7 SEG decode	7 bytes		Shift carry M900	-	Underflow M902	-
					Overflow M901	-	Operation error M90E	-

	Input X	OP link input Y	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X19C	X200 to X29C	Y000 to Y19C	Y200 to Y29C	M000 to M89C	L000 to L49C	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y198	Y200 to Y298	M000 to M898	L000 to L498	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



● Function description

This instruction reads 4-bit data S as BIN data and decodes it as 7-bit data for 7-segment use and passes the result to D.

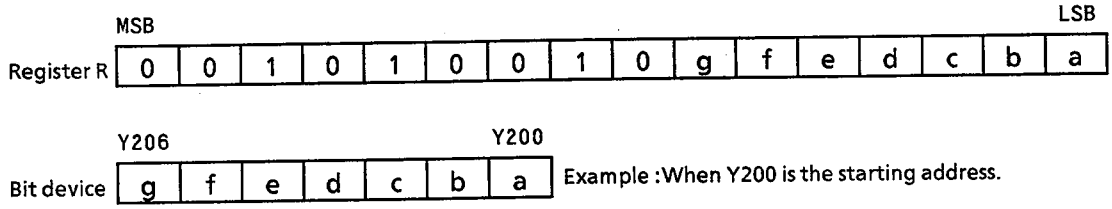
● Description

The low-order 4-bits of 1-word data (16-bit data) S in register R are decoded.
Four contacts starting from the specified contact number are decoded in the bit device.
The low-order 1-digit of the contact is specified as 0, 4, 8 or C.

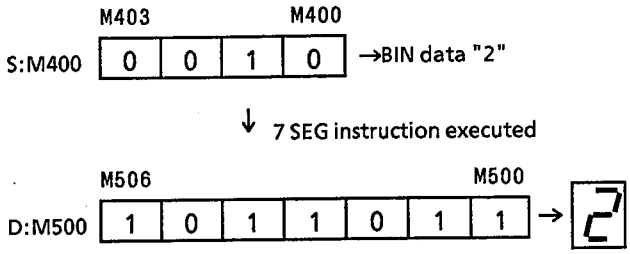
The decoded data is stored in the low-order 7-digits of the 16 bits in register R which is destination D.
The high-order 9 bits are all set to 0.

The low-order 1-digit of the bit device is specified as 0, or 8.

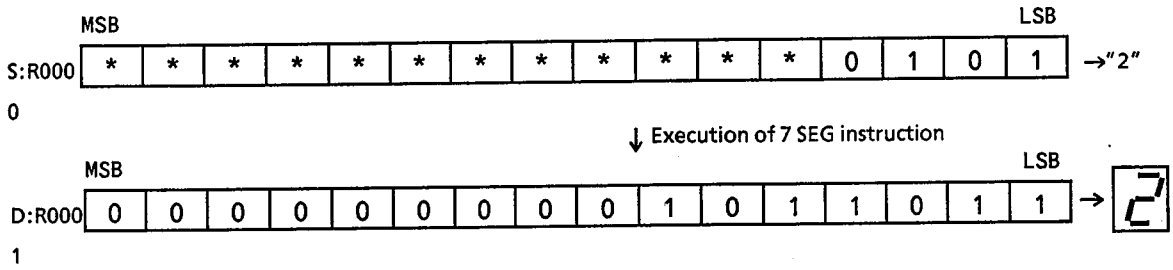
Detail drawing showing destination D



i) Example 1 : When S starts at M400 and D at M500



ii) Example 2 : When S starts at R0000 and D at R0001



7-segment decode table

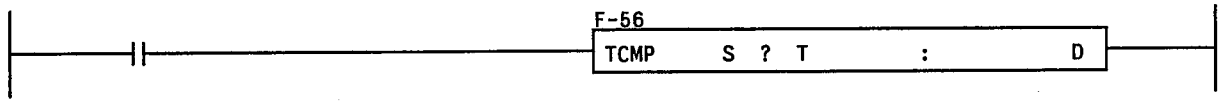
Data S, the data to be decoded (* arbitrary value) Low numbers of LSB or contacts ↓ MSB	D storing decoded data y f e d c b a Low numbers of LSB or contacts ↓ MSB	Display output a f g b e c d
* ⟷ *0000 0000	0 ⟷ 00111111 0111111	0
* ⟷ *0001 0001	0 ⟷ 00000110 0000110	1
* ⟷ *0010 0010	0 ⟷ 01011011 1011011	2
* ⟷ *0011 0011	0 ⟷ 01001111 1001111	3
* ⟷ *0100 0100	0 ⟷ 01100110 1100110	4
* ⟷ *0101 0101	0 ⟷ 01101101 1101101	5
* ⟷ *0110 0110	0 ⟷ 01111101 01111101	6
* ⟷ *0111 0111	0 ⟷ 00100111 0100111	7
* ⟷ *1000 1000	0 ⟷ 01111111 1111111	8
* ⟷ *1001 1001	0 ⟷ 01101111 1101111	9
* ⟷ *1010 1010	0 ⟷ 01110111 1110111	A
* ⟷ *1011 1011	0 ⟷ 01111100 1111100	b
* ⟷ *1100 1100	0 ⟷ 00111001 0111001	C
* ⟷ *1101 1101	0 ⟷ 01011110 1011110	d
* ⟷ *1110 1110	0 ⟷ 01111001 1111001	E
* ⟷ *1111 1111	0 ⟷ 01110001 1110001	F

Conversion instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
TCMP	FUN 56	Table matching	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
T	-	-	-	-	M000 to M740	L000 to L340	-	-	R0600 to R604
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
T	-	R0000 to R0484 R1000 to R4884	P0000 to P3784	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



● Function description

This instruction performs a word by word comparison of word data S with table data consisting of 16 words (16 words starting from T). The result of the comparison is placed in 16-bit device D (1 indicates a match and 0 a mismatch).

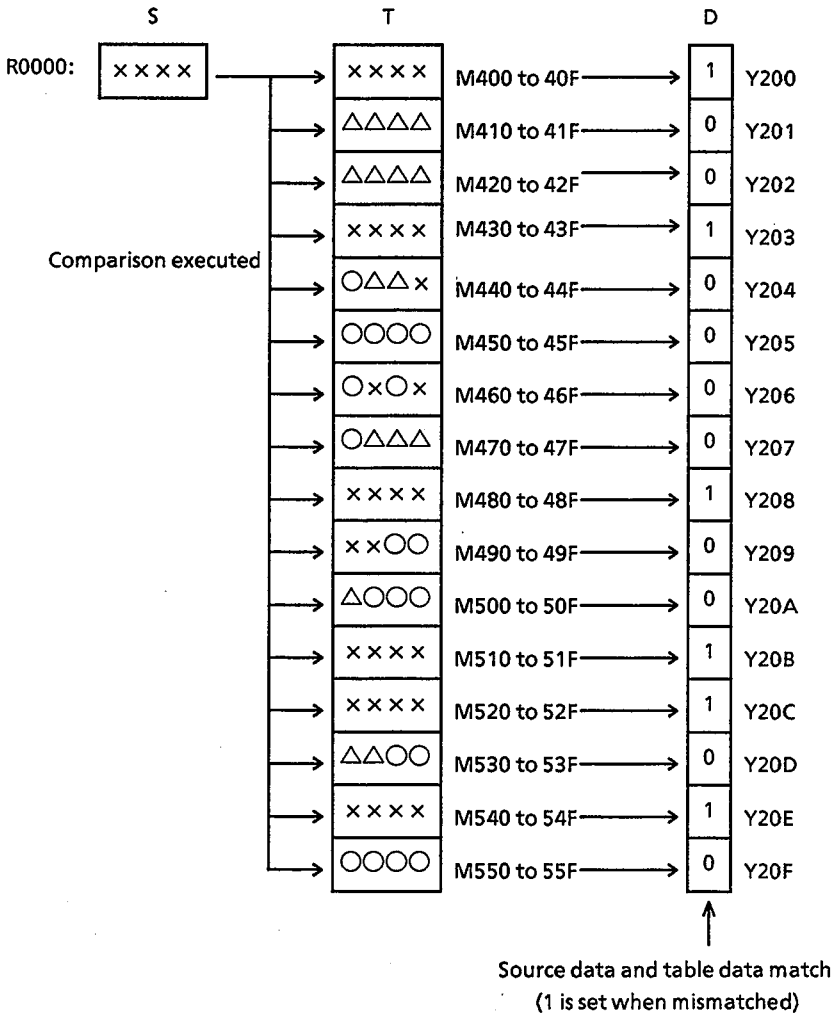
● Description

S: 1-word comparison data starting from the first number of a register, constant or bit device
Register: R0000 (example)
Bit device: X000 (X000 to X00F is one data unit) (example)

T: Table data used in comparison (1 table consists of 16 words.); From start of register or contact
Register: R0000 (table range R0000 to R0015) (example)
Bit device: M400 (table range M400 to M55F) (example)

D: Destination of comparison result, 16-bit data; First number in register or contact
Register: R0000 (example)
Bit device: Y200 (Y200 to Y20F is one data unit)

When S: is in R0000, T: starts from M400 and D: starts from Y200, the data is processed as shown below.

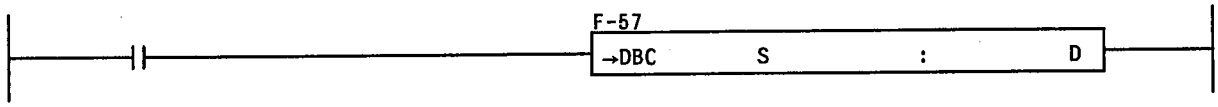


Conversion instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
→DBC	FUN 57	Double-word BIN→BCD conversion	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

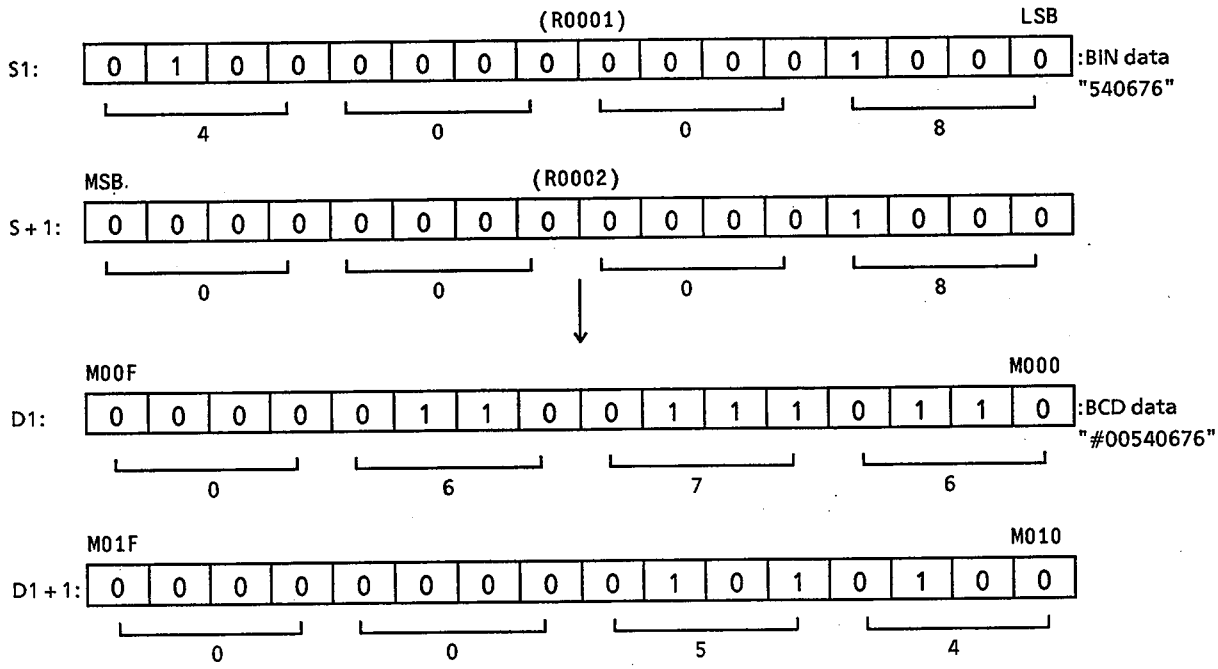


● Function description

This instruction converts two word 8-digit BCD data at S + 1 (high-order) and S (low-order) to BIN data which is passed to D + 1 (high-order) and D (low-order).

● Description

When S is in R0001 and D starts from M000, the data is processed as shown below.



◇Note◇ Since the data is converted to BCD data, S +1 and S cannot contain BIN data values that cause a result at D + 1 and D that exceeds 99999999. For example,

S + 1,S : 05F5E0FF(BIN data)

↓

D + 1,D : 99999999(BCD data)

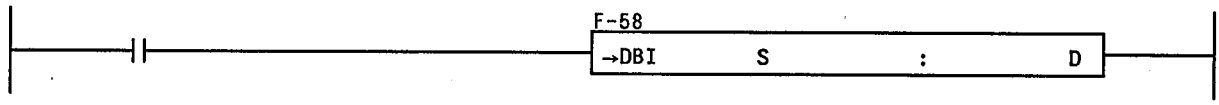
generates an operation error (M90E, M90F) and the process is aborted.

Conversion instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
→DBI	FUN 58	Double-word BCD→BIN conversion	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

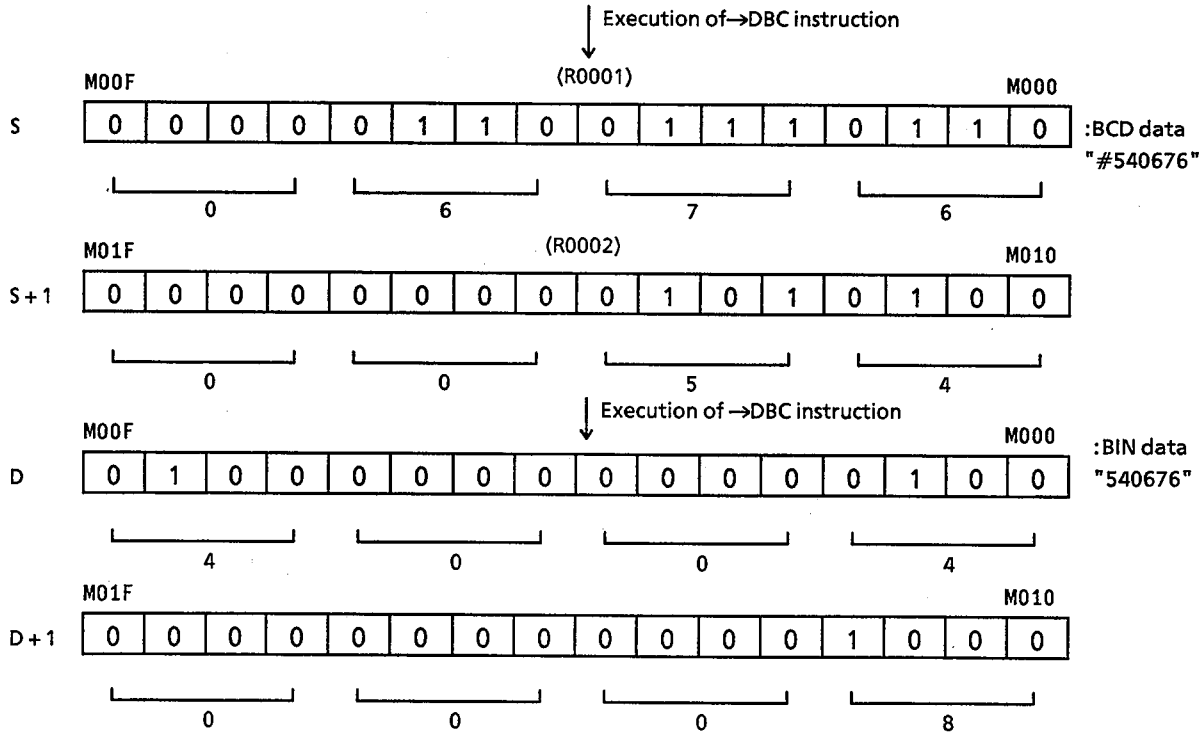


● Function description

This instruction converts two word 8-digit BCD data S + 1 (high-order), S (low-order) to BIN data which is passed to D + 1 (high-order) and D (low-order)

● Description

When S is in R0001 and D starts from M000, the data is processed as shown below.



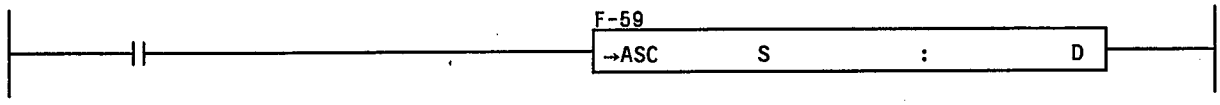
◇Note◇ When a digit of S + 1 or S1 is a value other than a BCD value (0 to 9) an operation error (M90E and M90F) is generated and the operation is aborted.

Conversion instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
→ASC	FUN 59	ASCII code conversion	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	#0000 to #FFFF	K-32768 to 32767
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

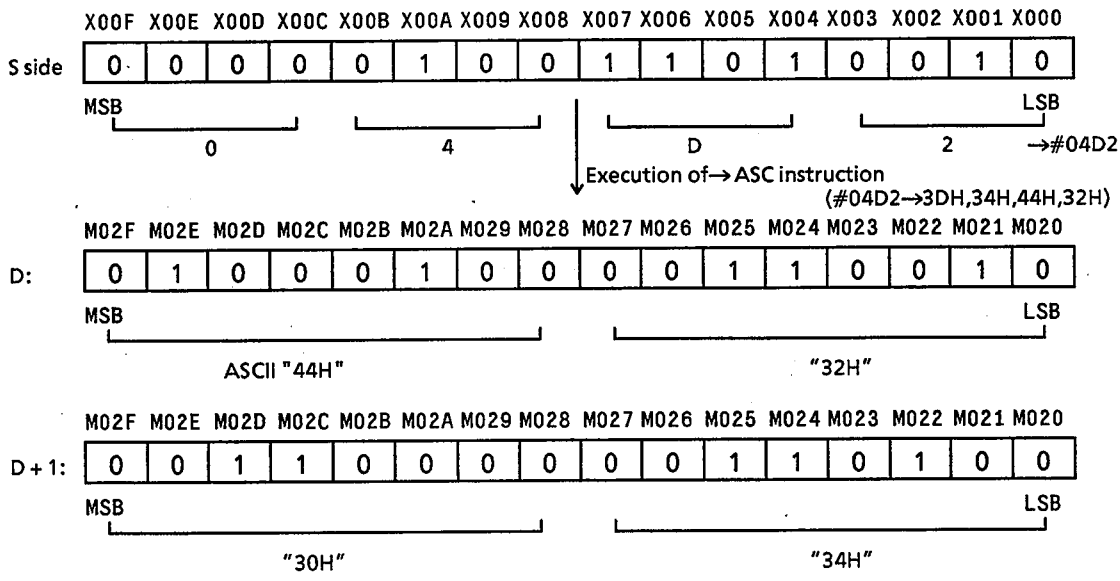


● Function description

This instruction converts one word 4-digit hexadecimal data S into two word ASCII data (0 to 9 and A to F). The high-order 2-digits are passed to D + 1 and the low-order 2-digits are passed to D.

● Description

When S starts at X000 and D at M020, the data is processed as shown below.



Shift instruction

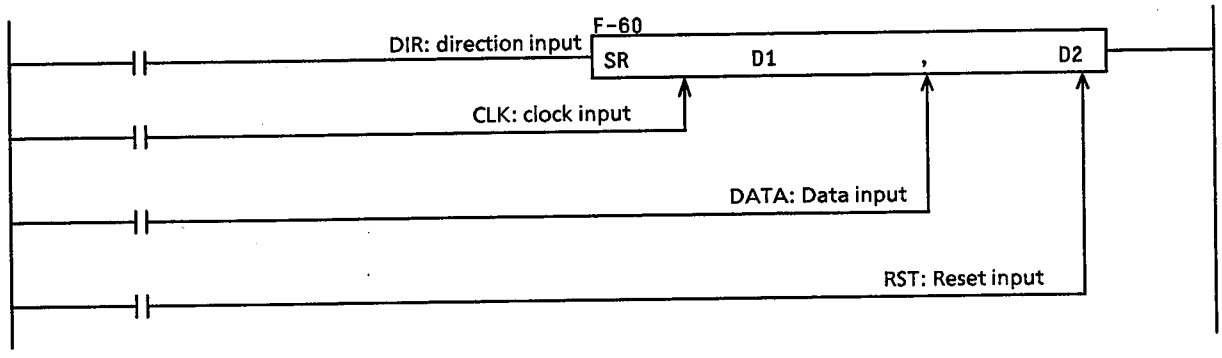
Execution condition	Rise	Pulse specification	-	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	○	Underflow M902	-
SR	FUN 60	Bidirectional shift	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D1	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

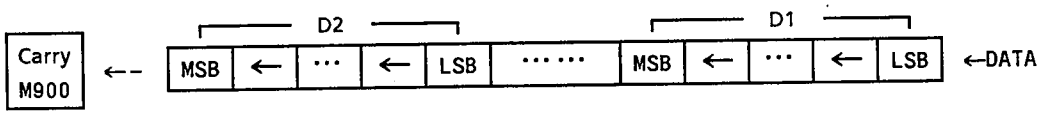


● Function description

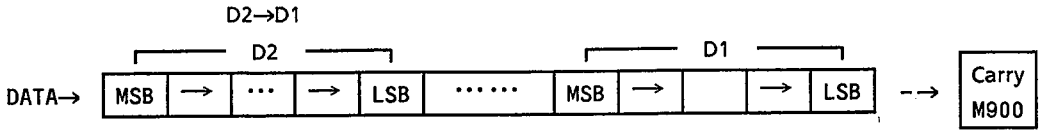
This instruction fetches data input (DATA) when the clock input rises and moves one bit of D1 to D2 (D1 is the first word and D2 is the last word, D1≦ D2) in the direction indicated by the direction input.

● Description

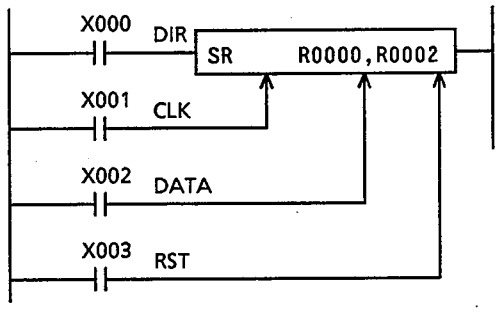
CLK : Shift execution clock which is triggered at signal
DIR : Shift direction
Input ON: Left shift LSB→MSB
D1→D2



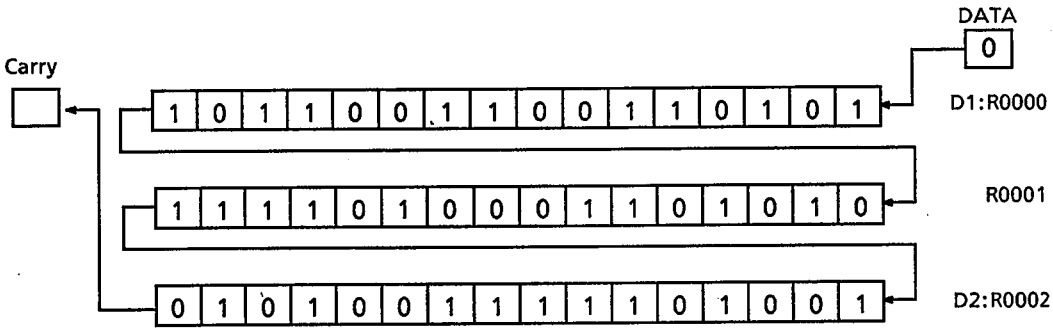
Input OFF: Right shift MSB → LSB



DATA : Data input Data fetched when the CLK input rises
RST : D1 to D2 are cleared (OFF) when the reset input goes on



Example : When D1 is in R0000 and D2 is in R0002, the data is processed as shown below.
X000 : When ON (left shift D1 → D2)
X002 : When OFF



When X001 rises from OFF to ON, DATA (0 in the example) and the data is shifted to the left. The carry flag stores the MSB 0 in R0002 (D2).

◇Notes◇

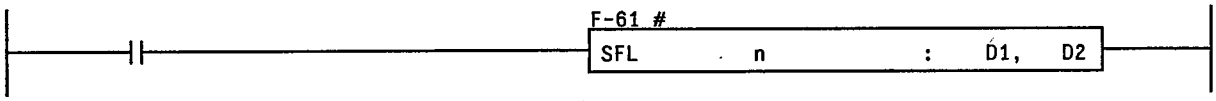
- When a shift is executed, the data fetched from D1 to D2 is stored in carry (M900).
  When DIR is ON, the MSB value of D2 is shifted to carry.
  When DIR is OFF, the LSB value of D1 is shifted to carry.
- D1 to D2 must be in the same area, for example, if D1 is in M000 and D2 is in R0000, a program error is generated.
- An output instruction (OUT, MOV or the like) cannot be input after an SR instruction.
- The total number of LD, LDNOT, LD compare and DLD compare instructions before an SR instruction must meet the conditions set for each instruction or a program error (error code: 25) is generated.
- The output instructions and LD, LDNOT, LD compare, DLD compare, ANB and ORB instructions before an SR instruction must meet the following conditions or a program error is generated (error code: 25).
  When there are two inputs: (total of LD, LDNOT and other instructions) = 2 + (total of ANB and ORB instructions)
  When there are three inputs: (total of LD, LDNOT and other instructions) = 3 + (total of ANB and ORB instructions)
  When there are four inputs: (total of LD, LDNOT and other instructions) = 4 + (total of ANB and ORB instructions)

Shift instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	○	Underflow M902	-
SFL	FUN 61	Left shift	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-
D1	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L480	-	-	R0600 to R0619
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	-	-	-	#1、4、8、16	-
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P000 to P3799	-	-	-

Ladder circuit example

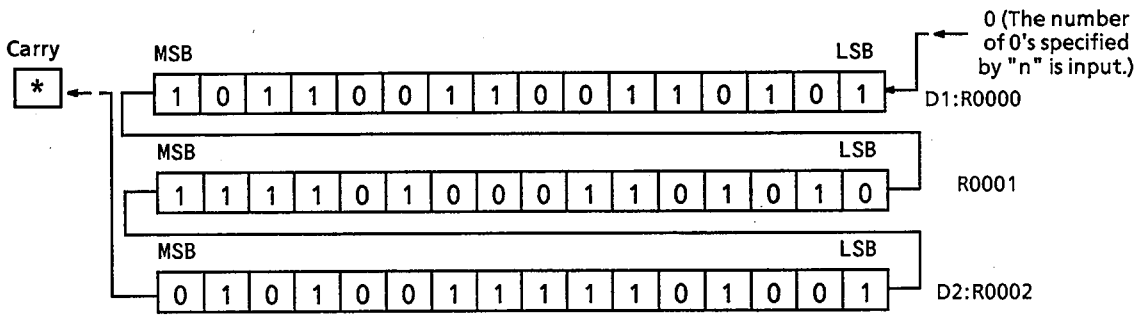


● Function description

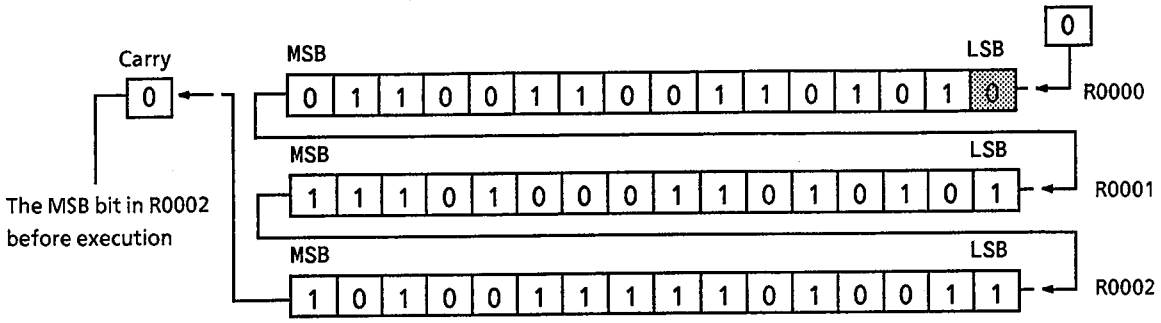
This instruction shifts the number of bits specified by "n" to the left (LSB→MSB) from a word in specified word area D1 to D2.
D1 specifies the first word and D2 the last word (D1≦D2)
When D1 = D2, word data (16 bits) is shifted by "n" bits.
When "n" is 1, data is shifted by 1 bit to the left.
When "n" is 4, data is shifted by 4 bits to the left.
When "n" is 8, data is shifted by 8 bits to the left.
When "n" is 16, data is shifted by 16 bits to the left.
The shift carry (M900) is connected to the MSB of D2.

● Description

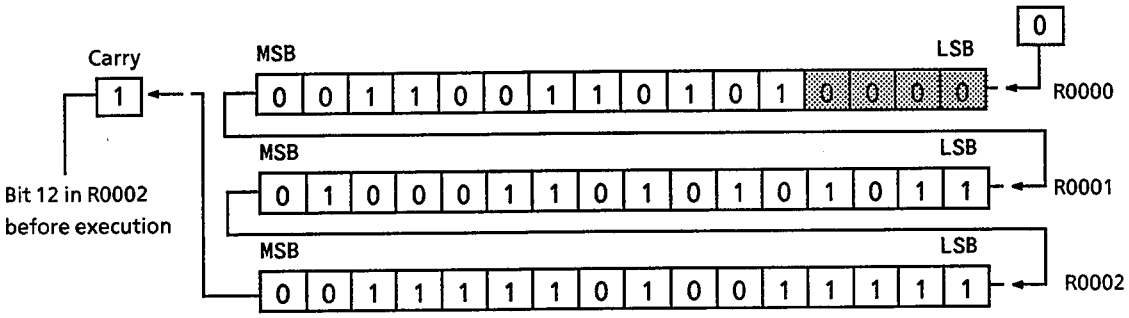
When D1 is in R0000 and D2 in R0002, the data is processed as shown below.



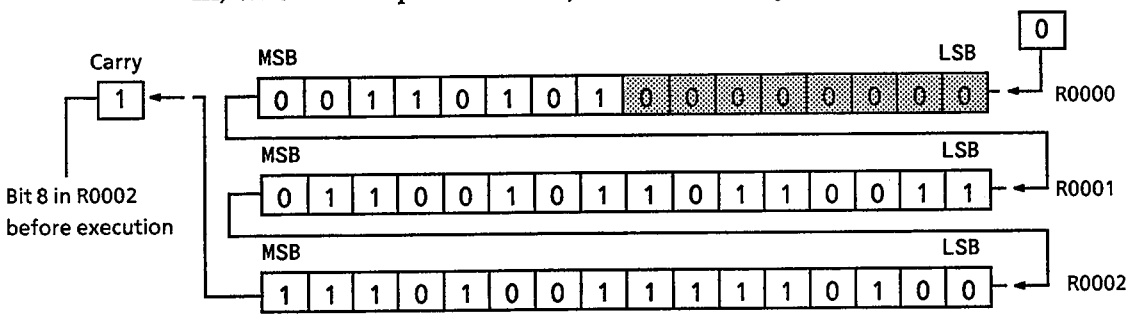
i) When "n" is specified as #1, data is shifted by 1 bit to the left.



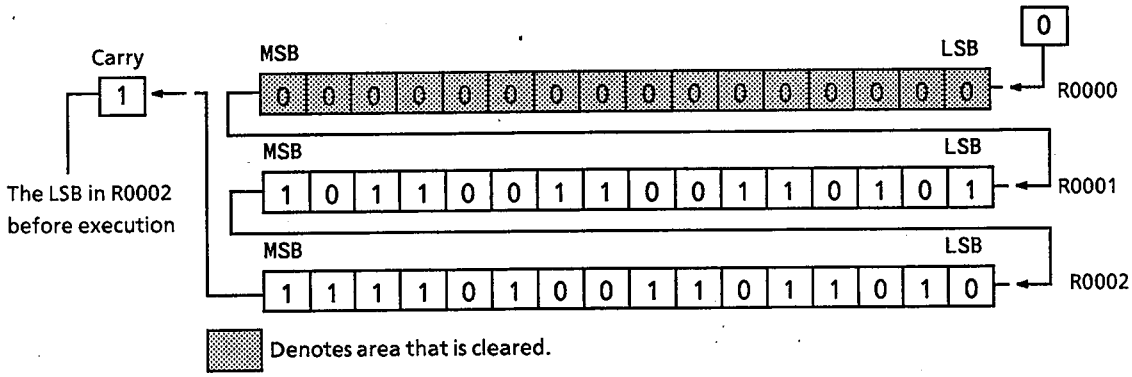
ii) When "n" is specified as #4, data is shifted by 4 bits to the left.



iii) When "n" is specified as #8, data is shifted by 8 bits to the left.



iv) When "n" is specified as #16, data is shifted by 16 bits to the left.



◇Notes◇

- A specification of D1>D2 cannot be made.
- D1 and D2 has to be in the same device area.
  For example, if D1 is in M000 and D2 is in R0010, a program error is generated.

Shift instruction

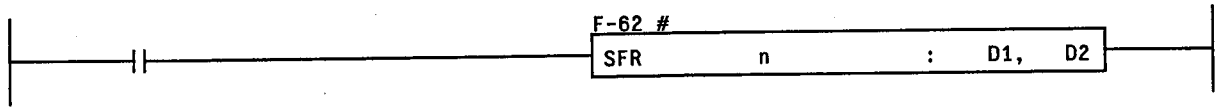
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	○	Underflow M902	-
SFR	FUN 62	Right shift	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-
D1	-	-	Y000 to Y190	Y200 to Y290	R0600 to R0619	L000 to L490	-	-	M000 to M890
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#1、4、8、16	-
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

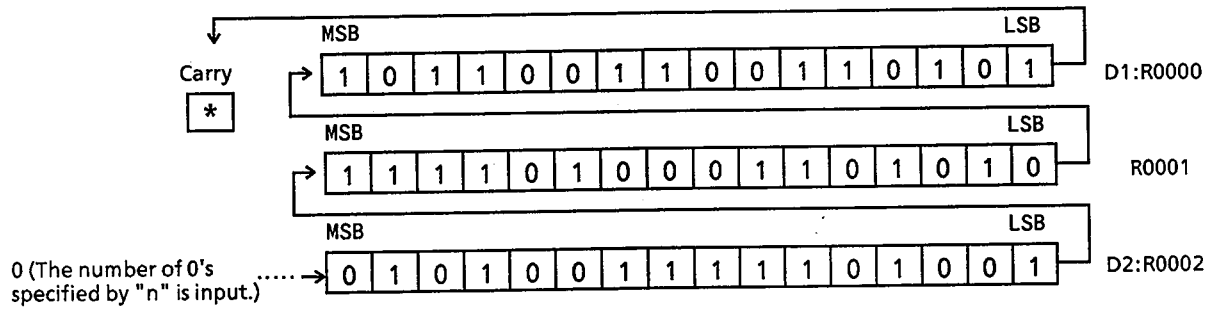


● Function description

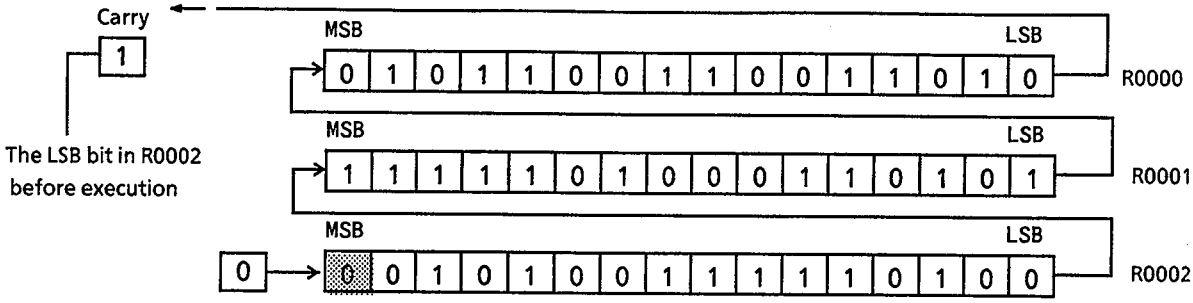
This instruction shifts the number of bits specified by "n" to the right (MSB→LSB) from a word in specified word area D1 to D2.
D1 specifies the first word and D2 the last word (D1≦D2).
When D1 = D2, word data (16 bits) is shifted by "n" bits.
When "n" is 1, data is shifted by 1 bit to the right.
When "n" is 4, data is shifted by 4 bits to the right.
When "n" is 8, data is shifted by 8 bits to the right.
When "n" is 16, data is shifted by 16 bits to the right.
The shift carry (M900) is connected to the LSB of D1.

● Description

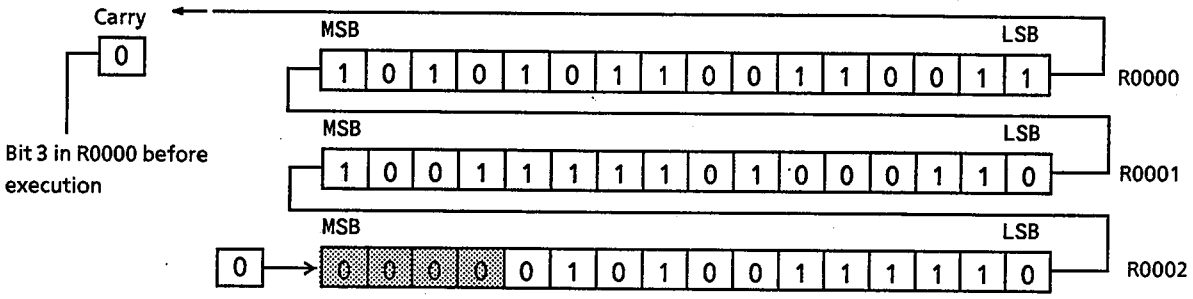
When D1 is in R0000 and D2 in R0002, the data is processed as shown below.



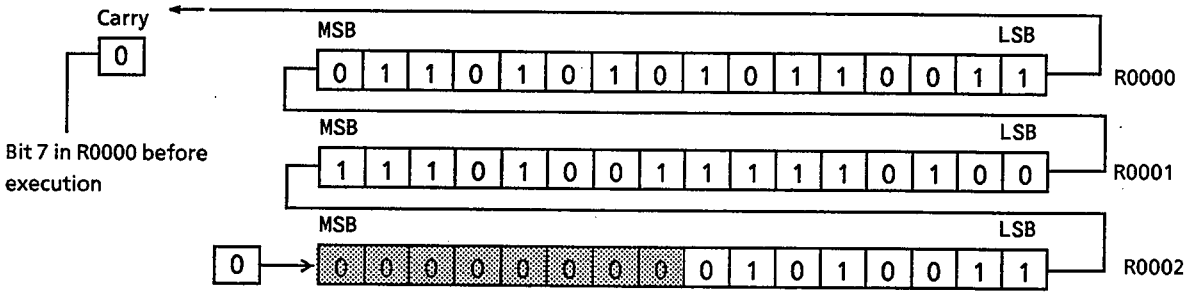
i) When "n" is specified as #1, data is shifted by 1 bit to the right.



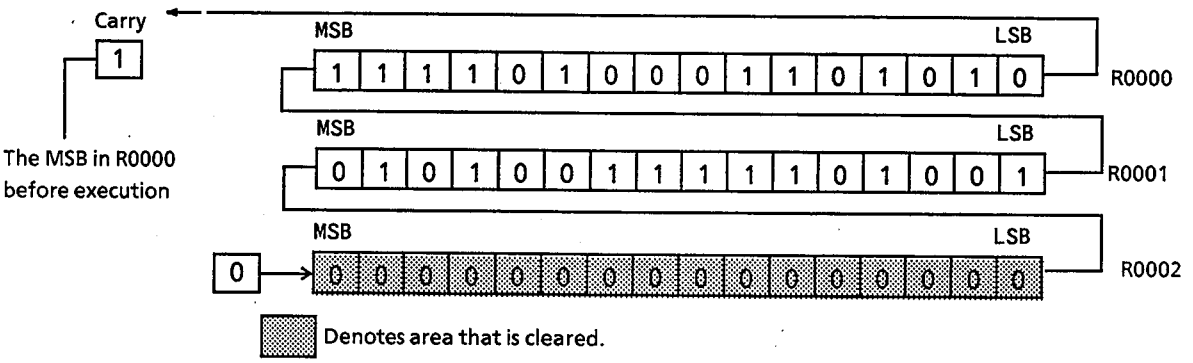
ii) When "n" is specified as #4, data is shifted by 4 bits to the right.



iii) When "n" is specified as #8, data is shifted by 8 bits to the right.



iv) When "n" is specified as #16, data is shifted by 16 bits to the right.



◇Notes◇

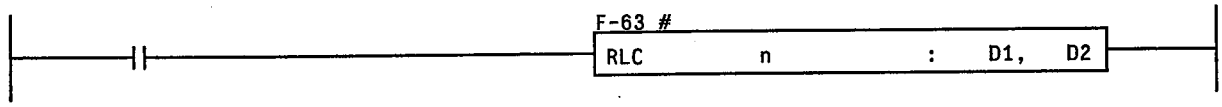
- A specification of D1 > D2 cannot be made.
- D1 and D2 has to be in the same device area.
  For example, if D1 is in M000 and D2 is in R0010, a program error is generated.

Shift instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	○	Underflow M902	-
RLC	FUN 63	Left rotation with carry	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-
D1	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#1、4、8、16	-
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

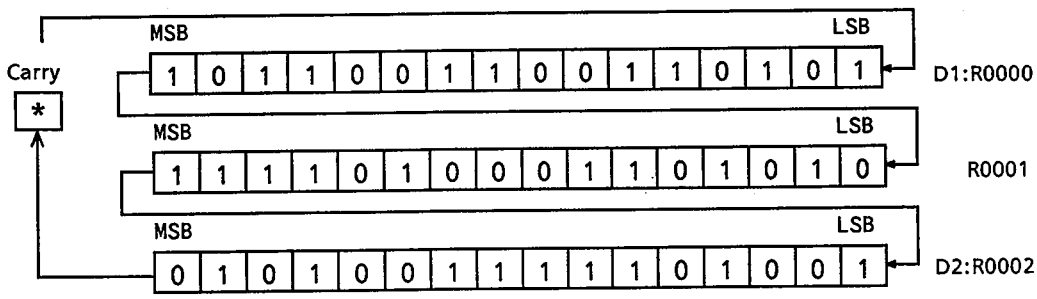


● Function description

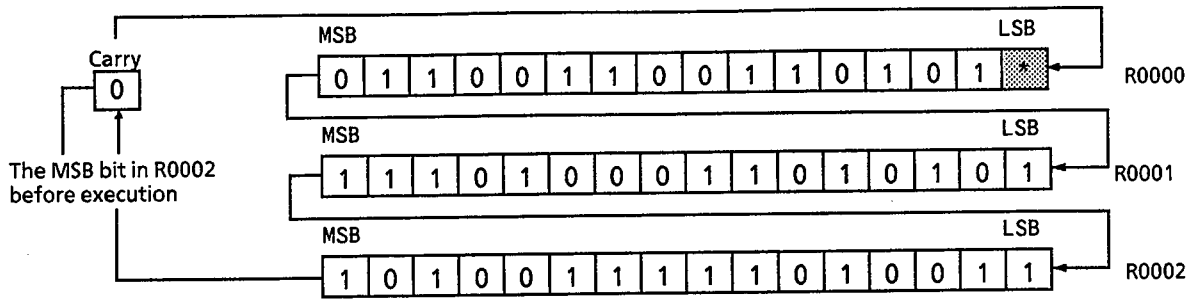
This instruction shifts the number of bits specified by "n" to the left (LSB→MSB) from a word in specified word area D1 to D2 to perform a rotation with carry (M900). The rotation is triggered when the input goes ON.
D1 specifies the first word and D2 the last word (D1≦D2).
When D1 = D, word data (16 bits) is rotated by "n" bits.
When "n" is 1, data is rotated by 1 bit to the left (bit rotation).
When "n" is 4, data is rotated by 4 bits to the left (digit rotation).
When "n" is 8, data is rotated by 8 bits to the left (byte rotation).
When "n" is 16, data is rotated by 16 bits to the left (word rotation).

● Description

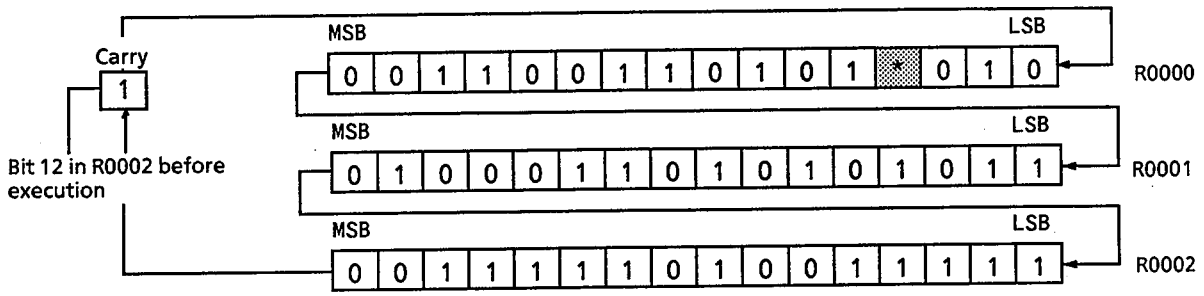
When D1 is in R0000 and D2 in R0002, the data is processed as shown below.



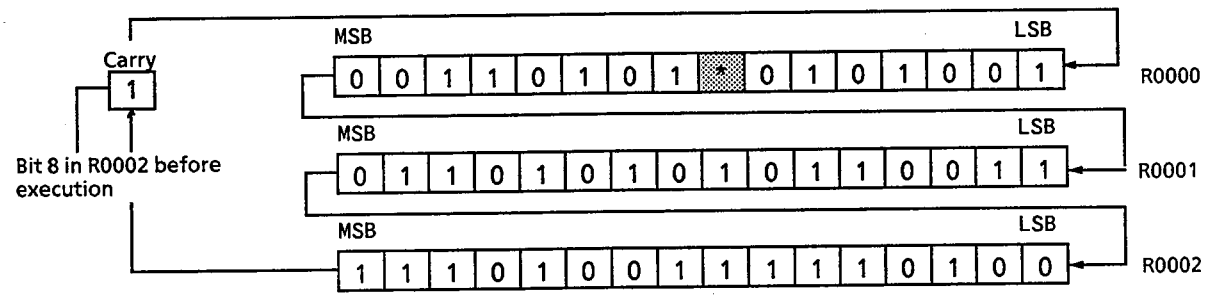
i) When "n" is specified as #1, data is rotated by 1 bit to the left.



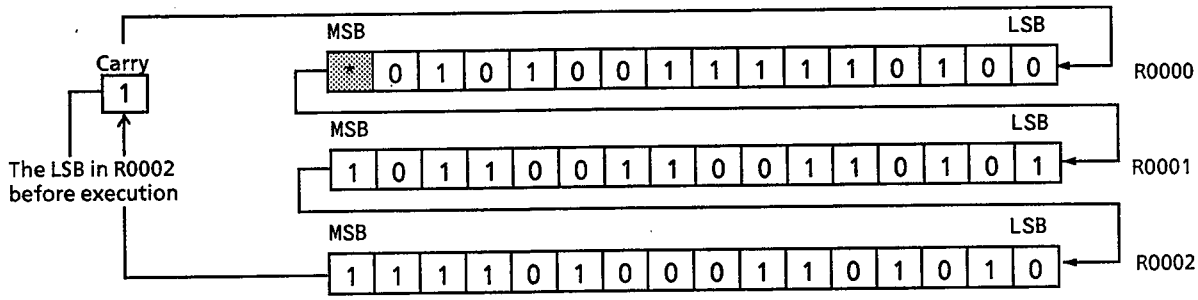
ii) When "n" is specified as #4, data is rotated by 4 bits to the left.



iii) When "n" is specified as #8, data is rotated by 1 byte to the left.



iv) When "n" is specified as #16, data is rotated by 1 word to the left.



Denotes area that is cleared.

◇Notes◇

- A specification of D1 > D2 cannot be made.
- D1 and D2 has to be in the same device area.
  For example, if D1 is in M000 and D2 is in R0010, a program error is generated.

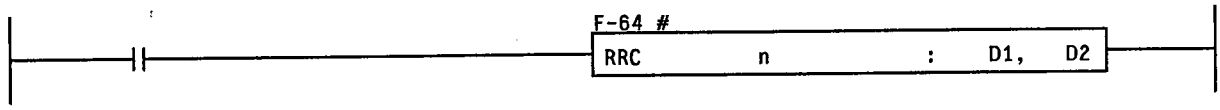
廃番

Shift instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	○	Underflow M902	-
RRC	FUN 64	Right rotation with carry	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-
D1	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#1、4、8、16	-
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

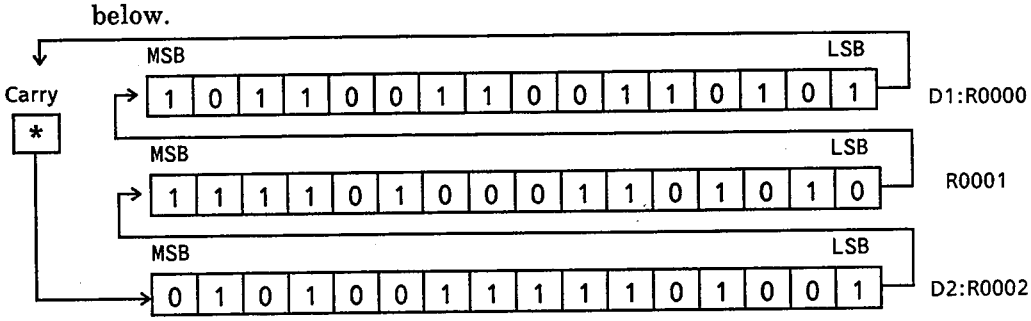


● Function description

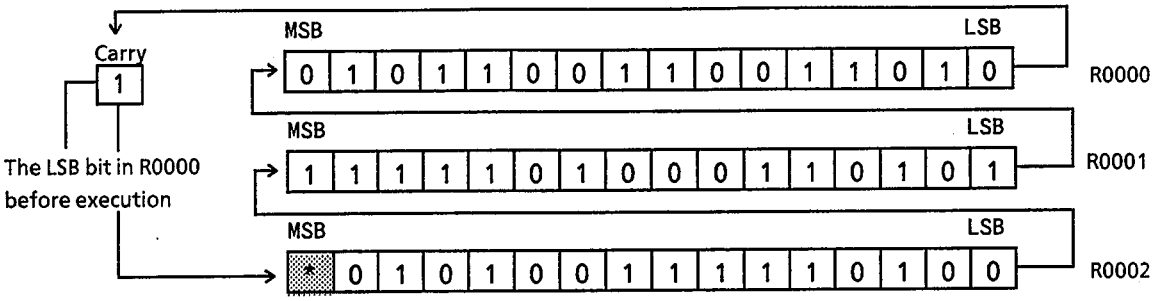
This instruction shifts the number of bits specified by "n" to the right (LSB→MSB) from a word in specified word area D1 to D2 to perform a rotation with carry (M900). The rotation is triggered when the input goes ON.
D1 specifies the first word and D2 the last word (D1≦D2).
When D1 = D, word data (16 bits) is rotated by "n" bits.
When "n" is 1, data is rotated by 1 bit to the right (bit rotation).
When "n" is 4, data is rotated by 4 bits to the right (digit rotation).
When "n" is 8, data is rotated by 8 bits to the right (byte rotation).
When "n" is 16, data is rotated by 16 bits to the right (word rotation).

● Description

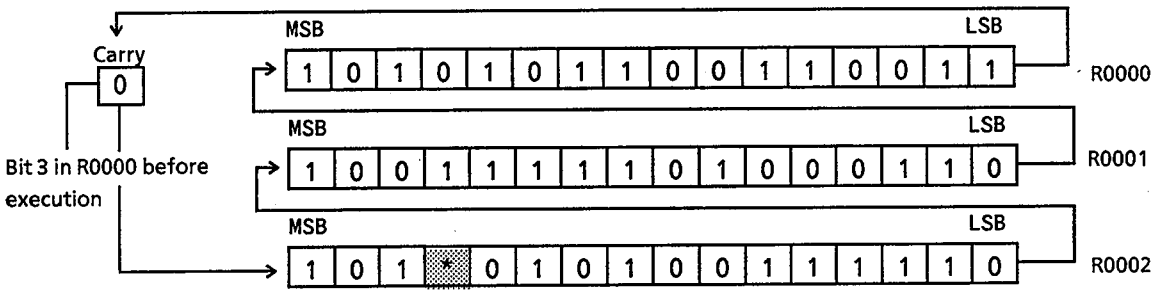
When D1 is in R0000 and D2 in R0002, the data is processed as shown below.



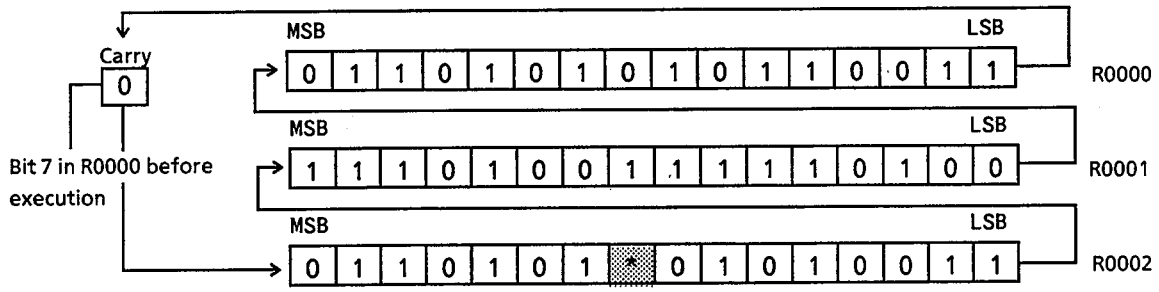
i) When "n" is specified as #1, data is rotated by 1 bit to the right.



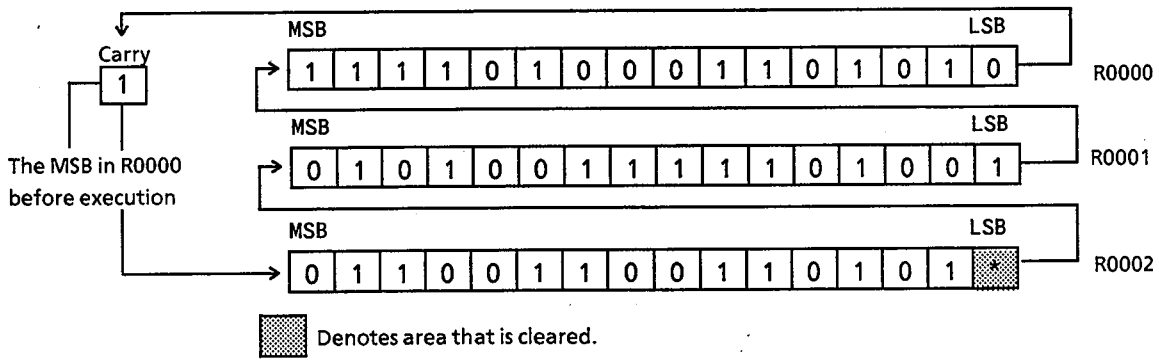
ii) When "n" is specified as #4, data is rotated by 4 bits to the right.



iii)When "n" is specified as #8, data is rotated by 1 byte to the right.



iv) When "n" is specified as #16, data is rotated by 1 word to the right.



◇Notes◇

- A specification of D1>D2 cannot be made.
- D1 and D2 has to be in the same device area.
  For example, if D1 is in M000 and D2 is in R0010, a program error is generated.

Shift instruction

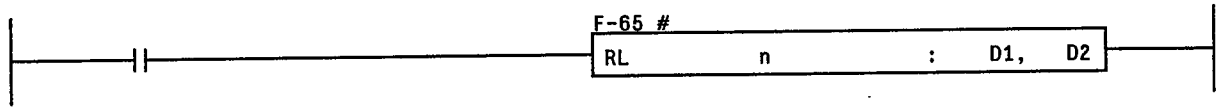
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	○	Underflow M902	-
RL	FUN 65	Left rotation	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-
D1	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#1、4、8、16	-
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

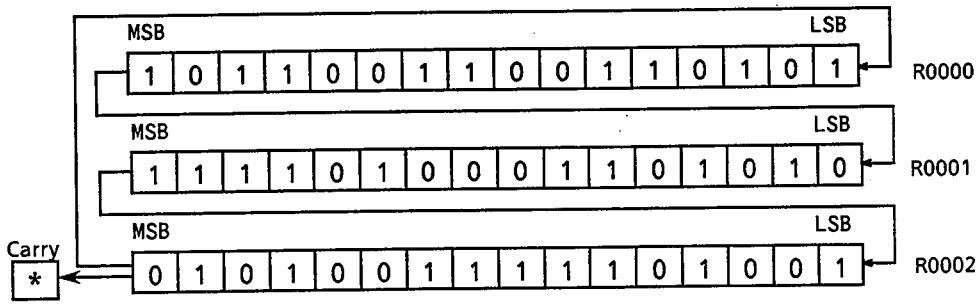


● Function description

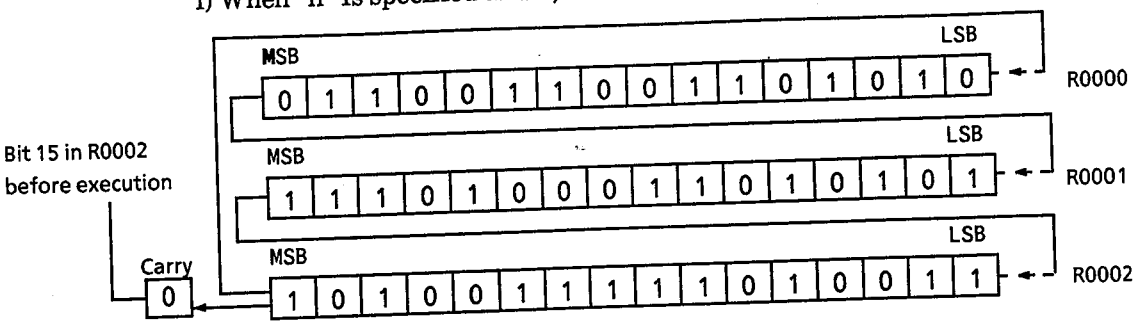
This instruction shifts the number of bits specified by "n" to the left (LSB→MSB) from a word in specified word area D1 to D2 to perform a rotation without carry (M900). The rotation is triggered when the input goes ON. Make a pulse setting when this operation is to be performed for each scan.
D1 specifies the first word and D2 the last word (D1≦D2).
When D1 = D, word data (16 bits) is rotated by "n" bits.
When "n" is 1, data is rotated by 1 bit to the left (bit rotation).
When "n" is 4, data is rotated by 4 bits to the left (digit rotation).
When "n" is 8, data is rotated by 8 bits to the left (byte rotation).
When "n" is 16, data is rotated by 16 bits to the left (word rotation).
A value identical to the LSB of D1 is placed in shift carry.

● Description

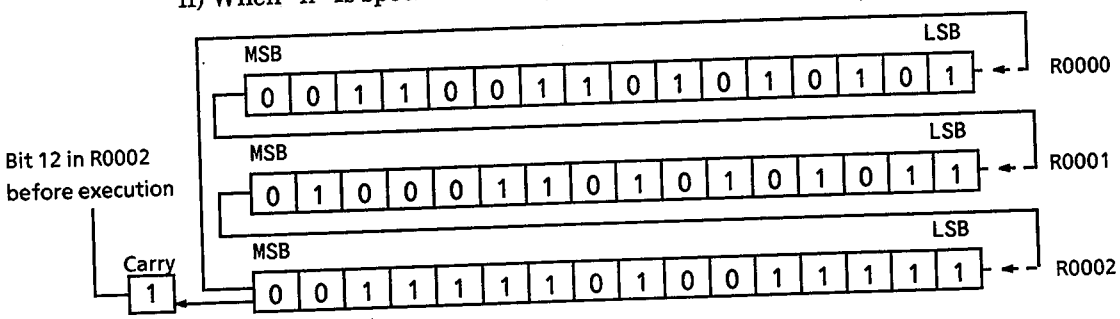
When D1 is in R0000 and D2 in R0002, the data is processed as shown below.



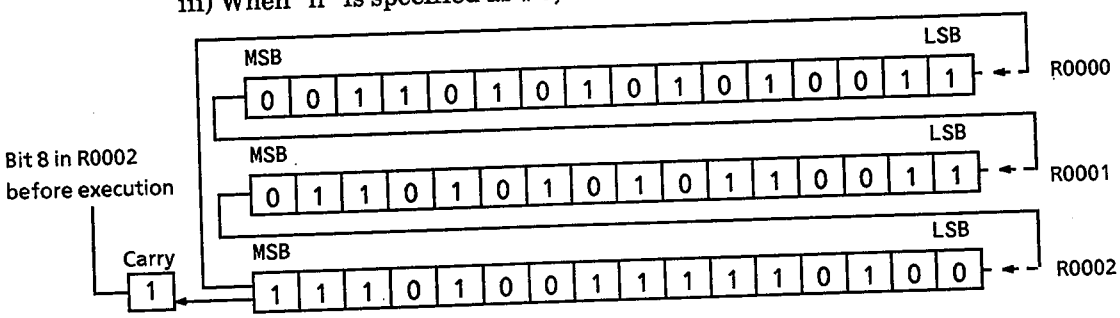
i) When "n" is specified as #1, data is rotated by 1 bit to the left.



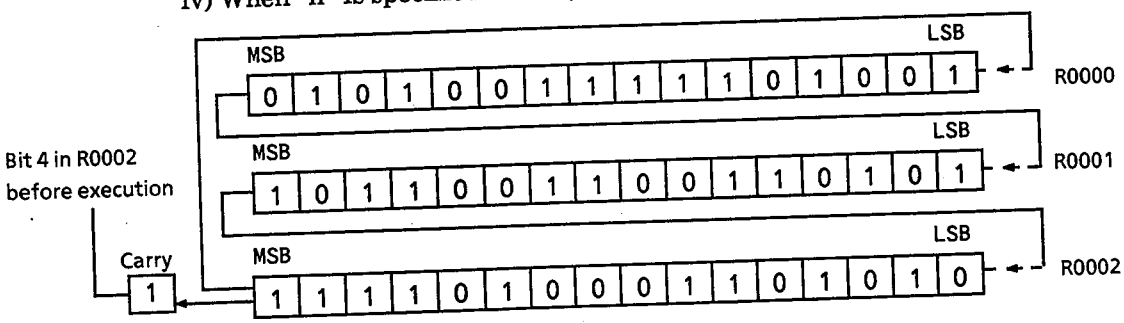
ii) When "n" is specified as #4, data is rotated by 4 bits to the left.



iii) When "n" is specified as #8, data is rotated by 1 byte to the left.



iv) When "n" is specified as #16, data is rotated by 1 word to the left.



◇Notes◇

- A specification of D1>D2 cannot be made.
- D1 and D2 has to be in the same device area.
  For example, if D1 is in M000 and D2 is in R0010, a program error is generated.

Shift instruction

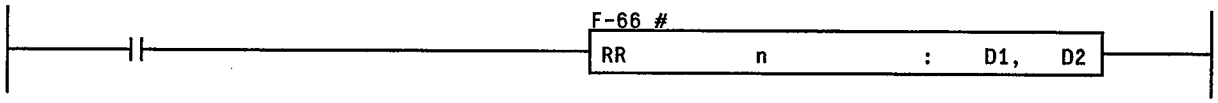
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	○	Underflow M902	-
RR	FUN 66	Right shift	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-
D1	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619
D2	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#1、4、8、16	-
D1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

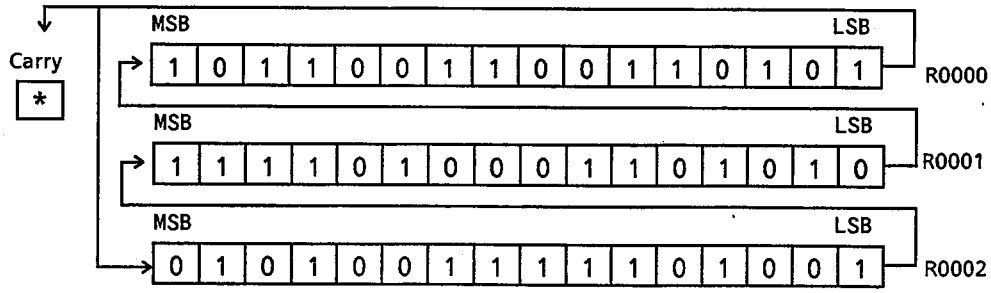


● Function description

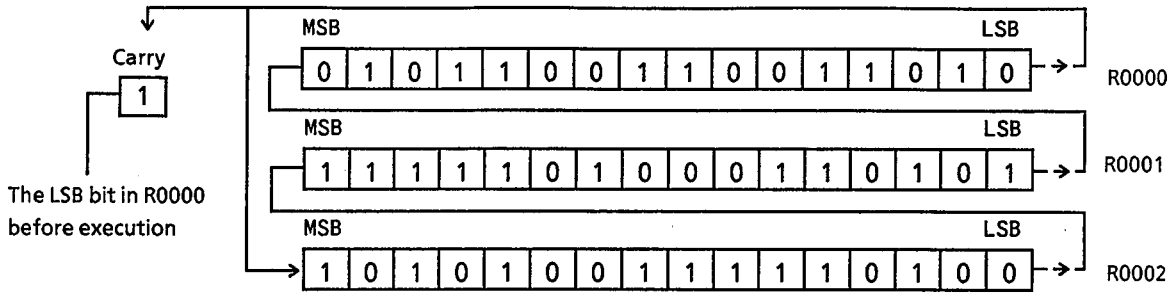
This instruction shifts the number of bits specified by "n" to the right (MSB→LSB) from a word in specified word area D1 to D2 to perform a rotation without carry (M900). The rotation is triggered when the input goes ON.
D1 specifies the first word and D2 the last word (D1≦D2).
When D1 = D, word data (16 bits) is rotated by "n" bits.
When "n" is 1, data is rotated by 1 bit to the right (bit rotation).
When "n" is 4, data is rotated by 4 bits to the right (digit rotation).
When "n" is 8, data is rotated by 8 bits to the right (byte rotation).
When "n" is 16, data is rotated by 16 bits to the right (word rotation).
A value identical to the MSB of D2 is placed in shift carry.

● Description

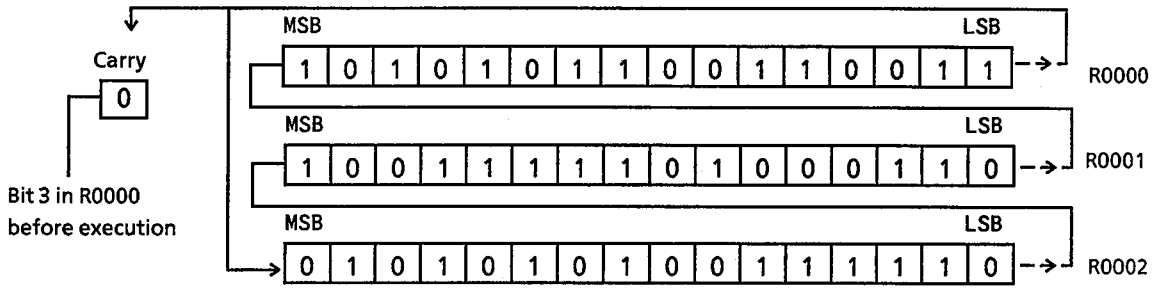
When D1 is in R0000 and D2 in R0002, the data is processed as shown below.



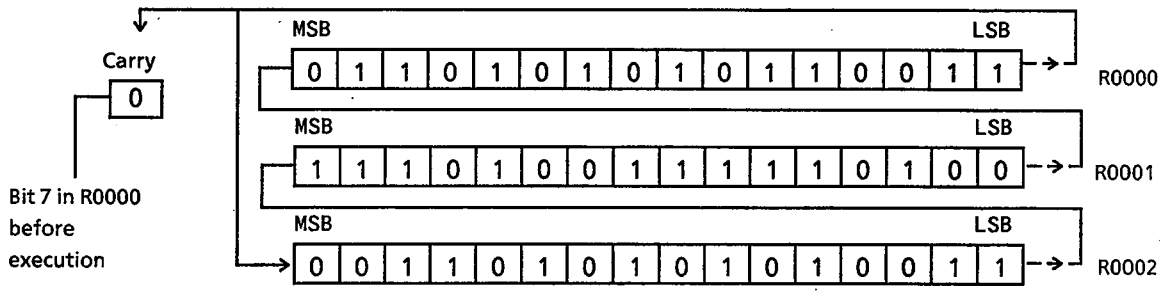
i) When "n" is specified as #1, data is rotated by 1 bit to the right.



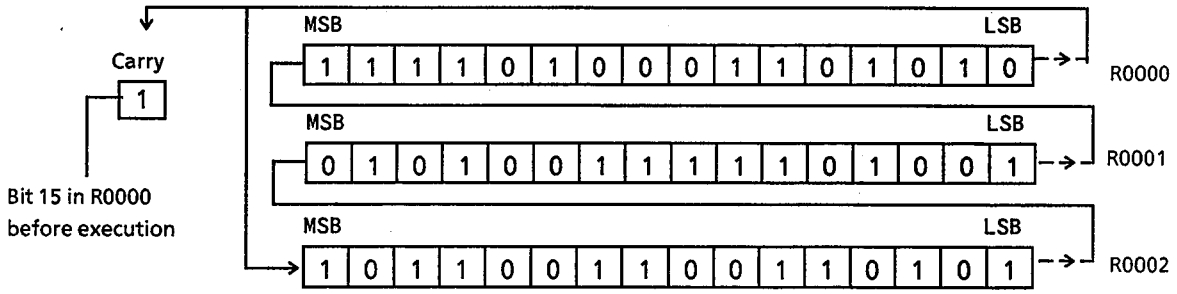
ii) When "n" is specified as #4, data is rotated by 4 bits to the right.



iii) When "n" is specified as #8, data is rotated by 1 byte to the right.



iv) When "n" is specified as #16, data is rotated by 1 word to the right.



◇Notes◇

- A specification of D1>D2 cannot be made.
- D1 and D2 has to be in the same device area.
  For example, if D1 is in M000 and D2 is in R0010, a program error is generated.

廃番

BIN arithmetic instruction

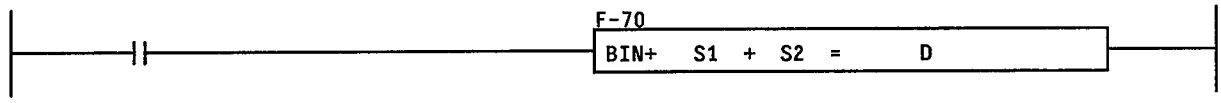
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BIN +	FUN 70	BIN addition	9 bytes		Overflow M901	○	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

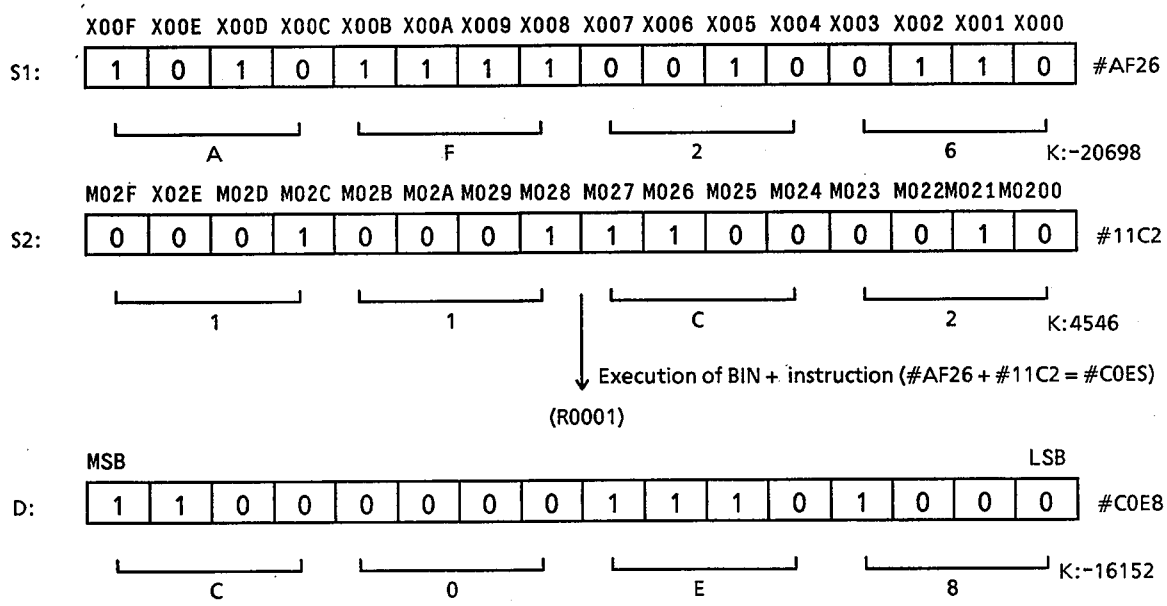


● Function description

This instruction adds binary data words S1 and S2 and passes the result to D. The content of S1 and S2 are not changed by the operation.
S1 contains the summand and S2 the addend.

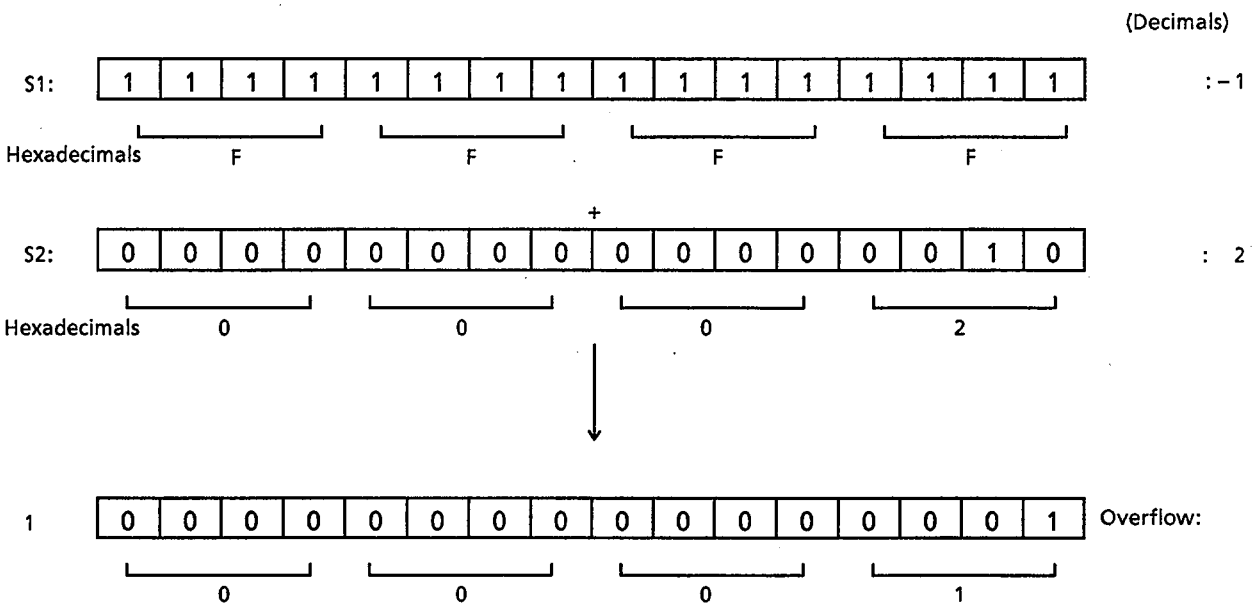
● Description

When S1 starts from X000, S2 from M020 and D from R0001, the data is processed as shown below.



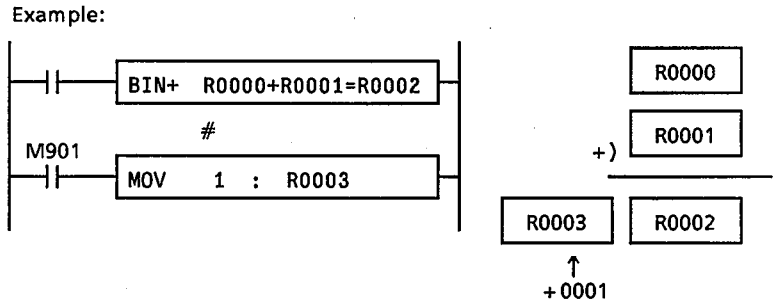
◇Note◇

- When the result of adding S1 and S2 exceeds #10000, the overflow flag (M901) is set.
  Then #FFFF + #0002 = #10001, thus D is #0001 and M901 goes on.



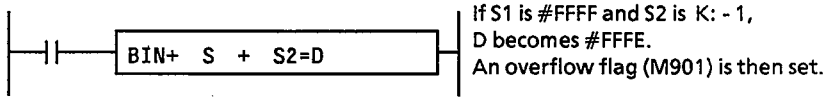
When hexadecimals are added, 0001 is transferred to the high-order bit to correct the result of the operation.

Example:



- Even if there are decimals in S2 in a binary addition, they are handled as binaries in the internal operation.

Example:

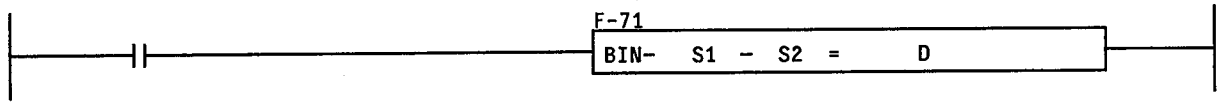


BIN arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	○
BIN –	FUN 71	BIN subtraction	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	ecimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

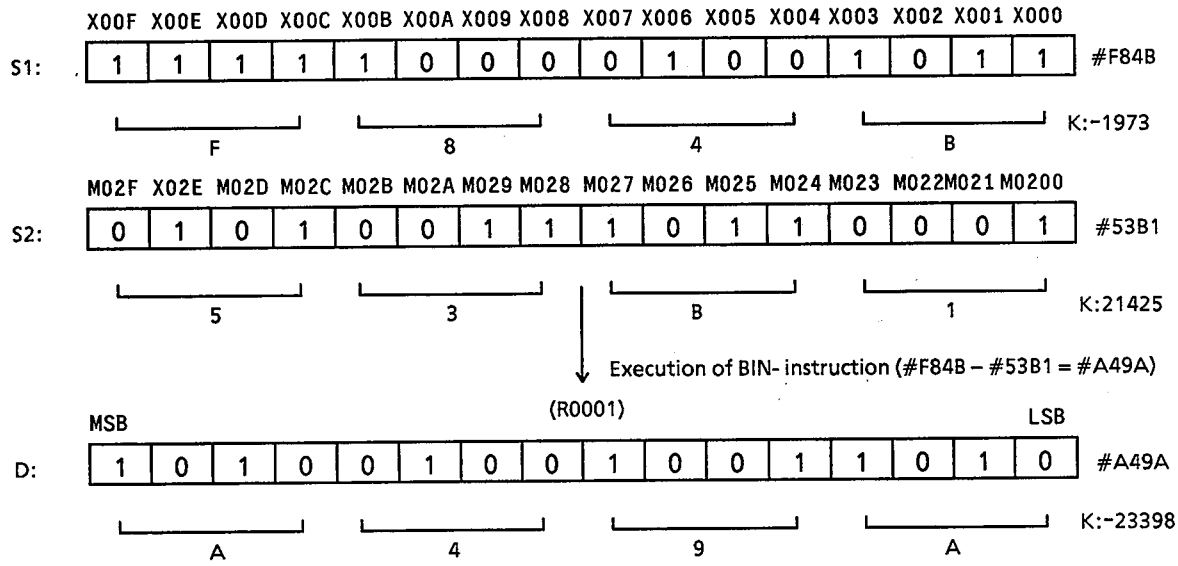


● Function description

This instruction subtracts binary data word S2 from S2 and passes the result to D. The content of S1 and S2 are not changed by the operation.
S1 contains the minuend and S2 the subtrahend.

● Description

When S1 starts from X000, S2 from M020 and D from R0001, the data is processed as shown below.



◇Note◇ A subtraction of S1 and S2 resulting in a borrow sets the underflow flag (M902).
When #1000 - #2000 = -#E000 and when #E000 is placed in D the M902 flag is set.

BIN arithmetic instruction

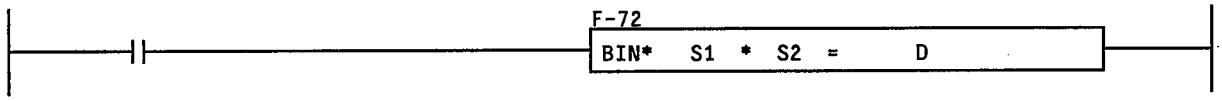
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BIN*	FUN 72	BIN multiplication	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

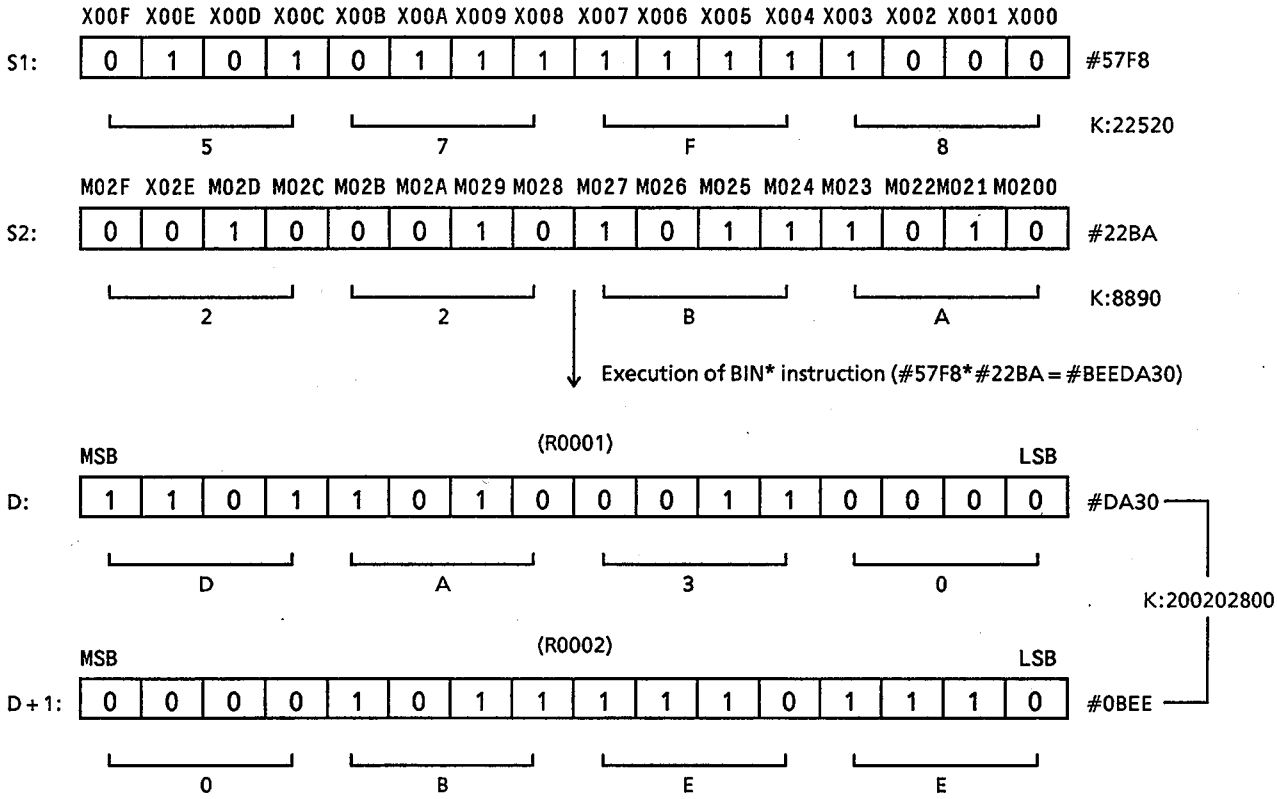


● Function description

This instruction multiplies binary data words S1 and S2 and passes the high-order 16 bits of the result to D + 1 and the low-order 16 bits to D. The content of S1 and S2 are not changed by the operation.
S1 contains the multiplicand and S2 the multiplier data.

● Description

When S1 starts from X000, S2 from M020 and D from R0001, the data is processed as shown below.



When decimal constant K is used and S1 in the example above is K22520, the same operation is performed.
When decimal constant K is used and S1 in the example above is K256, which is #100, the following operation is performed.
K256*#22BA=#100*#22BA=#22DCBA

◇Notes◇

- The most significant bit (B15) of multiplicand S1, multiplier S2 and destination D determines whether the numeric is positive or negative. When B15 is 0, it is positive and when it is 1, it is negative.
- The table below shows how positive and negative numerics are determined.

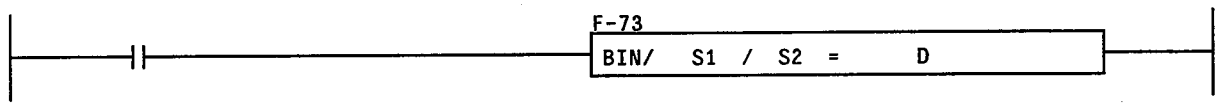
S1	S2	Product
+	+	+
+	−	−
−	+	−
−	−	+

BIN arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BIN/	FUN 73	BIN division	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example

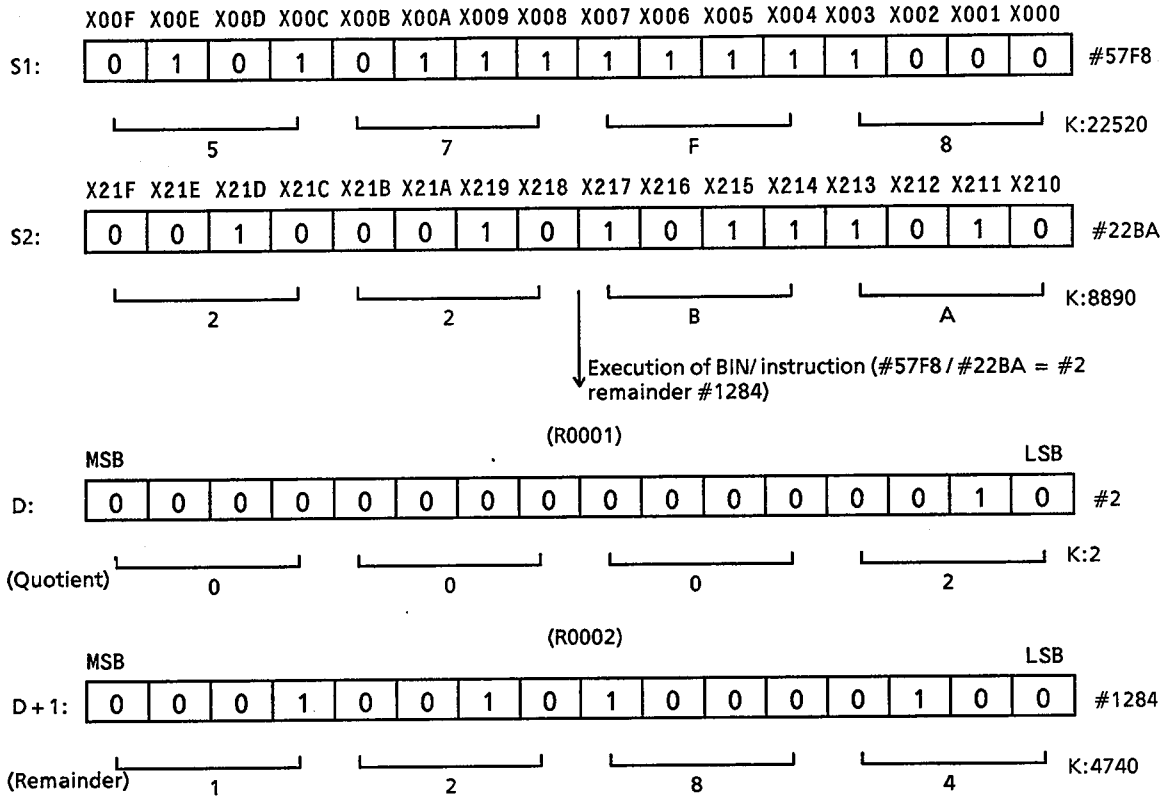


● Function description

This instruction divides binary data word S1 by S2 and passes the quotient to D and the remainder to D + 1.
The content of S1 and S2 is not changed by the operation.
S1 contains the dividend and S2 contains the divisor.

● Description

When S1 starts from X000, S2 from X210 and D from R0001, the data is processed as shown below.



When decimal constant K is used and S1 in the example above is K22520, the same operation is performed.

◇Notes◇

- The most significant bit of dividend S1, divisor S2 and destination D determines whether the numeric is positive or negative.
  When B15 is 0, the numeric is positive and when it is 1, it is negative.
- When the divisor S2 is 0, an operation error (M90E, M90F) is generated and the operation is aborted.
- The quotient and the remainder shall satisfy the formula "S2 x quotient + remainder = S1."
  Example: S1/S2

S1	S2	D	
		Quotient	Remainder
+3	+2	+1	+1
+3	−2	−1	−1
−3	+2	−1	−1
−3	−2	+1	−1

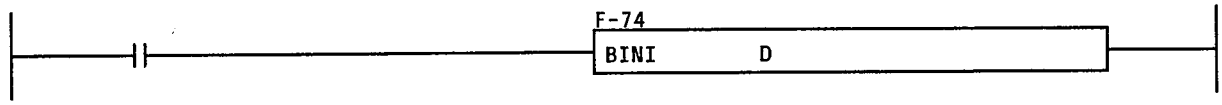
BIN arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BINI	FUN 74	BIN increment	5 bytes		Overflow M901	○	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	

Ladder circuit example

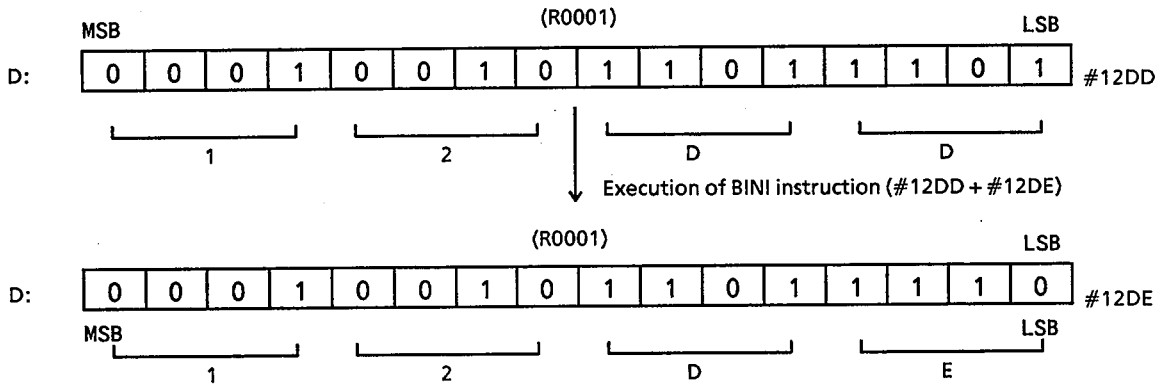


● Function description

This instruction increments (+1) word data D.

● Description

When D starts from R0001, the data is processed as shown below.



◇Note◇ When the result of an operation is #10000, the overflow flag (M901) is set and the result becomes #0000.
Since #FFFF + #1 = #10000, the result of #0000 placed in D causes M901 to go on.

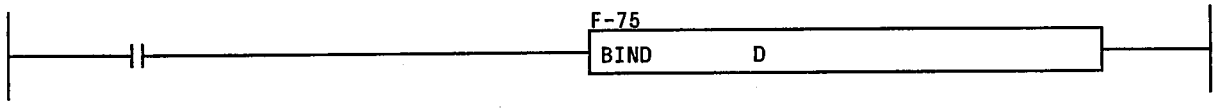
廃番

BIN arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	○
BIND	FUN 75	BIN decrement	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

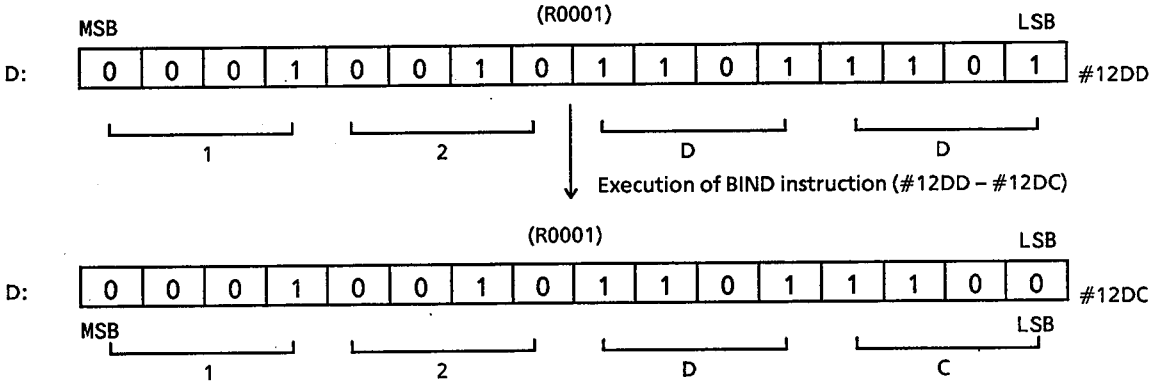


● Function description

This instruction decrements (-1) word data D.

● Description

When D starts from R0001, the data is processed as shown below.



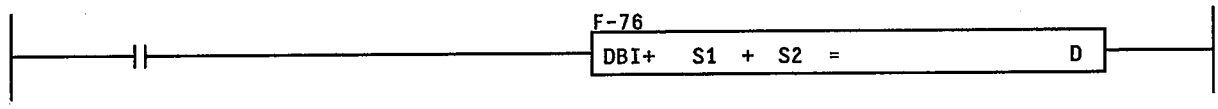
◇Note◇ When the result of an operation causes a borrow, the underflow flag (M902) is set and the result becomes #FFFF.
Since #0000 - #1 = #FFFF, the result of #FFFF placed in D causes M902 to go on.

BIN arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DBI+	FUN 76	Double-word BIN addition	13 bytes		Overflow M901	○	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #FFFFFFFF	K-2147483648 to K2147483647
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #FFFFFFFF	K-2147483648 to K2147483647
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example



● Function description

This instruction adds 8-digit BIN data in two words consisting of S1 + 1 (high-order) and S1 (low-order) to two words consisting of S2 + 1 (high-order) and S2 (low-order). The resulting 32-bit data string is passed to D+1 (high-order) and D (low-order). The content of S1 + 1, S1, S2 + 1 and S2 is not affected by the operation.
S1 + 1 and S1 contain the summand and S2 + 1 and S2 contain the addend.

● Description

This operation is identical to a BIN+ (FUN70) operation.

◇Note◇

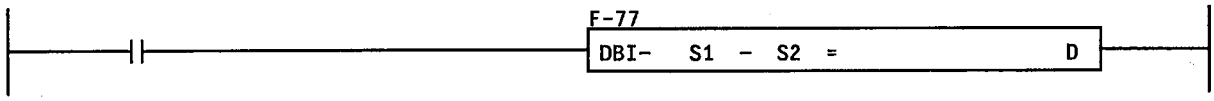
When the result of an addition of (S1 + 1, S1) and (S2 + 1, S2) exceeds #100000000, the overflow flag (M901) is set.
Since #FFFFFFFF + #00000002 = #100000001, the result of #00000001 placed in D + 1 causes M901 to go on.

BIN arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	○
DBI-	FUN 77	Double-word BIN subtraction	13 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link output Y	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	-	R0600 to R0618

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #FFFFFFFF	K-2147483648 to K2147483647
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #FFFFFFFF	K-21474873648 to K2147483647
D	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	-	-

Ladder circuit example



● Function description

A binary subtraction operation is performed on 2-word binary data S1+1(high-order), S1(low-order) and S2+1(high-order), S2(low-order) and the result is transferred to D+1(high-order), D(low-order). The content of S1+1, S1 and S2+1, S2 are not changed by the operation.
S1+1, S1 is the minuend and S2+1, S1 is the subtrahend.

● Description

This operation is identical to a BIN- (FUN71) operation.

◇Note◇ When the result of subtracting S2 + 1, S2 from S1 + 1, S1 causes a borrow, the underflow flag (M902) is set.
Since #10000000 - #20000000 = #E0000000, the result of #E0000000 placed in D + D, D causes M902 to go on.

BIN arithmetic instruction

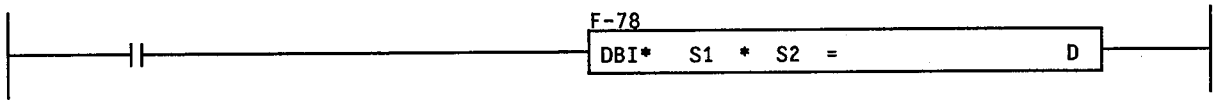
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DBI*	FUN 78	Double-word BIN	13 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y160	Y200 to Y260	M000 to M860	L000 to L460	-	-	R0600 to R0616

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #FFFFFFFF	K-2147483648 to K2147483647
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #FFFFFFFF	K-2147483648 to K2147483647
D	-	R0000 to R0496 R1000 to R4996	P0000 to P3796	-	-	-

Ladder circuit example



● Function description

This instruction multiplies binary data in two words consisting of S1 + 1 (high-order) and S1 (low-order) by two words consisting of S2 + 1 (high-order) and S2 (low-order) . The 4-word result is passed to D + 3 (high-order), D + 2, D + 1 and D (low-order).
The content of S1 + 1, S1, S2 and S2 + 1 is not affected by the operation.
S1 + 1 and S1 contain the multiplicand and S2 + 1 and S2 the multiplier.

● Description

This operation is identical to a BIN* (FUN72) operation.

◇Note◇

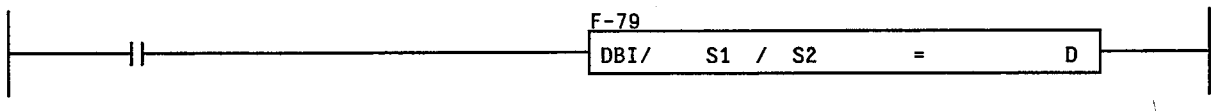
The most significant bit in S1 + 1 and S1, the multiplicand, S2 + 1 and S2, the multiplier, and in D + 3, D + 2, D + 1 and D, the result, indicate whether the numeric is positive or negative. A 0 indicates positive and a 1 indicates negative.

BIN arithmetic instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DBI/	FUN 79	Double-word BIN division	13 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link output Y	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
S2	X000 to X180	X200 to X280	Y000 to Y180	Y200 to Y280	M000 to M880	L000 to L480	-	R0500 to R0518	R0600 to R0618
D	-	-	Y000 to Y160	Y200 to Y260	M000 to M860	L000 to L460	-	-	R0600 to R0616

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #FFFFFFFF	K-2147483648 to K2147483647
S2	-	R0000 to R0498 R1000 to R4998	P0000 to P3798	-	#00000000 to #FFFFFFFF	K-2147483648 to K2147483647
D	-	R0000 to R0496 R1000 to R4996	P0000 to P3796	-	-	-

Ladder circuit example



● Function description

This instruction divides binary data in two words consisting of S1 + 1 (high-order) and S1 (low-order) with two words consisting of S2 + 1 (high-order) and S2 (low-order) . The quotient is passed to D + 1 (high-order) and D (low-order) and the remainder is passed to D + 3 (high-order) and D + 2(low-order).
The content of S1 + 1, S1, S2 and S2 + 1 is not affected by the operation.
S1 + 1 and S1 contain the dividend and S2 + 1 and S2 the divisor.

● Description

This operation is identical to a BIN/ (FUN73) operation.

◇Notes◇

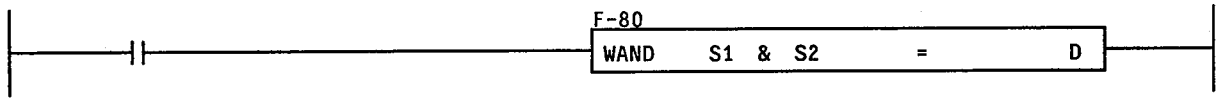
- The most significant bit in S1 + 1 and S1, the dividend, S2 + 1 and S2, the divisor, and in D + 1 and D, the result, indicates whether the numeric is positive or negative. A 0 indicates positive and a 1 indicates negative.
- When the divisor S2 is 0, an operation error (M90E, M90F) is generated and the operation is aborted.

Logical operation instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WAND	FUN 80	And	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



● Function description

This instruction performs an AND operation on word data S1 and S2. The result is passed to D. The content of S1 and S2 is not changed by the operation.

● Description

When S1 starts at X000, S2 at M020 and D is stored in R0001, the data is processed as shown below.

	X00F	X00E	X00D	X00C	X00B	X00A	X009	X008	X007	X006	X005	X004	X003	X002	X001	X000
S1:	0	1	1	1	0	0	1	0	0	1	1	0	0	0	0	0

	M02F	M02E	M02D	M02C	M02B	M02A	M029	M028	M027	M026	M025	M024	M023	M022	M021	M020
S2:	0	0	0	0	1	1	1	1	1	1	0	1	1	0	1	0

Execution of WAND instruction

	MSB						(R0001)									LSB
D:	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0

廃番

Logical operation instruction

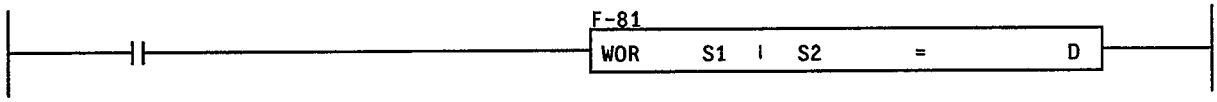
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WOR	FUN 81	Or	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

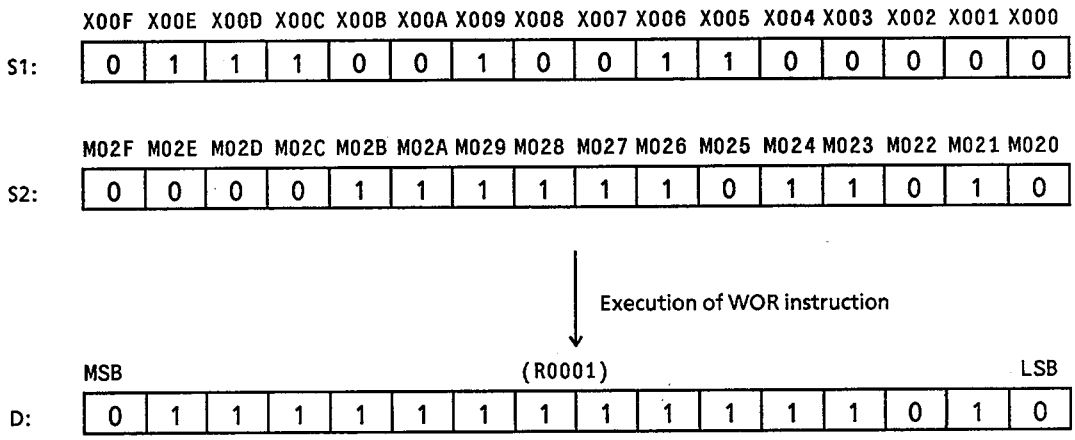


● Function description

This instruction performs an OR operation on word data S1 and S2. The result is passed to D. The content of S1 and S2 is not changed by the operation.

● Description

When S1 starts at X000, S2 at M020 and D is stored in R0001, the data is processed as shown below.



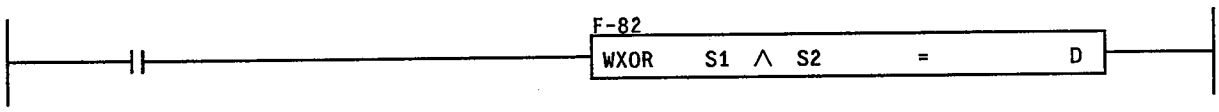
Logical operation instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WXOR	FUN 82	Exclusive or	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

Pulse specification

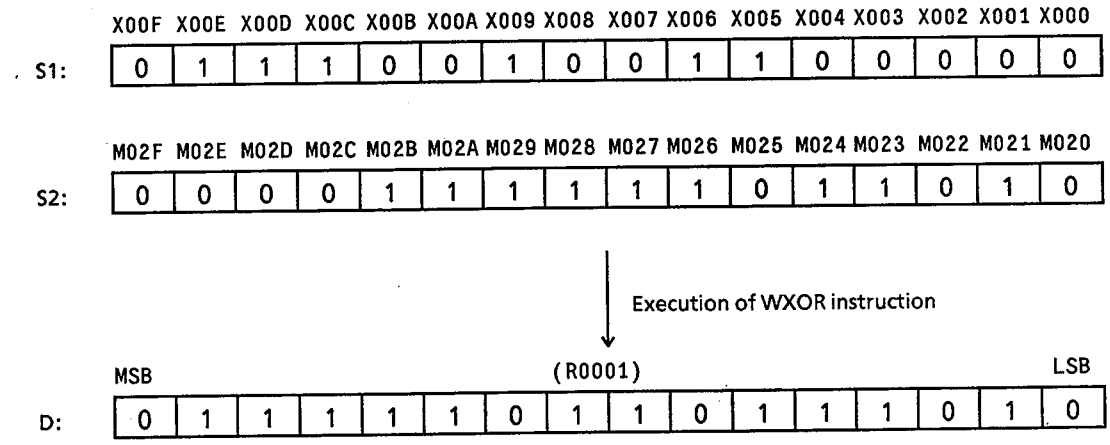


● Function description

This instruction performs an EXCLUSIVE OR operation on word data S1 and S2. The result is passed to D. The content of S1 and S2 is not changed by the operation.

● Description

When S1 starts at X000, S2 at M020 and D is stored in R0001, the data is processed as shown below.

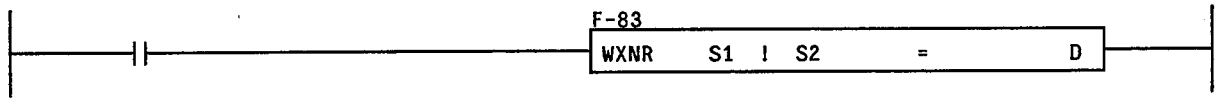


Logical operation instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WXNR	FUN 83	Exclusive nor	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



● Function description

This instruction performs an EXCLUSIVE NOR operation on word data S1 and S2. The result is passed to D. The content of S1 and S2 is not changed by the operation.

● Description

When S1 starts at X000, S2 at M020 and D is stored in R0001, the data is processed as shown below.

	X00F	X00E	X00D	X00C	X00B	X00A	X009	X008	X007	X006	X005	X004	X003	X002	X001	X000
S1:	0	1	1	1	0	0	1	0	0	1	1	0	0	0	0	0

	M02F	M02E	M02D	M02C	M02B	M02A	M029	M028	M027	M026	M025	M024	M023	M022	M021	M020
S2:	0	0	0	0	1	1	1	1	1	1	0	1	1	0	1	0

Execution of WXNR instruction

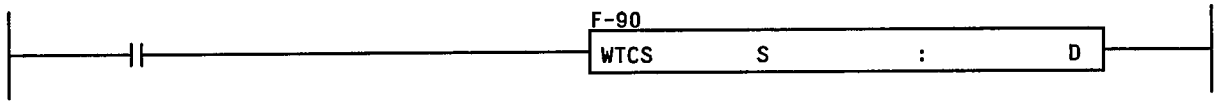
	MSB						(R0001)									LSB
D:	1	0	0	0	0	0	1	0	0	1	0	0	0	1	0	1

Special instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WTCS	FUN 90	Timer/counter setting write	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
D	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example



● Function description

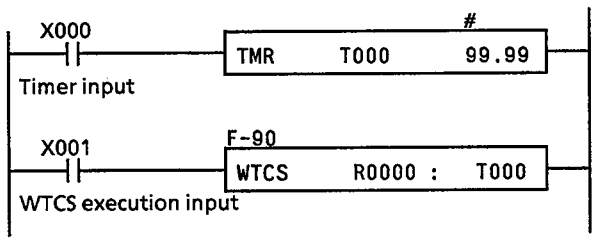
This is a special instruction that writes 4-digit BCD data S in the timer/counter setting D. This setting can be changed with a user program.

● Description

Like the MOV (FUN00) instruction, this is a data transfer instruction that writes the timer/counter setting at D.

◇Notes◇

- If the written data is not a BCD value, an error is generated causing the error flags (M90E, M90F) to be set and the operation to be aborted.
  The T/C setting is rewritten with this instruction and the new value is valid until reset or recompiled.
  (See the following example.)
  This example assumes that S is in R0000 and D in T000.



When R0000 = #9000, X001 goes on and #9000 is placed in T000 as a setting and the timer instruction performs a "TMR T000 #90.00" operation.

- Like in data changes, when the current value is larger than the setting after execution of the instruction, the timer/counter contact goes on to adjust the current value to the setting.
- Unlike a setting modification, this instruction cannot be changed using a program.

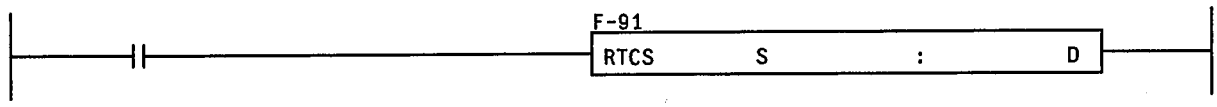
Special instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RTCS	FUN 91	Timer/counter setting read	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	T/C000 to T/C255	-	-	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

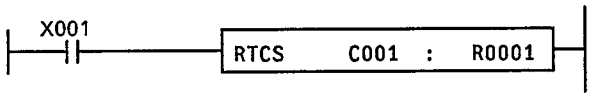


● Function description

This special instruction reads the 4-digit BCD setting S of the timer/counter and stores it in D.

● Description

Like MOV (FUN00), this is a transfer instruction, but can read only the timer/counter D setting.
When S is in C001 and D is in R0001, this instruction reads the counter C001 setting and stores it in R0001 when X001 goes on.

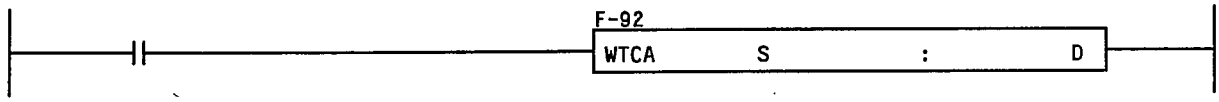


Special instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WTCA	FUN 92	Timer/counter current setting write	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9999	-
D	T/C000 to T/C255	-	-	-	-	-

Ladder circuit example



● Function description

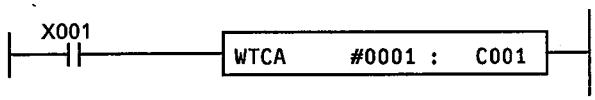
This is a special instruction that writes 4-digit BCD data S in timer/counter setting D. This current setting can be changed with a user program.

● Description

Like the MOV (FUN00) instruction, this is a data transfer instruction that writes the current timer/counter setting at D.

◇Notes◇

- If the written data is not a BCD value, an error is generated causing the error flags (M90E, M90F) to be set and the operation to be aborted.
  When S is in #0000 and D is in C001, this instruction places the current setting #0001 in counter C001.



- When the current setting becomes larger than the set value, the timer/counter goes on and the current setting is adjusted to the set value.

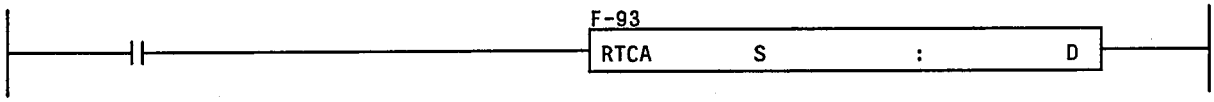
Special instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RTCA	FUN 93	Timer/counter current setting read	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	T/C000 to T/C255	-	-	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

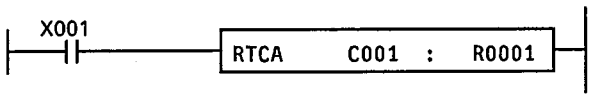


● Function description

This special instruction reads the 4-digit BCD setting S of the timer/counter and stores it in D.

● Description

Like MOV (FUN00), this is a transfer instruction, but can read only the timer/counter D setting.
When S is in C001 and D is in R0001, this instruction reads the counter C001 setting and stores it in R0001 when X001 goes on.

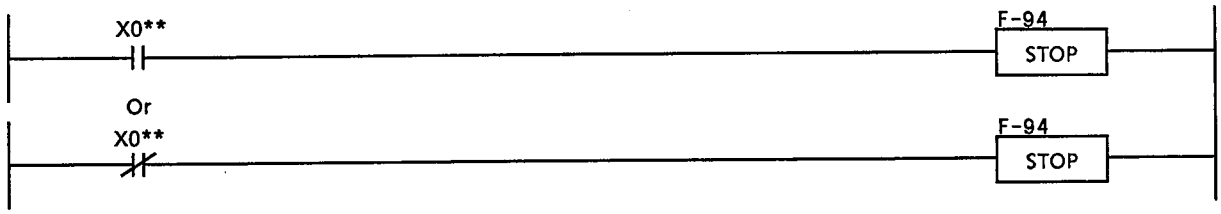


Special instruction			Execution condition	Input ON	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
STOP	FUN 94	Stop	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

Ladder circuit example



● Function description

- This instruction stops sequence program operation and resets the output when the line logic goes on.

● Description

- The stop input uses one input contact (X) as LD X0** or LD NOT X0** (** denotes a device number).
- The sequence program is executed from the start when the stop input is canceled (line logic set to OFF) after the execution of this instruction.

◇Note◇

- This instruction is not the same as the STOP mode.
- Operations differ between the MX100 / MX30 / MX20 and the MX200 / MX50 controllers on the following points:

i) Renewal of representative alarm (M92D / M987) under the STOP command execution.

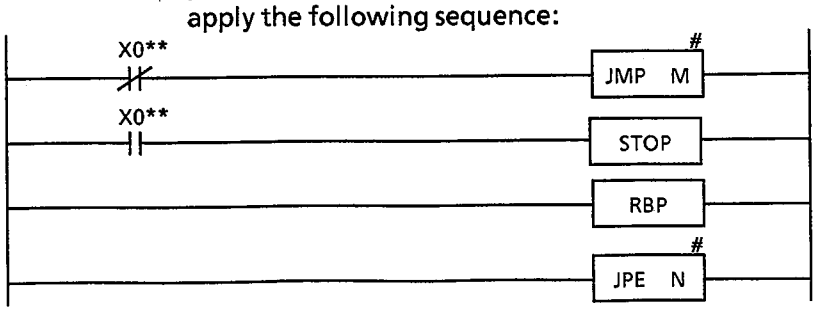
MX100 / MX30 / MX20 ..... Renewal is not performed within the STOP command. For example, the M92D is not made ON, even though the calculation error (M90E, M90F) is set ON before the STOP command execution. Renewal is made after the STOP command is canceled.

MX200 / MX50 . Renewal is made within the STOP command.

ii) Operation after cancellation of stop input (line logic OFF)

MX100 / MX30 / MX20 ..... No command after the STOP command is executed. In the same way as the END / RBP command, this execution is performed from the start of the sequence program.

MX200 / MX50 . The execution starts again from the command immediately after the STOP command. If the same processing as the MX100 / MX30 / MX20 is desired, apply the following sequence:



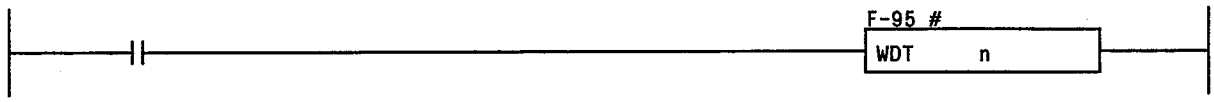
廃番

Special instruction			Execution condition	Input ON	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WDT	FUN 95	Watchdog timer refresh	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#0001 to #0020	-

Ladder circuit example

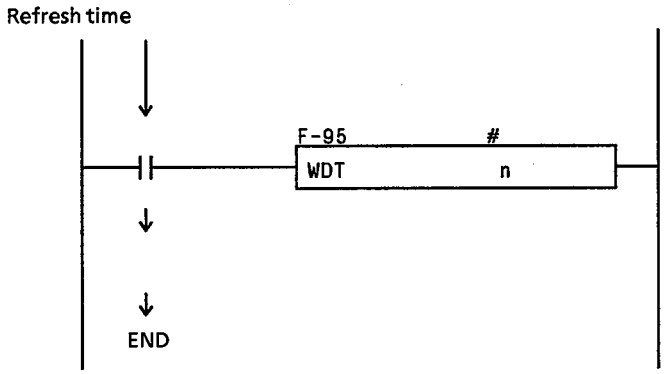


● Function description

This instruction changes the WDT setting in a user program. When the scan time exceeds 100 ms, the WDT instruction can be used to extend the time allotted to the watchdog timer.

● Description

Several WDT instructions can be used.
Tm=100×n(ms) n=1 to 20



◇Notes◇

- The watchdog timer is set to 100 ms when this instruction is not executed.
- The time setting of the watchdog timer is extended only when this instruction is executed, at other times the default setting of 100 ms is used.

Special instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ISS	FUN 96	Input refresh	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

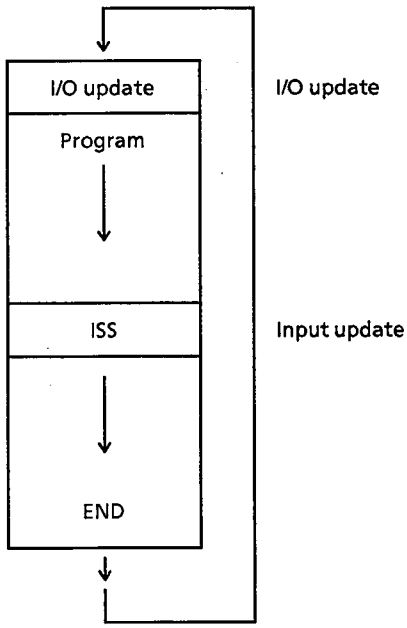
Ladder circuit example

● Function description

This instruction updates input (X000 to X19F) and special register input (R0500 to R0519) to the latest data value.

● Description

This instruction updates input (X) and output (Y) when a program is started and executes it from the beginning. The ISS instruction fetches new input (X) data when the program is running.



◇Note◇ The execution of the ISS instruction takes time and it should be used sparingly to prevent undue lengthening of the scan time.

廃番

Special instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
IOS	FUN 97	Output refresh	2 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
-	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
-	-	-	-	-	-	-

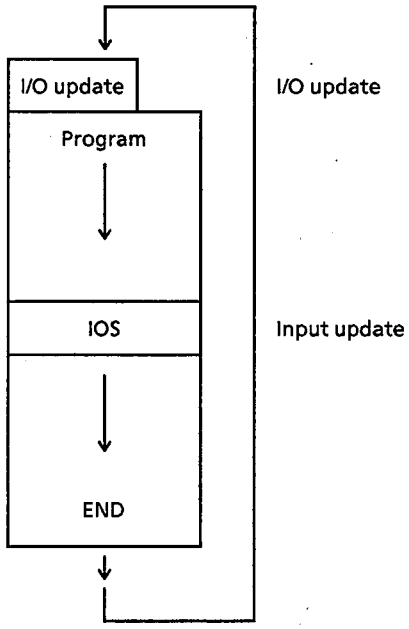
Ladder circuit example

● Function description

This instruction updates input (X000 to X19F), register input (R0500 to R0519), output (Y000 to Y19F) and register output (R0600 to R0619) to the latest data value.

● Description

This instruction updates input (X) and output (Y) when a program is started and executes it from the beginning. The IOS instruction inputs and outputs new input (X) and output (Y) data when the program is running.



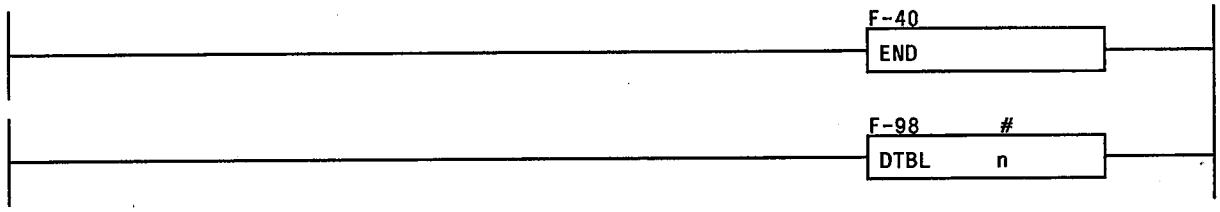
◇Note◇ The execution of the IOS instruction takes time and it should be used sparingly to prevent undue lengthening of the scan time.

Special instruction			Execution condition	Input ON	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
DTBL	FUN 98	Data table setting	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	-	-	-	#0001 to #6000	-

Ladder circuit example



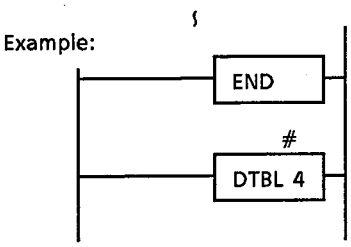
● Function description

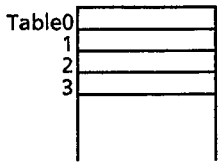
This instruction defines the number of word data items specified by n as data table.
This instruction cannot be used with the MX200 and MX50 controllers.
Data tables have to be defined with the data table management information editing tool in the loader program function for the MX200 and MX50.
The descriptions that follow are for the MX100, MX30 and MX20 controllers.

● Description

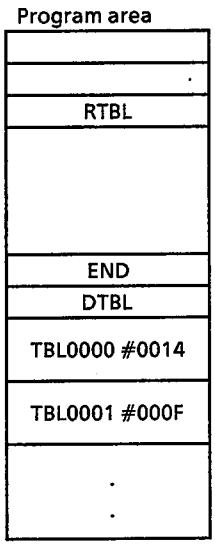
This instruction is used after the END (FUN40) to specify the number of data items to register.
A total of 6000 words can be registered in single word units.

Example:





This instruction is used to register data in data tables. Data entries are made after invoking the instruction.



Program area

Table data is read by the RTBL instruction

Determines the number of items to be registered in the data table

Data write
Valid up to the number set with DTBL

◇Notes◇

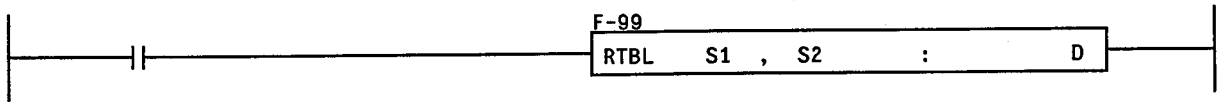
- Data tables are stored in the same area as user programs. Thus the sum of user programs and data tables must not exceed memory capacity. If memory capacity is exceeded, a program error is generated.
- If this instruction is not placed directly after the END instruction, the program check determines that an error (error code: 34) has occurred.
- When several DTBL instructions are used, only the first one is valid.

Special instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RTBL	FUN 99	Data table read	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #9998	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0001 to #4000	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

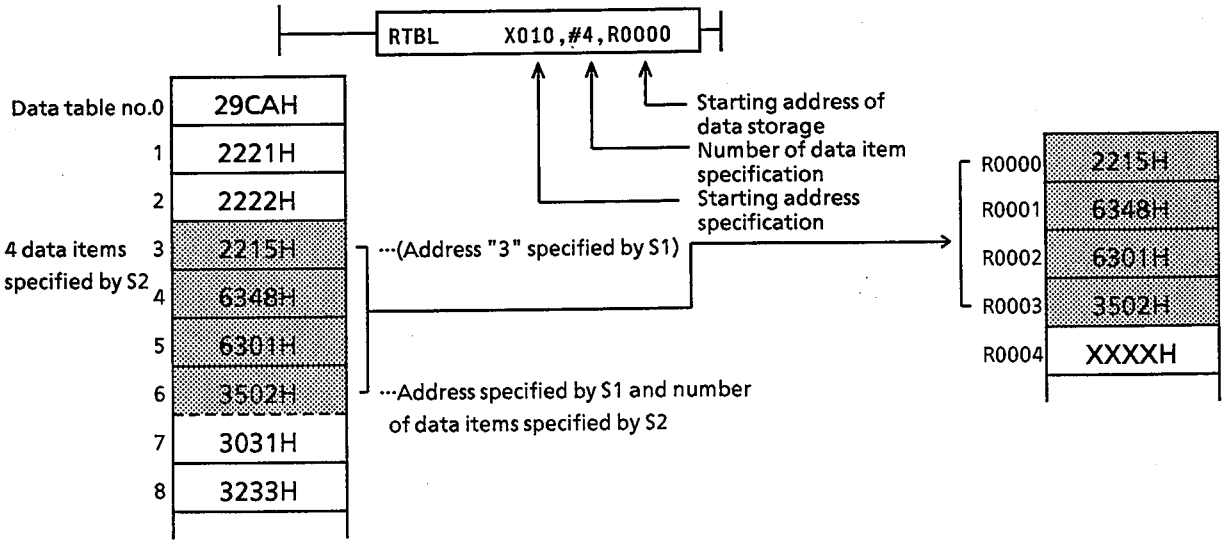


● Function description

This instruction reads the number of data blocks indicated by S2 from the starting address S1 in the data table and stores them from address D in the device table.

● Description

When the starting address specified by S1 is X010 (= 0003), and the number of data items specified by S2 is #4 and the starting address in the device table D is specified as R0000, the data is processed as shown below. S1 and S2 are BCD values that indicate the starting address and the number of data items, respectively.



◇Notes◇

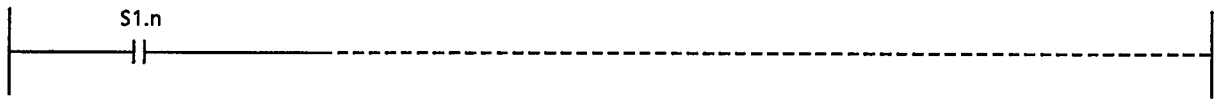
- An operation error (M90E, M90F) is generated in the following conditions and the operation is aborted.
  ① When the number of data items specified by S2 is 0.
  ② When S1 and S2 are not BCD values.
  ③ When the number of data items specified by S2 cannot be stored in area D.
  ④ When the data table specified by S1 and S2 exceeds the data table area.
- Since the ladder program and data tables are independent items in MX200, references to invalid areas are not checked during program creation. (This check is performed when the program is executed.)

Register bit processing instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RLD	FUN 100	Register load	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	R0500 to R0519	R0600 to R0619
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 R3800 to R3999	-	-
n	-	-	-	-	#0000 to #000F	-

Ladder circuit example

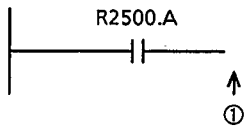


● Function description

This instruction reads the bit data specified by n in the registers (R0000 to R0999, R1000 to R4999, P0000 to P3999) specified by S1 in the line logic to perform logical operations.

● Description

In the following example S1 is in R2500 and n is #A.
Thus when bit A in R2500 is OFF (in the example R2500 = 0B00H), the line logic is off.
When bit A in R2500 is ON (in the example R2500 = 0400H), the line logic is on.



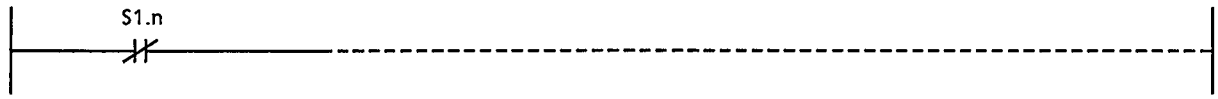
R2500 bit A	Line logic ①
OFF (in the example R2500 = 0B00H)	OFF
ON (in the example R2500 = 0400H)	ON

Register bit processing instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RLDN	FUN 101	Register load not	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	R0500 to R0519	R0600 to R0619
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 R3800 to R3999	-	-
n	-	-	-	-	#0000 to #000F	-

Ladder circuit example

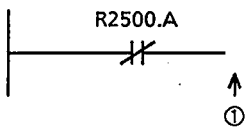


● Function description

This instruction inverts the bit data specified by n in the registers (R0000 to R0999, R1000 to R4999, P0000 to P3999) specified by S1 and reads the line logic to start a logical negation operation.

● Description

In the following example S1 is in R2500 and n is #A.
Thus when bit A in R2500 is OFF (in the example R2500 = 0B00H), the line logic is on.
When bit A in R2500 is ON (in the example R2500 = 0400H), the line logic is off.



R2500 bit A	Line logic ①
OFF (in the example R2500 = 0B00H)	ON
ON (in the example R2500 = 0400H)	OFF

Register bit processing instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RAND	FUN 102	Register and	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	R0500 to R0519	R0600 to R0619
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	-	-
n	-	-	-	-	#0000 to #000F	-

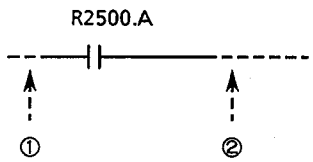
Ladder circuit example

● Function description

This instruction performs an AND operation of the previous line logic and the bit data specified by n in the registers (R0000 to R0999, R1000 to R4999, P0000 to P3999) specified by S1 and stores the result in the line logic.

● Description

The following example assumes that S1 is in R2500 and n is #A.
If the previous line logic is OFF, the line logic remains OFF regardless of S1 and n.
If the previous line logic is ON and bit A in R2500 is OFF (in the example R2500 = 0B00H), the line logic is OFF.
If the previous line logic is ON and bit A in R2500 is ON (in the example R2500 = 0400H), the line logic is ON.



Line logic ①	R2500 bit A	Line logic ②
OFF	OFF (in the example R2500 = 0B00H)	OFF
OFF	ON (in the example R2500 = 0400H)	OFF
ON	OFF (in the example R2500 = 0B00H)	OFF
ON	ON(in the example R2500 = 0400H)	ON

廃番

Register bit processing instruction

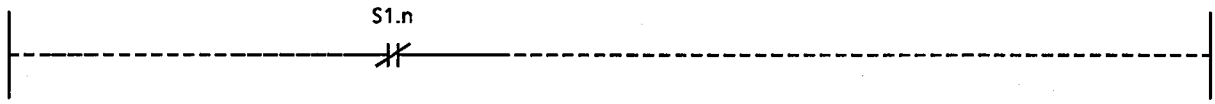
Execution condition	Always executed	Pulse specification	-	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RADN	FUN 103	Register and not	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	R0500 to R0519	R0600 to R0619
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	-	-
n	-	-	-	-	#0000 to #000F	-

Ladder circuit example

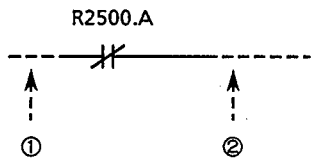


● Function description

This instruction performs an AND operation of the inverted values of the previous line logic and the bit data specified by n in the registers (R0000 to R0999, R1000 to R4999, P0000 to P3999) specified by S1 and stores the result in the line logic.

● Description

The following example assumes that S1 is in R2500 and n is #A.
If the previous line logic is OFF, the line logic remains OFF regardless of S1 and n.
If the previous line logic is ON and bit A in R2500 is OFF (in the example R2500 = 0B00H), the line logic is ON.
If the previous line logic is ON and bit A in R2500 is ON (in the example R2500 = 0400H), the line logic is OFF.



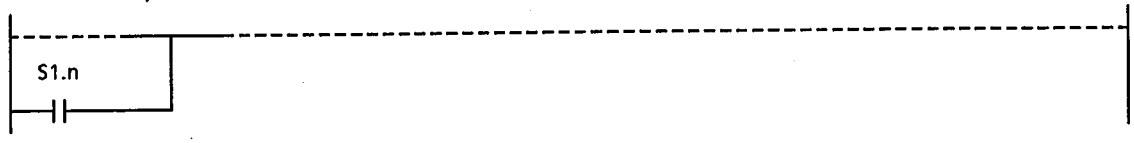
Line logic ①	R2500 bit A	Line logic ②
OFF	OFF (in the example R2500 = 0B00H)	OFF
OFF	ON (in the example R2500 = 0400H)	OFF
ON	OFF (in the example R2500 = 0B00H)	ON
ON	ON (in the example R2500 = 0400H)	OFF

Register bit processing instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ROR	FUN 104	Register or	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	R0500 to R0519	R0600 to R0619
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	-	-
n	-	-	-	-	#0000 to #000F	-

Ladder circuit example

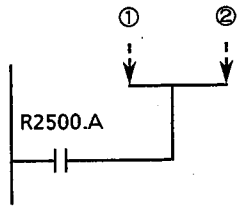


● Function description

This instruction performs an OR operation of the previous line logic and the bit data specified by n in the registers (R0000 to R0999, R1000 to R4999, P0000 to P3999) specified by S1 and stores the result in the line logic.

● Description

The following example assumes that S1 is in R2500, n is #A and that the previous line logic is OFF.If bit A in R2500 is OFF (in the example R2500 = 0B00H), the line logic is OFF. If bit A in R2500 is ON (in the example R2500 = 0400H), the line logic is ON. If the previous line logic is ON, the line logic remains ON regardless of the state of S1 and n.



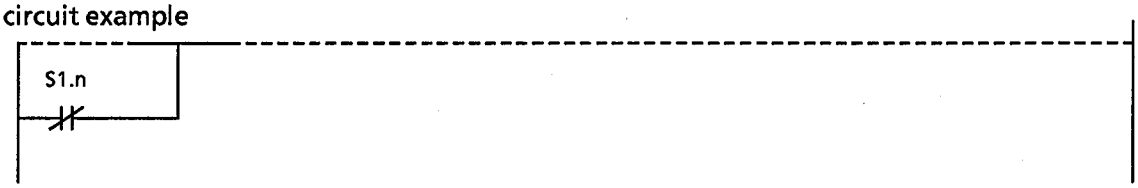
Line logic ①	R2500 bit A	Line logic ②
OFF	OFF (in the example R2500 = 0B00H)	OFF
OFF	ON (in the example R2500 = 0400H)	ON
ON	OFF (in the example R2500 = 0B00H)	ON
ON	ON (in the example R2500 = 0400H)	ON

Register bit processing instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RORN	FUN 105	Register or not	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	R0500 to R0519	R0600 to R0619
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	-	-
n	-	-	-	-	#0000 to #000F	-

Ladder circuit example

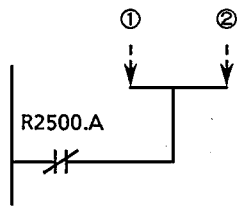


● Function description

This instruction performs an OR operation of the inverted values of the previous line logic and the bit data specified by n in the registers (R0000 to R0999, R1000 to R4999, P0000 to P3999) specified by S1 and stores the result in the line logic.

● Description

The following example assumes that S1 is in R2500, n is #A and that the previous line logic is OFF. If bit A in R2500 is OFF (in the example R2500 = 0B00H), the line logic is ON.If bit A in R2500 is ON (in the example R2500 = 0400H), the line logic is OFF. If the previous line logic is ON, the line logic remains ON regardless of the state of S1 and n.



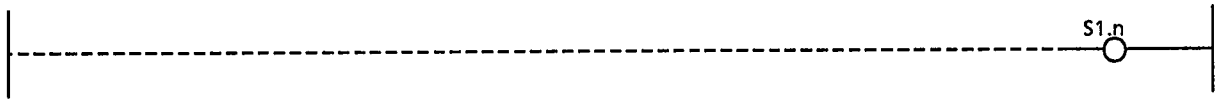
Line logic ①	R2500 bit A	Line logic ②
OFF	OFF (in the example R2500 = 0B00H)	ON
OFF	ON (in the example R2500 = 0400H)	OFF
ON	OFF (in the example R2500 = 0B00H)	ON
ON	ON (in the example R2500 = 0400H)	ON

Register bit processing instruction			Execution condition	Always executed	Pulse specification	-	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ROUT	FUN 108	Register out	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	R0500 to R0519	R0600 to R0619
n	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	-	-
n	-	-	-	-	#0000 to #000F	-

Ladder circuit example

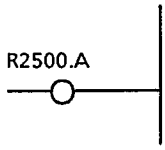


● Function description

This instruction outputs the line logic to bit data specified by n in the registers (R0000 to R0999, R1000 to R4999, P0000 to P3999) specified by S1 and stores the result in the line logic.

● Description

The following example assumes that S1 is in R2500 and n is #A.If the line logic is OFF, bit A is off (in the example R2500 = 0B00H).
If the line logic is ON, bit A is on (in the example R2500 = 0400H).



Line logic ①	R2500 bit A
OFF	OFF (in the example R2500 = 0B00H)
ON	ON (in the example R2500 = 0400H)

Expanded transfer instruction

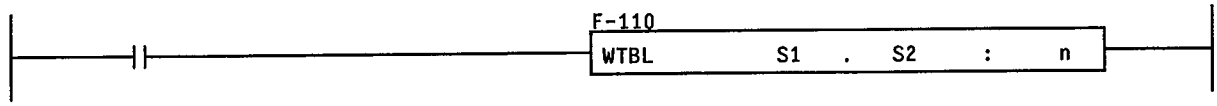
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WTBL	FUN 110	Data table write	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
n	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	-	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	-	-
n	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	R0900 to R0999 P3800 to P3999	#0000 to #9998	-

Ladder circuit example



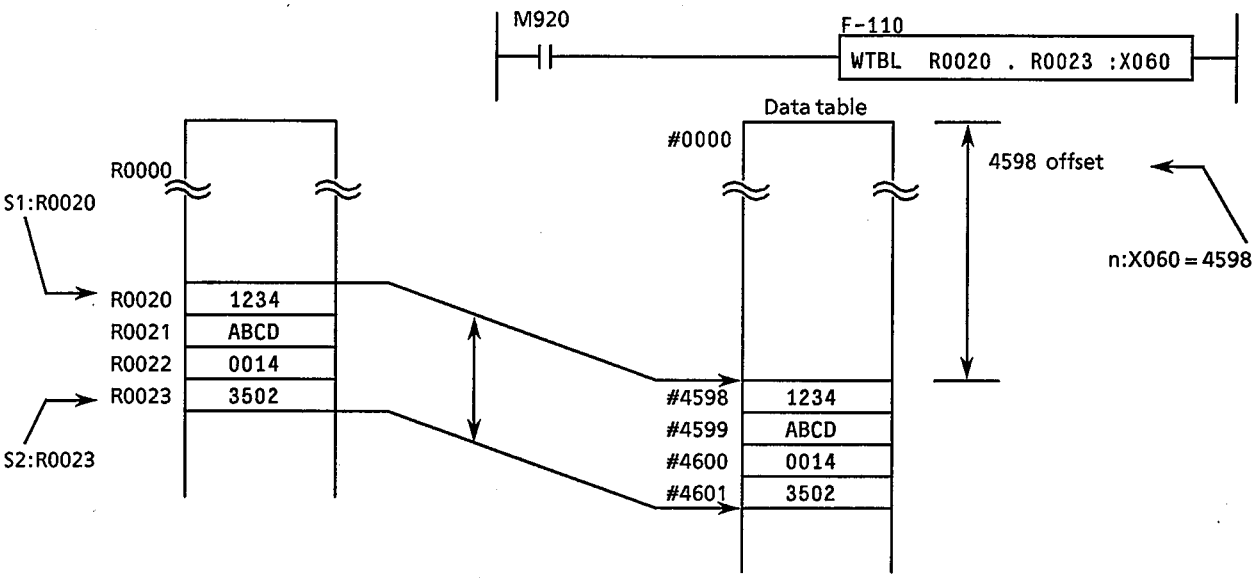
● Function description

This instruction batch transfers the content of devices S1 to S2 (X, Y, R ...) to the starting address indicated by n in a data table (in a user program).
The table write instruction generates an error in the following conditions.

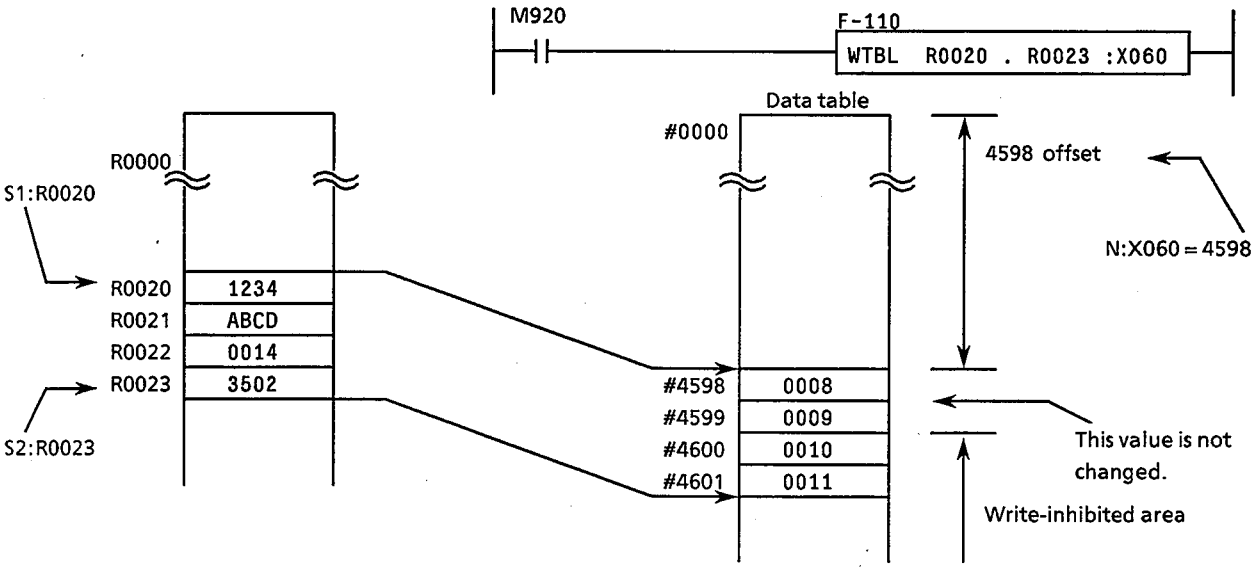
① When there is no table management information.
② When n number of data items specified in the table management information and the parameters (S1, S2 and n) of the table write instruction have the following relationship.
WTBL: S2 - S1 +1 > number of data items n in data table.
③ When the content of n is something other than a BCD value (for example, n being M020 = 89ABH).
④ When the area to which the table is written to or part of it is write-inhibited. (See example 2 on the next page.)
⑤ When the data to be written exceed 256 words.
WTBL: S2 - S1 + 1 > 256

● Description

【 Example 1 】
S1 : R0020
S2 : R0023
n : X060=4598H



【 Example 2 】
Starting address of write-inhibited area : 4600
Last address of write-inhibited area : 5000
S1 : R0020
S2 : R0023
n : X060=4598H



◇Notes◇

- Use only when the data table has been changed.
- Frequent writing will cause a scanning speed drop or data inconformity during scanning. The data inconformity will cause a program-damage error resulting in a problem not moving to the RUN mode.

Expanded transfer instruction

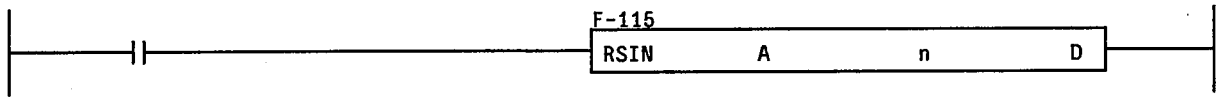
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RSIN	FUN 115	Word module input data read	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
A	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
n	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
A	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0001 to #0008	K1 to K8
n	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #000E	K0 to K14
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



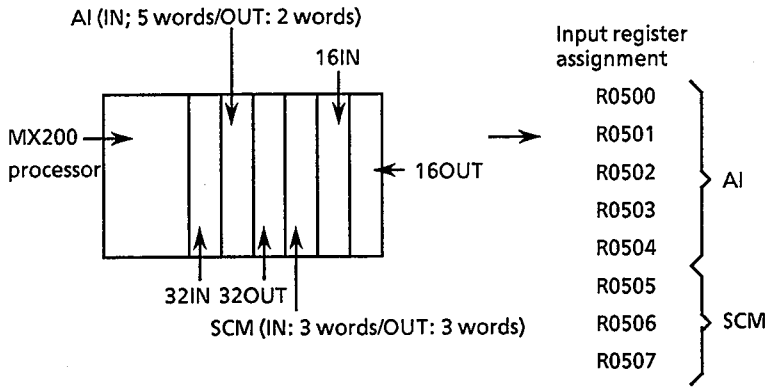
● Function description

This instruction reads one word of input data indicated by number n (zero origin = 0, 1, 2, 3 ...) in a special module inserted in slot number (01 to 08) specified by A to device specified by D when the line logic is on.
The data read is performed by device R0500 to R0519 as it cannot be directly read from the module.
The following conditions cause an operation error and the read operation is aborted.

① When no special module has been inserted in the slot number indicated by A.
② When A indicates slot number 0 or a number higher than 9.
③ When the number of input words in the special module in slot number specified by A does not meet the requirement n + 1.
For example, when the read value of the SCM module is "input 3 words/output 3 words."
The state is then n≧3, and an operation error occurs.

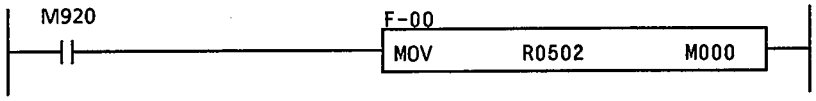
● Description

【 Example 】 Assume that the following I/O modules have been inserted in the MX200.



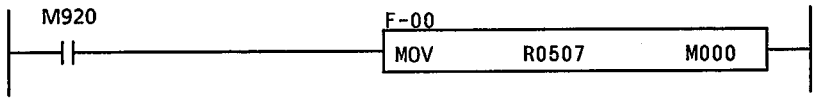
A : @R0000: slot number

① If R0000 = #0002, n = 2 and D: M000 ←R0502



This is processed in the same way.

② If R0000 = #0004, n = 2 and D: M000 ←R0507



This is processed in the same way.

③ R0000≠ #0002,0004, so an operation error occurs.

◇Note◇ Not usable for the MX50.

Expanded transfer instruction

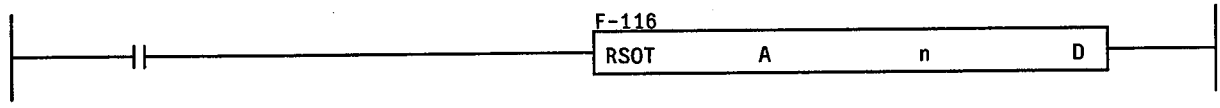
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
RSOT	FUN 116	Word module output data read	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
A	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
n	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D			Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-		R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
A	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0001 to #0008	K1 to K8
n	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #000E	K0 to K14
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



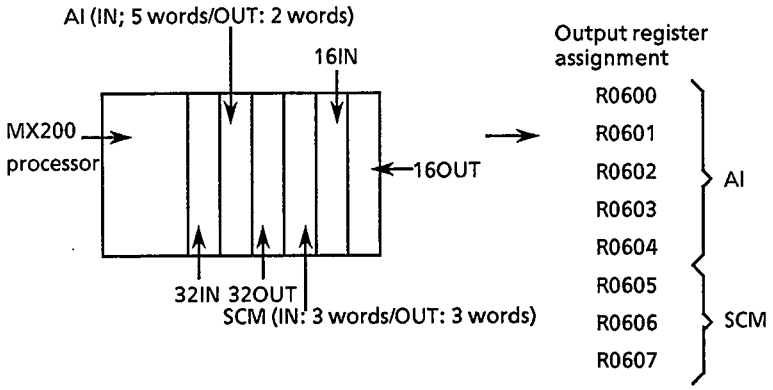
● Function description

This instruction reads one word of input data indicated by number n (zero origin = 0, 1, 2, 3 ...) in special module inserted in slot number (01 to 08) specified by A to device specified by D when the line logic is on.
The data read is performed by device R0600 to R0619 as it cannot be directly read from the module.
The following conditions cause an operation error and the read operation is aborted.

① When no special module has been inserted in the slot number indicated by A.
② When A indicates slot number 0 or a number higher than 9.
③ When the number of input words in the special module in slot number specified by A does not meet the requirement n + 1.
For example, when the read value SCM module is input 3 words / output 3 words, n≧3 and an operation error occurs.

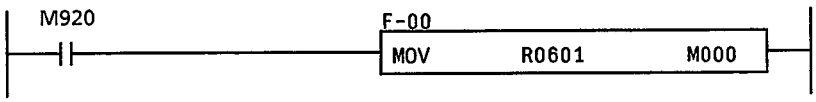
● Description

【 Example 】 Assume that the following I/O modules have been inserted in the MX200.



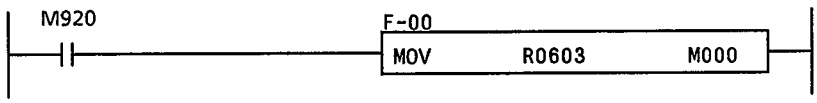
A : @R0000: slot number

① If R0000 = #0002, n = 1 and D: M000 ← R0601



This is processed in the same way.

② If R0000 = #0004, n = 1 and D:M000 ← R0603



This is processed in the same way.

③ R0000 ≠ #0002,0004, so an operation error occurs.

◇Note◇ Not usable for the MX50.

Expanded transfer instruction

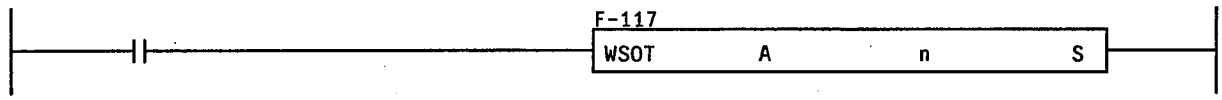
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
WSOT	FUN 117	Word module output data write	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
A	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
n	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
A	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0001 to #0008	K1 to K8
n	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #000E	K0 to K14
S	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767

Ladder circuit example



● Function description

This instruction writes a constant or content of a device specified by S to one word of output data indicated by number n (zero origin = 0, 1, 2, 3 ...) in special module inserted in slot number (01 to 08) specified by A when the line logic is on.

The data read is performed by device R0600 to R0619 as it cannot be directly read from the module.

The following conditions cause an operation error and the read operation is aborted.

① When no special module has been inserted in the slot number indicated by A.

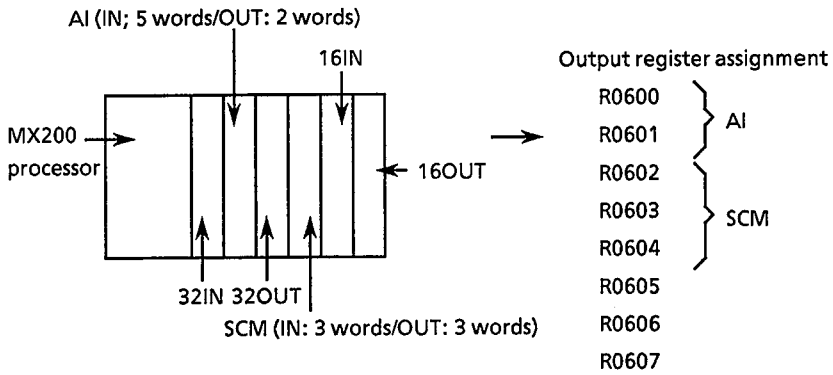
② When A indicates slot number 0 or a number higher than 9.

③ When the number of input words in the special module in slot number specified by A does not meet the requirement n + 1.

For example, when the read value SCM module is input 3 words / output 3 words, n≧3 and an operation error occurs.

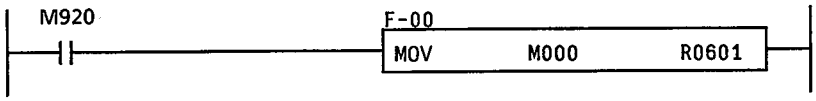
● Description

【 Example 】 Assume that the following I/O modules have been inserted in the MX200.



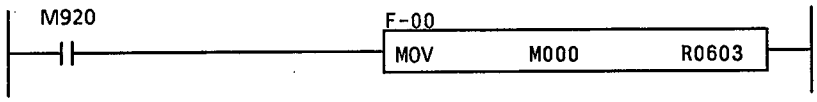
A : @R0000: slot number

① If R0000 = #0002, n = 1 and S: M000 ← R0601



This is processed in the same way.

② If R0000 = #0004, n = 1 and S: M000 ← R0603



This is processed in the same way.

③ R0000 ≠ #0002, 0004, so an operation error occurs.

◇Note◇ Not usable for the MX50.

ASCII communication instruction

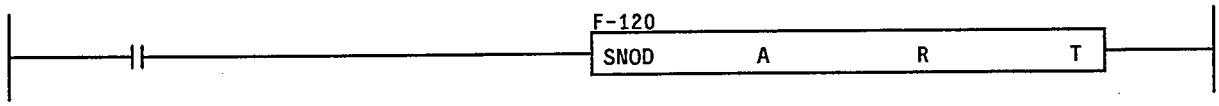
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SNOD	FUN 120	Node setting	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
A	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
R	-	-	-	-	-	-	-	-	-
T	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
A	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	-
R	-	-	-	-	#0000 to #000F	K0 to K15
T	-	-	-	-	#00.01 to #99.99	-

Ladder circuit example



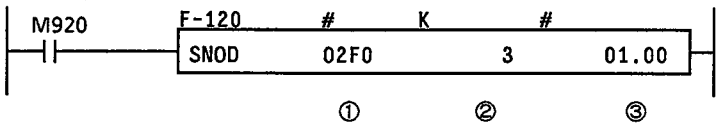
● Function description

This instruction sets A, R and T, the three settings, that are required by a dedicated ASCII communication mode when the line logic is ON.
A indicates the address of the other party.
The high-order bytes of A indicate the address of the other party (for example, address 63 → 3FH) while the low-order bytes indicate the sub-address of the other party.
R indicates the number of retries when a time-out condition occurs. The maximum number of retries is 15.
No retries are made when a communication error occurs.
The following is a list of communication errors.
① ASCII data error
② ASCII reception checksum error
③ ASCII response error
④ ASCII time-out error
T indicates the time after which a time-out occurs. It can be set in 10 ms increments in the range 10 ms to 99.99 ms. These values are stored in R0941.

● Description

【 Example 】

Other party address: 02 ①
Other party subaddress: F0
Number of retries: 3 times ②
Time-out: 1 s ③



Start of transfer : STX'0'2'F'0'X…CR LF

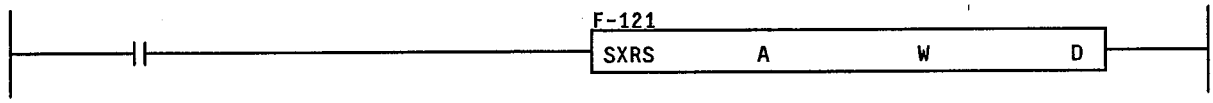
◇Note◇ When this instruction is used to make any settings, the following default values are used.
A...0100H = (the address of the other party is 01 and the subaddress is 00)
R...0001H = (1 retry)
T...Value set in R0941

ASCII communication instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SXRS	FUN 121	Continuous ASCII word read start	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
A	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
W	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
A	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to FFFF	K0 to K65535
W	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0001 to #0020	K1 to K32
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example



● Function description

This instruction sets A, W and D, the three settings required to start a continuous ASCII read operation when the line logic is ON. At the same time, it turns on the ASCII transmission request M938 and the ASCII reception request M939.
It also measures the time-out set by the SNOD instruction.

A indicates the read address which is a decimal in the range 0 to 9999.
A numeric that exceeds 9999 is handled as a positive numeric to 65535 and not as an error so the operation error flags M90E and M90F do not go ON.

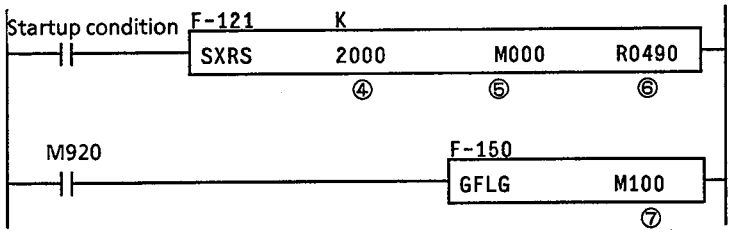
W indicates the number of read words of which a total of 32 words can be specified.
Settings of more than 32 words are handled as a setting of 32 words and does not generate an operation error so the error flag M90E does not go ON.

D indicates the address where the read data is stored.
When D+ (W: the number of words read - 1) exceeds the D area, the ASCII parameter error M936 flag goes ON and FFH is set in the low-order bytes of the internal flag and special contact M93F to M938 is stored in the high-order bytes (= internal flag **FFH).
This condition does not cause the M938 and M939 to go ON, but transmission cannot be performed.
When the process ends normally, the read data is transferred to the area specified by D, the ASCII reception request M939 is turned OFF and the internal flag is set to 0000H.

● Description

【 Example 】

Read address: 2000(decimal) ④
Number of words read: Specified by M000 ⑤
Read data storage: From R0490 ⑥
Device storing status when error occurs: M100 ⑦



1) Normal operation

When M000 = 2,
↓
Start of transmission:
STX****'X'R'S','2'0'0'0'W','2 ETX'sumH'sumL CR LF
Response:
STX****'X'0'0','1'2'3'4',' -'3 ETX'sumH'sumL CR LF
Register : R0490=04D2H
R0491=FFFDH

2) Error operation

① When SXRS number of transmission words M000>10(decimal), The to R0499 area is exceeded and the GFLG instruction stores #00FFH in M100.
② When there is a parameter error in the SXRS response, the GFLG instruction stores the parameter abort code 10 (decimal) in M100.
③ When a checksum error or a time-out condition occurs in the SXRS, the GFLG instruction stores special contacts M93F to M938 in high-order byte M100 and FE in and FE in low-order byte M100.

◇Notes◇

- When the number of retries specified by the SNOD instruction is exceeded, an ASCII time-out error is generated and M93A goes ON.
- Response failure
  When the response code of the reception data is something other than 00, M93C goes ON.
- When the number of retries is exceeded due to a response failure
  The response end code (a decimal) is stored in the low-order byte of the internal flag while the high-order byte is cleared (= internal flag 0001 to 0063H).
- When the number of retries is exceeded due to a time-out
  FEH is set in the low-order byte of the internal flag and special contact M93F to M938 is stored in the high-order byte (= internal flag **FEH).
  As a result, the ASCII transmission request M938 and the ASCII reception request M939 are both automatically turned OFF.
  The internal flag is read by the GFLG (FUN150) instruction.

ASCII communication instruction

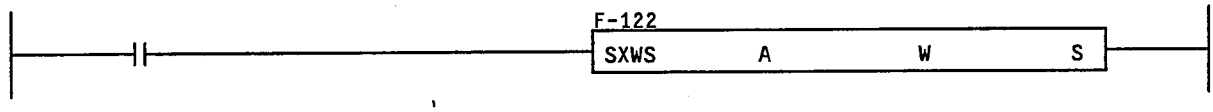
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SXWS	FUN 122	Continuous ASCII word write start	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
A	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
W	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
A	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to FFFF	K0 to K65535
W	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0001 to #0020	K1 to K32
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example



● Function description

This instruction sets A, W and D, the three settings required to start a continuous ASCII read operation when the line logic is ON. At the same time, the ASCII transmission request M938 and the ASCII reception request M939 go ON.
The instruction also measures the time-out set by the SNOD instruction.

A indicates the read address which is a decimal in the range 0 to 9999.
A numeric that exceeds 9999 is handled as a positive numeric to 65535 and not as an error so the operation error flags M90E does not go ON.

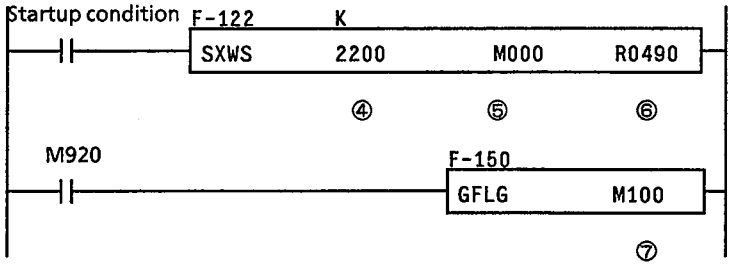
W indicates the number of read words of which a total of 32 words can be specified.
Settings of more than 32 words are handled as a setting of 32 words and does not generate an operation error so the error flag M90E does not go ON.

S indicates the address where the write data is stored.
The W number of words is transmitted from S to the transmission buffer.

● Description

【 Example 】

Write address : 2200(decimal) ④
Number of words write : Specified by M000 ⑤
Write data storage : From R0490 ⑥
Device storing status when error occurs : M100 ⑦



1) Normal operation

Register : R0100=ABCD H
R0101=1234 H
R0102=0002 H
When M000 = 3,
↓
Start of transmission :
STX****'X'W'S','2'2'0'0'W','4'3'9'8'1','4'6'6'0','2 ETX'
sumH'sumL CR LF
↓
Response:STX****'X'0'0 ETX'sumH'sumL CR LF

2) Error operation

① When SXRS number of transmission words M000>10(decimal), The to R0499 area is exceeded and the GFLG instruction stores #00FFH in M100.
② When there is a parameter error in the SXRS response, the GFLG instruction stores the parameter abort code 10 (decimal) in M100.
③ When SXRS causes a time-out or a checksum error, the GFLG instruction stores the special contact M93F to M938 in the high-order byte and FE in the low-order byte.

◇Notes◇

- When S + (W: the number of words written - 1) exceeds the S area, the ASCII parameter error M936 flag goes ON and FFH is set in the low-order byte of the internal flag and special contact M93F to M938 is stored in the high-order byte (= internal flag **FFH).
  This condition does not cause the M938 and M939 to go ON, but transmission cannot be performed.
  When the write process ends normally, the ASCII reception request M939 in the overhead is turned OFF and the internal flag is set to 0000H.
- Response failure
  When the response code of the reception data is something other than 00, M93C goes ON.
- When the number of retries specified by the SNOD instruction is exceeded, an ASCII time-out error is generated and M93A goes ON.
- When the number of retries is exceeded due to a response failure
  The response end code (a decimal) is stored in the low-order byte of the internal flag while the high-order byte is cleared (= internal flag 0001 to 0063H).
- When the number of retries is exceeded due to a time-outFEH is set in the low-order byte of the internal flag and special contact M93F to M938 is stored in the high-order byte (= internal flag **FEH).
  As a result, the ASCII transmission request M938 and the ASCII reception request M939 are both automatically turned OFF.
  The internal flag is read by the GFLG (FUN150) instruction.

Character string processing instruction

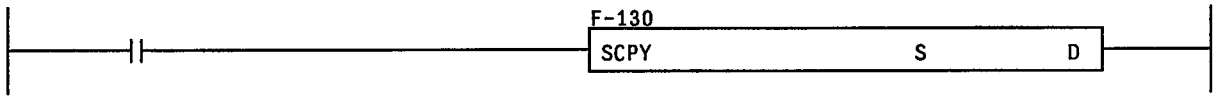
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SCPY	FUN 130	Character string copy	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example

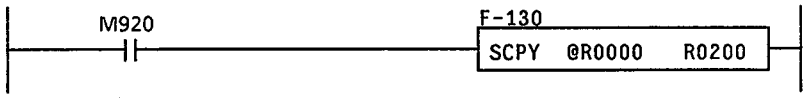


● Function description

This instruction copies the character string data starting from S to D. When the data at S is not character string D, an operation error is generated and the copy operation is aborted.
When the character string is too long to store in the D area, an operation error is generated and the copy operation is aborted.

● Description

【 Example 1 】



When R0000 = #0198H

Before execution of instruction

R0197	AA BB
R0198	00 30
R0199	01 31
R0200	02 32
R0201	03 33
R0202	0A 41
R0203	0B 42
R0204	AA 00
R0205	12 34
R0206	56 78
R0207	9A BC

After execution

R0197	AA BB
R0198	00 30
R0199	01 31
R0200	00 30
R0201	01 31
R0202	02 32
R0203	03 33
R0204	0A 41
R0205	0B 42
R0206	AA 00
R0207	9A BC

【 Example 2 】

When R100 = #0490h

Before execution of instruction

	AA	BB
R0489		
R0490	00	30
R0491	01	31
R0492	02	32
R0493	03	33
R0494	04	34
R0495	05	35
R0496	06	36
R0497	07	37
R0498	08	38
R0499	00	00

After execution

	AA	BB
R0489		
R0490	05	35
R0491	06	36
R0492	07	37
R0493	08	38
R0494	00	00
R0495	05	35
R0496	06	36
R0497	07	37
R0498	08	38
R0499	09	39

When R0499 = 0939H, an operation error is generated.

The character strings that are handled by the character string processing instructions (FUN130 to FUN139, FUN140 to FUN148) are defined as follows.

① A character string can be a maximum of 256 characters including NUL code (= **00H).

② Character strings can be used only in the register area (R0000 to R0499, R1000 to R4999) of the device table or the data table (defined by the DTBL instruction). (X, Y and L areas are invalid.)

③ The low-order bytes of specified data are seen as character strings and are processed as words.
Thus #000AH is not identified as a NUL code.
For example, R1000 = #1234H → character 34H = '4

④ When a register area is specified in a character string
The data from the start of the specified register to the NUL code (= **00H) is identified as a character string.
When no NUL code is found from the start of the specified register and up to the 256th word, the data is not identified as a character string.
When the NUL code cannot be found in the area from the beginning of the specified register and the upper limit of the storage register (= R0499 or R4999), the data is not identified as a character string.

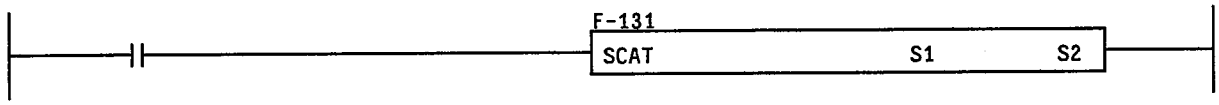
⑤ #0000 to 9998 was specified in a character string
Data starting from the table number of the data table defined by DTBL (FUN98) and ending in the NUL code (**00H) is recognized as a character string.
The data must be within 256 words from the table number to the upper limit of the data table (for example, DTBL #10 →10) and the NUL code must be within this range or the data is not identified as a character string.
When the NUL code cannot be found from the table number within 256 words, the data is not recognized as a character string.

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SCAT	FUN 131	Character string concatenation	7 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	-	-
S2	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	-	-	-	-
S2	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-

Ladder circuit example



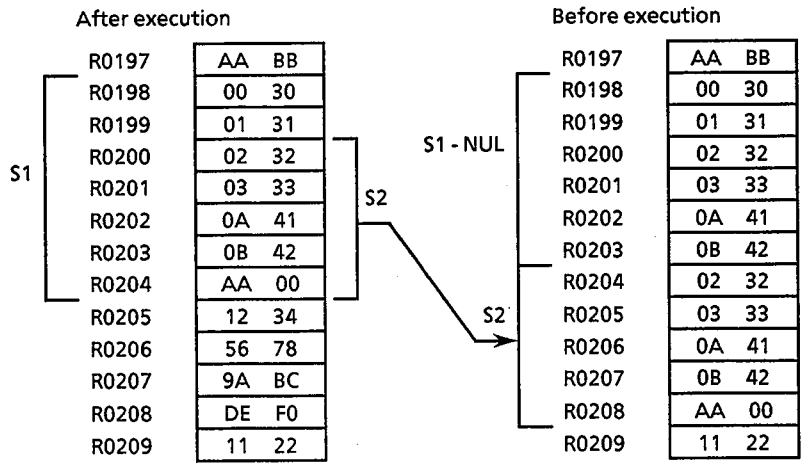
● Function description

This instruction concatenates the character string from S1 to NUL (**00H) with character data at S2 when the line logic is ON.
When the data at S1 or S2 cannot be identified as character data, an error is generated and the transfer is aborted.
When S1+S2 specifies a character string that is too long to store in the S1 area, an operation error occuurs and the transfer does not take place.

● Description

【 Example 】

R0000 = #0198h



See SCPY (FUN130) for information on character string constants.

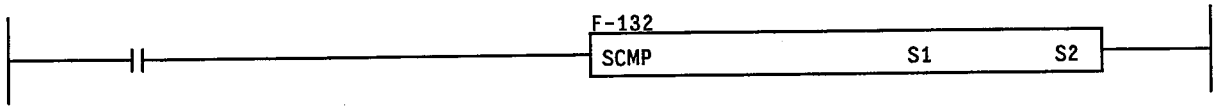
廃番

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SCMP	FUN 132	Character string compare	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	-	-
S2	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
S2	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-

Ladder circuit example

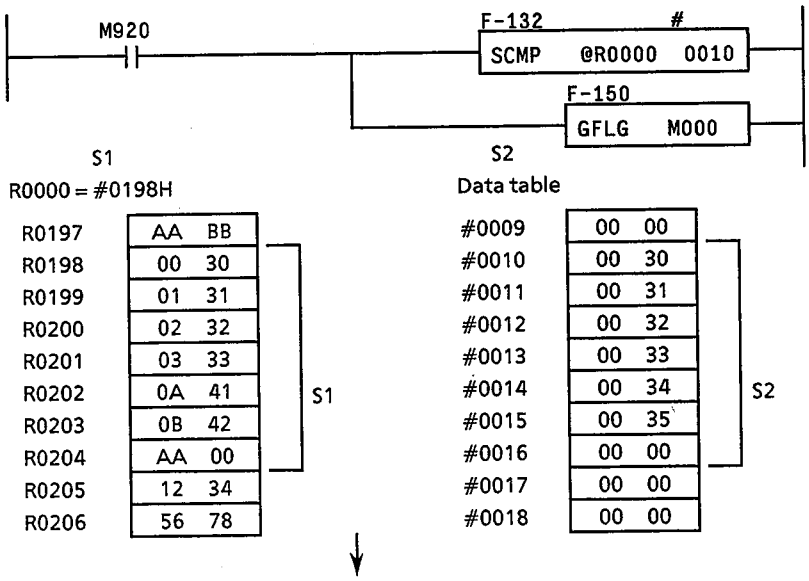


● Function description

This instruction compares S1 character string data with S2 character string data when the line logic is ON. If the data match, special contact M903 is turned on and 8000H is set in the internal flag.
If the data pairs do not match, special contact M903 goes off and the characters that do match are stored from the start of the internal flag.
When the data at S1 or S2 cannot be identified as character data, an operation error is generated, special contact M903 goes off and FFFFH is placed in the internal flag.
The internal flag can be read with the GFLG (FUN150) instruction.

● Description

【 Example 】



Processing result : M903 ← OFF mismatch
Internal flag = M000 : 0004 **30 to **34 match = 4 characters match

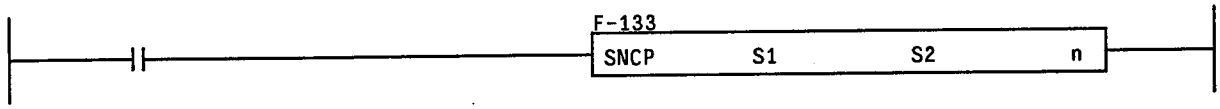
See SCPY (FUN130) for information on character string constants.

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SNCP	FUN 133	Character string character number compare	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	-	-
S2	-	-	-	-	-	-	-	-	-
n	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
S2	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
n	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0001 to #00FF	K1 to K255

Ladder circuit example



● Function description

This instruction compares "n" words from the starting address of character string data at S1 with character string data at S2. If the data match, special contact M903 is turned on and 8000H is set in the internal flag when the line logic is ON.

If the data does not match, special contact M903 goes off and the characters that do match are stored from the start of the internal flag.

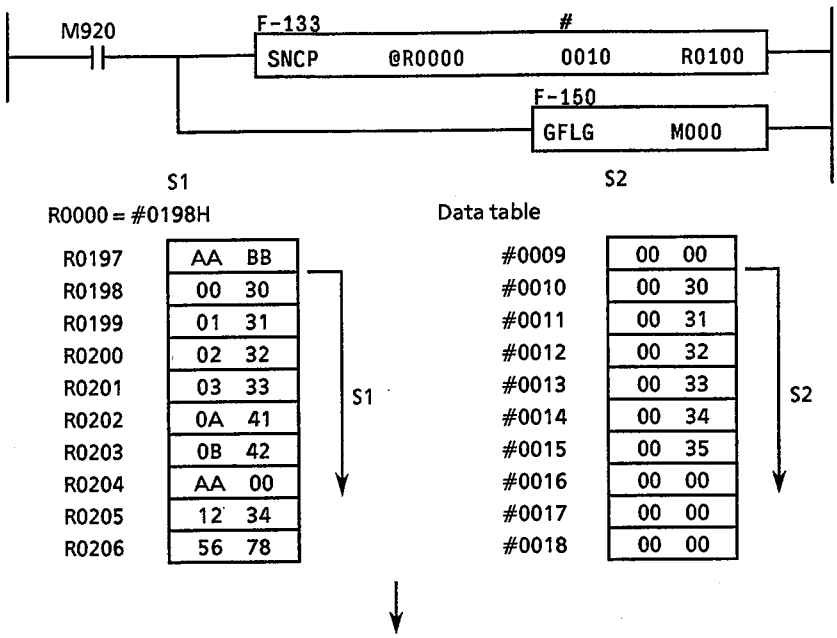
In the following cases, an operation error occurs, special contact M903 goes off and FFFFH is set in the internal flag.

①: When S1 and S2 data cannot be identified as character data.
②: When "n" is 0.
③: When "n" exceeds 255.
④: When the number of characters at S1 or S2 are fewer than "n".

The internal flag can be read with the GFLG (FUN150) instruction.

● Description

【 Example 】



① When 1≦R0100≦4
Processing result : M903 ← ON ...match
Internal flag=M000 : 8000

② When 4<R0100≦6
Processing result : M903 ← OFF...mismatch
Internal flag=M000 : 0004...**30 to **34 match = 4 characters match

③ When 6<R0100,R0100=0
An operation error is generated
Processing result : M903 ← OFF...mismatch
Internal flag=M000 : FFFF

See SCPY (FUN130) for information on character string constants.

Character string processing instruction

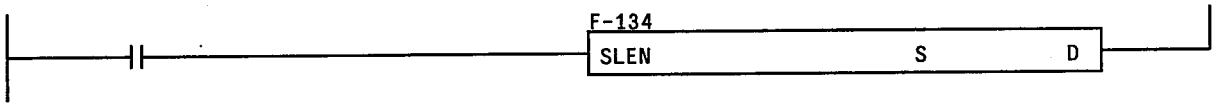
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SLEN	FUN 134	Character string length check	7 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example



● Function description

This instruction checks the number of characters at S and transfers all characters except NUL (= **00H) to D when the line logic is ON.
When the data at S cannot be identified as character data, an error is generated and the operation is aborted.

● Description

【 Example 1 】

R0000 = #0198H

R0197	AA	BB
R0198	00	30
R0199	01	31
R0200	02	32
R0201	03	33
R0202	0A	41
R0203	0B	42
R0204	AA	00
R0205	12	34

S →M000 = 6

R0198 to R0204 : 7 characters – NUL

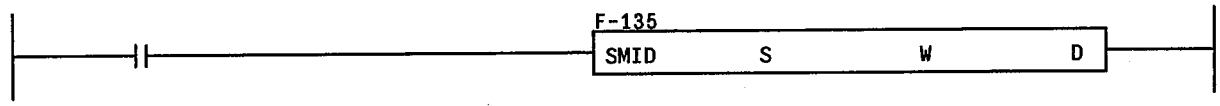
See SCPY (FUN130) for information on character string constants.

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SMID	FUN 135	Character string extraction	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
W	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
W	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #FFFF	-
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example



● Function description

This instruction transfers character string data from S+ (the high-order bytes of W) word to D when the logic is ON. A NUL code is added at the end of the extracted character string.
When a NUL code is encountered in the range from S to S+ (the high-order bytes of W) or when the upper limit of the area is detected, the low-order bytes of W and the NUL code is transferred to D.
When NUL or the area upper limit is encountered in the range from S+ (the high-order bytes of W) to S + (the high-order bytes of W) and (the low-order bytes of W), the NUL codes for the missing words are inserted.
An operation error occurs in the following cases and transfer is aborted.

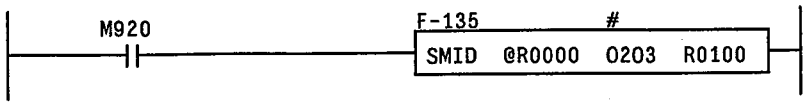
①: When the data from S cannot be identified as character string data.

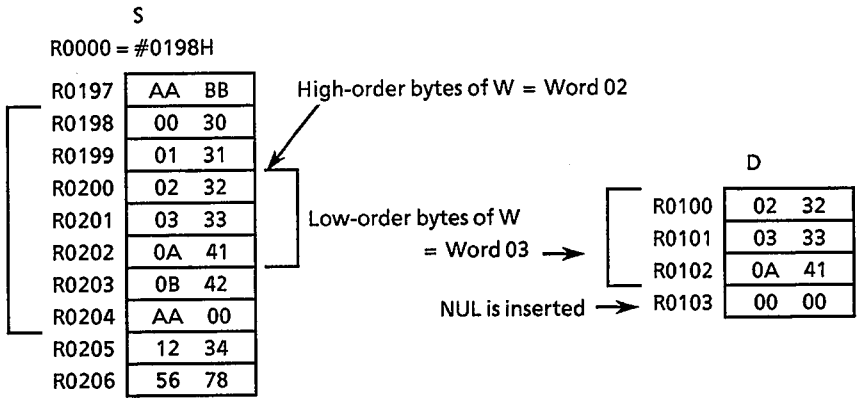
②: When the number of words to be extracted, i.e. the low-order bytes of W, is 0.

③: When the number of the words to be extracted, i.e. the low-order bytes of W, cannot be stored in area D.

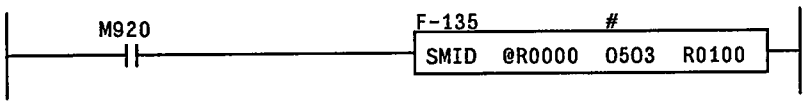
● Description

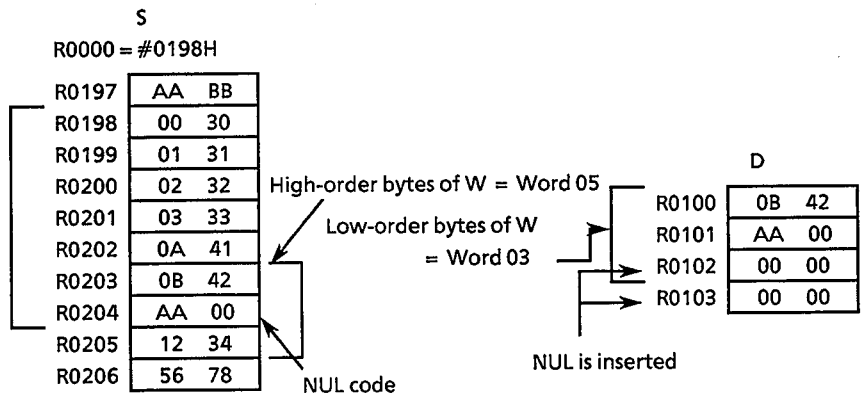
【 Example 1 】





【 Example 2 】





See SCPY (FUN130) for information on character string constants.

Character string processing instruction

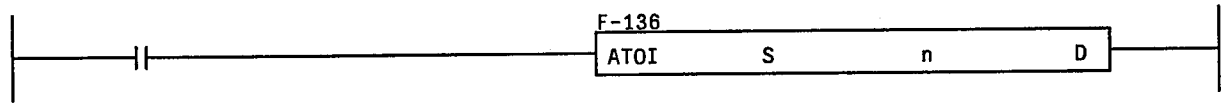
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ATOI	FUN 136	ASCII character string →Decimal conversion	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	-	-
n	-	R0000 to R0499 R1000 to R4999	-	-	#0001 to #0080	K1 to K128
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit diagram



● Function description

This instruction treats "n" words of ASCII numeric data from S or between delimiters (which are not decimal ASCII data) as decimals and converts the words to binary data which is stored at D when the line logic is ON.
The ASCII data at S is converted as follows:

①: When data starting from S or after the delimiters is '0 to '9 ASCII data, it is treated as decimal data in the range 0 to 65535 and converted.

②: When data starting from S or after the delimiters is ASCII' - (= **2DH) data, it is treated as decimal data in the range -32768 to 32767 and converted.

③: When data starting from S or after the delimiters is ASCII + (= **2BH) data, it is treated as decimal data in the range 0 to 65535 and converted.

④: The data is zero suppressed in the conversion, for example '0'0'1'2'3 = '1'2'3 → 007BH, however, the numeric "0" is not suppressed.
Thus an operation error occurs when the numeric "0" is converted to 0000H.

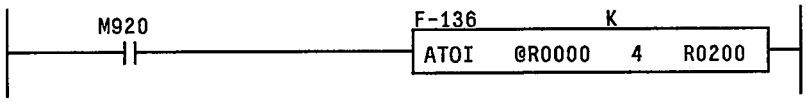
⑤: The use of signed data is limited to '+ and '-, and '0 cannot be used after a sign. For example, '+'0'1/'-'0'1 results in an operation error.

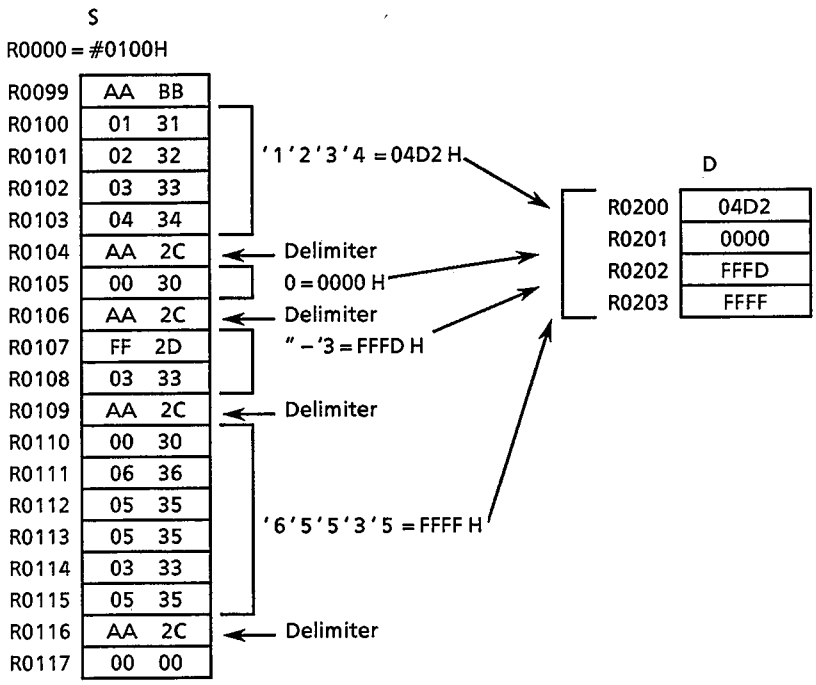
Values that exceed the decimal range of a word (for example, 99999 or -40000) or character strings other than '0 to '9 data cause an operation error and aborts the conversion.
When the number of words specified by "n" is not character string data, the specified number of words are converted and an operation error is generated. The string 0000H is inserted for the missing words at D.

The following cases cause an operation error and aborts the conversion.
①: When the data from S cannot be identified as character string data.
②: When "n" is 0.
③: When "n" exceeds 128.
④: When D + "n" - 1 exceeds the D area.

● Description

【 Example 】





See SCPY (FUN130) for information on character string constants.

Character string processing instruction

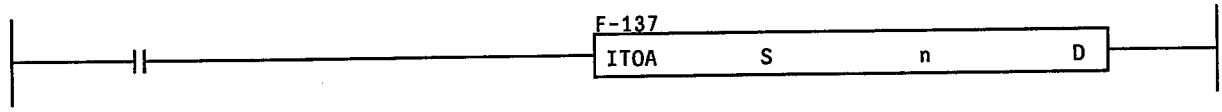
Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ITOA	FUN 137	Decimal→ASCII character string conversion	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	-	-
n	-	R0000 to R0499 R1000 to R4999	-	-	#0001 to #0080	K1 to K128
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example

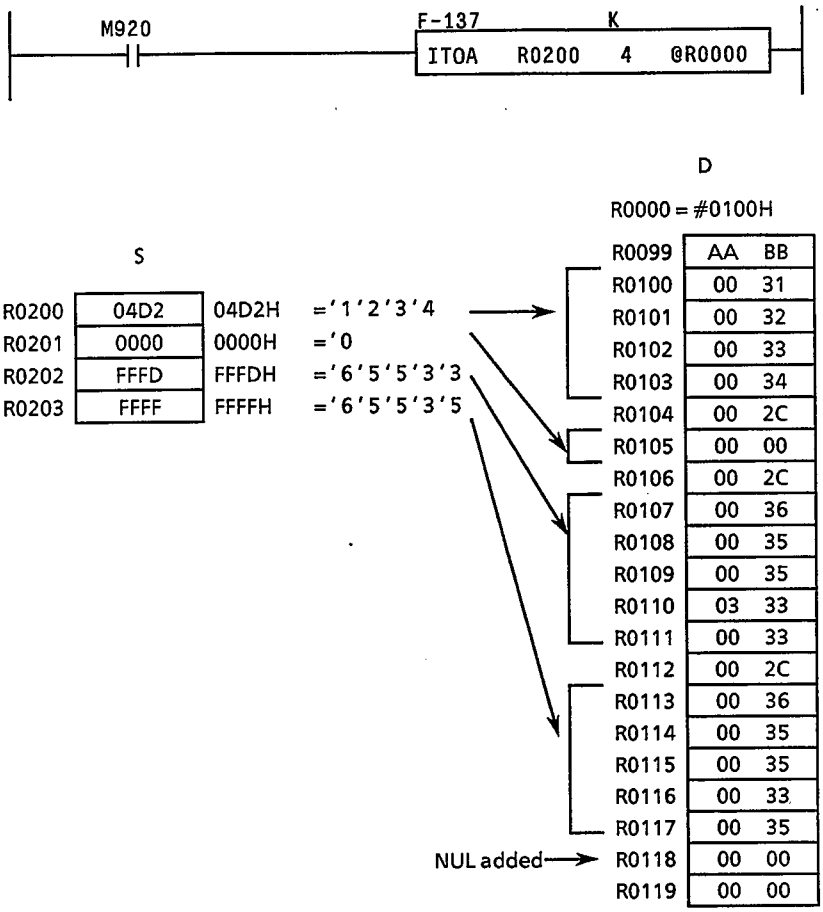


● Function description

This instruction converts "n" words of unsigned decimal data in S to ASCII data which is stored in D when the line logic is ON.
The converted ASCII data is delimited by commas (=, = 002CH). A NUL code is inserted at the end without a comma.
When D+ ("n" words of ASCII data) exceeds the D area, an operation error occurs and the data that cannot be stored in the D area is ignored.
An operation error is generated and conversion does not take place in the following cases.
①: When "n" is 0.
②: When "n" exceeds 128.

● Description

【 Example 】



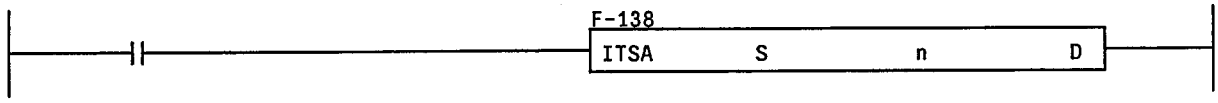
See SCPY (FUN130) for information on character string constants.

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ITSA	FUN 138	Decimal→Signed ASCII character string conversion	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	-	-
n	-	R0000 to R0499 R1000 to R4999	-	-	#0001 to #0080	K1 to K128
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example

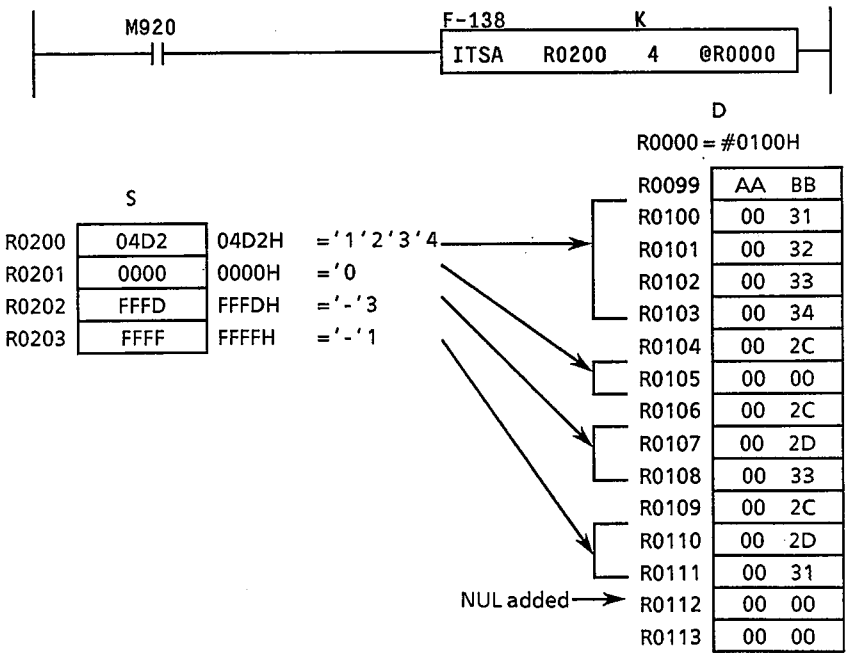


● Function description

This instruction converts "n" words of signed decimal data in S to ASCII data which is stored in D when the line logic is ON.
The converted ASCII data is delimited by commas (=, = 002CH). A NUL code is inserted at the end without a comma.
When D+ ("n" words of ASCII data) exceeds the D area, an operation error occurs and the data that cannot be stored in the D area is ignored.
An operation error is generated and conversion does not take place in the following cases.
①: When "n" is 0.
②: When "n" exceeds 128.

● Description

【 Example 】



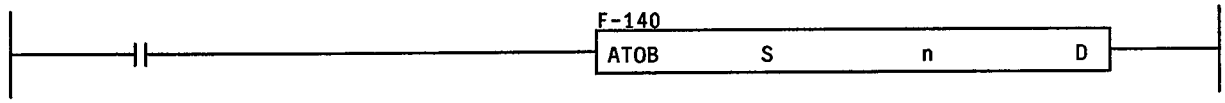
See SCPY (FUN130) for information on character string constants.

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
ATOB	FUN 140	ASCII hexadecimal character strings→Hexadecimals	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	-	-
n	-	R0000 to R0499 R1000 to R4999	-	-	#0001 to #003F	K1 to K63
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example



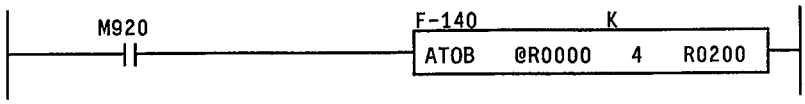
● Function description

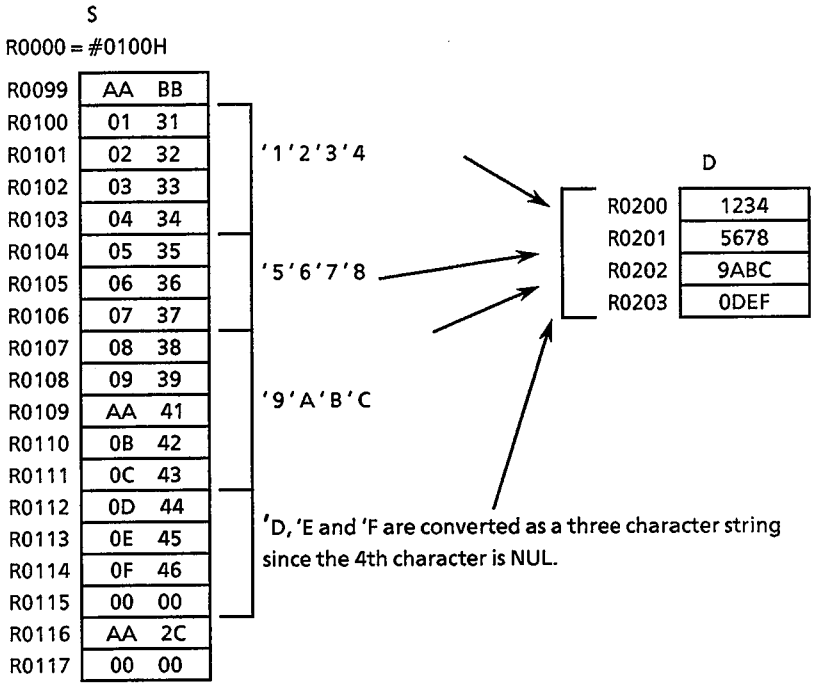
This instruction takes "n" words of four character ASCII hexadecimals in S handling them as one word hexadecimal strings and converts them into "n" words of binary data which is stored in D when the line logic is ON.
When character strings other than '0 to '9 and A to F are found, an operation error occurs and the conversion is aborted.
NUL data found in 4-character strings is converted as character strings with too few digits and generate an operation error.
When the number of character strings indicated by "n" words could not be found, the specified number of characters is converted and an operation error is generated. The character 0000H is set in the D area where there are too few characters.
An operation error is generated and conversion is aborted in the following cases.

①: When the data in S cannot be identified as character strings.
②: When "n" is 0.
③: When "n" exceeds 63.
④: When D + n - 1 exceeds the D area.

● Description

【 Example 】





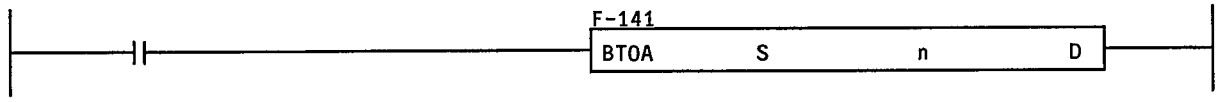
See SCPY (FUN130) for information on character string constants.

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
BTOA	FUN 141	Hexadecimal→ASCII hexadecimal character string conversion	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	-	-
n	-	R0000 to R0499 R1000 to R4999	-	-	#0001 to #003F	K1 to K63
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example

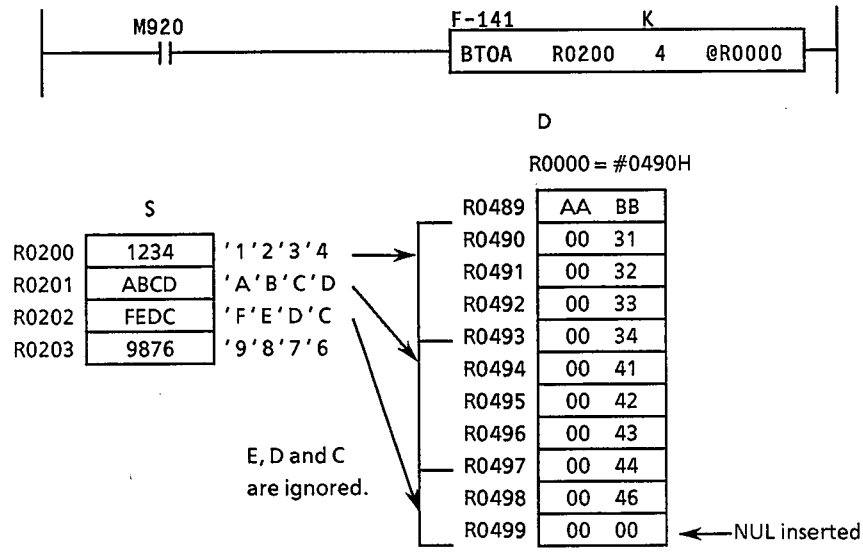


● Function description

This instruction converts "n" words of unsigned hexadecimal data in S to ASCII hexadecimal data which is stored in D when the line logic is ON.
The converted ASCII data is delimited by commas and a NUL code is inserted at the end.
When D + ("n" words of ASCII data) exceeds the D area, an operation error occurs and the data that cannot be stored in the D area is ignored while NUL is entered in R0499/R4999.
An operation error is generated and conversion is aborted in the following cases.
①: When "n" is 0.
②: When "n" exceeds 63.

● Description

【 Example 】



See SCPY (FUN130) for information on character string constants.

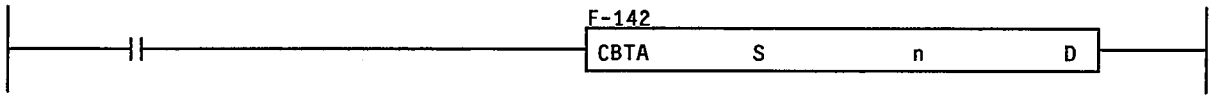
Character string processing instruction	Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
CBTA	FUN 142	Hexadecimal→ASCII hexadecimal conversion and concatenation	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	-	-
n	-	R0000 to R0499 R1000 to R4999	-	-	#0001 to #003F	K1 to K63
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example



● Function description

This instruction converts "n" words of unsigned hexadecimal data in S to ASCII hexadecimal data which is concatenated at D when the line logic is ON. A NUL code is inserted at the end of the converted ASCII data. When D + ("n" words of ASCII data) exceeds the D area, an operation error occurs and the data that cannot be stored in the D area is ignored while NUL is entered in R0499/R4999.

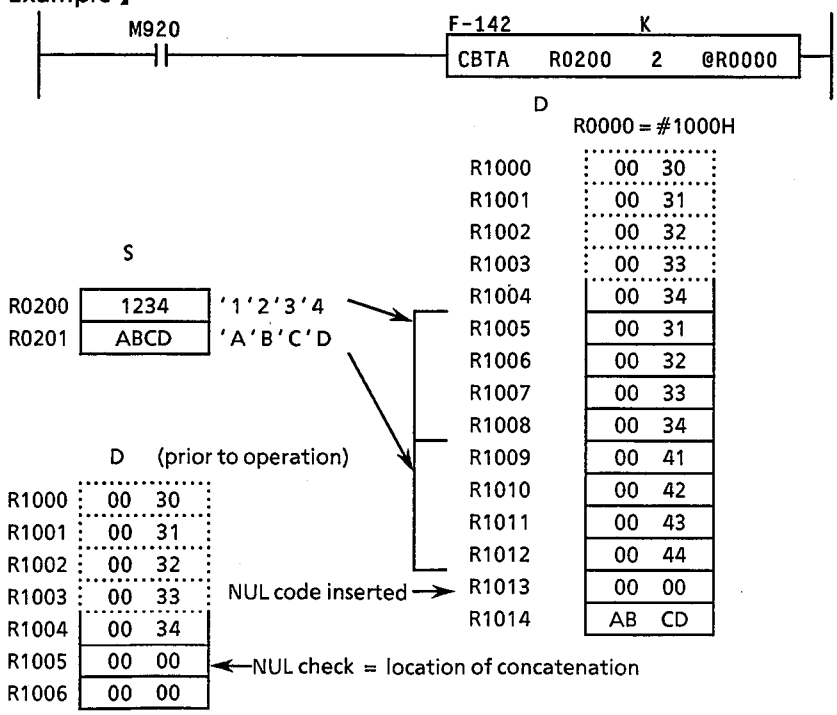
An operation error is generated and conversion does not take place in the following cases.

①: When "n" is 0.

②: When "n" exceeds 63.

● Description

【 Example 】



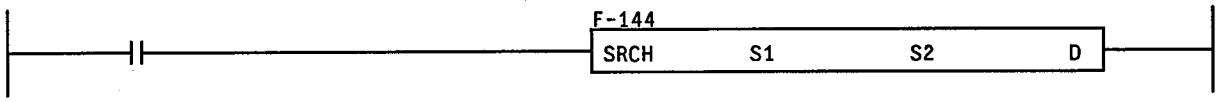
See SCPY (FUN130) for information on character string constants.

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SRCH	FUN 144	Character string retrieval	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	-	-	-	-	-	-	-	-	-
S2	-	-	-	-	-	-	-	-	-
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
S2	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

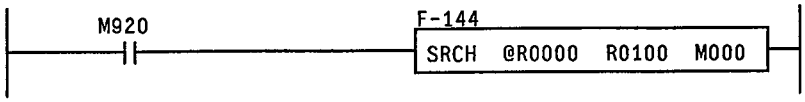


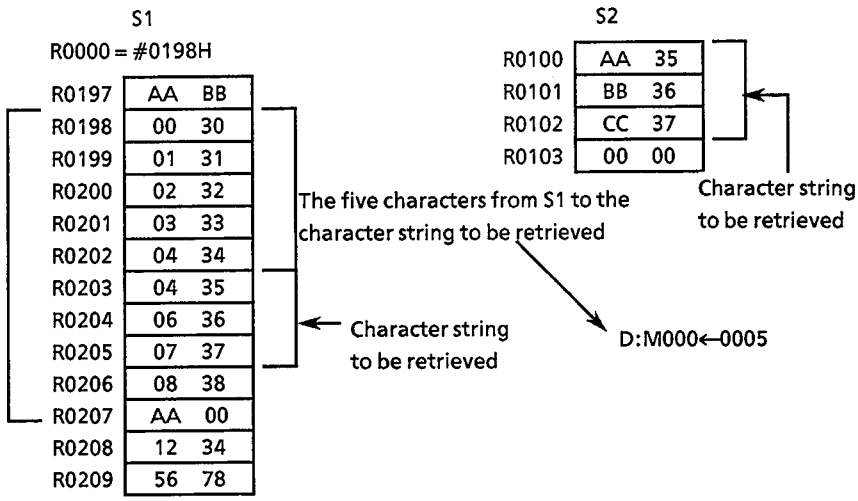
● Function description

This instruction searches for the character string specified at S2 in S1 when the line logic is ON. The number of characters from the start of S1 to the string searched for (not the character string itself) is transferred to D (D is a binary value so 100 characters is 0064H).
When the string searched for is found in data other than character data, the number of characters up to the NUL code is transferred to D.
This instruction thus performs the same operation as the F-134:SLEN instruction.
When the data at S1 cannot be identified as character data, an error is generated and the operation is aborted.

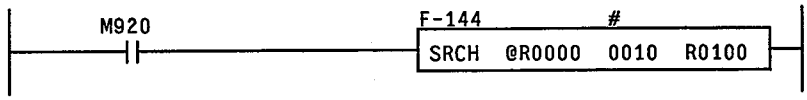
● Description

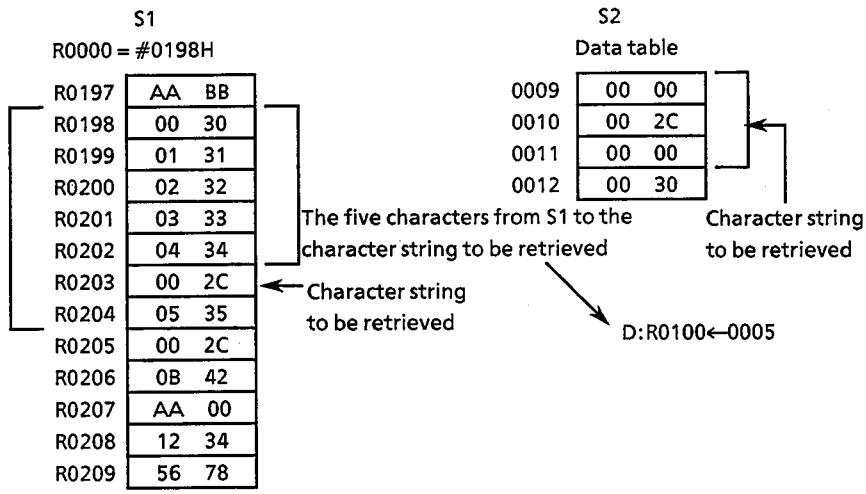
【 Example 1 】





【 Example 2 】





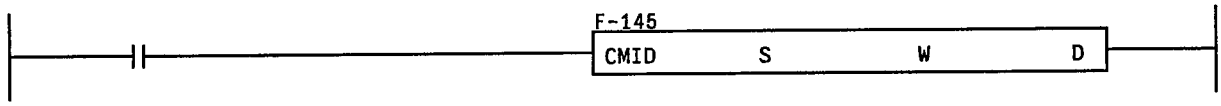
See SCPY (FUN130) for information on character string constants.

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
CMID	FUN 145	Character string extraction and concatenation	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
W	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #9998	-
W	-	R0000 to R0499 R1000 to R4999	-	-	#0000 to #FFFF	-
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example



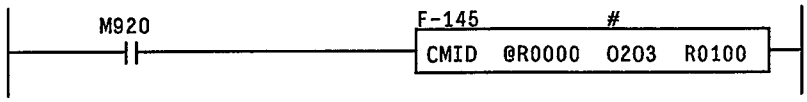
● Function description

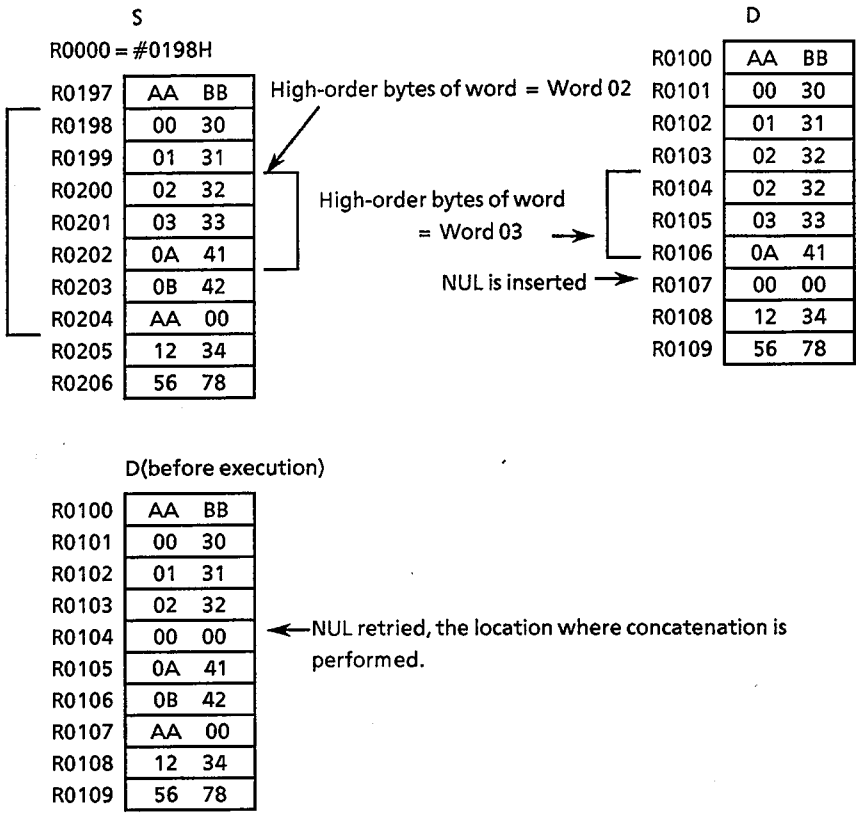
This instruction concatenates character string data from S+ (the high-order bytes of W) with word (the low-order bytes of W) at D when the line logic is ON.
A NUL code is added at the end of the extracted character string.
When a NUL code is encountered in the range from S to S+ (the high-order bytes of W) or when the upper limit of the area is detected, the NUL codes of the words (low-order bytes of W) are concatenated at D.
When NUL or the area upper limit is encountered in the range from S+ (the high-order bytes of W) to S + (the high-order bytes of W) and (the low-order bytes of W), the NUL codes for the missing words are inserted.
An operation error occurs in the following cases and the transfer is aborted.

①: When the data from S cannot be identified as character string data.
②: When the number of the words to be extracted, i.e. the low-order bytes of W, is 0.
③: When the number of words to be extracted, i.e. the low-order bytes of W, cannot be stored in area D.

● Description

【 Example 】





See SCPY (FUN130) for information on character string constants.

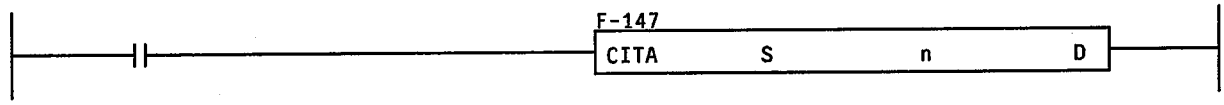
Character string processing instruction	Execution condition	Input ON	Pulse specification	○	Indirect specification	○

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
CITA	FUN 147	Decimal→ASCII character string conversion and concatenation	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	-	-
n	-	R0000 to R0499 R1000 to R4999	-	-	#0001 to #0080	K1 to K128
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example

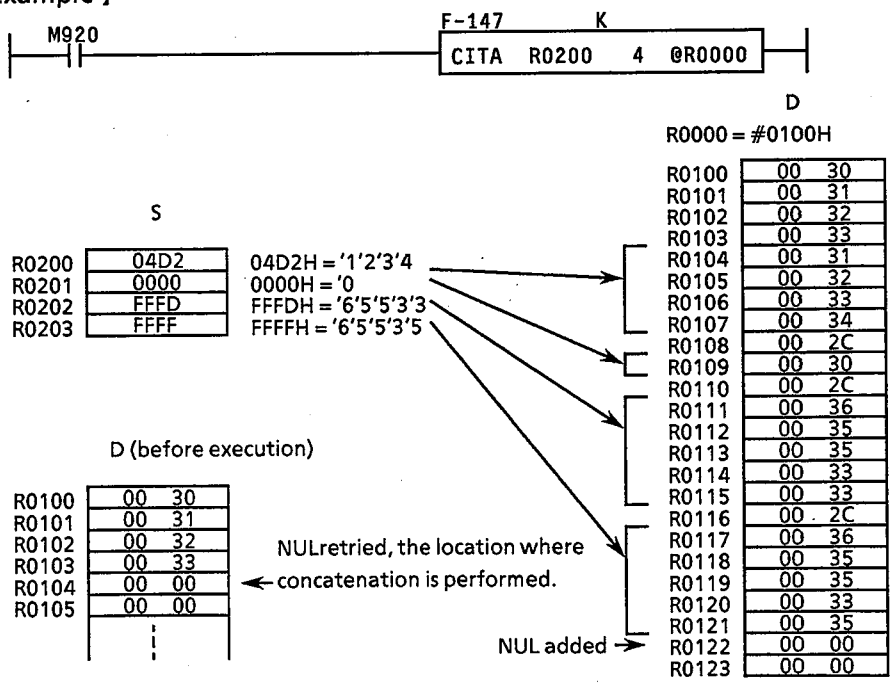


● Function description

This instruction converts "n" words of unsigned decimal data in S to ASCII data which is concatenated to data in D when the line logic is ON.
The converted ASCII data is delimited by commas (=, = 002CH).
A NUL code is inserted at the end without a comma.
When D+ ("n" words of ASCII data) exceeds the D area, an operation error occurs and the data that cannot be stored in the D area is ignored.
An operation error is generated and conversion does not take place in the following cases.
①: When "n" is 0.
②: When "n" exceeds 128.

● Description

【 Example 】



See SCPY (FUN130) for information on character string constants.

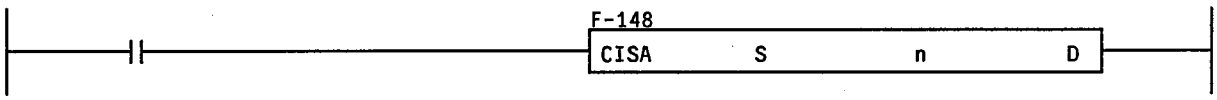
廃番

Character string processing instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
CISA	FUN 148	Decimal→Signed ASCII character string conversion and concatenation	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S	-	-	-	-	-	-	-	-	-
n	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S	-	R0000 to R0499 R1000 to R4999	-	-	-	-
n	-	R0000 to R0499 R1000 to R4999	-	-	#0001 to #0080	K1 to K128
D	-	R0000 to R0499 R1000 to R4999	-	-	-	-

Ladder circuit example

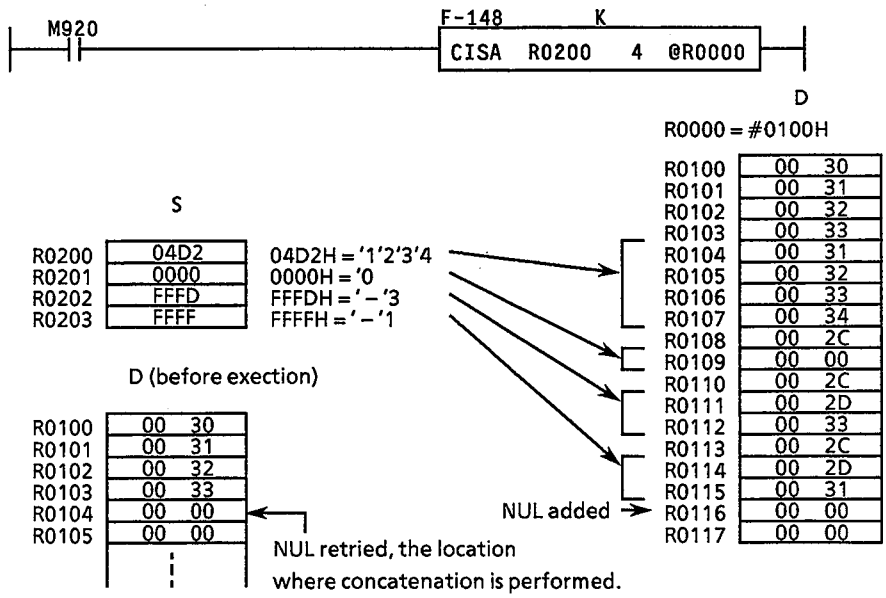


● Function description

This instruction converts "n" words of signed decimal data in S to ASCII data which is stored in D when the line logic is ON.
The converted ASCII data is delimited by commas (=, = 002CH). A NUL code is inserted at the end without a comma.
When D+ ("n" words of ASCII data) exceeds the D area, an operation error occurs and the data that cannot be stored in the D area is ignored.
An operation error is generated and conversion is aborted in the following cases.
①: When "n" is 0.
②: When "n" exceeds 128.

● Description

【 Example 】



See SCPY (FUN130) for information on character string constants.

Expanded auxiliary instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
GFLG	FUN 150	Internal flag read	5 bytes		Overflow M900	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

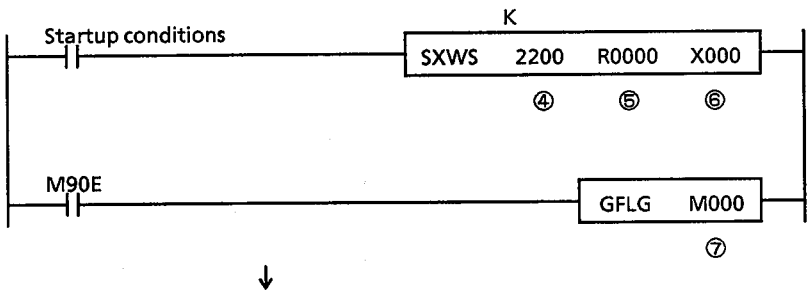
● Function description

This instruction outputs the data of the internal flag to D when the line logic is ON. When part of the result of ASCII processing instructions SXRS and SXWS or character string processing instructions SCMP and SNCP is stored in the flag, this instruction is used to output the content of the flag to a device.

● Description

【 Example 】

Place storing error statuses: M000 ⑦



1) When the number of words specified by an SXWS instruction is :R0000> #20, the X000 to 190 area overflows and the GFLG instruction places #00FFH in M000.
2) When a parameter error is found in the response to an SCWS instruction, the GFLG instruction places 10 (decimal: the parameter error end code) in M000.

Expanded auxiliary instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	○
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
EFLG	FUN 151	Detailed operation error information read	5 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

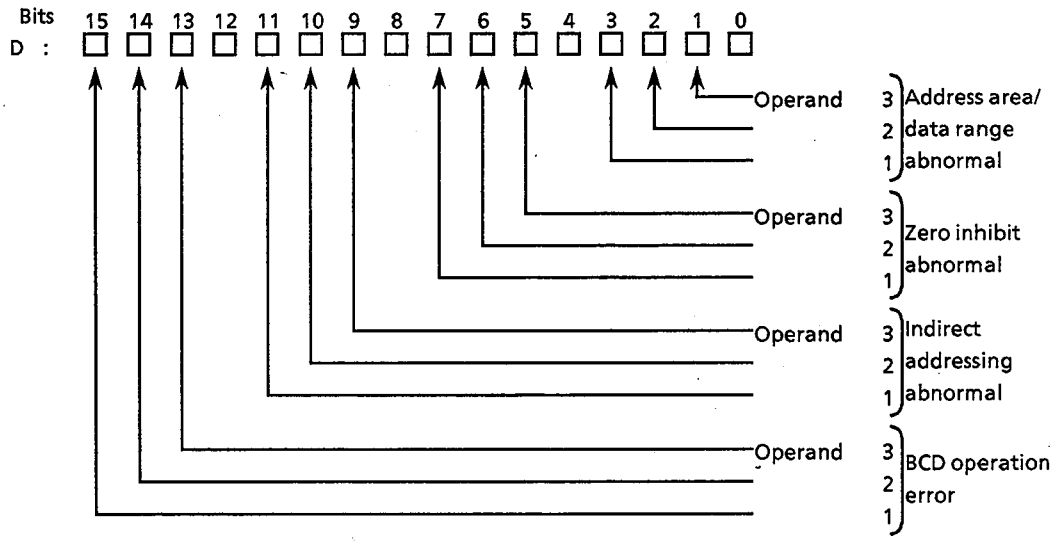
Ladder circuit example

● Function description

This instruction outputs detailed operation error information to D when the line logic is ON.
When indirect addressing is used and it is likely that several character string processing instructions have caused operation errors, the instruction is used to output information on which factor interrupted processing to the device table.

◇Note◇ This instruction provides information on only one factor that interrupted processing.
Consequently, there may be other factors that caused the interruption.
The procedures for checking factors causing operation errors depend on the instruction.

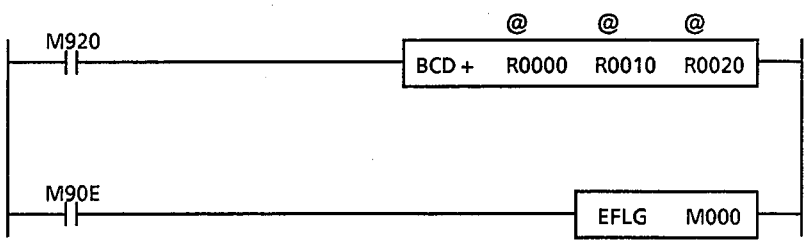
Operation error information is displayed as shown below.



● Description

【 Example 】

Place storing error information3 M000



↓

① When R0000＝#ABCD, R0010＝****, R0020＝****
→M000＝0000 1000 0000 0000 B
② When R0000＝#0100, R0010＝#ABCD, R0020＝****
→M000＝0000 0100 0000 0000 B
③ When R0000＝#0100, R0010＝#0200, R0020＝#ABCD
→M000＝0000 0010 0000 0000 B

From the above it follows : R0000＝#0100, R0010＝#0200, R20＝#0300

④ When R0100＝#ABCD, R0200＝****, R0300＝****
→M000＝1000 0000 0000 0000 B
⑤ When R0100＝#0100, R0200＝ABCD, R0300＝****
→M000＝0100 0000 0000 0000 B
⑥ When R0100＝#0100, R0200＝0200, R0300＝#ABCD
→M000＝0010 0000 0000 0000 B

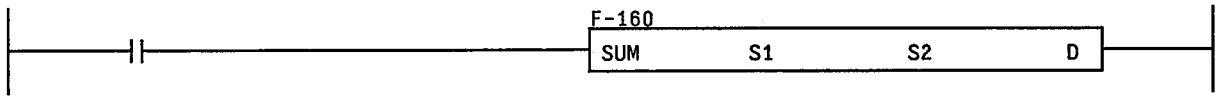
廃番

Expanded auxiliary instruction			Execution condition	Input ON	Pulse specification	○	Indirect specification	-
Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
SUN	FUN 160	Bit SUM	9 bytes		Overflow M901	-	Operation error M90E	-

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
S1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
S2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
D	-	-	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	-	R0600 to R0619

	imer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
S1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
S2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-
D	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	-	-

Ladder circuit example

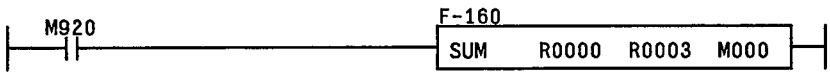


● Function description

This instruction counts the number of 1s in 16-bit data from S1 to S2 and stores the result in D when the line logic is ON. Both S1 and S2 must be in the same area.

● Description

【 Example 】



Number of 1 bits

R0000	0012	=0000	0000	0001	0010	··· 2
R0001	3456	=0011	0100	0101	0110	··· 7
R0002	ABCD	=1010	1011	1100	1101	··· 10
R0003	FF00	=1111	1111	0000	0000	··· 8
					Total	27 d=001B H

∴ M000 ← #001B H

Library instruction

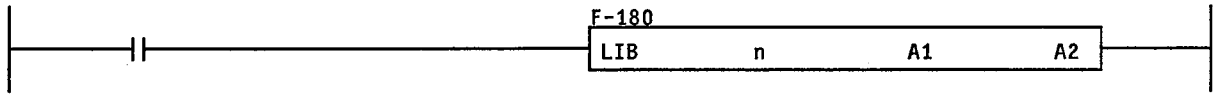
Execution condition	Input ON	Pulse specification	○	Indirect specification	-

Instruction	FUNCTION No.	Instruction name	Number of bytes used	Flags	Shift carry M900	-	Underflow M902	-
LIB	FUN 180	Library	9 bytes		Overflow M901	-	Operation error M90E	○

	Input X	OP link input X	Output Y	OP link output Y	Auxiliary M	Hold L	Special M	Input register	Output register
n	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
A1	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619
A2	X000 to X190	X200 to X290	Y000 to Y190	Y200 to Y290	M000 to M890	L000 to L490	-	R0500 to R0519	R0600 to R0619

	Timer/counter T/C	Register R	Link register P	Special register R/P	Constant #	Decimal constant K
n	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #000F	K0 to K15
A1	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767
A2	-	R0000 to R0499 R1000 to R4999	P0000 to P3799	-	#0000 to #FFFF	K-32768 to K32767

Ladder circuit example



● Function description

This instruction executes the library with the entry number (not a serial number) specified by "n".
When "n" specifies something other than a constant, the low-order 5 bits of the specified device become valid.
A1 and A2 are used as argument for library.

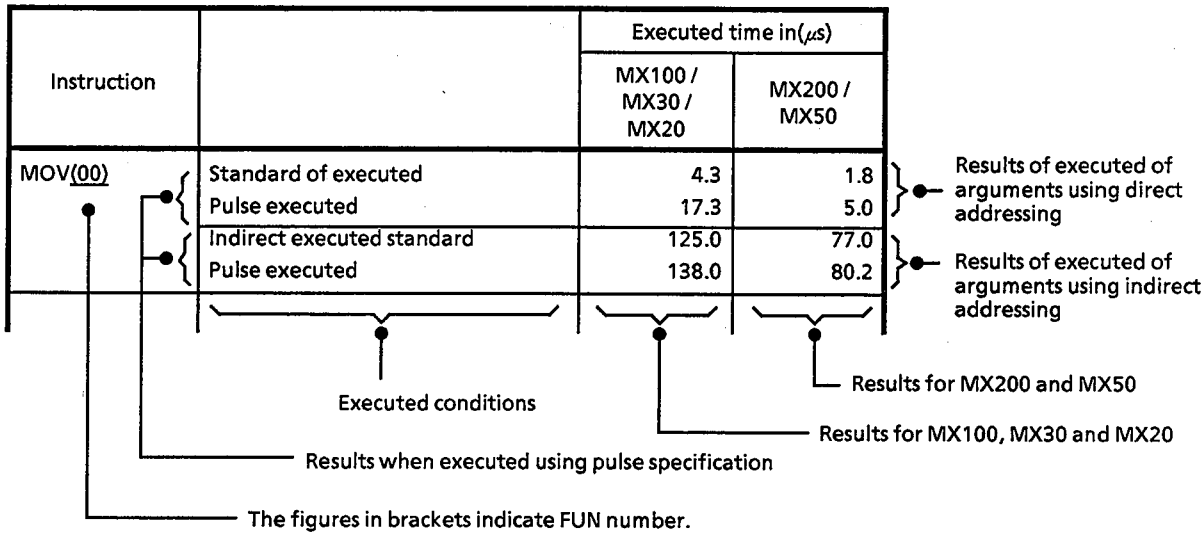
廃 番

APPENDIX

Instruction Executed Time

■ How to read the tables

【 Example 】



- The term "indirect executed" in the table indicates that the given result is obtained when all arguments for that instruction are executed through indirect addressing.
- Instruction FUN number 100 and beyond can be used only on the MX200 and MX50.
- The result of processing arguments consisting of instructions made up of variable number of words is indicated as "processing 10 words"and "processing 100 words"

■ Processing time when instructions are not processed

When an instruction does not meet the executed condition, it is not processed.

The processing time of the CPU at such times is shown below.

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
All instructions	When not executed	1.0	0.8
	Not executed Pulse specification	15.6	3.6

■ Processing time during pulse specification

Instructions that allow pulse specification require the time used for basic executed plus the time values given below when pulse specification is used.

- For MX100, MX30 and MX20 : +13.0[μs]
- For MX200 and MX50 : +3.2[μs]

Instruction		Executed time in (μs)	
		MX100 / MX30 / MX20	MX200 / MX50
LD	Basic instruction	2.6	1.4
	With ANB and ORB	9.3	5.5
	IN MC-MCR	9.6	6.4
	Using ANB and ORB in MC-MCR	11.6	7.2
LD NOT	Basic instruction	3.0	1.4
	With ANB and ORB	9.3	5.5
	IN MC-MCR	9.6	6.4
	Using ANB and ORB in MC-MCR	11.6	7.2
AND		2.6	1.4
AND NOT		2.6	1.4
OR	Basic instruction	2.6	1.4
	IN MC-MCR	9.6	6.5
OR NOT	Basic instruction	2.6	1.4
	IN MC-MCR	11.0	6.8
ANB		7.0	4.6
ORB		7.0	4.6
OUT		9.3	2.3
TMR	1.0s	45.0	17.0
	100ms	45.0	17.0
	10ms	45.3	17.0
CNT		45.6	18.0
LD=	Basic instruction	5.0	3.1
	With ANB and ORB	7.0	7.0
	IN MC-MCR	7.3	8.0
	Using ANB and ORB in MC-MCR	9.3	8.8
LD≠	Basic instruction	5.0	3.1
	With ANB and ORB	7.0	7.0
	IN MC-MCR	7.3	8.0
	Using ANB and ORB in MC-MCR	9.3	8.8
AND=		6.0	3.5
AND≠		6.0	3.5
OR=	Basic instruction	6.0	3.5
	IN MC-MCR	8.3	8.4
OR≠	Basic instruction	6.0	3.5
	IN MC-MCR	8.3	8.4
LD<	Basic instruction	3.3	1.4
	With ANB and ORB	5.3	5.4
	IN MC-MCR	5.6	6.3
	Using ANB and ORB in MC-MCR	7.6	7.1

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
LD>	Basic instruction	3.3	1.4
	With ANB and ORB	5.3	5.4
	IN MC-MCR	5.6	6.3
	Using ANB and ORB in MC-MCR	7.6	7.1
AND>		4.3	1.8
AND<		4.3	1.8
OR<	Basic instruction	4.3	1.8
	IN MC-MCR	6.6	6.7
OR>	Basic instruction	4.3	1.8
	IN MC-MCR	6.6	6.7
LD≦	Basic instruction	4.3	2.0
	With ANB and ORB	6.3	6.0
	IN MC-MCR	6.6	6.9
	Using ANB and ORB in MC-MCR	8.6	7.8

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
LD≧	Basic instruction	4.3	2.0
	With ANB and ORB	6.3	6.0
	IN MC-MCR	6.6	6.9
	Using ANB and ORB in MC-MCR	8.6	7.8
AND≦		5.3	2.5
AND≧		5.3	2.5
OR≦	Basic instruction	5.3	2.5
	IN MC-MCR	7.6	7.3
OR≧	Basic instruction	5.3	2.5
	IN MC-MCR	7.6	7.3
DLD=	Basic instruction	9.3	4.9
	With ANB and ORB	11.3	9.1
	IN MC-MCR	11.6	10.0
	Using ANB and ORB in MC-MCR	13.6	11.0
DLD≠	Basic instruction	9.3	4.9
	With ANB and ORB	11.3	9.1
	IN MC-MCR	11.6	10.0
	Using ANB and ORB in MC-MCR	13.6	11.0
DAND=		10.3	5.3
DAND≠		10.3	5.3
DOR=	Basic instruction	10.3	5.3
	IN MC-MCR	12.6	10.0
DOR≠	Basic instruction	10.3	5.3
	IN MC-MCR	12.6	10.0
DLD<	Basic instruction	7.6	3.3
	With ANB and ORB	9.6	7.5
	IN MC-MCR	10.0	8.4
	Using ANB and ORB in MC-MCR	12.0	9.2
DLD>	Basic instruction	7.6	3.3
	With ANB and ORB	9.6	7.5
	IN MC-MCR	10.0	8.4
	Using ANB and ORB in MC-MCR	12.0	9.2
DAND<		8.6	3.7
DAND>		8.6	3.7
DOR<	Basic instruction	8.6	3.7
	IN MC-MCR	11.0	8.8
DOR>	Basic instruction	8.6	3.7
	IN MC-MCR	11.0	8.8

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
DLD≦	Basic instruction	8.6	3.9
	With ANB and ORB	10.6	8.1
	IN MC-MCR	11.0	9.0
	Using ANB and ORB in MC-MCR	13.0	9.8
DLD≧	Basic instruction	8.6	3.9
	With ANB and ORB	10.6	8.1
	IN MC-MCR	11.0	9.0
	Using ANB and ORB in MC-MCR	13.0	9.8
DAND≦		9.6	4.3
DAND≧		9.6	4.3
DOR≦	Basic instruction	9.6	4.3
	IN MC-MCR	12.0	9.4
DOR≧	Basic instruction	9.6	4.3
	IN MC-MCR	12.0	9.4

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
MOV(00)	Normally executed	4.3	1.8
	Pulse executed	17.3	5.0
	Indirect Normally executed	125.0	77.0
	Pulse executed	138.0	80.2
BMOV(01)	Normally executed		
	10 words	156.3	33.0
	100 words	1263.3	217.0
	Indirect Normally executed		
	10 words	339.3	143.0
XCH(02)	Normally executed	22.6	11.0
	Pulse executed	35.6	14.2
	Indirect Normally executed	45.7	85.0
	Pulse executed	58.7	88.2
M4BT(03)	Normally executed	35.6	14.0
	Pulse executed	48.6	17.2
M8BT(04)	Normally executed	22.0	13.0
	Pulse executed	35.0	16.2
DMPX(05)	Normally executed	75.3	46.0
	Pulse executed	88.3	49.2
MPX(06)	Normally executed	76.3	47.0
	Pulse executed	89.3	50.2

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
DCPY(07)	Normally executed		
	Pulse executed		
	10 words	62.3	24.0
	100 words	485.3	172.0
STB(08)	Normally executed	72.6	27.0
BCD + (10)	Normally executed	78.3	33.0
	Pulse executed	91.3	36.2
	Indirect Normally executed	251.7	143.0
	Pulse executed	264.7	146.2
BCD – (11)	Normally executed	81.0	33.0
	Pulse executed	94.0	36.2
	Indirect Normally executed	254.3	143.0
	Pulse executed	267.3	146.2
BCD*(12)	Normally executed	704.7	199.0
	Pulse executed	717.7	202.2
BCD / (13)	Normally executed	549.0	204.0
	Pulse executed	562.0	207.3
BCDI(14)	Normally executed	56.0	24.0
	Pulse executed	69.0	27.2
BCDD(15)	Normally executed	57.0	24.0
	Pulse executed	70.0	27.2

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
DBC + (16)	Normally executed	136.3	55.0
	Pulse executed	149.3	58.2
DBC – (17)	Normally executed	140.3	55.0
	Pulse executed	153.3	58.2
DBC*(18)	Normally executed	2584.7	652.0
	Pulse executed	2597.7	655.2
DBC / (19)	Normally executed	4011.3	616.0
	Pulse executed	4024.3	619.2
PLS ↑ (20)	Normally executed	23.3	11.0
PLF ↓ (21)	Normally executed	23.0	11.0
INV(22)	Normally executed	1.0	1.0
SET(23)	Normally executed	4.3	1.3
RST(24)	Normally executed	4.6	1.4
	Normally not executed	1.0	0.8
STM(25)	Normally executed	42.0	22.0
UDC(26)	Normally executed	46.3	25.0
DCNT(27)	Normally executed	64.3	30.0
DUDC(28)	Normally executed	46.6	30.0
MC(30)	Normally executed	13.6	7.7
MCR(31)	Normally executed	28.0	15.0
JMP(32)	Normally executed	12.3	11.0
	Pulse executed	25.3	14.2

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
JPE(33)	Normally executed	0.0	0.0
LPS(34)	Normally executed	1.6	1.2
LRD(35)	Normally executed	2.6	0.9
LPP(36)	Normally executed	1.3	0.9
END(40)	Normally executed	3.6	3.1
RBP(41)	Normally executed	3.6	3.1
CALL(42)	Normally executed Pulse executed	22.0 35.0	11.0 14.2
SBR(43)	Normally executed	0.0	0.0
RET(44)	Normally executed	12.3	9.2
→BCD(50)	Normally executed Pulse executed	219.0 232.0	55.0 58.2
→BIN(51)	Normally executed Pulse executed	52.6 65.6	35.0 38.2
CPL(52)	Normally executed Pulse executed	13.6 26.6	7.5 10.7
4→16(53)	Normally executed Pulse executed	95.0 108.0	13.0 16.2
16→4(54)	Normally executed Pulse executed	116.3 129.3	39.0 42.2
7SEG(55)	Normally executed Pulse executed	42.8 55.3	16.0 19.2
TCMP(56)	Normally executed Pulse executed	201.7 214.7	60.0 63.2

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
→DBC(57)	Normally executed	779.3	130.0
	Pulse executed	792.3	133.2
→DBI(58)	Normally executed	935.7	109.0
	Pulse executed	948.7	112.2
→ASC(59)	Normally executed	41.0	23.0
	Pulse executed	54.0	26.2
SR(60)	Normally executed		
	10 words	140.7	65.0
	100 words	1040.7	368.0
SFL(61)	Normally executed of 1 bit		
	10 words	134.7	44.0
	100 words	1034.7	348.0
	Normally executed of 4 bits		
	10 words	498.7	153.0
	100 words	4098.7	370.0
	Normally executed of 8 bits		
	10 words	120.7	44.0
	100 words	867.7	347.0
	Normally executed of 16 bits		
	10 words	95.3	35.0
	100 words	662.3	246.0
SFR(62)	Normally executed of 1 bit		
	10 words	130.7	42.0
	100 words	1030.7	327.0
	Normally executed of 4 bits		
	10 words	491.3	145.0
	100 words	4091.3	1290.0
	Normally executed of 8 bits		
	10 words	117.0	44.0
	100 words	864.0	358.0
	Normally executed of 16 bits		
	10 words	92.3	32.0
	100 words	659.3	244.0

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
RLC(63)	Normally executed of 1 bit		
	10 words	145.3	48.0
	100 words	1045.3	352.0
	Normally executed of 4 bits		
	10 words	513.7	161.0
	100 words	4113.7	1380.0
	Normally executed of 8 bits		
	10 words	132.7	50.0
	100 words	879.7	353.0
	Normally executed of 16 bits		
	10 words	110.0	41.0
	100 words	677.0	253.0

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
RRC(64)	Normally executed of 1 bit		
	10 words	142.0	48.0
	100 words	1042.0	331.0
	Normally executed of 4 bits		
	10 words	506.3	153.0
	100 words	4106.3	1300.0
	Normally executed of 8 bits		
	10 words	129.0	50.0
	100 words	876.0	314.0
	Normally executed of 16 bits		
	10 words	107.3	39.0
	100 words	674.3	250.0
RL(65)	Normally executed of 1 bit		
	10 words	148.3	48.0
	100 words	1048.3	352.0
	Normally executed of 4 bits		
	10 words	512.3	161.0
	100 words	4112.3	1380.0
	Normally executed of 8 bits		
	10 words	130.7	49.0
	100 words	877.7	352.0
	Normally executed of 16 bits		
	10 words	108.0	40.0
	100 words	675.0	252.0
RR(66)	Normally executed of 1 bit		
	10 words	143.3	48.0
	100 words	1043.3	331.0
	Normally executed of 4 bits		
	10 words	505.0	152.0
	100 words	4105.0	1300.0
	Normally executed of 8 bits		
	10 words	129.7	49.0
	100 words	876.7	362.0
	Normally executed of 16 bits		
	10 words	107.7	37.0
	100 words	674.7	249.0
BIN + (70)	Normally executed	30.6	11.0
	Pulse executed	43.6	14.2
	Indirect Normally executed	204.0	121.0
	Pulse executed	217.0	124.2

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
BIN – (71)	Normally executed	30.6	11.0
	Pulse executed	43.6	14.2
	Indirect Normally executed	204.0	121.0
	Pulse executed	217.0	124.2
BIN*(72)	Normally executed	95.00	19.0
	Pulse executed	108.0	22.2
BIN / (73)	Normally executed	363.0	23.0
	Pulse executed	376.0	26.2
BINI(74)	Normally executed	26.0	10.0
	Pulse executed	39.0	13.2
BIND(75)	Normally executed	26.0	10.0
	Pulse executed	39.0	13.2
DBI + (76)	Normally executed	46.3	15.0
	Pulse executed	59.3	18.2
DBI – (77)	Normally executed	46.3	15.0
	Pulse executed	59.3	18.2
DBI*(78)	Normally executed	336.7	46.0
	Pulse executed	349.7	49.2
DBI / (79)	Normally executed	2015.7	260.0
	Pulse executed	2028.7	263.2
WAND(80)	Normally executed	20.3	9.4
	Pulse executed	33.3	12.6
	Indirect Normally executed	193.7	120.0
	Pulse executed	206.7	123.2
WOR(81)	Normally executed	20.3	9.4
	Pulse executed	33.3	12.6
	Indirect Normally executed	193.7	120.0
	Pulse executed	206.7	123.2

Instruction		Executed time in(μs)	
		MX100 / MX30 / MX20	MX200 / MX50
WXOR(82)	Normally executed	20.3	9.4
	Pulse executed	33.3	12.6
	Indirect Normally executed	193.7	120.0
	Pulse executed	206.7	123.2
WXNR(83)	Normally executed	18.6	9.6
	Pulse executed	31.6	12.6
WTCS(90)	Normally executed	41.0	20.0
	Pulse executed	54.0	23.2
RTCS(91)	Normally executed	4.3	1.8
	Pulse executed	17.3	5.0
WTCA(92)	Normally executed	41.0	20.0
	Pulse executed	54.0	23.2
RTCA(93)	Normally executed	4.3	1.8
	Pulse executed	17.3	5.0
STOP(94)	Normally executed	-	-
WDT(95)	Normally executed	4.3	12.0
ISS(96)	Normally executed *	Approx. 1.0 ms	
	Pulse executed *	Approx. 1.0 ms	
IOS(97)	Normally executed *	Approx. 1.5 ms	
	Pulse executed *	Approx. 1.5 ms	
DTBL(98)	This is a definition statement which does not have an executed speed.		
RTBL(99)	Normally executed		
	10 words	242.3	95.0
	100 words	1349.3	311.0

* Depends on the I/O configuration.

Instruction executed time(max)

Instruction		Executed time in(μs) MX200 / MX50
RLD(100)	Basic instruction With ANB and ORB IN MC-MCR Using ANB and ORB in MC-MCR	1.4 5.5 6.4 7.2
RLDN(101)	Basic instruction With ANB and ORB IN MC-MCR Using ANB and ORB in MC-MCR	1.4 5.5 6.4 7.2
RAND(102)		1.4
RADN(103)		1.4
ROR(104)	Basic instruction IN MC-MCR	1.4 6.5
RORN(105)	Basic instruction IN MC-MCR	1.4 6.8
ROUT(108)		2.3
WTBL(110)	10 words 100 words	97.0 382.0
RSIN(115)	Basic instruction Pulse executed	32.0 35.2
RSOT(116)	Basic instruction Pulse executed	31.0 34.2
WSOT(117)	Basic instruction Pulse executed	30.0 33.2
SNOD(120)	This is a declarative statement which does not have an executed speed.	
SXRS(121)		
SXWS(122)		
SCPY(130)	10 words 100 words	117.0 587.0
SCAT(131)	10 words 100 words	165.0 1250.0
SCMP(132)	10 words 100 words	147.0 794.0
SCMP(133)	10 words 100 words	153.0 799.0

Instruction executed time(max.)

Instruction		Executed time in(μs)
		MX200 / MX50
SLEN(134)	10 words 100 words	50.0 245.0
SMID(135)	10 words 100 words	120.0 595.0
ATOI(136)	4 words 16 words	462.0 953.0
ITOA(137)	4 words 16 words	369.0 1360.0
ITSA(138)	4 words 16 words	366.0 1330.0
ATOB(140)	4 words 16 words	397.0 715.0
BTOA(141)	4 words 16 words	143.0 454.0
CBTA(142)	10 words 100 words	174.0 486.0
SRCH(144)	10 words 100 words	131.0 520.0
CMID(145)	10 words 100 words	125.0 817.0
CITA(147)	4 words 16 words	377.0 1360.0
CISA(148)	4 words 16 words	372.0 1100.0
GFLG(150)	Basic instruction	13.0
EFLG(151)	Basic instruction	13.0
SUM(160)	10 words 100 words	165.0 1580.0
LIB(180)	Time for processing library opening and closing (does not include library processing time)	+110.0

廃 番

Revision History

Printed Date	Manual Number	Edition	Revised pages	Description
94-12	**CP-UM-1563E**	**1st Edition**		
98-11		**2nd Edition**		Company name changed MX50 and MX20 were added. Register 3 digits were changed to 4 digits.
			Chapter4.	Renew reference page
			4-2	Argument and Bit count information adde to the "TMR", "CNT" "Compare symmboles" and "Pouble word compare symbols" instruction items.
			4-20	Argument and Bit count information adde to the "RTBL"
			5-5 to 5-9	Old page 5-6 to 5-10
			5-10 to 5-34	Old page 5-12 to 5-36
			5-35	Old page 5-38
			5-36	Old page 5-40
			5-37	Old page 5-42
			5-38 to 5-59	Old page 5-44 to 5-65
			5-60 to 5-66	Old page 5-66 to 5-73
			5-67 to 5-69	Old page 5-74 to 5-76
			5-70 to 5-90	Old page 5-78 to 5-99
			5-91 to 5-93	Old page 5-100 to 5-101
			5-94 to 5-108	Old page 5-102 to 5-116
			5-109	MOV 0001 written in the example ladder diagram corrected to MOV 1 Old page 5-117
			5-110 to 5-146	Old page 5-118 to 5-154
			5-147	Notes was added. Old page 5-155
			5-148	Old page 5-156
			5-149	Note was added. Old page 5-157
			5-150	Old page 5-158
			5-151	Note was added. Old page 5-159
			5-152	Old page 5-160
			5-153	Note was added. Old page 5-161
			5-154 to 5-156	Old page 5-162 to 5-164
			5-157	1) Communication frame was changed. Old page 5-165
			5-158	Old page 5-166
			5-159	1) Communication frame was changed. Old page 5-167
			5-160 to 5-174	Old page 5-170 to 5-185
			5-176 to 5-178	Old page 5-186 to 5-189
			5-179	Old page 5-190 to 5-191
			5-180 to 5-183	Old page 5-192 to 5-195
			5-184	Old page 5-196 to 5-197
			5-185	Old page 5-198 to 5-199
			5-186 to 5-191	Old page 5-200 to 5-205
02-11		**3rd Edition**		restrictions on use changed

廃 番

Specifications are subject to change without notice.

YAMATAKE

Yamatake Corporation
Control Products Division
Head office : Totate International Building
2-12-19 Shibuya Shibuya-ku Tokyo 150-8316 Japan
Inquiries to : International Business Division
Phone : 81-3-3486-2331, Fax : 81-3-3486-2300 (Sales)
Phone : 81-466-20-2307, Fax : 81-466-27-9264 (Customer Service)
http://www.yamatake.com

Printed in Japan.
1st Edition: Issued in Dec., 1994
3rd Edition: Issued in Nov., 2002(N)

This has been printed on recycled paper.