YAMAHA
YAMAHA MOTOR CO., LTD.

YAMAHA SINGLE-AXIS ROBOT CONTROLLER

SRCP

User's Manual

ENGLISH E

E84-Ver. 2.04

General Contents

MEMO

Chapter 1 OVERVIEW

Thank you for purchasing the YAMAHA single-axis robot controller SRCP series (hereafter called "SRCP controller" or simply "SRCP" or "this controller"). This manual describes SRCP controller features and operating procedures.
When used with a YAMAHA single-axis PHASER series robot, the SRCP controller performs positioning and pick-and-place tasks of various mechanical parts and devices.
This first chapter explains basic information you should know before using the SRCP controller such as names and functions of the various parts, steps necessary to prepare the robot for operation, and the architecture of the system itself. Please read this chapter carefully for a basic overview of the SRCP controller.

1-1 Features of the SRCP Series Controller

The SRCP series is a high-performance robot controller using a 32-bit RISC chip CPU.
When used with a YAMAHA single-axis PHASER series robot, the SRCP controller performs positioning tasks of various mechanical parts and devices. The SRCP controller also performs I/O control of solenoid valves and sensors, and controls communication with a PC (personal computer).
Using only one SRCP controller allows configuring a complete system for simple applications such as pick-and-place tasks.

The SRCP series has the following features:

- A high-performance 32-bit RISC chip CPU is used for high-speed, high-precision software servo control.
- Program assets created with the previous SRC, SRCA, ERC, SRCH, ERCX and SRCX series can be used without any modifications.
- Ideal acceleration and deceleration speeds can be obtained by simply entering the number of the robot to control and the payload parameter. No troublesome servo adjustments are required.
- The I/O interface provides 8 input and 5 output points for general-purpose user wiring as a standard feature.
- The TPB programming box (option) allows interactive user operation by simple menus that permit immediate use. The robot can also be operated from a personal computer (PC) just the same as TPB when the POPCOM software (option) is installed in the PC.
- Programs for robot operation can be written with an easy-to-learn robot language that closely resembles BASIC. Even first-time users will find it easy to use.
- Users not accustomed to robot language can use a PLC (programmable logic controller) to directly move the robot by specifying the operation points.
- Users can create programs and control the robot on a personal computer (PC). Communication with the PC is performed with an easy-to-learn robot language similar to BASIC. Even first-time users will find it easy to use.
- A built-in multi-task function allows efficiently creating the programs.
- The I/O interface supports pulse trains to allow position control by input of a pulse train.

NOTE

The SRCP controller can be operated from either a TPB (programming box) or a PC running with communication software such as POPCOM. This user's manual mainly describes operations using the TPB. For details on operation with POPCOM, refer to the POPCOM manual. If you want to use your own methods to operate the SRCP controller from a PC, refer to Chapter 11 "Communications with PC" for pertinent information.

1-2 Setting Up for Operation

The chart below illustrates the basic steps to follow from the time of purchase of this controller until it is ready for use. The chapters of this user's manual are organized according to the operation procedures, and allow first time users to proceed one step at a time.

Basic steps

Operation	Information to be familiar with	Refer to
Installation	• Installing the controller	2-1
↓		
Wiring and connection	• Connecting the power supply	2-2
	• Grounding	2-3
	• Connecting peripheral equipment	2-4 to 2-8
	• Understanding the I/O interface	Chapter 3
↓		
Setting parameters	• Understanding basic TPB operations	Chapter 4
	• Setting the various parameters	Chapter 5
↓		
Programming	• Inputting or editing programs	Chapter 6
	• Editing point data	Chapter 7
	• Robot language	Chapter 8
↓		
Running the robot	• Return-to-origin	Chapter 9
	• Various operation steps	
	• Emergency stop	

1-3 External View and Part Names

This section explains part names of the SRCP controller and TPB along with their functions. Note that the external view and specifications are subject to change without prior notice to the user.

1-3-1 SRCP controller

1. Status Display Lamp
   This lamp indicates the operating status of the robot and controller.
   Refer to "15-1-3 LED display" for information on controller status and the matching LED display.

2. Escape Switch (ESC switch)
   Hold down this switch when connecting or disconnecting the TPB from the SRCP controller. (See "4-1 Connecting and Disconnecting the TPB.")

3. TPB Connector
   This is used to connect the TPB or the RS-232C terminal of a PC (personal computer).

4. COM Connector
   This is used to connect a network system when the optional network card is installed. (This is covered when the option is not in use.)

5. Robot I/O Connector
   Input/output connector for robot peripheral device signals such as position signals.

6. I/O. CN
   This is used to connect external equipment such as a PLC.

7. EXT. CN
   Connector for emergency stop signal input. This connector also supplies 24V power for the I/O devices.

8. Motor Connector
   This is the power line connector for the servo motor.

9. Regenerative Unit Connector (RGEN connector)
   Some types of robots require connection to a regenerative unit. In such cases, use this to connect the regenerative unit (RGU-2).

10. Terminal Block

    ACIN (L, N, ⏚)

    These are terminals for supplying AC power to the SRCP controller. The ground terminal must be properly grounded to prevent electrical shock to the human body and to maintain equipment reliability.

    NC

    No connection. Do not use.

    T1, T2

    These are input power voltage switching terminals. When an input power voltage of AC100 to 115V is used, short the T1 and T2 terminals. When an input power voltage of AC200 to 230V is used, leave the T1 and T2 terminals open. (SRCP-05A, 10A, 20A only)

11. Serial number nameplate

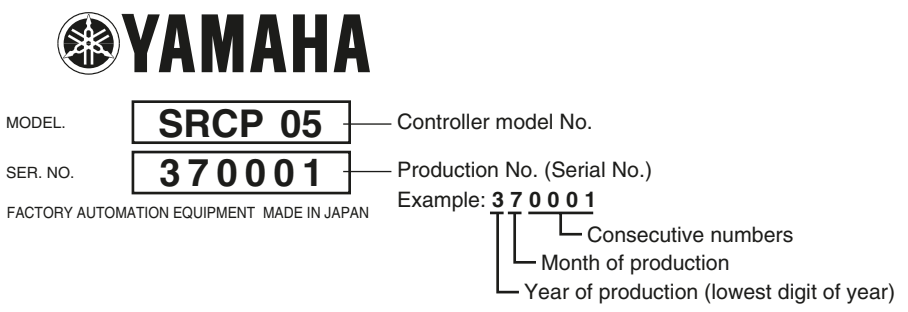
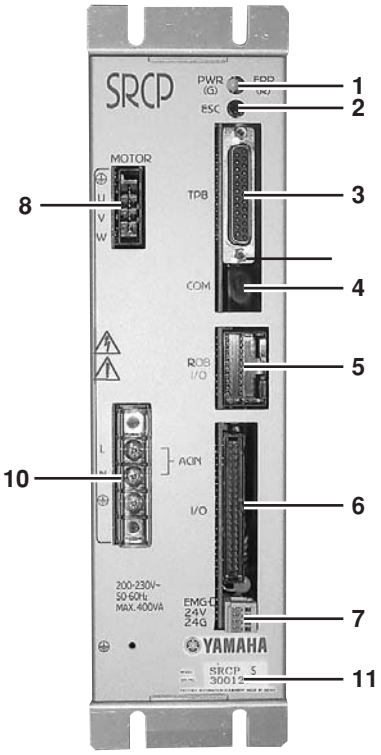
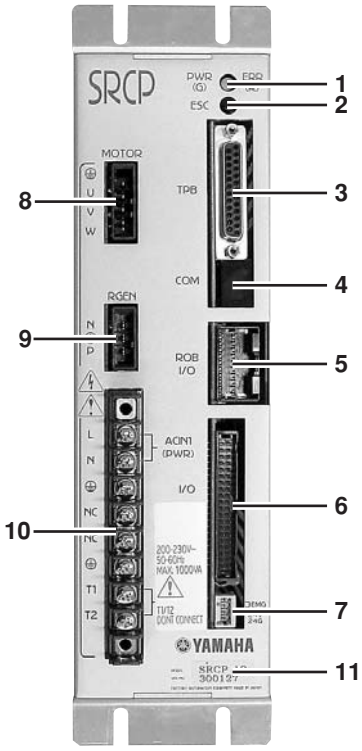


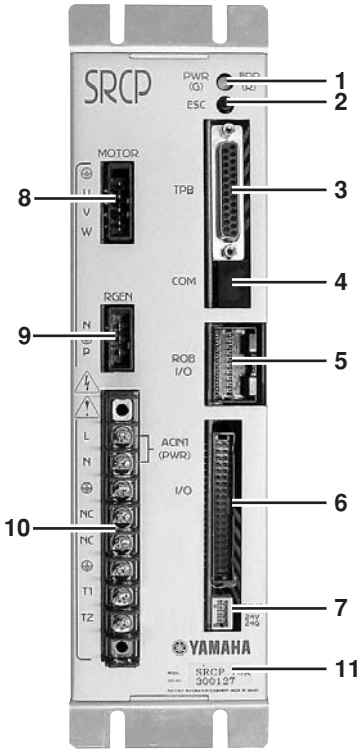
Fig. 1-1 Exterior of the SRCP controller



SRCP-05



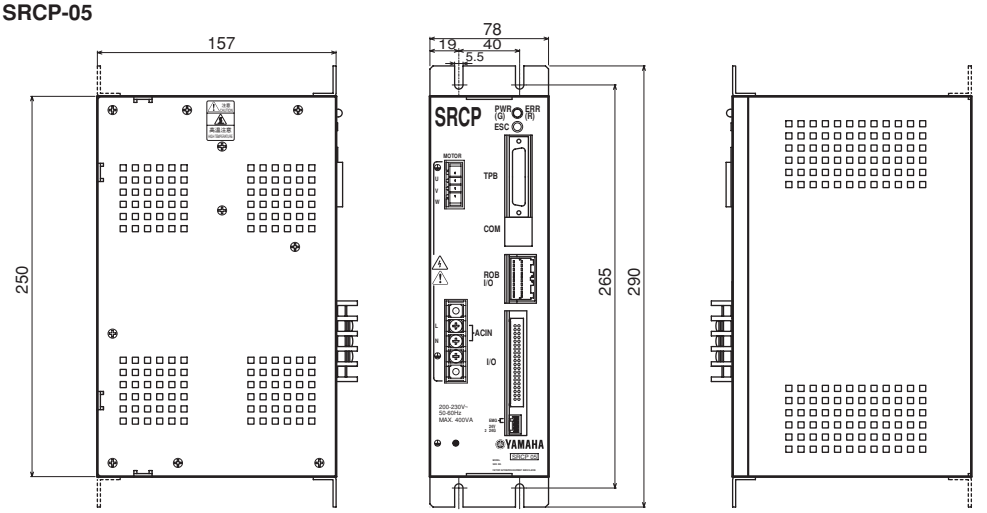
SRCP-10, 20



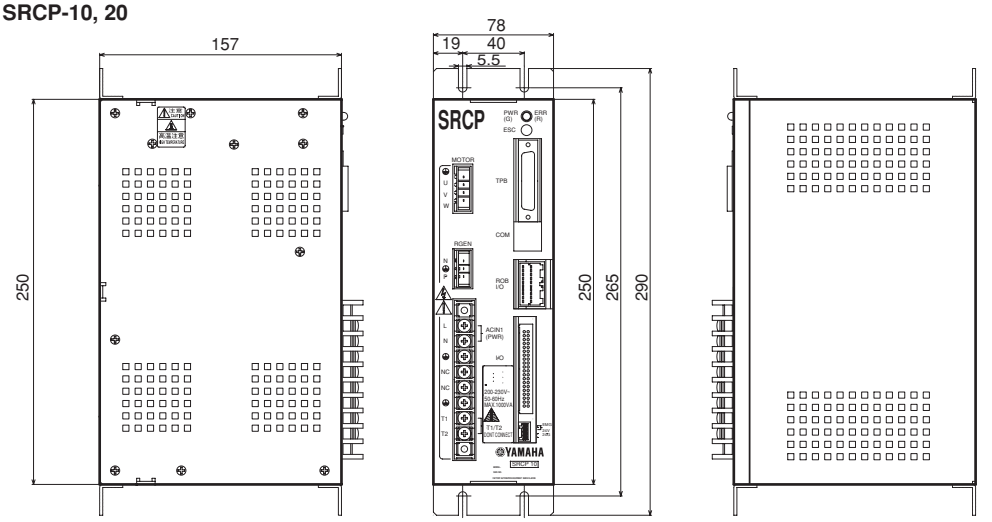
SRCP-05A, 10A, 20A

Fig. 1-2 Three-side view of the SRCP controller

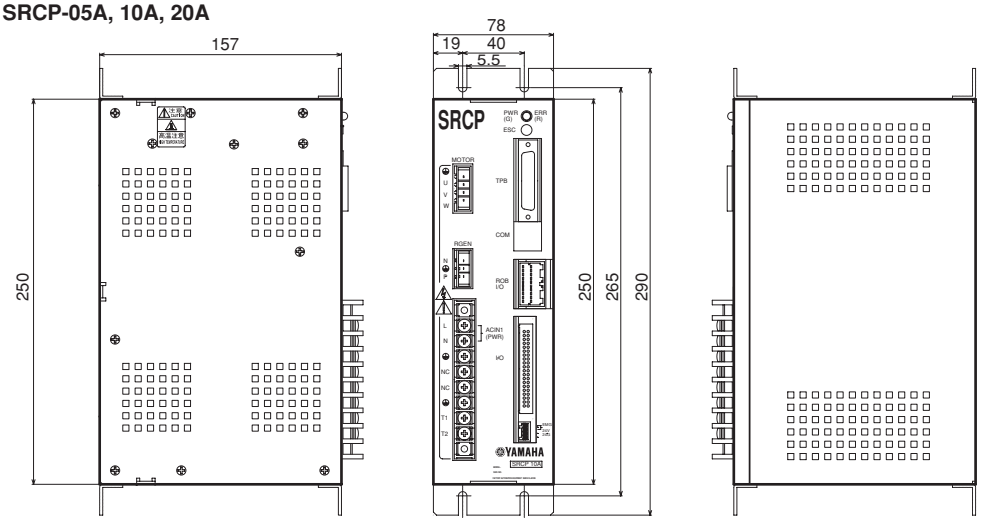
SRCP-05



SRCP-10, 20



SRCP-05A, 10A, 20A



1-3-2 TPB

1. Liquid Crystal Display (LCD) Screen
   This display has four lines of twenty characters each and is used as a program console.
2. Memory Card Slot
   An IC memory card can be inserted here. Be careful not to insert the card upside-down.
3. Control Keys
   The TPB can be operated in interactive data entry mode. Instructions are input through the control keys while reading the contents on the LCD screen.
4. Connection Cable
   This cable connects the TPB to the SRCP controller.
5. DC Power Input Terminal
   Not used.
6. Emergency Stop Button
   This is the emergency stop button. When pressed, it locks in the depressed position. To release this button, turn it clockwise.
   To cancel emergency stop, first release this button and then use the servo recovery command via the I/O interface or the servo recovery operation from the TPB.

Fig. 1-3 Exterior of the TPB

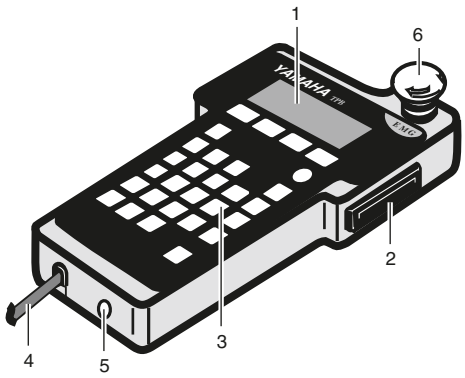
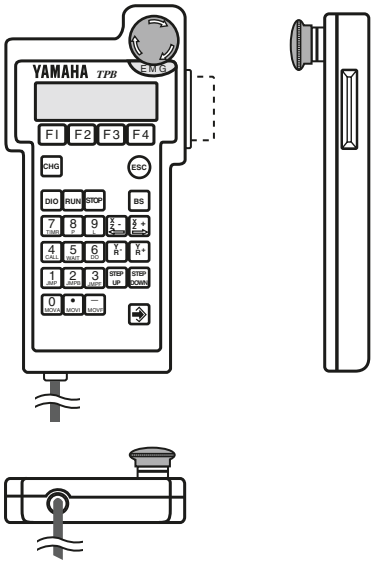


Fig. 1-4 Three-side view of the TPB

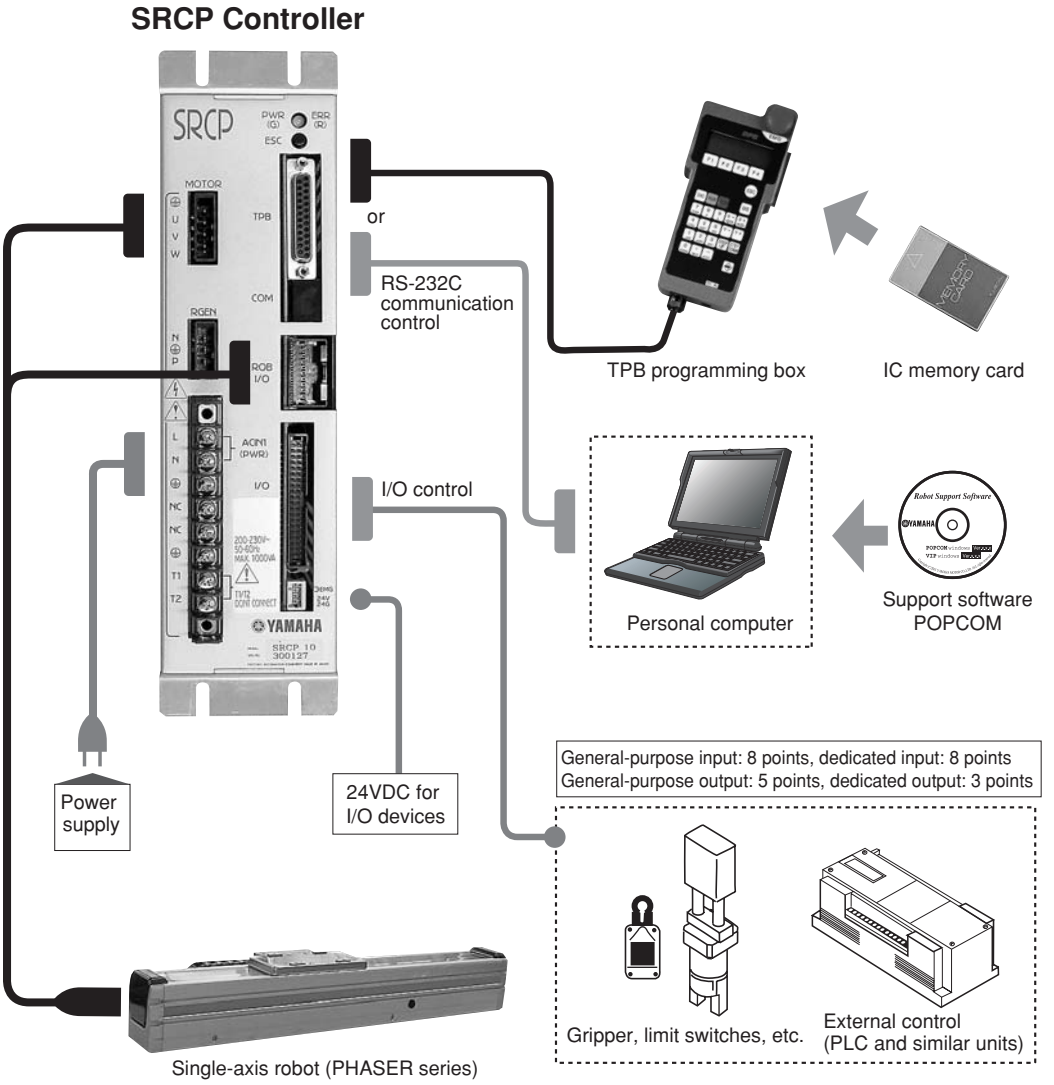


1-4 System Configuration

1-4-1 System configuration

The SRCP controller can be combined with various peripheral units and optional products to configure a robot system as shown below.

Fig.1-5 System configuration diagram



* Programming box TPB and support software POPCOM are sold separately.

1-5 Accessories and Options

1-5-1 Accessories

The SRCP robot controller comes with the following accessories. After unpacking, check that all items are included.

1. EXT. CN connector
   Connector : 733-104 made by WAGO 1 piece
2. I/O. CN connector with flat cable (option)
   Connector : XG4M-4030-U made by OMRON 1 piece
3. RS-232C dust cover
   XM2T-2501 made by OMRON 1 piece

1-5-2 Peripheral options

The following options are available for the SRCP controller:

1. TPB
   This is a hand-held programming box that connects to the SRCP controller for teaching point data, editing robot programs and operating the robot. The TPB allows interactive user operation by simple menus so that even first-time users can easily operate the robot with the TPB.
2. IC memory card
   An IC memory card can be used with the TPB to back up programs, point data and parameter data.
3. POPCOM
   The POPCOM is support software that runs on a PC (personal computer) connected to the SRCP controller. The POPCOM software allows easy editing of robot programs and operation of a robot just the same as with a TPB.

MEMO

Chapter 2 INSTALLATION AND CONNECTION

This chapter contains precautions that should be observed when installing the controller, as well as procedures and precautions for wiring the controller to the robot and to external equipment.

2-1 Installing the SRCP Controller

2-1-1 Installation method

Using the L-shaped brackets attached to the top and bottom of the controller, install the controller from the front or rear position. (See Fig.1-2 Three-side view of the SRCP controller.)

2-1-2 Installation location

- Install the controller in locations where the ambient temperature is between 0 to 40°C and the humidity is between 35 to 85% without condensation.
- Do not install the controller upside down or at an angle.
- Install the controller in locations with sufficient space (at least 20mm away from the wall or other object) for good ventilation and air flow.
- Do not install the controller in locations where corrosive gases such as sulfuric acid or hydrochloric acid gas are present, or in atmosphere containing flammable gases and liquids.
- Install the controller in locations with a minimal amount of dust.
- Avoid installing the controller in locations subject to cutting chips, oil or water from other machines.
- Avoid installing the controller in locations where electromagnetic noise or electrostatic noise is generated.
- Avoid installing the controller in locations subject to shock or large vibration.

2-2 Connecting the Power Supply

2-2-1 Power supply

Type and Item	Power supply voltage	No. of phases	Frequency	Max. power consumption
SRCP-05	AC200 to 230V ±10%	Single-phase	50/60Hz	400VA or less
SRCP-10	AC200 to 230V ±10%	Single-phase	50/60Hz	600VA or less
SRCP-20	AC200 to 230V ±10%	Single-phase	50/60Hz	1000VA or less
SRCP-05A	AC100 to 115/200 to 230V ±10%	Single-phase	50/60Hz	400VA or less
SRCP-10A	AC100 to 115/200 to 230V ±10%	Single-phase	50/60Hz	600VA or less
SRCP-20A	AC100 to 115/200 to 230V ±10%	Single-phase	50/60Hz	1000VA or less

⚠ ***CAUTION***

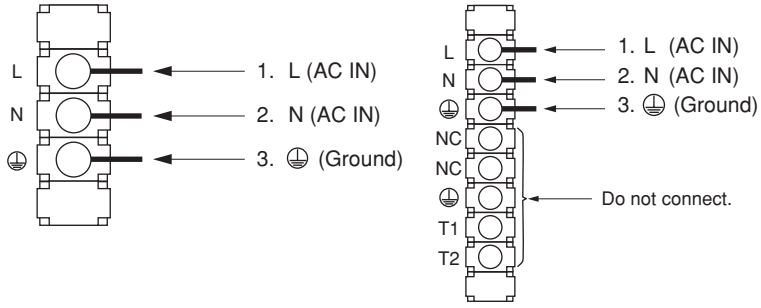
If the power supply voltage drops below the above range during operation, the alarm circuit will work and return the SRCP controller to the initial state the same as just after power-on, or stop operation. To avoid this problem, use a regulated power supply with voltage fluctuations of less than ±10%.

Since the SRCP controller uses a capacitor input type power supply circuit, a large inrush current flows when the power is turned on. Do not use fast-blow circuit breakers and fuses. For the same reason, avoid turning the power off and on again repeatedly in intervals of less than 10 seconds. This could harm the main circuit elements in the SRCP controller.

2-2-2 Connecting the power supply

Connect the power supply to the power terminal block on the front panel of the SRCP controller. Make correct connections while referring to the printed letters and mark. Misconnections may result in serious danger such as fire. Securely connect the end of each wire to the terminal so that it will not come loose.

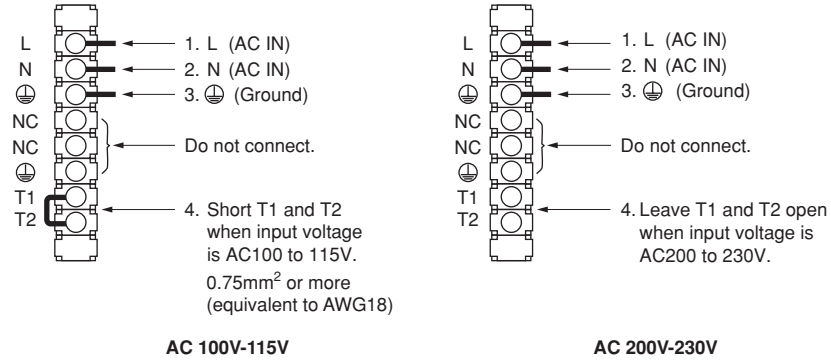
Fig. 2-1 Power supply connections (SRCP-05, SRCP-10, SRCP-20)



SRCP-05 **SRCP-10, 20**

The SRCP-05A, SRCP-10A and SRCP-20A have different connections to T1 and T2, depending on the input voltage.

Fig. 2-2 Power supply connections (SRCP-05A, SRCP-10A, SRCP-20A)



AC 100V-115V **AC 200V-230V**

⚠ *CAUTION*

The SRCP series controller does not have a power switch. Be sure to provide a power supply breaker (insulation) of the correct specifications that will turn the power on or off to the entire system including the robot controller. Power to EXT. CN must first be supplied before supplying power to the power supply terminal block. If this order is reversed, an alarm (06: 24V POWER OFF) might be issued to prevent operation. (See "2.7 Connecting to the EXT. CN Connector" in this chapter and Chapter 3, "I/O INTERFACE".)

⚠ **WARNING**

Before beginning the wiring work, make sure that the power supply for the entire system is turned off. Doing the wiring work while power is still turned on may cause electrical shocks.

2-2-3 Installing an external leakage breaker

To ensure safety, a leakage breaker must be installed in the power supply connection section of the robot controller. Since the robot controller drives the motors by PWM control, leakage current flows at high frequencies. This might cause the external leakage breaker to malfunction.
When installing an external leakage current breaker, it is important to choose the optimum sensitivity current rating (IΔn). (Check the leakage breaker manufacturer's data sheets to select the optimum product compatible with inverters.)

	Leakage current
SRCP	4mA (Max.)

⚠ *CAUTION*

1. ***Leak current was measured with a leak tester with a low-pass filter turned on (100Hz).***
   Leak tester: Hioki Electric 3283
2. ***When using two or more controllers, sum the leakage current of each controller.***
3. ***Make sure that the controller is securely grounded.***
4. ***Stray capacitance between the cable and FG may vary depending on the cable installation condition, causing the leakage current to fluctuate.***

⚠ **WARNING**

Electrical shocks, injuries or fires might occur if the motor breaks down while the robot controller is used without installing a leakage breaker.

2-2-4 Installing a circuit protector

To ensure safety, a circuit protector must be installed in the power supply connection section of the robot controller. An inrush current, which might be from several to nearly 20 times higher than the rated current, flows at the instant that the SRCP controller is turned on or the robot motors start to operate.
When installing an external circuit protector for the robot controller, select a circuit protector that provides optimum operating characteristics.
To ensure proper operation, we recommend using a medium to slow response circuit protector with an inertial delay function. (Refer to the circuit protector manufacturer's data sheets for making the selection.)

Example

	Rated current	Operating characteristics
SRCP	20A	Slow type with inertia delay (300% 2 sec. 1000% 0.01 sec.)

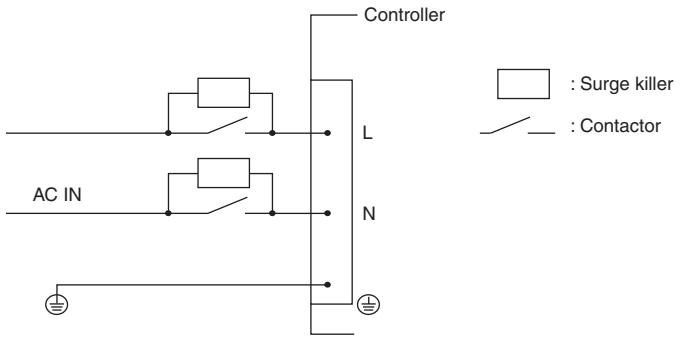
⚠ **WARNING**

Electrical shocks, injuries or fires might occur if the motor breaks down while the robot controller is used without installing a circuit protector.

2-2-5 Installing current control switches

When controlling the power on/off of the robot controller from an external device such as a PLC, a current control switch (contactor, breaker, etc.) may be used. In this case, the current control switch usually creates a large on/off inrush current. To minimize this on/off inrush current, surge killers must be installed for surge absorption. Connect a surge killer in parallel with and close to each contact of the current control switch.

Recommended surge killer:
Okaya Electric XE1201, XE1202, RE1202
Example:



2-2-6 Insulation resistance and voltage breakdown tests

Never attempt insulation resistance tests or voltage breakdown tests on the SRCP controller. Since capacitive grounding is provided between the controller body and 0V, these tests may mistakenly detect excess leakage current or damage the internal circuitry. If these tests are required, please consult your YAMAHA sales office or representative.

2-3 Grounding

The SRCP controller must be grounded to prevent danger to personnel from electrical shocks in case of electrical leakage and prevent equipment malfunctions due to electrical noise.
We strongly recommend that Class D (grounding resistance of 100 ohms or less) or higher grounding be provided. For grounding the controller, use the ground terminal on the power supply terminal block.

* Class D grounding is the same as Class 3 grounding previously used.

2-4 Connecting the SRCP to the Control Unit

The SRCP controller can be operated either through the TPB programming box or through a PC (personal computer) equipped with an RS-232C terminal.
When using the TPB, plug the TPB cable connector into the TPB connector of the SRCP controller. (Refer to "4-1-1 Connecting the TPB to the SRCP controller".)
When using a PC, plug the RS-232C interface cable connector (25 pins) into the TPB connector of the SRCP controller. (Refer to "11-2 Communication Cable Specifications".)
To prevent equipment malfunction due to noise, we strongly recommend that Class D (grounding resistance of 100 ohms or less) or higher grounding be provided.

2-5 Connecting to the Robot

First make sure that the power to the SRCP controller is turned off, and then connect the robot cable to the robot I/O connector and motor connector on the front panel of the SRCP controller. Fully insert the robot cable until it clicks in position.

* When the robot cable is disconnected from the controller, an alarm (15: FEEDBACK ERROR 2) is issued.

2-5-1 Robot I/O connector and signal table

Mating connector type No. : 0-174047-2 (AMP)
Mating connector contact type No. : 0-175180-2
SRCP's connector type No. : 0-174055-2

Signal table

Terminal No.	Signal name	Description	Terminal No.	Signal name	Description
1	PS+	Position SIN input (+)	11	NC	No connection
2	PS-	Position SIN input (-)	12	ORG	Origin sensor input
3	PC+	Position COS input (+)	13	+24V	Origin sensor, +24V
4	PC-	Position COS input (-)	14	+24V	Origin sensor, +24V
5	+5V	+5V	15	0V	Origin sensor, 24GND
6	GND	GND	16	0V	Origin sensor, 24GND
7	Z+	Linear scale Z+	17	BK+	Brake (+)
8	Z-	Linear scale Z-	18	BK-	Brake (-)
9	DG	Digital ground	19	NC	No connection
10	DG	Digital ground	20	FG	Frame ground

2-5-2 Motor connector and signal table

Mating connector type No. : 1-178128-4 (AMP)
Mating connector contact type No. : 1-175218-5
SRCP's connector type No. : 1-179277-5

Signal table

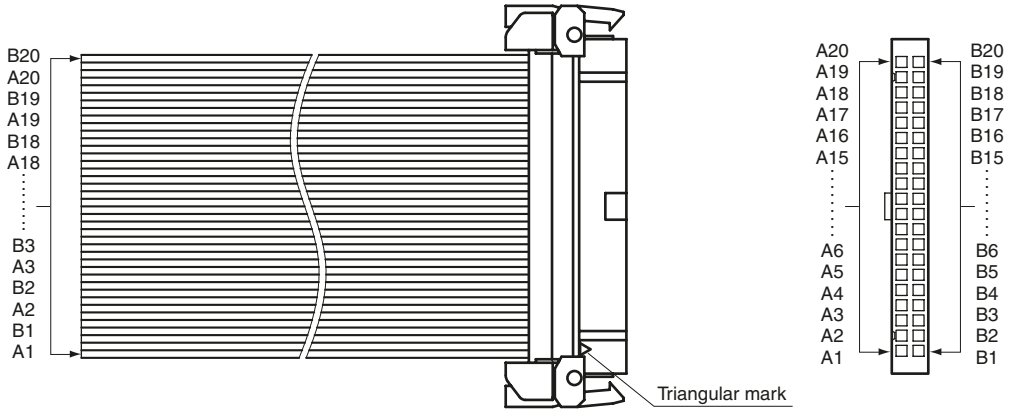
Terminal No.	Signal name	Description	Terminal No.	Signal name	Description
1	FG	Frame ground	3	MV	Motor V-phase output
2	MU	Motor U-phase output	4	MW	Motor W-phase output

2-6 Connecting to the I/O. CN Connector

The I/O. CN connector is used for connecting the SRCP controller to external equipment such as a PLC. When using external equipment for I/O control, connect the wiring to the I/O. CN connector (with a flat cable) supplied as an accessory and then plug it into the I/O. CN connector on the SRCP controller.
Signals assigned to the I/O. CN connector terminals and their functions are described in detail in Chapter 3.

The mating connector with a flat cable (option) for the I/O. CN terminal on the SRCP series controller is as follows:

Mating connector type No. : XG4M-4030-U (OMRON)
SRCP's I/O. CN connector type No. : XG4C-4034



⚠ ***CAUTION***

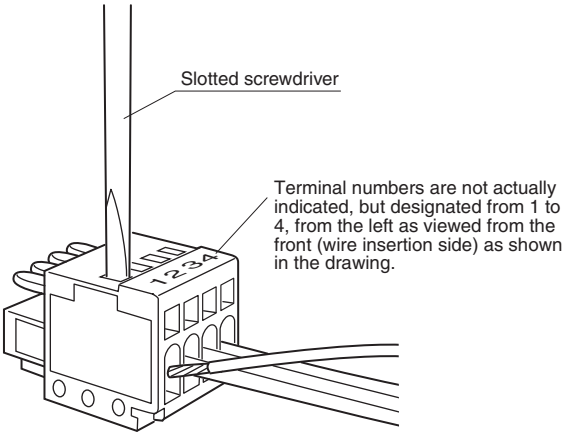
Regardless of whether I/O control is used or not, DC 24V power must be supplied to EXT. CN to enable robot operation. If no power is supplied to EXT. CN, an alarm (06: 24V POWER OFF) is issued to prevent operation. (See "2.7 Connecting to the EXT. CN Connector" in this chapter.)
If not using I/O control, disable the interlock function in PRM34 (System mode selection parameter). If the interlock function is not disabled, it will be triggered during operation to prohibit the robot from operating.

2-7 Connecting to the EXT. CN Connector

Connect an emergency stop circuit and a 24V power supply for I/O control to the EXT. CN connector. Make the necessary wiring hookup (see below) to the mating connector that comes with the SRCP controller and then plug it into the EXT. CN connector. Make sure the wiring is correct since miswiring may cause serious accidents such as fire. Regardless of whether I/O control is used or not, 24V power for I/O control must always be supplied to the EXT. CN connector.
The meaning and operation of signals assigned to each terminal on the EXT. CN connector are explained in detail in Chapter 3, "I/O INTERFACE".

The mating connector for the EXT. CN terminal on the SRCP series controller is as follows:

Mating connector type No. : 733-104 (WAGO)
SRCP's EXT. CN connector type No. : 733-364



To make the wiring hookup to the mating connector (WAGO 733-104), insert the wire lead into the terminal slot while pressing down the internal spring with a slotted screwdriver through the top slot. (If you have a dedicated tool, insert it into the smaller slot just above each terminal slot for wire insertion to make a quick connection.

⚠ ***CAUTION***

Regardless of whether I/O control is used or not, DC 24V power must be supplied to EXT. CN to enable robot operation. If no power is supplied to EXT. CN, an alarm (06: 24V POWER OFF) is issued to prevent operation. Power to EXT. CN must first be supplied before supplying AC power to the power supply terminal block. If this order is reversed, an alarm (06: 24V POWER OFF) might be issued.

⚠ ***CAUTION***

If you do not configure an emergency stop circuit, then short terminal No. 1 (EMG1) to terminal No. 2 (EMG2). Unless these terminals are shorted, emergency stop is always activated to prohibit the robot from operating.

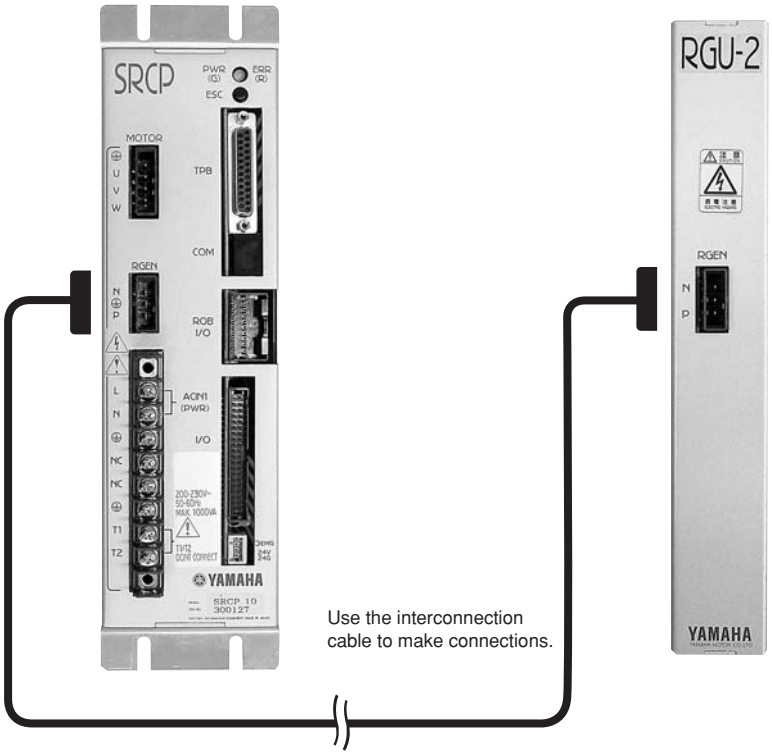
⚠ **DANGER**

Be sure to turn off the power to the entire robot system before doing any wiring to the SRCP controller. Failure to do so may cause electrical shocks.

2-8 Connecting to the Regenerative Unit

Some types of robots must be connected to a regenerative unit. In such cases, use the interconnection cable to connect the SRCP controller to the regenerative unit.

Fig. 2-3 Connecting the SRCP controller to a regenerative unit



MEMO

Chapter 3 I/O INTERFACE

The SRCP series has I/O interface connectors (EXT. CN and I/O. CN) as a standard feature. The EXT. CN is used for emergency stop input and 24V power input for I/O control. The I/O. CN consists of an interlock input, 7 dedicated command inputs, 3 dedicated outputs, 8 general-purpose inputs, 5 general-purpose outputs, feedback pulse outputs, etc. These I/O interfaces allow exchanging commands and data between the SRCP series and external equipment. These I/O interfaces can also directly connect to and control actuators such as valves and sensors. To construct a system utilizing the features of the SRCP series, you must understand the signals assigned to each terminal on the I/O. CN and EXT. CN and how they work. This chapter covers this fundamental information. This chapter also provides examples of I/O circuit connections and timing charts for expanding the system by using a PLC or similar devices. Refer to these diagrams and examples when creating sequence programs.
Terms "ON" and "OFF" used in this chapter mean "on" and "off" of switches connected to the input terminal when referring to input signals. They also mean "on" and "off" of output transistors when referring to output signals.

3-1 I/O Signals

The SRCP controller has two I/O interface connectors (EXT. CN and I/O. CN) as a standard feature. The EXT. CN is used for emergency stop input and 24V power input for I/O control. The I/O. CN is used for interlock signal input, dedicated command input, dedicated output, general-purpose input and output, and feedback pulse output.

3-1-1 I/O. CN connector signals

The I/O. CN connector of the SRCP controller has 40 pins, with an individual signal assigned to each pin. The following table shows the pin number as well as the name and description of each signal assigned to each pin. For a more detailed description of each signal, refer to "3-2 Input Signal Description" and onwards.

No.	Pin No.	Signal name	Description	No.	Pin No.	Signal name	Description
1	A1	ABS-PT	Absolute point movement command	2	B1	INC-PT	Relative point movement command
3	A2	AUTO-R	Automatic operation start command	4	B2	STEP-R	Step operation start command
5	A3	ORG-S	Return-to-origin command	6	B3	RESET	Reset command
7	A4	SERVO	Servo recovery command	8	B4	LOCK	Interlock
9	A5	DI0	General-purpose input 0	10	B5	DI1	General-purpose input 1
11	A6	DI2	General-purpose input 2	12	B6	DI3	General-purpose input 3
13	A7	DI4	General-purpose input 4	14	B7	DI5	General-purpose input 5
15	A8	DI6	General-purpose input 6	16	B8	DI7/SVCE	General-purpose input 7/SERVICE mode input
17	A9	DO0	General-purpose output 0	18	B9	DO1	General-purpose output 1
19	A10	DO2	General-purpose output 2	20	B10	DO3	General-purpose output 3
21	A11	DO4	General-purpose output 4	22	B11	END	End-of-run output
23	A12	BUSY	Command-in-progress output	24	B12	READY	Ready-to-operate output
25	A13	FG	Frame ground	26	B13	FG	Frame ground
27	A14	GND	Signal ground	28	B14	GND	Signal ground
29	A15	NC	Reserved (Do not use.)	30	B15	NC	Reserved (Do not use.)
31	A16	NC	Reserved (Do not use.)	32	B16	NC	Reserved (Do not use.)
33	A17	PA+	Feedback pulse output	34	B17	PA-	Feedback pulse output
35	A18	PB+	Feedback pulse output	36	B18	PB-	Feedback pulse output
37	A19	PZ+	Feedback pulse output	38	B19	PZ-	Feedback pulse output
39	A20	PZM+	Feedback pulse output	40	B20	PZM-	Feedback pulse output

NOTE

Pin B8 functions as the SERVICE mode input terminal only when the SERVICE mode function is enabled.

3-1-2 EXT. CN connector signals

The EXT. CN connector of the SRCP controller has 4 pins, with an individual signal assigned to each pin. The following table shows the pin number as well as the name and description of each signal assigned to each pin. For a more detailed description of each signal, refer to "3-2 Input Signal Description" and onwards.

Pin No.	Signal name	Description	Pin No.	Signal name	Description
1	EMG1	Emergency stop input 1 (used with EMG2)	2	EMG2	Emergency stop input 2 (used with EMG1)
3	24V	24V power supply for sequence input	4	24G	24V power supply for sequence input

NOTE

The positive polarity of the 24V DC must be connected to pin 3 (24V) and the negative polarity to pin 4 (24G).

3-2 Input Signal Description

Input signals consist of 7 dedicated command inputs, 8 general-purpose inputs and interlock signals fed to the I/O. CN terminal, as well as an emergency stop input fed to the EXT. CN terminal.

* DI7 functions as the SERVICE mode input when the SERVICE mode function is enabled. In this case, 7 general-purpose inputs are available.

All input circuits other than the emergency stop input use photocoupler-isolated input circuit specs. Only the emergency stop input circuit uses contact point input circuit specs. This contact point is directly connected to the relay coil that turns the internal motor power supply on and off.

3-2-1 Dedicated command input

The dedicated command input is used to control the SRCP controller from a PLC or other external equipment. To accept this input, the READY, BUSY and LOCK signals must be set as follows.

- ■ READY signal : ON
- ■ BUSY signal : OFF
- ■ LOCK signal : ON

If the above conditions are not satisfied, then dedicated command inputs cannot be accepted even if they are input from external equipment. For example, when the BUSY signal is on, this means that the controller is already executing a dedicated command, so other dedicated commands are ignored even if they are input. When the LOCK signal is off, no other commands can be accepted since an interlock is active. (One exception is the reset and servo recovery commands that can be executed even when the LOCK signal is off as long as the READY and BUSY signals meet the above conditions.)

A dedicated command input is accepted when the dedicated command input is switched from "off" to "on" (at the instant the contact point closes). Whether the controller accepts the command or not can be checked by monitoring the BUSY signal.
Note that dedicated command inputs cannot be used as data in a program.

⚠ ***CAUTION***

The dedicated command inputs explained below must always be pulse inputs. In other words, they must be turned off (contact open) after the BUSY signal turns on.
If a dedicated command input is not turned off, then the BUSY signal will remain on even when the command has ended normally. So the next command will not be accepted.

⚠ ***CAUTION***

When the SERVICE mode function is enabled, the following safety control will function. (See "10-4 SERVICE mode function" for more details.)
- ***No dedicated commands can be executed in "SERVICE mode state" when command input from other than the TPB is prohibited.***

■ Absolute point movement command (ABS-PT)

This command moves the robot to an absolute position specified by a point number at a specified speed along an axis coordinate whose origin is defined as 0. The point number and speed are specified by general-purpose input. (See "3-2-2 General-purpose input (DI0 to DI7)".)

NOTE

The number of general-purpose input (DI) points used to specify the point numbers and speed differs depending on whether SERVICE mode is enabled or disabled and also on the PRM7 (I/O point movement command speed parameter) setting. (See "3-2-2 General-purpose input (DI0 to DI7)".)

CAUTION

The DI0 to DI7 (DI0 to DI6 when SERVICE mode is enabled) status must be confirmed before ABS-PT is executed. (See "3-6-6 When executing a point movement command".)

■ Relative point movement command (INC-PT)

This command moves the robot a distance specified by a point number from the current position at a specified speed. The point number and speed are specified by general-purpose input. (See "3-2-2 General-purpose input (DI0 to DI7)".)

NOTE

Current position does not always indicate the actual robot position. More accurately, it is the current position data stored in the controller. Each time a movement command is executed correctly, the current position data in the controller is replaced with the target position data of the movement command.
Therefore, if the robot is stopped by an interlock while executing a relative movement command, re-executing the same relative movement command moves the robot to the target position. (The robot does not move a relative distance from the stopped position by the interlock.)
Similarly, after a robot movement command is executed, the controller still retains the target position data of that movement command as the current position data even if you move the robot to another position by manual operation.
When a relative movement command is executed under this condition, the robot moves the specified distance from the target position of the movement command that was previously executed, rather than the actual robot position, so use caution.
Current position data differs from the actual robot position when:

- *Emergency stop or interlock (LOCK) was activated while the robot was moving.*
- *A communication command ^C (movement interruption) was transmitted while the robot was moving.*
- *The SERVICE mode input was changed while the robot was moving.*
- *The robot was moved by manual operation.*
- *The robot was moved by hand during servo-off (including emergency stop).*

NOTE

The number of general-purpose input (DI) points used to specify the point numbers and speed differs depending on whether SERVICE mode is enabled or disabled and also on the PRM7 (I/O point movement command speed parameter) setting. (See "3-2-2 General-purpose input (DI0 to DI7)".)

CAUTION

The DI0 to DI7 (DI0 to DI6 when SERVICE mode is enabled) status must be specified before INC-PT is executed. (See "3-6-6 When executing a point movement command".)

■ Automatic operation start command (AUTO-R)

This command executes the robot program continuously, starting from the current step.

All tasks are executed if the robot program is a multi-task program.

■ Step operation start command (STEP-R)

This command executes the robot program one step at a time, starting from the current step.

Only the selected task is executed even if the robot program is a multi-task program.

■ Return-to-origin command (ORG-S)

This command returns the robot to its origin position by using stroke-end detection as the origin detection method.

NOTE

The magnetic pole is detected simultaneously with return-to-origin operation. Return-to-origin is incomplete each time the power is turned on. Always perform return-to-origin after turning on power to the controller, before starting operation. Return-to-origin is also always incomplete after a parameter related to the origin position is changed. Return-to-origin must be re-performed in this case.

CAUTION

If the robot is operated while return-to-origin is still incomplete, the necessary thrust to move the robot is unavailable and an alarm or abnormal operation occurs. Always perform return-to-origin before starting robot operation.

CAUTION

When performing return-to-origin by the stroke-end detection method, do not interrupt return-to-origin operation while the origin position is being detected (robot is making contact with its mechanical limit). Otherwise, the operation will stop due to a controller overload alarm and the power will need to be turned off and back on again.

CAUTION

Do not continuously repeat return-to-origin operation. If return-to-origin must be repeated by the stroke-end detection method, wait at least 5 seconds before repeating it.

■ Servo recovery command (SERVO)

After emergency stop, releasing the emergency stop button and turning this input on (closing the contact) turns the servo power on, so the robot is ready for restart.
(As with other dedicated command inputs, the servo recovery command should be a pulse input, so it must be turned off (contact open) when the BUSY signal turns on.)

NOTE

When the servo is first turned on after power-on, a sound is heard for 0.5 to 2 seconds during servo-on. This sound is produced by a small robot movement for acquiring information needed to control the robot, and is not an abnormal condition.

■ Reset command (RESET)

This command returns the program step to the first step of the lead program and turns off the general-purpose outputs and the memory I/O. It also clears the point variable "P" to 0.

* When PRM33 ("Operation at return-to-origin complete" parameter) is set to 1 or 3, DO4 does not turn off even if the reset command is executed. Likewise, when PRM46 ("Servo status output" parameter) is set to 1, DO3 does not turn off even if the reset command is executed.

NOTE

The lead program is the program that has been selected as the execution program by the TPB or POPCOM. (See "9-4 Switching the Execution Program".)
The lead program can also be selected by executing a communication command "@SWI". It may also be selected when the program data is loaded into the SRCP controller from the memory card.

3-2-2 General-purpose input (DI0 to DI7)

These general-purpose inputs are available to users for handling data input in a program.
These inputs are usually connected to sensors or switches. These inputs can also be directly connected to a PLC output circuit.

As a special function during execution of an ABS-PT or INC-PT point movement command, these general- purpose inputs can be used to specify the point numbers and movement speed.
The number of general-purpose input (DI) points used to specify the point numbers and speed differs depending on whether SERVICE mode is enabled or disabled and also on the PRM7 (I/O point movement command speed parameter) setting.
For example, when PRM7 is 100 in normal mode (SERVICE mode disabled), the point numbers should be input in binary code with DI0 to DI7 to specify P0 to P255 as shown in the table below.
If PRM7 is 0 in normal mode (SERVICE mode disabled), then point numbers should be input with DI0 to DI5 in binary code to specify P0 to P63 as shown in the table below. The movement speed is specified as 100% when both DI6 and DI7 are off. In other cases, it is set to the speed specified by the parameter. (See "5-2 Parameter Description".) Also see the tables below for more details.

	PRM7	DI used to specify point numbers	DI used to specify speed	Speed setting method
NORMAL mode (SERVICE mode disabled)	100 (default setting)	DI7 to DI0 (256 points)	Not specified	100%
	1 to 99	DI6 to DI0 (128 points)	DI7	Speed setting pattern 2
	0	DI5 to DI0 (64 points)	DI7, DI6	Speed setting pattern 1
SERVICE mode enabled	100 (default setting)	DI6 to DI0 (128 points)	Not specified	100%
	1 to 99	DI5 to DI0 (64 points)	DI6	Speed setting pattern 3

Example of point number setting

DI No. / Point No.	DI7 (2^7)	DI6 (2^6)	DI5 (2^5)	DI4 (2^4)	DI3 (2^3)	DI2 (2^2)	DI1 (2^1)	DI0 (2^0)
P0	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF
P1	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON
P3	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON
P7	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
P15	OFF	OFF	OFF	OFF	ON	ON	ON	ON
P31	OFF	OFF	OFF	ON	ON	ON	ON	ON
P63	OFF	OFF	ON	ON	ON	ON	ON	ON
P127	OFF	ON	ON	ON	ON	ON	ON	ON
P254	ON	ON	ON	ON	ON	ON	ON	OFF
P255	ON	ON	ON	ON	ON	ON	ON	ON

Example of point movement speed setting

Speed setting pattern 1

DI7	DI6	Movement speed
OFF	OFF	100%
OFF	ON	PRM41
ON	OFF	PRM42
ON	ON	PRM43

Speed setting pattern 2

DI7	Movement speed
OFF	100%
ON	PRM7

Speed setting pattern 3

DI6	Movement speed
OFF	100%
ON	PRM7

* DI7 functions as the SERVICE mode input when the SERVICE mode function is enabled. In this case, DI0 to DI6 can be used as the general-purpose inputs.

3-2-3 SERVICE mode input (SVCE)

When the SERVICE mode function is enabled, DI7 functions as the SERVICE mode input (SVCE). The SERVICE mode input is used to notify the SRCP controller whether the current state is a "SERVICE mode state". This input should be turned off (contact open) in "SERVICE mode state".
Refer to "10-4 SERVICE mode function" for details on the SERVICE mode function.

NOTE
Operation stops immediately if the SERVICE mode input status is changed during robot operation while the SERVICE mode function is enabled.

NOTE
Even with the SERVICE mode function enabled, the SERVICE mode input status can be checked in the program as DI7.

3-2-4 Interlock (LOCK)

This input is used to temporarily stop robot movement.
The robot immediately stops when this input is turned off (contact open) during execution of a dedicated I/O command or during program operation or return-to-origin operation from the TPB (or PC). (This also interrupts the robot program operation.)
As long as this input is off (contact open), no dedicated I/O commands can be executed, and also no programs and return-to-origin operation can be performed from the TPB (or PC). The only exceptions to this are the reset command and servo recovery command that can be executed regardless of whether the LOCK signal is on or off. Leave this LOCK signal turned on (contact closed) during normal operation.
Once this LOCK signal is turned off (contact open), the robot remains stopped even after this input is turned back on (contact closed), until another command (AUTO-R, ORG-S, etc.) is input.
Changing the PRM34 (System mode selection parameter) setting enables or disables the interlock function. (See "5-2 Parameter Description".)

3-2-5 Emergency stop inputs 1, 2 (EMG1, EMG2)

Use these inputs to trigger robot emergency stop from an external safety device (for example, safety enclosure, manual safety switch, etc.). Servo power turns off at the same time when the contact between EMG1 and EMG2 is open (turned off). Use a relay contact with a current capacity of at least 50mA.
To resume operation, close (turn on) the contact between EMG1 and EMG2, check that the READY signal is turned on, and then input the servo recovery command (SERVO). The servo will turn on to enable robot operation.
The TPB or PC can also be used to reset emergency stop when the SRCP controller is connected to the TPB or PC.

⚠ ***CAUTION***
Emergency stop inputs 1 and 2 (EMG1 and EMG2) are provided on the EXT. CN connector, and not on the I/O. CN connector. Do not use the different inputs.

3-3 Output Signal Description

The output signals consist of 3 dedicated outputs (READY, BUSY and END), 5 general-purpose outputs, and feedback pulse outputs. In this section, terms "ON" and "OFF" mean the output transistors are "on" and "off".

3-3-1 Dedicated output

The dedicated outputs are used for exchanging signals between the SRCP controller and an external device such as a PLC.

■ Ready-to-operate output (READY)

This output is on as long as the SRCP controller system is in normal operation. If an emergency stop or alarm occurs, then this output turns off to let the motor idle.

- When emergency stop was triggered:
  The READY signal turns on again when emergency stop is cancelled.
  Operation can be restarted by input of the servo recovery command (SERVO) after canceling emergency stop.
- When an alarm was issued:
  If the READY signal is off while the robot is not in emergency stop, this means that an alarm was issued. If this happens, correct the problem while referring to Chapter 13, "Troubleshooting".
  In this case, the SRCP controller should be turned off before attempting to restart operation.

■ Command-in-progress output (BUSY)

The BUSY signal is on during execution of a dedicated command input or a command from the TPB or PC. The BUSY signal turns on when the SRCP controller accepts a dedicated command input. The dedicated command input should be turned off (contact open) when the BUSY signal turns on. The BUSY signal turns off when command execution is complete. (At this point, all the dedicated command inputs must be turned off (contact open).)

⚠ *CAUTION*

The dedicated command input must be a pulse input so that it is off when the BUSY signal turns on. If the command input is left on, the BUSY signal cannot turn off even after the command execution is complete. As long as the BUSY signal is on, the SRCP controller will not accept other dedicated command inputs or commands from the TPB or PC. Avoid operating the TPB while the SRCP controller is being operated through the I/O interface. (Doing so might cause malfunctions during data exchange with a PLC or cause communication errors on the TPB side.)

■ End-of-run output (END)

The END signal turns off when a dedicated command input is received and turns on when command execution is complete. The END signal remains off if an error occurs or an interlock or emergency stop is triggered during command execution.

⚠ *CAUTION*

When a reset command or a movement command specifying a very small amount of movement is used, the command execution time will be very short. In other words, the period that the END signal is off will be very short (1ms or less in some cases).
The END signal does not change by operation from the TPB or PC.

NOTE

The PRM34 (system mode selection parameter) setting can be changed so that the execution result END signal output for the completed dedicated command occurs only after the dedicated command input turns off. (See section 5-2 "Parameter Description".)

3-3-2 General-purpose output (DO0 to DO4)

These general-purpose outputs are available to users for freely controlling on/off operation in a program.
These outputs are used in combination with an external 24V power supply, to drive loads such as solenoid valves and LED lamps. These outputs of course, can be directly connected to a PLC input circuit.
All general-purpose outputs are reset (turned off) when the SRCP controller is turned on or the program is reset.

* When PRM33 ("Operation at return-to-origin complete" parameter) is set to 1 or 3, DO4 does not turn off even if the program is reset. Similarly, when PRM46 ("Servo status output" parameter) is set to 1, DO3 does not turn off even if the program is reset.

General-purpose output (DO0 to DO4) can be used to perform the following specific functions by parameter setting.

Function name	Parameter	Usable general-purpose input
Alarm output	PRM32 (Alarm number output)	DO0 to DO4
Return-to-origin complete output	PRM33 (Operation at return-to-origin complete)	DO4
Servo-ON status output	PRM46 (Servo status)	DO3
Zone output	PRM53 (Zone output)	DO0 to 3

For more details, see "5-2 Parameter Description".

3-3-3 Feedback pulse output (PA±, PB±, PZ±, PZM±)

This outputs current position data as differential output.

Relation between pulse output and phase

Output pin	CW direction	CCW direction
A17:PA+ B17:PA-		
A18:PB+ B18:PB-		
A19:PZ+ B19:PZ-		

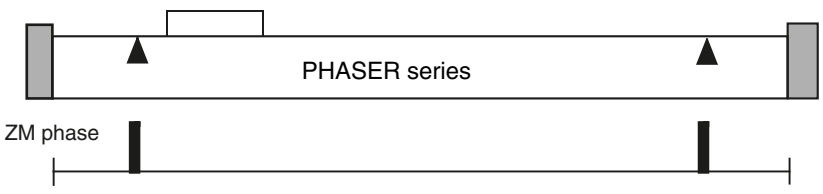
Number of output pulses and Z-phase timing

Number of output pulses*	Z-phase output timing
1 [pulses/μm]	Every 1024 μm

*: Number of output pulses is a count after being multiplied by 4.

ZM phase

The ZM phase is output at the magnetic pole detection points of the PHASER series. Refer to the drawing below.



▲ : Magnetic pole detection point

3-4 I/O Circuits

This section provides the SRCP controller I/O circuit specifications and examples of how the I/O circuits should be connected. Refer to these specifications and diagrams when connecting to external equipment such as a PLC.

3-4-1 I/O circuit specifications

■ Input Power

DC24V±10% (supplied through EXT.CN)

■ Input Circuit

Excluding emergency stop input circuit

Insulation method:	Photocoupler insulation
Input terminal:	Relay contact or NPN open collector transistor connected between input terminal and 0V terminal.
Input response:	30ms max.
Input current:	5mA/DC24V
Input sensitivity:	Current on: 3mA min. Current off: 1mA max.

Emergency stop input circuit

Input terminal:	Relay contact connected between emergency stop inputs 1 and 2 (between EMG1 and EMG2).
Input response:	5ms max.
Input current:	40mA/DC24V

■ Output Circuit

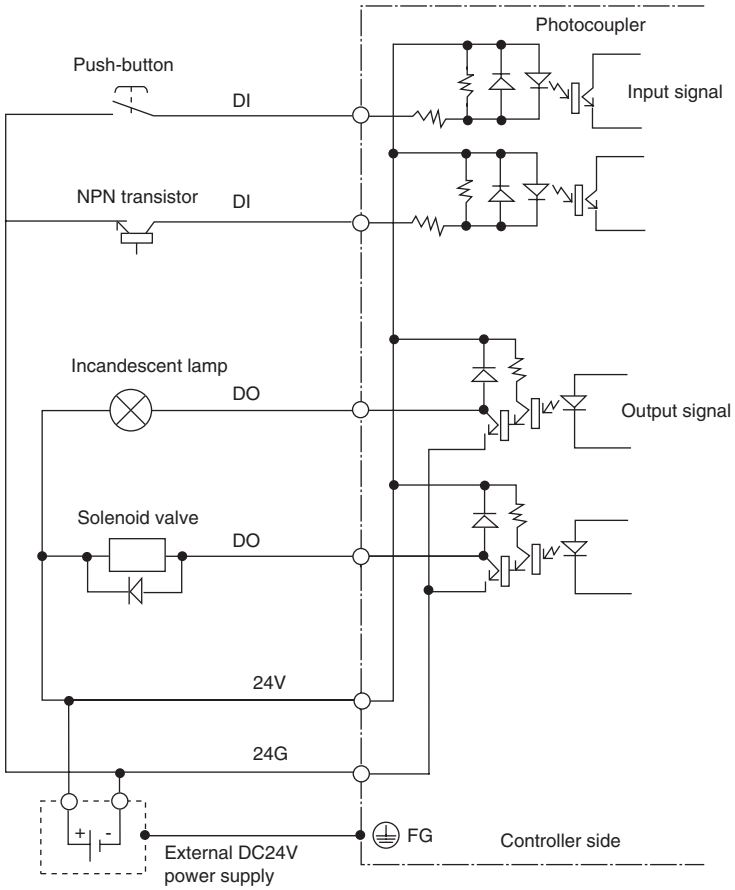
Insulation method:	Photocoupler insulation between internal circuit and output transistor
Output terminal:	NPN open collector output of all collective output common terminals (0V side)
Output response:	1ms max.
Max. output current:	50mA/DC24V per output
Residual ON voltage:	1.5V max.

■ Pulse Output Circuit

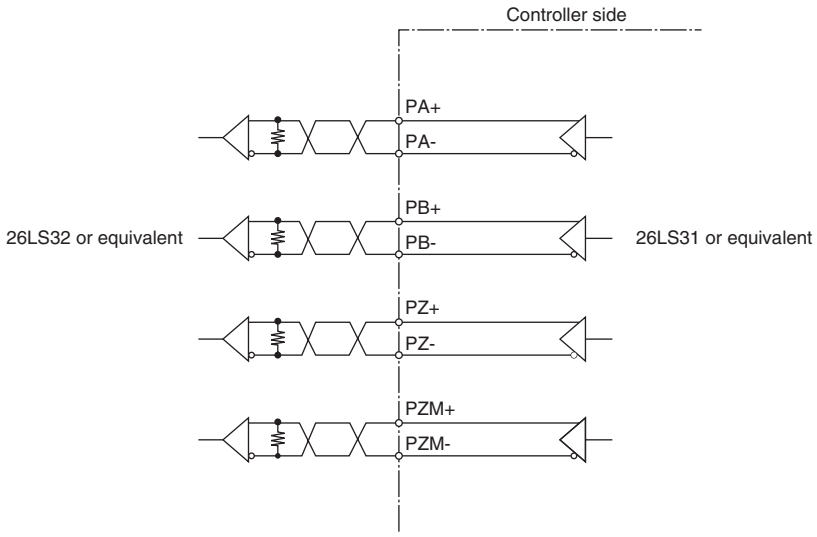
Output method:	Line driver (26LS31 or equivalent)
Maximum output current:	20mA

3-4-2 I/O circuit and connection example

I/O circuit and connection example



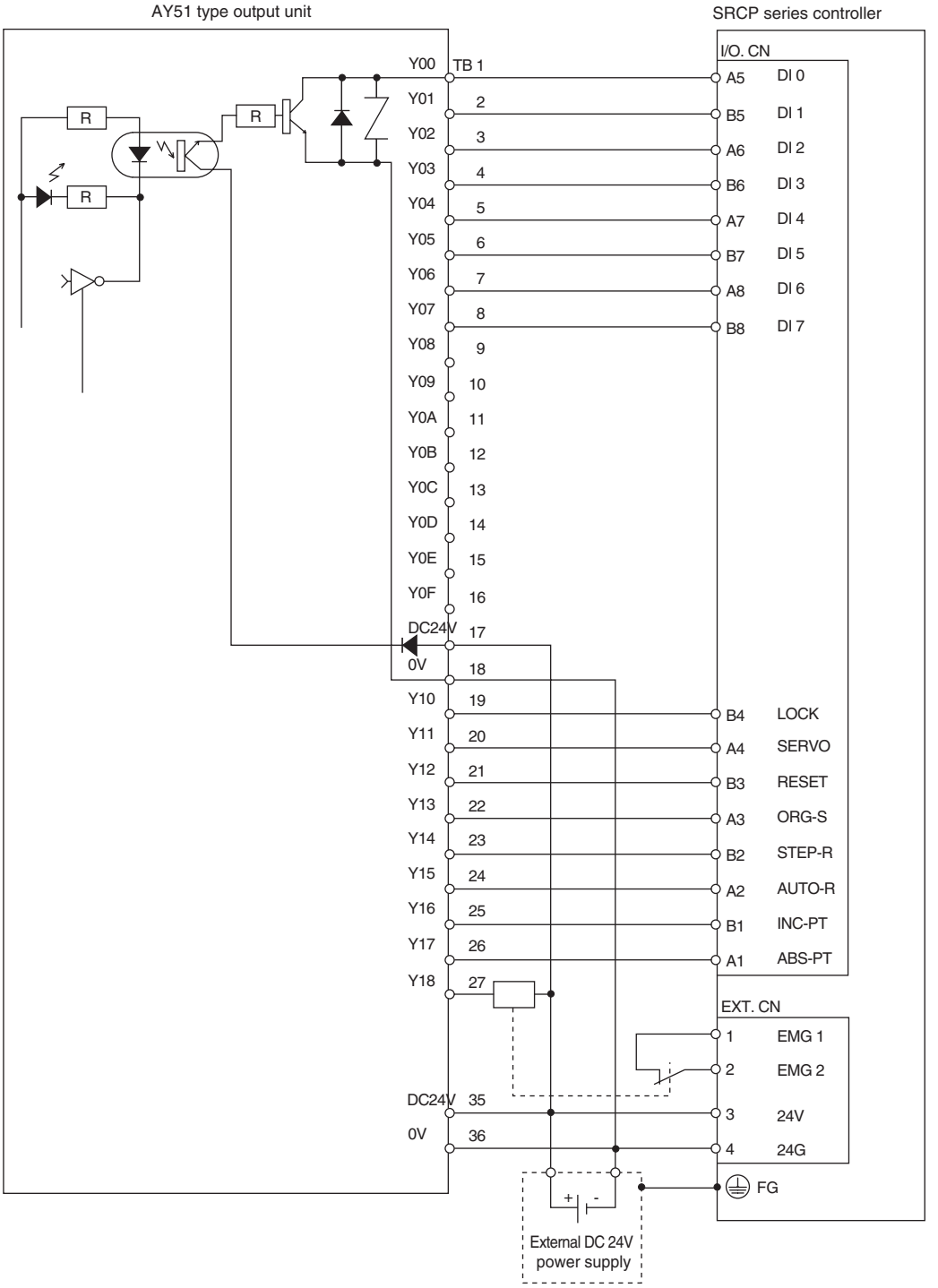
Pulse output circuit connection example



3-5 I/O Connection Diagram

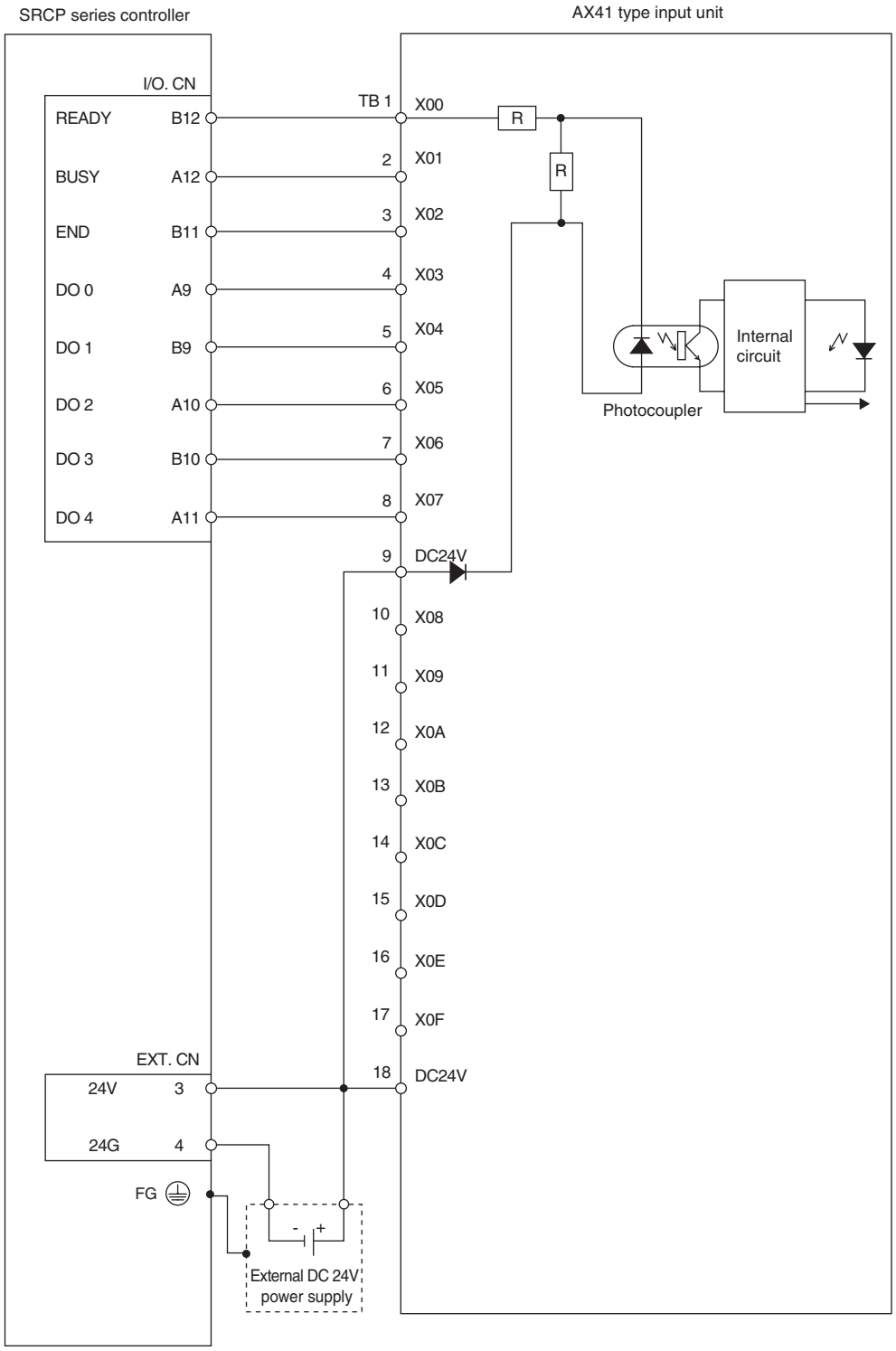
3-5-1 Connection to PLC output unit

Connection to the Mitsubishi© PLC AY51 output unit



3-5-2 Connection to PLC input unit

Connection to the Mitsubishi© PLC AX41 input unit

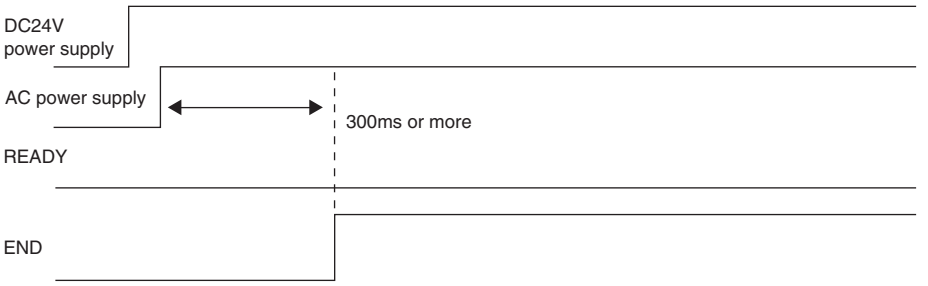


3-6 I/O Control Timing Charts

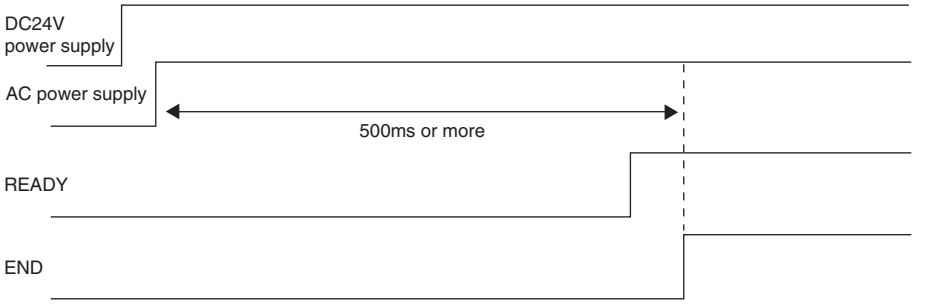
The following shows typical timing charts for I/O control. Refer to these diagrams when creating a sequence program.

3-6-1 When turning the power on

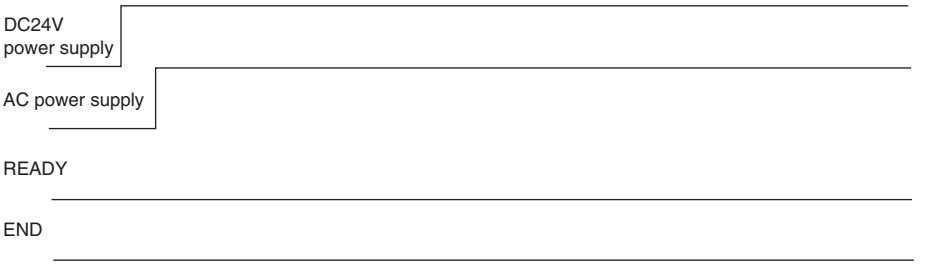
When emergency stop is triggered:



When emergency stop is canceled:



When an alarm is issued:



- The SRCP initial state depends on whether emergency stop is triggered when the power is turned on.
  When the power is turned on while emergency stop is cancelled, the SRCP controller starts with the READY signal and also the servo turned on. (Robot is ready to operate in this state.)
  In contrast, when the power is on while emergency stop is triggered, the SRCP controller starts with the READY signal turned off under emergency stop conditions. (Robot operation is prohibited in this state.)
  To enable robot operation, cancel the emergency stop to turn on the READY signal, and then input a servo recovery command (SERVO).
- After turning the power on, make sure that the END signal is on before inputting a dedicated command.
- If the READY and END signals are still off for more than the specified time after turning the power on, this means that an alarm has occurred. If that happens, correct the problem while referring to "13-2 Alarm and Countermeasures".
- Before supplying AC power to the power supply terminal block, DC 24V power to EXT. CN must be supplied.

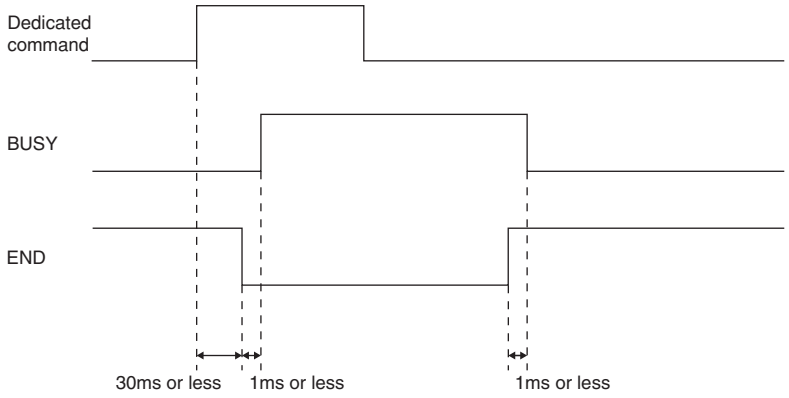
⚠ ***CAUTION***
DC 24V power to EXT. CN must first be supplied before supplying AC power to the power supply terminal block. If this order is reversed, an alarm (06: 24V POWER OFF) might be issued.

3-6-2 When executing a dedicated input command

■ The BUSY signal turns on when a dedicated command is received. Whether the received command has ended normally can be checked with the END signal status at the point that the BUSY signal turns off. When the END signal is on, this means that the command has ended normally. If it is off, the command has not ended normally.

■ The dedicated command input must be a pulse input. If the dedicated command input stays on, the BUSY signal does not turn off even after the command has been executed.

(1)When a command with a long execution time runs and ends normally:
(Command execution is still in progress and the END signal is off when turning off (contact open) the dedicated command input.)

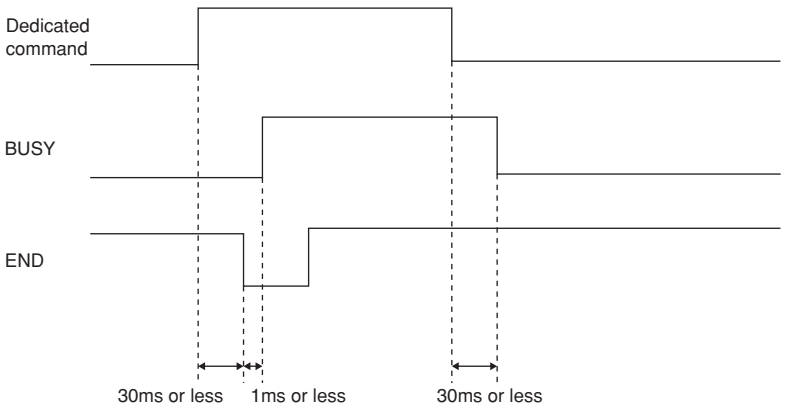


(1) At the rising edge of the dedicated command input, the END signal turns off and the BUSY signal turns on.
(2) Turn off (contact open) the dedicated command input after checking that the BUSY signal turns on.
(3) Wait until the BUSY signal turns off.
(4) The END signal should be on when the BUSY signal turns off, indicating that the command has ended normally.

⚠ ***CAUTION***
In the case of the automatic operation start command (AUTO-R), the END signal turns on and the BUSY signal turns off when the program ends or a STOP statement is executed. If an endless program (one that automatically returns to the top of the program from the last step) is executed, the BUSY signal will not turn off until an interlock or emergency stop is triggered.

(2)When a command with a short execution time runs and ends normally:
(Command execution has already ended and the END signal is on before turning off (contact open) the dedicated command input, as in the examples listed below.)

- A movement command (ABS-PT, INC-PT) for a very short distance was executed.
- A reset command (RESET) was executed.
- A step run was executed using a command with a very short execution time such as the L and DO statements.



(1) At the rising edge of the dedicated command input, the END signal turns off and the BUSY signal turns on.

(2) Turn off (contact open) the dedicated command input after checking that the BUSY signal turns on.

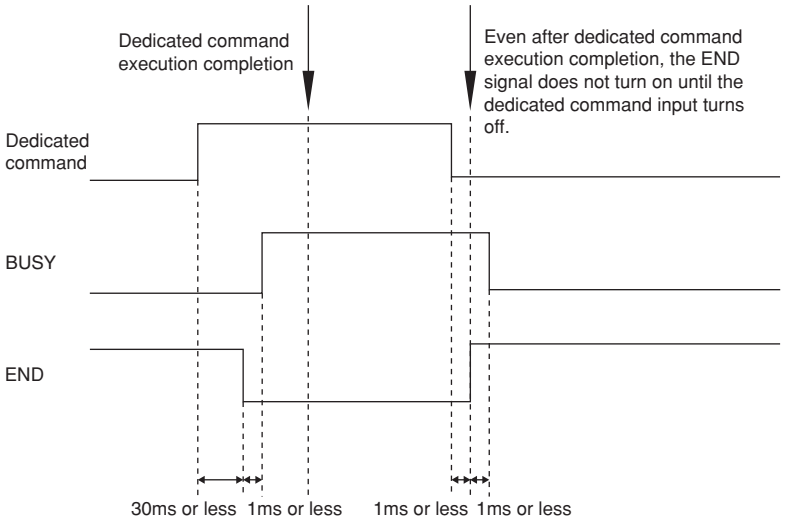
(3) Wait until the BUSY signal turns off. (The BUSY signal immediately turns off since the command execution is already complete.)

(4) The END signal should be on when the BUSY signal turns off, indicating that the command has ended normally.

However, the PRM34 (system mode selection parameter) "bit 7 END output sequence setting at command execution completion" setting can be changed so that the END signal turns on when the dedicated command input turns off.

NOTE

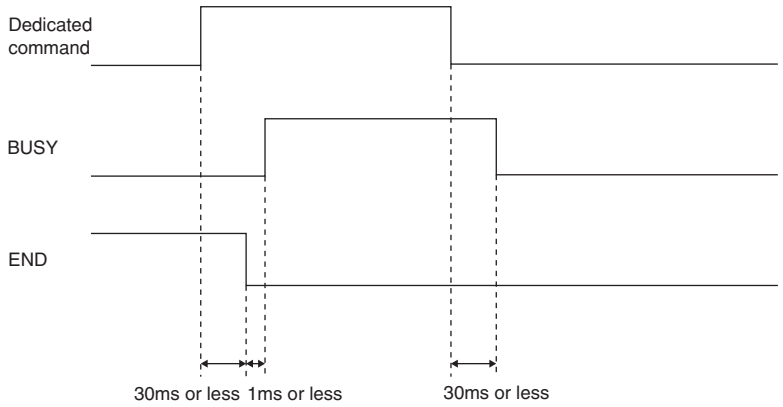
The PRM34 (system mode selection parameter) "bit 7 END output sequence setting at command execution completion" setting is supported only in Ver. 24.32 and later versions.



(3)When a command cannot be executed from the beginning:

(Command execution is impossible from the beginning and the END signal does not turn on, as in the examples listed below.)

- A movement command (ABS-PT, INC-PT) was executed without return-to-origin being completed.
- An operation start command (AUTO-R, STEP-R) was executed while return-to-origin is incomplete (except for cases where PRM48 (Pre-operation action selection parameter) is set to 1 or 3).
- A movement command (ABS-PT, INC-PT) was executed by specifying a point number whose point data is unregistered.
- A dedicated command was executed during interlock or emergency stop (except for the reset (RESET) and servo recovery (SERVO) commands).
- When a dedicated command input (ABS-PT, INC-PT, AUTO-R, STEP-R, ORG-S, SERVO, RESET) was executed in "SERVICE mode state".



(1) At the rising edge of the dedicated command input, the END signal turns off and the BUSY signal turns on.

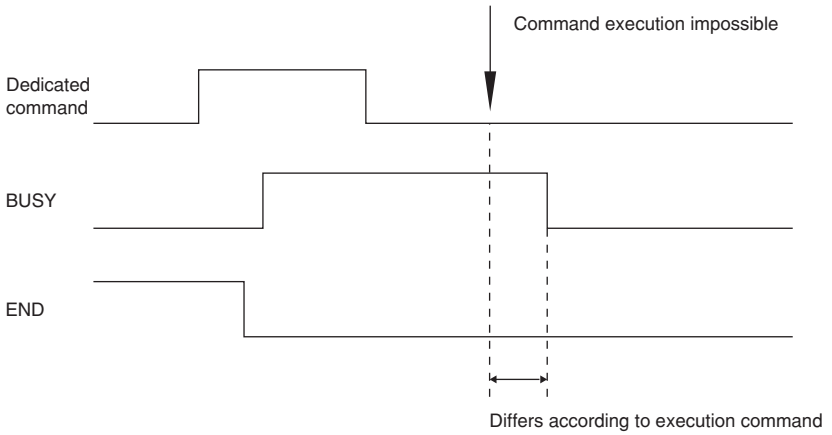
(2) Turn off (contact open) the dedicated command input after checking that the BUSY signal turns on.

(3) Wait until the BUSY signal turns off. (The BUSY signal immediately turns off since the command cannot be executed from the beginning.)

(4) The END signal remains off when the BUSY signal turns off, indicating that the command could not end normally.

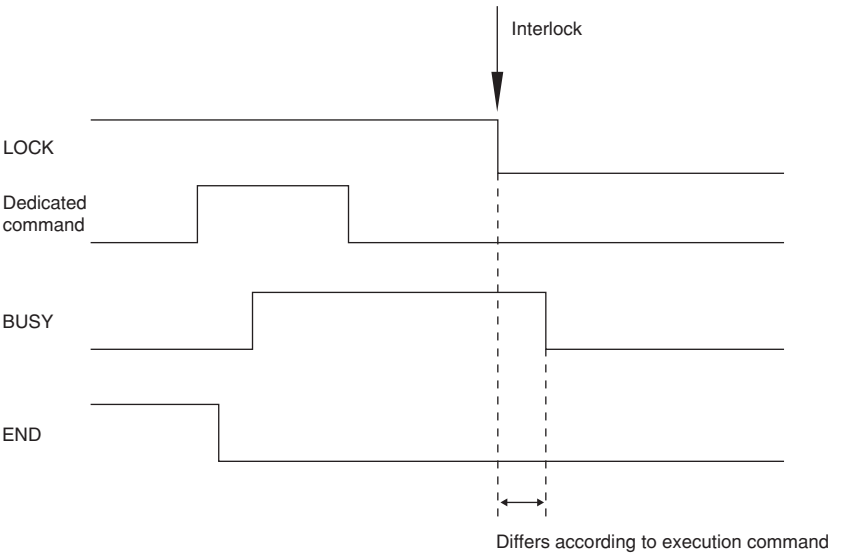
(4)When command execution cannot be completed:
(Command execution stops before completion and the END signal does not turn on, as in the examples listed below.)

- An interlock or emergency stop was triggered during execution of a dedicated command.
- The SERVICE mode input was changed during execution of a dedicated command.
- An error was caused due to a jump to an unregistered program or point during automatic operation.



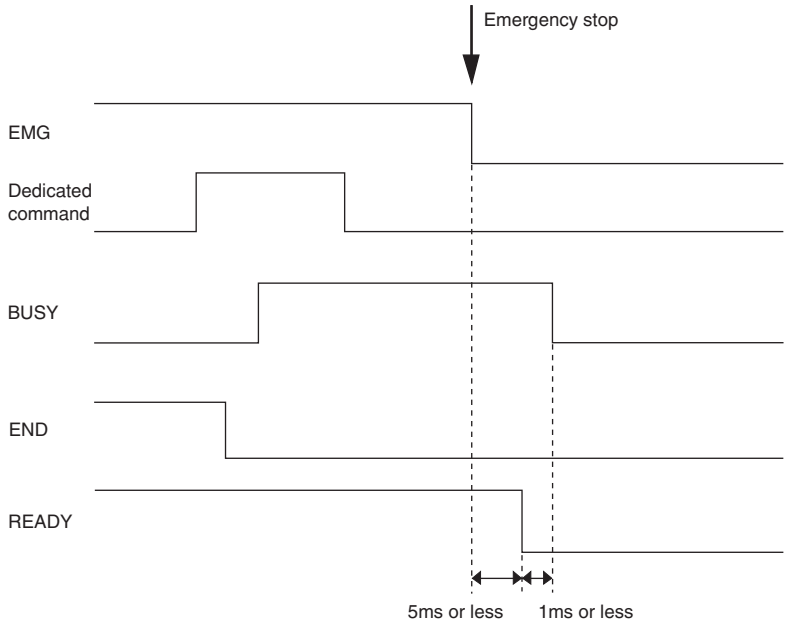
(1) At the rising edge of the dedicated command input, the END signal turns off and the BUSY signal turns on.
(2) Turn off (contact open) the dedicated command input after checking that the BUSY signal turns on.
(3) Wait until the BUSY signal turns off.
(4) The BUSY signal turns off since the command execution stops before completion.
(5) The END signal remains off when the BUSY signal turns off, indicating that the command could not end normally.

3-6-3 When interlock signal is input



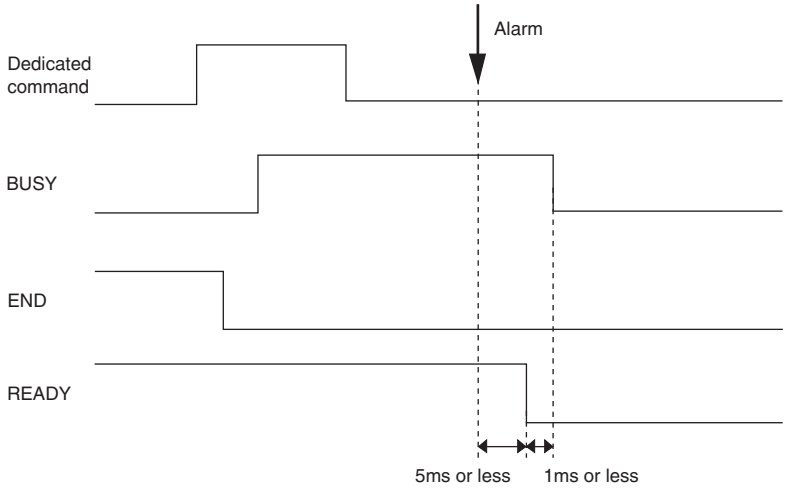
■ When a interlock signal is input while a dedicated command is being executed, the BUSY signal turns off. The READY and END signals remain unchanged.

3-6-4 When emergency stop is input



- The READY signal turns off. The BUSY signal also turns off while a dedicated command is being executed. The END signal remains unchanged.
- To enable robot operation, cancel emergency stop to turn on the READY signal, then input the servo recovery command (SERVO).

3-6-5 When alarm is issued

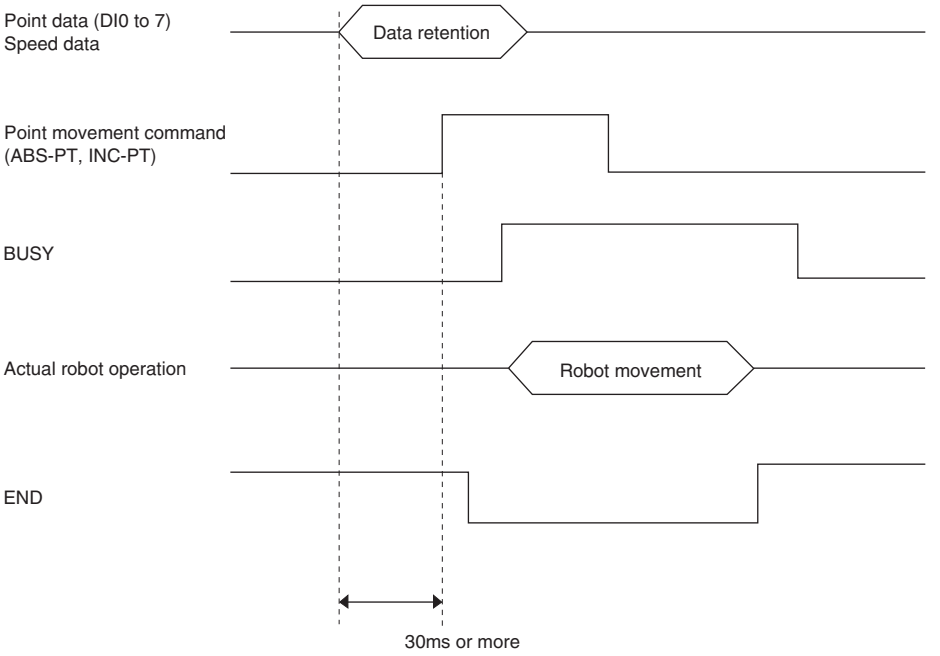


- The READY, BUSY and END signals all turn off.
- Correct the problem while referring to "13-2 Alarm and Countermeasures".

3-6-6 When executing a point movement command

■ When executing a point movement command (ABS-PT, INC-PT), the point data and speed data must first be input before inputting the command.

The point data and speed data can be specified with DI0 to DI7 (or DI0 to DI6 when SERVICE mode is enabled). Refer to "3-2-2 General-purpose input (DI0 to DI7)".



(1) Specify the point data and speed data, using the general-purpose input DI0 to DI7. These input conditions should be kept unchanged until the BUSY signal turns on. (If these conditions are changed before the BUSY signal turns on, then the data might be misrecognized.)

(2) When a minimum of 30ms has elapsed, input the point movement command (ABS-PT, INC-PT).

(3) At the rising edge of the dedicated command input, the END signal turns off and the BUSY signal turns on.

(4) Turn off (contact open) the dedicated command input after checking that the BUSY signal turns on.

Now, you may change the point data and speed data (DI0 to DI7) for the next movement.

(5) Wait until the BUSY signal turns off.

(6) The END signal should be on when the BUSY signal turns off, indicating that the command has ended normally.

NOTE

The number of general-purpose input (DI) points used to specify the point numbers and speed differs depending on whether SERVICE mode is enabled or disabled and also on the PRM7 (I/O point movement command speed parameter) setting. See "3-2-2 General-purpose input (DI0 to DI7)".

Chapter 4 BASIC OPERATION OF THE TPB

The TPB is a hand-held, pendant-type programming box that connects to the SRCP controller to edit or run programs for robot operation.
The TPB allows interactive user operation on the display screen so that even first-time users can easily operate the robot with the TPB. This chapter describes the basic operation of the TPB.
The TPB used with the SRCP series controller must be version 12.50 or later.

4-1 Connecting and Disconnecting the TPB

4-1-1 Connecting the TPB to the SRCP controller

⚠ ***CAUTION***

Do not modify the TPB cable or use any type of relay unit for connecting the TPB to the SRCP controller. Doing so might cause communication errors or malfunctions.

■ When the power supply to the controller is turned off

Connect the TPB connector to the connector labelled "TPB" on the front panel of the controller and supply power to the controller. A beep sounds for approximately 1 second and then the screen shown at the right appears. This screen is referred to as the "Initial screen" from this point onwards.

```
[MENU]
select menu


1EDIT2OPRT3SYS 4MON
```

■ When the power supply to the controller is turned on

The TPB can also be connected to the SRCP controller if the power supply to the controller is on. In this case, hold down the ESC switch on the front panel of the controller as you plug in the TPB connector. If the TPB is connected to the controller without pressing the ESC swith, emergency stop might be triggered causing the robot servo to turn off. Also, if the TPB is connected while the controller is executing a program or an I/O dedicated command, then the execution will be interrupted regardless of whether or not the ESC switch is held down.

⚠ ***CAUTION***

Any of the messages "08: PNT DATA DESTROY", "09: PRM DATA DESTROY" or "10: PGM DATA DESTROY" may appear on the TPB when the power to the controller is turned on. (See "13-2 Alarm and Countermeasures".) If one of these messages appears, turn off the power to the controller and then turn it back on again while the emergency stop button of the TPB is still depressed. In this state, the robot servo remains off, but the initial screen appears on the TPB to allow key operation, so initialize and restore the data.
If the message "05: BATT. LOW-VOLTAGE" appears on the TPB when the power is turned on, turn off the power to the controller and then turn it on again while the emergency stop button of the TPB is still depressed. In this state, the robot servo remains off, but the initial screen appears on the TPB to allow key operation, so make a backup of the data, and then replace the lithium battery in the controller (the lithium battery normally lasts five years). (See "14-2 Replacing the System Backup Battery".)
If the message "SIO error" is displayed on the TPB, check whether the I/O dedicated command input is on. If the dedicated command input is on, the TPB cannot be used, so the dedicated input must always be a pulse input (the dedicated command input must be off when the BUSY signal turns on.) (Refer to "3-2-1 Dedicated command input".)

4-1-2 Disconnecting the TPB from the SRCP controller

To disconnect the TPB from the controller while a program or an I/O dedicated command is being executed, pull out the TPB while holding down the ESC switch on the front panel of the controller. Failing to hold down the ESC switch will trigger emergency stop in the controller and turn off the servo.
When the TPB will be left disconnected from the controller for a long period of time, we recommend attaching the RS-232C connector dust cover (supplied) to the TPB connector on the controller.

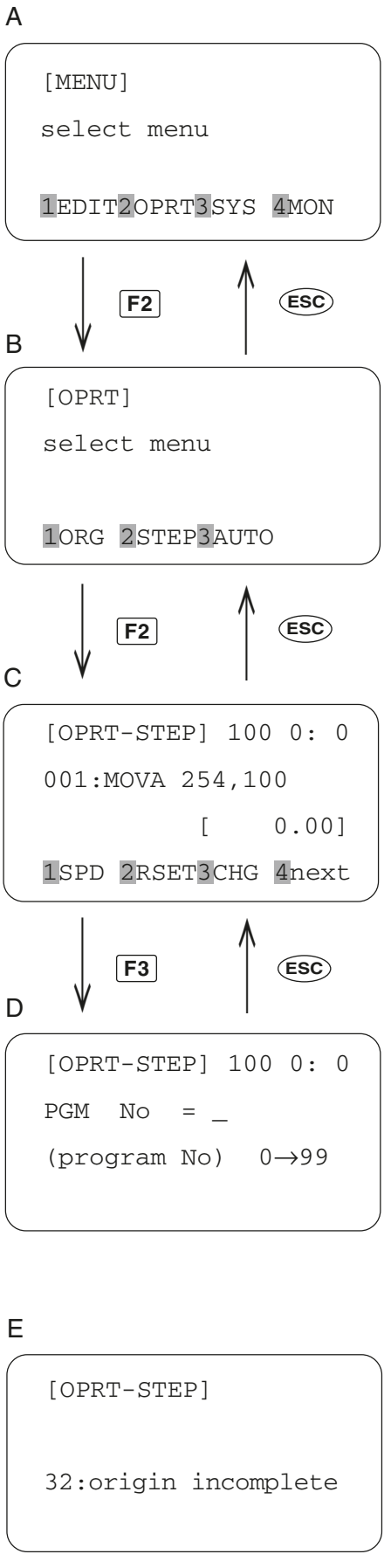
4-2 Basic Key Operation

1) Selectable menu items are displayed on the 4th line (bottom line) of the TPB screen.

   Example A is the initial screen that allows you to select the following modes.

   1 EDIT
   2 OPRT
   3 SYS
   4 MON

   The number to the left of each mode corresponds to the function keys from F1 to F4.



2) On the initial screen shown in A, pressing a function key moves to a lower level in the menu hierarchy. (A→B→C→D)

   To return to the previous screen or menu level, press the ESC key. (See "4-4 Hierarchical Menu Structure" in this chapter.)

3) If an error occurs during operation, a buzzer sounds for approximately 1 second and an error message like that shown in Example E appears on the 3rd line of the screen. If this happens, check the contents of the error message and then press the ESC key. The error message will be cleared to allow continuing operation. To correct the error, refer to the message tables in Chapter 12.

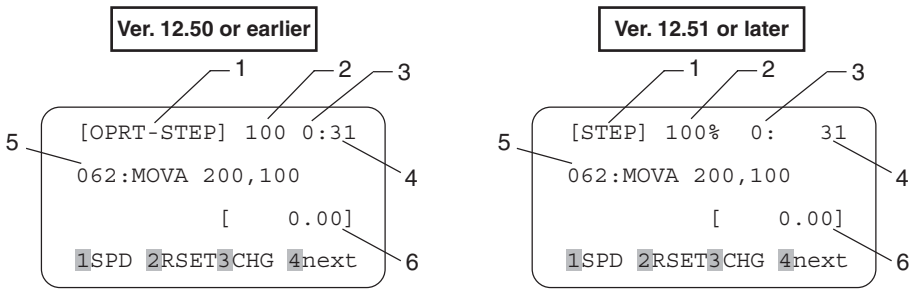
4) If an alarm occurs during operation, its alarm message appears on the 3rd line of the screen and a buzzer keeps sounding. The TPB cannot be used in this state. Turn off the power to the controller and then correct the problem by referring to "13-2 Alarm and Countermeasures".

4-3 Reading the Screen

The following explains the basic screen displays and what they mean.

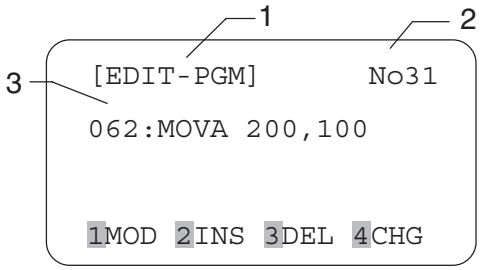
4-3-1 Program execution screen

The display method slightly differs depending on the version of TPB.



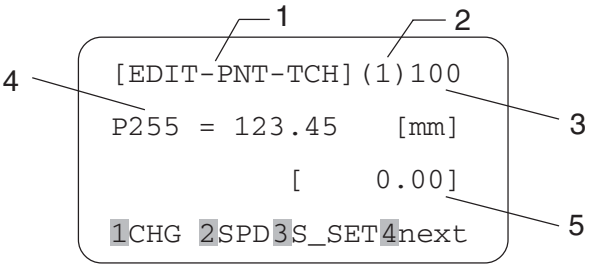
1. Current mode
2. Execution speed
3. No. of task being executed
4. No. of program being executed
   * On TPB version 12.51 or later, when switched from the lead program to another program, this area shows the program numbers as the "currently executed program / lead program".
5. No. of step being executed
6. Current position

4-3-2 Program edit screen



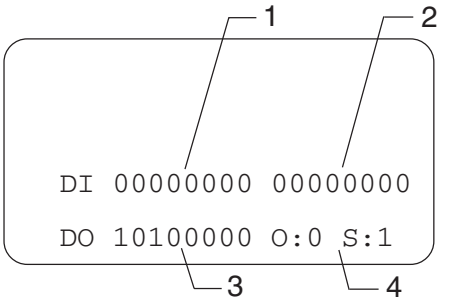
1. Current mode
2. No. of program being edited
3. No. of step being edited

4-3-3 Point edit screen (teaching playback)



1. Current mode
2. Speed selection number
3. Speed parameter (%)
4. Edit point number
5. Current position

4-3-4 DIO monitor screen



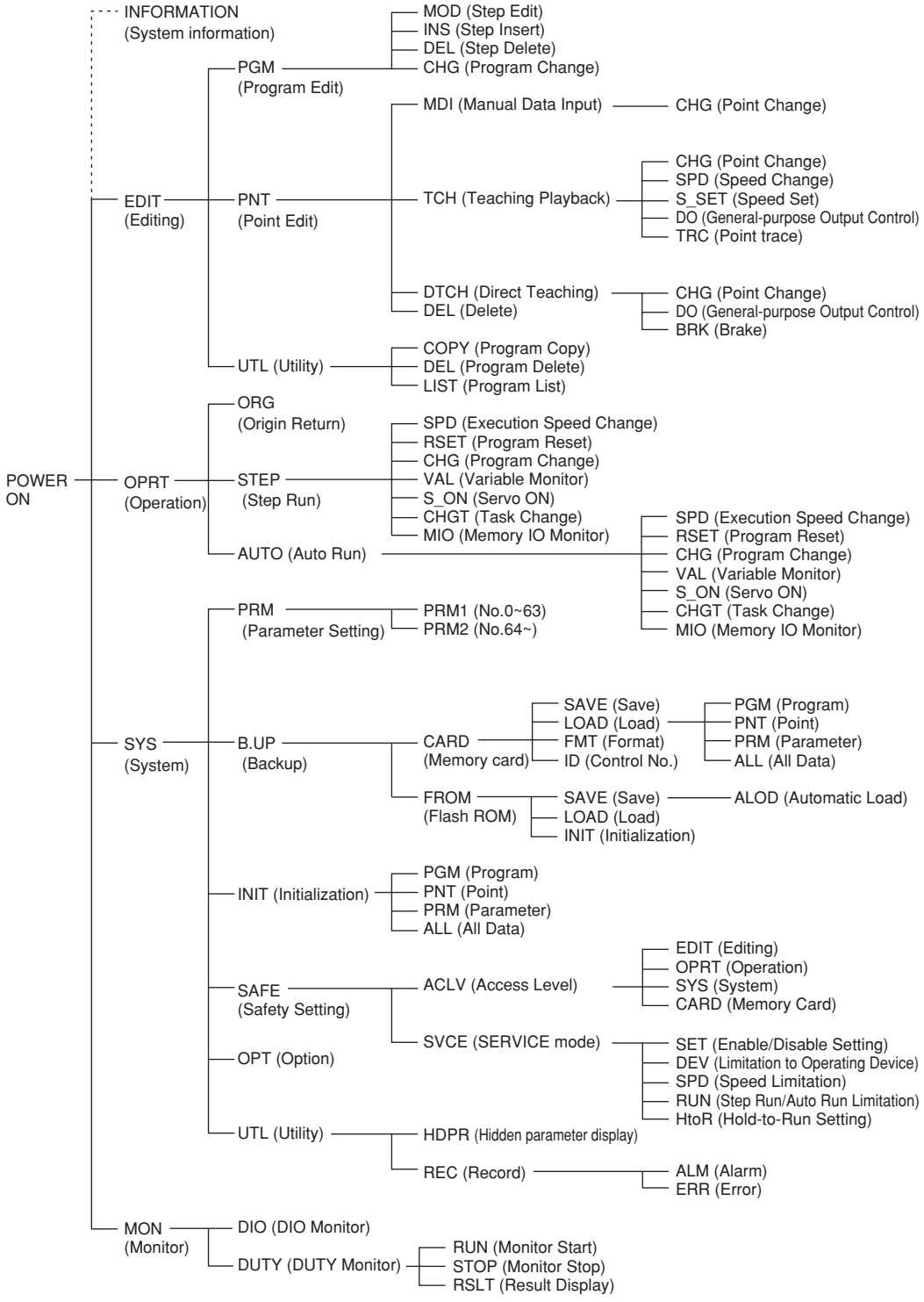
1. General-purpose input
   From left
   DI7 to DI0

2. Dedicated input
   From left
   Interlock (LOCK)
   0: Locked state (robot movement not possible)
   1: Unlocked state (robot movement possible)
   Return-to-origin command (ORG-S)
   Reset command (RESET)
   Automatic operation start command (AUTO-R)
   Step operation start command (STEP-R)
   Absolute point movement command (ABS-PT)
   Relative point movement command (INC-PT)
   Servo recovery command (SERVO)

3. Dedicated and general-purpose outputs
   From left
   READY, BUSY, END, DO4 to DO0

4. Origin sensor status and servo status
   From left
   O: Origin sensor status
   0: Off (Closed)
   1: On (Open)
   S: Servo status
   0: Servo off
   1: Servo on

4-4 Hierarchical Menu Structure



The menu hierarchy slightly differs depending on the versions of the controller and TPB.

4-5 Restricting Key Operation by Access Level

The TPB key operations can be limited by setting the access levels (operation levels).
A person not trained in robot operation might accidentally damage the robot system or endanger others by using the TPB incorrectly. Set the access levels to restrict TPB key operations and prevent such accidents.

NOTE
The access level settings are protected by a password so that changes cannot be instantly made.

4-5-1 Explanation of access level

The access levels can be set individually for editing, operation, system and memory card.
The details of the key operations limited at each level are explained below.

Editing

Level	Description
0	All operations are permitted.
1	Program editing is prohibited. (Program data can be checked.)
2	In addition to Level 1, point data editing, manual release of brake and point trace (movement to registered data point) are prohibited. (The [Σ-] and [Σ+] keys can be used to move the robot and general-purpose outputs can be controlled.)
3	Any operation in EDIT mode is prohibited. (Cannot enter EDIT mode.)

Operation

Level	Description
0	All operations are permitted.
1	Changing the execution speed and program is prohibited.
2	In addition to Level 1, automatic operation, step operation and program reset are prohibited. (Return-to-origin can be performed and variables can be monitored.)
3	Any operation in OPRT mode is prohibited. (Cannot enter OPRT mode.)

System-related data

Level	Description
0	All operations are permitted.
1	Initialization is prohibited.
2	In addition to Level 1, changing the parameters and setting the option units are prohibited. (Parameter data and option unit settings can be checked.)
3	Parameter editing, initialization and option setting are prohibited. (Cannot enter SYS-PRM, SYS-INIT and SYS-OPT modes.)

Memory card

Level	Description
0	All operations are permitted.
1	Loading the parameters and all data to the SRCP is prohibited. (Point data or program data can be loaded.)
2	Loading any data to the SRCP is prohibited. (Data can be saved and the memory card formatted.)
3	Use of memory card is prohibited. (Cannot enter SYS-B.UP mode.)

4-5-2 Changing an access level

1) Press F3 (SYS) on the initial screen.

```
[MENU]
select menu


1EDIT2OPRT3SYS 4MON
```

2) Press F4 (next) to switch the menu display and then press F1 (SAFE).

```
[SYS]
select menu


1SAFE2OPT 3UTL 4next
```

3) When the password entry screen appears, enter the password and press ➡.

```
[SYS-SAFE]
 Password: 24.00_
input password
```

4) When the password is accepted, the screen shown on the right appears.

   Press F1 (ACLV) here.

```
[SYS-SAFE]
select menu


1ACLV2SVCE
```

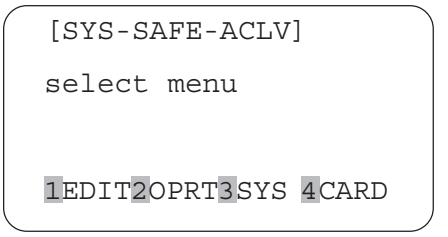
5) Select the item you want to change.

To change the access level for editing, press **F1** (EDIT).

To change the access level for operation, press **F2** (OPRT).

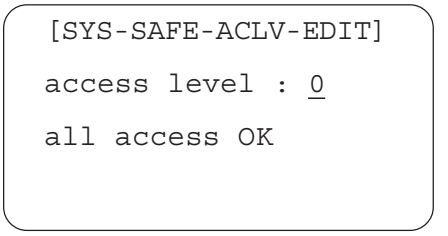
To change the access level for system-related data, press **F3** (SYS).

To change the access level for memory card, press **F4** (CARD).



6) The currently set access level appears.

To change this setting, use the number key to enter the access level and then press ➡.

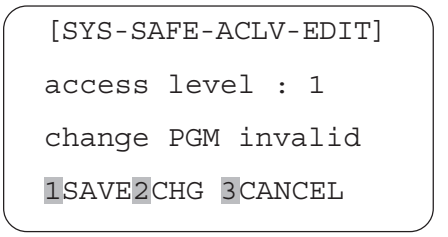


7) When the access level has been changed, the memory write screen appears.

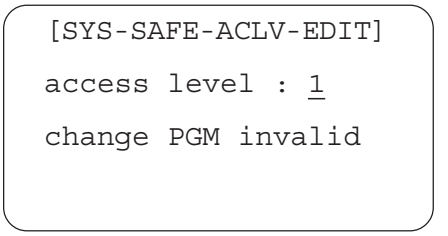
To save the change permanently (retain the change even after the controller power is turned off), press **F1** (SAVE).

To save the change temporarily (retain the change until the power is turned off), press **F2** (CHG).

To cancel changing of the setting, press **F3** (CANCEL).



8) When writing is complete, the screen returns to step 6.



NOTE

The password is identical to the SRCP controller's version number. For example, if the controller version is 24.00, enter 24.00 as the password. Once the password is accepted, it will not be requested unless the TPB is disconnected from the controller or the controller power is turned off.

NOTE

To avoid access level conflict between operation and others, the access levels may be automatically adjusted. For example, if the access levels related to editing, system and memory card are "0", they are automatically changed to "1" when the operation-related access level is "1" or "2" or "3". The access levels remain unchanged if they are "1" or "2" or "3".

Chapter 5 PARAMETERS

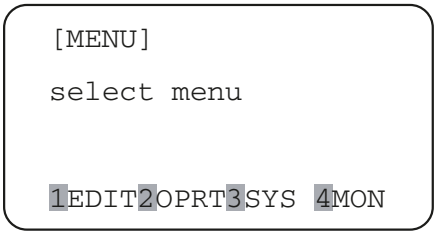
The SRCP controller uses a software servo system, so no adjustment of hardware components such as potentiometers or DIP switches are required. Instead, the SRCP controller uses parameters that can be easily set or changed by the TPB or PC (personal computer).
This chapter contains a detailed description of each of the parameters, and explains how to use the TPB to change and specify parameter settings.

SAFETY

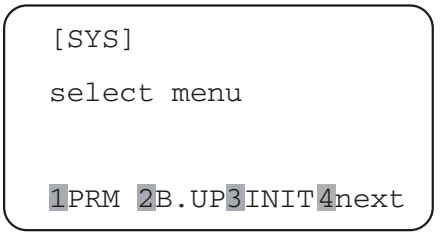
Errors such as motor overload are detected by the software, so the controller parameters must be set correctly to match the connected robot model. The parameters are initialized to match the robot model when the robot is shipped, so confirm them before starting use. If there is any trouble, please contact our sales office or sales representative.

5-1 Setting the Parameters

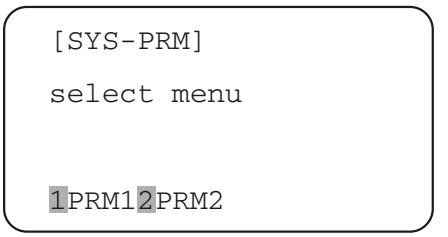
1) On the initial screen, press F3 (SYS).



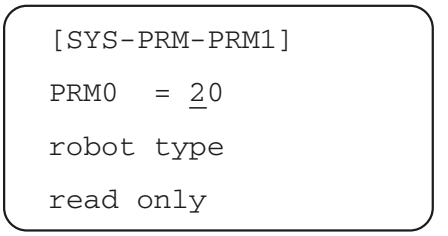
2) Next, press F1 (PRM).



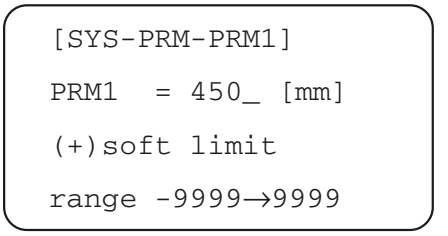
3) Select the parameter group you want to edit.
When editing PRM0 to PRM63, press F1 (PRM1).
When editing PRM64 onward, press F2 (PRM2).



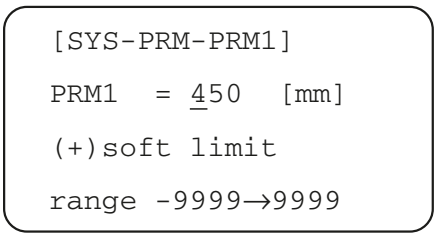
4) The current PRM0 (robot type number) setting appears on the screen. (The PRM64 setting appears when F2 (PRM2) was pressed in step 3.) Use the STEP UP and STEP DOWN keys to scroll the parameters until you find the parameter you want to set.



5) When the desired parameter is displayed, enter the new value with the number keys and then press ➡.



6) When the setting is complete, the cursor moves back to the beginning of the parameter data.



5-2 Parameter Description

The parameters are described in order below.

⚠ ***CAUTION***

Parameters not displayed on the TPB screen are automatically set or optimized to match the robot type when the robot parameters are initialized. You usually do not have to change these parameter settings.
If for some special reason you need to change or check these hidden parameters, use any of the following methods.

- ***Turn on the power to the controller while holding down the (ESC) key on the TPB.***
- ***Connect the TPB to the controller while holding down the (ESC) key on the TPB.***
- ***Use the system utility mode that allows you to display hidden parameters. (See "10-5-1 Viewing hidden parameters".)***

Take extra caution when handling hidden parameters.

PRM0: Robot type number

This parameter shows the robot number currently used. (See "15-1-2 Robot number list".) This is a read-only parameter. When changing the robot number or if the memory contents are corrupted, perform parameter initialization. (See "10-1 Initialization".)

PRM1: (+) soft limit

The + side robot movement range is set.
Set a suitable value for safety purposes.

Input range: -9999 to 9999 (mm)
Default value: Depends on robot type.

⚠ ***CAUTION***

The soft limit will not work unless return-to-origin has been completed.

PRM2: (-) soft limit

The - side robot movement range is set.
Set a suitable value for safety purposes.

Input range: -9999 to 9999 (mm)
Default value: Depends on robot type.

⚠ ***CAUTION***

The soft limit will not work unless return-to-origin has been completed.

PRM3: Payload

This specifies the total weight of the workpiece and tool attached to the robot. In cases where this weight varies, enter the maximum payload.
Based on this parameter, the controller determines the optimum acceleration speed for the robot, so ensure that the correct payload is set. If set too small, abnormal vibration or overheat may occur resulting in troubles with the robot or controller. Conversely, if this parameter is larger than the actual payload, a loss of the cycle time occurs which lowers productivity.

Input range: Depends on robot type. Units are in kilograms (kg).
Default value: 0

* This parameter is set to maximum payload when the controller is shipped from factory.

PRM4: Acceleration

This parameter sets the acceleration.
The controller will automatically set optimum acceleration according to the robot type and payload. If you want to increase the initial acceleration setting, manually enter the proper value by referring to the robot user's manual.

Input range: 1 to 200 (%)
Default value: 100

⚠ ***CAUTION***
If acceleration is too rapid, problems such as abnormal vibrations and reduced service life of the robot might result.

PRM5: Return-to-origin direction

This parameter sets the return-to-origin direction.
Normally, return-to-origin is performed toward the L side when this parameter is set to 1, and toward the R side when set to 1. (See our robot product catalog to check the L and R sides.)

Input range: 0 or 1
Default value: Depends on robot type.

PRM6: Positioning-completed pulse

This specifies the range in which the controller determines that positioning is complete.
When a movement command is executed, the robot moves toward the target position. The controller then determines that the positioning has been completed when the remaining distance to the target position is within this parameter setting. However, the robot continues moving until it reaches the target position even after the robot enters the "positioning-completed pulse" range.
Since executing the next movement command is not allowed until the positioning is complete, setting a large value for this parameter can reduce cycle time in cases where critical positioning accuracy is not required.

Input range: 1 to 4000 (pulses)
Default value: 80

* If the range specified by this parameter is larger than the range of the OUT valid position, the controller does not decide that the "positioning-completed pulse" range is entered until the axis reaches the OUT valid position.

PRM7: I/O point movement command speed

This parameter sets the movement speed to execute a point movement command (ABS-PT, INC-PT) and also determines the number of points that can be used with a point movement command. (See "3-2-2 General-purpose input (DI0 to DI7)".)

Input range: 0 to 100 (%)
Default value: 100

PRM8: No. of conditional input points

This parameter specifies the number of effective points for the third data conditional input for executing the JMPF statement of the robot language.
For example, when the default setting is selected for this parameter, the four points from DI0 to DI3 are used as the conditional inputs for the JMPF statement.

Input range: 1 to 8 (points)
Default value: 4

No. of conditional input points versus general-purpose input and setting range

No. of conditional input points	General-purpose input	Setting range
1	DI0	0 to 1
2	DI0 to DI1	0 to 3
3	DI0 to DI2	0 to 7
4	DI0 to DI3	0 to 15
5	DI0 to DI4	0 to 31
6	DI0 to DI5	0 to 63
7	DI0 to DI6	0 to 127
8	DI0 to DI7	0 to 255

NOTE
When SERVICE mode is enabled, DI7 functions as a service mode input (SVCE). Because of this, the DI7 status will be the same as SERVICE mode input (SVCE) status when the number of conditional input points is 8.

PRM9: MOVF speed

This sets the speed at which the robot moves when the program language MOVF statement is executed.

Input range: Depends on robot type. (mm/sec)
Default value: 10

PRM10: Return-to-origin speed

This specifies the movement speed during return- to-origin.

Input range: 1 to 100 (mm/sec)
Default value: 20

CAUTION
When the return-to-origin speed is increased, an alarm might be issued during return-to-origin depending on the robot type. We recommend using the default value as much as possible.

PRM11: No. of encoder pulses (4× mode)

This parameter indicates the constant that is determined by the linear scale.

Default value: Depends on robot type.

PRM12: Lead length

This parameter indicates the constant that is determined by the linear scale.

Default value: Depends on robot type.

PRM13: Origin detection method

This parameter specifies the origin (reference point) detection method. The SRCP controller uses the stroke-end detection method.

Default value: 1 (Stroke-end detection method)

⚠ ***CAUTION***

The origin detection method is predetermined by the machine specifications. Do not change the default setting. If changed inadvertently, serious problems might occur with the robot and controller.

PRM14: Overload current

This sets the reference current value used to detect an overload.

Default value: Equal to the motor rated current.

PRM15: Overload time

This specifies conditions such as time required to detect an overload.
The default value is set so that an overload alarm is issued when a current three times higher than the overload current (PRM14) flows for a period of 3 seconds or an equivalent condition is detected.

Default value: 240

PRM16: Current limit

This sets the maximum motor input current.

Default value: Depends on robot type.

PRM17: Speed proportional gain

This sets the speed control gain. Typically, PRM17 and PRM18 should be input at a ratio of 3 : 2.
Generally, the larger the gain, the higher the acceleration will be. However, if the gain is set too high, abnormal oscillation or noise might be generated, causing serious problems in the robot and controller. Use caution when selecting this parameter to avoid such problems.

Default value: Depends on robot type.

PRM18: Speed integration gain

This sets the speed control gain. Typically, PRM17 and PRM18 should be input at a ratio of 3 : 2.
Generally, the larger the gain, the higher the acceleration will be. However, if the gain is set too high, abnormal oscillation or noise might be generated, causing serious problems in the robot and controller. Use caution when selecting this parameter to avoid such problems.

Default value: Depends on robot type.

PRM19: Position proportional gain

This sets the position control gain.
If this parameter is changed carelessly, serious problems may occur in the robot and controller.

Default value: Depends on robot type.

PRM20: OUT valid position

This specifies the range in which the controller determines that movement command is complete.
When a movement command is executed, the robot moves toward the target position. The controller then determines that the movement command has ended when the remaining distance to the target position is within this parameter setting. The controller then initiates the subsequent step processing when the robot reaches this OUT valid position, so setting this parameter to a larger value can reduce cycle time.
However, if the subsequent command is a movement command, it is not executed until the ongoing positioning is complete.

Input range: 0 to 9999 (mm)
Default value: 1

PRM21: Position data unit

This parameter sets the units in which point data is to be displayed. Do not change this parameter for the PHASER series. Always use the default value.

Default value: 0 (shown in mm)

⚠ ***CAUTION***
Do not change this parameter for the PHASER series.

PRM22: English/Japanese selection

This parameter sets the language for the response messages displayed on the TPB or handled by RS-232C communications.

Input range: 0 or 1
Meaning: 0: English
1: Japanese
Default value: 0

PRM23: Payload-dependent acceleration coefficient

The value calculated from PRM0, PRM12 and PRM3 is set automatically for this parameter.

Default value: Depends on robot type.

PRM24: Teaching count data (TPB entry)

This is entered in the TPB and cannot be used.

Default value: 0

PRM25: Not used

Default value: 0

PRM26: Teaching movement data

This parameter is used during movement with a communication command @X+ or @XINC. This is also used for point teaching playback.

Input range: 1 to 100 (%)
Default value: 100

PRM27: Teaching movement data 1 (for TPB)

This is entered in the TPB and cannot be used.

Input range: 1 to 100 (%)
Default value: 100

* The TPB writes the contents of PRM27 into PRM26 when connected to the controller.

PRM28: Teaching movement data 2 (for TPB)

This is entered in the TPB and cannot be used.

Input range: 1 to 100 (%)
Default value: 50

PRM29: Teaching movement data 3 (for TPB)

This is entered in the TPB and cannot be used.

Input range: 1 to 100 (%)
Default value: 10

PRM30: Maximum program speed

The speed data defined by the MOVA, MOVI and MOVM statements in a program is multiplied by this parameter value to determine the maximum speed at which the robot actually moves. This is used to lower the speed of the overall program. When the TPB is used, any speed changes in the AUTO and STEP modes will also change this parameter value.
Max. speed (%) = PRM30 × speed operand (%) of movement command / 100

Input range: 1 to 100 (%)
Default value: 100

PRM31: Mechanical lock detection level

This parameter sets the sensitivity to detect mechanical locking caused by collision of the robot with an object. The upper limit of this parameter is 254. The sensitivity becomes lower as the parameter value increases. Leave this parameter set to 255 when you want to disable this function.

Input range: 1 to 255 (.01 sec.)
Default value: 255 (This function is disabled.)

PRM32: Alarm number output

When an alarm is issued, this parameter selects whether the alarm number is to be output as a general-purpose output. When this parameter is set to 1, the alarm number is output as a 5-bit binary signal through DO0 to DO4.

Input range: 0 or 1
Meaning: 0: No output
1: Output
Default value: 0

Example of alarm Number - DO output

Alarm No.	Alarm Message	DO4	DO3	DO2	DO1	DO0
01	OVER LOAD	OFF	OFF	OFF	OFF	ON
02	OVER CURRENT	OFF	OFF	OFF	ON	OFF
03	OVER HEAT	OFF	OFF	OFF	ON	ON
04	POWER DOWN	OFF	OFF	ON	OFF	OFF
:	:	:	:	:	:	:
:	:	:	:	:	:	:
16	ABNORMAL VOLTGAE	ON	OFF	OFF	OFF	OFF
17	SYSTEM FAULT 2	ON	OFF	OFF	OFF	ON
18	FEEDBACK ERROR 3	ON	OFF	OFF	ON	OFF
19	SYSTEM FAULT 3	ON	OFF	OFF	ON	ON
:	:	:	:	:	:	:
:	:	:	:	:	:	:

* For more details on the alarm No. and contents, refer to "13-2-2 Alarm message list".

PRM33: Operation at return-to-origin complete

Selects the operation to be executed simultaneously with completion of return-to-origin. A signal can be output as a general-purpose output indicating that return-to-origin has been completed or to reset the program.

Input range: 0 to 3
Meaning: 0: Nothing is executed
1: DO4 is turned on
2: Program reset is executed
3: DO4 turns on after program reset
Default value: 2

* When this parameter is set to 1 or 3, DO4 is not affected by program reset (in other words, DO4 does not turn off even when the program is reset). If you want to turn off DO4 after return-to-origin is complete, use the program command to execute DO 4,0 or manually operate the general-purpose output by using the TPB.
(See "7-4 Manual Control of General-Purpose Output".)

PRM34: System mode selection

This parameter specifies the system operation mode. When you want to use the SRCP series in operating specifications that differ from normal mode, change this parameter as explained below.
This parameter functions are allocated in bit units.

Input range: 0 to 255
Default value: 0

Function allocation in bit units

Bit	Function	Selected operating mode	Setting	Addition value
0	Reserved for system use		0	0
1	READY signal sequence setting	ON when emergency stop is canceled. (DRCA compatible)	0	0
		ON when servo is ON. (SRCA compatible)	1	2
2	END signal sequence setting when the controller has started normally	ON after controller has started normally.	0	0
		OFF after controller has started normally.	1	4
3	Voltage check setting for system backup battery	Check	0	0
		No check	1	8
4 to 5	Reserved for system use		0	0
6	Interlock function setting	Enable	0	0
		Disable	1	64
7	END output sequence setting at command execution completion	ON at normal command completion	0	0
		ON at command signal OFF at normal command completion	1	128
8 to 15	Reserved for system use		0	0

Example: To turn off the END output sequence after the controller has started normally, and disable the interlock function:
PRM34 should be set to "68" because of 0000000001000100 (binary)=68 (decimal)

Bit	15 to 8	7	6	5 to 4	3	2	1	0	PRM34
Setting	0	0	1	0	0	1	0	0	
Addition value	0	0	64	0	0	4	0	0	64+4=68

Bit 1: READY signal sequence setting

This selects whether to set the READY signal sequence compatible with the DRCA or SRCA controller.
In DRCA compatible mode, the READY signal turns on at the instant that emergency stop is released. In the SRCA compatible mode the READY signal turns on when the servo is turned on. (The READY signal will not turn on just by releasing emergency stop.)

Bit 2: END signal sequence setting when the controller has started normally

This selects whether to turn on the END signal when the controller has started normally.
In normal mode, the END signal turns on when the controller has started normally.
In conventional compatible mode, the END signal remains off even when the controller has started normally.

Bit 3: Voltage check setting for system backup battery

This selects whether to check the system backup battery voltage when the controller servo is turned on.
In such cases where you want to operate the robot immediately even when the battery needs to be replaced, you can temporarily disable this voltage check.
(System backup batteries are located inside the controller and used to back up the parameters and point data.)

Bit 6: Interlock function setting

This selects whether to enable or disable the interlock function. The interlock function is enabled by default. If the interlock function is disabled, use caution and be aware of the robot movement.

Bit 7: END output sequence setting at command execution completion (supported by Ver. 24.32 and later versions):

This selects the END output sequence at dedicated command completion.
With the standard setting ("0"), the command's execution result is output to the END output when the command is completed. When set to "1", the command's execution result is output to the END output when the command is completed, but only after the command signal turns off.

PRM35: Origin shift

This parameter specifies a shift to the origin position after return-to-origin is complete. The origin position is usually "0" when return-to-origin is complete. If for some reason the origin position needs to be shifted from the "0" point, then change this parameter. For example, if an unwanted position shift occurred, then reteaching of all point data needs to be performed. However, the time and effort needed for this reteaching can be eliminated by setting the shift amount for this parameter to quickly correct the point data.

Input range: -9999 to 9999 (0.01mm)
Default value: 0

* The parameter change is enabled after reperforming return-to-origin.

PRM36: Origin search data

This specifies the performance data for detecting the origin position during return-to-origin by the origin search method.

Default value: Depends on robot type.

PRM37: QP band width

This parameter specifies the control switching point (pulse width) that compensates for the frictional resistance during deceleration.

Input range: 1 to 1000 (pulses)
Default value: Depends on robot type.

PRM38: Speed delay compensation gain

Default value: Depends on robot type.

PRM39: Control mode selection

Default value: Depends on robot type.

PRM40: RESET execution condition selection

Selects the operation to be executed with the I/O reset command.

Input range: 0 to 2
Meaning: 0: Turns on the servo and resets the program.
1: Switches the operation depending on the LOCK signal status.
When OFF (interlocked), only the servo is turned on.
When ON, the servo is turned on and the program is reset.
2: Resets only the program.
Default value: 2

PRM41: I/O point movement command speed 1

This parameter sets the speed at which the robot moves when a point movement command (ABS-PT, INC-PT) is executed. The speed set here is the movement speed used in normal mode (SERVICE mode disabled) with PRM7set to 0, DI6 turned on and DI7 turned off.

Input range: 1 to 100 (%)
Default value: 10

* The actual speed is the speed obtained by multiplying the execution speed displayed on the AUTO or STEP mode by this parameter (see "4-3-1 Program execution screen").
  Example: When the execution speed displayed in AUTO or STEP mode is 50 and this parameter is set to 10, the actual speed will be 2000mm/sec. × (50/100) × (10/100) = 100mm/sec. (when PRM44=2000)

PRM42: I/O point movement command speed 2

This parameter sets the speed at which the robot moves when a point movement command (ABS-PT, INC-PT) is executed. The speed set here is the movement speed used in normal mode (SERVICE mode disabled) with PRM7 set to 0, DI6 turned off and DI7 turned on.

Input range: 1 to 100 (%)
Default value: 30

* The actual speed is the speed obtained by multiplying the execution speed displayed on the AUTO or STEP mode by this parameter (see "4-3-1 Program execution screen").
  Example: When the execution speed display in the AUTO or STEP mode is 50 and this parameter is set to 30, the actual speed will be 2000mm/sec. × (50/100) × (30/100) = 300mm/sec. (when PRM44=2000)

PRM43: I/O point movement command speed 3

This parameter sets the speed at which the robot moves when a point movement command (ABS-PT, INC-PT) is executed. The speed set here is the movement speed used in normal mode (SERVICE mode disabled) with PRM7 set to 0, DI6 turned on and DI7 turned on.

Input range: 1 to 100 (%)
Default value: 70

* The actual speed is the speed obtained by multiplying the execution speed displayed on the AUTO or STEP mode by this parameter (see "4-3-1 Program execution screen").
  Example: When the execution speed display in the AUTO or STEP mode is 50 and this parameter is set to 70, the actual speed will be 2000mm/sec. × (50/100) × (70/100) =700mm/sec. (when PRM44=2000)

PRM44: Maximum speed setting

This parameter sets the maximum robot speed.

Input range: 1 to 2500 (mm/sec.)
Default value: Depends on robot type.

⚠ ***CAUTION***
Changing this parameter carelessly might shorten the robot service life or cause other problems.

PRM45: Feed forward gain

Default value: Depends on robot type.

PRM46: Servo status output

This parameter selects whether to output the axis servo status as a general-purpose output. When this parameter is set to 1, DO3 turns on and off along with servo on/off.

Input range: 0 or 1
Meaning: 0: Does not output the servo status.
1: Outputs the servo status.
Default value: 0

* When this parameter is set to 1, DO3 is not affected by program reset (in other words, DO3 does not turn off even when the program is reset).

PRM47: Communication parameter setting

This sets communication parameters used for data transmission through RS-232C. For more details, see "11-1 Communication Parameter Specifications".

Default value: 0

PRM48: Pre-operation action selection

This parameter checks whether return-to-origin has been performed or resets the program before running automatic operation or step operation.
When set to 0 or 2, an error (return-to-origin incomplete) is issued if return-to-origin has not been performed and automatic operation and step operation are not accepted.
When set to 1 or 3, the program runs even when return-to-origin has not been performed. However, an error (return-to-origin incomplete) is issued when a movement command (MOVA, etc.) is executed if return-to-origin is still incomplete. To avoid this, perform return-to-origin in advance or insert the ORGN command into the program.

Input range: 0 to 3
Meaning: 0: Checks whether return-to-origin has been performed.
1: Nothing is executed.
2: Resets the program after checking return-to-origin.
3: Resets the program.
Default value: 1

* When set to 2 or 3, the program is reset only during automatic operation. (The program is not reset during step operation.)

PRM49: Controller version 1

This parameter reads out the version information (1) on the control software in the controller.
This is a read-only parameter.

PRM50: Deceleration

Use this parameter to reduce only the deceleration.
When this parameter is left set to the default value (100), the deceleration is the same as the acceleration. If vibration occurs during positioning, then set this parameter to a smaller value to reduce only the deceleration.
This parameter value can be changed in 1% steps, with 100% equal to the value determined by PRM4.

Input range: 1 to 100 (%)
Default value: 100

PRM51: Lead program number

This parameter sets the lead program number.

Default value: 0

NOTE

The lead program is the program that has been selected as the execution program by the TPB or POPCOM. (See "9-4 Switching the Execution Program".)
The lead program can also be selected by executing a communication command "@SWI". It may also be switched when the program data is loaded into the controller from the memory card.

PRM52: Hold gain

Default value: Depends on the robot.

PRM53: Zone output selection

This parameter is used to select the output destination and output logic when the zone output function is enabled. The zone output is used to control the signal output when the robot's current position is within the specified range.
A maximum of 4 zone outputs are available by setting for PRM53. The output logic can also be changed.
This parameter functions are allocated in bit units.

Input range: 0 to 255
Default value: 0

Function allocation in bit units

Bit	Function	Selected value	Addition value
0	Zone 0 output enable setting	0: Disabled	0
		1: Enabled	1
1	Zone 1 output enable setting	0: Disabled	0
		1: Enabled	2
2	Zone 2 output enable setting	0: Disabled	0
		1: Enabled	4
3	Zone 3 output enable setting	0: Disabled	0
		1: Enabled	8
4	Zone 0 output logic setting	0: Positive logic	0
		1: Negative logic	16
5	Zone 1 output logic setting	0: Positive logic	0
		1: Negative logic	32
6	Zone 2 output logic setting	0: Positive logic	0
		1: Negative logic	64
7	Zone 3 output logic setting	0: Positive logic	0
		1: Negative logic	128
8 to 15	Reserved for system use		0

Example: To set zone 1 output to positive logic and zone 2 output to negative logic while enabling zone 1 output and zone 2 output, make the following settings.
PRM53 should be set to "70" because of 0000000001000110 (binary)=70 (decimal).

Bit	15 to 8	7	6	5	4	3	2	1	0	PRM53
Selected value	0	0	1	0	0	0	1	1	0	
Addition value	0	0	64	0	0	0	4	2	0	64+4+2=70

Zone output function

To use the zone output function, the desired zone must be specified with point data. (See Chapter 7, "EDITING POINT DATA".) When the robot enters the specified zone, its result is output to the specified port. Point numbers and output port that can be used for each zone output are listed below.

Zone setting range and output port

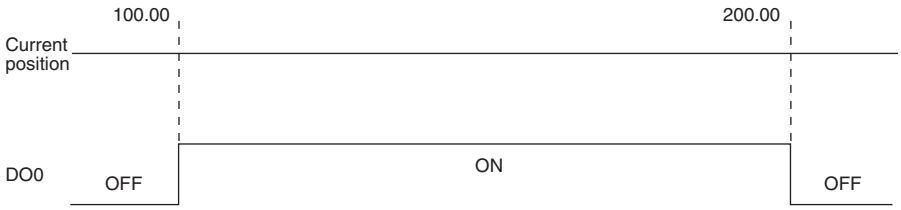
Zone No.	Specified range	Output port
Zone 0	P900-P901	DO0
Zone 1	P902-P903	DO1
Zone 2	P904-P905	DO2
Zone 3	P906-P907	DO3

⚠ ***CAUTION***

The zone output function does not work if one item of the point data is unspecified or return-to-origin is incomplete.

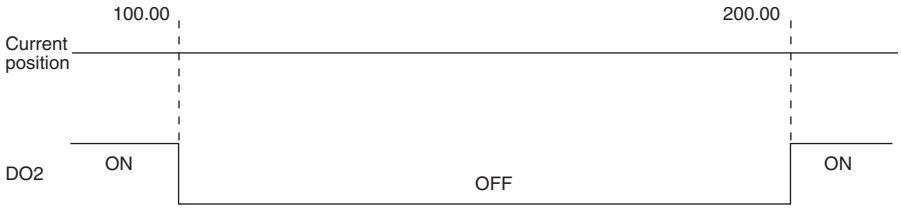
Example ①

PRM53=1 (Zone 0 output enabled, positive logic output)	
P900=100.00	P901=200.00



Example ②

PRM53=68 (Zone 2 output enabled, negative logic output)	
P904=100.00	P905=200.00



PRM54: Magnetic pole detection level

Default value: Depends on the robot.

PRM55: Magnetic pole position

Default value: 0

PRM56: Controller version 2

This parameter reads out the version information (2) on the control software in the controller.
This is a read-only parameter.

PRM57: Servo braking selection (available for version 24.15 or later)

This parameter is used to select the servo braking method.
When emergency stop is triggered while the robot is moving, the servo braking function immediately reduces the robot speed as much as possible before the servo turns off, so that the braking time and distance can be shortened.
Besides the case of emergency stop, this function is also effective when an overload or 24V power-off alarm occurs or when the power is cut off.

Default value: 2

⚠ ***CAUTION***
Do not change the setting.

PRM58: Not used

Default value: 0

PRM59: Not used

Default value: 0

PRM60 to 63: Spare

PRM64 to 99: Data area for pulse trains

Chapter 6 PROGRAMMING

In this chapter we will try programming some operations. First, you will learn how to enter a program using the TPB programming box.

6-1 Basic Contents

6-1-1 Robot language and point data

The SRCP controller uses the YAMAHA robot language that is very similar to BASIC. It allows you to easily create programs for robot operation.
In programs created with the YAMAHA robot language, the robot position data (absolute position, amount of movement) are not expressed in terms of direct numeric values. Instead, point numbers are used to express the position data indirectly. Point numbers and their corresponding position information are stored as point data separately from programs. This means that when you want to change only the position information while using the same program, all that you have to do is edit the point data.

Example

	Program	Point Data
	:	
005:	MOVA 0, 100	P0 = 50.00
006:	MOVI 1, 50	P1 = 100.00
	:	

In the above example, the robot first moves to a position (P0) 50mm from the origin point, and then moves to another point (P1) 100mm away from that position.

To change the above operation so that the robot first moves to a position (P0) 50.5mm from the origin point and then moves to another point (P1) 100mm away from that position, just change the P0 point data to P0=50.50.

6-1-2 Using the TPB to enter the robot language

Robot language commands frequently used to create programs are printed on the lower part of each number key on the TPB. When creating or editing a program, you can enter robot language commands simply by pressing these keys. To select other robot language commands not printed on these keys, use the function key matching that command.
During program editing, you can enter numbers (numerical values) with the number keys except when the edit cursor for robot language command input appears on the TPB screen.

6-1-3 Program specifications

The SRCP controller has the following memory capacity:

Total number of programs	: 100 programs (NO0 to NO99)
Max. number of steps per program	: 255 steps
Max. number of steps in all programs together	: 3000 steps
Max. number of points	: 1000 points (P0 to P999)

6-2 Editing Programs

"Program editing" refers to operations such as creating a program right after initialization, creating a new program, changing an existing program, and deleting or copying a program. In this section, you will learn the basic procedures for program editing using the TPB.

"Creating a program right after initialization" means creating a program for the first time after purchasing the controller or creating a program right after initialization while there are still no programs stored in the controller (see "10-1 Initialization").
"Creating a new program" means creating or editing a new program while at least one program has already been created and stored.
"Changing an existing program" means correcting, adding, deleting, or inserting steps in a program to change only part of it.

This section explains all the above program editing procedures, and also describes how to view program information such as the number of steps left in a program.

6-2-1 Creating programs after initialization

1) On the initial screen, press [F1] (EDIT).

```
[MENU]
select menu


1EDIT2OPRT3SYS 4MON
```

2) Next, press [F1] (PGM).

```
[EDIT]
select menu


1PGM 2PNT 3UTL
```

3) Since no program is registered after initialization, an error message appears on the screen, indicating that no program exists.

```
[EDIT]
select menu
43:cannot find PGM
1PGM 2PNT 3UTL
```

4) Press the (ESC) key to reset the error. A confirmation message then appears asking whether to create a new program as program No. 0.
To select and edit program No. 0, press [F1] (yes).
To select and edit a program other than No. 0, press [F2] (no).

```
[EDIT-PGM]
PGM  No = 0
New entry OK ?
1yes 2no
```

5) When you selected [F2] (no) in step 4, enter the number of the program to be edited with the number keys and press ➡. The screen returns to step 4. Make sure the program number is correct and press [F1] (yes).

```
[EDIT-PGM]
PGM  No = _
(Program No) 0→99
```

6) Select [F1] to [F3] or a robot language command shown on the lower part of each number key.

To change the robot language menu display, press [F4] (next). To go back to the previous menu display, press the [BS] key.

```
[EDIT-PGM]      No 0
001:_


1MOVA 2MOVI 3MOVF 4next
```

7) After selecting the robot language command, enter the operand data.

   When you press [X/Z+], the cursor moves to operand 1, so enter the data with the number keys. (Do not press ➡ at this point.)

   While pressing [X/Z+] or [X/Z−] to move the cursor, enter all necessary operand data as needed.

```
[EDIT-PGM]       No 0
001:MOVA 0   ,100
(point No)0→999
1P
```

8) After entering the operand data, press ➡.

```
[EDIT-PGM]       No 0
001:MOVA 1   ,80_
(speed)1→100
```

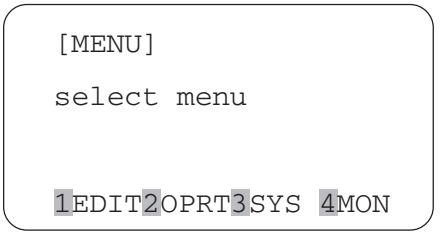
9) When entry is completed correctly, the cursor moves to the operation code part.

   To edit the next step, press [STEP UP] to scroll the step and repeat the procedure from step 6.

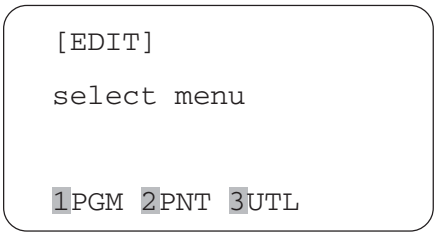
```
[EDIT-PGM]       No 0
001:MOVA 1   ,80


1MOVA 2MOVI 3MOVF 4next
```

6-2-2 Creating a new program

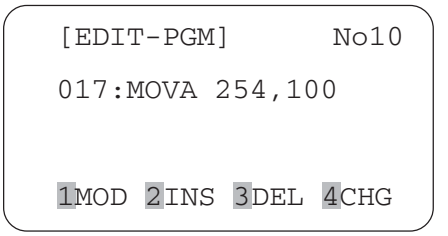
1) On the initial screen, press F1 (EDIT).



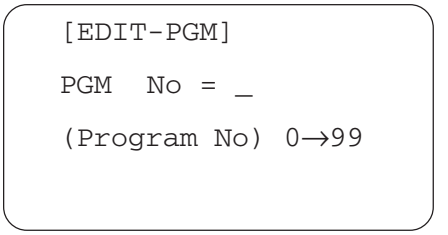
2) Next, press F1 (PGM).



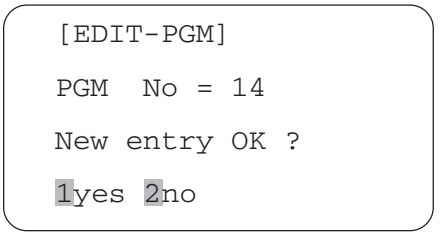
3) The execution program number and step are displayed on the screen. Press F4 (CHG) here.



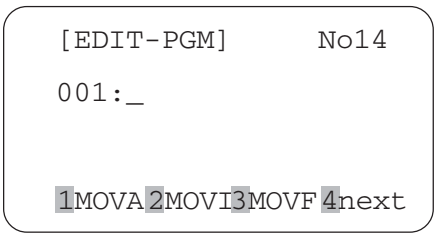
4) Enter the new program number with the number keys and press ➡.



5) A confirmation message appears. Make sure the program number is correct and press F1 (yes).

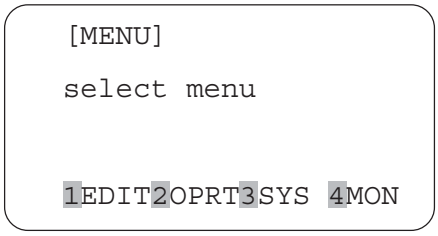


6) Proceed with program editing by following step 6 onward in "6-2-1 Creating programs after initialization."

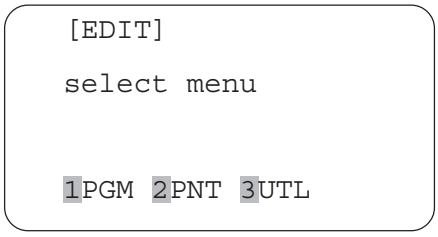


6-2-3 Adding a step

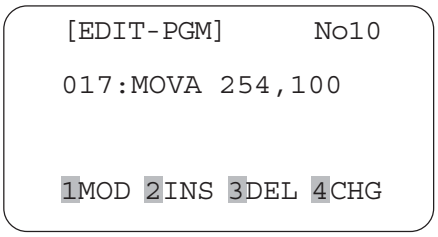
1) On the initial screen, press F1 (EDIT).



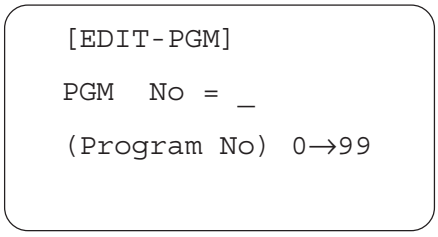
2) Next, press F1 (PGM).



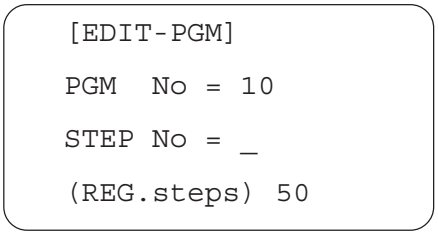
3) The execution program number and step are displayed on the screen. Press F4 (CHG) here.



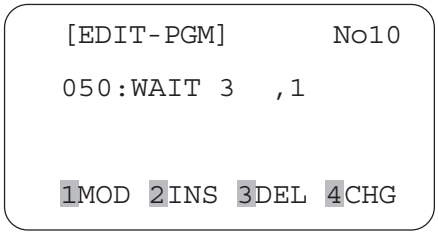
4) Enter the program number you want to edit with the number keys and press ➡.



5) Enter the last step number with the number keys and press ➡.

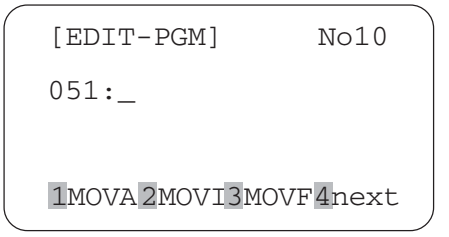


6) When the last step is displayed, press STEP UP.



7) Select F1 to F3 or a robot language command shown on the lower part of each number key.

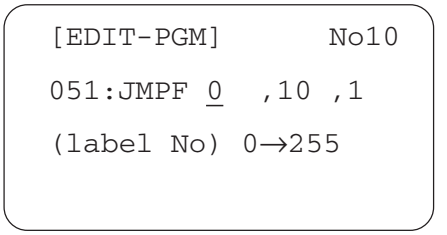
   To change the robot language menu display, press F4 (next). To go back to the previous menu display, press the BS key.



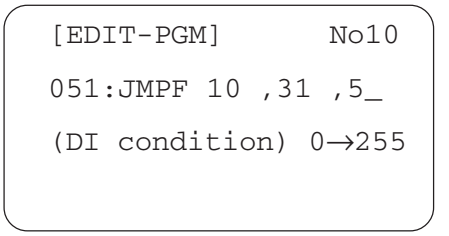
8) After selecting the robot language command, enter the operand data.

   When you press X/Z+, the cursor moves to operand 1, so enter the data with the number keys. (Do not press ➡ at this point.)

   While pressing X/Z+ or X/Z− to move the cursor, enter all necessary operand data as needed.

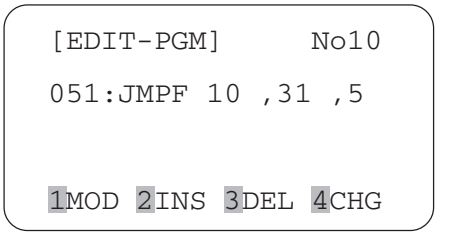


9) After entering the operand data, press ➡.



10) When the program has been edited correctly, the screen returns to step 6.

    When you want to add another step, press STEP UP to scroll to the next step and then repeat from step 7.



6-2-4 Correcting a step

1) Use the same procedure up to step 4 in "6-2-3 Adding a step".

2) Enter the number of the step you want to correct with the number keys and press ➡.

```
[EDIT-PGM]
PGM  No = 10
STEP No = _
(REG.steps) 50
```

3) Press F1 (MOD).

```
[EDIT-PGM]      No10
010:MOVA 999,100


1MOD 2INS 3DEL 4CHG
```

4) Select F1 to F3 or a robot language command shown on the lower part of each number key.

   To change the robot language menu display, press F4 (next). To go back to the previous menu display, press the BS key.

```
[EDIT-PGM]      No10
010:MOVA 999,100


1MOVA 2MOVI 3MOVF 4next
```

5) After selecting the robot language command, enter the operand data.

   When you press X/Z+, the cursor moves to operand 1, so enter the data with the number keys. (Do not press ➡ at this point.)

   While pressing X/Z+ or X/Z− to move the cursor, enter all necessary operand data as needed.

```
[EDIT-PGM]      No10
010:MOVA 10_,100
(point No) 0→999
1P
```

6) After entering the operand data, press ➡.

```
[EDIT-PGM]      No10
010:MOVA 10 ,100
(speed) 1→100
```

7) When entry is completed correctly, the cursor moves to the operation code part.

   If you want to change another step, press STEP UP to scroll the step and repeat the procedure from step 4.

```
[EDIT-PGM]     No10
010:MOVA 10 ,100


1MOVA 2MOVI 3MOVF 4next
```

6-2-5 Inserting a step

1) Use the same procedure up to step 4 in "6-2-3 Adding a step".

2) Enter the number of the step where you want to insert a step with the number keys and press ➡.

```
[EDIT-PGM]
PGM  No = 10
STEP No = _
(REG steps) 50
```

3) Press F2 (INS).

```
[EDIT-PGM]      No10
010:MOVA 999,100


1MOD 2INS 3DEL 4CHG
```

4) Select F1 to F3 or a robot language command shown on the lower part of each number key.

   To change the robot language menu display, press F4 (next). To go back to the previous menu display, press the BS key.

```
[EDIT-PGM]      No10
010:_


1MOVA 2MOVI 3MOVF 4next
```

5) After selecting the robot language command, enter the operand data.

   When you press X/Z+, the cursor moves to operand 1, so enter the data with the number keys. (Do not press ➡ at this point.)

   While pressing X/Z+ or X/Z- to move the cursor, enter all necessary operand data as needed.

```
[EDIT-PGM]      No10
010:MOVA 10_,100
(point No) 0→999
1P
```

6) After entering the operand data, press ➡.

```
[EDIT-PGM]      No10
010:MOVA 10 ,1̲00
(speed) 1→100
```

7) When entry is completed correctly, the screen returns to step 3.

```
[EDIT-PGM]      No10
010:MOVA 10 ,100


1MOD 2INS 3DEL 4CHG
```

6-2-6 Deleting a step

1) Use the same procedure up to step 4 in "6-2-3 Adding a step".

2) Enter the number of the step you want to delete with the number keys and press ➡.

```
[EDIT-PGM]
PGM  No = 10
STEP No = _
(REG steps) 50
```

3) Press F3 (DEL).

```
[EDIT-PGM]      No10
010:MOVA 999,100


1MOD 2INS 3DEL 4CHG
```

4) A confirmation message appears.
   To delete the step, press F1 (yes).
   To cancel the deletion, press F2 (no).

```
[EDIT-PGM]      No10
010:MOVA 999,100
delete OK ?
1yes 2no
```

5) When the step has been deleted, the screen returns to step 3.

```
[EDIT-PGM]      No10
010:WAIT 3   ,1


1MOD 2INS 3DEL 4CHG
```

6-3 Program Utility

6-3-1 Copying a program

1) On the initial screen, press F1 (EDIT).

```
[MENU]
select menu

1EDIT2OPRT3SYS 4MON
```

2) Next, press F3 (UTL).

```
[EDIT]
select menu

1PGM 2PNT 3UTL
```

3) Press F1 (COPY).

```
[EDIT-UTL]
select menu

1COPY2DEL 3LIST
```

4) Enter the program number you want to copy from with the number keys, and then press ➡.

```
[EDIT-UTL-COPY]
Copy from No = _
(Program No) 0→99
```

5) Enter the program number you want to copy to with the number keys, and then press ➡.

```
[EDIT-UTL-COPY]
Copy from No = 0
Copy to   No = 99_
(Program No) 0→99
```

6) If program data is already registered with the selected program number, a confirmation message appears.
To overwrite the program, press F1 (yes).
To cancel, press F2 (no).

```
[EDIT-UTL-COPY]
Copy from No = 0
No99 overwrite OK ?
1yes 2no
```

7) When the program has been copied, the screen returns to step 3.

```
[EDIT-UTL]
select menu

1COPY2DEL 3LIST
```

6-3-2 Deleting a program

1) Use the same procedure up to step 2 in "6-3-1 Copying a program".

2) Press F2 (DEL).

```
[EDIT-UTL]
select menu

1COPY2DEL 3LIST
```

3) Enter the number of the program you want to delete with the number keys and press ➡.

```
[EDIT-UTL-DEL]
delete PGM No = _
(Program No) 0→99
```

4) A confirmation message appears asking whether to delete the selected program.
To delete the program, press F1 (yes).
To cancel the deletion, press F2 (no).

```
[EDIT-UTL-DEL]
delete PGM No = 22
delete OK ?
1yes 2no
```

5) If the program has been deleted, the screen returns to step 2.

```
[EDIT-UTL]
select menu

1COPY2DEL 3LIST
```

6-3-3 Viewing the program information

1) Use the same procedure up to 2 in "6-3-1 Copying a program".

2) Press F3 (LIST).

```
[EDIT-UTL]
select menu


1COPY2DEL 3LIST
```

3) The program numbers are displayed on the screen, along with the number of registered steps and the number of available remaining steps.

   To view other program information, press the STEP UP and STEP DOWN keys to scroll the screen.

```
[EDIT-UTL-LIST]
  free   678 steps
  No 0    57 steps
  No 1   255 steps
```

4) Press the ESC key to return to the screen of step 2.

```
[EDIT-UTL]
select menu


1COPY2DEL 3LIST
```

* In addition to the number of existing steps, the steps equivalent to the number of programs are used internally as the program control steps. For example, if two programs are registered and their respective 50 and 100 steps are registered, then the number of available remaining steps will be as follows:
3000 – 2 – 50 – 100 = 2848 steps

Chapter 7 EDITING POINT DATA

There are three methods to enter point data: manual data input (MDI), teaching playback, and direct teaching. Manual data input allows you to directly enter point data with the TPB number keys.
Teaching playback moves the robot in manual operation to a desired position and then obtains that position as point data.
Direct teaching is basically the same as teaching playback, except that you move the robot by hand.

7-1 Manual Data Input

1) On the initial screen, press [F1] (EDIT).

```
[MENU]
select menu

1EDIT2OPRT3SYS 4MON
```

2) Next, press [F2] (PNT).

```
[EDIT]
select menu

1PGM 2PNT 3UTL
```

3) Press [F1] (MDI).

```
[EDIT-PNT]
select menu

1MDI 2TCH 3DTCH4DEL
```

4) The currently selected point data in the execution program appears on the screen.
If you want to edit another point data, press the [STEP UP] and [STEP DOWN] keys to scroll the point data.
To directly select the point data, press [F1] (CHG).

```
[EDIT-PNT-MDI]
P0    = 0.00      [mm]
input data[_        ]
1CHG
```

5) Enter the point number you want to edit with the number keys, and press ➡.

```
[EDIT-PNT-MDI]
 Pn : n = _
(point No) 0→999
```

6) Enter the point data with the number keys and press ➡.

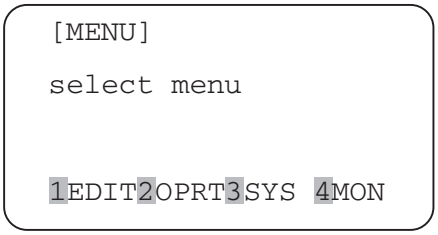
```
[EDIT-PNT-MDI]
P500 = -19.27    [mm]
input data[21. 76_ ]
1CHG
```

7) The input data is then registered as point data.

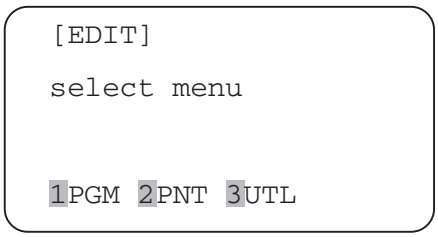
```
[EDIT-PNT-MDI]
P500 = 21.76     [mm]
input data[_        ]
1CHG
```

7-2 Teaching Playback

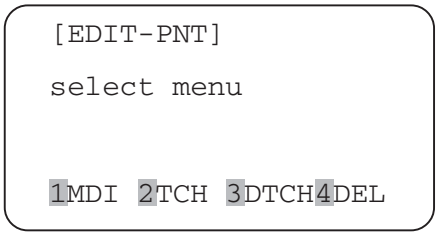
1) On the initial screen, press F1 (EDIT).



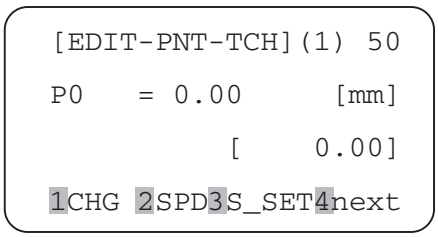
2) Next, press (PNT).



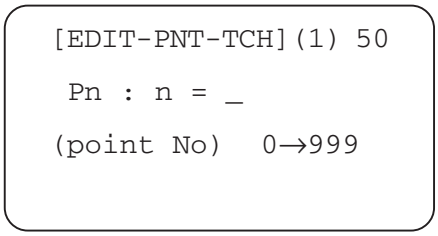
3) Press F2 (TCH).



4) The currently selected point data in the execution program appears on the screen.
If you want to edit another point data, press the STEP UP and STEP DOWN keys to scroll the point data.
To directly select the point data, press F1 (CHG).



5) Enter the point number you want to edit with the number keys, and press ➡.



6) Move the robot to the teaching position with the [X- Z-] or [X+ Z+] keys. Each time the [X- Z-] or [X+ Z+] key is pressed, the robot moves a certain amount in the direction indicated by the key and then stops.

Holding down the [X- Z-] or [X+ Z+] key moves the robot continuously at a constant speed until the key is released.

The amount of robot movement and the speed are proportional to the number (teaching movement data) displayed on the upper right of the screen.

In the example at the right, the teaching movement data is 50 (%), so the robot moves 0.5mm each time the [X- Z-] or [X+ Z+] key is pressed, as calculated below:

1mm (constant) × (50/100) = 0.5mm

If the [X- Z-] or [X+ Z+] key is kept pressed, the robot continuously moves at a speed of 50mm/s, as calculated below:

100mm/s (constant) × (50/100) = 50mm/s

```
[EDIT-PNT-TCH](1) 50
P500 = 19.27     [mm]
             [    0.00]
1CHG 2SPD3S_SET4next
```

7) Three different speed settings, SPEED (1), SPEED (2), and SPEED (3), are selectable as the teaching movement data. Each time **F2** (SPD) is pressed, the setting changes in the order of 1→2→3→1.

To change the teaching movement data setting, press **F3** (S_SET), enter the desired speed with the number keys, and press ➡. The screen then returns to step 6 when the data has been changed correctly.

```
[EDIT-PNT-TCH](1) 50
SPEED(1) = _
(speed) 1→100
```

8) Move the robot to the teaching position in this way and press ➡. The current position is input as point data.

```
[EDIT-PNT-TCH](1)100
P500 = 167.24    [mm]
             [  167.24]
1CHG 2SPD3S_SET4next
```

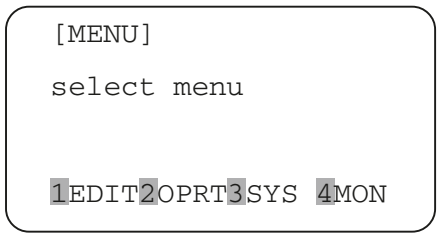
⚠ ***CAUTION***

When the SERVICE mode function is enabled, the following safety control will function. (See "10-4 SERVICE mode function".)

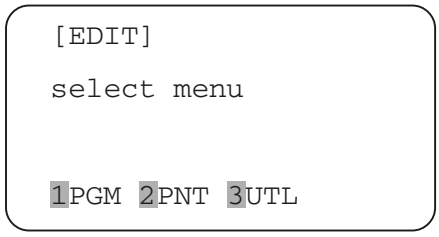
- ***Robot movement speed is limited to 10mm/s or less in "SERVICE mode state" when the robot movement speed limit is enabled.***

7-3 Direct Teaching

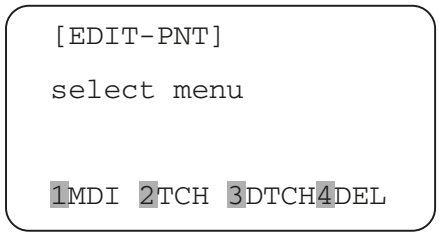
1) On the initial screen, press **F1** (EDIT).



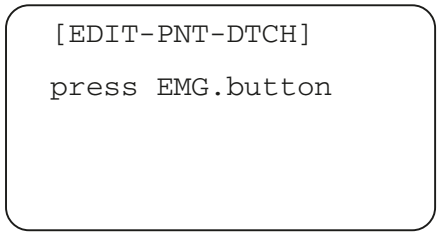
2) Next, press **F2** (PNT).



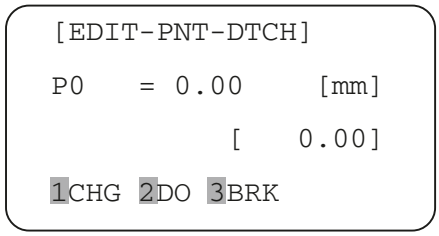
3) Press **F3** (DTCH).



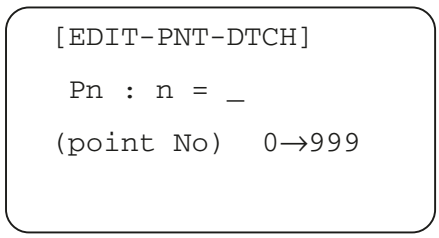
4) Following the message, press the emergency stop button on the TPB.



5) The currently selected point data in the execution program appears on the screen.
If you want to edit another point data, press the STEP UP and STEP DOWN keys to scroll the point data.
To directly select the point data, press **F1** (CHG).



6) Enter the point number you want to edit with the number keys, and press ➡.



7) Move the robot to the teaching position by hand.

```
[EDIT-PNT-DTCH]
P500 = 19.27     [mm]
            [     0.00]
1CHG 2DO 3BRK
```

8) Press ➡ to input the current position as point data.

   Use the same procedure to input all other necessary point data, and then press the (ESC) key.

```
[EDIT-PNT-DTCH]
P500 = 167.24    [mm]
            [  167.24]
1CHG 2DO  3BRK
```

9) Following the message, release the emergency stop button on the TPB.

```
[EDIT-PNT-DTCH]
release EMG.button
```

10) A confirmation message appears asking whether to turn the servo on.

   To turn the servo on, press [F1] (yes).

   To leave the servo off, press [F2] (no).

```
[EDIT-PNT-DTCH]
servo on ready ?

1yes 2no
```

11) The screen returns to step 3.

```
[EDIT-PNT]
select menu

1MDI 2TCH 3DTCH4DEL
```

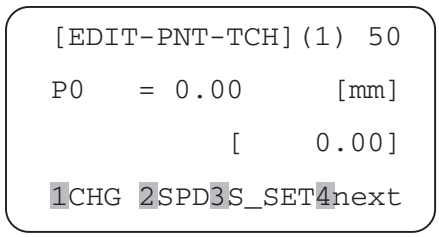
7-4 Manual Control of General-Purpose Output

When performing teaching playback or direct teaching with systems that use a general-purpose output through the I/O interface to operate a gripper or other tools, you may want to check the position of workpiece by actually moving it.
For this reason, the SRCP controller is designed to allow manual control of general-purpose outputs from the TPB.

1) Move the robot with the same procedure up to step 6 in "7-2 Teaching Playback" or up to step 7 in "7-3 Direct Teaching".

   The following steps are explained using the teaching playback screen.

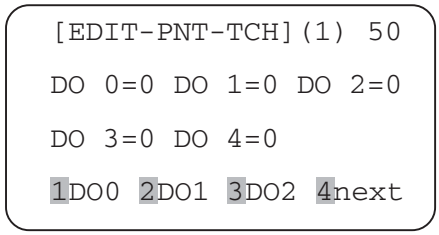
2) When the robot reaches the position where you want to operate general-purpose output, stop the robot. Then press **F4** (next) to change the menu display and then press **F1** (DO).



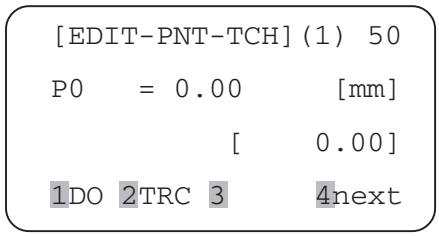
3) The current status of the general-purpose output appears on the screen.

   Press the function key that matches the DO number to switch the output on and off (on=1, off=0).

   If selecting DO3 to DO5, press **F4** (next) to change the menu display.



4) Press **ESC** to return to step 2.



7-5 Manual Release of Holding Brake

The holding brake on the vertical type robot can be released. Since the movable part will drop when the brake is released, attaching a stopper to protect the tool tip from being damaged is recommended.

1) Use the same procedure up to step 4 in "7-3 Direct Teaching".

2) Press F3 (BRK).

```
[EDIT-PNT-DTCH]
P0    = 0.00      [mm]
            [     0.00]
1CHG 2DO 3BRK
```

3) A confirmation message appears asking whether to release the brake.
   To release the brake, press F1 (yes).
   To cancel releasing the brake, press F2 (no).

```
[EDIT-PNT-DTCH]
take off the brake ?


1yes 2no
```

4) The screen returns to step 2.
   The brake stays released until F3 (BRK) is pressed again or the robot servo is turned on.

```
[EDIT-PNT-DTCH]
P0    = 0.00      [mm]
            [     0.00]
1CHG 2DO 3BRK
```

NOTE
Manual release of the holding brake is only possible on those robots equipped with a brake.

7-6 Deleting Point Data

1) Use the same procedure up to step 2 in "7-1 Manual Data Input".

2) Press F4 (DEL).

```
[EDIT-PNT]
select menu

1MDI 2TCH 3DTCH4DEL
```

3) Enter the point number at the start to delete point data with the number keys and press ➡.

```
[EDIT-PNT-DEL]
DEL range P_  -P
(point No) 0→999
```

4) Enter the point number at the end to delete point data with the number keys and press ➡.

```
[EDIT-PNT-DEL]
DEL range P100-P_
(point No) 0→999
```

5) A confirmation message appears asking whether to delete the data.
   To delete the data, press F1 (yes).
   To cancel the deletion, press F2 (no).

```
[EDIT-PNT-DEL]
DEL range P100-P110
delete OK ?
1yes 2no
```

6) When the point data has been deleted, the screen returns to step 2.

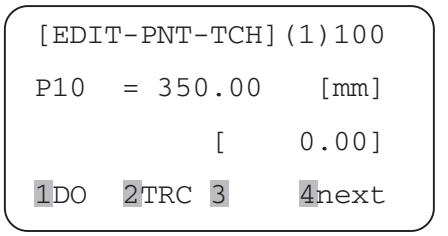
```
[EDIT-PNT]
select menu

1MDI 2TCH 3DTCH4DEL
```

7-7 Tracing Points (Moving to a registered data point)

The robot can be moved to the position specified by a registered data point. You can check the input point data by actually moving the robot.

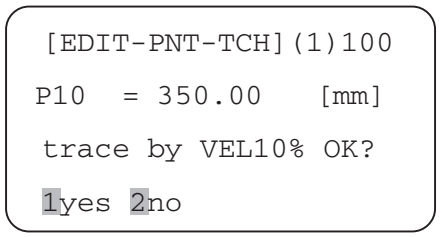
1) Use the same procedure up to step 5 in "7-2 Teaching Playback".

2) Press F4 (next) to change the menu display and then press F2 (TRC).

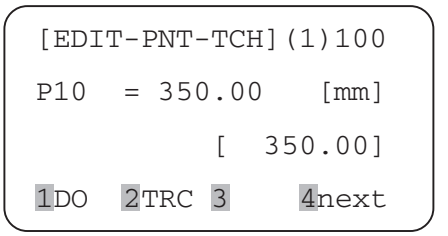


3) The coordinate data of the movement destination and the movement speed are displayed.
To move the robot, press F1 (yes).
To cancel moving the robot, press F2 (no).

The movement speed will be 10% of the number (speed parameter) displayed at the upper right of the screen.



4) When the movement is completed, the screen returns to step 2.



⚠ ***CAUTION***

When the SERVICE mode function is enabled, the following safety control will function. (See "10-4 SERVICE mode function".)

- ***Robot movement speed is limited to 3% or less of maximum speed in "SERVICE mode state" when the robot movement speed limit is enabled.***
- ***If the hold-to-run function is enabled, robot movement stops upon releasing F1 (yes) in "SERVICE mode state". (You must hold down F1 (yes) in step 3 until the robot reaches the target point.)***

Chapter 8 ROBOT LANGUAGE

This chapter explains the robot language. It describes what kind of commands are available and what they mean. The SRCP series uses the YAMAHA robot language. This is an easy-to-learn BASIC-like programming language. Even a first-time user can easily create programs to control complex robot and peripheral device movements.
At the beginning of this chapter, you will find a convenient table of robot language commands. At the end of this chapter, sample programs are listed for just your reference.

8-1 Robot Language Table

Instruction	Description and Format
MOVA	Moves to point data position. MOVA <point number>, <maximum. speed>
MOVI	Moves from current position by amount of point data. MOVI <point number>, <maximum. speed>
MOVF	Moves until specified DI input is received. MOVF <point number>, <DI number>, <DI status>
JMP	Jumps to a specified label in a specified program. JMP <label number>, <program number>
JMPF	Jumps to a specified label in a specified program according to the input condition. JMPF <label number>, <program number>, <input condition>
JMPB	Jumps to a specified label when a general-purpose input or memory input is in the specified state. JMPB <label number>, <DI or MI number>, <input status>
L	Defines the jump destination for a JMP or JMPF statement, etc. L <label number>
CALL	Runs another program. CALL <program number>, <number of times>
DO	Turns general-purpose output or memory output on and off. DO <DO or MO number>, <output status>
WAIT	Waits until a general-purpose input or memory input is in the specified state. WAIT <DI or MI number>, <input status>
TIMR	Waits the specified amount of time before advancing to the next step. TIMR <time>
P	Defines a point variable. P <point number>
P+	Adds 1 to a point variable. P+
P-	Subtracts 1 from a point variable. P-
SRVO	Turns a servo on and off. SRVO <servo status>
STOP	Temporarily stops program execution. STOP
ORGN	Performs return-to-origin. ORGN
TON	Runs a specified task. TON <task number>, <program number>, <start type>
TOFF	Stops a specified task. TOFF <task number>
JMPP	Jumps to a specified label when the axis positional relation meets the specified conditions. JMPP <label number>, <axis position condition>
MAT	Defines a matrix. MAT <number of rows>, <number of columns>, <pallet number>
MSEL	Specifies a matrix to move. MSEL <pallet number>
MOVM	Moves to a specified pallet work position on matrix. MOVM <pallet work position>, <maximum speed>
JMPC	Jumps to a specified label when counter array variable C equals to the specified value. JMPC <label number>, <counter value>
JMPD	Jumps to a specified label when counter variable D equals to the specified value. JMPD <label number>, <counter value>
CSEL	Specifies the array element of counter array variable C. CSEL <array element number>
C	Defines counter array variable C. C <counter value>
C+	Adds a specified value to counter array variable C. C+ [<addition value>]
C-	Subtracts a specified value from counter array variable C. C- [<subtraction value>]
D	Defines counter variable D. D <counter value>
D+	Adds a specified value to counter variable D. D+ [<addition value>]
D-	Subtracts a specified value from counter variable D. D- [<subtraction value>]
SHFT	Shifts the coordinate position by amount of specified point data. SHFT <point number>

Values in brackets [ ] can be omitted.

8-2 Robot Language Syntax Rules

8-2-1 Command statement format

The robot language command statement format for the SRCP controller is as follows. When creating a program using the TPB, each command statement can be automatically entered in this format, so you do not have to be aware of this format while creating the program.

```
<operation code>    [<operand 1>][,<operand 2>][,<operand 3>] [;<comment>]
```

- A command statement is basically composed of an operation code and an operand. Depending on the command statement, either no operand is used, or up to three operands are used.

  A comment can be written following the operand. (But, no comment can be written with the TPB.) A line consisting of only a comment cannot be created.

  Items in [ ] (brackets) can be omitted.

- A command statement must be entered with one-byte characters (alphanumeric characters, special characters) except for comment. Input characters can be upper case or lower case. The controller automatically converts the input characters to upper case.

- One command statement must be described within one line. It cannot be written over multiple lines. Multiple command statements cannot be described on one line. Up to 80 one-byte characters (including carriage line return) can be described on one line.

- One or more spaces must be inserted between the operation code and the operand.

- Operands enclosed in < > marks must be specified by the user. Check the description of each robot language and enter the appropriate data. (Refer to "8-4 Robot Language Description".)

- When two or more operands are entered, insert a comma (,) between them.

- Any entry after a semicolon (;) is recognized as a comment. When creating a program using a PC (personal computer), a comment is helpful to easily identify the program. Note, however, that the comment is not stored in the controller. A comment can be any number of characters as long as it is within one line. Characters that can be used as a comment are one-byte characters (alphanumeric characters, special characters) and two-byte characters (full space characters).

8-2-2 Variables

Variable are used in a program to hold data. The following variables can be used with the SRCP controller.

■ Point variable P

A point variable can contain a point number. It is used in movement commands such as MOVA and MOVI statements instead of specifying the point number directly. Sometimes the number of program steps can be reduced by using point variables.

■ Counter array variable C, Counter variable D

A counter variable can contain counter values and is used to specify the pallet work position number in a palletizing program and to count the number of runs. A counter array variable is an array of a total of 32 counter variables that can be selected by the CSEL statement of robot language.

■ Flag variable: memory input/output 100 to 147

A flag variable can only have a data value of 1 (ON) or 0 (OFF). It is used in a multi-task program to synchronize between tasks or in a condition judgement program.

Memory I/O from 100 to 131 can be freely turned on or off by the user or their values can be referenced. However, outputs 132 to 147 are controlled by the system so the user can only refer to their values.

Memory I/O description

Type	Memory I/O No.	Meaning
General-purpose	100 to 131	Memory I/O available to the user The user can freely set this with a DO statement.
Dedicated	132	Task 0 (main task) status Always set to 1.
	133	Task 1 status 1: Task has started. 0: Task has ended or has not yet started.
	134	Task 2 status 1: Task has started. 0: Task has ended or has not yet started.
	135	Task 3 status 1: Task has started. 0: Task has ended or has not yet started.
	136 to 139	Reserved for system use (Always set to 0.)
	140	X-axis hold status 1: Hold 0: Non hold
	141 to 143	Reserved for system use (Always set to 0.)
	144	X-axis constant movement status 1: Constantly moving 0: Accelerating, decelerating or in stop
	145 to 147	Reserved for system use (Always set to 0.)

8-3 Program Function

8-3-1 Multi-task function

A multi-task function allows simultaneous executing two or more programs (tasks). The SRCP controller can execute a maximum of 4 programs at the same time.
Since the multi-task function simultaneously executes two or more programs, the following processing can be performed.

- Other processing can be performed during robot movement.
  For example, a general-purpose output can be turned on or off while a robot movement command such as MOVA or MOVI statement is being executed. This reduces the cycle time.

A multi-task program can be written by the same method as normal programs. A TON statement used as the task start command is written into the main program and the subtask program is registered as another program number. When the TON command is processed during the program execution, the subtask starts to perform multiple tasks. The subtask will end when its last step has been executed or the TOFF command is issued.

Each task and data have the following relation.
Point variables P in a task are independent of those in other tasks.
Point data, general-purpose I/O and memory I/O are shared with each task.

⚠ ***CAUTION***
In addition to the tasks (up to 4 tasks) specified by the user, the system task starts inside the controller, so a maximum of 5 tasks are executed.
In general, the multi-task is defined as a function that simultaneously executes two or more programs (tasks). Strictly speaking, if the CPU is one unit, it executes two or more programs (tasks) while switching between them in an extremely short time almost as if they were being simultaneously executed. The SRCP controller uses this multi-task function to perform multiple tasks while switching the programs within a very short time (5ms maximum). Because of this, if 4 tasks are executed with this function, there is a maximum time of 20ms during which no processing is performed on one task. So the user must take this time into account when designing a system having multi-task functions.

8-4 Robot Language Description

8-4-1 MOVA

Function: Moves to a point specified by a point number (Moves to an absolute position relative to the origin point).

Format: MOVA <point number>, <maximum speed>

Example: MOVA 51, 80

Moves to P51 at speed 80.

Explanation: This command moves the robot to a position on the absolute coordinates whose origin position is defined as 0.

The robot starts moving when all axes enter the positioning-completed pulse range, and stops when all axes reach the OUT valid position.

(1) Point number

The point number is a number designated to each point from 0 to 999, a total of 1000 points, and used to create point date in PNT (point) mode. When a character "P" is entered here for special use, a point variable defined by the "P" statement is set as the point number. (See "8-4-12 P".)

(2) Maximum speed

The maximum speed can be set to any level between 1 and 100. If the execution speed in OPRT mode is 100, then 100 will be equal to 2000 mm/sec. (when PRM44=2000).

8-4-2 MOVI

Function: Moves a distance specified by a point number from the current position.

Format: MOVI <point number>, <maximum speed>

Example: MOVI 10, 80

Moves an amount equal to point data P10 from the current position at speed 80.

Explanation: This command moves the robot on the relative coordinates with the current position viewed as 0.

The robot starts moving when all axes enter the positioning-completed pulse range, and stops when all axes reach the OUT valid position.

(1) Point number

The point number is a number designated to each point from 0 to 999, a total of 1000 points, and used to create point date in PNT (point) mode. When a character "P" is entered here for special use, a point variable defined by the "P" statement is set as the point number. (See "8-4-12 P".)

(2) Maximum speed

The maximum speed can be set to any level between 1 and 100. If the execution speed in OPRT mode is 100, then 100 will be equal to 2000 mm/sec. (when PRM44=2000).

8-4-3 MOVF

Function: Moves until a specified DI number input is received.
Format: MOVF <point number> <DI number> <DI status>
Example: MOVF 1, 2, 1
The robot moves toward P1 and stops when DI2 turns on. Program execution then proceeds to the next step.
Explanation: This is used when searching for a target position using sensors or other devices.
The robot starts moving when all axes enter the positioning-completed pulse range, and stops when the DI conditions are met. Even if the DI conditions are not met, this command terminates when the robot reaches the specified point and proceeds to the next step.
(1) Point number
The point number is a number designated to each point from 0 to 999, a total of 1000 points, and used to create point date in PNT (point) mode. When a character "P" is entered here for special use, a point variable defined by the "P" statement is set as the point number. (See "8-4-12 P".)
(2) DI number
Specify one of the eight (0 to 7) general-purpose inputs.
(3) DI status
"1" means "on" and "0" means "off".
Other:
- The robot speed during execution of the MOVF movement can be specified by PRM9. (Refer to "PRM9: MOVF speed") Note that this will not be affected by the OPRT mode execution speed.

8-4-4 JMP

Function: Jumps to a specified step in a specified program.
Format: JMP <label number>, <program number>
Example: JMP 10, 8
Jumps to label 10 in program 8.
Explanation: This command is used to control the flow of program execution.
(1) Label number
The label number is a number defined by the "L" statement, and indicates the destination to jump. Any number from 0 to 255 can be specified. (See "8-4-7 L".)
(2) Program number
The program number is a number used to identify the 100 individual programs from 0 to 99.
Other:
- Even when the program number is changed by the JMP statement, resetting it will return to the original program number when program execution begins.

8-4-5 JMPF

Function: If the conditional jump input matches the setting value, program execution jumps to a specified label in a specified program.

Format: JMPF <label number>, <program number>, <input condition value>

Example: JMPF 12, 3, 5

If the conditional jump input is 5, program execution jumps to label 12 in program 3. If the jump input is not 5, program execution advances to the next step.

Explanation: This command is used to control the flow of program execution according to the conditional jump input.

(1) Label number

The label number is a number defined by the "L" statement, and indicates the destination to jump. Any number from 0 to 255 can be specified. (See "8-4-7 L".)

(2) Program number

The program number is a number used to identify the 100 individual programs from 0 to 99.

(3) Input condition value

This is the condition used to make a jump. A general-purpose input is viewed as binary code input, and if it meets this input condition value, a jump is executed.

The number of points that can be branched under the input condition depends on the number of conditional input points which is set with PRM8. (See "PRM8: No. of conditional input points".)

⚠ ***CAUTION***

Select a number of conditional input points that is large enough to accommodate the actual number of input conditions to be used. If an error is made in setting the number of conditional input points, there will be a discrepancy between the input condition value required by the program and that recognized by the controller. This could keep the program from operating properly.

General-purpose input status and input condition value when the number of conditional input points is 4 (input range 0 to 15)

General use DI Input				Input Condition Value
DI3 (2^3)	DI2 (2^2)	DI1 (2^1)	DI0 (2^0)	
OFF	OFF	OFF	OFF	0
OFF	OFF	OFF	ON	1
OFF	OFF	ON	OFF	2
OFF	OFF	ON	ON	3
OFF	ON	OFF	OFF	4
OFF	ON	OFF	ON	5
OFF	ON	ON	OFF	6
OFF	ON	ON	ON	7
ON	OFF	OFF	OFF	8
ON	OFF	OFF	ON	9
ON	OFF	ON	OFF	10
ON	OFF	ON	ON	11
ON	ON	OFF	OFF	12
ON	ON	OFF	ON	13
ON	ON	ON	OFF	14
ON	ON	ON	ON	15

8-4-6 JMPB

Function: Jumps to a specified label when a specified general-purpose input or memory input is ON or OFF.

Format: JMPB <label number>, <DI or MI number>, <input status>

Example: JMPB 12, 2, 1

Jumps to label 12 when DI2 input is ON.

If DI2 is OFF, the program execution proceeds to the next step.

Explanation: This command controls the program flow according to the general-purpose input or memory input status.

(1) Label number

The label number is a number defined by the "L" statement, and indicates the destination to jump. Any number from 0 to 255 can be specified. (See "8-4-7 L".)

(2) DI or MI number

Specify one of the general-purpose input numbers from 0 to 7 (8 points) or memory input numbers from 100 to 147 (48 points).

(3) Input status

"1" means "on" and "0" means "off".

8-4-7 L

Function: Defines the jump destination for JMP, JMPF or JMPB statements, etc.

Format: L <label number>

Example: L 100

Defines label 100.

Explanation: This command is used to define the destination to which program execution jumps with a jump command. The label number may be any number between 0 and 255. The same label numbers may be used if they are in different programs.

8-4-8 CALL

Function: Calls and executes another program.
Format: CALL <program number>, <number of times>
Example: CALL 5, 2
Calls program 5 and executes it twice. Program execution then proceeds to the next step.
Explanation: When repeating the same operation a number of times, the CALL statement is used as needed to call and execute the subroutine defined as a separate program.
(1) Program number
The program number is a number used to identify the 100 individual programs from 0 to 99.
(2) Number of times
This is the number of times that the program is to be repeated. This can be specified from 1 to 255.
Other:
- The nesting level is 6.
- When the end of the program initiated by the CALL statement is detected, program execution advances to the step following the CALL statement in the main program.
- An error occurs and program execution stops if the program being executed is called by the CALL statement.
- Even when the program number is changed by the CALL statement, resetting it will return to the original program number when program execution begins.
- An error "stack overflow" might occur if a jump is made to another program by the JMP or JMPF statement in a program called as a subroutine by the CALL statement.

8-4-9 DO

Function: Controls ON/OFF of general-purpose output or memory output.
Format: DO <DO or MO number>, <output status>
Example: DO 3, 1
Turns on DO3.
Explanation: This command turns the general-purpose output or memory output on and off.
(1) DO or MO number
Specify one of the general-purpose output numbers from 0 to 4 (5 points) or memory output numbers from 100 to 131 (32 points).
(2) Output status
"1" means "on" and "0" means "off".

8-4-10 WAIT

Function: Waits until a specified general-purpose input or memory input changes to a specified state.

Format: WAIT <DI or MI number>, <input status>

Example: WAIT 5, 1

Waits until DI5 turns on.

Explanation: This command adjusts the timing according to the general-purpose input or memory input state.

(1) DI or MI number

Specify one of the general-purpose input numbers from 0 to 7 (8 points) or memory input numbers from 100 to 147 (48 points).

(2) Input status

"1" means "on" and "0" means "off",

8-4-11 TIMR

Function: Waits for a specified amount of time before advancing to the next step.

Format: TIMR <time>

Example: TIMR 100

Moves to the next step after waiting one second.

Explanation: This command is used when adjusting the time within the program. Time may be specified in lengths from 1 to 65535, in units of 10ms. In other words, time may be specified from 0.01 seconds up to 655.35 seconds.

8-4-12 P

Function: Sets a point variable P.
Format: P <point number>
Example: P 200
Sets a point variable P to 200.
Explanation: The point variable can contain a point number as a variable, which can be from 0 to 999. By using a movement command such as MOVA with a P+ or P- statement, the number of steps required to create a repeating program can be reduced.
Other:
- The contents of point variable P are retained even when the controller power is turned off, but when the program is reset or when the program reset is applied for example by switching the execution program, the point variable P will be initialized to 0.
- Point variables P in a task are independent of those in other tasks. For example, the definition and edited contents of a point variable used in task 1 do not affect the point variable used in task 0.

8-4-13 P+

Function: Adds 1 to a point variable P.
Format: P+
Example: P+
Adds 1 to a point variable P. (P←P+1)
Explanation: Adds 1 to a point variable P.

8-4-14 P-

Function: Subtracts 1 from a point variable P.
Format: P-
Example: P-
Subtracts 1 from a point variable P. (P←P-1)
Explanation: Subtracts 1 from a point variable P.

8-4-15 SRVO

Function: Turns the servo on and off.
Format: SRVO <servo status>
Example: SRVO 1
This turns the servo on.
SRVO 0
This turns the servo off.
Explanation: This command is used to prevent an overload on the motor that may occur if the robot is locked mechanically after positioning is completed. This command is executed after the specified axis enters the positioning-completed pulse range.
(1) Servo status
"1" means "on" and "0" means "off."

8-4-16 STOP

Function: Temporarily interrupts program execution.
Format: STOP
Example: STOP
Temporarily interrupts program execution.
Explanation: This command temporarily interrupts execution of a program. If two or more tasks are being executed, then all those tasks are interrupted. This command can be used at any point in a program. The next execution will restart from the subsequent step.
Others:
- Normally, the program terminates when the last step is detected. At the same time, the program is reset and the execution step number will return to 1 (top line of the program).
- To interrupt only a subtask temporarily without stopping the main task, use the TOFF statement. (Refer to "8-4-19 TOFF".)

8-4-17 ORGN

Function: Performs return-to-origin by using the stroke-end detection method.
Format: ORGN
Example: ORGN
Performs return-to-origin by the stroke-end detection method.
Explanation: Return-to-origin is performed based on return-to-origin parameter data.
Others:
- The magnetic pole is detected simultaneously with return-to-origin operation. Each time the power is turned on, return-to-origin becomes incomplete. Always perform return-to-origin after turning on the power to the controller before starting operation. Return-to-origin is also incomplete after a parameter related to the origin position is changed. Return-to-origin must be reperformed in this case.
- When performing return-to-origin by the stroke-end detection method, do not interrupt the return-to-origin operation while detecting the origin (while contacting the mechanical limit). Otherwise, the operation will stop due to a controller overload alarm and the power will have to be turned on again.
- If return-to-origin must be repeated by the stroke-end detection method, wait at least 5 seconds before repeating it.

8-4-18 TON

Function: Executes a specified task.
Format: TON <task number>, <program number>, <start type>
Example: TON 1,2,0
Newly executes program 2 as task 1.
Explanation: This command starts multiple tasks and can be used to control the I/O signals in parallel with the axis movement and perform different processing for each axis.
(1) Task number
The task number is a number used to identify the four individual tasks from 0 to 3. Since task 0 is the main task, tasks numbers from 1 to 3 can be specified.
(2) Program number
The program number is a number used to identify the 100 individual programs from 0 to 99.
(3) Start type
This specifies whether to start a new task or suspended task. Set to 0 when executing a new task, and set to 1 when restarting a suspended task.
Others:
- A task number which is being executed cannot be specified. (A task number which has been suspended can be specified.)
- The task terminates when the last step is detected. When a subtask terminates, it does not affect operation of other tasks. But, if task 0 (main task) terminates, all other tasks in operation also terminate.

8-4-19 TOFF

Function: Suspends a specified task.
Format: TOFF <task number>
Example: TOFF 1
Suspends the program being executed as task 1.
Explanation: This command is used to suspend the execution of a particular task.
(1) Task number
The task number is a number used to identify the four individual tasks from 0 to 3. Since task 0 is the main task, tasks numbers from 1 to 3 can be specified.
Others:
- This command cannot suspend its own task.

8-4-20 JMPP

Function: Jumps to a specified label when the axis position relation meets the specified conditions.

Format: JMPP <label number>, <axis position condition>

Example: JMPP 3,1

Jumps to label 3 if the X-axis position is smaller than the point specified with the point variable P.

Explanation: This command controls the program flow according to the specified position of the axis, by comparing it with the point specified with the point variable P.

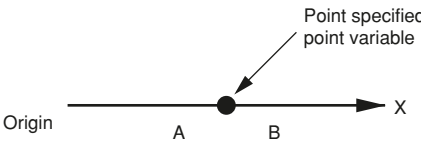
(1) Label number

The label number is a number defined by the "L" statement, and indicates the destination to jump. Any number from 0 to 255 can be specified. (See "8-4-7 L".)

(2) Axis position condition

When set to 1, this establishes the condition that the robot must be closer to the origin than the specified position. When set to 2, this establishes the condition that the robot must be farther away from the origin than the specified position.

Others:
- When the axis is at the specified coordinate position, this views that the condition is met.



Axis position condition	Robot position that meets the condition
1	Robot is in area A.
2	Robot is in area B.

8-4-21 MAT

Function: Defines the number of rows and columns of the matrix.

Format: MAT <number of rows>, <number of columns>, <pallet number>

Example: MAT 3, 6, 0

Defines a matrix of 3 × 6 on pallet number 0.

Explanation: This command defines a matrix for palletizing movement. A palletizing program can be easily created by using this command with MSEL or MOVM, etc.

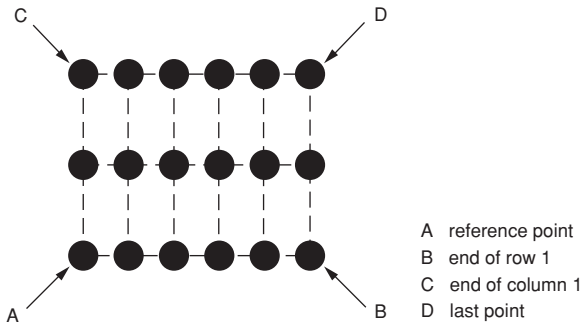
(1) Number of rows
Set any value from 1 to 255.

(2) Number of columns
Set any value from 1 to 255.

(3) Pallet number
This number is used for matrix identification and can be set from 0 to 31. A total of 32 matrix data can be handled.

Others:

- A common method for matrix coordinate definition specifies only the positions of the 4 corners of the matrix by 4-point teaching. The remaining points are then found by calculation. When teaching the positions of the 4 corners in PNT (point) mode to create point data, the point numbers are generally specified as follows: If pallet number is "n" for instance, enter the position of the reference point (row 1, column 1) in p(251-4n), the position at the end of row 1 in p(252-4n), the position at the end of column 1 in p(253-4n), and the position of the remaining corner in p(254-4n). To define a one-dimension matrix such as "row 1, column m", enter the position of the reference point (row 1, column 1) in p251, and the position of last point (row 1, column m) in p252. You do not have to enter any data in p253 and p254 (when pallet number is 0).
- The matrix definition contents are shared with each task.
- Because only a single-axis robot is controlled with the SRCP series, the actual movement is linear even if a 2-dimensional matrix is defined.



Matching point numbers for inputting pallet numbers and coordinate values A to D

Pallet No.	0	1	n	31
A	p251	p247	p(251-4n)	p127
B	p252	p248	p(252-4n)	p128
C	p253	p249	p(253-4n)	p129
D	p254	p250	p(254-4n)	p130

8-4-22 MSEL

Function: Specifies a matrix where the robot moves with a MOVM statement.

Format: MSEL <pallet number>

Example: MSEL 0

Points where the robot moves with a MOVM statement are calculated based on matrix data of pallet number 0.

Explanation: This command selects a matrix and is always used with a MOVM statement as a pair.

(1) Pallet number

This number is used for matrix identification and can be set from 0 to 31.

Others:

- The pallet number assigned with the MSEL statement is independent of each task. For example, when different pallet numbers are assigned to task 0 and task 1, then task 0 and task 1 execute the MOVM statement based on different pallet data.

8-4-23 MOVM

Function: Moves to a point on the specified matrix.

Format: MOVM <pallet work position>, <maximum speed>

Example: MOVM 23, 100

Moves to the point at row 3, column 7 at speed 100 when a matrix of 5 × 8 is defined by the MAT statement.

Explanation: This command moves the robot to each point on a matrix specified by the MSEL statement.

This command allows the robot to start moving when all axes are within the "positioning-completed pulse" range. This command ends when all axes enter the OUT valid position.

(1) Pallet work position

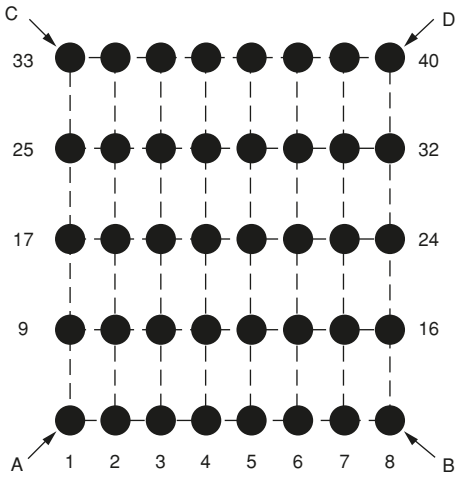
The pallet work position is a number used to identify each point on a matrix, and can be from 1 to 65025 (=255 × 255). For example, on a "row M, column N" matrix, the pallet work position at "row A, column B" is found by (A–1) × N+B. When a character "C" or "D" is entered here for special use, a counter variable is set in each pallet work position.

(2) Maximum speed

The maximum speed can be set to any level between 1 and 100. If the execution speed in OPRT mode is 100, then 100 will be equal to 2000 mm/sec. (when PRM44=2000).

Others:

- The MOVM statement performs calculation on the assumption that the robot operates on the Cartesian coordinate system.
- Because only a single-axis robot is controlled with the SRCP series, the actual movement is linear even if a 2-dimensional matrix is defined.



Example of pallet work position in 5 × 8 matrix

Work Position	Position No.
A: Reference point	1
B: End of row 1	8
C: End of column 1	33
D: Last point	40

8-4-24 JMPC

Function: Jumps to a specified label when the counter array variable C matches a specified value.

Format: JMPC <label number>, <counter value>

Example: JMPC 5, 100

Jumps to label 5 when the counter array variable C is 100. Program execution proceeds to the next step except when the counter array variable C is 100.

Explanation: This command controls the program flow according to the counter array variable C. The counter array variable C to be compared is the element number specified with the CSEL statement.

(1) Label number

The label number is a number defined by the "L" statement, and indicates the destination to jump. Any number from 0 to 255 can be specified. (See "8-4-7 L")

(2) Counter value

Set any value from 0 to 65535.

8-4-25 JMPD

Function: Jumps to a specified label when the counter variable D matches a specified value.

Format: JMPD <label number>, <counter value>

Example: JMPD 5, 100

Jumps to label 5 when the counter variable D is 100. Program execution proceeds to the next step except when the counter variable D is 100.

Explanation: This command controls the program flow according to the counter variable D.

(1) Label number

The label number is a number defined by the "L" statement, and indicates the destination to jump. Any number from 0 to 255 can be specified. (See "8-4-7 L")

(2) Counter value

Set any value from 0 to 65535

8-4-26 CSEL

Function: Specifies an array element of the counter array variable C to be used.

Format: CSEL <array element number>

Example: CSEL 1

The counter array variable of element number 1 is used in the subsequent steps.

Explanation: This command designates an array element number of the counter array variable C.

The array element data designated with the CSEL statement is used in the C statement, C+ statement, C- statement, JMPC statement and MOVM statement.

(1) Array element number

This is a number used to designate the array element number of a counter array variable and can be any value from 0 to 31.

When a character "D" is entered here, the counter variable D is used to designate the element of the counter array variable.

Others:

- The array element designation is held even when the controller power is turned off, but when the program is reset or when the program reset is applied by switching the execution program, the element designation number will be initialized to 0.
- The element number designated with the CSEL statement is independent of each task. For example, when different array elements are designated for task 0 and task 1, the definition or change in the counter array variable C of task 1 does not affect task 0.

8-4-27 C

Function: Sets the counter array variable C.

Format: C <counter value>

Example: C 200

Sets the counter array variable C to 200.

Explanation: This command sets a counter value for the counter array variable specified with the CSEL statement. The counter array variable is an array variable containing 32 elements, and can be set to any value from 0 to 65535. This command can be used with a C+ or C– statement and a JMPC statement for a repeating program and also with a MOVM statement for a palletizing program.

Others:

- Counter array variable C is not initialized even if the program is reset or the controller power is turned off. To initialize, rewrite the program.
- The counter array variable C is a variable shared with all tasks. For example, task 0 and task 1 use a counter array variable with the same element number, the edited contents of task 1 affect task 0.

8-4-28 C+

Function: Adds a specified value to the counter array variable C.
Format: C+ [<addition value>]
Example: C+ 100
Adds 100 to the counter array variable C. (C←C+100)
C+
Adds 1 to the counter array variable C. (C←C+1)
Explanation: This command adds a specified value to the counter array variable C specified with the CSEL statement. The addition value can be set to any value from 1 to 65535. If the addition value is omitted, then 1 is added to the counter array variable C.

8-4-29 C-

Function: Subtracts a specified value from the counter array variable C.
Format: C- [<subtraction value>]
Example: C- 100
Subtracts 100 from the counter array variable C. (C←C-100)
C-
Subtracts 1 from the counter array variable C. (C←C-1)
Explanation: This command subtracts a specified value from the counter array variable C specified with the CSEL statement. The subtraction value can be set to any value from 1 to 65535. If the subtraction value is omitted, then 1 is subtracted from the counter array variable C.

8-4-30 D

Function: Sets the counter variable D.
Format: D <counter value>
Example: D 200
Sets the counter variable D to 200.
Explanation: The counter variable D can be set to any value by the user from 0 to 65535. This command can be used with a D+ or D– statement and a JMPD statement for a repeating program, and also with a MOVM statement for a palletizing program.
Others:
- The counter variable D is not initialized even if the program is reset or the controller power is turned off. To initialize, rewrite the program.
- The counter variable D is a variable shared with all tasks.
  For example, task 0 and task 1 use the counter variable D, the edited contents of task 1 affect task 0.

8-4-31 D+

Function: Adds a specified value to the counter variable D.
Format: D+ [<addition value>]
Example: D+ 100
Adds 100 to the counter variable D. (D←D+100)
D+
Adds 1 to the counter variable D. (D←D+1)
Explanation: This command adds a specified value to the counter variable D. The addition value can be set to any value from 1 to 65535. If the addition value is omitted, then 1 is added to the counter variable D.

8-4-32 D-

Function: Subtracts a specified value from the counter variable D.
Format: D- [<subtraction value>]
Example: D- 100
Subtracts 100 from the counter variable D. (D←D-100)
D-
Subtracts 1 from the counter variable D. (D←D-1)
Explanation: This command subtracts a specified value from the counter variable D. The subtraction value can be set to any value from 1 to 65535. If the subtraction value is omitted, then 1 is subtracted from the counter variable D.

8-4-33 SHFT

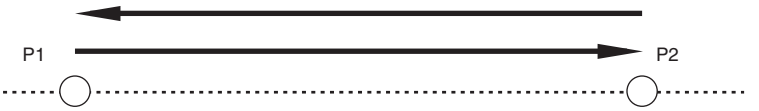
Function: Shifts the position data.
Format: SHFT <point number>
Example: SHFT 10
Shifts the coordinates on which the subsequent movement commands are executed, by a data amount defined by point 10.
Explanation: This command shifts position data in the subsequent movement commands to be executed, by coordinates equal to the specified point data. The shift data is valid until the SHFT statement is executed again or until the program reset is executed.
(1) Point number
The point number is a number designated to each point from 0 to 999, a total of 1000 points, and used to create point date in PNT (point) mode. When a character "P" is entered here for special use, a point variable defined by the "P" statement is set as the point number. (Refer to "8-4-12 P".)
Others:
- When the program is reset or when the program reset is applied by switching the execution program, the shift data will be initialized to (0.00).
- The SHFT statement affects MOVA, MOVF and MOVM, but does not affect MOVI.
- The coordinate shift amount specified with the SHFT statement is independent of each task. For example, when task 0 and task 1 are being executed, the coordinate shift of task 1 has no effect on the movement command for task 0.

```
004 :    :
005 : SHFT  1
006 : MOVA  0,100
007 :    :
```

For example, with a program shown on the left, the target position of MOVA statement in the 6th step will be the position of P0 + P1.

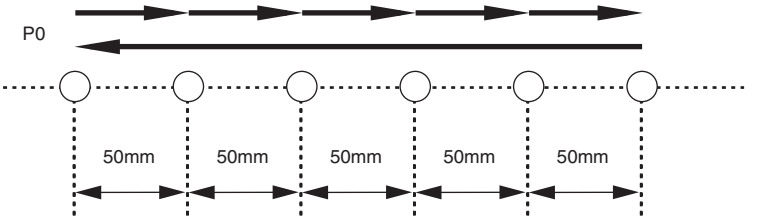
8-5 Sample Programs

8-5-1 Moving between two points



Program [NO0]			Comment
001: L	0		; Label definition
002: MOVA	1,	100	; Moves to P1
003: MOVA	2,	100	; Moves to P2
004: TIMR	100		; Delays for one second
005: JMP	0,	0	: Returns to L0

8-5-2 Moving at an equal pitch

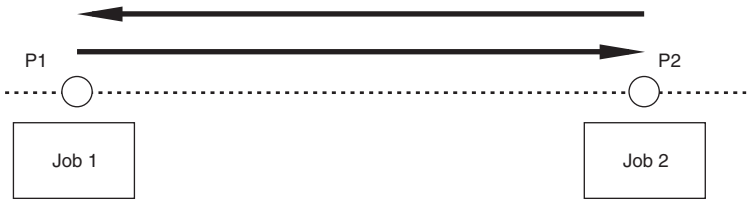


Point

P0	Start position
P1	Movement distance: 50mm

Program [NO0]			Comment
001: L	0		; Label definition
002: MOVA	0,	100	; Moves to P0
003: MOVI	1,	100	; Moves five times at a 50mm pitch
004: MOVI	1,	100	; Moves five times at a 50mm pitch
005: MOVI	1,	100	; Moves five times at a 50mm pitch
006: MOVI	1,	100	; Moves five times at a 50mm pitch
007: MOVI	1,	100	; Moves five times at a 50mm pitch
008: JMP	0,	0	; Returns to L0

8-5-3 Positioning 2 points and sending job commands to a PLC at each position



Point

P1	Position at which job 1 is complete
P2	Position at which job 2 is complete

General-purpose input

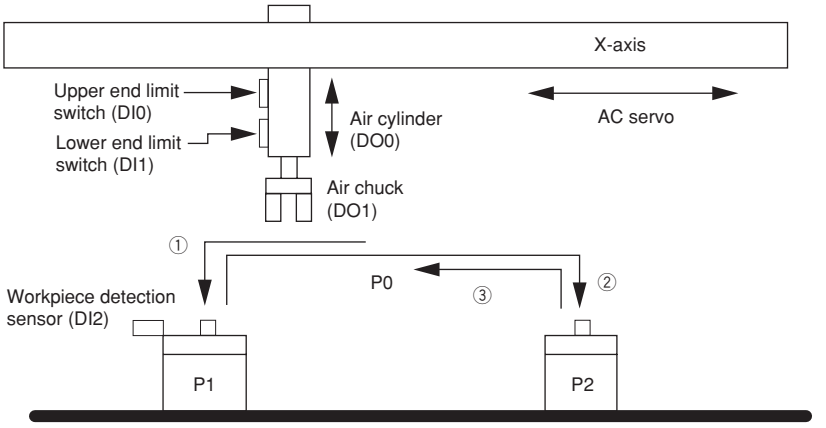
DI1	Job 1 completion	1: Complete	0: Not complete
DI2	Job 2 completion	1: Complete	0: Not complete

General-purpose output

DO1	Job 1 command	1: Output	0: Canceled
DO2	Job 2 command	1: Output	0: Canceled

```
Program                          Comment
[NO0]
001:DO      1,   0      ; Cancels job 1 command
002:DO      2,   0      ; Cancels job 2 command
003:L       1           ; Label definition
004:MOVA    1,   100    ; Moves to P1
005:DO      1,   1      ; Outputs job 1 command
006:WAIT    1,   1      ; Waits until job 1 is complete
007:DO      1,   0      ; Cancels job 1 command
008:MOVA    2,   100    ; Moves to P2
009:DO      2,   1      ; Outputs job 2 command
010:WAIT    2,   1      ; Waits until job 2 is complete
011:DO      2,   0      ; Cancels job 2 command
012:JMP     1,   0      ; Returns to L1
```

8-5-4 Robot stands by at P0, and moves to P1 and then to P2 to pick and place a workpiece



Operation

① Moves to the workpiece feed position from the standby position, and picks up a workpiece.
② Moves to the workpiece mount position and places the workpiece.
③ Returns to the standby position.

Actuator

Horizontal direction	AC servo motor
Vertical direction	Air cylinder
Hold	Air chuck

General-purpose input

DI0	Upper end limit switch	1: ON	0: OFF
DI1	Lower end limit switch	1: ON	0: OFF
DI2	Workpiece detection sensor	1: Detected	0: No

General-purpose output

DO0	Air cylinder	1: Down	0: Up
DO1	Air chuck	1: Close	0: Open

Point

P0	Robot standby position
P1	Workpiece feed position
P2	Workpiece mount position

Program **Comment**

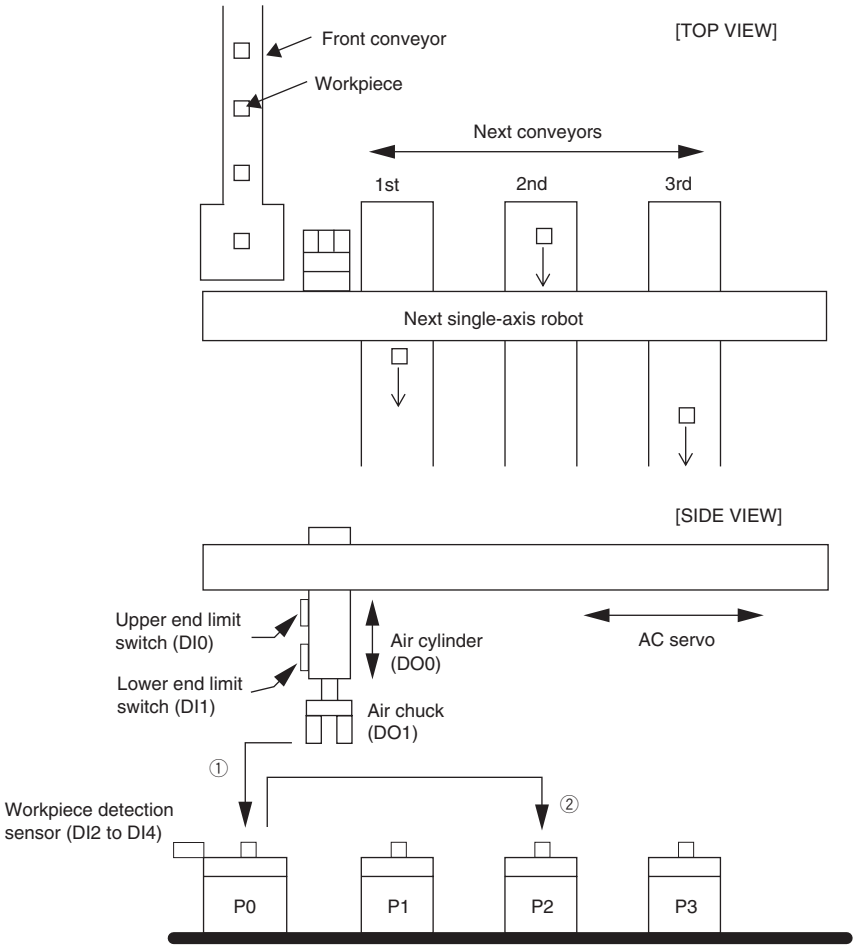
[NO1]

```
001:L     1          ; Label definition
002:MOVA  0,   100   ; Moves to the standby position
003:WAIT  2,   1     ; Waits for workpiece feed
004:MOVA  1,   100   ; Moves to the workpiece feed position
005:DO    0,   1     ; Air cylinder moves down
006:WAIT  1,   1     ; Waits until the air cylinder moves down
007:DO    1,   1     ; Chuck closes
008:TIMR  100        ; Delays for one second
009:DO    0,   0     ; Air cylinder moves up
010:WAIT  0,   1     ; Waits until the air cylinder moves up
011:MOVA  2,   100   ; Moves to the workpiece mount position
012:DO    0,   1     ; Air cylinder moves down
013:WAIT  1,   1     ; Waits until the air cylinder moves down
014:DO    1,   0     ; Chuck opens
015:TIMR  100        ; Delays for one second
016:DO    0,   0     ; Air cylinder moves up
017:WAIT  0,   1     ; Waits until the air cylinder moves up
018:JMP   1,   1     ; Returns to L1
```

Picks up workpiece

Placing workpiece

8-5-5 Picking up 3 kinds of workpieces flowing on the front conveyor and placing them on the next conveyors while sorting



Operation

① Moves to the workpiece feed position and picks up a workpiece.

② Moves to the workpiece mount position and places the workpiece.

Actuator

Horizontal direction	AC servo motor
Vertical direction	Air cylinder
Hold	Air chuck

General-purpose input

DI0	Upper end limit switch	1: ON	0: OFF
DI1	Lower end limit switch	1: ON	0: OFF
DI2	Workpiece A detection sensor	1: Detected	0: No
DI3	Workpiece B detection sensor	1: Detected	0: No
DI4	Workpiece C detection sensor	1: Detected	0: No

Point

P0	Workpiece feed position on the front conveyor
P1	Workpiece A mount position on next conveyor
P2	Workpiece B mount position on next conveyor
P3	Workpiece C mount position on next conveyor

General-purpose output

DO0	Air cylinder	1: Down	0: Up
DO1	Air chuck	1: Close	0: Open

```
Program                                  Comment
[NO1]                                    <<Main routine>>
001:L      1                             ; Label definition
002:JMPB   2,   2,     1                 ; Jumps to L2 when workpiece A is detected
003:JMPB   3,   3,     1                 ; Jumps to L3 when workpiece B is detected
004:JMPB   4,   4,     1                 ; Jumps to L4 when workpiece C is detected
005:JMP    1,   1                        ; Returns to L1
006:L      2                             ; Label definition
007:P      1                             ; Sets the point variable to 1
008:JMP    5,   1                        ; Jumps to L5
009:L      3                             ; Label definition
010:P      2                             ; Sets the point variable to 2
011:JMP    5,   1                        ; Jumps to L5
012:L      4                             ; Label definition
013:P      3                             ; Sets the point variable to 3
014:L      5                             ; Label definition
015:CALL   2,   1                        ; Executes a [PICK] subroutine
016:CALL   3,   1                        ; Executes a [PLACE] subroutine
017:JMP    1,   1                        ; Returns to L1

[NO2]                                    <<Picking up a workpiece>>
001:MOVA   0,   100                      ; Moves to the workpiece feed position
002:DO     0,   1                        ; Air cylinder moves down
003:WAIT   1,   1                        ; Waits until the air cylinder moves down
004:DO     1,   1                        ; Chuck closes
005:TIMR   100                           ; Delays for one second
006:DO     0,   0                        ; Air cylinder moves up
007:WAIT   0,   1                        ; Waits until the air cylinder moves up

[NO3]                                    <<Placing a workpiece>>
001:MOVA   P,   100                      ; Moves to the workpiece mount position
002:DO     0,   1                        ; Air cylinder moves down
003:WAIT   1,   1                        ; Waits until the air cylinder moves down
004:DO     1,   0                        ; Chuck opens
005:TIMR   100                           ; Delays for one second
006:DO     0,   0                        ; Air cylinder moves up
007:WAIT   0,   1                        ; Waits until the air cylinder moves up
```

8-5-6 Switching the program from I/O

The SRCP series controller does not accept dedicated command inputs for program switching. To switch the program through the I/O, use the program selection signal as a conditional jump input as explained below.
The following method is an example for switching among 16 kinds of programs.

Parameter

Since the number of programs to be selected is 16, set the PRM8 (No. of conditional input points) to 4.

⚠ ***CAUTION***
In actual programming, PRM8 must be set to match the number of programs you use. (See the table at the right.)

Number of programs to be selected	Number of DI points	DI numbers to be used
2 or less	1	DI0
4 or less	2	DI0 to DI1
8 or less	3	DI0 to DI2
16 or less	4	DI0 to DI3
32 or less	5	DI0 to DI4
64 or less	6	DI0 to DI5
100 or less	7	DI0 to DI6

General-purpose input

DI0	Program selection 0
DI1	Program selection 1
DI2	Program selection 2
DI3	Program selection 3
DI7	Confirmation of selected program

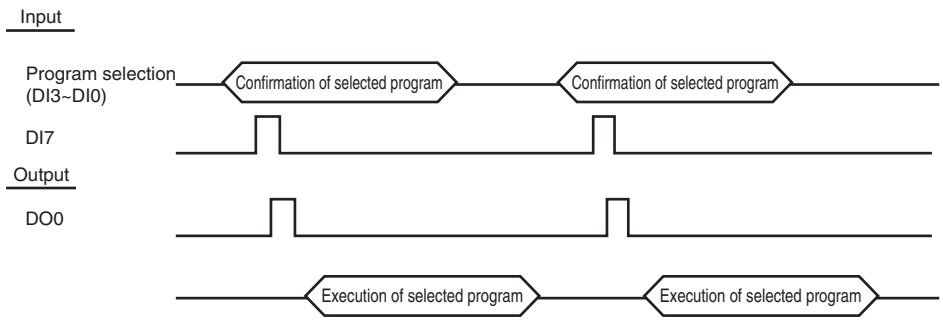
General-purpose output

DO0	Program selection start

<<Program description>>

When using the JMPF statement to select a program, select the general-purpose input/output points (DI7 and DO0 in this case) one at a time and perform the handshake. This is for synchronizing the SRCP controller program with an external device such as a PLC. If this part is omitted, the wrong program might be selected during program selection with the JMPF statement.
In specific operations, an external device should turn on DI7 after confirming DI3 to DI0. The SRCP controller then turns on DO0 just after detecting that DI7 is on, and informs the external device that the program is being selected. When the external device detects that DO0 is on, DI7 should turn off. (DI3 to DI0 should be retained.) Then, when DO0 turns off, this means that the program selection is complete, so it is okay to change DI3 to DI0. The program selection is now complete and actual program operations are executed.
When each selected program has been executed, the operation returns to the top of the program (L0 in the program NO0). The operation returns to the top of the program when there is no program matching the program selection input.

Time chart



Program **Comment**

[NO0]

```
001:L      0                ; Label definition
002:WAIT   7,   1           ; Waits for confirmation ON of the selected program
003:DO     0,   1           ; Program selection start turns on
004:WAIT   7,   0           ; Waits for confirmation OFF of the selected program
005:JMPF   1,   1,     0    ; Jumps to L1 of NO1 when input is 0
006:JMPF   1,   2,     1    ; Jumps to L1 of NO2 when input is 1
007:JMPF   1,   3,     2    ; Jumps to L1 of NO3 when input is 2
008:JMPF   1,   4,     3    ; Jumps to L1 of NO4 when input is 3
009:JMPF   1,   5,     4    ; Jumps to L1 of NO5 when input is 4
010:JMPF   1,   6,     5    ; Jumps to L1 of NO6 when input is 5
011:JMPF   1,   7,     6    ; Jumps to L1 of NO7 when input is 6
012:JMPF   1,   8,     7    ; Jumps to L1 of NO8 when input is 7
013:JMPF   1,   9,     8    ; Jumps to L1 of NO9 when input is 8
014:JMPF   1,   10,    9    ; Jumps to L1 of NO10 when input is 9
015:JMPF   1,   11,    10   ; Jumps to L1 of NO11 when input is 10
016:JMPF   1,   12,    11   ; Jumps to L1 of NO12 when input is 11
017:JMPF   1,   13,    12   ; Jumps to L1 of NO13 when input is 12
018:JMPF   1,   14,    13   ; Jumps to L1 of NO14 when input is 13
019:JMPF   1,   15,    14   ; Jumps to L1 of NO15 when input is 14
020:JMPF   1,   16,    15   ; Jumps to L1 of NO16 when input is 15
021:JMP    0,   0            ; Returns to L0 of program NO0
```

Handshaking

[NO1]

```
001:L      1                ; Label definition
002:DO     0,   0           ; Program selection is complete (selection start OFF)
    - - - - -               ; Actual program operation
    JMP    0,   0           ; Returns to L0 of program NO0
```

[NO2]

```
001:L      1                ; Label definition
002:DO     0,   0           ; Program selection is complete (selection start OFF)
    - - - - -               ; Actual program operation
    JMP    0,   0           ; Returns to L0 of program NO0
```

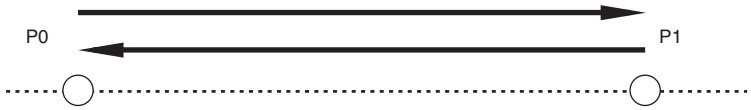
```
   :                        Programs NO3–NO15 should be created in the same way
   :
```

[NO16]

```
001:L      1                ; Label definition
002:DO     0,   0           ; Program selection is complete (selection start OFF)
    - - - - -               ; Actual program operation
    JMP    0,   0           ; Returns to L0 of program NO0
```

8-5-7 Axis movement and I/O multi-task

The robot moves between two points and performs multi-task I/O operation in asynchronous mode.



General-purpose input/output

DI0	Job status detection
DO0	Job owner's output

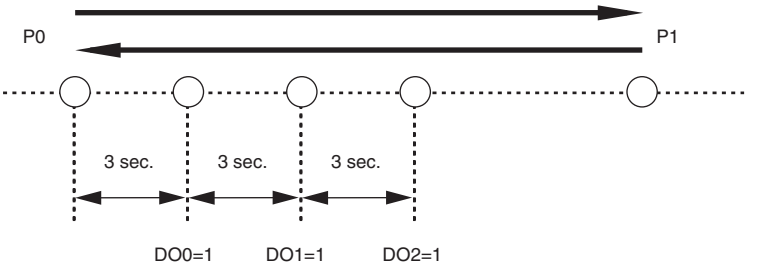
Program [NO0]

```
001:TON    1,   1,     0    ; Starts program NO1 as task 1
002:L      0                ; Label definition
003:MOVA   0,   100         ; Moves to P0 at speed 100
004:TIMR   100              ; Delays for one second
005:MOVA   1,   100         ; Moves to P1 at speed 100
006:TIMR   100              ; Delays for one second
007:JMP    0,   0           ; Returns to L0
```

Program [NO1]

```
001:L      0                ; Label definition
002:WAIT   0,   0           ; Waits until the job is finished
003:DO     0,   1           ; Issues the job start instruction
004:WAIT   0,   1           ; Confirms that the job has started
005:DO     0,   0           ; Turns off the job start signal
006:JMP    0,   1           ; Returns to L0
```

8-5-8 Turning ON general-purpose outputs during robot movement after a certain time has elapsed



Point

P0	Start position
P1	Target position

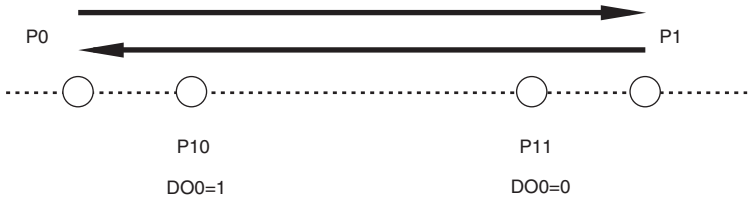
Program [NO0]

```
001:L       0                   ; Label definition
002:MOVA    0,    100           ; Moves to P0 at speed 100
003:DO      0,    0             ; Turns DO0 off
004:DO      1,    0             ; Turns DO1 off
005:DO      2,    0             ; Turns DO2 off
006:TON     1,    1,      0     ; Starts program NO1 as task 1
007:MOVA    1,    10            ; Moves to P1 at speed 10
008:JMP     0,    0             ; Returns to L0
```

Program [NO1]

```
001:TIMR    300                 ; Delays for 3 seconds
002:DO      0,    1             ; Turns DO0 on
003:TIMR    300                 ; Delays for 3 seconds
004:DO      1,    1             ; Turns DO1 on
005:TIMR    300                 ; Delays for 3 seconds
006:DO      2,    1             ; Turns DO2 on
```

8-5-9 Turning ON a general-purpose output during robot movement when it has passed a specified position



Point

P0	Start position
P1	Target position
P10	Position at DO0=1
P11	Position at DO0=0

■ When P1 is nearer to the plus side than P0:

Program [NO0] **Comment**

```
001:L      0                ; Label definition
002:MOVA   0,   100         ; Moves to P0 at speed 100
003:TON    1,   1,     0    ; Starts program NO1 as task 1
004:MOVA   1,   10          ; Moves to P1 at speed 10
005:JMP    0,   0           ; Returns to L0
```

Program [NO1] **Comment**

```
001:DO     0,   0           ; Turns DO0 off
002:P      10               ; Sets the point variable to 10
003:L      0                ; Label definition
004:JMPP   0,   1           ; Jumps to L0 when the robot does not reach P10
005:DO     0,   1           ; Turns DO0 on
006:P      11               ; Sets the point variable to 11
007:L      1                ; Label definition
008:JMPP   1,   1           ; Jumps to L1 when the robot does not reach P11
009:DO     0,   0           ; Turns DO0 off
```

MEMO

Chapter 9 OPERATING THE ROBOT

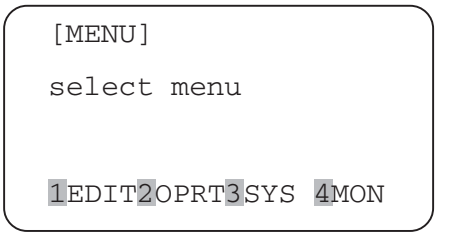
This chapter describes how to actually operate the robot. If the program has already been completed, you will be able to operate the robot by the time you finish reading this chapter.
There are two types of robot operation: step and automatic. In step operation, the program is executed one step at a time, with a step being carried out each time the RUN key on the TPB is pressed. This is used when you want to check the program as it is being carried out. In automatic operation, the entire program is executed without stopping, from beginning to end.
This chapter also covers how to initiate and recover from an emergency stop.

9-1 Performing Return-to-Origin

The stroke-end detection is used as the origin (reference point) detection method.
The following explains the procedure to perform return-to-origin using the stroke-end detection.

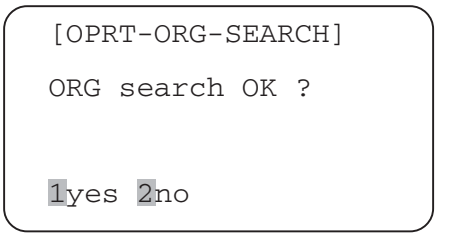
The magnetic pole is detected simultaneously with return-to-origin operation. Each time the power is turned on, return-to-origin becomes incomplete. Always perform return-to-origin after turning on the power to the controller, before starting operation. Return-to-origin also becomes incomplete after a parameter related to the origin position is changed. Return-to-origin must also be reperformed in that case.

1) On the initial screen, press **F2** (OPRT).

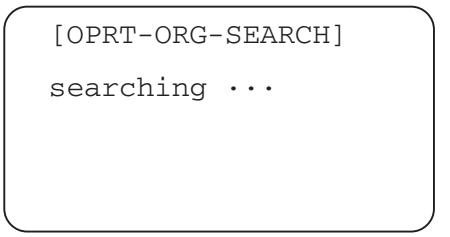


2) Next, press **F1** (ORG).

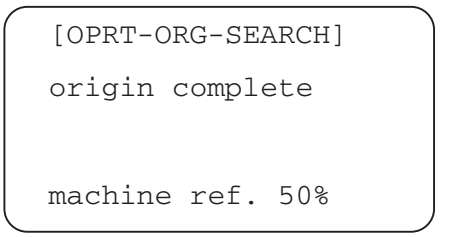
3) To perform return-to-origin, press **F1** (yes). To cancel the operation, press **F2** (no).



4) This screen is displayed during return-to-origin. Pressing **STOP** during the operation brings the robot to a halt and displays a message. Then, pressing the **ESC** key returns to the screen of step 2.



5) When return-to-origin is completed normally, the machine reference appears on the lower right of the screen. Pressing the **ESC** key returns to the screen of step 2.



⚠ ***CAUTION***

When the SERVICE mode function is enabled, the following safety control will function. (See "10-4 SERVICE mode function".)

- ***Return-to-origin movement speed is limited to 10mm/s or less in "SERVICE mode state" when the robot movement speed limit is enabled.***
- ***If the hold-to-run function is enabled, robot movement stops upon releasing F1 (yes) in step 3 in "SERVICE mode state". (You must hold down F1 (yes) until return-to-origin is complete.)***

⚠ ***CAUTION***

If the robot is operated while return-to-origin is still incomplete, the thrust necessary to move the robot cannot be obtained and causes an alarm or abnormal operation. Always perform return-to-origin before starting the robot operation.

⚠ ***CAUTION***

When performing return-to-origin by the stroke-end detection method, do not interrupt the return-to-origin operation while detecting the origin (while contacting the mechanical limit). Otherwise, the operation will stop due to a controller overload alarm and the power will have to be turned on again.

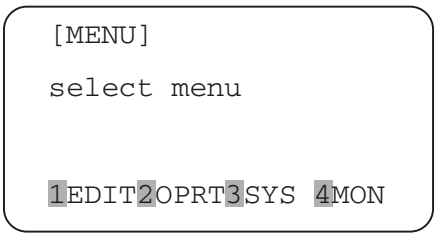
⚠ ***CAUTION***

Do not continuously repeat return-to-origin operations. If return-to-origin operation using the stroke-end detection method must be repeated, wait at least 5 seconds before repeating it.

9-2 Using Step Operation

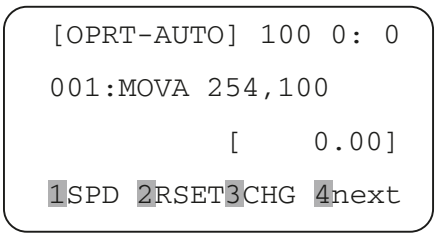
The following procedure explains how to perform step operation. In the case of a multi-task program, only the task currently selected is executed in step operation.

1) On the initial screen, press **F2** (OPRT).

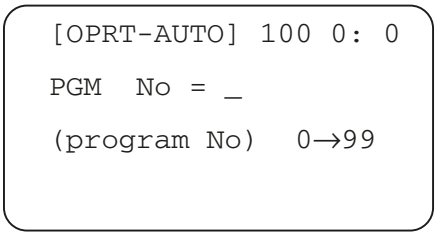


2) Next, press **F2** (STEP).

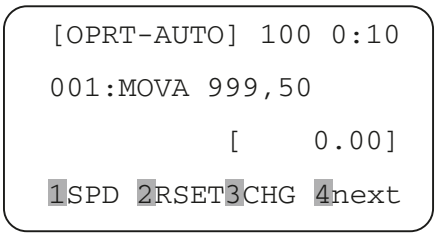
3) If the program number displayed on the screen is not the one to be run, press **F3** (CHG).



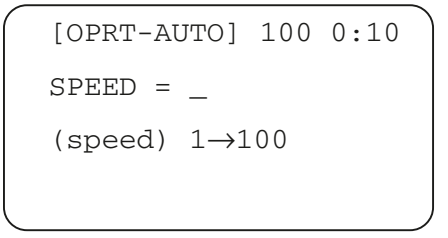
4) Using the number keys, enter the number of the program to be executed, and then press ➡.



5) The first step of the selected program is displayed on the screen. To change the execution speed, press **F1** (SPD).



6) Enter the execution speed using the number keys, and press ➡.



7) The screen returns to step 5. Pressing RUN at this point executes the first step.

```
[OPRT-STEP] 50   0:10
001:MOVA 999,50
           [    0.00]
1SPD 2RSET3CHG 4next
```

8) This screen is displayed while the program is being executed.

```
[OPRT-STEP]
     running ...
```

9) Pressing STOP during execution brings the robot to a halt and displays a message on the screen. To return to step 7, press the ESC key. Pressing RUN again executes the interrupted step.

```
[OPRT-STEP] 50  0:10
001:MOVA 999,50
           [  201.11]
1SPD 2RSET3CHG 4next
```

10) When execution is finished, the second step is displayed. Each time RUN is pressed from this point on, the currently displayed step is executed. When the last step has been executed, the message "program end" is displayed. To return to the first step from the program end, press the ESC key.

```
[OPRT-STEP]

60:program end
```

11) To switch the execution task in a multi-task program, press F4 (next) to change the function menu display and then press F3 (CHGT).

```
[OPRT-STEP] 50  1:11
001:WAIT 0   ,1
           [  250.00]
1VAL 2S_ON3CHGT4next
```

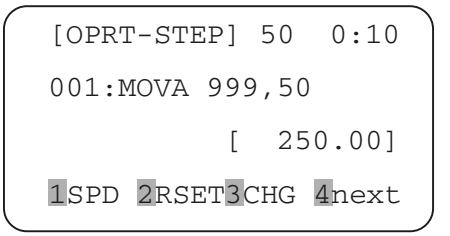
12) Each time you press F3 (CHGT), the task currently in progress is switched. When the task you want to execute is selected, press RUN to execute the step displayed for the selected task.

```
[OPRT-STEP] 50  2:12
001:DO   1   ,1
           [  250.00]
1VAL 2S_ON3CHGT4next
```

13) To return to the first step of the program from any other step and initiate execution again, press F2 (RSET).

```
[OPRT-STEP] 50  0:10
035:TIMR 100
           [  250.00]
1SPD 2RSET3CHG 4next
```

14) The screen returns to step 5, and the process is repeated from that point.



⚠ ***CAUTION***

When the SERVICE mode function is enabled, the following safety control will function. (See "10-4 SERVICE mode function".)

- ***Step operation cannot be performed in "SERVICE mode state" when automatic operation and step operation are prohibited.***
- ***Robot movement speed is limited to 3% or less of maximum speed in "SERVICE mode state" when the robot movement speed limit is enabled.***
- ***If the hold-to-run function is enabled, step operation stops upon releasing RUN in "SERVICE mode state". When one step has been executed, you must release RUN and then press RUN again to execute the next step.***

9-3 Using Automatic Operation

The following procedure explains how to perform automatic operation. All the tasks started in a multi-task program are executed by automatic operation.

1) On the initial screen, press [F2] (OPRT).

```
[MENU]
select menu


1EDIT2OPRT3SYS 4MON
```

2) Next, press [F3] (AUTO).

```
[OPRT]
select menu


1ORG 2STEP3AUTO
```

3) If the program number displayed on the screen is not the one to be run, press [F3] (CHG).

```
[OPRT-AUTO] 100 0: 0
001:MOVA 254,100
           [    0.00]
1SPD 2RSET3CHG 4next
```

4) Using the number keys, enter the number of the program to be executed and then press ➡.

```
[OPRT-AUTO] 100 0: 0
PGM  No = _
(program No) 0→99
```

5) The first step of the selected program is displayed on the screen. To change the execution speed, press [F1] (SPD).

```
[OPRT-AUTO] 100 0:10
001:MOVA 999,50
           [    0.00]
1SPD 2RSET3CHG 4next
```

6) Enter the execution speed with the number keys and press ➡.

```
[OPRT-AUTO] 100 0:10
SPEED = _
(speed) 1→100
```

7) The screen returns to step 5. Pressing [RUN] at this point executes the program all the way to the last step.

```
[OPRT-AUTO] 50  0:10
001:MOVA 999,50
           [    0.00]
1SPD 2RSET3CHG 4next
```

8) This is the screen displayed while the program is being executed.

```
[OPRT-AUTO]
     running ···
```

9) Pressing STOP during execution brings the robot to a halt and displays the message "stop key". Press the ESC key to display the step where execution was interrupted.

Pressing RUN will cause execution to resume from the step where it was interrupted. When the last step has been executed, the message "program end" is displayed. Pressing the ESC key returns the screen to that shown in Step 7.

```
[OPRT-AUTO]


60:program end
```

10) To switch to the display of another task in a multi-task program, press F4 (next) to change the function menu display and then press F3 (CHGT).

```
[OPRT-AUTO] 50  1:11
010:WAIT 0   ,1
           [  250.00]
1VAL 2S_ON3CHGT4next
```

11) Each time you press F3 (CHGT), the task display is switched.

```
[OPRT-AUTO] 50  2:12
015:DO   1   ,1
           [  250.00]
1VAL 2S_ON3CHGT4next
```

12) To return to the first step of the program from any other step and initiate execution again, press F2 (RSET).

```
[OPRT-AUTO] 50  0:10
035:TIMR 100
           [  250.00]
1SPD 2RSET3CHG 4next
```

13) The screen returns to step 5, and the process is repeated from that point.

```
[OPRT-AUTO] 50  0:10
001:MOVA 999,50
           [  250.00]
1SPD 2RSET3CHG 4next
```

⚠ ***CAUTION***

When the SERVICE mode function is enabled, the following safety control will function. (See "10-4 SERVICE mode function".)

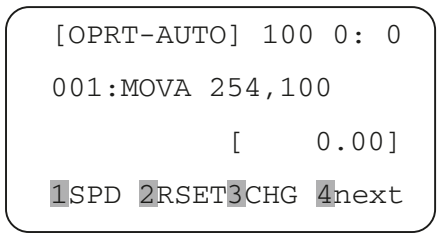
- ***Automatic operation cannot be performed in "SERVICE mode state" when automatic operation and step operation are prohibited.***
- ***Robot movement speed is limited to 3% or less of maximum speed in "SERVICE mode state" when the robot movement speed limit is enabled.***
- ***If the hold-to-run function is enabled, automatic operation stops upon releasing RUN in "SERVICE mode state".***

9-4 Switching the Execution Program

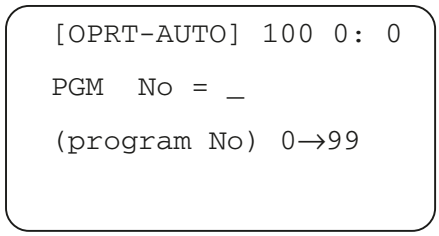
The following procedure explains how to switch the program in automatic operation. Use the same procedure in step operation.
The program selected by this procedure will be the lead program to which the execution sequence always returns after program reset.
When the program is switched, reset is automatically performed, so all general-purpose outputs are turned off.

* As exceptions, DO4 does not turn OFF when PRM33 (operation at return-to-origin complete parameter) is set to 1 or 3, and DO3 does not turn OFF when PRM46 (servo status output parameter) is set to 1.

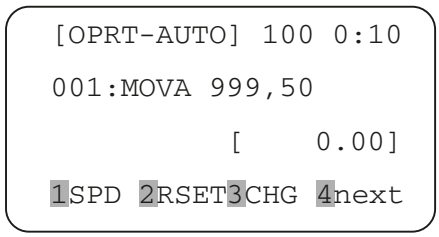
1) If the program number displayed on the screen is not the one to be run, press F3 (CHG).



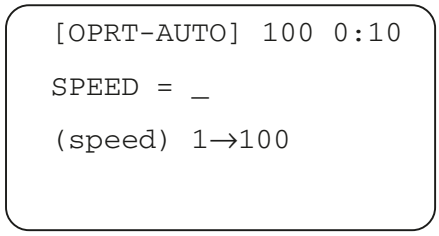
2) Using the number keys, enter the number of the program to be executed and then press ➡.



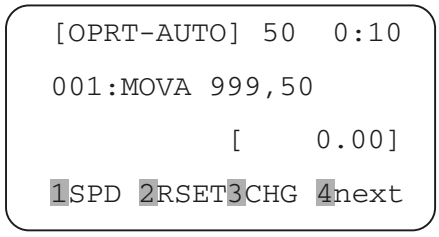
3) The first step of the selected program is displayed on the screen. To change the execution speed, press F1 (SPD).



4) Enter the execution speed using the number keys and press ➡.



5) The screen returns to step 3.



9-5 Emergency Stop Function

There are two ways to trigger emergency stop on the SRCP controller. One way is by using the push-button on the TPB. The other is to use the I/O emergency stop input. In either case for safety reasons, a contact B (normally closed) input is used (when the contact is opened, emergency stop is triggered). The SRCP controller can recover from an emergency stop condition without turning off the power so return-to-origin is not necessary.
This section explains how to initiate and recover from an emergency stop using the TPB.

9-5-1 Initiating an emergency stop

If for any reason you want to immediately stop the robot while operating it with the TPB, press the emergency stop button on the TPB. The emergency stop button locks in the depressed position, and can be released by turning it to the right.
In emergency stop, the robot assumes a "free" state so that commands initiating robot motion (for example, return-to-origin command) cannot be executed.

9-5-2 Recovering from an emergency stop

When recovery from an emergency stop is required during TPB operation, that procedure automatically appears on the TPB. Follow those instructions to reset the emergency stop condition.

Recovery from an emergency stop is required during TPB operation when you are going to:

- Perform return-to-origin.
- Run step operation.
- Run automatic operation.
- Edit point data using teaching playback.
- Exit the direct teaching mode.

The following steps explain the procedure for running step operation after emergency stop.
As this example shows, the emergency stop condition cannot be cancelled by just releasing the emergency stop button.

1) Press RUN to start operation.

```
[OPRT-STEP] 100 0: 7
001:MOVA 254,100
          [    0.00]
1SPD 2RSET3CHG 4next
```

2) Following the message displayed on the screen, release the emergency stop button.

```
[OPRT-STEP] 100 0: 7
release EMG.button
```

3) After the emergency stop is released, a message appears asking whether to turn the servo on.

   To turn the servo on, press F1 (yes).

   To leave the servo off , press F2 (no).

```
[OPRT-STEP] 100 0: 7
servo on ready ?

1yes 2no
```

4) Then, another message appears asking if ready to operate.

   To restart operation, press F1 (yes).

   To cancel restarting, press F2 (no).

```
[OPRT-STEP] 100 0: 7
continue OK ?

1yes 2no
```

5) Operation starts when F1 (yes) was pressed in step 4.

   If F2 (no) was pressed, the screen returns to step 1.

```
[OPRT-STEP] 100 0: 7
001:MOVA 254,100
           [    0.00]
1SPD 2RSET3CHG 4next
```

⚠ ***CAUTION***

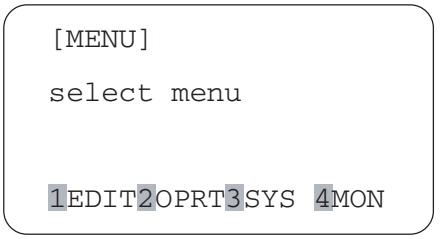
When the SERVICE mode function is enabled, the following safety control will function. (See "10-4 SERVICE mode function".)

- ***If the hold-to-run function is enabled, the robot stops upon releasing F1 (yes) in step 4 in "SERVICE mode state".***

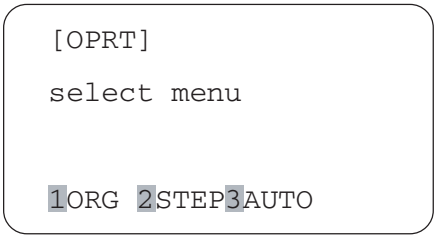
9-6 Displaying the Memory I/O Status

The memory I/O status can be displayed on the screen.

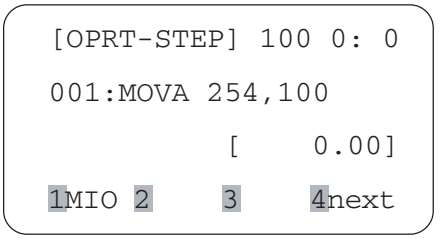
1) On the initial screen, press F2 (OPRT).



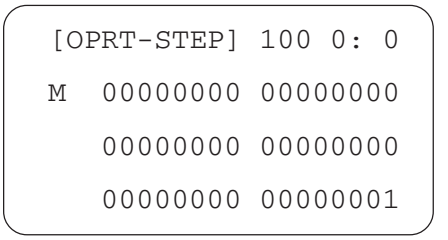
2) Press F2 (STEP) or F3 (AUTO).
The STEP or AUTO mode screen appears. The following steps are explained using the STEP mode screen.



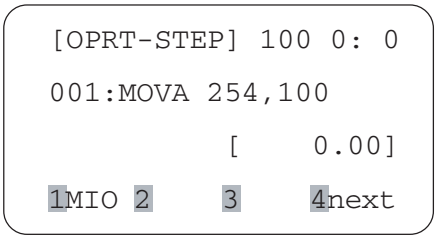
3) Press F4 (next) twice to change the menu display and then press F1 (MIO).



4) The I/O status of each memory is displayed.
From the left, the top line shows the status from 115 to 100, the middle line from 131 to 116, and the bottom line from 147 to 132.



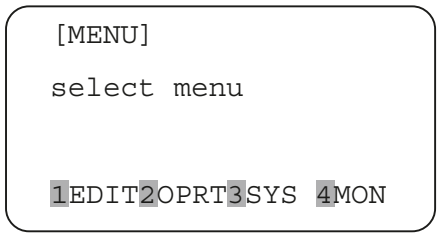
5) To return to the previous screen, press ESC.



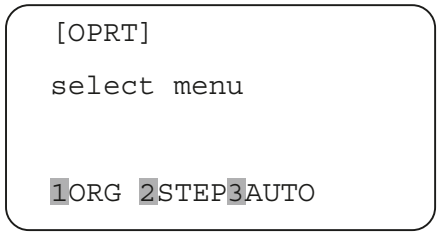
9-7 Displaying the Variables

The point data variable "P", counter array variable "C" and counter variable "D" values can be displayed on the TPB screen.

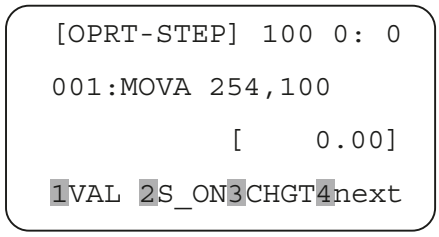
1) On the initial screen, press F2 (OPRT).



2) Press F2 (STEP) or F3 (AUTO).
The following explains the procedure for displaying the variables on the screens in step operation.

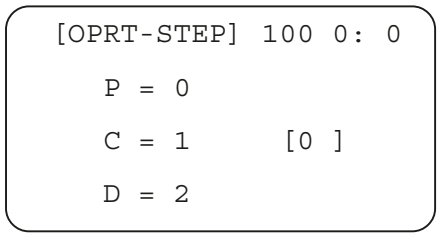


3) Press F4 (next) to change the menu display and then press F1 (VAL).

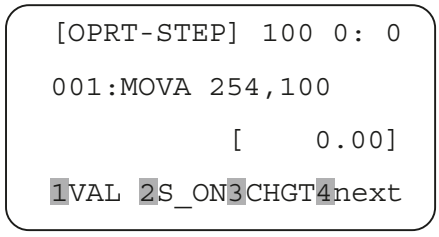


4) Continue to indicate the value of each variable.

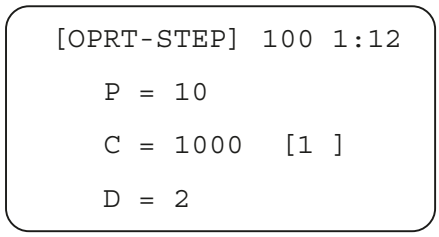
The item enclosed by brackets [ ] is an array element number selected with the CSEL statement.



5) To return to the previous screen, press ESC.



6) To display the variable of another task in a multi-task program, press F3 (CHGT) in step 3 so that the task is switched before pressing F1 (VAL).



MEMO

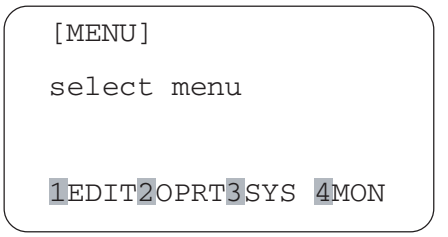
Chapter 10 OTHER OPERATIONS

The TPB has many convenient functions in addition to those already covered. For example, memories can be initialized, and options such as memory cards can be used. This chapter will describe these additional functions

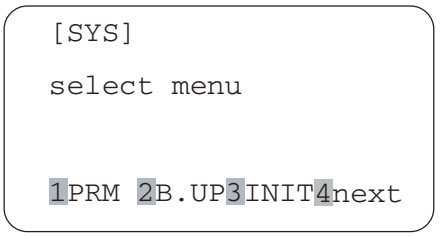
10-1 Initialization

Initializing the programs and points erases all the program data and point data currently stored in the controller.
Initializing the parameters resets the parameters to their initial values.

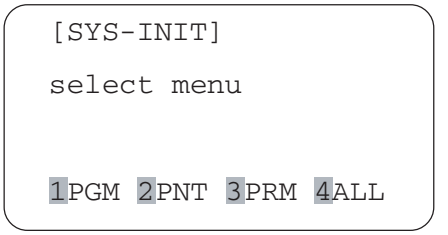
1) On the initial screen, press F3 (SYS).



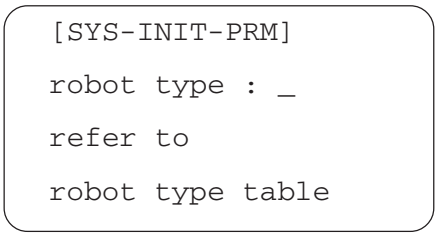
2) Next, press F3 (INIT).



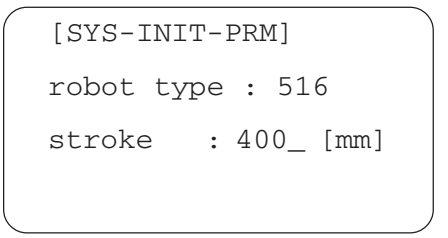
3) Select the data to be initialized.
   To initialize the program data, press F1 (PGM).
   To initialize the point data, press F2 (PNT).
   To initialize the parameter data, press F3 (PRM).
   To initialize all of the program, point and parameter data, press F4 (ALL).



4) If F3 (PRM) or F4 (ALL) was selected in step 3, the robot type must be specified.
   Enter the robot number with the number keys and then press ➡.
   To find the robot number, see "15-1-2 Robot number list".



5) Next, enter the robot stroke length.
   Enter the stroke length with the number keys and then press ➡.



6) Finally, enter the robot payload.
   Enter the payload with the number keys and then press ➡.

```
[SYS-INIT-PRM]
robot type : 516
stroke   : 400  [mm]
weight   : 3_   [kg]
```

7) A confirmation message appears on the screen.
   To execute the initialization, press F1 (yes).
   To cancel the initialization, press F2 (no).

```
[SYS-INIT-PRM]
parameter data
initialize OK ?
1yes 2no
```

8) When initialization is complete, the screen returns to step 3.

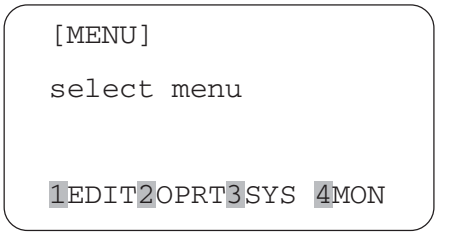
```
[SYS-INIT]
select menu


1PGM 2PNT 3PRM 4ALL
```

10-2 DIO Monitor Display

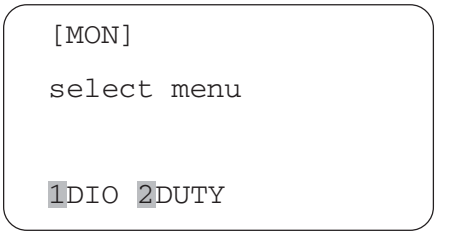
Data indicating whether the I/O signals are on or off can be displayed on the screen. The operation procedure is explained below.

10-2-1 Display from the monitor menu

1) On the initial screen, press F4 (MON).

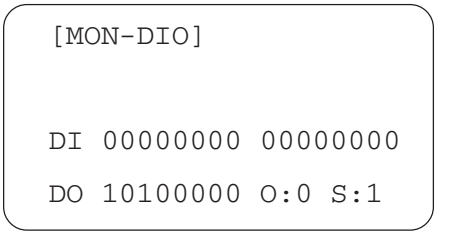


2) Next, press F1 (DIO).

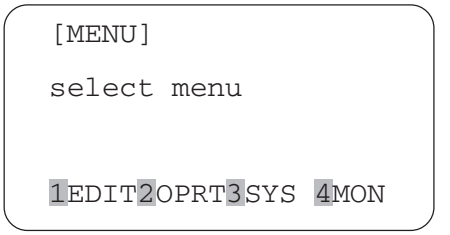


3) The ON/OFF status of I/O signals is displayed.

   For information about what the display shows, refer to "4-3-4 DIO monitor screen".



4) To return to the initial screen, press ESC twice.



10-2-2 Display from the DIO key operation

1) Hold down the DIO key.

```
[OPRT-AUTO]
     running...
```

2) The ON/OFF status of I/O signals is displayed as long as the key is held down.
   For information about what the display shows, refer to "4-3-4 DIO monitor screen".

```
[OPRT-AUTO]
     running...
DI 00000000 00000000
DO 11100000 O:0 S:1
```

3) Releasing the key returns the screen to the previous screen.

```
[OPRT-AUTO]
     running...
```

⚠ ***CAUTION***
The DIO Monitor key does not function during system operation.

10-3 System Information Display

1) On the initial screen, press the ESC key.

```
[MENU]
select menu


1EDIT2OPRT3SYS 4MON
```

2) The controller version number, TPB version number, and robot type are displayed. The screen returns to the initial screen after approximately two seconds.

```
[INFORMATION]
controller  V24.00
TPB         V12.50
robot type    516
```

10-4 SERVICE mode function

The SERVICE mode function is explained in this section.

The robot operator or others sometimes need to enter the hazardous area in the robot safety enclosure and move the robot to perform maintenance or adjustment while using the TPB. This situation is referred to as "SERVICE mode state" and requires extra caution. Limits should be placed on controller operation at this time to ensure operator safety.
A safety function called "SERVICE mode function" places limits on controller operation when in "SERVICE mode state".

When the SERVICE mode function is enabled, the SRCP controller constantly monitors status to check whether "SERVICE mode state" occurs. In "SERVICE mode state", the SERVICE mode function does the following:

- Limits command input from any device other than the TPB.
- Limits robot movement speed.
- Prohibits automatic operation and step operation.
- Enables hold-to-run function.

The controller recognizes "SERVICE mode state" when the SERVICE mode function is enabled and the SERVICE mode input (SVCE) is OFF (contact is open). (See "3-2-3 SERVICE mode input (SVCE)".)

NOTE

The SERVICE mode function is protected by a password so that the settings cannot be changed easily.

10-4-1 Safety settings for SERVICE mode

Safety controls that work in "SERVICE mode state" are explained in detail below.

■ Limiting command input from any device other than TPB

When the operator is working within the robot safety enclosure using the TPB, permitting any command input from devices (such as via I/O) other than the TPB is very hazardous to the TPB operator.

(For example, a hazardous situation may occur if someone outside the safety enclosure runs an automatic operation start command (AUTO-R) without letting the TPB operator know about it.)

To avoid this kind of hazard, the TPB can only be used to operate the robot in "SERVICE mode state", and all other device command inputs are disabled.

However, this limitation can be cancelled even in "SERVICE mode state" under the user's responsibility.

Setting value	Details
0	Only commands input from the TPB are permitted in SERVICE mode state.
1	Only commands input from the TPB and parallel I/O are permitted in SERVICE mode state.
2	Only commands input from the TPB and option unit are permitted in SERVICE mode state.
3	Command inputs are not limited even in SERVICE mode state.

■ Limiting the robot movement speed

Moving the robot at a high speed while an operator is working within the robot safety enclosure is very dangerous to that operator. Setting the robot movement speed to a safety speed of 250mm/s or less is advisable because most robot operation while the operator is working within the safety enclosure is for maintaining or adjusting the robot. In view of this, the robot movement speed in "SERVICE mode state" is limited to below 3% of maximum speed.

However, this speed limitation can be cancelled even in "SERVICE mode state" under the user's responsibility.

Setting value	Details
0	The robot movement speed is limited to 3% or less of maximum speed in SERVICE mode state.
1	The robot movement speed is not limited even in SERVICE mode state.

■ Prohibiting the automatic operation and step operation

Running an automatic operation or step operation while an operator is working within the robot safety enclosure is very dangerous to that operator.

(For example, when the operator is in the safety enclosure, a hazardous situation may occur if someone runs a robot program without letting the operator know about it.)

To avoid this kind of hazard, automatic operation and step operation are basically prohibited in "SERVICE mode state".

However, this limitation can be cancelled even in "SERVICE mode state" under the user's responsibility.

Setting value	Details
0	Automatic operation and step operation are prohibited in SERVICE mode state.
1	Automatic operation and step operation are permitted even in SERVICE mode state.

■ Hold-to-run function

If the robot continues to move while an operator is working within the robot safety enclosure without using the TPB, the operator may be exposed to a dangerous situation.

(For example, a hazardous situation may occur if the operator working within the safety enclosure should trip or fall by accident and blackout.)

To prevent this kind of hazard, the Hold-to-Run function allows the robot to move only during the time that the TPB key is kept pressed in "SERVICE mode state".

However, this function can be cancelled even in "SERVICE mode state" under the user's responsibility.

Setting value	Details
0	Hold-to-run function works in SERVICE mode state.
1	Hold-to-run function does not work even in SERVICE mode state.

CAUTION

The above safety controls can be cancelled in part or in whole under the user's responsibility. However, extra caution must be taken to maintain safety since hazardous situations may occur.

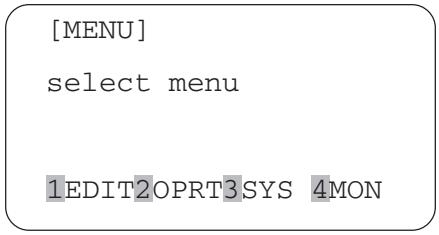
NOTE

When parameter initialization is performed, all safety control settings are initialized. (All settings will be set to "0".) However, the SERVICE mode function setting will not change even after parameter initialization.

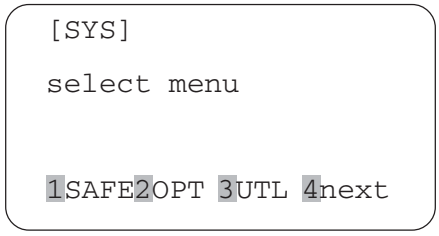
10-4-2 Enabling/disabling the SERVICE mode function

To enable or disable the SERVICE mode function, follow these steps.

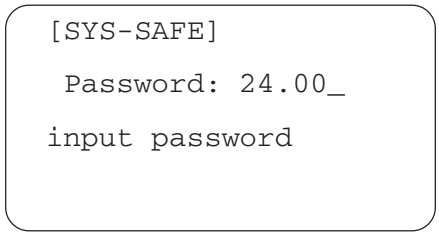
1) On the initial screen, press F3 (SYS).



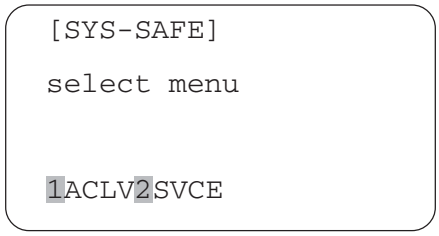
2) Press F4 (next) to change the menu display and then press F1 (SAFE).



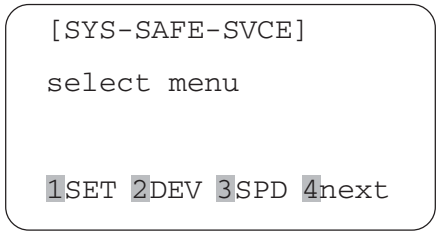
3) The password request screen appears. Enter the password and then press ➡.



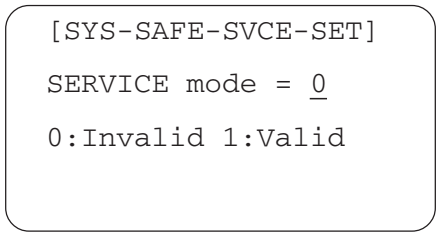
4) When the password is correct, the screen shown on the right appears.
Press F2 (SVCE) here.



5) Press F1 (SET).



6) The current SERVICE mode function setting appears.
To disable the SERVICE mode function, enter 0 with the number key. To enable it, enter 1. Then, press ➡.



7) When writing is complete, the screen returns to step 6.

```
[SYS-SAFE-SVCE-SET]
SERVICE mode = 1
0:Invalid 1:Valid
```

NOTE

The password is identical to the SRCP controller's version number. For example, if the controller version is 24.00, enter 24.00 as the password. Once the password is accepted, it will not be requested unless the TPB is disconnected from the controller or the controller power is turned off.

10-4-3 Setting the SERVICE mode functions

1) On the initial screen, press **F3** (SYS).

```
[MENU]
select menu

1EDIT2OPRT3SYS 4MON
```

2) Press **F4** (next) to change the menu display and then press **F1** (SAFE).

```
[SYS]
select menu

1SAFE2OPT 3UTL 4next
```

3) The password request screen appears. Enter the password and then press ➡.

```
[SYS-SAFE]
 Password: 24.00_
input password
```

4) When the password is correct, the screen shown on the right appears.
   Press **F2** (SVCE) here.

```
[SYS-SAFE]
select menu

1ACLV2SVCE
```

5) Select the item whose setting you want to change.
   To change the setting that limits the operation device, press **F2** (DEV).
   To change the setting that limits the speed, press **F3** (SPD).
   To change the setting that limits step operation and automatic operation, press **F4** (next) and then press **F1** (RUN).
   To change the setting for the hold-to-run function, press **F4** (next) and then press **F2** (HtoR).

```
[SYS-SAFE-SVCE]
select menu

1SET 2DEV 3SPD 4next
```

6) The current setting for the selected item appears.
   To change the setting, enter the data with the number key and then press ➡.

```
[SYS-SAFE-SVCE-DEV]
data = 0
PB only
```

7) When the setting has been changed, the memory write screen appears.
To save the change permanently (retain the change even after the controller power is turned off), press **F1** (SAVE).
To save the change temporarily (retain the change until the power is turned off), press **F2** (CHG).
To cancel changing the setting, press **F3** (CANCEL).

```
[SYS-SAFE-SVCE-DEV]
data = 1
PB/DI valid
1SAVE2CHG 3CANCEL
```

8) When writing is complete, the screen returns to step 6.

```
[SYS-SAFE-SVCE-DEV]
data = 1
PB/DI valid
```

NOTE

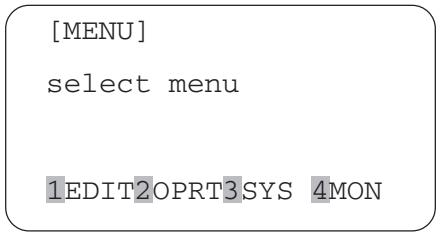
The password is identical to the SRCP controller's version number. For example, if the controller version is 24.00, enter 24.00 as the password. Once the password is accepted, it will not be requested unless the TPB is disconnected from the controller or the controller power is turned off.

10-5 System utilities

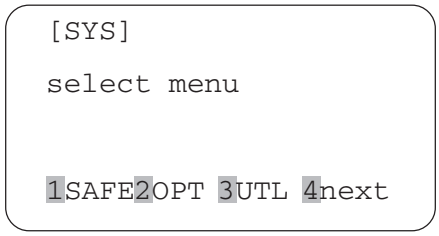
10-5-1 Viewing hidden parameters

Parameters hidden in the normal state can be viewed.
Use extra caution to avoid accidentally changing the parameters when these hidden parameters are displayed.

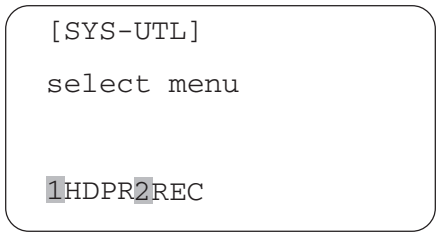
1) On the initial screen, press F3 (SYS).



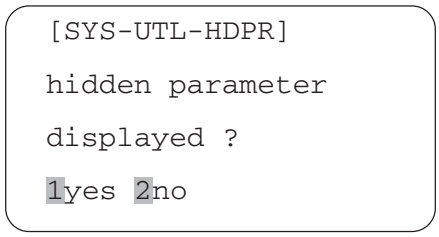
2) Press F4 (next) to change the menu display and then press F3 (UTL).



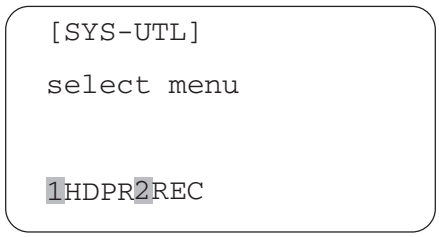
3) Press F1 (HDPR) here.



4) A confirmation message appears.
To permit display of the hidden parameters, press F1 (yes)
To hide the hidden parameters, press F2 (no).



5) The screen returns to step 3.
Display of hidden parameters is permitted until you press F1 (HDPR) and then F2 (no), or the SRCP controller is turned off, or until the TPB is disconnected.



NOTE
The hidden parameter display is also permitted by turning on the power to the controller while holding down the ESC key on the TPB, or by connecting the TPB to the controller while holding down the ESC key.

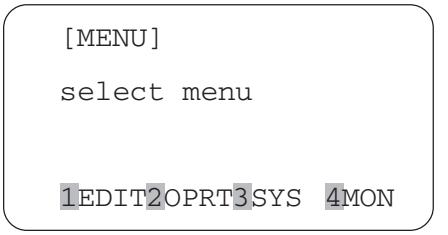
10-6 Using a Memory Card

A memory card can be used with the TPB to back up the data in the SRCP controller.
Refer to "16-1-1 Memory card" for the procedure for handling a memory card and for the number of data that can be stored.

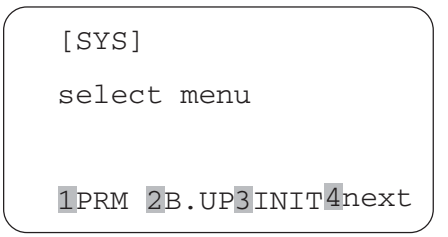
10-6-1 Saving controller data to a memory card

1) Insert the memory card into the TPB.

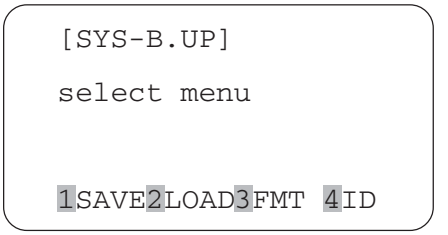
2) On the initial screen, press F3 (SYS).



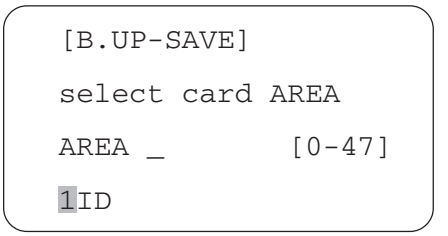
3) Next, press F2 (B.UP).



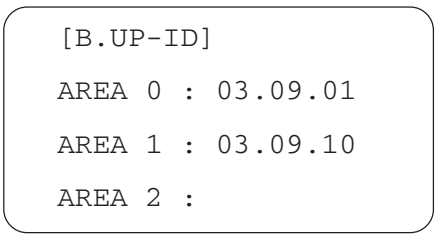
4) Press F1 (SAVE).
Press F1 (CARD) before pressing F1 (SAVE) if the SRCP controller version is 24.10 or later and the TPB version is 12.51 or later.



5) Specify the save area on the memory card.
Enter the save area with the number keys and press ➡.



6) The saved status of data on the memory card can be checked by pressing F1 (ID) in step 5.
To check the saved status in AREA 3 onward, press STEP UP or STEP DOWN to scroll the screen. To return to the screen in step 5, press ESC.



7) If data already exists in the area specified in step 5, a confirmation message appears.
To overwrite the data in the selected area, press F1 (yes).
To change the selected area, press F2 (no).

```
[B.UP-SAVE]
AREA 1 already saved
delete OK ?
1yes 2no
```

8) Set an ID number for the data being saved. Using the number keys (0 to 9), the "-" (minus) key, and the "." (period) key, enter a number of up to eight characters and then press ➡.

```
[B.UP-SAVE]AREA 1
make identification
 ID=_
effective key[0→9-.]
```

9) A confirmation message appears.
To save the data, press F1 (yes).
To cancel, press F2 (no).

```
[B.UP-SAVE]AREA 1
save OK ?
 ID=03.10.01
1yes 2no
```

10) This screen is displayed while the data is being saved.

```
[B.UP-SAVE]AREA 1
saving ...
```

11) When saving is finished, the screen returns to step 4.

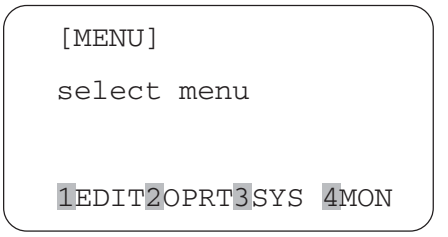
```
[SYS-B.UP]
select menu

1SAVE2LOAD3FMT 4ID
```

⚠ ***CAUTION***
Never eject the memory card during saving of data.
Do not leave the memory card inserted into the TPB when not in use. This shortens the backup battery life.

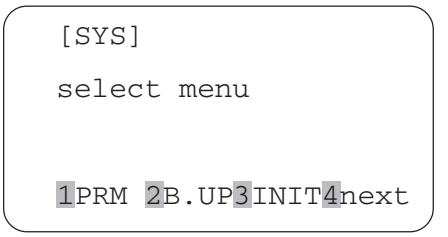
10-6-2 Loading data from a memory card

1) Insert the memory card into the TPB.

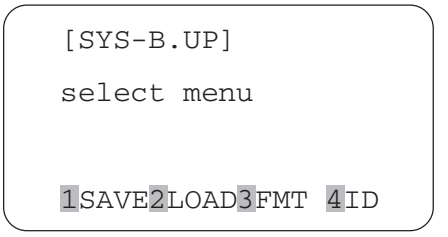
2) On the initial screen, press **F3** (SYS).



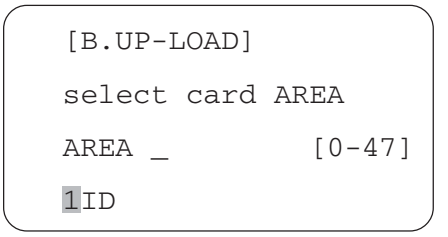
3) Next, press **F2** (B.UP).



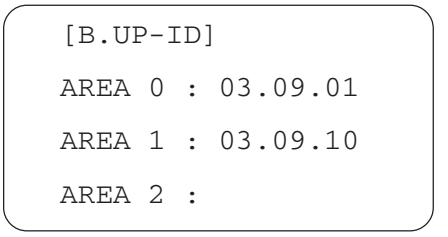
4) Press **F2** (LOAD).
Press **F1** (CARD) before pressing **F2** (LOAD) if the SRCP controller version is 24.10 or later and the TPB version is 12.51 or later.



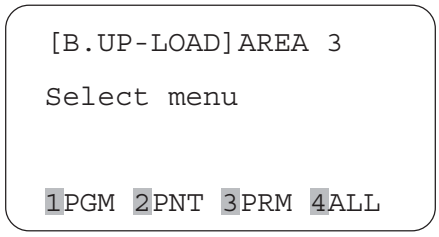
5) Specify the load area in the memory card.
Enter the load area with the number keys and press ➡.



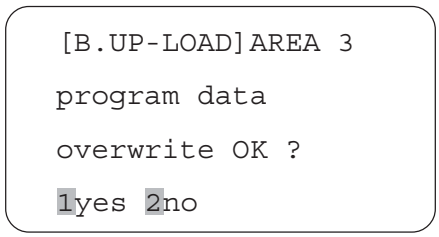
6) The saved status of data on the memory card can be checked by pressing **F1** (ID) in step 5.
To check the saved status in AREA 3 onward, press STEP UP or STEP DOWN to scroll the screen. To return to the screen in step 5, press ESC.



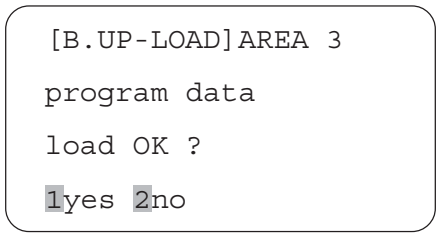
7) When the load area was selected in step 5, the data load screen appears. Select the data to be loaded.
   To load the program data, press F1 (PGM).
   To load the point data, press F2 (PNT).
   To load the parameter data, press F3 (PRM).
   To load all of the program, point and parameter data, press F4 (ALL).



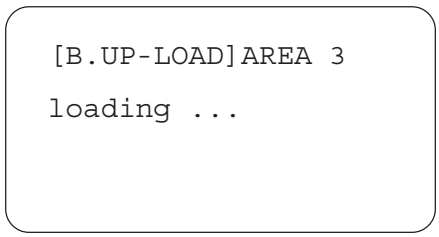
8) When F1 (PGM) or F2 (PNT) was selected in step 7, a confirmation message appears asking whether to overwrite the data.
   Pressing F1 (yes) overwrites the data only with the same program numbers or point numbers. (The previous data remains if its program or point number differs from the program or point number to be loaded.)
   Pressing F2 (no) loads the data after initializing the data in the SRCP controller.
   When F4 (ALL) was selected in step 7, all data in the SRCP controller will be initialized and then loaded.



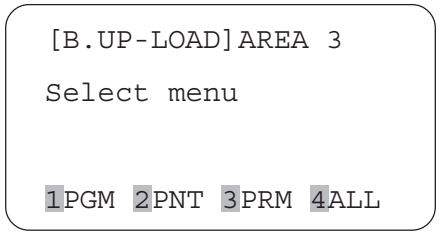
9) A confirmation message appears asking whether to load the data.
   To load the data, press F1 (yes)
   To cancel, press F2 (no).



10) This screen is displayed while data is being loaded.



11) When data loading is complete, the screen returns to step 7.



⚠ ***CAUTION***

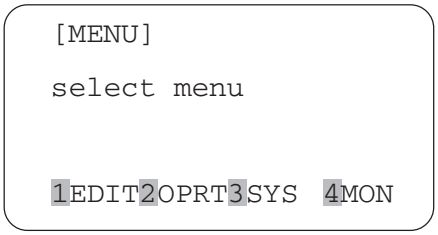
Never eject the memory card during loading of data.

Do not leave the memory card inserted into the TPB when not in use. This shortens the backup battery life.

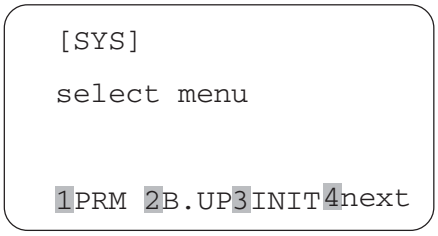
10-6-3 Formatting a memory card

1) Insert the memory card into the TPB.

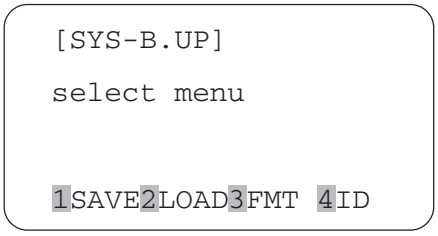
2) On the initial screen, press **F3** (SYS).



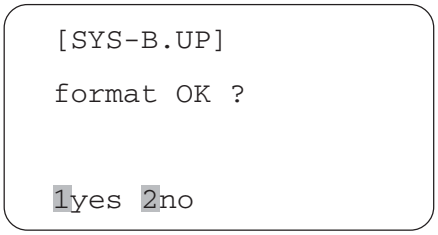
3) Next, press **F2** (B.UP).



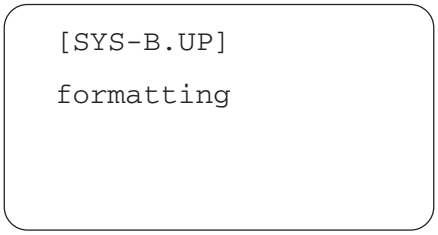
4) Press **F3** (FMT).
Press **F1** (CARD) before pressing **F3** (FMT) if the SRCP controller version is 24.10 or later and the TPB version is 12.51 or later.



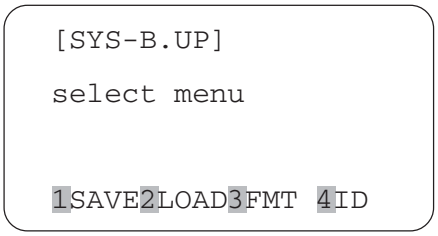
5) A confirmation message appears.
To format the memory card, press **F1** (yes).
To cancel, press **F2** (no).



6) This screen is displayed while the memory card is being formatted.



7) When formatting is complete, the screen returns to step 4.

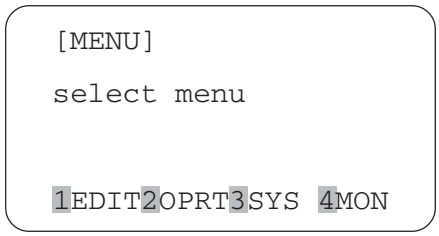


⚠ ***CAUTION***
Never eject the memory card during formatting.
Do not leave the memory card inserted into the TPB when not in use. This shortens the backup battery life.

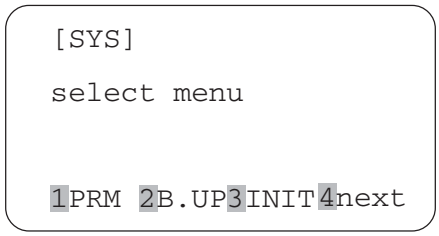
10-6-4 Viewing the ID number for memory card data

1) Insert the memory card into the TPB.

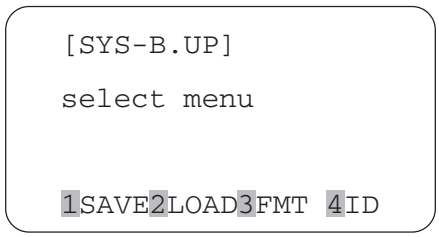
2) On the initial screen, press [F3] (SYS).



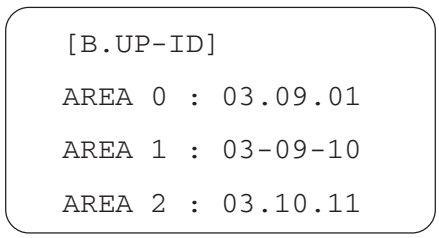
3) Next, press [F2] (B.UP).



4) Press [F4] (ID).
Press [F1] (CARD) before pressing [F4] (ID) if the SRCP controller version is 24.10 or later and the TPB version is 12.51 or later.



5) The ID number of each area is displayed on the screen.
To check the saved status in AREA 3 onward, press [STEP UP] or [STEP DOWN] to scroll the screen. To return to the screen in step 4, press (ESC).



10-7 Duty (load factor) monitor

The SRCP controller has a duty (load factor) monitor to allow you to operate the robot under the most optimal conditions. The duty monitor checks the robot's motor load factor and displays it in percent (%) versus the motor rating.

An overload error might appear if the duty exceeds 100% during robot operation. If this happens, either lower the robot acceleration or maximum speed, or increase the robot stop time (lower the duty ratio). On the other hand, if you want to shorten the cycle time even further, when there is currently no overload, you can raise the acceleration or maximum speed, or shorten the robot stop time (raise the duty ratio).

There are the following two methods to measure the duty.

Method 1: On the TPB, select DUTY mode and measure the duty during robot movement with a pulse train input (Pulse Train mode) or dedicated command input (Normal mode).

Method 2: Specify an interval in a program in which you want to measure the duty and run the program.

[Method 1]

1) Operate the robot with a pulse train input (Pulse Train mode) or dedicated command input (Normal mode).
2) On the TPB, select DUTY mode.
3) Measure the operation duty.
4) Check the measurement result.

Refer to "10-7-1 Measuring the duty (load factor)" for procedures to start and stop duty measurement and check the result.

NOTE

In method 1, the duty cannot be measured during robot movement by the TPB (RS-232C).

[Method 2]

1) Add the robot language command "DUTY 1" to the beginning of the interval in a program in which you want to measure the duty and also add the robot language command "DUTY 0" to the end of the interval.

```
005:    DO 0,1
006:    WAIT 1,1
007:    DO 0,0
008:    TIMR 100
009:    DUTY 1      ← Start operation duty measurement
010:    DO 0,0
011:    WAIT 0,1
012:    MOVA 2,100
013:    DO 0,1
014:    WAIT 1,1
015:    DO 1,0
016:    TIMR 100
017:    DUTY 0      ← Stop operation duty measurement
018:    DO 0,0
```

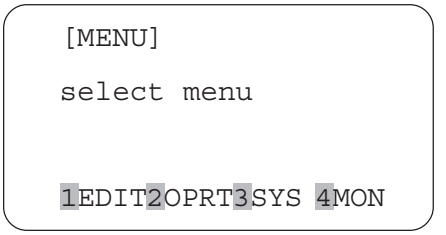
Operation duty measurement interval

2) Run the program including the operation duty measurement interval.
3) Stop (end) the program.
4) On the TPB, select DUTY mode and check the measurement result.

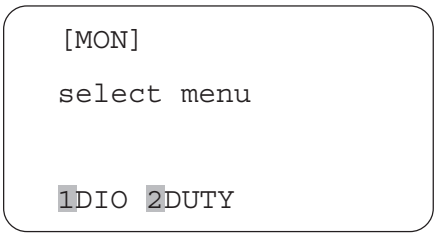
Refer to "10-7-1 Measuring the duty (load factor)" for the procedure to check the measurement result.

10-7-1 Measuring the duty (load factor)

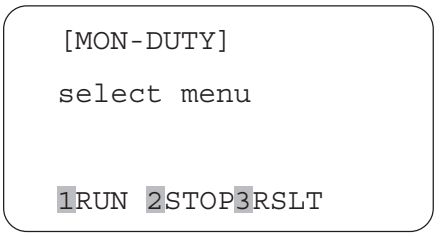
1) While moving the robot with a pulse train input (Pulse Train mode) or dedicated command input (Normal mode), press **F4** (MON) on the TPB initial menu screen to enter MON (monitor) mode.



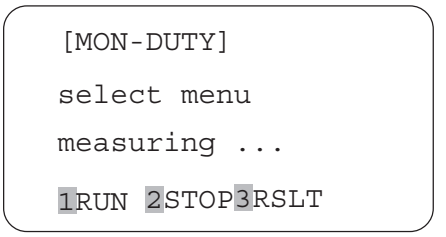
2) Next, press **F2** (DUTY).



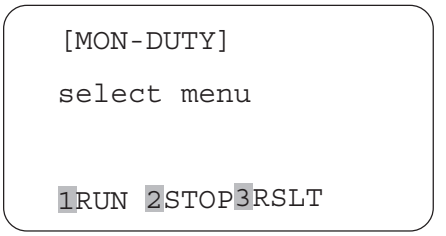
3) Press **F1** (RUN) to start measuring the operation duty.



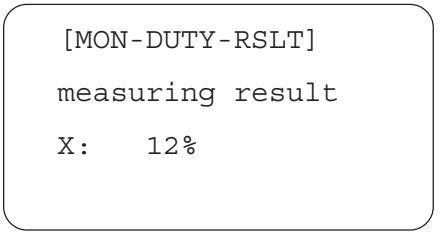
4) Press **F2** (STOP) to stop measuring the operation duty.



5) Next, press **F3** (RSLT) to display the measurement result on the TPB screen.



6) The operation duty value appears as a percent on the TPB screen.



NOTE

The operation duty can also be monitored while the program is being executed with a program command. For more information, see "10-7 Duty (load factor) monitor". The method for displaying the measurement result is the same as described above.

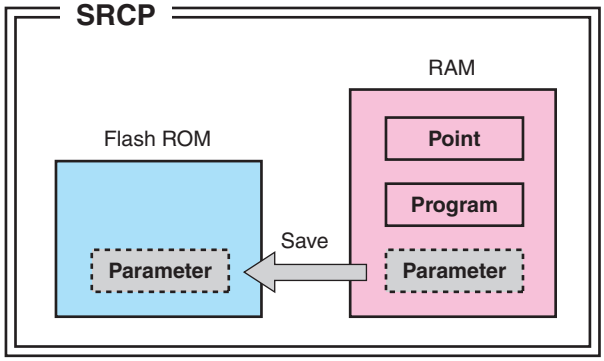
10-8 Using the internal flash ROM

When you set parameters using the TPB or POPCOM (options) or via the RS-232C, the parameter data stored in the RAM inside the SRCP is rewritten and the robot operates based on this parameter data written in the RAM. The SRCP also has an internal flash ROM for backup of this parameter data in the RAM. The parameter data backed up in the flash ROM can be loaded back into the RAM.

NOTE
The SRCP system backup battery retains the data in the RAM.

NOTE
The internal flash ROM can be used when the SRCP controller version is 24.10 or later and the TPB version is 12.51 or later.

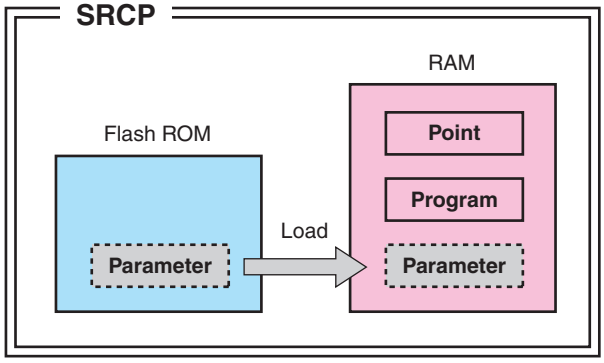
■ Saving the parameter data to the flash ROM



* Only parameter data can be saved to the flash ROM. Point data and program data cannot be saved.

The TPB or POPCOM is needed to save the parameter data to the flash ROM.

■ Loading the parameter data from the flash ROM to the RAM (Restoring the data)



There are two methods for loading the parameter data backed up in the flash ROM to the RAM.

(1)Manual load using the TPB or POPCOM (options)

Use the TPB or POPCOM to manually load the parameter data stored in the flash ROM into the RAM.

(2)Auto loading at SRCP power-on (Auto-load function)

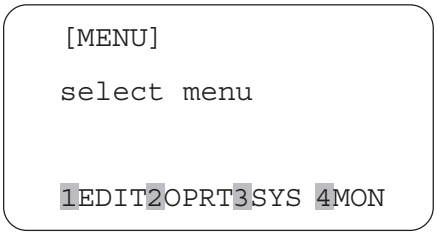
The auto-load function automatically loads the data backed up in the flash ROM onto the RAM, each time power to the SRCP is turned on. The auto-load function can be enabled or disabled when saving data onto the flash ROM.

CAUTION
When the auto-load function is enabled, changes you make to parameter data in the RAM are rewritten by the parameter data in the flash ROM if you turn the SRCP off without saving the changes into the flash ROM.

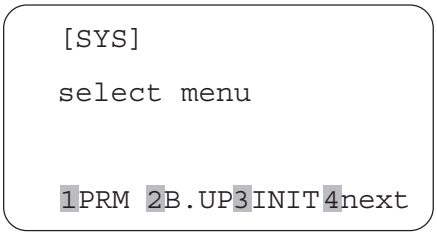
10-8-1 Saving the parameter data onto the flash ROM

NOTE
The internal flash ROM can be used when the SRCP controller version is 24.10 or later and the TPB version is 12.51 or later.

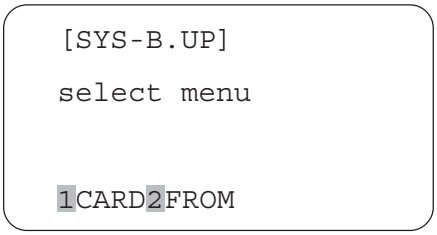
1) On the initial screen, press **F3** (SYS).



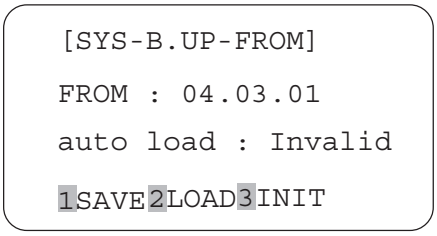
2) Next, press **F2** (B.UP).



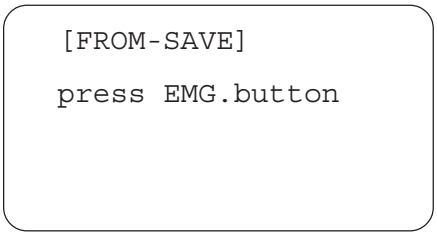
3) Press **F2** (FROM).



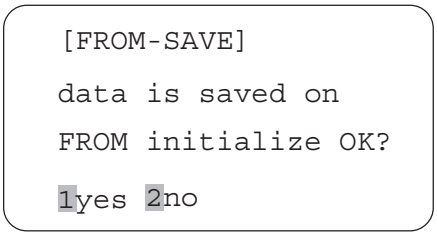
4) The ID number and auto load function setting in the flash ROM appear on the screen. Press **F1** (SAVE) here.



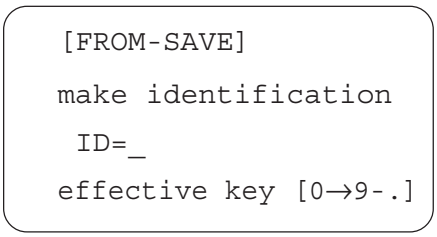
5) Press the emergency stop button on the TPB.



6) A confirmation message appears if data is stored in the flash ROM.
To initialize the flash ROM, press **F1** (yes).
To cancel, press **F2** (no).



7) Set an ID number for the data to be saved. Using the number keys, the "-" (minus sign) key and the "." (period) key, enter a number of up to 9 characters and then press ➡.



8) A confirmation message appears.
To save the parameter data, press F1 (yes). To cancel, press F2 (no).

```
[FROM-SAVE]
save OK ?
 ID=04.04.10
1yes 2no
```

9) This screen is displayed during saving of data.

```
[FROM-SAVE]
saving ...
```

10) After saving the data onto the flash ROM, the auto load function is set to "Invalid" (disabled).
If you want to change the auto load function, press F1 (ALOAD) to display the auto load function setup screen.

```
[FROM-SAVE]
saving is complete

1ALOD
```

11) The current status of the auto load function setting appears.
Leave this setting at "0" to disable the auto load function. To enable the auto load function, enter "1" with the number key and press ➡.

```
[FROM-SAVE-ALOD]
auto load = 0
0:Invalid 1:Valid
```

12) The screen returns to step 11 when the setting is complete.

```
[FROM-SAVE-ALOD]
auto load = 1
0:Invalid 1:Valid
```

⚠ ***CAUTION***
When saving the data onto the flash ROM, make sure that the I/O. CN connector is disconnected and the emergency stop button is pressed.
Do not move the robot or turn off the SRCP controller during saving of data.

10-8-2 Manually loading the data from flash ROM

NOTE
The internal flash ROM can be used when the SRCP controller version is 24.10 or later and the TPB version is 12.51 or later.

1) On the initial screen, press **F3** (SYS).

```
[MENU]
select menu

1EDIT2OPRT3SYS 4MON
```

2) Next, press **F2** (B.UP).

```
[SYS]
select menu

1PRM 2B.UP3INIT4next
```

3) Press **F2** (FROM).

```
[SYS-B.UP]
select menu

1CARD2FROM
```

4) The ID number and auto load function setting in the flash ROM appear on the screen. Press **F2** (LOAD) here.

```
[SYS-B.UP-FROM]
FROM : 04.03.01
auto load : Invalid
1SAVE2LOAD3INIT
```

5) Press the emergency stop button on the TPB.

```
[FROM-LOAD]
press EMG.button
```

6) A confirmation message appears asking whether to load the data from the flash ROM. To load the data, press **F1** (yes). To cancel, press **F2** (no).

```
[FROM-LOAD]
load OK ?

1yes 2no
```

7) This screen is displayed during loading of data.

```
[FROM-LOAD]
loading ...
```

8) The screen returns to step 4 when loading is complete.

```
[SYS-B.UP-FROM]
FROM : 04.03.01
auto load : Invalid
1SAVE2LOAD3INIT
```

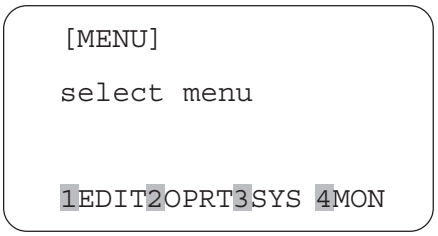
⚠ ***CAUTION***

When loading the data from the flash ROM, make sure that the I/O. CN connector is disconnected and the emergency stop button is pressed.
Do not move the robot or turn off the SRCP controller during loading of data.

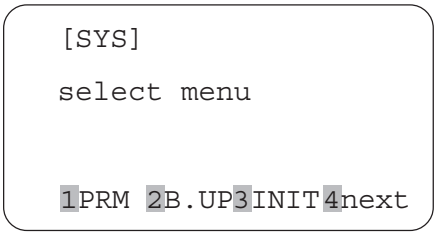
10-8-3 Initializing the flash ROM data

NOTE
The internal flash ROM can be used when the SRCP controller version is 24.10 or later and the TPB version is 12.51 or later.

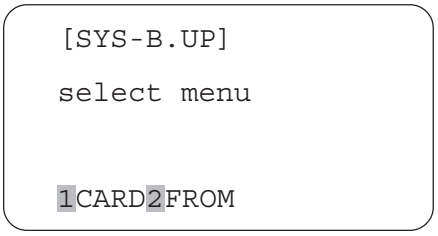
1) On the initial screen, press **F3** (SYS).



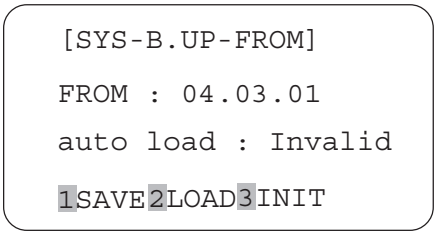
2) Next, press **F2** (B.UP).



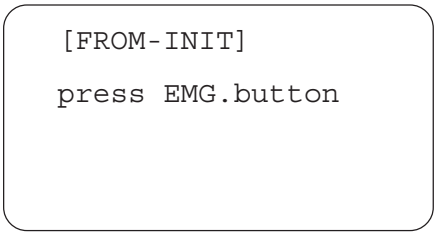
3) Press **F2** (FROM).



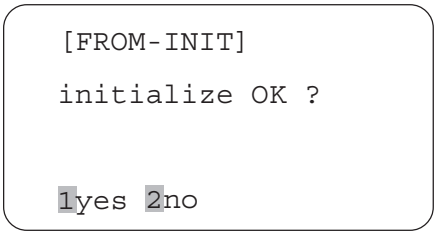
4) The ID number and auto load function setting in the flash ROM appear on the screen. Press **F3** (INIT) here.



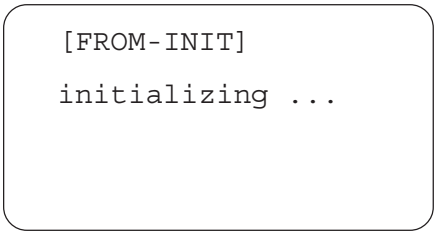
5) Following the message, press the emergency stop button on the TPB.



6) A confirmation message appears asking whether to initialize the flash ROM data. To initialize the flash ROM, press **F1** (yes). To cancel, press **F2** (no).



7) This screen is displayed during initialization.



8) The screen returns to step 4 when initialization is complete.
After initializing the flash ROM, the auto load function is set to "Invalid" (disabled).

```
[SYS-B.UP-FROM]
FROM :
auto load : Invalid
1SAVE 2LOAD 3INIT
```

⚠ ***CAUTION***

When initializing the flash ROM data, make sure that the I/O. CN connector is disconnected and the emergency stop button is pressed.
Do not move the robot or turn off the SRCP controller during initialization.

MEMO

Chapter 11 COMMUNICATION WITH PC

The SRCP controller allows you to edit the program data and point data or control the robot operation using a PC (personal computer) by RS-232C communication instead of using the TPB.
This chapter describes how to set the communication parameters required to communicate between the PC and the SRCP controller, and also explains the communication command specifications.

11-1 Communication Parameter Specifications

The communication parameters on the PC should be set as follows. For the setting procedure, refer to the computer operation manual.

■ Baud rate	9600 bps
■ Data bit length	8 bits
■ Stop bit length	1 bit
■ Parity check	On
■ Parity setting	Odd
■ Control method (X parameter)	XON/XOFF software control (Effective)
■ Communication method	Full duplex
■ Sync method	Asynchronous method
■ Return key transmission	CR code
■ CR code reception	For CR/LF reception : Return + line feed For CR reception : Return

* If the above parameter settings are not possible due to your equipment specifications, the robot controller settings can be changed by changing PRM47 (communication parameter settings) from the TPB.

PRM47 settings (default value: 0)

bit	Function	Selection			
15 to 9	Reserved	Always set to 0.			
8	Termination code	0: CR + LF		1: CR	
7 to 4	Transmission speed	0: 9600bps	1: 300bps	2: 600bps	3: 1200bps
		4: 2400bps	5: 4800bps	6: 9600bps	7 to 15: Cannot be set
3	Data bit length	0: 8 bits		1: 7 bits	
2	Stop bit length	0: 1 bit		1: 2 bits	
1 to 0	Parity check	0: Odd	1: Even	2 to 3: Non	

Example: To set the data bit length to "7 bits" and the parity check to "Non", enter "10" for PRM47, which is given by
0000000000001010 (binary) = 10 (decimal)

⚠ ***CAUTION***

Be sure to use a cable which conforms to specifications listed in "11-2 Communication Cable Specifications". The settings will be invalid if other cables such as POPCOM communication cables or those having different specifications are used.
After changing the parameters, turn the power off and then turn it on again to enable the settings.
The TPB can be used even if the parameters have been changed.

11-2 Communication Cable Specifications

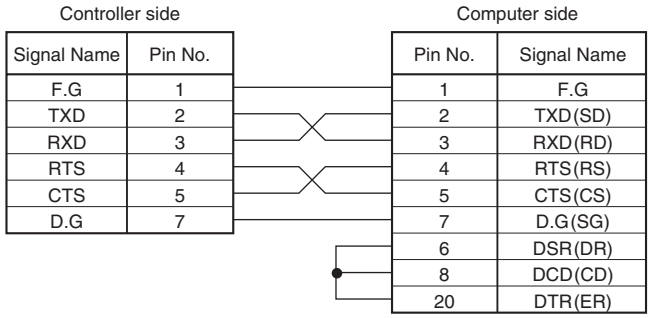
⚠ *CAUTION*

Pins 10, 12, 18 and 21 of the controller's connector are specifically used for TPB connection. To avoid possible accidents do not connect other inputs to these pins.
When using optional POPCOM software, make connections while referring to the POPCOM operation manual since it shows the different connection specifications.
The personal computer may have its own connector specifications, so be sure to check the computer operation manual to ensure the connections are correct.

11-2-1 Connecting to the computer with a 25-pin D-sub connector

Connector model

Mating connector type No. : XM2A-2501 (OMRON) or equivalent type
Mating connector cover type No. : XM2S-2511 (OMRON) or equivalent type



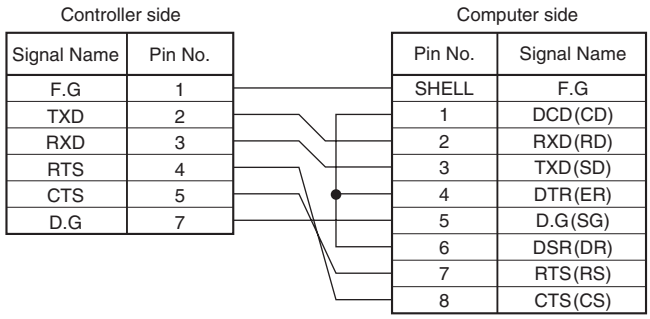
11-2-2 Connecting to the computer with a 9-pin D-sub connector

Connector model (controller side)

Mating connector type No. : XM2A-2501 (OMRON) or equivalent type
Mating connector cover type No. : XM2S-2511 (OMRON) or equivalent type

Connector model (computer side)

Mating connector type No. : XM2D-0901 (OMRON) or equivalent type
Mating connector cover type No. : XM2S-0913 (OMRON) or equivalent type



The "SHELL" is the metallic casing of the connector.

NOTE

Transmission stops while CTS on the controller side is off. If a robot alarm is issued while CTS is on, the controller keeps sending the message.
RTS on the controller side is always on.

11-3 Communication Command Specifications

On the SRCP controller, a command interface resembling the BASIC programming language is provided as standard, to facilitate easy communication with a PC. Communication commands are divided into the following four categories:

1. Robot movements
2. Data handling
3. Utilities
4. Special codes

Format: (except for special codes)

```
@<operation code> [<operand 1>][,<operand 2>][,<operand 3>]c/r l/f
```

- Basically, all of the commands begin with the start code '@' (=40H) and end with the code c/r (=0DH) l/f (=0AH). These two codes signal the controller that the statements between them constitute one command line. (The special codes are the only ones that do not require a start or an end code.)

- A communication command is basically composed of an operation code and an operand. Depending on the command statement, either no operand is used, or up to three operands are used.
  Items in [ ] (brackets) can be omitted.

- The character codes used in the SRCP series, are the JIS8 unit system codes (ASCII codes with katakana characters added). Input characters can be upper case or lower case.

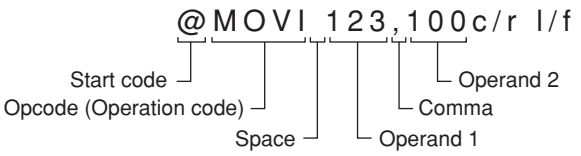
- One or more space must be inserted between the operation code and the operand.

- Items with the < > marks should be specified by the user. Check the description of each communication command and enter the appropriate data. (Refer to "11-5 Communication Command Description".)

- When two or more operands are entered, insert a comma (,) between them.

An example is shown below.

Transmission example



11-4 Communication Command List

1. Robot movement

No.	Operation code	Operand 1	Operand 2	Operand 3	Command details
1.	ORG ORGN				Returns to origin
2.	RESET				Resets program
3.	RUN				Starts automatic operation
4.	SRUN				Starts step operation
5.	SRVO	0 1			Turns servo off Turns servo on
6.	X+/X-				Performs jog movement (inching) along X-axis
7.	XINC/ XDEC				Performs jog movement along X-axis
8.	MOVD	X-axis position (mm)	speed		Directly moves to specified position
9.	MOVA	point number	speed		Moves to specified position
10.	MOVI	point number	speed		Moves specified movement amount
11.	MOVF	point number	DI number	0 or 1	Moves in response to general-purpose input
12.	DO	output number	0 1		Turns off general-purpose output or memory output Turns on general-purpose output or memory output
13.	WAIT	input number	0 or 1		Waits general-purpose input or memory input
14.	TIMR	time			Waits for specified time
15.	P	point number			Defines point variable P
16.	P+				Adds 1 to point variable P
17.	P-				Subtracts 1 from point variable P
18.	MOVM	pallet work position	speed		Moves to specified pallet work position
19.	MAT	number of rows	number of columns	pallet number	Defines matrix on specified pallet
20.	MSEL	pallet number			Specifies pallet number where to move
21.	CSEL	array element number			Specifies array element of counter array variable C
22.	C	counter value			Defines counter array variable C
23.	C+	[addition value]			Adds specified value to counter array variable C
24.	C-	[subtraction value]			Subtracts specified value from counter array variable C
25.	D	counter value			Defines counter variable D
26.	D+	[addition value]			Adds specified value to counter variable D
27.	D-	[subtraction value]			Subtracts specified value from counter variable D
28.	SHFT	point number			Performs point data shift

2. Data handling

No.	Operation code	Operand 1	Operand 2	Operand 3	Command details
1.	?POS				Reads current position
2.	?NO				Reads current program number
3.	?SNO				Reads current step number
4.	?TNO				Reads current task number
5.	?PNO				Reads current point number
6.	?STP	program number			Reads total number of steps in specified program
7.	?MEM				Reads number of steps that can be added
8.	?VER				Reads ROM version number
9.	?ROBOT				Reads robot number
10.	?CLOCK				Reads total operation time of controller
11.	?ALM	history number	[display count]		Reads alarm history
12.	?ERR	history number	[display count]		Reads error history
13.	?EMG				Confirms emergency stop status
14.	?SRVO				Confirms servo status
15.	?ORG				Confirms return-to-origin status
16.	?MODE				Confirms operation mode
17.	?PVA				Reads current point variable P
18.	?DI	input number			Reads general-purpose input or memory input status
19.	?DO	output number			Reads general-purpose output or memory output status
20.	?PRM	parameter number			Reads specified parameter data
		parameter number	parameter number		Reads specified multiple parameter data
21.	?P	point number			Reads specified point data
		point number	point number		Reads specified multiple point data
22.	READ	program number	step number	number of steps	Reads specified program data
		PGM			Reads all program data
		PNT			Reads all point data
		PRM			Reads all parameter data
		ALL			Batch reads all program, point and parameter data
		DIO			Reads input/output information
		MIO			Reads memory input/output information
		INF			Reads registered program information
23.	WRITE	PGM			Writes program data
		PNT			Writes point data
		PRM			Writes parameter data
		ALL			Batch writes program, point and parameter data
24.	?MAT	pallet number			Reads matrix definition contents
25.	?MSEL				Reads currently specified matrix number
26.	?CSEL				Reads currently specified element number of counter array variable C.
27.	?C	[array element number]			Reads current counter array variable C
28.	?D				Reads current counter variable D
29.	?SHFT				Reads current shift data

3. Utility

No.	Operation code	Operand 1	Operand 2	Operand 3	Command details
1.	INIT	PGM			Initializes program data
		PNT			Initializes point data
		PRM	robot number		Initializes robot parameters
		CLOCK			Initializes timer that measures total operation time
		ALM			Initializes alarm history
		ERR			Initializes error history
2.	SWI	program number			Switches program number to be run
3.	SWITSK	task number			Switches task number to be run
4.	SINS	program number	step number		Inserts one program step
5.	SDEL	program number	step number		Deletes one program step
6.	SMOD	program number	step number		Modifies one program step
7.	COPY	program number (copy source)	program number (copy destination)		Copies program
8.	DEL	program number			Deletes specified program
9.	PDEL	point number	number of points		Deletes point data

4. Special codes

No.	Code	Command details
1.	^C (=03H)	Interrupts RUN, SRUN, ORG, etc.
2.	^Z (=1AH)	Ends data transmission

11-5 Communication Command Description

11-5-1 Robot movements

(1) @ORG
@ORGN

Returns the robot to its origin position and outputs the machine reference value when completed correctly.

Transmission example : @ORG c/r l/f ..... Performs return-to-origin.

Response example 1 : OK c/r l/f
52% c/r l/f
OK c/r l/f

Response example 2 : NG c/r l/f ..... The robot is running.
31: running c/r l/f Execute the command again after stopping the robot.

NOTE

The magnetic pole is detected simultaneously with return-to-origin operation. Every time the power is turned on, return-to-origin becomes incomplete. Always perform return-to-origin after turning on the power to the controller, before starting operation. Return-to-origin also becomes incomplete after a parameter related to the origin position is changed. Return-to-origin must be reperformed in this case.

CAUTION

If the robot is operated while return-to-origin is still incomplete, the thrust necessary to move the robot cannot be obtained causing an alarm or abnormal operation. Always perform return-to-origin before starting the robot operation.

CAUTION

When performing return-to-origin by the stroke-end detection method, do not interrupt the return-to-origin operation while detecting the origin (while contacting the mechanical limit). Otherwise, the operation will stop due to a controller overload alarm and the power will need to be turned on again.

CAUTION

Avoid repeating return-to-origin operations. If you must repeat return-to-origin using the stroke-end detection method, wait at least 5 seconds before repeating it.

(2)@RESET

This returns the program execution step to the first step of the program selected with the '@SWI' statement, and turns all general-purpose outputs (DO0 to DO4) and memory output off. The "current position in the program" used as a reference for the relative movement command (MOVI) is initialized to the current position of the robot, and the point variable P is also cleared to 0.

Transmission example : @RESET c/r l/f

Response example 1 : OK c/r l/f

Response example 2 : NG c/r l/f ..... The robot is running.
31: running c/r l/f Execute the command again after stopping the robot.

* When PRM33 ("operation at return-to-origin complete" parameter) is set to 1 or 3, DO4 does not turn off even if the @RESET command is executed. Likewise, when PRM46 (servo status output parameter) is set to 1, DO3 does not turn off even if the @RESET command is executed.

(3)@RUN

This executes a program all the way to the last step.
In the case of a multi-task program, all tasks are executed.

Transmission example : @RUN c/r l/f

Response example 1 : STOP c/r l/f ..... The last step of the program has been executed.
60: program end c/r l/f

Response example 2 : NG c/r l/f ..... Return-to-origin has not been performed. Execute the command again after performing return-to-origin
32: origin incomplete c/r l/f

⚠ ***CAUTION***

When using an endless program (program that unconditionally returns to the head of the program at the last step), there will be no response.

(4)@SRUN

This executes only one step of a program.
In the case of a multi-task program, the selected task is executed.

Transmission example : @SRUN c/r l/f

Response example 1 : OK c/r l/f

Response example 2 : STOP c/r l/f ..... The last step of the program has been executed.
60: program end c/r l/f

Response example 3 : NG c/r l/f ..... Return-to-origin has not been performed. Execute the command again after performing return-to-origin.
32: origin incomplete c/r l/f

(5)@SRVO <servo status>

Turns the servo on or off.

Servo status : Specify 1 to turn the servo on or 0 to turn it off.

Transmission example : @SRVO 0 c/r l/f ..... Turns the servo off.

Response example : OK c/r l/f

(6)@X+, (@X-)

@X+ moves the robot to the + side and @X- to the - side based on the following equation.

Movement distance = 1 × (PRM26/100) (mm) PRM26: Teaching movement data (%)

(7)@XINC, (@XDEC)

@XINC moves the robot to the + side and @XDEC to the - side at a speed calculated by the equation below. The robot continues moving until the ^C code is input or the robot reaches the soft limit.

Movement speed = 100 × (PRM26/100) (mm/sec.) PRM26: Teaching movement data (%)

⚠ ***CAUTION***

The soft limit will not work unless return-to-origin has been performed.

(8)@MOVD <X-axis position (mm)>,<speed>

Moves the robot to a specified coordinate position.

X-axis position	:	Specify the target position to move the robot to.
Speed	:	The speed can be set to any level between 1 and 100. If PRM30 (Maximum program speed) is 100, then 100 will be equal to 2000 mm/sec (when PRM44=2000).
Transmission example	:	@MOVD 50.37,100 c/r l/f ..... Moves the robot to the position at 50.37 mm, at 100% speed.
Response example 1	:	OK c/r l/f
Response example 2	:	NG c/r l/f ..... The target position exceeds the soft limit. Change the target position or soft limit parameter. 30: soft limit over c/r l/f

(9)@MOVA <point number>,<speed>

Moves the robot to a position specified by a point number at a specified speed.

Point number	:	This is a number assigned to each point (position data) and can be from 0 to 999 (a total of 1,000 points). Data for the point numbers can be edited with the @WRITE PNT statement. The point variable "P" can also be used.
Speed	:	The speed can be set to any level between 1 and 100. If PRM30 (Maximum program speed) is 100, then 100 will be equal to 2000 mm/sec (when PRM44=2000).
Transmission example	:	@MOVA 123,100 c/r l/f ..... Moves the robot to point 123 at 100% speed.
Response example 1	:	OK c/r l/f
Response example 2	:	NG c/r l/f ..... The target position exceeds the 30: soft limit over c/r l/f soft limit. Change the point data or soft limit parameter.

(10)@MOVI <point number>,<speed>

Moves the robot a distance specified by a point number from the current position, at a specified speed.

Point number	:	This is a number assigned to each point (position data) and can be from 0 to 999 (a total of 1,000 points). Data for the point numbers can be edited with the @WRITE PNT statement. The point variable "P" can also be used.
Speed	:	The speed can be set to any level between 1 and 100. If PRM30 (Maximum program speed) is 100, then 100 will be equal to 2000 mm/sec (when PRM44=2000).
Transmission example	:	@MOVI 123,100 c/r l/f ..... Moves the robot a distance defined by point 123, at 100% speed.
Response example 1	:	OK c/r l/f
Response example 2	:	NG c/r l/f ..... The target position exceeds the 30: soft limit over c/r l/f soft limit. Change the point data or soft limit parameter.

⚠ ***CAUTION***

When movement is interrupted with a stop (^C) statement, the current position in the program stays unchanged so that the movement can be resumed by executing the @MOVI command again. However, if the command is reset, the current position in the program is initialized to the actual robot position.

(11) @MOVF <point number>,<DI number>,<DI status>

This command moves the robot toward a position specified by a point number until a specified DI input condition is met. When the DI condition is met, the robot stops and the command terminates. Even if the DI condition is not met, the command terminates when the target point is reached.

Point number	:	This is a number assigned to each point (position data) and can be from 0 to 999 (a total of 1,000 points). Data for the point numbers can be edited with the @WRITE PNT statement. The point variable "P" can also be used.
DI number	:	Specify one of the general-purpose inputs DI0 to DI7.
DI status	:	Specify 1 (ON) or 0 (OFF) as the input condition.
Transmission example	:	@MOVF 2,10,1 c/r l/f ..... Moves to point 2 until DI10 becomes 1 (ON).
Response example	:	OK c/r l/f

⚠ ***CAUTION***

The movement speed is set with PRM9 (MOVF speed) and independent of the PRM30 setting (Maximum program speed).

(12) @DO <general-purpose output or memory output number>,<output status>

Turns a general-purpose output or memory output on or off.

Output number	:	Specify one of the general-purpose outputs from 0 to 4 (5 points) or one of the memory outputs from 100 to 131 (32 points).
Output status	:	Specify 1 (ON) or 0 (OFF).
Transmission example	:	@DO 3,1 c/r l/f ..... Turns on general-purpose output 3.
Response example	:	OK c/r l/f

(13) @WAIT <general-purpose input or memory input number>,<input status>

Waits until a specified general-purpose input or memory input is switched to a specified status.

Input number	:	Specify one of the general-purpose inputs from 0 to 7 (8 points) or one of the memory inputs from 100 to 147 (48 points).
Input status	:	Specify 1 (ON) or 0 (OFF).
Transmission example	:	@WAIT 1,1 c/r l/f ..... Waits until DI1 becomes 1 (ON).
Response example	:	OK c/r l/f

(14) @TIMR <time>

Waits a specified amount of time.

Time	:	Set the time between 1 and 65535 in units of 10ms.
Transmission example	:	@TIMR 100 c/r l/f ..... Waits one second.
Response example	:	OK c/r l/f

(15)@P <point number>

Sets the point variable P.

Point number : This can be any value from 0 to 999.

Transmission example : @P 100 c/r l/f ..... Set the point variable P to 100.

Response example : OK c/r l/f

⚠ ***CAUTION***

The contents of the point variable P are held even when the SRCP is turned off. However, when the program is reset or when the program reset is applied for example by switching the execution program, the point variable P will be initialized to 0.

(16)@P+

Adds 1 to the point variable P.

Transmission example : @P+ c/r l/f ..... Increments the point variable P. (P ← P+1)

Response example : OK c/r l/f

(17)@P-

Subtracts 1 from the point variable P.

Transmission example : @P- c/r l/f ..... Decrements the point variable P. (P ← P-1)

Response example : OK c/r l/f

(18)@MOVM <pallet work position>,<speed>

Moves the robot to a specified pallet work position at a specified speed.

Pallet work position : The pallet work position is a number used to identify each point on a matrix, and can be from 1 to 65025 (=255 × 255). The counter array variable C or counter variable D can also be used.

Speed : The speed can be set to any level between 1 and 100. If PRM30 (Maximum program speed) is 100, then 100 will be equal to 2000 mm/sec (when PRM44=2000).

Transmission example : @MOVM 5,100 c/r l/f ..... When a 4 × 3 matrix is defined, the robot moves to the point at "row 2, column 2" at 100% speed.

Response example 1 : OK c/r l/f

Response example 2 : NG c/r l/f ..... Data error. The specified pallet work position is outside the matrix.
23: data error c/r l/f

⚠ ***CAUTION***
- ***The MOVM statement performs calculation on the assumption that the robot operates on the Cartesian coordinate system.***
- ***Because only a single-axis robot is controlled with the SRCP, the actual movement is linear even if a 2-dimensional matrix is defined.***

(19)@MAT <number of rows>,<number of columns>,<pallet number>

Defines a matrix.

Number of rows : Set the number of rows from 1 to 255.

Number of columns : Set the number of columns from 1 to 255.

Pallet number : The pallet number is a number used to identify each matrix (pallet) and can be from 0 to 31.

Transmission example : @MAT 5,2,1 c/r l/f ..... Defines a matrix of 5 × 2 on pallet number 1.

Response example : OK c/r l/f

⚠ ***CAUTION***
Because only a single-axis robot is controlled with the SRCP, the actual movement is linear even if a 2-dimensional matrix is defined.

(20)@MSEL <pallet number>

Specifies a matrix where the robot moves with a MOVM statement.

Pallet number : The pallet number is a number used to identify each matrix (pallet) and can be from 0 to 31.

Transmission example : @MSEL 0 c/r l/f ..... Specifies pallet number 0.

Response example : OK c/r l/f

(21)@CSEL <array element number>

Specifies an array element for the counter array variable C to be used.

Array element number : This is a number used to designate an array element for the counter array variable C, and can be from 0 to 31. The counter variable D can also be specified here as the array element.

Transmission example : @CSEL 1 c/r l/f ..... Uses the counter array variable C of element number 1 in the subsequent steps.

Response example : OK c/r l/f

(22)@C <counter value>

Sets a specified value in the counter array variable C specified with the CSEL statement.

Counter value : This can be any value from 0 to 65535.

Transmission example : @C 100 c/r l/f ..... Sets the counter array variable C to 100.

Response example : OK c/r l/f

(23)@C+ [<addition value>]

Adds a specified value to the counter array variable C.

Addition value : This can be any value from 1 to 65535. If this value is omitted, then 1 is added to the counter array variable.

Transmission example : @C+ c/r l/f ..... Increments the counter array variable C. (C ← C+1)

Response example : OK c/r l/f

(24)@C- [<subtraction value>]

Subtracts a specified value from the counter array variable C.

Subtraction value : This can be any value from 1 to 65535. If this value is omitted, then 1 is subtracted from the counter array variable.

Transmission example : @C- c/r l/f ..... Decrements the counter array variable C. (C ← C-1)

Response example : OK c/r l/f

(25)@D <counter value>

Sets a specified value in the counter variable D.

Counter value : This can be any value from 0 to 65535.

Transmission example : @D 100 c/r l/f ..... Sets the counter variable D to 100.

Response example : OK c/r l/f

(26)@D+ [<addition value>]

Adds a specified value to the counter variable D.

Addition value : This can be any value from 1 to 65535. If this value is omitted, then 1 is added to the counter variable.

Transmission example : @D+ c/r l/f ..... Increments the counter variable D. (D ← D+1)

Response example : OK c/r l/f

(27)@D- [<subtraction value>]

Subtracts a specified value from the counter variable D.

Subtraction value : This can be any value from 1 to 65535. If this value is omitted, then 1 is subtracted from the counter variable.

Transmission example : @D- c/r l/f ..... Decrements the counter variable D. (D ← D-1)

Response example : OK c/r l/f

(28)@SHFT <point number>

Shifts the position data by an amount equal to the distance defined by a specified point number. The shifted data is valid until the SHFT statement is executed again or until the program is reset.

Point number : This is a number used to identify each point (position data) and can be from 0 to 999 (a total of 1,000 points). Data for the point numbers can be edited with the @WRITE PNT statement. The point variable P can also be used.

Transmission example : @SHFT 1 c/r l/f..... Shifts the point data by an amount defined by point number 1 and the shifted data is used with the subsequent movement commands.

Response example : OK c/r l/f

⚠ ***CAUTION***

When the program is reset, the shift data will be initialized to 0.00.
The SHFT statement affects MOVA, MOVF and MOVM, but does not affect MOVD and MOVI.

11-5-2 Data handling

(1)@?POS

Reads the current position.

Transmission example : @?POS c/r l/f

Response example : 321.05 c/r l/f
OK c/r l/f

(2)@?NO

Reads the current program number. In multi-task operation, this command reads the program information on the task currently selected.

Transmission example : @?NO c/r l/f

Response example 1 : 31 c/r l/f ..... Program No.31 is being executed.
OK c/r l/f

Response example 2 : 10/l c/r l/f ..... Program No.1 is the lead program (program selected with @SWI statement), and program No.10 is currently being executed with the JMP or CALL statement, etc.
OK c/r l/f

(3)@?SNO

Reads the current step number. The @RUN and @SRUN commands are executed from the step read here. In multi-task operation, this command reads the program information on the task currently selected.

Transmission example : @?SNO c/r l/f

Response example : 170 c/r l/f
OK c/r l/f

(4)@?TNO

Reads the current task number.

Transmission example : @?TNO c/r l/f

Response example : 0 c/r l/f ..... Task 0 (main task) is currently selected.
OK c/r l/f

(5)@?PNO

Reads the currently selected point number. This is used to find which point data is being used for movement, or to find the point that caused an error if it occurs. In multi-task operation, this command reads the program information on the task currently selected.

Transmission example : @? PNO c/r l/f

Response example : 57 c/r l/f
OK c/r l/f

(6)@?STP <program number>

Reads the total number of steps in the specified program.

Program number : This is a number used to identify each program and can be 0 to 99 (a total of 100).

Transmission example : @?STP 10 c/r l/f ..... Reads the total number of steps for program No. 10.

Response example : 140 c/r l/f
OK c/r l/f

(7)@?MEM

Reads the number of steps that can be added.

Transmission example : @?MEM c/r l/f

Response example : 1001 c/r l/f
OK c/r l/f

⚠ *CAUTION*

In addition to the number of existing steps, the steps equivalent to the number of programs are used internally as the program control steps. For example, if one program with 50 steps is registered, the number of the available remaining steps will be 2949 steps (3000 - 1- 50 = 2949).

(8)@?VER

Reads the ROM version in the SRCP controller.

Transmission example : @?VER c/r l/f

Response example : 24.00 c/r l/f
OK c/r l/f

(9)@?ROBOT

Reads the type of the robot currently specified.

Transmission example : @?ROBOT c/r l/f

Response example : 516 c/r l/f
OK c/r l/f

(10)@?CLOCK

Reads the total operation time of the SRCP controller.

Transmission example : @?CLOCK c/r l/f

Response example : 00101,05:11:12 c/r l/f ..... Indicates that the total operation time is 101 days, 5 hours, 11 minutes and 12 seconds.
OK c/r l/f

(11) @?ALM <history number>[,<display count>]

Displays a specified number of past alarms, starting from a specified history number.
A maximum of 100 past alarms can be displayed.
This alarm history shows the time (total elapsed time from controller start-up) that each alarm occurred and a description of the alarm.

History number : This number is assigned to each alarm sequentially from 0 to 99 in the order the alarms occurred. History number 0 indicates the most recent alarm that occurred. A larger history number indicates it is an older alarm.

Display count : Specify the number of alarms you want to display from 1 to 100. If this entry is omitted, only one alarm is displayed.

Transmission example : @?ALM 0,2 c/r l/f ..... Displays the two most recent alarms that occurred.

Response example : 00101,05:11:12,X04: POWER DOWN c/r l/f
00096,18:10:02,X04: POWER DOWN c/r l/f
OK c/r l/f ..... The most recent alarm that occurred was a voltage drop alarm occurring 101 days, 5 hours, 11 minutes and 12 seconds after the SRCP controller has started. The next most recent alarm was a voltage drop alarm occurring 96 days, 18 hours, 10 minutes and 2 seconds after the SRCP controller has started.

(12) @?ERR <history number>[,<display count>]

Displays a specified number of past errors, starting from a specified history number.
A maximum of 100 past errors can be displayed.
This error history shows the time (total elapsed time from controller start-up) that each error occurred and a description of the error.

History number : This number is assigned to each error sequentially from 0 to 99 in the order the errors occurred. History number 0 indicates the most recent error that occurred. A larger history number indicates it is an older error.

Display count : Specify the number of errors you want to display from 1 to 100. If this entry is omitted, only one error is displayed.

Transmission example : @?ERR 0,2 c/r l/f ..... Displays the two most recent errors that occurred.

Response example : 00:00101,05:11:12,PIO,52 : NO POINT DATA c/r l/f
01:00096,18:10:02,CMU,30: SOFT LIMIT OVER c/r l/f
OK c/r l/f ..... The most recent error that occurred was a "no point data" error in a parallel I/O command occurring 101 days, 5 hours, 11 minutes and 12 seconds after the SRCP controller has started. The next most recent error was a "soft limit over" error during TPB or RS-232C operation occurring 96 days, 18 hours, 10 minutes and 2 seconds after the SRCP controller has started.

(13)@?EMG

Reads the emergency stop status.

Transmission example : @?EMG c/r l/f

Response example 1 : 0 c/r l/f ..... Emergency stop is off.
OK c/r l/f

Response example 2 : 1 c/r l/f ..... Emergency stop is on.
OK c/r l/f

(14)@?SRVO

Reads the servo status.

Transmission example : @?SRVO c/r l/f

Response example 1 : 0 c/r l/f ..... Servo is off.
OK c/r l/f

Response example 2 : 1 c/r l/f ..... Servo is on.
OK c/r l/f

(15)@?ORG

Reads whether or not return-to-origin has been completed.

Transmission example : @?ORG c/r l/f

Response example 1 : 0 c/r l/f ..... Return-to-origin not completed.
OK c/r l/f

Response example 2 : 1 c/r l/f ..... Return-to-origin completed.
OK c/r l/f

(16)@?MODE

Reads the robot status.

Transmission example : @?MODE c/r l/f

Response example 1 : 0 c/r l/f ..... Robot is stopped.
OK c/r l/f

Response example 2 : 1 c/r l/f ..... Program is being executed from TPB or PC.
OK c/r l/f

Response example 3 : 2 c/r l/f ..... Program is being executed by I/O command.
OK c/r l/f

(17)@?PVA

Reads the point variable P. In multi-task operation, this command reads the program information on the task currently selected.

Transmission example : @?PVA c/r l/f

Response example : 0 c/r l/f
OK c/r l/f

⚠ *CAUTION*

The contents of the point variable P are held even when the SRCP is turned off. However, when the program is reset or when the program reset is applied for example by switching the execution program, the point variable P will be initialized to 0.

(18)@?DI <general-purpose input or memory input number>

Reads the status of a general-purpose input or memory input.

Input number : Specify one of the general-purpose inputs 0 to 7 (8 points) or one of the memory inputs 100 to 147 (48 points).

Transmission example : @?DI 1 c/r l/f

Response example 1 : 0 c/r l/f ..... Input status is off.
OK c/r l/f

Response example 2 : 1 c/r l/f ..... Input status is on.
OK c/r l/f

(19)@?DO <general-purpose output or memory output number>

Reads the status of a general-purpose output or memory output.

Output number : Specify one of the general-purpose outputs 0 to 4 (5 points) or one of the memory outputs 100 to 131 (32 points).

Transmission example : @?DO 2 c/r l/f

Response example 1 : 0 c/r l/f ..... Output status is off.
OK c/r l/f

Response example 2 : 1 c/r l/f ..... Output status is on.
OK c/r l/f

(20-1) @?PRM <parameter number>

Reads the data from a specified parameter.

Parameter number : This is a number used to identify each parameter and can be from 0 to 99.

Transmission example : @?PRMl c/r l/f ..... Reads the data from PRM1 (parameter 1).

Response example 1 : 350 c/r l/f
OK c/r l/f

Response example 2 : c/r l/f ..... No data is registered in PRM1 (parameter 1).
OK c/r l/f

(20-2) @?PRM <parameter number>,<parameter number>

Reads multiple parameter data from the first parameter number to the second parameter number. If unregistered parameters exist, they will be skipped.

Parameter number : This is a number used to identify each parameter and can be from 0 to 99.

Transmission example : @?PRMl,5 c/r l/f ..... Reads the data from PRM1 to PRM5 (parameters 1 to 5).

Response example : PRM1=350 c/r l/f
PRM2=0 c/r l/f
PRM3=3 c/r l/f
PRM4=100 c/r l/f
PRM5=0 c/r l/f
OK c/r l/f

(21-1) @?P <point number>

Reads the data of a specified point.

Point number : This is a number used to identify each point data and can be from 0 to 999.

Transmission example : @?P 254 c/r l/f ..... Reads the data of point 254.

Response example 1 : -0.05 c/r l/f
OK c/r l/f

Response example 2 : c/r l/f ..... No data is registered in the specified point.
OK c/r l/f

(21-2) @?P <point number>,<point number>

Reads multiple point data from the first point number to the second point number. If unregistered points exist, they will be skipped.

Point number : This is a number used to identify each point data and can be from 0 to 999.

Transmission example : @?P15,22 c/r l/f ..... Reads the data from points 15 to 22.

Response example :
P15=100.00 c/r l/f
P16=32.11 c/r l/f
P20=220.00 c/r l/f
P22=0.50 c/r l/f
OK c/r l/f

(22-1) @READ <program number>,<step number>,<number of steps>

Reads a specified number of step data from a specified step in a program. If the number of steps from the specified step to the final step is less than the number of steps specified here, the command execution will end when the final step is read out.

Program number : This is a number used to identify each program and can be from 0 to 99.

Step number : This is a number assigned to each step and can be from 1 to 255.

Number of steps : Any number between 1 and 255 can be specified.

Transmission example : @READ 3,50,1 c/r l/f ..... Reads one step of data from step 50 in program No. 3.

Response example 1 :
MOVA 29,100 c/r l/f
^Z(=1AH)
OK c/r l/f

Response example 2 : NG c/r l/f ..... The specified step number is not registered
42: cannot find step c/r l/f

(22-2) @READ PGM

Reads all of the program data.

Transmission example : @READ PGM c/r l/f

Response example :
NO0 c/r l/f
MOVA 0,100 c/r l/f
JMPF 0,31,13 c/r l/f
NO31 c/r l/f
STOP c/r l/f
^Z (=1AH)
OK c/r l/f

(22-3) @READ PNT

Reads all point data.

Transmission example : @READ PNT c/r l/f

Response example :
```
P0=0.00 c/r l/f
P1=350.00 c/r l/f
P2=196.47 c/r l/f
P254=-0.27 c/r l/f
^Z (=1AH)
OK c/r l/f
```

(22-4) @READ PRM

Reads all parameter data.

Transmission example : @READ PRM c/r l/f

Response example :
```
PRM0=516 c/r l/f
PRM1=350 c/r l/f
   :
   :
PRM40=2 c/r l/f
^Z (=1AH)
OK c/r l/f
```

(22-5) @READ ALL

Reads all data (parameters, programs, points) at one time. Each data group (parameters, programs, points) is separated by an empty line (a carriage return only).

Transmission example : @READ ALL c/r l/f

Response example :
```
PRM0=516 c/r l/f
PRM1=350 c/r l/f
   :
   :
PRM40=2 c/r l/f
c/r l/f
NO0 c/r l/f
MOVA 0,100 c/r l/f
MOVA 1,100 c/r l/f
NO10 c/r l/f
CALL 0,10 c/r l/f
STOP c/r l/f
c/r l/f
P0=0.00 c/r l/f
P1=550.00 c/r l/f
^Z (=1AH)
OK c/r l/f
```

(22-6) @READ DIO

Reads the on/off status of DIO. Refer to "4-3-4 DIO monitor screen".

Transmission example : @READ DIO c/r l/f

Response example :
```
D I  00000000  00000000  c/r l/f
DO  11100000  O:0  S:1    c/r l/f
OK c/r l/f
```

(22-7) @READ MIO

Reads the on/off status of memory I/O. From the left, the top line shows MIO numbers from 115 to 100, the middle line from 131 to 116, and the bottom line from 147 to 132.

Transmission example : @READ MIO c/r l/f

Response example :
```
M   00000000  00000000  c/r l/f
    00000000  00000000  c/r l/f
    00000000  00000001  c/r l/f
OK c/r l/f
```

(22-8) @READ INF

Reads the status of the registered programs. The registered program numbers and number of steps are displayed.

Transmission example : @READ INF c/r l/f

Response example :
```
NO0- 43 steps c/r l/f
NO1- 52 steps c/r l/f
NO31- 21 steps c/r l/f
^Z (=1AH)
OK c/r l/f
```

(23-1) @WRITE PGM

Writes the program data. The controller will transmit READY when this command is received. Confirm that READY is received and then transmit the program data. Always transmit ^Z (=1AH) at the end of the data.

Transmission example :

Send	Receive
@WRITE PGM c/r l/f	
	READY c/r l/f
NO0 c/r l/f	
MOVA 0,100 c/r l/f	
JMPF 0,31,12 c/r l/f	
NO31 c/r l/f	
STOP c/r l/f	
^Z(=1AH)	
	OK c/r l/f

⚠ ***CAUTION***

When @WRITE PGM is executed, the previous data of the same program number is overwritten. (The previous data remains as long as its program number differs from the program number to be written.)

(23-2) @WRITE PNT

Writes the point data. The controller will transmit READY when this command is received. Confirm that READY is received and then transmit the point data. Always transmit ^Z (=1AH) at the end of the data.

Transmission example :	Send	Receive
	@WRITE PNT c/r l/f	
		READY c/r l/f
	P0=0.00 c/r l/f	
	P1=350.00 c/r l/f	
	P254=-0.27 c/r l/f	
	^Z(=1AH)	
		OK c/r l/f

⚠ ***CAUTION***

When @WRITE PNT is executed, the previous data of the same point number is overwritten. (The previous data remains as long as its point number differs from the point number to be written.)

(23-3) @WRITE PRM

Writes the parameter data. The controller will transmit READY when this command is received. Confirm that READY is received and then transmit the parameter data. Always transmit ^Z (=1AH) at the end of the data.

Transmission example :	Send	Receive
	@WRITE PRM c/r l/f	
		READY c/r l/f
	PRM1=550 c/r l/f	
	PRM2=10 c/r l/f	
	^Z(=1AH)	
		OK c/r l/f

⚠ ***CAUTION***

Loading unsuitable robot data to the SRCP can inhibit the robot controller performance, possibly resulting in failures, malfunctions, and errors.

(23-4) @WRITE ALL

Writes all data (parameters, programs and points) at one time. The controller will transmit READY when this command is received. Confirm that READY is received and then transmit all data. Always transmit ^Z (=1AH) at the end of the data.

		Send	Receive
Transmission example	:	@WRITE ALL c/r l/f	
			READY c/r l/f
		PRM0=516 c/r l/f	
		PRM1=350 c/r l/f	
		c/r l/f	
		NO10 c/r l/f	
		CALL 0, 20 c/r l/f	
		STOP c/r l/f	
		c/r l/f	
		P1=550.00 c/r l/f	
		^Z(=1AH)	
			OK c/r l/f

⚠ ***CAUTION***

- ***Always place one or more empty line to separate between each data group (parameters, programs, points).***
- ***There is no specific rule in the data group sequence. There can be data groups that are not written in.***
- ***When @WRITE ALL is executed, the previous data of the same program number or point number is overwritten. (The previous data remains as long as its program number or point number differs from the program number or point number to be written.)***
- ***Loading unsuitable robot data to the SRCP can inhibit the robot controller performance, possibly resulting in failures, malfunctions, and errors.***

(24) @?MAT <pallet number>

Reads the matrix data on a specified pallet.

Pallet number	:	This is a number used to identify each matrix (pallet) and can be from 0 to 31.
Transmission example	:	@?MAT 1 c/r l/f ..... Reads the matrix data on pallet number 1.
Response example	:	20,30 c/r l/f OK c/r l/f

(25) @?MSEL

Reads the pallet number for the currently specified matrix. In multi-task operation, this command reads the program information on the task currently selected.

Transmission example	:	@?MSEL c/r l/f
Response example	:	0 c/r l/f OK c/r l/f

(26)@?CSEL

Reads the currently specified element number of the counter array variable C. In multi-task operation, this command reads the program information on the task currently selected.

Transmission example : @?CSEL c/r l/f

Response example : 0 c/r l/f
OK c/r l/f

(27)@?C [<array element number>]

Reads the value in the counter array variable C of the specified element number.

Element number : This is a number used to specify each array element and can be from 0 to 31. If this entry is omitted, the element number selected with the @CSEL command is used.

Transmission example : @?C c/r l/f
OK c/r l/f

Response example : 21202 c/r l/f
OK c/r l/f

(28)@?D

Reads the counter variable D.

Transmission example : @?D c/r l/f

Response example : 21202 c/r l/f
OK c/r l/f

(29)@?SHFT

Reads the shift data currently set. In multi-task operation, this command reads the program information on the task currently selected.

Transmission example : @?SHFT c/r l/f

Response example : 150.00 c/r l/f
OK c/r l/f

11-5-3 Utilities

(1-1) @INIT PGM

Initializes all program data.

Transmission example : @INIT PGM c/r l/f

Response example : OK c/r l/f

(1-2) @INIT PNT

Initializes all point data.

Transmission example : @INIT PNT c/r l/f

Response example : OK c/r l/f

(1-3) @INIT PRM <robot number>

Initializes the parameter data to match the specified robot.
For robot numbers, refer to "15-1-2 Robot number list".

Transmission example : @INIT PRM 516 c/r l/f ..... Parameter data is initialized to match the model MR16T robot.

Response example : OK c/r l/f

(1-4) @INIT CLOCK

Initializes the timer to 0, which is used to measure the total operation time of the SRCP controller. The alarm history and error history are also initialized at this point.

Transmission example : @INIT CLOCK c/r l/f

Response example : OK c/r l/f

(1-5) @INIT ALM

Initializes the alarm history.

Transmission example : @INIT ALM c/r l/f

Response example : OK c/r l/f

(1-6) @INIT ERR

Initializes the error history.

Transmission example : @INIT ERR c/r l/f

Response example : OK c/r l/f

(2)@SWI <program number>

This command switches the execution program number. When a program is reset, program execution will always return to the first step of the program selected here. The program is reset when the @SWI command is executed.

Program number : This is a number used to identify each program and can be from 0 to 99.

Transmission example : @SWI 31 c/r l/f

Response example : OK c/r l/f

(3) @SWITSK <task number>

Switches the task number to be executed. In the subsequent step run, the program of the task selected here is executed. When the command such as @?NO or @?SNO is issued, the contents of this task replies to it.

Task number : This is a number used to identify each task and can be from 0 to 3.

Transmission example : @SWITSK 1 c/r l/f

Response example 1 : OK c/r l/f

Response example 2 : NG c/r l/f ..... The specified task was not found.
72: not execute task c/r l/f

(4)@SINS <program number>,<step number>

Inserts data in a specified step of a specified program. All data below the inserted data will shift down one line. If the step following the last step is specified, a new step will be added. If the first step of a program that does not exist is specified, a new program will be created. The SRCP controller will transmit READY when this command is received. Confirm that READY is received and then transmit the insertion data.

Program number : This is a number used to identify each program and can be from 0 to 99.

Step number : This is a number used to identify each step and can be from 1 to 255.

Transmission example 1 :

Send	Receive
@SINS 19,4 c/r l/f	
	READY c/r l/f
TIMR 50 c/r l/f	
	OK c/r l/f

Transmission example 2 :

Send	Receive
@SINS 19,4 c/r l/f	
	NG c/r l/f 43: cannot find PGM c/r l/f

(5)@SDEL <program number>,<step number>

Deletes a specified step.

Program number : This is a number used to identify each program and can be from 0 to 99.

Step number : This is a number used to identify each step and can be from 1 to 255.

Transmission example : @SDEL 31,99 c/r l/f ..... Deletes step 99 of program No. 31.

Response example 1 : OK c/r l/f

Response example 2 : NG c/r l/f ..... The specified step number is not registered.
42: cannot find step c/r l/f

(6)@SMOD <program number>,<step number>

Modifies data in a specified step. The SRCP controller will transmit READY when this command is received. Confirm that READY is received and then transmit the modification data.

Program number : This is a number used to identify each program and can be from 0 to 99.

Step number : This is a number used to identify each step and can be from 1 to 255.

Transmission example 1 :

Send	Receive
@SMOD 0,5 c/r l/f	
	READY c/r l/f
TIMR 50 c/r l/f	
	OK c/r l/f

Transmission example 2 :

Send	Receive
@SMOD 0,5 c/r l/f	
	NG c/r l/f 43: cannot find PGM c/r l/f

(7)@COPY <program number (copy source)>,<program number (copy destination)>

Copies a program. If a program exists in the copy destination, the program will be rewritten.

Program number : This is a number used to identify each program and can be from 0 to 99.

Transmission example : @COPY 0,1 c/r l/f ..... Copies program No. 0 to program No. 1.

Response example 1 : OK c/r l/f

Response example 2 : NG c/r l/f ..... The program to be copied is not registered.
43: cannot find PGM c/r l/f

(8)@DEL <program number>

Deletes a program.

Program number	:	This is a number used to identify each program and can be from 0 to 99.
Transmission example	:	@DEL 10 c/r l/f ..... Deletes program No. 10.
Response example 1	:	OK c/r l/f
Response example 2	:	NG c/r l/f ..... The program to be deleted is not registered. 43: cannot find PGM c/r l/f

(9)@PDEL <point number>,<number of points>

Deletes point data. Deletes the specified number of points starting with the point number specified here.

Point number	:	This is a number assigned to each point and can be from 0 to 999.
Number of points	:	Any number between 1 and 999 can be specified.
Transmission example	:	@PDEL 16,10 c/r l/f ..... Deletes 10 points starting from point 16 (up to point 25).
Response example	:	OK c/r l/f

Chapter 12 MESSAGE TABLES

This section lists all of the messages that are displayed on the TPB or sent to the PC (personal computer) to inform the operator of an error in operation or a current status. For a list of the alarm messages displayed if any trouble occurs, refer to "13-2 Alarm and Countermeasures".

12-1 Error Messages

12-1-1 Error message specifications

The error message transmission format is as follows.

```
<Error No.> : <Error message> c/r l/f
```

The length of the <error message> character string is 17 characters. (Spaces are added until the message contains 17 characters.) Thus, the character string length containing the c/r l/f will be 22 characters.

12-1-2 Command error message

Error No. 20	Message	no start code
	Cause	The start code (@) has not been added at the beginning of the command.
	Action	Always make sure the command begins with a start code (@).
Error No. 21	Message	illegal type
	Cause	The command is erroneous.
	Action	Use the correct command.
Error No. 22	Message	line buf overflow
	Cause	The number of characters in one line exceeds 80.
	Action	Limit the number of characters per line to 80 or less.
Error No. 23	Message	data error
	Cause	There is an error in numeric data.
	Action	Correct the data.
Error No. 24	Message	cannot access
	Cause	Execution is limited by the password or access level (operation level).
	Action	Cancel the limit.

12-1-3 Operation error message

Error No. 30	Message	soft limit over
	Cause	Executing the command will move the robot to a position that exceeds the soft limit set by parameter.
	Action	Review the point data or soft limit parameter.
Error No. 31	Message	running
	Cause	Another command is already being executed, so the command cannot be accepted.
	Action	Wait until execution of the current command finishes before inputting another command.
Error No. 32	Message	origin incomplete
	Cause	The command cannot be executed because a return to origin has not yet been completed.
	Action	Complete a return to origin first.
Error No. 33	Message	emergency stop
	Cause	The command cannot be executed because an emergency stop is triggered.
	Action	Cancel the emergency stop.
Error No. 34	Message	servo off
	Cause	The command cannot be executed because the servo is off.
	Action	Turn servo on.
Error No. 35	Message	system error 2
	Cause	An error interruption occurred due to noise or an unknown cause, so the status changed to servo off.
	Action	Turn servo on.
Error No. 36	Message	cannot restart
	Cause	Restart of the interpolation operation program was attempted.
	Action	Reset the program.
Error No. 37	Message	SVCE-port changed
	Cause	Execution was forcibly terminated because the SERVICE mode input state was changed.
	Action	Restart execution.
Error No. 38	Message	net link error
	Cause	The connection was forcibly disconnected because an error occurred in the network connection.
	Action	Remedy the network connection error, and then restart.

12-1-4 Program error message

Error No. 40	Message	stack overflow
	Cause	① Seven or more successive CALL statements were used within a CALL statement. ② In the program called as a subroutine by a CALL statement, a jump was made to another program by a JMP or JMPF statement.
	Action	① Reduce the number of CALL statements used in a CALL statement to 6 or less. ② Review the program.
Error No. 41	Message	cannot find label
	Cause	The specified label cannot be found.
	Action	Create the required label.
Error No. 42	Message	cannot find step
	Cause	The specified step cannot be found.
	Action	Check whether the step number is correct.
Error No. 43	Message	cannot find PGM
	Cause	The specified program cannot be found.
	Action	Check whether the program number is correct.
Error No. 44	Message	PGM memory full
	Cause	The total number of steps in all programs has exceeded 3000.
	Action	Delete unnecessary programs or steps.
Error No. 45	Message	step over
	Cause	The total number of steps in one program has exceeded 255.
	Action	Delete unnecessary steps or divide the program into two parts.

12-1-5 System error message

Error No. 50	Message	system error
	Cause	An unexpected error occurred.
	Action	Contact YAMAHA and describe the problem.
Error No. 51	Message	illegal opecode
	Cause	There is an error in a registered program.
	Action	Check the program.
Error No. 52	Message	no point data
	Cause	No data has been registered for the specified point number.
	Action	Register the point data.
Error No. 53	Message	PRM0 data error
	Cause	This error will not occur in the SRCP controller.
	Action	
Error No. 54	Message	PRM8 data error
	Cause	The number of conditional input points is set to something other than 1 to 8.
	Action	Correct the setting for the PRM8 parameter.
Error No. 56	Message	FROM write error
	Cause	An attempt was made to save data while the flash ROM contains data.
	Action	Initialize the flash ROM.
Error No. 57	Message	no FROM data
	Cause	An attempt was made to load data while the flash ROM contains no data.
	Action	Save data in the flash ROM.
Error No. 59	Message	robot type error
	Cause	Unsuitable parameter data was transmitted to the controller.
	Action	Initialize the parameters. Transmit the correct parameter data.

12-1-6 Multi-task error message

Error No. 70	Message	task running
	Cause	An attempt was made to start the task which is already in progress.
	Action	Check the program.
Error No. 71	Message	can't select task
	Cause	An attempt was made by a task to finish itself. An attempt was made to switch a task which is suspended.
	Action	Check the program. Check the task status.
Error No. 72	Message	not execute task
	Cause	An attempt was made to switch a task which has not started.
	Action	Check the task status.

12-2 TPB Error Messages

	Message	SIO error
	Cause	1. Parity error in data received from controller. 2. TPB was connected when dedicated command input was on.
	Action	1. Contact YAMAHA for consultation. 2. Turn all dedicated command inputs off before connecting the TPB.
	Message	bad format
	Cause	The memory card is not formatted.
	Action	Format the memory card.
	Message	save error
	Cause	Error in writing to the memory card.
	Action	Replace the memory card.
	Message	load error
	Cause	The memory card data is damaged.
	Action	Format or replace the memory card.
	Message	checksum error
	Cause	The memory card data is damaged.
	Action	Format or replace the memory card.
	Message	battery error
	Cause	The memory card battery voltage dropped.
	Action	Replace the memory card battery.
	Message	printer busy!!
	Cause	The printer is not ready.
	Remedy	Set the printer to print-ready state.

12-3 Stop Messages

12-3-1 Message specifications

The stop message transmission format is as follows.

```
<Message No.> : <Stop message> c/r l/f
```

The length of the <stop message> character string is 17 characters. (Spaces are added until the message contains 17 characters.) Thus, the character string length containing the c/r l/f will be 22 characters.

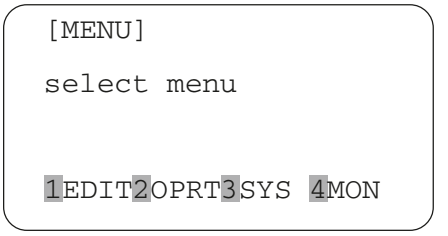
12-3-2 Stop messages

No. 60	Message	program end
	Meaning	Execution has stopped because the program has ended.
No. 61	Message	stop key
	Meaning	Execution has stopped because the Stop key on the TPB was pressed.
No. 62	Message	interlock
	Meaning	Execution has stopped because an I/O interlock was applied.
No. 63	Message	stop command
	Meaning	Execution has stopped because the STOP command was carried out.
No. 64	Message	key release
	Meaning	Execution has stopped by the hold-to-run function.

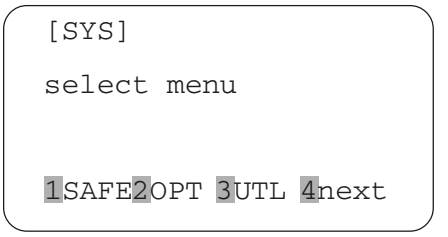
12-4 Displaying the Error History

A history of past errors can be displayed. Up to 100 errors can be stored in the controller.

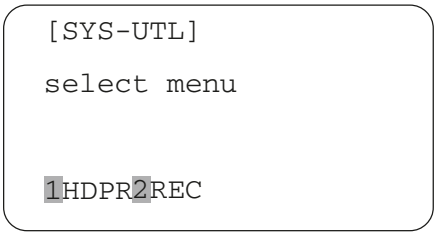
1) On the initial screen, press F3 (SYS).



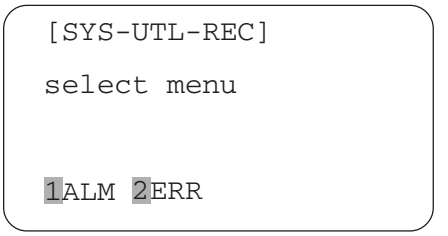
2) Next, press F4 (next) to change the menu display and then press F3 (UTL).



3) Press F2 (REC).



4) Press F2 (ERR).



5) History numbers, time that errors occurred (total elapsed time from controller start-up) and error descriptions are displayed. One screen displays the past 4 errors in the order from the most recent error.
Pressing the [X/Z-] and [X/Z+] keys displays the hidden items.
Press the [STEP UP] and [STEP DOWN] keys to sequentially scroll through the error list.

```
00:00101,05:11:12,CM
01:00096,18:10:02,PI
02:00080,10:07:33,CM
03:00015,20:35:45,CM
```

00 : 00101,05:11:12, CMU, 62:Interlock
↓ ① ↓ ② ↓ ③ ↓ ④

① History number

② Time the error occurred

(The above example means that the error occurred 101 days, 5 hours, 11 minutes and 12 seconds after controller start-up.)

③ Movement command control mode immediately before the error occurred.

CMU: TPB or RS-232C control
PIO: Parallel I/O control
SIO: Serial I/O control
WIO: Remote command (CC-Link) control

④ Error description

(See "12-1 Error Messages" and "12-3 Stop Messages".)

6) To return to the previous screen, press (ESC).

```
[SYS-UTL-REC]
select menu


1ALM 2ERR
```

MEMO

Chapter 13 TROUBLESHOOTING

This chapter explains how to take corrective action when a problem or breakdown occurs, by categorizing it into one of two cases depending on whether or not an alarm is output from the controller.

13-1 If A Trouble Occurs

If trouble or breakdown occurs, contact YAMAHA or your YAMAHA dealer, providing us with the following information in as much detail as possible.

Item	Description (example)
What you were using	・Controller model name : SRCP-xxxx ・Robot model name : MR16T-400 ・Controller version : Ver. 24.00 ・Power : AC 200V ・I/O 24V power supply : External power supply was used.
When	・When purchased ・How long used, how often used ・Problem happened at power on? One hour after power was turned on.
Under what conditions	・During automatic operation ・While writing a program ・When the robot was at a specific position
What happened	・Servo does not lock. ・Alarm (No. and message) is issued. ・Motor makes an unusual sound. ・A program was lost.
How often	・Always occurs. ・Occurs once an hour. ・Cannot be made to occur again.

13-2 Alarm and Countermeasures

If the READY signal is turned off except in cases of emergency stop, then an alarm has probably been issued. The status LED on the front panel of the controller lights up in red.

13-2-1 Alarm specifications

■ **If an alarm is issued:**

If an alarm is issued, keep the power turned on and connect the TPB or set the POPCOM on-line to check the contents of the alarm. An alarm message appears on the screen.

The transmission format for alarm messages is as follows.

```
<alarm number> : <alarm message> c/r l/f
```

The <alarm number> is displayed in two digits, so a one-digit number is prefixed with 0 like 01. The <alarm message> is displayed in a 17 character string length. (Spaces are added until the message contains 17 characters.) Therefore, an message including c/r and l/f consists of 22 characters.

■ **To cancel the alarm:**

To cancel the alarm, turn the power off and after first eliminating the problem, turn it back on again.

If an alarm is still issued while the power is turned on, then try turning the power on while the robot is in emergency stop. No alarm detection is performed with this method, so that the data can be checked, corrected or initialized. Normal alarm detection is performed when the servo is turned on after cancelling emergency stop.

13-2-2 Alarm message list

Alarm No.	Alarm Message	Meaning	Possible Cause	Action
01	OVER LOAD	Excessive load on motor	① Improper operation ② Motor failure ③ Parameter error ④ Wrong power supply voltage ⑤ Insufficient power supply capacity ⑥ Excessive friction in robot	① Lower the operation duty on the robot or reduce the acceleration parameter, or correct the payload parameter. ② If the motor armature resistance is too low, contact our sales office or representative. ③ Initialize the parameters and check the robot type setting. ④ Check the power supply voltage. ⑤ Check the power supply capacity. If too low, use a power supply of larger capacity. ⑥ Check whether the robot moving parts work sluggishly. Refer to the robot user's manual for corrective action.
02	OVER CURRENT	Motor drawing too much current	① Motor wire shorted ② Motor failure ③ Controller failure ④ Parameter error	① Check the motor wires for electrical continuity, and replace the motor assembly if abnormality is found. ② Replace the motor if internally shorted. ③ If the motor terminals U and W , V and W or U and V are shorted, the output transistor is defective, so replace the SRCP controller. ④ Initialize the parameters and check the robot type setting.
03	OVER HEAT	Transistor has heated to 90°C or more.	① Rise in ambient temperature (above 40°C) ② Excessive load on motor ③ Defective transistor	① Correct the ambient environmental conditions. (Install a cooling fan.) ② Lower the operation duty on the robot. ③ If the controller is being used correctly, the transistor is probably defective, so replace the SRCP controller.
04	POWER DOWN	Power supply voltage has dropped to less than 85% of rated value.	① Insufficient power supply capacity ② Wrong power supply voltage being used	① Check the power supply capacity. If insufficient, use a power supply having larger capacity. (Power is consumed mostly during return-to-origin of stroke end detection, robot start-up and acceleration/deceleration.) ② Check the voltage specifications indicated on the front panel of the controller.
05	BATT.LOW-VOLTAGE	System backup battery voltage is low.	① Battery worn out. ② Battery failure	①② Replace the battery. (If not possible to replace it immediately, then temporarily set bit 3 of PRM 34 to "1").
06	24V POWER OFF	24V power is not supplied.	① 24V power supply is not connected to terminals No. 3 and No.4 of EXT. CN connector. ② Fuse has blown due to short-circuit or excessive current flow in the 24V circuit. ③ 24V power was supplied to EXT.CN after supplying AC power to the power terminal block.	① Check the 24V power supply connection. ② Check for short-circuit using a multi-meter or recheck the I/O connections. ③ 24V power must first be supplied to EXT. CN before supplying power to the power terminal block.

Alarm No.	Alarm Message	Meaning	Possible Cause	Action
07	P.E. COUNTER OVER	Overflow in position deviation counter	① Mechanical lock ② Motor wire is broken or connected wrong. ③ Parameter error	① Check whether robot moving parts are locked. ② Check the motor wire and position signal wire connections. ③ Initialize the parameters.
08	PNT DATA DESTROY	Point data has been corrupted.	① Backup circuit failure ② Power was turned off while writing data. ③ Data was destroyed by external noise.	①② In emergency stop, turn power on and check point data. If part of the data is defective, correct the data. If all data are defective, initialize the point data and then reload the data. If there is no problem with the data, perform rewriting on any data. ③ Check the surrounding environment for noise.
09	PRM DATA DESTROY	Parameter data has been corrupted.	① Backup circuit failure ② Power was turned off while writing data. ③ Data was destroyed by external noise.	①② In emergency stop, turn power on and initialize the parameters. ③ Check the surrounding environment for noise.
10	PGM DATA DESTROY	Program data has been corrupted.	① Backup circuit failure ② Power was turned off while writing data. ③ Data was destroyed by external noise.	①② In emergency stop, turn power on and check program data. If part of the program is defective, correct the data. If all data are defective, initialize the program data and then reload the data. If there is no problem with the data, perform rewriting on any data. ③ Check the surrounding environment for noise.
11	SYSTEM FAULT	Software problem	① External noise or other factors have disrupted software program. ② Overflow in receiving buffer . When communicating with a PC, the XON/ XOFF control communication parameter was not selected on the PC.	① Check the surrounding environment for noise. ② Select the XON/XOFF control.
12	Not used			
13	Not used			
14	FEEDBACK ERROR 1	Detection of runaway	① Parameter error ② Wrong robot type setting ③ Motor is miswired. ④ Position signal wire is misconnected.	① Initialize the parameters. ② Check the combination of the robot with the controller. ③ Check the motor wire connection. ④ Check the position signal wire connection.
15	FEEDBACK ERROR 2	Position signal discontinuity	① Position signal wire is broken.	① Check the position signal wire connection.

Alarm No.	Alarm Message	Meaning	Possible Cause	Action
16	ABNORMAL VOLTAGE	Excessive voltage (higher than 420V) generated	① Rise in regenerative absorption resistor temperature (above 120°C). ② Wrong power supply voltage	① Lower the operation duty on the robot, or install a cooling fan. ② Check the power supply voltage.
17	SYSTEM FAULT 2	Controller's internal LSI error	① Internal LSI failure or malfunction	① If the error occurs frequently, then the LSI is probably defective, so replace the SRCP controller.
18	FEEDBACK ERROR 3	Mechanical lockup	① The robot slider struck on an obstacle or mechanical damper. ② Motor is miswired. ③ Mechanical lockup detection sensitivity is too high. ④ Parameter error	① Remove the obstacle or correct the point data or origin position. ② Check the motor wire connection. ③ Adjust the mechanical lockup detection sensitivity (PRM31). ④ Initialize the parameters.
19	SYSTEM FAULT 3	CPU error	① External noise or other factors have disrupted software program. ② CPU failure or malfunction	① Check the environment for noise. ② If the error occurs frequently, then the CPU is defective. Replace the SRCP controller.
20	Not used			
21	Not used			
22	VERSION MISMATCH	Wrong combination of PB and controller	① The PB used does not match the controller.	① Replace the PB.
23	Not used			
24	Not used			
25	Not used			
26	FEEDBACK ERROR 4	Motor wire breakage or misconnection	① Motor wire is broken or connected wrong. ② Parameter error ③ Wrong power supply voltage ④ Insufficient power supply capacity	① Check the motor wire connection. ② Initialize the parameters. ③ Check the power supply voltage. ④ Check the power supply capacity. If too low, use a power supply of larger capacity.
27	POLE SEARCH ERROR	Failed to detect magnetic pole.	① Motor wire is broken or misconnected ② Position signal wire is misconnected. ③ Parameter error	① Check the motor wire connection. ② Check the position signal wire connection. ③ Initialize the parameters.

13-3 Troubleshooting for Specific Symptom

If any problems develop while the controller is being used, check the items below for the appropriate way to handle them. If the problem cannot be corrected using the steps listed below, please contact our sales office or sales representative right away.

13-3-1 Relating to the robot movement

No.	Symptom	Possible Cause	Items to Check	Action
1	Servo of robot does not lock even after power is turned on.	1) Power is not being supplied.	• Check that the status LED on the front panel of the controller lights up or flashes.	• Check the voltage on the power input terminal block. If the voltage is correct, replace the SRCP controller.
		2) Emergency stop is activated.	• If the READY signal of the I/O connector is off and no alarm has been issued, an emergency stop is in effect. • Check whether the status LED is flashing.	• Check whether the emergency stop button of the TPB or the I/O emergency stop input (between EMG1 and EMG2) is on.
		3) The servo is off.	• Check whether the servo has been turned off in the program, and whether the TPB has been plugged or unplugged without pressing the ESC switch. • Check whether the status LED is flashing.	• Turn the servo on with the I/O servo recovery input or from the TPB operation.
		4) An alarm has occurred.	• Connect the TPB and check whether an alarm is displayed. • Check whether the status LED is lit in red.	• Take corrective action according to the instructions in the alarm message list.
2	Program does not run correctly.	1) Misprogramming	• Run step operations to check whether the program is correct.	• Correct the program if necessary.
		2) A different program is selected.	• Reset the program and check whether the desired program is selected.	• Change the program to select the desired program.
		3) The selected program No. was switched when the program was loaded into the controller from the memory card.	• Reset the program and check whether the desired program is selected.	• Change the program to select the desired program.
3	Abnormal noise or vibration occurs.	1) Parameter setting error	• Check the parameter data.	• Initialize the parameters.
		2) Controller failure	• Try using another controller if available.	• If another controller operates normally, then the currently used controller is defective, so replace it. • Use the correct controller and robot combination.
		3) Robot is at fault.		• Refer to the robot user's manual.

No.	Symptom	Possible Cause	Items to Check	Action
4	Position deviation or offset occurs. If this occurs, leave the power on and perform return-to-origin. Depending on the results of the return-to-origin, there are two possible causes of the problem: • If position offset is not corrected by the return-to-origin: Mechanical offset - See causes 1) to 2). • If position offset is corrected by the return-to-origin: Electrical offset - See causes 3) to 4).	1) Robot is not securely installed.	• Make sure there is no loose parts where the robot is installed.	• Reinstall the robot securely.
		2) Robot is at fault.		• Refer to the robot user's manual.
		3) Malfunction caused by noise	• Check whether the motor case is properly grounded. • Check that the resistance between the motor case and the controller's FG terminal is 1 ohm or less, and also that the controller is properly grounded.	• If the controller is used near a unit that generates noise such as welding machines and electric discharge machines, move it as far away as possible. If the entire unit cannot be moved, then at least move the power supply away. It might be necessary to use a noise filter or isolating transformer depending on the trouble.
		4) Controller failure	• Try using another controller if available.	• If another controller operates normally, then the currently used controller is defective, so replace it.
5	During return-to-origin, the robot stops due to alarm after striking on the stroke end (overload).	1) Wrong robot type number setting	• Connect the TPB and check the robot type number.	
		2) Parameter setting error	• Check the origin detection method parameter (PRM13) setting.	• When the parameter setting is "1" (stroke-end detection method), initialize the parameters. • When the parameter setting is "0" (sensor method), set the parameter to "1".
		3) The origin position is inappropriate so the robot slider makes contact with the damper when at the origin.	• Use the TPB to check whether the alarm occurs before or after return-to-origin is complete. If the alarm occurs after return-to-origin is complete, the damper position is inappropriate.	• Contact our sales office or representative.
6	Robot starts moving at high speed when the power is turned on. • The SRCP controller has a safety circuit to detect wire breakage, but check the points listed on the right anyway.	1) Motor and/or position signal are miswired.	• Check the motor wire and position signal wire connection.	• Correct the connections.
		2) Parameter error		• Try initializing the parameters.
7	Robot speed is abnormally fast or slow.	1) Parameter setting error	• Check whether the robot setting displayed on the TPB matches the robot actually used.	• If they do not match, initialize the parameters.
		2) Speed setting was changed.	• Check the speed parameter (PRM30).	• Correct the parameter.

13-3-2 Relating to the I/O

No.	Symptom	Possible Cause	Items to Check	Action
1	Output signal cannot be controlled. • In cases other than cause 2, the output signal cannot be controlled even with the manual instruction of TPB general-purpose output.	1) Wiring to external devices is incorrect.	• Check the wiring. • Check the operation with the manual instruction of the TPB general-purpose output. (Refer to "7-4 Manual Control of General-Purpose Output".)	• Make the correct wiring by referring to the connection diagram.
		2) Misprogramming	• Connect the TPB and check the program.	• Correct the program.
		3) Output transistor is defective.	• Measure the voltage at the PLC input terminal. ON: 0.5V or less. OFF: +IN COM (+24V)	• Replace the SRCP controller if the output transistor is defective.
2	Robot will not move even with dedicated command input.	1) Return-to-origin has not yet been completed.	• Connect the TPB and check the operation.	• Reperform return-to-origin.
		2) Program cannot be run.	• Connect the TPB and check the operation.	• Eliminate the cause of error by referring to the error message.
		3) Signal pulse width is too narrow.	• Check that the signal pulse width is 50ms or more.	• Increase the signal pulse width ("on" duration).
		4) Time interval before inputting a dedicated command after canceling emergency stop is too short.	• After canceling emergency stop, allow at least 200ms before inputting a dedicated command before inputting a dedicated command.	• Increase the delay time.
		5) Interlock signal remains off.	• Check the signal by using the TPB DIO monitor.	• Turn on the interlock signal.
		6) Another dedicated command input is on.	• Check the signal input (by using a PLC monitor, etc.).	• Turn off the dedicated command input.

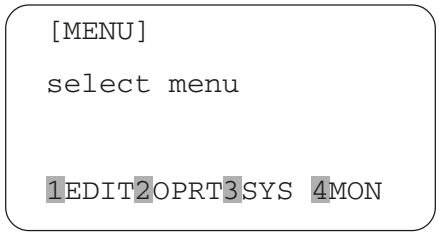
13-3-3 Other

No.	Symptom	Possible Cause	Items to Check	Action
1	An error occurs when the TPB is connected. The TPB cannot be used.	1) A dedicated I/O command input is on.	• Check the signal input (by using a PLC monitor, etc.).	• Always turn off dedicated command input signals when connecting the TPB to the controller.
		2) The TPB cable is broken.	• Check the cable wiring. • Try connecting another TPB if available.	• Replace the TPB if defective.
2	Programs can be input only up to NO. 31, or point data can be specified only up to P254, or DIO monitor display format is incorrect.	1) DPB is used as the TPB.		• Use TPB (Ver. 12.50 or later).
		2) TPB version is old.	• Check that the TPB version is 12.50 or later.	• Replace the ROM to upgrade the version.
		3) Communication cable specifications are wrong.	• Check whether the wrong cable (POPCOM cable, etc.) is being used.	• Use the specified communication cable. (POPCOM cable is different from the communication cable.) As an alternative, transmit "@DPBVER 250" in advance.
		4) POPCOM/WIN version is obsolete.	• Check that POPCOM/WIN version is 1.10.0 or later.	• Upgrade the POPCOM/WIN version.
		5) POPCOM/DOS is used.		• Use POPCOM/WIN.

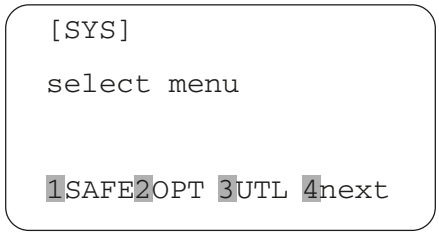
13-4 Displaying the Alarm History

A history of past alarms can be displayed. Up to 100 alarms can be stored in the controller.

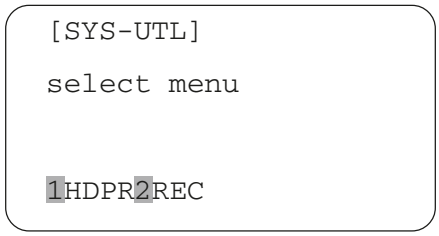
1) On the initial screen, press F3 (SYS).



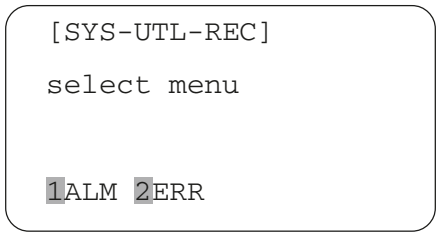
2) Next, press F4 (next) to change the menu display and then press F3 (UTL).



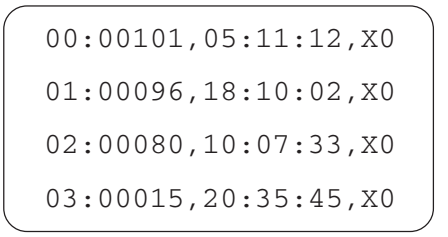
3) Press F2 (REC).



4) Press F1 (ALM).



5) History numbers, time that alarms occurred (total elapsed time from controller start-up) and alarm descriptions are displayed. One screen displays the past 4 alarms in the order from the most recent alarm.
Pressing the [X-/Z-] and [X+/Z+] keys displays the hidden items.
Press the [STEP UP] and [STEP DOWN] keys to sequentially scroll through the alarm list.



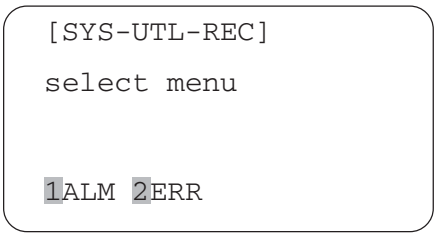
00 : 00101,05:11:12, X04: POWER DOWN
↓ ① ↓ ② ↓ ③

① History number

② Time the alarm occurred
(The above example means that the alarm occurred 101 days, 5 hours, 11 minutes and 12 seconds after controller start-up.)

③ Alarm description
(See "13-2-2 Alarm message list".)

6) To return to the previous screen, press (ESC).



Chapter 14 MAINTENANCE AND WARRANTY

For safety purposes, always turn the power off before starting robot maintenance, cleaning or repairs, etc.

14-1 Warranty

The YAMAHA robot and/or related product you have purchased are warranted against the defects or malfunctions as described below.

14-1-1 Warranty description

If a failure or breakdown occurs due to defects in materials or workmanship in the genuine parts constituting this YAMAHA robot and/or related product within the warranty period, then YAMAHA will repair or replace those parts free of charge (hereafter called "warranty repair").

14-1-2 Warranty Period

The warranty period ends when any of the following applies:

1) After 18 months (one and a half year) have elapsed from the date of shipment
2) After one year has elapsed from the date of installation
3) After 2,400 hours of operation

14-1-3 Exceptions to the Warranty

This warranty will not apply in the following cases:

1) Fatigue arising due to the passage of time, natural wear and tear occurring during operation (natural fading of painted or plated surfaces, deterioration of parts subject to wear, etc.)
2) Minor natural phenomena that do not affect the capabilities of the robot and/or related product (noise from computers, motors, etc.).
3) Programs, point data and other internal data that were changed or created by the user.

Failures resulting from the following causes are not covered by warranty repair.

1) Damage due to earthquakes, storms, floods, thunderbolt, fire or any other natural or man-made disasters.
2) Troubles caused by procedures prohibited in this manual.
3) Modifications to the robot and/or related product not approved by YAMAHA or YAMAHA sales representatives.
4) Use of any other than genuine parts and specified grease and lubricants.
5) Incorrect or inadequate maintenance and inspection.
6) Repairs by other than authorized dealers.

YAMAHA MOTOR CO., LTD. MAKES NO OTHER EXPRESS OR IMPLIED WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. THE WARRANTY SET FORTH ABOVE IS EXCLUSIVE AND IS IN LIEU OF ALL EXPRESSED OR IMPLIED WARRANTIES, INCLUDING WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES ARISING FROM A COURSE OF DEALING OR USAGE OF TRADE.
YAMAHA MOTOR CO., LTD. SOLE LIABILITY SHALL BE FOR THE DELIVERY OF THE EQUIPMENT AND YAMAHA MOTOR CO., LTD. SHALL NOT BE LIABLE FOR ANY CONSEQUENTIAL DAMAGES (WHETHER ARISING FROM CONTRACT, WARRANTY, NEGLIGENCE OR STRICT LIABILITY). YAMAHA MOTOR CO., LTD. MAKES NO WARRANTY WHATSOEVER WITH REGARD TO ACCESSORIES OR PARTS NOT SUPPLIED BY YAMAHA MOTOR CO., LTD.

14-2 Replacing the System Backup Battery

If an alarm is issued indicating that the system backup battery voltage is low, replace the battery using the procedure listed below.

(1) First, make a backup copy of all necessary data using a memory card or POPCOM software, because that data in the controller might be lost or destroyed during battery replacement.

(2) Unplug all connectors from the controller and then remove the top cover.

(3) You can now see the control board. Remove it from the controller.

(4) The lithium battery is soldered to the control board. Use a desoldering tool or similar tool to remove the solder and then remove the battery from the control board.

(5) Solder the new battery to the control board.

Battery product number: CR2450THE (Toshiba Battery)

(6) Install the control board back in its original position.

(7) Reattach the top cover.

(8) Initialize all data and then return the data you backed up into the controller.

> Please note that the state of California USA has legal restrictions on the handling of manganese dioxide lithium batteries. See the following website for more information:
> *http://www.dtsc.ca.gov/hazardouswaste/perchlorate*

14-3 Updating the System

YAMAHA may request, on occasion, that you update the system in your equipment. The following steps describe how to update the system.
Before updating the system, you must set up a system that allows communications between the controller and a PC (personal computer). Use a communication cable which conforms to the specifications listed in "11-2 Communication Cable Specifications".

(1) First, make a backup copy of the necessary data using a memory card or POPCOM software, because the data in the controller might be lost or destroyed while updating the system.

(2) With the controller started up, type "@SETUP" and press the Return (Enter) key.

(3) When a response "OK" is returned from the controller, turn off the power to the controller.

(4) Unplug the EXT.CN connector from the controller.

(5) With the EXT.CN connector still unplugged, turn on the power to the controller again.

(6) The controller enters the system setup mode and the YAMAHA copyright message appears on the PC screen.

(7) Type "@UPDATA" and press the Return (Enter) key.

(8) The controller then returns READY message, so transfer the new system data. (It will take about 5 minutes to transfer all the data.)

(9) An "OK" response is returned when the system transfer is complete. Now turn off the power to the controller.

(10) Plug in the EXT.CN connector.

(11) Turn on the power to the controller again. Type "@?VER" and press the Return (Enter) key. Then make sure that the controller version is updated.

(12) Initialize all data and then return the data you backed up into the controller.

⚠ ***CAUTION***

- ***The controller must remain in emergency stop until updating of the system is finished. (Specifically, terminal No. 1 (EMG 1) and terminal No. 2 (EMG 2) of EXT. CN should be left open.)***
- ***Before starting the system updating, we strongly recommend for safety reasons that the robot cable be disconnected from the controller.***

Chapter 15 SPECIFICATIONS

15-1 SRCP sereis

15-1-1 Basic specifications

Specification item \ Model					SRCP-05	SRCP-10	SRCP-20	SRCP-05A	SRCP-10A	SRCP-20A
Basic specifications	Applicable motor capacitance *1)				200V, 100W or less	200V, 200W or less	200V, 600W or less	200V, 100W or less	200V, 200W or less	200V, 600W or less
	Max. power consumption				400VA	600VA	1000VA	400VA	600VA	1000VA
	External dimensions				W78×H250×D157mm					
	Weight				1.5kg					
	Power supply voltage				Single-phase AC200 to 230V within ±10% 50/60Hz			Single-phase AC100 to 115/200 to 230V within ±10% 50/60Hz		
Axis control	No. of controllable axes				1 axis					
	Control method				AC full digital servo					
	Position detection method				Magnetic linear scale					
	Speed setting				1 to 100% in 1% steps					
	Acceleration setting				Automatically set according to robot type and payload. Acceleration can also be set by parameter (1 to 100% in 1% steps)					
	Servo adjustment				Handled with parameters (special). Servo gain, current limit, etc.					
Memory	ROM				256K bytes (with CPU incorporated)					
	RAM				128K bytes with lithium battery backup (5-year life)					
	No. of program steps				3000 steps or less in total, 255 steps/program					
	No. of programs				100					
	No. of points				1000 points					
	No. of multi tasks				4					
	Point teaching				MDI (coordinate value input), teaching playback, direct teaching					
	Auxiliary memory unit				IC memory card is available as TPB option.					
Command mode	I/O	Normal mode			Robot operation by dedicated command input					
		Pulse train mode	Type		1. phase A/phase B 2. pulse/code 3. CW/CCW One of the above should be selected.					
			Mode		Line driver (+5V)					
			Frequency		Maximum 2 Mpps (line driver)					
	Serial communication (RS-232C)				1. Data transmit/receive, parameter setting and robot operation by communication commands 2. Data transmit/receive, parameter setting, point teaching and robot operation by TPB (option)					
I/O	I/O interface	Normal mode	Sequence (I/O) input		General-purpose: 8 points, dedicated input: 8 points					
			Sequence (I/O) output		General-purpose: 5 points, dedicated output : 3 points, open collector output					
		Pulse train mode	Sequence (I/O) input		General-purpose: 8 points, dedicated input: 5 points					
			Sequence (I/O) output		General-purpose: 5 points, dedicated output: 3 points, open collector output					
			Command pulse input	Name	PULS+, PULS-, DIR+, DIR-					
				Type	One of the above should be selected.					
				Mode	Line driver (+5V)					
		Feedback pulse output		Name	PA+, PA-, PB+, PB-, PZ+, PZ-, PZM+, PZM-					
				Type	phase A/phase B					
				Mode	Line driver (+5V)					
		Power supply for sequence I/O			External DC +24V input for sequence input/output					
	Brake output				Relay output (for 24V/300mA brake), 1 ch; Uses power supply for sequence I/O					
	Origin sensor input				Connectable to DC24V normally-closed sensor; Uses power supply for sequence I/O					
	Emergency stop input				Normally-closed contact input (origin return not required after emergency stop is released)					
	Serial interface				RS-232C, 1ch (for communication with TPB or PC)					
	Network (option)				CC-Link, DeviceNet, Ethernet, PROFIBUS					

Protective function	Error detection items	Overcurrent, overload, wire breakage, runaway, etc.
General specification	Ambient temperature	0 to 40°C
	Storage temperature	-10 to 65°C
	Ambient humidity	35 to 85%RH (no condensation)
	Noise immunity	Conforms to IEC61000-4-4 Level 2
Others	Peripheral options	TPB (Ver. 12.50 or later), IC memory card, support software POPCOM

*1) The regenerative unit (RGU-2) is required to operate a load with large inertia or a robot model specified by YAMAHA.

⚠ ***CAUTION***

Specifications and external appearance are subject to change without prior notice.

15-1-2 Robot number list

Each robot model has an identification number as listed in the table below. After you initialize the parameters, enter the correct robot number that matches the robot model actually connected to the controller.

Robot model	MR12T	MR16T	MR16TH	MR20F	MR20FH	MR25F	MF50
Robot number	512	516	536	520	521	525	550

⚠ ***CAUTION***

Production of the MR20F was discontinued in March, 2006.
Using the robot with wrong setting may cause malfunctions.

15-1-3 LED display

The table below shows the specifications of the operation status LED on the front panel of the controller.

LED display	Robot or controller operation status
Not lit	The power is off or fuse is blown.
Lit in green	Servo motor is on. (Ready to operate)
Lit in red	Error occurred. (Alarm is being issued.)
Flashes green (0.5 sec.) and red (0.5 sec.)	Emergency stop
Flashes green (1.5 sec.) and red (0.5 sec.)	Emergency stop is canceled. (Servo off)

15-2 TPB

15-2-1 Basic specifications

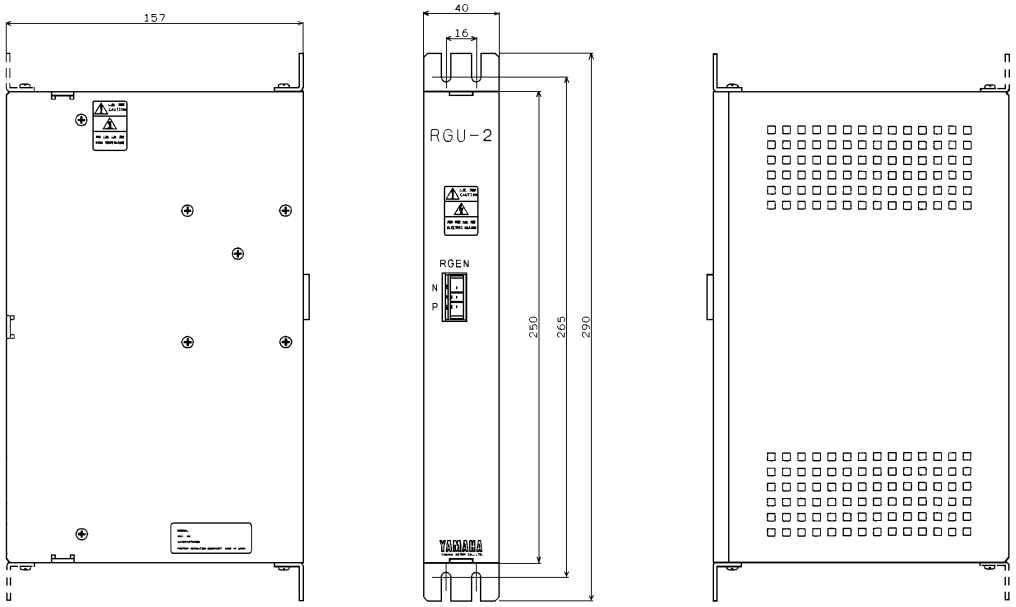
Specification item \ Model		TPB
Basic specifications	External dimensions	W107 H235 D47mm
	Weight	590g
	Power consumption	5 V, 200 mA max.
	Power supply	DC12V (supplied form the controller)
	Cable length	Standard 3.5m
I/O	Serial interface	RS-232C, one channel, for communications with controller
	Display	Liquid crystal, 20 characters 4 lines
	Keyboard	29 keys, membrane switch + emergency stop button
	Emergency stop button	Normally-closed contact (with lock function)
	Auxiliary memory device	IC memory card
General specification	Ambient temperature	0℃ to 40℃
	Storage temperature	-10℃ to 65℃
	Ambient humidity	35 to 85% RH (no condensation)
	Noise immunity	Conforms to IEC61000-4-4 Level 2
Others	Applicable TPB	Ver. 12.50 or later compatible with SRCP series

15-3 Regenerative Unit (RGU-2)

15-3-1 Basic specifications

Specification item / Model		RGU-2
Basic specifications	External dimensions	W40 H250 D157mm
	Weight	1.1kg
	Cable length	300mm
Special specifications	Regenerative voltage	Approx. 380V or more
	Regenerative stop voltage	Approx. 360V or less
General specifications	Ambient temperature	0℃ to 40℃
	Storage temperature	-10℃ to 65℃
	Ambient humidity	35 to 85% RH (no condensation)
	Noise immunity	Conforms to IEC61000-4-4 Level 2

15-3-2 Dimensions



MEMO

Chapter 16 APPENDIX

16-1 How to Handle Options

16-1-1 Memory card

A memory card (option) can be used with the TPB to back up the SRCP controller data.

■ Using the memory card

1. Insert the memory card into the TPB as shown in Fig. 16-1.
2. Back up the data by referring to section "10-6 Using a Memory Card" in Chapter 10.

Fig. 16-1 Inserting the memory card

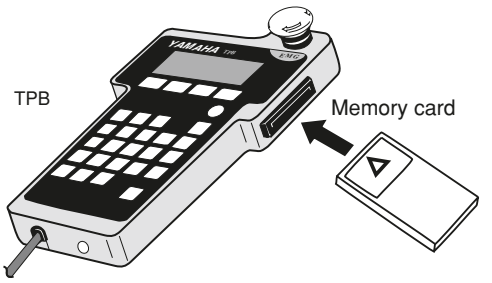
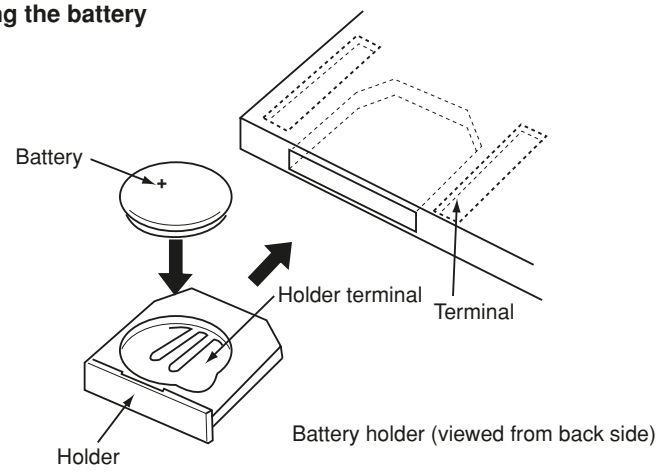


Fig. 16-2 Replacing the battery



■ Precautions when using the memory card

1. Insert the memory card all the way inwards until you feel it makes contact.
2. Be careful not to insert the memory card facing the wrong direction. The mark "Δ" should be facing upward. (A pin for preventing reverse insertion is provided.)
3. Insert or pull out the memory card only when the power is supplied to the TPB.
4. Never eject the memory card while backing up data.
5. The memory card should be used under the following environmental conditions:
   Ambient temperature range : -10 to 40°C
   Ambient humidity range : Below 85% RH
   Storage temperature range : -20 to 60°C
6. Do not leave the memory card inserted in the TPB when not in use, since this will shorten the battery life.
   The battery life is about 5 years (at ambient temperature of 25°C).
   If the battery voltage drops, an alert message appears on the TPB, so replace the battery by referring to Fig. 16-2.
   Battery product number : BR2325 or CR2325 (Panasonic) or equivalent type

■ Data size that can be saved

Data size that can be saved on one memory card is as follows:

Memory card capacity	DPB	TPB Ver. 2.18 or earlier	TPB Ver. 12.50 or later
8KB	Cannot be used.		
64KB	Cannot be used.		Up to 3 units of SRCP
1024KB (1MB)	Cannot be used.		Up to 48 units of SRCP

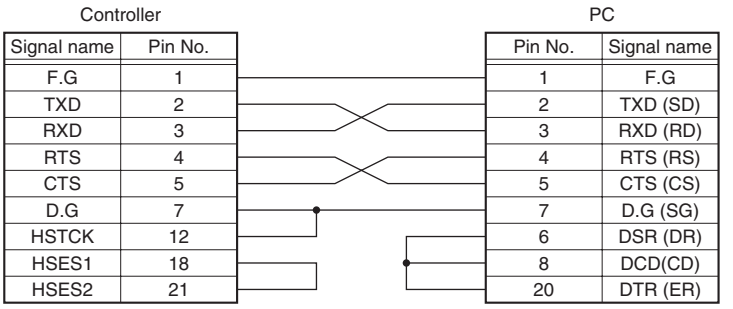
16-1-2 POPCOM communication cable

This cable is used to operate the SRCP controller from POPCOM software which runs on a PC and allows easy and efficient robot programming and operation.
This POPCOM cable is different from typical communication cables, so do not use it for other purpose.

Pins 18 and 21 on the SRCP controller are used for emergency stop input. Install a normally closed (contact B) switch of at least 50mA capacity between these pins when using emergency stop from the PC. Emergency stop is triggered when the switch opens the contact between pins 18 and 21.

Input response: 5ms or less
Input current: 33.3mA (DC24V)

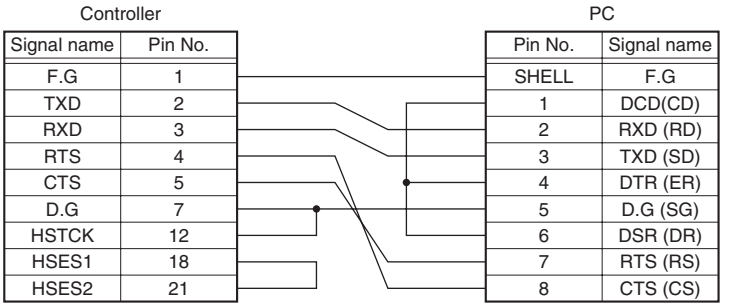
■ When the PC has a D-sub 25-pin connector:



Signal name	Pin No.
F.G	1
TXD	2
RXD	3
RTS	4
CTS	5
D.G	7
HSTCK	12
HSES1	18
HSES2	21

Pin No.	Signal name
1	F.G
2	TXD (SD)
3	RXD (RD)
4	RTS (RS)
5	CTS (CS)
7	D.G (SG)
6	DSR (DR)
8	DCD(CD)
20	DTR (ER)

■ When the PC has a D-sub 9-pin connector:



Signal name	Pin No.
F.G	1
TXD	2
RXD	3
RTS	4
CTS	5
D.G	7
HSTCK	12
HSES1	18
HSES2	21

Pin No.	Signal name
SHELL	F.G
1	DCD(CD)
2	RXD (RD)
3	TXD (SD)
4	DTR (ER)
5	D.G (SG)
6	DSR (DR)
7	RTS (RS)
8	CTS (CS)

The SHELL means a metallic casing of the connector.

⚠ ***CAUTION***
Pin 10 of the connector on the controller is used exclusively for connecting to the TPB. To prevent problems, do not attempt to wire anything to pin 10.

MEMO

Revision record

Manual version	Issue date	Description
Ver. 1.01	Nov. 2003	English manual Ver. 1.01 is based on Japanese manual Ver. 1.01.
Ver. 2.00	Apr. 2006	English manual Ver. 2.00 is based on Japanese manual Ver. 3.04.
Ver. 2.01	Aug. 2006	English manual Ver. 2.01 is based on Japanese manual Ver. 3.05.
Ver. 2.02	Jan. 2007	English manual Ver. 2.02 is based on Japanese manual Ver. 3.06.
Ver. 2.03	Jul. 2007	English manual Ver. 2.03 is based on Japanese manual Ver. 3.07.
Ver. 2.04	Oct. 2007	English manual Ver. 2.04 is based on Japanese manual Ver. 3.08.

User's Manual

YAMAHA Robot Controller SRCP

Oct. 2007
Ver. 2.04
This manual is based on Ver. 3.08 of Japanese manual.

IM Company