TSL3000 series Robot Controller

TSL3000 SCARA system
TSL3100 6AXIS system

INSTRUCTION MANUAL

SIMPLE PLC FUNCTION MANUAL

Notice

- Make sure that this instruction manual is delivered to the final user of Toshiba Machine's industrial robot.
- Before operating the industrial robot, read through and completely understand this manual.
- After reading through this manual, keep it nearby for future reference.

TOSHIBA MACHINE CO., LTD.

Preface

Structure of TH series robot and TSL3000 controller instruction manuals:
These instruction manuals were published in parts according to the application and purpose, and the name and outline of each manual are as follows:

[Safety Manual]

This manual contains the important information to use the robot safety and correctly. Be sure to read through and understand this manual before operating the robot. Also, strictly observe the descriptions made there.

[Operator's Manual]

This manual deals with the TSL3000 controller operating procedures. Read through this manual before operating the robot, and refer to it as necessary.

[Robot Language Manual]

This manual refers to the robot language called "SCOL". When you have to create a program based on this language, read through the manual.

[Interface Manual]

This manual describes the external signals for the robot. Concerning the interface conditions between the robot and peripheral equipment, specifications, timing, etc., refer to the manual when necessary.

[Installation & Transport Manual]

This manual describes the transport, unpacking and installation of the robot and controller. Be sure to read through this manual before unpacking the shipment containing the robot.

[Maintenance Manual]

This manual deals with the daily and regular inspections to be made on the robot and controller. Read through this manual to use the robot safely over long years to come.

[Communication Manual]

This manual describes the serial communication between the robot controller and other equipment. Refer to this manual when connecting the robot controller with a host computer, optical sensor, etc., via a serial cable.

[User Parameter Manual]

This manual describes the setting of the robot controller. Read this manual when performing the setting of communication, I/O, motion condition, etc.

[Alarm Manual]

This manual describes the alarms, their causes, and remedies. Refer to this manual when an alarm occurs.

Cautions on Safety

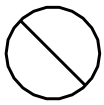
This manual contains the important information on the robot and controller to prevent injury to the operators and persons nearby, to prevent damage to assets and to assure correct use.
Make sure that the following details (indications and symbols) are well understood before reading this manual. Always observe the information that is noted.

[Explanation of indications]

Indication	Meaning of indication
 **DANGER**	This means that "incorrect handling will lead to fatalities or serious injuries."
 **CAUTION**	This means that "incorrect handling may lead to personal injuries *1) or physical damage *2).

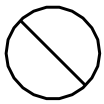
*1) Injuries refer to injuries, burns and electric shocks, etc., which do not require hospitalization or long-term medical treatment.
*2) Physical damage refers to damages due to destruction of assets or resources.

[Explanation of symbols]

Symbol	Meaning of symbol
	This means that the action is prohibited (must not be done). Details of the actions actually prohibited are indicated with pictures or words in or near the symbol.
	This means that the action is mandatory (must be done). Details of the actions that must be done are indicated with pictures or words in or near the symbol.
	This means danger. Details of the actual danger are indicated with pictures or words in or near the symbol.
	This means caution. Details of the actual caution are indicated with pictures or words in or near the symbol.

[Operation]

DANGER	
Prohibited	• During operation, NEVER enter the dangerous area of the robot. Otherwise, you will be injured seriously. • DO NOT leave in the working range any machinery or materials which will hinder the operation. If the equipment went wrong, a person nearby will be injured or involved in an accident. • Anyone other than the operator MUST NOT approach the equipment. Should he negligently touch the dangerous part of the equipment, he will get injured or involved in a serious accident. • NEVER perform an inappropriate operation which is not described in the instruction manual. Otherwise, the equipment will start by mistake, resulting in personal injury or serious accident.
Danger	• If you feel even a little that you are exposed to danger or that the equipment works abnormally, press the EMERGENCY stop pushbutton switch to stop the equipment. If the equipment is used as it is, you will be injured or involved in a serious accident. When this happens, ask our after-sale service agent for repair. • During operation, be sure to close the equipment cover. Should the cover be opened during operation, you will be struck by an electric shock or get injured. • Only a well-trained and qualified person is allowed to perform the operation. Should the equipment be operated improperly, it will start by mistake, causing a personal injury or serious accident. • If the equipment has malfunctioned, turn the power off, identify and remove the cause of the abnormality, maintain the peripheral equipment and completely restore the malfunctioned equipment. Then start the equipment at a slow speed. If the equipment starts, leaving the abnormality, you will be involved in a serious accident.

CAUTION	
 Prohibited	• DO NOT change the data of the system structure file. Otherwise, the robot will move abnormally, resulting in damage or an accident.
 Mandatory	• In principle, teaching operation should be performed outside the dangerous area of the robot. If it should be performed inevitably within the dangerous area, strictly observe the following matters. (1) The teaching operation should always be performed by two (2) persons. One person performs the job and the other person watches outside the dangerous area. Also, both persons should try to prevent mis-operation with each other. (2) The operator should do the job in an attitude ready to press the EMERGENCY stop pushbutton switch at any time. Also, he should perform the job at a position from which he can evacuate immediately at the time of an emergency after confirming the robot working range and shields nearby. (3) The supervisor should keep watch on the job at a position where he can see the entire robot system and operate the EMERGENCY stop pushbutton switch at the time of an emergency. Also, he should keep anyone from entering the dangerous area. Unless the operator or other person follows the instructions of the supervisor, an accident will be caused. • If an abnormality has generated or the POWER LED lamp on the control panel remains off after the main power switch of the equipment was turned on, turn off the main power immediately and confirm the wiring. Otherwise, you will be struck by an electric shock or a fire will break out. • Unless the robot operates toward a designated direction at manual guide, turn off the servo power. Otherwise, the robot will be damaged or you will be involved in an accident. When this happens, call us at the after-sale service agent. • Pushbutton operations on the control panel and teach pendant should be confirmed visually. Otherwise, you will be involved in an accident due to mis-operation. • After the power is turned on, be sure to reset a program to start an automatic operation. If the program is executed continuously, the robot will interfere with the peripheral equipment, resulting in damages or accidents.

CAUTION	
 Mandatory	• Before operating the equipment, perform the following inspection. (1) Make sure that visual appearance of the robot, controller, peripheral equipment and cables are in good condition. (2) Make sure that no obstacle stands in or near the working range of the robot and peripheral equipment. (3) Make sure that the emergency stop and other safety devices operate properly. (4) Make sure that no abnormal noise or vibration is involved in the robot operation. If the above prior inspection is skipped, the equipment will be damaged or you will be involved in an accident.
 Caution	• The speed of test operations is initially set at 25% of the maximum robot speed. • The speed of automatic operation is initially set at 100% of the maximum robot speed.

Table of Contents

Page

Section 1 General Descriptions

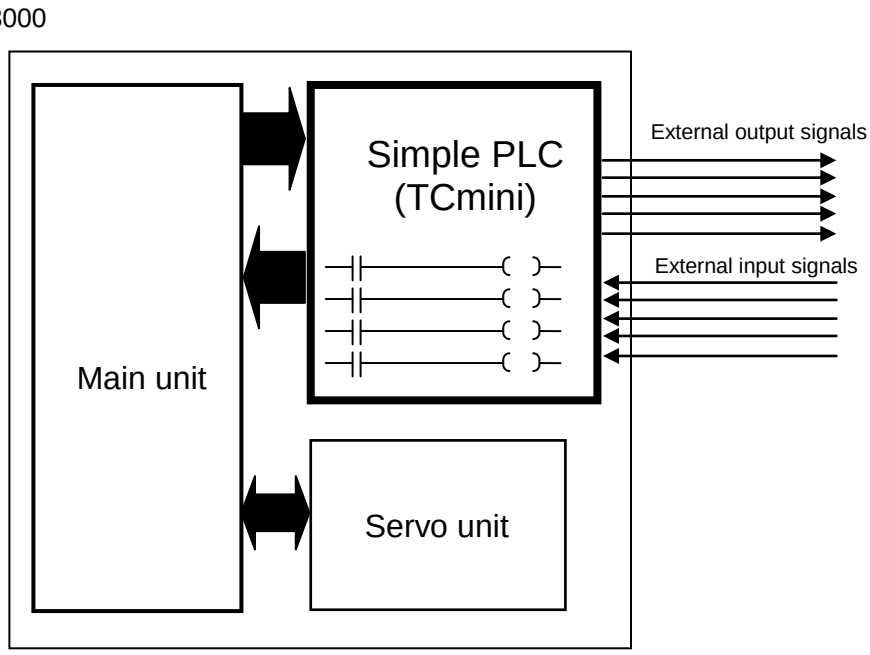
Robot controller TSL3000 incorporates a simple programmable controller (hereinafter called the "TCmini").

Programs (DOUT commands) analyzed by the main unit and system output signals are once sent to the TCmini where they are processed and output to an external device. The TCmini also receives external input signals and transmits them to the main unit after processing. In other words, the TCmini can unrestrictedly determine the locations where input/output processing of robot programs by the main unit and system signals are to be input or output.

Also, the TCmini can directly respond to (or output) input signals sent from the external device. As the TCmini operates while the TSL3000 controller power is turned on, it is capable of performing I/O control, irrespective of robot program execution.

External input/output signals described in the Interface Manual refer to the operation of a standard sequence program. When you create this sequence program, you can design a more flexible system.

This manual deals with the TCmini.



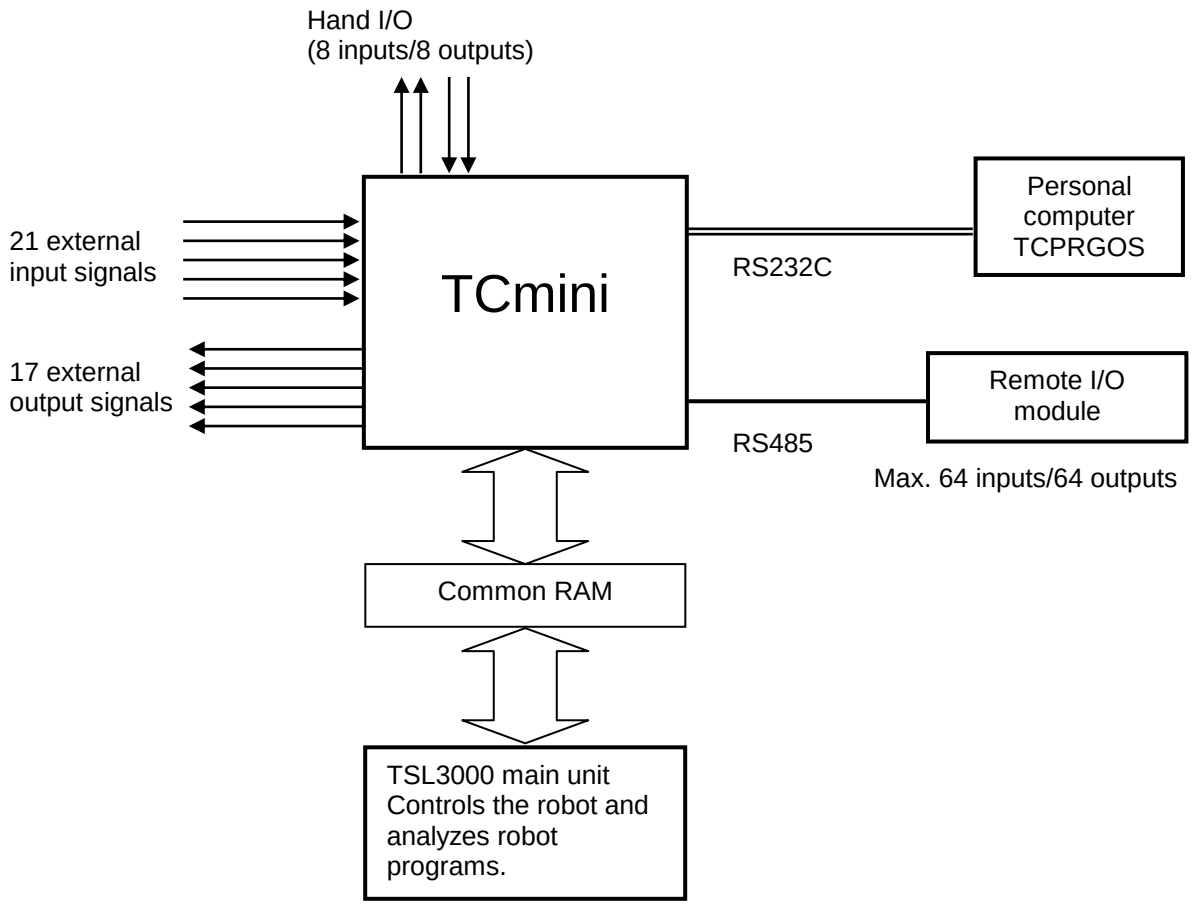
Section 2 TCmini

The **TCmini** built in robot controller TSL3000 is a simple programmable controller resided in the robot, which can control external I/Os (21 inputs and 17 outputs), hand I/Os (8 inputs and 8 outputs) and extension I/Os (64 inputs and 64 outputs).

The TCmini is connected with the TSL3000 main unit via common RAM to transfer I/O data and status.

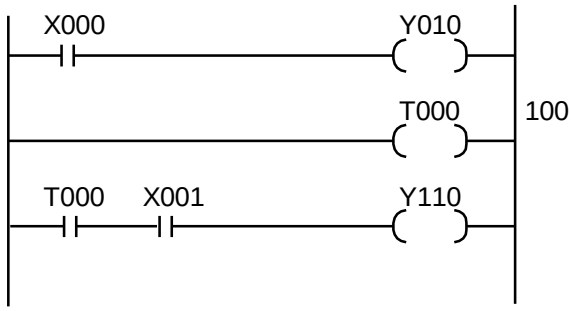
When this TCmini is connected with a remote I/O module via RS485 communication cable (EXTI/O), up to 64 each of inputs and outputs can be connected.

By connecting the TCPRG port (RS232C) on the TSL3000 front side with your personal computer, you can create, debug (I/O monitor) and transfer sequence programs, using **TCPRGOS**.



2.1 Sequence Program

Sequence programs of the TCmini use graphical (or ladder) programming.



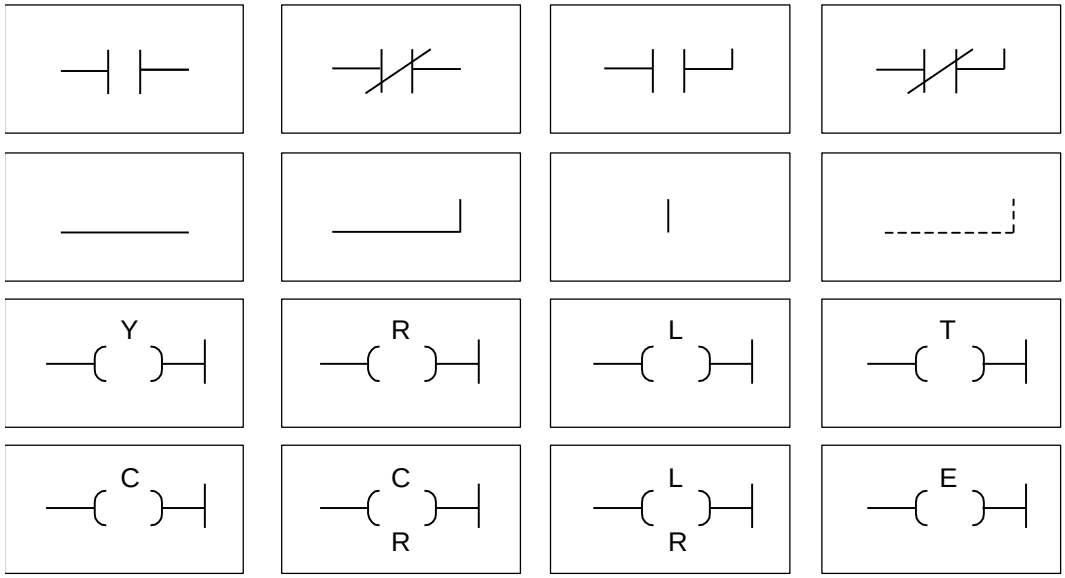
I/O processing in the TCmini is batch-refreshed. Before arithmetic operation of a sequence program, ON/OFF status of inputs is transferred to the data memory with results of preceding sequence operation transferred to outputs.
After the I/O processing, the sequence program is scanned sequentially from the top.

In the TCmini, scanning is executed by the CPU. If the program size increases, it takes more time to scan, resulting in delay of the input and output. Yardstick of the scanning time is about ten (10) msec per 1,000 basic commands (1 K words).

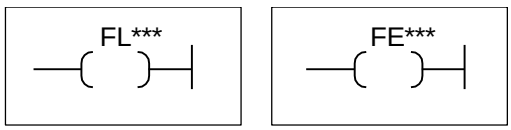
For details on the programming, see Section 7 of this manual.

2.2 Instruction Words

The TCmini can use the following sixteen (16) instruction words.



In the TCmini, thirty-two (32) application instructions (or commands) can be used.



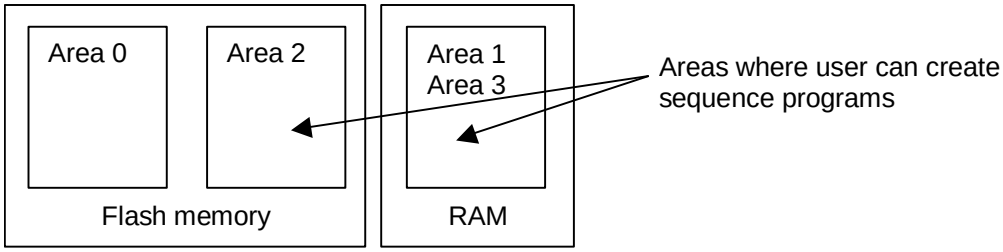
The application instruction comes with the following types.

- Transfer instruction
- Data conversion instruction
- BIN operation instruction
- Comparison instruction
- Bit operation instruction
- Subroutine instruction
- Pulse instruction

* For further information on the instructions, see Section 7 of this manual.

2.3 System Sequence and User Sequence

The PLC (TCmini) built in the TSL3000 has the following three (3) working areas and the sequence program capacity of each working area is 4 K words. The user can create sequence programs in areas 1 and 2 alone.



Area 0: Standard sequence area (Cannot be changed.)
Area 1: User's RAM area for debugging (RAM contents are cleared by power OFF and ON.)
Area 2: User's flash memory area (The number of writing counts is limited to 50.)
Area 3: User's RAM area same as Area 1. Due to battery backup, however, no sequence program will be cleared by power OFF and ON.

* The factory-setting is area 0, and the standard sequence is operating.
* The flash memory contents will not be cleared by power OFF and ON, but limitation is imposed on the writing counts (50 counts).
  Completely debug a sequence program in area 1 (RAM), then transfer it to area 2 (flash memory).

2.4 Change of Sequence Program Area

Changeover of working areas is performed by the user parameter.
The sequence to be operated is changed over by changing the set value of [U11] I/O mode of the USER.PAR file to 0, 1, 2 or 3.

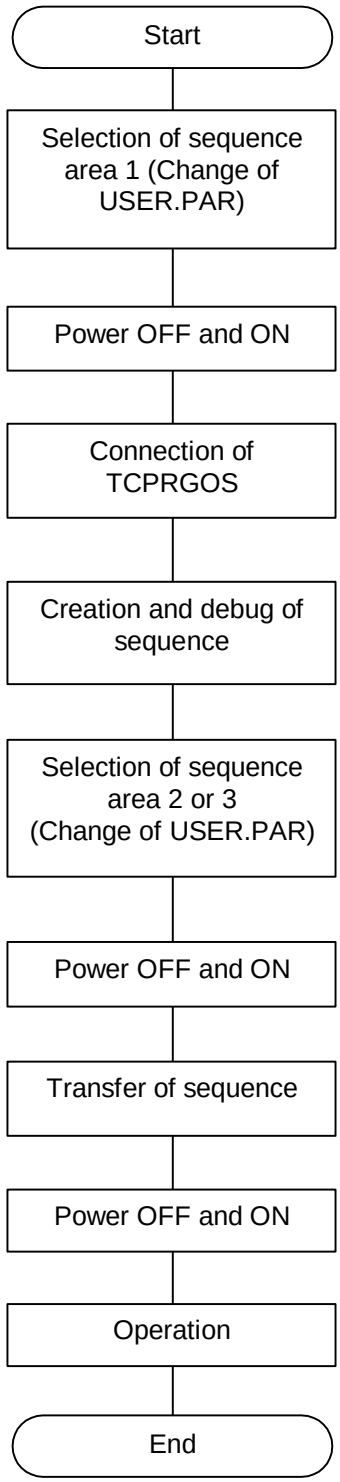
[U11] I/O mode (Setting of I/O operation mode)

[U11] I/O mode {Default/User} (0: Default, 1: User RAM, 2: User FLASH 3: User backup RAM) = 0

In the TSL3000, I/Os specified in the program are processed for input or output.
The storage location of this sequence program can be specified as shown below.

Set value	=	0	Standard sequence (Cannot be changed.)
	=	1	User's RAM for debugging (User's created sequence area)
	=	2	User's flash memory (User's created sequence area)
	=	3	User's RAM for debugging (battery backup) (User's created sequence area)

2.5 Procedures for Creating Sequence



See Para. 2.4.

Selected area becomes valid.

See Section 8.

See Para. 2.4.

Selected area becomes valid.

See Section 8.

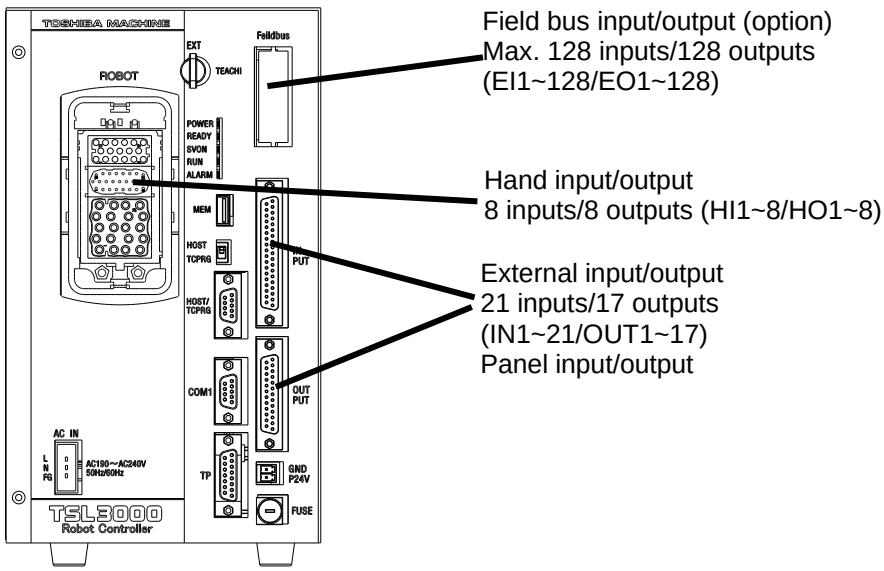
Transferred sequence becomes valid.

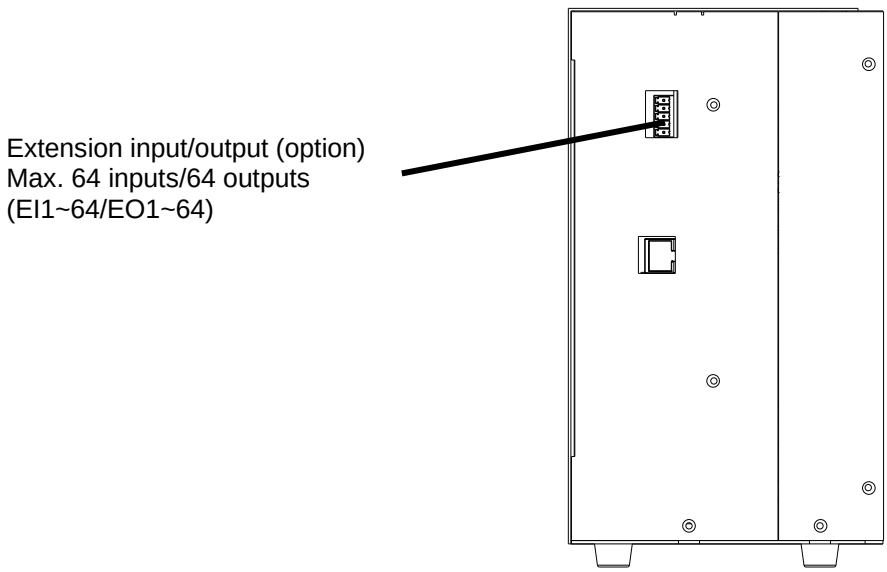
Section 3 Inputs and Outputs

3.1 Outline of Inputs and Outputs

The I/O common for the TSL3000 robot controller comes in the two (2) types; **Type N** which is compatible with that of the previous controller (SR7000) and **Type P** which is heteropolar. Either type can be selected at order entry. The extension I/O module is also provided with the two (2) types; TR48DIOCN (Type N) and TR48DIOC (Type P). Identify the type of your controller before reading this manual.

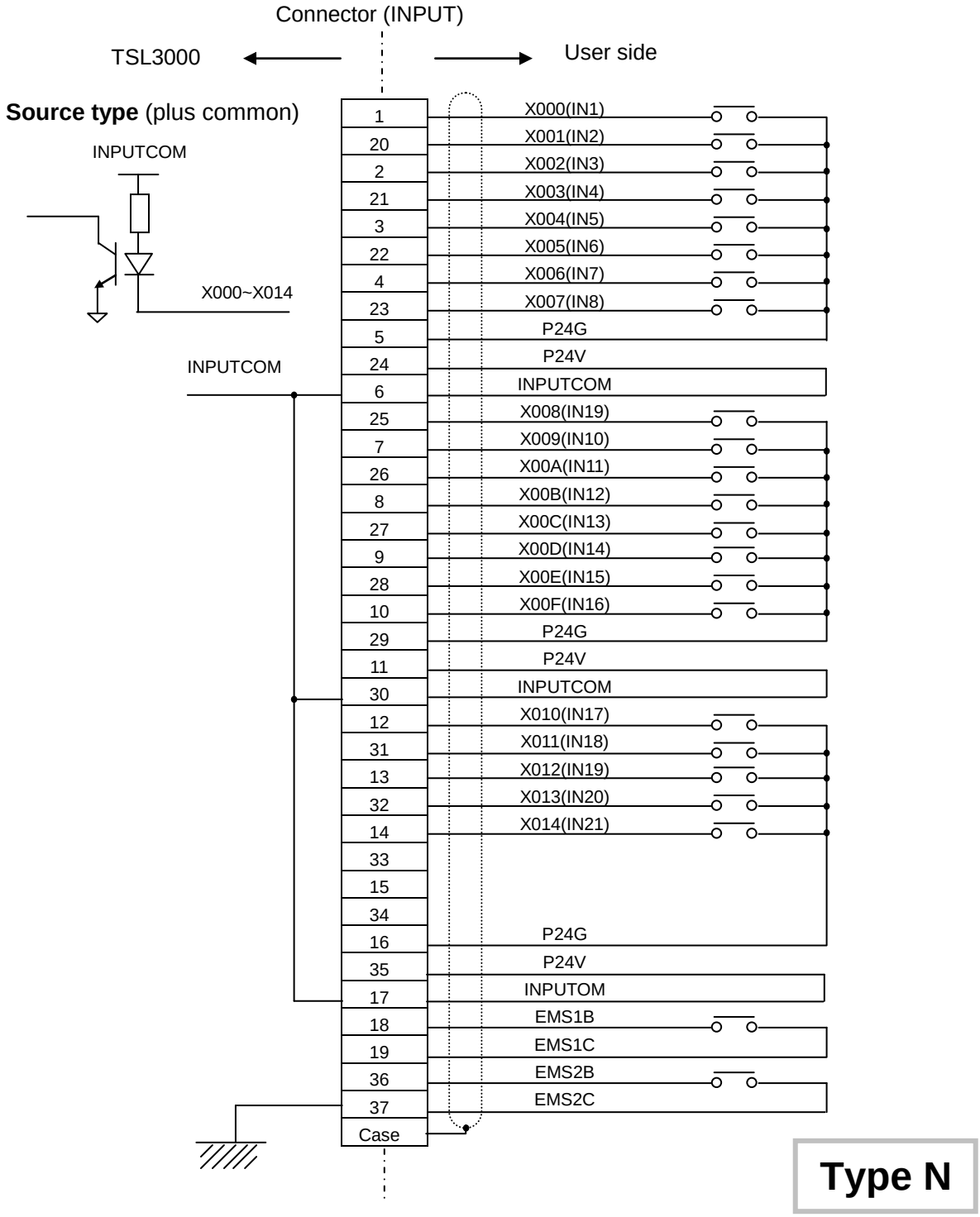
In the TSL3000, the input/output signals where sequence control is possible by the TCmini use the input and output ports shown below.

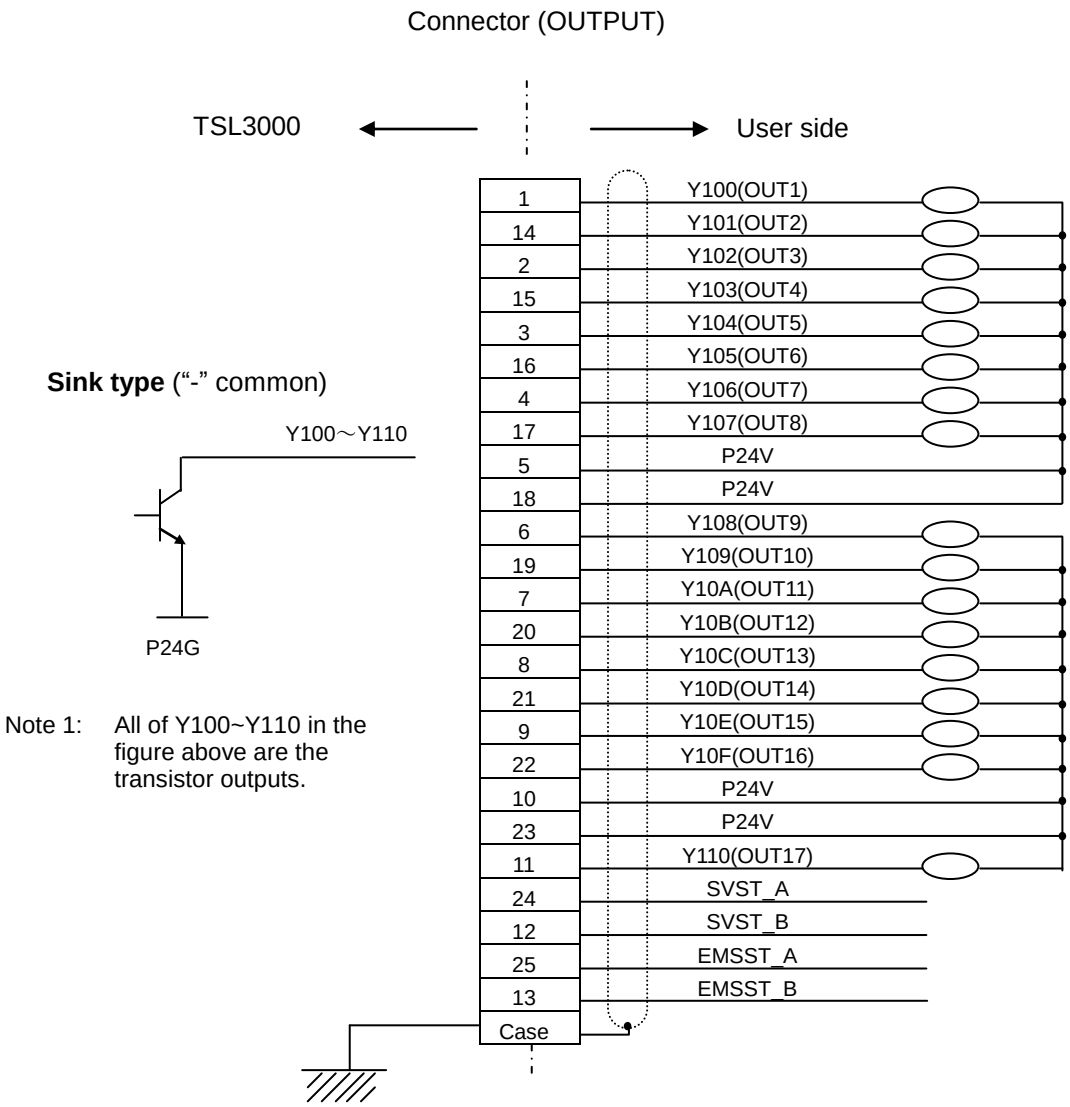




3.2 External Input and Output (Type N)

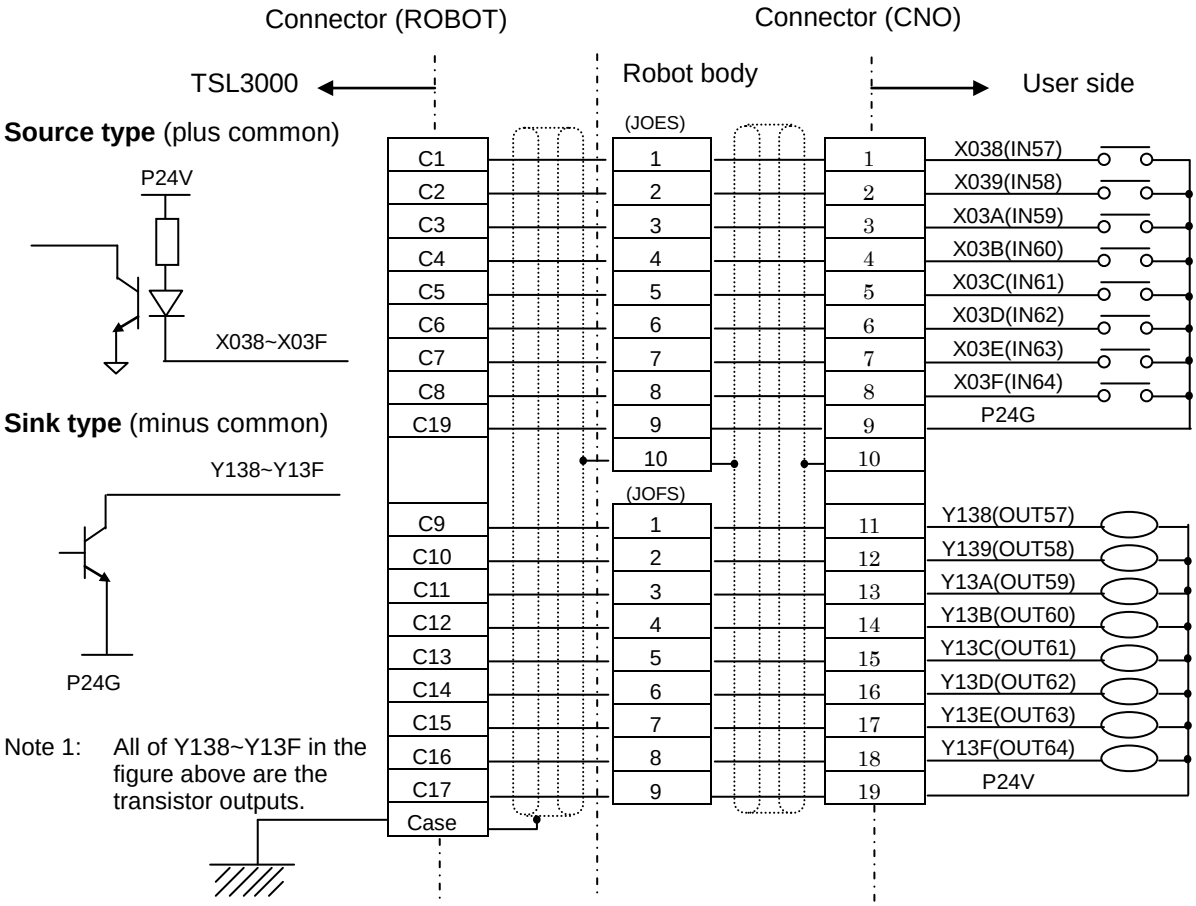
The external input signals should be connected to connectors INPUT, OUTPUT, SYSTEM, and HAND on the rear side of the controller.
The external signals are assigned to I/O relays X000 ~ X02F, Y100 ~ Y12F of the TCmini, using an attached dummy connector. (For details, see Section 4 of this manual.) To connect them, use the dummy connector attached to the controller.





Note 1: All of Y100~Y110 in the figure above are the transistor outputs.

Type N


Connector (ROBOT)
Connector (CNO)
TSL3000
Robot body
User side
Source type (plus common)
P24V
X038~X03F
Sink type (minus common)
Y138~Y13F
P24G
Note 1: All of Y138~Y13F in the figure above are the transistor outputs.
(JOES)
(JOFS)
C1
C2
C3
C4
C5
C6
C7
C8
C19
C9
C10
C11
C12
C13
C14
C15
C16
C17
Case
X038(IN57)
X039(IN58)
X03A(IN59)
X03B(IN60)
X03C(IN61)
X03D(IN62)
X03E(IN63)
X03F(IN64)
P24G
Y138(OUT57)
Y139(OUT58)
Y13A(OUT59)
Y13B(OUT60)
Y13C(OUT61)
Y13D(OUT62)
Y13E(OUT63)
Y13F(OUT64)
P24V

Type N

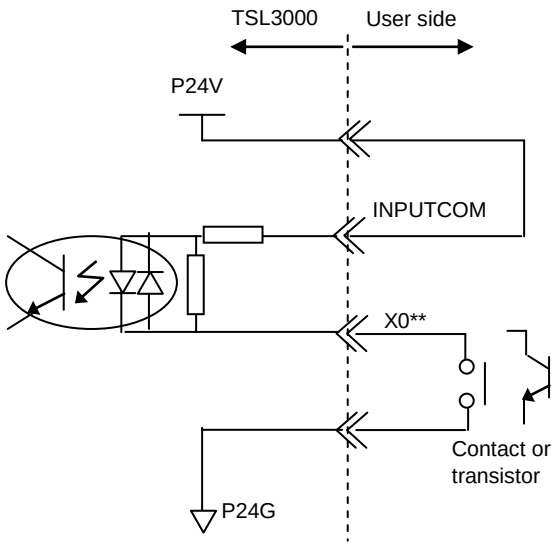
Specifications of external input signals

Type of input:

Non-voltage contact input or transistor open collector input.

Example of application circuit and structure of input circuit:

Source type (plus common)



Contact (transistor) specifications:

Non-voltage contact specification		Transistor specification	
Contact rating	24 VDC, 100 mA or over	Voltage between collector and emitter	30 V or over
Circuit current	Approx. 7 mA	Current between collector and emitter	10 mA
Minimum current	24 VDC, 1 mA	Circuit current	Approx. 7 mA
Connected impedance	100 Ω or less	Leak current between collector and emitter	100 μA

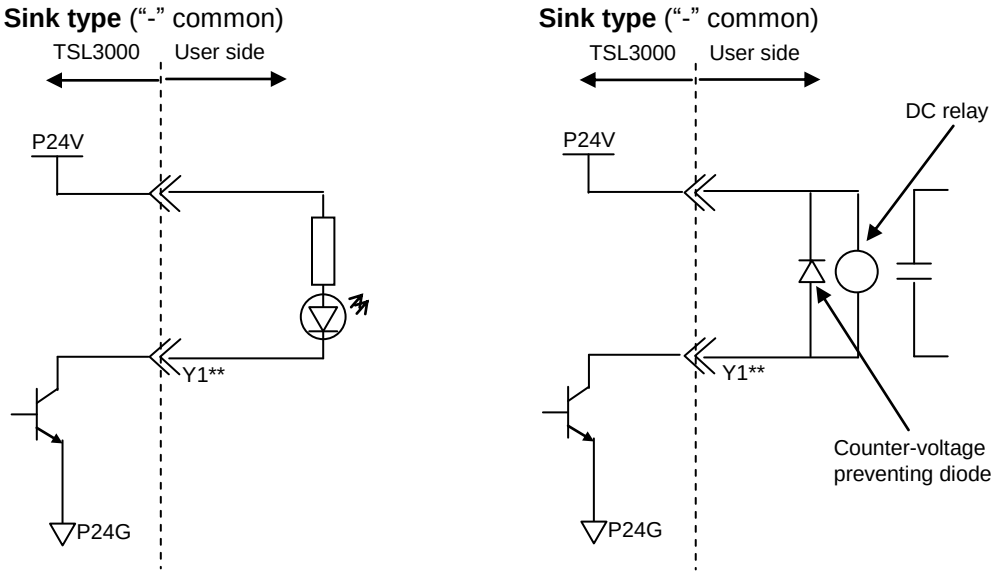
Type N

Specifications of external output signals

Type of output:

Transistor output

Example of application circuit and structure of output circuit:



- Electric rating:

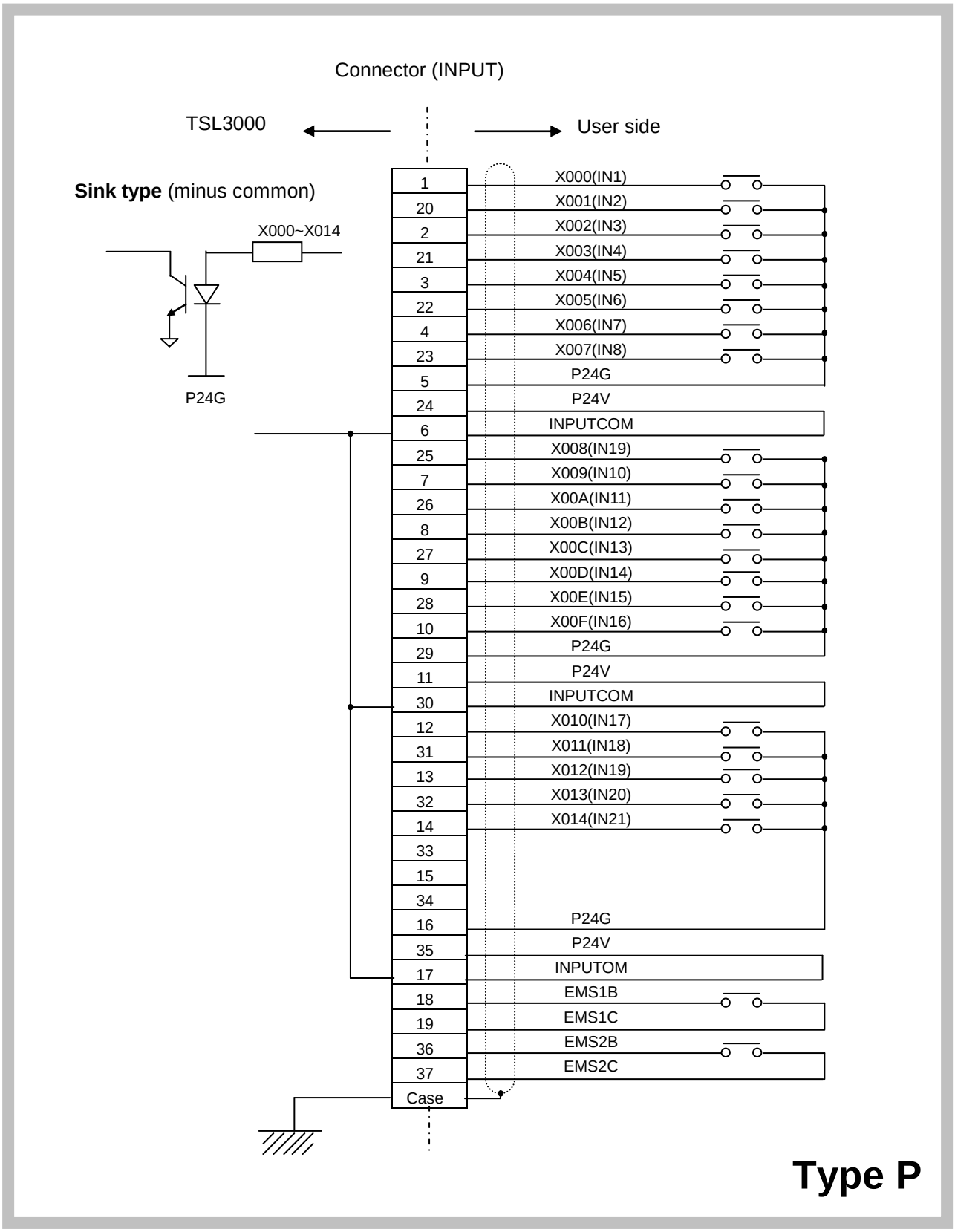
Electric rating	Caution
Rated voltage 24 VDC Rated current 100 mA (max.)	If the current which exceeds the rated output current is supplied, the output element may be damaged or the printed board may burn. To avoid this, be sure to use within the rated output current.

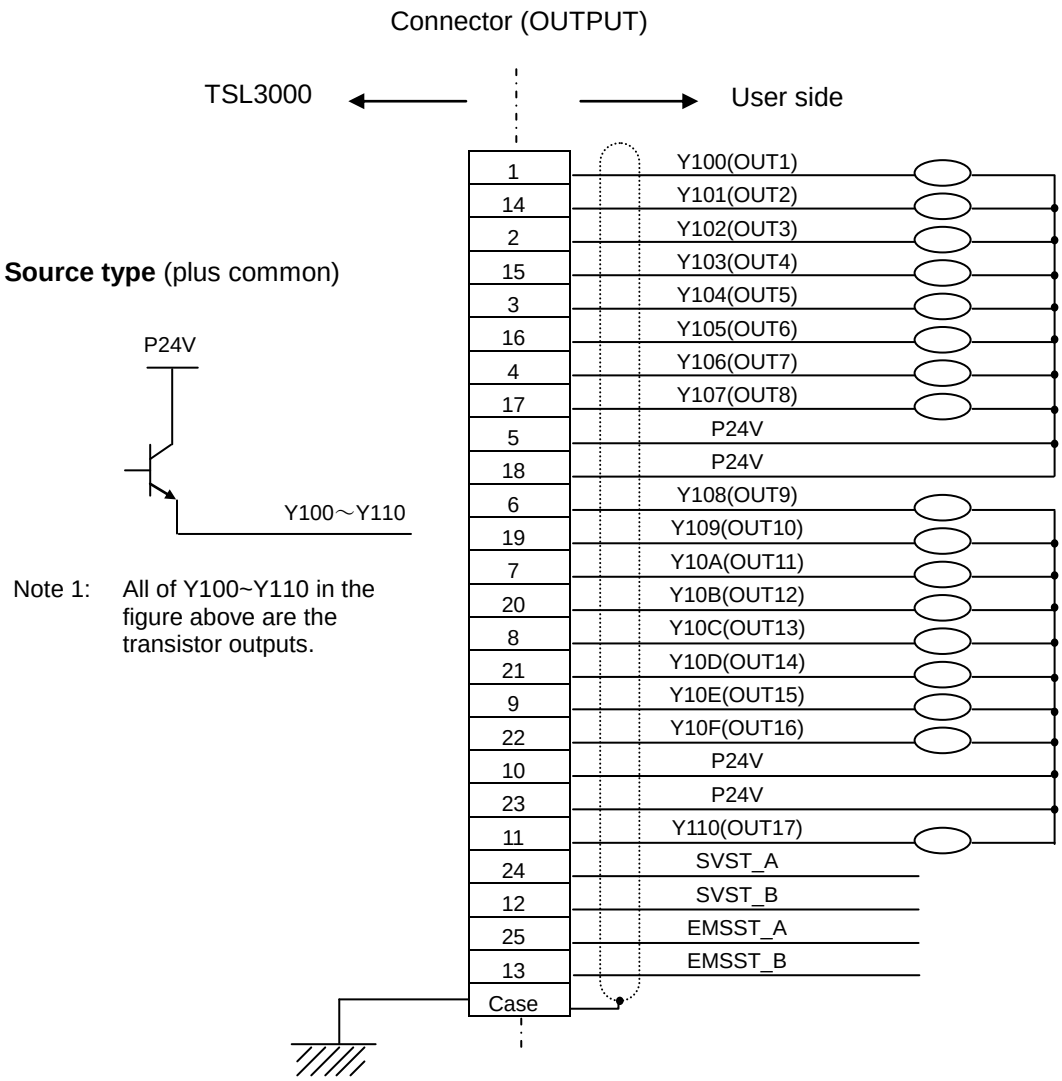
Type of connectors:

INPUT : XM3A-3721 (Plug type connector) (OMRON)
: XM2S-3711 (Connector cover) (OMRON)
OUTPUT : XM3A-2521 (Plug type connector) (OMRON)
: XM2S-2511 (Connector cover) (OMRON)
HAND(CN0): XM3A-2521 (Plug type connector) (OMRON)
: XM2S-2511 (Connector cover) (OMRON)

Type N

3.3 External Input and Output (Type P)

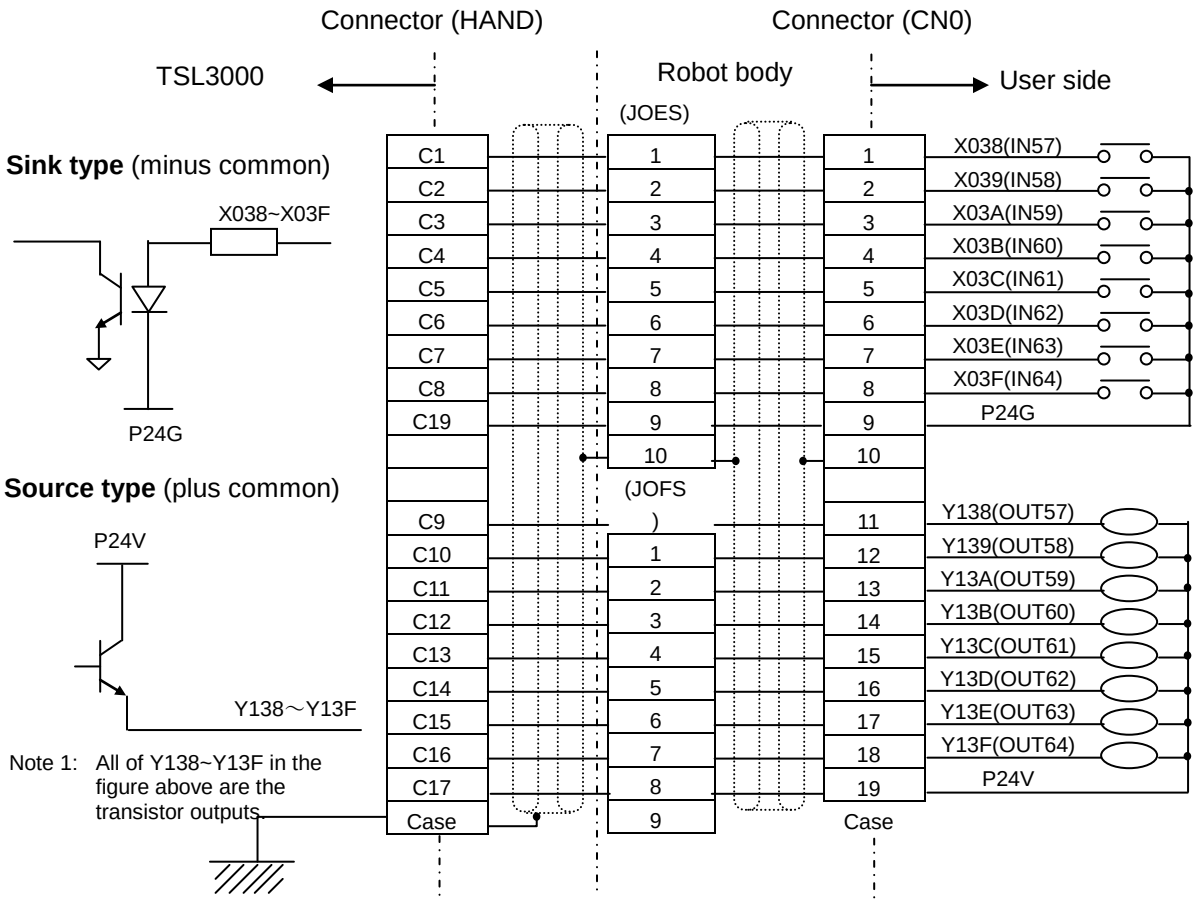




Source type (plus common)

Note 1: All of Y100~Y110 in the figure above are the transistor outputs.

Type P


Connector (HAND)
Connector (CN0)
TSL3000
Robot body
User side
(JOES)
Sink type (minus common)
X038~X03F
P24G
Source type (plus common)
P24V
Y138～Y13F
Note 1: All of Y138~Y13F in the figure above are the transistor outputs.
C1
C2
C3
C4
C5
C6
C7
C8
C19
C9
C10
C11
C12
C13
C14
C15
C16
C17
Case
1
2
3
4
5
6
7
8
9
10
(JOFS
)
1
2
3
4
5
6
7
8
9
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
Case
X038(IN57)
X039(IN58)
X03A(IN59)
X03B(IN60)
X03C(IN61)
X03D(IN62)
X03E(IN63)
X03F(IN64)
P24G
Y138(OUT57)
Y139(OUT58)
Y13A(OUT59)
Y13B(OUT60)
Y13C(OUT61)
Y13D(OUT62)
Y13E(OUT63)
Y13F(OUT64)
P24V

Type P

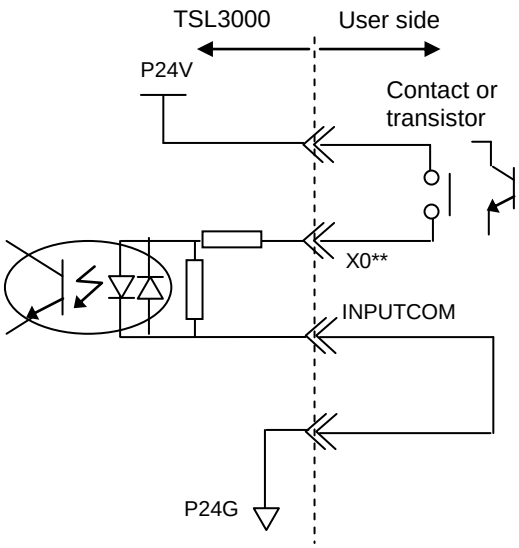
Specifications of external input signals

Type of input:

Non-voltage contact input or transistor open collector input.

Example of application circuit and structure of input circuit:

Sink type (minus common)



- Contact (transistor) specifications:

Non-voltage contact specification		Transistor specification
Contact rating	24 VDC, 100 mA or over	Voltage between collector and emitter 30 V or over
Circuit current	Approx. 7 mA	Current between collector and emitter 10 mA
Minimum current	24 VDC, 1 mA	Circuit current Approx. 7 mA
Connected impedance	100 Ω or less	Leak current between collector and emitter 100 μA

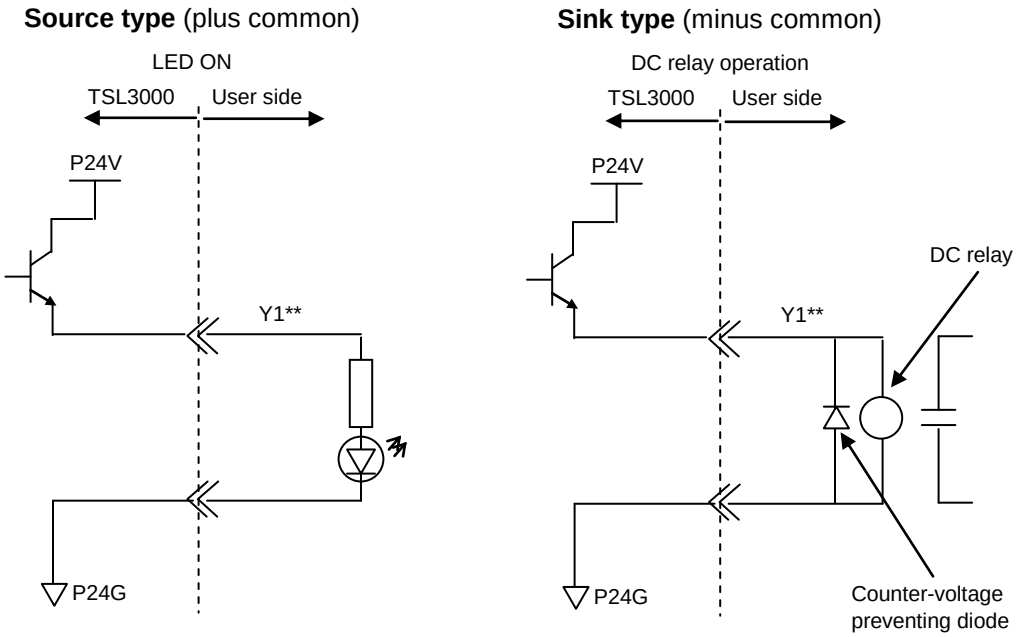
Type P

Specifications of external output signals

Type of output:

Transistor output

Example of application circuit and structure of output circuit:



Electric rating:

Electric rating		Caution
Rated voltage	24 VDC	If the current which exceeds the rated output current is supplied, the output element may be damaged or the printed board may burn. To avoid this, be sure to use within the rated output current.
Rated current	100 mA (max.)	

Type of connectors:

INPUT	:	XM3A-3721 (Plug type connector) (OMRON)
	:	XM2S-3711 (Connector cover) (OMRON)
OUTPUT	:	XM3A-2521 (Plug type connector) (OMRON)
	:	XM2S-2511 (Connector cover) (OMRON)
HAND (CN0)	:	XM3A-2521 (Plug type connector) (OMRON)
	:	XM2S-2511 (Connector cover) (OMRON)

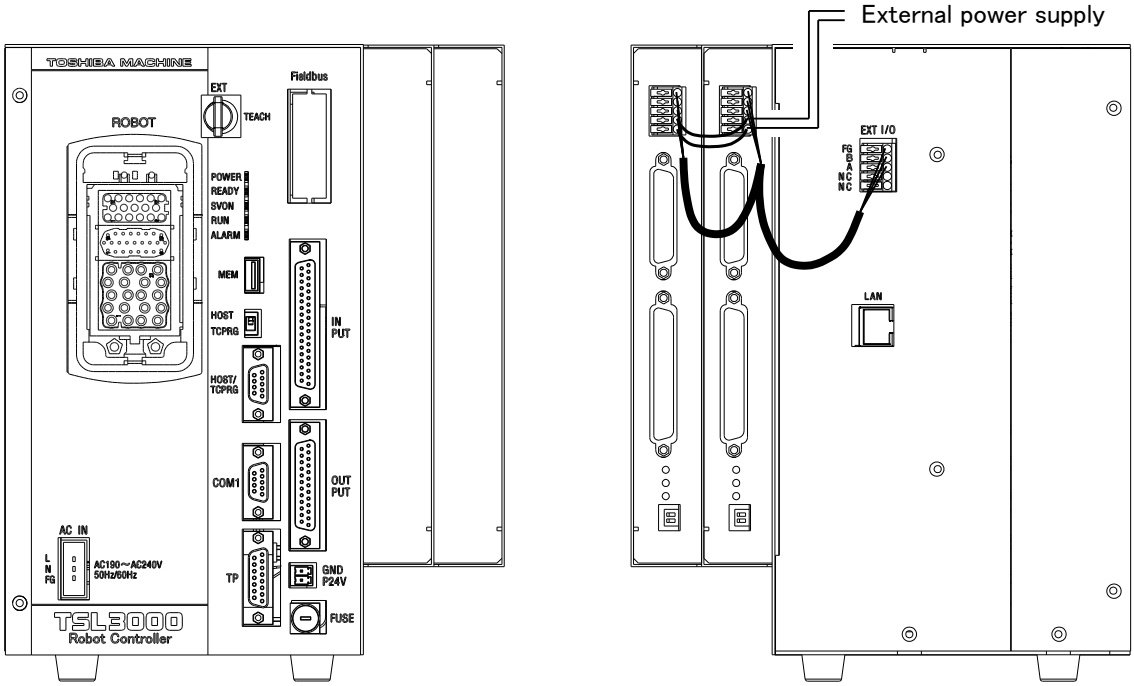
Type P

3.4 Extension Input and Output (Option)

The TCmini can extend I/Os as necessary. TR48DIOCN and TR48DIOC are available as the exclusive I/O extension modules for the TSL3000. Also, our standard remote I/O module can be connected.

3.4.1 Outline of TR48DIOCN/TR48DIOC

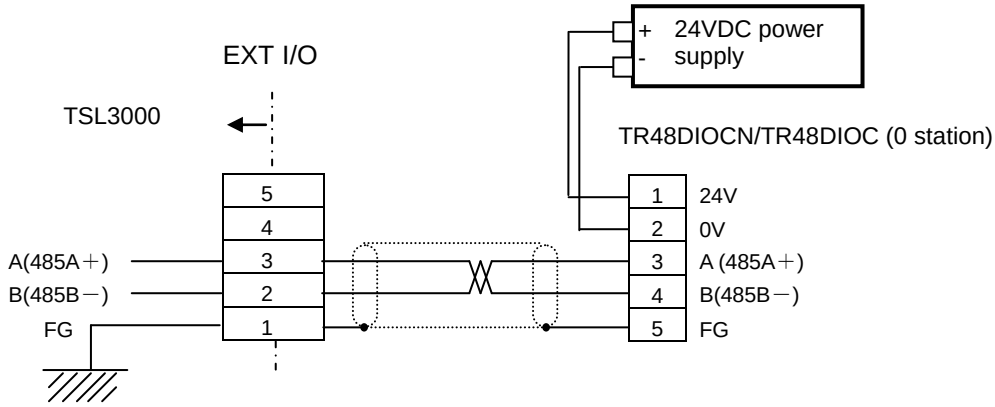
TR48DIOCN/TR48DIOC are exclusive I/O extension modules for the TSL3000, having 28 input signals and 20 output signals, respectively. In the TSL3000, up to two (2) stations can be extended, using TR48DIOCN/TR48DIOC modules. TR48DIOCN differs from TR48DIOC in the output specifications (source type/sink type).



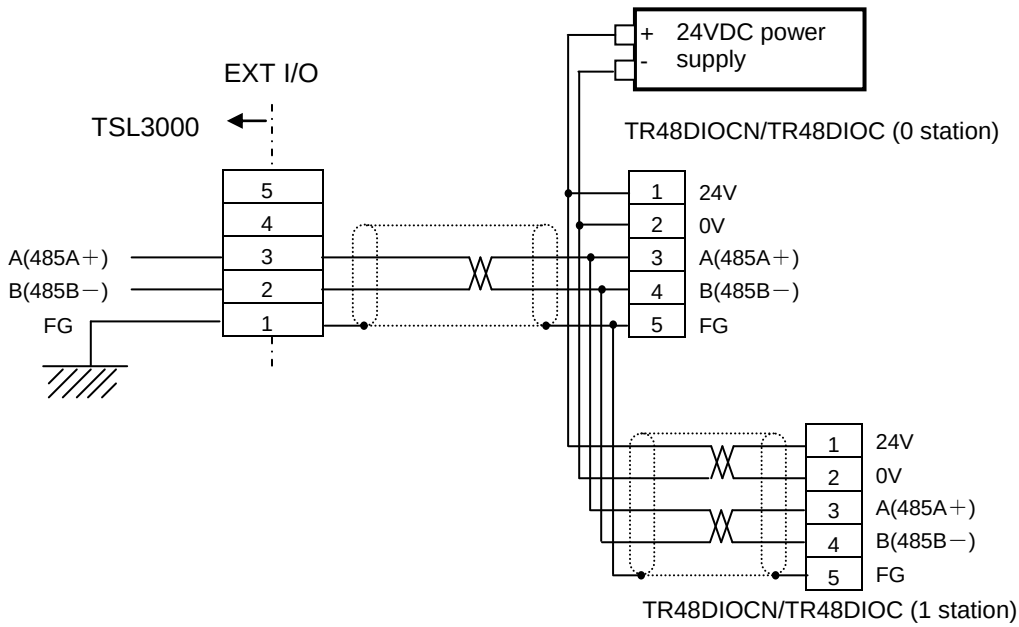
3.4.2 Connecting TR48DIOCN/TR48DIOC

The TR48DIOCN/TR48DIOC modules are connected as shown below.

When one (1) TR48DIOCN/TR48DIOC module is added:



When two (2) TR48DIOCN/TR48DIOC modules are added:



* Connect an external power supply which can supply 24 V and 0 V power. At this time, the external power supply should be turned on prior to the TSL3000 power supply. (With the TSL3000 power ON, presence or absence of extension I/Os is judged.)

3.4.3 Setting of TR48DIOCN/TR48DIOC

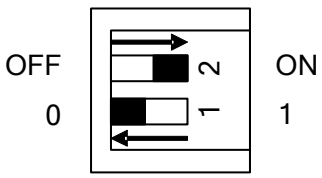
Setting of station number and terminator

For the TR48DIOCN/TR48DIOC module or modules connected, the station number and terminator should be specified.

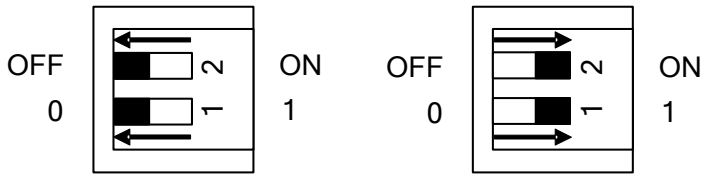
• Setting of station number and terminator:

The switch (2 pins) located on the front top side of the TR48DIOCN/TR48DIOC module is used to indicate the slave station number setting (pin 1) and terminator setting (pin 2).

Example 1: When one (1) TR48DIOCN/TR48DIOC module is added



Example 2: When two (2) TR48DIOCN/TR48DIOC modules are added



As shown in the figures above, set the station number for the TR48DIOCN/TR48DIOC module to the station number that is already set in the USER.PAR file.

For the terminator, when only one (1) TR48DIOCN/TR48DIOC module is used, set ON the terminator setting switch equipped on the module.

When two (2) TR48DIOCN/TR48DIOC modules are used, see the figure in Para. 3.6.1 for example. As the TR48DIOCN/TR48DIOC module on the extreme right side as viewed from the controller front side is the terminal station in terms of cable wiring, set ON the terminator setting switch equipped on this module alone.

Setting of user parameter:
To recognize an I/O extension module by the TCmini built in the TSL3000, setting of appropriate user parameter is necessary.

* To designate the slave station number of the TSL3000, observe the following steps. Under [U12] of the USER.PAR (user parameter) file, you can find the following parameter setting related to the extension input and output.

  [U12] Extend I/O setting
  {Use/Not Use} (0: Not Use, 1: Use)
  {Not Use}
  {Not Use}
  = 0 0 0 → Corresponds to station 0.
  = 0 0 0 → Corresponds to station 1.

  If you wish to use only station 0 (i.e., one (1) TR48DIOCN module), for instance, change the underlined bit corresponding to station 0 to "1".

  = 1 0 0
  = 0 0 0

  After saving the parameter file, turn off the controller power and on again. Then the above parameter becomes effective.

  If you wish to use both stations 0 and 1 (i.e., two (2) TR48DIOCN modules), specify as follows and perform the same operation as above to make the parameter valid.

  = 1 0 0
  = 1 0 0

When using the TR48DIOCN/TR48DIOC module, careful precautions should be taken on the following matters.

① Make sure that the slave station number set in "USER.PAR" coincides with the station number setting of the TR48DIOCN/TR48DIOC module.
② When the TR48DIOCN/TR48DIOC module or modules are connected, set ON the terminator setting switch equipped on the TR48DIOCN/TR48DIOC module which is located on the extreme right side in terms of cable wiring when seen from the controller front side.

The switches (2 pins) provided on the front upper side of the TR48DIOC module represent pin 1 (upper) for slave station number setting and pin 2 (lower) for terminator setting.

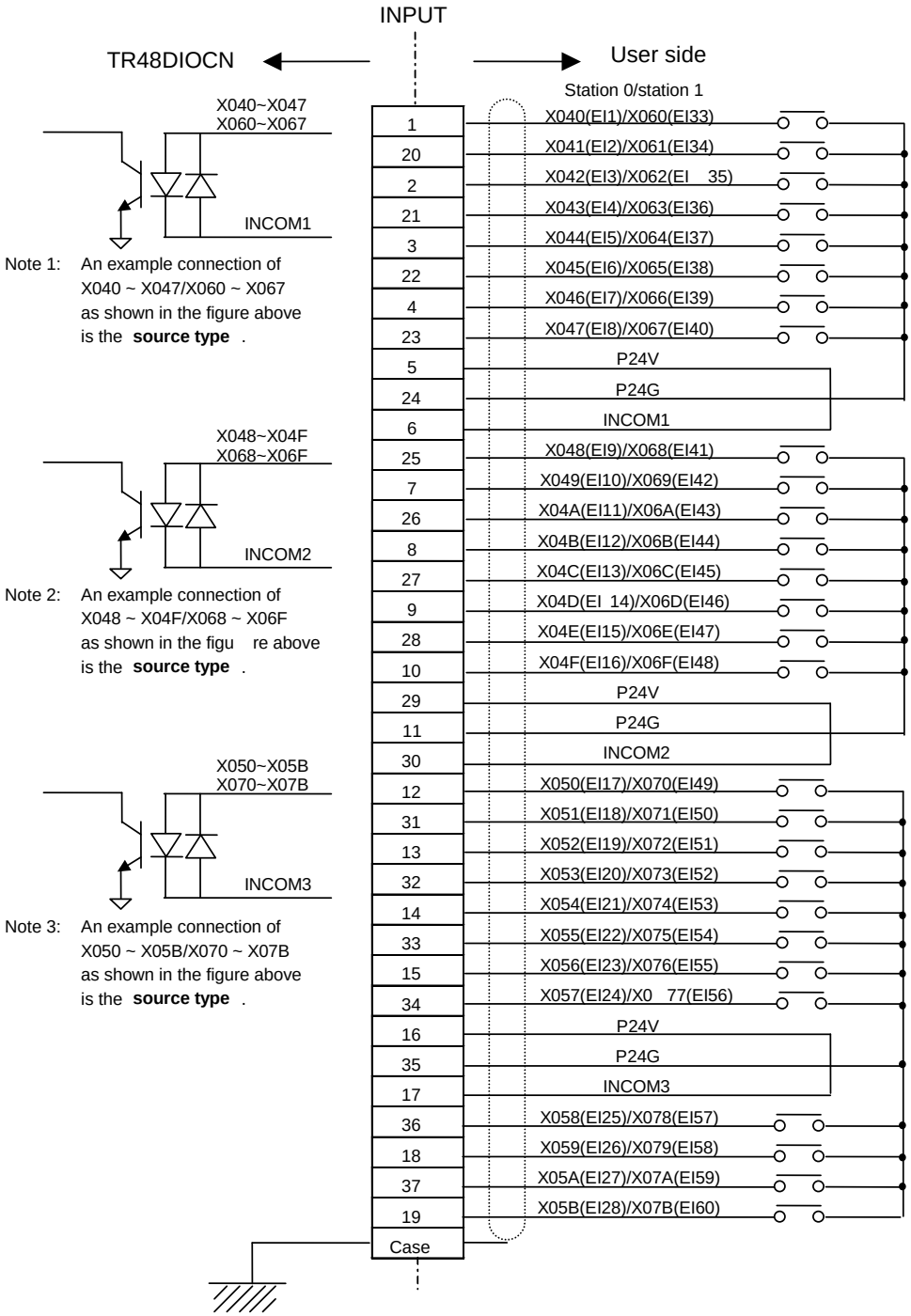
Be sure to execute Items ① and ② above, irrespective of the presence or absence of external power supply. Otherwise, the system may not function normally or go wrong.

With the above setting, make sure that both the POWER and RUN LEDs on each TR48DIOCN/TR48DIOC module are illuminated.

3.4.4 TR48DIOCN Input and Output

Input circuit

TR48DIOCN input signals EI01 ~ EI28 (for station 0) and EI33 ~ EI60 (for station 1) should be connected to the INPUT connector on the front side of the module. The extension input signals thus connected are assigned to I/O relays X040 ~ X05B (for station 0) and X060 ~ X07B (for station 1) of the TCmini.
(For details, see Section 4 of this manual.)



* A bilateral photo coupler is used in the input circuit. **The source type or sink type can be selected** by means of INCOM*. **The source type is exemplified above.**

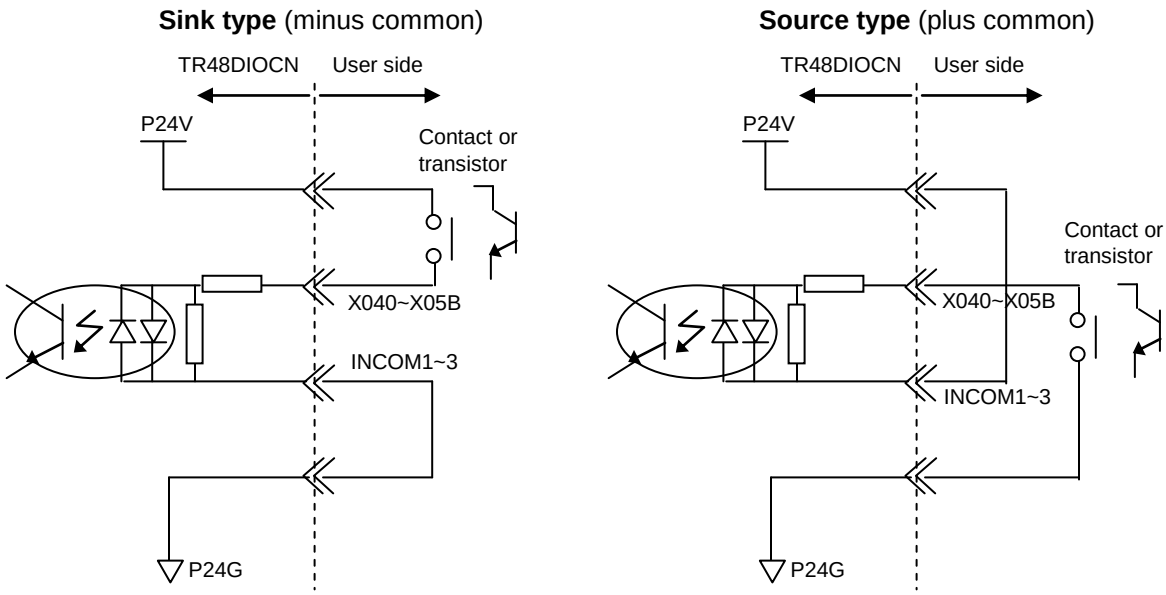
TR48DIOCN

Specifications of TR48DIOCN input signals

Type of input:

Non-voltage contact input or transistor open collector input.

Example of application circuit and structure of input circuit:



Contact (transistor) specifications:

Non-voltage contact specification		Transistor specification	
Contact rating	24 VDC, 10 mA or over	Voltage between collector and emitter	30 V or over
Circuit current	Approx. 7 mA	Current between collector and emitter	10 mA
Minimum current	24 VDC, 1 mA	Circuit current	Approx. 7 mA
Connected impedance	100 Ω or less	Leak current between collector and emitter	100 μA

Type of connector:

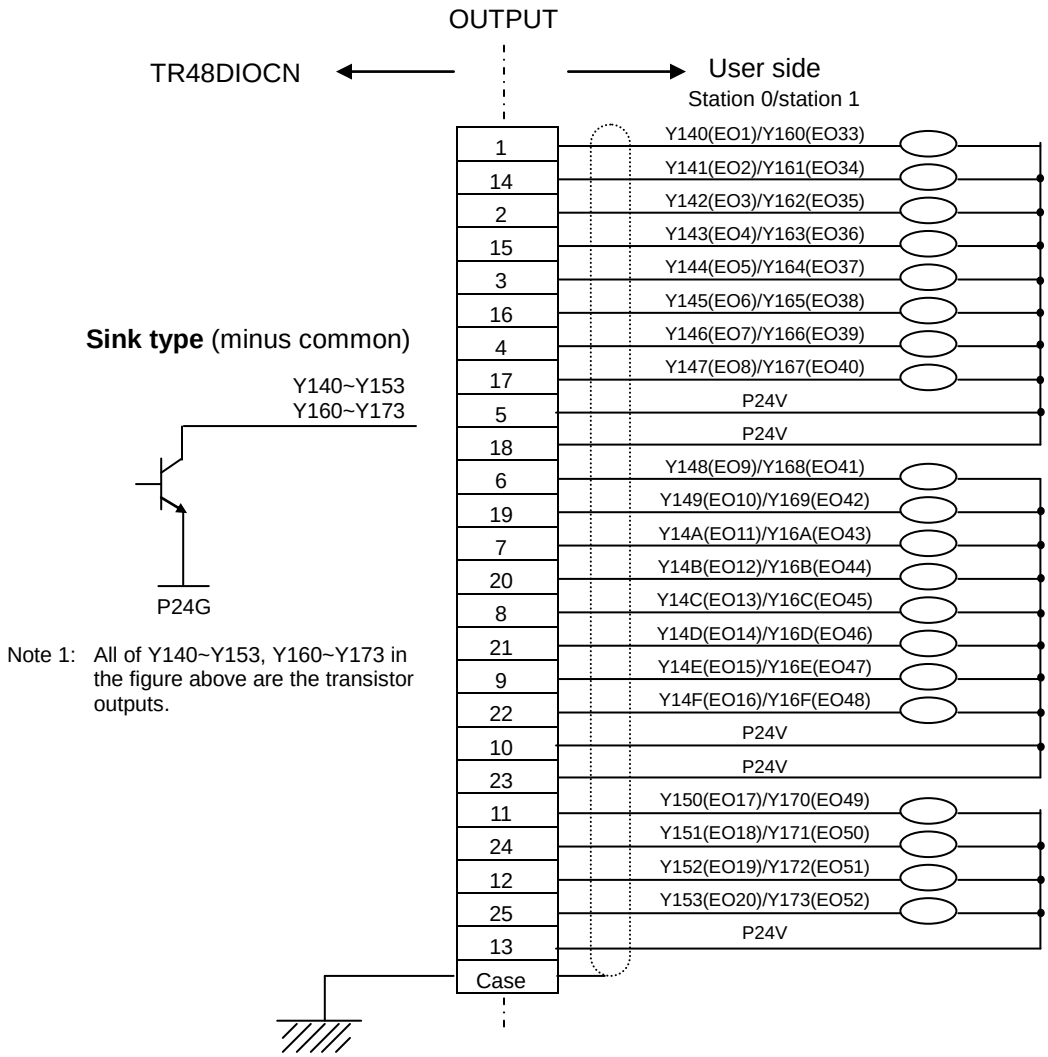
XM3A–3721 (Plug type connector) OMRON

XM2S–3711 (Connecter cover) OMRON

TR48DIOCN

Output circuit

TR48DIOCN output signals EO01 ~ EO20 (for station 0) and EO33 ~ EO52 (for station 1) should be connected to the OUTPUT connector on the front side of the module. The extension output signals thus connected are assigned to I/O relays Y140 ~ Y153 (for station 0) and Y160 ~ Y173 (for station 1) of the TCmini.
(For details, see Section 4 of this manual.)



Note 1: All of Y140~Y153, Y160~Y173 in the figure above are the transistor outputs.

TR48DIOCN

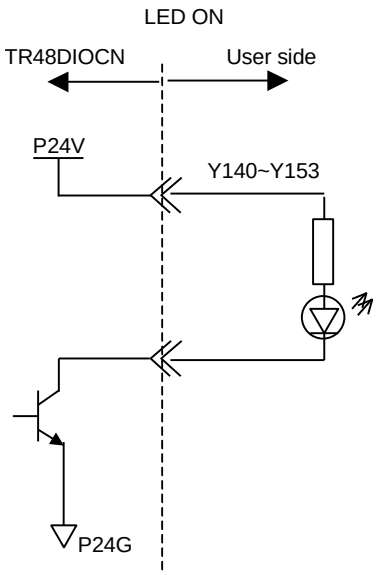
Specifications of TR48DIOCN output signals

Type of output:

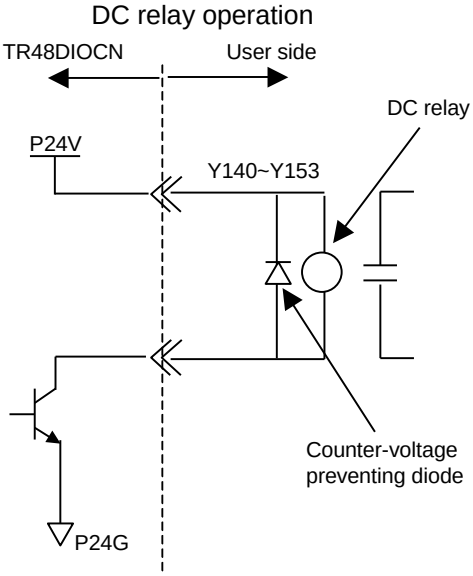
Transistor output

Example of application circuit and structure of output circuit:

Sink type ("–" common)



Sink type ("–" common)



- Electric rating:

Electric rating	Caution
Rated voltage 24 VDC Rated current 100 mA (max.)	If the current which exceeds the rated output current is supplied, the output element may be damaged or the printed board may burn. To avoid this, be sure to use within the rated output current.

Type of connector:

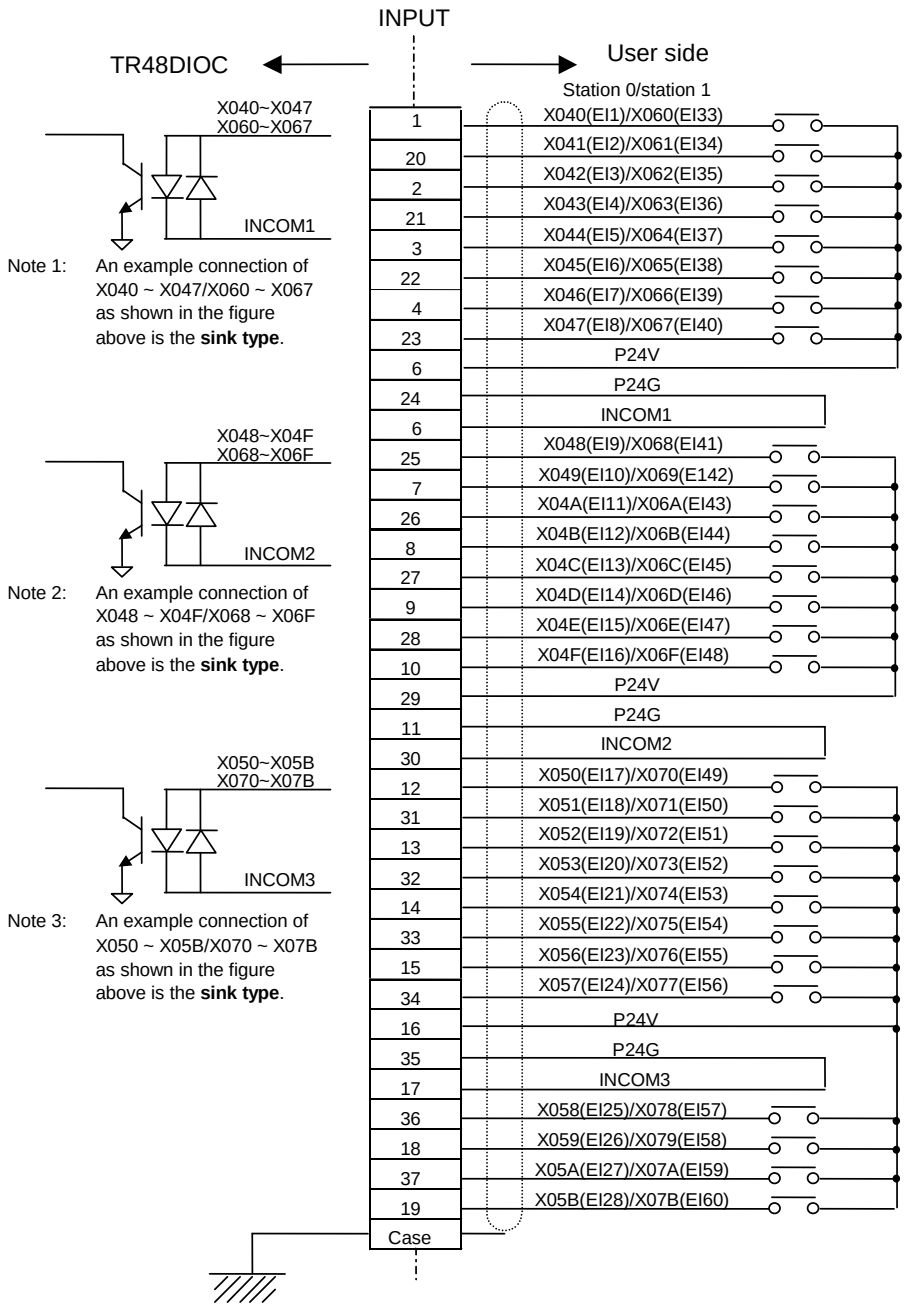
Pin type connector: XM2A–2501 made by OMRON

Connector cover: XM2S–2511 made by OMRON

3.4.5 TR48DIOC Input and Output

Input circuit

TR48DIOC input signals EI01 ~ EI28 (for station 0) and EI33 ~ EI60 (for station 1) should be connected to the INPUT connector on the front side of the module. The extension input signals thus connected are assigned to I/O relays X040 ~ X05B (for station 0) and X060 ~ X07B (for station 1) of the TCmini. (For details, see Section 4 of this manual.)



* A bilateral photo coupler is used in the input circuit. **The source type or sink type can be selected** by means of INCOM*. **The sink type is exemplified above.**

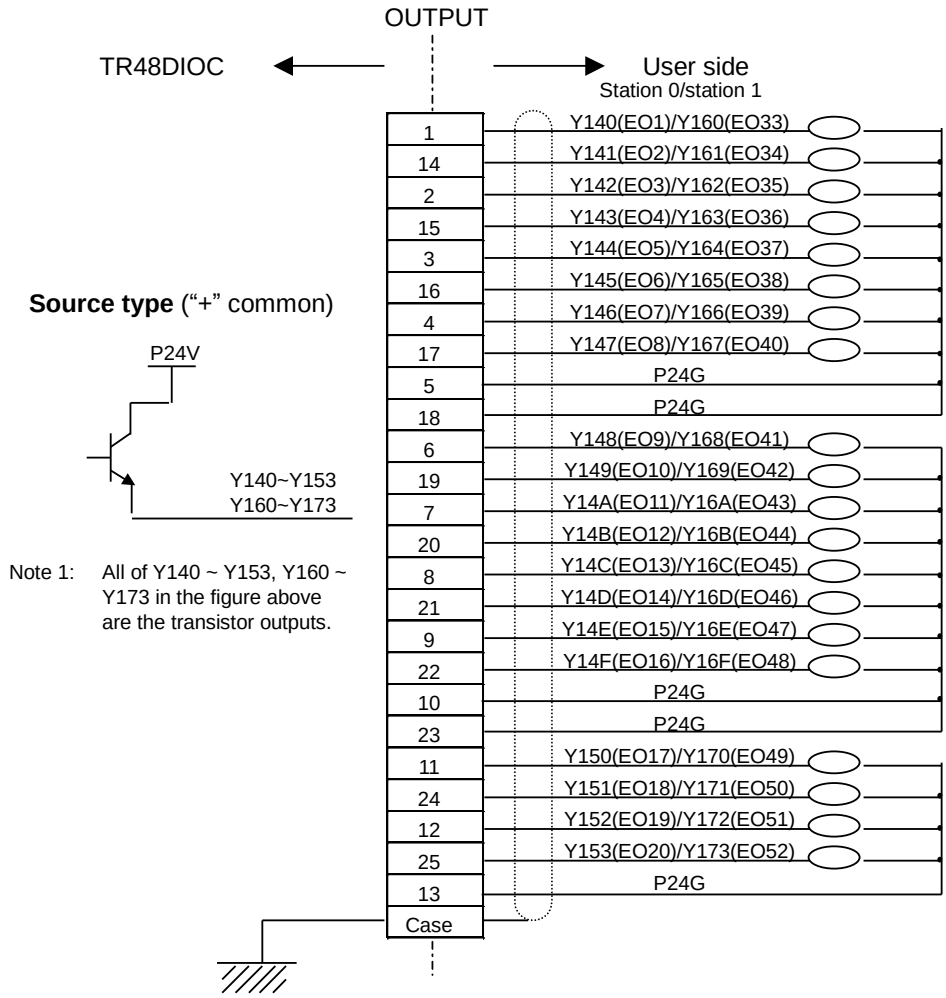
TR48DIOC

Specifications of TR48DIOC input signals

The **input specifications of TR48DIOC** are the same as those of TR48DIOCN. For details, see the input specifications in Para. **3.8.4**.

Output circuit

TR48DIOC output signals EO01 ~ EO20 (for station 0) and EO33 ~ EO52 (for station 1) should be connected to the OUTPUT connector on the front side of the module. The extension output signals thus connected are assigned to I/O relays Y140 ~ Y153 (for station 0) and Y160 ~ Y173 (for station 1) of the TCmini. (For details, see Section 4 of this manual.)



TR48DIOC

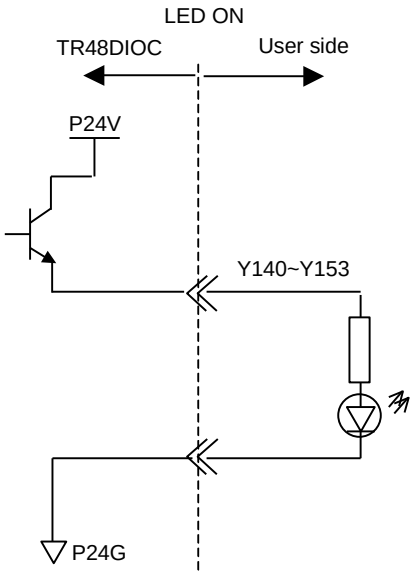
Specifications of TR48DIOC output signals

Type of output:

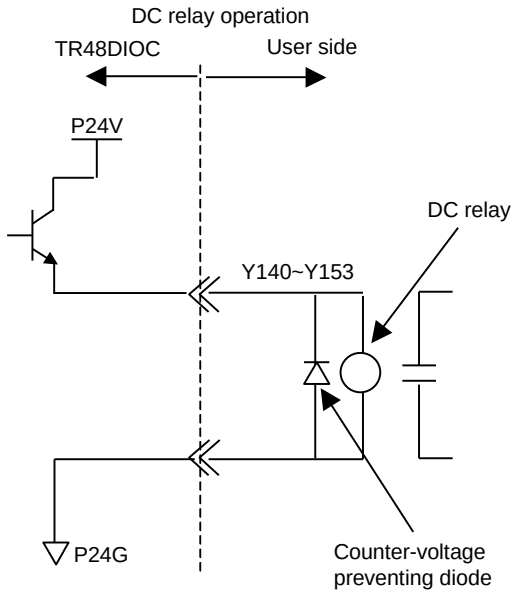
Transistor output

Example of application circuit and structure of output circuit:

Source type ("+" common)



Source type ("+" common)



- Electric rating:

Electric rating	Caution
Rated voltage 24 VDC Rated current 100 mA (max.)	If the current which exceeds the rated output current is supplied, the output element may be damaged or the printed board may burn. To avoid this, be sure to use within the rated output current.

Type of connector:

Pin type connector: XM2A–2501 made by OMRON

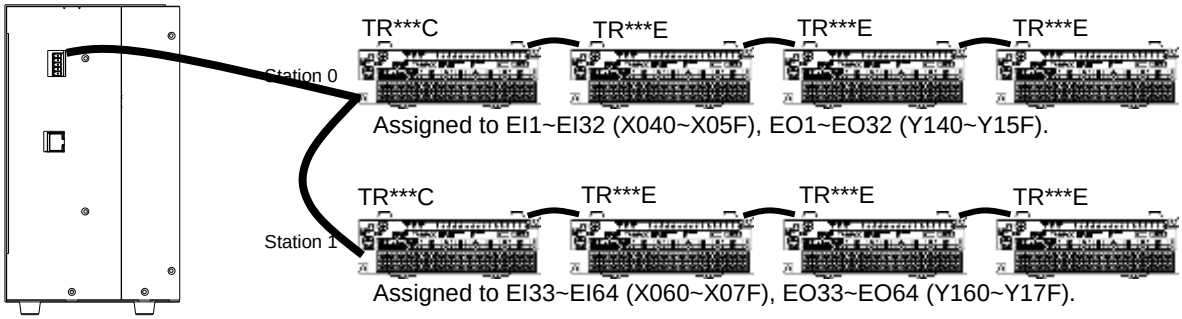
Connector cover: XM2S–2511 made by OMRON

3.4.6 Remote I/O Module

In addition to the TR48DIOC modules, the TCmini built in the TSL3000 can also connect our standard remote I/O modules which are shown in the table below.

Construction	Type	Input spec.	No. of inputs	Output spec.	No. of outputs	Remarks
Unit	TR48DIOCN	DC24	28	DC24	20	Exclusively used for the robot. Internal power supply.
Unit	TR48DIOC	DC24	28	DC24	20	Exclusively used for the robot. Internal power supply.
Printed board	TR64DIRYC	DC24	32	Relay	32	I/O external power supply
Printed board	TR32DIRYC	DC24	16	Relay	16	
Printed board	TR32DIDOPC	DC24	16	DC24	16	"+" common
Printed board	TR32DIDONC	DC24	16	DC24	16	"–" common
Terminal block	TR16DIC	DC24	16		0	Slave station
Terminal block	TR16DIE	DC24	16		0	Extension
Terminal block	TR16DOPC		0	DC24	16	Slave station, "+" common
Terminal block	TR16DOPE		0	DC24	16	Extension, "+" common
Terminal block	TR16DONC		0	DC24	16	Slave station, "–" common
Terminal block	TR16DONE		0	DC24	16	Extension, "–" common
Terminal block	TR16AOC		0	AC100	16	Slave station
Terminal block	TR16AOE		0	AC100	16	Extension
Terminal block	TR16RYC		0	Relay	16	Slave station
Terminal block	TR16RYE		0	Relay	16	Extension

Conditions for selecting remote I/O module:



① Up to two (2) remote I/O modules can be connected.
Alphabet "C" at the end of the type signifies the slave station, and a total of two (2) modules of this type can be connected.

② The number of inputs and outputs per station is 0 ~ 32 each.

③ One (1) printed board serves as one (1) station.

④ For the terminal block type module, up to three (3) TR16**E modules can be combined with a TR16**C module serving as the slave station. Be sure to observe the restrictions in Item ②, however.

3.5 Field Bus Input and Output (Option)

Input and output signals of the field bus slave function (option) can also be controlled from the TCmini.

For details on how to connect to the fieldbus, see the Fieldbus Slave Operation Manual.

Input signals (128) from the field bus are assigned to X200 ~ X27F.

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
X20W	FI16	FI15	FI14	FI13	FI12	FI11	FI10	FI9	FI8	FI7	FI6	FI5	FI4	FI3	FI2	FI1
X21W	FI32	FI31	FI30	FI29	FI28	FI27	FI26	FI25	FI24	FI23	FI22	FI21	FI20	FI19	FI18	FI17
X22W	FI48	FI47	FI46	FI45	FI44	FI43	FI42	FI41	FI40	FI39	FI38	FI37	FI36	FI35	FI34	FI33
X23W	FI64	FI63	FI62	FI61	FI60	FI59	FI58	EI57	FI56	FI55	FI54	FI53	FI52	FI51	FI50	FI49
X24W	FI80	FI79	FI78	FI77	FI76	FI75	FI74	FI73	FI72	FI71	FI70	FI69	FI68	FI67	FI66	FI65
X25W	FI96	FI95	FI94	FI93	FI92	FI91	FI90	FI89	FI88	FI87	FI86	FI85	FI84	FI83	FI82	FI81
X26W	FI112	FI111	FI110	FI109	FI108	FI107	FI106	FI105	FI104	FI103	FI102	FI101	FI100	FI99	FI98	FI97
X27W	FI128	FI127	FI126	FI125	FI124	FI123	FI122	FI121	FI120	FI119	FI118	FI117	FI116	FI115	FI114	FI113

Output signals (128) to the field bus are assigned to Y300 ~ Y37F.

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Y30W	FO16	FO15	FO14	FO13	FO12	FO11	FO10	FO9	FO8	FO7	FO6	FO5	FO4	FO3	FO2	FO1
Y31W	FO32	FO31	FO30	FO29	FO28	FO27	FO26	FO25	FO24	FO23	FO22	FO21	FO20	FO19	FO18	FO17
Y32W	FO48	FO47	FO46	FO45	FO44	FO43	FO42	FO41	FO40	FO39	FO38	FO37	FO36	FO35	FO34	FO33
Y33W	FO64	FO63	FO62	FO61	FO60	FO59	FO58	FO57	FO56	FO55	FO54	FO53	FO52	FO51	FO50	FO49
Y34W	FO80	FO79	FO78	FO77	FO76	FO75	FO74	FO73	FO72	FO71	FO70	FO69	FO68	FO67	FO66	FO65
Y35W	FO96	FO95	FO94	FO93	FO92	FO91	FO90	FO89	FO88	FO87	FO86	FO85	FO84	FO83	FO82	FO81
Y36W	FO112	FO111	FO110	FO109	FO108	FO107	FO106	FO105	FO104	FO103	FO102	FO101	FO100	FO99	FO98	FO97
Y37W	FO128	FO127	FO126	FO125	FO124	FO123	FO122	FO121	FO120	FO119	FO118	FO117	FO116	FO115	FO114	FO113

Data input (CC-Link→TCmini) *CCLink only

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D70*	FB RWr 16	FB RWr 15	FB RWr 14	FB RWr 13	FB RWr 12	FB RWr 11	FB RWr 10	FB RWr 9	FB RWr 8	FB RWr 7	FB RWr 6	FB RWr 5	FB RWr 4	FB RWr 3	FB RWr 2	FB RWr 1
D71*	Reserved area															
D72*																
D73*																

Data output (TCmini CC→Link) *CCLink only

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D74*	FB RWw 16	FB RWw 15	FB RWw 14	FB RWw 13	FB RWw 12	FB RWw 11	FB RWw 10	FB RWw 9	FB RWw 8	FB RWw 7	FB RWw 6	FB RWw 5	FB RWw 4	FB RWw 3	FB RWw 2	FB RWw 1
D75*	Reserved area															
D76*																
D77*																

FI1 ~ FI128

Address	**X200 ~ X27F**
Direction	Fieldbus → TCmini
Function	The bit input data received from the fieldbus is saved.

FO1 ~ FO128

Address	**Y300 ~ X37F**
Direction	TCmini → Fieldbus
Function	Bit data is output to the fieldbus.

FBRWr1 ~ FBRWr16

Address	**D700 ~ D70F**
Direction	Fieldbus → TCmini
Function	Word data (16-bit data) received from the fieldbus is saved. *This function supports CC-Link only.

FBRWw1 ~ FBRWw16

Address	**D740 ~ D74F**
Direction	TCmini → Fieldbus
Function	Word data (16-bit data) is output to the fieldbus. *This function supports CC-Link only.

Section 4 Relays

4.1 Input/Output Relays

Inputs to and outputs from the TCmini are assigned to X relays (X000 ~ X07F, X200 ~ X27F) and Y relays (Y100 ~ Y17F, Y300 ~ Y30F), respectively.

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
X00W	IN16	IN15	IN14	IN13	IN12	IN11	IN10	IN9	IN8	IN7	IN6	IN5	IN4	IN3	IN2	IN1
X01W	IN32	IN31	IN30	IN29	IN28	IN27	IN26	IN25	IN24	IN23	IN22	IN21	IN20	IN19	IN18	IN17
X02W	IN48	IN47	IN46	IN45	IN44	IN43	IN42	IN41	IN40	IN39	IN38	IN37	IN36	IN35	UFI2	UFI1
X03W	HI8	HI7	HI6	HI5	HI4	HI3	HI2	HI1	LI8	LI7	LI6	LI5	LI4	LI3	LI2	LI1
X04W	EI16	EI15	EI14	EI13	EI12	EI11	EI10	EI9	EI8	EI7	EI6	EI5	EI4	EI3	EI2	EI1
X05W	EI32	EI31	EI30	EI29	EI28	EI27	EI26	EI25	EI24	EI23	EI22	EI21	EI20	EI19	EI18	EI17
X06W	EI48	EI47	EI46	EI45	EI44	EI43	EI42	EI41	EI40	EI39	EI38	EI37	EI36	EI35	EI34	EI33
X07W	EI64	EI63	EI62	EI61	EI60	EI59	EI58	EI57	EI56	EI55	EI54	EI53	EI52	EI51	EI50	EI49
Y10W	OUT	OUT	OUT	OUT	OUT	OUT1	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT	OUT
Y11W	OUT3	OUT3	OUT3	OUT2	OUT2	OUT2	OUT2	OUT2	OUT2	OUT2	OUT2	OUT2	OUT2	OUT1	OUT1	OUT1
Y12W	OUT4	OUT4	OUT4	OUT4	OUT4	OUT4	OUT4	OUT4	OUT4	OUT3	OUT3	OUT3	OUT3	OUT3	UFO	UFO
Y13W	HO8	HO7	HO6	HO5	HO4	HO3	HO2	HO1								
Y14W	EO16	EO15	EO14	EO13	EO12	EO11	EO10	EO9	EO8	EO7	EO6	EO5	EO4	EO3	EO2	EO1
Y15W	EO32	EO31	EO30	EO29	EO28	EO27	EO26	EO25	EO24	EO23	EO22	EO21	EO20	EO19	EO18	EO17
Y16W	EO48	EO47	EO46	EO45	EO44	EO43	EO42	EO41	EO40	EO39	EO38	EO37	EO36	EO35	EO34	EO33
Y17W	EO64	EO63	EO62	EO61	EO60	EO59	EO58	EO57	EO56	EO55	EO54	EO53	EO52	EO51	EO50	EO49
X20W	FI16	FI15	FI14	FI13	FI12	FI11	FI10	FI9	FI8	FI7	FI6	FI5	FI4	FI3	FI2	FI1
X21W	FI32	FI31	FI30	FI29	FI28	FI27	FI26	FI25	FI24	FI23	FI22	FI21	FI20	FI19	FI18	FI17
X22W	FI48	FI47	FI46	FI45	FI44	FI43	FI42	FI41	FI40	FI39	FI38	FI37	FI36	FI35	FI34	FI33
X23W	FI64	FI63	FI62	FI61	FI60	FI59	FI58	FI57	FI56	FI55	FI54	FI53	FI52	FI51	FI50	FI49
X24W	FI80	FI79	FI78	FI77	FI76	FI75	FI74	FI73	FI72	FI71	FI70	FI69	FI68	FI67	FI66	FI65
X25W	FI96	FI95	FI94	FI93	FI92	FI91	FI90	FI89	FI88	FI87	FI86	FI85	FI84	FI83	FI82	FI81
X26W	FI112	FI111	FI110	FI109	FI108	FI107	FI106	FI105	FI104	FI103	FI102	FI101	FI100	FI99	FI98	FI97
X27W	FI128	FI127	FI126	FI125	FI124	FI123	FI122	FI121	FI120	FI119	FI118	FI117	FI116	FI115	FI114	FI113
Y30W	FO16	FO15	FO14	FO13	FO12	FO11	FO10	FO9	FO8	FO7	FO6	FO5	FO4	FO3	FO2	FO1
Y31W	FO32	FO31	FO30	FO29	FO28	FO27	FO26	FO25	FO24	FO23	FO22	FO21	FO20	FO19	FO18	FO17
Y32W	FO48	FO47	FO46	FO45	FO44	FO43	FO42	FO41	FO40	FO39	FO38	FO37	FO36	FO35	FO34	FO33
Y33W	FO64	FO63	FO62	FO61	FO60	FO59	FO58	FO57	FO56	FO55	FO54	FO53	FO52	FO51	FO50	FO49
Y34W	FO80	FO79	FO78	FO77	FO76	FO75	FO74	FO73	FO72	FO71	FO70	FO69	FO68	FO67	FO66	FO65
Y35W	FO96	FO95	FO94	FO93	FO92	FO91	FO90	FO89	FO88	FO87	FO86	FO85	FO84	FO83	FO82	FO81
Y36W	FO112	FO111	FO110	FO109	FO108	FO107	FO106	FO105	FO104	FO103	FO102	FO101	FO100	FO99	FO98	FO97
Y37W	FO128	FO127	FO126	FO125	FO124	FO123	FO122	FO121	FO120	FO119	FO118	FO117	FO116	FO115	FO114	FO113
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

X*** Represents an exclusive input relay which is connected to the standard input, panel input, hand input or extension input signal.
The ON or OFF status is read at I/O processing in each scanning cycle.
It can be used as the contact input information and data register source in the sequence program.

Y*** Signifies an exclusive output relay which is connected to the standard output, panel output, hand output or extension output signal.
It serves as the destination of coil and data register in the sequence program, where operation result is written.
The operation result can be used as the source of contact and data register in the sequence program.
The operation result is transferred as the ON/OFF information to the output device at I/O processing in each scanning cycle.

 Relays reserved for future extension of the system functions. Values are undecided and cannot be used.

For further information on the input and output specifications, see Section 3.

4.2 Internal Relay

A total of 1024 relays (R000 ~ R7F) can be used as the internal relay.

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
R00W	R00F															R000
R01W																
R02W																
R03W																
R04W																
R05W																
R06W																
R07W																
R10W	R10F															R100
R11W																
R12W																
R13W																
R14W																
R15W																
R16W																
R17W																
R20W	R20F															R200
R21W																
R22W																
R23W																
R24W																
R25W																
R26W																
R27W																
R30W	R30F															R300
R31W																
R32W																
R33W																
R34W																
R35W																
R36W																
R37W																
R40W	R40F															R400
R41W																
R42W																
R43W																
R44W																
R45W																
R46W																
R47W																

R50W	R50F															R500
R51W																
R52W																
R53W																
R54W																
R55W																
R56W																
R57W																
R60W	R60F															R600
R61W																
R62W																
R63W																
R64W																
R65W																
R66W																
R67W																
R70W	R70F															R700
R71W																
R72W																
R73W																
R74W																
R75W																
R76W																
R77W																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

R000 ~ R77F:

Serve as the destination of coil and data register in the user program, where operation result is written.

They can be used as a temporary storage of operation result which need not be output to the external device.

The operation result can be used as the source of contact and data register in the user program.

4.3 Interface Relay

The interface relay serves as the interface area for transferring signals with the main unit of the robot controller.
For details, see Section 6 of this manual.

TCmini → Robot controller main unit (G000 ~ G27F)

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
G00W	DIN 16	DIN 15	DIN 14	DIN 13	DIN 12	DIN 11	DIN 10	DIN 9	DIN 8	DIN 7	DIN 6	DIN 5	DIN 4	DIN 3	DIN 2	DIN 1
G01W	DIN 32	DIN 31	DIN 30	DIN 29	DIN 28	DIN 27	DIN 26	DIN 25	DIN 24	DIN 23	DIN 22	DIN 21	DIN 20	DIN 19	DIN 18	DIN 17
G02W	DIN 48	DIN 47	DIN 46	DIN 45	DIN 44	DIN 43	DIN 42	DIN 41	DIN 40	DIN 39	DIN 38	DIN 37	DIN 36	DIN 35	DIN 34	DIN 33
G03W	DIN 64	DIN 63	DIN 62	DIN 61	DIN 60	DIN 59	DIN 58	DIN 57	DIN 56	DIN 55	DIN 54	DIN 53	DIN 52	DIN 51	DIN 50	DIN 49
G04W	DIN 116	DIN 115	DIN 114	DIN 113	DIN 112	DIN 111	DIN 110	DIN 109	DIN 108	DIN 107	DIN 106	DIN 105	DIN 104	DIN 103	DIN 102	DIN 101
G05W	DIN 132	DIN 131	DIN 130	DIN 129	DIN 128	DIN 127	DIN 126	DIN 125	DIN 124	DIN 123	DIN 122	DIN 121	DIN 120	DIN 119	DIN 118	DIN 117
G06W	DIN 148	DIN 147	DIN 146	DIN 145	DIN 144	DIN 143	DIN 142	DIN 141	DIN 140	DIN 139	DIN 138	DIN 137	DIN 136	DIN 135	DIN 134	DIN 133
G07W	DIN 164	DIN 163	DIN 162	DIN 161	DIN 160	DIN 159	DIN 158	DIN 157	DIN 156	DIN 155	DIN 154	DIN 153	DIN 152	DIN 151	DIN 150	DIN 149
G10W	DATA _TRIG				DCONV TRIG	LMIT OFF	MLT RST	OFS MOD	HAND IN8	HAND IN7	HAND IN6	HAND IN5	HAND IN4	HAND IN3	HAND IN2	HAND IN1
G11W	AL8-272	AL8-271	AL8-270	AL8-269	AL4-080	AL4-079	AL4-078	AL4-077	AL1-044	AL1-043	AL1-042	AL1-041	AL1-040	AL1-039	AL1-038	AL1-037
G12W					J_ MOV E	J_DI RECT	J_AXIS				J_COORD		J_SPEED		J_REMOTE	
G13W			BZ_ RST	SV OFF	BREAK	LOW_ SPD	CYCLE	STOP	EX_ SVON	RUN	ALM _RST	DO _RST	CYC _RST	STEP _RST	PRG_ RST	STROBE
G14W ~ G16W	Reserved area															
G17W	PANEL BREAK	PANEL RUN	PANEL SV OFF	PANEL EX_ SVON		PANEL DO _RST	PANEL PRG_ RST	PANEL ALM _RST								
G20W	DIN 316	DIN 315	DIN 314	DIN 313	DIN 312	DIN 311	DIN 310	DIN 309	DIN 308	DIN 307	DIN 306	DIN 305	DIN 304	DIN 303	DIN 302	DIN 301
G21W	DIN 332	DIN 331	DIN 330	DIN 329	DIN 328	DIN 327	DIN 326	DIN 325	DIN 324	DIN 323	DIN 322	DIN 321	DIN 320	DIN 319	DIN 318	DIN 317
G22W	DIN 348	DIN 347	DIN 346	DIN 345	DIN 344	DIN 343	DIN 342	DIN 341	DIN 340	DIN 339	DIN 338	DIN 337	DIN 336	DIN 335	DIN 334	DIN 333
G23W	DIN 364	DIN 363	DIN 362	DIN 361	DIN 360	DIN 359	DIN 358	DIN 357	DIN 356	DIN 355	DIN 354	DIN 353	DIN 352	DIN 351	DIN 350	DIN 349
G24W	DIN 416	DIN 415	DIN 414	DIN 413	DIN 412	DIN 411	DIN 410	DIN 409	DIN 408	DIN 407	DIN 406	DIN 405	DIN 404	DIN 403	DIN 402	DIN 401
G25W	DIN 432	DIN 431	DIN 430	DIN 429	DIN 428	DIN 427	DIN 426	DIN 425	DIN 424	DIN 423	DIN 422	DIN 421	DIN 420	DIN 419	DIN 418	DIN 417
G26W	DIN 448	DIN 447	DIN 446	DIN 445	DIN 444	DIN 443	DIN 442	DIN 441	DIN 440	DIN 439	DIN 438	DIN 437	DIN 436	DIN 435	DIN 434	DIN 433
G27W	DIN 464	DIN 463	DIN 462	DIN 461	DIN 460	DIN 459	DIN 458	DIN 457	DIN 456	DIN 455	DIN 454	DIN 453	DIN 452	DIN 451	DIN 450	DIN 449
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Robot controller main unit → TCmini (H000 ~ H27F)

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
H00W	DOUT 16	DOUT 15	DOUT 14	DOUT 13	DOUT 12	DOUT 11	DOUT 10	DOUT 9	DOUT 8	DOUT 7	DOUT 6	DOUT 5	DOUT 4	DOUT 3	DOUT 2	DOUT 1
H01W	DOUT 32	DOUT 31	DOUT 30	DOUT 29	DOUT 28	DOUT 27	DOUT 26	DOUT 25	DOUT 24	DOUT 23	DOUT 22	DOUT 21	DOUT 20	DOUT 19	DOUT 18	DOUT 17
H02W	DOUT 48	DOUT 47	DOUT 46	DOUT 45	DOUT 44	DOUT 43	DOUT 42	DOUT 41	DOUT 40	DOUT 39	DOUT 38	DOUT 37	DOUT 36	DOUT 35	DOUT 34	DOUT 33
H03W	DOUT 64	DOUT 63	DOUT 62	DOUT 61	DOUT 60	DOUT 59	DOUT 58	DOUT 57	DOUT 56	DOUT 55	DOUT 54	DOUT 53	DOUT 52	DOUT 51	DOUT 50	DOUT 49
H04W	DOUT 116	DOUT 115	DOUT 114	DOUT 113	DOUT 112	DOUT 111	DOUT 110	DOUT 109	DOUT 108	DOUT 107	DOUT 106	DOUT 105	DOUT 104	DOUT 103	DOUT 102	DOUT 101
H05W	DOUT 132	DOUT 131	DOUT 130	DOUT 129	DOUT 128	DOUT 127	DOUT 126	DOUT 125	DOUT 124	DOUT 123	DOUT 122	DOUT 121	DOUT 120	DOUT 119	DOUT 118	DOUT 117
H06W	DOUT 148	DOUT 147	DOUT 146	DOUT 145	OUT 144	DOUT 143	DOUT 142	DOUT 141	DOUT 140	DOUT 139	DOUT 138	DOUT 137	DOUT 136	DOUT 135	DOUT 134	DOUT 133
H07W	DOUT 164	DOUT 163	DOUT 162	DOUT 161	DOUT 160	DOUT 159	DOUT 158	DOUT 157	DOUT 156	DOUT 155	DOUT 154	DOUT 153	DOUT 152	DOUT 151	DOUT 150	DOUT 149
H10W	DATA _ACK	DATA _ERR				TCP ERR	MLT END	OFS END	HAND OUT8	HAND OUT7	HAND OUT6	HAND OUT5	HAND OUT4	HAND OUT3	HAND OUT2	HAND OUT1
H11W	SEQ FSW8	SEQ FSW7	SEQ FSW6	SEQ FSW5	SEQ FSW4	SEQ FSW3	SEQ FSW2	SEQ FSW1	SEQ PAR8	SEQ PAR7	SEQ PAR6	SEQ PAR5	SEQ PAR4	SEQ PAR3	SEQ PAR2	SEQ PAR1
H12W																
H13W		EXT ETHER	ALARM	BT_ ALM	CYC _ST	LOW _ST	CYC _END	AUTO RUN	SYS_ RDY	EXT 232C	EXT SIG	INT	TEACH	ACK	SV_ RDY	EMG_ ST
H14W ~ H17W	Reserved area															
H20W	DOUT 316	DOUT 315	DOUT 314	DOUT 313	DOUT 312	DOUT 311	DOUT 310	DOUT 309	DOUT 308	DOUT 307	DOUT 306	DOUT 305	DOUT 304	DOUT 303	DOUT 302	DOUT 301
H21W	DOUT 332	DOUT 331	DOUT 330	DOUT 329	DOUT 328	DOUT 327	DOUT 326	DOUT 325	DOUT 324	DOUT 323	DOUT 322	DOUT 321	DOUT 320	DOUT 319	DOUT 318	DOUT 317
H22W	DOUT 348	DOUT 347	DOUT 346	DOUT 345	OUT 344	DOUT 343	DOUT 342	DOUT 341	DOUT 340	DOUT 339	DOUT 338	DOUT 337	DOUT 336	DOUT 335	DOUT 334	DOUT 333
H23W	DOUT 364	DOUT 363	DOUT 362	DOUT 361	DOUT 360	DOUT 359	DOUT 358	DOUT 357	DOUT 356	DOUT 355	DOUT 354	DOUT 353	DOUT 352	DOUT 351	DOUT 350	DOUT 349
H24W	DOUT 416	DOUT 415	DOUT 414	DOUT 413	DOUT 412	DOUT 411	DOUT 410	DOUT 409	DOUT 408	DOUT 407	DOUT 406	DOUT 405	DOUT 404	DOUT 403	DOUT 402	DOUT 401
H25W	DOUT 432	DOUT 431	DOUT 430	DOUT 429	DOUT 428	DOUT 427	DOUT 426	DOUT 425	DOUT 424	DOUT 323	DOUT 422	DOUT 421	DOUT 420	DOUT 419	DOUT 418	DOUT 417
H26W	DOUT 448	DOUT 447	DOUT 446	DOUT 445	DOUT 444	DOUT 443	DOUT 442	DOUT 441	DOUT 440	DOUT 339	DOUT 438	DOUT 437	DOUT 436	DOUT 435	DOUT 434	DOUT 433
H27W	DOUT 464	DOUT 463	DOUT 462	DOUT 461	DOUT 460	DOUT 459	DOUT 458	DOUT 457	DOUT 456	DOUT 355	DOUT 454	DOUT 453	DOUT 452	DOUT 451	DOUT 450	DOUT 449
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

4.4 Edge Relay

A total of 256 edge relays (E000 ~ E17F) are available.

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
E00W	E00F															E000
E01W																
E02W																
E03W																
E04W																
E05W																
E06W																
E07W																
E10W	E10F															E100
E11W																
E12W																
E13W																
E14W																
E15W																
E16W																
E17W																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

E*** Once the condition is ON, the edge relay turns on only for one (1) scan time. To turn on the relay again, set OFF the condition once, then ON.

4.5 Latch Relay

A total of 128 latch relays (L000 ~ L07F) are available.

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
L00W	L00F															L000
L01W																
L02W																
L03W																
L04W																
L05W																
L06W																
L07W																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

L*** Once the condition is ON, the latch relay remains on until the reset input is ON.

4.6 Timer/Counter

T/C000 ~ T/C27F are the common relays of the timer and counter.
Relay T000 used as the timer cannot be used as the C000 counter. Select either one.

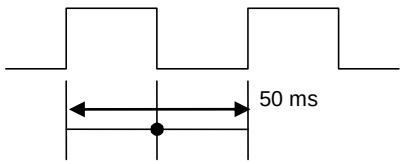
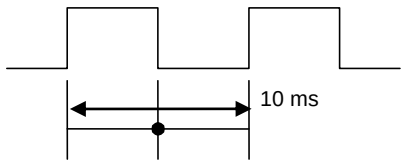
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
T/C00W	T00F															T000
T/C01W						Timer (100 ms) or counter										
T/C02W																
T/C03W																
T/C04W	T04F															T040
T/C05W																
T/C06W																
T/C07W																
T/C08W																
T/C09W						Timer (10 ms) or counter										
T/C10W																
T/C11W																
T/C12W																
T/C13W																
T/C14W																
T/C15W																
T/C16W																
T/C17W																
T/C20W	T20F															T200
T/C21W																
T/C22W																
T/C23W						Timer (100 ms) or counter										
T/C24W																
T/C25W																
T/C26W																
T/C27W																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

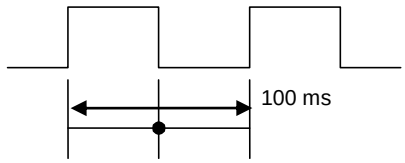
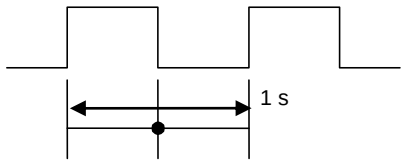
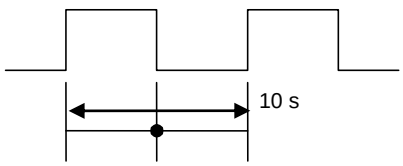
T*** 100 msec timer/10 msec timer
Once the condition is ON, the timer relay decrements a value set on the timer every 100 ms or 10 ms. When the value has reached zero (0), the timer turns on.

C*** The counter relay turns on when the number of pulses equal to a value set on the counter is input.
When the counter reset input turns on, the counter value is reset to the set value.

4.7 Special Auxiliary Relay

A relay whose address starts with A signifies the special auxiliary relay.

Register	Address	Descriptions	
Operation flag	A000	[Carry flag]: Turns on when the result of arithmetic operation includes a carry or borrow.	
	A002	[Overflow flag]: Turns on when the result of arithmetic operation includes an overflow.	
	A006	[Zero flag]: Turns on when the result of arithmetic operation is zero (0).	
	A007	[Sign flag]: Turns on when the MSB of arithmetic operation result is "1".	
Alarm flag	A016	Fuse blowout detection flag.	
Scan time	A03L	The scan time is displayed in BIN code and in units of "ms" under the byte register address (A03L).	
50 ms clock pulse	A038	50 ms	50 ms clock pulse The reference clock is 50 ms (A038). 
	A039	100 ms	
	A03A	200 ms	
	A03B	400 ms	
	A03C	800 ms	
	A03D	1600 ms	
	A03E	3200 ms	
	A03F	6400 ms	
10 ms clock pulse	A040	10 ms	10 ms clock pulse The reference clock is 10 ms (A040). 
	A041	20 ms	
	A042	40 ms	
	A043	80 ms	
	A044	160 ms	
	A045	320 ms	
	A046	640 ms	
	A047	1280 ms	

Register	Address	Descriptions	
100 ms clock pulse	A048	100 ms	100 ms clock pulse The reference clock is 100 ms (A048). 
	A049	200 ms	
	A04A	400 ms	
	A04B	800 ms	
	A04C	1600 ms	
	A04D	3200 ms	
	A04E	6400 ms	
	A04F	12800 ms	
1 s clock pulse	A050	1 s	1 s clock pulse The reference clock is 1 s (A050). 
	A051	2 s	
	A052	4 s	
	A053	8 s	
	A054	16 s	
	A055	32 s	
	A056	64 s	
	A057	128 s	
10 s clock pulse	A058	10 s	10 s clock pulse The reference clock is 10 s (A058). 
	A059	20 s	
	A05A	40 s	
	A05B	80 s	
	A05C	160 s	
	A05D	320 s	
	A05E	640 s	
	A05F	1280 s	

Section 5 Registers

A total of 512 registers (D000 ~ D37F) are available for the TCmini.

5.1 Data Register

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D00*	D00F															D000
D01*																
D02*								Data registers								
D03*																
D04*																
D05*																
D06*																
D07*																
D10*	D10F															D100
D11*																
D12*								Data registers								
D13*																
D14*																
D15*																
D16*																
D17*																
D20*	D20F															D200
D21*																
D22*								Data registers								
D23*																
D24*																
D25*																
D26*																
D27*																
D30*	D30F															D300
D31*																
D32*																
D33*							Data registers (backup)									
D34*																
D35*																
D36*																
D37*																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

The data register is a word length (16-bit) register which cannot be specified as the byte (8-bit) register.
This is a word length (16-bit) which cannot be specified as the byte register.
The operation result is written as the destination in the user's program.
The operation result can be used as the source in the user's program.

Each time the data of D060 ~ D11F has been changed, it is written to the EEPROM and held there. Writing of up to one (1) million counts is possible. When using this register as the job register, take careful precautions. (Each time the data has been changed, writing of EEPROM is executed.)

5.2 Interface Register

The interface register serves as the interface area for transferring data with the main unit of the robot controller.

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D40*	PLC SS R08	PLC SS R07	PLC SS R06	PLC SS R05	PLC SS R04	PLC SS R03	PLC SS R02	PLC SS R01	PLC DATA R8	PLC DATA R7	PLC DATA R6	PLC DATA R5	PLC DATA R4	PLC DATA R3	PLC DATA R2	PLC DATA R1
D41*	PLC SL R08H	PLC SL R08L	PLC SL R07H	PLC SL R07L	PLC SL R06H	PLC SL R06L	PLC SL R05H	PLC SL R05L	PLC SL R 04H	PLC SL R04L	PLC SL R03H	PLC SL R03L	PLC SL R02H	PLC SL R02L	PLC SL R01H	PLC SL R01L
D42*	Reserved area															
D43*																
D44*	Reserved area															
D45*																
D46*	DATA_CMD															
D47*																USER
D50*	Reserved area															
D51*																
D52*																
D53*																
D54*																
D55*																
D56*																
D57*																
D60*	PLC SS W08	PLC SS W07	PLC SS W06	PLC SS W05	PLC SS W04	PLC SS W03	PLC SS W02	PLC SS W01	PLC DATA W8	PLC DATA W7	PLC DATA W6	PLC DATA W5	PLC DATA W4	PLC DATA W3	PLC DATA W2	PLC DATA W1
D61*	PLC SL W08H	PLC SL W08L	PLC SL W07H	PLC SL W07L	PLC SL W06H	PLC SL W06L	PLC SL W05H	PLC SL W05L	PLC SL W04H	PLC SL W04L	PLC SL W03H	PLC SL W03L	PLC SL W02H	PLC SL W02L	PLC SL W01H	PLC SL W01L
D62*	Reserved area								PSN_ W8	PSN_ W7	PSN_ W6	PSN_ W5	PSN_ W4	PSN_ W3	PSN_ W2	PSN_ W1
D63*									PSN_ J8	PSN_ J7	PSN_ J6	PSN_ J5	PSN_ J4	PSN_ J3	PSN_ J2	PSN_ J1
D64*									TRQ_ J8	TRQ_ J7	TRQ_ J6	TRQ_ J5	TRQ_ J4	TRQ_ J3	TRQ_ J2	TRQ_ J1
D65*																
D66*	DATA_RESP															
D67*	AL10	AL09	AL08	AL07	AL06	AL05	AL04	AL03	AL02	AL01	ALNO	STEP				
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

For details, see Section 6.

5.3 Timer/Counter Current Value Register

P000 ~ P05F are the timer/counter current value registers. Current values can be referred to and set in the sequence program.

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
P00*																
P01*					Timer current values (100 ms)/counter set values											
P02*																
P03*																
P04*																
P05*																
P06*																
P07*																
P10*																
P11*					Timer current values (10 ms)/counter current values											
P12*																
P13*																
P14*																
P15*																
P16*																
P17*																
P20*																
P21*																
P22*																
P23*					Timer current values (10 ms)/counter current values											
P24*																
P25*																
P26*																
P27*																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

P*** If used as the source in the user's program, the timer/counter current value can be read. (Backward timer and counter)
If written as the destination in the user's program during counting, the current value can be changed.
For the register used as the timer current value, the set value is equal to the current value at power ON, fall time (RUN → STOP) or timer OFF.
This is a word length (16-bit) register which cannot be specified as the byte register.

5.4 Timer/Counter Set Value Register

V000 ~ V05F are the timer/counter set value registers. Current values can be referred to and set in the sequence program.

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
V00*																
V01*					Timer set values (100 ms)/counter set values											
V02*																
V03*																
V04*																
V05*																
V06*																
V07*																
V10*																
V11*					Timer current values (10 ms)/counter current values											
V12*																
V13*																
V14*																
V15*																
V16*																
V17*																
V20*																
V21*																
V22*																
V23*					Timer current values (10 ms)/counter current values											
V24*																
V25*																
V26*																
V27*																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

V*** If used as the destination in the user's program, the set value can be changed.
The area not used as the timer/counter can be used as the register.
This is a word length (16-bit) register which cannot be specified as the byte register.

Section 6 Robot Interface

Transfer of signals between the main unit of robot controller and TCmini is all performed through the interface relays and interface registers.
Each interface register has signal input and output directions.

6.1 TCmini → Main Unit of Robot Controller

TCmini → Main unit of robot controller (G000 ~ G27F)

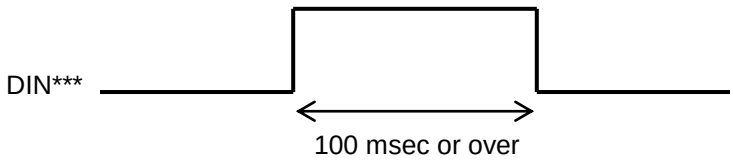
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
G00W	DIN 16	DIN 15	DIN 14	DIN 13	DIN 12	DIN 11	DIN 10	DIN 9	DIN 8	DIN 7	DIN 6	DIN 5	DIN 4	DIN 3	DIN 2	DIN 1
G01W	DIN 32	DIN 31	DIN 30	DIN 29	DIN 28	DIN 27	DIN 26	DIN 25	DIN 24	DIN 23	DIN 22	DIN 21	DIN 20	DIN 19	DIN 18	DIN 17
G02W	DIN 48	DIN 47	DIN 46	DIN 45	DIN 44	DIN 43	DIN 42	DIN 41	DIN 40	DIN 39	DIN 38	DIN 37	DIN 36	DIN 35	DIN 34	DIN 33
G03W	DIN 64	DIN 63	DIN 62	DIN 61	DIN 60	DIN 59	DIN 58	DIN 57	DIN 56	DIN 55	DIN 54	DIN 53	DIN 52	DIN 51	DIN 50	DIN 49
G04W	DIN 116	DIN 115	DIN 114	DIN 113	DIN 112	DIN 111	DIN 110	DIN 109	DIN 108	DIN 107	DIN 106	DIN 105	DIN 104	DIN 103	DIN 102	DIN 101
G05W	DIN 132	DIN 131	DIN 130	DIN 129	DIN 128	DIN 127	DIN 126	DIN 125	DIN 124	DIN 123	DIN 122	DIN 121	DIN 120	DIN 119	DIN 118	DIN 117
G06W	DIN 148	DIN 147	DIN 146	DIN 145	DIN 144	DIN 143	DIN 142	DIN 141	DIN 140	DIN 139	DIN 138	DIN 137	DIN 136	DIN 135	DIN 134	DIN 133
G07W	DIN 164	DIN 163	DIN 162	DIN 161	DIN 160	DIN 159	DIN 158	DIN 157	DIN 156	DIN 155	DIN 154	DIN 153	DIN 152	DIN 151	DIN 150	DIN 149
G10W	DATA _TRIG				DCO NV TRIG	LMIT OFF	MLT RST	OFS MOD	HAND IN8	HAND IN7	HAND IN6	HAND IN5	HAND IN4	HAND IN3	HAND IN2	HAND IN1
G11W	AL8-272	AL8-271	AL8-270	AL8-269	AL4-080	AL4-079	AL4-078	AL4-077	AL1-044	AL1-043	AL1-042	AL1-041	AL1-040	AL1-039	AL1-038	AL1-037
G12W					J_ MOVE	J_DI RECT	J_AXIS				J_COORD		J_SPEED		J_REMOTE	
G13W			BZ_ RST	SV OFF	BREAK	LOW_ SPD	CYCLE	STOP	EX_ SVON	RUN	ALM _RST	DO _RST	CYC _RST	STEP _RST	PRG_ RST	STROBE
G14W ~ G16W																
G17W	PANEL BREAK	PANEL RUN	PANEL SV OFF	PANEL EX_ SVON		PANEL DO _RST	PANEL PRG_ RST	PANEL ALM _RST								
G20W	DIN 316	DIN 315	DIN 314	DIN 313	DIN 312	DIN 311	DIN 310	DIN 309	DIN 308	DIN 307	DIN 306	DIN 305	DIN 304	DIN 303	DIN 302	DIN 301
G21W	DIN 332	DIN 331	DIN 330	DIN 329	DIN 328	DIN 327	DIN 326	DIN 325	DIN 324	DIN 323	DIN 322	DIN 321	DIN 320	DIN 319	DIN 318	DIN 317
G22W	DIN 348	DIN 347	DIN 346	DIN 345	DIN 344	DIN 343	DIN 342	DIN 341	DIN 340	DIN 339	DIN 338	DIN 337	DIN 336	DIN 335	DIN 334	DIN 333
G23W	DIN 364	DIN 363	DIN 362	DIN 361	DIN 360	DIN 359	DIN 358	DIN 357	DIN 356	DIN 355	DIN 354	DIN 353	DIN 352	DIN 351	DIN 350	DIN 349
G24W	DIN 416	DIN 415	DIN 414	DIN 413	DIN 412	DIN 411	DIN 410	DIN 409	DIN 408	DIN 407	DIN 406	DIN 405	DIN 404	DIN 403	DIN 402	DIN 401
G25W	DIN 432	DIN 431	DIN 430	DIN 429	DIN 428	DIN 427	DIN 426	DIN 425	DIN 424	DIN 423	DIN 422	DIN 421	DIN 420	DIN 419	DIN 418	DIN 417
G26W	DIN 448	DIN 447	DIN 446	DIN 445	DIN 444	DIN 443	DIN 442	DIN 441	DIN 440	DIN 439	DIN 438	DIN 437	DIN 436	DIN 435	DIN 434	DIN 433
G27W	DIN 464	DIN 463	DIN 462	DIN 461	DIN 460	DIN 459	DIN 458	DIN 457	DIN 456	DIN 455	DIN 454	DIN 453	DIN 452	DIN 451	DIN 450	DIN 449
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

G000 ~ G27F:
Signifies an exclusive output relay which outputs a signal to the main unit of the robot controller.
It serves as the destination of coil and data register in the sequence program, where operation result is written.
The operation result can be used as the source of contact and data register in the sequence program.
The operation result ON/OFF is transferred to the output device at I/O processing of every scan cycle.

Relays in the shaded areas are reserved for future extension of the system functions. Values are undecided and cannot be used.

The five (5) signals of STOP, CYCLE, LOW_SPD, BREAK and SVOFF are valid when they are low. Unless the signals are processed legally, the robot cannot be moved.
* G108 to G10B are optional signals for specific customers.

DIN1 ~ 64, DIN101 ~ 164, DIN301 ~ 364, DIN401 ~ 464

Address	**G000 ~ G07F, G200 ~ R27F**
Direction	TCmini → Main unit of robot controller
Function	Interface relay corresponding to the robot's DIN command. In the WAIT DIN (1) command, it waits until DIN1 (G000) turns on.
Timing chart	 The width of a signal to be output should be 100 msec or over. If the signal width is too short, change in signal status may not be identified by the DIN command in some circumstances.
Remarks	Though the relay name is DIN (digital input signal), it is the output relay.

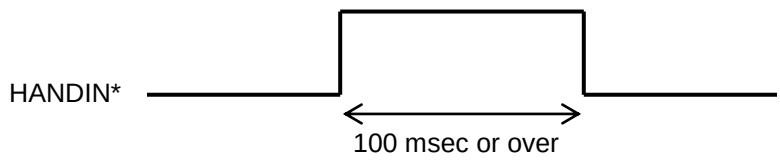
HANDIN1 ~ 8 (Hand Input Signal)

Address	**G100 ~ G107**
Direction	TCmini → Main unit of robot controller
Function	Interface relay corresponding to the robot's DIN command. Except for the exclusive monitor display for the hand (Para. 12.3 of Operator's Manual), the function is the same as DIN***.

Each signal corresponds to the DIN command as shown below.

Signal name	Relay address	Command
HANDIN1	G100	DIN201
HANDIN2	G101	DIN202
HANDIN3	G102	DIN203
HANDIN4	G103	DIN204
HANDIN5	G104	DIN205
HANDIN6	G105	DIN206
HANDIN7	G106	DIN207
HANDIN8	G107	DIN208

Timing chart



The width of a signal to be output should be 100 msec or over. If the signal width is too short, change in signal status may not be identified by the DIN command in some circumstances.

Remarks

Though the relay name is HANDIN (hand input signal), it is the output relay.

DATA_TRIG (Request trigger signal)

Address	**G10F**
Direction	TCmini → Robot main section
Function	This is the trigger signal for the Simple PLC data communication function. Set this signal to ON after DATA_CMD is set. Also, after the response (DATA_RESP) is read, set this signal to OFF.
Remarks	For details on the Simple PLC data communication function, see chapter 12.

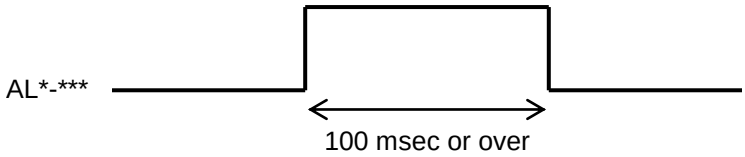
AL8–269 ~ 272 (Alarm of Level 8)
AL4–077 ~ 080 (Alarm of Level 4)
AL1–037 ~ 044 (Alarm of Level 1)

Address	AL8–269 ~ 272: **G11C ~ G11F** AL4–077 ~ 080: **G118 ~ G11B** AL1–037 ~ 044: **G110 ~ G117**
Direction	TCmini → Main unit of robot controller
Function	Makes the robot in an ALARM state from the TCmini.

Signal name	Relay address	Robot status
AL1–037	G110	Message display
AL1–038	G111	
AL1–039	G112	
AL1–040	G113	
AL1–041	G114	
AL1–042	G115	
AL1–043	G116	
AL1–044	G117	
AL4–077	G118	Motion stop
AL4–078	G119	
AL4–079	G11A	
AL4–080	G11B	
AL8–269	G11C	Emergency stop
AL8–270	G11D	
AL8–271	G11E	
AL8–272	G11F	

When alarm messages for respective alarms are registered in user parameters beforehand, any message can be displayed. For details, see the User Parameter Manual.

Timing chart



The width of a signal to be output should be 100 msec or over. If the signal width is too short, appropriate alarm may not be identified in some circumstances.

Remarks

Connect a signal relating to safety (such as emergency stop switch) to an external emergency stop signal processed by hardware (EMS1B ~ EMS1C, EMS2B ~ EMS2C).

J_MOVE (Jog operation)

J_REMOTE (JOG_REMOTE selection)

J_SPEED (JOG_SPEED selection)

J_COORD (JOG_COORDINATE selection)

J_DIRECT (JOG_Direction selection)

J_AXIS (JOG_Guiding axis selection)

Address

J_MOVE: G12B
J_REMOTE: G120,G121
J_SPEED: G122,G123
J_COORD: G124,G125
J_DIRECT: G12A
J_AXIS: G126 to G129

Direction

TCmini → Robot main section

Function

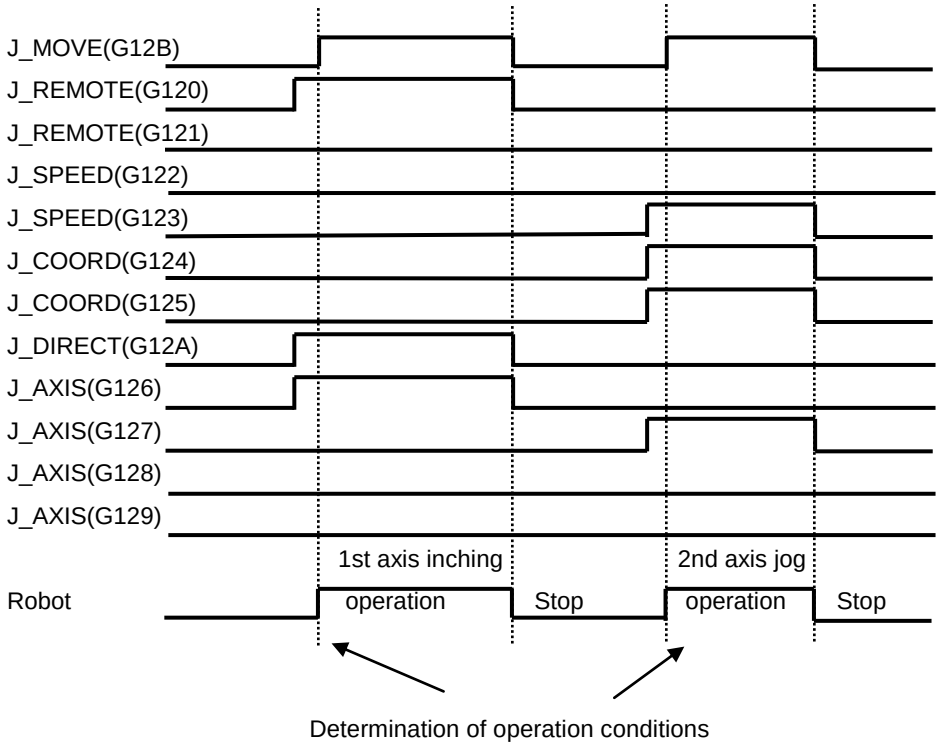
Jog operation is performed from TCmini.

This signal is enabled in EXT.SIG mode only.

While the J_MOVE signal (G12B) is ON, the axis specified in the conditions below performs the jog operation.
The operating conditions and axis selection signal are determined at the rising of J_MOVE.

Signal name / Value	J_REMOTE (Guiding method)	J_SPEED (Guiding speed)	J_COORD (Guiding coordinates)	J_AXIS (Axis)		J_DIRECT (Direction)	J_MOVE (Operation)
				SCARA	6-axis		
00(0)	JOG	LOW	JOINT	Not used	Not used	Negative direction	STOP
01(1)	INCHING	MEDIUM	TOOL	1-axis (X)	1-axis (X)	Positive direction	MOVE
10(2)	Not used	HIGH	WORK	2-axis (Y)	2-axis (Y)		
11(3)	Not used	Not used	WORLD	3-axis (Z)	3-axis (Z)		
0100(4)				4-axis (C)	4-axis (A)		
0101(5)				5-axis (T)	5-axis (B)		
0110(6)				Not used	6-axis (C)		
0111(7)				Not used	Not used		
1000(8)				Not used	Not used		

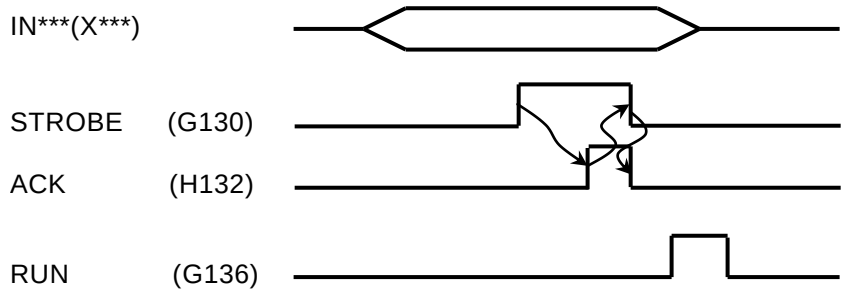
Timechart



Remarks

Operation does not change even if the operation conditions or axis selection signal is changed while J_MOVE is ON.
Be sure that you are always near the emergency stop switch when performing operation of this signal so that the emergency stop switch can be pressed immediately at any time.

STROBE (Strobe Signal)

Address	**G130**
Direction	TCmini → Main unit of robot controller
Function	Selects a program registered in the robot controller from the TCmini. This signal is valid only in the EXT.SIG mode. The program number selected uses any consecutive "n" external digital input signals (max. four (4) signals) and is input by code.
Timing chart	 With the start of STROBE, the above digital signals are read to select an appropriate program. When the program has been selected, the ACK signal turns on. Set ON the RUN signal and execute the program.
Remarks	DO NOT input the STROBE signal together with the PRG_RST, CYC_RST, STEP_RST or DO_RST signal. Because the ACK signal is used in common, only the first signal which is input becomes valid and all other signals are invalid. If a file other than the current file is selected, the program is reset to step 1 and all values of variables are also reset. The program name and external select signal should be set in the USER.PAR file.

```
[U07] Specify signal for EXTSELECT
   {Signal No } (1 – 64,101 – 164)
   {Bit length } (1 – 4)
= 2   3
```

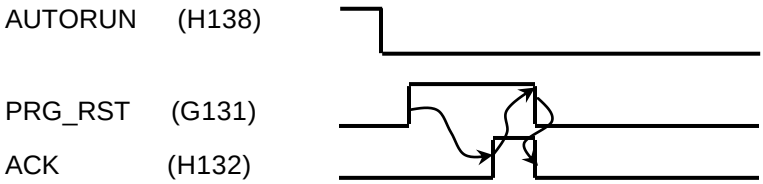
The selected program is set in the EXTRNSEL.PAR file.

```
*** [00-0F] ***
= "PROG0"
= "PROG1"
= "PROG2"
= "PROG3"
```

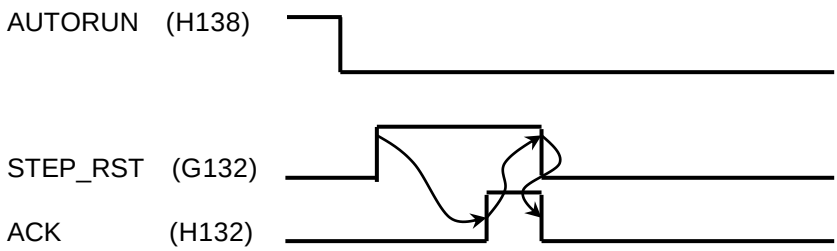
When the setting is as shown above, a file is selected by the three (3) signals, starting from DIN2 (R201 ~ R203).

DIN4	DIN3	DIN2	
OFF	OFF	ON	Selection of PROG1
OFF	ON	ON	Selection of PROG3

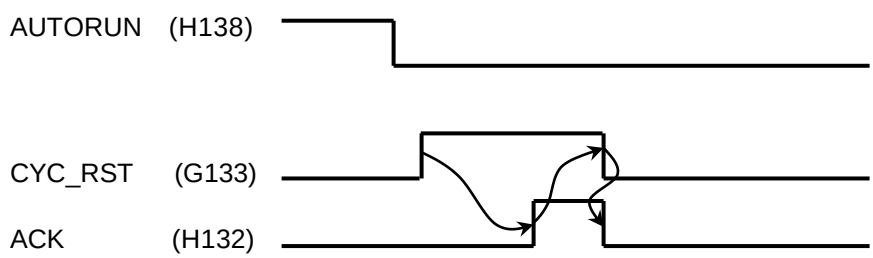
PRG_RST (Program Reset)

Address	**G131**
Direction	TCmini → Main unit of robot controller
Function	Resets an interrupted program to step 1 from the TCmini. Also resets the value of each variable to "0". This signal is operative only in the EXT.SIG mode.
Timing chart	
Remarks	DO NOT input the PRG_RST signal together with the STROBE, CYC_RST, STEP_RST or DO_RST signal. Because the ACK signal is used in common, only the first signal which is input becomes valid and all other signals are invalid.

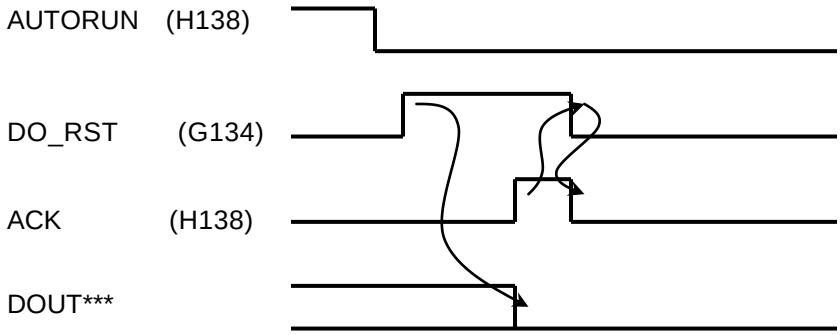
STEP_RST (Step Reset)

Address	**G132**
Direction	TCmini → Main unit of robot controller
Function	Resets an interrupted program to step 1 from the TCmini. The value of each variable used in the program remains intact. This signal is operative only in the EXT.SIG mode.
Timing chart	
Remarks	DO NOT input the STEP_RST signal together with the STROBE, PRG_RST, CYC_RST, or DO_RST signal. Because the ACK signal is used in common, only the first signal which is input becomes valid and all other signals are invalid.

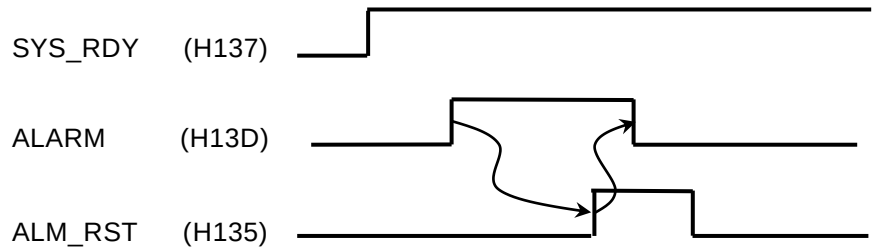
CYC_RST (Cycle Reset)

Address	**G133**
Direction	TCmini → Main unit of robot controller
Function	Resets an interrupted program to the step of RCYCLE label from the TCmini. The value of each variable used in the program remains intact. This signal is operative only in the EXT.SIG mode.
Timing chart	
Remarks	DO NOT input the CYC_RST signal together with the STROBE, PRG_RST, STEP_RST, or DO_RST signal. Because the ACK signal is used in common, only the first signal which is input becomes valid and all other signals are invalid.

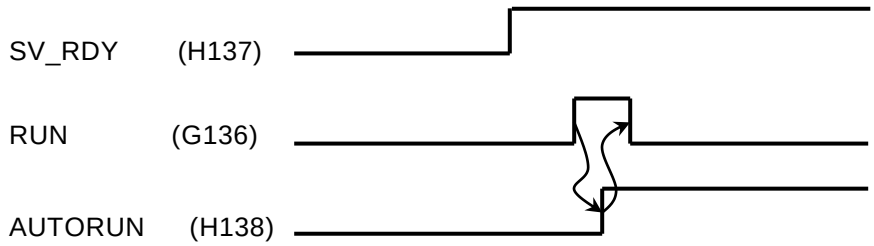
DO_RST (Output Signal Reset)

Address	**G134**
Direction	TCmini → Main unit of robot controller
Function	Interface relay which resets the digital output signals (DOUT1 ~ DOUT64, DOUT101 ~ DOUT164) from the TCmini. Once reset, all of R400 ~ R47F turn off. This signal is operative only in the EXT.SIG mode.
Timing chart	
Remarks	DO NOT input the DO_RST signal together with the STROBE, PRG_RST, CYC_RST or STEP_RST signal. Because the ACK signal is used in common, only the first signal which is input becomes valid and all other signals are invalid. The hand control signals (HANDOUT1 ~ 8) are not reset at all.

ALM_RST (Alarm Reset)

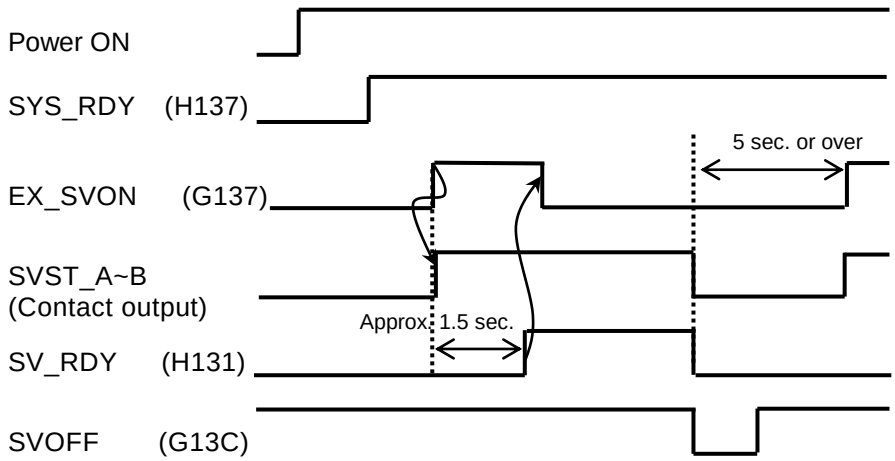
Address	**G135**
Direction	TCmini → Main unit of robot controller
Function	Resets from the TCmini an alarm which occurred in the robot controller. This signal is operative only in the EXT.SIG mode.
Timing chart	
Remarks	If an alarm of the emergency stop level in which servo ON is not possible, or the "emergency stop ON" is output, alarm reset by ALM_RST is not allowed.

RUN (Startup)

Address	**G136**
Direction	TCmini → Main unit of robot controller
Function	Starts a program registered in the robot controller from the TCmini to execute an automatic cycle operation. This signal is operative only in the EXT.SIG mode.
Timing chart	
Remarks	With the start of RUN, automatic operation starts. Set ON the RUN signal only after the SV_RDY signal is ON and the robot is ready to operate. If the RUN signal is set ON while the SV_RDY signal is OFF, the RUN signal is neglected.

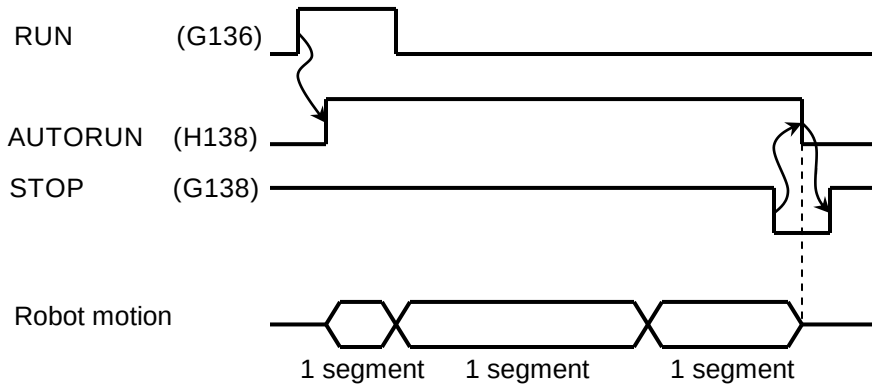
EX_SVON (External Servo ON)

Address	**G137**
Direction	TCmini → Main unit of robot controller
Function	Turns on the servo driver power from the TCmini. Once the servo power is turned on, it is maintained even after this signal turns off. This signal is operative only in the EXT.SIG mode.
Timing chart	



Remarks	It takes about 1.5 seconds from the servo ON to the time when the robot is actually ready to work (i.e., the time when the SV_RDY signal is ON). When setting on the RUN signal, etc., wait until SV_RDY turns on. To set ON this signal again just after the servo OFF, wait at least five (5) seconds.

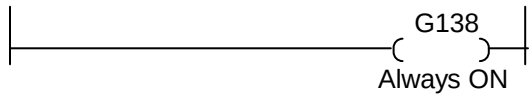
STOP (Stop)

Address	**G138**
Direction	TCmini → Main unit of robot controller
Function	Stops executing a program registered in the robot controller from the TCmini. The program stops only after the current motion command has been executed while this signal was OFF (i.e., low). After the stop, the robot enters a STOP (CONT) state. The robot will not work even if this signal is set ON after stop of the robot motion. This signal is always operative, irrespective of the master mode selected by means of the MODE switch.
Timing chart	

Note: Duration covering from the start of one motion command to just before the start of next motion command is called the"1 segment".

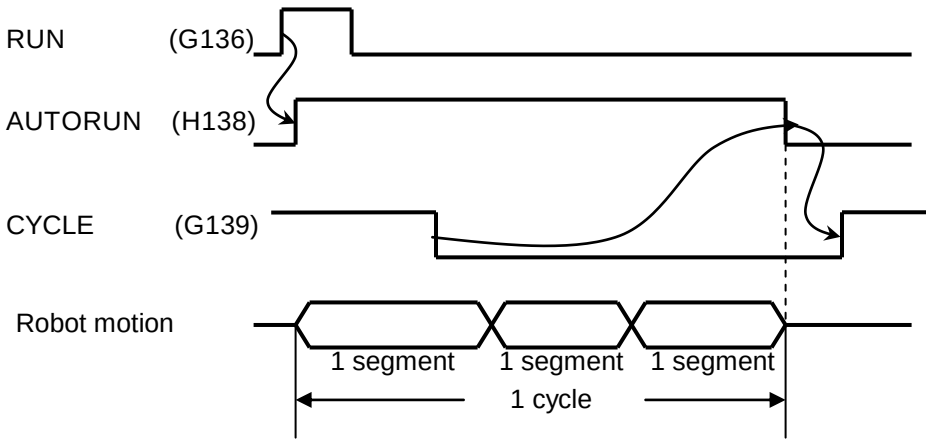
Remarks

1. When the RUN command is executed after cancel of the stop, the program restarts from the step next to the interrupted step.
2. RUN signal input is ineffective at the input of stop signal.
3. Unless this signal is used, always set it ON in the sequence circuit.



CYCLE (Cycle Operation Mode)

Address	**G139**
Direction	TCmini → Main unit of robot controller
Function	Stops from the TCmini a program registered in the robot controller after current one (1) cycle operation has been executed during automatic operation. When this signal is OFF (i.e., low), cycle stop is affected. This signal is operative only in the EXT.SIG mode.
Timing chart	

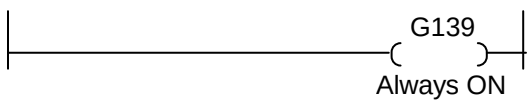


* Duration covering from the start of one motion command to just before the start of next motion command is called the "1 segment".

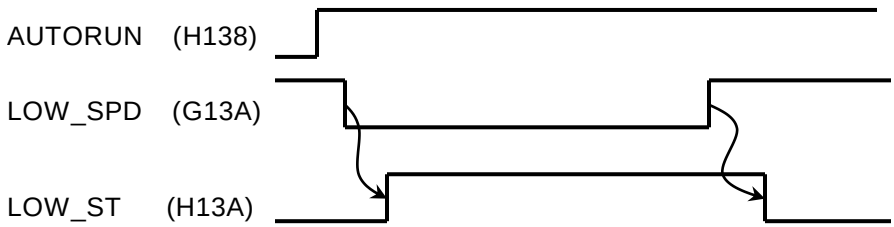
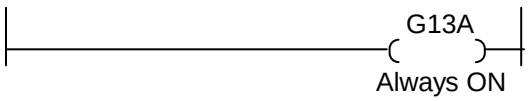
* Duration from the top of the main program to the END command is called the "1 cycle".

Remarks

1. When the RUN command is executed after cancel of the cycle operation mode, the continuous operation starts from the cycle next to the interrupted cycle.
2. Unless this signal is used, always set it ON in the sequence circuit.



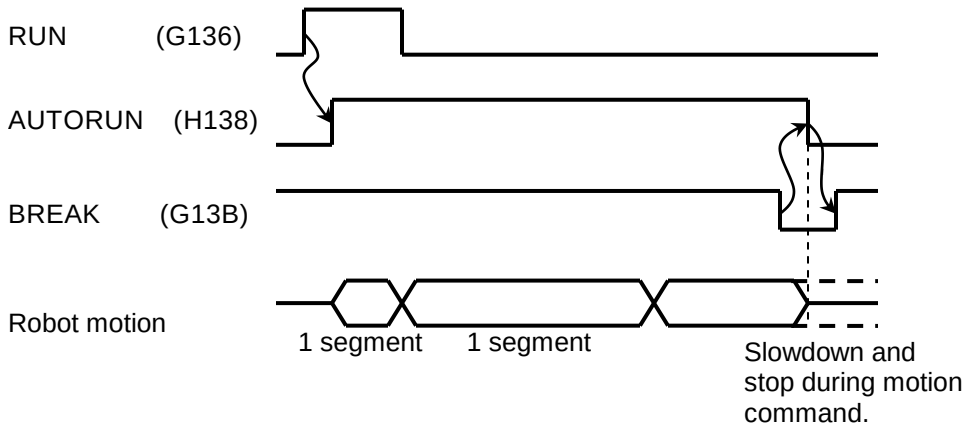
LOW_SPD (Low Speed Command)

Address	**G13A**
Direction	TCmini → Main unit of robot controller
Function	Causes the robot operation speed to low speed from the TCmini. The robot operates at a low speed while this signal is OFF (i.e., low). The robot operation speed in the low speed mode can be set by means of the parameter. (Initial set value: 25%) When this signal is OFF, the override value changes. When the signal is ON, the previously set value takes effect again. This signal is always effective, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	1. During the low speed command, an override set in the parameter beforehand is affected on all motion speeds. 2. Unless this signal is used, always set it ON in the sequence circuit. 

BREAK (Deceleration and Stop)

Address	**G13B**
Direction	TCmini → Main unit of robot controller
Function	Stops the robot motion from the TCmini. The robot slows down and stops at the same time that this signal is OFF (i.e., low). After the stop, the robot enters a STOP (RETRY) state. The robot will not work even if this signal is set ON after stop of the robot motion. This signal is always effective, irrespective of the master mode selected by means of the MODE switch.

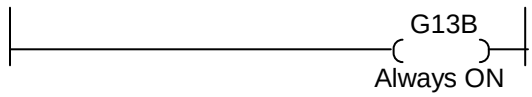
Timing chart



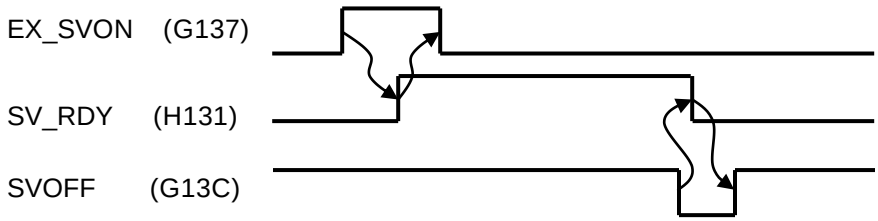
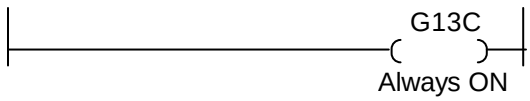
* Duration covering from the start of one motion command to just before the start of next motion command is called the"1 segment".

Remarks

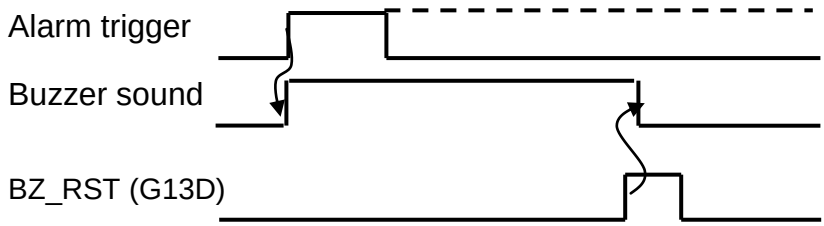
1. When the RUN command is executed after cancel of the break, the program restarts from the step next to the interrupted step.
2. RUN signal input is ineffective at the input of BREAK signal.
3. Unless this signal is used, always set it ON in the sequence circuit.



SVOFF (Servo OFF)

Address	**G13C**
Direction	TCmini → Main unit of robot controller
Function	Turns off the servo driver main power from the TCmini. While this signal is OFF (i.e, low), the servo power is turned off. This signal is always effective, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	1. While this signal is OFF, the servo power cannot be turned on in any mode. 2. Unless this signal is used, always set it ON in the sequence circuit. 

BZ_RST (Buzzer Reset)

Address	**G13D**
Direction	TCmini → Main unit of robot controller
Function	Turns off the buzzer sound from TCmini when an alarm occurs This signal is always valid regardless of the master mode selected by the MODE switch.
Timing chart	
Remarks	1. While this signal is ON, the alarm sound is not emitted even if a new alarm occurs.

6.2 Main Unit of Robot Controller → TCmini

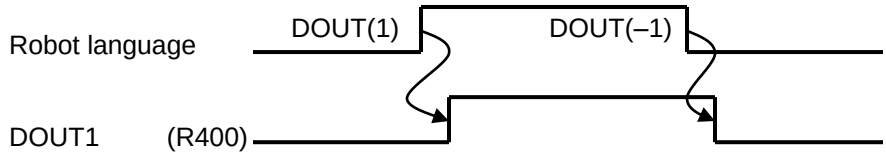
Main unit of robot controller → TCmini (H000 ~ H27F)

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
H00W	DOUT 16	DOUT 15	DOUT 14	DOUT 13	DOUT 12	DOUT 11	DOUT 10	DOUT 9	DOUT 8	DOUT 7	DOUT 6	DOUT 5	DOUT 4	DOUT 3	DOUT 2	DOUT 1
H01W	DOUT 32	DOUT 31	DOUT 30	DOUT 29	DOUT 28	DOUT 27	DOUT 26	DOUT 25	DOUT 24	DOUT 23	DOUT 22	DOUT 21	DOUT 20	DOUT 19	DOUT 18	DOUT 17
H02W	DOUT 48	DOUT 47	DOUT 46	DOUT 45	DOUT 44	DOUT 43	DOUT 42	DOUT 41	DOUT 40	DOUT 39	DOUT 38	DOUT 37	DOUT 36	DOUT 35	DOUT 34	DOUT 33
H03W	DOUT 64	DOUT 63	DOUT 62	DOUT 61	DOUT 60	DOUT 59	DOUT 58	DOUT 57	DOUT 56	DOUT 55	DOUT 54	DOUT 53	DOUT 52	DOUT 51	DOUT 50	DOUT 49
H04W	DOUT 116	DOUT 115	DOUT 114	DOUT 113	DOUT 112	DOUT 111	DOUT 110	DOUT 109	DOUT 108	DOUT 107	DOUT 106	DOUT 105	DOUT 104	DOUT 103	DOUT 102	DOUT 101
H05W	DOUT 132	DOUT 131	DOUT 130	DOUT 129	DOUT 128	DOUT 127	DOUT 126	DOUT 125	DOUT 124	DOUT 123	DOUT 122	DOUT 121	DOUT 120	DOUT 119	DOUT 118	DOUT 117
H06W	DOUT 148	DOUT 147	DOUT 146	DOUT 145	OUT 144	DOUT 143	DOUT 142	DOUT 141	DOUT 140	DOUT 139	DOUT 138	DOUT 137	DOUT 136	DOUT 135	DOUT 134	DOUT 133
H07W	DOUT 164	DOUT 163	DOUT 162	DOUT 161	DOUT 160	DOUT 159	DOUT 158	DOUT 157	DOUT 156	DOUT 155	DOUT 154	DOUT 153	DOUT 152	DOUT 151	DOUT 150	DOUT 149
H10W	DATA _ACK	DATA _ERR				TCP ERR	MLT END	OFS END	HAND OUT8	HAND OUT7	HAND OUT6	HAND OUT5	HAND OUT4	HAND OUT3	HAND OUT2	HAND OUT1
H11W	SEQ FSW8	SEQ FSW7	SEQ FSW6	SEQ FSW5	SEQ FSW4	SEQ FSW3	SEQ FSW2	SEQ FSW1	SEQ PAR8	SEQ PAR7	SEQ PAR6	SEQ PAR5	SEQ PAR4	SEQ PAR3	SEQ PAR2	SEQ PAR1
H12W																
H13W		EXT ETHER	ALAR M	BT_ ALM	CYC _ST	LOW _ST	CYC _END	AUTO RUN	SYS_ RDY	EXT 232C	EXT SIG	INT	TEAC H	ACK	SV_ RDY	EMG_ ST
H14W ~ H17W	Reserved area															
H20W	DOUT 316	DOUT 315	DOUT 314	DOUT 313	DOUT 312	DOUT 311	DOUT 310	DOUT 309	DOUT 308	DOUT 307	DOUT 306	DOUT 305	DOUT 304	DOUT 303	DOUT 302	DOUT 301
H21W	DOUT 332	DOUT 331	DOUT 330	DOUT 329	DOUT 328	DOUT 327	DOUT 326	DOUT 325	DOUT 324	DOUT 323	DOUT 322	DOUT 321	DOUT 320	DOUT 319	DOUT 318	DOUT 317
H22W	DOUT 348	DOUT 347	DOUT 346	DOUT 345	OUT 344	DOUT 343	DOUT 342	DOUT 341	DOUT 340	DOUT 339	DOUT 338	DOUT 337	DOUT 336	DOUT 335	DOUT 334	DOUT 333
H23W	DOUT 364	DOUT 363	DOUT 362	DOUT 361	DOUT 360	DOUT 359	DOUT 358	DOUT 357	DOUT 356	DOUT 355	DOUT 354	DOUT 353	DOUT 352	DOUT 351	DOUT 350	DOUT 349
H24W	DOUT 416	DOUT 415	DOUT 414	DOUT 413	DOUT 412	DOUT 411	DOUT 410	DOUT 409	DOUT 408	DOUT 407	DOUT 406	DOUT 405	DOUT 404	DOUT 403	DOUT 402	DOUT 401
H25W	DOUT 432	DOUT 431	DOUT 430	DOUT 429	DOUT 428	DOUT 427	DOUT 426	DOUT 425	DOUT 424	DOUT 423	DOUT 422	DOUT 421	DOUT 420	DOUT 419	DOUT 418	DOUT 417
H26W	DOUT 448	DOUT 447	DOUT 446	DOUT 445	DOUT 444	DOUT 443	DOUT 442	DOUT 441	DOUT 440	DOUT 439	DOUT 438	DOUT 437	DOUT 436	DOUT 435	DOUT 434	DOUT 433
H27W	DOUT 464	DOUT 463	DOUT 462	DOUT 461	DOUT 460	DOUT 459	DOUT 458	DOUT 457	DOUT 456	DOUT 455	DOUT 454	DOUT 453	DOUT 452	DOUT 451	DOUT 450	DOUT 449
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

H000 ~ H27F:
Signifies an exclusive input relay which receives an output signal from the main unit of the robot controller.
The ON/OFF is read at I/O processing of every scan cycle.
This relay can be used as the source of contact input information and data register in the sequence program. It cannot be used as the coil.
Relays in the shaded areas are reserved for future extension of the system functions. Values are undecided and cannot be used.

* H108 to H10A are optional signals for specific customers.

DOUT1 ~ 64, DOUT101 ~ 164, DOUT301 ~ 364, DOUT401 ~ 464

Address	**H000 ~ H07F, H200 ~ H27F**
Direction	Main unit of robot controller → TCmini
Function	Interface relay corresponding to the robot's DOUT command. It turns on and off when the I/O status has changed by the DOUT command, or AUX (auxiliary signal) operation or I/O (external I/O signal display) operation through the teach pendant. This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	Though the relay name is DOUT, it is an input signal from the TCmini.

HANDOUT1 ~ 8

Address	**H100 ~ H107**
Direction	Main unit of robot controller → TCmini
Function	Interface relay corresponding to the robot's DOUT command.

It turns on and off when the I/O status has changed by the DOUT command, or AUX (auxiliary signal) operation or I/O (external I/O signal display) operation through the teach pendant.

HANDOUT1 ~ 8 cannot be reset by the DO_RST signal or output signal reset operation through the teach pendant.

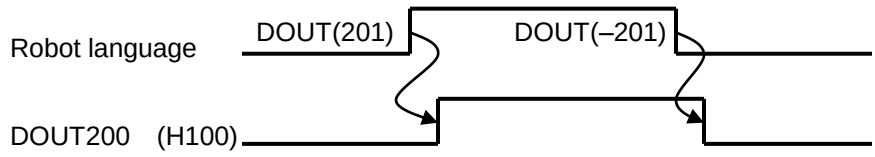
This relay can be turned on and off manually on the exclusive hand monitor screen. (See Para. 12.3 of Operator's Manual.)

Each signal corresponds to the DOUT command as shown below.

Signal name	Relay address	Command
HANDOUT1	H100	DOUT201
HANDOUT2	H101	DOUT202
HANDOUT3	H102	DOUT203
HANDOUT4	H103	DOUT204
HANDOUT5	H104	DOUT205
HANDOUT6	H105	DOUT206
HANDOUT7	H106	DOUT207
HANDOUT8	H107	DOUT208

This relay is output, irrespective of the master mode selected by means of the MODE switch.

Timing chart



Remarks

Though the relay name is HANDOUT, it is an input signal from the TCmini.

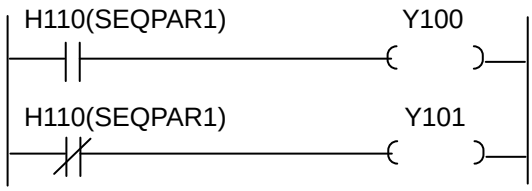
DATA_ERR (Command Error Signal)

Address	**H10E**
Direction	Robot main section → TCmini
Function	This is the command error signal for the Simple PLC data communication function. If there is an error in the value that was set to DATA_CMD, this signal is set to ON. When DATA_TRIG is set to OFF, this signal is also set to OFF.
Notes	For details on the Simple PLC data communication function, see chapter 12.

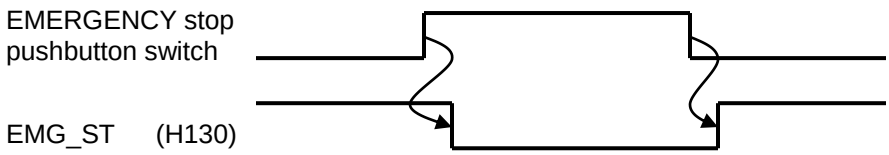
DATA_ACK (Response Complete Signal)

Address	**H10F**
Direction	Robot main section→ TCmini
Function	This is the response complete signal for the Simple PLC data communication function. When setting of the DATA_RESP data is completed, this signal is set to ON. When DATA_TRIG is set to OFF, this signal is also set to OFF.
Remarks	For details on the Simple PLC data communication function, see chapter 12.

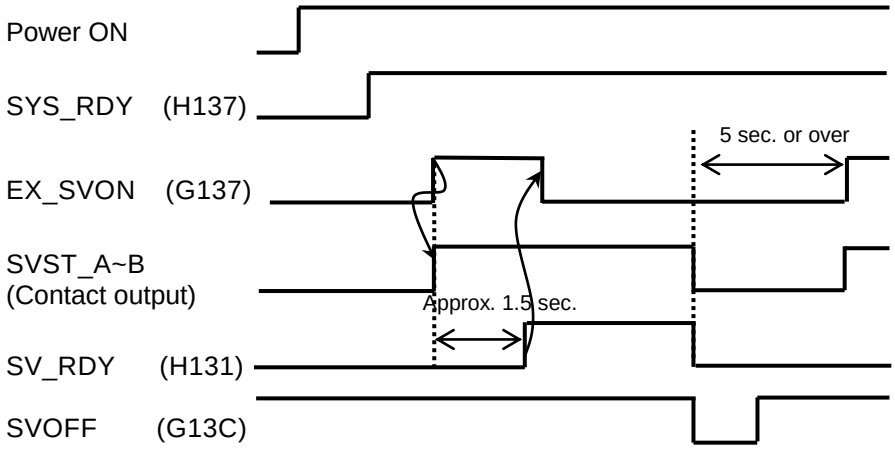
SEQPAR1 ~ 8

Address	**H110 ~ H117**
Direction	Main unit of robot controller → TCmini
Function	Interface relay corresponding to the robot's user parameter ([U13] Sequence parameter). When the power is turned on, parameter values are set in respective relays. If you form a circuit that allows access to this relay, the sequence motion can be changed by the user parameter, without changing over the sequence.
Use example	 The destination changes with the setting of user parameter [U13].
Remarks	The sequence parameter should be set in [U13] of USER.PAR. [U13] Sequence Parameter (User I/O mode only) = 0 0 0 0 0 0 0 0 Set in the order of **R510, R511, R512, R513, R514, R515, R516 and R517**. 0 : OFF 1 : ON

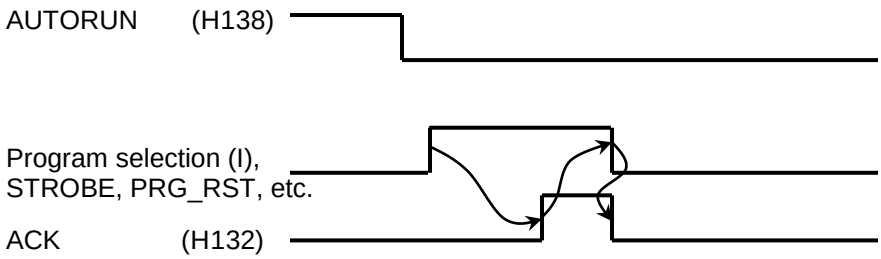
EMG_ST

Address	**H130**
Direction	Main unit of robot controller → TCmini
Function	This is an interface relay that indicates the state of the emergency stop pushbutton or the safety input contact. This is set to OFF when in an emergency stop state (8-014 or 8-017 has occurred). This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	

SV_RDY

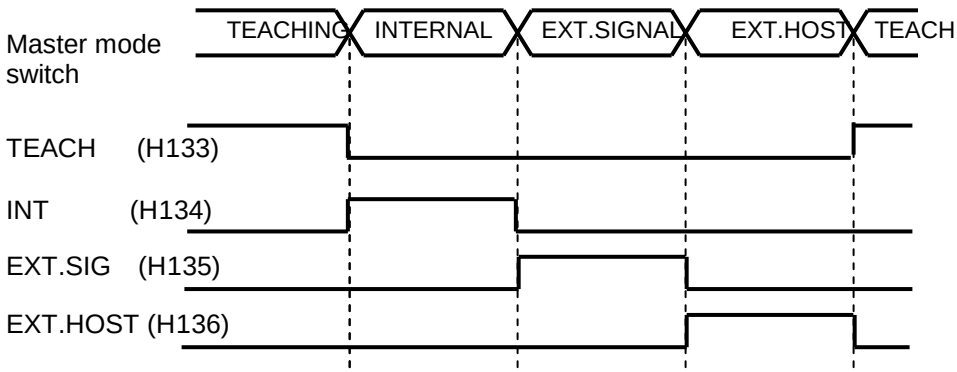
Address	**H131**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating the robot servo status. When the robot program can be started after servo ON, this relay also turns on. This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	It takes about 1.5 seconds until RV_RDY turns on after servo power ON. Keep five (5) seconds or over from the servo power OFF to the next servo power ON.

ACK (Acknowledge)

Address	**H132**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating a response to the input of STROBE, PRG_RST, STEP_RST, CYC_RST and DO_RST. When one of these signals is input, the ACK signal is sent back to inform that the appropriate processing has finished.
Timing chart	
Remarks	If two (2) or more signals shown above are input at the same time, only the signal which was input first is processed, then the ACK signal is output.

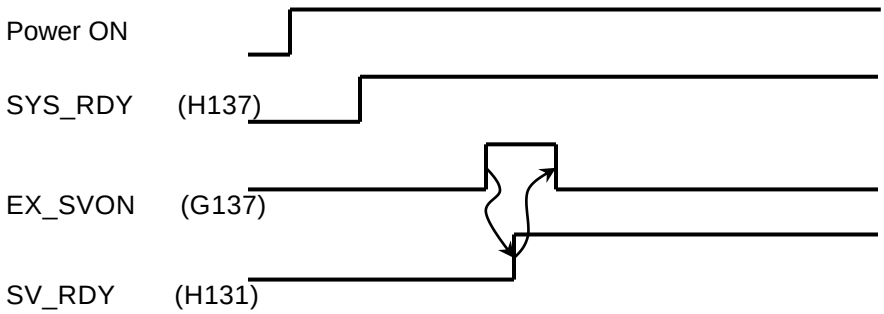
TEACH **(Teach Mode ON)**
INT **(Internal Automatic Mode ON)**
EXTSIG **(External Automatic Signal Mode ON)**
EXT 232C **(External automatic 232C Host Mode ON)**
EXT ETHER **(External ETHER Host Mode ON)**

Address	**H133 ~ H136, H13E**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating the master mode status of the robot. The master mode can be changed over by means of the KEY switch equipped on the control panel. In External Automatic (EXT) mode, the mode is selected by the user parameters.
Timing chart	

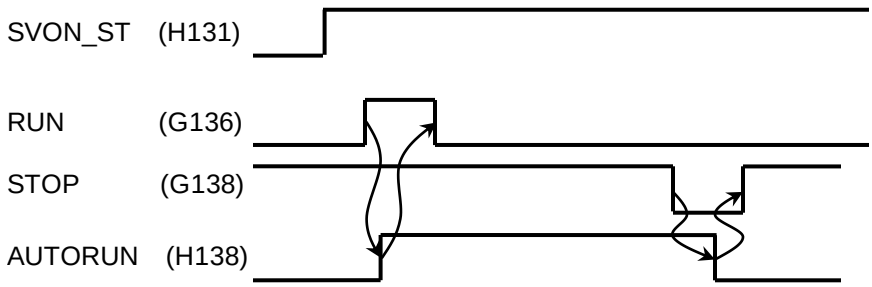


Remarks

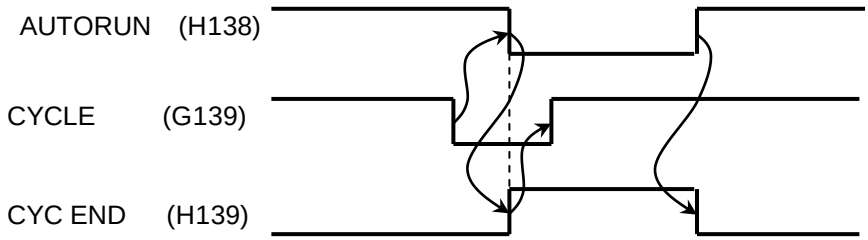
SYS_RDY (System Ready)

Address	**H137**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating a status in which the controller can be operated normally. When the internal startup processing has finished following power ON, this signal turns on. This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	

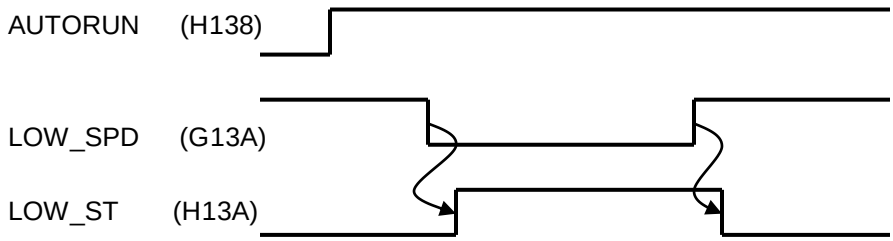
AUTORUN (Auto Mode ON)

Address	**H138**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating that the robot is working in the automatic operation mode. This signal remains on as long as the robot is working in the automatic operation mode. Note: The automatic operation stated above signifies a status of program execution (RUN) in the INTERNAL (i.e., internal automatic mode), EXT.SIGNAL (i.e., external automatic signal mode) or EXT.HOST (i.e., external automatic host mode).
Timing chart	
Remarks	This signal is not output during the TEACHING mode.

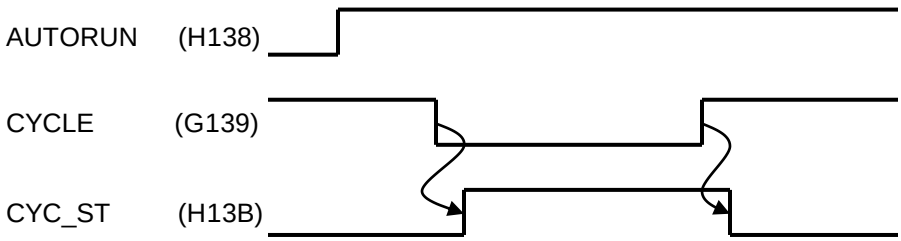
CYC_END (Cycle End)

Address	**H139**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating the finish of program execution. This signal turns on after the stop of 1-cycle automatic operation only when the cycle operation mode is selected and an automatic operation is executed. This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	This relay turns off when the program execution (RUN) mode takes effect. It turns on when the cycle operation has been stopped by the STOP, BREAK or ALARM signal.

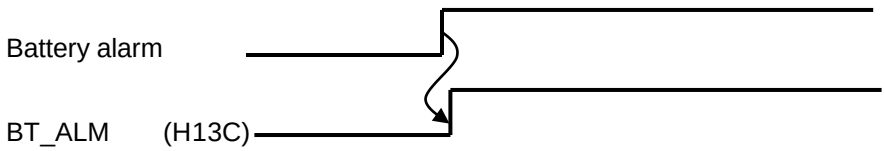
LOW_ST (Low Speed Mode ON)

Address	**H13A**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating that the robot is operating in the low speed mode. It turns on while the robot is working in the low speed mode by the input of LOW_SPD (interface relay). This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	

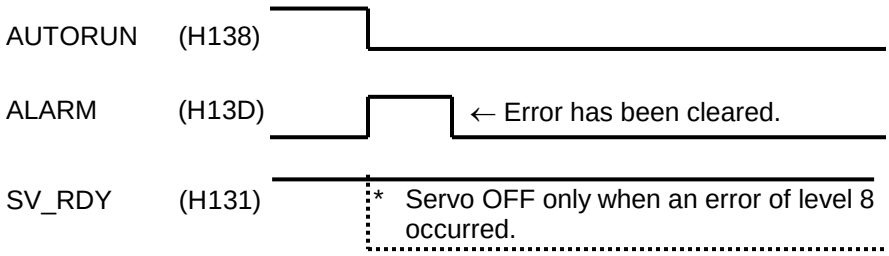
CYC_ST (Cycle Mode ON)

Address	**H13B**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating that the robot is operating in the cycle mode. It turns on while the robot is working in the cycle mode by the input of CYCLE (interface relay) or by the operation through the teach pendant. This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	

BT_ALM (Battery Alarm)

Address	**H13C**
Direction	Main unit of robot controller → TCmini
Function	Turns on if a battery alarm has occurred in the robot controller. The battery alarm comes in the following nine (9) kinds. 1-145 MAIN Battery alarm 1-401 Axis1 Enc Battery low (Battery Alarm) 1-402 Axis2 Enc Battery low (Battery Alarm) 1-403 Axis3 Enc Battery low (Battery Alarm) 1-404 Axis4 Enc Battery low (Battery Alarm) 1-405 Axis5 Enc Battery low (Battery Alarm) 1-406 Axis6 Enc Battery low (Battery Alarm) 1-407 Axis7 Enc Battery low (Battery Alarm) 1-408 Axis8 Enc Battery low (Battery Alarm) This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	If the battery alarm has occurred, replace the battery immediately, referring to the Maintenance Manual.

ALARM (Alarm)

Address	**H13D**
Direction	Main unit of robot controller → TCmini
Function	Interface relay indicating an error of level 2, 4 or 8 in the robot controller or robot. However, this signal does not turn on when a 8-014 Emergency Stop or 8-017 Safety SW ON alarm has occurred. This relay is kept ON during error detection and turns off after the error has been cleared. For details on the errors, see the Operator's Manual. This relay is output, irrespective of the master mode selected by means of the MODE switch.
Timing chart	
Remarks	

6.3 Interface Register

This is an interface area for transferring data with the main unit of the robot controller.

(TCmini → Main unit of robot controller)

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D40*	PLC SS R08	PLC SS R07	PLC SS R06	PLC SS R05	PLC SS R04	PLC SS R03	PLC SS R02	PLC SS R01	PLC DATA R8	PLC DATA R7	PLC DATA R6	PLC DATA R5	PLC DATA R4	PLC DATA R3	PLC DATA R2	PLC DATA R1
D41*	PLC SL R08H	PLC SL R08L	PLC SL R07H	PLC SL R07L	PLC SL R06H	PLC SL R06L	PLC SL R05H	PLC SL R05L	PLC SL R04H	PLC SL R04L	PLC SL R03H	PLC SL R03L	PLC SL R02H	PLC SL R02L	PLC SL R01H	PLC SL R01L
D42*	Reserved area															
D43*																
D44*	Reserved area															
D45*																
D46*	DATA_CMD															
D47*																USER

(Main unit of robot controller → TCmini)

D50*	Reserved area															
D51*																
D52*																
D53*																
D54*																
D55*																
D56*																
D57*																
D60*	PLC SS W08	PLC SS W07	PLC SS W06	PLC SS W05	PLC SS W04	PLC SS W03	PLC SS W02	PLC SS W01	PLC DATA W8	PLC DATA W7	PLC DATA W6	PLC DATA W5	PLC DATA W4	PLC DATA W3	PLC DATA W2	PLC DATA W1
D61*	PLC SL W08H	PLC SL W08L	PLC SL W07H	PLC SL W07L	PLC SL W06H	PLC SL W06L	PLC SL W05H	PLC SL W05L	PLC SL W04H	PLC SL W04L	PLC SL W03H	PLC SL W03L	PLC SL W02H	PLC SL W02L	PLC SL W01H	PLC SL W01L
D62*	Reserved area								PSN_ W8	PSN_ W7	PSN_ W6	PSN_ W5	PSN_ W4	PSN_ W3	PSN_ W2	PSN_ W1
D63*									PSN_ J8	PSN_ J7	PSN_ J6	PSN_ J5	PSN_ J4	PSN_ J3	PSN_ J2	PSN_ J1
D64*									TRQ_ J8	TRQ_ J7	TRQ_ J6	TRQ_ J5	TRQ_ J4	TRQ_ J3	TRQ_ J2	TRQ_ J1
D65*																
D66*	DATA_RESP															
D67*	AL10	AL09	AL08	AL07	AL06	AL05	AL04	AL03	AL02	AL01	ALNO	STEP				
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

STEP

Address	**D674**
Direction	Main unit of robot controller → TCmini
Function	Used to indicate the line number of an active program. As the robot program is executed during execution of pre-analysis, the indicated line number may not coincide with real robot motion. This data can provide only a yardstick.

ALNO, AL01 ~ AL10

Address	**D675 ~ D67F**
Direction	Main unit of robot controller → TCmini
Function	ALNO (D675) signifies the number of alarms currently generated. During alarm generation, relevant alarm number is indicated by AL01 (D136) ~ AL10 (D13F). Ex.) When the alarm of "8–014" has occurred: D675: 1 D676: 8014 (decimal number)

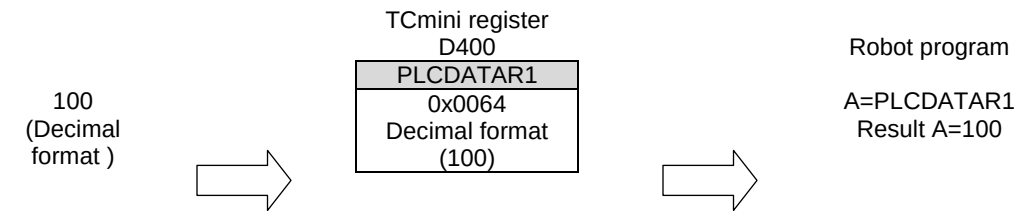
USER

Address	**D470**
Direction	TCmini → Main unit of robot controller
Function	Used to output a value to the 7-segment display on the user panel. The displayed value ranges from 0 to 65535.

PLCDATAR1 ~ 8

Address	**D400 ~ D407**
Direction	TCmini → Main unit of robot controller
Function	Transfers ladder operation result and other data to the main unit of the robot controller. Transferred data can be referred to by the robot program. In the robot program, values can be read by system variables PLCDATAR1 ~ 8. A value written in PLCDATAR* should be 0 ~ 65535. If it exceeds this range, it cannot be transferred accurately.

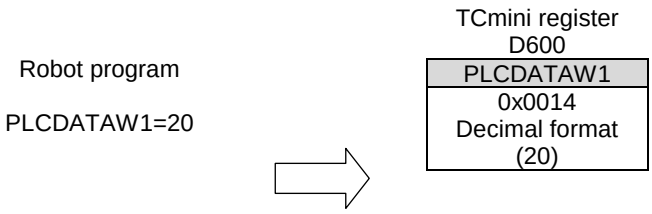
Example:



PLCDATAW1 ~ 8

Address	**D600 ~ D607**
Direction	Main unit of robot controller → TCmini
Function	The TCmini can receive the operation results of the robot program. In the robot program, values can be written to system variables PLCDATAW1 ~ 8. A value written in PLCDATAW* should be 0 ~ 65535. If it exceeds this range, it cannot be transferred accurately.

Example:



PSN_W1 ~ 8

Address	**D620 ~ D627**
Direction	Main unit of robot controller → TCmini
Function	Current position in the world coordinate system of robot can be received by the TCmini. This value is an integer in the range of –32768 ~ 32767 mm (deg). All fractions are ignored.

PSN_J1 ~ 8

Address	**D630 ~ D637**
Direction	Main unit of robot controller → TCmini
Function	Current position in the joint coordinate system of robot can be received by the TCmini. This value is an integer in the range of –32768 ~ 32767 mm (deg). All fractions are ignored.

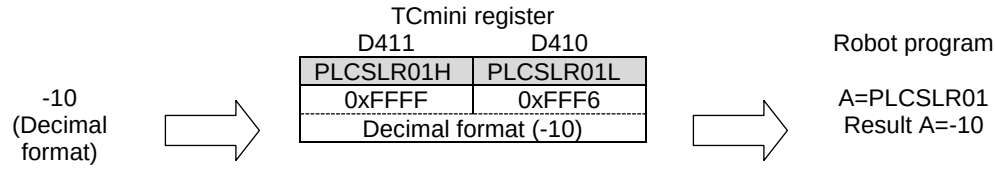
TRQ_J1 ~ 8

Address	**D640 ~ D647**
Direction	Main unit of robot controller → TCmini
Function	Each axis torque value of the robot can be received with TCmini. The value is an integer of -32768 ~ 32767 in every 0.1%. The value of less than 0.1% is rounded down.

PLCSLR01L to 08L and PLCSLR01H to 08H

Address	**D410 to D41F**
Direction	TCmini → Robot main section
Function	This transfers ladder calculation results and other data to the robot main section. The transferred data can be referenced by the robot program. Setting a value to two consecutive registers enables reading of 32-bit integers by the robot program system variables and **PLCSLR01 to 08**.

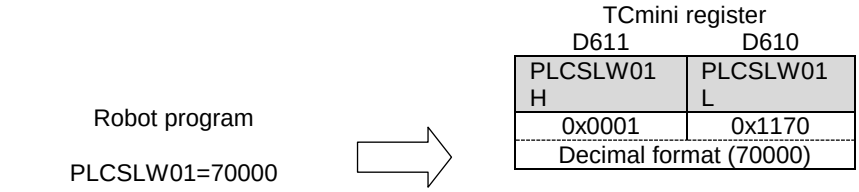
Example:



PLCSLW01L to 08L and PLCSLW01H to 08H

Address	**D610 to D61F**
Direction	Robot main section → TCmini
Function	Robot program calculation results and other data can be received by TCmini. In the robot program, 32-bit integers can be written to two consecutive registers by the system variables and **PLCSLW01 to 08**.

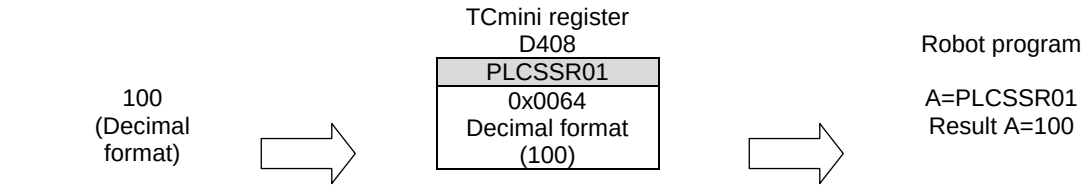
Example:



PLCSSR01 to 08

Address	**D408 to D40F**
Direction	TCmini → Robot main section
Function	Ladder calculation results and other data are transferred to the robot main section. The transferred data can be referenced by the robot program. In the robot program, values can be read by the system variables and **PLCSSR01 to 08**. Set the values to be read by PLCSSR** in the range from -32768 to 32767. Be aware that values outside this range cannot be transferred properly.

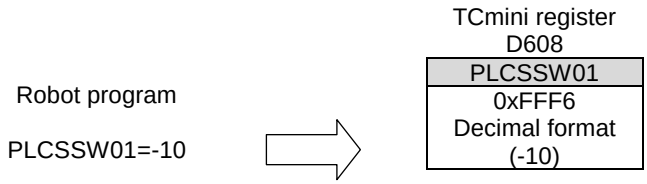
Example:



PLCSSW01 to 08

Address	**D608 to D60F**
Direction	Robot main section → TCmini
Function	Robot program calculation results and other data can be received by TCmini. In the robot program, values can be written by the system variables and **PLCSSW01 to 08**. Set the values to be written by PLCSSW** in the range from -32768 to 32767. Be aware that values outside this range cannot be transferred properly.

Example:



DATA_CMD (Command Set Register)

Address	**D460 to D46F**
Direction	TCmini → Robot main section
Function	This is the command register of the Simple PLC data communication function. Set a command corresponding to the data that you want to obtain. For details on the Simple PLC data communication function, see chapter 12.

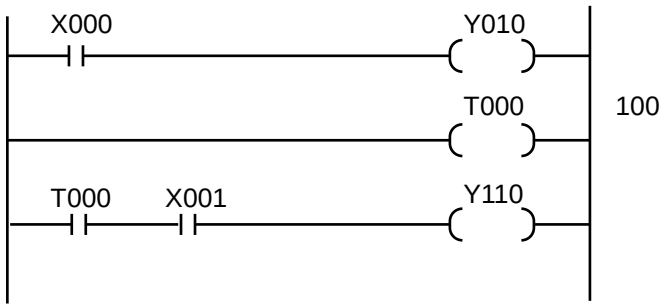
DATA_RESP (Command Response Register)

Address	**D660 to D66F**
Direction	Robot main section → TCmini
Function	This is the command response register of the Simple PLC data communication function. The response to the command that was set is saved. For details on the Simple PLC data communication function, see chapter 12.

Section 7 PLC Language

7.1 Sequence Program

The TCmini supports sequence programs made according to the graphical programming method. They can be created by the user in any format.



I/O processing

Batch refreshing of inputs and outputs is used for I/O processing.
(Before starting arithmetic operation, the input ON/OFF state is transferred to the data memory, and the arithmetic result of the data memory is transferred to the output device.)

Execution of user's program

On completion of I/O processing, the user's program is sequentially operated from the leading circuit.

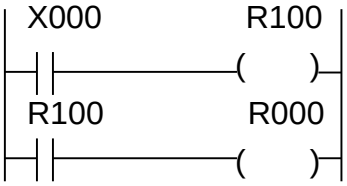
The operation is as follows:

- Sequentially on a circuit basis.
- From left to right on a column basis within one (1) circuit. (The input part is first operated, then the output part is processed .)
- The operation continues to the P. END instruction (program end instruction). The P. END instruction is automatically included at the end of the user's program. Thus only actually used words of the user's program are operated.

Note 1: The coil instruction is written into the data memory every time the operation is executed and has an influence on the subsequent contacts. However, the output state remains unchanged until the output processing is executed in the pack after completion of entire operation.

Note 2: State change of the contact written before the coil occurs in the scan next to the one in which the coil state changed.

Coil R000 is turned on with delay of one (1) scan after X000 is ON.



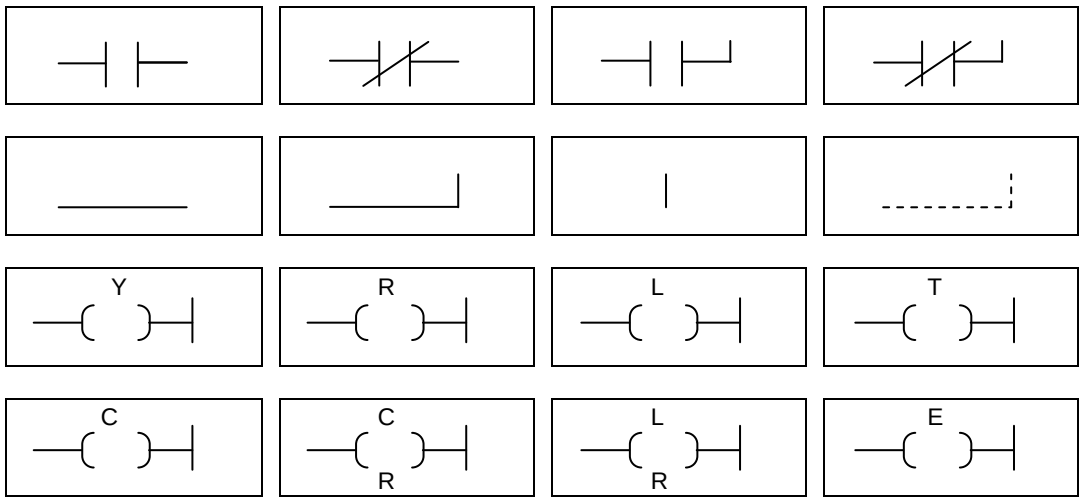
If the order of the left circuit is replaced, R000 is turned on in the same scan as X000 ON.

As known from the above, if the circuit order is replaced, the operation result may differ.

7.2 Program Capacity and Length of Instruction Word

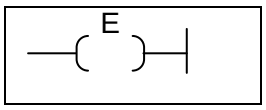
The TCmini instruction has 16 bits as the basic length. According to the type of instruction, five (5) types of word length (1-word, 2-word, 3-word, 4-word and 5-word) are available for the instruction length.
The program memory is Approx. 4000 words and secured according to each instruction.

1-word instruction



2-word instruction

Note: In addresses E000 ~ E03F, two (2)-word instructions are set.



3-word instruction

F*035 Increment	F*036 Decrement	F*040 Arithmetic left shift	F*041 Left rotate
F*042 Right shift	F*043 Right rotate	F*049 Subroutine start	F*058 Subroutine call
F*059 Subroutine return	F*063 1-scan ON		

4-word instruction

F*000 Data transfer	F*001 Constant set	F*002 High-order 8 bits data transfer	F*010 BIN → BCD conversion without sign
F*011 BCD → BIN conversion without sign	F*012 BIN → BCD conversion with sign	F*013 BCD → BIN conversion with sign	F*045 4 → 16 decoder

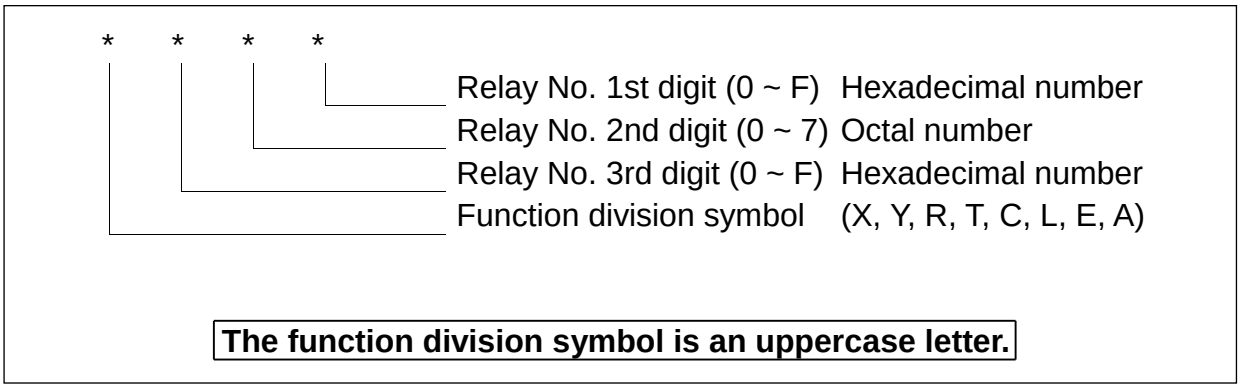
5-word instruction

F*006 Block transfer by constant designation	F*009 Data extraction and distribution	F*020 BIN addition	F*021 BIN addition with carry
F*022 BIN subtraction	F*023 BIN subtraction with borrow	F*024 BIN multiplication without sign	F*025 BIN division without sign
F*032 Logical product	F*033 Logical sum	F*034 Exclusive logical sum	F*037 Comparison without sign
F*038 Comparison with sign	F*047 Bit test by constant designation		

7.3 Address

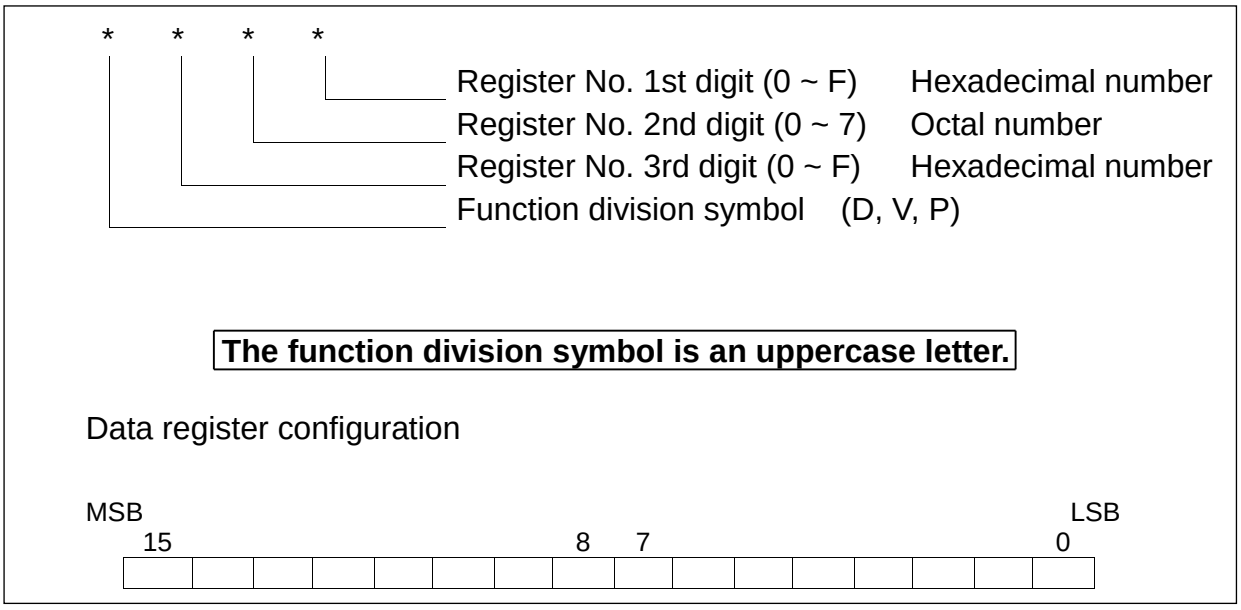
(1) Relay address

The relay address consists of a relay number following the function division symbol.
The I/O relay address corresponds to actual relay mounted position, and the other relays correspond to physically absent devices. The relay address is assigned for each I/O number (1 bit).



(2) Data register address

The data register address is represented in the same manner as the relay address.
The relay address is assigned on a number (1 bit) basis while the data register address is assigned on a word (16-bit) basis.



(3) Byte register address and word register address of relay area

The relay area can be used as a byte register on an eight (8)-number basis and as a word register on a 16-number basis.
For the register address, relay number 1st digit of the relay address signifies the register type instead.

* * * *

- Register type
  - W : Word register (16 numbers F ~ 0)
  - H : High-order byte register (8 numbers F ~ 8)
  - L : Low-order byte register (8 numbers 7 ~ 0)
- Relay/register No. 2nd digit (0 ~ 7)
- Relay/register No. 3rd digit (0 ~ F)
- Function division symbol (X, Y, R, T, C, L, E, A)

The function division symbol is an uppercase letter.

Example: When X130 to X13F are specified by the word register and byte register, each bit corresponds to the following relay address.

X13W

MSB															LSB
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X13F	X13E	X13D	X13C	X13B	X13A	X139	X138	X137	X136	X135	X134	X133	X132	X131	X130

X13L

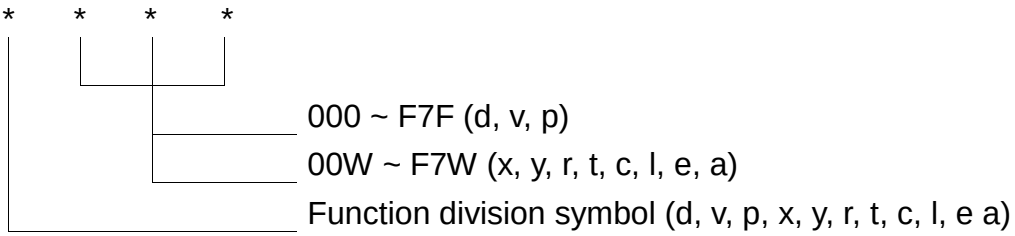
MSB							LSB
7	6	5	4	3	2	1	0
X137	X136	X135	X134	X133	X132	X131	X130

X13H

MSB							LSB
7	6	5	4	3	2	1	0
X13F	X13E	X13D	X13C	X13B	X13A	X139	X138

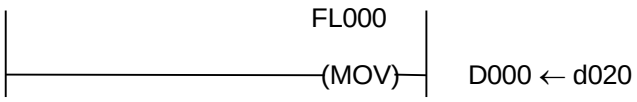
(4) Indirect register address

In the indirect register, an address can be identified by the specified register content (data) and a content under this address can be handled as the word data. The function division symbol at the head of the indirect register address is represented by a small letter (data register or relay register) and the remaining data of the same address are expressed in the same manner as in the data register address or relay register address.



The function division symbol is a lowercase letter.

Example:



If 100H is specified for D020 in the above circuit, for instance, d020 signifies D100. (See the correspondence table below.) Thus, in this circuit, data in D100 is transferred to D000.

* When x or y cannot be automatically identifed in the offline state, it may be displayed as z.
  (Even if x or y is displayed as z, once an online connection is established, the information is obtained from the PLC and updated to x or y.)

Correspondence table of data value and indirect address

Data value (Hex.)	Corresponding address
D register	
0000H (0001H)	D000
0002H (0003H)	D001
≈	
001EH (001FH)	D00F
0020H (0021H)	D010
≈	
00FEH (00FFH)	D07F
0100H (0101H)	D100
≈	
01FEH (01FFH)	D17F
P register	
1000H (1001H)	P000
1002H (1003H)	P001
≈	
101EH (101FH)	P00F
1020H (1021H)	P010
≈	
10BEH (10BFH)	P05F
V register	
1800H (1801H)	V000
1802H (1803H)	V001
≈	
181EH (181FH)	V00F
1820H (1821H)	V010
≈	
18BEH (18BFH)	V05F

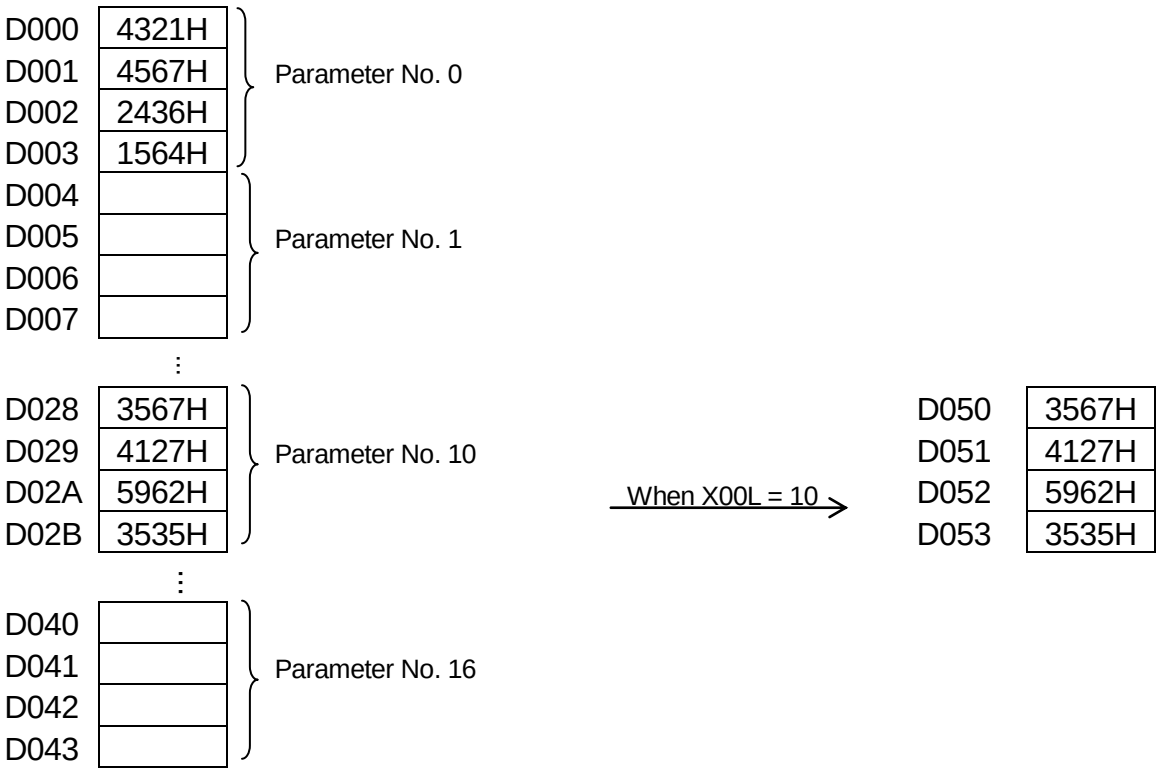
Data value (Hex.)	Corresponding address
X/Y register	
3000H (3001H)	X/Y00W
3002H (3003H)	X/Y01W
≈	
300EH (300FH)	X/Y07W
3010H (3011H)	X/Y10W
≈	
301EH (301FH)	X/Y17W
R register	
3100H (3101H)	R00W
3102H (3103H)	R01W
≈	
310EH (310FH)	R07W
3110H (3111H)	R10W
≈	
311EH (311FH)	R17W
T/C register	
3180H (3181H)	T/C00W
3182H (3183H)	T/C01W
≈	
31BAH (318BH)	T/C05W
L register	
31C0H (31C1H)	L00W
31C2H (31C3H)	L01W
A register	
31E0H (31E1H)	A00W
≈	
31FCH (31FDH)	A16W

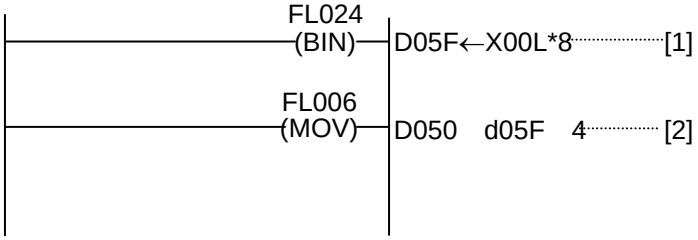
Data value (Hex.)	Corresponding address
E register	
3480H (3481H)	E00W
3482H (3483H)	E01W
3484H (3485H)	E02W
3486H (3487H)	E03W

A register signified by the indirect register is identified from the correspondence table given in Para. 7.3 (4) above. As data values and addresses are arranged consecutively for each register, this function is very useful for a program requiring table processing.

Example: Assume that positioning parameter information consisting of four (4) words per block is set in registers D000 to D043. To set in registers D050 to D053 the four (4)-word positioning information corresponding to the parameter number (0 ~ 16) input in external input register X00L, program as follows.

<Positioning information table>





<Descriptions on program>

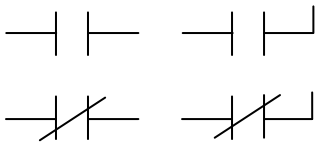
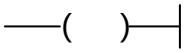
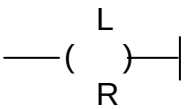
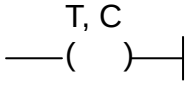
[1] The parameter number (X00L) is multiplied by the number of bytes per block (in this example, 4 words = 8) to identify a relative address from the table.

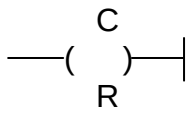
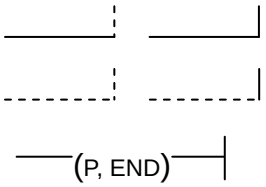
[2] Four (4)-word data are transferred in block into D050 and after, taking the data value set in D05F as the start address (in this example, either of D000 ~ D043).

7.4 Configuration of Instruction Word

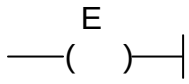
As the TCmini uses the ladder symbolic direct input method for the programming language, it has the instructions corresponding to the circuit diagram.

Configuration of 1-word instruction

Instruction code	Function division symbol	Address
⇓	⇓	⇓
	X: Input Y: Output	000 ~ 17F Note: If X and Y cannot be automatically identified because they share the same address, Z is displayed.
	R: Internal relay	000 ~ 17F
	R: Interface relay	200 ~ 57F
	L: Latch	000 ~ 01F
	T: Timer C: Counter	000 ~ 05F Note: Same addresses cannot be shared by T and C.
	A: Special auxiliary relay	000 ~ 16F
	Y: Output	000 ~ 17F
	R: Internal relay	000 ~ 17F
	R: Interface relay	200 ~ 37F
	L: Latch	000 ~ 01F
	L: Latch reset	000 ~ 01F
	T: Timer C: Counter	000 ~ 05F Note: Same addresses cannot be shared by T and C.

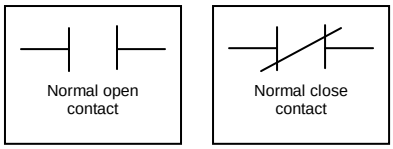
	C: Counter reset	000 ~ 05F Note: Same addresses can be shared by T and C.
	None	None

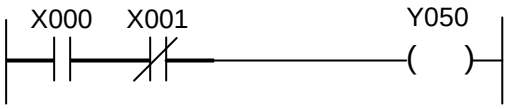
Configuration of 2-word instruction

Instruction code	Function division symbol	Address
⇓	⇓	⇓
	E: Differentiating relay	000 ~ 03F

7.4.1 Contact

(a) Serial connection operation

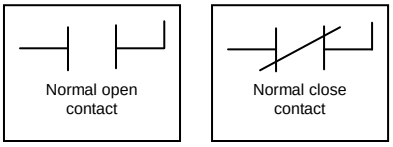


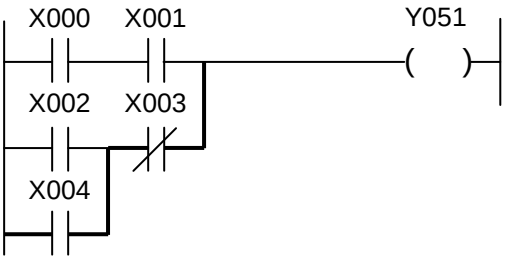


The operation serially connected to the operation result so far is made (logical product: AND).

Result A=

(b) Branch-connected operation

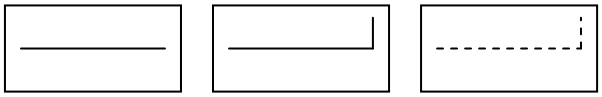


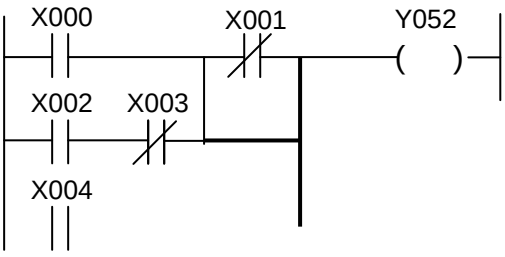


The operation branch-connected (in parallel) with the operation result so far is made (logical sum: OR).

7.4.2 Unconditional Connection

(a) Unconditional connection

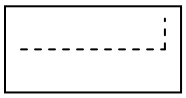




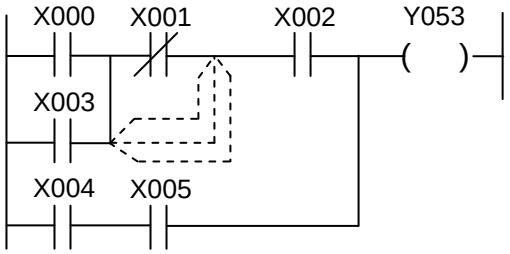
The operation result so far is operated serially, serially and in parallel, and in parallel.

7.4.3 Blank

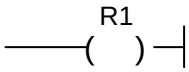
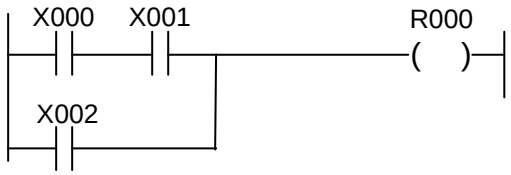
(a) Blank



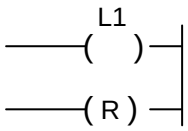
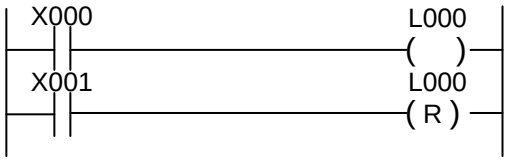
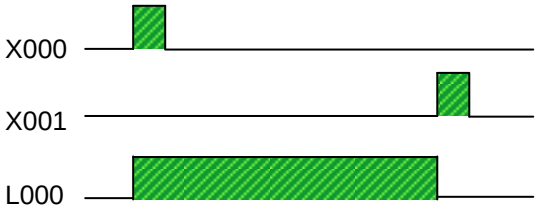
Corresponds to the part enclosed by dotted line in the figure. It can be ignored at programming.
This instruction is effective to delete an instruction at circuit correction.



7.4.4 Internal Relay

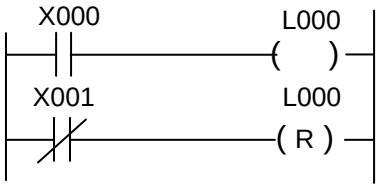
Symbol		
Function	Turned ON when the input signal is ON.	
Executing condition	The input signal must be ON.	
R1 range	R000 ~ R17F (256 Nos.) R200 ~ R37F (Interface relay)	When inputs X000 and X001 are turned ON or when X002 is turned ON, R000 is turned ON.

7.4.5 Latch Relay

Symbol	Latch condition Reset input 	
Function	Kept ON until the reset input is ON when the latch condition is turned ON.	When X000 is ON with X001 set OFF, L000 is ON and this state is held until X001 is ON.
Executing condition	The latch condition must be turned ON with the reset input OFF.	
L1 range	L000 ~ L01F (32 Nos.)	
Power failure backup function	Power failure can't be backed up through the entire area. Depending on the parameter, it is possible to make do memory.	

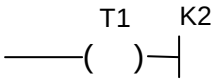
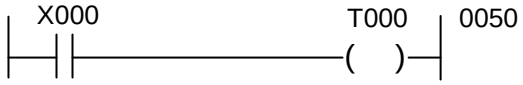
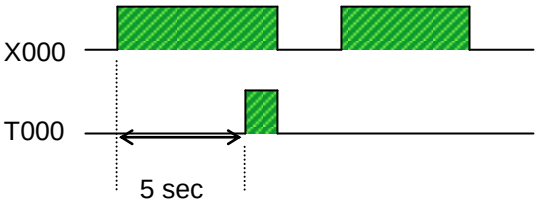
Note: At program loading, compulsively set a required latch relay.

<Power failure backup and input>

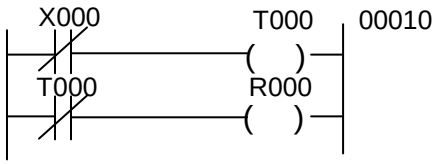


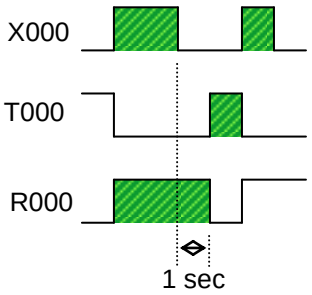
In the circuit shown left, as the "⫽" input contact is used for the reset input, the power may not be backed up due to time lag between input power OFF and PC power OFF.

7.4.6 Timer

Symbol		
Function	Relay T1 is turned ON after the time specified by constant K2 has passed.	T000 is ON five (5) seconds after X000 is ON.
Executing condition	The input signal must be ON.	
T1 range (BIN)	T000 ~ T05F (96 Nos.) 0.1 ~ 3276.7 sec. Note: Shared with counter address.	
Set value K2 range	1 ~ 65535 1H ~ FFFFH (BIN data)	• The timer set value is set in a program, which can be changed through data transfer to V000 to V05F.
Timer set value register	V000 ~ V05F (96 Nos.) Shared with counter.	
Timer current value register	P000 ~ P05F (96 Nos.) Shared with counter.	• Backward timer

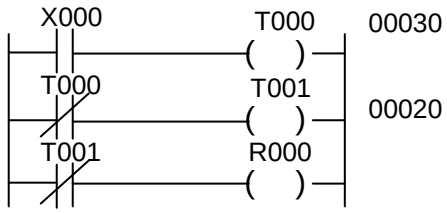
- OFF delay timer circuit

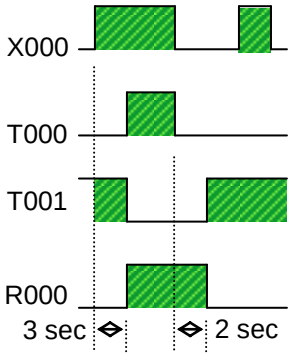




When X000 is ON, R000 is ON. X000 is OFF. One (1) second after the set time on T000 following X000 OFF, R000 is OFF.
It should be noted that R000 is ON for the T000 set time after the PC runs by power ON.

- ON/OFF delay timer circuit

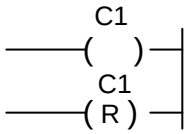
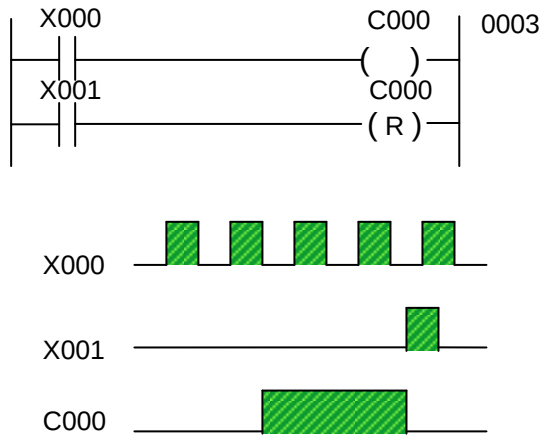




When X000 is ON, R000 is turned ON three (3) seconds after the set time of T000. R000 is OFF two (2) seconds after the set time of T001 following X000 OFF.

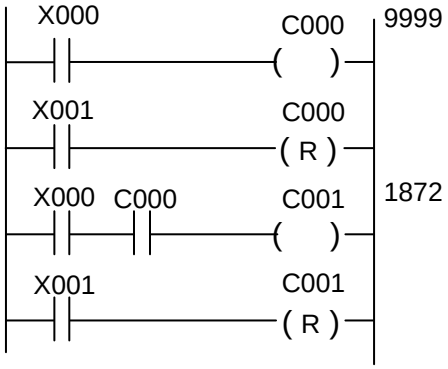
The timer is of a backward type and the current value is equal to the set value at start and zero (0) at ON.

7.4.7 Counter

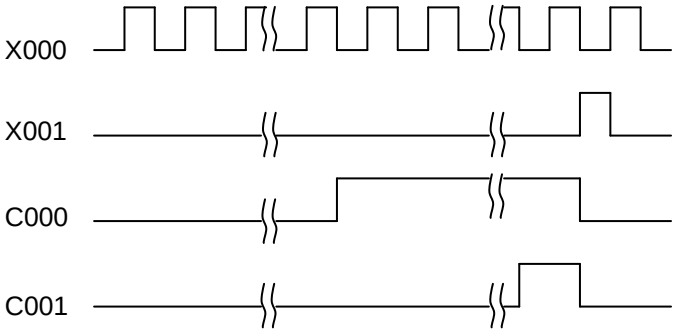
Symbol	Counter input Reset input 	
Function	Relay C1 is turned ON when pulses are input by the number specified by constant K2.	
Executing condition	When the counter input rises from OFF to ON.	
C1 range	C000 ~ C05F (96 Nos.) Note: Shared with timer address.	
Set value K2 range	1 ~ 32767 1H ~ FFFFH (BIN data)	• The counter set value is set in a program, which can be changed through data transfer to V000 to V05F.
Counter set value register	V000 ~ V05F (96 Nos.) Shared with timer.	
Counter current value register	P000 ~ P05F (96 Nos.) Shared with timer.	• Backward counter

The counter current value register is set to zero (0) at program loading. Reset the counter coil to the set value.

- Large-capacity counter circuit



With countup of C000, C001 is counted. C001 is turned ON after counting is made (9999 + 1872 –1 = 11870).



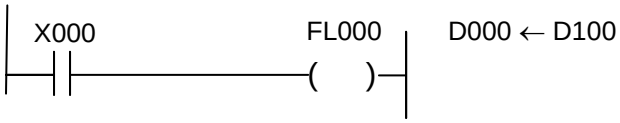
7.5 Standard Application Instruction

The TCmini has a total of thirty-two (32) kinds of application instructions.
In the user's program, up to 512 application instructions can be used.

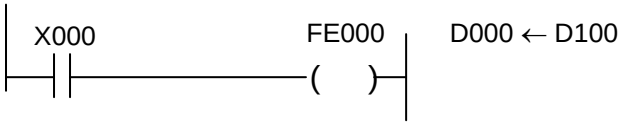
7.5.1 Selection of Executing Condition

The TCmini allows selection of the condition for executing an application instruction. When executing the application instruction while the condition is set ON, write "FL***" (L: level). To execute the application instruction at start, write "FE***" (E: edge).

Example:



When condition X000 is ON, data is transferred from D100 to D000 at each scanning.

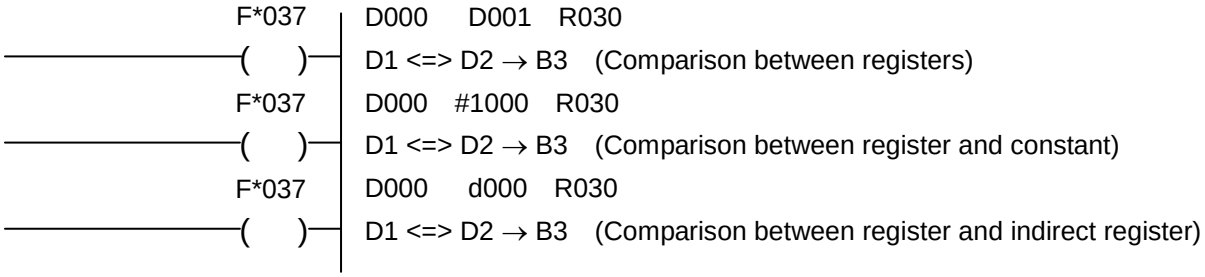


Only in one (1) scan in which condition X000 has changed from OFF to ON, data is transferred from D100 to D000.

7.5.2 Arbitrary Setting of Argument

In the TCmini application instruction, the type of argument (i.e., direct register, indirect register or constant) can be selected arbitrarily. (However, the type of argument is predetermined for some application instructions.)

Example:



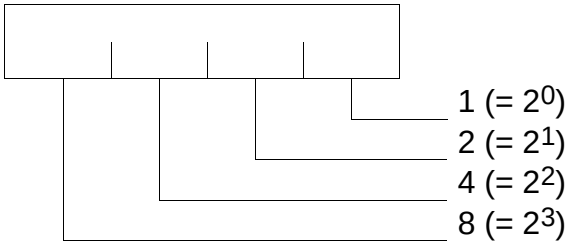
7.5.3 Numeric Expression

(1) Binary code

A numeric value represented by two (2) states of "0" (OFF) and "1" (ON) is called the binary code.
In a decimal number, a number increases to 0, 1, 2, ⋯ 8, 9. When it reaches 10, it is carried. In a binary number, a number next to 1 is carried to 10.
Binary number 10 corresponds to 2 in the decimal notation.

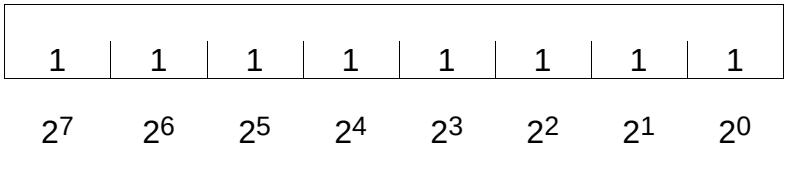
Decimal number	0	1	[2]	3	[4]	5	6	7	[8]
Binary number	0	1	[10]	11	[100]	101	110	111	[1000]

When the decimal number is compared with the binary number, binary numbers corresponding to 2, 4 and 8 in decimal notation are carried.
The decimal number is carried when it is 1, 10, 100 and 1000. When these numbers are expressed in power, they are 10^0 (= 1), 10^1 (= 10), 10^2 (= 100) and 10^3 (= 1000). These are called the "significance" of each digit of decimal numbers.
From the above table, each digit of binary number is as shown below.



Each digit has the significance of 2. Each digit of binary number is called the "bit". A set of eight (8) bits is called the "byte", and a set of 16 bits is called the "word". In the TCmini, an eight (8)-bit length register is called the "byte register".
Now let's investigate the range of byte numeric values.

When all eight (8) bits are 1, a byte is the maximum value.



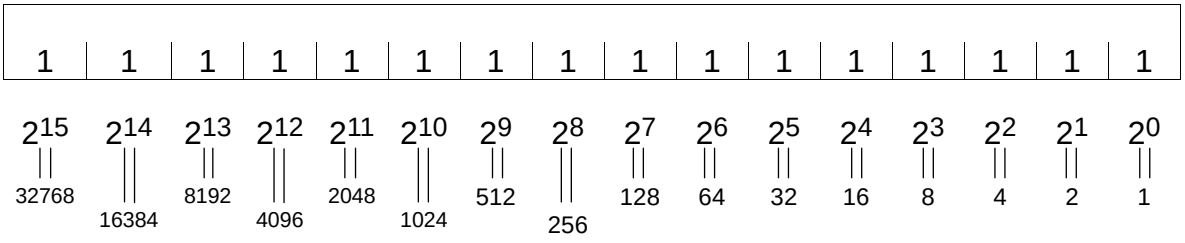
Summing up the significances of each bit,

$2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7$

$= 1 + 2 + 4 + 8 + 16 + 32 + 64 + 128$

$= 255$

Likewise, a word (16 bits) is as follows:



Summing up the significances of all bits, it can be expressed as a decimal number of 0 ~ 65535. Each register of the TCmini can handle the numeric values in binary notation of this range.

Additionally, the set value and current value of timer and counter are processed as the binary number.

(2) Negative expression of binary number (Expression of complement of 2)

In Para. (1) above, all binary numbers are positive. Then how is a negative binary number expressed? If the binary number is decreased one by one,

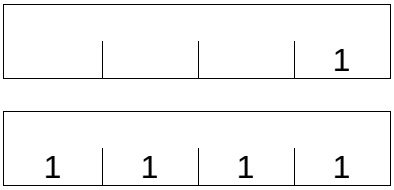
	1	1	1	7
	1	1	0	6
	1	0	1	5
	1	0	0	4
		1	1	3
		1	0	2
			1	1
			0	0

From the above, you can see that a borrow takes place at the time of 4 → 3 and 2 → 1 in decimal notation.

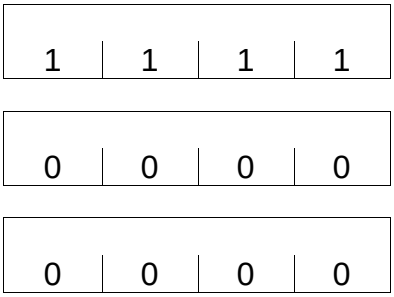
If there is a "1" at a position higher by one when decreasing 1 from 0, the expression is as follows:

1 Borrow				0	0
	1	1	1	1	–1
	1	1	1	0	–2

Let's compare 1 with –1.

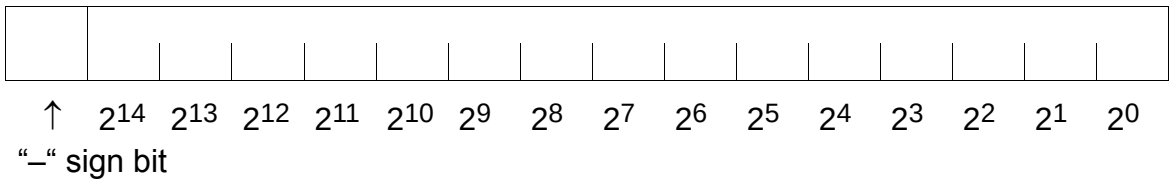


All bits of –1 are reversed (1 → 0) and –1 added with 1 becomes 1.
On the contrary, reverse all bits of 1, then add 1. The result is –1.



This operation is called taking the complement of 2, which is used for conversion from positive to negative and vice versa.
In the above example, the highest-order digit is 1 when the number is negative and 0 when it is positive. This highest-order digit is called the "sign bit", and four (4) bits in the above example can be expressed as a decimal number of –8 to 7.

For a word (16 bits), the expression is as shown below.



It can handle a decimal number of –32768 ~ 32767.
The TCmini can handle the numeric numbers in this range as a signed binary number. The programmer and CRT programmer can specify the numeric values in this range as a decimal number. In the CPU, a specified numeric value is converted into a signed binary number and processed. A signed binary number is handled only in the word register, and the byte register handles a word as a positive integer (0 ~ 255).

(3) Hexadecimal number

When 0 and 1 of binary number are arranged, the expression becomes longer and is difficult to read. To solve this problem, four (4) bits are taken as one (1) unit. A hexadecimal number has four (4) bits, and a carry takes place at 16 after $2^0 + 2^1 + 2^2 + 2^3 = 1 + 2 + 4 + 8 = 15$. It uses numbers 0 to 9 and alphabets A to F.

Binary number	Decimal number	Hexa-decimal number	Binary number	Decimal number	Hexa-decimal number
0000	0	0	1000	8	8
0001	1	1	1001	9	9
0010	2	2	1010	10	A
0011	3	3	1011	11	B
0100	4	4	1100	12	C
0101	5	5	1101	13	D
0110	6	6	1110	14	E
0111	7	7	1111	15	F

Example: A word (16 bits) can be expressed by a four (4) digit number in hexadecimal notation.

0	0	0	1	0	1	0	1	1	0	1	0	1	1	0	0
1				5				A				C			

(4) Binary coded decimal (BCD)

In the decimal number, 10 comes after 0, 1, ⋯ 9 and a carry takes place. A number having the function carrying 9 to 10 like the binary number is called the binary coded decimal (BCD).

Decimal number	Binary number	BCD		
0	0		0	
1	1		1	
2	10		10	
3	11		11	
4	100		100	
5	101		101	
6	110		110	
7	111		111	
8	1000		1000	
9	1001		1001	
10	1010	1	0000	Carry
11	1011	1	0001	
:	:		:	
99	1100011	1001	1001	

A binary number is classified by four (4) bits, and combinations of 1010 or more (1010 ~ 1111) are banned to cause a carry.
Each bit is expressed in the range of 0 to 9 of decimal number.

Example:

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

1 5 6 9

Thus, the BCD expression can be regarded as a variation of hexadecimal number.
The BCD numeric value can also be specified in hexadecimal notation. (Only 0 to 9 for each digit.)
The content of the register in which BCDs are stored can be displayed in hexadecimal notation.

(5) Negative expression of binary coded decimal (BCD)

A negative of BCD is handled as a sign + absolute value. In the TCmini, it can be handled in the two (2) instructions of F*012 (BIN → BCD conversion with sign) and F*013 (BCD → BIN conversion with sign).

The negative is set as the sign digit to "13 (1101)", the value not found in the BCD expression, and handled only as a long word (32-bit, BCD eight (8) digits). The eighth (8th) digit is assigned for the sign digit.
This is because the word (BCD, four (4) digits) is exceeded as a result of BCD conversion when the word numeric range is –32768 to 32767.

Example: Signed BIN to BCD conversion by F*012

–25089

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

Conversion

–	0	0	2	5	0	8	9
1 1 0 1	0 0 0 0	0 0 0 0	0 0 1 0	0 1 0 1	0 0 0 0	1 0 0 0	1 0 0 1

Sign digit: Negative (–) for 1101 and positive (+) for 0000

7.5.4 Operation Flag

(1) Type

The following four (4) types of flags are assigned to the relays (special auxiliary relays) to use the operation result in the coming operation.

Relay address	Flag name	Function
A000	Carry flag	Turned ON if there is a carry or borrow as a result of operation.
A002	Overflow flag	Turned ON if there is an overflow as a result of operation.
A006	Zero flag	Turned ON if the operation result is zero (0).
A007	Sign flag	Turned ON when the MSB (i.e., highest-order bit) of the word register is 1 as a result of operation.

(2) Instruction with change in operation flag

In the following ten (10) kinds of instructions, the flag changes with the operation result.

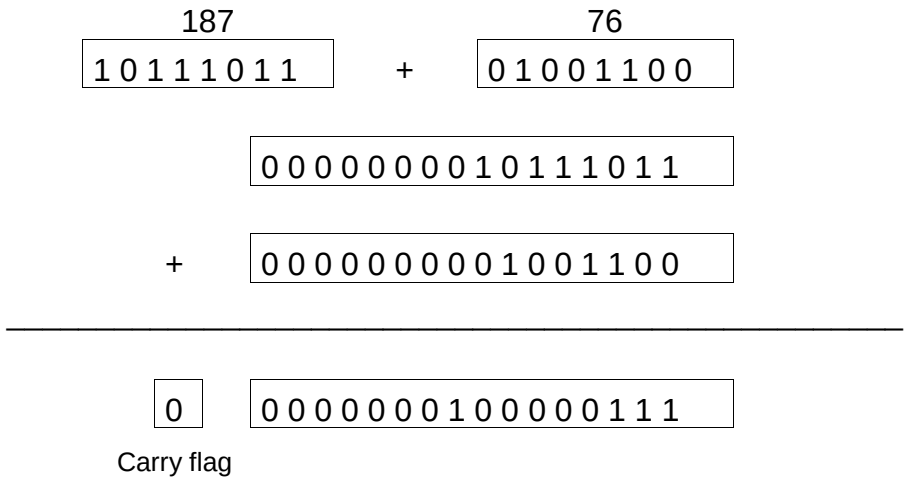
Type	Code	Function
BIN operation	F*010	BIN → BCD conversion
	F*020	BIN addition
	F*021	BIN addition with carry
	F*022	BIN subtraction
	F*023	BIN subtraction with borrow
	F*025	BIN division
Bit shift	F*040	Arithmetic left shift
	F*041	Left rotate
	F*042	Right shift
	F*043	Right rotate

The flag will not change in other than the above instructions.

(3) Cautions on flag

The carry flag and zero flag may change illegally when operated by the byte register.

Example:

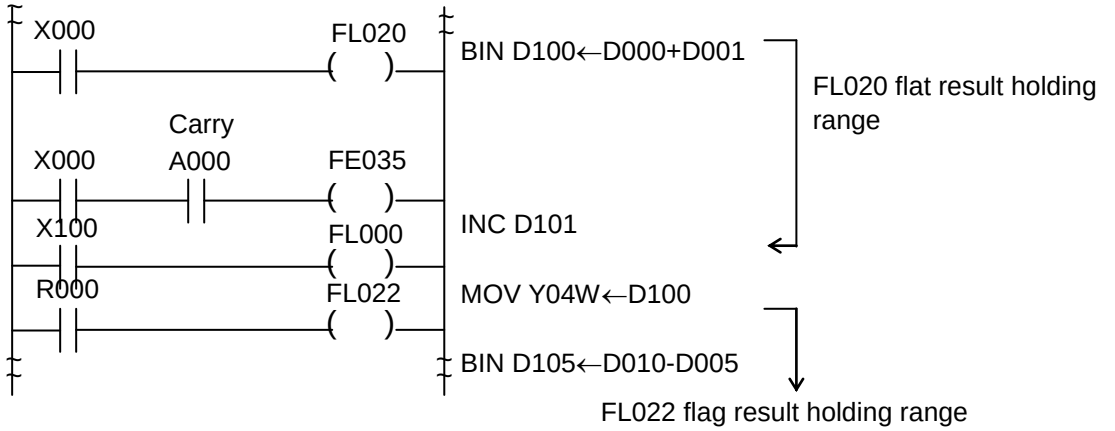


In the operation between byte registers, high-order eight (8) bits of word register are taken as zero (0) and operated as the byte register. Thus, even if the byte register is specified for the destination and a result error is checked by the carry bit, a carry will not take place at the seventh (7th) bit, but the 15th bit becomes the carry flag. In the example above, the carry flag is zero (0).

The sign flag indicates the state of the highest-order bit (15th bit) of the word register. Therefore, the positive or negative sign flag makes sense only when the range of numeric values is handled as a signed binary code (–32768 ~ 32767) by the user.

(4) Available range of operation flag

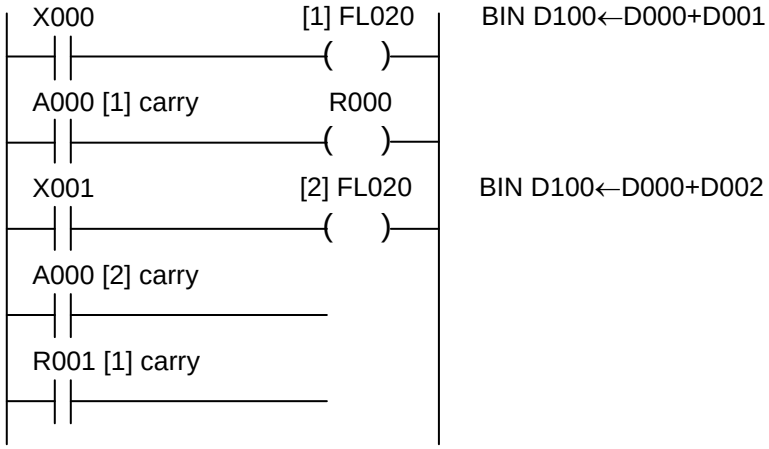
A flag in the user's program maintains its state during the time from a flag operating instruction to the next flag operating instruction.



Note: If a flag operating instruction is not encountered to the program end, the flag keeps its state until another flag operating instruction appears first in the program during the next scan.

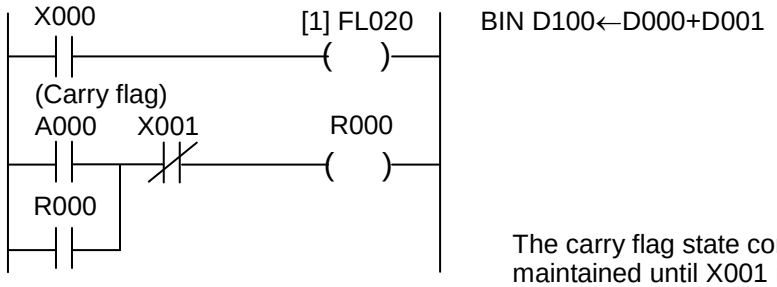
(5) Flag holding

Through the flag state is maintained until the next flag operating instruction, it is changed thereafter.
When the flag state must be held during one (1) scan cycle, move and hold it at the coil (internal relay, output relay, etc.).



R000 can maintain the flag state in one (1) scan cycle until the flag state is changed by the next [1] FL020 instruction. However, when X000 is turned OFF, the flag state of the flag operating instruction located in the circuit before [1] prevails.

To monitor and confirm the flag state through the peripheral equipment, it is convenient to make self backup or use a latch relay. If held by the coil, the flag state can be maintained only in one (1) scan cycle.

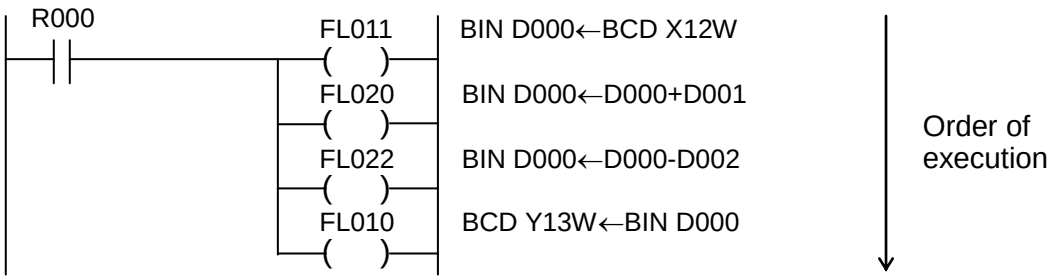


The carry flag state commanded by [1] FL020 is maintained until X001 is turned ON.

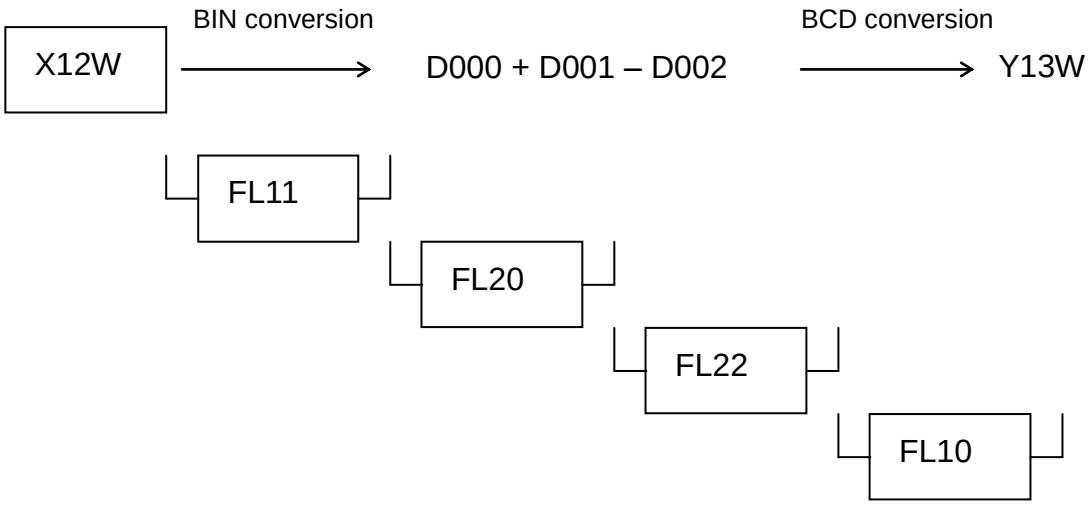
Or

7.5.5 Order of Execution

If the standard application instruction is programmed as a multi-output under the same operating condition, it is executed in the order of top to down in the circuit diagram. Thus it is possible to transfer the operation result to the next operation.



When R000 is ON:

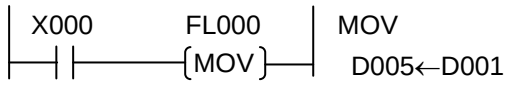


Register D000 is used as the temporary operation result storing register.

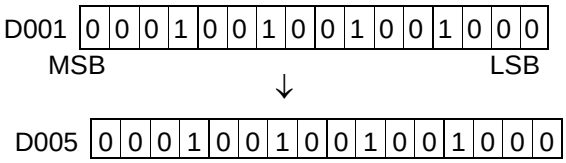
7.5.6 Descriptions on Standard Application Instruction

F*000 Data Transfer

Symbol MOV	Code	Argument Ag.1	Ag.2	Ag.3
	F*000	D_1 d_1	D_2 d_2	
Function	Transfers the content of the register specified by Ag.2 to the register specified by Ag.1.			
Content of operation	MOV $\begin{pmatrix} D_1 \\ d_1 \end{pmatrix} \leftarrow \begin{pmatrix} D_2 \\ d_2 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Ag.2 content • Flag	 Data specified by Ag.2. Unchanged Unchanged			

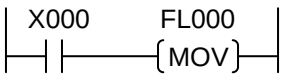


When input X000 is ON, the content of register D001 is transferred to register D005.

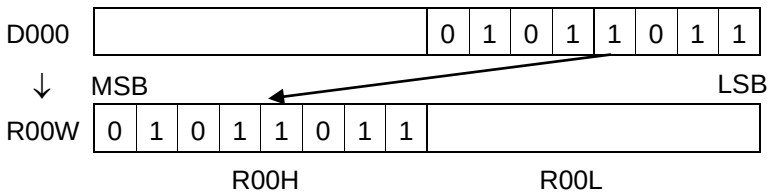


Both the word register and byte register can transfer the data.

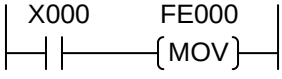
- When register D_1 is the byte register (R00H, Y01L, etc.) and register D_2 is the word register (R00W, Y01W, etc.),



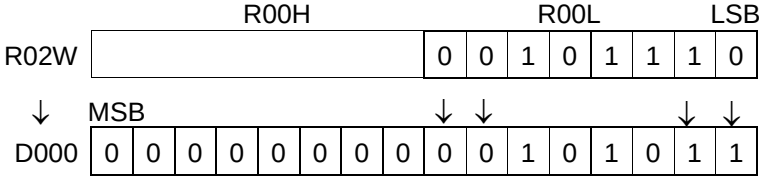
When X000 is ON, the low-order eight (8)-bit data of D000 is transferred to the eight (8) bits of R00H.



- When register D_1 is the word register and register D_2 is the byte register,

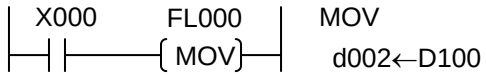


When X000 has turned ON from OFF, the eight (8)-bit data of R02L is transferred to the low-order eight (8) bits of D000. The high-order eight (8)-bit data is zero-cleared.



Note: For data transfer of two (2) words or more, use the block transfer instruction by F*006 or F*008.

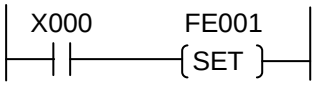
When the indirect register is used,



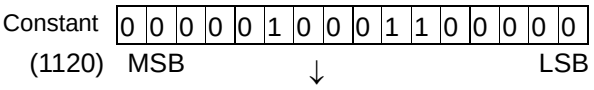
When X000 is ON, the content of D100 is transferred to the address which is identified as per the data set in D002. (If #100H is set in D002, D100 is the address.)

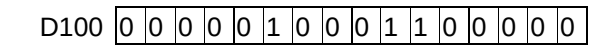
F*001 Constant Set

Symbol SET	Code	Argument Ag.1	Ag.2	Ag.3
	F*001	D_1 d_1	K_2	
Function	Stores the constant specified by Ag.2 in the register specified by Ag.1.			
Content of operation	SET $\begin{pmatrix} D_1 \\ d_1 \end{pmatrix} \leftarrow K_2$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Constant: –32768 ~ 32767			
After operation • Ag.1 content • Ag.2 content • Flag	 Value of constant K_2 Unchanged Unchanged			



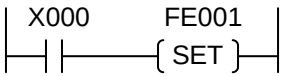
When input X000 has turned ON from OFF, binary constant 1120 is stored in register D100.



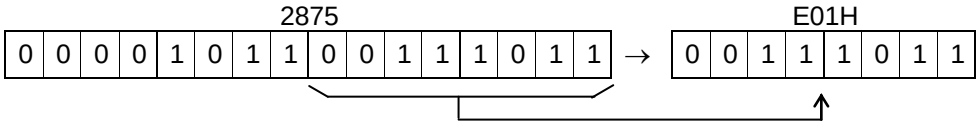


When the register is the byte register, the low-order eight (8)-bit data of the constant is stored in the register. (The high-order eight (8)-bit data of the constant is ignored.)

- When register D_1 is the byte register,

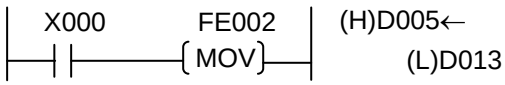


The high-order eight (8)-bit data of the constant is ignored.

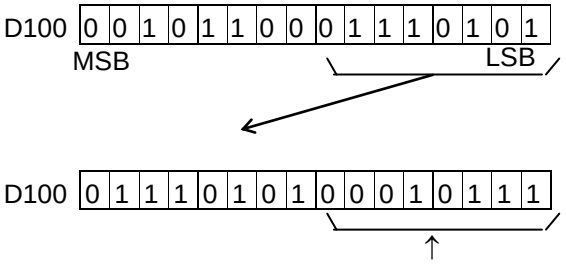


F*002 High-Order 8-Bit Data Transfer

Symbol MOV	Code	Argument Ag.1	Ag.2	Ag.3
	F*002	D_1 d_1	D_2 d_2	
Function	Transfers the low-order eight (8)-bit data of the register specified by Ag.2 to the high-order eight (8) bits of the register specified by Ag.1.			
Content of operation	$\begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$ (H) ← $\begin{pmatrix} D_2 \\ d_2 \end{pmatrix}$ (L)			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content	The high-order eight (8) bits are set as the low-order eight (8) bits of the data value specified by Ag.2.			
• Ag.2 content	Unchanged			
• Flag	Unchanged			

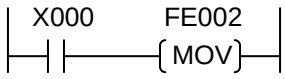


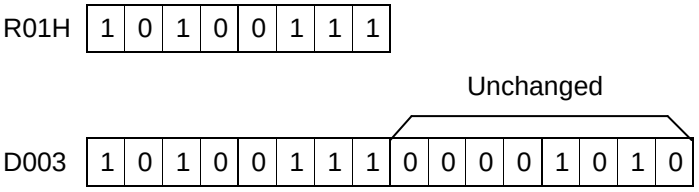
When input X000 is ON, the low-order eight (8)-bit data of register D013 is transferred to the high-order eight (8) bits of register D005.



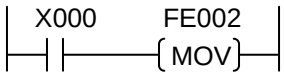
The low-order eight (8) bits of D005 remain unchanged after operation.

- When register D_2 is the byte register,

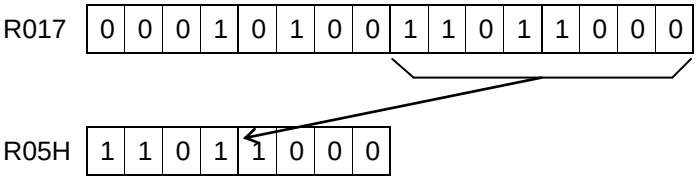




- When register D_1 is the byte register,



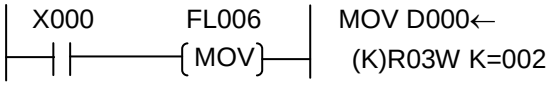
The low-order eight (8)-bit data of D017 is stored in R05H.



(Same operation as F*000.)

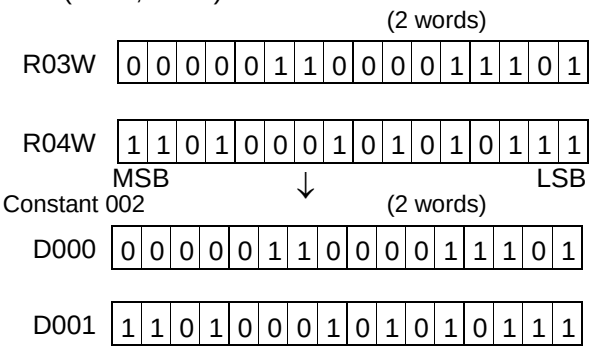
F*006 Block Transfer (Constant Designation)

Symbol MOV	Code	Argument Ag.1	Ag.2	Ag.3
	F*006	D_1 d_1	D_2 d_2	K_3



Function	Transfers block data of constant K3 word from registers headed by the one specified by Ag.2 to registers headed by the one specified by Ag.1.
Content of operation	$\begin{pmatrix}D_2\\d_2\end{pmatrix}\begin{pmatrix}D_2+1\\d_2+1\end{pmatrix}\cdots\begin{pmatrix}D_2+K_3-1\\d_2+K_3-1\end{pmatrix}$ ↓ $\begin{pmatrix}D_1\\d_1\end{pmatrix}\begin{pmatrix}D_1+1\\d_1+1\end{pmatrix}\begin{pmatrix}D_1+2\\d_1+2\end{pmatrix}-\begin{pmatrix}D_1+K_3-1\\D_1+K_3-1\end{pmatrix}$
Range of argument 1	Direct register: Entire range (word designation) Indirect register: Entire range (word designation)
Range of argument 2	Direct register: Entire range (word designation) Indirect register: Entire range (word designation)
Range of argument 3	Constant: 0 ~ 255
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Flag	 Data value specified by Ag.2. Unchanged Unchanged Unchanged

When input X000 is ON, the two (2)-word data stored in registers (R03W, R04W) headed by register R03W is transferred to register D000 and after (D000, D001).

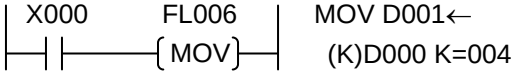


If the designation is outside the register area, data transfer is not performed.

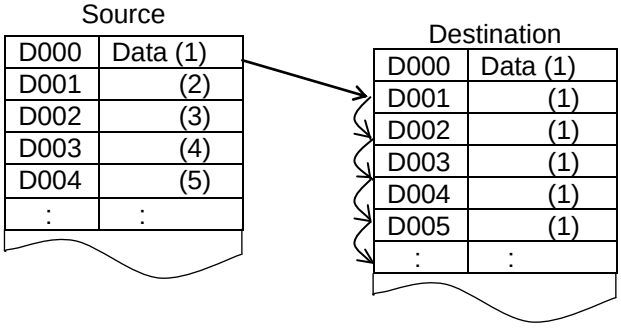
Note: Even if the byte register is specified for the source and destination registers, the word register takes effect.

If the same addresses are used both for the source and destination registers,

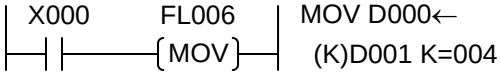
a) When register D_1 address > register D_2 address,



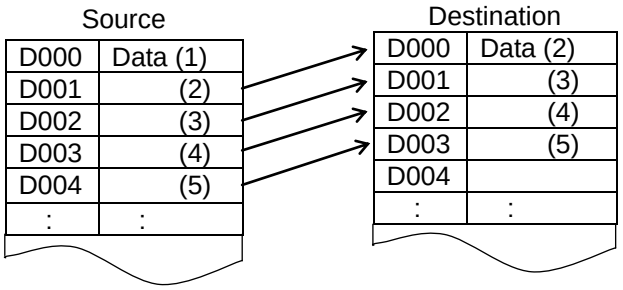
After operation, the same value is set in D000 to D004.



b) When register D_1 address < register D_2 address,

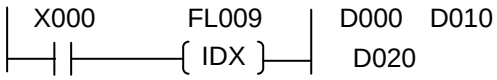


After operation, the content of each register is transferred into the registers with an address number just preceding the number of the source register, respectively.

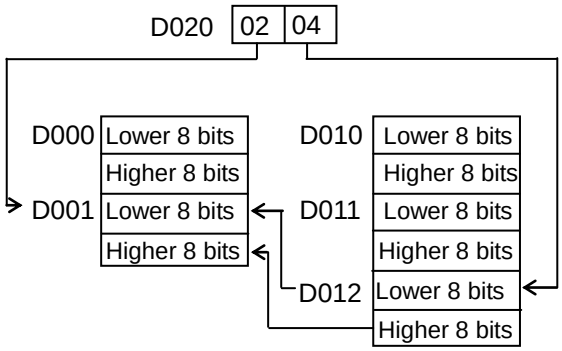


F*009 Data Extraction and Distribution

Symbol IDX	Code	Argument Ag.1	Ag.2	Ag.3
	F*009	D_1 d_1	D_2 d_2	D_3 d_3
Function	Offsets the 2-byte data of the table headed by the Ag.2 register by the low-order byte of the Ag.3 register and transfers the result to the table headed by the Ag.1 register which offsets the data by the high-order byte of the Ag.3 register.			
Content of operation	$\begin{pmatrix} D_2 \\ d_2 \end{pmatrix} + \begin{pmatrix} (D_3(L)) \\ (d_3(L)) \end{pmatrix}$ ↓ $\begin{pmatrix} D_1 \\ d_1 \end{pmatrix} + \begin{pmatrix} (D_3(H)) \\ (d_3(H)) \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range			
Range of argument 3	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Flag	 Operation result Unchanged Unchanged Unchanged			



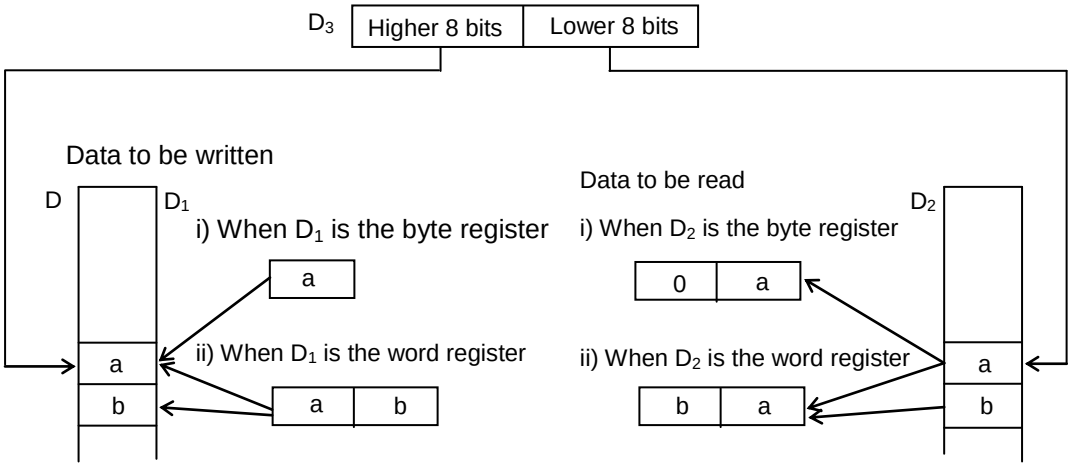
When register D020 = 0204H, the data are extracted and distributed as shown below.



Transferred data is the two (2) bytes.

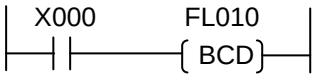
When register D_2 is the byte register, the high-order byte of D_1 becomes zero (0).

When register D_1 is the byte register, the low-order byte of the data read is written.

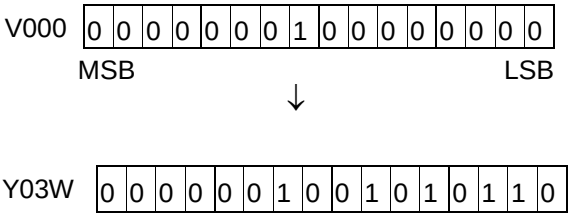


F*010 BIN → BCD Conversion (Unsigned)

Symbol BCD	Code	Argument Ag.1	Ag.2	Ag.3
	F*010	D_1 d_1	D_2 d_2	
Function	Converts the unsigned BIN data of register specified by Ag.2 into the BCD data and stores in the register specified by Ag.1.			
Content of operation	BIN $\begin{pmatrix} D_2 \\ d_2 \end{pmatrix}$ → BCD $\begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Ag.2 content • Flag	Operation result Unchanged Overflow is ON when the D_2 binary data is larger than 9999.			

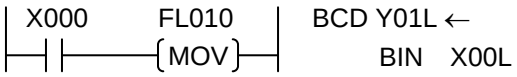


When input X000 is ON, the data of register V000 (timer T000 set value) is converted into the BCD data and stored in register Y03W (Y03F ~ Y030).



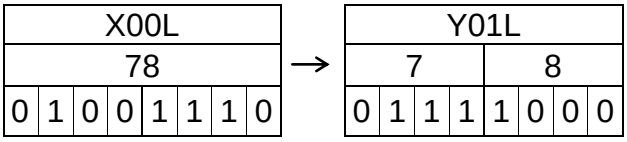
When the binary data of register D_2 is larger than 9999, a code other than BCD is stored in the highest-order position of the BCD, and normal conversion is not performed.

- When registers D_1 and D_2 are the byte registers,

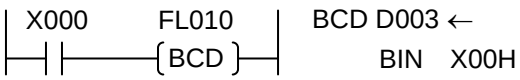


One (1)-byte binary data of X00L is converted into the BCD code and stored in Y01L.

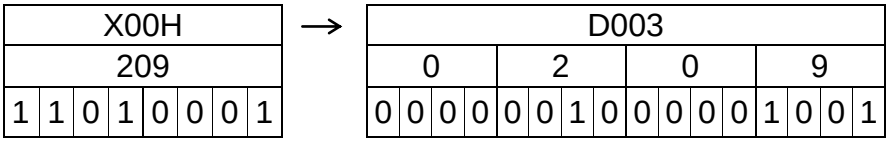
Note: If the X00L value exceeds 99, the third (3rd) digit and above are ignored.



- When register D_1 is the word register and register D_2 is the byte register,

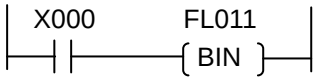


One (1)-byte binary data of X00H is converted into the BCD data and stored in D003.

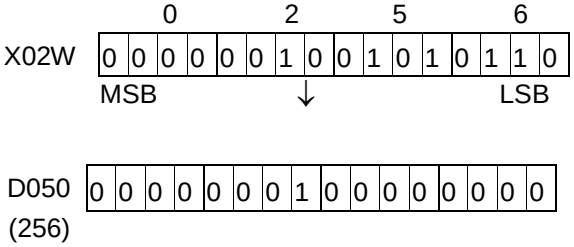


F*011 BCD → BIN Conversion (Unsigned)

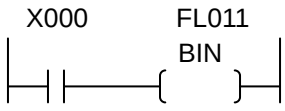
Symbol BIN	Code	Argument Ag.1	Ag.2	Ag.3
	F*011	D_1 d_1	D_2 d_2	—
Function	Converts the unsigned BCD data of register specified by Ag.2 into the BIN data and stores in the register specified by Ag.1.			
Content of operation	BCD $\begin{pmatrix} D_2 \\ d_2 \end{pmatrix}$ → BIN $\begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Ag.2 content • Flag	 Operation result Unchanged Unchanged			



When input X000 is ON, the BCD data of register X02W is converted into the BIN data and stored in D050.



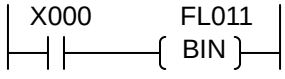
- When registers D_1 and D_2 are the byte registers,



One (1)-byte BCD data of X02H is converted into the BIN data and stored in R05L.

X02H								→	R05L							
6				3					63							
0	1	1	0	0	0	1	1		0	0	1	1	1	1	1	1

- When register D_1 is the word register and register D_2 is the byte register,



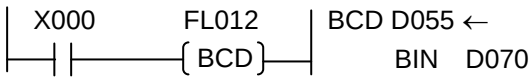
Two (2)-byte BCD data of X03W is converted into the BIN data and the low-order one (1) byte is stored in R04H.

X03W																→	R04H							
0				2				5				4					254							
0	0	0	0	0	0	1	0	0	1	0	1	0	1	0	0		1	1	1	1	1	1	1	0

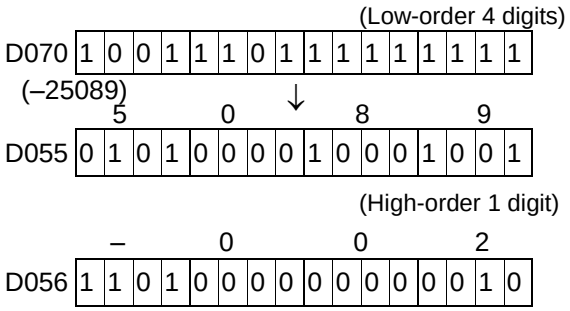
Note: If the X03W value exceeds 256, the low-order eight (8) bits that were converted into the BIN data are stored in R04H.

F*012 BIN → BCD Conversion (Signed)

Symbol BCD	Code	Argument Ag.1	Ag.2	Ag.3
	F*012	D_1 d_1	D_2 d_2	—
Function	Converts the signed BIN data of register specified by Ag.2 into the BCD data and stores in the register specified by Ag.1.			
Content of operation	BIN $\begin{pmatrix} D_2 \\ d_2 \end{pmatrix}$ → BCD $\begin{pmatrix} D_1, D_1+1 \\ d_1, d_1+1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Ag.2 content • Flag	 Operation result Unchanged Unchanged			

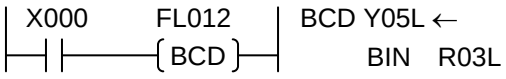


When input X000 is ON, the signed BIN data of register D070 is converted into the signed BCD data, and the low-order four (4)-digit data is stored in register D055 and the high-order one (1) digit and a sign are stored in register D056.



When the high-order four (4)-bit data of register D1+1 is 1101, it signifies the negative value. If 0000, it represents the positive value.

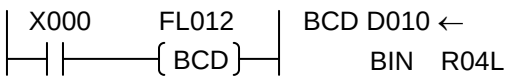
- When registers D_1 and D_2 are the byte registers,



The register D_1 data is four (4) bytes if it is the byte register or word register.

R03L								→	Y06H		Y06L		Y05H		Y05L	
195									0	0	0	0	0	1	9	5
1	1	0	0	0	0	1	1		0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 1		1 0 0 1 0 1 0 1	

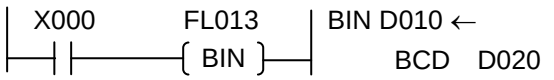
- When register D_1 is the word register and register D_2 is the byte register,



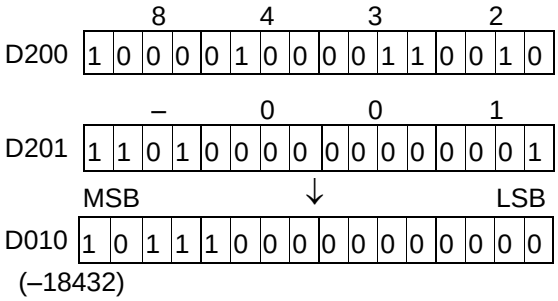
R04H								→	D011				D010			
195									0	0	0	0	0	1	9	5
1	1	0	0	0	0	1	1		0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 1		1 0 0 1 0 1 0 1	

F*013 BCD → BIN Conversion (Signed)

Symbol BIN	Code	Argument Ag.1	Ag.2	Ag.3
	F*013	D_1 d_1	D_2 d_2	
Function	Converts the signed BCD data of register specified by Ag.2 into the BIN data and stores in the register specified by Ag.1.			
Content of operation	BCD $\begin{pmatrix} D_2, D_2+1 \\ d_2, d_2+1 \end{pmatrix}$ → BIN $\begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Ag.2 content • Flag	 Operation result Unchanged Unchanged			

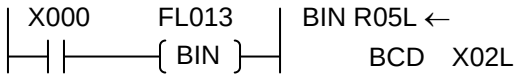


When input X000 is ON, the signed BCD data of registers D200 and D201 is converted into the signed BIN data and stored in register D010.



MSB: 1 when negative and 0 when positive.
If the BCD data of registers D_2, D_2+1 is larger than +32767 or smaller than –32768, the data is not converted into the BIN code legally.

- When registers D_1 and D_2 are the byte registers,



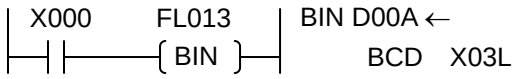
The signed BCD data of X02L, X02H, X03L and X03H are converted into the signed BIN data and stored in register R05L.

X03H				X03L				X02H				X02L					R05L							
–			0	0			0	0			0	1			0	3			–103						
1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1

→

R05L							
–103							
1	0	0	1	1	0	0	1

- When register D_1 is the word register and register D_2 is the byte register,



X04H				X04L				X03H				X03L			
0				0				0				0			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

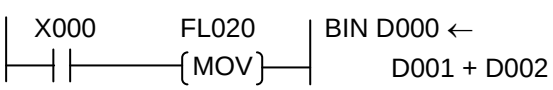
↓

D00A															
709															
0	0	0	0	0	0	1	0	1	1	0	0	0	1	0	1

Even if register D_2 is the byte register or word register, four (4)-byte data is converted.

F*020 BIN Addition

Symbol BIN	Code	Argument Ag.1	Ag.2	Ag.3
	F*020	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3



Function	Adds the BIN data of register specified by Ag.2 and BIN data of register specified by Ag.3 and stores the sum in register specified by Ag.1.
Content of operation	BIN $\begin{pmatrix} D_2 \\ d_2 \\ K_2 \end{pmatrix}$ + BIN $\begin{pmatrix} D_3 \\ d_3 \\ K_3 \end{pmatrix}$ → BIN $\begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$
Range of argument 1	Direct register: Entire range Indirect register: Entire range
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Carry flag • Zero flag • Sign flag	 Operation result Unchanged Unchanged A000: Turned ON when a carry takes place as a result of operation. A006: Turned ON when the operation result is zero (0). A007: Turned ON when the MSB is 1 as a result of operation.

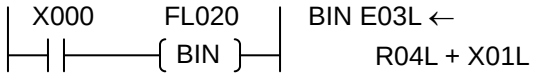
When input X000 is ON, the BIN data of register D001 is added to the BIN data of register D002, which is then stored in register D000.

D001	0	0	0	1	0	1	1	0	0	0	0	0	1	0	0	1
(5641)																
								+								
D002	0	0	0	0	0	1	0	0	1	0	0	0	0	1	1	0
(1158)																
								↓								
	MSB															LSB
D000	0	0	0	1	1	0	1	0	1	0	0	0	1	1	1	1
(6799)																

A000 0

If the word register is used for either register D_2 or D_3 and the byte register is used for register D_1, the sum of low-order eight (8) bits is stored in D_1 and the high-order eight (8) bits are ignored.

- When registers D_1 and D_2 are the byte registers,



The BIN data of R04L and X01L are added and the low-order one (1) byte is stored in E03L.

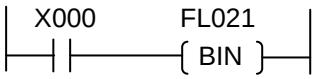
If a carry takes place (i.e., the data exceeds 256), the carry flag will not turn ON.

R04L									X01L									E03L							
198								+	211								→	153							
1	1	0	0	0	1	1	0		1	1	0	1	0	0	1	1		1	0	0	1	1	0	0	1

If the registers D_2 and D_3 are the byte registers, they are operated as the word register whose high-order eight (8) bits are zero (0).

F*021 Carried BIN Addition

Symbol BIN	Code	Argument Ag.1	Ag.2	Ag.3
	F*021	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3
Function	Adds the BIN data of register specified by Ag.2, BIN data of register specified by Ag.3 and carry flag and stores the sum in register specified by Ag.1.			
Content of operation	$\begin{pmatrix} D_2 \\ d_2 \\ K_2 \end{pmatrix} + \begin{pmatrix} D_3 \\ d_3 \\ K_3 \end{pmatrix} + C \rightarrow BIN \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Carry flag • Zero flag • Sign flag	 Operation result Unchanged Unchanged A000: Turned ON when a carry takes place as a result of operation. A006: Turned ON when the operation result is zero (0). A007: Turned ON when the MSB is 1 as a result of operation.			



When input X000 is ON, the BIN data of register D00F, BIN data of register D120 and carry flag (A000) are added, which is then stored in register D110.

D00F 0 0 1 0 1 1 0 0 0 0 0 1 0 0 1 0
(11282)

+

D120 0 0 0 1 0 0 1 1 0 1 0 0 0 1 0 0
(4932)

+

A000 1
(Carry flag)

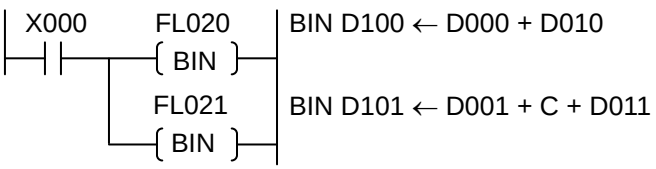
↓

MSB LSB
D110 0 0 1 1 1 1 1 1 0 1 0 1 0 1 1 1
(16215)

A000 0

If the word register is used for either register D_2 or D_3 and the byte register is used for register D_1, the sum of low-order eight (8) bits is stored in D_1 and the high-order eight (8) bits are ignored.

- When adding the binary data of one (1) word or over,



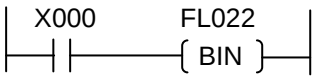
← Addition of low-order one (1) word data.

← Addition of high-order one (1) word data.

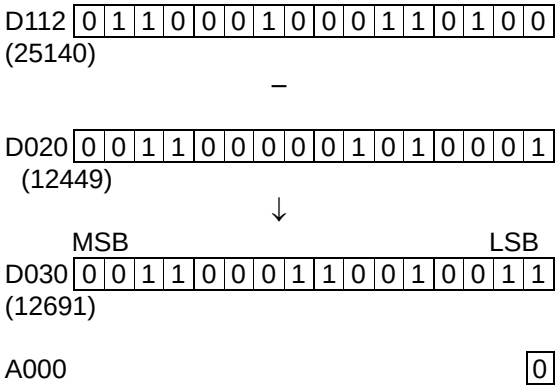
In the above circuit, BIN data of 0 ~ 4294967295 is stored in D100 (low-order word) and D101 (high-order word).

F*022 BIN Subtraction

Symbol BIN	Code	Argument Ag.1	Ag.2	Ag.3
	F*022	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3
Function	Figures out the difference between the BIN data of register specified by Ag.2 and BIN data of register specified by Ag.3 and stores the result in register specified by Ag.1.			
Content of operation	$BIN\begin{pmatrix}D_2\\d_2\\K_2\end{pmatrix} - BIN\begin{pmatrix}D_3\\d_3\\K_3\end{pmatrix} \rightarrow BIN\begin{pmatrix}D_1\\d_1\end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Carry flag • Zero flag • Sign flag	 Operation result Unchanged Unchanged A000: Turned ON when a borrow takes place as a result of operation. A006: Turned ON when the operation result is zero (0). A007: Turned ON when the MSB is 1 as a result of operation.			

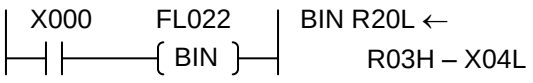


When input X000 is ON, the BIN data of register D020 is subtracted from the BIN data of register D112, which is then stored in register D030.



If the word register is used for either register D_2 or D_3 and the byte register is used for register D_1, the low-order eight (8) bits of the result are stored in D_1 and the high-order eight (8) bits are ignored.

- When registers D_1, D_2 and D_3 are the byte registers,

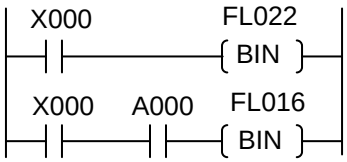


The BIN data of X04L is subtracted from the BIN data of R03H, and the difference is stored in R20L.

If a borrow takes place as a result of operation, the carry flag turns ON.

R03H									X04L									R20L								
57								–	108								→	–51								
0	0	1	1	1	0	0	1		0	1	1	0	1	1	0	0		1	1	0	0	1	1	0	1	

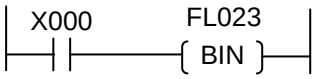
- When obtaining the absolute operation result,



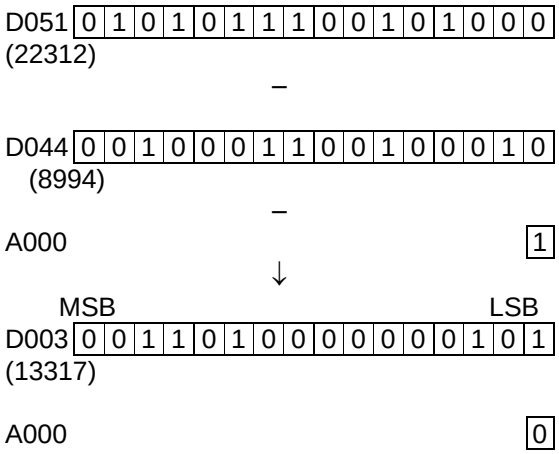
← When a borrow takes place, the value is changed to a positive one using the complement of 2 of the result.

F*023 Borrowed BIN Subtraction

Symbol BIN	Code	Argument Ag.1	Ag.2	Ag.3
	F*023	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3
Function	Subtracts the BIN data of register specified by Ag.3 and carry flag from the BIN data of register specified by Ag.2 and stores the difference in register specified by Ag.1.			
Content of operation	$\begin{pmatrix} D_2 \\ d_2 \\ K_2 \end{pmatrix} - \begin{pmatrix} D_3 \\ d_3 \\ K_3 \end{pmatrix} - C \rightarrow BIN \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Carry flag • Zero flag • Sign flag	 Operation result Unchanged Unchanged A000: Turned ON when a borrow takes place as a result of operation. A006: Turned ON when the operation result is zero (0). A007: Turned ON when the MSB is 1 as a result of operation.			

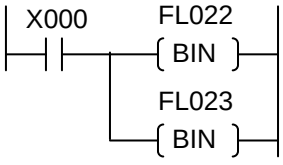


When input X000 is ON, the BIN data of register D044 and carry flag (A000) are subtracted from the BIN data of register D051, and the difference is stored in register D003.



If the word register is used for either register D_2 or D_3 and the byte register is used for register D_1, the low-order eight (8) bits of the result are stored in D_1 and the high-order eight (8) bits are ignored. Data of 0 ~ 65535 is also available.

- When subtracting the binary data of one (1) word or over,



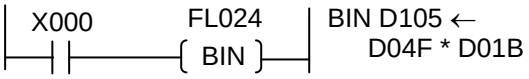
BIN D100 ← D010 – D000 ← Subtraction of low-order one (1)-word data.

BIN D101 ← D011 – C – D011 ← Subtraction of high-order one (1)-word data.

It should be noted that if a borrow takes place in the subtraction of high-order one (1)-word data, the data is illegal.

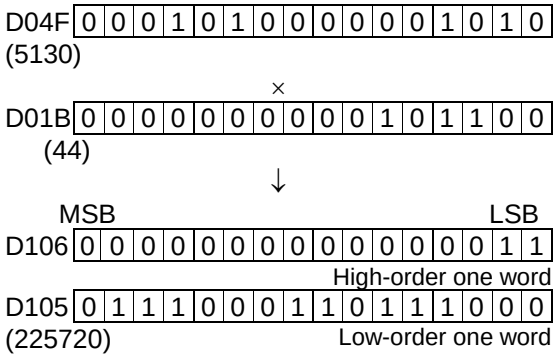
F*024 Unsigned BIN Multiplication

Symbol BIN	Code	Argument Ag.1	Ag.2	Ag.3
	F*024	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3



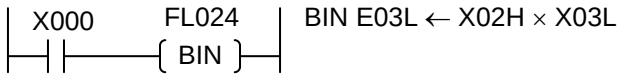
Function	Figures out the product of the BIN data of register specified by Ag.2 and BIN data of register specified by Ag.3 and stores the result in register specified by Ag.1.
Content of operation	$\begin{pmatrix} D_2 \\ d_2 \\ K_2 \end{pmatrix} * \begin{pmatrix} D_3 \\ d_3 \\ K_3 \end{pmatrix} \rightarrow \text{BIN}, \begin{pmatrix} D_1, D_1+1 \\ d_1, d_1+1 \end{pmatrix}$
Range of argument 1	Direct register: Entire range Indirect register: Entire range
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: 0 ~ 65535
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: 0 ~ 65535
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Flag	 Operation result Unchanged Unchanged Unchanged

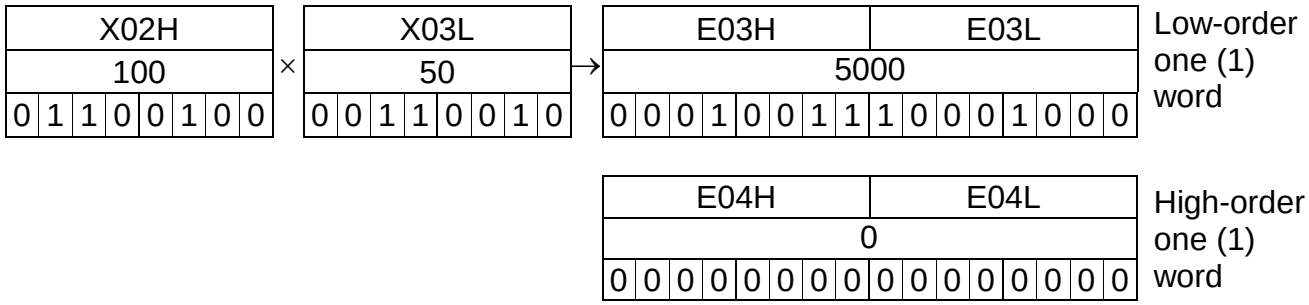
When input X000 is ON, the BIN data of register D04F is multiplied by the BIN data of register D01B, and the low-order 1-word data is stored in register D105 and the high-order 1-word data in register D106.



Mixed use of byte register and word register is possible. However, even if either register (byte register or word register) is specified for register D_1, four (4)-byte data is stored in register D_1. Data of 0 ~ 65535 are available.

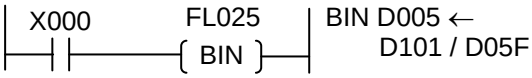
- When all registers are the byte registers,





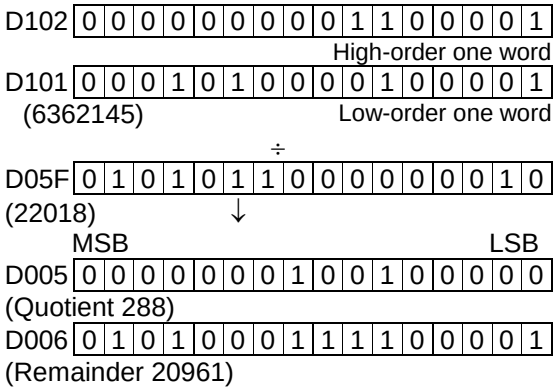
F*025 Unsigned BIN Division

Symbol BIN	Code	Argument Ag.1	Ag.2	Ag.3
	F*025	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3



Function	Divides the BIN data of registers specified by Ag.2 by the BIN data of register specified by Ag.3 and stores the quotient and remainder in registers specified by Ag.1.
Content of operation	$\begin{pmatrix} D_2, D_2+1 \\ d_2, d_2+1 \\ K_2 \end{pmatrix} / \begin{pmatrix} D_3 \\ d_3 \\ K_3 \end{pmatrix} \rightarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$ (quotient) $\begin{pmatrix} D_1+1 \\ d_1+1 \end{pmatrix}$ (remainder)
Range of argument 1	Direct register: Entire range Indirect register: Entire range
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: 0 ~ 65535
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: 0 ~ 65535
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Flag	 Operation result Unchanged Unchanged Unchanged

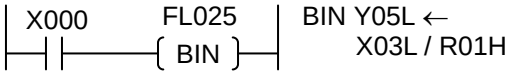
When input X000 is ON, the four (4)-byte BIN data of register D101 and register D102 is divided by the two (2)-byte BIN data of register D05F, and the quotient and remainder are stored in registers D005 and D006, respectively.



Even if either register (byte register or word register) is specified for registers D_1 and D_2, they are operated as the two (2) word data.
Data of 0 ~ 65535 are available.

- Values of dividends D_2, D_2+1 can exceed 16 bits, but quotient D_1 must be less than 16 bits.

- When all registers D_1, D_2, D_3 are the byte registers,



Two (2)-word BIN data of X04H, X04L, X03H and X03L is divided by the data of R01H, and the quotient is stored in Y015H and Y05L and the remainder in Y16H and Y16L.

X04H	X04L	X03H	X03L		R01H
		6735		÷	197
0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 1 1 0 1 0	0 1 0 0 1 1 1 1		1 1 0 0 0 1 0 1

Y05H	Y05L
34	
0 0 0 0 0 0 0 0	0 0 1 0 0 0 1 0

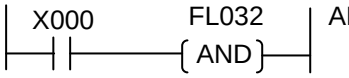
Quotient

Y06H	Y06L
37	
0 0 0 0 0 0 0 0	0 0 1 0 0 1 0 1

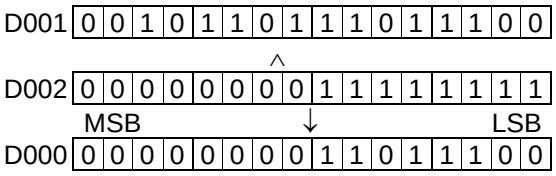
Remainder

F*032 Logical Product (AND)

Symbol AND	Code	Argument Ag.1	Ag.2	Ag.3
	F*032	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3
Function	Figures out the logical product (AND) of BIN data of registers specified by Ag.2 and Ag.3 and stores the result in register specified by Ag.1.			
Content of operation	$\begin{pmatrix} D_2 \\ d_2 \\ K_2 \end{pmatrix} \wedge \begin{pmatrix} D_3 \\ d_3 \\ K_3 \end{pmatrix} \rightarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Flag	 Operation result Unchanged Unchanged Unchanged			

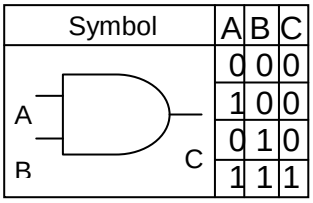


When input X000 is ON, the logical product of BIN data of registers D001 and D002 is figured out and the result is stored in register D000.



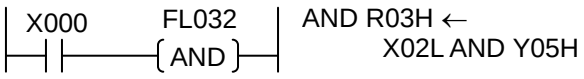
When either register D_2 or D_3 is the word register and register D_1 is the byte register, the logical product of low-order eight (8) bits is figured out and stored in register D_1. The high-order eight (8) bits are ignored.

AND truth table

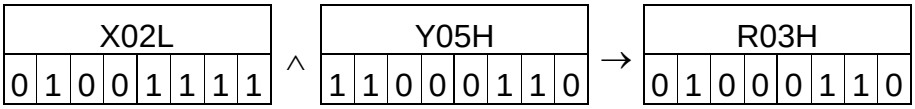


A	B	C
0	0	0
1	0	0
0	1	0
1	1	1

- When registers D_1, D_2 and D_3 are the byte registers,

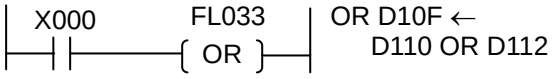


The logical product of X02L and Y05H is figured out and stored in R03H.



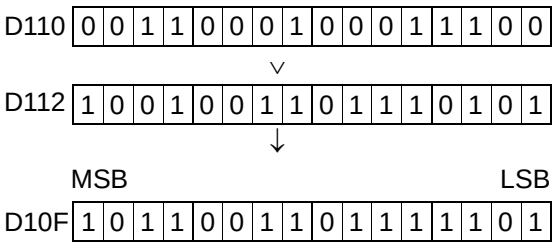
F*033 Logical Sum (OR)

Symbol OR	Code	Argument Ag.1	Ag.2	Ag.3
	F*033	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3



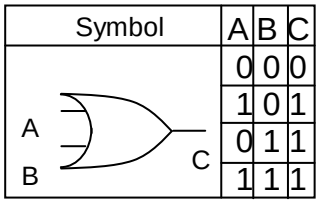
Function	Figures out the logical sum (OR) of BIN data of registers specified by Ag.2 and Ag.3 and stores the result in register specified by Ag.1.
Content of operation	$\begin{pmatrix} D_2 \\ d_2 \\ K_2 \end{pmatrix} \vee \begin{pmatrix} D_3 \\ d_3 \\ K_3 \end{pmatrix} \rightarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$
Range of argument 1	Direct register: Entire range Indirect register: Entire range
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Flag	 Operation result Unchanged Unchanged Unchanged

When input X000 is ON, the logical sum of BIN data of registers D110 and D112 is figured out and the result is stored in register D10F.

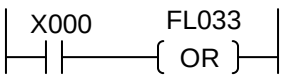


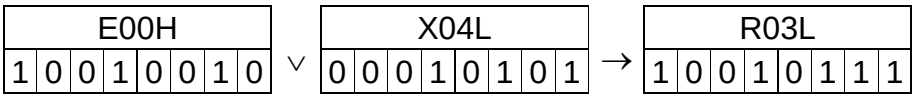
When either register D_2 or D_3 is the word register and register D_1 is the byte register, the logical sum of low-order eight (8) bits is figured out and stored in register D_1. The high-order eight (8) bits are ignored.

OR truth table



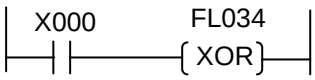
- When registers D_1, D_2 and D_3 are the byte registers,



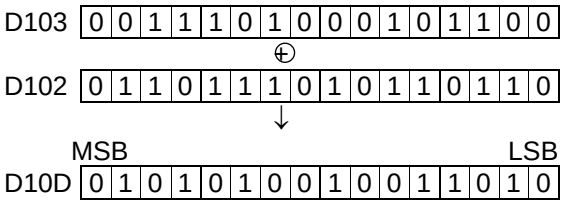


F*034 Exclusive Logical Sum (XOR)

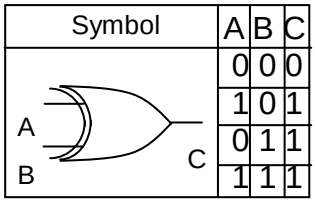
Symbol XOR	Code	Argument Ag.1	Ag.2	Ag.3
	F*034	D_1 d_1	D_2 d_2 K_2	D_3 d_3 K_3
Function	Figures out the exclusive logical sum (XOR) of BIN data of registers specified by Ag.2 and Ag.3 and stores the result in register specified by Ag.1.			
Content of operation	$\begin{pmatrix} D_2 \\ d_2 \\ K_2 \end{pmatrix} \oplus \begin{pmatrix} D_3 \\ d_3 \\ K_3 \end{pmatrix} \rightarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
Range of argument 3	Direct register: Entire range Indirect register: Entire range Constant: –32768 ~ 32767			
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Flag	 Operation result Unchanged Unchanged Unchanged			



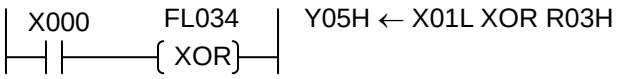
When input X000 is ON, the exclusive logical sum of BIN data of registers D103 and D102 is figured out and the result is stored in register D10D.



XOR truth table



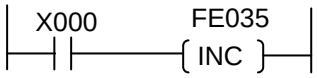
- When registers D_1, D_2 and D_3 are the byte registers,



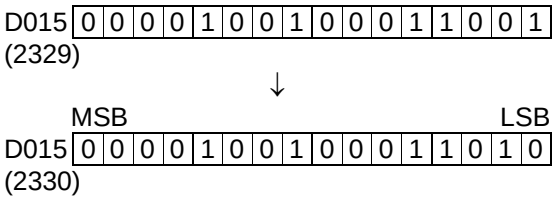
X01L									R03H								Y05H							
1	0	1	1	0	1	1	1	⊕	1	0	0	1	1	0	0	1	→	0	0	1	0	1	1	1	0

F*035 Increment

Symbol	Code	Argument Ag.1	Ag.2	Ag.3
INC	F*035	D_1 d_1		
Function	Increments the BIN data of register specified by Ag.1 and stores in register specified Ag.1.			
Content of operation	$\begin{pmatrix} D_1 \\ d_1 \end{pmatrix} + 1 \rightarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Flag	Low-order four (4) digits of operation result Unchanged			



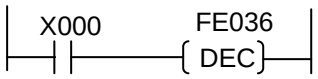
When input X000 has turned ON from OFF, the BIN data of register D015 is incremented and stored in register D015.



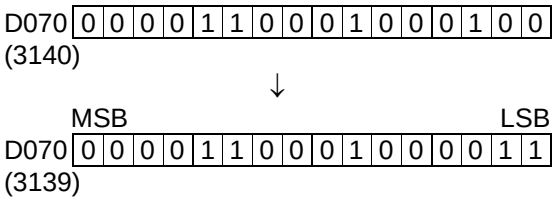
The data ranges from –32768 to 32767.

F*036 Decrement

Symbol	Code	Argument Ag.1	Ag.2	Ag.3
DEC	F*036	D_1 d_1		
Function	Decrements the BIN data of register specified by Ag.1 and stores in register specified Ag.1.			
Content of operation	$\begin{pmatrix} D_1 \\ d_1 \end{pmatrix} - 1 \rightarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Flag	Low-order four (4) digits of operation result Unchanged			

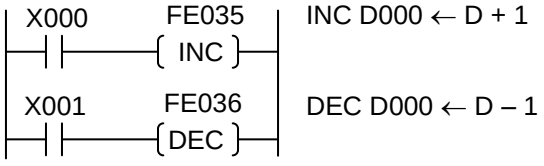


When input X000 has turned ON from OFF, the BIN data of register D070 is decremented and stored in register D070.



The data ranges from –32768 to 32767.

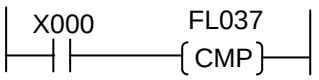
- Forward/backward counter as per increment or decrement (signed BIN data)



When X000 is ON, the counter value increments. Likewise, when X001 is ON, the counter value decrements. The data ranges from –32768 to 32767.

F*037 Unsigned Comparison

Symbol	Code	Argument Ag.1	Ag.2	Ag.3
CMP	F*037	D_1 d_1 K_1	D_2 d_2 K_2	B_3



Function

Compares the BIN data of register specified by Ag.1 with the BIN data of register specified by Ag.2, then stores the result in relay B3.

When input X000 is ON, the BIN data of register D100 is compared with the BIN data of register D005, and the result is output into relay R050, R051.

Content of operation

$$\begin{pmatrix} D_1 \\ d_1 \\ K_1 \end{pmatrix} \leftrightarrow \begin{pmatrix} D_2 \\ d_2 \\ K_2 \end{pmatrix} \rightarrow B_3$$

D100 (2339): 0 0 0 0 1 0 0 1 0 0 1 0 0 0 1 1

MSB LSB

D005 (9752): 0 0 1 0 0 1 1 0 0 0 0 1 1 0 0 0

↓

R050:ON, R051:OFF

Range of argument 1

Direct register: Entire range
Indirect register: Entire range
Constant: 0 ~ 65534

Range of argument 2

Direct register: Entire range
Indirect register: Entire range
Constant: 0 ~ 65534

Range of argument 3

Relay: Entire range

When the BCD data are compared, both are regarded as the BIN data.

If either register D_1 or D_2 is the byte register, the data of the byte register is taken as the 16-bit data whose high-order eight (8) bits are zero (0) and compared.

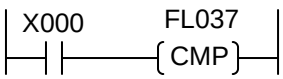
After operation

- Ag.1 content: Unchanged
- Ag.2 content: Unchanged
- Ag.3 content

Relay address: When B_3 is even number	Relay address: When B_3 is odd number	Ag.1 = Ag.2	Ag.1 < Ag.2	Ag.1> Ag.2
B_3	B_3–1	1	1	0
B_3+1	B_3	1	0	0

- Flag: Unchanged

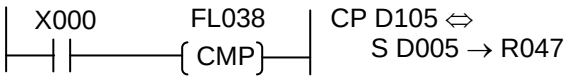
- Comparison of byte register with word register (BIN data)



D000	R02H	Result	R104	R105
(72) 0 0 0 0 | 0 0 0 0 | 0 1 0 0 | 1 0 0 0	(202) 1 1 0 0 | 1 0 1 0	<	1	0
(202) 0 0 0 0 | 0 0 0 0 | 1 1 0 0 | 1 0 1 0	(202) 1 1 0 0 | 1 0 1 0	=	1	1
(2304) 0 0 0 0 | 1 0 0 1 | 0 0 0 0 | 0 0 0 0	(174) 1 0 1 0 | 1 1 1 0	>	0	0

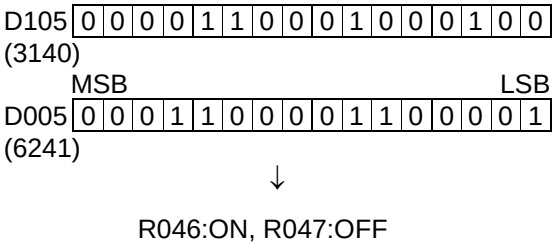
F*038 Signed Comparison

Symbol CMP	Code	Argument Ag.1	Ag.2	Ag.3
	F*038	D_1 d_1 K_1	D_2 d_2 K_2	B_3



Function	Compares the BIN data of register specified by Ag.1 with the BIN data of register specified by Ag.2, then outputs the result into relay B3.
Content of operation	$\begin{pmatrix} K_1 \\ d_1 \\ D_1 \end{pmatrix}_S \leftrightarrow \begin{pmatrix} K_2 \\ d_2 \\ D_2 \end{pmatrix}_S \rightarrow B_3$
Range of argument 1	Direct register: Entire range Indirect register: Entire range Constant: –327678 ~ 32767
Range of argument 2	Direct register: Entire range Indirect register: Entire range Constant: –327678 ~ 32767
Range of argument 3	Relay: Entire range
After operation • Ag.1 content • Ag.2 content • Ag.3 content	 Unchanged Unchanged
• Flag	Unchanged

When input X000 is ON, the BIN data of register D105 is compared with the BIN data of register D005, and the result is output into relay R046, R047.



↓

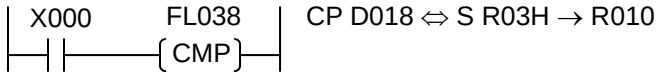
R046:ON, R047:OFF

When the BCD data are compared, they are regarded as the BIN data.

If either register D_1 or D_2 is the byte register, the data of the byte register is taken as the 16-bit data whose high-order eight (8) bits are zero (0) and compared.

Relay address: When B_3 is even number	Relay address: When B_3 is odd number	Ag.1 = Ag.2	Ag.1 < Ag.2	Ag.1> Ag.2
B_3	B_3–1	1	1	0
B_3+1	B_3	1	0	0

- Comparison of byte register with word register (BIN data)

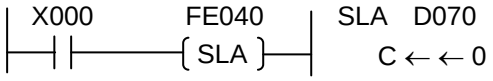


D180	R03H	Result	R010	R011
(–73) 1 1 1 1 | 1 1 1 1 | 1 0 1 1 | 0 1 1 1	(153) 1 0 0 1 | 1 0 0 1	<	1	0
(–174) 1 1 1 1 | 1 1 1 1 | 0 1 0 1 | 0 0 1 0	(184) 1 0 1 1 | 1 0 0 0	<	1	0
(–3140) 1 1 1 1 | 0 0 1 1 | 1 0 1 1 | 1 1 0 0	(83) 0 1 0 1 | 0 0 1 1	<	1	0
(206) 0 0 0 0 | 0 0 0 0 | 1 1 0 0 | 1 1 1 0	(206) 1 1 0 0 | 1 1 1 0	=	1	1
(456) 0 0 0 0 | 0 0 0 1 | 1 1 0 0 | 1 0 0 0	(226) 1 1 1 0 | 0 0 1 0	>	0	0

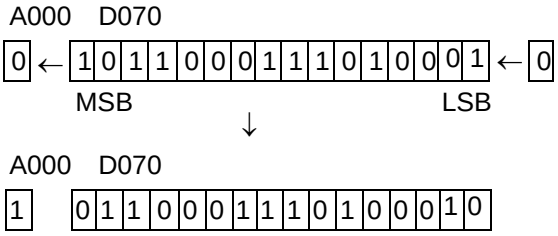
F*040 Arithmetic Left Shift

Symbol SLA	Code	Argument Ag.1	Ag.2	Ag.3
	F*040	D_1 d_1		

Function	Shifts the data of register specified by Ag.1 by one (1) bit to the left and sets the most significant bit (MSB) in the carry flag.
Content of operation	$C \leftarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix} \leftarrow 0$
Range of argument 1	Direct register: Entire range Indirect register: Entire range
After operation • Ag.1 content • Carry flag • Other flags	Data shifted by one (1) bit to the left. A000: Status of MSB of Ag.1 before shifted. A006, A007: Unchanged

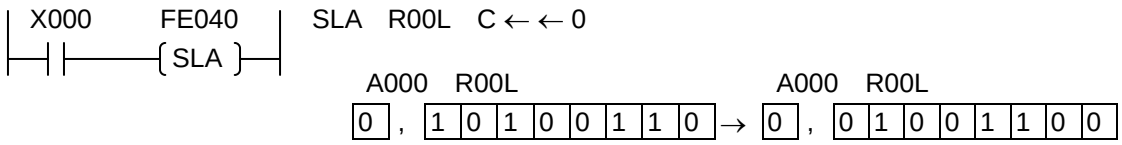


When input X000 has turned ON from OFF, the data of register D070 is shifted by one (1) bit to the left, the MSB is set in A000 and LSB is set to zero (0).

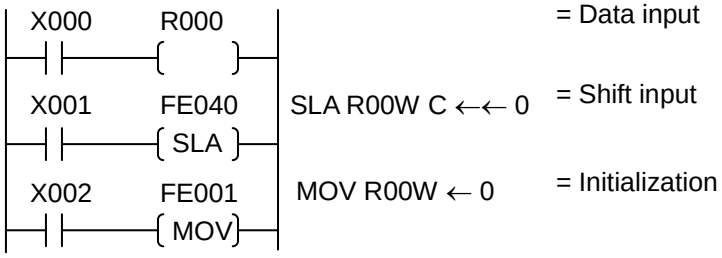


When register D_1 is the byte register, the high-order eight (8) bits are brought into zero (0) and shifted. As a result, the carry flag is turned OFF.

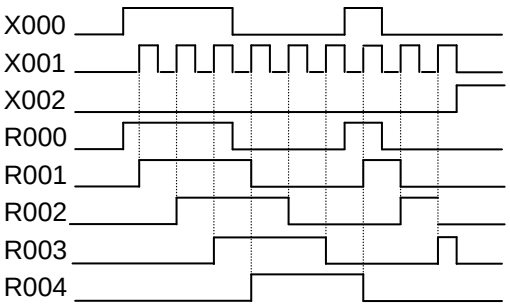
- When register D_1 is the byte register,



- Example of using register as one (1)-word length shift register:



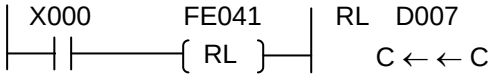
<Time chart>



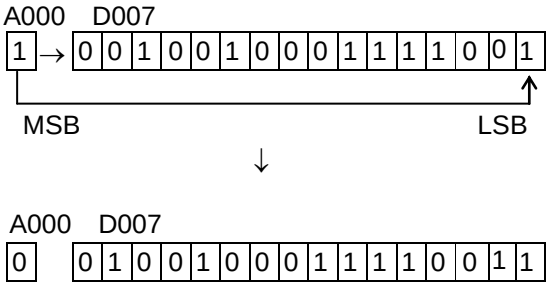
F*041 Left rotation

Symbol RL	Code	Argument Ag.1	Ag.2	Ag.3
	F*041	D_1 d_1		

Function	Shifts the data of register specified by Ag.1 by one (1) bit to the left and sets the carry flag in the least significant bit (LSB), then the MSB in the carry flag.
Content of operation	$C \leftarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix} \leftarrow C$
Range of argument 1	Direct register: Entire range Indirect register: Entire range
After operation • Ag.1 content • Carry flag • Other flags	Data shifted by one (1) bit to the left. LSB is the value of carry flag before shifted. A000: Status of MSB of Ag.1 before shifted. A006, A007: Unchanged

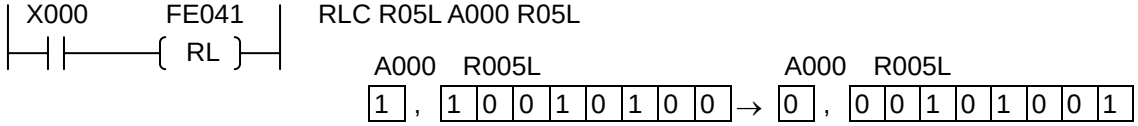


When input X000 has turned ON from OFF, the data of register D007 is shifted by one (1) bit to the left, the A000 data is set in the LSB, then the MSB in A000.

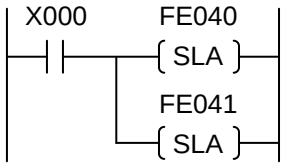


When register D_1 is the byte register, the high-order eight (8) bits are brought into zero (0) and shifted. After operation, the carry flag data turns OFF.

- When register D_1 is the byte register,



- When shifting one (1) word or more to the left,

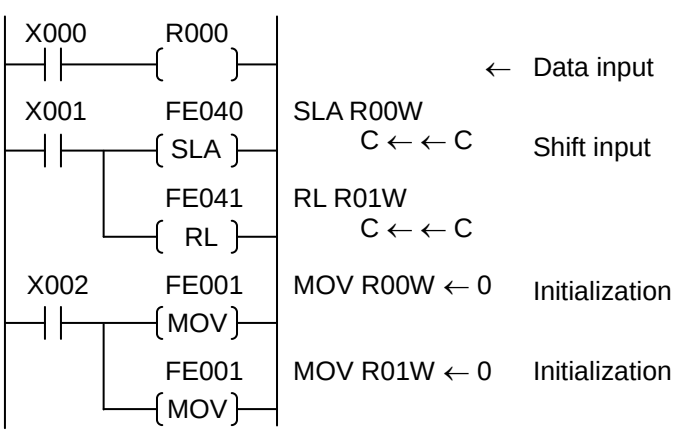


SLA D000 C ←← 0 = Left shift of low-order one (1) word.

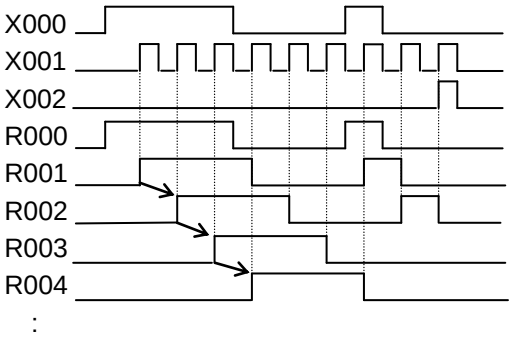
RL D001 C ←← 0 = Left rotation of high-order one (1) word.

□ A carry is executed by the carry flag.

- Example of using register as two (2)-word shift register:



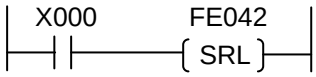
<Time chart>



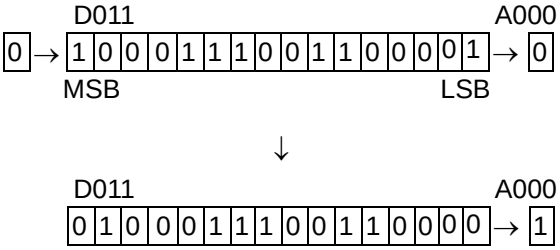
F*042 Right Shift

Symbol SRL	Code	Argument Ag.1	Ag.2	Ag.3
	F*042	D_1 d_1		

Function	Shifts the data of register specified by Ag.1 by one (1) bit to the right and sets the LSB in the carry flag.
Content of operation	$0 \rightarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix} \rightarrow C$
Range of argument 1	Direct register: Entire range Indirect register: Entire range
After operation • Ag.1 content • Carry flag • Other flags	Data shifted by one (1) bit to the right. A000: Status of LSB of Ag.1 before shifted. A006, A007: Unchanged

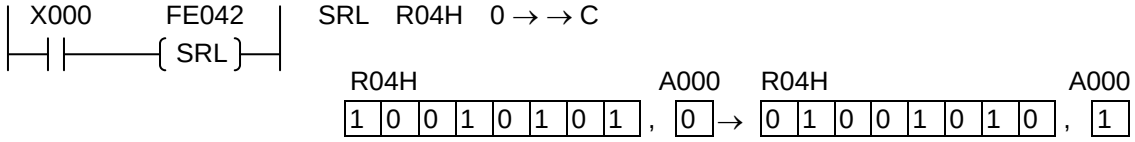


When input X000 has turned ON from OFF, the data of register D011 is shifted by one (1) bit to the right, the LSB is set in A000 and MSB is set to zero (0).

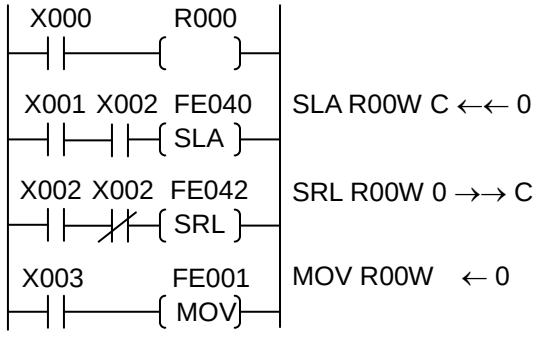


When register D_1 is the byte register, the high-order eight (8) bits are brought into zero (0) and shifted to the right.

- When register D_1 is the byte register,



- Example of using register as bi-directional one (1)-word length shift register:



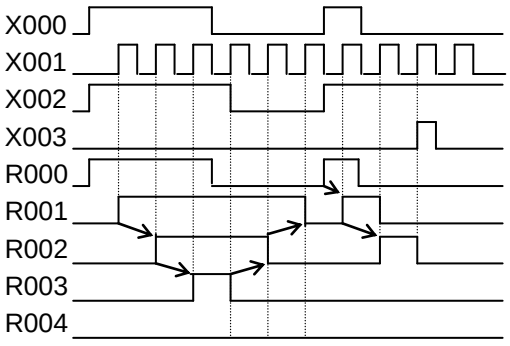
Data input

Forward shift

Reverse shift

Initialization

<Time chart>

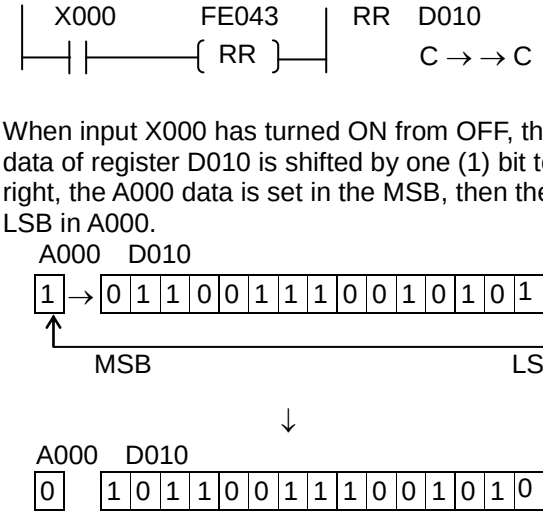


X000: Data signal
X001: Shift signal
X002: Forward/reverse signal
X003: Reset signal

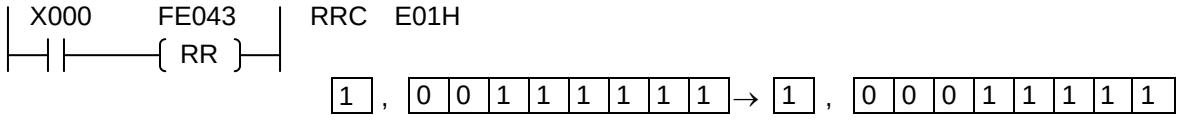
F*043 Right Rotation

Symbol RR	Code	Argument Ag.1	Ag.2	Ag.3
	F*043	D_1 d_1		

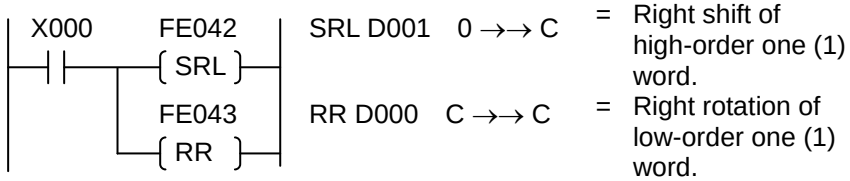
Function	Shifts the data of register specified by Ag.1 by one (1) bit to the right and sets the carry flag data in the MSB, then sets the LSB in the carry flag.
Content of operation	$C \rightarrow \begin{pmatrix} D_1 \\ d_1 \end{pmatrix} \rightarrow C$
Range of argument 1	Direct register: Entire range Indirect register: Entire range
After operation • Ag.1 content • Carry flag • Zero flag • Sign flag	Data shifted by one (1) bit to the right. MSB is the value of carry flag before shifted. A000: Status of LSB of Ag.1 before shifted. A006: Status of LSB of Ag.1 before shifted. A007: Unchanged



- When register D_1 is the byte register,

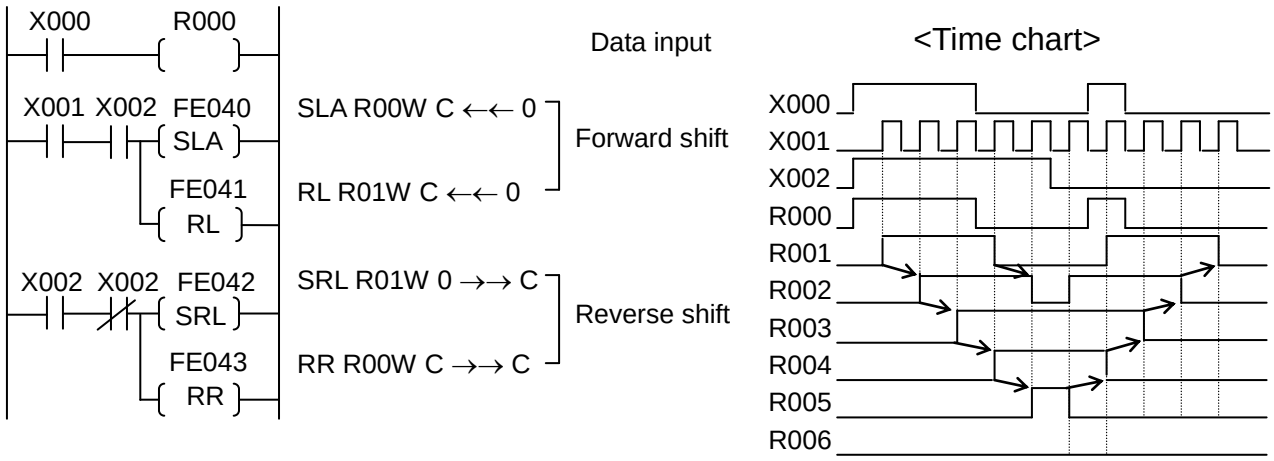


- When shifting one (1) word or more to the right,



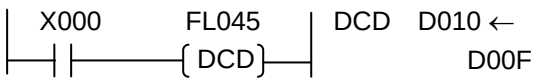
Note: A borrow is executed by the carry flag.

- Example of using register as bi-directional two (2)-word length shift register:

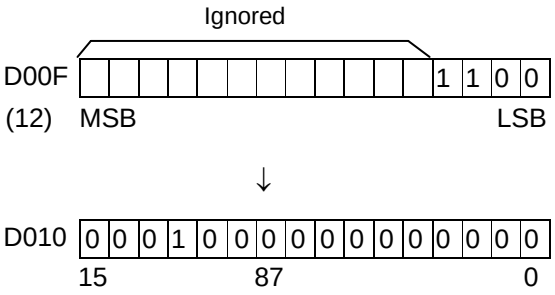


F*045 4 → 16 Decoder

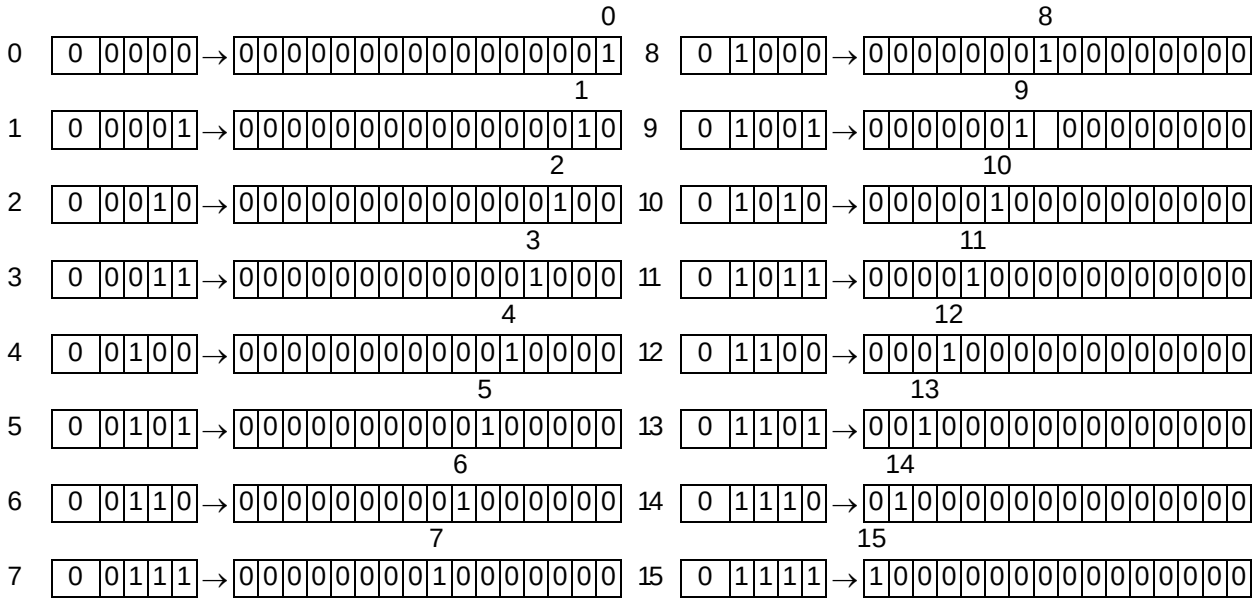
Symbol DCD	Code	Argument Ag.1	Ag.2	Ag.3
	F*045	D_1 d_1	D_2 d_2	
Function	Decodes the low-order four (4) bits of register specified by Ag.2 into 16 bits and stores in register specified by Ag.1.			
Content of operation	$\begin{pmatrix} D_2 \\ D_2 \end{pmatrix} \rightarrow DCD \begin{pmatrix} D_1 \\ d_1 \end{pmatrix}$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Direct register: Entire range Indirect register: Entire range			
After operation • Ag.1 content • Ag.2 content • Flag	 Operation result Unchanged Unchanged			



When input X000 is ON, the low-order four (4)-bit data of register D00F is decoded and stored in register D010 as the 16-bit data.

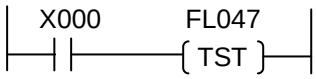


When register D_1 is the byte register, the low-order eight (8) bits of the decoded result are stored in register D_1.
If the data of register D_2 is 8 or over, zero (0) is stored.

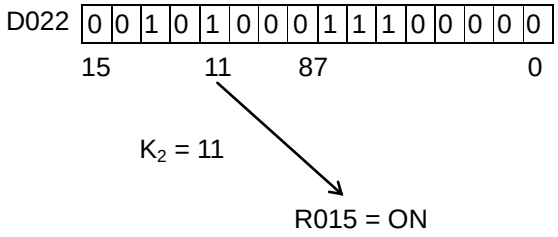


F*047 Bit Test (Constant Designation)

Symbol TST	Code	Argument Ag.1	Ag.2	Ag.3
	F*047	D_1 d_1	K_2	B_3
Function	Stores in relay specified by Ag.3 the bit of the data in register specified by Ag.1, which is at a position specified by constant K2.			
Content of operation	$\begin{pmatrix} D_1 \\ d_1 \\ d_1 \end{pmatrix}(K_2) \rightarrow B_3$			
Range of argument 1	Direct register: Entire range Indirect register: Entire range			
Range of argument 2	Constant: 0 ~ 15			
Range of argument 3	Relay: Entire range			
After operation • Ag.1 content • Ag.2 content • Ag.3 content • Flag	 Unchanged Unchanged Status of specified bit Unchanged			



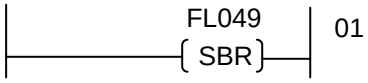
When input X000 is ON, the status of Bit 11 of the data in register D022 is stored in relay R015.



When register D_1 is the byte register, the high-order eight (8) bits are brought into zero (0) and tested.

F*049 Subroutine Start

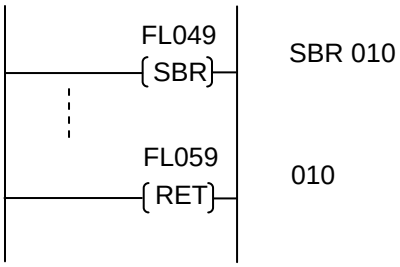
Symbol	Code	Argument		
		Ag.1	Ag.2	Ag.3
SBR	F*049	K_1		
Function	Defines the top of the subroutine program specified by constant Ag.1.			
Range of argument 1	Constant: 0 ~ 32			
After operation • Ag.1 content • Flag	 Unchanged Unchanged			



The top of the subroutine program is indicated.
A subroutine program can be written in any area of program.

Note: When programming a subroutine program, include the subroutine start instruction at the top and the subroutine return instruction at the end of a program.

Example:



When you have to repeat the same processing in the program, register it beforehand as a subroutine, and you can execute it by calling this subroutine, when necessary. Thus, you can reduce the number of program steps and make the program easy-to-see.

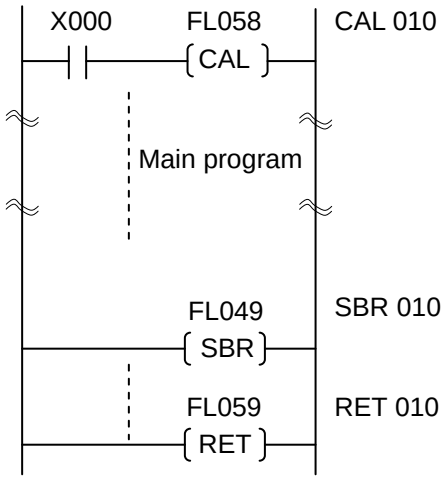
Data from the subroutine start instruction (F*049) to the subroutine return instruction (F*059) is registered as one (1) subroutine. Up to thirty-two (32) subroutines can be registered.
Set a subroutine number in the argument of subroutine start instruction. Likewise, set the subroutine number in the argument of subroutine return instruction.
To call a registered subroutine, use the subroutine call instruction (F*058).

Cautions on use of subroutine:
Though there is not any limitation on the call of another subroutine from one subroutine (i.e., nesting), DO NOT execute the same subroutine. Otherwise, the system will operate incorrectly.

F*058 Subroutine Call

Symbol	Code	Argument		
		Ag.1	Ag.2	Ag.3
CAL	F*058	K_1		
Function	Executes the subroutine program defined by the constant specified by Ag.1.			
Content of operation	CAL K_1			
Range of argument 1	Constant: 0 ~ 32			
After operation • Ag.1 content	Unchanged			

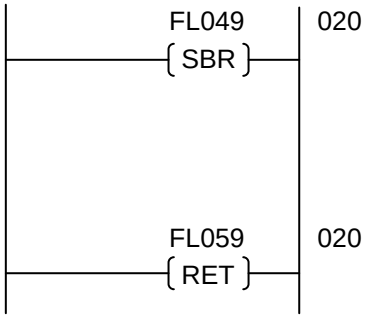
When input X000 is ON, the subroutine program 010 is executed.



F*059 Subroutine Return/Program End

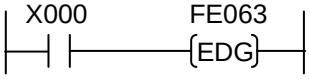
Symbol	Code	Argument		
		Ag.1	Ag.2	Ag.3
RET	F*059	K_1		
Function	Specifies the end of the subroutine program designated by the constant of Ag.1.			
Content of operation	RET			
Range of argument 1	Constant: 0 ~ 32			
After operation • Ag.1 content	Unchanged			

The end of subroutine program 20 is executed.

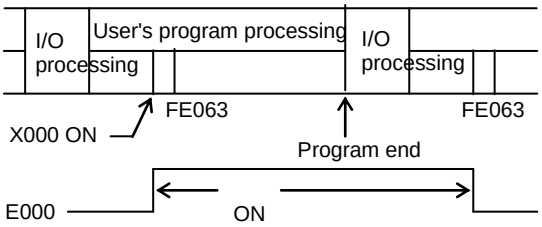


F*063 1-Scan ON

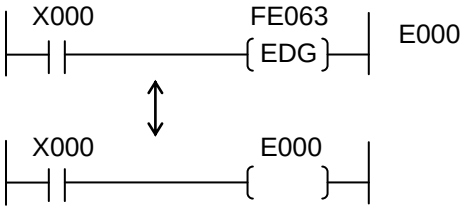
Symbol EDG	Code	Argument Ag.1	Ag.2	Ag.3
	FE063	B_1		
Function	Turns ON the relay specified by Ag.1 during one (1) scan.			
Content of operation	B_1: ON			
Range of argument 1	Relay: Entire range			
After operation • Ag.1 content • Flag	Turned ON relay B_1 for one (1) scan. Unchanged			



When input X000 has turned ON from OFF, relay E000 is turned ON, which is turned OFF after this instruction is executed in the next scan.



Note: This instruction works exactly in the same manner as differentiating relay E***.



In terms of the processing speed, differentiating relay E*** is faster than this instruction. It is recommended, therefore, to use differentiating relay E*** where possible.

This instruction cannot be used as F[L]063.

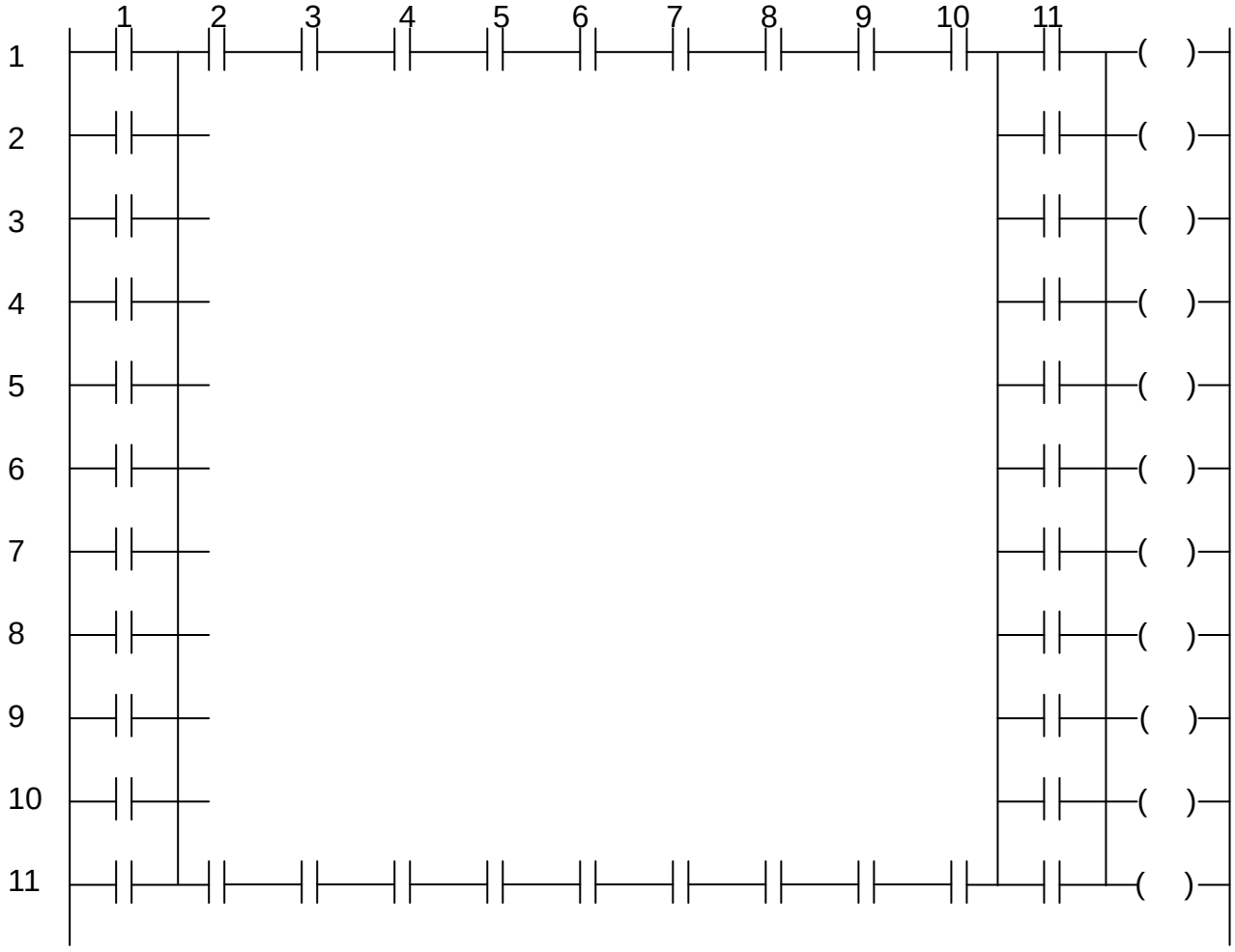
7.6 Programming

The PLC may perform the operation slightly different from the actual relay circuit because it successively reads data from the program memory and performs instructions one by one. Additionally, a run-around preventing diode is not required, which is needed for the relay board, and the number of auxiliary contacts is unlimited.

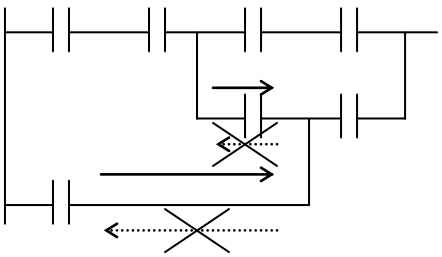
Design the circuit efficiently by fully understanding the difference in circuit design between the PLC and relay board.

7.6.1 Limitation on Creating Circuit

(1) One (1) circuit must consist of (11 columns × 1 output) × 11 lines or less as shown below.

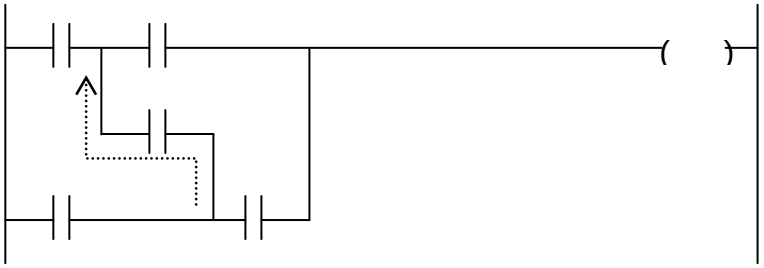


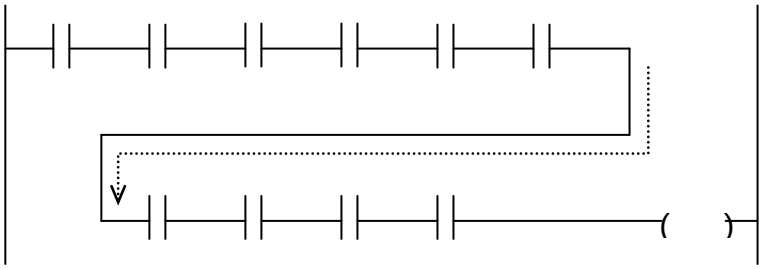
(2) Current in the same circuit only flows from left to right. It will not flow from right to left. That is, the circuit includes a one-way diode at each contact.

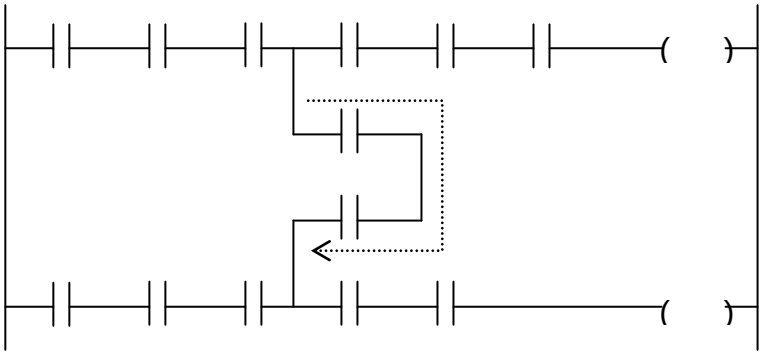


→ Current flows in the direction shown by
⇠ the full-line arrow mark, but not in the dotted-line arrow mark direction.

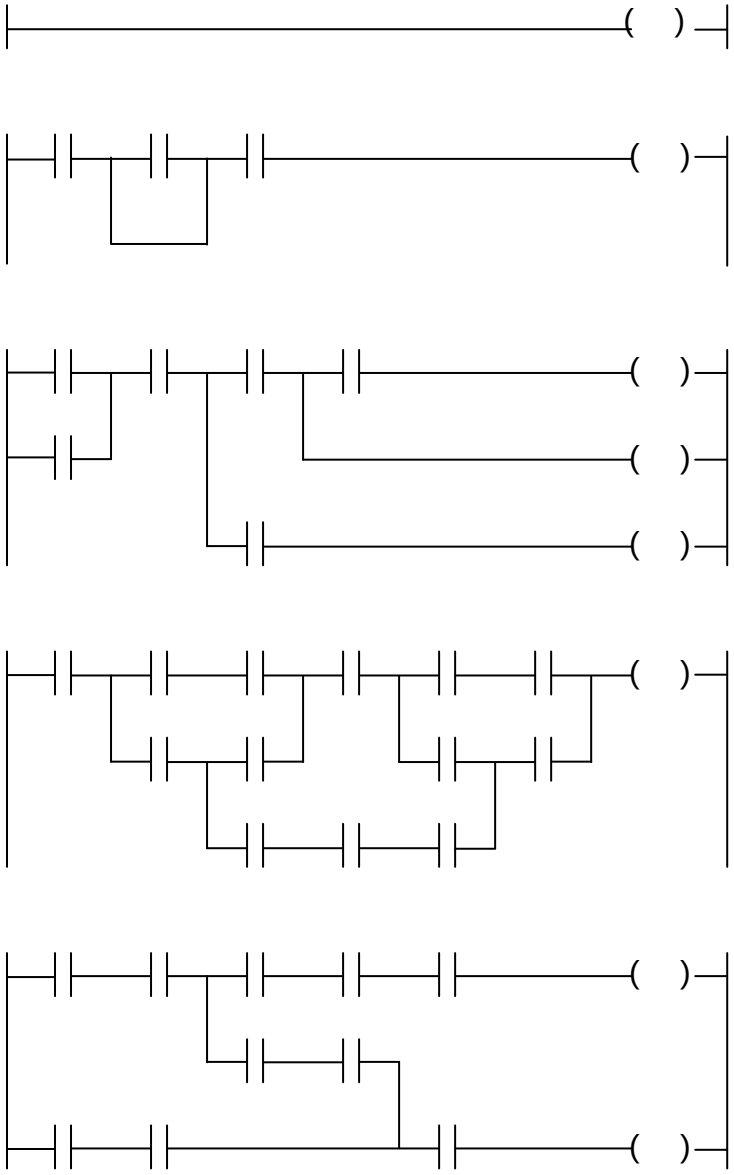
For the current below, the current in the dotted-line arrow mark direction is not existent.



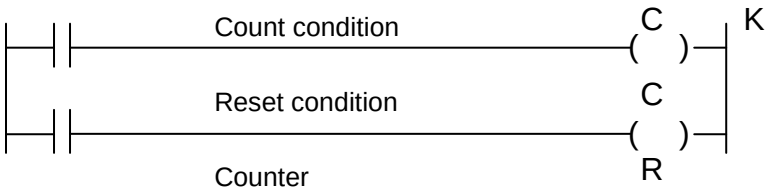


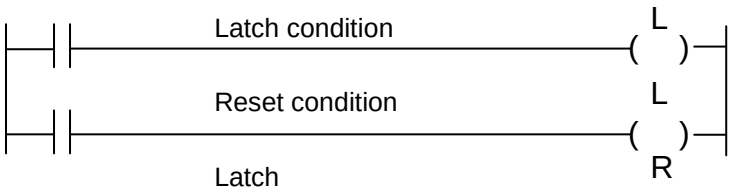


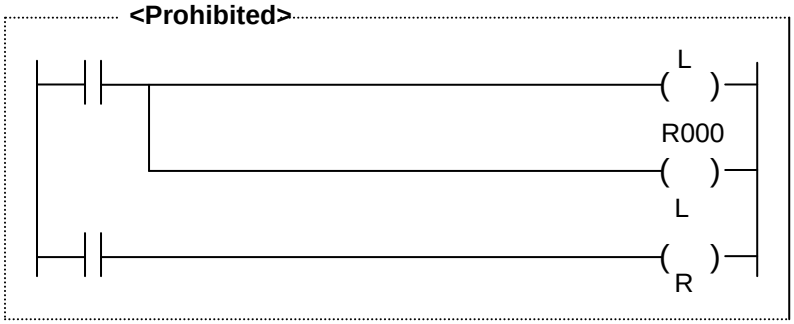
(3) No limitation is imposed on the circuit if items (1) and (2) are considered.



(4) The pair coil is considered as one (1) circuit.
The counter and latch are considered as one (1) circuit, including the reset condition. It is not possible to include another coil between the pair coil.







(Pair coil error occurs.)

7.6.2 Influence by Program Sequence

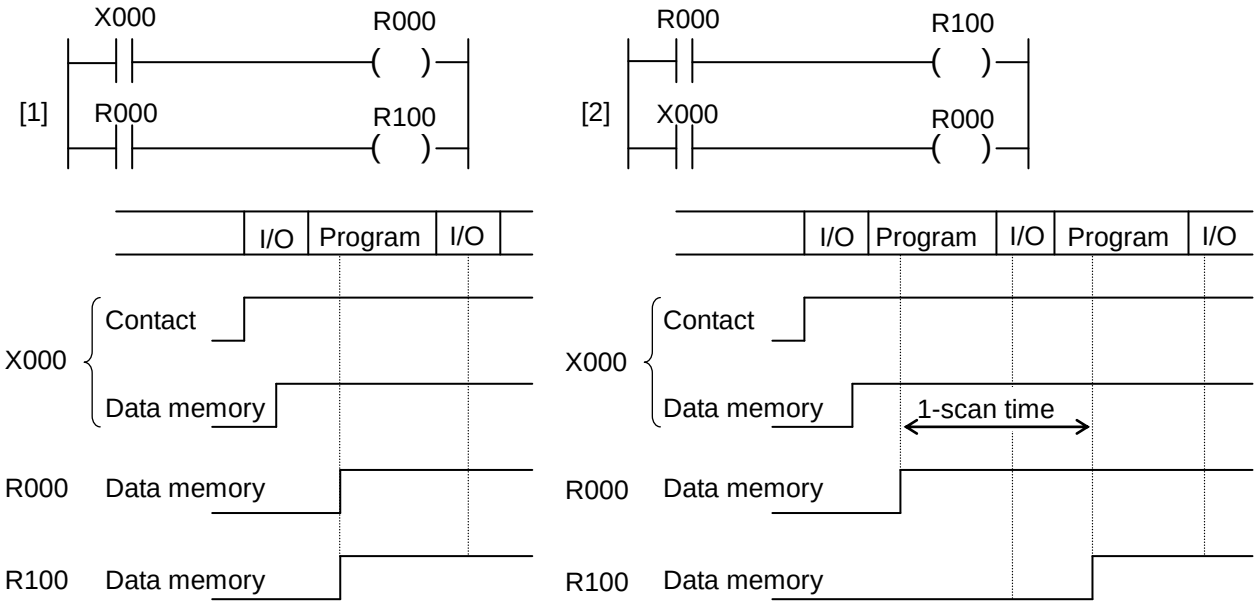
The PLC performs operation from the top to the end (P. END instruction) of the program repeatedly. (Cyclic scanning operation system)

For this reason,

(a) The operation is performed from the top to bottom of the program for each circuit (i.e., each circuit block with AND and OR connections).

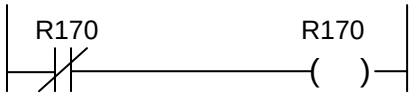
(b) The operation is performed from the left to right for each column in one (1) circuit. (The operation of the input part is performed before the operation of the output part.)

(1) When the program sequence is changed, the operation may differ.



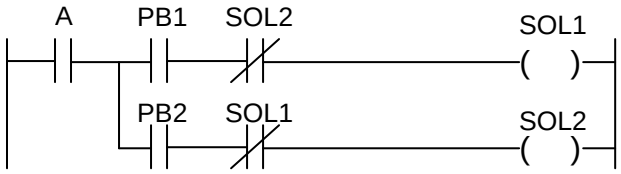
In program [1] above, R000 and R100 are turned ON within the same scan when X000 is turned ON. In program [2], however, R100 is turned ON one (1) scan later. This is because the status of contact R000 is changed one (1) scan later due to the presence of contact R000 before coil R000.
Generally, the status of the contact programmed before the coil is changed at the next scan to the one in which the coil state changed. There is a programming technique to use such scan delay constructively.

Example:

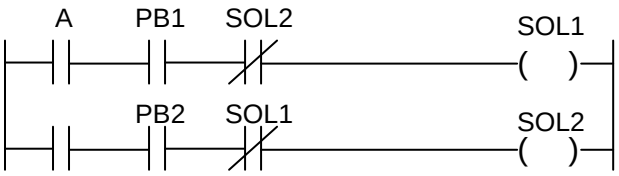


This is an oscillation circuit which repeats ON and OFF every scan.

(2) If an interlock is made in one (1) circuit, malfunction may result in.

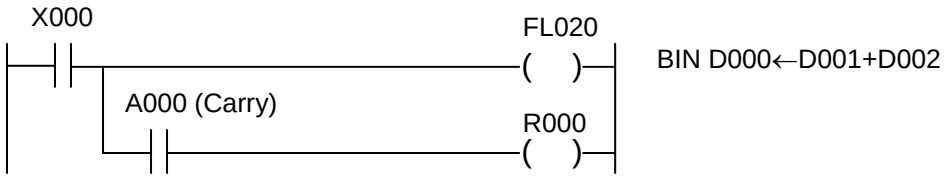


Even if an interlock is made mutually in the above circuit, SOL1 and SOL2 repeat ON and OFF every scan when PB1 and PB2 are pressed at the same time.
This is because the operation is performed from the left to right in the same circuit according to the principle in Item (b) of Para. 7.6.2. To prevent malfunction, the circuit must be divided into the two (2) parts.

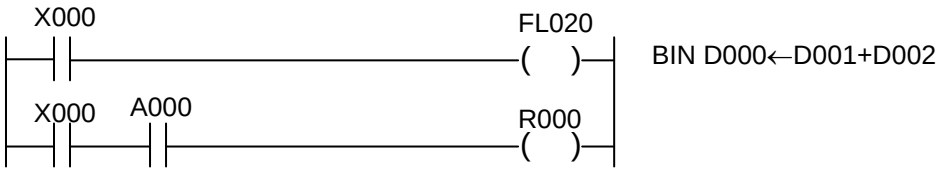


When PB1 and PB2 are pressed at the same time, the circuit in which priority is given to SOL1 is formed.

(3) If an application instruction with change in flag status is used together with a flag contact in the same circuit, the flag result is incorrectly reflected.



The above is the circuit to output an error as a result of operation (overflow) to R000. As the input part is operated before the output part according to the principle in Item (b) of Para. 7.6.2, the carry result after addition is not reflected. It is necessary to divide the circuit into the two (2) parts.



For the application instruction that performs operation with carry, however, multi-output circuits pose no problem because the flag is processed automatically and internally.

7.6.3 Influence by Input/Output Processing

The input/output relay can be used as a byte or word register. When it is used as a source or destination, however, all data are read and written in and out of the data memory. At the time when the application instruction has been executed, the result will not be output to the I/O device.

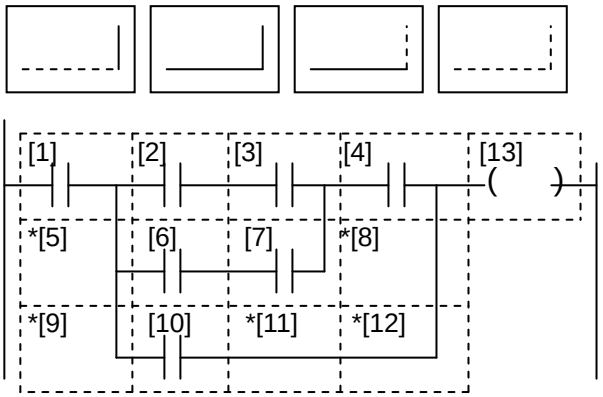
The final result of the operation is processed in batch and exchanged between the I/O device and data memory. At this time, the result is actually output to the I/O device.

7.6.4 Promoting Programming Efficiency

The TCmini need not consider stacks unlike the PLC with serial Boolean algebra processing system, because it has the special direct ladder-diagram input system. When the characteristics are fully understood, however, a program becomes more efficient with fewer instruction words.

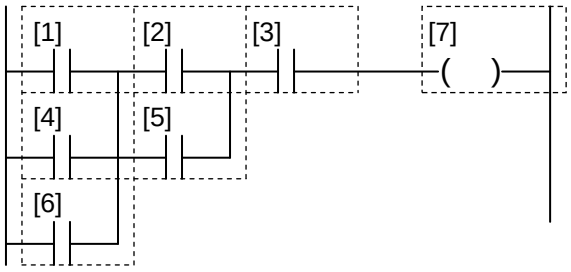
(1) How to count the number of instruction words

The following sections necessary for constructing a circuit are counted as one (1) word.

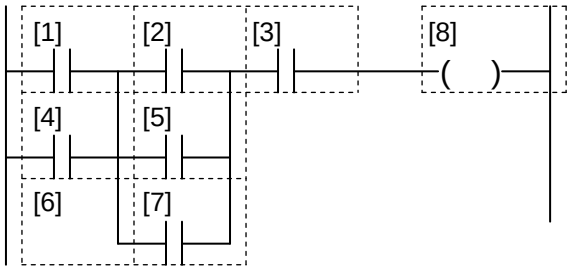


Each section enclosed by dotted line is counted as one (1) word. The line required for constructing a circuit as marked "*" is also counted as one (1) word. In the above example, there are 13 words.

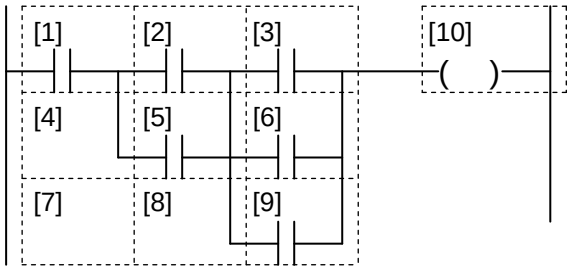
(2) Generally, if the circuit is left-down, a program becomes efficient with fewer number of instruction words.



[1] Left-down circuit, 7 words



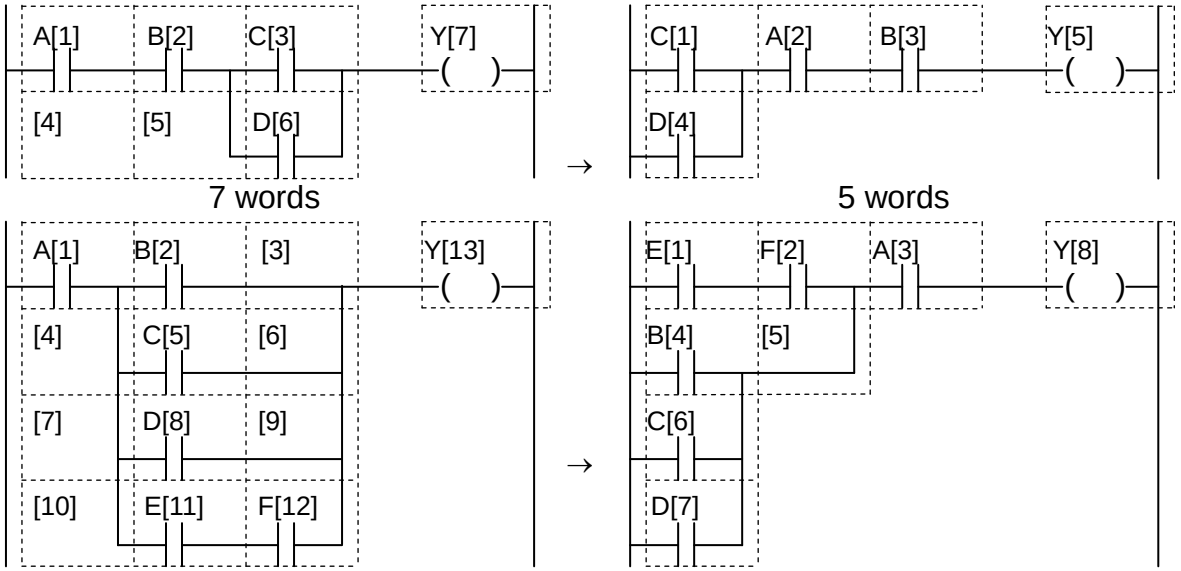
[2] Middle-down circuit, 8 words



[3] Right-down circuit, 10 words

Left-down circuit [1] shown above is more efficient than circuits [2] and [3].
The number of instruction words of circuit [1], [2] and [3] is seven (7), eight (8) and ten (10), respectively.
All input instructions including the contact and connection ones are counted as one (1) word or two (2) words.
The output (coil) instructions including the application instruction are counted as one (1) word to five (5) words. For details, see Para. 7.2.

The following circuits require fewer words when formed left-down.



Section 8 TCPRGOS

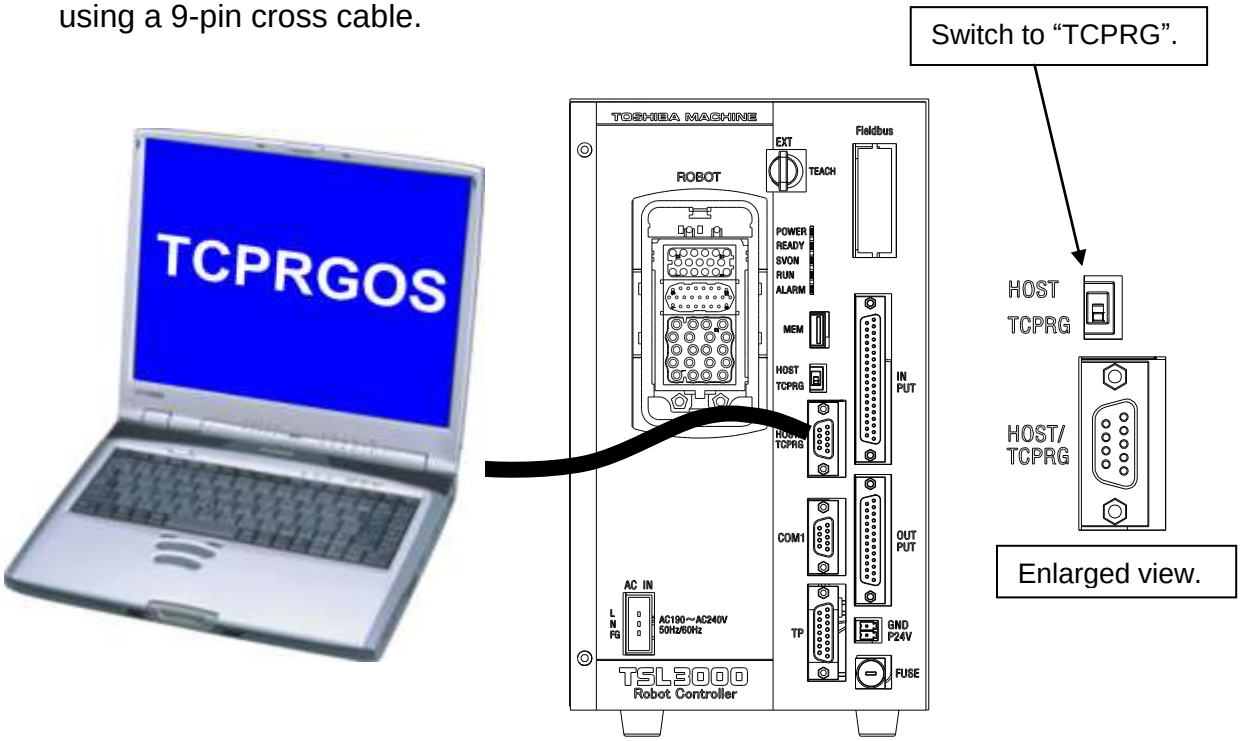
TCPRGOS–W(E) is the sequence program development tool which operates on your personal computer.

8.1 Operating Environment of TCPRGOS

Items	Conditions
CPU	Personal computer incorporating CPU whose capacity is Pentium 200 MHz or over (Use of Pentium II 233 MHz or over is recommended.).
Operating system	Windows98 or WindowsNT 4.0/2000/XP
Hard disk capacity	Free space of 20 M byte or over
Minimum available memory	Windows98: 16 MB or over (Use of 32 MB or over is recommended.) WindowsNT: 32 MB or over Windows 2000: 128 MB or over Windows XP: 256 MB or over
CD drive	Required for installing TCPRGOS.
Display	Display that can be connected with computer and designed for Windows98 or WindowsNT with resolution of 640 × 480 dots or over. (Recommendable resolution is 1024 × 768 dots or over.)
RS232C serial communication port	One (1) port
Others	Keyboard, mouse, printer, FD, modem (for remote maintenance system) that can be connected with computer and designed for Windows98 or WindowsNT.

8.2 Connection

Connect the TSL3000 TCPRG port with the COM port of your personal computer, using a 9-pin cross cable.



PC COM port TSL3000 TCPRG port

Cross-connecting cable

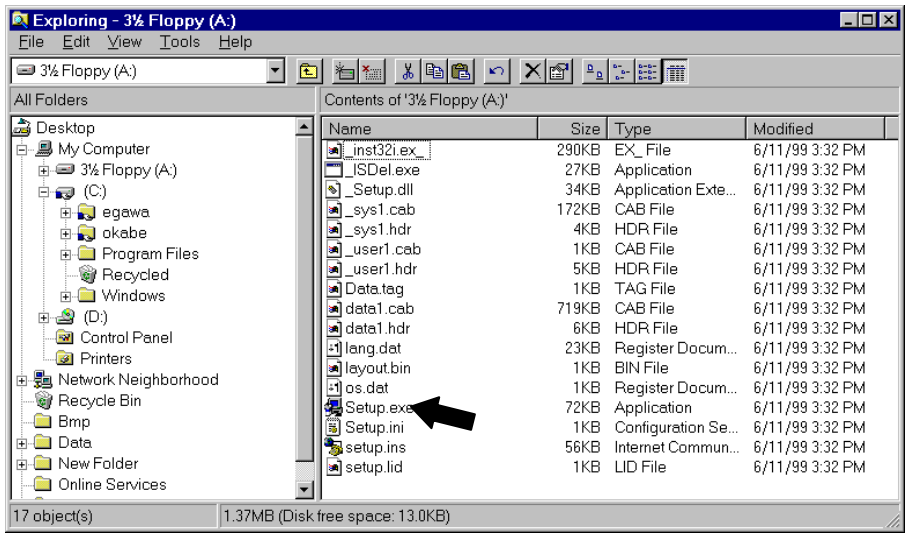
DOS/V D–sub 9-pin female		TSL3000 D–sub 9-pin female
1		1
RXD 2		2 RXD
TXD 3		3 TXD
DTR 4		4 DTR
FG 5		5 FG
6		6 DSR
RTS 7		7 RTS
CTS 8		8 CTS
9		9
Frame		Frame

8.3 How to Install the TCPRGOS

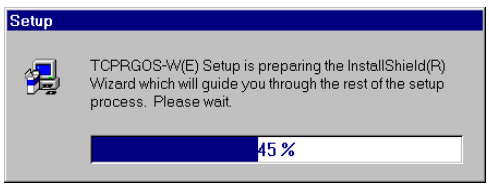
CAUTION	Before upgrading, be sure to uninstall the TCPRGOS–W(E) of the previous version. For details, see Section 6.

This paragraph describes the setup procedures when the floppy drive is set as the "A" drive.

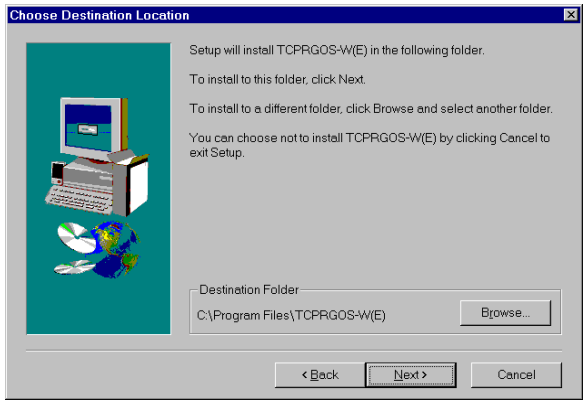
- Start up the Explorer and double-click "SETUP.EXE" in the "A" drive.

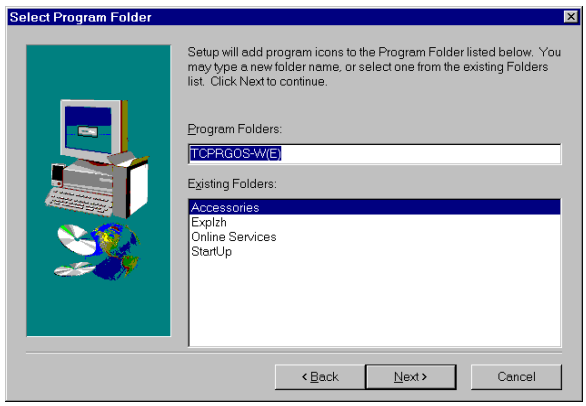


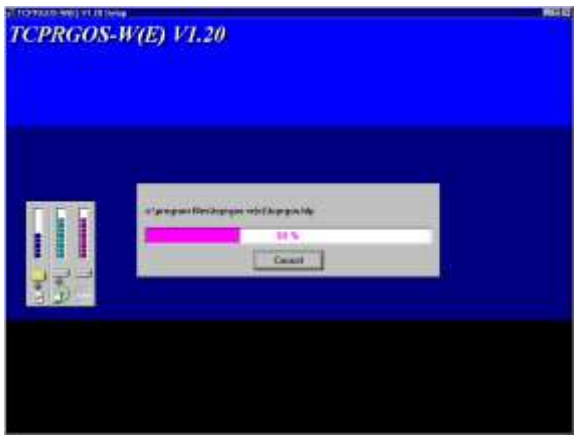
- The TCPRGOS–W(E) V*.** Installer starts. Press the [Next] button.



- Now, execute the operation according to the directions of the Installer.







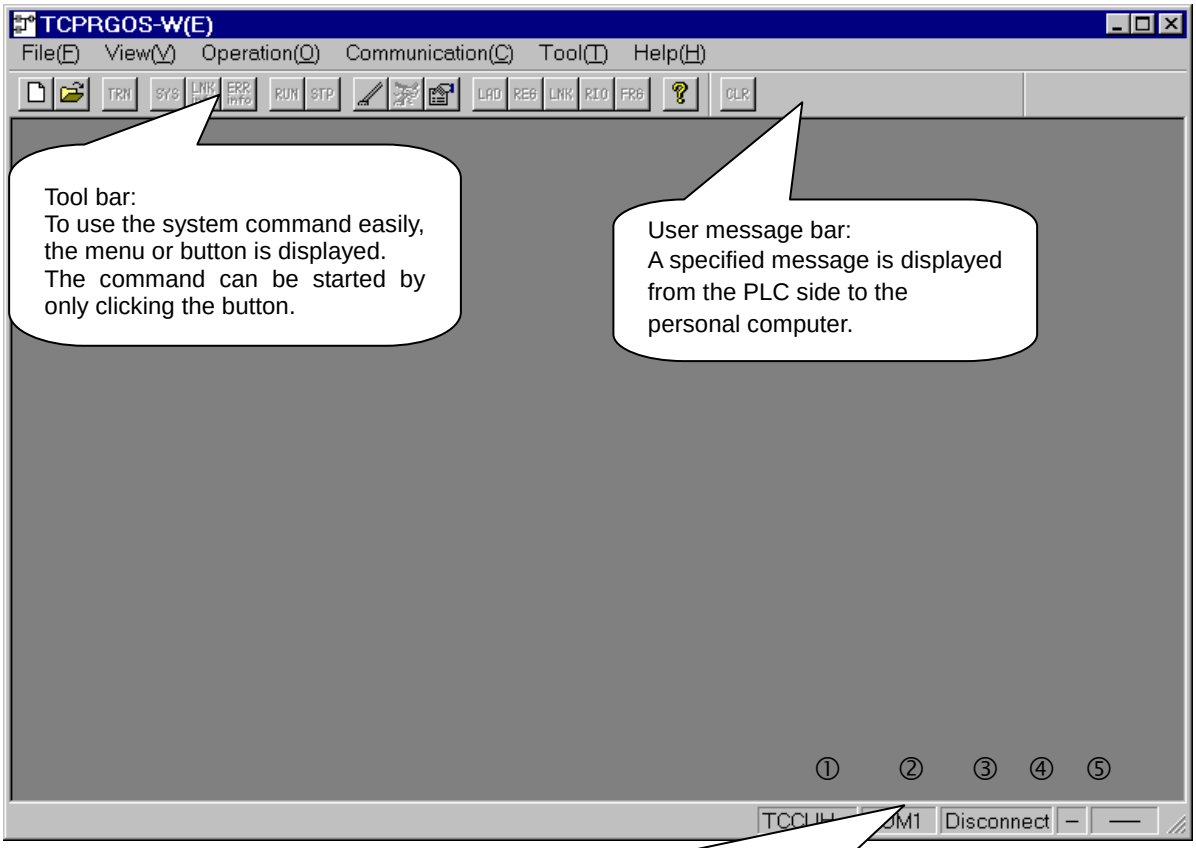
- Click the [Finish] button, and the TCPRGOS–W(E) has been installed.

8.4 Starting the TCPRGOS

◆ Starting TCPRGOS–W

① Double-click the TCPRGOS–W icon on the desktop.

② The TCPRGOS–W starts.



Status bar:

Information on current system connection status, message from application to user, etc. is displayed.

① Connected PLC model is displayed.

② Connection method is displayed, which is specified in communication setup.

③ Connection/disconnection status is displayed.

④ Connected PLC number is displayed.

⑤ RUN/STOP status of PLC is displayed.

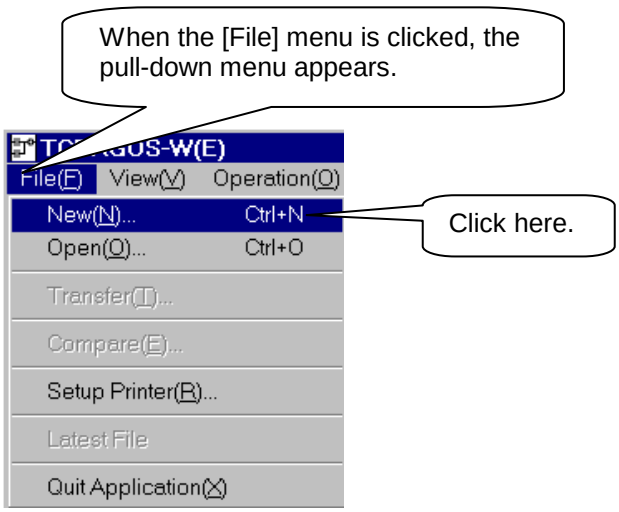
8.5 Creating a Circuit

◆ Creating a New Ladder Program and Saving in a File

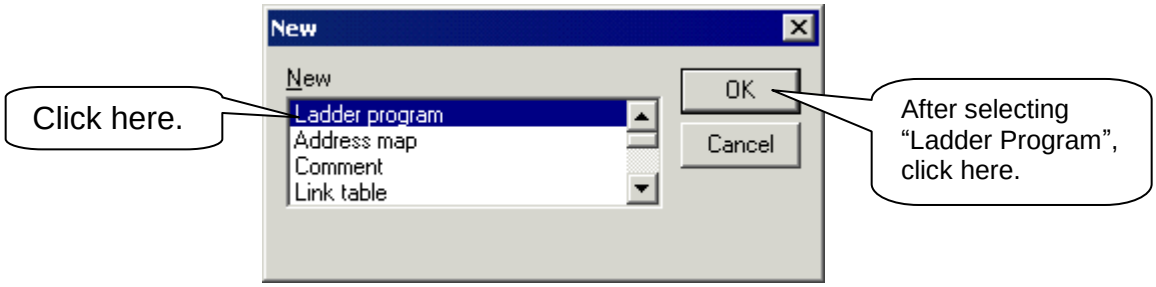
★ Program to be created.

8.5.1 Start of New Ladder Editor

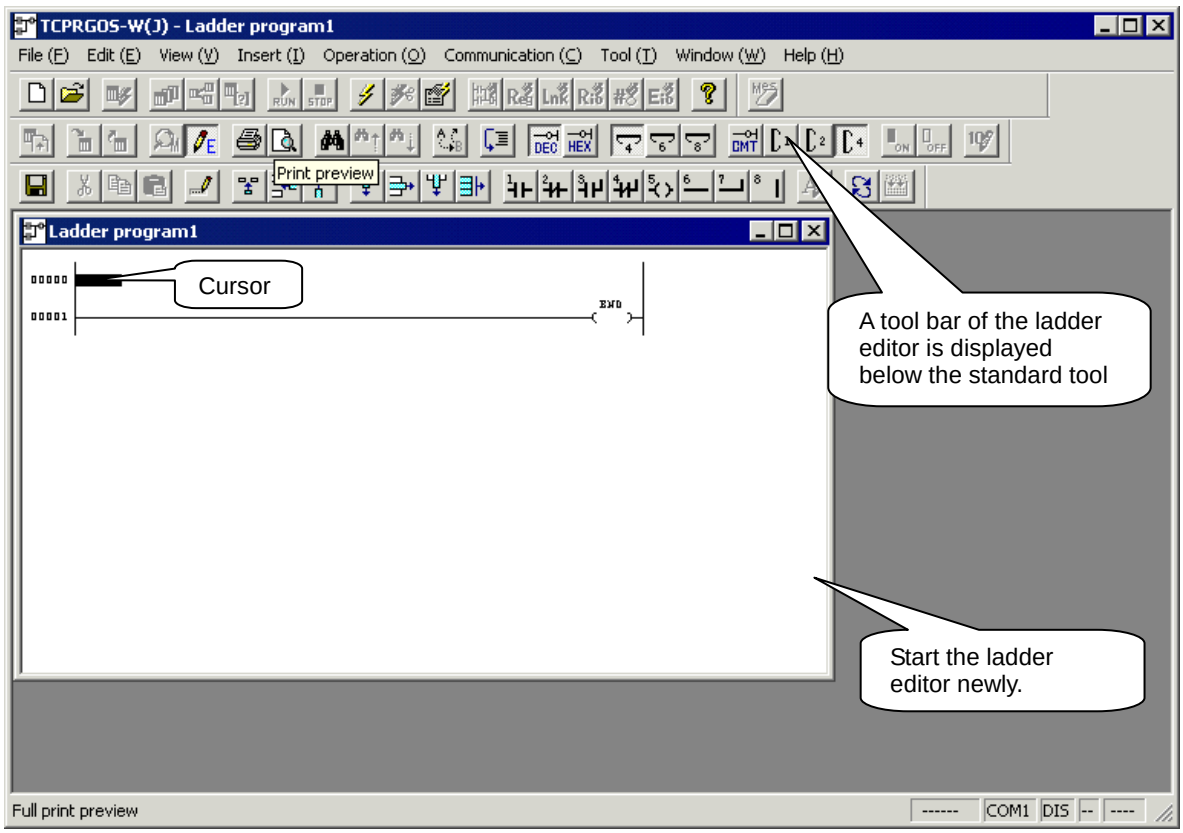
① Click [New] from the [File] menu.



② As the [New] dialog box appears, click [Ladder Program] from the [New] box.

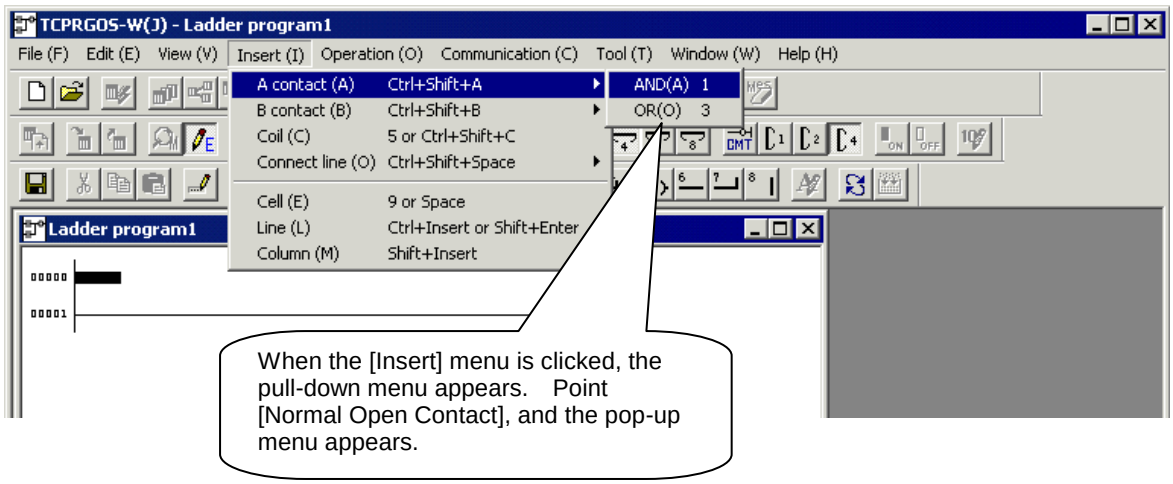


③ The ladder editor starts newly.

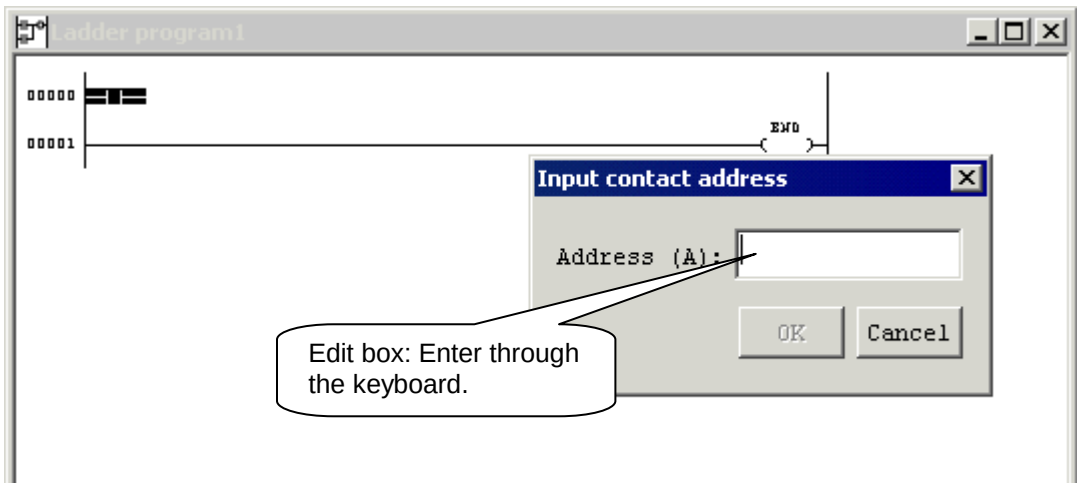


8.5.2 Creation of Circuit

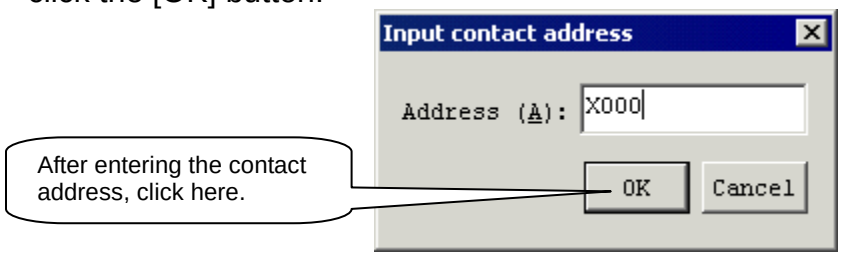
① Point [Normal Open Contact] from the [Insert] menu, then click [AND].



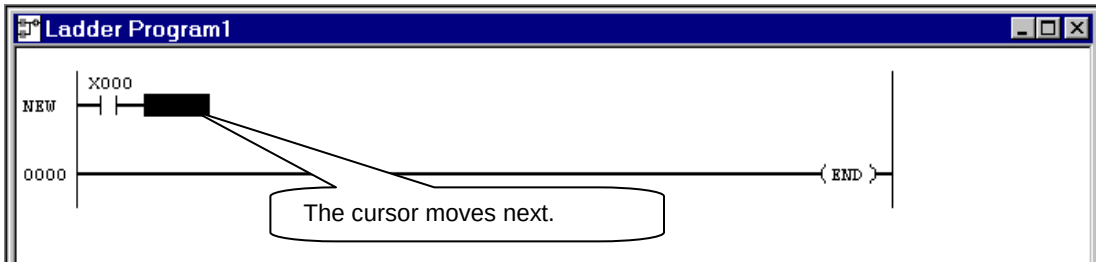
② The symbol of normal open contact is entered at the cursor position and the contact address input dialog box appears.



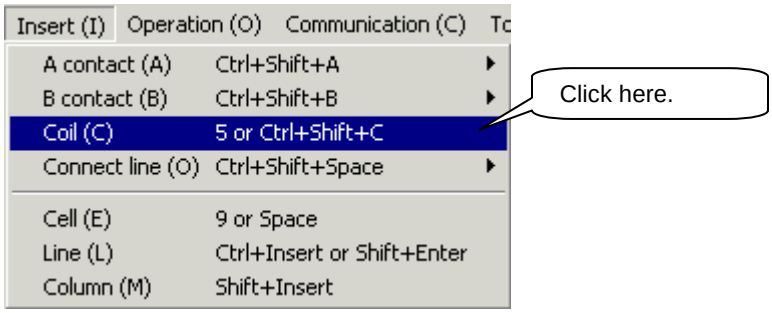
③ Move the cursor to the contact address edit box and enter “X000”. Then click the [OK] button.



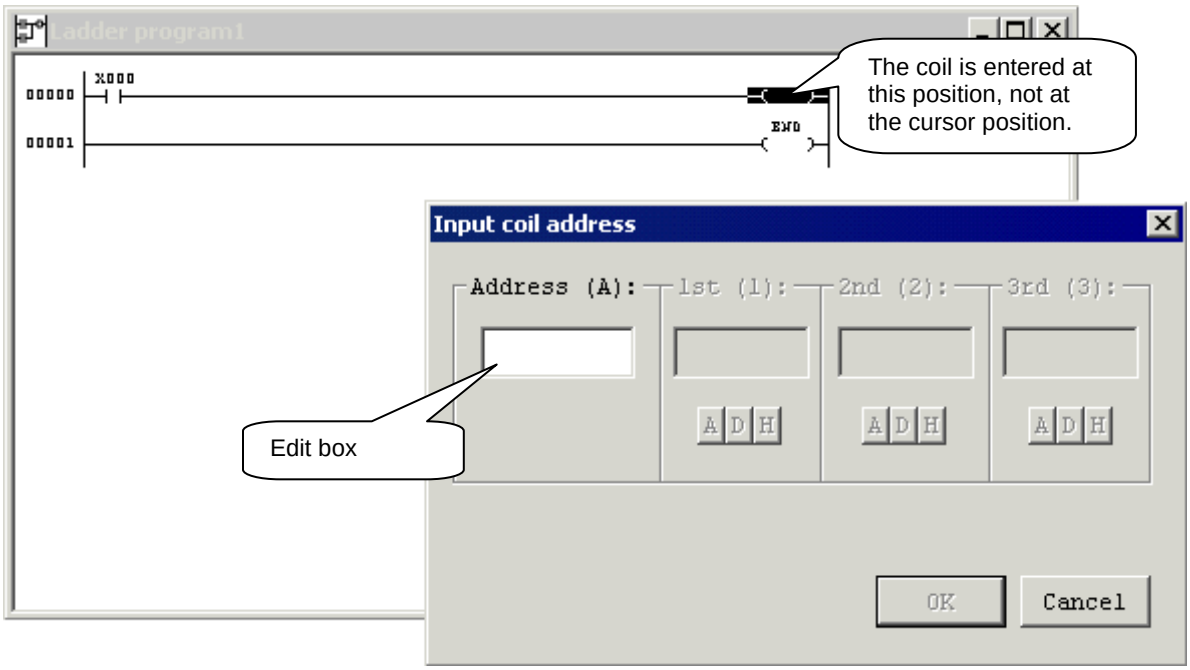
④ The normal open contact of address X000 is entered.



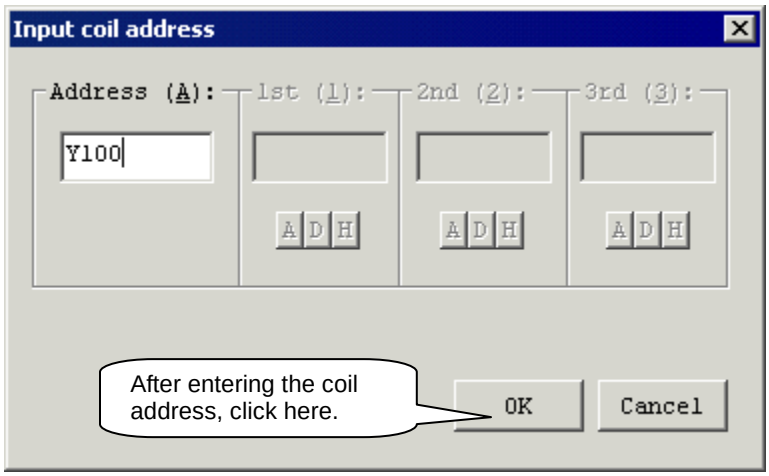
⑤ Click [Coil] from the [Insert] menu.



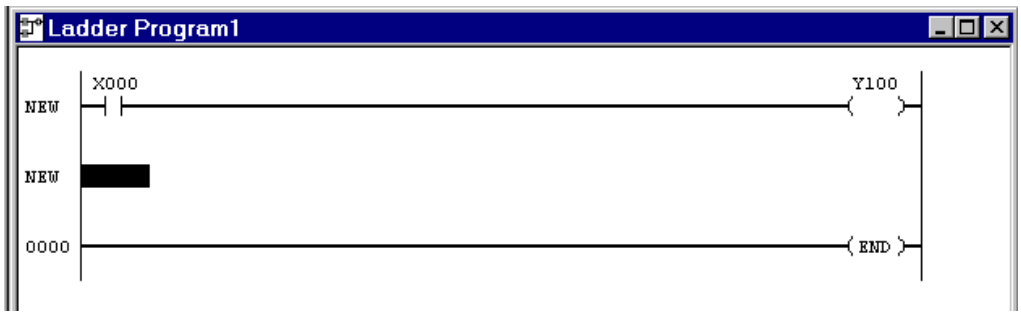
⑥ The symbol of coil is entered and the coil input dialog box appears.



⑦ Enter “Y100” in the coil address edit box and click the [OK] button.



⑧ The following circuit is created.



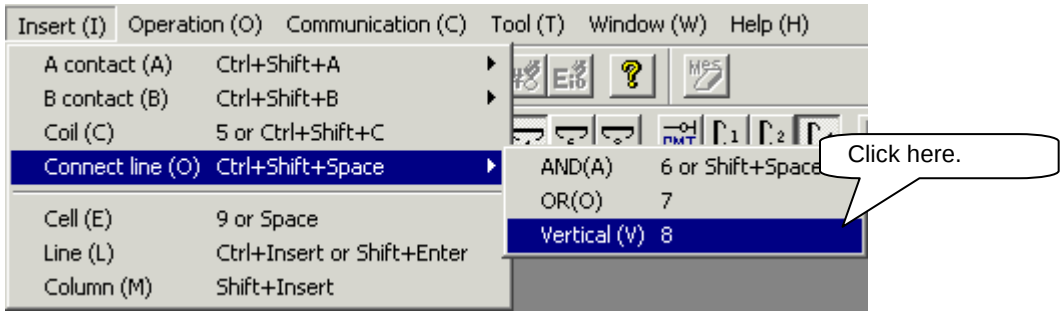
One Point Advice

⊙ Cursor movement
If a desired position is clicked by means of the mouse, the cursor moves to that position. The cursor can also move vertically and horizontally by means of the arrow keys. The symbol other than coil is entered at the cursor position.

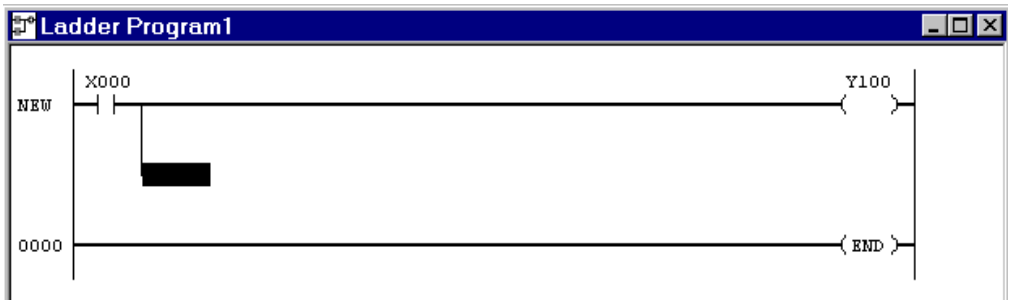
⊙ Address change
When the mouse is moved to an already entered contact or coil and double-clicked, the address input dialog box appears. Move the cursor to the edit box where the address is to be entered, modify the address, then click the [OK] button.

8.5.3 Input of Perpendicular Line Symbol

① Point [Connecting Line] from the [Insert] menu and click [Perpendicular].

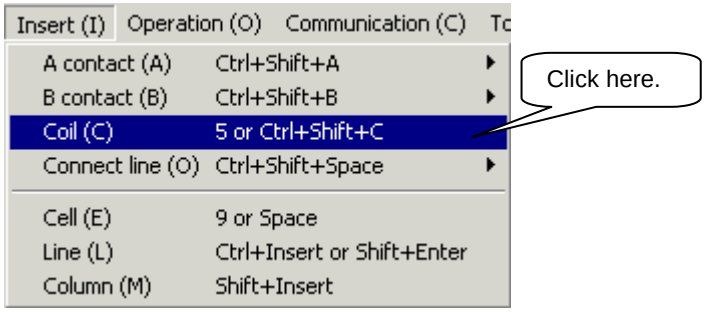


② The symbol of perpendicular line is entered.



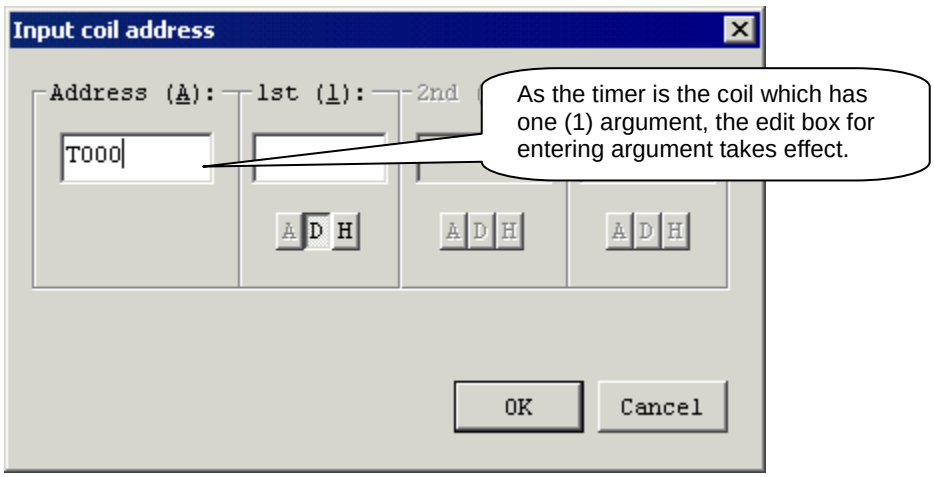
8.5.4 Input of Timer in Coil

① Click [Coil] from the [Insert] menu.

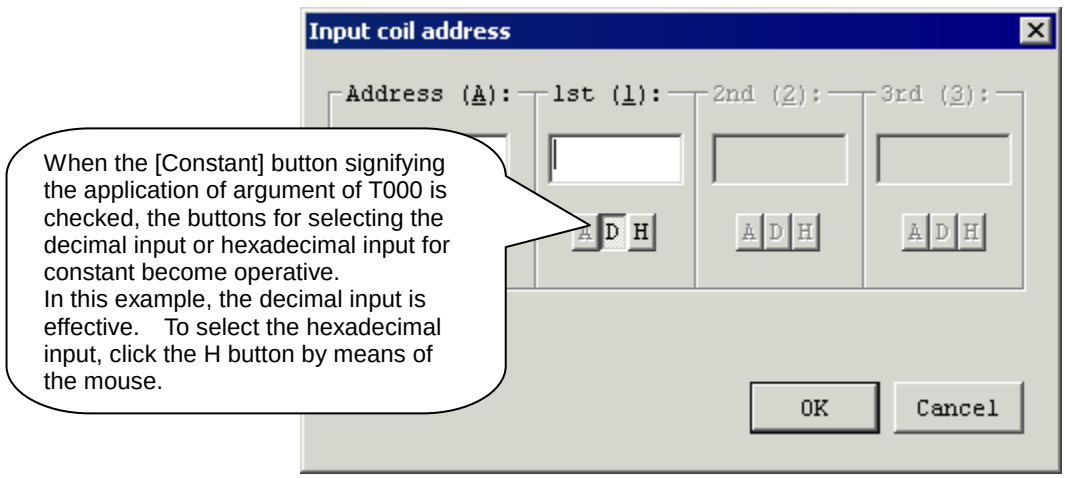


② The symbol of coil is entered and the coil address input dialog box appears.

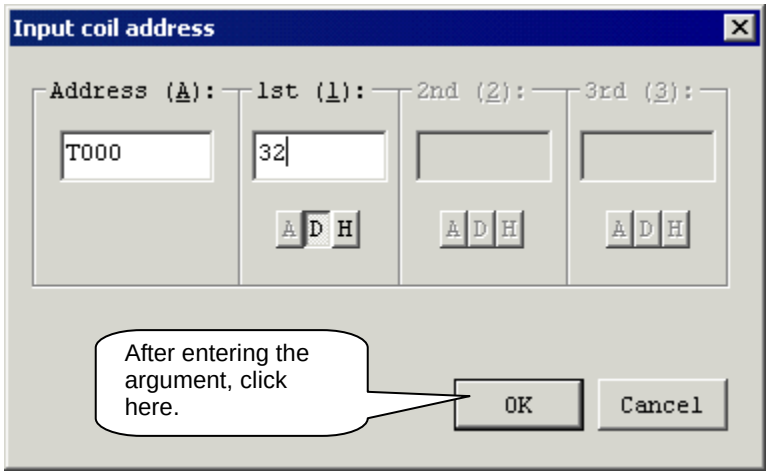
③ Enter “T000” in the coil address.



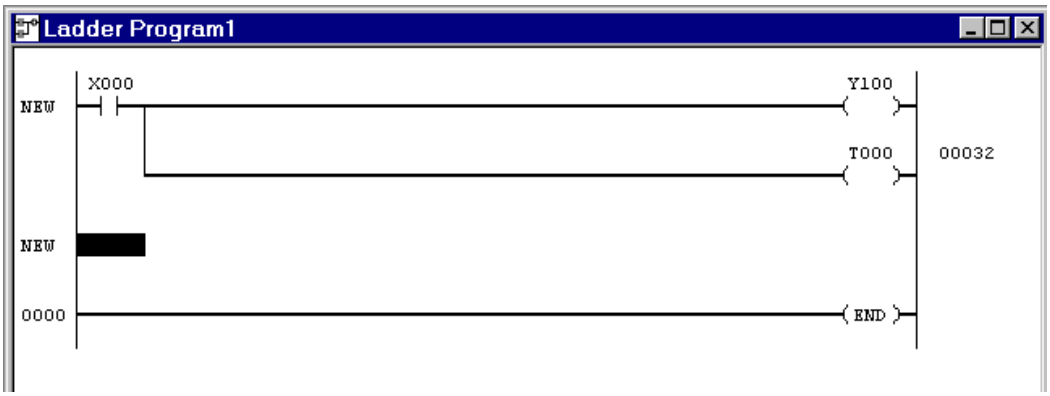
④ Click the first argument edit box to move the cursor.



⑤ Enter “32” in the first argument and click the [OK] button.

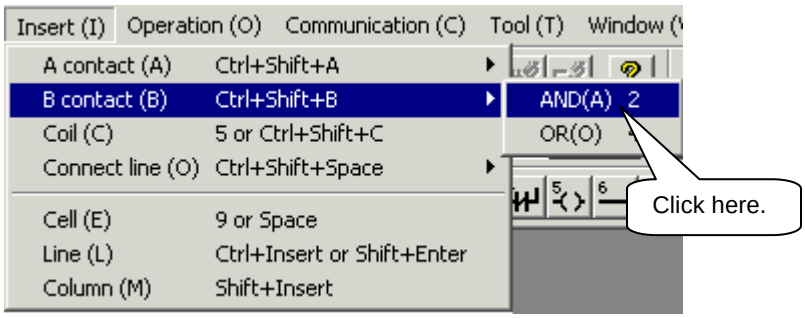


⑥ The following circuit is created.

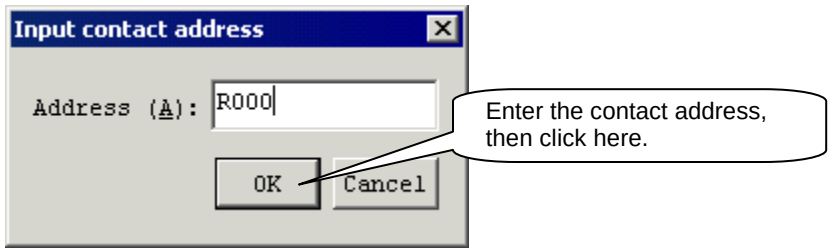


8.5.5 Input of Function Command in Coil

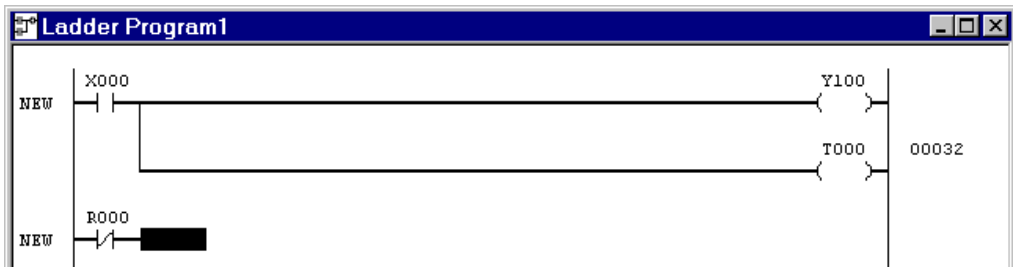
① Point [Normal Close Contact] from the [Insert] menu and click [AND].



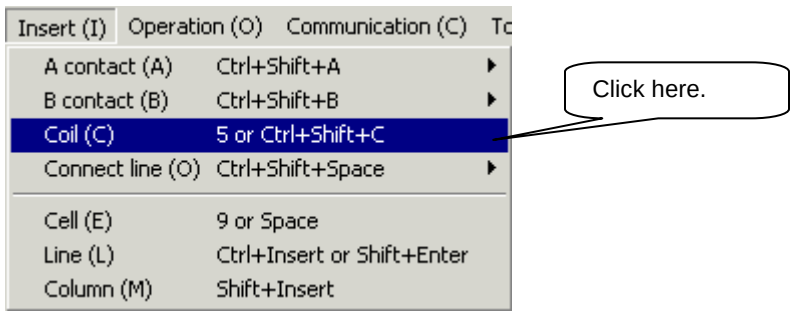
② The symbol of normal close contact is entered and the contact address input dialog box appears. Enter the contact address “R000”.



③ The normal close contact of R000 is entered.

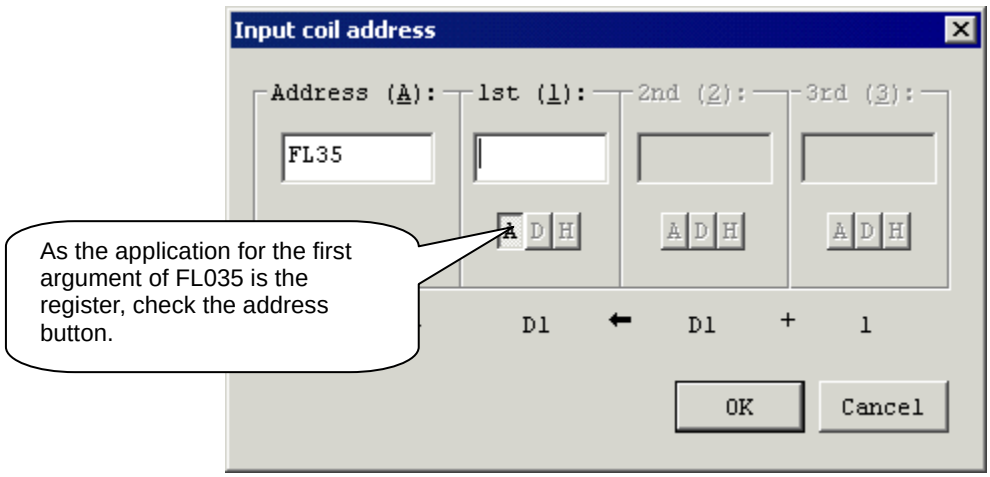


④ Click [Coil] from the [Insert] menu and the coil address input dialog box appears.

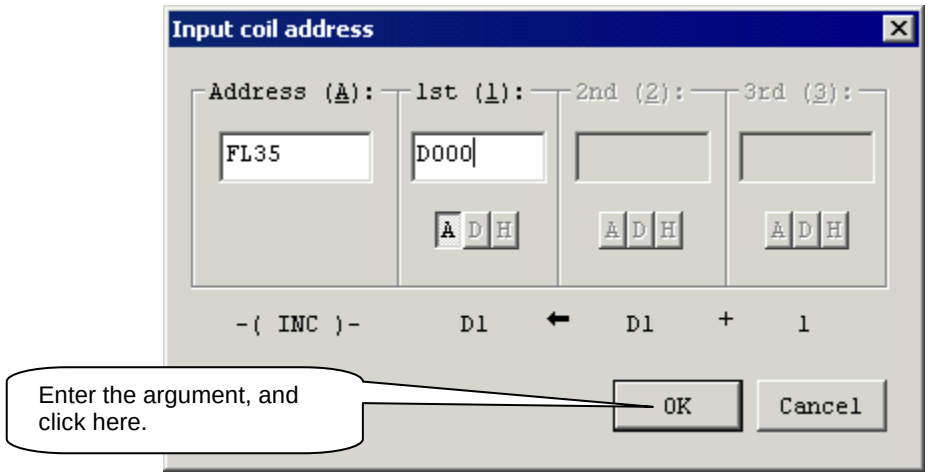


⑤ The symbol of coil is entered and the coil address input dialog box appears.

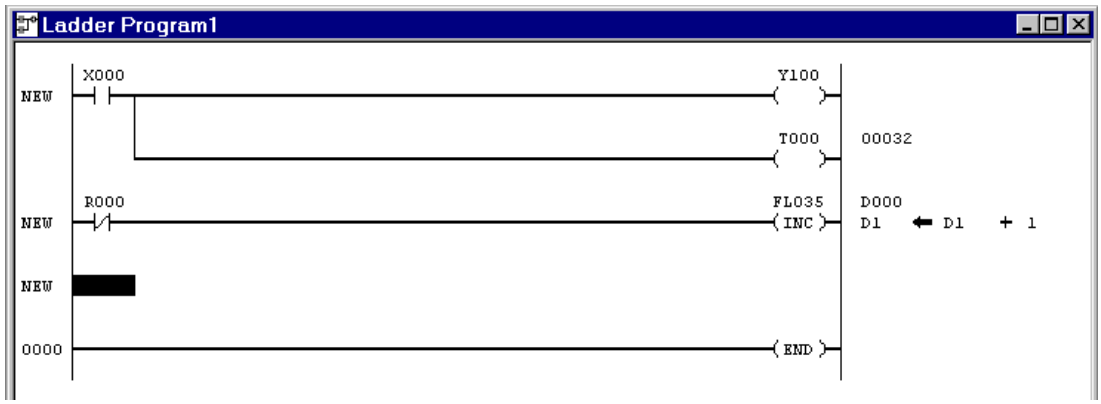
⑥ Enter “FL035” in the coil address, and move the cursor to the first argument edit box.



⑦ Enter “D000” in the first argument.

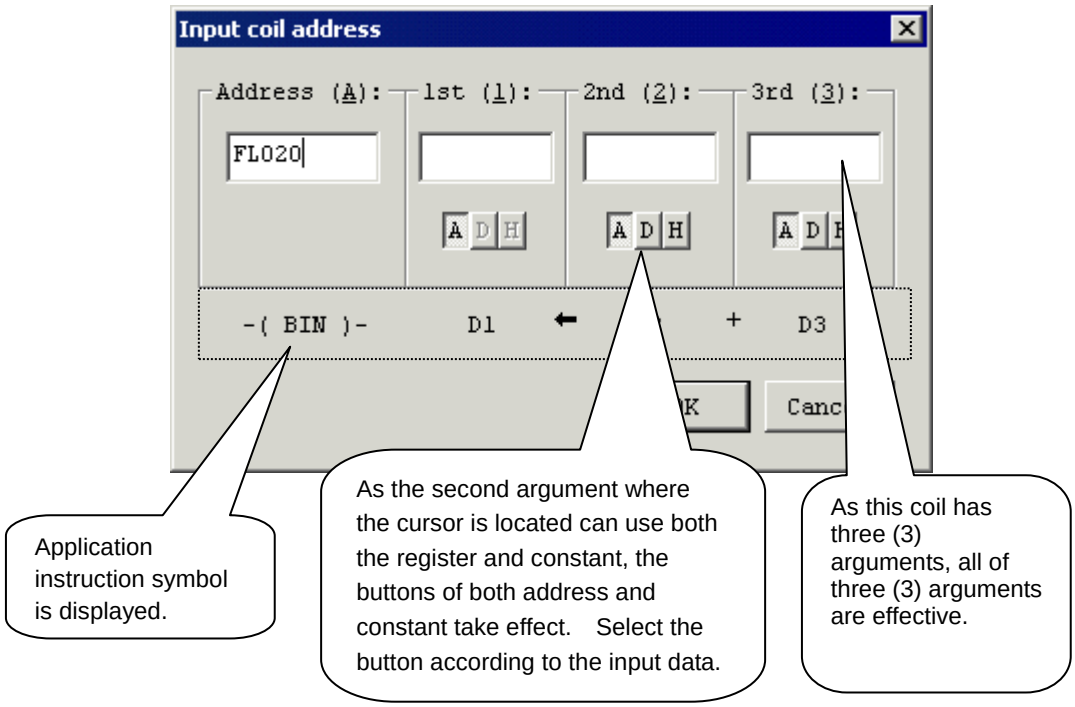


⑧ The following circuit is created.



One Point Advice

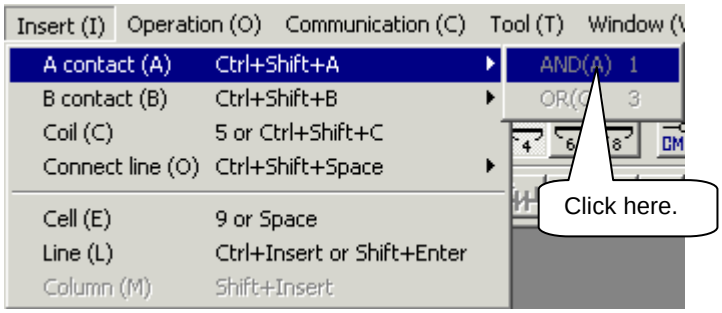
⊙ Coil input dialog box
After “FL020” is entered, the coil input dialog box is as follows:



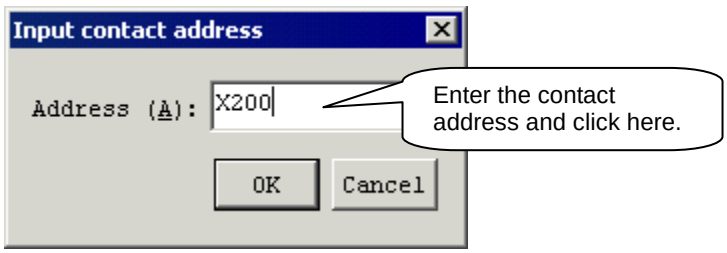
In this way, the status of the coil input dialog box differs with the type of the coil.

8.5.6 Input of Pair Coil

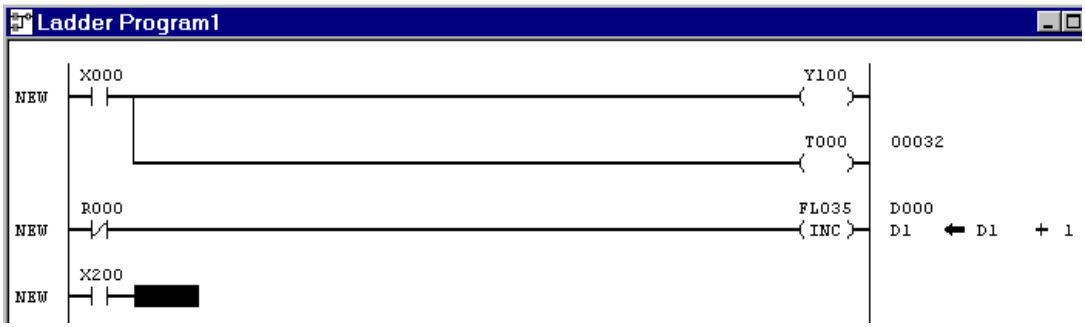
① Point [Normal Open Contact] from the [Insert] menu and click [AND].



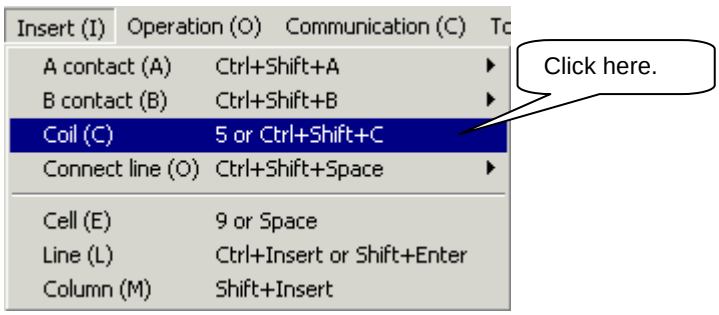
② The symbol of normal open contact is entered and the contact address input dialog box appears. Enter “X200” in the contact address.



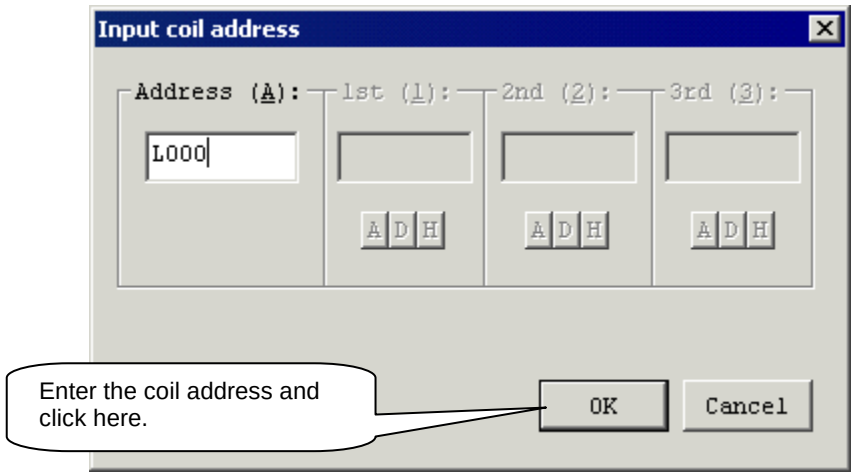
③ The normal open contact of X200 is entered.



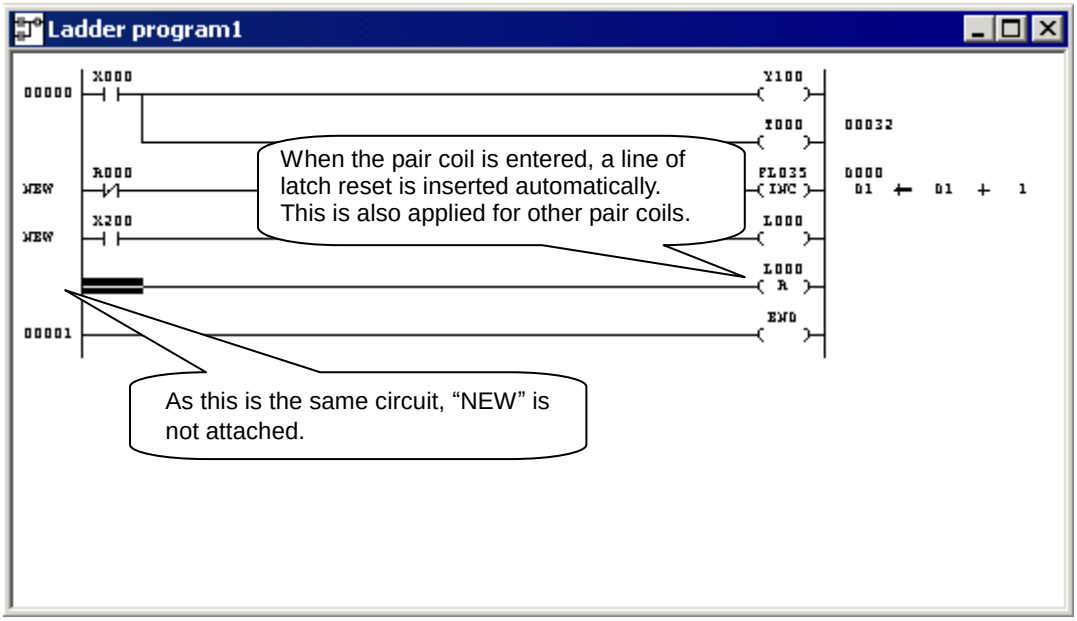
④ Click [Coil] from the [Insert] menu.



⑤ The symbol of coil is entered and the coil address input dialog box appears. Enter “L000” in the coil address.

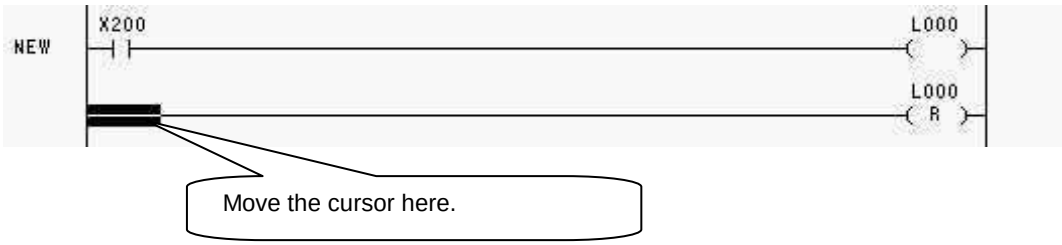


⑥ The following circuit is created.

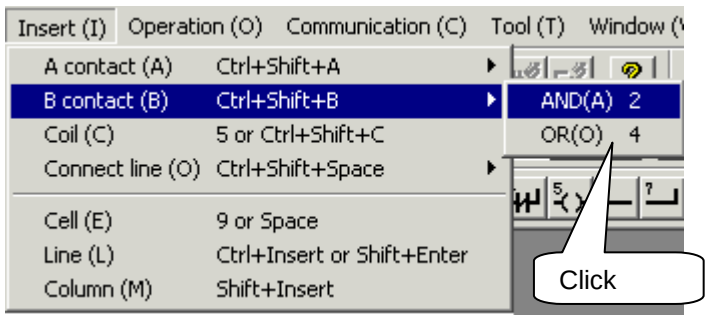


8.5.7 Change of Connecting Cine with Contact

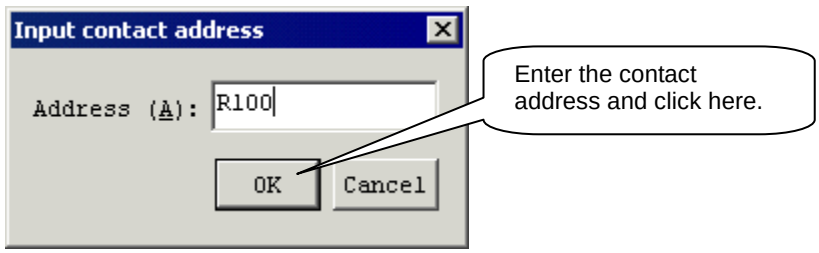
① Move the mouse to the head of the latch reset and click. The cursor moves to the head position.



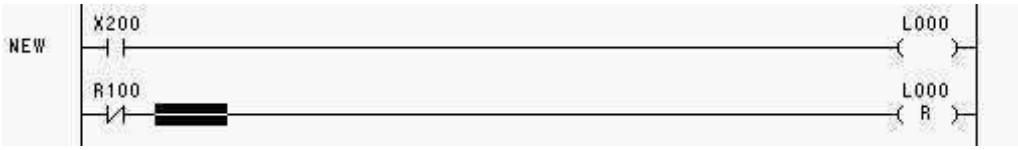
② Point [Normal Close Contact] from the [Insert] menu and click [AND].



③ The symbol of normal close contact is entered and the contact address input dialog box appears. Enter “R100” in the contact address.

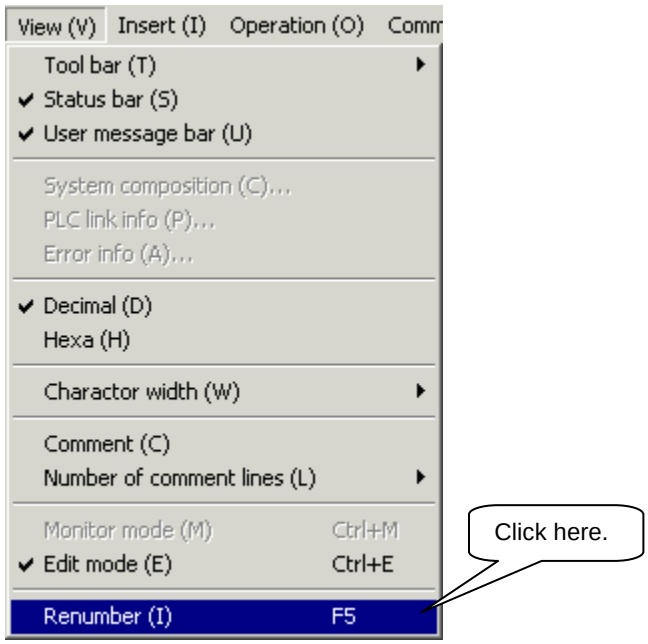


④ The normal close contact is inserted at a place where the connecting line was input and the circuit is changed as shown below.

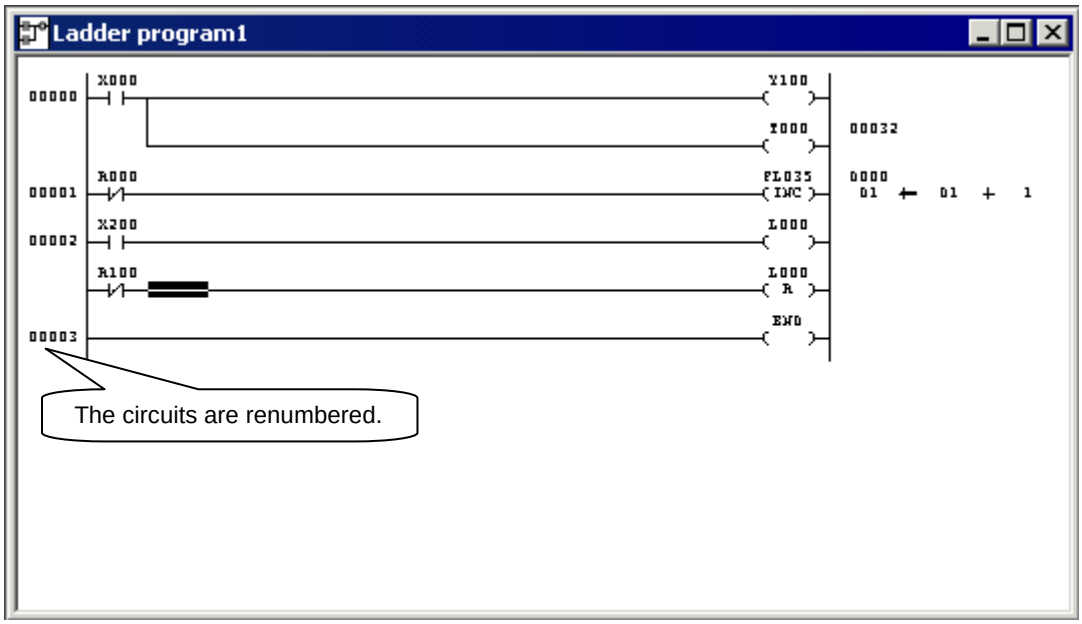


8.5.8 Renumber of Circuits

① Click [Renumber] from the [View] menu.

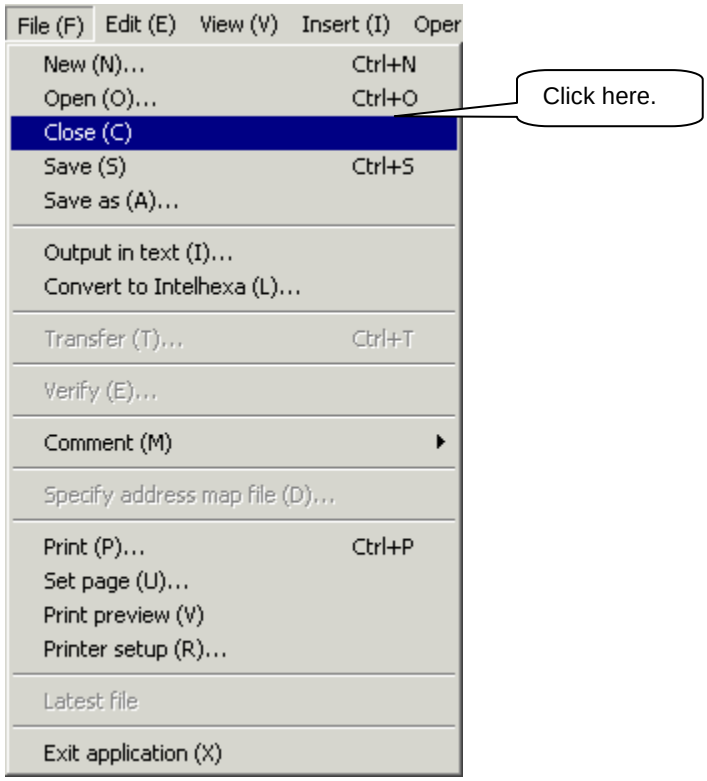


② Words [NEW] are cleared and the circuits are renumbered.

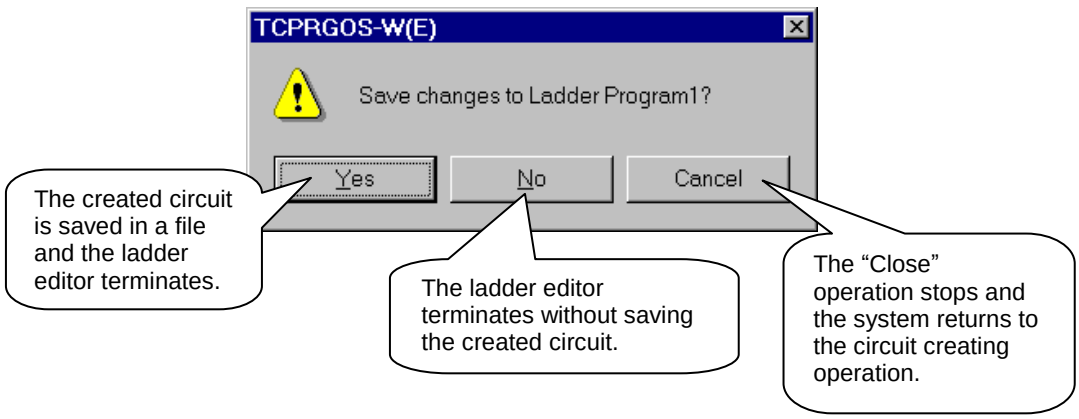


8.5.9 Saving in File

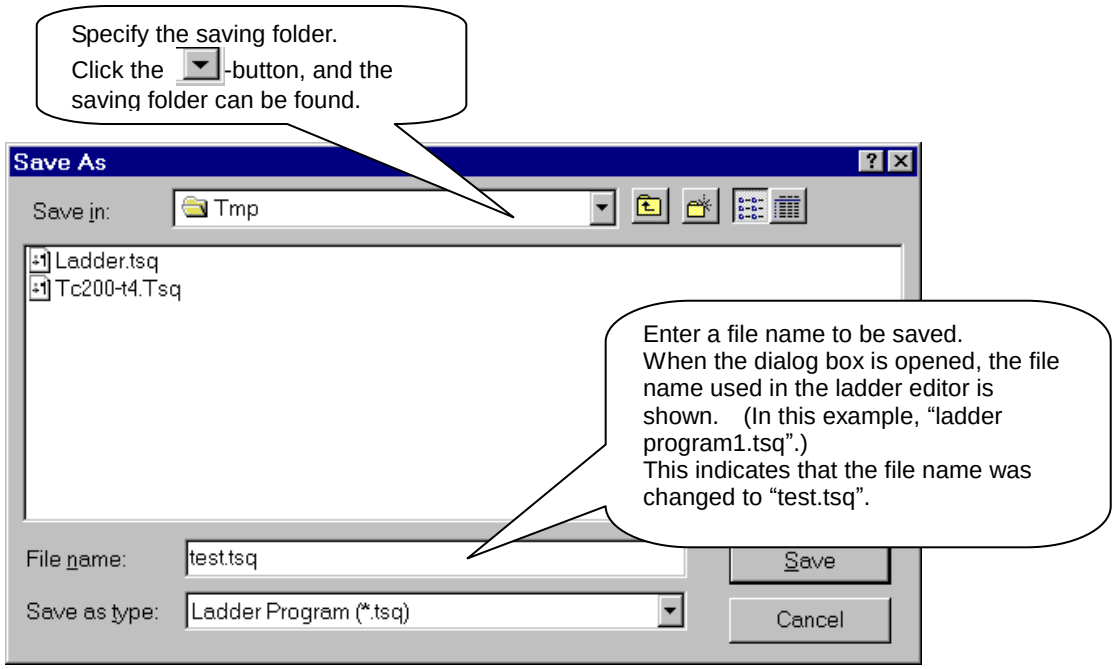
① Click [Close] from the [File] menu.



② As the following message box appears, click the [YES] button.



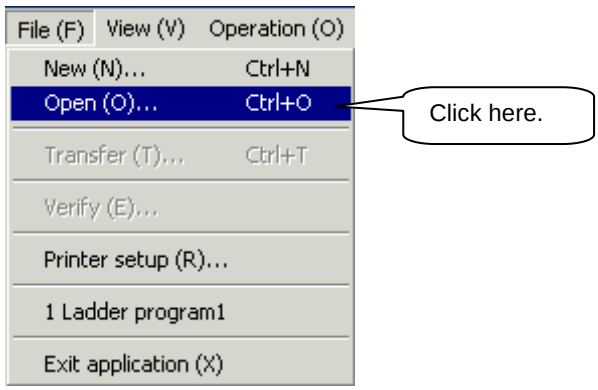
③ The following file saving dialog box appears.



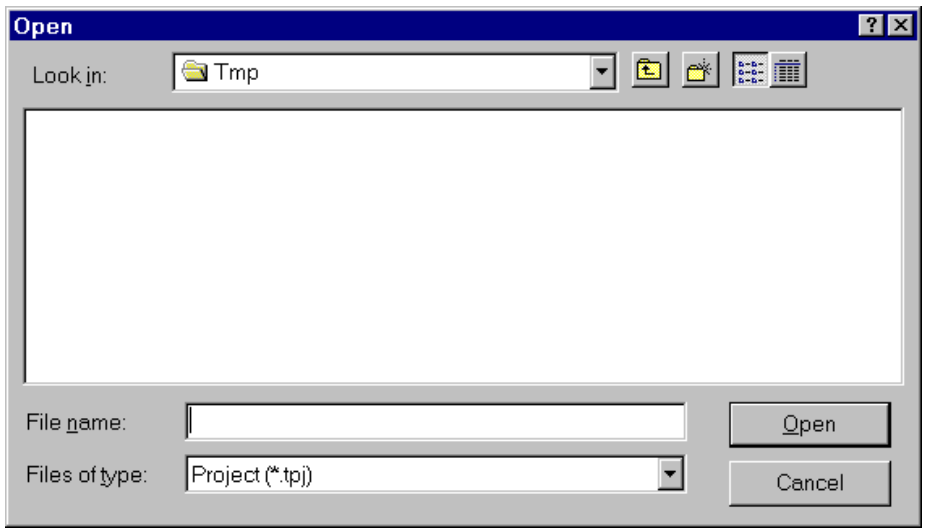
④ Click the [Save] button. The ladder editor is closed and the created circuit is saved in the file of “test.tsq”.

8.5.10 Opening of Existing File

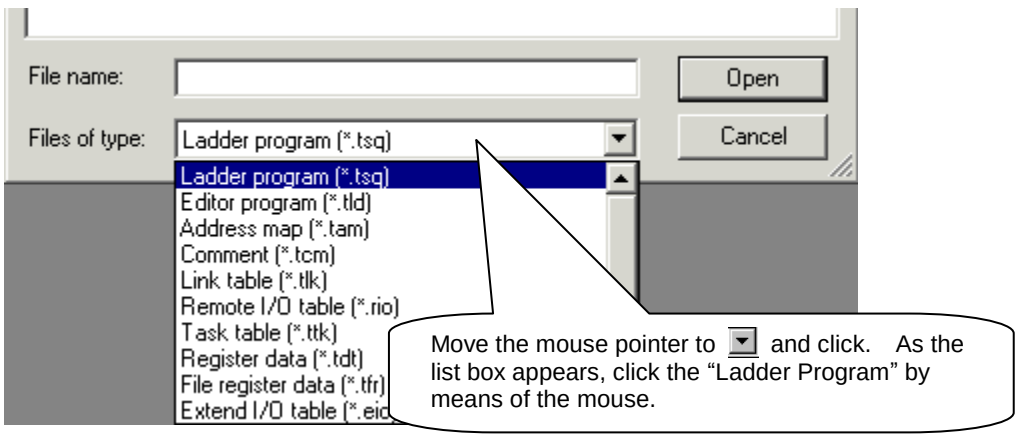
① Click [Open] from the [File] menu.



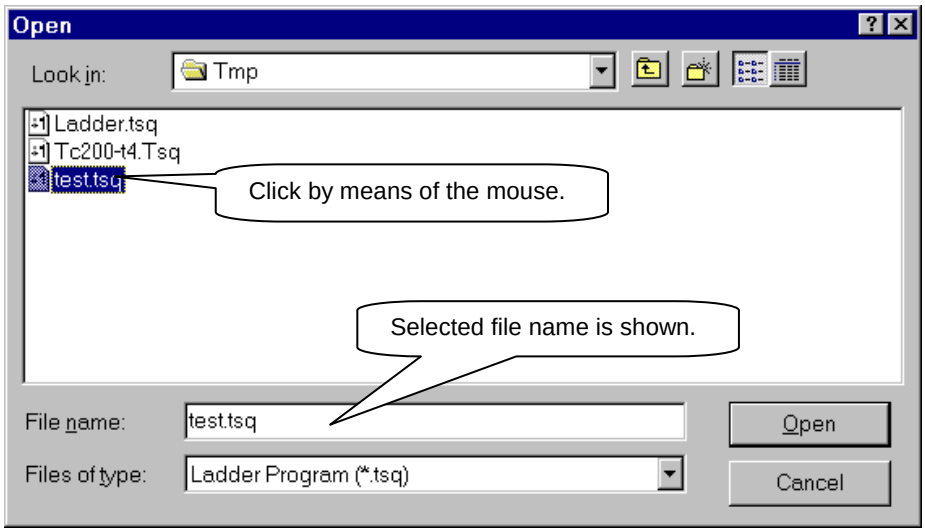
② The following [Open] dialog box appears.



③ Change the type of file to “Ladder Program”.



④ Select “test.tsq” among the displayed ladder programs.

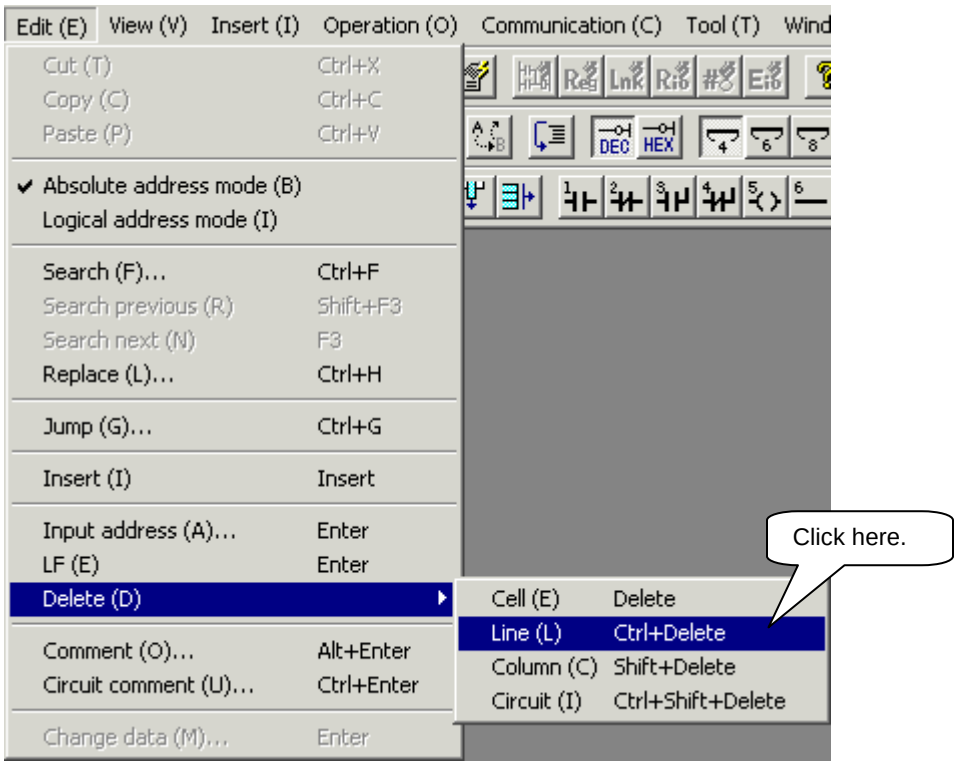


⑤ Click the [Open] button, and the file of “test.tsq” is opened.

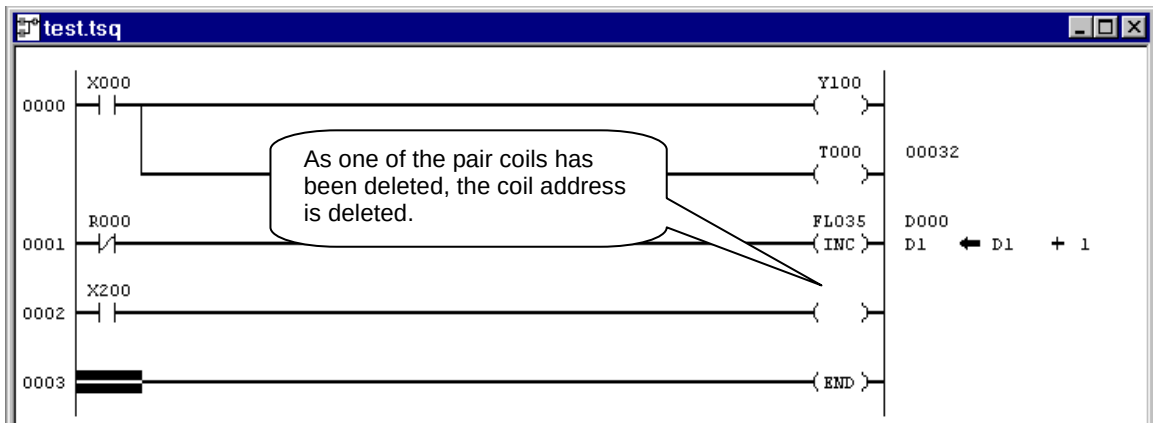
8.5.11 Modifying and Saving of Existing Circuit

① Move the cursor to the circuit of the latch reset.

② Point [Delete] from the [Edit] menu and click [Line].

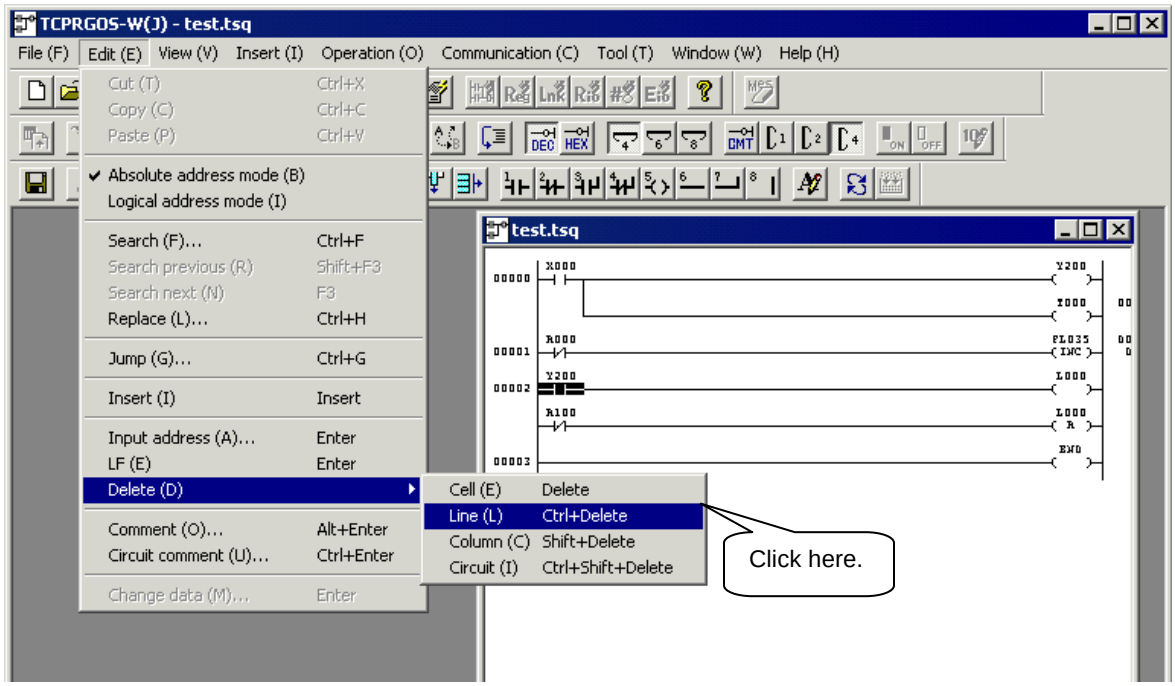


③ The specified line is deleted.

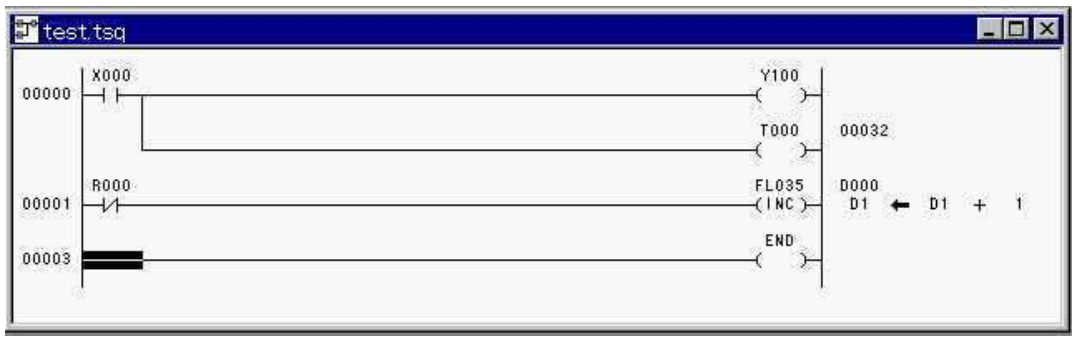


④ Move the mouse pointer to circuit number 0002 and click (circuit selection).

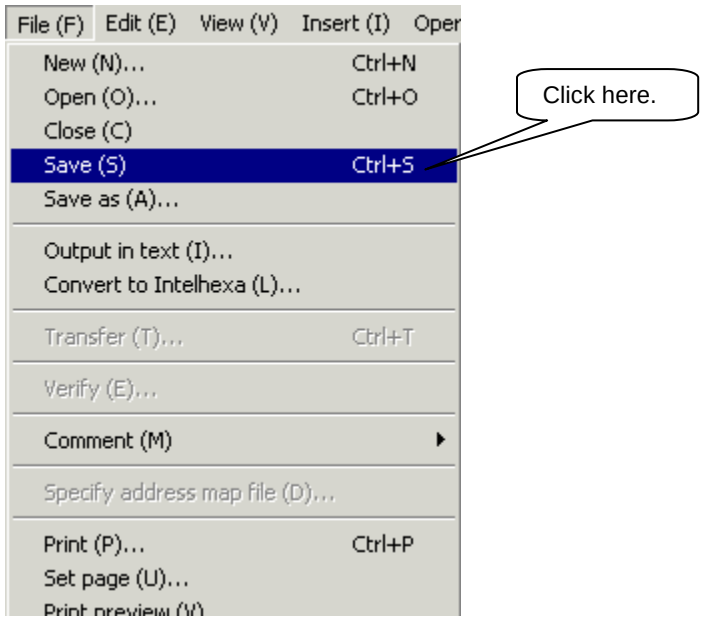
⑤ Point [Delete] from the [Edit] menu and click [Circuit].



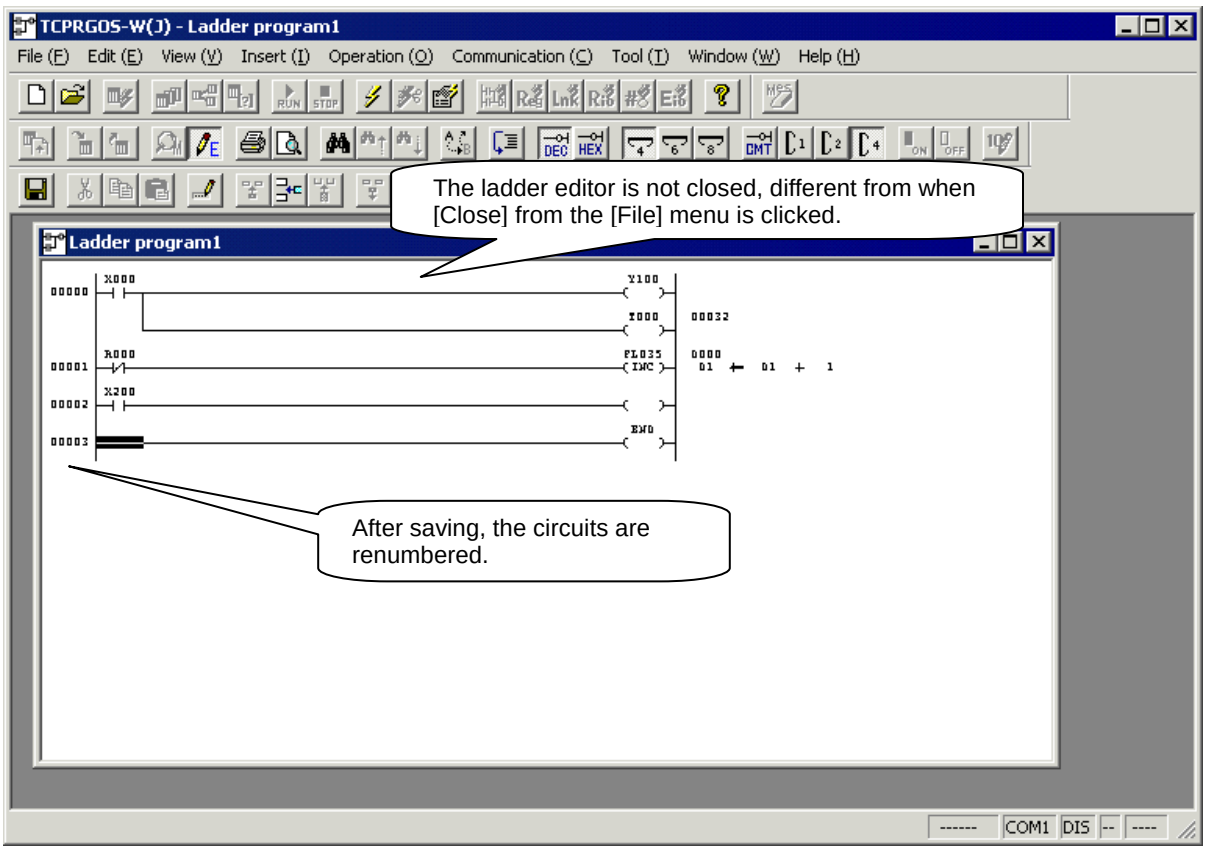
⑥ The circuit of circuit number 0002 is deleted.



⑦ Click [Save] from the [File] menu.



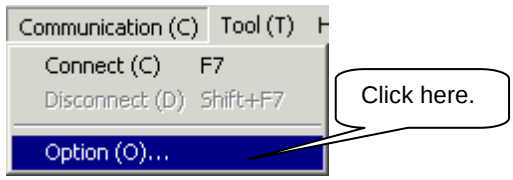
⑧ The file of “test.tsq” is overwritten.



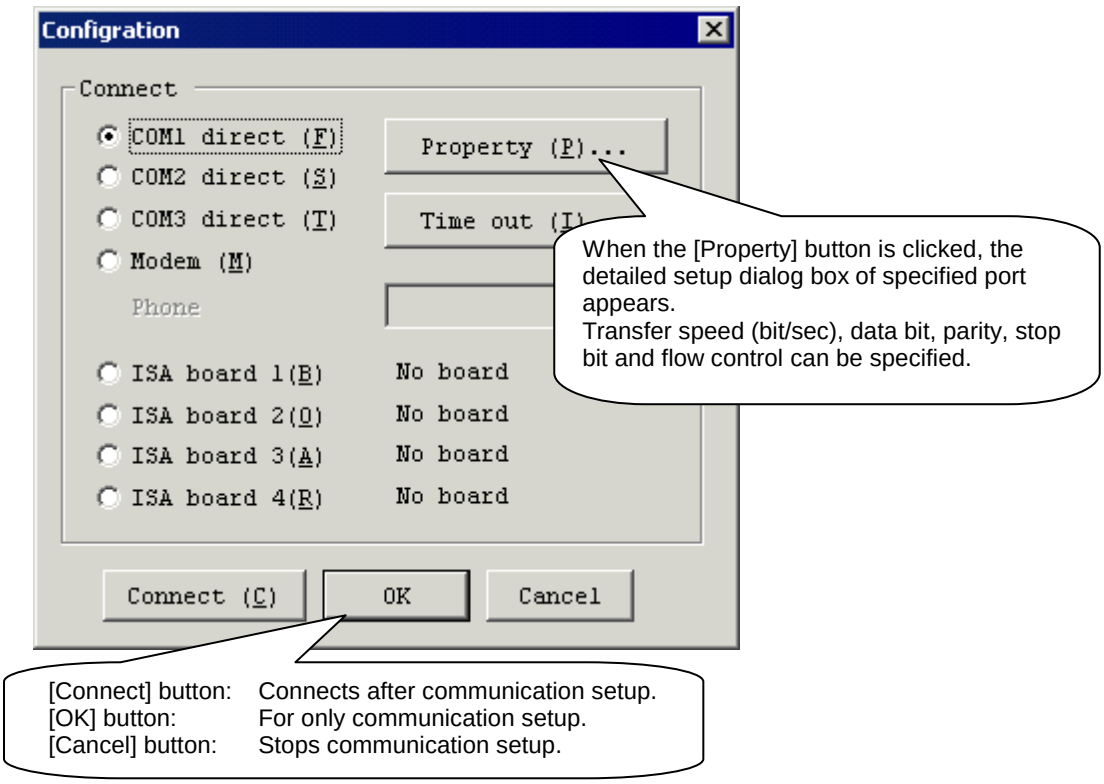
8.6 Transferring Circuit

8.6.1 Check for Communication Setup

① Click [Option] from the [Communication] menu.



② The following communication setup dialog box appears.



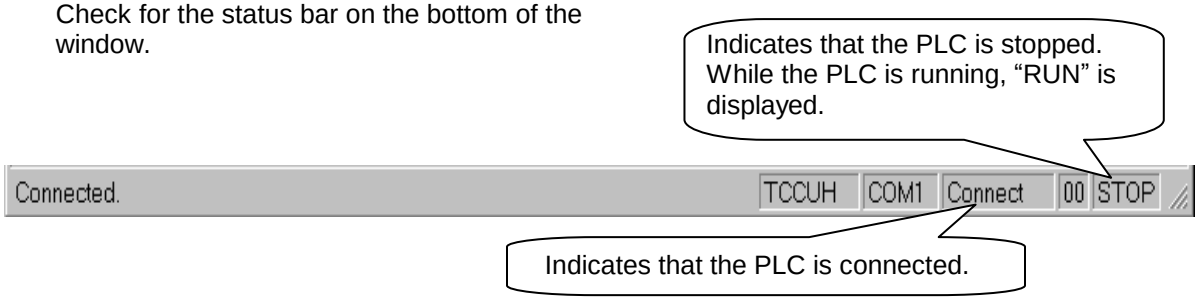
Caution !

The baud rate is changed over automatically.
Select the connection method alone. Normally, the property need not be changed.

8.6.2 Connection

① Click [Connect] from the [Communication] menu, and the PLC is connected with the personal computer.

Check for the status bar on the bottom of the window.

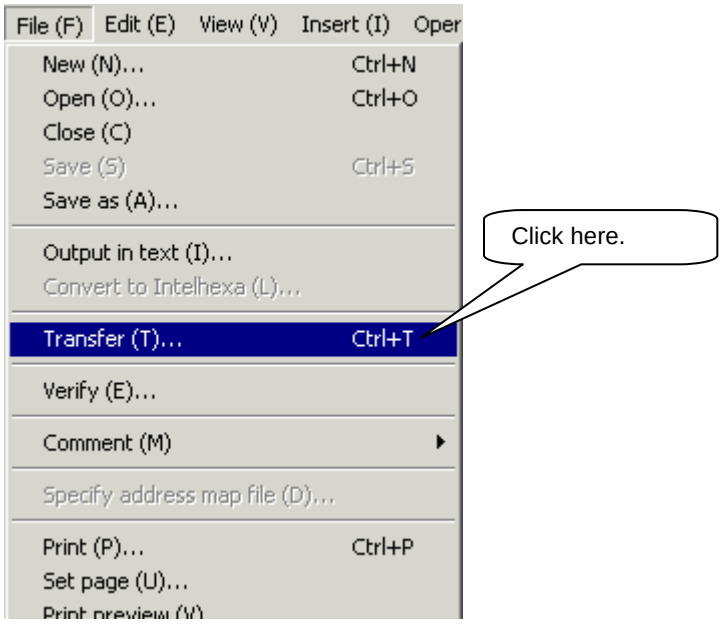


Caution !

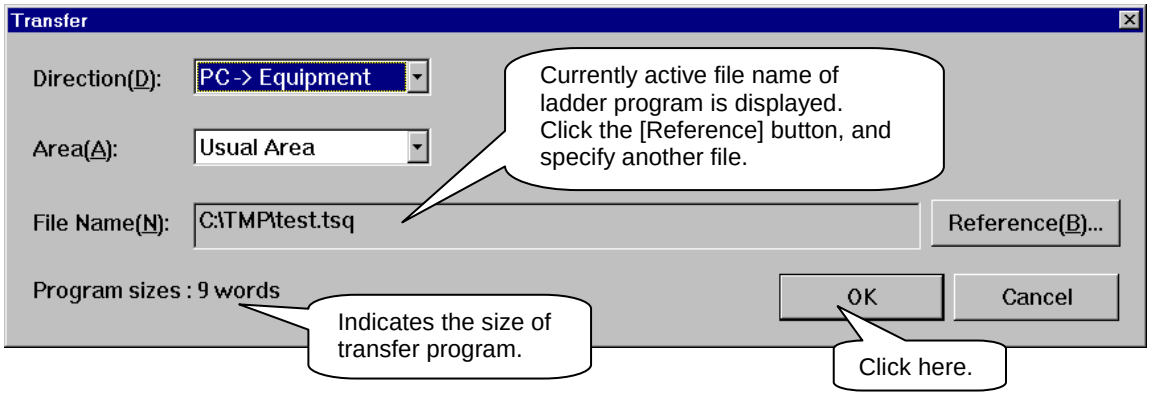
Baud rate that can be connected is recognized automatically, and the connected baud rate is displayed on the status bar.

8.6.3 Transfer

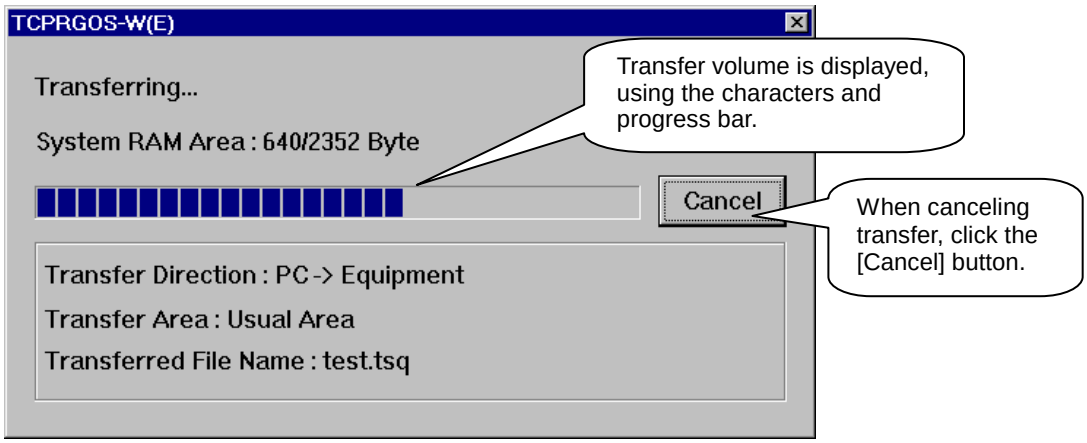
① Click [Transfer] from the [File] menu.



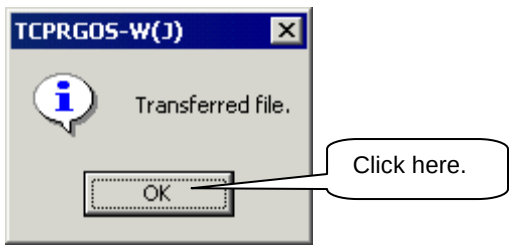
② When the following transfer dialog box appears, click the [OK] button.



③ Transfer starts. During the transfer, the following transfer progress dialog box appears.

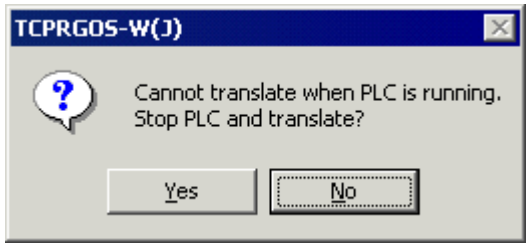


④ When the transfer has finished, the transfer finish message box appears. Click the [OK] button then.



Caution !

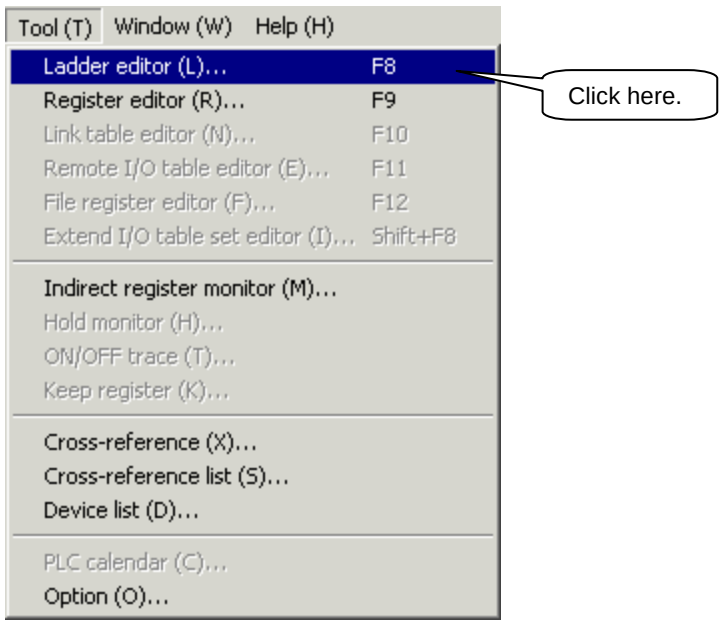
While the TCmini is running, the following message box appears. Select [No] because data change is not possible while the TCmini is running.



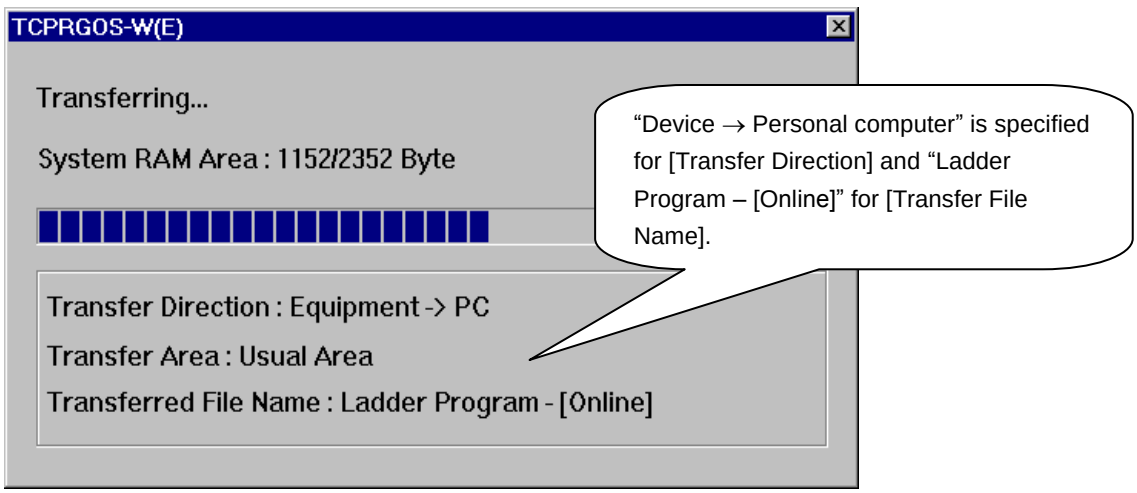
8.7 Online Operation

8.7.1 Start Online Ladder Editor

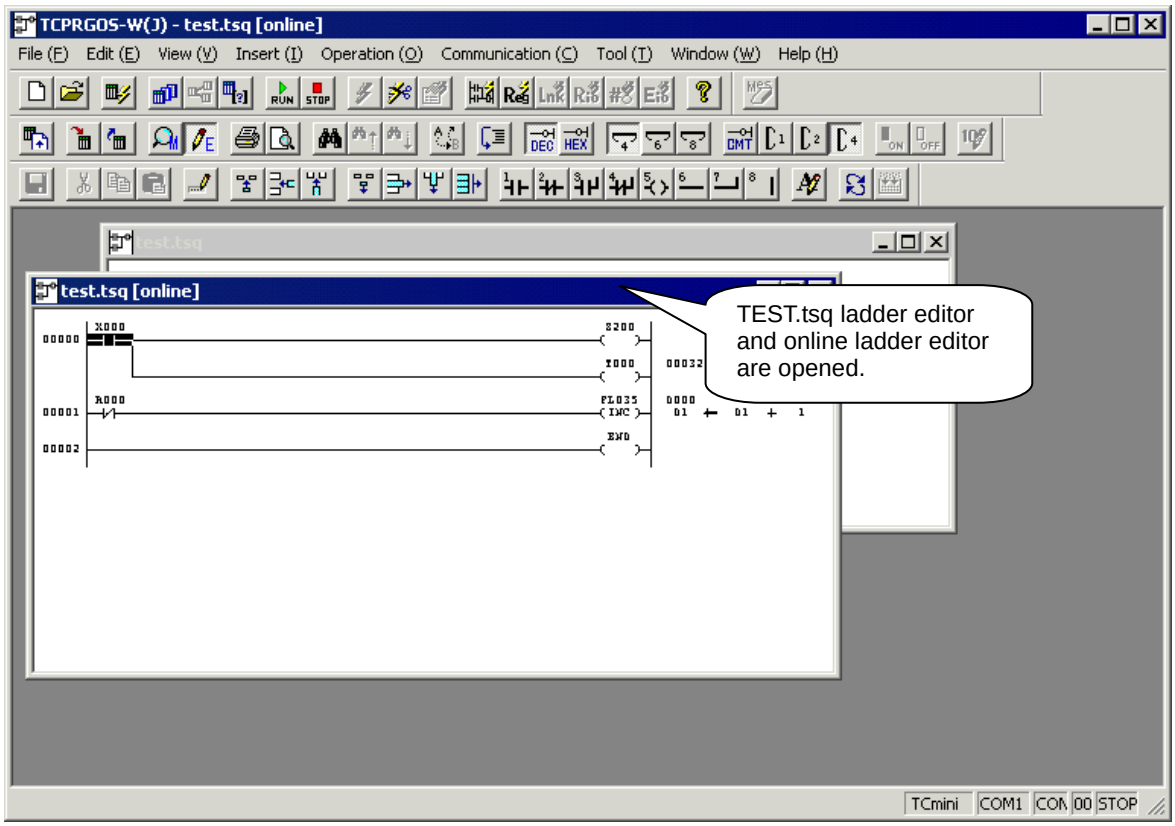
① Click [Ladder Editor] from the [Tool] menu.



② Data transfer from the PLC to the personal computer starts and transfer progress dialog box is displayed.



③ When the transfer has finished, the transfer progress dialog box disappears and the online ladder editor starts.



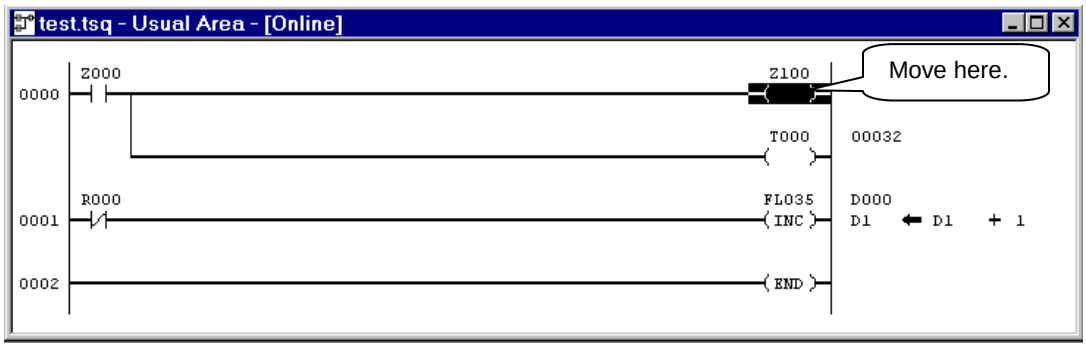
Caution !

In an offline state or other situation where it cannot be determined whether the result is X000 or Y000, it appears as Z000. Then, when an online connection is established and the PLC information is obtained, X or Y is set. (However, there is no problem with leaving this as Z.)

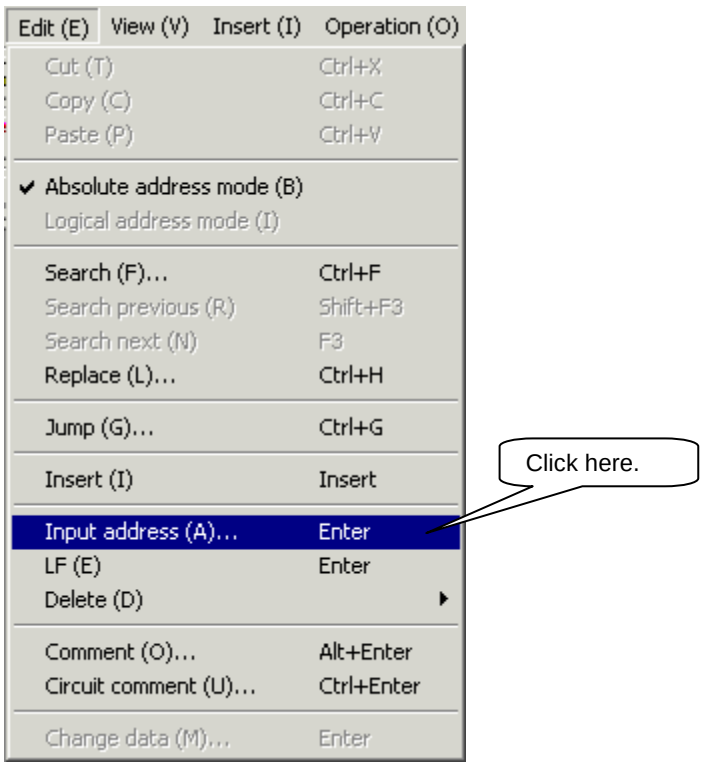
8.7.2 Change of Online Ladder Editor Circuit

★ Modification can be made in the same manner as the offline ladder editor. To change the coil address, observe the following steps.

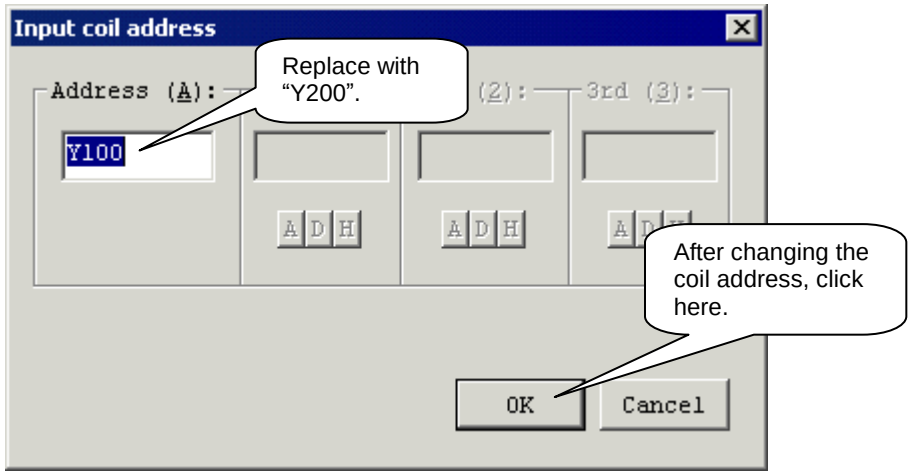
① Move the cursor to the coil (Z100) of circuit number 0000.



② Click [Input Address] from the [Edit] menu.

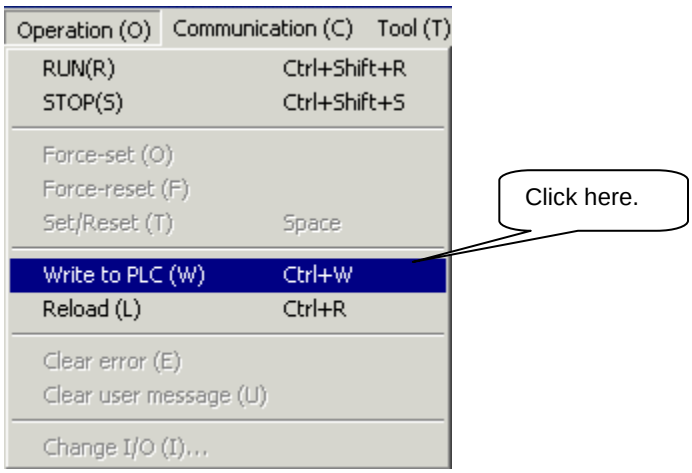


③ As the coil address input dialog box appears, change the coil address to “Y200”.

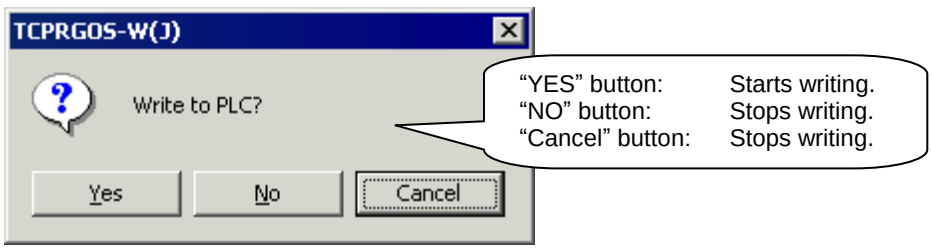


④ The coil address is changed.

⑤ Click [Write to PLC] from the [Operation] menu.

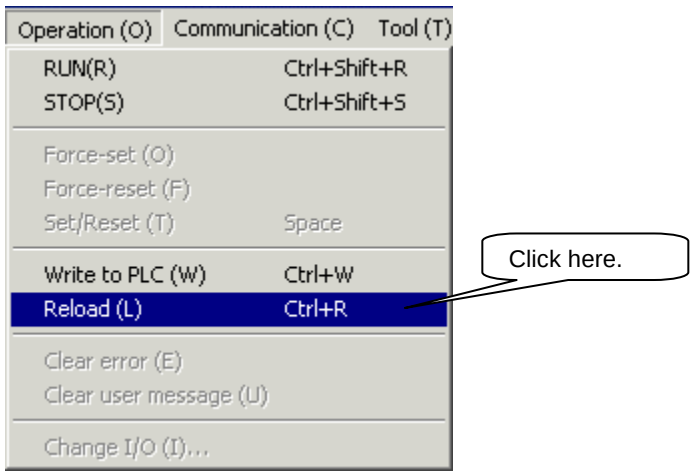


⑥ As the following message box appears, click the [YES] button.

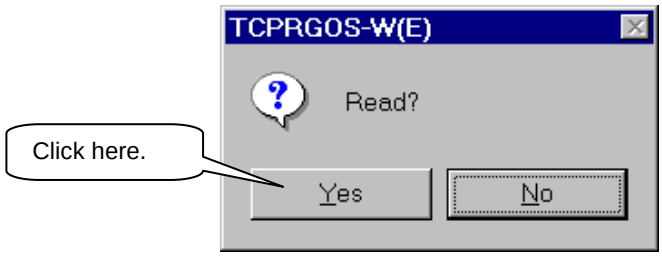


⑦ The transfer progress dialog box is displayed and writing to PLC starts. When it has finished, the transfer progress dialog box is cleared.

⑧ Click [Reload] from the [Operation] menu.



⑨ As the following message dialog box appears, click the [YES] button.



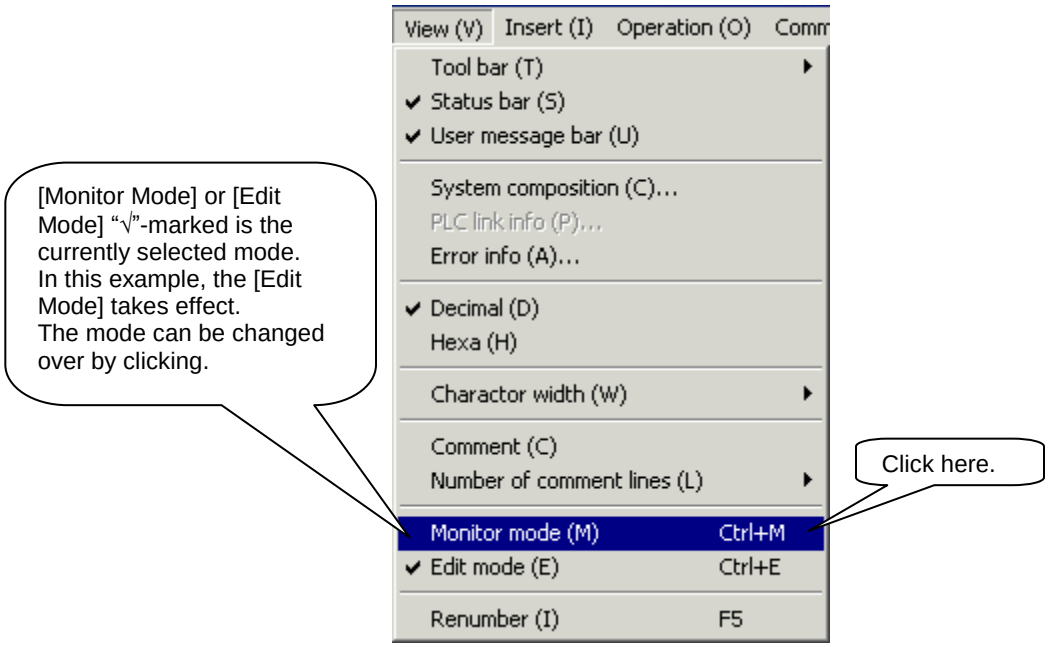
⑩ The online ladder editor is closed and the transfer progress dialog box appears. Then the transfer starts.

⑪ When the transfer has finished, the transfer progress dialog box is closed and the online ladder editor starts again.

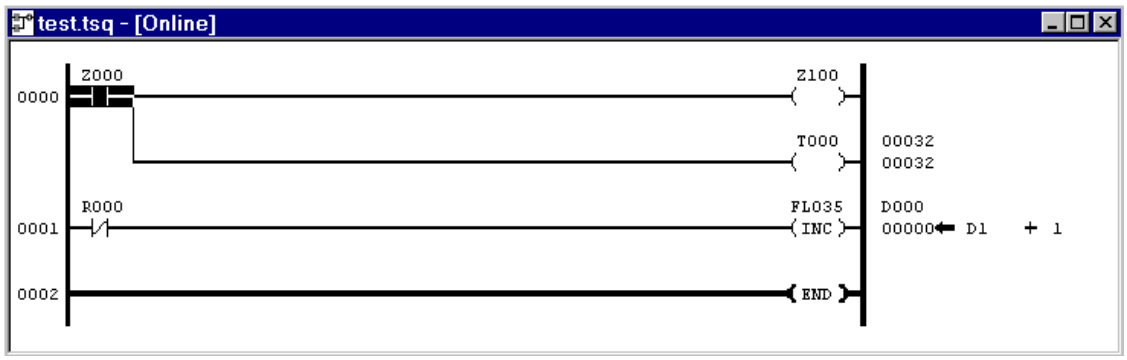
8.8 Monitor

8.8.1 Selection of Monitor Mode

① Click [Monitor Mode] from the [View] menu.

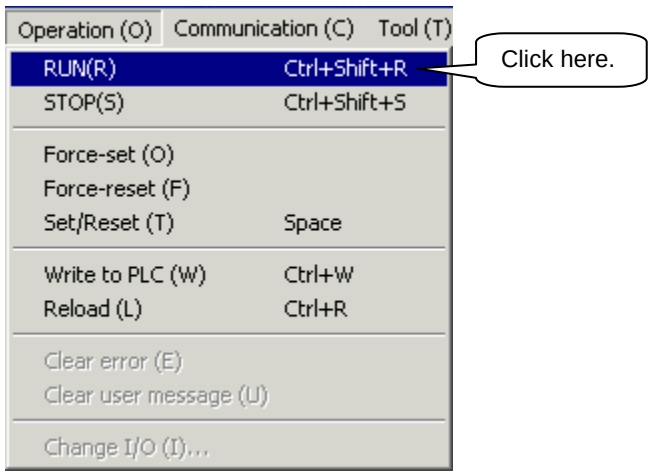


② The monitor mode replaces the edit mode.

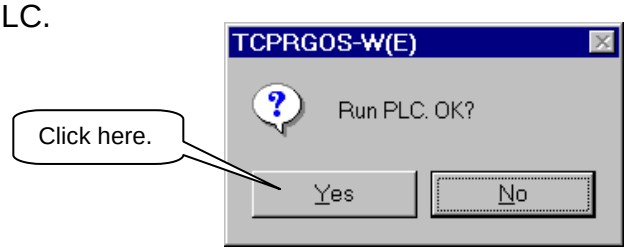


8.8.2 Designation of PLC RUN or STOP and Compulsive Setting of Contact

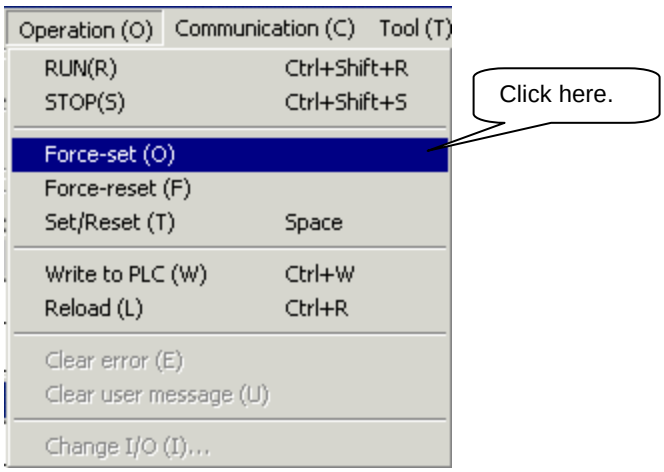
① Click [RUN] from the [Operation] menu.



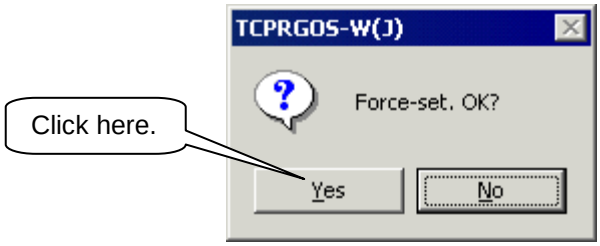
② As the following message box appears, click the [YES] button to run the PLC.



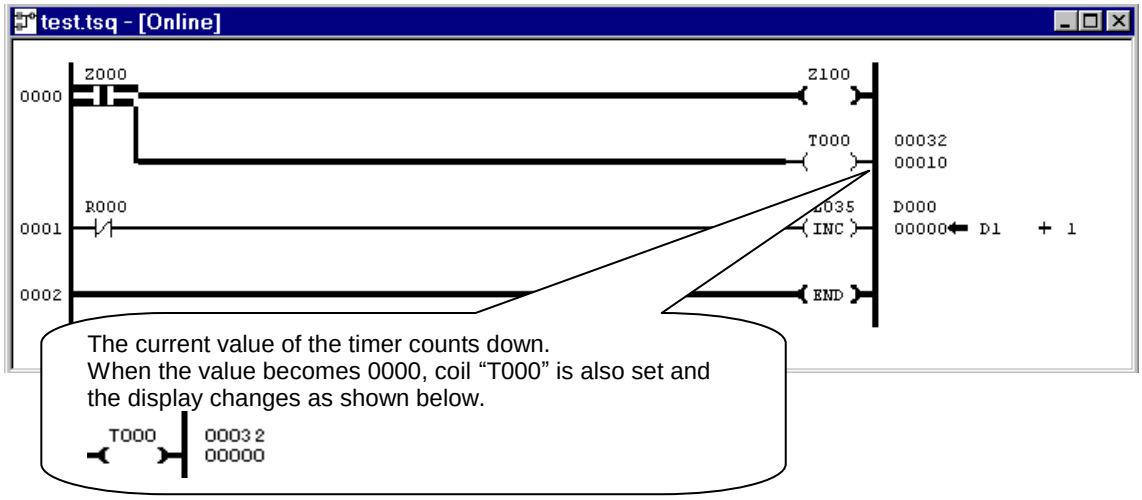
③ Make sure that the cursor is located at the normal open contact of “Z000” in circuit number 0000. Then click [Force-set] from the [Operation] menu.



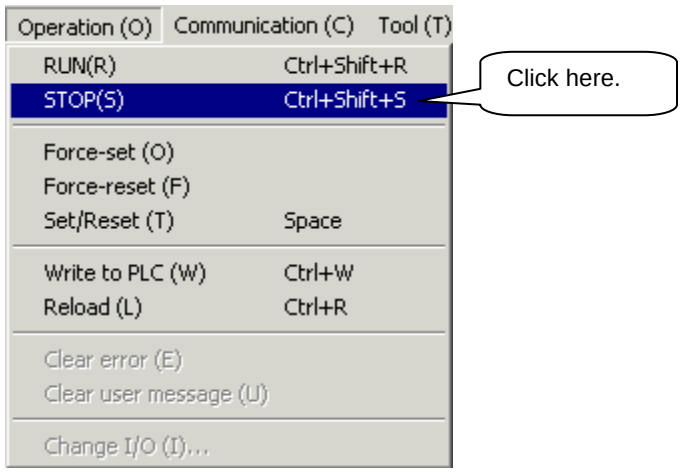
④ As the following message box appears, click the [YES] button.



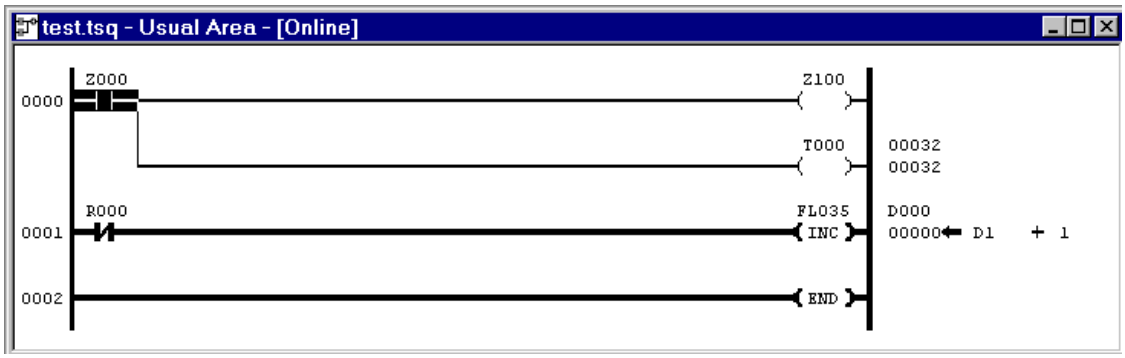
⑤ The normal open contact of “Z000” in circuit number 0000 is compulsively set.



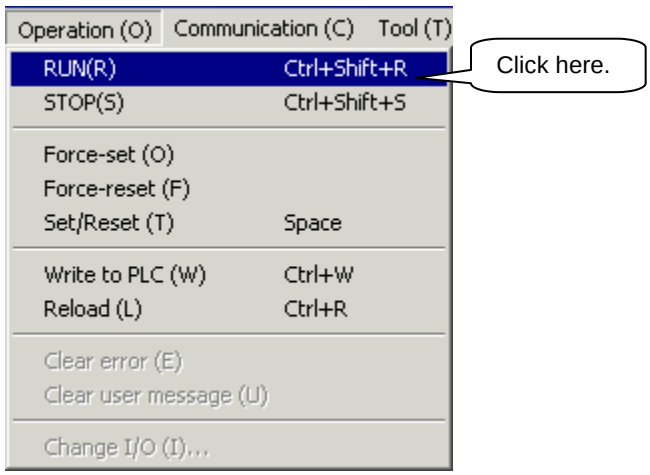
⑥ Click [STOP] from the [Operation] menu to stop the PLC.



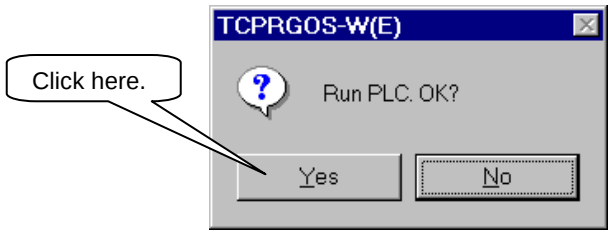
⑦ The normal close contact is set and monitored as shown below.



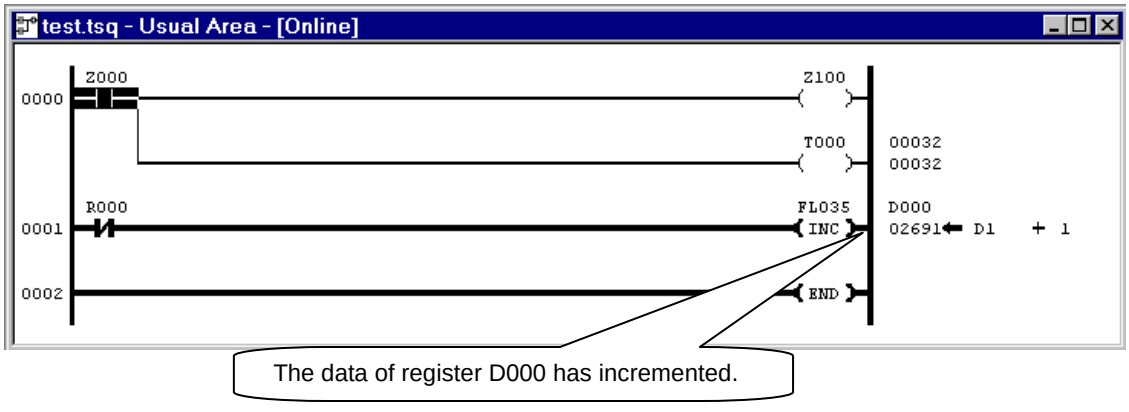
⑧ Click [RUN] from the [Operation] menu.



⑨ As the following message box appears, click the [YES] button to run the PLC.

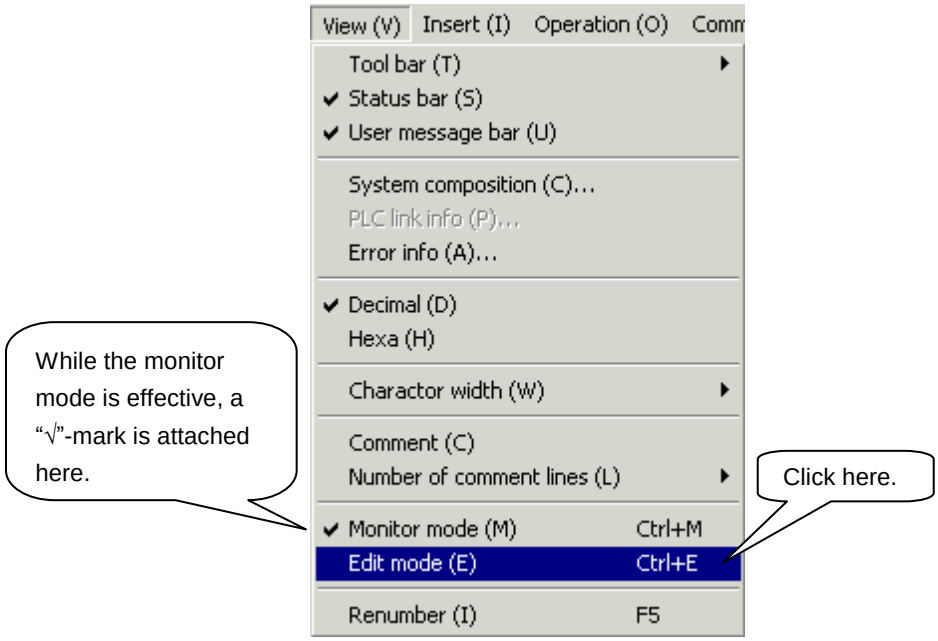


⑩ Monitoring is performed as shown below.



8.8.3 Quit of Monitor Mode

① Click [Edit Mode] from the [View], and the monitor mode terminates with the edit mode selected.



Caution !

In the monitor mode, editing of a circuit (modification, addition, deletion, etc.) is not allowed. Edit after selecting the edit mode. A modified circuit cannot be monitored unless written to the PLC. After writing it to the PLC, read and call the latest online ladder editor.

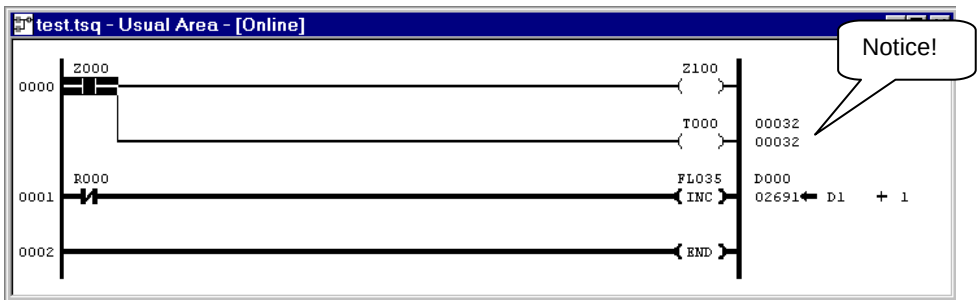
One Point Advice

⊙ Changeover of decimal/hexadecimal notation

The notation comes in two (2) types; decimal notation and hexadecimal notation.

To change over the notation, click [Decimal] or [Hexadecimal] from the [View] menu. The menu item marked with "√" on the left side is the currently selected notation.

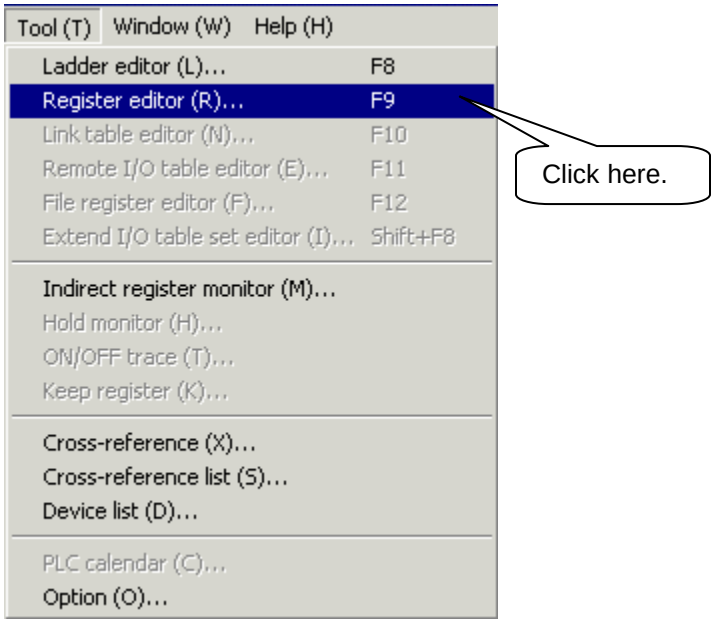
Example of hexadecimal notation:



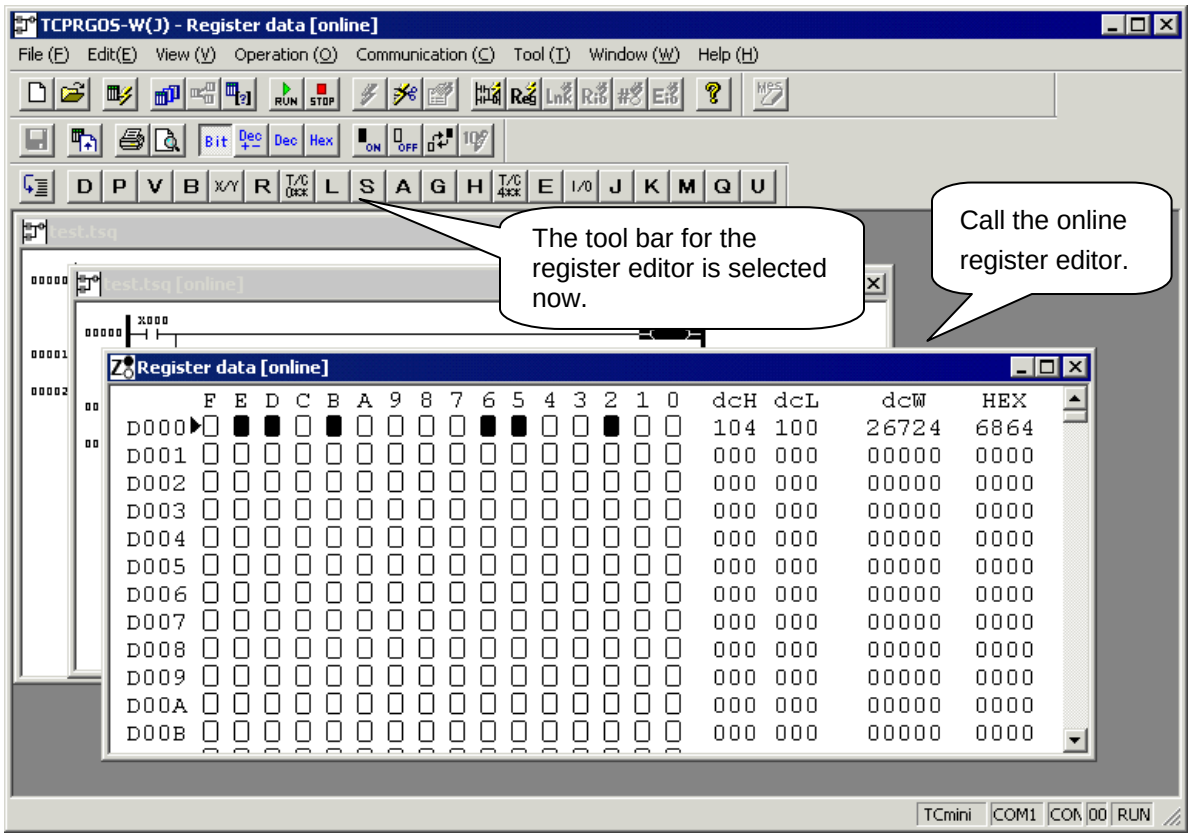
8.9 Monitoring Register Data

8.9.1 Start of Online Register Editor

① Click [Register Editor] from the [Tool] menu.

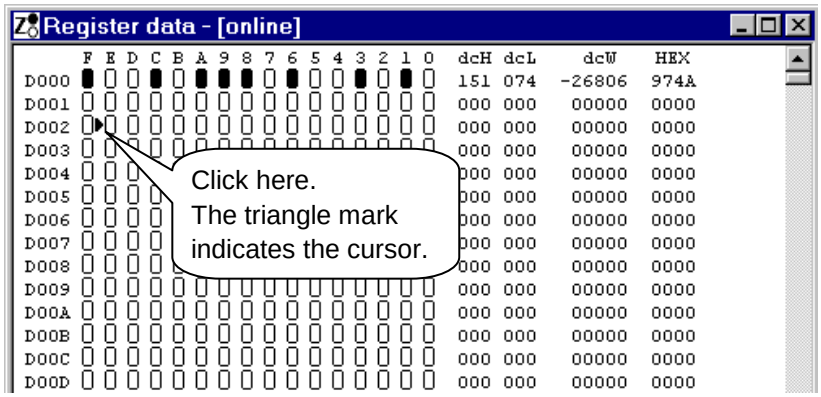


② The online register editor starts.

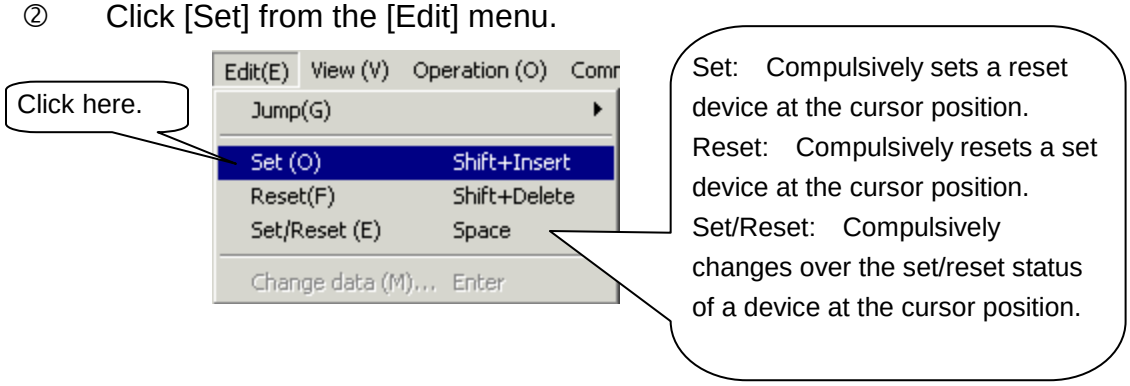


8.9.2 Compulsive Setting of Device

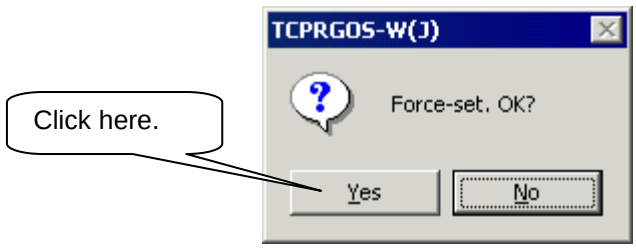
① Click Bit E of D002 by means of the mouse and move the cursor.



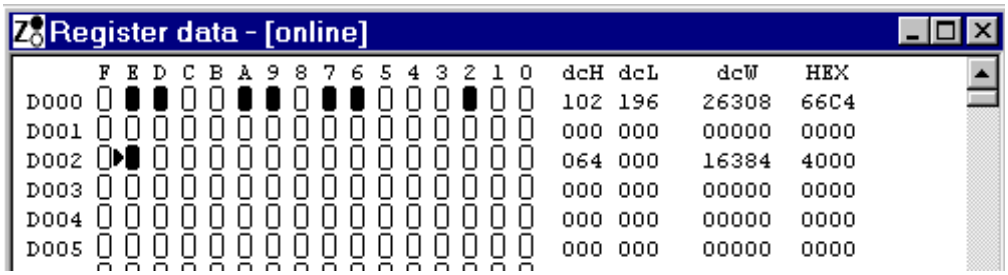
② Click [Set] from the [Edit] menu.



③ As the following message dialog appears, click the [OK] button.

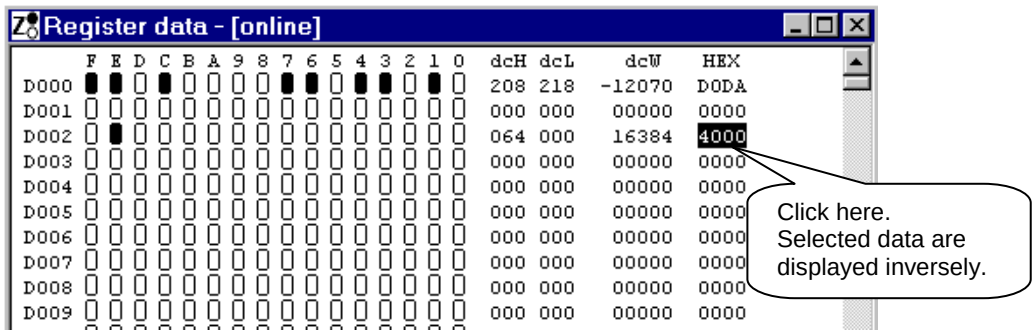


④ Bit E of D000 is set compulsively.

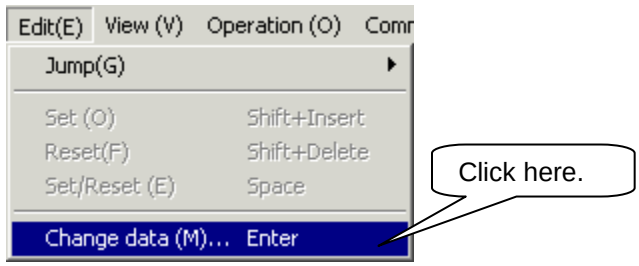


8.9.3 Modification of Word Data Specified by Address

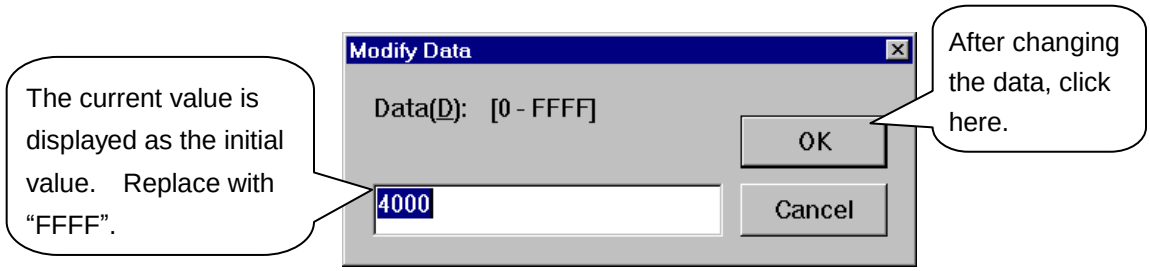
① Move the mouse pointer to the hexadecimal data area of D002, then click.



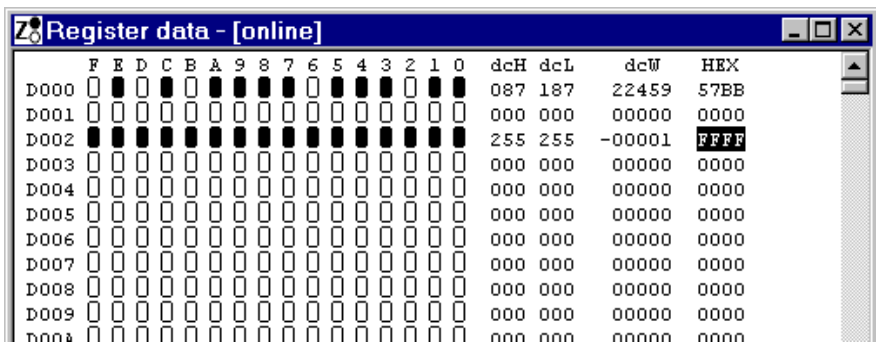
② Click [Change Data] from the [Edit] menu.



③ The following [Change Data] dialog box appears. Change the data to "FFFF" and click the [OK] button.



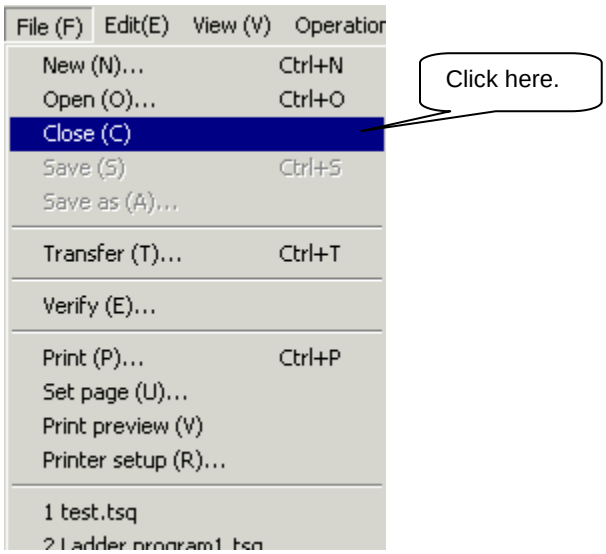
④ The data are changed.



8.10 Quitting the TCPRGOS

8.10.1 Exit from Active Editor

① Click [Close] from the [File] menu. The active editor terminates.

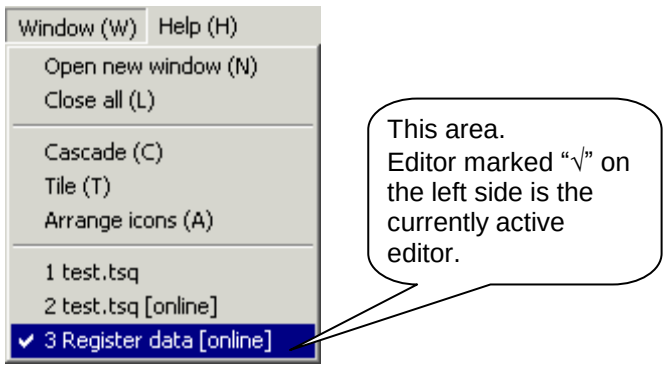


② Repeat Step ① above and quit all editors (online ladder editor, online register editor, offline ladder editor (“test.tsq”)).

One Point Advice

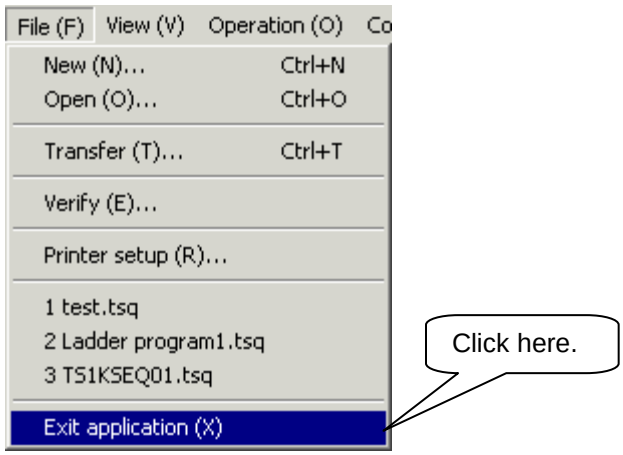
⊙ Active editor

Indicates the editor which is operated currently. It is located on the forefront among the TCPRGOS–W windows. To change over the active editor, click the editor name shown in the “Window” menu.



8.10.2 Quit of TCPRGOS-W

① Click [Quit Application] from the [File] menu. The TCPRGOS–W terminates to close the window.



Caution !

You need not quit the TCPRGOS–W after closing all active editors. You can quit by only clicking [Quit Application] from the [File] menu with the editors opened. When change was made in any editor, appropriate message box appears to ask whether each change should be saved or not.

8.11 How to Uninstall

Before uninstalling the TCPRGOS–W(E), delete the registered OCX file. Otherwise, unnecessary key is left in the registry.

Deletion of registered OCX file

Double-click and execute **"Unregist.bat"** in the install directory.

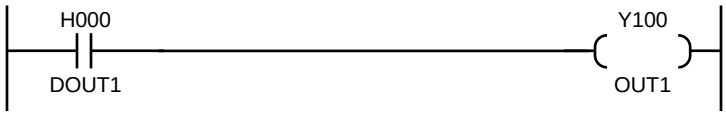
When the system displays a message box, register or deletion of OCX file has finished.

Uninstall the **TCPRGOS–W(E)** by selecting [Control panel] – [Add or delete application].

Section 9 Advice of Sequence Program Creation

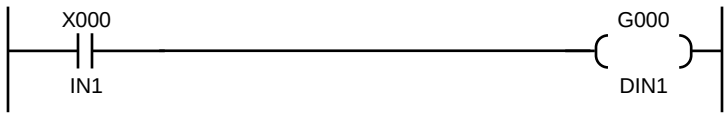
This section describes some sample circuits which you can refer to when creating a sequence program. Also refer to the standard ladder sequences in Section 11.

9.1 Turning ON/OFF External Output by DOUT Command



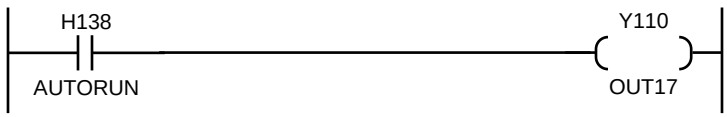
H000 turns on and off by the DOUT(1) and DOUT(–1) commands, respectively.
As Y100 operates, interlocked with H100 ON/OFF, the H000 state can be output to an external device, interlocked with the DOUT command.
(The H000 state can be output to any destination by changing the output relay address.)

9.2 Input of External Signal by DIN Command



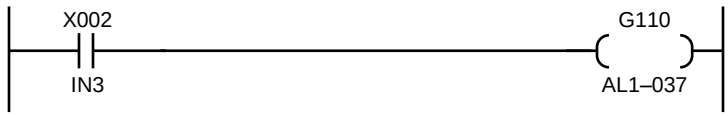
The G000 state is read by means of DIN(1) and DIN(–1) commands.
As G000 operates, interlocked with X000 ON/OFF, the X000 state can be read by the robot's DIN1 command.
(Any input destination can be ready by the DIN(1) command by changing the input relay address.)

9.3 Output of AUTORUN Signal to External Device



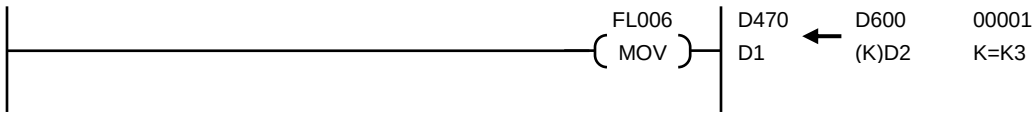
H138 is the interface relay indicating that the robot is working in the automatic operation. This state is output to an external device. (It can be output to any output designation by changing the output relay address.)

9.4 Generating Alarm in TS1000 by External Signal



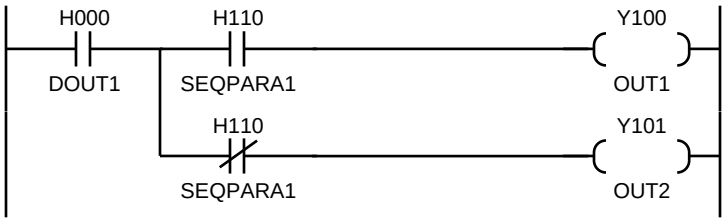
When external input IN3 (X002) turns on, G110 turns on accordingly. When G110 turns on, an alarm (1–037) is generated in the robot controller.
(The alarm can be generated by any input by changing the input relay address.)

9.5 Output of Value to Control Panel



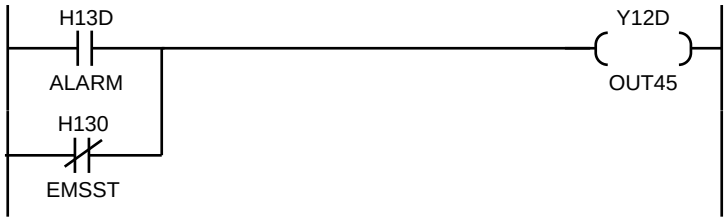
The content of D600 is transferred to D470 by extension instruction (or command) FL006. The FL command executes a command as per the signal state. In this example, as the signal is always ON, data is transferred in each scanning.
In D600, a value written to system variable PLCDATAW1 (robot language) by the robot program is stored. In the above program, this value is transferred to D470 which is the exclusive register indicating a 7-segment value of the control panel.
By changing the source register, any data such as constant and sequence operation result can be displayed.

9.6 Change of Output Destination by Sequence Parameter



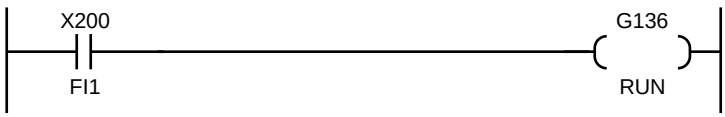
H110 is the relay which can be turned on and off by user parameter of TS3000.
When "1" (ON) is specified for appropriate parameter, DOUT(1) turns on Y100.
When "0" (OFF) is set, DOUT(1) turns on Y101. When SEQPARA* of H110 ~ H117 is used, you can change the sequence motion by the robot parameter without changing the ladder sequence.

9.7 Fault Signal Turns On Even When Emergency Stop Pushbutton Was Pressed



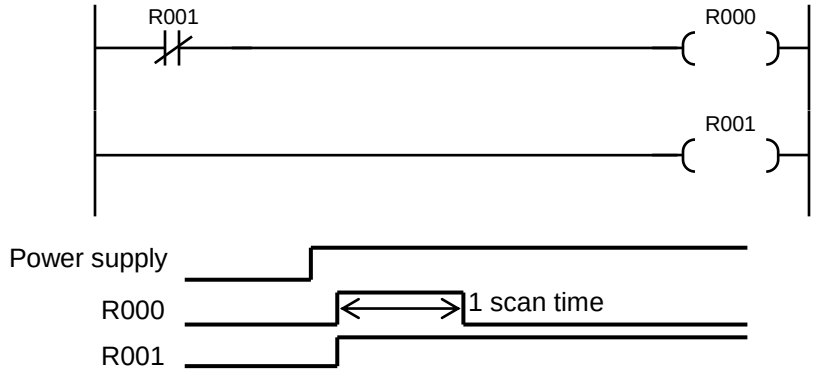
The ALARM (H13D) signal is set to ON when an alarm occurs. However, the fault signal does not turn on when the emergency stop pushbutton is turned on (8-014 occurs). By also monitoring the EMSST state, the states when an alarm occurs or when an emergency stop pushbutton was pressed can be output.

9.8 Starting the Robot by an Input Signal from Fieldbus



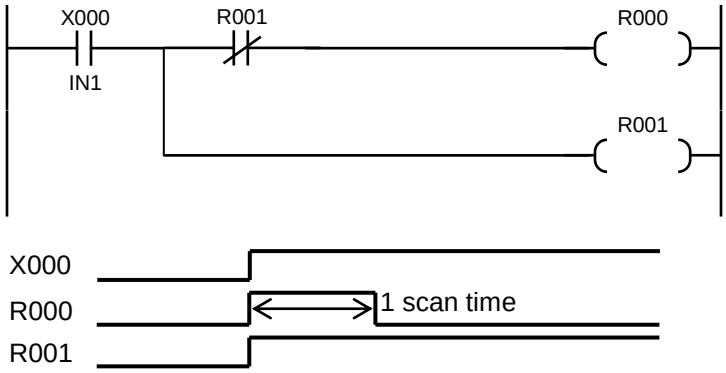
When (X200) in the input from a fieldbus connection is set to ON, G136 turns on, and the robot program runs.

9.9 Circuit Generating 1 Pulse at Power ON



Just after the power is turned on, R000 turns on in the initial scanning. As R001 is ON in the 2nd scanning, R000 turns off.
This can be realized by one (1) command when F*63 is used.

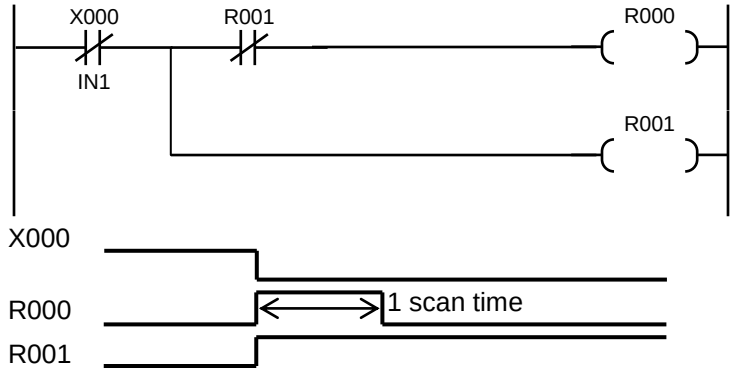
9.10 Circuit Generating 1 Pulse at Start



At the start of input X000, one (1) pulse is generated.

Note: If the PLC starts running while X000 is ON, R000 turns on in the 1st scanning.

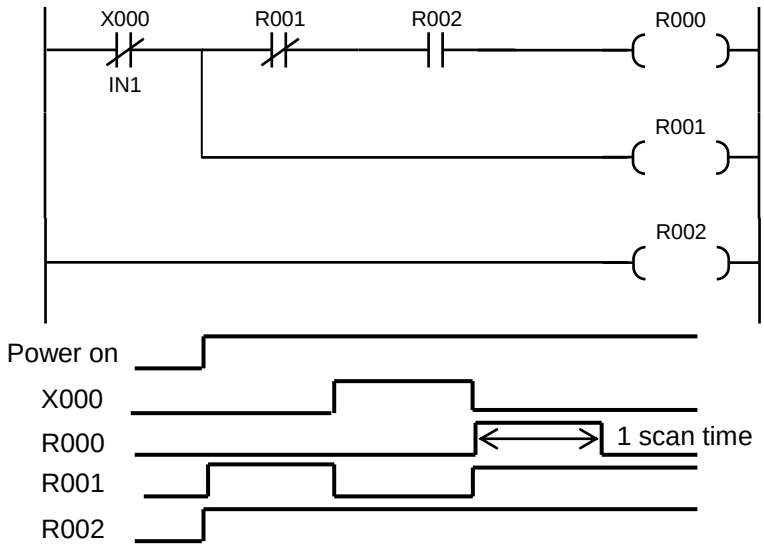
9.11 Circuit Generating 1 Pulse at Fall (1)



With the fall of input X000, one (1) pulse is generated.

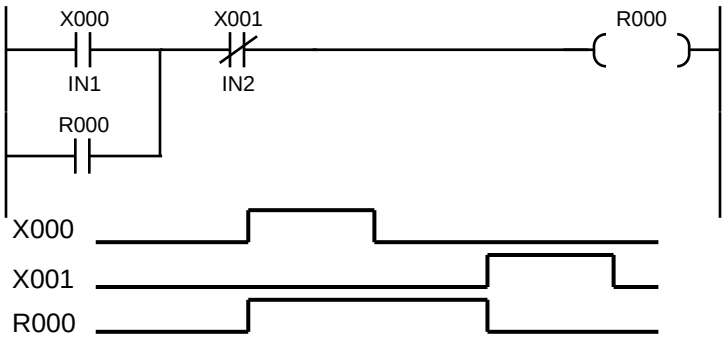
Caution: When X000 is OFF, R000 turns on in the 1st scanning after power ON. (Necessary measures are taken in Para. 9.10.)
If the PLC starts running while X000 is OFF, R000 turns on in the 1st scanning.

9.12 Circuit Generating 1 Pulse at Fall (2)



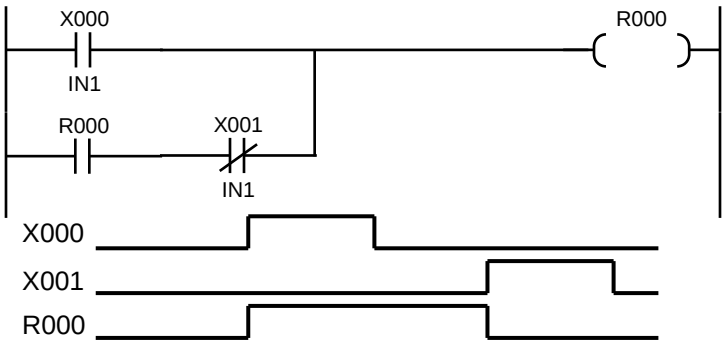
With the fall of input X000, one (1) pulse is generated. R000 will not turn on at power ON.

9.13 Self-Holding Circuit (Priority Is Given to Reset)



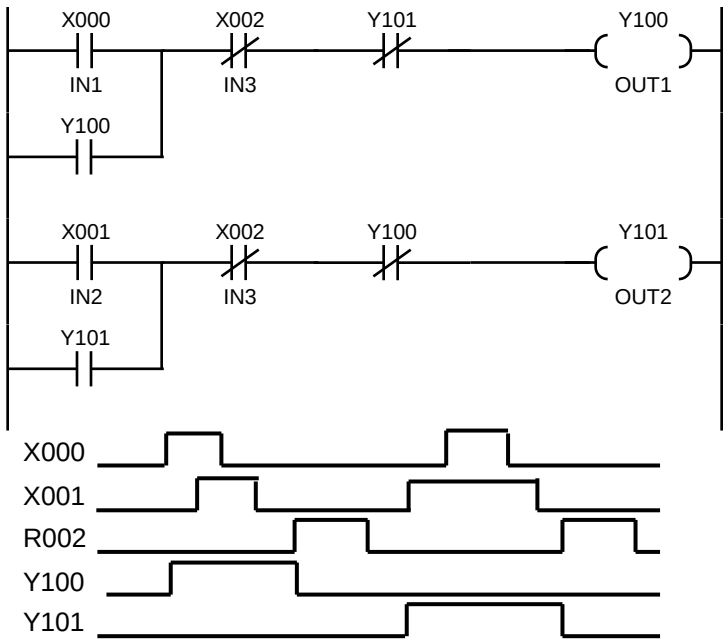
When X000 is turned on once while X001 is OFF, output R000 turns on, which is kept even after X000 is OFF. When X001 turns on, R000 turns off.

9.14 Self-Holding Circuit (Priority Is Given to Set)



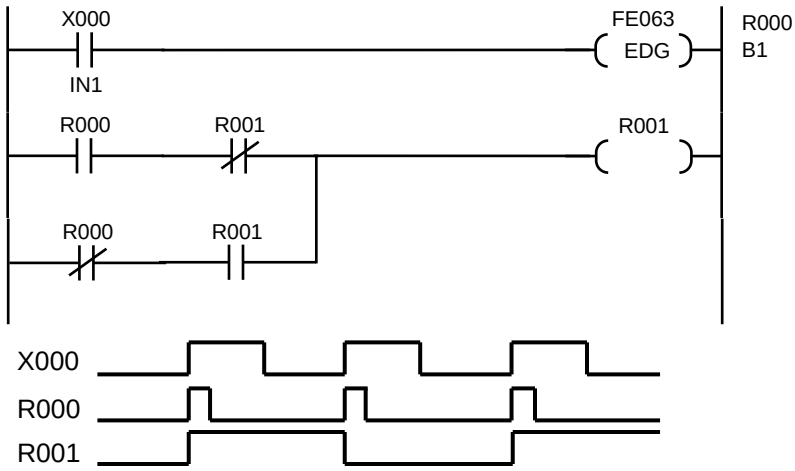
When X000 is turned on once while X001 is OFF, output R000 turns on, which is kept even after X000 is OFF. When X001 turns on, R000 turns off.
If X000 is ON while X001 is ON, R000 turns on.

9.15 Interlock Circuit



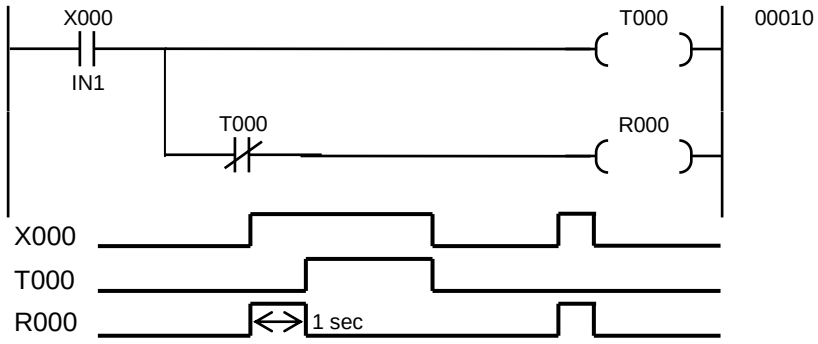
Either X000 or X001 whose circuit turns on first is ON, and the other circuit will not turn on. When both circuits are turned on at the same time, the upper circuit (Y100) turns on by the scanning system. Useful for the motor CW/CCW circuit.

9.16 Alternate Circuit



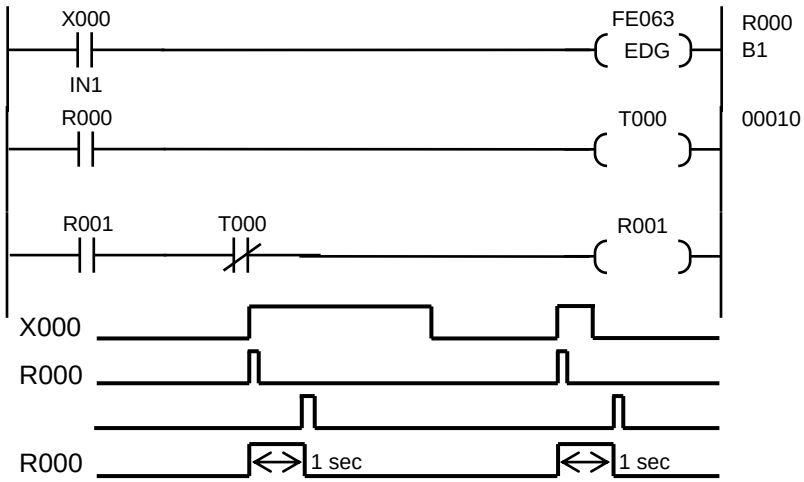
Every time input X000 has turned on, output R001 is reversed. Thus, an alternate output can be given, using the normal open contact of the momentary switch.

9.17 Non-Modal Timer Circuit



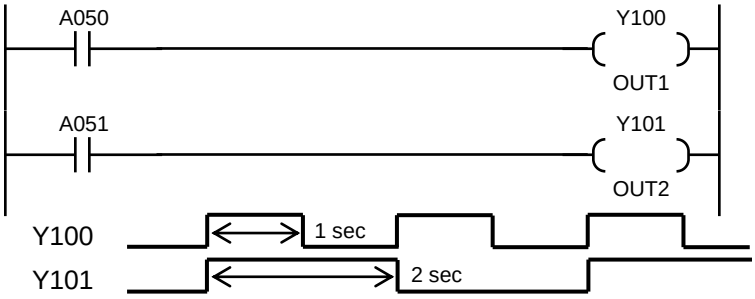
When input X000 turns on, R000 turns on for the time set on the timer.
If the ON time of input X000 is shorter than the timer-set value, the ON time of R000 is identical with the ON time of X000.

9.18 Non-Modal Start Timer Circuit



When input X000 turns on, R000 turns on for the time set on the timer.

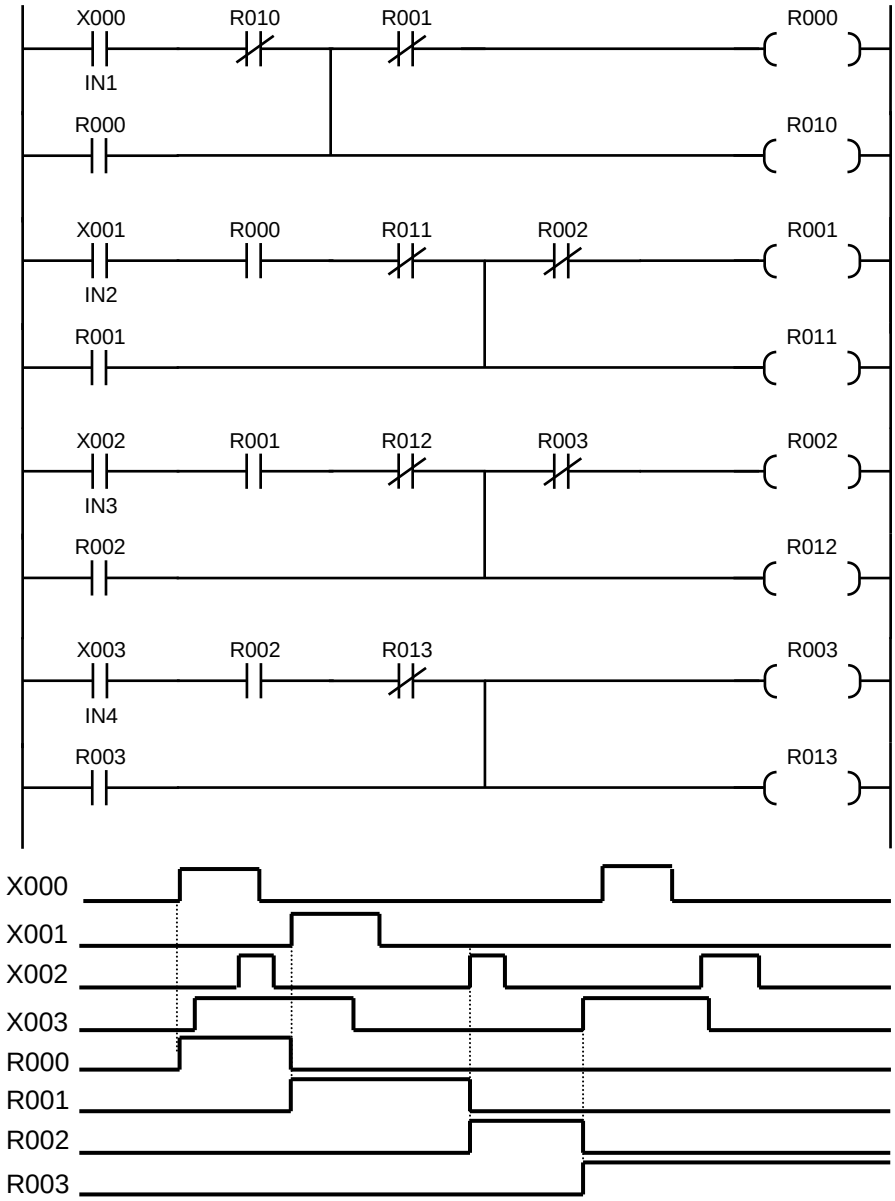
9.19 Circuit Turning ON and OFF at Predetermined Intervals



A050 is the special auxiliary relay which repeats ON and OFF at one (1)-second intervals.

A051 is the special auxiliary relay which repeats ON and OFF at two (2)-second intervals.

9.20 Step Sequence Circuit



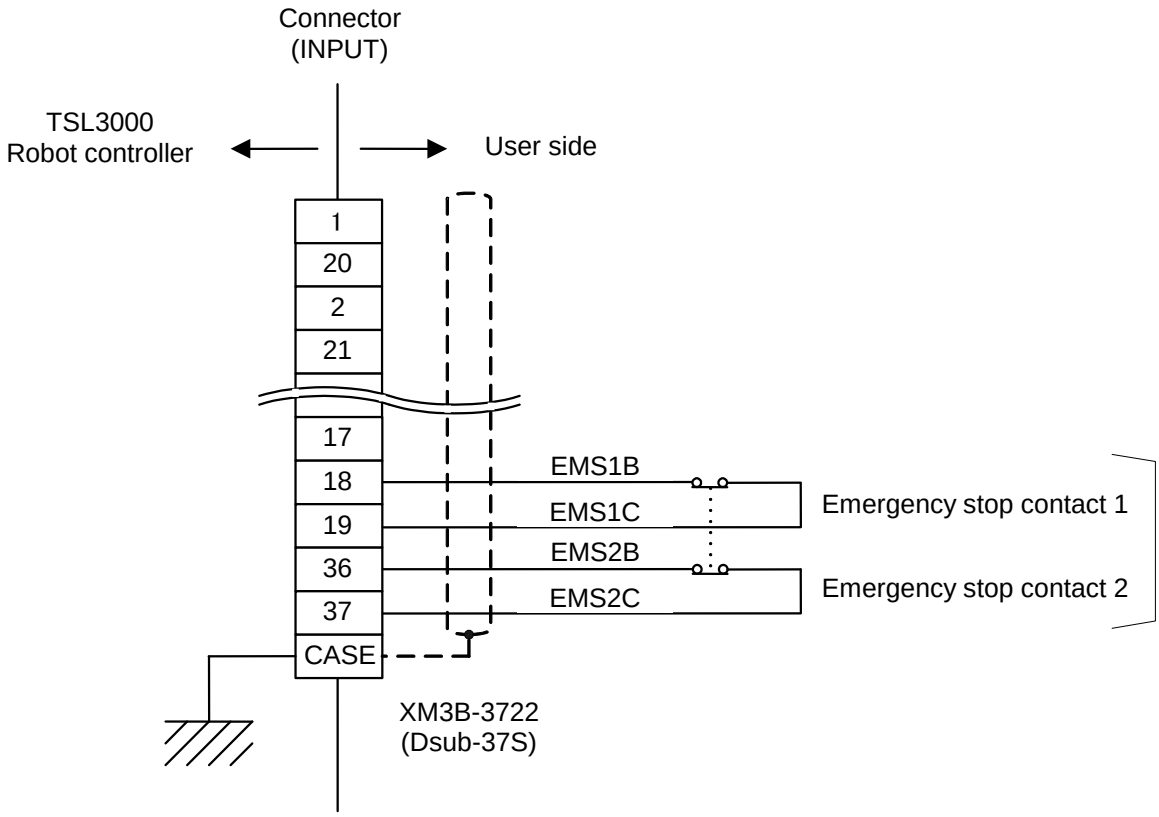
When X000 turns on, R000 turns on accordingly.
When X001 is ON, R001 is ON with R000 OFF.
When X002 is ON, R002 is ON with R001 OFF.

Section 10 Troubleshooting

10.1 Robot Servo Will Not Turn ON

① Is interface relay **SVOFF (R33C)** turned on?
When the SVOFF signal turns off, the servo turns off. Design the sequence and external circuit, therefore, so that interface relay SVOFF can be turned on.

② Is the alarm (8–014 Emergency Stop SW ON) generated?
If this alarm is generated, make sure that the EMERGENCY stop pushbutton switch is not pressed. In addition to this, make sure that the TP or TP jumper connector is connected.
Wiring of the EMERGENCY stop switch is the special line. Perform check on the wiring of EMS** and EMA** pin connected to the EMS connector.



10.2 Program Cannot Run

① Is the program selected?

② Is the servo ON?

③ Is interface relay STOP (R338) ON?
When the STOP signal turns off, the robot stops. Design the sequence and external circuit, therefore, so that interface relay STOP can be turned on.

④ Is interface relay BREAK (R33B) ON?
Like the STOP signal, when the BREAK signal turns off, the robot slows down and stops.

10.3 Sequence Program Cannot Be Transferred by TCPRGOS

The sequence program areas which can be used by the user are areas 1 and 2. Change the sequence area, referring to Para. 2.4.

10.4 Sequence Program Is Cleared at Power ON

Sequence program area 1 is the RAM area for debugging. The sequence program is cleared after the power is turned off, then on again. To use the sequence program after debugging (check of operation), transfer it to program area 2 beforehand.

10.5 TCPRGOS Will Not Start

Are the specifications of your personal computer identical with those of the TCPRGOS? (See Para. 8.1.)

If the TCPRGOS will not operate just after it is installed, the DLL file version may be old. When this is the case, install the newest DLL file according to the following procedures.

① Quit all other applications, then make sure that the following files are present in the system directory.

MFC42.DLL MFC42LOC.DLL MSVCRT.DLL
CMCTLJP.DLL COMCTL32.OCX

② Confirm the version of each file. If it is older than the version below, delete the file.

File name	Version
MFC42.DLL	5.0.000
MFC42LOC.DLL	5.0.000
MSVCRT.DLL	5.00.7303
CMCTLJP.DLL	5.01.4319
COMCTL32.OCX	5.01.4319

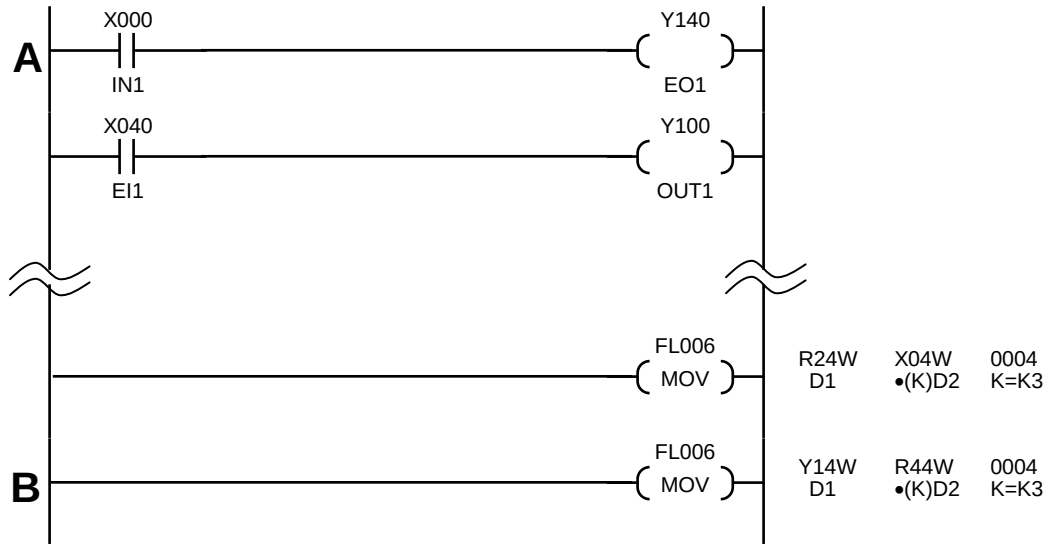
Right-click a relevant file and select [Property]. When the dialog window is displayed, select the version information tag and check for the version.

③ Install the TCPRGOS software again, referring to Para. 8.3.

10.6 Extension I/O Does Not Operate (Malfunctions)

① The extension I/O RUN lamp is off.
See Para. 3.8 and check the connection and user parameter settings.

② The extension I/O sequence is defined twice.



In the above figure, the X000 input is output to Y140 at **A**, and the state of R440 to R47F is transferred (output) to Y140 to Y17F at **B**. As a result, the R440 state is output to Y140, and the **A** circuit result is not output.
To prevent this problem, the B circuit must be deleted, but this will also delete the transfer to Y141 to Y17F. Therefore, a separate circuit needs to be added.

10.7 Sequence-Related Alarm Occurred in TSL3000

8–227 PLC STOP

Possible Cause: The sequence of the PLC remains stopped.

Remedies: If the sequence program has been stopped intentionally from the TCPRGOS, there is no problem at all. Turn the power off, then on again, or start the sequence program from the TCPRGOS.
This alarm may occur if the hardware went wrong or an error was found in the sequence program. When this happens, the following alarm (1–***) is also generated at the same time. Identify and remove the cause of the error to start the PLC.

1–164 PLC Backup data error

Possible Cause: The sequence program of TCmini has been destroyed.

Remedy: Transfer the program again, using the TCPRGOS.
If a backup RAM (domain 3) is used, set to domain 1, and turn the power off, then on again to transfer the program.

1–166 PLC Remote unit error

Possible Causes:
① The setting of connecting the extension I/O unit is not identical with the user parameter setting.
② Wiring of the cable is done incorrectly.
③ The connected I/O unit will not operate normally.

Remedies:
① Perform check on the user parameter setting. (See Para. 3.7.3.)
② Connect the cable. (See Para. 3.7.2.)
③ If the external power supply is used, turn on the power of the extension I/O unit before the controller power.
If the I/O unit has malfunctioned, contact our after-sale service agent in your territory.

1–169 PLC Undefined label

Possible Cause: The label used in the sequence program is not defined.

Remedy: Correct the sequence program, using the TCPRGOS.

1–170 PLC Invalid command

Possible Cause: An illegal command (i.e., instruction word) is used in the sequence program.

Remedy: Correct the sequence program, using the TCPRGOS.

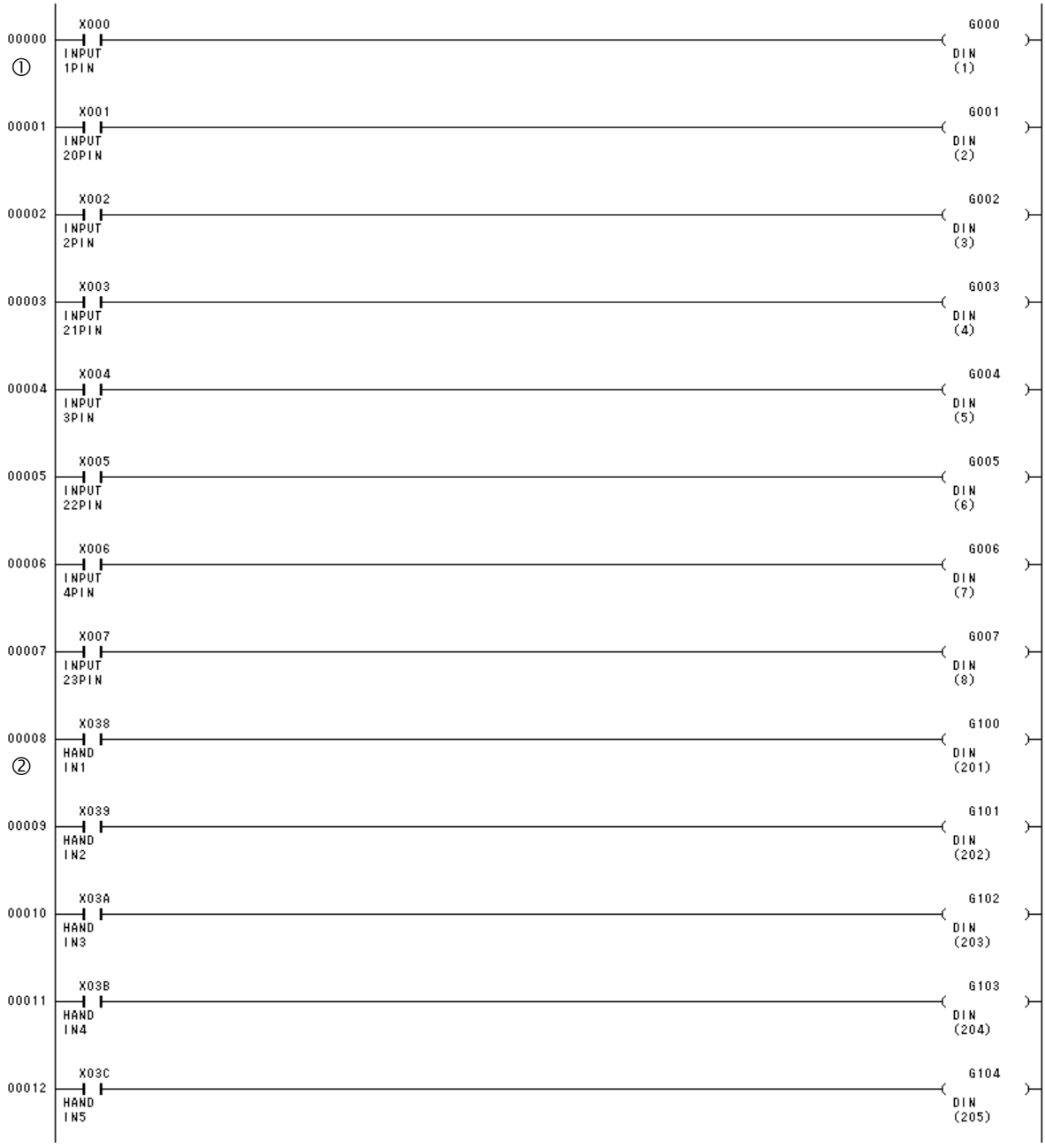
1–173 PLC Overlap label

Possible Cause: Duplicate definition of the label used in the sequence program.

Remedy: Correct the sequence program, using the TCPRGOS.

Section 11 Standard ladder sequence program

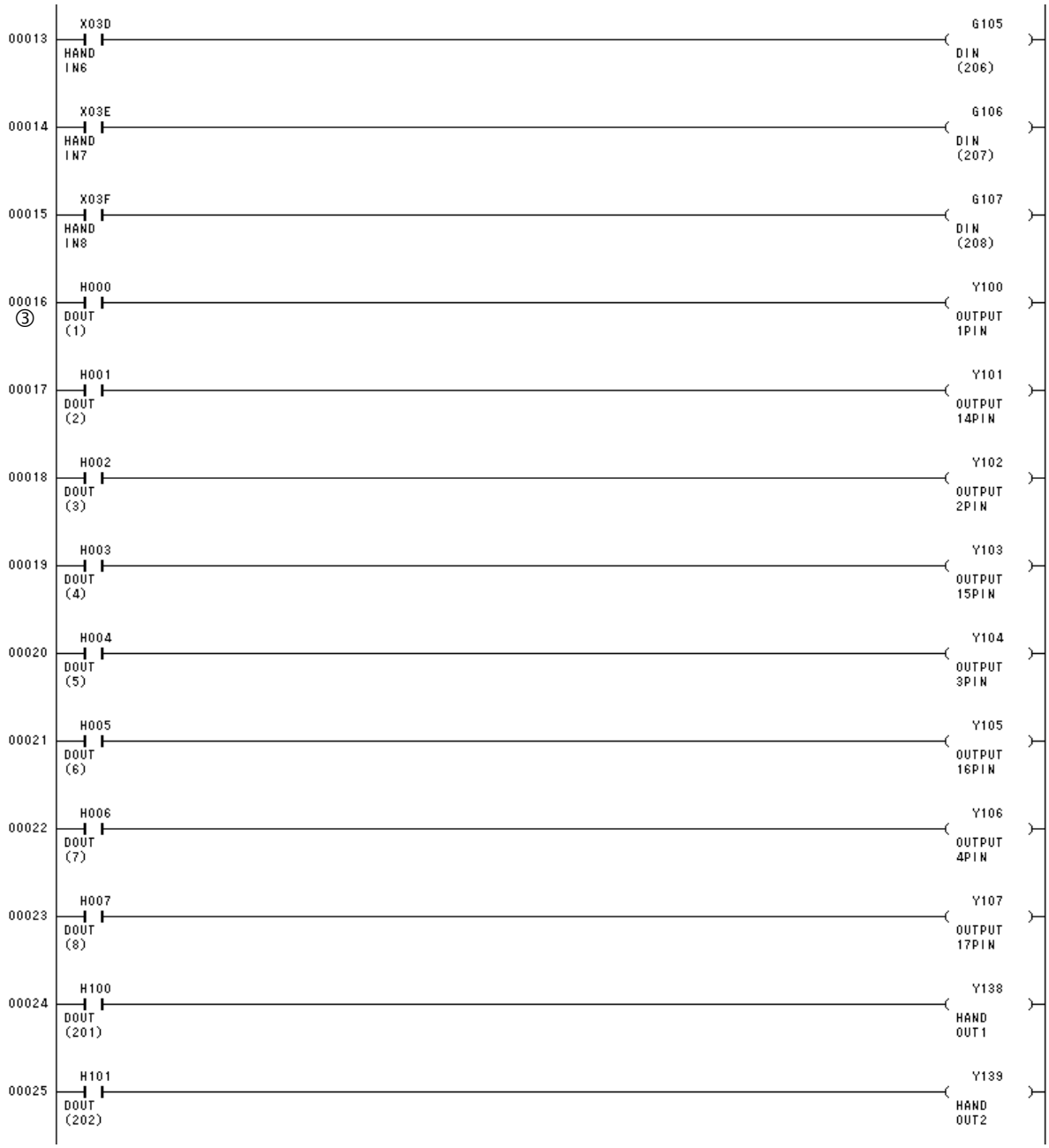
Standard ladder sequence program



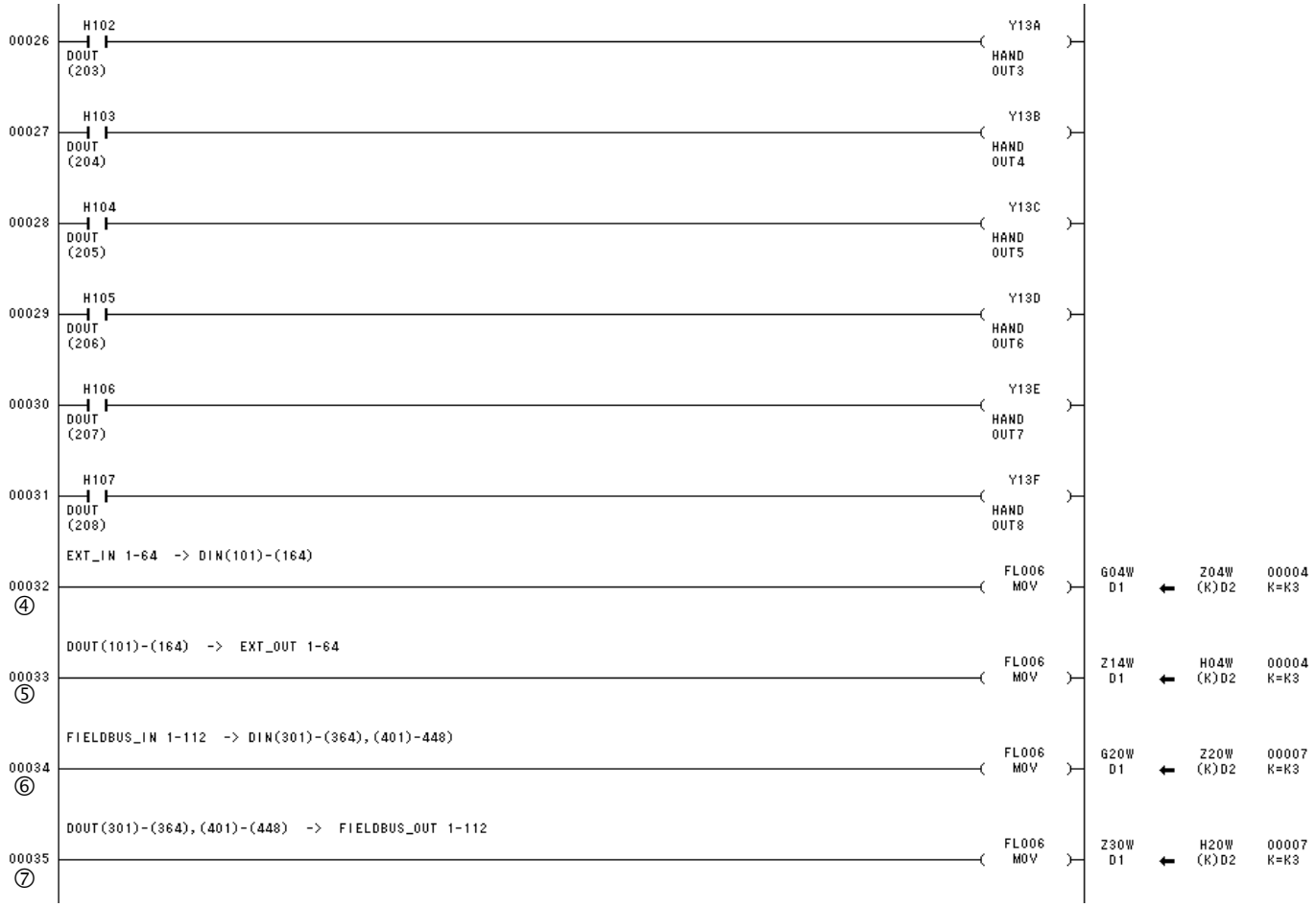
*Explanations of typical circuits are shown below.

① The external input (input connector pin no. 1) contact X000 is connected to the G000 (DIN1) coil, and the robot program "DIN(1)" input is set.

② The hand input (hand pin no. 1) contact X038 is connected to the G100 (DIN201) coil, and the robot program "DIN(201)" input is set.



③ The robot program "DOUT(1)" output coil H000(DOUT1) is connected to the Y100 (OUT1) coil and output to an external output (output connector pin no. 1).

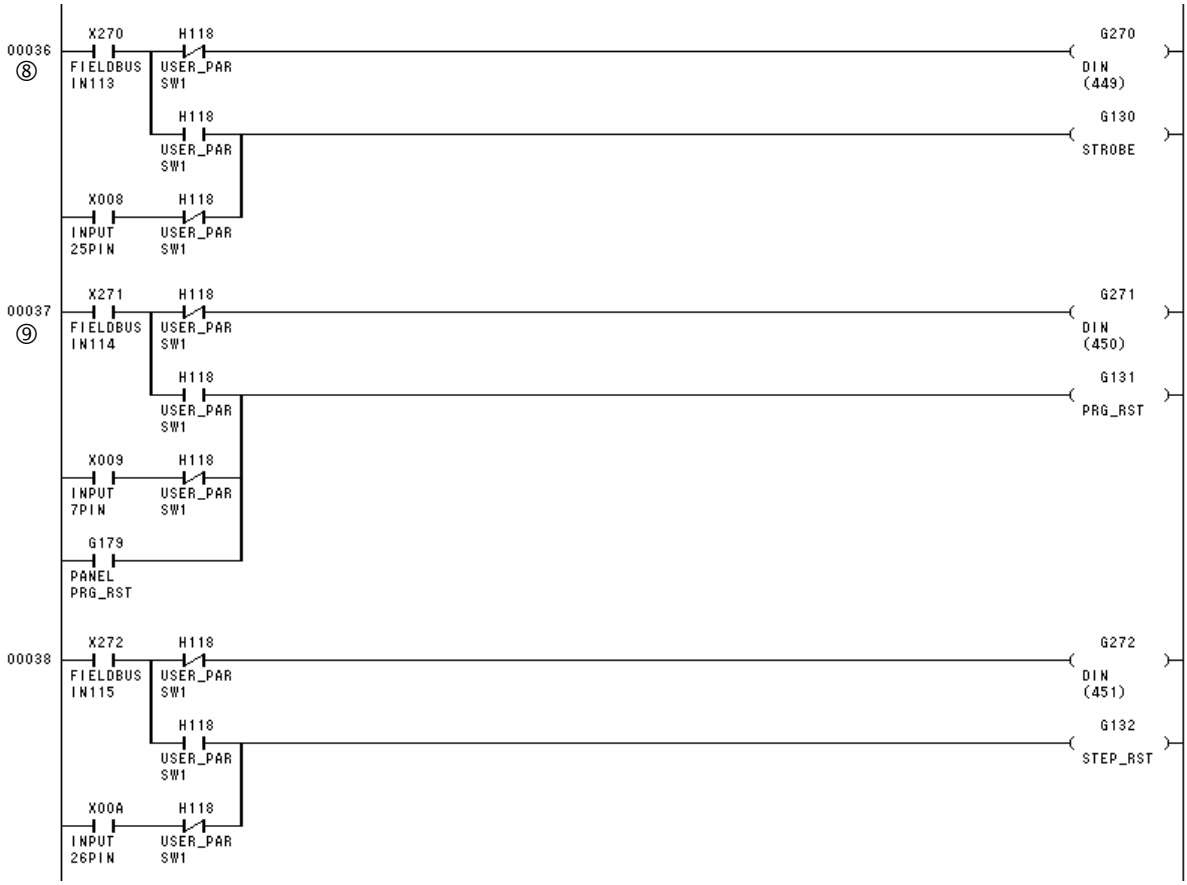


④ The extension input (EI1 to EI64) is used as the robot program "DIN (101 to 164)" input.

⑤ The robot program "DOUT (101 to 164)" is output to the extension output (EO1 to EO64).

⑥ The fieldbus input (FI1 to FI112) is used as the robot program "DIN (301 to 364) and DIN (401 to 448)" input.

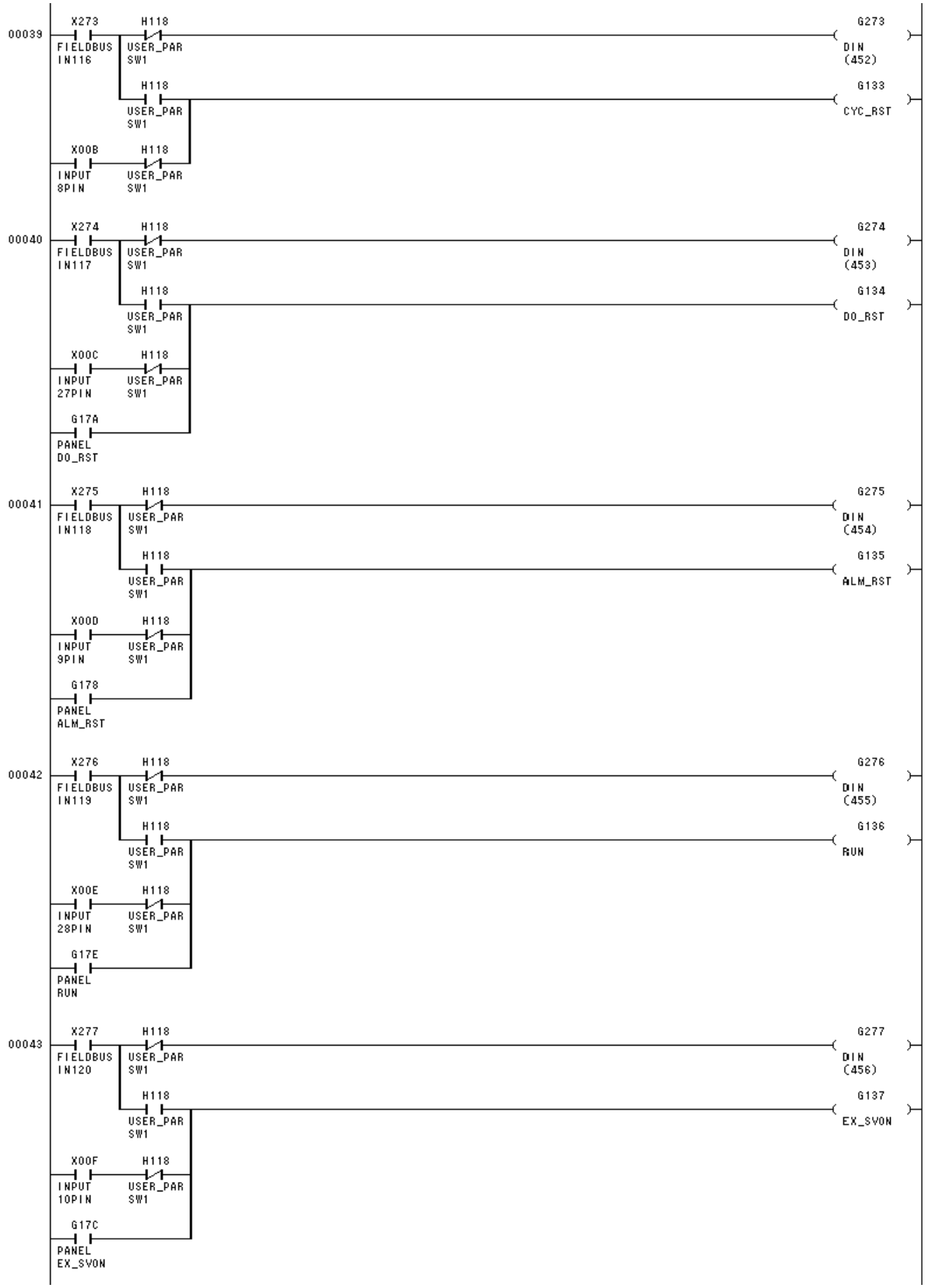
⑦ The robot program "DOUT (301 to 364) and DOUT (401 to 448)" is output to the fieldbus output (FO1 to FO112).

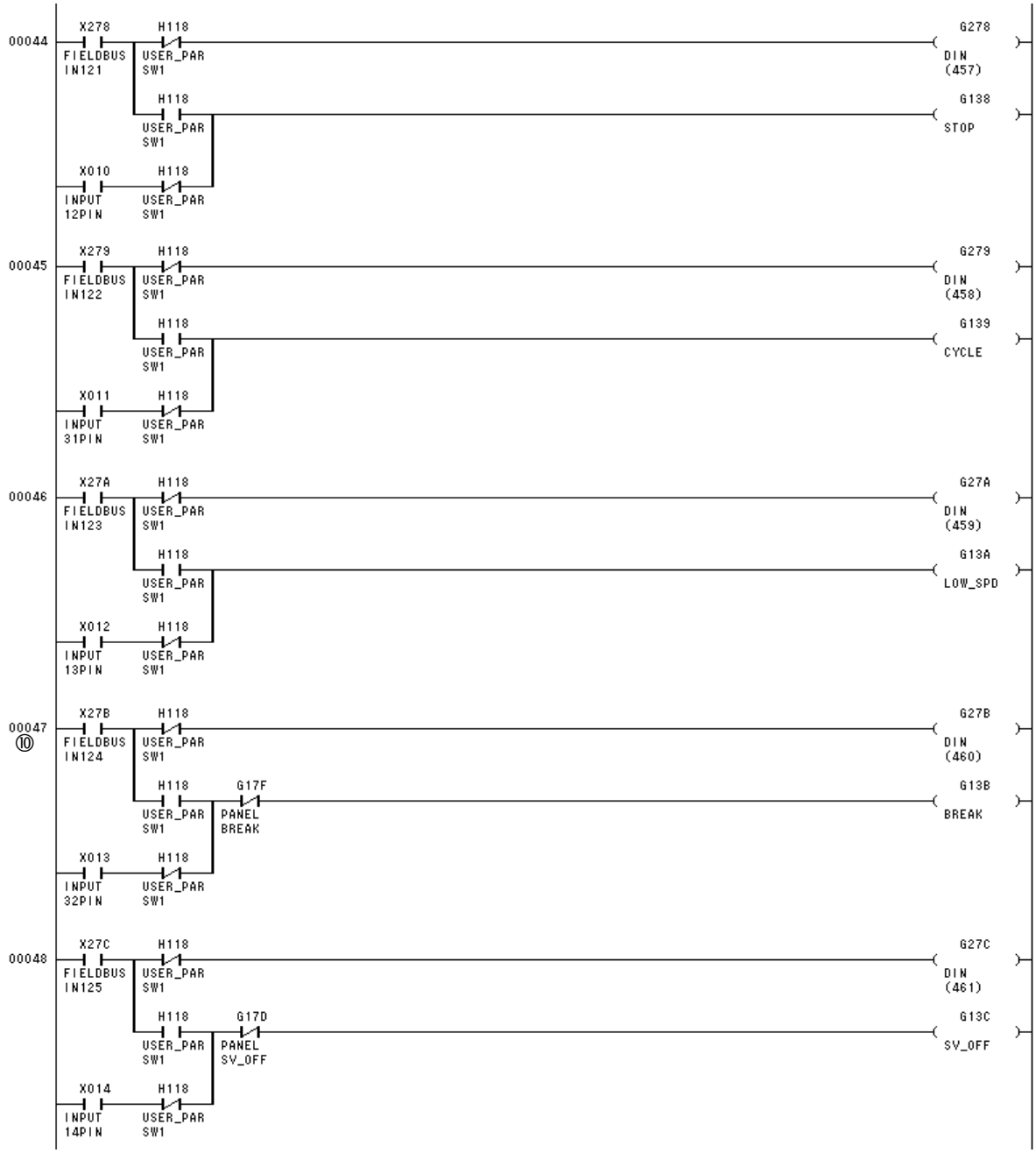


⑧ When H118 is OFF, the fieldbus input (FI113) contact X270 is connected to the G270 (DIN449) coil, the robot program "DIN(449)" input is set, the external input (input connector pin no. 25) contact X008 is connected to the G130 (strobe) coil, and the system signal "STROBE" input is set.
When H118 is ON, the fieldbus input (FI113) contact X270 is connected to the G130 (strobe) coil, and the system signal "STROBE" input is set. At this time, the external input (input connector pin no. 25) is disabled.

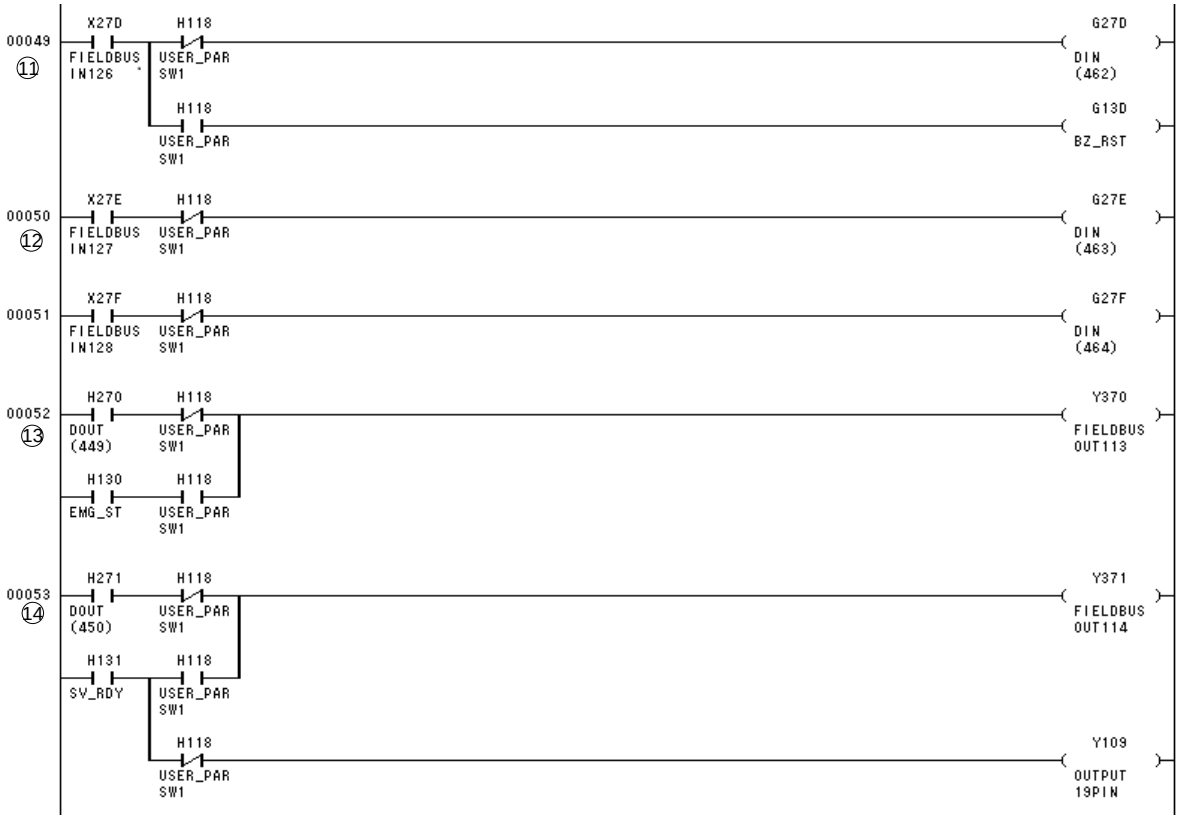
⑨ When H118 is OFF, the fieldbus input (FI114) contact X271 is connected to the G271 (DIN450) coil, the robot program "DIN(450)" input is set, the external input (input connector pin no. 7) contact X009 is connected to the G131 (PRG_RST) coil, and the system signal "PRG_RST input is set.
When H118 is ON, the fieldbus input (FI114) contact X271 is connected to the G131 (PRG_RST) coil, and the system signal "PRG_RST" input is set. At this time, the external input (input connector pin no. 7) is disabled.
When connected to the touch panel, the touch panel input (PANEL_PRG_RST) contact G179 is connected to the G131 (PRG_RST) coil, and the system signal "PRG_RST" input is set.

* The H118 signal is switched on and off by the user parameter [U35] setting. For details, see the User Parameter Operation Manual.


00039
X273
FIELDBUS IN116
H118
USER_PAR SW1
G273
DIN (452)
H118
USER_PAR SW1
G133
CYC_RST
X00B
INPUT 8PIN
H118
USER_PAR SW1
00040
X274
FIELDBUS IN117
H118
USER_PAR SW1
G274
DIN (453)
H118
USER_PAR SW1
G134
DO_RST
X00C
INPUT 27PIN
H118
USER_PAR SW1
G17A
PANEL DO_RST
00041
X275
FIELDBUS IN118
H118
USER_PAR SW1
G275
DIN (454)
H118
USER_PAR SW1
G135
ALM_RST
X00D
INPUT 9PIN
H118
USER_PAR SW1
G178
PANEL ALM_RST
00042
X276
FIELDBUS IN119
H118
USER_PAR SW1
G276
DIN (455)
H118
USER_PAR SW1
G136
RUN
X00E
INPUT 28PIN
H118
USER_PAR SW1
G17E
PANEL RUN
00043
X277
FIELDBUS IN120
H118
USER_PAR SW1
G277
DIN (456)
H118
USER_PAR SW1
G137
EX_SVON
X00F
INPUT 10PIN
H118
USER_PAR SW1
G17C
PANEL EX_SVON



⑩ When H118 is OFF, the fieldbus input (FI124) contact X27B is connected to the G27B (DIN460) coil, the robot program "DIN(460)" input is set, the external input (input connector pin no. 32) contact X013 is connected to the G13B (break) coil, and the system signal "BREAK" input is set.
When H118 is ON, the fieldbus input (FI124) contact X27B is connected to the G13B (break) coil, and the system signal "BREAK" input is set. At this time, the external input (input connector pin no. 32) is disabled.
When connected to the touch panel, the touch panel input (PANEL_BREAK) contact G17F is connected to the G13B (BREAK) coil, and the system signal "BREAK" input is set.

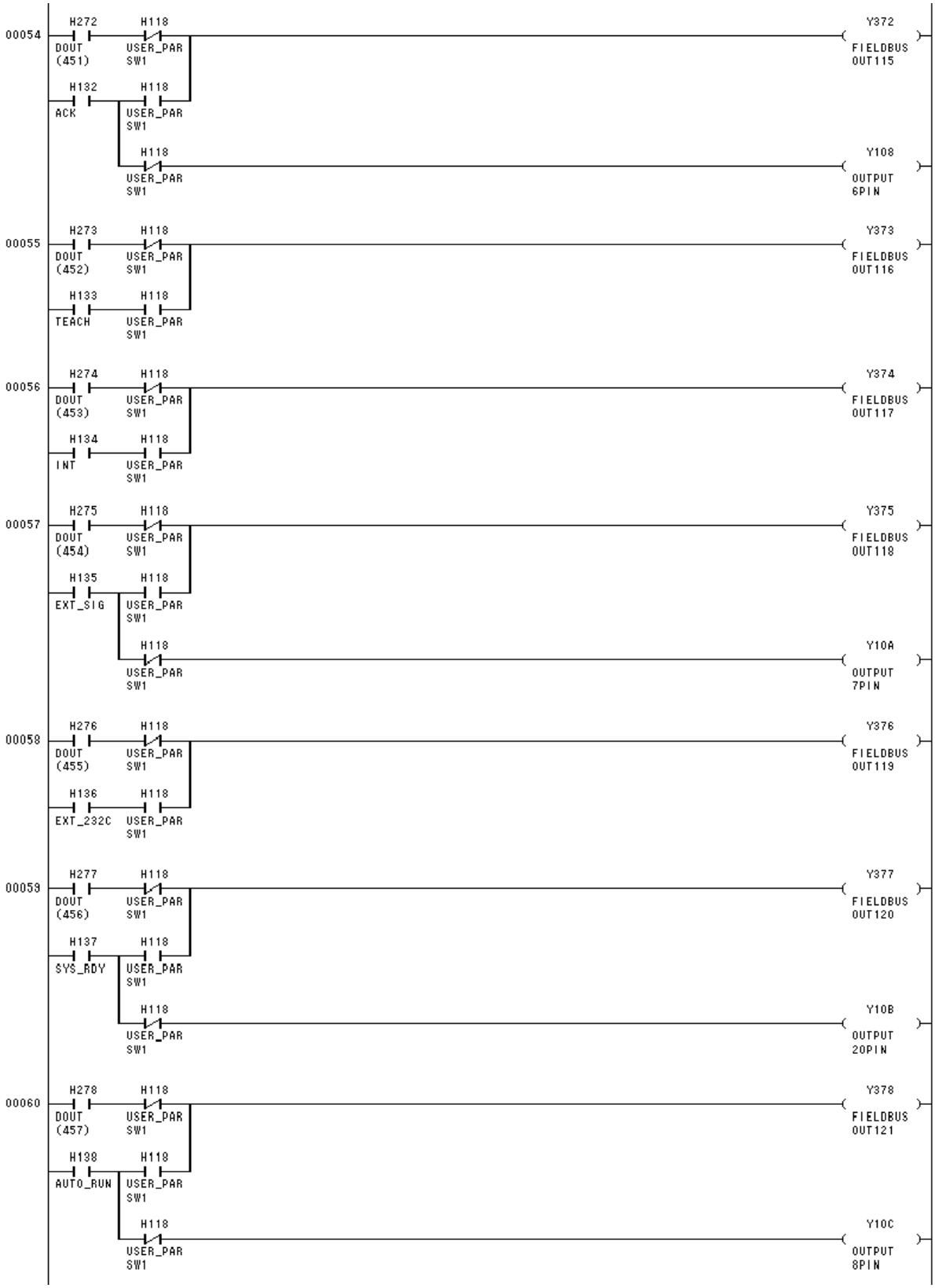


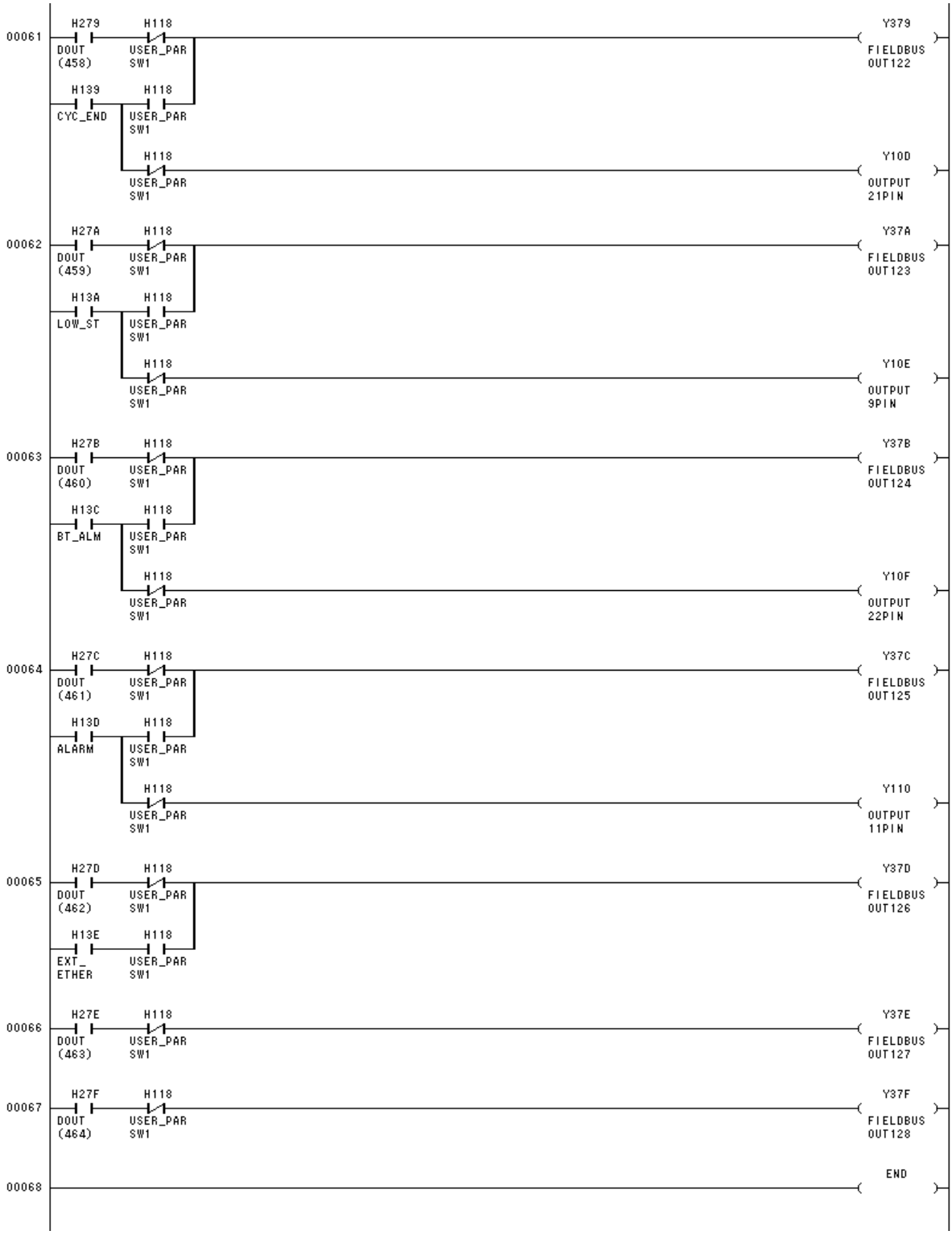
⑪ When H118 is OFF, the fieldbus input (FI126) contact X27D is connected to the G27D (DIN462) coil, and the robot program "DIN(462)" input is set.
When H118 is ON, the fieldbus input (FI126) contact X27D is connected to the G13D (BZ_RST) coil, and the system signal "BZ_RST" input is set.

⑫ When H118 is OFF, the fieldbus input (FI127) contact X27E is connected to the G27E (DIN463) coil, and the robot program "DIN(463)" input is set.

⑬ When H118 is OFF, the robot program "DOUT(449)" output coil H270(DOUT449) is connected to the Y370(FIELDBUSOUT113) coil and is output to the fieldbus output (FO113).
When H118 is ON, the system signal "EMG_ST" output coil H130 (EMG_ST) is connected to the Y370(FIELDBUSOUT113) coil and is output to the fieldbus output (FO113).

⑭ When H118 is OFF, the robot program "DOUT(450)" output coil H271(DOUT450) is connected to the Y371(FIELDBUSOUT114) coil and is output to the fieldbus output (FO114), and the system signal "SV_RDY" output coil H131(SV_RDY) is connected to the Y109(OUT10) coil and is output to the external output (output connector pin no. 19).
When H118 is ON, the system signal "SV_RDY" output coil H131 (SV_RDY) is connected to the Y371(FIELDBUSOUT114) coil and is output to the fieldbus output (FO114).


00054
H272
DOUT
(451)
H118
USER_PAR
SW1
H132
ACK
H118
USER_PAR
SW1
H118
USER_PAR
SW1
Y372
FIELDBUS
OUT115
Y108
OUTPUT
6PIN
00055
H273
DOUT
(452)
H118
USER_PAR
SW1
H133
TEACH
H118
USER_PAR
SW1
Y373
FIELDBUS
OUT116
00056
H274
DOUT
(453)
H118
USER_PAR
SW1
H134
INT
H118
USER_PAR
SW1
Y374
FIELDBUS
OUT117
00057
H275
DOUT
(454)
H118
USER_PAR
SW1
H135
EXT_SIG
H118
USER_PAR
SW1
H118
USER_PAR
SW1
Y375
FIELDBUS
OUT118
Y10A
OUTPUT
7PIN
00058
H276
DOUT
(455)
H118
USER_PAR
SW1
H136
EXT_232C
H118
USER_PAR
SW1
Y376
FIELDBUS
OUT119
00059
H277
DOUT
(456)
H118
USER_PAR
SW1
H137
SYS_RDY
H118
USER_PAR
SW1
H118
USER_PAR
SW1
Y377
FIELDBUS
OUT120
Y10B
OUTPUT
20PIN
00060
H278
DOUT
(457)
H118
USER_PAR
SW1
H138
AUTO_RUN
H118
USER_PAR
SW1
H118
USER_PAR
SW1
Y378
FIELDBUS
OUT121
Y10C
OUTPUT
8PIN


00061
H279
DOUT
(458)
H118
USER_PAR
SW1
H139
CYC_END
H118
USER_PAR
SW1
H118
USER_PAR
SW1
Y379
FIELDBUS
OUT122
Y10D
OUTPUT
21PIN
00062
H27A
DOUT
(459)
H118
USER_PAR
SW1
H13A
LOW_ST
H118
USER_PAR
SW1
H118
USER_PAR
SW1
Y37A
FIELDBUS
OUT123
Y10E
OUTPUT
9PIN
00063
H27B
DOUT
(460)
H118
USER_PAR
SW1
H13C
BT_ALM
H118
USER_PAR
SW1
H118
USER_PAR
SW1
Y37B
FIELDBUS
OUT124
Y10F
OUTPUT
22PIN
00064
H27C
DOUT
(461)
H118
USER_PAR
SW1
H13D
ALARM
H118
USER_PAR
SW1
H118
USER_PAR
SW1
Y37C
FIELDBUS
OUT125
Y110
OUTPUT
11PIN
00065
H27D
DOUT
(462)
H118
USER_PAR
SW1
H13E
EXT_
ETHER
H118
USER_PAR
SW1
Y37D
FIELDBUS
OUT126
00066
H27E
DOUT
(463)
H118
USER_PAR
SW1
Y37E
FIELDBUS
OUT127
00067
H27F
DOUT
(464)
H118
USER_PAR
SW1
Y37F
FIELDBUS
OUT128
00068
END

Touch Panel Connection

The standard ladder sequence includes contacts where operation can be performed from a touch panel.

When the G130 to G13D coils are operated from a touch panel, a conflict occurs with the input from the SYSTEM connector, and they do not operate properly, and so operate the coils shown below.

For the output signals to the touch panel, refer directly to H130 to H13E.

Touch panel input assignments

Touch panel input (address)	Signal name (address)
PANEL_ALM_RST (G178)	ALM_RST (Alarm reset) (G135)
PANEL_PRG_RST (G179)	PRG_RST (Program reset) (G131)
PANEL_DO_RST (G17A)	DO_RST (Output signal reset) (G134)
PANEL_EX_SVON (G17C)	EX_SVON (External input servo ON) (G137)
PANEL_SV_OFF (G17D)	SV_OFF (Servo OFF) (G13C)
PANEL_RUN (G17E)	RUN (Run) (G136)
PANEL_BREAK (G17F)	BREAK (Decelerate and stop) (G13B)

Section 12 Simple PLC Data Communication Function

In the Simple PLC data communication function, commands are set to the register for obtaining and setting the robot current position and teaching point position.
The Simple PLC data communication function has two types of operation: command mode and monitor mode. Operation can be selected based on the command to be written. In command mode, one reply is sent to a command. In monitor mode, the latest data for a command is constantly saved. (The data remains saved until the next command is issued.)

12.1 Monitor Mode

In monitor mode, the controller constantly saves the latest value to DATA_RESP (comment response register) for the command that was set to DATA_CMD (command set register).

12.1.1 Monitor Mode Register

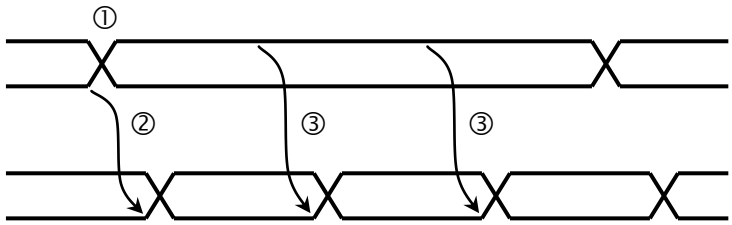
DATA_CMD (Command set register) D460 to D46F
This is the command register for the Simple PLC data communication function. Set the command corresponding to the data that you want to obtain.

DATA_RESP (Command response register) D660 to D66F
This is the command response register for the Simple PLC data communication function. The response to the command that was set is saved.

12.1.2 Monitor Mode Operation Sequence

DATA_CMD
(Comment set register) D46*

DATA_RESP
(Command response register)
D66*



① Set the command to DATA_CMD (command set register).
② The controller saves the response to the command in DATA_RESP (command response register).
③ The latest response data is constantly saved. (The data remains saved until the next command is issued.)

12.1.3 Monitor Mode Command List

Joint coordinate current position acquire command (Monitor mode): E310

DATA_CMD (Command set register) TCmini→Robot

D46*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value															0	0x E310

Register	Item	Setting value
D460	Command (Hexadecimal format)	0xE310
D461	Task	0 (Always specify 0)
D462 to D46F	Not used	0

DATA_RESP (Command response register) Robot→TCmini

D66*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
SCARA			5-axis joint coordinate current position		4-axis joint coordinate current position		3-axis joint coordinate current position		2-axis joint coordinate current position		1-axis joint coordinate current position		Orientation		0	0x E310
6-axis	6-axis joint coordinate current position															

Register	Item	Response value
D660	Command (Hexadecimal format)	0xE310
D661	Task	0 (Always 0)
D662	Not used	0
D663	Orientation (CONFIG)	0: FREE/1:LEFTY/2:RIGHTY (SCARA) 0 to 22222 (6-axis)*
D664, D665	1-axis joint coordinate current position	Value of 1-axis joint coordinate current position×1000
D666, D667	2-axis joint coordinate current position	Value of 2-axis joint coordinate current position×1000
D668, D669	3-axis joint coordinate current position	Value of 3-axis joint coordinate current position×1000
D66A, D66B	4-axis joint coordinate current position	Value of 4-axis joint coordinate current position×1000
D66C, D66D	5-axis joint coordinate current position	Value of 5-axis joint coordinate current position×1000
D66E, D66F	Not used (SCARA) 6-axis joint coordinate current position (6-axis)	0 (SCARA) Value of 6-axis joint coordinate current position×1000 (6-axis)

* For details on the 6-axis robot orientation (CONFIG) notation, see section 12.3.

World coordinate current position acquire command (Monitor mode): E311

DATA_CMD (Command set register) TCmini→Robot

D46*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value															0	0x E311

Register	Item	Setting value
D460	Command (Hexadecimal format)	0xE311
D461	Task	0 (Always specify 0)
D462 to D46F	Not used	0

DATA_RESP (Command response register) Robot→TCmini

D66*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
SCARA			World T-coordinate current position		World C-coordinate current position		World Z-coordinate current position		World Y-coordinate current position		World X-coordinate current position		Orientation		0	0x E311
6-axis	World C-coordinate current position		World B-coordinate current position		World A-coordinate current position											

Register	Item	Response value
D660	Command (Hexadecimal format)	0xE311
D661	Task	0 (Always 0)
D662	Not used	0
D663	Orientation (CONFIG)	0:FREE/1:LEFTY/2:RIGHTY (SCARA) 0 to 22222 (6-axis)*
D664, D665	World X-coordinate current position	Value of world X-coordinate current position×1000
D666, D667	World Y-coordinate current position	Value of world Y-coordinate current position×1000
D668, D669	World Z-coordinate current position	Value of world Z-coordinate current position×1000
D66A, D66B	World C-coordinate current position (SCARA) World A-coordinate current position (6-axis)	Value of world C-coordinate current position×1000 (SCARA) Value of world A-coordinate current position×1000 (6-axis)
D66C, D66D	World T-coordinate current position (SCARA) World B-coordinate current position (6-axis)	Value of world T-coordinate current position×1000 (SCARA) Value of world B-coordinate current position×1000 (6-axis)
D66E, D66F	Not used (SCARA) World C-coordinate current position (6-axis)	0 (SCARA) Value of world C-coordinate current position×1000 (6-axis)

* For details on the 6-axis robot orientation (CONFIG) notation, see section 12.3.

Workpiece coordinate current position acquire command (Monitor mode): E312

DATA_CMD (Command set register) TCmini→Robot

D46*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value															0	0x E312

Register	Item	Setting value
D460	Command (Hexadecimal format)	0xE312
D461	Task	0 (Always specify 0)
D462 to D46F	Not used	0

DATA_RESP (Command response register) Robot→TCmini

D66*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
SCARA			Workpiece T-coordinate current position		Workpiece C-coordinate current position		Workpiece Z-coordinate current position		Workpiece Y-coordinate current position		Workpiece X-coordinate current position		Orientation		0	0x E312
6-axis	Workpiece C-coordinate current position		Workpiece B-coordinate current position		Workpiece A-coordinate current position											

Register	Item	Response value
D660	Command (Hexadecimal format)	0xE312
D661	Task	0 (Always 0)
D662	Not used	0
D663	Orientation (CONFIG)	0:FREE/1:LEFTY/2:RIGHTY (SCARA) 0 to 22222 (6-axis)*
D664, D665	Workpiece X-coordinate current position	Value of workpiece X-coordinate current position×1000
D666, D667	Workpiece Y-coordinate current position	Value of workpiece Y-coordinate current position×1000
D668, D669	Workpiece Z-coordinate current position	Value of workpiece Z-coordinate current position×1000
D66A, D66B	Workpiece C-coordinate current position (SCARA) Workpiece A-coordinate current position (6-axis)	Value of workpiece C-coordinate current position×1000 (SCARA) Value of workpiece A-coordinate current position×1000 (6-axis)
D66C, D66D	Workpiece T-coordinate current position (SCARA) Workpiece B-coordinate current position (6-axis)	Value of workpiece T-coordinate current position×1000 (SCARA) Value of workpiece B-coordinate current position×1000 (6-axis)
D66E, D66F	Not used (SCARA) Workpiece C-coordinate current position (6-axis)	0 (SCARA) Value of workpiece C-coordinate current position×1000 (6-axis)

* For details on the 6-axis robot orientation (CONFIG) notation, see section 12.3.

12.2 Command Mode

In command mode, a command is set to DATA_CMD (Command set register), and when DATA_TRIG (Request trigger signal) is turned on, the robot sends a response to this command. (1-shot command)
Compared to monitor mode, command mode enables the execution of data writing and other complex commands.

12.2.1 Command Mode Register

DATA_TRIG (Request trigger signal) G10F
This is the trigger signal of the Simple PLC data communication function. After DATA_CMD is set, set this signal to ON.
Also, after the response (DATA_RESP) is read, set this signal to OFF.

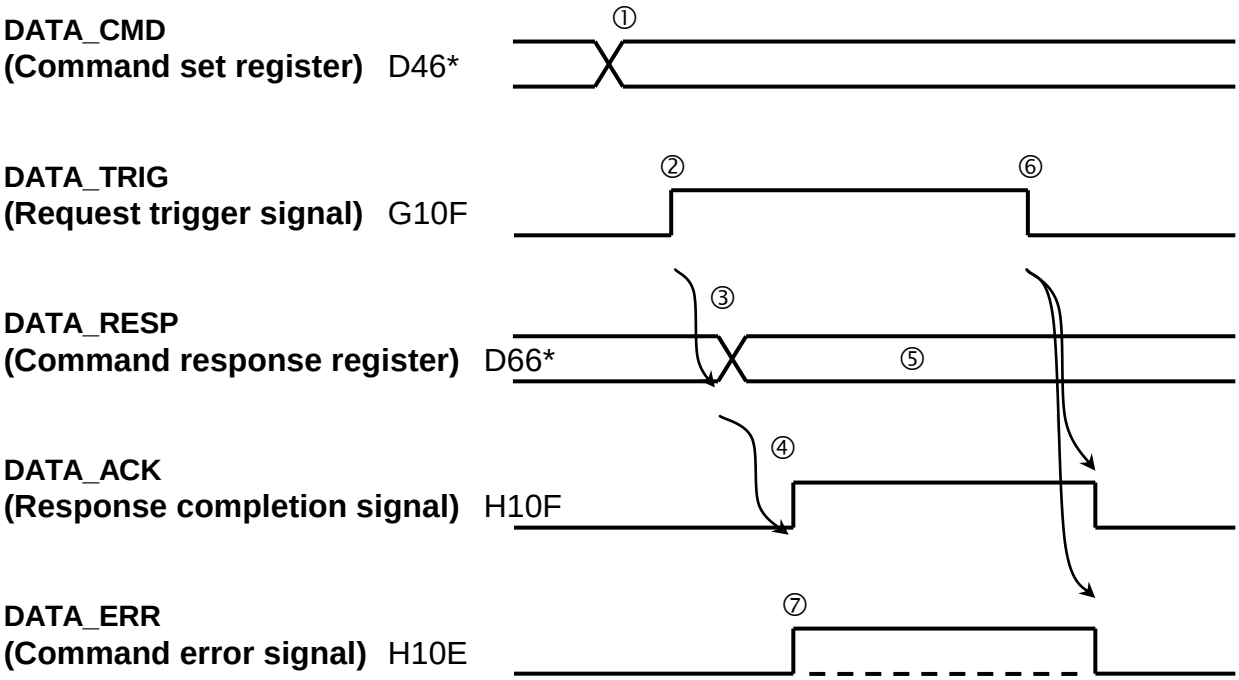
DATA_ACK (Response completion signal) H10F
This is the response completion signal of the Simple PLC data communication function. When setting of the DATA_ RESP data is completed, this signal turns on.
When DATA_TRIG is set to OFF, this signal is also set to OFF.

DATA_ERR (Command error signal) H10E
This is the command error signal of the Simple PLC data communication function. If an error is found in the value that was set to DATA_CMD, this signal is set to ON. When DATA_TRIG is set to OFF, this signal is also set to OFF.

DATA_CMD (Command set register) D460 to D46F
This is the command register of the Simple PLC data communication function. Set a command corresponding to the data that you want to obtain.

DATA_RESP (Command response register) D660 to D66F
This is the command response register of the Simple PLC data communication function. The response to the command that was set is saved.

12.2.2 Command Mode Operation Sequence



① Set the command to DATA_CMD (Command set register).
② When setting of DATA_CMD is completed, set DATA_TRIG (Request trigger signal) to ON.
③ The controller saves the response to the command in DATA_RESP (Command response register).
④ When data saving to DATA_RESP is completed, the controller turns on the DATA_ACK signal.
⑤ Check that the DATA_ACK signal is turned on, and read the response data.
⑥ After the reading process is completed, set the request trigger signal to OFF.
⑦ If an error occurred in the written command data, the command error signal is set to ON. (An error code is saved to the command response register when the command data is normal, but the command could not be executed due to the controller state.)

12.2.3 Command Mode Command List

Joint coordinate current position acquire command (Command mode): E300

DATA_CMD (Command set register) TCmini→Robot

D46*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value															0	0x E300

Register	Item	Setting value
D460	Command (Hexadecimal format)	0xE300
D461	Task	0 (Always specify 0)
D462 to D46F	Not used	0

DATA_RESP (Command response register) Robot→TCmini

D66*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
SCARA			5-axis joint coordinate current position		4-axis joint coordinate current position		3-axis joint coordinate current position		2-axis joint coordinate current position		1-axis joint coordinate current position		Orientation	Error code	0	0x E300
6-axis	6-axis joint coordinate current position															

Register	Item	Response value
D660	Command (Hexadecimal format)	0xE300
D661	Task	0 (Always 0)
D662	Error code (Hexadecimal format)	0x0000: Normal operation 0x1000: Command error (H10E is ON)
D663	Orientation (CONFIG)	0:FREE/1:LEFTY/2:RIGHTY (SCARA) 0 to 22222 (6-axis)*
D664, D665	1-axis joint coordinate current position	Value of 1-axis joint coordinate current position×1000
D666, D667	2-axis joint coordinate current position	Value of 2-axis joint coordinate current position×1000
D668, D669	3-axis joint coordinate current position	Value of 3-axis joint coordinate current position×1000
D66A, D66B	4-axis joint coordinate current position	Value of 4-axis joint coordinate current position×1000
D66C, D66D	5-axis joint coordinate current position	Value of 5-axis joint coordinate current position×1000
D66E, D66F	Not used (SCARA) 6-axis joint coordinate current position (6-axis)	0 (SCARA) Value of 6-axis joint coordinate current position×1000 (6-axis)

* For details on the 6-axis robot orientation (CONFIG) notation, see section 12.3.

World coordinate current position acquire command (Command mode) : E301

DATA_CMD (Command set register) TCmini→Robot

D46*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value															0	0x E301

Register	Item	Setting value
D460	Command (Hexadecimal format)	0xE301
D461	Task	0 (Always specify 0)
D462 to D46F	Not used	0

DATA_RESP (Command response register) Robot→TCmini

D66*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
SCARA			World T-coordinate current position		World C-coordinate current position		World Z-coordinate current position		World Y-coordinate current position		World X-coordinate current position		Orientation	Error code	0	0x E301
6-axis	World C-coordinate current position		World B-coordinate current position		World A-coordinate current position											

Register	Item	Response value
D660	Command (Hexadecimal format)	0xE301
D661	Task	0 (Always 0)
D662	Error code (Hexadecimal format)	0x0000: Normal operation 0x1000: Command error (H10E is ON)
D663	Orientation (CONFIG)	0:FREE/1:LEFTY/2:RIGHTY (SCARA) 0 to 22222 (6-axis)*
D664, D665	World X-coordinate current position	Value of world X-coordinate current position×1000
D666, D667	World Y-coordinate current position	Value of world Y-coordinate current position×1000
D668, D669	World Z-coordinate current position	Value of world Z-coordinate current position×1000
D66A, D66B	World C-coordinate current position (SCARA) World A-coordinate current position (6-axis)	Value of world C-coordinate current position×1000 (SCARA) Value of world A-coordinate current position×1000 (6-axis)
D66C, D66D	World T-coordinate current position (SCARA) World B-coordinate current position (6-axis)	Value of world T-coordinate current position×1000 (SCARA) Value of world B-coordinate current position×1000 (6-axis)
D66E, D66F	Not used (SCARA) World C-coordinate current position (6-axis)	0 (SCARA) Value of world C-coordinate current position×1000 (6-axis)

*For details on the 6-axis robot orientation (CONFIG) notation, see section 12.3.

Workpiece coordinate current position acquire command (Command mode): E302

DATA_CMD (Command set register) TCmini→Robot

D46*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value															0	0x E302

Register	Item	Setting value
D460	Command (Hexadecimal format)	0xE302
D461	Task	0 (Always specify 0)
D462 to D46F	Not used	0

DATA_RESP (Command response register) Robot→TCmini

D66*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
SCARA			Workpiece T-coordinate current position		Workpiece C-coordinate current position		Workpiece Z-coordinate current position		Workpiece Y-coordinate current position		Workpiece X-coordinate current position		Orienta - tion	Error code	0	0x E302
6-axis	Workpiece C-coordinate current position		Workpiece B-coordinate current position		Workpiece A-coordinate current position											

Register	Item	Response value
D660	Command (Hexadecimal format)	0xE302
D661	Task	0 (Always 0)
D662	Error code (Hexadecimal format)	0x0000: Normal operation 0x1000: Command error (H10E is ON)
D663	Orientation (CONFIG)	0:FREE/1:LEFTY/2:RIGHTY (SCARA) 0 to 22222 (6-axis)*
D664, D665	Workpiece X-coordinate current position	Value of workpiece X-coordinate current position×1000
D666, D667	Workpiece Y-coordinate current position	Value of workpiece Y-coordinate current position×1000
D668, D669	Workpiece Z-coordinate current position	Value of workpiece Z-coordinate current position×1000
D66A, D66B	Workpiece C-coordinate current position (SCARA) Workpiece A-coordinate current position (6-axis)	Value of workpiece C-coordinate current position×1000 (SCARA) Value of workpiece A-coordinate current position×1000 (6-axis)
D66C, D66D	Workpiece T-coordinate current position (SCARA) Workpiece B-coordinate current position (6-axis)	Value of workpiece T-coordinate current position×1000 (SCARA) Value of workpiece B-coordinate current position×1000 (6-axis)
D66E, D66F	Not used (SCARA) Workpiece C-coordinate current position (6-axis)	0 (SCARA) Value of workpiece C-coordinate current position×1000 (6-axis)

* For details on the 6-axis robot orientation (CONFIG) notation, see section 12.3.

Teaching point write command: C2C1

DATA_CMD (Command set register) TCmini→Robot

D46*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

SCARA			Teaching point data T-coordinate value	Teaching point data C-coordinate value	Teaching point data Z-coordinate value	Teaching point data Y-coordinate value	Teaching point data X-coordinate value	Orientation	0	Table number	0x C2C1
6-axis	Teaching point data C-coordinate value		Teaching point data B-coordinate value	Teaching point data A-coordinate value							

Register	Item	Setting value
D460	Command (Hexadecimal format)	0xC2C1
D461	Table number	1 to 999(Specify 1 when the teaching data name is "P001")*1
D462	Task	0 (Always specify 0)
D463	Orientation (CONFIG)(Hexadecimal format)	0:FREE/1:LEFTY/2:RIGHTY (SCARA) 0 to 22222 (6-axis)*2
D464, D465	Teaching point data X-coordinate value	Value of teaching point data X-coordinate value×1000
D466, D467	Teaching point data Y-coordinate value	Value of teaching point data Y-coordinate value×1000
D468, D469	Teaching point data Z-coordinate value	Value of teaching point data Z-coordinate value×1000
D46A, D46B	Teaching point data C-coordinate value (SCARA) Teaching point data A-coordinate value (6-axis)	Value of teaching point data C-coordinate value×1000 (SCARA) Value of teaching point data A-coordinate value×1000 (6-axis)
D46C, D46D	Teaching point data T-coordinate value (SCARA) Teaching point data B-coordinate value (6-axis)	Value of teaching point data T-coordinate value×1000 (SCARA) Value of teaching point data B-coordinate value×1000 (6-axis)
D46E, D46F	Not used (SCARA) Teaching point data C-coordinate value (6-axis)	0 (SCARA) Value of teaching point data C-coordinate value×1000 (6-axis)

*1 The table number indicates the number of the teaching point data in the file selected as the execution file.
Teaching point data names must be created in the range from P001 to P999 in the file selected as the execution file and registered as teaching point data.

*2 For details on the 6-axis robot orientation (CONFIG) notation, see section 12.3.

DATA_RESP (Command response register) Robot→TCmini

D66*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value													0	Error code	Table number	0x C2C1

Register	Item	Response value
D660	Command (Hexadecimal format)	0xC2C1
D661	Table number	1 to 999
D662	Error code (Hexadecimal format)	0x0000: Normal operation 0x1000: Command error (H10E is ON) 0x2001: Program execution in progress 0x2002: Outside teaching point name range (1 to 999) 0x2003: Teaching point name does not exist 0x2004: Teaching point write failed
D663	Task	0 (Always 0)
D664 to D66F	Not used	0

Teaching point data acquire command: C3C1

DATA_CMD (Command set register) TCmini→Robot

D46*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
Value														0	Table number	0x C3C1

Register	Item	Setting value
D460	Command (Hexadecimal format)	0xC3C1
D461	Table number	1 to 999 (Specify 1 when the teaching point data name is "P001")*1
D462	Task	0 (Always specify 0)
D463 to D46F	Not used	0

*1 The table number indicates the number of the teaching point data in the file selected as the execution file.
Teaching point data names must be created in the range from P001 to P999 in the file selected as the execution file and registered as teaching point data.
If a teaching point name does not exist, 0 is returned for the teaching point data.

DATA_RESP (Command response register) Robot→TCmini

D66*	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
SCARA			Teaching point data T-coordinate value		Teaching point data C-coordinate value		Teaching point data Z-coordinate value		Teaching point data Y-coordinate value		Teaching point data X-coordinate value		Orientation	Error code	Table number	0x C3C1
6-axis	Teaching point data C-coordinate value		Teaching point data B-coordinate value		Teaching point data A-coordinate value											

Register	Item	Response value
D660	Command (Hexadecimal format)	0xC3C1
D661	Table number	1 to 999
D662	Error code (Hexadecimal format)	0x0000: Normal operation 0x1000: Command error (H10E is ON) 0x2002: Outside teaching point name range (1 to 999)
D663	Orientation (CONFIG)	0:FREE/1:LEFTY/2:RIGHTY (SCARA) 0 to 22222 (6-axis)*2
D664, D665	Teaching point data X-coordinate value	Value of teaching point data X-coordinate value×1000
D666, D667	Teaching point data Y-coordinate value	Value of teaching point data Y-coordinate value×1000
D668, D669	Teaching point data Z-coordinate value	Value of teaching point data Z-coordinate value×1000
D66A, D66B	Teaching point data C-coordinate value (SCARA) Teaching point data A-coordinate value (6-axis)	Value of teaching point data C-coordinate value×1000 (SCARA) Value of teaching point data A-coordinate value×1000 (6-axis)
D66C, D66D	Teaching point data T-coordinate value (SCARA) Teaching point data B-coordinate value (6-axis)	Value of teaching point data T-coordinate value×1000 (SCARA) Value of teaching point data B-coordinate value×1000 (6-axis)
D66E, D66F	Not used (SCARA) Teaching point data C-coordinate value (6-axis)	0 (SCARA) Value of teaching point data C-coordinate value×1000 (6-axis)

*2 For details on the 6-axis robot orientation (CONFIG) notation, see section 12.3.

12.3 6-axis Robot Orientation (CONFIG) Notation

The 6-axis robot orientation (CONFIG) notation appears as "0", "1102", "21111", or similar based on the integer values shown in the table below.

Shoulder orientation	Elbow orientation	Wrist orientation	4-axis (Rot4)	6-axis (Rot6)
0:FREE(x) 1:LEFTY(L) 2:RIGHTY(R)	0:FREE(x) 1:ABOVE(A) 2:BELOW(B)	0:FREE(x) 1:NFLIP(N) 2:FLIP(F)	0:FREE(x) 1:SINGLE4(S) 2:DOUBLE4(D)	0:FREE(x) 1:SINGLE6(S) 2:DOUBLE6(D)

Examples:

0 indicates FREE / FREE / FREE / FREE / FREE (xxxxx).

1102 indicates FREE / ABOVE / FLIP / FREE / DOUBLE6 (xAFxD).

21111 indicates LEFTY / ABOVE / NFLIP / SINGLE4 / SINGLE6 (LANSS).

22010 indicates RIGHTY / BELOW / FREE / SINGLE4 / FREE (RBxSx).

22222 indicates RIGHTY / BELOW / FLIP / DOUBLE4 / DOUBLE6 (RBFDD).

Section 13 Appendix

List of Relays

I/O

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
X00W	IN16	IN15	IN14	IN13	IN12	IN11	IN10	IN9	IN8	IN7	IN6	IN5	IN4	IN3	IN2	IN1
X01W	IN32	IN31	IN30	IN29	IN28	IN27	IN26	IN25	IN24	IN23	IN22	IN21	IN20	IN19	IN18	IN17
X02W	IN48	IN47	IN46	IN45	IN44	IN43	IN42	IN41	IN40	IN39	IN38	IN37	IN36	IN35	UFI2	UFI1
X03W	HI8	HI7	HI6	HI5	HI4	HI3	HI2	HI1	LI8	LI7	LI6	LI5	LI4	LI3	LI2	LI1
X04W	EI16	EI15	EI14	EI13	EI12	EI11	EI10	EI9	EI8	EI7	EI6	EI5	EI4	EI3	EI2	EI1
X05W	EI32	EI31	EI30	EI29	EI28	EI27	EI26	EI25	EI24	EI23	EI22	EI21	EI20	EI19	EI18	EI17
X06W	EI48	EI47	EI46	EI45	EI44	EI43	EI42	EI41	EI40	EI39	EI38	EI37	EI36	EI35	EI34	EI33
X07W	EI64	EI63	EI62	EI61	EI60	EI59	EI58	EI57	EI56	EI55	EI54	EI53	EI52	EI51	EI50	EI49
Y10W	OUT16	OUT15	OUT14	OUT13	OUT12	OUT11	OUT10	OUT9	OUT8	OUT7	OUT6	OUT5	OUT4	OUT3	OUT2	OUT1
Y11W	OUT32	OUT31	OUT30	OUT29	OUT28	OUT27	OUT26	OUT25	OUT24	OUT23	OUT22	OUT21	OUT20	OUT19	OUT18	OUT17
Y12W	OUT48	OUT47	OUT46	OUT45	OUT44	OUT43	OUT42	OUT41	OUT40	OUT39	OUT38	OUT37	OUT36	OUT35	UFO2	UFO1
Y13W	HO8	HO7	HO6	HO5	HO4	HO3	HO2	HO1								
Y14W	EO16	EO15	EO14	EO13	EO12	EO11	EO10	EO9	EO8	EO7	EO6	EO5	EO4	EO3	EO2	EO1
Y15W	EO32	EO31	EO30	EO29	EO28	EO27	EO26	EO25	EO24	EO23	EO22	EO21	EO20	EO19	EO18	EO17
Y16W	EO48	EO47	EO46	EO45	EO44	EO43	EO42	EO41	EO40	EO39	EO38	EO37	EO36	EO35	EO34	EO33
Y17W	EO64	EO63	EO62	EO61	EO60	EO59	EO58	EO57	EO56	EO55	EO54	EO53	EO52	EO51	EO50	EO49
X20W	FI16	FI15	FI14	FI13	FI12	FI11	FI10	FI9	FI8	FI7	FI6	FI5	FI4	FI3	FI2	FI1
X21W	FI32	FI31	FI30	FI29	FI28	FI27	FI26	FI25	FI24	FI23	FI22	FI21	FI20	FI19	FI18	FI17
X22W	FI48	FI47	FI46	FI45	FI44	FI43	FI42	FI41	FI40	FI39	FI38	FI37	FI36	FI35	FI34	FI33
X23W	FI64	FI63	FI62	FI61	FI60	FI59	FI58	FI57	FI56	FI55	FI54	FI53	FI52	FI51	FI50	FI49
X24W	FI80	FI79	FI78	FI77	FI76	FI75	FI74	FI73	FI72	FI71	FI70	FI69	FI68	FI67	FI66	FI65
X25W	FI96	FI95	FI94	FI93	FI92	FI91	FI90	FI89	FI88	FI87	FI86	FI85	FI84	FI83	FI82	FI81
X26W	FI112	FI111	FI110	FI109	FI108	FI107	FI106	FI105	FI104	FI103	FI102	FI101	FI100	FI99	FI98	FI97
X27W	FI128	FI127	FI126	FI125	FI124	FI123	FI122	FI121	FI120	FI119	FI118	FI117	FI116	FI115	FI114	FI113
Y30W	FO16	FO15	FO14	FO13	FO12	FO11	FO10	FO9	FO8	FO7	FO6	FO5	FO4	FO3	FO2	FO1
Y31W	FO32	FO31	FO30	FO29	FO28	FO27	FO26	FO25	FO24	FO23	FO22	FO21	FO20	FO19	FO18	FO17
Y32W	FO48	FO47	FO46	FO45	FO44	FO43	FO42	FO41	FO40	FO39	FO38	FO37	FO36	FO35	FO34	FO33
Y33W	FO64	FO63	FO62	FO61	FO60	FO59	FO58	FO57	FO56	FO55	FO54	FO53	FO52	FO51	FO50	FO49
Y34W	FO80	FO79	FO78	FO77	FO76	FO75	FO74	FO73	FO72	FO71	FO70	FO69	FO68	FO67	FO66	FO65
Y35W	FO96	FO95	FO94	FO93	FO92	FO91	FO90	FO89	FO88	FO87	FO86	FO85	FO84	FO83	FO82	FO81
Y36W	FO112	FO111	FO110	FO109	FO108	FO107	FO106	FO105	FO104	FO103	FO102	FO101	FO100	FO99	FO98	FO97
Y37W	FO128	FO127	FO126	FO125	FO124	FO123	FO122	FO121	FO120	FO119	FO118	FO117	FO116	FO115	FO114	FO113

Internal relays

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
R00W																
R01W																
R02W																
R03W																
R04W																
R05W																
R06W																
R07W																
R10W																
R11W																
R12W																
R13W																
R14W																
R15W																
R16W																
R17W																
R20W																
R21W																
R22W																
R23W																
R24W																
R25W																
R26W																
R27W																
R30W																
R31W																
R32W																
R33W																
R34W																
R35W																
R36W																
R37W																
R40W																
R41W																
R42W																
R43W																
R44W																
R45W																
R46W																
R47W																
R50W																
R51W																
R52W																
R53W																
R54W																
R55W																
R56W																
R57W																
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Internal relays

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
R60W																
R61W																
R62W																
R63W																
R64W																
R65W																
R66W																
R67W																
R70W																
R71W																
R72W																
R73W																
R74W																
R75W																
R76W																
R77W																

TCmini → Main unit of robot controller (G000 ~ G27F)

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
G00W	DIN 16	DIN 15	DIN 14	DIN 13	DIN 12	DIN 11	DIN 10	DIN 9	DIN 8	DIN 7	DIN 6	DIN 5	DIN 4	DIN 3	DIN 2	DIN 1
G01W	DIN 32	DIN 31	DIN 30	DIN 29	DIN 28	DIN 27	DIN 26	DIN 25	DIN 24	DIN 23	DIN 22	DIN 21	DIN 20	DIN 19	DIN 18	DIN 17
G02W	DIN 48	DIN 47	DIN 46	DIN 45	DIN 44	DIN 43	DIN 42	DIN 41	DIN 40	DIN 39	DIN 38	DIN 37	DIN 36	DIN 35	DIN 34	DIN 33
G03W	DIN 64	DIN 63	DIN 62	DIN 61	DIN 60	DIN 59	DIN 58	DIN 57	DIN 56	DIN 55	DIN 54	DIN 53	DIN 52	DIN 51	DIN 50	DIN 49
G04W	DIN 116	DIN 115	DIN 114	DIN 113	DIN 112	DIN 111	DIN 110	DIN 109	DIN 108	DIN 107	DIN 106	DIN 105	DIN 104	DIN 103	DIN 102	DIN 101
G05W	DIN 132	DIN 131	DIN 130	DIN 129	DIN 128	DIN 127	DIN 126	DIN 125	DIN 124	DIN 123	DIN 122	DIN 121	DIN 120	DIN 119	DIN 118	DIN 117
G06W	DIN 148	DIN 147	DIN 146	DIN 145	DIN 144	DIN 143	DIN 142	DIN 141	DIN 140	DIN 139	DIN 138	DIN 137	DIN 136	DIN 135	DIN 134	DIN 133
G07W	DIN 164	DIN 163	DIN 162	DIN 161	DIN 160	DIN 159	DIN 158	DIN 157	DIN 156	DIN 155	DIN 154	DIN 153	DIN 152	DIN 151	DIN 150	DIN 149
G10W	DATA _TRIG				DCON V TRIG	LMIT OFF	MLT RST	OFS MOD	HAND IN8	HAND IN7	HAND IN6	HAND IN5	HAND IN4	HAND IN3	HAND IN2	HAND IN1
G11W	AL8-272	AL8-271	AL8-270	AL8-269	AL4-080	AL4-079	AL4-078	AL4-077	AL1-044	AL1-043	AL1-042	AL1-041	AL1-040	AL1-039	AL1-038	AL1-037
G12W					J_ MOV E	J_DI RECT	J_AXIS				J_COORD		J_SPEED		J_REMOTE	
G13W			BZ_ RST	SV OFF	BREAK	LOW_ SPD	CYCLE	STOP	EX_ SVON	RUN	ALM _RST	DO _RST	CYC _RST	STEP _RST	PRG_ RST	STROBE
G14W ~ G16W	Reserved area															
G17W	PANEL BREAK	PANEL RUN	PANEL SV OFF	PANEL EX_ SVON		PANEL DO _RST	PANEL PRG_ RST	PANEL ALM _RST								
G20W	DIN 316	DIN 315	DIN 314	DIN 313	DIN 312	DIN 311	DIN 310	DIN 309	DIN 308	DIN 307	DIN 306	DIN 305	DIN 304	DIN 303	DIN 302	DIN 301
G21W	DIN 332	DIN 331	DIN 330	DIN 329	DIN 328	DIN 327	DIN 326	DIN 325	DIN 324	DIN 323	DIN 322	DIN 321	DIN 320	DIN 319	DIN 318	DIN 317
G22W	DIN 348	DIN 347	DIN 346	DIN 345	DIN 344	DIN 343	DIN 342	DIN 341	DIN 340	DIN 339	DIN 338	DIN 337	DIN 336	DIN 335	DIN 334	DIN 333
G23W	DIN 364	DIN 363	DIN 362	DIN 361	DIN 360	DIN 359	DIN 358	DIN 357	DIN 356	DIN 355	DIN 354	DIN 353	DIN 352	DIN 351	DIN 350	DIN 349
G24W	DIN 416	DIN 415	DIN 414	DIN 413	DIN 412	DIN 411	DIN 410	DIN 409	DIN 408	DIN 407	DIN 406	DIN 405	DIN 404	DIN 403	DIN 402	DIN 401
G25W	DIN 432	DIN 431	DIN 430	DIN 429	DIN 428	DIN 427	DIN 426	DIN 425	DIN 424	DIN 423	DIN 422	DIN 421	DIN 420	DIN 419	DIN 418	DIN 417
G26W	DIN 448	DIN 447	DIN 446	DIN 445	DIN 444	DIN 443	DIN 442	DIN 441	DIN 440	DIN 439	DIN 438	DIN 437	DIN 436	DIN 435	DIN 434	DIN 433
G27W	DIN 464	DIN 463	DIN 462	DIN 461	DIN 460	DIN 459	DIN 458	DIN 457	DIN 456	DIN 455	DIN 454	DIN 453	DIN 452	DIN 451	DIN 450	DIN 449
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Note: G108~ G10B are optional signals for specific customers.

Main unit of robot controller → TCmini (H000 ~ H27F)

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
H00W	DOUT 16	DOUT 15	DOUT 14	DOUT 13	DOUT 12	DOUT 11	DOUT 10	DOUT 9	DOUT 8	DOUT 7	DOUT 6	DOUT 5	DOUT 4	DOUT 3	DOUT 2	DOUT 1
H01W	DOUT 32	DOUT 31	DOUT 30	DOUT 29	DOUT 28	DOUT 27	DOUT 26	DOUT 25	DOUT 24	DOUT 23	DOUT 22	DOUT 21	DOUT 20	DOUT 19	DOUT 18	DOUT 17
H02W	DOUT 48	DOUT 47	DOUT 46	DOUT 45	DOUT 44	DOUT 43	DOUT 42	DOUT 41	DOUT 40	DOUT 39	DOUT 38	DOUT 37	DOUT 36	DOUT 35	DOUT 34	DOUT 33
H03W	DOUT 64	DOUT 63	DOUT 62	DOUT 61	DOUT 60	DOUT 59	DOUT 58	DOUT 57	DOUT 56	DOUT 55	DOUT 54	DOUT 53	DOUT 52	DOUT 51	DOUT 50	DOUT 49
H04W	DOUT 116	DOUT 115	DOUT 114	DOUT 113	DOUT 112	DOUT 111	DOUT 110	DOUT 109	DOUT 108	DOUT 107	DOUT 106	DOUT 105	DOUT 104	DOUT 103	DOUT 102	DOUT 101
H05W	DOUT 132	DOUT 131	DOUT 130	DOUT 129	DOUT 128	DOUT 127	DOUT 126	DOUT 125	DOUT 124	DOUT 123	DOUT 122	DOUT 121	DOUT 120	DOUT 119	DOUT 118	DOUT 117
H06W	DOUT 148	DOUT 147	DOUT 146	DOUT 145	OUT 144	DOUT 143	DOUT 142	DOUT 141	DOUT 140	DOUT 139	DOUT 138	DOUT 137	DOUT 136	DOUT 135	DOUT 134	DOUT 133
H07W	DOUT 164	DOUT 163	DOUT 162	DOUT 161	DOUT 160	DOUT 159	DOUT 158	DOUT 157	DOUT 156	DOUT 155	DOUT 154	DOUT 153	DOUT 152	DOUT 151	DOUT 150	DOUT 149
H10W	DATA _ACK	DATA _ERR				TCP ERR	MLT END	OFS END	HAND OUT8	HAND OUT7	HAND OUT6	HAND OUT5	HAND OUT4	HAND OUT3	HAND OUT2	HAND OUT1
H11W	SEQ FSW8	SEQ FSW7	SEQ FSW6	SEQ FSW5	SEQ FSW4	SEQ FSW3	SEQ FSW2	SEQ FSW1	SEQ PAR8	SEQ PAR7	SEQ PAR6	SEQ PAR5	SEQ PAR4	SEQ PAR3	SEQ PAR2	SEQ PAR1
H12W																
H13W		EXT ETHER	ALARM	BT_ ALM	CYC _ST	LOW _ST	CYC _END	AUTO RUN	SYS_ RDY	EXT 232C	EXT SIG	INT	TEACH	ACK	SV_ RDY	EMG_ ST
H14W ~ H17W	Reserved area															
H20W	DOUT 316	DOUT 315	DOUT 314	DOUT 313	DOUT 312	DOUT 311	DOUT 310	DOUT 309	DOUT 308	DOUT 307	DOUT 306	DOUT 305	DOUT 304	DOUT 303	DOUT 302	DOUT 301
H21W	DOUT 332	DOUT 331	DOUT 330	DOUT 329	DOUT 328	DOUT 327	DOUT 326	DOUT 325	DOUT 324	DOUT 323	DOUT 322	DOUT 321	DOUT 320	DOUT 319	DOUT 318	DOUT 317
H22W	DOUT 348	DOUT 347	DOUT 346	DOUT 345	OUT 344	DOUT 343	DOUT 342	DOUT 341	DOUT 340	DOUT 339	DOUT 338	DOUT 337	DOUT 336	DOUT 335	DOUT 334	DOUT 333
H23W	DOUT 364	DOUT 363	DOUT 362	DOUT 361	DOUT 360	DOUT 359	DOUT 358	DOUT 357	DOUT 356	DOUT 355	DOUT 354	DOUT 353	DOUT 352	DOUT 351	DOUT 350	DOUT 349
H24W	DOUT 416	DOUT 415	DOUT 414	DOUT 413	DOUT 412	DOUT 411	DOUT 410	DOUT 409	DOUT 408	DOUT 407	DOUT 406	DOUT 405	DOUT 404	DOUT 403	DOUT 402	DOUT 401
H25W	DOUT 432	DOUT 431	DOUT 430	DOUT 429	DOUT 428	DOUT 427	DOUT 426	DOUT 425	DOUT 424	DOUT 423	DOUT 422	DOUT 421	DOUT 420	DOUT 419	DOUT 418	DOUT 417
H26W	DOUT 448	DOUT 447	DOUT 446	DOUT 445	DOUT 444	DOUT 443	DOUT 442	DOUT 441	DOUT 440	DOUT 439	DOUT 438	DOUT 437	DOUT 436	DOUT 435	DOUT 434	DOUT 433
H27W	DOUT 464	DOUT 463	DOUT 462	DOUT 461	DOUT 460	DOUT 459	DOUT 458	DOUT 457	DOUT 456	DOUT 455	DOUT 454	DOUT 453	DOUT 452	DOUT 451	DOUT 450	DOUT 449
Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Note: H108~ H10A are optional signals for specific customers.

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
E00W																
E01W																
E02W																
E03W																
E04W																
E05W																
E06W																
E07W					Edge relays											
E10W																
E11W																
E12W																
E13W																
E14W																
E15W																
E16W																
E17W																
L00W																
L01W																
L02W																
L03W					Latch relays											
L04W																
L05W																
L06W																
L07W																
T/C00W																
T/C01W						Timers (100 ms)/counters											
T/C02W																
T/C03W																
T/C04W																
T/C05W																
T/C06W																
T/C07W																
T/C08W																
T/C09W																
T/C10W						Timers (10 ms)/counters											
T/C11W																
T/C12W																
T/C13W																
T/C14W																
T/C15W																
T/C16W																
T/C17W																

Bit	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
T/C20W																
T/C21W																
T/C22W																
T/C23W						Timers (100 ms)/counters											
T/C24W																
T/C25W																
T/C26W																
T/C27W																
A00W									Sign flag	Zero flag				Over-flow		Carry flag
A01W										Fuse flag						
A02W																
A03W	6400 ms	3200 ms	1600 ms	800 ms	400 ms	200 ms	100 ms	50 ms	Scan time (ms)							
A04W	12800 ms	6400 ms	3200 ms	1600 ms	800 ms	400 ms	200 ms	100 ms	1280 ms	640 ms	320 ms	160 ms	80 ms	40 ms	20 ms	10 ms
A05W	1280 s	640 s	320 s	160 s	80 s	40 s	20 s	10 s	128 s	64 s	32 s	16 s	8 s	4 s	2 s	1 s

List of Registers

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D00*																
D01*																
D02*							Data registers									
D03*																
D04*																
D05*																
D06*																
D07*																
D10*																
D11*																
D12*							Data registers									
D13*																
D14*																
D15*																
D16*																
D17*																
D20*																
D21*																
D22*							Data registers									
D23*																
D24*																
D25*																
D26*																
D27*																
D30*																
D31*																
D32*																
D33*							Data registers (backup)									
D34*																
D35*																
D36*																
D37*																

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D40*	PLC SS R08	PLC SS R07	PLC SS R06	PLC SS R05	PLC SS R04	PLC SS R03	PLC SS R02	PLC SS R01	PLC DATA R8	PLC DATA R7	PLC DATA R6	PLC DATA R5	PLC DATA R4	PLC DATA R3	PLC DATA R2	PLC DATA R1
D41*	PLC SL R08H	PLC SL R08L	PLC SL R07H	PLC SL R07L	PLC SL R06H	PLC SL R06L	PLC SL R05H	PLC SL R05L	PLC SL R04H	PLC SL R04L	PLC SL R03H	PLC SL R03L	PLC SL R02H	PLC SL R02L	PLC SL R01H	PLC SL R01L
D42*	Reserved area															
D43*																
D44*	Reserved area															
D45*																
D46*	DATA_CMD															
D47*																USER
D50*	Reserved area															
D51*																
D52*																
D53*																
D54*																
D55*																
D56*																
D57*																
D60*	PLC SS W08	PLC SS W07	PLC SS W06	PLC SS W05	PLC SS W04	PLC SS W03	PLC SS W02	PLC SS W01	PLC DATA W8	PLC DATA W7	PLC DATA W6	PLC DATA W5	PLC DATA W4	PLC DATA W3	PLC DATA W2	PLC DATA W1
D61*	PLC SL W08H	PLC SL W08L	PLC SL W07H	PLC SL W07L	PLC SL W06H	PLC SL W06L	PLC SL W05H	PLC SL W05L	PLC SL W04H	PLC SL W04L	PLC SL W03H	PLC SL W03L	PLC SL W02H	PLC SL W02L	PLC SL W01H	PLC SL W01L
D62*	Reserved area								PSN_ W8	PSN_ W7	PSN_ W6	PSN_ W5	PSN_ W4	PSN_ W3	PSN_ W2	PSN_ W1
D63*									PSN_ J8	PSN_ J7	PSN_ J6	PSN_ J5	PSN_ J4	PSN_ J3	PSN_ J2	PSN_ J1
D64*									TRQ_ J8	TRQ_ J7	TRQ_ J6	TRQ_ J5	TRQ_ J4	TRQ_ J3	TRQ_ J2	TRQ_ J1
D65*																
D66*	DATA_RESP															
D67*	AL10	AL09	AL08	AL07	AL06	AL05	AL04	AL03	AL02	AL01	ALNO	STEP				
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
D70*	FB RWr 16	FB RWr 15	FB RWr 14	FB RWr 13	FB RWr 12	FB RWr 11	FB RWr 10	FB RWr 9	FB RWr 8	FB RWr 7	FB RWr 6	FB RWr 5	FB RWr 4	FB RWr 3	FB RWr 2	FB RWr 1
D71*	Reserved area															
D72*																
D73*																
D74*	FB RWw 16	FB RWw 15	FB RWw 14	FB RWw 13	FB RWw 12	FB RWw 11	FB RWw 10	FB RWw 9	FB RWw 8	FB RWw 7	FB RWw 6	FB RWw 5	FB RWw 4	FB RWw 3	FB RWw 2	FB RWw 1
D75*	Reserved area															
D76*																
D77*																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
P00*																
P01*					Timer current values (100 ms)/counter current values											
P02*																
P03*																
P04*																
P05*																
P06*																
P07*																
P10*																
P11*					Timer current values (10 ms)/counter current values											
P12*																
P13*																
P14*																
P15*																
P16*																
P17*																
P20*																
P21*																
P22*																
P23*					Timer current values (10 ms)/counter current values											
P24*																
P25*																
P26*																
P27*																
Register	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

The information contained in this manual is subject to change without prior notice to effect improvements.