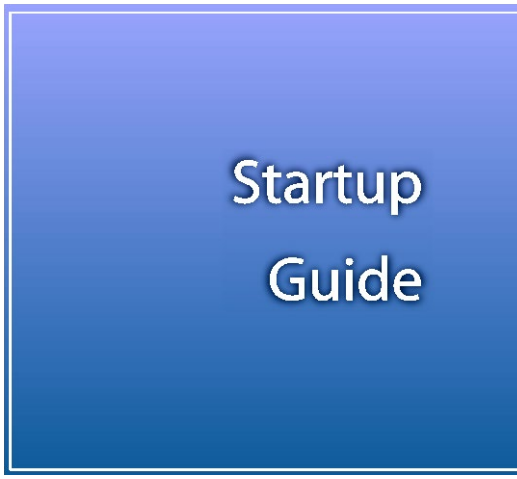
AC Servo System 1S-series with Built-in SS1/SLS Safety Sub-Functions

Startup Guide

CK3M/CK5M CPU Unit (Programmable Multi-axis Motion Controller) for Connection

R88M-1L□/-1M□ (AC Servomotor)
R88D-1SN□-ECT-51 (AC Servo Drive)
CK3M/CK5M CPU Unit
(Programmable Multi-axis Motion Controller)
NX-series Safety Units
Sysmac Studio
Power PMAC IDE



always in control

I929-E1-01

NOTE

Introduction

The AC Servo System 1S-Series Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions Startup Guide (hereinafter, may be referred to as “this Guide”) describes the installation and setup of 1S-series Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions (hereinafter, may referred to as “1S-51”). This Guide assumes a system that consists of a combination of a Programmable Multi-axis Motion Controller CK3M/CK5M CPU Unit, a 1S-series AC Servomotor/Servo Drive with Built-in SS1/SLS Safety Sub-Functions, an NX-series Safety Control Unit, the Sysmac Studio, and the Power PMAC IDE. A simple installation model is used for the discussion. You can perform the procedures that are presented in this Guide to quickly gain a basic understanding of 1S-series AC Servomotor/Servo Drive with Built-in SS1/SLS Safety Sub-Functions.
Note that this Guide does not contain safety information and other details that are required for actual use. Thoroughly read and understand the manuals for all of the devices that are used in this Guide to ensure that the system is used safely. Review the entire contents of these materials, including all safety precautions, precautions for safe use, and precautions for correct use.

Intended Audience

This Guide is intended for the following personnel.

- Personnel in charge of introducing FA systems
- Personnel in charge of designing FA systems

The personnel must also have the following knowledge.

- Knowledge of electrical systems (an electrical engineer or the equivalent)
- Knowledge of Programmable Multi-axis Motion Controller CK3M/CK5M CPU Units
- Knowledge of NX-series Safety Units
- Knowledge of AC Servomotors/Drives
- Knowledge of operation procedure of Sysmac Studio and Power PMAC IDE

Applicable Products

This Guide covers the following products.

- Programmable Multi-axis Motion Controller CK3M/CK5M CPU Unit
- Automation Software Sysmac Studio
- Power PMAC Software Package Power PMAC IDE
- 1S-series Servomotor/Servo Drive with Built-in SS1/SLS Safety Sub-Functions
- NX-series EtherCAT Coupler Unit
- NX-series Safety Control Unit

Special Information

Special information in this Guide is classified as follows:

Precautions for Correct Use

Precautions on what to do and what not to do to ensure safe usage of the product.

Additional Information

Additional information to read as required.

This information is provided to increase understanding or make operation easier.

Terms and Conditions Agreement

Warranties

(a) Exclusive Warranty. Omron's exclusive warranty is that the Products will be free from defects in materials and workmanship for a period of twelve months from the date of sale by Omron (or such other period expressed in writing by Omron). Omron disclaims all other warranties, express or implied.

(b) Limitations. OMRON MAKES NO WARRANTY OR REPRESENTATION, EXPRESS OR IMPLIED, ABOUT NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OF THE PRODUCTS. BUYER ACKNOWLEDGES THAT IT ALONE HAS DETERMINED THAT THE PRODUCTS WILL SUITABLY MEET THE REQUIREMENTS OF THEIR INTENDED USE.

Omron further disclaims all warranties and responsibility of any type for claims or expenses based on infringement by the Products or otherwise of any intellectual property right. (c) Buyer Remedy.

Omron's sole obligation hereunder shall be, at Omron's election, to (i) replace (in the form originally shipped with Buyer responsible for labor charges for removal or replacement thereof) the non-complying Product, (ii) repair the non-complying Product, or (iii) repay or credit Buyer an amount equal to the purchase price of the non-complying Product; provided that in no event shall Omron be responsible for warranty, repair, indemnity or any other claims or expenses regarding the Products unless Omron's analysis confirms that the Products were properly handled, stored, installed and maintained and not subject to contamination, abuse, misuse or inappropriate modification. Return of any Products by Buyer must be approved in writing by Omron before shipment. Omron Companies shall not be liable for the suitability or unsuitability or the results from the use of Products in combination with any electrical or electronic components, circuits, system assemblies or any other materials or substances or environments. Any advice, recommendations or information given orally or in writing, are not to be construed as an amendment or addition to the above warranty.

See http://www.omron.com/global/ or contact your Omron representative for published information.

Limitation on Liability; Etc

OMRON COMPANIES SHALL NOT BE LIABLE FOR SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, LOSS OF PROFITS OR PRODUCTION OR COMMERCIAL LOSS IN ANY WAY CONNECTED WITH THE PRODUCTS, WHETHER SUCH CLAIM IS BASED IN CONTRACT, WARRANTY, NEGLIGENCE OR STRICT LIABILITY.

Further, in no event shall liability of Omron Companies exceed the individual price of the Product on which liability is asserted.

Suitability of Use

Omron Companies shall not be responsible for conformity with any standards, codes or regulations which apply to the combination of the Product in the Buyer's application or use of the Product. At Buyer's request, Omron will provide applicable third party certification documents identifying ratings and limitations of use which apply to the Product. This information by itself is not sufficient for a complete determination of the suitability of the

Product in combination with the end product, machine, system, or other application or use. Buyer shall be solely responsible for determining appropriateness of the particular Product with respect to Buyer's application, product or system. Buyer shall take application responsibility in all cases.
NEVER USE THE PRODUCT FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE OMRON PRODUCT(S) IS PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.

Programmable Products
Omron Companies shall not be responsible for the user's programming of a programmable Product, or any consequence thereof.

Performance Data
Data presented in Omron Company websites, catalogs and other materials is provided as a guide for the user in determining suitability and does not constitute a warranty. It may represent the result of Omron's test conditions, and the user must correlate it to actual application requirements. Actual performance is subject to the Omron's Warranty and Limitations of Liability

Change in Specifications
Product specifications and accessories may be changed at any time based on improvements and other reasons. It is our practice to change part numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the Product may be changed without any notice. When in doubt, special part numbers may be assigned to fix or establish key specifications for your application. Please consult with your Omron's representative at any time to confirm actual specifications of purchased Product.

Errors and Omissions
Information presented by Omron Companies has been checked and is believed to be accurate; however, no responsibility is assumed for clerical, typographical or proofreading errors or omissions.

Precautions

- When building a system, check the specifications for all devices and equipment that will make up the system and make sure that the OMRON products are used well within their rated specifications and performances. Safety measures, such as safety circuits, must be implemented in order to minimize the risks in the event of a malfunction.
- Thoroughly read and understand the manuals for all devices and equipment that will make up the system to ensure that the system is used safely. Review the entire contents of these materials, including all safety precautions, precautions for safe use, and precautions for correct use.
- Confirm all regulations, standards, and restrictions that the system must adhere to.
- Check the user program for proper execution before you use it for actual operation.

Trademarks

- Sysmac and SYSMAC are trademarks or registered trademarks of OMRON Corporation in Japan and other countries for OMRON factory automation products.
- Windows is either a registered trademark or trademark of Microsoft Corporation in the USA and other countries.
- EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
- Safety over EtherCAT® is a registered trademark and a patented technology licensed by Beckhoff Automation GmbH, Germany.
- Microsoft product screen shot(s) reprinted with permission from Microsoft Corporation.
- Other company names and product names in this Guide are the trademarks or registered trademarks of their respective companies.

Software Licenses and Copyrights

The Sysmac Studio incorporates certain third party software. For the license and copyright information associated with this software, please refer to NJ/NX/NY series/Sysmac Studio: Third Party License List & Source code Download: Tools: OMRON (omron.co.jp).

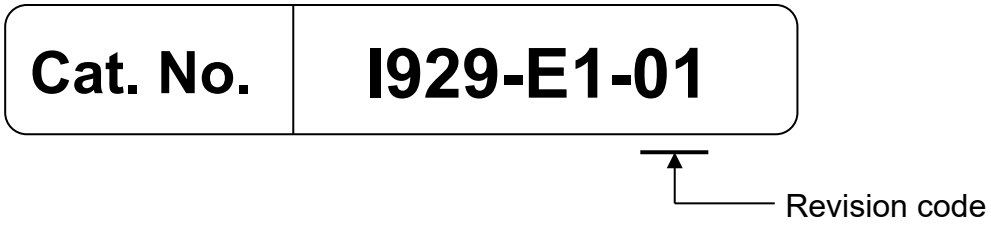
Related Manuals

The following manuals are related. Use these manuals for reference.

Manual name	Cat. No.	Model	Application	Description
1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual	I696	R88D-1S□-ECT-51 R88M-1□	Learning detailed specifications of a 1S-series Servo Drive with Built-in SS1/SLS Safety Sub-Functions.	Describes how to install and wire the Servo Drive, set parameters needed to operate the Servo Drive, and remedies to be taken and inspection methods to be used in case that problem occur.
Sysmac Studio Version 1 Operation Manual	W504	SYSMAC-SE2□□□	Learning about the operating procedures and functions of the Sysmac Studio.	Describes the operating procedures of the Sysmac Studio.
Sysmac Studio Drive Functions Operation Manual	I589	SYSMAC-SE2□□□	Learning how to set up and adjust the Servo Drives.	Describes the operating procedures of the Sysmac Studio.
NX series EtherCAT Coupler Unit User's Manual	W519	NX-ECC201 NX-ECC202 NX-ECC203	Learning how to use the EtherCAT® Coupler Units.	Describes the hardware, setup methods and functions of the EtherCAT® Coupler Units.
NX Series Digital I/O Units User's Manual	W521	NX-ID□□□□ NX-IA□□□□ NX-OC□□□□ NX-OD□□□□ NX-MD□□□□	Learning how to use the NX-series Digital I/O Units.	Describes the hardware, setup methods and functions of the NX-series Digital I/O Units.
NX Series System Units User's Manual	W523	NX-PD1□□□ NX-PF0□□□ NX-PC0□□□ NX-TBX01	Learning how to use the NX-series System Units.	Describes the hardware, setup methods and functions of the NX-series System Units.
NX-series Safety Control Units User's Manual	Z930	NX-SL□□□□ NX-SI□□□□ NX-SO□□□□	Learning how to use the NX-series Safety Control Units.	Describes the hardware, setup methods and functions of the NX-series Safety Control Units.
Programmable Multi-Axis Controller User's Manual (Hardware)	O036	CK5M-CPU1□1 CK3M-CPU1□1 CK3W-PD048 CK3W-AX1313□/-AX1414□/-AX1515□/-AX2323□ CK3W-MD71□0 CK3W-AD□100 CK3W-ECS300 CK3W-GC□□00 CK3W-EXM01/-EXS02 CK5W-EXS01	Learning how to use the CK3M/CK5M-series Programmable Multi-axis Motion Controllers.	Describes the hardware, setup methods and functions of the CK3M/CK5M-series Programmable Multi-axis Motion Controller.
Power PMAC IDE Software User Manual	---	Power PMAC IDE	Learning how to use the Power PMAC IDE.	Describes the operating procedures of the Power PMAC IDE

Revision History

A manual revision code appears as a suffix to the catalog number on the front and back covers of the manual.



Revision code	Date	Revised content
01	February 2025	Original production

CONTENTS

Introduction **3**
Intended Audience 3
Applicable Products 3
Special Information 4
Terms and Conditions Agreement **5**
Precautions **7**
Trademarks 7
Software Licenses and Copyrights 7
Related Manuals **8**
Revision History **9**
1. Servo System Configuration and Peripheral Products **12**
1.1. Outline 12
1.2. Servo System Constructed in This Guide 13
1.3. System Configuration 14
2. Before You Begin **16**
3. Performing Setup **18**
3.1. Installation and Wiring 20
3.2. System Configuration with NX-series Safety Control Units 28
3.3. Setup of NX-series Safety Control Units 29
3.3.1. Creating a Network Configuration 29
3.3.2. Setting to Transfer Data from the Standard Controller to the Safety Controller 33
3.3.3. Setting the Safety Controller 34
3.3.4. Creating a Safety Program 37
3.4. Setup of the Motion Controller CK5M 46
3.4.1. Preparing the Controller for Setup 46
3.4.2. Installing the ESI File 51
3.4.3. Configuring EtherCAT Communications Settings 52
3.4.4. Setting Axis Variables 62
3.4.5. Creating an Operation Program 64
3.4.6. Downloading the Project and Checking the Operation 66
3.5. Setup of the Servo Drive 68
3.5.1. Connecting the Servo Drive via CK5M 68
3.5.2. Motor, ABS Encoder and I/O Setup 69
3.6. Operation Check of the STO Function 72
4. Adding Safety Functions **73**
4.1. Adding the Safe Stop 1 (SS1) Function 73
4.1.1. Setting the Safety Controller 75
4.1.2. Setup of the CK5M 80
4.1.3. Checking Operation of the SS1 Function 84

4.2. Adding the Safely-limited Speed (SLS) Function 85
4.2.1. Setting the Safety Controller 88
4.2.2. Setup of the CK5M 94
4.2.3. Checking Operation of the SLS Functions 98
Appendices 100
Test Run and Data Trace 100
Easy Tuning 103
Manual Tuning 107
Advanced Auto-Tuning 111

1. Servo System Configuration and Peripheral Products

1.1. Outline

The 1S-series Servo Drives with Built-in EtherCAT® Communications and SS1/SLS Safety Sub-Functions support 100-Mbps EtherCAT.
When you use the 1S-series Servo Drive with a Machine Automation Controller NJ-series CPU Unit or a Programmable Multi-axis Motion Controller CK3M/CK5M CPU Unit, you can construct a high-speed and so sophisticated position control system.
You need only one communications cable to connect the Servo Drive and the Controller. Therefore, you can realize a position control system easily with reduced wiring effort.
With tuning functions, adaptive notch filter, notch filter, and damping control, you can set up a system that provides stable operation by suppressing vibration in low-rigidity machines.
The 1S-series Servo Drives with Built-in EtherCAT® Communications and SS1/SLS Safety Sub-Functions support the FSoE (Safety over EtherCAT) protocol as a safety communications feature. You can build a safety system that uses the STO, SS1, and SLS functions from NX-series Safety Control Units on an EtherCAT network.

Additional Information

For further details on the 1S-series Servo Drives with Built-in EtherCAT® Communications and SS1/SLS Safety Sub-Functions, refer to the *1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual* (Cat. No. I696).

1.2. Servo System Constructed in This Guide

This Guide contains instructions from assembling the hardware that makes up a servo system to constructing a system for safety functions and performing debugging on the system. The servo system is built through the following steps:

■ Performing Setup

1	Installation and Wiring
▼	
2	System Configuration with NX-series Safety Control Units
▼	
3	Setup of NX-series Safety Control Units
▼	
4	Setup of the Motion Controller CK5M
▼	
5	Setup of the Servo Drive
6	Operation Check of the STO Function
▼	

■ Adding Safety Functions

1	Select safety functions from below and add them. • Adding the Safe Stop 1 (SS1) Function • Adding the Safely-limited Speed (SLS) Function

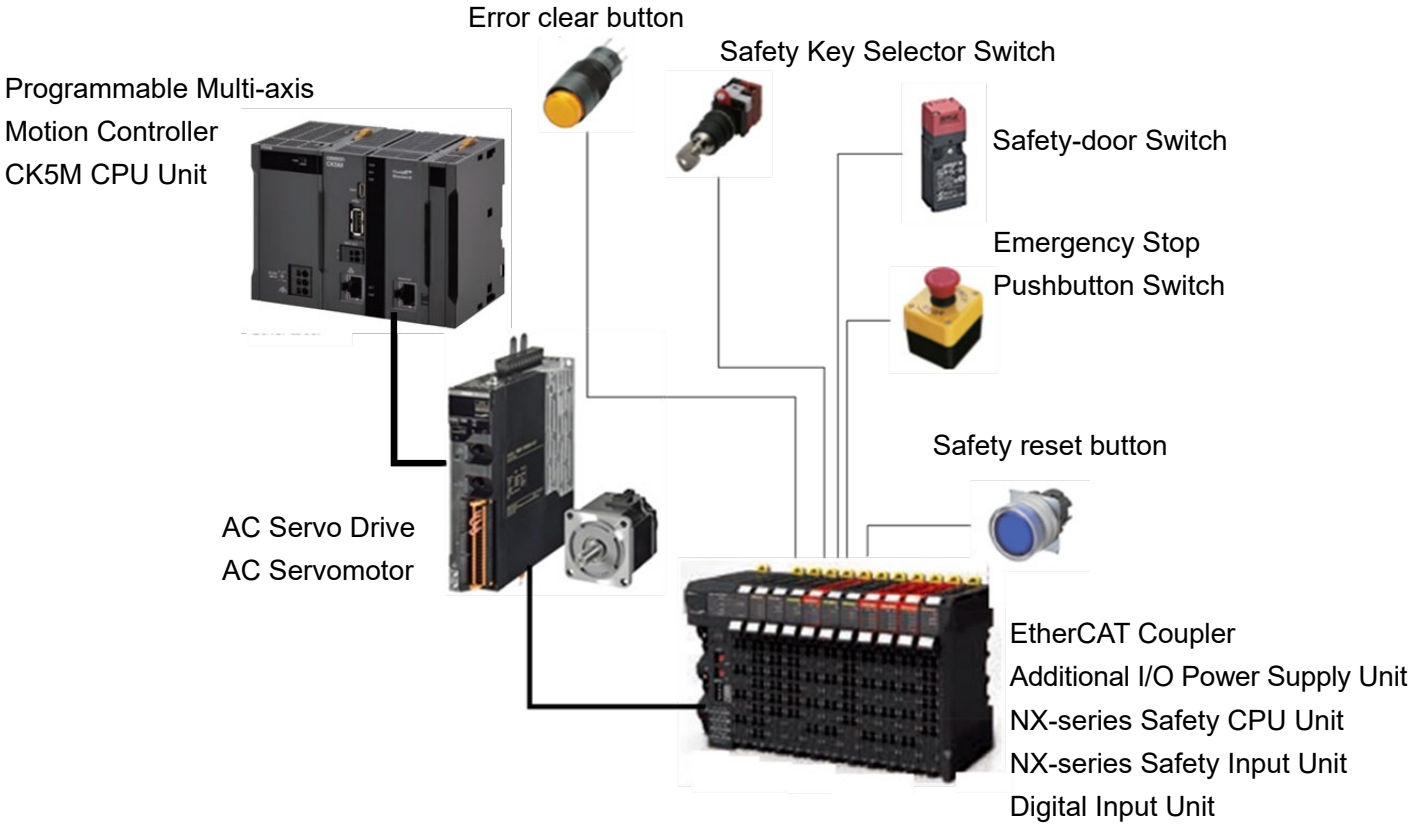
Additional Information

For information on how to set up the motion controller, refer to the *Programmable Multi-Axis Controller User's Manual (Hardware)* (Cat. No. O036).

1.3. System Configuration

The following figure shows the system configuration and devices that are used in this Guide.

- **Standard System Configuration**



- **Configuration Devices**

The models of the devices that are described in this Guide are given in the following table. When selecting devices for an actual application, refer to the device manuals.

Device name	Model	Manual name
Programmable Multi-axis Motion Controller CK5M CPU Unit (Standard controller)	CK5M	Programmable Multi-axis Motion Controller Hardware User's Manual
EtherCAT Coupler	NX-ECC20[]	EtherCAT® Coupler Unit User's Manual (Cat. No. W519)
Additional I/O Power Supply Unit	NX-PF0[]	NX Series System Units User's Manual (Cat. No. W523)
Digital Input Unit	NX-ID[]	NX Series Digital I/O Units User's Manual (Cat. No. W521)
NX-series Safety Control Unit (Safety controller)	NX-SL3500	NX-series Safety Control Units User's Manual (Z930)
NX-series Safety Input Unit	NX-SID[]	

Device name	Model	Manual name
Ethernet/EtherCAT Communications Cable	XS5W-T[]	---
AC Servo Drive	R88D-1SN[]-51	1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual (Cat. No. I696)
AC Servomotor	R88M-1[]	
Power Cable	R88A-CA[]	
Encoder Cable	R88A-CR[]	
Error clear button	A3[]	---
Safety Key Selector Switch	A22TK[]	---
Safety-door Switch	D4NS[]	---
Emergency Stop Pushbutton Switch	A22[]	---

- **Automation Software**

Product	Number of licenses	Model
Sysmac Studio Standard Edition Version 1.59.0.0	None (DVD only)	SYSMAC-SE200D
	From 1 license to site license	SYSMAC-SE[]
Power PMAC IDE Version 4.6.4 or later	---	---

Additional Information

This Startup Guide is written on the assumption that you will use the safety functions. If you do not use the safety functions, refer to sections 5 and 6 of the *Programmable Multi-Axis Controller Startup Guide for 1S-series Servo Drive (IDEv4)* (Cat. No. O039).

2. Before You Begin

■ Unpacking

1.	**Unpack the motor package.** The package includes only one Servomotor and the instruction sheet. Cables are provided separately.  
2.	**Unpack the drive package.** The Servo Drive comes with the following accessories. • INSTRUCTION MANUAL × 1 copy • INSTRUCTION MANUAL supplement × 1 copy • Warning label × 1 sheet • General Compliance Information and instructions for EU × 1 copy • Attached connector  

Additional Information

For further details on how to handle the Servo Drive and Servomotor, refer to the *Items to Check After Unpacking* in the *1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual* (Cat. No. I696).

■ Installing the Sysmac Studio Standard Edition Version 1.59

Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for details on how to install the Sysmac Studio.

■ Installing Power PMAC IDE Version 4.6.4 or Later

Refer to the *Power PMAC IDE Software User Manual* (V4) for the download procedure.

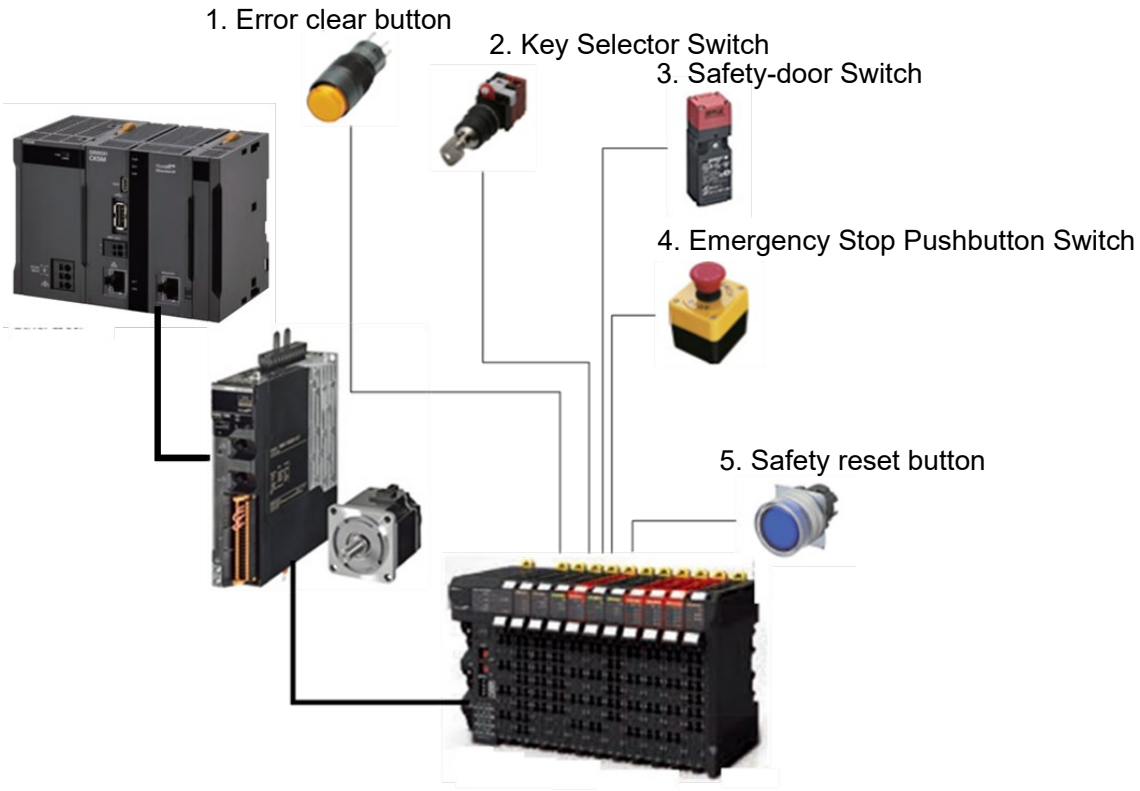
Note: The setup procedures described in this Startup Guide are available only when the language setting of the Power PMAC IDE is *English*. After installing the Power PMAC IDE, change the language setting to *English*.

3. Performing Setup

This section explains from assembling the hardware that makes up the servo system to adding the STO function via FSoE and creating a motor control program. The next section 4. Adding Safety Functions describes how to add safety functions other than the STO function.

The operation of the servo system set up in this section is explained below.

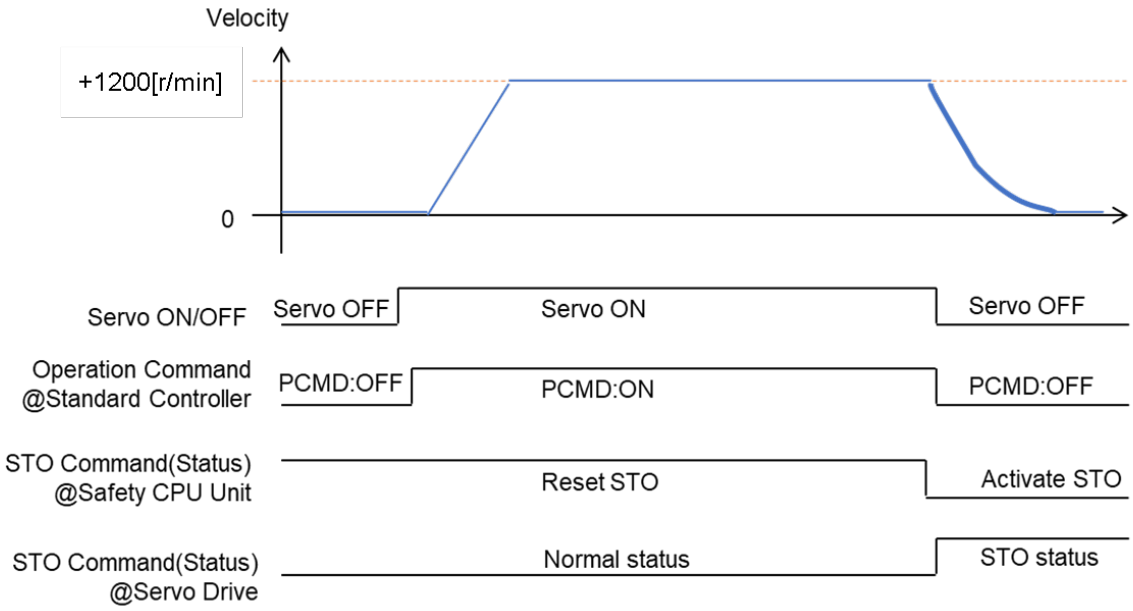
1. When the error clear button is pressed, the errors of the standard controller and Servo Drive are reset.
2. When the guard with the Safety-door Switch is opened, the motor torque is turned OFF.
3. When the Emergency Stop Pushbutton Switch is pressed, the motor torque is turned OFF.
4. When the safety reset button is pressed, the STO status is reset.



Input device	State	Operation
1. Error clear button	ON	Enable error reset command
	OFF	Disable error reset command
2. Safety Key Selector Switch * Used in section 4 and later.	Normal operating mode	Set the Servomotor to normal velocity.
	SS1/SLS safety function active	Activate assigned safety function. * The assignment procedure is described in section 4.
3. Safety-door Switch	Open	Enable STO command
	Close	Disable STO command
4. Emergency Stop Pushbutton Switch	ON	Enable STO command
	OFF	Disable STO command
5. Safety reset button	ON	Enable reset STO status command
	OFF	Disable reset STO status command

■ Operation of STO Function with Motion Control

1. When the Servo ON command is enabled, the Servo Drive turns ON the Servo.
2. When the STO function is executed, the Servo Drive shifts to the STO state and turns OFF torque.
3. When an operation command is enabled, the command velocity to the Servomotor is set to 1,200 r/min.



3.1. Installation and Wiring

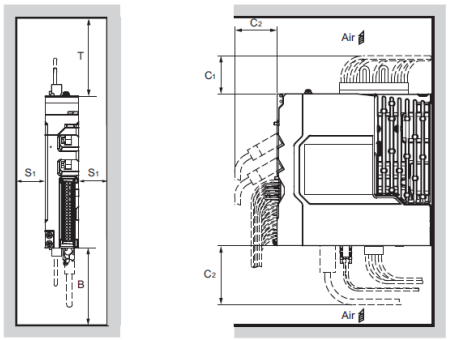
This section describes the installation and wiring of the AC Servo Drive, using the R88D-1SN01L-ECT-51 as an example.

For models for which the installation and wiring procedures differ from those for the R88D-1SN01L-ECT-51, refer to the *1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual* (Cat. No. I696).

■ Space Conditions around Servo Drives

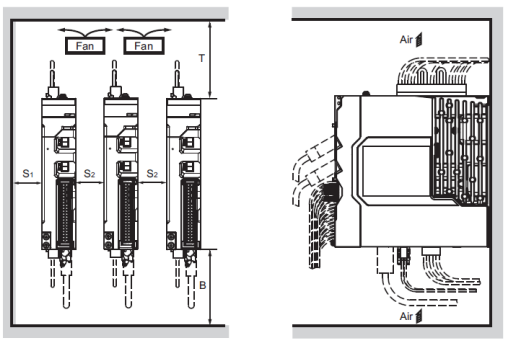
Install the Servo Drive according to the following instructions.

● Single-unit Installation



Front view　　Side view

● Side-by-side Installation



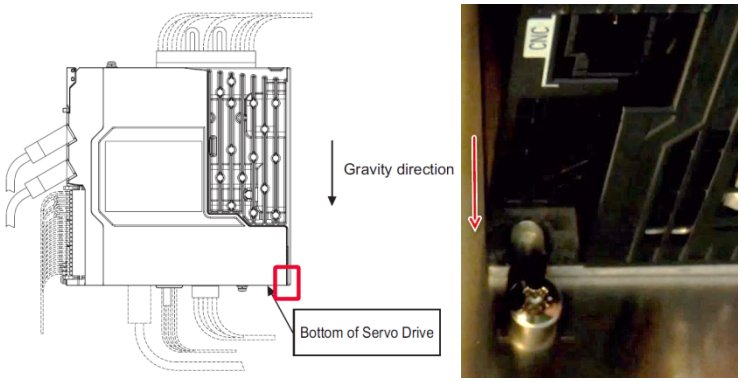
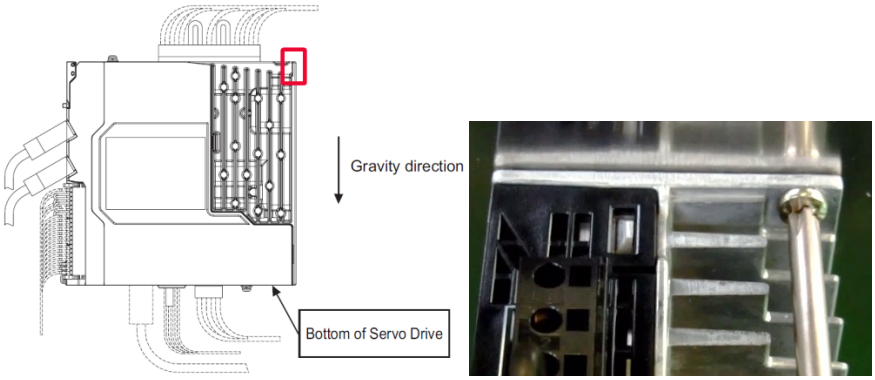
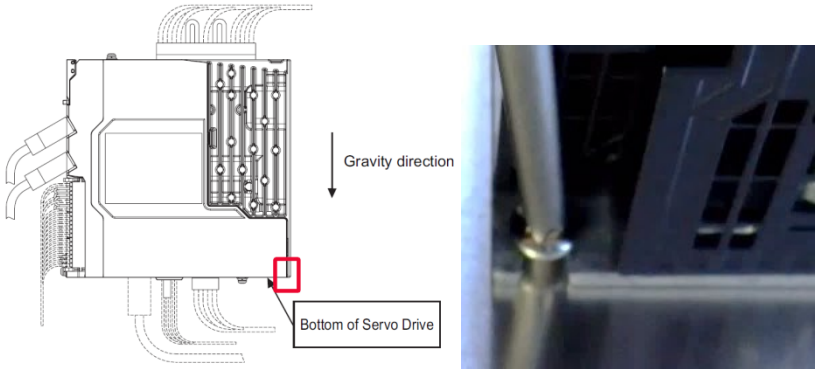
Front view　　Side view

Dimension	Distance	
T	100mm min.	
B	100mm min.	
S1	40mm min.	
S2	10 mm min. *1	
C1	R88D-1SN01L-ECT-51 / R88D-1SN02L-ECT-51/ R88D-1SN04L-ECT-51 / R88D-1SN01H-ECT-51/ R88D-1SN02H-ECT-51 / R88D-1SN04H-ECT-51/ R88D-1SN08H-ECT-51 / R88D-1SN10H-ECT-51	45mm min.
	R88D-1SN15H-ECT-51 / R88D-1SN20H-ECT-51/ R88D-1SN30H-ECT-51 / R88D-1SN06F-ECT-51/ R88D-1SN10F-ECT-51 / R88D-1SN15F-ECT-51/ R88D-1SN20F-ECT-51 / R88D-1SN30F-ECT-51	60mm min.
C2	50mm min.	
C3	70mm min.	

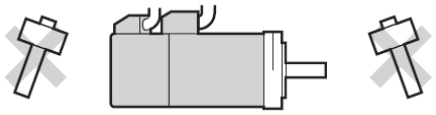
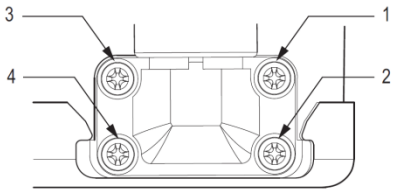
*1. If S2 must be less than 10 mm, keep the operating ambient temperature of the Servo Drive within 0 to 45ºC.

- Install the Servo Drive on the vertical metal surface.
- To provide electrical conduction, remove any paint from the surface on which you install the Servo Drives. Also, it is recommended that you apply conductive plating if you make the mounting bracket by yourself.
- The recommended tightening torque for installing the Servo Drive is 1.5 N·m. Make sure that the threaded portion has the sufficient strength to withstand the recommended torque.

■ Mounting the Servo Drive

1.	Put the Servo Drive in place and secure the top first and then the bottom. Temporarily tighten the screw to attach the lower part of the Servo Drive and slide the Servo Drive from top to bottom to put the U-groove of the Servo Drive on the screw. 
2.	Tighten the screw at the upper part. 
3.	Tighten the screw at the lower part. 

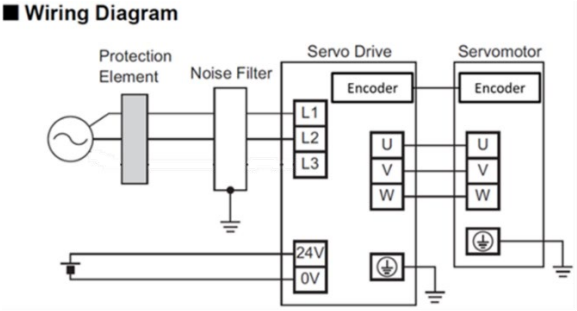
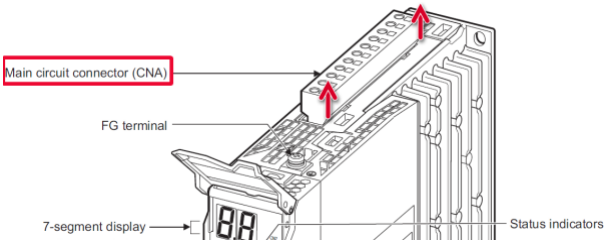
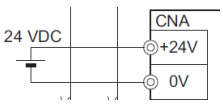
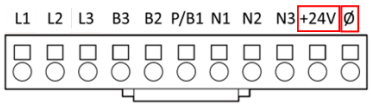
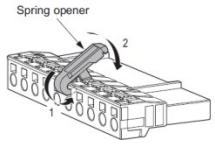
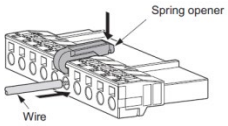
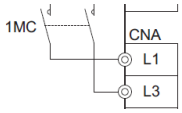
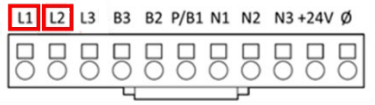
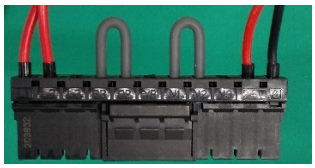
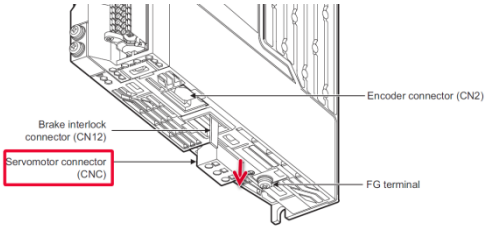
■ **Mounting the Servomotor** (The order of steps 2 and 3 may be reversed depending on how the motor is mounted on your machine)

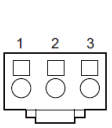
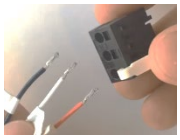
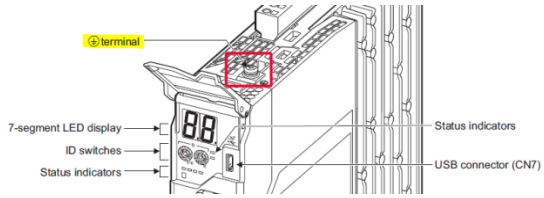
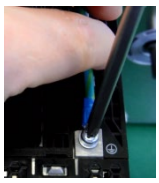
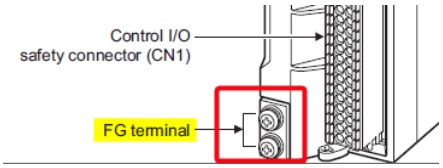
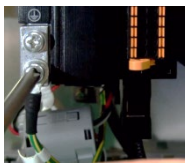
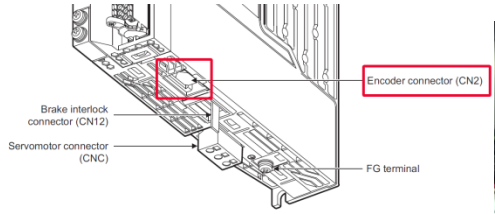
1.	Handle the Servomotor carefully and do not apply heavy impacts or loads during transport, installation, or removal of the Servomotor. 
2.	Fix and connect the Servomotor to the mechanical system.  First, check motor operation without any load.
3.	Connect the power cable and the encoder cable.    Example: Tighten screws in the order of 1, 4, 3, and 2.

Additional Information

For further details on how to connect the cables, refer to *4-1 Installation Conditions* in the *1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual* (Cat. No. I696).

■ Wiring

1.	Overview **■ Wiring Diagram** 
2.	Remove the main circuit connector (CNA) from the Servo Drive. 
3.	Wire the 24 V control power supply.   Connect wires with the spring opener.  
4.	Wire the main circuit power supply.   
5.	Remove the motor connector (CNC) from the Servo Drive. 

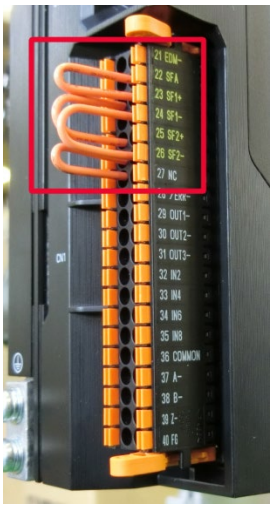
6.	Connect the power line to the phase U, V, and W of the Servomotor. **Pin No.** / **Name** 1 / U 2 / V 3 / W  
7.	Screw the PE wire of the main circuit power supply cable to the Servo Drive.  
8.	Plug back the above connectors to the Servo Drive. (Power and motor)
9.	Screw the FG wire of the motor cable to the Servo Drive.  
10.	Plug the encoder cable into the Servo Drive. (CN2)  

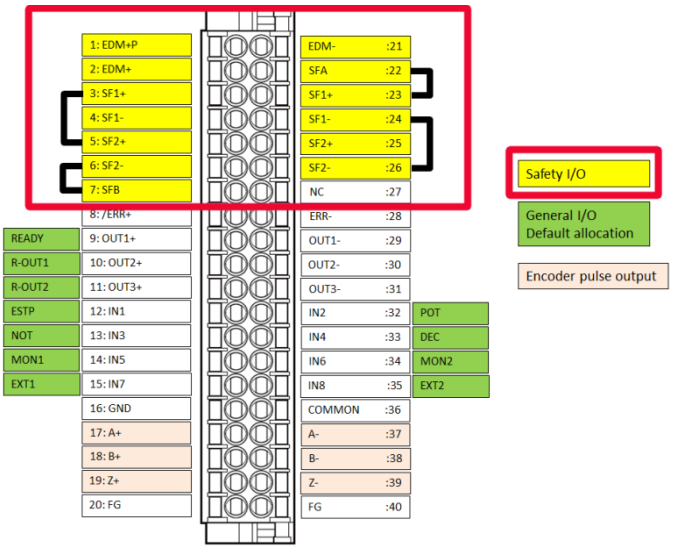
Additional Information

For further details on how to make the wiring, refer to *4-2 Wiring* in the *1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual* (Cat. No. I696).

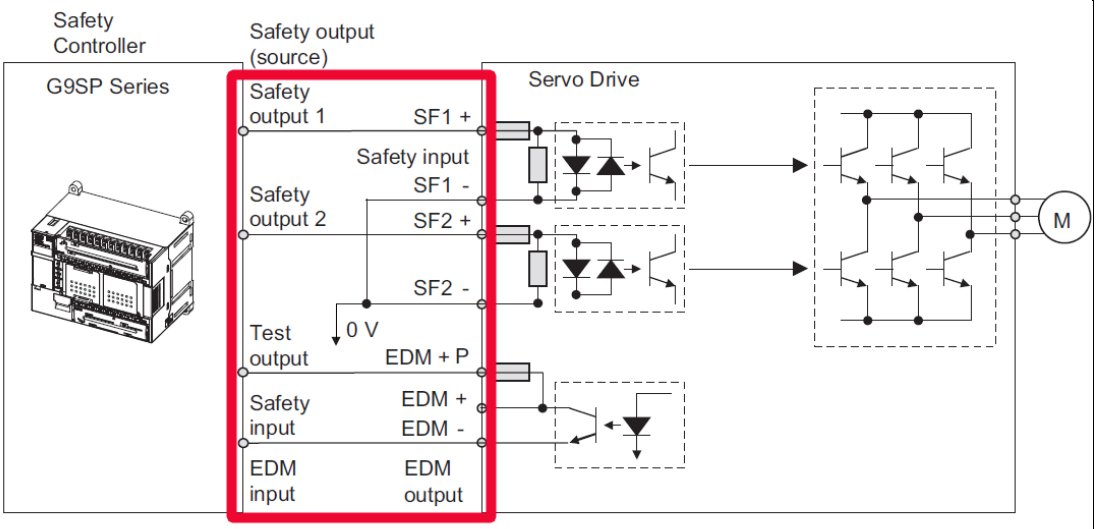
■ I/O and Safety Wiring

1. By default, STO is disabled and bypassed with jumpers.

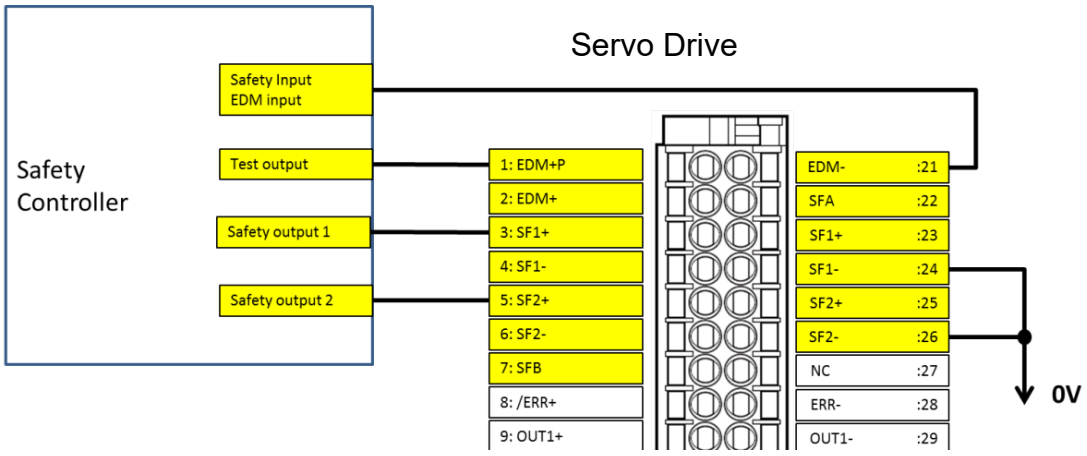




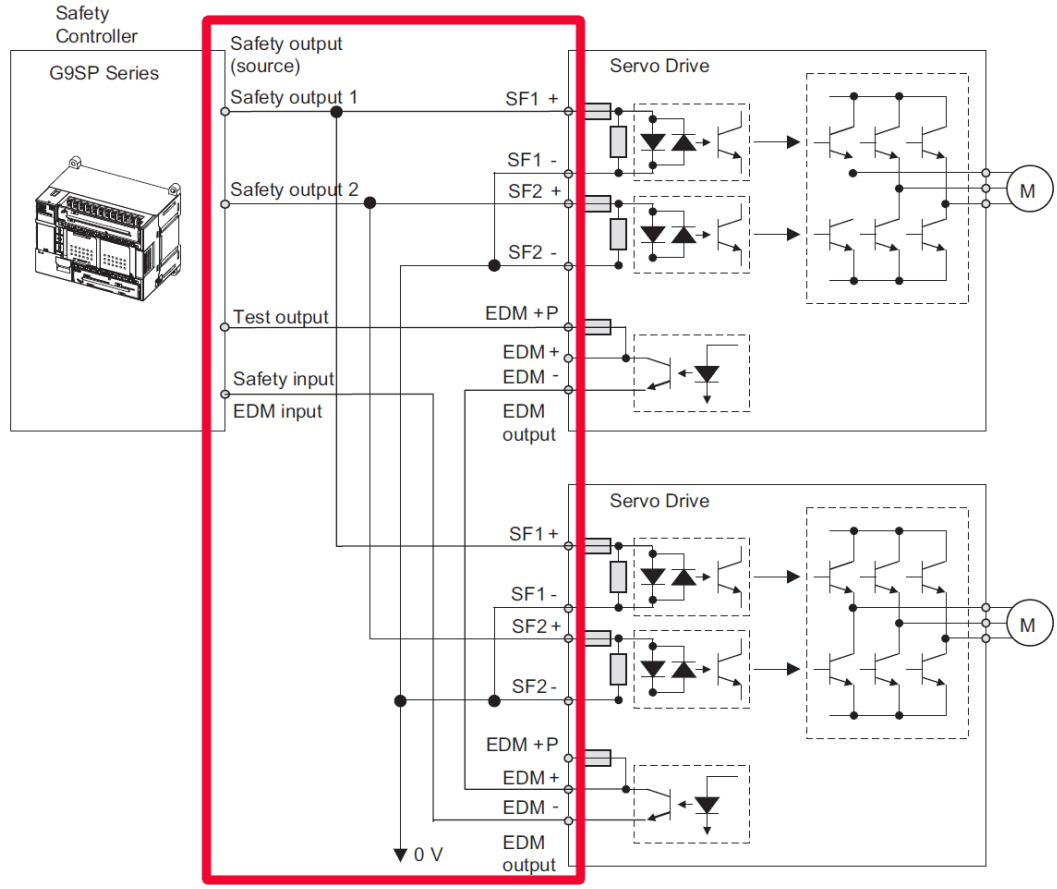
2. When using STO, make the wiring between the safety controller and the Servo Drive as shown below.

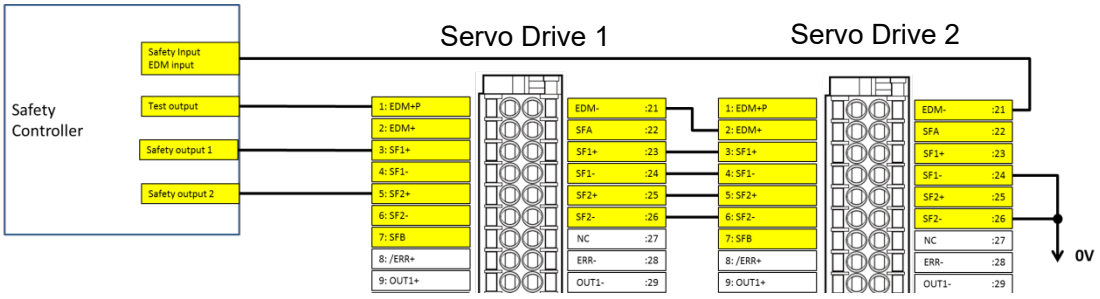


Wire SF1 and SF2 to different safety outputs.



3.	When using hard-wired STO for multiple Servo Drives, make the wiring between the safety controller and the Servo Drives as shown below.

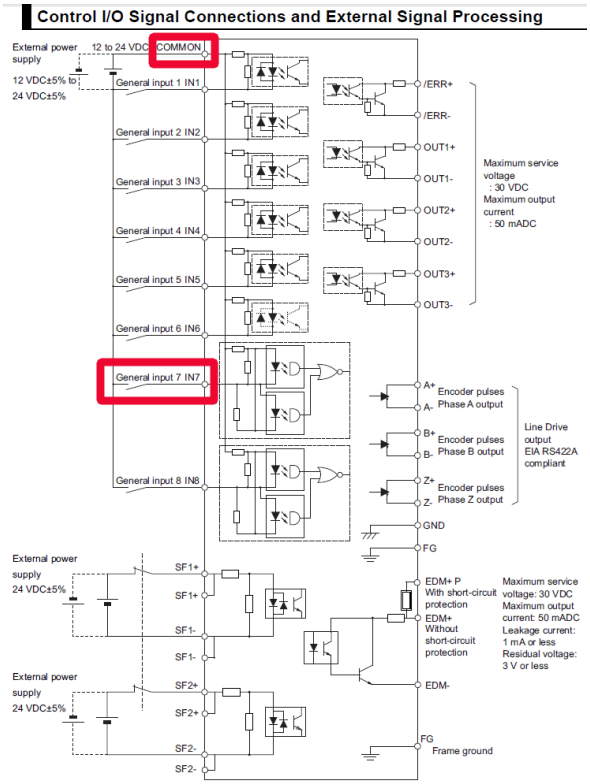




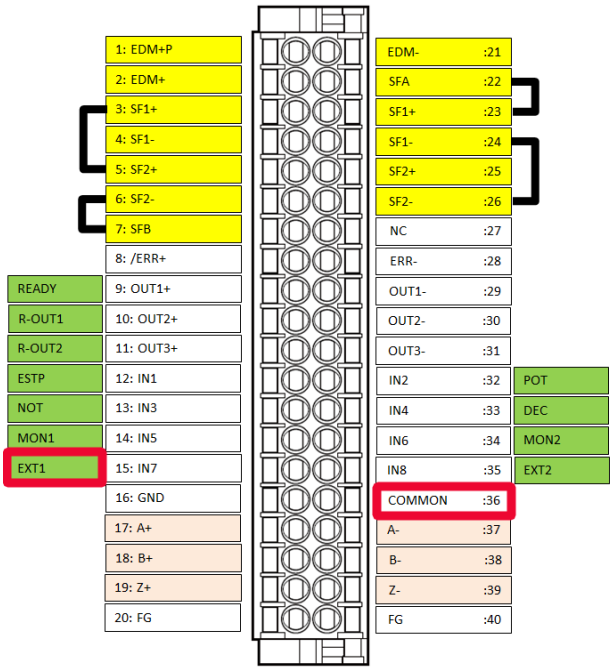
Note: When using a 69SP-series Safety Controller, you can connect up to four 1S-series Servo Drives.

4. When using general-purpose I/O, make the wiring as shown below.

Example of Latch Input 1:



Servo Drive connector view:

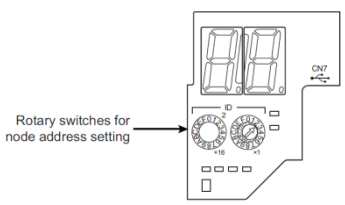
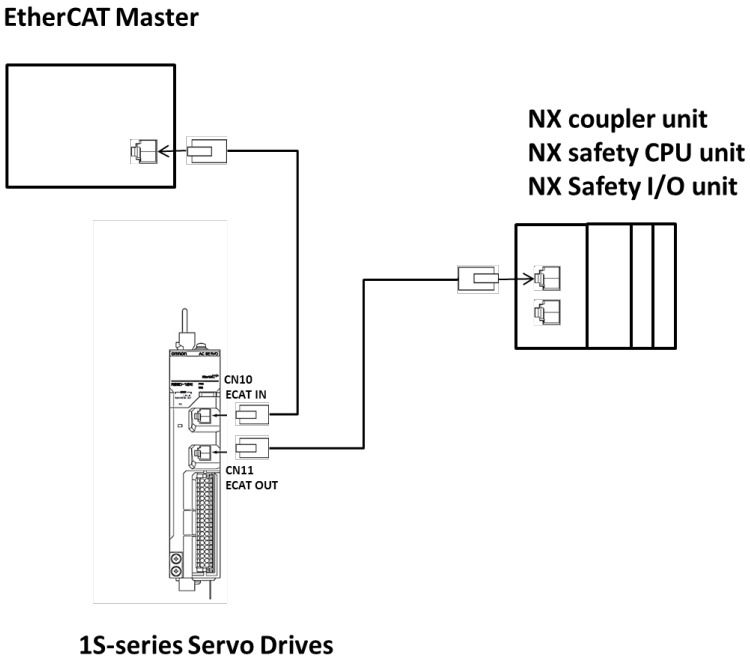


Additional Information

For further details on how to make the wiring, refer to *8-2 Safe Torque OFF (STO) Function* in the *1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual* (Cat. No. I696).

3.2. System Configuration with NX-series Safety Control Units

■ EtherCAT Node Address Configuration

1.	Use the ID switches to set the EtherCAT node address of the Servo Drive. (Example: Set to node 1)  
2.	When using the NX-series Safety Control Unit, set the node address of the coupler. (Example: Set to node 2)
3.	Connect an EtherCAT communications cable to the device.  ECAT IN CN10: EtherCAT cable from CK5M CPU Unit CN11 EtherCAT OUT: EtherCAT cable to coupler unit 

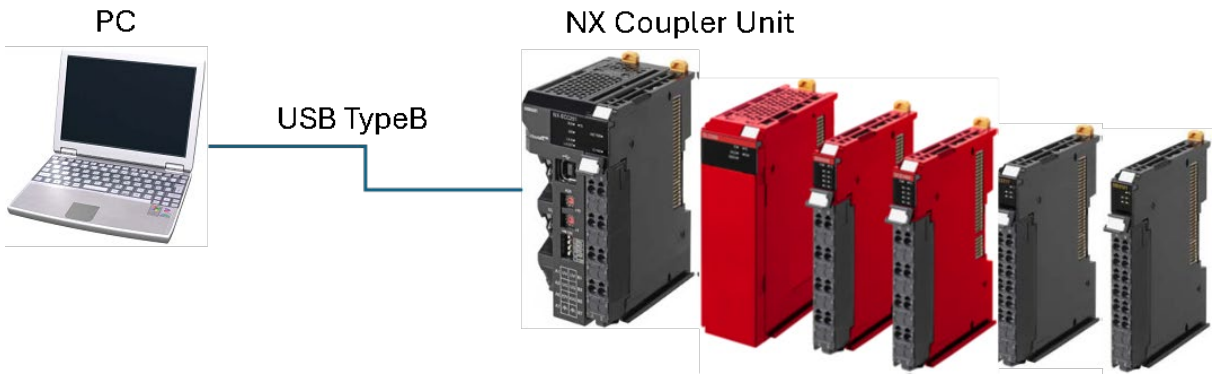
Additional Information

For further details on the safety controller, refer to the *NX-series Safety Control Unit User's Manual* (Cat. No. Z930).

3.3. Setup of NX-series Safety Control Units

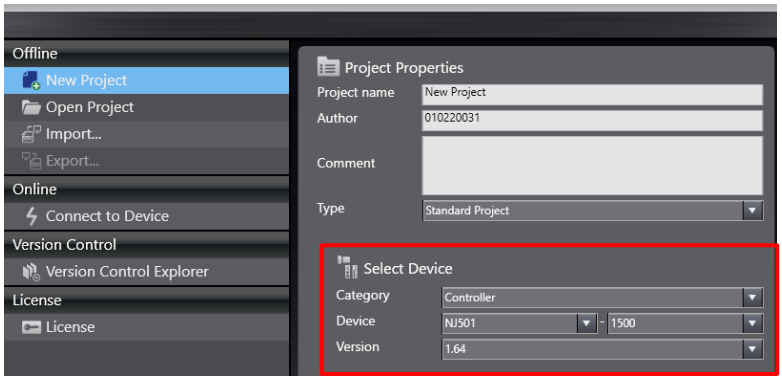
Set up the NX-series Safety Control Units using the Sysmac Studio.

To do so, connect the computer via USB to the coupler unit in the system configuration described in 3.2 System Configuration with NX-series Safety Control Units, as shown below.



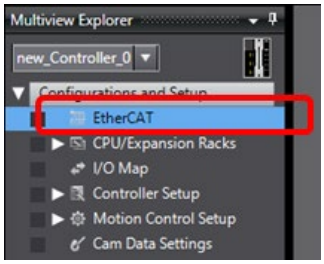
3.3.1. Creating a Network Configuration

1. Select the NJ501-1500 Controller from Select Device.

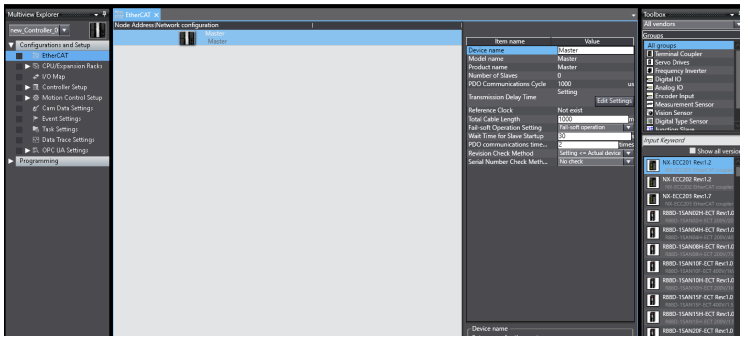


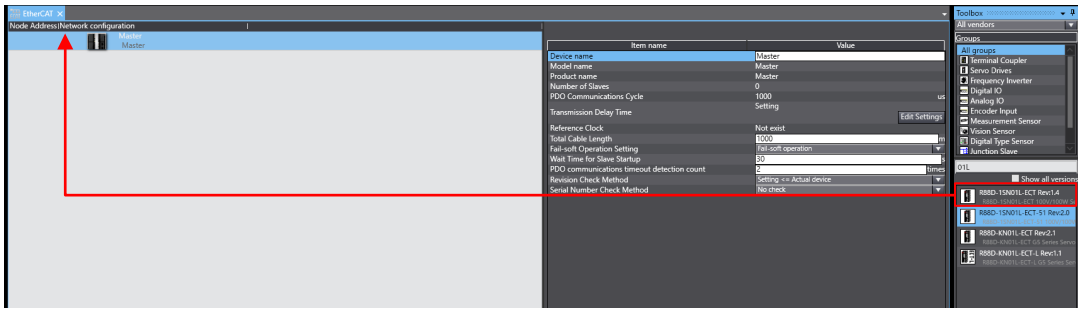
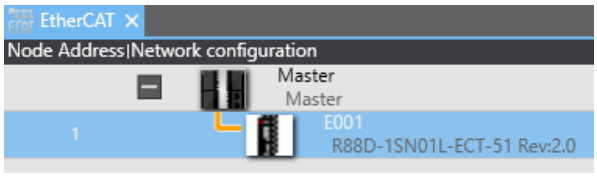
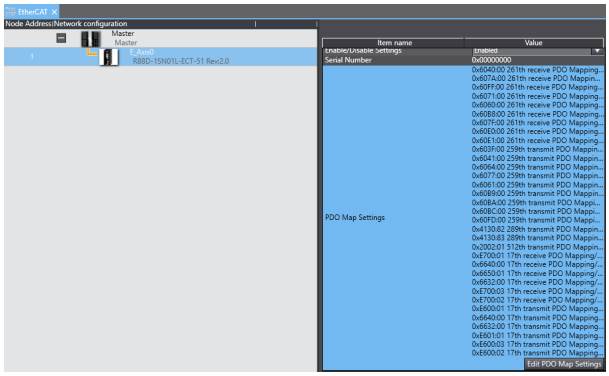
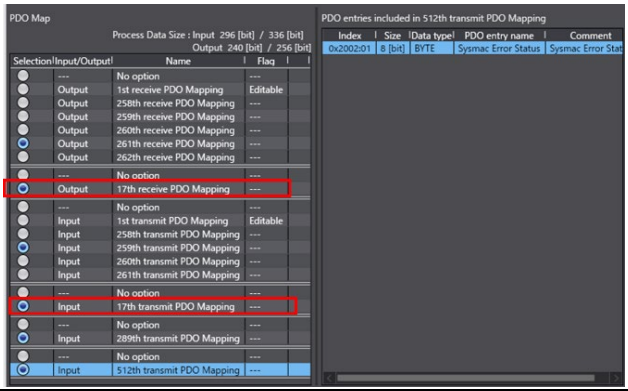
Note: Create a project using the default settings for **Device** and **Version**, as these settings do not affect the configuration being created.

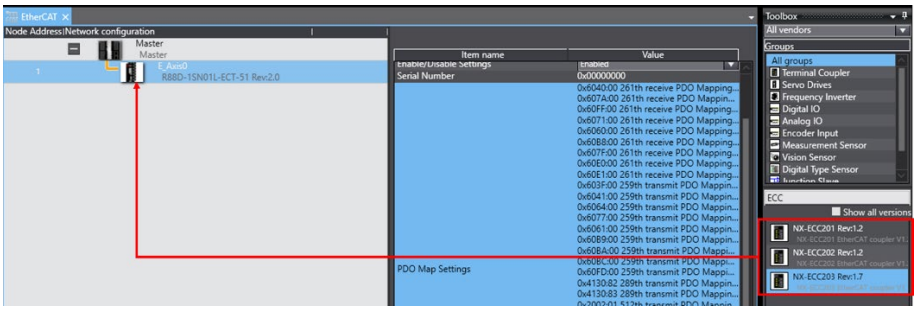
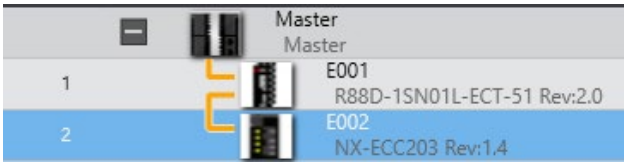
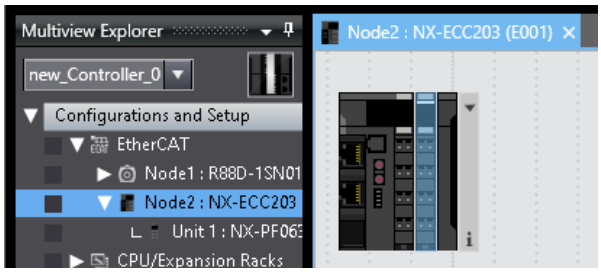
2. Double-click *EtherCAT* under *Configurations and Setup* in the Multiview Explorer.

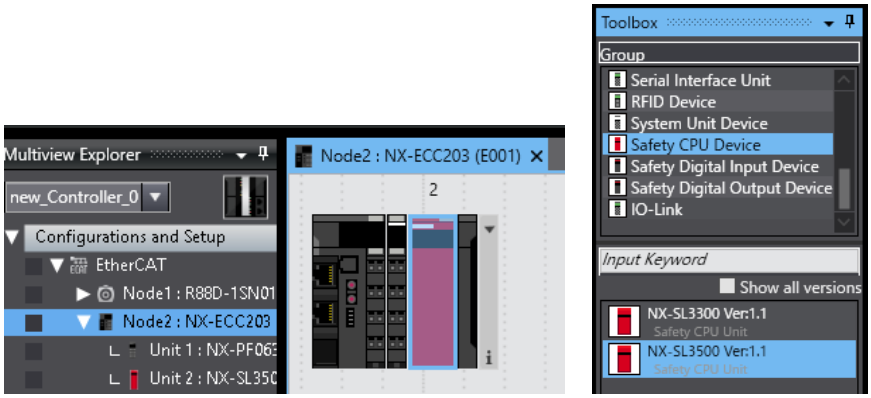
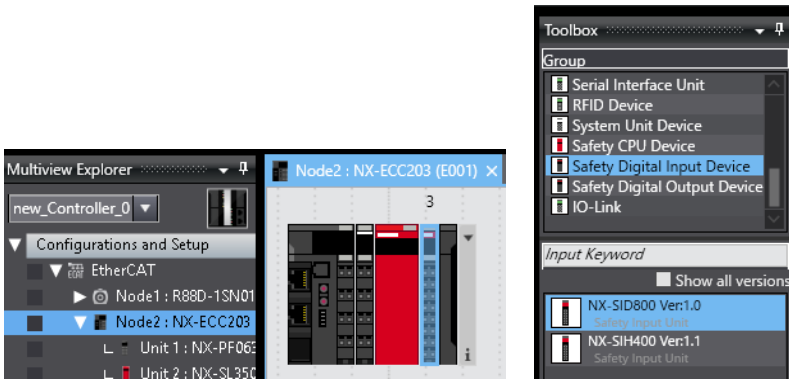
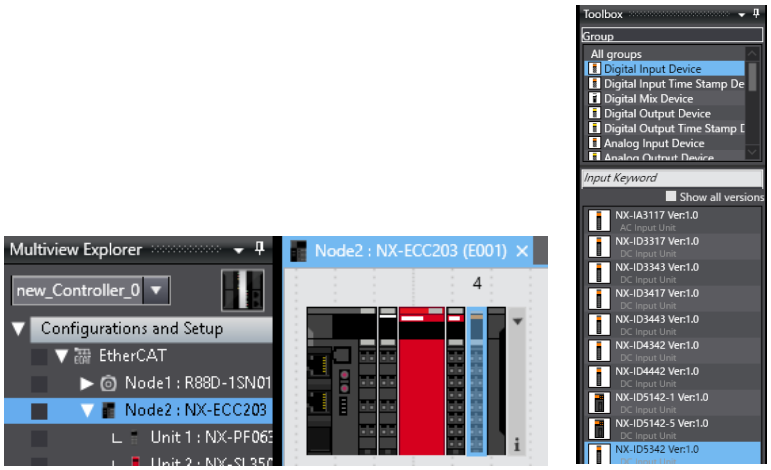


The EtherCAT Tab Page is displayed in the Edit Pane.



3.	Drag a Servo Drive from the Toolbox to the Master in the EtherCAT Tab Page.  The Servo Drive with a node address of 1 is added under the Master. 
4.	Select the Servo Drive and click the *Edit PDO Map Settings* Button. Note: Make sure to go offline before editing PDO mappings. 
5.	Select the 17th safety I/O. Note: PDOs for safety I/O are assigned by default. 

6.	Then, drag an EtherCAT coupler from the Toolbox to the Master in the EtherCAT Tab Page.  The EtherCAT couple with a node address of 2 is added under the Master. 
7.	Double-click *Node1: NX-ECC203*. 
8.	Drag a system unit device to the location where you want to add it in the CPU and Expansion Racks Tab Page. In this example, drag the NX-PF0630 Additional I/O Power Supply Unit.  

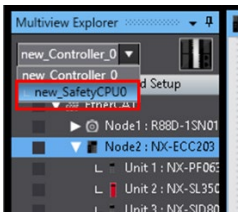
9.	Drag an NX-series safety CPU device to the location where you want to add it in the CPU and Expansion Racks Tab Page. In this example, drag the NX-SL3500 Safety CPU Unit. 
10.	Drag a safety digital input device to the location where you want to add it in the CPU and Expansion Racks Tab Page. In this example, drag the NX-SID800 Safety Input Unit. 
11.	Drag a digital input device to the location where you want to add it in the CPU and Expansion Racks Tab Page. In this example, drag the NX-ID5342 Digital Input Unit. 

Additional Information

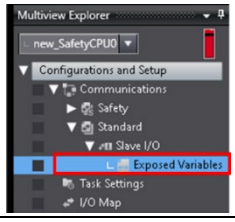
If the physical EtherCAT network configuration is already connected, you can automatically create the virtual network configuration on the Sysmac Studio from the physical network configuration.
Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for the procedure.

3.3.2. Setting to Transfer Data from the Standard Controller to the Safety Controller

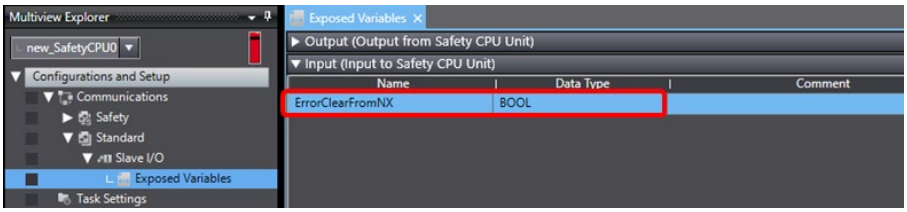
1. Select *new_SafetyCPU0* from the list.



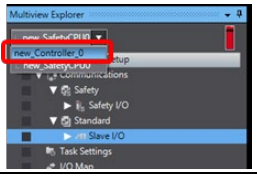
2. Double-click *Exposed Variables*.



3. Add the *ErrorClearFromNX* BOOL variable to Input (Input to the Safety CPU Unit).

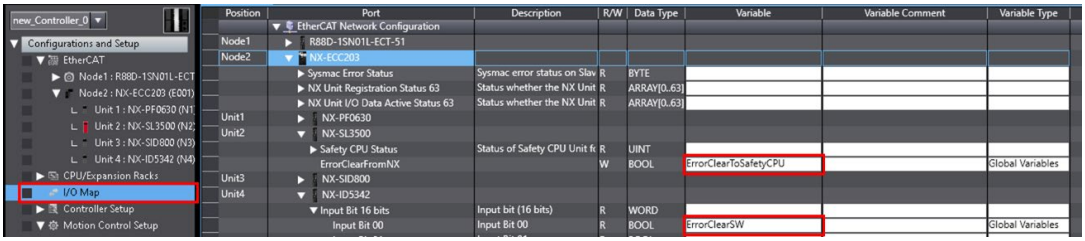


4. Select *new_Controller_0* from the list.



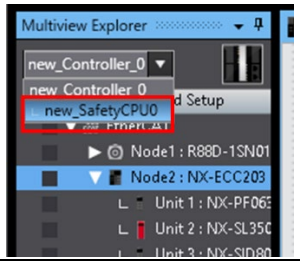
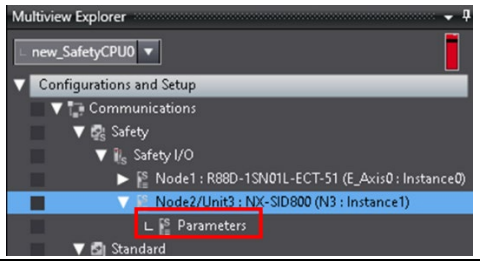
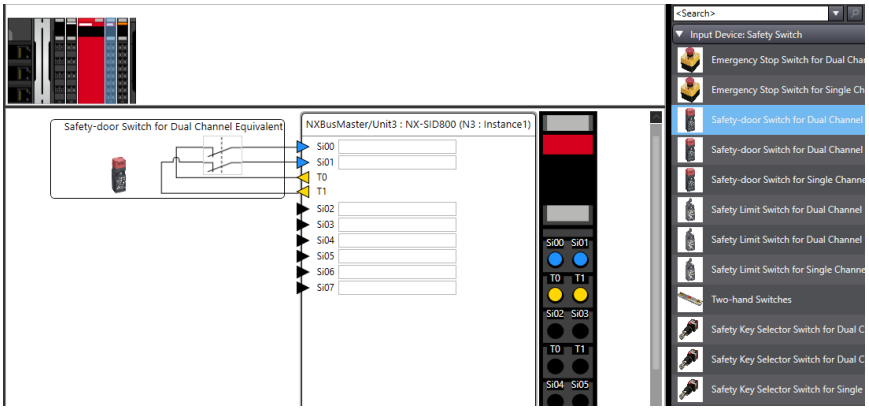
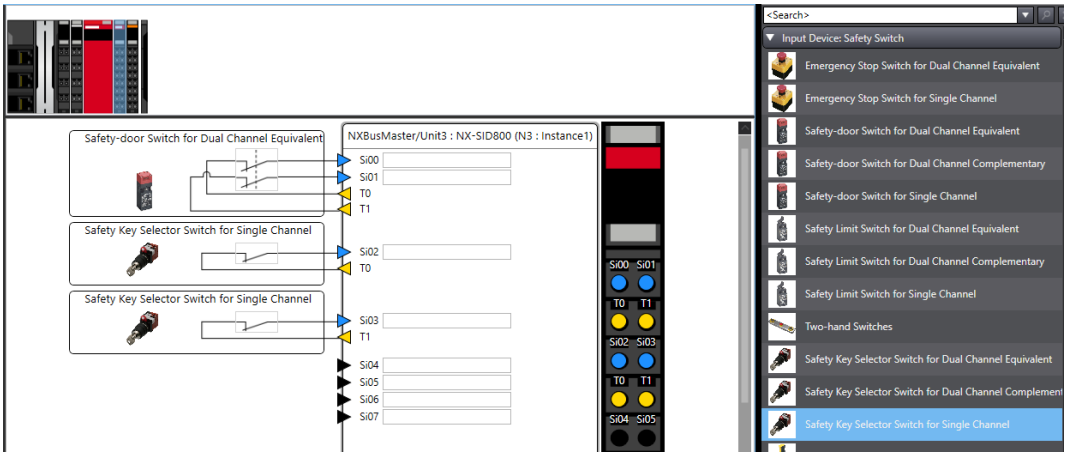
5. Open the I/O Map and create device variables.

Port	Variable name
ErrorClearFromNX	ErrorClearToSafetyCPU
Input Bit 00	ErrorClearSW

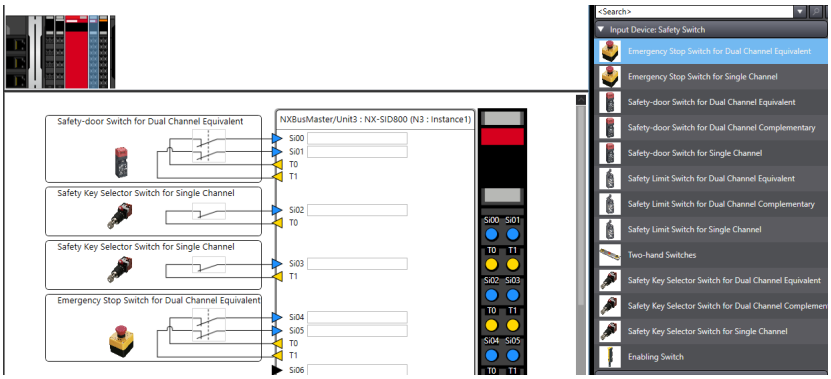


3.3.3. Setting the Safety Controller

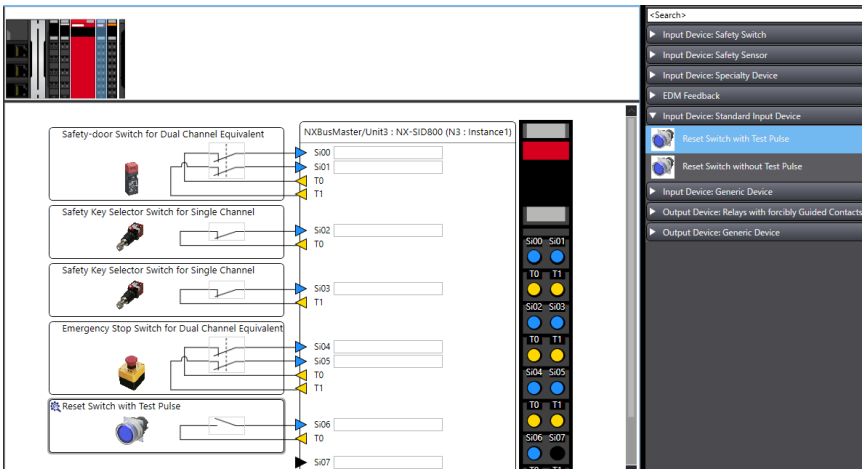
This section describes how to set safety input devices and create a safety program to release STO.

1.	Select *new_SafetyCPU0* from the list. 
2.	Double-click *Parameters*. 
3.	Drag a safety-door switch to the desired I/O terminal. 
4.	Drag two safety key selector switches for single channel to the desired I/O terminals. Note: You will use them in section 4 and later. 

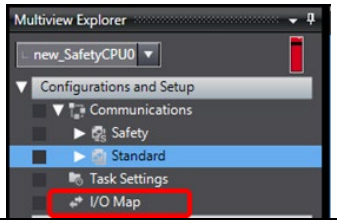
5. Drag an emergency stop pushbutton switch to the desired I/O terminal.



6. Drag a reset switch to the desired I/O terminal.

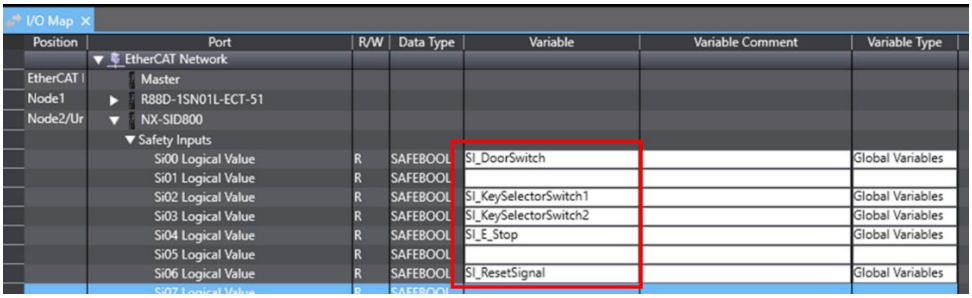


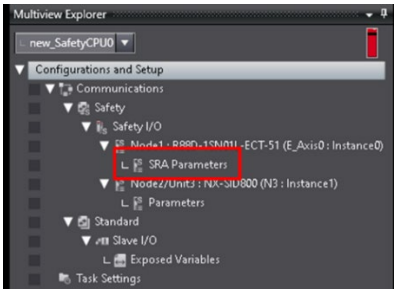
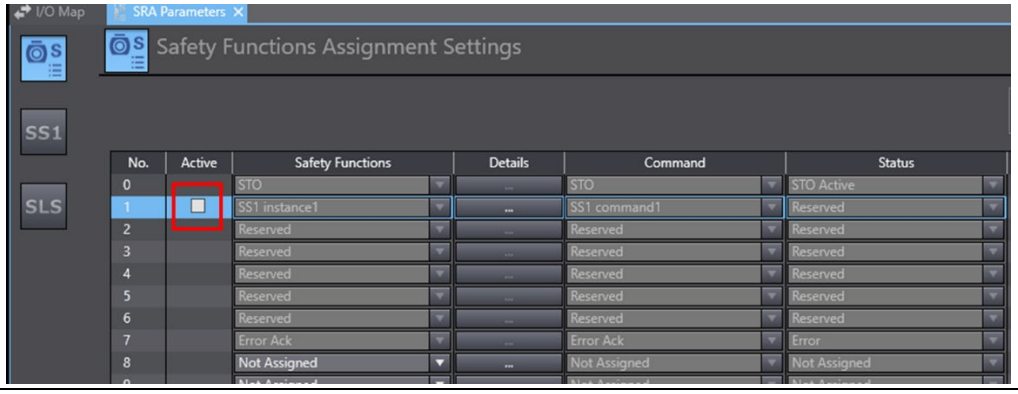
7. Double-click *I/O Map*.



8. Create device variables for the safety input devices.

Port	Variable name
Si00 Logical Value	SI_DoorSwitch
Si02 Logical Value	SI_KeySelectorSwitch1
Si03 Logical Value	SI_KeySelectorSwitch2
Si04 Logical Value	SI_E_Stop
Si06 Logical Value	SI_ResetSignal



9.	Double-click *SRA Parameters*. 
10.	Deactivate safety functions except for the STO function. Clear the *Active* Check Box for SS1 instance1 to deactivate the unused safety function. 

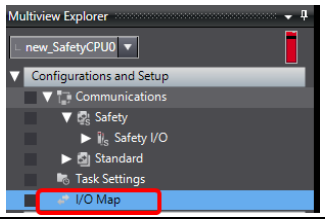
3.3.4. Creating a Safety Program

■ Create a safety program by following the steps below:

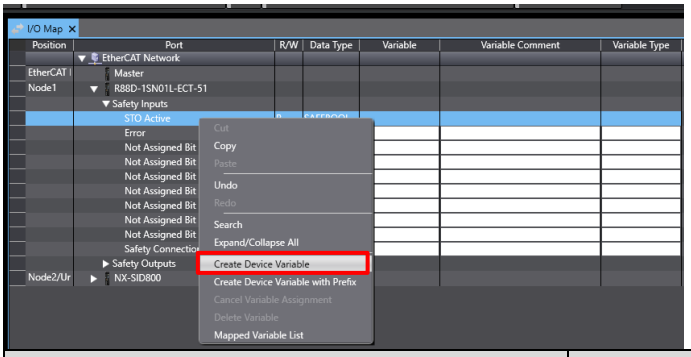
1. Create device variables.
2. Create a safety program using the Automatic Programming function.
3. Modify the created safety program.

1. Create device variables.

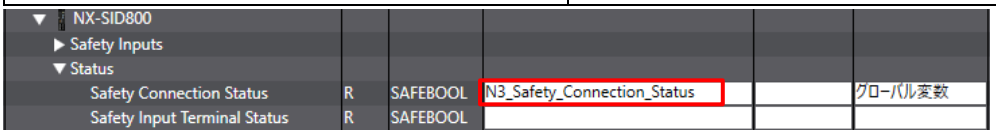
1. Select *new_SafetyCPU0* from the list and double-click *I/O Map*.

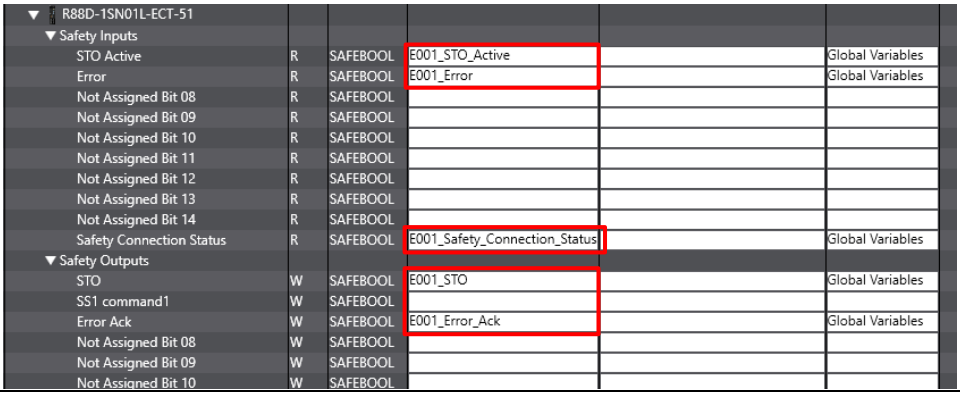


2. Create device variables by right-clicking the port to set and clicking *Create Device Variable* from the menu.

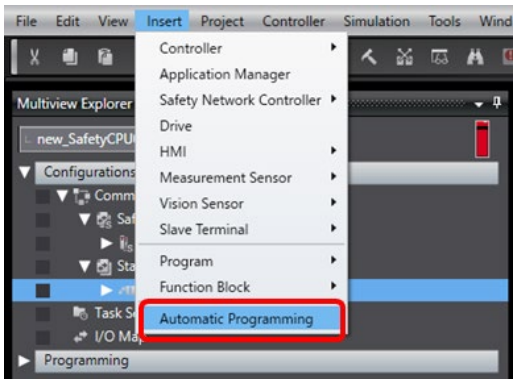
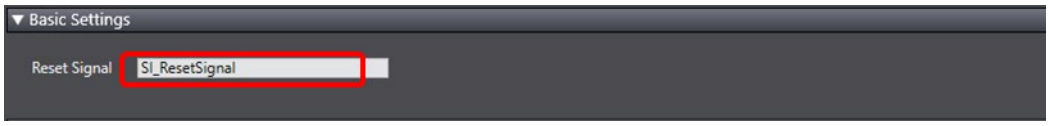
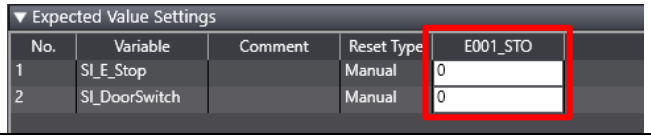
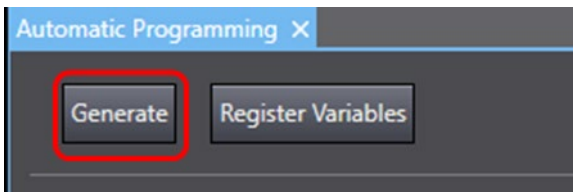
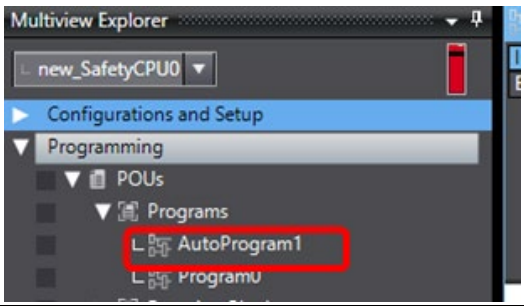


Port	Variable name
Safety Connection Status for NX-SID800	N3_Safety_Connection_Status
STO Active for R88D-1SN01L-ECT-51	E001_STO_Active
Error for R88D-1SN01L-ECT-51	E001_Error
Safety Connection Status for R88D-1SN01L-ECT-51	E001_Safety_Connection_Status
STO for R88D-1SN01L-ECT-51	E001_STO
Error Ack for R88D-1SN01L-ECT-51	E001_Error_Ack



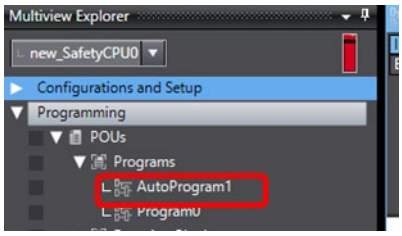


2. Create a safety program using the Automatic Programming function.

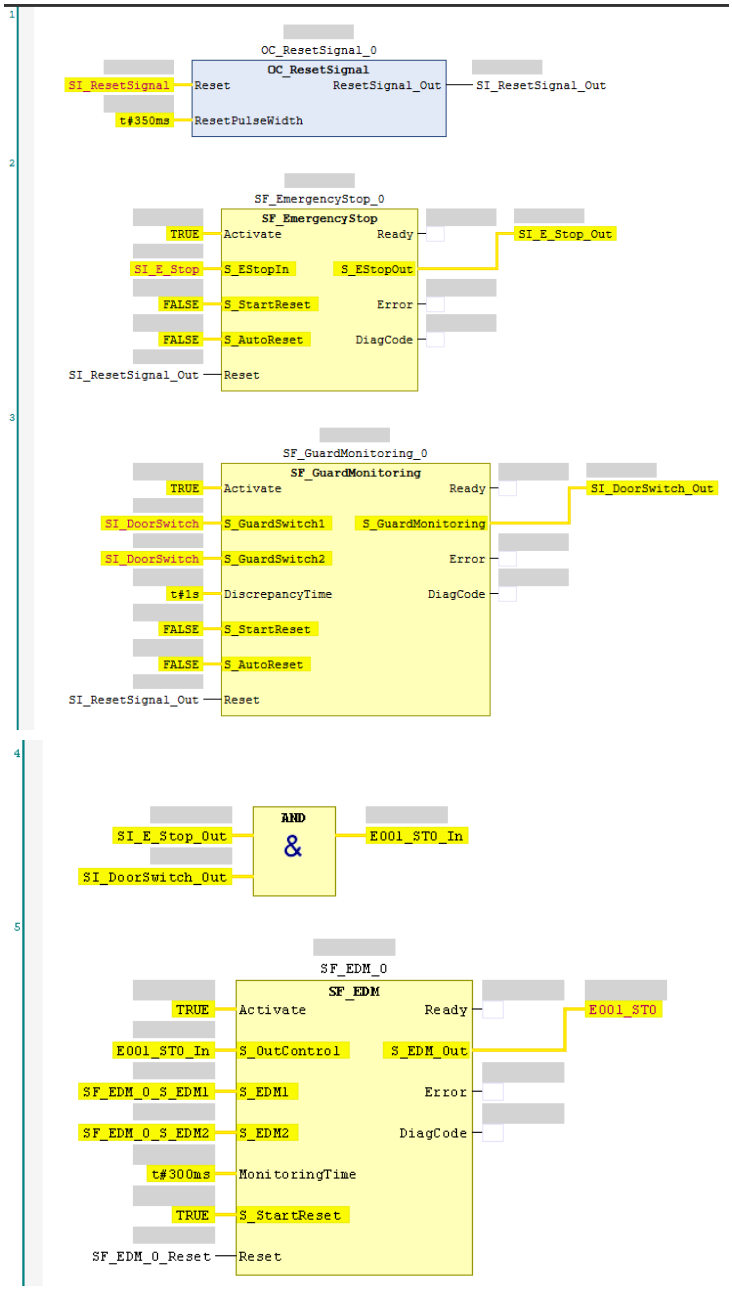
1.	Click *Automatic Programming* from the Insert Menu. 
2.	Set the reset signal to *SI_ResetSignal* in the Basic Settings Field. 
3.	Set *SI_E-Stop* and *SI_DoorSwitch* in the Input Settings Field. 
4.	Set *E001_STO* in the Output Settings Field. Set the *Use EDM* Column to *TRUE*. 
5.	Set the *E001_STO* Column to *0* for the *SI_E-Stop* and *SI_DoorSwitch* variables in the Expected Value Settings Field. 
6.	Click the *Generate* Button to create a safety program.  AutoProgram1 is created. 

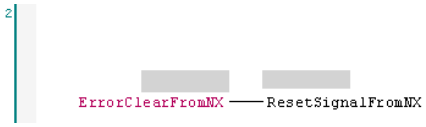
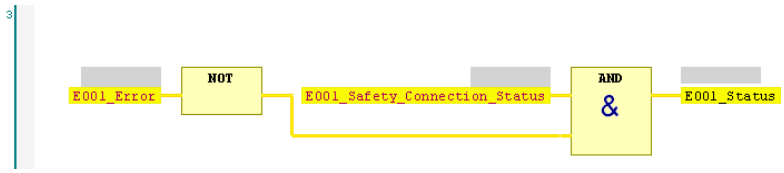
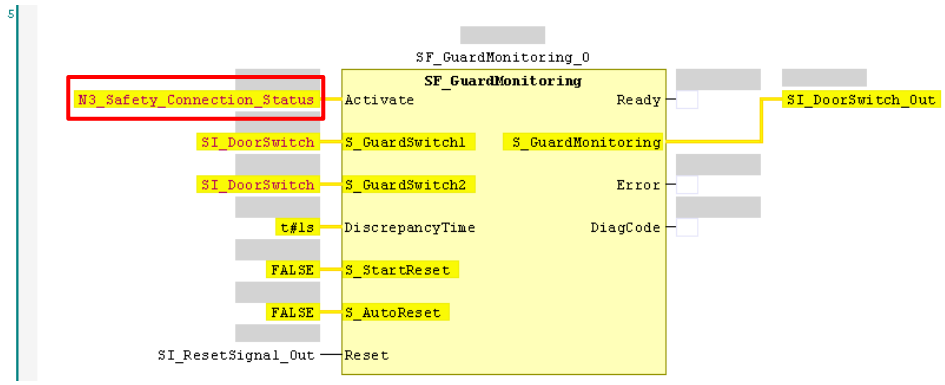
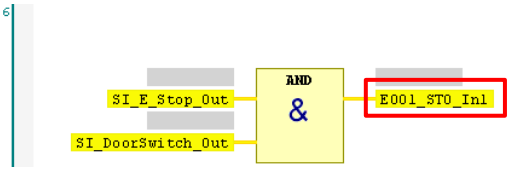
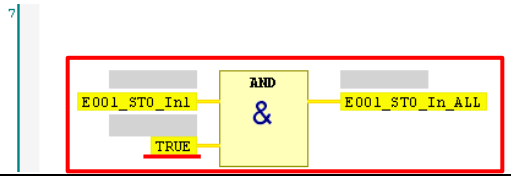
3. Modify the created safety program.

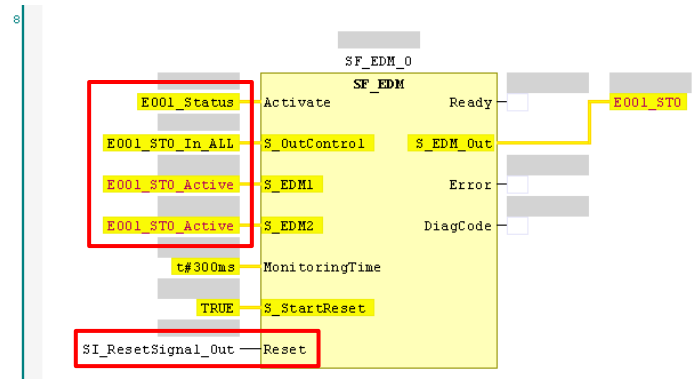
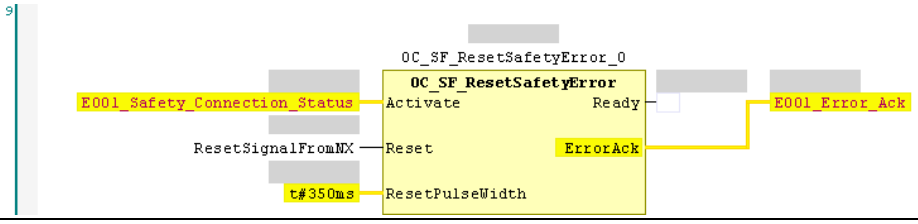
1. Double-click *AutoProgram1*.



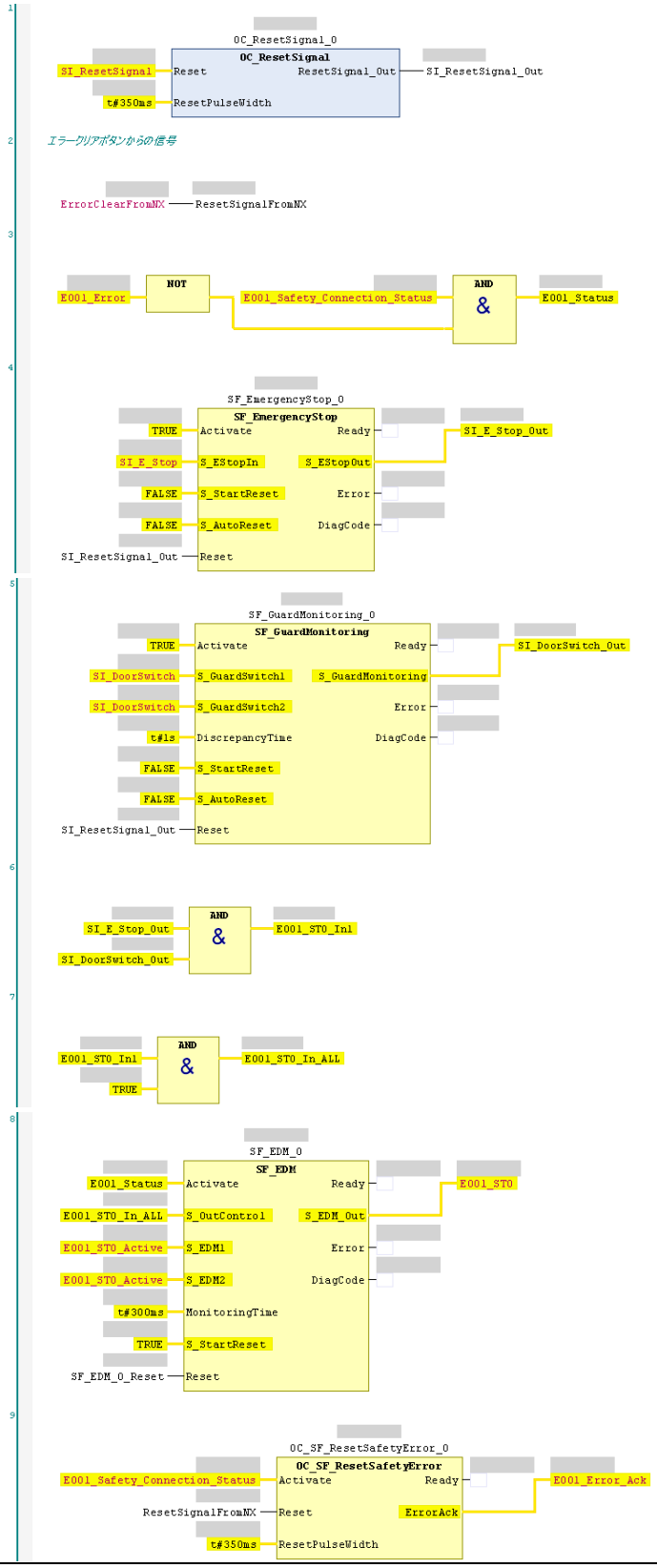
The program shown below appears.

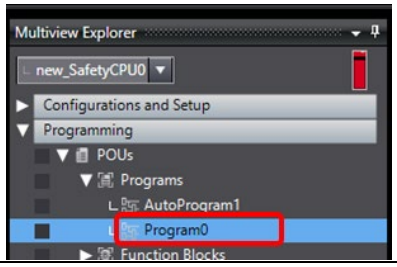
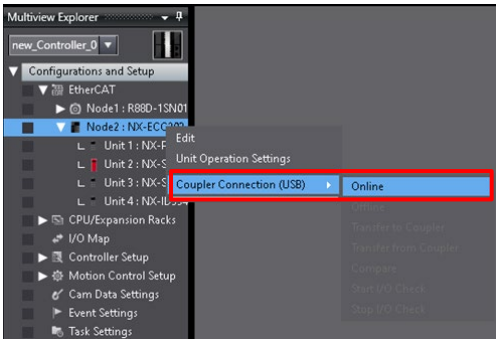
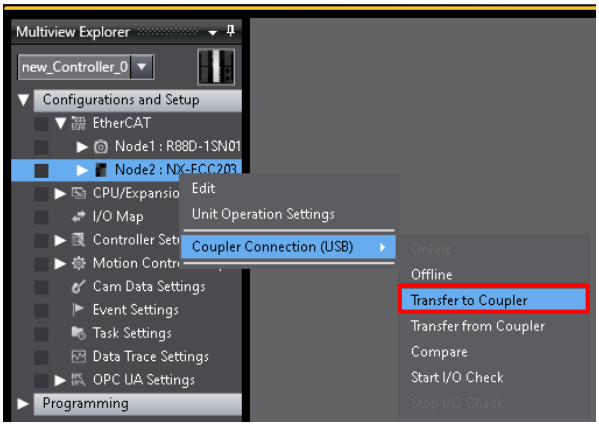
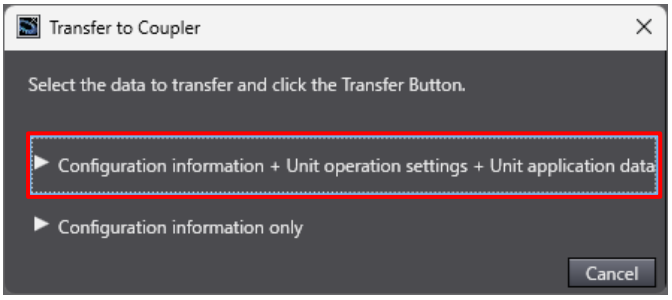
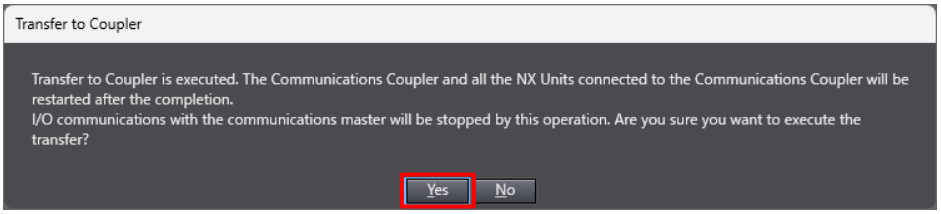


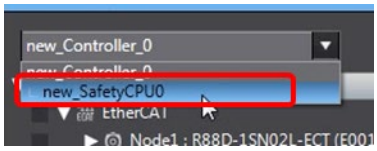
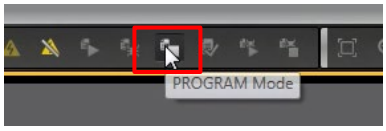
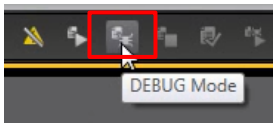
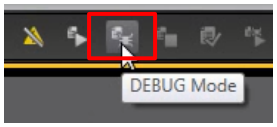
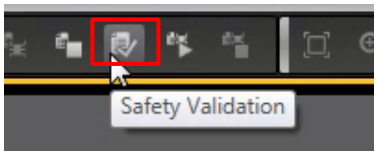
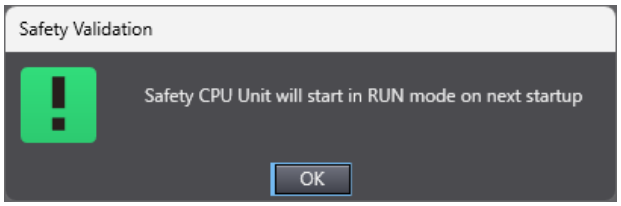
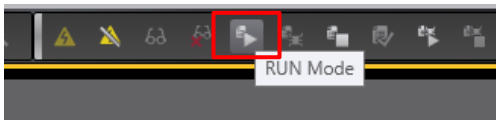
2.	Create the code to assign *ErrorClearFromNX* to the *ResetSignalFromNX* local variable. 
3.	Add the following code. This is used to enable the function block that uses safety functions of the Servo Drive. 
4.	Set the input parameter to the *Activate* input variable to *N3_Safety_Connection_Status* in the SF_GuardMonitoring function block . When the FSoE communications are established for the NX-SID800 Safety Input Unit, this function block is enabled. 
5.	Change the output variable from *E001_STO_In* to *E001_STO_In1*. 
6.	Create the following code. The STO command is enabled by the emergency stop input or a safety-door switch, and sometimes by the SS1 command or another device such as a robot. When the other device enables the STO command, assign the STO signal from the other device to *TRUE*. 

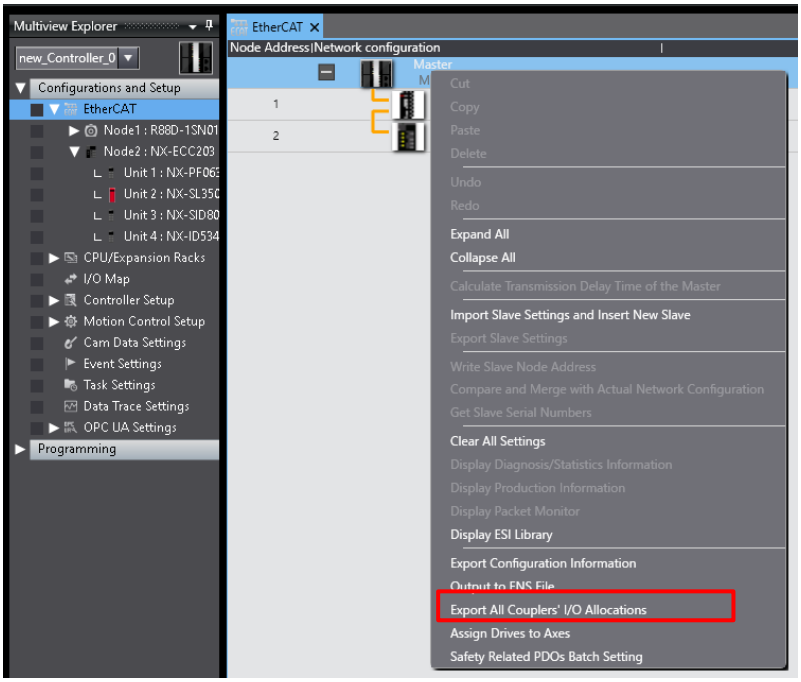
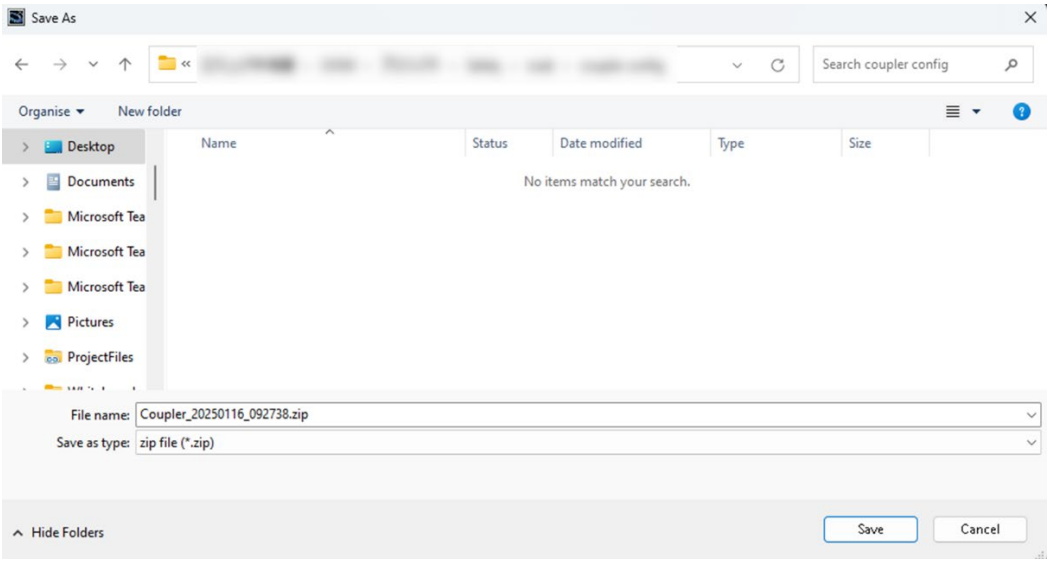
7.	Set the input parameter to the *Activate* input variable to *E001_Status*, the input parameter to the *S_OutControl* to *E001_STO_In_ALL*, the input parameter to the *Reset* to *SI_ResetSignal_Out*, and the input parameters to both *S_EDM1* and *S_EDM2* to *E001_STO_Active* in the SF_EDM function block. When the FSoE communications are established between the safety controller and Servo Drive and there is no error of safety functions in the Servo Drive, this function block is enabled. 
8.	Create the code to reset errors of safety functions in the Servo Drive. 

9.	Check that the created program is the same as shown below.



10.	Delete *Program0*. 
11.	Connect to the NX-series Coupler Unit. 
12.	Select *Transfer to Coupler*.  Select *Configuration information + Unit operation settings + Unit application data*.  Select *Yes* to transfer the data to the NX-series Coupler Unit. 

13.	Download the safety application. Select *new_SafetyCPU0* from the list.  Click the **PROGRAM Mode** Button to switch to PROGRAM mode.  Click the **DEBUG Mode** Button to enter DEBUG mode.  Click the **DEBUG Mode** Button to start DEBUG mode.  Click the **Safety Validation** Button.  The safety application is now ready to run.  Click the **RUN Mode** Button. 

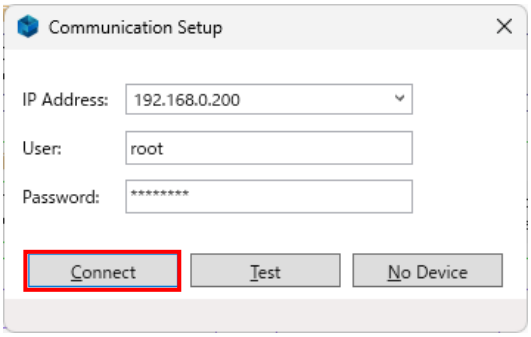
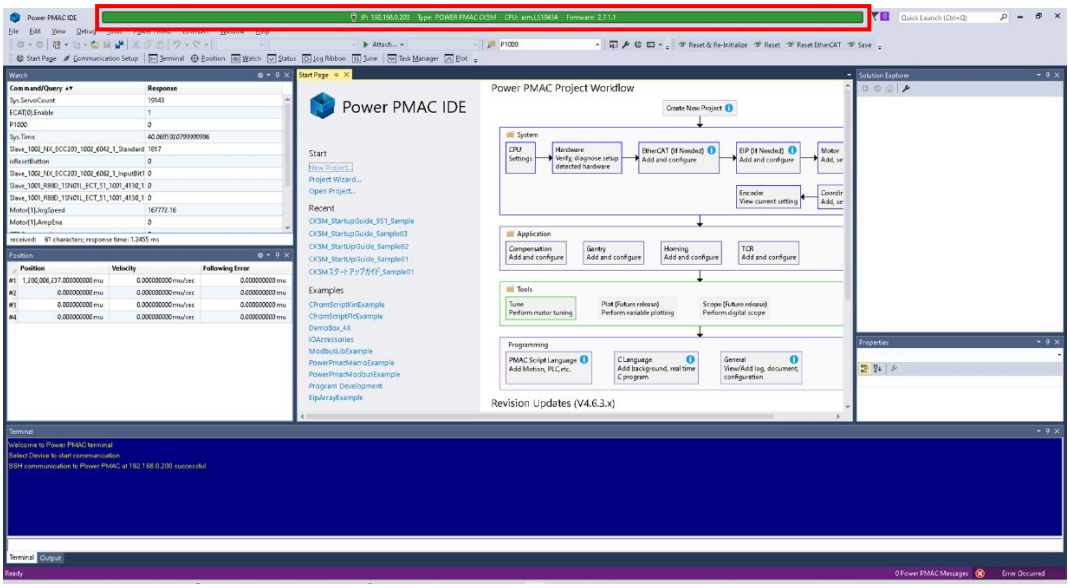
14.	When the data is successfully written, disconnect the coupler unit. Select *EtherCAT*, right-click the Master and select *Export All Couplers' I/O Allocations* from the menu.  Save the exported data file in any folder.  Note: You will use the exported file in the next section and later.

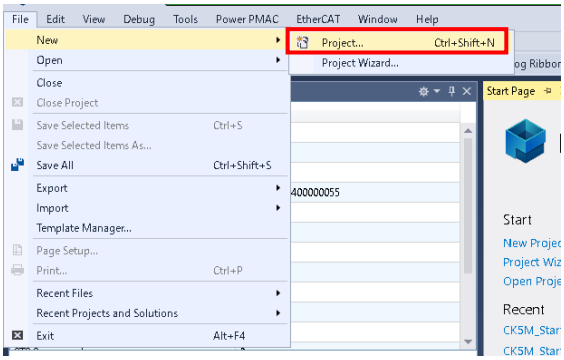
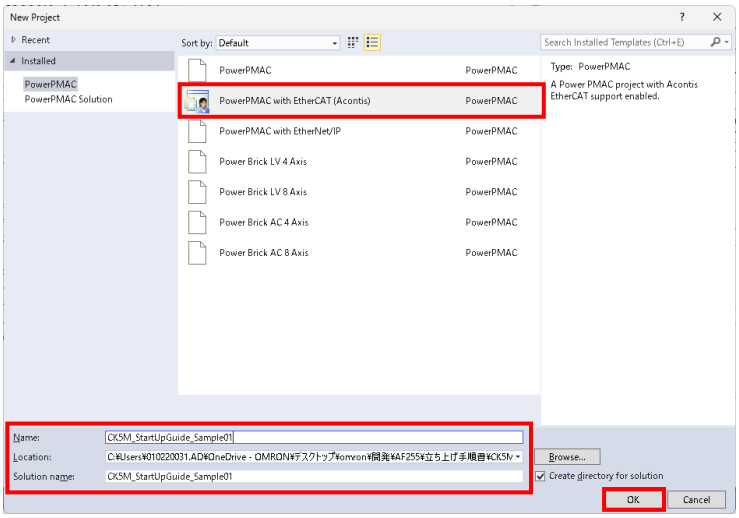
3.4. Setup of the Motion Controller CK5M

To prepare for setting up the CK5M, use the Power PMAC IDE.
Install the Power PMAC IDE on the computer in advance, with reference to 2. Before You Begin.

3.4.1. Preparing the Controller for Setup

■ Creating a New Project

1.	Turn ON the power supply to the Controller.
2.	Start the Power PMAC IDE.  Note: If a dialog box for access right confirming is displayed at startup, make selections to start the Power PMAC IDE.
3.	In the Communication Setup Dialog Box displayed, specify the IP address of the Controller to connect to and click the *Connect* Button.  The default IP address of the Controller is 192.168.0.200. If necessary, change the IP address in Windows to *192.168.0.X*.
4.	The Power PMAC IDE starts and goes online with the Controller. 

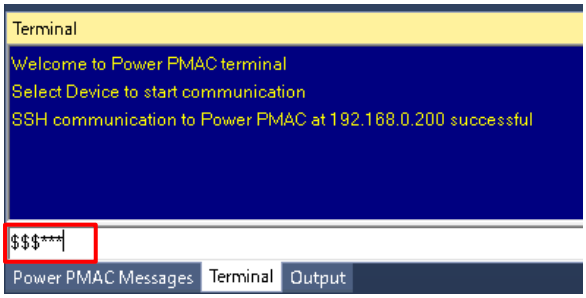
5.	Select *New - Project* from the File menu. 
6.	Click *PowerPMAC with EtherCAT (Acontis)*. Enter a project name and click the *OK* Button. 

■ Initially Setting Up the Controller

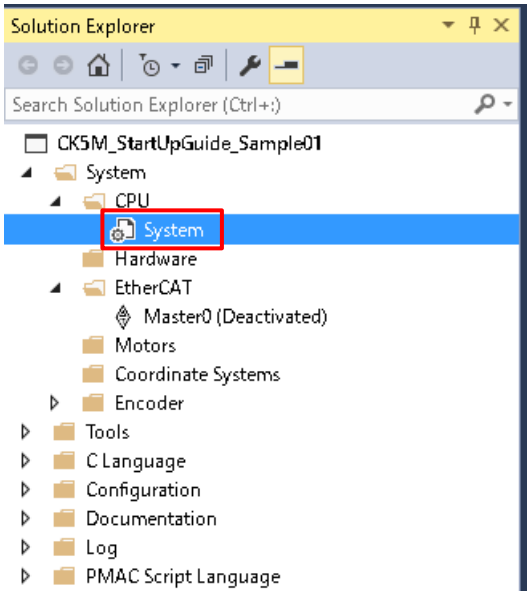
Perform the initial setup of the Controller.

Precautions for Correct Use

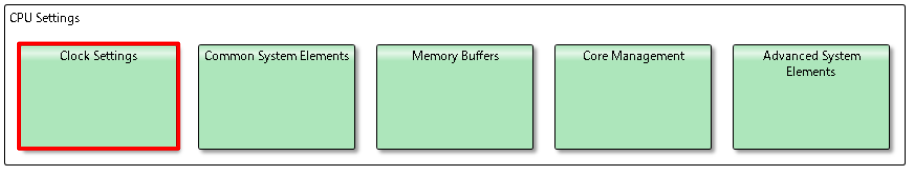
Save any necessary data in the Controller before proceeding, since the initial setup procedure clears all the memory.

1.	Execute the *$$$*** from the Terminal of the Power PMAC IDE to reset the Controller to the default settings. 

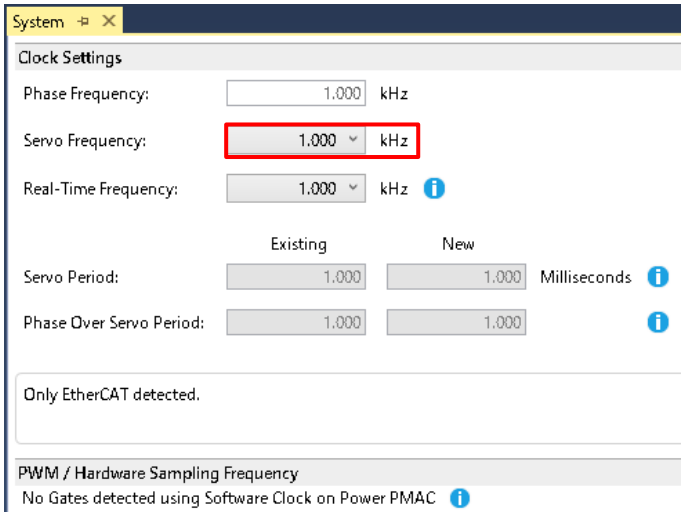
2. Open *System - CPU - System* from the Solution Explorer.



3. Click the *Clock Settings* Button.

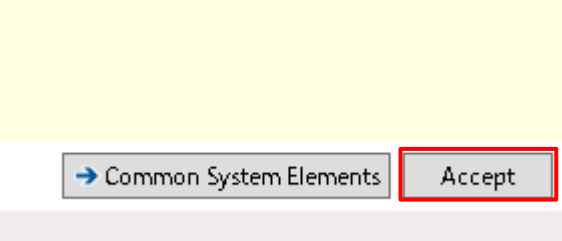
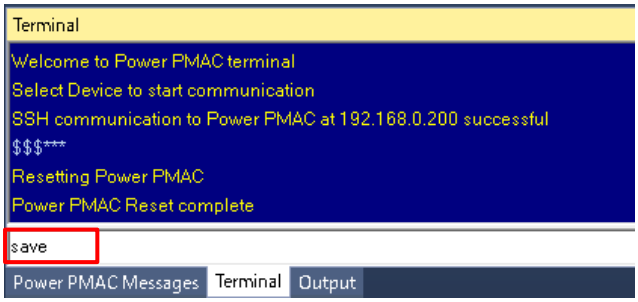
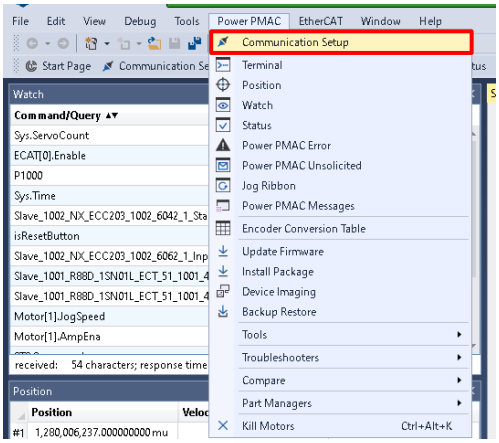
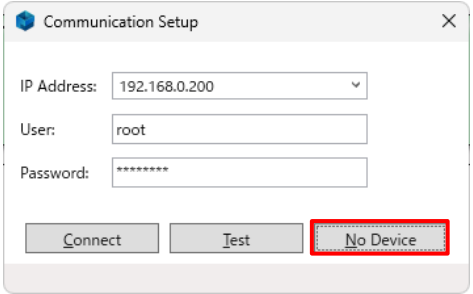


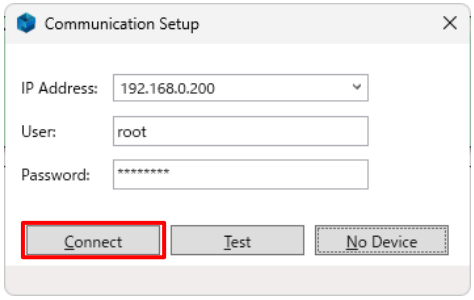
4. Set the *Servo Frequency* to 1 kHz.



Note: The servo frequency that you set here has the following relationship with the Servo Drive's communications cycle. Set the frequency according to the communications cycle you want to set.

Communication cycle [μs]	Servo Frequency[kHz]
1,000	1.0
500	2.0
250	4.0
125	8.0

5.	Click the *Accept* Button. 
6.	If you change the servo frequency setting, enter the *save* command from the Terminal of the Power PMAC IDE. When the command execution is completed, *Save Completed* is displayed in the Terminal. 
7.	Click the *Communication Setup* icon in the toolbar to display the Communication Setup Dialog Box. 
8.	Click the *No Device* Button in the *Communication Setup* Dialog Box.  The Controller goes offline.
9.	Cycle the power supply to the Controller. The set servo frequency will be reflected after the Controller restarts.

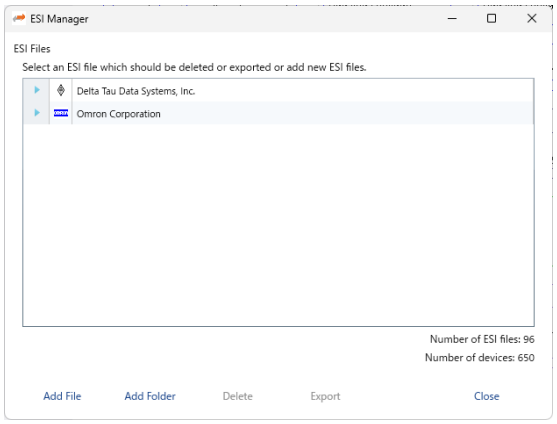
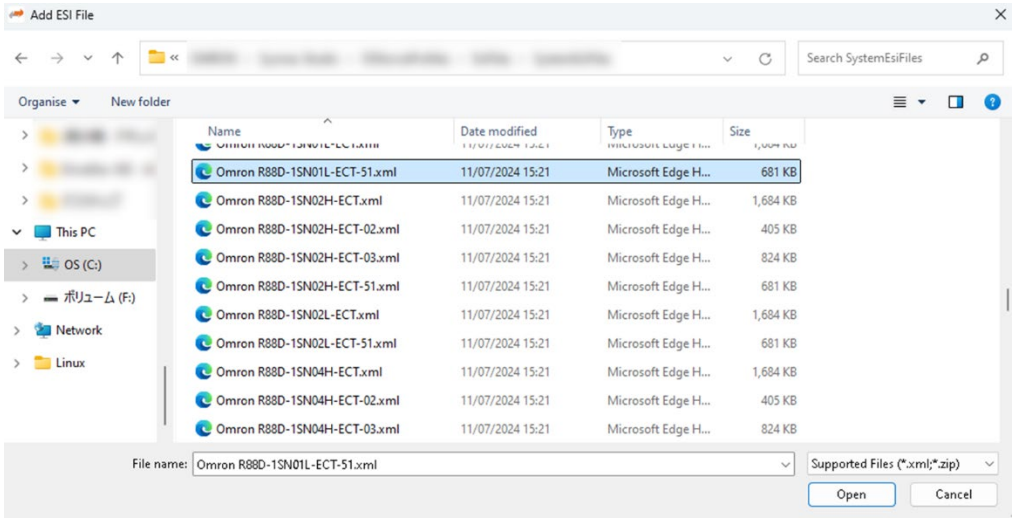
10.	Wait for the Controller to complete starting up, click *Communication Setup* in the toolbar to display the Communication Setup Dialog Box. Click the *Connect* button in the Communication Setup Dialog Box.  The Controller goes online.

3.4.2. Installing the ESI File

Install the ESI file for the Servo Drive in the Power PMAC IDE.

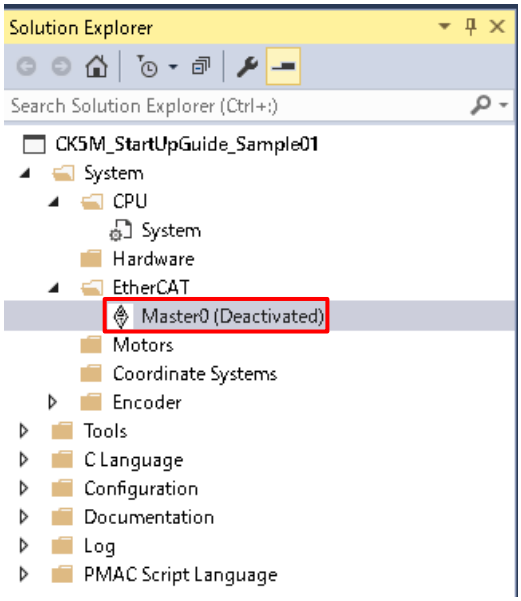
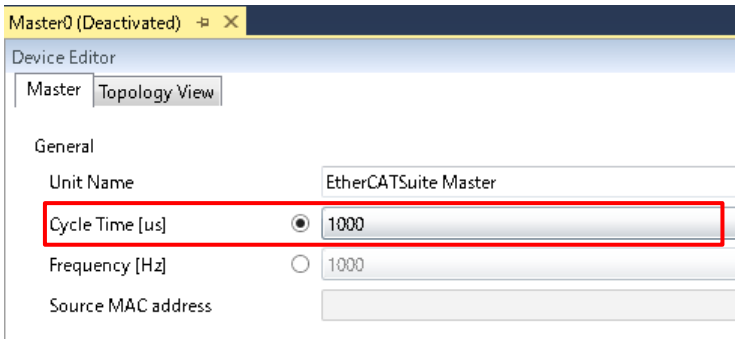
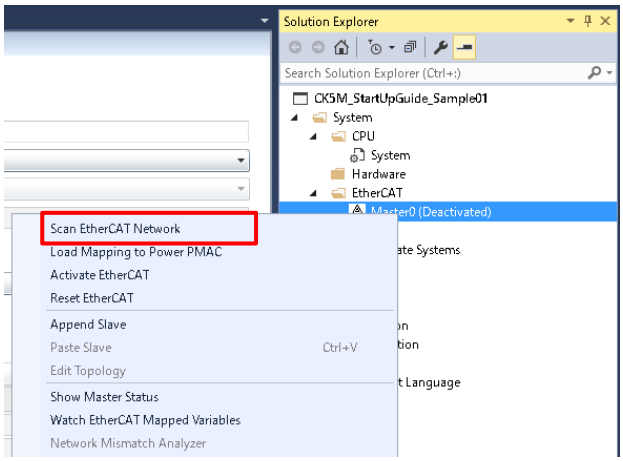
Precautions for Correct Use

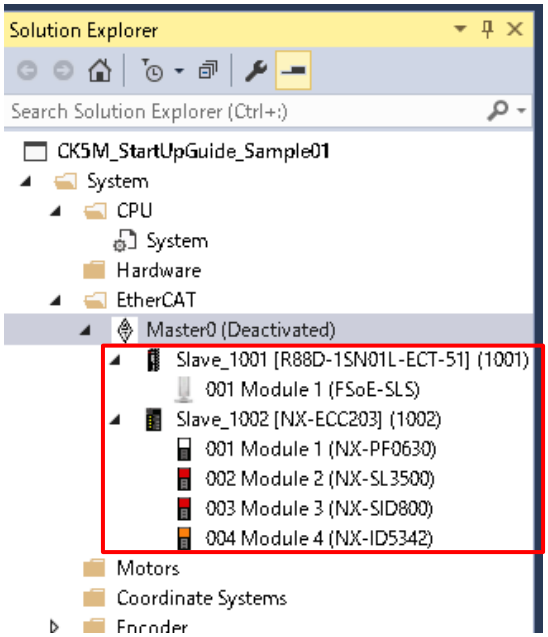
Prepare the ESI file mentioned in this section in advance. The ESI file can be downloaded from the OMRON website.

1.	Select *ESI Manager* from the EtherCAT Menu in the Power PMAC IDE. 
2.	Check if Omron R88D-1SN01L-ECT-51.xml is registered in the list of ESI files in the ESI Manager.  If it is not registered, click *Add File* and register Omron R88D-1SN01L-ECT-51.xml. 
3.	Click the *Close* Button to close the ESI Manager Pane.

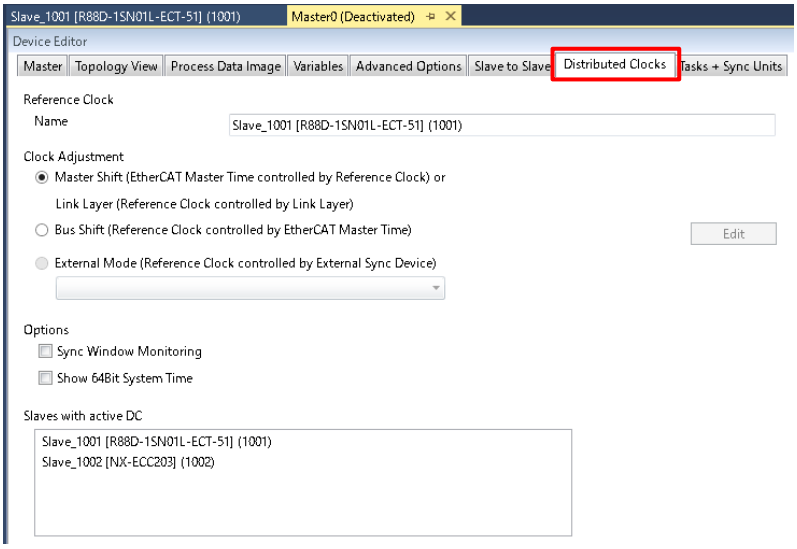
3.4.3. Configuring EtherCAT Communications Settings

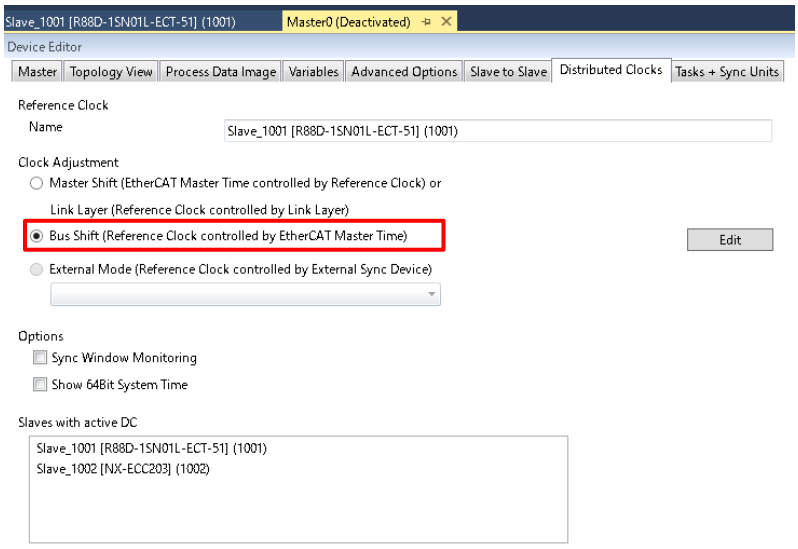
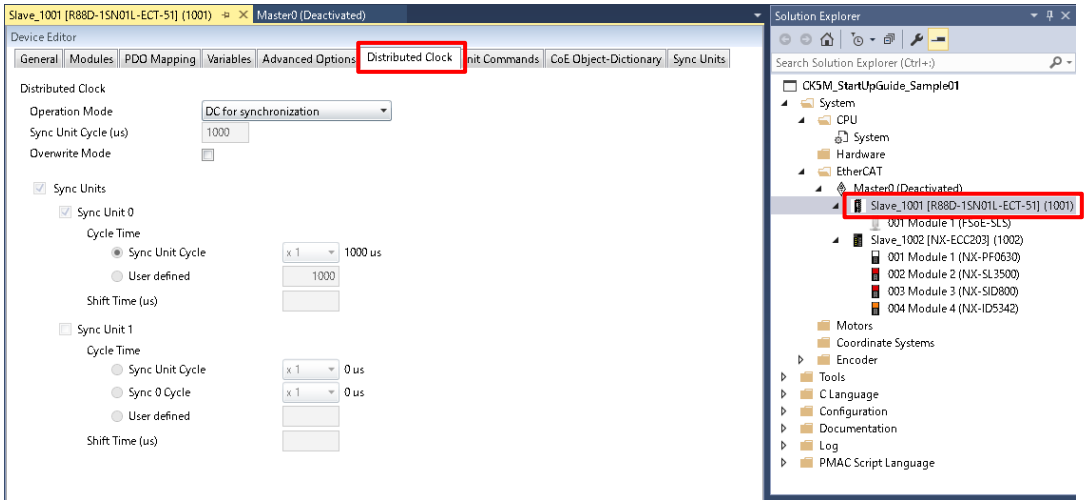
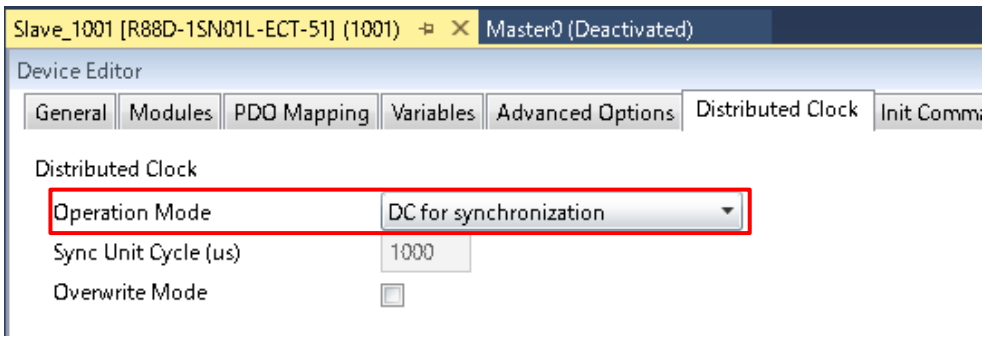
■ Configuring EtherCAT Master Communications Settings

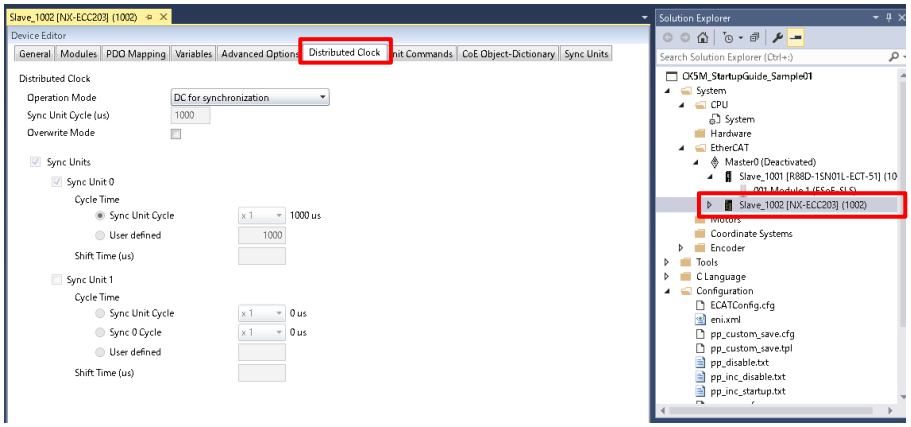
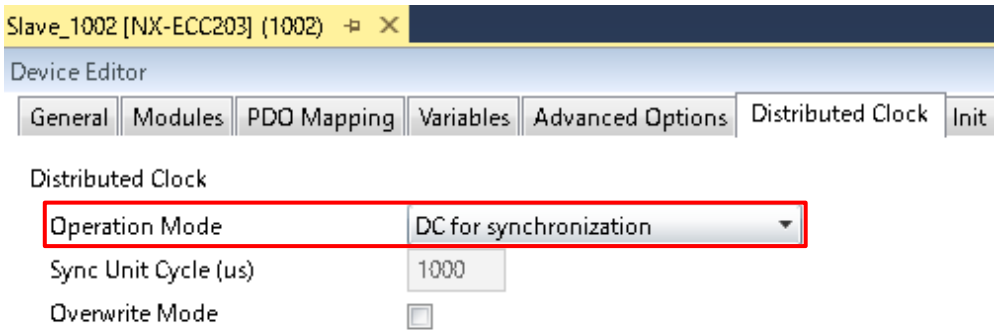
1.	Open *System - EtherCAT - Master0(Deactivated)* from the Solution Explorer. 
2.	Set the communications cycle in *Cycle Time [µs]* in the Master Tab Page.  Note: The communications cycle must be set to match the servo frequency of the Controller. In this Guide, set it to 1,000 µs.
3.	In the Solution Explorer, right-click *Master0 (Deactivated)* under *System - EtherCAT* and select *Scan EtherCAT Network* from the menu. 

4.	Check that the following slaves are displayed in the Solution Explorer.  Note: If the names of the slaves differ from the above, the variable names and data names that you need to set in the following procedures may differ from those mentioned in this Guide. In the following procedures, please proceed with the setup by replacing the names of the slaves that have been set.

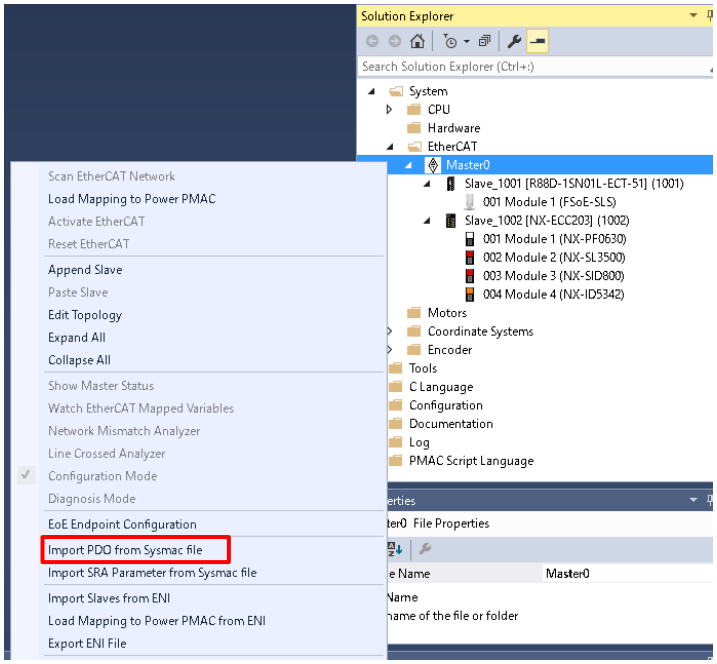
■ Configuring Distributed Clock Settings

1.	Open the Distributed Clocks Tab Page of *Master0 (Deactivated)*. 

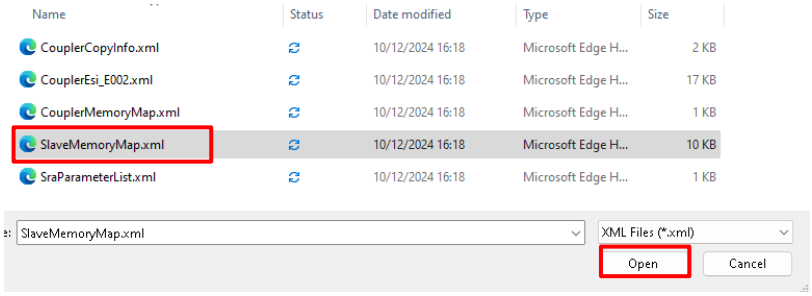
2.	Select the *Bus Shift (Reference Clock controlled by EtherCAT Master Time)* option. 
3.	Open *Slave_1001 [R88D-1SN01L-ECT-51] (1001)* from the Solution Explorer and display the Distributed Clock Tab Page. 
4.	Set *Operation Mode* to *DC for synchronization*. 

5.	Similarly, open *Slave_1002 [NX-ECC203](1002)* from the *Solution Explorer* and display the *Distributed Clock* Tab Page. 
6.	Set *Operation Mode* to *DC for synchronization*. 

■ Configuring PDO Map Settings

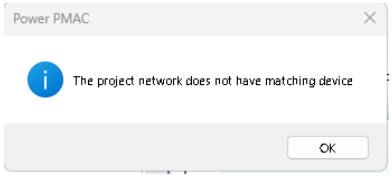
1.	In the Solution Explorer, right-click *Master0 (Deactivated)* and select *Import PDO from Sysmac file* from the menu. 

2. Decompress the zip file exported from the Sysmac Studio (see Export Procedure for the Safety Configuration File) and click *SlaveMemoryMap.xml* in the folder to open it.

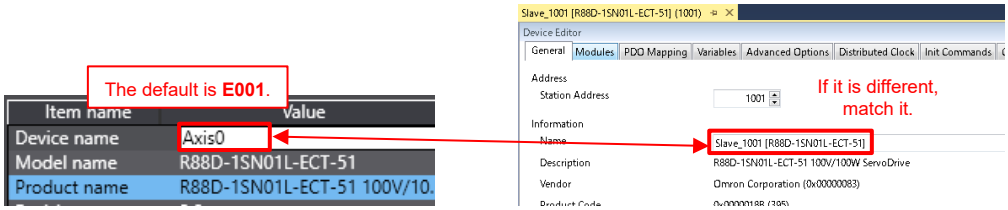


When the following dialog box appears, click the *OK* Button.

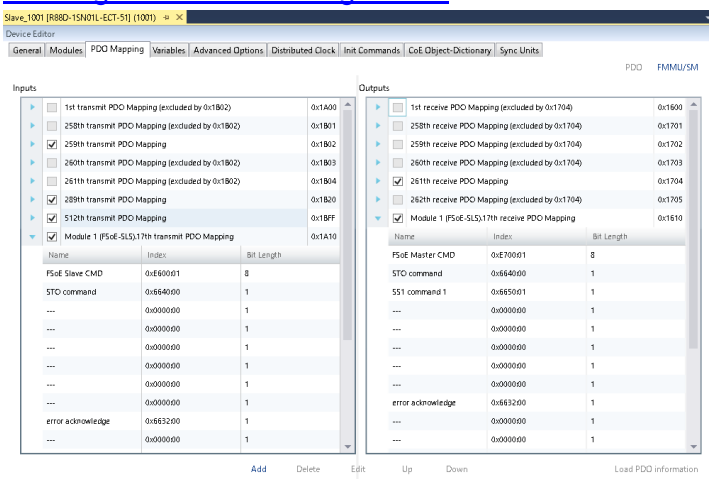
Note: If you have changed the device name of the Servo Drive from the default name in the network configuration created in 3.3 Setup of NX-series Safety Control Units, the following dialog box appears and you cannot configure the PDOs.



Check the device name of the Servo Drive in the Power PMAC IDE and the device name set in the Sysmac Studio, and change the device name in the Power PMAC IDE so that they match.

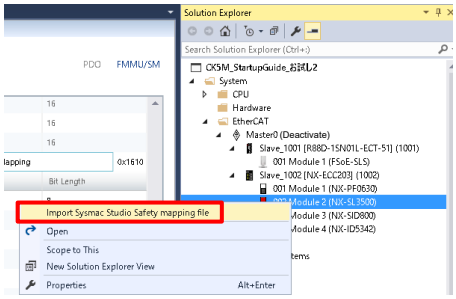


* To check the device name in the Power PMAC IDE, open *Slave_1001 [R88D-1SN01L-ECT-51] (1001)* from the Solution Explorer and display the *General* Tab Page.

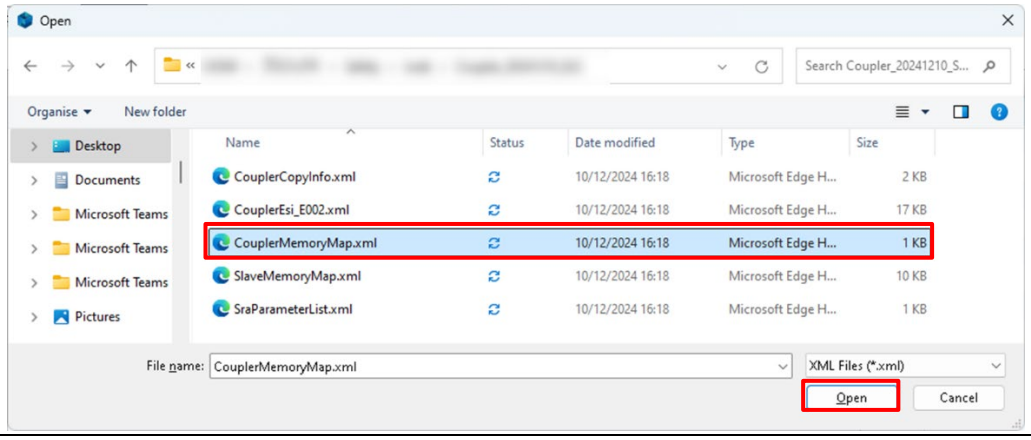
3.	Open *Slave_1001 [R88D-1SN01L-ECT-51] (1001)* from the Solution Explorer and display the *PDO Mapping* Tab Page. Check that the PDO map settings are the same as the those you configured in 3.3.1 Creating the Network Configuration. 

■ Setting the Safety Controller

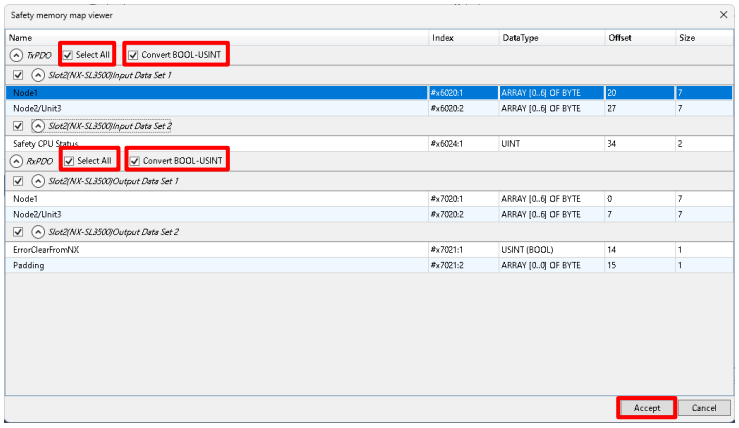
1. In the Solution Explorer, right-click *002 Module2(NX-SL3500)* under *Master0 (Deactivate) - Slave_1002[NX-EC203* and select *Import Sysmac Studio Safety mapping file* from the menu.



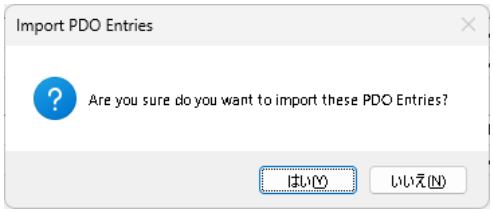
2. Among the files that you exported in Export Procedure for the Safety Configuration File, click *CouplerMemoryMap.xml* to open it.

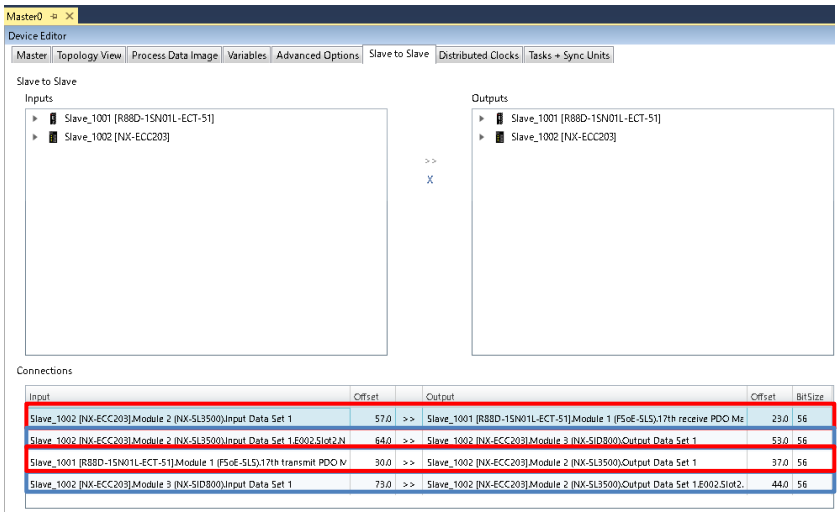


3. The *Safety memory map viewer* Pane opens.
Check that the *Select All* and *Convert BOOL-USINT* Check Boxes for TxPDO and RxPDO are selected and click the *Accept* Button.



Click the **Yes** Button.

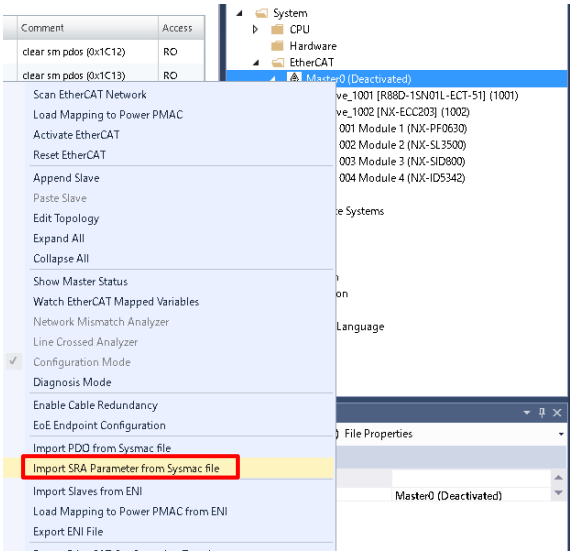


4.	Open *Master0 (Deactivate)* from the Solution Explorer and display the Slave_to Slave Tab Page. In *Connections*, check that 1S-51 and SL3500 are linked (in the red frame), and that SID800 and SL3500 are linked (in the blue frame). 

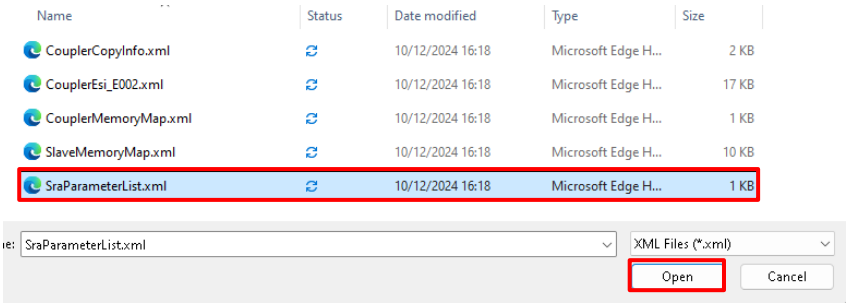
■ Setting the SRA Parameters

Check that the SRA parameters settings of the CK5M match those of the safety controller. For information on setting the safety controller, refer to 3.3.3 Setting the Safety Controller.

1. In the Solution Explorer, right-click *Master0 (Deactivated)* and select *Import SRA Parameter from Sysmac file* from the menu.

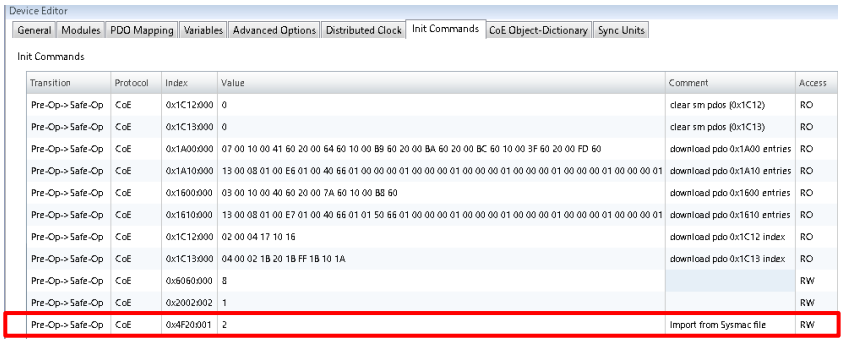


2. Among the files that you exported in Export Procedure for the Safety Configuration File, select *SraParameterList.xml* and click the *Open* Button.

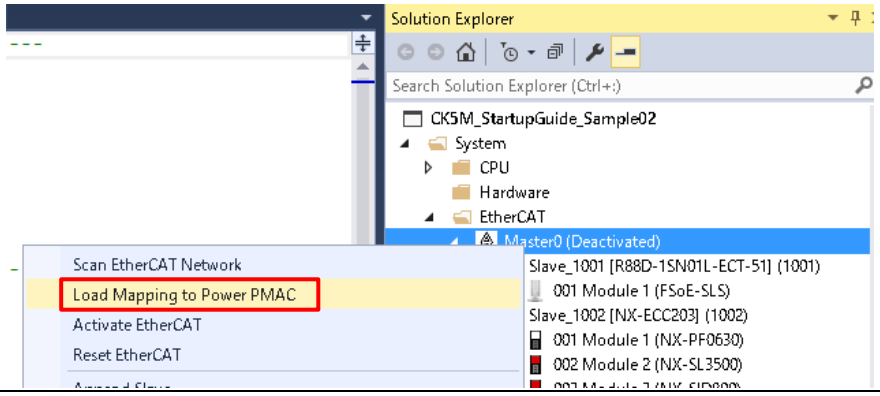
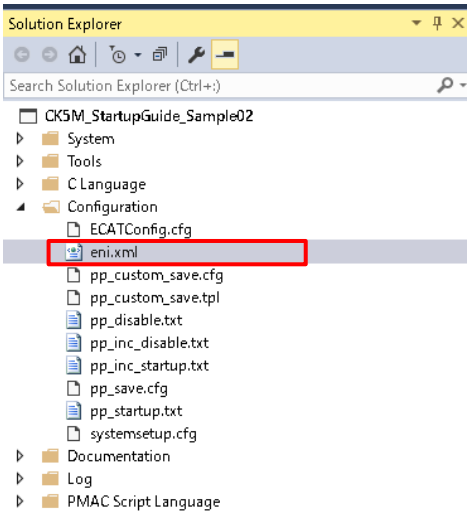
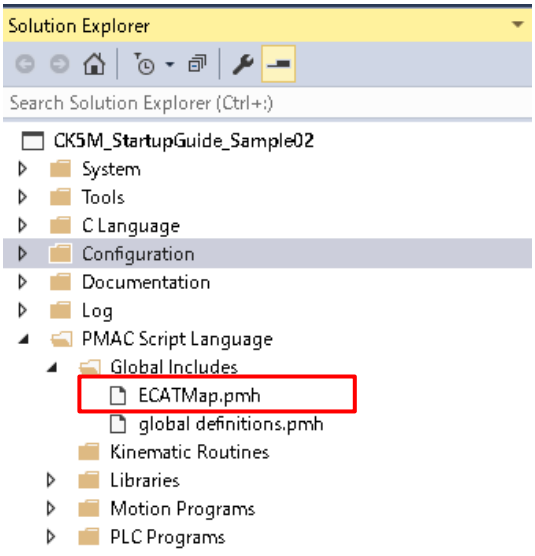


When the following dialog box appears, click the *OK* Button.

3. Open *Slave_1001 [R88D-1SN01L-ECT-51] (1001)* from the Solution Explorer and display the Init Commands Tab Page.
Check that OD is added, as shown below.

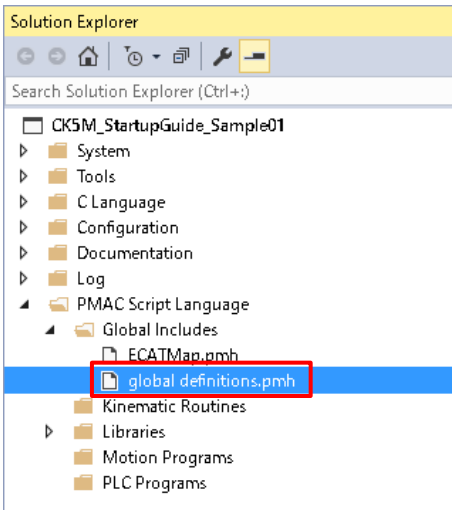


■ Creating an EtherCAT Network Configuration File

1.	In the Solution Explorer, right-click *Master(Deactivated)* under *System - EtherCAT* and select *Load Mapping to PowerPMAC* from the menu. 
2.	*eni.xml* is added under *Configuration* in the Solution Explorer. 
3.	*ECATMap.pmh* is added under *PMAC Script Language* in the Solution Explorer. 

3.4.4. Setting Axis Variables

1. Open *global definitions.pmh* under *ProjectName - PMAC Script Language - Global Includes* from the Solution Explorer.

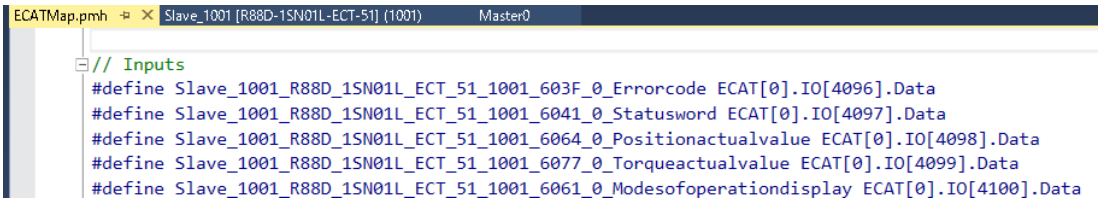


2. Write the following code in the programming area.

```
//define
#define true 1
#define false 0

// Motor
Motor[1].ServoCtrl=1                                    //enable motor control
Motor[1].Ctrl=Sys.PosCtrl                               // Output command position to pdac
Motor[1].EncType=0                                      //Encoder type None
Motor[1].pDac=Slave_1001_R88D_1SN01L_ECT_51_1001_607A_0_Targetposition.a
                                                        //Position command pointer
Motor[1].pEnc=EncTable[1].a                             //Position feedback pointer
Motor[1].pEnc2=EncTable[1].a                            //Speed feedback pointer
Motor[1].pAmpEnable=Slave_1001_R88D_1SN01L_ECT_51_1001_6040_0_Controlword.a
                                                        //driver directive pointer
Motor[1].AmpEnableBit=3                                 //Reference bits in the driver directive pointer
Motor[1].pAmpFault=Slave_1001_R88D_1SN01L_ECT_51_1001_6041_0_Statusword.a
                                                        //Pointer driver error
Motor[1].AmpFaultBit=3                                  //Reference bit of the driver fault pointe(Fault bit)
Motor[1].AmpFaultLevel=3                                //Logical state of driver error.
Motor[1].EcatAmpFaultLimit=1000
Motor[1].pLimits=0                                      //Software limit disabled
Motor[1].LimitBits=0                                    //Number of limit signal bits for motors
Motor[1].WarnFeLimit=1000                               //Deviation warning limit value
Motor[1].FatalFeLimit=0                                 //Deviation Abnormal Limit Value
Motor[1].Stime=ECAT[0].ServoExtension                   //Extension of servo cycle
Motor[1].MaxSpeed=3000/60*(1<<23)/1000                  //Maximum speed 23-bit encoder 3000 r/min
Motor[1].pAdc=Sys.pushm
Motor[1].pPhaseEnc=Sys.pushm
Motor[1].pEncStatus=Sys.pushm
Motor[1].pEncCtrl=Sys.pushm
Motor[1].CaptControl=0
Motor[1].pCaptFlag=Sys.pushm
Motor[1].pCaptPos=Sys.pushm
Motor[1].PosSf=1                                        //electronic gear ratio
Motor[1].Pos2Sf=1                                       //electronic gear ratio
Motor[1].MasterPosSf=1
```

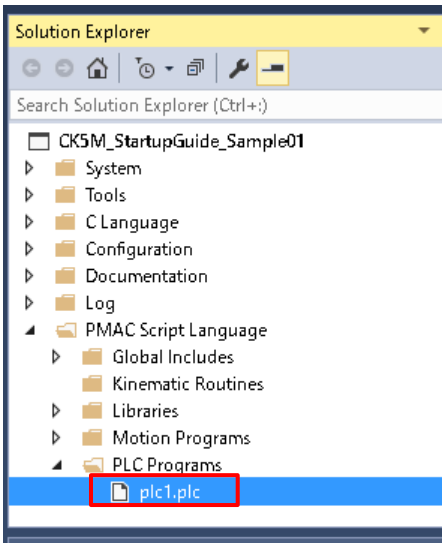
```
Motor[1].JogTa=500                                  //Jog acceleration/deceleration time
Motor[1].JogTs=500                                  //Jog S-curve acceleration/deceleration
Motor[1].AbortTa=100                          //Deceleration during Abort
Motor[1].JogSpeed=1200/60*(1<<23)/1000      //Jog speed/ms 23-bit encoder 1200 r/min
Motor[1].HomeVel=10                                 //Home search speed/ms
Motor[1].InvAmax=10                                 //Maximum acceleration
Motor[1].InvDmax=10                                 //Maximum deceleration
Motor[1].InvJmax=50                                 //Max Jerk
Motor[1].MaxDac=32768                               //Maximum DAC output

//Encoder
EncTable[1].type=1
EncTable[1].index1=0
EncTable[1].index2=0
EncTable[1].index3=0
EncTable[1].index4=0
EncTable[1].index5=0
EncTable[1].pEnc=Slave_1001_R88D_1SN01L_ECT_51_1001_6064_0_Positionactualvalue.a
EncTable[1].pEnc1=Slave_1001_R88D_1SN01L_ECT_51_1001_6064_0_Positionactualvalue.a
EncTable[1].ScaleFactor=1
```

Note: Variables such as *Slave_1001_R88D_1SN01L_ECT_51_1001_6064_0_Positionactualvalue* are the parameter variables included in *ECATMap.pmh* that you created in 3.4.3 Configuring EtherCAT Communications Settings. Check the variable names in advance, as they vary depending on the slave names specified in the Sysmac Studio project that you created in 3.3 Setup of NX-series Safety Control Units or the Power PMAC IDE project that you created in 3.4.3 Configuring EtherCAT Communications Settings.



3.4.5. Creating an Operation Program

1. Open *plc1.plc* under *ProjectName - PLC Programs - Global Includes* from the Solution Explorer.

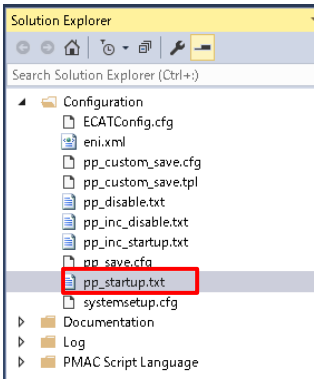


2. Write the following program in the programming area of the plc1.plc Tab Page.
This program causes the motor shaft to run at 1,200 rpm three seconds after the Servo is turned ON after the reset switch is pressed.

```
/*For more information see notes.txt in the Documentation folder */

global moveSequence = 0;                    //for plc1 Sequence
global isResetButton = false;          //Check if the reset button is pressed
global STOCommand = true;                   //sto command(true:Deactivate)
global STOStatus = false;                   //sto command(false:Deactivate)

open plc 1

        while(sys.ecatMasterReady==0){};
        isResetButton = (Slave_1002_NX_ECC203_1002_6042_1_StandardInput1stWord >> 6) & 1;
        STOCommand = (Slave_1001_R88D_1SN01L_ECT_51_1001_4130_130_MirrorSafetycontrolword & 1);
        STOStatus = (Slave_1001_R88D_1SN01L_ECT_51_1001_4130_131_MirrorSafetystatusword & 1);

        switch(moveSequence)
        {
                case 0://Setting
                        //EtherCAT Enabled
                        ECAT[0].Enable=true;
                        //Positive/Negative torque limit value set to 30%
                        Slave_1001_R88D_1SN01L_ECT_51_1001_60E0_0_Positivetorquelimitvalue = 300;
                        Slave_1001_R88D_1SN01L_ECT_51_1001_60E1_0_Negativetorquelimitvalue = 300;
                        Ldata.motor = 1;
                        moveSequence = 1;
                        break;
```

```
                case 1://Check if the reset button is pressed.
                        if(isResetButton == 1)
                        {
                                moveSequence = 2;
                        }
                        break;
                case 2: //servo ON
                        jog/; //servo ON & Full closed loop setting

                        //wait 3sec
                        P1000 = Sys.Time+3;
                        while(P1000>Sys.Time){};

                        moveSequence = 3;
                        break;

                case 3: //ckeck the servo ON & STO deactivate
                        if(Motor[1].AmpEna == true && STOStatus == false)
                        {
                                Motor[1].JogSpeed=1200/60*(1<<23)/1000 //Jog velocity per ms
                                jog+; //move motor(1200rpm)
                        }
                        else
                        {
                                kill;//servo OFF
                                moveSequence = 1; //Start over from sequence1
                        }
                        break;

                default:
                        moveSequence = 0;
                        break;
        }
        //Detection of rising error switch
        Slave_1002_NX_ECC203_1002_7002_1_E002Slot2ErrorClearFromNX =
Slave_1002_NX_ECC203_1002_6062_1_InputBit16bits;
close
```

3. Open *pp_startup.txt* under *ProjectName - Configuration* from the Solution Explorer.



4. Add the following code to the end of the program in the programming area of the pp_startup.txt Tab Page.

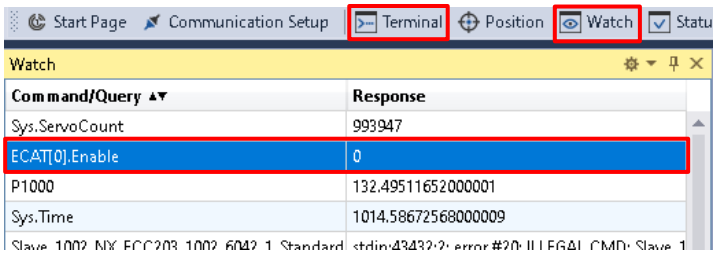
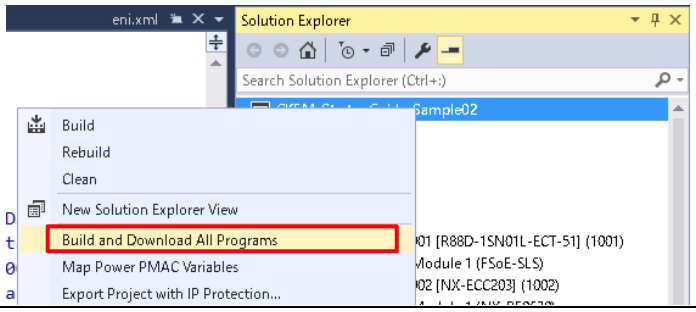
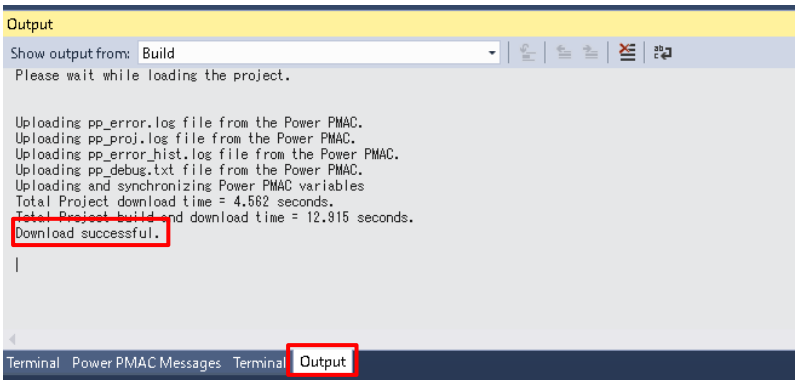
```
enable plc 1;
```

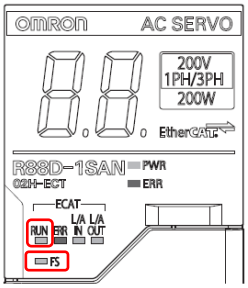
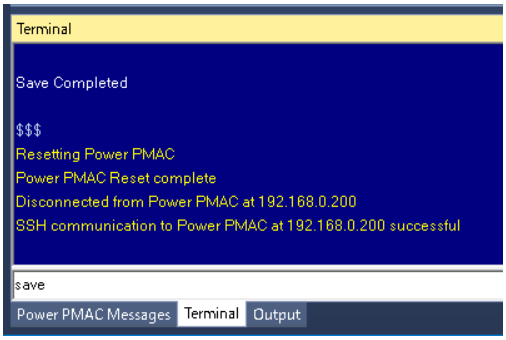
The pp_startup.txt file is automatically executed at the startup of the Controller.
In this example, the script for PLC1 is executed.

3.4.6. Downloading the Project and Checking the Operation

Download the created project to the Controller.

After the project is downloaded, the program starts automatically.

1.	Check that the value of *ECAT[0].Enable* is 0 in the Terminal or Watch Pane.  If the value is 1, execute the command ECAT[0].Enable=0 from the Terminal to stop EtherCAT communications.
2.	Right-click the project name in the Solution Explorer at the right of the IDE Window and select *Build and Download All Programs* from the menu to build and download the project. 
3.	Check that there are no errors in the *Output* Pane. If *Download successful* is displayed, the download is completed successfully.  Note: If the download fails, check if an error has occurred in the *Output* Pane. If it is a program error, review the program. If it is related to EtherCAT settings, go back to 3.4.3 Configuring EtherCAT Communications Settings and check if there are no errors in the settings.

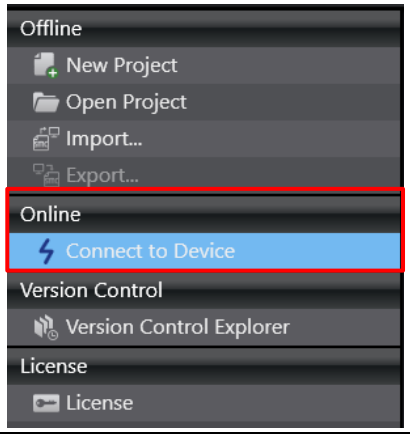
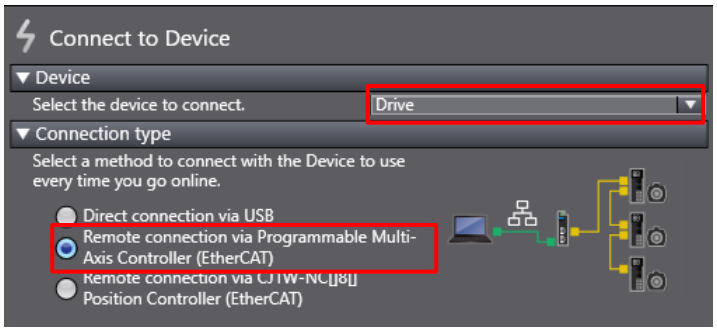
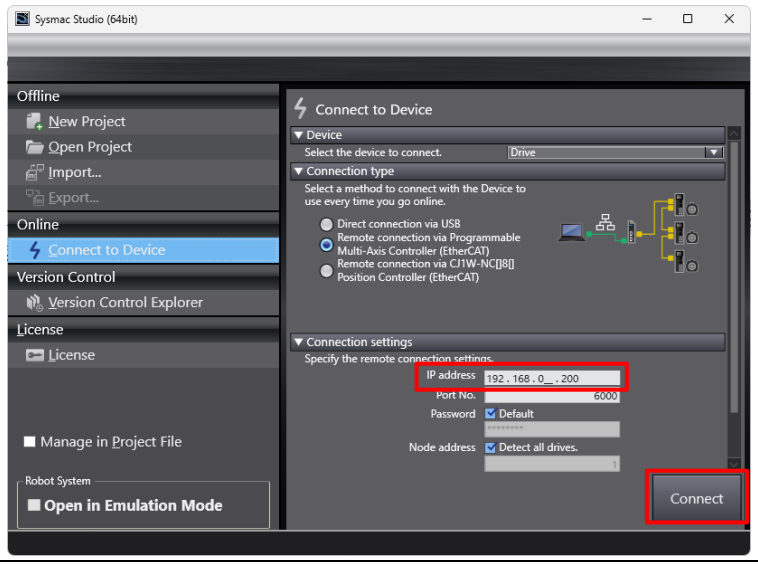
4.	When the download is completed successfully, EtherCAT communications and FSoE communications are established. You can check the communications status by two status indicators located on the front panel of the Servo Drive. When EtherCAT communications are established, the ECAT-RUN indicator is lit green. When FSoE communications are established, the FS indicator is lit green.  Note: If EtherCAT communications or FSoE communications are not established, perform the procedures in 3.3 Setup of NX-series Safety Control Units and 3.4 Motion Controller CK5M again.
5.	In the Terminal, enter *save* to save the downloaded project to the Controller.  Note: If you turn OFF the power supply to the Controller without entering *save*, the downloaded project will be discarded.

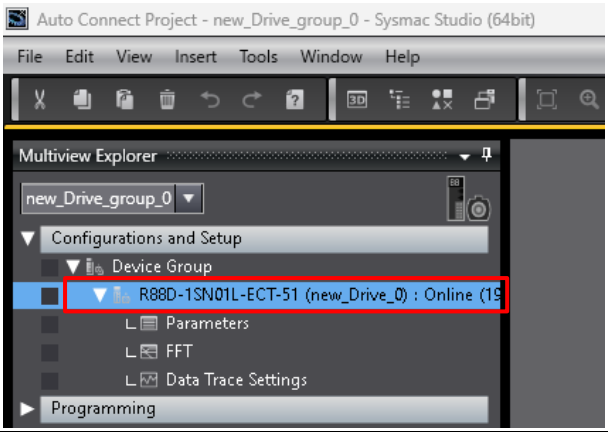
3.5. Setup of the Servo Drive

Set up the Servo Drive using the Sysmac Studio.

If you are continuing the operation from 3.4 Setup of the Motion Controller CK5M, set the connection status of the CK5M to *No Device* from *Communication Setup* in the Power PMAC IDE.

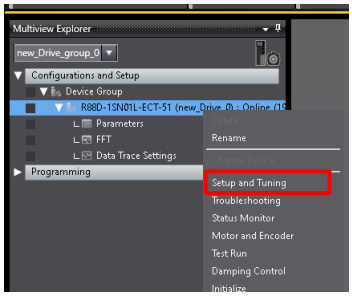
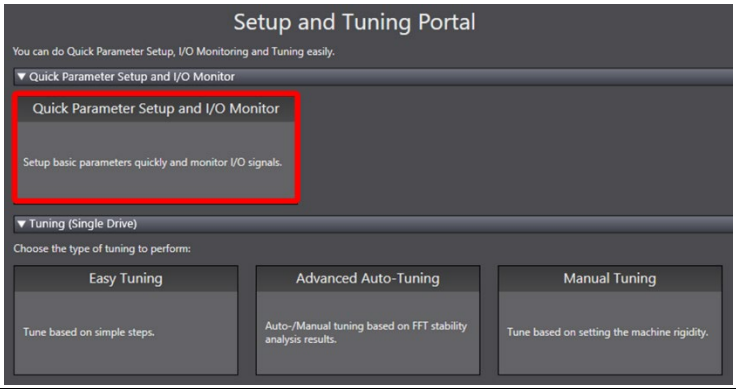
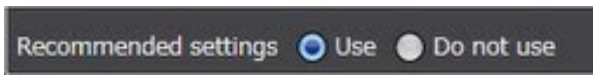
3.5.1. Connecting the Servo Drive via CK5M

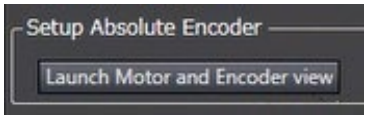
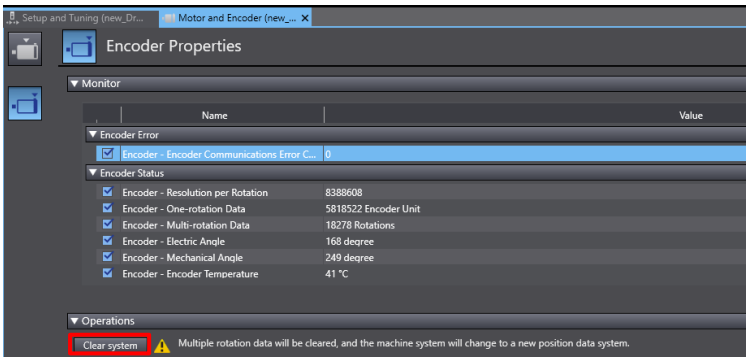
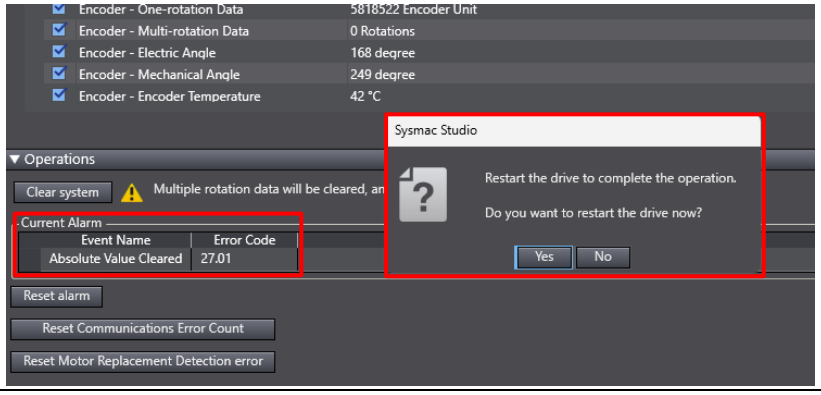
1.	Start the Sysmac Studio and click *Connect to Device* under *Online*. 
2.	Select *Drive* as the device to connect to, and select the *Remote connection via Programmable Multi-Axis Controller (EtherCAT)* option as the method to connect to the device. 
3.	Next, in the connection settings, enter the IP address *192.168.0.200* and click the *Connect* Button. 

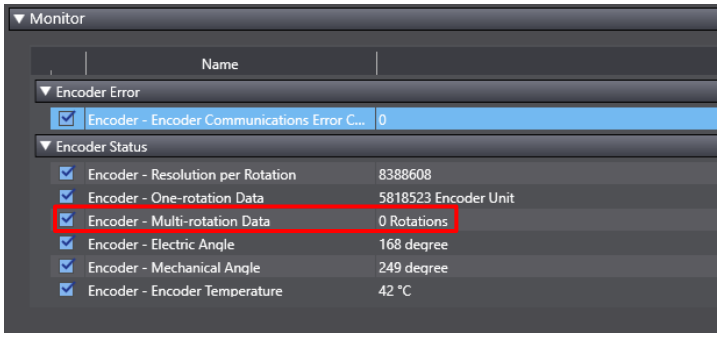
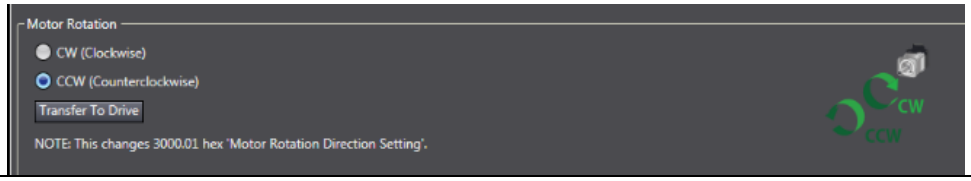
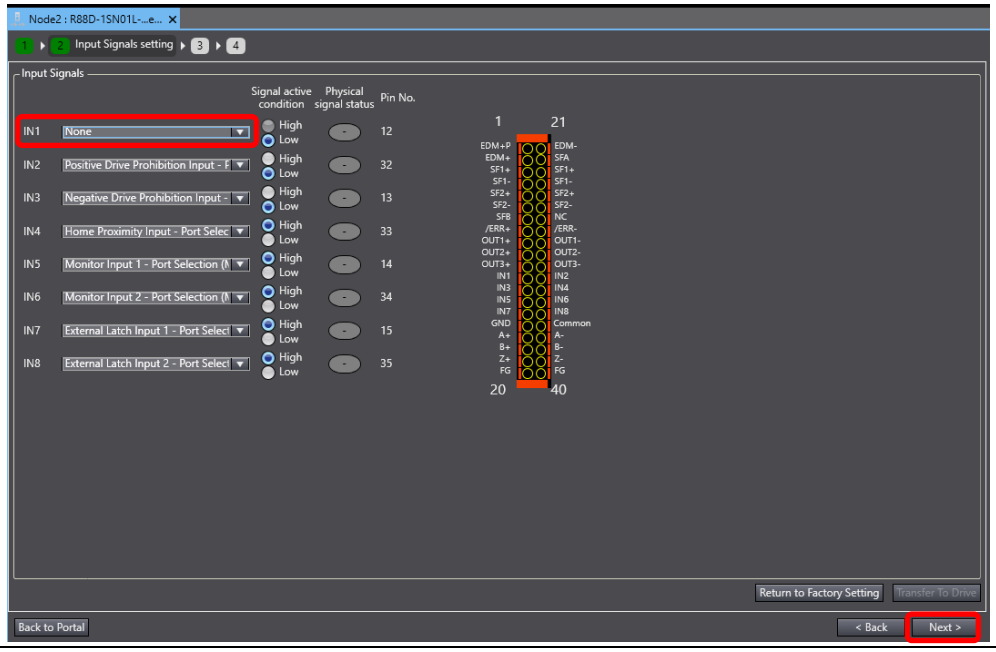
4.	The display changes and *R88D-1SN01L-ECT-51* in the Multiview Explorer, indicating that the Servo Drive is now online. 

3.5.2. Motor, ABS Encoder and I/O Setup

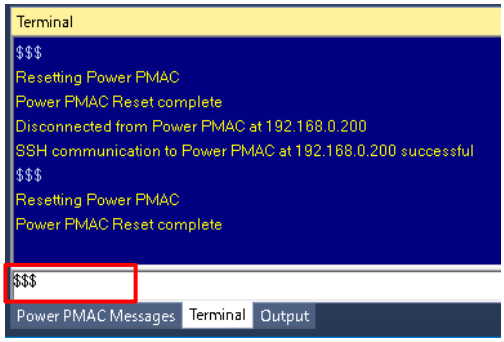
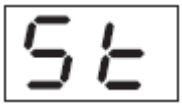
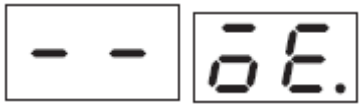
■ Quick Parameter Setup and I/O Monitor Wizard

1.	Right-click the Servo Drive and select *Setup and Tuning* from the menu. 
2.	Click the *Quick Parameter Setup and I/O Monitor* Button. 
3.	Select whether or not an OMRON Controller is connected, and set the recommended setting to *Use*.  Related inputs) IN2: POT, IN3: NOT, IN4: DEC, IN7: EXT1, IN8: EXT2, How to Use Absolute Encoder: Use as absolute encoder but ignore multi-rotation counter overflow

4.	Only when using a Servomotor with an absolute encoder, select the *Use as absolute encoder but ignore multi-rotation counter overflow* option for *How to Use Absolute Encoder*. 
5.	Set up the absolute encoder (if required).  Use this function when clearing the multiple rotation data or when replacing a Servomotor in the actual machine. To reset multiple rotation data, click the **Clear system** Button.  You need to restart the Servo Drive to complete the operation. Click the **Yes** Button. 

	Check that multiple rotation data of the encoder has been cleared. 
6.	Select the motor rotation direction and transfer the settings to the Servo Drive. 
7.	Set input signals and transfer the settings to the Servo Drive. The Error Stop Input (ESTP) is ON by default. Turn it OFF as follows if necessary. When ESTP is ON, the Error No. 87.00 is displayed on the Servo Drive. 
8.	Set output signals if necessary and transfer the settings to the Servo Drive.
9.	Click the *Finish* Button. 

3.6. Operation Check of the STO Function

1.	Execute the *$$$* command from the Terminal of the Power PMAC IDE to restart the CK5M. 
2.	After the restart of the Controller is completed, press the safety reset button. 
3.	Check that the 7-segment LED display shows ‘oE.’. 
4.	After about three seconds, check that the Servomotor rotates at about 1,200 r/min.
5.	Press the Emergency Stop Pushbutton Switch.  Check that the 7-segment LED display shows ‘st’. 
6.	Release the Emergency Stop Pushbutton Switch and press the safety rest button.  Check that the STO is released and that the 7-segment LED display initially shows “--” and then immediately changes to 'oE.' .  Note: If the LED display does not change to ‘oE.’, press the safety reset button again.

4. Adding Safety Functions

This section describes how to add safety functions to the servo system that you built in 3. Performing Setup.
Refer to the section of the safety function to add.

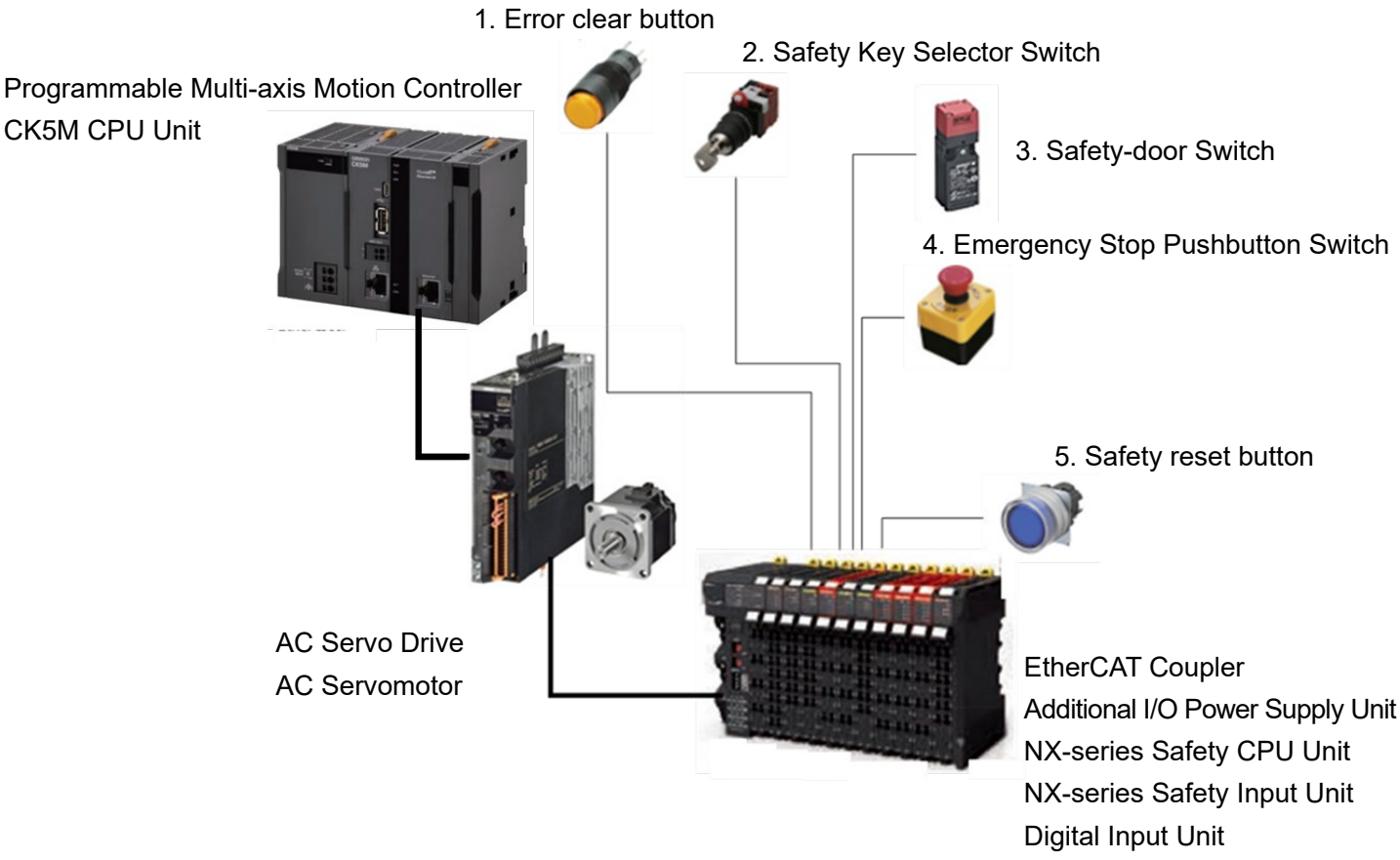
4.1. Adding the Safe Stop 1 (SS1) Function

This section describes how to add the SS1 function to the project that you created in 3. Performing Setup.

The operation of the servo system set up in this section is explained below.

1. When the error clear button is pressed, the errors of the standard controller and Servo Drive are reset.
2. When the Safety Key Selector Switch is operated to switch to safety active mode, the standard controller lets the Servomotor decelerate to a stop.
   The Servo Drive activates the STO function using the SS1 function to turn OFF the motor torque. When the Emergency Stop Pushbutton Switch is pressed, the motor torque is turned OFF.
3. When the safety reset button is pressed, the STO status is reset.

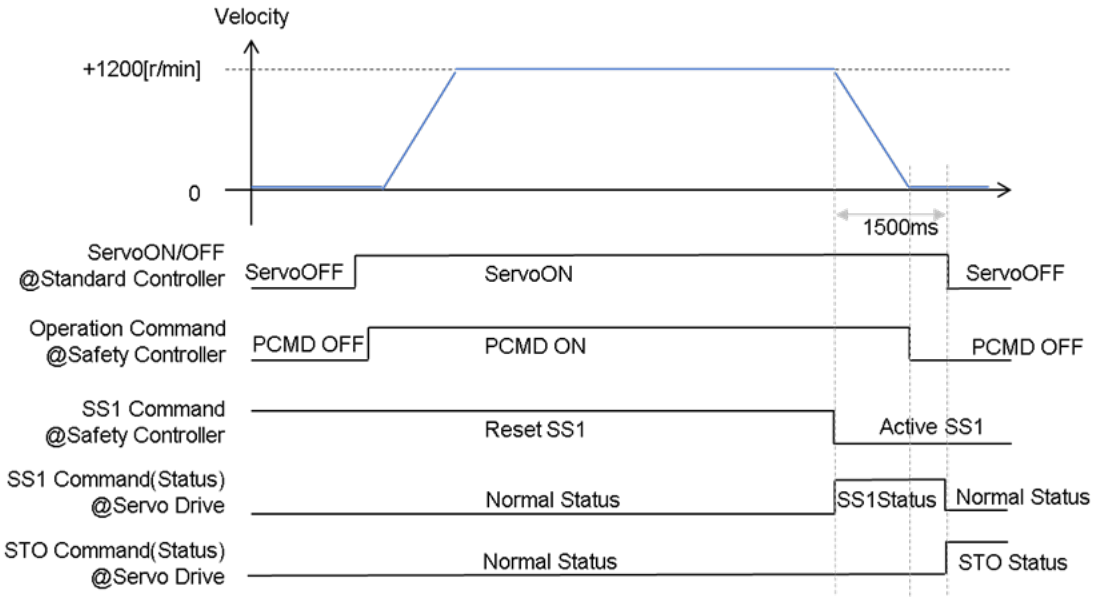
■ System Configuration



Input device	State	Operation
1. Error clear button	ON	Enable error reset command
	OFF	Disable error reset command
2. Safety Key Selector Switch	Normal operating mode	Set the Servomotor to normal velocity.
	Safety active mode	Make Servomotor decelerate to a stop and activate STO function using SS1 function.
3. Safety-door Switch	Open	SS1 function deactivated: Enable STO command SS1 function activated (during deceleration): Enable STO command SS1 function activated (after STO activation from SS1): Disable STO command
	Close	Disable STO command
4. Emergency Stop Pushbutton Switch	ON	Enable STO command
	OFF	Disable STO command
5. Safety reset button	ON	Enable reset STO status command
	OFF	Disable reset STO status command

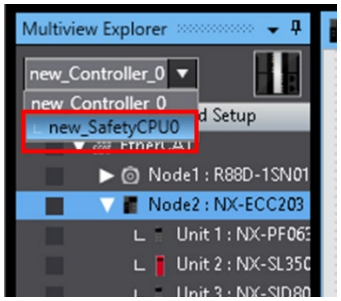
■ Operation of SS1 Function Interlocked with Motion Control

1. When the Servo ON command is enabled, the Servo Drive turns ON the Servo.
2. When an operation command is enabled, the command velocity to the Servomotor is 1,200 r/min.
3. The standard controller lets the Servomotor decelerate to a stop.
   When the SS1 function is executed, the Servo Drive shifts to the STO state after the wait time (SS1 time to STO 1) and turns OFF torque.

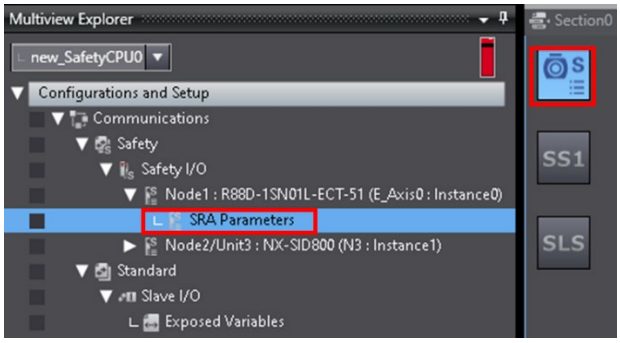


4.1.1. Setting the Safety Controller

1. Click *new_SafetyCPU0*.



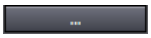
2. Open the Safety Function Assignment Settings view for *SRA Parameters*.



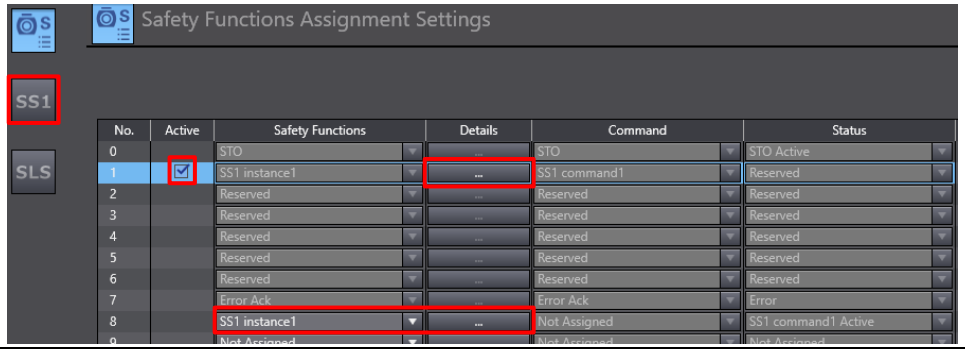
3. Check the *Active* Check Box for function No.1 and assign *SS1 instance1* to function No.8.

Next, click the 

Button to display the SS1 Detailed Settings view.

You can also use the 

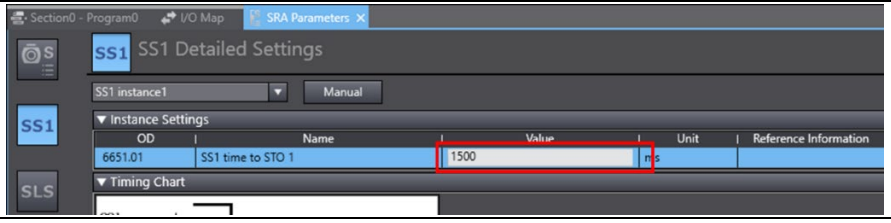
Button to display the SS1 Detailed Settings view.



5. Set SS1 parameters.

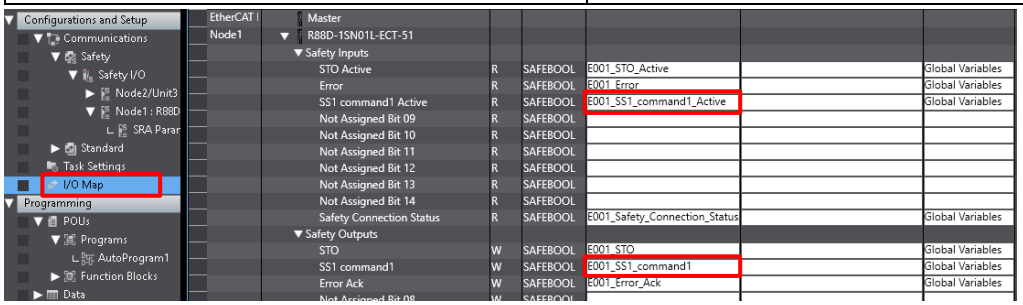
In this Guide, set as follows:

Name	Value	Unit
SS1 time to STO 1	1,500	ms



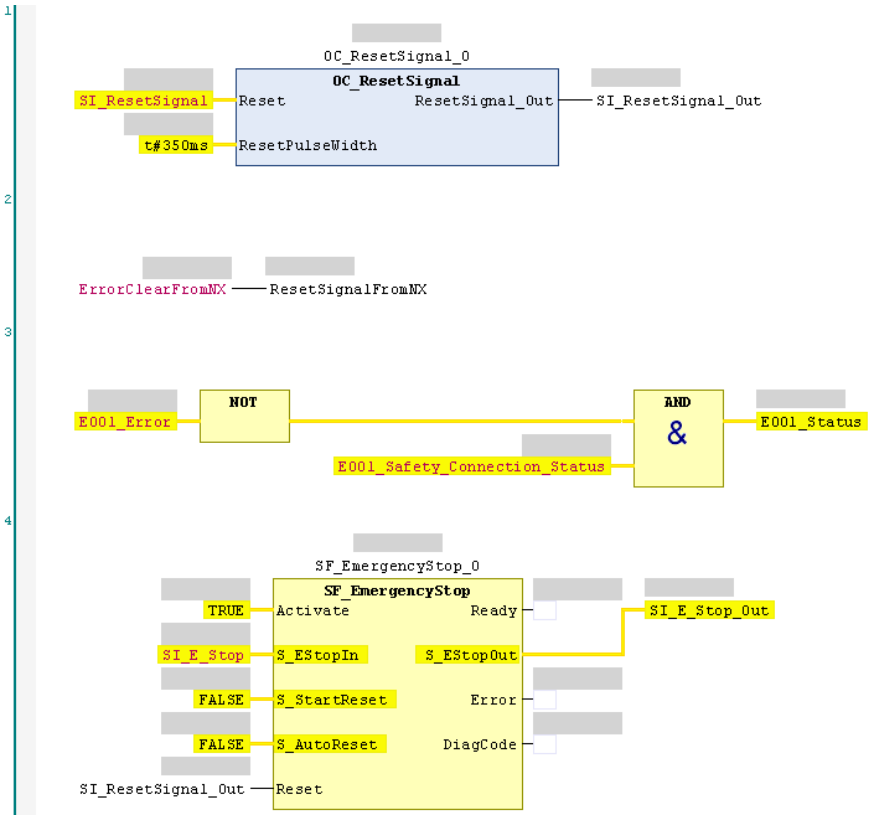
6. Open the I/O Map and create device variables by right-clicking the port to set and clicking *Create Device Variable* from the menu.

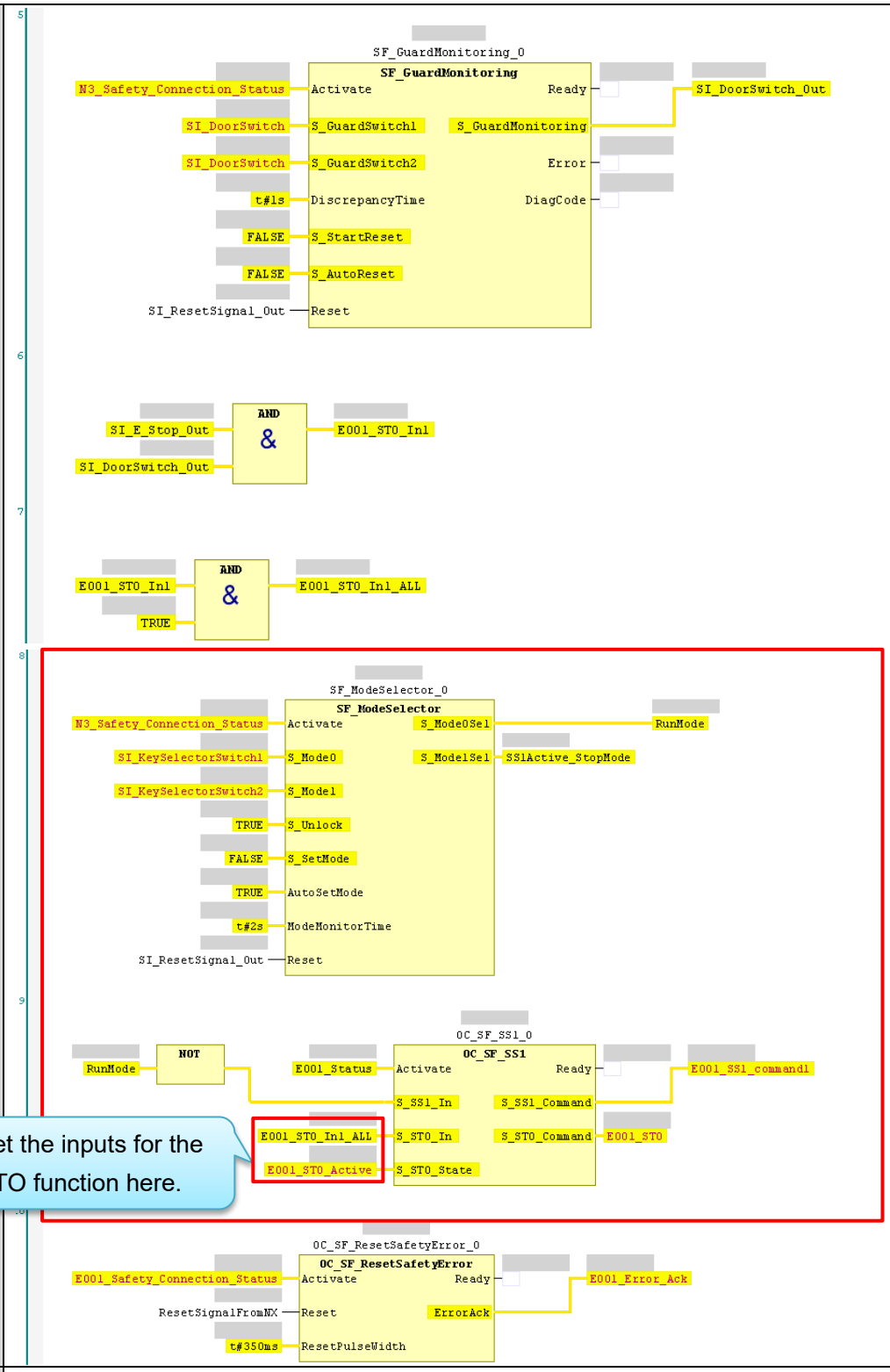
Port	Variable name
SS1 command1 Active for R88D-1SN01L-ECT-51	E001_SS1_command1_Active
SS1 command1 for R88D-1SN01L-ECT-51	E001_SS1_command1



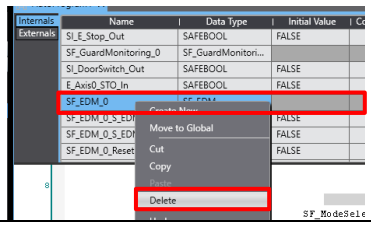
7. Create a safety program.

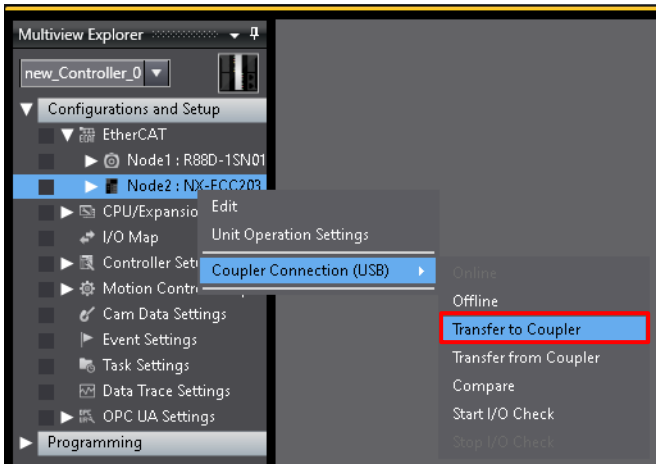
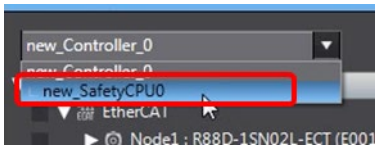
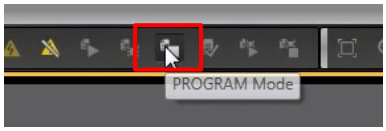
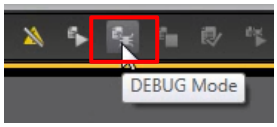
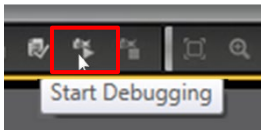
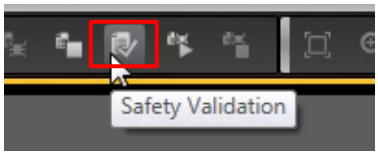
Add the following code to AutoProgram1.

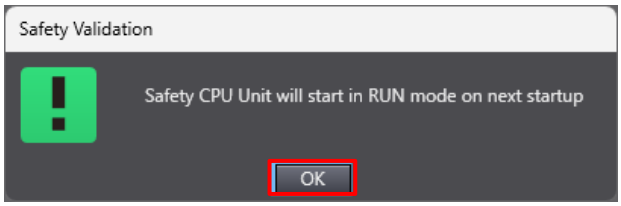
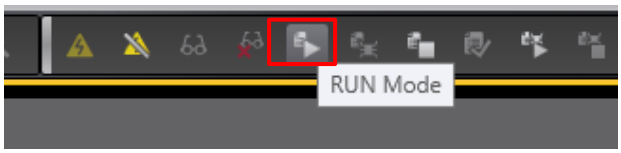
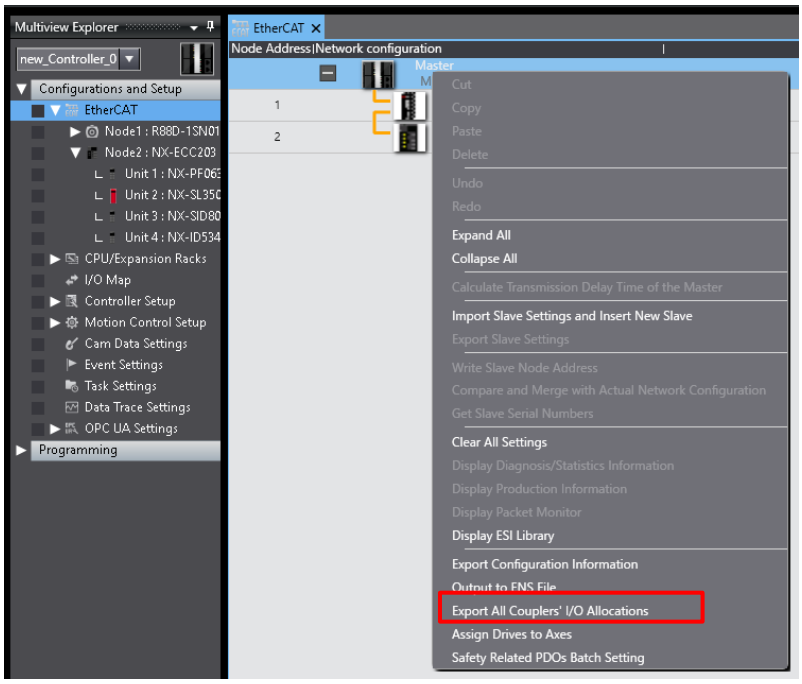




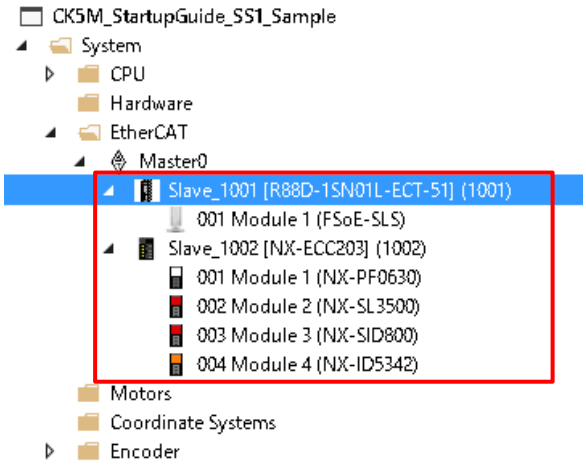
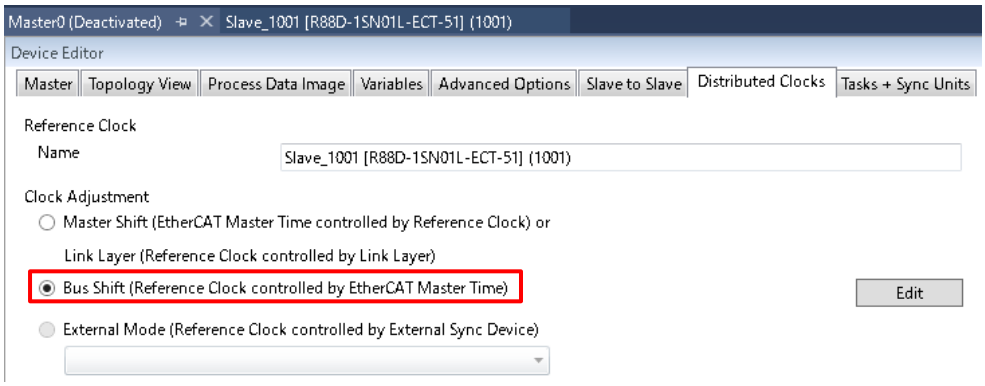
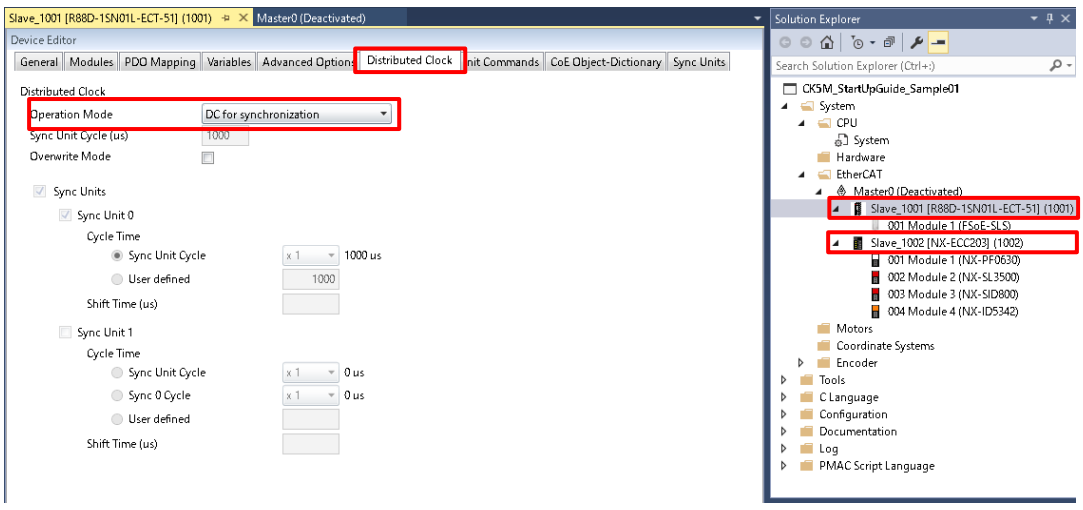
8. Since the SF_EDM function block is deleted, right-click the internal variable *SF_EDM_0* and select *Delete* to delete it.

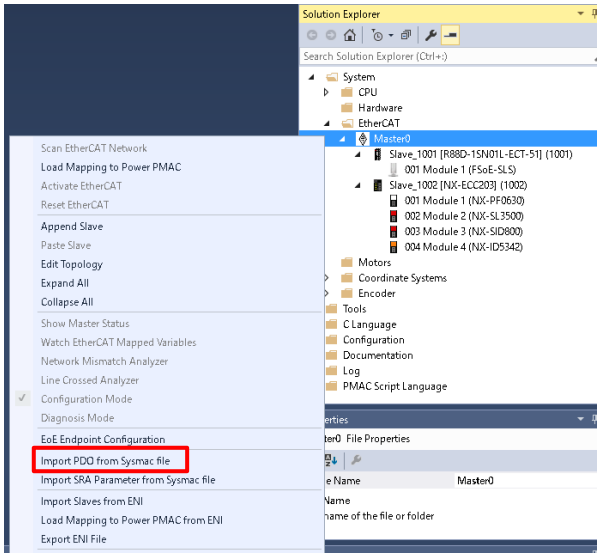
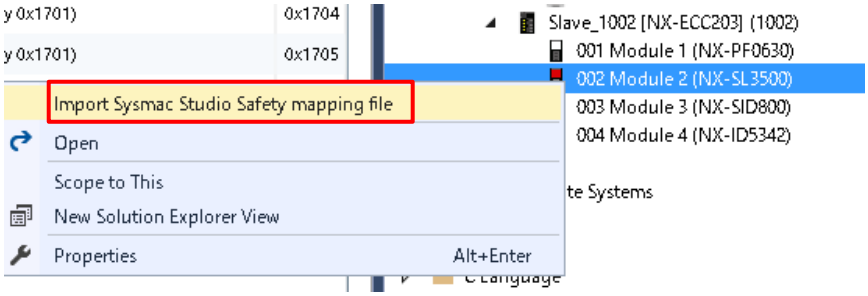
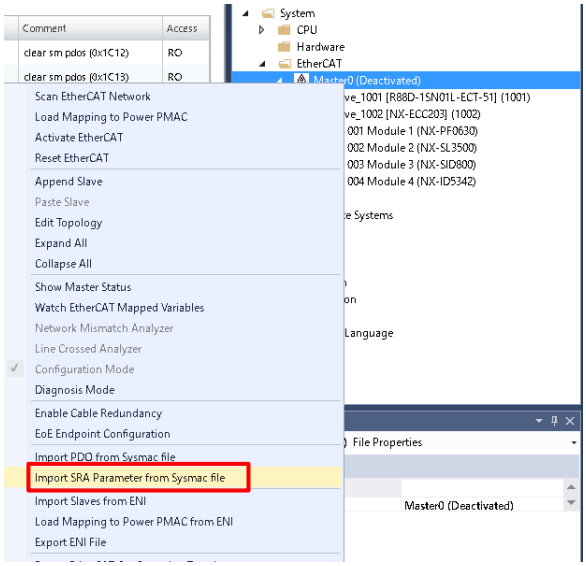


9.	Execute *Transfer to Coupler* to transfer the settings.  Note: Refer to Transfer of Settings and Projects for details on the procedure.
10.	Download the safety application. Select *new_SafetyCPU0* from the list.  Click the **PROGRAM Mode** Button to switch to PROGRAM mode.  Click the **DEBUG Mode** Button to enter DEBUG mode.  Click the **DEBUG Mode** Button to start DEBUG mode.  Click the **Safety Validation** Button. 

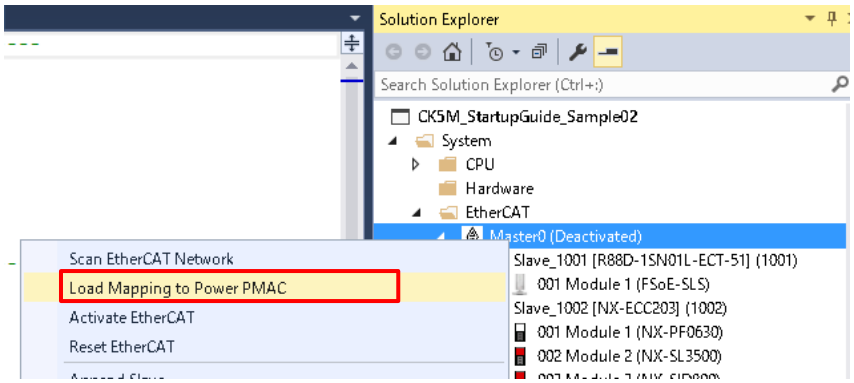
	The safety application is now ready to run.  Click the **RUN Mode** Button. 
11.	Execute *Export All Couplers' I/O Allocations*.  Note: Refer to Export Procedure for the Safety Configuration File for details on the procedure.

4.1.2. Setup of the CK5M

1.	Execute *Scan EtherCAT Network* to scan the current configuration.  Note: Refer to Configuring EtherCAT Master Communications in section 3.4.3 for details on the procedure.
2.	Open the Distributed Clocks Tab Page for *Master0 (Deactivated)* and select the *Bus Shift (Reference Clock controlled by EtherCAT Master Time)* option for *Clock Adjustment*.  Note: Refer to Configuring Distributed Clock Settings in section 3.4.3 for details on the procedure.
3.	Open *Slave_1001 [R88D-1SN01L-ECT-51] (1001)* and *Slave_1002 [NX-ECC203](1002)* from the Solution Explorer and display the Distributed Clock Tab Page. Set *Operation Mode* to *DC for synchronization*. 

4.	Decompress the zip file exported from the Sysmac Studio (see 4.1.1 Setting the Safety Controller), import *SlaveMemoryMap.xml* from the folder, and configure the PDO to open it.  Note: Refer to Configuring PDO Map Settings in section 3.4.3 for details on the procedure.
5.	Import *CouplerMemoryMap.xml* from the folder decompressed in step 4, and set up the safety controller.  Note: Refer to Setting the Safety Controller in section 3.4.3 for details on the procedure.
6.	Import *SraParameterList.xml* from the folder decompressed in step 4, and set the SRA parameters.  Note: Refer to Setting the SRA Parameters in section 3.4.3 for details on the procedure.

7. Create an EtherCAT network configuration file.



Note: Refer to Creating an EtherCAT Network Configuration File in section 3.4.3 for details on the procedure.

8. Set up the Servomotor.
Follow the same procedure as in section 3.4.4 Setting Axis Variables to set axis variables.

9. Modify the operation program. Add the following codes to *plc1.plc*.
The codes shown in the red frame below are added or modified from the program that you created in 3.4.5 Creating an Operation Program.

```
/*For more information see notes.txt in the Documentation folder */

global moveSequence = 0;                    //for plc1 Sequence
global isResetButton = false;          //Check if the reset button is pressed
global STOCommand = true;                   //sto command(true:Deactivate)
global STOStatus = false;                   //sto command(false:Deactivate)
global SS1Status = false;                   //ss1 command(false:Deactivate)

open plc 1

        while(sys.ecatMasterReady==0){};
        isResetButton = (Slave_1002_NX_ECC203_1002_6042_1_StandardInput1stWord >> 6) & 1;
        STOCommand = (Slave_1001_R88D_1SN01L_ECT_51_1001_4130_130_MirrorSafetycontrolword & 1);
        STOStatus = (Slave_1001_R88D_1SN01L_ECT_51_1001_4130_131_MirrorSafetystatusword & 1);
        SS1Status = ((Slave_1001_R88D_1SN01L_ECT_51_1001_4130_131_MirrorSafetystatusword>>8)&1);
        switch(moveSequence)
        {
                case 0://Setting
                        //EtherCAT Enabled
                        ECAT[0].Enable=true;
                        //Positive/Negative torque limit value set to 30%
                        Slave_1001_R88D_1SN01L_ECT_51_1001_60E0_0_Positivetorquelimitvalue = 300;
                        Slave_1001_R88D_1SN01L_ECT_51_1001_60E1_0_Negativetorquelimitvalue = 300;
                        Ldata.motor = 1;
                        moveSequence = 1;
                        break;

                case 1://Check if the reset button is pressed.
                        if(isResetButton == 1)
                        {
                                moveSequence = 2;
                        }
                        break;
```

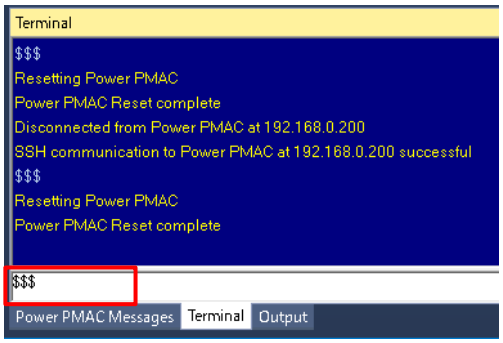
```
                case 2: //servo ON
                        jog/; //servo ON & Full closed loop setting

                        //wait 3sec
                        P1000 = Sys.Time+3;
                        while(P1000>Sys.Time){};

                        moveSequence = 3;
                        break;

                case 3: //ckeck the servo ON & STO deactivate
                        if(Motor[1].AmpEna == true && STOStatus == false)
                        {
                                if(SS1Status == true)
                                {
                                        jog/; //stop motor(0rpm)
                                }
                                else
                                {
                                        Motor[1].JogSpeed=1200/60*(1<<23)/1000 //Jog velocity per ms
                                        jog+; //move motor(1200rpm)
                                }
                        }
                        else
                        {
                                kill;//servo OFF
                                moveSequence = 1; //Start over from sequence1
                        }
                        break;

                default:
                        moveSequence = 0;
                        break;
        }
        //Detection of rising error switch
        Slave_1002_NX_ECC203_1002_7002_1_E002Slot2ErrorClearFromNX =
Slave_1002_NX_ECC203_1002_6062_1_InputBit16bits;

close
```

10. Add the following code to the end of the program in the programming area of the pp_startup.txt Tab Page.

```
enable plc 1;
```

11. Download the created project to the Controller.
Note: Refer to 3.4.6 Downloading the Project and Checking the Operation for details on the procedure.

4.1.3. Checking Operation of the SS1 Function

1.	Check that the Safety Key Selector Switch is in normal operating mode. 
2.	Execute the *$$$* command from the Terminal of the Power PMAC IDE to restart the CK5M. 
3.	After the restart of the Controller is completed, press the safety reset button. 
4.	Check that the 7-segment LED display shows ‘oE.’. 
5.	After about three seconds, check that the Servomotor rotates at about 1,200 r/min.
6.	Operate the Safety Key Selector Switch to switch to safety active mode.  The Servomotor decelerates to a stop and the Servo Drive goes into STO state. Check that the 7-segment LED display shows ‘st’. 

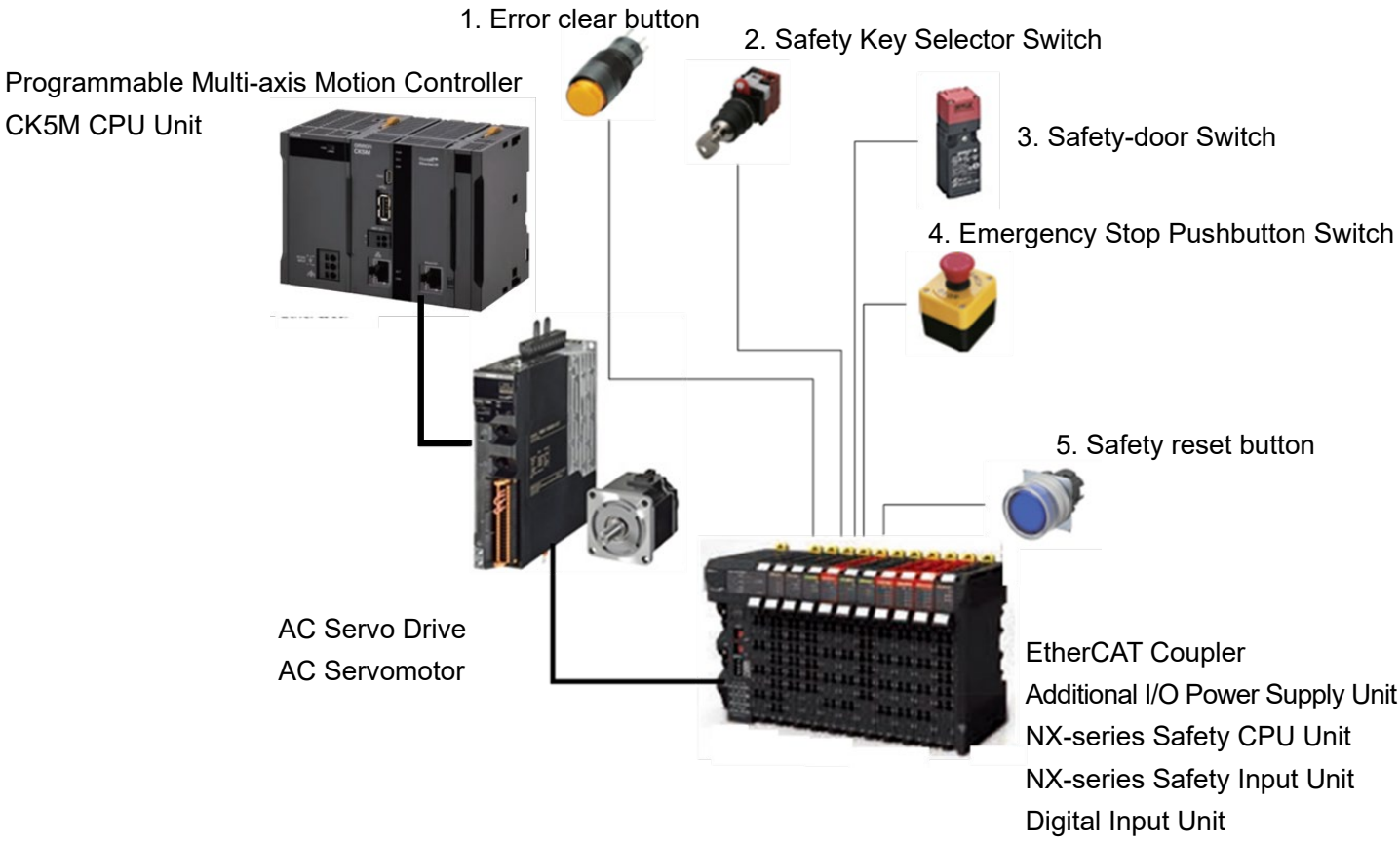
4.2. Adding the Safely-limited Speed (SLS) Function

This section describes how to add the SLS function to the project that you created in 3. Performing Setup.

The operation of the servo system set up in this section is explained below.

1. When the error clear button is pressed, the errors of the standard controller and Servo Drive are reset.
2. When the Safety Key Selector Switch is operated to switch to safety active mode, the standard controller changes the velocity command value to low speed. The Servo Drive activates the SLS function and monitors the motor velocity.
3. When the guard with the Safety-door Switch is opened while the SLS function is inactive, the motor torque is turned OFF.
4. When the Emergency Stop Pushbutton Switch is pressed, the motor torque is turned OFF.
5. When the safety reset button is pressed, the STO status is reset.

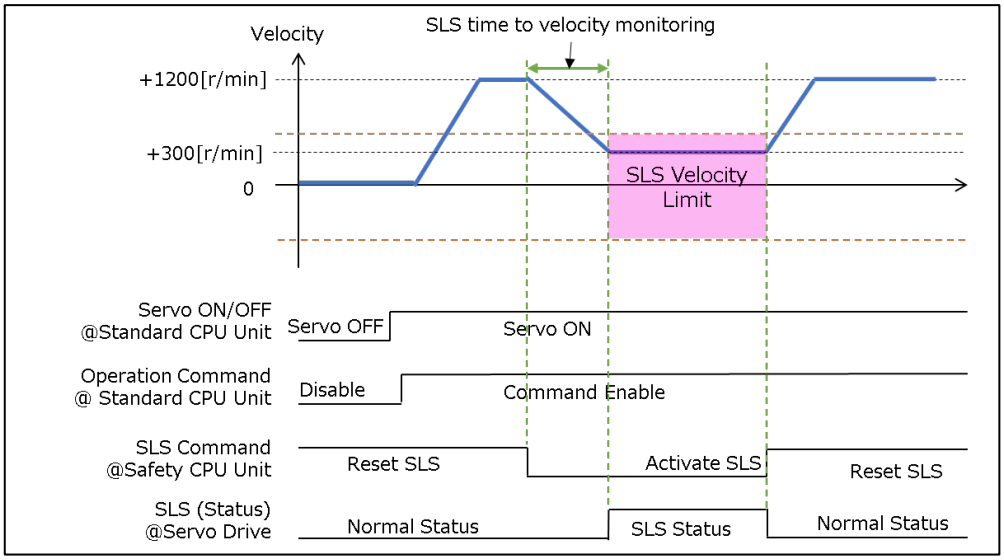
■ System Configuration



Input device	State	Operation
1. Error clear button	ON	Enable error reset command
	OFF	Disable error reset command
2. Safety Key Selector Switch	Normal operating mode	Run Servomotor at 1200 [r/min] and deactivate SLS function.
	Safety active mode	Run Servomotor at 300 [r/min] and activate SLS function. When velocity exceeds SLS velocity limit, Servo Drive goes into STO state and SLS Excessive Limit Value Error occurs.
3. Safety-door Switch	Open	SLS function deactivated: Enable STO command SLS function activated: Disable STO command
	Close	Disable STO command
4. Emergency Stop Pushbutton Switch	ON	Enable STO command
	OFF	Disable STO command
5. Safety reset button	ON	Enable reset STO status command
	OFF	Disable reset STO status command

■ Operation of SLS Function with Motion Control

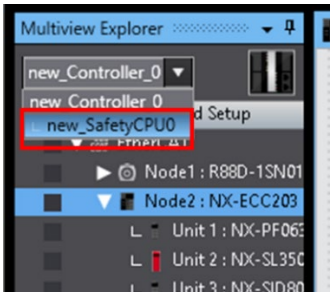
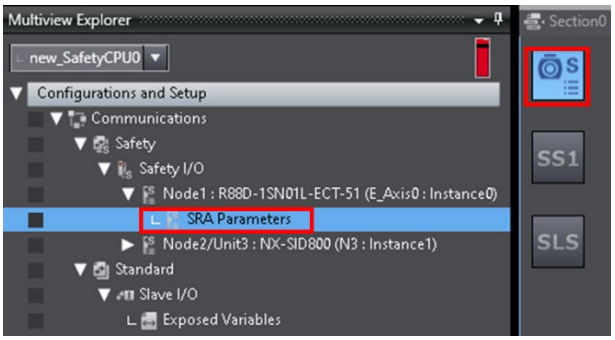
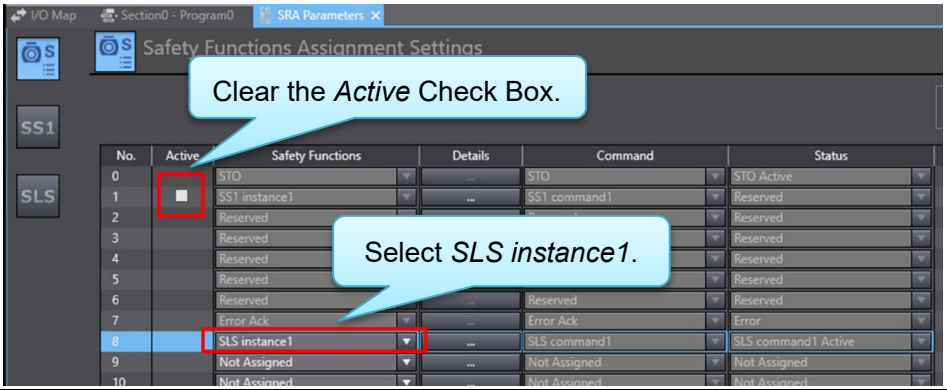
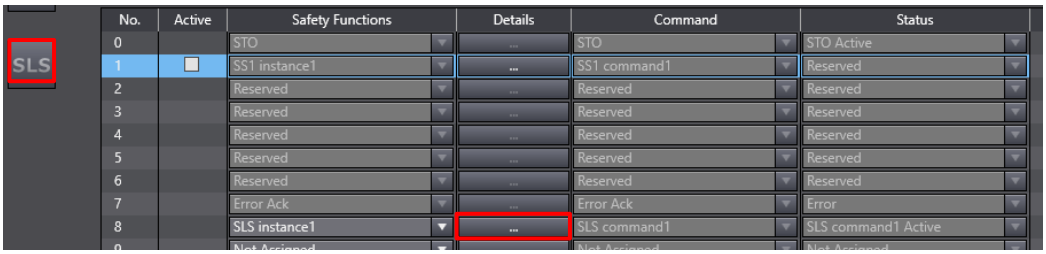
1. When the Servo ON command is enabled, the Servo Drive turns ON the Servo.
2. When an operation command is enabled, the command velocity to the Servomotor is 1,200 r/min.
3. When the SLS function is executed, the Servo Drive shifts to the SLS state after the wait time (SLS time to velocity monitoring 1) and monitors the motor velocity.
   The standard controller sets the command velocity to the Servomotor to 300 r/min.
4. When the SLS function is released, the Servo Drive goes into the normal state and stops monitoring the motor velocity.
   The standard controller sets the command velocity to the Servomotor to 1,200 r/min.



Precautions for Correct Use

- When using the SLS function, depending on the operation conditions or operating environment in which it is used, an error may occur during normal operation.
  For further details, refer to *8-4 Safely-limited Speed (SLS) Function* in the *1S-series AC Servomotors/Servo Drives with Built-in EtherCAT Communications and SS1/SLS Safety Sub-Functions User's Manual* (Cat. No. I696).

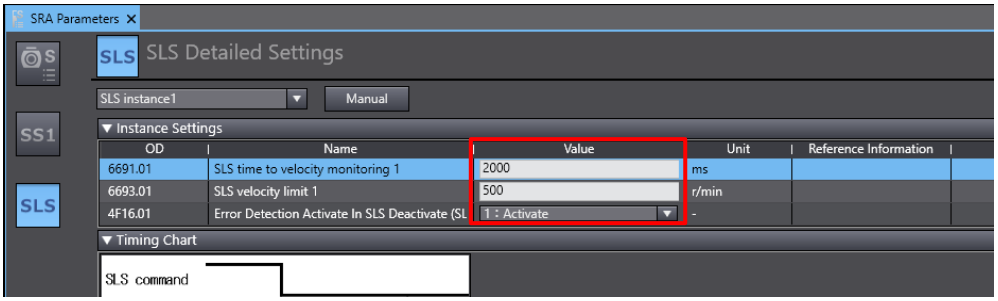
4.2.1. Setting the Safety Controller

1.	Click *new_SafetyCPU0*. 
2.	Double-click *SRA Parameters* and click the *Safety Function Assignment Settings* Button. 
3.	Assign SLS instance1. 
4.	Click the SLS Button to display the SLS Detailed Settings view. You can also use the ... Button to display the SLS Detailed Settings view. 

5. Set SLS parameters.

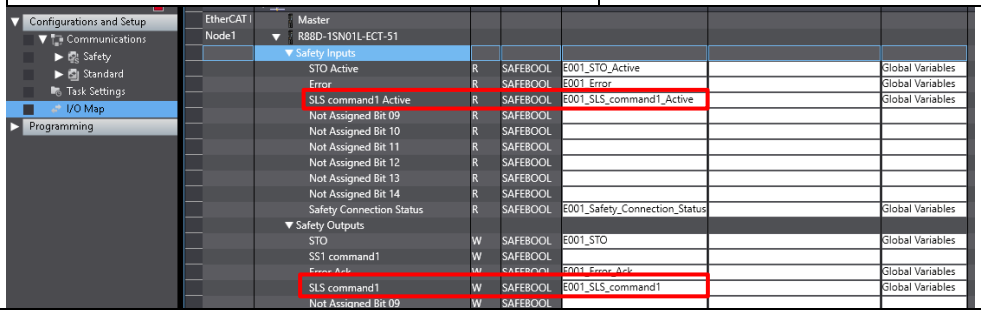
In this Guide, set as follows:

Name	Value	Unit
SLS time to velocity monitoring 1	2,000	ms
SLS velocity limit 1	500	r/min
Error Detection Activate In SLS Deactivate	Activate	---

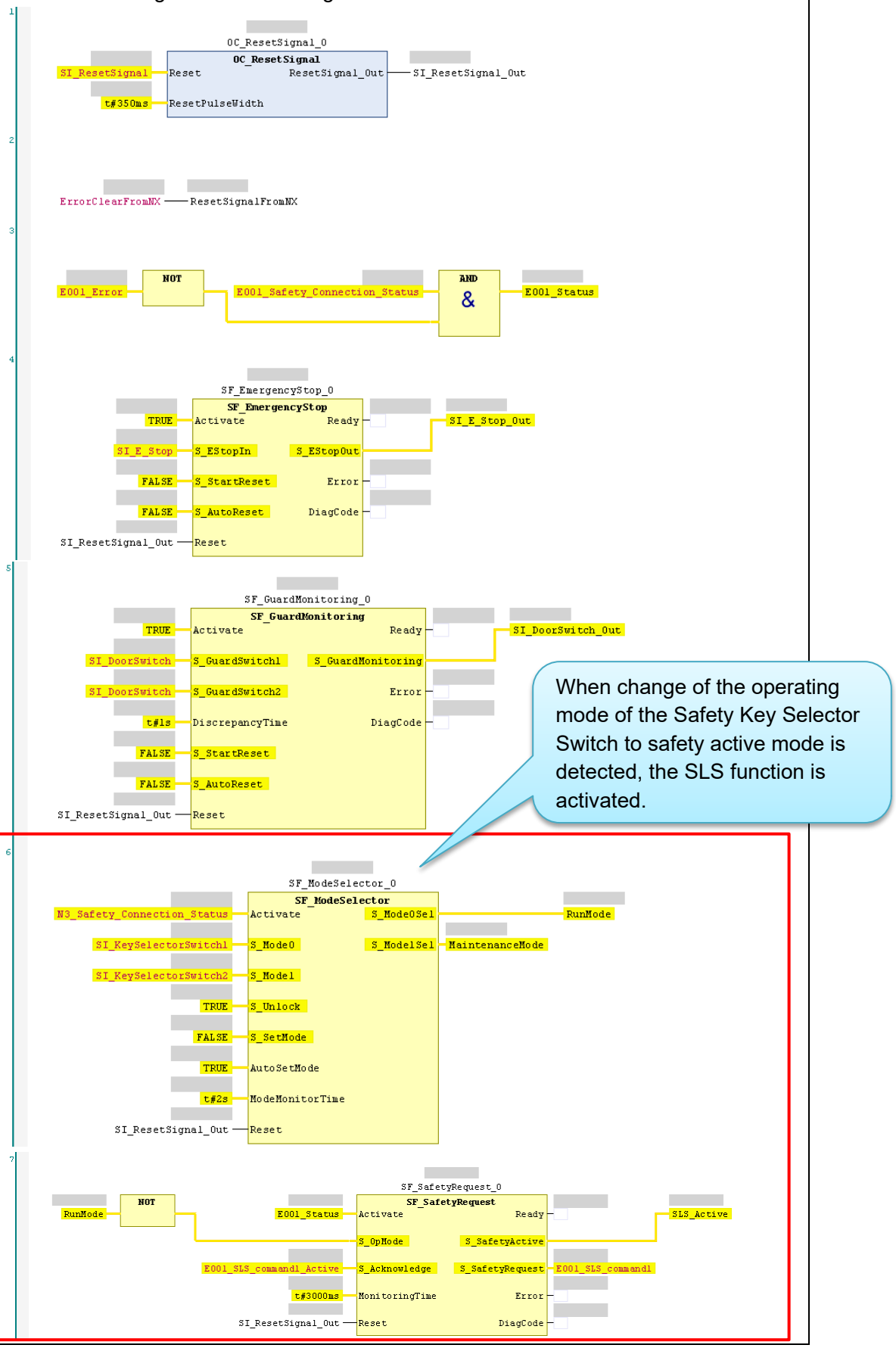


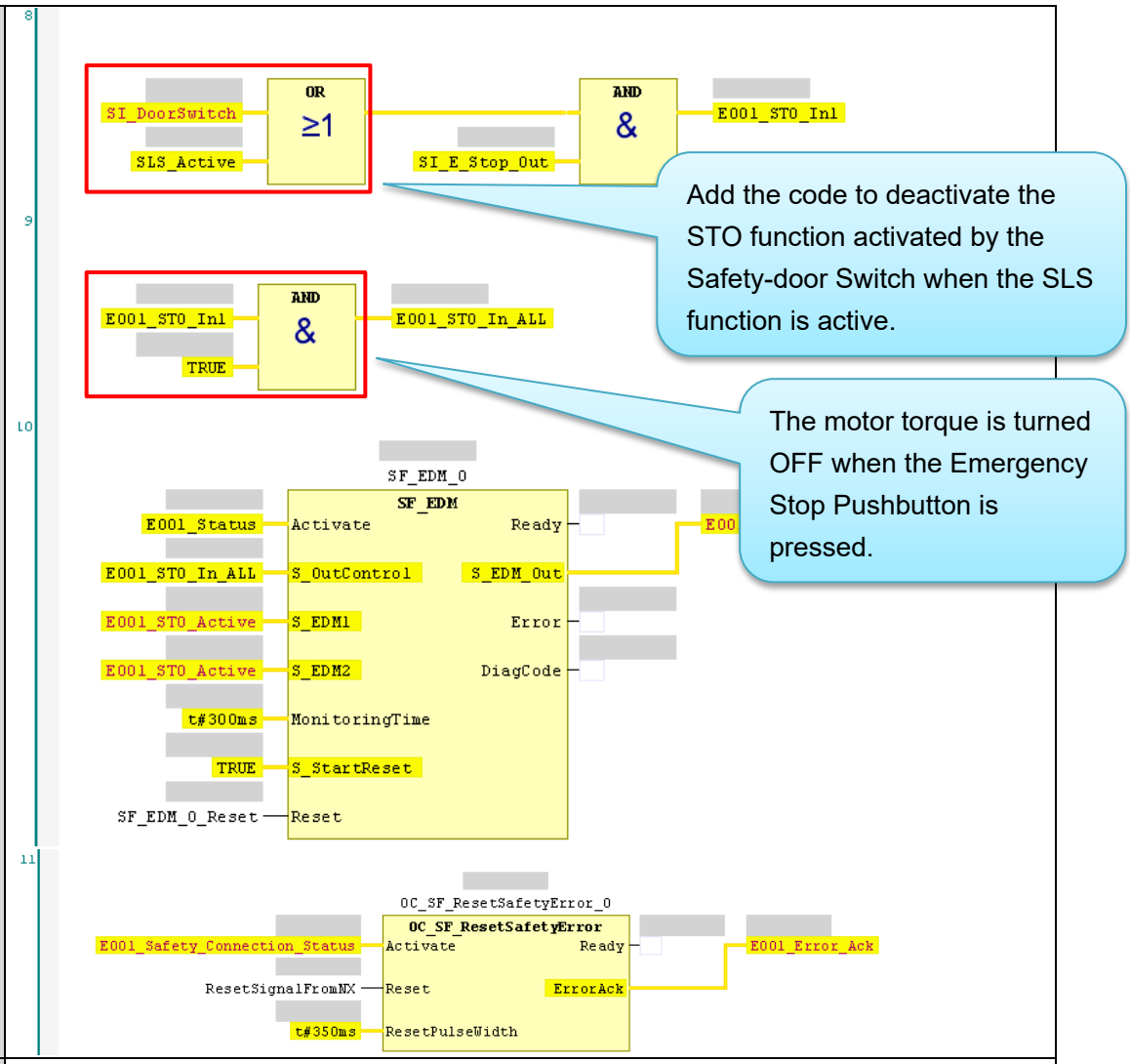
6. Open the I/O Map and create device variables by right-clicking the port to set and clicking *Create Device Variable* from the menu.

Port	Variable name
SLS command1 Active for R88D-1SN01L-ECT-51	E001_SLS_command1_Active
SLS command1 for R88D-1SN01L-ECT-51	E001_SLS_command1

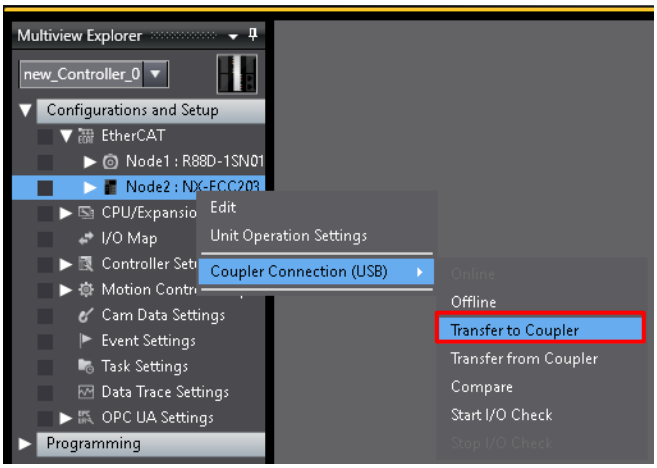


7.	Create a safety program. Add the following code to AutoProgram1.

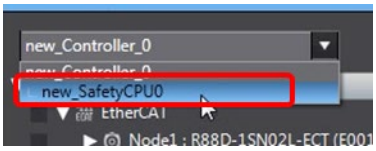
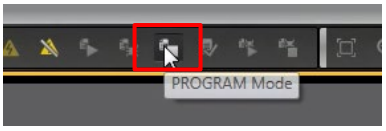
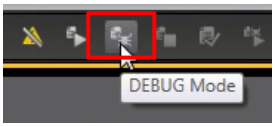
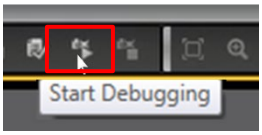
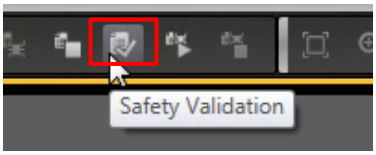
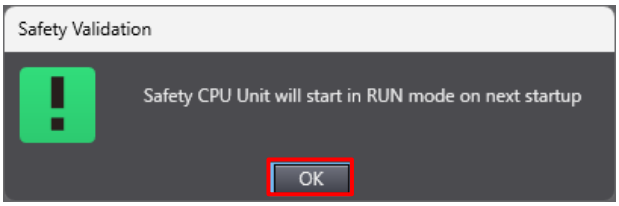
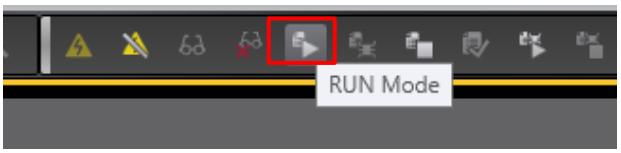


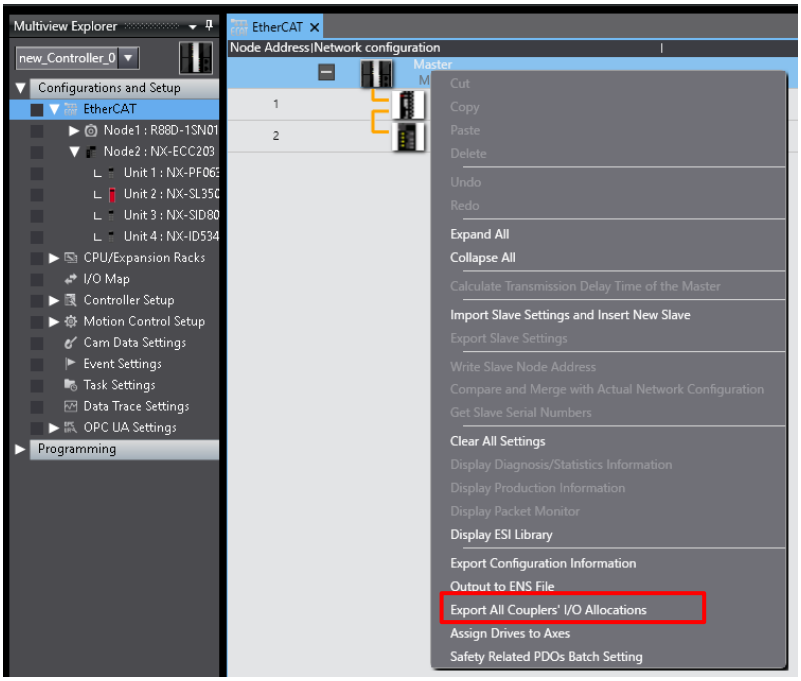


8. Execute *Transfer to Coupler* to transfer the settings.

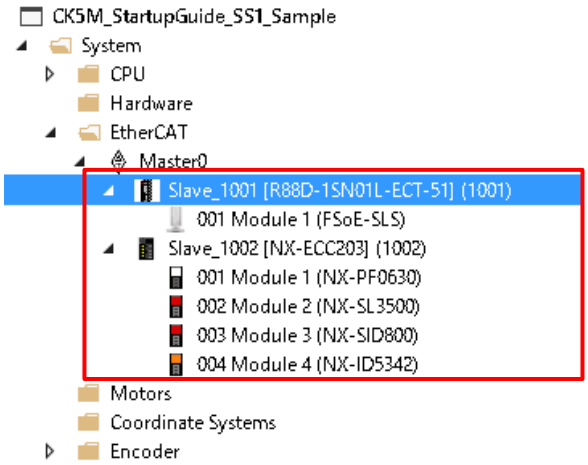
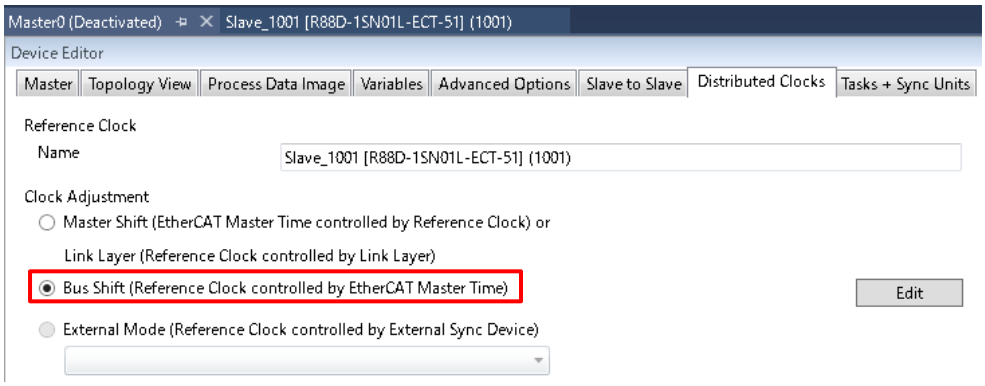
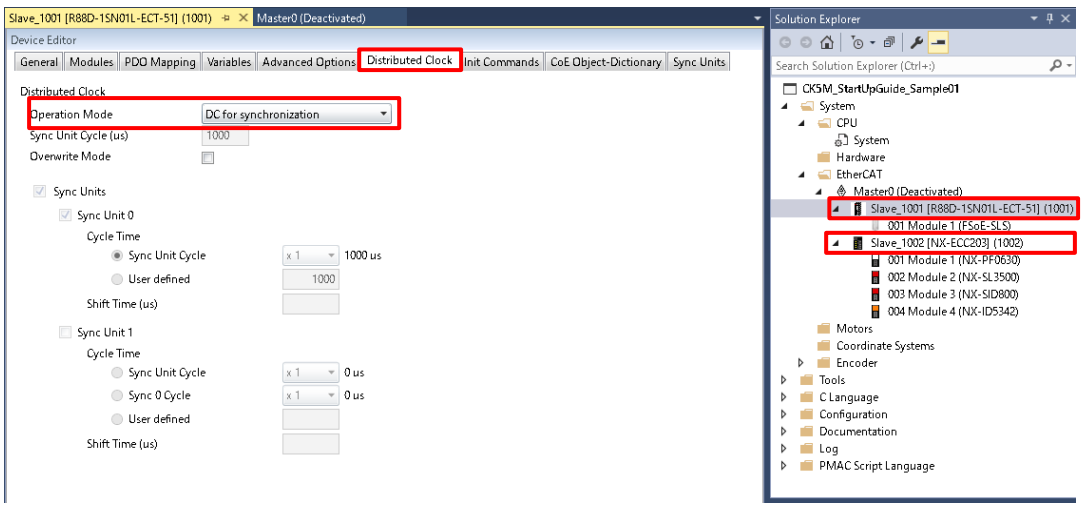


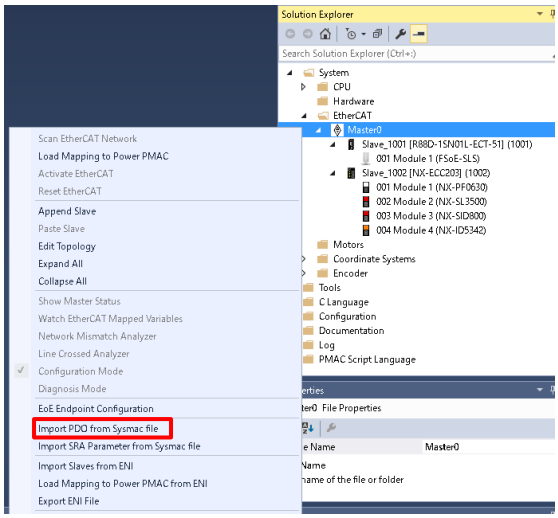
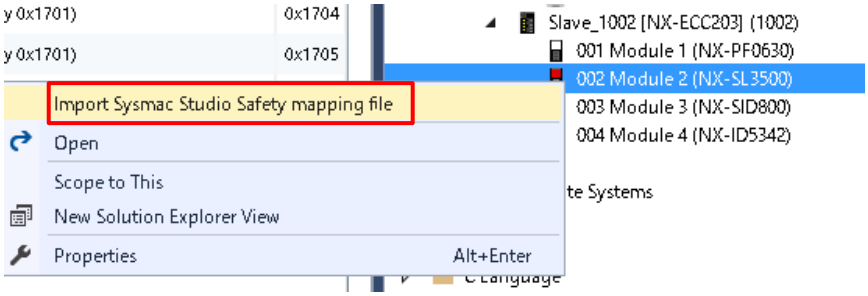
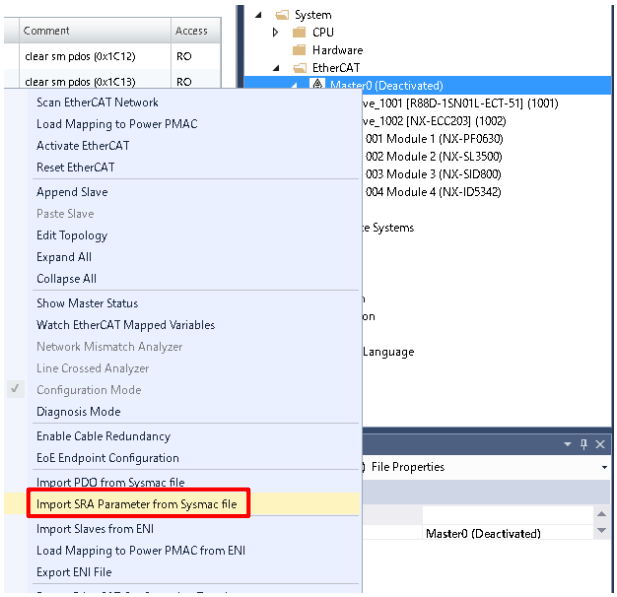
Note: Refer to Transfer of Settings and Projects for details on the procedure.

9.	Download the safety application. Select *new_SafetyCPU0* from the list.  Click the **PROGRAM Mode** Button to switch to PROGRAM mode.  Click the **DEBUG Mode** Button to enter DEBUG mode.  Click the **DEBUG Mode** Button to start DEBUG mode.  Click the **Safety Validation** Button.  The safety application is now ready to run.  Click the **RUN Mode** Button. 

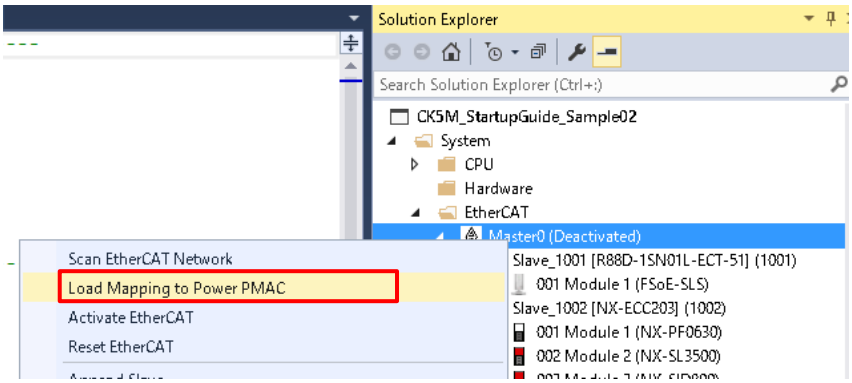
10.	Execute *Export All Couplers' I/O Allocations*.  Note: Refer to Export Procedure for the Safety Configuration File for details on the procedure.

4.2.2. Setup of the CK5M

1.	Execute *Scan EtherCAT Network* to scan the current configuration.  Note: Refer to Configuring EtherCAT Master Communications in section 3.4.3 for details on the procedure.
2.	Open the Distributed Clocks Tab Page for *Master0 (Deactivated)* and select the *Bus Shift (Reference Clock controlled by EtherCAT Master Time)* option for *Clock Adjustment*.  Note: Refer to Configuring Distributed Clock Settings in section 3.4.3 for details on the procedure.
3.	Open *Slave_1001 [R88D-1SN01L-ECT-51] (1001)* and *Slave_1002 [NX-ECC203](1002)* from the Solution Explorer and display the Distributed Clock Tab Page. Set *Operation Mode* to *DC for synchronization*. 

4.	Decompress the zip file exported from the Sysmac Studio (see 4.2.1 Setting the Safety Controller), import *SlaveMemoryMap.xml* from the folder, and configure the PDO to open it.  Note: Refer to Configuring PDO Map Settings in section 3.4.3 for details on the procedure.
5.	Import *CouplerMemoryMap.xml* from the folder decompressed in step 4, and set up the safety controller.  Note: Refer to Setting the Safety Controller in section 3.4.3 for details on the procedure.
6.	Import *SraParameterList.xml* from the folder decompressed in step 4, and set the SRA parameters.  Note: Refer to Setting the SRA Parameters in section 3.4.3 for details on the procedure.

7. Create an EtherCAT network configuration file.



Note: Refer to Creating an EtherCAT Network Configuration File in section 3.4.3 for details on the procedure.

8. Set up the Servomotor.
Set it in the same way as in 3.4.4 Setting the Axis Variables.

9. Modify the operation program. Add the following codes to *plc1.plc*.
The codes shown in the red frame below are added or modified from the program that you created in 3.4.5 Creating an Operation Program.

```
/*For more information see notes.txt in the Documentation folder */

global moveSequence = 0;                    //for plc1 Sequence
global SLSSequence = 0;                     //SLS Sequence
global isResetButton = false;           //Check if the reset button is pressed
global STOCommand = true;                   //sto command(true:Deactivate)
global SLSCommand = true;                   //sls command(true:Deactivate)
global STOStatus = false;                   //sto command(false:Deactivate)
global SLSStatus = false;                   //ss1 command(false:Deactivate)

open plc 1

        while(sys.ecatMasterReady==0){};
        isResetButton = (Slave_1002_NX_ECC203_1002_6042_1_StandardInput1stWord >> 6) & 1;
        STOCommand = (Slave_1001_R88D_1SN01L_ECT_51_1001_4130_130_MirrorSafetycontrolword & 1);
        SLSCommand = ((Slave_1001_R88D_1SN01L_ECT_51_1001_4130_130_MirrorSafetycontrolword>>8)&1);
        STOStatus = (Slave_1001_R88D_1SN01L_ECT_51_1001_4130_131_MirrorSafetystatusword & 1);
        SLSStatus = ((Slave_1001_R88D_1SN01L_ECT_51_1001_4130_131_MirrorSafetystatusword>>8)&1);

        switch(moveSequence)
        {
                case 0://Setting
                        //EtherCAT Enabled
                        ECAT[0].Enable=true;
                        //Positive/Negative torque limit value set to 30%
                        Slave_1001_R88D_1SN01L_ECT_51_1001_60E0_0_Positivetorquelimitvalue = 300;
                        Slave_1001_R88D_1SN01L_ECT_51_1001_60E1_0_Negativetorquelimitvalue = 300;
                        Ldata.motor = 1;
                        moveSequence = 1;
                        break;

                case 1://Check if the reset button is pressed.
                        if(isResetButton == 1)
                        {
                                moveSequence = 2;
                        }
                        break;
```

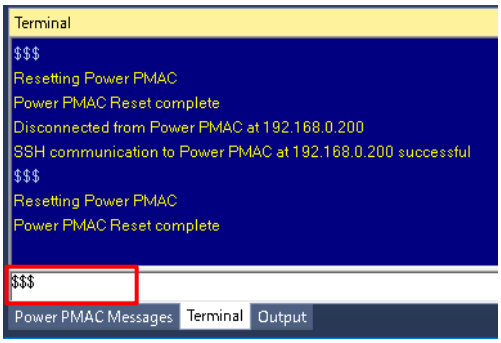
```
                case 2: //servo ON
                        jog/; //servo ON & Full closed loop setting

                        //wait 3sec
                        P1000 = Sys.Time+3;
                        while(P1000>Sys.Time){};

                        moveSequence = 3;
                        break;
                case 3: //ckeck the servo ON & STO deactivate
                        if(Motor[1].AmpEna == true && STOStatus == false)
                        {
                                if(SLSCommand == false || SLSStatus == true)
                                {
                                        switch(SLSSequence)
                                        {
                                                case 0:
                                                        jog/;     //servo OFF Once
                                                        SLSSequence = 1;
                                                        break;
                                                case 1:
                                                        Motor[1].JogSpeed=300/60*(1<<23)/1000//Jog velocity per ms
                                                        jog+; //move motor(300rpm)
                                                        break;
                                        }
                                }
                                else
                                {
                                        Motor[1].JogSpeed=1200/60*(1<<23)/1000 //Jog velocity per ms
                                        jog+; //move motor(1200rpm)
                                        SLSSequence = 0;
                                }
                        }
                        else
                        {
                                kill;//servo OFF
                                moveSequence = 1; //Start over from sequence1
                        }
                        break;

                default:
                        moveSequence = 0;
                        break;
        }
        //error clear switch
        Slave_1002_NX_ECC203_1002_7002_1_E002Slot2ErrorClearFromNX =
Slave_1002_NX_ECC203_1002_6062_1_InputBit16bits;


close
```

10. Add the following code to the end of the program in the programming area of the pp_startup.txt Tab Page.

```
enable plc 1;
```

11. Download the created project to the Controller.
Note: Refer to 3.4.6 Downloading the Project and Checking the Operation for details on the procedure.

4.2.3. Checking Operation of the SLS Functions

1.	Check that the Safety Key Selector Switch is in normal operating mode. 
2.	Execute the *$$$* command from the Terminal of the Power PMAC IDE to restart the CK5M. 
3.	After the restart of the Controller is completed, press the safety reset button. 
4.	Check that the 7-segment LED display shows ‘oE.’. 
5.	After about three seconds, check that the Servomotor rotates at about 1,200 r/min.
6.	Operate the Safety Key Selector Switch to switch to safety active mode.  Check that the Servomotor rotates at about 300 r/min. Check that the 7-segment LED display shows ‘SF’. 

6.	Open the guard with the Safety-door Switch.  Check that the 7-segment LED display still shows ‘SF’.
7.	Close the guard and press the safety reset switch.
8.	Operate the Safety key Selector Switch to switch to normal operating mode.  Check that the Servomotor rotates at about 1200 r/min. Check that the 7-segment LED display shows ‘oE.’. 

Appendices

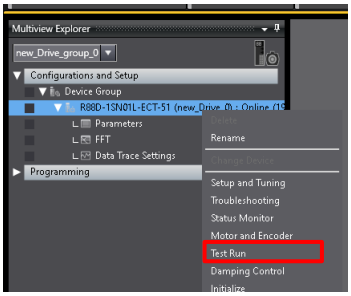
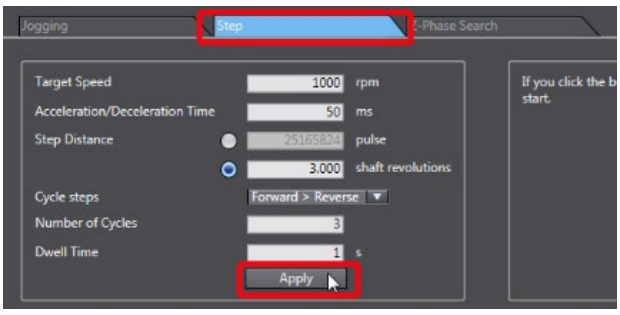
Test Run and Data Trace, *Easy Tuning*, *Manual Tuning*, and *Advanced Auto-Tuning* in Appendices are all performed using the Sysmac Studio.

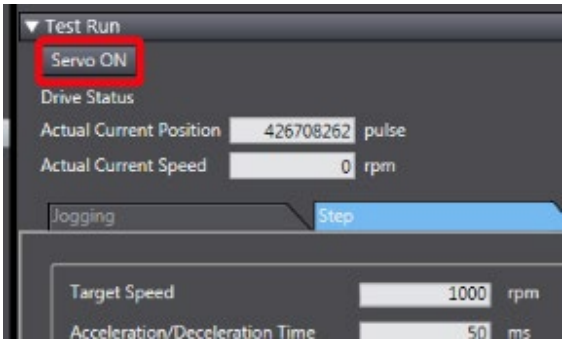
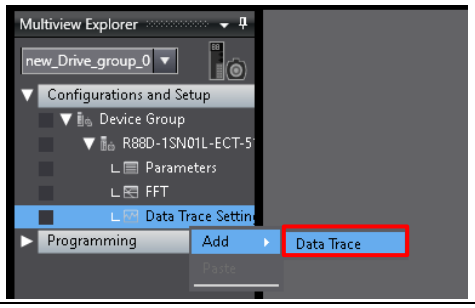
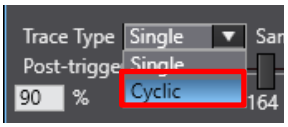
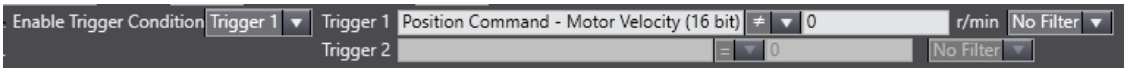
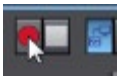
Before you proceed, refer to 3.5.1 Connecting the Servo Drive via CK5M and connect the Sysmac Studio to the Servo Drive.

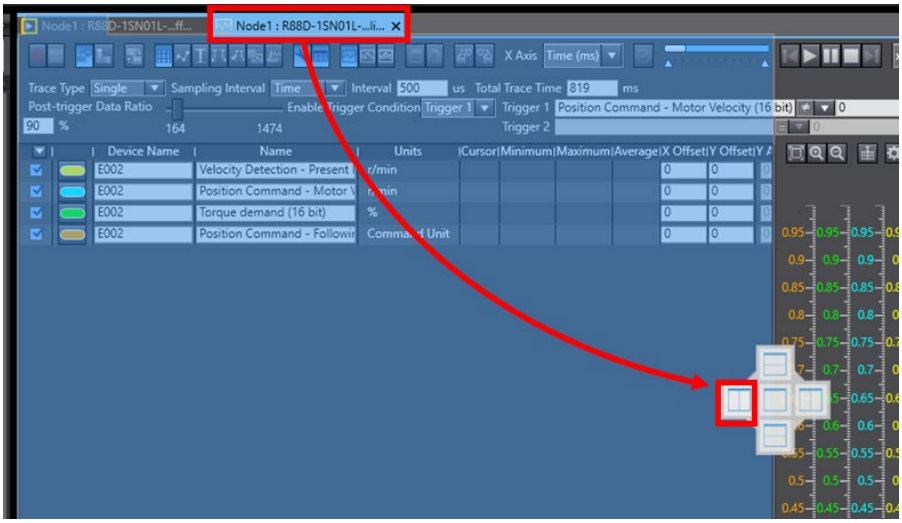
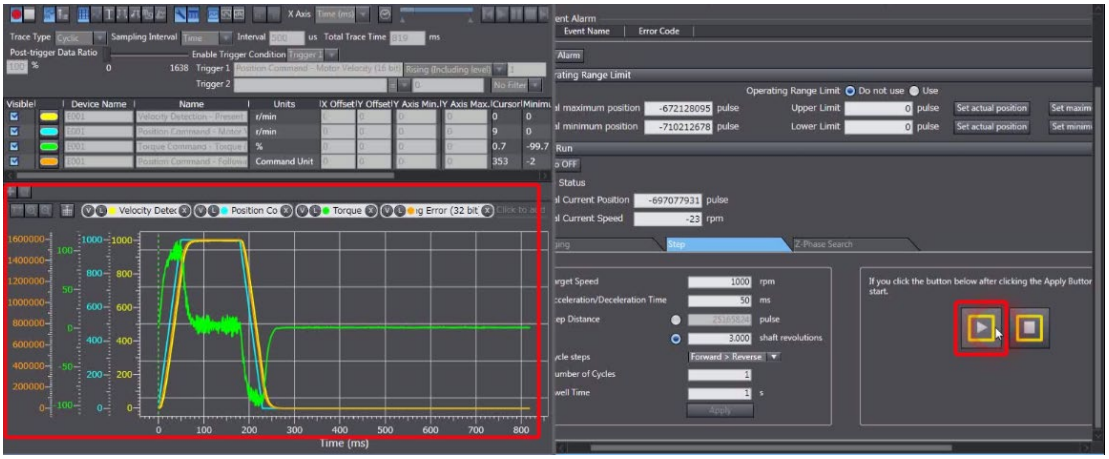
Note: Please add the following code to *plc1.plc*, which is included in the CK5M. The project that you created in 3.4 or section 4 cannot be used for trial run because it is designed to run the Servomotor automatically from servo ON.

```
/*For more information see notes.txt in the Documentation folder */

open plc 1

        while(sys.ecatMasterReady==0){};

        ECAT[0].Enable=true;
        //Positive/Negative torque limit value set to 30%
        Slave_1001_R88D_1SN01L_ECT_51_1001_60E0_0_Positivetorquelimitvalue = 300;
        Slave_1001_R88D_1SN01L_ECT_51_1001_60E1_0_Negativetorquelimitvalue = 300;

close
```

Test Run and Data Trace

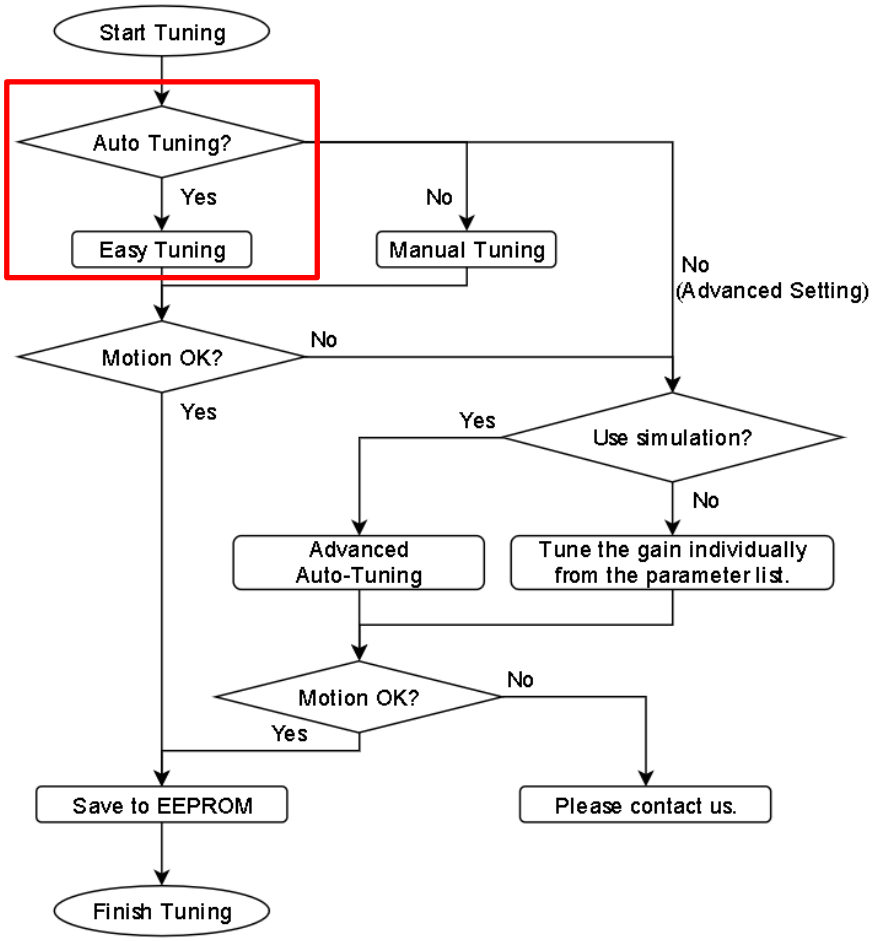
1.	Right-click the Servo Drive and select *Test Run* from the menu. 
2.	Click the *Step* Tab. Adjust the motion profile. Click the **Apply** Button. 

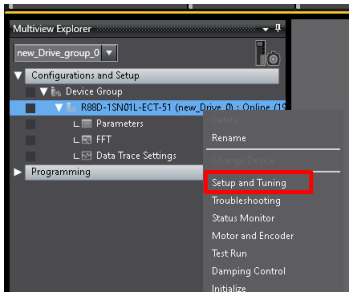
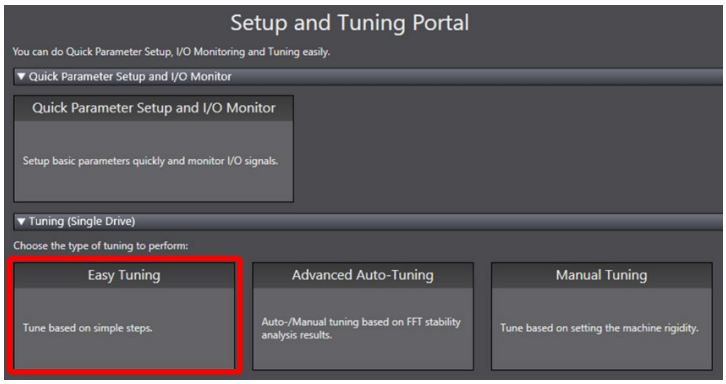
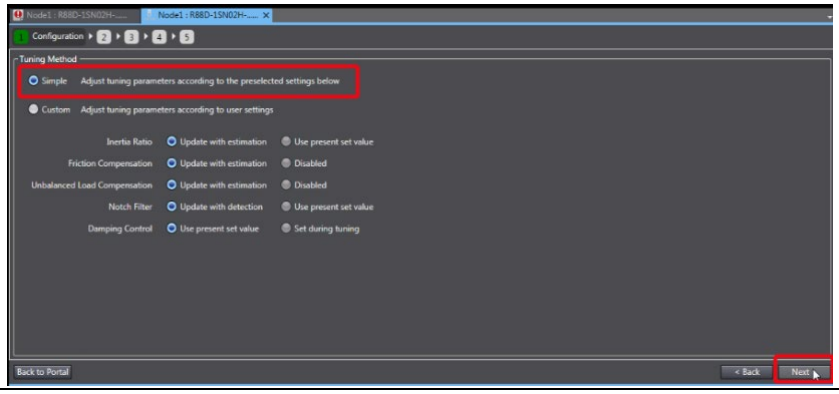
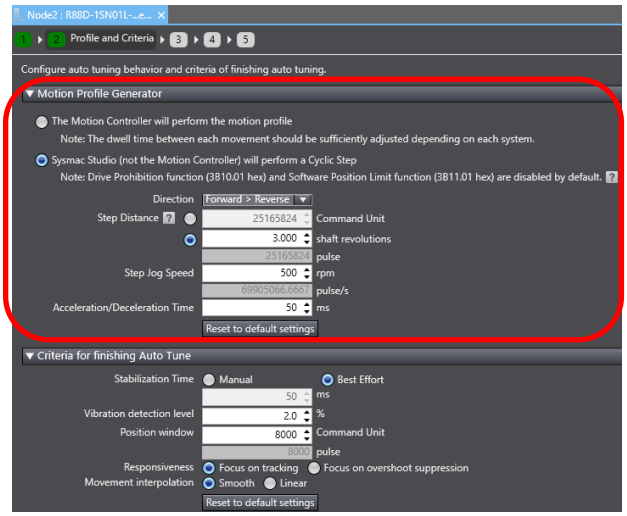
3.	Click the *Servo ON* Button to turn ON the Servo. 
4.	Right-click *Data Trace Settings* and select *Add - Data Trace* from the menu to add a new data trace. 
5.	Select *Cyclic* for the trace type. 
6.	Specify the sampling interval. 
7.	Specify the enable trigger condition. 
8.	Click the *Execute* Button. 

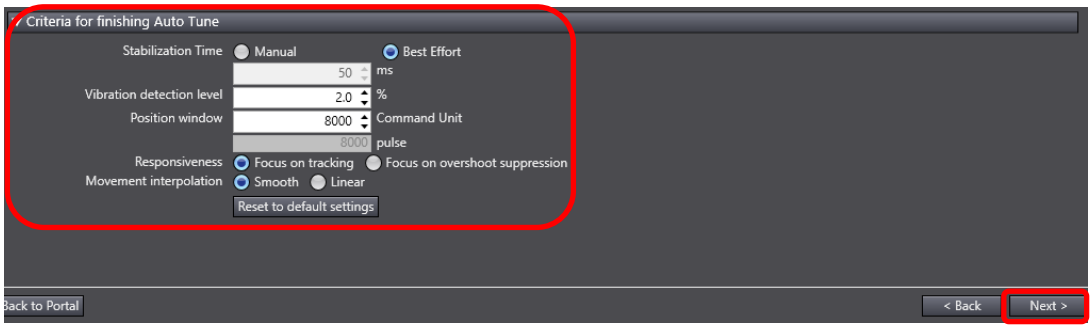
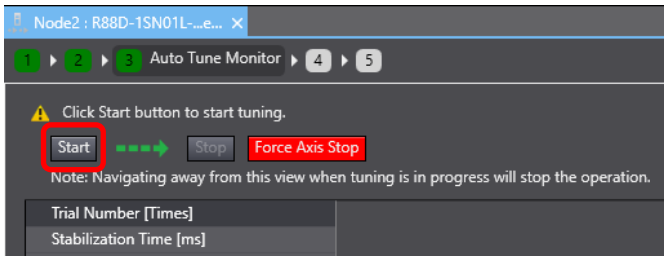
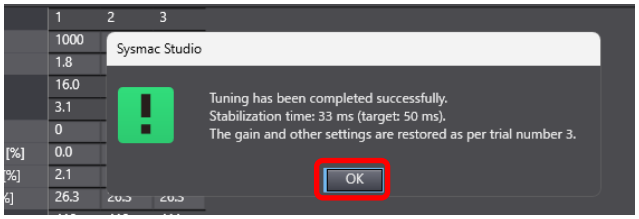
9.	Align the Test Run Window and the Data Trace Window horizontally by docking them.  Click the **Start** Button in the Test Run Window to run the Servomotor. Traced data will appear cyclically. 

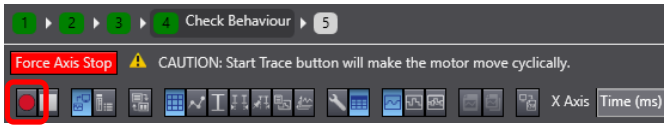
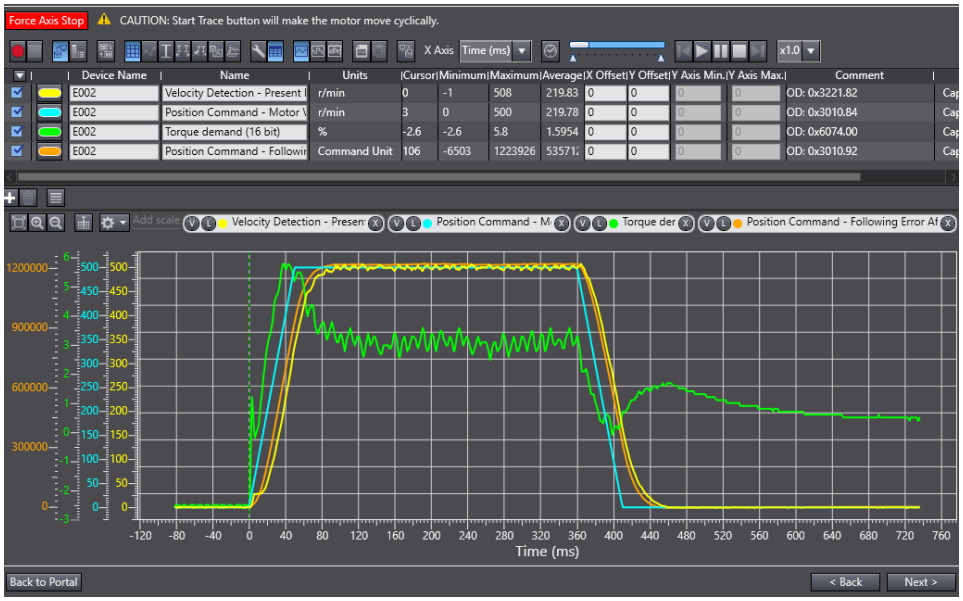
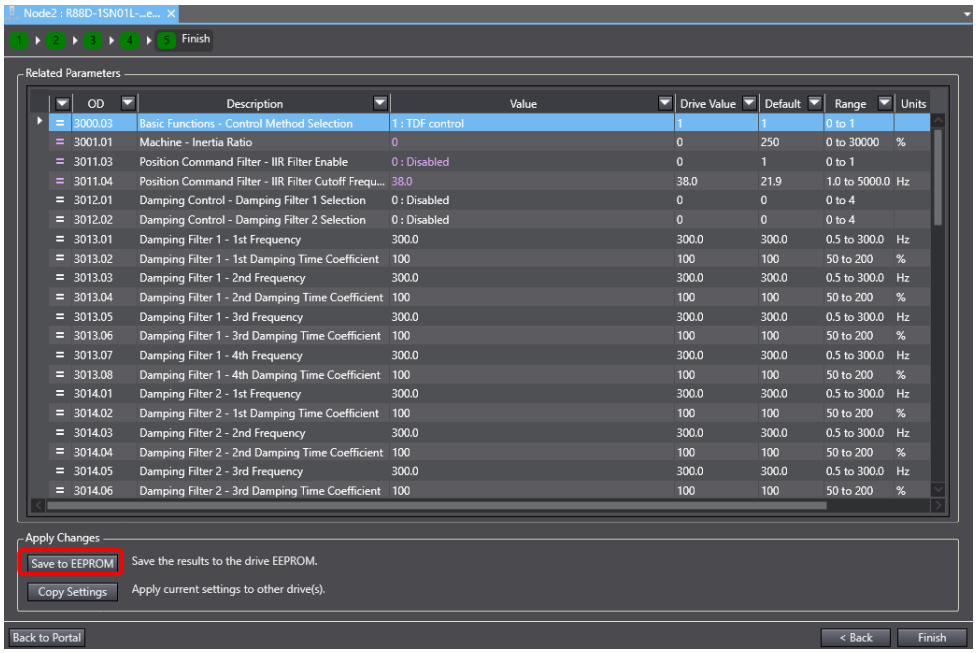
Easy Tuning

Use the following procedure to perform adjustment. It is recommended to perform Easy Tuning first.



1.	Right-click the Servo Drive and select *Setup and Tuning* from the menu. 
2.	Click the *Easy Tuning* Button. 
3.	Select *Simple* for the tuning method and click the *Next* Button. 
4.	Set the motion profile generator. 

5.	Set the criteria for finishing auto tuning. Click the **Next** Button.  ● When you select *Manual* for the stabilization time, gain will be increased gradually until the stabilization time reaches the specified time. Specify the following error for the position window to determine that the positioning is completed. If a vibration above the vibration detection level is detected during tuning, an adjustment failure will occur. ● When you select *Best Effort* for the stabilization time, gain will be increased gradually so that the vibration does not exceed the vibration detection level. Set the vibration detection level so that the machine does not vibrate. The lower the vibration detection level, the less likely it is to generate vibration, but the gain is less likely to increase.
6.	Click the *Start* Button. Be careful because the Servomotor will start running.  Easy Tuning has been completed.  Click the **Next** Button. 

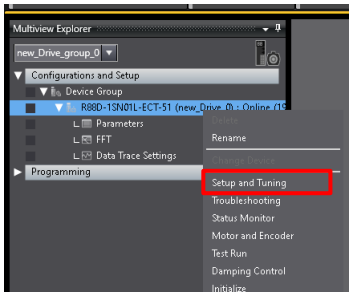
7.	Click the *Start Trace* Button. The Servomotor will run, and the traced data will be displayed in the graph area.   Click the **Next** Button. 
8.	Check the tuning results of the gain parameters. Click the *Save to EEPROM* Button to save the gain parameters.  Click the *Finish* Button. 

Manual Tuning

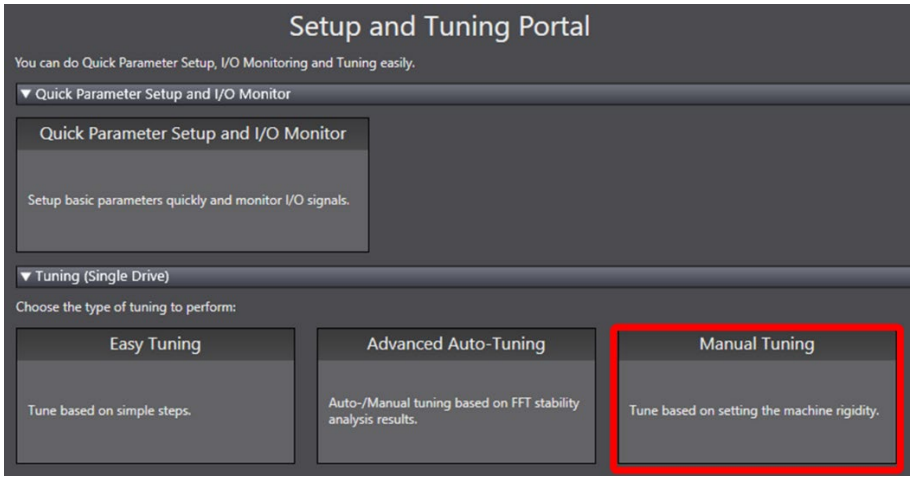
■ How to Perform Manual Tuning

This section describes how to change machine rigidity parameters for gain adjustment.

1. Right-click the Servo Drive and select *Setup and Tuning* from the menu.

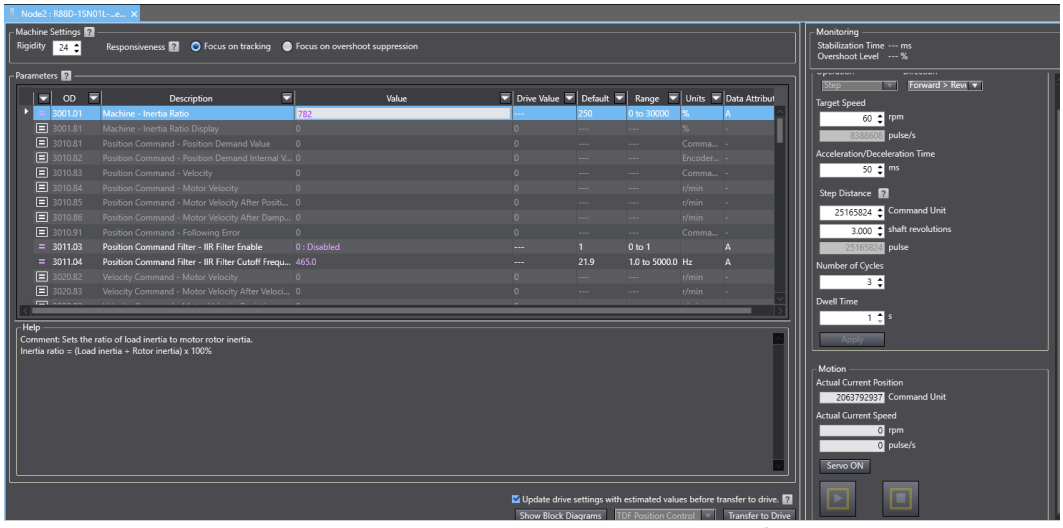


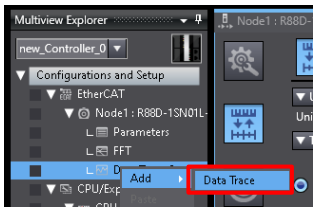
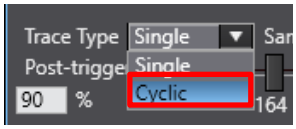
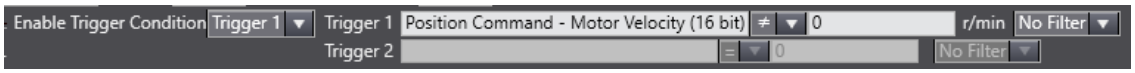
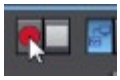
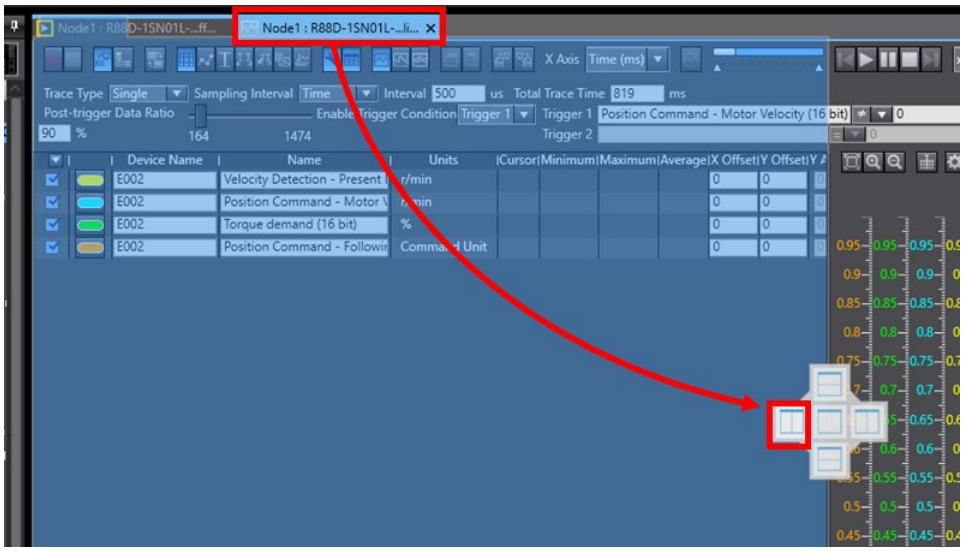
2. Click the *Manual Tuning* Button.

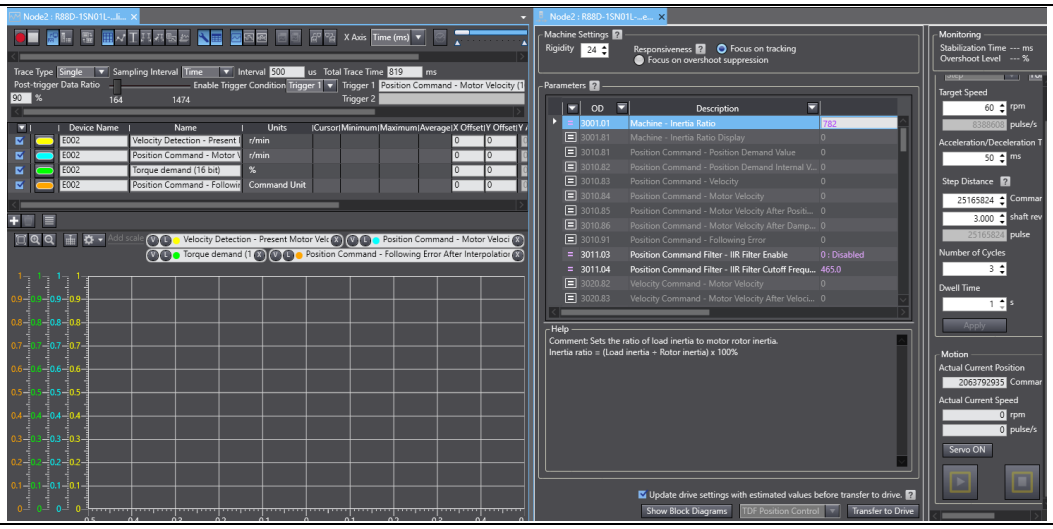


The Manual Tuning Window is displayed.

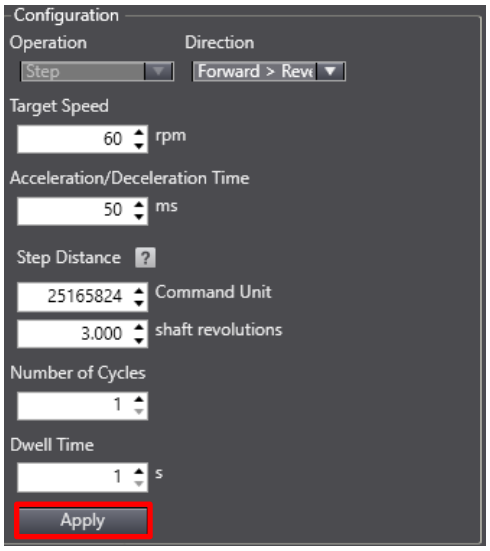
It includes machine rigidity settings, gain parameters, and Servo Drive test run.



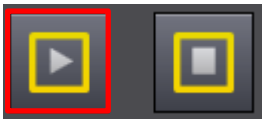
3.	In order to check the behavior of the Servomotor, right-click *Data Trace Settings* and select *Add - Data Trace* from the menu to add a new data trace. 
4.	Select *Cyclic*. 
5.	Adjust the sampling interval. 
6.	Adjust the enable trigger condition. 
7.	Click the *Transfer Parameters from Drive after Trace* Button to disable uploading parameters.  Click the *Execute* Button.  The Sysmac Studio waits for the trigger. 
8.	Align the Test Run Window and the Manual Tuning Window horizontally by docking them. 



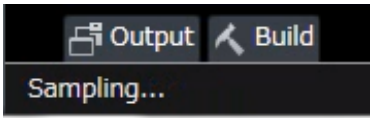
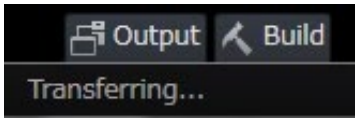
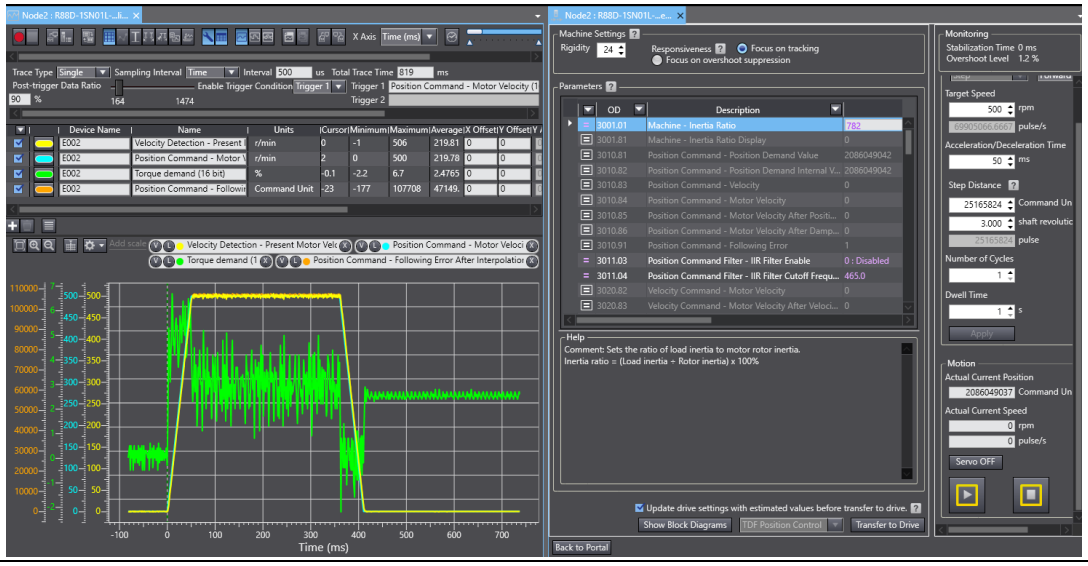
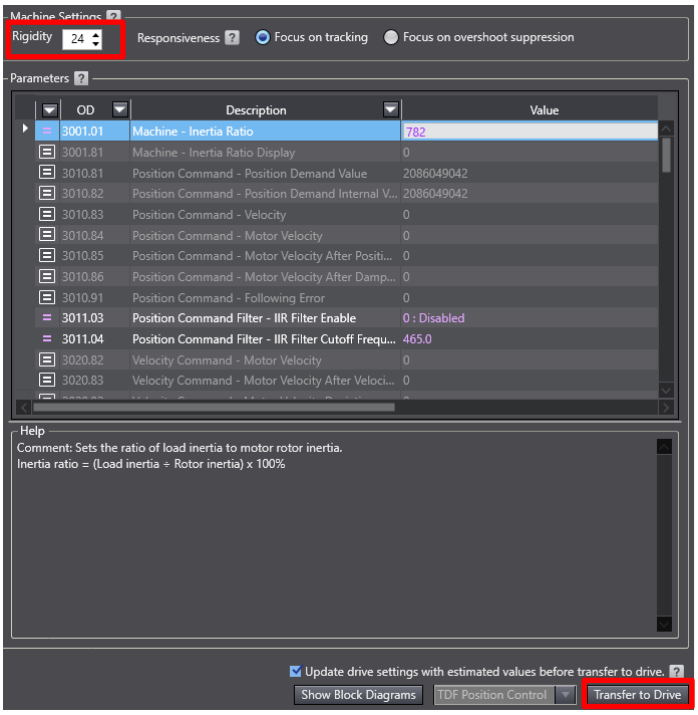
9. Configure the motion profile and click the **Apply** Button.



10. Click the *Servo ON* Button to turn ON the Servo and then click the *Start* Button.



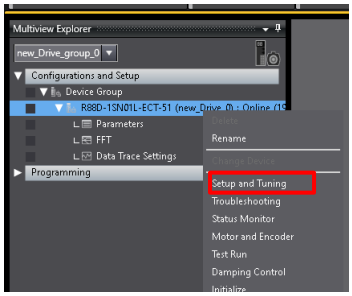
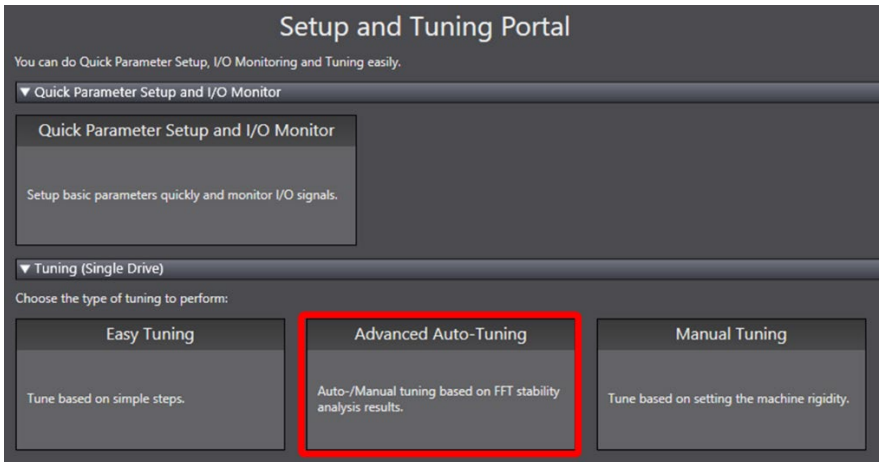
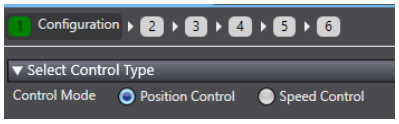
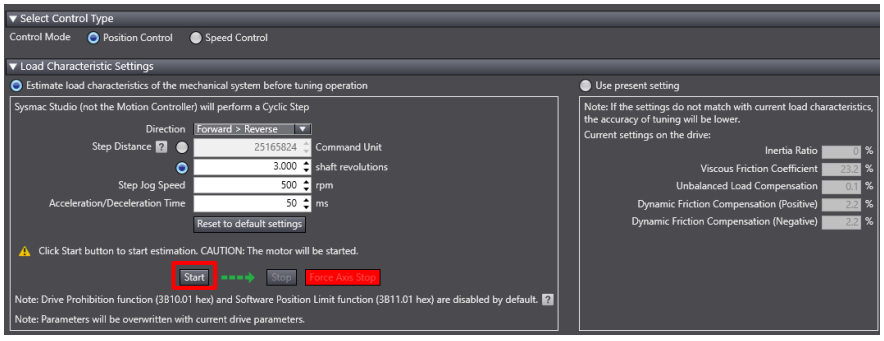
Be careful because the Servomotor will move in forward and reverse directions.

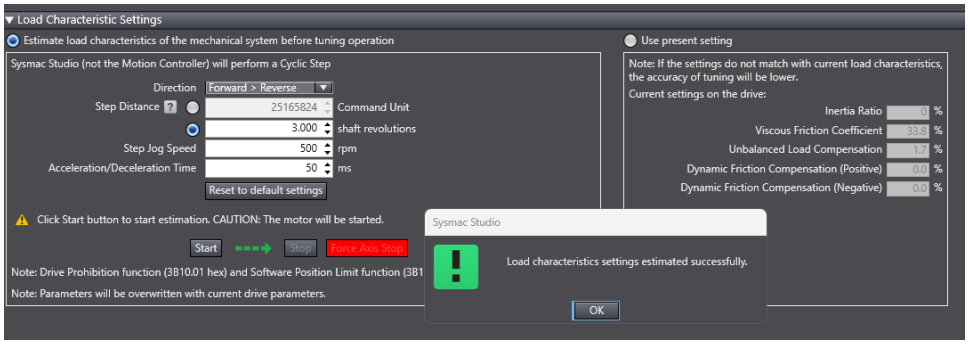
11.	Data trace is now triggered, and trace results are displayed.   Each time the Servomotor moves, traced data will appear cyclically. 
12.	It is possible to change gain values at once by changing the machine rigidity settings.  Click the **Transfer to Drive** Button to transfer the gain parameters to the Servo Drive.
13.	Repeat step 10, 11, and 12 until the desired performance is achieved. If vibrations occur, reduce the rigidity settings. If required, it is possible to increase responsiveness by applying notch filters in Advanced Auto-Tuning and adjusting gains. Refer to *Advanced Auto-Tuning* in Appendices.

Advanced Auto-Tuning

This function uses FFT measurement data-based simulation to adjust the gain and filter settings automatically. Repeating actual Servomotor operation is not necessary, and a fine adjustment is possible in a short period of time.

■ How to Perform Advanced Auto-Tuning

1.	Right-click the Servo Drive and select *Setup and Tuning* from the menu. 
2.	Click the *Advanced Auto-Tuning* Button. 
3.	Configuration (Wizard Step 1) Select the control mode.  Click the *Start* Button to estimate load characteristics. (The motor will move.) When easy tuning has been performed already to determine load characteristics, select *Use present setting* for the load characteristic settings.  Load characteristics have been updated.

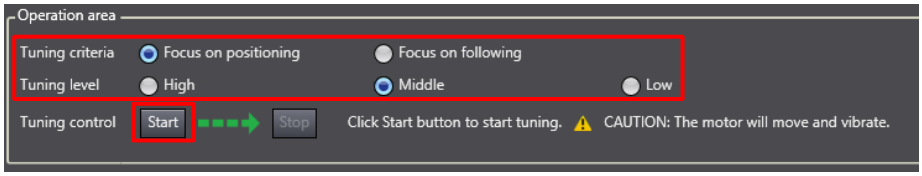


Click the **Next** Button.

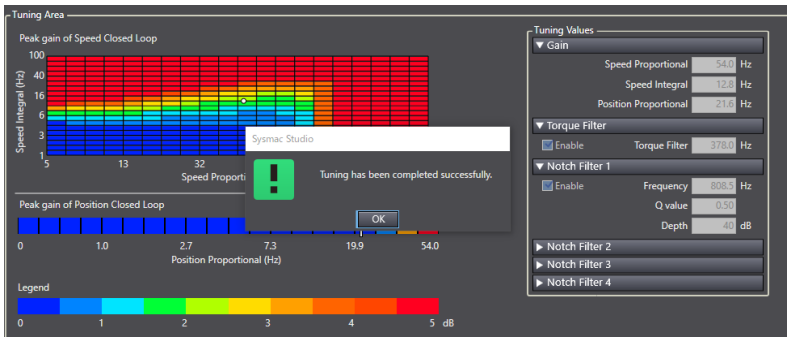
4. Advanced Auto-Tuning (Wizard Step 2)

In the operation area, configure the tuning settings.

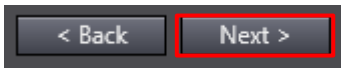
After completing the settings, click the Start button to start tuning.



After a while, the results are displayed in the tuning area.



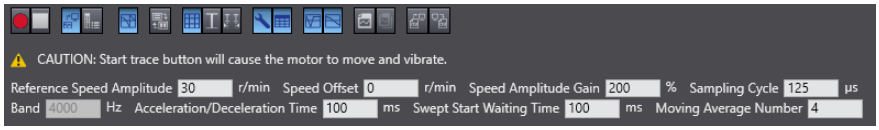
Click the **Next** Button.



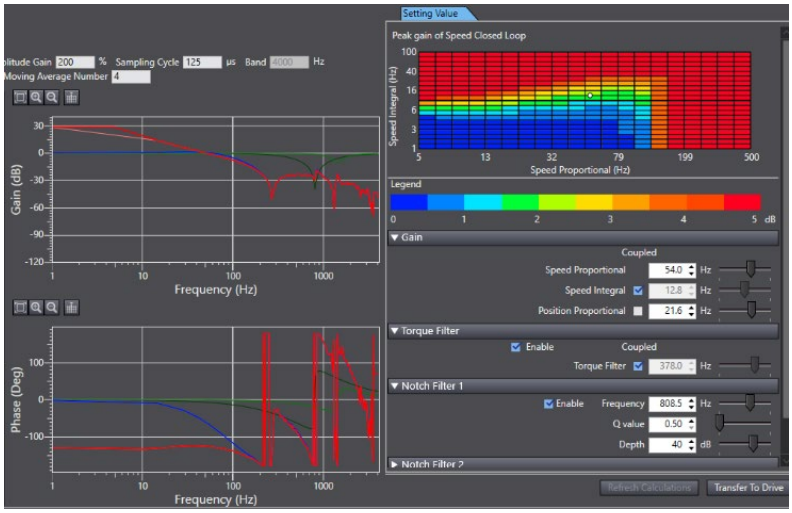
5.

Frequency Response Simulation (Wizard Step 3)

Start tracing. (The Servomotor moves, and frequency characteristics are measured.)

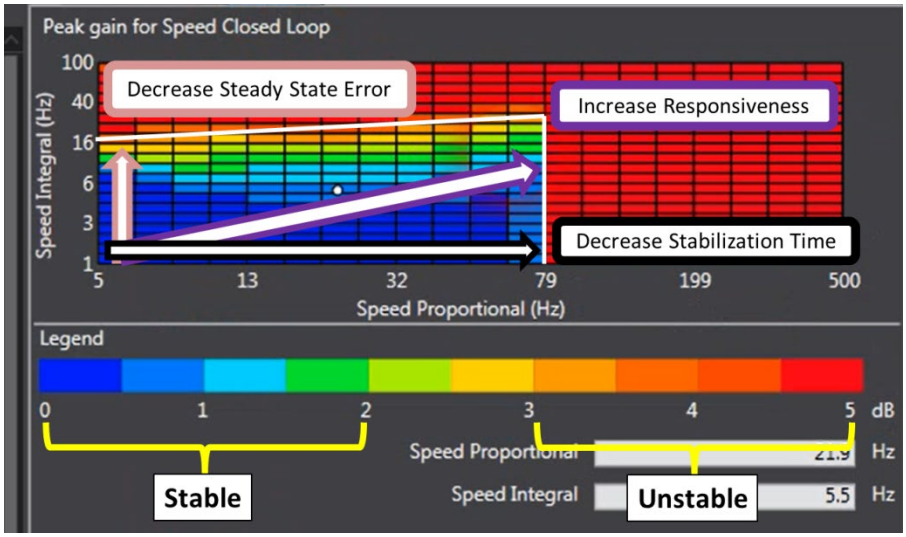


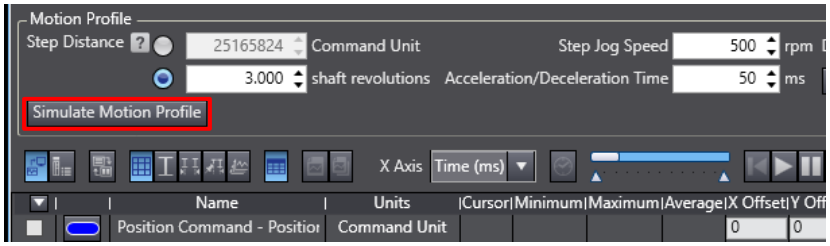
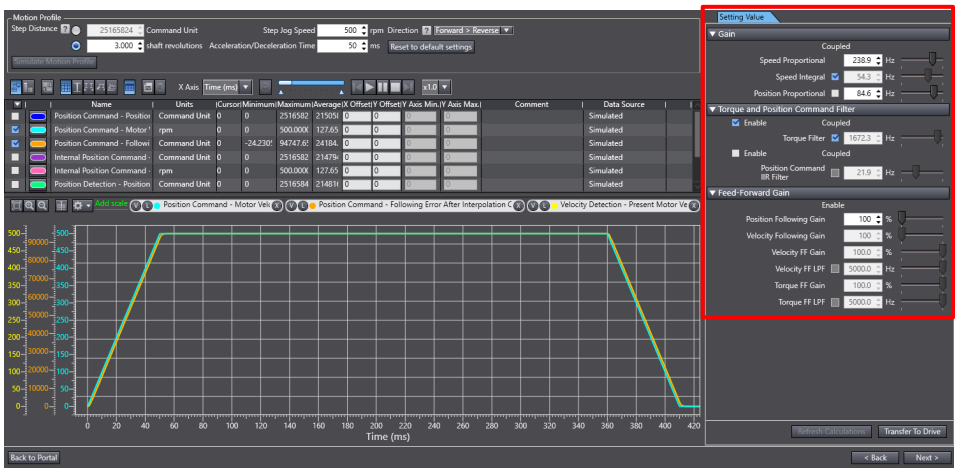
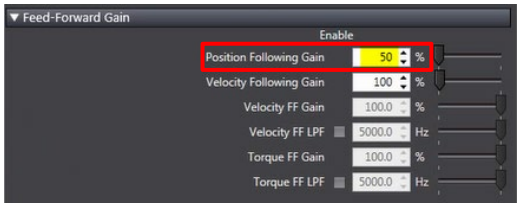
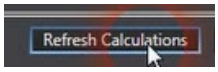
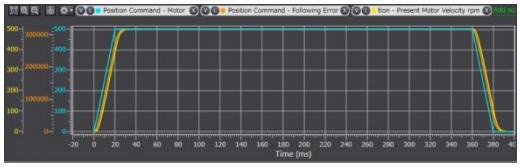
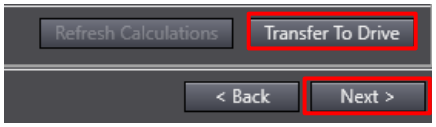
Gain and phase characteristics that were measured by FFT and simulated are displayed.

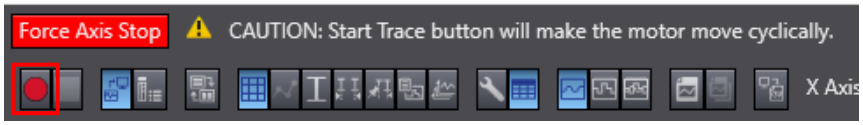
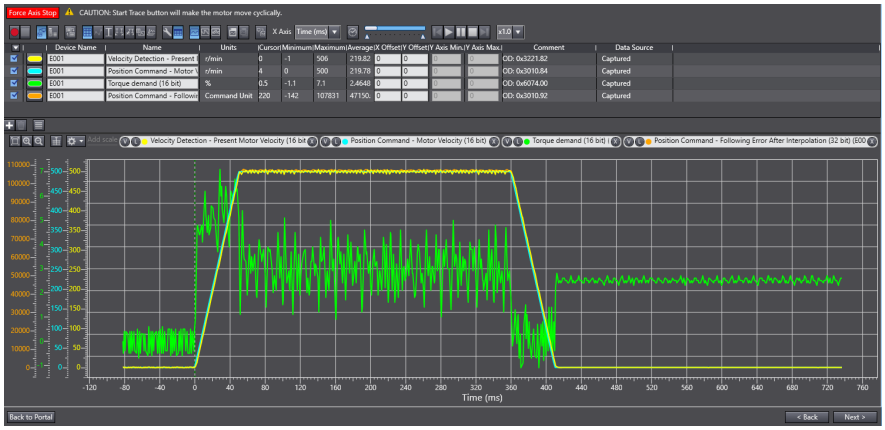
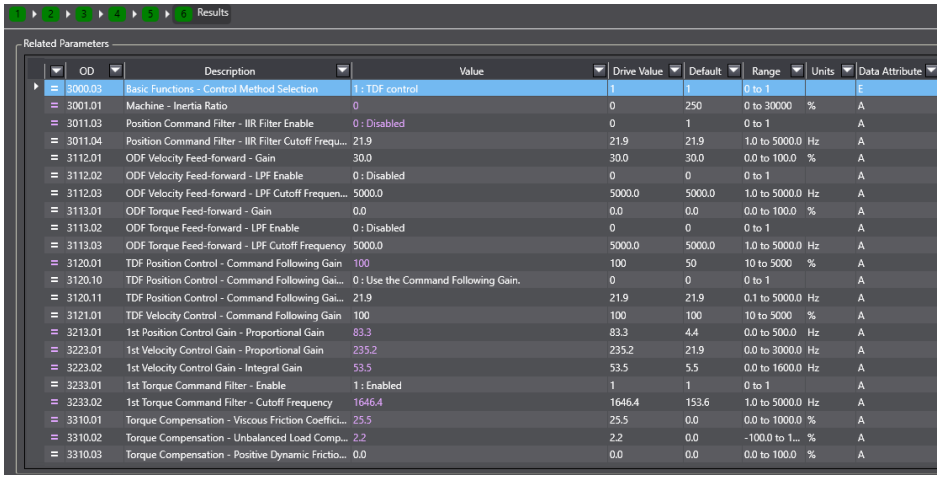
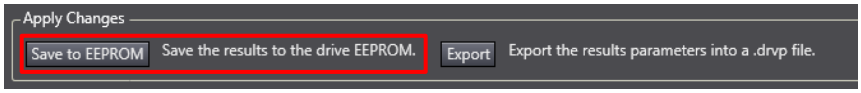


To manually make further adjustments...

Responsiveness may fluctuate when the combination of gains in the red area is set. Click the area in other colors than red where the combination of gains is as large as possible and then click the **Refresh Calculations** Button.



6.	Time Response Simulation (Wizard Step 4) In time response simulation, response to the motion profile can be simulated. Click the **Simulate Motion Profile** Button.  The chart is updated. If necessary, adjust gains.  If your application requires a small following error, here is an example of position following gain adjustment.      The following error has been reduced.
7.	When satisfied with the simulation results, transfer parameters to the Servo Drive and click the *Next* Button. 

8.	Check Behavior (Wizard Step 5) Click the *Execute* Button. (The Servomotor will run based on the motion profile set in Wizard Step 3 (estimation of load characteristics).)   (Motion profile set in Step 3 of the procedure) The traced data is displayed in charts.  When satisfied with the results, click the **Next** Button. 
9.	Results (Wizard Step 6) A parameter list showing the adjustment results is displayed.  Click the *Save to EEPROM* Button to save the results to the non-volatile memory.  Click the *Finish* Button. Advanced Auto-Tuning has been completed. 

Note: Do not use this document to operate the Unit.

OMRON Corporation **Industrial Automation Company**

Kyoto, JAPAN **Contact : www.ia.omron.com**

Regional Headquarters

OMRON EUROPE B.V.
Wegalaan 67-69, 2132 JD Hoofddorp
The Netherlands
Tel: (31) 2356-81-300 Fax: (31) 2356-81-388

OMRON ELECTRONICS LLC
2895 Greenspoint Parkway, Suite 200
Hoffman Estates, IL 60169 U.S.A.
Tel: (1) 847-843-7900 Fax: (1) 847-843-7787

OMRON ASIA PACIFIC PTE. LTD.
438B Alexandra Road, #08-01/02 Alexandra
Technopark, Singapore 119968
Tel: (65) 6835-3011 Fax: (65) 6835-3011

OMRON (CHINA) CO., LTD.
Room 2211, Bank of China Tower,
200 Yin Cheng Zhong Road,
PuDong New Area, Shanghai, 200120, China
Tel: (86) 21-6023-0333 Fax: (86) 21-5037-2388

Authorized Distributor:

Cat. No. I929-E1-01 0225 (0225)