EPSON

EPSON RC+ 5.0 | Ver.5.4

User's Guide

Project Management and Development

Rev.4   EM12XS2357F

EPSON RC+ 5.0 (Ver.5.4)

User's Guide

Rev.4

FOREWORD

Thank you for purchasing our robot products.

This manual contains the information necessary for the correct use of the Manipulator.

Please carefully read this manual and other related manuals before installing the robot system.

Keep this manual handy for easy access at all times.

WARRANTY

The robot and its optional parts are shipped to our customers only after being subjected to the strictest quality controls, tests, and inspections to certify its compliance with our high performance standards.

Product malfunctions resulting from normal handling or operation will be repaired free of charge during the normal warranty period. (Please ask your Regional Sales Office for warranty period information.)

However, customers will be charged for repairs in the following cases (even if they occur during the warranty period):

1. Damage or malfunction caused by improper use which is not described in the manual, or careless use.
2. Malfunctions caused by customers' unauthorized disassembly.
3. Damage due to improper adjustments or unauthorized repair attempts.
4. Damage caused by natural disasters such as earthquake, flood, etc.

Warnings, Cautions, Usage:

1. If the robot or associated equipment is used outside of the usage conditions and product specifications described in the manuals, this warranty is void.
2. If you do not follow the WARNINGS and CAUTIONS in this manual, we cannot be responsible for any malfunction or accident, even if the result is injury or death.
3. We cannot foresee all possible dangers and consequences. Therefore, this manual cannot warn the user of all possible hazards.

TRADEMARKS

Microsoft, Windows, and Windows logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Other brand and product names are trademarks or registered trademarks of the respective holders.

TRADEMARK NOTATION IN THIS MANUAL

Microsoft® Windows® XP Operating system
Microsoft® Windows® Vista Operating system
Microsoft® Windows® 7 Operating system
Throughout this manual, Windows XP, Windows Vista, and Windows 7 refer to above respective operating systems. In some cases, Windows refers generically to Windows XP, Windows Vista, and Windows 7.

NOTICE

No part of this manual may be copied or reproduced without authorization.
The contents of this manual are subject to change without notice.
Please notify us if you should find any errors in this manual or if you have any comments regarding its contents.

INQUIRIES

Contact the following service center for robot repairs, inspections or adjustments.
If service center information is not indicated below, please contact the supplier office for your region.

Please prepare the following items before you contact us.

- Your controller model and its serial number
- Your manipulator model and its serial number
- Software and its version in your robot system
- A description of the problem

SERVICE CENTER
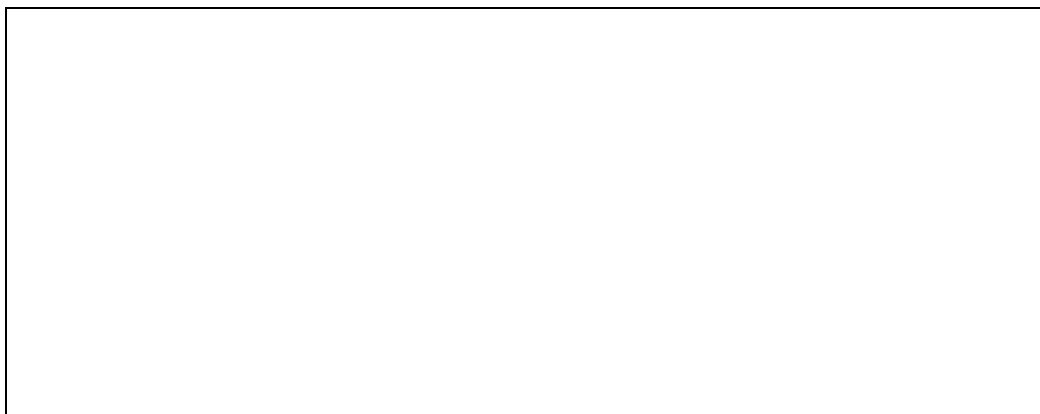
MANUFACTURER

SEIKO EPSON CORPORATION

Toyoshina Plant
Factory Automation Systems Dept.
6925 Toyoshina Tazawa,
Azumino-shi, Nagano, 399-8285
JAPAN
TEL   : +81-(0)263-72-1530
FAX   : +81-(0)263-72-1495

SUPPLIERS

China
EPSON China Co., Ltd

Factory Automation Division
7F, Jinbao Building No. 89 Jinbao Street
Dongcheng District, Beijing,
China, 100005
TEL   : +86-(0)-10-8522-1199
FAX   : +86-(0)-10-8522-1120

Taiwan
EPSON Taiwan Technology & Trading Ltd.

Factory Automation Division
14F, No.7, Song Ren Road, Taipei 110
Taiwan, ROC
TEL   : +886-(0)-2-8786-6688
FAX   : +886-(0)-2-8786-6677

North & South America
EPSON AMERICA, INC.

Factory Automation/Robotics
18300 Central Avenue
Carson, CA 90746
USA
TEL   : +1-562-290-5900
FAX   : +1-562-290-5999
E-MAIL : info@robots.epson.com

Europe
EPSON DEUTSCHLAND GmbH

Factory Automation Division
Otto-Hahn-Str.4
D-40670 Meerbusch
Germany
TEL   : +49-(0)-2159-538-1391
FAX   : +49-(0)-2159-538-3170
E-MAIL : robot.infos@epson.de

Southeast Asia **EPSON Singapore Pte Ltd.**
India   Factory Automation System
                1 HarbourFrontPlace, #03-02
                HarbourFront Tower one, Singapore
                098633
                TEL   : +65-(0)-6586-5696
                FAX   : +65-(0)-6271-3182

Korea   **EPSON Korea Co, Ltd.**
                Marketing Team (Robot Business)
                11F Milim Tower, 825-22
                Yeoksam-dong, Gangnam-gu, Seoul, 135-934
                Korea
                TEL   : +82-(0)-2-3420-6692
                FAX   : +82-(0)-2-558-4271

Japan   **EPSON SALES JAPAN CORPORATION**
                Factory Automation Systems Department
                Nishi-Shinjuku Mitsui Bldg.6-24-1
                Nishishinjuku. Shinjuku-ku. Tokyo. 160-8324
                JAPAN
                TEL   : +81-(0)3-5321-4161

TABLE OF CONTENTS

7 Building SPEL+ Applications   186

1. Introduction

1.1 Welcome to EPSON RC+ 5.0

Welcome to the EPSON RC+ 5.0 Project Management and Development Environment. EPSON RC+ 5.0 is used to develop application software for the EPSON RC90/RC180 Robot Controller.

EPSON RC+ 5.0 features:

- Runs on Microsoft Windows XP, Windows Vista, and Windows 7.

- Integrated application development environment.

- Communicates with controller using USB or Ethernet.

- One PC can be used to manage several controllers.

- Multiple simultaneous sessions.

- SPEL$^+$ programming language. A powerful, easy to use BASIC-like programming language that supports multi-tasking, robot motion control, I/O control, and networking.

- Controller I/O configuration.

- Controller TCP/IP and RS-232 communications setup.

- Wizards for robot tools, local coordinate systems, robot calibration.

- Vision Guide option. Enables you to create a vision system using an EPSON smart camera and Compact Vision.

- VB Guide option. Enables you to control the system using Microsoft .NET programming languages.

- ECP option supports CP motion relative to a fixed point.

- GUI Builder option (Integrated screen creation environment)

1.2 System Overview

The EPSON RC+ 5.0 software is used on a PC that connects with the EPSON RC90/RC180 Controller. EPSON RC+ 5.0 can communicate with the Controller using USB or Ethernet.

EPSON RC+ 5.0 and the Controller can be used in the following configurations:

- **Slave System**   In this configuration, the Controller is a PLC or PC cell slave. EPSON RC+ 5.0 is only used to develop the application. Once the object code is stored in the controller, the PC is not required to be connected to the Controller. The Controller is controlled by remote digital I/O or Fieldbus.

- **Stand Alone System**   In this configuration, a PC is used as a mini cell controller. EPSON RC+ 5.0 can be started in Auto mode to display a simple operator window for the Controller. Or, the VB Guide option can be used to control the Controller from a .NET application.

- **Offline Development System**   In this configuration, a PC is used to edit programs and build projects on an offline computer.

- **Simulation System**   EPSON RC+ 5.0 is used on a PC connected to an Controller without a robot using virtual I/O. I/O simulation tasks can be created to run the application using simulated I/O.

1.2.1 RC180 Controller

The RC180 Controller is a compact robot controller that can drive EPSON SCARA and Pro Six robots.

Features:

- Small and compact, yet powerful.

- Built in motion drive system. The motion drive system can control one robot with up to six motors.

- Includes standard digital I/O.

- Optional digital I/O expansion boards.

- Optional Fieldbus slave support for DeviceNet, EtherNetI/P, PROFIBUS-DP, PROFINET, and CC-Link.

- Optional RS232 ports.

For detailed information on the Controller, refer to the controller manual.

1.2.2 RC90 Controller

The RC90 Controller is a robot controller that can drive EPSON LS series manipulators.

Features:

- Built in motion drive system. The motion drive system can control one robot.

- Standard I/O

- Optional digital I/O expansion boards

- Optional Fieldbus slave support for DeviceNet, PROFIBUS-DP, and CC-Link.

- RS232 ports (standard + optional)

For detailed information on the Controller, refer to the RC90 controller manual.

1.2.3 Software

The EPSON RC+ 5.0 software is installed on the development PC or laptop. To communicate with the controller, the PC must support USB 1.1 / 2.0 or Ethernet communications. Using EPSON RC+ 5.0, you can develop application software for the SPEL+ language that runs in the Controller.

1.2.4 Simulator

Simulator functions enable easy robot motion check on your PC, which gives you flexibility to consider the system layout, measure the operation time, and create the robot programs.

They are useful in all the way from introduction stage of robot automation to launch of robot system.

Simulator is supported from EPSON RC+ 5.0 Ver.5.4 or later.

For details, refer to *8. Simulator*.

1.2.5 System Block Diagram

The following system block diagram shows different methods for connecting a PC running EPSON RC+ 5.0 to one or more controllers.

Method 1: Connect PC to one controller using USB 1.1 or USB 2.0

```
┌──────────┐        USB        ┌──────────────┐
│          │ ◄─────────────► │  RC90/RC180  │
│    PC    │                   │  Controller  │
└──────────┘                   └──────────────┘
```

Method 2: Connect PC to one controller using Ethernet

```
┌──────────┐      Ethernet     ┌──────────────┐
│          │ ◄─────────────► │  RC90/RC180  │
│    PC    │                   │  Controller  │
└──────────┘                   └──────────────┘
```

Method 3: Connect PC to more than one controller using Ethernet.

```
                              ┌──────────────┐
                              │ Controller 1 │
                              │  RC90/RC180  │
                              └──────────────┘
┌──────────┐
│          │ ◄──────────┐
│    PC    │            │     ┌──────────────┐
└──────────┘            │     │ Controller 2 │
                        │     │  RC90/RC180  │
               Ethernet │     └──────────────┘

                              ┌──────────────┐
                              │ Controller 3 │
                              │  RC90/RC180  │
                              └──────────────┘
```
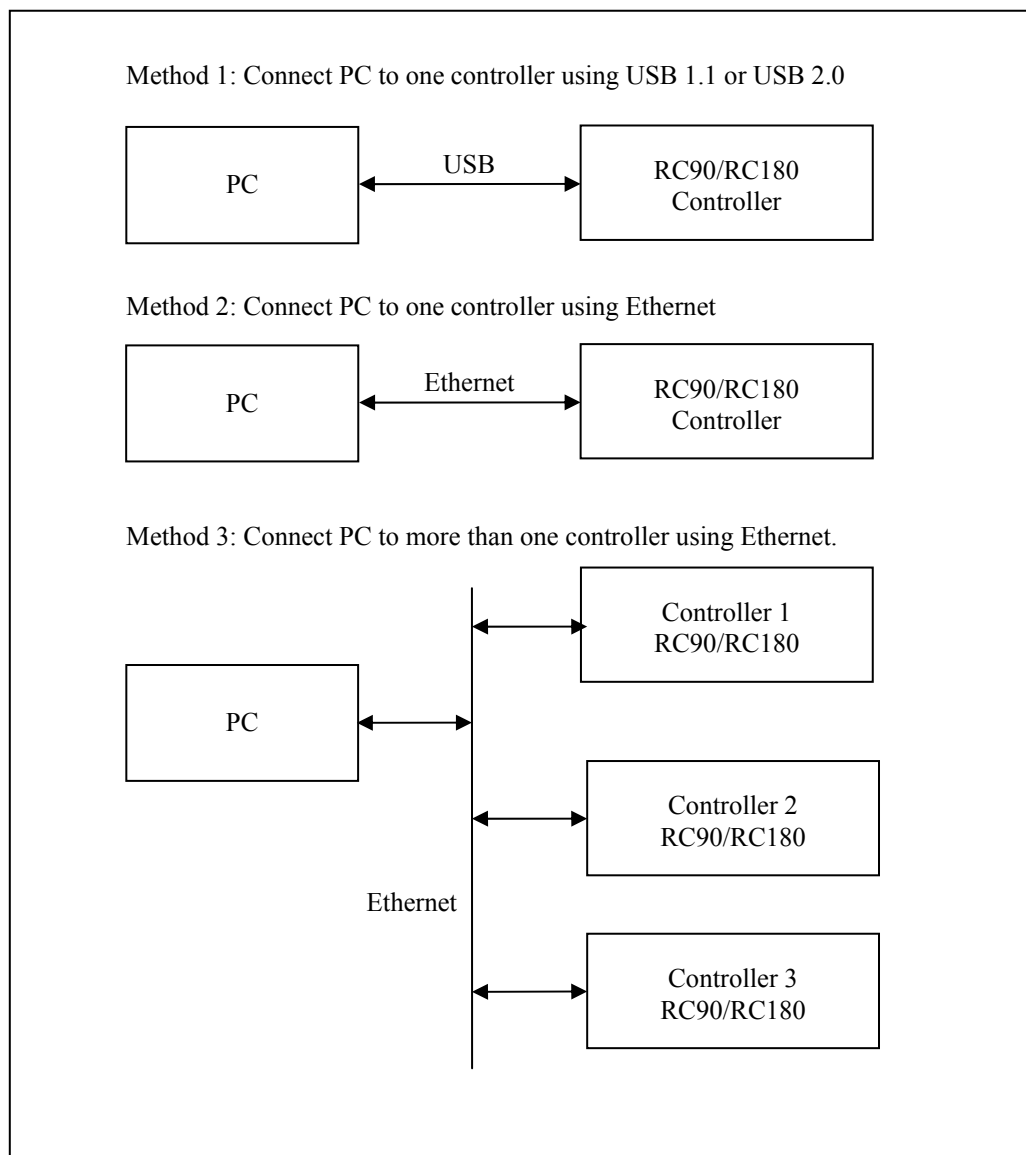
1.3 Software Version

There are the following major software versions for EPSONRC+ 5.0.
   Ver.5.0.*
   Ver.5.1.*
   Ver.5.2.*
   Ver.5.3.*
   Ver.5.4.*

There are the following major firmware versions for the RC180 controller firmware.
   For Controllers with S/N less than 10000

   Ver.1.0.*.*
   Ver.1.2.*.*
   Ver.1.4.*.*
   Ver.1.6.*.*
   Ver.1.8.*.*
   Ver.1.10.*.*
   Ver.1.12.*.*
   Ver.1.14.*.*

   For Controllers with S/N 10001or later
   Ver.1.22.*.*

There are the following major firmware versions for the RC90 firmware.
   Ver.3.0.*.*
   Ver.3.2.*.*

You can connect a controller and PC with any major version combinations.

		RC180 Controller Firmware Version				
		Ver.1.0.*.*	Ver.1.2.*.* Ver.1.4.*.*	Ver.1.6.*.*	Ver.1.8.*.*	Ver.1.10.*.* Ver.1.12.*.* Ver.1.14.*.* Ver.1.22.*.*
EPSON RC+ 5.0	Ver.5.0.*	OK *1	!	!!	!!	!!
	Ver.5.1.*	!	OK *2	!!	!!	!!
	Ver.5.2.*	!	!	OK *3	!!	!!
	Ver.5.3.*	!	!	!	OK *4	!!
	Ver.5.4.*	!	!	!	!	OK

OK: Compatible All functions of the EPSON RC+ 5.0 and the Controller are available.
      *1 Refer to Ver.5.0 manuals.
      *2 Refer to Ver.5.1 manuals.
      *3 Refer to Ver.5.2 manuals.
      *4 Refer to Ver.5.3 manuals.

!: Compatible Functions are partly limited.

!!: Compatible Functions are partly limited.
When using additional functions at the controller, EPSON RC+ 5.0 may not display the proper dialog. The version update of EPSON RC+ 5.0 is recommended.

		RC90 Controller Firmware Version
		Ver.3.0.*.*
EPSON RC+ 5.0	Before Ver.5.4.0	!!!
	Ver.5.4.1 or later	OK

OK: Compatible All functions of the EPSON RC+ 5.0 and the Controller are available.

!!!: Compatible Connection is OK. We recommend using EPSON RC+5.0 Ver. 5.4.1 or later.
Controller firmware update cannot be executed.

The latest software and firmware versions are used for the description in this manual.

EPSON RC+ 5.0   :Ver.5.4.3 or later
RC180 Controller firmware   :Ver.1.14.*.*and Ver.1.22.*.*
RC90 Controller firmware   :Ver.3.2.*.*

1.3.1 Version Confirmation

To confirm the EPSON RC+ 5.0 version, refer to *5.14.6 About EPSON RC+ 5.0 Command (Help Menu)*.

To confirm the RC180 Controller firmware version, refer to *5.12.2 Controller Command (Setup Menu)*.

1.3.2 Restrictions for Version Combination

The functions are restricted according to the combination of the EPSON RC+ 5.0 and the controller firmware.

Combination and Available Functions

EPSON RC+ 5.0 Ver.5.1.* or later / Controller firmware Ver.1.2.*.* or later

		RC180 Controller Firmware	
		Ver.1.0.*.*	Ver.1.2.*.* Ver.1.4.*.* Ver.1.6.*.* Ver.1.8.*.* Ver.1.10.*.* Ver.1.12.*.* Ver.1.14.*.* Ver.1.22.*.*
	Ver.5.0.*	×	×
EPSON RC+ 5.0	Ver.5.1.* Ver.5.2.* Ver.5.3.* Ver.5.4.*	×	OK

OK : Available   × : Not available

- Vision Guide option and command for Vision.

- EtherNet/IP option

- Select items for controller restore

- Setting "Safeguard open stops all tasks"

- TCPSpeed function

EPSON RC+ 5.0 Ver.5.2.* or later / Controller firmware Ver.1.6.*.* or later

		RC180 Controller Firmware	
		Ver.1.0.*.* Ver.1.2.*.* Ver.1.4.*.*	Ver.1.6.*.* Ver.1.8.*.* Ver.1.10.*.* Ver.1.12.*.* Ver.1.14.*.* Ver.1.22.*.*
EPSON RC+ 5.0	Ver.5.0.* Ver.5.1.*	×	×
	Ver.5.2.* Ver.5.3.* Ver.5.4.*	×	OK

OK : Available   × : Not available

- Task execution and monitoring of the following task types
  NoPause   NoEmgAbort
- Tool setting from 4 to 15   - Arm setting from 4 to 15
- Approach check area (Box) function   - Approach check plane (Plane) function
- Remote output for approach check area and approach check plane

- Following commands

AccelMax Function	ErrorOn Function	OLAccel Statement	PlaneClr Statement
Box Statement	EstopOn Function	OLAccel Function	PlaneDef Function
Box Function	InsideBox Function	PauseOn Function	SaftyOn Function
BoxClr Statement	InsidePlane Function	Plane Statement	SysErr Function
BoxDef Function	LJM Function	Plane Function	

  Forced keyword (On, Off, Out, OutW, OpBCD)
  Normal, NoPause, NoEmgAbort keyword (Xqt)

EPSON RC+ 5.0 Ver.5.3.* or later / Controller firmware Ver.1.8.*.* or later

		RC180 Controller Firmware	
		Ver.1.0.*.* Ver.1.2.*.* Ver.1.4.*.* Ver.1.6.*.*	Ver.1.8.*.* Ver.1.10.*.* Ver.1.12.*.* Ver.1.14.*.* Ver.1.22.*.*
EPSON RC+ 5.0	Ver.5.0.* Ver.5.1.* Ver.5.2.*	×	×
	Ver.5.3.* Ver.5.4.*	×	OK *1

OK : Available   × : Not available

- RS series support
- Remote Ethernet function

- Following commands

TC Statement	TCLim Statement	TCLim Function
TCSpeed Statement	TCSpeed Function	RealTorque Statement
J1Flag Statement	J1Flag Function	J2Flag Statement

J2Flag Function

*1 EPSON RC+ 5.0 Ver.5.3.4 or later and the controller firmware Ver.1.8.5.0 or later
   enable the following functions:

   - Go, Pass, Jump3, Jump3CP   - LJM parameter support
   - InReal, OutReal, SetInReal commands   - SysErr parameter support
   - AtHome Function

EPSON RC+ 5.0 Ver.5.4.* or later / Controller firmware Ver.1.10.*.* or later

		RC180 Controller Firmware	
		Ver.1.0.*.* Ver.1.2.*.* Ver.1.4.*.* Ver.1.6.*.* Ver.1.8.*.*	Ver.1.10.*.* Ver.1.12.*.* Ver.1.14.*.* Ver.1.22.*.*
EPSON RC+ 5.0	Ver.5.0.* Ver.5.1.* Ver.5.2.* Ver.5.3.*	×	×
	Ver.5.4.*	×	OK*1

OK : Available   × : Not available

- Simulator function

- Chinese support (Simplified / Traditional)

- Remote input of ForcePowerLow

- Remote output of PowerHigh

- Following commands
  QPDecelR   QPDecelS   ElapsedTime function
  QPDecelR function   QPDecelS function   ResetElapsedTime

*1 EPSON RC+ 5.0 Ver.5.4.1 or later and the controller firmware Ver.1.10.2.0 or later
    enable the following functions:

  - Vx** Commands
  - PROFINET option

    EPSON RC+ 5.0 Ver.5.4.2 or later and the controller firmware Ver.1.12.0.0 or later
    enable the following functions:

  - Compact Vision CV1 option
  - New property methods added to VB Guide 5.0

    EPSON RC+ 5.0 Ver.5.4.3 or later and the controller firmware Ver.1.14.0.0 or later
    enable the following functions:

  - GUI Builder option

1.4 Options

You can purchase options that must be enabled in the controller. EPSON RC+ 5.0 is used to enable these options. See the chapter *15. Installing Controller Options* for details.

1.5 Precautions When Using Windows Vista / Windows 7

Connecting development PC to robot controller using Ethernet
The robot controller does not support internet protocol version 6 (TCP/IPv6). When connecting the development PC to the robot controller using the Ethernet, be sure to use internet protocol version 4 (TCP/IPv4).

Cannot find camera in the network using camera search
When the **Network discovery** is OFF, you cannot find the camera in the network using the camera search. Make sure that the **Network discovery** is ON.

For details, refer to the *EPSON RC+5.0 Option: Vision Guide 5.0 manual*.

For Windows Vista

Set the **Network discovery** ON from [Control Panel]-[Network and Internet]-[Network and Sharing Center] in Windows.

For Windows 7

Set the **Network discovery** ON from [Control Panel]-[Network and Internet]-[Network and Sharing Center]-[Change advanced sharing settings] in Windows.

1.6 EPSON RC+ 3.x and 4.x Users

If you have used previous 3.x and 4.x versions of EPSON RC+ software, you will find that EPSON RC+ 5.0 is very similar.

You can convert your previous EPSON RC+ projects by using Project | Import. EPSON RC+ 5.0 will copy the entire project from the \EPSONRC\Project directory to the \EpsonRC50\Project directory. You can then use the project in the EPSON RC+ 5.0 environment.

1.7 SPEL for Windows Users

If you have used SPEL for Windows 1.x or 2.x software, you will find that EPSON RC+ 5.0 is very similar.

Many new commands have been added to the SPEL$^+$ language, which replaces SPEL. Also, several commands were made obsolete or replaced.

You can convert your SPEL for Windows 2.x project by using Project | Import. EPSON RC+ 5.0 will copy the files to a new directory and optionally convert the programs.

1.8 Documentation

All documentation is installed on the PC in Adobe PDF format. To view manuals on the PC, select Manuals from the Help Menu in EPSON RC+ 5.0. You can also access the manuals from the Windows desktop by clicking Start - Programs - EPSON RC+ 5.0 and click on the manual you want to view.

EPSON RC+ 5.0 User's Guide

This manual contains detailed information for the entire system.

SPEL$^+$ Language Reference Manual

This manual contains detailed information for the entire SPEL$^+$ language.

Vision Guide 5.0 Manual

This manual contains information for using the Vision Guide 5.0 option.

VB Guide 5.0 Manual

This manual contains information for using the VB Guide 5.0 option.

GUI Builder 5.0 Manual

This manual contains information for using the GUI Builder 5.0 option.

Manipulator Manual

This manual contains detailed information for the manipulator you have purchased. Each series of manipulators has its own manuals.

NOTE ☞	The "NOTE" sections describe important information to be followed for operating the Robot system.
TIP ☞	The "TIP" sections describe hints for easier or alternative operations.

2. Safety

2.1 Overview

This chapter explains the important safety requirements for robotic systems using EPSON RC+ 5.0 and the robot Controller.

Installation of robots and robotic equipment should only be performed by qualified personnel in accordance with national and local codes. Please read and understand this entire chapter before using your EPSON RC+ 5.0 system.

Remember that safety is the most important consideration when designing and operating any robotic system.

⚠ WARNING	This symbol indicates that a danger of possible serious injury or death exists if the associated instructions are not followed properly.
⚡ WARNING	This symbol indicates that a danger of possible harm to people caused by electric shock exists if the associated instructions are not followed properly.
⚠ CAUTION	This symbol indicates that a danger of possible harm to people or physical damage to equipment and facilities exists if the associated instructions are not followed properly.

2.2 Definitions

2.2.1 Robot Power

The status of robot power is explained below in terms of restriction to operation:

Operation-prohibited status: Robot cannot be operated.

Restricted (low power) status: Robot can operate at low speed and low torque.

Unrestricted (high power) status: Robot can operate without restriction.

The robot will not operate regardless of the control actions taken by the operator when in the operation-prohibited state. During operation, when the safeguard circuit opens, the system will switch to operation-prohibited state.

The robot will operate at low speed and torque in the restricted state (low power). In the unrestricted state (high power), the robot will operate at the programmed speed and torque.

In the event that the robot should make an unexpected movement, the restricted state (low power) decreases operating speed allowing the operator to avoid danger. The torque is also decreased to minimize serious injury to the operator should one be struck by the robot. The maximum values of the decreased speed and torque are set according to the robot used and cannot be changed by the user.

As a safety precaution the initial power state of the robot will be set to either the restricted (low power) state or the operation-prohibited state. The system will not change to the unrestricted (high power) state if the appropriate procedures are not followed.

When the system is in restricted (low power) state or operation-prohibited state, a single failure will not cause a runaway action that surpasses the assigned speed or torque decrease. This is due to the multi-protect circuit and mutual monitoring circuit in the control system.

2.2.2 Safeguard

To ensure safe operation of the robotic work cell, you must install a safety system using safety doors, light curtains, safety floor mats, etc.

⚠️ WARNING	▪ There is a safeguard input circuit in the EMERGENCY connector on the controller that connects with the safety device interlock switch. In order to protect those working with the robot be sure that the interlock switch is connected and working properly.

If a closed safeguard is open during robot motion, the robot stops immediately and enters pause state. All robot motors are turned off. The descriptions below explain how the safeguard input works.

Safeguard closed: The safeguard input is turned ON. The robot can automatically operate in unrestricted (high power) state.

Safeguard open: The safeguard input is turned OFF, and the interlock function operates. The robot stops immediately, motors are turned off, and further operation is impossible until either the safeguard is closed or Teach mode is turned ON and the enable circuit is engaged.

For further details on the safeguard and interlock, refer to the section *Installation and Design Precautions* later in this chapter. For detailed wiring instructions, refer to the *Setup & Operation EMERGENCY* in the controller manual.

2.2.3 Operation Modes

The operation mode is defined as the single control point for the controller, therefore you cannot use more than one operation mode at the same time.

There are three operation modes for the controller: AUTO, PROGRAM, and TEACH.

- AUTO operation modes allow you to execute programs in the controller when the safeguard is closed.

- PROGRAM operation mode allows you to execute and debug programs when the safeguard is closed.

- TEACH operation mode allows you to jog and teach the robot at slow speed while inside the safeguarded area.

2.2.4 Start Mode

The Start mode specifies the operation mode for EPSON RC+ 5.0 when it starts.

You can set the EPSON RC+ 5.0 to start in AUTO or PROGRAM mode.

For information on how to change the start mode, refer to the *Operation* chapter.

2.2.5 Changing Operation Mode

You can change from AUTO operation mode or PROGRAM operation mode to TEACH mode by setting the mode selector key switch on the Teach Pendant to the TEACH position.

When the mode selector key switch is changed back to AUTO, the operation mode is returned to previous operation mode (AUTO or PROGRAM).

The AUTO operation mode can be changed to PROGRAM mode during the EPSON RC+ 5.0 startup sequence. A password can be used to allow only certain personnel to change the startup operation mode.

When EPSON RC+ 5.0 starts in AUTO operation mode, the AUTO operation mode cannot be changed to PROGRAM operation mode after the system has started. To change the operation mode, restart the system and log into PROGRAM mode, then set the start mode again and restart EPSON RC+ 5.0.

For more information, refer to the *Operation* chapter.

2.2.6 Emergency Stop

The controller is equipped with an emergency stop input terminal. If the normally closed emergency stop circuit is broken, the power supplied to all motors will be shut off (and enter servo-free status) and the robot will be stopped by dynamic braking.

⚠ CAUTION	■ The path that the robot will follow from the time the emergency stop switch is pressed until the device stops, as well as the stop position itself, cannot be positively determined. In many cases, the stop position will not exceed the target position for the operation prior the emergency stop. Depending on the robot's loading condition and operation speed, overruns are inevitable. Taking this into consideration, be sure the layout for the peripheral equipment includes extra space.

For detailed wiring instructions, refer to the section *EMERGENCY* in the controller manual.

2.2.7 Teach Control Device

Operators can use the TP1 teach pendant to operate the robot in the TEACH operation mode.

Refer to the TP1 manual for operation instructions.

2.3 Safety-related Requirements

Specific tolerances and operating conditions for safety are contained in the manuals for the robot, controller and other devices. Be sure to read those manuals as well.

For the installation and operation of the robot system, be sure to comply with the applicable local and national regulations.

Robot systems safety standards and other examples are given in this chapter. Therefore, to ensure that safety measures are complete, please refer to the other standards listed as well.

(Note: The following is only a partial list of the necessary safety standards.)

EN775	European Standard; Manipulating industrial robots - Safety.
ANSI/RIA R15.06	American National Standard; Industrial Robots and Robot Systems - Safety Requirements .
IEC204-1 (EN60204-1)	Safety of machinery - Electrical equipment of machines
	Part 1. Specification for general requirements.
EN292-1,-2	Safety of machinery - Basic concepts, general principles for design
	Part 1. Basic terminology, methodology
	Part 2. Technical principles and specifications
EN418	Emergency stop equipment, functional aspects - principles for design.
EN953	General requirements for design and construction of guards.
EN55011	Limits and methods of measurement of radio disturbance characteristic of industrial scientific and medical (ISM) radio frequency equipment
EN61000-6-2	Electromagnetic compatibility (EMC)
	Part 6-2. Generic standards - Immunity for industrial environments
EN1050	Safety of machinery Principles for risk assessment
EN954-1	Safety of machinery-Safety related parts of control systems

RC180 Controller - UL Specification

Compatibility assessment of the UL specification model is performed according to the following standards.

UL1740 (Third Edition, Dated December 7, 2007)
ANSI/RIA R15.06-1999
NFPA 79 (2007 Edition)
CSA/CAN Z434-03 (February 2003)

2.4 Installation and Design Precautions

2.4.1 Designing a Safe Robot System

It is important to operate robots safely. It is also important for robot users to give careful consideration to the safety of the overall robot system design.

This section summarizes the minimum conditions that should be observed when using EPSON robots in your robot systems.

Please design and manufacture robot systems in accordance with the principles described in this and the following sections.

Environmental Conditions
Carefully observe the conditions for installing robots and robot systems that are listed in the "Environmental Conditions" tables included in the manuals for all equipment used in the system.

System Layout
When designing the layout for a robot system, carefully consider the possibility of error between robots and peripheral equipment. Emergency stops require particular attention, since a robot will stop after following a path that is different from its normal movement path. The layout design should provide enough margins for safety. Refer to the manuals for each manipulator, and ensure that the layout secures ample space for maintenance and inspection work.

When designing a robot system to restrict the area of motion of the robots, do so in accordance with the methods described in each manipulator manual. Utilize both software and mechanical stops as measures to restrict motion.

Install the emergency stop switch at a location near the operation unit for the robot system where the operator can easily press and hold it in an emergency.

Do not install the controller at a location where water or other liquids can leak inside the controller. In addition, never use liquids to clean the controller.

Disabling Power to the System using lock out / tag out
The power connection for the robot controller should be such that it can be locked and tagged in the off position to prevent anyone from turning on power while someone else is in the safeguarded area. For further details, refer to the section *Procedure of Lockout/Tagout* in the chapter *Safety Precautions* in the controller manual.

End Effector Design
Provide wiring and piping that will prevent the robot end effector from releasing the object held (the work piece) when the robot system power is shut off.

Design the robot end effector such that its weight and moment of inertia do not exceed the allowable limits. Use of values that exceed the allowable limits can subject the robot to excessive loads. This will not only shorten the service life of the robot but can lead to unexpectedly dangerous situations due to additional external forces applied to the end effector and the work piece.

Design the size of the end effector with care, since the robot body and robot end effector can interfere with each other.

Peripheral Equipment Design

When designing equipment that removes and supplies parts and materials to the robot system, ensure that the design provides the operator with sufficient safety. If there is a need to remove and supply materials without stopping the robot, install a shuttle device or take other measures to ensure that the operator does not need to enter a potentially dangerous zone.

Ensure that an interruption to the power supply (power shutoff) of peripheral equipment does not lead to a dangerous situation. Take measures that not only prevent a work piece held from being released as mentioned in "End effector Design" but that also ensure peripheral equipment other than the robots can stop safely. Verify equipment safety to ensure that, when the power shuts off, the area is safe.

Remote Control

To prevent operation by remote control from being dangerous, start signals from the remote controller are allowed only when the control device is set to REMOTE, TEACH mode is OFF, and the system is configured to accept remote signals. Also when remote is valid, motion command execution and I/O output are available only from remote. For the safety of the overall system, however, safety measures are needed to eliminate the risks associated with the start-up and shutdown of peripheral equipment by remote control.

Emergency Stop

Each robot system needs equipment that will allow the operator to immediately stop the system's operation. Install an emergency stop device that utilizes emergency stop input from the controller and all other equipment.

During an emergency stop, the power that is supplied to the motor driving the robot is shut off, and the robot is stopped by dynamic braking.

The emergency stop circuit should also remove power from all external components that must be turned off during an emergency. Do not assume that the robot controller will turn off all outputs if configured to. For example, if an I/O card is faulty, the controller cannot turn off a component connected to an output. The emergency stop on the controller is hardwired to remove motor power from the robot, but not external power supplies.

Do not press the Emergency Stop switch unnecessarily while the Manipulator is operating. Pressing the switch during the operation makes the brakes work. This will shorten the life of the brakes due to the worn friction plates.
 Normal brake life cycle: About 2 years (when the brakes are used 100 times/day)

Before using the Emergency Stop switch, be aware of the followings.
- The Emergency Stop (E-STOP) switch should be used to stop the Manipulator only in case of emergencies.
- To stop the Manipulator operating the program except in emergency, use Pause (halt) or STOP (program stop) commands.
  Pause and STOP commands do not turn OFF the motors. Therefore, the brake does not function.
- For the Safeguard system, do not use the circuit for E-STOP.

For details of the Safeguard system, refer to the following manuals.
 Safety and Installation 2.6 Connection to EMERGENCY Connector
To check brake problems, refer to the following manuals.
 Manipulator Manual Maintenance 2.2.2 Inspection While the Power is ON
                (Manipulator is operating)
 Safety and Installation 5.2 Inspection Point - Inspection While the
                Power is ON (Manipulator is operating)

Safeguard System

To ensure safety, a safeguard system should be installed for the robot system.

When installing the safeguard system, strictly observe the following points:

Refer to each robot manual, and install the safeguard system outside the maximum space. Carefully consider the size of the end effector and the work pieces to be held so that there will be no error between the moving parts and the safeguard system.

Manufacture the safeguard system to withstand calculated external forces (forces that will be added during operation and forces from the surrounding environment).

When designing the safeguard system, make sure that it is free of sharp corners and projections, and that the safeguard system itself is not a hazard.

Make sure that the safeguard system can only be removed by using a tool.

There are several types of safeguard devices, including safety doors, safety barriers, light curtains, safety gates, and safety floor mats. Install the interlocking function in the safeguard device. The safeguard interlock must be installed so that the safeguard interlock is forced to work in case of a device failure or other unexpected accident. For example, when using a door with a switch as the interlock, do not rely on the switch's own spring force to open the contact. The contact mechanism must open immediately in case of an accident.

Connect the interlock switch to the safeguard input of the drive unit's EMERGENCY connector. The safeguard input informs the robot controller that an operator may be inside the safeguard area. When the safeguard input is activated, the robot stops immediately and enters pause status, as well as either operation-prohibited status or restricted status (low power status).

Make sure not to enter the safeguarded area except through the point where the safeguard interlock is installed.

The safeguard interlock must be installed so that it can maintain a safe condition until the interlock is released on purpose once it initiates. The latch-release input is provided for the EMERGENCY connector on the Controller to release the latch condition of the safeguard interlock. The latch release switch of the safeguard interlock must be installed outside of the safeguarded area and wired to the latch-release input.

It is dangerous to allow someone else to release the safeguard interlock by mistake while the operator is working inside the safeguarded area. To protect the operator working inside the safeguarded area, take measures to lock out and tag out the latch-release switch.

Presence Sensing Device

The above mentioned safeguard interlock is a type of presence sensing device, since it indicates the possibility of somebody being inside the safeguard system. When separately installing a presence sensing device, however, perform a satisfactory risk assessment and pay thorough attention to its dependability.

Here are precautions that should be noted:

- Design the system so that when the presence sensing device is not activated or a dangerous situation still exists that no personnel can go inside the safeguard area or place their hands inside it.

- Design the presence sensing device so that regardless of the situation the system operates safely.

- If the robot stops operating when the presence sensing device is activated, it is necessary to ensure that it does not start again until the detected object has been removed. Make sure that the robot cannot automatically restart.

Resetting the Safeguard

Ensure that the robot system can only be restarted through careful operation from outside the safeguarded system. The robot will never restart simply by resetting the safeguard interlock switch. Apply this concept to the interlock gates and presence sensing devices for the entire system.

Robot Operation Panel

When using the robot operation panel, it must be installed so as to operate the robot system from outside the safeguard.

2.4.2 Robot System Installation, Start-up, and Testing

Installation

When installing the robot and robot system, follow the instructions contained in each of the manipulator and controller manuals.

Start-up and Functional Testing

If the safeguard system is not ready at the time of start-up and functional testing, specify an area to install the safeguard system (as a temporary measure) and then begin.

During start-up and functional testing, do not allow workers inside the safeguarded area until the safeguard function is activated.

Before start-up and functional testing, carefully read the related manuals and obtain a good understanding of safety-related precautions.

Before supplying the robot and robot system with power for the first time, verify the items listed below.

Items to check before supplying with power

- Prescribed bolts are securely tightened to the robot.

- Electrical connections are set up correctly, and power supply conditions (including voltage, frequency, and error level) are within the specified range.

- Compressed air source (if applicable) is properly connected.

- Peripheral devices are properly connected.

- Safety device is equipped with an interlock switch, and it functions properly.

- Operating environment conditions conform to the conditions specified in the manipulator and controller manuals.

Items to check after supplying with power

- Start/stop, mode selection, and other functions work properly.

- Moving axes operate normally, and that the area of motion is limited as stipulated in the specifications.

- Emergency stop circuit functions correctly.

- Power supply can be shut off.

- Teach operation mode is functioning properly.

- Safety device and interlock switch function correctly.

- Other safeguards (if applicable) are installed correctly in their prescribed locations.

- Robot operates accurately in restricted status (low power status).

- Robot operates properly under rated loads and at maximum speed.

Restarting after a Change
When restarting the robot system after its hardware or software has been corrected or serviced, strictly observe the following:

- Before supplying the system with power, check the locations where the hardware was modified.

- Test the functions of the robot system to make sure that it operates correctly.

2.5 Precautions regarding Robot Operation

2.5.1 General Precautions

Before operation, become familiar with the location of all emergency stop switches.

During an emergency, always press the nearest emergency stop switch. There should never be any emergency stop switches in the system that do not operate.

After an emergency, do not restore the emergency stop circuit until it has been determined that the entire system is safe to restart.

If your robot is a 6-Axis type, record the pulse values of the reference points used for calibration. For details, refer to the section *Setup & Operation 3.6 Setting the Reference Points for Calibration* in the 6-Axis manipulator manual.

2.5.2 Automatic Operation

Ensure that system automatic operation is enabled only while the following requirements are being met:

- Emergency stop switches are installed in the prescribed location and operate correctly.

- No personnel are inside the safeguarded area of the system.

- Safety procedures that are established separately for the robot system (if applicable) are being followed.

2.5.3 Teaching Robot Points

If possible, teaching should be performed with no personnel inside the safeguarded area.

Teach mode can be used to allow the robot to be jogged or moved at slow speed when the safeguard is open. Before going inside the safeguarded area, robot operators that need to move the robot under servo control must switch Teach mode to ON by using the mode selector key switch of the teach pendant. Operators then carry the teach pendant while inside the safeguarded area. As a result, the operation mode cannot be changed from outside the safeguarded system while somebody is inside the safeguarded area.

Auto Mode and Program Mode

With the safeguard circuit open, the robot motors will be turned off and the robot cannot be jogged under power. However, the robot can be moved by hand to a position with the safeguard circuit open and the position can then be taught.

Teach Mode

The robot can be jogged or moved at slow speed as long as the three position enable (dead man) switch is engaged.

Please observe the following guidelines for teaching points:

- Robot operators must receive training that utilizes the same type of robot. Before teaching, the operator should be thoroughly familiar with teaching procedures.

- Before teaching, remove all errors and malfunctions.

- Before the robot operator goes inside the safeguard system, confirm that the robot motors go off when the safeguard is open and that emergency stop switches are functioning correctly.

- The robot operator should visually check the robot system and safeguard system interior to ensure that there are no potential hazards.

- Design the system in such a way that prevents the overall robot system from being started up from any location while the operator is inside the safeguarded area.

- If there is a possibility of a dangerous situation arising from the operation of a device other than the robot, such as an actuator, take steps to prevent such operation or ensure that these devices can only be controlled by the teaching operator.

2.5.4 Return to Automatic Operation

If there are safety devices that have been temporarily disabled for system inspection or other reasons, always return them to their original status before restarting automatic operation.

2.5.5 Program Verification

If it is necessary to verify a program while the system is in unrestricted (high power) status, first make sure that no personnel are inside the safeguarded area.

2.5.6 Troubleshooting

Troubleshoot from outside the safeguard system. If that is not possible, strictly observe the requirements below.

- Operators responsible for troubleshooting should be trained and qualified to perform such work.

- Establish work safety procedures to minimize the danger that operators inside the safeguard system will be exposed to.

2.5.7 Maintenance

In order to keep the robot and robot system operating safely, maintenance (and inspection) is important. Adequately trained personnel should perform the procedures required to do the maintenance work safely. Make sure that maintenance is performed according to the instructions in the manipulator and controller manuals (maintenance sections).

If maintenance is required inside the safeguarded area, take the following precautions:

- Shut off the power supply using lockout / tagout to prevent anyone from turning ON the robot power supply by mistake. For further details, refer to the section *Procedure of Lockout/Tagout* in the chapter *Safety Precautions* in the controller manual.

- If the robot system power supply cannot be shut off, strictly observe the following:

  (1) Visually inspect the robot system to ensure that there are no conditions that could lead to a malfunction.

  (2) If it is discovered that the robot system is damaged or malfunctioning, perform the required repairs and retest it before allowing the operator to go inside the safeguard system.

- Grant full control of the robot and robot system to those performing maintenance and/or repairs inside the safeguard system.

- Ensure that the robot system does not respond to any remote control devices.

- Ensure that all emergency stop devices are functioning correctly.

- Before starting the robot system in automatic operation, return all temporarily disabled safety devices to their original enabled status.

- Do not use tweezers or other metal tools to aid in battery replacement. This could cause a battery short. Replace a battery using only the specified type and be careful to observe the polarity of the battery.

2.5.8 Backup of Projects and Controller

After a project has been created or edited, or after system data including robot parameters has been edited, the project and controller files should be copied and stored in media other than the hard disk on the PC (e.g. USB memory key). Keep the backup media in a safe place in case of damaged data on the hard disk

To backup, select Controller from the EPSON RC+ 5.0 Tools Menu and execute Backup Controller. Refer to the section *Controller Command (Tools Menu)* in the chapter *The EPSON RC+ 5.0 GUI* in this manual.
Backup Controller is a function to backup both the project and the controller.

To backup only the project data, select Copy from the EPSON RC+ 5.0 Project Menu. Refer to the section *Copy Command (Project Menu)* in the chapter *The EPSON RC+ 5.0 GUI* in this manual.

NOTE

☞

- When the version of the connected RC180 controller firmware is Ver.1.0.*.*, Backup Controller does not backup the project data. Be sure to backup the project data individually.
- You cannot restore the backup data of the RC180 robot controller to RC90 robot controller. Also, you cannot restore the backup data of RC90 robot controller to RC180 robot controller.

⚠ CAUTION	■ If your system cannot be restored using Restore Controller, you must restore robot calibration parameters (Hofs, CalPls) before operating the robot. Failure to do so will cause the robot to move to incorrect positions.

2.6 End User Instruction Manual

Be sure that the robot system instruction manual supplies a list of all the equipment included in the system (such as the robots, associated equipment, and safety devices) as well as a description of how to use each.

Be sure to provide the following in the manual:

- An easy to follow explanation of the robot system and how to install it, as well as a step by step summary of the system installation and external power supply connections.
- A description of all hazards and how to avoid them.
- A description (including interconnection diagrams) of the safety devices, interacting functions, and safeguard's interlocking function against hazardous conditions, in particular, the safeguard's interlocking function for devices installed to perform interactively.
- Precise instructions regarding usage of the system.

2.7 End User Training

Be sure those in charge of safety management confirm that the operators who program, operate, and maintain the robot and robot system obtain proper training and have the expertise to conduct the work safely.

Training should include at least the following:

- Study of regulation safety procedures, and safety-related recommendations by robot manufacturers and system designers.
- Clear explanation of the work involved.
- Description of all control equipment required for the work and their functions.
- Explanation of potential hazards involved in the work.
- Work safety procedures and specific methods of avoiding potential hazards.
- Safety device and interlock function testing and confirmation methods are working properly.

3. Getting Started

This chapter contains instructions for setting up and using EPSON RC+ 5.0. It is recommended that first time users first read the preceding Safety chapter, then read through this chapter to get more familiar with the system.

Contents

- Hardware Installation
- Software Installation

3.1 Hardware Installation

EPSON RC+ 5.0 is used with the EPSON RC90/RC180 robot Controller. You need to install the controller and robot before you can use EPSON RC+ 5.0 to develop and run SPEL+ applications. You also need a PC running Windows that will run EPSON RC+ 5.0 and connect to the controller via USB or Ethernet.

The Controller comes pre-configured at the factory. For instructions on installation, refer to the RC90/RC180 robot Controller manual.

3.2 Software Installation

EPSON RC+ 5.0 must be installed on a user supplied PC that runs Windows. Refer to the section *Installing EPSON RC+ 5.0 Software* in *Appendix A* for instructions on installing EPSON RC+ 5.0.

4. Operation

This chapter contains instructions for operation of the EPSON RC+ 5.0 system. The main topics are:

- Simple Mode
- System Power Up Procedure
- Starting EPSON RC+ 5.0
- System Shutdown Procedure
- Communications with Controller
- Writing your first program

4.1 Simple Mode

EPSON RC+ 5.0 supports two development environments: Simple Mode and Normal Mode. Simple Mode is ideal for beginners or for simple application development.

Simple Mode features include:

- **One program file and one point file** No file management is required.

- **No project configuration required** You do not need to edit the project configuration.

- **Simple Robot Manager** Three pages of the Robot Manager are used in simple mode: Control Panel, Jog & Teach, and Points.

- **Simple Help** When using Simple Mode, the Help system is tailored for simple mode.

To run EPSON RC+ 5.0 in Simple Mode, the command line option /Simple is used. During installation, a desktop shortcut is created for Simple Mode. You can also start in Simple Mode by select EPSON RC+ 5.0 Simple from Start | Programs | EPSON RC+ 5.0.

NOTE ☞ You cannot use Vision Guide 5.0 option in Simple mode.
To use Vision Guide 5.0, make sure that the mode is in Normal mode.

4.2 System Power Up Procedure

Follow this procedure to power up the system:

1. Ensure that all safeguards are in place and that all personnel are clear of the equipment.

2. Apply power to the Controller and I/O devices. Also apply power to the PC and monitor, if used.

3. Start the EPSON RC+ 5.0 software on the PC if a PC is being used in the system.

4.3 Starting EPSON RC+ 5.0

There are three ways to start EPSON RC+ 5.0. You can also configure the mode that EPSON RC+ 5.0 starts in.

Start Method 1

1. To use Simple Mode:
Double click on the EPSON RC+ 5.0 Simple robot icon located on the Windows desktop.

To use Normal Mode:
Double click on the EPSON RC+ 5.0 robot icon located on the Windows desktop.

Start Method 2

1. Click the Windows Start button.

2. Select the EPSON RC+ 5.0 Program Group.

3. To use Simple Mode:
   Select EPSON RC+ 5.0 Simple

   To use Normal Mode:
   Select EPSON RC+ 5.0.

Start Method 3

1. Configure EPSON RC+ 5.0 to start automatically after Windows starts. The details are described later in this chapter in the section *Startup Configuration*.

NOTE
☞

When using the VB Guide option, you do not need to start EPSON RC+ 5.0. The library provided with VB Guide will load EPSON RC+ 5.0 into your .NET application process automatically.

4.3.1 Startup sequence

When EPSON RC+ 5.0 starts, it reads initial settings for the current user and local system from the Windows registry.

If there are no project files specified on the startup command line, then the last project that was opened will be opened at startup time. The program files that were last opened will be opened again.

If the Start Mode is Auto, the Start Mode dialog is opened (see the *Start Mode Dialog* section later in this chapter).

If the Start Mode is Program, the EPSON RC+ 5.0 GUI is opened.

4.3.2 Startup Configuration

To configure startup, select Preferences from the Setup Menu. The Startup section has pages for Start Mode, Auto Start, and Windows Login.

4.3.3 Start Mode

This page has settings for the EPSON RC+ 5.0 start mode.
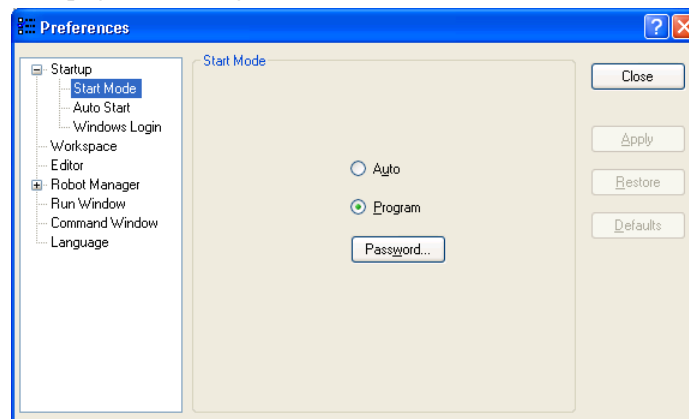
There are two start modes:

Program   This mode allows you to develop your projects. This is the default startup mode.

Auto   This mode starts the system and displays the Operator Window.

Use the **Password** button to change the start mode password.

4.3.4 Start Mode Dialog

When the start mode is set for Auto, then a dialog is displayed at start up that allows you to change the startup mode using a password. After several seconds, if the **Change To Program Mode** button has not been clicked, the system will initialize and the Operator Window will be displayed.

You can disable this startup dialog using command line options described later in this section.
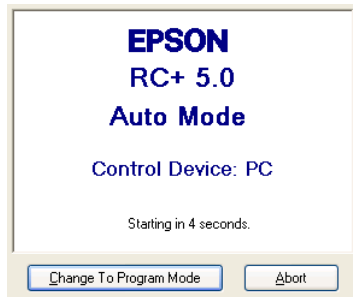
If you click the **Change To Program Mode** button, another dialog will be displayed, as shown below:
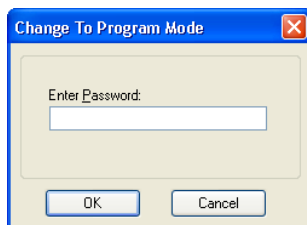
To change to Program mode, you must supply the password and click **OK**, or you can abort startup all together by clicking **Cancel**.

This allows authorized personnel to enter Program mode temporarily to make changes or adjustments.

NOTE
☞
When you change to PROGRAM mode from this dialog, it is only temporary. The next time EPSON RC+ 5.0 runs, the original start mode setting will be used. To permanently change the start mode, you must change it from the Setup | Preferences | Startup | Start Mode page.

4.3.5 Start Mode: Program

Program mode is the default start mode. This is the EPSON RC+ 5.0 development environment, from which you can:

- Create / edit projects.

- Configure the controller and set preferences.

- Run and debug programs.

4.3.6 Start Mode: Auto

Auto mode displays the Operator Window. The Operator Window is configured according to the settings in Project | Properties.

The current controller Control Device sets the mode of Auto operation.

Control Device	Description
PC	The Operator Window can be used as a simple operator interface for production.
Remote I/O	The Operator window is displayed with no operator buttons to allow any diagnostic messages to be viewed.
OP	
Remote Ethernet	

4.3.7 Auto Start

You can configure EPSON RC+ 5.0 to automatically start when Windows starts.

From the Setup | Preferences | Auto Start page, set the **Start EPSON RC+ 5.0 after Windows start** check box.

You can specify EPSON RC+ 5.0 command line options (/auto, /nosplash, etc.) in the Command line options text box. Refer to the section *Command Line Options* in this chapter for details.
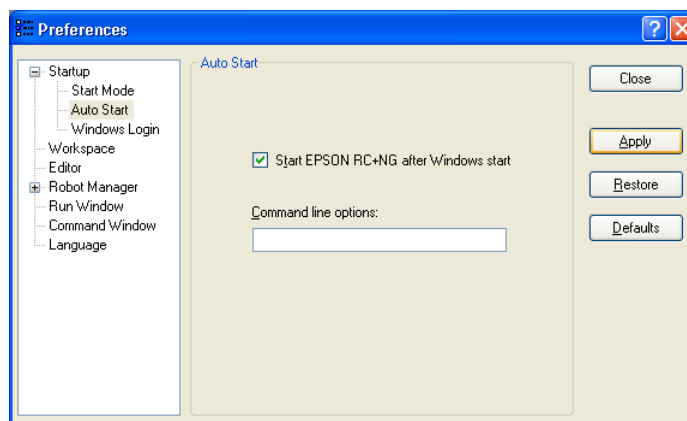
4.3.8 Windows Login

You can configure automatic Windows login from EPSON RC+ 5.0. From the Setup |
Preferences | Windows Login page, type in the name and password of the user logging in.
Set the **Enable windows auto login** check box. Optionally, you can supply a domain, if
required.

You must have Windows Administrator rights to set login parameters. After making
changes, you must reboot the system the first time. After the first reboot, Windows login
will be automatic for boot or logout.
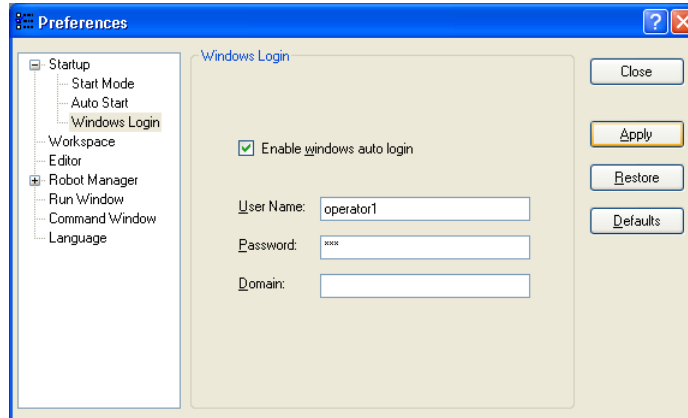
4.3.9 Command Line Options

There are command line options for EPSON RC+ 5.0 that provide the following functions:

- Start EPSON RC+ 5.0 in Simple Mode.

- Start EPSON RC+ 5.0 for a specified project.

- Change the EPSON RC+ 5.0 startup mode without a dialog.

- Disable splash screen

Starting EPSON RC+ 5.0 in Simple Mode
You can start EPSON RC+ 5.0 in Simple Mode using the /Simple command line option.

```
ERC50.EXE  /SIMPLE
```

Starting EPSON RC+ 5.0 for a specific project
When you start EPSON RC+ 5.0, you can optionally specify a project name in the
command line.

```
ERC50.EXE  [drive:project_name]
```
drive:project_name The drive letter and name of a project. The name can include a
subfolder of the \EpsonRC50\Projects directory.

Example

Open project *myapp* on drive C: at startup:

```
ERC50.EXE c:myapp
```

Change the EPSON RC+ 5.0 startup mode
You can select the startup mode and override the startup dialog using command line options.

To start in Program mode (no password required)

```
ERC50.EXE  /PROG
```
To start in Auto mode

```
ERC50.EXE  /AUTO
```
Use these command line options to override and hide the startup dialog and open the Operator Window directly.

If only the AUTO flag is supplied and the control device is PC, EPSON RC+ 5.0 will open the project from the last session and display the operator window. EPSON RC+ 5.0 will only be visible in the Windows Task Manager. When the operator window is closed, EPSON RC+ 5.0 will be terminated.

NOTE
☞

When the control device is PC, you cannot close the operator window while tasks are running.

Example

Open project myapp on drive C and display the operator window:

```
ERC50.EXE c:myapp /AUTO
```

NOTE
☞

The Controller should be ON before starting EPSON RC+ 5.0 with the /AUTO command line option. If EPSON RC+ 5.0 cannot communicate with the controller, then an error message will be displayed with a retry button.

For more details, see the Operator Window section in the chapter *Building Applications in SPEL⁺*.

Disabling the EPSON RC+ 5.0 splash window
You can suppress the splash window displayed at startup using the following syntax:

```
ERC50.EXE /NOSPLASH
```

4.3.10 Using Command Line Options

Examples of command line options are:

- Command input from Windows Start Menu | Run

- Target property of desktop shortcut

Running from the Windows Run Box
You can specify a command (e.g. C:\EpsonRC50\exe\erc50.exe C:myapp) from the Windows Start menu | Run | Open text box.

Making startup icons for your projects
You can create icons that automatically start EPSON RC+ 5.0 for different projects and start modes.

1. Create a new shortcut to ERC50.EXE on your desktop by selecting the EPSON RC+ 5.0 icon, then right clicking and selecting Create Shortcut.

2. Right click on the icon and select Properties.
    In the command line text box, type ERC50.EXE followed by the name of the project file you want to open as described above. Add any other options to the command line, such as /AUTO, /PROG, etc.

4.4 Communications with Controller

Your PC running EPSON RC+ 5.0 can communicate with a controller using USB or Ethernet.

4.4.1 Configuring Communications with the Controller

To configure communications with the controller, select PC to Controller Communications from the Setup Menu. This will open the dialog shown below:
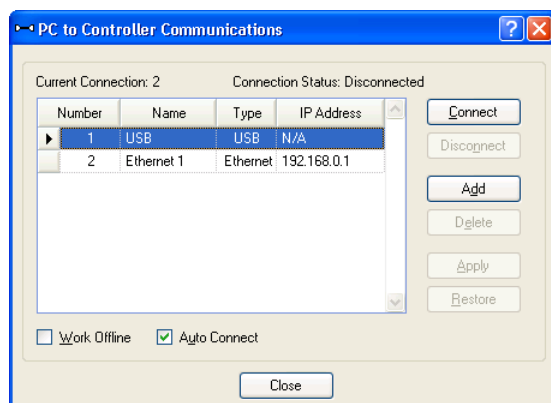
The dialog has a list of connections. The first connection is for USB and is fixed. You cannot delete it or rename it.

You can add one or more Ethernet connections and give each one a meaningful name.

The name for each connection is also shown in the Connections dropdown list on the main toolbar. If no name is supplied, the Ethernet IP address is shown in the dropdown list.

For more information on PC to Controller Communications, see the PC to Controller Communications section in chapter 5.

4.4.2 USB Communications

USB 2.0 or USB 1.1 can be used to communicate with one controller. This is the default communication method for EPSON RC+ 5.0 and requires no configuration.

To connect to a controller via USB:

1. Connect a USB cable between the PC and the controller.

2. Turn on power to the controller.

3. Start EPSON RC+ 5.0.

4. Click the PC to Controller Communications button on the tool bar.

5. Ensure that connection #1 (USB) is selected.

6. Click the Connect button.

7. Click the Close button.

4.4.3 Ethernet Communications

NOTE
☞

The robot controller does not support internet protocol version 6 (TCP/IPv6). When connecting the development PC to the robot controller using the Ethernet, be sure to use internet protocol version 4 (TCP/IPv4).

You can communicate with one or more controllers from one PC using Ethernet. With Ethernet communications, each controller must have a unique IP address. You can set the IP address, mask, and gateway for the controller from the Setup | Controller | Configuration. The gateway setting is only required if you will be accessing the controller from outside of the local network.

You can connect a PC to a controller directly using an Ethernet cross-over cable, or you can connect the PC and controller to an Ethernet switch or hub.

Before you can communicate with a controller using Ethernet, you must configure the controller's IP address, IP mask, and IP gateway. This is accomplished by first connecting to the controller with USB, and then from the EPSON RC+ 5.0 Setup | Controller | Configuration page, set the IP address, IP mask, and IP gateway of the controller as shown below.

The following is the configuration of the controller at the time of shipment.

IP Address : 192.168.0.1
IP Mask : 255.255.255.0
IP Gateway : 0.0.0.0

Use the USB connection to configure Ethernet communications.
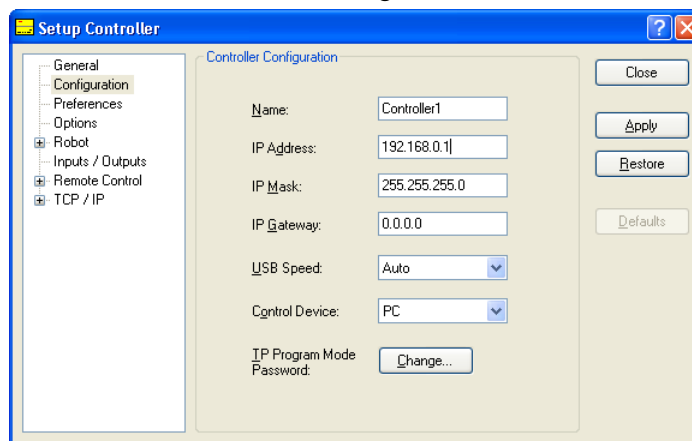
4.4.4 Connecting When Control Device is no PC

Connecting while control device is not PC and tasks are not running

When you connect to a controller whose control device is not PC and tasks are not running, you will see the following message box:
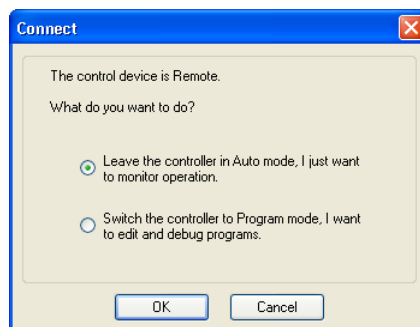
This allows you leave the controller in Auto mode to monitor operation, or switch the controller to Program mode so you can edit and debug programs. If you choose to switch to Program mode, then the remote device cannot start programs until remote I/O has been enabled from the Run window.

Connecting while tasks are running from Remote Control or OP

If the controller is running tasks with the control device set to Remote or OP, you can connect the PC to the controller to monitor operation. For example, you can connect to a controller that is running tasks to temporarily monitor display output, tasks, and I/O, and then disconnect while tasks continue to run.

If the project on the PC is the same as on the controller, you will see the following message box when connection is established:
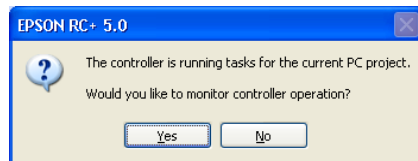
If the project on the PC is not that same as the project in the controller, you will see the following message box when connection is established:
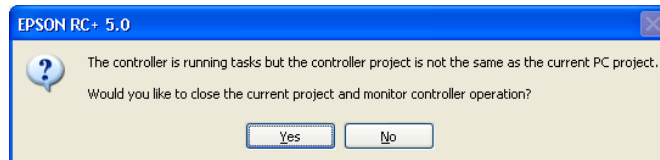
If you choose to monitor controller operation, the Run window will open if EPSON RC+ 5.0 is started in Program mode. If EPSON RC+ 5.0 is started in Auto mode, the Operator window will appear. From the Run window or Operator window, you can view display output from Print statements executing in the application. You can also use the Task Manager and I/O monitor.

When monitoring controller operation, the controller remains in Auto mode. You cannot stop tasks from EPSON RC+ 5.0, because the control device is not PC. If you want to switch the controller to Program mode, you must first stop all tasks from the current control device, then connect to the controller from EPSON RC+ 5.0 and choose to switch to Program mode (see the previous section *Connecting while control device is not PC and tasks are not running*).

Disconnecting while tasks are running

You can only disconnect from the controller with tasks running when the control device is set to Remote.

1. Stop communications with the controller by selecting Offline from the Connections dropdown list on the toolbar.

2. You may now disconnect the communications cable between the PC and the controller. Tasks will continue to run in the controller.

4.5 Writing your first program

After installing the Controller, robot, and EPSON RC+ 5.0 software on your PC, follow these instructions to create a simple application program so that you will become more familiar with the EPSON RC+ 5.0 development environment.

1. Connect your PC to the Controller

Install a USB cable between your PC and the Controller. Refer to the Communications with Controller section later in this chapter for more information.

2. Start EPSON RC+ 5.0 Simple Mode

Double-click the EPSON RC+ 5.0 Simple icon on the desktop.

3. Create a new project

(1) Select New from the Project Menu.
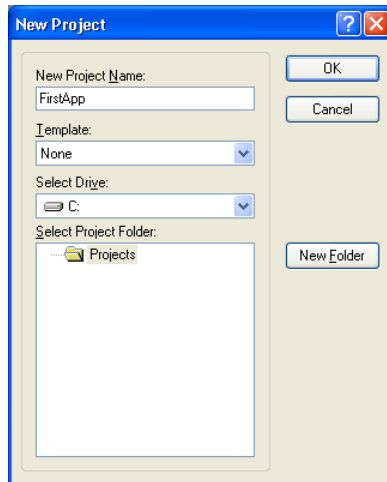
(2) Type in a name for a project, for example, FirstApp.

(3) Click **OK** to create the new project.

When the new project is created, a program called Program.prg is created. You will see a window open with the title Program.prg with a cursor flashing in the upper left corner. Now you are ready to start entering your first program.

4. Edit the program

Type in the following program lines in the Program.prg edit window.

```
Function main
    Print "This is my first program."
Fend
```
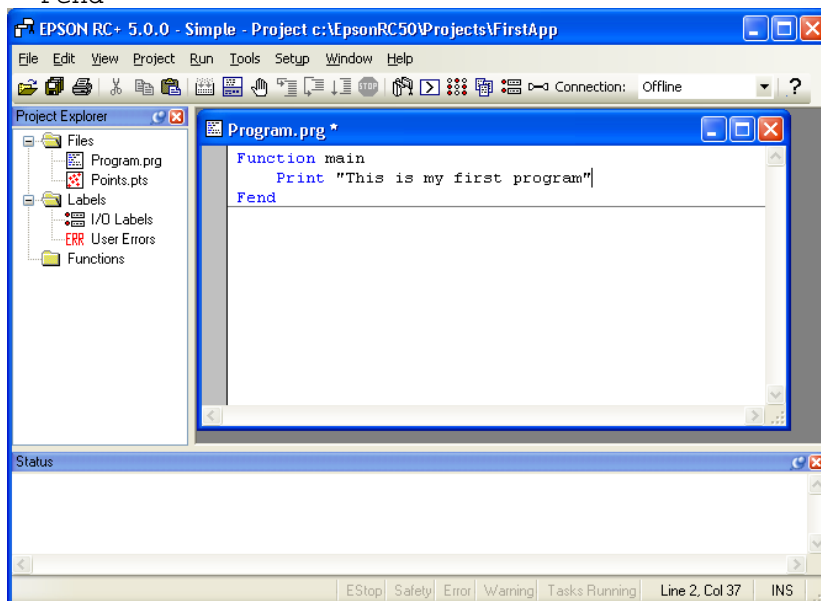

5. Run the program

(1) Press **F5** to run the program. (F5 is the hot key for the Run Window selection of the Run Menu). You will see the Status window located at the bottom of the main window showing the build operation status.

Run Window is displayed differently depending on the control device setting. Refer to *11.3 Control Device Configuration*.

(2) During project build, your program is compiled and linked. Then communications is established with the controller and project files are sent to the controller. If there are no errors during build, the Run window will appear.
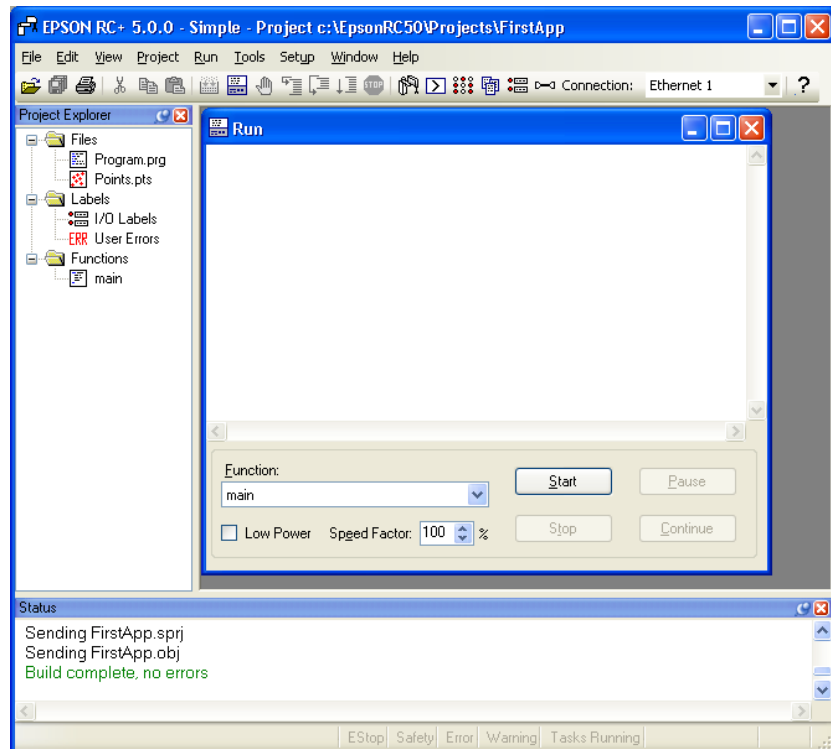
(3) Click the Start button on the Run window to run the program.

(4) You should see text similar to the following displayed in the Status window:

```
08:24:18 Task main started

08:24:18 All tasks stopped
```

On the Run window, you will see the output of the print statement.

Now let's teach some robot points and modify the program to move the robot.

6. Teach robot points

(1) Ensure that it is safe to operate the robot. Click the Robot Manager button on the toolbar. You will see the Robot Manager window with the **Control Panel** page displayed.
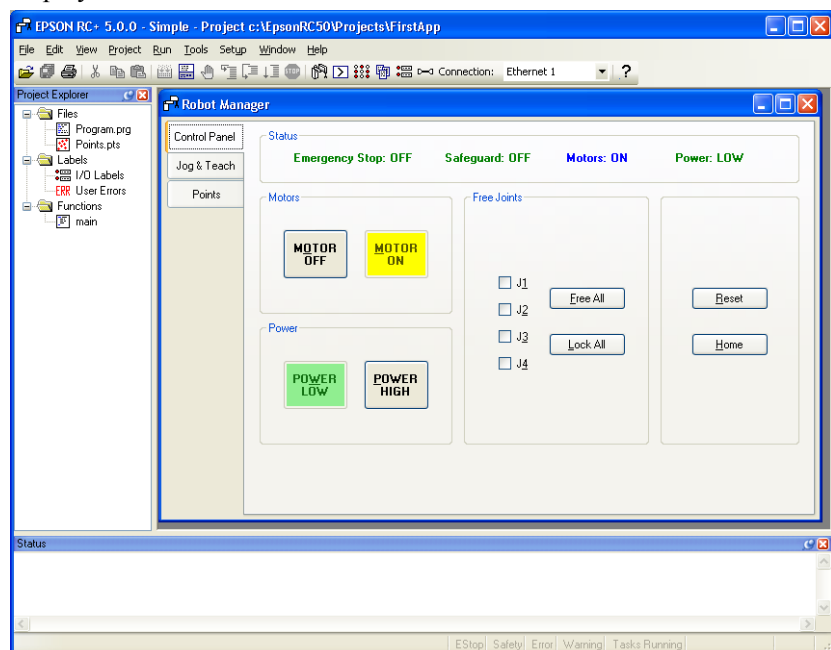
(2) Click on the **Motor On** button to turn on the robot motors. You will be prompted to confirm the operation.

(3) Answer Yes to continue.

(4) Click the Jog **& Teach** tab in the Robot Manager.
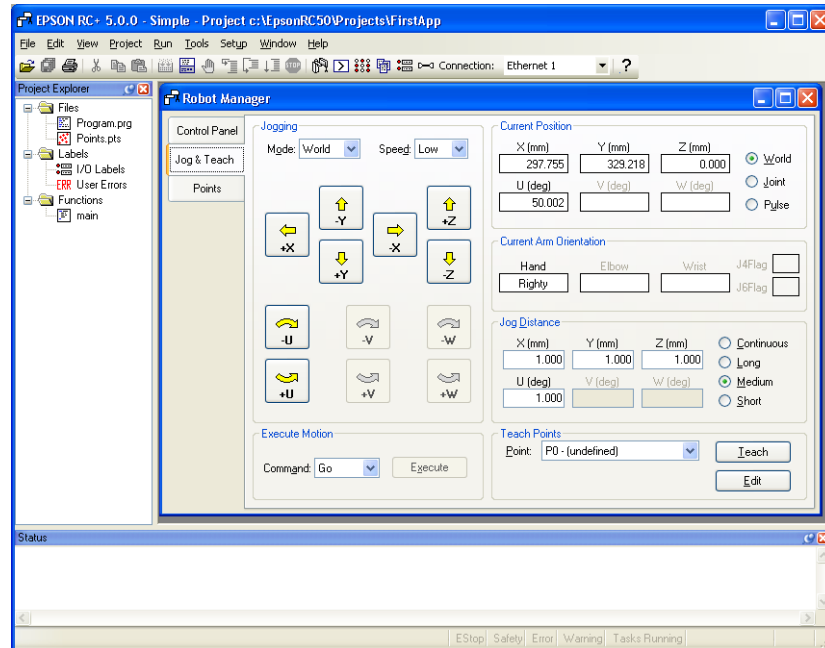
(5) Click the **Teach** button in the lower right corner to teach point P0. You will be prompted for a point label and description.

(6) Jog the robot by clicking the **+Y** jog button. Hold the button down to continue jogging. Let go when the robot is about half way out in the work envelope.

(7) Jog the robot down by clicking the **-Z** button.

(8) Now change the current point to P1 by selecting P1 in the Point dropdown list next to the Teach button.

(9) Click the **Teach** button. You will see a confirmation message to teach the point.

(10) Answer Yes.

(11) Click the **+X** button to jog the robot in the +X direction.

(12) Change the current point to P2 by selecting P2 in the Point dropdown list.

(13) Click the **Teach** button. You will see a confirmation message to teach the point.

(14) Answer Yes.

(15) Click the Save Project [icon] toolbar button to save the changes.

7. Modify the program to include robot motion commands

(1) Insert three new Go statements into the Program.prg program as shown below:

```
Function main
      Print "This is my first program."
      Go P1
      Go P2
      Go P0
Fend
```

(2) Run the program by pressing **F5** and then click on the **Start** button on the Run window.

(3) The robot should move to each of the points you taught.

8. Modify the program to change speed of robot motion commands

(1) Insert the Power, Speed, and Accel commands as shown in the program below:

```
Function main

    Print "This is my first program."

    Power High

    Speed 50

    Accel 50, 50

    Go P1

    Go P2

    Go P0

Fend
```

(2) Run the program by pressing **F5**

(3) Click on the **Start** button on the Run window.
The robot should go to each of the points you taught at 50% speed, acceleration, and deceleration. The Power High statement enables your program to run the robot at high (normal) power, which in turn allows the robot speed and acceleration to be increased.

9. Backup the project and system configuration

Even though this is only a sample project, we will backup the project and controller configuration. This is easy to do with EPSON RC+ 5.0. It is important that you keep regular backups of your applications on an external media such as a USB memory key.

Follow these steps to backup the project and system configuration:

(1) From the Project Menu, select Copy.

(2) Change the Destination Drive to another drive.

(3) Click OK. The project will be copied to the external media.

(4) From the Tools Menu, select Controller.

(5) Click on the **Backup Controller** button.

(6) Select the desired drive.

(7) Click OK. The system configuration will be backed up on the external media.

Now that you have written your first program, you should read the chapter *Building SPEL$^+$ Applications*.

5. The EPSON RC+ 5.0 GUI

The chapter contains information on the EPSON RC+ 5.0 GUI. After an overview and discussing online help and toolbars, the remainder of the chapter is divided into sections that follow the menu system.

Contents

- Overview

- Project Explorer Pane

- Status Window Pane

- Status Bar

- Online Help

- File Menu

- Edit Menu

- Project Menu

- Run Menu

- Tools Menu

- Setup Menu

- Window Menu

- Help Menu

5.1 GUI Overview

EPSON RC+ 5.0 is a multiple document interface (MDI) application. There is one main parent window and several child windows which can be opened simultaneously. The main window has a menu bar, tool bar, and status bar, as shown below.   In addition, there is a Project Explorer pane and Status Window pane.

EPSON RC+ 5.0 (Ver.5.4) User's Guide Rev.4
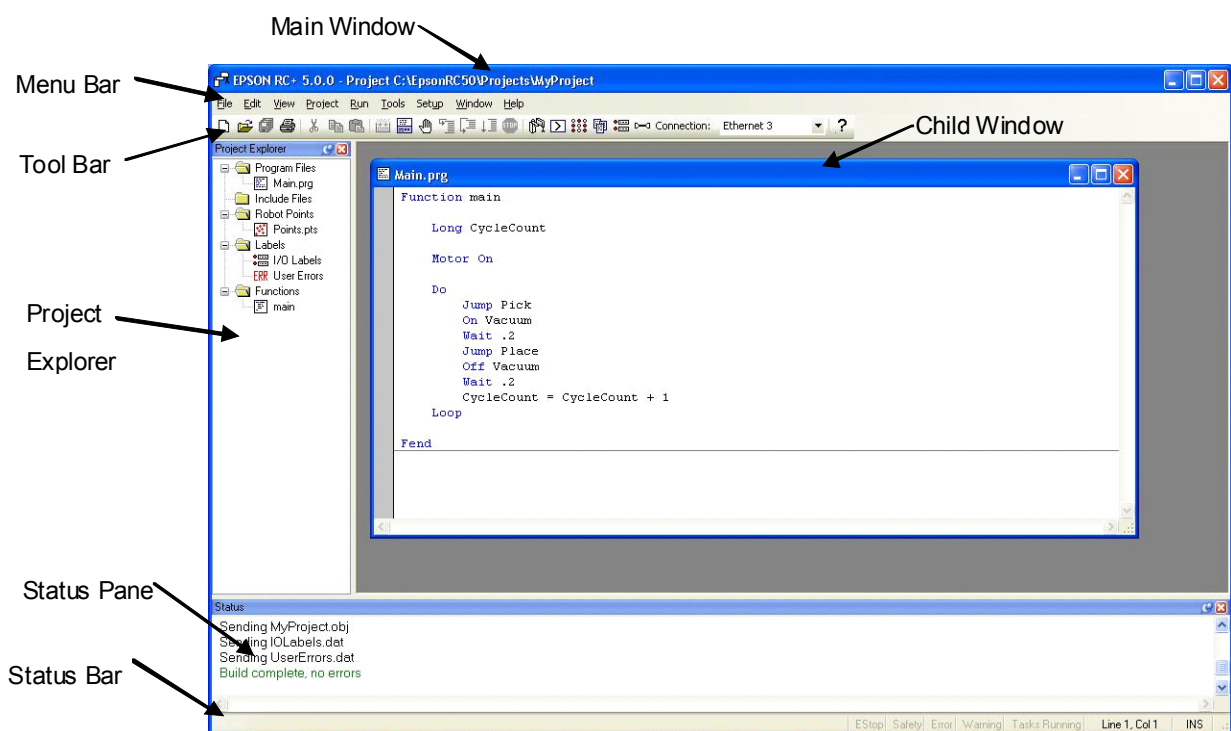
5.2 Project Explorer Pane

The Project Explorer pane enables you to quickly open any file in the current project or jump to any function. The project files and functions are organized in a sorted tree structure.

Open a file or jump to a function : Double-click on the item.

Hide the Project Explorer   : Click the X button on the bar above the pane.

Show the Project Explorer   : Select Project Explorer from the View Menu.

Resize the Project Explorer   : Move the mouse cursor over the right side of the pane, then drag the pane right or left to the desired width.

You can move the Project Explorer pane to either the left or right side of the main window. To move the pane, click down on the bar above the pane, then drag to either the left or right side of the main window and release the mouse button.

5.2.1 Context Menu

The Project Explorer Pane has a context menu for various operations for elements in the project tree. To access the context menu, right click on an item in the project tree.
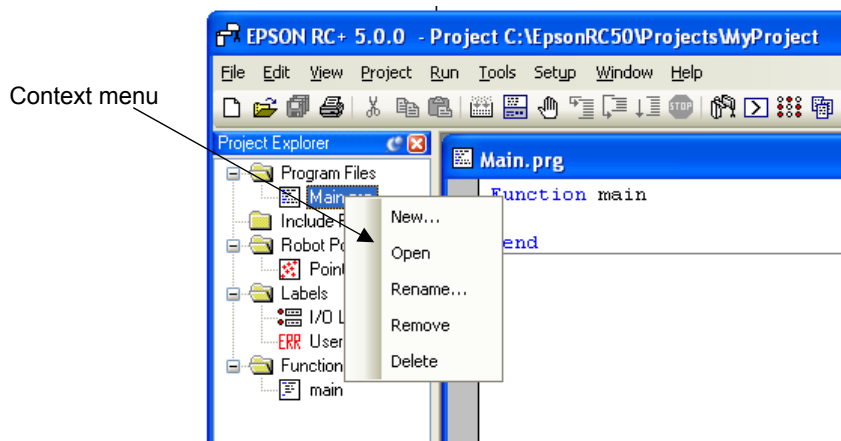
5.3 Status Window Pane

The status pane displays status messages, such as project build status, system errors and warnings, etc.

Hide the Status pane   : Click the X button on the bar above the pane.

Show the Status pane   : Select Status Window from the View Menu.

To resize the Status pane, move the mouse cursor over the top edge of the pane, then drag the top edge up or down.

The Status pane is always located at the bottom of the main window and cannot be moved.

NOTE

☞

If the Status pane is closed and an error message is displayed on the status pane, such as during project build, the Status pane will automatically be opened so that the error message can be seen.

5.4 Status Bar

The status bar located at the bottom of the main window displays the following:

Message area	Syntax error for the current line, system messages.
Operation Mode status	Indicates the controller operation mode. When EPSON RC+ 5.0 is connected to a controller, the controller operation status is shown: Program, Auto, and Teach.
Emergency Stop status	Indicates if emergency stop is active.
Safeguard status	Indicates if one or more safeguard circuits is open.
Error status	Indicates if the controller is in the error state. Position the mouse cursor over the Error status area to see the error message.
Warning status	Indicates if there is a warning. Position the mouse cursor over the Warning status area to see the warning message.
Tasks running status	Indicates that one or more tasks are running.
Current Line and Column	When a program editor window is active, the current line and column are displayed.
INS / OVR status	Indicates insert or overtype mode.

5.5 Online Help

EPSON RC+ 5.0 has an extensive context sensitive help system.

There are several methods to get help.

- Select **Contents** from the Help Menu to browse help topics.

- Select **Index** from the Help Menu to enter the name of a specific topic.

- Select **Search** from the Help Menu to search for a specific topic.

- When editing programs, press F1 with the caret in the keyword of interest.

When a dialog is open, press F1 or click the Help button. For dialogs, the Help button is located in the window title bar on the right side and is shown as a question mark icon as shown below.

Help button for a dialog
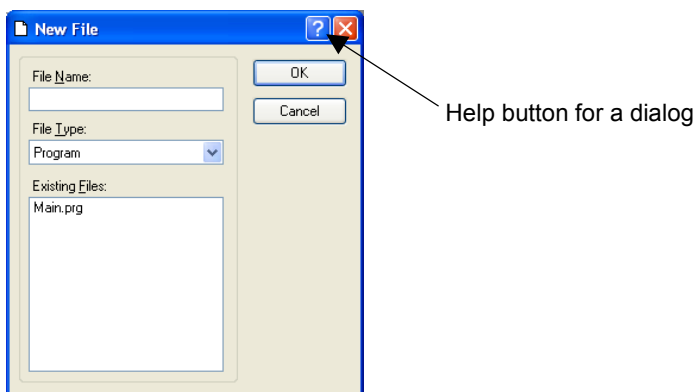
For MDI child windows, the Help button is located on the main toolbar and is also shown as a question mark icon as shown below.

Help button for the
active MDI child window
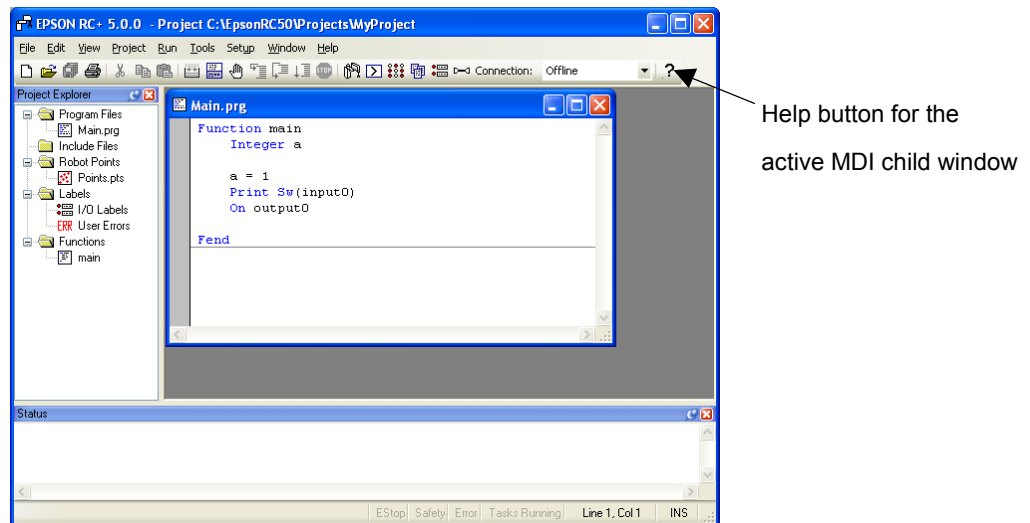
5.6 File Menu

The EPSON RC+ 5.0 File Menu includes commands for managing and printing files in the current project.

5.6.1 New Command (File Menu)

The New command is used to add new files to the current project. When the New command is selected, the New File dialog is opened.

Shortcuts

Toolbar:
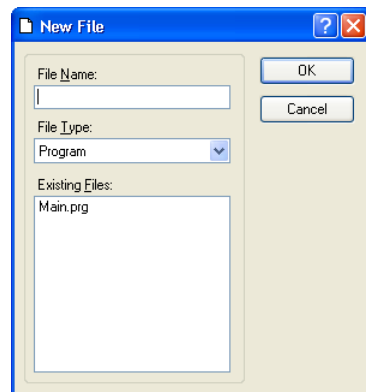
Item	Description
File Name	Enter a name for the new file in this box. If you supply a valid file extension, the File Type selection will change to match the extension.
File Type	Use this dropdown list to select Program, Include or Point file.
Existing Files	This section shows the files for the selected type currently in the Project directory.
OK	Click OK when you are ready to create the new file.
Cancel	Cancels the operation.

5.6.2 Open Command (File Menu)

Open one or more files in the current project for editing. You can open program files, include files, or point files.

If there is a file in the current project directory (as shown in the Edit Project dialog box) and the file is not in the current project, you will not be able to open the file. You must add the file to the project before you can open it. This also applies to include files and point files.

Shortcuts

Toolbar:

Keys: Ctrl+O
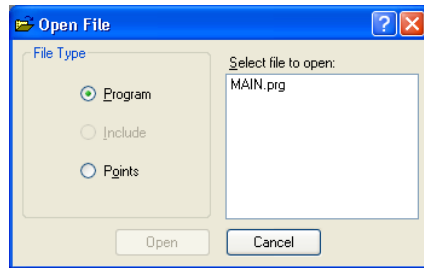
Item	Description
Program	Select this radio button to show a list of program files in the current project.
Include	Select this radio button to show a list of include files in the current project.
Points	Select this radio button to show a list of point files in the current project.
Select file to open	Click on the file name you want to open. You can select more than one file by using the Ctrl key or Shift key. The Ctrl key allows you to select or deselect any file. The Shift key allows you to select a group of files.
Open	Opens the selected file(s).
Cancel	Cancels the open operation.

TIP

You can also double click on a file name in the **Select file to open** list box to open the file without having to choose the **OK** button.

5.6.3 Close Command (File Menu)

Close the currently active window.

Shortcuts

Keys: Ctrl + D, Alt + F4

Any of the windows can be closed with this command: Programs, Include files, Points, Command Window, Run, Status, I/O Label Editor.

TIP

You can also close a window or dialog box by double clicking on the control box button located in the upper left corner of the window or dialog box.

5.6.4 Save Command (File Menu)

The Save command writes the current file to disk. The current file can be a program file, include file, points file, I/O label editor, etc. This command is disabled if the current file does not need to be saved.

Shortcuts

Keys: Ctrl + S

5.6.5 Save As Command (File Menu)

Save the program, include file, or point file in the currently active window with a new file name. The original file will be left in the project directory unchanged. The new name will be used throughout the current project in place of the old name.

If you use **Save As** on an include file, you must rename the file in each of your #include statements that refer to it.
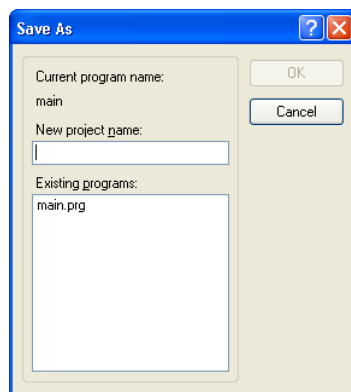
5.6.6 Save All Command (File Menu)

This command saves all files in the current project and any changes made to the Simulator. This menu selection is grayed out if nothing needs to be saved.

Shortcuts

Toolbar:

TIP
☞

It's a good idea to save frequently while you are editing project and Simulator files. Just click the disk button on the toolbar to save all of your files.

5.6.7 Restore Command (File Menu)

Restores the currently active program, include file, I/O labels, user errors, or points from disk.

Use this command to change a document to the state it was in since last saved.

You will be prompted to confirm this operation.

5.6.8 Rename Command (File Menu)

Use Rename to change the name of the program, include file, or point file you are currently editing.
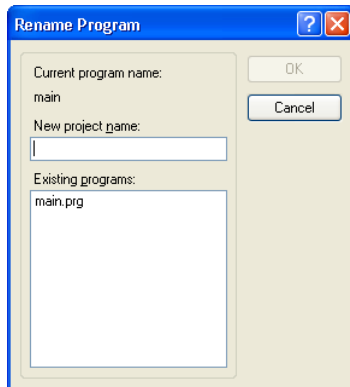
To rename a file

1. Make the file you want to rename the current active window by clicking anywhere on the program window, or selecting it from the list in the Window Menu, or by opening the file with the Open command from the File Menu.

2. Select Rename from the File Menu.

3. Type in a new name for the file and click **OK**. Refer to the existing files list to see what other programs are in the current project directory. The new file name you enter cannot be the same as any other file in the project directory. You will get an error message if you enter a new name that is already being used.

If you use **Rename** on an include file, you must rename the file in each of your #include statements that refer to it.

5.6.9 Delete Command (File Menu)

This command allows you to delete a file in the current project directory. You can delete program files, include files, and point files. The file does not have to be in the project make list to delete it.
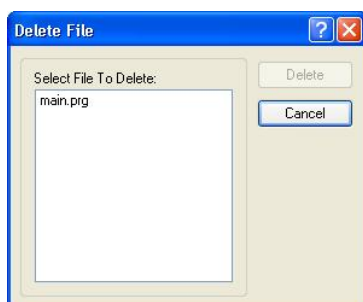
Item	Description
Select file to delete	Click on the file name you want to delete. This file list displays all .PRG, .INC, and .PTS files in the current project directory.
Delete	Delete the selected file. You will be prompted with a confirmation message before the file is deleted. If the file is currently open, it will be closed and removed from the current project before it is deleted from disk.
Cancel	Cancels the delete operation.

5.6.10 Import Command (File Menu)

Import a file into the current project. Use this command to import programs, points, labels, etc. from other EPSON RC+ 5.0 projects.

- Program file names for importing must have a .PRG extension.

- Include file names for importing must have a .INC extension.

- Point file names for importing must have a .PTS extension.

- I/O labels must have the file name IOLABEL.DAT

- User errors must have the file name USERERRORS.DAT.

- Macros must have the .MAC extension.
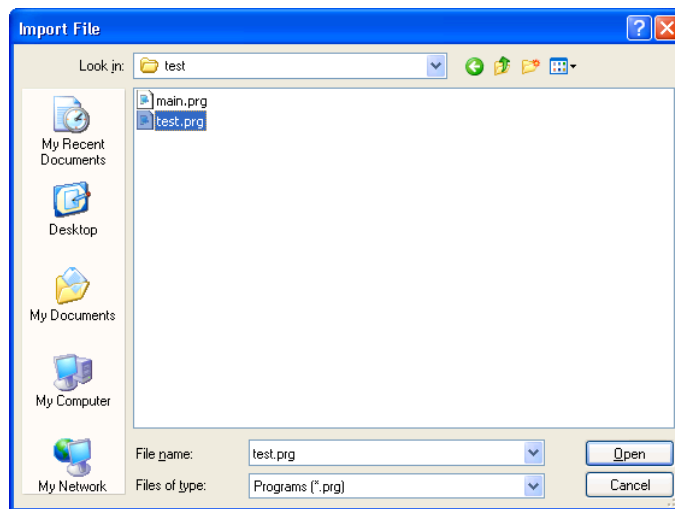
To import a file

1. Select the file type from the **File Type** list box.

2. Navigate to the file you want to import.

3. Click **Open** to continue. If a file is being overwritten, you will be prompted to continue. The file will then be copied to the current project's directory.

NOTE

☞

If you need to import files from previous versions of EPSON RC+ or from SPEL for Windows 2.0, you must first import the project using Project | Import, which converts the point files and label files into EPSON RC+ 5.0 formats. Then you can use File Import to import the desired files.

5.6.11 Print Command (File Menu)

This command opens the Print dialog box. From this dialog, you can print one or more program, include, points, or label files in the current project. You can also print out a project summary.

Each document is printed with a header that includes the file name, date and time, and page number.

Shortcuts

Toolbar: 🖨

Keys: Ctrl+P
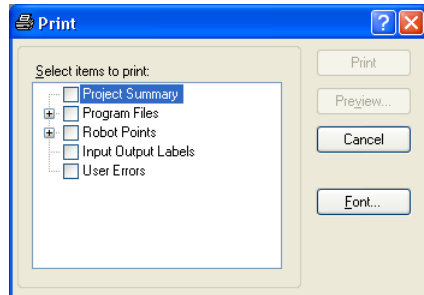
Item	Description
Select items to print	Check the items in the tree that you would like to print out.
Project Summary	Select this check box to print a summary of the programs and points used in the current project.
Program Files	Select this check box to print all program files, or click on the + button to view all program files and check the ones you want printed.
Include Files	Select this check box to print all include files, or click on the + button to view all include files and check the ones you want printed. This check box is not shown if there are no include files in the current project.
Robot Points	Select this check box to print all point files, or click on the + button to view all point files and check the ones you want printed.
Input Output Labels	Select this check box to print a listing of the all of the I/O labels used in the project.
User Errors	Prints a listing of all user errors for the current project. If either the label or message is non-blank, then the error definition will be printed.
Print	Print the selected files. This button will be dimmed if nothing is selected to be printed.
Preview	Preview the selected files before printing. This button will be dimmed if nothing is selected to be printed.
Font...	Opens a dialog for selecting the printer font. The selected font is saved for subsequent printing.
Cancel	Closes the dialog box without printing anything.

5.6.12 Exit Command (File Menu)

Exits from EPSON RC+ 5.0.

If you are running a program from the Run Window and the control device is PC, you will see a message that a program is running and you will not be allowed to exit. You must stop all tasks first before you can exit.

If there are any open program or point files that have not been saved, you will be prompted to save these files with Yes, No, or Cancel.

If you select **Yes**, then all unsaved files in the project will be saved.

If you select **No**, then the program will exit without saving the files.

If you select **Cancel**, then the exit command will be aborted.

5.7 Edit Menu

TIP

☞

The EPSON RC+ 5.0 Edit Menu includes commands for editing files.

You can also access the Edit Menu by right-clicking anywhere in a program editor window.

5.7.1 Undo Command (Edit Menu)

Undo the changes to the currently active program since it was open.

Shortcuts

Keys: Ctrl+Z

5.7.2 Redo Command (Edit Menu)

Redo the previous undo.

Shortcuts

Keys: Ctrl+Y

5.7.3 Cut Command (Edit Menu)

Copies the current selection into the Clipboard and then deletes the selection.

Shortcuts

Toolbar: ✂

Keys: Ctrl+X

5.7.4 Copy Command (Edit Menu)

Copies the current selection into the Clipboard.

Shortcuts

Toolbar: 🗐

Keys: Ctrl+C

5.7.5 Paste Command (Edit Menu)

Puts the contents of the Clipboard into the currently active document starting at the insertion point.

Shortcuts

Toolbar:

Keys: Ctrl+V

5.7.6 Find Command (Edit Menu)

Find a text string in the current program or all programs in the project. The first time you execute this command, the dialog box will be centered over the main window. If you reposition it, then the next time Find is executed, the dialog will appear where you last positioned it.

Shortcuts

Keys: Ctrl + F

Item	Description
Find What	Type the text you want to search for. If any text was selected when you execute the Find command, it will be displayed here. If no text was selected, then the text from the last Find will be displayed. You are limited to one line of text. If you select more than one line before executing Find, then only the first line will be displayed here. Tabs are represented by "\t".
Current File	Search only in the current program file.
All Files	Search all files in the project.
Whole Word	Search for the full word by itself and not as part of another word. This check box is not available if the search string contains more than one word.
Match Case	Text must also match lower and upper case in order to be found.
Find	Start the search. If the text is found in a file that is not open, then it will be opened.
Close	Closes the dialog box.

5.7.7 Find Next Command (Edit Menu)

Find the next occurrence of the search text specified in the last Find command.

Shortcuts

Keys: F3

5.7.8 Replace Command (Edit Menu)

Search for a text string and replace it with new text. The first time you execute this command, the dialog box will be centered over the main window. If you reposition it, then the next time Replace is executed, the dialog will appear where you last positioned it.

Shortcuts

Keys: Ctrl+R
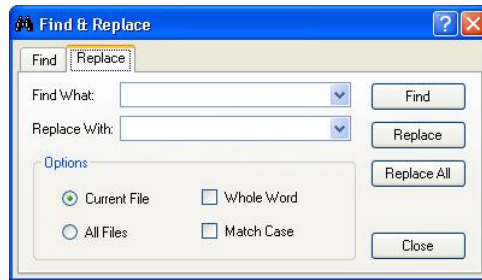
Item	Description
Find What	Type the text you want to search for. If any text was selected when you execute the Replace command, it will be displayed here. If no text was selected, then the text from the last Find will be displayed.
Replace With	Enter the replacement text here. The replacement text can be empty.
Current File	Search only in the current program file.
All Files	Search all files in the project.
Whole Word	Search for the full word by itself and not as part of another word. This check box is not available if the search string contains more than one word.
Match Case	Text must also match lower and upper case in order to be found.
Find	Find the next occurrence.
Replace	If already found, replace the current find, otherwise search for the next occurrence.
Replace All	Replace all occurrences in the current project.
Close	Closes the dialog box.

5.7.9 Select All Command (Edit Menu)

Select the entire file. You can then execute Cut or Copy.

Shortcuts

Keys: Ctrl+A

5.7.10 Indent Command (Edit Menu)

Move the selected text block one tab to the right.

Shortcuts

Keys: Tab

5.7.11 Outdent Command (Edit Menu)

Move the selected text block one tab to the left.

Shortcuts

Keys: Shift+Tab

5.7.12 Comment Block Command (Edit Menu)

Comments out the selected block of lines by adding the comment character to the beginning of each line. To use, select one or more lines to be commented. Then select Comment Block from the Edit Menu, or right click and select Comment Block from the pop-up menu. A comment character will be added to the beginning of each of the selected lines.

5.7.13 Uncomment Block Command (Edit Menu)

Removes leading comment character from the selected block of lines. To use, select one or more lines to be uncommented. Then select Uncomment Block from the Edit Menu, or right click and select Uncomment Block from the pop-up menu. The first comment character from each of the selected lines will be removed.

5.7.14 Go To Definition Command (Edit Menu)

Opens the window and sets the line where a function, variable, macro, point label, I/O label, or user error label is defined. To use, click on an identifier in a program window, then select Go To Definition from the Edit Menu or toolbar. You can also right click on the identifier and select Go To Definition from the pop-up menu.

If the identifier is a function name or variable, the program window where it is declared will be opened and the current line number will be set.

If the identifier is an point label, the point file will be opened and the current row for the label will be displayed.

If the identifier is an I/O label, the I/O label editor will be opened and the current row for the label will be displayed.

If the identifier is a user error label, the User Error editor will be opened and the current row for the label will be displayed.

5.8 View Menu

The EPSON RC+ 5.0 View Menu includes commands for opening the Project Explorer and Status panes. In addition, there is a command for viewing the system history.

5.8.1 Project Explorer Command (View Menu)

If you have closed the Project Explorer pane, you can open it by using this command. For details, see the section *Project Explorer Pane* in this chapter.

5.8.2 Status Window Command (View Menu)

If you have closed the Status Window pane, you can open it by using this command. For details, see the section *Status Window Pane* in this chapter.

5.8.3 System History Command (View Menu)

This command opens the System History window. This window shows events, errors, and warnings that have been logged in the current controller's system history.

The data can be sorted by clicking on any column header. To sort multiple columns, hold down the shift key and click on multiple columns headers.
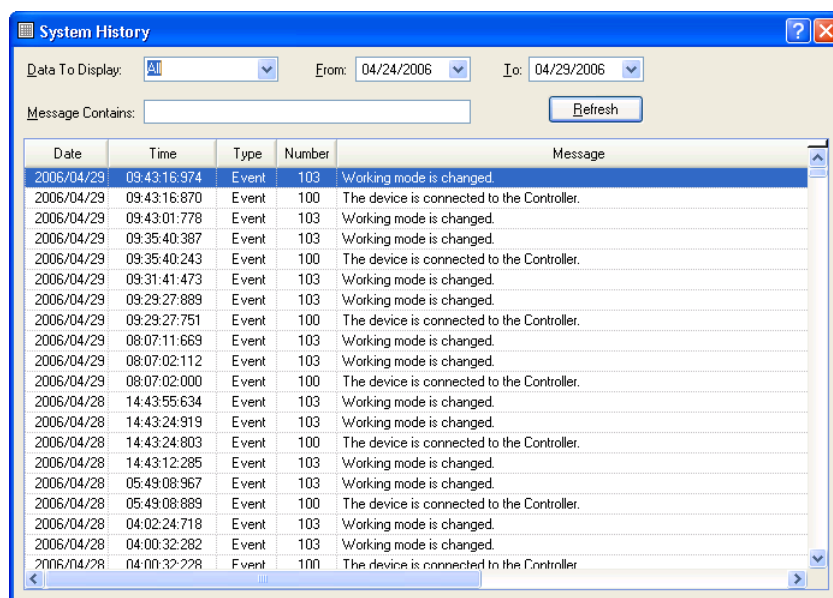
Item	Description
Data To Display	Select which data you would like to view. Choices are All, Events, Errors, and Warnings.
From / To	Select the dates you want to view data from. When the window is first opened, these are automatically set to the first and last dates in the history data.
Message Contains	Type in text to be found in the error message. After typing in the text, click the Refresh button.
Refresh	Click this button to reload the data from the controller.
Type **Event**	Information for operation and mode change.
Error	Program can be executed continuously, however, needs countermeasure.
Warning	Error occurred in the program or the Robot.
Number	For details of the number, refer to *SPEL+ Error Message* in the *SPEL+ Language Reference*.
Message	
Function and Line number	Function name and the line number are displayed when error occurred while executing a program.
Robot and axis number	Robot and the axis number are displayed when Robot error occurred.
Task number	Task number of the task with error is displayed when error occurred while executing the program. "0" is displayed for others.
Additional information 1 and 2	More details are displayed for some errors. For details, refer to *SPEL+ Error Message* in the *SPEL+ Language Reference*.

5.9 Project Menu

The EPSON RC+ 5.0 Project Menu includes commands for managing and building projects.

5.9.1 New Command (Project Menu)

The New command is used to create a new EPSON RC+ 5.0 project. Projects can be on any disk drive on the system. They are stored in the \EpsonRC50\Projects directory on the selected drive. Subdirectories can also be created.

Item	Description
Project Name	Type in a new name for the project. The name can include alphanumeric characters along with underscores.
Template	Select a project template. The new project will be a copy of the template project.
Select Drive	Select the desired disk drive for the new project.
Select Project Folder	This is a list of folders and projects on the selected drive. If you click on a name in this list, it will be displayed in the New Project Name text box. You can then edit the name, or you can create a new project with the same name as one that has already been created. In the later case, you will be prompted to overwrite the old project if it is in the same folder.
New Folder	Click this button to create a new folder in the currently selected folder.
OK	Click OK to create the new project.
Cancel	Click Cancel to abort creating a new project.

5.9.2 Open Command (Project Menu)

Use this command to open an EPSON RC+ 5.0 project. When the project is opened, the previous project is closed. You will be prompted to save changes.
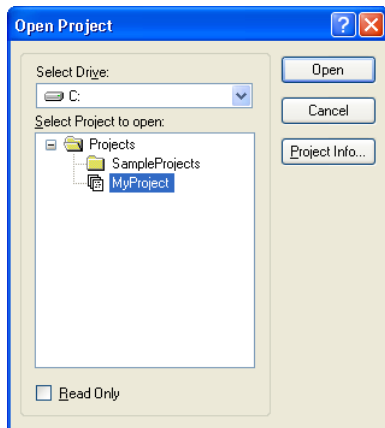
Item	Description
Select Drive	Select the desired disk drive for the project you want to open.
Select Project to Open	Select a project name from the list box. To open a folder, double click on the folder or click the + box located to the left of the folder.
Open	Open the selected project.
Cancel	Cancel the operation.
Project Info	Displays general project properties for the selected project. To view project information, first select a project in the list, then click the **Project Info** button.

NOTE
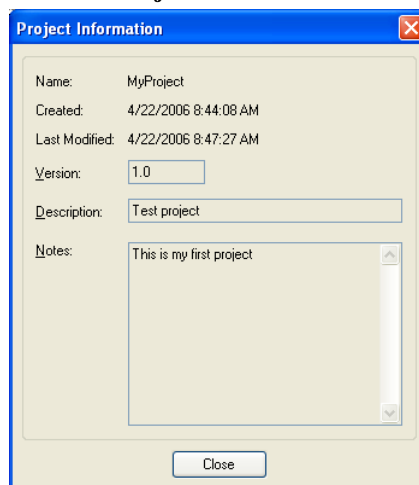
☞ Project information for a project can be changed by selecting Properties from the Project Menu after opening the project.

5.9.3 Recent Projects Submenu (Project Menu)

The Recent Projects submenu contains up to eight of the most recently used projects. When you select a project in the menu, the current project is closed and the selected project is opened the same as if you used the Open command from the Project Menu.
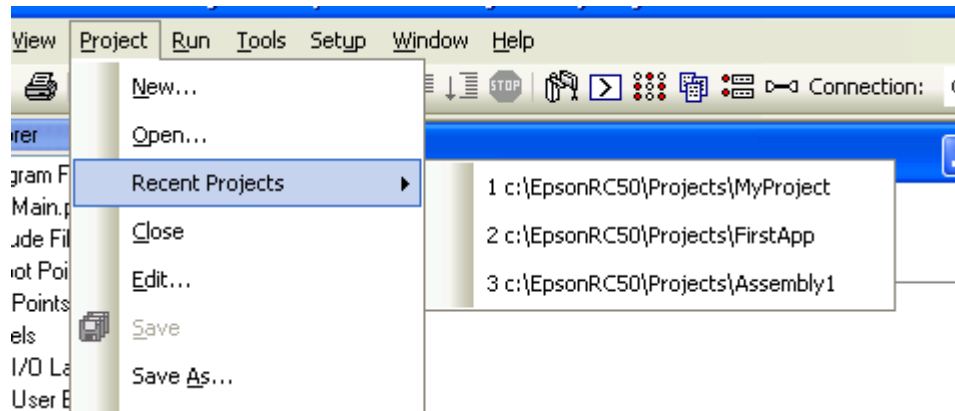
5.9.4 Close Command (Project Menu)

Use the Close command to close the current project. Several menu and toolbar commands will be disabled after the project is closed.

5.9.5 Edit Command (Project Menu)

The Edit command is used to define which program source files and point files are to be used in the current project. It is divided into two main sections. The left section contains a list of files in the current project disk directory. You can select which files to view from the **File Type** list box. The right section contains a project make tree that includes program files, include files, and robot point files.
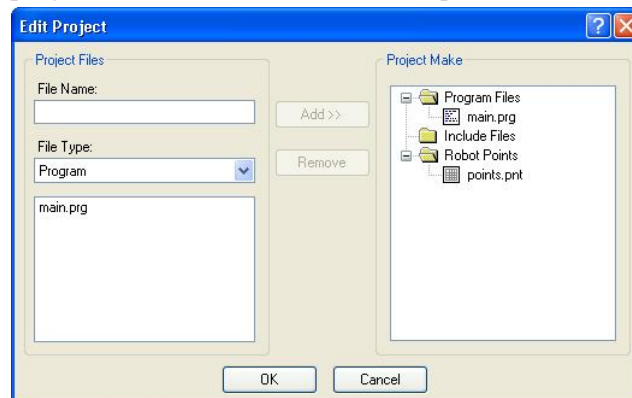
The files shown in the file list are in the current project disk directory. Before you can use a file in the project, you must put it into the project make tree using the Add button.

To create a new program

1. Type the name of the program in the **File Name** text box in the Program Files section.

2. Click the **Add >>** button next to the program files list. You will get a message asking if it's okay to create the new file. Answer **Yes** to create the file and put it in Program Files folder in the project make tree.

To add an existing program source file

1. Select the program name you want to add by clicking on a program name in the program files list box.

2. Click the **Add >>** button

   Or

   Double click on the program name in the file list box.

The file will be added to the list of programs.

To create a new include file

1. Type the name of the include file in the **File Name** text box. The name of the include file can also be the same name as a program.

2. Click the **Add >>** button. You will get a message asking if it's okay to create the new file. Click **Yes** to create the file and put it in the Include Files folder in the project make tree.

To add an existing include file to the project

1. Select the include file name you want to add to the project by clicking on an include file name in the file list box.

2. Click the Add >> button.

   Or

   Double click on the include file name in the file list box.

The file will be added to the list of include files for project make.

To add a new robot point file

1. Enter the name of the point file you want to create. Use the PTS extension.

2. Click the **Add >>** button. You will get a message asking if it's okay to create the new file. Click **Yes** to create the file and put it in the Robot Points folder in the project make tree

To add an existing robot point file

1. Select Points from the **File Type** list box.

2. Select the point file you want to add from the list.

3. Click the **Add >>** button.

To remove a file from the project

1. Select the file you want to remove in the project make tree.

2. Click the **Remove** button to remove the file from project make. The file is not deleted from the project folder, so you will still see the file in the file list.

5.9.6 Save Command (Project Menu)

This command saves all currently open files in the current project that need to be saved. This menu selection is grayed out if nothing needs to be saved.

NOTE
☞

This command saves project files. It does not save changes made in the Simulator. Use the File | Save All command to save all project files and Simulator changes.

5.9.7 Save As Command (Project Menu)

Saves all files in the current project to a new drive and/or project name. The current project will be preserved.

Item	Description
New Project Name	Type in a new name for the project. The name can include alphanumeric characters along with underscores. The maximum number of characters is 24. You can use the same name as the current project if you select a drive and folder that is not the same as the current project's drive and folder.
New Project Drive	Drive for the new project location.
Select Project Folder	Click on the desired folder for the project.
New Folder	Click this button to create a new folder under the currently selected folder.
OK	Save the project using the new name and location.
Cancel	Cancel the operation.

5.9.8 Rename Command (Project Menu)

This command renames the current project. The project directory and all associated project files are also renamed.
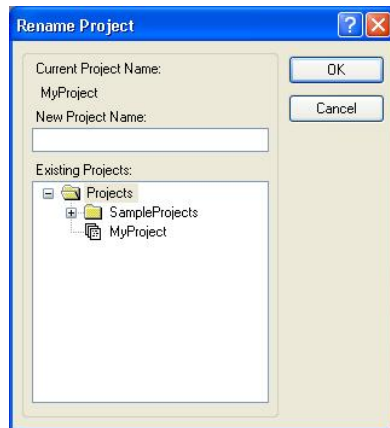
Item	Description
New Project Name	Type in a new name for the project. The name can include alphanumeric characters along with underscores.
Existing Project	This list box shows other projects on the selected drive. The new name you choose cannot be one of the names in this list.
OK	Renames the project.
Cancel	Cancels the operation.

After a project is renamed, the next time you build, the entire project is rebuilt and when connection is established with the controller, you will be prompted to overwrite the project in the controller (since it has a different name).

5.9.9 Import Command (Project Menu)

The Project Menu Import Command uses a wizard to import projects from a PC, the current controller, or a controller status folder.

When a project is imported, the files from the source are copied to a new project folder, so the original project is not changed.

NOTE
☞

If the project to be imported is an EPSON RC+ 3.x / 4.x project or a SPEL for Windows 2.0 project, the files are converted to EPSON RC+ 5.0 format.

The sections below have instructions for importing a project from each type of source location.

Importing a PC project

Follow these steps to import a project from a PC:

1. Select Import from the Project Menu to open the Import Project dialog.
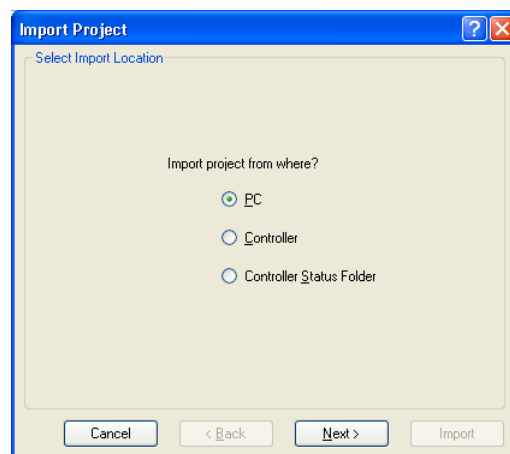
2. Select **PC** and click **Next**.
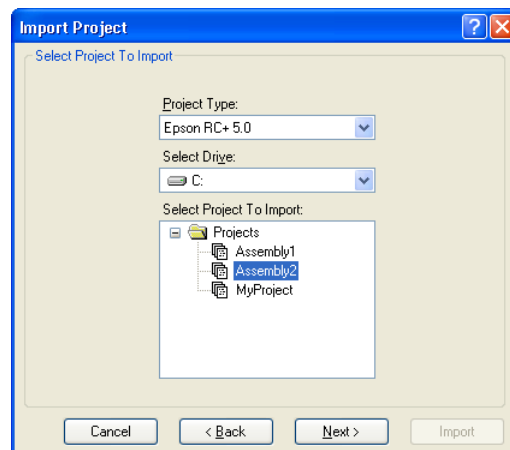
3. Select the project type. You can select from the following:

- Epson RC+ 5.0

- Epson RC+ 3.x / 4.x

- SPEL for Windows 2.0

NOTE

☞ When a project for EPSON RC+ 3.* / 4.*or SPEL for Windows 2.0 is imported, the project is automatically converted to project for EPSON RC+ 5.0 format.
For details, refer to *Appendix B: Automatic Processing of Project Import*.

4. Select the drive. After you select the project type and drive, the project list will be updated to show the projects available for import. Select the project to import in the list and click **Next**.
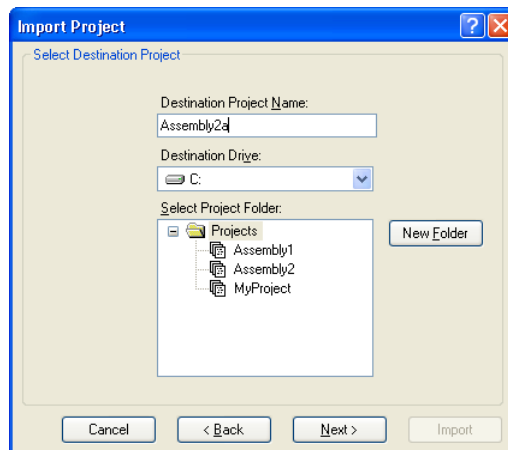
5. The destination project name is set to the name of the imported project. You can modify the destination project name if desired. Select the destination drive and folder, then click **Next**.
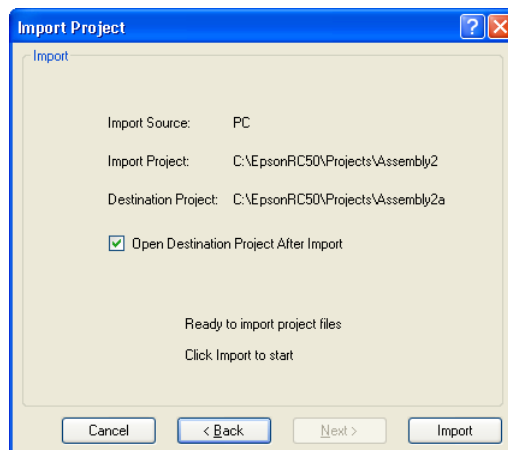
6. Verify the import source, import project, and destination project. Check **Open Destination Project After Import** if you want the project to open after import.

7. Click the **Import** button. If the destination project already exists, you will be asked if you want to overwrite it.

Importing a Controller project

NOTE
☞

For projects using Vision Guide, the calibration data and model files are not imported from the Controller because they are not stored there. After import, you will need to re-teach the vision models and perform the calibrations.

Follow these steps to import a project from a controller:

1. Select Import from the Project Menu to open the Import Project dialog.
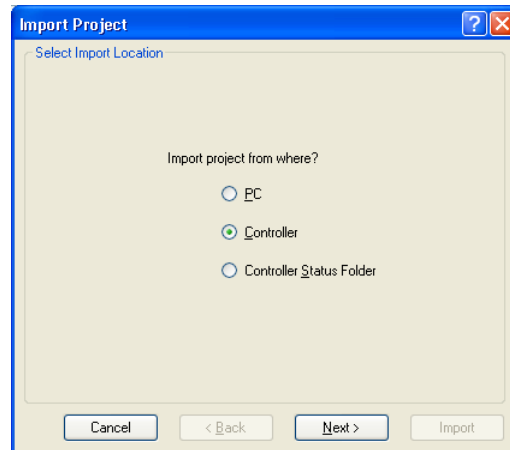
2. Select **Controller** and click **Next**. Communications with the controller will be established if EPSON RC+ 5.0 is not already connected.
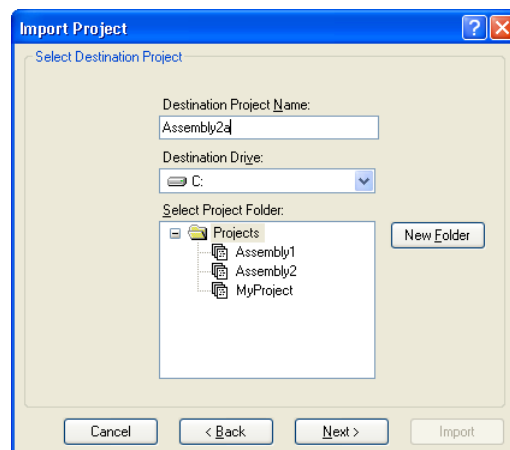
3. The destination project name is set to the name of the current project in the controller. You can modify the destination project name if desired. Select the destination drive and folder, then click **Next**.
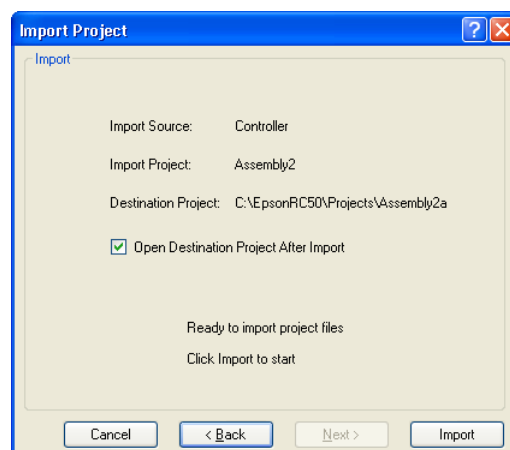
4. Verify the import source, import project, and destination project. Check **Open Destination Project After Import** if you want the project to open after import.

5. Click the **Import** button. If the destination project already exists, you will be asked if you want to overwrite it.

Importing a Controller Status project

NOTE

For projects using Vision Guide, the calibration data and model files are not imported from the Controller Status folder because they are not stored there. After import, you will need to re-teach the vision models and perform the calibrations.

Follow these steps to import a project from a controller status folder:

1. Select Import from the Project Menu to open the Import Project dialog.
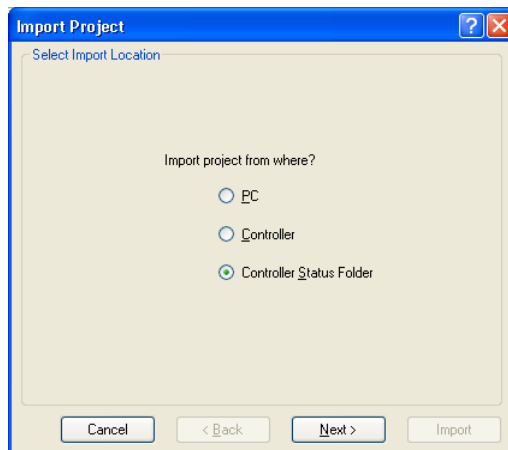
2. Select **Controller Status Folder** and click **Next**.
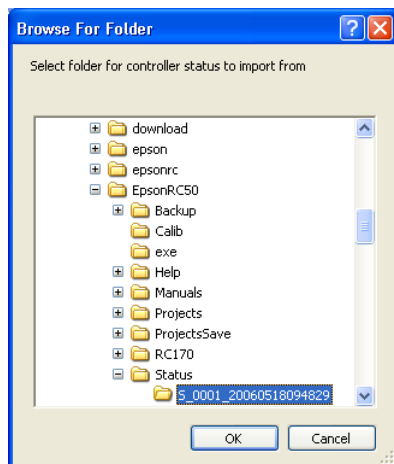
3. Select a controller status folder and click **OK**.
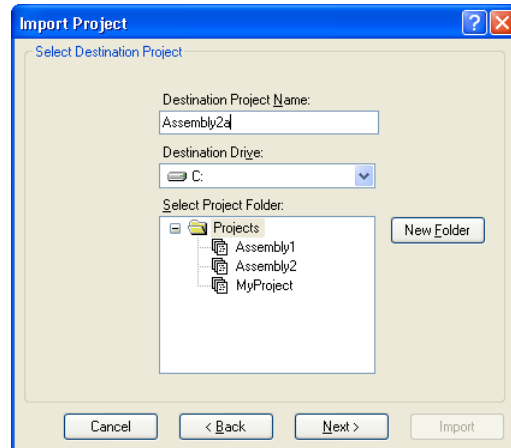
4. The destination project name is set to the project found in the controller status folder.
    You can modify the destination project name if desired. Select the destination drive
    and folder, then click **Next**.
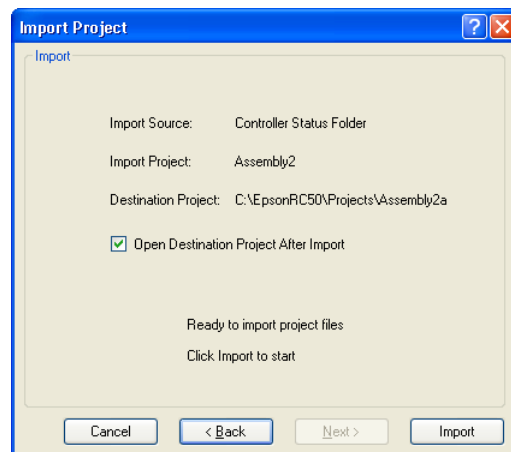
5. Verify the import source, import project, and destination project. Check **Open
    Destination Project After Import** if you want the project to open after import.

6. Click the **Import** button. If the destination project already exists, you will be asked if
    you want to overwrite it.

5.9.10 Copy Command (Project Menu)

The Copy command copies all files in the current project to a specified drive, folder, and project name. You can use the current project name for the destination name if you select a new drive or folder. You can also specify a new name for the destination project.

You should use the Copy command to make backup copies of your project on a regular basis.
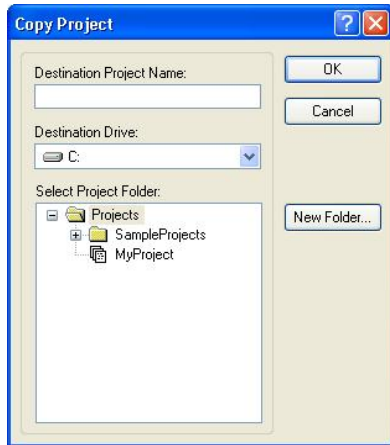
Item	Description
Destination Project Name	Type in a name for the new copy of the project. The name can include alphanumeric characters along with underscores. The maximum number of characters is 24. You can use the same name as the current project if you select a drive and folder that is not the same as the current project's drive and folder.
Destination Drive	Drive for the project copy.
OK	Perform the copy process.
Cancel	Cancel the operation.

5.9.11 Delete Command (Project Menu)

This command deletes an entire project from a PC disk. All files in the project directory will be destroyed.
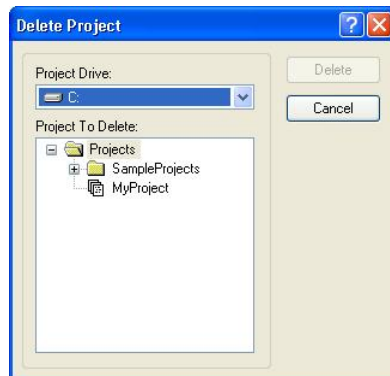
Item	Description
Project Drive	Select drive for the project to delete.
Project To Delete	Select a project to delete from the list.
Delete	Delete the project. You will be prompted to confirm the operation.
Cancel	Cancel the operation.

5.9.12 Build Command (Project Menu)

This command builds the current project so that it can be executed. The Build command does the minimum amount of work required to bring the project up to date in the robot controller. For example, if a change was made to one program file in the project, then Build will compile the changed program, link it with the remaining object files (if they exist), and send the new files to the controller.

During the build process, the status pane displays each step of the build. If there are any errors, they will be displayed on the status pane.

Shortcuts

Toolbar:

Keys: Ctrl+B

5.9.13 Rebuild Command (Project Menu)

Rebuilds the entire current project. All program files are re-compiled, linked, and sent to the controller. All point files in the project are sent to the controller.

Shortcuts

Keys: Ctrl + Shift + B

5.9.14 Synchronize Command (Project Menu)

The Synchronize command is used to update files on the PC and controller. This command is not available if the current project has not been built successfully.

This dialog will also appear automatically when you are connecting to a controller that has the same project as the current PC project, but the files on the controller are newer. For example, if the teach pendant is used to teach a point and save a point file in the project, then the point file will be newer than the corresponding file in the PC project, so it must be updated on the PC after it is connected to the controller.

NOTE
☞
You can also transfer files that are not in the current project between the PC and controller, such as curve files (.CVT) and point files (.PTS).

If there are any project files to be transferred, the status for these files is "Pending". For non-project files, you must select the direction of transfer.

Click the Start button to transfer the files in the direction specified in the Transfer column. After a successful transfer, the status for a file will change to "Complete". If an error occurs, the status will change to "Error".
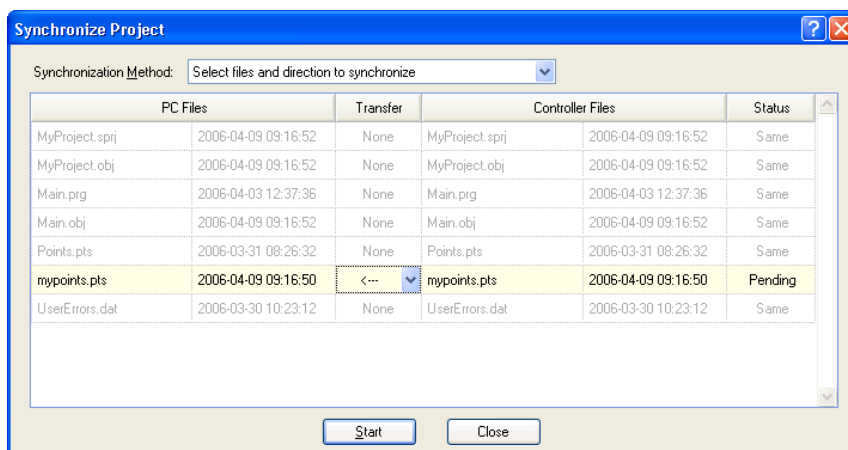
Item Description

Synchronization Method	Select the desired method for synchronization. There are three methods:
	- Select files and direction to synchronize
	- Send files only from PC to Controller
	- Send files only from Controller to PC
Transfer	When the synchronization method is set for *Select files and direction to synchronize*, you can change the direction of transfer by clicking in the transfer cell for the desired file and then clicking on the dropdown list arrow.

TIP
☞
If the status is "Select", the files are updated in both the Controller and PC. Select the direction to synchronize.

Start	Click this button to start the transfer of files. This button is grayed out if there are not files to transfer.
Close	Close the dialog.

5.9.15 Properties Command (Project Menu)

Project: Properties: General Page

Use this page to view and edit general properties for the current project. All project property settings are stored in the project file, which is also stored in the controller during project build.
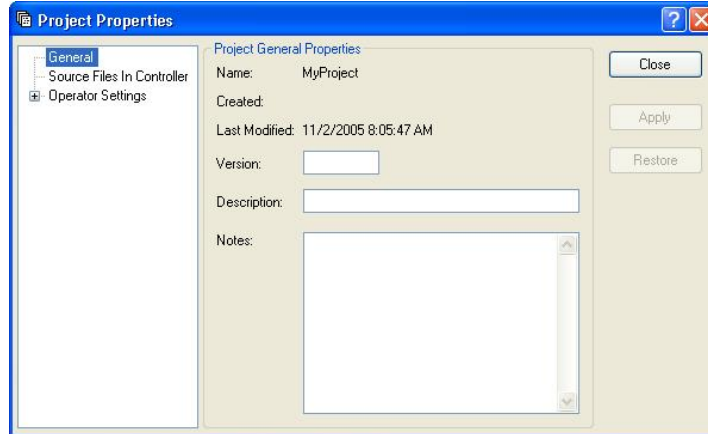
Item	Description
Name	The name of the current project.
Created	Date and time when the project was created.
Last Modified	Date and time when the project was last modified.
Version	User version number of the project. You can type any text here.
Description	A description of the project. You can type any text here.
Notes	Any project notes can be entered into this section.
Close	Close the Project Properties dialog.
Apply	Set current values after changes have been made.
Restore	Revert back to previous values.

TIP
☞ When the Open Project dialog is used, the **Project Info** button will open a dialog that contains the general project properties entered on this page.

Project: Properties: Compiler Page

This page allows you to configure the compiler settings.
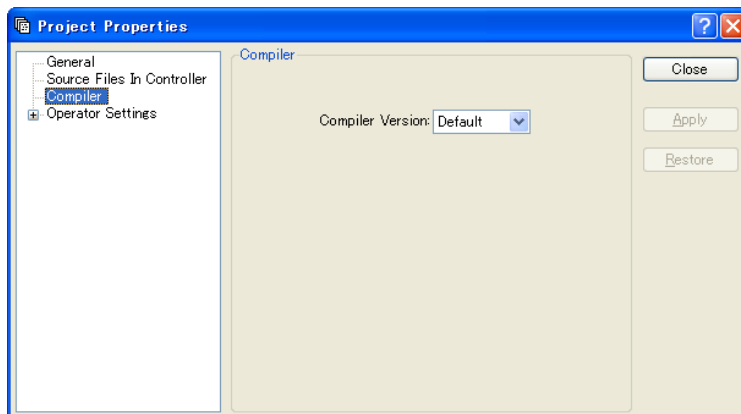
Item	Description
Compiler Version	[Default] is the normal setting. When a project cannot be built because new keywords have been added to the language that interfere with declared variables or functions, you can select the compatible compiler version to build the project. Specify the controller firmware version that is used for the project.

Project: Properties: Source Files In Controller Page

This page allows you to select which source files will be stored in the controller during project build.

Operators can edit one or more programs in the controller via the teach pendant by providing a password. So you can store just the programs you want the operators to be able to change. (Only TP1. TP2 does not support this function.)

After applying changes, the next project build will clear the project in the controller and perform a rebuild.
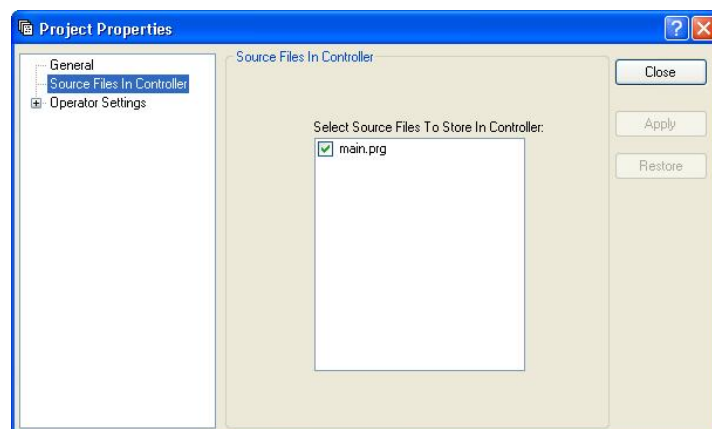
Item	Description
Select Source Files To Store in Controller	This is a list of the source files in the project. Select which source files you want to have stored in the controller.
Apply	Set current values after changes have been made.
Restore	Revert back to the previous values.
Close	Close the Project Properties dialog.

Project: Properties: Operator Settings
: Operator Window: General Page

This page allows you to configure the general settings for the Operator Window.
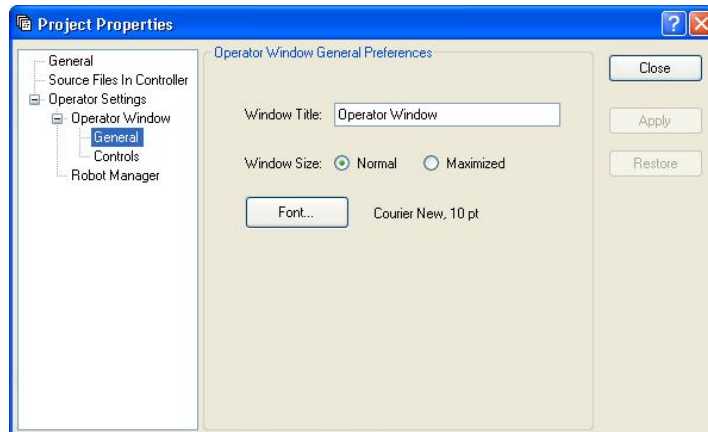
Item	Description
Window Title	Type in the title that you want to appear at the top of the operator window.
Window Size	Choose Normal (normal size, centered on screen) or Maximized (full screen).
Font	Click on the Font button to open the fonts dialog. Choose the font you desire for the operator window. The current font name and size is displayed next to the Font button.
Apply	Set current values after changes have been made.
Restore	Revert back to the previous values.
Close	Close the Project Properties dialog.

Project: Properties: Operator Settings
: Operator Window: Controls Page

This page allows you to configure the controls for the Operator Window.
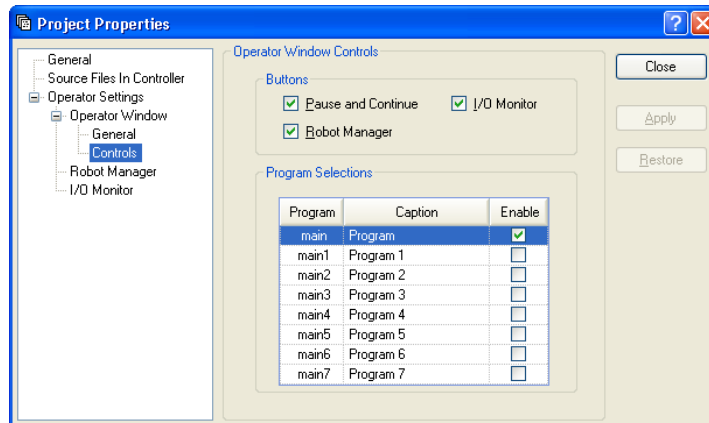
Item	Description
Pause and Continue	Check this box if you want the Pause and Continue buttons to be displayed. This will allow the operator to pause and continue from the operator window.
I/O Monitor	Check this box if you want the I/O Monitor button to be displayed. This will allow the operator to view input and output status.
Robot Manager	Check this box if you want the Robot Manager button to be displayed. This will allow the operator to open the Robot Manager from the operator window.
Program Selections	For each program, you can edit the Caption and select if you want that program to be enabled for the Operator Window. See details below.
Apply	Set current values after changes have been made.
Restore	Revert back to the previous values.
Close	Close the Project Properties dialog.

Program Selections Details

Each project can have up to 8 programs that can be started from the Operator Window.
The programs are named main, main1 main2, … main7. Each program has an associated
startup function using the same name as the program (main, main1, main2…main7). In the
program selections grid, you can define an operator friendly name for each of the eight
programs. You can also define which selections will be displayed in the Operator Window
program list by checking the Enable checkbox.

Project: Properties: Operator Settings: Robot Manager Page

Use this page to configure the Robot Manager for operators.
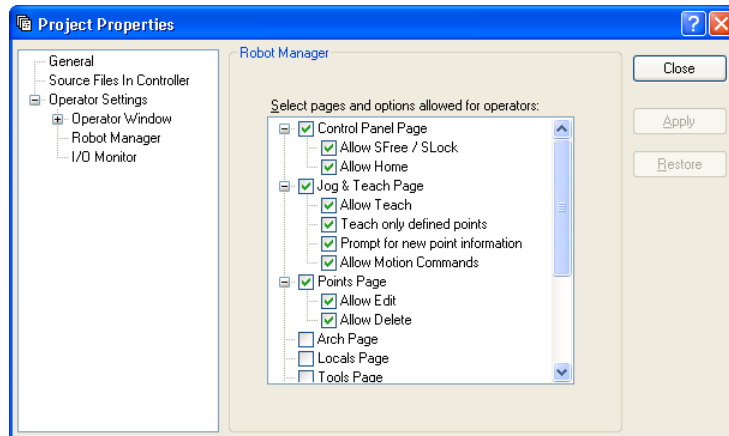
Item	Description
Page and options enabled for operators	Check the pages that you want the operator to have access to when the Robot Manager is displayed from the operator window. In some pages, there are additional options.
Allow SFree / SLock	Allows the operator to free or lock joints from the Control Panel page.
Allow Home	Allows the operator to home the robot from the Control Panel page.
Allow Teach	Allows the operator to teach points from the Jog & Teach page.
Teach only defined points	Only defined points are shown in the point list on the Jog & Teach page.
Prompt for new point information	When the operator teaches a new point, a dialog will be displayed for entering the point label and description.
Allow Motion Commands	Allows the operator to execute motion commands from the Jog & Teach page.
Allow Edit	Allows the operator to edit point data on the Points page.
Allow Delete	Allows the operator to delete points on the Points page.
Apply	Set current values after changes have been made.
Restore	Revert back to the previous values.
Close	Close the Project Properties dialog.

Project: Properties: Operator Settings: I/O Monitor Page

Use this page to configure the I/O Monitor for operators.
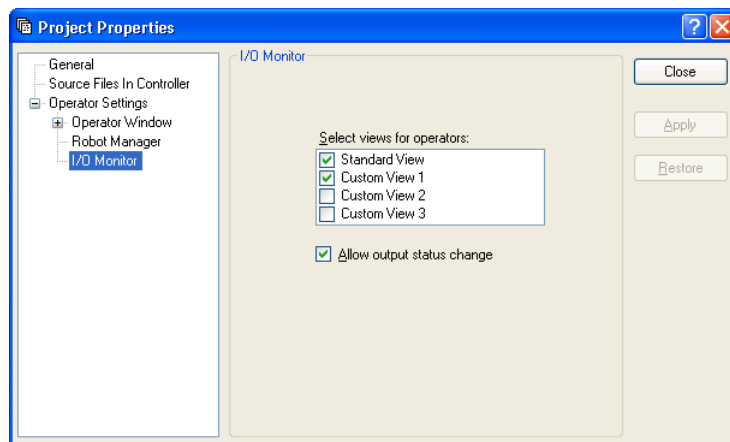
Item	Description
Views Enabled for Operators	Check that I/O views that you want your operators to use when they open the I/O Monitor from the operator window. You can configure the custom views from the I/O Monitor when it is open from the main window. Operators cannot configure custom views.
Allow output status change	Check this box if you want to allow operators to turn outputs on or off.
Apply	Set current values after changes have been made.
Restore	Revert back to the previous values.
Close	Close the Project Properties dialog.

Project: Properties: Vision

The EPSON Smart Camera supports two vision projects simultaneously. Each vision project can be used by one controller, so two controllers can use the same camera. Project 1 is used by default. In this page, you can configure the project number used by each Smart Camera for this project.
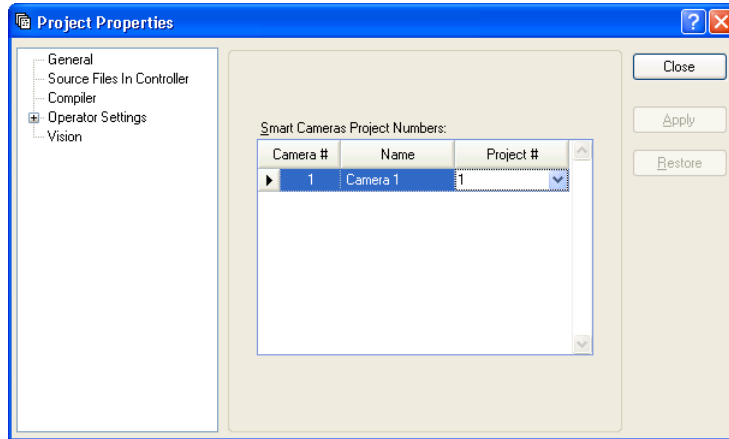
Item	Description
Smart Camera Project Numbers	Select the project number used by each Smart Camera for this project in the "Project #" column.
Apply	Set current values after changes have been made.
Restore	Revert back to the previous values.
Close	Close the Project Properties dialog.

5.10 Run Menu

The EPSON RC+ 5.0 Run Menu includes commands for running and debugging programs.

5.10.1 Run Window Command (Run Menu)

Opens the Run window to run a program in the current project. Before opening the Run window, you are prompted to save all files if there are any unsaved files if the *Auto File Save* preference is off (see Setup | Preferences | Workspace). Otherwise, all files are saved automatically. Then the project is built, if required. If there are any errors during build, the Run window will not be opened.

After the Run window is open, you must click on the Start button to initialize program execution.

For more information, see *The Run Window* section in the chapter *Running and Debugging Programs*.

Shortcuts

Toolbar:

Keys: F5

5.10.2 Operator Window Command (Run Menu)

Opens the Operator Window to run the current project. Before opening the Operator window, you are prompted to save all files if there are any unsaved files. Then the project is built, if required. If there are any errors during build, the Operator Window will not be opened.

If the project is ready to run (last build was successful), then the Operator Window will be opened.

For more information, see the Operator Window section in the chapter *Building Applications in SPEL$^+$*.

Shortcuts

Keys: Shift+F5

5.10.3 Step Into Command (Run Menu)

Execute the current source line. If the current line is a function, the next step will be the first line in the function.

Shortcuts

Toolbar:

Keys: F10, F8

5.10.4 Step Over Command (Run Menu)

Execute the current source line. If the current line is a function, the entire function will be executed.

Shortcuts

Toolbar:

Keys: F11, Shift+F8

5.10.5 Walk Command (Run Menu)

Execute lines until after the next motion command or output command, depending on the *Walk stops for output commands* preference on the Setup | Controller | Preferences page.

Shortcuts

Keys: F12

5.10.6 Resume Command (Run Menu)

Opens the Resume Tasks dialog box. Use this command to resume one or more halted tasks. This command is available only when one or more tasks is in halt mode.

Shortcuts

Toolbar:

Keys: F7
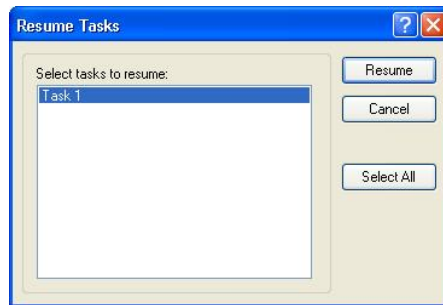
Item	Description
Select tasks to resume	A list of all currently halted tasks. Click on one or more tasks to resume.
Resume	Click to resume.
Select All	Click to select all of the tasks in the list.

5.10.7 Stop Command (Run Menu)

Stops all tasks. This command is disabled when no tasks are running.

Shortcuts

Toolbar:

5.10.8 Toggle Breakpoint Command (Run Menu)

Sets the current line as a breakpoint or returns the current line to normal. When a line is a breakpoint, a breakpoint icon is displayed in the program window left margin.

You can set breakpoints while tasks are running.

If a line cannot be a breakpoint (such as a blank line), then the breakpoint icon will not appear for that line.

Shortcuts

Toolbar: 🖑

Keys: F9

5.10.9 Clear All Breakpoints Command (Run Menu)

Clears all breakpoints.

Shortcuts

Keys: Ctrl+Shift+F9

5.10.10 Display Variables Command (Run Menu)

Displays a dialog box that shows the values for all variables in robot controller memory.

Shortcuts

Keys: F4
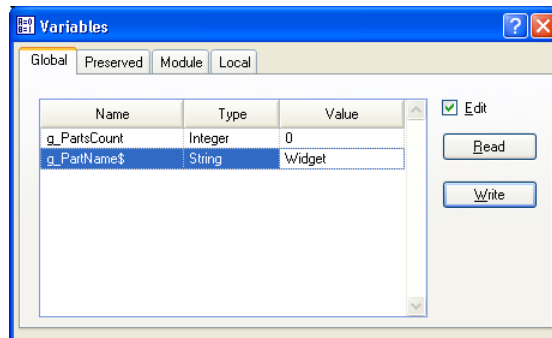
To change a variable value

1. Check the **Edit** checkbox.

2. Type the new value in the Value column. As you type in new values, the text color changes to red, indicating that the value is new and as not been written.

3. Click the **Write** button to save the changes. Click **Read** or uncheck Edit to cancel changes and restore the previous values.

When an array is displayed, the first element is shown. You can change which element to view by typing in the desired array subscript and then clicking the **Read** button.
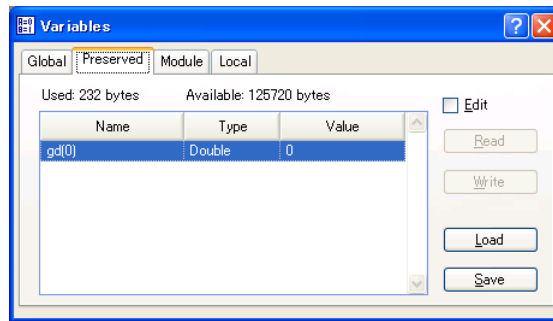
The Preserved page displays the Global Preserve variables. The numbers of used and available bytes for preserved variables are also displayed.
You can save the values of Global Preserve variables in the controller to a file on the PC by clicking the **Save** button. The default file name is GlobalPreserves.dat.

A GlobalPreserves.dat file is also saved by using Backup Controller from the Tools Menu.

You can load the variables that are stored in the file on the PC by clicking **Load** button.

For module variables, you must select the desired program.

Local variables are not displayed unless one or more tasks have reached a breakpoint or have been halted from the Task Manager. You can view local variables for each function in the call stack for each halted task.

5.10.11 Call Stack Command (Run Menu)

The Call Stack dialog displays the function call stack for one task.
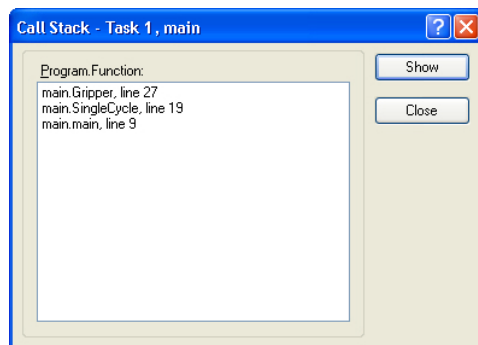
The Call Stack command is available when a program window is clicked which contains a function that is currently halted.

The most recent function is at the top of the list, and parent functions are listed afterwards in descending order. The last function is the task function.

Each row in the list shows a program, function, and line number.

You can view the code for any of the function calls in the list by selecting a function, then clicking **Show**. The program window for the function you selected is then displayed and the line of the function call is marked by a yellow arrow in the editor left margin.

5.11 Tools Menu

EPSON RC+ 5.0 has several GUI tools to aid in development. All tools can be accessed from the Tools Menu. Many also have tool bar buttons and hot keys.

The Tools Menu includes the following selections:

- Robot Manager
  Motor control, Jog & Teach, change robot parameters.
- Command Window
  Execute SPEL commands directly.
- I/O Monitor
  Monitor and change I/O status.
- Task Manager
  Monitor and control task execution.
- Macros
  Opens the Command Macros Window.
- I/O Label Editor
  Define and edit I/O labels.
- User Error Editor
  Define and edit user errors.
- Controller
  Use controller tools, such as backup and restore.
- Simulator
  Opens the Simulator window.
- Vision
  Opens the Vision Guide window.

5.11.1 Robot Manager Command (Tools Menu)

This command opens the Robot Manager window. This window contains several tabs that are used to control the robot motors and power, jog the robot and teach points, and view/edit several parameters for the robot.

You can configure how the Robot Manager window can be viewed in the development environment from the Setup | Preferences | Robot Manager | General page.

MDI window   The Robot Manager is displayed as a child window along with the other child windows inside the EPSON RC+ 5.0 development environment main window.

Dialog   The Robot Manager is displayed as a modal dialog which is displayed in the foreground over the development environment main window.

NOTE
☞   If the screen resolution is less than 1024 x 768, the Robot Manager will always be displayed in dialog mode so it can fit on the screen.

Shortcuts

Toolbar:

Keys: F6

Tools: Robot Manager: Control Panel Page

The Control Panel page contains buttons for basic robot operations, such as turning motors on/off and homing the robot. It also shows status for Emergency Stop, Safeguard, Motors, and Power.
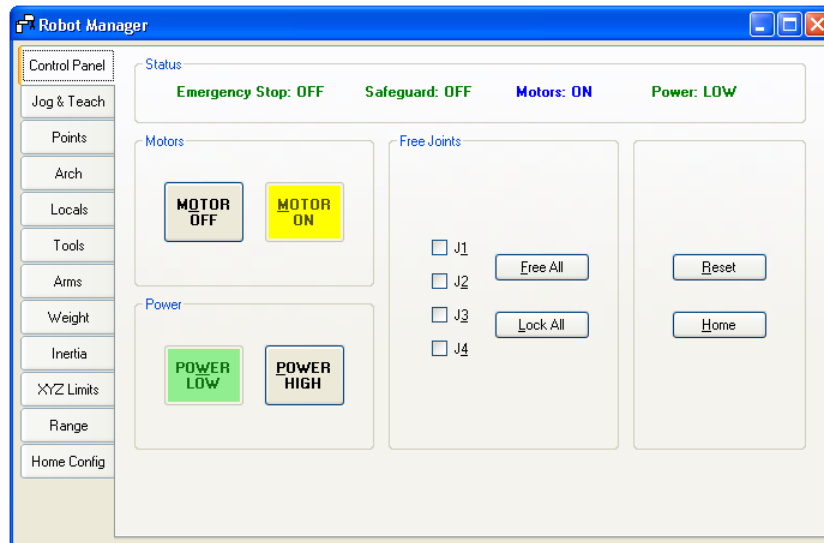
Status Indicators

Indicator	Description
Emergency Stop	Indicates if Emergency Stop has occurred. Once an Emergency Stop occurs, you must execute a Reset to clear it.
Safeguard	Indicates whether the Safeguard input is on or off.
Motors	Indicates whether the robot motors are on or off.
Power	Indicates whether the robot motor power is high or low.

Controls	Description
MOTOR ON	Turns on all robot motors for the robot. You will be prompted to confirm this operation.
MOTOR OFF	Turns off all robot motors for the robot.
POWER LOW	Puts the robot servo system in low power mode.
POWER HIGH	Puts the robot servo system in high power mode.
J1 to J4 checkboxes	You can free one or more joints using the checkboxes. These will be disabled for 6-axis robots.
Free All	Click this button to free all joints from servo control.
Lock All	Click this button to lock all joints under servo control.
Reset button	Resets the robot servo system and Emergency Stop condition.
Home button	Moves the robot to the position specified with the HomeSet command.

Tools: Robot Manager: Jog and Teach Page

The Jog & Teach page is primarily used for jogging the robot to a desired position and teaching a point using the current coordinates and orientation.

You can jog the robot in World, Tool, Local, Joint, or ECP modes. You can also execute motion commands.
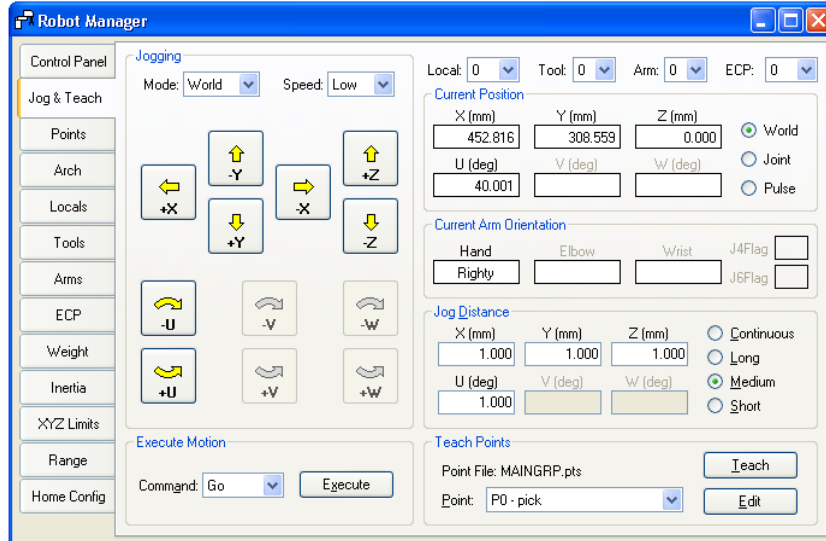
Jog Controls

The Robot Manager Jog & Teach page contains several controls, described below.

Jogging Group

This group contains controls for setting jog mode, speed, and jogging.

Mode

This dropdown list contains the following choices jog mode.

World   Jogs the robot along the X, Y, Z axes in the current local, tool, arm, and ECP. For robots with 4 DOF, you can also jog U (roll). For robots with 6 DOF, you can jog U (roll), V (pitch), and W (yaw). This is the default setting.

Tool   Jogs the robot in the coordinate system defined by the current tool.

Local   Jogs the robot in the coordinate system defined by the current local.

Joint   Jogs each joint of the robot. A separate set of jog buttons will appear when using joint mode when using non-Cartesian robots.

ECP   Jogs the robot along the axes of the coordinate system defined by the current external control point. Coordinates are World coordinates.

Speed

The speed for jogging and motion commands can be changed by selecting Low or High. When the Robot Manager is first open, the speed is set to Low. Jogging is always in low power mode. The speeds and accelerations associated with the jog speed settings are shown in the next page.

Jog Speed	Jog Method	Speed	Accel	Decel
Low	Continuous World/Tool/ECP XYZ	10 mm/sec	100 mm/sec^2	200 mm/sec^2
	Continuous World/Tool/ECP UVW	2 deg/sec	20 mm/sec^2	40 mm/sec^2
	Continuous Joint	*	10 mm/sec^2	20 mm/sec^2
	Step	1/5 of default PTP speed	Default PTP acceleration	Default PTP deceleration
High	Continuous World/Tool/ECP XYZ	50 mm/sec	100 mm/sec^2	200 mm/sec^2
	Continuous World/Tool/ECP UVW	10 deg/sec	20 mm/sec^2	40 mm/sec^2
	Continuous Joint	*	10 mm/sec^2	20 mm/sec^2
	Step	Default PTP speed	Default PTP acceleration	Default PTP deceleration

*Continuous joint speed and acceleration depends on robot model

Jog Buttons

The jog buttons are used to jog the robot throughout the work envelope. They can be actuated only with the mouse.

You can jog one step at a time by setting the Jog Distance to Long, Medium, or Short and then clicking on a button and releasing. You can step continuously by holding the button down. To jog continuously without stepping, set the Jog Distance to Continuous. See *How to jog robot* for details

You can change the orientation of the jog buttons to align your PC monitor with the robot from Setup: Preferences: Robot Manager: Jogging.

The jog buttons that are displayed depend on the Jog mode. For World, Local, Tool, and ECP jogging, the X, Y, Z, U, V, W buttons appear (V and W are enabled only for 6-Axis robots). For Joint jogging, the joint buttons appear, labeled J1 - J6.

The X, Y, and Z buttons jog the robot along the associated Cartesian axis.

The U buttons rotate the tool coordinate system about the Z axis. This is also known as *roll*.

For 6-Axis robots, the V buttons rotate the tool coordinate system about the Y axis. This is also known as *pitch*. The W buttons rotate the tool coordinate system about the X axis. This is also known as *yaw*.

Execute Motion Group

This group has controls for executing motion commands. Select the desired motion command from the Command dropdown list. Then click the Execute button. A dialog is displayed that lets you select the desired parameters for the motion command. Click Execute from this dialog to execute the motion.
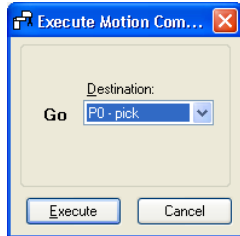
Motion commands can be disabled from Setup: Preferences: Robot Manager: Jogging.

Local

This drop down list is used to select the current Local for jogging and teaching. Only Locals that have been defined are shown in the list. When you teach a point, the Local point attribute defaults to the current local number.

Tool

This drop down list is used to select the current Tool for jogging and teaching. Only Tools that have been defined are shown in the list.

Arm

This drop down list is used to select the current Arm for jogging and teaching. Only Arms that have been defined are shown in the list. Arms are not used with 6-axis robots.

ECP

This drop down list is used to select the current ECP for jogging. Only ECPs that have been defined are shown in the list. ECPs are only allowed if the External Control Point option has been activated.

Current Position Group

This group displays the current position of the robot. There are three ways to display position. **World** displays the current position and tool orientation in the selected local coordinate system, **Joint** displays the current joint values, and **Pulse** displays the current encoder pulse count for each joint.

Current Arm Orientation Group

The current Hand, Elbow, Wrist, J4Flag, and J6Flag values are displayed. For 4 axes robots, only Hand is displayed.

Jog Distance Group

This group contains text boxes that are used to specify the distance that each axis moves when its corresponding jog button is pressed. There are radio buttons for selecting Continuous, Long, Medium, and Short jog distances. When Continuous is selected, the robot is jogged in continuous mode and the jog distance text boxes are grayed out. When Long, Medium, or Short are selected, the robot is jogged in step mode for the distance specified in the jog distance text box for the axis being jogged.

To change a jog distance, first select the distance to be changed, then type in the new value.

Distance	Set Value *	Default Value
Short	0 to 10	0.1
Medium	0 to 30	1
Long	More than 0 to 180	10

* If you enter a too large value, an error message appears when you attempt to jog.

When the jog mode is changed, the jog distance units change appropriately between millimeters (mm) and degrees (deg).

NOTE
☞
When the jog distance is longer than the default, jog distance is reset to default status by rebooting the controller.

Teach Points Group

This group shows the current point number and name for Teach and Edit commands.

Use the **Teach** button to read the current robot position into the coordinates of the current point.

Use the **Edit** button to select and view the current point in the Points tab.

See *How to teach points* for more information.

How to jog

In the upper left hand corner of the Jog & Teach page, you will see a control group called Jogging that contains jog buttons. In the World, Local, Tool, and ECP jog modes, the robot is jogged in the Cartesian coordinate system (X, Y, Z). In the Joint jog mode, each robot joint can be jogged separately.

The jog speed is determined by the Speed setting. In step mode, each time you click a jog button, the robot moves along the appropriate axis by the amount specified in the **Jog Distance** control group. In continuous mode, when a jog button held down, the robot moves continuously using linear interpolated motion.

NOTE
☞
For robots other than the 6-axis robots, the jog motion in step mode is PTP (point to point) motion. It is difficult to predict exact jog motion trajectory. Therefore, be careful that the robot doesn't collide with peripheral equipment and that the robot arms don't collide with the robot itself during jogging.

For the 6-axis robots, the jog motion in step mode is CP (Continuous Path) motion. Please note that when jogging near the singularity, if you try to pass through the singularity, a warning dialog below will appear.
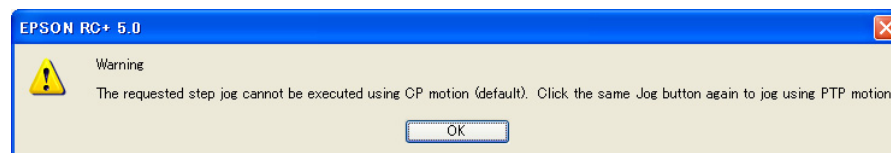
Click the <OK> button and click the same Jog button again to jog using PTP motion and pass the singularity.

It is difficult to predict exact jog motion trajectory in the PTP motion. Therefore, be careful that the robot doesn't collide with peripheral equipment and that the robot arms don't collide with the robot itself during jogging. Also, if you attempt the other jogs or operations, it cancels the switching to PTP motion. So when jogging near the singularity again, the same warning dialog will appear.

When jogging in continuous mode, if an out of range condition occurs, the robot motors will turn off and an error will be displayed. In this case you must execute a Reset and Motor On from the Control Panel page to continue the jog.

To jog

Select the jog mode: World, Tool, Local, Joint, or ECP.

Select the jog speed: Low or High.

Select **Continuous**, **Long**, **Medium**, or **Short** jog distance. You can type in the desired jog distance when Continuous is not selected.

Click on one of the jog buttons with the left mouse button. If you hold the mouse button down, the robot will continue to jog.

When jogging is started, the jog button picture color will change from yellow to cyan. After jogging is completed, the jog button picture color will change back to yellow.

If you click any jog button during a step jog, the robot will stop.

TIP

☞

You can change the orientation of the jog buttons for the robot by selecting Preferences: Robot Manager: Jogging from the Setup Menu. This will allow you to align your PC monitor with the orientation of the robot.

Jogging in Teach Mode

You can jog and move the robot at slow speed with the safeguard open by using the Teach Pendant. See the *Teach Pendant manual* for instructions.

How to teach points

Teaching a robot point means that you are defining a location for the robot to move to using the current robot position.

Follow these steps to teach points from the Robot Manager:

1. Select the point file you are teaching points for by using the **Point File** dropdown list box on the **Points** page.

2. On the **Jog & Teach** page, select the point number you want to teach in the **Teach Points** control group in the lower right corner of the window.

3. Jog the robot to the desired position or free some or all axes and manually move the robot into position.

4. Click on the **Teach** button. This will retrieve the robot's current position and store it in the coordinates for the point. If the Prompt for New Point Data preference is active, you will be prompted for a point label and description.

   Point labels can include up to 32 alphanumeric characters and the underscore character. Characters can be upper case or lower case. Only alphabets can be used for the first letter.
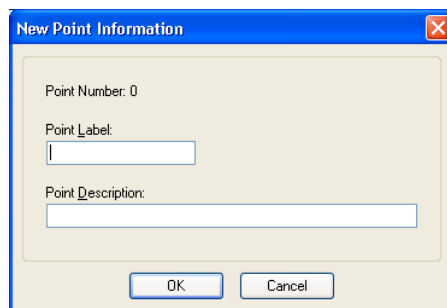
5. As an alternative to clicking the **Teach** button, on the **Points** tab you can type in the coordinates of the point.

Saving or aborting your work

Click the **Save** button on the Robot Manager Points page to save the changes.

Robot Manager MDI Child

When the Robot Manager is used as an MDI child, use the File Menu to select Save or Restore for the current point file. You can also execute Project | Save or click the Save all files toolbar button.

Robot Manager Dialog

When you close the Robot Manager, you will be prompted if you want to save your changes. Answer Yes to make your changes permanent or No to abort all changes.

Tools: Robot Manager: Points Page

The Points page is used for managing the current Robot Manager point file.

You can select any of the point files in the current project. When a point file is selected, the robot controller loads the file into memory.

When a point file is open in the Robot Manager, it cannot be opened in a separate point editor window.

As points are taught on the Robot Manager Jog & Teach page, the spreadsheet on the Points page is updated.

If a point is defined on the Points page by typing in coordinates, the Jog & Teach page will show the point in the Teach Points list.

When the Robot Manager is used as an MDI child window, you can type Ctrl + S to the point file. You can also access the File Menu to save or restore point data.
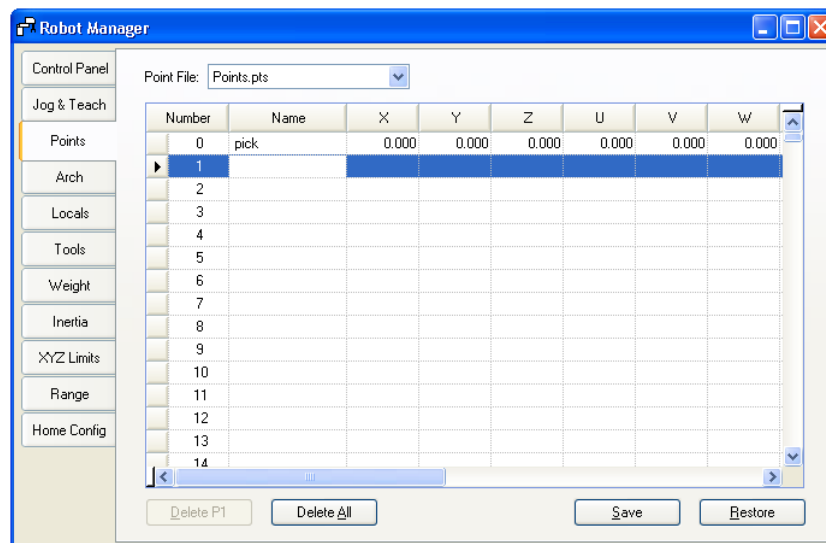
Item	Description
Point File	Selects a point file in the current project for the Points page and the Jog & Teach page.
Delete Pxxx	Deletes the selected point. You will be asked to confirm the operation.
Delete All	Deletes all points in the file. You will be asked to confirm the operation.
Save	Save the current values.
Restore	Revert back to the previous values.

Tools: Robot Manager: Arch Page

This page allows you to configure the depart Z and approach Z settings in the robot's Arch table. Arch is used for the Jump, Jump3, and Jump3CP motion commands.

For details on using Arch with motion commands, see the SPEL+ Language Reference.

There are seven different setting pairs in the Arch table. Each setting pair is one row in the grid on the Arch page.
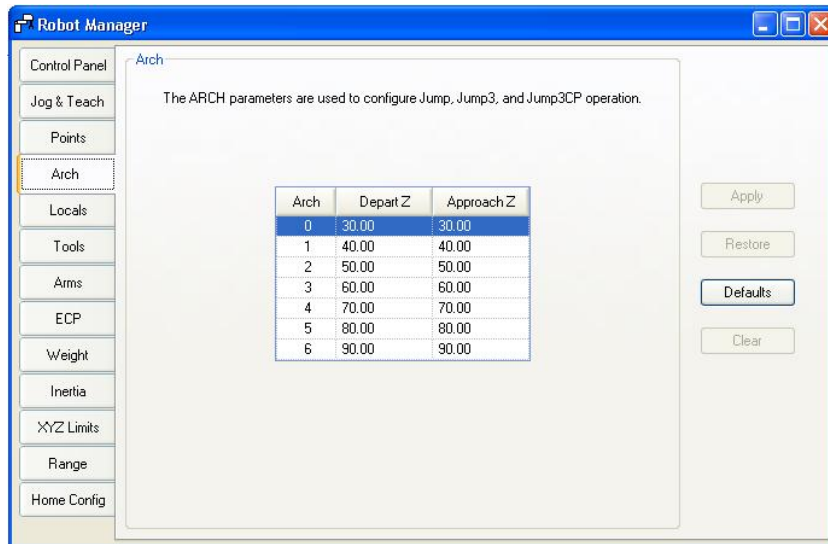
To change Arch settings

1. Put the cursor in the Depart Z or Approach Z cell of the row you want to change.

2. Type in the new value.

Press the TAB key to move to the next cell.

Item	Description
Apply	Set the current values.
Restore	Revert back to the previous values.
Defaults	Click the defaults button to display factory default settings.

Tools: Robot Manager: Locals Page

This page allows you to define local coordinate systems for a robot. When the page is selected, the current values are displayed.

A grid is used to display all of the values for the locals you can define. Local 0 is the base coordinate system and cannot be changed from this page.

NOTE
☞

To change the base coordinate system, use the Base command from the command window. See the SPEL+ Language Reference for more information.

When a local is undefined, then all fields for that local will be blank. When you enter a value in any of the fields for an undefined local, then the remaining fields will be set to zero and the local will be defined when you click the **Apply** button.
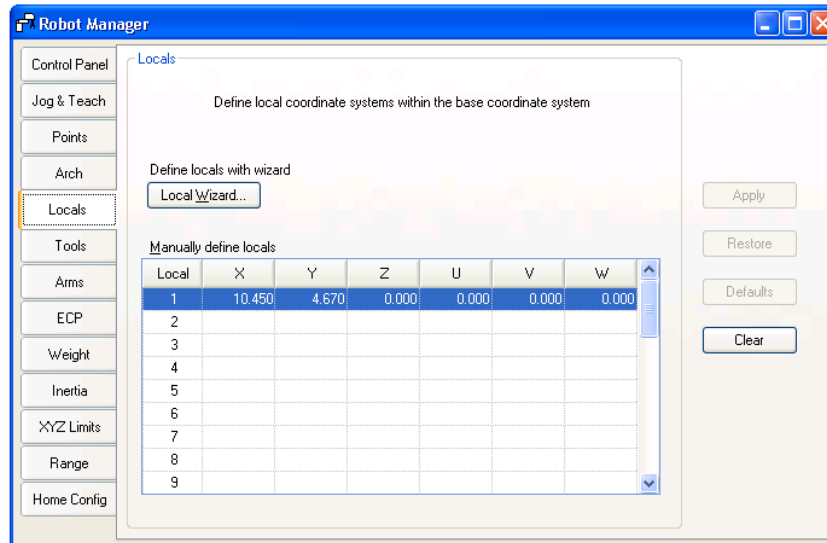
Navigating the grid

Use the TAB key to move to the next field. Use the arrow keys or the mouse to move to any field.

For more information on locals, see the description for the Local command.

Item	Description
Local Wizard	Click this button to start the Local Wizard. Follow the instructions for each step to define a local. See details in the next section.
X	The X coordinate of the local origin in the base coordinate system.
Y	The Y coordinate of the local origin in the base coordinate system.
Z	The Z coordinate of the local origin in the base coordinate system.
U	Rotation angle of the local about the base Z axis (roll).
V	Rotation angle of the local about the base Y axis (pitch).
W	Rotation angle of the local about the base X axis (yaw).
Apply	Save the current changes.
Restore	Revert back to the previous values.
Clear	Clear all values for the selected local.

Using the Local Wizard

A wizard is provided for defining a local coordinate system. You can define a local using a single point or three points, as described in the following sections.

Using the Local Wizard to teach a single point local

1. Open the Robot Manager and click on Locals to show the Locals page.

2. Click the **Local Wizard** button. You will see the dialog shown below.
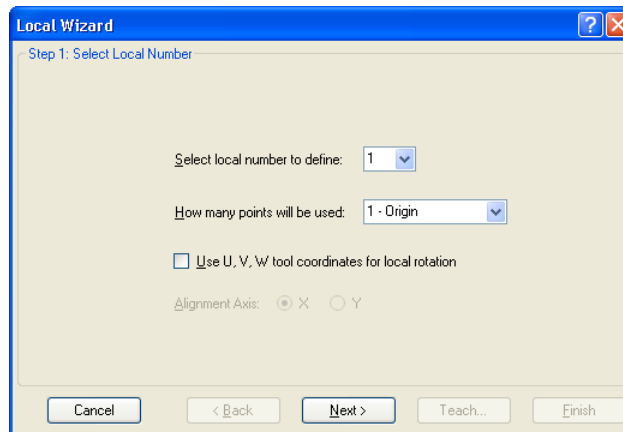
3. Select the local number you want to define. For **How many points will be used**, select **1 - Origin**. Since this is a single point local, you will just teach the origin of the new coordinate system. If you want to use the U, V, or W axes for the orientation of the coordinate system, check the **Use U, V, W tool coordinates for local rotation** checkbox. If this checkbox is unchecked, the new coordinate system is offset from local 0 in X and Y, but is not rotated about any axis. Click the **Next** button.
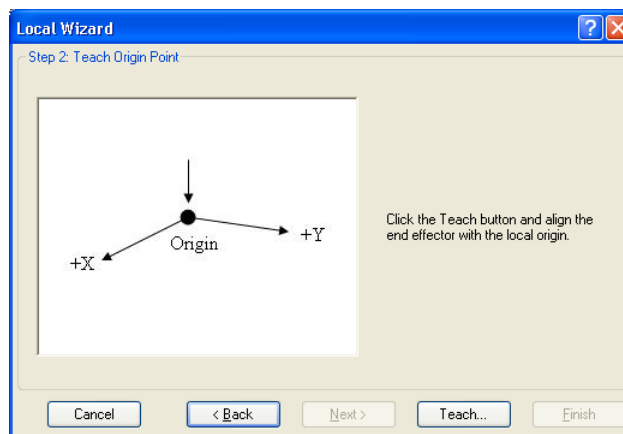
4. We will now teach the local origin point. Click the **Teach** button to open the Local Wizard Teach Point dialog.
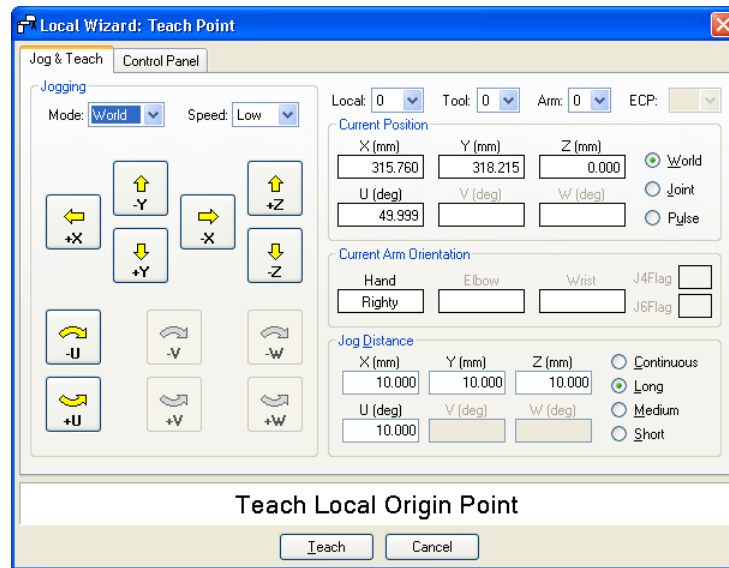
5. Jog the robot until the end effector is aligned with the local origin point. Then click the **Teach** button.

6. The new local definition is displayed as shown below. Click **Finish** to accept the new definition.

Using the Local Wizard to teach a three point local

1. Open the Robot Manager and click on Locals to show the Locals page.

2. Click the **Local Wizard** button. You will see the dialog shown below.
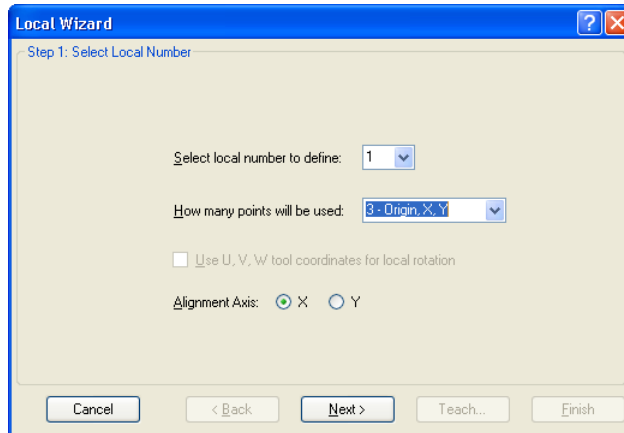
3. Select the local number you want to define. For **How many points will be used**, select **3 - Origin, X, Y**. Since this is a three point local, you will teach the origin of the new coordinate system, and then teach one point anywhere along the X axis and one point anywhere along the Y axis. Select which axis will be used to align the coordinate system. For example, if you select X, then the new coordinate system X axis will be aligned to the X axis point that you will teach in a later step. The Y axis point will be used to determine tilt. Click the **Next** button.
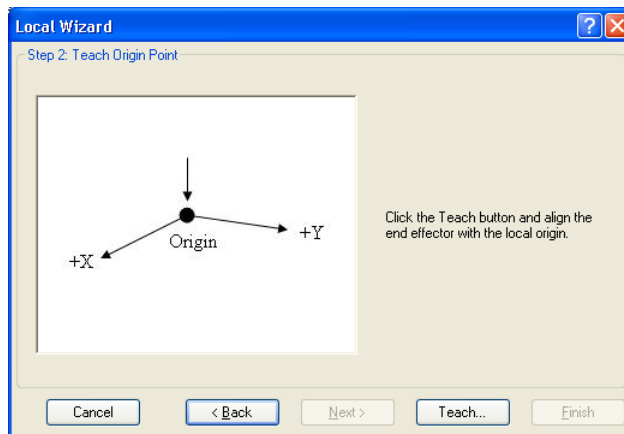
4. We will now teach the local origin point. Click the **Teach** button to open the Local Wizard Teach Point dialog.

5. Jog the robot until the end effector is aligned with the origin point. Then click the **Teach** button. The next step will be displayed.
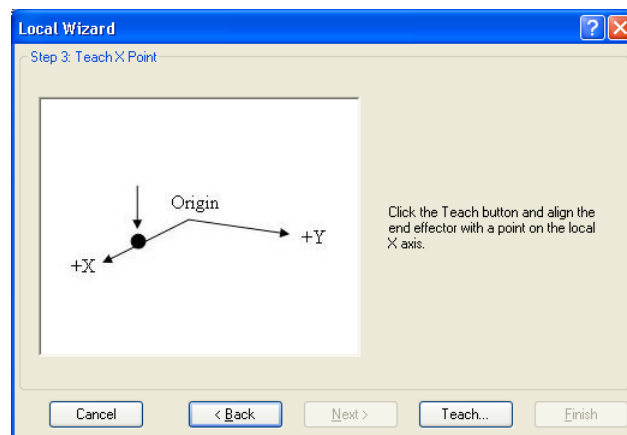
6. We will now teach a point on the local X axis. Click the **Teach** button and jog the robot until the end effector is aligned with a point anywhere along the X axis of the new coordinate system. Click the **Teach** button on the Teach Point dialog to continue.
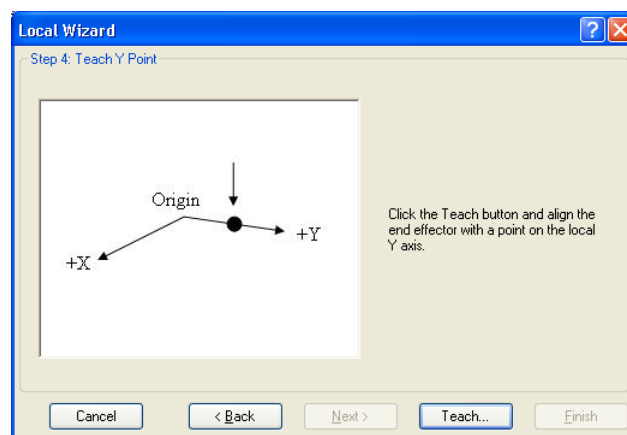
7. We will now teach a point on the local Y axis. Click the **Teach** button and jog the robot until the end effector is aligned with a point anywhere along the Y axis of the new coordinate system. Click the **Teach** button on the Teach Point dialog to continue.

8. The new local definition is displayed as shown below. Click **Finish** to accept the new definition.
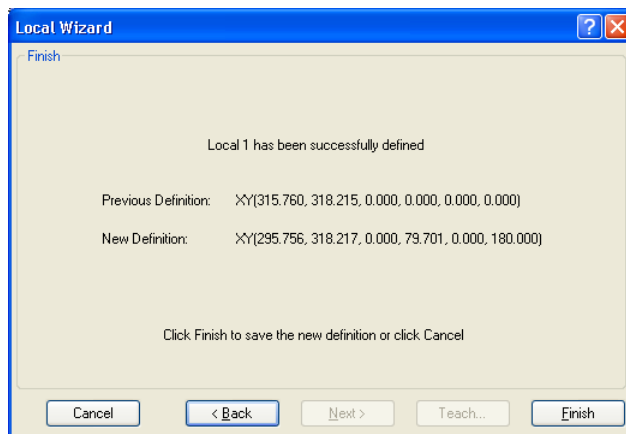
Tools: Robot Manager: Tools Page

This page allows you to define tool settings for a robot. When the tab is selected, the current values are displayed.

A grid is used to display all the values for all 15 tools you can define.

NOTE
☞

When the RC180 Controller firmware is Ver.1.4.*.* or before, the row from 4 to 15 will not be displayed.

When a tool is undefined, then all fields for that tool will be blank. When you enter a value in any of the fields for an undefined tool, then the remaining fields will be set to zero and the tool will be defined when you click the **Apply** button.
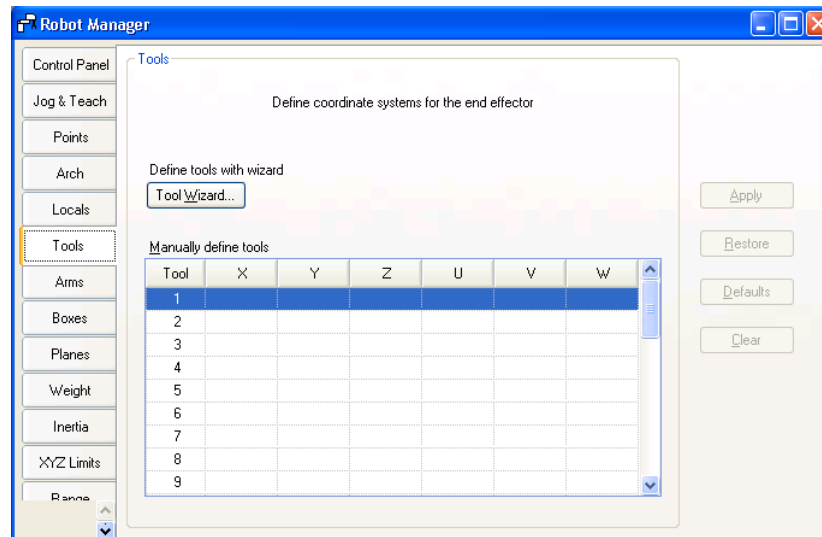
Navigating the grid

Use the TAB key to move to the next field. Use the arrow keys or the mouse to move to any field.

For more information on tools, see the description for the TLSet command.

| Item	Description
Tool Wizard	This button starts the Tool Wizard. Follow the instructions for each step of the wizard to define a tool. See details in the next section.
X	The X coordinate of the tool.
Y	The Y coordinate of the tool.
Z	Z offset of tool.
U	Rotation angle of the tool about the Z axis (roll).
V	Rotation angle of the tool about the Y axis (pitch).
W	Rotation angle of the tool about the X axis (yaw).
Apply	Set the current values.
Restore	Revert back to the previous values
Clear	Clear all values for the selected tool.

Using the Tool Wizard

1. Open the Robot Manager and click on Tools to show the Tools page.

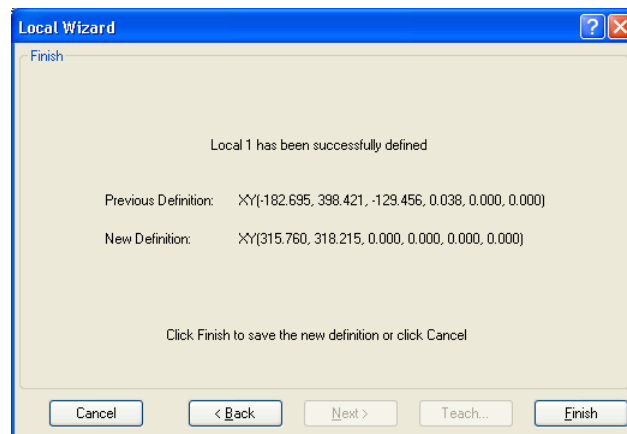2. Click the **Tool Wizard** button. You will see the dialog shown below.
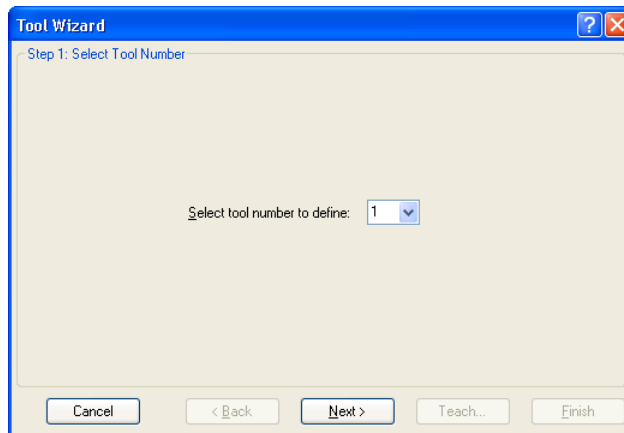
3. Select the tool you want to define and click **Next**.
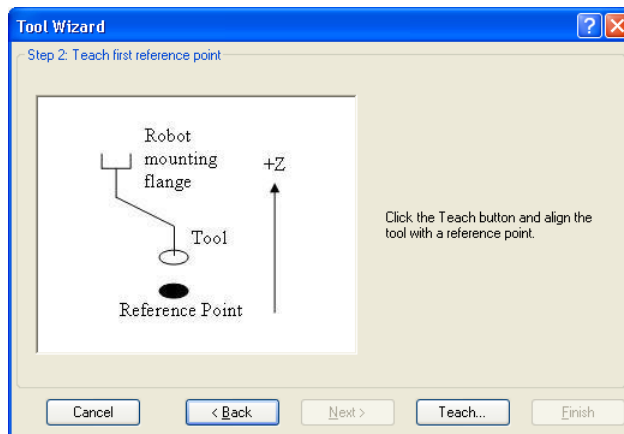
4. Click the **Teach** button to open the Tool Wizard Teach Point dialog.
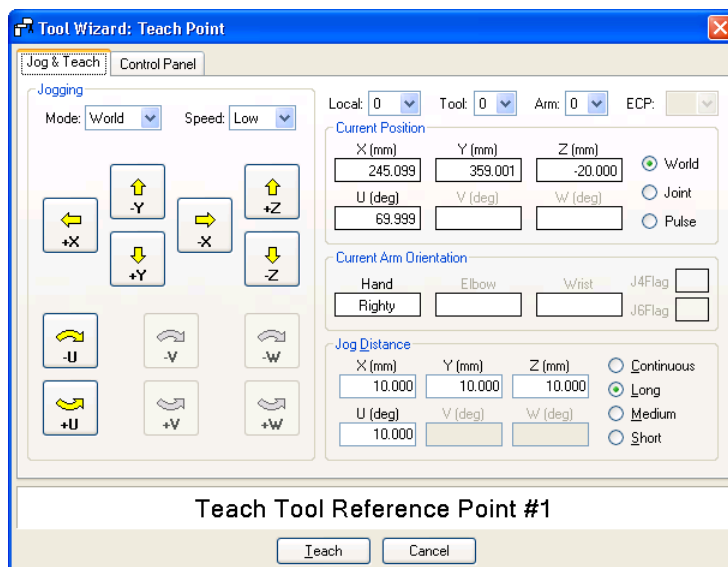
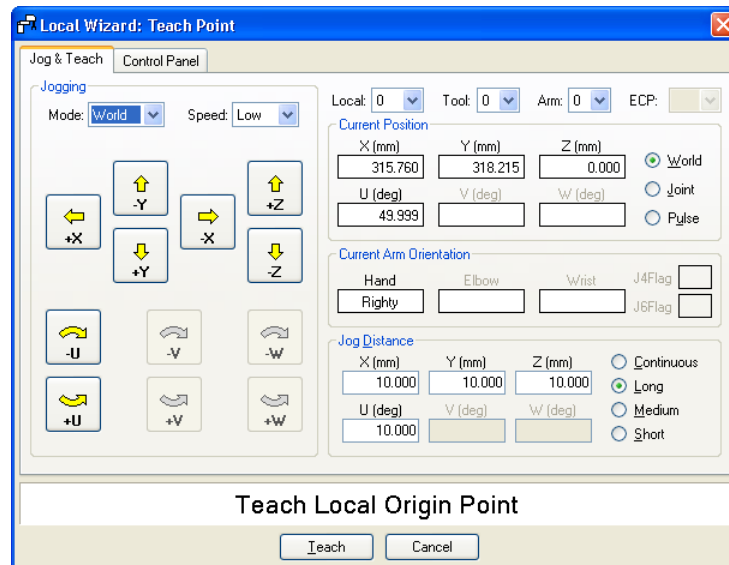5. Jog the robot until the tool is aligned with the reference point. For 6-axis robots, ensure that U = 0, V = 0, and W = 180 at the reference point. Then click the **Teach** button. The next step will be displayed in the wizard.
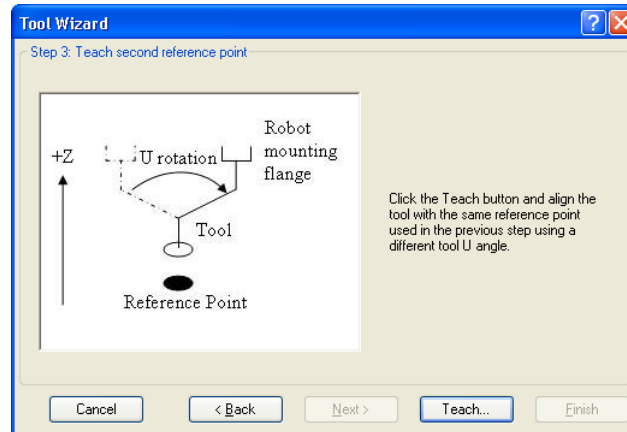
6. Click the **Teach** button. Jog the robot so that the tool is aligned with the reference point, but from another U angle. To do this, jog the U axis several degrees, then jog X & Y until the tool is aligned with the reference point. After you have finished jogging to the reference point, click **Teach**.

7. For 6-axis robots only: Jog the robot back to the reference point with U = 0, V = 0, and W = 180. Jog the W axis several degrees, then jog X and Y until the tool is aligned with the reference point. Ensure that U and V remain at 0 degrees.
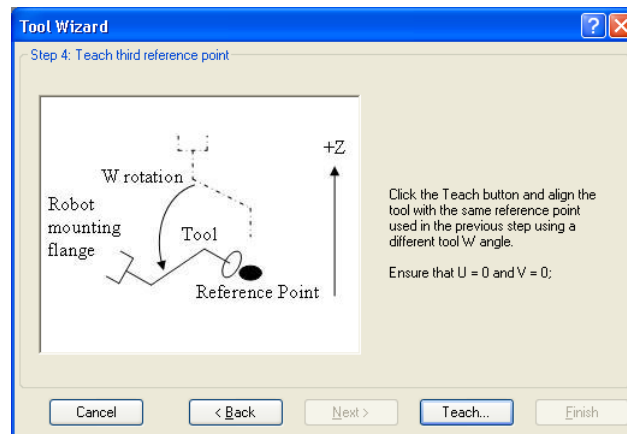
8. The new tool definition is displayed as shown below. Click **Finish** to accept the new definition.
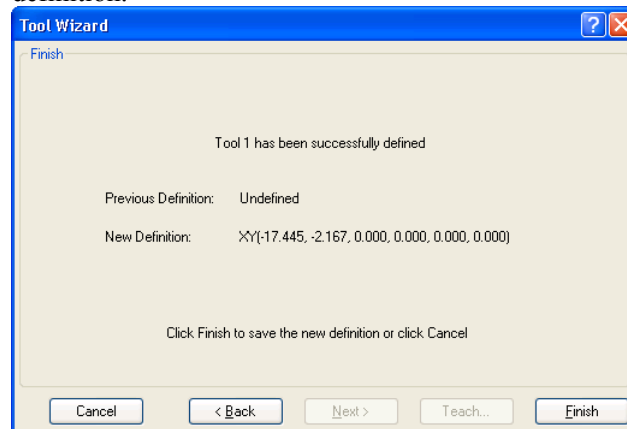
Tools: Robot Manager: Arms Page

This page allows you to define Arm settings for a robot. When the tab is selected, the current Arm values are displayed. The tab is disabled if the current robot does not support the Arm command.

A grid is used to display all the values for all 15 arm configurations you can define.

NOTE
☞
When the RC180 Controller firmware is Ver.1.4.*.* or before, the row from 4 to 15 will not be displayed.
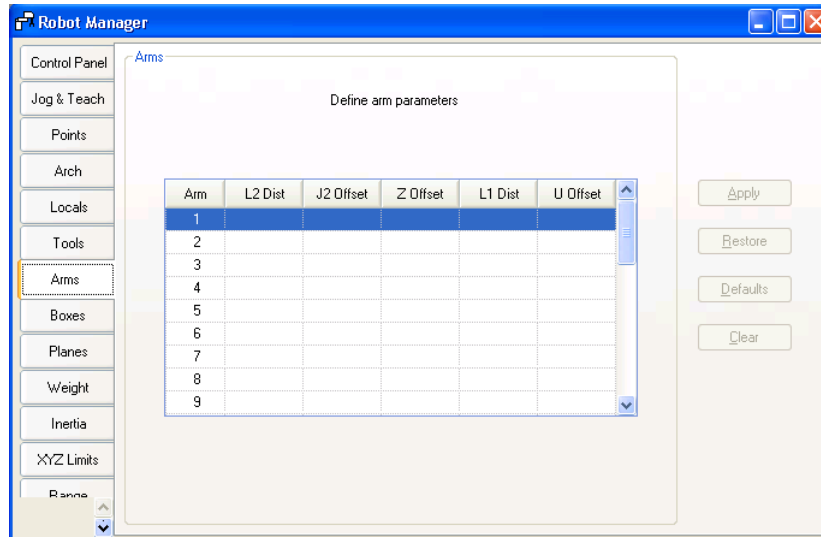
When an arm is undefined, then all fields for that arm will be blank. When you enter a value in any of the fields for an undefined arm, then the remaining fields will be set to zero and the tool will be defined when you click the **Apply** button.

Navigating the grid

Use the TAB key to move to the next field. Use the arrow keys or the mouse to move to any field.

For more information on arm parameters, see the description for the ArmSet command in the SPEL$^+$ Language Reference.

Item	Description
L2 Dist	Distance between the center of joint 2 and the center of the orientation joint in millimeters.
J2 Offset	Angle of the line from the center of joint 2 to the center of the orientation joint in degrees.
Z Offset	The Z offset between the new orientation axis and the standard orientation axis.
L1 Dist	Distance between the center of the shoulder joint and the center of the elbow joint in millimeters.
U Offset	The angle offset between the standard orientation zero position and the new orientation axis zero position in degrees.
Apply	Set current values.
Restore	Revert to the previous values.
Clear	Clear all values for the selected arm

Tools: Robot Manager: ECP Page

This page allows you to define ECP (external control point) settings for a robot. When the page is selected, the current values are displayed.

NOTE
☞

If the ECP option is not enabled in the controller, this page will not be visible.

For detailed information on using external control points in your application, see the chapter *ECP Motion*.

A grid is used to display all of the values for all ECPs you can define.

When an ECP is undefined, then all fields for that ECP will be blank. When you enter a value in any of the fields for an undefined ECP, then the remaining fields will be set to zero and the ECP will be defined when you press the **Apply** button.
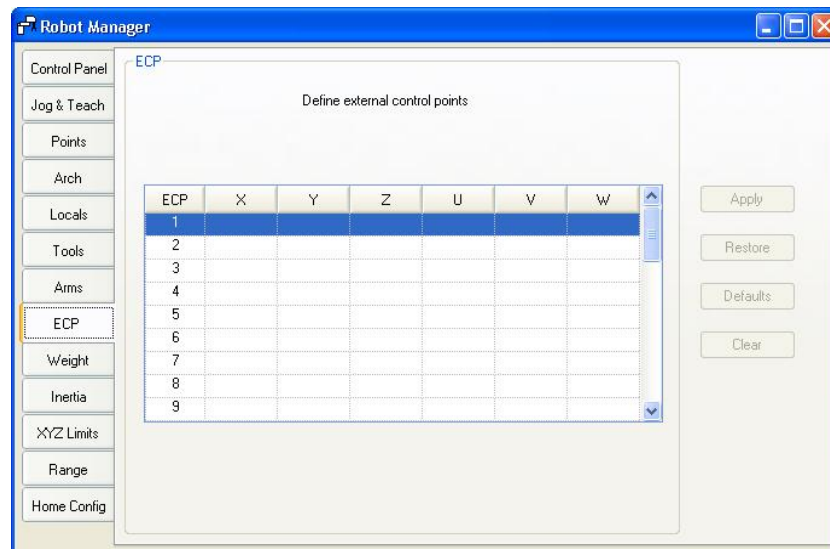

Navigating the grid

Use the TAB key to move to the next field. Use the arrow keys or the mouse to move to any field.

Item	Description
X	The X coordinate of the ECP.
Y	The Y coordinate of the ECP.
Z	The Z coordinate of the ECP.
U	Rotation angle of the ECP about the Z axis (roll).
V	Rotation angle of the ECP about the Y axis (pitch).
W	Rotation angle of the ECP about the X axis (yaw).
Apply	Set current values.
Restore	Revert back to the previous values.
Clear	Clear all values for the selected ECP.

Tools: Robot Manager: Box Page

NOTE

☞

When the RC180 Controller firmware is Ver.1.4.*.* or before, this page will not be displayed.

This page allows you to define Box (approach check area) settings for a robot. When the page is selected, the current values are displayed. When a Box is undefined, then all fields for that Box will be blank. When you enter a value in any of the fields for an undefined Box, then the remaining fields will be set to zero and the Box will be defined when you press the **Apply** button.
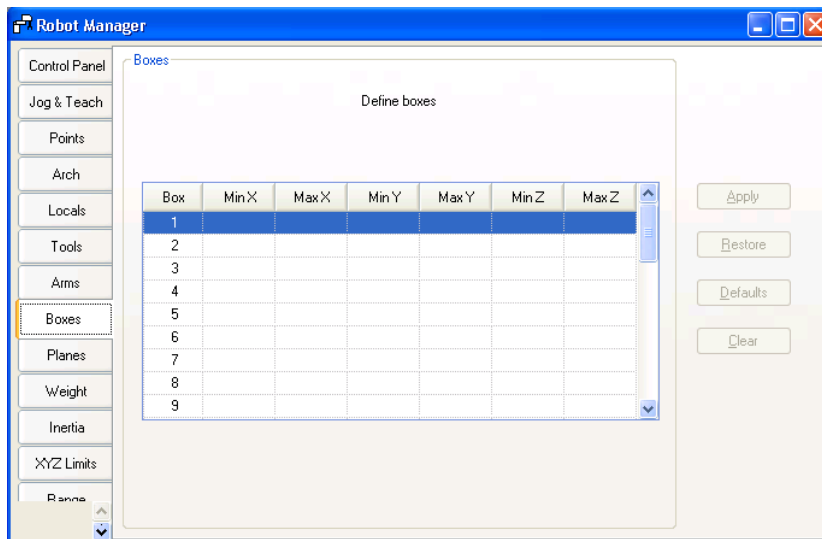
Navigating the grid

Use the TAB key to move to the next field. Use the arrow keys or the mouse to move to any field.

Item	Description
Min X	Type in the minimum X limit value in millimeters.
Max X	Type in the maximum X limit value in millimeters.
Min Y	Type in the minimum Y limit value in millimeters.
Max Y	Type in the maximum Y limit value in millimeters.
Min Z	Type in the minimum Z limit value in millimeters.
Max Z	Type in the maximum Z limit value in millimeters.
Apply	Set current values.
Restore	Revert back to the previous values.
Clear	Clear all values.

Setting both values to zero disables the limits.

Tools: Robot Manager: Plane Page

NOTE

☞

When the RC180 Controller firmware is Ver.1.4.*.* or before, this page will not be displayed.

This page allows you to define Plane (approach check plane) settings for a robot. When the page is selected, the current values are displayed. When a Plane is undefined, then all fields for that Plane will be blank. When you enter a value in any of the fields for an undefined Plane, then the remaining fields will be set to zero and the Plane will be defined when you press the **Apply** button.
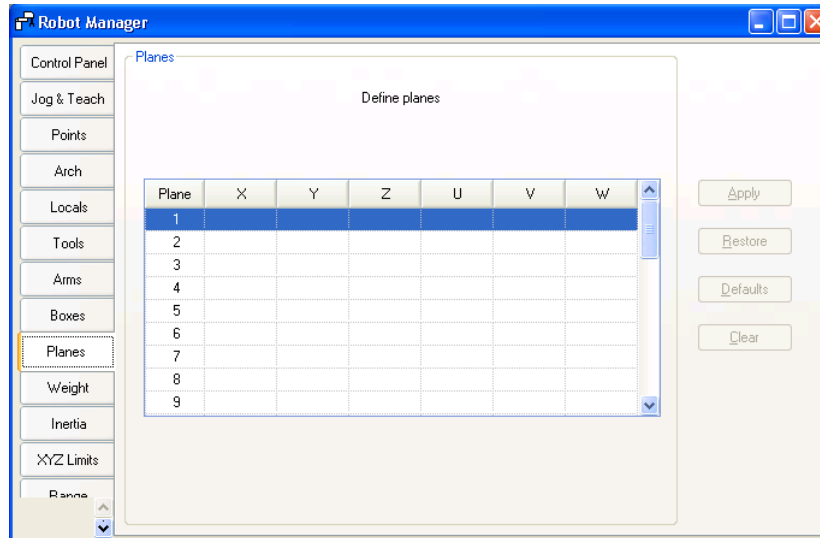
Navigating the grid

Use the TAB key to move to the next field. Use the arrow keys or the mouse to move to any field.

Item	Description
X	Sets the X origin of the coordinate for approach check plane.
Y	Sets the Y origin of the coordinate for approach check plane.
Z	Sets the Z origin of the coordinate for approach check plane.
U	Sets the U origin of the coordinate for approach check plane.
V	Sets the V origin of the coordinate for approach check plane.
W	Sets the W origin of the coordinate for approach check plane.
Apply	Set current values.
Restore	Revert back to the previous values.
Clear	Clear all values.

Tools: Robot Manager: Weight Page

This page is for changing the Weight parameters for the robot.

For details on the Weight parameters, see the SPEL+ Language Reference.
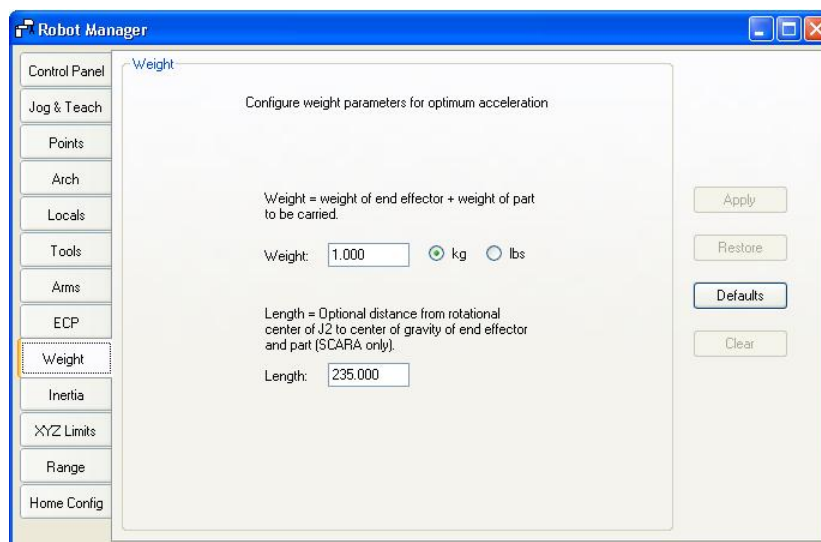
Item	Description
Weight	Type in the new total weight of the payload on the robot.
Kg/Lb	Choose which unit the weight is represented in: kilograms or pounds.
Length	Type in the new length.
Apply	Set the current values.
Restore	Revert back to the previous values.
Defaults	Click the defaults button to display factory default settings.

Tools: Robot Manager: Inertia Page

This page is for changing the Inertia parameters.

For details on the Inertia parameters, see the SPEL+ Language Reference.
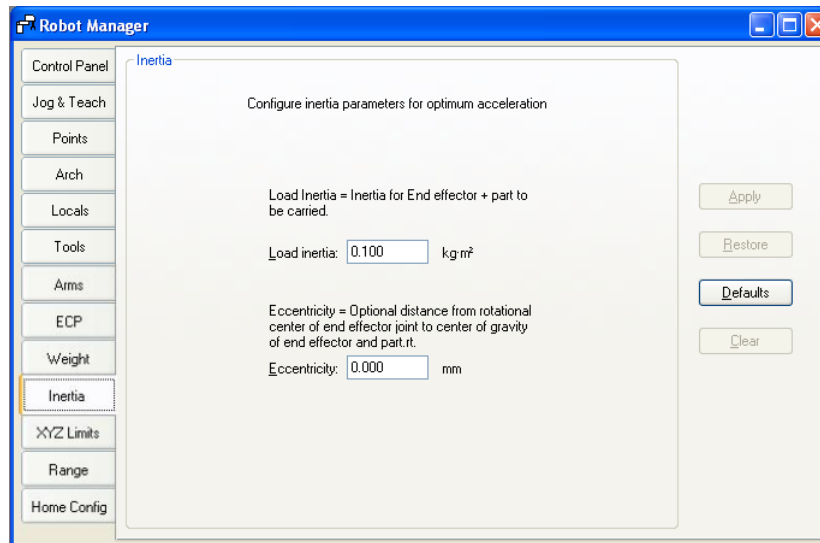
Item	Description
Load inertia	Type in the new load inertia of the payload on the robot in kg·m^2. This includes the inertia of end effector plus the part to be carried.
Eccentricity	Type in the new eccentricity value in millimeters. This is the distance from rotational center of joint 4 to the center of gravity of end effector and part.
Apply	Set the current values.
Restore	Revert back to the previous values.
Defaults	Press the defaults button to display factory default settings.

Tools: Robot Manager: XYZ Limits Page
This page allows you to configure limits for X, Y and Z motion in the robot envelope.
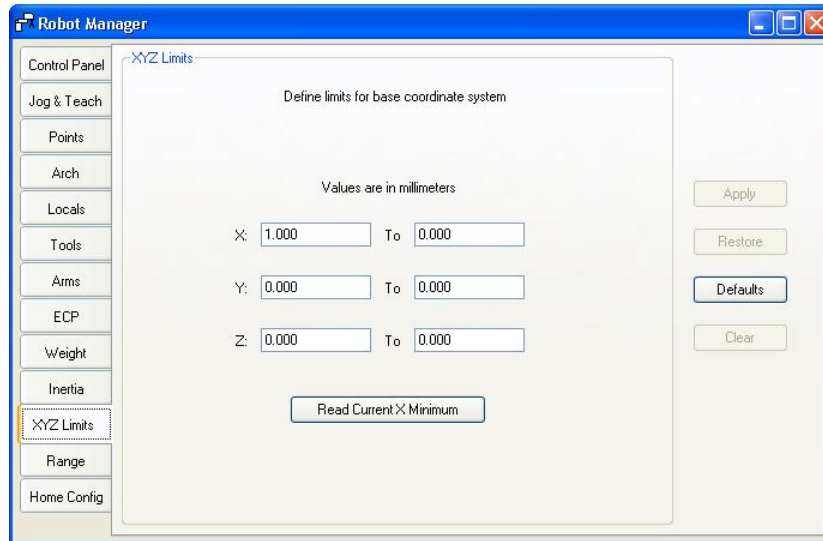
Item	Description
X, Y, Z	Type in the minimum and maximum X, Y, and Z limit values. Setting both values to zero disables the limits.
Read Current	Click this button to read the value from the current robot position. The button text shows the axis and minimum or maximum depending on which text field has the current focus.
Apply	Set the current values.
Restore	Revert back to previous values.
Defaults	Set default values.

Tools: Robot Manager: Range Page

This page allows you to configure the robot joint software limits.

For more information on Range, see the SPEL+ Language Reference and the manual for the robot you are using.
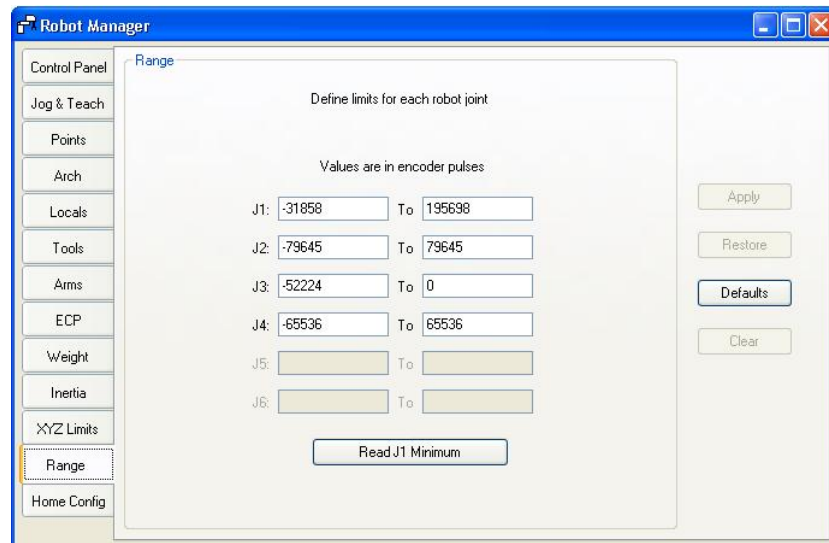
Item	Description
J1 - J6	Type in the minimum and maximum encoder pulse values for each joint.
Read Current	Click this button to read the current joint value of the robot into the currently field. The button text will change depending on which text field has focus.
Apply	Save the current changes.
Restore	Revert back to the previous values.
Defaults	Set the default values.

Tools: Robot Manager: Home Config Page

Home Config allows you to configure the optional user home position and home order used by the Home command.
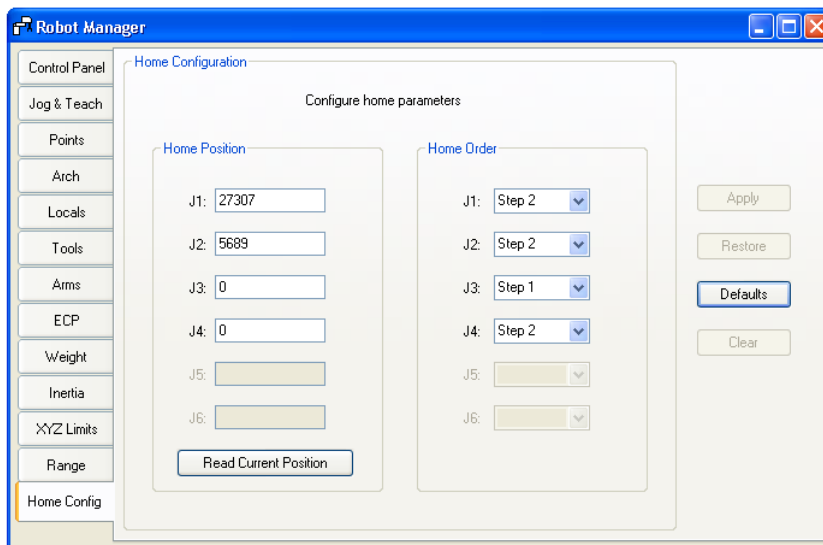
Changing home position

When you select the Home Config tab, the current home position is read from the robot controller and displayed in the text boxes. If the home position has never been defined, then the text boxes will be blank.

To define the home position, you can enter an encoder position value for each of the four robot joints in the text boxes, or you can select the Jog & Teach page to jog the robot to the desired home position, then select the Home Config page and click the **Read Current Position** button to read the current encoder position values.

Changing home order

The home command executes in steps. The number of steps equals the number of joints on the robot. Select the home step number for each joint using the dropdown list for each joint. More than one joint can be homed in the same step.

Testing home

After making changes to the home position and home order, you can click the Robot Manager **Control Panel** tab and click the **Home** button.

Item	Description
Read Current	Click this button to read the current position encoder pulse value into the currently selected text field. The button text will change according which text field is selected.
Defaults	Set the value of the [Home order] group box to the default value.
Apply	Save the current changes.
Restore	Revert back to the previous values.

5.11.2 Task Manager Command (Tools Menu)

The Task Manager window allows you to Halt (suspend), Resume, and Quit (abort) tasks.

To open the Task Manager

Select Task Manager from the Tools Menu.
    Or
Type Ctrl + T.
    Or
Click on the 🔳 button on the toolbar.

Operation

The Task Manager is used for suspending, resuming, stepping, and stopping tasks.

When Task Manager is started, you will see a grid containing status information for 16 tasks standard tasks and 4 trap tasks. There are several items shown for each task. To view all of the columns, use the scroll bar or resize the window.

Task	Number of task from 1 to 16 and four trap tasks.
Name	Name of the function that was started as a task.
Status	Current task status: Run, Waiting, Halt, Pause, Quit, Step.
Type	Task types

	Normal	This task is a normal task
	NoPause	This task does not pause with Pause statement or when Pause input or Safety Door open occur.
	NoEmgAbort	This task continuously processes during the Emergency Stop or error occurrence.

Line	Current task line number.
Function	Current task function name.
Program	Current task program name.
Start	The date and time that the task was started.

Task	Name	Status	Type	Line	Function	Program	Start	
1	main	Run	Normal	6	main	Main.prg	2007/09/18 15:13:23	Halt
2	RobotTask	Run	Normal	11	RobotTask	Main.prg	2007/09/18 15:13:23	Resume
3	IoTask	Wait	Normal	16	IoTask	Main.prg	2007/09/18 15:13:23	Quit
4	MonitorTask	Wait	NoPause	21	MonitorTask	Main.prg	2007/09/18 15:13:23	
5								Pause
6								
7								Continue
8								
9								Stop
10								
11								
12								
13								
14								
15								
16								
Trap 1								
Trap 2								
Trap 3								
Trap 4								

Item	Description
Halt	Suspends the selected task. This temporarily stops the task so that it can be started again with the **Resume** button. **Halt** can only be executed when a task is running (status is Run). When **Halt** is executed, the **Resume** button will be enabled. If a motion command is executing with Halt occurs, the motion will be completed before the task reaches the Halt state. The task also temporarily stops when the task is NoPause type or NoEmgAbort type.
Resume	After one or more tasks have been suspended with the **Halt** button, clicking on **Resume** will make the halted tasks continue where they left off. First, a confirmation dialog is displayed.
Quit	This button stops the selected task permanently. You cannot **Resume** a task once you have executed **Quit**. To restart the task, you must start it from within a program or from the Run window. The task also stops when the task is NoPause type or NoEmgAbort type.
Pause	This button pauses tasks that can be paused. After pause, you must use either **Continue** or **Stop**. The task does not pause when the task is NoPause type or NoEmgAbort type.
Continue	This button continues all tasks that were paused with the **Pause** button.
Stop	This button stops all tasks.

To Halt, Step, Walk, and Resume a task

Use the mouse to click on the desired task in the task list. You can click on any field to select a task. The current task status must be Run before it can be halted. The Halt button will become active after you have selected the task.

After a task has been halted, the source code will be displayed and the next step will be indicated. You can either click on the **Resume** button to resume execution, or executed Step Into, Step Over, or Walk from the Run Menu.

To Pause and Continue tasks

Pause allows you to "suspend" all tasks that can be suspended.

Click on the **Pause** button to pause available tasks. The robot will decelerate to a stop immediately.

After executing Pause, click on **Continue** to resume all suspended tasks.

To view source code at the current execution line

Select a task row. Then right click and select Go To Line. The program editor will be opened at the current execution line

5.11.3 I/O Label Editor Command (Tools Menu)

The I/O label editor lets you define meaningful names for inputs, outputs, and memory I/O for each project. The labels can be used in your programs, from the Command window, or in macros. They are also displayed in the I/O Monitor window.

To open the I/O Label Editor

Select I/O Label Editor from the Tools Menu.

> Or

Type Ctrl + L.

> Or

Click on the ▦ button on the toolbar.
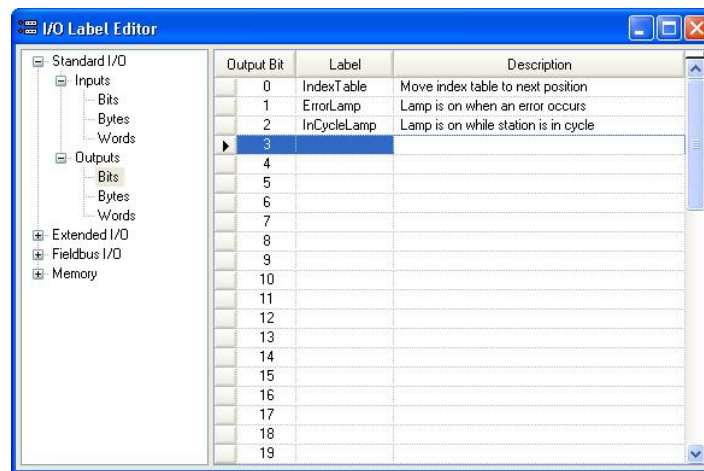
The I/O Label Spreadsheet

When you select I/O Label Editor from the Tools Menu, a window opens that contains a tree and a spreadsheet editor.

The tree on the left side of the window shows the various types of I/O for the controller. For each type of I/O you can view and edit labels for bits, bytes (8 bits), and words (16 bits).

The first column of the spreadsheet shows the bit, byte, or word number, depending on which type of I/O you are viewing.

The second column contains the label for each bit, byte, or word in column 1. You can type in up to 16 characters for a label. Label characters can be alphanumeric or underscore.

The third column contains the description associated with the label.

If you add a description to an I/O point, then the description will be displayed as a tool tip on the I/O Monitor.

NOTE
☞
All I/O types are shown in the I/O Label Editor, but some may not be available on your controller. For example, you can edit Fieldbus I/O labels, but you may not have a Fieldbus board installed in the controller. This allows you to edit labels when you are disconnected from a controller. Verify that the controller to be used with the project has the hardware installed for the I/O you are using.

To add or edit a label

Select the type of I/O you want to label in the tree. After you select the I/O type, the spreadsheet will be refreshed to display the labels for that type. The number of rows in the spreadsheet equals the number of bits, bytes, or words available for the type you have selected.

Use the mouse to scroll through the spreadsheet and put the cursor in the Label field next to the bit, bytes, or words number that you want to add a label to. Type in the label, which can be up to 16 alphanumeric characters without any spaces. Optionally, you can type a description for the label in the Description field.

After adding or editing labels, save the changes by executing Save from the File Menu or by clicking on the Save Project ![icon] toolbar button. If any duplicate labels are detected, an error message will be displayed and the save operation will be aborted. You must correct the duplication before you can save the labels successfully.

To cut and paste labels and descriptions

You can cut and paste labels and descriptions by selecting with the mouse, then executing Copy, Cut, and Paste from the Edit Menu.

You can also cut and paste entire rows using the following steps:

1. Select one or more rows by using the row selectors on the left and execute either the Cut or Copy command from the Edit Menu. When selecting multiple rows, hold down the shift or control key while selecting rows with the mouse.

2. Select the destination I/O type, if required.

3. Select the row where you want to start the paste by clicking the row selector on the left of the row.

4. Execute the Paste command from the Edit Menu.

5.11.4 I/O Monitor Command (Tools Menu)

The I/O Monitor window lets you monitor all controller hardware inputs and outputs and also memory I/O. There are up to four views available: one standard view and three custom views.

On the standard view, there are two grids. For each grid you can specify which type and size of I/O to monitor.

For each custom view, you can specify a list of any combination of input, output, or memory. By default, there is one custom view available. To use the other two custom views, right click on a tab and check the views you want to be visible. See the section *Custom I/O Views* later in this chapter.

Labels that have been defined using the I/O Label Editor are displayed next to each bit, byte, or word.

After the I/O Monitor window has been opened, the input and output status for the current view is constantly updated.

The I/O monitor will always be displayed on top of other child windows, such as program windows and point windows.

If a description has been entered for an I/O port (bit, byte, or word) in the I/O Label Editor, then a tool tip will be displayed when the mouse pointer is over the row containing the port.

You can turn outputs on and off by double clicking on the output LED images in the Status column.
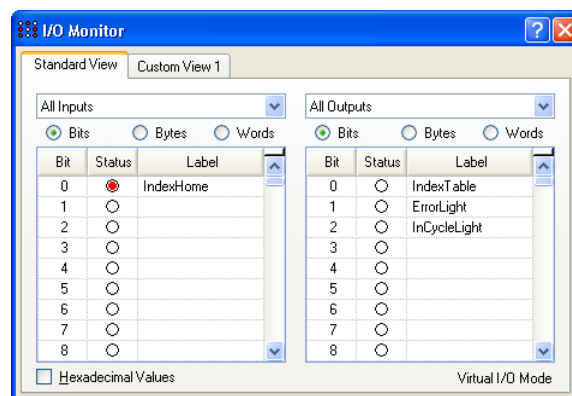
To open the I/O Monitor

Select I/O Monitor from the Tools Menu.
> Or

Click on the ⠿ tool bar button.
> Or

Type Ctrl + I.

Using the I/O Monitor

Select the tab for the type of I/O you want to monitor.

Scroll through the grids to locate the desired inputs or outputs to monitor.

You can split each grid into two scroll regions by selecting the split bar in the upper right corner of the grid and dragging it down. Each scroll region can be individually scrolled.

To turn an output off or on, double click on the LED image for the desired output.

When Virtual I/O is active, you can turn input bits on and off by double clicking on the input LED images in the Status column.

To view bytes and words in hexadecimal format, check the **Hexadecimal Values** checkbox.

You can resize the I/O monitor in the vertical direction to show more data. Move your mouse pointer to the lower right corner of the window to activate a size handle, then click down and drag the window down or up to the desired size.

Custom I/O Views

You can configure up to three custom I/O views. In each view, you can add any combination of I/O. You can also change the name of each view and hide each view.
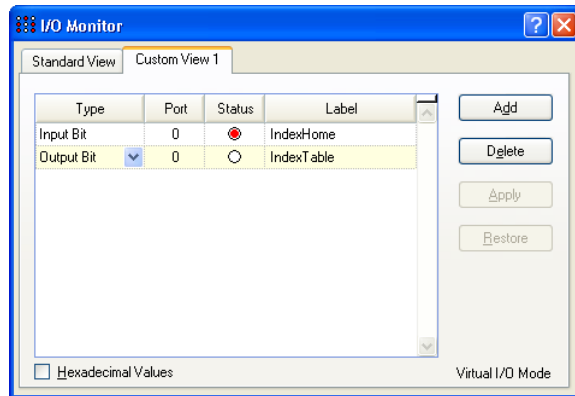
To change a view

1. Click on a custom view tab. If none are currently shown, right click on the **Standard View** tab and select one of the three custom views to show it.
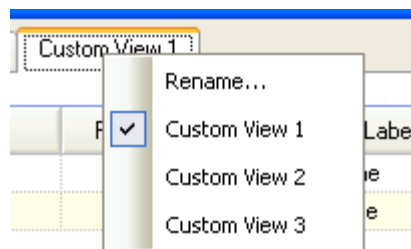
2. Click the **Add** button to add a new row to the list.

3. Select the **Type** by clicking in the **Type** column, then click the arrow to view a list of I/O types and select one.

4. In the **Port** column, select the port (bit, byte, or word, depending on I/O type).

5. Add more rows as needed by repeating steps 2 - 4.

6. Save the changes by clicking the **Apply** button.

7. Click the **Delete** button to delete a row.

8. Use the **Restore** button to cancel changes.

To rename a view

1. Click on a custom view tab. If none are currently shown, right click on the **Standard View** tab and select one of the three custom views to show it.

2. Right click on the view tab and select **Rename**.

3. Enter the new name for the view.

5.11.5 Command Window Command (Tools Menu)

The command window emulates an ASCII terminal that communicates directly with the robot controller. You can execute SPEL+ immediate commands from this window and view the results.

To open the Command window

Select Command Window from the Tools Menu

> Or

Click on the ▷ button on the toolbar.

> Or

Type Ctrl+M
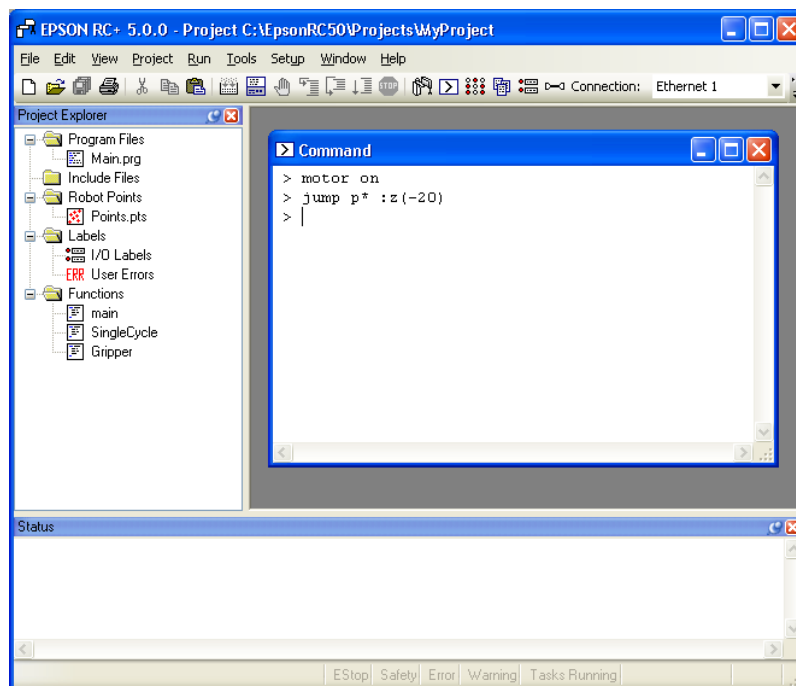
To execute SPEL+ commands from the Command window

1. Type in the desired command after the prompt (>). Commands can be entered in upper or lower case.

2. Press the Enter key to execute the command. The cursor can be anywhere on the line when you press Enter.

3. Wait for the prompt to return before typing in a new command.

When an error occurs, an error number will be displayed along with an error message.

You can use the arrow keys or the mouse to move the cursor to any line in the window that starts with a prompt (>) character and execute it by pressing Enter.

Command Window Keys

Key	Action
Ctrl+A	Select entire window.
Ctrl+C	Stop the program and initialize robot controller. If a robot motion command is in progress, the prompt will return when the command has been completed.

Key	Action
Ctrl+V	Execute Paste command. Paste from Clipboard to current selection.
Ctrl+W	Re-display last command line after the prompt.
Ctrl+X	Execute Cut command. Cut current selection and put in Clipboard.
Ctrl+Z	Undo last change.
Ctrl+Home	Go to the top of the window.
Ctrl+End	Go to last prompt at end of the window.
?	Translates to "PRINT " when used as the first character of a command. This can be used to display variables or any statement that requires a PRINT command.

5.11.6 Macros Command (Tools Menu)

You can create SPEL+ command macros using the Macro Editor. Macros consist of one or more SPEL+ statements that can be executed from the command window. A macro statement may use global variables, I/O labels, and point labels. You can assign a macro to each of the Alt function keys except for Alt+F4, which is a Windows shortcut to close the application.

1. Select Tools | Macros to open the Command Macros dialog.
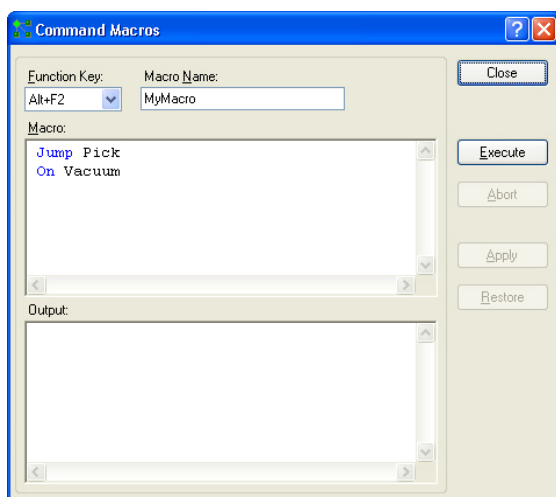
2. Type the macro statements in the **Macro** text box.

3. Click the **Apply** button to save changes.

4. Click **Execute** to run the macro.

5. Click **Close** to close the dialog. You will be prompted to save the macros you have created or changed.

To open a macro and execute it, type Alt+function key to open the macro. Then click **Execute** to run it. Macros never execute by pressing the function key. The separate execute step is provided for safety, since macros can move the robot and control I/O.

Macros can be executed while tasks are running. If one or more commands in a macro are not valid while tasks are running, an error will occur.

5.11.7 Controller Command (Tools Menu)

Select Controller from the Tools Menu to open the Controller Tools dialog.

From the Controller Tools dialog, you can save and restore the complete controller configuration and the project using the Backup Controller and Restore Controller commands. You can also save and view controller status, and reset the controller.

Before servicing the system, you should execute Backup Controller and store the system configuration on an external media such as a USB memory key.

If required, you can use Restore Controller to restore previously stored data.
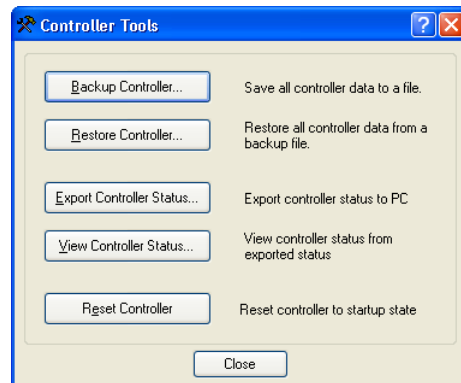
Backup Controller
Use Backup Controller to save controller configuration data on your PC. You cannot backup the controller data while tasks are running. If you attempt to do so, an error message will be displayed.

1. Select Tools | Controller.

2. Click on the **Backup Controller** button to open the **Browse For Folder** dialog.
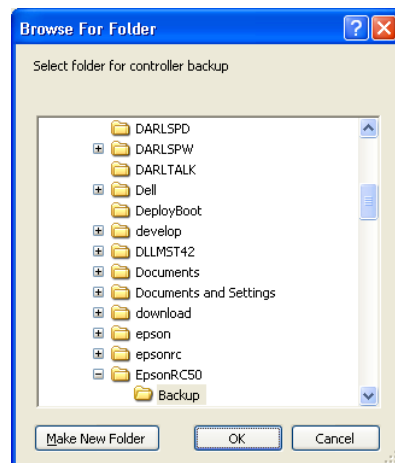
3. Select the disk drive and parent folder where you want to save the information. You can create a new parent folder by clicking the **Make New Folder** button.

4. Click **OK**. A new folder containing the backup files will be created in the selected folder named "B_" plus the controller serial number followed by the current date / time.

NOTE
☞
When the version of the connected RC180 controller firmware is Ver.1.0.*.*, Backup Controller does not backup the project data.
Be sure to backup the project data individually by selecting Copy from the Project Menu.

Restore Controller

Use Restore Controller to load controller settings from previously saved backup data. You cannot restore the controller data while tasks are running. If you attempt to do so, an error message will be displayed.

To restore controller configuration:

1. Select Tools | Controller.

2. Click on the **Restore Controller** button to open the **Browse For Folder** dialog.
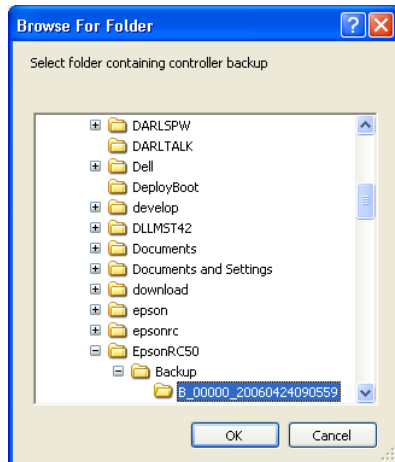
3. Select the drive and folder where the information is stored. Controller backup information is stored in a folder that is named "B" plus the controller serial number followed by the date / time.

NOTE

You can also select a folder containing export controller status information.
Export controller status is stored in a folder that is named "S" plus the controller serial number followed by the date / time.

4. Click **OK** to display the dialog to select the restore data.

Robot name, serial #, calibration Checkbox
This checkbox allows you to restore the robot name, robot serial number, Hofs data, and CalPls data. Make sure that the correct Hofs data is restored. If the wrong Hofs data is restored, the robot may move to wrong positions.
The default setting is unchecked.

Project Checkbox
This checkbox allows you to restore the files related to projects.
The default setting is unchecked.
When the project is restored, all the values of Global Preserve variables are restored.
For details about Global Preserve variable backup, refer to *5.10.10 Display Variables Command (Run Menu)*.

NOTE

When the version of the connected RC180 controller firmware is Ver.1.0.*.*, this dialog does not appear. Robot name, serial number, and calibration data is always restored.
The project is not restored.

5. Click the <OK> button to restore the system information.

NOTE
☞ Restore the system configuration saved using Backup Controller only for the same system. When different system information is restored, the following warning message appears.
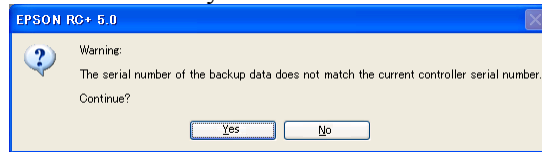
Click the **No** button cancel restoration of data except for special situations such as controller replacement.

Export Controller Status

You can export the current status of the controller using this button. Tasks can be running when this command is executed. The current status is saved in a folder containing several files. The controller configuration settings, task status, I/O status, robot status, etc. are saved in these files. This is useful for an end user to send a snapshot of the controller status to a system vendor or to Epson technical support, should the need arise.

NOTE
☞ Export Controller Status is equivalent to connecting a USB memory key to the controller and pressing the TRIG button on the controller to save controller status.

You can configure the controller whether to save the project files in the status folder or not. See Setup | Controller | Preferences.

You can view the exported status using the **View Controller Status** button (see next section).

To export controller status
1. Select Tools | Controller.

2. Click on the **Export Controller Status** button to open the **Browse For Folder** dialog.
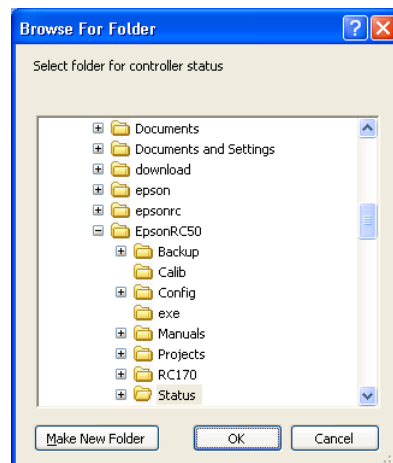
3. Select the drive and parent folder where the information is to be saved.

4. Click **OK** to save the current controller status. A new folder will be created inside the parent folder using the controller serial number and the current date / time.

View Controller Status

Click the **View Controller Status** button to view the status data stored from a previous status export (see the Export Controller Status section above).

To view controller status:

1. Select Tools | Controller.

2. Click on the **View Controller Status** button to open the **Browse For Folder** dialog.
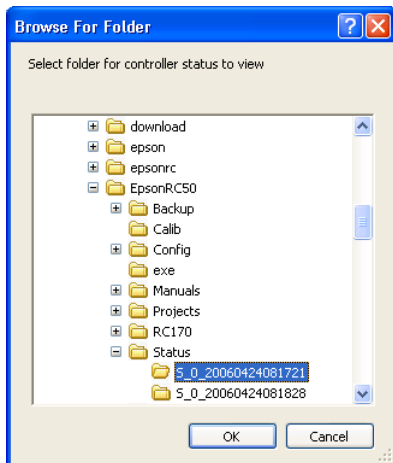
3. Select the drive and folder where the information is stored. Controller status information is stored in a folder that is named "S_" plus the controller serial number followed by the date / time.

4. Click **OK** to view the selected controller status.

5. The Controller Status Viewer dialog will be displayed.
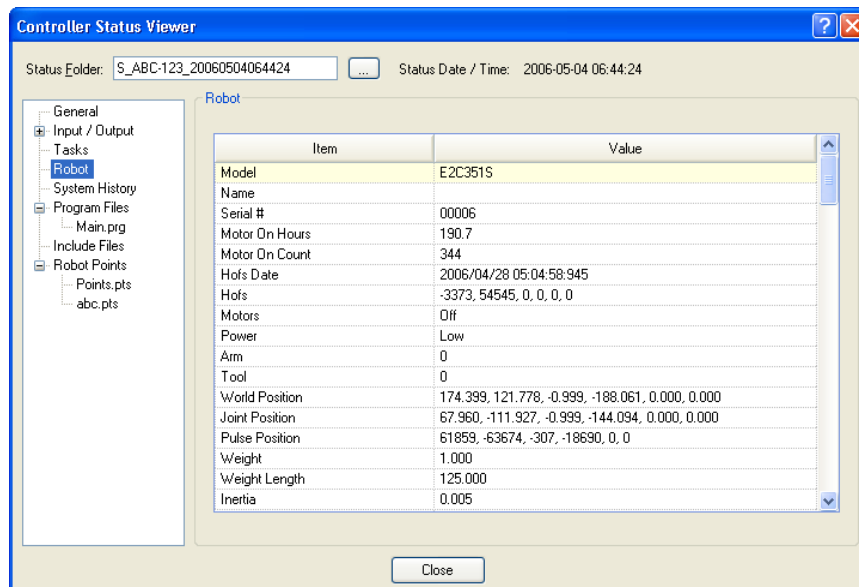
6. Select items to view from the tree on the left side of the dialog.

7. To view another controller status, click the ellipses button next to the Status Folder name and select a new status folder.

Reset Controller

Use **Reset Controller** to reset the current controller from within EPSON RC+ 5.0. This is equivalent to powering the controller off and on. It is useful for when you want to reset the controller state remotely because you do not have easy access to the controller power switch. For example, a development team could be using a controller for development by connecting to it using Ethernet. By executing this command, a team member can reset the controller from his development system.

5.11.8 User Error Editor Command (Tools Menu)

The User Error Editor allows you to define user errors for the current project. If a user error is caught in an error handler, then the ErrMsg$ function will return the error message defined in the editor. The label associated with an error can be used in any integer expression.
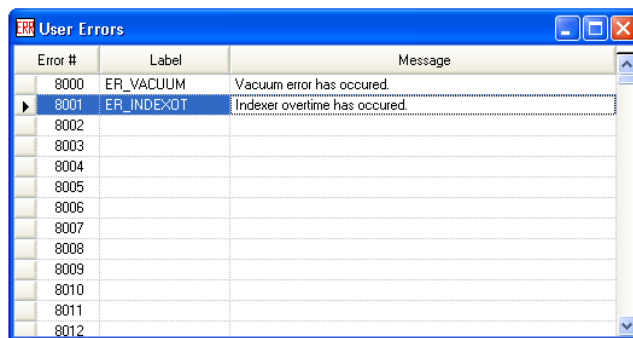
User error numbers can be from 8000 to 8999.

Labels can be up to 16 characters in length.

It is recommended that you use the ER_ prefix for each error label and use all caps for the label. This makes it easy to see error labels in your code.

Some user error examples:

Error #	Label	Message
8000	ER_VACUUM	Vacuum error has occurred.
8001	ER_INDEXOT	Indexer overtime has occurred.

In your program code, use the Error statement to generate a user error. For example:

```
On Vacuum
Wait Sw(VacOn), 1
If Tw = 1 Then
   Error ER_VACUUM
EndIf
```

The user error information is stored in the current project directory in a file called UserErrors.dat.

You can use the Import command from the File Menu to import user errors from other projects.

After adding new error definitions, save the changes by executing Save from the File Menu or by clicking on the Save all files 🖫 toolbar button. If any duplicate labels are detected, an error message will be displayed and the save operation will be aborted. You must correct the duplication before you can save the labels successfully.

5.11.9 Simulator Command (Tools Menu)

This command opens the Simulator window. For instructions on using the Simulator, see chapter *8. Simulator*.

Shortcuts

Toolbar:

5.11.10 Vision Command (Tools Menu)

This command opens the Vision Guide window. For instructions on using Vision Guide, refer to the *EPSON RC+ 5.0 Option Vision Guide 5.0 Manual.*

Shortcuts

Toolbar:

5.12 Setup Menu

The Setup Menu contains the following commands:

- PC to Controller Communications

- Controller

- Vision

- Preferences

For details on the Vision command, refer to *EPSON RC+ 5.0 Option Vision Guide 5.0.*
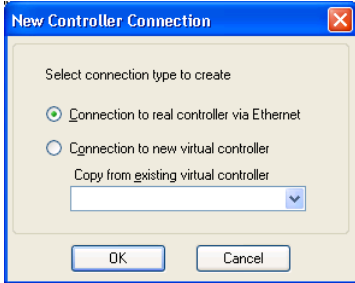
5.12.1 PC to Controller Communications Command (Setup Menu)

The PC to Controller Communications command opens a dialog box that is used for configuring communications between your PC and the Controller.

Shortcuts

Toolbar:
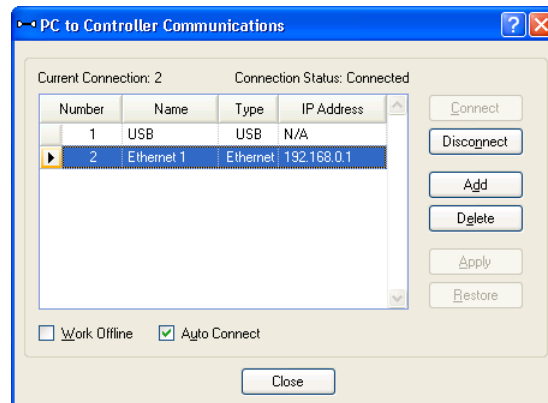
Item	Description
Connect	Connect to the currently selected connection.
Disconnect	Disconnect from the current connection.
Add	Add a new Ethernet or virtual controller connection. When you click this button, another dialog will open that allows you to specify the connection type.
Delete	Delete the currently selected connection. You cannot delete connection #1, USB.
Apply	Save the current changes.
Restore	Revert back to the previous settings.
Work Offline	When Work Offline is checked, EPSON RC+ 5.0 does not attempt to connect to a controller during operations such as building a project. Some windows such as Robot Manager will not be available while working offline.
Auto Connect	When Auto Connect is selected, EPSON RC+ 5.0 will automatically connect if the current connection is available.
Close	Close the dialog.

5.12.2 Controller Command (Setup Menu)

The Controller command opens the Setup Controller dialog. This dialog has several pages that can be accessed by selecting them from the tree on the left side of the dialog.

Setup: Controller: General Page
This page allows the user to view general information about the controller.
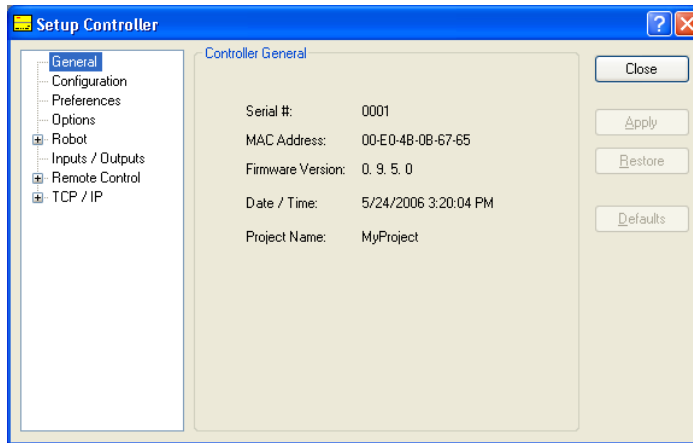
Item	Description
Serial #	Displays the serial number of the current controller.
MAC Address	Displays the MAC Address of the controller.
Firmware Version	Displays the firmware version used in the current controller.
Date / Time	Displays the current date and time in the controller.
Project Name	Displays the name of the project in the controller.
Close	Close the Setup Controller dialog.

Setup: Controller: Configuration Page

This page allows the user to view and change the controller configuration settings.
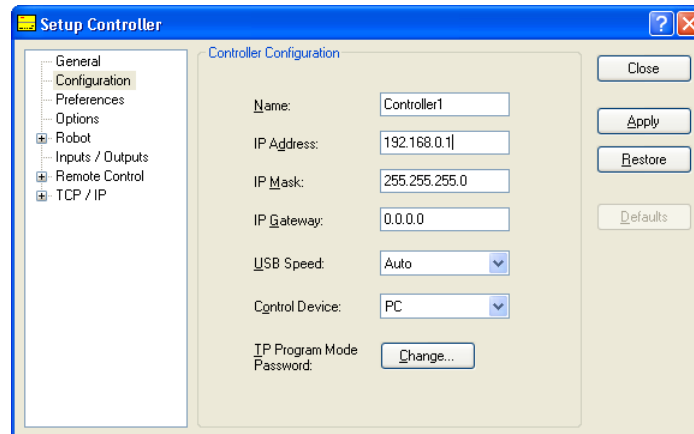
Item	Description
Name	Use this text box to change the controller name. You may use any name up to 16 characters long using alphanumeric characters and underscore.
IP Address	Use this text box to set current IP address of the controller. The IP Address must be on the same subnet as the PC.
IP Mask	Use this text box to set the IP mask of the controller. Note that the IP Mask must match the IP mask used for your network.
IP Gateway	Use this text box to set the IP gateway of the controller. This is only required if you will be accessing the controller from outside of the local network.
USB Speed	Select the speed of the USB connection.
Control Device	Allows you to select the Control Device.
TP Program Mode Password	Allows you to change the TP Program Mode password.
Close	Close the Setup Controller dialog.
Apply	Save the current changes. If necessary, the controller will be reset to use the new settings.
Restore	Revert back to previous settings.

Setup: Controller: Preferences Page

This page contains controller preference settings.
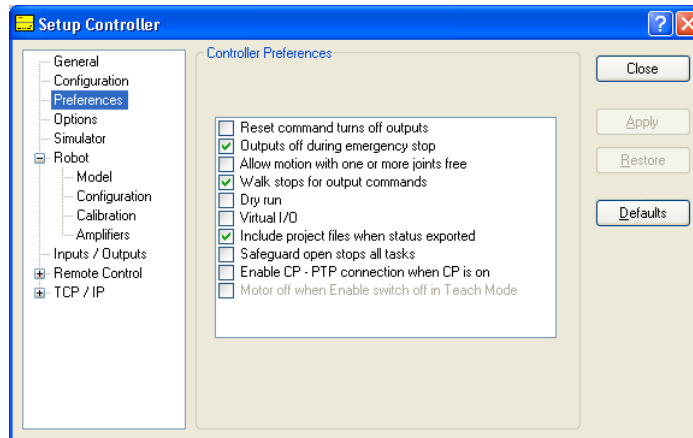
RESET command turns off outputs

When this preference is turned on, all outputs other than remote control outputs will be turned off when a Reset instruction is executed. The default setting is off.

NOTE
☞
The outputs of the standard I/O, expansion I/O, and Fieldbus I/O are included in the "outputs" mentioned in the above preferences **RESET command turns off outputs** and **Outputs off during Emergency Stop**. Memory I/O is not affected by these preferences. Therefore, memory I/O bits are not turned off by the RESET command execution or during Emergency Stop.

Outputs off during Emergency Stop

When this preference is turned on, all outputs other than remote control outputs will be turned off when emergency stop occurs. Also, no outputs can be turned on until the emergency stop condition is cleared. The default setting is on.

NOTE
☞
You should design your system to always remove all power to output devices when emergency stop occurs. Even if the controller turns off outputs, the I/O hardware could malfunction.

Allow motion with one or more joints free

When this preference is turned on, motion commands can be executed after SFree has been used to free one or more joints. The default setting is off.

Walk stops for output commands

When checked, the Walk command from the Run Menu will execute lines until after the next motion or output statement (whichever comes first). When unchecked, the Walk command will execute lines until after the next motion statement and will not stop for output statements. The default setting is on.

Dry run

This preference allows you to run programs without a robot connected to the controller. All program statements will work. Motion statements will execute approximately the same amount of time as when connected to a robot.

Virtual I/O

This preference allows you to run programs using virtual I/O. When Virtual I/O is enabled, I/O commands do not affect the hardware I/O. There are also several commands available for turning on inputs from within a program. The default setting is off.

NOTE
☞
Remote function is also available when virtual I/O is enabled.

Include project files in when status exported

This preference allows you to configure whether project files are included or not when the controller status is exported. See Tools | Controller for information on exporting and viewing controller status. The default setting is on.

NOTE
☞

The project using Vision Guide cannot perfectly be stored by the Status storage. This is because some files are not transferred to the Controller.
If you are using Vision Guide, please back up the project files with RC+.

Safeguard open stops all tasks

Check this option to cause all tasks to stop when the safeguard is open.
This option can be used in applications where pause / continue are not required.
The default setting is off.

NOTE
☞

When the version of the connected RC180 controller firmware is Ver.1.0.*.*, this check box cannot be checked.

Enable CP – PTP connection when CP is on

This preference allows you to overlap the trajectories of CP motion and PTP motion during CP ON.

NOTE
☞

Over-speed error or Over-acceleration-speed error may occur according to the motion acceleration / deceleration speed setting. If the error occurs, adjust the acceleration / deceleration speed setting or uncheck this checkbox.

Auto LJM (Least Joint Motion)

This preference allows you to enable Auto LJM at the controller start up. To disable Auto LJM temporarily, use AutoLJM Off command.

NOTE
☞

If Auto LJM is enabled at all times, this function automatically adjusts the posture of the manipulator to reduce the motion distance, even when you intended to move the joint widely. Therefore, it is recommended to disable Auto LJM at the controller start up and operate the manipulator as you desired using AutoLJM On command or LJM function.

Disable Point flag check

This preference allows you to continue operation even when point flags, one was specified as a target point and the other one after the motion completion, do not match in a CP motion.

However, if the flags do not match at the transferring point while CP On is used, the manipulator will stop at the point and the motion will not become a path motion.

Motor off when Enable switch off in Teach Mode

This preference is read-only. It shows whether motors will be turned off when the Enable switch is off during Teach Mode.

Setup: Controller: Options

This page is used to configure options in the controller. For instructions, refer to *15. Installing Controller Options.*

Setup: Controller: Simulator

This page is used to configure the Simulator. For virtual controllers, the Enable Simulator checkbox is grayed out and checked. For real controllers, if the firmware supports simulation, then you can check the Enable Simulator checkbox.
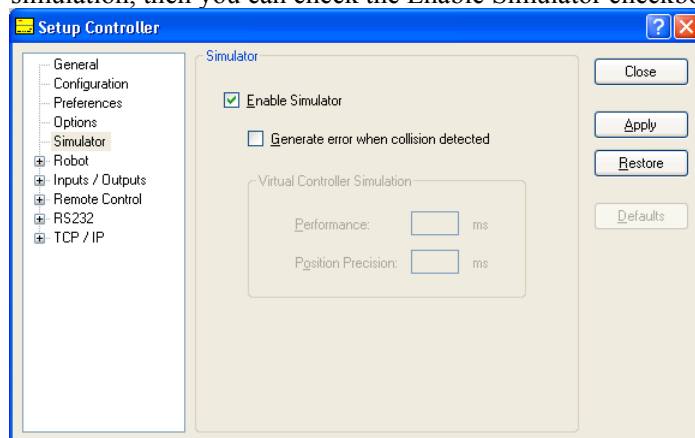
For more details, see *Virtual Controller Configuration* in section 8.3.4.

Setup: Controller: Robot

Setup: Controller: Robot Model Page

Use this page the set the model for the robot connected to the controller.

Item	Description
Model	Allows the user to select the robot model.
	You can only select a model that the current amplifier power ratings will allow.
	If Dry Run is active, or the controller is a virtual controller, then all robot models will be displayed.
	If you select "None", then the current robot will be removed from the controller when Apply is clicked.
Type	Displays the robot type.
Length	Displays the robot length (J1 + J2 for SCARA robots) or reach for 6-axis robots.
Apply	Save the current changes.
Restore	Revert back to previous settings.
Close	Close the Setup Controller dialog.

Setup: Controller: Robot Configuration Page

Item	Description
Name	Enter a Name for the robot.
Serial #	Enter the Serial number of the robot.
Joints	These checkboxes determine if the respective joint is enabled or disabled.
Close	Close the Setup Controller dialog.
Apply	Save the current changes.
Restore	Revert back to previous settings.

Setup: Controller: Robot Calibration Page
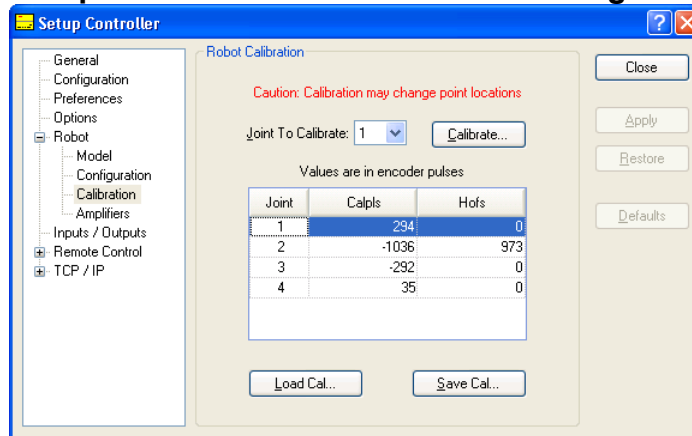
You can calibrate each joint of the robot from this page. It is not displayed if the controller is virtual.

Item	Description
Joint to Calibrate	Select the joint that you want to Calibrate.
Calibrate…	This button starts the Calibration Wizard dialog that walks you through the calibration process.
Calpls	These are the Calpls settings for each joint. Normally, the calibration wizard will calculate these values.
Hofs	These are the Hofs settings for each joint. Normally, the calibration wizard will calculate these values.
Load Cal	This button is used to load data from a previously save calibration file. After the data is loaded, the grid will be refreshed to show the values.
Save Cal	This button is used to save the calibration data to a calibration file.
Apply	Save the current changes.
Restore	Revert back to previous settings.
Close	Close the Setup Controller dialog.

Setup: Controller: Robot Amplifiers Page

This page shows the power values for the motor amplifiers installed in the controller. It is not displayed if the controller is virtual.
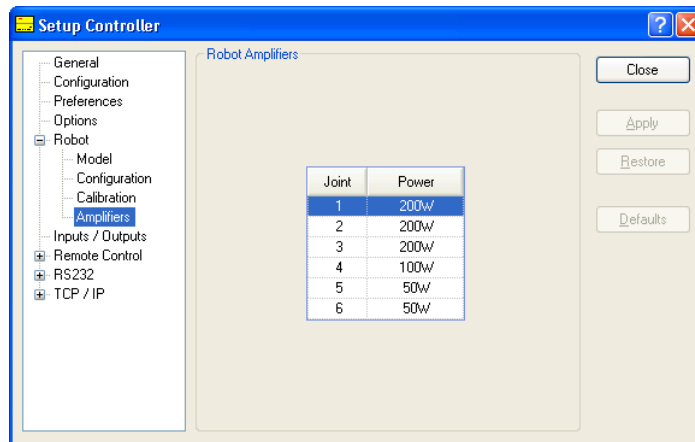
Item	Description
Robot Amplifiers	This shows the power for each robot amplifier currently in the controller.
Close	Close the Setup Controller dialog.

The motor amplifiers and the motor powers may not be the same.

Setup: Controller: Inputs / Outputs Page

This page shows the I/O hardware installed in the controller. There are no settings to configure.
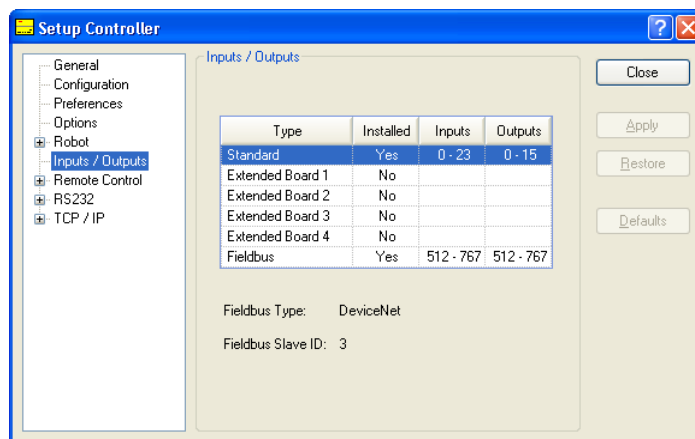
Setup: Controller: EtherNet/IP Page

If an EtherNet/IP board is installed in the controller, a configuration page is shown under Inputs / Outputs. For details on EtherNet/IP configuration, refer to the controller manual.

Setup: Controller: Remote Control

For details of Remote function, refer to *10. Remote Control*.

Setup: Controller: Remote Control Inputs Page

Use this page to configure the controller remote control inputs.
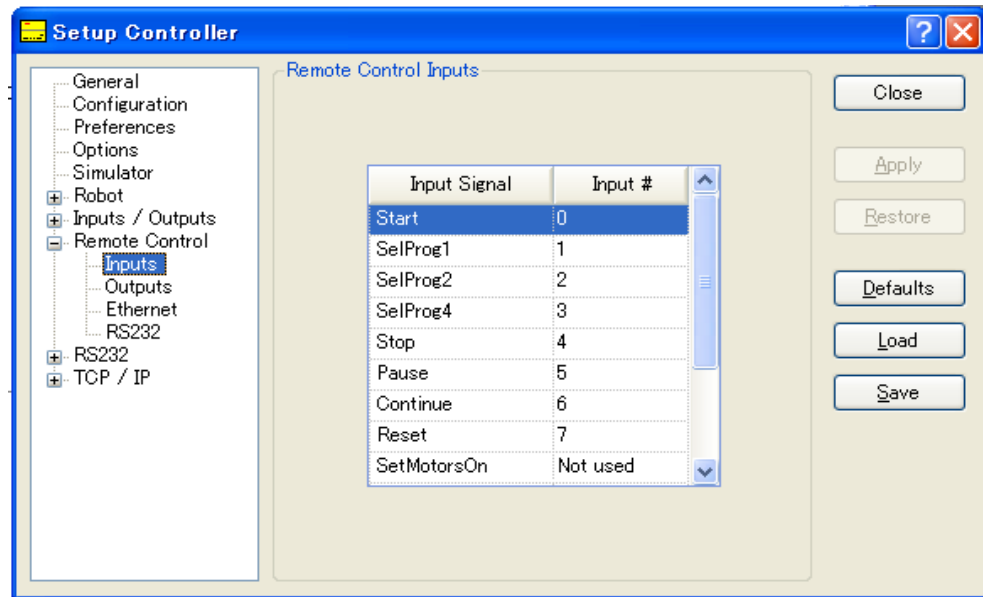
Item	Description
Input #	Select an input bit to use for the corresponding input signal. Select "Not used" to disable the remote input.
	For example, if "Start" is assigned to I/O input bit 0, select "Not used" to use this as a normal I/O input.

Item	Description
Close	Close the Setup Controller dialog.
Apply	Save the current changes.
Restore	Revert back to previous settings.
Defaults	Click this button to set the default remote inputs. First, a dialog box will be displayed asking you which type of inputs to use for defaults: Standard I/O or Fieldbus. You can also select Clear All to set all remote inputs to Not used.
Load	Read the assigned remote inputs and outputs from a file on the PC and save it in the controller.
Save	Save the assigned remote inputs and outputs shown in the dialog to a file on the PC.

NOTE

☞ Both the remote inputs and outputs are loaded or saved together when using **Load** or **Save**.

Setup: Controller: Remote Control Outputs Page

Use this page to configure the controller remote control outputs.
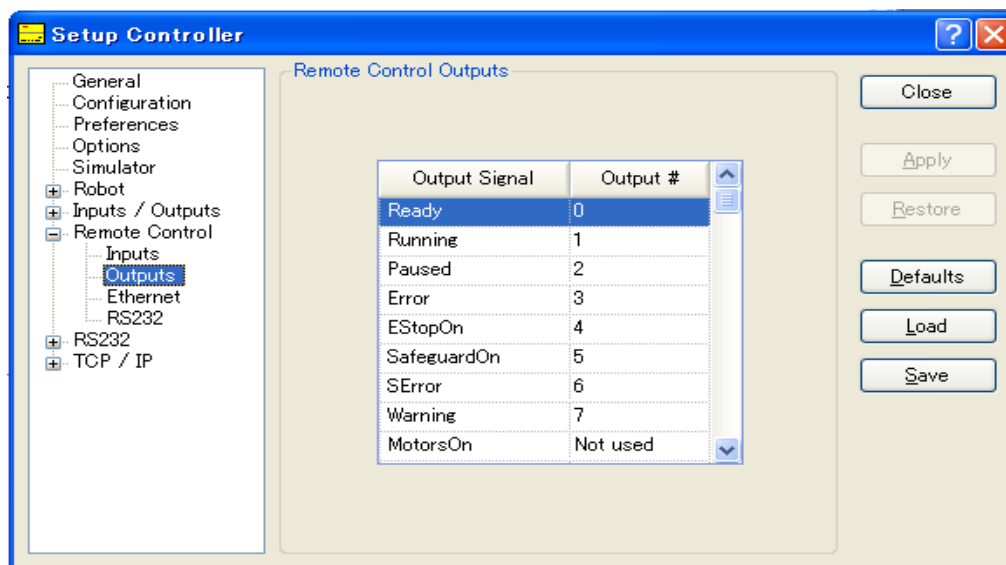
Item	Description
Output #	Select an output bit to use for the corresponding output signal. Select "Not used" to disable the remote output. For example, if "Ready" is assigned to I/O output bit 0, select "Not used" to use this as a normal I/O output.
Close	Close the Setup Controller dialog.
Apply	Save the current changes.
Restore	Revert back to previous settings.
Defaults	Click this button to set the default remote outputs. First, a dialog box will be displayed asking you which type of outputs to use for defaults: Standard I/O or Fieldbus. You can also select Clear All to set all remote outputs to Not used.
Load	Read the assigned remote inputs and outputs from a file on the PC and save it in the controller.
Save	Save the assigned remote inputs and outputs shown in the dialog to a file on the PC.

NOTE

☞ Both the remote inputs and outputs are loaded or saved together when using **Load** or **Save**.

Setup: Controller: Remote: Ethernet

Use this page to configure the Remote Ethernet.

These settings are necessary when the control device is "Remote Ethernet", or when the operator acquires the value using the Remote Ethernet while the control device is not "Remote Ethernet".
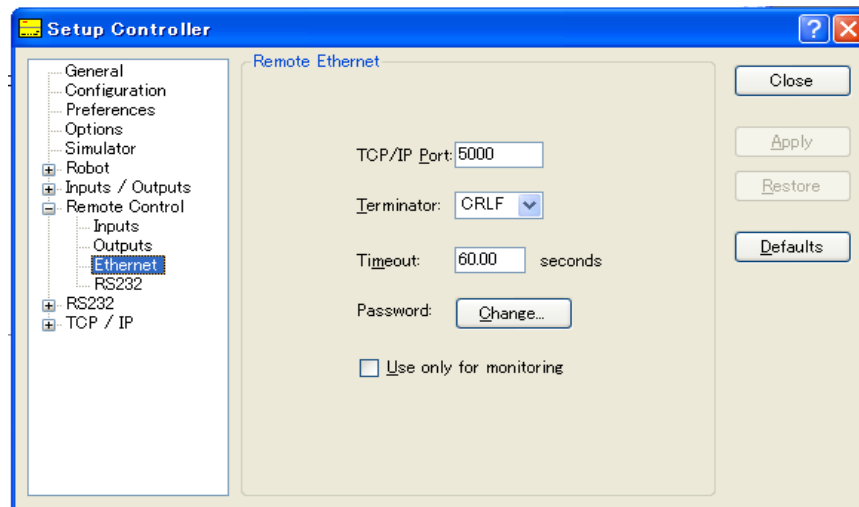
Item	Description
TCP/IP port	Input the port number used by the Remote Ethernet client. Select a port that is not used for other TCP/IP communications.
Terminator	Select the code that indicates the line terminator (end-of-data) for the Remote Ethernet client.
Timeout	Specify the longest interval time for a remote command during Remote Ethernet communications (in units of seconds). If there is no remote command from the client during the time set in [Timeout], task execution stops with a communication error.
Password	Specify a password for client authentication to connect. Authentication is accomplished with the Login command.
Use only for monitoring	Specify to acquire the value using Remote Ethernet, if the control device is not "Remote Ethernet".
Close	Close the dialog.
Apply	Save the current changes.
Restore	Revert back to the previous settings.
Defaults	Click this button to set the Remote Ethernet default settings.

Default setting
TCP/IP port No.   : 5000
Terminator   : CRLF
Timeout   : 60 seconds
Password   : No setting
Use only for monitoring   : Not checked

⚠ CAUTION	■ If you set "0" in the [Timeout] textbox, then the time out duration is infinite. In this case, the task(s) continue to execute even without the communication from client. This means the robot may keep moving and cause unexpected damage. Ensure that tasks will be stopped appropriately in your system.

Setup: Controller: Remote: RS232

Use this page to configure the Remote RS232.

When the Remote RS-232C port is not installed to the Controller, RS232 is not shown in the tree.

These settings are necessary when the control device is "Remote Ethernet", or when the operator acquires the value using the Remote Ethernet while the control device is not "Remote Ethernet".

Item	Description
Port	Specify a port No. used for remote RS232 communication.
	For example, RS-232C port of the Controller is allotted to the Remote RS232 connection in the above dialog.
	Specify the port No. that is not used for other RS-232C communication.
	RS-232C port setting is the value specified at Controller | RS-232C | Specified port No.
Password	Specify a password for client authentication to connect. Authentication is accomplished with the Login command.
Use only for monitoring	Select to acquire the value using Remote Ethernet, if Control device is not Remote Ethernet.
Close	Close the dialog.
Apply	Save the current changes.
Restore	Revert back to the previous settings.
Defaults	Click this button to set the Remote RS232 default settings.
	Default setting RS-232C port No. : 1 Password : No setting Use for monitoring : No

---

CAUTION

■ If you set "0" in the [Timeout] textbox of the specified RC-232C port, then the time out duration is infinite. In this case, the task(s) continue to execute even without the communication from client. This means the robot may keep moving and cause unexpected damage. Ensure that tasks will be stopped appropriately in your system.

Setup: Controller: RS232 Pages

There is one page for each RS232 port installed in the controller. If there are no RS232 ports installed in the controller, then no RS232 selections are visible in the tree.
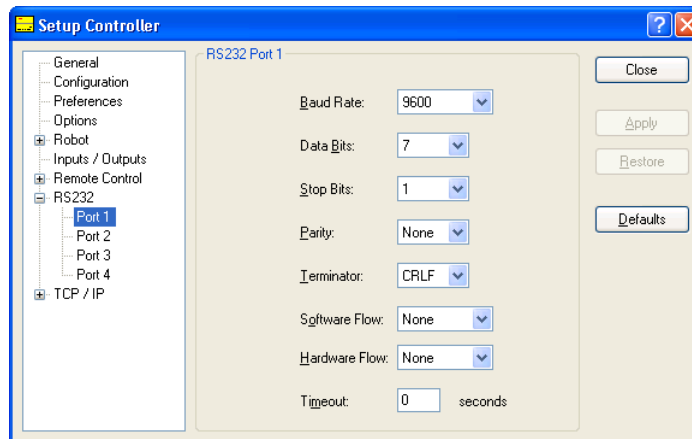
To configure an RS-232 port

1. Select Controller from the Setup Menu and select the page for the RS232 port you want to configure.

2. Change the settings as desired. You can also click the **Defaults** button to use default settings.

3. Click **Apply** to save the new settings and click **Close**.

Setup: Controller: TCP/IP Pages

You can configure TCP/IP settings for the robot controller in a SPEL+ program using the SetNet command. You can also configure settings from the TCP/IP tab on the Setup | Controller dialog.
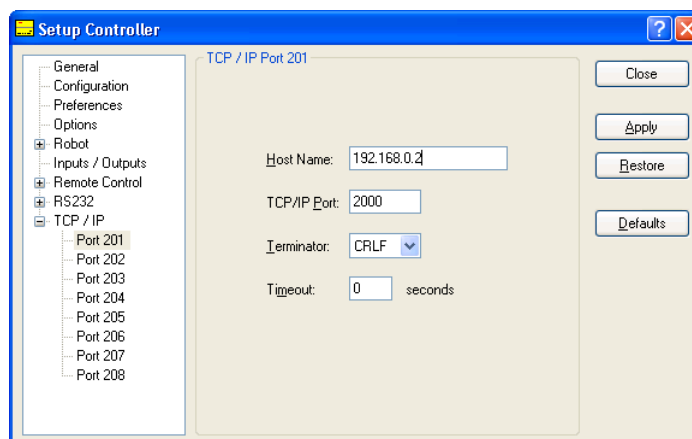
To configure a TCP/IP port

1. Select Controller from the Setup Menu and select the page for the TCP/IP port you want to configure.

2. Enter the host name or IP address for the controller or PC that you want this controller to communicate with.

NOTE
☞   The controller does not support DNS, so you must specify an IP address for the host you are communicating with. You cannot specify a name for the host.

3. Enter the TCP/IP port number. This must be the same port number that is used on the host device. It must be different from any of the other TCP/IP port numbers used for the other TCP/IP ports.

4. Change the other settings as desired.

5. Click **Apply** to save the new settings and click **Close**.

5.12.3 Preferences Command (Setup Menu)

The Preferences command opens a dialog that contains several pages that are used to configure user preferences for the EPSON RC+ 5.0 environment.

To open the Preferences dialog, select Setup | Preferences

Setup: Preferences: Startup

Setup: Preferences: Startup: Start Mode Page

From the Start Mode page, you can choose whether EPSON RC+ 5.0 starts in Auto mode or Program mode.
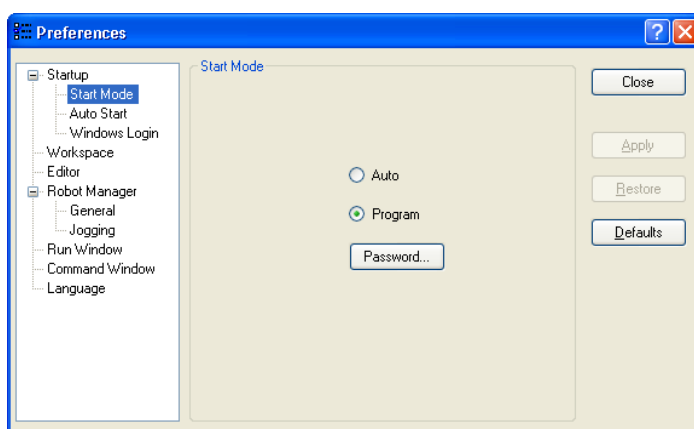
Item	Description
Auto	Select Auto to start EPSON RC+ 5.0 in Auto mode. See the Operation chapter for details.
Program	Select Program to start EPSON RC+ 5.0 in Program mode. See the Operation chapter for details.
Password	Click this button the change the password required to enter Program mode from Auto mode when EPSON RC+ 5.0 starts.
Apply	Save the current changes.
Restore	Revert back to previous settings.
Defaults	Click this button to set the default startup mode.
Close	Close the Preferences dialog.

Setup: Preferences: Startup: Auto Start Page
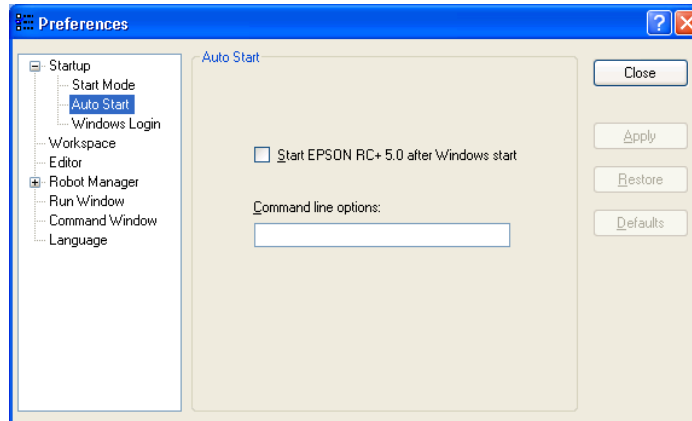
Item	Description
Start EPSON RC+ after Windows start	Check this box if you want EPSON RC+ 5.0 to automatically start after Windows starts.
Command line options	Enter the command line options used when EPSON RC+ 5.0 is automatically started. This has no effect if the Start EPSON RC+ 5.0 with Windows start checkbox is not checked.
Apply	Save the current changes.
Restore	Revert back to previous settings.
Close	Close the Preferences dialog.

Setup: Preferences: Startup: Windows Login Page

The Windows Login page allows you to configure your PC to automatically login a user when Windows starts. This is useful when the PC is used to run EPSON RC+ 5.0 in Auto mode and display an operator window. The operator does not have to login to Windows.
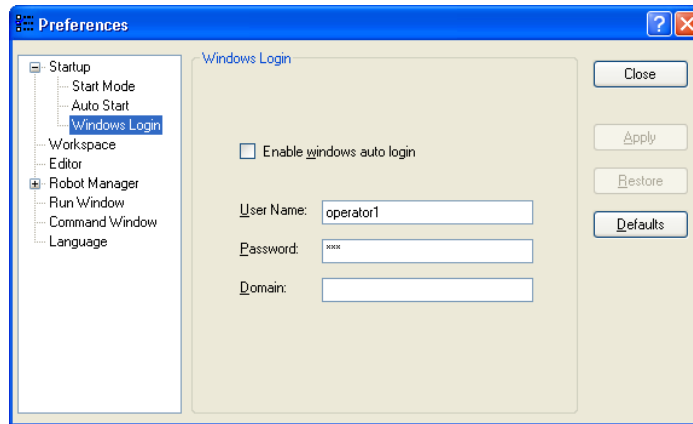
Item	Description
Enable windows auto login	Check this box if you want to automatically login to Windows when it starts. You must supply a valid user name, password, and domain.
User Name	Enter the name of a valid Windows user on the system.
Password	Enter the login password for the user.
Domain	Optional. If the PC is the member of a domain, enter the name here.
Apply	Save the current changes.
Restore	Revert back to previous settings.
Close	Close the Preferences dialog.

Setup: Preferences: Workspace Page

From this page, you can configure your workspace preferences.

Item	Description
Auto file save	Checking this box will cause EPSON RC+ 5.0 to automatically save any open files before executing a command that requires the file to be saved. For example, if a file needs to be saved before executing a project build, the file will automatically be saved before running the build. Default is On.
Stop compile after first error	Stops compile after first error occurs. This makes it easier to see the first error in the status pane and allows you to fix one error at a time. Default is On.
Do not open last project	If this radio button is selected, the last project will not be opened when EPSON RC+ 5.0 is started.
Open last project and program windows	If this radio button is selected, the last project will be opened and any program windows that were previously opened will be opened.
Open last project and all windows	If this radio button is selected, the last project will be opened and all windows will be restored to their previous locations. This is the default setting.
Show Tip of the Day	If this check box is on, the Tip of the Day dialog will be displayed when EPSON RC+ 5.0 is started.
Apply	Save the current changes.
Restore	Revert back to previous settings.
Close	Close the Preferences dialog.

Setup: Preferences: Editor Page
This page is used to configure your preferences for the program editor windows.
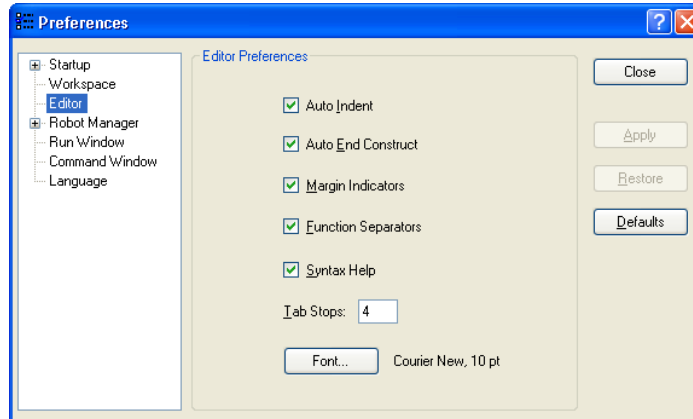
Item	Description
Auto Indent	Check this box if you want new lines to follow the indentation for the previous line. Also, lines will automatically be indented after Do, If, Else, For, Select, and Case statements. Default is on.
Auto End Construct	Check this box if you want EPSON RC+ 5.0 to add the end construct statement for a loop construct. For example, if you enter a For statement, then a Next statement will be added automatically. Default is on.
Margin Indicators	Check this box to display a margin on the left side. This margin is used to indicate lines with breakpoints, current step line, and current execution line. Default is on.
Function Separators	Check this box to display a line after each Fend statement. Default is on.
Syntax Help	Check this box to enable the Syntax Help window. The Syntax Help window displays syntax for a keyword after it has been typed. Default is on.
Tab Stops	Type in the number of columns to move for the TAB key. Default is 4.
Font	Click on the Font button to open the fonts dialog. Choose the font you desire for the editor. The monitor window also uses the editor font. The current font name and size is displayed next to the Font button.
Apply	Apply the current settings.
Restore	Revert back to the previous settings.
Defaults	Set default values.
Close	Close the Preferences dialog.

Setup: Preferences: Robot Manager: General Page

This page lets you configure your preferences for the Robot Manager.
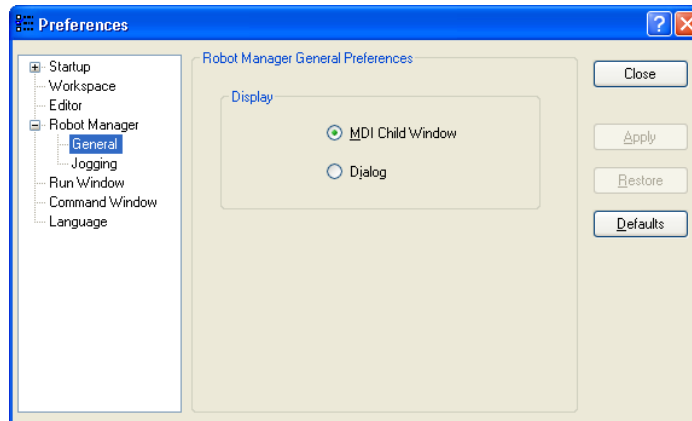
Item	Description
Display	Choose if you want the Robot Manager to be displayed as an MDI Window or as a Dialog.
Close	Close the Preferences dialog
Apply	Apply the current settings.
Restore	Revert back to the previous settings.
Defaults	Set default values.
Close	Close the Preferences dialog.

The Robot Manager can be displayed as an MDI child window (default) or as a dialog. When displayed as an MDI child, the Robot Manager is displayed in the MDI document area and can remain open while you work with other windows and dialogs. When displayed as a dialog, you can only work with the Robot Manager controls until you close the dialog. When using screen resolutions less than 1024 x 768, the Robot Manager MDI window will not fit in the MDI document area. In this case, only the Dialog mode is allowed.

Setup: Preferences: Robot Manager: Jogging Page
This page lets you configure the Robot Manager Jog and Teach page.
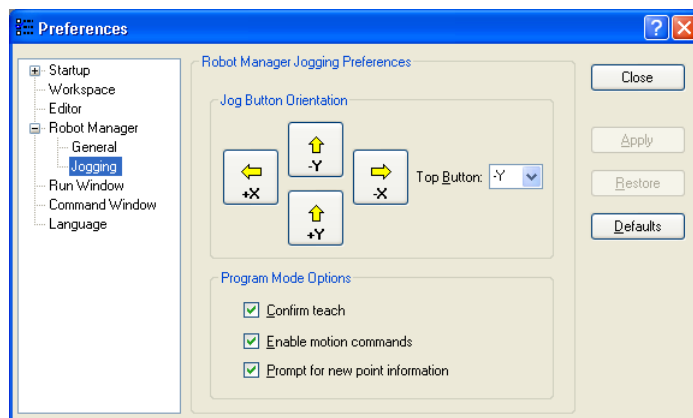
Setting Jog Button Orientation

The jog button orientations are useful for "aligning" your PC monitor with the robot's Cartesian coordinate system. Align the buttons so that the robot moves in the direction of the arrows.

You can change the orientation of the jogging buttons and arrow keys for the X and Y axes by selecting the desired top button from the **Top Button** dropdown list.

You can also click on one of the buttons to change it to the top button position.

Program Mode Options

NOTE
☞

These options affect the Robot Manager Jog & Teach page when used from program mode.

These settings do not affect the Robot Manager when used for operators in auto mode, such as for the Operator Window or from VB Guide. To configure the Robot Manager for operators, see *Project | Properties | Operator Settings | Robot Manager*.

Item	Description
Confirm teach	Check this box if you want a confirmation prompt each time you press the Teach button on the Robot Manager Jog & Teach page.
Enable motion commands	Check this box if you want to execute motion commands (Go, Jump, etc.) from the Robot Manager Jog & Teach page.
Prompt for new point information	Check this box if you want to be prompted for point label and description when a new point is taught using the Teach button.
Apply	Apply the current settings.
Restore	Revert back to the previous settings.
Defaults	Set default values.
Close	Close the Preferences dialog.

Setup: Preferences: Run Window Page
This page allows you to change preferences for the Run Window.
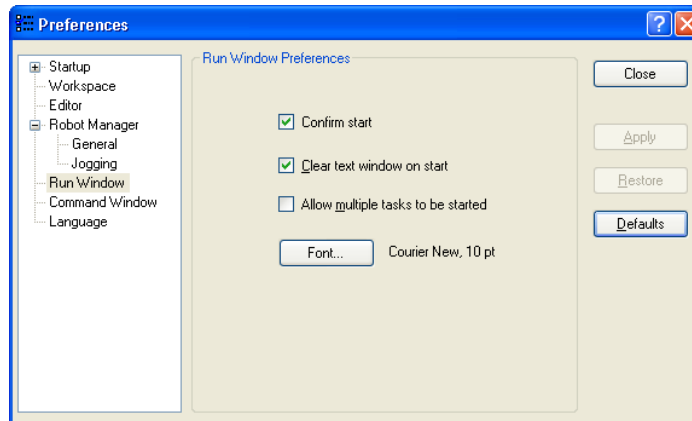
Item	Description
Confirm Start	This checkbox allows you to select if you want to see a confirmation message box before a program is started.
Clear text window on start	Checking this will cause the Run Window text pane to be cleared each time the **Start** button is clicked.
Allow multiple tasks to be started	Checking this allows you to start a task from the Run window while other tasks are running. The Start button will not be disabled after starting a task.
Font	Click on the Font button to open the fonts dialog. Choose the font you desire for the Run window. The current font name and size is displayed next to the Font button.
Apply	Apply the current settings.
Restore	Revert back to the previous settings.
Defaults	Set default values.
Close	Close the Preferences dialog.

Setup: Preferences: Command Window Page
This page allows you to change preferences for the Command Window.
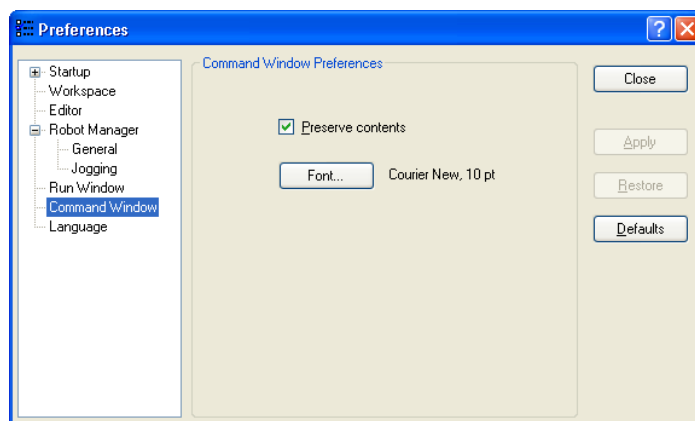
Item	Description
Preserve contents	Checking this option will cause the command window to preserve its contents between sessions.
Font	Click on the Font button to change the font for the Command window.
Apply	Save the current changes.
Restore	Revert back to the previous values.
Defaults	Set default values.
Close	Close the Preferences dialog.

Setup: Preferences: Language
This page allows you to change the EPSON RC+ 5.0 GUI language.
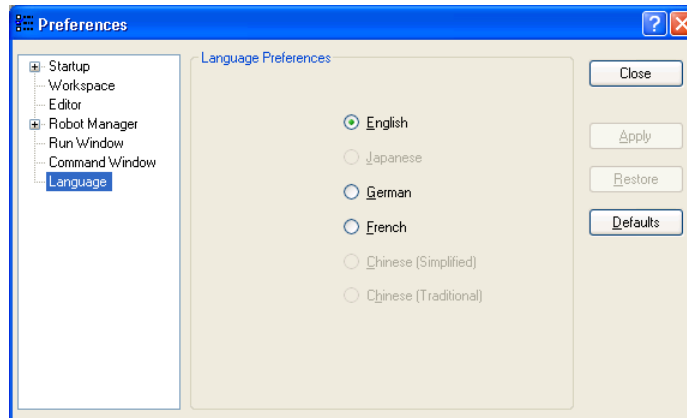
When EPSON RC+ 5.0 is installed on a Windows system using a Western language, then the English, German, and French selections are available.

When it is installed on a Windows system using Japanese, then Japanese and English selections are available.

When it is installed on a Windows system using Chinese, then English, Chinese (Simplified), and Chinese (Traditional) are available.

After selecting the desired language, you must re-start EPSON RC+ 5.0.

If you change the language to a language that does not match the Windows language, then some system dialogs will be in the Windows language. For example, if you select German for EPSON RC+ 5.0, but you are using English Windows, then messages boxes and some file navigation dialogs will be in English.

Item	Description
Language	This set of option buttons allows you to choose which language to use for the EPSON RC+ 5.0 GUI.
Apply	Save the current changes.
Restore	Revert back to the previous values.
Defaults	Set the default language.
Close	Close the Preferences dialog.

5.13 Window Menu

The Window Menu contains selections for managing the currently open EPSON RC+ 5.0 child windows.

5.13.1 Cascade Command (Window Menu)

Use Cascade to show all of the currently open files in windows of the same size, stacked one on top of another.
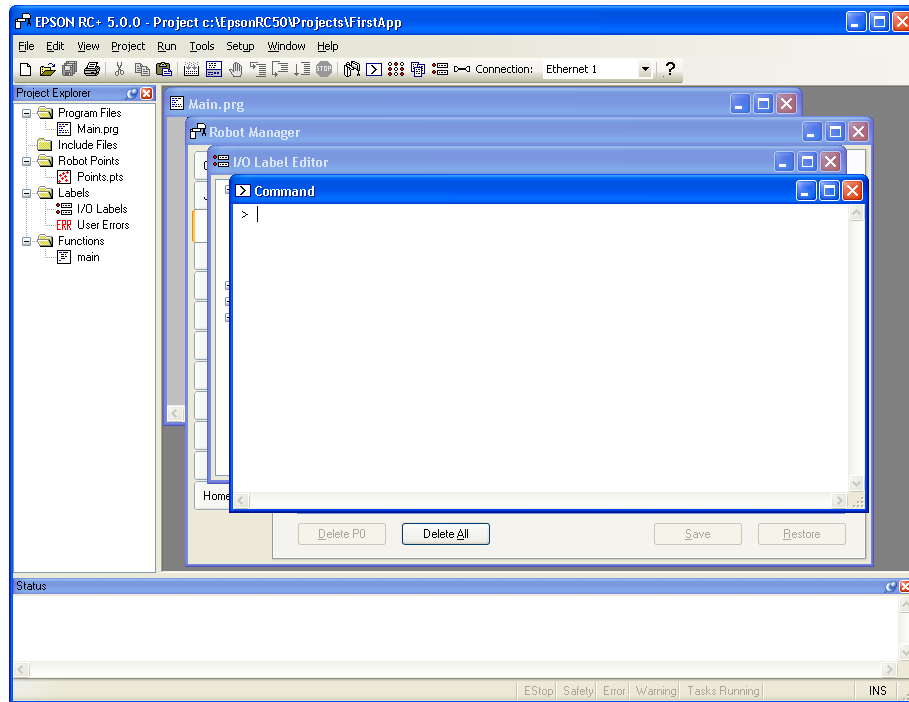
5.13.2 Tile Horizontal Command (Window Menu)

Use Tile Horizontal to evenly display all open windows horizontally.
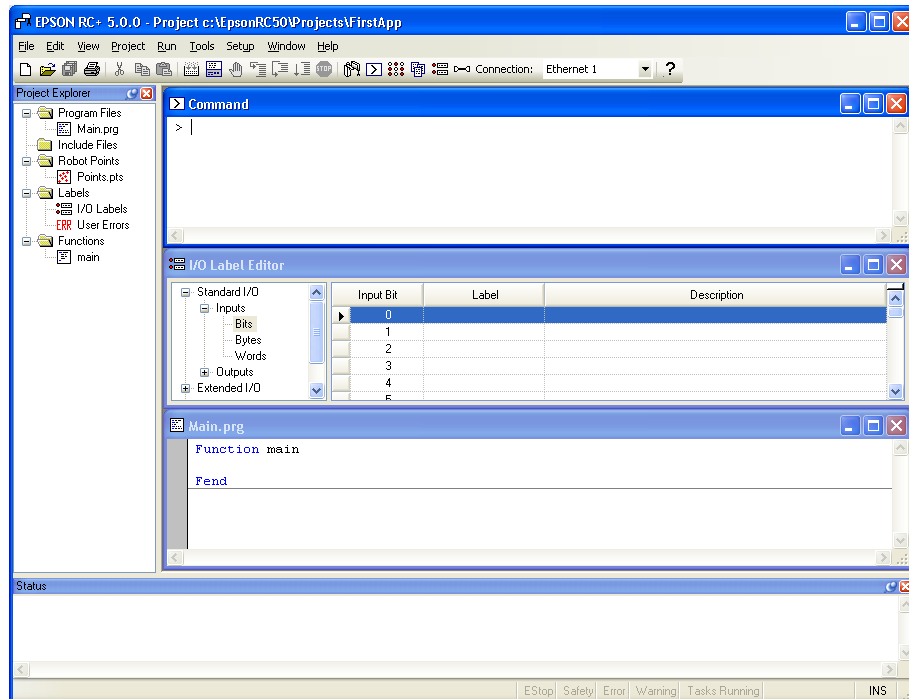
5.13.3 Tile Vertical Command (Window Menu)

Use Tile Vertical to evenly display all open windows vertically.
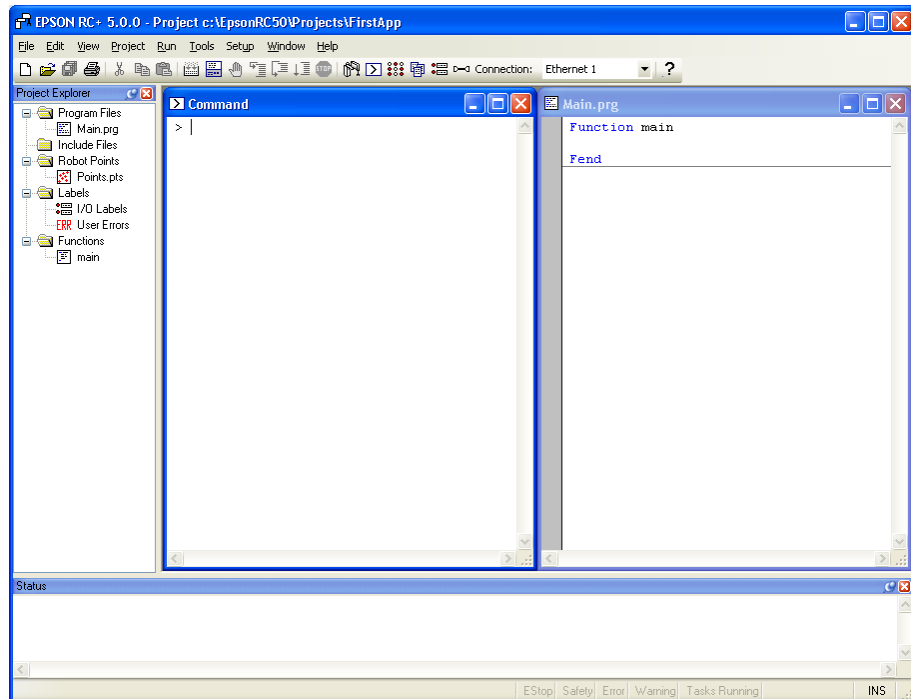
5.13.4 Arrange Icons Command (Window Menu)

Arrange the icons for all child windows that have been minimized.
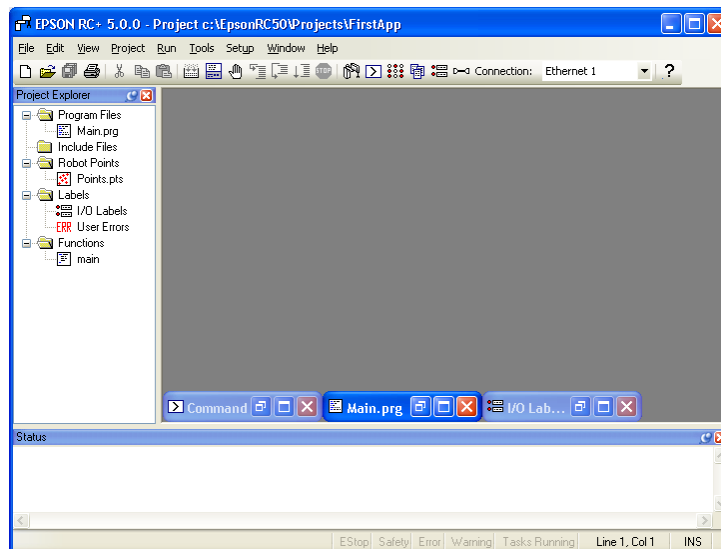
5.13.5 Close All Command (Window Menu)

This command closes all EPSON RC+ 5.0 child windows.

5.13.6 1, 2, 3 Command (Window Menu)

A list of currently open document windows is displayed at the bottom of the Window Menu.

When you choose an open window from the listing, you make that document active. A check mark appears in front of the document name of the currently active window.
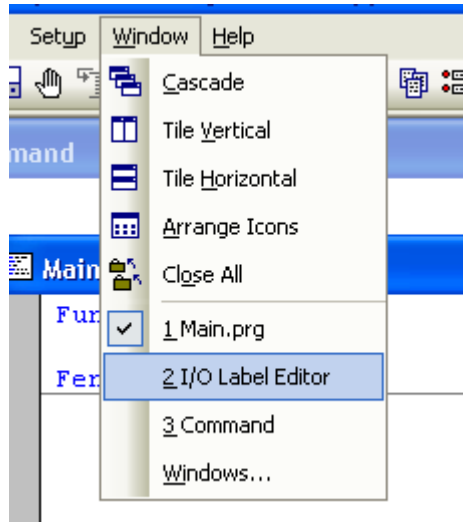
5.13.7 Windows Command (Window Menu)

This command displays a dialog that contains a list of all currently open EPSON RC+ 5.0 windows.
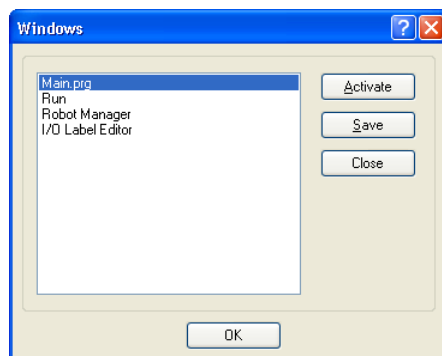
Item	Description
Activate	Brings the selected window into focus.
Save	Saves the contents of the selected windows.
Close	Closes the selected windows.
OK	Closes the dialog.

5.14 Help Menu

The Help Menu contains selections for accessing the help system and manuals along with version information.

5.14.1 How Do I Command (Help Menu)

Select How Do I to view topics that contain information for performing common tasks in EPSON RC+ 5.0.

Shortcuts

Keys: Ctrl +F1

5.14.2 Contents Command (Help Menu)

This command opens the Contents view for the EPSON RC+ 5.0 on-line help system.

From the Contents view, you can navigate through all of the topics in the help system. Double-click on a book icon to open or close the subtopic list contained within the book folder.

5.14.3 Index Command (Help Menu)

This command opens the Index view for the EPSON RC+ 5.0 on-line help system.

From the Index view, as you begin typing in a keyword, the alphabetical topic list will show the keywords starting with the letters you have typed.
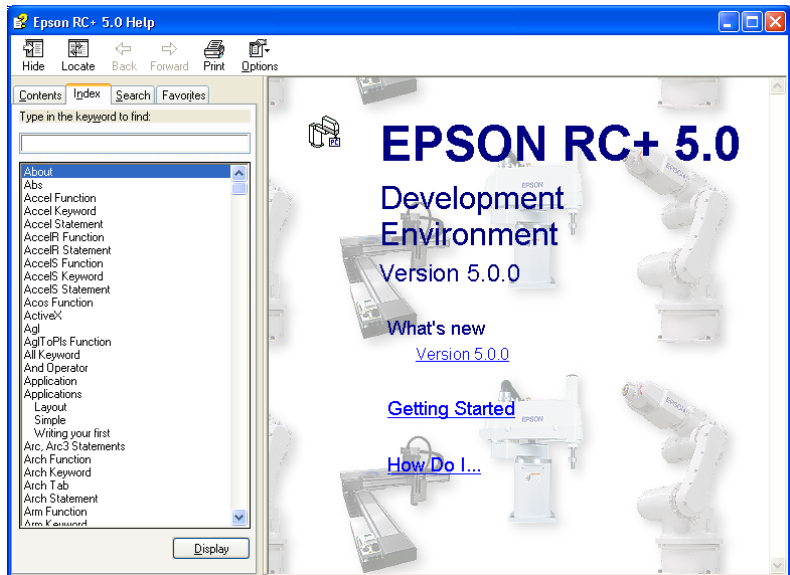
5.14.4 Search Command (Help Menu)

This command opens the Search view for the EPSON RC+ 5.0 on-line help system.

From the Search view, you can type in one or more keywords and click List Topics to show a list of all topics containing one or more of the keywords. The keywords are highlighted in the topics as shown below.
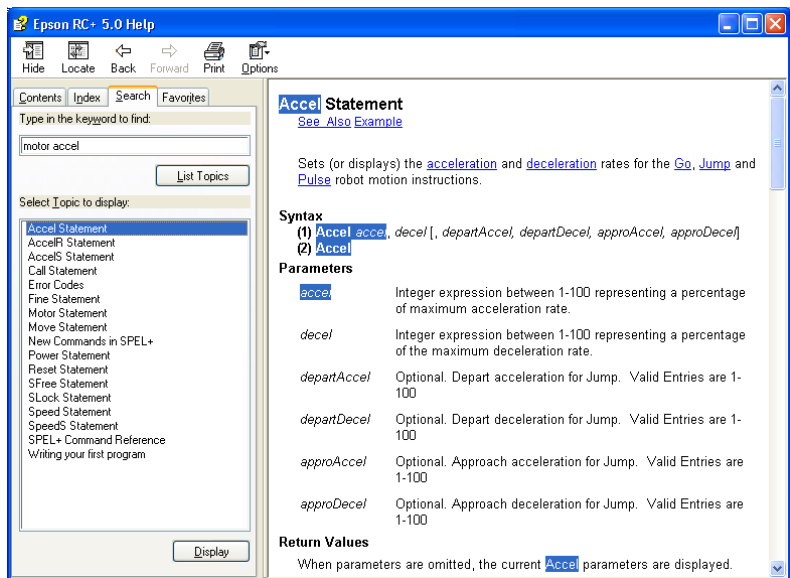
5.14.5 Manuals Submenu (Help Menu)

The Help Menu Manuals submenu contains selections for each of the manuals installed on the PC in Adobe PDF format.   These include manuals for EPSON RC+ 5.0, SPEL+ Language Reference, VB Guide 5.0, Controller, and various robot models.

5.14.6 About EPSON RC+ 5.0 Command (Help Menu)

The About command displays a dialog box showing the current version of the EPSON RC+ 5.0 software, along with copyright and license information. When calling technical support about EPSON RC+ 5.0, you should report the version you are using from this dialog.

6. The SPEL+ Language

This chapter contains information about the SPEL[+] Language.

Contents

- Overview

- Program structure

- Commands and statements

- Function and variable names

- Date types

- Operators

- Working with variables

- Working with strings

- Multi-statements

- Labels

- Comments

- Error handling

- Multi-tasking

- Robot coordinate systems

- Robot arm orientations

- Robot motion commands

- Working with robot points

- Input and output control

- Using Traps

6.1 Overview

SPEL$^+$ is a BASIC-like programming language that runs in the controller. It supports multitasking, motion control, I/O control.

Programs are written in ASCII text and then compiled into executable object files.
Several language instructions can also be executed in immediate mode from the Command window.

6.2 Program structure

6.2.1 What is a SPEL+ program?

A SPEL+ program is a collection of functions, variables, and macros. Each line in a program is a statement. The program statements are stored in one or more program text files. Every program file has a .PRG extension and resides in the project directory on the PC and optionally in the controller.

Each project must contain at least one program and there must be one function called "main". This is the default function. A compile error will occur if function main is not found.

In addition, seven more programs can be defined in the same project. Each program has its own start function: main1, main2…main7. Each of the eight programs can be started from the Operator window, from the remote console, or from VB Guide.

A function definition begins with the Function statement and ends with the Fend statement. All source statements to be executed by a function must be included within the body of the function.

The following program file contains two function definitions. Function Main calls function Func1.

```
MAIN.PRG
Function Main
   Call Func1
   ...
Fend
Function Func1
   Jump pickpnt
   ...
Fend
```

6.2.2 Calling functions

You can execute a user function by using the Call statement. The function can reside in any program file in the current project. To get a return value, you must use the function as an argument in a statement. You can also omit the Call statement if you don't need the return value. When Call is omitted, then parentheses for the arguments must not be supplied

Here are some examples:

```
Call MyFunc(1, 2)
MyFunc 1, 2
Print MyFunc(1, 2)
```

6.3 Commands and statements

Commands and statements consist of a SPEL$^+$ instruction followed by the parameters for that instruction.

A command is executed immediately. You can execute commands from the Command window or from the Macros dialog box.

Statements can be used only in programs.

Statements can include more than one SPEL$^+$ instruction by using a semi-colon (;) to separate instructions.

The maximum length for a statement is 200 characters. This includes the line number.

6.4 Function and variable names (Naming restriction)

Function and variable names can include up to 32 alphanumeric characters and the underscore character. Characters can be upper case or lower case.

The following names are valid:

```
Function main
Real real_var
Integer IntVar
```

Function and variable names **cannot** begin with a numeric digit or underscore.

The strings that are already used as keywords cannot be use. (Example: Go / On)

String variables must have an additional dollar sign ('$') suffix, as shown in the example below:

```
Function Test
   String modname$
   Print "Enter model name:"
   Line Input modname$
   Print "model is ", modname$
Fend
```

Restriction for naming in SPEL+ language

- Characters can be alphanumeric characters or underscore character.

- Use alphabets for the first letter.

- Characters can be upper case or lower case.

- No keywords can be used.

- Maximum limits of names are as follows.

Name	Max. limit
Point label	16
I/O label	16
User error label	16
Function name	32
Variable name	32
Line label	32

6.5 Data types

You can declare different types of data in your program. All variables must be declared.

The following table shows the different data types for the SPEL$^+$ language.

Data Type	Size	Range
Boolean	2 byte	True or False
Byte	2 byte	−128 to +127
Double	8 bytes	−1.79E+308 to 1.79E+308 Number of significant figure is 14
Integer	2 bytes	−32768 to +32767
Long	4 bytes	−2147483648 to +2147483647
Real	4 bytes	−3.40E+38 to 3.40E+38 Number of significant figure is 6
String	256 bytes	All ASCII characters Up to 255 characters

6.6 Operators

The following table shows the different operators for the SPEL$^+$ language.

Keyword or Symbol	Description
+	Addition
−	Subtraction
*	Multiplication
/	Division
**	Exponentiation
=	Equal
>	Greater than
<	Less than
>=	Greater or equal
<=	Less or equal
<>	Not equal
And	Performs logical and bitwise AND operation.
Mod	Returns the remainder obtained by dividing a numeric expression by another numeric expression.
Not	Performs logical or bitwise negation of the operand.
Or	Performs the bitwise Or operation on the values of the operands.
Xor	Performs the bitwise Xor operation on the values of the operands.

6.7 Working with variables

6.7.1 Variable scopes

There are three different scopes for variables in SPEL$^+$:

- Local

- Module

- Global

6.7.2 Local variables

Local variables are available to all statements in the same function. Functions using local variable names can not refer to the same local variables in other functions. This is why they are called locals, because they are local to the function they are being used in.

To declare local variables in a function, use one of the variable declaration instructions at the beginning of the function after the Function statement:

Boolean, Byte, Integer, Long, Real, Double, String

For example, the following function declares several local variables:

```
Function test
   Integer intVar1, intVar2
   Real realVar
   String dataStr$
   Integer array(10)
   .....
Fend
```

6.7.3 Module variables

Module variables are available to all functions in the same program file.

To declare module variables in a program, use one of the variable declaration instructions at the beginning of the program before any Function statements:

Boolean, Byte, Integer, Long, Real, Double, String

One way to indicate that a variable is module level is to precede the name with "m_", as shown in the example below.

For example, the following function declares several module level variables:

```
' Module level vars, used by all functions in this file
Integer m_IntVar1, m_IntVar2
Real m_RealVar
String m_DataStr$
Integer m_Array(10)

Function main
  m_IntVar1 = 25
  Call test
Fend


Function test
   Print m_IntVar1
Fend
```

6.7.4 Global variables

Global variables can be shared between all functions in a project. The Global instruction is used to declare a global variable.

To declare global variables in a program, use the Global instruction with the desired variable type (Boolean, Byte, Integer, Long, Real, Double, String) at the beginning of the program before any Function statements:

One way to indicate that variables are global is to precede the name with "g_", as shown in the example below.

Program: MAIN.PRG

```
Global Integer g_TotalCycles

Function main
  Call LoadPart
  :::
Fend
```

Program: LOADPART.PRG

```
Function LoadPart
  Jump pick
  On gripper
  Wait .1
  Jump place
  Off gripper
  Wait .1
  g_TotalCycles = g_TotalCycles + 1
Fend
```

For more information, see Data Types.

6.7.5 Global Preserve variables

You can preserve global variable values by using the optional Preserve parameter when you declare global variables.

Preserved variables are stored in the controller's SRAM.

If the data type of a preserved variable is changed, or the number of dimensions is changed, then the variable values will be cleared.

6.7.6 Arrays

You can declare local, module, and global arrays with up to three dimensions for all data types.

To declare an array, use this syntax:

dataType *name* **(** *ubound1* [, *ubound2* [, *ubound3*]] **)**

SPEL$^+$ arrays are zero based. The first element is referenced with a value of zero.

The total available number of array elements for local and global preserve variables is 100 for strings and 1000 for all other types.

The total available number of array elements for global and module variables is 1000 for strings and 10000 for all other types.

To calculate the total elements used in an array, use the following formula. (If a dimension is not used, substitute 0 for the ubound value.)
total elements = (ubound1 + 1) * (ubound2 + 1) * (ubound3 + 1)

Array declaration examples:

```
' Global string array
Global String gData$(10)
Function main
  ' Arrays local to this function
  Integer intArray(10)
  Real coords(20, 10)
```

Use Redim to change the bounds of an array at run time.

```
Integer a(10)
Redim a(20)
```

To preserve values when using Redim, add the Preserve optional argument.

```
Integer a(10)
Redim Preserve a(20)
```

Use UBound to get the upper bound of an array dimension.

```
Integer i, a(10)
For i = 1 to UBound(a)
  a(i) = i
Next i
```

6.7.7 Initial values

All variables are initialized when first used except for Global Preserve variables. Strings are set to empty, and all other variables are set to zero.

6.7.8 Clearing arrays

Use Redim (without Preserve) to clear all of the elements of an array at run time.

6.8 Working with strings

A string in SPEL$^+$ is a set of ASCII characters with a maximum length of 255.

You must declare strings in your programs with the String instruction.

All string variable names must end with a dollar sign ($) suffix.

You can declare string arrays with up to three dimensions.

The following table shows the string commands available in SPEL$^+$.

Keyword	Description
Asc	Returns the decimal ASCII value of the first character in a string.
Chr$	Converts an ASCII value into a one character string.
FmtStr	Formats a numerical or date/time expression.
Hex$	Returns a string containing the hexadecimal value of a number.
InStr	Returns the position of a substring within a string.
LCase$	Returns the specified string in lower case characters.
Left$	Returns a substring beginning with the first character of a string.
Len	Returns the length (number of characters) of a string.
LTrim$	Returns the specified string with left spaces removed.
Mid$	Returns a substring of a string.
ParseStr	Parses a string into an array of tokens.
Right$	Returns a substring from the end of a string.
RTrim$	Returns the specified string with right spaces removed.
Space$	Returns a string containing a specified number of space (ASCII 32) characters.
Str$	Converts a number to a string.
String	Declare a string variable in a program.
Trim$	Returns the specified string with left and right spaces removed.
UCase$	Returns the specified string in upper case characters.
Val	Converts a string to a number.

6.9 Multi-statements

A program statement can contain several statements separated by semi-colons. The total length of a multi-statement program line cannot exceed 200 characters.

For example:

```
Function Test
    Pass P1; Pass P2; Go P3    ' Multi-statement
Fend
```

Even though multi-statements are supported (mainly for compatibility), it is strongly recommended that they not be used. Multi-statements can make your code more difficult to read and debug.

6.10 Labels

A program label is an alphanumeric name followed by a colon (":") that marks a location in a program for a GoTo or GoSub statement. The name may be up to 32 characters long and can include alphanumeric characters and the underscore ("_") character if it is not the first character. You cannot use any SPEL[+] keywords as labels.

For example:

```
Function Main
  Do
    Jump P1
    Jump P2
    If Sw(1) Then GoTo MainAbort
  Loop
MainAbort:   ' Program label
  Print "Program aborted"
Fend
```

6.11 Comments

Use comments to add notes to your programs. An apostrophe character (') starts a comment.

Examples of comments

```
Function Main
  ' ****** Main Demo Program ******
  Xqt conveyor     ' Start up the task for conveyor
  Do
     Print "Press ENTER to run demo cycle"
     Print "Press CTRL+C to quit"
     Input dumy
     Call demo    ' Execute the demo function
  Loop             ' Return to start of main loop
```

6.12 Error handling

When an error occurs in a SPEL+ function, you can cause execution to be transferred to an error handling routine for processing the error. The routine must be inside a function definition.

The table on the next page shows the program instructions that are used for error handling.

Item	Purpose
OnErr	Use the OnErr statement to define the location of the error handling routine.
Err	Use Err to retrieve the number for the current error status. Use this in the error handling routine to determine which error has occurred.
Error	Generate a user defined error which can be caught by an error handler.
Era	Use Era to retrieve the axis number for which the error occurred. This is normally used in the error handling routine.
Erl	Use Erl to retrieve the line number in which the error occurred. This is normally used in the error handling routine.
Ert	Use Ert to retrieve the task number in which the error occurred. This is normally used in the error handling routine.
ErrMsg$	Use ErrMsg$ to retrieve the error message associated with a specified error number.

User Errors

You can define your own error messages by using the User Error Editor which is available from the Tools Menu. See the section on User Error Editor in the chapter *EPSON RC+ 5.0 GUI* for details.

Example

The following example shows a simple error handling routine. When an error occurs, program execution goes to the ErrHandler label, where the error handler starts. The error number is displayed and the operator is asked to continue or not. If the operator enters "N" then the program executes the Quit All statement to end the program.

```
Function Main
    String cont$
    Integer i
    OnErr Goto Errhandler
    For i = 1 To 10
       Jump P(i)
    Next i
    Exit Function
' *** Error handler ***
Errhandler:
    enum = Err
    Print "Error #", enum, " occurred"
    Print "Continue (Y or N)?"
    Line Input cont$
    Select cont$
       Case "y", "Y"
          EResume Next
       Default
          Quit All
       Send
    Fend
```

6.13 Multi-tasking

For some applications, you may want to control other equipment besides the robot, such as conveyors, pick and place units, etc. By using multi-tasking, you can control this other equipment with their own tasks.

SPEL$^+$ supports up to 16 tasks running simultaneously. A task is a function that has been started by the system or by the Xqt statement.

Use the Xqt statement to start another task from within a function. You can optionally specify a task number from 1 to 16 in the **Xqt** statement.

The table below shows the program instructions that are used for multitasking.

Statement	Purpose
Xqt	Starts a function as a task.
Halt	Temporarily suspends execution of a task.
Resume	Resumes a task that has been halted.
Quit	Stops a task.
Signal	Sends a signal to one or more tasks that are waiting for the signal using WaitSig.
SyncLock	Locks a resource for use by the current task and blocks other tasks from using the resource until **SyncUnlock** is executed.
WaitSig	Waits for a signal from another task.
Pause	Pause all tasks.

One example for starting another task is to run a conveyor system for the robot work cell.

Program: MAINTASK.PRG

```
Function Main
  Xqt Conveyor              ' Start the conveyor task
  Do
    ...
    ...
  Loop
Fend
```

Program: CONVTASK.PRG

```
Function Conveyor
  Do
    Select True
      Case Sw(10) = On
        Off convCtrl
      Case Sw(11) = On
        On convCtrl
    Send
  Loop
Fend
```

6.14 Robot coordinate systems

6.14.1 Overview

In this section we discuss the coordinate systems for different types of robots supported in SPEL$^+$.

The following coordinate systems are used in SPEL$^+$:

Robot Coordinate System   This is the native coordinate system of the robot. This is also known as the default base coordinate system.

Local Coordinate System   This is a user defined coordinate system located somewhere within the working envelop.

Tool Coordinate System   This is the coordinate system of the tool mounted on the robot end-effector.

6.14.2 Robot Coordinate System

Robot Coordinate System of SCARA Robot
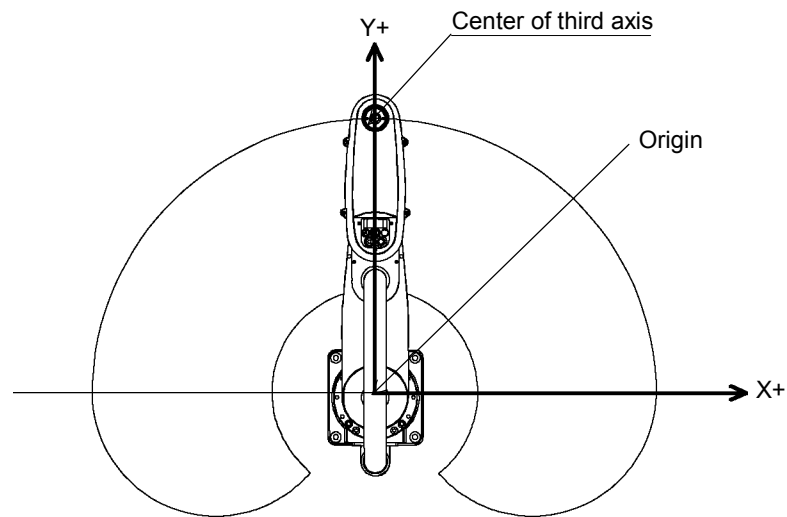
Z axis in Robot Coordinate System
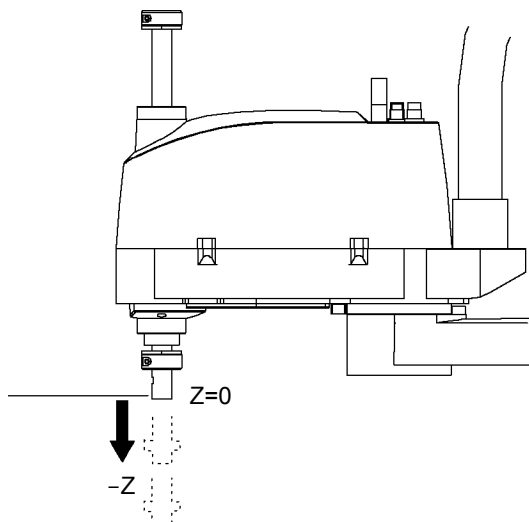
U axis in Robot Coordinate System
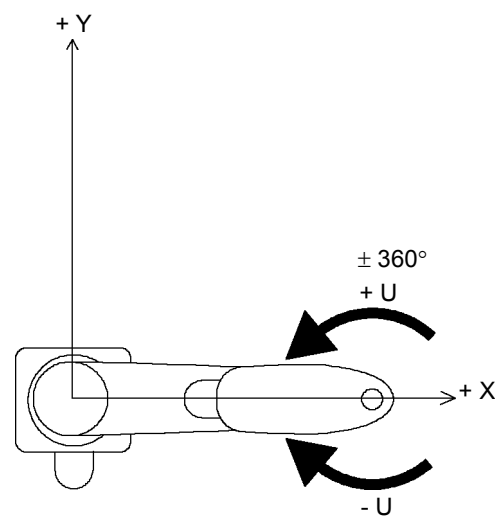
Robot Coordinate Systems for 6-Axis Robot
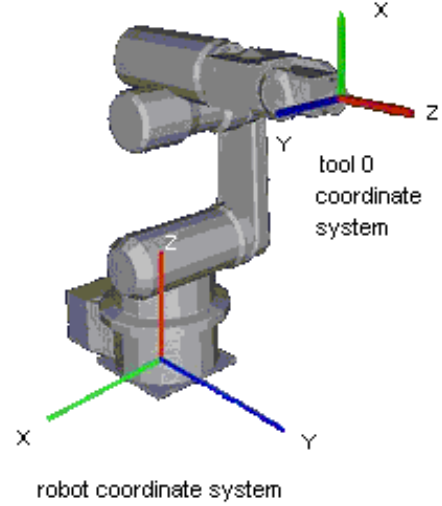
Floor Mounting   Ceiling Mounting
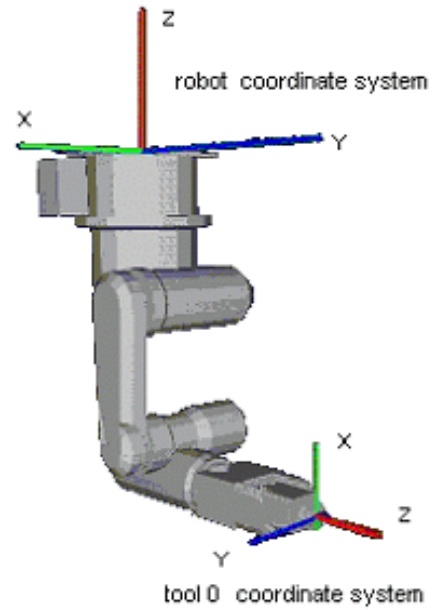
Side (Wall) Mounting
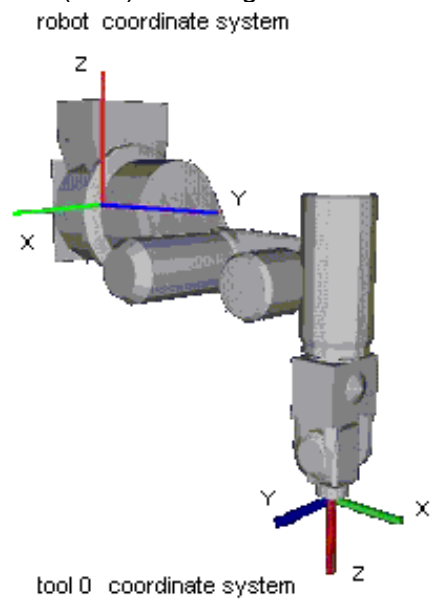
Robot Coordinate Systems for RS series
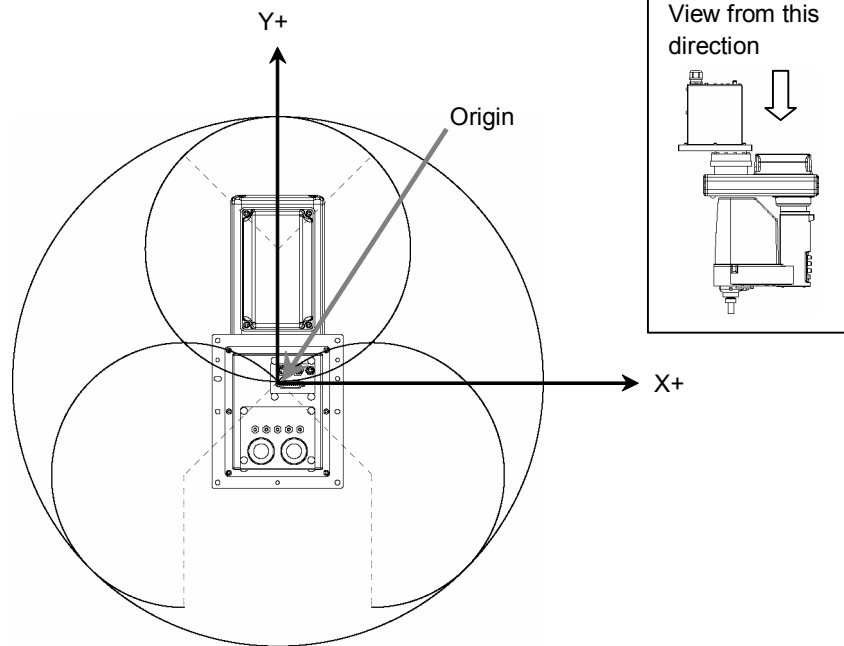
Robot coordinate system Z axis   Robot coordinate system U axis

Z=0

−Z

Joint #3 in the center

+ Y

± 360°

- X

− U   + U

View from this direction
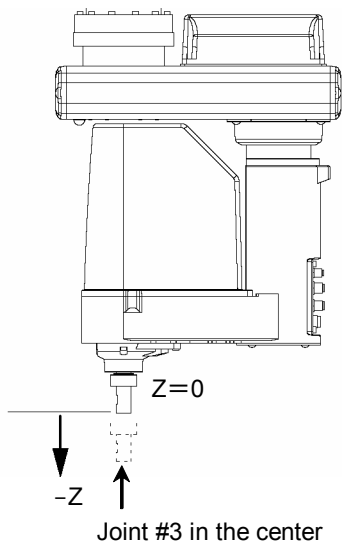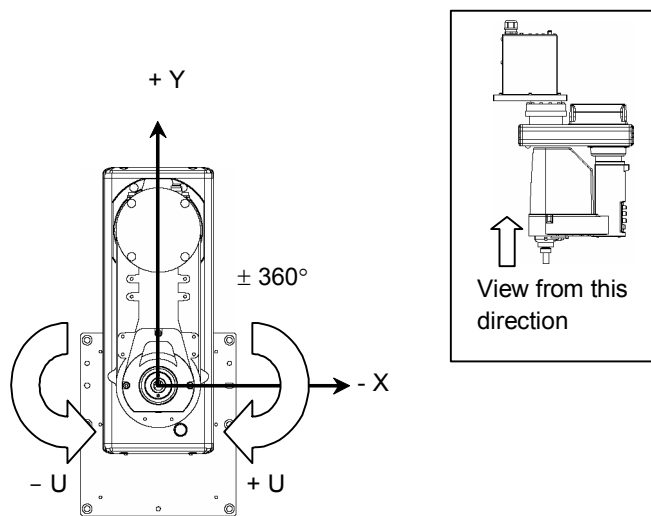
6.14.3 Local Coordinate Systems

With SPEL+, a maximum of 15 local coordinate systems can be defined. SPEL+ correlates robot coordinate systems and local coordinate systems by defining in advance the relative positional relationship of the local coordinate system from the robot coordinate system, assigning local numbers (1 to 15), and then assigning the local numbers to coordinate system attributes (local).

To define a local coordinate system, use the Local statement.

6.14.4 Tool Coordinate Systems

Point data is defined by the position and orientation of the tool coordinate system with respect to some reference rectangular coordinate system. The position is specified by the position data (X, Y, Z) and the orientation is specified by the orientation data (U, V, W) that correspond with roll, pitch, and yaw.

You can also define and use your own tool coordinate systems. To define the tool coordinate systems, use Tlset.

The default TOOL 0 coordinate systems are defined as follows according to the robot type.

SCARA Tool 0 coordinate system

The origin of tool 0 for SCARA robots is the center of the forth joint (rotation joint). When the fourth joint is adjusted to the position of 0 degrees, the tool 0 coordinate system axes are parallel to the robot coordinate system axes (see the figure below.)

The tool 0 coordinate system rotates as the fourth joint rotates.
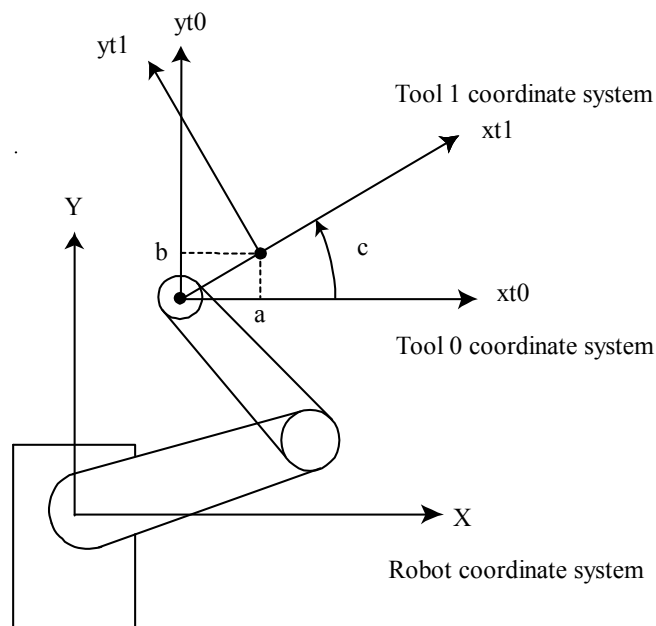
6-Axis Tool 0 coordinate system

For table mounting, the origin of TOOL 0 is the flange side center of the sixth joint. In TOOL 0, the tool Z axis is perpendicular to the sixth joint flange. (See the figure in the previous section *Robot Coordinate Systems*). The TOOL 0 coordinate system moves as the 6-Axis robot changes its orientation.

For ceiling mounting and wall mounting robots, the TOOL 0 coordinate systems are defined as shown in the figures in the section *Robot Coordinate Systems*.

6.14.5 ECP Coordinate Systems (Option)

Specify a coordinate system whose origin point is on the tip of the outside fixed tool (hereafter referred to as the external control point or ECP) to move the robot arm holding a part in the trajectory made on the external control point along with the part's edges.
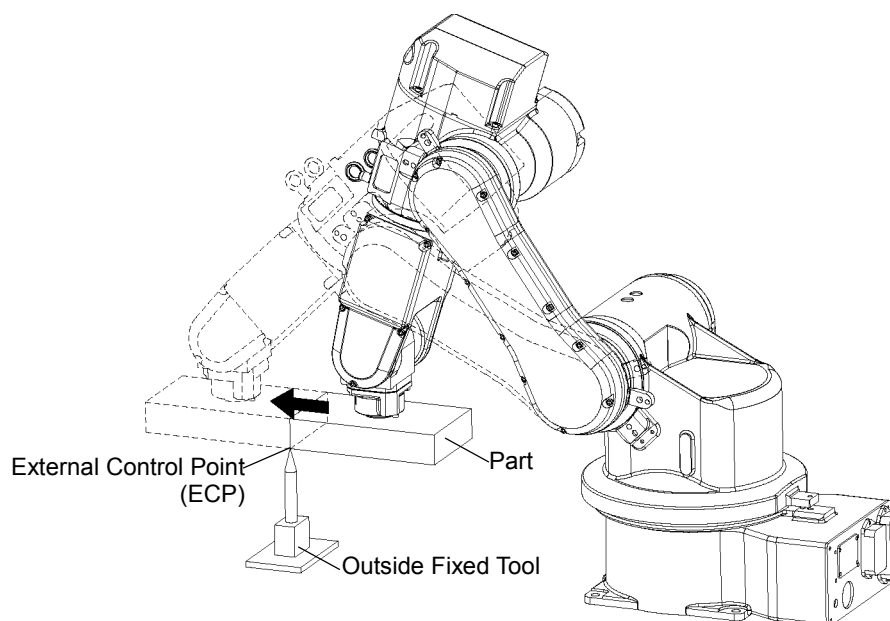
Use the ECPSet statement for defining an ECP coordinate system. A maximum of 15 ECP coordinate systems can be defined.

The following commands are available for optional ECP:

- Move command

- Arc3 command

- Curve and CVMove commands

- ECP jog motion

For details, refer to the chapter ECP Motion.

6.15 Robot arm orientations

When developing a robot a program, it is necessary to specify the point data taught for a particular arm orientation. If you fail to do so, the position can deviate slightly depending on the arm orientation, which in turn can cause the arm to follow an unexpected path, resulting in interference with peripheral equipment. This can be dangerous! To prevent this from happening, the orientation that the arm will be in when moved to the given point should be specified ahead of time in the point data. Such information can also be changed from the program.

6.15.1 SCARA robot arm orientations

With two types of arm orientation, a SCARA robot can move to nearly any position and orientation within a given work envelope. Examples are shown in the figures on the next page.
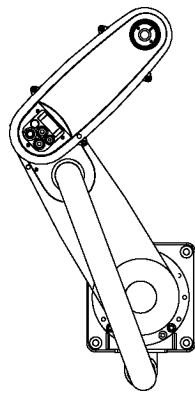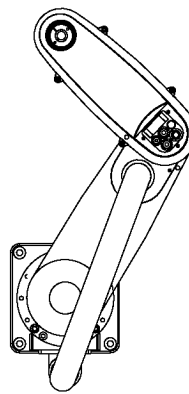
Lefty arm orientation   Righty arm orientation

Examples of moving to the same point using Lefty and Righty arm orientations

6.15.2 6-Axis robot arm orientations

The 6-Axis robot can be operated in various arm orientations within a given work envelope as shown below:
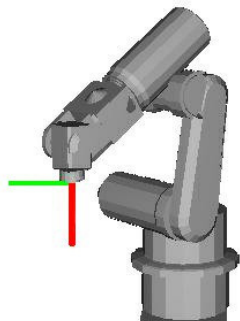
Righty hand orientation   Lefty hand orientation
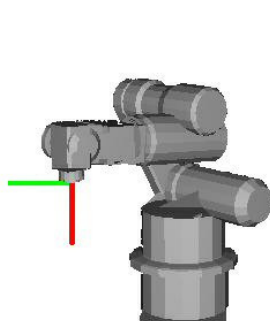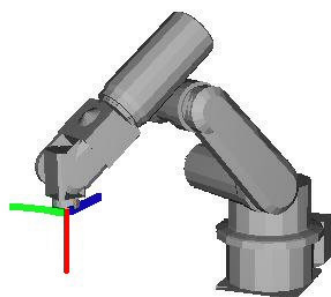
Above elbow orientation   Below elbow orientation
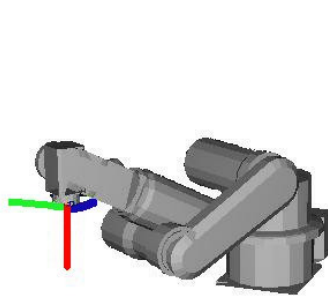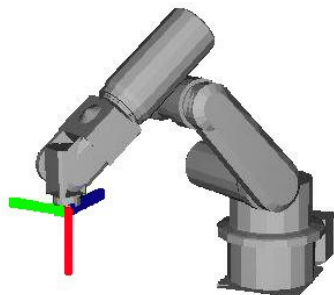
NoFlip wrist orientation   Flip wrist orientation
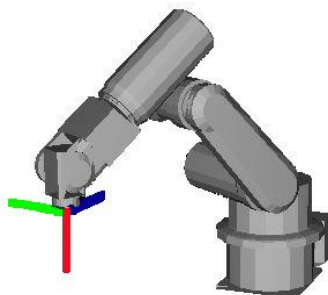
To specify orientation for the 6-Axis robot, add a forward slash (/) followed by L (for Lefty hand orientation) or R (Righty hand orientation), A (Above elbow orientation) or B (Below elbow orientation), and NF (NoFlip wrist orientation) or F (Flip wrist orientation).

There are eight available orientations as shown below, however, the 6-Axis robot cannot be operated in all of the orientations depending on point.

Available Orientation

1	/R /A /NF	5	/R /A /F
2	/L /A /NF	6	/L /A /F
3	/R /B /NF	7	/R /B /F
4	/L /B /NF	8	/L /B /F

At some points in the work envelope, the 6-Axis robot can have the same position and orientation even if the fourth joint or the sixth joint are rotated 360 degrees. To distinguish these points, the J4Flag and J6Flag point attributes are provided.

To specify the J4Flag, add a forward slash (/) followed by J4F0 (-180 < the forth joint angle <= 180) or J4F1 (the forth joint angle <= -180 or 180 < the forth joint angle).

To specify the J6Flag, add a forward slash (/) followed by J6F0 (-180 < the sixth joint angle <= 180), J6F1 (-360 < the sixth joint angle <= -180 or 180 <the sixth joint angle <= 360), or J6Fn (-180*(n+1) < the sixth joint angle <= 180*n or 180*n <the sixth joint angle <= 180*(n+1)).

Singularity

The orientation in the boundary where the arm orientation switches to the other

Hand singularity : The boundary where Righty hand orientation and Lefty hand orientation switch

Elbow singularity : The boundary where Above elbow orientation and Below elbow orientation switch

Wrist singularity : The boundary where NoFlip wrist orientation and Flip wrist orientation switch
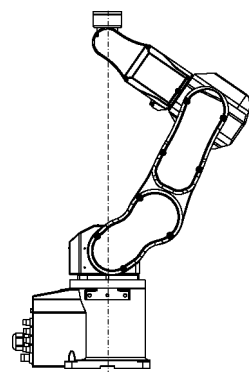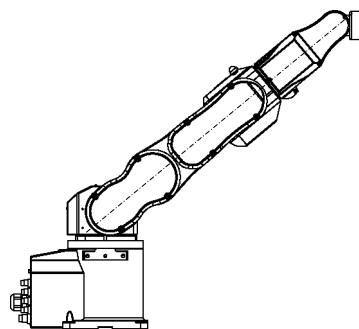
Hand singularity   Elbow singularity   Wrist singularity
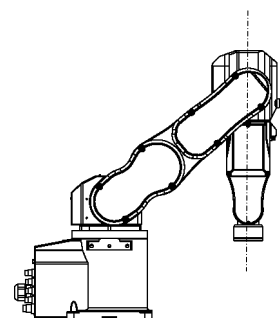
For the 6-axis robot, Hand / Wrist singularities exist also inside the motion range. When jogging near the singularity, follow the directions below.

PTP motion near the singularity

When jogging a robot from a point near the singularity to a point calculated by point operations such as P1+X(10), the robot may move to unintended direction because the arm orientation is not properly specified.

For example, when jogging from a point where the wrist is NoFlip to another point calculated by point operations, if the wrist keeps the NoFlip orientation while jogging, Joints #4 and #6 may rotate widely (by approx. 180 degrees).

In this case, switch to the Flip wrist orientation to jog smoothly through the wrist singularity. This phenomenon occurs not only with the point operations but also when creating points automatically with Pallet command or the result values that run from vision sequence.

Proper motion

Unintended motion (Joints #4, #6 rotate 180 degrees)
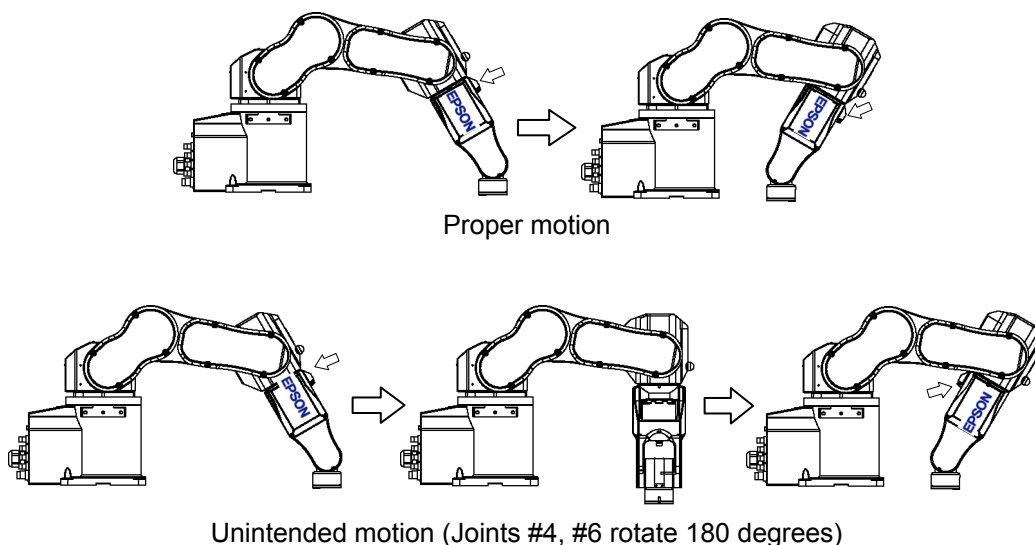
However in the cases like this, it is difficult for users to specify the proper arm orientations by a program. For this LJM function is a useful command. LJM function switches the arm orientations to enable the least motion of the joints. For the details of LJM function, refer to *SPEL+ Language Reference manual*.

Also, AutoLJM command can automatically apply LJM function to the motion commands which are included in a particular section of the program without using LJM function. For details of AutoLJM command, refer to the *SPEL+ Language Reference*.

In addition, you can set AutoLJM function to be enabled at the controller start up by setting preferences of the controller. However, if Auto LJM is enabled in preferences, this function automatically adjusts the posture of the manipulator to reduce the motion distance, even when you intend to move the joint widely. Therefore, it is recommended to build a program using AutoLJM command or LJM function to operate the manipulator as you desired.

If you specify all points by teaching, the arm orientations are also recorded. Therefore, the manipulator moves to the taught position without using LJM function or AutoLJM. Instead, the manipulator may move differently from the taught position by the use of LJM and AutoLJM.

LJM function for CP motion command

LJM function and AutoLJM command described above are also available for CP motion commands. However, since CP motion commands give priority to operate based on specified trajectories, the manipulator sometimes reach to the point with a different posture from the specified one. At this time, if CP motion command is used with CP On, an error from 4274 to 4278 will occur according to the mismatched point flag. To avoid the error, operate the manipulator with CP Off, or match the point flag of a target point and the one after motion completion. If operated with CP Off, the error does not occur and the manipulator can continue operation from the point where the mismatch happened.

Also, you can set the controller's preference so that the mismatches of flags are not considered as an error at the controller startup. However, path motions which use CP On will be disabled.

CP motion near the singularity (singularity avoiding function in CP motion)

When executing Move or CP motion near the singularity, joint speed may increase rapidly. The over speed error will occur and the joints will move widely and interfere with peripherals. Particularly the position of Joint #1 changes greatly near the hand singularity and the position of Joints #2 - #6 near the wrist singularity.

RC+5.0 has a singularity avoiding function to prevent acceleration errors during the execution of CP motion commands that pass the wrist singularity described above. With this function, the manipulator takes a detour to avoid an acceleration error by passing a different trajectory and returns to the original trajectory after passing the singularity. For details of the singularity avoiding function, refer to *AvoidSingularity* in the *SPEL+ Language Reference*.

Singularity avoiding function is enabled as default. If you want to avoid the error by reducing the motion speed in order to maintain the trajectory accuracy, you can disable the function temporarily by setting "0" to AvoidSingularity.

If you cannot avoid errors even if you use the singularity avoiding function, use PTP motion to enable the least motion of the joints or arrange the manipulator installation position and hand offset volume to prevent the CP motion near the singularity.

6.15.3 RS series arm orientations

The RS series can be operated in various arm orientations within a given work envelope as shown below:
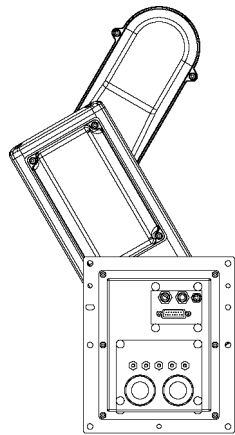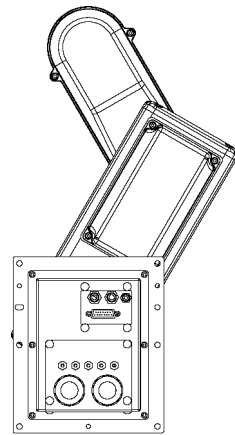
Lefty arm orientation   Righty arm orientation
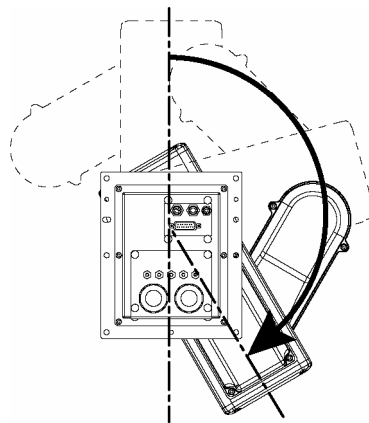
J1 F0 arm orientation   J1 F1 arm orientation
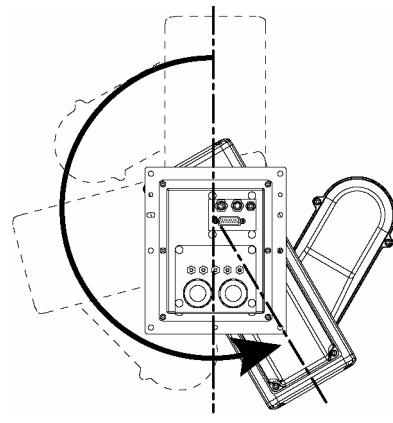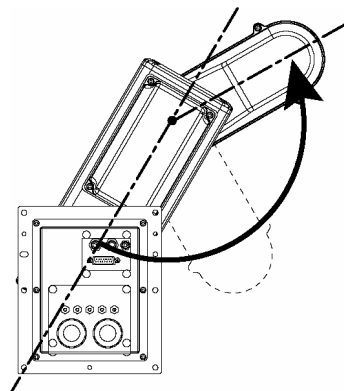
J2 F0 arm orientation   J2 F1 arm orientation
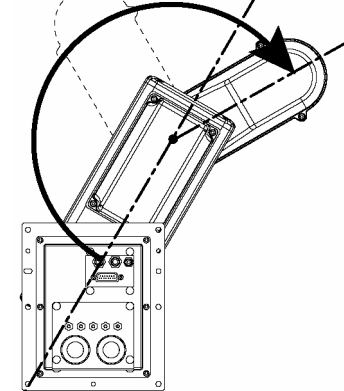
To specify orientation for the RS series, add a forward slash (/) followed by L (for Lefty hand orientation) or R (Righty hand orientation), J1F0 or J1F1, J2F0 or J2F1.

The RS series can have the same position and orientation even if the first joint or second joint are rotated 360 degrees. To distinguish these points, the J1Flag and J2Flag point attributes are provided.

To specify the J1Flag, add a forward slash (/) followed by J1F0 (-90 < the first joint angle <= 270) or J1F1 (-270 < the first joint angle <= -90 or 270 < the first joint angle <= 450).

To specify the J2Flag, add a forward slash (/) followed by J2F0 (-180 < the second joint angle <= 180) or J2F1 (-360 < the second joint angle <= -180 or 180 < the second joint angle <= 360).

There are eight available orientations as shown below. However, the RS series cannot be operated in all of the orientations depending on point.

Available Orientation

1	/R /J1F0 /J2F0	5	/R /J1F0 / J2F1
2	/L /J1F0 /J2F0	6	/L /J1F0 / J2F1
3	/R /J1F1 /J2F0	7	/R /J1F1 / J2F1
4	/L /J1F1 /J2F0	8	/L /J1F1 / J2F1

6.16 Robot motion commands

SPEL$^+$ includes several commands for controlling the robot from your programs.

6.16.1 Homing the robot

The Home command moves the robot to a user defined "park" or "idle" position. This command works for all robots. It is mainly used for absolute encoder robots that normally do not need to be mechanically homed. Use the HomeSet command to set the home position and the Hordr command to set the home order.

6.16.2 Point to point motion

Point to point (PTP) commands move the robot from its current position to a specified point. Motion may not be in a straight line.

To set the speed for point to point commands, use the Speed command. To set acceleration and deceleration, use the Accel command.

Command	Description
Go	Move directly to a point using point to point motion.
Jump	Jump to a point. First move up to the current LimZ setting, the move over the destination point, then move to the point. The Arch table settings determine the Jump profile.
Jump3	Jump to a point in 3 dimensions.
Pass	Move near one or more points.
TGo	Move directly to a point in a tool coordinate system.

6.16.3 Linear motion

Linear motion commands move the robot from its current position to a specified point in a *straight line*. Liner motion is a CP (Continuous Path) motion.

To set velocity (speed) for straight motion, use the SpeedS command. To set acceleration and deceleration, use the AccelS command.

Command	Description
Move	Move in a straight line to the specified point.
TMove	Move in a straight line to the specified point in a tool coordinate system.
Jump3CP	Jump to a point in 3 dimensions using CP motion.

6.16.4 Curves

Curves commands move the robot in a circular arc. Curves is a CP (Continuous Path) motion.

To set velocity (speed) for Curves, use the SpeedS command. To set acceleration and deceleration, use the AccelS command.

Command	Description
Arc	Move the robot through one point to another point using circular interpolation.
Arc3	Move the robot in 3D using circular interpolation.
Curve	Creates a file containing a path specification.
CVMove	Executes a path specified by Curve.

6.16.5 Joint motion

Command	Description
JTran	The JTran command can be used to move one joint of the robot to position specified in degrees or millimeters, depending on the joint type. The speed and acceleration are the same as for point to point motion commands.
PTran	The PTran command can be used to move one joint of the robot to an encoder pulse position. The speed and acceleration are the same as for point to point motion commands.
Pulse	The Pulse command can be used to move all joints of the robot to encoder pulse positions. The speed and acceleration are the same as for point to point motion commands.

6.16.6 Controlling position accuracy

Use the Fine command to adjust position accuracy for the end of a motion command. Fine specifies, for each joint, the allowable positioning error for detecting completion of any given move. The lower the Fine settings, the more accurate the final position of the joint, which can cause slower performance. Conversely, large Fine settings can speed up motion commands, but position accuracy will decrease. For many applications, the default settings can be used.

6.16.7 CP Motion Speed / Acceleration and Tool Orientation

When you attempt to change only the tool orientation while keeping the tool tip of the robot arm at the specified coordinate point or when the tool orientation variation is larger than the travel distance of the tool tip, moving the arm by normal CP motion commands will cause an increase in the variation of speed, acceleration and deceleration of tool orientation. In some cases, an error will occur.

To prevent these situations, add the ROT parameter to the CP motion commands. The arm will be moved based on the specified angular velocity and acceleration/deceleration of the main axis regarding the orientation variation.

The angular velocity and acceleration/deceleration of the main axis regarding the orientation variation should be specified with the SpeedR and AccelR commands in advance.

For example:

```
SpeedR 50              ' degree/sec
AccelR 200, 200        ' degree/sec²
Move P1 ROT
```

NOTE
☞
The tool orientation variation is normally comprised of orientation variations of more than one rotation axis.

The SpeedR and AccelR parameters specify the angular velocity and acceleration/ deceleration of the main axis regarding the orientation variation. Therefore, actual angular velocity and acceleration/deceleration of the orientation variation are different from the parameters except for the case where the rotation axis of the orientation is only one.

While the motion command with the ROT parameter is executed, the specified SpeedS and AccelS parameters are invalid.

The ROT parameter can be used with the following motion commands:

Move	BMove
Arc	TMove
Arc3	Jump3CP

6.16.8 PTP Speed / Acceleration for Small Distances

You can change the speed and acceleration for small distances using PTPBoost and PTPBoostOK. Normally, PTPBoost is not required. In certain cases, you may want to shorten the cycle time even if vibration becomes larger, or conversely you may want to reduce vibration even if cycle time becomes longer. PTPBoost is a robot parameter with values from 0 – 100 that affects the speed and acceleration for small distances. Normally, for small distance motion, the desired speed cannot be attained using the current acceleration. By increasing PTPBoost, acceralation, deceleration, and speed are increased for small distance motion. To check if a motion command will be affected by PTPBoost, use the PTPBoostOK function. See PTPBoost and PTPBoostOK in the *SPEL+ Language Reference* manual for more details.

6.17 Working with robot points

A robot point is a set of coordinates that define a position in the robot work envelope. For SCARA and Cartesian robots, a point is defined by the position data (X, Y, Z) within the reference rectangular coordinate space and the orientation data (U) which is the rotation about the Z axis of the rectangular coordinate.

For 6-Axis robots, a point is defined by the position and orientation of the tool coordinate system with respect to a reference rectangular coordinate system. The point is specified by the position data (X, Y, Z) and the orientation is specified by the orientation data (U, V, W) which correspond with *roll* (rotation about the Z axis), *pitch* (rotation about the Y axis), and *yaw* (rotation about the X axis).

The X, Y, and Z coordinates of a point are specified in millimeters. The U, V, and W coordinates are specified in degrees.

Points are referenced using the letter P followed by an integer number or integer expression or by a label defined in the point file editor or Robot Manager Jog & Teach page.

6.17.1 Defining points

You can define points in a program statement, points editor window, Robot Manager Jog and Teach page, or at the Command window.

In a program statement or at the Command window, you can assign coordinates to a point, or define a point that is the current robot arm position.

```
P1 = XY(200, 100, -25, 0)    'Assign coordinates to point P1
Pick = XY(300, 200, -45, 0) 'Assign coordinates to point pick
P10 = P*                      'Assign a point to current position
```

6.17.2 Referencing points by name

You can assign names to point numbers so you can refer to points by name in a program. Assign names from the point editor (see Editing Points) or the Robot Manager Jog and Teach page. Names must be unique for each point number when used in the same point file.
Point labels can include up to 32 alphanumeric characters and the underscore character. Characters can be upper case or lower case. Only alphabets can be used for the first letter.

```
For i = 0 To 10
   Go pick
   Jump place
Next i
```

6.17.3 Referencing points with variables

Use the letter P followed by a variable name within parentheses that represents the point number you are referencing.

```
For i = 0 To 10
   Go P(i)
Next i
```

NOTE Although you can define points at the Command window for test purposes, it is recommended that all points be defined in a program, point editor, or with the Robot Manager Jog and Teach page. Points defined at the Command window will be cleared from memory when you build a project or run a program unless you execute SavePoints.

6.17.4 Using points in a program

When starting programs, the default point file for the robot is loaded. You can also load other points in the program using the LoadPoints statement.

```
Function main
   Integer i

   LoadPoints "model1.pts"
   For i = 0 To 10
      Jump pick
      Jump place
   Next i
Fend
```

6.17.5 Saving and loading points

Use LoadPoints to load a point file in the current project. You can optionally specify the **Merge** parameter to combine points in a file with points that have already been loaded.

Use SavePoints to save the points in a point file. If the point file is in the current project, it will be updated on the PC when it is connected and the same project is open.

If the point file is not the current project, it will not be automatically updated on the PC. Use Project Synchronize to copy the file to the PC if desired.

NOTE

☞

There are limits to the number of point files and number of total point. You can register up to 100 files and 10,000 points in one project. If each number is over the limit, an error occurs.

6.17.6 Point attributes

Each point definition can optionally specify a local number and various arm orientations, depending on the robot type.   You can specify point attributes in point assignment statements or use individual statements and functions to change the attributes of a previously defined point.

Local point attribute
To specify a local coordinate system number for a point in an assignment statement, add a forward slash (/) followed by the local number after the coordinates of the point.

```
P1 = XY(300, -125.54, -42.3, 0) /1  ' P1 is in local 1
```

The local number can also be an expression enclosed in parentheses.

```
P2 = P3 /(mylocal)
```

Use the PLocal function and statement to read and set the local attribute of a point.

Hand point attribute
To specify orientation for the SCARA or 6-Axis robot, add a forward slash (/) followed by L (for Lefty hand orientation) or R (for Righty hand orientation).

```
P2 = XY(200, 100, -20, -45) /L ' Hand orientation is Lefty
P3 = XY(50, 0, 0, 0) /2 /R      ' Righty in Local 2
```

You can read and set point hand orientation using the Hand statement and function.

```
Hand P1, Righty
```

Elbow point attribute

To specify elbow orientation for the 6-Axis robot in a point assignment statement, add a forward slash (/) followed by A (Above elbow orientation) or B (Below elbow orientation),

Elbow orientation is Below.

```
P1 = XY(0, 600, 400, 90, 0, 180) /B
```

You can read and set point elbow orientation using the Elbow statement and function.

Wrist point attribute

To specify wrist orientation for the 6-Axis robot in a point assignment statement, add a forward slash (/) followed by NF (NoFlip wrist orientation) or F (Flip wrist orientation).

Wrist orientation is Flip.

```
P2 = XY(0, 600, 400, 90, 0, 180) /F
```

You can read and set point wrist orientation using the Wrist statement and function.

J4Flag and J6Flag point attributes

At some points in the work envelope, the 6-Axis robot can have the same position and orientation even if the fourth joint or the sixth joint is rotated 360 degrees. To distinguish these points, the J4Flag and J6Flag point attributes are provided.   These flags allow you to specify a position range for joint 4 and joint 6 for a given point.

To specify the J4Flag in a point assignment statement, add a forward slash (/) followed by J4F0 (-180 < the forth joint angle <= 180) or J4F1 (the forth joint angle <= -180 or 180 < the forth joint angle).

```
P2 = XY(0, 600, 400, 90, 0, 180) /J4F1
```

To specify the J6Flag in a point assignment statement, add a forward slash (/) followed by J6F0 (-180 < the sixth joint angle <= 180), J6F1 (-360 < the sixth joint angle <= -180 or 180 <the sixth joint angle <= 360), or J6Fn (-180*(n+1) < the sixth joint angle <= 180*n or 180*n <the sixth joint angle <= 180*(n+1)).

```
P2 = XY(50, 400, 400, 90, 0, 180) /J6F2
```

J1Flag and J2Flag point attributes

At some points in the work envelope, the RS series can have the same position and orientation even if the first joint or the second joint is rotated 360 degrees. To distinguish these points, the J1Flag and J2Flag point attributes are provided.   These flags allow you to specify a position range for joint 1 and joint 2 for a given point.

To specify the J1Flag in a point assignment statement, add a forward slash (/) followed by J1F0 (-90 < the first joint angle <= 270) or J1F1 (-270 <= the first joint angle <= -90 or 270 < the first joint angle <= 450).

```
P2 = XY(-175, -175, 0, 90) /J1F1
```

To specify the J2Flag in a point assignment statement, add a forward slash (/) followed by J2F0 (-180 < the second joint angle <= 180), J2F1 (-360 < the second joint angle <= -180 or 180 < the second joint angle <= 360).

```
P2 = XY(300, 175, 40, 90) /J2F1
```

6.17.7 Extracting and setting point coordinates

Use the CX, CY, CZ, CU, CV, and CW commands to get a coordinate of a point or set it.

```
xcoord = CX(P1)

P2 = XY(xcoord, 200, -20, 0)

ycoord = CY(P*)              ' Gets current Y position coordinate


CX(pick) = 25.5
CY(pick) = CY(pick) + 2.3
```

6.17.8 Alteration of points

There are several ways of modifying a point without re-teaching it. You can change one or more coordinate values with relative offsets or absolute values.

To set an absolute value for a coordinate, use a colon followed by the axis letter and the value.

To add a relative offset to a coordinate, use an axis letter followed by the offset value or expression in parentheses. If the offset is negative, then precede the axis letter with the minus sign. If parentheses are omitted, they will be automatically added.

`Go P1 -Z(20)`	Move to P1 with a z offset of -20mm
`Go P1 :Z(-25)`	Move to P1 with a z absolute position of -25mm
`Go P1 -X(20) +Y(50) :Z(-25)`	Move to P1 with offsets for X and Y relative offsets and an absolute position for Z

6.18 Input and output control

6.18.1 Hardware I/O

There are 24 DC inputs and 16 DC outputs on a standard controller. 8 inputs and 8 outputs are assigned to remote function. However, these inputs and outputs can be reset and be used as standard I/O. By purchasing I/O boards, you can add additional 128 inputs and 128 outputs. You can also add Fieldbus I/O option that allows the controller to function as a Fieldbus slave. See *I/O Systems* chapter for details.

6.18.2 Memory I/O

There are 32 bytes (256 bits) of memory I/O. Memory I/O is especially useful for synchronizing multi-tasking. Each memory bit can be treated as both an input and an output.

Use the commands with the "Mem" prefix for memory I/O.

6.18.3 I/O Commands

Command	Description
In	Reads one byte (eight bits) of input data.
InW	Reads one word (sixteen bits) of input data.
MemIn	Reads one byte (eight bits) of Memory I/O.
MemInW	Reads one word (sixteen bits) of Memory I/O.
MemOff	Turns off one Memory I/O bit.
MemOn	Turns on one Memory I/O bit.
MemSw	Read status of one bit of memory I/O.
Off	Turns off one output bit.
On	Turns on one output bit.
Out	Sets/reads one byte (eight bits) of output data.
OutW	Sets/reads one word (sixteen bits) of output data.
Oport	Reads the status of one output bit.
InBCD	Reads one byte of input data in BCD (binary coded decimal) format.
OpBCD	Outputs one byte of output data in BCD format.
Sw	Read status of one bit of hardware inputs or memory inputs.

6.19 Using Traps

Traps are used to cause program execution to jump to a label or enable a function to be called or executed as a task when a certain event occurs. Four traps are user defined and are triggered by an input condition.

Note that Traps must be re-armed before they will fire again. To re-arm a trap, execute a Trap statement at the end of the trap function.

You should keep trap functions short and avoid continuous loops. You cannot execute motion commands in trap functions.

For details on Trap statement, see the SPEL+ Language Reference manual.

Here is a simple example for a trap. In this example, when input 1 turns on, the Sw1 Trap function executes as a task.

```
Function main
  ' Arm the trap
  Trap 1 Sw(1) = On Xqt Sw1Trap
  Do
    RunCycle
  Loop
Fend
Function Sw1Trap
  ' Turn on output 1 for 2 seconds
  On 1, 2
  ' Wait for trap condition to clear
  Wait Sw(1) = Off
  ' Re-arm the trap
  Trap 1 Sw(1) = On Xqt Sw1Trap
Fend
```

6.20 Special Tasks

Each task of SPEL+ pauses by Pause input or Safety Door open and stops by Emergency Stop or Error. Therefore you cannot create a system that monitors the whole system.

RC90/RC180 controller is designed to use upper level PLC or PC to monitor the whole system.

However, you may require a system that continues at Pause input or Safety Door open and need to handle the Emergency Stop or error with the controller.

You can execute a task that continues at Pause input or Safety Door open by specifying NoPause or NoEmgAbort task type when executing a task using Xqt.

These special tasks are useful tasks but may reduce the safety of the system by using them improperly.

Be sure to understand the followings when using these tasks.

6.20.1 Precautions to Use the Special Tasks

⚠️ CAUTION	**Forced Flag** Specify Forced flag in the I/O output commands such as On/Off command to enable On/Off of the I/O outputs during Emergency Stop, Safety Door open, and error. Do not connect external equipment that operates mechanically such as actuator to the I/O output that specifies Forced flag. Connecting external equipment may cause serious safety problems and operate the external equipment during Emergency Stop, Safety Door Open, or error occurrence. Forced flag is designed to be specified for I/O outputs connected to external equipment without mechanical motion such as status display LEDs.

NoEmgAbort Task

When Emergency Stop or error occur, finish the task promptly after completing the error handling.

If you do not complete the NoEmgAbort task, the controller does not change to Ready status and you cannot cancel the Emergency Stop or the error. You cannot execute Reset command from the NoEmgAbort task to cancel the Emergency Stop or the error automatically.

NoEmgAbort task is designed for I/O process without motion and communication with external device using the Ethernet. Therefore there are commands such as robot motion commands that cannot be executed in the NoEmgAbort task. An error occurs if you use these commands. The list of these commands are in the next section.

For details, refer to EPSON RC+ 5.0 *Online Help* or *Xqt* in *SPEL⁺ Language Reference*.

Outputs off during Emergency Stop

Uncheck **Outputs off during Emergency Stop** in the Preferences page of Setup Controller to execute I/O On/Off using the NoEmgAbort task after Emergency Stop. If this check box is checked, the execution order of turn Off by the controller and turn On using the task are not guaranteed.

NoPause Task

NoPause task continues the operation during the Pause or Safety Door open condition. However, when a robot is operating NoPause task, the task pauses as the the robot pauses.

Setting of Safeguard open stops all tasks

When **Outputs off during Emergency Stop** is checked in the Preferences page of Setup Controller, NoPause task stops by Safety Door open. NoEmgAbort task continues the task.

6.20.2 Special Task Specification

Status by Event and Task

Event	Task Type		
	Normal	NoPause	NoEmgAbort
Puase Statement Pause Input Pause Button	Pause	Continue *1	Continue
Safety Door Open	Pause *2	Continue *1 *2	Continue
Error during Auto Mode	Stop	Stop	Continue
Error during Program Mode	Pause	Pause	Continue
Emergency Stop	Stop	Stop	Continue
Stop Button Stop Input	Stop	Stop	Stop
Halt Statement Halt Button	Pause	Pause	Pause
Brake Point	Pause	Pause	Pause
Switching to Teach Mode	Stop	Stop	Stop

*1 When the robot is operating, the task pauses as the robot pauses.

*2 When **Outputs off during Emergency Stop** is checked in the Preferences page of Setup Controller, normal tasks and NoPause tasks stop by Safety Door open.

Task Exection

Normal	Omit the task type in Xqt statement, or specify Normal for the task type.
	Xqt NormalTask
	Xqt NormalTask, Normal
NoPause	Specify NoPause in Xqt statement.
	Xqt NoPauseTask, NoPause
NoEmgAbort	Specify NoEmgAbort in Xqt statement.
	Xqt NoEmgAbortTask, NoEmgAbort

You cannot change the task type after executing a task.

main to main7 that are executed at the beginning of the program are executed as normal tasks.

Tasks executed in Trap Xqt are normal tasks.

Restricted Commands by Task Types

Normal	No restriction
NoPause	No restriction
NoEmgAbort	Cannot execute the following commands.
	Command for robot motion
	Commands for vision
	Reset, Xqt, Trap, etc.
	For details, refer to EPSON RC+ 5.0 *Online Help* or *Xqt* in *SPEL+ Language Reference*.

6.20.3 Special Task Example

The following example shows a program that monitors the error of the controller and switches the I/O On/Off when error occurs according to the error number.

The program example of ErrOn, EStopOn, SafetyOn are indicated in the *EPSON RC+ 5.0 SPEL+ Language Reference*.

```
Function main
    Xqt ErrorMonitor, NoEmgAbort
    :
    :
Fend

Function ErrorMonitor
    Wait ErrorOn
    If 4000 < SysErr Then
        Print "Mortion Error = ", SysErr
        Off 10, Forced
        On 12, Forced
    Else
        Print "Other Error = ", SysErr
        Off 11, Forced
        On 13, Forced
    EndIf
Fend
```

7. Building SPEL+ Applications

7.1 Designing Applications

7.1.1 Creating the simplest application

The simplest SPEL$^+$ application has one program and one point file. This is what is automatically defined for you when you create a new project. A blank program named Main.prg (Program.prg for simple mode) and a blank point file named Points.pts are created.

To write and run a simple application

1. Select New Project from the Project Menu to create a new project.

2. Write your program source code in the file that was created for you called Main.prg.

3. Teach the robot points using the Robot Manager Jog and Teach page .

4. Run the program by selecting Run Window from the Run Menu or by pressing F5 (the shortcut key for the Start command).

7.1.2 Application Layout

Before writing your application, you need to decide what your application will accomplish and how the project will be structured. Here are some general guide lines.

Programs

Each project can contain up to eight programs that can be started from the Operator Window, Remote Control, or VB Guide. Each program has a start function, as shown in the table below.

Program #	Program Name	Start Function
0	main	main
1	main1	main1
2	main2	main2
3	main3	main3
4	main4	main4
5	main5	main5
6	main6	main6
7	main7	main7

Your project must always define function main so that the main program can be started. The other programs are optional. If you use the Operator Window for your operator interface, you can define meaningful names for each of the programs used in your project in Project | Properties | Operator Window.

Operator interface

Operator Window

Use the operator window provided with EPSON RC+ 5.0. You can configure EPSON RC+ 5.0 so that after Windows starts, EPSON RC+ 5.0 will start in Auto mode, which will automatically open the Operator Window.

Operators can select from 1 to 8 programs that you specify. They can also optionally use the I/O Monitor, Robot Manager, and System History viewer.

To use the Operator Window, a PC must be connected to the controller. The RC90/RC180 Control Device must be set to PC.

For details on configuring EPSON RC+ 5.0 for auto start, see section *Start Mode* in the *Operation* chapter.

Remote Control

Use remote control to turn motors on/off, home the robot, start programs, etc. A simple push button box can be used, or a PLC can be connected.

When using Remote Control, a PC does not need to be connected to the controller. The RC90/RC180 Control Device must be set to Remote.

Operator Panel

Use the OP1 Operator Panel. With the OP1, operators can run programs and monitor I/O and task status. See the OP1 Manual for more information.

When using the Operator Panel, a PC does not need to be connected to the controller. The RC90/RC180 Control Device must be set to OP.

Windows Application using VB Guide

Use the VB Guide Option along with a Windows development tool such as Visual Basic, C#, or Visual C++. See the VB Guide Manual for more information.

Safety interface

Use guard doors, safety mats, light curtains, etc. to protect the operator from injury.

Robot Points, Pallets, Tools, Locals

Decide on which points you need for the work cell. In many cases you will only need one point file.

Take advantage of Pallets, Tools, and Locals. Time spent on using these can save hours later on the production line. For example, if your cell has many points that take a lot of time to train, consider using Locals so that if the end effecter is damaged or replaced, you only need to redefine the Locals, not retrain all of the points.

Try to design in automatic or semi-automatic procedures for calibrating tools and locals. Even if you define them manually, write instructions on how to define them so the process can be repeated easily.

Inputs and outputs

Layout your I/O early in the design stages. Use I/O labels in your programs. You must purchase additional I/O boards if you need more than 24 inputs or 16 outputs. You can also use the Fieldbus option so the controller can be a Fieldbus slave.

Peripherals

RS-232C is not standard for RC90/RC180 Controller. You can also add more ports by installing an optional RS232 expansion board. See the chapter RS-232 Communications.

The RC90 controllers have one RS-232C port as standard. In addition, you can add more four ports by mounting the optional RS-232C expansion boards.

You can use TCP/IP to connect peripheral equipment. See the chapter TCP/IP Communications.

7.2 Managing Projects

7.2.1 Overview

What is an EPSON RC+ 5.0 Project?

An EPSON RC+ 5.0 project is a collection of SPEL$^+$ program files, robot point files, and setup parameters used to run a SPEL$^+$ application.

Why do you need projects?

Projects are a safe and convenient way to manage your SPEL$^+$ applications. All the information for each application is kept in one project. By keeping all of your application code and point definitions in one project, it's easy to open a project and begin running or editing. Also, it's easy to create new versions of an application and run older versions.

Projects make it easier to maintain your application code with less chance of a program being lost.

There are also commands for copying and renaming projects, making it easy to create new projects from previous versions and for backing up projects to an external media such as a USB memory key.

What's in an EPSON RC+ 5.0 project?

Each project has its own directory on disk. This keeps all of the files for each project separate from each other. The project directories are under the \EpsonRC50\Projects directory on any drive. Sub directories may also be created.

The following paragraphs describe the components of a project.

Project file

This file contains all of the information that describes the project. This file is automatically created by EPSON RC+ 5.0. You should never edit this file. Doing so may cause errors when you open the project. This file has a .SPRJ extension.

Program source files

A program source file is an ASCII text file that contains one or more SPEL$^+$ functions. Each function in SPEL$^+$ can be run as a separate task (thread) in the controller or called from other functions.

Include files can also be used. These contain macro definitions and must be included in a program file using the #include statement.

Point files

A point file contains a list of robot points. The file extension is PTS. Each point describes a Cartesian coordinate in the robot work envelope.

The Default Project

When you create a new project, one program (Main.prg) and one point file (Points.pts) are created for you. For some users, this is all they need to get started. They don't have to configure the project. They can just type in a program, teach points using the Robot Manager Jog and Teach Tool, and run the program.

7.2.2 Creating a new project

Projects can be created on any drive on your system. Projects always reside in the \EpsonRC50\Projects directory or sub-directory.
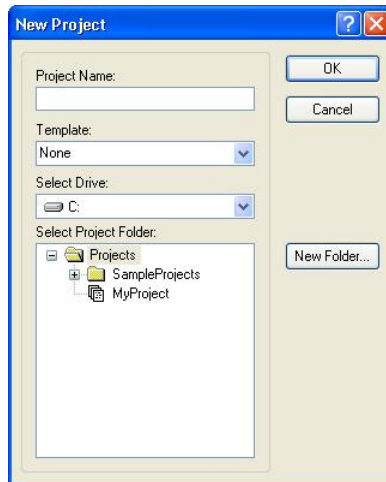
To create a new project

1. Select New Project from the Project Menu. The New Project dialog box will appear.

2. Select the disk drive where you want the project to be stored on.

3. Select the project directory or create a new directory by clicking the **New Folder** button after selecting the parent directory.

4. Type in the name for the new project.

5. Optionally, select a template to base the project on.

6. Choose **Ok** to create the project.

7.2.3 Configuring a project

Each application project you create must be configured properly before you can run the application.

There are two commands in the Project Menu that allow you to configure a project: Edit and Properties.

Editing a project

Select Edit from the Project Menu to open the Edit Project dialog. From this dialog, you configure which programs, include files, and point files are used in the current project.
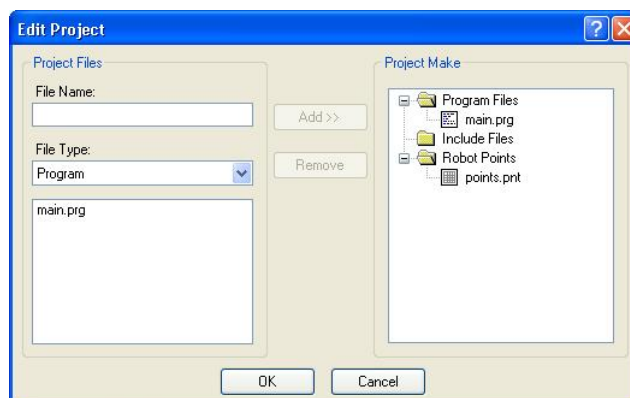
For details on Project | Edit, see the Edit Command (Project Menu) in the chapter *EPSON RC+ 5.0 GUI*.

7.2.4 Building a project

Before you can run any of the code in your application, you must build the project.

To build your application project

Select Build from the Project Menu or click on the Build button on the toolbar.

Or

Select Rebuild from the Project Menu. This will rebuild the entire project.

Or

Select Run Window from the Run Menu or click the Run button on the toolbar. The project will be built before the Run Window appears.

Or

Select Operator Window from the Run Menu. The project will be built before the Operator Window appears.

After the files have been compiled and linked, the project files are sent to the controller.

Status Pane

This window shows progress messages and error messages during project build.

Double click this line to go to the source code.
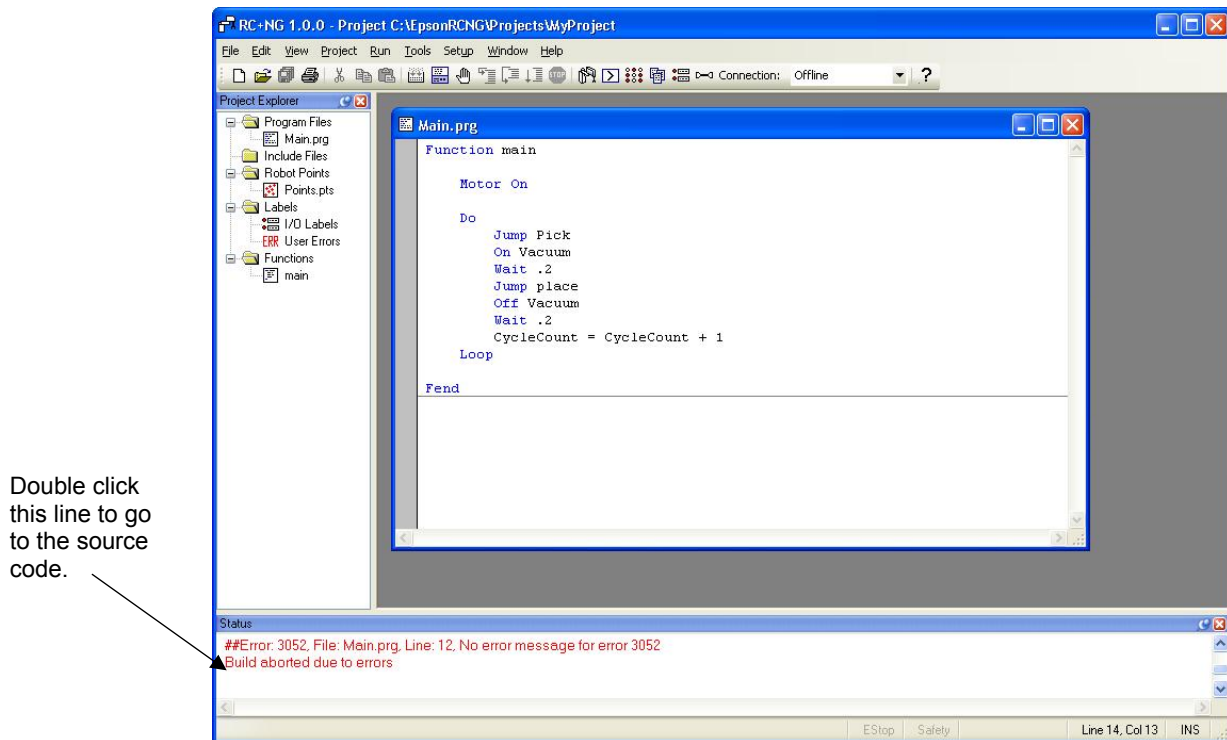
When errors occur during the build process, a message is displayed that includes the error number, program name, and line number. Double click on the line with the error to go directly to the source code that caused the error.

7.2.5 Making copies of projects

Use the Copy Project command in the Project Menu to copy all of the files in the current project to another disk drive or directory. You can also save the project under a different name. This command is useful for making backups or for transferring a project to an external media such as a USB memory key so that it can be loaded on another machine.

Use the Save As command in the Project Menu to save the current project under a different name or drive and re-open it using the new name and location. This command is useful for making a test version of a project without making changes to the original project.

7.2.6 Backing up a project

To make a backup copy of the current project, use the Copy Project command in the Project Menu to copy the project to another disk drive. Typically you will want to copy the project to a USB memory key.

To restore a project from the external media such as a USB memory key (or different hard drive), open the target project, then use the Copy Project command to copy the project to the original disk

7.2.7 Converting a simple mode project

A project created in Simple mode can be converted to Normal mode by opening it while operating in Normal mode. However, once the project is transferred to Normal mode, it cannot be used in Simple mode.

7.3 Editing Programs

Before you can edit a program, it must be in the current project and opened in a program editor window. When you first create a new project, this is done for you automatically. You can open several program editors during one session.

To open a program for editing

1. Select Open from the File Menu.

2. Select the file(s) you want to open.

3. Choose **OK** to open the file.

7.3.1 Program rules

A program contain one or more SPEL$^+$ function definitions.

Lines can be blank. You can insert any number of blank lines to separate subroutines and functions, if desired.

The maximum length for each line is 200 characters, including the line number, if used.

The maximum line number is 32767. You will get an error if you try to run a program with a line number that is too large.

7.3.2 Typing in program code

Each line can optionally start with a line number.

You can enter program statements in upper or lower case. Whenever you leave a line that has been changed, the line will be formatted. SPEL$^+$ keywords are case formatted and spaces are inserted around operators and after semi-colons and commas.

Consider using mixed case or lower case for variables and function names instead of all CAPs. This will make your code easier to read.

Use indentation for statements within loops. The **Auto Indent** feature automatically moves the cursor under the start of the previous line. It also indents lines after If, Else, For, Select, Case, and While statements.

```
For i = 1 To 10
    Jump P(i)
    Jump P0
Next i
```

Use the **Auto End Construct** feature to automatically add the end construct statement. For example, when you enter a For statement and press Enter, a Next statement is automatically created with an indented blank line above it.

7.3.3 Syntax Help

When you type in a SPEL+ keyword or user function, the syntax help window will appear to show the syntax of the statement or function.   After the statement is entered, the syntax helper will automatically close, or you can press the Esc key to close it. You can enable / disable Syntax Help from the Setup | Preferences | Editor tab.

A list box will be displayed for some parameters as you type. To select a value in the list, use the up and down arrow keys, or type the first few characters, to highlight the desired item, then press Tab to select the item. You can also type in a value not shown in the list, such as a variable or literal constant. Press Esc to hide the list box. In addition to Tab, you can use a comma or period to select an item.   In the example shown below, the first parameter of the On statement can be an output label, so a list of output labels in the current project is displayed.
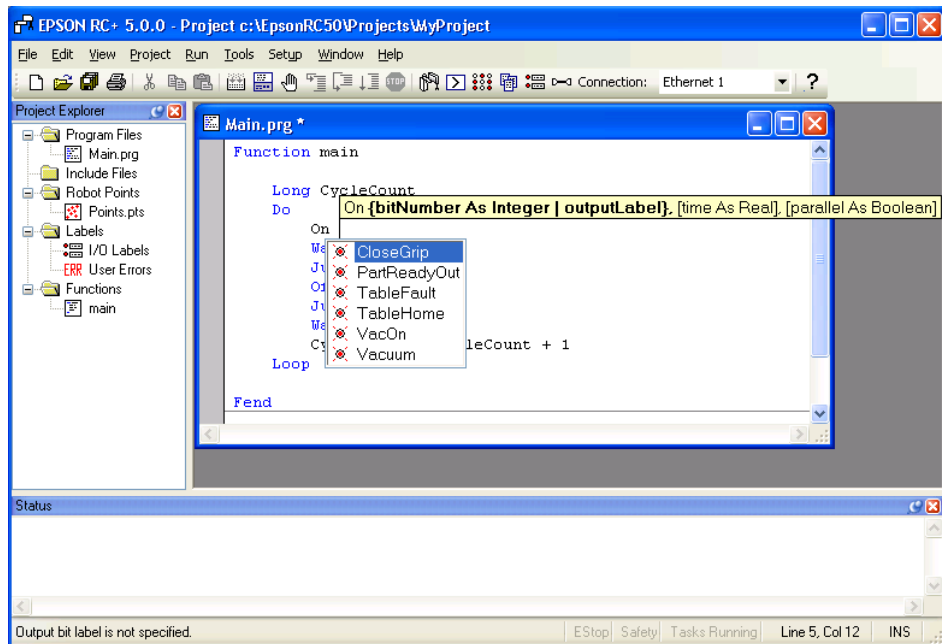
7.3.4 Syntax Errors

When a syntax error is detected, the line with the error will be displayed in red. If the caret is placed on the line with the error, then a brief message will be displayed on the status bar. For example, in the program shown below, the message "Expression expected" is displayed on the status bar.

Red indicates syntax error

Error message in status bar
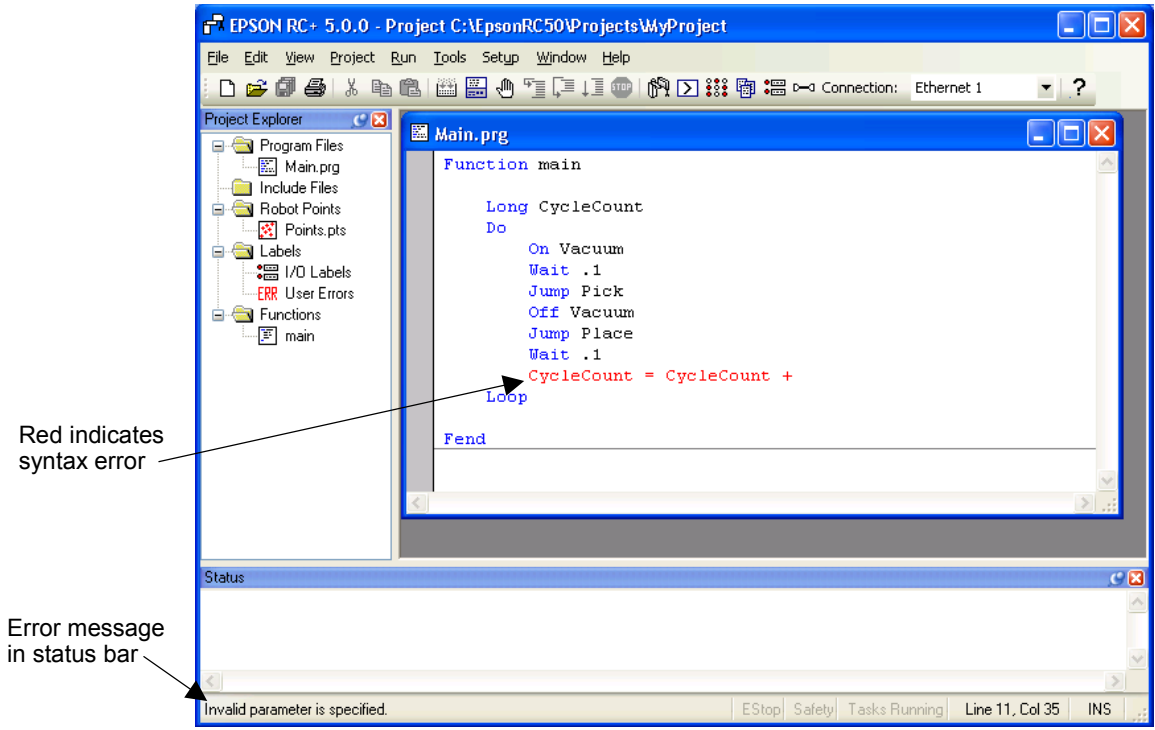
7.4 Editing Points

EPSON RC+ 5.0 provides a spreadsheet editor for robot points. You can define new points or cut, copy, and paste points from one point file to another, including between projects.

Robot teach point definitions are kept in files on your PC and in the robot controller. EPSON RC+ 5.0 always keeps project point files in the robot controller up to date with the files on the PC.

To open a point file for editing

1. Select Open from the File Menu to show the Open dialog box.

2. Choose the **Points** radio button. You will see a list of point file names in the files list box.

3. Select the point file you want to edit by clicking on the name.

4. Press the **Open** button to open the file. You will see a spread sheet window for the point file you selected.

The robot points spread sheet window
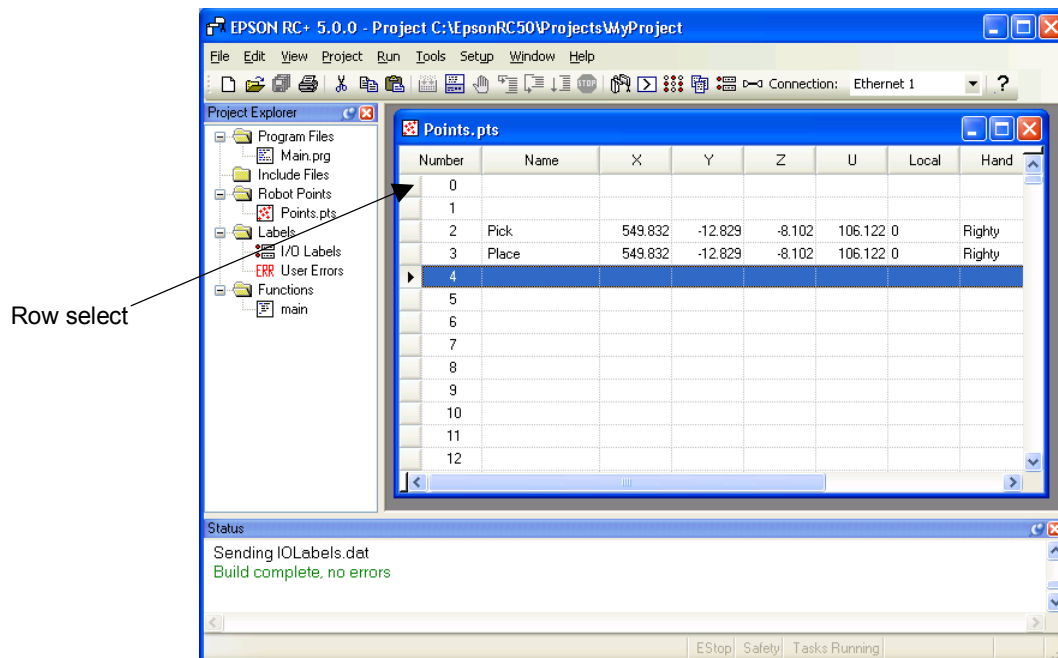
The spreadsheet window contains one row for each point in the file. The spreadsheet always contains rows for all points, even if they are not defined. The cells for an undefined point are blank.

Row select column	This is the first column on the left. Click on this column to select a row.
Number column	Point number. Range is from 0 to 999.
Name column	Name of the point.
Coordinate columns	Coordinates in millimeters for X, Y, Z and degrees for U, V, and W.
Local number column	Local number drop down list. Range is from 0 to 15.
Hand column	Drop down list with two values for robot hand orientation: Lefty and Righty.
Elbow column	Drop down list with two values for robot elbow orientation: Above and Below. This column is shown only for 6-axis robots.

Wrist column	Drop down list with two values for robot wrist orientation: Flip and NoFlip. This column is shown only for 6-axis robots.
J4Flag column	Drop down list with two values for robot J4Flag: 0 and 1. This column is shown only for 6-axis robots.
J6Flag column	Drop down list with values for robot J6Flag: 0 - 127. This column is shown only for 6-axis robots.
J1Flag column	Drop down list with values for robot J1Flag: 0 and 1. This column is shown only for RS series.
J2Flag column	Drop down list with values for robot J2Flag: 0 and 1. This column is shown only for RS series.

To select one or more rows

Click on the row select column (first column on the left) to select a row. To select more than one row, point to the row select column of the first row you want to select. Hold down the left mouse button and drag the mouse down or up to select more rows.

To select all rows

Execute Select All from the Edit Menu, or type Ctrl+A.

To define a new point

Using the mouse put the spreadsheet cursor anywhere on the row of the point you want to define and start entering information for the point. This automatically defines the point, which means it will be sent to the robot controller at the next project build or Jog and Teach command.

For example, put the cursor in the Name column and type in a name for the point.

Press the TAB key to move to the X coordinate column. Type a coordinate value, then press Enter. You will see zeros automatically entered for all of the other coordinates of the point. This indicates that the point as been defined.

To delete a point

Select the row containing the point and cut it by selecting Cut from the Edit Menu or by typing Ctrl+X.

To cut and paste points

1. Select one or more rows and execute either the Cut or Copy command from the Edit Menu.

2. Select the row where you want to start the paste.

3. Execute the Paste command from the Edit Menu.

7.5 Running and Debugging Programs

You can run programs from the Run Window or from the Operator Window. The Run Window is used primarily for testing and debugging. The Operator Window is used as an operator interface for simple applications or demos.

To run a program

Select Run Window from the Run Menu. This command will build the project (if required) and open the Run Window. The Run Window allows you to choose which function to execute. Select a function, then click **Start**.

7.5.1 The Run Window

The Run window includes controls for running the programs in the current project.

To open the Run window

Select Run Window from the Run Menu, or click on the Run button ▦ on the toolbar. If necessary, all changed open files will be saved and the project will be built. If the build is successful, the Run window will appear.

To close the Run window

Choose Close from the File Menu or double click on the Control menu in the upper left hand corner of the window.

Item	Description
Text area	This is the area that takes up most of the run window. Output from your programs is displayed here. When your program uses an Input statement, you can type in the requested input from this text box. You can use the scroll bars to view the entire text buffer. If an error occurs while running a program, the error number, line number and function name will be displayed in this text area. You can double click on the line where the error is displayed to directly go to the source line that caused the problem. You can also get help on the error that occurred by putting the cursor anywhere on the line and pressing F1.
Function	Select a function to start. Functions are sorted alphabetically. Function main is selected by default.
Low Power	When this box is checked, SPEL+ ignores the Power High command. This allows you to run your program in low power mode to verify operation without having to change the program.
Speed Factor	Specifies the robot motion speed factor. The speed factor is a percentage of maximum point to point speed and linear interpolated speed. For example, if you program executes Speed 80 and the speed factor is 50%, the robot will move at speed 40.
Start	Starts the function shown in the function drop down list.
Stop	Stops all tasks. If the robot is executing a motion command when this button is pressed, the robot will decelerate to a stop.
Pause	Pause all tasks with pause enabled. Activates the **Continue** button. If the robot is executing a motion command when this button is pressed, the robot will decelerate to a stop.
Continue	Continue paused tasks.
CTRL+C	Same as **Stop** button.

7.5.2 Debugging

EPSON RC+ 5.0 supports source level debugging. You can set breakpoints and step through your source code. You can also suspend a task using the Task Manager Halt command and then step through the code.

Setting and clearing breakpoints

Open the program where you want to set a breakpoint, then click on the line where you want to stop. Use one of the following methods to set a breakpoint:

- If Margin Indicators are enabled, then click in the margin next to the line on the left. You will see a breakpoint symbol next to the line.

Or

- Type F9.

Or

- Select Toggle Breakpoint from the Run Menu.

Execute one of the methods above to clear a breakpoint, or select Clear All Breakpoints from the Run Menu.

You cannot set a breakpoint on non-executing statements, such as #define, #include, or blank lines.

You can set a breakpoint while a task is running. After setting the breakpoint, the task will halt when the line is reached during execution. You can also remove a breakpoint while a task is running.

When a breakpoint is reached, the editor window containing the program source line at the breakpoint is opened and the line is highlighted in yellow. The task number is shown in the title of the editor window.

If a breakpoint is reached by more than one task, then an editor window will be opened for each task that is at the breakpoint. This allows you to step through each task that reached the breakpoint.
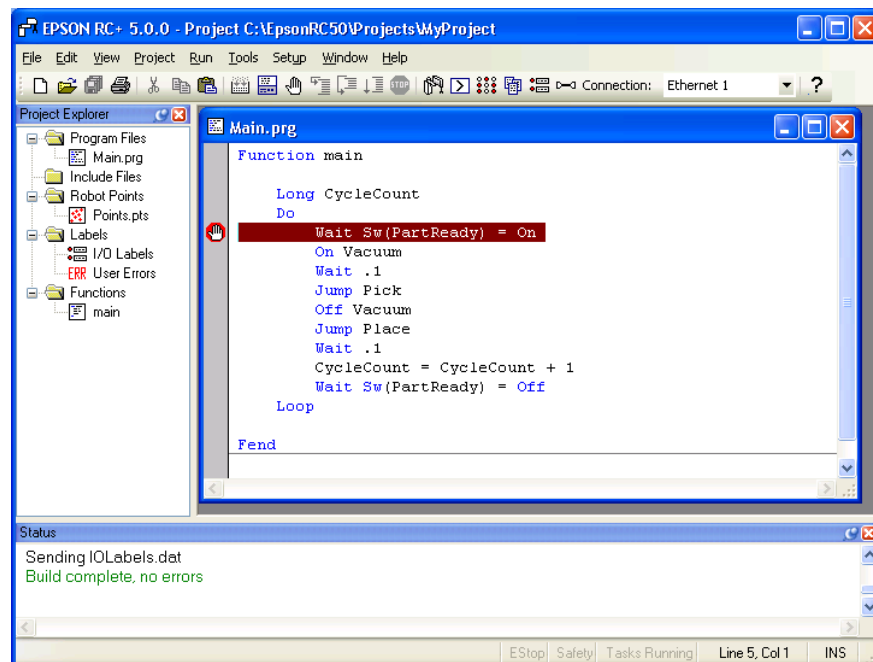
Stepping through a program

There are three commands on the Run Menu that are used for stepping through code.

Step Into steps through each line and also steps into functions when a step is executed on a Call statement.

Step Over steps through each line but when a Call statement is encountered, the function in the statement is executed completely.

Walk executes lines until after the next motion command and then halts the task. It will halt after the next output command if the Setup | Controller | Preferences *Walk stops for outputs* checkbox is checked.

To step through code, you must either set a breakpoint and run until the breakpoint is reached, or suspend a task from the Task Manager using the Halt button.

Viewing the Call Stack

Sometimes you may want to examine the call stack for the current task after you halt the task from the task manager, or reach a break point.

To view the call stack, select Call Stack from the Run Menu. The Call Stack list will be displayed, as shown below.
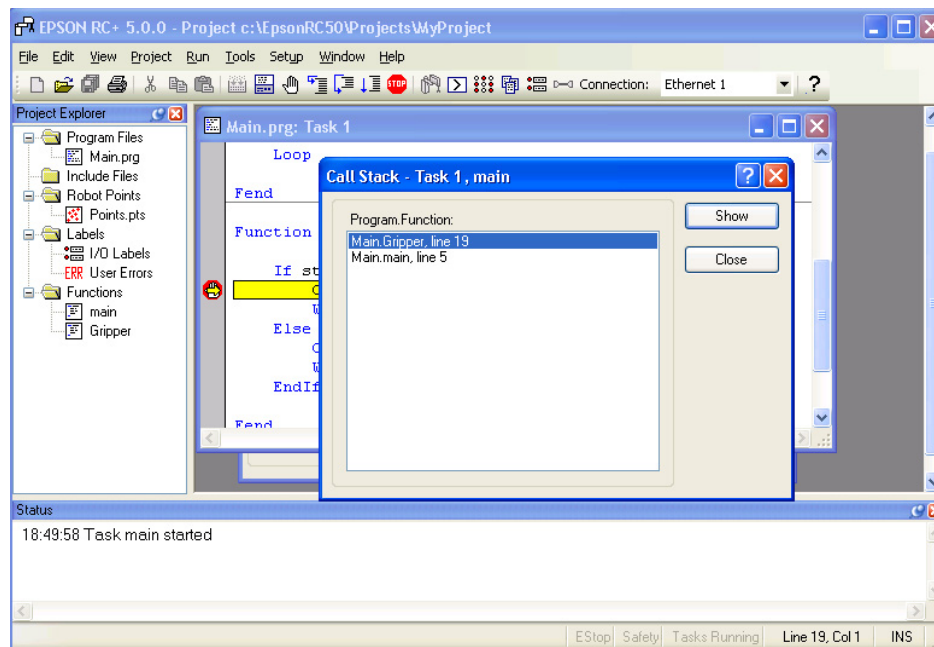
After you select a function in the Call Stack list, the function will be displayed in a program window and an arrow in the left margin will point to the line where the next function in the call stack is being called. In the example below, the arrow in the SingleCycle function is pointing to the Gripper On statement to indicate that Gripper was called from SingleCycle.
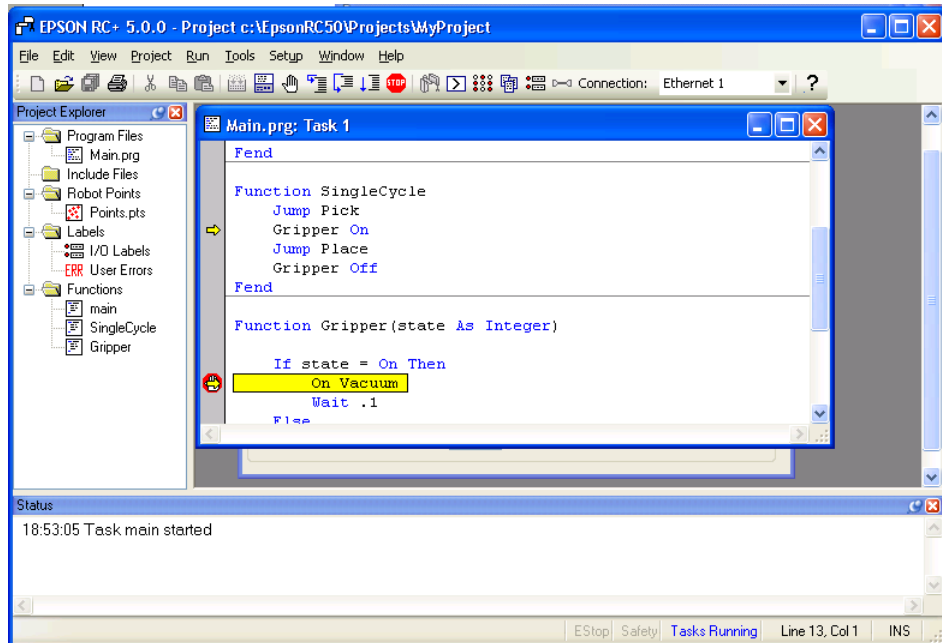
Displaying variables

To view variable values, you can do one of the following:

1. When a task is halted by halt or breakpoint, you can view the value for a variable by moving the mouse cursor over the variable name. The value will be displayed in a tool tip type window beneath the variable name.
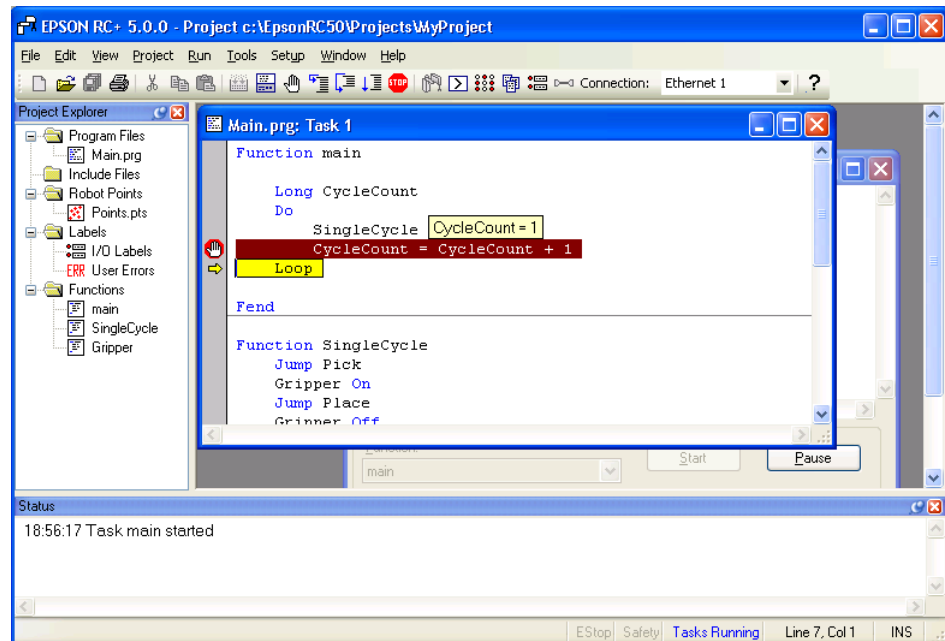
2. Select Display Variables from the Run Menu to display the variable display dialog. This dialog has three tabs for viewing Global, Module, and Local variables.
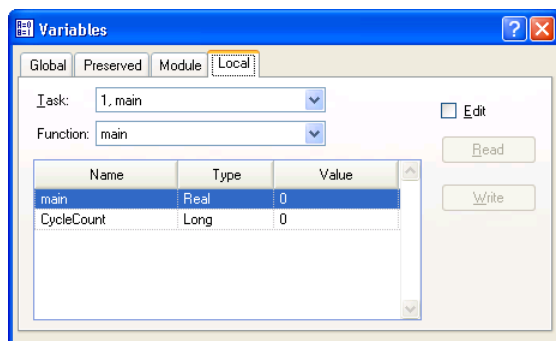
You can change the value of a variable by checking the Edit check box, then type in the new value in the value column. Next, click the Write button to change the variable. When the Edit box is checked, the variable values are not automatically updated. You can click the Read button to update all values.

7.6 The Operator Window

The Operator Window can be used as a simple interface for operators. You can configure EPSON RC+ 5.0 to open only the Operator Window when started. In addition, when Remote Control is being used, the Operator Window can be displayed for monitoring purposes.

| Item	Description
Program to Run	Start the program to run.
Start	Start the first function in the main program.
Stop	Stops all tasks.
Pause	Pause all tasks that are enabled for pause.
Continue	Continue paused tasks.
Robot Manager	Opens the Robot Manager dialog in operator mode.
I/O Monitor	Opens the I/O Monitor in operator mode. This window can remain open while programs are running.
System History	Opens the System History window. This window can remain open while programs are running.
Simulator	Opens the Simulator Window.
Status Bar	The status bar is located at the top of the window and shows emergency stop and safeguard status. In addition, if a warning is detected from the controller (such as low encoder battery), a warning label will be displayed on the right side of the status bar. If the mouse is over this label, you can see the warning error message. When there is no warning, the warning label is hidden.

7.6.1 Operator Window Configuration

You can configure the Operator Window from the Operator Window pages in Project | Properties.
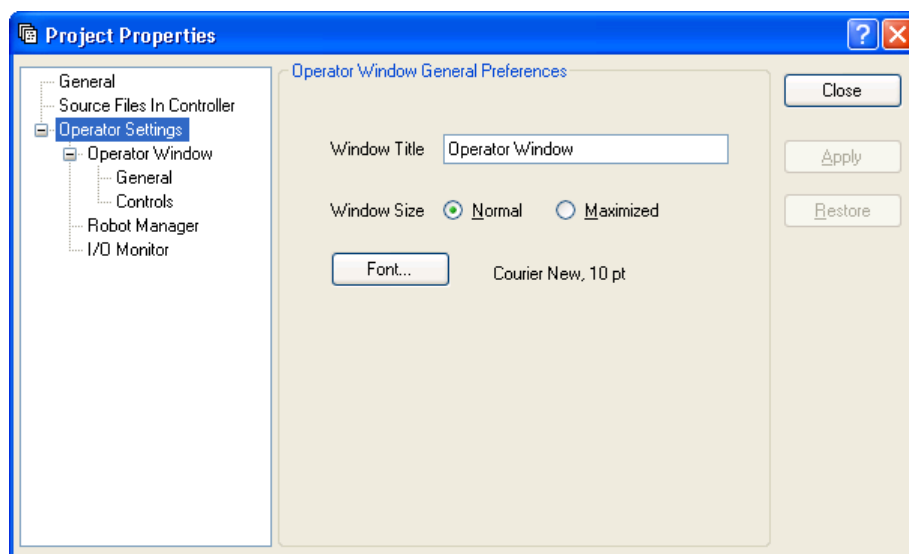
There are several settings for operator Robot Manager and I/O Monitor.

See *Properties Command (Project Menu)* section in the chapter *EPSON RC+ 5.0 GUI* for details.

7.7 Using Remote Control

You can design your application to be run from external equipment using hardware I/O control. This includes push button boxes, PLCs, and other PC systems.

See the chapter *Remote Control* for details.

8. Simulator

8.1 Simulator Functions

Simulator functions enable easy robot motion checking on your PC, which gives you flexibility to consider the system layout, measure the operation time, and create robot programs.

They are useful from the introduction stage to the launch of robot system.

The Simulator is supported in EPSON RC+ 5.0 Ver.5.4 or later.

8.1.1 Overview

The following are the major simulator functions:

Robot motion 3D display

- Shows robot orientation and motion in a 3D display from various viewpoints.
- Offers accurate display data based on design data.

Interference check

- Checks whether the robot (including the hand) interferes with itself or its peripherals.

Robot operation time prediction

- Predicts the robot operation time for a program.
- Considers the speed setting (Speed, etc.) and acceleration / deceleration setting (Accel, etc.) when predicting the robot motion time.

SPEL+ program execution

- Allows you to create, execute, and debug SPEL$^+$ programs.

The restrictions on the simulator functions are described in *8.4 Specification and Restriction of Simulator*.

Recommended specification

When using CAD data, we recommend using a PC running Windows 7 Professional 32 bit version.

OS	Windows 7 Professional 32 bit version
CPU	Core i3 or more
Memory	1 GB or more
Hard disk spare capacity	4 GB or more
Graphic	DirectX9 or later must be available. OpenGL2.1 or later must be supported.

Minimum specification

To use one robot with several peripherals and operate them in a simple manner, the following environment is required.

OS	Windows 7 Professional 32 bit version Windows XP Professional 32 bit version, SP3 or later Windows Vista Business 32 bit version, SP2 or later
CPU	1.6 GHz or more, 32 bit (x86) Multi Core Processor
Memory	512 MB or more
Hard disk spare capacity	4 GB or more

8.2 Using the Simulator

You can try the simulation functions using the provided sample virtual controllers and projects. See section *8.2.1 Working with the samples*.

You can also select the robot model and build your own system. See section *8.2.2 Working with a user created system*.

8.2.1 Working with the samples

You can operate a robot easily using the samples provided. Follow the steps below:

1. Connect with a sample virtual controller (robot)
2. Open the corresponding sample project
3. Display the simulator window
4. Operate the robot by executing a program
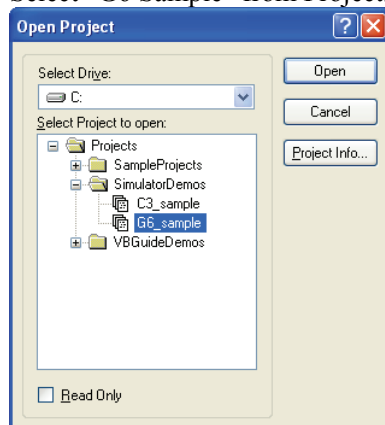5. Next step

1. Connect with a sample virtual controller

Select "G6 Sample" from the EPSON RC+ 5.0 Tool bar-<Connection> list box.
When the connection is complete, the <Connection> list box shows "G6 Sample".

2. Open a corresponding sample project

(1) Click the EPSON RC+ 5.0 menu-[Project]-[Open…].
(2) Select "G6 Sample" from Projects\SimulatorDemos.

(3) Click the <Open> button. Then, the following program window appears.

3. Display the simulator window

Click the Tool bar-<Simulator > button. The [Robot Simulator] window appears.
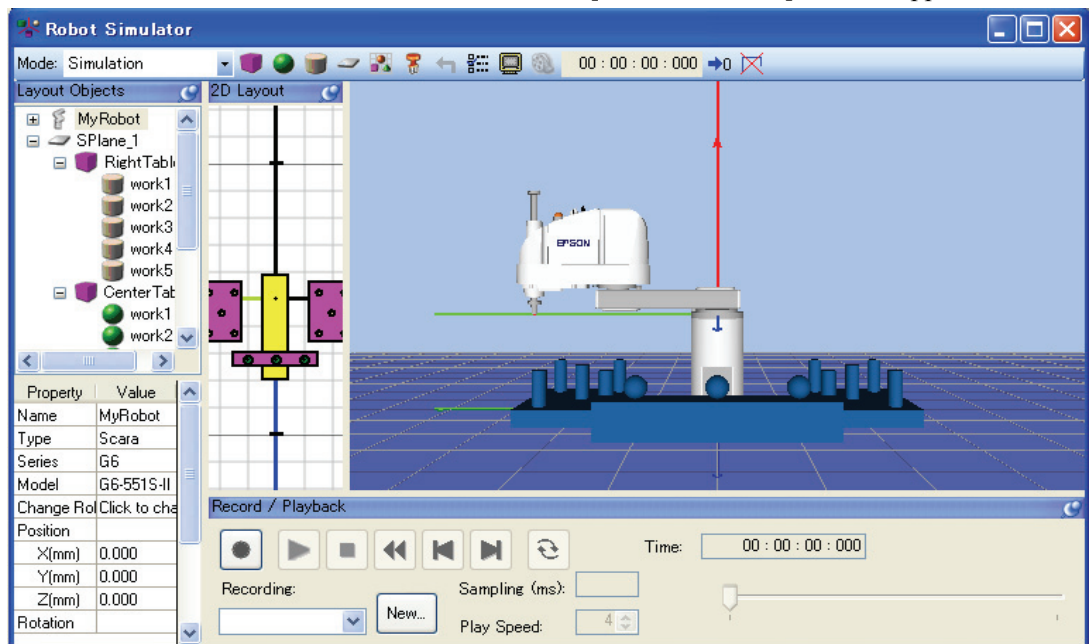
4. Operate the robot by executing program

(1) Click the Tool bar-<Run Window > button. The project will be built and the [Run] window will appear.
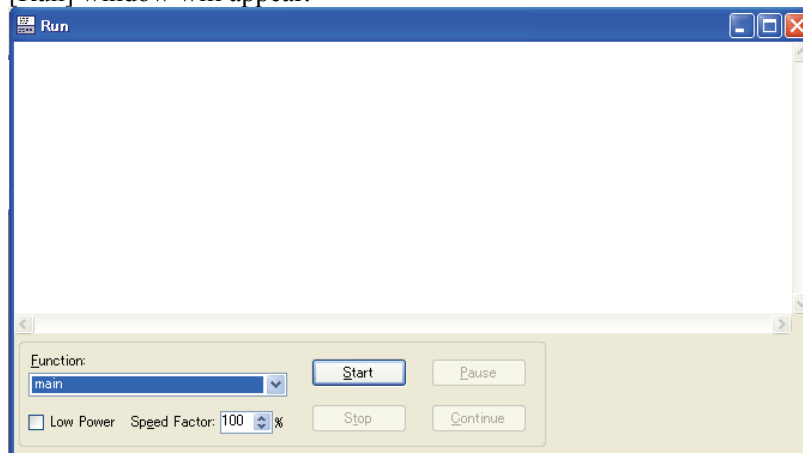
(2) Click the <Start> button.
The message "Are you ready to start?" appears. Click the <Yes> button.
The program starts and the robot moves in the 3D display.

5. Next step

If you want to change the sample, follow the steps in *8.2.2 Using with a user created system - Steps 5 to 7*. If you want to create your own system, start from *Step 1*.

If you want to change the sample virtual controller, follow the steps in *8.3.4 Virtual controller - Copy the sample or configured virtual controller* and change the copied sample.

8.2.2 Working with a user created system

You can create your own system and simulate the robot operation on your PC.
Follow the steps below:

1. Create a new virtual controller (Connection setting)
2. Connect with the virtual controller
3. Configure a robot
4. Display the simulator window
5. Create and place objects
6. Create a project and program
7. Operate the robot by executing the program
8. Measure the robot operation time
9. Test for collisions

1. Create a new virtual controller (Connection setting)

(1) Click the EPSON RC+ 5.0 Tool bar-<Setup PC to robot controller communications. > button. Then, the [PC to Controller Communications] dialog appears.

Click the <Add> button.
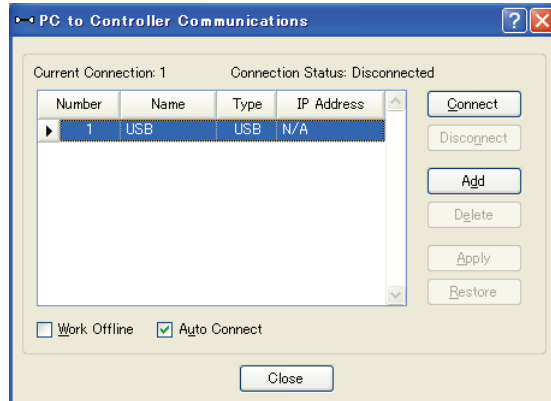
(2)   The [New Controller Connection] dialog appears.
Select the <Connection to new virtual controller> option button and click the <OK> button.
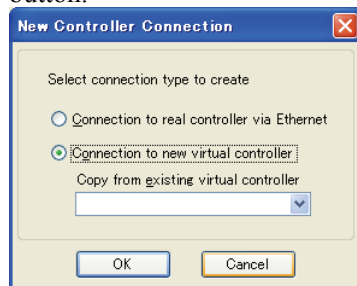
(3)   A new virtual controller named "Virtual 1" is created. Click the <Apply> button.
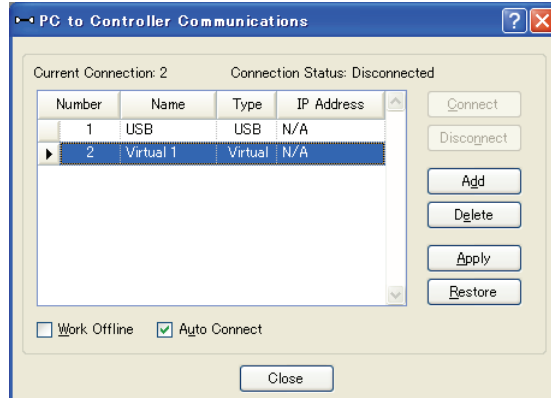
(4)   Close the dialog to return to the EPSON RC+ 5.0 main window.

2. Connect with the virtual controller

(1)   Select the created "Virtual 1" connection from the EPSON RC+ 5.0 Tool bar-
<Connection> list box. When the connection is complete, the <Connection> list box shows "Virtual 1".
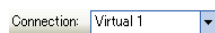
(2)   "Virtual 1" has no configured robot, so the following warning appears.
Click the <OK> button.
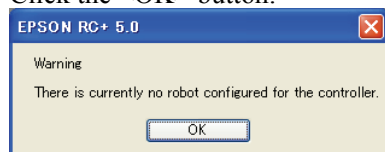
3. Configure a robot

In this tutorial, a G6-551S robot model is used.

(1) Select the EPSON RC+ 5.0 menu-[Setup]-[Controller].

(2) Select the [Robot]-[Model] from the tree on the left side of the window. The [Robot Model] page appears.
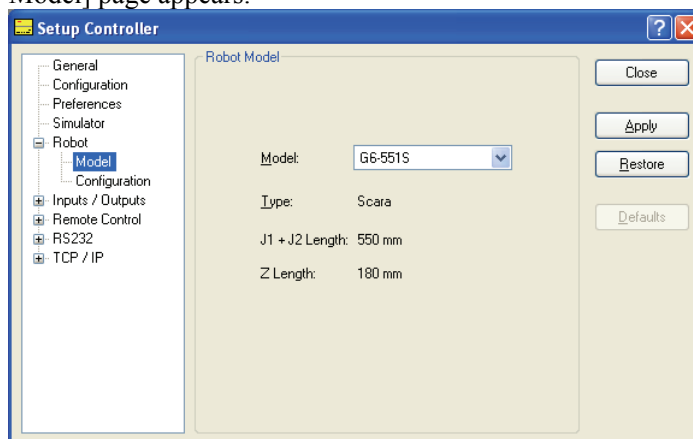
(3) Select "G6-551S" from the [Model] list box.

(4) Click the <Apply> button. The message "Restarting Controller" appears.

(5) When the message disappears, close the window and go back to the EPSON RC+ 5.0 main window.

4. Display the [Robot Simulator] window

(1) Click the Tool bar-<Simulator 🐾> button. The following window appears.
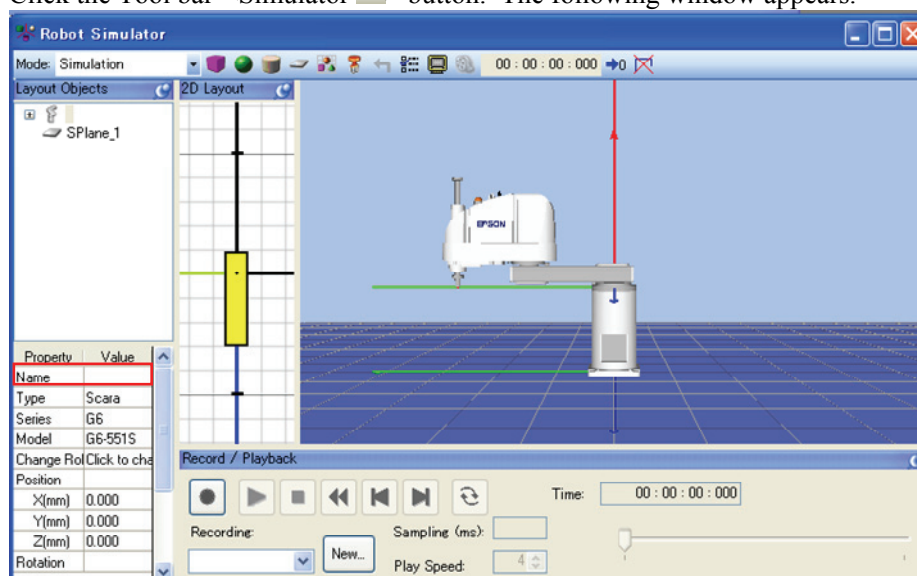
(2) Configure the robot name.
Select the robot in the [Layout Objects] and set its name in the [Property]-[Name] on the lower left of the window. For this tutorial, enter "MyRobot".

5. Place the objects

For this tutorial, we will add a box to the layout.

(1) Drag & drop the <Box   > button on the Tool bar to the [2D Layout] panel.
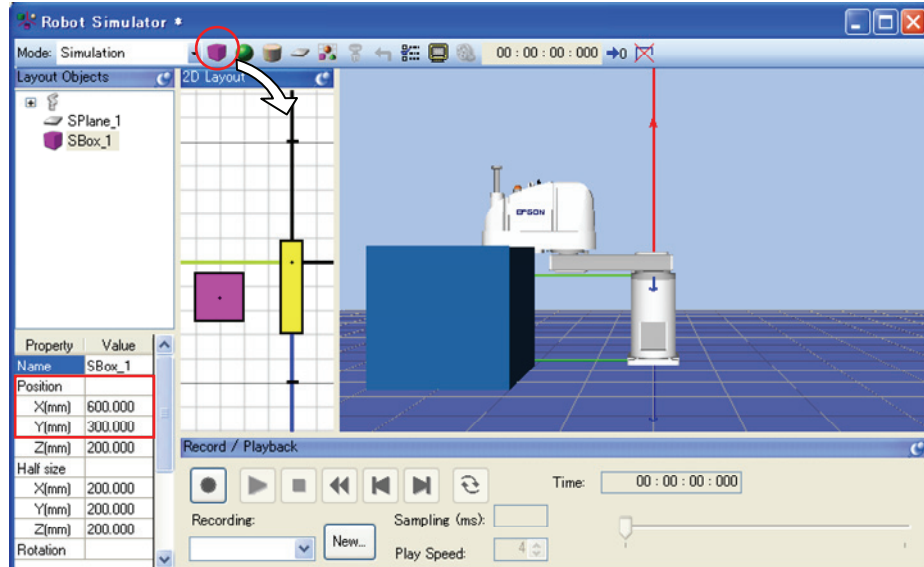
(2) Select "SBox_1" from the [Layout Objects] and change the [Property]-[Position]. For this tutorial, enter X = 600, Y = 300.

TIP
☞
To change the position, you can also drag the objects in the [2D Layout] panel.

To save the layout change, execute the EPSON RC+ 5.0 menu-[File]-[Save].

6. Create a project and program

(1) Create a new project

(1)-1 Click the EPSON RC+ 5.0 menu-[Project]-[New Project].

(1)-2 Enter a new project name. For this tutorial, enter "Test".

(1)-3 Click the <OK> button. Then, the project "Test" is created.

(2) Operate the robot and teach points.

(2)-1 Click the Tool bar-<Robot Manager   > button. The [Robot Manager] window appears.

(2)-2 Select the [Control Panel] tab and click the <MOTOR ON> button.
The message appears to confirm the operation. Click the <Yes> button.

EPSON RC+ 5.0 (Ver.5.4) User's Guide Rev.4

(2)-3 Select the [Jog & Teach] tab. The following window appears.
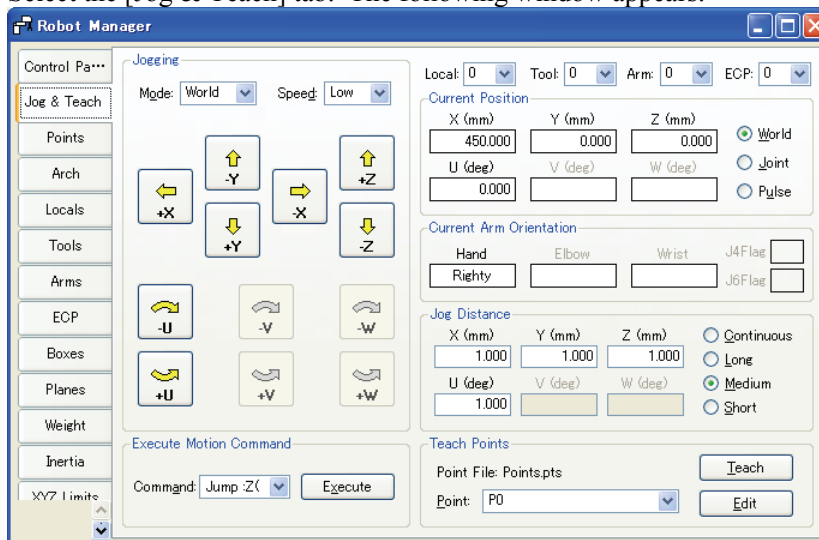
(2)-4 In the [Robot Simulator] window, move the robot joint to a point where it does not interfere with the box.
To move the robot joint, drag the robot joint while simultaneously pressing the <Ctrl> key.
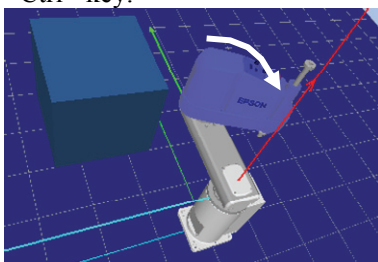
(2)-5 Go back to the [Robot Manager] window and click the <Teach> button on the lower right. The message appears to confirm the operation. Click the <Yes> button.

(2)-6 The [New Point Information] dialog appears. Click the <OK> button.

(2)-7 Select "P1 - (undefined)" from the [Point] list box on the lower right.

(2)-8 In the [Robot Simulator] window, drag the robot joint while simultaneously pressing the <Ctrl> key to another point without interfering with the box.

(2)-9 Go back to the [Robot Manager] window and click the <Teach> button. The message appears to confirm the operation. Click the <Yes> button.

(2)-10 The [New Point Information] window appears. Click the <OK> button.

(2)-11 Click the Tool bar-<Save all files> button to save the P0 and P1 data.

TIP

☞ You can also use the [Jog & Teach] window to move the robot.

(3) Create and execute a program with robot motion.

(3)-1 Create the following program in the program "Main.prg".

```
Function main
    Go P0
    Go P1
Fend
```

(3)-2 Click the Tool bar-<Build> button to build the program.
When the program building is completed, the message "Build complete, no errors" appears in the [Status] window.

7. Operate the robot by executing program

(1) Click the Tool bar-<Run Window> button. The following dialog appears.
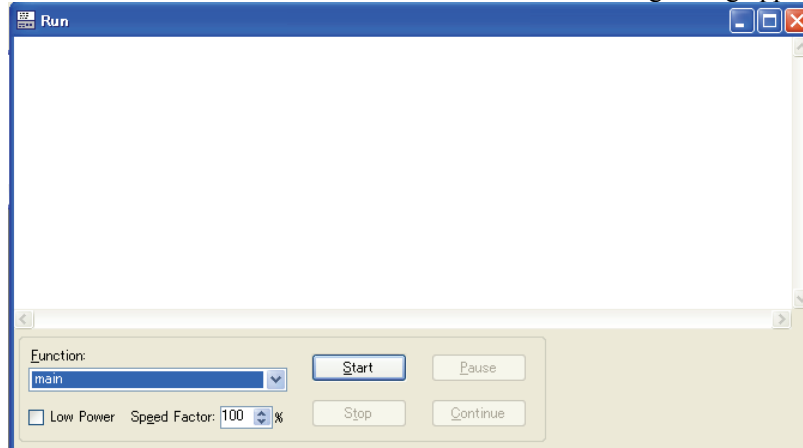
(2) Click the <Start> button.
The message "Are you ready to start?" appears. Click <Yes> button.
The program starts and the robot moves in the 3D display.

8. Measure the robot operation time

The elapsed program run time (cycle time) is displayed in the Tool bar of the [Robot Simulator] window.

It is the execution time of the program from start to finish.

The following describes how to measure the operation time between two points (P0 → P1).

(1) Change the program in the "Main.prg" file to the following program.

```
Function main
    Motor On
    Power High
    Speed 100
    Accel 100,100
    Go P0
Fend

Function main2
    Go P1
Fend
```

(2) Click the Tool bar-<Build> button to build the project.
When the project build is complete, the message "Build complete, no errors" appears in the [Status] window.

(3) Click the Tool bar-<Run Window> button.

(4) Confirm that "main" is selected in the [Function] dropdown list and click the <Start> button.
The message "Are you ready to start?" appears. Click <Yes> button.
The program starts and the robot goes to P0, the point to start the time measurement, in the 3D display.

(5) Select "main2" in the [Function] dropdown list.

(6) Click the <Start> button.
The message "Are you ready to start?" appears. Click the <Yes> button.
The program starts and the robot moves in the 3D display.
Now, the cycle time displayed in the Tool bar is the execution time to move the robot from P0 to P1.

NOTE ☞ When you operate the real robot, the actual cycle time will be longer than the simulated cycle time according to the model, Fine, load settings. For details, refer to *8.4 Specification and Restriction of Simulator*.

Also, when Speed, Accel values in the program are changed, the cycle time will reflect it.

TIP ☞ Motion command includes Move and Jump as well as Go.

For the information on how to use these commands, refer to *Online Help* or *SPEL⁺ Language Reference manual*.

9. Test the collision detection

(1) Go back to the [Robot Simulator] window.

(2) Drag the robot joint while simultaneously pressing the <Ctrl> key to a point where it interferes with the box. When the robot joint hits the box, the display turns to red.

(3) In the [Robot Manager] window, select "P2 - (undefined)" from the [Point] list box on the lower right of the window.

(4) Click the <Teach> button.
The message to confirm the operation appears. Click the <Yes> button.

(5) The [New Point Information] dialog appears. Click the <OK> button.

(6) Click the Tool bar-<Save all files> button and save the P2 information.

(7) Go back to the [Robot Simulator] window and drag the robot joint while simultaneously pressing the <Ctrl> key to the point where it does not interfere with the box.

(8) Click the Tool bar-<Reset Collision ↰ > button. Then, the red display returns to normal.

(9) Add the following function to the "Main.prg" program file.

```
Function main3
    Go P2
Fend
```

(10) Click the Tool bar-<Build> button to build the project.
When the project build is complete, the message "Build complete, no errors" appears in the [Status] window.

(11) Click the Tool bar-<Run Window> button.

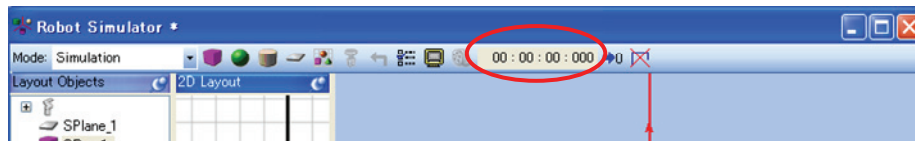(12) Select "main2" in the [Function].

(13) Click the <Start> button. The message "Are you ready to start?" appears. Click the <Yes> button. The program starts and the robot moves in the 3D display. When the robot joint hits the box, the display turns to red.

TIP ☞ When a collision happens, the users can stop the controller program execution with an error. For details, refer to *8.3.3 Collision detection*.

8.3 Description of Functions

This section describes how to use the simulator window and its functions.
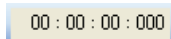
8.3.1 Simulator window layout

(1) Tool bar

(2) Layout Objects
shows the list of robots and layout objects.

(3) Property Grid
shows the settings of the robots and layout objects.

(5) 3D Display
shows the robot orientation and motion from various points of view.

(4) 2D Layout
Places and shows the robot and layout object positions.

(6) Record / Playback
Records and reproduce the simulation product.
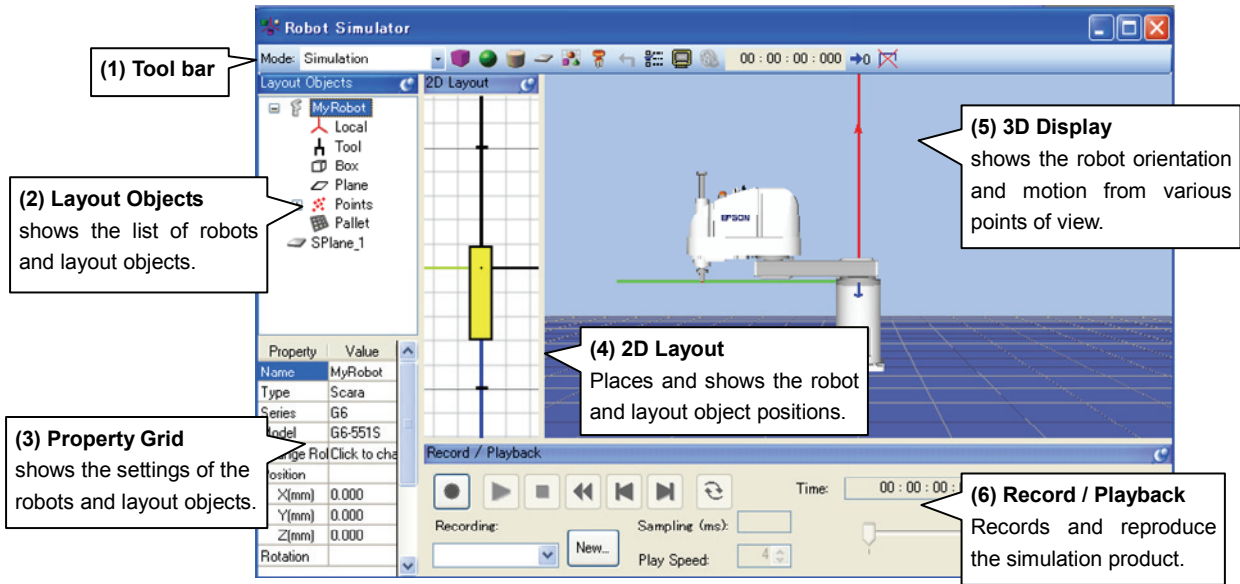
(1) Tool bar

Button	Description
Mode: Simulation	Simulator operating mode. It switches between <Simulation Mode> and <Playback Mode>.
Layout Box	Adds a box object in the 2D layout.
Layout Sphere	Adds a sphere object in the 2D layout.
Layout Cylinder	Adds a cylinder object in the 2D layout.
Layout Plane	Adds a floor / wall object in the 2D layout.
CAD	Adds a CAD object. When you drag & drop this button in 2D layout, a dialog appears to load the CAD data from a file.
Hand	Adds a hand object. When you click this button, the dialog appears to load CAD data from a file.
Reset Collision	Resets the collision detection status. When you click this button while the robot is not interfering with any layout objects, the red display turns to normal.
Simulator Settings	Shows the [Simulator Settings] dialog. In this dialog, the 3D [Render Options] can be configured.
Screenshot	Saves the current 3D display as an image file. A dialog appears to specify a file name and format before saving.

Button	Description
Create Movie	Plays a simulation result (log file) in the Playback mode and saves into a movie file. A dialog appears to specify the file and format to save.
`00 : 00 : 00 : 000` Elapsed Time	Shows the program execution time as if you ran the same program with a real controller (RC180/RC90).
	When a program starts, the elapsed time counter counts from 0 and stops when the program finished. It pauses counting when the program is paused and resumes when the program continues execution.
→0 Clear Elapsed Time	Resets the elapsed time.
⊠ Clear end effector path	Clear the end effector path which the robot displays.

(2) Layout Objects Pane

The Layout Objects pane shows the robot objects and layout objects in a tree format.

When you right-click on a layout object, the deletion menu appears.
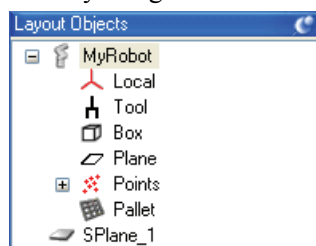
What is an Object?

The objects in the simulator are either a "Robot Object" or a "Layout Object":

A "Robot Object" includes the robot itself, its hand, local coordinates, point information, etc.

A "Layout Object" includes objects to be placed around the robot to simulate the robot peripheral environment in the 3D display.

◆ **Robot Object**

Robot	: The robot itself. The display data is handled by the simulator.
Hand	: The hand is created by loading CAD data (VRML2.0, STEP, or IGES) from a file.
Object to reflect a robot parameter	: Local, Tool, Box, Plane
Object to reflect robot point data	: Point, Pallet

◆ **Layout Objects**

Simple object	: Box, Sphere, Cylinder, Floor / Wall The display data for these objects is handled by the simulator. The object size can be changed as desired by editing properties.
CAD object	: These objects are created by loading CAD data (VRML2.0, STEP, or IGES) from a file.

(3) Property Grid Pane

In the Property Grid pane, you can view and change the settings of the robot objects and layout objects in the Layout Object pane.

◆ Robot Object Properties

Robot

Property	Value
Name	MyRobot
Type	Scara
Series	G6
Model	G6-551S
Change Robot	Click to change
Position	
X(mm)	0.000
Y(mm)	0.000
Z(mm)	0.000
Rotation	
X(degree)	0.00
Y(degree)	0.00
Z(degree)	0.00
Check Collision	True
Check Self Collision	True

Property	Value
Name	Robot name You can specify any name for a robot.
Type	Robot type The robot type (Scara, 6-Axis, and RS) is shown. This property is read-only.
Series	Robot series The robot series is shown. This property is read-only.
Model	Robot model name The robot model is shown. This property is read-only.
Change Robot	If you want to change the robot model, click on ⊡ button. When you click on this button, a dialog appears to change the robot. For details, see *Changing the robot model* described later in this chapter.
Position	Robot position Specifies the robot's base center in the simulator World coordinates.
Rotation	Robot angle
Check Collision	Enables / disables the collision detection for layout objects. Enable : True (default) Disable : False Even if this is enabled, it does not detect collision between the robot base and layout objects.

Check Self Collision   Enables / disables the collision detection for a robot itself.

Enable   : True (default)
Disable : False

Changing the robot model

When you want to change the displayed robot model, click on the ⬚ button in the Change Robot property and the dialog shown below will be displayed. If you cannot see the ⬚ button, increase the property grid width and click once on the [Value] column of the grid.
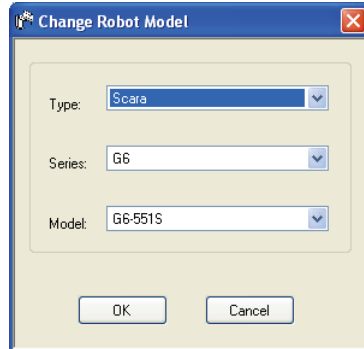
Select a new robot type, series, mode and click on the <OK> button. After the message "Restarting Controller", the new robot will be displayed.

NOTE
☞   When you change the displayed robot model, all the settings for the robot (Local coordinates, Tool coordinates, etc.) will be initialized to the default values.

Type and Series

Type	Series
Scara	G series, LS series
6-Axis	C3 series, S5 series
RS	RS series

Local / Tool / Box / Plane / Pallet

If the local coordinate system of the corresponding number is not defined yet, the check box is grayed.

Property	Description
Visible	Displays a base / local coordinate system Visible : Check Not visible : Uncheck (default)

TIP
☞   For Local 0 (Base), Visible is the default.

Points

Display the point display setting status in the point file. Switch to show/non-show all points.

Property	Description
File Name	Show a point file name.
Visible	Shows / Not show all points Visible : Check Not visible : Uncheck If it is set to show some points, the check box shows indeterminate state.

Point

If the point of the corresponding number is not defined yet, the check box is grayed.

Property	Description
Name	Shows a point label In the Property Grid, point labels cannot be configured or edited.
Visible	Shows / Not show a point Visible : Check Not visible : Uncheck (default)

TIP
☞ If you cannot see the [Visible] column, increase the property grid display width.

Hand

When a hand is registered with a robot, "Hand" is added in the Layout Objects tree.

Property	Value
Name	Hand
Position	
X(mm)	0.000
Y(mm)	0.000
Z(mm)	0.000
Rotation	
X(degree)	0.00
Y(degree)	0.00
Z(degree)	0.00
Filename	BarrettHand.wrl
Visible	True
Check Collision	True

Property	Value
Name	Hand name You can specify any name for a hand. (Default: Hand)
Position	Mounting offset from the robot end effector position.
Rotation	Hand mounting direction
File name	CAD data file name of the hand It cannot be changed.
Visible	Visible : True (default) Not visible : False
Check Collision	Detect collision : True (default) Not detect collision : False

◆ Layout Object
Layout Box / Layout Sphere / Layout Cylinder / Layout Plane / CAD

There are common attributes for all objects and others are for particular objects.

Property	Object	Description
Name	All	You can specify any name.
Type	Plane	Floor : Horizontal (default) Wall : Vertical
Position	All	Specifies a center point in simulator World coordinates. Layout Cylinder: Bottom surface center
Half size	Box	Specifies a length from the center. The box length is double this length.
Radius	Sphere	Sphere radius
	Cylinder	Cylinder radius

Height	Cylinder	Cylinder height
	Plane	Floor length / Wall height
Width	Plane	Floor width / Wall width
Rotation	All	Object angle (Z-axis centering)
File name	CAD	CAD data file name. It cannot be changed.
Color	Box	Display color
	Sphere	Click the drop-down to change display color.
	Cylinder	The display color setting dialog will be displayed.
		Refer to *Change layout object color* for the details.
Visible	All	Visible : True (default) Not visible : False
Check Collision	All	Detect collision : True (default) Not detect collision : False

Change layout object color

When you want to change layout object color, click on the drop-down ⬇ in the Color property and the dialog shown below will be displayed. If you cannot see the drop-down ⬇, increase the property grid width.

Click color you want to display. Layout object color will be changed.

If you do not want to change color, click anywhere other than the display color setting window. The window will be closed.

If you create a custom color, right-click any color in the bottom two rows (16 colors) in the [Custom] tab, and the color setting dialog will be displayed.
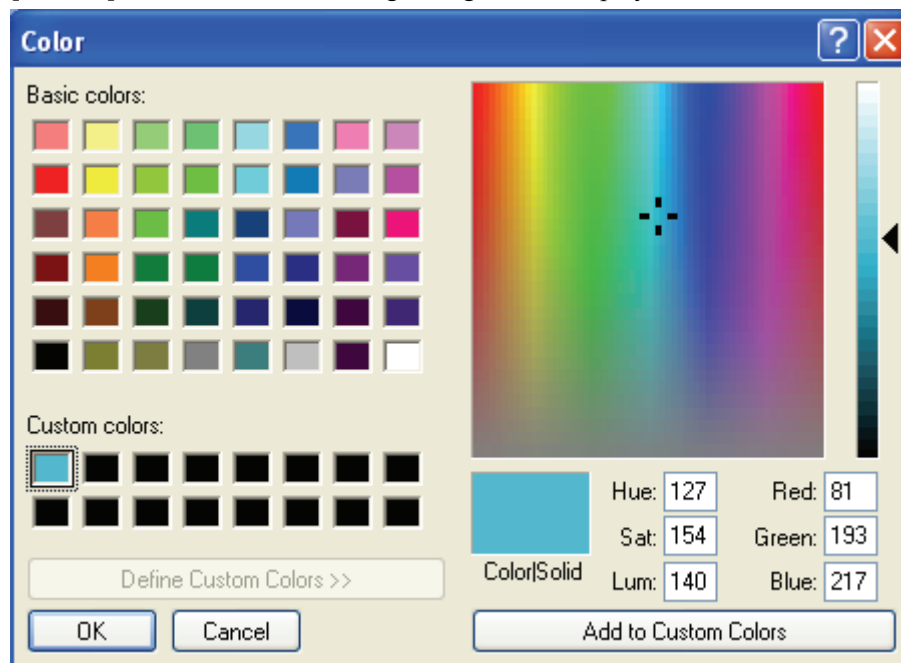
Create a custom color and click the <OK> button.

Created color will be displayed in the display color setting window.

(4) 2D Layout

In the [2D Layout] panel, you can add layout objects, or modify and check the robot objects and layout objects positions.
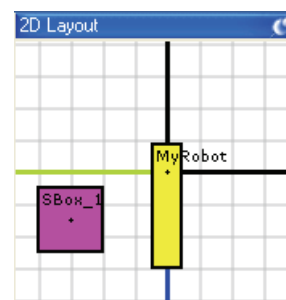
Adding a layout object

Drag and drop a tool button (Box, Sphere, Cylinder, etc.) onto the [2D layout] panel.

When you add a layout object while a robot object is selected in the [Layout Objects], the added object is a separate object.
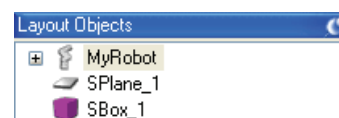
When you add a layout object while a layout object is selected in the [Layout Objects], the added object will be in a group with the selected objects.

Grouped objects move together when the parent object moves.

RightTable/CenterTable/LeftTable of the sample virtual controller "G6 Sample" is an example of grouping.
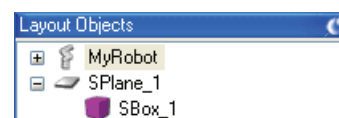
Change robot and layout object position

Drag an object (robot, box, etc.) to change its position.

To move an object in Z direction, use the <D> and <U> keys.

Drag ○ to change the size of an object, and drag ● to rotate an object.

If you are moving a box, it is shown as in the figure on the right:
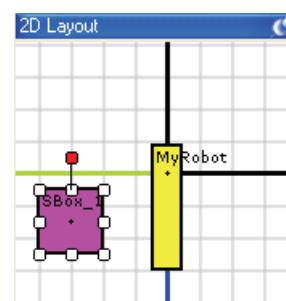
Zoom the layout

To zoom the 2D layout, use the mouse wheel to scroll.

Move the display area

To move the 2D layout display area, drag the 2D layout while simultaneously pressing the <Shift> key.

Rotate the display

To rotate the 2D layout display area, right-click on the 2D layout and use the options - [Rotate Clockwise] [Rotate CounterClockwise].

(5) 3D Display

In the 3D display, you can check the robot orientation and motion from various points of view.
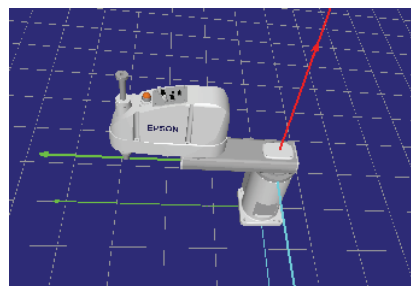
Change the robot / layout object position

To display a guide which indicates the directions to move, press the <Shift> key and click on the object (box, etc.).

To move the object, drag the grid corresponding to the axis.
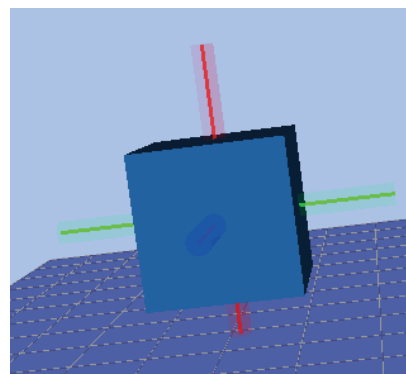
Rotate the robot / layout object

To rotate a layout object around the z-axis, press the <Ctrl> key and move the object (box, etc.).

Move the robot joint

To move a robot joint, press the <Ctrl> key and drag the robot joint. The selected joint is displayed in blue.

If a robot moves to a point out of the motion range, the joint comes back to the previous point.
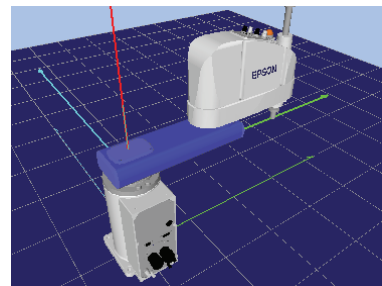
Change the view point

To rotate the view point, press the mouse left button and drag the 3D display.

To move the view point up and down, press the mouse right and left button and drag the 3D display.

Also, you can use the <L>, <R>, <D>, and <U> key to move the view point.

You can reset the view point from the menu opened by right click.

Zoom the layout

To zoom the 3D display, use the mouse wheel to scroll.

You can change the zoom level from the menu opened by right clicking with the mouse.

Check for collisions

When a collision between a robot and layout object is detected, the collided robot joint and layout object are displayed in red.

For details, refer to *8.3.3 Collision detection*.
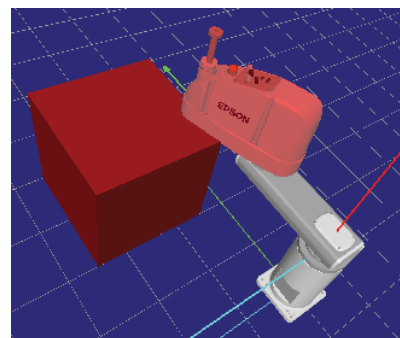
(6) Record / Playback

In Playback mode, you can record and produce simulation results.

Also, you can store the simulation results in movie files.

Function	Description
● RECORD	When the button is red ●, it saves the simulation result into the specified log file. Every time you execute the program, the log file is overwritten with the new information.
	When the button is gray ●, it doesn't save the simulation result. As the default, it doesn't save the simulation result.
▶ PLAY	Plays a simulation result of a specified log file.
■ STOP	Stops the simulation playback.
◀◀ REWIND	Puts the playback step to the starting point.
◀ BACK	Goes back one step. The number of steps back is specified in [Play Speed].
▶ NEXT	Goes to next step. The number of steps to go is specified in [Play Speed].
↻ REPEAT	When this button is pressed, repeats the simulation playback.
Log list	Specifies a recording file to record and play.
New button	Create a new log file.
Sampling	Displays the log file sampling interval.
Play Speed	Specifies the playback interval with a number of steps.
Play position	Display the current playing position.

8.3.2 Simulator Settings

When you click the Tool bar-<Simulator Settings ▦ > button, the [Simulator Settings] dialog appears.

These settings are valid while EPSON RC+ 5.0 is running. However, when EPSON RC+ 5.0 is restarted, they return to the default settings.
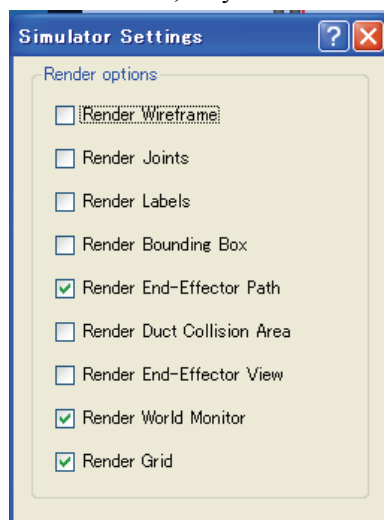
Render options

In the [Render options] dialog above, you can specify the 3D display method.

The specified display method is valid while EPSON RC+ 5.0 is running. However, when EPSON RC+ 5.0 is restarted, it returns to the default display settings.

Function	Description
Render Wireframe	Changes to the wire frame display (a three dimensional graphic using lines and points).
Render Joints	Displays fulcrum points of robot joints.
Render Labels	Displays names of robots and layout objects.
Render Bounding Box	Displays robots and layout objects in bounding box.
Render End-Effector Path	Displays robot end effectors path of a certain period of time.
Render Duct Collision Area	Available with G1 series and LS series manipulators. Shows the collision detection range of the robot ducts with the bounding view.
Render End-Effector View	Displays the view from robot end effectors.
Render World Monitor	Displays the simulator World coordinates.
Render Grid	Displays the simulator grid.

Produce the robot motion by outputting to a recording file

(1)   Confirm that the mode is "Simulation" mode on the Simulator Tool bar.

Mode: Simulation

(2)   Click on the <New> button in the [Record / Playback] window.   The [New Recording] dialog appears.

(3)   Here, enter "TestPlayBack" and click the <OK> button.
Now, you can see "TestPlayBack" in the Recording list.

(4)   Click on the <RECORD> button in the [Record / Playback] window, which enables recording. Now, the <RECORD> button is in red ●.

(5)   Start a program from the [Run] window to move the robot. The simulation result is saved in the recording file while executing the program.

(6)   Change the simulator operating mode to "Playback Mode" again.

(7)   Click on the <PLAY> button and the simulation result starts to play.

Produce the robot motion with saving in a movie file at once

(1)   Confirm that the mode is set to "Playback" on the Simulator Tool bar.

(2)   Click the <Create Movie> button in the Simulator Tool bar.

(3)   When the [Create Movie] dialog appears, select "TestPlayBack" from the Log list.
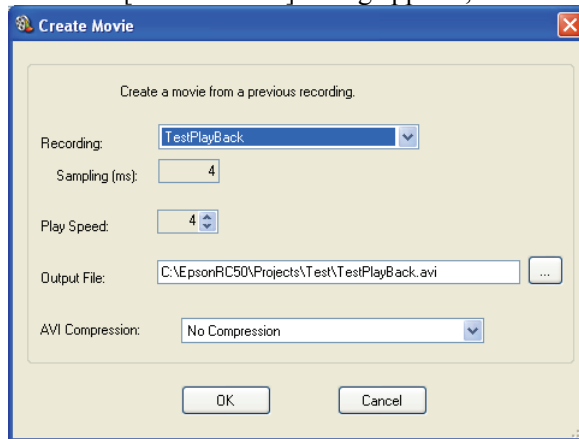
(4)   Specify the [Output File] and [AVI Compression] if necessary.

(5)   Click on the <OK> button.
The [Create Movie] status window appears and the specified movie file is created with playing the log file.

(6)   The created movie file is "TestPlayBack.avi" in EPSON RC+ 5.0 project folder (\EpsonRC50\projects\"a project name").

8.3.3 Collision detection

In the simulation, collisions can be detected between the robots including its hand and the layout objects. Here we describe the settings and details of collision detection.

Basic settings for collision detection

In the [Property Grid] of the robot, the following can be configured.

Property	Value
Check Collision	Enables / disables the collision detection for layout objects.
	Enable: True (default)
	Disable: False
	Even if this is enabled, it does not detect collision between the robot base and layout objects.
Check Self Collision	Enables / disables the collision detection for a robot itself.
	Enable: True (default)
	Disable: False

Target of collision detection

In the Property Grid of the layout objects, the following can be configured.

Property	Value
Check Collision	Enables / disables the collision detection for a robot.) Enable: True (default)
	Disable: False
	Even if this is enabled, it does not detect collision when the robot collision detection is disabled.

Generate error when collision is detected

When you open the [Setup]-[Controller]-[Simulator] and check the [Generate error when collision detected] checkbox, if a collision is detected during SPEL$^+$ program execution, an error occurs in the controller and the program stops.

After checking the check box, click the <Apply> button and then click the <Close> button.
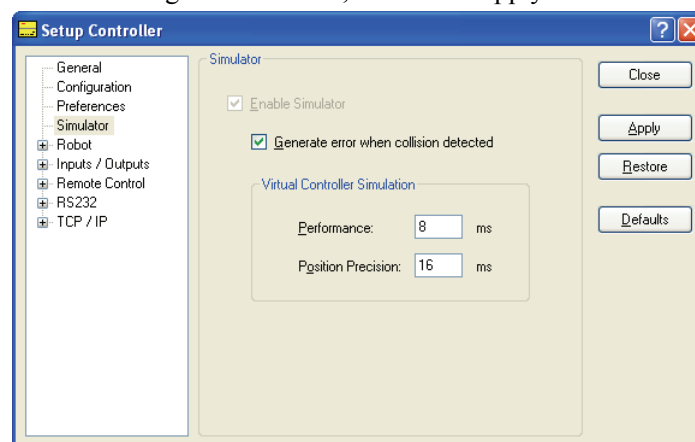
NOTE
☞   The purpose of this function is to find where the program has a problem and not to prevent the collision of robots.

It cannot guarantee that it has enough time for robots to stop when the simulator detects the collision.

Caution about the collision detection of Floor / Wall

A collision is detected when a floor or wall is in contact with the robot. If the robot or plane positions are changed so that the robot passes completely through the plane, then no collision is detected.

Accuracy of collision detection

The collision detection in the simulator cannot guarantee accuracy. Make sure to have a margin when you apply the simulation result for a real robot system.

The restrictions of the simulator are described in *8.4 Specification and Restriction of Simulator*.

8.3.4 Virtual controller

To execute programs in the simulator, you need to create a virtual controller with defined robot and layout.

Robot settings and layout settings for the 3D display are saved for each virtual controller. If you want to transfer the robot or layout data, you can copy and transfer the data.

Create a new virtual controller

Refer to *8.2.2 Working with a user created system*.

Copy the sample or configured virtual controller

(1) Click the EPSON RC+ 5.0 Tool bar-<Connection ⌷ > button.
    The [PC to Controller Communications] dialog appears.
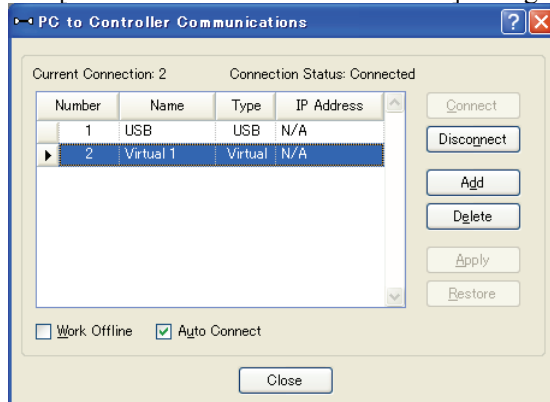
(2) Click the <Add> button. The [New Controller Connection] dialog appears.

(3) Select the <Connection to new virtual controller> option button and specify a virtual controller from the list box. Click the <OK> button.

(4) New "Virtual 2" is created. Click the <Apply> button.
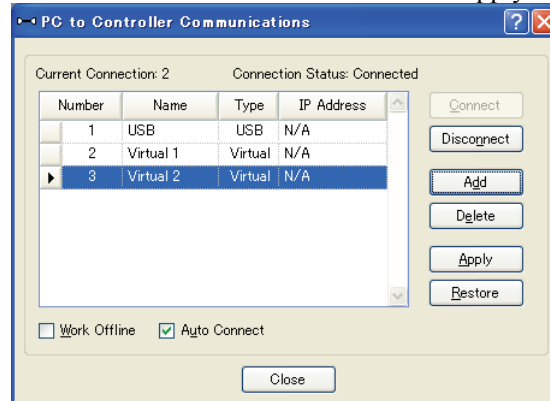
(5) Close the dialog and go back to the EPSON RC+ 5.0 main window.

(6) Connect to "Virtual 2" and display the simulator window.
      The robot setting and layout setting of 3D display has been taken over from "Virtual 1".

(7) When you want to change the robot type, use the [Change Robot] in the robot object property.
      For details, refer to *8.3.1 Simulator window structure – (3) Property Grid*.

Virtual controller configuration

Normally you don't need to configure a virtual controller.

The configuration is available from the [Setup]-[Controller]-[Simulator] page.
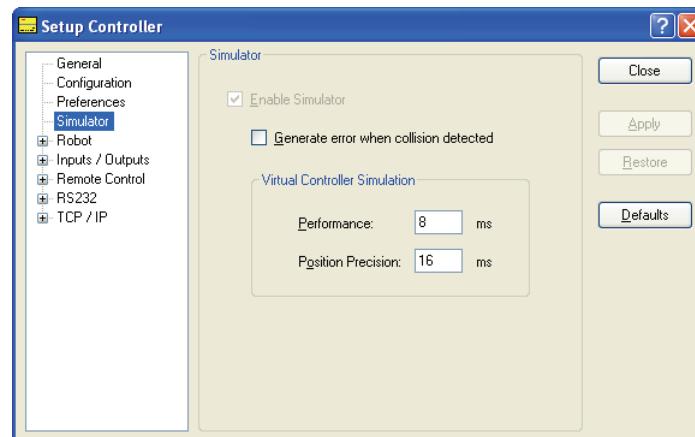
[Performance]   : You normally don't need to change the setting from 8 ms (default).

[Position Precision] : You normally don't need to change the setting from 16 ms (default).

The situations when you need to change these settings are described in *8.4 Specification and Restriction of Simulator*.

8.3.5 Connection with controller

When the firmware of the RC170/RC180 controller is Ver.1.10.*.* or later, you can enable the simulator function with the controller connected.

For the RC90 controllers, regardless of the firmware version, you can enable the simulator function with the controller connected.

Enable the Simulator in controller

From the [Setup]-[Controller]-[Simulator], check the [Enable Simulator] check box to enable the simulator function.

After checking the check box, click the <Apply> button and then click the <Close> button.
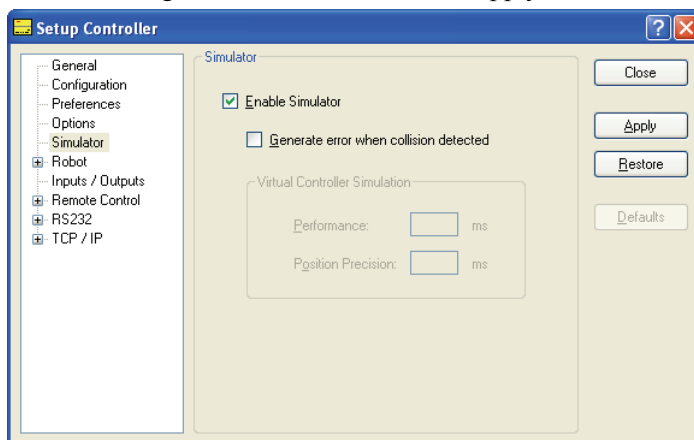
If collision with the simulator object is detected during a Jog motion or a robot motion command execution when the Simulator is enabled, the Manipulator stops operation and a Warning occurs.

To avoid collision with peripherals by using the Simulator, set 15 mm or greater margins to the simulator object.

Function restrictions when connected with controller

- Pallets cannot be displayed.
- You cannot change the robot from the Simulator window.
- You cannot select and move the robot arms in the Simulator window, except during the controller Dry run.
- When the robot configured in the controller is not supported in the Simulator, a warning message is displayed when the Simulator window is opened, and the object list and the robot in 3D window are not displayed.
- The [Record/Playback] functions are not available.

Operator Window

When you enable the Simulator, the <Simulator> button is added to the Operator Window. When you click on the <Simulator> button, the 3D display window appears.
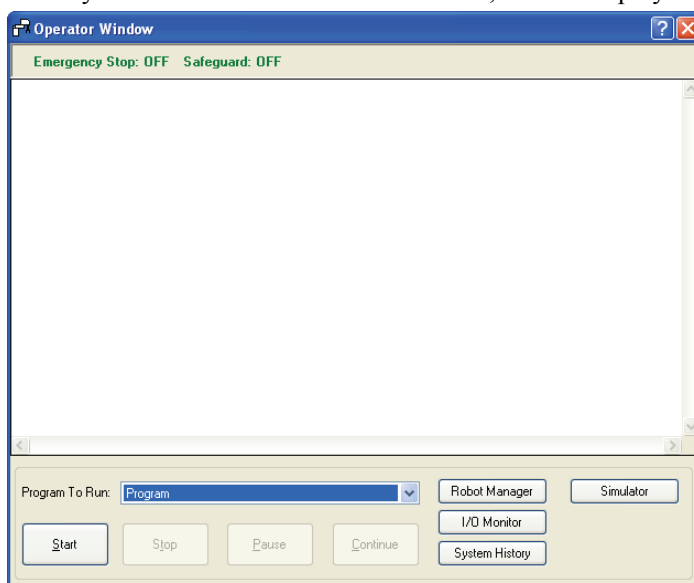
8.4 Simulator Specifications and Restrictions

This section describes the simulator specification, its restrictions, and precautionary statements.

8.4.1 EPSON RC+ 5.0 package

EPSON RC+ 5.0 has two packages:

- EPSON RC+ 5.0   : Standard package for developing the robot system
- EPSON RC+ 5.0 Trial : Trial package for limited use (program execution on PC)
                                           * It cannot connect with an RC180 robot controller.

	Program execution on PC	Connection with controller	Connection with controller + 3D display
EPSON RC+ 5.0	OK	OK	OK *1
EPSON RC+ 5.0 Trial	× *2	×	×

*1   Requires the configuration to enable the simulator functions in EPSON RC+ 5.0. Refer to *8.3.5 Connection with RC170/180 controller* for the detail information.

*2   Continuous execution time of program is limited.

8.4.2 Specifications and precautions for the 3D display

Available robots for 3D display

In the future, we will add more robots for 3D display. Call your EPSON Regional Sales Manager for the latest information.

NOTE
☞

The duct of the G1 series and LS series is displayed roughly.

・   Check the dimensions in the manipulator manual.

・   The duct actually vibrates while the manipulator is moving, the simulator doesn't display the vibration. Check how the duct vibrates with your real manipulator.

The bellow for Cleanroom or Protection model is displayed roughly

・   Check the dimensions in the manipulator manual.

Available CAD data for 3D display

The following format is available for 3D display to show the robot hand and CAD object.

- VRML 2.0

   Limits of reading: VRML2.0 prototype is not supported.

- STEP (AP203/AP214)

   Limits of reading:

   If Color is configured in Face, the specified Color is displayed.

- IGES

NOTE
☞

The data file must be saved in the specified folder on the PC and not saved in EPSON RC+.

CAD data setup orientation

Some CAD data coordinates may be different from those of the simulator.

Adjust the coordinates to the correct position by changing [Property]-[Rotation] after loading the CAD data.

When loading CAD data as a hand, set the origin of the CAD data in the Tool0 position of the manipulator. Set the coordinates to the correct position by changing [Property]-[Position] after loading the CAD data.

Number of available layout objects

You can create as many layout objects as you want.

However, when there are many objects to display, the display update interval becomes longer and the judgment of collision detection becomes rough. Especially for CAD data, displaying data that is too complicated is not recommended.

8.4.3 Specifications and precautions for Simulation (program execution on PC)

Overview

The Simulator produces the robot motions virtually on your PC.

It is designed to make the performance gap between the real system and the virtual system as small as possible. However, a few differences in the virtual system are inevitable. The operation time prediction and collision detection do not guarantee the precision.

Fully understand the contents in this chapter and check if the real system operates without any problems before you go to full-scale operation.

Operation time prediction

Operation time displayed in the Simulator window is approximate time required for executing the program.

Time for the motion commands such as Go, Jump reflects the Speed and Accel values in the program. The operation time may vary when you operate the real robots from the displayed operation time according to conditions such as the Fine setting and servo delay.

In particular, when small ranges are used with the Fine instruction, the real robots need a longer operation time for accurate positioning.

The simulation cannot guarantee the precision but the margin of error in the operation time is within 10% when you execute motions with the standard cycle time (with the default Fine settings).

Considered in the operation time prediction	Not considered in the operation time prediction

Robot model
Speed settings (Speed, Speeds, etc.)
Acceleration settings (Accel, Accels, etc.)
Load (Weight, Inertia)
Others (ARCH, CP)
	Fine setting
	  Error within 10% from the default
	  (Motions of standard cycle time)
	  With larger setting than the default,
	  the operation time will be shorter.
	  With smaller setting than the default,
	  the operation time will be shorter.
	Servo delay
	  With the real robots, the operation will be longer.

Time for the other commands than the motion commands is a virtually executed time on PC; therefore the actual time varies widely depending on the PC performance.

When measuring the motion time between two points, as simple program as possible is recommended. Refer to *8.2.2 Working with a user created system - 8. Measure the robot operation time*.

Collision Detection Precision

The Simulator Collision Detection provides an indication whether robots collide with the peripheral equipment or not when the program is executed. It does not consider the error in trajectory due to servo delay. Be aware that a margin is necessary for the real robot system.

The Simulator judges collisions more accurately when the robot motion speed is slow.

The judgment of collision detection during program execution is accomplished with the 3D display update. When your PC has high graphics performance, the collision judgment becomes more accurate.

In Playback mode, the Simulator judges collisions in all steps and is useful when you need accurate detection.

The Simulator cannot guarantee the precision but the margin of error in the collision detection is within 10 mm when you execute motions with Speed 100% on a PC of the recommended specifications.

Motion duty and Overload error

In the Simulator, you cannot detect the overload error. Even when the motion duty is too high and the robot should have the overload error and stop, it keeps moving.

Duty 50 % - As a measure of possible duty, the robot can really keeps moving at duty 50% with the maximum acceleration/deceleration speed and without the overload error. However, it depends on the robot model type, load, points to go to, and acceleration/deceleration speed setting, etc.

Time progress difference by PC condition

On a PC that meets the system condition, progress of the time in the Simulator and the real time (as you see on the watch) are almost the same (few percentages difference).

If you are running other applications such as Windows Media Player simultaneously, the time progress in the Simulator can widely vary from the real time. In this case, use the Simulator function while other applications are not running.

Also, on some PC models, the time progress in the Simulator may widely vary from the real time. In this case, set the [Performance] at 16 ms and the [Position Precision] at 20 ms, which may close the gap of the time progress.

Time confirmation program
(If the two printed times are within 27 to 33 seconds, there is no problem.)

```
Function main
    Print Time$
    Wait 30
    Print Time$
Fend
```

Execution on PC below the minimum of specification

You can install the EPSON RC+ and use the Simulator functions on a PC that doesn't meet the minimum of the specification.

However, it doesn't guarantee the correct motions because the following may happen:

- Operation time prediction is not accurate
- Collision Detection has a large margin of error
- 3D display skips updates

<div style="background:#ccc">8.4.4 Specification and precaution of EPSON RC+</div>

Restriction on the controller settings

When you connect with a virtual controller, the following items are grayed and become unavailable to change.

- Setup: Controller: Configuration Page: IP Address, etc.
- Setup: Controller: Preference Page: Dry run, etc.

Backup and restore of the controller setting

The setting data that you backup in the virtual controller cannot be restored in a controller. Also, the setting data that you backup in a controller cannot be restored in a virtual controller.

<div style="background:#ccc">8.4.5 Restriction on SPEL⁺ command execution</div>

(1) I/O operation and commands (On, Off, SW, Ctr, etc.)

All I/O including the option boards are available in a virtual controller. Operating I/O data is stored in the PC memory (virtual I/O mode). The I/O input status can be changed from the EPSON RC+ I/O Monitor window. Also, the I/O input status can be changed using the SetSw or SetIn statements in a SPEL$^+$ program.

NOTE Even if you specify an asynchronous On/Off command, the I/O status cannot be changed after the specified time and the Ctr function always returns 0.

(2) Pallet definition (Pallet statement)

In the Simulator, you can use the Pallet statement as in the controller.
However, once you define a pallet in a project, you cannot clear the pallet definition unless you use the same project.
Even if you switch to another project, the pallet definition will be redefined when you come back to the original project.

	Controller	Simulator (Virtual controller)
Timing of a pallet definition	When executing a Pallet statement in a program, etc.	When executing a Pallet statement in a program, etc.
Timing of a pallet reset	- When building a project - When turning OFF the controller - When switching to other project	None

(3) Ethernet / RS-232C communication command
(Print #, Input #, OpenCom, OpenNet, etc.)

All ports including the option RS-232C board are available in a virtual controller. However, an Ethernet port requires configuration of the IP address and TCP/IP port. Ethernet / RS-232C communication commands do not perform actual communication.

Output data from Print #, etc. is saved in the communication output file. In the input by Input#, etc, the return value is 0 (numeric data) or blank (string).   However, if you create a communication response file, the return value depends on the file content.

Communication output file

When calling OpenCom or OpenNet command, a communication output file is created in the \EpsonRC\Virtual\Mounted Volume\Project folder on the PC.

> DummySend***.dat   : Communication output file (*** is the port number)

When a communication output file already exists, the previous output data is deleted. The file is deleted when you switch the project; save the file in a proper folder if you need.

When executing the following program,

```
OpenCom #1
Print #1, 123
Print #1, "TEST DATA"
CloseCom #1
```

the DummySend001.dat file will contain...

```
123
TEST DATA
```

Communication response file

Copy the communication response file to the \EpsonRC\Virtual\Mounted Volume\Project folder in before running a program. The file is deleted when you change the project; save the file in another folder if you need to keep it.

When calling OpenCom or OpenNet command, the communication response file is loaded.

> DummyRead***.dat   : Communication response file (*** is a port number)

When the following DummyRead001.dat file is used,

```
321
Test Data
```

and the following program is executed,

```
Integer i
String s$
OpenCom #1
Input #1, i
Input #1, s$
CloseCom #1
Print i
Print s$
```

the return values are i = 321 (numeric data), and s$ = "Test Data" (string).

(4) Vision command (VRun, VGet, etc.)

For Vision-related commands, the communication with Smart Camera (SC300 / SC1200) is not performed and the command quickly executes.

If you execute VGet, the return value is 0 (numeric data) or blank (string).

(5) Other restrictions

For the Wait command, the following syntax is not supported:

Wait InsideBox()

Wait InsidePlane()

For the Time and Date commands, the time can be displayed, but the time setting is not available.

For the Box command, it always executes the default operation (I/O output On when the Box approaches) even if you specify the remote output logic.

8.4.6 Specification and precaution of EPSON RC+ 5.0 Trial

Version upgrade from EPSON RC+ 5.0 Trial to EPSON RC+ 5.0

Follow the procedures in *Appendix A: Software Installation* to upgrade to EPSON RC+ 5.0. The EPSON RC+ 5.0 Trial version doesn't need to be uninstalled.

NOTE
☞

You can continue to use the projects and virtual controllers (layout) that you used in the EPSON RC+ 5.0 Trial in the EPSON RC+ 5.0 Standard version.

Program execution time in EPSON RC+ Trial version

In the Trial version, programs will execute continuously for up to one hour.

If continuous execution is over one hour, the following warning appears.
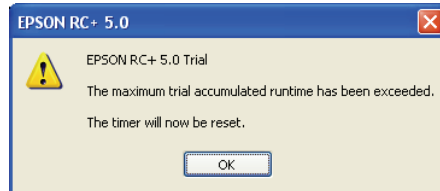
You can execute the program again after this warning is displayed, and the continuous execution timer will be reset.

9. Robot Configuration

This chapter contains information for configuring robots installed on the controller.

Robots are configured from the Robots folder on the Setup | Controller dialog tree.

9.1 Setting the Robot Model

1. From the Setup Menu, select Controller.

2. Click on the Robot folder in the tree on the left side to open the Robot Model page.
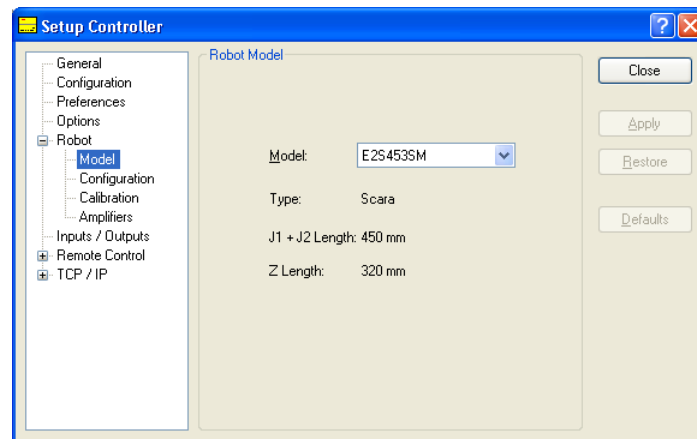
3. Select the model for your robot from the Model dropdown list. Only robot models that can be used with the amplifiers currently in the controller are listed. When using Dry Run, all robot models are shown in the list.

4. Click the **Apply** button to save settings.

5. Click on Robot Configuration in the tree to open the Robot Configuration page.
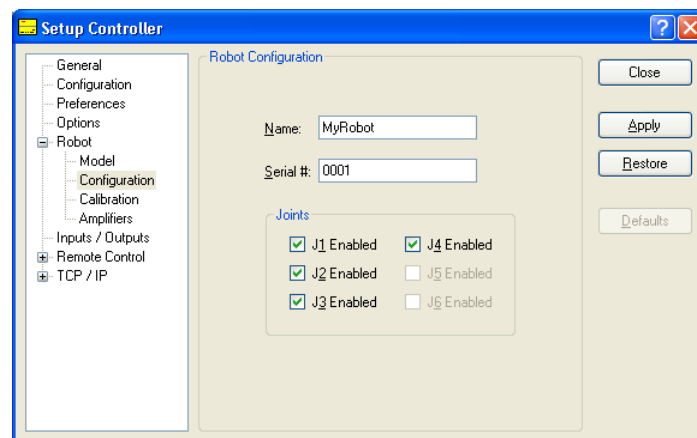
6. Type in a name for the new robot and enter the serial number from the robot's nameplate. Any serial number can be used, but it is recommended that you use the number that is stamped on the robot.

7. Click **Apply** to save settings.

9.2 Calibrating the Robot

EPSON RC+ 5.0 provides a calibration wizard for calculating the robot calibration parameters Hofs and Calpls.

To calibrate the robot:

1. From the Setup Menu, select Controller.

2. Click on the Robot folder in the tree on the left side to open the Robot Calibration page.
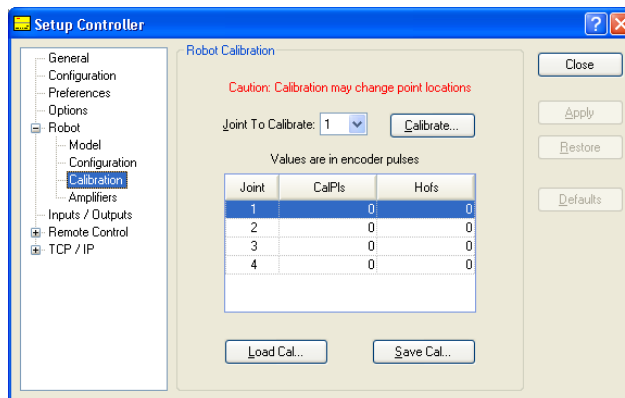
3. Select the joint to calibrate.

4. Click the **Calibrate** button to start the calibration wizard. The instructions for each step are shown on the right side of the dialog.
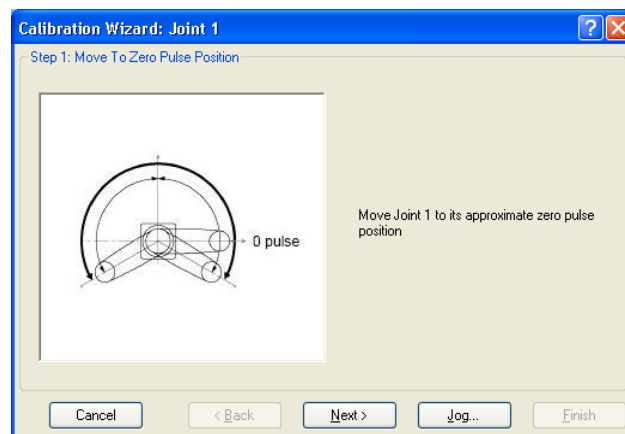
5. Click the **Jog** button to jog the robot to the approximate zero pulse position, as shown in the picture.

6. Click the **Next** button to view the next step.
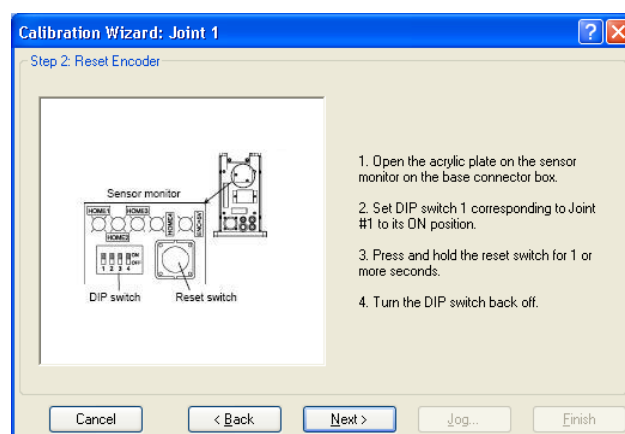
7. For SCARA robots, reset the encoder for the selected joint as described on the dialog. For 6-axis robots, the encoder for the selected joint is automatically reset.

8. Click the **Next** button to view the next step. The controller is restarted before step 3 is displayed.
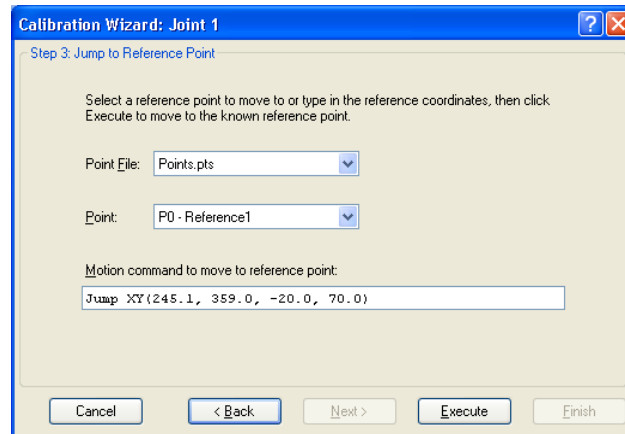
9. Select a point in the current project. By default, a Jump command will be inserted into the **Motion command to move to reference point** text box. You can change this command as necessary to move to the known reference position.

10. Click the **Execute** button to execute the motion command to move to the reference point. Note that the robot will only be close to the reference point, which will be corrected in the next step.

11. Click the **Next** button to view the next step.
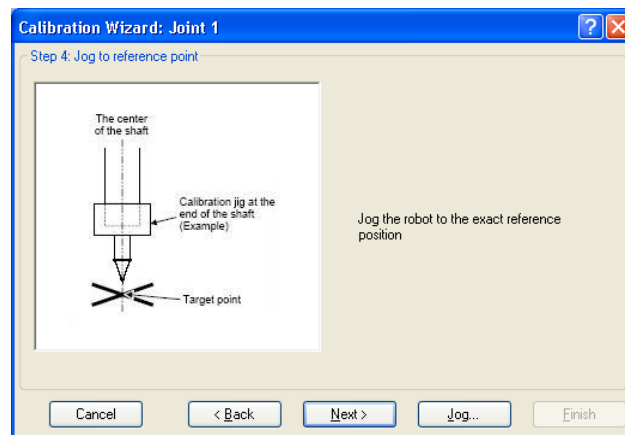
12. Click the **Jog** button to carefully jog the robot to the exact reference point position.

13. Click the **Next** button to view the next step.
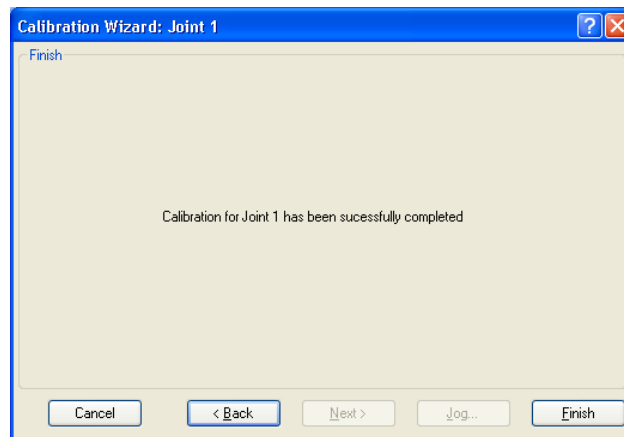
14. Click the **Finish** button to save the calibration settings for the selected joint.

9.3 Changing Robot System Parameters

The following system parameters for the robot can be changed from EPSON RC+ 5.0:

- **Enable/Disable Joints**

  You can disable one or more joints from Setup | Controller | Robot | Configuration. On robots with ball screw Z axis, you must disable both joints 3 and 4 together.

- **Hofs**

  Hofs are the joint home offsets. You can view and edit the values from Setup | Controller | Robot | Hofs. However, it is recommended that you use the Robot Calibration wizard to set these values. These values are unique for each robot and are supplied from the factory. Hofs are especially important for SCARA robots because the values determine that both lefty and righty hand orientation will position the robot at the same point.

- **CalPls**

  CalPls values are joint calibration offsets. You can view and edit the values from Setup | Controller | Robot | Calpls. However, it is recommended that you use the Robot Calibration wizard to set these values. These values are unique for each robot and are supplied from the factory. CalPls values are used to calibrate joint position after replacing a motor / encoder.

These are one-time settings for each robot. Additional robot parameters can be set from the Robot Manager.

To change robot parameters, follow these steps

1. Select Controller from the Setup Menu.

2. Under the Robot folder in the tree on the left, select Calibration, Hofs, or CalPls.

3. Perform a calibration or change values for Hofs or CalPls.

4. Click **Apply** to make the changes permanent.

9.3.1 Saving robot calibration data

You can save and load individual robot calibration files. This is useful for moving a robot from one controller to another. When you save calibration data, a file is created with an MPD file extension. This file contains Hofs and CalPls values.

To save robot calibration data

1. Select Controller from the Setup Menu.

2. Under the Robot folder in the tree on the left, select Calibration.

3. Ensure that the robot serial number is correct. The serial number will be used to create the default file name. It is recommended that the serial number be used.

4. Click the **Save Cal** button. Browse to a destination directory and click Save.

9.3.2 Loading robot calibration data

To load robot calibration data

1. Select Controller from the Setup Menu.

2. Under the Robot folder in the tree on the left, select Calibration.

3. Click the **Load Cal** button. Browse to the desired MPD file and click Open.

10. Inputs and Outputs

10.1 Overview

The RC90/RC180 controller I/O has the following types of I/O:

Standard I/O   This digital I/O comes standard with the controller.

Expansion I/O   This is optional digital I/O that can be added to the controller to expand standard I/O.
For the RC180 controllers, up to four boards can be added, each with 32 inputs and 32 outputs.
For the RC90 controllers, up to two boards can be added, each with 24 inputs and 16 outputs.

Fieldbus I/O   An optional board can be added that supports fieldbus slave mode for DeviceNet, EtherNetI/P, PROFIBUS-DP, PROFINET, or CC-Link.
EtherNet/IP and PROFINET are not supported by the RC90 controller.

Memory I/O   This is built-in memory bits that can be used for inter-task communications.

For Standard, Expansion, and Fieldbus I/O, there are input bits numbered starting with 0, and output bits numbered starting with 0. For example, there is an input bit 3 and a separate output bit 3.

For memory I/O, each memory bit is both an input and an output.

For specifications and instructions on wiring I/O, see the controller manual.

10.2 I/O Commands

The SPEL+ language has several commands for inputs and outputs listed below. For details on each command, see the SPEL+ Language Reference.

Input Commands

In   Reads one byte of input bits.

InBCD   Reads one byte of input bits in Binary Coded Decimal format.

InW   Reads one word of input bits.

Oport   Reads one output bit.

Sw   Reads one input bit.

Output Commands

Off   Turns off one output bit with optional time.

On   Turns on one output bit with optional time.

OpBCD   Sets one byte of output bits in Binary Coded Decimal format.

Out   Sets / reads one byte of output bits.

OutW   Sets / reads one word of output bits.

Memory I/O Commands

MemOff   Turns off one memory bit.

MemOn   Turns on one memory bit.

MemOut   Sets / reads one byte of memory bits.

MemSw   Reads one bit of memory.

10.3 I/O Configuration

To view the current I/O configuration, select Setup | Controller | Inputs and Outputs. This will show you what I/O is installed on the controller.
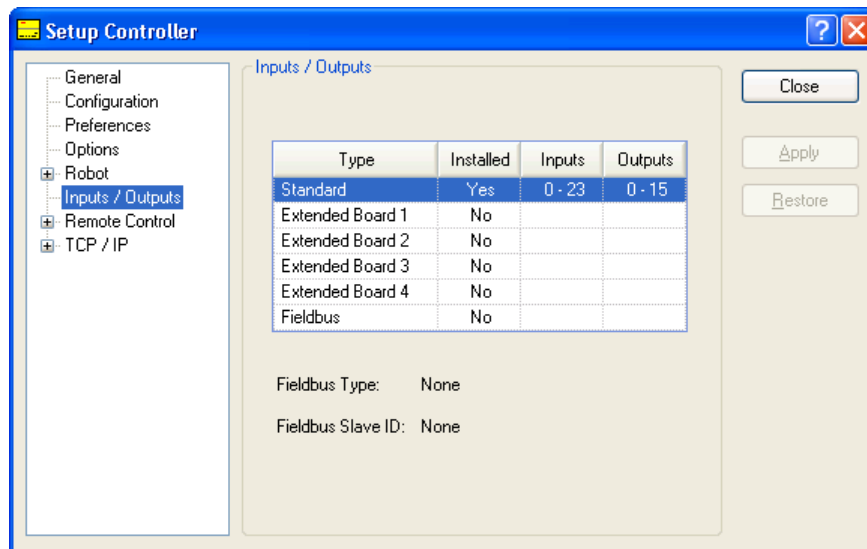
Standard and expansion I/O is automatically configured by the controller. To add expansion I/O boards, see the controller manual.

The optional fieldbus I/O board is automatically detected. You must configure the device ID using a DIP switch on the controller. See the controller manual for details.

10.4 Monitoring I/O

To monitor I/O, use the I/O Monitor tool by selecting Tools | I/O Monitor. From the I/O monitor, you can view inputs and outputs or memory I/O in bit, byte, and word formats. You can also configure up to three custom I/O views.

For details on how to use the I/O Monitor tool, see section I/O Monitor Command in the GUI chapter.

10.5 Virtual I/O

The RC90/RC180 controller supports virtual I/O. When enabled, virtual I/O allows you to simulate your hardwired I/O. You can turn on / off any input bit or output bit. Normally this is used when the controller is in Dry Run mode with no robot or I/O connected.

Virtual I/O Commands

Command	Description
SetIn	Set the value of an 8 bit input port.
SetInW	Set the value of a 16 bit input port.
SetSw	Set the value of one input bit.

11. Remote Control

11.1 Overview

The controller can be controlled by external equipment using Input/Output, Ethernet (TCP/IP), and RS-232C. The external equipment can execute several commands, including Motor On/Off, Start, Pause, Continue, and Stop.

For details on extended function of the remote I/O, refer to *EPSON RC+ 5.0 Remote Control Reference manual*.

For use of Ethernet, refer to the section *11.10 Remote Ethernet*.

For use of RS-232C, refer to the section *11.11 Remote RS232*.

This section to *11.9 Remote Input Handshake timing* describes the Remote control using Input/Output.

There are two steps required for remote control:

1. Configure Remote Control inputs and outputs using the Remote Control tab on the Setup | Controller | Remote Control page.
   Inputs and outputs from 0 to 7 are initially assigned to remote functions. Change the setting if needed.

2. Set the control device to remote on the Setup | Controller | Configuration page.
   To enable external remote inputs, assign the remote functions and also set the control device to remote. When control device setting is remote, the controller is only controllable from the remote device.

Remote control function can be used in the following systems.

Example: Control the robot from a PLC
   Use remote control to control the robot (controller) from a PLC.
   When using a PLC, you will need to be familiar with the handshake required to use remote inputs. See details below.

Example: Control the robot using a push button box with buttons and lights
   The lights are connected to remote control outputs on the controller to indicate status, such as AutoMode, MotorOn, Error, etc. The buttons are connected to remote inputs to control motor power and start programs.

For details of each I/O connection, refer to the following:

Controller manual - Setup & Operation: I/O Connector
I/O Remote Set Up
Expansion I/O Board
Fieldbus I/O Board

11.2 Remote Control Input Output Configuration

This is the procedure to assign the remote functions to the I/O.

1. Select Controller from the Setup Menu and select the **Remote Control Inputs** or **Remote Control Outputs** page**.**

2. For each input or output you want to use for remote control, click on the Input # or Output # cell for the desired signal, then click the arrow and select a bit number.

3. Click **OK** to save the new settings.
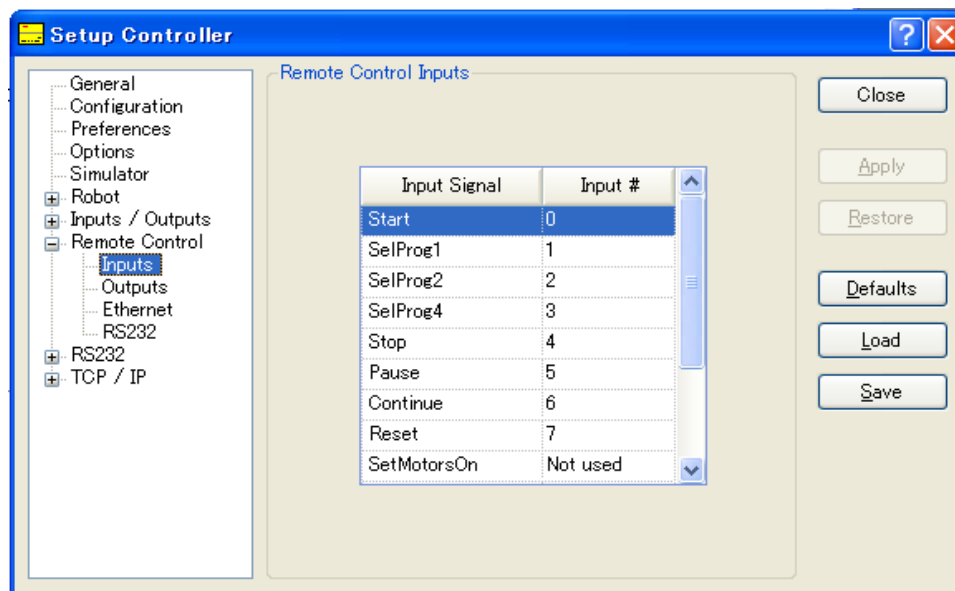
For details of dialog setting, refer to *5.12.2 Controller Command (Setup Menu)*.

11.3 Control Device Configuration

This is the procedure to set the control device to "Remote I/O".

1. Set the controller control device to Remote. Select Controller from the Run Menu, click on Configuration in the tree on the left, then select "Remote I/O" from the Control Device dropdown list.

2. Click **Apply** to save the new setting and the click **Close**.
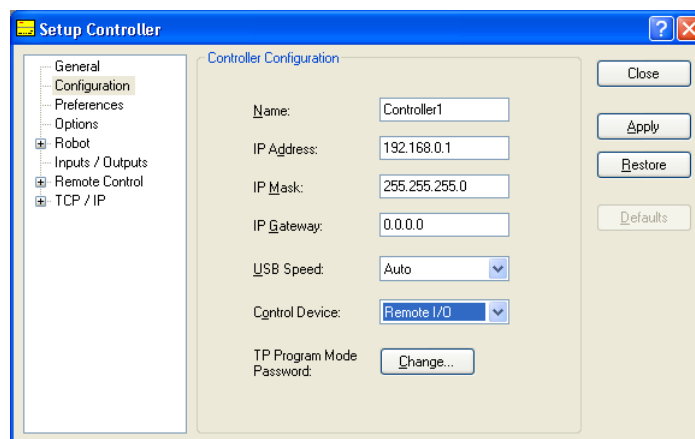
For details of setting, refer to *5.12.2 Controller Command (Setup Menu)*.

11.4 Auto Mode with Remote Control

To run in auto cycle with remote control

1. The host device (e.g. PLC) should wait for the AutoMode or Ready remote output to turn on before issuing remote commands.

2. Now the remote input commands will be accepted.

To monitor remote operation from the EPSON RC+ 5.0 Operator Window

1. Set the EPSON RC+ 5.0 Start Up Mode to **Auto**. See the section *Start Up Mode* in the chapter *Operation* for details.

2. The PC should also be configured to auto log into Windows and start EPSON RC+ 5.0 during Windows start. See the section *Start Up Mode* in the chapter *Operation* for details.

11.5 Teach Mode with Remote Control

When using Teach Mode with remote control, when Teach mode is ON, no remote input commands can be used. Remote status outputs will still operate.

⚠ WARNING	■ Remote status outputs (such as MotorOn, Home, etc.) will operate when Teach Mode is ON, even when the enable switch (dead man) is disengaged. Therefore, DO NOT use remote status outputs to drive any devices that cause motion or any other safety hazard.

You can monitor teach mode status using the TeachMode remote output.

11.6 Debugging Remote Control

You can debug programs using Remote Control from the EPSON RC+ 5.0 development environment.

To run programs by remote control for debugging:

1. Create a program (in the same procedure as usual).

2. Open the Run Window and click Enable Remote I/O.

3. Now the remote commands will be accepted.

You can set breakpoints and print messages to the run window.

NOTE
☞
If you cannot wire the I/O, use virtual I/O mode for debugging. Remote function is also available when virtual I/O is enabled.

11.7 Remote Inputs

Remote inputs are used to control the Manipulators and start programs. Certain conditions must be met before inputs are enabled, as shown in the table below.

To accept external remote inputs, assign the remote function and set remote to the control device. When external remote input is available , "AutoMode output" turns ON.

Except "SelProg", the signals execute each function when the signal starts in input acceptance condition. The function executes automatically. Therefore, no special programming is needed.

NOTE ☞ When an error occurs, you must execute a "Reset" to clear the error condition before any other remote input commands can be executed. Use the "Error output" and "Reset input" to monitor the error status and clear error conditions from the remote device.

Name	Initial	Description	Input Acceptance Condition (*1)
Start	0	Executes function selected at SelProg.(*	Ready output ON Error output OFF EStopOn output OFF SafeguardOn output OFF Pause input OFF Stop input OFF
SelProg1 SelProg2 SelProg4	1 2 3	Specifies the executing Main function number.　　　　　　　　　(*2)	
Stop	4	Stops all tasks and commands.	
Pause	5	Pauses all tasks.　　　　　(*3)	Running output ON
Continue	6	Continues the paused task.	Paused output ON Pause input OFF Stop input OFF
Reset	7	Resets emergency stop and error. (*4)	Ready output ON
SetMotorsOn	NA	Turns ON robot motors.　　　(*5)	Ready output ON EStopOn output OFF SafeguardOn output OFF SetMotorsOff input OFF
SetMotorsOff	NA	Turns OFF robot motors.	Ready output ON
Home	NA	Moves the Robot Arm to the home position defined by the user.	Ready output ON Error output OFF EStopOn output OFF SafeguardOn output OFF MotorsOn output ON Pause input OFF Stop input OFF
ForcePowerLow (*6)	NA	Stops all tasks and commands. Sets the motor power at Low. The status is Low power mode while the input is ON even executing Power High command.	Any time This input is acceptable even AutoMode output is OFF.

(*1) "AutoMode output" ON is omitted from the table. This is an input acceptance condition for all functions.

(*2) "Start input" executes Function specified by the "SelPlog1, 2, 4, and 3" bits.

Function	SelProg1	SelProg2	SelProg4
Main	0	0	0
Main1	1	0	0
Main2	0	1	0
Main3	1	1	0
Main4	0	0	1
Main5	1	0	1
Main6	0	1	1
Main7	1	1	1

0=OFF, 1=ON

(*3) "NoPause task" and "NoEmgAbort task" do not pause.
For details, refer to *EPSON RC+ 5.0 Online Help* or *Pause* in *SPEL⁺ Language Reference*.

(*4) Turns OFF the I/O output and initializes the robot parameter.
For details, refer to *EPSON RC+ 5.0 Online Help* or *Reset* in *SPEL⁺ Language Reference*.

(*5) Initializes the robot parameter.
For details, refer to *EPSON RC+ 5.0 Online Help* or *Motor* in *SPEL⁺ Language Reference*.

(*6) This is for the experienced. Make sure that you fully understand the input specification before using.
When the input changes from ON to OFF, all tasks and commands will stop.
It is supported by EPSON RC+ 5.0 Ver.5.4 and Firmware Ver. 1.10.*.*.

11.8 Remote Outputs

Remote outputs provide status for the Manipulator and Controller.

Remote outputs provide the assigned function using with any control device The outputs execute automatically. Therefore, no special programming is needed.

Name	Initial	Description
Ready	0	Turns ON when the controller startup completes and no task is running.
Running	1	Turns ON when task is running. However, turns OFF when "Paused output" is OFF.
Paused	2	Turns ON when pause task exists.
Error	3	Turns ON when an error occurs. Use "Reset input" to recover from the error.
EStopOn	4	Turns ON at Emergency Stop.
SafeguardOn	5	Turns ON when the safeguard is open.
SError	6	Turns ON when critical error occurs. When a critical error occurs, "Reset input" does not function. Reboot the controller to recover.
Warnig	7	Turns ON when warning occurs. The task runs as normal with the warning. However, be sure to eliminate the cause of the warning as soon as possible.
MotorsOn	NA	Turns ON when the motor is ON.
AtHome	NA	Turns ON when the robot is in the home position.
CurrProg1 CurrProg2 CurrProg4	NA	Indicates the running or the last main function number (*1)
AutoMode	NA	Turns ON in remote input acceptable status. (*2)
TeachMode	NA	Turns ON in TEACH mode.
ErrorCode1 . . . ErrorCode8192	NA	Indicates the error number.
InsideBox1 (*3) . . . InsideBox15	NA	Turns ON when the robot is in the approach check area. For details, refer to EPSON RC+ 5.0 *Online Help* or *Box* in *SPEL+ Language Reference*.
InsidePlane1(*3) . . . InsidePlane15	NA	Turns ON when the robot is in the approach check plane. For details, refer to EPSON RC+ 5.0 *Online Help* or *Plane* in *SPEL+ Language Reference*.
Power High (*4)	NA	Turns ON when the power status is High.

(*1) Outputs the current or the last function number of CurrProg1, CurrProg2, or CurrProg4.

Function	CurrProg1	CurrProg2	CurrProg4
Main	0	0	0
Main1	1	0	0
Main2	0	1	0
Main3	1	1	0
Main4	0	0	1
Main5	1	0	1
Main6	0	1	1
Main7	1	1	1

0=OFF, 1=ON

(*2) Remote function is available in the followings conditions.

- The setting is Auto mode and the control device is remote.

- The setting is Program mode and Remote I/O is enabled.

(*3) When the Controller firmware version is Ver.1.4.*.* or before, the remote outputs do not provide InsideBox1 to 15 and InsidePlane1 to 15.

(*4) It is supported by EPSON RC+ 5.0 Ver.5.4 and Firmware Ver. 1.10.*.*.

11.9 Remote Input Handshake Timing

The following charts indicate the timing sequences for the primary operations of the Controller.

The indicated time lapses (time durations) should be referred to only as reference values since the actual timing values vary depending on the number of tasks running, as well as CPU speed of the Controller. Check carefully and refer to the following charts for the timing interrelation when you enter an input signal.

During system design, make sure that you actuate only one remote input operation at a time, otherwise an error will occur.

The pulse width of an input signal must be 25 or more milliseconds to be detected.

[Unit: msec]

Timing Diagram for Operation Execution Sequence
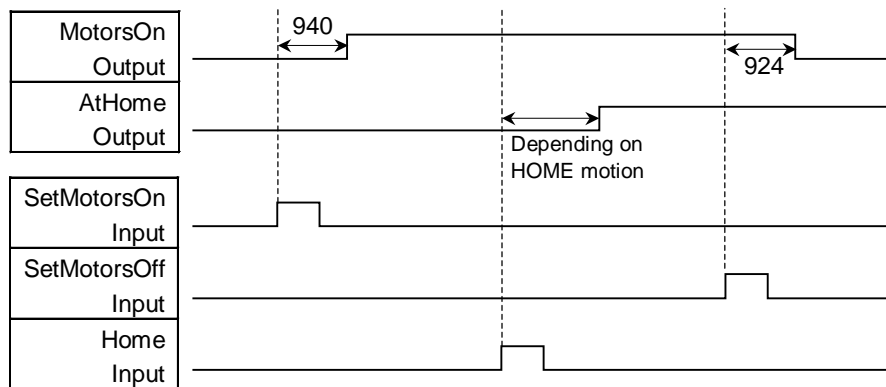
Timing Diagram for Program Execution Sequence
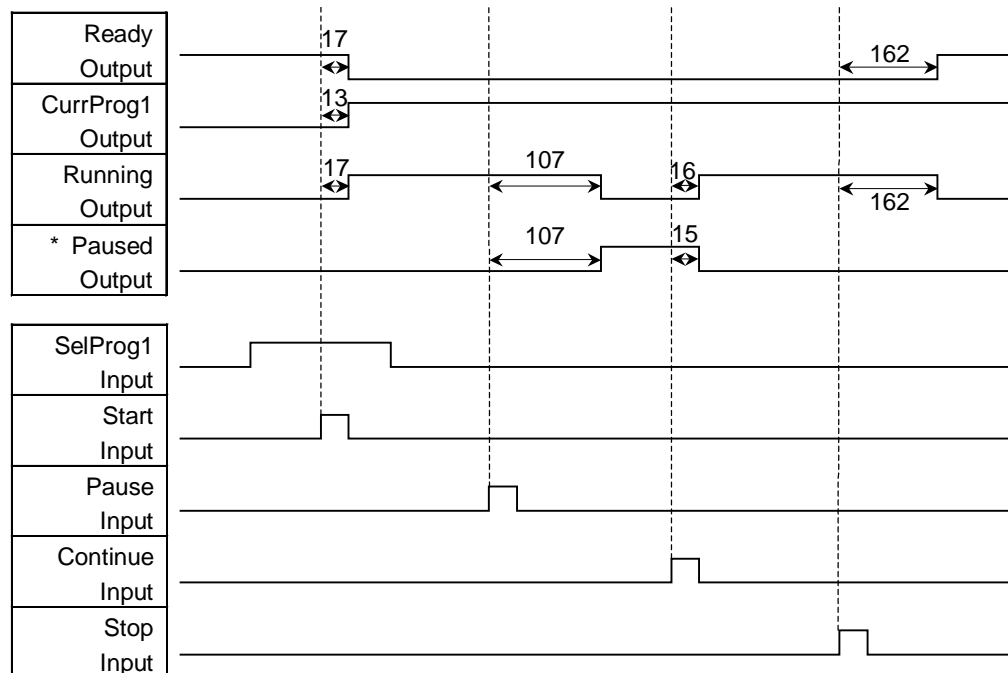
* The duration varies depending on the Quick Pause (QP) setting and the program's operating status at the time of Pause input

Timing Diagram for Safety Door Input Sequence
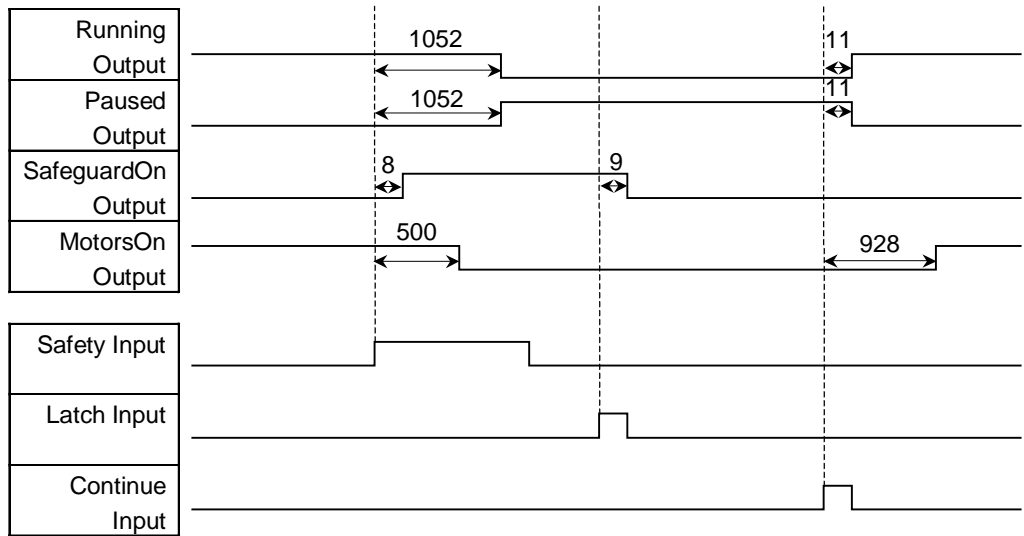
Timing Diagram for Emergency Stop Sequence
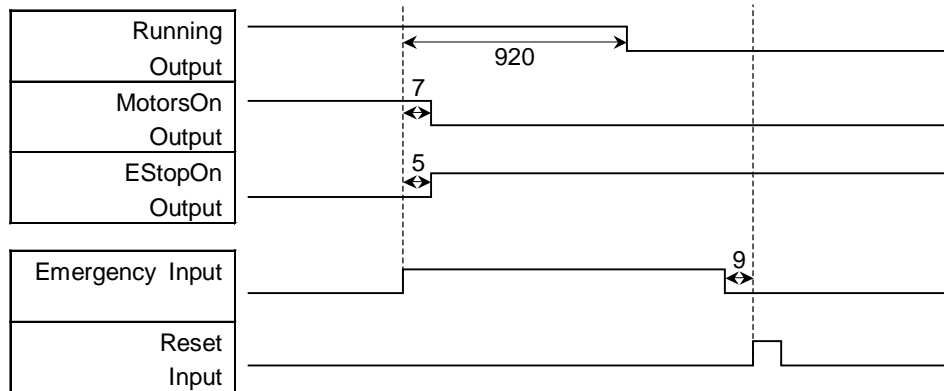
If an error occurs, the Error output will turn on. To clear the error, you must turn on the Reset input. No other inputs will be accepted when there is an error condition.

11.10 Remote Ethernet

Remote Ethernet makes it possible to control the robot and controller from external equipment by sending the remote commands through Ethernet (TCP/IP).

To set the remote Ethernet functions valid, follow the procedures below to configure the parameter.

(1) Select Controller from the Setup Menu and select the Remote Ethernet page.

(2) Configure the necessary items for the remote Ethernet control.

(3) Click **Apply** to save the new setting and click **Close**.
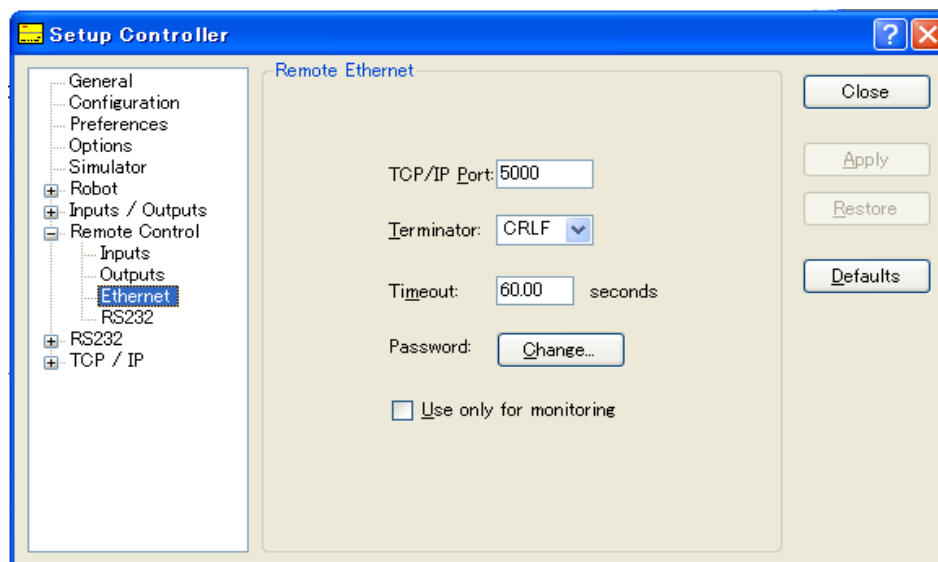
For the details of the dialog setting, refer to the section *5.12.2 Controller (Setup menu) - Setup: Controller: Remote*.

Following setting for the control device is not necessary when you select "Use only for monitoring", and only acquire the value using the Remote Ethernet control.

Set the control device to "Remote Ethernet" by the following procedure.

(1) Select Controller from the Setup Menu and select Controller Configuration page. Select "Remote Ethernet" in the [Controller device] box.
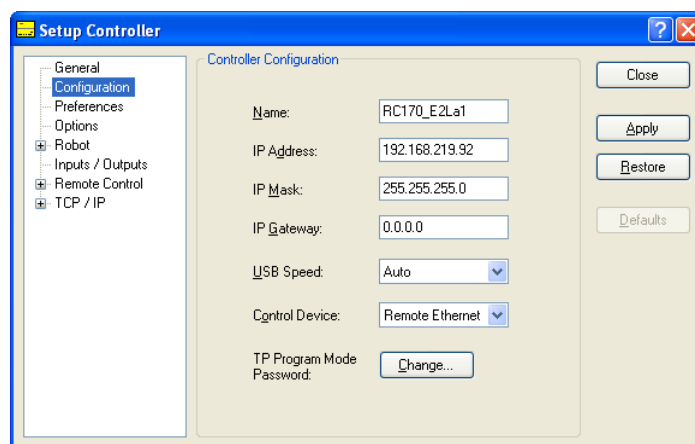
(2) Click **Apply** button to save the new settings, click **Close**.

For the details of the dialog setting, refer to the section *5.12.2 Controller (Setup menu) - Setup: Controller: Setup*.

Set the remote control available by the following procedure.

(1) Connect from client equipment to the specified port in the Remote Ethernet of the Controller.

(2) Specify the password set in the Remote Ethernet to the parameter and send the Login command.

(3) Client equipment has to wait until Auto (GetStatus command response) is ON, before execution of remote command.

(4) Now remote command will be accepted.
Each command executes the function the input acceptance condition.

Program debug from EPSON RC+ 5.0 development environment is capable as follows.

(1) Build a program as usual.

(2) Open the Run window and Click the Ethernet Enable button.

When you only acquire the value using the Remote Ethernet control, the Ethernet Enable button is not displayed. Click the Start button of the device specified as the control device.

(3) Now remote command will be accepted.

Brakepoint setting and output to the Run window is available.

NOTE
☞
If not Login within 5 minutes from external equipment, the connection will be cut down automatically. After Login, if no command is send within the timeout duration of the remote Ethernet, connection will be cut down. In this case, establish the connection again.

If error occurs execute the Reset command to clear the error condition before executing the operation command. To clear the error condition from external equipment by monitoring, use the "GetStatus" and "Reset" command.

⚠
CAUTION
■ If you set "0" in the [Timeout] box, time out duration is infinite. In this case, the task continues to execute even without the communication from client. This means the robot may keep moving and cause unexpected damage. Ensure the ways other than the communication to stop the task.

Remote command

Format: $ remote command {, parameter....} terminator

Remote command	Parameter	Contents	Input acceptance condition (*1)
Login	password	Start the Controller Remote Ethernet function Authentication by password Execute Login correctly, commands execution is enabled until Logout	Available any time (*10)
Logout		Exit Controller Remote Ethernet function After Logout, execute the Login command to start remote Ethernet function. Logout during task execution causes an error.	Available any time (*10)
Start	Function No.	Execute the function of specified number (*2)	Auto ON Ready ON Error OFF EStop OFF Safeguard ON
Stop		Stop all tasks and commands	Auto ON
Pause		Pause all tasks (*3)	Auto ON Running ON
Continue		Continue paused tasks	Auto ON Paused ON
Reset		Clear emergency stop and error (*4)	Auto ON Ready ON
SetMotorsOn		Turn ON the robot motor (*5)	Auto ON Ready ON EStop OFF Safeguard OFF
SetMotorsOff		Turn OFF the robot motor	Auto ON Ready ON
Home		Move the arm to home position defined by user	Auto ON Ready ON Error OFF EStop OFF Safeguard OFF
GetIO	I/O bit No.	Acquire the specified I/O bit	Available any time (*10)
SetIO	I/O bit No. & value	Set the I/O specified bit 1: Turn ON the bit 0: Turn OFF the bit	Ready ON
GetIOByte	I/O port No.	Acquire the specified I/O port (8 bit) (*8)	Available any time (*10)
SetIOByte	I/O port No. & value	Set the I/O specified port (8 bit) (*8)	Ready ON
GetIOWord	I/O word port No.	Acquire the specified I/O word port (16 bit) (*8)	Available any time (*10)

Remote command	Parameter	Contents	Input acceptance condition (*1)
SetIOWord	I/O word port No. & value	Set the I/O specified word port (8 bit) (*8)	Auto ON Ready ON
GetMemIO	Memory I/O bit No.	Acquire the specified memory I/O bit (*8)	Available any time (*10)
SetMemIO	Memory I/O bit No. & value	Set the specified memory I/O bit (*8) 1: Turn ON the bit 0: Turn OFF the bit	Auto ON Ready ON
GetMemIOByte	Memory I/O port No.	Acquire the specified memory I/O port (*8)	Available any time (*10)
SetMemIOByte	Memory I/O port No. & value	Set the specified memory I/O port (8 bit) (*8)	Auto ON Ready ON
GetMemIOWord	Memory I/O word port No.	Acquire the specified memory I/O word port (16 bit) (*8)	Available any time (*10)
SetMemIOWord	Memory I/O word port No. & value	Set the specified memory I/O word port (16 bit) (*8)	Auto ON Ready ON
GetVariable	Parameter name {, type}	Acquire the value of backup (Global Preserve) parameter (*6)	Available any time (*10)
	[Parameter name] (Array element), [Parameter name type], [Number to acquire]	Acquire the value of backup (Global Preserve) array parameter (*7)	
SetVariable	Parameter name & Value {, type}	Set the value in the backup (Global Preserve) parameter (*6)	Auto ON Ready ON
GetStatus		Acquire the Controller state	Available any time (*10)
Execute	Command string	Execute the command (*8) (*9)	Auto ON Ready ON Error OFF EStop OFF Safeguard OFF
Abort		Abort the command execution (*8)	Auto ON

(*1) The Controller state bit from GetStatus.

(*2) Execute the function specified in the Main[Function No.].

Function Name	Function No.
Main	0
Main1	1
Main2	2
Main3	3
Main4	4
Main5	5
Main6	6
Main7	7

(*3) Pause command is not available for "NoPause task" and "NoEmgAbort task".
For the details, refer to the help or the section "*Pause*" in the EPSON RC+5.0 Language reference manual.

(*4) I/O output will be turned OFF and the robot parameter will be initialized.
For the details, refer to the help or the section "*Reset*" in the EPSON RC+5.0 Language reference manual.

(*5) The robot parameter will be initialized.
For the details, refer to the help or the section "*Motor*" in the EPSON RC+5.0 Language reference manual.

(*6) Parameter type means {Boolean | Byte | Double | Integer | Long | Real | String}.
Type specified: for the backup parameters when the parameter name and type are same.
Type not specified: for the backup parameters when the parameter names are same.

(*7) Acquiring the value of array parameter is supported in the firmware Ver.1.10.*.* or later.

For the array element, specify an element you acquire as the following:

You need to specify an element if when acquiring from the head of the array.

1D array	Parameter name (0)	Acquire from the head.
	Parameter name (Element number)	Acquire from the specified element number.
2D array	Parameter name (0,0)	Acquire from the head.
	Parameter name (Element number 1, 2)	Acquire from the specified element number.
3D array	Parameter name (0,0,0)	Acquire from the head.
	Parameter name (Element number 1, 2, 3)	Acquire from the specified element number.

You cannot omit the parameter type and number to acquire.

You cannot specify a string for the parameter type.

Available number to acquire is up to 100. If you specify a number over the number of array elements, you have an error.

e.g.) "$GetVariable,gby2(3,0),Byte,3"

It acquires values of gby2(3,0), gby2(3,1), gby2(3,2) of Byte type 2D array parameter gby2.

(*8) Following functions are supported in the firmware 1.13.*.* or later.
    - Acquiring and setting I/O port,
    - Acquiring and setting I/O word port,
    - Acquiring and setting memory I/O,
    - Acquiring and setting memory I/O port,
    - Acquiring and setting memory I/O word port,
    - Execution and abortion of command

(*9) Specify the command and parameters in the double quotation marks.

Following commands are available while Execute is running.

Commands available while Execute is running

Remote Command
Abort
GetStatus

(*10) "Available anytime" applies only when the following conditions are met.

When "Remote Ethernet" is set as the control device,

or

"Remote Ethernet" is not set as the control device, but set to be used for monitoring.

Commands available for monitoring

When the Remote Ethernet control is not set as the control device but set to be used for monitoring, following commands are only available to be executed.

Remote command
Login
Logout
GetIO
GetIOByte
GetIOWord
GetMemIO
GetMemIOByte
GetMemIOWord
GetVariable
GetStatus

Response

When the Controller receives the command correctly, the response in the following format is shown in the executing command.

Command	Format
Remote command that acquire the value Except GetIO, GetVariable, and GetStatus	#[Remote command],[0] terminator
GetIO	#GetIO,[0 | 1] terminator *1
GetMemIO	#GetMemIO,[0 | 1] terminator *1
GetIOByte	#GetIOByte,[Hex string (00 to FF) of Byte (8Bit)] terminator
GetMemIOByte	#GetMemIOByte,[Hex string (00 to FF) of Byte (8Bit)] terminator
GetIOWord	#GetIOWord,[Hex string (0000 to FFFF) of Word (16Bit)] terminator
GetIOMemWord	#GetMemIOWord,[Hex string (0000 to FFFF) of Word (16Bit)] terminator
GetVariable	# GetVariable,[Parameter value] terminator
GetVariable (in case of array)	# GetVariable,[Parameter value 1],[Parameter value 2],...,terminator *4
GetStatus	#GetStatus,[Status],[Error, Warning code] tenimator Example) #GetStatus,aaaaaaaaaa,bbbb *2 *3
Execute	If the value is returned as a result of command execution #Execute,"[Execution result]" terminator

*1   [0 \| 1] I/O bit   ON: 1/ OFF: 0

*2   Status
      In the example above, 10 digits "aaaaaaaaaa" is for the following 10 flags.
      Teach/Auto/Warning/SError/Safeguard/EStop/Error/Paused/Running/Ready
      1 is ON /0 is OFF
      If Teach and Auto are ON, it is 1100000000.

*3   Error / Warning code
      It is indicated in 4 digits. If there is no error and warning, it is 0000.

e.g.)1 : #GetStatus,0100000001,0000
      The bits for Auto and Ready us ON (1).
      This means that AutoMode is ON and be in Ready state. Command execution is enabled.

e.g.)2 : #GetStatus,0110000010,0517
      This means the warning occurs during the operation. Take appropriate action for the warning code. (In this case, warning code is 0517)

Flag	Contents
Teach	Turn ON in the TEACH mode
Auto	Turn ON in the remote input acceptance condition

Flag	Contents
Warnig	Turn ON in the warning condition Task can be executed as usual even a warning condition. However, take action for the warning as soon as possible.
SError	Turn ON in the serious error condition When a serious error occurs, Reboot the Controller to recover from the error condition. "Reset input" is not available.
Safeguard	Turn ON with safety door open
EStop	Turn ON in the emergency condition
Error	Turn ON in the error condition Use "Reset input" to recover from the error condition.
Paused	Turn ON with paused task
Running	Turn ON with task executing Turn OFF when "Paused output" is ON
Ready	Turn ON with the Controller completed the startup and no task executing

*4   It returns values of specified number in the Number to acquire.

Error response

When the Controller can not receive the remote command correctly, the error response is shown in the following format.

Format: ![Remote command],[Error code] terminator

Error code	Contents
10	Remote command does not begin with $
11	Remote command is wrong Login is not executed
12	Remote command format is wrong
13	Login command password is wrong
14	Specified number to acquire is out of range (Less than 1 or more than 100) Number to acquire is omitted Specified a string parameter
15	Parameter is not existed Dimension of parameter is wrong Element out of range is called
19	Request time out
20	Controller is not ready
21	Cannot execute since the Execute is running
99	System error Communication error

Communication sequence
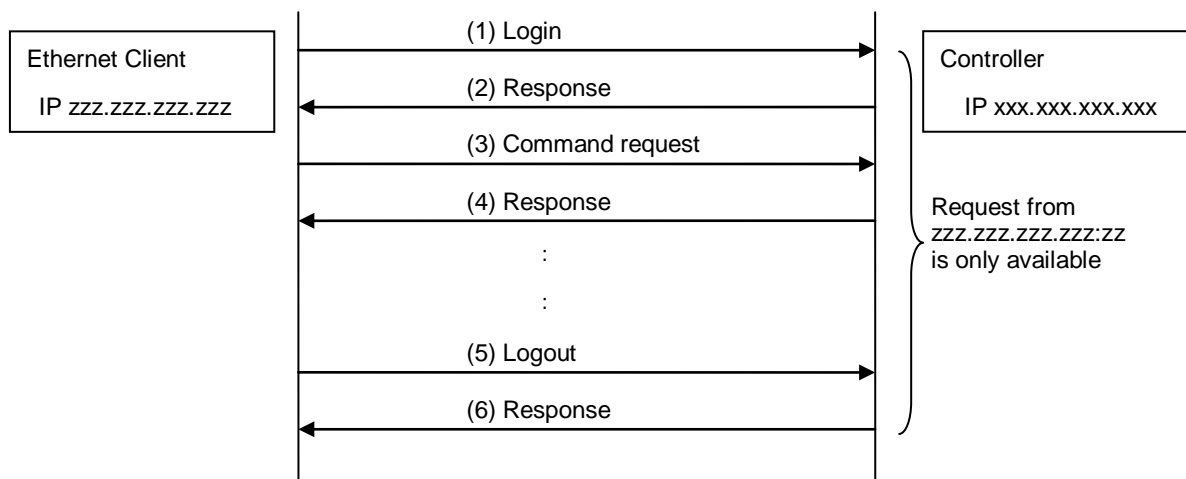
11.11 Remote RS232

Remote RS232 makes it possible to control the robot and controller from external equipment by sending the remote commands through RS-232C.

To set the remote RS232 functions valid, follow the procedures below to configure the parameter.

(1) Select Controller from the Setup Menu and select the Remote RS232 page.

(2) Configure the necessary items for the remote RS232 control.

(3) Click **Apply** to save the new setting and click **Close**.
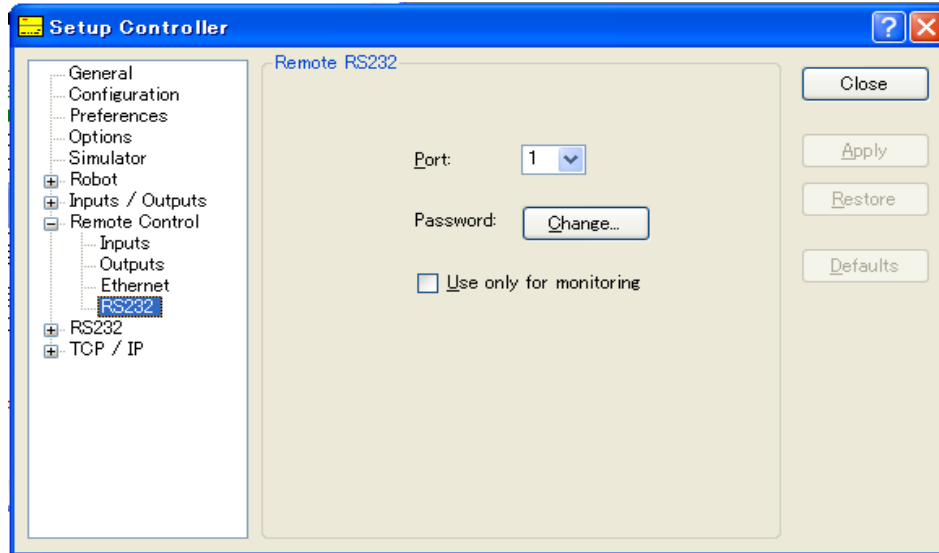
For the details of the dialog setting, refer to the section *5.12.2 Controller (Setup menu) - Setup: Controller: Remote*.

Following setting for the control device is not necessary when you select "Use only for monitoring", and only acquire the value using the remote RS232 control.

Set the control device to "Remote RS232" by the following procedure.

(1) Select Controller from the Setup Menu and select Controller Configuration page.
    Select "Remote RS232" in the [Controller device] box.
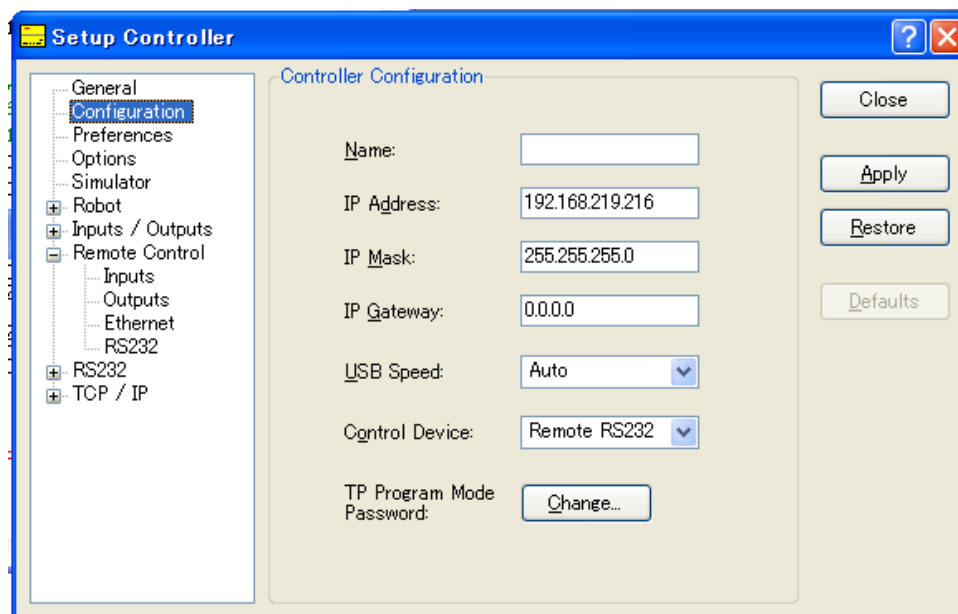
(2) Click **Apply** button to save the new settings, click **Close**.

For the details of the dialog setting, refer to the section *5.12.2 Controller (Setup menu) - Setup: Controller: Setup*.

Set the remote control available by the following procedure.

(1) Open RS-232C port that is connected from client equipment to the specified port in the Remote RS232 of the Controller, using the communication parameter specified in the RS-232C port setting.

(2) Send the remote start command (EOT).

(3) Specify the password set in the Remote RS232 to the parameter and send the Login command.

(4) Client equipment has to wait until Auto (GetStatus command response) is ON, before execution of remote command.

(5) Now remote command will be accepted.
    Each command executes the function when the input acceptance condition is satisfied.

Program debug from EPSON RC+ 5.0 development environment is capable as follows.

(1) Build a program as usual.

(2) Open the Run window and Click the RS232 Enable button.

    When you only acquire the value using the Remote RS232 control, the RS232 Enable button is not displayed. Click the Start button of the device specified as the control device.

(3) Now remote command will be accepted.

Brakepoint setting and output to the Run window is available.

NOTE
☞

After Login, if no command is send within the timeout duration of the RS-232C, the timeout error will be returned. In this case, re-execute from sending remote start command. If error occurs, execute the Reset command to clear the error condition before executing the operation command. To clear the error condition from external equipment by monitoring, use the "GetStatus" and "Reset" command.

	■ If you set "0" in the [Timeout] box, time out duration is infinite. In this case, the task continues to execute even without the communication from client. This means the robot may keep moving and cause unexpected damage. Ensure the ways other than the communication to stop the task.
CAUTION	

Remote Command

Remote start

Start the Remote RS232function of the Controller.

EOT
1byte

EOT：&H04(&H is hexadecimal)

Request format

STX	Command	Data	ETX	BCC
1byte	1Byte	Variable	1Byte	1Byte

STX ：&H02

ETX ：&H03

BCC：Checksum of sent and received data

XOR value from the command to ETX per 1Byte

Remote Command	Send command	Data	Description	Input Acceptance Condition (*1)
Login	'L' &H4C	password	Authentication by password Execute Login correctly, commands execution is enabled until Logout	Available any time (*10)
Logout	'l' &H6C		After Logout, execute the Login command to start remote RS232 function. Logout during task execution causes an error.	Available any time (*10)
Start	'G' &H47	Function No. (1Byte)	Execute the function of specified number Example) Execute 'main' &H02&H47&H00&H03&H44	Auto ON Ready ON Error OFF EStop OFF Safeguard ON
Stop	'Q' &H51		Stop all tasks and commands	Auto ON
Pause	'P' &H50		Pause all tasks	Auto ON Running ON
Continue	'C' &H43		Continue paused tasks	Auto ON Paused ON

Remote Command	Send command	Data	Description	Input Acceptance Condition (*1)
Reset	'R' &H52		Clear emergency stop and error	Auto ON Ready ON
SetMotorsOn	'M' &H4D		Turn ON the robot motor	Auto ON Ready ON EStop OFF Safeguard OFF
SetMotorsOff	'N' &H4E		Turn OFF the robot motor	Auto ON Ready ON
Home	'H' &H48		Move the arm to home position defined by user	Auto ON Ready ON Error OFF EStop OFF Safeguard OFF
GetIO	'i' &H69	I/O bit No. (2Byte)	Acquire the specified I/O bit Example) Acquire the I/O bit 1 &H02&H69&H0001&H03&H6B	Available any time (*10)
SetIO	'I' &H49	[I/O bit No.](2Byte) [value] (1Byte)	Set the I/O specified bit &H01: Turn ON the bit &H00: Turn OFF the bit Example) Turn ON the I/O bit 1 &H02&H49&H0001&H01&H03&H4A	Auto ON Ready ON
GetIOByte	'b' &H62	I/O port No.(1Byte)	Acquire the specified I/O port (8 bit) (*8) Example) Acquire the I/O port 1 &H02&H62&H01&H03&H60	Available any time (*10)
SetIOByte	'B' &H42	[I/O port No.](1Byte) [value] (1Byte)	Set the I/O specified port (8 bit) (*8) Example) Set &H0F to the I/O port 1 &H02&H42&H01&H0F&H03&H4F	Auto ON Ready ON
GetIOWord	'w' &H77	I/O word Port No. (1Byte)	Acquire the specified I/O word port (16 bit) (*8) Example) Acquire the I/O word port 1 &H02&H77&H01&H03&H75	Available any time (*10)
SetIOWord	'W' &H57	[I/O word port No.] (1Byte) [value] (2Byte)	Set the I/O specified word port (16 bit) (*8) Example) Set &H010F to the I/O word port 1 &H02&H57&H01&H010F&H03&H5B	Auto ON Ready ON
GetMemIO	'o' &H6F	Memory I/O bit No. (2Byte)	Acquire the specified memory I/O bit (*8) Example) Acquire memory I/O bit 1 &H02&H6F&H0001&H03&H6D	Available any time (*10)

Remote Command	Send command	Data	Description	Input Acceptance Condition (*1)
SetMemIO	'O' &H4F	[Memory I/O bit No.] (2Byte)[value] (1Byte)	Set the I/O specified bit (*8) &H01: Turn ON the bit &H00: Turn OFF the bit Example) Turn ON the memory I/O bit 1 &H02&H4F&H0001&H01&H03&H4C	Auto ON Ready ON
GetMemIOByte	't' &H74	Memory I/O port No. (1Byte)	Acquire the specified memory I/O port (8 bit) (*8) Example) Acquire the memory I/O port 1 &H02&H74&H01&H03&H76	Available any time (*10)
SetMemIOByte	'T' &H54	[Memory I/O port No.] (1Byte)[value] (1Byte)	Set the I/O specified port (8 bit) (*8) Example) Set &H0F to the memory I/O port 1 &H02&H54&H01&H0F&H03&H59	Auto ON Ready ON
GetMemIOWord	'u' &H75	Memory I/O word port No.(1Byte)	Acquire the specified memory I/O word port (16 bit) (*8) Example) Acquire the memory I/O word port 1 &H02&H75&H01&H03&H77	Available any time (*10)
SetMemIOWord	'U' &H55	[Memory I/O word port No.] (1Byte) [value](1Byte)	Set the I/O specified word port (16 bit) (*8) Example) Set &H010F to the memory I/O word port 1 &H02&H55&H01&H010F&H03&H59	Auto ON Ready ON
GetVariable	'v' &H76	[Parameter name],(&H2C) [type](1Byte)	Acquire the value of backup (Global Preserve) parameter Example) Acquire the Global Integer g_Status &H02&H76&H67&H5F&H53&H74&H61&H74 &H75&H73&H2C&H03&H03&H56	Available any time (*10)
		[Parameter name] (&H2C) (Array element) (&H2C), [Parameter type](1Byte), (&H2C) [Number to acquire] (2Byte)	Acquire the value of backup (Global Preserve) array parameter Example) Acquire all of Global Integer g_intArray(10) &H02&H76&H67&H5F&H69&H6E&H74&H41 &H72&H72&H61&H79&H2C &H0000&H2C&H03&H2C&H000A&H03&H42 E Example) Acquire 10 elements from elements (3,5,0) of Global Integer g_int3Array(10,10,10) &H02&H76&H67&H5F&H69&H6E&H74&H33 &H41&H72&H72&H61&H79&H2C &H0003&H2C&H0005&H2C&H0000&H2C&H 03&H2C&H000A&H03&H77	

Remote Command	Send command	Data	Description	Input Acceptance Condition (*1)
SetVariable	'V' &H56	[Parameter name], (&H2C) [value] (type size) (&H2C),[type] (1Byte)	Set the value in the backup (Global Preserve) parameter Example) Set &H0 to Global Integer g_Status &H02&H56&H67&H5F&H53&H74&H61&H74 &H75&H73&H2C&H0000&H2C&H03&H03& H5A	Auto ON Ready ON
GetStatus	'S' &H53		Acquire the Controller state	Available any time (*10)
Execute	'X' &H58	Command string	Execute the command (*8) (*9) Example) Execute 'print here' &H02&H58&H22&H70&H72&H69&H6E&H7 4&H20&H68&H65&H72&H65&H22&H03&H 10	Auto ON Ready ON Error OFF EStop OFF Safeguard OFF
Abort	'A' &H41		Abort the command execution (*8)	Auto ON

(*1) The Controller state bit from GetStatus.

(*2) Execute the function specified in the Main[Function No.].

Function Name	Function No.
Main	0
Main1	1
Main2	2
Main3	3
Main4	4
Main5	5
Main6	6
Main7	7

(*3) Pause command is not available for "NoPause task" and "NoEmgAbort task".
      For the details, refer to the help or the section "*Pause*" in the EPSON RC+5.0 Language reference manual.

(*4) I/O output will be turned OFF and the robot parameter will be initialized.
      For the details, refer to the help or the section "*Reset*" in the EPSON RC+5.0 Language reference manual.

(*5) The robot parameter will be initialized.
       For the details, refer to the help or the section "*Motor*" in the EPSON RC+5.0 Language reference manual.

(*6) Parameter type

Parameter type	Type value(1Byte)
Boolean	&H00
Byte	&H01
Double	&H02
Integer	&H03
Long	&H04
Real	&H05
String	&H06

For the backup parameters when the parameter name and type are same.

(*7) For the array element, specify an element you acquire as the following:

You need to specify an element when acquiring from the head of the array.

Specify the array element in 2Byte value.

1D array	Parameter name&H2C&H0000	Acquire from the head.
	Parameter name, element number.	Acquire from the specified element number.
2D array	Parameter name &H2C&H0000&H2C&H0000	Acquire from the head.
	Parameter name, element number 1, element number 2	Acquire from the specified element number.
3D array	Parameter name &H2C&H0000&H2C&H0000&H2C&H0000	Acquire from the head.
	Parameter name, element number 1, element number 2, element number 3	Acquire from the specified element number.

You cannot specify a string for the parameter type.

Available number to acquire is up to 100. If you specify a number over the number of array elements, you have an error.

(*8) Following functions are supported in the firmware 1.13.*.* or later.
- Acquiring and setting I/O port,
- Acquiring and setting I/O word port,
- Acquiring and setting memory I/O,
- Acquiring and setting memory I/O port,
- Acquiring and setting memory I/O word port,
- Execution and abortion of command

(*9) Specify the command and parameters in the double quotation marks.

Following commands are available while Execute is running.

Commands available while Execute is running

Remote Command
Abort
GetStatus

(*10) "Available any time" applies only when the following conditions are met.

When "Remote RS232" is set as the control device,

or

"Remote RS232" is not set as the control device, but set to be used for monitoring.

Commands available for monitoring

When the Remote RS232 control is not set as the control device but set to be used for monitoring, following commands are only available to execute.

Remote command
Login
Logout
GetIO
GetIOByte
GetIOWord
GetMemIO
GetMemIOByte
GetMemIOWord
GetVariable
GetStatus

Response

When the Controller receives the command correctly, the response in the following format is shown in the executing command.

Response Format

ACK	Command	Data	ETX	BCC
1Byte	1Byte	Variable	1Byte	1Byte

ACK : &H06

ETX : &H03

BCC : Checksum of sent and received data

XOR value from the command to ETX per 1Byte

Command	Format	
Remote command that acquire the value Except GetIO, GetVariable, and GetStatus	[ACK][Command](1Byte)[ETX][BCC]	
GetIO	[ACK] 'i'[&H00 | &H01] [ETX][BCC]	*1
GetMemIO	[ACK] 'o'[&H00 | &H01] [ETX][BCC]	*1
GetIOByte	[ACK] 'b'[Byte value(8Bit) (&H00〜&HFF)] [ETX][BCC]	
GetMemIOByte	[ACK] 't'[Byte value(8Bit) (&H00〜&HFF)] [ETX][BCC]	
GetIOWord	[ACK] 'w'[Word value(16Bit) (&H0000〜&HFFFF)] [ETX][BCC]	

Command	Format
GetIOMemWord	[ACK] 'u'[Word value(16Bit) (&H0000～ &HFFFF)] [ETX][BCC]
GetVariable	[ACK] 'u'[parameter] [ETX][BCC]
GetVariable (in case of array)	[ACK] 'u'[parameter 1][parameter 2]... [ETX][BCC] *4
GetStatus	[ACK] 'S' [Status],[Error, Warning code] [ETX][BCC] Example) [ACK] 'S'[aaaaaaaaaa][bbbb][ETX][BCC] *2 *3
Execute	If the value is returned as a result of command execution [ACK] 'X'"[Execution result]" [ETX][BCC]

*1   [&H00 | &H01] I/O bit   ON: &H01/ OFF: &H00

*2   Status
      In the example above, 10 digits [aaaaaaaaaa] is for the following 10 flags.
      Teach/Auto/Warning/SError/Safeguard/EStop/Error/Paused/Running/Ready
      &H01 is ON /&H00 is OFF
      If Teach and Auto are ON, it is
      [&H00&H01&H00&H00&H00&H00&H00&H00&H00&H01].

*3   Error / Warning code
      It is indicated in 4 digits. If there is no error and warning, it is 0000
      (&H30&H30&H30&h30).

e.g.)1 : [ACK]
      'S[&H00&H01&H00&H00&H00&H00&H00&H00&H00&H01][&H30&H30&H
      30&h30]

      The bits for Auto and Ready us ON (&H01).
      This means that AutoMode is ON and be in Ready state. Command execution is
      enabled.

e.g.)2 : :[ACK]
      'S[&H00&H01&H01&H00&H00&H00&H00&H00&H01&H00][&H30&H35&H
      31&h37]
      This means the warning occurs during the operation. Take appropriate action for
      the warning code. (In this case, warning code is 0517)

Flag	Contents
Teach	Turn ON in the TEACH mode
Auto	Turn ON in the remote input acceptance condition
Warnig	Turn ON in the warning condition Task can be executed as usual even a warning condition. However, take action for the warning as soon as possible.
SError	Turn ON in the serious error condition When a serious error occurs, Reboot the Controller to recover from the error condition. "Reset input" is not available.
Safeguard	Turn ON with safety door open
EStop	Turn ON in the emergency condition
Error	Turn ON in the error condition Use "Reset input" to recover from the error condition.

Flag	Contents
Paused	Turn ON with paused task
Running	Turn ON with task executing Turn OFF when "Paused output" is ON
Ready	Turn ON with the Controller completed the startup and no task executing

*4   It returns values of specified number in the Number to acquire.

Error response

When the Controller cannot receive the remote command correctly, the error response is shown in the following format.

NAK 1Byte	Command 1Byte	Error code 2Byte	ETX 1Byte	BCC 1Byte

NAK : &H15

ETX : &H03

BCC : Checksum of sent and received data

XOR value from the command to ETX per 1Byte

Error code	Contents
10	Remote command does not begin with $
11	Remote command is wrong Login is not executed
12	Remote command format is wrong
13	Login command password is wrong
14	Specified number to acquire is out of range (Less than 1 or more than 100) Number to acquire is omitted Specified a string parameter
15	Parameter is not existed Dimension of parameter is wrong Element out of range is called
16	BCC is wrong
19	Request time out
20	Controller is not ready
21	Cannot execute since Execute is running
99	System error Communication error

Communication sequence

RS-232C RemoteController Port zz		180Controller Port xx
	①EOT →	
	②Login →	
	← ③Response	
	④Command request →	Request from the Remote Controller is only available
	← ⑤Response	
	: : :	
	⑥Logout →	
	← ⑦Response	
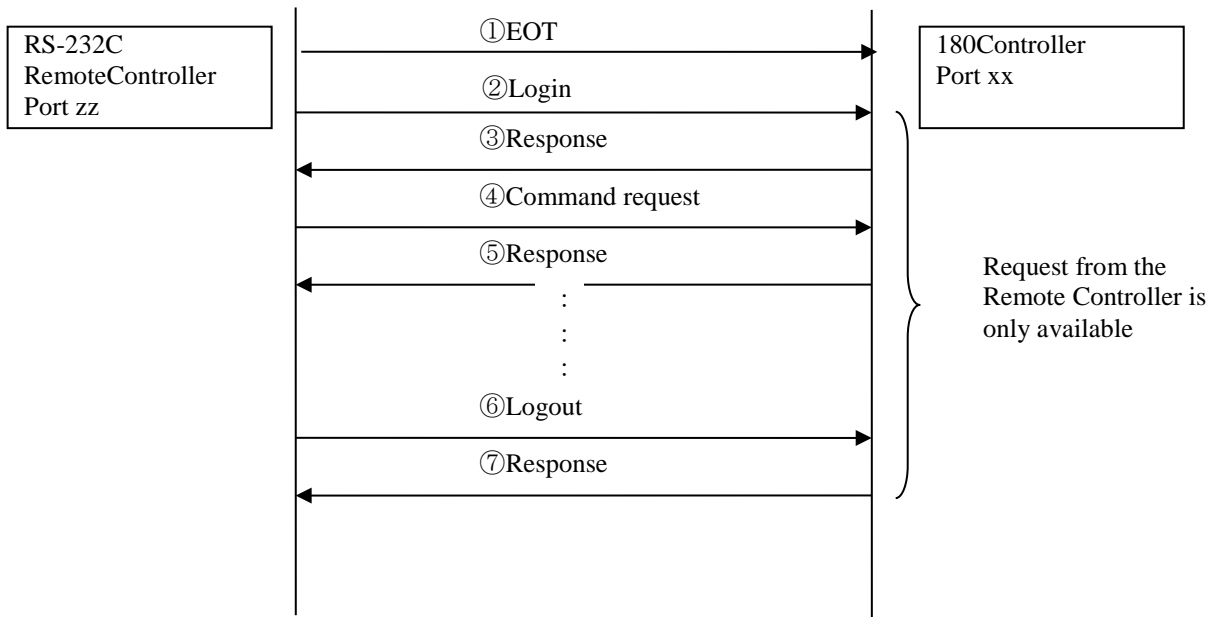
12. RS-232 Communications

The RC180 controllers can support up to 8 optional RS232 ports, 4 ports per board.

The RC90 controllers can support one port as standard and up to 4 optional RS232 ports, 2 ports per board.

Refer to the controller manual for instructions on how to install RS232 boards.

12.1 RS-232 Software Configuration

To configure an RS-232 port

1. Select Controller from the Setup Menu and open the **RS232** folder on the tree located on the left side of the dialog.

NOTE
☞
If RS232 boards have not been installed on the controller, you will not see the corresponding ports in the tree.

2. Select the port you want to configure.

3. Change the settings as desired.

4. Click **Apply** to save the new settings.
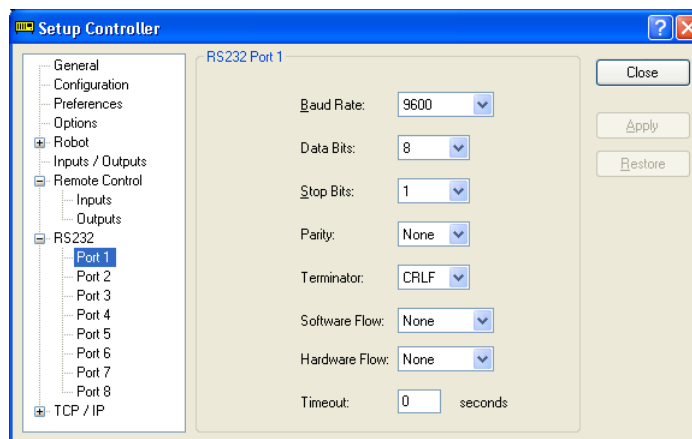
NOTE
☞
If the several ports are used in the communication at one time with more than 19200 baud rate, error 2929 or 2922 may occur. In this case, select the lower baud rate or avoid using several ports at one time.

12.2 RS-232 Commands

Here is a list of all of the commands associated with RS-232 communications. For details, please see the online help or SPEL$^+$ Language Reference Manual.

OpenCom	Opens a communications port.
ChkCom	Returns port status: the number of bytes waiting to be read or error condition.
CloseCom	Closes a communications port.
SetCom	Sets communications port parameters at runtime or from the Command window.
Print #	Sends characters out of the port.
Input #	Receives characters from the port into one or more variables.
Line Input #	Receives one line characters from the port into one string variable.
Read #	Receives one or more characters from the port into one string variable.
ReadBin #	Receives one or more bytes from the port.
Write #	Sends characters out of the port.
WriteBin #	Sends one or more bytes out of the port.

13. TCP / IP Communications

EPSON RC+ 5.0 supports eight TCP/IP ports that allow peer to peer communications. This chapter contains instructions on using TCP/IP, including IP addresses, Windows TCP/IP configuration, and SPEL$^+$ examples.

13.1 TCP/IP Setup

Before you can use TCP/IP communications between PCs and controllers, you must configure your network. The following sections describe basic network configuration.

13.1.1 Ethernet Hardware

The RC180 controller includes a built in Ethernet interface with an RJ45 connector accessible from the controller front panel. It supports 10BaseT (10 Mbps) and 10BaseTX (100 Mbps).

Your PC will need a 10BaseT 10/100 adapter to communicate with a controller via Ethernet.

13.1.2 IP Addresses

The RC180 controller has a fixed IP address that you can configure from EPSON RC+ 5.0. To configure the IP address, mask, and gateway for an RC180 controller, see the section *Setup | Controller | Configuration* in the chapter *EPSON RC+ 5.0 GUI* for instructions.

The following table shows a typical IP address configuration.

Host Name	IP Address	Subnet	Subnet Mask
PC1	192.168.0.1	192.168.0	255.255.255.0
PC2	192.168.0.2	192.168.0	255.255.255.0
Controller1	192.168.0.3	192.168.0	255.255.255.0
Controller2	192.168.0.4	192.168.0	255.255.255.0

In this example, the network address (subnet) is 192.168.0. With a subnet mask of 255.255.255.0, there can be 254 hosts on this subnet (0 and 255 cannot be used).

Refer to the Microsoft Windows operating system manual for instructions on setting the PC IP address.

13.1.3 IP Gateway

If you are connecting PCs and controllers on different networks, you will need to route traffic between the networks using one or more routers. Each device communicating via Ethernet will need to have their default gateway address set to the address of the router for its subnet.

To configure the gateway address for the RC180 controller, see the section *Setup | Controller | Configuration* in the chapter *EPSON RC+ 5.0 GUI* for instructions.

13.1.4 Testing Windows TCP/IP setup

Use the ping command from a Command Window to test communications.

First, do a loopback test to check if you can ping your own address by using the local IP address:

```
C:\>ping 127.0.0.1
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<10ms TTL=128
Reply from 127.0.0.1: bytes=32 time<10ms TTL=128
Reply from 127.0.0.1: bytes=32 time<10ms TTL=128
Reply from 127.0.0.1: bytes=32 time<10ms TTL=128
C:\>
```

Ping your PC's IP address:

```
C:\>ping 192.168.0.1
Pinging 192.168.0.1 with 32 bytes of data:
Reply from 192.168.0.1: bytes=32 time<10ms TTL=128
Reply from 192.168.0.1: bytes=32 time<10ms TTL=128
Reply from 192.168.0.1: bytes=32 time<10ms TTL=128
Reply from 192.168.0.1: bytes=32 time<10ms TTL=128
C:\>
```

Now ping controller on the network. :

```
C:\>ping 192.168.0.3
Pinging pc2 [192.168.0.3] with 32 bytes of data:
Reply from 192.168.0.3: bytes=32 time<10ms TTL=128
Reply from 192.168.0.3: bytes=32 time<10ms TTL=128
Reply from 192.168.0.3: bytes=32 time<10ms TTL=128
Reply from 192.168.0.3: bytes=32 time<10ms TTL=128
C:\>
```

13.2 TCP/IP Software Configuration

You can configure TCP/IP settings for the RC180 controller in a SPEL$^+$ program using the SetNet command. You can also configure settings from the TCP/IP tab on the Setup | Controller dialog.
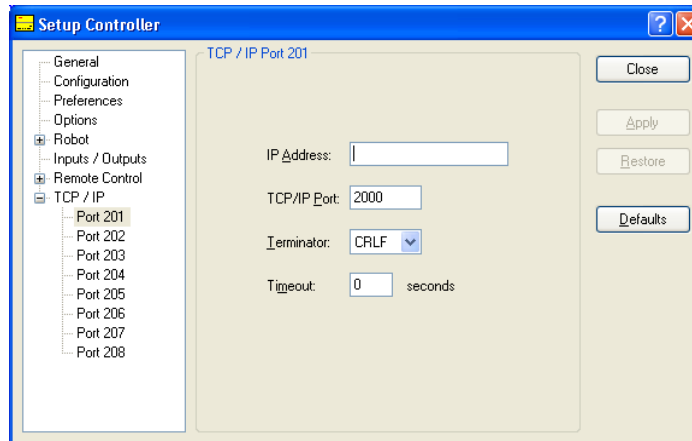
To configure a TCP/IP port

1. Select Controller from the Setup Menu and select the page for the TCP/IP port you want to configure.

2. Enter the IP address for the controller or PC that you want this controller to communicate with.

NOTE
☞
The controller does not support DNS, so you must specify an IP address for the host you are communicating with. You cannot specify a name for the host.

3. Enter the TCP/IP port number. This must be the same port number that is used on the host device. It must be different from any of the other TCP/IP port numbers used for the other TCP/IP ports.

4. Change the other settings as desired.

5. Click **Apply** to save the new settings and click **Close**.

13.3 TCP/IP Commands

Here is a list of all of the commands associated with TCP/IP communications. For details, please see the online help or SPEL$^+$ Language Reference Manual.

OpenNet	Opens a TCP/IP port.
ChkNet	Returns port status: the number of bytes waiting to be read or error condition.
CloseNet	Closes a TCP/IP port.
SetNet	Sets communications port parameters at runtime or from the Command window.
Print #	Sends characters out of the port.
Input #	Receives characters from the port into one or more variables.
Line Input #	Receives one line characters from the port into one string variable.
Read #	Receives one or more characters from the port into one string variable.
ReadBin #	Receives one or more bytes from the port.
Write #	Sends characters out of the port.
WriteBin #	Sends one or more bytes out of the port.

13.4 TCP/IP Example

This example shows how to configure 2 PCs running EPSON RC+ 5.0 to communicate using TCP/IP. Note that for the OpenNet command, one PC is the server, and the other is a client.

Controller #1:

Port: #201

IP Address: 192.168.0.1

TCP/IP Port: 1000

```
Function tcpip1
   String reply$
   OpenNet #201 As Server
   WaitNet #201
   Do
      Print "Sending data to host 2"
      Print #201, "Data from host 1"
      Input #201, reply$
      Print "Received reply: ", reply$
   Loop
Fend
```

Controller #2:

Port: #201

IP Address: 192.168.0.2

TCP/IP Port: 1000

```
Function tcpip2
   String data$
   OpenNet #201 As Client
   WaitNet #201
   Do
      Input #201, data$
      Print "Received '", data$, "' from host 1"
      Print #201, "Received data"
   Loop
Fend
```

14. ECP Motion

14.1 Overview

An ECP (external control point) motion is when the robot arm holding a part follows a specified trajectory (part's edges, etc.) using an outside fixed tool.
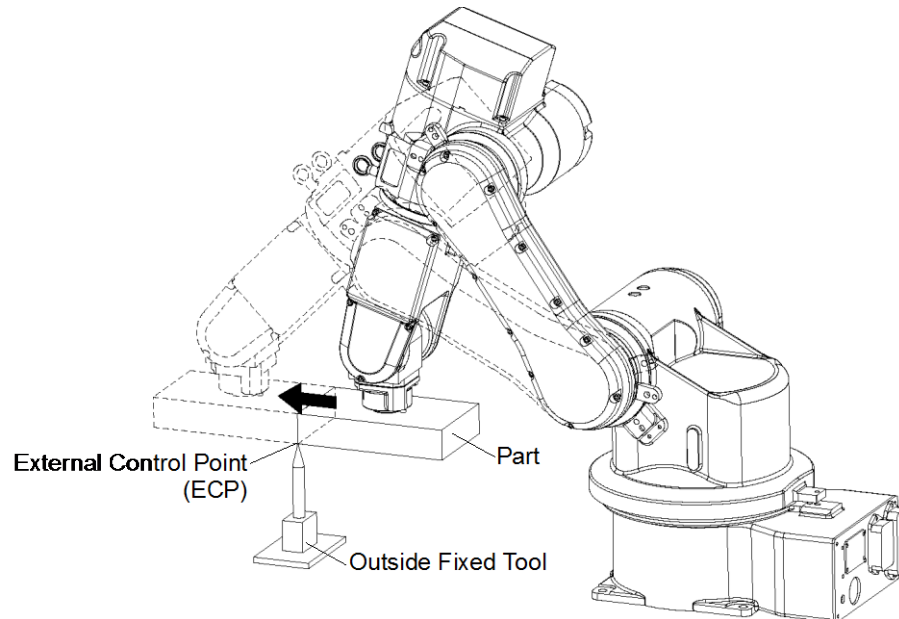
The ECP option supports the following:

- ECP definition by ECPSet statement and selection by ECP statement

- ECP motion commands (additional functions of Move, Arc3, Curve and CVMove commands)

- Teaching with ECP jogging

This option is available for SCARA, Cartesian and 6-Axis robots. Also, it can be used with multi-robot systems.

Up to 15 ECP coordinate systems can be defined.

14.1.1 How to move the arm with ECP motion

In the following paragraphs, the process for moving the 6-Axis robot arm with ECP motion is explained as an example.

1. Setting the ECP

The ECP (external control point) is coordinate system data used for defining the robot position and orientation at a processing point on the tip of the outside fixed tool.

The ECP should be defined based on the robot coordinate system or desired local coordinate system.

For example, when a drawing shows that the ECP is located at X=300, Y=300, Z=300 based on the robot coordinate system, specify it as shown below.

```
ECPSet 1,XY(300,300,300,0,0,0)   ' Defines ECP No.1
```

When you have no ECP location data, you can specify it by teaching.

As an example, attach the tool of which you know the data precisely and bring the tip of the tool close to the ECP and then teach its position anywhere as P0. Then, specify the ECP using P0 coordinate data as shown below.

```
ECPSet 1,P0 :U(0) :V(0) :W(0)    ' Defines ECP No.1
```

The orientation data (U, V, W) were set to 0 in the above examples. In these cases, the orientation in the ECP coordinate system is equal to that in the reference robot coordinate system.

You can specify U, V, and W coordinates in the ECP coordinate system. However, this data is valid only during the tangential correction mode ON in the Curve statement and ECP jog motion.
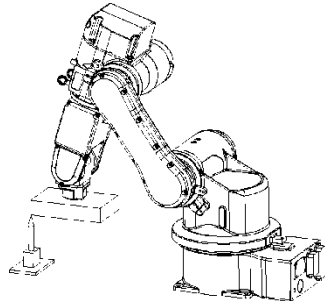
2. Teaching

Teach the point data while moving the robot arm holding the actual part. In this section, the part is assumed a rectangular solid and the arm is moved straight so that it touches one side of the part to the ECP specified in the former section *1. Setting the ECP*.

For details of teaching, refer to the section *Jog and Teach Command (Tools Menu)* in the chapter *The EPSON RC+ 5.0 GUI.*

2-1 Teaching the motion start point

Move the arm to the motion start point and teach it as P1.

2-2 Teaching the motion end point

Move the arm to the motion end point and teach it as P2.

NOTE
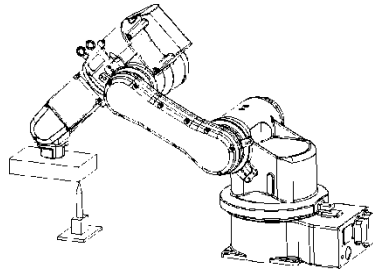
ECP Jog Motion:

The ECP jog motion is added to available jog motions for teaching besides Joint, World, and Tool jog motions.

The ECP jog motion is based on the selected ECP coordinate systems.

3. Executing Motion

To move the arm with ECP motion, add an "ECP" parameter to a motion command.

```
Go P1' Moves the arm to the motion start point
Move P2 ECP' Executes ECP motion
```

Use the Arc3 command to move the arm in an arc trajectory with the fixed tool. Use the Curve and CVMove commands to move the arm in cubic spline curves.

15. Installing Controller Options

When you purchase options with your system, the options are already installed on your system. However, you can enable additional options on site.

The RC180 Controller has option software keys that are enabled from EPSON RC+ 5.0.
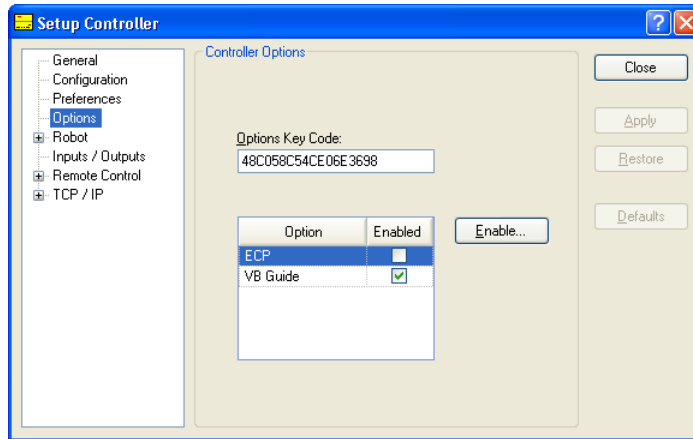
To see what options are enabled on your system, select Setup | Controller | Options. The following dialog will be displayed, showing which options are enabled.

Item	Description
Option	Name of the option.
Enabled	Indicates that the option is enabled in the controller.

To enable an option on site

1. Write down the controller options key code. You can view this from the Setup | Controller | Options page.

2. Call your distributor to purchase the option and give him the options code.

3. You will receive a code to enable the option from your distributor.

4. Select the option to enable on the grid, and then click the Enable button.

5. Enter in the code you received from your distributor.

NOTE
☞
Key code is case sensitive.
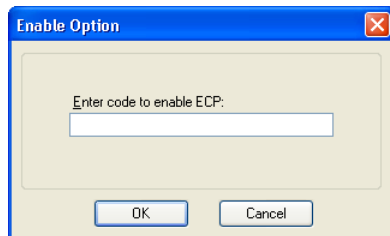
Enabling an option

16. Software License Agreement

THIS IS A CONTRACT. CAREFULLY READ ALL THE TERMS AND CONDITIONS CONTAINED IN THIS AGREEMENT. INSTALLING THE SOFTWARE (EPSON RC+ 5.0) INDICATES YOUR ACCEPTANCE OF THESE TERMS AND CONDITIONS. IF YOU DO NOT AGREE TO THESE TERMS AND CONDITIONS, SIMPLY DO NOT INSTALL OR USE THE EPSON RC+ 5.0 SOFTWARE.

LICENSE

SEIKO EPSON CORPORATION (the "Licenser") hereby grants to you (the "Licensee") a non-exclusive and transferable right to use the EPSON RC+ 5.0 Software program and documentation (the "Licensed Materials"). You may only use this software on one (1) CPU. You may not use, copy, or modify the Licensed Materials, in whole or in part, except as expressly provided for by this agreement.

OWNERSHIP

As the Licensee you own the magnetic or other physical media on which the Licensed Materials are recorded or fixed, but the Licenser retains sole and exclusive title to and ownership of the Licensed Materials recorded on the original disk and all subsequent copies regardless of the form or media in or on which the original and other copies may exist. By paying the fee required for this license, you do not become the owner of the Licensed Materials, but are entitled to use the Licensed Materials according to the terms of this agreement. You acknowledge that the Licensed Materials are not your property and understand that giving away or selling copies of the Licensed Materials is theft.

TERM

This license is effective until terminated. You may terminate this license by destroying the Licensed Materials together with any backup copies that may have been made. This license will also terminate if you fail to comply with any term or conditions of this Agreement. You agree upon such termination to destroy the Licensed Materials together with any copies which may have been made.

BACKUP AND TRANSFER

The Licensed Materials are copyrighted and contain proprietary information and trade secrets of the Licenser. Unauthorized copying, modifying or reproducing of the Licensed Materials, even if modified, merged or included with other software, is expressly forbidden. You may make one (1) archival copy of the Licensed Materials for the sole purpose of backing up the Licensed Materials. In no event does the limited copying or reproduction permitted hereunder include the right to decompile, disassemble or electronically transfer the Licensed Materials, or translate the Licensed Materials into another language. You may sell your license rights in the software to another party. If you sell your license rights in the Licensed Materials you must at the same time transfer the documentation and the backup copy or destroy the backup copy. You cannot sell your license rights in the Licensed Materials to another party unless that party also agrees to the terms and conditions of this agreement.

PROTECTION AND SECURITY

You agree not to deliver or otherwise make available the Licensed Materials or any part thereof, including, without limitation, the object code, to any person other than the Licenser or its employees, except for purposes specifically related to your use of the Licensed Materials on one (1) CPU, without the prior written consent of the Licenser. You agree to use your best efforts and take all reasonable steps to safeguard the Licensed Materials to ensure that no unauthorized person shall have access thereto and that no unauthorized copy, publication, disclosure or distribution thereof, in whole or in part, in any form, shall be made. You recognize that the Licensed Materials contain valuable confidential information and trade secrets.

LIMITED WARRANTY

The only warranty the Licenser makes to you in connection with this license to use the Licensed Materials is that the media on which the Licensed Materials are recorded will be replaced without charge, as long as the original diskette(s) are returned to the licenser, with satisfactory proof of date of purchase, within ninety (90) days of purchase. This warranty is limited to the Licensee and is not transferable. The foregoing warranty does not extend to any Licensed Materials that have been damaged as a result of accident, misuse or abuse.

EXCEPT FOR THE LIMITED WARRANTY DESCRIBED ABOVE, THESE LICENSED MATERIALS ARE PROVIDED "AS IS". THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE OF THE LICENSED MATERIALS IS ASSUMED BY YOU. THE LICENSER DOES NOT WARRANT, GUARANTEE OR MAKE ANY REPRESENTATIONS REGARDING THE USE OF, OR THE RESULTS OBTAINED WITH THE LICENSED MATERIALS IN TERMS OF CORRECTNESS AND RELIABILITY OR LEGALITY. THE ABOVE IS THE ONLY WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED. INCLUDING, WITHOUT LIMITATION, LOSS OF PROFITS OR INABILITY TO USE THE LICENSED MATERIALS, EVEN IF THE LICENSER OR SUCH OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. YOU AGREE THAT YOUR EXCLUSIVE REMEDIES, AND THE LICENSER'S OR SUCH OTHER PARTY'S ENTIRE LIABILITY WITH RESPECT TO THE LICENSED MATERIALS SHALL BE AS SET FORTH HEREIN, AND IN NO EVENT SHALL THE LICENSER'S OR SUCH OTHER PARTY'S LIABILITY FOR ANY DAMAGES OR LOSS TO YOU OR ANY OTHER PARTY EXCEED THE LICENSED FEE PAID FOR THE LICENSED MATERIALS.

ENTIRE AGREEMENT

This Agreement represents the entire agreement between Licenser and Licensee with respect to the License and use of the Licensed Materials.

SEVERABILITY

If any provision or a portion of this agreement is determined to be invalid under any applicable law, it shall be deemed omitted and the remaining provisions and partial provisions of this agreement shall continue in full force and effect.

NOTICE

Any notice or other communication relating to this license agreement to be sent to the Licenser must be mailed by certified mail to your distributor.

Appendix A: Software Installation

NOTE
☞
EPSON RC+ 5.0 supports the following OS.

　Windows XP Professional Service Pack 3
　　　(EPSON RC+ 5.0 Ver.5.2.0 SP3 or greater is required.)
　Windows Vista Business Service Pack 2
　　　(EPSON RC+ 5.0 Ver.5.3.1 or greater is required.)
　Windows 7 Professional
　　　(EPSON RC+ 5.0 Ver.5.3.4 or greater is required.)

Installing EPSON RC+ 5.0 Software

The EPSON RC+ 5.0 software needs to be installed on your development PC.

If your application uses the EPSON RC+ 5.0 Operator Window you will also need to install it on the production PC.

To install EPSON RC+ 5.0:

1. Insert the EPSON RC+ 5.0 Setup CD in the CD drive.

2. If EPSON RC+ 5.0 has previously been installed, you will be asked to uninstall the previous version, otherwise go to step 3.
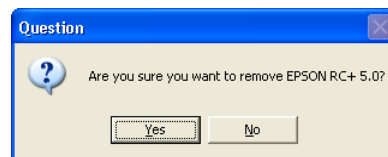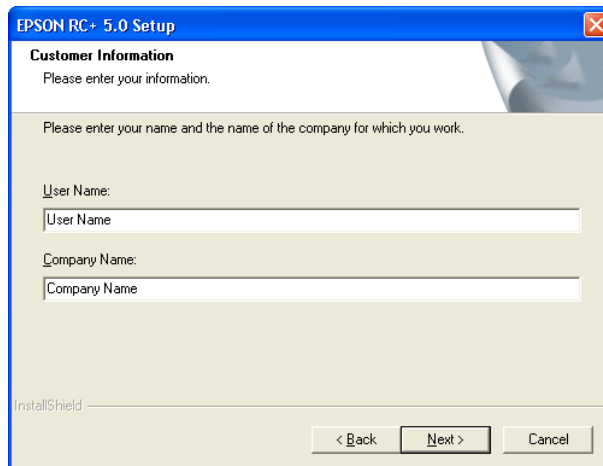

After uninstalling, you will be prompted to restart the computer. Start the setup program again by double-clicking the installer CD icon in My Computer, or by re-inserting the CD.
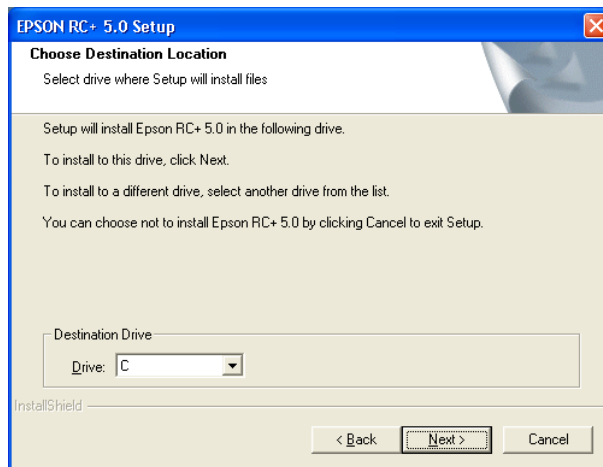
3. The welcome dialog will be displayed as shown below.
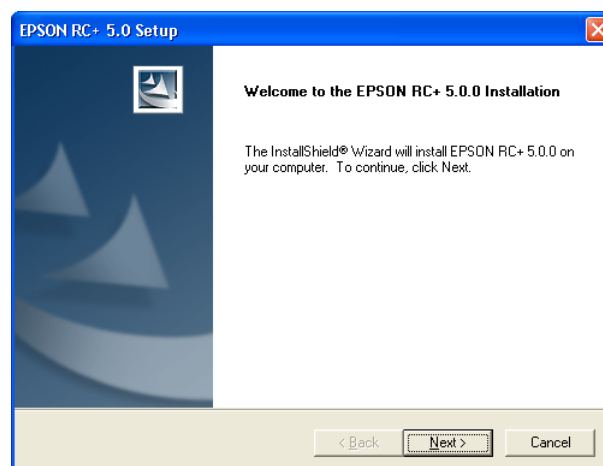
4. Click the **Next** button to continue.

5. Enter your user name and company name, then click Next.

6. Select the drive where you want to install EPSON RC+ 5.0 and click **Next**.

NOTE
☞ The installation directory is called EpsonRC50 and cannot be changed.

You will be prompted to confirm installation.
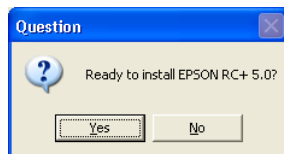
Click **Yes** to continue.

7. If required, Windows Installer and Microsoft .NET Framework 3.5 are installed on your system. This may take several minutes.

8. Adobe Reader needs to be installed on your PC in order to view the EPSON RC+ 5.0 manuals. If the installer cannot find Adobe Reader on your system, it will be installed at this time. Follow the instructions in the Adobe installer. Do not restart the system after the Adobe Reader installation has completed.

9. After the installation has completed, you may be prompted to restart your computer.

10. The EPSON RC+ 5.0 software installation is now completed.

Appendix B: Project Import Automatic Processing

Project for EPSON RC+ 3.* / 4.*

When projects created in EPSON RC+ 3.* / 4.* are imported, the following processes are executed automatically:

- User program conversion
- Point file conversion
- I/O label file conversion
- User error label file conversion

User Program Conversion

The tables below show the syntax conversions from EPSON RC+ 3.* / 4.* to EPSON RC+ 5.0.

Project Type	EPSON RC+ 4.*	EPSON RC+ 5.0
	While	Do While
Syntax	Wend	Loop
	Trap...Call	Trap...Xqt

Project Type	EPSON RC+ 3.*	EPSON RC+ 5.0
	While	Do While
	Wend	Loop
	Trap...Call	Trap...Xqt
	On $, Off $	MemOn, MemOff
	Sw($	MemSw(
	Sw $(	MemSw(
Syntax	In($	MemIn(
	In $(	MemIn(
	Out $	MemOut
	Xqt !	Xqt
	Quit !	Quit
	Resume !	Resume
	Halt !	Halt

Point File Conversion

For EPSON RC+ 3.*, the EPSON RC+ 5.0 .PTS files are generated automatically from the .PNT files and corresponding .DEF files.

Project Type	EPSON RC+ 3.*	EPSON RC+ 5.0
Point File	*.PNT file (Point file)	*.PTS
	*.DEF file (Point label file)	

For EPSON RC+ 4.*, the EPSON RC+ 5.0 .PTS files are generated automatically from the .PNT files.

Project Type	EPSON RC+ 4.*	EPSON RC+ 5.0
Point File	*.PNT file (Point file)	*.PTS

I/O Label File Conversion

IOLabels.dat is generated automatically from the following three files.

Project Type	EPSON RC+ 3.* / 4.*	EPSON RC+ 5.0
I/O Label File	inplabel.txt	IOLabels.dat
	outlabel.txt	
	memlabel.txt	

User Error Label File Conversion

Files are changed automatically as the user error numbers are changed.

Project Type	EPSON RC+ 3.* / 4.*	EPSON RC+ 5.0
User Error Label	30000 to 30999	8000 to 8999
User Error Label File	UserErrors.txt	UserErrors.dat

Project Import for SPEL for Windows 2.*

When projects created in SPEL for Windows 2.* are imported, the following processes are executed automatically.

- User program conversion
- Point file conversion
- I/O label file conversion
- Global Preserve variable table conversion
- Global variable conversion
- Local variable conversion

User Program Conversion

The table below shows the syntax conversions from SPEL for Windows 2.* to EPSON RC+ 5.0.

Project Type	SPEL for Windows 2.*	EPSON RC+ 5.0
Syntax	While	Do While
	Wend	Loop
	Trap n...Call	Trap n...Xqt
	On $, Off $	MemOn, MemOff
	Sw($	MemSw(
	Sw $(	MemSw(
	In($	MemIn(
	In $(	MemIn(
	Out $	MemOut
	Xqt !	Xqt
	Quit !	Quit
	Resume !	Resume
	Halt !	Halt
	Palet	Pallet
	Print"	Print "
	Date$(0)	Date$
	Time$(0)	Time$
	JS(0)	JS
	TW(0)	TW
	ZeroFlg(0)	ZeroFlg
	Entry	Global
	Config statement	SetCom statement
	Cooked	Line deleted
	SetRaw	Line deleted
	SelRB	Line deleted
	SelRB1	Line deleted
	Extern	Line deleted
	End	Quit All
	GetDate d$	d$ = Date$
	GetTime t$	t$ = Time$

Point File Conversion

EPSON RC+ 5.0 .PTS files are generated automatically from the .PNT files and corresponding .DEF files.

Project Type	SPEL for Windows 2.*	EPSON RC+ 5.0
Point File	*.PNT file (Point file) *.DEF file (Point label)	*.PTS

I/O Label File Conversion

Converts the I/O labels automatically.

Project Type	SPEL for Windows 2.*	EPSON RC+ 5.0
I/O Label File	*ProjectName*.IOL	IOLabels.dat

Global Preserve Variable Table Conversion

Backup variable definitions created in the SPEL for Windows 2.* Project Menu are converted into Global Preserve declaration statements in the first program file.

(Example)

If the SPEL for Windows 2.* project defines an integer backup variable called

"s_iValue", the following statement is generated in the first program of the project.

Global Preserve Integer s_iValue

Global Variable Conversion

Global variables (Entry / Extern) in SPEL for Windows 2.* projects are converted to Global variables in EPSON RC+ 5.0.

Project Type	SPEL for Windows 2.*	EPSON RC+ 5.0
Global Variable (Command)	Entry / Extern command	Global command

Local Variable Conversion

Local variables in SPEL for Windows 2.* functions can be used throughout the entire file in which they are declared. These variables are converted to module variables or local variables in EPSON RC+ 5.0, depending on their scope.

If the variable is used in only one function, it is converted to a local variable in that function.

If the variable is used in more than one function, it is converted to a module variable.