MITSUBISHI

Mitsubishi Electric Industrial Robots

CRn-500 series
CRn-700 series

RT ToolBox2 / RT ToolBox2 mini User's Manual

(3D-11C-WINE/3D-12C-WINE)

MELFA

BFP-A8618-A

⚠ Safety Precautions

Before using the robot, always carefully read the precautions below and the separate "Safety Manual" and take all necessary safety measures.

A. These show precautions based on Labor Health and Safety Regulations (Articles 36, 104, 150, 151).

⚠Caution For the sake of safety, teaching work should only be performed by workers who have received special education.
(The same is true for any maintenance work done with the power source not cut off.)
→Implementation of safety education

⚠Caution For teaching work, prepare work regulations concerning robot operation methods and procedures, measures for when there is an abnormality and when restarting, etc. Perform teaching work according to these regulations.
(The same is true for any maintenance work done with the power source not cut off.)
→Prepare work regulations.

⚠Warning For teaching work, set up a device that can stop operation immediately.
(The same is true for any maintenance work done with the power source not cut off.)
→Emergency stop switch setting

⚠Caution During teaching work, label the start switch etc. to indicate that teaching work is underway.
(The same is true for any maintenance work done with the power source not cut off.)
→Display that teaching work is underway

⚠Danger During operation, set up a fence or barrier to prevent contact between workers and the robot.
→ Setting up a safety fence

⚠Caution Determine a uniform signal to relevant staff for the start of operation and use that signal.
→ Signal for the start of operation

⚠Caution For maintenance work, in principle, cut off the power and label the start switch etc. to indicate that maintenance work is underway.
→Display that maintenance work is underway

⚠Caution Before starting work, check the robot, emergency stop switches, related devices, etc. and make sure there are no abnormalities.
→ Check before the start of work

B. This shows precaution points given in the separate "Safety Manual".
For details, please read the text of the "Safety Manual".

⚠Caution	Use the robot in an environment that is within the range of its specifications. Failure to do this can cause a drop in reliability and breakdown. (Temperature, humidity, atmosphere, noise, etc.)
⚠Caution	When transporting the robot, put it into its specified transport posture. Failure to do this can cause a drop in reliability and breakdown.
⚠Caution	Install the robot on a solid platform. If the robot is in an unstable posture, this can cause positional deviation and vibration.
⚠Caution	As much as possible, wire cables away from noise sources. If cables are brought too close to noise sources, this can cause positional deviation and malfunction.
⚠Caution	Do not apply excess force to a connector or bend a cable excessively. Doing so can cause a contact defect or cut line.
⚠Caution	Set work masses, including hands, so that they do not excess rated load or permitted torque. Exceeding either of these can cause an alarm or breakdown.
⚠Warning	Install hands and tools and hold work securely. Failure to do this can cause objects to fly loose during operation and cause personnel injury or damage.
⚠Warning	Ground the robot and controller reliably. Failure to do this can cause malfunction due to noise or in an extreme case, electrical shock.
⚠Caution	Display the operating state while the robot is operating. Lack of such a display can result in someone coming too close to the robot by mistake or mistaken operation.
⚠Warning	Always secure the priority right for control of the robot before doing any teaching work within the robot's operating range. Failure to do this can allow the robot to start upon instruction from the outside and cause personnel injury or damage.
⚠Caution	Make the jog speed as slow as possible and do not take your eyes off the robot. Failure to do this may cause a collision between a work piece and peripheral devices.
⚠Caution	After completing program editing but before starting automatic operation, always check operations with step operation. Failure to do this may cause a collision with a peripheral device due to a programming mistake or the like.
⚠Caution	Set up the safety fence in such a way that, while the equipment is running on automatic, either the safety fence door is locked or if anyone tries to open the door, the robot is stopped. Failure to take these protective measures can cause an accident resulting in injury.

⚠Caution Never on your own judgment make an alterations or use maintenance parts other than those designated. Doing so can cause breakdown and problems.

⚠Warning When moving the robot arm from the outside, never stick a hand or finger into an opening. Depending on the posture, the hand or finger could get caught in the equipment.

⚠Caution Do not switch the robot Off or make an emergency stop of the robot by switching Off the robot controller's main power supply.
If the robot controller's main power supply is switched Off during automatic operation, this can reduce the robot's precision. It could also cause the arm to fall or allow inertia to result in collisions with peripheral device or the like.

⚠Caution When rewriting a program, parameters, or other internal information within the robot's controller, do not switch Off the robot controller's main power supply.
If the robot controller's main power supply is switched Off during automatic operation or while a program or parameter is being rewritten, there is a danger of the internal information in the robot controller being destroyed.

⚠Caution For using RH-5AH/10AH/15AH series or RH-6SH/12SH/18SH series.
While pressing the brake releasing switch on the robot arm, beware of the arm which may drop with its own weight.
Dropping of the hand could lead to a collision with the peripheral equipment or catch the hands or fingers.

User's Manual Revision History

Printing Date	Manual No.	Revision Contents
2008/04	BFP-A8618-*	First edition (Corresponds to the Ver.1.1)
2008/08	BFP-A8618-A	Corresponds to the Ver.1.2 (Refer to the software revision history.)

Software Revision History

Version	Release Date	Revision Contents
1.0	2008/01	Initial release (Japanese version only)
1.1	2008/04	CRnQ Communications Added the GOT transparent function. Parameter editing Added the Multiple CPU setting screen.
1.2	2008/08	[Communication Setting] - Changed the Initial value of USB, TCP/IP and RS-232C setting. - Added the CRnQ communication routes when "Ethernet" is selected. (Added the Ethernet port communications.) [Program editor] - Added “Comment Selection”/”Uncomment Selection” function. - Added the function to edit the backed up program data. [Project] - Added the function to import the project. [Parameter] - Added the function to display the parameter changed from initial value. [Restore] - Added the function to restore individually data backed up by selecting "All files". etc.

PREFACE

Thank you for purchasing this MELFA Mitsubishi Electric industrial robot.
This document is the user's manual for the MELSOFT "RT ToolBox2" and "RT ToolBox2 mini".
This document will help you to use the functions of this software to the maximum over a wide range of stages, from initial robot start to program writing, editing, and management.
In order to operate the robot safely, carefully read this document and the safety manual that comes with the robot main unit before operating the robot. Also, store this manual carefully so that you can take it out and read it whenever needed.

Target versions for this document

This document is for the MELSOFT "RT ToolBox2" and "RT ToolBox2 mini" Ver. 1.2 and supports the following robot controllers.

- CRn-500 series controllers
- CRnQ-700 series controllers
- CRnD-700 series controllers
  In some locations, this document writes about the "CRn-700 series".

Target readers for this document

This document assumes that the reader understands basic Microsoft Windows operation methods and the robot controller.
Those who have not mastered basic computer operation methods should read the user's manual for their computer.

Notation method in this document

⚠**Danger**	This indicates an item for which incorrect handling could present imminent danger of death or injury.
⚠**Warning**	This indicates an item for which incorrect handling could present a danger of death or injury.
⚠**Caution**	This indicates an item for which incorrect handling could present a danger of impairment. It could also present a danger of just physical damage.

This document uses the following general terms and abbreviations

General Term/Abbreviation	Contents
RT ToolBox2	General name for the RT ToolBox2 and RT ToolBox2 mini To distinguish them in explanations, these two are called the "standard edition" and "mini edition".
Universal model QCPU	General term for Mitsubishi PLC CPU modules of Q02U, Q03UD, Q03UDE, Q04UDH, Q04UDEH, Q06UDH, Q06UDEH, Q13UDH, Q13UDEH, Q26UDH and Q26UDEH.
Built-in Ethernet port QCPU	General term for Mitsubishi PLC CPU modules of Q03UDE, Q04UDEH, Q06UDEH, Q13UDEH and Q26UDEH.
GX Developer	Abbreviation of SW D5C-GPPW-E(-EV) / SW D5F-GPPW-E type of Mitsubishi PLC programming software package.

Contents

1. Usage

This explains precautions you need to know before using this software.

1.1. How to Use this Document

The manual is in the CR-ROM as the Adobe PDF file.
D:/Doc/BFP-A8618.pdf (Example for the CD-ROM drive is “D:”.)

For reading the manual, Adobe Acrobat Reader Ver.5.0 or more is required.
If Adobe Acrobat Reader isn’t installed, please download from following Adobe Systems Incorporated URL (As of December, 2007)
URL: http://www.adobe.com/

1.2. Checking the Product

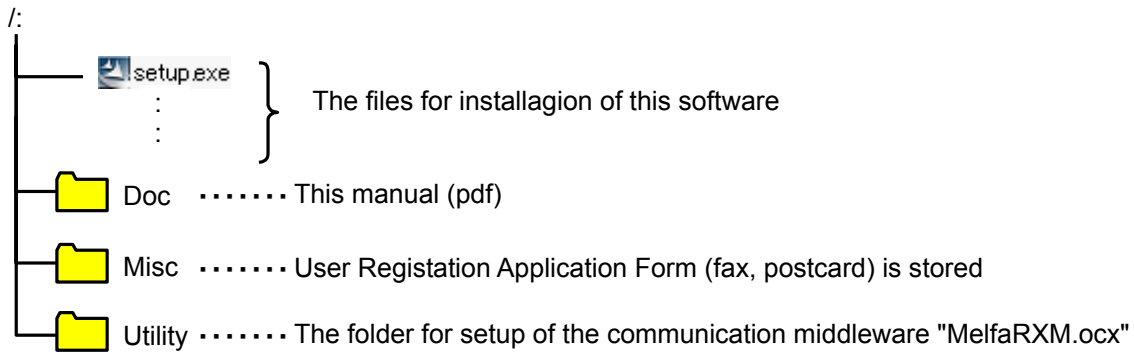
1.2.1. Checking the package

Please check if all items shown below are included in the package.
- CD-ROM "RT ToolBox2"
- Setup Guide
- END-USER SOFTWARE LICENSE AGREEMENT
- License Certification
  (Please make sure Product ID is printed on it.)

* Please contact the branch office or the agency if there is some shortage in the package.

1.2.2. Checking the CD-ROM contents

The CD-Rom has the following configuration.



1.2.3. About the "MelfaRXM.ocx" communications middleware

MelfaRXM.ocx is the ActiveX control that communicates to robot-controller. You can create the Windows Application of "MELFA ROBOT" by using this control.
You can use "MelfaRXM.ocx" in only standard version of this software.
For information on how to set up "MelfaRXM.ocx", refer to **"14 MelfaRXM.ocx Communications Middleware Setup"**.
In case of using only the function of **“RT ToolBox2”**, you don't need to install this software.

1.3. Items to be prepared by the customer

This explains what the customer needs to prepare in order to use this software

1.3.1. Computer system

Use a computer that meets the specifications given in: **"1.4 Operating Environment".**

1.3.2. Computer cable

Prepare the cable for connecting the controller and the computer. The cable required depends on the connection specifications and controller used, as shown below.

For the RS-232 cable refer to the "Standard Specifications" for your robot.

Table 1-1 CRnD-700 Series, CRn-500 Series Communication Cables

Method	Description		Model name	Manufacturer
USB	USB A type, USB mini B type		-	-
Ethernet	10BASE-T, 100BASE-TX		-	-
RS-232	For controller front panel	CRnD-700 series	2D-232CBL03M	Mitsubishi Electric
		CRn-500 series	RS-MAXY-CBL RS-AT-RCBL (for expansion serial interface (option))	Mitsubishi Electric
	For expansion option box (CR1-EB3)		RS-AT-RCBL	Mitsubishi Electric

Table 1-2 CRnQ-700 Series Communication Cables

Method	Description	Cables confirmed by Mitsubishi Electric to operate properly	
		Model name	Manufacturer
USB	USB A type to mini B type	ZUM-430	Loas Co.
		USB-M53	Elecom Co.
		GT09-C20USB-5P	Mitsubishi Electric System Service
		MR-J3USBCBL3M	Mitsubishi Electric
Ethernet	10BASE-T, 100BASE-TX		
RS-232	For connecting by personal computer - PLC CPU (when Personal computer connector is D-sub, 9-pin)	QC30R2	Mitsubishi Electric

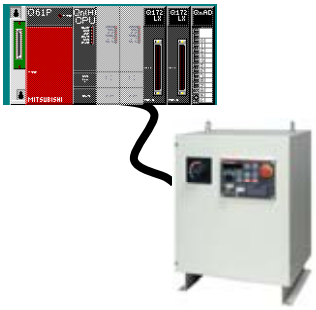
1.4. Operating Environment

This explains the operating environment.

1.4.1. Connectable robot controllers

This software can be connected with the robot controllers shown below

Table 1-3 Connectable Robot Controllers and Communications Types

Robot controller	Communications (*1)		Remark
CrnD-700 series 	USB (*2)		
	Ethernet (TCP/IP)		
	RS-232		
CrnQ-700 series 	CRnQ communications	USB (*2)	
		Ethernet (TCP/IP)	The PLC Ethernet interface module or Built-in Ethernet port QCPU (*3) is required.
		RS-232	
CRn-500 series 	Ethernet (TCP/IP)		The robot controller must have the "Ethernet interface" option.
	RS-232		

(*1) The computer must have each ports for communications.
(*2) When using USB connection, 1 computer can connect to only 1 robot controller.
(*3) Built-in Ethernet port QCPU can be used with this software Ver.1.2 or later.

This software can be connected to a maximum of 32 controllers at the same time. These controllers may be different models.

1.4.2. Computer system

This software operates on PC/AT compatible computers that meet the following specifications.

Item	Recommended environment
CPU	Pentium III 1 GHz or higher However, for using the simulation function, Pentinum IV 2 GHz or higher.
Main memory	512 MB min.
Hard disk	Available capacity 300 MB min.
Display	XGA (1024x768) or higher
Optical device	CD-ROM drive
Keyboard	PC/AT compatible keyboard
Pointing device	Must operate on Windows
Communications functions Communications port	- USB2.0 (Caution: This cannot be used for connection with the CRn-500 series controller.) - LAN: 100Base-TX/10Base-T - RS-232 communications port that operates on Windows (Minimum 9600bps: 1 port) Must have one of the above interfaces
OSs for which operation is warranted	Windows 2000 Professional Windows XP Professional (*1) Windows XP Home Edition (*1) Windows Vista Ultimate (*1) Windows Vista Business (*1) Windows Vista Home Premium (*1) Windows Vista Home Basic (*1) English versions for each edition

*1: Only supports 32-bit versions

1.5. Installation, Uninstallation

This section explains the method for installing the software and the method for uninstalling it.

Caution

Uninstall the old version before installing the new one.

If an old version of "RT ToolBox2" is installed, uninstall the old version of "RT ToolBox2", then install the new version of "RT ToolBox2".

1.5.1. Installation

Caution

When installing, log in as a user with administrator authority.

When installing, log in as a user with administrator authority. The system will not let you install if you log in as a user who does not have administrator authority.

Install this software with the procedure below.

(1) When you insert this product into the computer's CD-ROM drive, the setup screen is displayed automatically.

(2) If the setup screen is not displayed when you insert this product into the computer's CD-ROM drive, display the setup screen with the following method.
- For any OS other than Windows Vista
- In Windows Vista, when using the [Start] menu with the classic display
  1) Select [Start] button -> [Run].
  2) Check the CD-ROM drive name, then input "drive name":\Setup.exe.
     (If the CD-ROM drive is "D:", input "D:\Setup.exe".)

- In Windows Vista, when not using the [Start] menu with the classic display
  1) Click [Start] button -> [All Programs] -> [Accessories], then select [Run].
  2) Check the CD-ROM drive name, then input "drive name":\Setup.exe.
     (If the CD-ROM drive is "D:", input "D:\Setup.exe".)

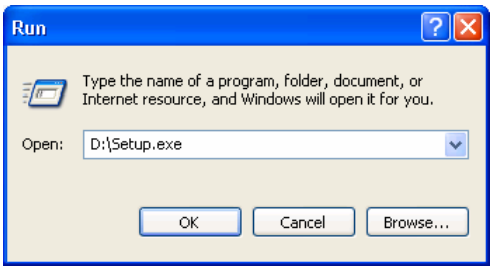
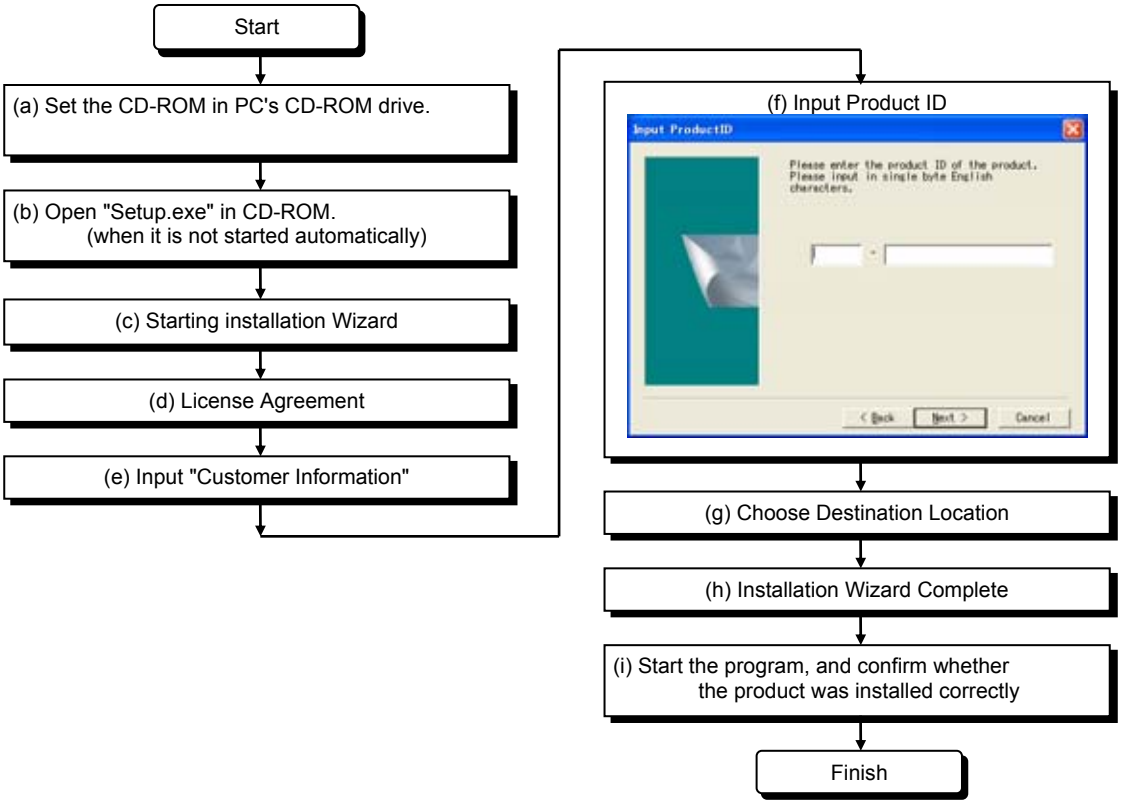


Figure 1-1 Specify the File Name and Execute

With Windows Vista, when using the classic display, when not using the [Start] menu with the classic display, you can use the [Start] menu Search box instead of executing the [Run] command.



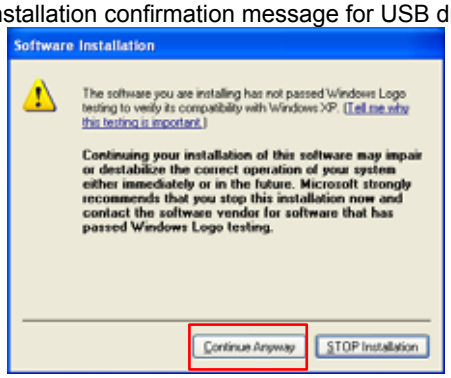
* Product ID is printed on the Certificate of License permission
* After the installation is completed, the computer should be likely to be rebooted.

⚠ Caution

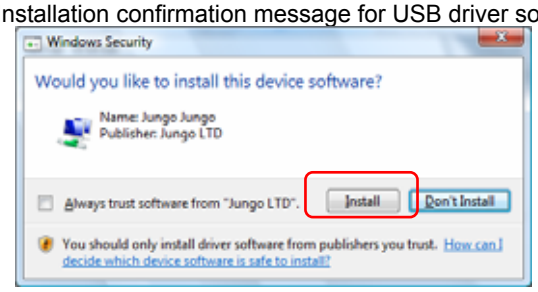
About the confirmation and warning message displayed during installation

During installation on Windows XP or Windows Vista, the following confirmation and warning messages are displayed, but select to continue installation. If you select not to install, please execute the installation again.

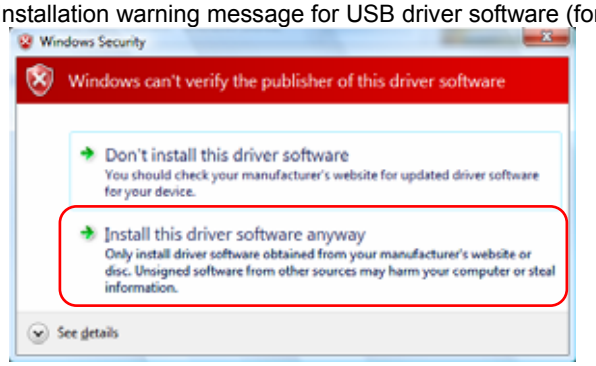
(1) Installation confirmation message for USB driver software (for Windows XP)



(2) Installation confirmation message for USB driver software (for Windows Vista)



(3) Installation warning message for USB driver software (for Windows Vista)



<Remark>
We have confirmed operation at our company. No problem occurs after installation.

1.5.2. Uninstall

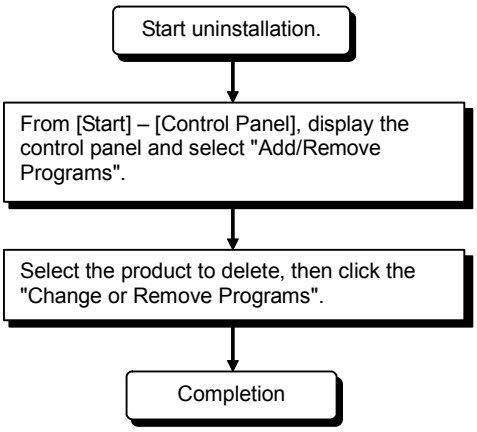
Uninstall with the following method.

- For any OS other than Windows Vista
  Execute [Start] – [Control Panel] [Program Add and Delete].

- Windows Vista
  Open [Start] – [Control Panel].

  When not using the classic display
  With [Control Panel] [Program], execute [Uninstall Program].

  For classic display
  With [Control Panel] [Program Functions], select the application name, then execute the uninstallation.



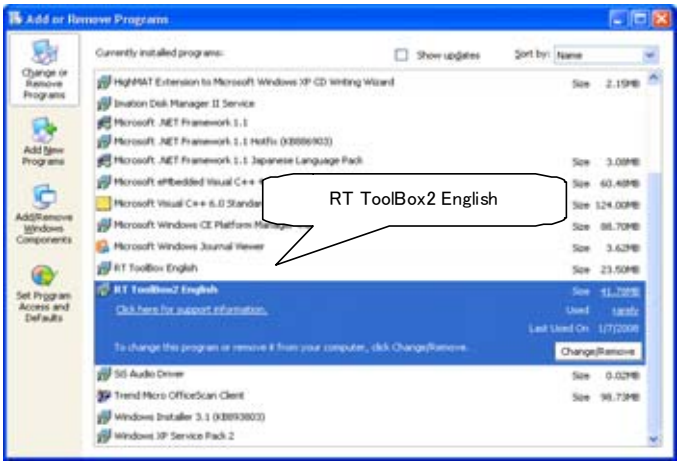


Table 1-2 Uninstalling Applications (WindowsXP)

1.5.3. USB driver (CRnD-700 series robot controller) installation

Connecting the CRnD-700 series robot controller with USB requires installation of the robot USB driver. Install with the following procedure.

⚠ Caution

If the USB driver cannot be installed, check the following setting.

<When Windows 2000 is used>
If you have selected "Block-Prevent installation of unsigned files" after [Control Panel] - [System] - [Hardware] - [Driver Signing], the USB driver may not be installed.
Choose "Ignore-Install all files, regardless of file signature" or "Warn-Display a message before installing an unsigned file" for [Driver Signing], and install the USB driver.

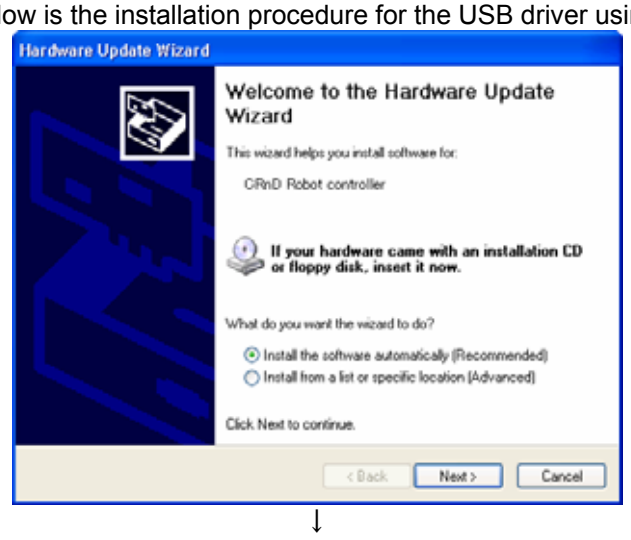
<When Windows XP is used>
If you have selected "Block-Never install unsigned driver software" after [Control Panel] - [System] - [Hardware] - [Driver Signing], the USB driver may not be installed.
Choose "Ignore-Install the software anyway and don't ask for my approval" or "Warn-Prompt me each time to choose an action" for [Driver Signing], and install the USB driver.

1.5.3.1. When using Windows 2000

When you connect the CRnD-700 robot controller and the computer with a USB cable, installation starts and completes automatically.

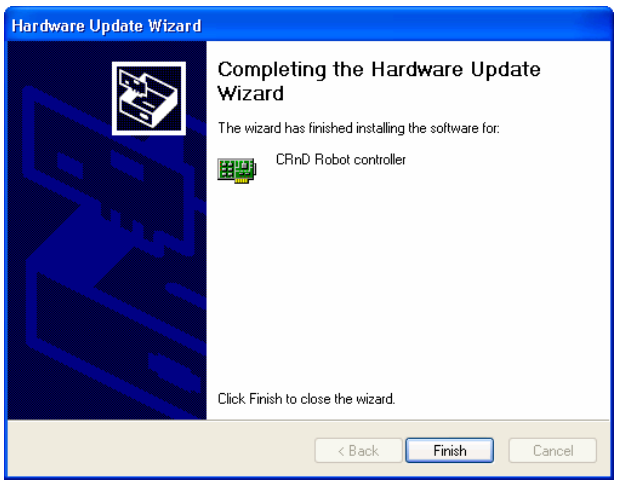
1.5.3.2. When Using Windows XP

Below is the installation procedure for the USB driver using Windows XP (Professional).



↓

1) When you connect the computer and CRnD-700 series robot controller with a USB cable, the screen on the left is displayed. Select "Install the software automatically (Recommended)", then click the [Next] button. Installation of the USB driver starts.



↓

(Completed)

2) When the screen on the left is displayed, the installation is complete. Click the [Finish] button to end the installation.

1.5.3.3. When using Windows Vista

When you connect the CRnD-700 robot controller and the computer with a USB cable, installation starts and completes automatically.

1.5.4. CRnQ communications USB driver installation

Connecting the CRnQ-700 series robot controller with USB requires installation of the robot USB driver. Install with the following procedure.

⚠ Caution

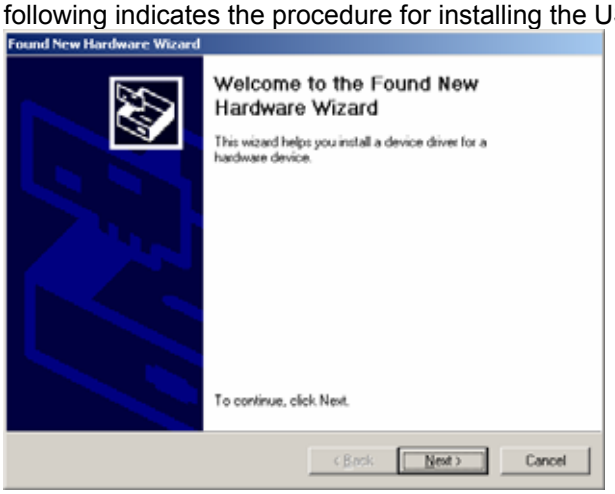
If the USB driver cannot be installed, check the following setting.

<When Windows 2000 is used>
If you have selected "Block-Prevent installation of unsigned files" after [Control Panel] - [System] - [Hardware] - [Driver Signing], the USB driver may not be installed.
Choose "Ignore-Install all files, regardless of file signature" or "Warn-Display a message before installing an unsigned file" for [Driver Signing], and install the USB driver.

<When Windows XP is used>
If you have selected "Block-Never install unsigned driver software" after [Control Panel] - [System] - [Hardware] - [Driver Signing], the USB driver may not be installed.
Choose "Ignore-Install the software anyway and don't ask for my approval" or "Warn-Prompt me each time to choose an action" for [Driver Signing], and install the USB driver.

1.5.4.1. When using Windows 2000

The following indicates the procedure for installing the USB driver when using Windows 2000.

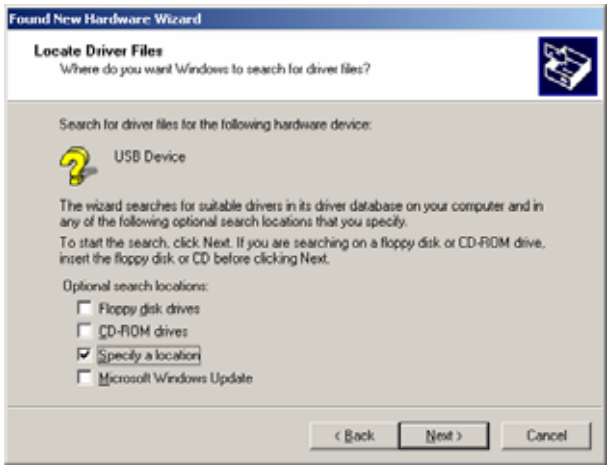


1) The screen shown on the left appears when you connect the personal computer and Universal model QCPU by the USB cable.
Click the [Next] button.

↓

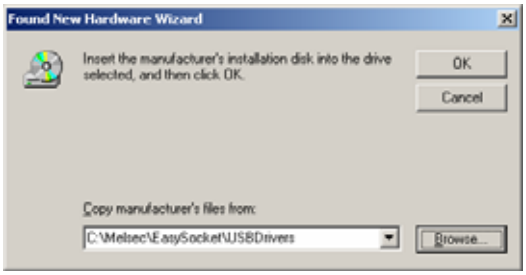
2) Choose "Search for a suitable driver for my device [recommended]" and click the [Next] button.

↓



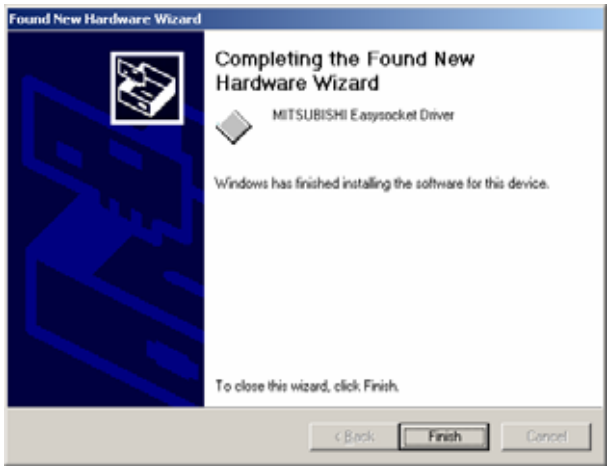
3) Check "Specify a location" and click the [Next] button.

↓



4) As the left screen appears, set the "C:\Melsec\EasySocket\USBDrivers" and click the [Next] button.
If volume MELSOFT products have been installed, browse the installation destination "EasySocket\USBDrivers" of the first product.

↓



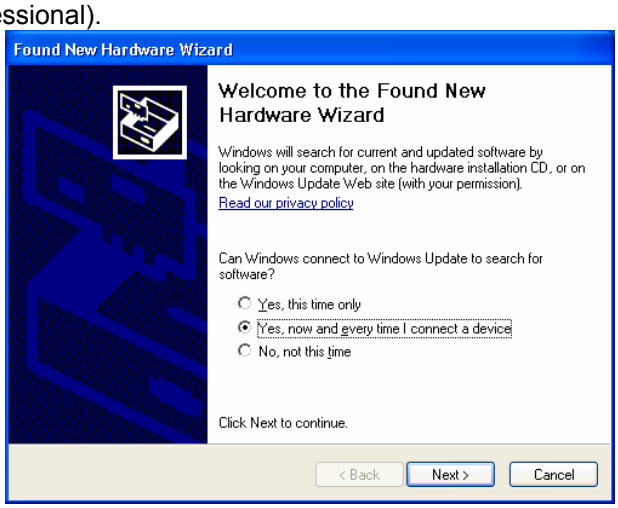
5) The screen on the left appears to indicate completion of installation.
Click the [Finish] button to terminate installation.

↓

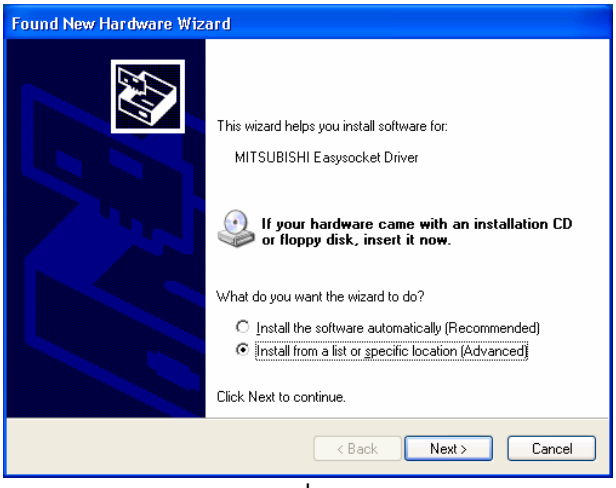
(Completed)

1.5.4.2. When using Windows XP

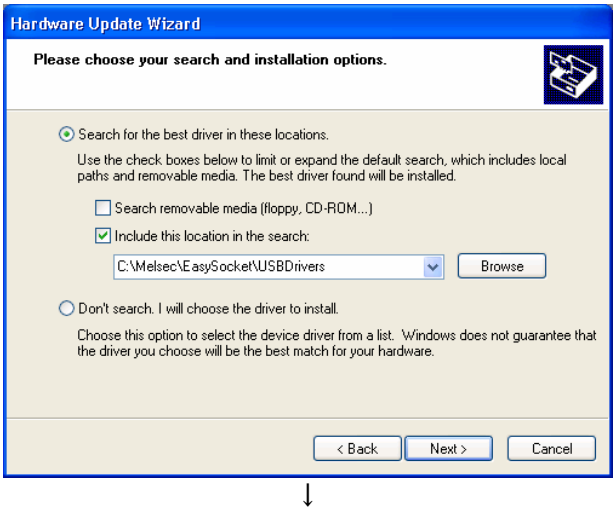
The following indicates the procedure for installing the USB driver when using Windows XP (Professional).



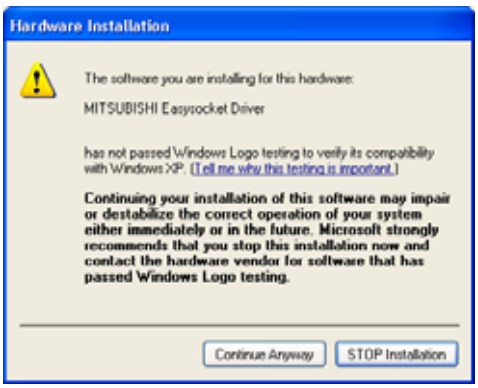
1) The screen shown on the left appears when you connect the personal computer and Universal model QCPU by the USB cable.
Choose "Yes, now and every time I connect a device" and click the [Next] button.



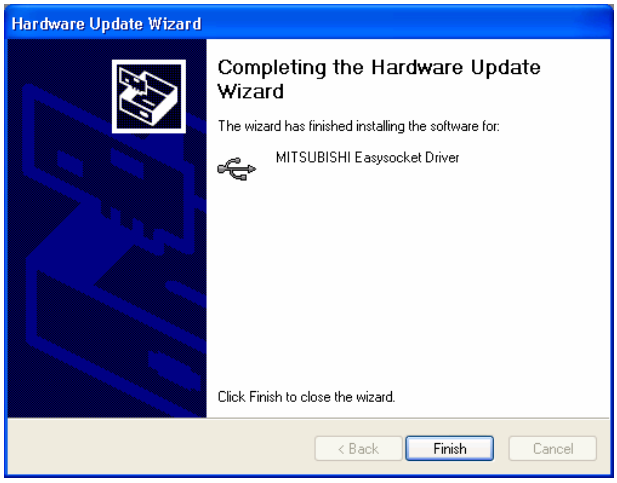
2) As the screen on the left appears, choose "Install from a list or specific location [Advanced]" and click the [Next] button.



3) As the screen on the left appears, choose "Search for the best driver in these locations".
Check "Include this location in the search" and set the “C:\Melsec\EasySocket\USBDrivers”.
After setting, click the [Next] button.
If volume MELSOFT products have been installed, browse the installation destination "EasySocket\USBDrivers" of the first product.



↓



↓

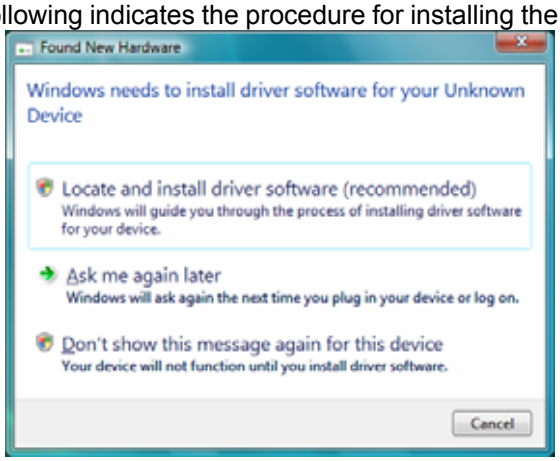
(Completed)

4) As the screen on the left appears, click the [Continue Anyway] button to continue the installation of the USB driver.
(No problem will occur after installation of the USB driver.)

5) The screen on the left appears to indicate completion of installation.
Click the [Finish] button to terminate installation.

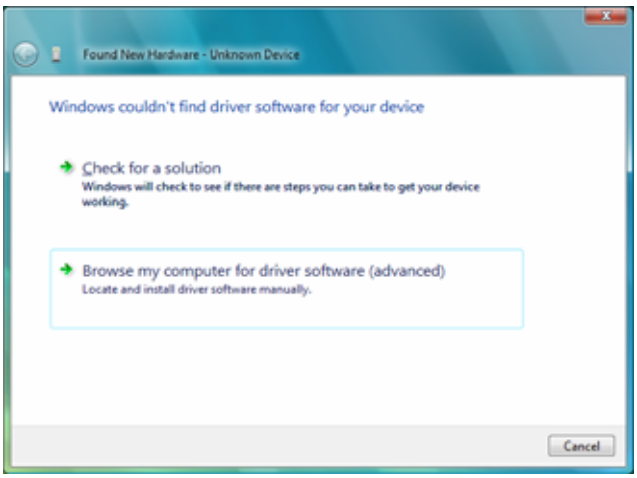
1.5.4.3. When using Windows Vista

The following indicates the procedure for installing the USB driver when using Windows Vista (Business).



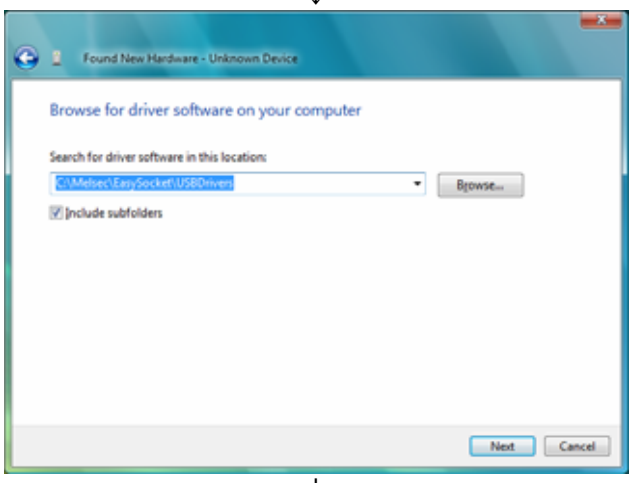
↓

1) The screen shown on the left appears when you connect the personal computer and Universal model QCPU by the USB cable.
Select "Locate and install driver software (recommended)" and wait for the search to end.



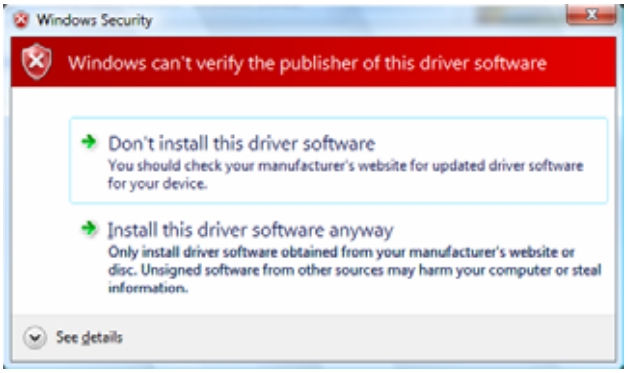
↓

2) The screen on the left is displayed, so select "Browse my computer for driver software (advanced)".



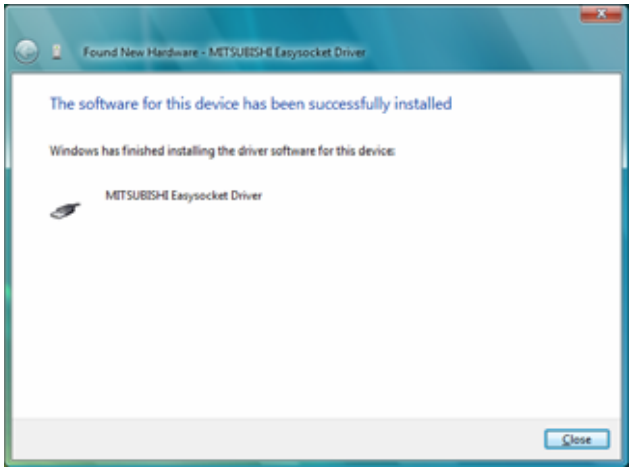
↓

3) The screen on the left is displayed, so select "C:\Melsec\EasySocket\USBDrivers". After making the setting, click the [Next] button.
When multiple MELSOFT products are installed, set the default product installation folder "EasySocket\USBdrivers".



↓

4) The screen on the left is displayed, so select "Install this driver software anyway".



↓

5) The screen on the left is displayed. Click the "Close" button. This completes the installation.

↓

(Completed)

1.5.5. CRnQ Communications USB driver for GOT transparent installation

Connecting the CRnQ-700 series robot controller with USB via GOT transparent mode requires installation of the USB driver for the GOT transparent. Install with the following procedure.
This function is available from RT ToolBox2 Ver.1.1 or later.

⚠ Caution

If the USB driver cannot be installed, check the following setting.

<When Windows 2000 is used>
If you have selected "Block-Prevent installation of unsigned files" after [Control Panel] - [System] - [Hardware] - [Driver Signing], the USB driver may not be installed.
Choose "Ignore-Install all files, regardless of file signature" or "Warn-Display a message before installing an unsigned file" for [Driver Signing], and install the USB driver.

<When Windows XP is used>
If you have selected "Block-Never install unsigned driver software" after [Control Panel] - [System] - [Hardware] - [Driver Signing], the USB driver may not be installed.
Choose "Ignore-Install the software anyway and don't ask for my approval" or "Warn-Prompt me each time to choose an action" for [Driver Signing], and install the USB driver.

1.5.5.1. When using Windows 2000

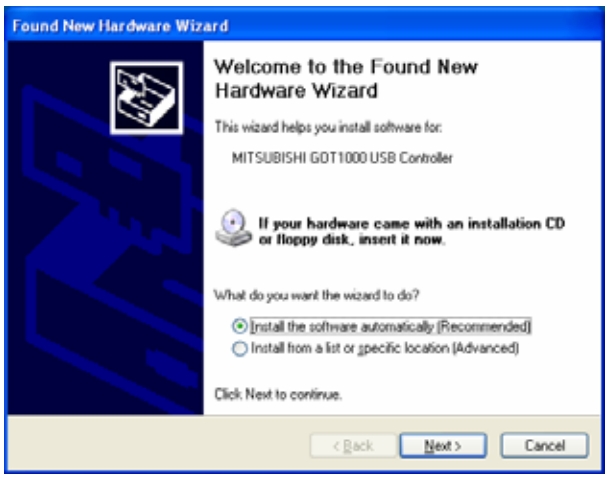
When you connect the GOT and the computer by the USB cable, installation starts and completes automatically.

1.5.5.2. When using Windows XP

The following indicates the procedure for installing the USB driver when using Windows (Professional)

↓

1) The screen shown on the left appears when you connect the personal computer and GOT by the USB cable. Choose "Yes, now and every time I connect a device" and click the [Next] button.

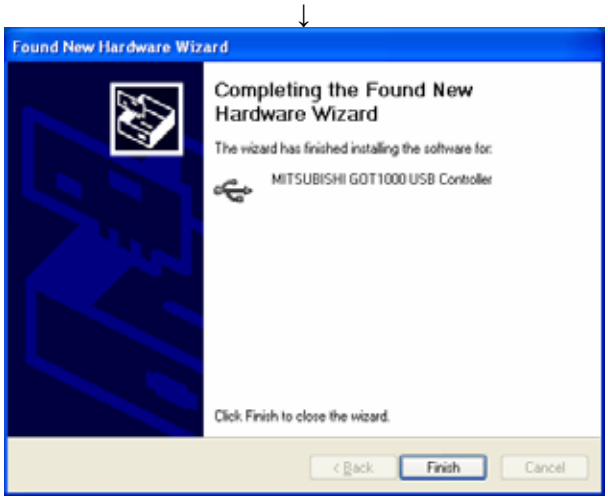


2) As the screen on the left appears, select "Install the software automatically (Recommended)", then click the [Next] button.

↓

3) As the screen on the left appears, click the [Continue Anyway] button to continue the installation of the USB driver.
(No problem will occur after installation of the USB driver.)

↓



4) The screen on the left appears to indicate completion of installation.
Click the [Finish] button to terminate installation.

↓

(Completed)

1.5.5.3. When using Windows Vista

When you connect the GOT and the computer by the USB cable, installation starts and completes automatically.

1.6. When Starting at the Same Time as Another Product

When starting this software and another one of our products at the same time, follow the following precaution. Correct communications and screen display are sometimes not possible.

Table 1-4 Precautions for Starting at the Same Time with Another Product

Product name	Explanation	Precaution
RT ToolBox computer support software	Older version of this software	Can not be used at the same time as this software
MELFA-Works	3D robot simulator	Can not be used at the same time as this software
MELFA-Vision	Network vision sensor software	Start this software first.
E/EN series computer support software for Windows	E/EN series software	Can not be used at the same time as this software
P/P-2	R-250R/R-300R series software	Can be used at the same time as this software

2. RT ToolBox2 Usage

This explains the usage of this software simply.

2.1. Starting RT ToolBox2

When you install this software, a shortcut is prepared on the desktop. Start RT ToolBox2 by double clicking this short cut.

Figure 2-1 RT ToolBox2 Shortcut

From [Start] button -> [All Programs] -> [MELSOFT Applications], select [RT ToolBox2] and start it.

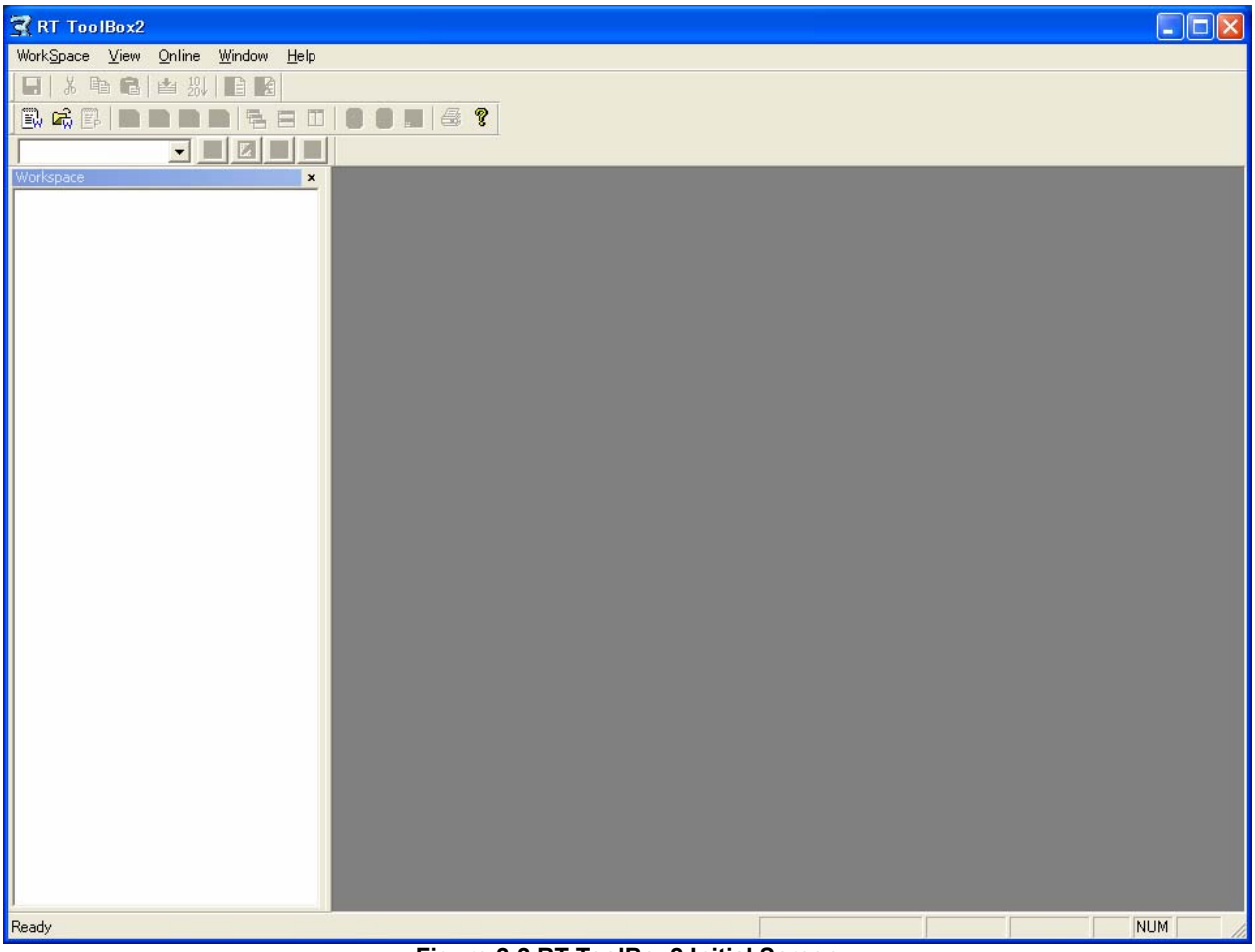


Figure 2-2 RT ToolBox2 Initial Screen

When you start RT ToolBox2, "Communications Server 2" is started up as an icon.
This Communications Server 2 has functions for connecting with a robot controller or during a simulation, a virtual controller. Do not close Communications Server 2.

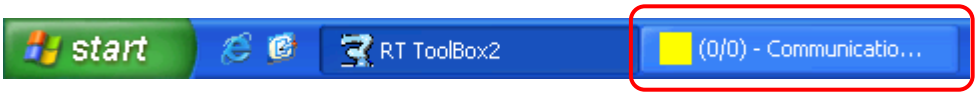


Figure 2-3 Communication Server 2

2.2. Explanation of RT ToolBox2 Screens

The composition of the main RT ToolBox2 screen is as follows.

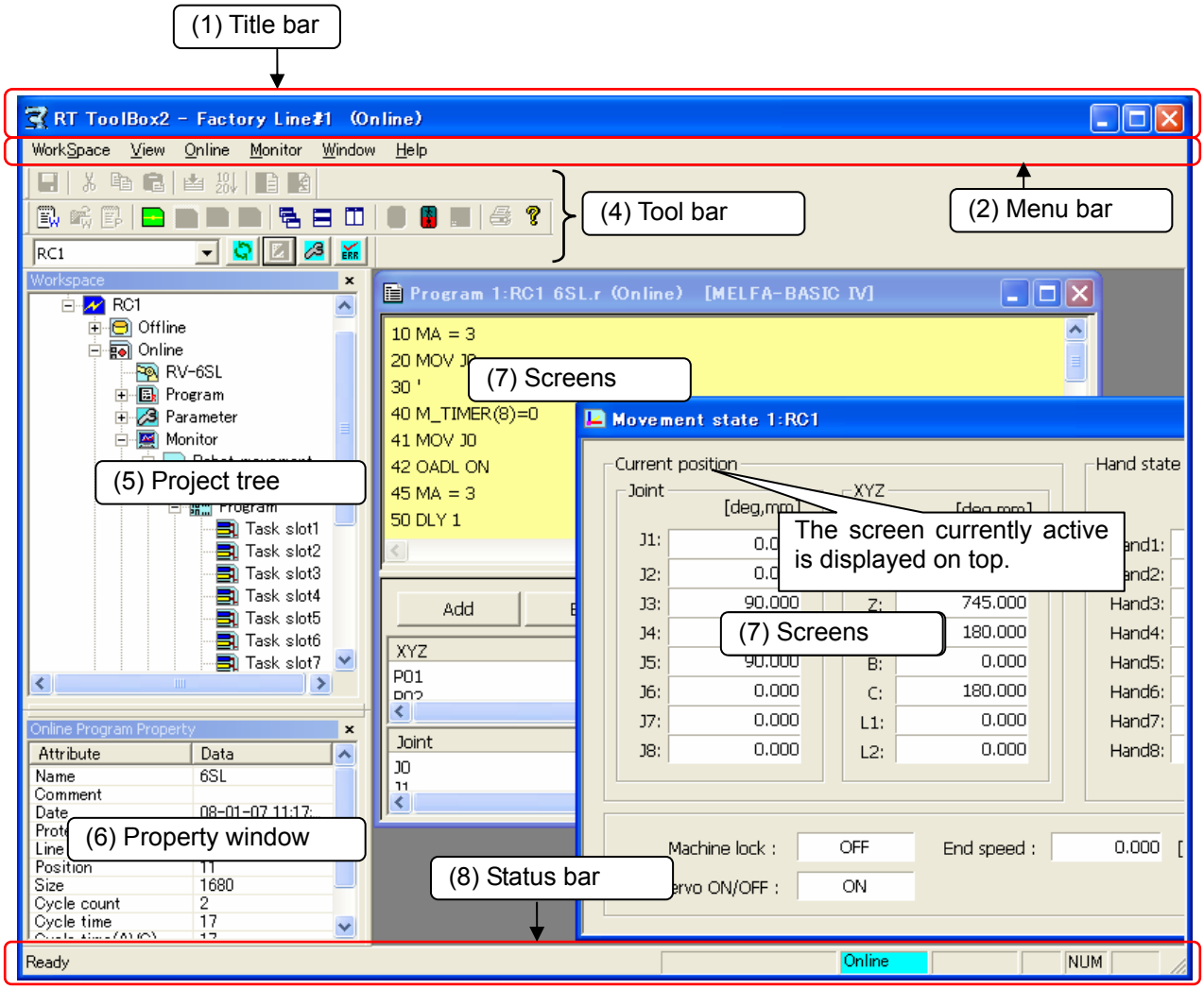
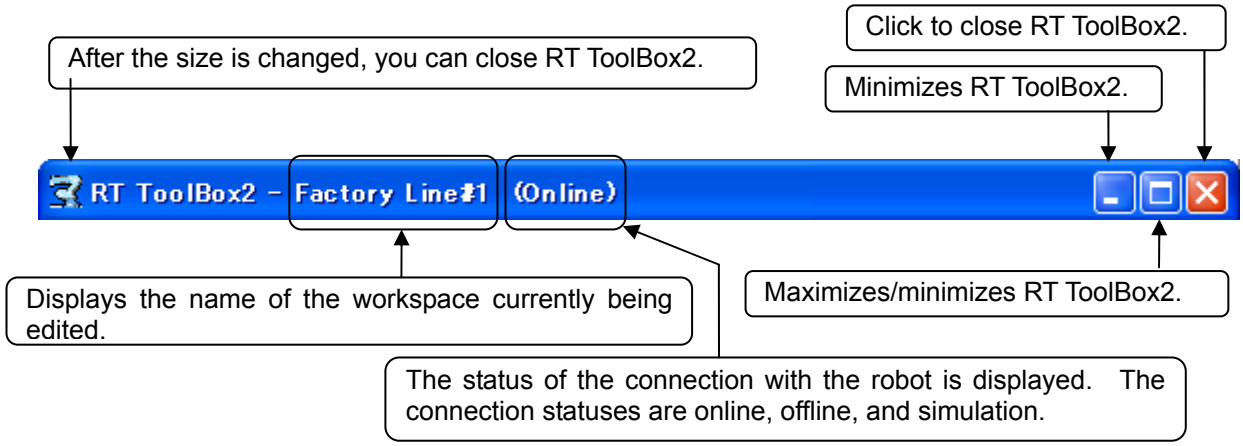


Figure 2-4 Explanation of Main Screen

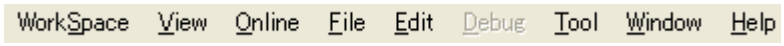
(1) Title bar

Displays the name of the workspace currently being edited.



(2) Menu bar

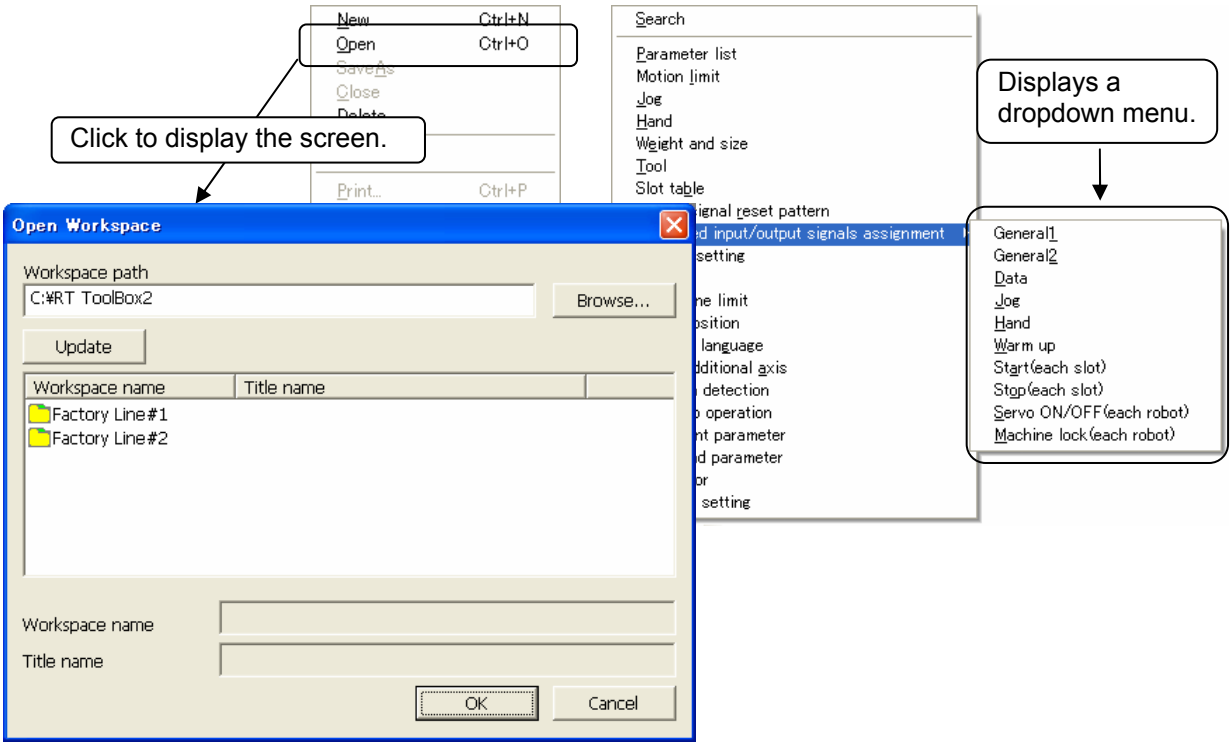
Displays the names of the menus that can be used in RT ToolBox2.
When a menu is selected, a dropdown menu is displayed from which you can use various functions.



The menu bar display contents and their enabled/disabled status depend on which screen is currently active.

(3) Dropdown menu

Displays the names of the functions you can use in RT ToolBox2.
When you click a function name, it displays a screen with the settings etc. for the selected function.
When "▶" is displayed at the right end of a dropdown menu, a dropdown menu for the selected function is displayed



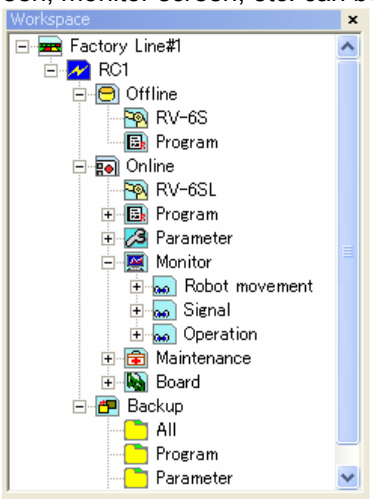
(4) Tool bar

Displays buttons for the functions assigned to the menu bar.

The tool bar display depends on which screen is currently active and on the robot connection status.

(5) Project tree

Displays a list of all the projects registered in the workspace and by functions.
From this tree, the program edit screen, monitor screen, etc. can be started.

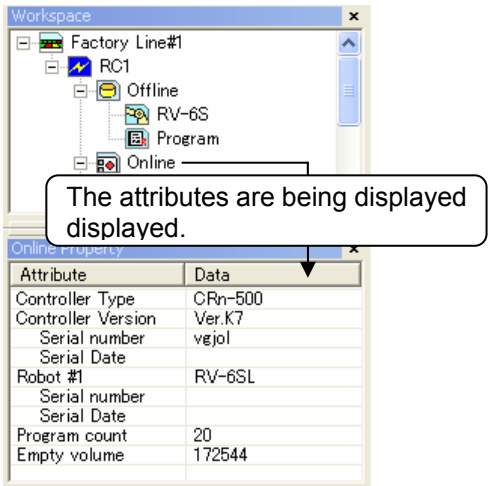


The project tree is a docking window. By dragging the title section with the mouse, you can dock the project tree at the top, bottom, left, or right edge of the main screen.

When the project tree is closed, you can display it again by clicking on the menu bar [View] -> [Project Tree].

(6) Property window

You can reference various attributes of the workspace being edited.
If you click an item on the project tree, its attributes are displayed.

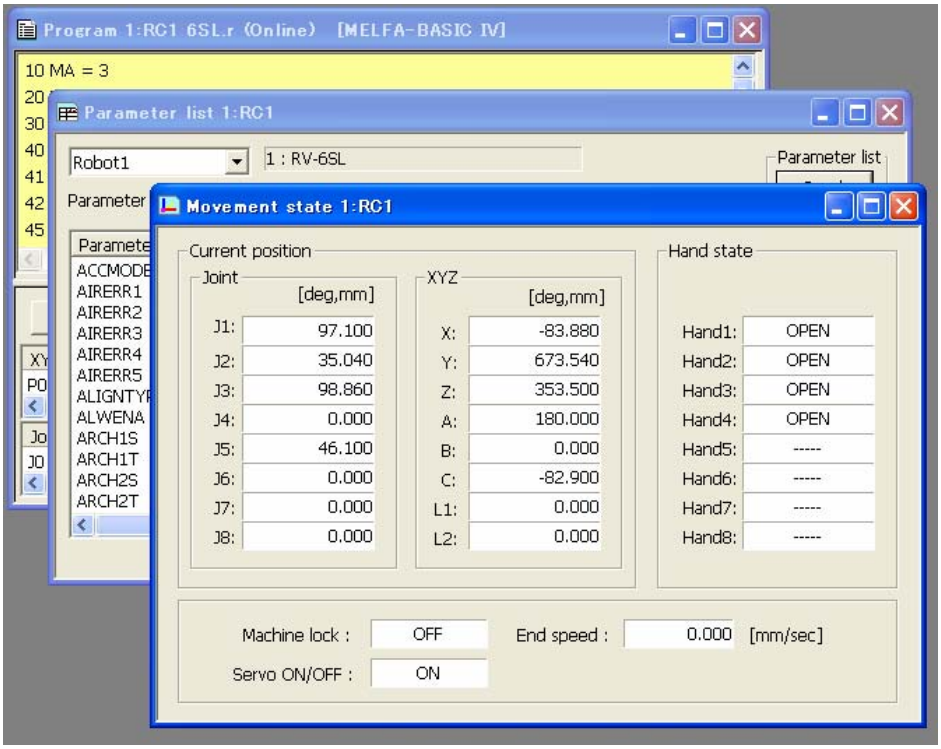


The property window is a docking window. By dragging the title section with the mouse, you can dock the project tree at the top, bottom, left, or right edge of the main screen.

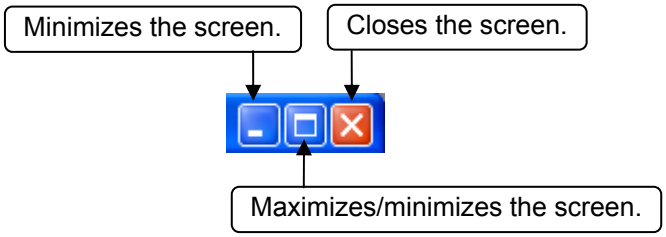
The default setting for the property window is not to be displayed. You can display the property window with the menu bar [View] -> [Property].

(7) Screens

Displays the screens that can be started from the project tree, including the program edit screen and monitor screen. The currently active screen is displayed on top.

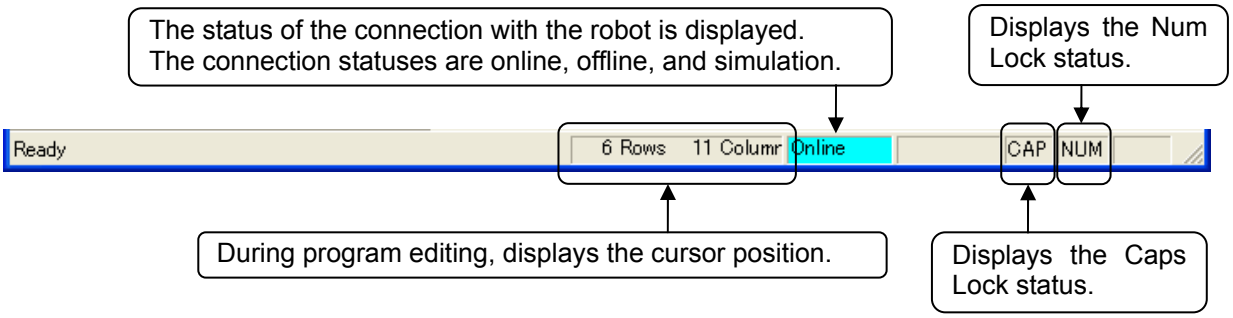


To close a screen click the [ ] at the top right of the screen.
You can also change the screen size with the button at the top right of the screen.



(8) Status bar

Displays RT ToolBox2 status information.



(9) Handy menu display

By clicking the right button on the mouse, you can display the right button menu, depending on the work contents. Using this menu can increase work efficiency.

During program editing, the right button menu includes such functions as [Copy], [Paste], and [Cut].

2.3. Communications Server 2

When you start RT ToolBox2, "Communications Server 2" is started up as an icon.
This Communications Server 2 has functions for connecting with a robot controller or during a simulation, a virtual controller.

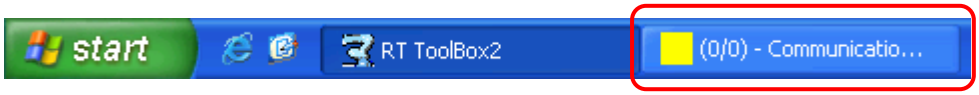


Figure 2-5 Communications Server 2 as Icon

⚠ Caution

Do not close Communication Server 2 manually.

Communication Server 2 has functions for connecting to a robot controller or during simulation, a virtual controller. Do not close Communication Server 2. Communication Server 2 closes automatically when RT ToolBox2 is closed.

Turn Communication Server 2 into an icon.

Turn Communication Server 2 into an icon with the button.

When you return Communications Server 2 from an icon to its original size, you can check the status of the connection with the robot.

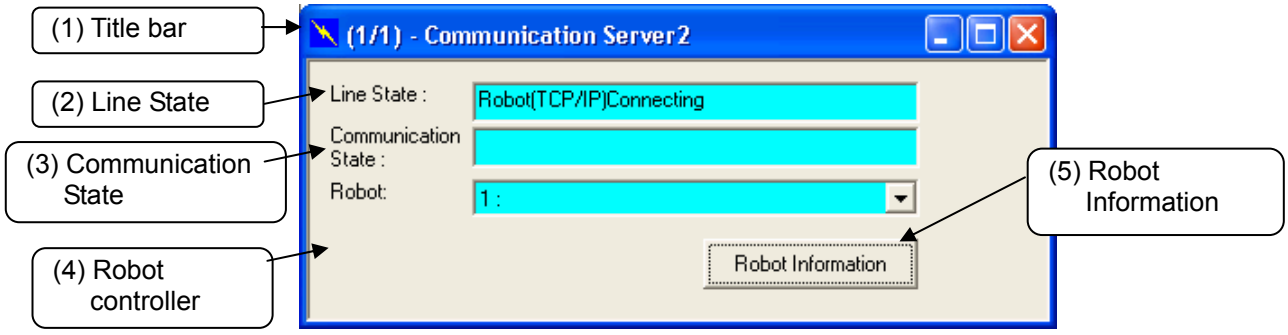


Figure 2-6 Communication Server2

(1) Title bar

(AA/BB) AA: shows the number of robot controllers with which connections are established and BB: shows the number of projects switched online.

(2) Line State

The connection status of the communication line with the robot is displayed. The status color indicates the status of the robot controller that is currently being selected.

Table 2-1 Line State

Status	Content	Color
Connecting	Indicates that the connection with the robot has been established.	Blue
Connection wait	Indicates that a communication to verify connection is being made in the case of RS-232 connection. Indicates the wait status for communication port connection in the case of TCP/IP connection.	Green
Connection error	Displayed when the data reception enable signal cannot be detected because a cable has been disconnected or the robot has not been started in the case of RS-232 connection. Displayed when the communication port cannot be opened in the case of TCP/IP connection.	Red
Communication Setting error	Displayed when the communication port cannot be opened in the case of RS-232 connection. This is not displayed in the case of TCP/IP connection	Red
Waiting	Indicates the idling status displayed at the start of remote maintenance.	Yellow

(3) Communication State

The contents of communication with the robot controller are displayed.

(4) Robot

This changes the robot controller for which the "Line status" and "Communication status" are displayed.
This is only displayed for robot controllers that are online or have been switched to simulation status.

(5) Robot Information

Information on the currently connected robot can be referenced.

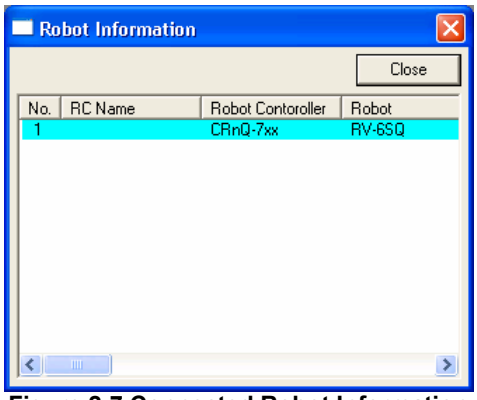


Figure 2-7 Connected Robot Information

Caution

Cautions when connected on USB with CRnQ communications

When connected on USB with CRnQ communications, after connecting normally, if the communications are cut off due to any of the following external causes, the line status display remains "Connected". If this happens, return this software offline, remove the cause of the cutoff, and then go back online.

- The robot controller power supply went off.
- The communication cable was disconnected.

2.4. Closing RT ToolBox2

To close RT ToolBox2, on the menu bar, click [Workspace] -> [Close Application]. You can also close with the [ ] button on the top right of the screen.

When you close RT ToolBox2, Communications Server 2 also closes automatically.

3. Basic Functions

The basic functions of this software are shown below.

Table 3-1 RT ToolBox2 Basic Functions

Function			Explanation
Offline			Targets files on the computer.
	Robot model		Displays the model of the robot used.
	Program		Displays the names of the robot programs written on the computer.
Online			These are used with the robot controller connected.
	Robot model		Displays the model of the robot connected to the robot controller.
	Program		Displays the names of the programs stored on the robot controller.
	Parameters		
		Parameter list	The parameters can be set individually.
		Parameter settings by function	The parameters can be set by robot function individually. The functions are as follows. ・Operating range ・Free plane limit ・Jog ・Evacuation point ・Hand ・Robot language ・Weight and size ・Added axes ・Tool ・Collision detection ・Slot table ・Heater operation ・Output signal reset pattern ・Operation parameters ・Dedicated input/output signal allocation ・Program parameters ・User error ・RS-232 settings ・Ethernet setting ・Zone
	Monitors		
		Operation monitor	You can reference the slot status, program, monitor, operation status, and current errors.
		Signal monitor	You can check the statuses of signals input to the robot and signals output from the robot.
		Work monitor	You can check the robot work time and production information for each robot program.
	Maintenance		
		Home position data	This sets the robot home position.
		Initialization	This sets the robot controller's internal time, deletes all the programs in the controller, initializes the battery time remaining, and sets the serial number for the connected robot.
		Maintenance forecast	Forecasts the time for maintenance.
		Position repair	Supports recovery from home position deviation
		Servo monitor	Monitors servo system information.
	Option card		You can check information on option cards mounted in the robot controller.
Simulation			This can be used connected with a virtual robot. *This can only be used with the standard edition.
	Same functions as online		The same functions can be used with a virtual robot controller as with an online one.
	Tact time measurement		The tact time for the robot program with the specified contents can be measured in the simulation.
Backup, restore			You can back up the information in the robot controller and restore backed up information into the controller.

4. Workspaces and Projects

This explains about workspaces and projects.

4.1. Workspaces and Projects

This software has workspaces and projects.

The information for one controller is managed as one project. A workspace can manage up to 32 projects. If Ethernet is used for communications, you can simultaneously reference information on multiple projects (robot controllers) registered in the workspace.

When using multiple robot controllers, it is convenient to manage with separate workspaces for each manufacturing line and installation location.

(1) Using multiple robot controllers (up to 32)

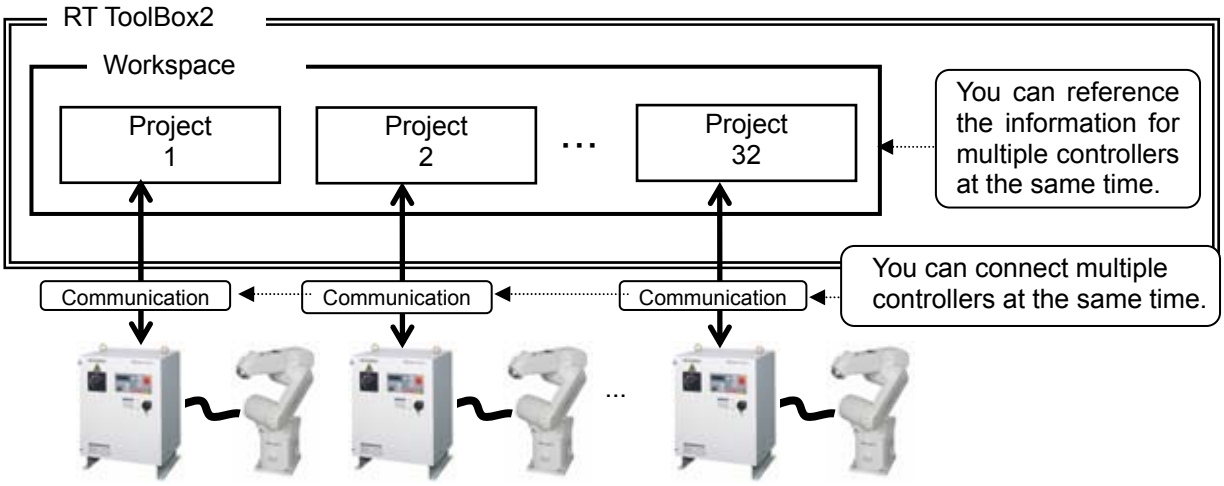


Figure 4-1 Using Multiple Robot Controllers

(2) Using 1 robot controller

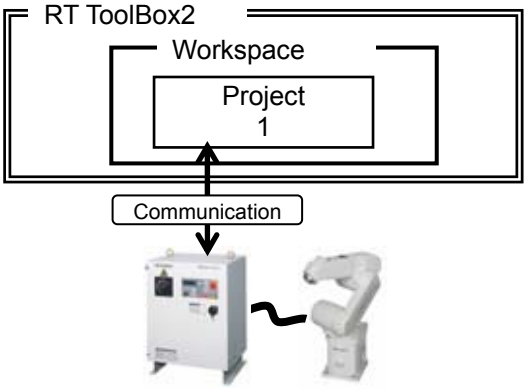


Figure 4-2 Using One Robot Controller

⚠ Caution

You cannot edit multiple workspaces at the same time.

With this software, you cannot edit multiple workspaces at the same time. Register into a single workspace all the projects (controllers) you want to reference at the same time.

Simultaneous connection with 32 units

The maximum of 32 robot connections is a theoretical value. It does not guarantee that if you actually connect 32 robots, you will achieve the same performance as when only 1 robot is connected. For example, if you monitor all the robots when 32 are connected, status updating is slower than for when 1 robot is connected.

Workspace editing and storage and project addition etc. are explained from the next section on.

⚠ Caution

Update workspaces and projects offline.

In the online status or while a simulation is running, you can not perform the following operations. Switch offline, then perform the operations.

- Opening an existing workspace
- Saving the workspace with a different name
- Adding a project
- Changing a project name
- Deleting a project

For details on switching offline, see **"4.15 Offline/Online/Simulation".**

4.2. Creating a New Workspace

Click [Workspace] -> [New Workspace] or click (Ctrl + N).

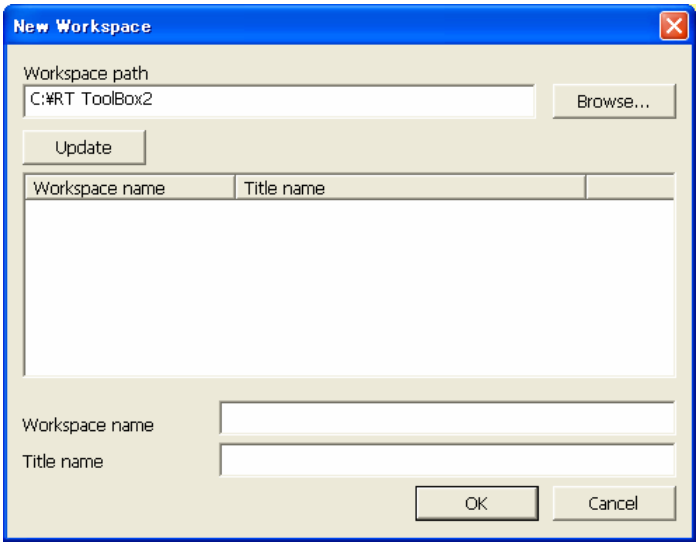


Figure 4-3 Creating a New Workspace

Input the workspace name and title, then click the [OK] button.

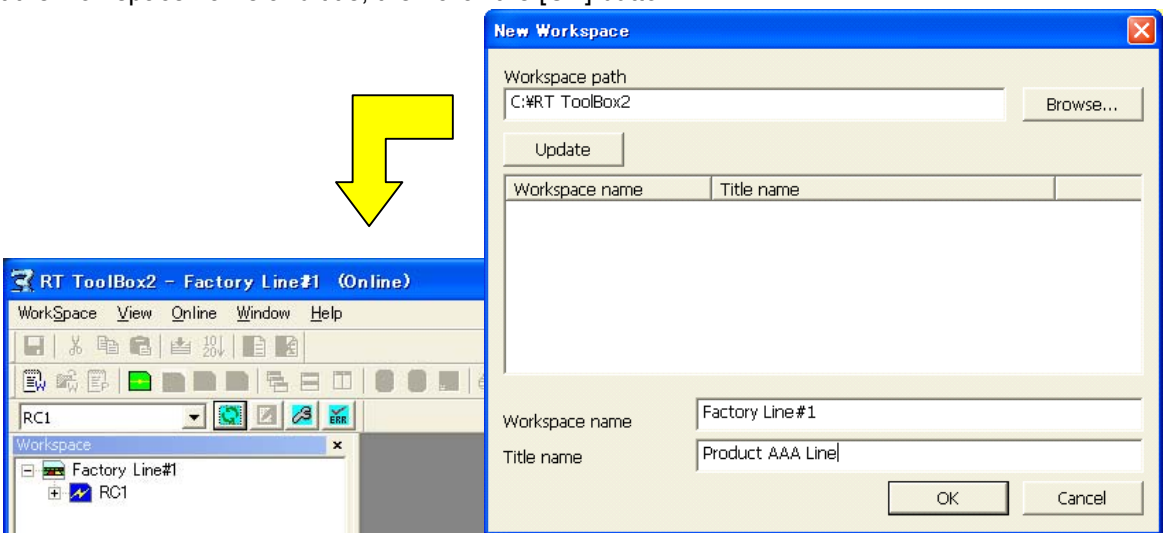


Figure 4-4 Screen After New Workspace Created

The project tree is displayed in the workspace and the "RC1" project is created as the default value. This project name can be change later. For details, see **"4.10 Changing a Project Name".**

⚠ Caution

Workspace names

Workspace names are used as folder names in Windows, so you can not use characters that can not be used in Windows folders names (¥ / : * ? " < > |).

Once a workspace has been created, you can not change its name.

Be aware that once a workspace has been created, you can not change its name.

To change the name of a workspace, select "Workspace"on the menu → "Save As", save with the name you want to change to, then delete the original workspace with the old name.

4.3. Opening an Existing Workspace

To open an already existing workspace, click [Workspace] -> [Open] or click (Ctrl + O). After selecting the workspace to edit, click the [OK] button.

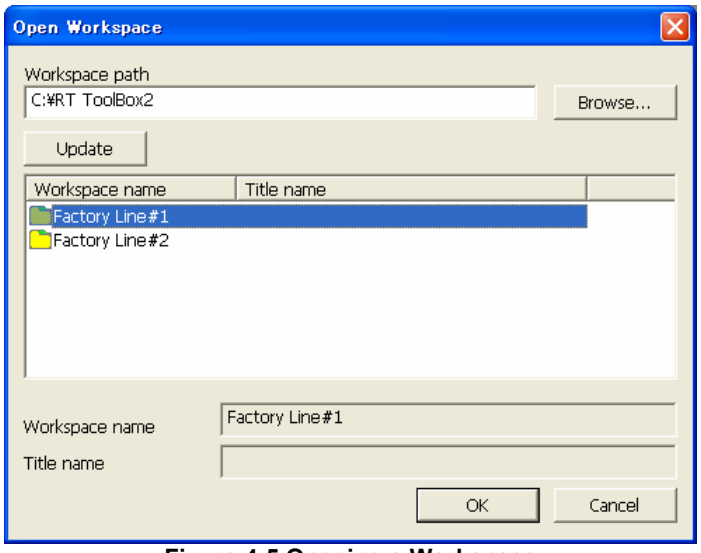


Figure 4-5 Opening a Workspace

The names of the last up to four workspaces used are added to [Workspace] on the menu bar. You can open one of these workspaces by clicking its name here.

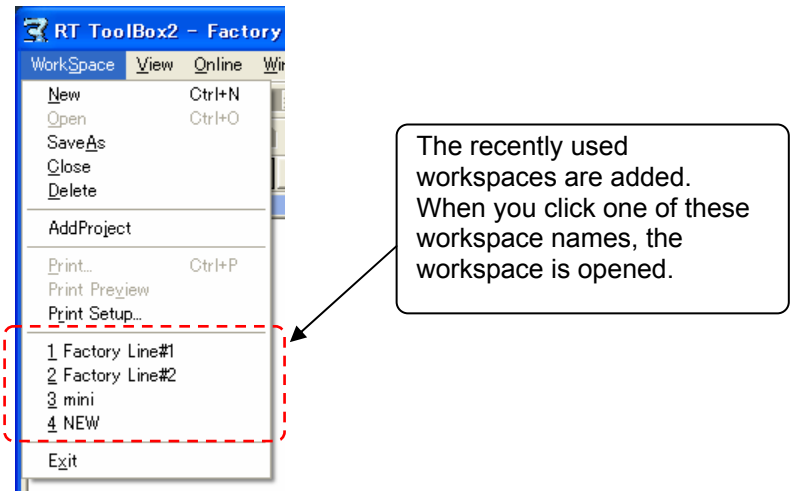


Figure 4-6 Recently Used Workspace Names

4.4. Closing a Workspace

To close the workspace being edited, click [Workspace] -> [Close]. Note that to delete a workspce, you must close it first.

4.5. Deleting a Workspace

Click [Workspace] -> [Delete]. The "Delete Workspace" screen is displayed, so check the name of the workspace to delete, then click the [OK] button.

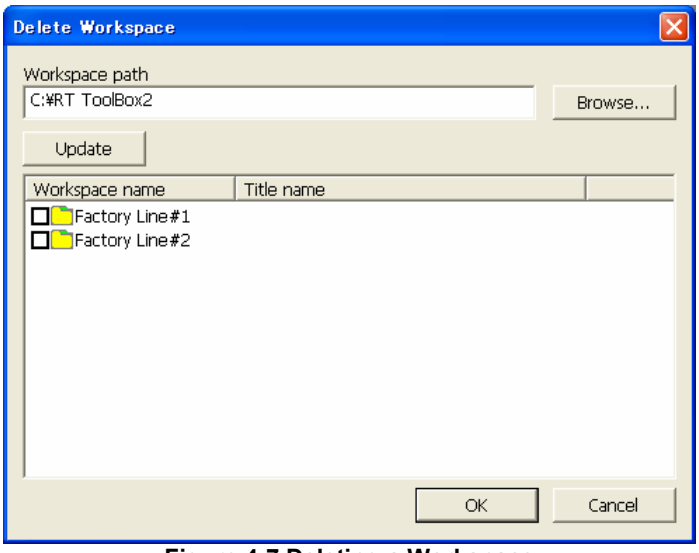


Figure 4-7 Deleting a Workspace

Caution

Deleting a workspace

To delete a workspace, close that workspace. You can not delete a workspace that is being edited.

4.6. Saving a Workspace

The information added to the workspace is saved automatically.

You can also save a workspace with a different name. Click [Workspace] -> [Save As], input the new workspace name and title, then click the [OK] button.

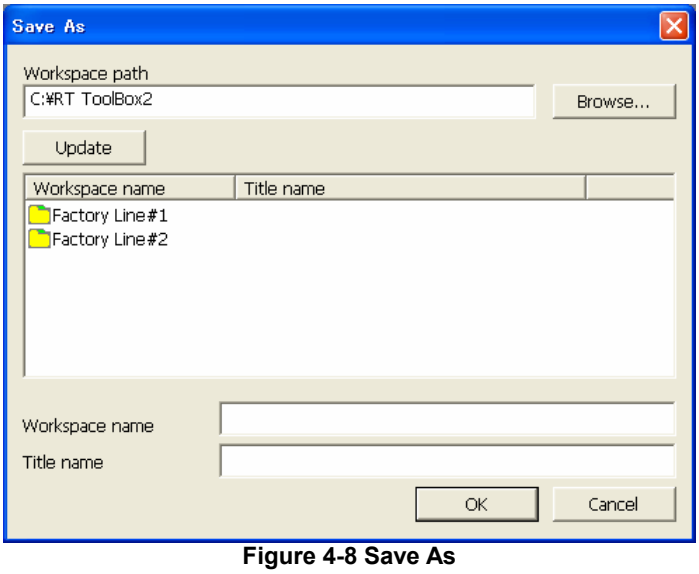


Figure 4-8 Save As

4.7. Changing a Workspace Name

Once a workspace has been created, you can not directly change its name. To change a workspace name, use the menu bar [Workspace] -> [Save As] to save the workspace with the desired new name, then delete the old workspace.

4.8. Changing a Workspace Title

You can change the title of a workspace even after it has been created.
Click the name of the workspace with the right button on the mouse, then click "Edit workspace title" on the right button menu. The workspace title screen is displayed, so input the new workspace title, then click the [OK] button.

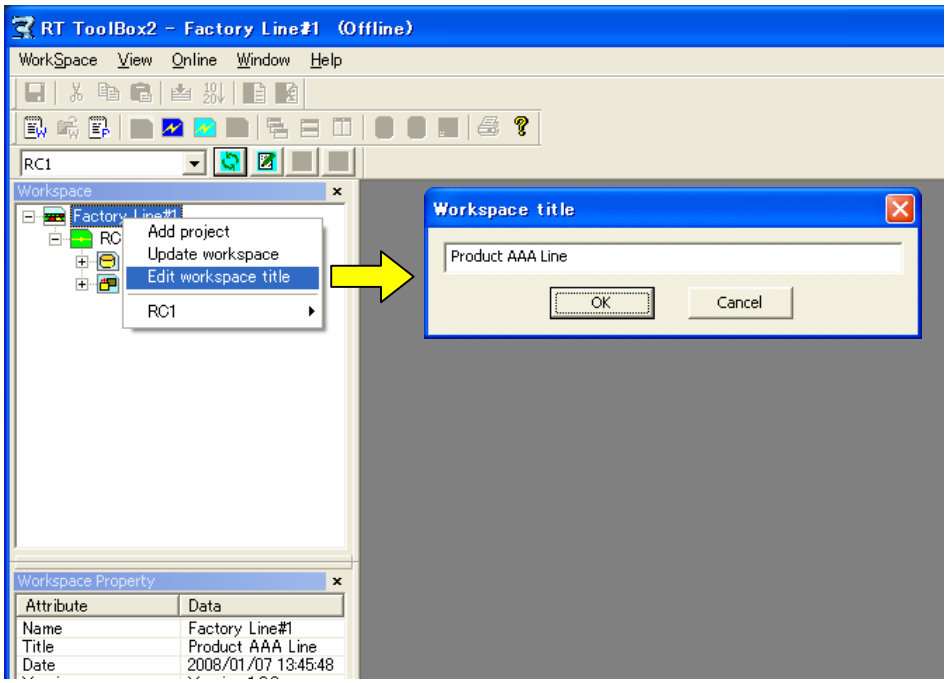


Figure 4-9 Changing a Title

4.9. Adding a Project

You can create up to 32 projects in one workspace.
Click [Workspace] -> [Add Project].

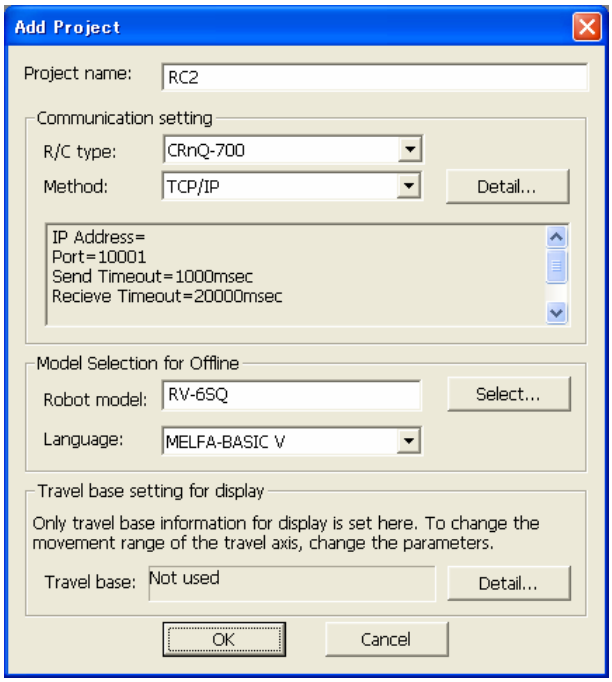


Figure 4-10 Adding a Project

Set the project name, information on the robot connected, and the communication method, then click the [OK] button.
The project is added to the workspace.

Projects can also be added from a project tree.
After selecting the workspace, use the mouse right button to select [Add project].

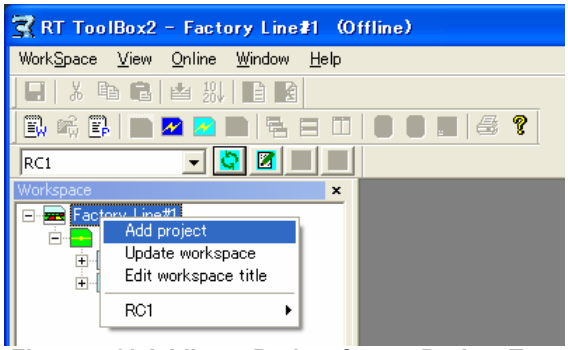


Figure 4-11 Adding a Project from a Project Tree

4.10. Changing a Project Name

You can change the name of a project even after it has been created.
Click the name of the project with the right button on the mouse, then click "Edit Project" on the right button menu. The project edit screen is displayed, so input the new workspace title, then click the [OK] button.

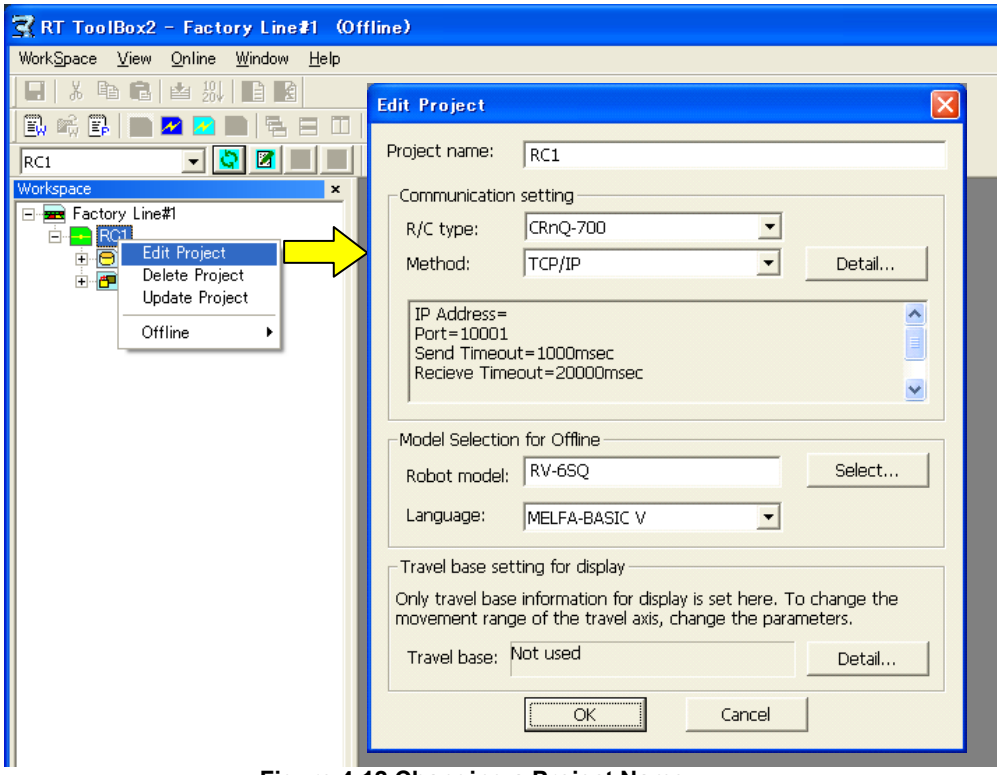


Figure 4-12 Changing a Project Name

4.11. Deleting a Project

A project can not be deleted when it is running a simulation or has a robot connected online.
Click the name of the project with the right button on the mouse, then click "Delete Project" on the right button menu. The deletion confirmation message is displayed, so check that you have clicked the correct project name, then click the [OK] button.

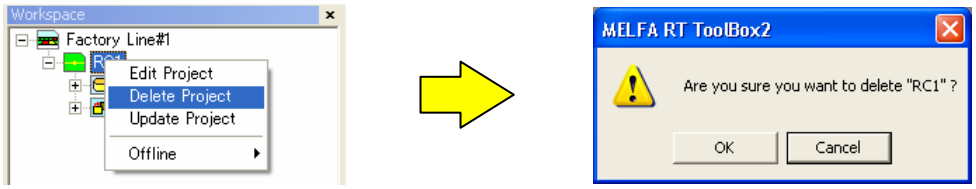


Figure 4-13 Deleting a Project

4.12. Contents of Project Tree

A project tree shows the current workspace's project configuration in a hierarchical manner. You can start all functions from the project tree, including program editing, monitor, etc.

The contents of the project tree depend on the state of connection with the robot controller.

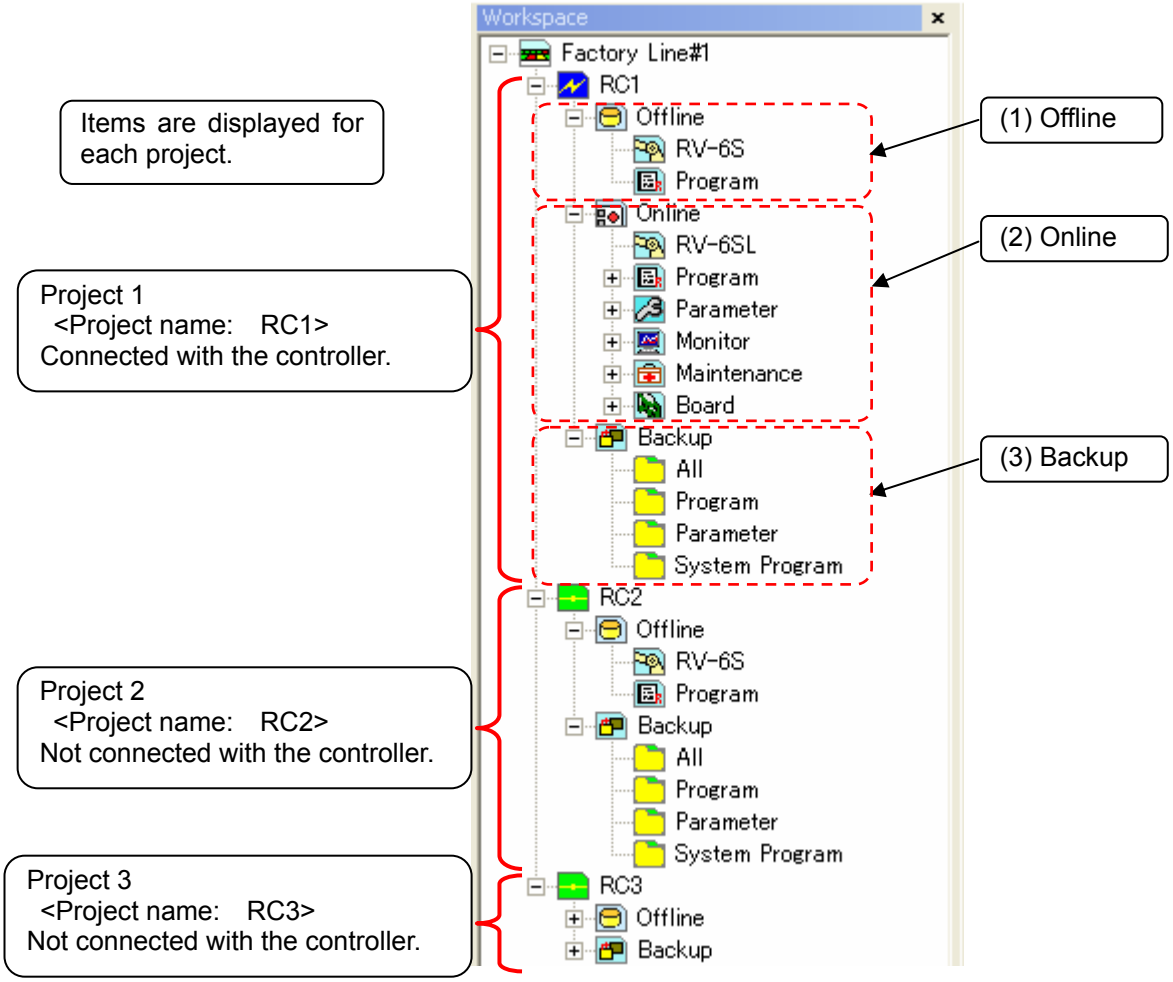


Figure 4-14 Structure of Project Tree in Workspace

(1) Offline

Displays the information stored in computer.
Displays the currently set robot model and the name of robot programs created.

(2) Online

The "Online" section is displayed when a robot is switched into being connected online with its controller or starts a simulation. This section displays the connected robot model and the information items that can be referenced in the controller or a simulation controller.

(3) Backup

Controls the information backed up from the controller.

4.13. Copying Programs Between Projects

To copy or move a created robot program to another project, do this with program management. For details on the operation methods, see **"7.10 Program Management"**, **"7.10.2 Copy"**, or **"7.10.3 Move".**

4.14. Import of project

It is possible that the project in other workspace is imported to the current workspace. Note that this function can be used with Version 1.2 or later of this software.

The project can be imported only at offline. Please operate after change to offline when current mode is online or simulation.

Click [Work space]->[Import project] on the menu bar.
After "Select Workspace" window is displayed, select the workspace including the project that you want to import, and click [OK] button. The list of the project in selected workspace is displayed on "Select the projects" window. Check the project, and click [OK] button.

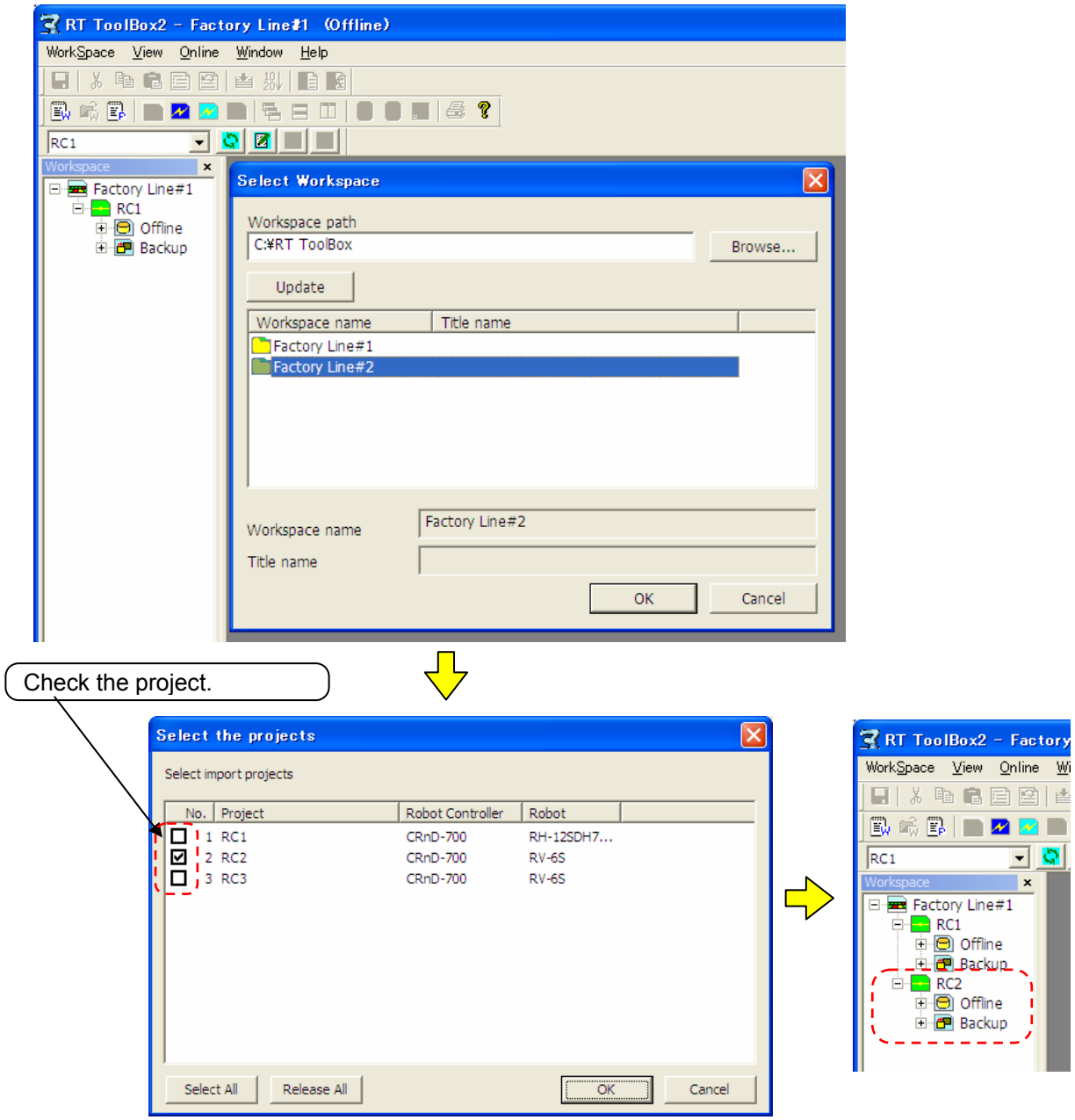


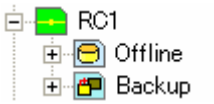
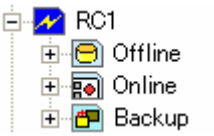
Figure 4-15 Import of project

The maximum number of the projects in one workspace is 32.

4.15. Offline/Online/Simulation

The project status are offline, online, and simulation. The meaning of each status and the contents displayed in the project tree are shown below.

Table 4-1 RT ToolBox2 Status

Status	Explanation	Project tree display
Offline	Targets files stored on the computer. When a robot is offline, the icon displayed on the left of the project name on the project tree turns green and Offline and Backup are displayed on the project tree.	
Online	The robot is connected to the robot controller and you can check and change the information in the controller When a robot is online, the icon displayed on the left of the project name on the project tree turns blue and Offline, Online, and Backup are displayed on the project tree.	
Simulation *This can only be used with the standard edition.	This targets a virtual robot controller running on the computer and you can check and change the information in the virtual controller When a simulation is running, the icon displayed on the left of the project name on the project tree turns blue and Offline, Online, and Backup are displayed on the project tree.	

When this software starts, it goes into "Offline" status.

Caution

The simulation function only supports the "RT ToolBox2" standard edition. It can not be used with the mini edition.

Also, please be aware that even with the standard edition, the simulation function can not be used when Movemaster commands are selected.

To switch among offline, online, and simulation, do so from [Online] on the menu bar or from the tool bar. With the tool bar, you can switch as follows.

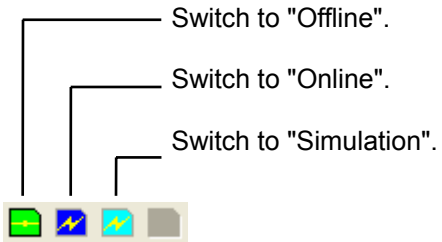


Figure 4-16 Explanation of Tool Bar

The current status and the statuses into which the current status can be switched are as followed.

Table 4-2 Current Status and Switchable Status

Current status	Switchable status	Menu bar [Online] display	Tool bar display
Offline	Can be switched to online or simulation.		
Online	Can only be switched to offline.		
Simulation	Can only be switched to offline.		

The current status is displayed checked at [Online] on the menu bar.

When switching to online or simulation, if multiple projects are registered in the workspace being edited, the project selection screen is displayed as in **"Figure 4-17 Screen for Selecting Project to Switch Online"**. Check only the project to be switched online or to simulation, then click [OK]. Only one project can be switched to simulation.

This screen is not displayed if there is just one project in the workspace.

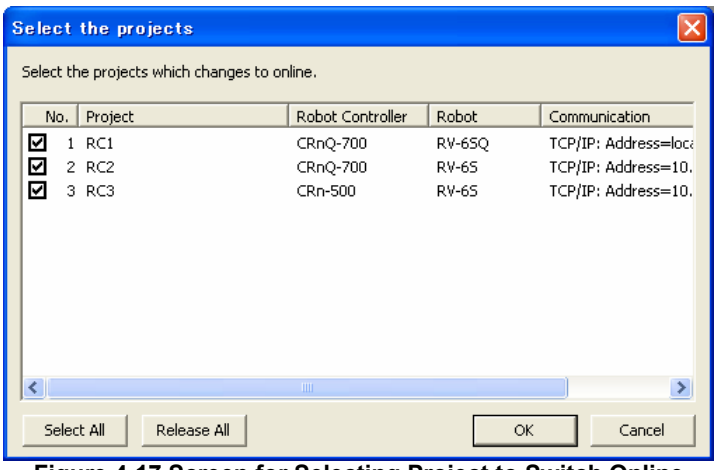


Figure 4-17 Screen for Selecting Project to Switch Online

Caution

When using Ethernet, do not set an unconnected controller "Online".

When using Ethernet, if you try to switch "**Online**" a robot controller that is not connected, incessant retries are made, which slows execution. Set **"Offline"** a robot controller that is not connected or whose power is Off.

Do not connect from multiple projects to 1 controller at the same time.

Do not connect from multiple projects to 1 controller at the same time.
Data might not be able to be read correctly referring to the same data.

Do not connect USB to multiple controllers from 1 computer.

When using USB connection, 1 computer can connect to only 1 robot controller. It is also not possible to connect to multiple controllers by using the USB hub.

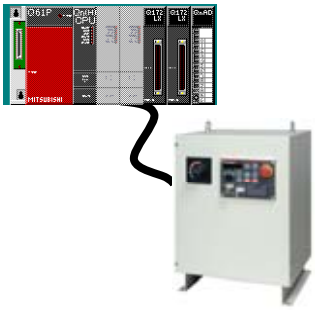
5. Connecting with the Robot

This explains the method for connecting the robot controller and the computer.
With this software, you must make the communication settings for each project.

5.1. Robots Connected and Types of Communication

When connecting the robot controller and computer, there are the methods shown below. Be aware that the connection method that can be used is restricted by the robot controller connected.

Table 5-1 Robot Controllers That Can Be Connected and Types of Communication

Robot Controller	Communication Type		Remarks
CRnD-700 series 	USB (*4)		
	Ethernet(TCP/IP) (*1)		
	RS-232		
CRnQ-700 series 	CRnQ communications (*2)	USB	
		Ethernet (TCP/IP)	The PLC Ethernet interface module or Built-in Ethernet port QCPU (*5) is required.
		RS-232	
CRn-500 series 	Ethernet(TCP/IP) (*3)		The robot controller must have the "Ethernet interface" option.
	RS-232		

(*1) For details on the communication settings on the robot controller side for CRn-700 series controllers, see "**Operations and Detailed Explanation**" and "**Standard Specifications**" in the user's manual for the robot controller.
(*2) For details on the communication settings on the robot controller side for CRnQ-700 series controllers, see "**Operations and Detailed Explanation**" and "**Standard Specifications**" in the user's manual for the robot controller.
Moreover, please use the “GX Developer” to set the communication setting of PLC.
(*3) For details on Ethernet connections on the robot controller side for CRn-500 series controllers, see the "**Ethernet Interface Option User's Manual**".
(*4) When using USB connection, 1 computer can connect to only 1 robot controller.
(*5) Built-in Ethernet port QCPU can be used with this software Ver.1.2 or later.

5.2. Connection Settings

Click the name of the project to display the project tree for with the right button of the mouse, then from the right button menu, click [Edit Project]. The project edit screen is displayed.

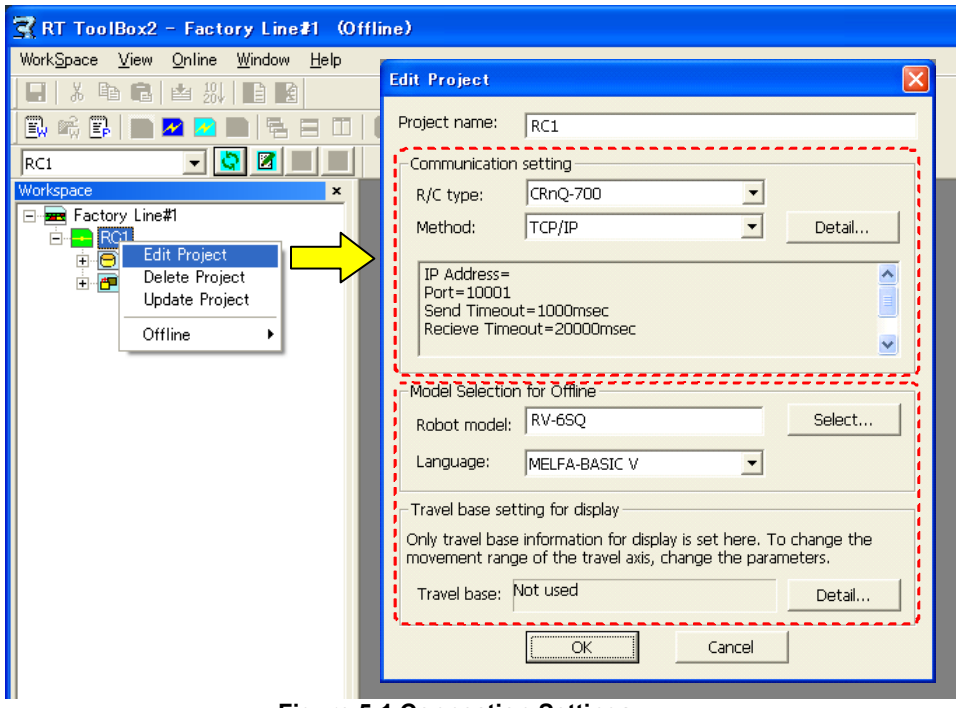


Figure 5-1 Connection Settings

Select the controller to connect to and the communications method, then click the "Detailed Settings" button to set the details.
Normally, the offline robot settings and the travel platform settings for display are also set according to the controller connected to.

After you complete the details settings, click the [OK] button.

The detailed settings screen will be explained from the next item.

⚠ Caution

Do not connect from multiple projects to 1 controller at the same time.

Do not connect from multiple projects to 1 controller at the same time.
Data might not be able to be read correctly referring to the same data.

Do not connect USB to multiple controllers from 1 computer.

When using USB connection, 1 computer can connect to only 1 robot controller. It is also not possible to connect to multiple controllers by using the USB hub.

5.2.1. USB Communication Settings

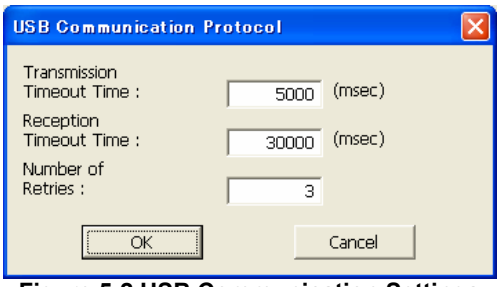


Figure 5-2 USB Communication Settings

Table 5-2 TCP/IPCommunication Setting Items and Default values

Item	Explanation	Default value	
		Ver.1.1 or earlier	Ver.1.2 or later
Send timeout	Timeout time for sending Range that can be set (*1) Minimum value : 1000 msec Maximum value : 30000 msec	3000 msec	5000 msec
Receive timeout	Timeout time for receiving Range that can be set (*1) Minimum value : 5000 msec Maximum value :120000 msec	3000 msec	30000 msec
Retries	Number of communication retries Range that can be set (*1) Minimum value : 0 Maximum value : 10	3	3

(*1) The setting range of Send timeout, Receive timeout and Retries are limited function of this software Ver.1.2 or later.

When the USB cable is connected to the computer, the screen for installing the USB driver may be displayed. For details on USB driver installation, see **"1.5.3 USB driver (CRnD-700 series robot controller) installation "**.

5.2.2. TCP/IP (Ethernet) Communication Settings

Input the IP address assigned to the robot controller connected to, then click the [OK] button.

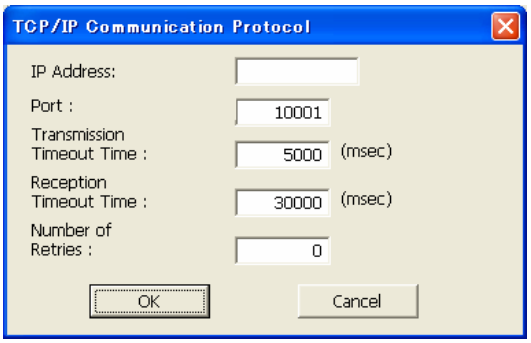


Figure 5-3 TCP/IP Communication Settings

Table 5-3 TCP/IP Communication Setting Items and Default Values

Item	Explanation	Default value	
		Ver.1.1 or earlier	Ver.1.2 or later
IP address	Sets the IP address of the robot controller connected to.	Blank	
Port used	Number of the port used for communications	10001	
Send timeout	Timeout time for sending Range that can be set (*1) Minimum value : 1000 msec Maximum value : 30000 msec	1000 msec	5000 msec
Receive timeout	Timeout time for receiving Range that can be set (*1) Minimum value : 5000 msec Maximum value :120000 msec	20000 msec	30000 msec
Retries	Number of communication retries Range that can be set (*1) Minimum value : 0 Maximum value : 10	3	0

(*1) The setting range of Send timeout, Receive timeout and Retries are limited function of this software Ver.1.2 or later.

For help with the computer side network settings (IP address, subnet mask, default gateway, etc.), ask the network administrator.
Moreover, for details on the communication settings on the robot controller side, please refer as follow.

CRn-700 series controllers : See **"Operations and Detailed Explanation"** and **"Standard Specifications"** in the user's manual for the robot controller.
CRn-500 series controllers : See the **"Ethernet Interface Option User's Manual"**.

⚠ Caution

When connecting to 10 or more robot controllers at the same time.

When connecting to 10 or more robot controllers at the same time, set the reception timeout time to at least 10000 msec, because it takes time to connected processing.

⚠ Caution

Caution for connecting with robot controller by Ethernet.

In the program edit function, please change "Receive timeout" and "Retries" of TCP/IP communication setting when it corresponds to all of the following use conditions, and the program is written to robot controller.

<< Use conditions >>
1. It connects with CRn-500/CRnQ-700/CRnD-700 controller by using Ethernet .
   (* In CRn-500 controller, Ethernet is an option.)
2. When you write only command line in the controller.
   (Only "Command lines" is selected when the program is written to robot controller.)
3. The number of lines of the command line of the robot program is 1600 or more.
4. In TCP/IP Communication Settings, the both value of "Receive timeout" and "Retries" are numbers except 0.

⇩

item	Value
Receive timeout (msec)	30000 or more
Retries	0

When using it on the condition of corresponding without changing the communication setting, the robot program might not be correctly written or the error might occur while writing the robot program. Therefore, it is necessary to check all content of the robot program. So please change the communication setting.

5.2.3. RS-232 Communication Settings

Change the RS-232 communication settings to match the robot controller side communications settings.

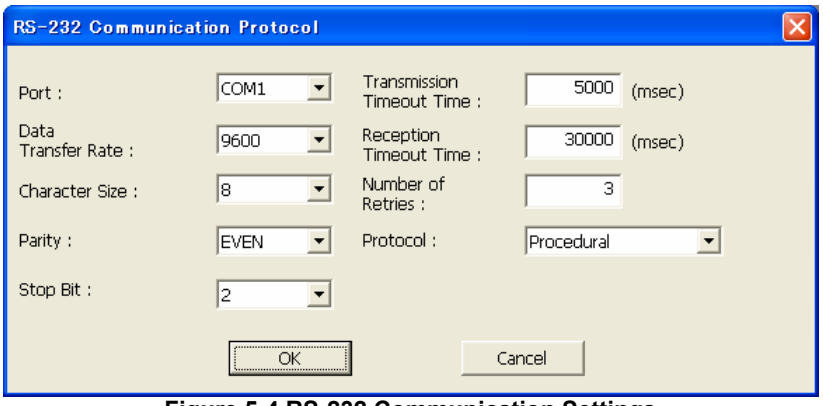


Figure 5-4 RS-232 Communication Settings

Table 5-4 RS-232 Communication Setting Items and Default Values

Item	Explanation	Default value
Port used	COM1 – COM10 can be selected.	COM1
Communications speed	Select from 4800, 9600, and 19200. + It is possible to select Baud rate 38400 only when connecting this software with CRnD-700 controller.	9600
Character size	7 or 8 can be selected, but select 8.	8
Parity	None, Odd, or Even can be selected.	EVEN
Stop bit	Select from 1, 1.5, and 2.	2
Send timeout	Timeout time for sending Range that can be set (*1) Minimum value : 1000 msec Maximum value : 30000 msec	5000 msec
Receive timeout	Timeout time for receiving Range that can be set (*1) Minimum value : 5000 msec Maximum value : 120000 msec	30000 msec
Retries	Number of communication retries Range that can be set (*1) Minimum value : 0 Maximum value : 10	3
Usage protocol	Non-Procedural or Procedural can be selected.	Procedural

(*1) The setting range of Send timeout, Receive timeout and Retries are limited function of this software Ver.1.2 or later.

Set the following to perform a high-speed, stable communication.

Baud rate : 19200 bps
Protocol used : Procedural

It is also necessary to change the communication settings to the same value of the robot controllers at this time.

5.2.4. CRnQ Communications Settings

The CRnQ communication is the method to connect with the CRnQ-700 series controller by using the PLC Universal model QCPU module or the PLC Ethernet Interface module.

When using RS-232 or USB, please connect to connector of the PLC Universal model QCPU module.

When using Ethernet, please connect to connector of the PLC Ethernet Interface module or Built-in Ethernet port QCPU. (*Built-in Ethernet port QCPU can be used with this software Ver.1.2 or later.)

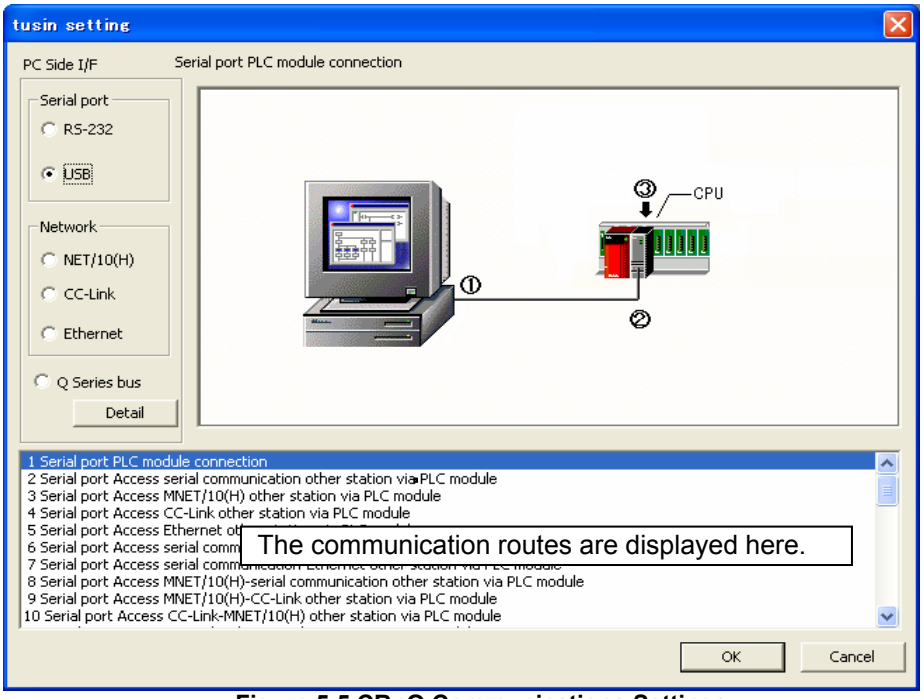


Figure 5-5 CRnQ Communications Settings

On the CRnQ communications setting screen, when you select the computer interface, the communications routes using the selected interface are displayed at the bottom of the screen. Select the communications route, then press the [Detailed Settings] button. The detailed settings screen corresponding to the selected communications route is displayed. These detailed settings will be explained from the next item.

5.2.4.1. When RS-232 is selected

When "RS-232" is selected as the computer interface, the communications routes that can be selected are as follows.

Table 5-5 Communications Routes That Can Be Selected for "RS-232"

No.	Communications routes
1	Serial communications CPU unit connection
2	Serial communications Serial communications other station access through CPU unit
3	Serial communication NET/-0(H) other station access through CPU unit
4	Serial communication CC-Link other station access through CPU unit
5	Serial communication Ethernet other station access through CPU unit
6	Serial communication Serial communication through CPU unit - NET/10(H) other station access
7	Serial communication Serial communication through CPU unit-Ethernet other station access
8	Serial communication NET/10(H) through CPU unit - serial communication other station access
9	Serial communication NET/10(H) through CPU unit - CCLink other station access
10	Serial communication Other station access CCLink - NET/10(H) through CPU unit
11	Serial communication CCLink through CPU unit - Ethernet other station access
12	Serial communication Ethernet through CPU unit - serial communication other station access
13	Serial communication Ethernet through CPU unit - CCLink other station access
14	Serial communication Serial communication connection
15	Serial communication Other station access serial communication through serial communication
16	Serial communication Other station access NET/10(H) through serial communication
17	Serial communication Other station access CC-Link through serial communication
18	Serial communication Other station access Ethernet communication through serial communication

No.	Communications routes
19	Serial communication Serial communication through serial communication - NET/10(H) other station access
20	Serial communication Serial communication through serial communication - Ethernet other station access
21	Serial communication NET/10(H) through serial communication - serial communication other station access
22	Serial communication NET/10(H) through serial communication - CCLink other station access
23	Serial communication CCLink through serial communication - NET/10(H) other station access
24	Serial communication CCLink through serial communication - Ethernet other station access
25	Serial communication Ethernet through serial communication - serial communication other station access
26	Serial communication Ethernet through serial communication - CCLink other station access
27	Serial communication CC-Link other station access through G4
28	Serial communication CCLink through G4 - NET/10(H) other station access
29	Serial communication CC-Link through G4 – Ethernet other station access
30	Serial communication Serial communication other station access through NET/10(H) remote
31	Serial communication NET/10(H) other station access through NET/10(H) remote
32	Serial communication CC-Link other station access through NET/10(H) remote
33	Serial communication Ethernet other station access through NET/10(H) remote
34	Serial communication Serial communication other station access through NET/10(H) remote
35	Serial communication Serial communication through NET/10(H) remote - Ethernet other station access
36	Serial communication NET/10(H) through NET/10(H) remote - Serial communication other station access
37	Serial communication NET/10(H) through NET/10(H) remote - CC-Link other station access
38	Serial communication CC-Link through NET/10(H) remote - NET/10(H) other station access
39	Serial communication CC-Link through NET/10(H) remote - Ethernet other station access
40	Serial communication Ethernet through NET/10(H) remote - Serial communication other station access
41	Serial communication Ethernet through NET/10(H) remote - CC-Link other station access

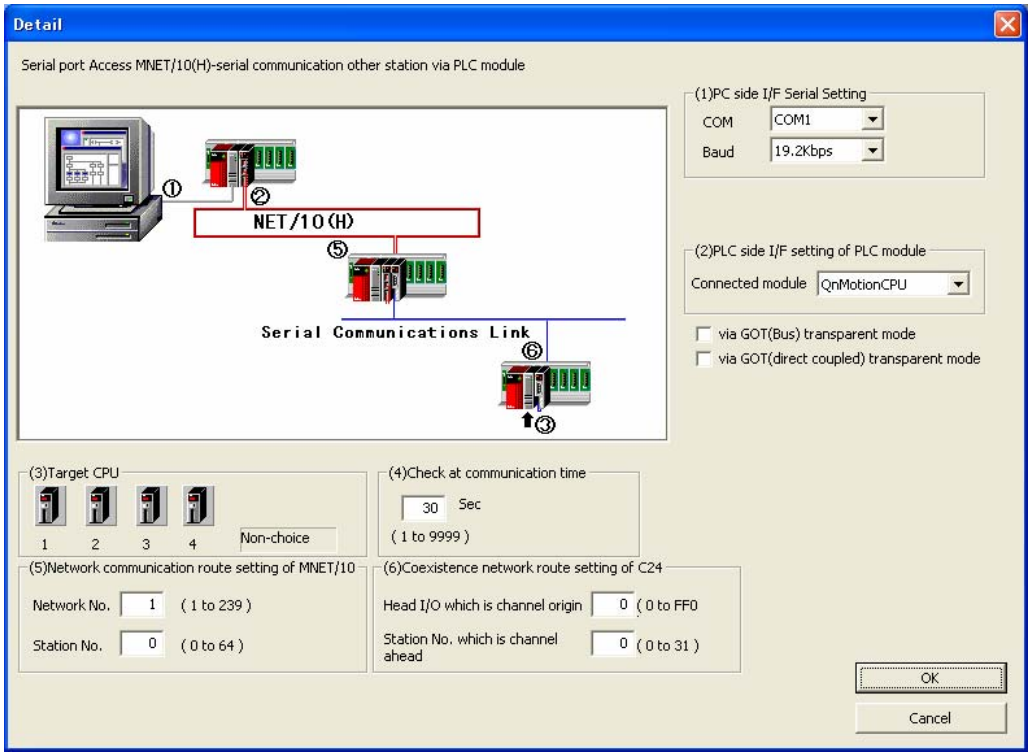


Figure 5-6 RS-232 (Serial Communication NET/10(H) through CPU Unit – CCLink Other Station Access) Detailed Settings

For some communications routes that you can select, "(5)Network communication route setting of MNET/10" and "(6)Coexistence network route setting of C24" can not be set.

5.2.4.2. When USB is selected

When "USB" is selected as the computer interface, the communications routes that you can select are the same as for RS-232. See **"Table 5-5 Communications Routes That Can Be Selected for** "RS-232"**".**

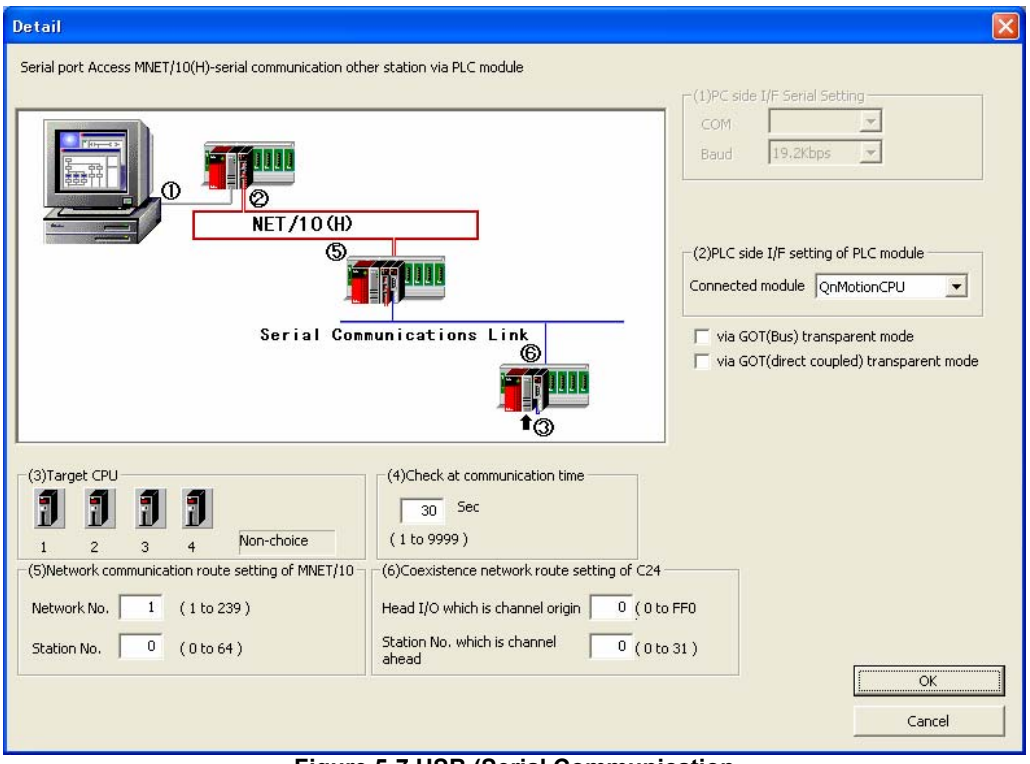


Figure 5-7 USB (Serial Communication NET/10(H) through CPU Unit – Serial Communication Other Station Access) Detailed Settings

For some communications routes that you can select, "(5)Network communication route setting of MNET/10" and "(6)Coexistence network route setting of C24" can not be set.

5.2.4.3. When NET/10(H) is selected

When "NET/10" is selected as the computer interface, the communications routes that you can select are as follows.

Table 5-6 Communications Routes That Can Be Selected for "NET/10(H)"

No.	Communications Route
1	NET/10 board communication Other station access through NET/10(H) unit
2	NET/10 board communication NET/10(H) through NET/10(H) unit – CC-Link other station access
3	NET/10(H) board communication NET/10(H) through NET/10(H) unit - Serial communication other station access

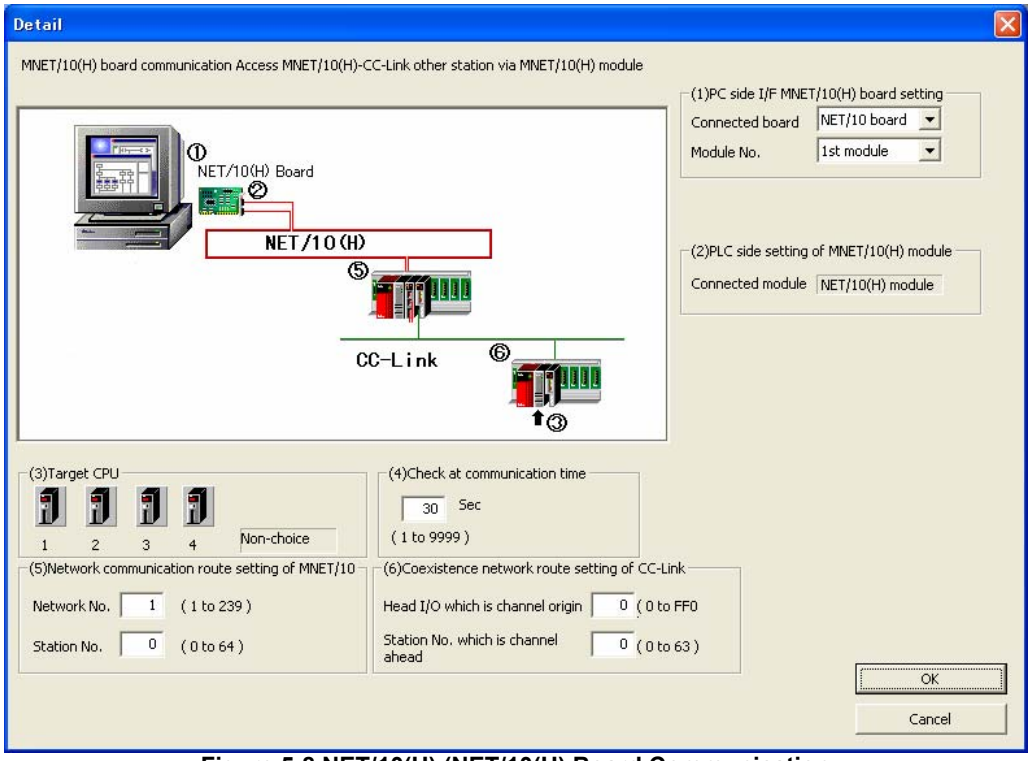


Figure 5-8 NET/10(H) (NET/10(H) Board Communication NET/10(H) through NET/10(H) Unit – CC-Link Other Station Access) Detailed Settings

For some communications routes that you can select, "(5)Network communication route setting of MNET/10" and "(6)Coexistence network route setting of C24" can not be set.

5.2.4.4. When CC-Link is selected

When "CC-Link" is selected as the computer interface, the communications routes that you can select are as follows.

Table 5-7 Communications Routes That Can Be Selected for "CC-Link"

No.	Communications Route	
1	CC-Link board communication	Other station access through CC-link unit
2	CC-Link board communication	CC-Link through CC-link unit – NET/10(H) other station access
3	CC-Link board communication	CC-Link through CC-link unit – Ethernet other station access

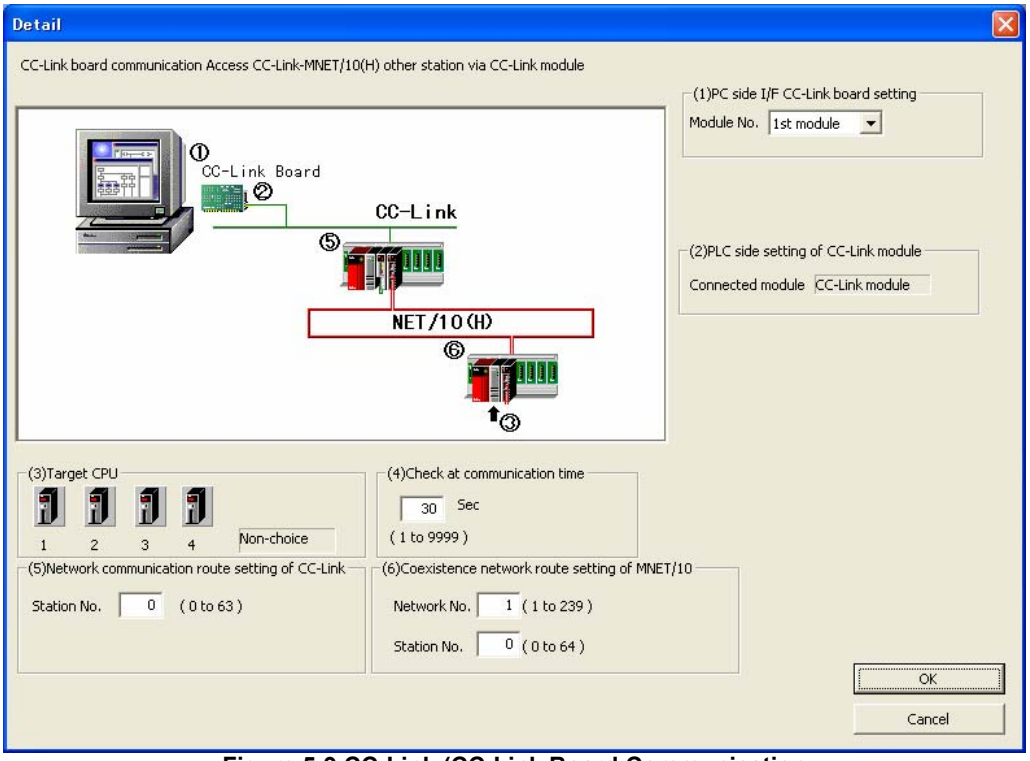


Figure 5-9 CC-Link (CC-Link Board Communication CC-Link through CC-Link Unit – NET/10(H) Other Station Access) Detailed Settings

For some communications routes that you can select, "(5)Network communication route setting of MNET/10" and "(6)Coexistence network route setting of C24" can not be set.

5.2.4.5. When Ethernet is selected

When "Ethernet" is selected as the computer interface, the communications routes that you can select are as follows.

Table 5-8 Communications Routes That Can Be Selected for "Ethernet"

No.	Communications Route	Special mention
1	Ethernet board communication Access other station via Ethernet module	
2	Ethernet board communication Access Ethernet-serial communication other station via Ethernet module	
3	Ethernet board communication Access Ethernet-CC-Link other station via Ethernet module	
4	Ethernet port direct communication PLC module connection	Ver.1.2 or later
5	Ethernet port direct communication Access serial communication other station via PLC module	
6	Ethernet port direct communication Access CC-Link other station via PLC module	
7	Ethernet port IP address designate communication PLC module connection	
8	Ethernet port IP address designate communication Access serial communication other station via PLC module	
9	Ethernet port IP address designate communication Access CC-Link other station via PLC module	

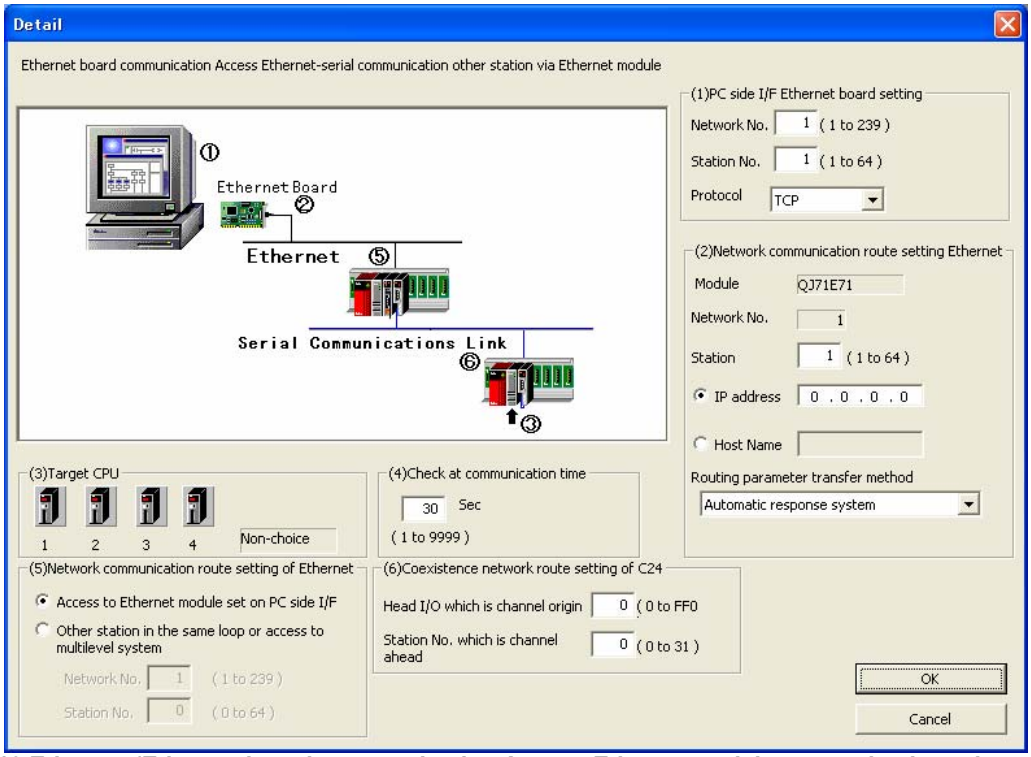


Figure 5-10 Ethernet (Ethernet board communication Access Ethernet-serial communication other station via Ethernet module) Detailed Settings

For some communications routes that you can select, “(1)PC side I/F Ethernet board setting”, “(2)Network communication route setting Ethernet”, "(5)Network communication route setting of MNET/10" and "(6)Coexistence network route setting of C24" can not be set.

5.2.4.6. When Q series bus is selected

When the "Q series bus" is selected as the computer interface, the communication route that can be selected is as follows.

Table 5-9 Communication Route That Can Be Selected with the "Q Series Bus"

No.	Communications Route
1	Q bus communications CPU unit connection

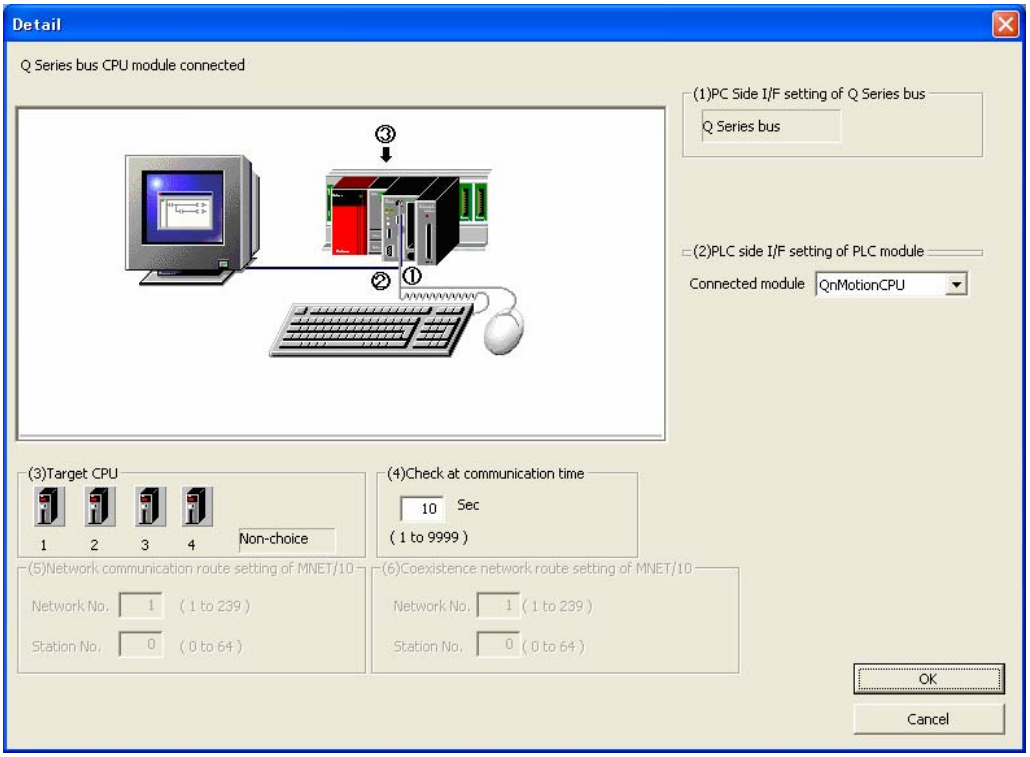


Figure 5-11 Q Series Bus (Q Bus Communications CPU Unit Connection) Detailed Settings

6. Robot Program Language Setting

This switches the robot program language used. The languages you can switch to are as follows.

Table 6-1 Robot Program Languages for Each Controller

Robot controller	Robot program language		
	MELFA-BASIC V	MELFA-BASIC IV	Movemaster command
CRnD-700 series	○	○	○
CRnQ-700 series	○	○	○
CRn-500 series	×	○	○

The Movemaster commands are restricted by the robot models that can be used. Before trying to use Movemaster commands, check in the "Standard Specifications" that the model you are using supports Movemaster commands.

Also, this software allows you to select Movemaster commands for CRnD-700 series and CRnQ-700 series robot controllers, but these robot controllers themselves do not actually support Movemaster commands. (As of December 2007)

⚠ Caution

When Movemaster commands are used, the simulation functions cannot be used.

On the project tree, click the desired project name with the right mouse button, then from the right button menu, select "Edit Project". The project edit screen is displayed.

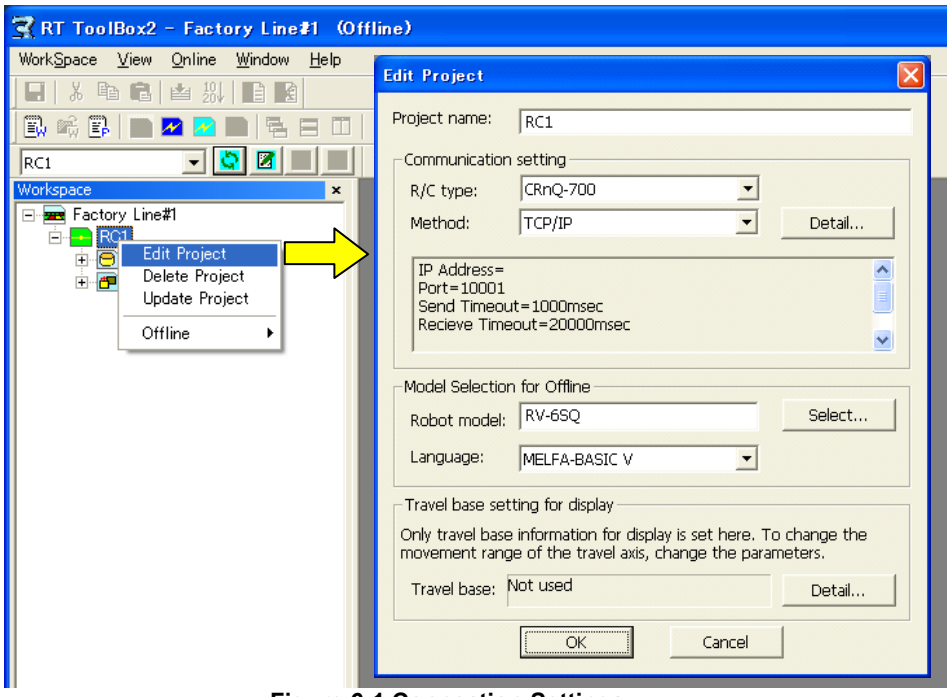


Figure 6-1 Connection Settings

Select the controller connected, the robot model, and the robot language used, then click the [OK] button

7. Writing Programs

This chapter explains robot program editing methods. You can directly edit programs in the robot controller or edit programs stored on the computer.

Caution

Program names that can not be handled on the computer

If a program name in the robot controller is the same as a "reserved term" in Windows, when you try to open that program for program editing, this causes an error. In this case, it is necessary to change the program name in the controller.

"Reserved terms" are special character strings that the Windows system uses and therefore can not be used as file names. These "reserved terms" are character strings such as the following.

AUX, COM1 to COM9, CON, LPT1 to 9, NUL, PRN

7.1. Writing a New Program

7.1.1. Writing a new program on the computer

For the new program you are going to write, select [Offline] -> [Program], then click the right mouse button. The right button menu is displayed, so click [New].

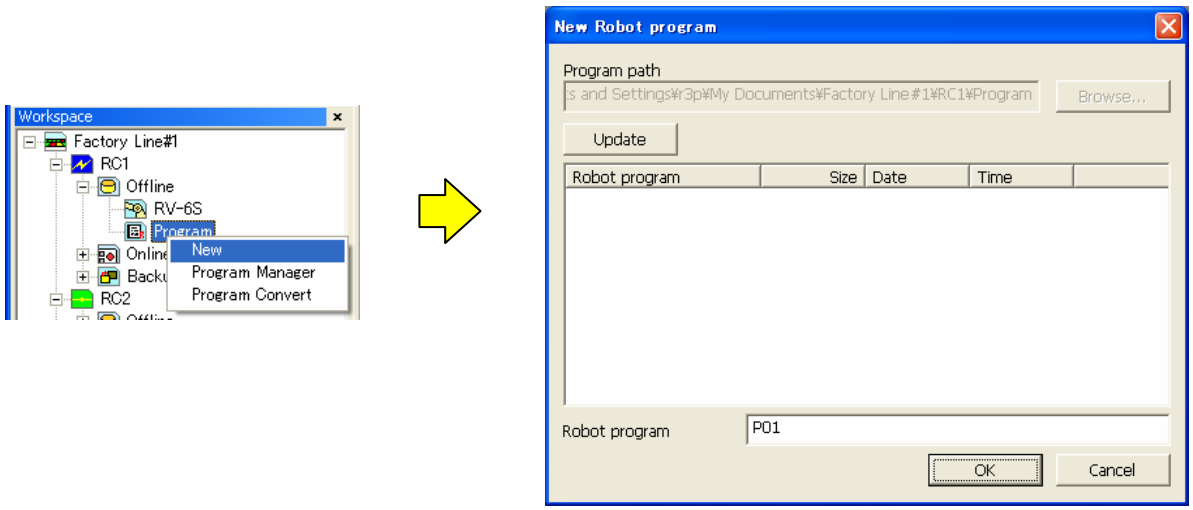


Figure 7-1 Writing a New Program on the Computer

Input the robot program name, then click the [OK] button.

Caution

Folder in which programs are stored

Programs on the computer are managed in units of workspace projects. The folder they are stored into is workspace writing folder¥project name¥Program.

To store into any other folder, first store in this folder, then copy into the desired folder with the program management copy function.

7.1.2. Writing a new program in the robot controller

For the new program you are going to write, select [Online] -> [Program], then click the right mouse button. The right button menu is displayed, so click [New].

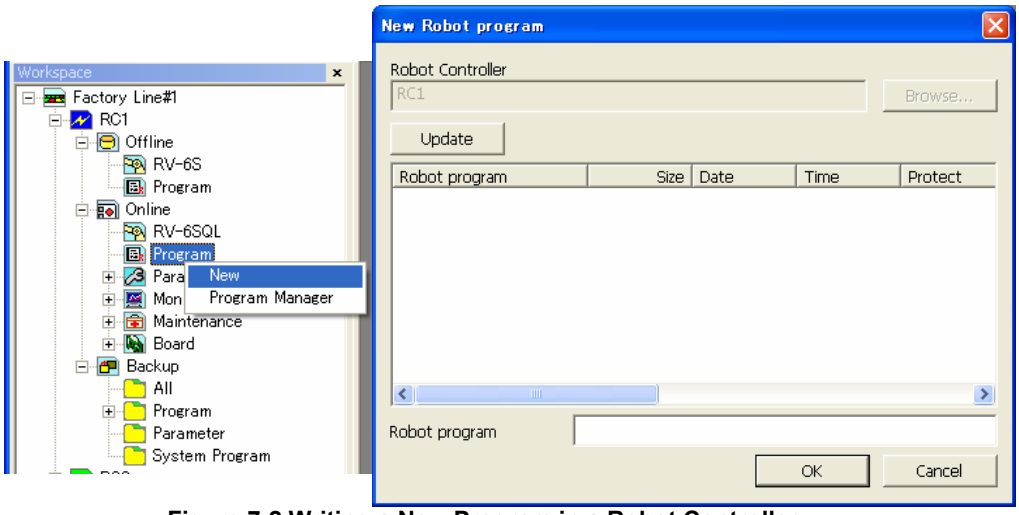


Figure 7-2 Writing a New Program in a Robot Controller

Input the robot program name, then click the [OK] button. The "Read Item" screen is displayed, so check the read items, then click the [OK] button. For details on the read items, see **"7.2.3 Read Items when opening program in robot controller".**

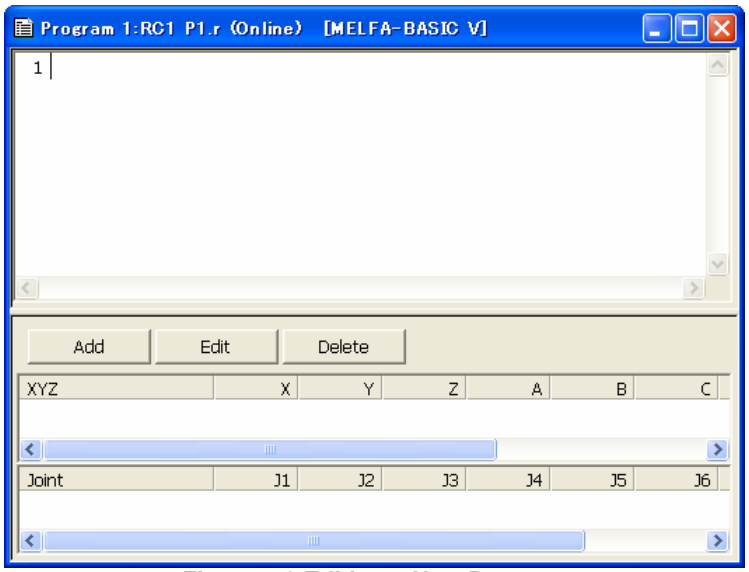


Figure 7-3 Editing a New Program

7.2. Opening an Existing Program

7.2.1. Opening an existing program on the computer

Open the target project in the project tree with [Offline] -> [Program]. The stored programs are displayed on the project tree, so double click the program you want to edit.

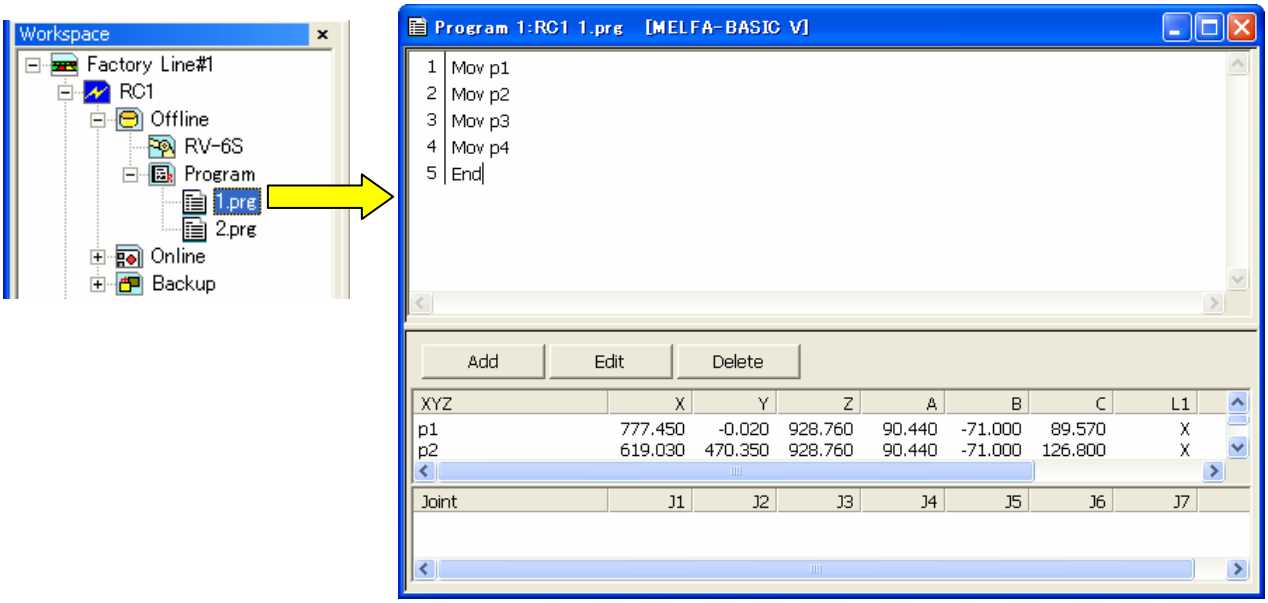


Figure 7-4 Opening a Program on the Computer

7.2.2. Opening a program in a robot controller

Open the target project in the project tree with [Online] -> [Program]. The stored programs are displayed on the project tree, so double click the program you want to edit.

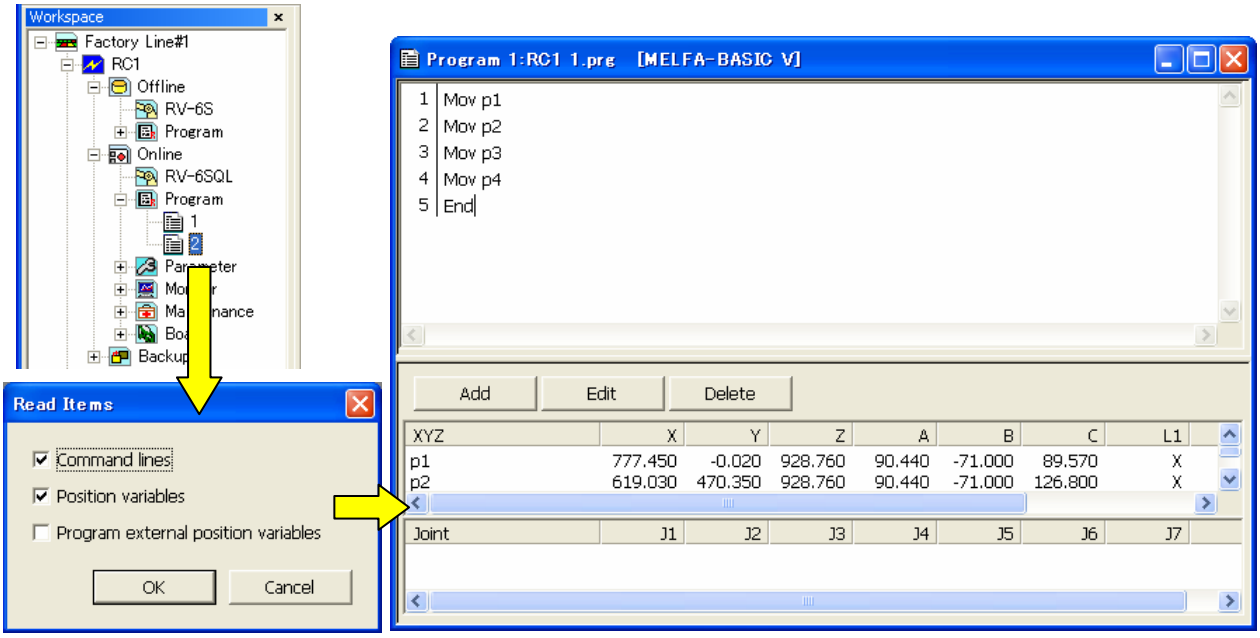


Figure 7-5 Opening a Program in a Robot Controller

Input the robot program name, then click the [OK] button. The "Read Item" screen is displayed, so check the read items, then click the [OK] button. For details on the read items, see **"7.2.3 Read Items when opening program in robot controller".**

7.2.3. Read Items when opening program in robot controller

You can set the robot program read items divided into command lines, position variables, and program external position variables. This function is displayed after H1 edition on CRn-500 series controller.

The default values of read items are as follows.

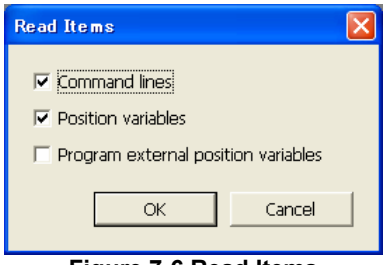


Figure 7-6 Read Items

The program external position variable read operations are shown in **"Table 7-1 ".** (For details on program external position variables, see **"Details of functions and operations"** in the robot controller's user's manual.)

Table 7-1 Program External Position Variable Read Operations

		Read Item			Position variable, joint variable (P_01, J_02, etc.) (MOVEMASTER command : 901-999)	Position array variable, joint array variable (P_100( ), J_102( ), etc.)
		Command	Position	External position variable		
CRn-700 series robot controller		☑	☑	☑	Reads only the external position variables (position variables, joint variables) used in instruction statements. (*1)	Reads all elements used in instruction statements. (*2)
		☑	☐	☑		
		☐	☑	☑	Reads all external position variables (position variables, joint variables, position array variables, joint array variables).	
		☐	☐	☑		
CRn-500 series robot controller's software version	Ver.J1 or later	☑	☑	☑	Reads only the external position variables (position variables, joint variables) used in instruction statements. (*1)	Reads all elements used in instruction statements. (*2)
		☑	☐	☑		
		☐	☑	☑	Reads all external position variables (position variables, joint variables, position array variables, joint array variables).	
		☐	☐	☑		
	Ver.H1 to H7	☑	☑	☑	Reads all external position variables (position variables, joint variables, position array variables, joint array variables).	
		☑	☐	☑		
		☐	☑	☑		
		☐	☐	☑		
	Ver.G9 orearlier	This screen is not displayed.				

*1: When Movemaster commands are used, all external position variables are read.

*2: When only P_100(1) is used in the command statement, P_100(1) to P_100(10) are all read. However, the number of valid elements depends on the robot controller software version used.

7.2.4. Opening a program in the backup data

Program data backed up on the personal computer by the backup function can be opened by the program edit. Note that this function can be used with Version 1.2 or later of this software.

Open the target project in the project tree with [Backup]. Select the backup data (All files or Program) stored the program you want to edit. From the right mouse button menu, click [Open Backup program]. The stored programs are displayed on “Open Backup Program” window, so select the program you want to edit, and click [OK] button.
The extension of program data file in the backup data is “*.MB5” or “*.MB4”.

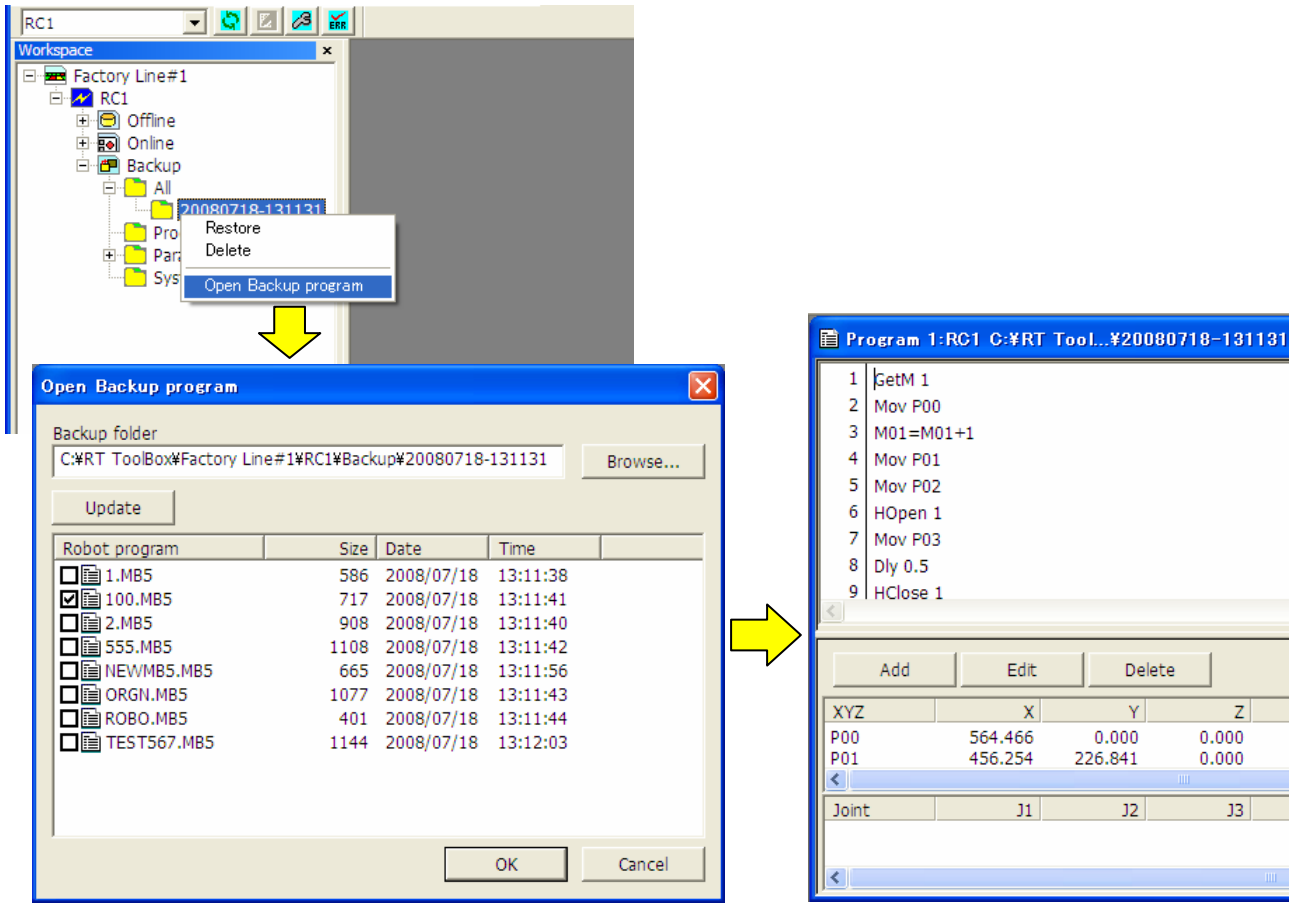


Figure 7-7 Opening a program in the backup data

The opened program in the backup data can be saved as a usual program (text type file whose extension is “*.prg”).

⚠ Caution

Notes of opening the program backed up.

The notes of when the program backed up is opened are as follows,

(1) The program that can be opened is only a program made in the robot program language specified by the current project.
(2) The program external position variables used in the program cannot be read.
(3) If the program is opened from Program backup data, the values of the user definition external variables become 0.
(4) If the program is opened from Program backup data, the values of the additional axis used might be not converted correctly.

7.3. Explanation of Program Edit screen

This explains the program edit screen.

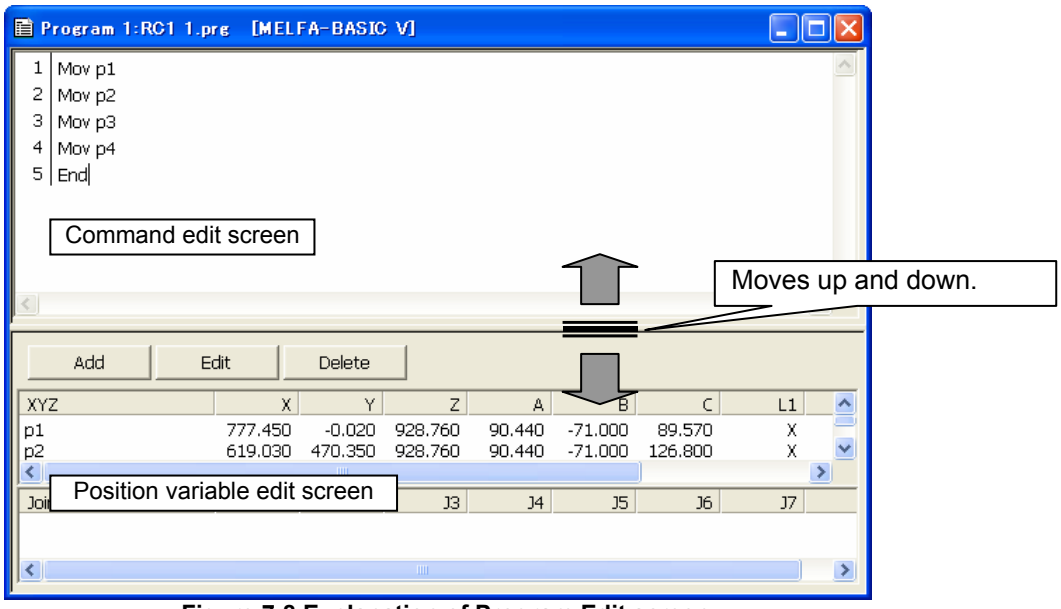


Figure 7-8 Explanation of Program Edit screen

The top part of the screen is the program command statement edit screen and the bottom part is the position variable edit screen.

To change the position dividing the top and bottom screens, drag the border line with the mouse. This is handy when you want to expand the command edit screen, for example because there are many lines of commands.

This edit screen display area can be customized. For details, see **"7.5.1 Changing the display area".**

Also, the background color for the command edit screen depends on the robot language used.

Table 7-2 Command Edit Screen Background Color

Robot language	Background color	
	Normal	Debugging
MELFA-BASIC V	White	Light blue
MELFA-BASIC IV	Light yellow	
Movemaster commands	Light green	

Debugging means when the program is opened in debugging status.

7.4. Program Editing Menu Bar

During program editing "File", "Edit", "Debug", and "Tool" are added to the menu bar.

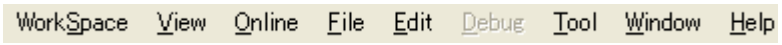
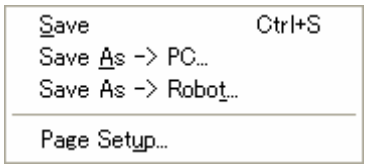
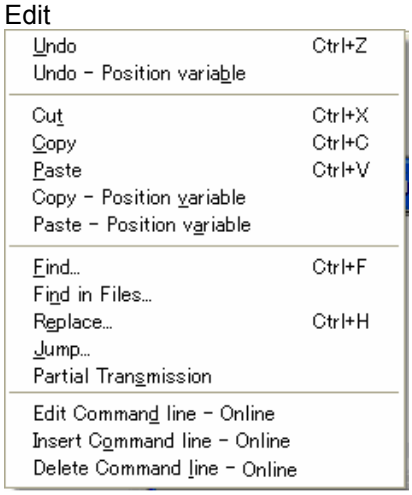
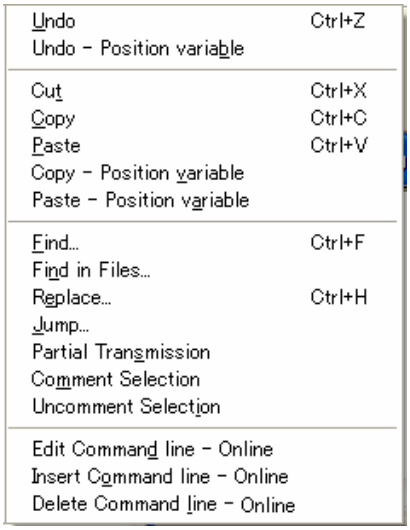
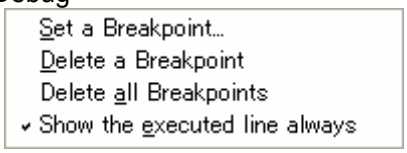
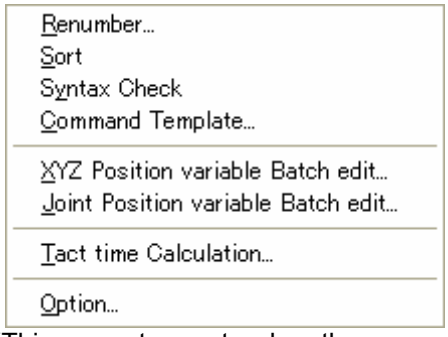


Figure 7-1 Program Editing Menu Bar

The menus are as follows.

Menu item		Explanation	Menu item
File  	Save	Saves the program being edited with its current name.	7.7.1
	Save As -> PC	Saves the program being edited with its current name on the PC	7.7.2
	Save As -> Robot	Saves the program being edited with its current name on Robot controller	7.7.3
	Page Setup	You can customize the pages the program is printed with.	7.8.3
Edit  (Ver.1.1 or earlier)  (Ver.1.2 or later) (The function of “Comment Selection/Uncomment Selection” can be used in Ver.1.2 or later.)	Undo	Cancel the edition by the command and return to the previous state	
	Undo - Position variable	Cancel the edition by the position variable and return to the previous state	
	Cut	Cuts a character string from the command being edited.	7.6.4.2
	Copy	Copies a character string in the command being edited.	7.6.4.1
	Paste	Pastes the copied or cut character string to the specified location.	7.6.4.3
	Copy - Position variable	Copies position data.	7.6.4.4
	Paste - Position variable	This pastes the copied position data.	7.6.4.5
	Find	This searches for the specified character string.	7.6.4.6
	Find in Files	This searches for the specified character string in the program files.	7.6.4.7
	Replace	This replaces the specified character string with another character string.	7.6.4.8
	Jump	Jumps to the specified step number or label.	7.6.4.9
	Partial transmission	Writes the selected program lines to the robot controller. This can not be used when the program is opened in debugging status.	7.6.4.10
	Edit Command line - Online	Edit the command lines for a program pened in debugging status.	7.9.3
	Insert Command line - Online	Insert the command lines for a program opened in debugging status.	
	Delete Command line - Online	Delete the command lines for a program opened in debugging status.	
	Comment Selection	The selected lines are exchanged by the batch as a comment.	7.6.4.16
	Uncommnet Selection	The comment on the selected lines are released by the batch.	

Menu item		Explanation	Menu item
Debug  (This can only operate when the program has been opened in debugging status.)	Set a Breakpoint	You can set a breakpoint in a program opened in debugging status.	7.9.4
	Delete a Breakpoint	You can delete a breakpoint in a program opened in debugging status.	
	Delete all Breakpoints	You can delete all breakpoints in a program opened in debugging status	
	Show the executed line always	You can set to dispplay/not to display the executed line always to the program opened in debugging status.	7.9.1
Tool  (This can not operate when the program has been opened in debugging status.) Renumbering and sorting can only be used with MELFA-BASIC IV and Movemaster commands. Tact time calculation can only be used in a simulation.	Renumber	The renumbering function can only be used with MELFA-BASIC IV and Movemaster commands. You can renumber line numbers in a batch.	7.6.4.14
	Sort	The sorting function can only be used with MELFA-BASIC IV and Movemaster commands. This sorts the edited program by line number.	7.6.4.15
	Syntax Check	You can check whether or not the edited robot program is syntactically correct.	7.6.4.11
	Command Template	You can display a list of the commands and make insertions on the program command edit screen.	7.6.4.12
	XYZ Position variable Batch Edit	You can change the position variables in the program being edited in a batch and can sum up all the values	7.6.4.13
	Joint Position variable Batch Edit		
	Tact time calculation	Tact time calculation can only be used in a simulation. Tact time of the program can be calculated.	13.12
	Option	You can customize the program edit area, screen display area and syntax check before saving a program	7.5 7.7.5

7.5. Customizing the Program Edit Screen

You can customize the program edit screen.

With the program opened, on the menu bar, click [Tool] -> [Option] and set the program edit screen as you want with the displayed option screen.

Clicking the [Restore Defaults] button restores the default settings.

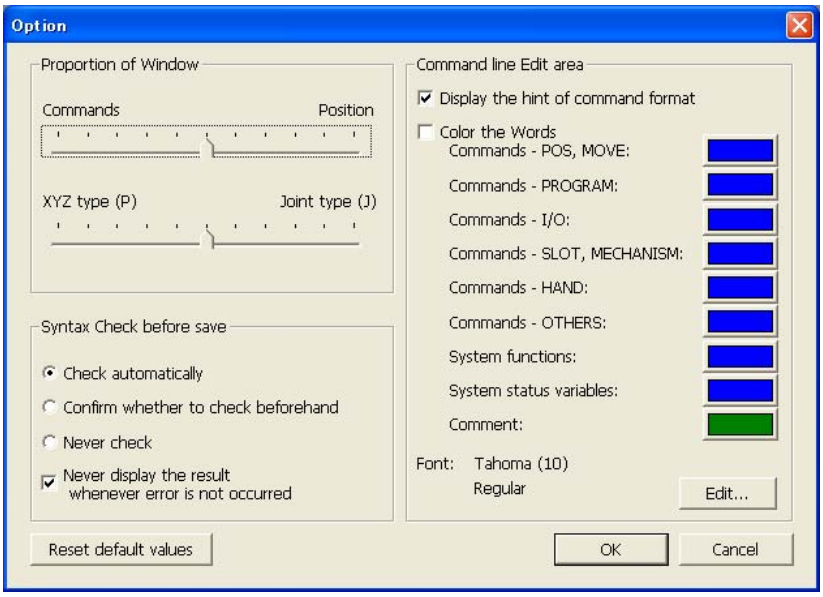


Figure 7-9 Option Screen

7.5.1. Changing the display area

In the "Display area" group, you can set the command display area and position edit area display ratios and for the position edit area, the XYZ (P) variable and joint (J) variable proportions.

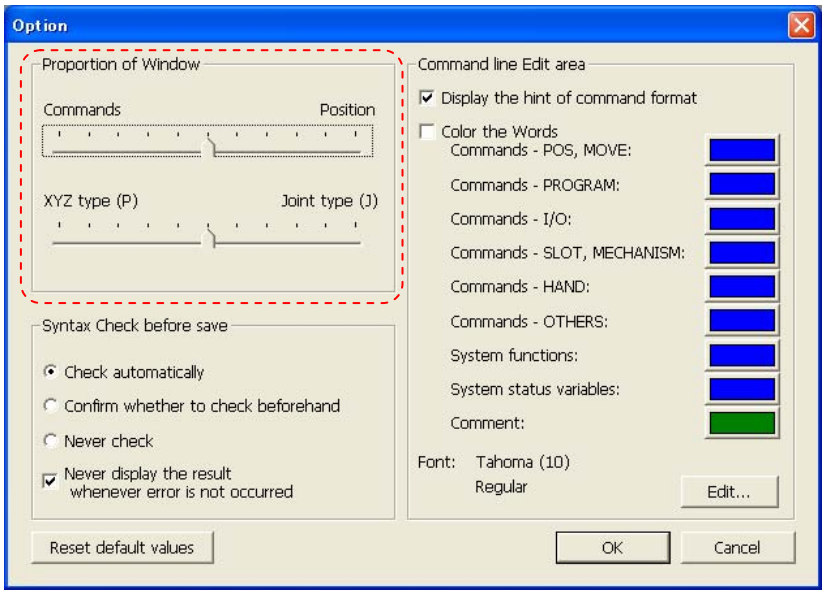


Figure 7-10 Program Edit Screen Display Area Change

7.5.2. Command format hints

You can use pop-up hint display to display the format for the robot program command, system functions, and system status variables displayed in the command edit area.

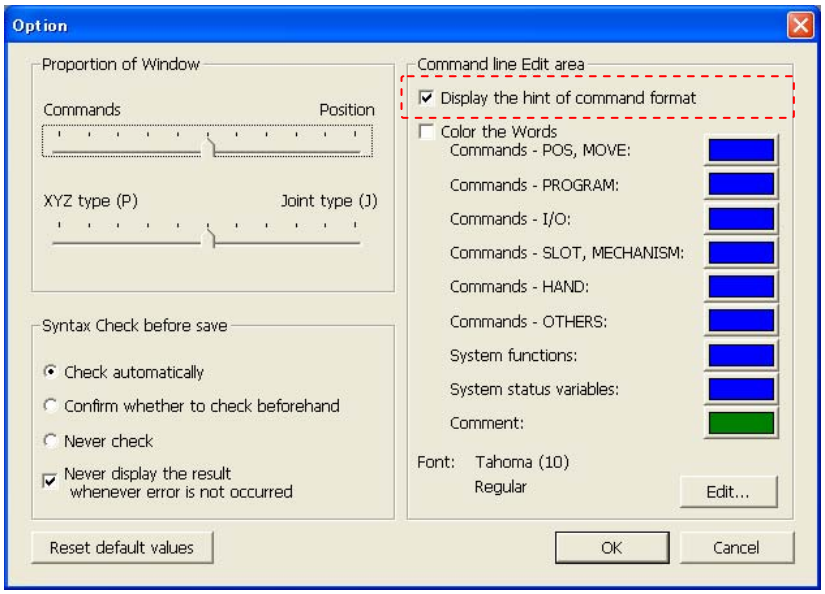


Figure 7-11 Command Edit Area Command Format Hint Display Settings

7.5.3. Character colors

You can assign the colors for displaying robot program command, system functions, and system status variables displayed in the command edit area.

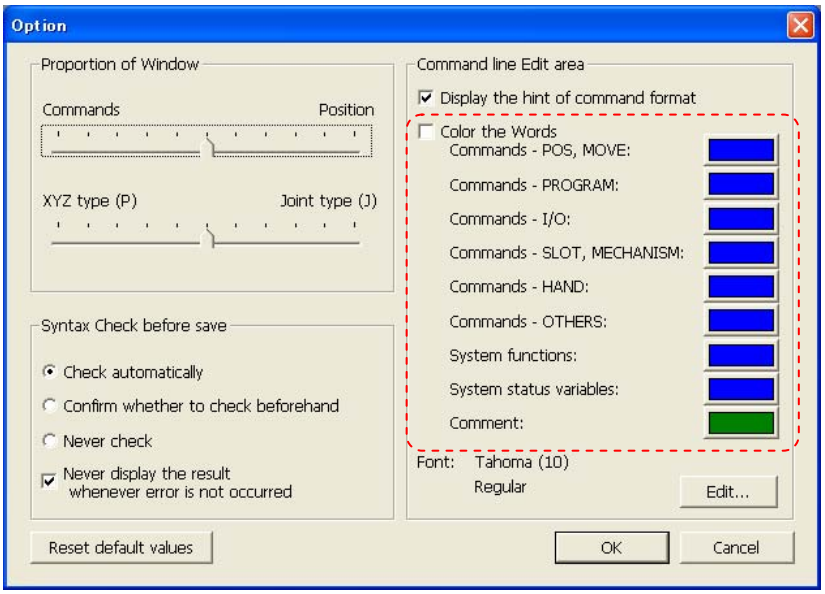


Figure 7-12 Command Edit Area Character Color Settings

7.5.4. Changing the font

You can change the font displayed in the command edit area.

Click the option screen font [Change] button. The font setting screen is displayed, so after setting the font name, style, and size, click the [OK] button.

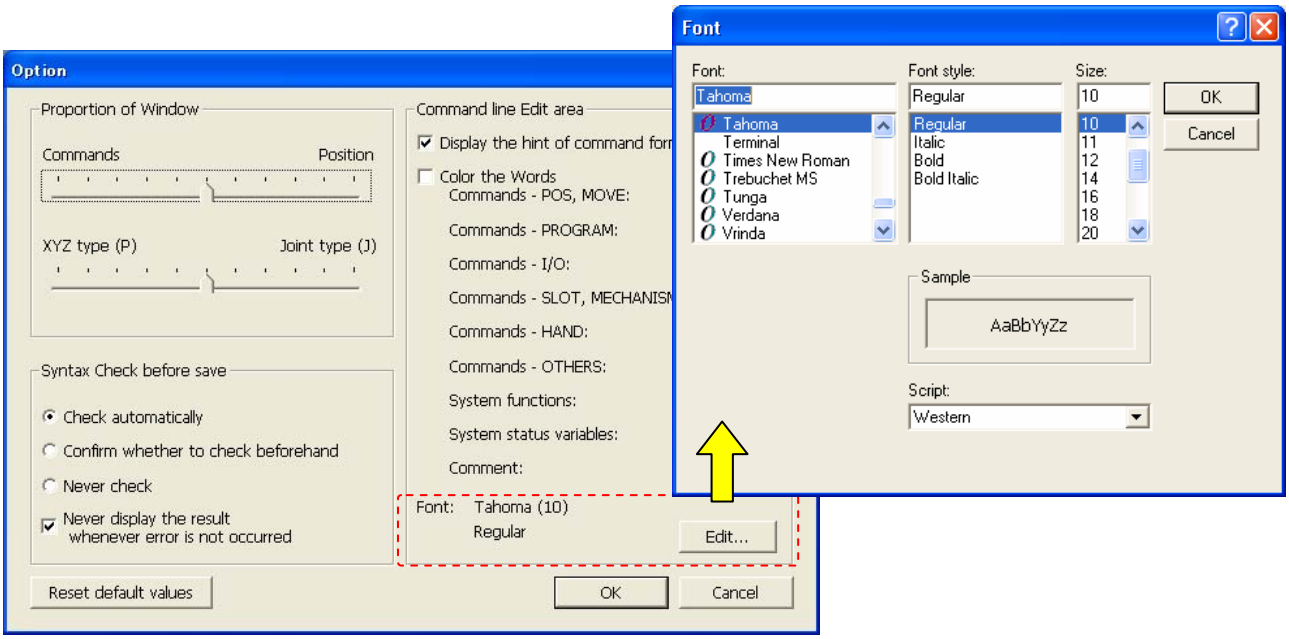


Figure 7-13 Command Edit Area Font Setting

7.6. Program Editing

This chapter explains the methods for editing MELFA-BASIC V programs.
For RT ToolBox2 command statement editing, you can input in the same way as with a general editor like a notebook. There is no need to input the [Enter] key for each line as was the case with RT ToolBox.

7.6.1. MELFA-BASIC V command statement editing

When writing a program using MELFA-BASIC V, you do not use line numbers, unlike MELFA-BASIC IV or Movemaster commands. Instead the step position is displayed on the left end. These step numbers are automatically displayed with the keyboard [Enter] key.

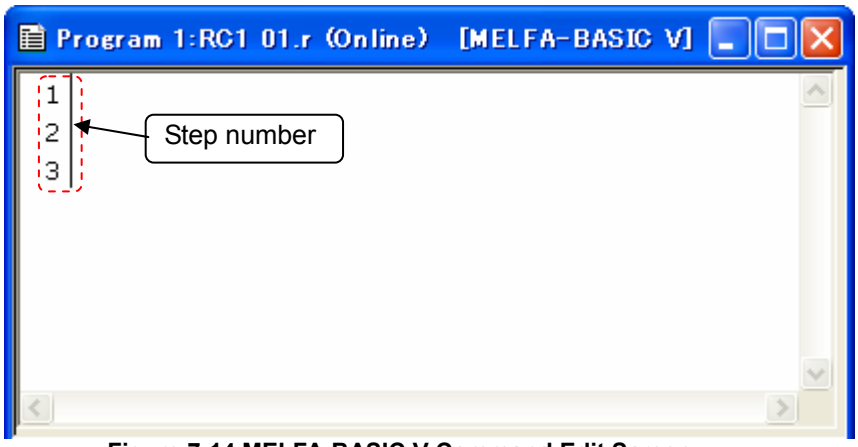


Figure 7-14 MELFA-BASIC V Command Edit Screen

Even if you input MELFA-BASIC V commands with all lowercase characters, when the program is written to the robot controllers, the commands are converted into the correct mixture of uppercase and lowercase letters.

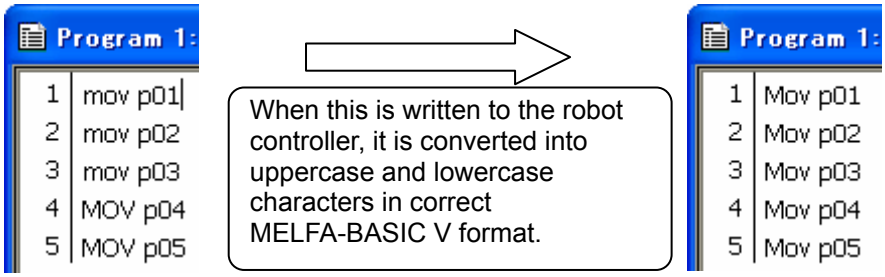


Figure 7-15 MELFA-BASIC V Character Input

7.6.2. MELFA-BASIC IV and Movemaster command command statement editing

When writing a program using MELFA-BASIC IV or Movemaster commands, step numbers like those of MELFA-BASIC V are not displayed. Input the line number at the front of the command statement.

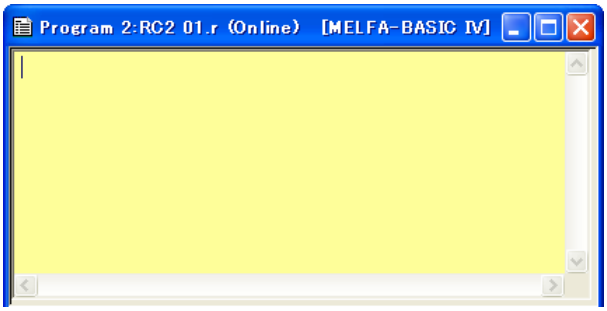


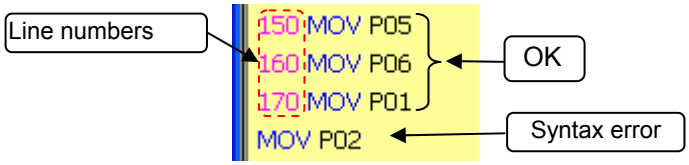
Figure 7-16 MELFA-BASIC IV Command Edit screen

Caution

For MELFA-BASIC IV and Movemaster commands, input line numbers.

Step numbers are not displayed on the MELFA-BASIC IV and Movemaster command edit screen. When using MELFA-BASIC IV and Movemaster commands, always input line numbers.

If there are no line numbers, this is a syntax error.



There is a function for sorting commands in order of line number even if you do not input them in order of line numbers. For details, see **"7.6.4.15 Sorting".**

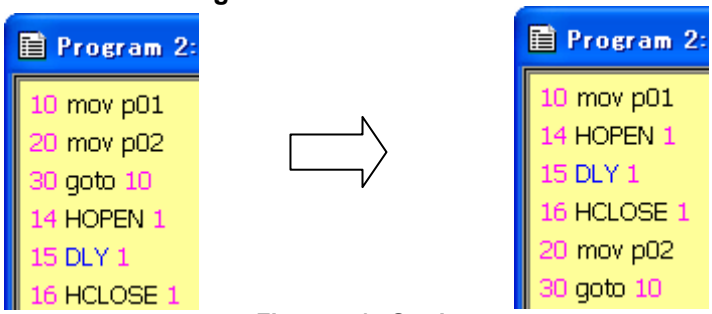


Figure 7-17 Sorting

There is a function for reordering commands in order of line number even if you do not input them in order of line numbers. For details, see

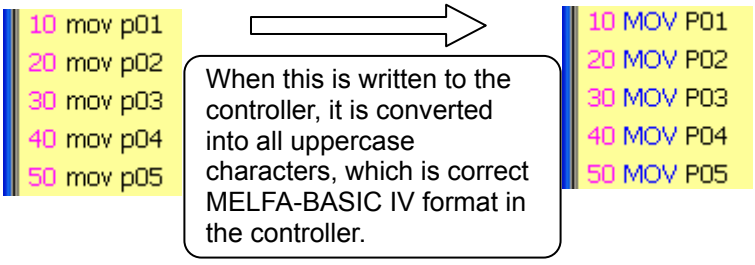


Figure 7-18 MELFA-BASIC IV Character Input

7.6.3. Position variable editing

Position variables are edited on the position edit screen. The upper list is a list of XYZ coordinate variables and the lower list is a list of joint coordinate variables.

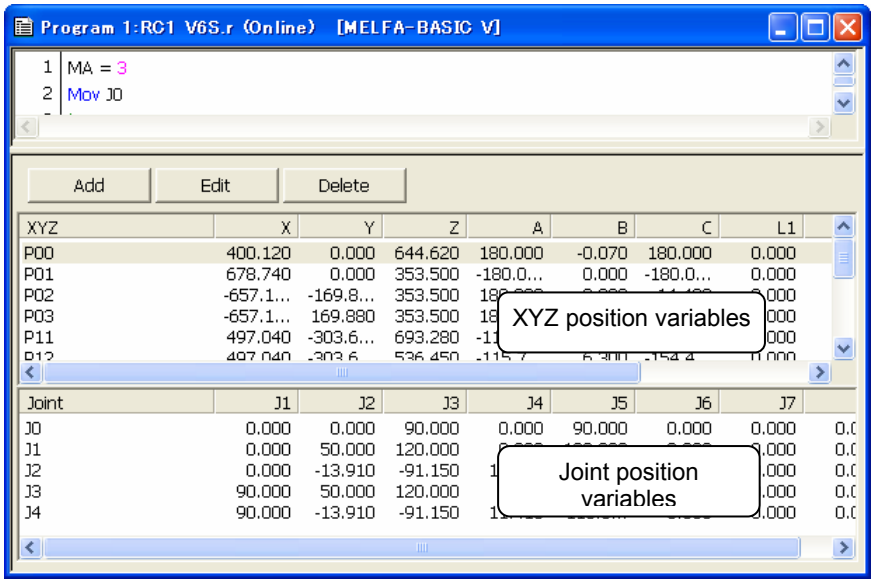


Figure 7-19 Position Variable Edit Screen

Array variables are displayed developed in their own lists.

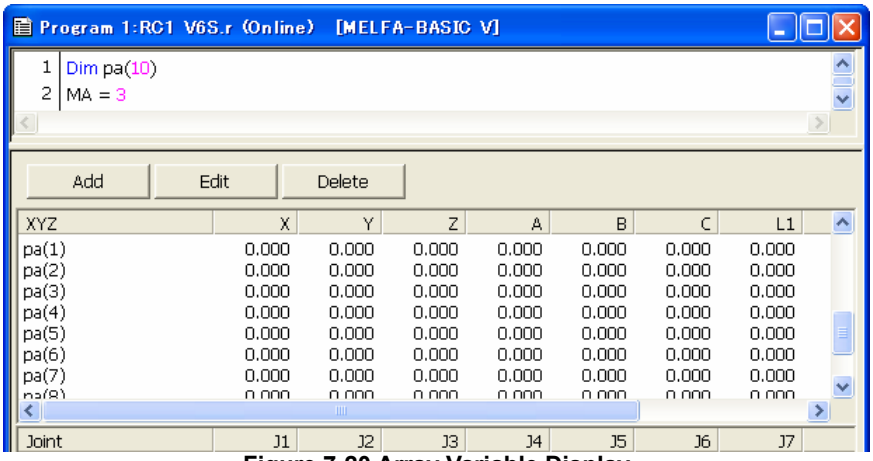


Figure 7-20 Array Variable Display

Caution

About uppercase characters and lowercase characters in position variables names

With MELFA-BASIC V, you can use lowercase letters in variable names.

This software does not differentiate between uppercase letters and lowercase letters in variable names. For example, the position variables PA and pa are recognized as the same position variables.

The controller converts all later variables to match the first position variable name defined. For example, if you write a program like that below, the position variable "pa" is used and "PA" is converted into "pa".

```
1 Mov pa        1 Mov pa
2 Mov PA        2 Mov pa
```

When this is written to the controller, it is converted in the controller as on the right.

7.6.3.1. Adding/changing position variables

To add a position variable, click the [Add] button. The position variable add screen is displayed. At this time, if position data is selected in the list, the contents of that position data (XYZ/joint, position information) are displayed. However, the variable name remains blank.

To revise a position variable, select the position variable to be revised in the list, then click the [Change] button.
The selected position variable is displayed. At this time, the variable name can not be changed.

Select either XYZ coordinate type or joint coordinate type, input the values of each element of the position data, input the position variable name, then click the [OK] button.
While editing an online program in online status or simulation status, you can read the current robot position by pressing the [Read Current Position] button.

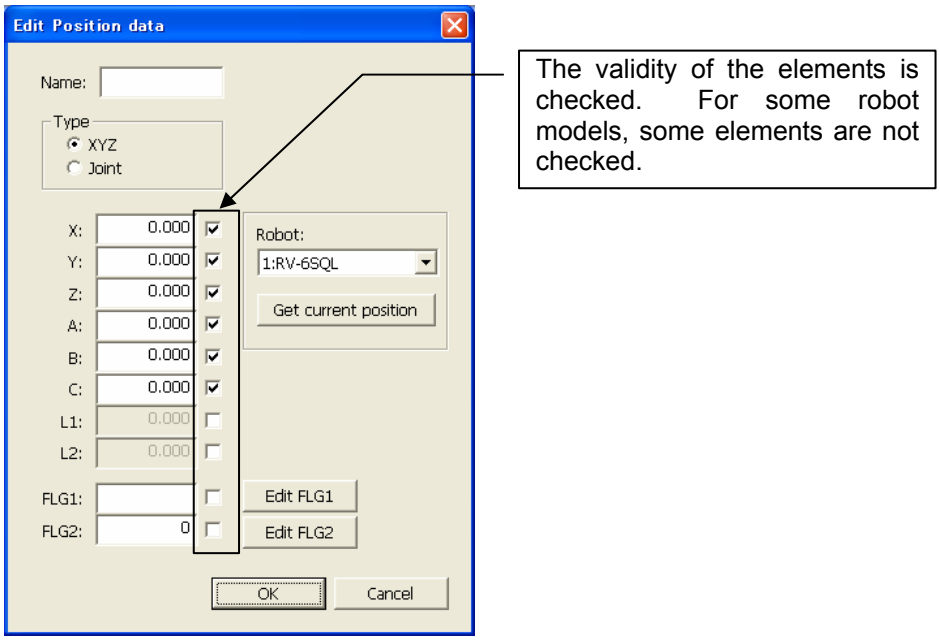


Figure 7-21 Position Data Variables

Caution

When adding a position array variable, specify the array name and element numbers as the variable name.
Also, when writing to a robot, always write the Dim declaration in a command statement. If there is no Dim declaration, an error occurs when you write to the robot controller.

Caution

The units for elements used depend on the robot. See the robot's own operations manual.

7.6.3.2. Deleting position variables

After selecting the target position variable, click the [Delete] button. The selected position variable is deleted. You can also delete multiple position variables at the same time. You can select multiple position variables by clicking position variables while holding down the [Ctrl] key or the [Shift] key on the keyboard. However, you cannot select XYZ position variables and joint position variables at the same time.

⚠ Caution

To delete a position array variable, delete the command statement. "Dim" declaration.

With this software, even if a position variable is deleted, if a program with a "Dim" declaration is written to a robot controller, the position array variables declared with the "Dim" are left with 0 for all their components.

7.6.4. Edit assist functions

This explains the edit assist functions, which help in command editing.

Edit assist functions such as copy, cut, find, find and replace, and jump are used from [Edit] and [Tool] on the menu bar.

7.6.4.1. Copy

Copies a character string in the command being edited. You can also copy multiple lines.
After selecting the character string to be copied, click on the menu bar [Edit] -> [Copy].
You can use the paste function, explained below, to paste this copied character string to another location in the program.
For details on position data copying, see **"7.6.4.4 Copy position data".**

7.6.4.2. Cut

Cuts a character string from the command being edited. You can also cut multiple lines.
After selecting the character string to be cut, click on the menu bar [Edit] -> [Cut].
You can use the paste function, explained below, to paste this cut character string to another location in the program.

7.6.4.3. Paste

Pastes the copied or cut character string to the specified location.
Put the cursor where you want to paste, then click on the menu bar [Edit] -> [Paste]. The copied or cut character string is inserted at the specified location.
For details on position data pasting, see **"7.6.4.5 Pasting position data".**

7.6.4.4. Copy position data

Copies position data. You can also copy multiple position data items.
After selecting the position data to be copied, click on the menu bar [Edit] -> [Copy – Position data].

7.6.4.5. Pasting position data

This pastes the copied position data.
Make active the program you want to paste into, then click on the menu bar [Edit] -> [Paste – Position data].
The copied position data is inserted into the specified program.
At this time, if there is already position data with the same name in that program, a confirmation message is displayed.

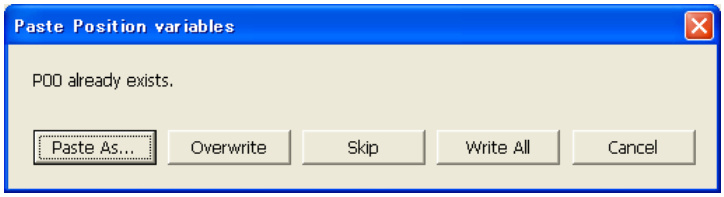


Figure 7-22 Paste Position Data Confirmation Message

7.6.4.6. Find

This searches for the specified character string.
Click the menu bar [Edit] -> [Find]. The find screen is displayed.

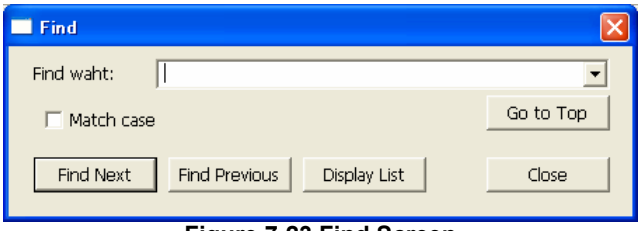


Figure 7-23 Find Screen

Input the character string to find, then click [Find Next] or [Find Previous]. The character string search starts.

If you click [Display List], all the instances of the specified character string are found from programs and displayed in a list.

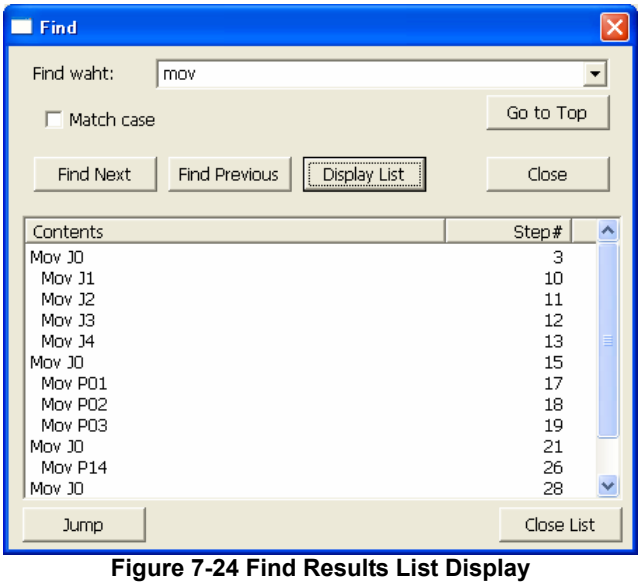


Figure 7-24 Find Results List Display

When you select an item from the find results list and click the [Jump] button, the display jumps to the line that includes the selected item.

7.6.4.7. Find in Files

This searches for the specified character string in the "Online" or "Offline" program files registered in the current project.

Click the menu bar [Edit] -> [Find in Files]. The find in files screen is displayed.

You can also display this screen by clicking the program management [Find in Files] button.

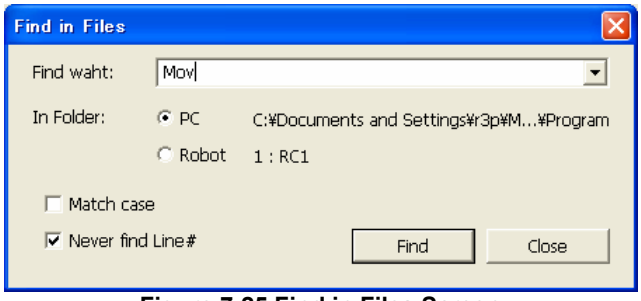


Figure 7-25 Find in Files Screen

Input the character string to find and select the location to search.

When "PC" is selected as the location to search, all the offline programs registered in the current project are searched.

When "Robot" is selected as the location to search, all the online programs registered in the current project are searched. In other words, all the programs in robot controllers in the "online" status connected to a robot are searched. If a simulation is running, all the programs in the virtual controller are searched. When you search in controllers, the communications with the controllers may take time.

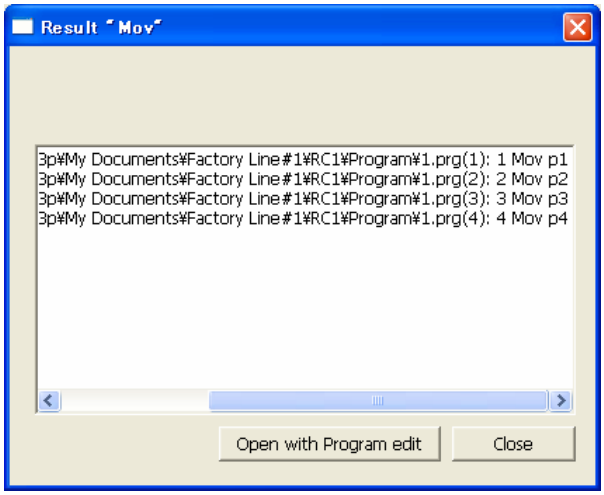
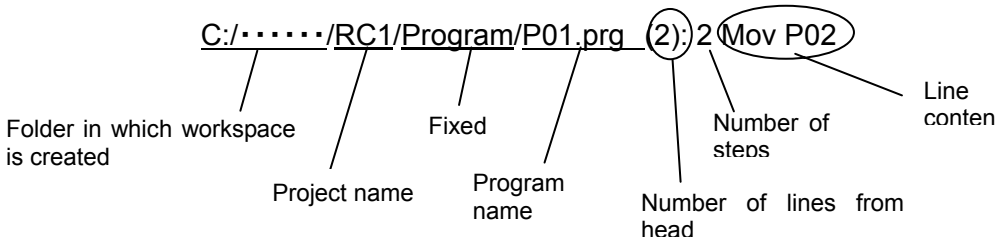


Figure 7-26 Results of Search from File



With the [Open with Program edit] button, you can open the program that includes the contents of the line selected from the list of search results.

7.6.4.8. Replace

This replaces the specified character string with another character string.
Click the menu bar [Edit] -> [Replace]. The Replace Screen is displayed.

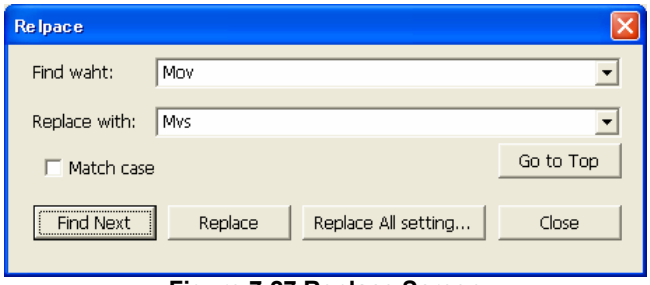


Figure 7-27 Replace Screen

[Find Next] : Searches for the next instance of the character string to be replaced.
[Replace] : Replaces the found character string.
[Replace All] : An item is displayed for specifying the range in which to replace all instances of the specified character string.

With Replace All, you can specify a range in which to replace.
When you click the [Replace All] button, all the instances in the specified range are replaced.

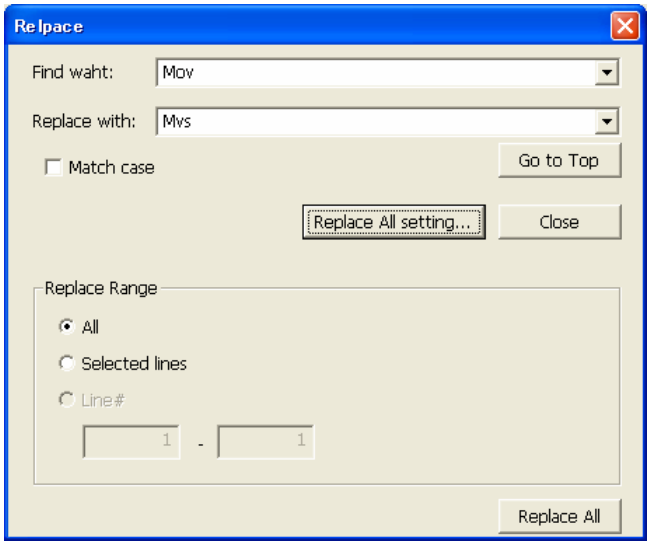


Figure 7-28 Replace All Setting Screen

7.6.4.9. Jump to specified line

Jumps to the specified step number or label.
Click on the menu bar [Edit] -> [Jump]. The jump screen is displayed.

Figure 7-29 Jump Screen

Input the step number or label to jump to, then click the [OK] button. Display jumps to the specified step number or label.

7.6.4.10. Partial writing

Writes the selected program lines to the robot controller.

This is handy for reflecting the contents of the partially revised program in the robot controller, but be careful. Only the selected part of the program is written.

Select the lines to be written to the robot controller, then click on the menu bar [Edit] -> [Partial Write]. Check the contents to be written, then click [Yes].

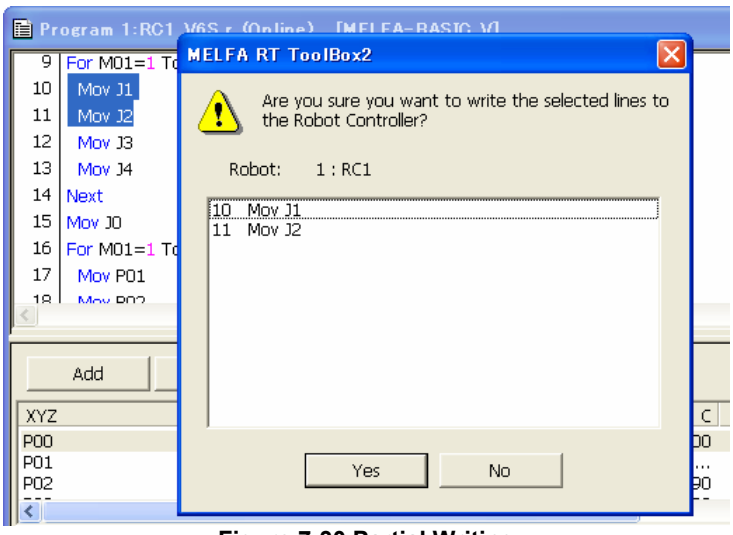


Figure 7-30 Partial Writing

7.6.4.11. Syntax check

You can check whether or not the edited robot program is syntactically correct. Execute this before writing the program to the robot controller.

Click on the menu bar [Tool] -> [Check Syntax]. If there is a syntax error, the error location and details are displayed.

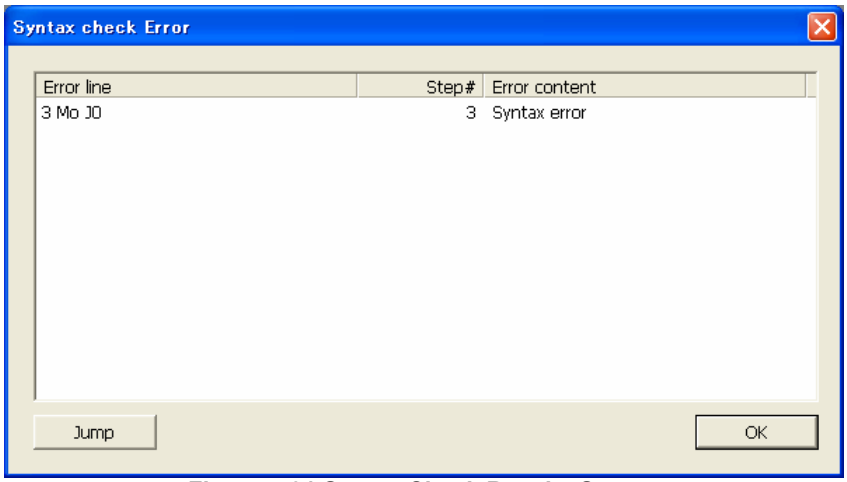


Figure 7-31 Syntax Check Results Screen

If you select the detected error and click the [Jump] button, it jumps to the command statement with the error.

7.6.4.12. Command template

You can display a list of the commands and make insertions on the program command edit screen.
Click on the menu bar [Tool] -> [Command Template].

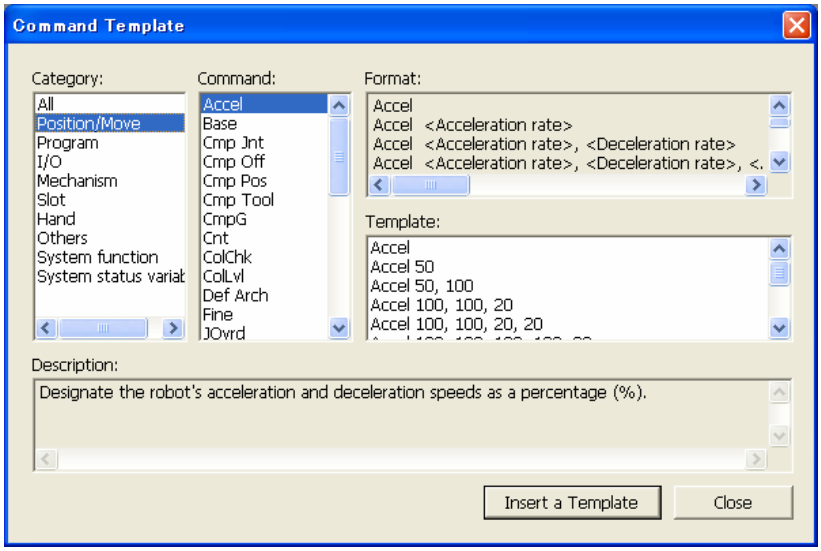


Figure 7-32 Command Template

When you select the template for the selected command from the list, then either click the [Insert Template] button or double click, the command is inserted onto the program command edit screen.

7.6.4.13. XYZ position data batch editing/joint position data editing

You can change the position variables in the program being edited in a batch and can sum up all the values. For example, you can add 10.00 to the X components of the P00, P01, P02, P03, and P04.

To batch edit XYZ position variables, click on the menu bar [Tool] -> [Batch Edit XYZ Position Data].
To batch edit joint position variables, click on the menu bar [Tool] -> [Batch Edit Joint Position Data].
All the position variables of the respective type are displayed.

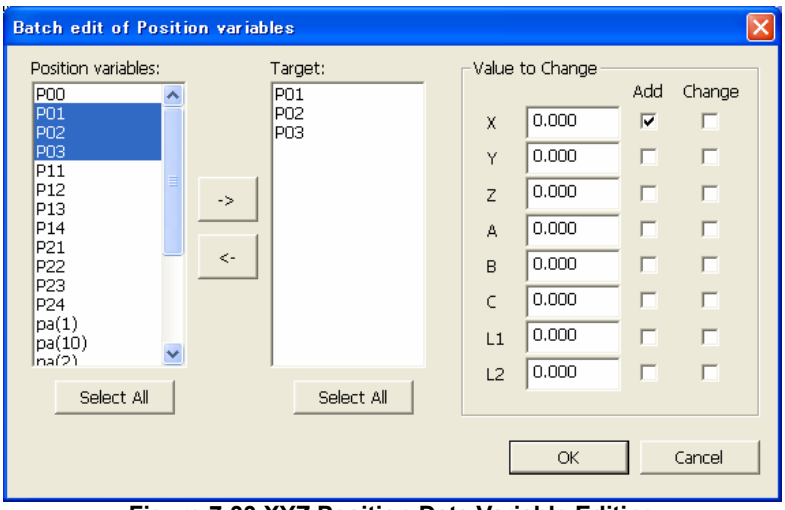


Figure 7-33 XYZ Position Data Variable Editing

From the position variable list, select the position variable to change, then add it to the change list with the [->] button.
Input the change value, select the change method [Add] / [Change], then click the [OK] button. You can not set both [Add] and [Change] for the same element.

You can cancel the registration of a position variable for change with the [<-] button.

7.6.4.14. Renumbering

The renumbering function can only be used with MELFA-BASIC IV and Movemaster commands.
You can renumber line numbers in a batch. You can specify the range for renumbering.
With the setting dialog, you can specify the start and end lines numbers, the new starting line number, and the line number interval.

While editing a program created with MELFA-BASIC IV or Movemaster commands, click the tool bar [Tool] -> [Renumber]. The renumber set screen is displayed.

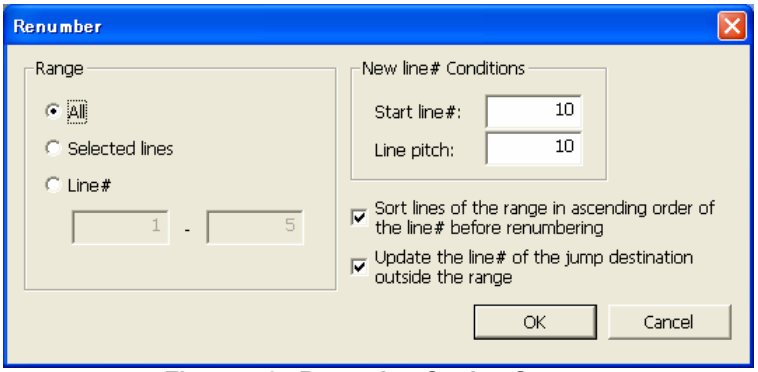


Figure 7-34 Renumber Setting Screen

Caution

Renumbering can only be used with MELFA-BASIC IV and Movemaster commands.

The renumbering function can only be used with MELFA-BASIC IV and Movemaster commands. It can not be used with standard MELFA-BASIC V.

7.6.4.15. Sorting

The sorting function can only be used with MELFA-BASIC IV and Movemaster commands.
This sorts the edited program by line number.
While editing a program created with MELFA-BASIC IV or Movemaster commands, click the tool bar [Tool] -> [Sort]. The confirmation message is displayed, then the line numbers are sorted in ascending order.

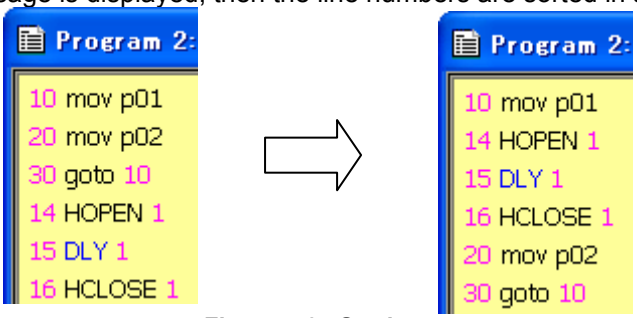


Figure 7-35 Sorting

Caution

Sorting can only be used with MELFA-BASIC IV and Movemaster commands.

The sorting function can only be used with MELFA-BASIC IV and Movemaster commands. It can not be used with standard MELFA-BASIC V.

7.6.4.16. Comment Selection/Uncomment Selection

The selected lines are exchanged as a comment by the batch. Or the comments in the selected lines are removed comment by the batch.

This function can be used in Ver. 1.2 or later.

In command edit screen, after selecting the lines you want to change into the comment, click [Edit]->[Comment Selection] on menu bar. Comment character " ' " is added to the head of the selected lines.

Moreover, after selecting the lines you want to release the comment, click [Edit]->[Uncomment Selection] on menu bar. Then the comments in the selected lines are removed comment.

However, even if you select the line where "Rem" command is included, "Rem" is not removed.

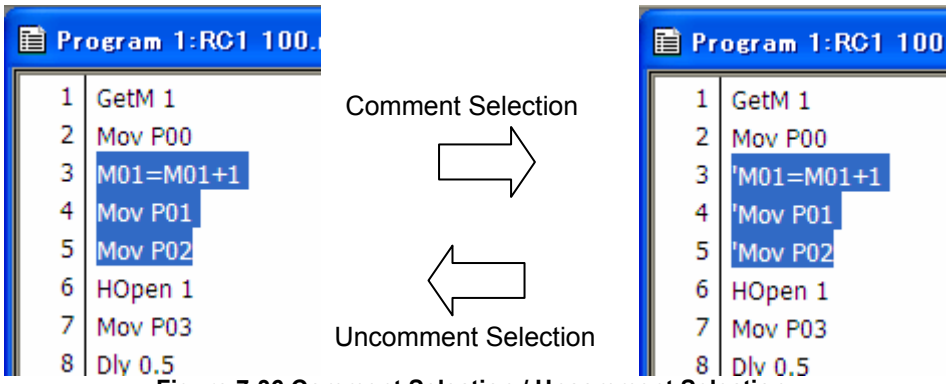


Figure 7-36 Comment Selection / Uncomment Selection

Example for setting / removing of comment is as follows;

Table 7-3 Example for setting / removing comment

program language	Set comment	Remove comment
MELFA-BASIC V	1 Mov P1 2 Mov P2 3 'Mov P3 4 REM ABCDEFG ⇨ 1 'Mov P1 2 ' Mov P2 3 ''Mov P3 4 'REM ABCDEFG	1 'Mov P1 2 ' Mov P2 3 ''Mov P3 4 'REM ABCDEFG 5 REM P4 ⇨ 1 Mov P1 2 Mov P2 3 'Mov P3 4 REM ABCDEFG 5 REM P4
MELFA-BASIC IV	10 MOV P1 20 MOV P2 30MOV P3 40 REM ABCDEFG ⇨ 10 'MOV P1 20 ' MOV P2 30 'MOV P3 40 'REM ABCDEFG	10 'MOV P1 20 ' MOV P2 30 'MOV P3 40 'REM ABCDEFG 50 REM MOV P4 ⇨ 10 MOV P1 20 MOV P2 30 MOV P3 40 REM ABCDEFG 50 REM MOV P4

⚠ Caution

Caution for comment in the robot program in debugging status

When the program is opened in debugging status, it is written as soon as the comment is set or removed. At this time, when the robot controller is driving, it becomes an error.

7.7. Saving Programs

Always save the edited program.
There are three methods for saving: saving, saving to computer, saving to robot controller.

Caution

Do not write a program to the controller with duplicate step numbers.

Even programs with duplicate step numbers or that are still being edited and are not yet syntactically correct can be saved on the computer. However, be careful not to copy or otherwise transfer such a program to a robot controller with the program management functions.

If a program with duplicate step numbers is written to the robot controller, the duplicate step lines are written over each other.

7.7.1. Save

Saves the program being edited with its current name.
When you click on the menu bar [File] -> [Save], the program is saved, overwriting the older version of itself.

If you are editing a program on a robot controller, the "Save Items" setting screen is displayed. Set the items to write, then click the [OK] button.
For details on the "save items", see**"7.7.4 Items written when saving in robot".**

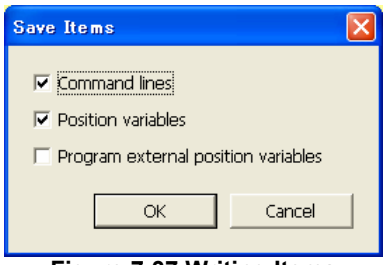


Figure 7-37 Writing Items

7.7.2. Saving on computer

Saves the program being edited to the computer. At this time, you can set a new program name.
Click on the menu bar [File] -> [Save to PC].

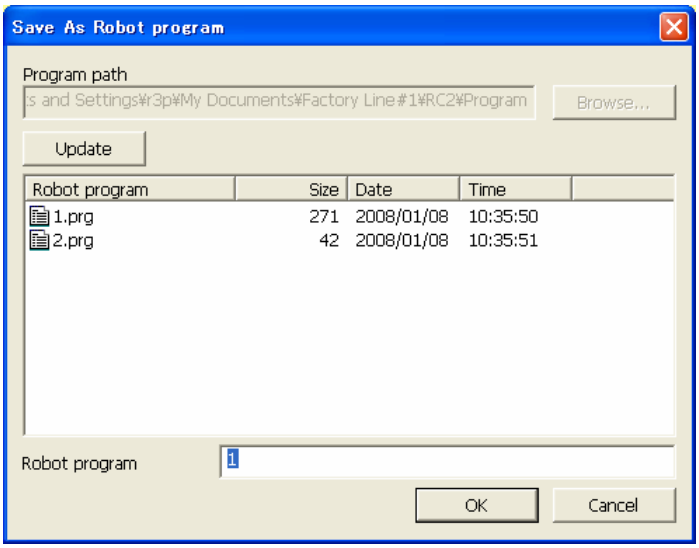


Figure 7-38 Saving on Computer

Input the robot program name, then click the [OK] button.

Caution

Folder in which programs are stored

Programs on the computer are managed in units of workspace projects. The folder they are stored into is workspace writing folder/project name/Program.

To store into any other folder, first store in this folder, then copy into the desired folder with the program management copy function.

Caution

About the program name which is disable on PC

Windows, error is occurred in the Program editing tool when opening that program. To solve this problem, it is necessary to change the program name in the robot controller.

The “Reserved words” are the special words used by Windows system. Therefore, it is impossible to use these words as the file name on PC. The following words are in “Reserved words”.

AUX, COM1 to 9, CON, LPT1 to 9, NUL, PRN

7.7.3. Saving in robot controller

Saves the program being edited to the robot controller with a new name. At time, you can set a new program name.

Click on the menu bar [File] -> [Save in Robot]. The "Save Items" setting screen is displayed, so set the save items, then click the [OK] button.

For details on the "save items", see **"7.7.4 Items written when saving in robot"**.

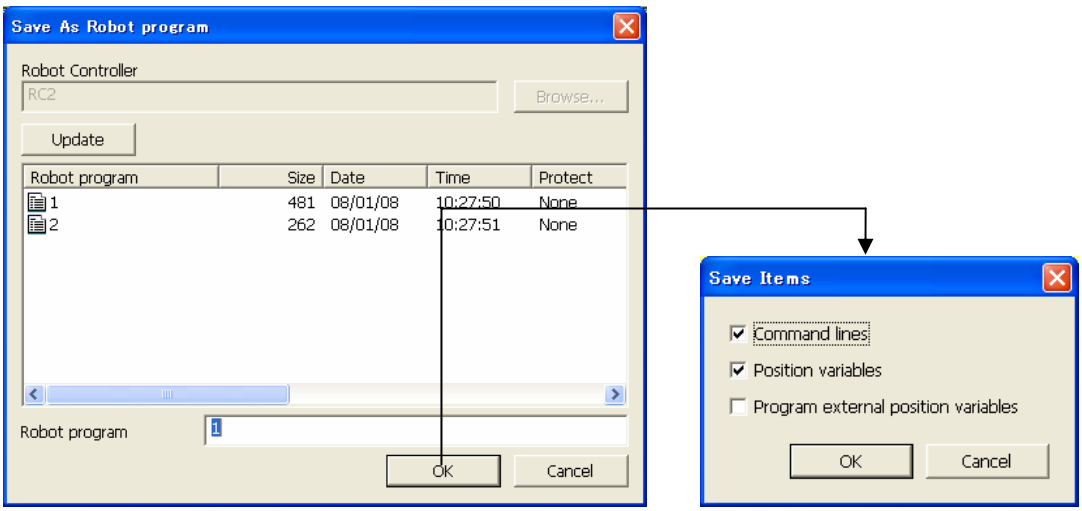


Figure 7-39 Saving on Robot Controller

7.7.4. Items written when saving in robot

When saving a robot program in a robot controller, write items can be set by categorizing them into instructions, position variables and program external position variables.

This function is displayed on CRn-500 series robot controllers from the H1 edition on.

This save item default values for when you have read a robot controller program are the same as the "Read Items" when you read the program. When you have created a new program or opened a program on the computer, the display becomes as in **"Save Items".**

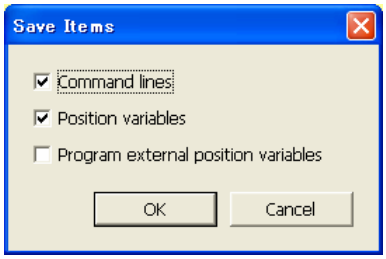


Figure 7-40 Save Items

We will explain the operations for saving to a robot when only command lines or only position variables are specified, using the following example for illustration.

Example: When there are programs in program editing on the computer or in the robot controller

<<Program in the robot controller>>

```
10 Mov   P1
20 Mov   P2
30 Mov   P3
40 End
P1=( 400.000,     0.000, 100.000, , , 90.000)(4,0)
P2=(     0.000, 400.000, 150.000, , ,    0.000)(4,0)
P3=(-351.704, -49.369,   22.000, , ,-95.168)(0,0)
P4=( 276.499,-599.066, 264.966, , , 29.170)(0,0)
```

<< Program on computer >>

```
100 Mov   P1
200 Mov   P2
300 Mov   P5          ' <- Change
400 End
P1=( 400.000,     0.000, 100.000, , ,-90.000)(4,0)
P2=(     0.000, 400.000, 150.000, , ,    0.000)(4,0)
P3=(-351.704, -49.369,   22.000, , ,-95.168)(0,0)
P5=( 535.786, 295.021, 102.000, , ,148.420)(0,0)
```

(1) When only command line written

<<Program in the robot controller>>

```
100 Mov   P1
200 Mov   P2
300 Mov   P5
400 End
P1=( 400.000,     0.000, 100.000,,, 90.000)(4,0)
P2=(     0.000, 400.000, 150.000,,,    0.000)(4,0)
P3=(-351.700, -49.370,   22.000,,,-95.170)(0,0)
P4=( 276.500,-599.070, 264.970,,, 29.170)(0,0)
```

Changing only the command line

The position variables are not changed.

(2) When writing position variable only

<<Program in the robot controller >>

```
10 Mov   P1
20 Mov   P2
30 Mov   P3
40 End
P1=( 400.000,     0.000, 100.000,,,-90.000)(4,0)
P2=(     0.000, 400.000, 150.000,,,    0.000)(4,0)
P3=(-351.700, -49.370,   22.000,,,-95.170)(0,0)
P4=( 276.500,-599.070, 264.970,,, 29.170)(0,0)
P5=( 535.790, 295.020, 102.000,,,148.420)(0,0)
```

The command line is not changed.

The position variables on the computer are overwritten. However, position variables that are in the robot controller, not on the computer, are left as is. (Reference)

7.7.5. Setting the syntax check for before program saving

You can set whether or not to have the syntax checked when you save a program and whether or not to display a message when there are no syntax errors.

With the program opened, on the menu bar, click [Tool] -> [Option] and set with the option screen.

The default setting is automatic syntax checks with no message displayed if there is no syntax error.

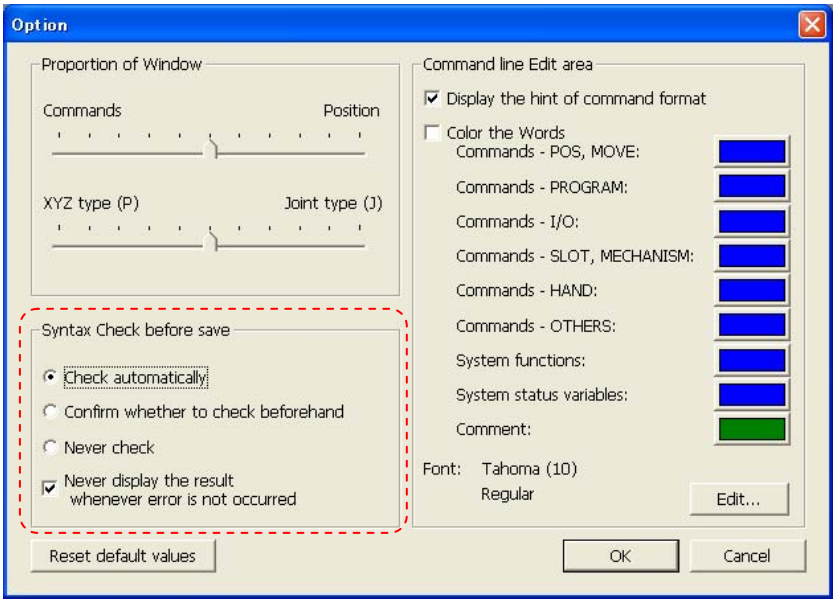


Figure 7-41 Settings for Syntax Check Before Saving

7.8. Program Printing

You can print programs you have written.

7.8.1. Checking a print image

You can display a print image of the program on the screen.
Make active the program you want to print, then click on the menu bar [Workspace] -> [Print Preview]. The print image for currently active program is displayed.

7.8.2. Printing a program

Make active the program you want to print, then click on the menu bar [Workspace] -> [Print]. The currently active program can be printed.

7.8.3. Setting to print a program

You can customize the pages the program is printed with.
When you click on the menu bar [File] -> [Page Setup], the page setup screen is displayed.
You can set whether or not to print the file name, print date and time, and page numbers, the space between lines and the margin sizes.

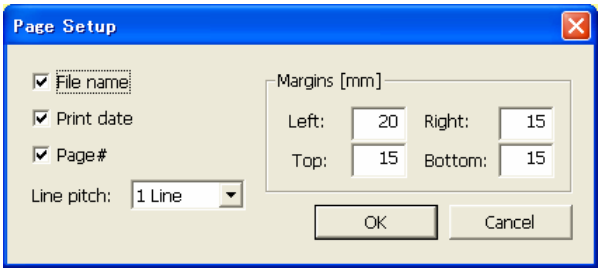


Figure 7-42 Page Setup for Printing

7.9. Program Debugging

You can debug robot programs you have written.

Caution

Debugging is for programs on a controller or on a virtual controller with a simulation running.

When debugging, use a program on a controller or on a virtual controller with a simulation running. You can not debug a program stored on a computer.

7.9.1. Starting debugging

Open the robot program in debugging status. From the project tree [Online] -> [Program], select the program, then click the right mouse button. From the right mouse button menu, click [Debug Open].

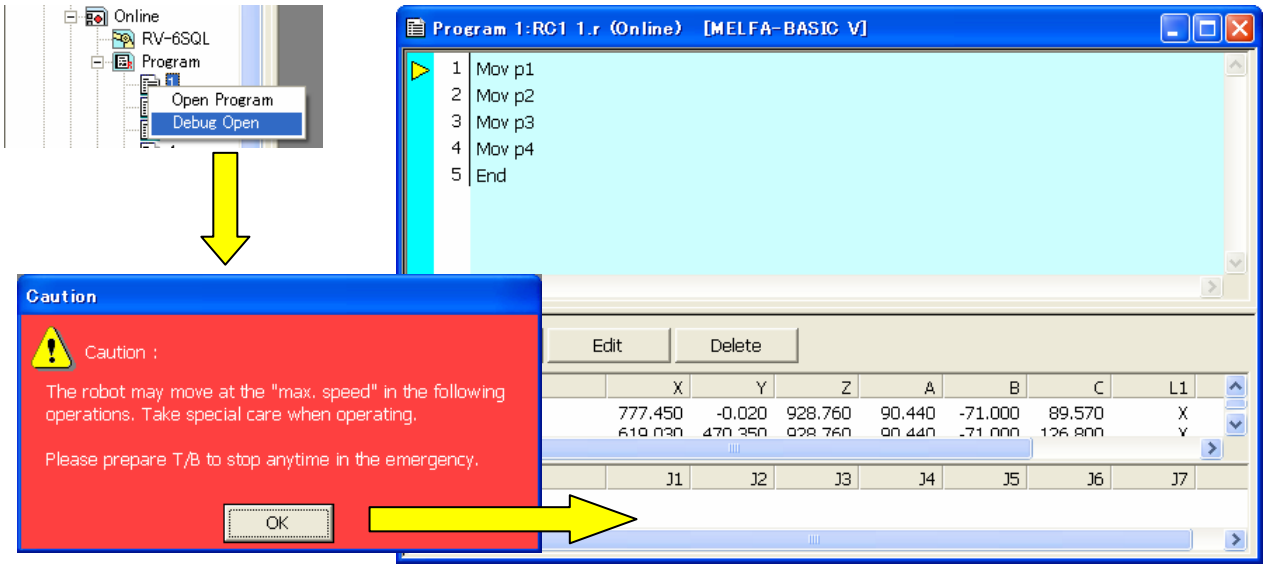


Figure 7-43 Figure 7-2 Opening a Program in Debug Status

The specified robot program is opened in debugging status. The execution line cursor "▶" is displayed at the left end of the command edit area. The line on which this execution line cursor is displayed is the line currently being executed.

The display of this execution line cursor can be switched on/off with the menu bar [Debug] -> [Display/Do not display Execution Line].

7.9.2. Executing programs step by step

Danger

With program debugging, the robot may operate at 100% speed.
Watch out for the safety around the robot.
Also, prepare a T/B at hand and use the robot in a status in which an emergency stop can be made at any time.

A program that has been opened in debugging status can be run step by step.
Operate using the debug operation screen that is displayed at the same time the program was opened in debugging status.

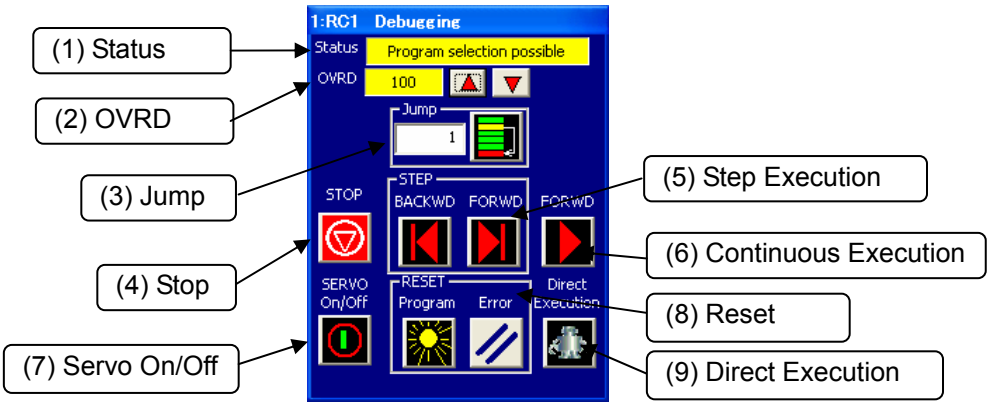


Figure 7-44 Debug Operation Screen

(1) Status
The controller's task slot status is displayed.

(2) OVRD
This displays and sets the robot speed override.

(3) Jump
You can specify the line in the program to execute.

(4) Stop
You can stop the program.

(5) Step Execution
This executes the specified program one line at a time. Pressing the [FORWD] button executes the command on the current execution line and advances the current execution line one line. Pressing the [BACKWD] button executes the command on the current execution line and returns the current execution line one line.

(6) Continuous Execution
This executes the program continuously from the current line.

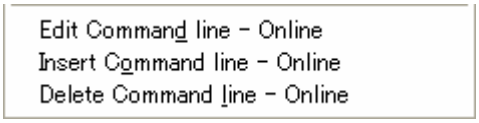
(7) Servo On/Off
You can switch the robot servo On/Off.

(8) Reset
You can reset the program and any errors that have occurred.

(9) Direct Execution
You can execute any command without relationship to the robot program.

7.9.3. Revising programs

The command statements for a program that has been opened in debugging status can not be edited in the command edit area. You can revise command statements from [Edit] on the menu bar. Click on the menu bar [Edit] -> [Edit Command line (Online)], [Insert Command line (Online)], and [Delete Command line (Online)]. Position variables can be edited as usual.



(1) Edit command line

You can edit the contents of the specified command line.

Click the command line to be edited with the mouse, click on the menu bar [Edit] -> [Edit Command line (Online)]. The screen for editing the command line is displayed.

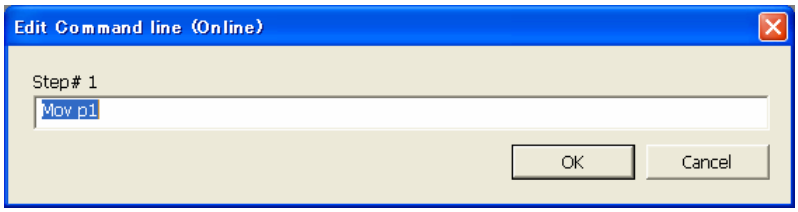


Figure 7-45 Command Line Editing (Online)

Revise the command line, then click the [OK] button.

(2) Insert command line

You can insert a command statement at the specified line.

Click the line at which the command statement is to be inserted with the mouse, then click on the menu bar [Edit] -> [Insert Command line (Online)]. The screen for inserting the command line is displayed.

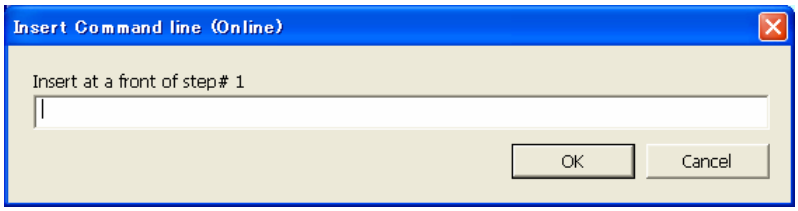


Figure 7-46 Command Line Insertion (Online)

(3) Delete command line

You can delete the specified command line.

Click the line at the line with the command statement to be deleted with the mouse, then click on the menu bar [Edit] -> [Delete Command line (Online)]. The confirmation screen for the command line deletion is displayed.

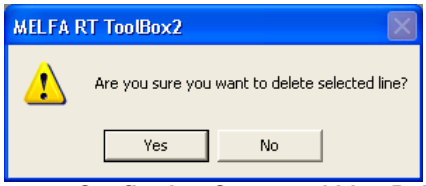


Figure 7-47 Confirming Command Line Deletion

(4) Edit position variables
For details on the method for editing the position variable, **"7.6.3 Position variable editing".**

Caution

Be careful when changing the value of a variable.

When you change the value of a variable, the operation target position of the robot may change and result in a collision. This is particularly dangerous during robot operation, so check carefully before changing the value of a parameter.

Caution

Partial writing can not be performed while editing a program in debugging status.

7.9.4. Setting and deleting breakpoints

You can set a breakpoint in a program that has been opened in debugging status.
If you set a breakpoint, when you open the program in debugging status, you can stop the program at the line while executing the Continuous execution. After stops, you can execute the program continuously.

Breakpoints can be set up to 128. Moreover, when the program is quitted, every breakpoint is deleted.
There are the following two types of breakpoints.

Permanent breakpoint	:	After stopping, the breakpoint keeps being set.
One-time breakpoint	:	After stopping, the breakpoint is automatically deleted at the same time as stopping.

Caution

Breakpoints can only be used with MELFA-BASIC V.

(1) Set a Breakpoint
The breakpoint is set according to the following procedure.
1) Click the command line where breakpoint is set with the mouse, then click on the menu bar [Debug] -> [Set Breakpoint].

2) The breakpoint setting screen is displayed.
Select the type of breakpoint to set, then click the [OK] button. The breakpoint is set at the specified command line.

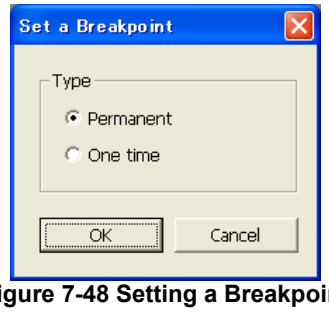


Figure 7-48 Setting a Breakpoint

3) "●" is displayed at the left end of command lines at which breakpoints are set.

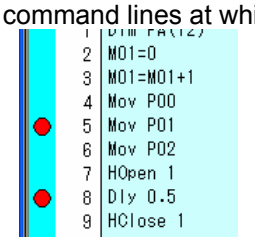


Figure 7-49 Display of Lines with Breakpoints Set

(2) Delete a Breakpoint

To delete a breakpoint, click the command line with the breakpoint to be deleted with the mouse, then click on the menu bar [Debug] -> [Delete a Breakpoint].

To delete all the breakpoints set in this program, click on the menu bar [Debug] -> [Delete All Breakpoints].

You can also perform the breakpoint setting and deleting operations with tool bar buttons.

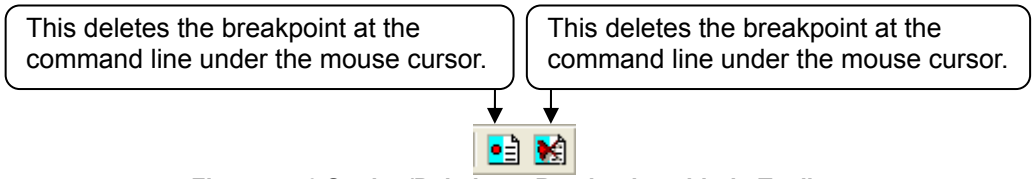


Figure 7-50 Setting/Deleting a Breakpoint with theToolbar

7.9.5. Ending debugging

To end debugging, close the program with the "×" button at the top right of the edit screen for the program opened in debugging status.

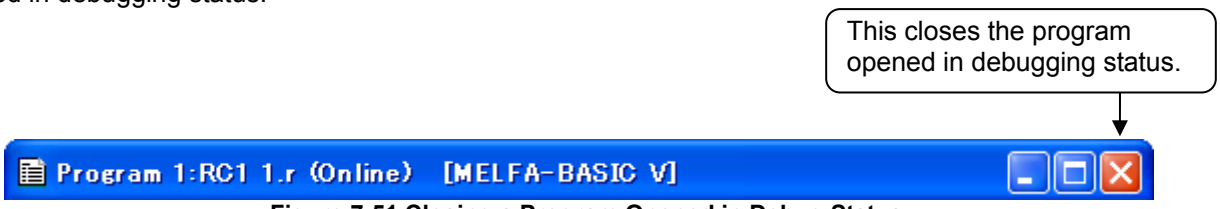


Figure 7-51 Closing a Program Opened in Debug Status

At this time, if the program has been changed, a confirmation message is displayed asking if you want to save the changed contents.

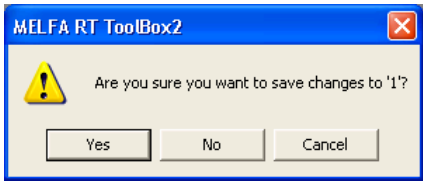


Figure 7-52 Confirmation Message for Saving the Changed Contents

Here, if you select "No", the changed contents are all thrown out. To put the changed contents into effect, always select "Yes".

7.10. Program Management

You can copy, move, delete, compare the contents of, rename, and set protection for robot programs.

From the project tree, select the target project program, then click the right mouse button.
The right button menu is displayed, so select [Manage Programs]. The manage programs screen is displayed.

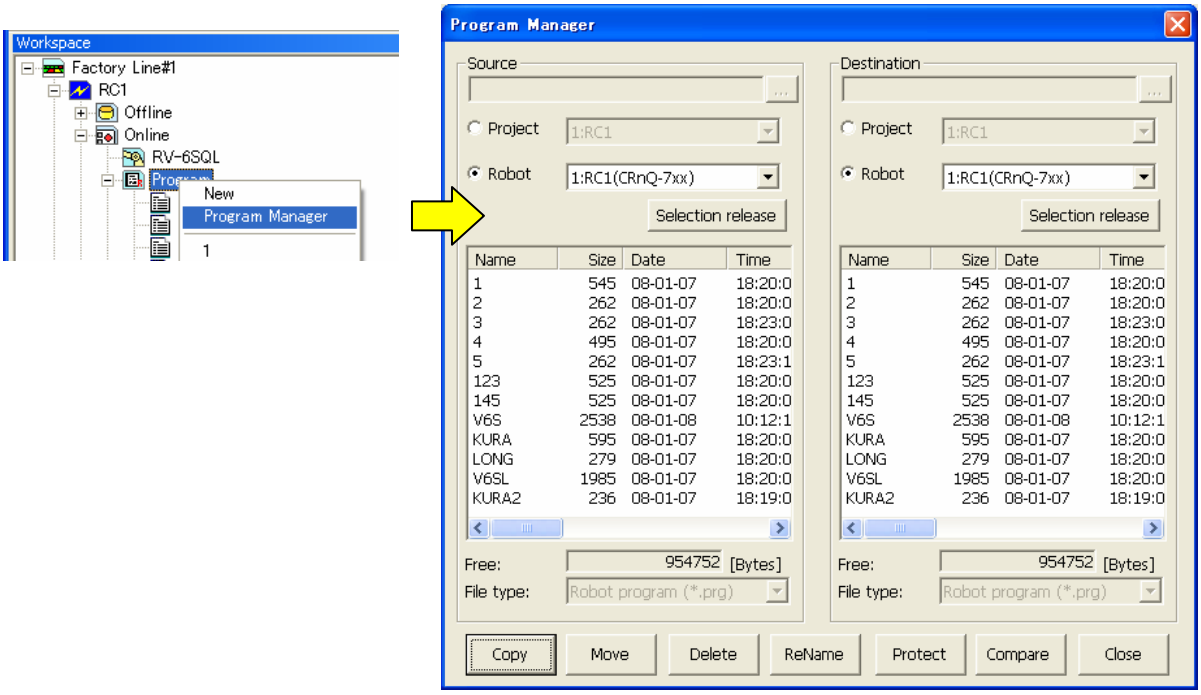


Figure 7-53 Starting Program Management

Caution

All the operations of these functions are for robot programs.

7.10.1. Program list display

On the left and right lists, the lists are programs of the displayed in the robot controller and the specified folder.

① Project　　You can specify projects in the workspace.
② Robot　　You can specify a robot controller that is currently connected.
③ […] button　　When you select [Project], you can specify any folder on the computer.

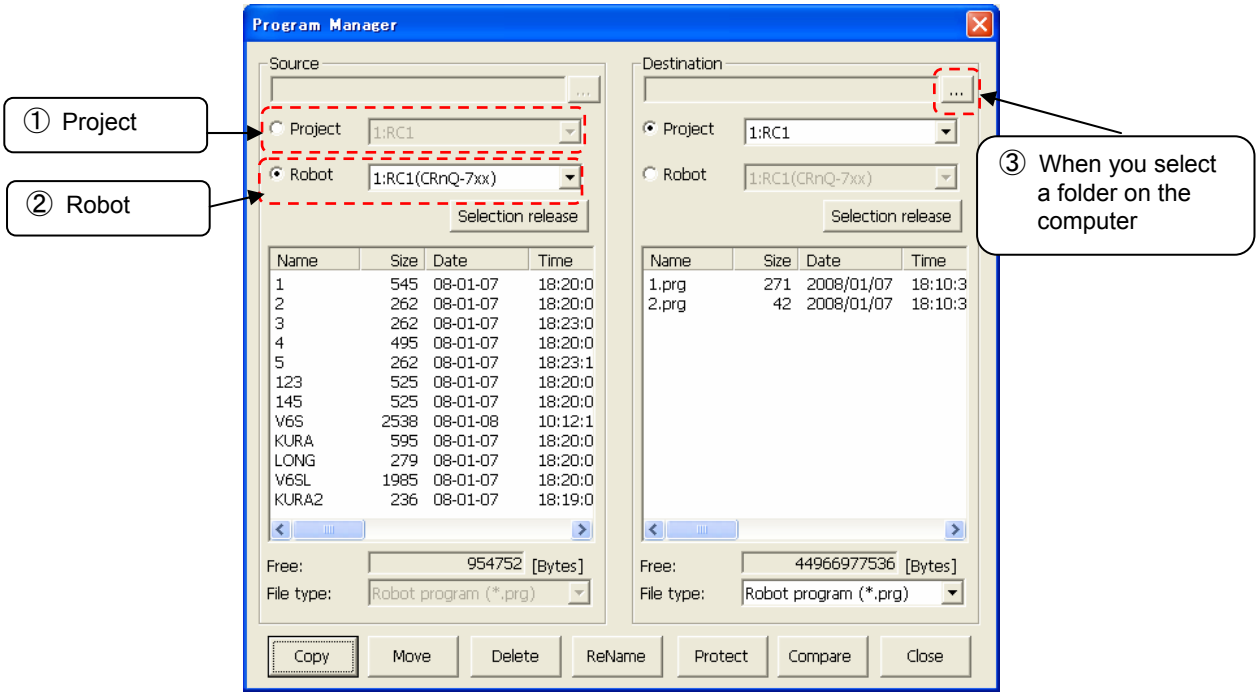


Figure 7-54 Program List Display

7.10.2. Copy

The program files are copied. Copying of the entire program file or only the command statements or only the position variables is possible.

Select the transmission source program names from the list at the left, and designate the transmission destination folder on the right side. The multiple transmission source programs can be selected at the same time, but for copying with changing its name, only one program must be selected. Copying is executed when the [Copy] button is clicked on and [Setting for copy] dialog is set.

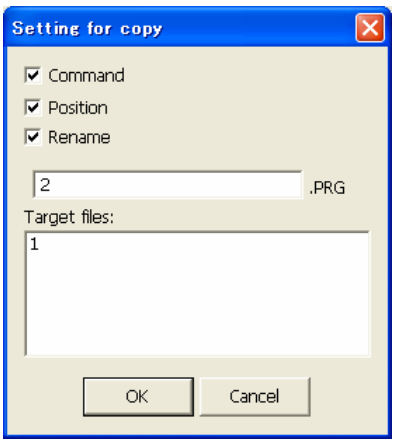


Figure 7-55 Copy Setting Screen

7.10.3. Move

The program files can be moved.

Select the transmission source program names from the list at the left, and designate the transmission destination folder on the right side. The multiple programs can be selected at the same time. Movement is executed when the [Move] button is clicked on.

7.10.4. Delete

The program files can be deleted.

Select the names of the programs to be deleted from the lists. The multiple programs can be selected at the same time. The programs can be selected at the both lists. Delete is executed when the [Delete] button is clicked on.

⚠ Caution

That once the program files are deleted, they cannot be recovered.

7.10.5. Rename

A program file name is renamed.

Select the name of the only one program to be renamed from the lists. The program can be selected at the both lists. Rename is executed when the [Rename] button is clicked on and a new file name is set at the [Setup for ReName] dialog.

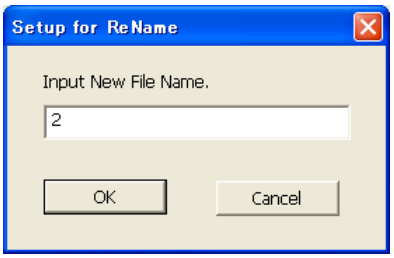


Figure 7-56 Rename Screen

7.10.6. Protect settings

The program files in the controller can be protected. The entire program file can be protected, or just the command statements or position variables can be protected.

You cannot move, delete, or rename a protected file. Release the protection before any of these operations.

Select the names of the programs to be protected from the lists. The multiple programs can be selected at the same time. The programs can be selected at the both lists. Protect is executed when the [Protect] button is clicked on and [Setting for protect] dialog is set.

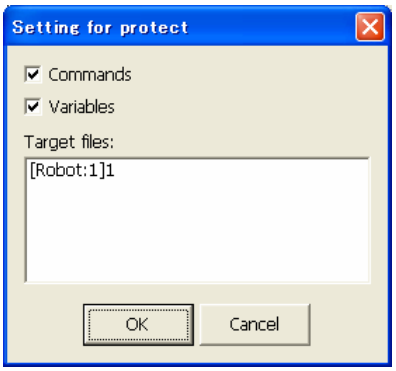


Figure 7-57 Protect Settings Dialog

Caution

The only programs to which protect operations apply are programs in robot controllers.

7.10.7. Comparison

The program files can be compared. Comparison of only the command statements or only the position variables is possible. Select the names of the programs to be compared from the left and right lists. A dialog displaying the corresponding comparison results will appear when the [Compare] button is clicked on and [Setting for compare] dialog is set.

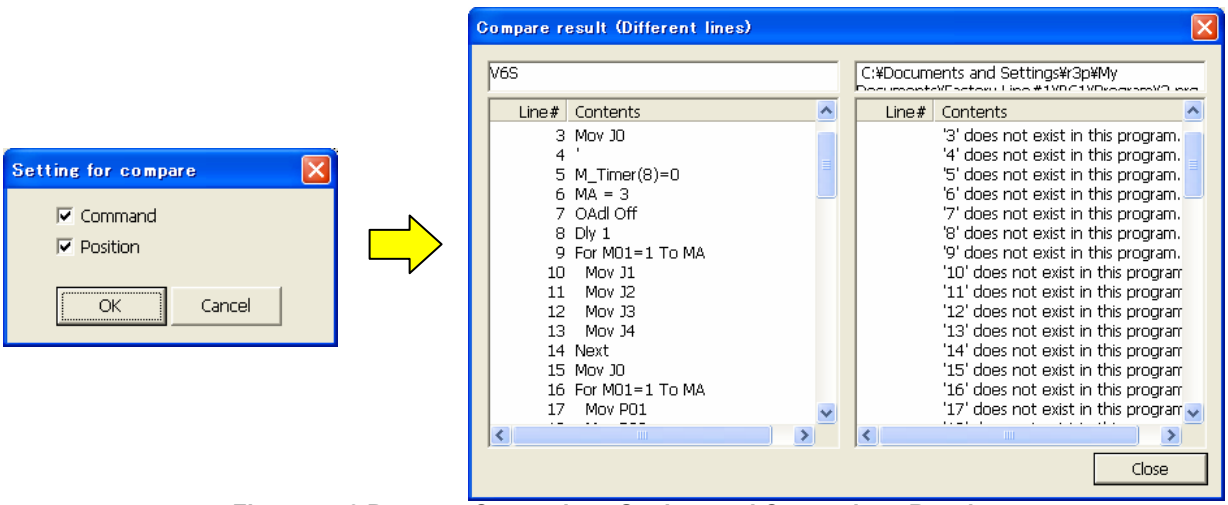


Figure 7-58 Program Comparison Settingsand Comparison Results

When both files are the same, the result dialog displays nothing.

7.11. Program Conversion

You can convert existing robot programs written in a different program language into the currently set program language.

Table 7-4 Combination list of program conversion

No.	Source	Target	Content
1	MELFA-BASIC III	MELFA-BASIC IV	The position data is converted. If “MELFA-BASIC V” is chosen for the target, the lines No. are also converted in addition to this conversion.
2	MELFA-BASIC III	MELFA-BASIC V	
3	MELFA-BASIC IV	MELFA-BASIC V	The lines No. are converted.
4	MOVEMASTER command (CR-116/356)	MOVEMASTER command (CRn-500/700)	The position data is converted.

There are the following two types of conversion.
For details, see **"7.11.2 Line number conversion (from MELFA-BASIC IV to MELFA-BASIC V", "7.11.3 Position data conversion (from E/EN/M1/M2 series to CRn-500/700 series"**.

(1) Conversion of the lines No.
Convert the line No. and relevant command line, because the deal of line No. in MELFA-BASIC IV is different from MELFA-BASIC V.

(2) Conversion of the position data
Convert the format of position data and relevant command, because the configuration of the E/EN/M1/M2 E/EN/M1/M2 series controller’s position data is different from the CRn-500/700 series controller’s.

⚠ Caution

Only programs on the computer can be converted.

Program conversion is only possible for programs on the computer. It is not possible to directly convert a program on a controller or on a virtual controller with a simulation running. To convert a program on a controller or on a virtual controller with a simulation running, first use program management to copy it onto the computer, then convert it there.

7.11.1. Starting program conversion

From the project tree, select the conversion destination project with [Offline] -> [Program], then click the right mouse button. From the right mouse button menu, click [Program Convert].
The destination is in the program language set for this project.

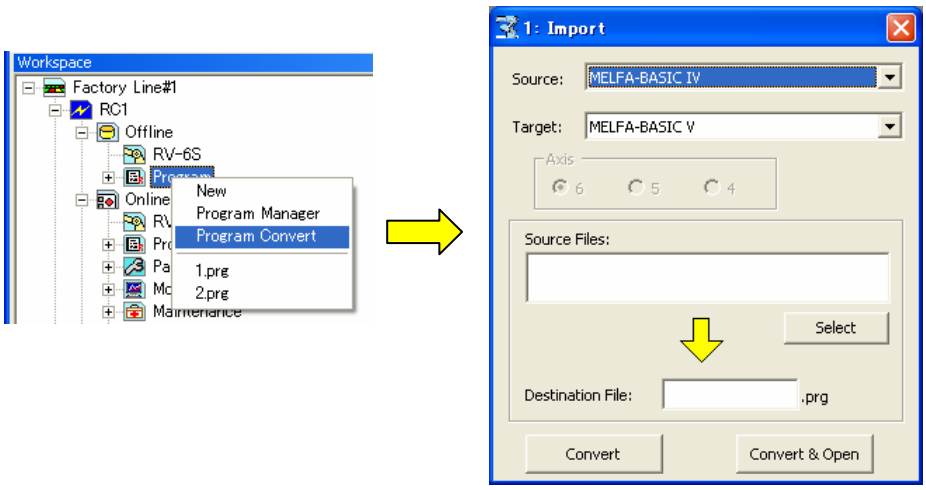
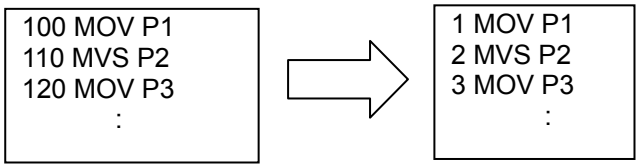


Figure 7-59 Starting Program Conversion

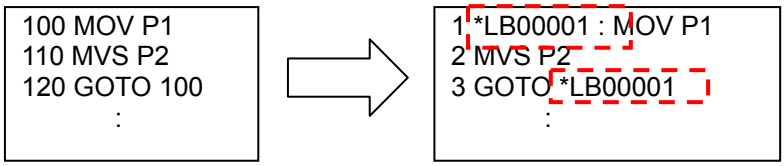
7.11.2. Line number conversion (from MELFA-BASIC IV to MELFA-BASIC V)

Convert the lines No. and relevant command line, because the deal of line No. in MELFA-BASIC IV is different from MELFA-BASIC V.

For the MELFA-BASIC V, convert as follows.
- The line No. is converted to the step No. (sequential No. which starts from 1).



- The command line using line No. jump is converted to the command line using label jump.



*LB00001 is the label name created automatically by this conversion.

Convert the program according to the following procedure.

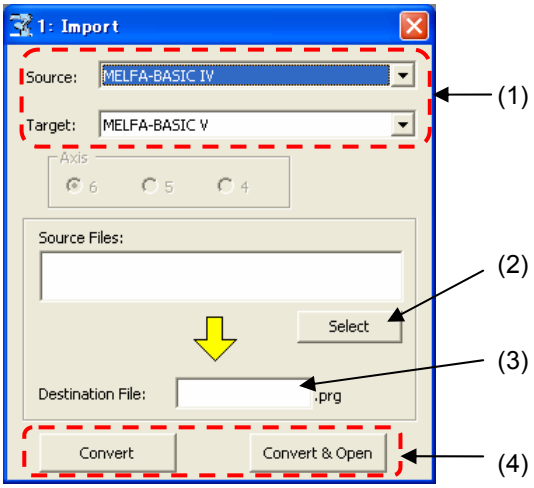
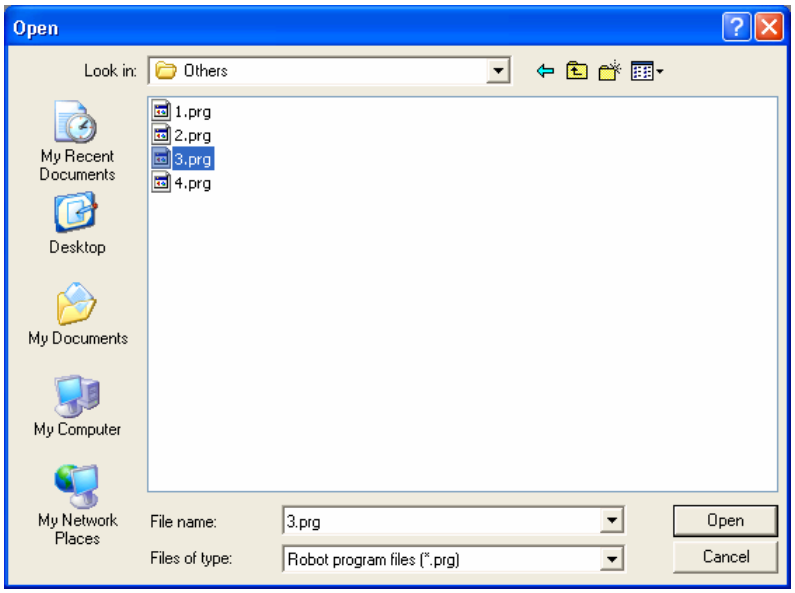


Figure 7-60 Conversion from MELFA-BASIC IV to MELFA-BASIC V

(1) Select the program language of the source and target for conversion.
Start program conversion with the project set with "MELFA-BASIC V"
Choose “MELFA-BASIC IV” for the source and choose “MELFA-BASIC V” for the target.
It is NOT necessary to select the axis of robot.

(2) Select the file as source.
Click the [Select] button of the source, and select the file of MELFA-BASIC IV program.



(3) Input the name of the file to save the converted program into.

(4) After designating the conversion source file and the conversion destination file, click the [Convert] button or the [Convert & Open] button.
When [Convert] button is clicked, the designated selected file is converted and written in the target file.
When [Convert & Open] button is clicked, the file is converted, written in the target file and opened through at Program edit tool.

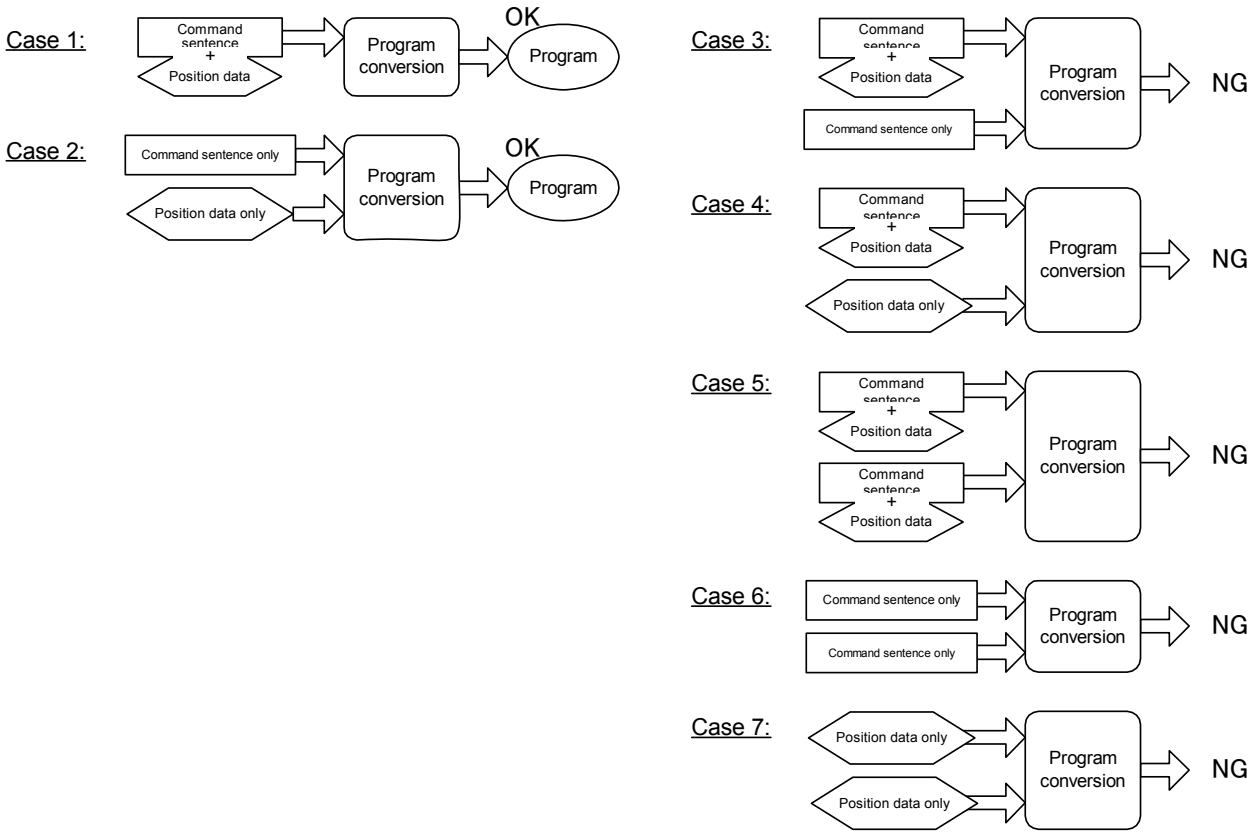
7.11.3. Position data conversion (from E/EN/M1/M2 series to CRn-500/700 series)

This function converts the format of position data and relevant command (DJ, MP and PD command of MOVEMASTER command), because the configuration of the E/EN/M1/M2 series controller's position data is different from the CRn-500/700 series controller's.

Note1: The program conversion converts the position data as well as the commands related to the position data.
It is not possible to convert commands automatically. Be sure to make grammatical check using this software before using the program of E/EN/M1/M2 series by the CRn-500/700 series, and change the commands if necessary.

Note2: The position data of the MOVEMASTER command program for M1/M2 series has the base-coordinate rotated 90º. When using the program for M1/M2 by the CRn-500/700 series, convert M1/M2 program into EN program with the E/EN/M1/M2 series support software DOS version, before carrying out the program conversion.

Note3: In the case of the E/EN/M1/M2 series support software DOS version and E/EN series Robot programming supporter for Windows, the data can be saved separately such as command sentence only, position data only or command sentence and position data. In this program conversion, it is possible to convert the file of command sentence only and the file of position data only to one program. The other combinations occurs error. (As following case 3 to 7.)



Convert the program according to the following procedure.

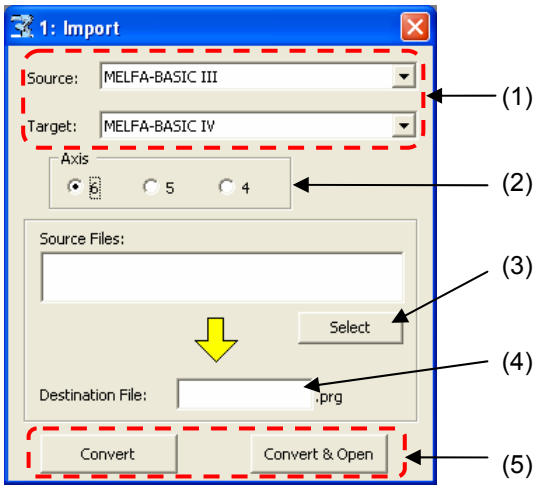
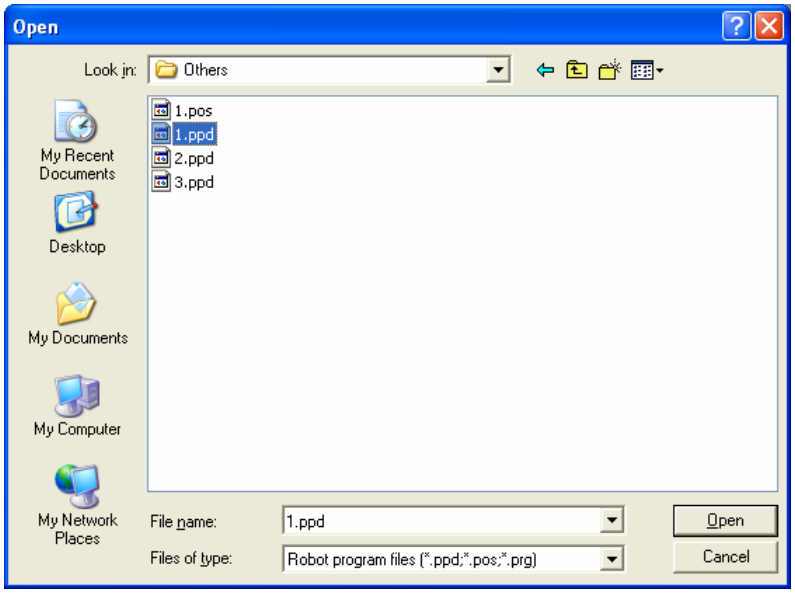


Figure 7-61 Conversion from MELFA-BASIC III to MELFA-BASIC IV

(1) Select the program language of the source and target for conversion.
Start program conversion with a project set for "MELFA-BASIC V" or "MELFA-BASIC IV".
For the source, choose “MELFA-BASIC III” or “MOVEMASTER (CR-116/356)” which is the program language of the E/EN/M1/M2 series.
If “MELFA-BASIC V” is chosen for the target, the lines No. are also converted in addition to this conversion.

(2) Select the axis of robot.
Select the correct number of axes, so that the configuration of the E/EN/M1/M2 series controller’s position data is different depending on the number of axes.

(3) Select the file(s) as source.
Click the [Select] button of the source, and select the file(s) of E/EN/M1/M2 series program.
When selecting the multiple files, click the file while pushing the [Ctrl] key.



(4) This specifies the file to write the converted program into.
Input the name of the file to save the converted program into.

(5) After designating the conversion source file and the conversion destination file, click the [Convert] button or the [Convert & Open] button.
When [Convert] button is clicked, the designated selected file is converted and written in the target file.
When [Convert & Open] button is clicked, the file is converted, written in the target file and opened through at Program edit tool.

8. Setting Parameters

You can reference and rewrite parameter information set in a robot controller.
You can set parameters with the method of specifying parameter names and setting them or with the method of making the settings arranged by function.

8.1. Editing from parameter list

You can reference and rewrite individual items of parameter information set in a robot controller by specifying the name of the parameter.

8.1.1. Starting

This is used in the state with the robot controller connected.
From the project tree, double click [Online] -> [Parameter] -> [Parameter List].

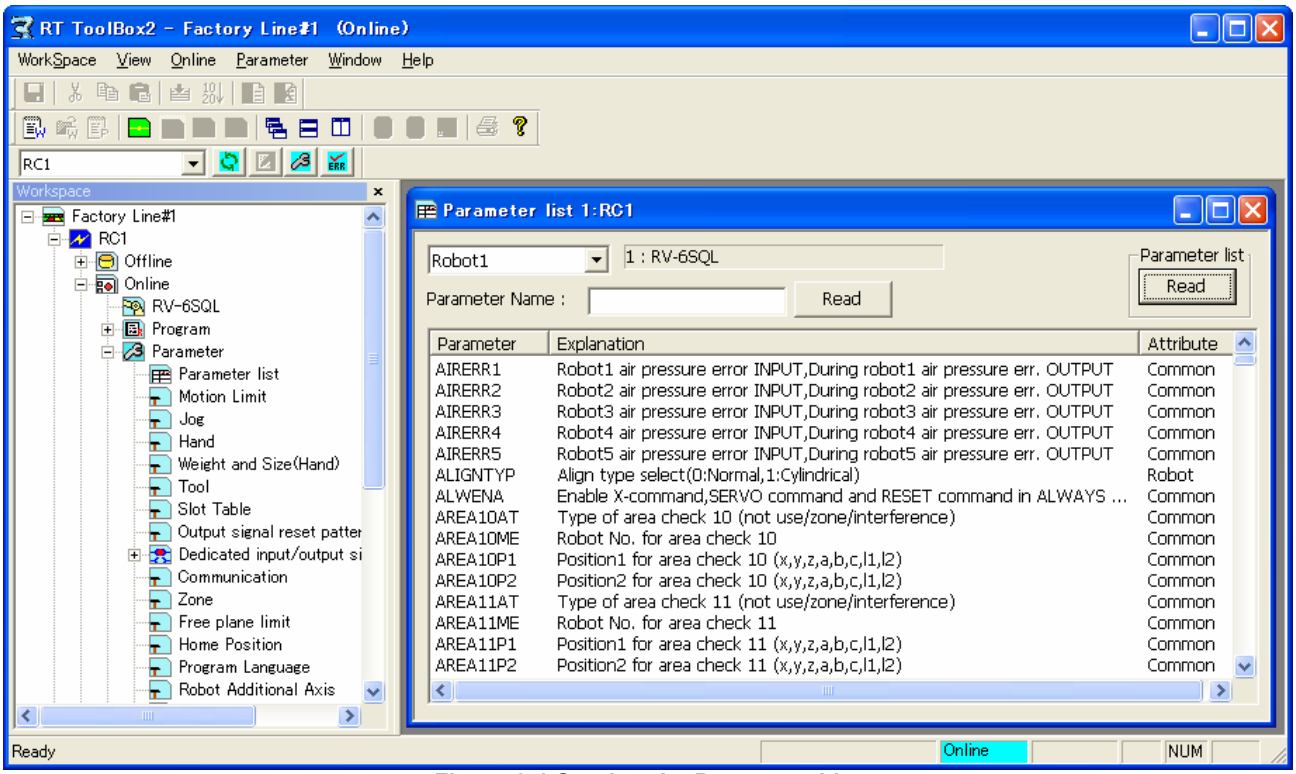


Figure 8-1 Starting the Parameter List

At this time, a confirmation message like the following concerning the parameter list is displayed.

Figure 8-2 Confirmation Message for Parameter List

This window will appear in the following cases.

- When there is no parameter list information in the personal computer.
- When the parameter list used in the robot controller is newer than the parameter list already stored in the personal computer.

Memo

What is the parameter list?

The parameter list is a list of parameter information comprised of parameter names, explanatory text, etc. displayed on the parameter list screen. This parameter list can be downloaded from the controller. The version of the parameter list may vary with the version of the software on the controller. We recommend that you download the latest parameter list from the controller.

When parameter list reading is specified, the screen for selecting where to read it from is displayed.

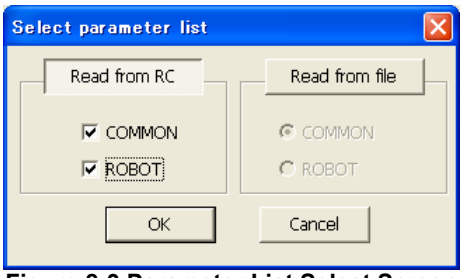


Figure 8-3 Parameter List Select Screen

Select either "Read from RC" or "Read from file", set the parameter list to be read, then click the [OK] button.

8.1.2. Parameter editing

Double click a parameter displayed in the list or input its name, then click the [Read] button. The specified parameter information in the robot controller is displayed.

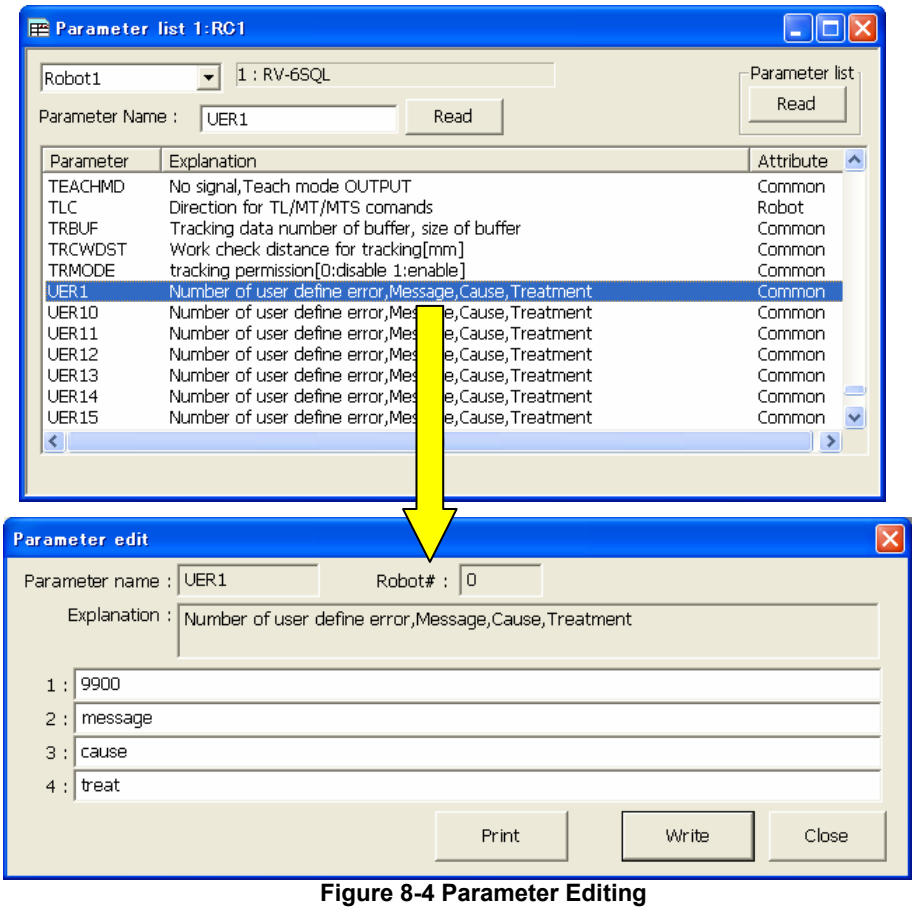


Figure 8-4 Parameter Editing

After you change a parameter, you can rewrite the specified parameter information in the robot controller by clicking the "Write" button.

You can print the displayed parameter information by clicking the [Print] button.

⚠ Caution

Use upper case letters when naming the programs in alphabetic characters.

Lower case alphabetic characters can be used in this parameter setting.
Use upper case letters when naming the programs in alphabetic characters for the parameters of the base program (PRGUSR) or slot table (SLT*), etc. All of the program names within the robot controller will be expressed in upper case letters.
If lower case letters are used, the programs will not be properly recognized.

To put a changed parameter value into effect, switch the robot controller power Off, then On again.

To validate the rewritten parameter information in the robot controller, the robot controller power must be turned ON again.

8.1.3. Edit of parameter changed from initial value

It is possible to confirm the changed parameter by displaying the list of parameter changed from an initial value. This function has the limitation in this software version and the software version of the connected controller. Please refer to the table as follows.

Table 8-1 Compliant version of this function and controller

		Software version of robot controller		
		CRnD-700	CRnQ-700	CRn-500
Version of this software	Ver.1.1 or earlier	not use	not use	not use
	Ver.1.2 or later	Ver.P6 or later	Ver.N6 or later	not use

When combination of “not use”, the button for the display switch is not displayed in the upper part of the screen

When the parameter list is displayed after connecting with the controller of corresponding version, the button for the display switch is displayed in the upper part of the screen.
When the parameter list screen is started, "All" has been selected. When "Changed" is selected, the list of the parameters that have been changed from initial value is displayed.
(However, according to the specification of the robot type, some parameters might be changed before shipment.)

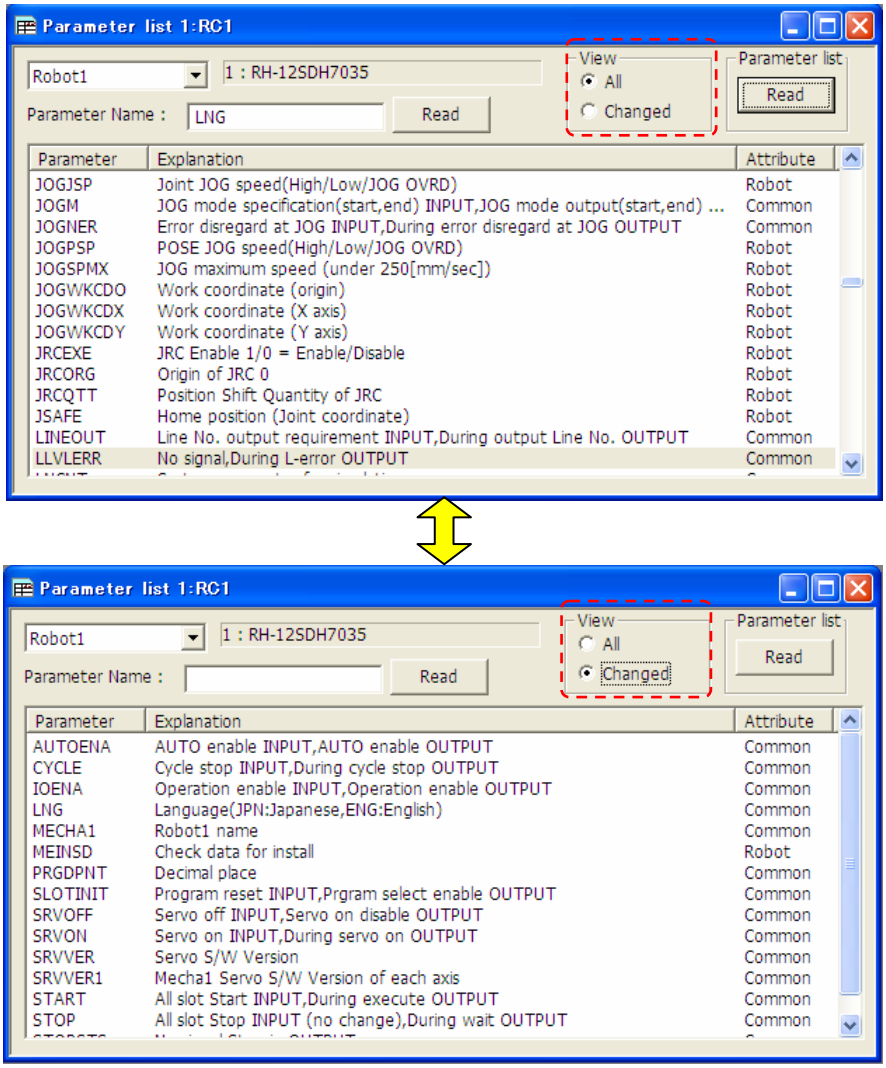


Figure 8-5 List of parameter changed from initial value

8.1.4. Parameter list reading

If no parameter name is displayed on the parameter list screen, you can read the parameter list.
Click the parameter list [Read] button at the top right of the parameter list screen. The parameter list select screen is displayed.

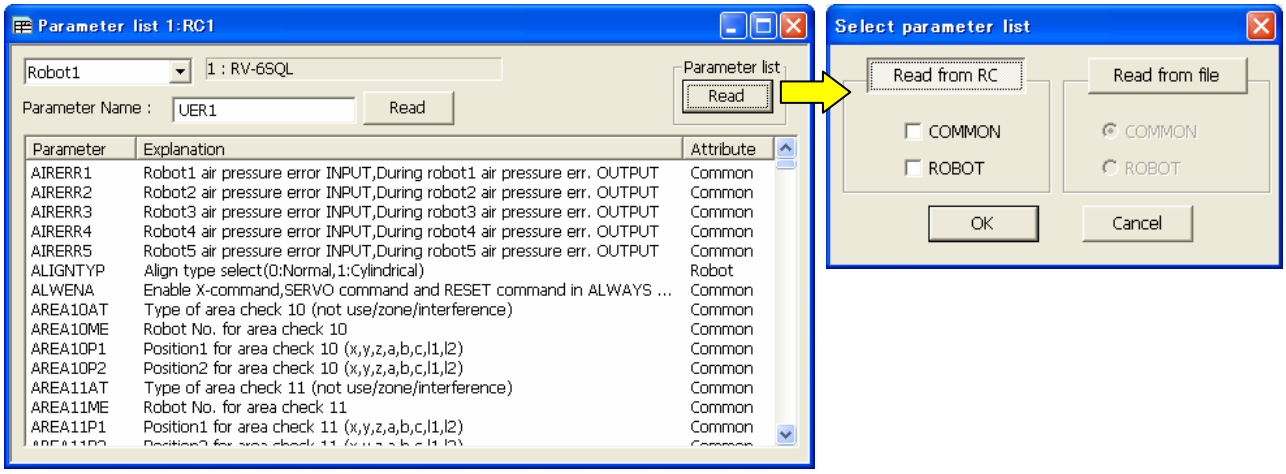


Figure 8-6 Parameter List Reading

Select the source to read from ("Read from RC" or "Read from file"), then set the parameter list to read and click the [OK] button.

8.1.5. Finding parameters

You can find a character string in the displayed parameter list. With the "Parameter List" screen active, click on the menu bar [Parameter] -> [Find]. The character string find screen is displayed.

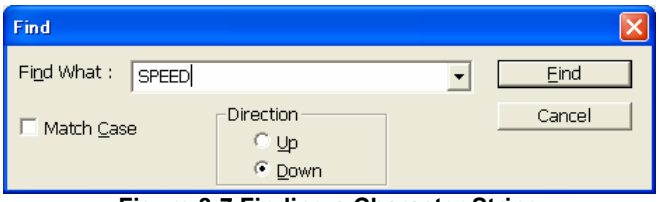


Figure 8-7 Finding a Character String

Input the character string to search for, then click the [Find] button. Parameters that include the input character string are displayed.

8.2. Robot Controller Operation Modes for Parameter Writing

With CRn-500 series robot controllers, when you write parameters to the robot controller, controller software version may place restrictions on the operating mode in which you can write. For details, see below.

However, parameters cannot be written while any program with any startup condition other than Always has been started. In such a case, stop the program, then write the parameters.

Table 8-2 Download Operating Modes for Writing (for CRn-500 series only)

CRn-500 series robot controller	Operating mode		
	TEACH	Auto (OP)	Auto (Ext)
J1 edition or later	○	○	○
H7 edition or earlier	○	×	×

○: Writing possible ×: Writing impossible

8.3. Operating Range Parameters

Set the operating range of the robot
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Operating Range].
After you change the parameter value, you can rewrite the operating range parameter in the robot controller by clicking the [Write] button.

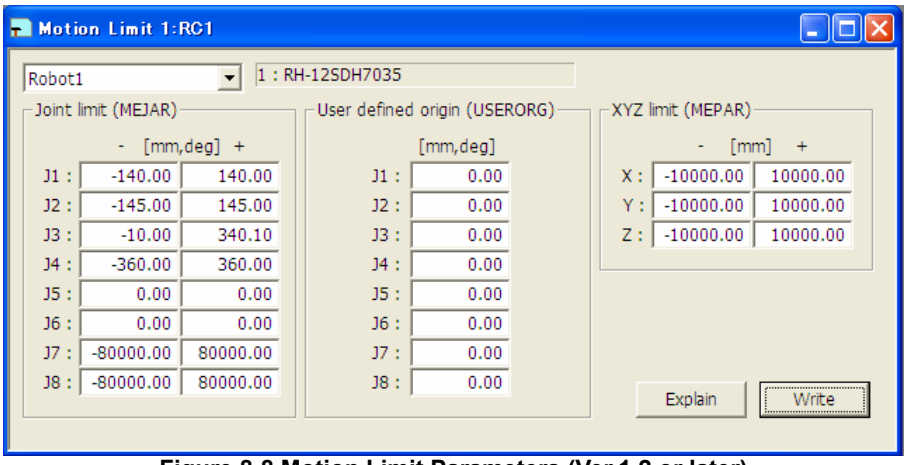


Figure 8-8 Motion Limit Parameters (Ver.1.2 or later)

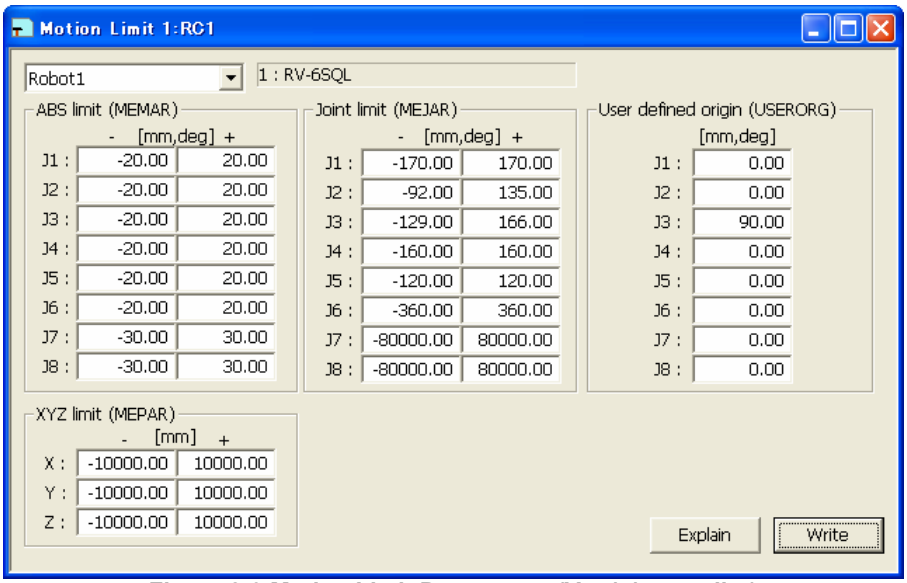


Figure 8-9 Motion Limit Parameters (Ver.1.1 or earlier)

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.4. Jog Parameters

Set the speeds for joint jogging and orthogonal jogging.
Set parameters while connected to the robot controller. From the project tree, double click the target project [Online] -> [Parameter] -> [Jog].
After you change the parameter value, you can rewrite the jog parameter in the robot controller by clicking the [Write] button.

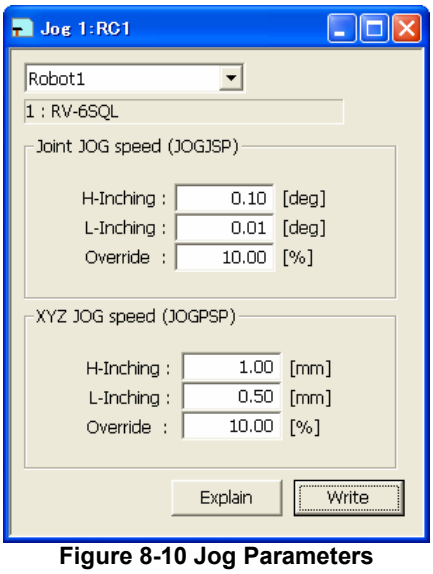


Figure 8-10 Jog Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.5. Hand Parameters

Set the type of the hand (single solenoid/double solenoid, etc.) and work holding/non-holding when HOPEN* (open hand) and HCLOSE* (close hand) are executed.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Hand].
After you change the parameter value, you can rewrite the parameters concerning the hand in the robot controller by clicking the [Write] button.

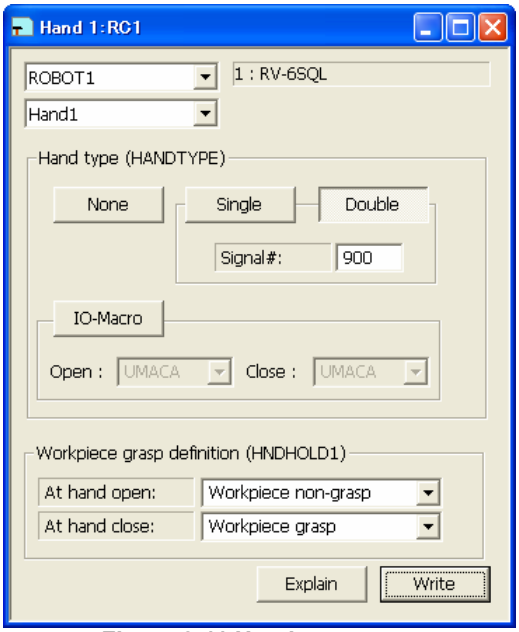


Figure 8-11 Hand parameter

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.6. Weight and Size Parameters

You can set the conditions for the hand mounted on the robot and the conditions for the work the robot grasps. Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Weight and Size].
After you change the parameter value, you can rewrite the weight and size parameters in the robot controller by clicking the [Write] button.

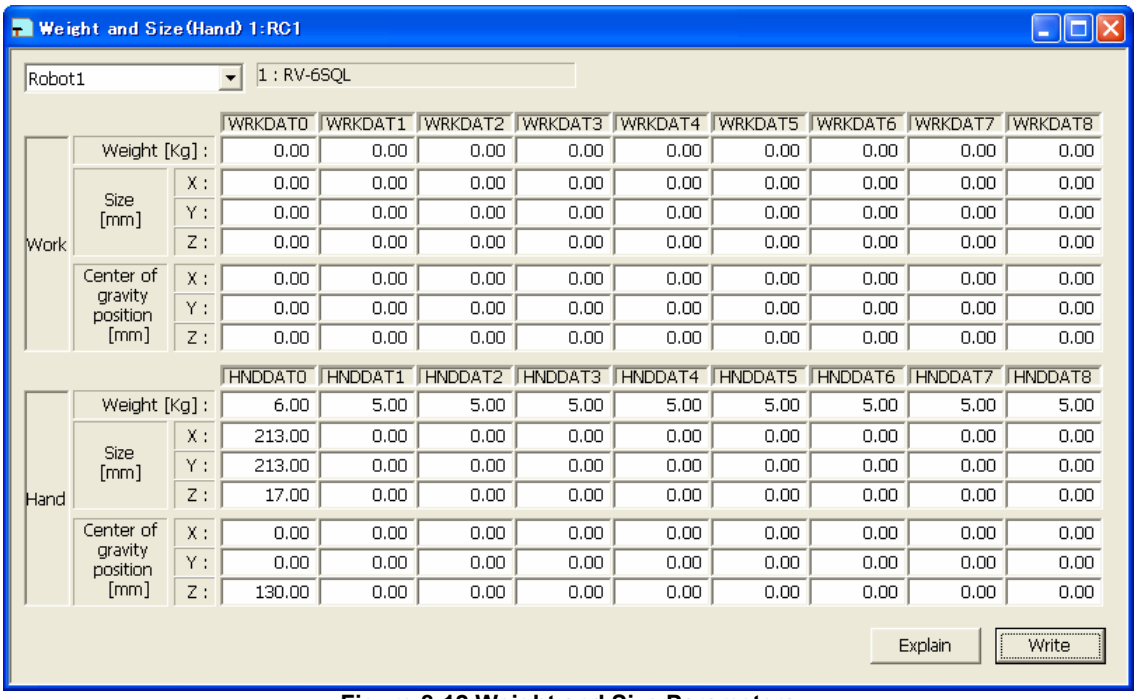


Figure 8-12 Weight and Size Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.7. Tool Parameters

Set the standard tool coordinates and standard base coordinates. Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Tool].
After you change the parameter value, you can rewrite the tool parameters in the robot controller by clicking the [Write] button.

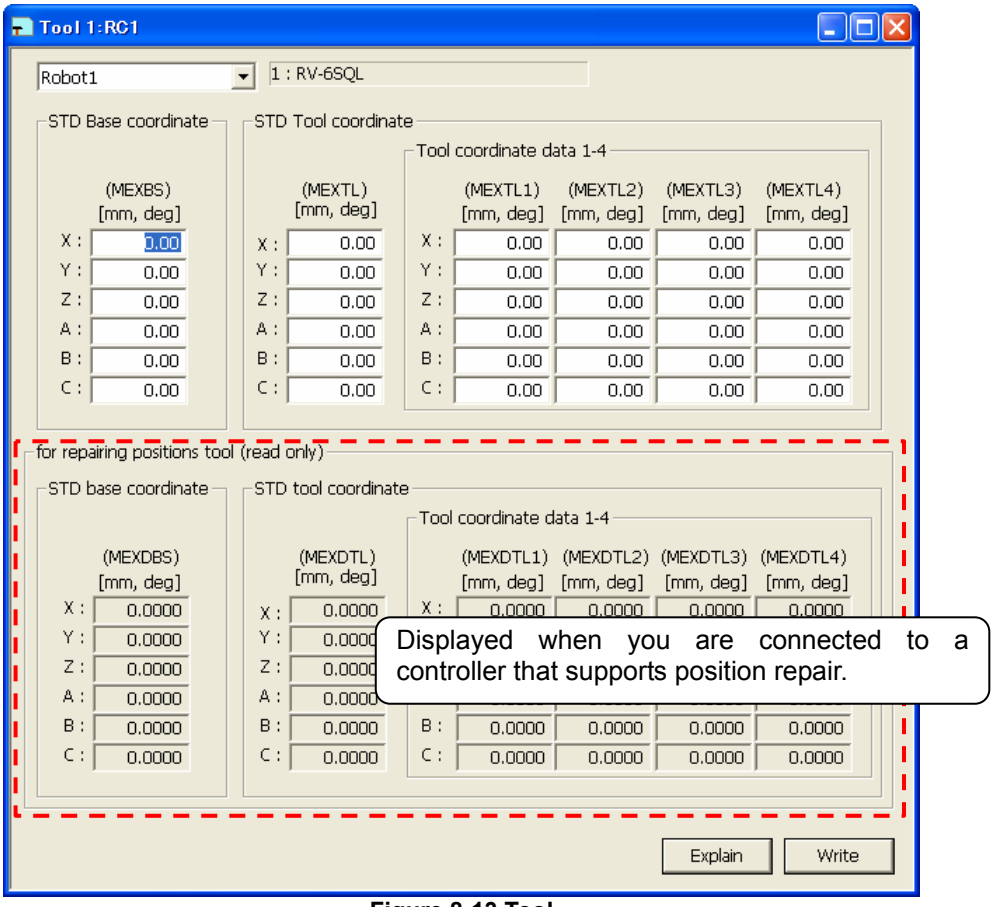


Figure 8-13 Tool

8.8. Slot Tables

Slot tables set the operating conditions of each task slot during multi-task operation.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Slot Table].
After you change the parameter value, you can rewrite the operating range parameter in the robot controller by clicking the [Write] button.

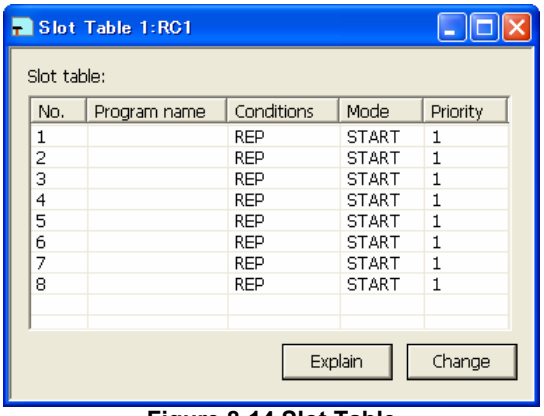


Figure 8-14 Slot Table

Select the task slot number you are changing and click the [Change] button.

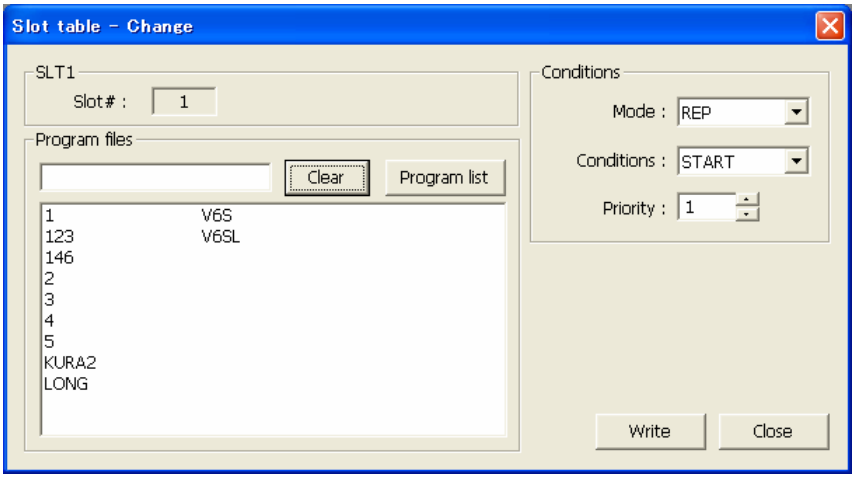


Figure 8-15 Slot Table - Change

When the modification window appears, set the program name, operating conditions, startup conditions and task priority, and then click [Write].

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.9. Output Signal Reset Pattern Parameters

These parameters set the operation when resetting the general-purpose output signals such as the CLR instruction and dedicated input (OUTRESET).
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Output Signal Reset Pattern].

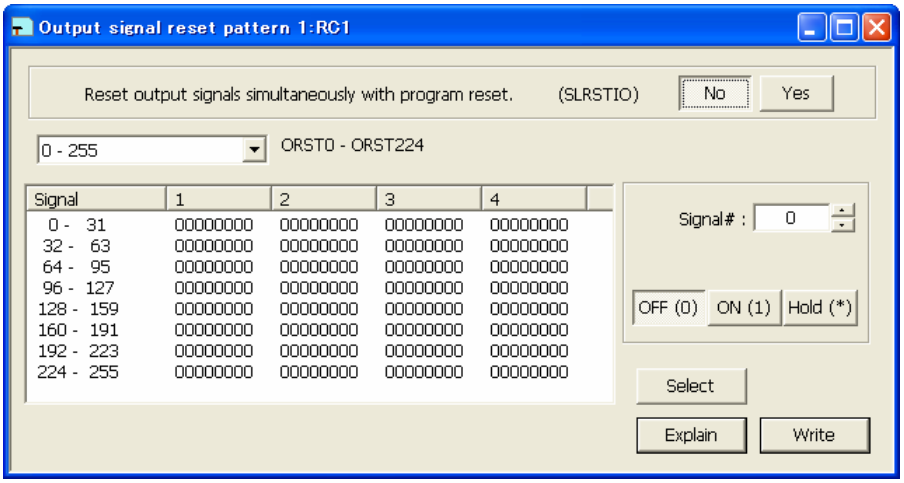


Figure 8-16 OUTPUT Signals reset pattern

Set a signal number, and then select one of [OFF]/[ON]/[Hold]. The value of the signal having the specified number displayed in the list changes. After you changed the parameter value, you can rewrite the output signal reset parameter in the robot controller by clicking the [Write] button.

Also, selecting a signal group (for example, "32-0") and then clicking the [Select] button changes 32 signals at once.

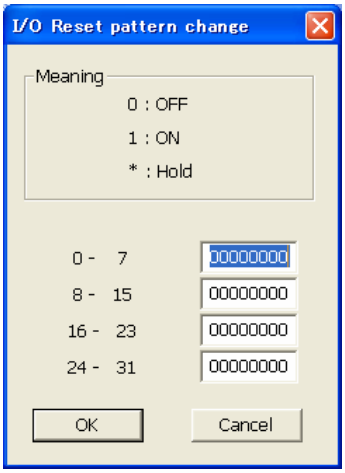


Figure 8-17 I/O Reset Pattern Change

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10. Assigning Dedicated Input/Output Signals

Assign signal numbers to functions in order to perform the remote operations to execute and stop robot programs, and display/operate the execution progress information and servo power supply status, etc.
Set parameters while connected to the robot controller.

8.10.1. General 1 Parameters

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [General 1].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

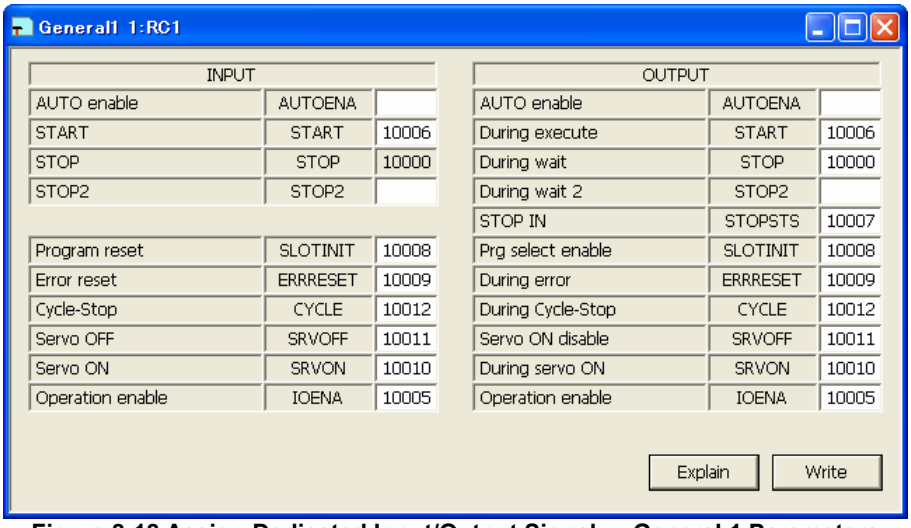


Figure 8-18 Assign Dedicated Input/Output Signal General 1 Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10.2. General 2 parameters

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [General 2].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

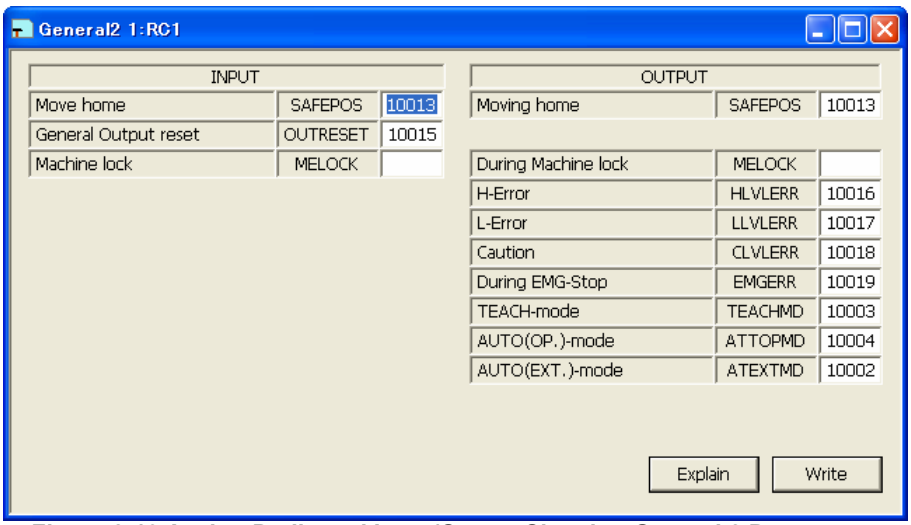


Figure 8-19 Assign Dedicated Input/Output Signal General 2 Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10.3. Data parameters

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [Data].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

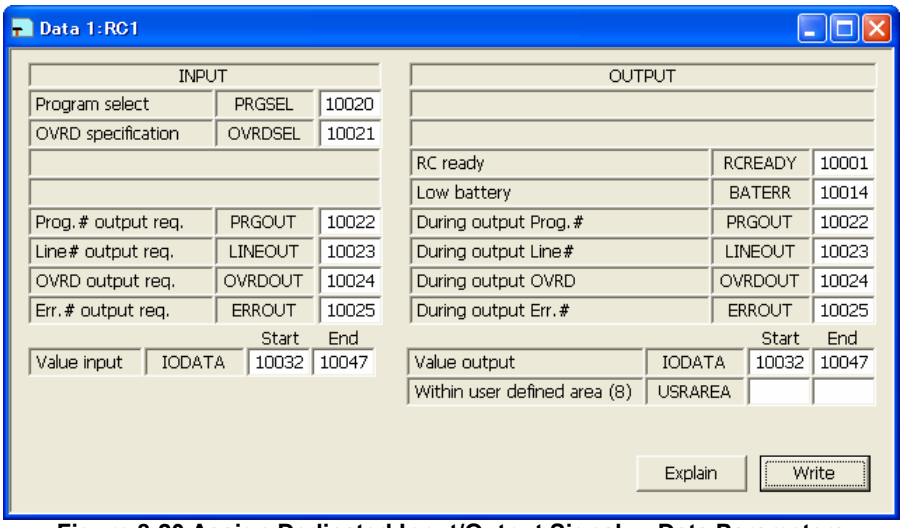


Figure 8-20 Assign Dedicated Input/Output Signal Data Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10.4. Jog parameters

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [Jog].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

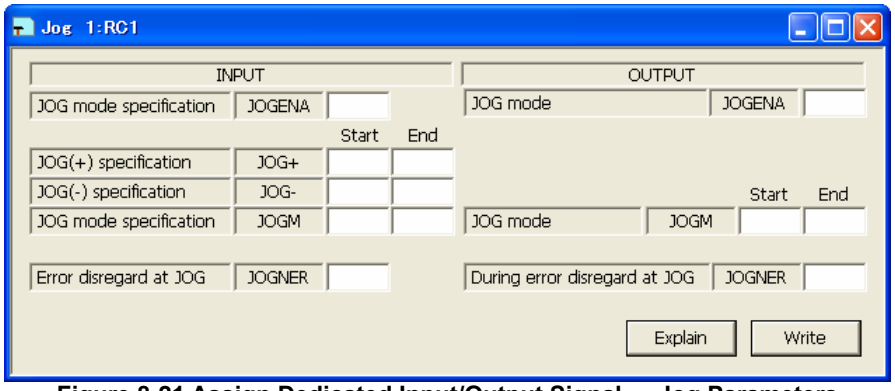


Figure 8-21 Assign Dedicated Input/Output Signal Jog Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

JOGNER(JOG command INPUT signal, During JOG OUTPUT signal) can be used with Version J2 or later of the CRn-500 series controller, or CRn-700 series controller

8.10.5. Hand parameters

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [Hand].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

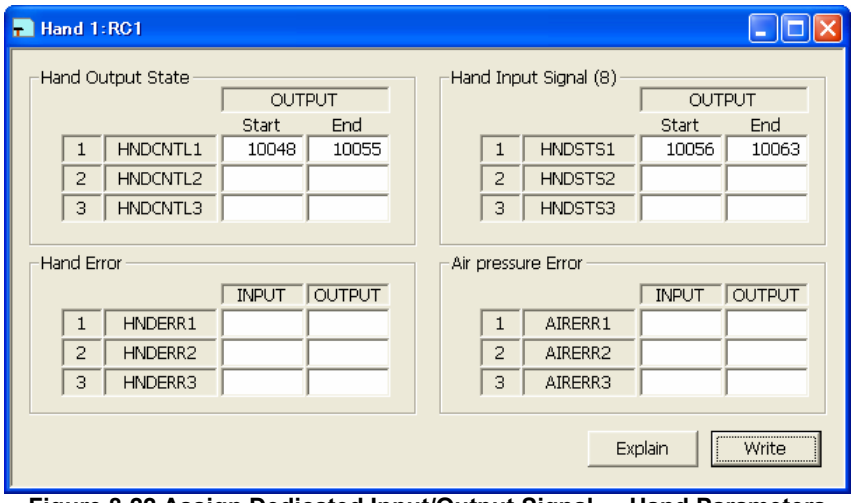


Figure 8-22 Assign Dedicated Input/Output Signal Hand Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10.6. Warm-up operation parameters

The warm-up operation parameters can be set when you are connected to a CRn-500 series robot controller ofJ8 edition or later or a CRn-700 series robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [Warm up].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

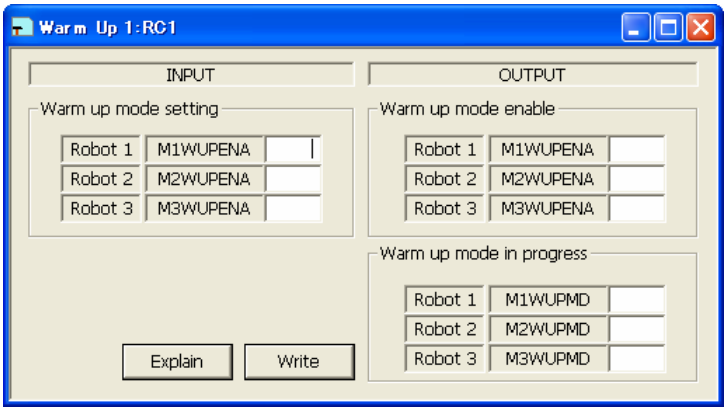


Figure 8-23 Assign Dedicated Input/Output Signal Warm-Up Operation Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10.7. Slot start (each slots) parameters

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [Start Slot].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

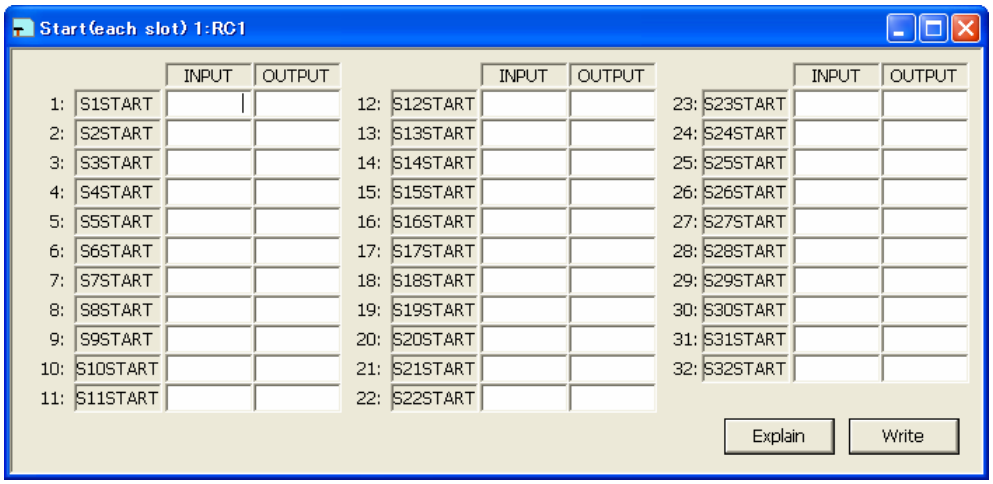


Figure 8-24 Assign Dedicated Input/Output Signal Slot Start Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10.8. Slot stop (each slots) parameters

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [Stop Slot].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

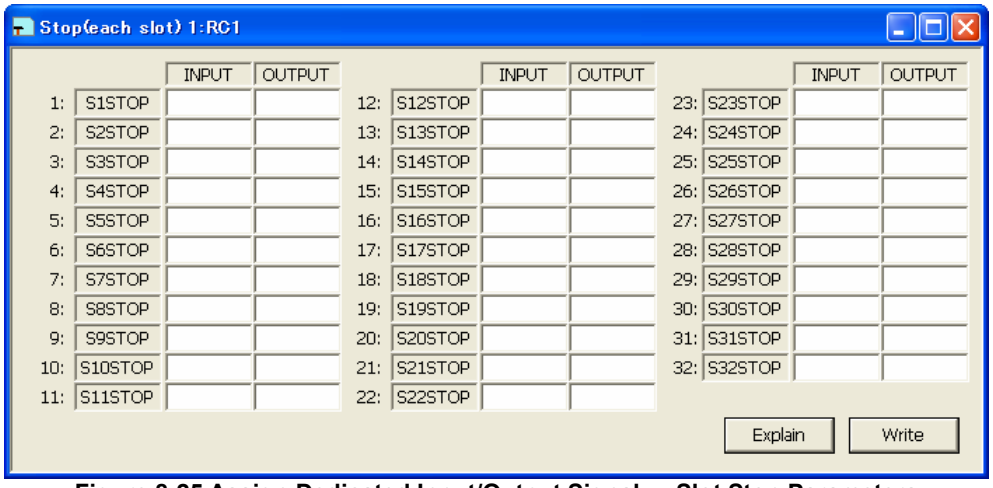


Figure 8-25 Assign Dedicated Input/Output Signal Slot Stop Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10.9. Servo On/Off (each robot) parameter

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [Servo On/Off].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

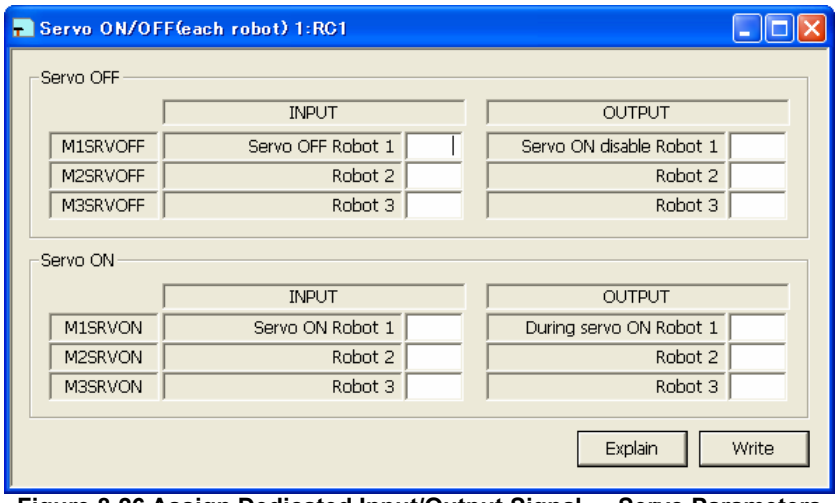


Figure 8-26 Assign Dedicated Input/Output Signal Servo Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.10.10. Machine lock (each robot) parameters

From the project tree, double click the target project [Online] -> [Parameter] -> [Assign dedicated input/output signal] -> [Machine lock].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

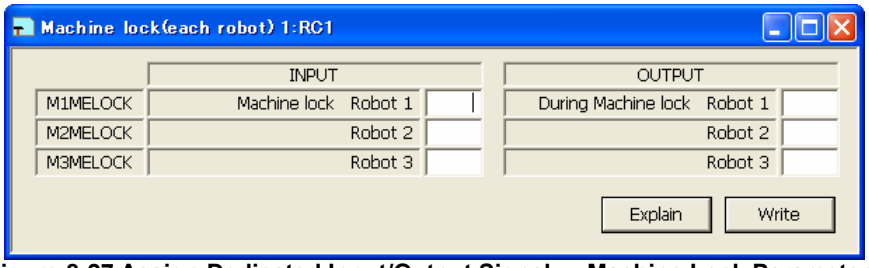


Figure 8-27 Assign Dedicated Input/Output Signal Machine Lock Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.11. RS-232 Setup Parameters

These parameters set up the communication environment of the RS-232 interface of the robot controller.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Communication].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

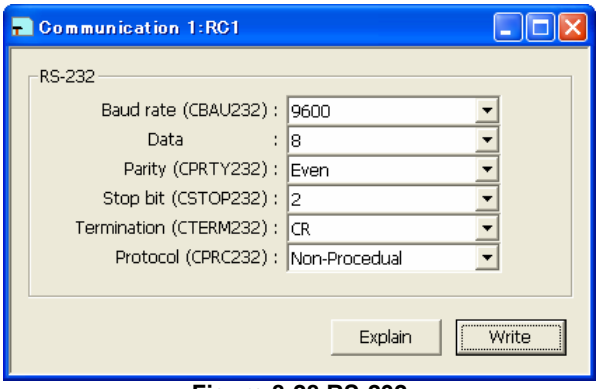


Figure 8-28 RS-232

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.12. Zone Parameters

You can specify the region (cuboid) defined with two points in the robot XYZ coordinates and set the behavior for when the robot enters this region. You can set up to 8 zones.

Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Zone].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

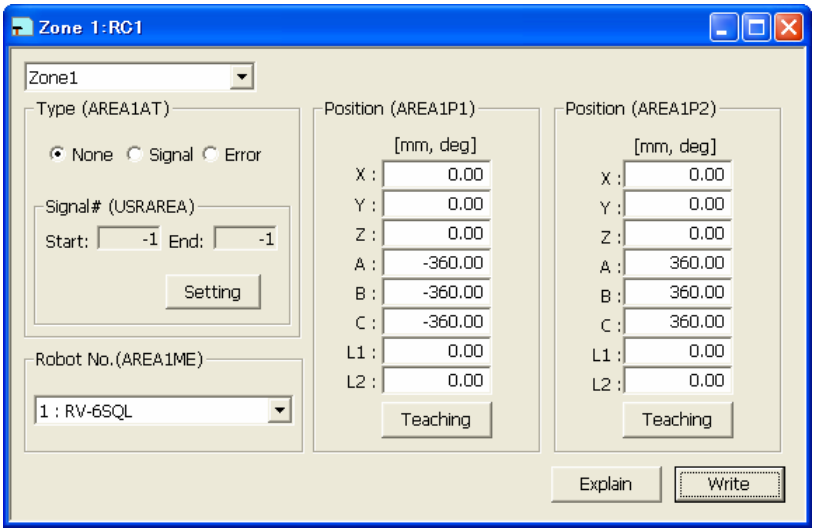


Figure 8-29 Zone Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.13. Free Plane Limit Parameters

You can set the overrun limit for using the robot on a free plane.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Free Plane Limit].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

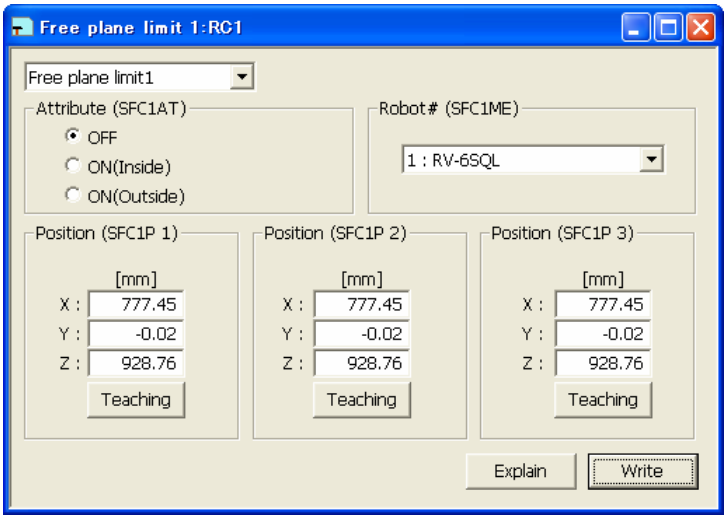


Figure 8-30 Free plane limit parameter

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.14. Escape Point Parameters

Set the position of the escape point.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Escape Point].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

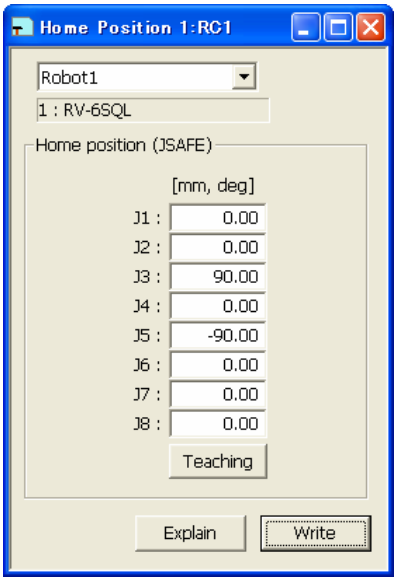


Figure 8-31 Home Position Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.15. Robot Program Language Parameters

You can set the robot program language used in a robot controller (MELFA-BASIC V/MELFA-BASIC IV/Movemaster commands). From the project tree, double click the target project [Online] -> [Parameter] -> [Robot Language].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

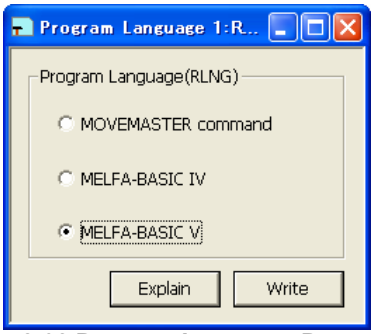


Figure 8-32 Program Language Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

For the usable robot program languages, see **"Table 6-1 Robot Program Languages for Each Controller"**.

Caution

About MELFA-BASIC V

MELFA-BASIC V. can only be used with CRn-700 series robot controllers.

About Movemaster commands

The Movemaster commands are restricted by the robot models that can be used. For details on whether you can use Movemaster commands with your robot, refer to its standard specifications.

8.16. Additional Axis Parameters

You can set information related to addition axes of robots.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Additional Axis].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

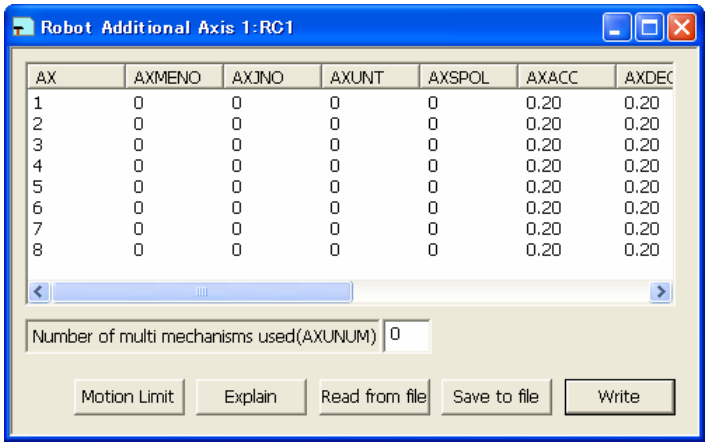


Figure 8-33 Additional Axis Parameter List

Motion Limit parameter screen starts with the [Motion Limit] button.
You can reference explanations of displayed parameters by pressing the [Explain] button.
You can read Additional Axis data saved in the file by pressing the [Read from file] button.
You can save Additional Axis parameter to the file by pressing the [Save to file] button.
But Motion Limit parameter is not included in the file saved by pressing the [Save to file] button.
([Moton Limit] button, [Read from file] button, and [Save to file] button correspond with RT ToolBox2 Ver.1.1 or later.)

Select the additional axis information to be edited and double click. The screen for setting information for the additional axis is displayed.

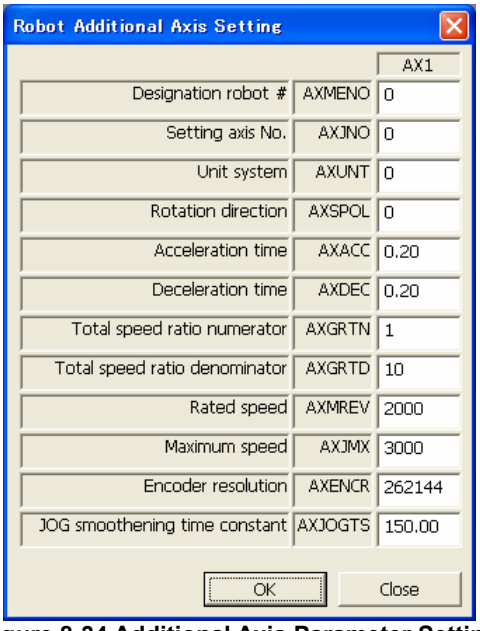


Figure 8-34 Additional Axis Parameter Setting

8.17. Collision Detection Parameters

You can set information related to the robot's collision detection functions.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Collision Detection].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

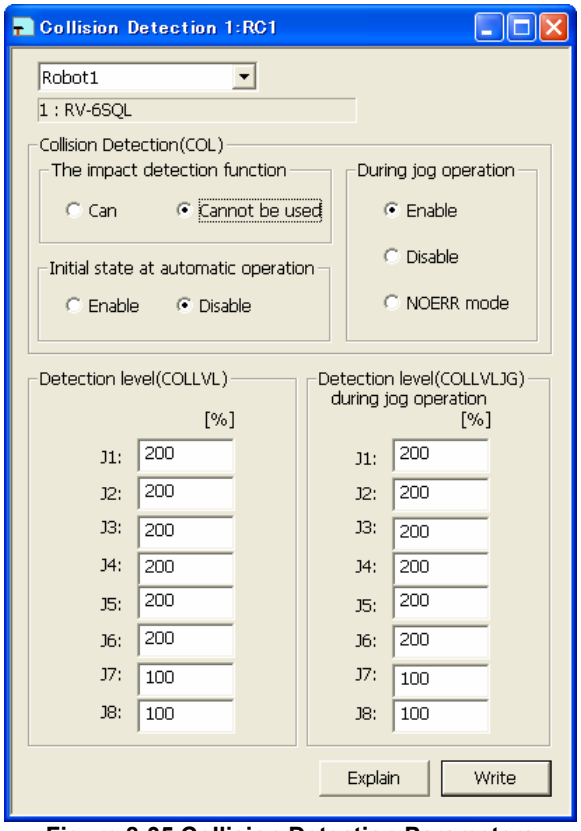


Figure 8-35 Collision Detection Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.18. Warm-Up Operation Parameters

You can set information related to the robot's warm-up function.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Warm-Up].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

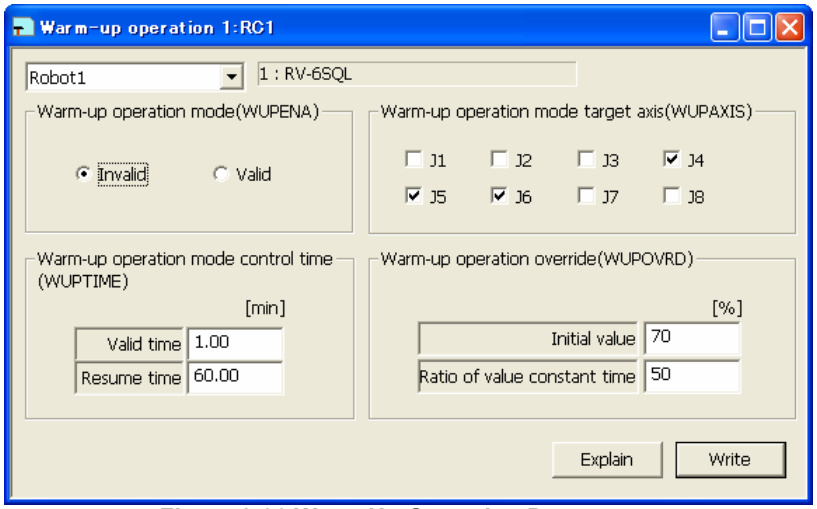


Figure 8-36 Warm-Up Operation Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.19. Movement Parameters

You can set information related to the optimum acceleration/deceleration for robot operation and set compliance errors.
The information that can be set concerning movement parameters depends on the robot controller connected.

	CRn-700 series	CRn-500 series
Maximum acceleration/deceleration correction ratio	Can be used with all versions	J2 edition or later
Compliance error function settings		H6 edition or later
Optimum acceleration/deceleration setting		G1 edition or later
Direction of gravity		H4 edition or later

Set parameters while connected to the robot controller.
From the project tree, double click [Online] -> [Parameter] -> [Movement Parameters].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

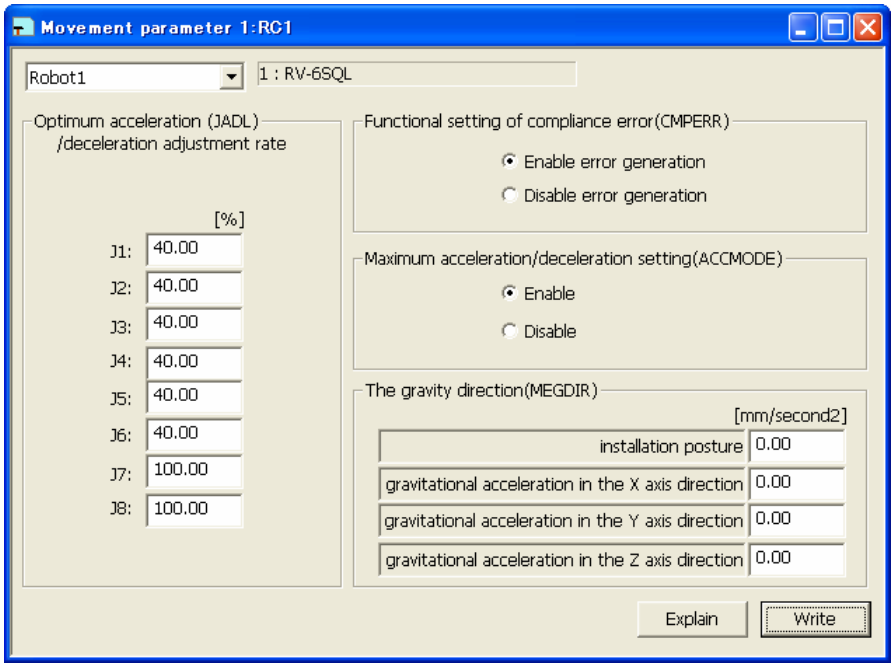


Figure 8-37 Movement Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.20. Program Parameters

You can set parameters related to robot programs.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Program Parameters].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

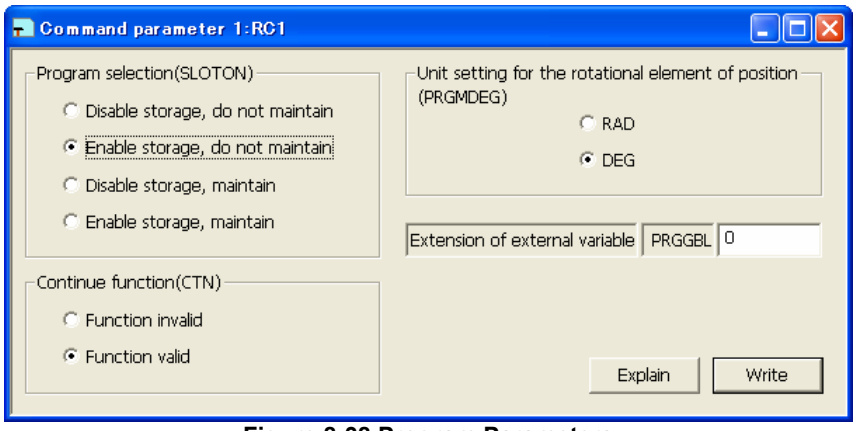


Figure 8-38 Program Parameters

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.21. User Error Parameters

You can set the message, cause, and recovery method for user errors set with a program.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [User Error].

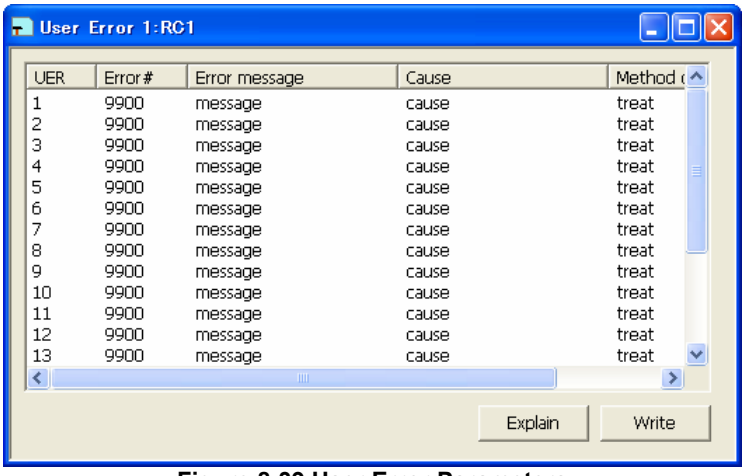


Figure 8-39 User Error Parameters

Double click the error number from the list. The "User Error" edit screen is displayed.

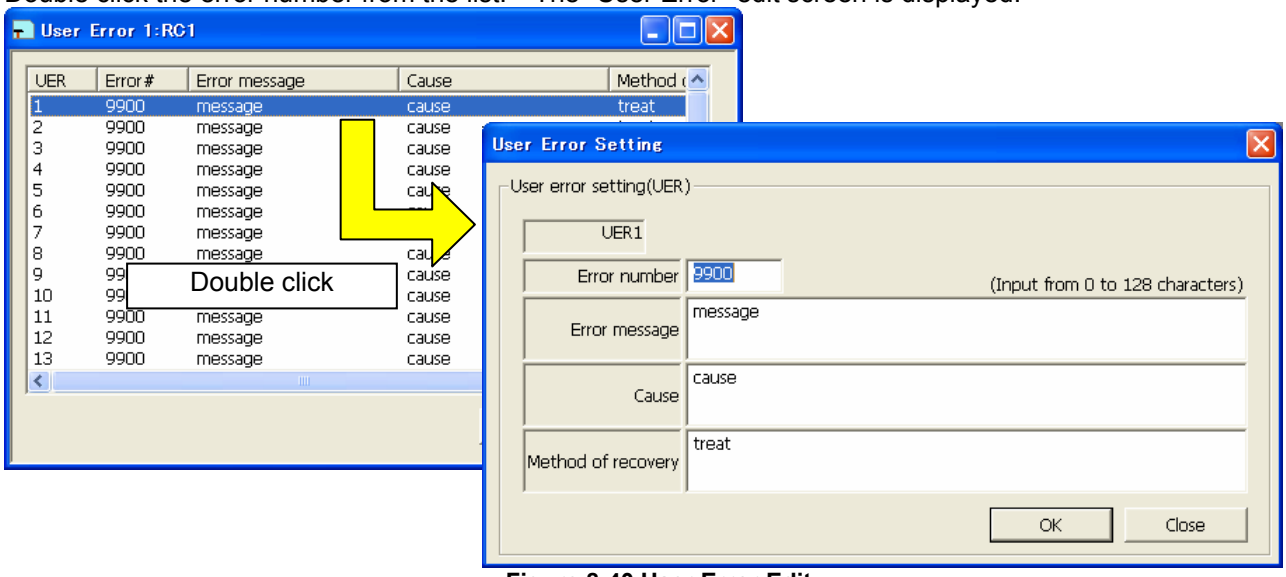


Figure 8-40 User Error Edit

Input the error number, error message, cause, and recovery method, then click the [OK] button. The user errors input to the list are displayed.
At this time, input an error number from "9000" – "9200".
After you confirm the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.
Please note that the list display is not sorted by error number.

You can reference explanations of displayed parameters by pressing the [Explain] button.

8.22. Ethernet Settings

You can set robot controller Ethernet information.
Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] -> [Parameter] -> [Ethernet Settings].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

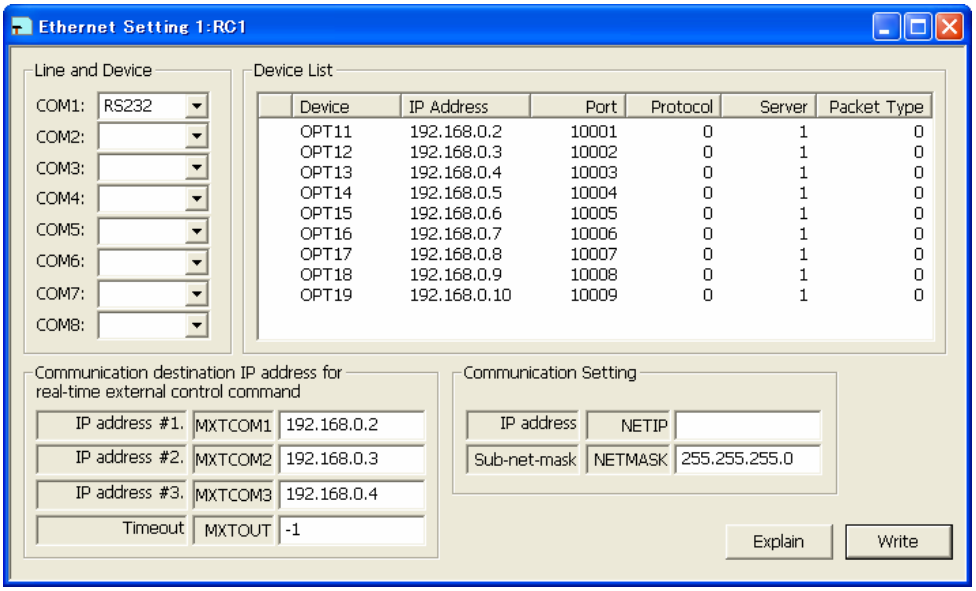


Figure 8-41 Ethernet Settings

You can reference explanations of displayed parameters by pressing the [Explain] button.

Select the device to edit and double click.　A screen is displayed for setting a variety of device information.

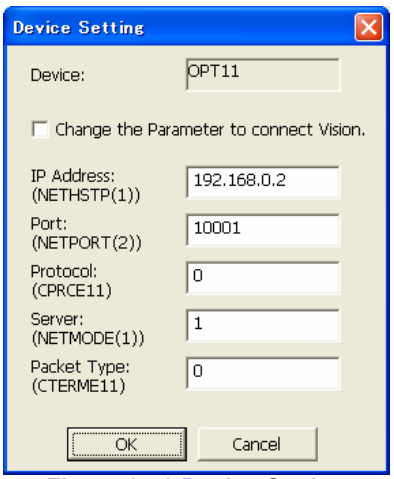


Figure 8-42 Device Setting

If you check the [Change the Parameter to connect Vision] checkbox, the items from "Port" downward take on the values for a network vision sensor.

Caution

Using a CRn-500 series robot controller

When using Ethernet with a CRn-500 series robot controller, the "Ethernet interface" option is required.

8.23. Multiple CPU Settings

You can set the parameters related to the Multiple CPU to use the CRnQ-700 series robot controller.
These parameters can be set when you are connected to CRnQ-700 series robot controller.
This function is available from RT ToolBox2 Ver.1.1 or later.

Set parameters while connected to the robot controller.
From the project tree, double click the target project [Online] → [Parameter] → [Multiple CPU setting].
After you change the parameter value, you can rewrite the parameters in the robot controller by clicking the [Write] button.

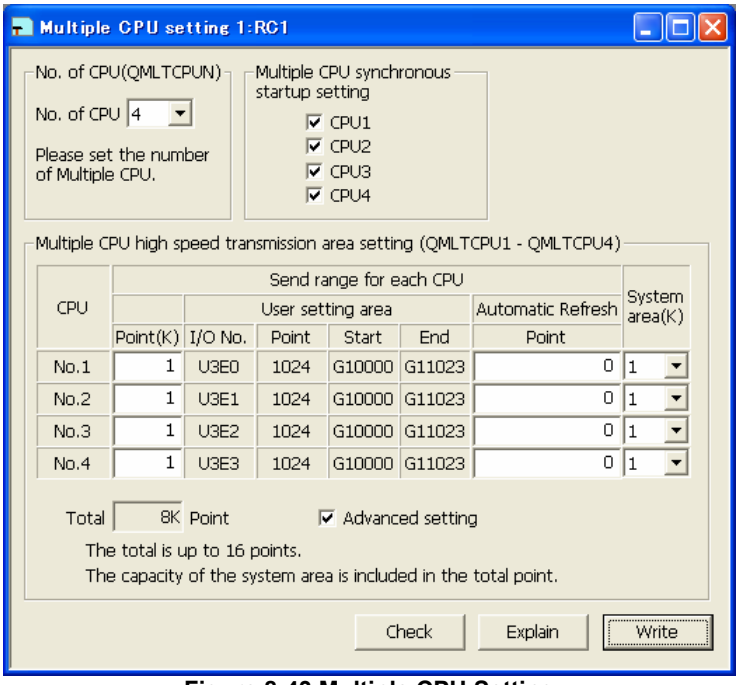


Figure 8-43 Multiple CPU Setting

You can reference explanations of displayed parameters by pressing the [Explain] button.
You can check the number of points input and range of total by pressing the [Check] button.

Please refer to the manual of Universal model QCPU (QCPU User's Manual (Multiple CPU System)) for details of Multiple CPU setting.

8.24. Parameter printing

You can print the parameter values held in a robot controller.

Display the parameters to print, then click on the menu bar [Workspace] -> [Print]. The print screen is displayed, so check the printer, then click the [OK] button. Printing starts.
Also, you can click on the menu bar [Workspace] -> [Print Preview] to look at the print image.

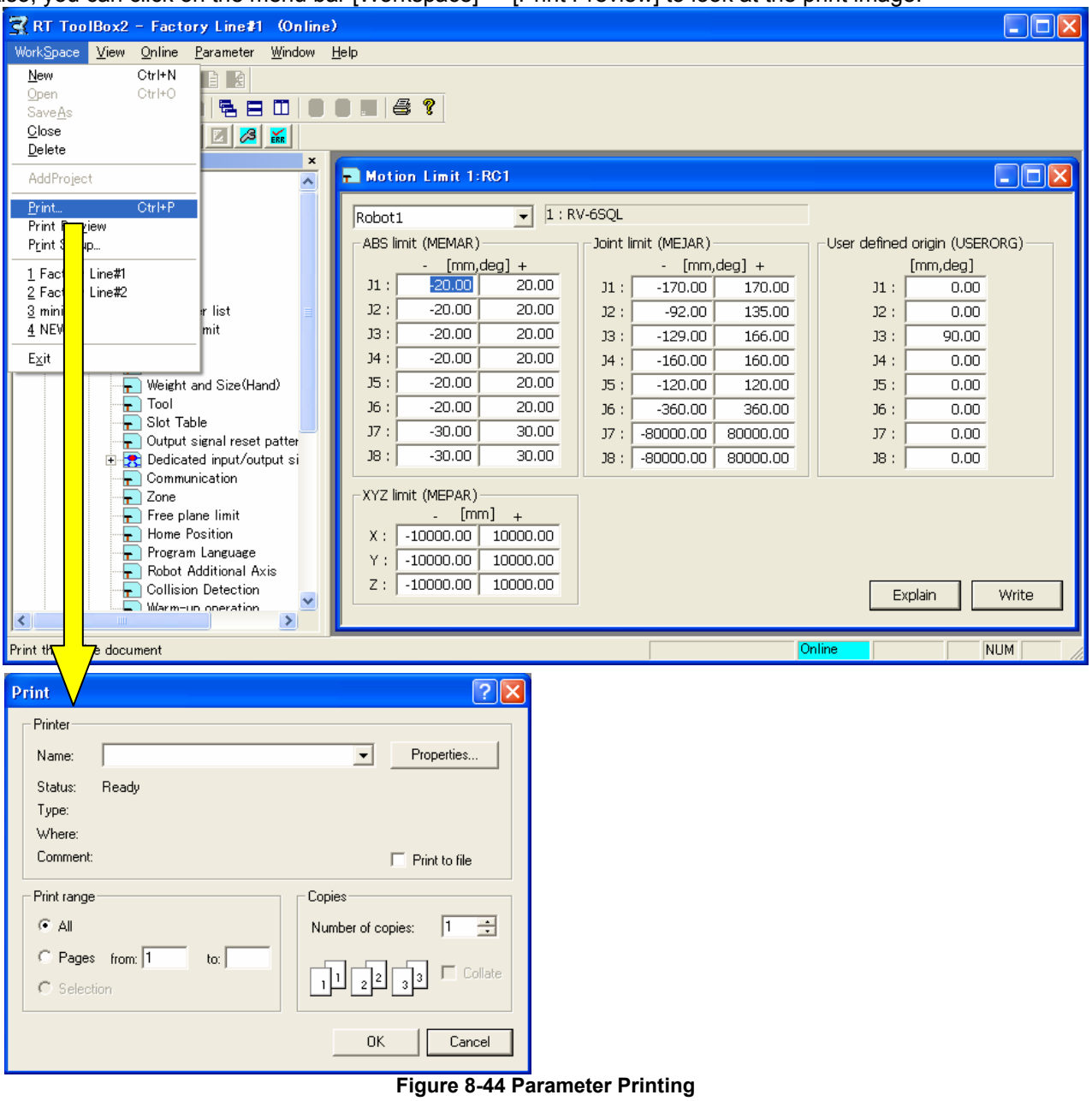


Figure 8-44 Parameter Printing

⚠ Caution

Install the printer beforehand.

Install the printer beforehand. For details on the installation method for the printer, refer to the operations manuals for your printer and computer.

9. Status Monitoring

You can set various information in the currently connected robot controllers to be constantly displayed.
The monitor functions are roughly divided into the following three.

1. Robot movement monitor ..... Items related to robot movement are monitored.
2. Operation monitor ..... Items related to the robot's operation are monitored.
3. Servo monitor ..... The robot's servo system information is monitored.

Table 9-1 Summary of Each Monitor

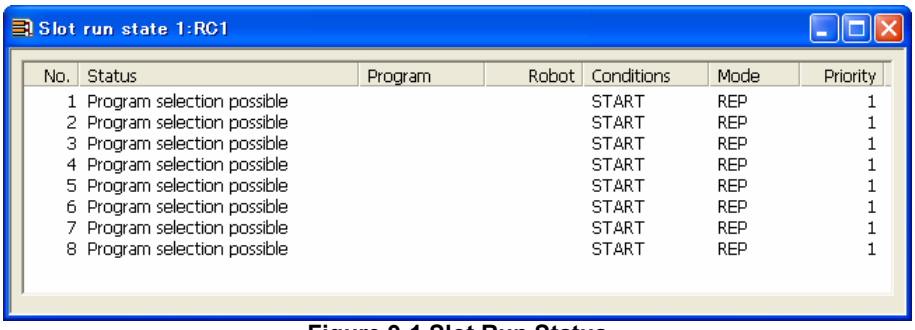
	Monitor name	Explanation
Robot movement monitor	Slot run state	The operation state of each slot can be confirmed.
	Program monitor	The program execution line set for each slot, the contents of the variable used in the program, and the robot current position, etc., can be confirmed.
	Movement State	The current position information and hand open/close state of each connected mechanism can be confirmed.
	Error	The currently occurring error can be confirmed. The history of the errors that have occurred can be confirmed.
	Robot status	You can display the robot and its movements in 3D to check them.
Signal monitor	General signals	You can check the statuses of signals input to the robot controller from outside equipment and signals output from the robot controller to outside equipment. Pseudo-input and forced output of signals are also possible.
	Named signal	The status can be checked by naming the status of the dedicated I/O signal that has been set in the robot controller, as well as each bit or within the range of 32 bits of the general-purpose signal. The signals are set via parameter setting (maintenance tool).
	Stop signal	The stop signal input into the robot controller can be confirmed.
	Registers (CC-Link)	You can monitor the input registers and output registers for the CC-Link functions. Pseudo-input and forced output of registers are also possible.
Operating monitor	Operating time	The robot operation time (power ON, etc.) can be confirmed.
	Production information	The operating time of the program in the robot controller and the No. of program cycles can be confirmed.

9.1. Robot Operation Monitoring

9.1.1. Slot operation statusmonitaring

The state of the slots in the robot controller can be monitored.

From the project tree, double click the target project [Online] -> [Monitor] -> [Movement Monitor] -> [Slot Status].



No.	Status	Program	Robot	Conditions	Mode	Priority
1	Program selection possible			START	REP	1
2	Program selection possible			START	REP	1
3	Program selection possible			START	REP	1
4	Program selection possible			START	REP	1
5	Program selection possible			START	REP	1
6	Program selection possible			START	REP	1
7	Program selection possible			START	REP	1
8	Program selection possible			START	REP	1

Figure 9-1 Slot Run Status

The No. of displayed slots is determined with the parameters.

9.1.2. Program monitoring

Information on the running program can be monitored.

From the project tree, click the target project [Online] -> [Monitor] -> [Movement Monitor] -> [Program Monitor], then double click the "Task slot" to monitor.

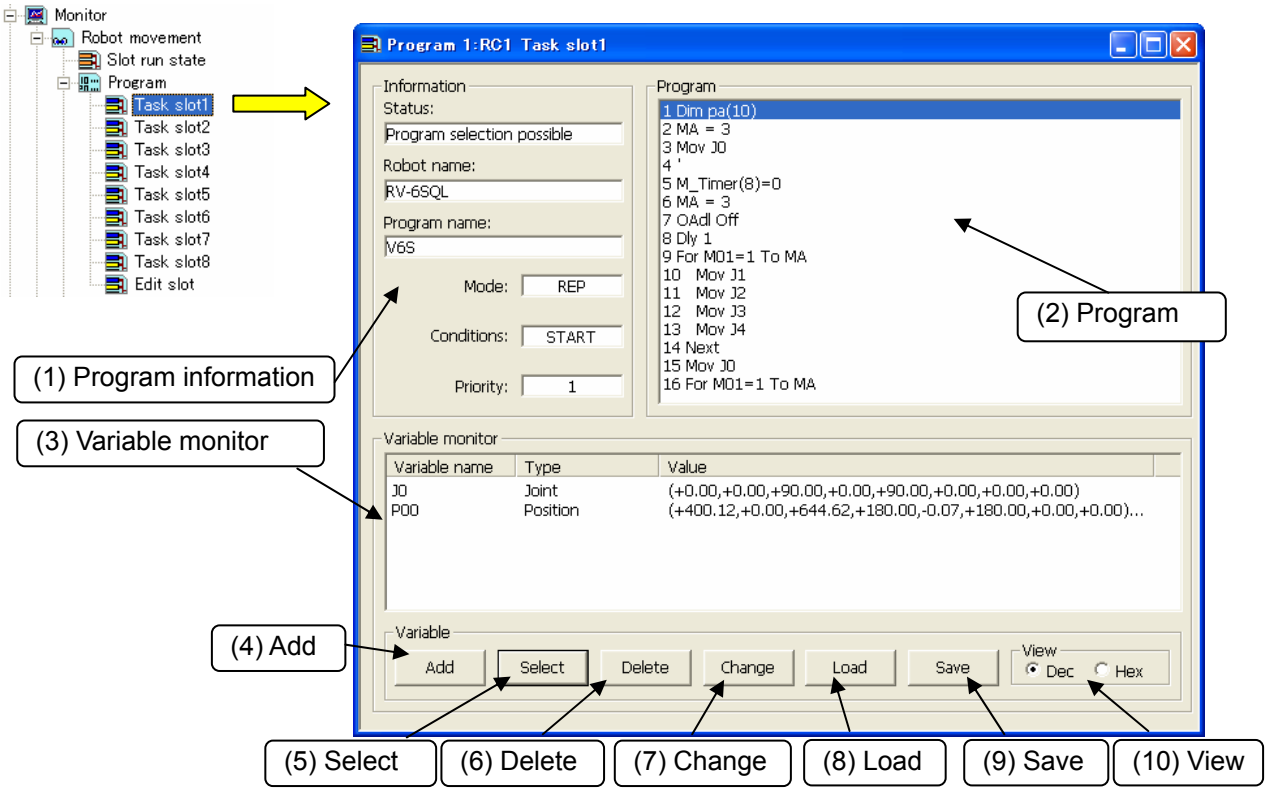


Figure 9-2 Program monitor

(1) Program information

You can check the currently selected program name and operation status and the name of the currently connected robot model.

(2) Program

The currently selected program is displayed. The currently executing line is displayed inverted.

(3) Variable monitor:

You can check the names of variables being used in the selected program. You can select the variables to monitor with the buttons displayed at the bottom of the screen.

(4) Adding variables

This adds more variables to monitor.

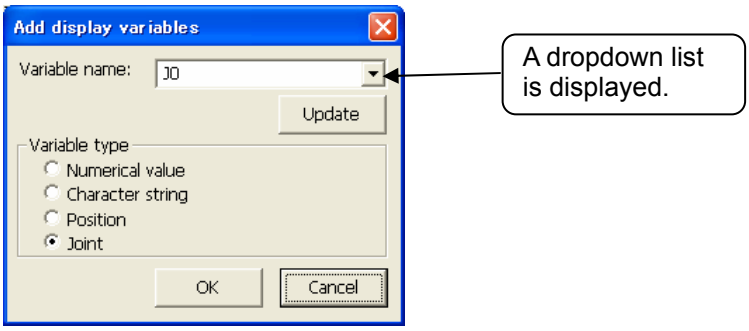


Figure 9-3 Add Variables

Input the variable name or select it from the dropdown menu, set the variable type, then click the [OK] button.

The variables being used in the program are displayed in a dropdown list. When you select variables from the dropdown list, the variable type is automatically selected.

(5) Selecting variables

You can batch select variables to monitor from the variable list used in the program.

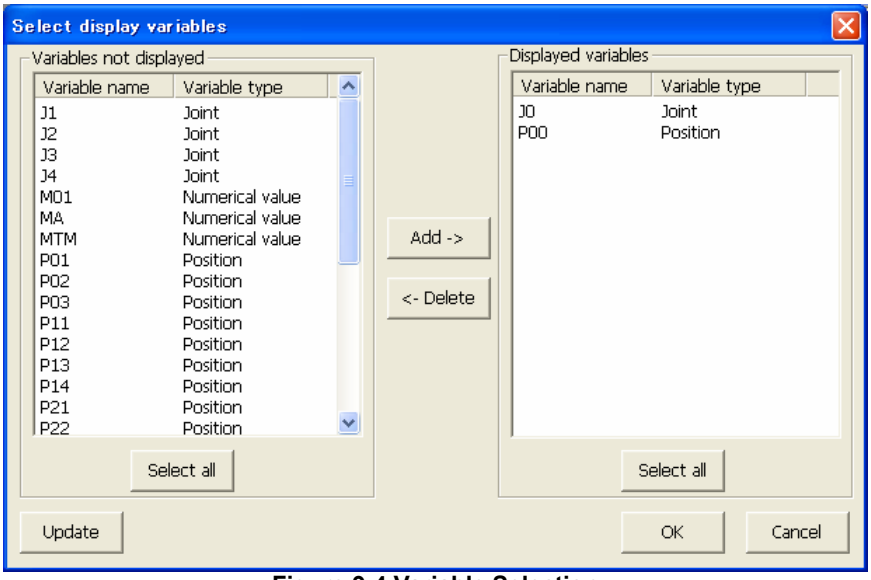


Figure 9-4 Variable Selection

The "variables not to display" are displayed in the list on the left side and the "variables to display" are displayed in the list on the right side. From the "variables not to display" list, select the variables to monitor, then click the [Add->] button. The selected variables are added to the "variables to display" list. If you select variables from the "variables to display" list, then click the [<-Delete] button, the selected variables are deleted from the "variables to display" list and added to the "variables not to display" list.

When you click the [OK] button, the variables registered in the "variables to display" list are displayed on the variables monitor and you can reference their values.

(6) Deleting variables

This deletes variables registered on the variables monitor from the monitor list. This operation does not delete the variables themselves from the program.

(7) Changing variables

You can change the values of variables registered on the variables monitor.

On the variables monitor, select the variables to change the values, then click the [Change] button.

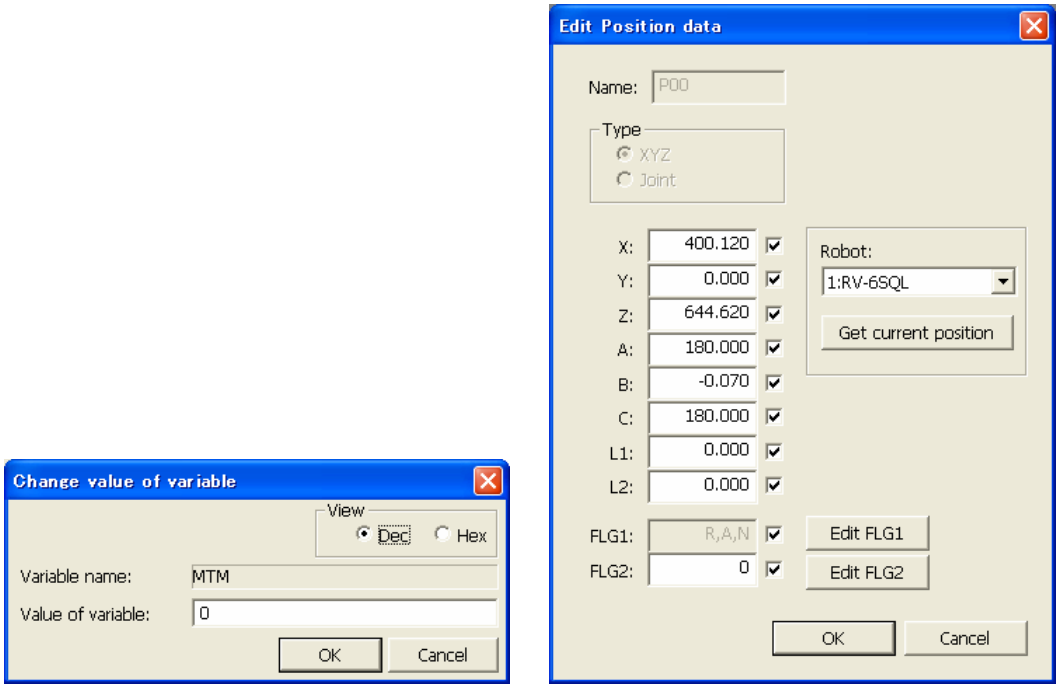


Figure 9-5 Changing Variable Values

After confirming the variable name, input the value of the variable, then click the [OK] button.

Caution

Be careful when changing the value of a variable.

When you change the value of a variable, the operation target position of the robot may change and result in a collision. This is particularly dangerous during robot operation, so check carefully before changing the value of a parameter.

(8) Load

You can load variables to be monitored on the variables monitor from a file.

When you click the [Load] button, you can load variable names and variable types and add them as variables to be monitored.

(9) Save

You can save as a file a list of the variables being monitored on the variables monitor.

When you click the [Save] button, you can save the names, types, and values of the variables currently being monitored into a file. This file is saved in text format.

(10) View

You can switch the values of the variables displayed on the variables monitor between hexadecimal display and decimal display.

The variables that can be displayed in hexadecimal, see below.

Table 9-2 16Variables that can be displayed in Hexadecimal

Integer	The displayed variable can be switched to the hexadecimal number / the decimal number.
Float	When it is 0 below the decimal point, it is possible to switch to the hexadecimal number / the decimal number. However, the value is the one within the range of -9999999 - 9999999.
String	The hexadecimal number is not displayed.
Location	The hexadecimal number is not displayed.

As for the value displayed by the hexadecimal number, “&H” is added to the head of the value.

9.1.3. Movement status

You can check the robot current position, destination position, hand open/close status, etc.

* Destination position corresponds with RT ToolBox2 Ver1.1 or later.

From the project tree, double click the target project [Online] -> [Monitor] -> [Movement Monitor] -> [Movement Status].

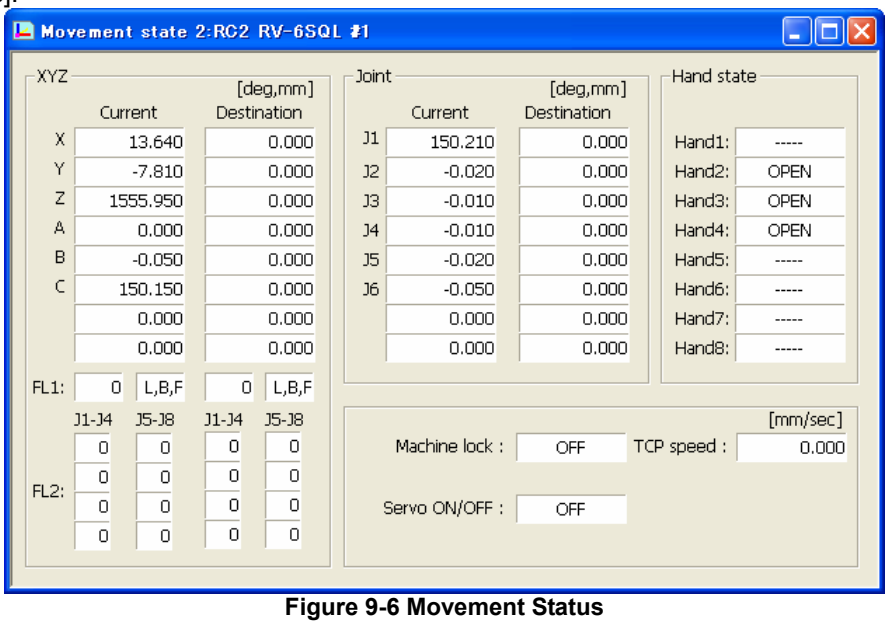


Figure 9-6 Movement Status

9.1.4. Errors

The errors currently occurring in the robot controller are displayed.

9.1.4.1. Referencing the current error

From the project tree, double click the target project [Online] -> [Monitor] -> [Movement Monitor] -> [Error].

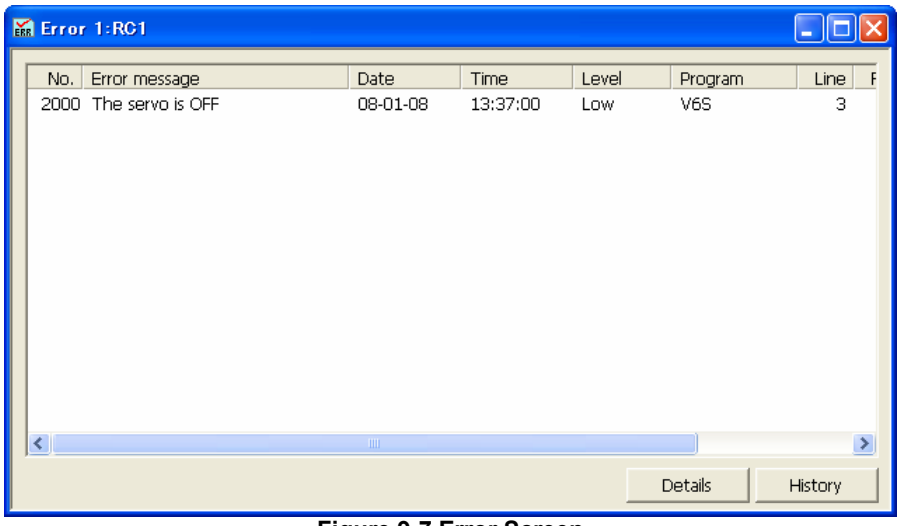


Figure 9-7 Error Screen

[Details] You can check details (cause and recovery method) on errors.
[History] You can reference the history of errors that have occurred.

9.1.4.2. Details

You can check details (cause and recovery method) on errors. On the error screen, select an error, then either click the "Details" button or double click the error.

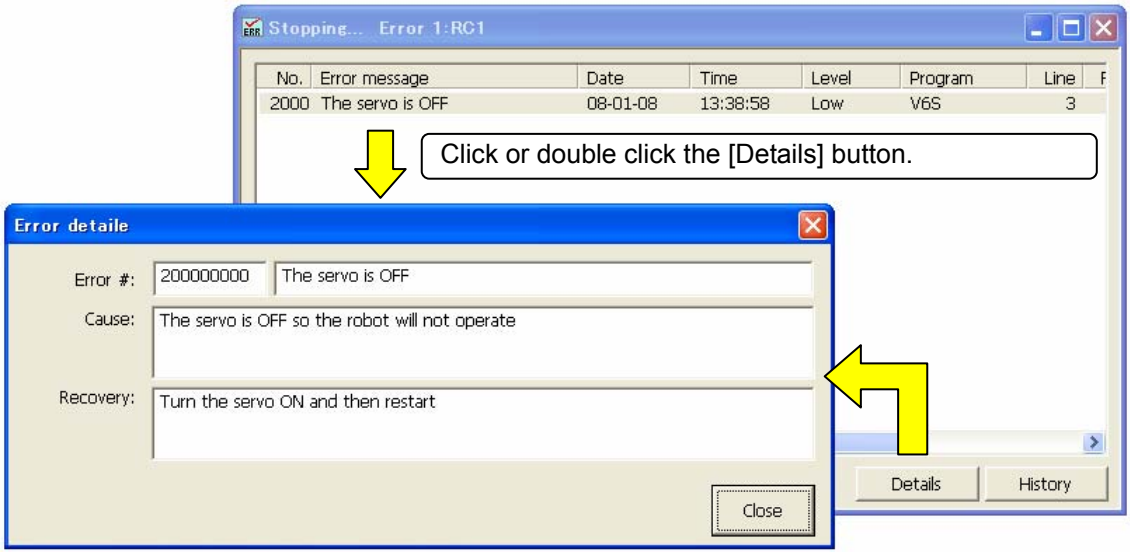


Figure 9-8 Error Details Screen

9.1.4.3. History information

The history of errors that have occurred in the past can be referred to.
You can check the error history in the robot controller for each error level (high level, low level, caution). You can also save the error history into a file.

Click the [History] button. After selecting the level to reference, click the [OK] button.

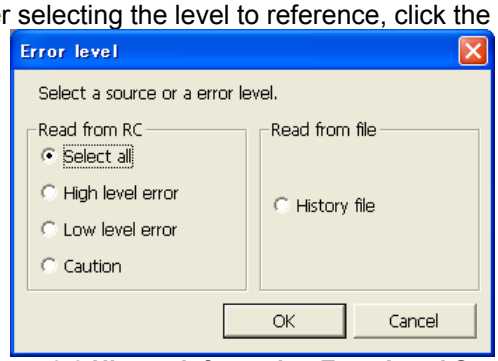


Figure 9-9 History Information Error Level Setting

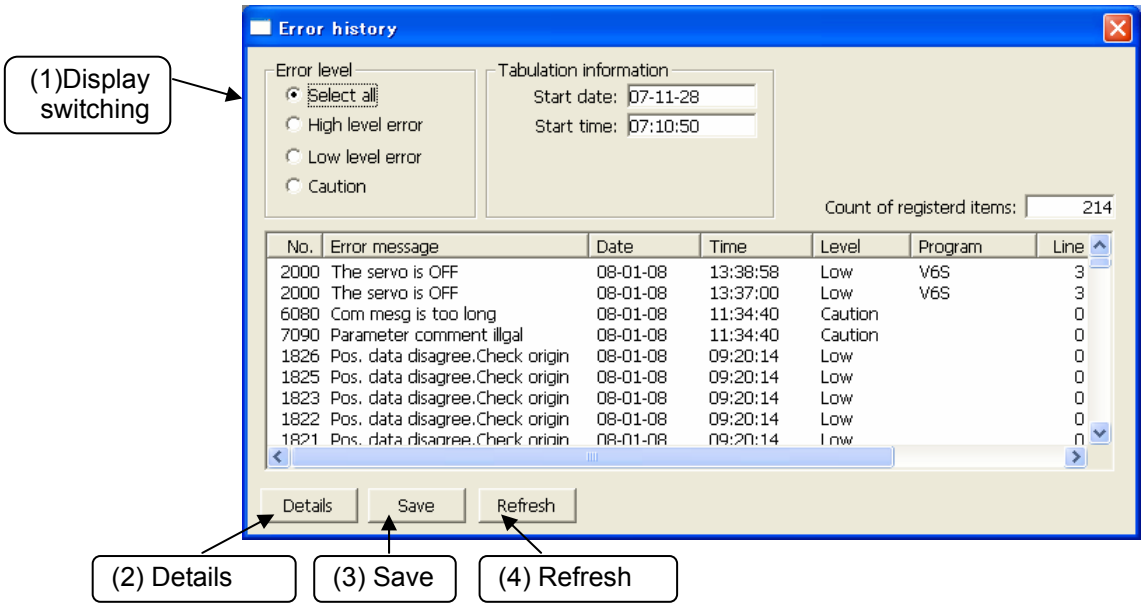


Figure 9-10 Error History

This display is not constantly displayed. To refresh information, click the [Refresh] button.

(1) Display switching
This redisplays the displayed error history for each error level.

(2) Details
After selecting a displayed error, you can check the cause and recovery method for the error by clicking this button.
You can also check details by double checking a displayed error.

(3) Save
This saves the displayed error history information into a file. For the saved error history information, select **"**From history file**"** with **"Figure 9-9 "**. Refer to **"Figure 9-11 "** on the screen, after you selected "From history file".
The error file is saved in text format. You can also print it with Notepad or other general text editors.

(4) Refresh
This refreshes the displayed information. The error history screen is not a self refreshing display monitor, so even if an error occurs after the error history screen is displayed, it is not automatically added to the display. Click the [Refresh] button.

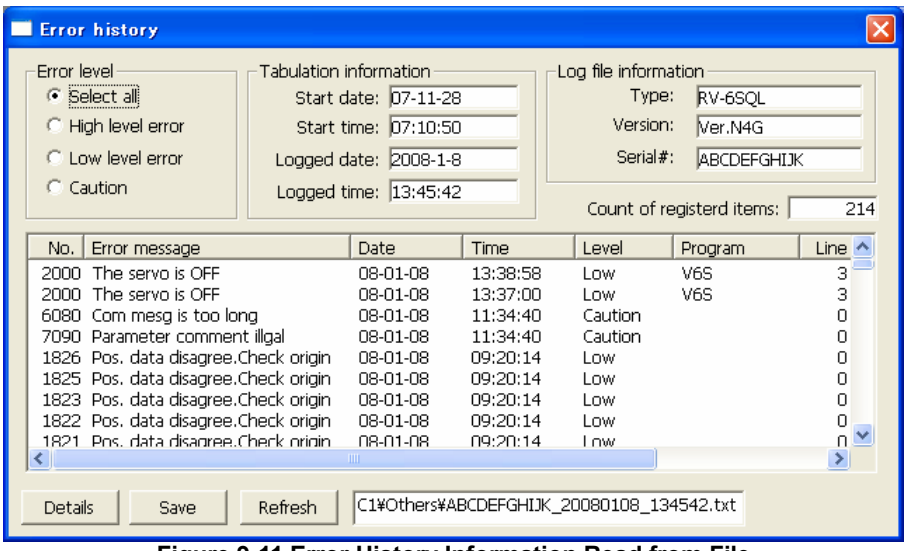


Figure 9-11 Error History Information Read from File

9.1.5. Robot status

You can display the robot and its movements in 3D to check them.

From the project tree, either double click the target project [Online] -> [<Robot model name>] or select [3D Monitor] with the mouse right button menu. The robot 3D display screen is displayed. If the "display travel table setting" is made for the project, the travel table is also displayed.

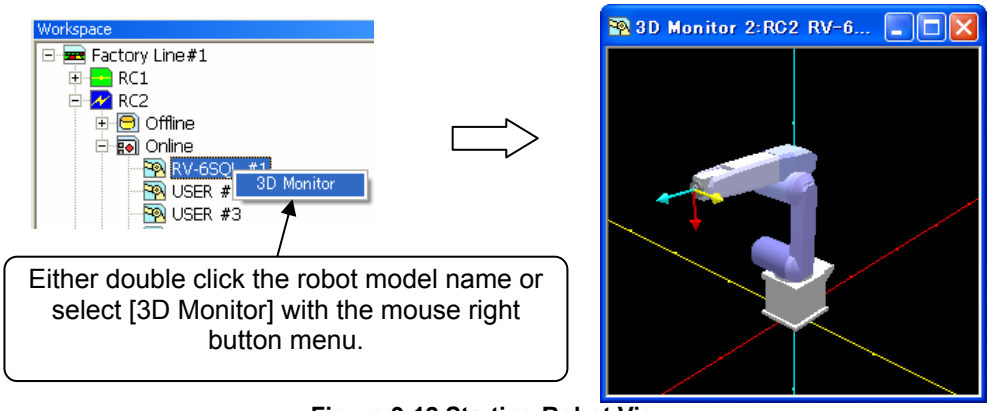


Figure 9-12 Starting Robot View

You can change the robot view perspective with mouse operation.

Table 9-3 Robot View Perspective Change Operation

Perspective to change	Graphic mouse operation
Perspective rotation	While clicking the left button, move left/right→ Rotation around Z axis Move up/down→Rotation around X axis Move left/right while clicking the left + right buttons→ Rotation around Y axis
Perspective movement	Move up/down/left/right while clicking the right button
Graphic enlargement/reduction	Move up/down/left/right while clicking pressing [Shift] key and clicking the left button

9.2. Signal Monitoring

9.2.1. General signal

You can check the statuses of signals input to the robot controller from outside equipment and signals output from the robot controller to outside equipment.

From the project tree, double click the target project [Online] -> [Monitor] -> [Signal Monitor] -> [General Signals].

The upper level displays the status of input signals and the lower level displays the status of output signals.

A continuous range of signals to display can be set freely with [Monitor Settings].

Pseudo-input and forced output of registers are also possible.

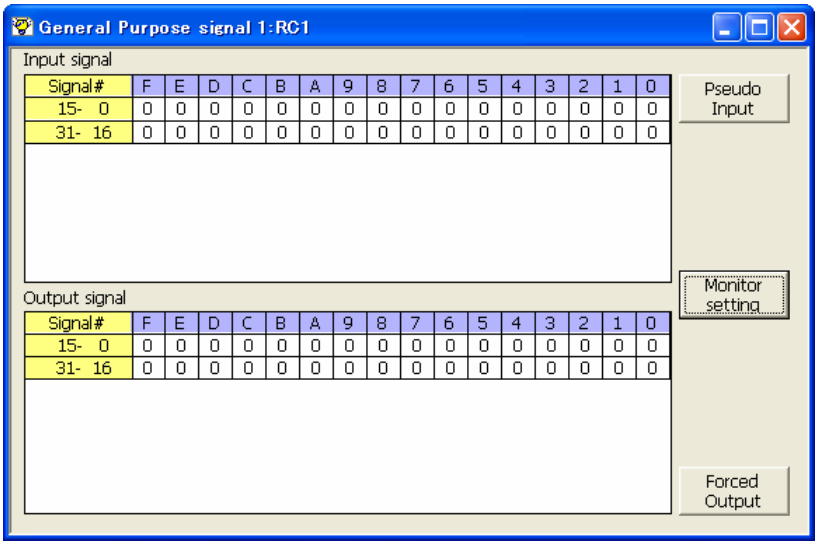


Figure 9-13 General Signals

9.2.1.1. Monitor settings

For the displayed signals, the continuous range can be set freely.

Set the lead numbers for the input signal number and output signal numbers to display, set their respective display ranges on the line, then click the [OK] button.

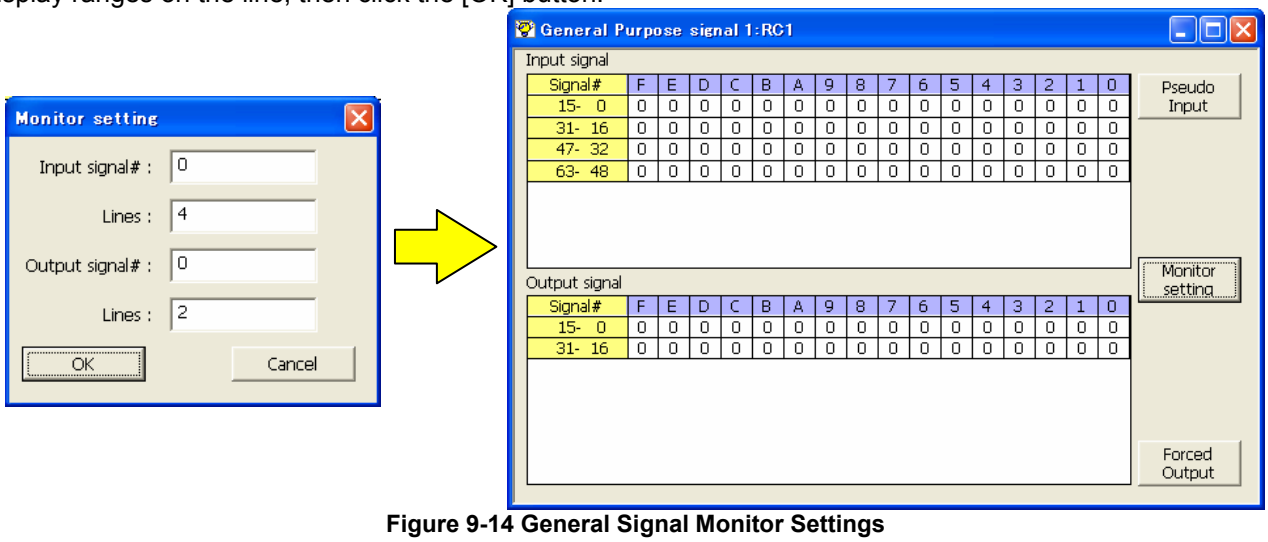


Figure 9-14 General Signal Monitor Settings

9.2.1.2. Pseudo-input

Pseudo-input means signals that are input to the robot controller from the computer, not from outside equipment.

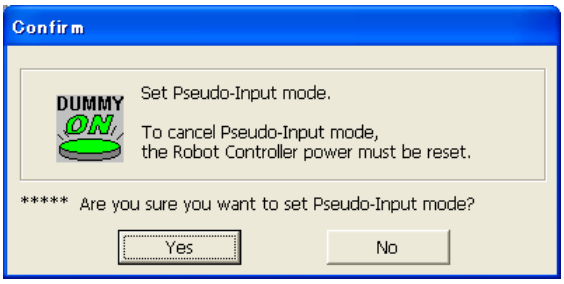
Click the [Pseudo-Input] button. A screen for inputting pseudo signals is displayed.

⚠ Caution

While the robot controller is in pseudo-input mode, signal input from outside devices is not accepted.

To use pseudo-input, put the robot controller into pseudo-input mode. While the robot controller is in pseudo-input mode, the robot controller does not accept signal input from outside devices.

Click the [Pseudo-Input] button. The confirmation message below is displayed before the robot controller goes into pseudo-input mode.



To release a robot controller from pseudo-input mode, switch the power for the robot controller Off, then On again.

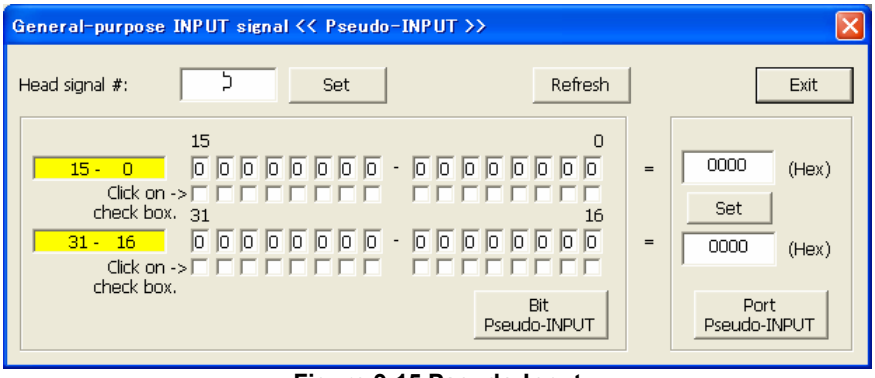


Figure 9-15 Pseudo-Input

① First, read the signals you want to pseudo input.
You can set 32 signals at the same time. Input the head number for the signals you want to read, then click the [Set] button.

② The input statuses of the 32 signals starting from the specified head signal number are displayed. Set the pseudo-input status, then click the [Bit Pseudo-INPUT] button.

③ You can specify the hexadecimal values and make pseudo input for the 32 signals starting from the head signal number.
Input the values in hexadecimal, then click the [Port Pseudo-INPUT] button.

9.2.1.3. Forced output

You can force signals to outside equipment from robot controllers.
Click the [Forced Output] button. A screen for forcibly outputting signals is displayed.

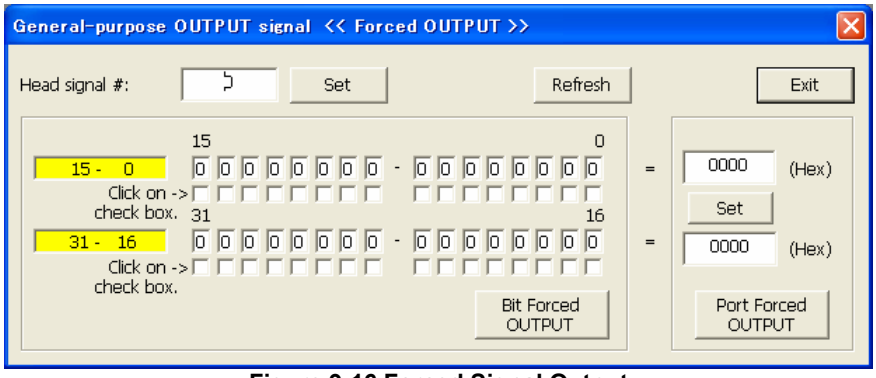


Figure 9-16 Forced Signal Output

① First, read the signals you want to forcibly output.
You can output 32 signals at the same time. Input the head number for the signals you want to read, then click the [Set] button.
② The output statuses of the 32 signals started from the specified head signal number are displayed. Set the output status, then click the [Bit Forced OUTPUT] button.
③ You can specify the hexadecimal value and force output for the 32 signals starting from the head signal number.
Input the values in hexadecimal, then click [Port Forced OUTPUT] button.

⚠ Caution

Forced signal output

- Signal numbers assigned (used) as dedicated output signals cannot be forcibly output.
- Forced output is possible if the robot controller mode is either [AUTOMATIC] or [MANUAL] (for a CRn-500 series robot controller, [TEACH], [AUTO (OP)], OR [AUTO (EXT.)]), but if even one program is running, forced output is not possible. (Except an ALWAYS program)

9.2.2. Named signals

You can give names to general input/output signals and check their statuses.

With "Named signals", you can check the status of dedicated input/output signals and named general input/output signals. When starting up, you can load a definition file for named signals in the robot controller.

From the project tree, double click the target project [Online] -> [Monitor] -> [Signal Monitor] -> [Named Signals].

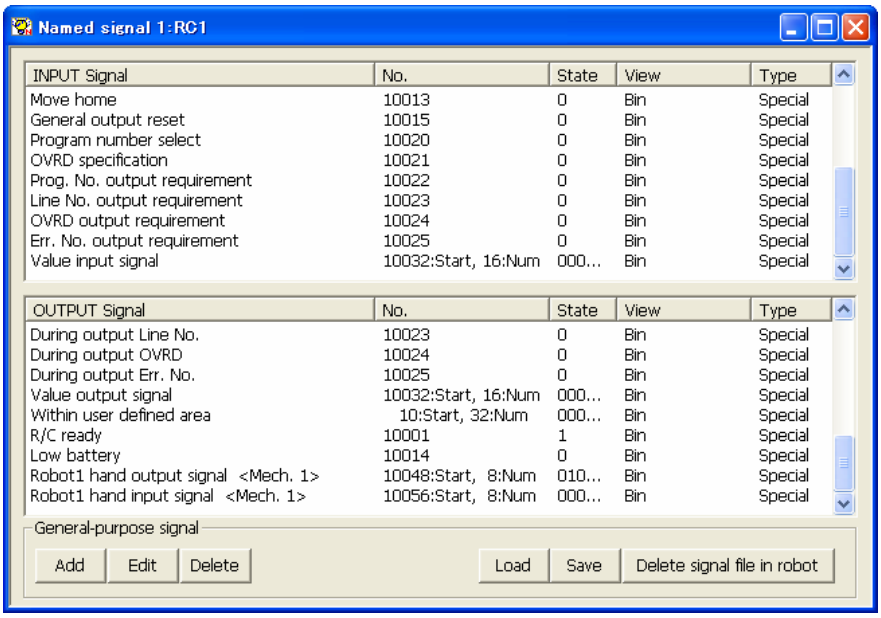


Figure 9-17 Named Signal

[Add] : This registers a new general input/output signal name.

[Edit] : This changes the setting for an existing general input/output signal selected in the list.

[Delete] : This deletes a signal selected in the list.

[Load] : This loads a file defining saved named signals in the robot controller and in the computer.

[Save] : This saves the information on the set named signals to the robot controller or computer.

[Delete signal file in robot] : This deletes the named signal information in the robot controller.

9.2.2.1. Adding new named signal or revising one

To add a new named signal, click the [Add] button. To revise one, select the signal to be revised from the list and click the [Edit] button.

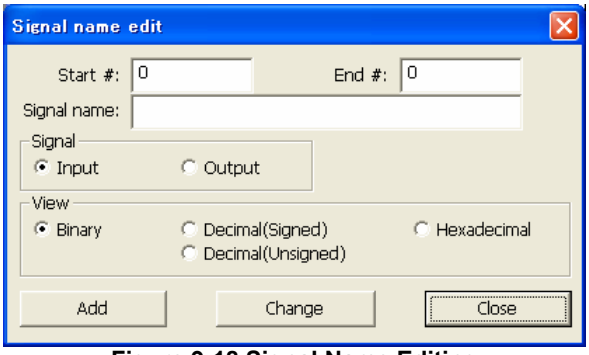


Figure 9-18 Signal Name Editing

Input the general signal numbers and names, select the signal type and display method, then when adding a new named signal, click the [Add] button. The set signal is added to the list.

The [Add] and [Change] buttons do not close the "Signal name edit" screen, so you can continue to add more signals.

9.2.2.2. Deleting a named signal

Select the signals to delete from the list. You can select multiple signals by clicking them while holding down the [Ctrl] key or the [Shift] key on the keyboard.
However, dedicated input/output signals can not be deleted.
Also, you can not delete input signals and output signals at the same time. Finally, delete the select signals in the list with a mouse click.

9.2.2.3. Named signal definition information reading

When you click the [Load] button, a message asking you to confirm that you want to delete a registered general input/output signal is displayed.
This message asks you to confirm that you want to clear the current general input/output signal display.

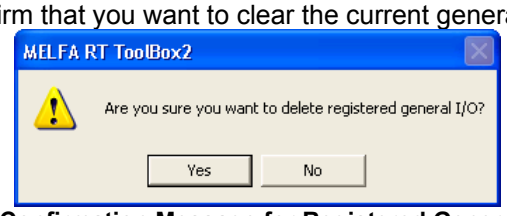


Figure 9-19 Deletion Confirmation Message for Registered General Input/Output Signal

If you select [Yes] on this confirmation screen, the "Select locate" screen asking you to select the load source is displayed.

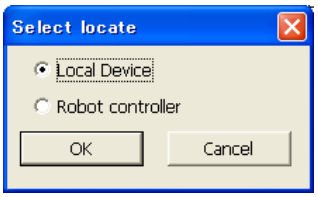


Figure 9-20 Device Selection

Select the device to load from, then click the [OK] button.
If you select the local device, the screen for selecting a file in the computer is displayed.
If you select the robot controller, the named signal definition file is loaded from the robot controller.

9.2.2.4. Saving definition information for named signals to a controller

When you click the [Save] button, the "Select locate" screen asking you to select the save destination is displayed.

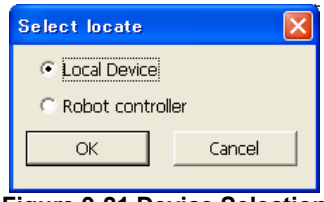


Figure 9-21 Device Selection

Select the device to save to, then click the [OK] button.
If you select the local device, the screen for selecting a file in the computer is displayed.
If you select the robot controller, the named signal definition file is saved to the robot controller.

9.2.3. Stop signal

You can reference the statuses of stop signals (stop/not stop) input to the robot controller.
From the project tree, double click the target project [Online] -> [Monitor] -> [Signal Monitor] -> [Stop Signals].

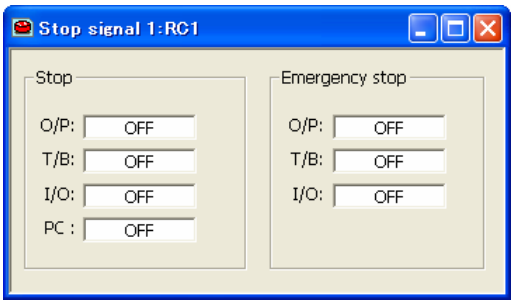


Figure 9-22 Stop Signal

9.2.4. Register (CC-Link) monitoring

⚠ Caution

This can only be used if the CC-Link option card is mounted in the robot controller.

This function can only be used with a CRn-500 series or CRnD-700 series robot controller with the CC-Link option card mounted. Also, this function can not be used with CRnQ-700 series robot controllers.

You can check the statuses of registers input to the robot controller from outside equipment and registers output from the robot controller to outside equipment.

From the project tree, double click the target project [Online] -> [Monitor] -> [Signal Monitor] -> [Register (CC-Link)].

The left side of the screen displays the status of input registers and the right side displays the status of output registers.

A continuous range of registers to display can be set freely with [Monitor Settings]. Pseudo-input and forced output of registers are also possible.

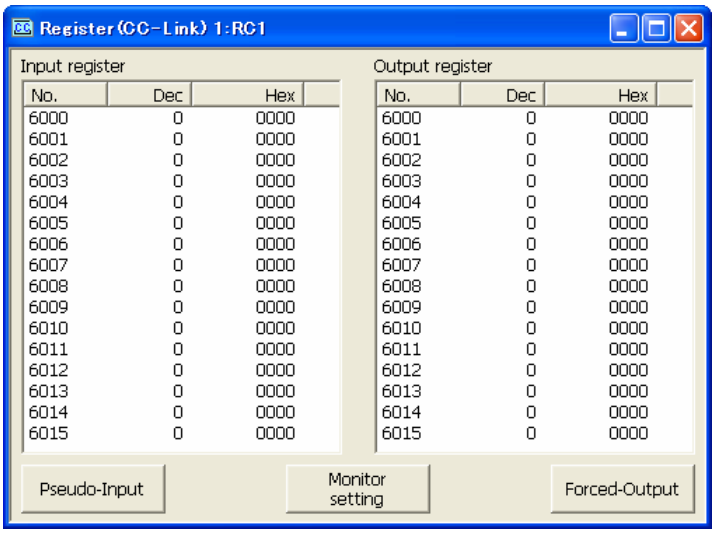


Figure 9-23 Registers (CC-Link)

9.2.4.1. Monitor settings

A continuous range of registers to display can be set freely.

Set the lead numbers for the input register number and output register numbers to display, set their respective display ranges on the line, then click the [OK] button.

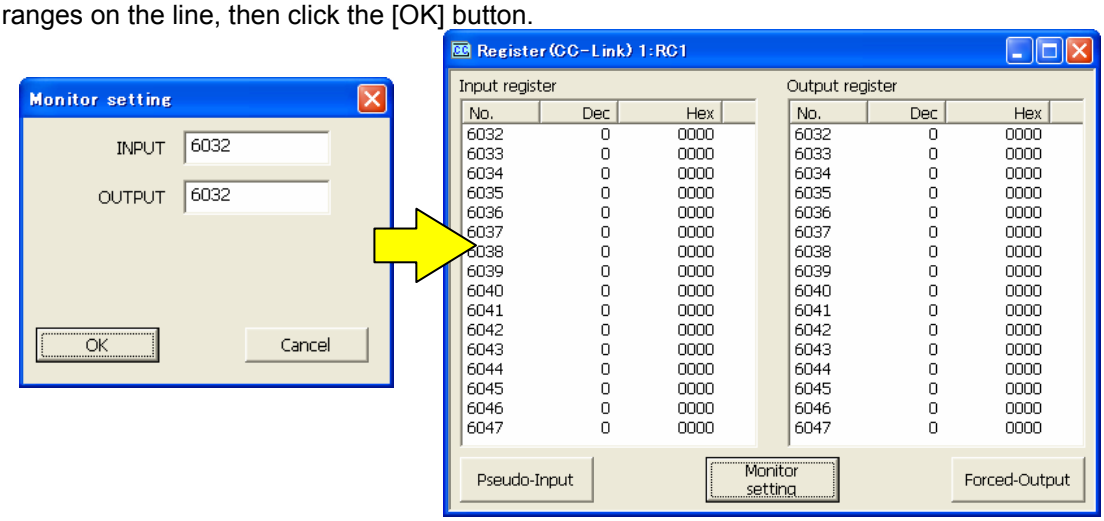


Figure 9-24 Registers (CC-Link) Monitor Settings

9.2.4.2. Pseudo-input monitoring:Pseudo-input

Pseudo-input means registers that are input to the robot controller from the computer, not from outside equipment.

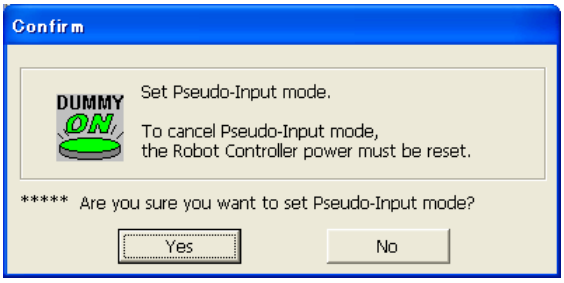
Click the [Pseudo-Input] button. A screen for inputting pseudo signals is displayed.

⚠ Caution

While the robot controller is in pseudo-input mode, register input from outside devices is not accepted.

To use pseudo-input, put the robot controller into pseudo-input mode. While the robot controller is in pseudo-input mode, the robot controller does not accept register input from outside devices.

Click the [Pseudo-Input] button. The confirmation message below is displayed before the robot controller goes into pseudo-input mode.



To release a robot controller from pseudo-input mode, switch the power for the robot controller Off, then On again.

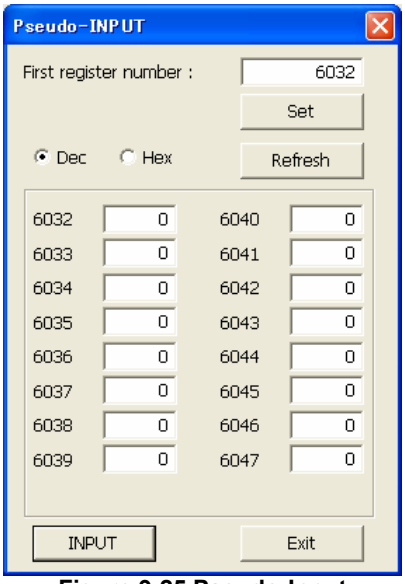


Figure 9-25 Pseudo-Input

① Read the signal you want to pseudo input.
You can set 16 registers at the same time. Input the head number for the signals you want to read, then click the [Set] button. The input statuses of the 16 registers starting from the specified head signal number are displayed.

② Set the pseudo-input status, then click the [Input] button. The set register values are pseudo input to the robot controller.

③ You can display and make pseudo input in hexadecimal for register values. If you have selected [Hex], input values as hexadecimals.

When you click the [Refresh] button, the latest register information is displayed.

9.2.4.3. Forced Output

You can forcibly output register values to outside equipment from robot controllers.
Click the [Forced Output] button. A screen for forcibly outputting registers is displayed.

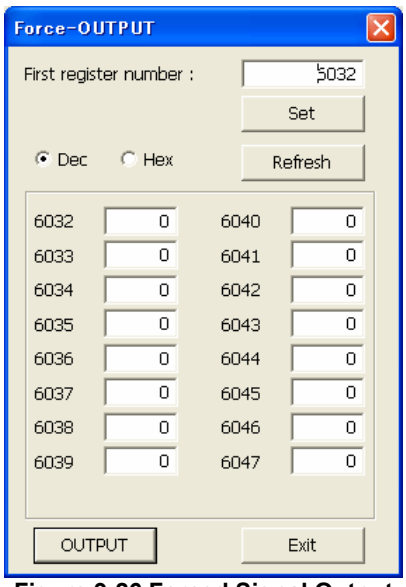


Figure 9-26 Forced Signal Output

① First, read the registers you want to forcibly output.
You can output 16 registers at the same time. Input the head number for the registers you want to forcibly output, then click the [Set] button. The output statuses of the 16 registers starting from the specified head register number are displayed.
② Set the output status, then click the [OUTPUT] button. The specified register values are forcibly output from the robot controller.
③ You can also display and forcibly output register values in hexadecimal. If you have selected [Hex], input values as hexadecimals.

When you click the [Refresh] button, the latest register information is displayed.

Caution

Forced register output

Forced output is possible if the robot controller mode is either [AUTOMATIC] or [MANUAL] (for a CRn-500 series robot controller, [TEACH], [AUTO (OP)], OR [AUTO (EXT.)]), but if even one program is running, forced output is not possible. (Except an ALWAYS program)

9.3. Production Condition Monitoring

9.3.1. Operation hours

You can check the robot work time, battery usage time, etc.

From the project tree, double click the target project [Online] → [Monitor] → [Operation Monitor] → [Operation Time].

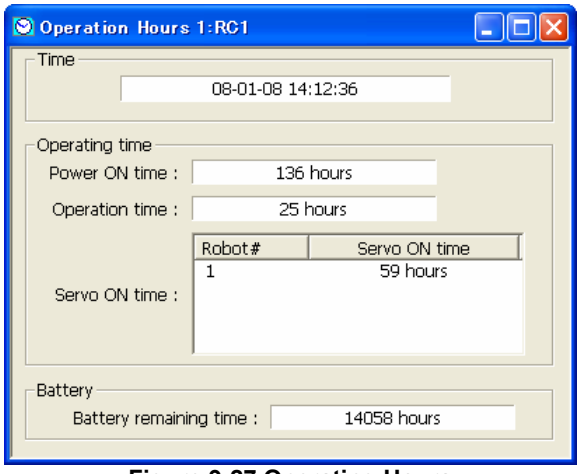


Figure 9-27 Operating Hours

You can initialize the battery remaining time with [Maintenance] → [Initialize]. For details on operation methods, see **"10.2.4 Initializing the battery remaining time"**.

9.3.2. Production information

You can check the latest tact time, run time, cycle count, and average tact time for each program in the robot controller. The latest tact time, run time, and average tact time can be measured up to 1193 hours (about 49 days).

From the project tree, double click the target project [Online] → [Monitor] → [Operation Monitor] → [Production Information].

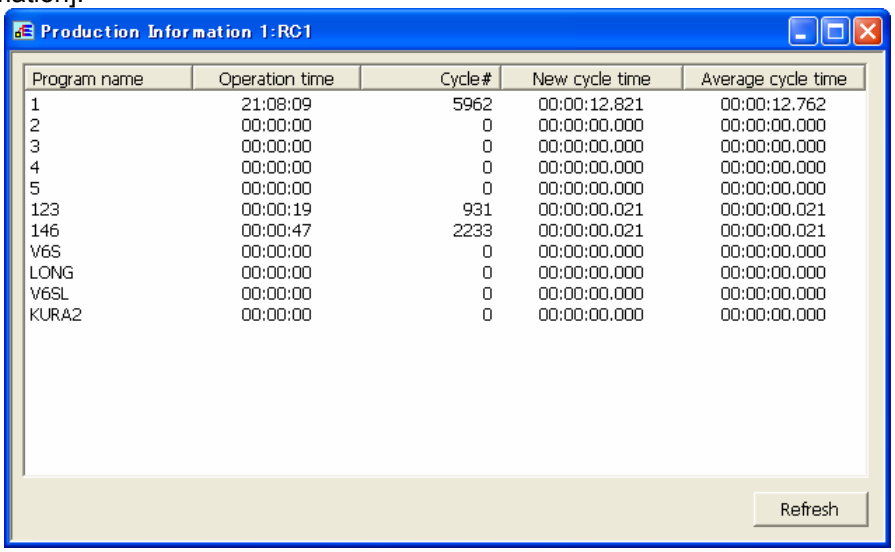


Figure 9-28 Production Information

The production information is not constantly updated. Click the [Refresh] button as necessary.

10. Maintenance

With maintenance, you can maintain the robot in various ways, including setting origin data and initializing various information.

10.1. Setting Origin Data

You can save robot origin data to a file, edit it, and transfer it to a robot controller.

Set origin data while connected to the robot controller.
From the project tree, double click the target project [Online] → [Maintenance] → [Origin Data].

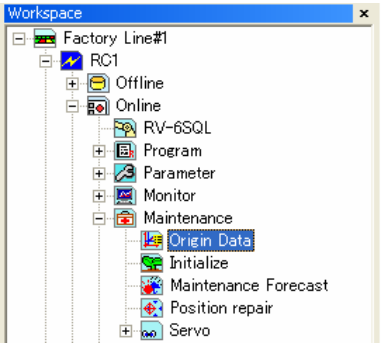


Figure 10-1 Starting up the Origin Data Setting Screen

Caution

About controller modes in which origin data can be read/written

The controller modes (TEACH/AUTO (Op.)/AUTO (Ext.)) in which origin data can be read/written depend on the CRn-500 series robot controller version. For details, see **"Table 10-1 Origin Data Reading Robot Control Operation Mode"** and **"Table 10-2 Origin Data Writing Robot Control Operation Mode"**.

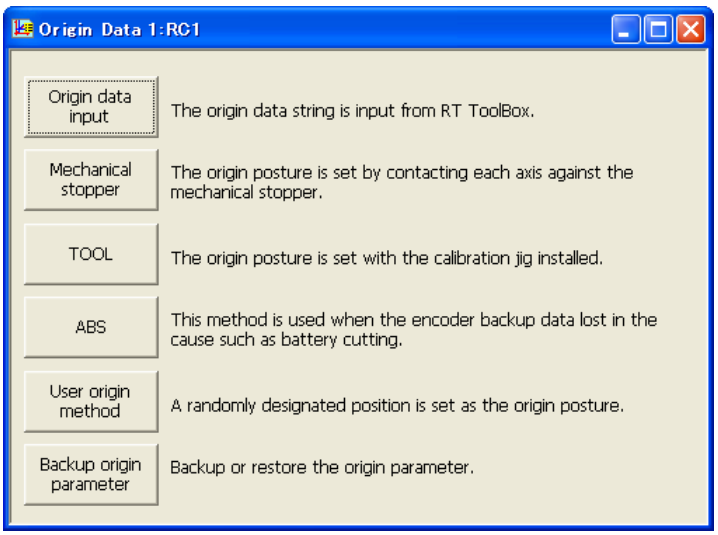


Figure 10-2 Origin Data Screen

* *About robot controller run modes when origin data is read/written*

With CRn-500 series robot controllers, when robot origin data is read/written using this software, there are restrictions on the controller run mode according to the robot controller software version.
Reference the table below.

(1) Reading

Table 10-1 Origin Data Reading Robot Control Operation Mode

CRn-500 series robot controller	Operating mode		
	TEACH	Auto (OP)	Auto (Ext)
J1 edition or later	○	○	○
H7 edition or earlier	×	×	○

○: Reading possible, ×: Reading not possible

(2) Writing

Table 10-2 Origin Data Writing Robot Control Operation Mode

CRn-500 series robot controller	Operating mode		
	TEACH	Auto (OP)	Auto (Ext)
J1 edition or later	○	○	○
G9 edition – H7 edition	○	×	○
G8 edition or earlier	○	×	×

○: Writing possible, ×: Writing not possible

10.1.1. Origin data input technique

You can save robot origin data to a file, edit it, and transfer it to a robot controller.
Click the origin data screen "origin data input technique" to display the screen.

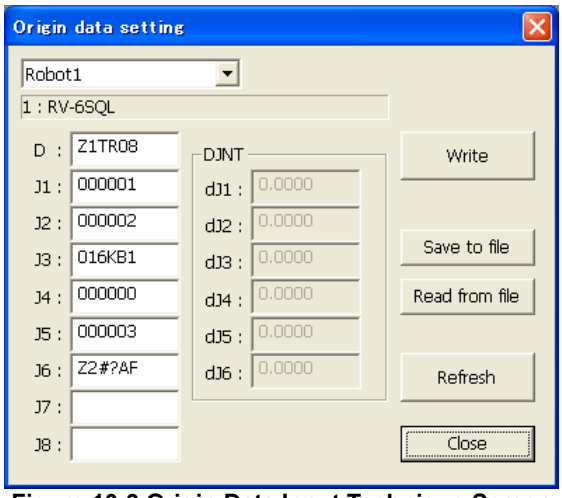


Figure 10-3 Origin Data Input Technique Screen

[Write]	:	Writes the origin data displayed on the screen to the robot controller.
[Save to file]	:	The displayed origin data can be saved to a file.
[Read from a file]	:	Stored origin data can be read from a file and displayed on the screen.
[Refresh]	:	Reads the origin data from a robot controller and displays the latest status.

Caution

Input the correct values for the J7 and J8 origin data.

For the J1-J6 axis origin data, the compatibility of values in the robot controller are checked but the J7 and J8 origin data are not checked. Always input correct values.

The J7 and J8 axis origin data is only displayed when there is a supported additional axis.

Memo

About DJNT (origin error) parameters

DJNT shows the origin position error. When revising the origin position using the position repair tool, the value is set in DJNT. (When not revising the origin position using the position repair tool, all the elements become 0. However, for RV-4A, the values are entered beforehand.)

DJNT is not released to general customers, so the values can not be directly changed.

About DJNT parameter display

Sometimes DJNT parameters are not displayed for certain robots (for example, robots that do not support the position repair function).

10.1.2. Mechanical stopper technique

This uses the robot mechanical stoppers to set the robot origin.
Click the origin data screen [Mechanical stopper] button to display the screen.

After moving the robot to a mechanical stopper origin position, select the axis to set the origin for with the checkbox, then click the [Set origin] button.
"Last" displays the origin setting technique used the previous time. For some robot controller software versions, this is sometimes not displayed.

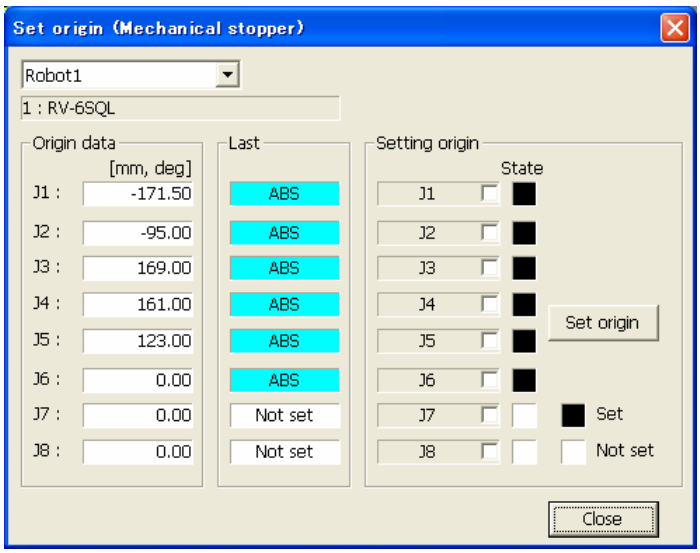


Figure 10-4 Origin Setting (Mechanical Stopper) Screen

10.1.3. Tool technique

This uses the origin setting tool to set the robot origin.
Click the origin data screen [Tool] button to display the screen.

After moving the robot to the tool origin position, select the axis to set the origin for with the checkbox, then click the [Set origin] button.
"Last" displays the origin setting technique used the previous time. For some robot controller software versions, this is sometimes not displayed.

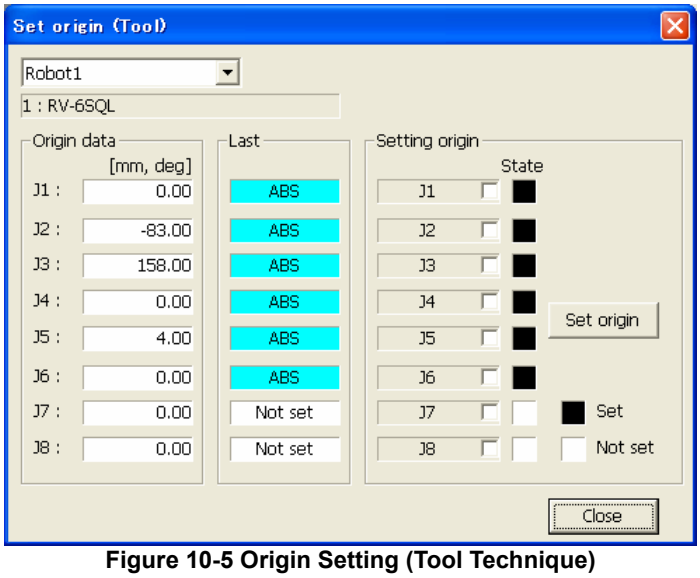


Figure 10-5 Origin Setting (Tool Technique)

10.1.4. ABS origin technique

This uses the robot's ABS origin position robot to set the robot origin.
Click the origin data screen [ABS] button to display the screen.

After moving the robot to the ABS origin position, select the axis to set the origin for with the checkbox, then click the [Set origin] button.
"Last" displays the origin setting technique used the previous time. For some robot controller software versions, this is sometimes not displayed.

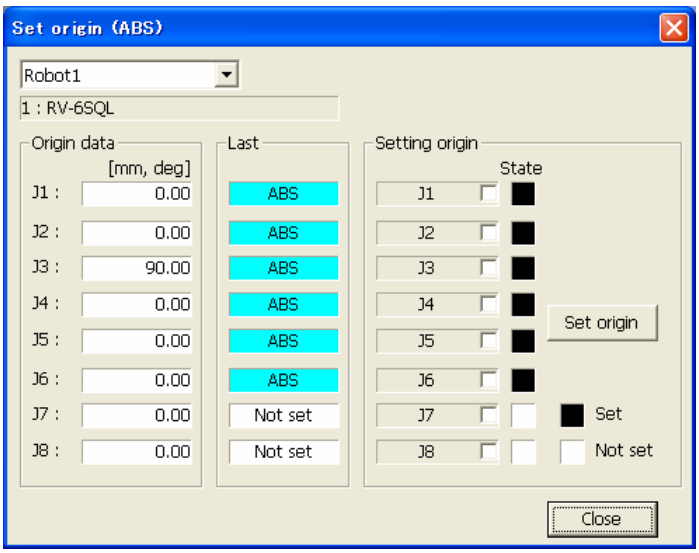


Figure 10-6 Origin Setting (ABS Origin Technique) Screen

10.1.5. User Origin Technique

This uses the robot user origin to set the robot origin.
Click the origin data screen [User origin method] button to display the screen.

After moving the robot to the user origin position, use the checkbox to select the axis to set the origin of, then click the [Set origin] button.
"Last" displays the origin setting technique used the previous time. For some robot controller software versions, this is sometimes not displayed.

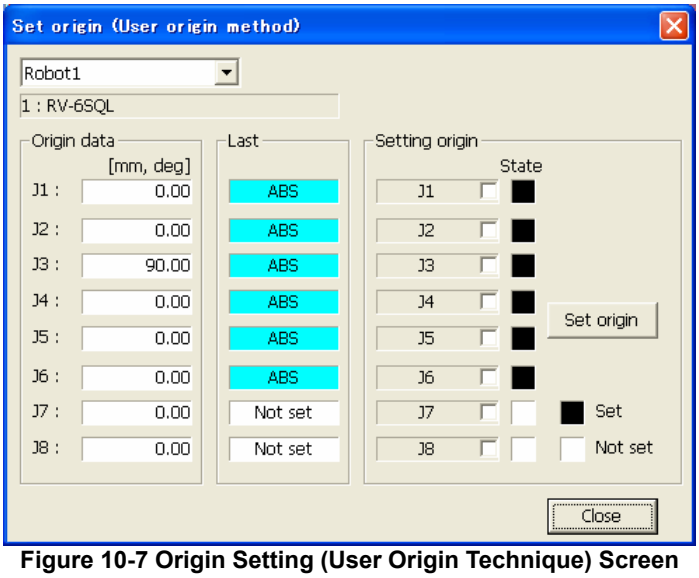


Figure 10-7 Origin Setting (User Origin Technique) Screen

10.1.6. Origin Parameter Backup

You can back up the parameters that make up the origin data. Also, you can transfer the backed-up data to a robot controller.

Click the origin data screen [Backup origin parameter] button to display the screen.

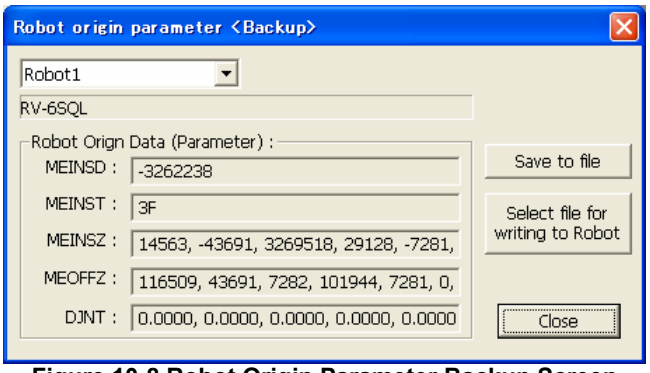


Figure 10-8 Robot Origin Parameter Backup Screen

[Save to file]	:	This saves origin parameters read from a robot controller (displayed parameters) to a file.
[Select file for writing to robot]	:	Transfer origin parameters stored in a file to a robot controller. When transferring to a CRn-500 series robot controller with edition H7 or earlier, set "Teach" mode.

10.2. Initialization

This initializes information in a robot controller.
Here, you can initialize the following information in a robot controller.
(1) Set the robot controller clock
(2) Initialize all programs in the robot controller
(3) Initialize the remaining battery time in the robot controller
(4) Check the serial number in the robot controller and set the serial number for the connected robot
(Serial number checking and setting can only be used with CRn-700 series robot controllers.)

10.2.1. Starting

Use the initialization function while connected to the robot controller.
From the project tree, double click the target project [Maintenance] → [Initialize].

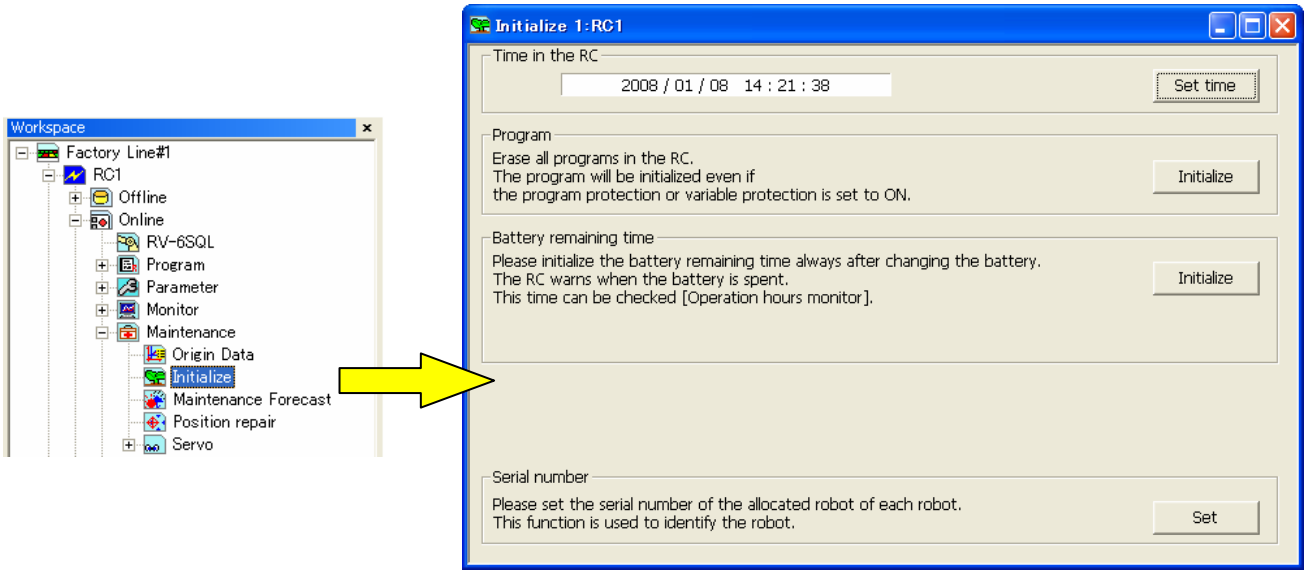


Figure 10-9 Starting Initialization Screen

10.2.2. Setting the time in the robot controller

You can set the robot controller clock.
On the initialization screen, click the [Set time] button.

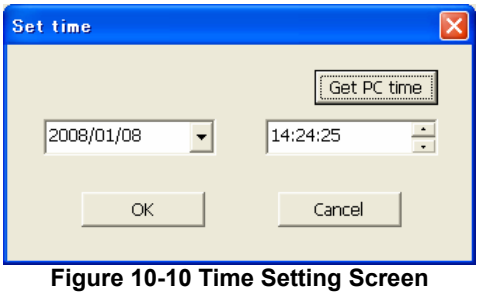


Figure 10-10 Time Setting Screen

Set the new date and time, then click the [OK] button.
You can set the current date and time from your computer by clicking the [Get PC time] button.

10.2.3. Deletion of all robot programs

This deletes all the programs in the robot controller.
On the initialization screen, click the program group [Initialize] button.
A confirmation screen is displayed, so input "Yes", then click the [OK] button.

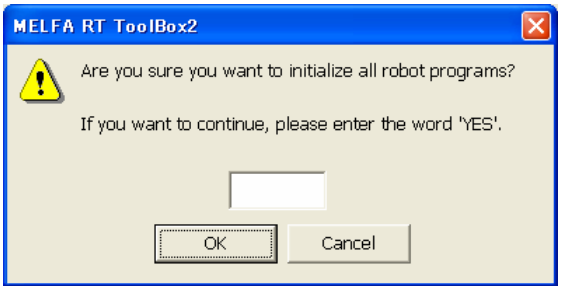


Figure 10-11 Confirmation Screen for Program File Initialization

10.2.4. Initializing the battery remaining time

This initializes the remaining battery time in the robot controller.
On the initialization screen, click the remaining battery time [Initialize] button.
A confirmation screen is displayed, so input "Yes", then click the [OK] button.

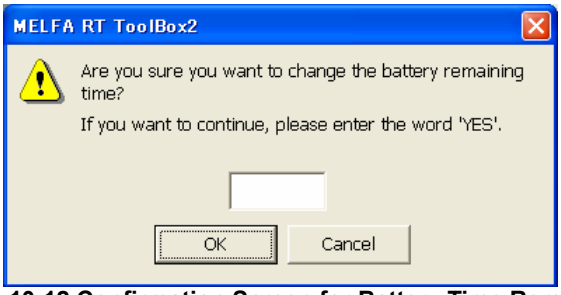


Figure 10-12 Confirmation Screen for Battery Time Remaining

⚠ Caution

Do not initialize unless the battery has been replaced.

Be careful. If you initialize the remaining battery time other than when the battery is replaced, it becomes impossible to reference a correct reading for the remaining battery time.

10.2.5. Serial number

This checks the serial number in the robot controller and sets the serial number for the connected robot
On the initialization screen, click the serial number group [Set] button.

This function can only be used with CRn-700 series robot controllers.
The first time you start up a CRn-700 robot controller after purchase, the C0150 warning (robot main unit serial number not set) is generated. On this screen, set the robot main unit serial number.

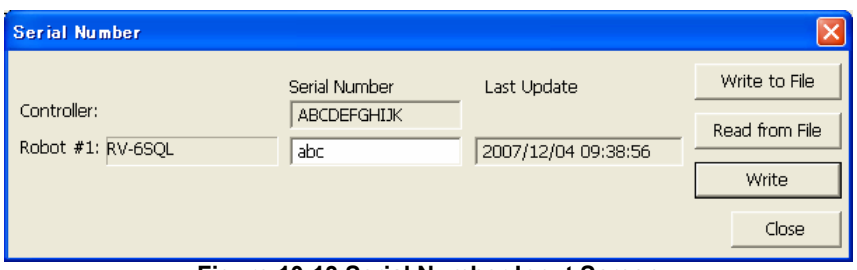


Figure 10-13 Serial Number Input Screen

10.3. Maintenance Forecasting

With "Maintenance forecasting", you can reference the parts replacement timing (greasing and battery and belt replacement) from operation data collected up till now in the robot controller.

⚠ Caution

The results of calculations in Maintenance Forecast merely show reference values.
Please execute the daily inspection and the periodic inspection to prevent the breakdown beforehand, and to secure safety.

10.3.1. Specifications

With CRn-500 series robot controllers, there are restrictions on the maintenance forecast functions according to robot models and versions supported. The robot controller software versions and models supported by the maintenance forecast function are as follows.

Table 10-3 Supported models and software versions

No.	Robot	CRn-700 series Robot controller	CRn-500 series Robot controller
1	RV-6S series RV-12S series	The supported models are not restricted	Ver. **J2** or later
2	RV-3S series RV-3SJ series		Ver. **K1** or later
3	RH-6SH series RH-12SH series RH-18SH series		Ver. **K4** or later

10.3.2. Starting

Use the maintenance forecast function while connected to the robot controller.
From the project tree, double click the target project [Online] → [Maintenance] → [Maintenance Forecast].

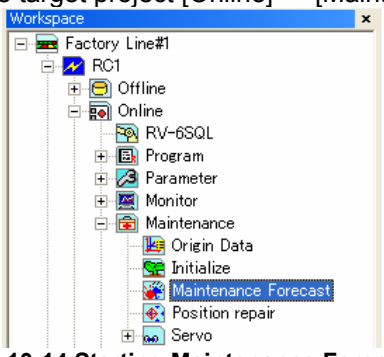


Figure 10-14 Starting Maintenance Forecasting

10.3.3. Forecasting

You can reference the "time until battery replacement", "time until regreasing, and time until belt replacement".

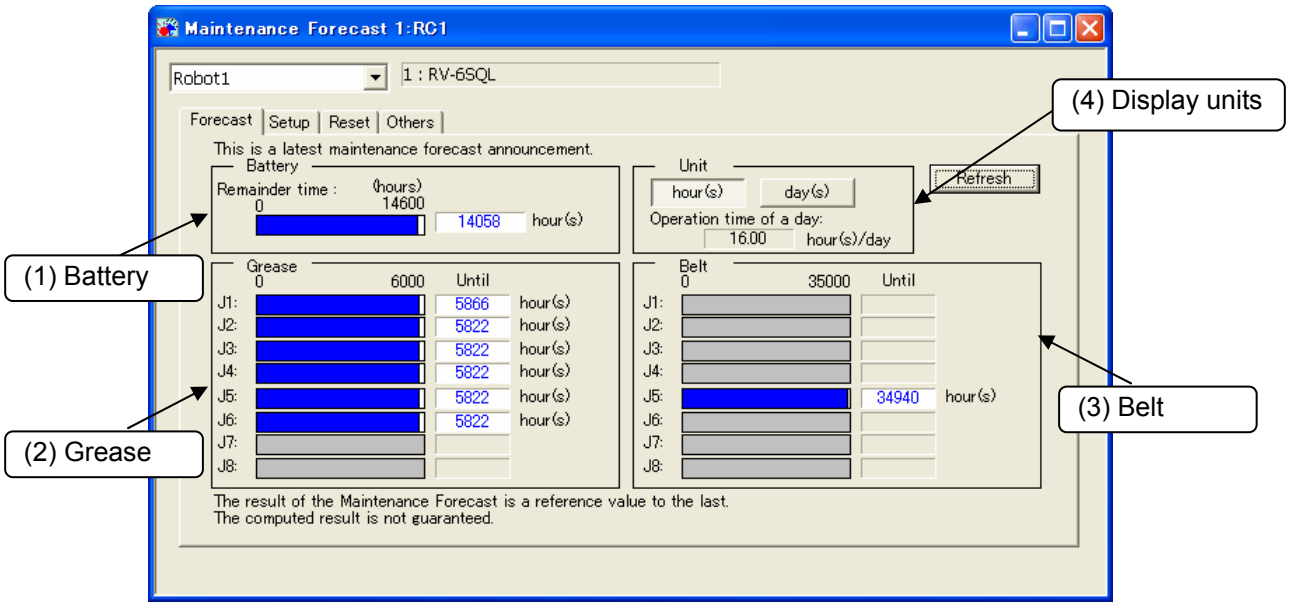


Figure 10-15 Forecast Screen

By clicking the [Refresh] button, you can reacquire information on maintenance from the robot controller.

(1) Battery

If the number of remaining hours of battery life has reached

(Remainder time) < [The remainder days until presumed maintenance time] on the Setup screen) x (24 - [Operation time of a day])

the hours and bar graphs are displayed in orange.

(The battery replacement time is calculated during the time when the controller's power is not on.)

(2) Grease

If the hours until replenishment time has reached

(Hours until replenishment time) < ([The remainder days until presumed maintenance time] on the Setup screen) x ([Operation time of a day]),

the hours and bar graphs are displayed in orange.

(3) Belt

If the hours until belt replacement time has reached

(Hours until belt replacement time) < ([The remainder days until presumed maintenance time] on the Setup screen) x ([Operation time of a day]),

the hours and bar graphs are displayed in orange.

(4) Display unit

You can switch the display units for "Grease" and "Belts" between hours and days. When days are selected as the display unit, the number of days of operation is calculated from the number of operating hours per day and that number of days is displayed.

10.3.4. Settings

Here, you can set the timing for collecting information concerning maintenance forecasts, the notification method, etc.

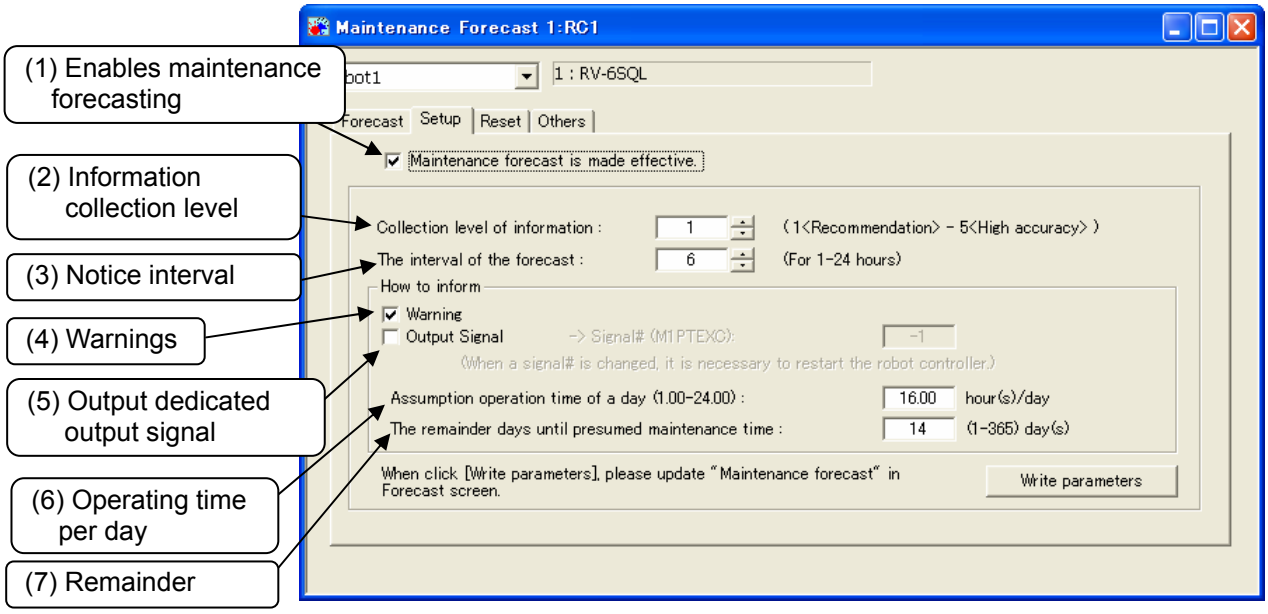


Figure 10-16 Setup

When the [Write parameters] button is clicked after setting each item, the setting values are written into the controller. All items other than the signal numbers of dedicated outputs take effect after they are written into the controller. If a dedicated output signal has been changed, it is necessary to power on the controller again.

For details on the setting items, see **"Table 10-4 Description of the Setup Screen"**.

⚠ Caution

Information needed to "Maintenance Forecast" is not accumulated while the Maintenance forecast is being invalidly set.

Factory preset value is invalidity. When the Maintenance Forecast is invalidated, information of Maintenance Forecast is not accumulated. When switched effectively from invalidity again, the reservoir of information is continued from the last value. If you have invalidated the Maintenance Forecast for a long term, the correct maintenance times cannot be calculated.

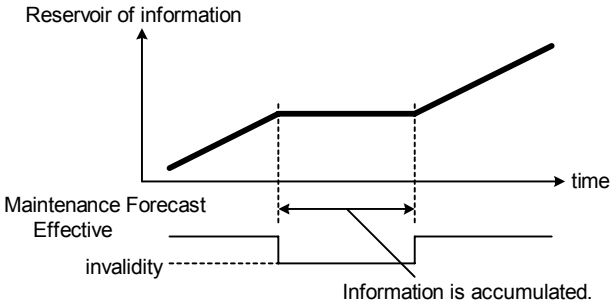


Table 10-4 Description of the Setup Screen

Item	Explanation	Factory preset value
(1) Maintenance Forecast is made effective.	If this is checked, the Maintenance Forecast function takes effect. * If a checkmark is removed, the collection of the information for Maintenance Forecast stops, and the correct maintenance times cannot be calculated.	Check ON
(2) Collection level of information	Five levels can be specified to collect the information about the maintenance. * As an information collection level gets higher, the accuracy of the maintenance improves, but it affects the tact time more.	1 (Recommended)
(3) The interval of the forecast	Specify the interval to notify the maintenance time.	6 hours
How to inform	When the grease replenishment, belt replacement and other maintenance times have reached, they can be notified by generating a warning or outputting a dedicated signal. As for the battery replacement time, one of warnings, C7500, C7510 and C7520, is generated, regardless of whether or not [Warning] under [How to inform] is checked. A warning to be generated varies depending on each situation.	
(4) Warning	If this item is checked, the maintenance time is notified as a warning. The warning numbers are listed as follows: Grease : C753* (* is the axis No.) Belt : C754* (* is the axis No.)	Check ON
(5) Output Signal	If this item is checked, signal numbers can be entered. If this item is checked and a signal number is entered correctly, the maintenance time is notified using the output of the designated signal.	Check OFF
(6) Assumption operation time of a day	Enter an estimated robot operation hours per day.	16 hours
(7) The remainder days until presumed maintenance time	Specify the number of days remaining until presumed maintenance time to be used as a reference to notify the maintenance time.	14 days

Memo

Methods for resetting the alarm and alarm signal output

As a method of notifying the replacement time of each part, an alarm (C753* and C754* (* represents the axis number)), or a dedicated output signal (M*PTEXC (* represents the robot number)) will be output.

If both are set up as the notification methods, executing the error reset operation will reset the alarm and end the signal output.

If the “alarm” method is disabled and only the output of the dedicated output signal is selected as the notification method, pushing the reset button on the front side of the controller will not end the signal output. In this case, push the [ERROR RESET] key on the teaching box or enter the error reset signal (ERRRESET) to end the signal output.

Notification method setting		Notification method	Methods to reset the notification (alarm or dedicated signal output)		
Warning	Output Signal		[RESET] key on the front of the controller	[ERROR RESET] key on the T/B	External error reset signal
☑	☐	Alarm	Will reset the alarm	Will reset the alarm	Will reset the alarm
☑	☑	Alarm and dedicated signal output	Will reset the alarm and the dedicated signal output	Will reset the alarm and the dedicated signal output	Will reset the alarm and the dedicated signal output
☐	☑	Dedicated signal output	Will not reset the dedicated signal output	Will ENABLE reset the dedicated signal output	Will ENABLE reset the dedicated signal output

10.3.5. Reset screen

The information (about battery, grease and belt) for Maintenance Forecast kept in the controller can be reset.

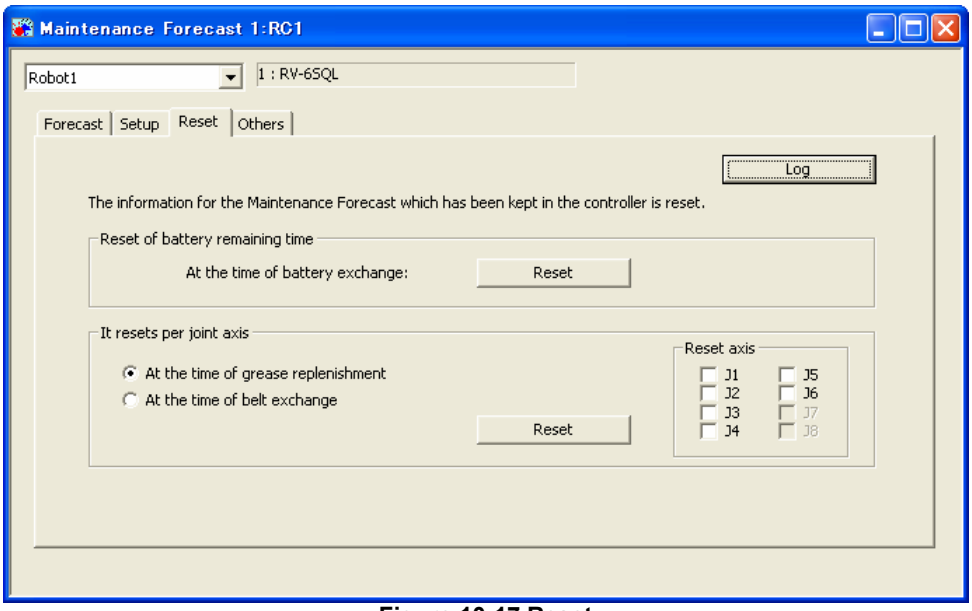


Figure 10-17 Reset

Table 10-5 Description about each reset

Types of resets	Explanation	Note
At the time of battery exchange	It is used when an alarm urging to replace the batteries (C7500, C7510 or C7520) occurs and the batteries have been replaced. Be sure to reset the battery remaining time after a battery has been replaced.	
At the time of grease replenishment	When an alarm urging to perform periodic inspections and replenish grease (alarm numbers in the 7530s) occurs, replenish the grease and reset the replenished axis.	Axes are reset in units of joint axes. Multiple joint axes can be reset at the same time.
At the time of belt exchange	When an alarm urging to perform periodic inspections and to replace the belt when it is damaged (alarm numbers in the 7540s) occurs, replace the belt and reset the axis for which the belt is replaced.	Axes are reset in units of joint axes. Multiple joint axes can be reset at the same time.

These reset operations can be executed using the teaching box. See the following section for further details.

When the [Log] button is clicked in the upper-right corner of the window, the previous reset date/time and reset count can be checked.

However, the battery reset count is not displayed.

If no reset has not made previously, “----/--/-- --:--:--“ is displayed.

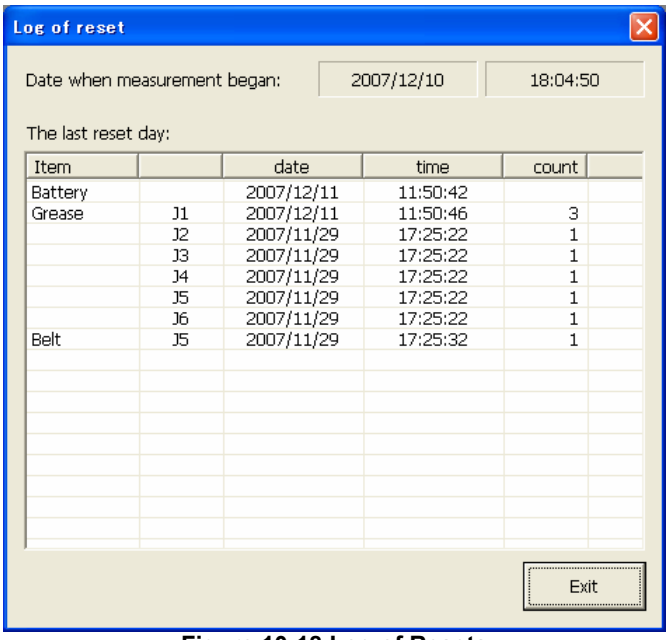


Figure 10-18 Log of Resets

10.3.6. Resetting maintenance forecast information with teaching box

When an alarm urges to replace the batteries, replenish the grease, or to replace the belt based on the Maintenance Forecast function and these parts are replaced or replenished, the information that has been accumulated within the controller needs to be reset for the axis where such replacement or replenishment has been performed.

The information that has been accumulated within the controller can be reset using not only this software, but also the teaching box.

(1) Resetting the time of battery remaining

Table 10-6 Resetting the time of battery remaining

	Explanation	Operation
the time of battery remaining	It is used when an alarm urging to replace the batteries (C7500, C7510 or C7520) occurs and the batteries have been replaced. Be sure to reset the battery remaining time after a battery has been replaced.	From the teaching box (R32TB) menu screen, execute "5. Settings and Initialization" → "1. Initialization" → Battery.

For details on the method for initializing the battery remaining time using the teaching box, in the robot controller's operations manual, see "**Details of Functions and Operations".**

(2) Resetting the grease and belt information

The grease and belt information can be reset by entering parameters to the controller.

The following is the list of parameter names and the values to be entered.

Table 10-7 Resetting the grease and belt information

	Explanation	Parameter	Value
Grease information	When an alarm urging to perform periodic inspections and replenish grease (alarm numbers in the 7530s) occurs, replenish the grease and reset the replenished axis.	MFGRST	0 : Reset information on all axes 1 to 8 : Reset information on the specified axis
Belt information	When an alarm urging to perform periodic inspections and to replace the belt when it is damaged (alarm numbers in the 7540s) occurs, replace the belt and reset the axis for which the belt is replaced.	MFBRST	

(* These parameters cannot be read not to input all characters in the teaching box.)

The grease or belt information will be reset immediately after a parameter name and the value are entered. (In this case, the controller power does not need to be restarted.) If a value other than 0 is entered, the reset process will be executed for each axis.

Repeat the parameter input operation when resetting information on two or more axes.

Also note that the value read is always 0 regardless of the previously entered value. If you continue the input operation in this state, all axes will be reset. Exercise with caution.

See “Controller INSTURCTION MANUAL – Detailed explanations of functions and operations” for how to input parameters using the teaching box.

10.3.7. Others

The information for Maintenance Forecast kept in the controller can be backed up and/or restored.

Caution

The backup and restore operations are performed when the controller (CPU) is replaced.

When the controller (CPU) is replaced, perform both backup and restore operations in a batch using the Backup/Restore tool. Also, be sure to back up the information for Maintenance Forecast before replacement, and restore the backed up information after replacement.

After the controller (CPU) has been replaced, if the information for Maintenance Forecast is not restored, or it is restored after a substantial time has elapsed since the time of backup, please note that the reliability of Maintenance Forecast will be degraded.

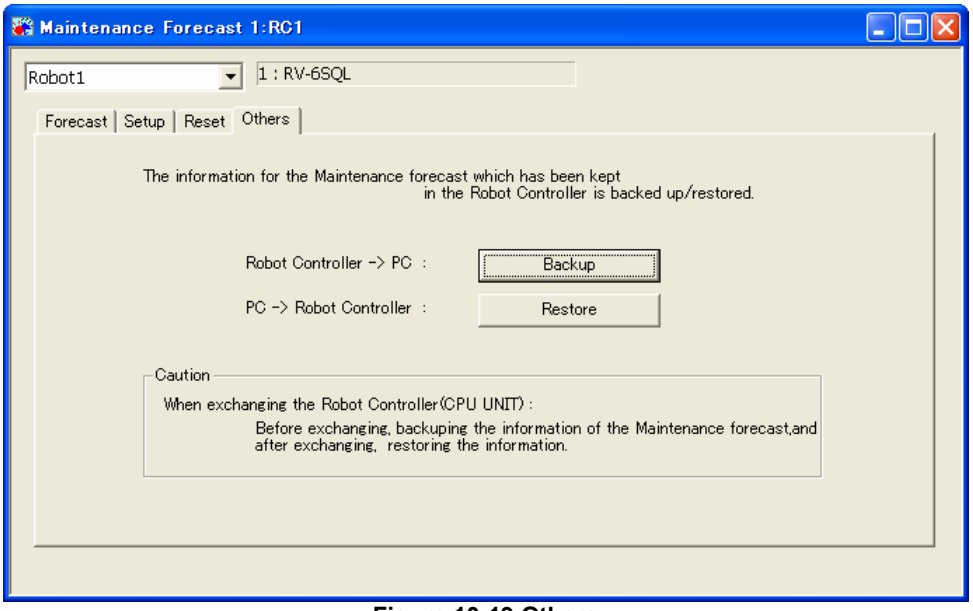


Figure 10-19 Others

10.4. Position repair Function

The position repair function is restricted by the usable models and controller software versions. See **"Table 10-9 Supported Robot Controllers and Model"**.

The "position repair function" is used when a tool is deformed by a collision or the origin is out of place because the motor has been replaced. Just reteaching part of the position data within the robot program makes it possible to use the previous position data in the controller. (Position repair generates parameters to correct the position deviation and corrects all the position data in the robot controller.)

However, please understand that there are some cases that position repair can not restore, such as applications requiring high precision and major mechanical damage to a robot from a collision.

Also, restrictions on a robot's degrees of freedom can make it impossible to recover with position repair. Since vertical 5-axis robots and horizontal 4-axis robots are restricted as shown in **"Table 10-8 The limit by degree of freedom"**, positional deviations related to these restrictions can not be corrected with this function. In this case, either reteach manually or correct the deviating section (for example, by replacing a bent hand).

Table 10-8 The limit by degree of freedom

No.	Robot model	The limit by degree of freedom
1	Vertical 5-axis robot	It can't move in the direction of C element of the Cartesian position.
2	Horizontal 4-axis robot	It can't move in the direction of A, B element of the Cartesian position.

⚠ Caution

The position repair function is only supported by MELFA-BASIC IV and MELFA-BASIC V.

The position repair function is only supported by MELFA-BASIC IV and MELFA-BASIC V. It cannot be used with Movemaster commands.

10.4.1. Specifications

The robot models and robot controller versionw with which the position repair function can be used are as follows.

Table 10-9 Supported Robot Controllers and Models

No.	Robot model	CRn-700 series Robot Controller	CRn-500 series Robot Controller
1	Vertical 6-axis robot	The supported models are not restricted by the version	Version **J2** or later Only correction of origin data is supported in versions prior to **J2**.
2	Vertical 5-axis robot		Version **K1** or later Only correction of origin data is supported in versions prior to **K1**.
3	Horizontal 4-axis robot (RH-SH series only)		Version **K4** or later Any versions prior to **K4** are not available. Moreover, This function cannot be used for the RH-AH series robot.

10.4.2. Starting

Use the position repair function while connected to the robot controller.
From the project tree, double click the target project [Online] → [Maintenance] → [Position repair].

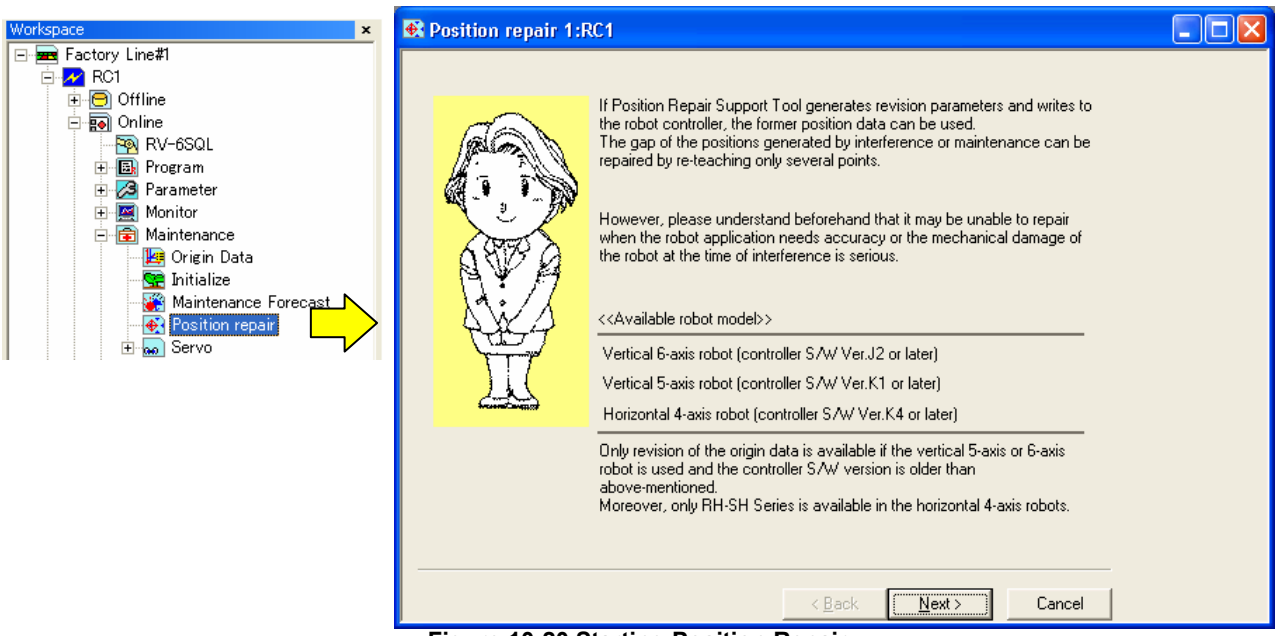


Figure 10-20 Starting Position Repair

10.4.3. Flow of operations

The position repair takes the form of a wizard. You can automatically generate the parameters by proceeding with operations according to the instructions on each screen. You can directly set parameter values.

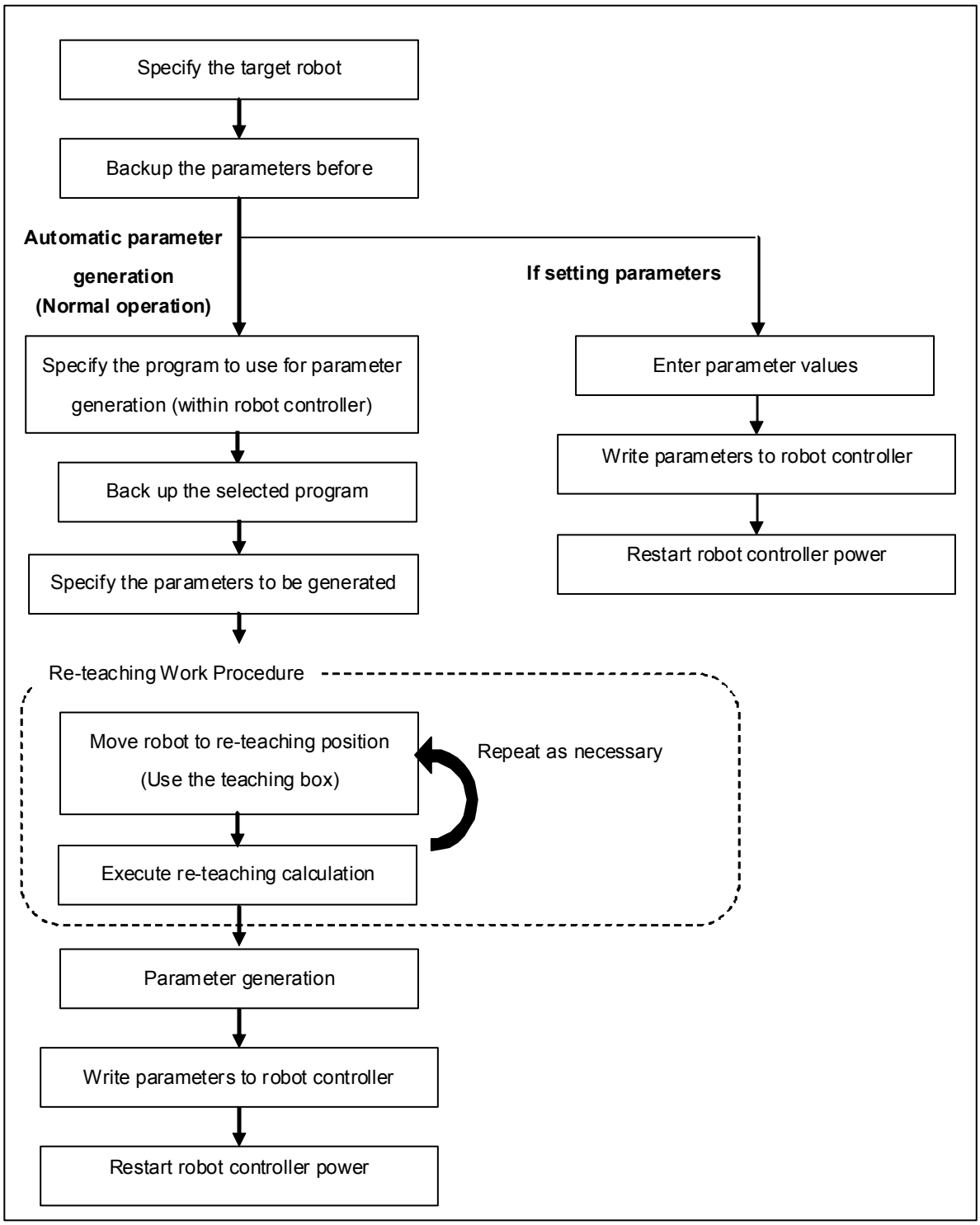


Figure 10-21 Operation Flow

The explanation follows the normal operations flow. For the explanation when setting parameter values, see **"10.4.16 Revision parameter editing".**

10.4.4. Introduction

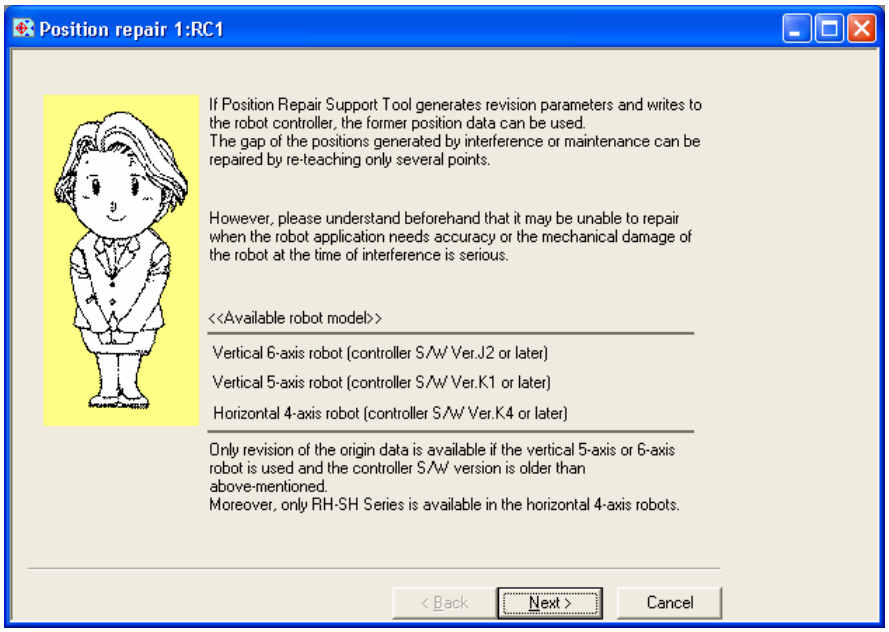
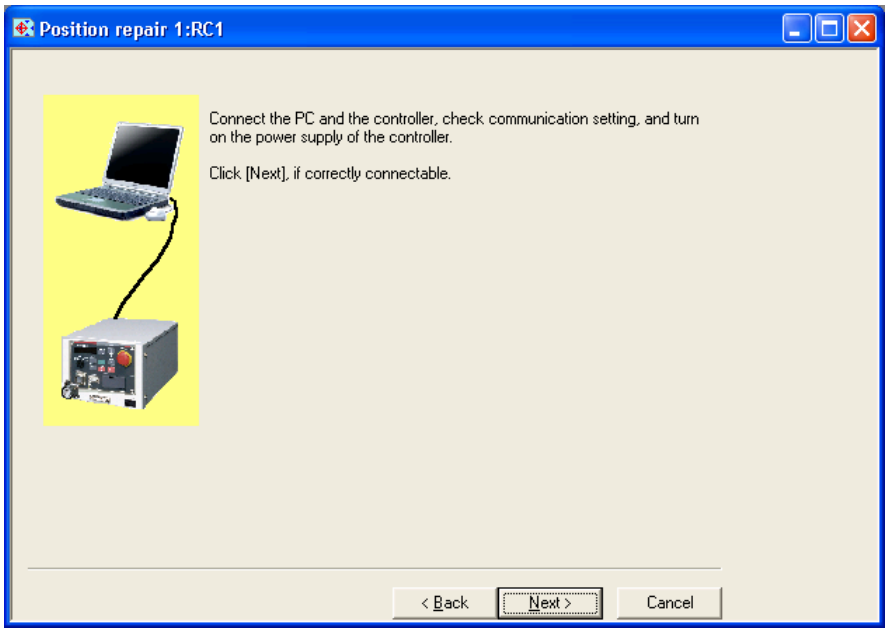


Figure 10-22 Starting Use Window

This is an explanation of the position repair function. Read it carefully, then click the [Next] button.

10.4.5. Communications settings



Check the communication settings and connected to the robot controller, click the [Next] button. For the setting method, see **"5 Connecting with the Robot".**

10.4.6. Robot selection and parameter backup

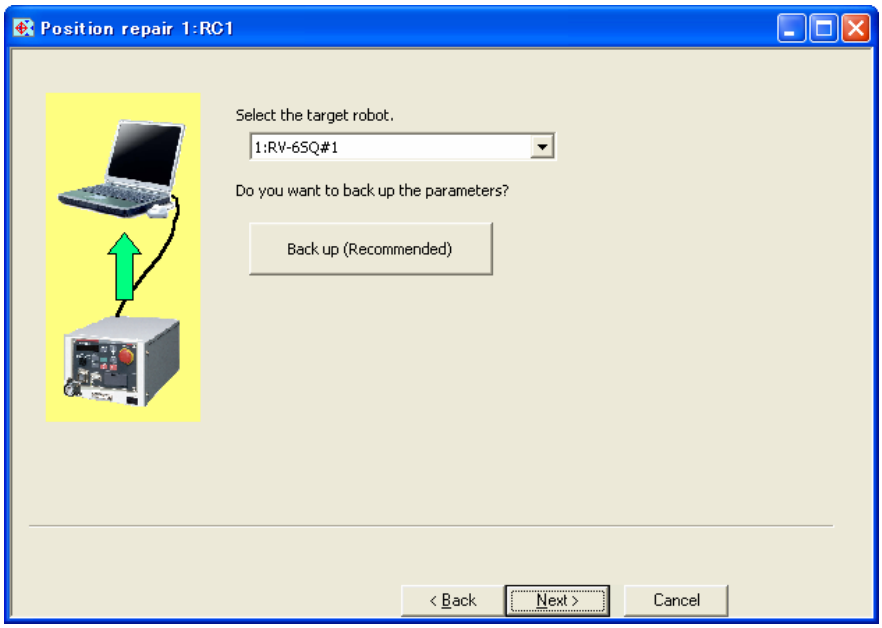


Figure 10-23 Robot and Backup Parameters Selection Window

Select the robot to execute the re-teaching.
The Robot is displayed as follows.

Ver.1.1 or earlier	Controller number : Controller Name + Mechanism Name #Mechanism No.
Ver.1.2 or later	Controller number : Mechanism Name #Mechanism No.

Displayed only in multi-mechanism mode.

To backup parameters, click [Backup].
The dedicated backup screen starts. For more details on backups, see **"12 Backup and Restore".**
Backed up parameter files can be written back to a robot controller using the "backup/restore" functions of this software.

When the preparations are complete, click the [Next] button.

Memo

Parameter Backup

During its operation, this software overwrites parameters to the robot controller.
It is recommended that the parameters be backed up at this point to allow the controller to revert to the original parameters.

10.4.7. Revision parameter generation procedure selection

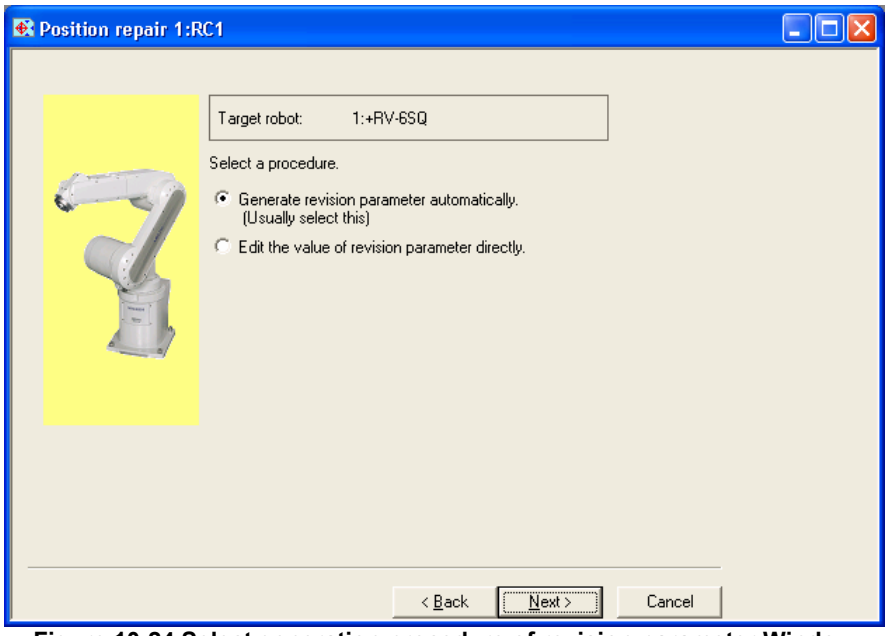


Figure 10-24 Select generation procedure of revision parameter Window

In the next step, the software can either automatically generate parameters or accept manually entered parameter values. Normally, [Generate revision parameter automatically] is selected.

Select [Generate revision parameter automatically] and click [Next] to proceed to **"Select Program"** window.
Select [Edit the value of revision parameter directly] to proceed to **"Edit Revision Parameter"** window.

10.4.8. Program selection

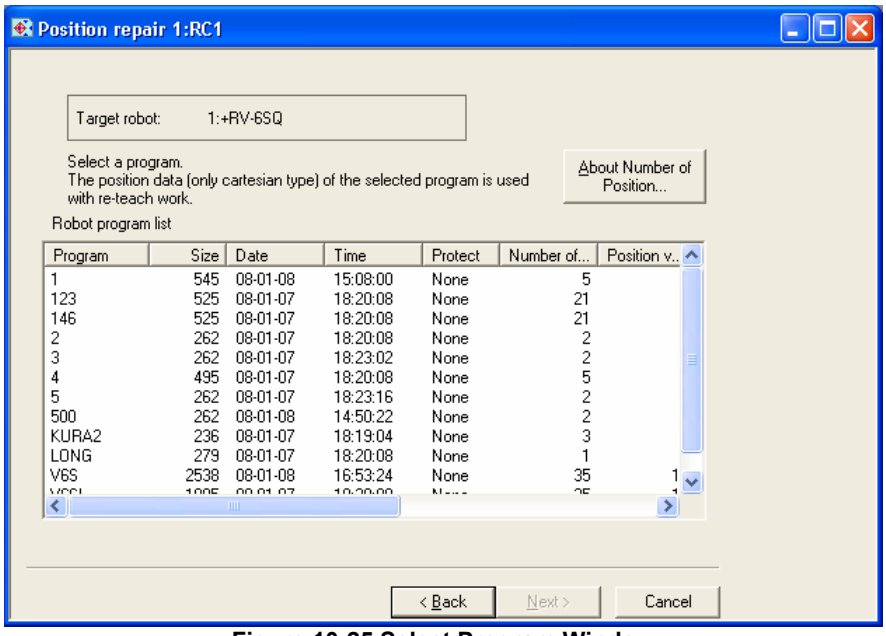


Figure 10-25 Select Program Window

Select the robot program to use for revision parameter generation, then click the [Next] button.
Here, perform the reteaching using the XYZ-coordinate position data in the selected program.

For details on the required position data numbers, see **"Table 10-10 Selecting Revision Parameters".**

Memo

The points on selecting the program

Select the program with the positions of various location and posture.
Moreover, higher accuracy of revision parameters can be obtained by selecting the following type of position data program.

- Program with positions that are easy to re-teach
- Program with positions that require high precision

10.4.9. Program reading and backing up

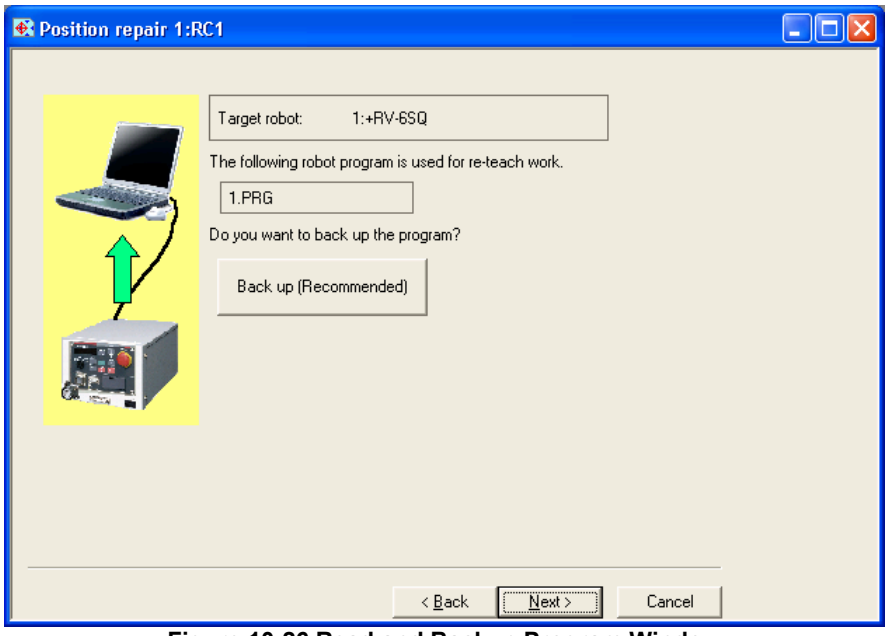


Figure 10-26 Read and Backup Program Window

To backup a program, click [Backup].
The special screen for backup starts. For details on backups, see **"12 Backup and Restore".**
You can use the backup/restore functions in this software to write a backed up parameter file back into a robot controller.

When the preparations are complete, click the [Next] button.

Memo

Robot Program Backup

During its operation, this software may overwrite robot controller program (position data).
It is recommended that the program be backed up at this point to allow the controller to revert to the original program.

10.4.10. Tool setting check

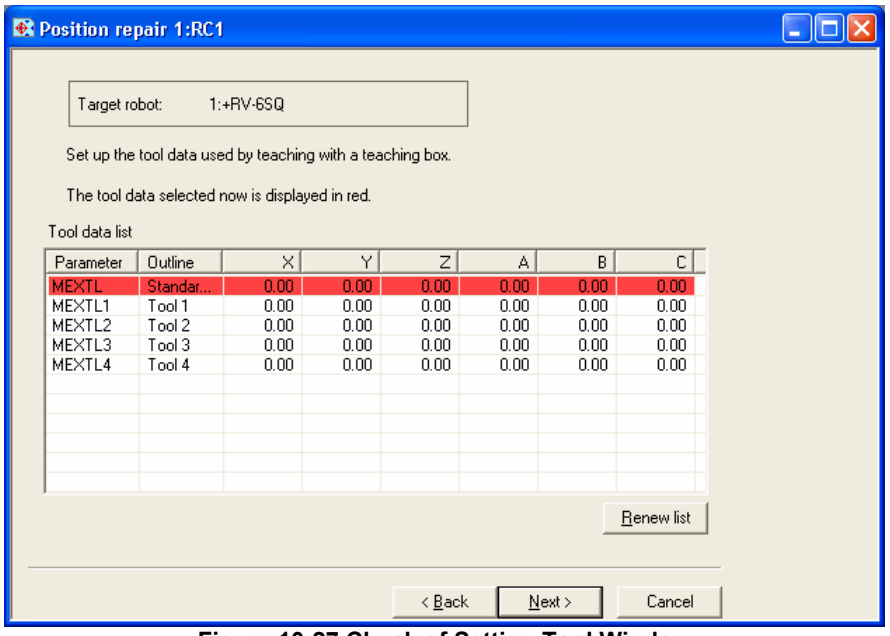


Figure 10-27 Check of Setting Tool Window

Parameter values set in the present robot controller for tool data are displayed. The row for the tool selected by the tool number (MEXTLNO) is highlighted in red.

Please check if the tool data and tool number used during teaching is set.

If necessary, change the value from parameter setting in teaching box. Click [Renew List] to update the contents of the display.

(If the CRn-500 series robot controller's version is older than **J2**, only the standard tool (MEXTL) will be displayed.)

Click [Next] when ready to proceed.

⚠ Caution

Do not change tool data or base data.

After this window, do not change tool data or base data.

If they are changed during re-teaching operation, re-teach calculation cannot be done correctly.

When correcting tool data, if teaching was performed switching back and forth between multiple tools, perform re-teaching operation for each tool.

10.4.11. Revision parameter selection

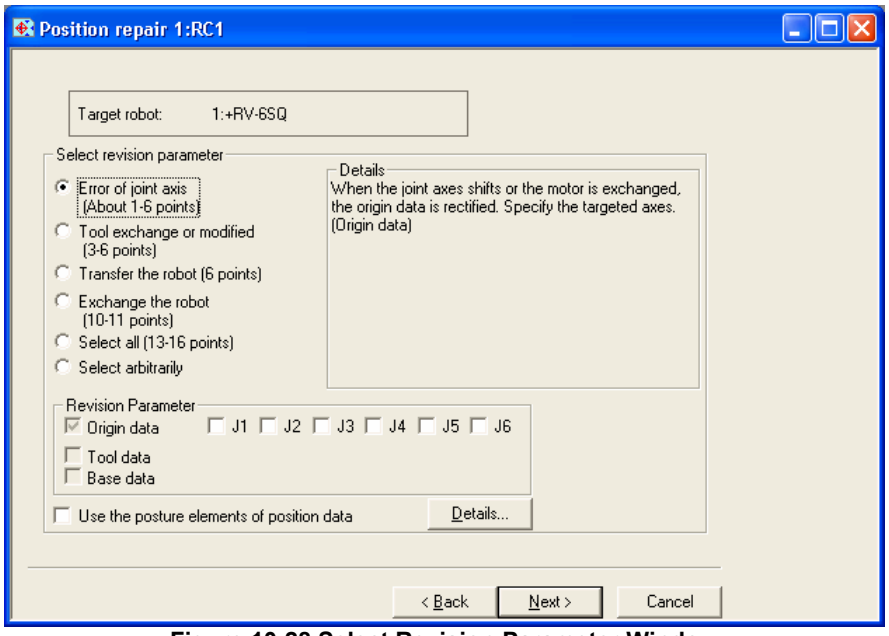


Figure 10-28 Select Revision Parameter Window

Select the revision parameter that becomes the target for re-teaching calculation.

Revision parameter will be selected automatically if an item is selected from [Select revision parameter].
Choose [Select all] to select all the revision parameters. If you wish to specify a particular combination of revision parameters, choose [Select arbitrarily] and specify the revision parameters.

Vertical 6-axis robot	If the CRn-500 series robot controller's version is older than **J2**, only [Error of joint axis] can be selected.
Vertical 5-axis robot	If the CRn-500 series robot controller's version is older than **K1**, only [Error of joint axis] can be selected.
Horizontal 4-axis robot (RH-SH series only)	The CRn-500 series robot controller's any versions prior to **K4** are not available. (This function cannot be used for the RH-AH series robot.)

The supported models are not restricted by the version in CRn-700 series robot controller.

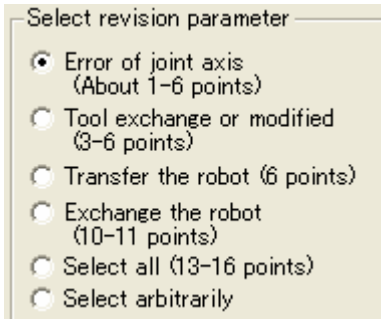
Click [Details] to see the description of the difference between checking and not checking [Use the posture elements of position data].

In the following section, details regarding revision parameters and posture elements of position data are explained.

After choosing the revision parameters, click [Next].

10.4.11.1. Revision parameters

Parameters revised by items selected with "Select revision parameters" become as in **"Table 10-10 Selecting Revision Parameters".**



Memo

Some elements cannot be calculated according to the robot type and the combination of revision parameters.

In this function, the amount of the gap of the robot is calculated as a correction value, and the revision parameter is generated. However, some elements cannot be calculated (the value becomes 0) as the following two kinds of cases.

*** The case which cannot be calculated by the limits of degree of freedom of robot**

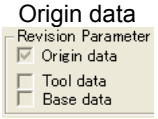
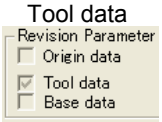
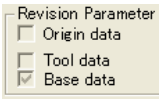
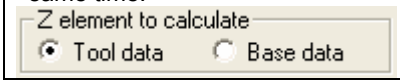
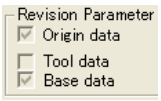
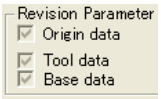
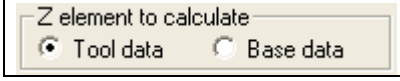
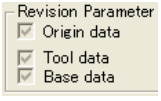
Some elements of revision parameter cannot be reflected because the Vertical 5-axis robot and horizontal 4-axis robot have the limitation.

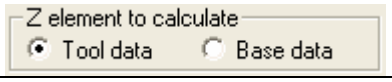
*** The case which condensed by the combination of robot mechanism and revision parameter**

Some elements of revision parameter become the value on the same rotation axis according to the combination of robot mechanism and revision parameter. In such case, calculated value of gap is condensed to the one element of revision parameter.

At this case, though the other element becomes 0, it condenses in other elements and it is corrected. So it is not necessary to reflect it again.

Table 10-10 Selecting Revision Parameters

No.	Item	Description	Revised Parameter	Minimum number of teach points		
				Vertical 6-axis robot	Vertical 5-axis robot	Horizontal 4-axis robot
1	Error of joint axis	Rectifies origin data when joint axis moves or when motor is replaced. Specify the target axes using the check boxes. The number of teaching points is different according to how the axis was specified.	Origin data 	1 to 6 points	1 to 5 points	1 to 4 points
2	Tool exchange or modified	Rectifies attachment error when robot tool is exchanged. In addition, rectifies tool data error when the tool is transformed due to interference between robot and peripheral devices. **Vertical 5-axis robot:** * Only Z element of position data is corrected.	Tool data 	3 to 6 points	1 point	3 to 4 points
3	Transfer the robot	Rectifies base data of robot position setup when the robot is transferred to another location. **Vertical 5-axis robot:** * Only X, Y, Z elements of position data are corrected.	Base Data 	6 points	3 points	4 points
4	Exchange the robot	When robot is exchanged with the tools on, rectifies origin data error and base data of robot position setup. Only for horizontal 4-axis robot, attachment error is also rectified. **Vertical 6-axis robot:** * Origin data J1 is included in base data. **Vertical 5-axis robot:** * As to base data, only X, Y, Z elements are Corrected. **Horizontal 4-axis robot:** * Origin data J1 and J3 are included in base data. * Origin data J3 and J4 are included in tool data. * Select which to be requested because Z elements of tool data and base data are not corrected at the same time. 	Base data Origin data Tool data (4-axis robot only)  (5-axis and 6-axis robot)  (4-axis robot)	10 to 11 points	7 to 8 points	7 to 8 points
5	Select all	Selects all revision parameters. **Vertical 6-axis robot:** * Origin data J1 is included in base data. * Origin data J6 is included in tool data. **Vertical 5-axis robot:** * As to tool data, only Z element is corrected. * As to base data, only X, Y, Z elements are corrected. **Horizontal 4-axis robot:** * Origin data J1 and J3 are included in basedata. * Origin data J3 and J4 are included in tooldata. * Select which to be requested because Z elements of tool data and base data are not corrected at the same time. 	Origin data Tool data Base data 	13 to 16 points	8 to 9 points	7 to 8 points

No.	Item	Description	Revised Parameter	Minimum number of teach points		
				Vertical 6-axis robot	Vertical 5-axis robot	Horizontal 4-axis robot
6	Select Arbitrarily	Specify revision parameters. **Vertical 6-axis robot:** * Since origin data J1 is included in base data, if base data is selected, turn off the Checkbox of origin data J1. * Since origin data J6 is included in tool data, if tool data is selected, turn off the Checkbox of origin data J6. **Vertical 5-axis robot:** * As to tool data, only Z element is corrected. * As to base data, only X, Y, Z elements are corrected. **Horizontal 4-axis robot:** * If base data is selected, turn off the Checkboxes of origin data J1 and J3. Origin data J1 and J3 are included in base data. * If tool data is selected, turn off the Checkboxes of origin data J3 and J4. Origin data J3 and J4 are included in tool data. * If tool data and base data are selected together, select which Z element to be requested, because Z elements of tool data and base data are not corrected at the same time. 				

* Revision parameter names correspond to the following.
  Origin data: DJNT
  Tool data: MEXDTL, MEXDTL1 to 4 (Parameter of the tool selected by tool number)
  Base data: MEXDBS

10.4.11.2. Position data posture components

Position data of MELFA-BASIC IV consists of tip position (X, Y, Z) and tip posture elements (A, B, C) (*1).
This section describes the cases where [Use the posture elements of position data] is checked and not checked.

Memo

(*1) The posture elements of position data

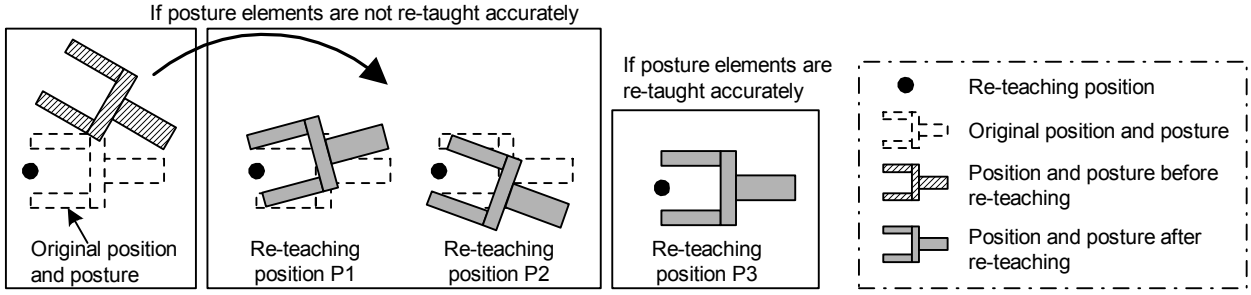
In case of the vertical 6-axis robot, the posture elements of position data are (A, B, C).
In case of the vertical 5-axis robot, the posture elements of position data are (A, B).
In case of the horizontal 4-axis robot, the posture element of position data is (C).

(1) [Use the posture elements of position data] is checked ☑ Use the posture elements of position data

Not just the robot tip position (X, Y, Z) but also the tip posture elements are used for position correction calculation.

Precision of generated revision parameter improves if the tip posture elements are also re-taught correctly.

However, if the tip posture elements are not re-taught correctly, as shown in the diagrams below (re-teaching positions P1 and P2), error occurs in position correction calculation, decreasing the precision of the calculation result.



(2) [Use the posture elements of position data] is not checked ☐ Use the posture elements of position data

Posture elements in the position data taught during re-teaching are not used for position correction calculation.

If it is not necessary to match exactly the tip posture elements during re-teaching, clear the checkbox [Use the posture elements of position data]. In such case, position correction calculation is performed using only the tool tip position data (X, Y, Z), ignoring the error from posture deviation. This increases the precision of location correction.

However, there are some restrictions. For details, see **"Table 10-11 About Posture Elements of Re-teaching Position Data".**

Table 10-11 About Posture Elements of Re-teaching Position Data

Condition	Merit	Note
When using posture elements of position data ☑ Use the posture elements of position data	Precision of generated revision parameter improves if the tip position (X, Y, Z) and tip posture elements are re-taught correctly.	During re-teaching, posture must be taught correctly. If posture data is incorrect, ***precision of revision parameter actually decreases.***
When not using posture elements of position data ☐ Use the posture elements of position data	During re-teaching, revision parameters can be generated simply by correctly teaching position (X, Y, Z). (Posture elements need not be accurate.)	**In case of the vertical 6-axis robot :** * Posture elements (A, B, C) of tool revision parameters cannot be obtained. * J6 axis of origin revision parameter cannot be obtained if both X and Y components of the tool parameter are 0.0.
		In case of the vertical 5-axis robot : * J6 axis of origin revision parameter cannot be obtained.
		In case of the horizontal 4-axis robot : * Posture elements (C) of tool revision parameters cannot be obtained. * J4 axis of origin revision parameter cannot be obtained if both X and Y components of the tool parameter are 0.0.

10.4.12. Reteaching work

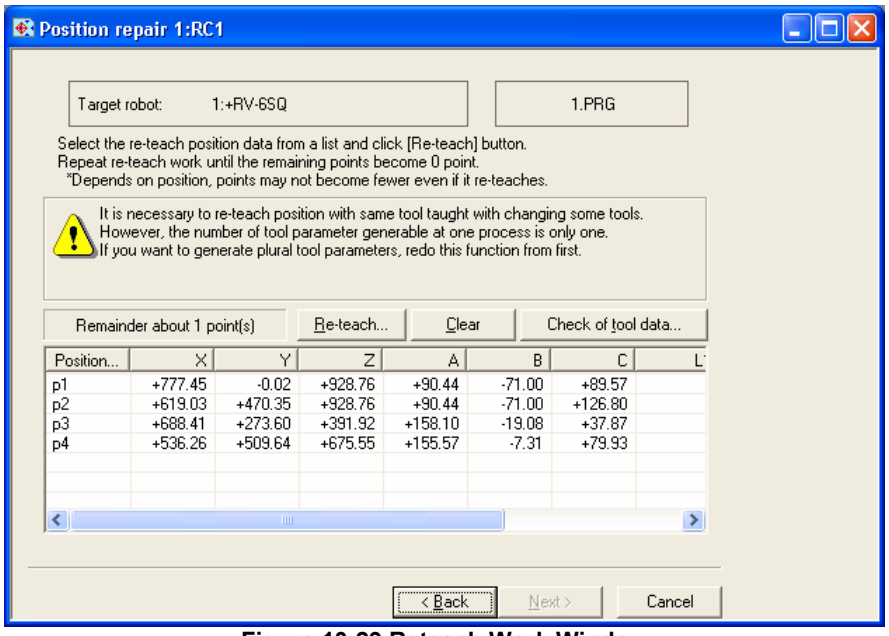


Figure 10-29 Reteach Work Window

[Remainder] Displays the number of remaining points until revision parameters are generated.

Remainder about 1 point(s)

However, at some positions, re-teaching may not decrease the number of remaining points.

[Re-teach] button — Specifies the positions selected in the list and opens **"Re-teach the position"** screen.

[Clear] button — Clears the re-teaching information for positions selected in the list.

[Check of tool data] — Displays current tool data setting in the robot controller.

⚠ Caution

Position data of the targeted program is write-protected.

During showing this window, the position data of the targeted program in the controller is write-protected. If this tool is interrupted when not communicating with the controller, the position data cannot be unprotected. Please release the protect by using the Teaching Box or Program manager of this software.

Position data for the program selected are displayed.
Select the position to re-teach from the list and repeat re-teaching to generate revision parameters.

Re-teaching work procedure can be described as follows. While the "Re-teach the position" screen is open, move the robot to the re-teaching position and click the [Load current position] button on the screen.

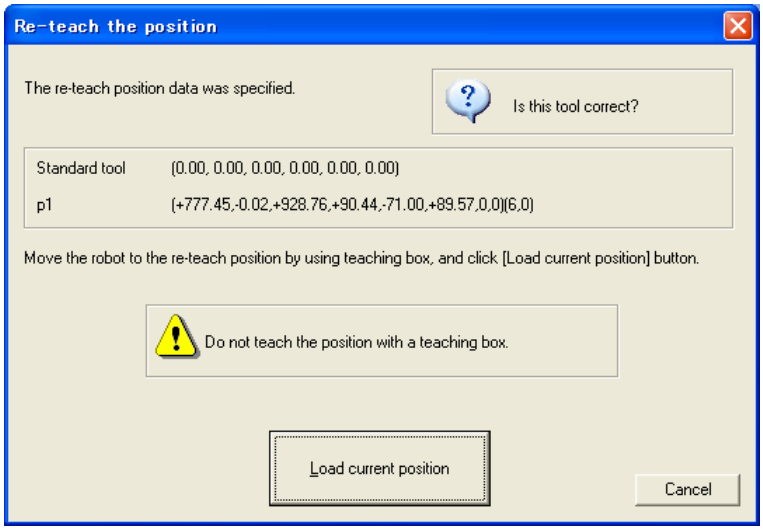


Figure 10-30 Re-teach the position Screen

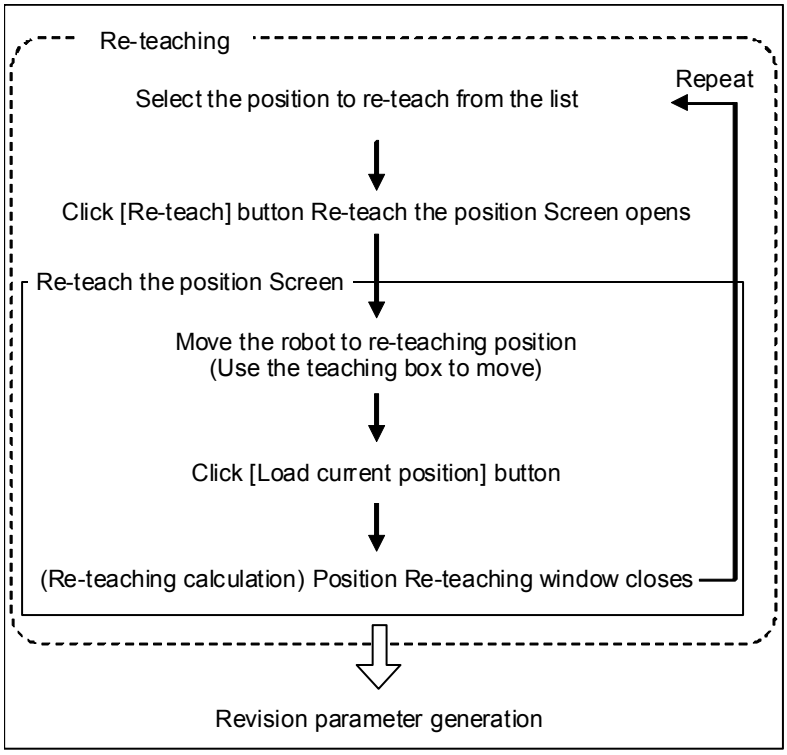


Figure 10-31 Re-teaching Work Procedure

Row for re-taught position will be highlighted in light blue.
Re-teaching does not change the position data values shown in the list.

⚠ Caution

Do not perform position correction using the teaching box.

When you move the robot to the re-teaching position using the teaching box, be careful not to correct the position.

During re-teaching, position data of the applicable program in the controller is write-protected.

Caution

Do not change tool data, tool number, or base data.

Do not change tool data, tool number, or base data during re-teaching. Re-teaching calculation will not be performed correctly.

When correcting tool data, if teaching was performed switching back and forth between multiple tools, perform re-teaching operation for each tool.

Caution

Select position data with a different posture element, when re-teaching two or more positions.

Select position data with a different posture element, when re-teaching two or more positions.

When position data of the same posture element are selected, there is a possibility that the parameter is not correctly calculated.

Caution

The cautions when using a robot with the additional axis.

When restoring the position with a robot with a travel axis, reteach at a position where the travel position becomes the same as in the original position travel axis data. (Move the robot so that the travel axis data becomes the same as the original position.) If the retaught position travel axis data differs from the original travel axis data, it is impossible to find the correct revision parameters.

It is possible to change the revision parameters to be generated.

Go back one step to the **"Select revision parameter"** window to change the setting. Note that if you return one more step to **"Check of setting tool"** window, all information set by re-teaching work will be cleared.

Caution

When go back to "Check of setting tool" window, all information set by re-teaching work will be cleared.

10.4.13. Writing parameters

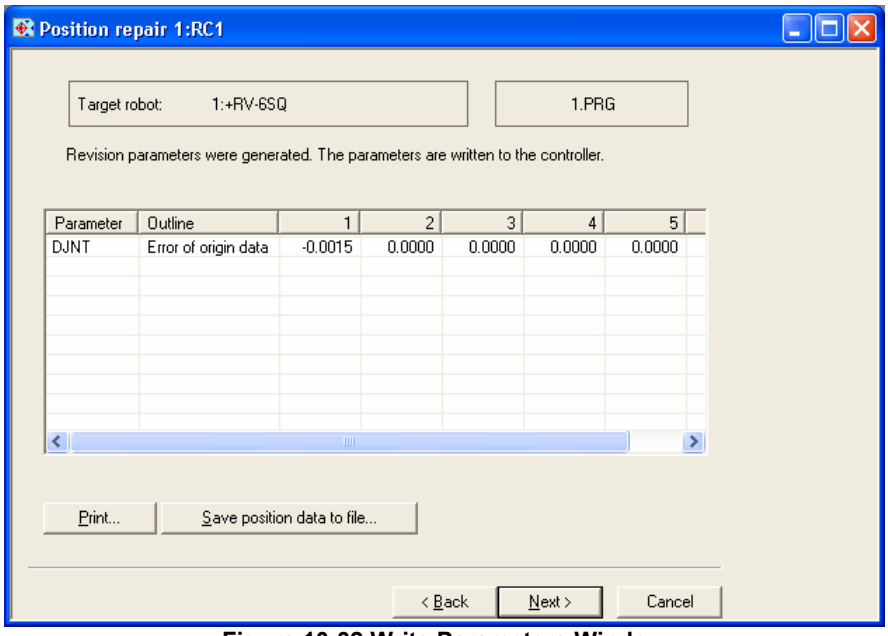


Figure 10-32 Write Parameters Window

[Print] Prints the revision parameter information displayed in the list.

[Save position data to file] Saves position data used in re-teaching as a robot program with positions only. Position data will be values converted by the revision parameters.

Revision parameters and their values generated by re-teaching are displayed.
Click [Next] button to write the parameters into the robot controller.

⚠ Caution

If revision parameters could not be generated

If revision parameters could not be generated, parameters are not displayed in the list.
If you click the [Next] button, position data used in re-teaching is written into the robot controller.
Since parameters are not generated, position data will not be converted.

Parameters may not be generated under the following conditions.
* When one of the specified re-teaching positions is of a significantly low precision
* When one of the original position data is of a significantly low precision
* When the difference between the original position data and the re-teaching position is too large
* When tool data or base data was changed during re-teaching

Clicking the [Back] button and redoing a part of the re-teaching may generate revision parameters.
Please delete the re-teaching information for the position data that meets one of the criteria mentioned above and perform re-teaching again.

10.4.14. Controller power supply Off, On

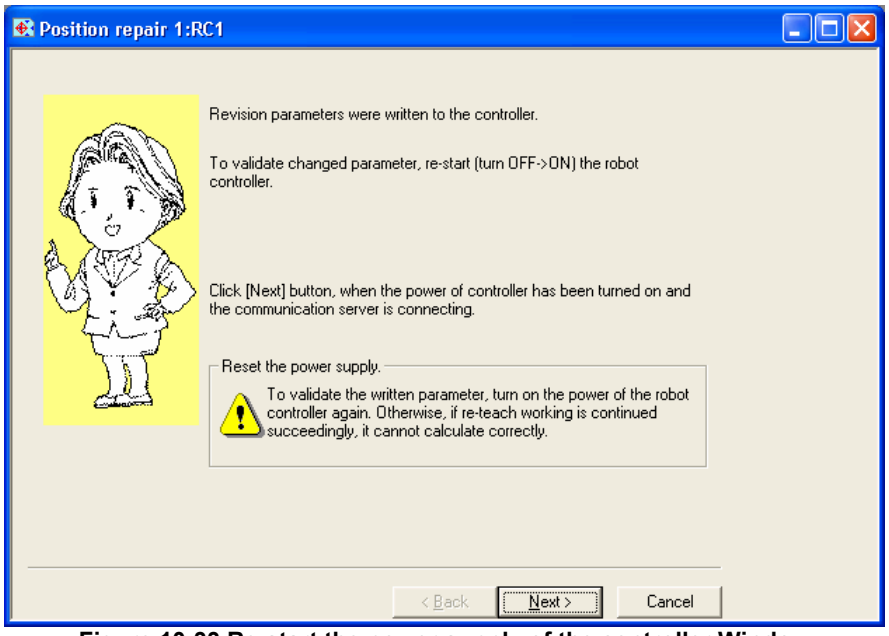


Figure 10-33 Re-start the power supply of the controller Window

To activate the written parameters, turn off and then turn on the power of robot controller.

10.4.15. Exit

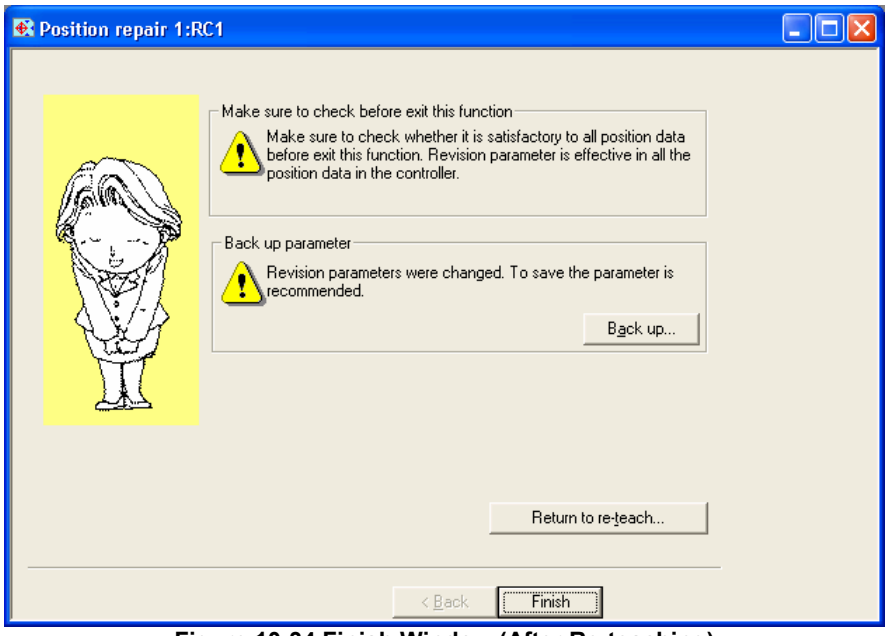


Figure 10-34 Finish Window (After Re-teaching)

When the writing of revision parameters is done, operation of this function is complete.

Caution

Perform an operation check before exit this function.

Before exit this window, make sure that all position data works properly.
If revision is not correct, click the [Return to re-teach] button to continue re-teaching. (However, if you exited from “Edit revision parameters” window, [Return to re-teach] button will not be shown.)

Caution

Back up the parameters.

This Function has changed the revision parameters. Back up the parameters before exit this window.

Caution

Position data close to operation area boundaries may not be rectifiable.

Around the operation area boundaries, position error may put a point outside the operation area, in which case this function cannot rectify the point.

10.4.16. Revision parameter editing

When you select "Edit the value of revision parameter directly" with **"10.4.7 Revision parameter generation procedure selection"**, this screen is displayed.

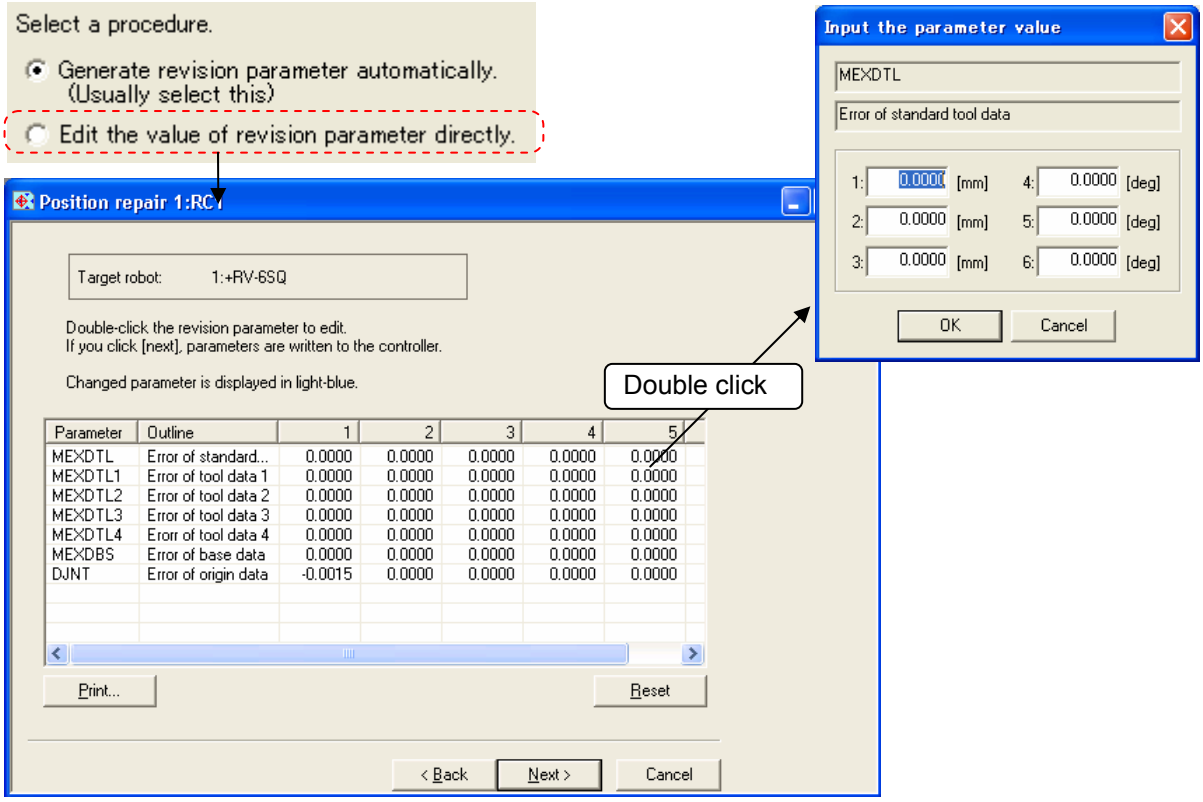


Figure 10-35 Edit Revision Parameters Window

[Print]	Prints the revision parameter information displayed in the list.
[Reset]	Resets all changes.

Displays values of current revision parameters in the robot controller.
Select parameter from the list and double-click it to display the setting screen. Set the parameter values.
(If the CRn-500 series robot controller's version is older than **J2**, only the origin revision parameter (DJNT) is shown.)

Click [Next] to write all parameters into the robot controller and proceed to **"Re-start the Power Supply of the Controller"** window.

10.5. Servo Monitor

This monitors servo information.

10.5.1. Load

The load state of each robot axis can be monitored.
Click [Reset] button to reset the max axis load level of monitored load information.

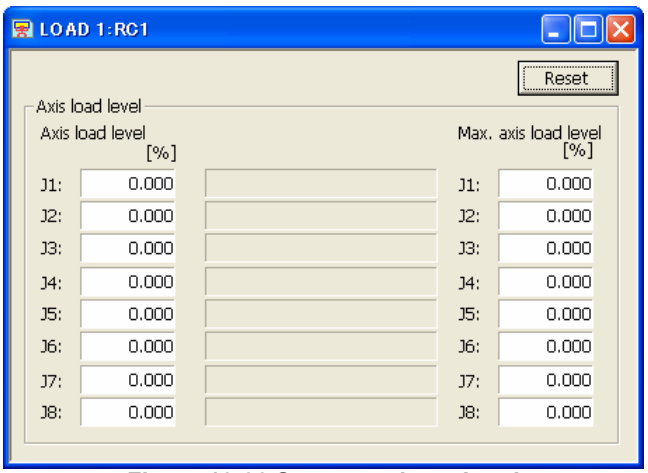


Figure 10-36 Servo monitor Load

11.Option Card

You can check information on option cards mounted in the robot controller.

When you open [Option Card] on the project tree, the slots in which option cards are currently mounted and the option card names are displayed. If you place the mouse cursor on an option card name, the information for that option card is displayed.

If no option card is mounted on the robot controller, nothing is displayed.

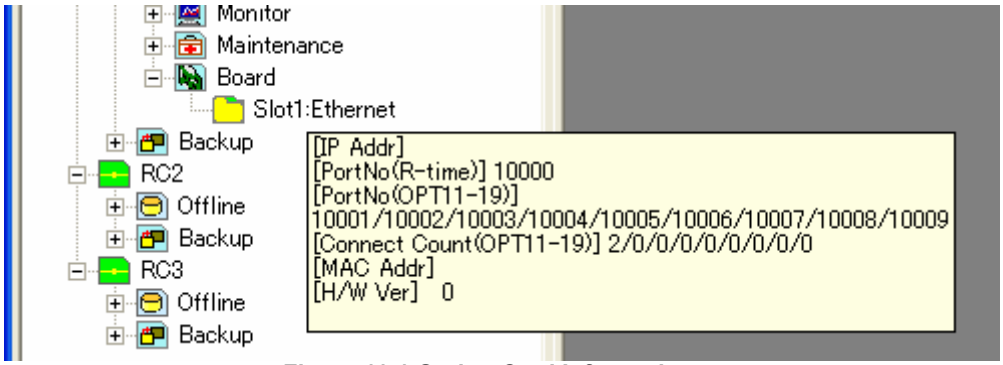


Figure 11-1 Option Card Information

12. Backup and Restore

You can back up information in a robot controller to the computer.
You can also restore backup information saved to the computer back into a robot controller.

Backup (Robot -> Personal computer)	Saves the backup data on the robot controller to the personal computer.
Restore (Personal computer -> Robot)	Transfers the backup data saved on the personal computer to the robot controller.

⚠ Caution

Precautions when executing a backup/restore operation during the replacement of a controller (CPU) that supports Maintenance Forecast

When executing a backup/restore operation during the replacement of a controller (CPU) that supports Maintenance Forecast, also perform the backup/restore operation using the Maintenance Forecast tool.

After a backup operation is performed on a controller that supports Maintenance Forecast, the following message is displayed:

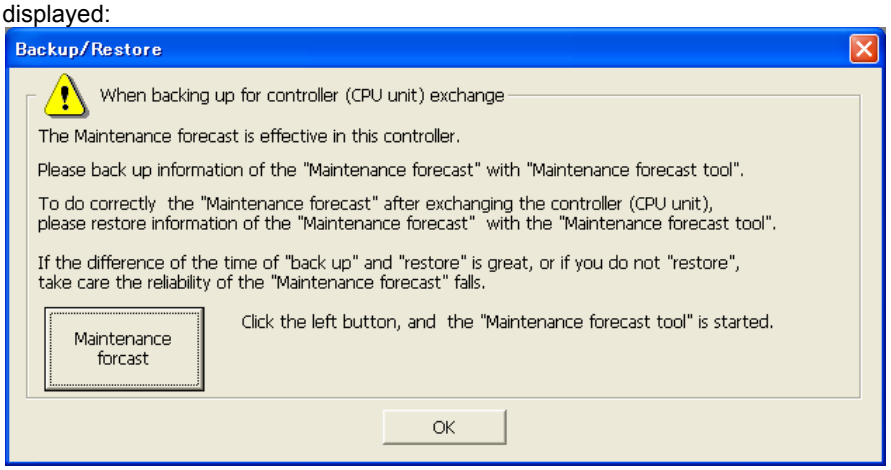


Figure 12-1 Backup When Maintenance Forecasting Information Is Enabled

* When using a CRn-500 series robot controller, maintenance forecasting is supported for software versions J2 and later.

12.1. Backup(Robot -> PC)

You can save information in a robot controller to a file in the computer.
Use the backup function while connected to the robot controller.

(1) From the project tree, open the target project [Backup]. In the backup tree, "All file", "Program information", "Parameter files", and "System program" are displayed.

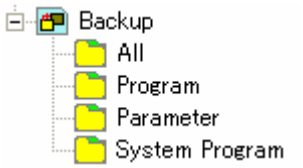


Figure 12-2 Backup Project Tree Diagram

All Files	Saves all files (robot program, parameter files, etc.) in the robot controller into the designated folder.
Program	Saves the robot program file into the designated folder.
Parameter Files	Saves the parameter files into the designated folder.
System Program	Saves the system base program file into the designated folder.

(2) Double click the items to backup.

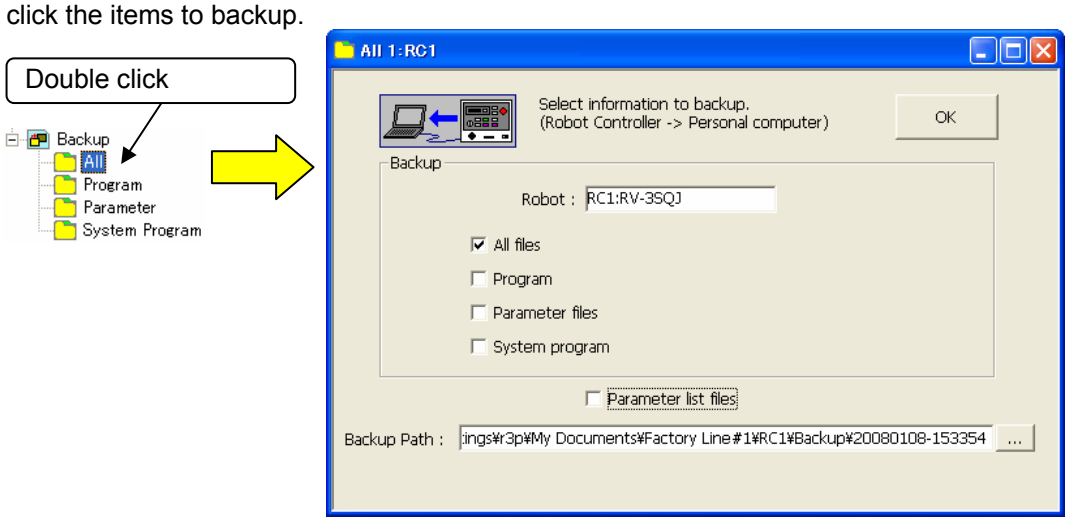


Figure 12-3 Backup (When Batch Is Selected)

[Parameter List Files] : This is used to edit the parameter information saved by backup in offline mode, and is not required for backup. If this is not checked, the time required to save all files will be shortened.

(3) Specify the backup destination. The default value is the folder that created the workspace/project name/Backup/today's date and time.

You can change the backup destination folder with the […] button at the right of the displayed backup destination.

You can back up other items at the same time too by putting checkmarks in their checkboxes. In this case, the data is backed up to the selected backup destination, with the identifier by the backup type is added to the folder name. The identifiers are as follows.

All…ALL Program information…PRG Parameter information…PRM System program…SYS

The folder name at this time is displayed on the right side of the respective backup items.

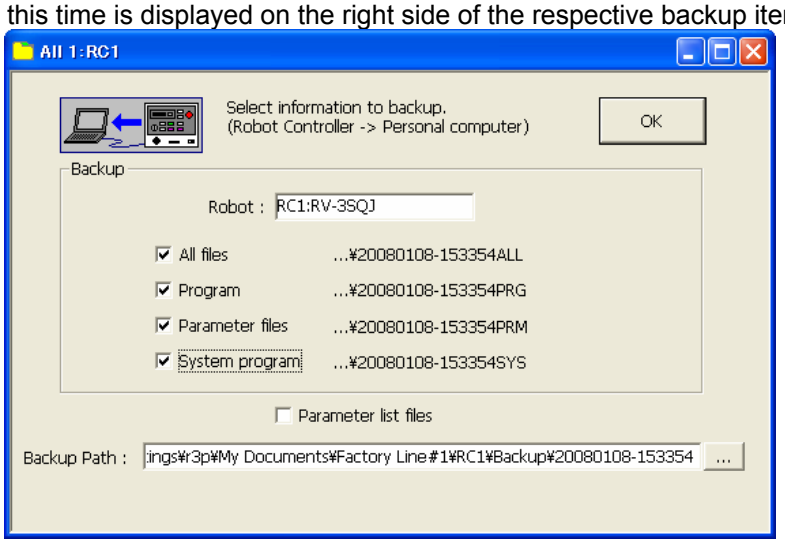


Figure 12-4 Backup (When Multiple Selected)

(4) Specify the back destination, then click the [OK] button.
The confirmation screen is displayed. When you confirm, then click [Yes], the backup is started.

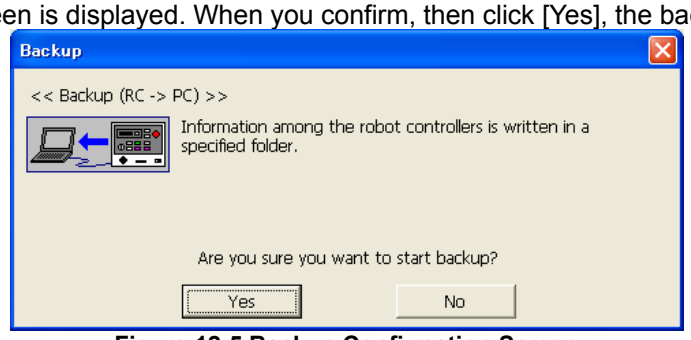


Figure 12-5 Backup Confirmation Screen

When the backup is completed, the backup data is displayed at [Backup] on the project tree.

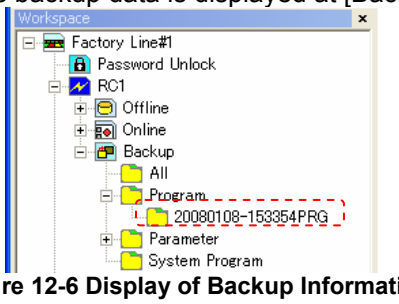


Figure 12-6 Display of Backup Information

⚠ Caution

Cautions for backups

When backing up system status variable values and program external variable values, switch the robot controller power Off, then On again, then perform the backup operations.

The files BKUP.SYS and MECHA.SYS are automatically created in the specified folder. These files record the saved robot controller mechanical information and save format. Be careful. If you delete or rewrite these files, this may make it impossible to restore them to the robot controller.

The objective of backup data is to back up robot controller information. However, with this software version 1.2 or later, it is possible to open a backed up program with program editor. Please refer to **"7.2.4 Opening a program in the backup data"** in this manual for details.

12.2. Restore (PC -> Robot)

You can take information back up to the computer and transfer it to robot controller information.
Use the restore function while connected to the robot controller.

Caution

Caution of when data backed up by selecting "All files" is individually restored.

In version 1.2 or later of this software, it is possible to restore the each item (Program, Parameter files and System program) of data backed up by selecting "All files".
At this time, the data files of the specified item are transferred to the robot controller, without making all information in the robot controller cleared (initialized).

(1) From the project tree, open the target project [Backup]. In the backup tree, the information back up for "All file", "Program information", "Parameter files", and "System program", respectively, is displayed.

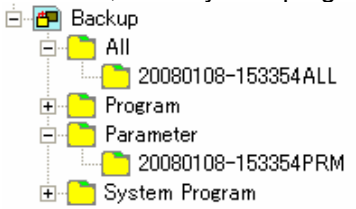


Figure 12-7 Backup Project Tree Diagram

(2) Select the information listed on the controller, then click the right mouse button. From the right mouse button menu, select [Restore].

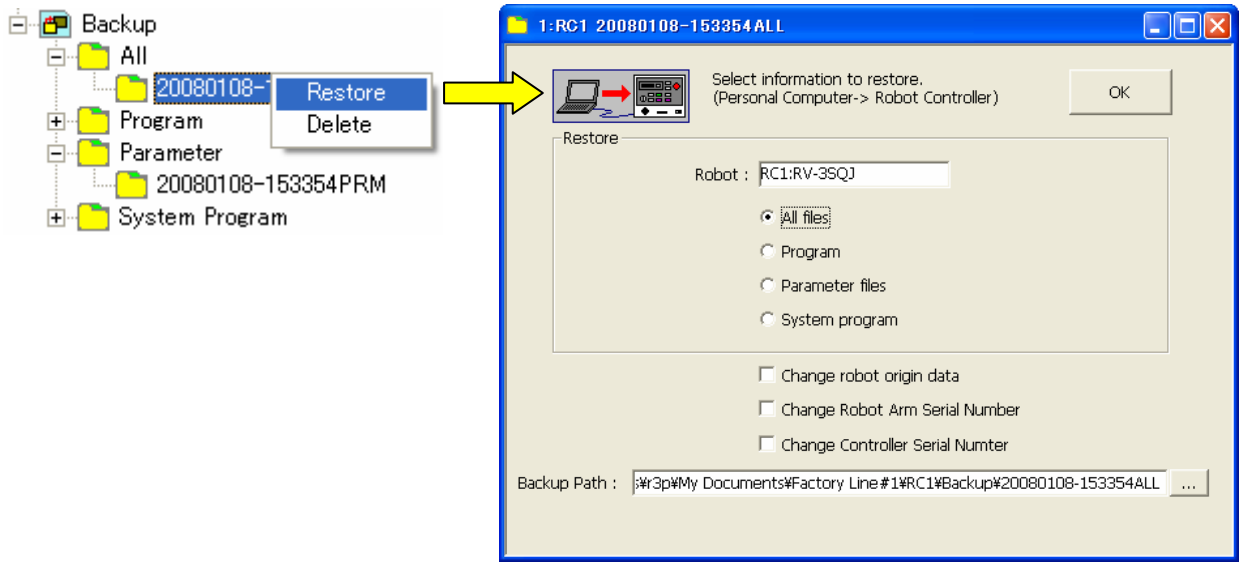
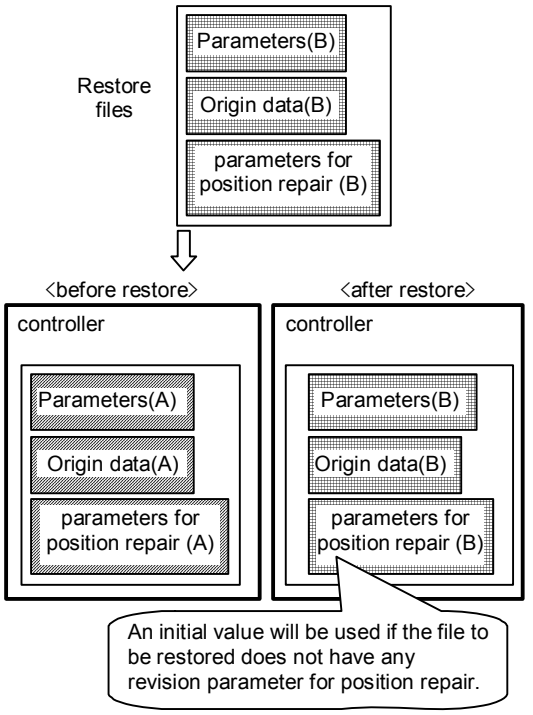
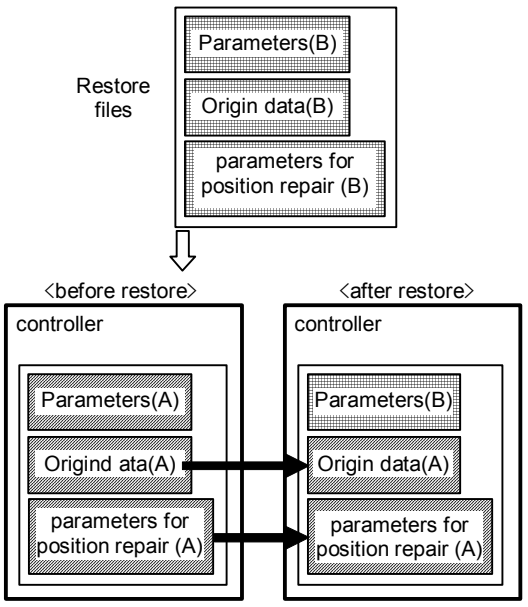


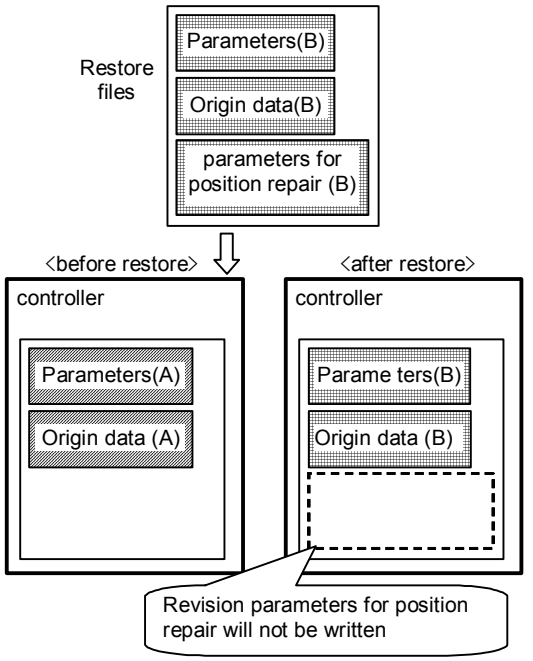
Figure 12-8 Restore

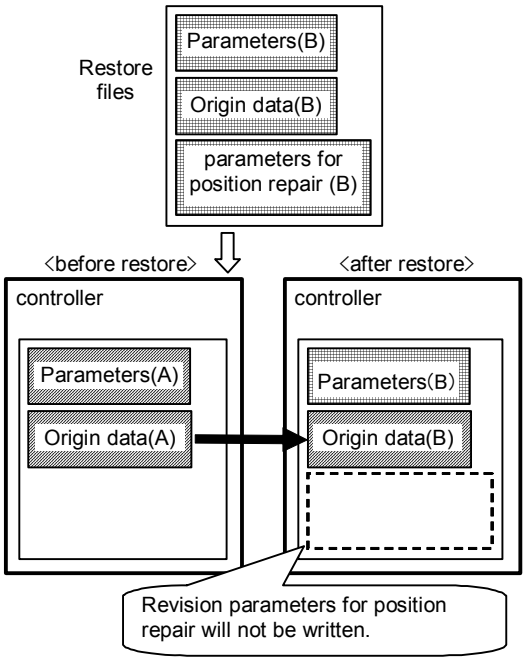
All Files	:	Transfers all files (except BKUP.SYS and MECHA.SYS) in the designated folder to the robot controller after all information in the robot controller is cleared (initialized).
Program	:	Transfers the robot program file in the designated folder to the robot controller.
Parameter Files	:	Transfers the parameter file in the designated folder to the robot controller.
System Program	:	Transfers the system base program file in the designated folder to the robot controller.
Change Robot Origin Data	:	This is backup information and it only valid when All or Parameter files is selected. Operations for when this checkbox is checked and when it is not are as in **"Table 12-1 ".**
Change Robot Arm Serial Number	:	If you check this checkbox, the robot main unit serial number is also rewritten.

Table 12-1 Operations for "Change Robot Arm Serial Number Too"

<table>
<tr><th>Controller</th><th>Check ON
☑ Change robot origin data</th><th>Check OFF
☐ Change robot origin data</th></tr>
<tr><td>CRn-700 series
or
CRn-500 series J2 edition or later</td><td> 

Transfers a backed up file as is. The origin data is replaced.</td><td> 

A backed up file is transferred. However, as for the origin data and the parameters for recovering positions, the information inside the controller is retained.</td></tr>
<tr><td>CRn-500 series H7 edition or earlier</td><td> 

Transfers a backed up file as is. The origin data is replaced.</td><td> 

A backed up file is transferred. However, as for the origin data, the information inside the controller is retained.</td></tr>
</table>

Be careful. If communication is cancelled during a series of restore processing, the position revision parameters generated with the "Position repair" function and the origin data may be changed.

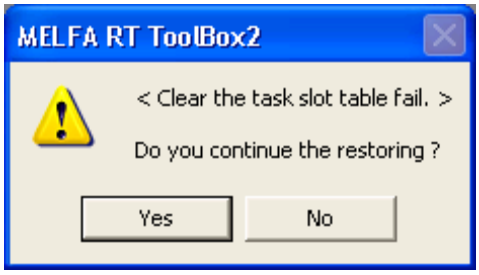
⚠ Caution

Precaution for Restore

If a batch restoration or a program restoration is executed when the program is being started, the program will automatically be stopped.

At this time, if there is an error in the controller, the program in operation cannot be stopped, and the message shown on the right will be displayed.

Although a restoration process can be executed even in such a case, the program currently selected or the program that is started by ALWAYS cannot be re-written. If it is possible to remove the cause of the error, reset the error and execute the restoration process again.



12.3. Deleting Backup Data

You can delete the backed up information.
Select the information to delete, then click the right mouse button. From the right mouse button menu, select [Delete].

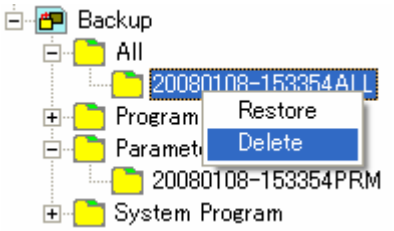


Figure 12-9 Deleting Backup Data

It is also possible to delete all the backup data.
Right click [Backup] for the target project. From the right mouse button, click "Delete All".

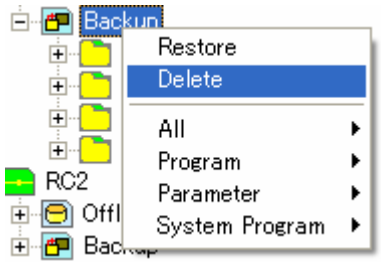


Figure 12-10 Deleting All the Backup Data

Caution

Be aware that once backup data is deleted, you can not restore it.

13.Simulation

This chapter explains the simulation operation methods.

⚠ Caution

Simulation can not be used with the mini edition.

The simulation function only supports the "RT ToolBox2" standard edition. It can not be used with the mini edition.

Simulation can not be used with Movemaster commands.

Be aware that even with the standard edition, the simulation function can not be used when Movemaster commands are selected.

You can not communicate with the robot controller during a simulation.

You can not communicate with the robot controller during a simulation. To communicate with the robot controller, click on the menu bar [Online] → [Offline] to end the simulation, then again click on the menu bar [Online] → [Online] to connect the robot controller.

With an actual robot, an overload error might occur.

Be aware that even if you run the simulation with the work and hand weight set and this works properly in the simulation, when you actually operate with the robot, an overload may occur and make operation impossible.

If there is an input signal wait in the program, use pseudo-input.

When you execute a program in simulation and there is an input signal wait in the program, the program does not move to the next step until that command is executed. Therefore, if there is an input signal wait, use the pseudo-input function from the signal monitor.

13.1. Starting a Simulation

Click on the menu bar [Option] → [Simulator]. At that time, if there are two or more projects in the workspace, the screen for selecting the screen to conduct the simulation is displayed.

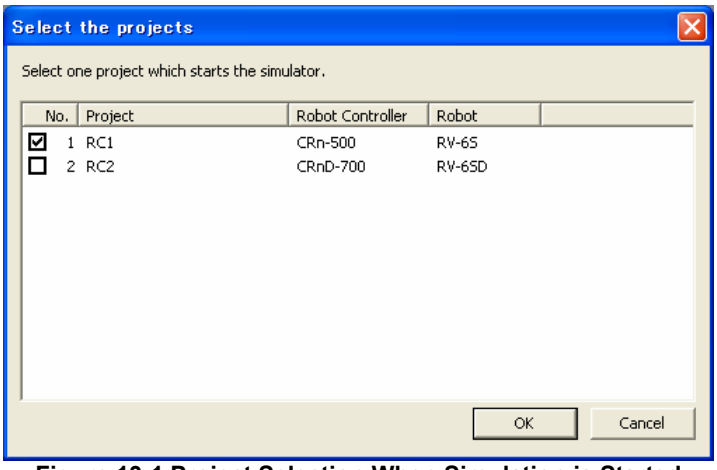


Figure 13-1 Project Selection When Simulation is Started

Select the project to start the simulation of, then click the [OK] button.

Caution

You can only simulate one project.

You cannot simulate multiple projects at the same time.

You can also start a simulation through operations from the tool bar.

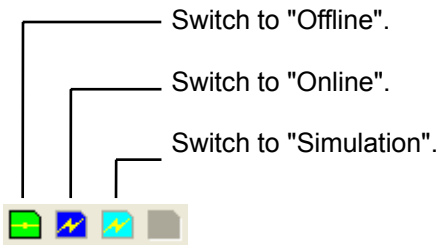


Figure 13-2 Explanation of Toolbar

When the simulation start-up is complete, the operation screen for the simulation is displayed on the screen. Also, the virtual controller for the simulation is automatically started up as an icon. The simulation is run by this virtual controller. The virtual controller ends automatically when the simulation ends. Do not end the virtual controller manually.

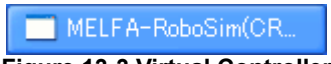


Figure 13-3 Virtual Controller

13.2. Explanation of the Simulation Operation Screen

This is a simple explanation of the simulation operation screen.

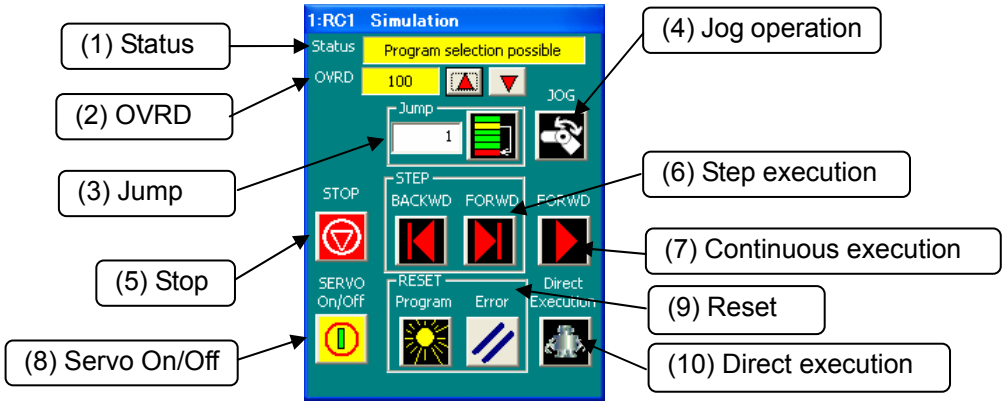


Figure 13-4 Simulation Operation Screen

(1) Status
This shows the task slot status for simulation.

(2) OVRD
This displays and sets the robot speed override.

(3) Jump
You can specify the line in the program to execute.

(4) Jog operation
This conducts jog operations for the simulation robot. When you click this button, the jog operation screen is displayed.

(5) Stop
When a program is running in a simulation, this stops the program.

(6) Step execution
This executes the specified program one line at a time.

(7) Continuous execution
You can execute the stopped program again from the line on which is was during with a stop instruction or a breakpoint during program running.

(8) Servo On/Off
You can switch the simulation robot servo On/Off.

(9) Reset
You can reset the program and any errors that have occurred.

(10) Direct execution
You can execute any command without relationship to the robot program.

13.3. Robot View Start

You can display the robot whose simulation you are running and its movements in 3D to check them.

With the simulation running, from the project tree, either double click [Online] → [<Robot model name>] or select [3D Monitor] with the mouse right button menu. The 3D display screen for the set robot is displayed.

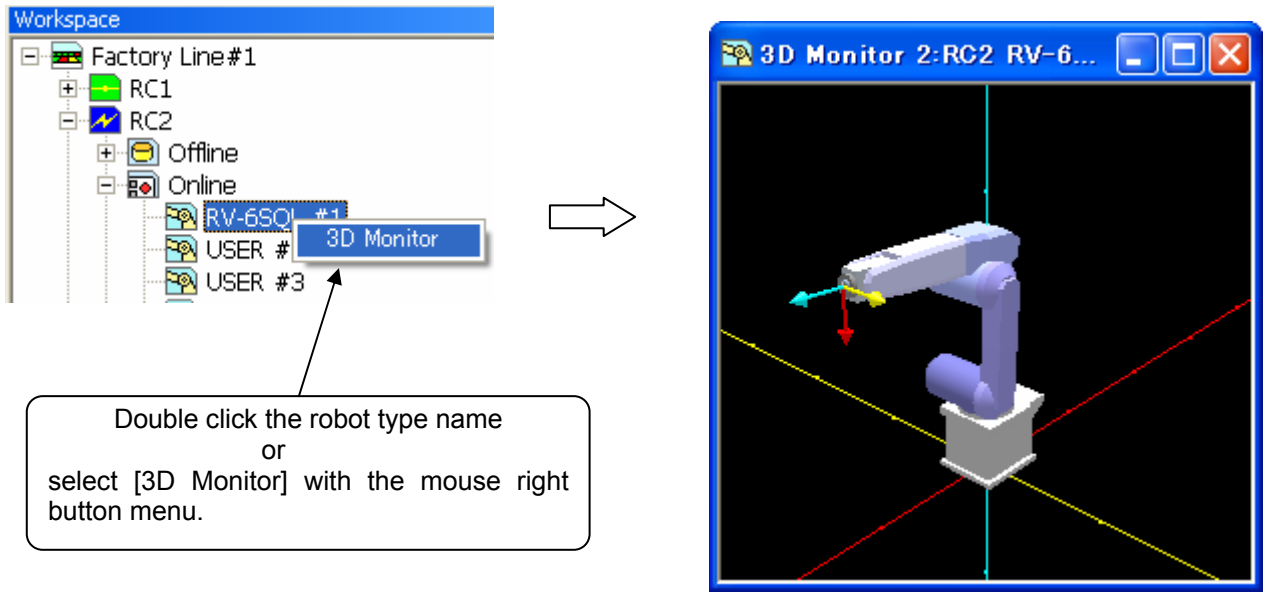


Figure 13-5 Robot View Start

The robot view perspective can be changed with mouse operation.

Table 13-1 Robot View Perspective Change Operations

Viewpoint of changing	Mouse operations on the graphic
Rotation	While clicking the left button, move left/right → Rotation around Z axis Move up/down → Rotation around X axis Move left/right while clicking the left + right buttons → Rotation around Y axis
Move	Move up/down/left/right while clicking the right button
Enlargement/reduction	Move up/down/left/right while clicking pressing [Shift] key and clicking the left button

13.4. Robot Program Selection

Select the robot program to run the simulation. Step operation or direct execution in the simulation are not possible unless a program is selected.

In the simulation, from the project tree, with [Online] → [Program], execute "Open in debugging status" for the program from the right mouse button.

If the robot program being simulated is not [Online], use program management to copy to the virtual controller.

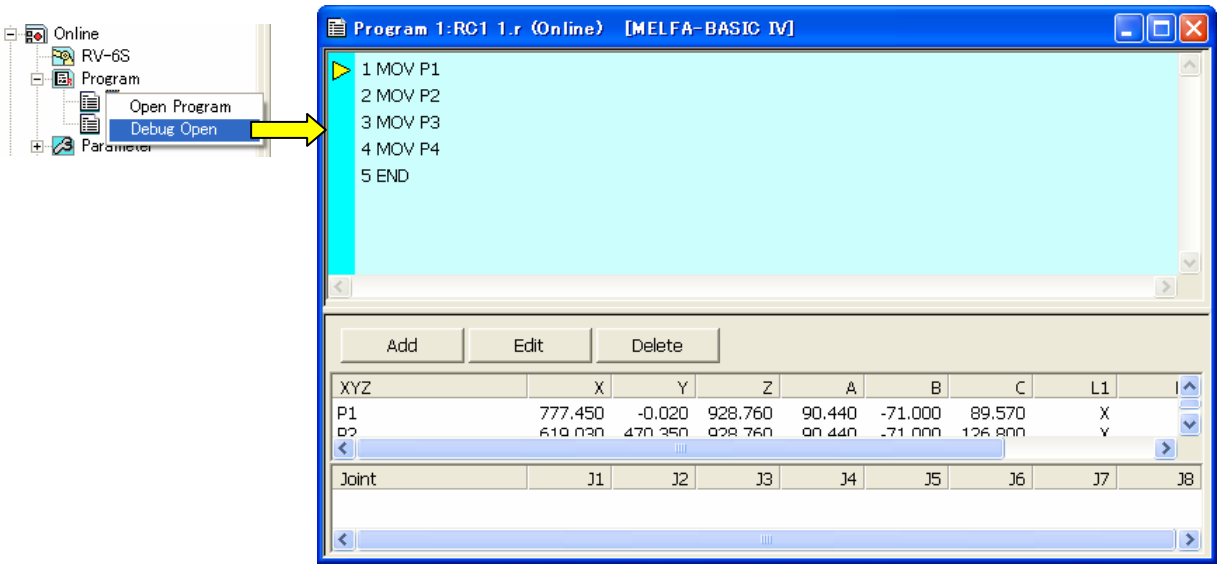


Figure 13-6 Program Selection for Simulation

The specified robot program is opened in debugging status. The execution line cursor "▷" is displayed at the left end of the command statement edit area. The line on which this execution line cursor is displayed is the line currently being executed.

The display of this execution line cursor can be switched on/off with the menu bar [Debug] → [Display/Do not display].

13.5. Program Execution

You can execute a program that has been opened in debugging status.

Start the simulation, then open the robot program in debugging status. When you click the [Continuous Execute] button on the simulation operation screen, automatic operation of the program is started. This automatic operation completes in one cycle. Be careful. If the program repeats infinitely, it does not stop automatically.

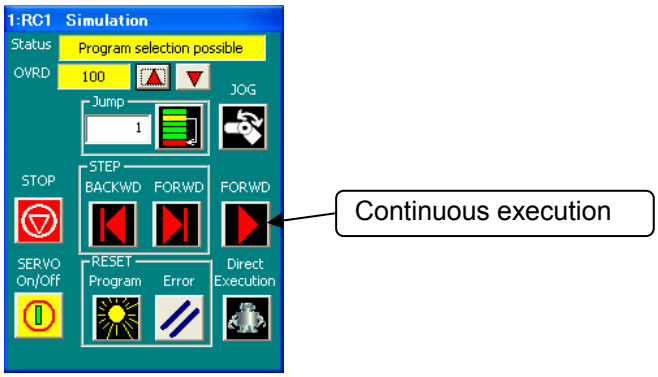


Figure 13-7 Program Execution

To forcibly stop a program that is executing, click the [Stop] button on the simulation operation screen.

13.6. Specifying the Starting Line for Program Execution

You can freely specify the line in the program from which to start execution. Use [Jump] on the simulation operation screen Input the step number to start execution from, then click the button on the right side of the input box. The current execution line moves to the specified step number.

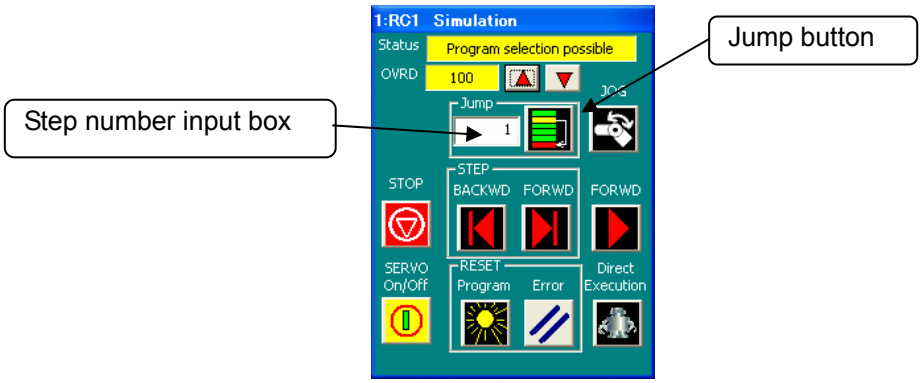


Figure 13-8 Program Execution Line Execution

13.7. Breakpoint Setting

You can also use breakpoints in a simulation. For details on the operation method for breakpoints, see **"7.9.4 Setting and deleting breakpoints"**.

13.8. Step Operation

A program that has been opened in debugging status can be executed step by step.

Start the simulation, then open the robot program in debugging status. Step operation in simulation is performed from the simulation operation screen.

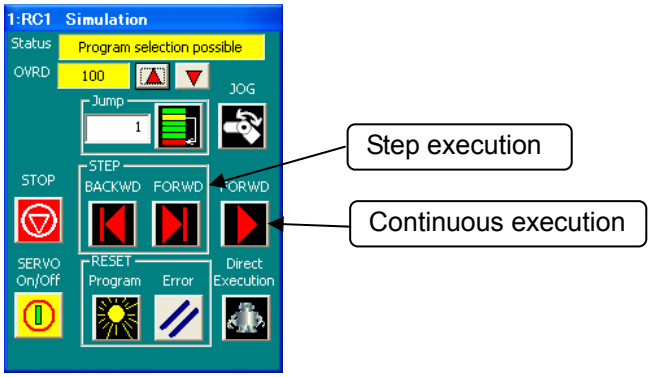


Figure 13-9 Simulation Operation Screen

Operate with the [FORWD] button and [BACKWD] button in the [Step Execution] group on the simulation operation screen.

Pressing the [FORWD] button executes the command on the current execution line and advances the current execution line by line. Pressing the [BACKWD] button executes the command on the current execution line and returns the current execution line by line.

13.9. Direct Execution

You can input command statements and operate the robot directly.
Click the simulation operation screen [Direct execution] button.

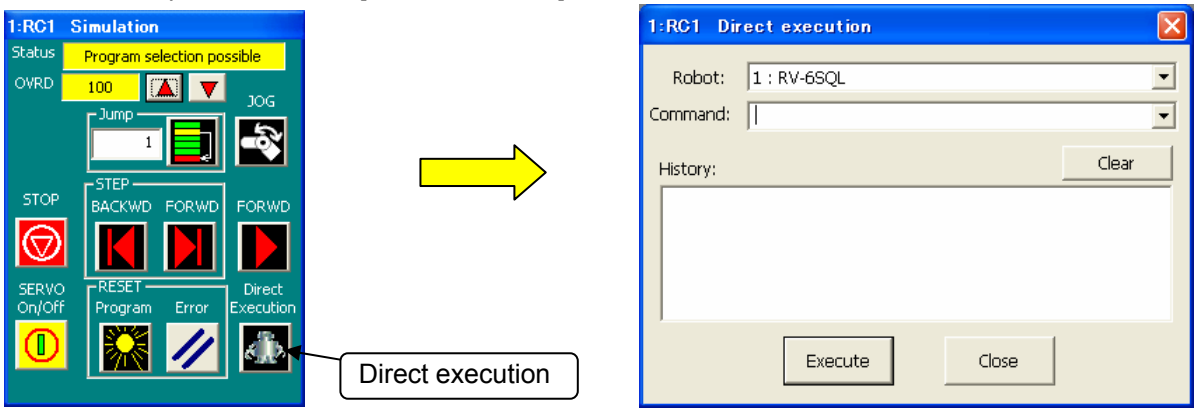


Figure 13-10 Starting Direct Execution

Input the command to execute into the command box, then either press the keyboard [Enter] key or click the [Execute] button. The input command is executed. At this time, if a position variable is specified in a move command or a like, the position variables defined in the currently open program are used. A position variable not defined in the program can not be used.

A command that has been input once into the command box is added to the history and can be selected from the command box dropdown list. However, when the simulation is ended, the history and dropdown list commands are commanded

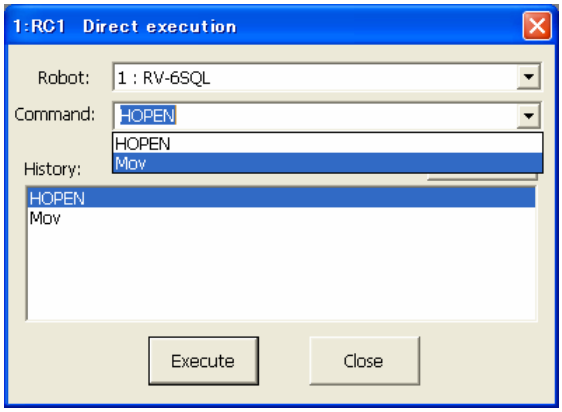


Figure 13-11 Command Dropdown Liston Direct Execution Screen

This operation is not possible during automatic running with the [Continuous Execution] button.

⚠ Caution

When executing directly, select the program to execute.

Direct execution can not be used unless a program is selected. Always open a program in debugging status and put it in selected status.

13.10. Jog Operation

You can perform the jog operations displayed in the robot view in the simulation status.
Click the simulation operation screen [Jog] button. The screen for jog operations is displayed at the bottom of the simulation operation screen.

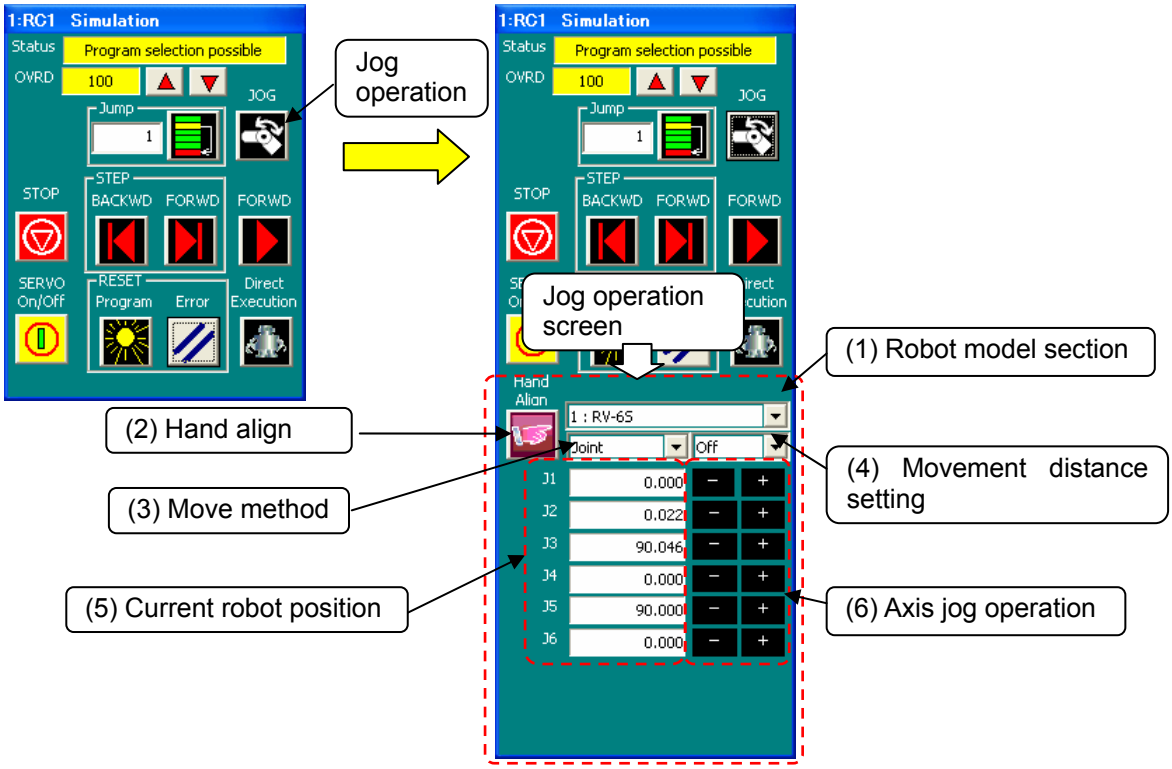


Figure 13-12 Starting Jog Operation

(1) Robot model select

When multiple robots are set as connected, select the robot model to operate.

(2) Hand align

You can align the posture of a hand installed on the robot in units of 90 degrees.

This function moves the value to the multiple of 90° that is closest to the A, B, and C components of the current position.

(3) Move method

Select the robot move method. The robot move methods are "Joint" and "XYZ".

The robot current position display and the jog operation button display for each axis use the method selected here. Also, these displays depend on the axis configuration of the connected robot.

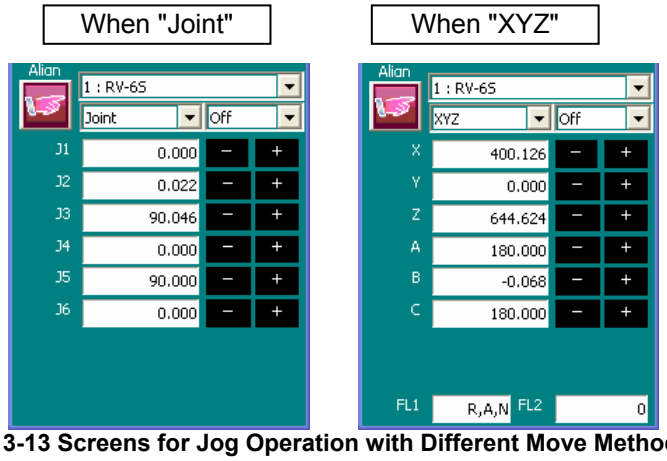


Figure 13-13 Screens for Jog Operation with Different Move Methods Selected

(4) Distance setting for moving

This selects the robot move distance. The robot move distances are "off", "High", and "Low".

For a detailed explanation of the move distance, see **"Detailed Explanation of Functions and Operations"** in the robot controller's user's manual.

(5) Robot current position

This displays the current robot position.

(6) Jog operations on each axis

This conducts jog operations on each robot axis.

Moves the selected robot axis in the "-" direction.

Moves the selected robot axis in the "+" direction.

These buttons move the robot while the mouse button is held down.

13.11. Simulation Robot Position Variable Editing

You can edit position variables by moving the simulation robot and reading the position variables from the simulation robot posture.

Move the robot to the target position with jog operations.

Click the [Add] button or [Change] button for position variables in a program opened in debugging status to display the position edit screen, then click the [Read current position] button. You can read in the current position of the simulation robot.

13.12. Tact Time Calculation

You can use the simulation function to calculate the tact time for a program prepared.

⚠ Caution

Tact time calculation

The calculated tact time varies with the capacity of the computer used and its load status and does not completely match the actual robot operating time (tact time).
Use this function as a rough yardstick for tact time study.

Under correct conditions, the results of tact time calculation with this software have an error of about ±3% compared to the actual robot operating time (tact time).

For details on tact time deviation, see **"13.12.3 Causes of tact time deviation".**

13.12.1. Conditions for tact time measurement

Be aware of the following restrictions on tact time calculation.

⚠ Caution

Do not use a program that has signal input or robot status changes.

For programs that have signal input from the outside or that have changes in robot status variables or the like, either comment out such sections or extract just the section you want to calculate the tact time for and calculate it.
When studying the tact time for such a program, take this into account by adding an approximate input wait time to the calculation results.

Do not use a program with an infinite loop.

The tact time for a program that falls into an infinite loop cannot be correctly calculated. Confirm that the robot program does not fall into an infinite loop with a FOR statement or GOTO statement.

Do not use M_TIMER(1).

Do not use M_TIMER(1). This software's tact time calculation uses M_TIMER(1). If M_TIMER(1) is used during the program, the tact time can not be calculated accurately. Change the program to use any timer from M_TIMER(2) to M_TIMER(8) instead, then calculate the tact time.

If a position array variable or joint array variable is included, also include the corresponding "Dim" declaration in the tact time calculation range.

If a program whose tact time is being measured includes a position array variable or joint array variable also include the corresponding "Dim" declaration in the tact time calculation range.

Include the destinations for any GoTo or GoSub's in the tact time calculation range.

For example, the tact time can not be calculated for a program like the following.
The location displayed inverted in black in the program is set for tact time calculation.)

No destination selected for GoTo statement

```
1 xxx
 :   :
10 Mov P01
11 Mov P02
12 Mov P03
13 GoTo *L10
14 END
```

No destination selected for sub-routine

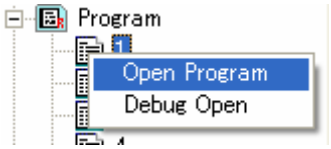
```
1 Mov P00
 :   :
10 GoSub *SUB1
11 End
12  :
20 *SUB1
21 Mov P01
22 Return
```

13.12.2. Tact time measurement

This explains tact time measurement.

1. **Start simulation.**
   Start simulation. For details, see **"13.1 Starting a Simulation"**.

2. **Open the program.**
   Open the program whose tact time you will calculate. You can calculate the tact time for "offline" and "online" programs. However, you can not calculate the tact time for a program that has been opened in debugging status. Always open the program with "Open program".



   For details on the operations for opening a program, see **"7.2 Opening an Existing Program".**

3. **Specify the range for measuring the program tact time.**
   Drag the section to measure the program tact time for, then drag it to select it.

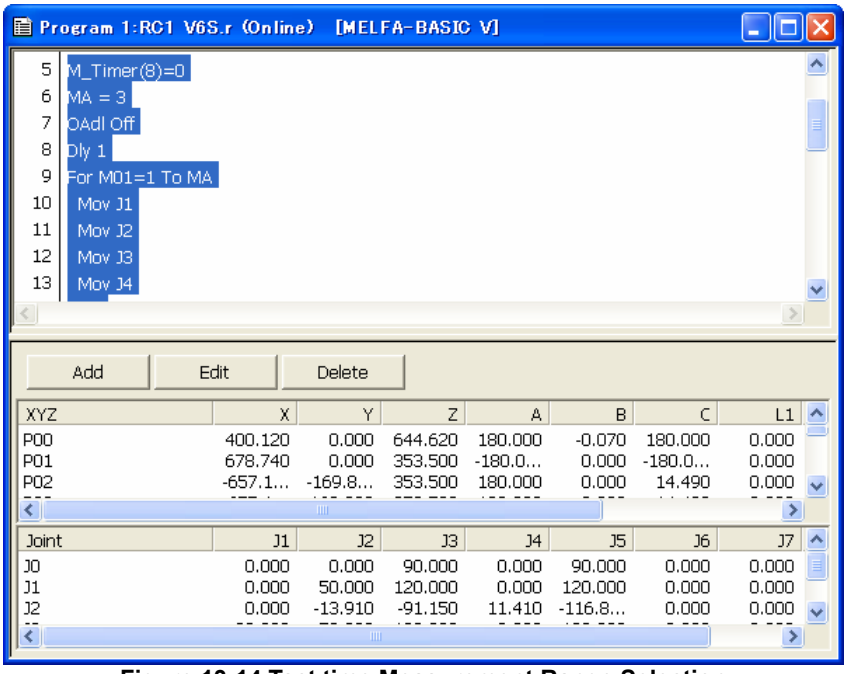


Figure 13-14 Tact time Measurement Range Selection

4. **Click on the menu bar [Tool] -> [Tact time].**

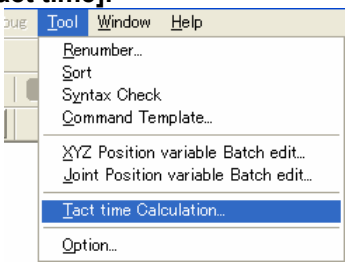


Figure 13-15 Toolbar "Tool" Menu

5. Check the range over which you will calculate the tact time.

Open the program whose tact time you will calculate. Check the contents of this range, then click the [OK] button.

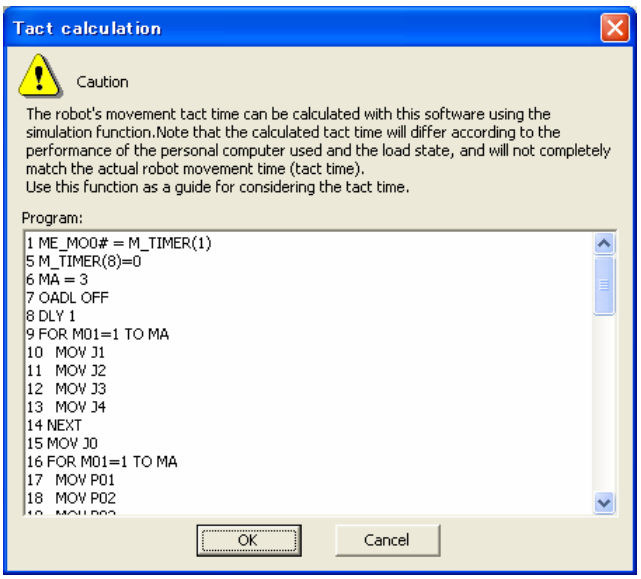


Figure 13-16 Checking the Tact time Measurement Range

The tact time calculation is started. During tact time calculation, "Calculating tact time" is displayed on the simulation operation screen. Do not perform any other operations until this display goes out.

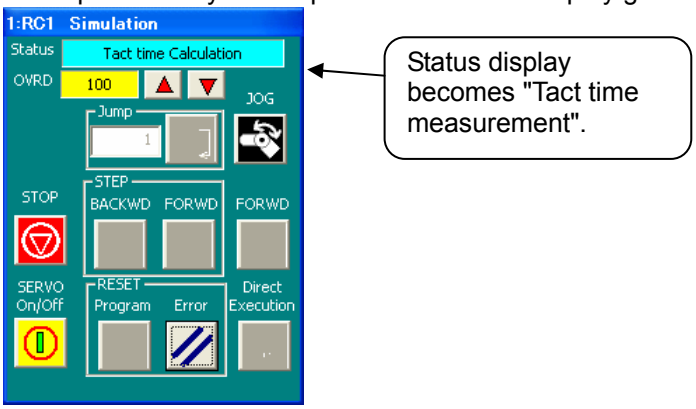


Figure 13-17 Simulation Operation Screen During Tact time Measurement

6. The tact time measurement results are displayed.

When tact time measurement is complete, the "Take time calculation results" screen is displayed. The tact time measurement results are displayed in ms [milli seconds].

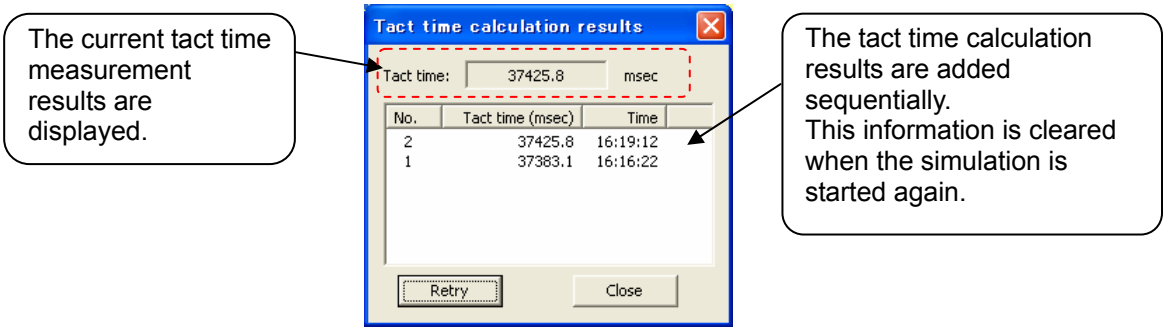


Figure 13-18 Tact time Measurement Results

13.12.3. Causes of tact time deviation

With this software, you can use the simulation function to calculate the robot movement tact time.
However, the calculated tact time varies with the capacity of the computer used and its load status and does not completely match the actual robot operating time (tact time).
Use this function as a rough yardstick for tact time study.
Also, there are the following causes for the tact time calculation results deviating from actual robot tact time.

(1) An application other than "program editing" of this software is running.

This software's tact time calculation has parts that are processed on the computer in the background. For example, if you are working running an application such as a Word document that is saved automatically, it takes more time for the background processing for this software and sometimes the correct tact time is not calculated. The tact time is increased.)
When calculating the tact time with this software, first close other applications.
Also, on this software itself, first close all the screens beside "Program edit".

(2) This uses commands that depend on the robot main unit status and external equipment.

There is no connection with the robot main unit or external equipment in this software's tact time calculation. Therefore, commands that are executed communicating with that equipment are executed as if ideal information were sent from that equipment. Therefore, the calculated cycle time is shorter than the actual cycle time.
The commands to which this applies are shown in the following table.

	Function	Command	Explanation
1	Positioning completion wait	FINE	Monitors the robot's status until it reaches the target position.
2	Compliance	CMP JNT/POS/TOOL/OFF CMPG	Monitors external force on the robot.
3	Collision detection	COLCHK、COLLVL	Monitors external force on the robot.
4	Servo control	SERVO ON/OFF	Monitors the servo amp status

For example, if the servo On command is executed in the program, the actual robot takes a few seconds, but the simulation requires almost no time at all (500 ms or less). In order calculate the tact time precisely, program using only movement commands and none of the above commands. If you are executing a program you already have, either comment out such commands or extract and use just the part you want to measure.

(3) This uses functions that depend on the CPU processing speed and OS.

The tact time calculation in this software runs on Windows, but the robot controller control software runs on a real-time OS, so the internal operations are different. For example, with a real-time OS, the tact time may be increased by the postponement of calculations with a high load that could not be fully processed, but in this software's tact time calculations, such an increase in the tact time can not be calculated. Therefore, if you use the high-load functions below, the calculated tact time may be a few percent less than the actual tact time.
The commands to which this applies are shown in the following table.

	Function	Cause of increased processing on actual robot
1	CC-Link	Because there is more signal processing with the CC-Link option than without it
2	Multi tasking	Because multiple robot programs are executed at the same time
3	Added axes	Because of the need to control the additional axes
4	Maximum acceleration/deceleration control	Because the optimum movement for the robot load is calculated
5	Collision detection	Because processing to detect collisions is executed (*1)
6	Maintenance forecast	Because the processing time is longer when the maintenance forecasting information collection level is raised (*1) In tact time calculation, the same results are calculated as for information collection level 1 (the factory default setting).

(*1) On actual robots, the tact time is roughly 3-10% longer.

By changing a robot program to eliminate the above factors, you can reduce the difference between the simulation tact time and that on the actual robot. However, because the simulation executes the next movement command without waiting for

static determinacy after movement or for the conditions to be established, the tact time calculated differs from the tact time on the actual robot.
We recommend that you finally confirm operation on the actual robot.

13.13. Ending Simulation

To end the simulation, close the robot program in debugging status. Then click on the menu bar [Online] → [Offline] or click on the tool bar "Offline".

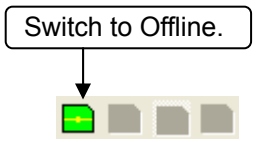


Figure 13-19 Toolbar "Switch to Offline"

When the simulation ends, the virtual robot controller, robot view, and simulation operation screen end and the controller goes back offline.

14. MelfaRXM.ocx Communications Middleware Setup

14.1. Summary

MelfaRXM.ocx is an ActiveX controller that communicates with CRn-500 series robot controllers. MELFARXM.ocx can only be used if you have purchased the standard edition of RT ToolBox2. (Customers who have purchased the mini edition can not install MELFARXM.ocx.)

When using only the RT ToolBox2 functions, there is no need to set up "MELFARXM.ocx".

Using MELFARXM.ocx enables you to simply produce Windows applications connected to a CRn-500 series robot controller on the customer's device.

> The user's manual for using MelfaRXM.ocx and the cautions are on the RT ToolBox2 standard edition CD-ROM.
> This MelfaRXM.ocx is for CRn-500 series robot controllers. It can connect with CRn-700 series robot controllers, but the only modes available for communications with robot controllers are RS-232 and Ethernet.

14.2. CD-ROM Contents

The contents of the "RT ToolBox2" (standard edition) CD-ROM are as follows.

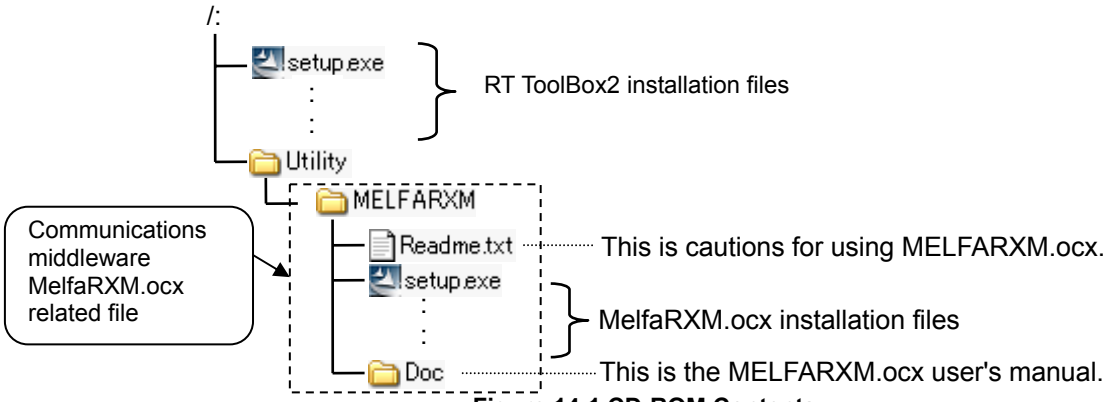


Figure 14-1 CD-ROM Contents

For details on MelfaRXM.ocx usage methods, refer to the user's manual on the CD-ROM.

14.3. User's Manual Reading Guide

The manual is in the CR-ROM as the Adobe PDF file.

D:/Utility/MELFARXM/Doc/MelfaRXME.pdf

* Example for the CD-ROM drive is "D:".

(1) Preparation for viewing

1) Preparing computer

Prepare a computer that has a CD-ROM drive.

2) Preparation of viewing software

Viewing requires Acrobat Reader Ver 5.0 or higher.

If neither Acrobat Reader (nor Adobe Reader) is installed, please download it from the Adobe Systems web site. (As of December 2007)

URL: http://www.adobe.com

(2) Viewing methods

1) Starting From Windows Explorer

When you start Windows Explorer, then select the file, Acrobat Reader (or Adobe Reader) starts and the user's manual is displayed.

2) Starting directly from Acrobat Reader (or Adobe Reader)

When you start Windows Explorer Acrobat Reader (or Adobe Reader), then select the file, the user's manual is displayed.

14.4. Installation

Perform installation according to the following procedure

(1) Insert the program CD-ROM into the CD-ROM drive of your personal computer. Setup of **“RT ToolBox2”** automatically starts. Please click “cancel”.

(2) Select [Run] from the [Start] button.

Figure 14-2 Selecting [Run]

(3) Check the drive name of the CD-ROM drive. Enter the following and click the [OK] button.
"Drive name":/Utility/MelfaRXM/Setup.exe
(If the CD-ROM drive is "D", enter "D:/Utility/MelfaRXM/Setup.exe")

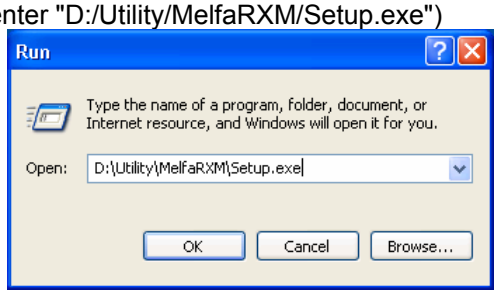


Figure 14-3 [Run] Screen

(4) Installer starts and the Setup screen appears. Install according to the instructions that appear on the screen. The Product ID is needed when this "MelfaRXM.ocx" is installed. Input the Product ID same as the Product ID of "RT ToolBox2". "MelfaRXM.ocx" can be installed with the Product ID of standard version, but can not be installed with mini version's.

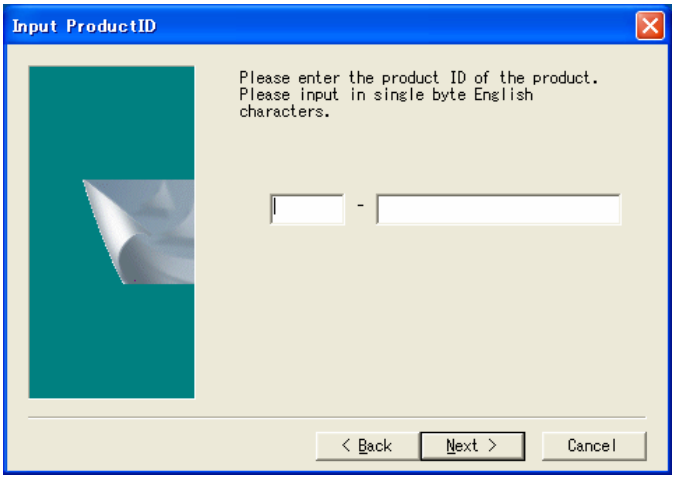


Figure 14-4 Input Product ID

The product is installed as in **"Table 14-1 Files to be Installed".**

Table 14-1 Files to be Installed

No.	Description	Install destination
1	MelfaRXM.ocx	/Windows/System folder ex) Windows 2000 :/Winnt/System32 Windows XP :/Windows/System32 MelfaRXM.ocx and EzSocketRC.dll are registered in the registry.
2	EZSocketRC.dll (communication DLL)	
3	NarcServerApiM.dll (communication DLL)	
4	RoboCom.exe (communication DLL)	A folder specified during install operation (Normally, C:/ is used.) └ [MelfaRXM_Dev] ├ ReadMe.tx .....text file which indicated notes ├ [RoboCom].....folder of communication server ├ [Doc].....folder of Instruction Manual ├ [Sample] .....folder of sample programs ├[BCB] .....Borland C++ Builder (5.0) ├[VB] .....Visual Basic (6.0) └[VC++] .....Visual C++ (6.0) ├ [Redist] ├[Installer] .....folder of system files Installer (for redistribution) └[SysFiles] .....folder of system files (for redistribution)
5	Instruction Manual	
6	Sample programs	
7	Redistribution files	

15. Appendix

15.1. Q&A

This explains frequently asked questions in a Q&A format.

1. Version

(1) Where is the software version information?

It is printed on the surface of your CD-ROM for this software.
You can also check the version information on the screen displayed with this software's menu bar [Help] → [About RT Tool Box2].

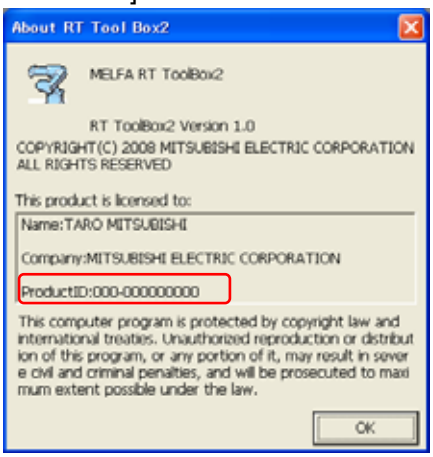
(2) Where is the robot controller software version information displayed.

When this software is connected, you can check with the project tree, [Online] properties.
The version information is also displayed on the title screen for the optional teaching box.

2. Product ID

(1) Where can I check the product ID?

This software requires the product ID for installation.
The package containing your software contains a sheet of paper on which is written the product ID.
Also, after this software has been installed, you can also check the product ID on the screen displayed with the menu bar [Help] → [About RT Tool Box2].



(2) I checked "About RT ToolBox2" from the menu, but the product ID is not displayed.

Is "No Product ID!" displayed?
If "No Product ID!" is displayed in the version information, the installation of this software may have failed. Unfortunately, you need to install the software again.

3. Communications (general)

(1) What are the means for communication with the robot controller?

With a CRn-700 series robot controller, you can communicate via USB, Ethernet, and RS-232.
With a CRn-500 series robot controller, you can communicate via Ethernet and RS-232. However, when using Ethernet with a CRn-500 series robot controller, the robot controller must have the optional Ethernet card.
For details, see **"5.1 Robots Connected and Types of Communication"**.

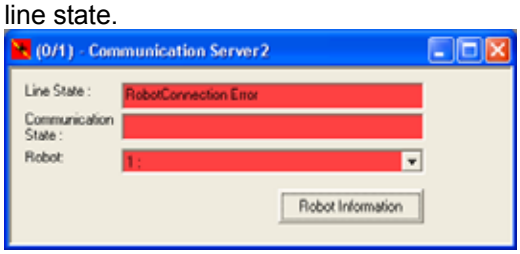
(2) Where are the communications settings made?

They are made on the project edit screen. For details, see **"5.2 Connection Settings"**.
For the robot controller side, change the communications parameters with the optional teaching box. For details, see **"Detailed Explanation of Functions and Operations"** in the robot controller's user's manual.
[Caution] The communication settings must be made on both this software and the robot controller.
If you change the robot controller settings, you must change the settings in this software too.

(3) I can not communicate with the robot controller.

If you can not communicate with the robot controller, check the following.
Also see **"4. Communications (RS-232)"**, **"5. Communications (Ethernet)"**, and **"6. Communications (USB)"**.

Check item or cause	Solution
Is the connected robot controller selected correctly?	Check if the correct project is selected with this software. For details, see **"4.15 Offline/Online/Simulation"**.
Is the robot controller power supply On?	Switch On the robot controller's power supply and check that the robot controller starts up normally.
Is Communications Server 2 running? Communications Server 2 is started automatically when this software is started. Communication Server 2 is started as an icon.	If you closed Communications Server 2 by mistake, close this software, then restart it.
Are you connected to a robot controller that is neither the CRn-500 nor CRn-700 series?	Check the robot controller model name in its user's manual (or standard specifications or the like).
Are the robot controller's network settings correct?	The robot controller's network settings are made with parameters. Check the robot controller's network settings with the optional teaching box.
Are the communications settings on this software correct?	Set the correct communications settings. For details, see **"5.2 Connection Settings"**.
Is some other MELFA product running? For example is E/EN series computer support software or P/P or P/P-2 running?	Close any MELFA product other than this software. For details, see **"1.6 When Starting at the Same Time as Another Product"**.

Check item or cause	Solution	
What color is Communications Server 2? Return Communications Server 2 to icon status and check the color displays for the line state. 	Red	The problem may be that the robot controller is not connected correctly. Check the items in **"(4) When Communications Server 2 is red (overall)"**.
	Green	The problem may be that the robot controller and the computer have different communications settings. Check the items in **"(5) When Communications Server 2 is green (overall)"**.
	Yellow	Check the items in **"(6) When Communications Server 2 is yellow (overall)"**.
	Light blue	Check the items in **"(7) When Communications Server 2 is light blue (overall)"**.

(4) When Communications Server 2 is red (overall)

When Communications Server 2 is displayed red, check the following.

Check item or cause	Solution
Is the communications cable connected correctly?	Connect the cable correctly.
Are you using the correct communication cable?	Check the communication cable specifications. Be careful. RS-232 cables and Ethernet cables can be either cross cables or straight cables. For the cable specifications, check your robot's "Standard Specifications".
When connected with CRnQ communications, is the target CPU set correctly?	Set the target CPU correctly. For details, see **"5.2.4 CRnQ Communications Settings"**.

(5) When Communications Server 2 is green (overall)

When Communications Server 2 is displayed green, check the following.

Check item or cause	Solution
Do the communications settings in this software and on the robot controller match?	The communications settings in this software and on the robot controller must match. The communications settings for this software are made on the project edit screen. For details, see **"5.2 Connection Settings"**. For the robot controller side, change the communications parameters with the optional teaching box. For details, see **"Detailed Explanation of Functions and Operations"** in the robot controller's user's manual.

(6) When Communications Server 2 is yellow (overall)

When Communications Server 2 is displayed yellow, check the following.

Check item or cause	Solution
Is this software in the "Online" status?	This software started up in the "Offline" status. When you put it into "Online" status, it communicates with the robot controller. For details, see **"4.15 Offline/Online/Simulation"**.

(7) When Communications Server 2 is light blue (overall)

If Communications Server 2 is displayed light blue, but communications are still not possible, check the following.

Check item or cause	Solution
Is there a communications error in this software?	Check the contents of the communications error and close the communications error window.
Is a simulation underway? (Note) Only the standard edition has the simulation function.	End the simulation.
Is the screen server running on the computer?	End the computer's screen server.
Is the computer's hard disk is power save mode?	End computer hard disk power save mode setting.

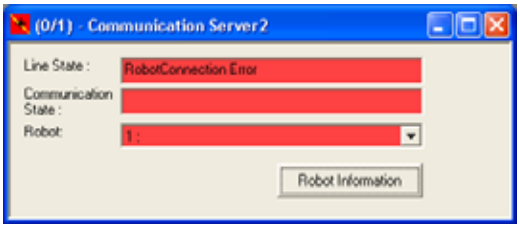
(8) Communications with the robot controller are cut off mid-way through.

If you can not communicate with the robot controller, check the following.

Check item or cause	Solution
Is the screen saver set, hard disk power off set, system standby set, or system shut-down set?	If any of these functions are running, they may cause a communications time-out, so do not use any of these functions.
Is some other product running that uses communications? Or is any permanent resident software running that uses communications?	A computer communications port can not be opened for two applications, so either close the other application or change the port that this software uses.

4. Communications (RS-232)

If you can not communicate with the robot controller using RS-232, check the following.
Also, see **"3. Communications (general)"**.

Check item or cause	Solution	
What color is Communications Server 2? Return Communications Server 2 to icon status and check the color displayed for the line state. 	Red	The problem may be that the robot controller is not connected correctly.Check the items in **"(1) When Communications Server 2 is red (RS-232)"**.
	Green	The problem may be that the robot controller and the computer have different communications settings.Check the items in **"(2) When Communications Server 2 is green (RS-232)"**.
	Yellow	See **"3. Communications (general)"**.
	Light blue	Check the items in **"(3) When Communications Server 2 is light blue (RS-232)"**.

(1) When Communications Server 2 is red (RS-232)

If Communications Server 2 is displayed red with communications with the robot controller set to RS-232, check the following.

Check item or cause	Solution
Is Communications Server 2 set for the correct communications port (COM1-COM10)?	The default value for this software is COM1. On some computers, COM1 is not allocated to RS-232, but to an infrared port, modem, or the like. Also, when using RS-232 over USB, the port used for RS-232 may be other than COM1. Use the Windows device manager or the like to check the COM number allocated to RS-232 and change the "Port" setting in the communications settings to that port.
Is some other product running that uses the communication port? Or is any permanent resident software running that uses the communications port?	A computer communications port can not be opened for two applications, so either close the other application or change the port that this software uses.
In the communications settings for this software, is the communications method set to TCP/IP or USB?	Change the communications method to RS-232.

(2) When Communications Server 2 is green (RS-232)

If Communications Server 2 is displayed green with communications with the robot controller set to RS-232, check the following.

Check item or cause	Solution
Do the communications settings in this software and on the robot controller match? In particular, check that the protocol settings match.	The communications settings in this software and on the robot controller must match. The default protocol for this software is "Procedural", but for CRn-500 series robot controllers, the default protocol is "Non-Procedural"

(3) When Communications Server 2 is light blue (RS-232)

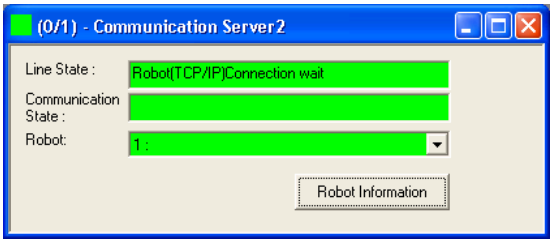
If Communications Server 2 is displayed light blue with communications with the robot controller set to RS-232, check the following.

Check item or cause	Solution
Is anti-virus software running virus checks on RS-232 communications?	Switch off virus checking for RS-232.
When a computer starts up, sometimes this generates noise. This noise can cause a communications error in the robot controller.	Either start the computer before connecting the cable or start the computer before starting the robot controller.

5. Communications (Ethernet)

[Note] When using Ethernet with a CRn-500 series robot controller, the robot controller must have the optional Ethernet card.

If you can not communicate with the robot controller using Ethernet, check the following.
Also, see **"3. Communications (general)"**.

Check item or cause	Solution	
What color is Communications Server 2? Return Communications Server 2 to icon status and check the color displayed for the line state. 	Red	The problem may be that the robot controller is not connected correctly. See the items in **"(1) When Communications Server 2 is red (Ethernet)"**.
	Green	The problem may be that the robot controller and the computer have different communications settings. See the items in **"(2) When Communications Server 2 is green (Ethernet)"**.
	Yellow	See **"3. Communications (general)"**.
	Light blue	See **"3. Communications (general)"**.

(1) When Communications Server 2 is red (Ethernet)

If Communications Server 2 is displayed red with communications with the robot controller set to Ethernet, check the following.

Check item or cause	Solution
In the communications settings for this software, is the communications method set to USB or RS-232?	Change the communications method to TCP/IP.

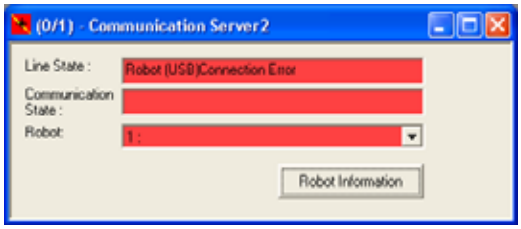
(2) When Communications Server 2 is green (Ethernet)

If Communications Server 2 is displayed green with communications with the robot controller set to Ethernet, check the following.

Check item or cause	Solution
Are the robot controller's network settings correct?	Make the computer's network settings. Check that the IP address, gateway, subnet mask, and other network settings are correct. Check on the computer [Control Panel] – [Network Settings]. * For details on the network settings, please consult with your network administrator.
Is the robot controller's IP address set correctly in the communications settings?	Correctly set the IP address of the robot controller connected to.

6. Communications (USB)

If you can not communicate with the robot controller using USB, check the following.
Also, see "**3.** Communications (general)".

Check item or cause	Solution	
What color is Communications Server 2? Return Communications Server 2 to icon status and check the color displayed for the line state. 	Red	The problem may be that the robot controller is not connected correctly. Check the items in **"(1) When Communications Server 2 is red (RS-232)"**.
	Green	The problem may be that the robot controller and the computer have different communications settings. See **"3. Communications (general)"**.
	Yellow	See **"3. Communications (general)"**.
	Light blue	Check the items in **"(2) When Communications Server 2 is light blue (USB)"**.

(1) When Communications Server 2 is red (USB)

If Communications Server 2 is displayed red with communications with the robot controller set to USB, check the following.

Check item or cause	Solution
In the communications settings for this software, is the communications method set to TCP/IP or RS-232	Change the communications method to USB.
Is the USB driver installed?	Communicating with USB requires that the USB driver be installed. For details, see **"1.5.3 USB driver (CRnD-700 series robot controller) installation"**, **"1.5.4 CRnQ communications USB driver installation"** and **"1.5.5 CRnQ Communications USB driver for GOT transparent installation".**

(2) When Communications Server 2 is light blue (USB)

With communications with the robot set to USB, if Communications Server 2 is displayed light blue, but communications are still not possible, check the following.

Check item or cause	Solution
Is the robot controller power supply Off?	When connected on USB with CRnQ communications, if the robot controller power goes Off after a normal connection was established, the display remains light blue. Switch Offline with this software, switch the robot controller power On, then go back online.
Is the communications cable connected correctly?	When connected on USB with CRnQ communications, if the communications cable is disconnected after a normal connection was established, the display remains light blue. Switch Offline with this software, connect the communications cable, then go back online.

7. Robot program

(1) Did you write the program with Movemaster commands?

The robots that can use Movemaster commands are restricted. Check in your robot's standard specifications to see whether it supports Movemaster commands. If your robot supports Movemaster commands, change the language used with **"6 Robot Program Language Setting"** in this document

(2) Is it possible to use programs as is that we used with an E/EN series robot controller?

Position data prepared with MELFA-BASIC III (for E/EN series) can not be used as is. Convert the position data with the "Program conversion" function. For details, see **"7.11 Program Conversion"**.
Also, some commands have changed. For details, see **"Detailed Explanation of Functions and Operations"** in the robot controller's user's manual.

(3) Is it possible to use R-250R series and R-300R series programs?

R-250R series and R-300R series robot programs written in the MELFA II language can not be used as is. Write new programs.

8. Program edit

(1) When we check syntax, "Error in input command statement syntax" occurs frequently.

Check item or cause	Solution
Is the command statement syntax correct?	Program in correct syntax.
Are double-byte spaces used?	Use only single-byte spaces.
Is the language to use set correctly? This software supports MELFA-BASIC IV, MELFA-BASIC V, and Movemaster commands, but there are terms that must be switched and set for whichever one of these you use.	Set the robot program language you are using. For details, see **"6 Robot Program Language Setting"**.
Does the version of this software support your robot controller? For some robot functions, new commands are added.	Please purchase a version that supports your robot controller. (Please contact the store you purchased from or one of our branches.) * A program can be written to the robot controller even if the syntax check finds many "syntax errors".

(2) How should we change the robot program language setting?

They are made on the project edit screen. For details, see **"6 Robot Program Language Setting"**.
The robot program language set here is enabled when you edit a program offline. Online programs are displayed in the robot program language set with the connected robot controller.

(3) When you open a program, the robot program language is different from the one set for this project.

Is there any online program open?
Online programs are displayed in the robot program language set with the connected robot controller. The robot program language set with the project is enabled when you edit a program offline.

(4) Is it possible to edit or copy a program that is running?

You can neither edit nor copy a program that is running. Stop the program, then edit or copy it.

(5) How should we edit a program for which the start condition is "Always"?

A program for which the start condition is "Always" is executed immediately after the robot controller power comes On. To edit such a program, use the following procedure.

(1) Change the starting condition in the "Slot table (SLT * * 1-32)" parameter to "Start (normal)". (Write this parameter to the robot controller.)
(2) Reset the power supply for the robot controller.
(3) Edit the target program and save it to the robot controller.
(4) Return the starting condition in the "Slot table (SLT * * 1-32)" parameter to "Always". (Write this parameter to the robot controller.)
(5) Reset the power supply for the robot controller.

(6) Is it possible to change the font for a robot program displayed with the program edit tool?

Yes.
Change the font used with **"7.5.4 Changing the font"** in this document.

(7) We are not using joint position variables (J variables), so is it possible to make the display area smaller?

Yes.
Change the display proportions with **"7.5.1 Changing the display area"** in this document.

(8) Is it possible to edit a backed up program data with program editor?

In version 1.2 or later, it is possible to open a backed up program data with program editor. Please refer to **"7.2.4 Opening a program in the backup data"** in this manual for details.
When you use the software Ver.1.1 or earlier, please restore the backed up program to robot controller, then open it with program editor or copy it to the computer with program management, and edit it.

(9) "Use defined external variable can not be used (481000000)" is displayed and the program can not be edited.

This error is displayed if a user defined external variable is used even though the user base program is not defined. (Normally, a user base program is defined with the "PRGUSR" parameter, but if nothing is set in the "PRGUSR" parameter, this error occurs.)
To use a user defined external variable, define the user base program.
Also, for details on user base programs, see **"Detailed Explanation of Functions and Operations"** in the robot controller's user's manual.

(10) Is it possible to copy position data to another program?

Yes.
For details, see **"7.6.4.4 Copy position data"** and **"7.6.4.5 Pasting position data"** in this document.

(11) Does it cause an error to not discriminate uppercase and lowercase letters when inputting commands?

No.
You can input either uppercase letters or lowercase letters with the program editor, but when the program is saved to a robot controller, the commands are converted correctly.

9. Variable monitor

(1) How are external variables (system status variables, program external variables, and user defined external variables) monitored?

Use the program monitor.
For details, see **"9.1.2 Program monitoring"** in this document.

10. Option card

(1) How should one check what option cards are mounted in a robot controller?

Check from the project tree.
For details, see **"11 Option Card"** in this document.

11. Parameter editing

(1) No parameter list is displayed in the parameter editing tool.

Download the parameter list from the robot controller.
For details, see **"8.1.4 Parameter list reading"**.

(2) We changed a parameter, but the new value does not take effect.

After you changed the parameter, did you switch the power for the robot controller Off, then On again?
The new parameter value does not take effect until you switch the robot controller power Off, then On again.
Switch the robot controller power Off, then On again.

12. Backup/restore

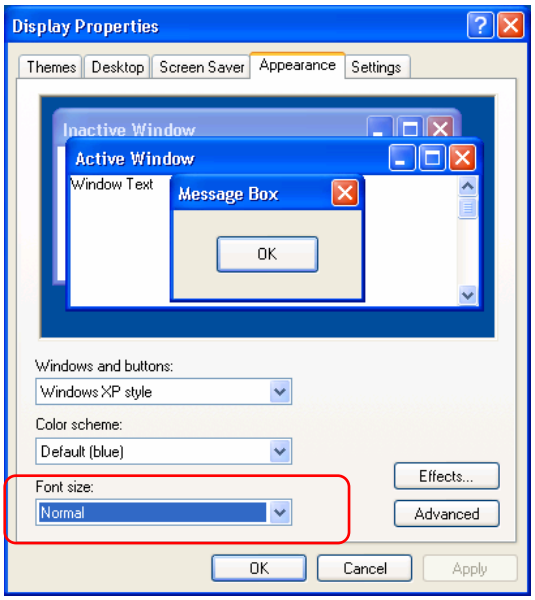
(1) Is it possible to edit a backed up program data with program editor?

In version 1.2 or later, it is possible to open a backed up program data with program editor. Please refer to **"7.2.4 Opening a program in the backup data"** in this manual for details.
When you use the software Ver.1.1 or earlier, please restore the backed up program to robot controller, then open it with program editor or copy it to the computer with program management, and edit it.

13. Other

(1) Characters are displayed on the screen on top of each other or with some characters missing.

Is the font size in the computer screen settings something other than "Standard"?
Use this software with the font size set to "Standard". For Windows XP, to make the screen settings, click [Control Panel] → [Display]. Now from the "Display Properties" window "Appearance" tab, set the font size with [Font Size].



15.2. Index

P

R

S

T

U

W

X

Z

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE : MITSUBISHI DENKI BLDG MARUNOUCHI TOKYO 100-8310 TELEX : J24532 CABLE MELCO TOKYO
NAGOYA WORKS : 1-14, YADA-MINAMI 5, HIGASHI-KU, NAGOYA, JAPAN