HITACHI PROGRAMMABLE CONTROLLER

H-SERIES

CPU MODULE

INSTRUCTION MANUAL
(SOFTWARE EDITION)

TYPE : CPU2-20H | CPU2-07H | CPU2-03H
CPUP-20H | CPUP-07H | CPUP-03H
CPU-20Ha | CPU-07Ha | CPU-03Ha

NOTICE : Make sure to have this manual available to the person directly responsible for use and maintenance of this unit. After installation and initialization, file the manual for future reference.

Hitachi, Ltd.

NB323AX

Warranty Period and Scope of Warranty

The warranty period of the delivered Module is one (1) year after delivery to the place specified by the purchaser. Should any failure occur during this warranty period during use under normal working conditions within the range of the product specification in accordance with the instructions given in this instruction manual, repair or replacement of the failed portion(s) will be made at no cost to the purchaser.

However, if said failure occurred due to any of the following factors, it will be excluded from the scope of this warranty.

(1) The failure was caused by improper handling and/or use by the user.

(2) The failure was caused by an external factor.

(3) The failure was caused by modification or repair made by other than the supplier.

(4) the failure was caused by natural calamities or disasters for which the supplier is not responsible.

The warranty in this case menas warranty of the delivered article alone, and damage induced by a failure of the delivered article will not be covered by the warranty.

Repair for Value

Investigation and repair will be made for value in all cases after said warranty period has elapsed. Even during the warranty period, repair and investigation for the cause of the failure (except for the case where it is covered by warranty) for the reasons outside of the scope of warranty stated above will be made for value by your dealer.

Placement of Orders for Parts and Inquiries

Kindly inform the following particulars to your dealer on occurrence of a failure to the product, for placement of orders for parts and/or for other inquiries.

(1) Type

(2) Manufacturing No. (MFG. No.)

(3) Description of failure

The contents of this manual may be modified without previous notice.

CONTENTS

CONTENTS – (cont'd)

CONTENTS – (cont'd)

CONTENTS – (cont'd)

C A U T I O N

This software manual includes Application commands, Transfer commands and FUN instructions of H-SERIES CPU modules.

But, this software manual does not include Basic command and Arithmetic command.

So, please refer to "HI-LADDER/HI-COMMAND PROGAMMING MANUAL" (Manual No.:NB9913X) for the Basic command and Arithmetic command of H-SERIES CPU modules.

INTRODUCTION

1. I/O SIGNALS

1.1 About I/O Signals

There are three types of I/O signals.

(1) External I/O area

The PC transfers signals to and from external devices via I/O modules.

(2) Internal I/O area

Data is transferred in registers in the PC.

(3) CPU link area

Data is transferred to and from another CPU at CPU link time.

For these three signals, data is handled in bits, words (16 bits), or double words (32 bits).

Table 1.1 H-series PC I/O Table

		Bit data	Word data	Double-word data
External I/O area	Input	X □□□□□	WX □□□□	DX □□□□
	Output	Y □□□□□	WY □□□□	DY □□□□
Internal I/O area		R □□□□	-	-
		-	WR □□□□	DR □□□□
		M □□□□	WM □□□	DM □□□
CPU line area		L □□□□□	WL □□□□	DL □□□□

- o An I/O number is assigned to □ □ □ □ in the above table. For external I/O assignment (including remote external I/O assignment), see Section 1.5, "Rules for External I/O Assignment."
- o Bit data, word data, and double-word data on the same line in the above table have mutual relationships.
- o The area for X, WX, and DX are shared by bit data and word data. This is also true for the area for Y, WY and DY, the area for M, WM and DM, and the area for L, WL and DL.

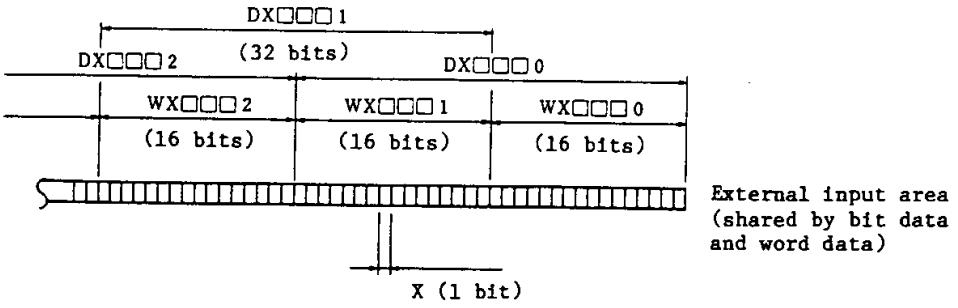


Figure 1.1 Relationship between X, WX, and DX

(This figure also applies to relationship between Y, WY, and DY, relationship between M, WM, and DM, and relationship between L, WL, and DL.)

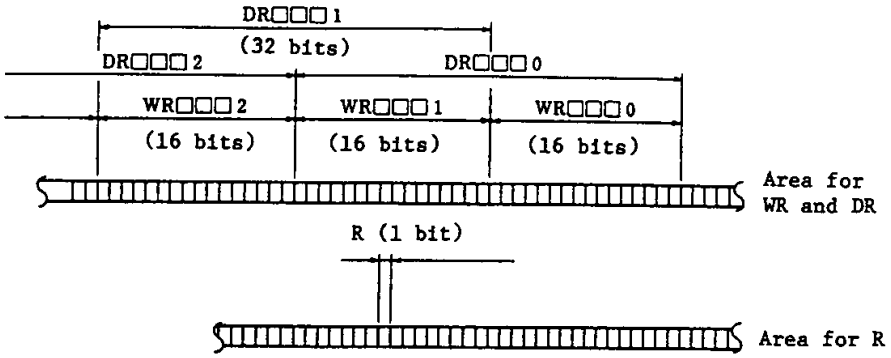


Figure 1.2 Relationship between R, WR, and DR

- o R and the WR and DR pair use different areas. In the area for WR, bit manipulation is not allowed.
- o To perform bit manipulation in the internal output area, use the area for M, WM, and DM shared by bit data and word data.
- o If a system using the CPU link module (see Section 2.4, "CPU Link System" in the hardware manual) is not used, the CPU link area (L, WL, and DL) can be used as the internal output area shared by bit data and word data.

1.2 I/O Assignment

Explanation of columns in the table below

Size column:
- B: Bit
- W: Word (16 points)
- D: Double-word (32 points)

Point count column: This column indicates the maximum number of points that can be used when only bits are used. In word notation, the column indicates the maximum number of words that can be used when only words are used.

I/O assignment: This column indicates the assignable range. For external I/O assignment, see Section 1.5, "Rules for External I/O Assignment."

Table 1.2 I/O Assignment

Function		Symbol	Size	Name	Point count	I/O assignment	Remarks
External I/O	External I/O	X	B	External bit input	4,096 points (256 words)	X 0 u s b	u: Unit number (0 to 5) s: Slot number (0 to A in hexadecimal) b: Intra-module bit number (00 to 95 in decimal) m: Intra-module word number (0 to 7. For DX and DY, however, 0 to 6) Storage is not possible during power outage.
		Y	B	External bit output		Y 0 u s b	
		W X	W	External word input		WY 0 u s m	
		W Y	W	External word output		WY 0 u s m	
		D X	D	External double-word input		DX 0 u s m	
		D Y	D	External double-word output		DY 0 u s m	
	Remote external I/O	X	B	Remote external bit input	2,048 (128 words) (Up to 512 points for one host station)	Y r St s b	r: Remote host station number (1 to 4) St: Remote substation number (0 to 9) s: Slot number (0 to 9) b: Intra-module bit number (00 to 95 in decimal) m: Intra-module word number (0 to 7. For DX and DY, however, 0 to 6) Storage is not possible during power outage.
		Y	B	Remote external bit output		Y r St s b	
		W X	W	Remote external word input		WY r St s m	
		W Y	W	Remote external word output		WY r St s m	
		D X	D	Remote external double-word input		DX r St s m	
		D Y	D	Remote external double-word output		DY r St s m	

Function		Symbol	Size	Name	Point count	I/O assignment	Remarks
CPU link area		L	B	Bit CPU link area 1	16,384 points (1,024 words)	L 0 - 3FFF	Numbers are in hexadecimal. Storage is not possible during power outage. When the CPU changes from the stop state to the run state, however, storage is not cleared.
		W L	W	Word CPU link area 1		WL 0 - 3FF	
		D L	D	Double-word CPU link area 1		DL 0 - 3FE	
		L	B	Bit CPU link area 2	16,384 points (1,024 words)	L 10000 - 13FFF	
		W L	W	Word CPU link area 2		WL 1000 - 13FF	
		D L	D	Double-word CPU link area 2		DL 1000 - 13FE	
Internal output	Internal bit output	R	B	Internal bit output	1,984 points	R 0 - 7BF	Storage is possible during power outage. Numbers are in hexadecimal.
		R	B	Internal special bit output	64 points	R7C 0 - 7FF	Storage is always possible during power outage. Numbers are in hexadecimal.
	Internal word output	W R	W	Internal word output	a. 1,024 words	WR 0 - 3FF	a. RAM*-04H, RAM*-08H, b. RAM*-16H, ROM*-16H, c. RAM*-48H, ROM2-48H. An asterisk (*) is omitted or replaced with 2 or 3. Numbers are in hexadecimal. Storage is possible during power outage. If a range exceeding the capacity on the left is read, 0 is assumed.
						DR 0 - 3FE	
					b. 17,408 words	WR 0 - 43FF	
		D R	D	Internal double-word output		DR 0 - 43FE	
					c. 50,176 words	WR 0 - C3FF	
						DR 0 - C3FF	
		W R	W	Internal special word output	512 words	WR F000 - F1FF	Storage is always possible during power outage. Numbers are in hexadecimal.
		D R	DF	Internal special double-word output		DR F000 - F1FE	
	Data area	M	B	Bit data area	16,384 points (1,024 words)	M 0 - 3FF	Storage is possible during power outage. Numbers are in hexadecimal.
		W M	W	Word data area		WM 0 - 3FF	
		D M	D	Double-word data area		DM 0 - 3FE	
Others	Edge detection	DIF	B	Rise edge detection	512 points	DIF 0 - 511	Storage is possible during power outage. Numbers are in decimal. (Duplicated numbers cannot be used except for DIF0 and DFN0.)
		DFN	B	Fall edge detection	512 points	DFN 0 - 511	

Function		Symbol	Size	Name	Point count	I/O assignment	Remarks
Others	Master control	MCS	B	Master control set	50 points	MCS 0 - 49	Numbers are in decimal.
		MCR	B	Master control reset		MCR 0 - 49	
	Timer counter	T D	B	On-delay timer	512 points (256 points or less for the timer)	TD 0 - 255	Storage is possible during power outage. Numbers are in decimal.
		S S	B	Single-shot timer		SS 0 - 255	
		WDT	B	Watchdog timer		WDT 0 - 255	Use CTU, CTD, and CT up/down counters in combination.
		M S	B	Monostable timer		MS 0 - 255	
		TMR	B	Accumulation timer		TMR 0 - 255	The same timer counter number must not be used twice or more.
		C U	B	Up counter		CU 0 - 511	
		RCU	B	Ring counter		RCU 0 - 511	Only timers 0 to 63 can use 0.01 s as the time base.
		CTU	B	Up/down counter increment		CTU 0 - 511	
		CTD	B	Up/down counter decrement		CTD 0 - 511	
		C T	B	Up/down counter output		CT 0 - 511	
		W T	-	Wait timer (HI-FLOW)		WT 0 - 255	
		P T	-	Parallel timer (HI-FLOW)		PT 0 - 255	
		C N	-	Loop timer (HI-FLOW)		CN 0 - 511	
		C L	B	Elapsed count clear	512 points	CL 0 - 511	
		T C	W	Timer counter elapsed time	512 words	TC 0 - 511	
	Constant	-	W	Decimal word constant		0 - 65 535	Singed numbers from -32,768 to 32,767
		-	D	Decimal double-word constant		0 - 4 294 967 295	Signed numbers from -2,147,483,648 to 2,147,483,647
		H	W	Hexadecimal word constant		H0000 - HFFFF	
		H	D	Hexadecimal double-word constant		H00000000 - HFFFFFFFF	
		-	B	Bit constant		0 or 1	

1.3 I/O Storage during Power Outage

Usually, internal outputs are cleared when the PC starts operation. The internal outputs listed in Table 1.3 are retained unless the maximum number of points stored during power outage is exceeded. For operations, see the instruction manual for GPCL and PGM peripherals.

Table 1.3 I/O Storage during Power Outage

I/O type	Maximum number of points stored during power outage	I/O range	Remarks
Internal bit output	1,984 points	R000 - 7BF	
Internal word output	1,024 words	WR0000 - 03FF	RAM-04H, -08H RAM2-04H, -08H RAM3-08H
	17,408 words	WR0000 - 43FF	RAM-16H, ROM-16H RAM2-16H, RAM3-16H RAM2-16H, ROM2-16H
	50,176 words	WR0000 - C3FF	RAM-48H, ROM2-48H RAM2-48H, RAM3-48H
Bit and word shared internal output	16,384 points (1,024 words)	M0000 - 3FFF (WM000 - 3FF)	
Rise edge detection	512 points	DIF 0 - 511	
Fall edge detection	512 points	DFN 0 - 511	
Timer counter	512 points	TD 0 - 511	Elapsed time indicated by the timer counter. TC0 to TC511 are also retained.

1.4 Internal Special Outputs

(1) Functions

Internal special outputs are divided into internal special bit outputs (R7C0 to R7FF) and internal special word outputs (WRF000 to WRF1FF). They have the following functions and purposes:

Table 1.4 Functions of Internal Special Outputs

Function	Purpose	Example
[1] Control the system status.	Permit execution of CPU functions.	Remote run enabled (R7C3) Debug enabled (R7C5)
	Set conditions to start and stop CPU operation.	Continued processing during cycle time over (R7C0)
[2] Record the system status.	Record CPU error types and details of errors.	I/O information mismatching (R7CD) I/O information mismatching details (WRF002)
	Record the system status.	Self-diagnostic error (R7DB) Self-diagnostic error details (WRF000)

In item [1] above, the CPU always checks the contents of internal special outputs during operation. For this reason, the user can forcibly set or reset internal special output data to permit execution of CPU functions, set operation conditions, and perform other functions.

In item [2] above, the CPU always records the system operation status in internal special outputs. The user can check the system status by monitoring the appropriate internal special output.

Notes

1) Check the set and reset conditions before setting data in internal special outputs. Particularly, do not set data in the areas where the system sets data.

2) Do not use the internal special bit outputs (R7C0 to R7FF) and internal special word outputs (WRF000 to WRF1FF) as normal outputs (coils and arithmetic boxes).

(2) Internal special outputs

I/O No.	Contents	ON	OFF
R7C0	Continued processing during cycle time over (normal scan)	U	U
R7C1	Continued processing during cycle time over (periodic scan)		
R7C2	Continued processing during cycle time over (interrupt scan)		
R7C3	Remote Run enabled		
R7C4	Remote Stop enabled		
R7C5	Debug enabled		
R7C6	Simulation enabled		
R7C7	Enabled modification during run		
R7C8	Severe error	S	
R7C9	Sequence processor error		
R7CA	User memory error		
R7CB	PI/O bus error		
R7CC	Too large memory size		
R7CD	I/O module I/O information mismatching		
R7CE	Communication module I/O information mismatching		
R7CF	(Undefined)	-	-
R7D0	Remote error	S	U
R7D1	Cycle time over error (normal scan)		
R7D2	Cycle time over error (periodic scan)		
R7D3	Cycle time over error (interrupt scan)		
R7D4	Syntax or assemble error		
R7D5	I/O module error		
R7D6	Excessive point assignment		
R7D7	Communication module error		
R7D8	System bus error		
R7D9	Battery error		
R7DA	Instantaneous power outage detection		
R7DB	Self-diagnostic error		
R7DC	(Undefined)		
R7DD	Excessive communication module assignment		
R7DE	Link module error		
R7DF	Operation enabled at HI-FLOW assembler error	U	
R7E0	Mode key switch in STOP position	S	S
R7E1	Mode key switch in REMOTE position		
R7E2	Mode key switch in RUN position		
R7E3	Single scan ON after RUN		
R7E4	Always ON		X
R7E5	0.02-s clock		S
R7E6	0.1-s clock		
R7E7	1-s clock		
R7E8	Occupancy flag		
R7E9	Disabled RUN		
R7EA	Modification during RUN		
R7EB	Segment display clear	U	U
R7EC	Internal special erroneous output clear		
R7ED	I/O reset during HI-FLOW debug RUN		
R7EE	(Undefined)	-	-
R7EF	(Undefined)		
R7F0	Carry flag (CY)	S	S
R7F1	Overflow (V)		
R7F2	Shift data (SD)		
R7F3	Calculation error (ERR)		
R7F4	Data error (DER)		
R7F5 ⟆ R7F7	(Undefined)	-	-
*R7F8	Request to read calendar clock	U	S
*R7F9	Request to set calendar clock		
*R7FA	Request to adjust calendar clock		
*R7FB	Incorrect calendar clock setting	S	U
*R7FC	Trigger condition matching flag		S
*R7FD	Trace monitor flag		
*R7FE *R7FF	(Undefined)	-	-

The user must control U. S is controlled by the system.
The user must not control S.

* Valid when CPU2-**H is combined with RAM2-**H, RAM3-**H, or ROM2-**H.

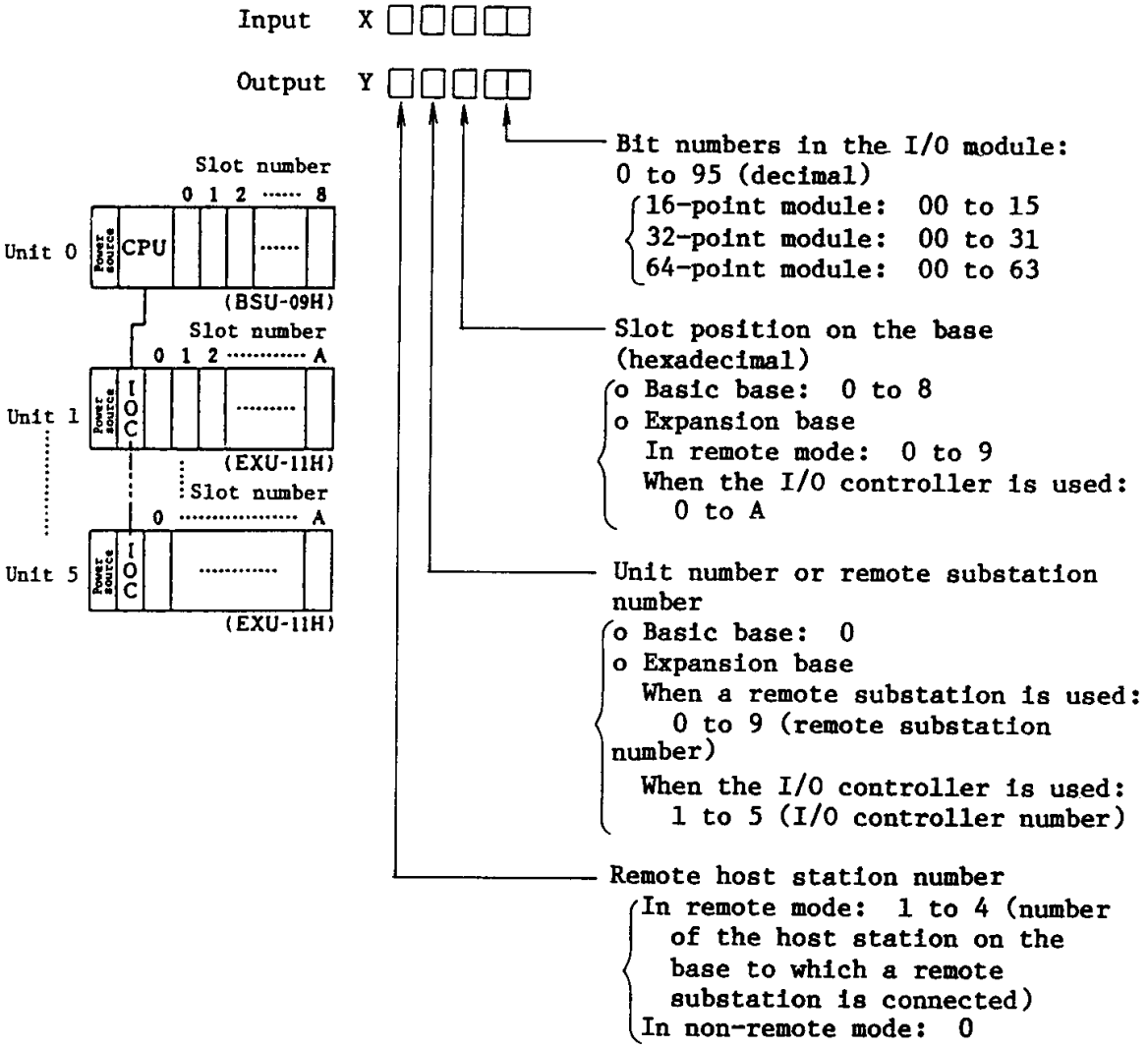
I/O No.	Contents	ON	OFF
WRF000	Self-diagnostic error code	S	U
WRF001	Syntax and assembler error details		
WRF002	I/O module I/O information mismatching details		
WRF003	Communication module I/O information mismatching details		
WRF004	Invalid communication module slot number		
WRF005	Invalid I/O module slot number		
WRF006	Invalid remote I/O slot number		
WRF007	Invalid link module slot number		
WRF008 ⟆ WRF00A	(Undefined)	-	-
*WRF00B	Calendar clock reading (year)	S	S
*WRF00C	Calendar clock reading (month, day)		
*WRF00D	Calendar clock reading (week)		
*WRF00E	Calendar clock reading (hour, minute)		
*WRF00F	Calendar clock reading (second)		
WRF010	Maximum scan time		
WRF011	Current scan time		
WRF012	Minimum scan time		
WRF013	CPU status		
WRF014	Internal word output capacity		
WRF015	Calculation error code		U
WRF016	Residue register (low-order)		S
WRF017	Residue register (upper-order)		
WRF018	Communication module startup flag		
WRF019	BASIC status		
WRF01A	(Undefined)	-	-
*WRF01B	Calendar clock reading or setting (year)	S	S
*WRF01C	Calendar clock reading or setting (month, day)		
*WRF01D	Calendar clock reading or setting (week)		
*WRF01E	Calendar clock reading or setting (hour, minute)		
*WRF01F	Calendar clock reading or setting		
WRF020 WRF021 ⟆ WRF030 WRF031	Communication module slot-0 status ⟆ Communication module slot-8 status		
WRF032 ⟆ WRF03F	(Undefined)	-	-
WRF040 WRF041 WRF042 ⟆ WRF049 WRF04A WRF04B	Member registration area 1 ⟆ Member registration area 4	S	S
WRF04C WRF04D WRF04E	Debug registration area		
WRF04F ⟆ WRF07F	(Undefined)	-	-
WRF080 ⟆ WRF097	Remote host-1 error flag	S	S
WRF098 ⟆ WRF0AF	Remote host-2 error flag		
WRF0B0 ⟆ WRF0C7	Remote host-3 error flag		
WRF0C8 ⟆ WRF0DF	Remote host-4 error flag		
WRF0E0 ⟆ WRF13F	Link-1 error flag		
WRF140 ⟆ WRF19F	Link-2 error flag		
WRF1A0 ⟆ WRF1FF	(Undefined)	-	-

1.5 Rules for External I/O Assignment

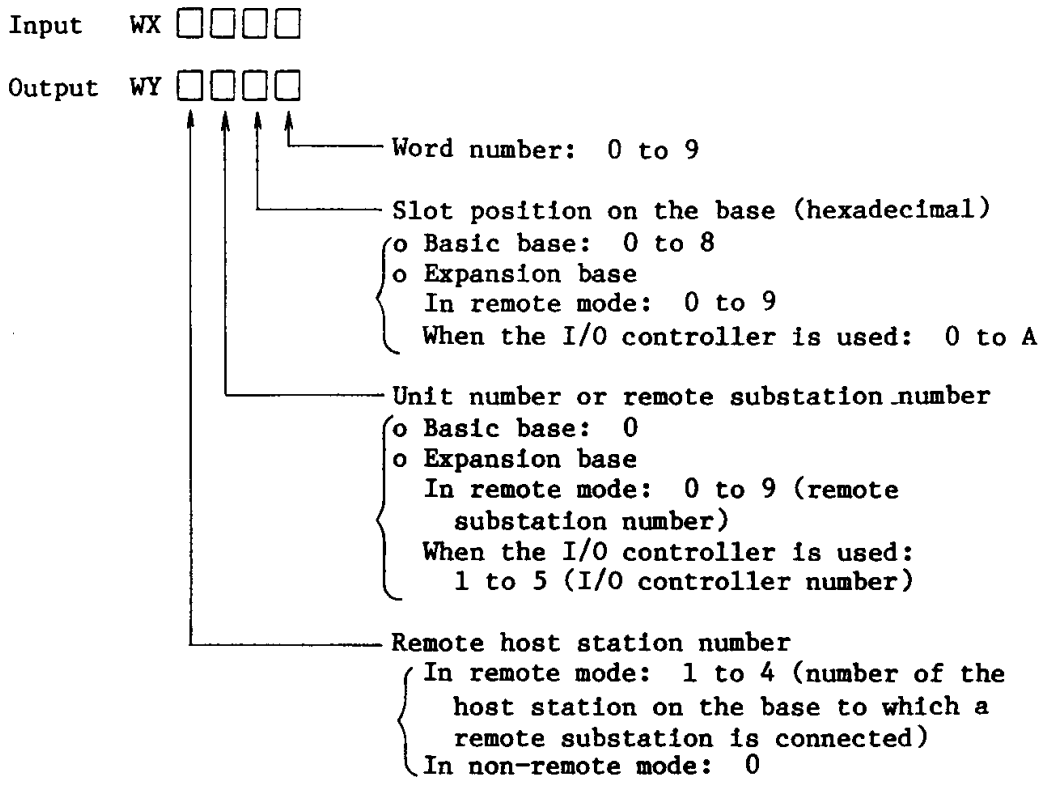
External outputs consist of X and Y in bits, WX and WY in words (16 bits), and DX and DY in double-words (32 bits).

These external outputs are assigned according to types of inputs and outputs, classification as to whether the I/O is basic, extended or remote I/O, the slot position, and bit and word numbers in the module.

(a) Bit I/O number



(b) Word I/O number



Bit number in the I/O module	Word number
00 - 15	0
16 - 31	1
32 - 47	2
48 - 63	3
64 - 79	4
80 - 95	5
[96 - 111]*	6
[112 - 127]*	7

* Bits 96 to 127 cannot be accessed in bit units. They can be accessed only in word units.

(c) Double-word I/O number

Input DX □□□□

Output DY □□□□

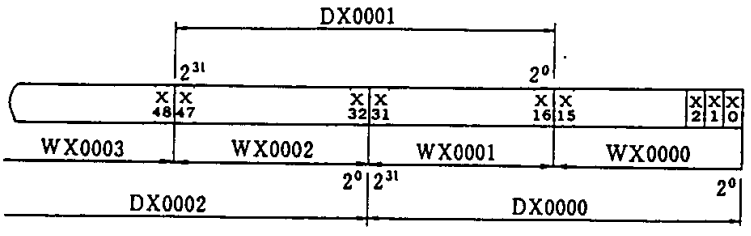
- Smaller of consecutive word numbers: 0 to 6
- Slot position on the base (hexadecimal)
  - o Basic base: 0 to 8
  - o Expansion base
    - In remote mode: 0 to 9
    - When the I/O controller is used: 0 to A
- Unit number or remote substation number
  - o Basic base: 0
  - o Expansion base
    - In remote mode: 0 to 9 (remote substation number)
    - When the I/O controller is used: 1 to 5 (I/O controller number)
- Remote host station number
  - In remote mode: 1 to 4 (number of the host station on the base)
  - In non-remote mode: 0

Bit number	Word number	Double-word number
00 - 15	0	0
16 - 31	1	1
32 - 47	2	2
48 - 63	3	3
64 - 79	4	4
80 - 95	5	5
[96 - 111]*	6	6
[112 - 127]*	7	

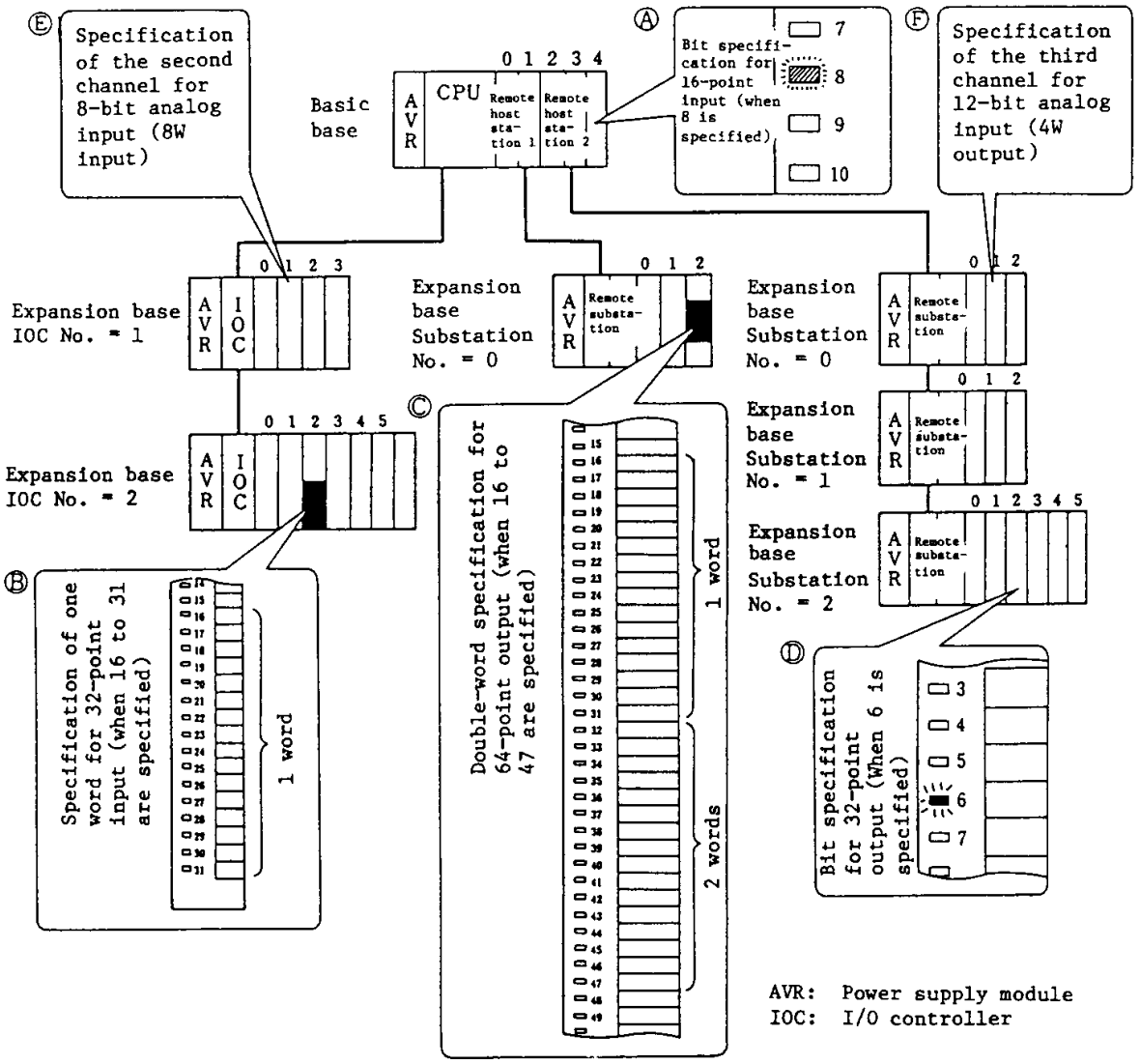
* Bits 96 to 127 cannot be accessed in bit units. They can be accessed only in word units.

(d) Relationship between bits, words, and double-words (areas shared by bits and words)

In the following example, X, WX, and DX are used.



The following system is used to show an example of external I/O assignment at individual sections.



Ⓐ X 0 0 4 0 8
- 8 bits
- Fourth slot
- Basic base

Basic base

Ⓑ WX 0 2 2 1
- Word number (bits 16 to 31)
- Second slot
- I/O controller number

No remote substation is used on the expansion base.

Ⓒ DY 1 0 2 1
- Smaller of consecutive word numbers
- Second slot
- Remote substation number

Remote host station number

Ⓓ Y 2 2 2 0 6
- 6 bits
- Second slot
- Remote substation number

Remote host station number

Ⓔ WX 0 1 1 2
- Word number (channel number)
- First slot
- I/O controller number

No remote substation is used on the expansion base.

Ⓕ WY 2 0 1 3
- Word number (channel number)
- First slot
- Remote substation number

Remote host station number

Figure 1.3 Example of External I/O Assignment

[1] Application Command

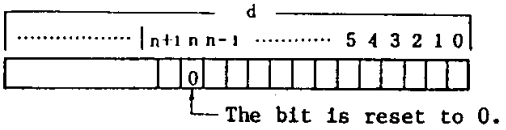
[Name] Bit reset

Ladder format	Condition code					Processing time (μs)				Remarks
BRES (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	7.2 8.7	←	13.8 17.1	←	
Command format	No. of steps					H-2002		H-702/302		
BRES (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		6.0 7.2	←	11.5 14.2	←	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O with bits reset						o	o	o		o	o		
n	Bit position to be reset					o	o	o	o				o	The constant is specified in decimal.

[Function]

- The "n"th bit of the I/O (word or double word) specified by d is set to 0.
- The contents of the other bits are not changed.



When d indicates a word:

- The bit position is specified by the contents (0 to 15) of the low-order 4 bits (b3 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One bf 0 to 15 can be specified as n (constant). (Decimal)

When d indicates a double word:

- The bit position is specified by the contents (0 to 31) of the low-order 5 bits (b4 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 31 can be specified as n (constant). (Decimal)

[Precautions]

[Name] Bit test

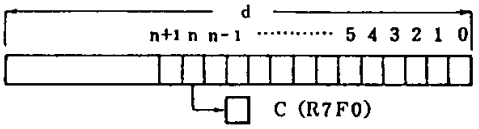
Ladder format	Condition code					Processing time (μs)				Remarks
BTS (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	↕	8.2 10.0	←	16.0 19.6	←	
Command format	No. of steps					H-2002		H-702/302		
BTS (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		6.8 8.3	←	13.3 16.3	←	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be tested						o	o	o		o	o		
n	Bit position to be tested					o	o	o	o				o	The constant is specified in decimal.

[Function]

. The contents of the "n"th bit of the I/O (word or double word) specified by d are checked. When the contents are 1, C (R7F0) is set to 1. When the contents are 0, C (R7F0) is reset to 0.

. The contents of d are not changed.



When d indicates a word:

. The bit position is specified by the contents (0 to 15) of the low-order 4 bits (b3 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)

. One of 0 to 15 can be specified as n (constant). (Decimal)

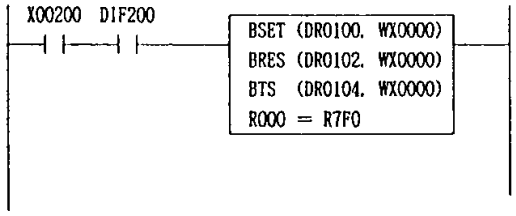
When d indicates a double word:

. The bit position is specified by the contents (0 to 31) of the low-order 5 bits (b4 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)

. One of 0 to 31 can be specified as n (constant). (Decimal)

[Precautions]

[Program example]




```
LD X00200
AND DIF200
[
BSET (DR0100, WX0000)
BRES (DR0102, WX0000)
BTS (DR0104, WX0000)
R000 = R7F0
]
```

[Program explanation]

When WX0000 = H1234 at the leading edge of X00200 (WX0000 = 0001001000110100)
20 (decimal)

Assuming that DR0100 = H00000000, DR0102 = HFFFFFFFF, and DR0104 = H5555AAAA are set, at the leading edge of X00200:
the 20th bit of DR0100 is set to 1 by BSET,

```
       b31 ── b20 ──────── b0
DR0100=00000000000000000000000000000000
              ↑
    This bit is set to 1.
```

the 20th bit of DR0102 is reset to 0 by BRES, and

```
       b31 ── b20 ──────── b0
DR0102=11111111111111111111111111111111
              ↑
    This bit is reset to 0.
```

the 20th bit of DR0104 is checked by BTS.

```
       b31 ── b20 ──────── b0
DR0104=01010101010101011010101010101010
              ↑
    This bit is checked.
    Since the 20th bit is 1, C (R7F0) is set to 1.
```

BTS

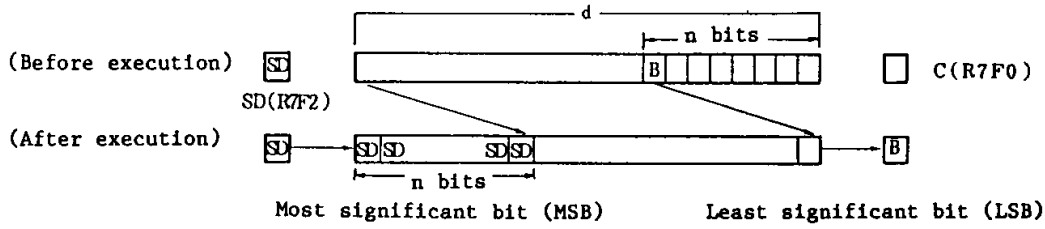
[Name] Shift right

Ladder format	Condition code					Processing time (μs)				Remarks
SHR (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	↕	16.6	18.1	33.4	36.7	
						27.2	33.1	55.7	68.5	
Command format	No. of steps					H-2002		H-702/302		
SHR (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		13.8	15.1	27.8	30.6	
						22.7	27.6	46.4	57.1	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be shifted						o	o	o		o	o		
n	No. of bits to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

- The contents of d are shifted right (low-order direction) n bit positions.
- The SD (R7F2) contents are set in n bits from the most significant bit.
- The contents of the "n"th bit from the least significant bit are set in C (R7F0).



When d indicates a word:

- The shift amount is specified by the contents (0 to 15) of the low-order 4 bits (b3 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 15 can be specified as n (constant). (Decimal)

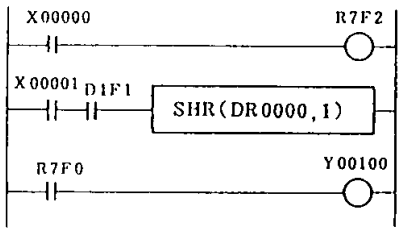
When d indicates a double word:

- The shift amount is specified by the contents (0 to 31) of the low-order 5 bits (b4 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 31 can be specified as n (constant). (Decimal)

[Precautions]

. When n = 0, the contents of d are not shifted. C holds the previous status.

[Program example]



X00000 ····· Input of a defective To SD

X00001 ····· Conveyor movement

Y00100 Output of a defective carry

```
LD  X00000
OUT R7F2
LD  X00001
AND DIF1
[
SHR (DR0000, 1)
]
LD  R7F0
OUT Y00100
```

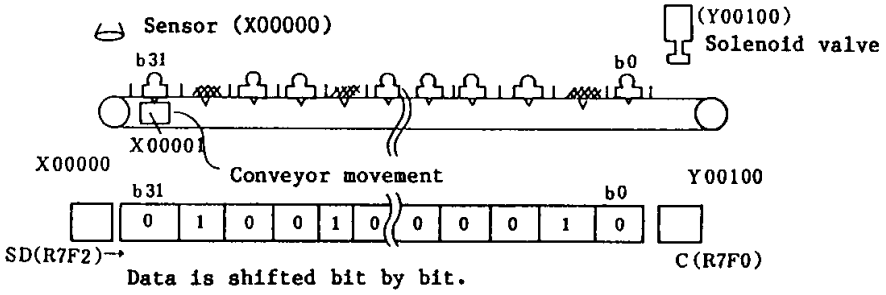
[Program explanation]

. A conveyor with 32 stands moves to the right.

. Whenever one stand moves to the right, 1 pulse is supplied to X1.

. A sensor is mounted at the left end of the conveyor. When a defective is put on the conveyor, X00000 is turned ON.
The X00000 (sensor input) and X00001 (conveyor movement) signals are as follows:

. When the conveyor moves to the right, data is shifted bit by bit. When the data is outputted to the carrier (at the right end of the conveyor), the (Y00100) solenoid valve is turned ON and the defective is ejected.



SHR

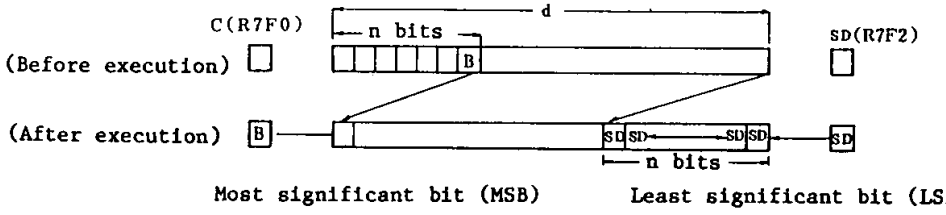
[Name] Shift left

Ladder format	Condition code					Processing time (μs)				Remarks
SHL (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	↕	16.6	18.1	33.4	36.7	
						27.2	33.1	55.7	68.5	
Command format	No. of steps					H-2002		H-702/302		
SHL (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		13.8	15.1	27.8	30.6	
						22.7	27.6	46.4	57.1	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be shifted						o	o	o		o	o		
n	No. of bits to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

- The contents of d are shifted left (high-order direction) n bit positions.
- The SD (R7F2) contents are set in n bits from the least significant bit.
- The contents of the "n"th bit from the most significant bit are set in C (R7F0).



When d indicates a word:

- The shift amount is specified by the contents (0 to 15) of the low-order 4 bits (b3 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 15 can be specified as n (constant). (Decimal)

When d indicates a double word:

- The shift amount is specified by the contents (0 to 31) of the low-order 5 bits (b4 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 31 can be specified as n (constant). (Decimal)

[Precautions]

- When n (shift amount) is 0, the contents of d are not shifted. C holds the previous status.

[Name] Rotate right

Ladder format	Condition code					Processing time (μs)				Remarks
ROR (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	↕	15.9	19.6	32.4	40.6	
						25.0	33.0	51.4	69.0	
Command format	No. of steps					H-2002		H-702/302		
ROR (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		13.2	16.3	27.0	33.8	
						20.8	27.5	42.8	57.5	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be rotated						o	o	o		o	o		
n	No. of bits to be rotated					o	o	o	o				o	The constant is specified in decimal.

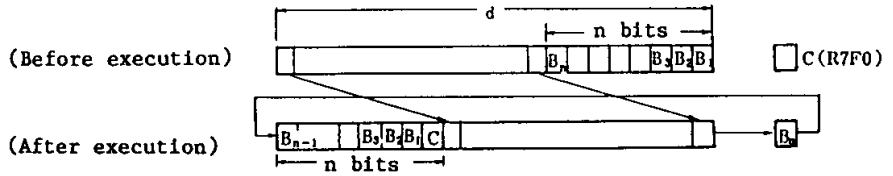
[Function]

. The contents of d are rotated right (low-order direction) n bit positions.

. The contents of C (R7F0) are inputted into the most significant bit and the contents of the least significant bit are inputted into C (R7F0). This processing is repeated n times.

. The contents of C (R7F0) are set in the "n"th bit from the most significant bit.

. The contents of the "n"th bit from the least significant bit are set in C (R7F0).



Most significant bit (MSB) Least significant bit (LSB)

When d indicates a word:

. The shift amount is specified by the contents (0 to 15) of the low-order 4 bits (b3 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)

. One of 0 to 15 can be specified as n (constant). (Decimal)

When d indicates a double word:

. The shift amount is specified by the contents (0 to 31) of the low-order 5 bits (b4 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)

. One of 0 to 31 can be specified as n (constant). (Decimal)

[Precautions]

. When n (rotation amount) is 0, the contents of d are not rotated. C holds the previous status.

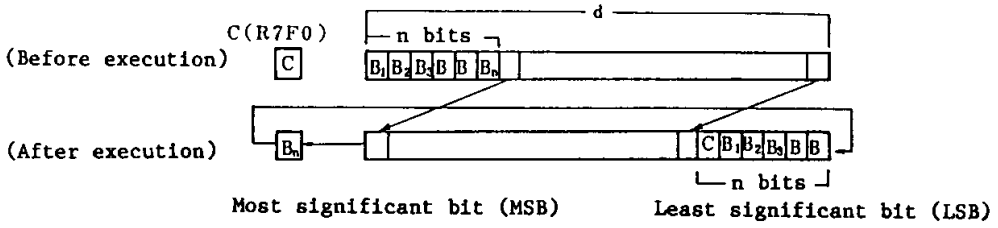
[Name] Rotate left

Ladder format	Condition code					Processing time (μs)				Remarks
ROL (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	↕	15.9	19.6	32.4	40.6	
						25.0	33.0	51.4	69.0	
Command format	No. of steps					H-2002		H-702/302		
ROL (d, n)	Conditions		Step			Average	Maximum	Average	Maximum	
	-		3			13.2	16.3	27.0	33.8	
						20.8	27.5	42.8	57.5	

Usable I/O		Bit			Word				Double word			Constant	Other
		X Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be rotated					o	o	o		o	o		
n	No. of bits to be rotated				o	o	o	o				o	The constant is specified in decimal.

[Function]

- The contents of d are rotated left (high-order direction) n bit positions.
- The contents of C (R7F0) are set in the "n"th bit from the most significant bit.
- The contents of the "n"th bit from the least significant bit are set in C (R7F0).



When d indicates a word:

- The shift amount is specified by the contents (0 to 15) of the low-order 4 bits (b3 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 15 can be specified as n (constant). (Decimal)

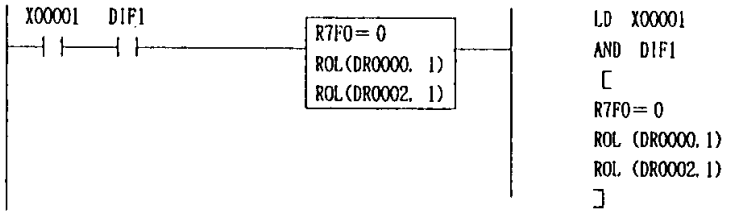
When d indicates a double word:

- The shift amount is specified by the contents (0 to 31) of the low-order 5 bits (b4 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 31 can be specified as n (constant). (Decimal)

[Precautions]

. When n (rotation amount) is 0, the contents of d are not rotated. C holds the previous status.

[Program example]

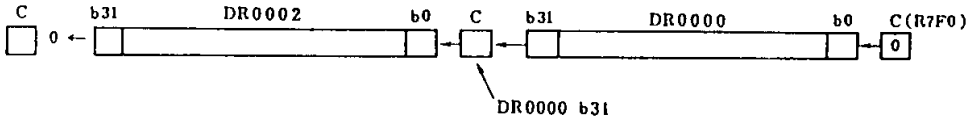



```
LD  X00001
AND DIF1
[
R7F0=0
ROL (DR0000, 1)
ROL (DR0002, 1)
]
```

[Program explanation]

. 64-bit data is shifted bit by bit at the leading edge of X00001. 0 is inputted in the shifted empty area.

Entire movement



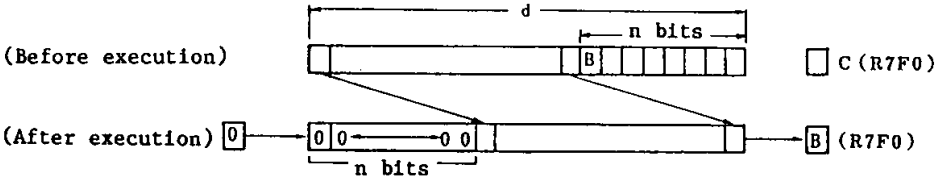
[Name] Logical shift right

Ladder format	Condition code					Processing time (μs)				Remarks
LSR (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	↕	14.1	15.3	28.8	31.5	
						24.0	29.7	49.5	62.1	
Command format	No. of steps					H-2002		H-702/302		
LSR (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		11.7	12.7	24.0	26.2	
						20.0	24.7	41.2	51.7	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be shifted						o	o	o		o	o		
n	No. of bits to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

- The contents of d are shifted right (low-order direction) n bit positions.
- 0 is set in n bits from the most significant bit.
- The contents of the "n"th bit from the least significant bit are set in C (R7F0).



When d indicates a word:

- The shift amount is specified by the contents (0 to 15) of the low-order 4 bits (b3 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 15 can be specified as n (constant). (Decimal)

When d indicates a double word:

- The shift amount is specified by the contents (0 to 31) of the low-order 5 bits (b4 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 31 can be specified as n (constant). (Decimal)

[Precautions]

- When n (shift amount) is 0, the contents of d are not shifted. C holds the previous status.

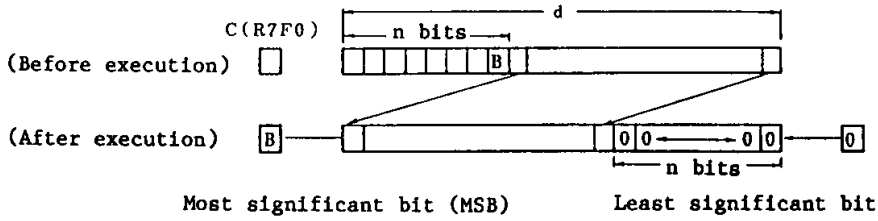
[Name] Logical shift left

Ladder format	Condition code					Processing time (μs)				Remarks
LSL (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	-	-	-	-	↕	14.1	15.3	28.8	31.5	
						24.0	29.7	49.5	62.1	
Command format	No. of steps					H-2002		H-702/302		
LSL (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		11.7	12.7	24.0	26.2	
						20.0	24.7	41.2	51.7	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be shifted						o	o	o		o	o		
n	No. of bits to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

- The contents of d are shifted right (low-order direction) n bit positions.
- 0 is set in n bits from the least significant bit.
- The contents of the "n"th bit from the most significant bit are set in C (R7F0).



When d indicates a word:

- The shift amount is specified by the contents (0 to 15) of the low-order 4 bits (b3 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 15 can be specified as n (constant). (Decimal)

When d indicates a double word:

- The shift amount is specified by the contents (0 to 31) of the low-order 5 bits (b4 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)
- One of 0 to 31 can be specified as n (constant). (Decimal)

[Precautions]

- When n (shift amount) is 0, the contents of d are not shifted. C holds the previous status.

[Name] BCD shift right

Ladder format	Condition code					Processing time (μs)				Remarks
BSR (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	8.2	8.7	16.6	17.7	
						17.1	21.1	35.1	43.9	
Command format	No. of steps					H-2002		H-702/302		
BSR (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		6.8	7.2	13.8	14.7	
						14.2	17.6	29.2	36.6	

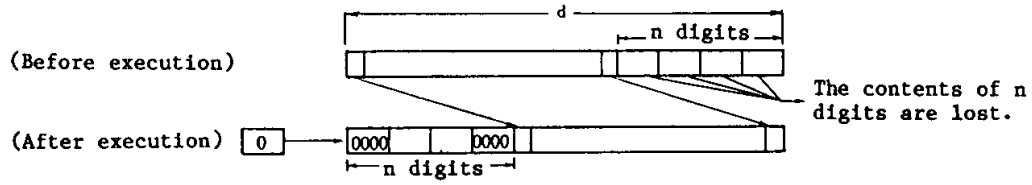
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be shifted						o	o	o		o	o		
n	No. of bits to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

. The contents of d are shifted right (low-order direction) n digit positions. (One digit is 4 bits long.)

. 0 is set in n digits from the high-order position.

. The contents of n digits from the low-order position are lost.



When d indicates a word:

. The shift amount is specified by the contents (0 to 3) of the low-order 2 bits (b1 and b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)

. One of 0 to 3 can be specified as n (constant). (Decimal)

When d indicates a double word:

. The shift amount is specified by the contents (0 to 7) of the low-order 3 bits (b2 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)

. One of 0 to 7 can be specified as n (constant). (Decimal)

[Precautions]

. When n (shift amount) is 0, the contents of d are not shifted.

[Name] BCD shift left

Ladder format	Condition code					Processing time (μs)				Remarks
BSL (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	8.2	8.7	16.6	17.7	
						17.1	21.1	35.1	43.9	
Command format	No. of steps					H-2002		H-702/302		
BSL (d, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		6.8	7.2	13.8	14.7	
						14.2	17.6	29.2	36.6	

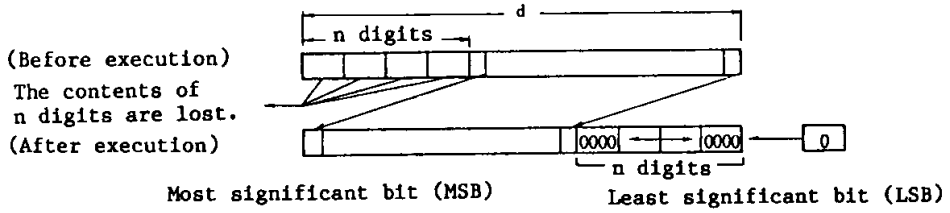
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be shifted						o	o	o		o	o		
n	No. of bits to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

. The contents of d are shifted left (high-order direction) n digit positions. (One digit is 4 bits long.)

. 0 is set in n digits from the low-order position.

. The contents of n digits from the high-order position are lost.



When d indicates a word:

. The shift amount is specified by the contents (0 to 3) of the low-order 2 bits (b1 and b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)

. One of 0 to 3 can be specified as n (constant). (Decimal)

When d indicates a double word:

. The shift amount is specified by the contents (0 to 7) of the low-order 3 bits (b2 to b0) of n (WX, WY, WR, WM, TC). (The high-order bits are ignored and assumed as 0.)

. One of 0 to 7 can be specified as n (constant). (Decimal)

[Precautions]

. When n (shift amount) is 0, the contents of d are not shifted.

[Name] Batch shift right (Shift right block)

Ladder format	Condition code					Processing time (μs)		Remarks
WSHR (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	The upper values are values when d indicates a bit and the lower values are values when d indicates a word. The contents of ( ) are values when d indicates WL.
	DER	ERR	SD	V	C	75.7 + 0.3 n	147.4 + 0.6 n	
	↕	.	.	.	.	66.0 + 3.9 n (6.8)	132.6 + 7.5 n (10.4)	
Command format	No. of steps					H-2002	H-702/302	
WSHR (d, n)	Conditions			Step		63.1 + 0.2 n	122.8 + 0.5 n	
	-			3		55.0 + 3.2 n (5.7)	110.5 + 6.2 n (8.7)	

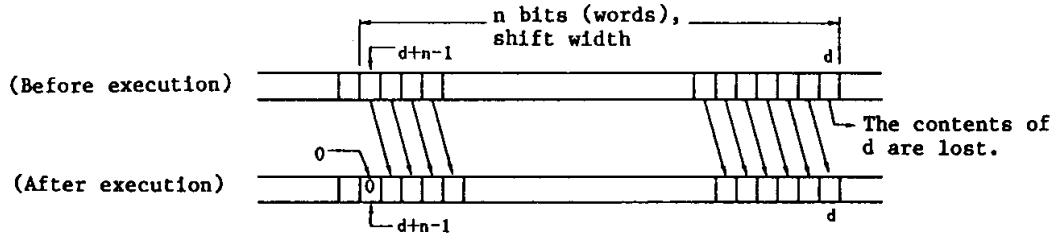
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O to be shifted			o				o						
n	No. of bits (words) to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

. n bits (words) from d to d+n-1 are shifted right (in the I/O number decreasing direction) one bit position.

. 0 (H0000) is set in the bit (word) of d+n-1.

. The contents of d are lost.



When n is one of WX, WY, WR, WL, WM, and TC:

The contents (0 to 255) of the low-order 8 bits (b7 to b0) of n are the number of bits (words) to be shifted.

When n is a constant:

One of 0 to 255 can be specified as the number of bits (words) to be shifted. (Decimal)

[Precautions]

. Keep d+n-1 within the I/O limits (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). When d+n-1 is beyond the limits, DER = 1 and the contents between d and the maximum limit are shifted.

. When n is 0, no batch shift is performed. DER (R7F4) is 0.

[Name] Batch shift left (Shift left block)

Ladder format	Condition code					Processing time (μs)		Remarks
WSHL (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	The upper values are values when d indicates a bit and the lower values are values when d indicates a word. The contents of ( ) are values when d indicates WL.
	DER	ERR	SD	V	C	75.7 + 0.3 n	147.4 + 0.6 n	
	↕	.	.	.	.	66.4 + 3.9 n (6.8)	133.7 + 7.5 n (10.4)	
Command format	No. of steps					H-2002	H-702/302	
WSHL (d, n)	Conditions			Step		63.1 + 0.2 n	122.8 + 0.5 n	
	-			3		55.3 + 3.2 n (5.7)	111.4 + 6.2 n (8.7)	

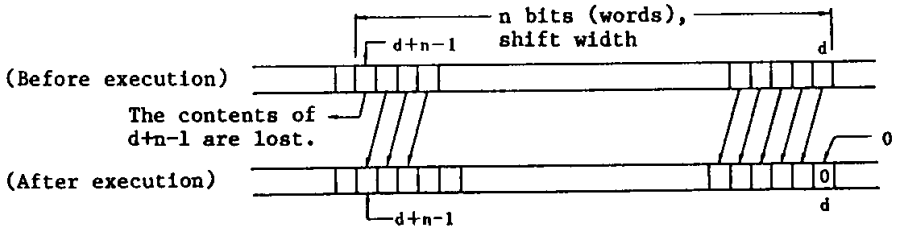
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O to be shifted			o				o						
n	No. of bits (words) to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

. n bits (words) from d to d+n-1 are shifted left (in the I/O number increasing direction) one bit position.

. 0 (H0000) is set in the bit (word) of d.

. The contents of d+n-1 are lost.



When n is one of WX, WY, WR, WL, and WM:

The contents (0 to 255) of the low-order 8 bits (b7 to b0) of n are the number of bits (words) to be shifted.

When n is a constant:

One of 0 to 255 can be specified as the number of bits (words) to be shifted. (Decimal)

[Precautions]

. Keep d+n-1 within the I/O limits (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). When d+n-1 is beyond the limits, DER = 1 and the contents between d and the maximum limit are shifted.

. When n is 0, no batch shift is performed. DER (R7F4) is 0.

[Name] Batch BCD shift right (BCD shift right block)

Ladder format	Condition code					Processing time (μs)		Remarks
WBSR (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	The contents of ( ) are values when d indicates WL.
	DER	ERR	SD	V	C			
	↕	.	.	.	.	64.5 + 6.7 n (9.6)	130.4 + 13.3 n (16.2)	
Command format	No. of steps					H-2002	H-702/302	
WBSR (d, n)	Conditions			Step				
	-			3		53.7 + 5.6 n (8.0)	107.7 + 11.1 n (13.5)	

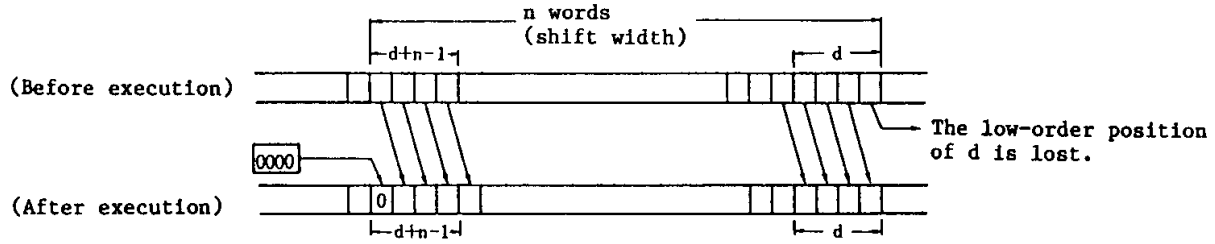
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O to be shifted							o						
n	No. of words to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

. n words from d to d+n-1 are shifted right (in the I/O number decreasing direction) one digit position as BCD data 4n digits long. (One digit is 4 bits long.)

. 0 is set in the high-order position of d+n-1.

. The low-order position of d is lost.



When n is one of WX, WY, WR, WL, and WM:

The contents (0 to 255) of the low-order 8 bits (b7 to b0) of n are the number of words to be shifted.

When n is a constant:

One of 0 to 255 can be specified as the number of words to be shifted. (Decimal)

[Precautions]

. Keep d+n-1 within the I/O limits (WRC3FF, WL3FF, WL13FF, WM3FF). When d+n-1 is beyond the limits, DER = 1 and the contents between d and the maximum limit are shifted.

. When n is 0, no batch shift is performed. DER (R7F4) is 0.

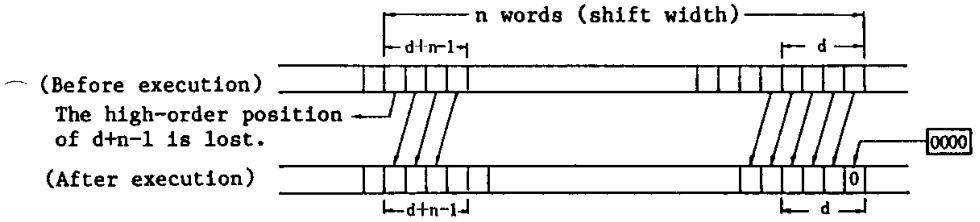
[Name] Batch BCD shift left (BCD shift left block)

Ladder format	Condition code					Processing time (μs)		Remarks
WBSL (d, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	The contents of ( ) are values when d indicates WL.
	DER	ERR	SD	V	C			
	↕	.	.	.	.	64.5 + 6.7 n (9.6)	130.4 + 13.3 n (16.2)	
Command format	No. of steps					H-2002	H-702/302	
WBSL (d, n)	Conditions			Step				
	-			3		53.7 + 5.6 n (8.0)	108.7 + 11.1 n (13.5)	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O to be shifted							o						
n	No. of words to be shifted					o	o	o	o				o	The constant is specified in decimal.

[Function]

- n words from d to d+n-1 are shifted left (in the I/O number increasing direction) one digit position as BCD data 4n digits long. (One digit is 4 bits long.)
- 0 is set in the low-order position of d+n-1.
- The high-order position of d+n-1 is lost.



When n is one of WX, WY, WR, WL, and WM:

The contents (0 to 255) of the low-order 8 bits (b7 to b0) of n are the number of words to be shifted.

When n is a constant:

One of 0 to 255 can be specified as the number of words to be shifted. (Decimal)

[Precautions]

- Keep d+n-1 within the I/O limits (WRC3FF, WL3FF, WL13FF, WM3FF). When d+n-1 is beyond the limits, DER = 1 and the contents between d and the maximum limit are shifted.
- When n is 0, no batch shift is performed. DER (R7F4) is 0.

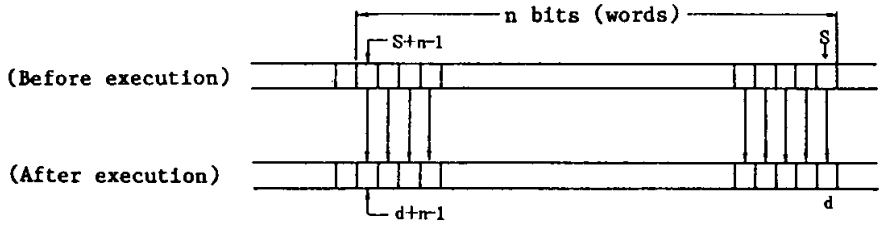
[Name] Block transfer (Move)

Ladder format	Condition code					Processing time (μs)		Remarks
MOV (d, S, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	The contents of ( ) are values when d and S indicate L and WL.
	DER	ERR	SD	V	C			
	↕	·	·	·	·	139.5 + 3.9 n (6.8)	276.3 + 7.5 n (10.4)	
Command format	No. of steps					H-2002	H-702/302	
MOV (d, S, n)	Conditions		Step					
	-		4			116.2 + 3.2 n (5.7)	230.2 + 6.2 n (8.7)	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O of transfer destination			o				o						
S	Top I/O of transfer source			o				o						
n	Number of bits (words) to be transferred					o	o	o	o				o	The constant is specified in decimal.

[Function]

- n bits (words) from S to S+n-1 are transferred to d+n-1.
- The values from S to S+n-1 are held. When the range of the transfer source is overlapped with that of the transfer destination, the transferred value is selected.



When n is one of WX, WY, WR, WL, and WM:

The contents (0 to 255) of the low-order 8 bits (b7 to b0) of n are the number of bits (words) to be transferred.

When n is a constant:

One of 0 to 255 can be specified as the number of bits (words) to be transferred. (Decimal)

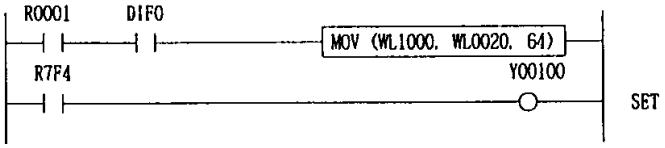
[Precautions]

. Keep d+n-1 and S+n-1 within the I/O limits (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). When d+n-1 and S+n-1 are beyond the limits, DER = 1 and the contents between d and the maximum limit are transferred.

. When n is 0, no batch transfer is performed. DER (R7F4) is 0.

[Program example]

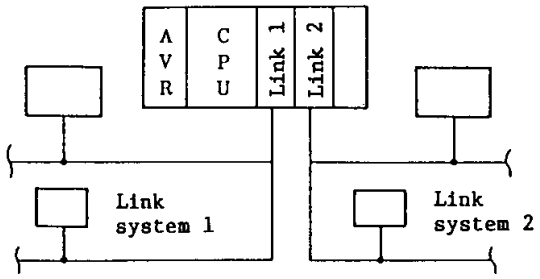
Data in the first link area (WL0020 to WL005F) is transferred to the second link area (WL1000 to WL103F).

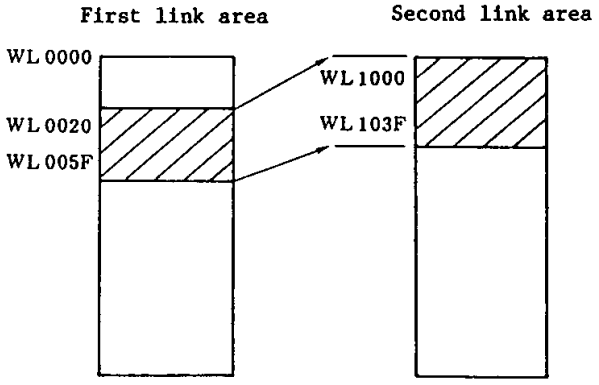



```
LD  R0001
AND DIF0
 [
MOV (WL1000, WL0020, 64)
 ]
LD  R7F4
SET Y00100
```

[Program explanation]

64-word data is transferred from link system 1 of the first link to link system 2 of the second link.
The transfer areas are WL0020 to WL005F and WL1000 to WL103F.





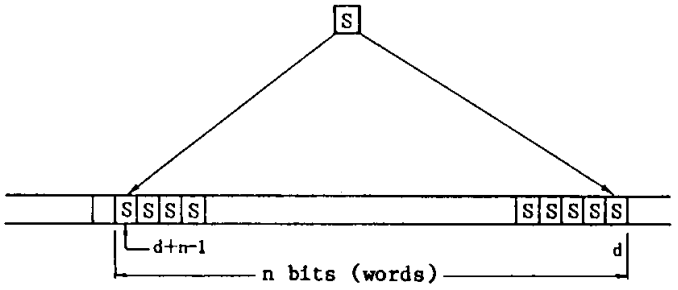
[Name] Copy

Ladder format	Condition code					Processing time (μs)		Remarks
COPY (d, S, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	The upper values are values when d indicates a bit and the lower values are values when d indicates a word. The contents of ( ) are values when d and S indicate L and WL.
	DER	ERR	SD	V	C	60.1 + 0.3 n	118.0 + 0.6 n	
	↕	.	.	.	.	68.8 + 2.6 n (3.9)	136.6 + 5.0 n (6.3)	
Command format	No. of steps					H-2002	H-702/302	
COPY (d, S, n)	Conditions			Step		50.1 + 0.2 n	98.3 + 0.5 n	
	-			4		57.3 + 2.2 n (3.2)	113.8 + 4.2 n (5.2)	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O of copy destination			o				o						
S	I/O of copy source	o	o	o		o	o	o	o				o	
n	Number of bits (words) to be copied					o	o	o	o				o	The constant is specified in decimal.

[Function]

- The value of S (bit, word) is copied from d to d+n-1.
- The value of S is held.
- Bits are copied in bits and words are copied in words.



When n is one of WX, WY, WR, WL, and WM:

The contents (0 to 255) of the low-order 8 bits (b7 to b0) of n are the number of bits (words) to be copied.

When n is a constant:

One of 0 to 255 can be specified as the number of bits (words) to be copied. (Decimal)

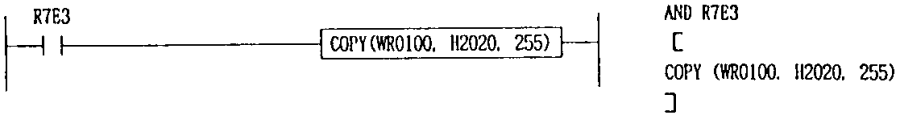
[Precautions]

. Keep d+n-1 within the I/O limits (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). When d+n-1 is beyond the limits, DER = 1 and the contents between d and the maximum limit are copied.

. When n is 0, no batch copy is performed. DER (R7F4) is 0.

[Program example]

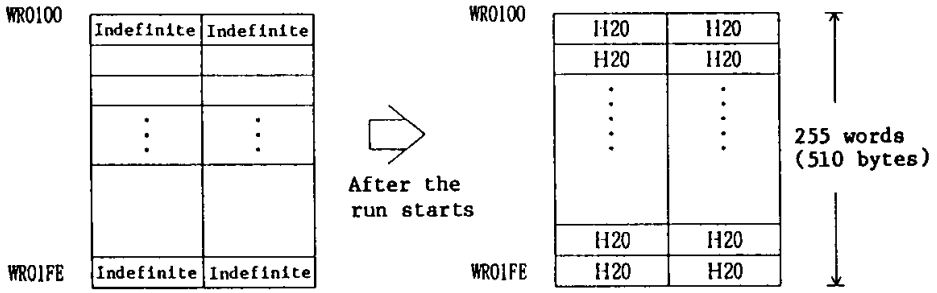
The default value (H2020) is set in the range from WR0100 to WR01FE.



[Program explanation]

The communication data area from WR0100 to WR01FE is filled with the space code (H20) as a default value at the first scan after the run starts.

R7E3: 1 scan ON after the run starts



COPY

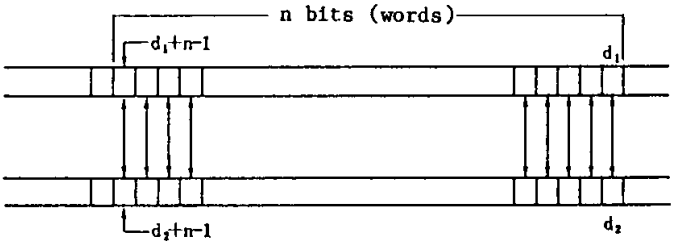
[Name] Block exchange (Exchange)

Ladder format	Condition code					Processing time (μs)		Remarks
XCG (d_1, d_2, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	The upper values are values when d_1 and d_2 indicate a bit and the lower values are values when d_1 and d_2 indicate a word. The contents of ( ) are values when d_1 and d_2 indicate L and WL.
	DER	ERR	SD	V	C	137.8 + 5.8 n (64.0 + 9.6 n)	273.1 + 10.6 n (129.3 + 16.2 n)	
	↕	·	·	·	·	138.0 + 6.3 n (12.1)	273.9 + 11.7 n (17.5)	
Command format	No. of steps					H-2002	H-702/302	
XCG (d_1, d_2, n)	Conditions			Step		114.8 + 4.8 n (53.3 + 8.0 n)	227.6 + 8.8 n (107.7 + 13.5 n)	
	-			4		115.0 + 5.2 n (10.1)	228.2 + 9.7 n (14.6)	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d_1	Top I/O of exchange destination			o				o						
d_2	I/O of exchange source			o				o						
n	Number of bits (words) to be exchanged					o	o	o	o				o	The constant is specified in decimal.

[Function]

- The contents of n bits (words) from dl to d_1+n-1 are exchanged with the contents of n bits (words) from d_2 to d_2+n-1.
- Bits are exchanged with bits and words are exchanged with words.



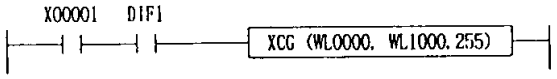
When n is one of WX, WY, WR, WL, and WM:

The contents (0 to 255) of the low-order 8 bits (b7 to b0) of n are the number of bits (words) to be exchanged.

When n is a constant:

One of 0 to 255 can be specified as the number of bits (words) to be exchanged. (Decimal) 10

Example:



WL0000 to WL00FE are exchanged with WL1000 to WL10FE.

[Precautions]

- Keep d+n-1 and S+n-1 within the I/O limits (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). When d+n-1 and S+n-1 are beyond the limits, DER = 1 and the contents up to the maximum limits of the numbers of bits (words) specified as d_1 and d_2 whichever smaller are exchanged.
- When n (block width) is 0, no batch exchange is performed. DER (R7F4) is 0.

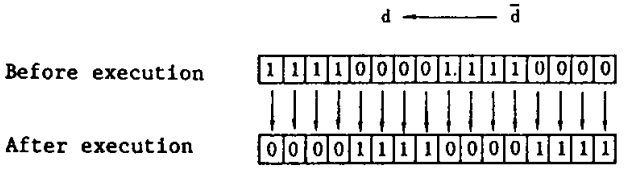
[Name] Inversion (NOT)

Ladder format	Condition code					Processing time (μs)				Remarks
NOT (d)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a bit or word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	4.4 6.3	←	8.6 11.7	←	
Command format	No. of steps					H-2002		H-702/302		
NOT (d)	Conditions		Step			Average	Maximum	Average	Maximum	
	-		2			3.7 5.2	←	7.2 9.7	←	

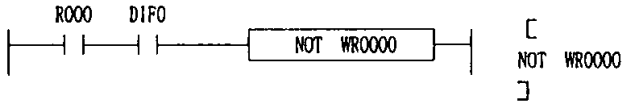
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be inverted		o	o			o	o			o	o		

[Function]

. The bits of the contents of d are inverted.



[Program example]



[Program explanation]

The contents of WR0000 are inverted at the leading edge of R000.

Example: When the instruction is executed when WR0000 is H1234, WR0000 = HEDCB. When the instruction is executed once again, WR0000 = H1234.

[Precautions]

. The start condition of this instruction should be Edge Trigger.

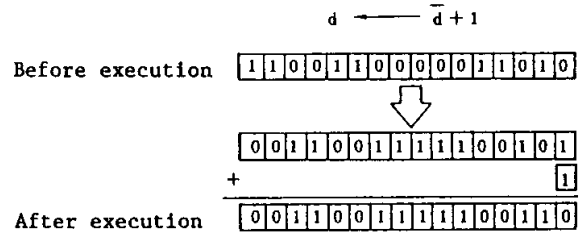
[Name] Twos complement (Negate)

Ladder format	Condition code					Processing time (μs)				Remarks
NEG (d)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a bit or word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	4.9 7.3	←	9.7 17.2	←	
Command format	No. of steps					H-2002		H-702/302		
NEG (d)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			2		4.1 6.1	←	8.1 14.3	←	

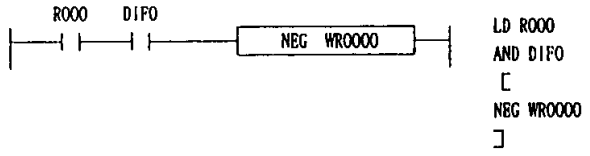
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be complemented		o	o			o	o			o	o		

[Function]

. The twos complement of d is calculated. (The bits of the contents of d are inverted and added with 1. C (R7F0) is not changed.



[Program example]



[Program explanation]

. The twos complement of the contents of WR0000 is obtained at the leading edge of R000.

Example: When the instruction is executed when WR0000 is H1234, WR0000 = HEDCC. When the instruction is executed once again, WR0000 = H1234.

[Precautions]

. The start condition of this instruction should be Edge Trigger.

[Name] Absolute value (Absolute)

Ladder format	Condition code					Processing time (μs)				Remarks
ABS (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a bit or word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	↕	7.5	←	14.6	←	
						12.5		25.5		
Command format	No. of steps					H-2002		H-702/302		
ABS (d, S)	Conditions			Step		Average	Maximum	Average	Maximum	
	Word			3		6.2	←	12.2	←	
	Double word			4		10.4		21.2		

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O after absolute value taken						o	o			o	o		
S	I/O before absolute value taken					o	o	o	o	o	o	o	o	

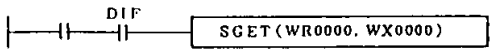
[Function]

. When S is positive or 0: The contents of S are set in d. C (R7F0) is 0.

. When S is negative: The twos complement of the contents of S is set in d. C (R7F0) is 1.

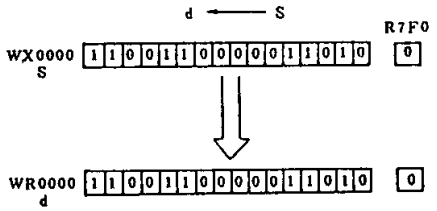
. Use words or double words for d and S.

Example:



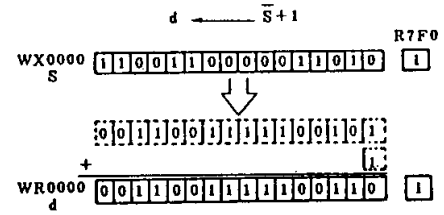
(When the value of WX is positive or 0
WX0000 = H4C1A

(When R7F0 is 0)



(When the value of WX is negative)
WX0000 = HCC1A

(When R7F0 is 7)



When S indicates a word: 0 to 65 535 (decimal), H0000 to HFFFF (hexadecimal)

When S indicates a double word: 0 to 4 294 967 295 (decimal), H00000000 to HFFFFFFFF (hexadecimal)

[Precautions]

. The start condition of this instruction should be Edge Trigger.

[Name] Sign addition (Sign get)

Ladder format	Condition code					Processing time (μs)				Remarks
SGET (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d and S indicate words and the lower values are values when d and S indicate double words.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	6.7	←	13.3	←	
						12.1		24.1		

Command format	No. of steps		H-2002		H-702/302	
SGET (d, S)	Conditions	Step	Average	Maximum	Average	Maximum
	Word	3	5.6	←	11.1	←
	Double word	4	10.1		20.1	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O after sign addition						o	o			o	o		
S	I/O before sign addition					o	o	o	o	o	o	o	o	

[Function]

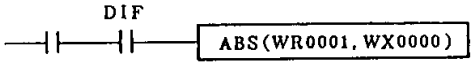
. When C (R7F0) is 0: The contents of S are set in d.

. When C (R7F0) is 1: The twos complement of the contents of S is set in d.

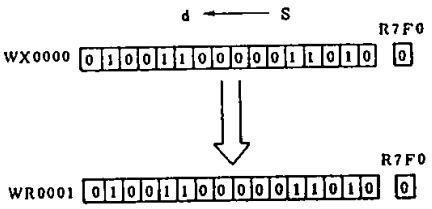
. The contents of C (R7F0) are not changed.

. Use words or double words for d and S.

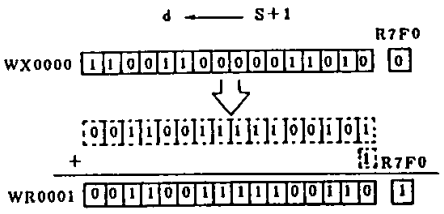
Example:



When C (R7F0) is 0



When C (R7F0) is 1



[Precautions]

. The start condition of this instruction should be Edge Trigger.

[Name] Sign extension

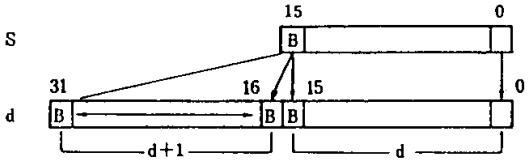
Ladder format	Condition code					Processing time (μs)			Remarks
EXT (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300	
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum
	.	.	.	.	.	8.8	←	17.2	←
Command format	No. of steps					H-2002		H-702/302	
EXT (d, S)	Conditions			Step		Average	Maximum	Average	Maximum
	-			3		7.3	←	14.3	←

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O after sign extension										o	o		
S	I/O before sign extension					o	o	o	o				o	

[Function]

. The sign bit (most significant bit) of S is extended to a high-order word of d.

. The low-order word of d is the contents of S.



[Program example]


```
LD  X00000
[
EXT (DR0100. WX0000)
]
```

[Program explanation]

. When X00000 is turned ON, the contents of WX0000 are extended to DR0100.

When WX00000 is positive or 0

Example: WX00000 = H7FFF (+32 767)
↓
DR0100 = H00007FFF (+32 767)

When WX00000 is negative

Example: WX00000 = H8000 (-32 768)
↓
DR0100 = HFFFF8000 (-32 768)

[Precautions]

[Name] Binary -> BCD conversion (BCD)

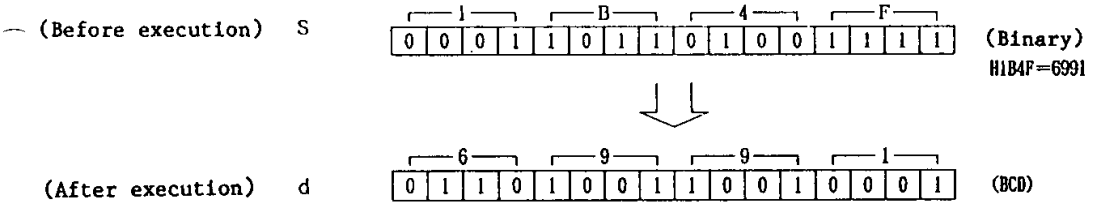
Ladder format	Condition code					Processing time (μs)				Remarks
BCD (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d and S indicate words and the lower values are values when d and S indicate double words.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	31.9	33.4	65.5	68.8	
						93.2	117.4	197.8	251.1	
Command format	No. of steps					H-2002		H-702/302		
BCD (d, S)	Conditions			Step		Average	Maximum	Average	Maximum	
	Word			3		26.6	27.8	54.6	57.3	
	Double word			4		77.7	97.8	164.8	209.2	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O (BCD) after conversion						o	o			o	o		
S	I/O (BIN) before conversion					o	o	o	o	o	o	o	o	

[Function]

- The contents of S are converted from a binary number to a BCD number and the result is outputted to d.
- When the conversion result of S is larger than the number of digits of the BCD data of d, DER (R7F4) is 1 and no conversion is executed.

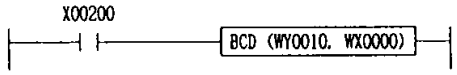
When S is a word: Set it as follows: H0000 ≦ S ≦ H270F (0 to 9999)
When S is a double word: Set it as follows: H00000000 ≦ S ≦ H05F5E0FF (0 to 9999999)



Combination of d and S

d	S
Word	Word
Double word	Double word

Example:



[Precautions]

- When a data error occurs, the contents of d are left unchanged.

BCD

[Name] BCD -> Binary conversion (Binary)

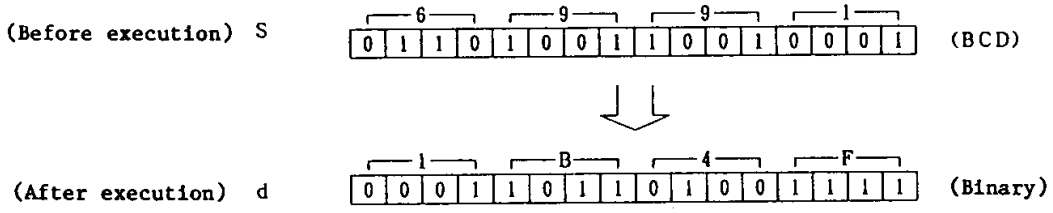
Ladder format	Condition code					Processing time (μs)				Remarks
BIN (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d and S indicate words and the lower values are values when d and S indicate double words.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	34.2	34.2	70.2	70.2	
						78.5	89.5	163.7	187.9	

Command format	No. of steps		H-2002		H-702/302	
BIN (d, S)	Conditions	Step	Average	Maximum	Average	Maximum
	Word	3	28.5	28.5	58.5	58.5
	Double word	4	74.6	65.4	156.6	136.4

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O (BCD) after conversion						o	o			o	o		
S	I/O (BIN) before conversion					o	o	o	o	o	o	o	o	

[Function]

- The contents of S are converted from a BCD number to a binary number and the result is outputted to d.
- When the contents of S are not BCD data (when A to F are found in the data), DER (R7F4) is 1 and no conversion is executed. (d is left unchanged.)



Combination of d and S

d	S
Word	Word
Double word	Double word

Example:

[Precautions]

- When a data error occurs, the contents of d are left unchanged.

[Name] Decode

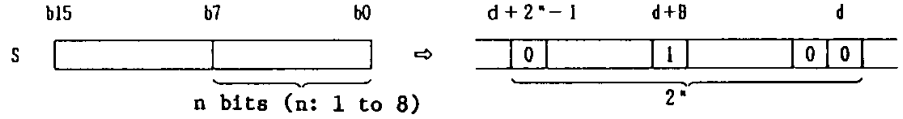
Ladder format	Condition code					Processing time (μs)				Remarks
DECO (d, S, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	.	.	.	.	59.9 + 0.3 n		115.3 + 0.6 n		
Command format	No. of steps					H-2002		H-702/302		
DECO (d, S, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			4		49.9 + 0.2 n		96.1 + 0.5 n		

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O of decoding destination			o										
S	Word I/O to be decoded					o	o	o	o				o	
n	No. of bits to be decoded												o	1 to 8 (decimal)

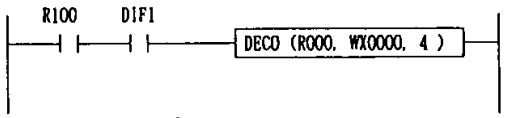
[Function]

. The low-order n bits of S are decoded to 2^n and 1 is outputted to the decoded bits of the bit string from d to d+2n-1. (n: 1 to 8)

. When n is 0, no data is decoded. The contents from d to $d+2^n-1$ are left unchanged.



[Program example]




```
LD R100
AND DIF1
[
DECO (R000. WX0000. 4 )
]
```

[Program explanation]

When R100 is set from 0 to 1, a bit, that is appointed with the value of low-order 4 bits of WX0000 from top address R000, is set to 1.

Example: When WX0000 = HFFFF, 1 is set in the 15th bit, that is, R00F from R000 in this program.

[Precautions]

. Keep $d+2^n-1$ within the I/O limits (R7FF, L03FFF, L13FFF, M3FFF). When $d+2^n-1$ is beyond the limits, DER = 1 and no data is decoded.

. The value of n should be one of 1 to 8.

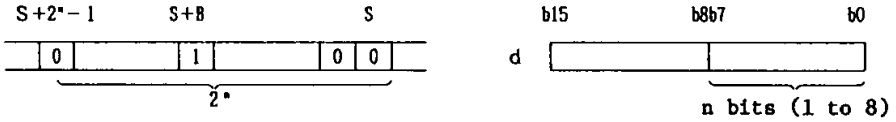
[Name] Encode

Ladder format	Condition code					Processing time (μs)				Remarks
ENCO (d, S, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	↕	76.5 + 0.4 n		159.0 + 0.8 n		
Command format	No. of steps					H-2002		H-702/302		
ENCO (d, S, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			4		63.7 + 0.3 n		132.5 + 0.7 n		

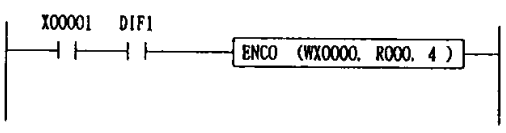
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O of encoding destination						o	o						
S	Top I/O of bit string to be encoded			o										
n	No. of bits to be encoded												o	1 to 8 (decimal)

[Function]

- The value of bit position, that is set to 1 and within 2^n bits from S to S+2^n-1, is encoded and set in d.
- When n is 0, no data is encoded. The contents of d are left unchanged.
- When a plurality of ones are found between S and S+2^n-1, the high-order bit is encoded.



[Program example]




```
LD  X00001
AND DIF1
[
ENCO (WR0000, R000, 4)
]
```

[Program explanation]

- The most significant bit where 1 is set is detected from the bit string between R000 and R000F (24-1=15 bits) at the leading edge of X00001 and a binary number 4 bits in length is set in the word I/O of d.

  Example: When 1 is set in the 7th and 6th bits of R00C to R00F, H0007 is set in WR0000.

[Precautions]

. Keep $S+2^n-1$ within the I/O limits (R7FF, L03FFF, L13FFF, M3FFF). When $S+2^n-1$ is beyond the limits, DER = 1 and no data is encoded.

. The value of n should be one of 1 to 8.

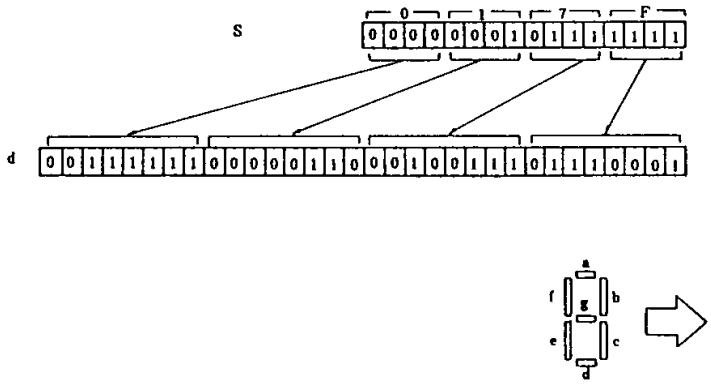
[Name] 7-segment decode (Segment)

Ladder format	Condition code					Processing time (μs)				Remarks
SEG (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	19.1	←	37.9	←	
Command format	No. of steps					H-2002		H-702/302		
SEG (d, S)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		15.9	←	31.6	←	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O of decoding destination										o	o		
S	Decoding contents					o	o	o	o				o	

[Function]

. The contents of S are converted to a 4-digit 7-segment display code on the assumption that one digit is 4 bits in length and the result is outputted to d.



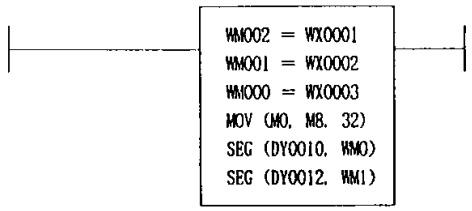
Conversion table

Input data, 4 bits	Output data								Display
		g	f	e	d	c	b	a	
0	0	0	1	1	1	1	1	1	0
1	0	0	0	0	0	1	1	0	1
2	0	1	0	1	1	0	1	1	2
3	0	1	0	0	1	1	1	1	3
4	0	1	1	0	0	1	1	0	4
5	0	1	1	0	1	1	0	1	5
6	0	1	1	1	1	1	0	1	6
7	0	0	1	0	0	1	1	1	7
8	0	1	1	1	1	1	1	1	8
9	0	1	1	0	1	1	1	1	9
A	0	1	1	1	0	1	1	1	A
B	0	1	1	1	1	1	0	0	b
C	0	0	1	1	1	0	0	1	C
D	0	1	0	1	1	1	1	0	d
E	0	1	1	1	1	0	0	1	E
F	0	1	1	1	0	0	0	1	F

[Precautions]

[Program example]

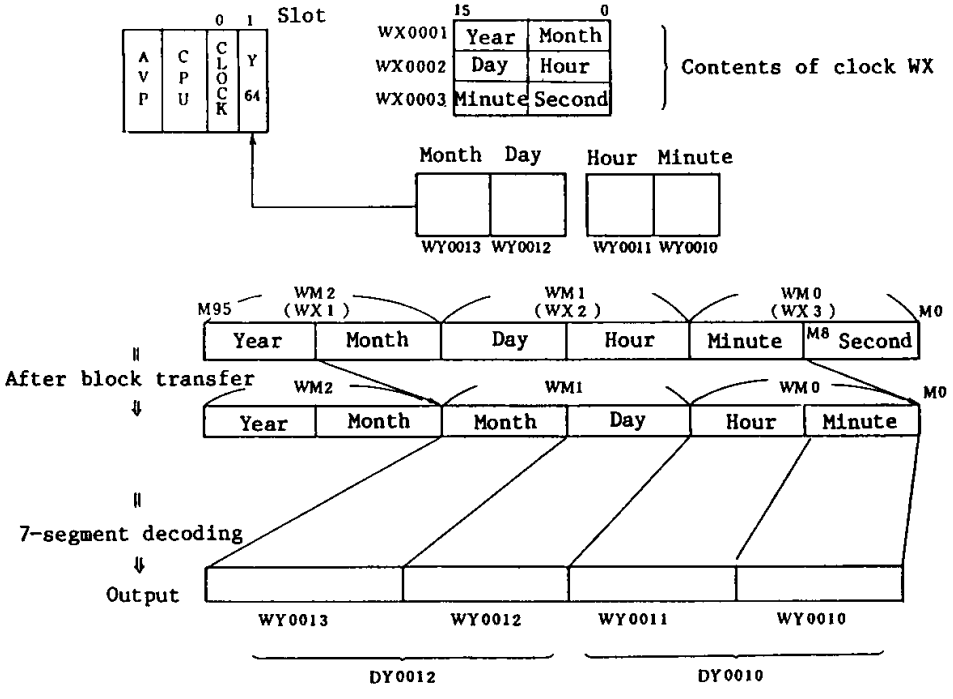
Data (WX0001 to WX0003) of the clock module is outputted to the 7-segment display unit of DY0010 and DY0012 (64-point output).




```
[
WM002 = WX0001
WM001 = WX0002
WM000 = WX0003
MOV (M0, M8, 32)
SEG (DY0010, WM0)
SEG (DY0012, WM1)
]
```

[Program explanation]

The month, day, hour, and minute, which are read from the clock module, are outputted to a 8-digit 7-segment display unit.



SEG

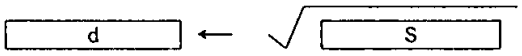
[Name] Square root

Ladder format	Condition code					Processing time (μs)			Remarks
SQR (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300	
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum
	↕	.	.	.	.	409.3	421.8	882.6	855.1
Command format	No. of steps					H-2002		H-702/302	
SQR (d, S)	Conditions			Step		Average	Maximum	Average	Maximum
	-			4		341.1	351.5	735.5	712.6

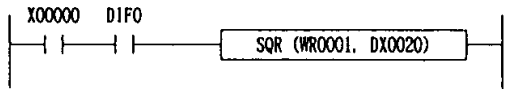
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Square root (BCD)						o	o						
S	I/O (BCD) with square root to be calculated									o	o	o	o	

[Function]

- The square root of the contents of S is calculated and the result is outputted to d.
- Set BCD data in S.
  When the BCD data of S is faulty, DER (R7F4) is 1 and no operation is performed. (There is data other than H0 to H9.)
- The figures below the decimal point are omitted.



[Program example]




```
LD  X00000
AND DIF0
[
SQR (WR0001, DX0020)
]
```

[Program explanation]

- The square root of the value of DX0020 is calculated at the leading edge of X00000 and the result is substituted in WR0001.

  Example: The following is obtained by operation when DX0020 = H00002159 (BCD).
  WR0001 = H0046 (BCD) (2159 = 46.465 ...)

[Precautions]

[Name] Bit count

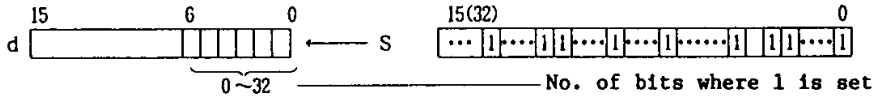
Ladder format	Condition code					Processing time (μs)				Remarks
BCU (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		The upper values are values when d indicates a word and the lower values are values when d indicates a double word.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	47.9	55.9	104.3	121.9	
						90.7	106.7	198.1	233.3	

Command format	No. of steps		H-2002		H-702/302	
BCU (d, S)	Conditions	Step	Average	Maximum	Average	Maximum
	Word	3	39.9	46.6	86.9	101.6
	Double word	4	75.6	88.9	194.4	165.1

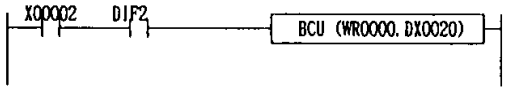
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	No. of bits of 1						o	o						
S	I/O for counting bits of 1					o	o	o	o	o	o	o	o	

[Function]

. The number of bits of the contents (16 bits for word or 32 bits for double word) of S where 1 is set is outputted to d.



[Program example]




```
LD  X00002
AND DIF2
[
BCU (WR0000, DX0020)
]
```

[Program explanation]

. The number of bits where 1 is set is counted from the data inputted in DX0020 at the leading edge of X00002 and the result is set in WR0000.

Example: The number of bits where 1 is set is 16 (decimal) when

DX0020 = A 7 E 4 F 1 5 3

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

Therefore, the result is WR0000 = H0010.

[Precautions]

[Name] Exchange (Swap)

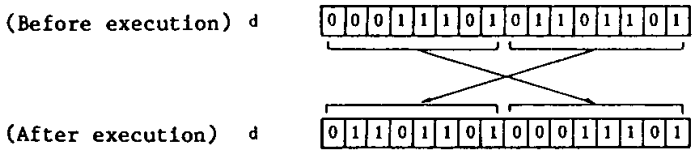
Ladder format	Condition code					Processing time (μs)				Remarks
SWAP (d)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	6.7	←	13.3	←	

Command format	No. of steps		Processing time (μs)			
SWAP (d)	Conditions	Step	H-2002		H-702/302	
			Average	Maximum	Average	Maximum
	-	2	5.6	←	11.1	←

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O to be exchanged						o	o						

[Function]

· The high-order 8 bits of the contents of d are exchanged with the low-order 8 bits.



[Precautions]

· The start condition of this instruction should be Edge Trigger.

[Name] FIFO initial (FIFO initialize)

Ladder format	Condition code					Processing time (μs)				Remarks
FIFIT (P, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	.	.	.	.	47.8	66.7	95.8	134.5	
Command format	No. of steps					H-2002		H-702/302		
FIFIT (P, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		39.8	55.6	79.8	112.1	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
P	FIFO top I/O							o						
n	FIFO size												o	
														0 to 256

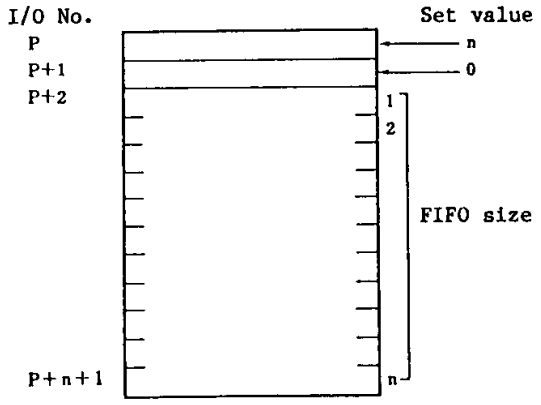
[Function]

FIFO is an abbreviation for first-in first-out. Data is stored in the buffer and fetched in the order of entry. This instruction initializes FIFO.

. The FIFO top I/O number P and the FIFO size n are set. When $0 \leqq n \leqq 256$, n is set in P.
When $257 \leqq n$, 256 is set in P.

. The initial value 0 is set in P+1 as a use count of FIFO.

. n+2 words from P to P+n+1 are used for FIFO.



[Precautions]

. When P+n+1 is beyond the I/O limits (WRC3FF, WL03FF, WL13FF, WM3FF), DER (R7F4) is 1 and the maximum value (last) of the range - (P+1) is set in P.

. Set n to one of 0 to 256. When n > 256, DER = 1 and no processing is performed.

[Name] FIFO write

Ladder format	Condition code					Processing time (μs)				Remarks
FIFWR (P, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	53.2	72.3	105.6	144.3	
Command format	No. of steps					H-2002		H-702/302		
FIFWR (P, S)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		44.3	60.2	88.0	120.2	

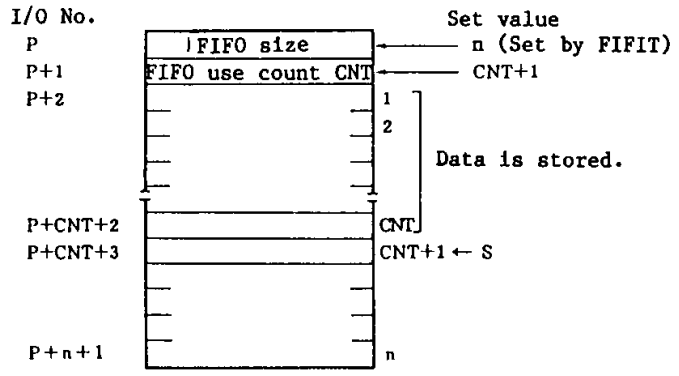
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
P	FIFO top I/O							o						
n	Contents to be written into FIFO					o	o	o	o				o	

[Function]

Data is written into the FIFO buffer of the top I/O number P.

When use count CNT < size n, the contents of S are written into P+CNT+2. 1 is added to the use count CNT.

When use count CNT ≥ size n, DER (R7F4) is 1 and no data is written.



[Precautions]

- When P+n+1 is beyond the I/O limits (WRC3FF, WL03FF, WL13FF, WM3FF), DER (R7F4) is 1 and no data is written.
- Set n to one of 0 to 256. When n > 256, DER = 1 and no processing is performed.

[Name] FIFO Read

Ladder format	Condition code					Processing time (μs)				Remarks
FIFRD (P, d)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		n indicates the FIFO size. The contents of ( ) are values when P and d indicate WL and WY.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	.	.	.	.	75.8 + 3.9 n (6.8)		151.7 + 7.5 n (10.4)		
Command format	No. of steps					H-2002		H-702/302		
FIFRD (P, d)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		63.2 + 3.2 n (5.7)		126.4 + 6.2 n (8.7)		

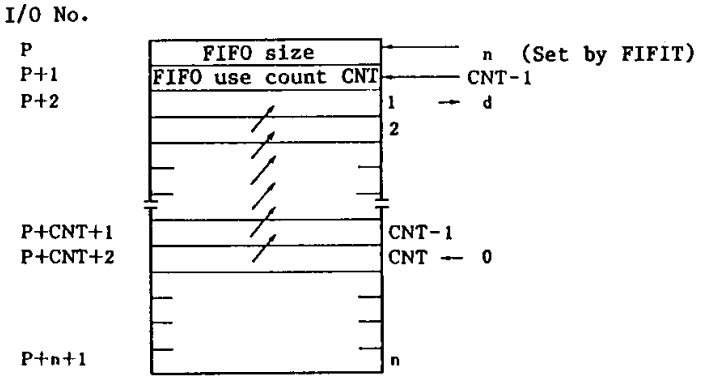
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
P	FIFO top I/O							o						
d	I/O for storing read data						o	o	o				o	

[Function]

Data in the FIFO buffer of the top I/O number P is read.

When 1 ≦ use count CNT ≦ size n:
The contents of P+2 are read and written into d.
The contents of P+3 to P+CNT+2 are transferred to the previous I/O respectively.
0 is written into P+CNT+2.
The contents of CNT are set to -1. When CNT = 0, no data is read. The previous data is held.

When use count CNT > size n:
DER (R7F4) is 1 and no data is read.



[Precautions]

. When P+n+1 is beyond the I/O limits (WRC3FF, WLO3FF, WL13FF, WM3FF), DER (R7F4) is 1 and no data is read.

. Set n to one of 0 to 256. When n > 256, DER = 1 and no processing is performed.

FIFRD

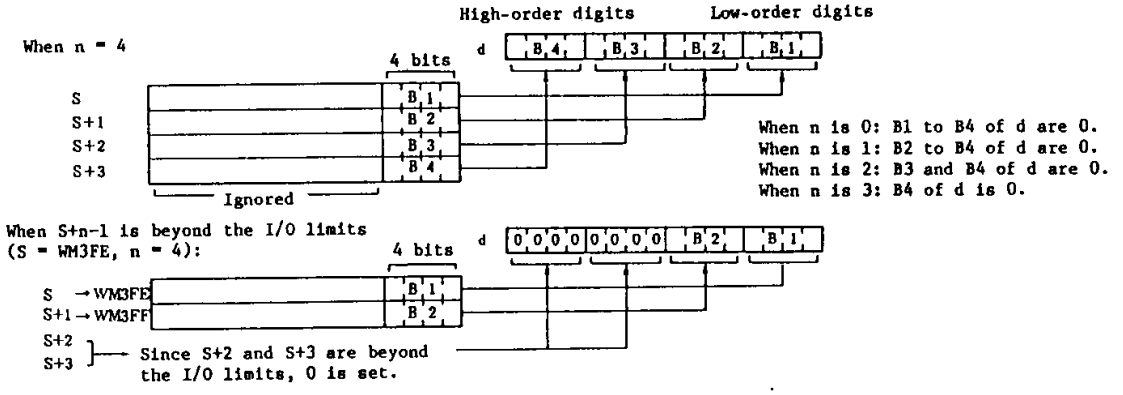
[Name] Connection (Unit)

Ladder format	Condition code					Processing time (μs)				Remarks
UNIT (d, S, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	.	.	.	.	70.4	89.3	142.6	181.3	
Command format	No. of steps					H-2002		H-702/302		
UNIT (d, S, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			4		58.7	74.4	118.8	151.1	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	I/O of connection result writing destination						o	o						
S	Top I/O to be connected							o						
n	No. of words to be connected												o	n = 0 to 4

[Function]

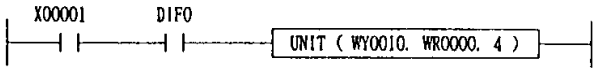
The values of low-order 4 bits of n (1 to 4) words from S are set in units of 4 bits from the low-order position of d.
When n is one of 1 to 3, the bits which are not set in d are set to 0.
The data from S to S+n-1 is held even if the Unit instruction is executed.
When S+n-1 is beyond the I/O limits (WRC3FF, WL03FF, WL13FF, WM3FF), DER (R7F4) is 1 and only the low-order 4 bits from S to the I/O range are set in d.



[Precautions]

When n = 0, the instruction is not executed.

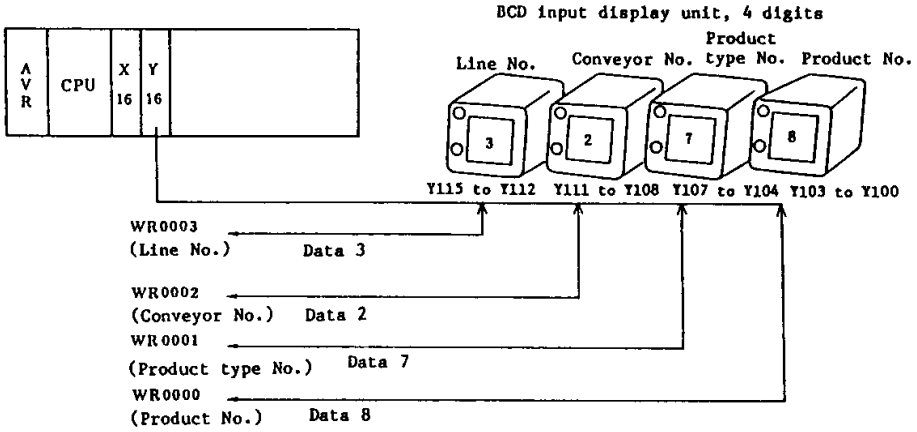
[Program example]




```
LD X00001
AND DIF0
[
UNIT (WY0010. WR0000. 4 )
]
```

[Program explanation]

A 4-digit BCD input display unit is connected to WY0010 and independent data of WR0 to WR3 is displayed in each digit. (Data of only low-order 4 bits of WR0 to WR3 are valid.)



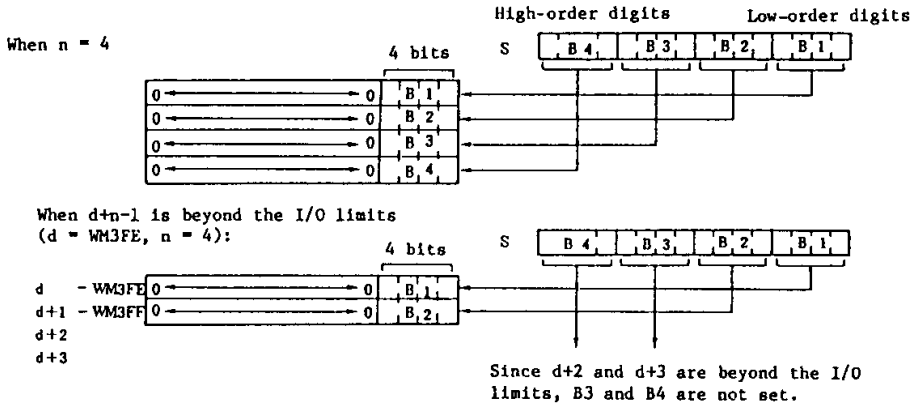
[Name] Separation (Distribute)

Ladder format	Condition code					Processing time (μs)				Remarks
DIST (d, S, n)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	.	.	.	.	69.1	88.0	138.7	177.4	
Command format	No. of steps					H-2002		H-702/302		
DIST (d, S, n)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			4		57.6	73.3	115.6	147.8	

	Usable I/O	Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Top I/O of separation destination							o						
S	I/O to be separated					o	o	o	o				o	
n	No. of words to be separated												o	n = 0 to 4

[Function]

S is separated to values of each 4 bits and set in the low-order 4 bits of n words from d.
The high-order 12 bits from d to d+n-1 are 0.
The data of S is held even when the DIST instruction is executed.
When d+n-1 is beyond the I/O limits (WRC3FF, WL03FF, WL13FF, WM3FF), DER (R7F4) is 1 and the separated data of S is set in the low-order 4 bits from d to the I/O range.



[Precautions]

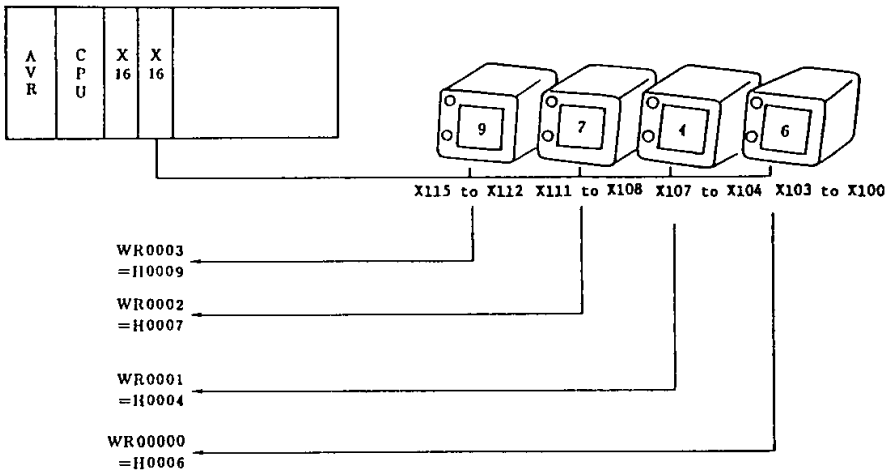
. When n = 0, the instruction is not executed.

[Program example]


```
LD X00001
AND DIF0
[
DIST (WR0000, WX0010, 4 )
]
```

[Program explanation]

The input of the 4-bit and 4-digit Dight switch is connected to WX0010 and each digit data is stored in WR0000 to WR0003 as independent data.



[Name] I/O address conversion

Ladder format	Condition code					Processing time (μs)				Remarks
ADRIO (d, S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		H-2000, H-700, and H-300 are for CPUP-**H.
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	2.3	←	4.2	←	
Command format	No. of steps					H-2002		H-702/302		
ADRIO (d, S)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		1.9	←	3.5	←	

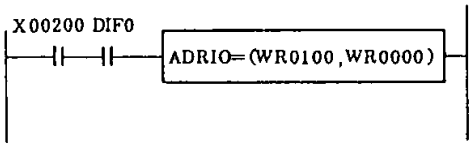
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
d	Conversion address						o	o						
S	I/O to be converted	o	o	o		o	o	o						

[Function]

Real address from S to d

· The real address of the I/O specified by S is determined and set in d.

[Program example]



[Program explanation]

· The real address (H3C00) of WR0 is set in WR0100 at the leading edge of X00200.
After the instruction is executed: WR0000 WR0100 H3C00

[Precautions]

· In this instruction, only CPU2-**H and CPUP-**H are valid. The CPUs other than them are the same as NOP (no operation).

[2] Control Command

[Name] Ending normal scan (END)

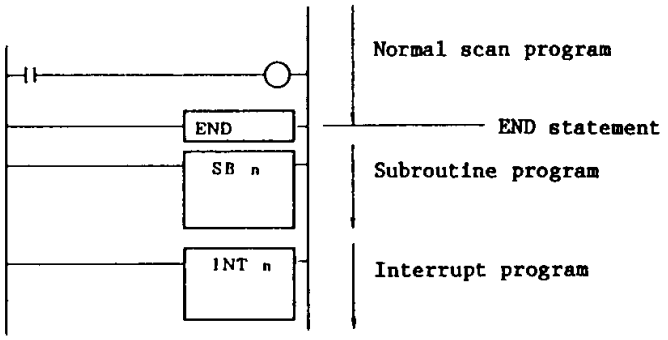
Ladder format	Condition code					Processing time (μs)				Remarks
END	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	5.5	←	11.5	←	
Command format	No. of steps					H-2002		H-702/302		
END	Conditions			Step		Average	Maximum	Average	Maximum	
	-			1		4.6	←	9.6	←	

Usable I/O	Bit				Word				Double word			Constant	Other
	X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		

[Function]

- This statement indicates the end of a normal scan program. (When this statement is executed, control is returned to the top of the program and normal scan is executed.)
- If no subroutine or interrupt scan program is available, this statement is not required.
- If a subroutine or interrupt scan program is available, code this statement at the end of a normal scan program.
- Code this statement only once in a program. Do not enter a start condition.

[Usage]



[Precautions]

- The END statement is checked before operation starts. If an error is found, one of the following error codes is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0010	No END statement is coded.	Code an END statement.
		H0022	Two or more END statements are coded.	Delete the unnecessary END statements.
		H0032	A start condition is set in the END statement.	Delete the start condition.

[Name] Conditional end of scan (CONDITIONAL END)

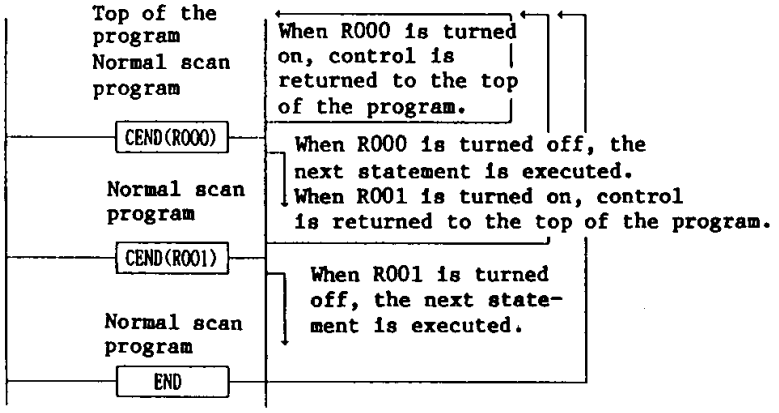
Ladder format	Condition code					Processing time (μs)				Remarks
CEND (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	9.0	←	18.0	←	
Command format	No. of steps					H-2002		H-702/302		
CEND (S)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			2		7.5	←	15.0	←	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Scan end condition	o	o	o										

[Function]

- When this statement is executed with scan end condition S satisfied, control is returned to the top of the normal scan program and the program is executed.
- When this statement is executed with scan end condition S not satisfied, the next statement is executed.
- This statement can be executed only in a normal scan program. This statement can be coded as many times as desired.
- A start condition can be set in this statement. When both S and the start condition are satisfied, this statement is executed.

[Usage]



[Precautions]

- The CEND statement is checked before operation starts. If an error is found, the following error code is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0023	A CEND statement follows an END statement.	Place a CEND statement before an END statement.

[Name] Unconditional jump (JUMP)

Ladder format	Condition code					Processing time (μs)				Remarks
JMP n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	①	.	.	.	57.8	69.3	117.5	141.0	
Command format	No. of steps					H-2002		H-702/302		
JMP n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			2		48.2	57.7	97.9	117.5	

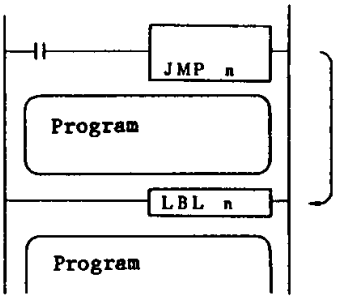
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Code No.												o	0 to 255 (decimal)

JMP

[Function]

- When the start condition in JMPn is satisfied, the program is jumped from this statement to LBLn having the same code number as JMPn. JMPn must be paired with LBLn.
- When the start condition is not satisfied, the next statement is executed.
- When this statement is placed together with other statements in an arithmetic box, this statement must be the last statement in the box.
- The effect of this statement is valid only in the same scan program. (This statement cannot cause a jump from a normal scan program to a subroutine or interrupt scan program, or vice versa.)
- JMPn statements can be nested. However, take case so that cycle time over errors are not caused.

[Usage]



- When the start condition is satisfied, a jump to LBLn occurs.
- If the program to which control is jumped has a timer, the elapsed time is updated but no statement is executed. Therefore, output is disabled even if the ON condition is satisfied.

[Precautions]

- This statement is checked during execution. If an error is found, one of the following error codes is set in internal special output registers R7E3 and WRF015. In this case, no jump occurs and the next program is executed.

Internal special output register		Error code	Nature of error	Action
R7E3 = "1"	WRF015	H0015	No LBLn is found.	Define LBLn as a jump destination.
		H0040	An attempt was made to jump to another program.	Correct the program so that a jump to another program area does not occur.

[Name] Conditional jump (CONDITIONAL JUMP)

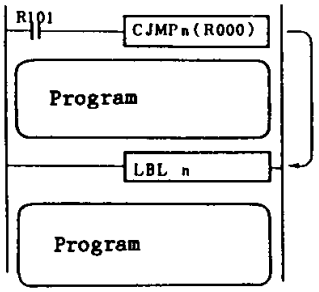
Ladder format	Condition code					Processing time (μs)				Remarks
CJMP n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	1	.	.	.	59.4	70.9	120.6	144.1	
Command format	No. of steps					H-2002		H-702/302		
CJMP n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		49.5	59.1	100.5	120.1	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Code No.												o	0 to 255 (decimal)
S	Jump condition	o	o	o										

[Function]

- When jump condition S in CJMPn (S) is satisfied, the program is jumped from this statement to LBLn having the same code number as CJMPn (S). CJMPn (S) must be paired with LBLn.
- When the start condition or jump condition is not satisfied, the next statement is executed.
- When placing this statement together with other statements in an arithmetic box, take care. If the condition is satisfied, a jump occurs without the remaining operations in the box being performed.
- The effect of this statement is valid only in the same scan program. (This statement cannot cause a jump from a normal scan program to a subroutine or interrupt scan program, or vice versa.)
- CJMPn (S) statements can be nested. However, take case so that cycle time over error is not caused.

[Usage]



- When both the I/O R000 bit for the start condition and that for the jump condition are satisfied, a jump to LBLn occurs.
- If the program to which control is jumped has a timer, the elapsed time is updated but no statement is executed. Therefore, the timer coil is not turned on even if the ON condition is satisfied.

[Precautions]

- This statement is checked during execution. If an error is found, one of the following error codes is set in internal special output registers R7E3 and WRF015. In this case, no jump occurs and the next program is executed.

Internal special output register		Error code	Nature of error	Action
R7E3 = "1"3.	WRF015	H0015	No LBLn is found.	Define LBLn as a jump destination.
		H0040	An attempt was made to jump to another program.	Correct the program so that a jump to another program area does not occur.

[Name] Reserve (RESERVE)

Ladder format	Condition code					Processing time (μs)				Remarks
RSRV n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	1	.	.	.	60.2	71.7	122.0	145.7	
Command format	No. of steps					H-2002		H-702/302		
RSRV n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			2		50.2	59.4	101.7	121.4	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Code No.												o	0 to 255 (decimal)

RSRV

[Function]

. This statement can be used for exclusive control between a BASIC module and a CPU module. The statement must be paired with a FREE statement. When the CPU module executes an RSRVn statement, the operation of the CPU depends on the status of the BASIC module.

1. If the BASIC module executes a RESERVE statement and locks the CPU, the CPU jumps to LBLn indicated by the code number in RSRVn.

2. If the BASIC module executes a FREE statement and releases the CPU, the CPU executes the subsequent statements.

. When the BASIC module executes a RESERVE statement, the operation of the BASIC module depends on the status of the CPU module.

1. If the CPU module executes a RESERVE statement and locks the CPU, the BASIC module waits for 10 ms.

2. If the CPU module executes a FREE statement and releases the CPU, the BASIC module executes the subsequent statements.

. RSRVn must be paired with LBLn having the same code number.

. The effect of RSRVn statement is valid only in the same scan program. (This statement cannot cause a jump from a normal scan program to a subroutine or interrupt scan program, or vice versa.)

. RSRVn statements can be nested. However, take case so that cycle time over error is not caused.

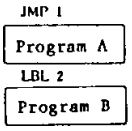
[Precautions]

. This statement is checked during execution. If an error is found, one of the following error codes is set in internal special output registers R7E3 and WRF015. In this case, no jump occurs and the next program is executed.

Internal special output register		Error code	Nature of error	Action
R7E3 = "1"	WRF015	H0015	No LBLn is found.	Define LBLn as a jump destination.
		H0040	An attempt was made to jump to another program.	Correct the program so that a jump to another program area does not occur.

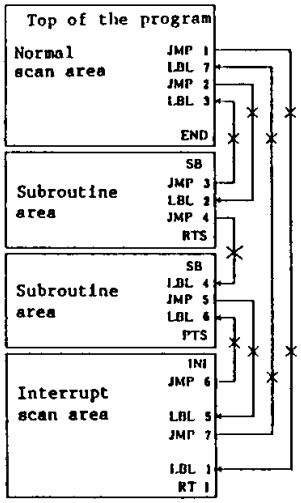
[Syntax of JMP, CJMP, and RSRV]

① LBLn having the same code number (n) as the JMP statement is required.



. If JMP1 is executed without LBL1, an error indicating that no LBL is defined is caused.
When this happens, the next program A is executed without JMP1 being executed.

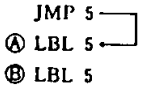
② A jump beyond the area including the JMP statement cannot be caused.



. LBL1 is not in the normal scan area. If the JMP1 statement is executed, an error indicating an out-of area jump occurs.
When this happens, the next statement is executed without the JMP1 statement being executed.

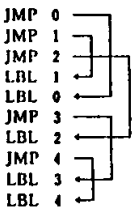
. JMP2 to JMP7 function in the same way as JMP1.

③ LBLn having the same code number (n) as the JMP statement must not be duplicated.

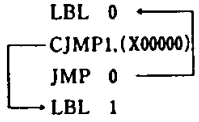


. In processing before operation, label statements Ⓐ and Ⓑ have the same code number whose value is 5. This causes a duplicated definition error.

④ JMP statements can be nested.



⑤ The JMP statement can cause a jump to an statement before it.



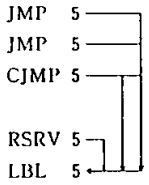
. When input X00000 is turned on, control exits from the loop between LBL0 and JMP0 by jumping from CJMP1, (X00000) to LBL1.

. If CJMP1, (X00000) or another statement that exits control from the loop is not coded, the program loops indefinitely between LBL0 and JMP0.

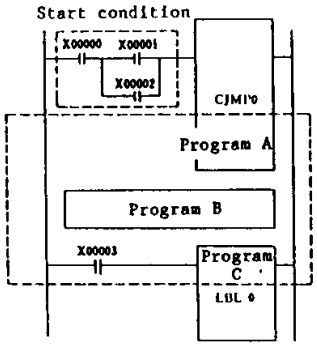
. JMP0 causes a jump to JBL0 placed before it.

JMP
CJMP
RSRV

⑥ JMP statements having the same code number can be duplicated.



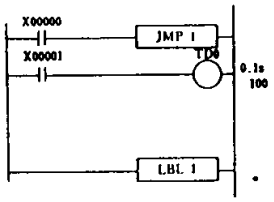
⑦ A start condition can be programmed for a JMP statement.



. If a jump is made from CJMP0 to LBL0, programs A, B, and C are not executed.

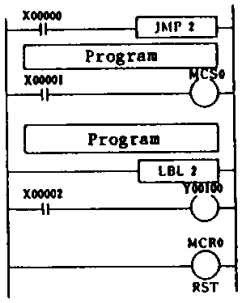
⑧ The CJMP and RSRV statements also conform to the syntax described in ① to ⑦ above.

Note 1: When a JMP statement causes a jump to LBL, the status of each I/O between JMP and LBL is retained. However, the elapsed time in the timer is updated.



. If X00001 is turned on then X00000 is turned on, the elapsed time in TD0 is updated even if a jump is made from JMP1 to LBL1.
If X00000 is left on, TD0 is not turned on even if it exceeds 100.

Note 2: When a JMP instruction is combined with an MCS or MCR statement, operation is performed as follows. Take care when programming this combination.



. When JMP2 does not cause a jump, Y00100 is turned on with X00001 and X00002 turned on.

. When JMP2 causes a jump and X00000 is turned on, Y00100 conforms to the on-off status of X00002 regardless of whether X00001 is turned on or off.

Note 3: Do not create a circuit that causes a jump outside between MCS and MCR.

[Name] Freeing reserve (FREE)

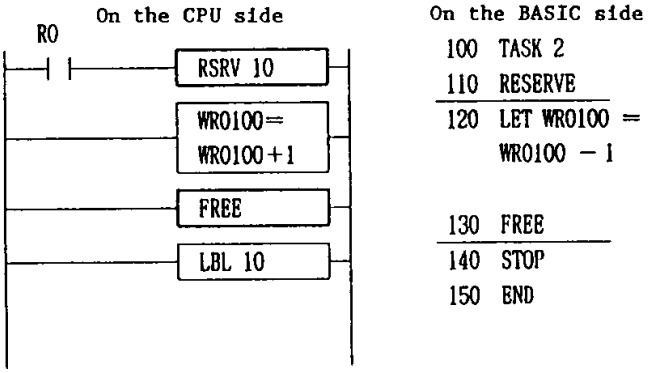
Ladder format	Condition code					Processing time (μs)				Remarks
FREE	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	3.1	←	6.1	←	

Command format	No. of steps		Processing time (μs)			
FREE	Conditions	Step	H-2002		H-702/302	
			Average	Maximum	Average	Maximum
	-	1	2.6	←	5.1	←

Usable I/O	Bit				Word				Double word			Constant	Other
	X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		

[Function]

- This statement uses a BASIC module for exclusive control.
- This statement must be paired with an RSRVn statement. (After RSRVn is executed, be sure to execute FREE.)
- This statement must be placed before LBLn to which a jump is caused by RSRVn.



On the BASIC side

```
100 TASK 2
110 RESERVE
120 LET WR0100 =
    WR0100 - 1

130 FREE
140 STOP
150 END
```

[Usage]

- The ladder program executes RSRV10 first. The BASIC program waits at RESERVE until the ladder program executes the FREE statement.
- The BASIC program executes the RESERVE statement first. The CPU uses RSRV10 to jump to LBL10. The CPU continues jumping until the BASIC program executes the FREE statement.

[Precautions]

[Name] Label (LABEL)

Ladder format	Condition code					Processing time (μs)				Remarks
LBL n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	0.5	←	1.1	←	
Command format	No. of steps					H-2002		H-702/302		
LBL n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			1		0.4	←	0.9	←	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Code No.												o	0 to 255 (decimal)

[Function]

- This statement is a destination to which a jump is made when JMPn, CJMPn(S), or RSRVn is executed. (The number n in LBLn must be the same as that in JMPn, CJMPn(S), or RSRVn.)
- The number n in LBLn must not be duplicated in the same program.
- This statement itself performs nothing.
- Any start condition in LBLn is ignored.

[Precautions]

- This statement is checked before operation starts. If an error is found, the following error code is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0001	LBL is defined twice.	Delete the unnecessary LBLn.

[Name] FOR

Ladder format	Condition code					Processing time (μs)				Remarks
FOR n (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	①	.	.	.	11.7	83.0	24.3	166.4	
Command format	No. of steps					H-2002		H-702/302		
FOR n (S)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			3		9.7	69.2	20.2	138.7	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Code No.												o	0 to 49 (decimal)
S	Repetition count						o	o						

[Function]

. This statement is a destination to which a jump is made from NEXTn having the same code number as this statement.
When repetition count S is greater than 0, the next statement after FORn(S) is executed.
When repetition count S is equal to 0, a jump is made to the next statement after NEXTn.

. FORn(S) must be paired with NEXTn. NEXTn must be placed after FORn.

. FORn(S) must not be duplicated.

. FORn(S) and NEXTn must be in the same program area. (For example, a program containing FORn(S) in the normal scan area and NEXTn in the subroutine area is not allowed.)

. FORn(S)-NEXTn loops can be nested up to five levels. For details, see NEXTn.

[Precautions]

. This statement is checked before operation starts. If an error is found, the following error code is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0001	LBL is defined twice.	Delete the unnecessary LBLn.

. If an error occurs during execution of this statement, one of the following error codes is set in internal special output registers R7E3 and WRF015 and the next program is executed.

Internal special output register		Error code	Nature of error	Action
R7E3 ="1"	WRF015	H0017	No NEXT is defined.	Define NEXT corresponding to FORn.
		H0043	Invalid NEXT for FOR	Code NEXTn after FORn.
		H0044	NEXT in an invalid area	Place NEXTn in the area containing FORn.
		H0045	Invalid FOR-NEXT nesting	FOR-NEXT loops are not nested.
		H0046	Too many FOR nesting levels	Reduce the number of FOR-NEXT nesting levels to 5 or less.

[Name] NEXT

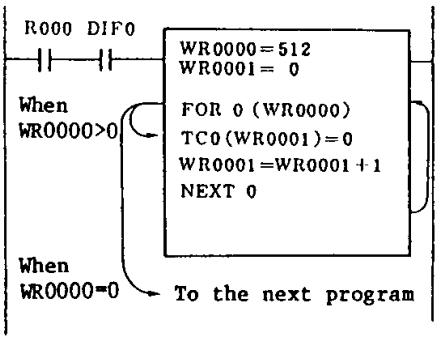
Ladder format	Condition code					Processing time (μs)				Remarks
NEXT n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	①	.	.	.	17.6	34.8	34.4	70.8	
Command format	No. of steps					H-2002		H-702/302		
NEXT n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			2		14.7	29.0	28.7	59.0	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Code No.												o	0 to 49 (decimal)

[Function]

. This statement subtracts 1 from repetition count S in FORn(S) having the same code number as this statement. Then, this statement causes a jump to FORn(S).

[Usage]



. When R0 is turned on, the value (TCn) in the timer counter is zero-cleared for 512 points.

. Once the FOR-NEXT loop is executed, the execution continues until S becomes 0.

. FOR0 (WR0000) executes statements below TC0 (WR0001) = 0 while WR0000 > 0. When NEXT0 is reached, 1 is subtracted from WR0000 and a jump is made to FOR0 (WR0000).

. FOR0 (WR0000) causes a jump to the statement next to this box when WR0000 becomes 0.

[Precautions]

. This statement is checked before operation starts. If an error is found, the following error code is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

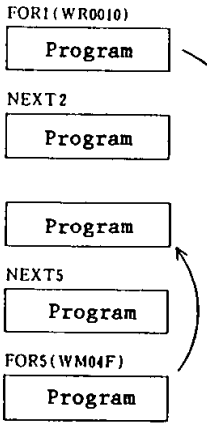
CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0003	The NEXT statement is defined twice.	Delete the unnecessary NEXTn.

. If an error occurs during execution of this statement, one of the following error codes is set in internal special output registers R7E3 and WRF015 and the next program is executed.

Internal special output register		Error code	Nature of error	Action
R7F3 ="1"	WRF015	H0016	No FOR is defined.	Define FORn corresponding to NEXTn.
		H0046	Too many FOR nesting levels	Reduce the number of FOR-NEXT nesting levels to 5 or less.

[Syntax of FOR-NEXT]

(1) A NEXT statement having the same code number (n) as a FOR statement is required after the FOR statement.

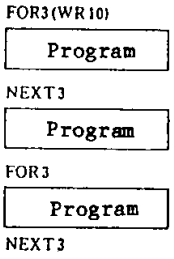


. Missing NEXT
A NEXT statement corresponding to the FOR statement is not coded in the user program.

. Missing FOR
A FOR statement is not coded before a NEXT statement.

. Incorrect NEXT-FOR loop
A NEXT statement is coded before a FOR statement.

(2) FOR or NEXT statements having the same code number (n) must not be duplicated.

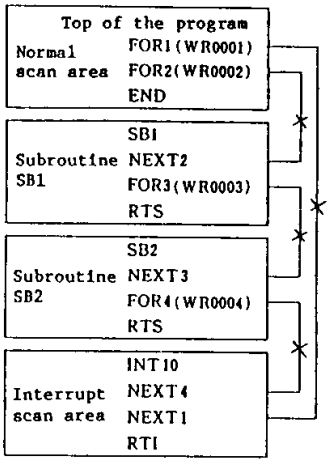


. Duplicated FOR statements
FOR statements having the same code number (n) are coded.

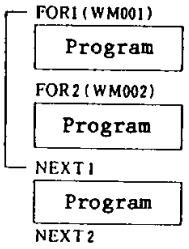
. Duplicated NEXT statements
NEXT statements having the same code number (n) are coded.

Note:
FOR or NEXT statement may be duplicated in processing before operation starts.

(3) FOR and NEXT statements must be in the same program area.



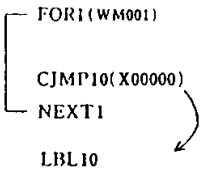
(4) Use a FOR-NEXT nesting structure.



The statements from FOR1 (WM001) to NEXT are executed normally.

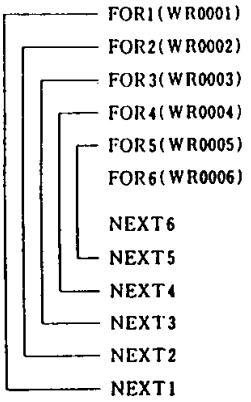
o Nesting error
When WM002 = 0
The statements from FOR1 (WM001) to NEXT1 are given priority. Therefore, a jump from FOR2 across NEXT1 to NEXT2 does not occur. NEXT2 causes an error due to an undefined FOR2.
When WM002 ≠ 0
FOR2 does nothing. NEXT2 causes an error due to an undefined FOR2.

⑤ A jump statement can be used to exit from a FOR-NEXT loop.



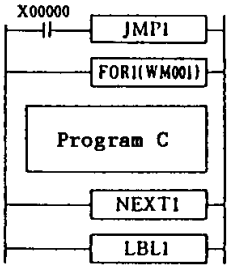
. When X00000 is turned on before the FOR1-NEXT1 loop is executed by the specified number of repetitions (contents of WH1), the program can exit from the loop.

⑥ Up to five FOR-NEXT loops can be nested. If a subroutine is included, the FOR-NEXT loops in it are also counted.



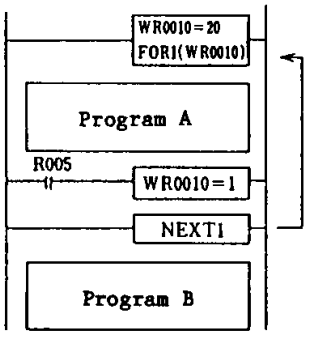
. Too many nesting levels

⑦ Do not include start conditions in a FOR-NEXT loop. (See the circuit diagram below.)



[Explanation]
When X00000 is off, program C is executed by the number specified by WM1.
When X00000 is on, a jump occurs from JMP1 to LBL1. Program C is not executed.

⑧ The repetition count can be changed in the program.



The contents of WR0010 are subtracted by one and a jump is made to FOR1 (WR0010).

. When R005 is off
After program A is executed 20 times, program B is executed.

. When R005 is on
The number of repetitions indicated by WR0010 becomes 1 and 1 is subtracted in processing of NEXT1. So, the contents of WR0010 become 0. This terminates repetition of program A and executes program B.

[Name] Subroutine call (CALL)

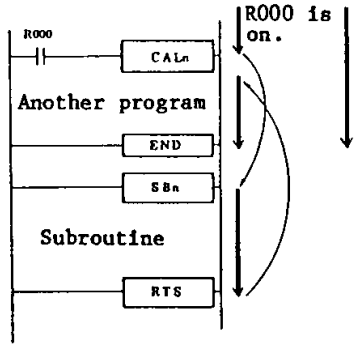
Ladder format	Condition code					Processing time (μs)				Remarks
CAL n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	①	.	.	.	19.0	←	36.4	←	
Command format	No. of steps					H-2002		H-702/302		
CAL n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			2		15.8	←	15.8	←	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Code No.												o	0 to 99 (decimal)

[Function]

- When the start condition in the CALn statement is satisfied, the subroutine having the same code number (program bracketed by SBn and RTS) as the statement is executed.
- When the start condition is not satisfied, the next program is executed.
- Up to five other nested subroutines can be called from a subroutine. (The maximum nesting level is 5.)
- This statement can also call a subroutine in the interrupt scan program.

[Usage]



- When R000 is on, CALn executes a subroutine. After the subroutine has been executed, the next program after CALn is executed.
- When R000 is off, the next program is executed without a subroutine program being executed.

[Precautions]

- If an error is found during execution of this statement, one of the following error codes is set in internal special output registers R7E3 and WRF015. In this case, the next program is executed.

Internal special output register		Error code	Nature of error	Action
R7F3 ="1"	WRF015	H0013	No SB is found.	Define an SBn-RTS pair for CALn.
		H0041	Too many nesting levels	Reduce the number of subroutine nesting levels to 5 or less.

[Name] Subroutine start (SUBROUTINE)

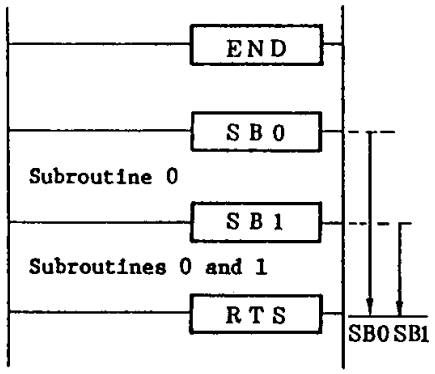
Ladder format	Condition code					Processing time (μs)				Remarks
SB n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	①	.	.	.	0.5	←	1.1	←	
Command format	No. of steps					H-2002		H-702/302		
SB n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			1		0.4	←	0.9	←	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Code No.												o	0 to 99 (decimal)

[Function]

- This statement declares the beginning of a subroutine. (This statement processes nothing.)
- The number n in SBn must not be duplicated in the same program.
- Any start condition in SBn is ignored.
- SBn must be paired with RTS.
- Place the subroutines from SBn to RTS after an END statement.

[Usage]



- When CAL0 is executed, the program from SB0 to RTS is executed as a subroutine.
- When CAL1 is executed, the program from SB1 to RTS is executed as a subroutine.

[Precautions]

- This statement is checked before operation starts. If an error is found, one of the following error codes is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0004	SB is defined twice.	Delete the unnecessary SBn.
		H0013	SB is not defined.	Define SBn corresponding to RTS.

[Name] Subroutine end (RETURN SUBROUTINE)

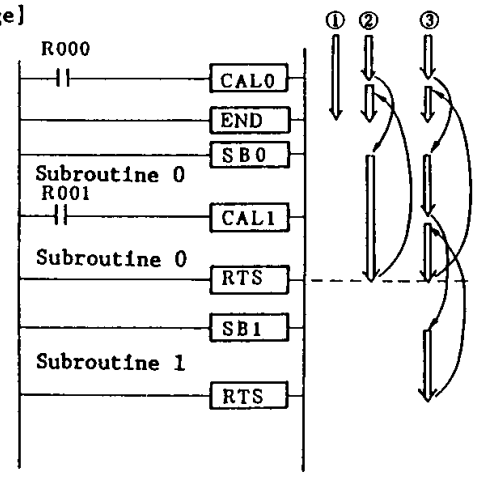
Ladder format	Condition code					Processing time (μs)				Remarks
RTS	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	12.5	←	24.1	←	
Command format	No. of steps					H-2002		H-702/302		
RTS	Conditions			Step		Average	Maximum	Average	Maximum	
	-			1		10.4	←	20.1	←	

Usable I/O	Bit				Word				Double word			Constant	Other
	X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		

[Function]

- This statement declares the end of a subroutine.
- When this statement is executed, the next program after the CALn statement that called a subroutine is executed.
- Any start condition must not be entered in this statement.

[Usage]



(1) Program execution when both R000 and R001 are off.

(2) Program execution when R000 is on but R001 is off.
CAL0 is executed and control is passed to subroutine 0. CAL1 is not executed. When subroutine 0 is terminated, control is returned to the next program after CAL0.

(3) Program execution when both R000 and R001 are on.
CAL0 is executed and control is passed to subroutine 0.
CAL1 is executed and control is passed to subroutine 1.
When subroutine 1 is terminated, control is returned to the next program after CAL1.
When subroutine 0 is terminated, control is returned to the next program after CAL1.

[Precautions]

- This statement is checked before operation starts. If an error is found, one of the following error codes is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0011	RTS is not defined.	Define RTS corresponding to SBn.
		H0020	RTS in an invalid area	Place RTS after END. RTS cannot be used in the interrupt program.
		H0030	Invalid start condition in RTS	Do not enter any start condition in RTS.

[Name] Starting interrupt scan (INTERRUPT)

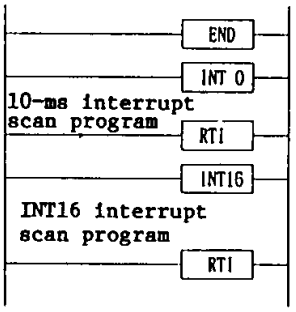
Ladder format	Condition code					Processing time (μs)				Remarks
INT n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	0.5	←	1.1	←	
Command format	No. of steps					H-2002		H-702/302		
INT n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			1		0.4	←	0.9	←	

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	Interrupt priority												o	0 to 2, 16 to 31 (decimal)

[Function]

- This statement declares the beginning of an interrupt scan program.
- When n is from 0 to 2, periodic interrupt scan takes place. When n is from 16 to 31, interrupt scan is caused by an interrupt module.
- A smaller value of n indicates a higher interrupt priority.
- INTn must be paired with RTI.
- Any start condition in INTn is ignored.
- Place the interrupt scan programs from INTn to RTI after an END statement.
- The number n in INTn must not be duplicated in the same program.

[Usage]



- The program from INT0 to RTI is started and executed at 10 ms intervals.
- The program from INT16 to RTI is started and executed when bit 1 in the interrupt module is turned on.

[Precautions]

- This statement is checked before operation starts. If an error is found, one of the following error codes is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0005	INT is defined twice.	Delete the unnecessary INTn.
		H0014	INT is not defined.	Define INTn corresponding to RTI.

[Name] Ending interrupt scan program (RETURN INTERRUPT)

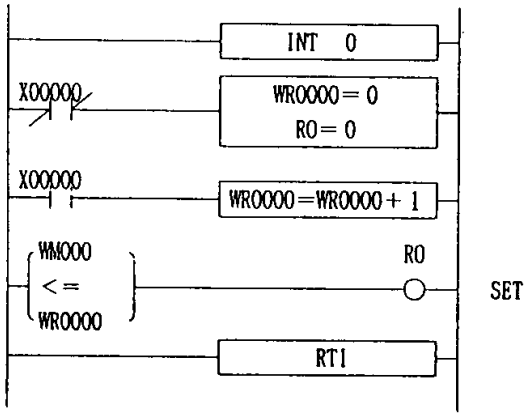
Ladder format	Condition code					Processing time (μs)				Remarks
RTI	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	5.1	←	10.5	←	
Command format	No. of steps					H-2002		H-702/302		
RTI	Conditions			Step		Average	Maximum	Average	Maximum	
	-			1		4.2	←	8.7	←	

Usable I/O	Bit				Word				Double word				
	X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM	Constant	Other

[Function]

- This statement declares the end of an interrupt scan program.
- When this statement is executed, control is returned to the program which was being executed before interrupt scan was performed.
- Any start condition must not be entered in this statement.

[Usage]



- A 0.01-s timer is created using interrupts generated at 10 ms intervals. The setting is set in WM000. The expired time is set in WR0000. R0 is used as a timer coil.
- When X00000 is off, the expired time and timer coil are cleared.
- When X00000 is on, the expired time is incremented by one at 10 ms intervals.
- When WM000 < = WR0000, the timer coil is turned on.

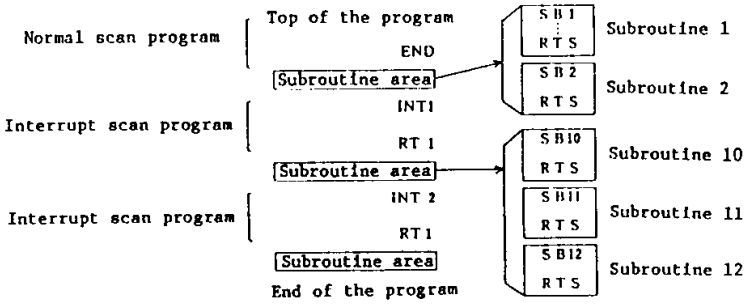
[Precautions]

- This statement is checked before operation starts. If an error is found, one of the following error codes is set in internal special output register WRF001. In this case, 34 is displayed as the CPU error code.

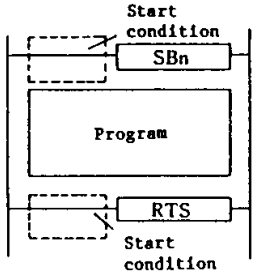
CPU display	Internal special output register	Error code	Nature of error	Action
34	WRF001	H0012	RTI is not defined.	Define RTI corresponding to INTn.
		H0021	RTI in an invalid area	Place RTI after END. RTI cannot be used in SBn.
		H0031	Invalid start condition in RTI	Do not enter any start condition in RTI.

[Syntax of SBn, RTS, INTn and RTI]

(1) Subroutines can be placed between a normal scan program and interrupt scan program, between interrupt scan programs, or after the last interrupt scan program.

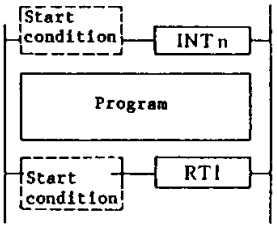


(2) The Subroutine Start (SBn) and Subroutine End (RTS) statements must not contain any start condition.

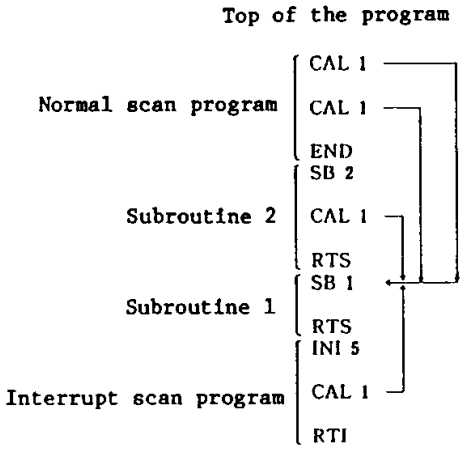


. Since a start condition is set in RTS, an error occurs in processing before operation.

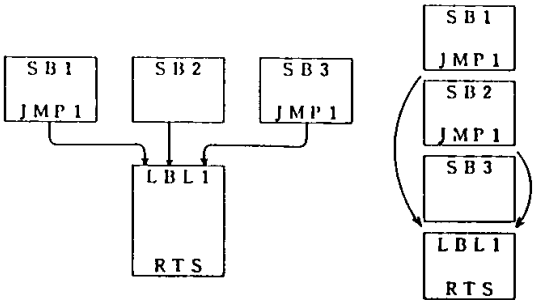
(3) The Starting Interrupt Scan (INTn) and Ending Interrupt Scan Program (RTI) statements must not contain any start condition.



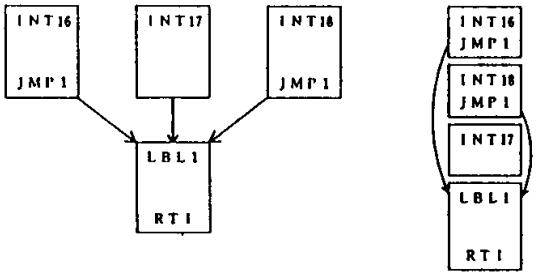
(4) A subroutine can be called from the normal scan program, the interrupt scan program, or another subroutine.



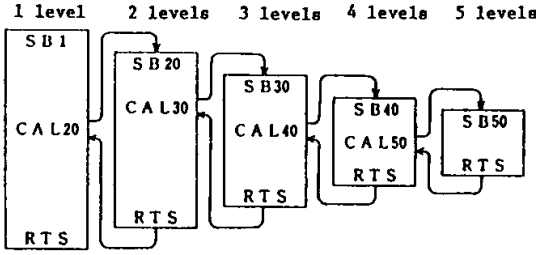
(5) Subroutines having many entries and one exit can also be coded.



(6) Scan programs having many entries and one exit can also be coded.

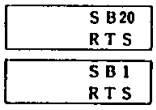


(7) Subroutines can be nested at up to five levels.



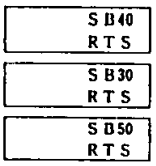
Top of the program

END



INT5

RT1



(1) There is no relationship between the order of subroutines and the nesting order, as shown on the left.

[Name] Starting BASIC program task (START)

Ladder format	Condition code					Processing time (μs)				Remarks
START n	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
		•	•	•	•	7.1	←	14.9	←	
Command format	No. of steps					H-2002		H-702/302		
START n	Conditions			Step		Average	Maximum	Average	Maximum	
	-			2		5.9	←	12.4	←	

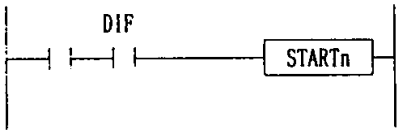
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
n	No. of task to be started												o	1 to 16 (decimal)

[Function]

- This statement can be used to start a BASIC program task from a ladder program with a BASIC module.
- When the value of n is not from 1 to 16, DER is set to 1, preventing this statement from being executed.
- When this statement is executed normally, DER is set to 0.

[Usage]

On the ladder program side



BASIC program

```
100   TASKn
110     :
120     :
 :      :
900   END
```

- When the ladder program executes this statement, a BASIC program task is started.
- When TASKn corresponding to STARTn is not present in the BASIC program, the BASIC module displays error code 90 indicating that an attempt was made to start an undefined task. However, the ladder program terminates normally.

[Precautions]

- Be sure to specify edge detection in this statement as a start condition.

[3] Transfer Command for Sophisticated Function Module

Name General purpose port communication command

Ladder format	Condition code					Processing time (μs)				Remarks
TRNSO (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
TRNSO (d,s,t)	Conditions		Step			Average	Maximum	Average	Maximum	
	-		5			96	177	180	334	

Usable I/O		Bit				Word				Double word			Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o								
s	Top of parameter area							o							In regard to "s," up to s + 14 is used.
t	Top of communications control bit area			o											In regard to "t," up to t + 11 is used.

TRNS 0

Function

1. This is the communication command for the general purpose port used by the CPU ladder program.

2. The top I/O No. of the parameter area is set in s, which in turn sets the various parameters (top of the transmitted/received data area and size, time out value, transmission code) used for communications.

3. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

4. The arbitrarily mounted WY is set in d. (Dummy constant)

5. The TRNSO command can receive after transmission.

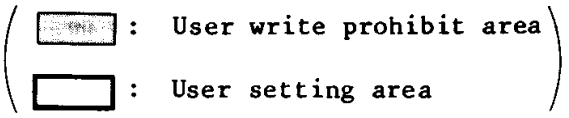
Precautions

. Array variables cannot be used in d, s and t.

. Use s + 14 and t + 11 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area

S	(1) Return code
S+1	(2) System area (not usable by the user)
S+3	(3) Time out time
S+4	(4) Top I/O of the transmission data area
S+6	(5) Transmission data area size
S+7	(6) Top I/O of the transmission data area
S+9	(7) Reception data area size
S+10	(8) Reception data length
S+11	(9) Starting code
S+12	(10) Ending code
S+13	(11) Transmission speed
S+14	(12) Transmission format



(1) Return code:

The execution results of the TRNS0 command are set.
Normal end → 0
Abnormal end → ≠ 0 (See Table _._ for details.)

(2) System area:

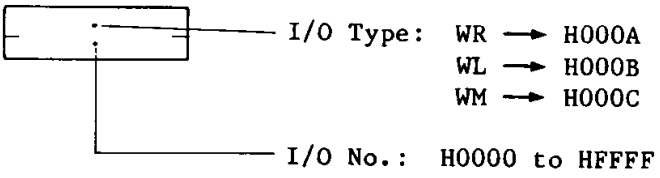
When the TRNS0 command is executed, this area is used for system processing of the TRNS0 command. This area cannot be used by the user.

(3) Time out time:

This sets the time out time from the top of the TRNS0 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(4) Top I/O of the transmission data area:

This designates top I/O type and number of the area holding the transmission data when the TRNS0 command is sent.



(5) Transmission data area size:

This designates the size of the transmission data area in word units.

(6) Top I/O of the reception data area:

After the command and data are transmitted, this designates the top I/O type and number of the area which stores the corresponding response data. (The area configuration is the same as for transmission.)

(7) Reception data area size:

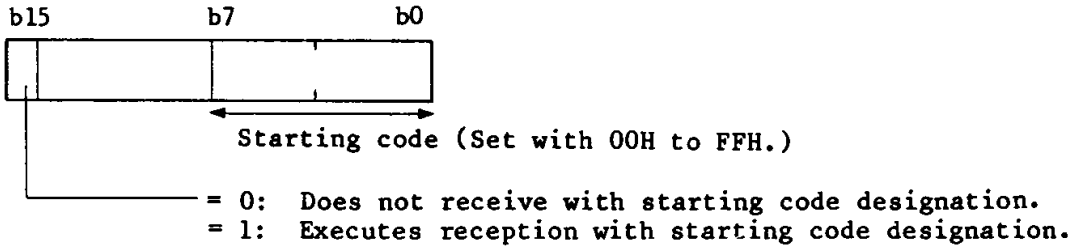
This designates the size of the reception data area in word units.

(8)*1 Reception data length:

This designates the size of reception data length in byte units. Do not, however, exceed the maximum value (256 bytes) or the reception data area. If either of these are exceeded, then DER = 1 will be true, and the process will end in an error.

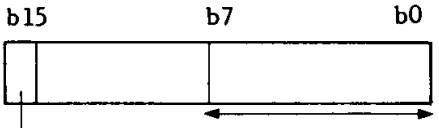
(9)*1 Starting code:

This designates the code for which reception is started.



(10)*1 Ending code:

This designates the code to end reception.



Ending code (Set with 00H to FFH.)

= 0: Does not receive with ending code designation.
= 1: Executes reception with ending code designation.

(11) Transmission rate:

This designates the baud rate.

Baud Rate Settings

Baud rate	Setting value
4800	H0004
2400	H0003
1200	H0002
600	H0001
300	H0000

(12) Transmission format:

This designates the transmission format.

Transmission Format Settings

Transmission code	Setting value
7 bit, even parity, 2 stop bits	H0000
7 bit, odd parity, 2 stop bits	H0001
7 bit, even parity, 1 stop bit	H0002
7 bit, odd parity, 1 stop bit	H0003
8 bit, no parity, 2 stop bits	H0004
8 bit, no parity, 1 stop bit	H0005
8 bit, even parity, 1 stop bit	H0006
8 bit, odd parity, 1 stop bit	H0007

Table 1. Return Code Table

Return code	Name	Contents	Action
H00	Normal end	Transmission/reception has ended normally.	
H10	Module error	Watchdog timer error	Check the fixture to the CPU basic unit, turn off/on the power supply, or replace the module.
H21	Range check error	The s parameter and t parameter of the TRNS and RECV commands exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF).	Set the parameters to their correct ranges.
H22	Transmission area setting error	The setting for the top of the transmission area (type or corresponding I/O number) is not correct.	Set the top of the transmission area so that it is within the I/O range.
H23	Transmission area range error	The bottom area of the transmission area exceeds the I/O range (WRC3FF, WL3FF, WL13FF, WM3FF).	Set the transmission area so that it fits into the I/O range.
H24	Reception area setting error	The setting for the top of the reception area (type or corresponding I/O number) is not correct.	Set the top of the reception area, so that it is within the correct I/O range.
H25	Reception area range error	The bottom area of the reception area exceeds the I/O range (WRC3FF, WL3FF, WL13FF, WM3FF).	Set the reception area range so that it is within the I/O range.
H26	Transmission data length setting error	The transmission data length setting is longer than the transmission area length.	Set the transmission data length, so that it is within the transmission area range.
H27	Reception data length setting error	The reception data length is longer than the reception area length. Or, a reception area which is not as long as necessary is not secured.	Set the reception data length so that it is within the reception area range. Secure the reception area.

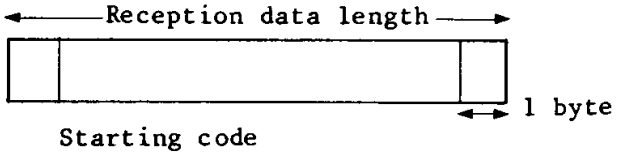
Return code	Name	Contents	Action
H28	Area overlap error (Note)	There are overlapping areas within the s parameters, t parameters, transmission areas, or reception areas of the TRNS and RECV commands.	Set the parameters or transmission/reception areas so that they do not overlap.
H29	1 scan time out value error	The time for execution of one scan is set with the QTRNS and QRECV commands, however, the time set exceeds the maximum value (6.55 seconds).	Set the one scan time out value so that it is equal to or smaller than the maximum value.
H2A	Control data setting error	Control data other than that specified was set and communications executed. (Generated only by TRNS3 and QTRNS3.)	Use the control data which is specified.

Return code	Name	Contents	Action
H30	Time out	The transmission/reception processes of the TRNS or RECV command was not completed within the specified time.	
H40	Reception area full	The reception data has filled the data area, and there is no more open area to store the next reception data.	Increase the size of the reception area.
H41	Parity error	A parity error has been generated during TRNS0 and RECV0 communication processing.	Check the transmission path of the general purpose serial port, data format, etc.
H42	Framing error	A framing error has been generated during TRNS0 and RECV0 communication processing.	
H43	Overrun error	A overrun error has been generated during TRNS0 and RECV0 communication processing.	
H44	Contention error	The TRNS0 and RECV0 commands have been generated simultaneously in two or more locations in the user program.	Make sure that the TRNS0 and RECV0 communications commands are only executed in one location in the user program.
H80	Module unique error	The high function module has received transmission data and detected an error.	Refer to the instruction manual of the module.

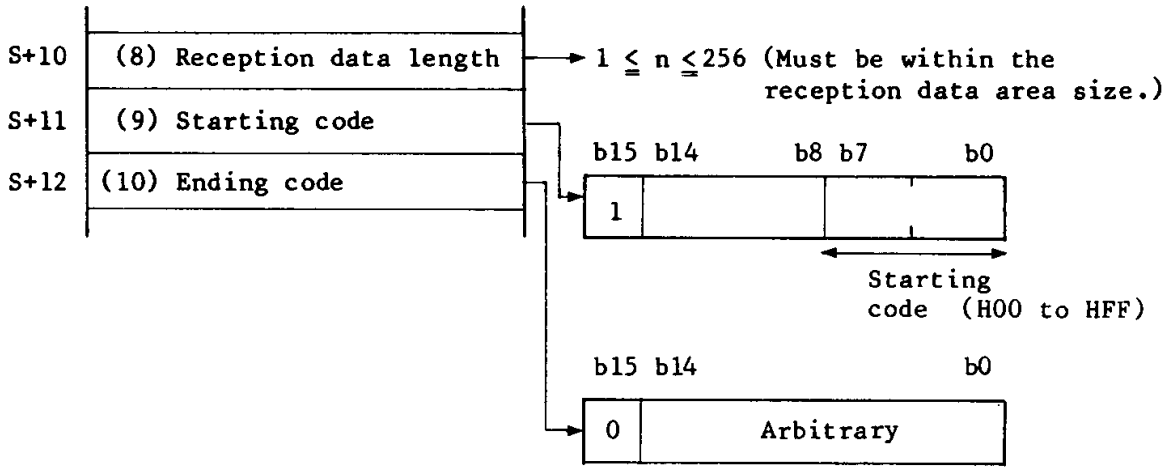
*1. The four following methods can be designated as the data reception method.

(a) Designating the starting code and reception data length.

1. Reception data configuration

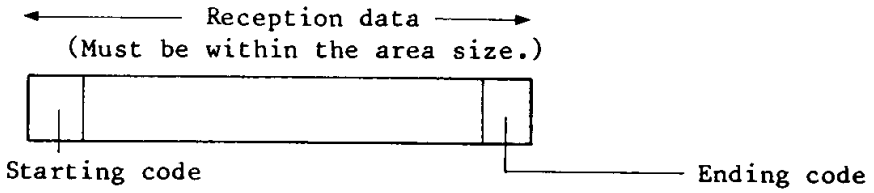


2. Parameter settings

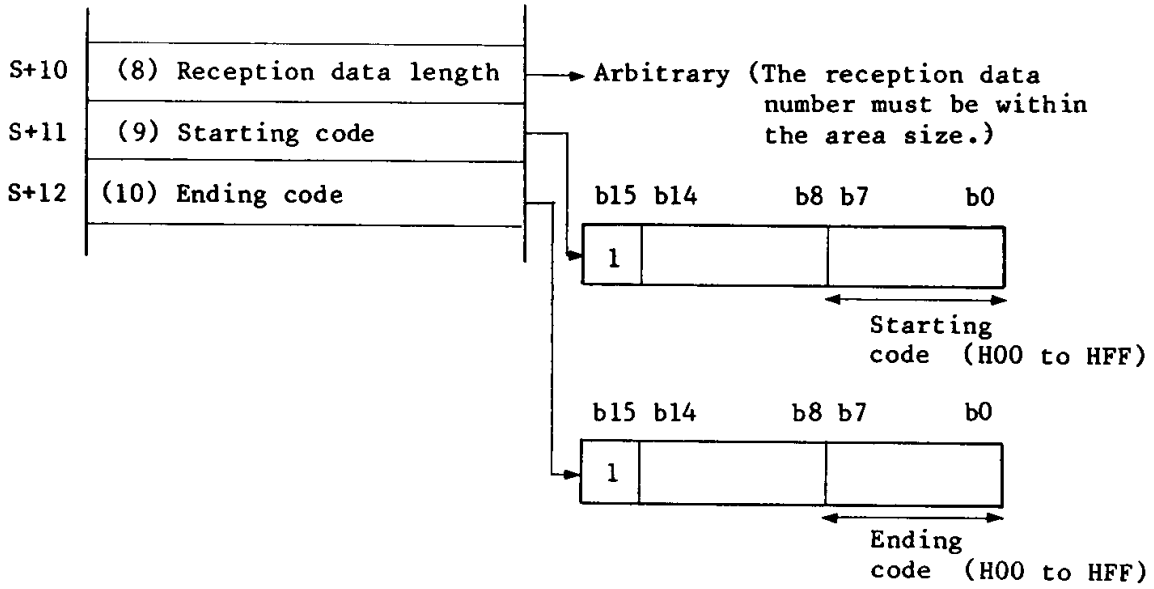


(b) Designating the starting code and ending code.

1. Reception data configuration

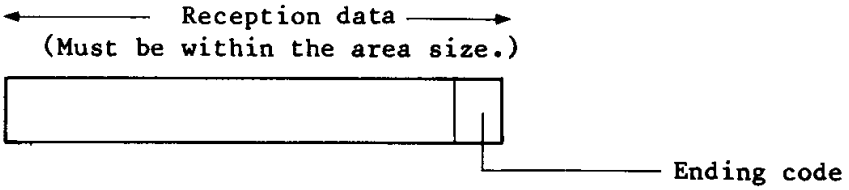


2. Parameter settings



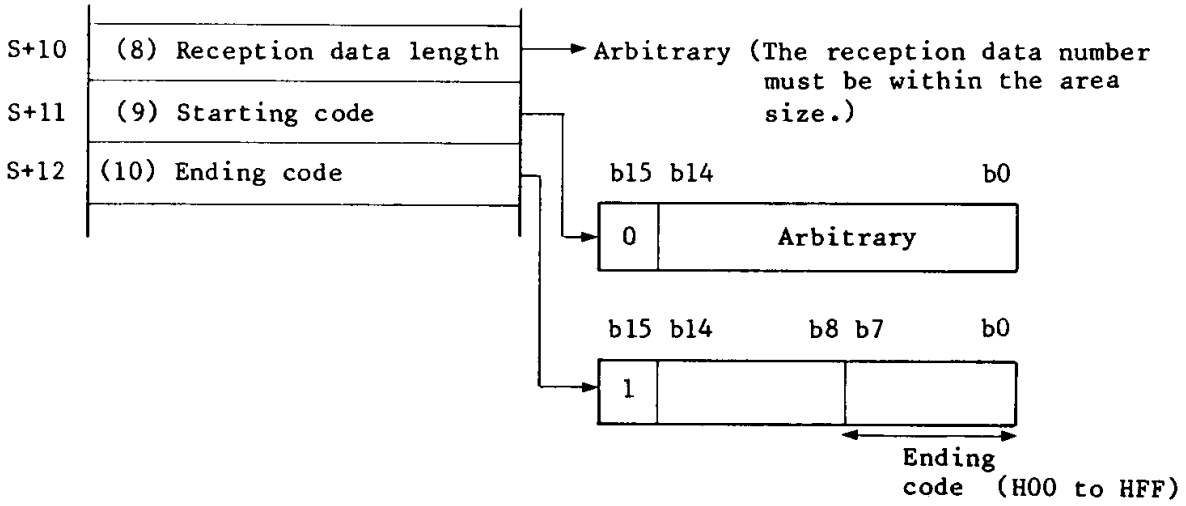
(c) Designating the ending code.

1. Reception code configuration



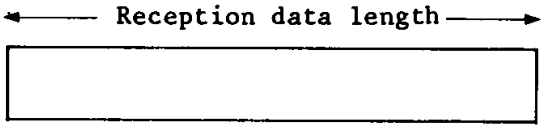
TRSN 0

2. Parameter settings

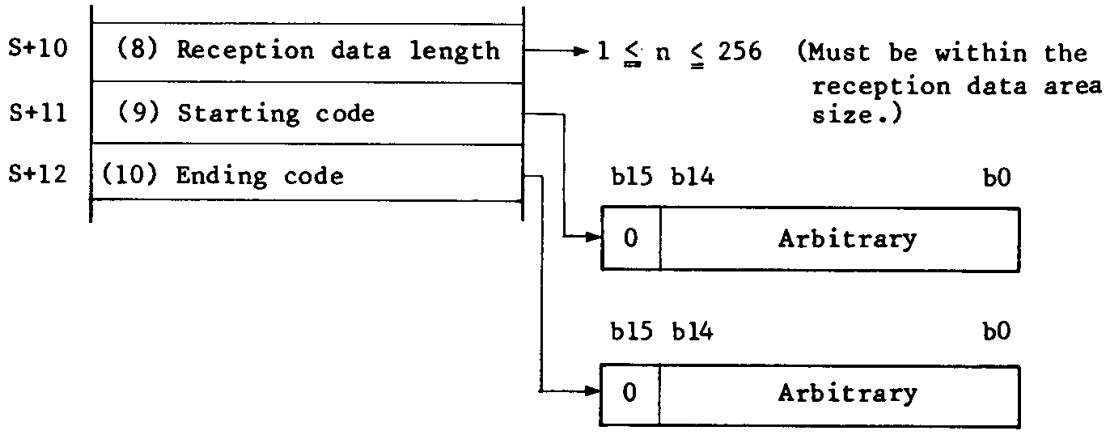


(d) Designating the reception data length.

1. Reception data configuration



2. Parameter settings



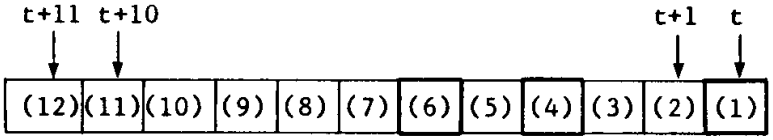
Priority of the starting code, ending code and reception data length.

Starting code	0	1	0	1
Ending code	0	0	1	1
Operation	Received by the reception data length.	Received by the starting code and reception data length.	Received by the ending code.	Received by the code. Data length ignored.

6. Description of t area

(1)

(2)



(□: Bits set by the user.)

(1) Communication execution:

When communication is executed with the TRNS0 command, the user program should set "1." When communication is ended, the TRNS0 command should be reset to "0."

(2) Normal end:

When communications with the TRNS0 command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the TRNS0 command ends in an error, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request:

When the TRNS0 command is set to the initial state, this is set to "1," however, if an initial request is output during communication, communication will be forced to end.

(5) Initialization end:

When initialization of the TRNS0 command ends normally, this bit is set to "1." At this time the initialization request (4) is reset to "0."

(6) Continue:

If reception is to follow transmission, this is set to "1." After communication is completed, the TRNS0 command will reset this to "0."

(7) Parity error:

If a parity error is generated during communications, then "1" will be set.

(8) Framing error:

If a framing error is generated during communications, then "1" will be set.

(9) Overrun error:

If a overrun error is generated during communications, then "1" will be set.

(10) Time out:

When communications reaches time out, "1" will be set.

(11) Input buffer full:

When the reception buffer is full, "1" will be set.

(12) Contention error:

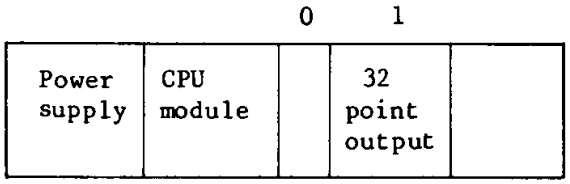
When the TRNS0 command is activated by two or more locations in the user program, "1" will be set. Communications will be forced to end. This is the same if the TRNS0 and RECV0 commands are activated by two or more locations.

In regard to (7) through (12) above, during initialization and activation of the TRNS0 command, TRNS0 will reset these to "0."

Programming Example

A programming example is shown in which data is output from the CPU module through the general purpose port to external devices using the TRNSO command.

1. Module installation



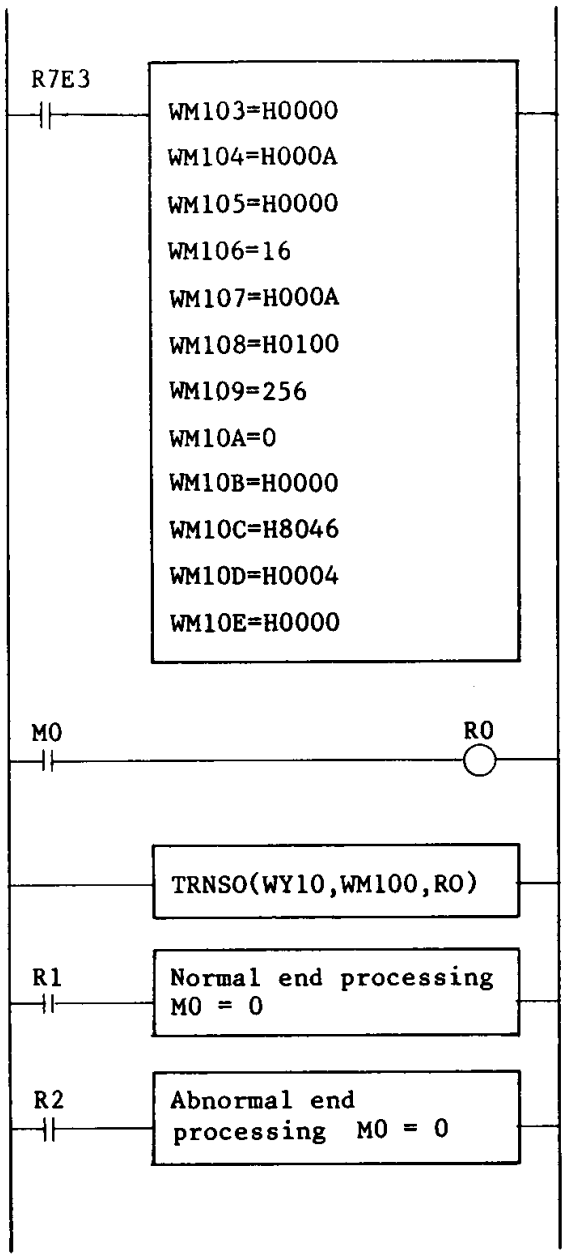
The 32 point output module is installed in slot "1" of the basic unit. Therefore, the I/O assignment of the output module is WY10 to 11.

The d parameter of the TRNSO command must designate WY10.
(The TRNSO command will use this as a dummy, so that other WYs can also be used.)

2. Internal input/output assignment

A programming example will be shown using the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
WM	0100 to 010E	TRNSO command parameter area (s to s+14)	
R	0000 to 000B	TRNSO command communication control bit area (t to t+11)	
WR	0000 to 000F	Transmission data area (16 words)	
	0100 to 01FF	Reception data area (256 words)	
M	0000	Communication request flag (TRNSO execution)	



(00001)
No time out
Transmission area top designation
: Transmission area size
Reception area top designation
: Reception area size
: Reception data length
: Starting code designation
: Ending code designation -- if yes "F"
: Baud rate: 4800 bits/second
: Transmission code: 7 bit, even,
2 stop bits

(00002)
SET

(00003)

(00004)

(00005)

TRNS 0

Program Description

1. Special internal output R7E3 (1 scan on) is used to set the TRNS0 command parameters.

2. If communication request flag M0 is set, then communication execution flag M0 is turned on.

3. Communications processing is executed for the general purpose port and external device.

4. When the TRNS0 normal end flag R1 is set, the end process starts and the communications request flag M0 is cleared.

5. When the TRNS0 abnormal end flag R2 is set, the abnormal end process starts and the communications request flag M0 is cleared.

Name

Ladder format	Condition code					Processing time (μs)				Remarks
RECV0 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
RECV0 (d,s,t)	Conditions		Step			Average	Maximum	Average	Maximum	
	-		5			96	177	180	334	

Usable I/O		Bit				Word				Double word			Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location					o									
s	Top of parameter area							o							In regard to "s," up to s + 14 is used.
t	Top of communications control bit area			o											In regard to "t," up to t + 11 is used.

RECV 0

Function

1. This is the communication command for the general purpose port used by the CPU ladder program.

2. The top I/O No. of the parameter area is set in s, which in turn sets the various parameters (top of the transmitted/received data area and size, time out value, transmission code) used for communications.

3. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

4. The arbitrarily mounted WX is set in d.

5. The RECV0 command can transmit after reception.

Precautions

. Array variables cannot be used in d, s and t.

. Use s + 14 and t + 11 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area

S	(1) Return code
S+1	(2) System area
S+3	(3) Time out time
S+4	(4) Top I/O of the transmission data area
S+6	(5) Transmission data area size
S+7	(6) Top I/O of the transmission data area
S+9	(7) Reception data area size
S+10	(8) Reception data length
S+11	(9) Starting code
S+12	(10) Ending code
S+13	(11) Transmission speed
S+14	(12) Transmission format

(: User write prohibit area
: User setting area)

(1) Return code:

The execution results of the RECV0 command are set. (Refer to TRNS0 for details.)
Normal end → 0
Abnormal end → ≠ 0

(2) System area:

When the RECV0 command is executed, this area is used for system processing of the RECV0 command. This area cannot be used by the user.

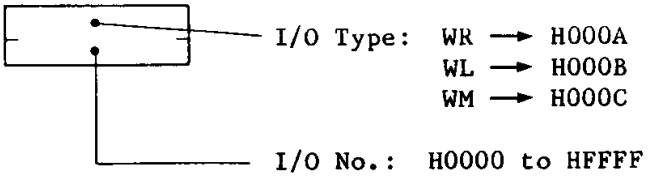
(3) Time out time:

This sets the time out time from the top of the RECV0 command execution to its completion.

= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(4) Top I/O of the transmission data area:

This designates top I/O type and number of the area holding the transmission data when the RECV0 command is to transmit after reception of data.



(5) Transmission data area size:

This designates the size of the transmission data area in word units.

(6) Top I/O of the reception data size:

This designates the I/O type and number of the top of the area which stores the reception data. (The area configuration is the same as for transmission.)

(7)*1 Reception data area size:

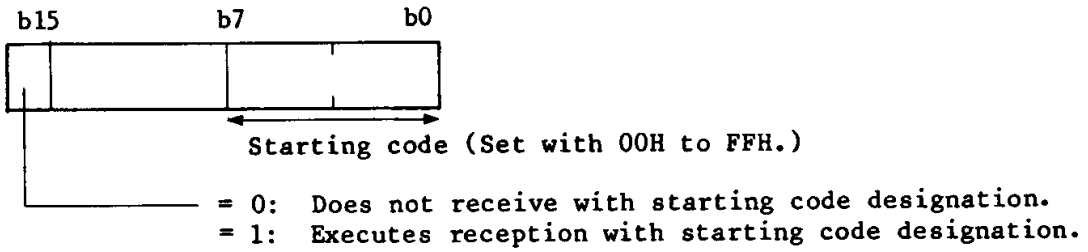
This designates the size of the reception data area in word units.

(8)*1 Reception data length:

This designates the size of reception data length in byte units. Do not, however, exceed the maximum value (256 bytes) or the reception data area. If either of these are exceeded, then DER = 1 will be true, and the process will end in an error.

(9)*1 Starting code:

This designates the code for which reception is started.



(10) Ending code:

This designates the code to end reception.

Ending code (Set with 00H to FFH.)

= 0: Does not receive with ending code designation.
≠ 1: Executes reception with ending code designation.

(11) Transmission rate:

This designates the baud rate.

(12) Transmission format:

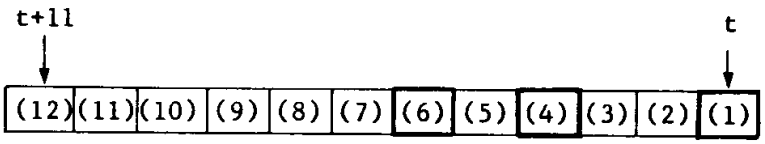
This designates the transmission format.

} Refer to TRNS0.

Set (4) and (5) if you wish to transmit after receiving with the RECV0 command.

*1 Refer to TRNS0.

6. Description of t area



(□ : Bits set by the user.)

(1) Communication execution:

When communication is executed with the RECV0 command, the user program should set "1." When communication is ended, the RECV0 command should be reset to "0."

(2) Normal end:

When communications with the RECV0 command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the RECV0 command ends in an error, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request:

When the RECV0 command is set to the initial state, this is set to "1," however, if an initial request is output during communication, communication will be forced to end.

(5) Initialization end:

When initialization of the RECV0 command ends normally, this bit is set to "1." At this time the initialization request (4) is reset to "0."

(6) Continue:

If transmission is to follow reception, this is set to "1." After communication is completed, the RECV0 command will reset this to "0."

(7) Parity error:

If a parity error is generated during communications, then "1" will be set.

(8) Framing error:

If a framing error is generated during communications, then "1" will be set.

(9) Overrun error:

If a overrun error is generated during communications, then "1" will be set.

(10) Time out:

When communications reaches time out, "1" will be set.

(11) Input buffer full:

When the reception buffer is full, "1" will be set.

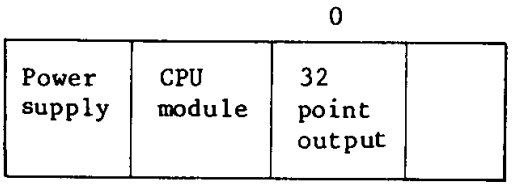
(12) Contention error:

When the RECV0 command is activated by two or more locations in the user program, "1" will be set. Communications will be forced to end. In regard to (7) through (12) above, during initialization and activation of the RECV0 command, RECV0 will reset these to "0."

Programming Example

A programming example is shown in which data is input to the CPU module through the general purpose port from external devices using the RECV0 command.

1. Module installation



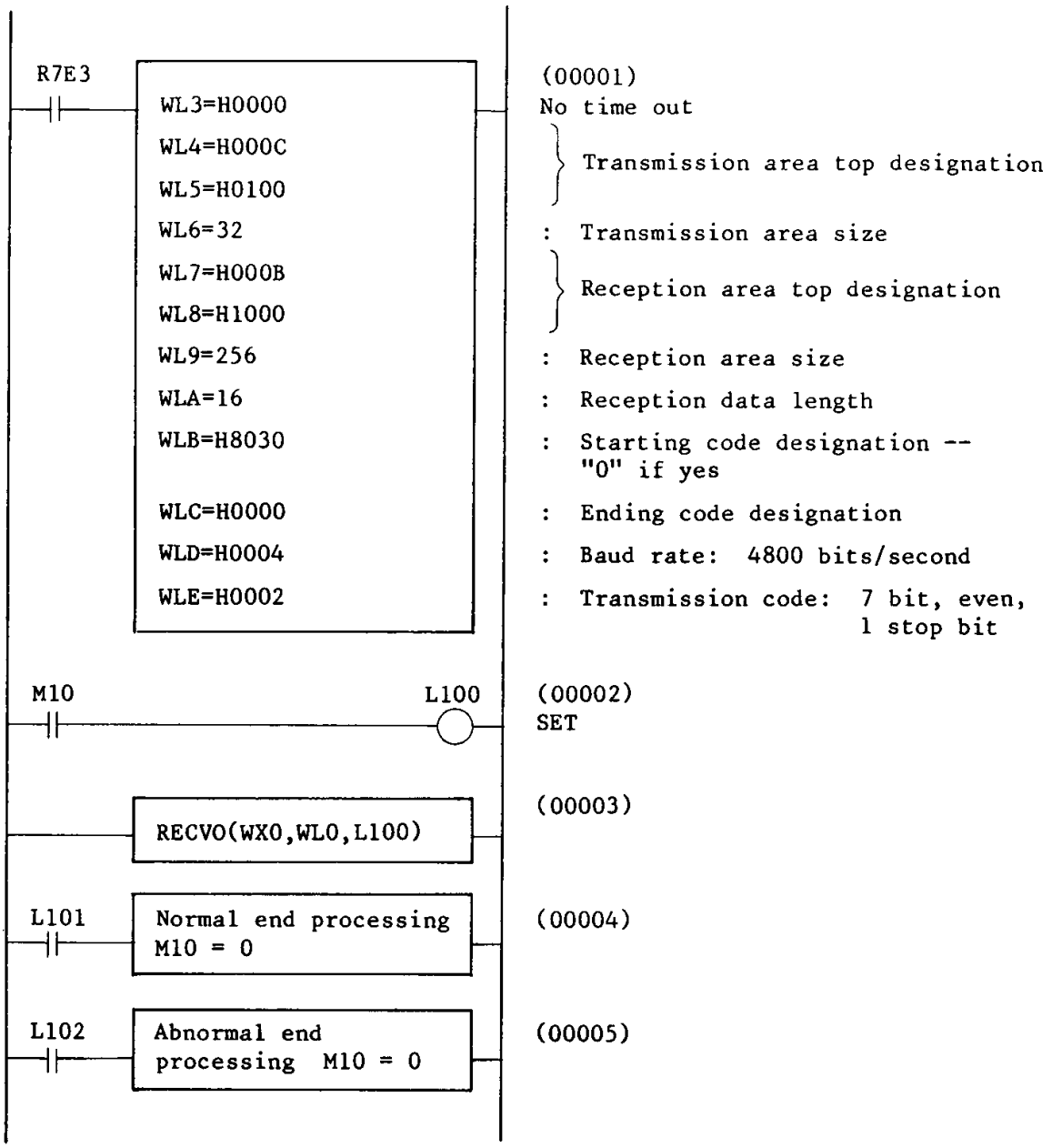
The 32 point output module is installed in slot "0" of the basic unit. Therefore, the I/O assignment of the input module is WX0-1.

The d parameter of the RECV0 command must designate WX0. (The RECV0 command will use this as a dummy, so that other WXs can also be used.)

2. Internal input/output assignment

A programming example will be shown using the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
WL	0000 to 000E	RECV0 command parameter area (s to s+14)	
L	00100 to 0010B	RECV0 command communication control bit area (t to t+11)	
WM	0100 to 0120	Transmission data area (32 words)	
WL	1000 to 10FF	Reception data area (256 words)	
M	0010	Communication request flag (RECV0 execution)	



RECV 0

Program Description

1. Special internal output R7E3 (1 scan on) is used to set the RECV0 command parameters.

2. If communication request flag M10 is set, then communication execution flag L100 is turned on.

3. Communications processing is executed for the external device and general purpose port.

4. When the RECV0 normal end flag L101 is set, the end process starts and the communications request flag M10 is cleared.

5. When the RECV0 abnormal end flag L102 is set, the abnormal end process starts and the communications request flag M10 is cleared.

Name SIO-H, CLOCK-H (REM-MMH, LMH) communications command

Ladder format	Condition code					Processing time (μs)				Remarks
TRNS1 (d, s, t) (QTRNS1 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
TRNS1 (d, s, t) (QTRNS1 (d,s,t))	Conditions			Step		Average	Maximum	Average	Maximum	
	-			5						

	Usable I/O	Bit				Word				Double word			Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o								
s	Top of parameter area							o							In regard to "s," up to s + 16 is used.
t	Top of communications control bit area			o											In regard to "t," up to t + 5 is used.

TRNS !
QTRNS !

Function

1. This is a communication command for high function modules (serial I/O (SIO-H), the clock module (CLOCK-H), and remote mini module (REM-MMH, LMH)) and the CPU. This command is used to transmit the command and related data from the CPU, then receive the response data.
2. The top I/O number of the bit area used in communications by the TRNS1 command is set in d. In regard to the installation location, the smallest number of WY for I/O assignment is designated. (Designate as WY***4.)
3. The top I/O No. of the parameter area is set in s (s to s+17 for QTRNS1), which in turn sets the various parameters (time out value, top of the transmission data area and size, top of the reception data area and size) used for communications.
4. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.
5. The TRNS1 command will transmit data one word at a time with one scan. The QTRNS1 command will communicate data for the duration set by the parameter with one scan.

Precautions

- Array variables cannot be used in d, s and t.
- Use s + 16 and t + 5 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area

S	(1) Return code
S+1	(2) Status
S+2	(3) System area (not usable by users)
S+10	**(4) Time out time**
S+11	**(5) Top I/O of the transmission data area**
S+13	**(6) Transmission data area size**
S+14	**(7) Top I/O of the reception data area**
S+16	**(8) Reception data area size**
S+17	**(9) Execution time**

TRNS1 does not use this

(□ : User access prohibited area
□ : User setting area)

(1) Return code:

The execution results of the TRNS1 command are set.
Normal end ⟶ 0
Abnormal end ⟶ ≠ 0 (See TRNS0 for details.)

(2) Status:

The contents of the status words (WX***0) of the various high function modules are set.

(3) System area:

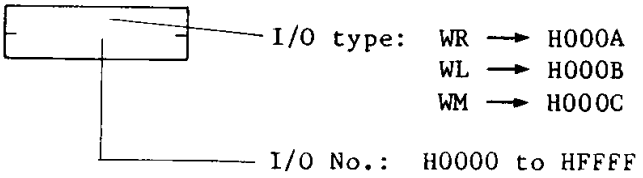
When the TRNS1 command is executed, this area is used for system processing. This area cannot be used by the user.

(4) Time out time:

This sets the time out time from the top of the TRNS1 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(5) Top I/O of the transmission data area:

This designates top I/O type and number of the area holding the transmission data when the TRNS1 command is sent.

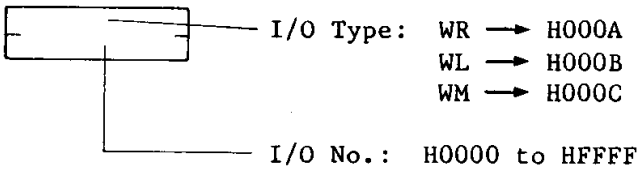


(6) Transmission data area size:

This designates the size of the transmission data area in word units.

(7) Top I/O of the reception data area:

After the command and data are transmitted, this designates the top I/O type and number of the area which stores the corresponding response data.



(8) Reception data area size:

This designates the size of the reception data area in word units.

(9) Execution time:

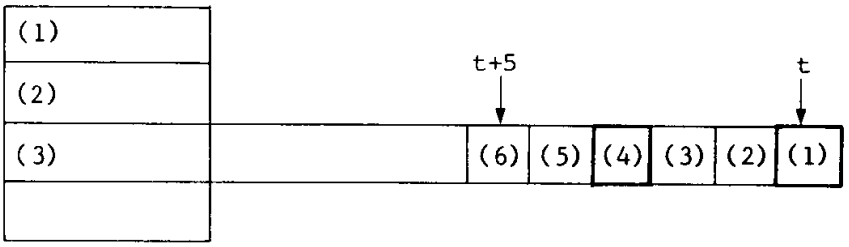
This sets the communication time executed by one scan of the QTRNS1 command.

(*ms) If 0 is set, then the operation is the same as TRNS1.

Precaution

• The system area (3) must not be used by the user program.

6. Description of t area



(□: Bits set by the user.)

(1) Communication execution:

When communication is executed with the TRNS1 command, the user program should set "1." When communication is ended, the TRNS1 command should be reset to "0."

(2) Normal end:

When communications with the TRNS1 command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the TRNS1 command ends in an error or there is an error in a parameter, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request:

When the TRNS1 command is set to the initial state, this is set to "1."

(5) Initialization end:

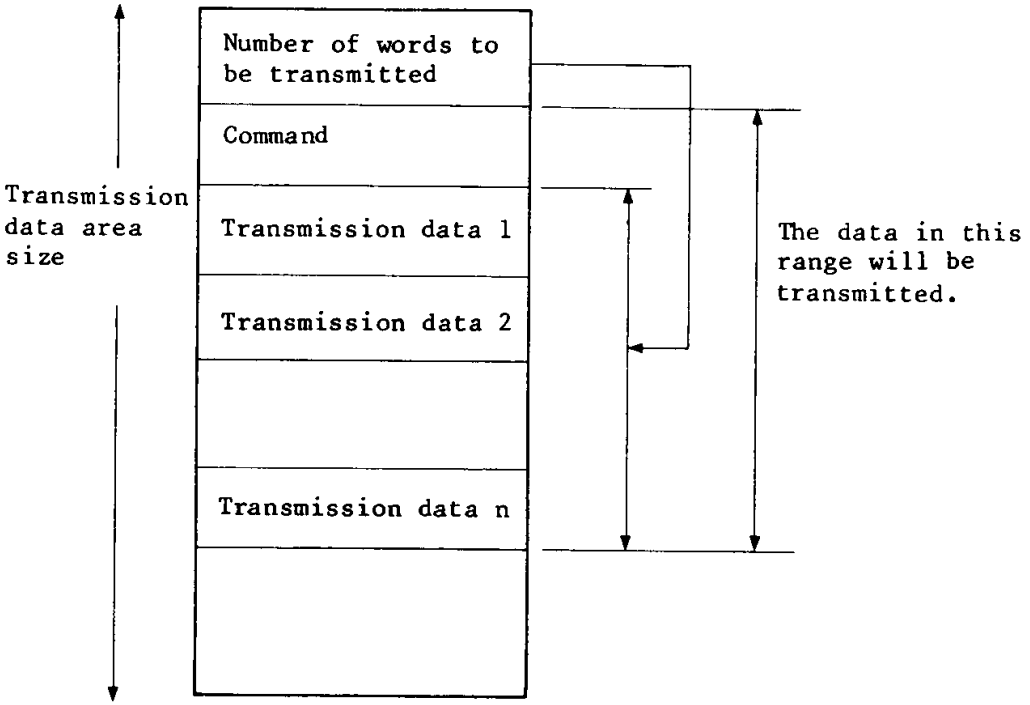
When initialization of the TRNS1 command ends normally, this bit is set to "1."
At this time

(6) Module initialization request:

When the reset switch of the installed high function module is depressed and the module requests the initialization process of the CPU, "1" is set. When this bit is ON, the CPU will use the TRNS1 command to execute the initialization process (generation of the initialization command, etc.) for the module.

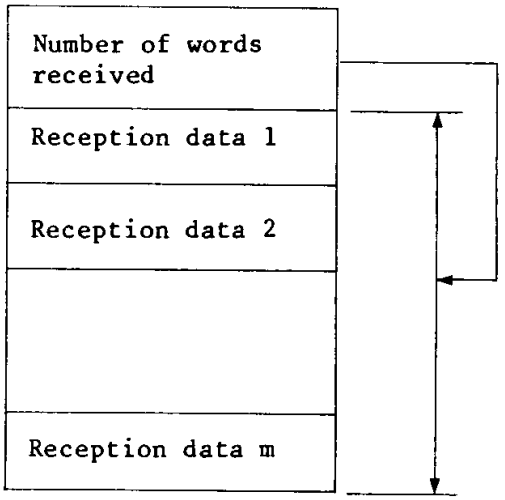
7. Description of the transmission data area

This is the area which stores the command and data which is transmitted by the TRNS1 command. Set the data to be transmitted in the high function module according to the following configuration.



8. Description of the reception data area

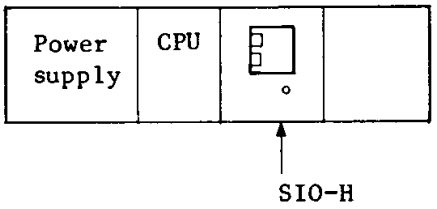
This is the area for the reception data, including the command and data, transmitted by the TRNS1 command. This is set according to the following configuration.



Programming Example 1

In this programming example, the serial I/O module (SIO-H) is used to output data in the CPU to an external device.

1. SIO-H Installation



SIO-H is installed in slot 0 of the basic unit. Therefore, the I/O assignment of SIO-H is WX0 to 3 and WY4 to 7.
Designate WY4 as the d parameter of the TRNS command.

2. Internal output assignment

The programming example will use the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
WM	010 to 020	TRNS1 command parameter area (s to s+16)	
M	0000 to 0005	TRNS1 command communication control bit area (t to t+5)	
WR	0100 to 011F	Transmission data area (32 words)	WR0100: Number of transmission data WR0101: Command code WR0102: Command execution data
	0200 to 021F	Reception data area (32 words)	WR0200: Number of reception data WR0201: Reception data
R	00000	TRNS1 command initialization request	
	00001	Operation mode setting request	
	00002	Data output request	
	00100	Communications request (TRNS1 execution)	

3. Data to be passed from the CPU to SIO-H

In regard to programming example 1, the command and data which are to be set in the transmission data area as the data to be passed from the CPU to SIO-H are shown below. Refer to the SIO-H manual for details.

(a) Initialization process command and data

I/O No.	Data	Contents	Remarks
WR0100	H0000	Number of transmission data	
WR0101	H0000	Command code	Initialization command

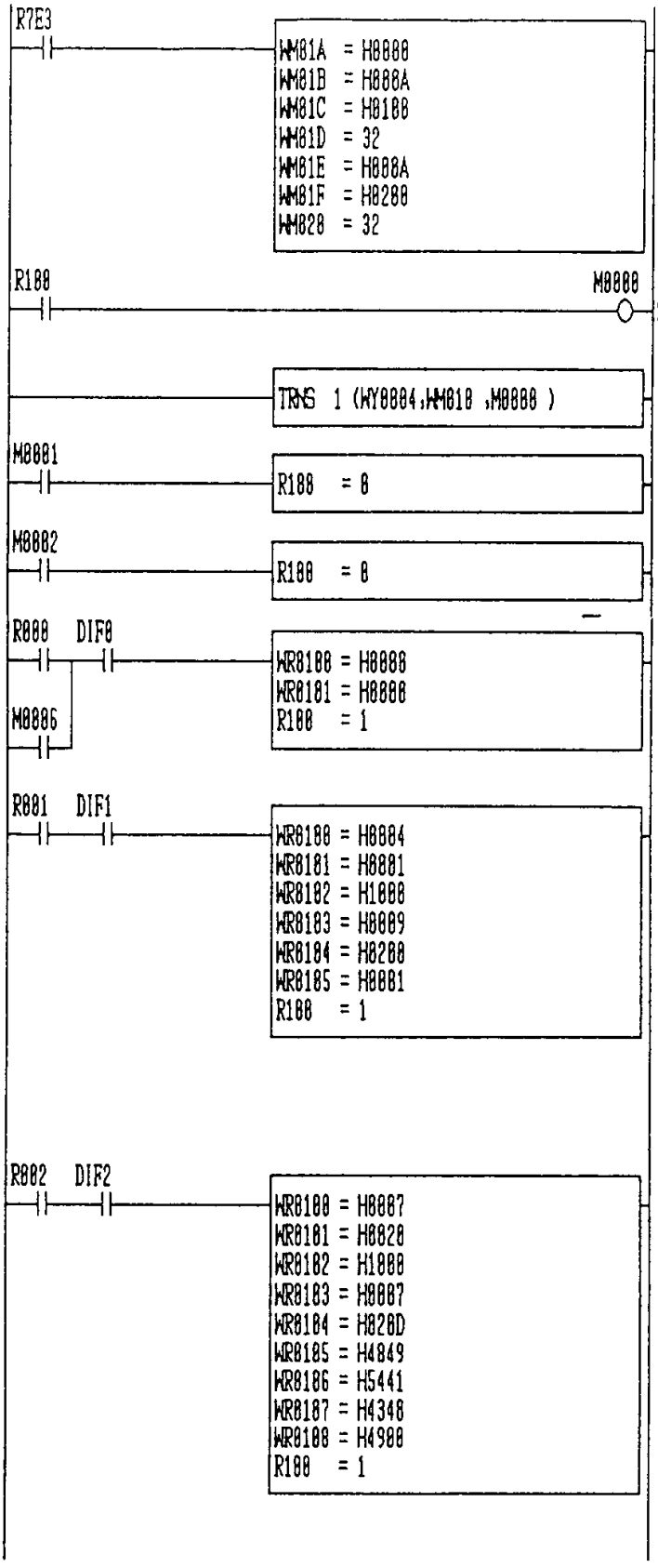
(b) Mode setting processor command and data

I/O No.	Data	Contents	Remarks
WR0100	H0004	Number of transmission data	
WR0101	H0001	Command code	
WR0102	H1000	Port type, time out	RS232C/no time out setting
WR0103	H0008	Mode setting	After executing the output command, clear the buffer.
WR0104	H0200	Output buffer empty flag operation condition	Controlled by the buffer empty byte number (H0200)
WR0105	H0001	Input buffer valid flag operation condition	Controlled by the received message length

TRNS I
QTRNS I

(c) Output processor command and data to the output buffer

I/O No.	Data	Contents	Remarks
WR0100	H0007	Number of transmission data	
WR0101	H0020	Command code	
WR0102	H1000	Port type, time out	
WR0103	H0007	Data type, data length	Byte data, 7 byte
WR0104	H020D	Message configuration	STX + data + CR configuration
WR0105	H4849	Data (ASCII)	"H", "I"
WR0106	H5451	↓	"T", "A"
WR0107	H4348	↓	"C", "H"
WR0108	H4900	↓	"I"



TRNS1 parameter setting
No time out
Transmission data I/O WR designation
Transmission data I/O No. 100 designation
Transmission data I/O size 32 words
Reception data I/O WR designation
Reception data I/O No. 200 designation
Reception data I/O size 32 words

Execute normal end processing, then clear R100.
Execute error processing, then clear R100.
Initialization process

Operation mode setting process

Port type (RS232C)
After executing the output command, clear the buffer.
Output buffer empty flag operation condition
Input buffer valid flag operation condition

Data output processor

Port type (RS232C)
Byte data, 7 byte
STX + [ ] + CR
Data Hi
Data TA
Data CH
Data i

Program Description 1

1. Use special internal output R7E3 (1 scan on) to set the TRNS1 command parameters (no time out check, transmission area 32 words from WR100, reception 32 words from WR200) in the first scan after starting operation.

2. When the communications start condition R100 is set, communication execution flag M0 is turned on, and the TRNS1 command is executed.

3. Communications processing is executed for the serial I/O module.

4. When the normal end flag M1 is set, normal end processing is executed and the communication start condition R100 is cleared.

5. When the abnormal end flag M2 is set, abnormal end processing is executed and the communication start condition R100 is cleared.

6. When the initialization start condition R0 is set, the initialization command of the serial I/O module is set in the transmission data area, and the communication start condition R100 is turned on.

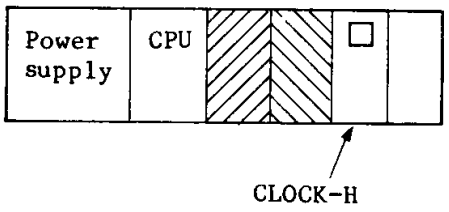
7. When the operation mode setting condition R1 is set, the operation mode setting command of the serial I/O module and the data are set in the transmission data area, then communication start condition R100 is turned on.

8. When data output condition R2 is set, the data output command of the serial I/O module and the data are set in the transmission data area, then communication start condition R100 is turned on.

Programming Example 2

In this programming example, the real time clock module (CLOCK-H) is used to set the clock data and read it.

1. CLOCK-H installation



CLOCK-H is installed in slot 2 of the basic unit. Therefore, the I/O assignment of CLOCK-H is WX20 to 23 and WY24 to 27.
Designate WY24 as the d parameter of the TRNS1 command.

2. Internal output assignment

The programming example will use the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
WM	030, 031	Hour, minute, and second data	
	110 to 120	TRNS1 command parameter area (s to s+16)	
M	1000 to 1005	TRNS1 command communication control bit area (t to t+5)	
WR	0300 to 030F	Transmission data area (16 words)	WR0100: Number of transmission data WR0101: Command code WR0102: Command execution data
	0400 to 040F	Reception data area (16 words)	WR0200: Number of reception data WR0201: Reception data
R	00010	Clock data write	
	00011	Clock data read	
	00110	Communication request (TRNS1 execution)	

3. Data to be passed from the CPU to SIO-H

In regard to programming example 1, the command and data which are to be set in the transmission data area as the data to be passed from the CPU to CLOCK-H are shown below.

(a) Clock data write

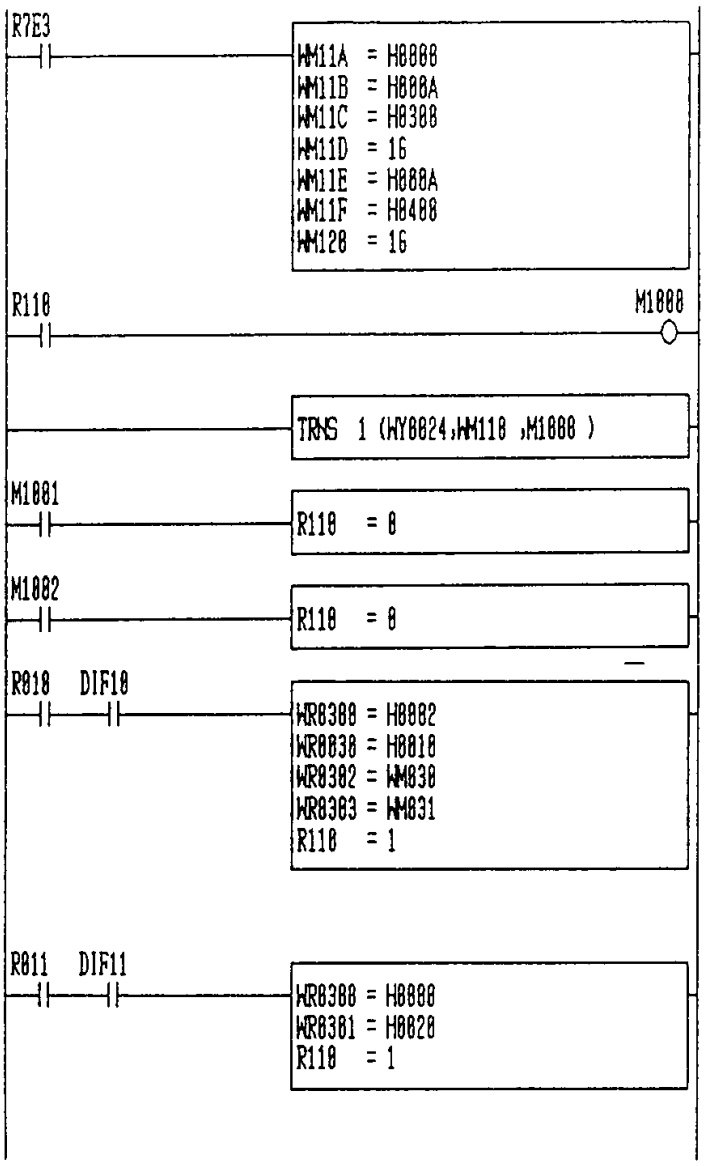
I/O No.	Data	Contents	Remarks
WR0300	H0002	Number of transmission data	
WR0301	H0010	Command code	Clock data setting command
WR0302	WM030	Hour and minute data	
WR0303	WM031	Second data	

(b) Clock data read

I/O No.	Data	Contents	Remarks
WR0300	H0000	Number of transmission data	
WR0301	H0020	Command code	Clock data read command

The clock data which is read is set in WR401 and WR402.

WR400	H0002	
WR401	Hour	Minute
WR402	Second	00



TRNS1 parameter setting
No time out
Transmission data I/O WR designation
Transmission data I/O No. 300 designation
Transmission data I/O size 16 words
Reception data I/O WR designation
Reception data I/O No. 400 designation
Reception data I/O size 16 words

Normal end processing is executed, then R110 is cleared.
Error processing is executed, then R110 is cleared.
Clock write

Clock read

Program Description 1

1. Use special internal output R7E3 (1 scan on) to set the TRNS1 command parameters (no time out check, transmission area 16 words from WR300, reception 16 words from WR400) in the first scan after starting operation.

2. When the communications start condition R110 is set, communication execution flag M1000 is turned on, and the TRNS1 command is executed.

3. Communications processing is executed for the real time clock module.

4. When the normal end flag M1001 is set, normal end processing is executed and the communication start condition R110 is cleared.

5. When the abnormal end flag M1002 is set, abnormal end processing is executed and the communication start condition R110 is cleared.

6. When the clock data write conditions (R10) are set, the hour, minute and second data set in WM30 and WM31 are set in the transmission data area, and the communication start condition with the real time clock module R110 is turned on.

7. When the clock data read condition (R11) is set, the clock data read command of the real time clock module is set in the transmission data area, and the communication start condition R110 is turned on. The read hour, minute and second data are written in WR401 and WR402.

Name ASCII module communication command

Ladder format	Condition code					Processing time (μs)				Remarks
TRNS2 (d,s,t) (QTRNS2 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps			H-2002		H-702/302				
TRNS2 (d,s,t) (QTRNS2 (d,s,t)	Conditions		Step	Average	Maximum	Average	Maximum			
	—		5		850 (4710)		1420 (5910)			

Usable I/O		Bit				Word				Double word			Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o								
s	Top of parameter area							o							In regard to "s," up to s + 20 is used.
t	Top of communications control bit area			o											In regard to "t," up to t + 5 is used.

TRNS 2
QTRNS 2

Function

1. This is the communication command used between the high function module (ASCII-1H, 2H) and the CPU.

2. In parameter d, the smallest WY is designated for the installation location I/O assignment WY of ASCII-1H, 2H with which the TRNS2 (QTRNS2) command will communicate. (Designate WY***4.)

3. The top I/O numbers of the parameter areas for the various parameters (time out data, top I/O numbers and sizes of transmission/reception data areas, Tms counter data) for communication are set in s.

4. The control bit which starts and initializes communications and clears errors, and top I/O No. of the communication control bit area in which the results of communications are set in t.

5. The TRNS2 command sends data one word per one scan. The QTRNS2 command sends the period data of time set by the parameters in one scan.

Precautions

. Array variables cannot be used in d, s and t.

. Use s + 21 and t + 5 so that they do not exceed the I/O range (R7FF, L03FF, L13FF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area

S	(1) Return code
S+1	(2) Status
S+2	(3) System area
S+14	(4) Time out time
S+15	(5) Top I/O of the transmission data area
S+17	(6) Transmission data area size
S+18	(7) Top I/O of the reception data area
S+20	(8) Reception data area size
S+21	(9) 1 scan execution time

TRNS2 does not use this.

(1) Return code:

The execution results of the TRNS2 command are set.
Normal end ⟶ 0
Abnormal end ⟶ ≠ 0 (See TRNS0 for details.)

(2) Status:

The contents of the status words (WX***0) of ASCII-1H, 2H are set.

(3) System area:

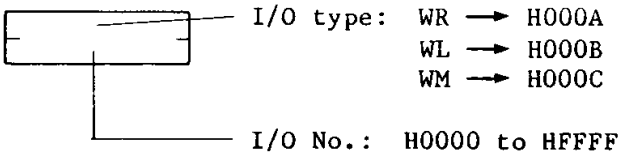
When the TRNS2 (QTRNS2) command is executed, this area is used for system processing. This area cannot be used by the user.

(4) Time out time:

This sets the time out time from the top of the TRNS2 (QTRNS2) command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(5) Top I/O of the transmission data area:

This designates top I/O type and number of the area holding the transmission data when the TRNS2 (QTRNS2) command is sent.

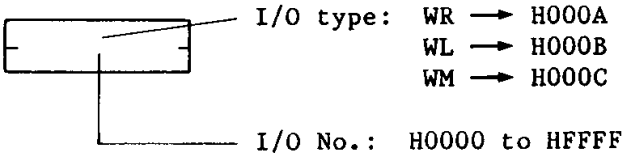


(6) Transmission data area size:

This designates the size of the transmission data area in word units.

(7) Top I/O of the reception data area:

This designates the top I/O type and number of the area which stores the data to be received by TRNS2 (QTRNS2).



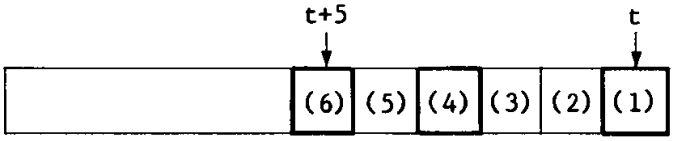
(8) Reception data area size:

This designates the size of the reception data area in word units.

(9) 1 scan execution time: (*ms)

This sets the communication time when one scan of QTRNS2 is executed. If 0 is set, then the operation will be the same as that for TRNS2.

6. Description of the t area



(1) Communication execution:

When communication is executed with the TRNS2 (QTRNS2) command, the user program should set "1." When communication is ended, the TRNS2 (QTRNS2) command should be reset to "0."

(2) Normal end:

When communications with the TRNS2 (QTRNS2) command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the TRNS2 (QTRNS2) command ends in an error or there is an error in a parameter, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request:

When the TRNS2 (QTRNS2) command is set to the initial state, this is set to "1." If initialization ends normally, this is reset to "0."

(5) Initialization end:

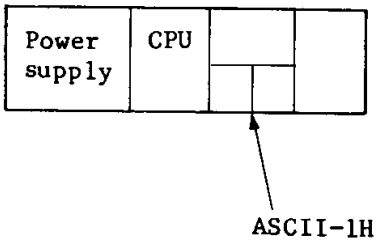
When initialization of the TRNS2 (QTRNS2) command ends normally, this bit is set to "1." The initialization request, (4) will be reset to "0."

(6) Error clear request:

In order to clear errors generated by the TRNS2 (QTRNS2) command, set this to "1."

Programming Example

An example of transmitting data with the ASCII module is shown below. The parameters are set so that SB1 in this program can be replaced with SB1 in the ASCII module sequence program.



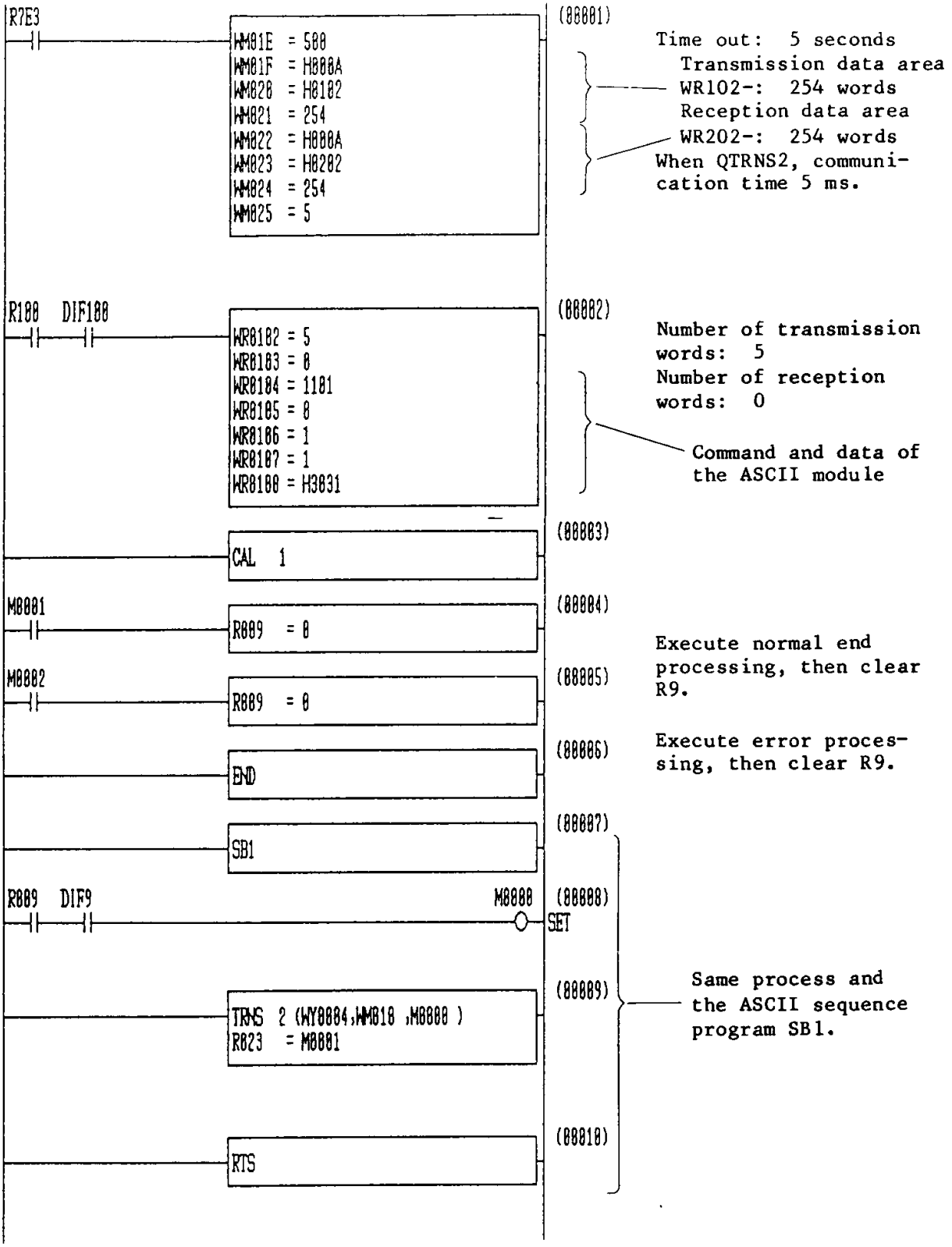
ASCII-1H is set in slots 0 and 1 of the basic unit. (This is because this module has a two slot width.) Therefore, the I/O assignments for ASCII-1H are WX10-WX13 and WY14-WY17.

WY14 is designated for the d parameter of TRNS2 (QTRNS2).

(1) Internal output assignment

The programming example will use the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
W/M	010 to 025	TRNS2 (QTRNS2) command parameter area (s to s+21)	
M	0000 to 0005	TRNS2 (QTRNS2) command communication control bit area (t to t+5)	
WR	0102 to 01FF	Transmission data area (254 words)	This program does not use FIFO, so that WR0100 and WR0101 of SB1 are not necessary. In order to match SB1, WR0102- is used.
	0202 to 021F	Reception data area (254 words)	This program does not use FIFO, so that WR0200 and WR0201 of SB1 are not necessary. In order to match SB1, WR0202- is used.
R	0009	Transmission start flag	
	0023	Transmission end flag	


R7E3
WM01E = 500
WM01F = H000A
WM020 = H0102
WM021 = 254
WM022 = H000A
WM023 = H0202
WM024 = 254
WM025 = 5
(00001)
Time out: 5 seconds
Transmission data area
WR102-: 254 words
Reception data area
WR202-: 254 words
When QTRNS2, communication time 5 ms.
R100 DIF100
WR0102 = 5
WR0103 = 0
WR0104 = 1101
WR0105 = 0
WR0106 = 1
WR0107 = 1
WR0108 = H3031
(00002)
Number of transmission words: 5
Number of reception words: 0
Command and data of the ASCII module
CAL 1
(00003)
M0001
R009 = 0
(00004)
M0002
R009 = 0
(00005)
Execute normal end processing, then clear R9.
END
(00006)
Execute error processing, then clear R9.
SB1
(00007)
R009 DIF9
M0000
SET
(00008)
TRNS 2 (WY0004,WM010 ,M0000)
R023 = M0001
(00009)
Same process and the ASCII sequence program SB1.
RTS
(00010)

Program Description

1. Use the special internal output R7E3 (1 scan on) to set the TRNS2 command parameters.

2. The data to be sent to the ASCII module is set in the transmission data area with R100.

3. Call the subroutine of the TRNS2 command.

4. When the normal end flag M1 is set, the normal end process will be executed, and the communication start flag R9 will be cleared.

5. When the error end flag M2 is set, the error process will be executed, and the communication start flag R9 will be cleared.

6. End of program.

7. Start subroutine.

8. The communication start flag of R9 is used to turn on the communication execution flag M0.

9. Communication processing is executed.

10. End of subroutine.

Name

Ladder format	Condition code					Processing time (μs)				Remarks
TRNS3 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
TRNS3 (d,s,t)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			5						

Usable I/O		Bit				Word				Double word			Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o								
s	Top of parameter area							o							In regard to "s," up to s + 10 is used.
t	Top of communications control bit area			o											In regard to "t," up to t + 4 is used.

TRNS 3

Function

1. This command is used between the one axis positioning module (POSIT-H) and the CPU. This command is used to send the command and data to the CPU, then receive the response data.

2. The smallest WY number of the I/O assignment WY for the installation location of POSIT-H which communicates with the TRNS3 command is set in d. (Designate WY***4.)

3. The top I/O number of the parameter area which sets the various parameters (time out value, top I/O number and size of the transmission data area, control data) for communications is set in s.

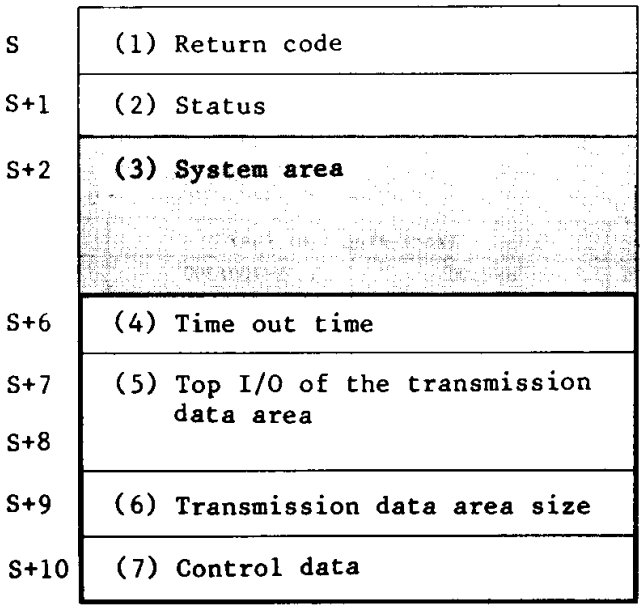
4. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

Precautions

. Array variables cannot be used in d, s and t.

. Use s + 10 and t + 4, so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area



(1) Return code:

The execution results of the TRNS3 command are set.
Normal end → 0
Abnormal end → ≠ 0 (See TRNS0 for details.)

(2) Status:

The contents of the status words (WX***0) of POSIT-H are set.

(3) System area:

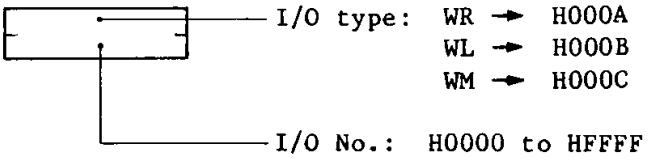
When the TRNS3 command is executed, this area is used for system processing. This area cannot be used by the user.

(4) Time out time:

This sets the time out time from the top of the TRNS3 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(5) Top I/O of the transmission data area:

This designates top I/O type and number of the area holding the transmission data when the TRNS3 command is sent.

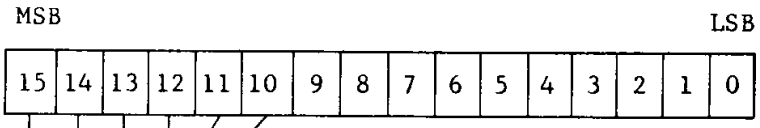


(6) Transmission data area size:

This designates the size of the transmission data area in word units.

(7) Control data:

This is set according to the table on the next depending on the contents to be communicated.

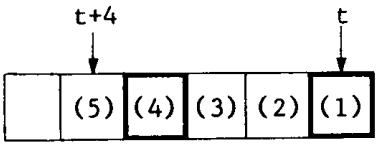


0: OFF 1: ON

Bit	Contents/definitions
bit 0 to 5	Writes the number of words to be sent. The maximum is 17 words. When only a control register is to be written, set 0.
bit 6, 7	Not used. Set this to 0.
bit 8, 9	Designates the operation mode of the positioning module.
bit 10	Operation bit. Start operation = 1. Pause = 0. The module will operate according to the leading edge and trailing edge of this bit. Once pause is executed the module will pause and will not accept operation input until it enters the standby status.
bit 11	Synchronous/asynchronous mode. Asynchronous = 0. Synchronous = 1. This determines the mode for synchronous and asynchronous operation through the MS and SL signals. Be sure to set the asynchronous mode when the MS and SL signals are not used.
bit 12	Master/slave setting. Slave axis = 0. Master axis = 1. This sets whether the positioning module is for the master axis or slave axis. If the asynchronous mode is used, however, this setting does not have to be specified.
bit 13	Communications data mode. Positioning data = 0. Common parameters = 1. Sets the type of communications data.
bit 14	Not used. Set this to 0.
bit 15	Stop. Normal = 0. Stop = 1. When this bit is set to 1, the module will be stopped. Pulse row output is stopped.

Bit		Mode
9	8	
0	0	Wait
0	1	Manual
1	0	Automatic
1	1	Zero return

6. Description of the t area



(□ : Bits set by the user.)

(1) Communication execution:

When communication is executed with the TRNS3 command, the user program should set "1." When communication is ended, the TRNS3 command should be reset to "0."

(2) Normal end:

When communications with the TRNS3 command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the TRNS3 command ends in an error or there is an error in a parameter, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request:

When the TRNS3 command is set to the initial state, this is set to "1." (This is not initialization of the high function module.) When initialization ends normally, this is reset to "0."

(5) Initialization end:

When initialization of the TRNS3 command ends normally, this bit is set to "1." At this time, the initialization request (4) will be reset to "0."

Name

Ladder format	Condition code					Processing time (μs)				Remarks
RECV3 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
RECV3 (d,s,t)	Conditions		Step			Average	Maximum	Average	Maximum	
	—		5			102	160	190	300	

Usable I/O		Bit				Word				Double word				Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o									
s	Top of parameter area							o								In regard to "s," up to s + 5 is used.
t	Top of communications control bit area			o												In regard to "t," up to t + 4 is used.

RECV 3

Function

1. This command is used between the one axis positioning module (POSIT-H) and the CPU. This command is used to receive position data indicated by the module.

2. The smallest WX number of the I/O assignment WX for the installation location of POSIT-H which communicates with the RECV3 command is set in d. (Designate WX***0.)

3. The top I/O number of the parameter area which sets the various parameters (time out value, top I/O number and size of the reception data area, control data) for communications is set in s.

4. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

Precautions

- Array variables cannot be used in d, s and t.

- Use s + 5 and t + 4 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area

S	(1) Return code
S+1	(2) Status
S+2	(3) Time out time
S+3	(4) Top I/O of the reception data area
S+4	
S+5	(5) Reception data area size

(□ : User setting area)

(1) Return code:

The execution results of the RECV3 command are set.
Normal end → 0
Abnormal end → ≠ 0 (See TRNS0 for details.)

(2) Status:

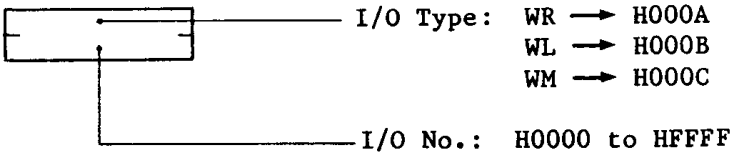
The contents of the status words (WX***0) of POSIT-H are set.

(3) Time out time:

This sets the time out time from the top of the RECV3 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(4) Top I/O of the reception data area:

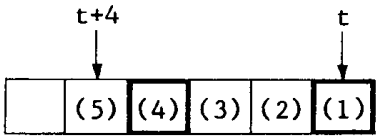
This designates top I/O type and number of the area holding the reception data when the RECV3 command used.



(5) Reception data area size:

This designates the size of the reception data area in word units.

6. Description of the t area



(☐ : Bits set by the user.)

(1) Communication execution:

When communication is executed with the RECV3 command, the user program should set "1." When communication is ended, the RECV3 command should be reset to "0."

(2) Normal end:

When communications with the RECV3 command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the RECV3 command ends in an error or there is an error in a parameter, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request:

When the RECV3 command is set to the initial state, this is set to "1." (This is not initialization of the high function module.) When initialization ends normally, this is reset to "0."

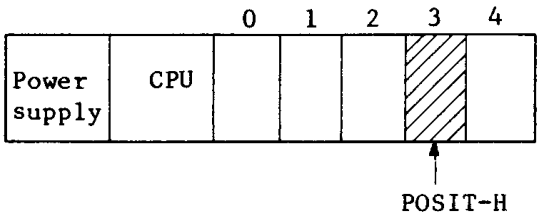
(5) Initialization end:

When initialization of the RECV3 command ends normally, this bit is set to "1." At this time, the initialization request (4) will be reset to "0."

Programming Example

This is a programming example in which a one axis positioning module (POSIT-H) is used, and goes from initialization to operation.

1. POSIT-H installation

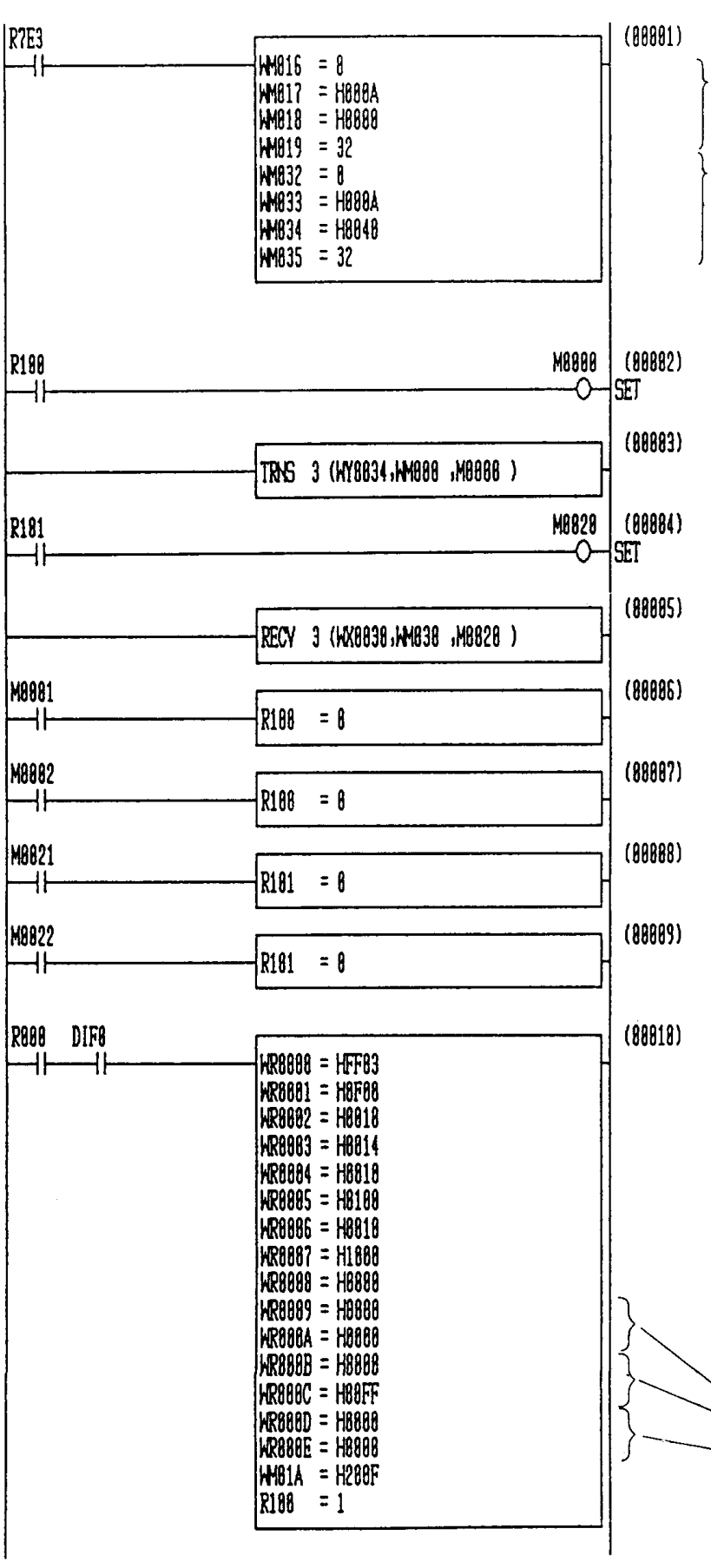


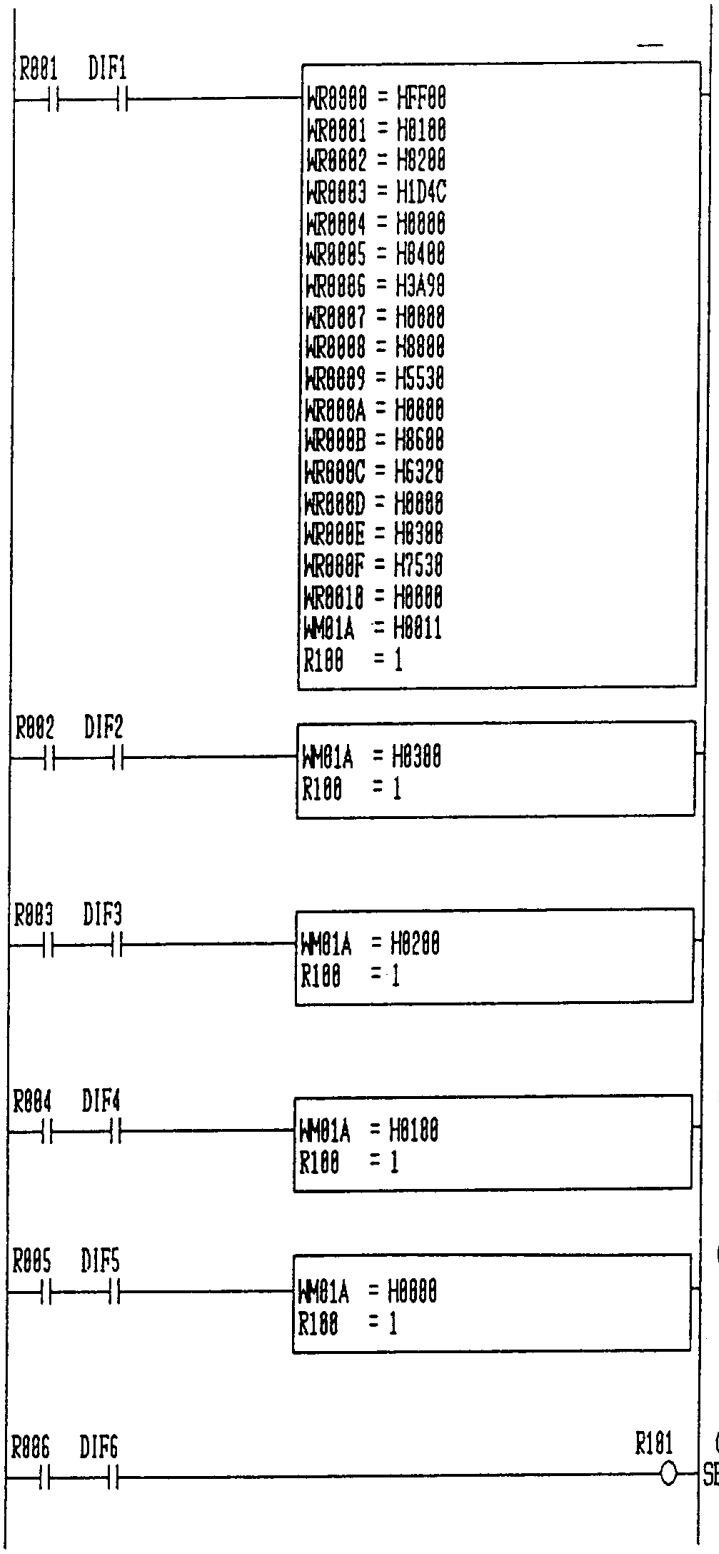
POSIT-H is installed in slot three of the basic unit. Therefore, the I/O assignments for POSIT-H are WX30 to WX33 and WY34 to WY37. WY34 is designated for the d parameter of TRNS3, and WX30 is designated for the d parameter of RECV3.

(2) Internal output assignment

The programming example will use the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
WM	010 to 01A	TRNS3 command parameter area (s to s+10)	
M	0000 to 0004	TRNS3 command communication control bit area (t to t+4)	
WR	0000 to 0020	TRNS3 command transmission data area (32 words)	
WM	030 to 035	RECV3 command parameter area (s to s+5)	
M	0020 to 0024	RECV3 command communication control bit area (t to t+4)	
WR	0040 to 0050	RECV3 command reception data area (32 words)	
R	00000	Common parameter transmission	
	00001	Positioning data transmission	
	00002	Zero return request flag	
	00003	Automatic operation request flag	
	00004	Manual operation request flag	
	00005	Stop request flag	
	00006	Receive request flag	
	00100	TRNS3 activation flag	
	00101	RECV3 activation flag	


R7E3
WM816 = 0
WM817 = H888A
WM818 = H8888
WM819 = 32
WM832 = 0
WM833 = H888A
WM834 = H8848
WM835 = 32
(00001)
TRNS3 and RECV3 parameter setting
No time out
Transmission data area
WR0-: 32 words
No time out
Reception data area
WR40-: 32 words
R100
M8000
SET
(00002)
(00003)
TRNS 3 (WY0034,WM000 ,M0000)
R101
M8020
SET
(00004)
(00005)
RECV 3 (WX0030,WM030 ,M0020)
TRNS3 normal end processing
M8001
R100 = 0
(00006)
Communication start R100 flag clear
TRNS3 error processing
M8002
R100 = 0
(00007)
Communication start R100 flag clear
RECV3 normal end processing
M8021
R101 = 0
(00008)
Communication start R100 flag clear
RECV3 error processing
M8022
R101 = 0
(00009)
Communication start R100 flag clear
R000 DIF0
WR0000 = HFF03
WR0001 = H0F00
WR0002 = H0010
WR0003 = H0014
WR0004 = H0010
WR0005 = H0100
WR0006 = H0010
WR0007 = H1000
WR0008 = H0000
WR0009 = H0000
WR000A = H0000
WR000B = H0000
WR000C = H00FF
WR000D = H0000
WR000E = H0000
WM01A = H200F
R100 = 1
(00010)
Common parameters
: Maximum speed
: Manual starting speed
: Manual speed
: Automatic starting speed
: High speed zero return speed
: Low speed zero return speed
: Acceleration/ deceleration percentage
: Backlash compensation
Upper limit
Lower limit
Zero point
Control data
Common parameters, number of transmission words.



Positioning data
: Automatic operation acceleration/ deceleration percentage
: Speed data 1

Goal position 1

Goal position 2

Goal position 3

Goal position 4

Goal position 5

Control data
Positioning data, number of transmission words

Zero return

Automatic

Manual

Stop

Reception

Program Description

1. Special internal output R7E3 (one scan on) is used to set the TRNS3 and RECV3 parameters.

2. When the communication start condition R100 is set, the communication execution flag M0 is turned on.

3. Communication processing (transmission) is executed for the one axis positioning module installed in slot 3 of the basic unit.

4. When communication start condition R101 is set, the communication execution flag M20 is turned on.

5. Communication processing (reception) is executed for the one axis positioning module installed in slot 3 of the basic unit.

6. When the normal end (TRNS3) flag M1 is set, normal end processing is executed and the communication start condition R100 is cleared.

7. When the abnormal end (TRNS3) flag M2 is set, abnormal end processing is executed and the communication start condition R100 is cleared.

8. When the normal end (RECV3) flag M21 is set, normal end processing is executed and the communication start condition R101 is cleared.

9. When the abnormal end (RECV3) flag M22 is set, abnormal end processing is executed and the communication start condition R101 is cleared.

10. When the common parameter setting condition R0 is set, the common parameters, transmission data address, data length, and control data are set, and the communication start condition R100 is turned on.

11. When the positioning data setting condition R1 is set, the positioning data, transmission data address, data length, and control data are set, and the communication start condition R100 is turned on.

12. When zero return condition R2 is set, the control data is set, and the communication start condition R100 is turned on.

13. When the automatic operation condition R3 is set, the control data is set, and the communication start condition R100 is turned on.

14. When the manual operation condition R4 is set, the control data is set, and the communication start condition R100 is turned on.

15. When the stop condition R5 is set, the control data is set, and the communication start condition R100 is turned on.

16. When the reception condition R6 is set, the reception area address and data length are set, and the communication start condition R101 is turned on.

Name

Ladder format	Condition code					Processing time (μs)				Remarks
QTRNS3 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
QTRNS3 (d,s,t)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			5		2620	5200	3314	6550	

Usable I/O		Bit				Word				Double word			Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o								
s	Top of parameter area							o							In regard to "s," up to s + 13 is used.
t	Top of communications control bit area			o											In regard to "t," up to t + 4 is used.

QTRNS 3

Function

1. This command is used between the one axis positioning module (POSIT-H) and the CPU. This command is used to send the command and data to the CPU, then receive the response data.

2. The smallest WY number of the I/O assignment WY for the installation location of POSIT-H which communicates with the QTRNS3 command is set in d. (Designate WY***4.)

3. The top I/O number of the parameter area which sets the various parameters (time out value, scan time out value, transmission area top I/O number and size, control data) for communications is set in s.

4. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

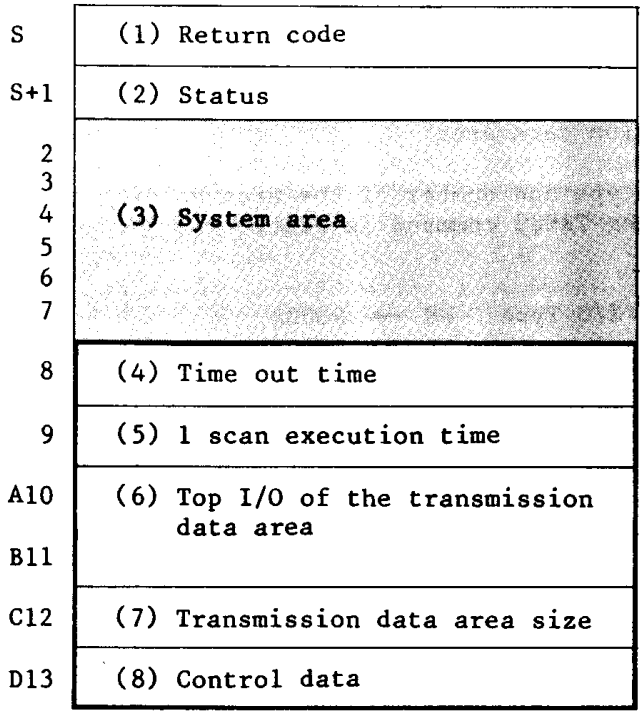
5. The QTRNS3 command will communicate data for the time designated by the parameters for one scan.

Precautions

- Array variables cannot be used in d, s and t.

- Use s + 13 and t + 4 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area



: User access prohibited area

: User setting area

(1) Return code:

The execution results of the TRNS3 command are set.
Normal end → 0
Abnormal end → ≠ 0 (See TRNSO for details.)

(2) Status:

The contents of the status words (WX***0) of POSIT-H are set.

(3) System area:

When the TRNS3 command is executed, this area is used for system processing. This area cannot be used by the user.

(4) Time out time:

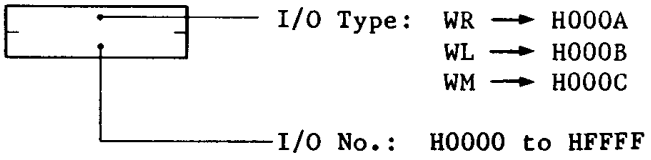
This sets the time out time from the top of the TRNS3 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(5) 1 scan execution time:

Sets the time executed during one scan.
= 0: Invalid (Executed in the same manner as TRNS3.)
≠ 0: For a *ms period, executes QTRNS processing.

(6) Top I/O of the transmission data area:

This designates top I/O type and number of the area holding the transmission data when the TRNS3 command is sent.

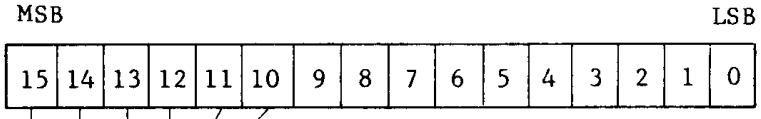


(7) Transmission data area size:

This designates the size of the transmission data area in word units.

(8) Control data:

This is set according to the table on the next depending on the contents to be communicated.

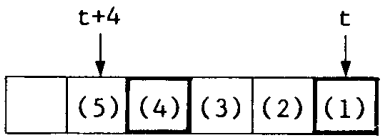


0: OFF 1: ON

Bit	Contents/definitions
bit 0 to 5	Writes the number of words to be sent. The maximum is 17 words. When only a control register is to be written, set 0.
bit 6, 7	Not used. Set this to 0.
bit 8, 9	Designates the operation mode of the positioning module.
bit 10	Operation bit. Start operation = 1. Pause = 0. The module will operate according to the leading edge and trailing edge of this bit. Once pause is executed the module will pause and will not accept operation input until it enters the standby status.
bit 11	Synchronous/asynchronous mode. Asynchronous = 0. Synchronous = 1. This determines the mode for synchronous and asynchronous operation through the MS and SL signals. Be sure to set the asynchronous mode when the MS and SL signals are not used.
bit 12	Master/slave setting. Slave axis = 0. Master axis = 1. This sets whether the positioning module is for the master axis or slave axis. If the asynchronous mode is used, however, this setting does not have to be specified.
bit 13	Communications data mode. Positioning data = 0. Common parameters = 1. Sets the type of communications data.
bit 14	Not used. Set this to 0.
bit 15	Stop. Normal = 0. Stop = 1. When this bit is set to 1, the module will be stopped. Pulse row output is stopped.

Bit		Mode
9	8	
0	0	Wait
0	1	Manual
1	0	Automatic
1	1	Zero return

6. Description of the t area



(☐ : Bits set by the user.)

(1) Communication execution:

When communication is executed with the QTRNS3 command, the user program should set "1." When communication is ended, the QTRNS3 command should be reset to "0."

(2) Normal end:

When communications with the QTRNS3 command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the QTRNS3 command ends in an error or there is an error in a parameter, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request: When the QTRNS3 command is set to the initial state, this is set to "1." (This is not initialization of the high function module.) When initialization ends normally, this is reset to "0."

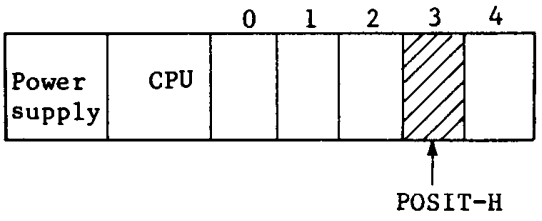
(5) Initialization end:

When initialization of the QTRNS3 command ends normally, this bit is set to "1." At this time, the initialization request (4) will be reset to "0."

Programming Example

This is a programming example in which a one axis positioning module (POSIT-H) is used, and goes from initialization to operation.

1. POSIT-H installation



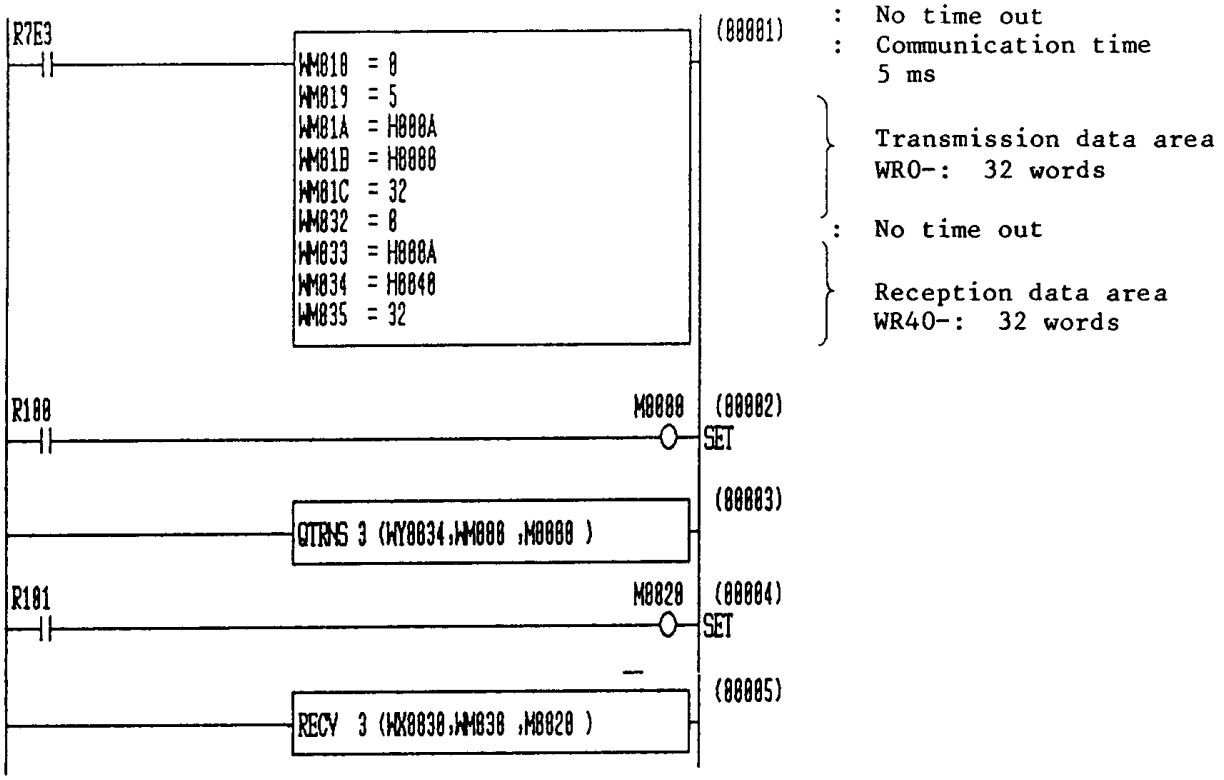
POSIT-H is installed in slot three of the basic unit. Therefore, the I/O assignments for POSIT-H are WX30 to WX33 and WY34 to WY37. WY34 is designated for the d parameter of QTRNS3, and WX30 is designated for the d parameter of RECV3.

(2) Internal output assignment

The programming example will use the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
WM	010 to 01D	QTRNS3 command parameter area (s to s+13)	
M	0000 to 0004	QTRNS3 command communication control bit area (t to t+4)	
WR	0000 to 0020	QTRNS3 command transmission data area (32 words)	
WM	030 to 035	RECV3 command parameter area (s to s+5)	
M	0020 to 0024	RECV3 command communication control bit area (t to t+4)	
WR	0040 t0 0050	RECV3 command reception data area (32 words)	
R	00000	Common parameter transmission	
	00001	Positioning data transmission	
	00002	Zero return request flag	
	00003	Automatic operation request flag	
	00004	Manual operation request flag	
	00005	Stop request flag	
	00006	Receive request flag	
	00100	QTRNS3 activation flag	
	00101	RECV3 activation flag	

Programming Example



The remainder of the program is the same as that for TRNS3.

Program Description

- Refer to the program description for TRNS3.

Name

Ladder format	Condition code					Processing time (μs)				Remarks
TRNS4 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps			H-2002		H-702/302				
TRNS4 (d,s,t)	Conditions		Step	Average		Maximum		Average	Maximum	
	-		5	109		176		207	334	

Usable I/O		Bit				Word				Double word				Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o									
s	Top of parameter area							o								In regard to "s," up to s + 16 is used.
t	Top of communications control bit area			o												In regard to "t," up to t + 5 is used.

TRNS 4

Function

1. This command is used between the two axis positioning module (POSIT-2H, POSITA-2H) and the CPU. This command is used to send the command and data to the CPU, then receive the response data.

2. The smallest WY number of the I/O assignment WY for the installation location of POSIT-2H and POSITA-2H which communicates with this command is set in d. (Designate WY***4.)

3. The top I/O number of the parameter area which sets the various parameters (time out value, top I/O number and size of the transmission and reception data area, control data) for communications is set in s.

4. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

Precautions

- Array variables cannot be used in d, s and t.

- Use s + 16 and t + 5 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area

S	(1) Return code
S+1	(2) Status
2	(3) Status
	(4) System area (Not usable by the user.)
9	(5) Time out time
10	(6) Top I/O of the transmission area
11	
12	(7) Transmission area size
13	(8) Top I/O of the reception area
14	
15	(9) Reception area size
16	(10) Control data

(: User access prohibited area
: User setting area)

(1) Return code:

The execution results of the TRNS4 command are set.
Normal end → 0
Abnormal end → ≠ 0 (See TRNS0 for details.)

(2) and (3) Status:

The contents of the status words (WX***0) of POSIT-2H, POSITA-2H are set.

(4) System area:

When the TRNS4 command is executed, this area is used for system processing. This area cannot be used by the user.

(5) Time out time:

This sets the time out time from the top of the TRNS4 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(6) Top I/O of the transmission data area:

This designates top I/O type and number of the area holding the transmission data when the TRNS4 command is sent. (Note)

(7) Transmission data area size:

This designates the size of the transmission data area in word units.

(8) Top I/O of the reception area:

This designates top I/O type and number of the area holding the transmission data when the TRNS4 command is sent. (Note)

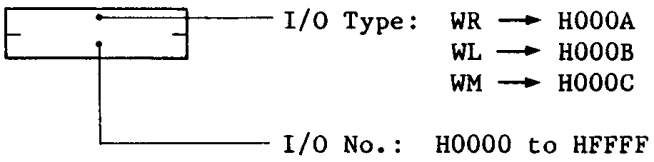
(9) Reception area size:

This designates the size of the reception data area in word units.

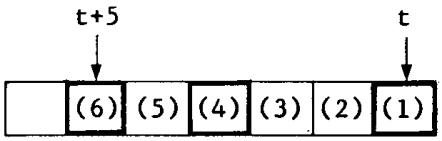
(10) Control data:

This sets the command for the two axis positioning module. Refer to the two axis positioning module manual or instruction manual for details on the command.

(Note)



6. Description of the t area



(□ : Bits set by the user.)

(1) Communication execution:

When communication is executed with the TRNS4 command, the user program should set "1." When communication is ended, the TRNS4 command should be reset to "0."

(2) Normal end:

When communications with the TRNS4 command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the TRNS4 command ends in an error or there is an error in a parameter, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request:

When the TRNS4 command is set to the initial state, this is set to "1." (This is not initialization of the high function module.)

(5) Initialization end:

When initialization of the TRNS4 command ends normally, this bit is set to "1." At this time, the initialization request (4) will be reset to "0."

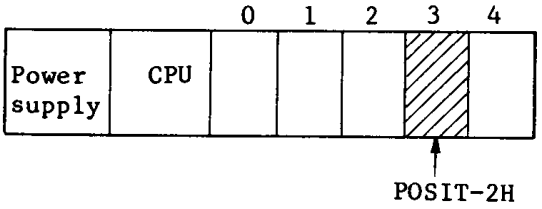
(6) Transmission/reception request:

When reception is to be executed after transmission with this command, set 1 in the user program. Reset should also be executed by the user program.

Programming Example

This is a programming example in which a two axes positioning module (POSIT-2H) is used, and goes from initialization to operation.

(1) POSIT-2H installation



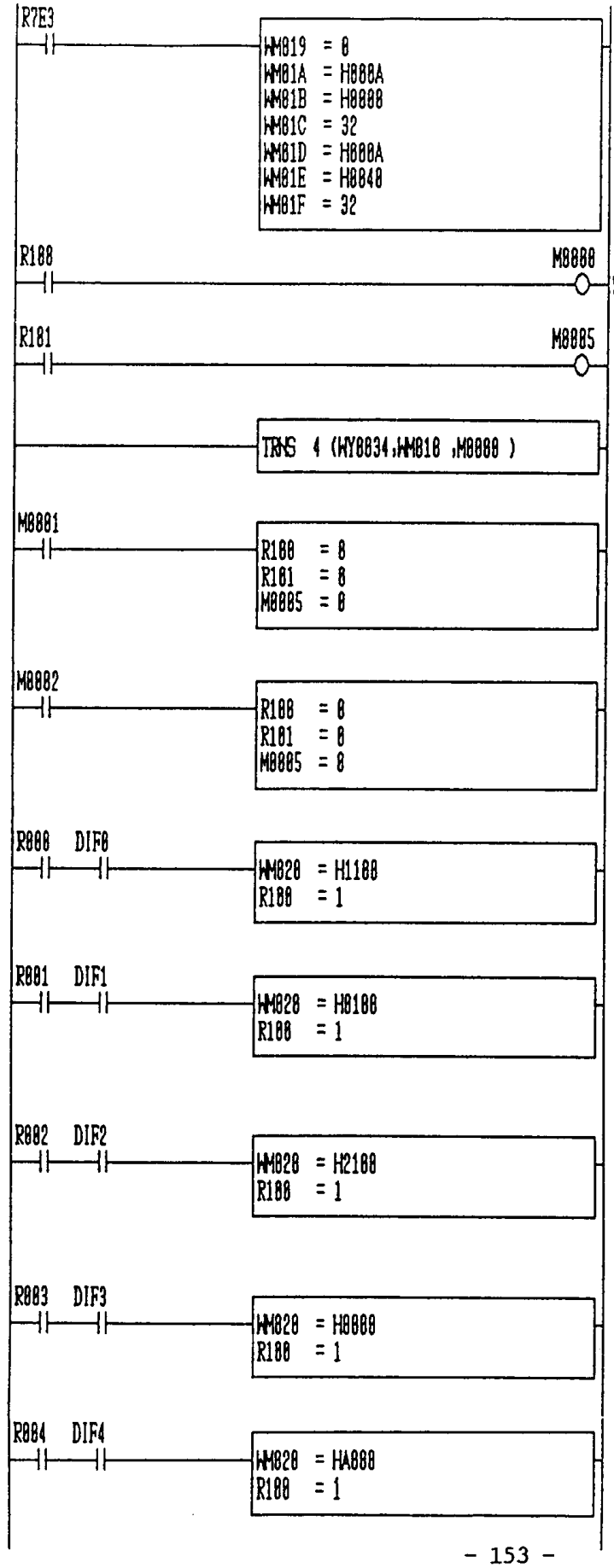
POSIT-2H is installed in slot three of the basic unit. Therefore, the I/O assignments for POSIT-2H are WX30 to WX33 and WY34 to WY37. WY34 is designated for the d parameter of TRNS4.

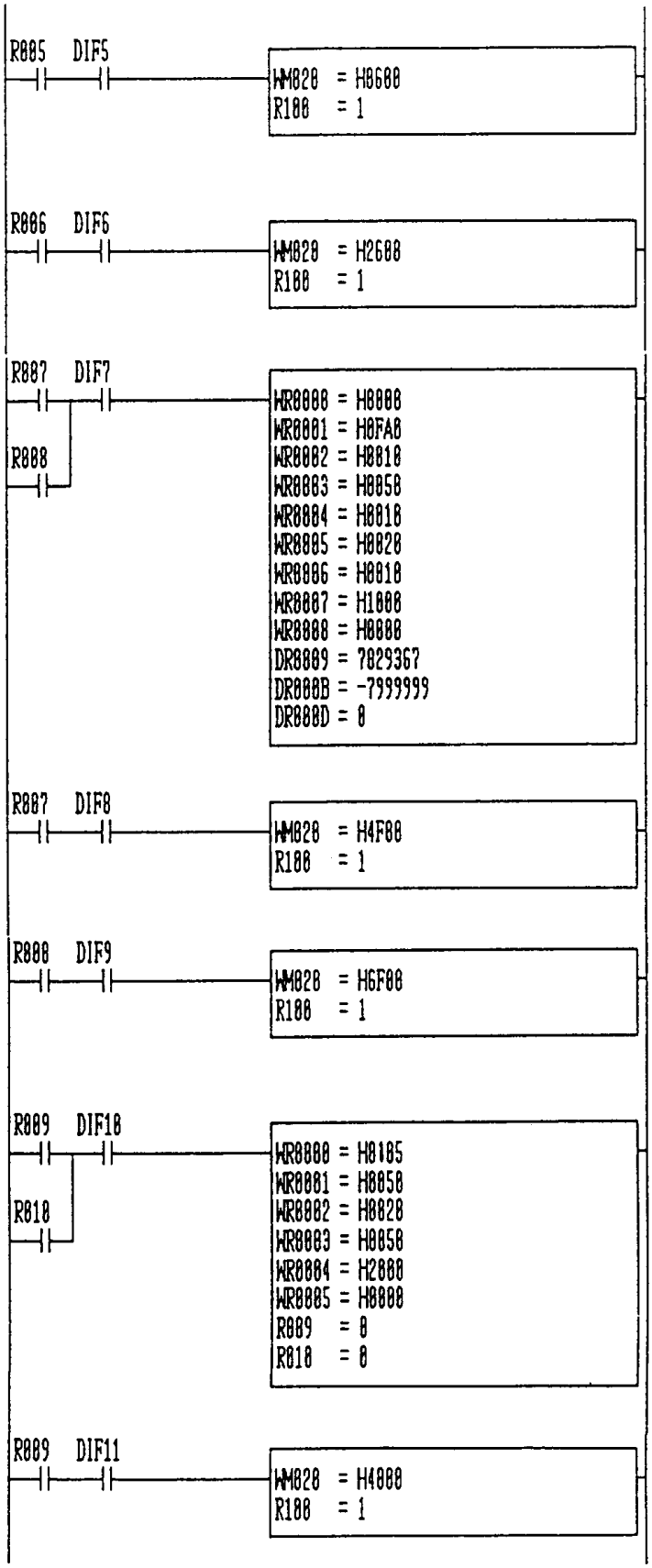
(2) Internal output assignment

The programming example will use the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
WM	010 to 01A	TRNS4 command parameter area (s to s+10)	
M	0000 to 0005	TRNS4 command communication control bit area (t to t+5)	
WR	0000 to 001F	TRNS4 command transmission data area (32 words)	
WR	0040 to 003F	TRNS4 command reception data area (32 words)	
R	00000	Communications initialization request	
	00001	X axis zero return request	
	00002	Y axis zero return request	
	00003	X axis stop request	
	00004	Y axis stop request	
	00005	X axis automatic operation 3	
	00006	Y axis automatic operation 3	
	00007	X axis common parameter setting	
	00008	Y axis common parameter setting	
	00009	X axis positioning data setting	
	00010	Y axis positioning data setting	
	00011	X axis current position data request	
	00012	Y axis current position data request	
	00100	TRNS4 activation flag	
	00101	TRNS4 reception flag	

Programming Example





(00012) : X axis automatic Control data H0600

(00013) : Y axis automatic Control data H2600

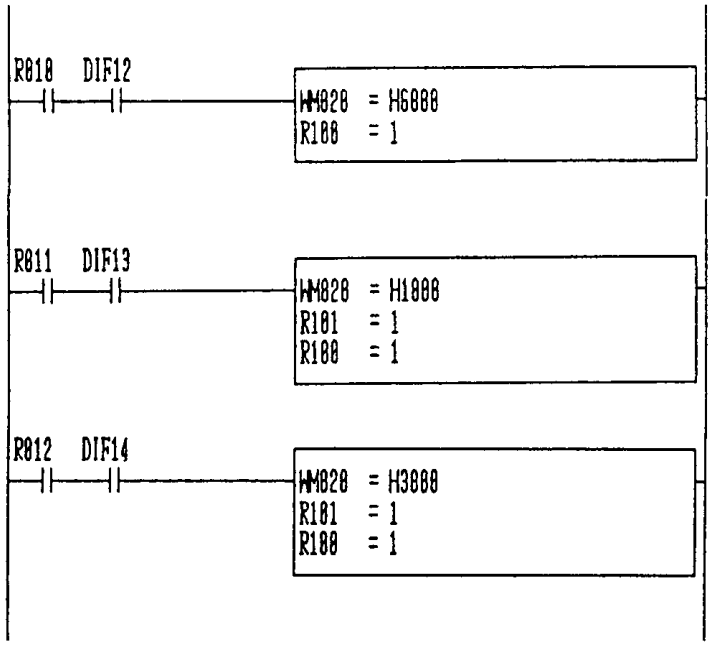
(00014) : X, Y axis Common parameter data

(00015) : X axis common parameter setting Control data H4F00

(00016) : Y axis common parameter setting Control data H6F00

(00017) : X, Y axis Positioning data

(00018) : X axis position data setting Control data H4000



: Y axis position data setting
Control data H6000

: X axis current position

: Y axis current position

. Current position data

WR41	WR40
Upper	Lower

Programming Example

1. Special internal output R7E3 (one scan on) is used to set the TRNS4 parameters.

2. When the communication start condition R100 is set, the communication execution flag M0 is turned on.

3. When the reception condition R101 is set, the reception flag used after communication M5 is turned on.

4. Communications processing is executed for the two axes positioning module installed in slot 3 of the basic unit.

5. When the normal end flag M1 is set, normal end processing is executed, and the communication start condition R100, reception condition R101, and the reception flag used after communication M5 are cleared.

6. When the abnormal end flag M2 is set, normal end processing is executed, and the communication start condition R100, reception condition R101, and the reception flag used after communication M5 are cleared.

7. When the communication initialization condition R0 is set, the control data is set, and the communication start condition R100 is turned on.

8. When the X axis zero return condition R1 is set, the control data is set, and the communication start condition R100 is turned on.

9. When the Y axis zero return condition R2 is set, the control data is set, and the communication start condition R100 is turned on.

10. When the X axis stop condition R3 is set, the control data is set, and the communication start condition R100 is turned on.

11. When the Y axis stop condition R4 is set, the control data is set, and the communication start condition R100 is turned on.

12. When the X axis automatic 3 condition R5 is set, the control data is set, and the communication start condition R100 is turned on.

13. When the Y axis automatic 3 condition R6 is set, the control data is set, and the communication start condition R100 is turned on.

14. When the common parameter setting condition R7 or R8 is set, the common parameters, transmission area address and length are set.

15. When the X axis common parameter setting condition R7 is set, the control data is set, and the communication start condition R100 is turned on.

16. When the Y axis common parameter setting condition R8 is set, the control data is set, and the communication start condition R100 is turned on.

17. When the positioning data setting condition R9 or R10 is set, the positioning data, transmission area address, and length are set.

18. When the X axis positioning setting condition R9 is set, the control data is set, and the communication start condition R100 is turned on.

19. When the Y axis positioning setting condition R10 is set, the control data is set, and the communication start condition R100 is turned on.

20. When the X axis current value read condition R11 is set, the control data is set, and after communication with the communication start condition R100, the reception condition R101 is turned on.

21. When the Y axis current value read condition R12 is set, the control data is set, and after communication with the communication start condition R100, the reception condition R101 is turned on.

Name

Ladder format	Condition code					Processing time (μs)				Remarks
QTRNS4 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
QTRNS4 (d,s,t)	Conditions		Step			Average	Maximum	Average	Maximum	
	-		5			516	2990	1791	3500	

	Usable I/O	Bit				Word				Double word				Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o									
s	Top of parameter area							o								In regard to "s," up to s + 19 is used.
t	Top of communications control bit area			o												In regard to "t," up to t + 5 is used.

QTRNS 4

Function

1. This command is used between the two axis positioning module (POSIT-H, POSITA-2H) and the CPU. This command is used to send the command and data to the CPU, then receive the response data.

2. The smallest WY number of the I/O assignment WY for the installation location of POSIT-H and POSITA-2H which communicates with this command is set in d. (Designate WY***4.)

3. The top I/O number of the parameter area which sets the various parameters (time out value, top I/O number and size of the transmission and reception data area, control data) for communications is set in s.

4. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

5. The QTRNS4 command will send data for the period designated by the parameters during one scan.

Precautions

. Array variables cannot be used in d, s and t.

. Use s + 19 and t + 5 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area

S	(1) Return code
S+1	(2) Status
2	(3) Status
	(4) System area (Not usable by the user.)
11	(5) Time out time
12	(6) Maximum execution time
13	(7) Top I/O of the transmission area
14	
15	(8) Transmission area size
16	(9) Top I/O of the reception area
17	
18	(10) Reception area size
19	(11) Control data

(: User access prohibited area
: User setting area)

(1) Return code

The execution results of QTRNS4 command are set.
Normal end → 0
Abnormal end → ≠ 0 (See TRNS0 for details.)

(2) and (3) Status:

The contents of the status words (WX***0) of the various high function modules are set.

(4) System area:

When the QTRNS4 command is executed, this area is used for QTRNS4 system processing. This area cannot be used by the user.

(5) Time out time:

This sets the time out time from the top of the QTRNS4 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed.

(6) Maximum execution time:

This designates the time of execution during one scan.
= 0: Invalid (Executes in the normal manner.)
≠ 0: For *ms, executes the process of this command.

(7) Top I/O of the transmission data area:

This designates top I/O type and number of the area holding the transmission data when the QTRNS4 command is sent. (Note)

(8) Transmission data area size:

This designates the size of the transmission data area in word units.

(9) Top I/O of the reception area:

This designates top I/O type and number of the area holding the transmission data when the QTRNS4 command is sent. (Note)

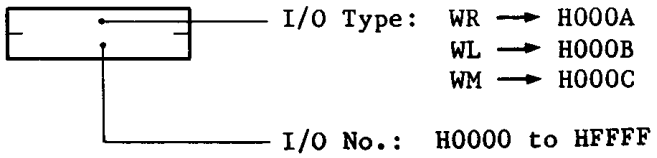
(10) Reception area size:

This designates the size of the reception data area in word units.

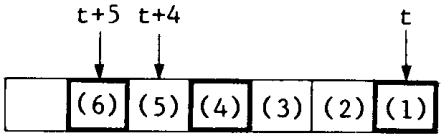
(11) Control data:

Specify according to the contents to be communicated. Refer to TRNS4.

(Note)



6. Description of the t area



(□ : Bits set by the user.)

(1) Communication execution:

When communication is executed with the QTRNS4 command, the user program should set "1." When communication is ended, the QTRNS4 command should be reset to "0."

(2) Normal end:

When communications with the QTRNS4 command ends normally, "1" is set. When communication is started, this bit is reset to "0."

(3) Abnormal end:

When communications with the QTRNS4 command ends in an error or there is an error in a parameter, this is set to "1." When communication is started, this bit is reset to "0."

(4) Initialization request:

When the QTRNS4 command is set to the initial state, this is set to "1." (This is not initialization of the high function module.)

(5) Initialization end:

When initialization of the QTRNS4 command ends normally, this bit is set to "1." At this time, the initialization request (4) will be reset to "0."

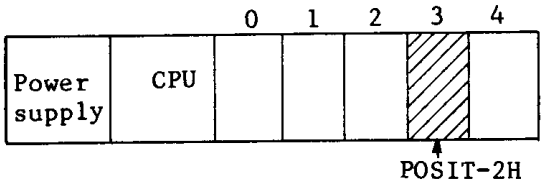
(6) Transmission/reception request:

When reception is to be executed after transmission with this command, set 1 in the user program. Reset should also be executed by the user program.

Programming Example

This is a programming example in which a two axes positioning module (POSIT-2H) is used, and goes from initialization to operation.

(1) POSIT-2H installation

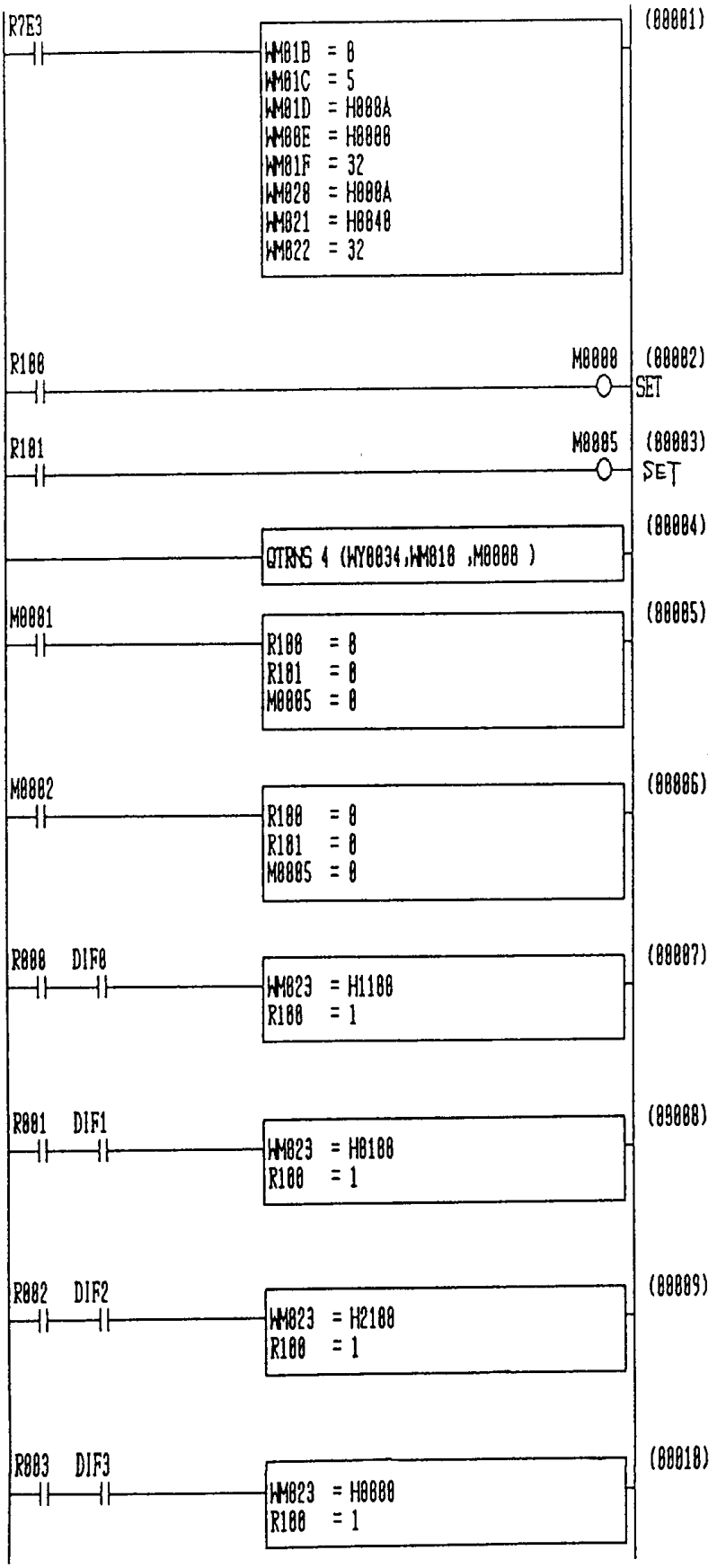


POSIT-2H is installed in slot three of the basic unit. Therefore, the I/O assignments for POSIT-2H are WX30 to WX33 and WY34 to WY37.
WY34 is designated for the d parameter of QTRNS4.

(2) Internal output assignment

The programming example will use the following assignments. In actual applications, be sure to modify the I/O numbers, etc.

I/O	No.	Purpose	Remarks
WM	010 to 023	QTRNS4 command parameter area (s to s+19)	
M	0000 to 0005	QTRNS4 command communication control bit area (t to t+5)	
WR	0000 to 001F	QTRNS4 command transmission data area (32 words)	
WR	0040 to 003F	QTRNS4 command reception data area (32 words)	
R	00000	Communications initialization request	
	00001	X axis zero return request	
	00002	Y axis zero return request	
	00003	X axis stop request	
	00004	Y axis stop request	
	00005	X axis automatic operation 3	
	00006	Y axis automatic operation 3	
	00007	X axis common parameter setting	
	00008	Y axis common parameter setting	
	00009	X axis positioning data setting	
	00010	Y axis positioning data setting	
	00011	X axis current position data request	
	00012	Y axis current position data request	
	00100	QTRNS4 activation flag	
	00101	QTRNS4 reception flag	



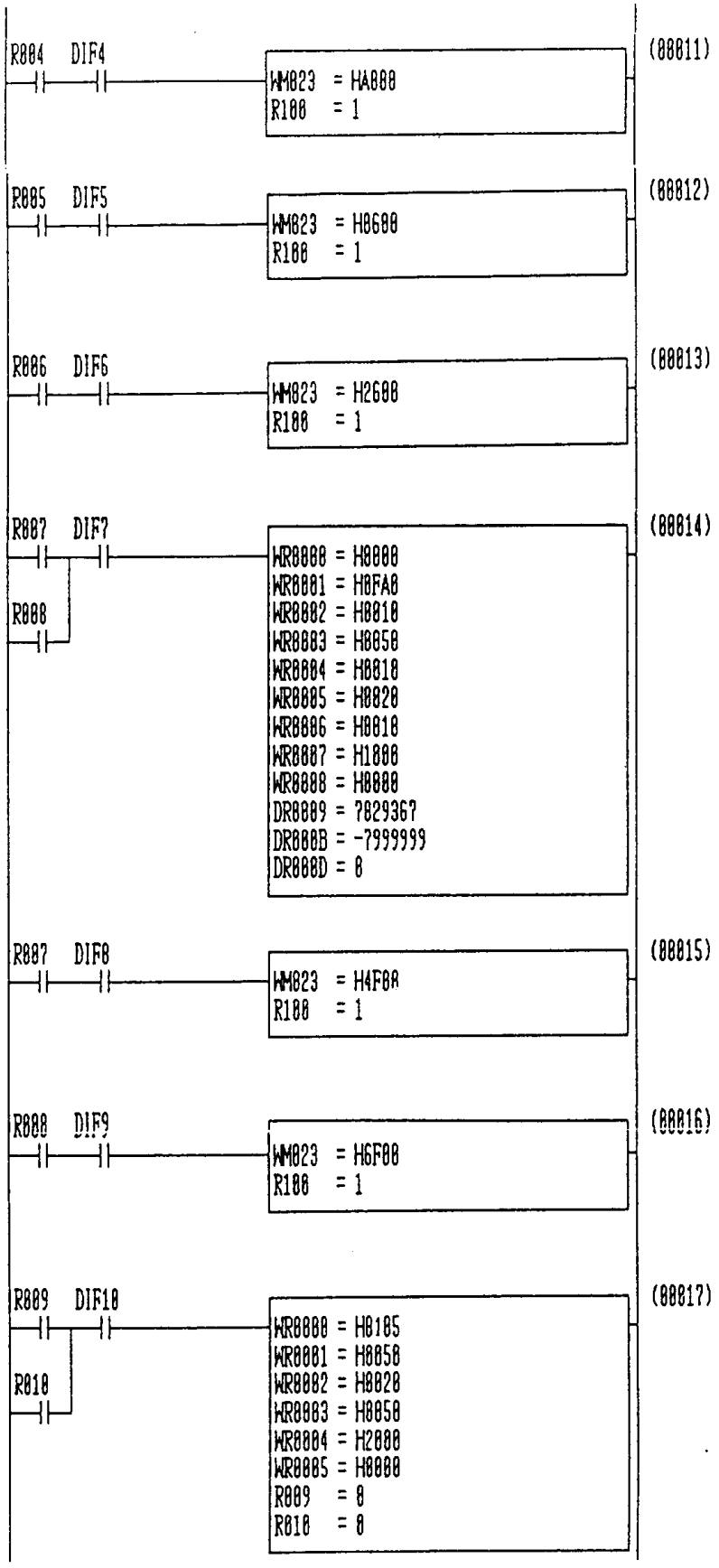
QTRNS4 parameter setting
No time out
Execution time 5 ms

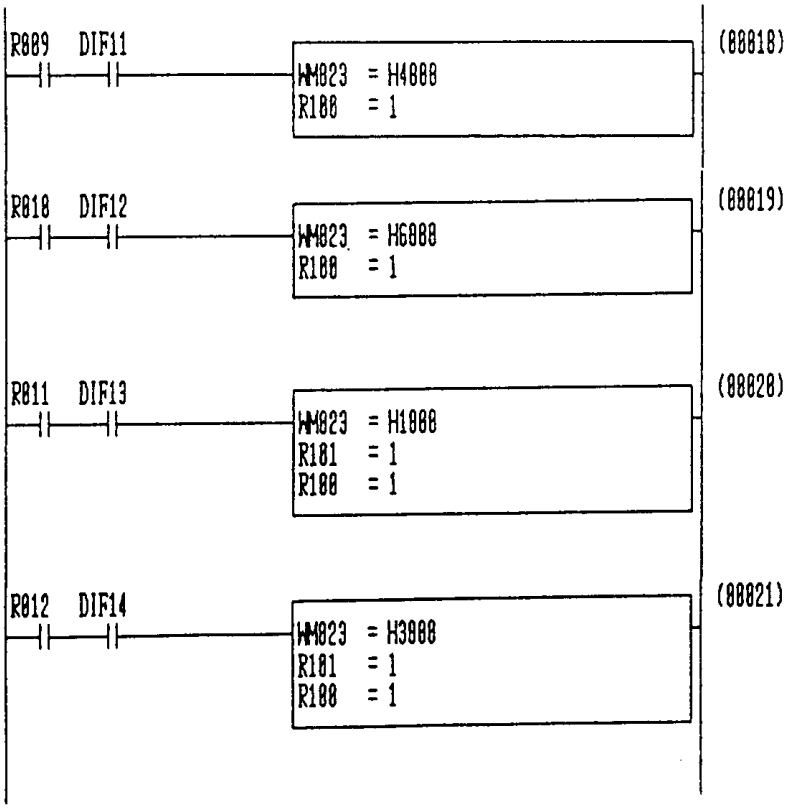
Transmission data area WR0-: 32 words

Reception data area WR40-: 32 words

The rest is the same as TRNS4.

Only the word which sets the control data is changed.


R004 DIF4
WM023 = HA000
R100 = 1
(00011)
R005 DIF5
WM023 = H0600
R100 = 1
(00012)
R006 DIF6
WM023 = H2600
R100 = 1
(00013)
R007 DIF7
R008
WR0000 = H0000
WR0001 = H0FA0
WR0002 = H0010
WR0003 = H0050
WR0004 = H0010
WR0005 = H0020
WR0006 = H0010
WR0007 = H1000
WR0008 = H0000
DR0009 = 7829367
DR000B = -7999999
DR000D = 0
(00014)
R007 DIF8
WM023 = H4F00
R100 = 1
(00015)
R008 DIF9
WM023 = H6F00
R100 = 1
(00016)
R009 DIF10
R010
WR0000 = H0105
WR0001 = H0050
WR0002 = H0020
WR0003 = H0050
WR0004 = H2000
WR0005 = H0000
R009 = 0
R010 = 0
(00017)


R009 DIF11
HM823 = H4000
R100 = 1
(00018)
R010 DIF12
HM823 = H6000
R100 = 1
(00019)
R011 DIF13
HM823 = H1000
R101 = 1
R100 = 1
(00020)
R012 DIF14
HM823 = H3000
R101 = 1
R100 = 1
(00021)

Name 1 quantity counter module communications command

Ladder format	Condition code					Processing time (μs)				Remarks
TRNS5 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
TRNS5 (d,s,t)	Conditions		Step			Average	Maximum	Average	Maximum	
	-		5							

Usable I/O		Bit				Word				Double word			Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o								
s	Top of parameter area							o							In regard to "s," up to s + 9 is used.
t	Top of communications control bit area			o											In regard to "t," up to t + 2 is used.

TRNS 5

Function

1. This is the communication command used between the one quantity counter module (XCU001H) and the CPU. This command sets the presetting for the count elapsed value, elapsed value reading, first setting, and second setting for XCU001H.

2. The smallest WY number of the I/O assignment WY for the installation location of the counter module which communicates with this command is set in d. (Designate WY***4.)

3. The top I/O number of the parameter area which sets the various parameters (presetting value for the control content count elapsed value, first setting, second setting) for communications is set in s.

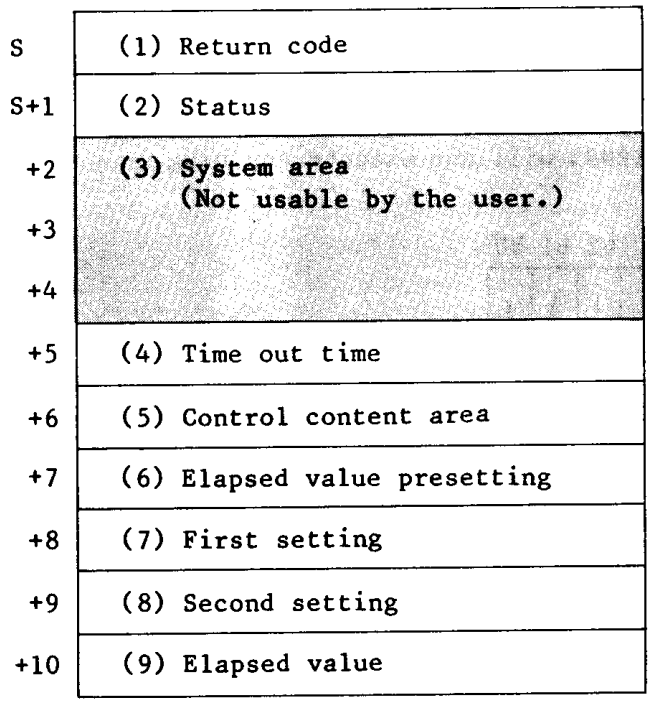
4. The control bit which controls the start of communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

Precautions

- Array variables cannot be used in d, s and t.

- Use s + 9 and t + 2 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area



(: User access prohibited area
: User setting area)

(1) Return code:

The execution results of the TRNS5 command are set.
Normal end → 0
Abnormal end → ≠ 0 (See TRNS0 for details.)

(2) Status:

The contents of the status word (WX***0) of the XCU001H are set.

(3) System area:

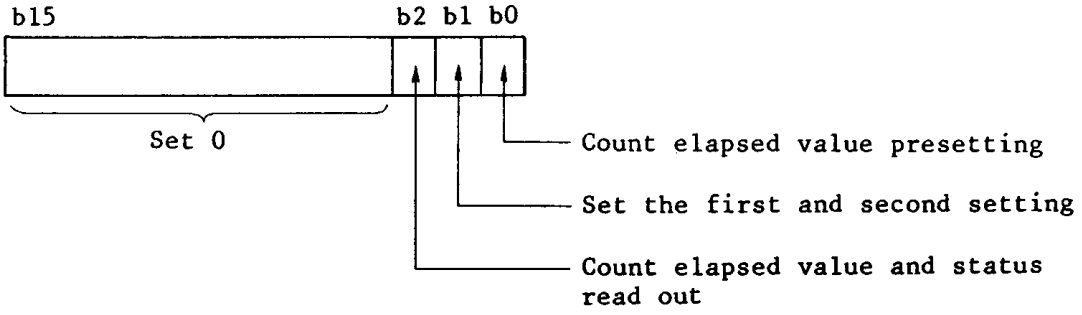
When the TRNS5 command is executed, this area is used for TRNS5 system processing. This area cannot be used by the user.

(4) Time out time:

This sets the time out time from the top of the TRNS5 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed. (Note 1)

(5) Control contents:

This designates the communications process executed by the TRNS command for the counter module. Select the corresponding bit, and set "1."
"1" can be set simultaneously for the following depending on the control contents.
(1 to 7: If 0 is set, TRNS5 will not execute anything.)



(6) Elapsed value presetting:

This sets the presetting in the counter module elapsed value.

(7) First setting:

This sets the first setting in the counter module.

(8) Second setting:

This sets the second setting in the counter module.

(9) Elapsed value:

The current elapsed value data (WX***1) of the counter module is set.

Precautions

Note 1:

. Set the time out time as follows.

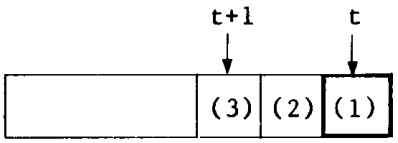
(1) When the one quantity counter module is installed in the basic or expansion unit:

Set a value larger than the user program scan time.

(2) When the one quantity counter module is installed in a remote unit:

T = max (user program scanning time, 30 msec), so that a value larger than T should be set.

6. Description of the t area



(☐ : Bits set by user.)

(1) Communication execution:

When communication is started with the TRNS5 command, "1" is set by the user program.
After communications, this bit will be reset to "0" by the TRNS5 command.

(2) Normal end:

When the communications by the TRNS5 command ends normally, "1" is set.

When communications are started, this bit will be reset to "0" by the TRNS5 command.

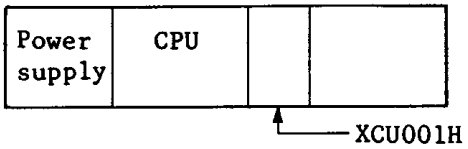
(3) Abnormal end:

When the communications by the TRNS5 command ends abnormally, or there is an error in the parameters, "1" is set.
When communications are started, this bit will be reset to "0" by the TRNS5 command.

Program Description

The following is an example of a program which uses the counter 1 quantity module (XCU001H), and sets the counter elapsed value presetting, and the first and second setting registers.

(1) XCU001H installation



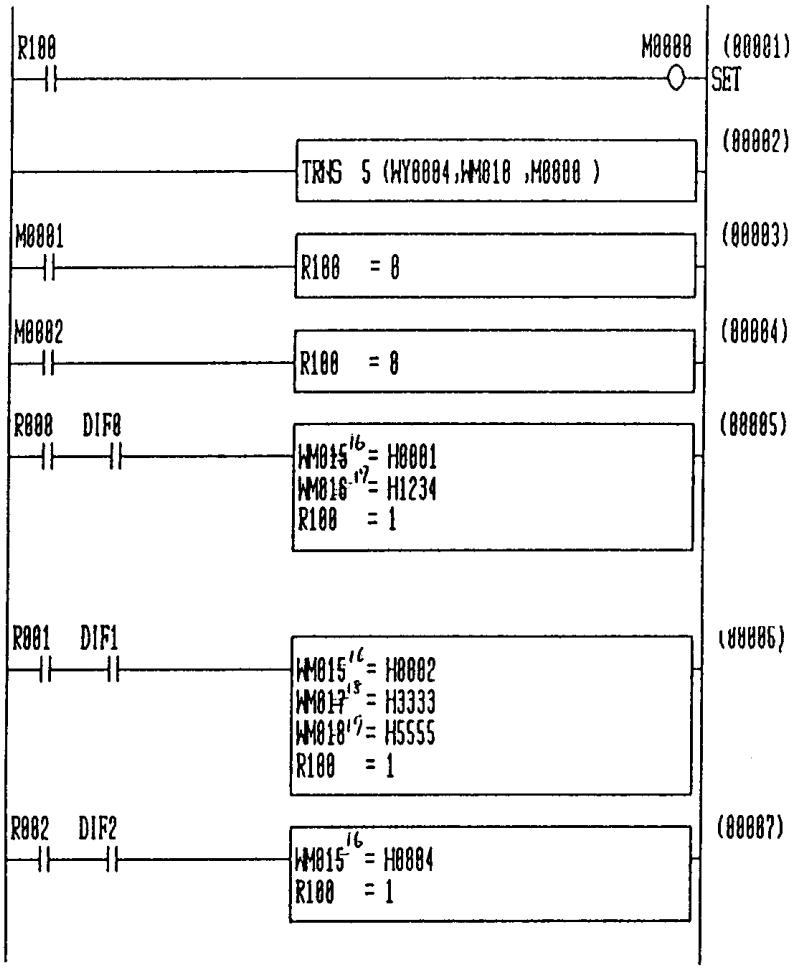
XCU001H is installed in slot 0 of the basic unit. XCU001H is assigned to WX0 to 3 and WY4 to 7.
The d parameter of the TRNS5 command should designate WY4.

(2) Internal output assignment

The programming example will use the following assignments. In actual applications, be sure to modify the I/O numbers, etc. according to the user program.

I/O	No.	Purpose	Remarks
WM	10 to 1A	Parameter area	
M	0 to 2	Communication control bit	
R	0	Counter presetting flag	
	1	Setting request flag for settings	
	2	Counter elapsed value read request flag	
	100	Communication start flag	

Programming Example



Normal end processing
Request flag R100 clear

Error processing
Request flag R100 clear

: Control content bit = Presetting
: Presetting value

: Control content bit = Set setting
: First setting
: Second setting

: Control content bit = Read out

Program Description

Name Two quantity counter module communications command

Ladder format	Condition code					Processing time (μs)				Remarks
TRNS6 (d,s,t)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		
	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					H-2002		H-702/302		
TRNS6 (d,s,t)	Conditions			Step		Average	Maximum	Average	Maximum	
	-			5		450	840	599	1080	

Usable I/O		Bit				Word				Double word			Constant	Array	Other
		X	Y	R,L,M	TD,SS,WDT, MS,TMR,CU, CUR,CT	WX	WY	WR,WL WM	TC	DX	DY	DR,DL DM			
d	Module installation location						o								
s	Top of parameter area							o							In regard to "s," up to s + 25 is used.
t	Top of communications control bit area			o											In regard to "t," up to t + 2 is used.

TRNS 6

Function

1. This command is used between the two quantity counter module (XCU232H) and the CPU. This command is used to send the command and data to the CPU, then receive the response data.

2. The smallest WY number of the I/O assignment WY for the installation location of XCU232H which communicates with the TRNS6 command is set in d. (Designate WY***4.)

3. The top I/O number of the parameter area which sets the various parameters (time out value, top I/O number and size of the transmission and reception data area, etc.) for communications is set in s.

4. The control bit which starts and initializes communications, and top I/O No. of the communication control bit area in which the results of communications are set in t.

Precautions

. Array variables cannot be used in d, s and t.

. Use s + 25 and t + 2 so that they do not exceed the I/O range (R7FF, L03FFF, L13FFF, M3FFF, WRC3FF, WL3FF, WL13FF, WM3FF). If they do exceed the range, then DER = 1 will be true, and communications will not be executed.

5. Description of the s area

S	(1) Return code
S+1	(2) Status
	(3) System area
S+8	(4) Time out time
S+9	(5) Control content area
S+10	(6) 1 quantity current value
S+12	(7) 1 quantity current value setting
S+14	(8) 1 quantity comparative value
S+16	(9) 1 quantity comparative value setting
S+18	(10) 2 quantity current value
S+20	(11) 2 quantity current value setting
S+22	(12) 2 quantity comparative value
S+24	(13) 2 quantity comparative value setting

(: User access prohibited area
: User setting area)

(1) Return code:

The execution results of the TRNS6 command are set.
Normal end → 0
Abnormal end → ≠ 0 (See TRNS0 for details.)

(2) Status:

The contents of the status word (WX***0) of the XCU232H are set.

(3) System area:

When the TRNS6 command is executed, this area is used for TRNS6 system processing. This area cannot be used by the user.

(4) Time out time:

This sets the time out time from the top of the TRNS6 command execution to its completion.
= 0: No time out checking.
≠ 0: *10 ms time out check executed. (Note 1)

(5) Control contents:

This designates the communications process executed by the TRNS command for the counter module. Select the corresponding bit, and set "1."

11	10	9	8					3	2	1	0				
0 0 0 0	2 quantity				0 0 0 0	1 quantity									
	Comparative value		Current value			Comparative value		Current value							
	Setting	Read out	Setting	Read out		Setting	Read out	Setting	Read out						

"1" can be set simultaneously for these depending on the control contents.
If H0000 is set, TRNS6 cannot execute anything.

(6) 1 quantity current value read out area (s+4: Lower, s+5: Upper)

(7) 1 quantity setting data storage area (lower then upper)

(8) 1 quantity comparative value read out area (lower then upper)

(9) 1 quantity comparative value setting data storage area (lower then upper)

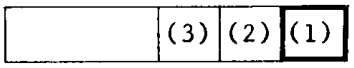
(10) 2 quantity current value read out area (lower then upper)

(11) 2 quantity current value setting data storage area (lower then upper)

(12) 2 quantity comparative value read out area (upper then lower)

(13) 2 quantity comparative value setting data storage area (lower then upper)

6. Description of the t area



(☐ : Bits set by the user.)

(1) Communication execution:

When communication is started with the TRNS6 command, "1" is set by the user program. After communications, this bit will be reset to "0" by the TRNS6 command.

(2) Normal end:

When the communications by the TRNS6 command ends normally, "1" is set.
When communications are started, this bit will be reset to "0" by the TRNS6 command.

(3) Abnormal end:

When the communications by the TRNS6 command ends abnormally, or there is an error in the parameters, "1" is set.
When communications are started, this bit will be reset to "0" by the TRNS6 command.

Precautions

Note 1:

. Set the time out time as follows.

(1) When the two quantity counter module is installed in the basic or expansion unit: Set a value larger than User Program Scanning Time x Control Contents x 2.

Example: Scanning time = 30 msec

Control contents = 3 (1 quantity current value read out and setting, 2 quantity current value setting)

In the above example, a value larger than 30 x 3 x 2 = 180 msec must be set.

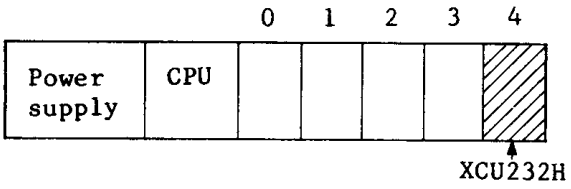
(2) When the one quantity counter module is installed in a remote unit:

T = max (user program scanning time, 30 msec), so that a value larger than T x Control Contents x 2 must be set.

Programming Example

The following is a programming example in which the 2 quantity counter module (XCU232H) is used from initialization to operation.

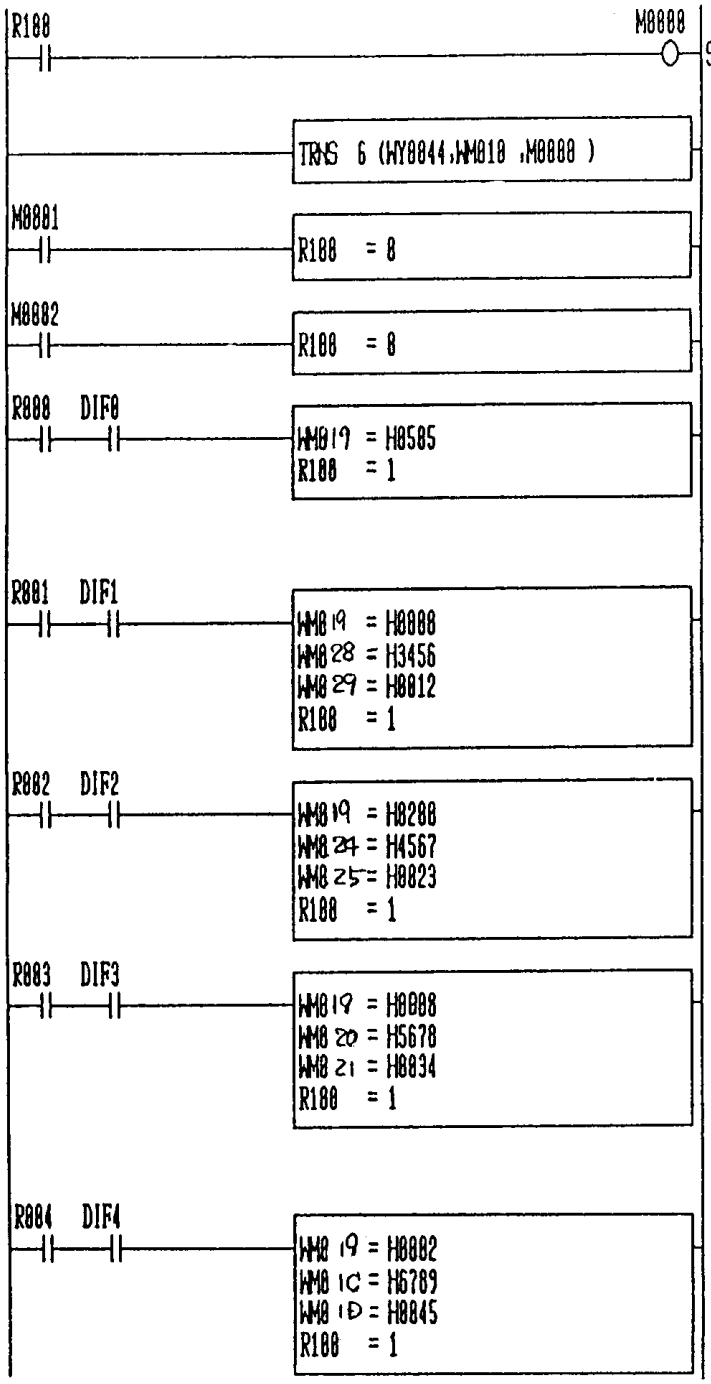
1. XCU232H installation



XCU232H is installed in slot 4 of the basic unit. XCU232H is assigned to WX40 to 43 and WY44 to 47.
The d parameter of the TRNS6 command should designate WY44.

I/O	No.	Type	Remarks
WM	10 to 25	Parameter area	
M	0 to 2	Communication control bit	
R	D	All read out flag	
	1	2 quantity comparative value setting flag	
	2	2 quantity current value setting flag	
	3	1 quantity comparative value setting flag	
	4	1 quantity current value setting flag	
	100	Communication start flag	

Programming Example



: Normal end processing
Request flag R100 clear

: Error processing
Request flag R100 clear

: Control content
All read out

: Control content:
2 quantity comparative value setting

: Control content:
2 quantity current value setting

: Control content:
1 quantity comparative value setting

: Control content:
1 quantity current value setting

Program Description

1. When communication start condition R100 is set, the communication execution flag M0 is turned on.

2. Communications processing is executed for the 2 quantity counter module.

3. When the normal end flag M1 is set, normal end processing is executed and communication start condition R100 is cleared.

4. When the abnormal end flag M2 is set, abnormal end processing is executed and communication start condition R100 is cleared.

5. When all the all read out condition for 1 quantity and 2 quantity R0 is set, the control contents are set, and the communication start condition R100 is turned on.

6. When the 2 quantity comparative value setting condition R1 is set, the control contents are set, and the communication start condition R100 is turned on.

7. When the 2 quantity current value setting condition R2 is set, the control contents are set, and the communication start condition R100 is turned on.

8. When the 1 quantity comparative value setting condition R3 is set, the control contents are set, and the communication start condition R100 is turned on.

9. When the 1 quantity current value setting condition R4 is set, the control contents are set, and the communication start condition R100 is turned on.

[4] FUN Command

[Name] SIN function

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 10 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (SIN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·					
Command format	No. of steps					25	27	50	54	
FUN 10 (S)	Conditions			Step						
* (SIN (S))	-			3						

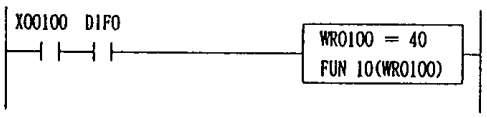
Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument							o						S to S+2 are used.

[Function]

- The SIN value having a sign specified by S or a binary value as an argument is calculated, and the integer part is set in S+2 and the fraction part is set in S+1.
- The SIN value is expressed by a binary value and a negative number is expressed by twos complement.
- When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]




```
LD X00100
AND DIF0
 [
WR0100=40
FUN10 (WR0100)
 ]
```

[Program explanation]

- An angle of 40° is set in WR0100.
- The SIN operation is performed at the leading edge of X00100, and the fraction part of the result is set in WR0101 in a binary value and the integer part is set in WR0102.

  Execution result: WR0102 = H0000, WR0101 = HA48E, WR0100 = H0028

[Precautions]

- The argument S is given by an angle (degree) and ranges from 0 to 360. When S is beyond the limits, DER = 1 and no operation is performed.
- When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no operation is performed.

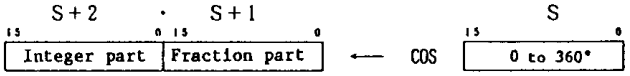
FUN 10 (SIN)

[Name] COS function

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 11 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (COS (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					35	37	52	56	
FUN 11 (S)	Conditions			Step						
* (COS (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument							o						S to S+2 are used.

[Function]



. The COS value having a sign specified by S or a binary value as an argument is calculated, and the integer part is set in S+2 and the fraction part is set in S+1.

. The COS value is expressed by a binary value and a negative number is expressed by twos complement.

. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]


```
LD X00101
AND DIF1
 [
WR0110=110
FUN11  (WR0110)
 ]
```

[Program explanation]

. An angle of 110° is set in WR0110.

. The COS operation is performed at the leading edge of X00101, and the fraction part of the result is set in WR0111 in a binary value and the integer part is set in WR0112.

Execution result: WR0112 = HFFFF, WR0111 = HA871, WR0110 = H006E

[Precautions]

. The argument S is given by an angle (degree) and ranges from 0 to 360. When S is beyond the limits, DER = 1 and no operation is performed.

. When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no operation is performed.

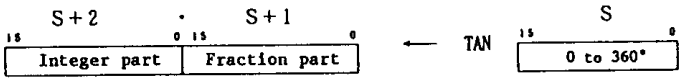
FUN 11 (COS)

[Name] TAN function

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 12 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (TAN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	36	38	53	55	
Command format	No. of steps									
FUN 12 (S)	Conditions			Step						
* (TAN (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument							o						S to S+2 are used.

[Function]



. The TAN value having a sign specified by S or a binary value as an argument is calculated, and the integer part is set in S+2 and the fraction part is set in S+1.

. The TAN value is expressed by a binary value and a negative number is expressed by twos complement.

. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]


```
LD X00102
AND DIF2
 [
WR0105=45
FUN 12 (WR0105)
 ]
```

[Program explanation]

. An angle of 45° is set in WR0105.

. The TAN operation is performed at the leading edge of X00102, and the fraction part of the result is set in WR0106 in a binary value and the integer part is set in WR0107.

Execution result: WR0107 = H0001, WR0106 = H0000, WR0105 = H002D

[Precautions]

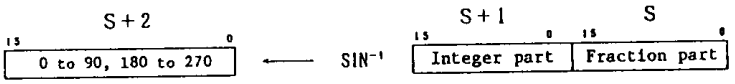
. The argument S is given by an angle (degree) and ranges from 0 to 360. When S = 90 or 270, H7FFF or HFFFF is set in S+2 or S+1. When S is beyond the limits, DER = 1 and no operation is performed. When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no operation is performed.

[Name] ARC SIN function

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 13 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (ASIN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	112	114	216	220	
Command format	No. of steps									
FUN 13 (S)	Conditions			Step						
* (ASIN (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (fraction)							o						S to S+2 are used.
S+1	Argument (integer)							o						

[Function]



. The SIN^{-1} value having a binary value specified by S (fraction part) and S+1 (integer part) as an argument is calculated and outputted to S+2.

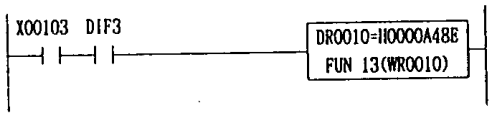
. The SIN^{-1} value is an angle (degree) ranging from 0 to 90 or from 180 to 270 and expressed by a binary value.

. When operations are performed normally, DER = 0.

. The data in the fraction part is the real value x 65535.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]




```
LD X00103
AND DIF3
 [
DR0010=H0000A48E
FUN 13 (WR0010)
 ]
```

[Program explanation]

. Data is set in DR0010 (WR0010, WR0011).

. The SIN^{-1} operation is performed at the leading edge of X00103, and the result is set in WR0012 in a binary value.

Execution result: WR0012 = H0028, WR0011 = H0000, WR0010 = HA48E

[Precautions]

. When | S+1.S | > 1, DER = 1 and no operation is performed.

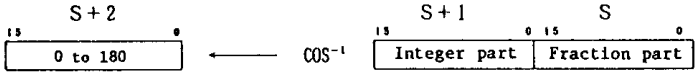
. When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no operation is performed.

[Name] ARC COS function

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 14 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (ACOS (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	114	115	218	221	
Command format	No. of steps									
FUN 14 (S)	Conditions			Step						
* (ACOS (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (fraction)							o						S to S+2 are used.
S+1	Argument (integer)							o						

[Function]



. The COS^{-1} value having a binary value specified by S (fraction part) and S+1 (integer part) as an argument is calculated and outputted to S+2.

. The COS^{-1} value is an angle (degree) ranging from 0 to 180 and expressed by a binary value.

. When operations are performed normally, DER = 0.

. The data in the fraction part is the real value x 65535.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]


```
LD X00104
AND D1F4
 [
DR0024=HFFFFA871
FUN 14 (WR0024)
 ]
```

[Program explanation]

. Data is set in DR0024 (WR0024, WR0025).

. The COS^{-1} operation is performed at the leading edge of X00104, and the result is set in WR0026 in a binary value.

  Execution result: WR0026 = H006E, WR0025 = HFFFF, WR0024 = HA871

[Precautions]

. When | S+1.S | > 1, DER = 1 and no operation is performed.

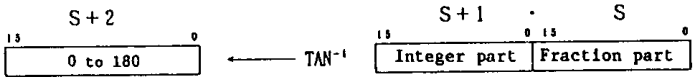
. When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no operation is performed.

[Name] ARC TAN function

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 15 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (ATAN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	123	125	237	241	
Command format	No. of steps									
FUN 15 (S)	Conditions			Step						
* (ATAN (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (fraction)							o						S to S+2 are used.
S+1	Argument (integer)							o						

[Function]



. The TAN^{-1} value having a binary value specified by S (fraction part) and S+1 (integer part) as an argument is calculated and outputted to S+2.

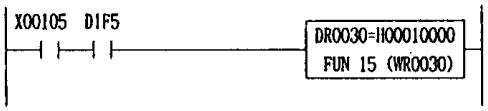
. The TAN^{-1} value is an angle (degree) ranging from 0 to 180 and expressed by a binary value.

. When operations are performed normally, DER = 0.

. The data in the fraction part is the real value x 65535.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]




```
LD X00105
AND DIF5
 [
DR0030=H00010000
FUN 15 (WR0030)
 ]
```

[Program explanation]

. Data is set in DR0030 (WR0030, WR0031).

. The TAN^{-1} operation is performed at the leading edge of X00105, and the result is set in WR0032 in a binary value.

Execution result: WR0032 = H002D, WR0031 = H0001, WR0030 = H0000

[Precautions]

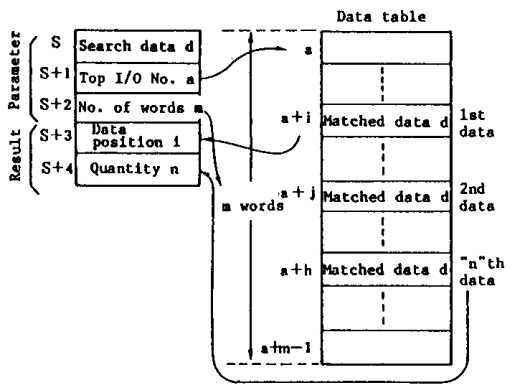
. When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no operation is performed.

[Name] Data search (Data search function)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 20 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		Two matched data are found in the 10-word table.
* (DSRCH (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·					
Command format	No. of steps					104		270		
FUN 20 (S)	Conditions			Step						
* (DSRCH (S))	-			3						

Usable I/O		Bit				Word				Double word				Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Search data							o							
S+1	Top I/O No. of data table							o							The real address is set.
S+2	No. of search words							o							S to S+4 are used.

[Function]



. The search data specified by the word internal output is searched for within the word number range, specified by S+2, of the top I/O (the real address is specified) specified by S+1, and the data position (relative position from the top I/O), which is found first, is outputted to S+3 and the data quantity, which is found, is outputted to S+4.

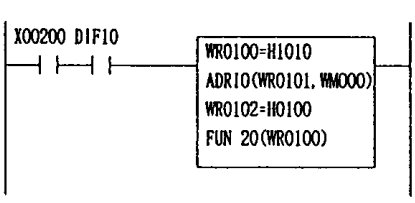
. When no matched data is found, 0 is set in S+3.

. Set the real address of the search top I/O in S+1 using the ADRIO instruction. ADRIO (S+1, top I/O)

. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]




```
LD X00200
AND DIF10
 [
WR0100=H1010
ADRIO (WR0101, WM000)
WR0102=H0100
FUN20 (WR0100)
 ]
```

Data H1010 is searched for among 256 words of WM000. After searching, the data position is set in WR0103 and the data quantity is set in WR0104.

[Precautions]

. Set the real address of WR, WL, or WM as a parameter of S+1 using the ADRIO instruction. When other addresses are set, DER = 1 and no processing is performed.

. When S to S+4 or the areas specified by them are overlapped, DER = 1 and no processing is performed.

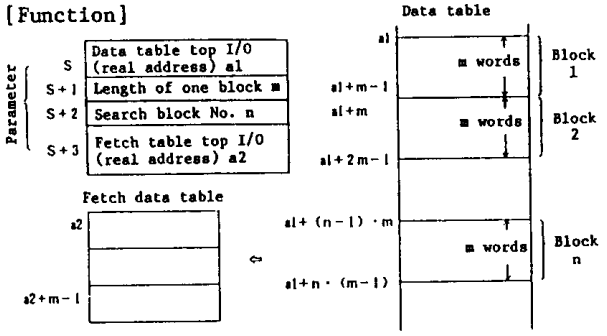
. When S+1 and S+2 or the areas specified by them are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] Table search (Table search function)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 21 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		The 10th block (one block is 2 words in length) is searched for.
* (TSRCH (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	235		476		
Command format	No. of steps									
FUN 21 (S)	Conditions			Step						
* (TSRCH (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Data table top I/O No.							o						The real address is set.
S+1	Length of one block							o						S to S+3 are used.
S+2	Search block No.							o						
S+3	Fetch table top I/O No.							o						The real address is set.

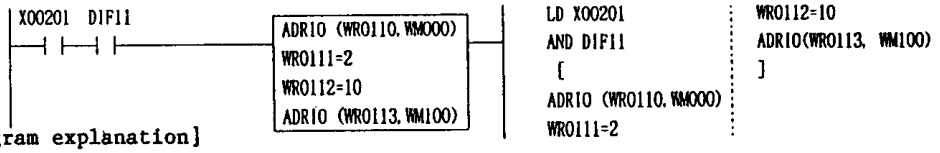
[Function]



- The data block whose number is specified by S+2 is fetched from the data table, which is defined by the word internal outputs S and S+1, and stored from the I/O specified by S+3.
- Set the real addresses of top I/O in S+1 and S+3 using the ADRIO instruction. ADRIO (S+1, top I/O)
- When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]



[Program explanation]

The 10th data block is fetched from the data table (one block consists of 2 words) of WM000 and set in the area of WM100.

[Precautions]

- Set the real addresses of WR, WL, or WM as parameters of S+1 and S+3 using the ADRIO instruction. When other addresses are set, DER = 1 and no processing is performed.
- When S to S+3 or the areas specified by them are overlapped, DER = 1 and no processing is performed.
- When S to S+3 or the areas specified by them are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] 16-bit unsigned binary -> Decimal ASCII conversion (Binary to decimal ASCII)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 30 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (BINDA (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·					
Command format	No. of steps					63	82	130	170	
FUN 30 (S)	Conditions			Step						
* (BINDA (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (conversion data)							o						S to S+3 are used.

[Function]

16-bit unsigned binary data

S [0 to 65535] ⇨

10^n: ASCII code in the 10^n place

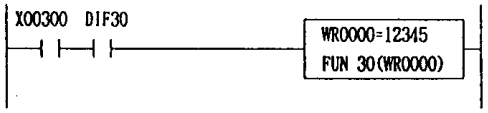
Decimal ASCII data

	15 ... 8	7 ... 0
S+1	10^4	10^3
S+2	10^2	10^1
S+3	10^0	NULL

* The contents of ( ) are indications when the ladder editor is used.

- The 16-bit unsigned binary data specified by the argument S is converted to a 5-digit decimal ASCII code and set in S+1 to S+3.
- The conversion result is suppressed to 0 and the digits are H20 (space).
- The remaining digits after ASCII conversion are null, indicating the end of character string.
- When operations are performed normally, DER = 0.

[Program example]




```
LD X00300
AND DIF30
 [
WR0000=12345
FUN 30 (WR0000)
 ]
```

[Program explanation]

- Binary data of 12345 which is set in WR0000 is converted to an ASCII. The conversion result is set in WR0001 to WR0003.

  Execution result: WR0000 = 12345 (H3039), WR0001 = H3132, WR0002 = H3334, WR0003 = H3500

[Precautions]

- When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] 32-bit signed binary -> Decimal ASCII conversion (Double binary to decimal ASCII)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 31 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (DBINDA (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	160	226	310	438	

Command format	No. of steps
FUN 31 (S)	Conditions / Step
* (DBINDA (S))	- / 3

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (low order)							o						-2147483648 to 2147483647
S+1	Argument (high order)							o						

[Function]

32-bit signed binary data

S: Low-order 16 bits
S+1: High-order 16 bits

⇨

	15	8 7	0
S+2	Sign		10^9
S+3	10^8		10^7
S+4	10^6		10^5
S+5	10^4		10^3
S+6	10^2		10^1
S+7	10^0		NULL

Sign: Positive: H20 (space)
Negative: H2D (-)

10^n: ASCII code in the 10^n place

. The 32-bit signed binary data specified by S (low order) and S+1 (high order) is converted to a 10-digit decimal ASCII code and set in S+2 to S+7.

. When the sign is positive, H20 (space) is provided. When the sign is negative, H2D (-) is provided.

. The conversion result is suppressed to 0 and the digits are H20 (space).

. The remaining digits after ASCII conversion are null, indicating the end of character string.

. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]


```
LD X00301
AND DIF31
 [
DR0010=-1234567
FUN 31 (WR0010)
 ]
```

[Program explanation]

. Binary data of -1234567 which is set in DR0010 (WR0010, WR0011) is converted to an ASCII.

. The conversion result is set in WR0012 to WR0017.

Execution result: DR0010 = 1234567 (HFFED2979), WR0012 = H2D20, WR0013 = H2020, WR0014 = H3132, WR0015 = H3334, WR0016 = H3536, WR0017 = H3700

[Precautions]

. When S+1 to S+7 are beyond the maximum I/O number, DER = 1 and no processing is performed.

FUN 31 (DBINDA)

[Name] 16-bit binary -> Hexadecimal ASCII conversion (Binary to hexa ASCII)

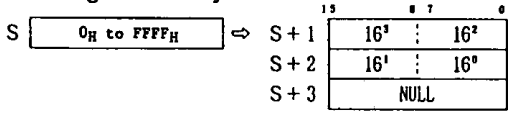
Ladder format	Condition code					Processing time (μs)				Remarks
FUN 32 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (BINHA (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•					
Command format	No. of steps					48	64	97	131	
FUN 32 (S)	Conditions			Step						
* (BINHA (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument							o						S to S+3 are used.

[Function]

Hexadecimal ASCII data

16-bit unsigned binary data



16^n: ASCII code in the 16^n place

- The 16-bit unsigned binary data specified by the argument S is converted to a 4-digit hexadecimal ASCII code and set in S+1 to S+3.
- The conversion result is not suppressed to 0.
- Null behind the ASCII data means the end of character string.
- When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]


```
LD X00302
AND DIF32
 [
WR0020=H1234
FUN 32 (WR0020)
 ]
```

[Program explanation]

- Binary data of H1234 which is set in WR0020 is converted to an ASCII.
- The conversion result is set in WR0021 to WR0023.

  Execution result: WR0020 = H1234, WR0021 = H3132, WR0022 = H3334, WR0023 = H0000

[Precautions]

- When S+1 to S+3 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] 32-bit binary -> Hexadecimal ASCII conversion (Double binary to hexa ASCII)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 33 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (DBINHA (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	.	.	.	.	61	79	124	160	
Command format	No. of steps									
FUN 33 (S)	Conditions			Step						
* (DBINHA (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (low order)							o						H00000000 to HFFFFFFFF
S+1	Argument (high order)							o						S to S+6 are used.

[Function]

32-bit unsigned binary data → Hexadecimal ASCII data

S	Low-order 16 bits	⇨	S+2	16^7 : 16^6
S+1	Higher 16 bits		S+3	16^5 : 16^4
			S+4	16^3 : 16^2
			S+5	16^1 : 16^0
			S+6	NULL

16^n: ASCII code in the 16^n place

* The contents of ( ) are indications when the ladder editor is used.

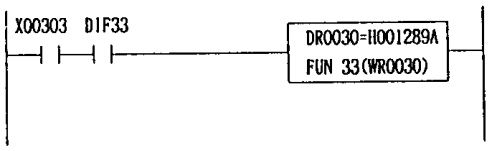
. The 32-bit binary data specified by S (low order) and S+1 (high order) is converted to a 8-digit hexadecimal ASCII code and set in S+2 to S+6.

. The conversion result is not suppressed to 0.

. Null behind the ASCII data means the end of character string.

. When operations are performed normally, DER = 0.

[Program example]




```
LD X00303
AND D1F33
 [
DR0030=H001289AB
FUN 33 (WR0030)
 ]
```

[Program explanation]

. Binary data of H001289AB which is set in DR0030 is converted to an ASCII.

. The conversion result is set in WR0032 to WR0036.

Execution result: DR0030 = H001289AB, WR0032 = H3030, WR0033 = H3132, WR0034 = H3839, WR0035 = H4142, WR0036 = H0000

[Precautions]

. When S+1 to S+6 are beyond the maximum I/O number, DER = 1 and no processing is performed.

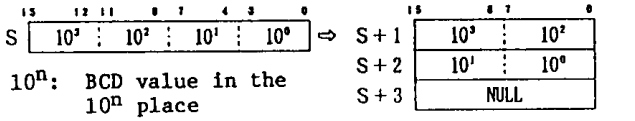
FUN 33 (DBINHA)

[Name] 16-bit BCD -> Decimal ASCII conversion (BCD to decimal ASCII)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 34 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-700/302		
* (BCDDA (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	•	•	•	•	54	71	110	145	
Command format	No. of steps									
FUN 34 (S)	Conditions			Step						
* (BCDDA (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (BCD)							o						S to S+3 are used.

[Function]



10^n: BCD value in the 10^n place

10^m: ASCII code in the 10^m place

* The contents of ( ) are indications when the ladder editor is used.

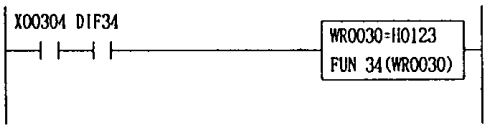
. The 16-bit BCD data specified by the argument S is converted to a 4-digit decimal ASCII code and set in S+1 to S+3.

. The conversion result is suppressed to 0 and the digits are H20 (space).

. Null behind the ASCII data means the end of character string.

. When operations are performed normally, DER = 0.

[Program example]




```
LD X00304
AND DIF34
 [
WR0030=H0123
FUN 34 (WR0030)
 ]
```

[Program explanation]

. BCD data of H0123 which is set in WR0030 is converted to an ASCII.

. The conversion result is set in WR0031 to WR0033.

Execution result: WR0030 = H0123, WR0031 = H2031, WR0032 = H3233, WR0033 = H0000

[Precautions]

. When S is other than BCD data, DER = 1 and no processing is performed.

. When S+1 to S+3 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] 32-bit BCD -> Decimal ASCII conversion (Double BCD to decimal ASCII)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 35 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (DBCDDA (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	72	92	149	190	

Command format	No. of steps
FUN 35 (S)	Conditions / Step
* (DBCDDA (S))	- / 3

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (low order)							o						S indicates BCD data.
S+1	Argument (high order)							o						S to S+6 are used.

[Function]

32-bit BCD data

	15 12 11 8 7 4 3 0			
S	10^3	10^2	10^1	10^0
S+1	10^7	10^6	10^5	10^4

⇨

Hexadecimal ASCII data

	15 8 7 0	
S+2	10^7	10^6
S+3	10^5	10^4
S+4	10^3	10^2
S+5	10^1	10^0
S+6	NULL	

10^n: BCD value in the 10^n place

10^m: ASCII code in the 10^m place

* The contents of ( ) are indications when the ladder editor is used.

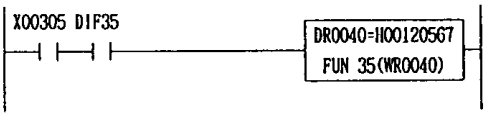
. The 32-bit BCD data specified by S (low order) and S+1 (high order) is converted to a 8-digit decimal ASCII code and set in S+2 to S+6.

. The conversion result is suppressed to 0 and the digits are H20 (space).

. Null behind the ASCII data means the end of character string.

. When operations are performed normally, DER = 0.

[Program example]




```
LD X00305
AND DIF35
 [
DR0040=H00120567
FUN 35(WR0040)
 ]
```

[Program explanation]

. BCD data of H00120567 which is set in DR0040 is converted to an ASCII.

. The conversion result is set in WR0042 to WR0046.

Execution result: DR0040 = H00120567, WR0042 = H2020, WR0043 = H3132, WR0044 = H3035, WR0045 = H3637, WR0046 = H0000

[Precautions]

. When S and S+1 are other than BCD data, DER = 1 and no processing is performed.

. When S+1 to S+6 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] Unsigned 5-digit decimal ASCII -> 16-bit binary conversion (Decimal ASCII to binary)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 36 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (DABIN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	101	123	207	254	
Command format	No. of steps									
FUN 36 (S)	Conditions			Step						
* (DABIN(S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (high order)							o						S to S+2 are combinations of H00, H20, and H30 to H39. S to S+2 are used.
S+1	Argument (middle order)							o						
S+2	Argument (low order)							o						

FUN 36 (DABIN)

[Function]

Unsigned decimal ASCII data → 16-bit binary data

S	10^4 : 10^3	⇨ S+3	0 to 65535
S+1	10^2 : 10^1		
S+2	10^0 : H00		

10^n: ASCII code in the 10^n place

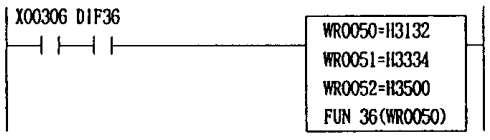
* The contents of ( ) are indications when the ladder editor is used.

. The unsigned 5-digit decimal ASCII data specified by S (high order), S+1 (middle order), and S+2 (low order) is converted to 16-bit binary data and set in S+3.

. H00 and H20 (null, space) in the high order positions are processed as H30 (0). (0-suppressed digits)

. When operations are performed normally, DER = 0.

[Program example]




```
LD X00306       :  WR0052=H3500
AND DIF36       :  FUN 36(WR0050)
 [              :  ]
  WR0050=H3132
  WR0051=H3334
```

[Program explanation]

. ASCII data 1, 2, 3, 4, and 5 which are set in WR0050 to WR0052 are converted to binary data.

. The conversion results are set in WR0053.

Execution result: WR0050 = H3132, WR0051 = H3334, WR0052 = H3500, WR0053 = 12345 (H3039)

[Precautions]

. When 5-digit ASCII codes which are set in S to S+2 are other than H30 to H39 (0 to 9), DER = 1 and no processing is performed. This is not applied to H00 and H20 (null, space) in digits which are suppressed to 0.

. When S+1 to S+3 are beyond the maximum I/O number, DER = 1 and no processing is performed.

. When data is more than 65 536, DER = 1 and no processing is performed.

[Name] Signed 10-digit decimal ASCII -> 32-bit binary conversion (Double decimal ASCII to binary)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 37 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (DDABIN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	154	193	329	413	

Command format	No. of steps	
FUN 37 (S)	Conditions	Step
* (DDABIN(S))	-	3

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (ASCII code)							o						The sign is H00 or H2D. The other digits are combinations of H00, H20, and H30 to H39. S to S+7 are used.
〃	〃							〃						
S+5	Argument (ASCII code)							o						

[Function]

Signed ASCII data

S	Sign	10^8
S+1	10^8	10^7
S+2	10^6	10^5
S+3	10^4	10^3
S+4	10^2	10^1
S+5	10^0	H00

⇨ Signed 32-bit binary data

S+6	Low-order 16 bits
S+7	High-order 16 bits

10^n: ASCII code in the 10^n place

Sign: H2C (space): Positive
H2D (-): Negative

. The signed 10-digit decimal ASCII data specified by the arguments S to S+5 is converted to 32-bit binary data and set in S+7 (high order) and S+6 (low order).

. The argument is a combination of H00, H20, H30 to H39, and H2D (-).

. H00 and H20 (null, space) in the high order positions are processed as H30 (0). (0-suppressed digits)

. When operations are performed normally, DER = 0.

. Signed data ranges from -2147483648 to 214783647.

* The contents of ( ) are indications when the ladder editor is used.

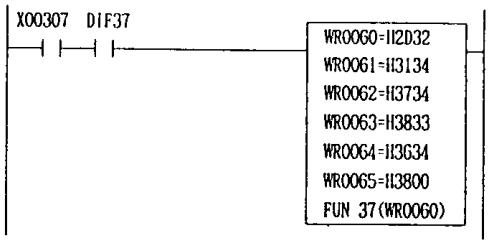
[Precautions]

. In cases except for that the sign is H20 or H2D and the other digits are H30 to H39 (0 to 9), DER = 1 and no processing is performed. This is not applied to H00 and H20 (null, space) in digits which are suppressed to 0.

. When data is other than -2147483648 to 214783647, DER = 1 and no processing is performed.

. When S+1 to S+7 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Program example]




```
LD X00307
AND DIF37
 [
WR0060=H2D32
WR0061=H3134
WR0062=H3734
WR0063=H3833
WR0064=H3634
WR0065=H3800
FUN 37(WR0060)
]
```

[Program explanation]

- ASCII data "-", 2, 1, 4, 7, 4, 8, 3, 6, 4, and 8 which are set in WR0060 to WR0065 are converted to binary data.
- The conversion results are set in WR0067 (high order) and WR0066 (low order).

  Execution result: WR0060 = H2D32, WR0061 = H3134, WR0062 = H3734, WR0063 = H3833, WR0064 = H3634, WR0065 = H3800, DR0066 = -2147483648 (H80000000)

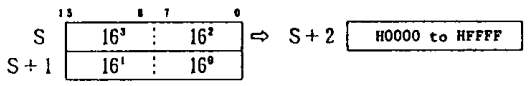
[Name] 4-digit hexadecimal ASCII -> 16-bit binary conversion (Hexa ASCII to binary)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 38 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (HABIN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	81	102	168	214	
Command format	No. of steps									
FUN 38 (S)	Conditions			Step						
* (HABIN (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (high order ASCII data)							o						Combinations of H00, H20, H30 to H39, and H41 to H46
S+5	Argument (low order ASCII data)							o						S to S+2 are used.

[Function]

Hexadecimal ASCII data 16-bit binary data



16^n: ASCII code in the 16^n place

* The contents of ( ) are indications when the ladder editor is used.

. The 4-digit hexadecimal ASCII data specified by the arguments S and S+1 is converted to binary data and outputted to S+2.

. H00 and H20 (null, space) in the high order positions are processed as H30 (0). (0-suppressed digits)

. The argument is a combination of H30 to H39 and H41 to H46 (0 to 9, A to F).

. When operations are performed normally, DER = 0.

[Program example]


```
LD X00308
AND DIF38
 [
 WR0070=H3132
 WR0071=H4142
 FUN 38(WR0070)
 ]
```

[Program explanation]

. ASCII data 1, 2, A, and B which are set in WR0070 and WR0071 are converted to binary data.

. The conversion results are set in WR0072.

Execution result: WR0070 = H3132, WR0071 = H4142, WR0072 = H12AB

[Precautions]

. When 4-digit ASCII codes which are set in S and S+1 are other than H30 to H39 and H41 to H46, DER = 1 and no processing is performed. This is not applied to H00 and H20 (null, space) in digits which are suppressed to 0.

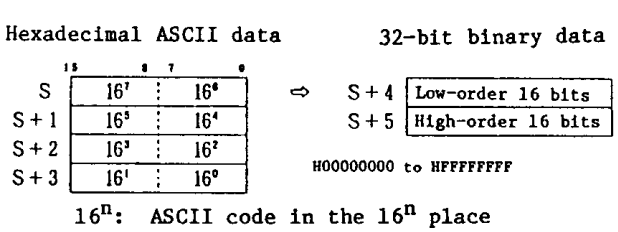
. When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] 8-digit hexadecimal ASCII -> 32-bit binary conversion (Double hexa ASCII to binary)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 39 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (DHABIN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	116	142	244	298	
Command format	No. of steps									
FUN 39 (S)	Conditions			Step						
* (DHABIN (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (ASCII data)							o						Combinations of H00, H20, H30 to H39, and H41 to H49
〃	〃							〃						
S+3	Argument (ASCII data)							o						S to S+5 are used.

[Function]

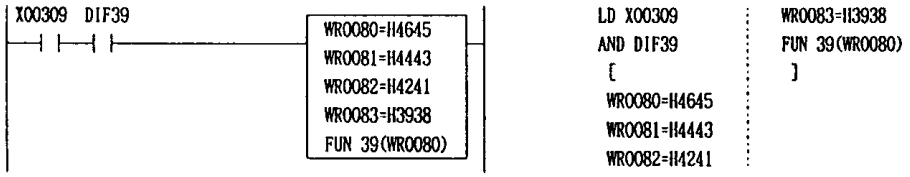


16^n: ASCII code in the 16^n place

* The contents of ( ) are indications when the ladder editor is used.

- The 8-digit hexadecimal ASCII data specified by the arguments S to S+3 is converted to binary data and outputted to S+4 and S+5.
- H00 and H20 (null, space) in the high order positions are processed as H30 (0). (0-suppressed digits)
- The argument is a combination of H30 to H39 and H41 to H46 (0 to 9, A to F).
- When operations are performed normally, DER = 0.

[Program example]




```
LD X00309        :  WR0083=H3938
AND DIF39        :  FUN 39(WR0080)
 [               :   ]
 WR0080=H4645    :
 WR0081=H4443    :
 WR0082=H4241    :
```

[Program explanation]

- ASCII data F, E, D, C, B, A, 9, and 8 which are set in WR0080 to WR0083 are converted to binary data. The conversion results are set in WR0084 and WR0085.

  Execution result: WR0080 = H4645, WR0081 = H4443, WR0082 = H4241, WR0083 = H3938, DR0084 = HFEDCBA98

[Precautions]

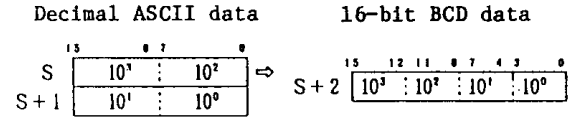
- When 8-digit ASCII codes which are set in S and S+3 are other than H30 to H39 and H41 to H46, DER = 1 and no processing is performed. This is not applied to H00 and H20 (null, space) in digits which are suppressed to 0.
- When S to S+5 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] 4-digit decimal ASCII -> 16-bit BCD conversion (Decimal ASCII to BCD)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 40 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (DABCD (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·					
Command format	No. of steps					51	56	107	117	
FUN 40 (S)	Conditions			Step						
* (DABCD (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (ASCII data)							o						Combinations of H00, H20, and H30 to H39
S+1	Argument (ASCII data)							o						S to S+2 are used.

[Function]

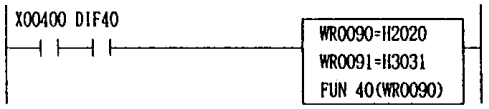


10^m: ASCII code in the 10^m place
10^n: BCD code in the 10^n place

* The contents of ( ) are indications when the ladder editor is used.

- The 4-digit decimal ASCII data specified by the arguments S and S+1 is converted to 16-bit BCD data and set in S+2.
- H00 and H20 (null, space) in the high order positions are processed as H30 (0). (0-suppressed digits)
- The argument is a combination of H30 to H39 (0 to 9).
- When operations are performed normally, DER = 0.

[Program example]




```
LD X00400
AND DIF40
 [
WR0090=H2020
WR0091=H3031
FUN 40(WR0090)
 ]
```

[Program explanation]

- ASCII data ␣, ␣, 0, and 1 which are set in WR0090 and WR0091 are converted to 16-bit BCD data. The conversion results are set in WR0092. (␣ = H20)

  Execution result: WR0090 = H2020, WR0091 = H3031, WR0092 = H0001

[Precautions]

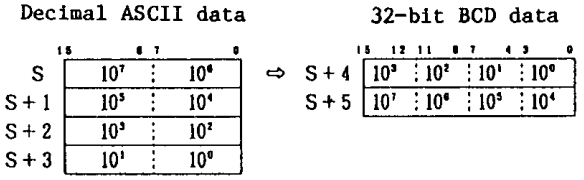
- When 4-digit ASCII codes which are set in S and S+1 are other than H30 to H39, DER = 1 and no processing is performed. This is not applied to H00 and H20 (null, space) in digits which are suppressed to 0.
- When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] 8-digit decimal ASCII -> 32-bit BCD conversion (Double decimal ASCII to BCD)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 41 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (DDABCD (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·					
Command format	No. of steps					82	90	173	191	
FUN 41 (S)	Conditions			Step						
* (DDABCD (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (ASCII data)							o						Combinations of H00, H20, and H30 to H39
S+3	Argument (ASCII data)							o						

[Function]

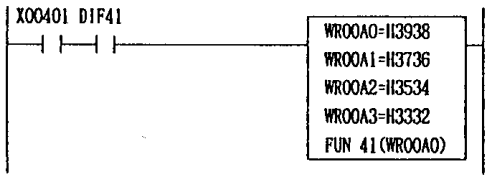


10^m: ASCII code in the 10^m place

* The contents of ( ) are indications when the ladder editor is used.

- The 8-digit decimal ASCII data specified by the arguments S to S+3 is converted to 32-bit BCD data and set in S+4 (low order) and S+5 (high order).
- H00 and H20 (null, space) in the high order positions are processed as H30 (0). (0-suppressed digits)
- The argument is a combination of H30 to H39 (0 to 9).
- When operations are performed normally, DER = 0.

[Program example]




```
LD X00401          :  WR00A3=H3332
AND DIF41          :  FUN 41(WR00A0)
 [                 :   ]
WR00A0=H3938       :
WR00A1=H3736       :
WR00A2=H3534       :
```

[Program explanation]

- ASCII data 9, 8, 7, 6, 5, 4, 3, and 2 which are set in WR00A0 to WR00A3 are converted to 32-bit BCD data. The conversion results are set in WR00A4 and WR00A5.

  Execution result: WR00A0 = H3938, WR00A1 = H3736, WR00A2 = H3534, WR00A3 = H3332, DR00A4 = H98765432

[Precautions]

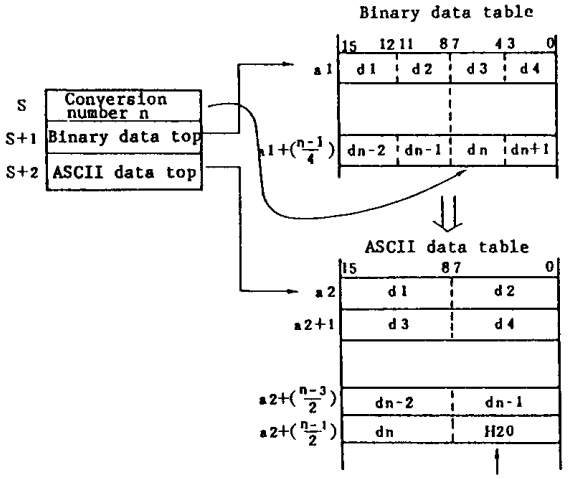
- When 8-digit ASCII codes which are set in S to S+3 are other than H30 to H39, DER = 1 and no processing is performed. This is not applied to H00 and H20 (null, space) in digits which are suppressed to 0.
- When S to S+5 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] Hexadecimal binary -> Hexadecimal ASCII conversion (Binary to ASCII)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 42 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		10-character conversion
* (ASC (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	281		568		
Command format	No. of steps									
FUN 42 (S)	Conditions			Step						
* (ASC (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	No. of conversion characters							o						S to S+2 are used.
S+1	Binary data top I/O No.							o						The real address is set.
S+2	Conversion ASCII top I/O No.							o						The real address is set.

[Function]



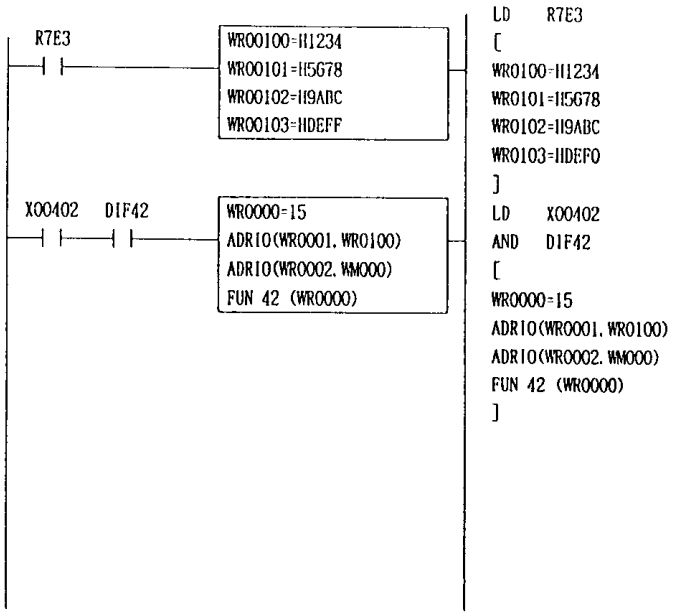
- The characters specified by S from the top I/O of the hexadecimal binary data specified by S+1 are converted to hexadecimal ASCII codes and set starting with the top I/O specified by S+2.
- When the number of characters is odd, the low-order 8 bits of the data of the output destination is H20 (space).
- Set the real addresses in the top I/O of S+1 and S+2 using the ADRIO instruction.
  Example: ADRIO (S+1, data top I/O)
- When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

- Set the real addresses in S+1 and S+2 using the ADRIO instruction. When any other addresses are set, DER = 1 and no processing is performed.
- When S to S+2 or the areas specified by them are overlapped, DER = 1 and no processing is performed.
- When S to S+2 or the areas specified by S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Program example]



[Program explanation]

1) Data is set in the data tables WR0100 and the following by the special internal output R7E3 (1 scan is turned ON after the running starts).

2) A hexadecimal number is converted to a hexadecimal ASCII at the leading edge of X00402 and the converted data is set in WM000 and the following.

Execution result:

WR0100=H1234	⇨	WM000=H3132, WM001=H3334
WR0101=H5678		WM002=H3536, WM003=H3738
WR0102=H9ABC		WM004=H3941, WM005=H4243
WR0103=HDEFF		WM006=H4445, WM007=H46 20

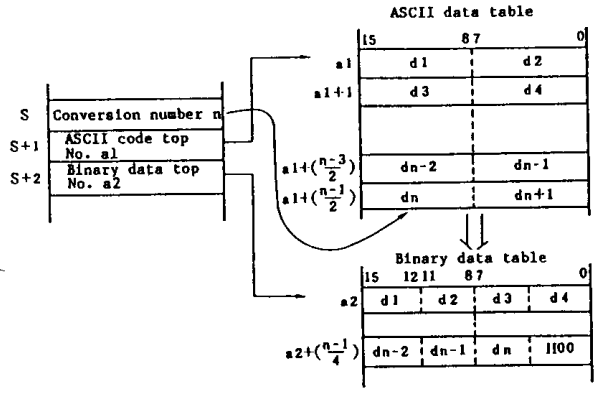
↑ Space

[Name] Hexadecimal ASCII -> Hexadecimal binary conversion (ASCII to binary)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 43 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		10-character conversion
* (HEX (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	195		370		
Command format	No. of steps									
FUN 43 (S)	Conditions			Step						
* (HEX (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	No. of conversion characters							o						S to S+2 are used.
S+1	ASCII code top I/O No.							o						The real address is set.
S+2	Conversion binary data top I/O No.							o						The real address is set.

[Function]



* The contents of ( ) are indications when the ladder editor is used.

. The characters specified by S from the beginning of the hexadecimal ASCII code specified by S+1 are converted to binary data and set starting with the top I/O specified by S+2.

. When the number of characters is odd, the data of low-order 4 bits of the output destination is 0.

. Set the real addresses in the top I/O of S+1 and S+2 using the ADRIO instruction.
ADRIO (S+1, data top I/O)

. H00 and H20 (null, space) in the high order positions are processed as H30 (0).
(0-suppressed digits)

. When operations are performed normally, DER = 0.

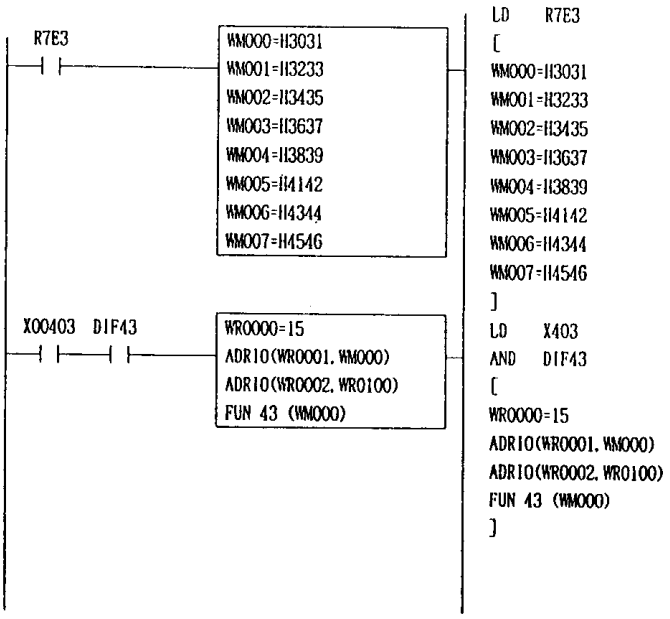
[Precautions]

. Set the real addresses in S+1 and S+2 using the ADRIO instruction. When any other addresses are set, DER = 1 and no processing is performed.

. When S to S+2 or the areas specified by them are overlapped, DER = 1 and no processing is performed.

. When S to S+2 or the areas specified by S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Program example]



[Program explanation]

1) Data is set in the data tables WM000 and the following by the special internal output R7E3 (1 scan is turned ON after the running starts).

2) A hexadecimal ASCII is converted to a hexadecimal binary number at the leading edge of X00403 and the converted data is set in WR0100 and the following.

Execution result: WM000=H3031, WM001=H3233
WM002=H3435, WM003=H3637
WM004=H3839, WM005=H4142
WM006=H4344, WM007=H4546
⇨
WR0100=H0123
WR0101=H4567
WR0102=H89AB
WR0103=HCDE0

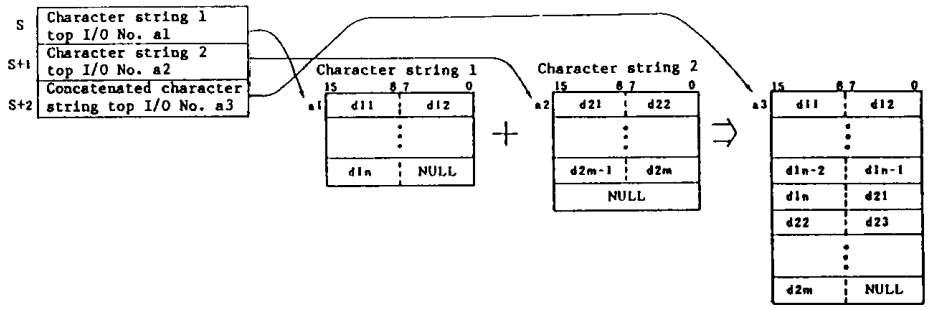
FUN 43 (HEX)

[Name] Character string concatenation

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 44 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		10 characters + 10 characters -> 20 characters
* (SADD (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	261		514		
Command format	No. of steps									
FUN 44 (S)	Conditions				Step					
* (SADD (S))	-				3					

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Character string 1 top I/O No.							o						The real addresses are set in S to S+2.
S+1	Character string 2 top I/O No.							o						
S+2	Concatenated character string top I/O No.							o						

[Function]



. The character string starting with the top I/O specified by S is concatenated with the character string starting with the top I/O specified by S+1 and set in the top I/O area specified by S+2.

. The character string to be concatenated is assumed to end prior to null (H00).

. Null is set behind the concatenated character string.

. Set the real addresses in the top I/O of S to S+2 using the ADRIO instruction.

. When operations are performed normally, DER = 0.

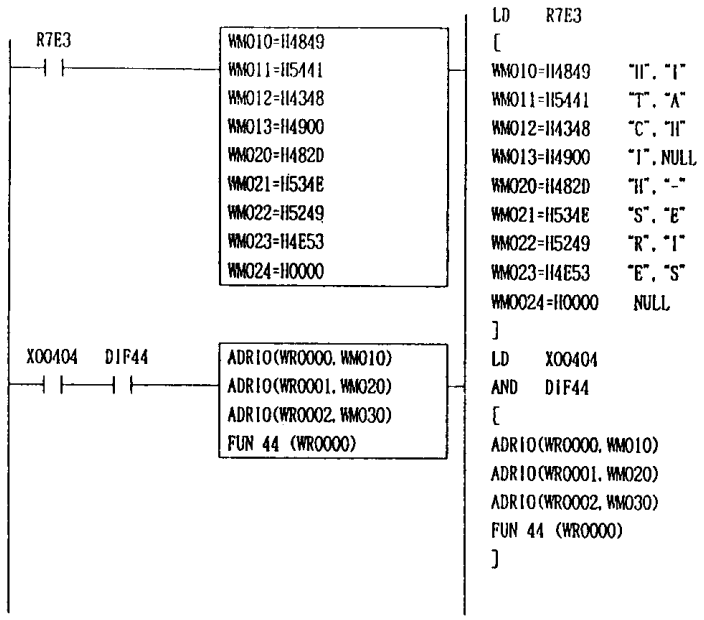
* The contents of ( ) are indications when the ladder editor is used.

FUN 44 (SADD)

[Precautions]

- When S to S+2 or the areas specified by them are overlapped, DER = 1 and no processing is performed.
- When S to S+2 or the areas specified by S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.
- Set the real addresses in S to S+2 using the ADRIO instruction. When any other addresses are set, DER = 1 and no processing is performed.

[Program example]



[Program explanation]

1) WM010 or the first character string and WM020 or the second character string are set using the special internal output R7E3 (1 scan is turned ON after the running starts).

2) The character strings are concatenated at the leading edge of X00404 and outputted to WM030 and the following.

Execution result:

WM010=H4849		WM020=H482D		WM030=H4849
WM011=H5441	+	WM021=H534E	⇨	WM031=H5441
WM012=H4348		WM022=H5249		WM032=H4348
WM013=H4900		WM023=H4E53		WM033=H4948
		WM024=H0000		WM034=H2D53
				WM035=H4E52
				WM036=H494E
				WM037=H5300

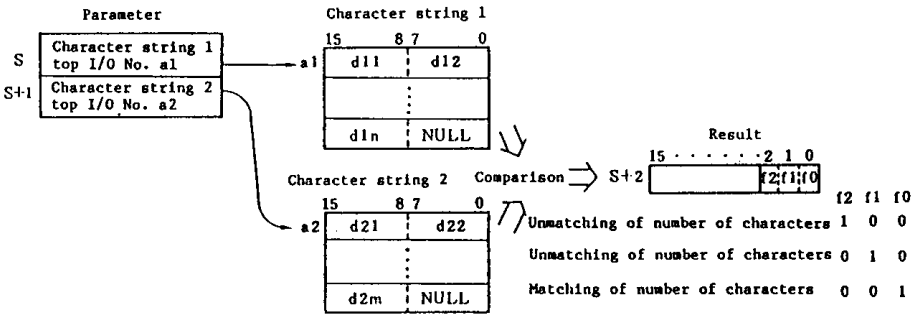
FUN 44 (SADD)

[Name] Character string comparison

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 45 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		10 characters are compared with 10 characters.
* (SCMP (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	239		486		
Command format	No. of steps									
FUN 45 (S)	Conditions			Step						
* (SCMP (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Character string 1 top I/O No.							o						The real addresses are set in S and S+1.
S+1	Character string 2 top I/O No.							o						S to S+2 are used.

[Function]



. The character string starting with the top I/O specified by S is compared with the character string starting with the top I/O specified by S+1 and the result is set in S+2.

. The character string to be compared is assumed to end prior to null (H00).

. Set the real addresses in the top I/O of S and S+1 using the ADRIO instruction.

. The character strings are compared with each other in unmatching of the number of characters. When the numbers of characters are not matched with each other, 1 is set in bit 2. When they are matched, the character strings are compared with each other. When the character strings are not matched with each other, 1 is set in bit 1. When the numbers of characters and the character strings are matched respectively, 1 is set in bit 0.

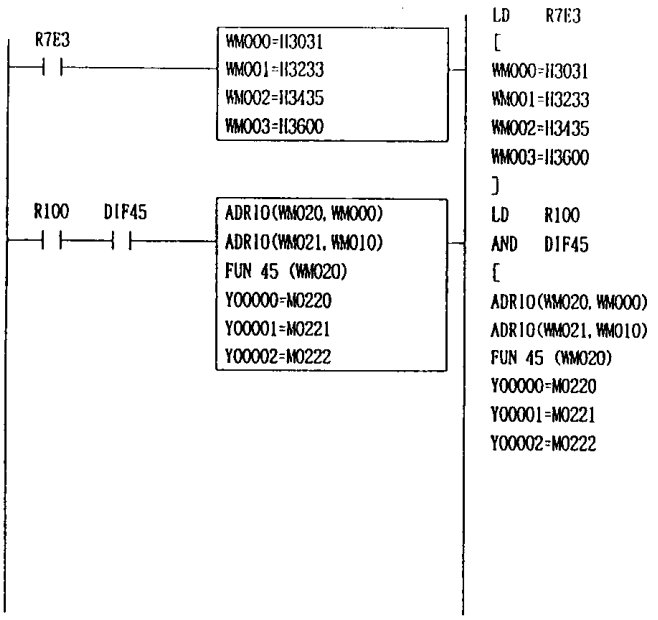
. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

- Set the real addresses in S and S+1 using the ADRIO instruction. When any other addresses are set, DER = 1 and no processing is performed.
- When S and S+1 or the areas specified by them are overlapped, DER = 1 and no processing is performed.
- When S+2 or the areas specified by S and S+1 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Program example]



[Program explanation]

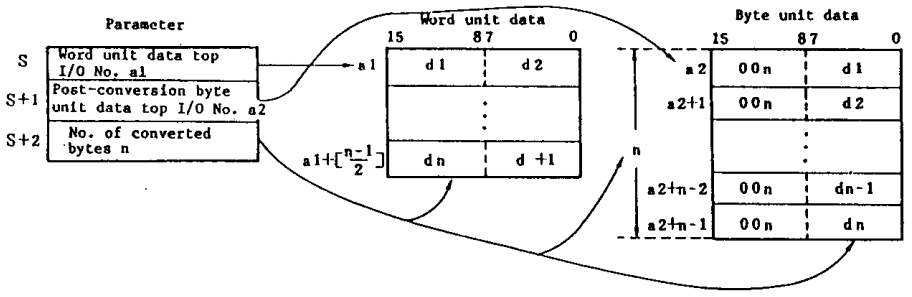
1) The comparison data is set in WM000 and the following using the special internal output R7E3 (1 scan is turned ON after the running starts).
2) The data of WM000 and the following is compared with the data of WM010 and the following at the leading edge of R100.
3) Y0000 to Y00002 are turned ON depending on the results.

[Name] Word unit -> Byte unit conversion (Conversion words to bytes)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 46 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		5 words -> 10 bytes (words)
* (WTOB (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·					
Command format	No. of steps					156		313		
FUN 46 (S)	Conditions			Step						
* (WTOB (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Word data top I/O No.							o						The real addresses are set in S and S+1.
S+1	Byte conversion data top I/O No.							o						
S+2	No. of converted bytes							o						S to S+2 are used.

[Function]



. The word character string data of the top I/O specified by S is split byte by byte for the number of bytes specified by S+2 and set in the top I/O area specified by S+1.

. Set the real addresses in the top I/O of S and S+1 using the ADRIO instruction.

. H00 is set in the high-order bytes of each split data.

. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

. Set the real addresses in S and S+1 using the ADRIO instruction. When any other addresses are set, DER = 1 and no processing is performed.

. hen S to S+2 or the areas specified by them are overlapped, DER = 1 and no processing is performed.

. When S+1 and S+2 or the areas specified by S to S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.

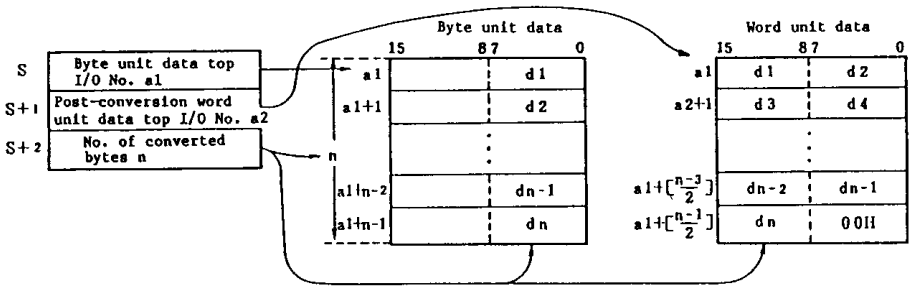
[Name] Byte unit -> Word unit conversion (Conversion bytes to words)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 47 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		10 bytes (words) -> 5 words
* (BTOW (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	.	.	.	.	151		304		
Command format	No. of steps									
FUN 47 (S)	Conditions			Step						
* (BTOW (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Byte unit data top I/O No.							o						The real addresses are set in S and S+1.
S+1	Word unit data top I/O No.							o						
S+2	No. of converted bytes							o						S to S+2 are used.

FUN 47 (BTOW)

[Function]



- The byte unit character string data in the number of bytes specified by S+2 is combined word by word from the data of the top I/O specified by S and set in the top I/O area specified by S+1.
- The high-order bytes of the byte unit data are ignored.
- When the number of converted bytes is odd, the low-order 8 bits at the end of the output destination are H00.
- Set the real addresses in the top I/O of S and S+1 using the ADRIO instruction.

* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

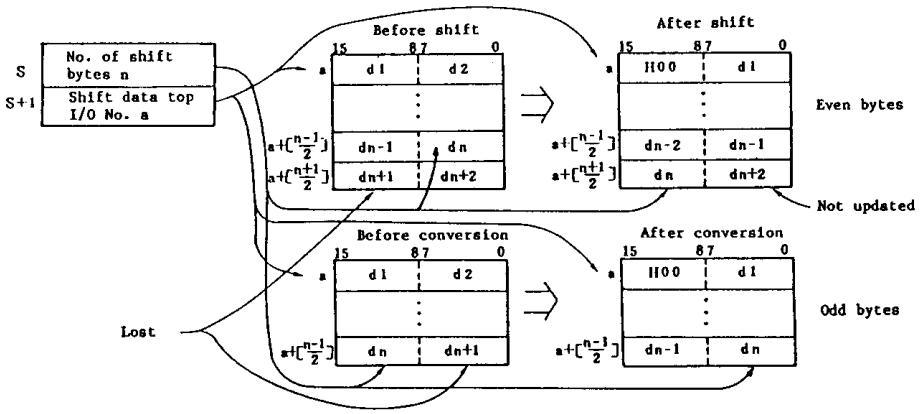
- Set the real addresses in S and S+1 using the ADRIO instruction. When any other addresses are set, DER = 1 and no processing is performed.
- When S to S+2 or the areas specified by them are overlapped, DER = 1 and no processing is performed.
- When S+1 and S+2 or the areas specified by S to S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] Byte right shift

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 48 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		10 characters are shifted one byte position.
* (BSHR (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	145		294		
Command format	No. of steps									
FUN 48 (S)	Conditions			Step						
* (BSHR (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD, SS, WDT, MS, TMR, CU, RCU, CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	No. of shift bytes							o						
S+1	Shift data top I/O No.							o						The real address is specified.
														S and S+1 are used.

[Function]



. The data in the number of bytes specified by S is shifted right one byte position from the top I/O data specified by S+1.

. In the empty area after shifting, H00 is set. The next data of the specified number of bytes is lost by shifting.

. Set the real address in the top I/O of S+1 using the ADRIO instruction.

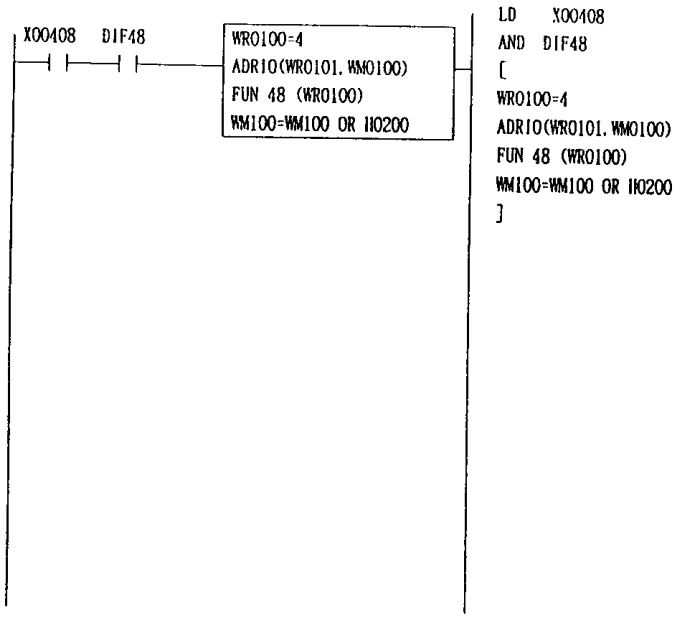
. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

- Set the real address in S+1 using the ADRIO instruction. When any other address is set, DER = 1 and no processing is performed.
- When S and S+1 or the areas specified by them are overlapped, DER = 1 and no processing is performed.
- When S+1 or the areas specified by S and S+1 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Program example]



[Program explanation]

- 4-byte transmission data is stored in WM100 and the following. A communication control code of H02 (STX) is added at the beginning of the data.

  Execution result:

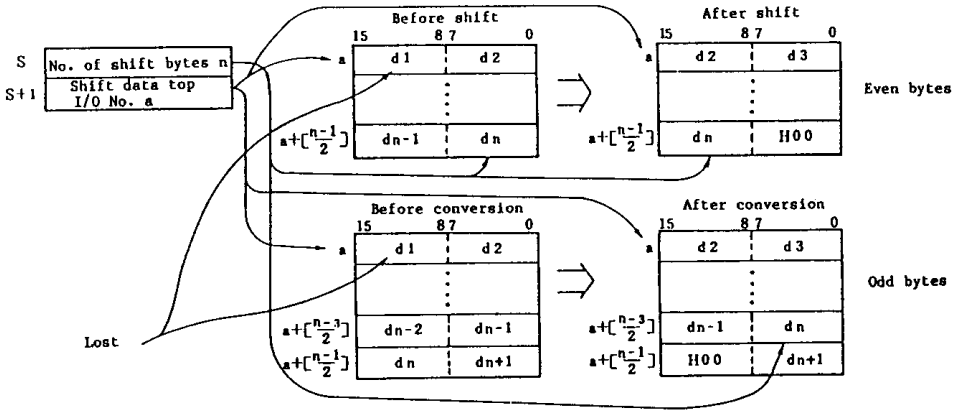
WM100	"T"	"E"	⇨	WM100	H02	"T"
WM101	"X"	"T"		WM101	"E"	"X"
WM102				WM102	"T"	

[Name] Byte left shift

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 49 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		10 characters
* (BSHL (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	159		306		
Command format	No. of steps									
FUN 49 (S)	Conditions			Step						
* (BSHL (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	No. of shift bytes							o						
S+1	Shift data top I/O No.							o						The real address is specified.
														S and S+1 are used.

[Function]



. The data in the number of bytes specified by S is shifted left one byte position from the top I/O data specified by S+1.

. In the empty area after shifting, H00 is set. The next data of the specified number of bytes is lost by shifting.

. Set the real address in the top I/O of S+1 using the ADRIO instruction.
ADRIO (S+1, data top I/O)

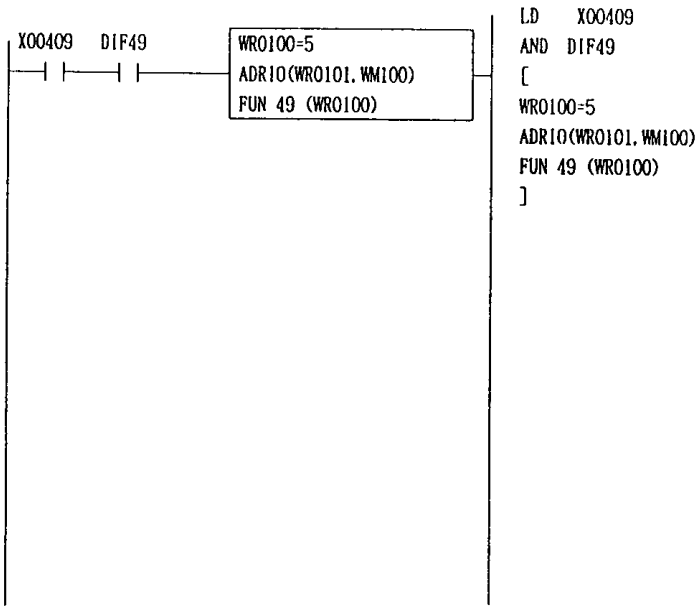
. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

- Set the real address in S+1 using the ADRIO instruction. When any other address is set, DER = 1 and no processing is performed.
- When S and S+1 or the areas specified by them are overlapped, DER = 1 and no processing is performed.
- When S+1 or the areas specified by S and S+1 are beyond the maximum I/O number, DER = 1 and no processing is performed.

[Program example]



[Program explanation]

- 5-byte data with a control code is stored in WM100 and the following. The control code is removed from each data and only the data forms a data string.

  Execution result:

[Name] Sampling trace set

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 50 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		Relevant instructions FUN 51 (S) FUN 52 (S)
* (TRSET (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·					
Command format	No. of steps									
FUN 50 (S)	Conditions			Step						
* (TRSE1 (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (dummy constant)							o						

[Function]

. The CPU instruction tracing is in the sampling execution enable state.

. When the instruction sampling is put into the execution state, the special internal output R7FD (sampling status) is turned ON.

. The instruction sampling is an end trigger function. When the sampling trace reset instruction (FUN52(S)) is normally executed, the special internal output R7FC (trigger matching flag) is turned ON. When the sampling trace set instruction is executed, the special internal output R7FC is turned OFF.

* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

. The special internal outputs R7FC (trigger matching flag) and R7FD (sampling status) are stored when the power fails.

. The word I/O (WR, WL, WM) specified by the argument (S) is used as a system area. Allocate one word to it. The word I/O (WR, WL, WM) specified as an argument (S) is not affected.

[Name] Sampling trace execution

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 51 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		Relevant instructions FUN 50 (S) FUN 52 (S)
* (TRASE (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.					
Command format	No. of steps									
FUN 51 (S)	Conditions			Step						
* (TRASE (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (dummy constant)							o						

[Function]

- The set sampling trace data is sampled.
- When the instruction sampling is put into the execution state, the special internal output R7FD (sampling status) is turned ON.

* The contents of ( ) are indications when the ladder editor is used.

FUN 51 (TRASE)

[Precautions]

- The special internal outputs R7FC (trigger matching flag) and R7FD (sampling status) are stored when the power fails.
- The word I/O (WR, WL, WM) specified by the argument (S) is used as a system area. Allocate one word to it. The word I/O (WR, WL, WM) specified as an argument (S) is not affected.

[Name] Sampling trace set

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 52 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		Relevant instructions FUN 50 (S) FUN 51 (S)
* (TRRES (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·					
Command format	No. of steps									
FUN 52 (S)	Conditions			Step						
* (TRRES (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (dummy constant)							o						

[Function]

- The CPU instruction tracing is in the sampling execution disable state.
- When the instruction sampling is put into the execution state, the special internal output R7FD (sampling status) is turned ON.
- The instruction sampling is an end trigger function. When the sampling trace reset instruction (FUN52(S)) is normally executed, the special internal output R7FC (trigger matching flag) is turned ON.

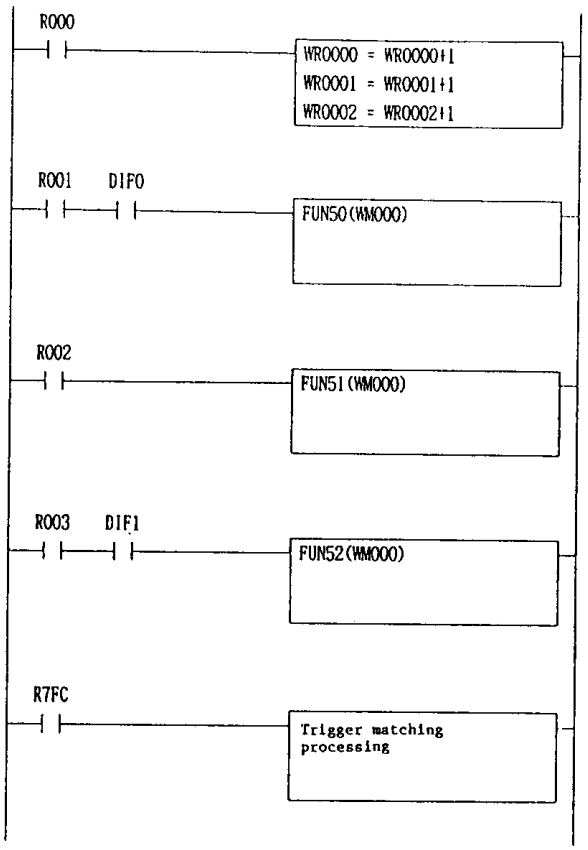
* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

- The special internal outputs R7FC (trigger matching flag) and R7FD (sampling status) are stored when the power fails.
- The word I/O (WR, WL, WM) specified by the argument (S) is used as a system area. Allocate one word to it. The word I/O (WR, WL, WM) specified as an argument (S) is not affected.

[Program example]

• When WR0002 is sampled from I/O WR0000




```
LD  R000
[
WR0000 = WR0000+1
WR0001 = WR0001+1
WR0002 = WR0002+1
]

LD  R001
AND DIF0
[
FUN 50 (WM000)
]

LD  R002
[
FUN 51 (WM000)
]

LD  R003
AND DIF1
[
FUN 52 (WM000)
]

LD  R7FC
[
Trigger matching processing
]
```


[Program explanation]

1) The I/O type and number to be sampled from the peripheral equipment (ladder editor) are set in the CPU.

2) The CPU is operated.

3) When the contact R000 is forced to be set, the status of the I/O to be sampled hereafter changes.

4) When the I/O status is sampled according to the sampling data (I/O type and number) which is set by the peripheral equipment, the sampling trace set instruction is activated by forcing the contact R001 to set and the sampling enable status is set.

5) When the contact R002 is forced to be set, the sampling trace instruction is executed and the I/O status is sampled in the sampling buffer which is set by the peripheral equipment.

6) When the sampling is finished, the sampling trace reset instruction is executed by forcing the contact R3 to set and the I/O status is not sampled even if the sampling trace instruction is activated.

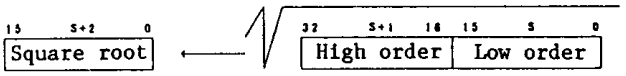
For further details, refer to the ladder editor operation manual.

[Name] Binary square root

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 60 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (BSQR (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	168	185	312	382	
Command format	No. of steps									
FUN 60 (S)	Conditions		Step							
* (BSQR (S))	-		3							

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (low order)							o						
S+1	Argument (high order)							o						S to S+2 are used.

[Function]

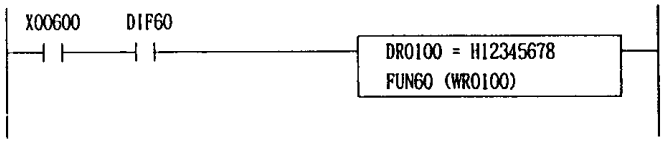


. The square root of the 32-bit binary value specified by S (low order) and S+1 (high order) is calculated and the result is set in S+2.

. When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Program example]




```
LD   X00600
AND DIF60
[
DR0100 = H12345678
FUN60 (WR0100)
]
```

[Program explanation]

. 32-bit binary data H12345678 is set in WR0100 and WR0101.

. The operation result is set in WR0102.

Result: DR0100 = H12345678
WR0102 = H4444(17476)

[Precautions]

. When S+1 and S+2 are beyond the maximum I/O number, DER = 1 and no processing is performed.

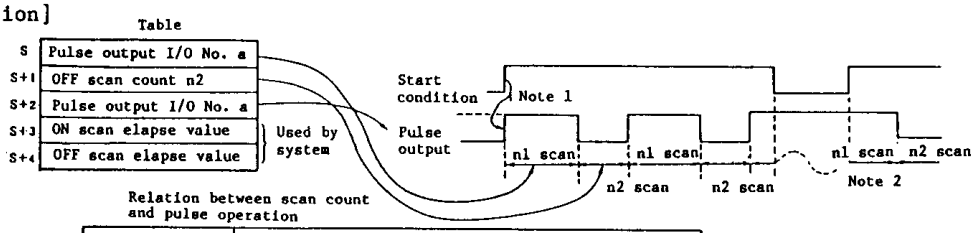
[Name] Dynamic scan pulse instruction (Pulse generating function)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 61 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (PGEN (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	↕	·	·	·	·	117	151	234	304	
Command format	No. of steps									
FUN 61 (S)	Conditions			Step						
* (PGEN (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	ON scan count							o						
S+1	OFF scan count							o						
S+2	Pulse output I/O			o										The real address is set.
S+3 S+4	System area													Cannot be used by the user.

FUN 61 (PGEN)

[Function]



Relation between scan count and pulse operation

Scan count		Pulse operation
n1	n2	
n1 = 0	n2 = 0	The pulse output is turned OFF.
	n2 ≥ 1	
n1 ≥ 1	n2 = 0	The pulse output is turned ON.
	n2 ≥ 1	The pulse output turns n1 scan ON and n2 scan OFF.

- The operation that the bit internal output specified by S+2 is turned ON by the scan count specified by S and OFF by the scan count specified by S+1 is repeated.
- When this instruction is executed several times during one scan, the output is turned ON or OFF depending on the execution count.
- When both S and S+1 are 0, the output is left OFF.
- When the start condition is turned OFF, the output and the elapse values of S+3 and S+4 are held.
- When S+3 and S+4 are not cleared at the time of initialization, the pulse width of the first one cycle may be changed.
- Set the real address in the pulse output I/O using the ADRIO instruction.
  ADRIO (S+2, bit internal output)
- When operations are performed normally, DER = 0.

* The contents of ( ) are indications when the ladder editor is used.

[Precautions]

- The pulse output is changed a maximum of one scan behind the start condition. When the start condition is changed from ON to OFF and then to ON halfway, the pulse width during that time is changed ±1 scan.
- Set the real address in the pulse output I/O of S+2 using the ADRIO instruction. When any other address is set, no processing is performed.
- When S+1 to S+4 or the area specified by S+2 is beyond the maximum I/O number, DER = 1 and no processing is performed.

[Name] Box comment

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 254 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (BOXC (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	·	·	·	·	·	1.0	←	3.0	←	
Command format	No. of steps									
FUN 254 (S)	Conditions			Step						
* (BOXC (S))	-			3						

		Bit				Word				Double word				
Usable I/O		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM	Constant	Other
S	Argument (dummy constant)							o						

[Function]

- This instruction executes nothing but prints comments on the right of the arithmetic box in combination with the ladder editor.
- The maximum number of comment characters is 32.

* The contents of ( ) are indications when the ladder editor is used.

[Name] Memo comments

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 255 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2002		H-702/302		
* (MEMC (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	.	.	.	.	.	1.0	←	3.0	←	
Command format	No. of steps									
FUN 255 (S)	Conditions			Step						
* (MEMC (S))	-			3						

Usable I/O		Bit				Word				Double word			Constant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR, WL, WM	TC	DX	DY	DR, DL, DM		
S	Argument (dummy constant)							o						

[Function]

. This instruction executes nothing but prints comments between the circuits under the arithmetic box in combination with the ladder editor.

. A maximum of one screen (66 characters by 16 lines) can be used for comments.

* The contents of ( ) are indications when the ladder editor is used.

PID function

The CPU2-**H and CPUP-**H have instructions for PID control.
For the outline of the PID function, refer to Section 3.4 of the hardware manual.

(1) PID operation instructions

There are three PID instructions available as shown below.

FUN 0 (S) * (PIDIT (S))	Initialization of the PID operation function
FUN 1 (S) * (PIDOP (S))	Execution management of PID operation
FUN 2 (S) * (PIDCL (S))	Execution of PID operation

* The contents of ( ) are indications when the ladder editor is used.

(2) Details of the PID instructions

Details are shown on the next and subsequent pages.

[Name] PID initialization (PID initialize)

Ladder format	Condition code					Processing time (μs)		Remarks
FUN 0 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	H-2000, H-700, and H-300 are only for the CPU of CPUP-**H. n = Loop count
* (PIDIT (S))	DER	ERR	SD	V	C	875 + 146 n	1786 + 300 n	
	.	.	.	.	.			
Command format	No. of steps					H-2002	H-702/302	
FUN 0 (S)	Conditions			Step		730 + 122 n	1490 + 249 n	
* (PIDIT (S))	-			3				

Usable I/O		Bit				Word					Double word			Con-stant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR	WL, WM	TC	DX	DY	DR, DL, DM		
S	PID management table							o							

[Function]

. FUN0(S) initializes the area for storing the initialization data necessary for PID operations.

. S of FUN0(S) specifies the WR top No. of the PID management table.

. When any error is found in the contents shown in the PID management table, an error code is set in Error Code 0 of the PID management table and the area is not initialized.

. When FUN0 is executed once again after the area is correctly initialized (FUN0 normal termination (1) of the PID management table) once, an error is caused.

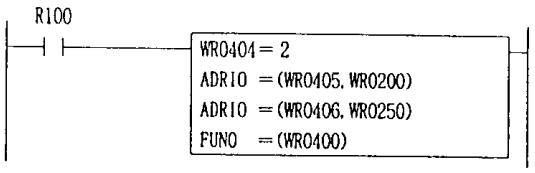
* The contents of ( ) are indications when the ladder editor is used.

For the PID management table, loop word table, and loop bit table, see Section 3.4(3), "Details of PID management table".

[Precautions]

If a control fault occurs when the area to be used for the PID operation is cleared at the time of operation start or start after power failure, specify Power Failure Storage.

[Program example]



1) The loop count is specified as 2.

2) } The real address of the WR top I/O of
3) } the loop word table area is set.

4) The PID area is initialized.

[Program explanation]

The following is a program example that WR0400 and the following of the PID management table are used, the loop count is set to 2, the loop 1 word table area ranges from WR0200 to WR022F (48 words), and the loop 2 word table area ranges from WR0250 to WR027F (48 words).

PID management table

WR0400	Error code 0 (R)	
WR0401	Error code 1 (R)	
WR0402	Error code 2 (R)	
WR0403	FUN 0 normal termination (R)	
WR0404	Loop count (W)	WR404 = 2
WR0405	Real address of WR top No. of the loop 1 word table (W)	ADRIO = (WR0405, WR0200)
WR0406	Real address of WR top No. of the loop 2 word table (W)	ADRIO = (WR0406, WR0250)

When FUN0 (WR0400) is executed, the areas from WR0200 to WR022F and from WR0250 to WR027F are checked and initialized.

[Name] PID execution management (PID operation control)

Ladder format	Condition code					Processing time (μs)				Remarks
FUN 1 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000		H-700/300		H-2000, H-700, and H-300 are only for the CPU of CPUP-**H.
* (PIDOP (S))	DER	ERR	SD	V	C	Average	Maximum	Average	Maximum	
	•	•	•	•	•		155		310	
Command format	No. of steps					H-2002		H-702/302		
FUN 1 (S)	Conditions			Step		Average	Maximum	Average	Maximum	
* (PIDOP (S))	-			3			129		258	

Usable I/O		Bit					Word					Double word				
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR	WL, WM	TC	DX	DY	DR, DL, DM	Con-stant	Other
S	PID management table top							o							

[Function]

. FUN1(S) reads the PLD execution flag and the PLD constant change flag of the loop bit table area and determines the loop for performing operations.

. Set the WR top No. of the PID management table in S of FUN1(S). When any other number is set, an error is caused. An error code is set in Error Codes 0 and 1 of the PID management table and FUN1 is not executed.

. Create a program so that FUN1(S) is executed once during the periodic scan of 20 ms.

* The contents of ( ) are indications when the ladder editor is used.

For the PID management table, loop word table, and loop bit table, see Section 3.4 (3), "Details of PID management table".

[Name] PID calculation

Ladder format	Condition code					Processing time (μs)		Remarks
FUN 2 (S)	R7F4	R7F3	R7F2	R7F1	R7F0	H-2000	H-700/300	H-2000, H-700, and H-300 are only for the CPU of CPUP-**H. n = Loop count
* (PIDCL (S))	DER	ERR	SD	V	C	1900 + 15 n	3590 + 30 n	
	.	.	.	.	.			
Command format	No. of steps					H-2002	H-702/302	
FUN 2 (S)	Conditions			Step		1583 + 13 n	2992 + 25 n	
* (PIDCL (S))	-			3				

Usable I/O		Bit					Word				Double word			Con-stant	Other
		X	Y	R, L, M	TD,SS,WDT, MS,TMR,CU, RCU,CT	WX	WY	WR	WL, WM	TC	DX	DY	DR, DL, DM		
S	Word table top							o							

[Function]

. Whether or not to calculate PID is determined by the sampling time set in each loop word table.

. FUN2(S) turns the loop PID calculation flag to be calculated ON.

. Set all the WR top numbers of each PID loop word table by FUN2(S).

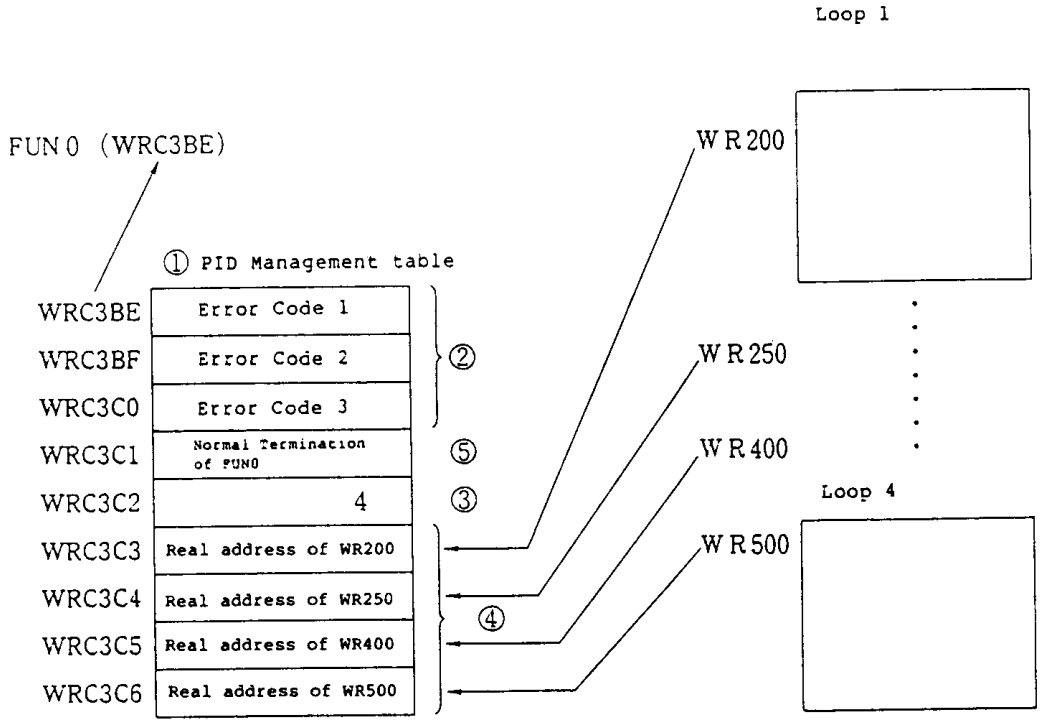
. FUN2(S) checks each loop output upper limit value, lower limit value, set value bit pattern range, and output value bit pattern range. When an error occurs, the FUN2 error flag of the loop bit table is turned ON and an error code is set in Error Code 2 of the PID management table. Even if an error occurs, the FUN2 processing is continued.

. Create a program so that FUN2(S) is executed once during the periodic scan of 20 ms.

* The contents of ( ) are indications when the ladder editor is used.

For the PID management table, loop word table, and loop bit table, see Section 3.4 (3), "Details of PID management table".

Details of FUN0(S)



(a) FUN0 performs the following on respective loop tables (WR200, WR250, WR400, and WR500):

1) Validity check of the address of the top of each loop bit table (11), high output limit (17) to initial value (19), and SV bit pattern (23) to MV bit pattern (25)

The result of the check is set in error code 0 (2).

2) Initialization of the PID operation area (26) of the loop which passed the validity check

(b) When all loops (specified by No. of Loops (3) and Top Address of WR of each loop word table) are valid, the "Normal Termination of FUN0" field is set "1". Then, FUN1 and FUN2 start PID operations. When an error occurs in the execution of FUN0, the "Normal Termination of FUN0" field is set "0". The error is set in the "Error Code 0" field (2). Correct the program, then try the FUN0 instruction again.

(c) If the FUN0 instruction is executed again after the "Normal Termination of FUN0" fields of all loops are set "1", the instruction abends.

FUN1(S) (Manage PID Execution)

<table>
<tr><td></td><td>Function 1
Management of
PID Execution</td><td colspan="5"><table><tr><td>DER</td><td>ERR</td><td>SD</td><td>V</td><td>C</td></tr><tr><td>.</td><td>.</td><td>.</td><td>.</td><td>.</td></tr></table></td><td></td></tr>
<tr><td>Instruction format</td><td>FUN1(S)</td><td colspan="5"><table><tr><td>Number of steps</td></tr><tr><td>3</td></tr></table></td><td></td></tr>
<tr><td>I/O type</td><td colspan="6">1. S must not be an array variable.
2. The WR area specified for S must be protected against unexpected power failures (by the specification of "Power-Off Protection"). Otherwise, the data in the WR area would be all cleared when the power is shut off.<table><tr><td>Words</td><td>S</td><td>WX</td><td>WY</td><td>WR</td><td>WL</td><td>WM</td><td>TC</td><td>Constant</td></tr><tr><td></td><td></td><td></td><td></td><td>o</td><td></td><td></td><td></td><td></td></tr></table></td></tr>
</table>

Outline of FUN1(S) processing

The FUN1(S) instruction fetches the Execution flag 50 and the PID Constant Change flag 52 and determines a process loop to be PID-operated. See 3.9.1 "Timing diagrams example 1".

Programming notes

(a) The content of "S" of the FUN1(S) must be equal to that of "S" of the FUN0(S). If not, an error occurs and the FUN1 instruction is disabled.

(b) The FUN1(S) should be coded so that it may be executed only once in each 20-ms periodic scanning.

(c) Do not code a 10-ms periodic scanning in this period. If coded, a congestion error will occur.

FUN2(S) (Execution of PID Operation)

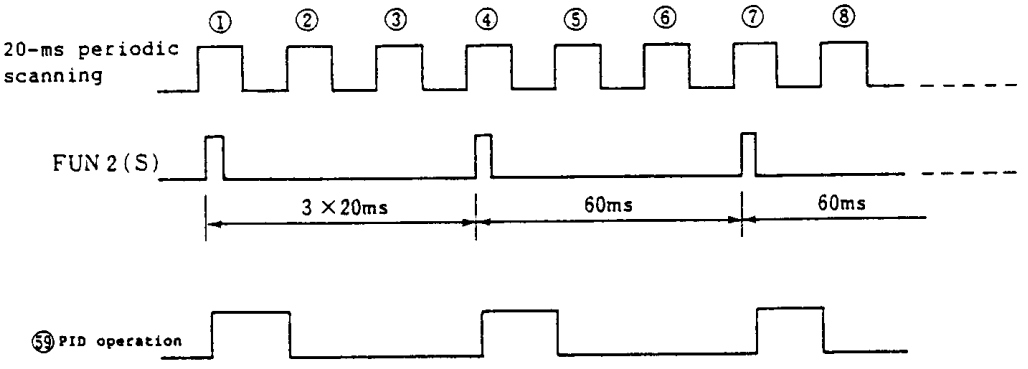
	Function 2 Execution of PID Operation	DER	ERR	SD	V	C	
		.	.	.	.	.	
Instruction format	FUN2(S)	Number of steps					
		3					
I/O type	1. S must not be an array variable. 2. The WR area specified for S must be protected against unexpected power failures (by the specification of "Power-Off Protection"). Otherwise, the data in the WR area would be all cleared when the power is shut off.						

Words	S	WX	WY	WR	WL	WM	TC	Constant
				o				

Outline of FUN2(S) processing

(a) The FUN2(S) instruction determines whether or not the PID operation is to be executed according to the preset sampling time (12).

For example, when the preset sampling time (12) is 3 ms and the actual sampling time is 60 ms, PID operations are carried out at periodic scans (1), (4), and (7). At the other periodic scans ((2), (3), (5), (6), (8), ...), PID operations are not carried out.



The FUN2(S) instruction turns on the PID Execution flag (59) of a process loop on which the PID operation will be performed.

For execution of PID operations on more than one process loop, see 3.4 "FUN1(S)".

(b) The FUN2(S) instruction checks the contents of High Output Limit (17), Low Output Limit (18), and SV Bit Pattern (23) to MV Bit Pattern (25) parameters. If an error is detected, the "FUN2 Error" field (63) is set to "1" and the error code is set in the "Error Code 2". The FUN2 instruction is not held by this error. When no error is detected, the "FUN2 Error" field (63) is set to "0".

(c) The PID operation is performed by the following expressions:

(55) D-FREI = 1

$$\text{Output value} = \frac{⑬K_P}{100} \cdot XD + \frac{⑬K_P}{100} \cdot \frac{XD}{⑭T_I/TZ} + Y1(n-1)$$

$$+ \frac{⑯T_n/TZ}{1+⑯T_n/TZ} YDT1(n-1) + \frac{⑮T_D/TZ}{⑯T_n/TZ} \times \frac{⑬K_P}{100}(XD - XD(n-1))$$

(55) D-FREI = 0

$$\text{Output value} = \frac{⑬K_P}{100} \cdot XD + \frac{⑬K_P}{100} \cdot \frac{XD}{⑭T_I/TZ} + Y1(n-1)$$

where XD: Deviation (= Set value - Measured value)
K_P: Proportional gain
T_n/TZ: Differential delay constant
T_D/TZ: Differential constant
T_I/TZ: Integral constant
XD(n-1): Previous deviation
Y1: Integral value expressed by

$$Y1 = \frac{⑬K_P}{100} \cdot \frac{XD}{⑭T_I/TZ} + Y1(n-1)$$

Y1(n-1): Previous integral value
YDT1: Differential value

$$YDT1 = \frac{⑯T_n/TZ}{1+⑯T_n/TZ} \left\{YDT1(n-1) + \frac{⑮T_D/TZ}{⑯T_n/TZ} \times \frac{⑬K_P}{100}(XD-(n-1))\right\}$$

YDT1(n-1): Previous differential value

(53) S flag = "1"

Output value = 19 INT

(54) R flag = "1"

Output value = 0

(d) The output value (obtained by (c)) is corrected by the high output limit (17) and the low output limit (18), as shown below.

Low output value (18) High output value (17)
Output value
The high output limit becomes an output value.

Low output value (18) Output value High output value (17)
The output value becomes an output value.

Output value Low output value (18) High output value (17)
The high output limit becomes an output value.

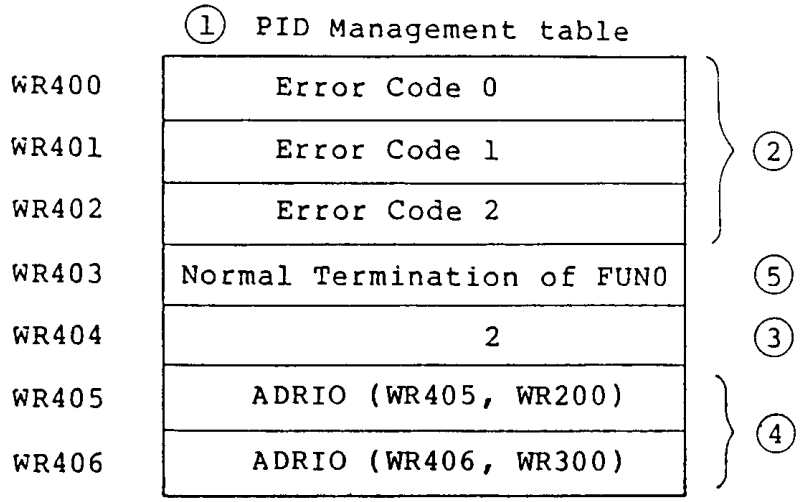
Low output value (18) High output value (17)
No output value
The error code is set in the "Error Code 2" field.

Programming notes

(a) The WR number specified for "S" of the FUN2(S) must be equal to the WR number of the WR area on the PID Management table for a selected process loop. If not, an error occurs and the FUN2 instruction is disabled.

(b) The FUN2(S) should be coded in 20-ms periodic scanning.

(c) It is required to code all FUN2(S) that are set in the top numbers of the WR areas of the word table for process loops (4) (in the PID Management table).



In the above PID Management table, FUN2 (WR200) and FUN2 (WR300) are alternately executed once in each 20-ms periodic scanning.

(d) FUN1 and FUN2 instructions should be coded so that the FUN2 instruction may be executed after the FUN1 instruction is executed in a 20-ms periodic scanning.

ADRIO (I/O Address Conversion)

	I/O Address Conversion	DER	ERR	S D	V	C		PROCESS TIME			
		·	·	·	·	·		H-2000		H-700 H-300	
Instruction format	ADRIO (d, S)	Number of steps						Ave.	Max.	Ave.	Max.
		3									

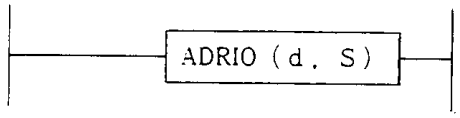
I/O type

1. Specify "d" in words and "S" in words or bits.
2. "d" and "S" must not be an array variable.

	WX	WY	WR	WL	WM	TC	Constant
d		○	○	○	○		
S	○	○	○	○	○		

	X	Y	R	L	M	TD	SS	SDT	MS	TMR	CU	RCT	CT	CTU	CTD	CL	DIF	DEN	Con-stant
S	○	○	○	○	○														

Operation

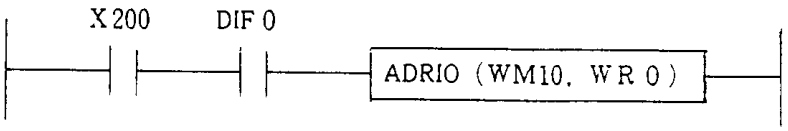


Explanation

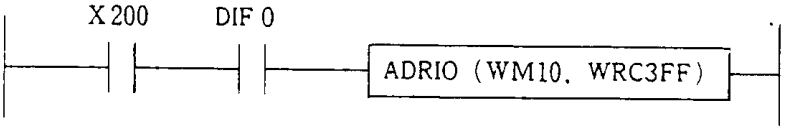
1. This instruction outputs the real address of "S" to "d".

Examples

1. Setting the real address "3C00H" of WR0 in WM10.



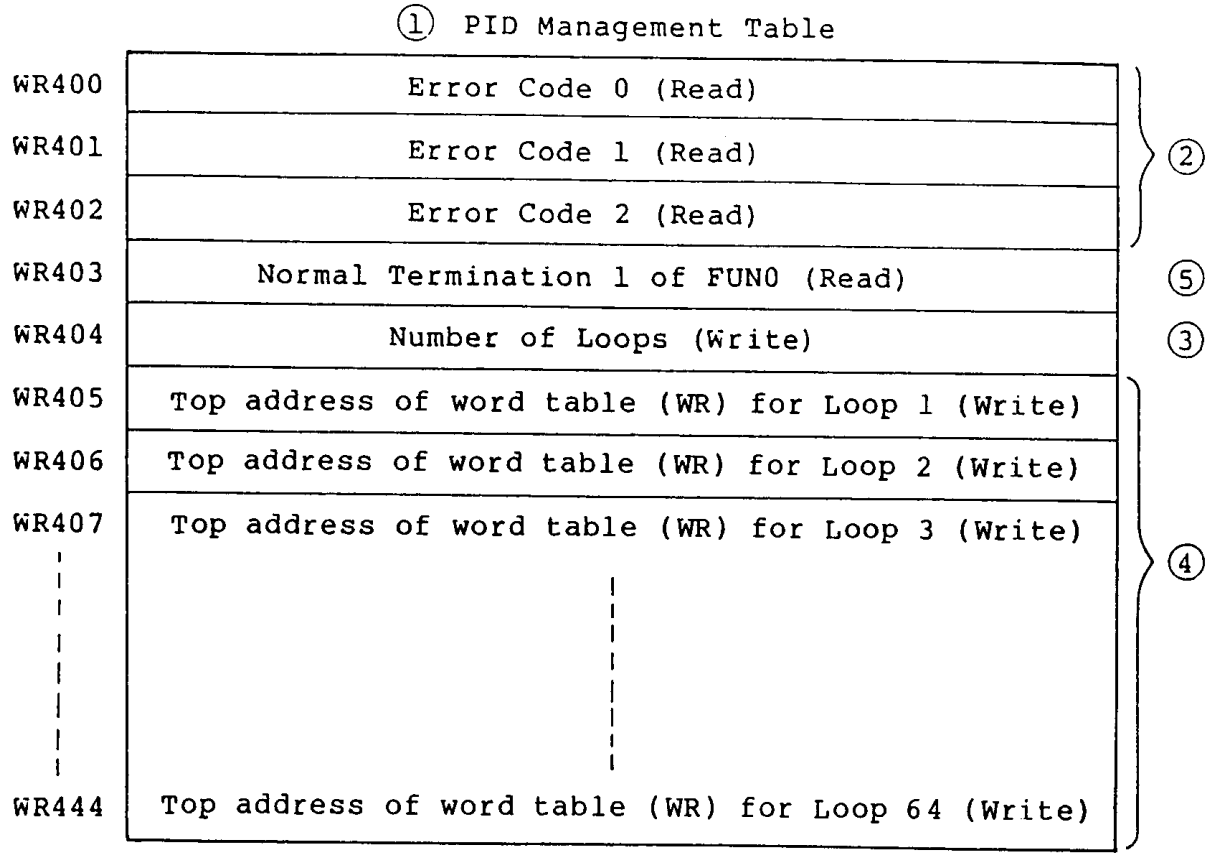
2. Setting the real address "FFFFH" of WRC3FF in WM10



PID Management Table

Explanation

[Example] FUN0 (WR400)



① PID Management Table

(a) "S" of the FUN0(S) instruction should be the top WR number of the PID Management table.

(b) The PID Management table consists of four parts: ②, ③, ④ and ⑤. The size of part 5 varies according to the number of loops. Part 5 must not exceed the end of the WR (see (c)). If part 5 exceeds the end of the WR, error code 0004 is written in the "Error Code 0" area.

(c) The available range of the WR varies according to the type of a memory cassette installed in the CPU module.

RAM-04H, RAM-08H:	WR0 to WR3FF
RAM-16H:	WR0 to WR43FF
RAM-48H:	WR0 to WRC3FF
ROM-16H:	WR0 to WR43FF

Note:
(Write): Area in which the user can write data (by programming), from which the user can read data

② Error Code 0, Error Code 1, and Error Code 2 (Read-only)

(a) Error code 0

An error code is set in this area when an error occurs in the execution of the FUN0 instruction or FUN1 instruction (partially). While no error is detected, this area holds the preceding status.

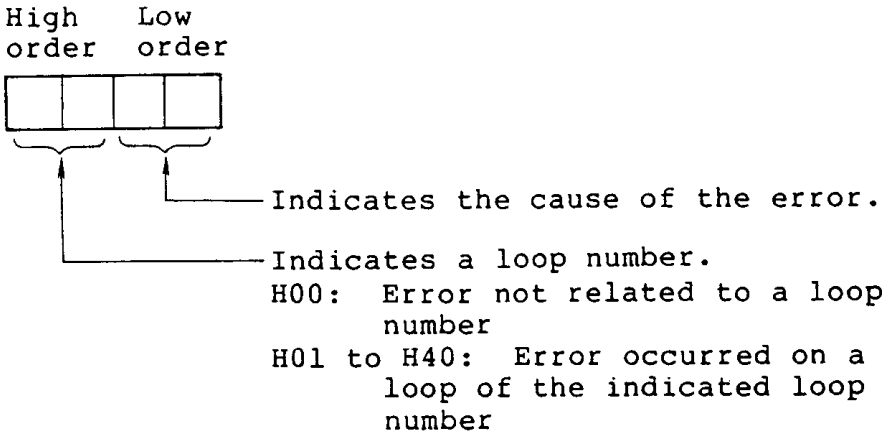
(b) Error code 1

An error code is set in this area when an error occurs in the execution of the FUN1 instruction. While no error is detected, this area holds the preceding status.

(c) Error code 2

An error code is set in this area when an error occurs in the execution of the FUN2 instruction. While no error is detected, this area holds the preceding status.

(d) An error code is expressed by four hexadecimal digits.



See 3.10 "Error Codes" for error codes set in the Error Code 0, Error Code 1, and Error Code 2 areas.

The loop number of H01 to H40 is set in the high-order "xx" of error codes in the error code list.

③ Number of Loops (Write)

(a) Set the number of loops used (1 to 64).

When "0" is set in this area, error code 0002 is set in the "Error Code 0" area and the PID control function is disabled (even when FUN1 and FUN2 instructions are already coded).

④ Top address of word table (WR) for Loop n (Write)

(a) One loop requires a WR of 48 words long for Input of PID constants and for internal calculation of the PID. Set the top address of the WR in this area by the address conversion instruction. The address must not exceed WRC3FF, (the end of the WR). If the address exceeds WRC3FF, error code xx05 is written in the Error Code 0 area. WRs of loops must not overlap each other.

(b) Specify WR addresses as many as the number of loops specified.

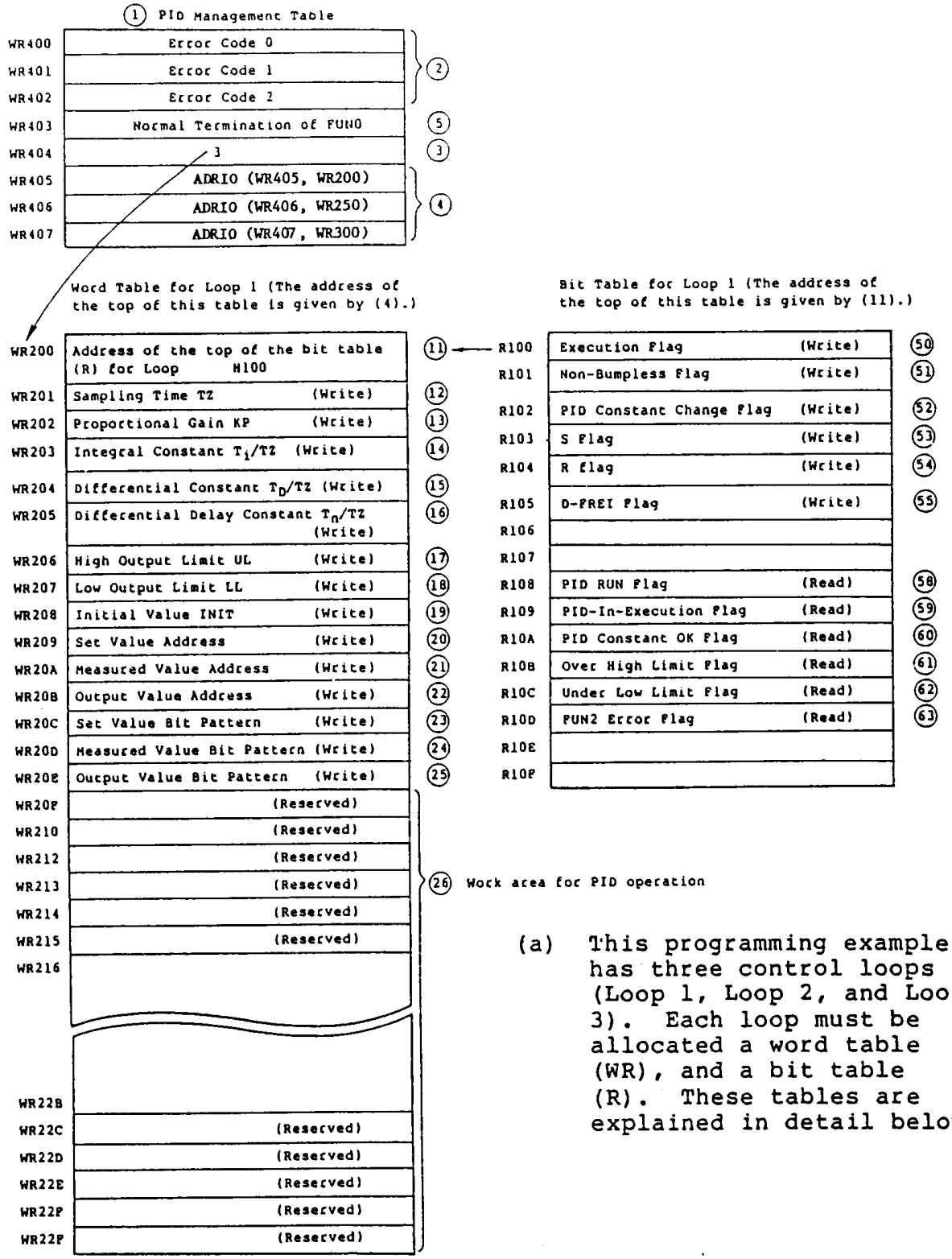
For example, when you write "5" in the "Number of Loops" area, specify the address of WR for loop 1 to the the address of WR for loop 5 (a total of five addresses).

See Section 3.6 for explanation of the ADRIO instruction.

⑤ Normal Termination of FUN0 (Read-only)

(a) Code 0001 is set in this area when the FUN0 instruction ends normally. When an error occurs in the execution of the FUN0 instruction, an error code is set in the Error Code 0 area.

Word tables and bit tables



(a) This programming example has three control loops (Loop 1, Loop 2, and Loop 3). Each loop must be allocated a word table (WR), and a bit table (R). These tables are explained in detail below.

⑪ Address of the Top of the Bit Table for Loop (Write)

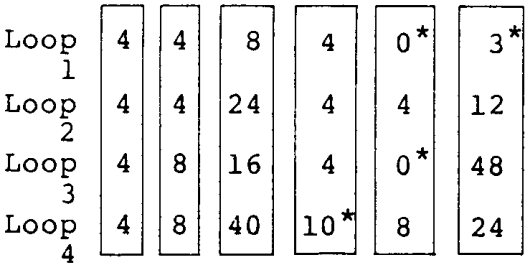
(a) One loop requires a bit table (R) of 16 bits long for control and monitor of the loop. Specify the bit number of the bit table (R) in the range of 0 to H7B0 by the ADRIO instruction. The bit number (address) must not exceed R7BF (the end of the bit table).

⑫ to ⑲

<table>
<tr><th>No.</th><th>Parameters</th><th>Specification</th><th>Remarks</th></tr>
<tr><td rowspan="2">⑫</td><td rowspan="2">Sampling Time TZ</td><td>1 to 200 ms (a multiple of 20 ms) (When the analog input output module is installed on the basic or expansion unit)</td><td rowspan="2">o The sampling time must be a multiple of a minimum set value.

o The minimum set value is what is set in the Number of Loops area (3).

(Example) Setting of sampling times (TZ) of 4 loops

Ex.1 to Ex.3: Valid
Ex.4 to Ex.6: Not valid (*)</td></tr>
<tr><td>4 to 200 ms (a multiple of 20 ms) (when the analog input output module is installed on the remote I/O sub-station)</td></tr>
<tr><td>⑬</td><td>Proportional Gain KP</td><td>-1000 - +1000</td><td>-10.00 to +10.00</td></tr>
</table>

No.	Parameters	Specification	Remarks
⑭	Integral Constant T_i/TZ	1 - 32767	Integration time (T_i) / Sampling time (TZ) Differential time (T_D) / Sampling time (TZ) Differential delay time (T_n) / Sampling time (TZ)
⑮	Differential Constant T_D/TZ		
⑯	Differential Delay Constant T_n/TZ		
⑰	High Output Limit UL	-32767 - +32767	These values must satisfy the following condition: $LL \leqq INIT \leqq UL$ The UL and LL values can be given hysteresis by a user program.
⑱	Low Output Limit LL		
⑲	Initial Value INIT		

⑳ Set Value Address (Write)

(a) This area sets the address of a word which stores a set value.

(Example) To set the address of WX500 in WR20A, code ADRIO (WR20A, WX500)

㉑ Measured Value Address (Write)

(a) This area sets the address of a word which stores the measured value.

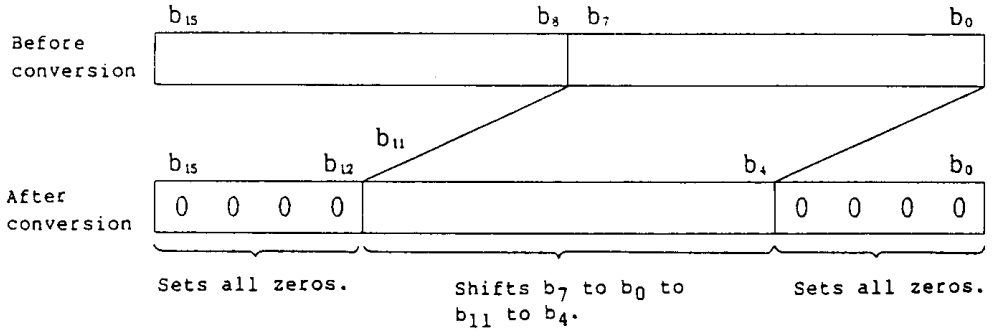
㉒ Output Value Address (Write)

(a) This area sets the address of a word from which the result of a PID operation will be output.

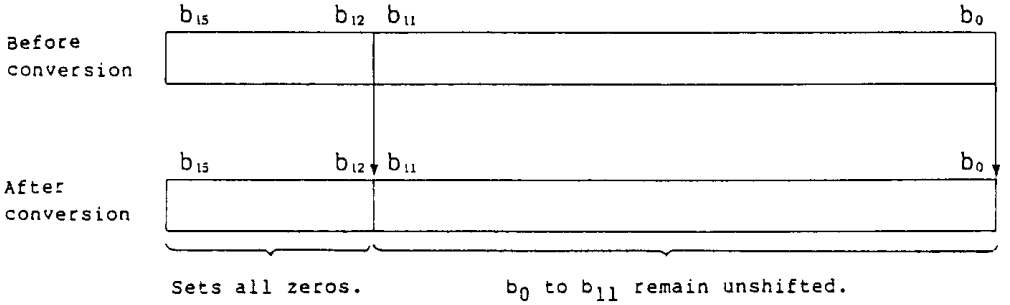
㉓ Set Value Bit Pattern (Write)

This area determines a method of converting a set value into 16-bit data for PID operation. Select and write one of H0001 to H0004 according to the conditions shown below. See Table 3.2 for bits patterns for analog input modules.

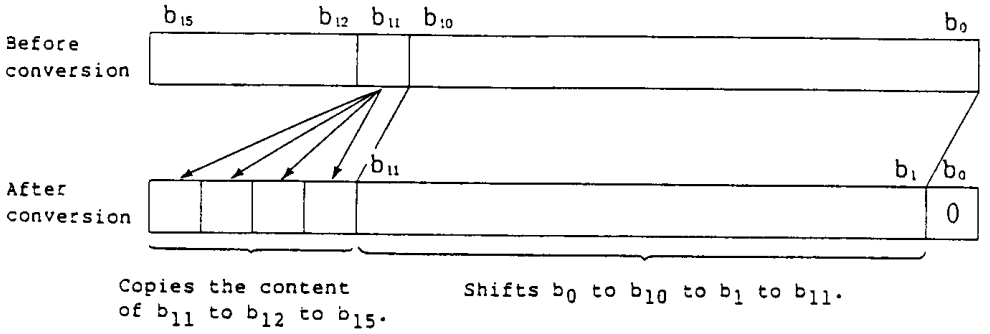
H0001: 8 bits to 16 bits



H0002: Unsigned 12 bits to 16 bits



H0003: Signed 12 bits to 16 bits (Sign expansion)



H0004: No conversion

(24) Measured Value Bit Pattern (Write)

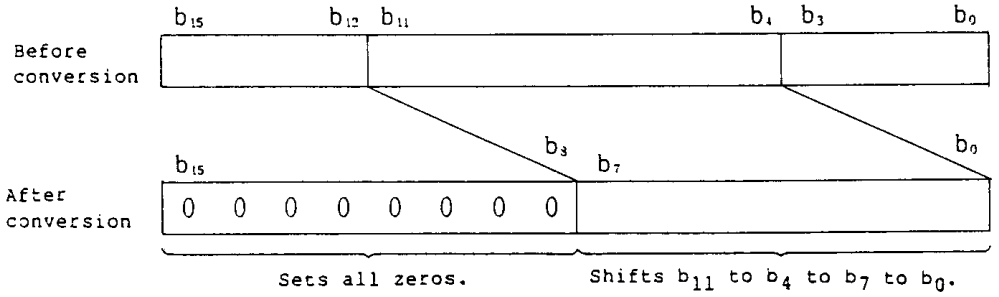
(a) This area sets a method (H0001 to H0004) of converting data read from the Measured Value Address area (21) into 16-bit data. See Table 3.2 for bit patterns for analog input modules. See (23) "Set Value Bit Patterns" for the converting methods.

(25) Output Value Bit Patterns (Write)

(a) The result of FUN2 processing or PID operation is converted according to the bit pattern set in this area and written in the Output Value Address area (22). See Table 3.2 for bit patterns for analog output modules.

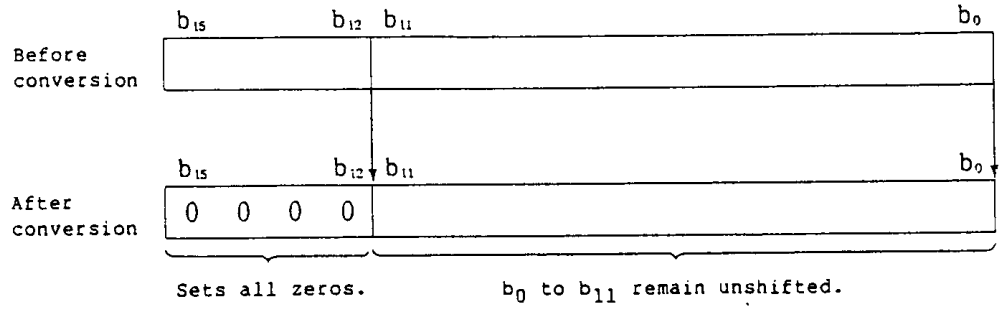
Select and write one of H0001 to H0004 in this area (25) according to the content of the Output Value Address area (Analog Output WY, WR, WL, or WM).

H0001: 18 bits to 8 bits



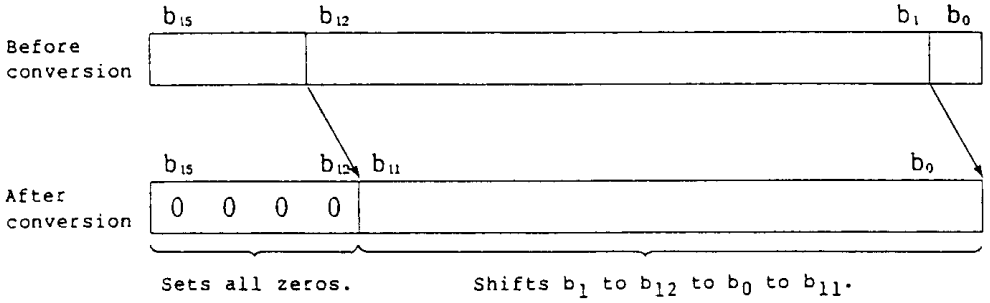
HFFF to H7FFF (before conversion) is converted to HFF.
H8000 to HFFFF (before conversion) is converted to H0.

H0002: 16 bits to Unsigned 12 bits



HFFF to H7FFF (before conversion) is converted to HFF.
H8000 to HFFFF (before conversion) is converted to H0.

H0003: Signed 16 bits to Signed 12 bits



HFFF to H7FFF (before conversion) is converted to H7FF.
H8000 to HFFFF (before conversion) is converted to H80.

H0004: No conversion

Note: The contents of the areas (23) (Set Value Bit Pattern), (24) (Measured Value Bit Pattern), and 25 (Output Value Bit Pattern) vary according to the type of the analog I/O module that reads or writes the values.

Table 3.2 Analog I/O modules and bit patterns

Module	Type	Specification	SV, PV, or MV	Bit pattern
Analog Input Module	XAGV08H	0 - 10 VDC, 8 bits	SV (Set value) and PV (Measured value)	H0001
	XAGC08H	4 - 20 mA DC, 8 bits	"	H0001
	XAGV12H	-10 - +10 VDC, 12 bits	"	H0003
	XAGC12H	4 - 20 mA DC, 12 bits	"	H0002
	XAGV121H	0 - 10 VDC, 12 bits	"	H0002
	XAGV122H	1 - 5 VDC, 12 bits	"	H0002
Analog Output Module	YAGV08H	0 - 10 VDC, 8 bits	MV (Output value)	H0001
	YAGC08H	4 - 20 mA DC, 8 bits	"	H0001
	YAGV12H	-10 - +10 VDC, 12 bits	"	H0003
	YAGC12H	4 - 20 mA DC, 12 bits	"	H0002
	YAGV121H	0 - 10 VDC, 12 bits	"	H0002
	YAGV122H	1 - 5 VDC, 12 bits	"	H0002
RTD Input Module	XRTD01H	-50°C - 400°C, Signed 15 bits	SV (Set value) and PV (Measured value)	H0004

㉖ Work area for PID Operation (Reserved)

(a) The FUN0, FUN1, and FUN2 instructions use this area for PID operations. The user program is not allowed to use this area.

If the user program uses this area, the result of the PID operation will not be assured.

㊿ Execution Flag (Write)

(a) When the Execution flag turns on ("0" to "1"), the PID constants at the time are checked and the PID values are initialized. When the PID constants are all valid, the PID RUN flag ㊽ is turned on ("1").

When any PID constant is found invalid, the PID RUN flag 58 is turned off ("0"). The PID operation will not be performed.

(b) The PID operation is performed while the Execution flag is on ("1").

(c) When the PID RUN flag (58) is turned off ("1"), the PID operation ends and the output becomes 0.

(51) Non-Bumpless Flag (Write)

"Non-bumpless" operation is to store the PID result internally without stopping the PID operation when the S or R flag is turned on. Therefore, when the S or R flag is turned off, the output value abruptly changes according to the intermediate operation value.

Contrarily, in the Bumpless status, the PID operation is suppressed while the S or R flag is on. When the flag is turned on, the output value changes in sequence starting from the value at the time point. See 3.9.3 Timing diagram example 3.

(a) When the Non-Bumpless flag is on, non-bumpless processing is performed.

(b) When the Non-Bumpless flag is off, bumpless processing is performed.

(52) PID Constant Change Flag (Write)

(a) When any PID constant ((12) to (16)) is changed, the PID Constant Change flag must be turned on. When this flag is turned on, the change of PID constants ((12) to (16)) is reflected on the result (previous value).

(b) When the change of the PID constants is completed, this flag must be set to off ("0").

(c) When any PID constant is not valid (when the "PID Constant OK Flag area (60) is "0"), the PID operation is performed with the previous PID constants.

Do not change the content of the flags ((58) to (63)) (for reference only). If changed, the result of the PID operation will not be assured.

(53) S Flag (Write)

(a) When the S flag is on ("1"), the output value is reset to the initial value. The output value is determined according to the relationship of the initial value (19), the the high output limit (17) and low output limit (18), as shown below.

Low output limit > High output limit
No output

Low output limit ≦ Initial value ≦ High output limit
The output value is the initial value (19).

Low output limit ≦ High output limit ≦ Initial value
The output value is the high output limit (17).

Initial value ≦ Low output limit ≦ High output limit
The output value is the low output limit (18).

The S flag is given a higher precedence than the R flag.

(54) R Flag (Write)

(a) When this flag is on ("1"), the output value is cleared to all zeros.

(55) D-FREI Flag (Write)

(a) When this flag is on ("1"), the P, I, and D operations are performed.

(b) When this flag is off ("0"), the P and I operations are performed. (The differential operation (D) is not done.)

(58) PID RUN Flag (Read-only)

(a) When detecting that the Execution flag 50 is on, the FUN1 instruction validity-checks the parameters (12) to (16) and (20) to (22) and sets the result in the PID RUN Flag area (58).

- PID RUN Flag = "1": Valid

  The PID operation starts.

- PID RUN Flag = "0": Not Valid

  The PID operation is suppressed. The error factor is set in the Error Code 1 area (2).

(b) When detecting the on-to-off transition of the Execution flag (50) (while the PID RUN flag 58 is on), the FUN1 instruction turns off the PID RUN flag (58) and ends the PID operation.

(59) PID-In-Execution Flag (Read-only)

(a) The FUN2 instruction turns on this flag of a loop when performing the PID operation on it and turns off the PID-In-Execution flags of the other loops.

(60) PID Constant OK Flag (Read-only)

(a) When detecting that the PID Constant Change flag (52) is on, the FUN1 instruction validity-checks the PID constants (12) to (16) and sets the result in the PID Constant OK flag (60).

- PID Constant OK Flag = "1": The PID constants are valid.
- PID Constant OK Flag = "0": The PID constants are not valid.

  The FUNC1 instruction performs the PID operation with the previous PID constants.

  The error code is set in the Error Code 1 area (2).

(61) Over High Limit Flag MV > UL (Read-only)

(a) When the output value calculated by the FUN2 instruction is greater than the high output limit UL (17), this flag is turned on ("1").

(62) Under Low Limit Flag MV < LL (Read-only)

(a) When the output value calculated by the FUN2 instruction is less than the low output limit LL (12), this flag is turned on ("1").

(63) FUN2 Error Flag (Read-only)

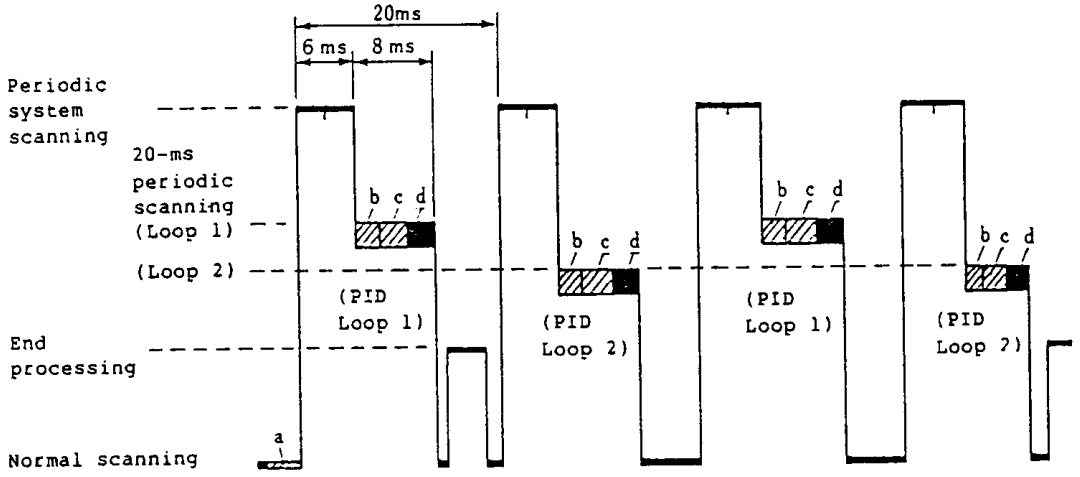
(a) When an error is found in the high output limit (17), the low output limit (18), or bit patterns (23) to (25)), the FUN2 instruction turns on the FUN2 Error flag (63). Its error code is set in the Error Code 2 area (2). The PID operation is not suppressed by this error.

When no error is found in the execution of the FUN2 instruction, the FUN2 Error flag (63) remains off ("0") and the Error Code 2 area (2) remains blank.

Executions of PID Operation Instructions in the CPU

[Example 1] PID control on two loops whose sampling times are respectively 2 (a multiple of 20 ms)

Note: For legibility, this timing diagram combines two 10-ms system interrupts into one.



a: FUN0 processing
b: FUN1 processing
c: FUN2 processing
d: Other periodic interrupt processing

Fig. 3.8.1 Management of execution of PID operation (2 loops)

[Example 2] PID control on the following three loops

Loop 1: TZ = 3 (x 20 ms)
Loop 2: TZ = 6 (x 20 ms)
Loop 1: TZ = 12 (x 20 ms)

Note: For legibility, this timing diagram combines two 10-ms system interrupts into one.

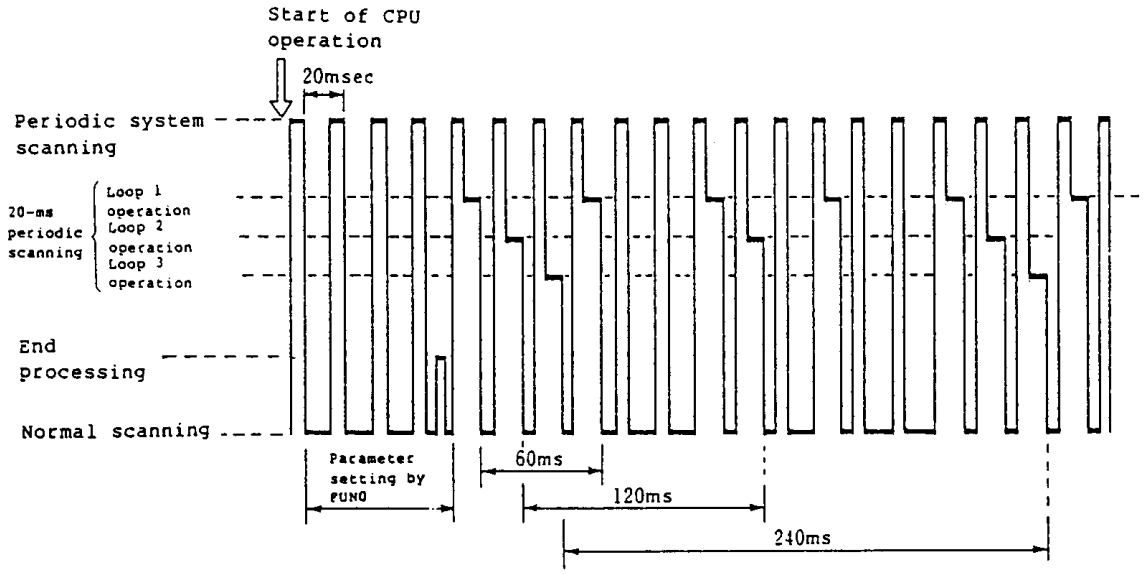
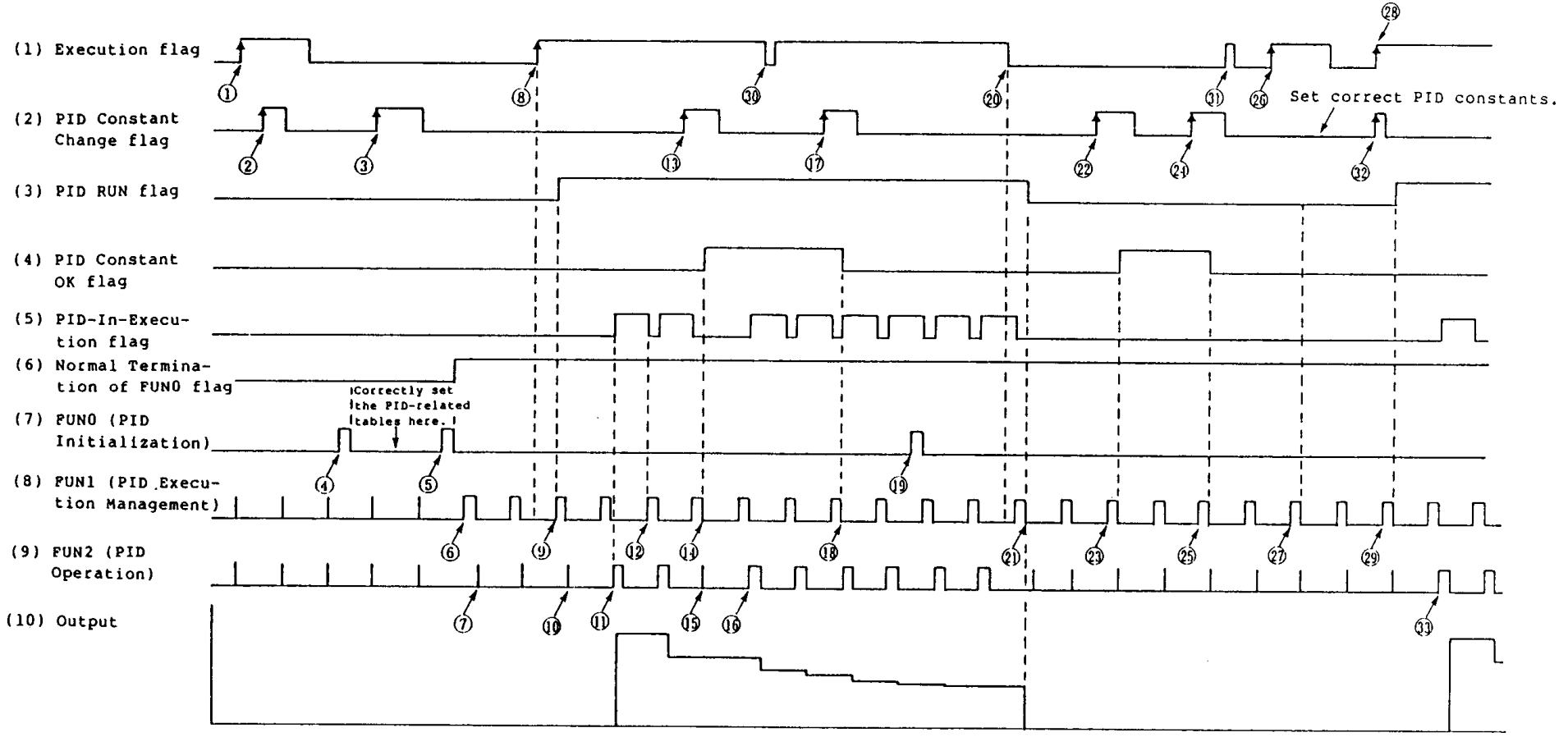


Fig. 3.8.2 Management of execution of PID operation (3 loops)

Timing Diagram

Timing diagram example 1

This example shows waveforms of flags (PID RUN, PID Constant OK, and PID-In-Execution) and instructions (FUN0, FUN1, and FUN2) at the on-to-off transition of both the Execution flag and the PID Constant Change flag for a loop.



[Explanation]

①, ②, ③
These off-to-on transitions are ignored as the FUN0 instruction is not executed normally.

④ As the FUN0 instruction finds an error in the PID-related table, the FUN1 instruction is disabled.

⑤ The FUN0 instruction has ended normally and

⑥ FUN1 processing starts.

⑦ The FUN2 instruction does not perform the PID operation as the Execution flag is off ("0").

⑧ The FUN1 instruction ⑨ detects the off-to-on transition of the Execution flag and validity-checks PID constants. As the PID constants are valid, the PID RUN flag is turned on.

⑩ The PID operation of the FUN2 instruction starts at ⑪ (FUN2) as the first scanning is not executed.

⑪ The PID-In-Execution flag is turned on before the PID operation of the FUN2.

⑫ The FUN1 instruction turns off the PID-In-Execution flag.

⑬ At the off-to-on transition of the PID Constant Change flag,

⑭ The FUN1 turns on the PID Constant OK flag (as the PID constant check passed) and changes the PID constants.

⑮ The FUN2 instruction does not perform the PID operation.

⑯ The PID operation starts with the new PID constants.

⑰ At the off-to-on transition of the PID Constant Change flag,

⑱ The FUN1 turns off the PID Constant OK flag (as the FUN1 instruction detects an error in the PID constants). The PID Constant flag remains unchanged.

⑲ Another FUN0 instruction is ignored when it is executed while the PID operation is executed.

⑳ As the on-to-off transition of the Execution flag is detected by the FUN1 instruction ㉑,

㉑ the PID RUN flag is turned off and the output becomes 0.

(22) As the on-to-off transition of the PID Constant Change flag is detected by the FUN1 instruction (23) while the Execution flag is off, the PID constants are validity-checked. As they are valid, the instruction changes the PID constants and turns on the PID Constant OK flag.

(24) As the on-to-off transition of the PID Constant Change flag is detected by the FUN1 instruction (25) while the Execution flag is off, the PID constants are validity-checked. As they are not valid, the instruction turns on the PID Constant OK flag.

(26) The off-to-on transition of the Execution flag

(27) The FUN1 instruction detects this transition and checks the PID constants. As they have an error, the PID RUN flag is turned off.

(29) The FUN1 instruction detects the simultaneous off-to-on transition of the Execution flag (28) and the PID Constant Change flag (32) . The off-to-on transition of the PID Constant Change flag is ignored.

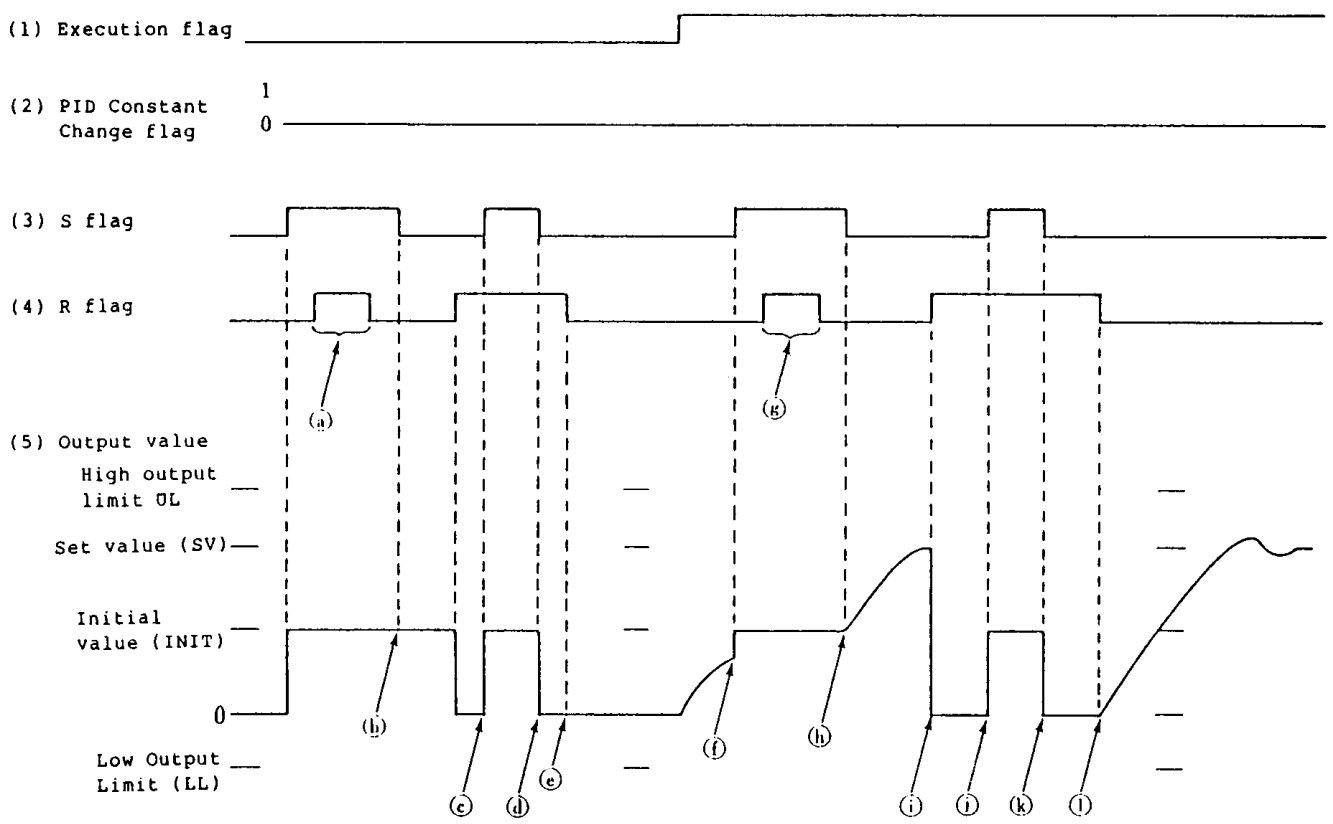
As the PID constants are valid, the FUN1 turns on the PID RUN flag. The PID operation restarts from the FUN2 instruction (33) .

(30) , (31)
The on-to-off transitions of the Execution flag are ignored if they are not detected in normal periodic scanning.

Timing diagram example 2

This example shows timing of the operations of the S and R flags (in Bumpless mode).

S flag: Resets the output value to the initial value.
R flag: Resets the output value to all zeros.



(a) (g) The output value remains the initial value (INIT) as the S flag is preferentially used.

(b) (e) The output value is held as the Execution flag is off.

(c) (j) The output value remains the initial value (INIT) as the S flag is preferentially used.

(d) (k) The output value becomes 0 as the S flag is off and the R flag is on.

(f) The output value becomes the initial value (INIT).

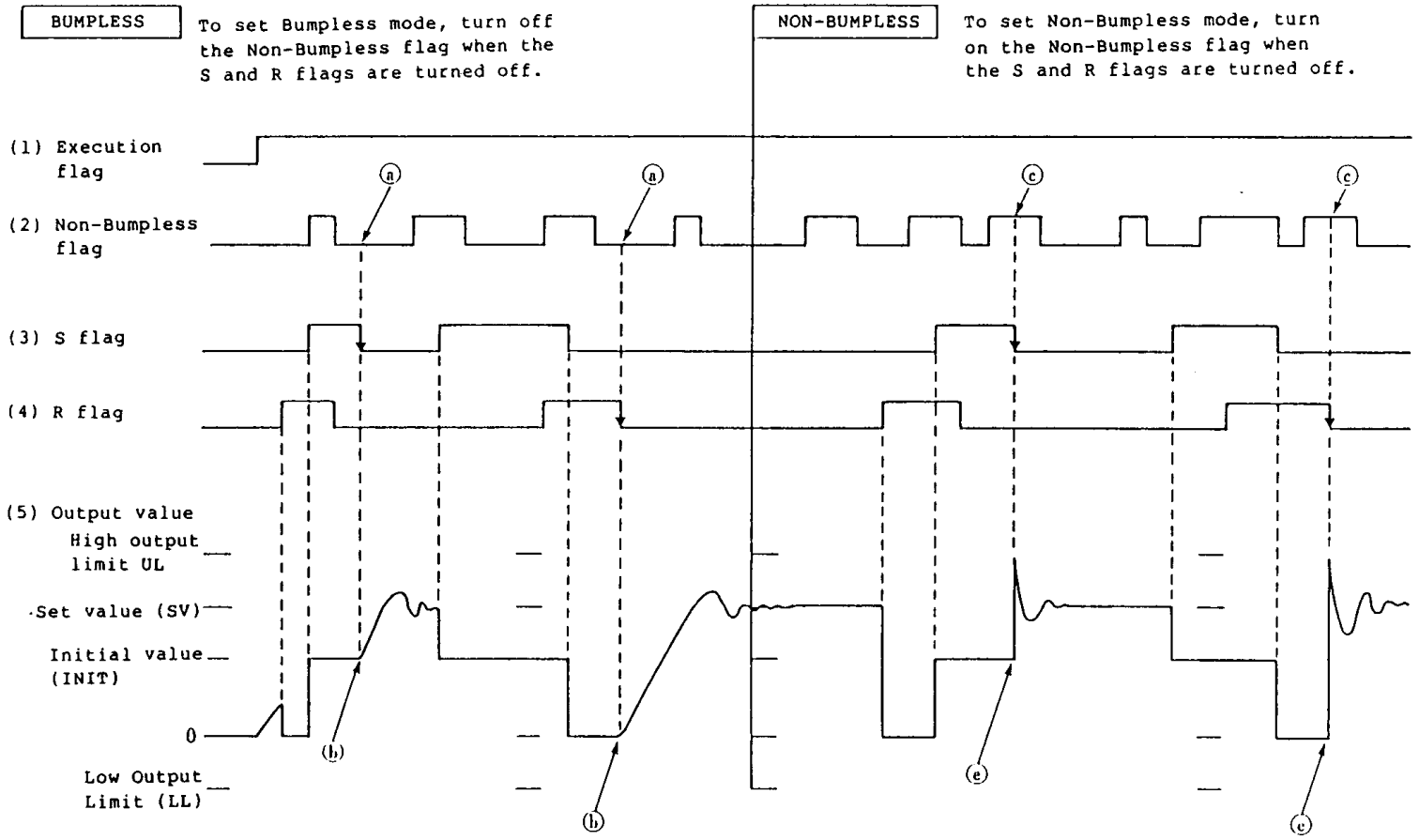
(h) (l) The output value goes toward a target value continuously as the Execution flag is on and the operation is in the Bumpless mode.

(i) The output value becomes 0.

→ The execution of the FUN0 instruction has ended normally.

Timing diagram example 3

This example shows timing of bumpless and non-bumpless operations.

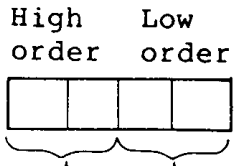


ⓑ When the S flag and the R flag turn off, the output value goes toward a preset value continuously.

ⓔ When the S flag and the R flag turn off, the output value goes toward a preset value abruptly.

Error Codes

The error code is expressed by four hexadecimal digits.



Indicates the cause of the error.

Indicates a loop number.
H00: Error not related to a loop number
H01 to H40: Error occurred on a loop of the indicated loop number

The loop number of H01 to H40 is set in the high-order "xx" of error codes in the error code list.

The circled numbers in this table are those used in the PID management table (3.7.1), and word and bit tables for loops (3.7.2).

Error codes set in the Error Code 0 area

The Error Code 0 area sets the code of an error which is detected in the execution of the FUN0 or FUN1 (partially) instruction. When no error is detected in the execution of the instruction, this area holds the previous status.

Error Code 0 Area (1 of 3)			
Error code	Error cause	Action to repair	Remarks
0001	Although a FUN0 instruction has ended normally, but another FUN0 instruction is issued for execution.	Do not try to execute an FUN0 instruction again after it ended normally.	The Normal Termination of FUN0 area ⑤ holds the previous value.
0002	The loop number is 0.	Specify a value of 1 to 64 as the number of loops (in the Number of Loops area ③).	
0003	The loop number is greater than 64	Specify a value of 1 to 64 as the number of loops (in the Number of Loops area ③).	
0004	The PID management table exceeds the I/O limit.	Correct the address of the top of the PID management table or the number of loops so that the PID management table is within the I/O limit.	The size of the PID management table is variable. When the number of loops ③ exceeds the I/O limit, the Normal termination of FUN0 area ⑤ holds the previous value.
xx05	The word table of loop xx exceeds the I/O I/O limit.	Specify the correct address of the word table (WR) for the loop.	The word table is 48 words long.
xx06	The bit table of loop xx exceeds the I/O limit.	Specify the correct address of the bit table (R) for the loop.	The bit table is 16 bits long.

Error Code 0 Area (2 of 3)			
Error code	Error cause	Action to repair	Remarks
xx07	The high output value (17) of loop xx is invalid.	The high output value (17) should be in the range of -32767 to 32767.	
xx08	The low output value (18) of loop xx is invalid.	The low output value (18) should be in the range of -32767 to 32767.	
xx09	The initial value (19) of loop xx is invalid.	The initial value (19) should be in the range of -32767 to 32767.	
xx0A	The high output limit (17), the low output limit (18), and the initial value (19) for loop xx are related improperly.	Correct the high output limit (17), the low output limit (18), and the initial value (19) for loop so that they may be related normally. Low output limit ≦ Initial value ≦ High output limit	
xx0B	The set value bit pattern (23) for loop xx is not valid.	Specify a value ranging from 1 to 4 in the SV Bit Pattern area (23).	
xx0C	The measured value bit pattern (24) for loop xx is not valid.	Specify a value ranging from 1 to 4 in the PV Bit Pattern area (24).	
xx0D	The output value bit pattern (25) for loop xx is not valid.	Specify a value ranging from 1 to 4 in the MV Bit Pattern area (25).	
0020 (Note)	FUN1 was executed although FUN0 did not end normally.	Program so that FUN1 may be executed after FUN0 ends normally.	This error code is set in the Error Code 0 area specified by "S" of FUN1(S).

Error Code 0 Area (3 of 3)			
Error code	Error cause	Action to repair	Remarks
0021 (Note)	The content of "S" of FUN1(S) is not equal to "S" of FUN0(S) (with respect to the PID management table).	Specify a correct WR address so that the content of "S" of FUN1(S) may be equal to "S" of FUN0(S).	This error code is set in the Error Code 0 area specified by "S" of FUN1(S).

Note: Error codes 0020 and 0021 are written over an error code (0001 to xx0D, if any) which was already set in the Error Code 0 area. Make sure that FUN0 has been executed normally before FUN1 is executed. See Chapter 4 for programming examples.

Error codes set in the Error Code 1 area

The Error Code 0 area sets the code of an error which is detected in the execution of the FUN1 instruction. When no error is detected in the execution of the instruction, this area holds the previous status.

Error Code 1 Area (1 of 2)			
Error code	Error cause	Action to repair	Remarks
0020	FUN1 was executed although FUN0 did not end normally.	Program so that FUN1 may be executed after FUN0 ends normally.	This error code is set in the Error Code 0 area specified by "S" of FUN1(S).
0021	The content of "S" of FUN1(S) is not equal to "S" of FUN0(S) (with respect to the PID management table).	Specify a correct WR address so that the content of "S" of FUN1(S) may be equal to "S" of FUN0(S).	This error code is set in the Error Code 0 area specified by "S" of FUN1(S).
xx22	The set value address (20) for loop xx is not valid.	Specify a correct set value address (20) by the ADRIO instruction.	This error can occur at the off-to-on transition of the Execution flag.
xx23	The measured value address (21) for loop xx is not valid.	Specify a correct measured value address (21) by the ADRIO instruction.	
xx24	The output value address (22) for loop xx is not valid.	Specify a correct output value address (22) by the ADRIO instruction.	
xx25	The sampling time of loop xx is no valid.	Specify a correct sampling time (1 to 200).	These errors can occur at the off-to-on transition of the Execution flag or the PID Constant Change flag.
xx26	The sampling time (12) of loop xx is not a multiple of the number of loops (3).	The sampling time (12) should be a multiple of the number of loops (3).	

Error Code 1 Area (2 of 2)			
Error code	Error cause	Action to repair	Remarks
xx27	The proportional gain (13) of loop xx is not valid.	Specify a correct proportional gain (-1000 to 1000).	
xx28	The integral constant (14) of loop xx is not valid.	Specify a correct integral constant (1 to 32767).	
xx29	The differential constant (15) of loop xx is not valid.	Specify a correct differential constant (1 to 32767).	
xx2A	The differential delay constant (16) of loop xx is not valid.	Specify a correct differential delay constant (1 to 32767).	
xx30	The low output limit (18) of loop xx is greater than the high output limit (17) of loop xx.	The low output limit (18) of loop xx must be equal to or less than the high output limit (17) of loop xx.	This error can occur when the S flag (53) turns on while the PID RUN flag (58) is off.
xx31	The output value address (22) of loop xx is not valid.	Specify a correct output value address (22) by the ADRIO instruction.	This error can occur when the S flag (53) or R flag (54) turns on while the PID RUN flag (58) is off.
xx32	The output value bit pattern (25) of loop xx is not valid.	Specify a correct output value bit pattern (25) (1 to 4).	

Error codes set in the Error Code 2 area

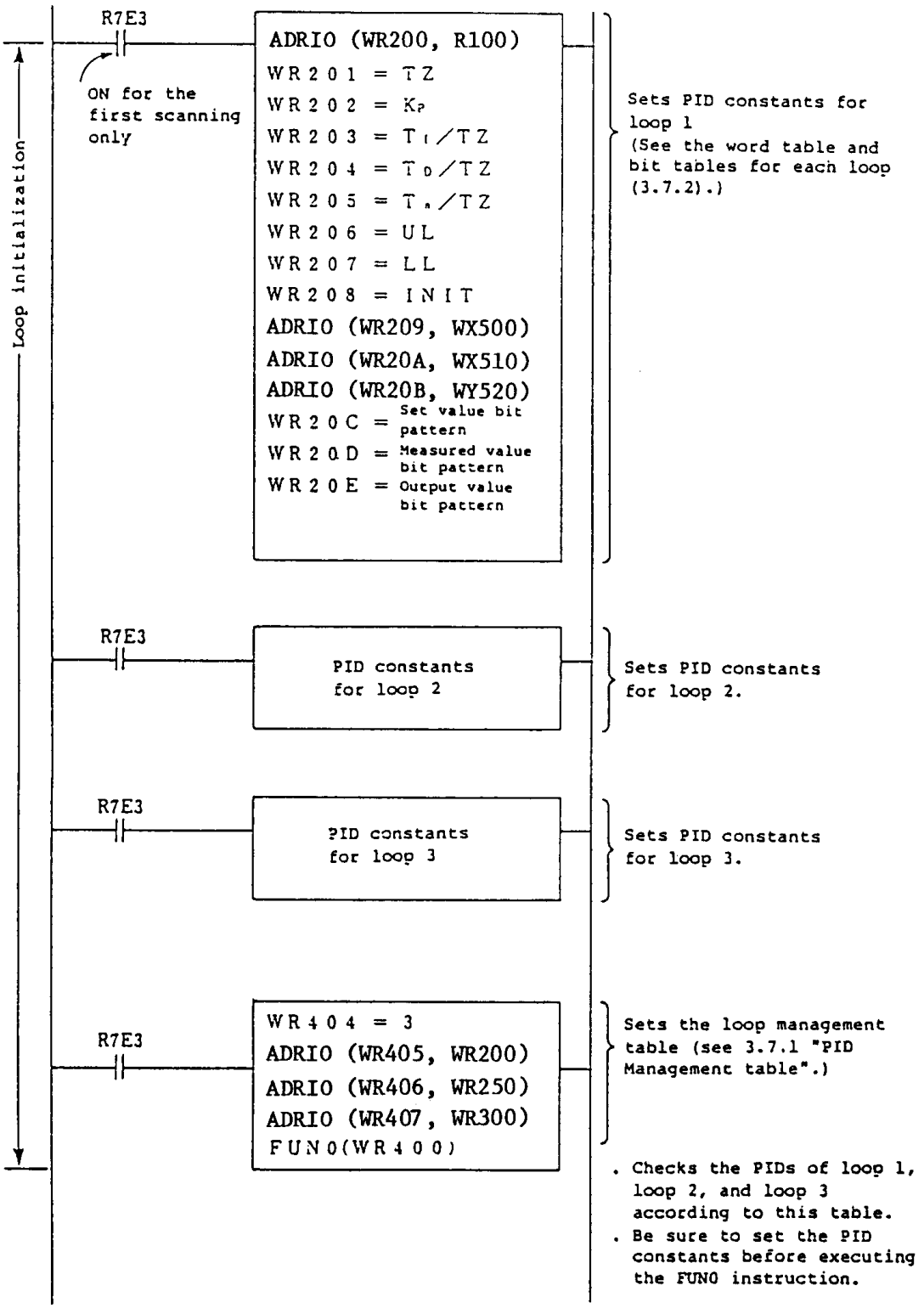
The Error Code 2 area sets the code of an error which is detected in the execution of the FUN2 instruction. When no error is detected in the execution of the instruction, this area holds the previous status.

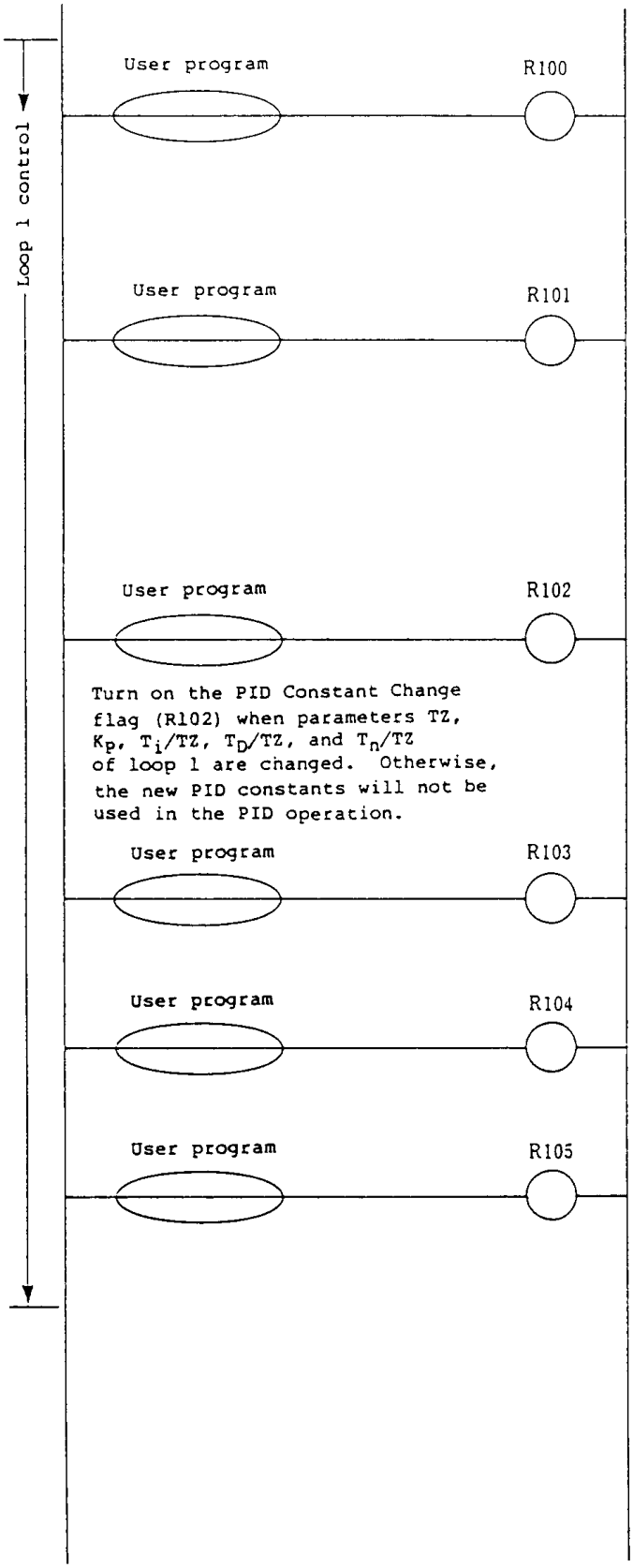
Error Code 2 Area			
Error code	Error cause	Action to repair	Remarks
xx41	The set value bit pattern (23) for loop xx is not valid.	Specify a value ranging from 1 to 4 in the SV Bit Pattern area (23).	When the specified bit pattern is not valid, the FUN2 instruction continues processing with "4 NO CONVERSION".
xx42	The measured value bit pattern (24) for loop xx is not valid.	Specify a value ranging from 1 to 4 in the PV Bit Pattern area (24).	
xx43	The output value bit pattern (25) for loop xx is not valid.	Specify a value ranging from 1 to 4 in the MV Bit Pattern area (25).	
xx44	The low output limit (18) of loop xx is greater than the high output limit (17) of loop xx.	The low output limit (18) of loop xx must be equal to or less than the high output limit (17) of loop xx.	When this error occurs, FUN2 processing continues but the result is not output.

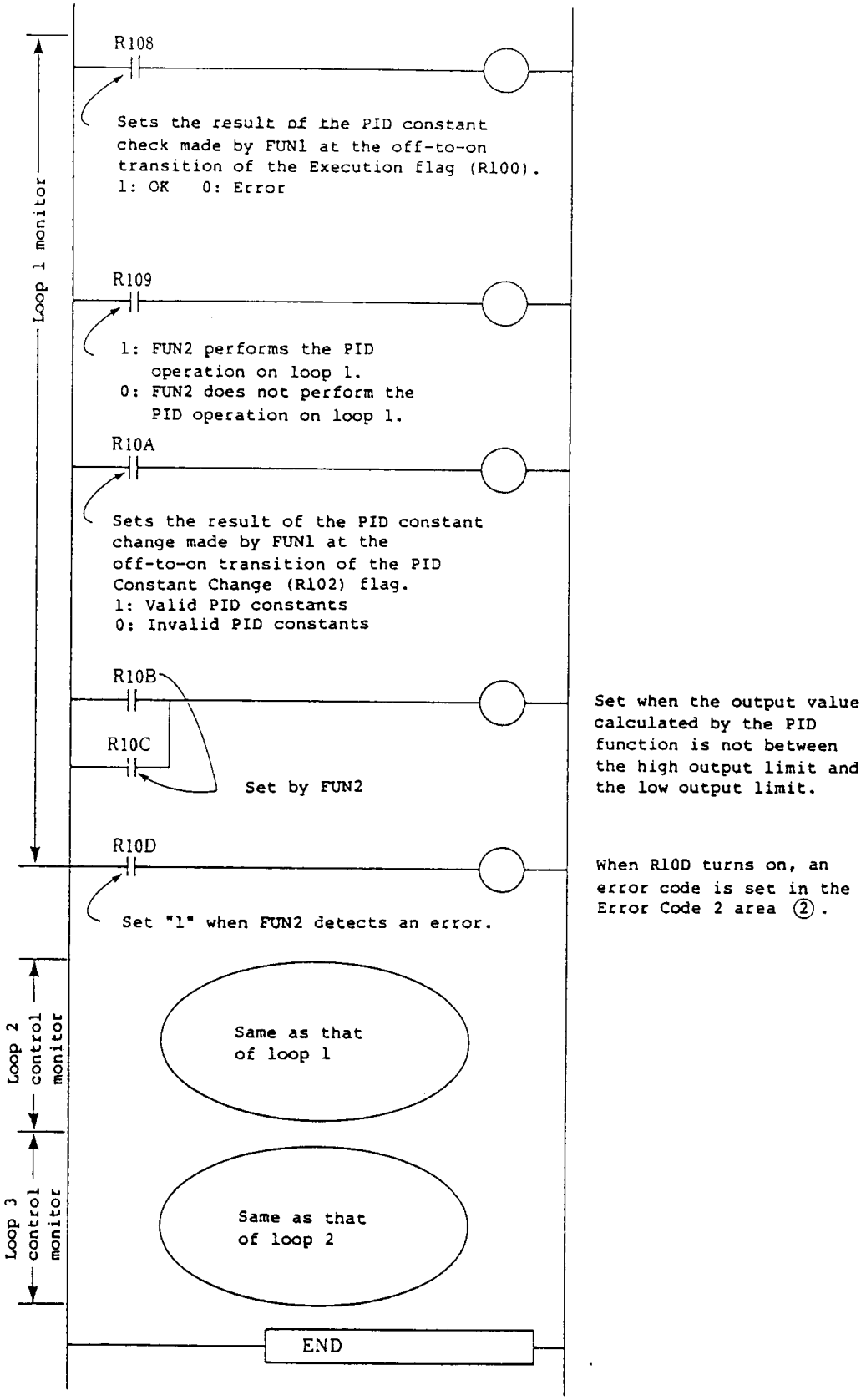
Programming Examples

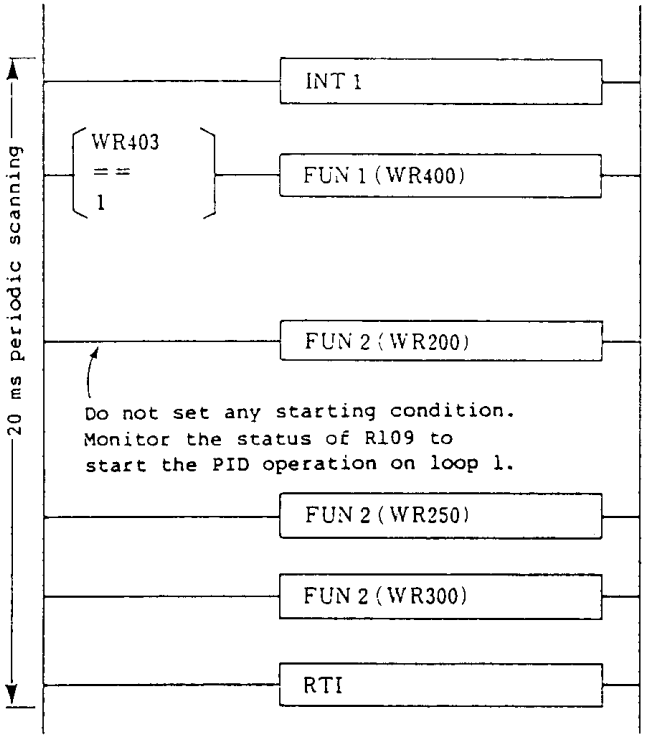
Programming Example 1

This programming example is to make the CPU module to control three process loops. In this programming example, the PID constants are changed each time the CPU starts to run. To store the forcibly-changed PID constants, see Section 4.2.


Loop initialization
R7E3
ON for the first scanning only
ADRIO (WR200, R100)
WR201 = TZ
WR202 = KP
WR203 = TI/TZ
WR204 = TD/TZ
WR205 = TS/TZ
WR206 = UL
WR207 = LL
WR208 = INIT
ADRIO (WR209, WX500)
ADRIO (WR20A, WX510)
ADRIO (WR20B, WY520)
WR20C = Set value bit pattern
WR20D = Measured value bit pattern
WR20E = Output value bit pattern
Sets PID constants for loop 1 (See the word table and bit tables for each loop (3.7.2).)
R7E3
PID constants for loop 2
Sets PID constants for loop 2.
R7E3
PID constants for loop 3
Sets PID constants for loop 3.
R7E3
WR404 = 3
ADRIO (WR405, WR200)
ADRIO (WR406, WR250)
ADRIO (WR407, WR300)
FUN0(WR400)
Sets the loop management table (see 3.7.1 "PID Management table".)
. Checks the PIDs of loop 1, loop 2, and loop 3 according to this table.
. Be sure to set the PID constants before executing the FUN0 instruction.


Loop 1 control
User program
R100
Execution flag
1: Executes the PID operation on loop 1.
0: Ends the PID operation on loop 1 and resets the output value to 0.
User program
R101
Non-Bumpless flag
1: Non-bumpless
0: Bumpless
User program
R102
PID Constant Change flag
1: At the off-to-on transition of this flag, the operation to change PID constants starts.
Turn on the PID Constant Change flag (R102) when parameters TZ, Kp, Ti/TZ, TD/TZ, and Tn/TZ of loop 1 are changed. Otherwise, the new PID constants will not be used in the PID operation.
User program
R103
S flag
1: The output value is an initial value 19 .
The S flag is give higher precedence than the R flag.
User program
R104
R flag
1: The output value is 0.
User program
R105
D-FREI FLAG
1: Performs the PID operation.
0: Performs the PI operation.


R108
Sets the result of the PID constant check made by FUN1 at the off-to-on transition of the Execution flag (R100).
1: OK 0: Error
R109
1: FUN2 performs the PID operation on loop 1.
0: FUN2 does not perform the PID operation on loop 1.
R10A
Sets the result of the PID constant change made by FUN1 at the off-to-on transition of the PID Constant Change (R102) flag.
1: Valid PID constants
0: Invalid PID constants
R10B
R10C
Set by FUN2
Set when the output value calculated by the PID function is not between the high output limit and the low output limit.
R10D
When R10D turns on, an error code is set in the Error Code 2 area ②.
Set "1" when FUN2 detects an error.
Loop 1 monitor
Loop 2 control monitor
Same as that of loop 1
Loop 3 control monitor
Same as that of loop 2
END



Start of 20 ms periodic scanning

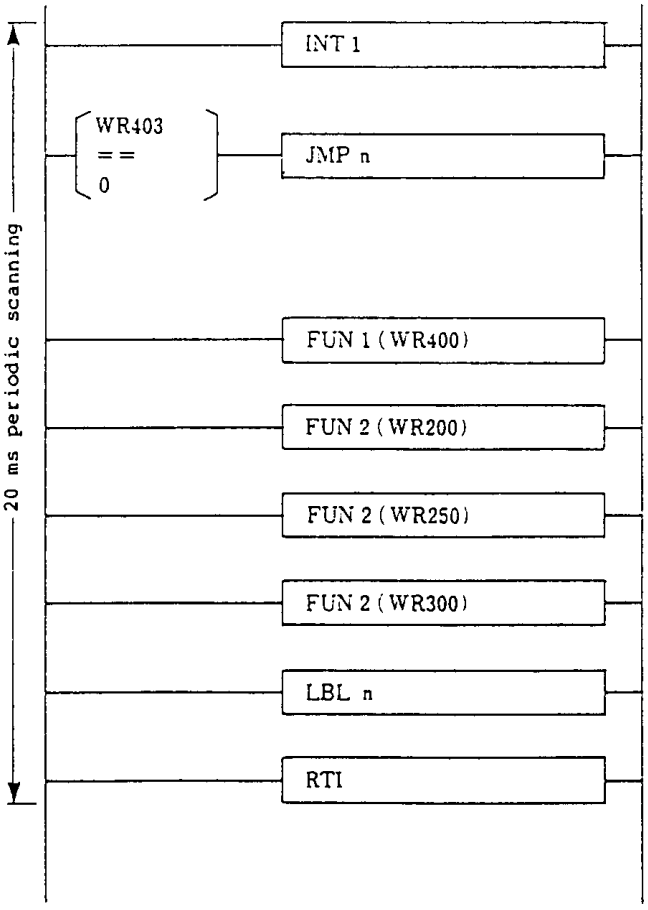
Code so that FUN1 (WR400) is executed after the execution of FUN0 ends normally (when WR403 = "1"). With this coding, the Error Code 0 area holds the error code when FUN0 detects an error.

Performs the PID operation on loop 1.

Performs the PID operation on loop 2.

Performs the PID operation on loop 3.

The above programming for 20 ms-periodic scanning can be replaced by the following:



Start of 20 ms periodic scanning

Jump to LBLn when the execution of FUN0 abends. When the execution of FUN0 ends normally, the succeeding instruction is executed. With this coding, the Error Code 0 area holds the error code when FUN0 detects an error.

Determines the loop number to be executed by this periodic scanning.

Performs the PID operation on loop 1.

Performs the PID operation on loop 2.

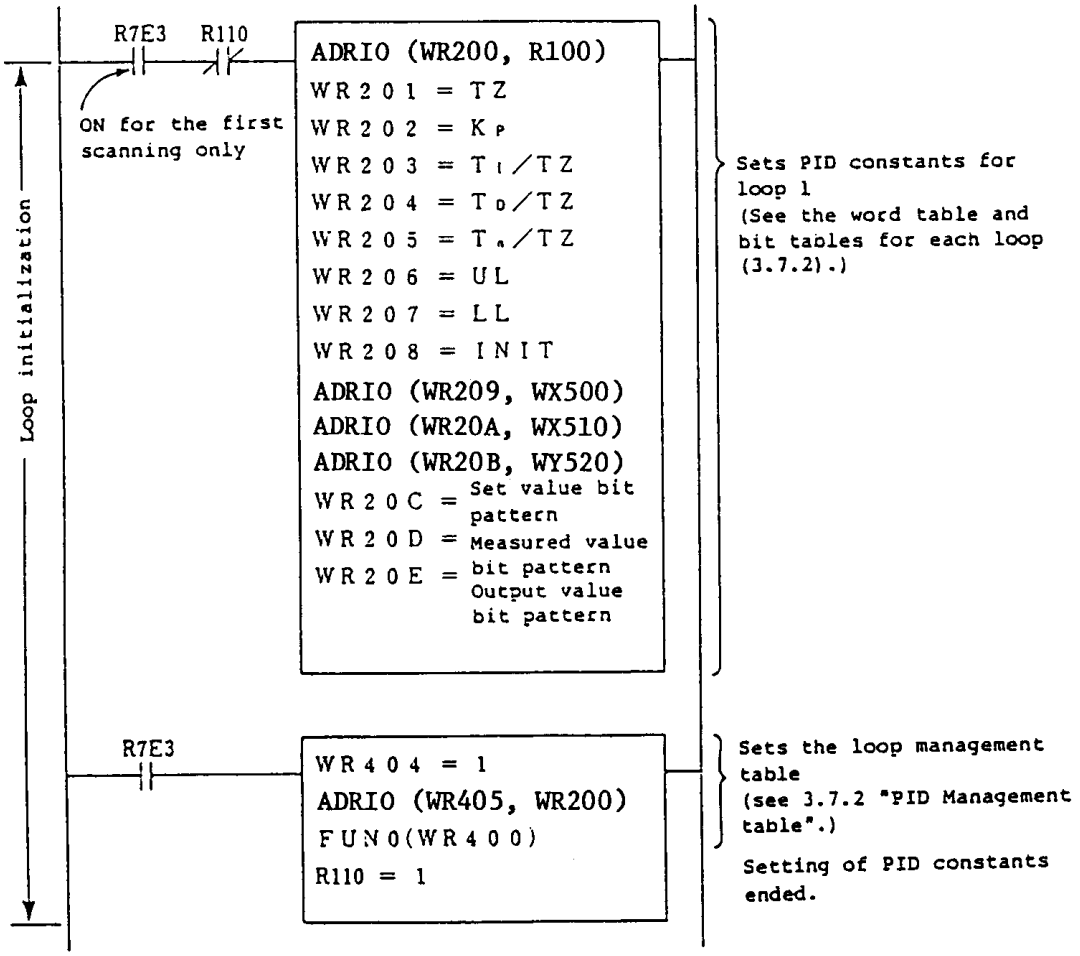
Performs the PID operation on loop 3.

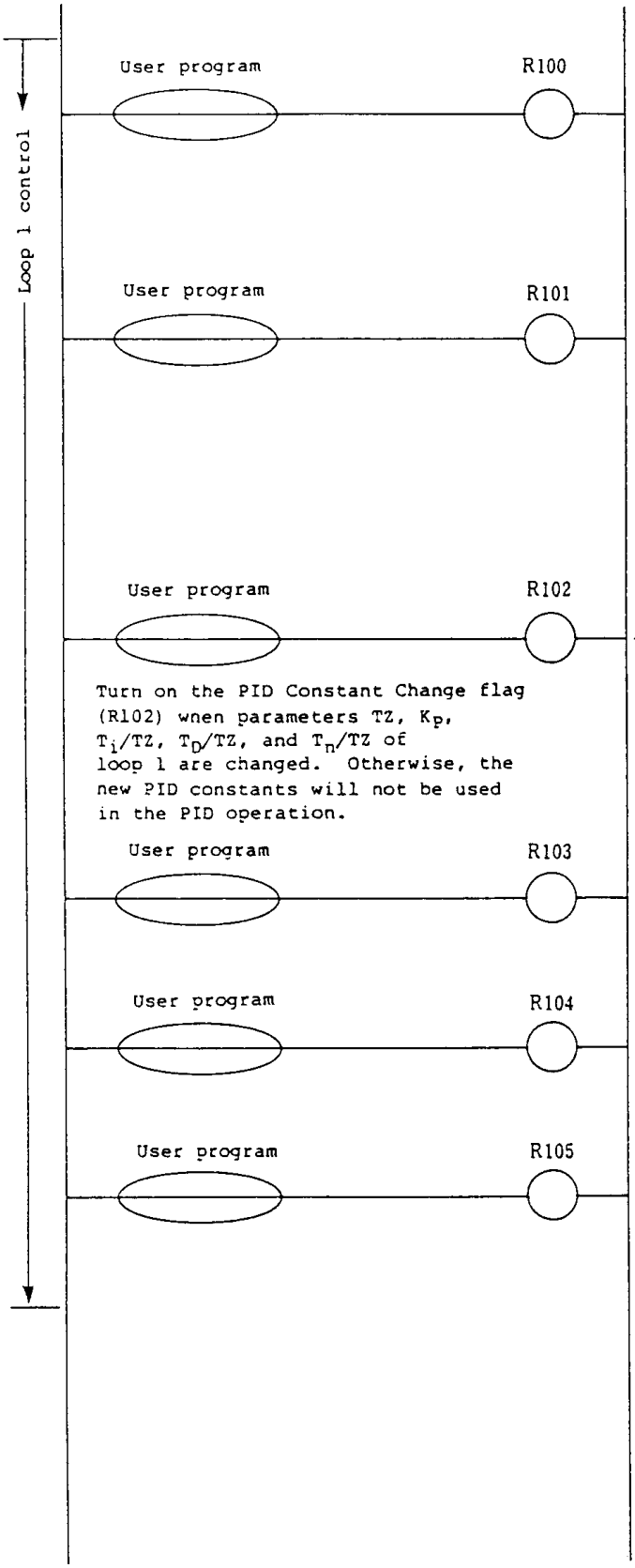
Control jumps here when the execution of FUN0 abends.

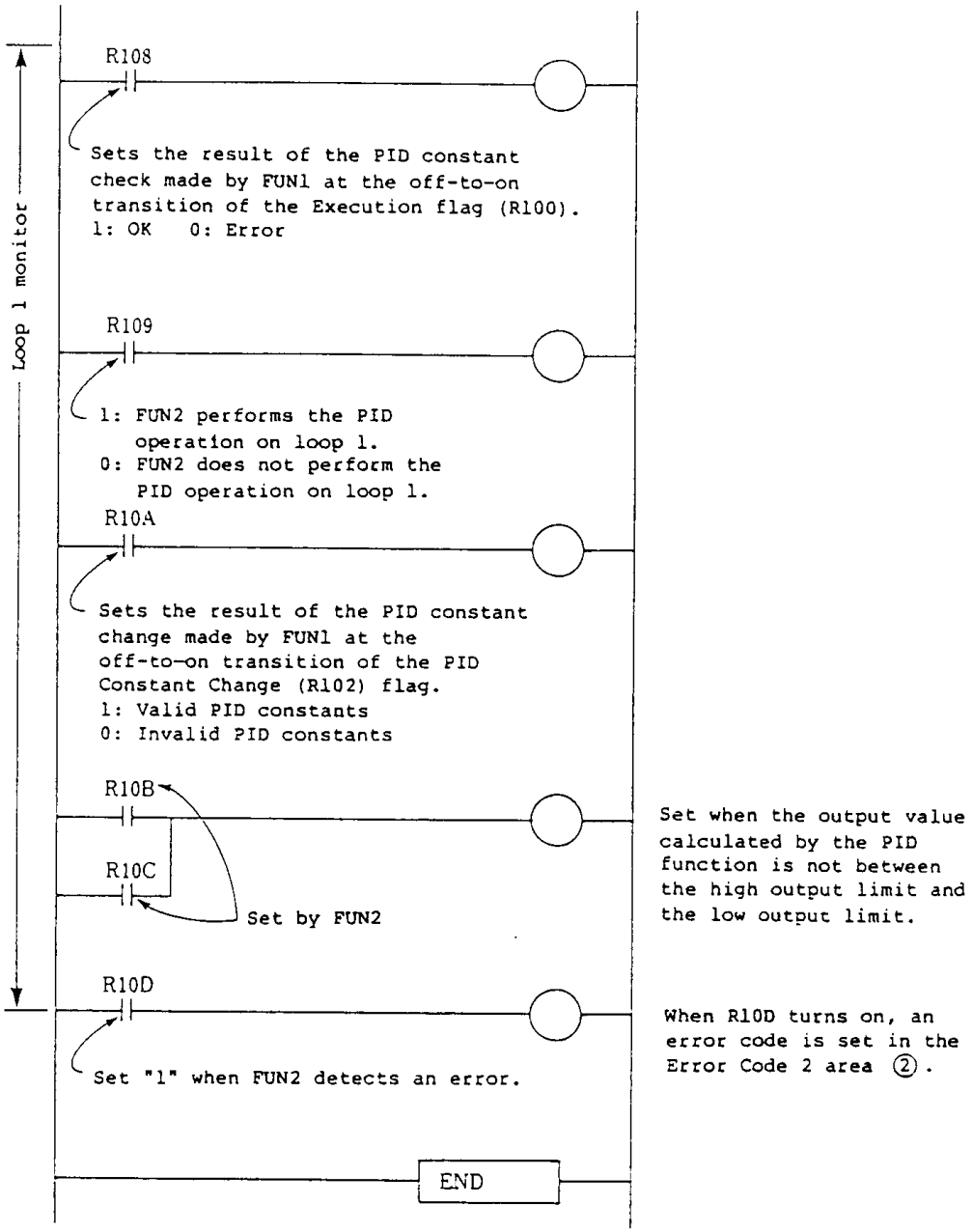
Programming Example 2

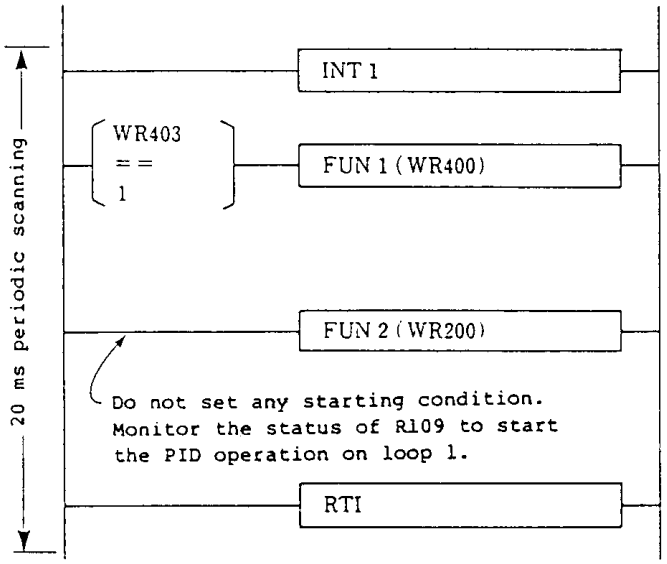
This programming example is to make the CPU module to control a single process loops. In this programming example, the PID constants are written only once. The PID constants changed in the execution of an instruction are held until they are forcibly changed.

For this purpose, "Power-off protection" must be specified for the PID management table (WR400 to WR405), the word table for loop 1 (WR200 to WR22F), and R110 (PID Constant Setting Ended).




Loop 1 control
User program
R100
Execution flag
1: Executes the PID operation on loop 1.
0: Ends the PID operation on loop 1 and resets the output value to 0.
User program
R101
Non-Bumpless flag
1: Non-bumpless
0: Bumpless
User program
R102
PID Constant Change flag
↑: At the off-to-on transition of this flag, the operation to change PID constants starts.
Turn on the PID Constant Change flag (R102) wnen parameters TZ, Kp, Ti/TZ, TD/TZ, and Tn/TZ of loop 1 are changed. Otherwise, the new PID constants will not be used in the PID operation.
User program
R103
S flag
1: The output value is an initial value (19).
The S flag is give higher precedence than the R flag.
User program
R104
R flag
1: The output value is 0.
User program
R105
D-FREI FLAG
1: Performs the PID operation.
0: Performs the PI operation.


Loop 1 monitor
R108
Sets the result of the PID constant check made by FUN1 at the off-to-on transition of the Execution flag (R100).
1: OK 0: Error
R109
1: FUN2 performs the PID operation on loop 1.
0: FUN2 does not perform the PID operation on loop 1.
R10A
Sets the result of the PID constant change made by FUN1 at the off-to-on transition of the PID Constant Change (R102) flag.
1: Valid PID constants
0: Invalid PID constants
R10B
R10C
Set by FUN2
Set when the output value calculated by the PID function is not between the high output limit and the low output limit.
R10D
Set "1" when FUN2 detects an error.
When R10D turns on, an error code is set in the Error Code 2 area ②.
END

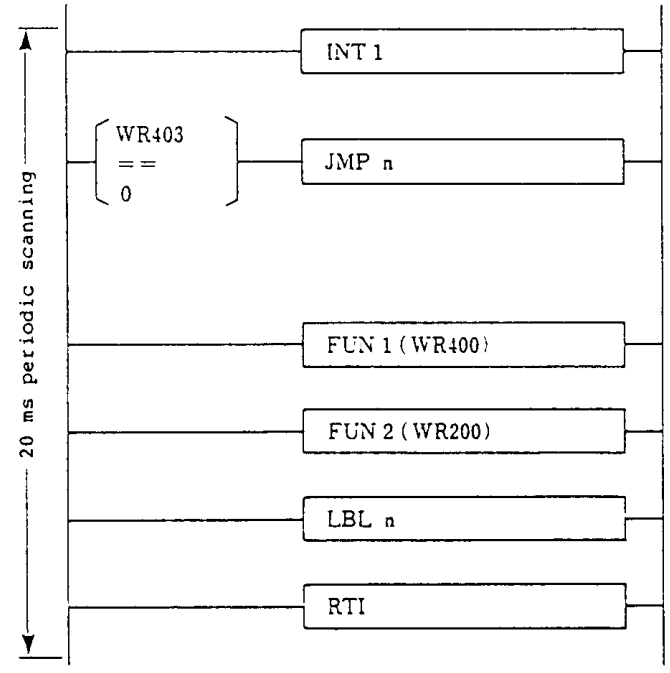


Start of 20 ms periodic scanning

Code so that FUN1 (WR400) is executed after the execution of FUN0 ends normally (when WR403 = "1"). With this coding, the Error Code 0 area holds the error code when FUN0 detects an error.

Performs the PID operation on loop 1.

The above programming for 20 ms-periodic scanning can be replaced by the following:



Start of 20 ms periodic scanning

Jump to LBLn when the execution of FUN0 abends. When the execution of FUN0 ends normally, the succeeding instruction is executed. With this coding, the Error Code 0 area holds the error code when FUN0 detects an error.

Determines the loop number to be executed by this periodic scanning.

Performs the PID operation on loop 1.

Control jumps here when the execution of FUN0 abends.