EPSON

Robot Controller Option
SRCI

Original instructions

© Seiko Epson Corporation 2026

Rev.2
ENM269C9285F

Table of Contents

1. FOREWORD 5

1.1 Foreword 6

1.2 Trademarks 6

1.3 Notation 6

1.4 Terms of Use 6

1.5 Manufacturer 6

1.6 Contact Information 7

1.7 Getting Started 7

1.7.1 Safety precautions 7

1.7.2 Meaning of symbols 8

1.7.3 Security measures for network connections 8

1.7.4 The installation folder for Epson RC+ 8.0 8

2. Overview of the Optional Functions 9

2.1 Overview of the Controller Function 10

2.2 Required Equipment and Software 12

3. Connecting the PLC and the Controller 13

3.1 Hardware Connection Configuration 14

3.2 Software Connection Settings 15

3.2.1 Controller settings 15

3.2.2 Setting a Siemens PLC 16

3.2.3 Example of Robot Control by SRCI 19

4. Setting the Command Input/Output Parameters 27

4.1 Setting the Command Input/Output Parameters 28

4.1.1 ChangeSpeedOverride 28

4.1.2 CreateServerLog 28

4.1.3 EnableRobot 28

4.1.4 GroupContinue 28

4.1.5 GroupInterrupt 28

4.1.6 GroupJog 29

4.1.7 GroupReset 29

4.1.8 GroupStop 29

4.1.9 MoveAxesAbsolute 29
4.1.10 MoveDirectAbsolute 30
4.1.11 MoveLinearAbsolute 31
4.1.12 ReadActualPosition 31
4.1.13 ReadActualPositionCyclic 32
4.1.14 ReadFrameData 32
4.1.15 ReadLoadData 32
4.1.16 ReadMessages 33
4.1.17 ReadRobotData 33
4.1.18 ReadRobotDefaultDynamics 33
4.1.19 ReadRobotReferenceDynamics 33
4.1.20 ReadRobotSWLimit 33
4.1.21 ReadServerLog 33
4.1.22 ReadToolData 33
4.1.23 ReturnToPrimary 34
4.1.24 RobotTask 34
4.1.25 SetOperationMode 35
4.1.26 SetSequence 35
4.1.27 WriteFrameData 35
4.1.28 WriteLoadData 36
4.1.29 WriteRobotDefaultDynamics 36
4.1.30 WriteRobotReferenceDynamics 36
4.1.31 WriteToolData 36

5. Alarm message code 37

5.1 Alarm message code 38

6. Mode LEDs and 7-segment LED 39

6.1 Mode LEDs and 7-segment LED 40

7. Method of Setting the Safety Board 42

7.1 Method of Setting the Safety Board 43
7.1.1 Setting Safety Position Monitoring 43
7.1.2 Setting the Software Axis Restrictions 44
7.1.3 Safety Limited Speed of T1 and T2 External Mode 45
7.1.4 How to Recover from Stopping by SLP 45

8. Other Restrictions 47

8.1 Restrictions on the Controller Function 48

1. FOREWORD

1.1 Foreword

Thank you for purchasing our robot system.

This manual contains information necessary to correctly use the SRCI.

Before using the system, read this manual and related manuals to ensure correct use.

After reading the manual, keep it in a place where it can be easily accessed at any time, and reread it if you have any questions.

Epson conducts rigorous testing and inspections to ensure that the performance of our robot systems meets our company standards. Note that the basic performance of the product will not be achieved if the robot system is used outside the conditions of use described in the manual.

This manual describes dangers and problems foreseen by Epson. To use our robot system safely and correctly, be sure to observe the safety precautions described in this manual.

1.2 Trademarks

Microsoft, Windows, and the Windows logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

TIA Portal is a registered trademark of Siemens Corporation.

SIMATIC is a registered trademark of Siemens Corporation.

Other company names, brand names, and product names are either registered trademarks or trademarks of their respective companies.

1.3 Notation

Microsoft® Windows® 10 operating system
Microsoft® Windows® 11 operating system
In this manual, the above operating systems are referred to as Windows 10 and Windows 11, respectively. Windows 10 and Windows 11 are sometimes collectively referred to as Windows.

1.4 Terms of Use

No part of this instruction manual may be reproduced or reprinted in any form without express written permission.
The information in this document is subject to change without notice.
Please contact us if you find any errors in this document or if you have any questions about the information in this document.

1.5 Manufacturer

SEIKO EPSON CORPORATION

1.6 Contact Information

Contact information details are listed in the "Supplier" section in the following manual.
Note that the contact information may vary depending on your region.
"Safety Manual - Contact Information"

The Safety Manual is also available at the following site.
URL: https://download.epson.biz/robots/

1.7 Getting Started

Before using this manual, be sure that you understand the following information.

1.7.1 Safety precautions

The SRCI (Standard Robot Command Interface) described in this manual is a product that is used by connecting to an Epson controller for industrial use. The transportation and installation of the robot and related equipment shall be carried out by qualified personnel. Also, ensure that the laws and regulations of each country are followed.

Before using the system, thoroughly read this manual and related manuals to ensure correct use.

After reading the manual, keep it in a place where it can be easily accessed at any time, and reread it if you have any questions.

1.7.2 Meaning of symbols

The following symbols are used in this manual to indicate important safety information. Be sure to read the descriptions shown with each symbol.

⚠ WARNING

This symbol indicates an imminently hazardous situation which, if operation is not performed properly, will result in death or serious injury.

⚠ WARNING

This symbol indicates a potentially hazardous situation which, if operation is not performed properly, could result in an injury due to electric shock.

⚠ CAUTION

This symbol indicates a potentially hazardous situation which, if operation is not performed properly, may result in a minor or moderate injury or in property damage only.

1.7.3 Security measures for network connections

The network connection function (Ethernet) installed in this product is expected to be used on a local network such as a factory LAN. Do not perform external connections to the Internet.

We request you to take security measures such as anti-virus protection due to network connections.

1.7.4 The installation folder for Epson RC+ 8.0

You can change the path for the installation folder for Epson RC+ 8.0 to any location. This manual assumes that Epson RC+ 8.0 is installed in `C:¥EpsonRC80`.

2. Overview of the Optional Functions

2.1 Overview of the Controller Function

This option enables control of the Epson robot controller from a PLC that supports the SRCI (Standard Robot Command Interface). The versions and operating environments supported by this option are described below.

Overview of supported functions

Item	Support details
SRCI specifications	Conforms to version 1.5
Supported SRCI functions	- All commands belonging to Core profile, and the SetOperationMode command - The trigger function and multi-robot function are not supported
Supported controllers	RC800 Series
Controller firmware version	Ver. 8.2.0.0 or later
Supported RC+ version	RC+ Ver. 8.1.4.0 or later
Supported Fieldbus	PROFINET

List of supported SRCI commands

Command type value	Command name	Command function overview
2000	ChangeSpeedOverride	Changes the speed override for the move command.
9003	CreateServerLog	Creates a snapshot of the event logging buffer inside the controller, the various system states, and the present command contents.
1000	EnableRobot	Enables the drive unit of the robot by setting the robot power status to "Enabled".
2001	GroupContinue	Continues the processing of an interrupted active sequence.
2002	GroupInterrupt	Interrupts the command in the active sequence.
2100	GroupJog	Performs manual jog operation of the TCP or robot joints.
1001	GroupReset	Deletes messages from the message buffer, and resets the controller state.
2003	GroupStop	Interrupts all commands in the active sequence.
2101	MoveAxesAbsolute	Instructs movement to the absolute position specified at the joint position.
2102	MoveDirectAbsolute	Moves the TCP position using PTP motion.
2103	MoveLinearAbsolute	Moves the TCP position using CP motion.
5100	ReadActualPosition	Reads the present position of the TCP.
-	ReadActualPositionCyclic	Periodically reads the present position of the robot.
5102	ReadFrameData	Reads the configuration of the defined user coordinate system.
5103	ReadLoadData	Reads the configuration of the defined load data.
9001	ReadMessage	Reads the errors, warnings, and information messages from the message buffer of the controller.

Command type value	Command name	Command function overview
9002	ReadRobotData	Reads specific parameters from the controller.
5202	ReadRobotDefaultDynamics	Reads the default value of the speed, acceleration, and deceleration during robot movement.
5105	ReadRobotReferenceDynamics	Reads the reference values for speed, acceleration, deceleration, and other parameters during robot movement.
5106	ReadRobotSWLimit	Reads the software limit value of the robot axis.
9004	ReadServerLog	Reads the log from the controller.
5107	ReadToolData	Reads the settings of the defined tool.
2104	ReturnToPrimary	Moves the TCP up to the interrupt position in the primary sequence.
-	RobotTask	Processes multiple mechanisms necessary for the SRCI function to operate.
1003	SetOperationMode	Switches the operation mode.
1004	SetSequence	Changes the active sequence to primary or secondary.
5200	WriteFrameData	Writes the configuration data for the user coordinate system.
5201	WriteLoadData	Writes the load data.
5202	WriteRobotDefaultDynamics	Writes the default value of the speed, acceleration, and deceleration during robot movement.
5203	WriteRobotReferenceDynamics	Writes the reference value of the speed, acceleration, and deceleration during robot movement.
5205	WriteToolData	Writes the configuration data of the defined tool.

2.2 Required Equipment and Software

If this optional function is used to perform control through the SRCI, the equipment and software described below are required.

Equipment and software related to the controller

Category	Item	Notes
Controller	RC800 Series	Firmware version 8.2.0.0 or later
	PROFINET slave module	-
	D-Sub/TP connector conversion cable	Necessary depending on the device configuration See: **Hardware Connection Configuration**
	Optional license of this function	-
Controller client software	RC+ version 8.1.4.0 or later	Mainly used for the initial setup of the controller

Equipment and software related to the Siemens PLC

Category	Item	Notes
PLC	SRCI-compliant PLCs - S7-1500 controller version V2.8 or later - ST-1500V controller version V3.1 or later - ST-1500 Software controller version V20.8 or later	-
PLC client software	TIA Portal V19 or later	-
PLC library	SIMATIC Robot Library V1.2 or later	-
Safety function module	Output module of the ET200 Series and the F-DQ 8x24V	Necessary depending on the device configuration See: **Hardware Connection Configuration**

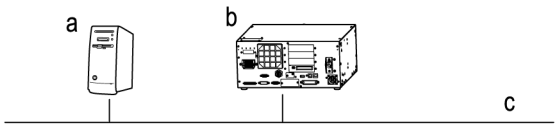
- Contact Siemens for more information on supported devices and software.

3. Connecting the PLC and the Controller

3.1 Hardware Connection Configuration

This section describes the hardware connection configuration of the PLC and controller.

Hardware connection example 1

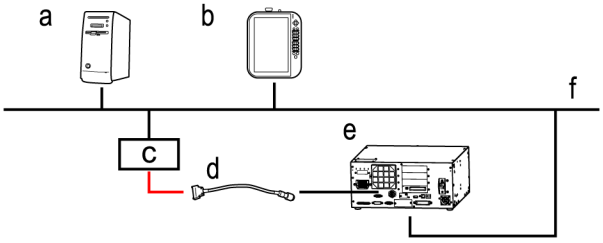


Symbol	Description
a	PLC
b	Controller
c	Fieldbus

Configure the hardware connection of the PLC and controller as shown in the connection example above.

- Connect the PLC and, if necessary, an HMI (Human Machine Interface) to the fieldbus.
  For details on how to connect the PLC and HMI to the fieldbus, refer to the manual of each device.
- Connect the controller to the fieldbus. Attach the fieldbus module to the controller and insert the cable into the fieldbus module.

Hardware connection example 2



Symbol	Description
a	PLC
b	HMI
c	Safety function module
d	D-Sub/TP connector conversion cable
e	Controller
f	Fieldbus

When inputting the emergency stop signal and enable switch status sent from an HMI or similar device via a safety fieldbus to the controller, configure the system as shown in the connection example above.

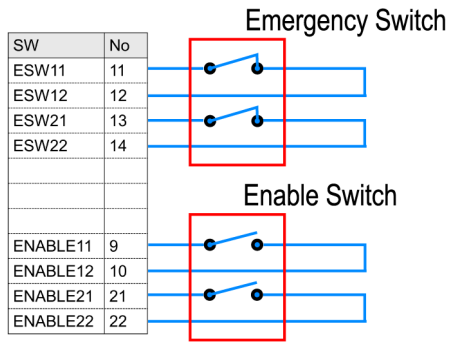
- Connect a safety function module (a device that interprets the safety fieldbus communication and converts it to a signal) to the fieldbus.

- Connect the system so that the control results generated by the signals output from the safety function module are input to the controller's TP connector via a D-Sub/TP connector conversion cable.

⚠ NOTE

Prepare the cable that connects the safety function module to the D-Sub 25-pin side of the conversion cable separately (the portion indicated by the red line in "Hardware connection example 2").

The pin arrangement of the D-Sub 25-pin connector of the D-Sub/TP conversion cable required to create a device that inputs the emergency stop and Enable switch status signals output from the safety function module to the controller is shown below.



SW	No
ESW11	11
ESW12	12
ESW21	13
ESW22	14
ENABLE11	9
ENABLE12	10
ENABLE21	21
ENABLE22	22

3.2 Software Connection Settings

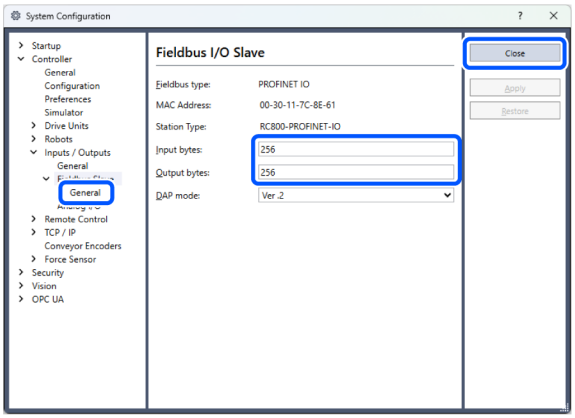
3.2.1 Controller settings

This section describes the procedure for making controller settings to communicate with an SRCI-compliant PLC.
Make the controller settings according to the procedure below.

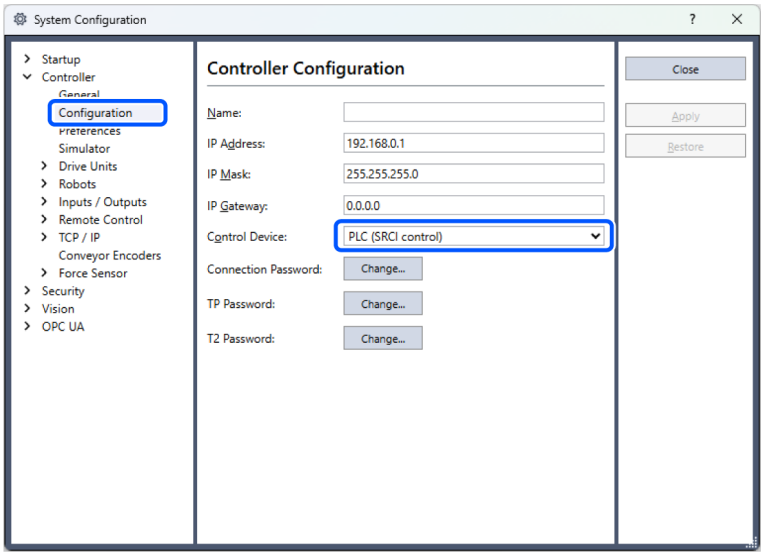
1. Mount the PROFINET slave module and install the software. For details on the mounting method and installation method, refer to the following manual.
   "Robot Controller Option Fieldbus I/O"
2. From RC+, select [Setup] - [System Configuration] to open the [System Configuration] dialog box.
3. Select [Controller] - [Inputs / Outputs] - [Fieldbus Slave] - [General], enter the input bytes and output bytes of the fieldbus, and press the [Apply] button.

⚠ NOTE

This setting is the communication size with the PLC. The recommended value is 256 bytes, but is not limited to this size. However, if you set a size that is too small, you may need to exchange data multiple times to send and receive one command, which may reduce the response speed.



4. Select [Controller] - [Configuration], switch the control device to "PLC (SRCI control)", and then press the [Apply] button.



5. Press [Close] to restart the controller.
6. After restarting, switch the [Connection Status] of RC+ to [Offline], and cut it off from the controller.

3.2.2 Setting a Siemens PLC

This section describes the procedure of making PLC settings when connecting to a Siemens PLC. For the detailed contents, refer to the manual provided by the vendor.

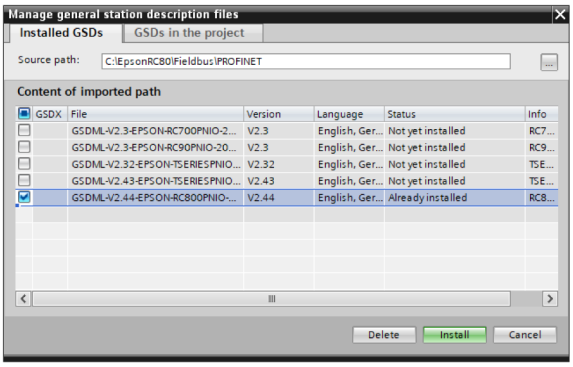
Make the PLC settings according to the procedure below.
We will explain this procedure using the example below.

- Use "TIA Portal V20" as the client software of Siemens PLC.

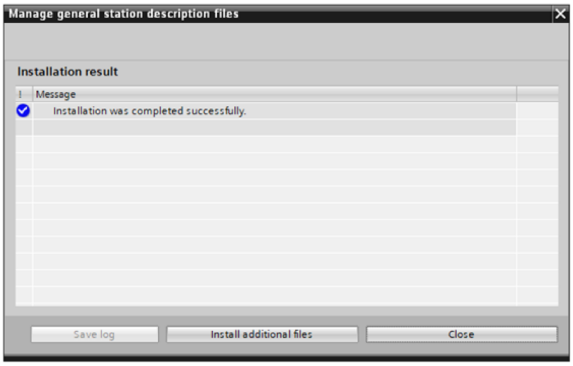
1. Import the basic performance information of the fieldbus module (PROFINET Slave Module).

   i. Select [Options] - [Manage general station description files(GSD)].

   ii. In [Source path:] in the [Manage general station description files] dialog box, specify the Fieldbus\PROFINET folder below the RC+ installation folder. If RC+ is installed with the default settings, the folder will be [C:\EpsonRC80\Fieldbus\PROFINET].



iii. Place a check mark at [GSDML-V2.44-EPSON-RC800PNIO-20240410.xml] in the file list displayed under [Content of imported path], and press the [Install] button.



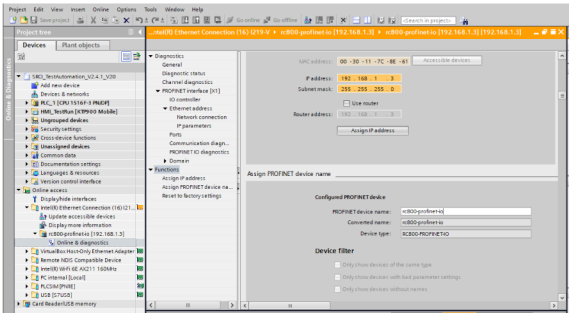
2. Check the name and address of the fieldbus module, and change the settings, if necessary.

i. Turn the controller ON.

ii. Connect the Ethernet port of the PC on which the TIA Portal is installed to the fieldbus module of the controller using an Ethernet cable.

iii. On the TIA Portal, open [Online access] under [Project tree], and then open the tree of the device name indicating the PC Ethernet.

iv. Double-click [Update accessible devices].
A tree containing the PROFINET device name and IP address (Example: [rc800-profinet-io [192.168.1.3]]) will be displayed.



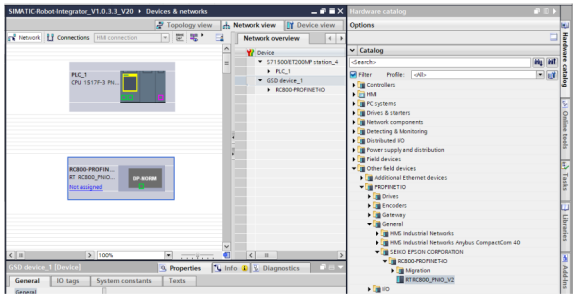
v. Set the PROFINET device name and IP address according to your network configuration.

vi. Once the setting is complete, pull out the Ethernet cable from the PC.

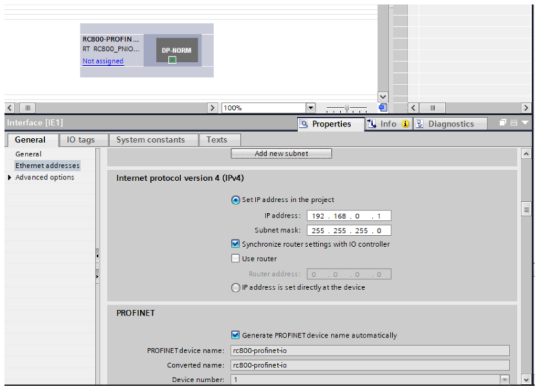
3. Make the settings for communication between the PLC and the fieldbus module.

i. Under [Project tree], open [Devices & network].

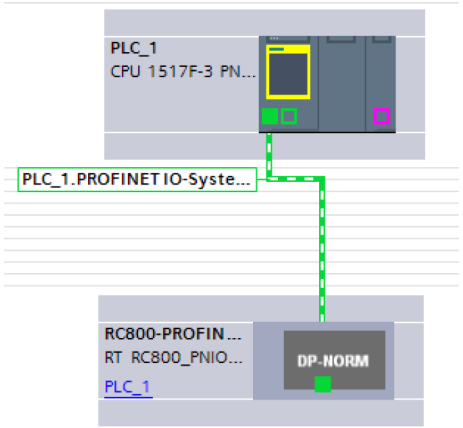
ii. Open the tree of [Hardware catalog], and from [Other field devices] - [PROFINET-IO] - [General] - [SEIKO EPSON CORPORATION] - [RC800-PROFINET-IO], drag & drop [RT RC800_PNIO_V2] to the center of the screen.
The device added here is the PROFINET interface of the controller.



iii. Select the interface part (green square in the figure) of [RC800-PROFINET-IO] dropped in ii., and set the PROFINET device name and IP address set in step ②.



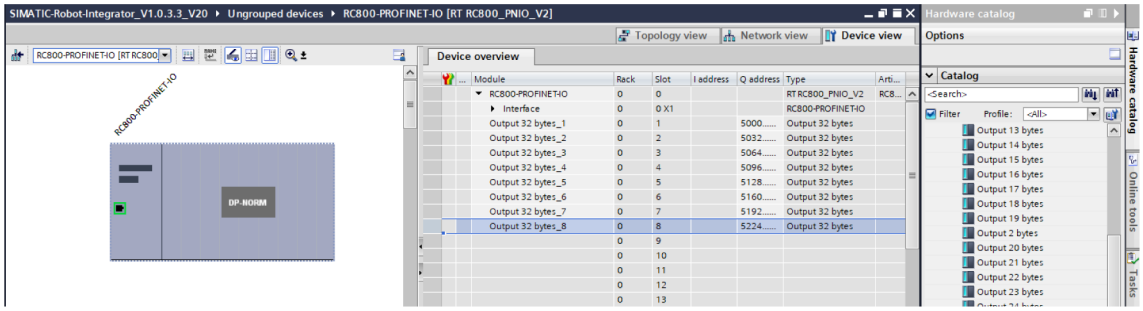
iv. Connect the PLC and PROFINET interface.
Drag the interface part of the PLC and drop it to the interface part of [RC800-PROFINET-IO].
Set the PROFINET device name and IP address of the connected PLC's PROFINET interface so that it can communicate with the interface configured in iii.



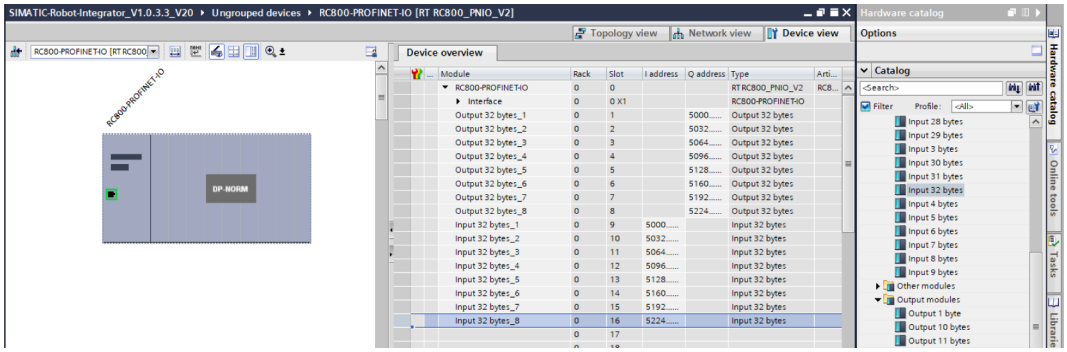
4. Set the communication width of the fieldbus, and then set the I/O address to be used for communication.

i. Double-click the figure of [RC800-PROFINET-IO] to open [Device view].

ii. Under [Hardware catalog], open [Module] - [Output modules], and then drag & drop the input module to [Device overview].
If the output bytes of the controller's fieldbus slave are 256 bytes, you can configure a 256-byte width input by dragging and dropping one [Output 32 bytes] into [Device overview], changing the value of [Q address] to the desired address, and copying and pasting the [Output 32 bytes_1] record seven times.



iii. Similar to ii., under [Hardware catalog], open [Module] - [Input modules], and then drag & drop the output module to [Device overview].
If the input sizes of the controller's fieldbus slave are 256 bytes, you can configure a 256-byte width input by dragging and dropping one [Input 32 bytes] into [Device overview], changing the value of [I address] to the desired address, and copying and pasting the [Input 32 bytes_1] record seven times.



5. Set the data block in which the I/O address specified in 4. is set to the RobotInData and RobotOutData of the RobotTask function block.
See below for a setting example.
Example of Robot Control by SRCI

3.2.3 Example of Robot Control by SRCI

An example of performing robot control from a Siemens PLC is described below.

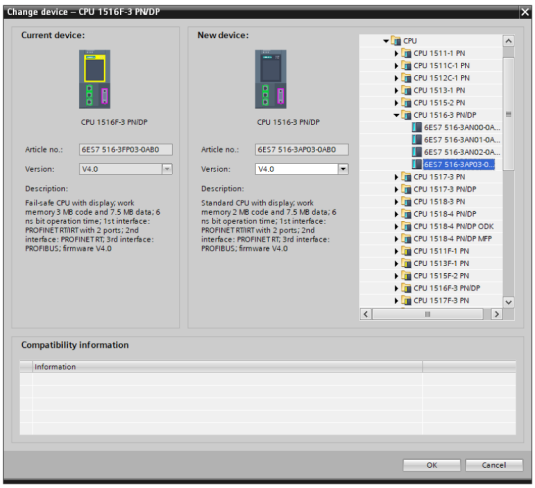
- Use "TIA Portal V20" as the client software of Siemens PLC.
- Create a new project on the TIA Portal from the Archive file in "SIMATIC Robot Library V1.2" provided by Siemens.
- Perform robot control through SCL (Structured Control Language).

1. Change the setting of PLC model number to your PLC model number.

   i. Under [Project tree], double-click [Device & networks], and then double-click [PLC_1] at the center of the screen.

   ii. Right-click the PLC shown in [Device view], and select [Change device].

   iii. Select your PLC model number, and press the [OK] button.



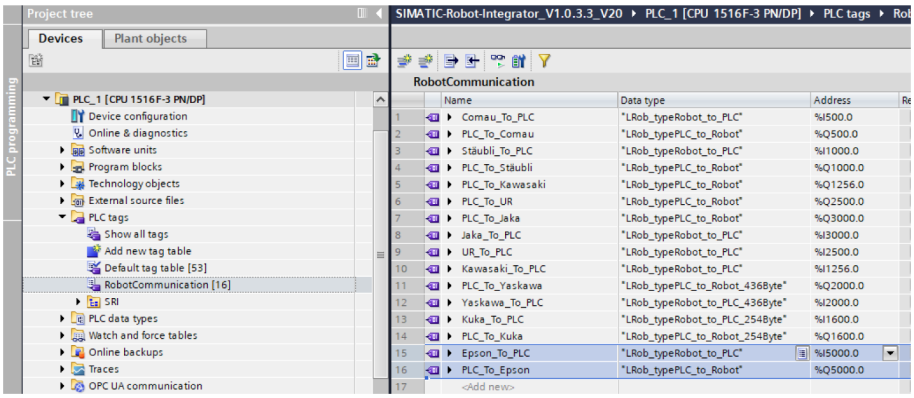
2. Make the settings for communication between the PLC and controller.

   i. See below and make the settings for communication between the PLC and controller.
   Setting a Siemens PLC

   ii. Under [Project tree], select [PLC_1] - [PLC_tags], and then double-click and open [RobotCommunication].

   iii. Copy and paste [Xxx_To_PLC] and [PLC_To_Xxx] defined for other companies, and then change the name.

   iv. Change the [Address] setting to the value set in the input/output module of [RC800-PROFINET-IO].

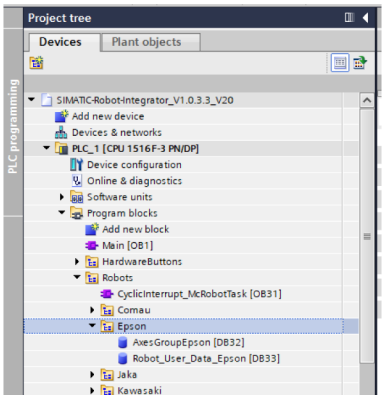


3. Set the RobotTask function block for Epson robot control.

   i. Under [Project tree], create the [Epson] group below [PLC_1] - [Program blocks] - [Robots].

   ii. Under [Project tree], copy the following data blocks set in the group for other robot vendors below [PLC_1] - [Program blocks] - [Robots], paste them to the group created in i., and change the names:

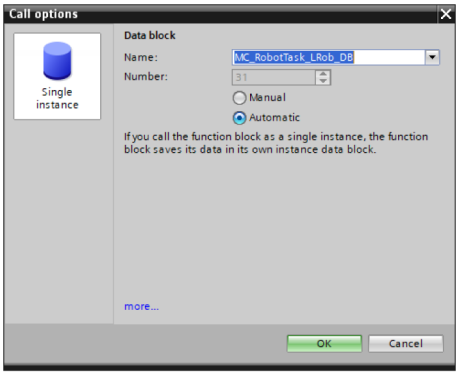
   - AxesGroupXxx
   - Robot_User_Data_Xxx



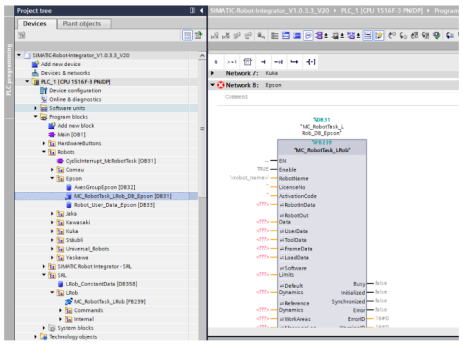
iii. Right-click [PLC_1] and select [Compile] - [Hardware and software (only changes)]. A warning is displayed regarding duplication of the data block numbers copied in ii. Press [OK] to resolve the duplicate numbers.

iv. From [PLC_1] - [Program blocks] - [Robots], double-click [Cyclicinterrupt_McRobotTask].

v. Open [Network 8], and from [PLC_1] - [Program blocks] - [SRL] - [LRob], drag & drop [MC_RobotTask_LRob], and press the [OK] button in the [Call options] dialog box.



vi. The data block configured in step v. appears under [PLC_1] - [Program blocks] - [Robots]. Move it to the [Epson] group and rename it.



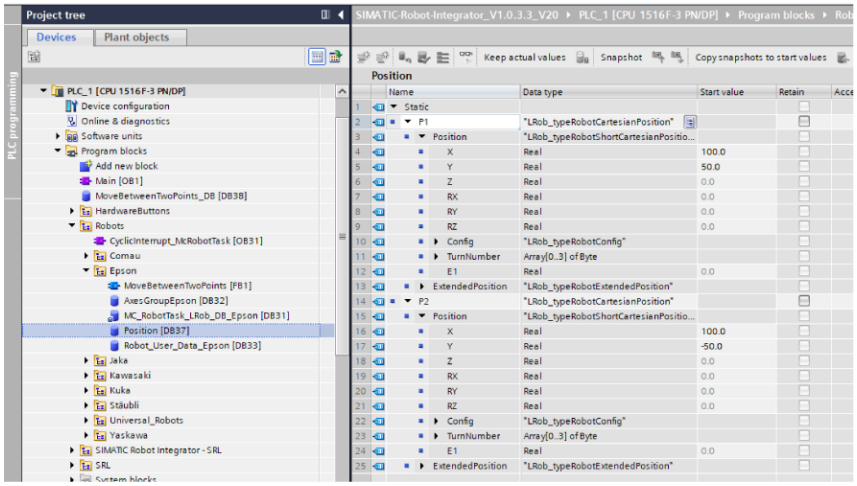
vii. Set each parameter of the RobotTask function block.

Parameter name	Data name
RobotInData	“Epson_To_PLC”.Arm1
RobotOutData	“PLC_To_Epson”.Arm1
UserData	“Robot_User_Data_Epson”.UserData
ToolData	“Robot_User_Data_Epson”.ToolData

Parameter name	Data name
FrameData	“Robot_User_Data_Epson”.FrameData
LoadData	“Robot_User_Data_Epson”.LoadData
SoftwareLimit	“Robot_User_Data_Epson”.SwLimits
DefaultDynamics	“Robot_User_Data_Epson”.DefaultDynamics
ReferenceDynamics	“Robot_User_Data_Epson”.ReferenceDynamics
WorkArea	“Robot_User_Data_Epson”.WorkAreaData
MessageLog	“Robot_User_Data_Epson”.MessageLog
SystemLog	“Robot_User_Data_Epson”.SystemLog
ConstantData	“Lrob_ConstantData”
AxesGroup	“AxesGroupEpson”

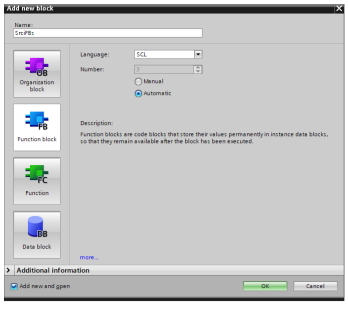
4. Write the SCL program for robot control.

   i. From [PLC_1] - [Program blocks] - [Robots], right-click [Epson], select [Add new block], set [Type] to Global DB, and create a data block. In the data block that is created, define two instances of data having the Data type as Lrob_typeRobotCartesianPosition, as shown in the figure below.

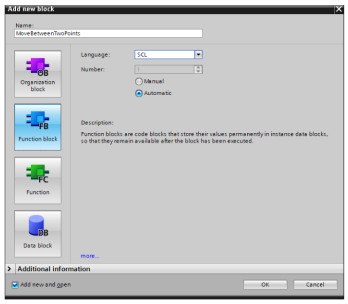


   ii. From [PLC_1] - [Program blocks] - [Robots], right-click [Epson], select [Add new block], set [Language] to SCL, and create the two function blocks described below.

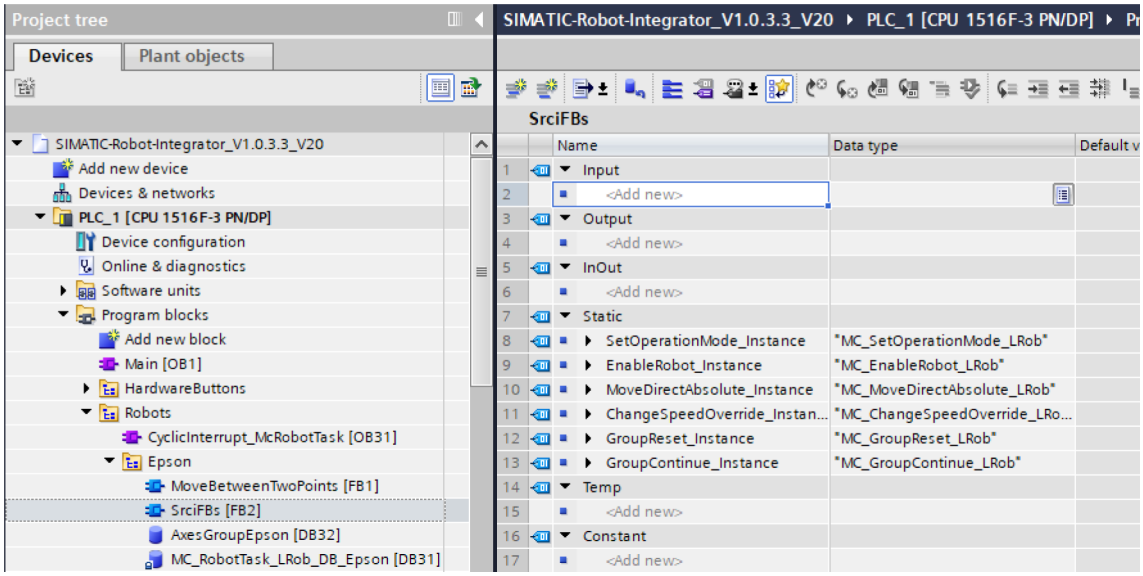
   - SrciFBs



- MoveBetweenTwoPoints



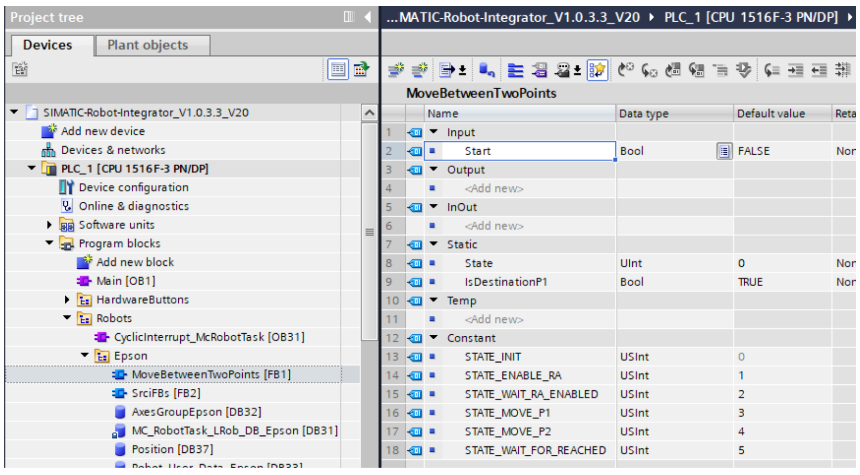
iii. In the function blocks created in ii., define the data as shown in the figure below.



iv. In the function block created in ii., write the following source code.

```
#SetOperationMode_Instance(AxesGroup := "AxesGroupEpson");
#EnableRobot_Instance(AxesGroup := "AxesGroupEpson");
#MoveDirectAbsolute_Instance(AxesGroup := "AxesGroupEpson");
#ChangeSpeedOverride_Instance(AxesGroup := "AxesGroupEpson");
#GroupReset_Instance(AxesGroup := "AxesGroupEpson");
#GroupContinue_Instance(AxesGroup := "AxesGroupEpson");
```

v. In the function blocks created in ii., define the data as shown in the figure below.

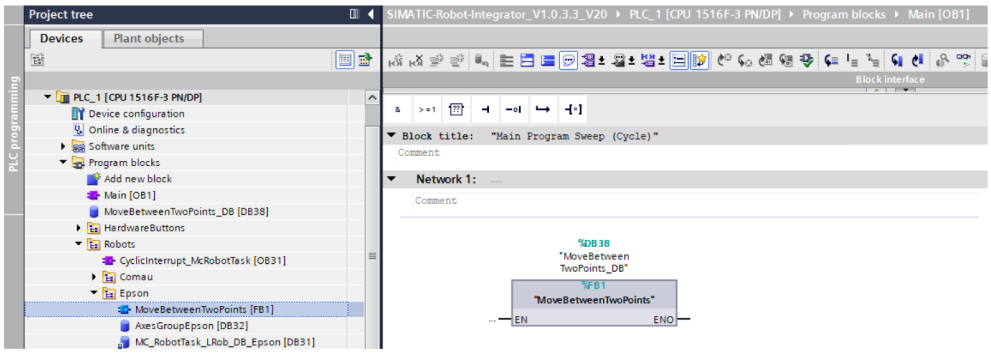


vi. In the function block created in ii., write the following source code.

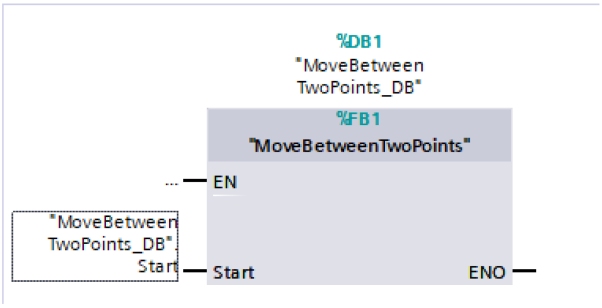
```
CASE #State OF
#STATE_INIT:
    IF #Start THEN
        "SrciFBs_DB".ChangeSpeedOverride_Instance.Override := 10.0;
        "SrciFBs_DB".MoveDirectAbsolute_Instance.VelocityRate := 100.0;
        "SrciFBs_DB".MoveDirectAbsolute_Instance.AccelerationRate := 100.0;
        "SrciFBs_DB".MoveDirectAbsolute_Instance.DecelerationRate := 100.0;
        "SrciFBs_DB".MoveDirectAbsolute_Instance.JerkRate := -1.0;

        "SrciFBs_DB".SetOperationMode_Instance.OperationMode := 4;
        "SrciFBs_DB".SetOperationMode_Instance.Execute := TRUE;

        "SrciFBs_DB".GroupReset_Instance.Execute := TRUE;
        #State := #STATE_ENABLE_RA;
    END_IF;

#STATE_ENABLE_RA:
    IF "SrciFBs_DB".SetOperationMode_Instance.Done THEN
        "SrciFBs_DB".EnableRobot_Instance.Enable := TRUE;
        #State := #STATE_WAIT_RA_ENABLED;
    END_IF;

#STATE_WAIT_RA_ENABLED:
    IF "SrciFBs_DB".EnableRobot_Instance.Enabled THEN
        "SrciFBs_DB".ChangeSpeedOverride_Instance.Execute := TRUE;
        "SrciFBs_DB".GroupContinue_Instance.Execute := TRUE;
        #State := #STATE_MOVE_P1;
    END_IF;

#STATE_MOVE_P1:
    "SrciFBs_DB".MoveDirectAbsolute_Instance.Position := "Position".P1;
    "SrciFBs_DB".MoveDirectAbsolute_Instance.Execute := TRUE;
    #State := #STATE_WAIT_FOR_REACHED;

#STATE_MOVE_P2:
    "SrciFBs_DB".MoveDirectAbsolute_Instance.Position := "Position".P2;
    "SrciFBs_DB".MoveDirectAbsolute_Instance.Execute := TRUE;
    #State := #STATE_WAIT_FOR_REACHED;

#STATE_WAIT_FOR_REACHED:
    IF "SrciFBs_DB".MoveDirectAbsolute_Instance.Done THEN
        "SrciFBs_DB".MoveDirectAbsolute_Instance.Execute := FALSE;
        IF #IsDestinationP1 THEN
            #State := #STATE_MOVE_P2;
        ELSE
            #State := #STATE_MOVE_P1;
        END_IF;
        #IsDestinationP1 := NOT #IsDestinationP1;
    END_IF;

    IF NOT #Start THEN
        "SrciFBs_DB".SetOperationMode_Instance.Execute := FALSE;
```

```
        "SrciFBs_DB".EnableRobot_Instance.Enable := FALSE;
        "SrciFBs_DB".ChangeSpeedOverride_Instance.Execute := FALSE;
        "SrciFBs_DB".GroupReset_Instance.Execute := FALSE;
        "SrciFBs_DB".GroupContinue_Instance.Execute := FALSE;
        #State := #STATE_INIT;
    END_IF;
END_CASE;
```

vii. From [PLC_1] - [Program blocks], double-click and open [Main], drag & drop the function blocks that were created to [Network 1], and then press the [OK] button in the dialog box that is displayed.



viii. From [PLC_1] - [Program blocks], change the "Start" parameter of the function block "MoveBetweenTwoPoints" set in [Main] to ""MoveBetweenTwoPoint_DB".Start".



7. Operate the robot using the project that is created.

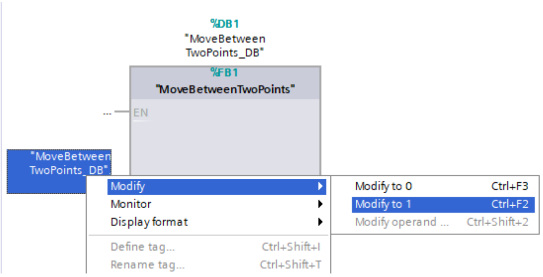
i. Right-click [PLC_1] and select [Compile] - [Hardware and software (only changes)].

ii. Right-click [PLC_1] while the PC and the PLC are connected, select [Download to device] - [Hardware and software (only changes)], and update the hardware settings and software settings of the PLC unit.

iii. Press either the [Run] button on the PLC unit or the [Start CPU] button on the TIA Portal to start the program on the PLC.

iv. From [PLC_1] - [Program blocks], open [Main], and then press the [Monitoring on/off] button to enable monitoring.

v. Right-click "Start" for the "MoveBetweenTwoPoints" function block, and then select [Modify] - [Modify to 1]. The state transition of the [MoveBetweenTwoPoints] function block starts, and the robot operates.



vi. Right-click "Start" for the "MoveBetweenTwoPoints" function block, and then select [Modify] - [Modify to 0] to stop the robot operation.

4. Setting the Command Input/Output Parameters

4.1 Setting the Command Input/Output Parameters

This section describes the input/output parameters with restrictions for the commands supported by this option.
For the detailed specifications of each command operation, see the manual provided by the PLC vendor.

4.1.1 ChangeSpeedOverride

There are no restrictions on the input/output parameters of this command.

4.1.2 CreateServerLog

Input parameters

Parameter name	Restriction
ProcessingMode	The following values are supported: 0：Buffered 1：Aborting 2：Parallel The trigger function is not supported.
ListenerID	Since the trigger function is not supported, specify zero in ListenerID.

Output parameters

Parameter name	Restriction
OriginID	Since the trigger function is not supported, zero is output at all times.
InvocationCounter	Same as above

4.1.3 EnableRobot

Input parameters

Parameter name	Restriction
StepMode	The following parameters are supported: - 0：Deactive - 2：Exact stop

There are no restrictions on the output parameters of this command.

4.1.4 GroupContinue

There are no restrictions on the input/output parameters of this command.

4.1.5 GroupInterrupt

There are no restrictions on the input/output parameters of this command.

4.1.6 GroupJog

Input parameters

Parameter name	Restriction
Override	Only 10.0 or 75.0 can be specified.
IncrementalTranslation	If the J3 axis of a SCARA robot is operated by as much as the specified amount in Mode=2 (JogAxes), the value of IncrementalTranslation is used as the amount of movement of J3. Since the J3 axis of a SCARA robot is a ball screw spline, IncrementalTranslation, which specifies the distance, is used instead of IncrementalRotation, which specifies the angle.
IncrementalRotation	

There are no restrictions on the output parameters of this command.

4.1.7 GroupReset

There are no restrictions on the input/output parameters of this command.

4.1.8 GroupStop

There are no restrictions on the input/output parameters of this command.

4.1.9 MoveAxesAbsolute

Input parameters

Parameter name	Restriction
JointPosition	Since the external axis is not supported, the specification of E1 to E6 is ignored.
VelocityRate	Values exceeding 100.0 are not supported. Specify within the range of 0.0 to 100.0.
AccelerationRate	Same as above
DecelerationRate	Same as above
JerkRate	Not supported. Specify -1.0.
BlendingMode	The following values are supported: - 0：Exact stop - 11：RampOverlap
BlendingParameter	If 11 (RampOverlap) is specified in BlendingMode, specify the percentage 0.0 to 100.0 of overlap in BlendingParameter[0]. BlendingParameter[1] is not used.
Time	Not supported. Specify zero.
Manipulation	Not supported. Specify FALSE.
EmmiterID	Not supported. Specify zero.

Output parameters

Parameter name	Restriction
Progress	Not supported. Zero is output at all times.
FollowID	Same as above

4.1.10 MoveDirectAbsolute

Input parameters

Parameter name	Restriction
Position	Since an external axis is not supported, the specification of E1 to E6 is ignored. In the case of a SCARA robot, the Z-axis direction of the Frame coordinates and Tool coordinates is inverted (rotated 180° around the X-axis) according to the SRCI specification, so specify 180.0 for RX.
VelocityRate	Values exceeding 100.0 are not supported. Specify within the range of 0.0 to 100.0.
AccelerationRate	Same as above
DecelerationRate	Same as above
JerkRate	Not supported. Specify -1.0.
BlendingMode	The following values are supported: 0：Exact stop 11：RampOverlap
BlendingParameter	If 11 (RampOverlap) is specified in BlendingMode, specify the percentage 0.0 to 100.0 of overlap in BlendingParameter[0]. BlendingParameter[1] is not used.
Time	Not supported. Specify zero.
ConfigMode	The following values are supported for the Shoulder, Elbow, Wrist: 0：UseConfig 1：NoChange
TurnMode	The following values are supported: 0：UseTurnNumber 2：Free
Manipulation	Not supported. Specify FALSE.
EmmiterID	Not supported. Specify zero at all times.

Output parameters

Parameter name	Restriction
RemainingDistance	Not supported. -1.0 is output at all times.
Progress	Same as above
FollowID	Not supported. Zero is output at all times.

4.1.11 MoveLinearAbsolute

Input parameters

Parameter name	Restriction
Position	Since an external axis is not supported, the specification of E1 to E6 is ignored. In the case of a SCARA robot, the Z-axis direction of the Frame coordinates and Tool coordinates is inverted (rotated 180° around the X-axis) according to the SRCI specification, so specify 180.0 for RX.
VelocityRate	Values exceeding 100.0 are not supported. Specify within the range of 0.0 to 100.0.
AccelerationRate	Same as above
DecelerationRate	Same as above
JerkRate	Not supported. Specify -1.0.
BlendingMode	The following values are supported: 0：Exact stop 11：RampOverlap
BlendingParameter	If 11 (RampOverlap) is specified in BlendingMode, specify the percentage 0.0 to 100.0 of overlap in BlendingParameter[0]. BlendingParameter[1] is not used.
OriMode	The following values are supported: 1：Linear interpolated
Time	Not supported. Specify zero.
ConfigMode	The following values are supported for the Shoulder, Elbow, Wrist: 1：NoChange 2：Free
TurnMode	The following values are supported: 2：Free
Manipulation	Not supported. Specify FALSE.
EmmiterID	Not supported. Specify zero at all times.

Output parameters

Parameter name	Restriction
RemainingDistance	Not supported. -1.0 is output at all times.
Progress	Same as above
FollowID	Not supported. Zero is output at all times.

4.1.12 ReadActualPosition

Input parameters

Parameter name	Restriction
ProcessingMode	The following values are supported: 0：Buffered 1：Aborting 2：Parallel The trigger function is not supported.
ListenerID	Since the trigger function is not supported, specify zero in ListenerID.

Output parameters

Parameter name	Restriction
Position	Since the external axis is not supported, zero is output at all times for E1 to E6.
OriginID	Since the trigger function is not supported, zero is output at all times.
InvocationCounter	Same as above

4.1.13 ReadActualPositionCyclic

Input parameters

Parameter name	Restriction
ReadExtCartesianPosition	Not supported. Specify FALSE.
ReadExtJointPosition	Same as above

Output parameters

Parameter name	Restriction
ReadingExtCartesianPosition	Not supported. FALSE is output at all times.
ReadingExtJoinPosition	Same as above
ExtCartesianPosition	Not supported. Zero is output at all times.
ExtJointPosition	Same as above

4.1.14 ReadFrameData

Input parameters

Parameter name	Restriction
FrameNo	-1, or 0 to 15 can be specified.

There are no restrictions on the output parameters of this command.

4.1.15 ReadLoadData

Input parameters

Parameter name	Restriction
LoadNo	-1, or 1 to 16 can be specified.

There are no restrictions on the output parameters of this command.

4.1.16 ReadMessages

There are no restrictions on the input/output parameters of this command.

4.1.17 ReadRobotData

There are no restrictions on the input/output parameters of this command.

4.1.18 ReadRobotDefaultDynamics

There are no restrictions on the input parameters of this command.

Output parameters

Parameter name	Restriction
DynamicValues	Since JerkRate is not supported, -1.0 is output at all times for this item.

4.1.19 ReadRobotReferenceDynamics

There are no restrictions on the input parameters of this command.

Output parameters

Parameter name	Restriction
DynamicValues	Since JerkRate is not supported, -1.0 is output at all times for this item.

4.1.20 ReadRobotSWLimit

There are no restrictions on the input parameters of this command.

Output parameters

Parameter name	Restriction
LimitValues	Since the external axis is not supported, zero is output at all times for E1 to E6.

4.1.21 ReadServerLog

There are no restrictions on the input/output parameters of this command.

4.1.22 ReadToolData

Input parameters

Parameter name	Restriction
ToolNo	-1, or 0 to 15 can be specified.

There are no restrictions on the output parameters of this command.

4.1.23 ReturnToPrimary

Input parameters

Parameter name	Restriction
VelocityRate	Values exceeding 100.0 are not supported. Specify within the range of 0.0 to 100.0.
AccelerationRate	Same as above
DecelerationRate	Same as above
JerkRate	Not supported. Specify -1.0.
Time	Not supported. Specify zero.

Output parameters

Parameter name	Restriction
RemainingDistance	Not supported. -1.0 is output at all times.
Progress	Same as above

4.1.24 RobotTask

This command is executed on the PLC, and some of the settings are synchronized with the controller. Only the restrictions on parameters related to the controller are described below.

Input parameters

Parameter name	Restriction
UserData	The ReadWorkArea, WriteWorkArea, and WriteRobotSWLimit commands are not supported. Specify FALSE for the settings below. Items for which FALSE is to be set: - Config.PLC.EnableSync.WorkAreas - Config.PLC.EnableSync.SwLimit Items that are not supported (FALSE is specified): - Config.RC.WaitAtBlendingZone - Config.RC.AllowDynamicBlending - Config.RC.AllowSecSeqWhileSubprogram

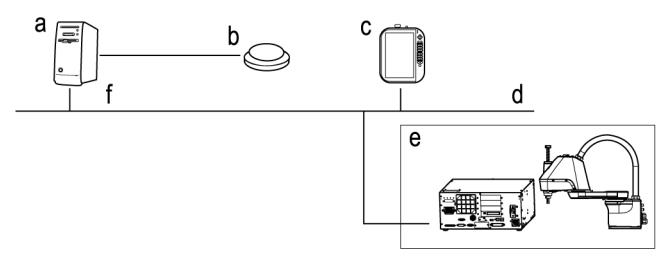
4.1.25 SetOperationMode

There are no restrictions on the input/output parameters of this command.

⚠ Warning

When switching the operation mode from "T1 External" to "Auto External" by operating the pendant (an HMI equipped with an emergency stop and Enable switch), configure the system so that SetOperationMode is executed only after confirming that switching authorization has been granted by another confirmation device installed outside the safety protection space. An example of device configuration and control is shown below.
If the robot is operated by switching the operation mode to "Auto External" without confirming that the operator is outside the safety protection space, the robot will operate at a high speed, which can be very dangerous, causing severe injury or serious damage.

Example of device configuration and control



Symbol	Description
a	PLC
b	Switch installed outside the safety protection space
c	HMI
d	Filedbus
e	Safety protection space
f	I/O control module

1. Connect the I/O control module to the PLC, and connect the signal from the switch installed outside the safety protection space to the I/O control module.
2. When the operation mode is switched from "T1 External" to "Auto External" with a pendant (HMI), if the pressed state of the switch can be confirmed from the system control software and it can be determined that the operator is outside the safety protection space, run the SetOperationMode command, and switch the operation mode from "T1 External" to "Auto External".

4.1.26 SetSequence

There are no restrictions on the input/output parameters of this command.

4.1.27 WriteFrameData

Input parameters

Parameter name	Restriction
FrameNo	1 to 15 can be specified.
FrameData	0 to 15 can be specified for ReferenFrame. - In the case of a SCARA robot, specify zero for RX and RY.

There are no restrictions on the output parameters of this command.

4.1.28 WriteLoadData

Input parameters

Parameter name	Restriction
LoadNo	1 to 16 can be specified.

There are no restrictions on the output parameters of this command.

4.1.29 WriteRobotDefaultDynamics

Input parameters

Parameter name	Restriction
DynamicValues	JerkRate is not supported. Specify -1.0 in JerkRate.

There are no restrictions on the output parameters of this command.

4.1.30 WriteRobotReferenceDynamics

Input parameters

Parameter name	Restriction
DynamicValues	JerkReference is not supported. Specify -1.0 in JerkReference.

There are no restrictions on the output parameters of this command.

4.1.31 WriteToolData

Input parameters

Parameter name	Restriction
ToolNo	1 to 15 can be specified.
ToolData	- 1 to 16 can be specified for LoadNo. - In the case of a SCARA robot, specify zero for RX and RY.

There are no restrictions on the output parameters of this command.

5. Alarm message code

5.1 Alarm message code

In the case of occurrence of an event that must be notified to the user, the controller issues an alarm message code. This code is classified as shown below.

Message type list

Message type	Description
RI	A message that occurs during communication with the PLC
RA	A message that occurs in connection with robot control
RC	A message that notifies the controller state
CMD	A message concerning command execution

If the message type is "RI" or "CMD", its details are defined in the SRCI specifications. Refer to the PLC manual, etc. The message type can be identified from the MessageLog, which is the input/output data of RobotTask.

The list of codes of the "RA" or "RC" type is described below.

Type	Code	Description
RA	0x0FA1～0x176F (4001～5999)	This is an error related to robot control. The meaning of the error code is same as the details specified in the "List of Status Codes and Error Codes (RC800 Series)" manual. See the manual separately. When an RA error is issued, an RC error may be issued with the same code.
RC	0x1AF4～0x1AF9 (6900～6905)	This is an error caused by the safety board. The meaning of the error code is same as the details specified in the "List of Status Codes and Error Codes (RC800 Series)" manual. See the manual separately.
	0x5801	Emergency stop has been pressed.
	0x5001	The safeguard has been enabled (safeguard open).
	0x5C01	Failed to initialize the controller. Restart the controller.
	0x5FFF	An RC internal error occurred while the controller was running. Restart the controller.

6. Mode LEDs and 7-segment LED

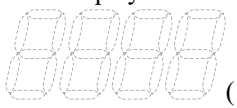
6.1 Mode LEDs and 7-segment LED

When SRCI control is enabled, the display of the mode LEDs and 7-segment LED on the controller is as described below.

Mode LEDs

LED name	ON condition
TEST	When the mode switches to T1 External or T2 External mode
TEACH	Does not turn ON.
AUTO	When the mode switches to Auto External mode
PROGRAM	When the control authority is transferred from the PLC to RC+ by connecting RC+ (when the mode switches to PROGRAM mode)

7-segment LED

Controller state	Display contents
Standby state	The following display is repeated - (S ---) Display - ( ) Display
Running state	The following display is repeated - LED display of the command type value -  ( ) Display * The command type value when the controller is running in the order buffer, or has paused is displayed. Example: When MoveLinearAbsolute is executed, 2104 is displayed. See: **Overview of the Controller Function**
Warning display	The following display is repeated - (SHLP) display - LED display of alarm message code See: **Alarm message code**
Error status	The following is repeated in the case of an error - (SE) display - LED display of alarm message code The following is repeated in the case of a fatal error - (SFE) display - LED display of alarm message code See: **Alarm message code**

- The alarm message code is a hexadecimal display (b and d are represented in lower case).

Errors and warnings not caused by SRCI control (such as fieldbus communication errors, emergency stops, safeguard open detection, and parts wear management warnings) follow the display methods and codes used during normal operation. For the

meaning of the display contents and codes, refer to the following manuals:
"Robot Controller RC800-A/RC800L Manual"
"List of Status Codes and Error Codes (RC800 Series)"

7. Method of Setting the Safety Board

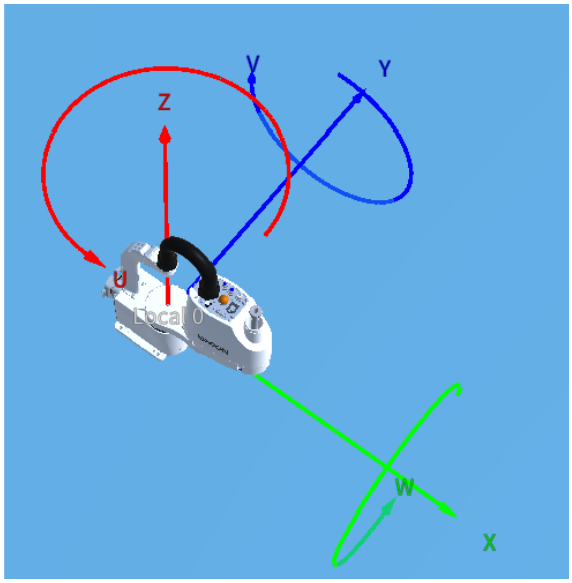
7.1 Method of Setting the Safety Board

The SRCI BASE coordinate system differs from the coordinate system assumed by the RC+ Safety Function Manager (the BASE coordinate system used when control is performed using the Epson robot control programming language, SPEL+).

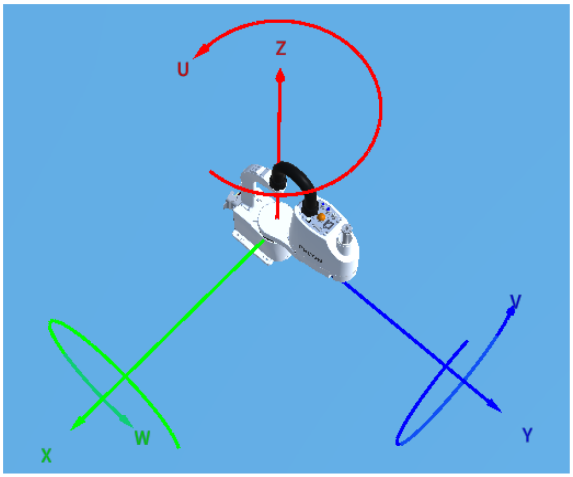
Therefore, when restricting the robot operating range by the safety board, be sure to first understand the precautions below.

BASE coordinate system	Description
SRCI	The front of the robot base corresponds to the +X axis. The 0° of the J1 rotation axis is on the +X axis.
SPEL+	The front of the robot base is the +Y axis. The 0° of the J1 rotation axis is on the +X axis.

Coordinate system during control by the SRCI.

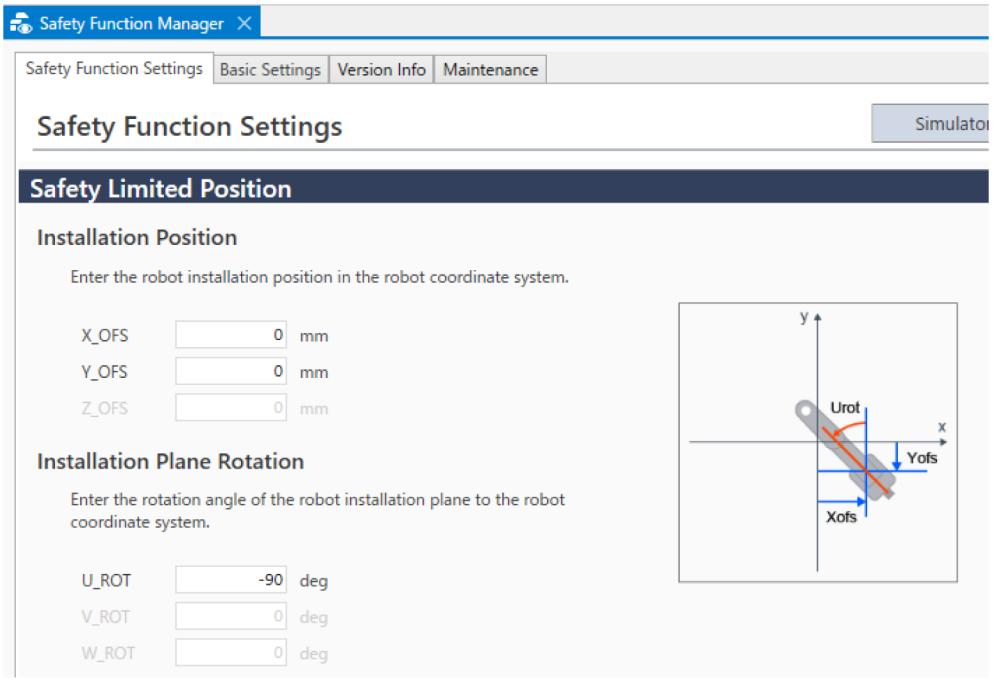


Coordinate system during control by SPEL+ language.



7.1.1 Setting Safety Position Monitoring

The SRCI coordinate system is rotated 90° around the Z-axis. Therefore, by setting "U_ROT" to -90, the X and Y coordinate values of the monitoring area can be specified in the SRCI coordinate system.



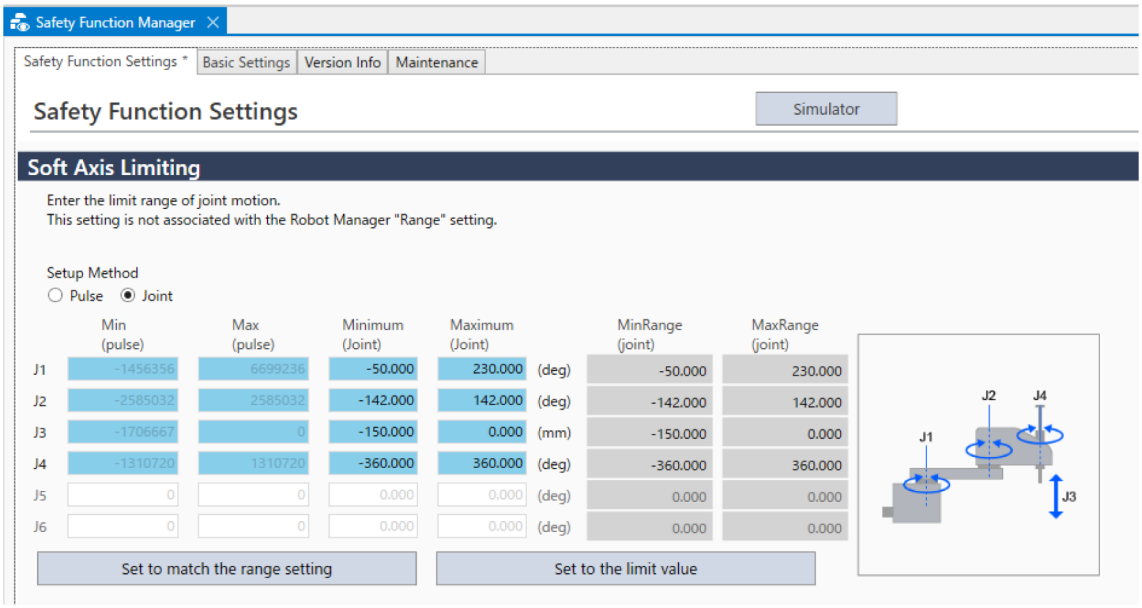
⚠ Warning

If the controller does not run any SRCI commands, it will switch to the PROGRAM mode when RC+ is connected, and the robot can be operated on RC+.
When safety position monitoring is set in the Safety Function Manager, the XY coordinates specified by RC+ and the restricted area will be in a positional relationship of 90° deviation around the Z-axis, if -90 is set in U_ROT.
When the robot is to be moved with RC+, ensure that the robot does not enter the restricted area by taking into consideration the differences in the coordinates.
If the restricted area is not identified correctly, there is a risk that "a worker may accidentally enter the robot motion range" or "an obstacle may be placed within the motion range". Collisions with high-speed robots may result in serious injury and/or severe equipment damage.

7.1.2 Setting the Software Axis Restrictions

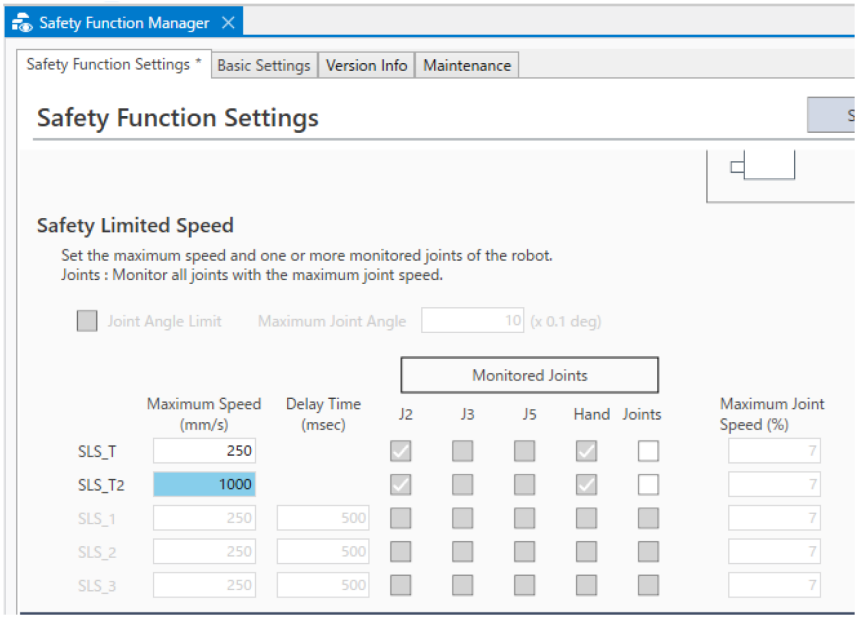
The zero degree of the J1 rotation axis in SRCI is obtained by subtracting 90° from the J1 rotation axis value in SPEL+ control.

Therefore, when the J1 rotation axis is restricted by Joint specification, set a value by adding 90°. (Example: If the J1 axis is in the range of "-140 to 140 degrees" in the SRCI specification, set "-50 to 230 degrees".)



7.1.3 Safety Limited Speed of T1 and T2 External Mode

The robot moving speed during T1 and T2 external mode is restricted by the safety limited speed. This setting is prioritized more than the settings of the SRCI's speed override change command and the moving command's input parameter. When operating the robot at a moving speed of 250 mm/s or higher, change the safety limited speed settings.



7.1.4 How to Recover from Stopping by SLP

If the robot is stopped by the SLP (detection of intrusion into the restricted area), move the robot outside the restricted area according to the method described below.
Perform this task after checking the "Safety Manual". You must take proper safety measures in situations where the robot may fall under its own weight when the brake is released.

Robot type	Recovery method
6-axis robot	1. Turn the controller OFF. 2. Disconnect the wiring from the fieldbus module so that it is not connected to the PLC. 3. Turn the controller ON. 4. Connect the controller with the PC and release the brakes from RC+. 5. Manually move the robot position outside the restricted area.
SCARA	**J1, J2, J4** 1. Turn the controller OFF. 2. Manually move the robot position outside the restricted area. **J3** After turning the robot ON, manually move the robot position outside the restricted area while pressing the J3 brake release switch. For details on the position of the brake release switch, refer to the section on "How to operate the arms in an emergency stop state" in the manual of each manipulator.

8. Other Restrictions

8.1 Restrictions on the Controller Function

The SRCI optional functions cannot be used in conjunction with the following operations and functions:

- Execution of an SPEL program
- Optional controls such as force sensors, vision systems, part feeders, or PGs
- Use of the TP2 or TP4