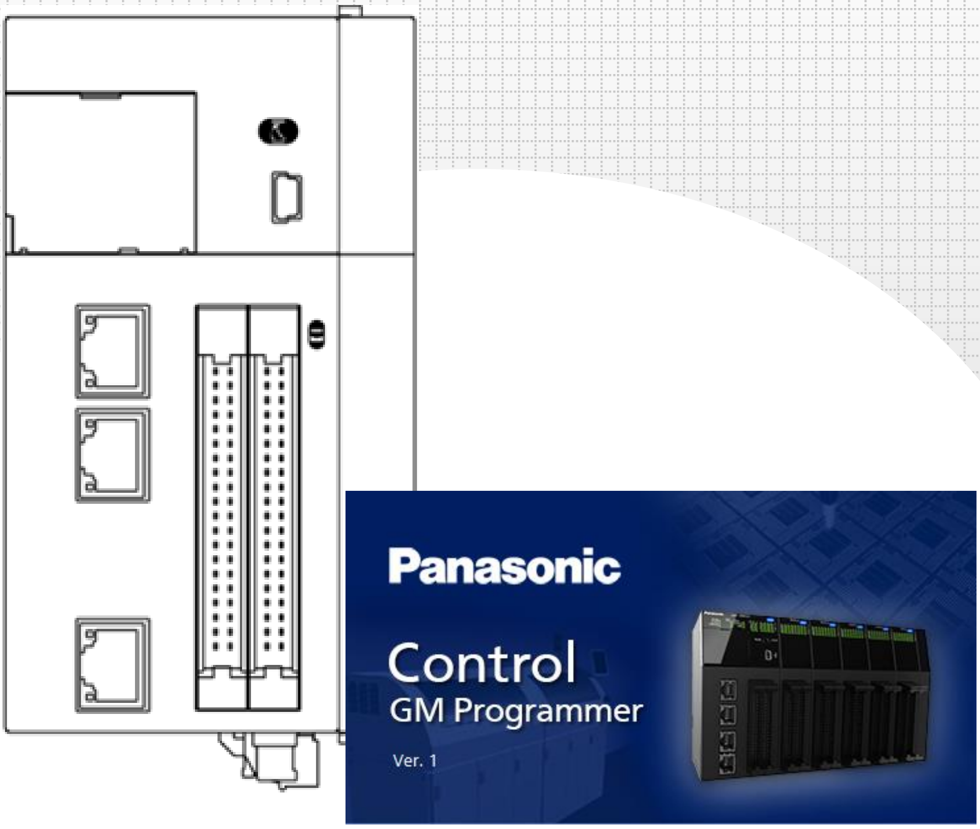
Panasonic®

Hello! GM1 Expansion Unit Edition



memo

Copyright and Trademarks

・The copyright of this manual is owned by Panasonic Industry Co., Ltd.
・Unauthorized reproduction of this manual is strictly prohibited.
・Windows is a registered trademark of Microsoft Corporation in the United States and other countries.
・Ethernet is a registered trademark of Fuji Xerox Co., Ltd. and Xerox Corporation in the United States.
・EtherCAT is a registered trademark of and patented technology licensed by Beckhoff Automation GmbH in Germany.
・SDHC and SD logos are trademarks of SD-3C, LLC.
・Other company and product names are trademarks or registered trademarks of their respective companies.

Safety Precautions

This section explains important rules that must be observed to prevent personal injury and property damage.
・Injuries and damages that may occur as a result of incorrect use are classified into the following levels and safety precautions are explained according to the level.

⚠ WARNING	Indicates that there is a risk of death or serious injury
⚠ CAUTION	Indicates that there is a risk of minor injury or property damage
⊘	Indicates an action that is prohibited
❗	Indicates an action that must be taken

⚠ WARNING

❗ ・Take safety measures outside this product to ensure the safety of the entire system even if this product fails or an error occurs due to external factors.
⊘ ・Do not use this product in atmospheres that contain flammable gases. Doing so may result in explosion.
⊘ ・Do not throw this product into the fire. Doing so may cause the batteries or other electronic parts to explode.

⚠ CAUTION

❗ ・To prevent abnormal heat generation or smoke generation, use this product with some leeway from the guaranteed characteristics and performance values of the product.
⊘ ・Do not disassemble or modify this product. Doing so may result in abnormal heat generation or smoke generation.
⊘ ・Do not touch any terminals while the power is on.
❗ ・Configure emergency stop and interlock circuits outside this product.
❗ ・Connect wires and connectors properly. Failure to do so may result in abnormal heat generation or smoke generation.
⊘ ・Do not perform work (such as connection or removal) with the power turned on.
❗ ・If this product is used in any way that is not specified by Panasonic, its protection function may be impaired.
❗ ・This product has been developed and manufactured for factory use only.

Contents of This Textbook and Scope of Responsibility

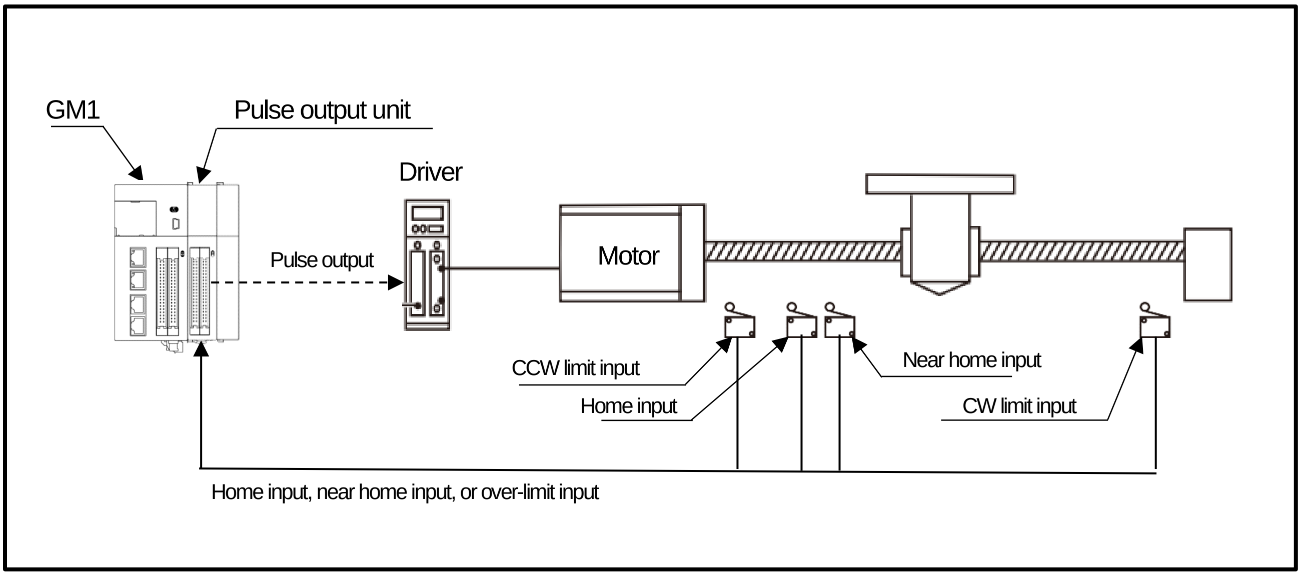
This textbook describes how to set up the GM1 Series Expansion Unit and operate GM Programmer, but does not describe safety precautions or notes on use for each device.
Be sure to obtain the instruction manuals and other manuals for the devices used in this textbook and read safety precautions and notes on use before using the devices.
Panasonic shall not take any responsibility for any damages that may arise in relation to our products or software, or this textbook.

GM1 Expansion Unit Edition (Pulse Output Unit)

GM1 Expansion Unit Edition (Analog I/O Unit)

Pulse Output Unit

Installation Overview



● Operation patterns

After home return is complete, Operation ① to Operation ③ are performed continuously.

Home return

The object moves to its home position (0).

Operation ①: Absolute value positioning

The object moves from its home position to target position ① (5000).

Operation ②: Relative value positioning

The object moves from target position ① (5000) to target position ② (15000).

Operation ③: Relative value positioning

The object moves from target position ② (15000) to target position ③ (12000).

INFO

The command unit used by the GM1 Pulse Output Unit is "u", and 1u is equivalent to 1 pulse.

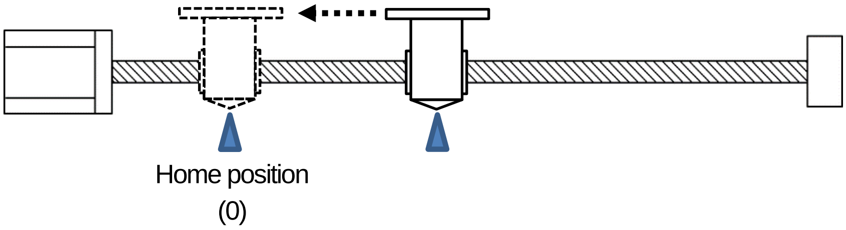
Conversion to the unit used in a mechanical system to be prepared by the customer must be added by executing processing in programs or implementing other measures as necessary.

Unless otherwise specified, units are omitted in this textbook.

- Operation images

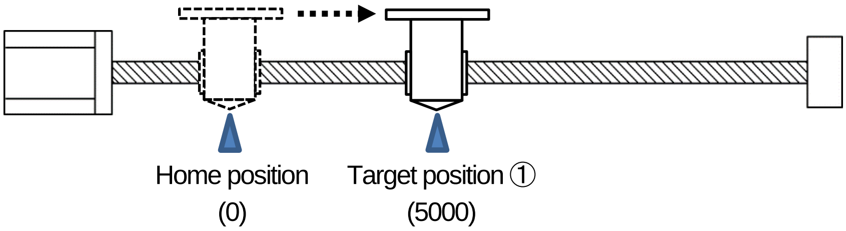
Home return

The object is moved to its home position (0).



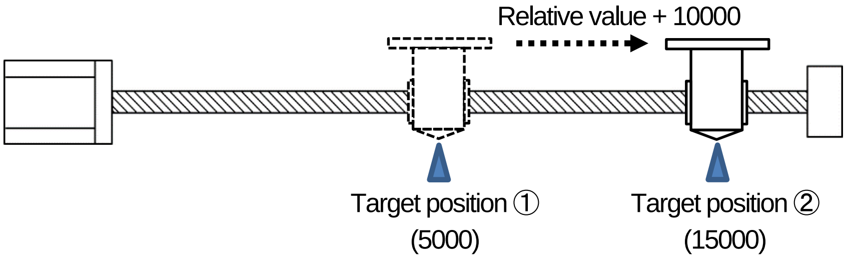
Operation ①: Absolute value positioning

The object is moved from its home position (0) to target position ① (5000).



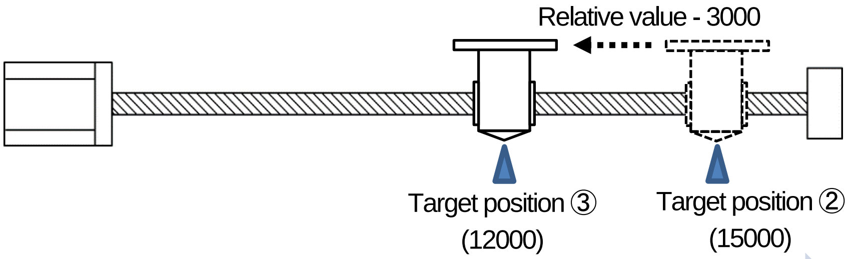
Operation ②: Relative value positioning

The object is moved from target position ① (5000) to target position ② (15000).

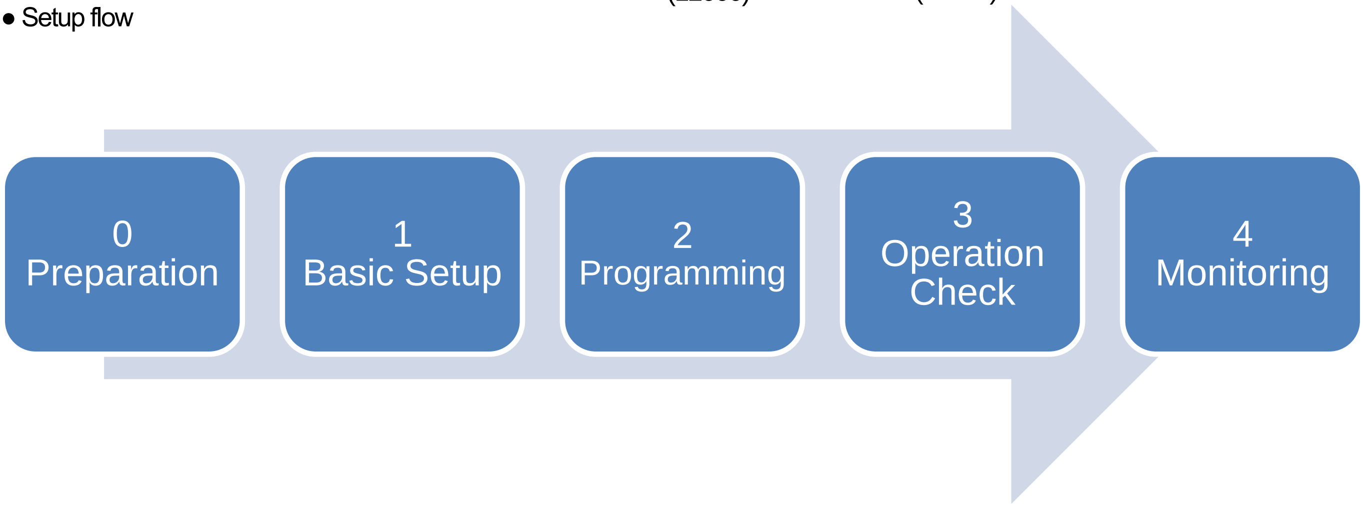


Operation ③: Relative value positioning

The object is moved from target position ② (15000) to target position ③ (12000).



- Setup flow



0 Preparation

1 Basic Setup

2 Programming

3 Operation Check

4 Monitoring

0 Preparation

Installing tool software

Install GM Programmer from the following website:
GM Programmer: https://industrial.panasonic.com/ac/j/motor/motion-controller/mc/gm1/index.jsp

INFO

When GM Programmer is installed, PANATERM Lite for GM, Gateway (CODESYS Gateway), and CodeMeter applications are installed at the same time.

- GM Programmer: This is a setup tool for the GM1 controller. Using GM Programmer makes it possible to set positioning data and various positioning parameters, and perform various monitoring operations.
- PANATERM Lite for GM1 (not used in this textbook): This is a setup support tool for the MINAS series servo amplifiers manufactured by Panasonic Corporation. When GM Programmer is installed, PANATERM Lite for GM is also installed at the same time. By using this tool, parameter setup within servo amplifiers, control status monitoring, setup support, machine analysis, and other operations can be executed on a PC.

Before installing GM Programmer on a PC, log on to the PC with Administrator privileges.
If other applications are running, be sure to close all the applications before installing GM Programmer.

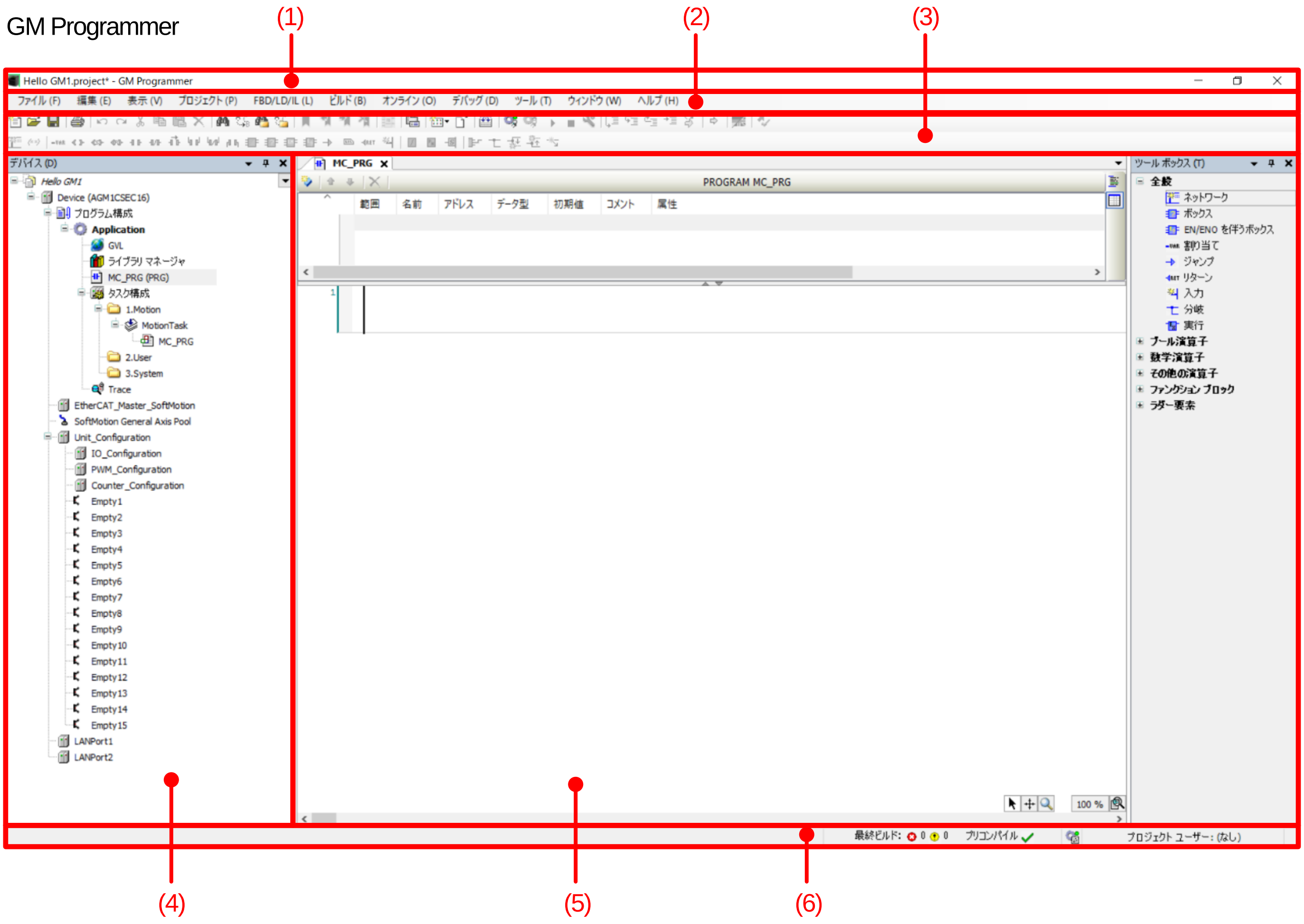
Column ①: Installing PANATERM

If a PC on which PANATERM is installed is connected to a MINAS servo amplifier with a USB cable, parameter setup, control status monitoring, setup support, machine analysis, and other operations can be performed easily.
If necessary, install PANATERM from the following Panasonic website.

- PANATERM: https://www3.panasonic.biz/ac/j/dl/software/index.jsp?series_cd=3514

Column ②: Components of each tool software product

GM Programmer



No.	Name	Description
(1)	Title bar	The title bar displays the project file name, minimize button, maximize button, and close button.
(2)	Menu bar	The menu bar displays the menu commands for each purpose in list format.
(3)	Toolbar	The toolbar displays each command as an icon.
(4)	Navigation pane	The navigation pane displays the objects (such as devices, applications, and programs) added to the project in a tree structure.
(5)	Main pane	The main pane displays a program, function settings, messages, and other data. The sub-pane can be switched by selecting a desired tab.
(6)	Status bar	The status bar displays the build status, logged-in users, and other information.

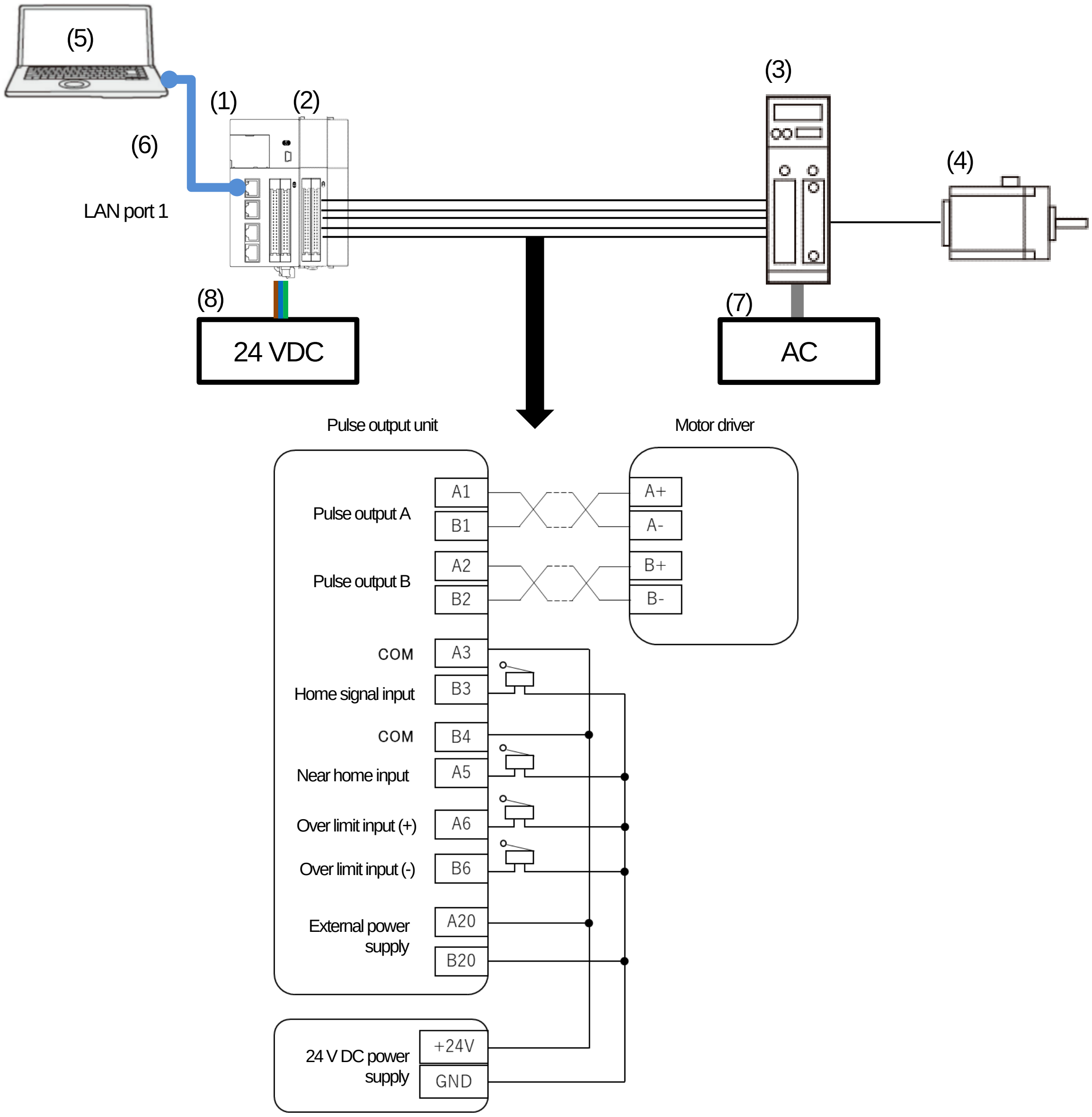
1 Basic Setup

1.1 Preparing and Wiring the Required Devices

Prepare the following devices.

No.	Name
(1)	GM1 controller (RTEX type)
(2)	GM1 pulse output unit (Line driver output type): AGM1PG04L
(3)	Motor driver
(4)	Stepping motor (Line driver input type)
(5)	PC (with GM Programmer installed)
(6)	LAN cable
(7)	AC power supply
(8)	24 V DC power supply

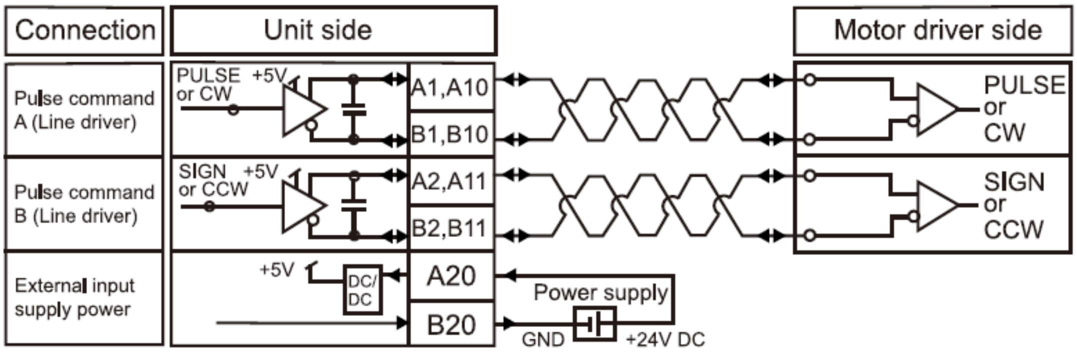
Wire each device as shown below.



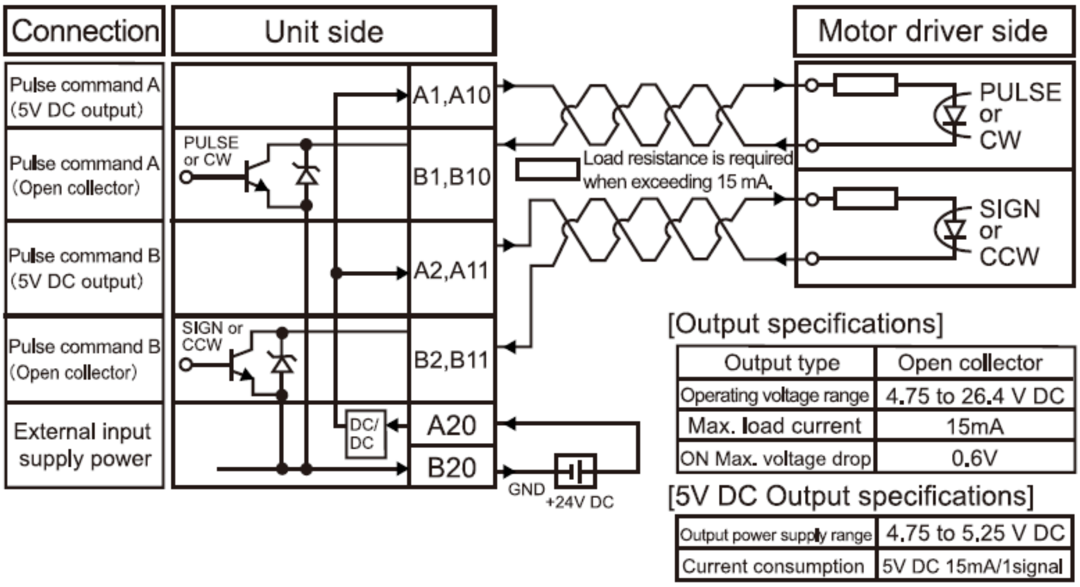
Column ③: Wiring pulse command outputs

Pulse output units are divided into two types: Line driver output type and transistor output type.
Select and connect an appropriate pulse output unit according to the interface of the motor driver to be used.

Line driver output type



Transistor output type


Connection
Unit side
Motor driver side
Pulse command A (5V DC output)
A1,A10
Pulse command A (Open collector)
PULSE or CW
B1,B10
PULSE or CW
Load resistance is required when exceeding 15 mA.
Pulse command B (5V DC output)
A2,A11
SIGN or CCW
Pulse command B (Open collector)
SIGN or CCW
B2,B11
External input supply power
DC/DC
A20
B20
GND
+24V DC

[Output specifications]

Output type	Open collector
Operating voltage range	4.75 to 26.4 V DC
Max. load current	15mA
ON Max. voltage drop	0.6V

[5V DC Output specifications]

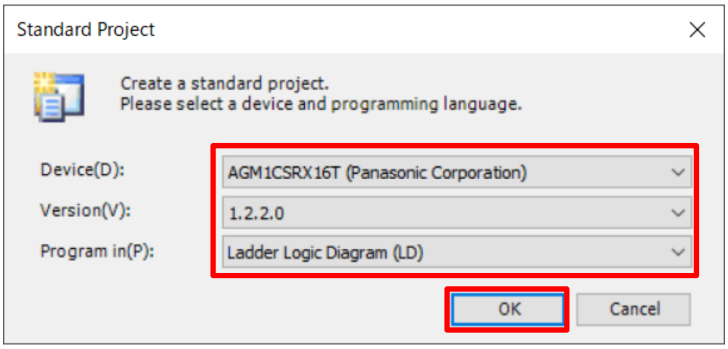
Output power supply range	4.75 to 5.25 V DC
Current consumption	5V DC 15mA/1signal

1.2 Registering an Expansion Unit

This section describes how to register an expansion unit.

Step 1

Start GM Programmer. Select **New Project** from the **File** menu, assign any name, and then click **OK**.
Device: AGM1CSEC16 (Panasonic Corporation)
Version: Any applicable version
Program in: Ladder Logic Diagram (LD)
Select the above values and click **OK**.



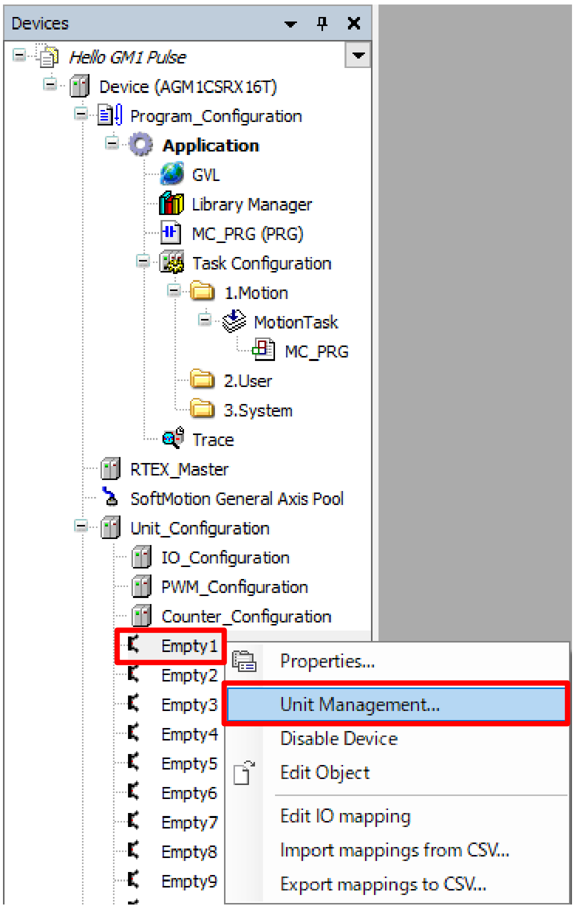
Standard Project

Create a standard project.
Please select a device and programming language.

Device(D): AGM1CSRX16T (Panasonic Corporation)
Version(V): 1.2.2.0
Program in(P): Ladder Logic Diagram (LD)

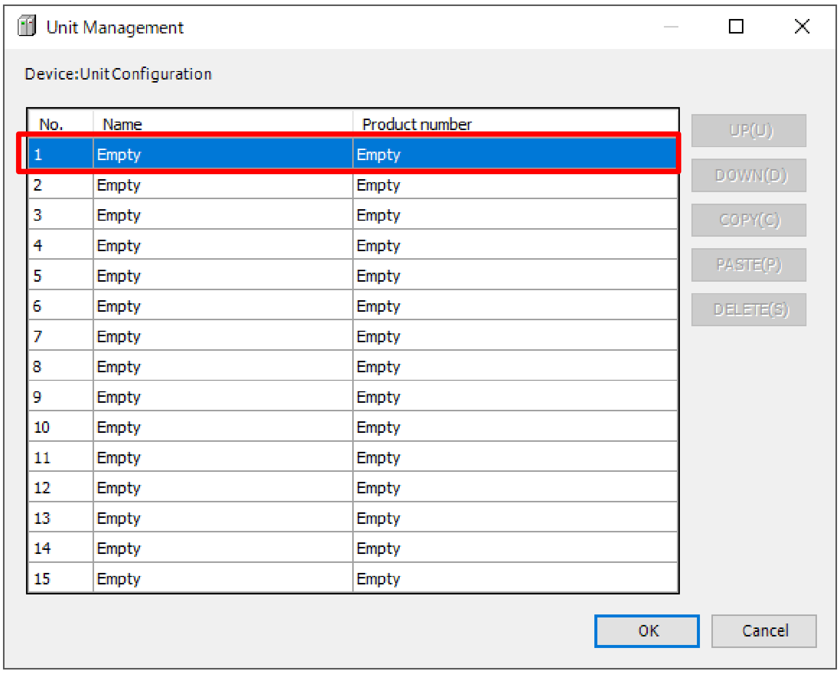
Step 2

Right-click **Empty1** and select **Unit Management**.



Step 3

The **Unit Management** pop-up window will be displayed. Register an expansion unit to be attached to the GM1 controller. Double-click on the first row.



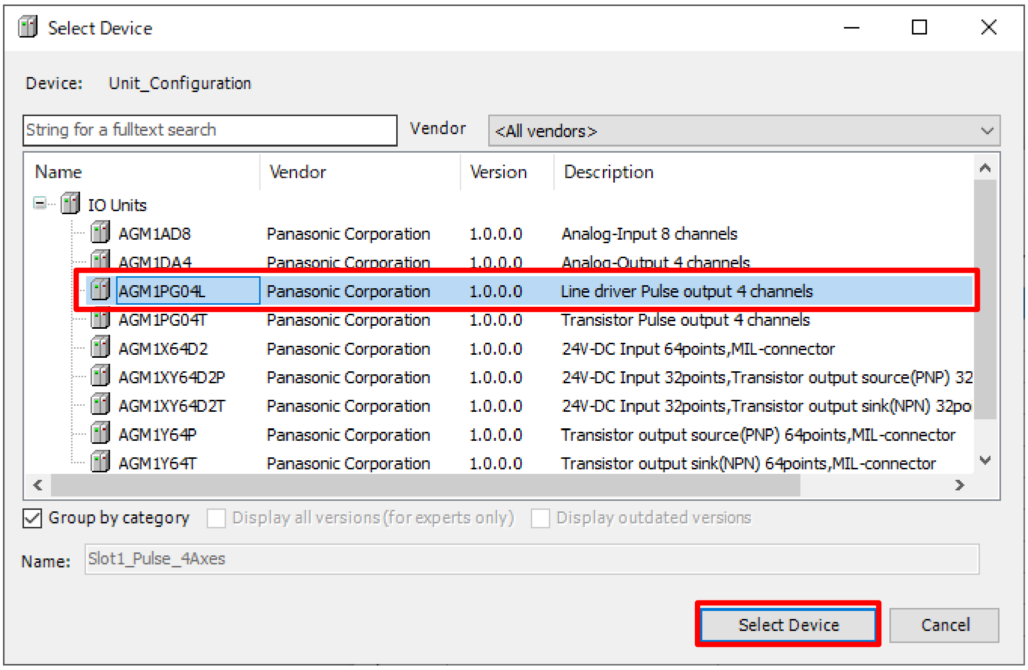
Unit Management

Device:Unit Configuration

No.	Name	Product number
1	Empty	Empty
2	Empty	Empty
3	Empty	Empty
4	Empty	Empty
5	Empty	Empty
6	Empty	Empty
7	Empty	Empty
8	Empty	Empty
9	Empty	Empty
10	Empty	Empty
11	Empty	Empty
12	Empty	Empty
13	Empty	Empty
14	Empty	Empty
15	Empty	Empty

Step 4

The **Select Device** pop-up window will be displayed. Select **AGM1PG04L** and click **Select Device**.



Select Device

Device: Unit_Configuration

String for a fulltext search

Vendor <All vendors>

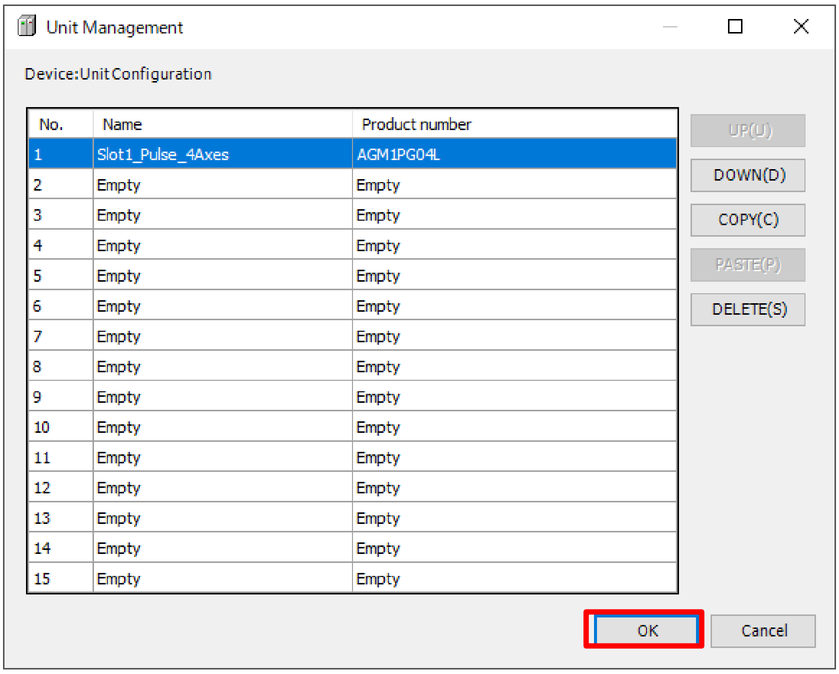
Name	Vendor	Version	Description
IO Units			
AGM1AD8	Panasonic Corporation	1.0.0.0	Analog-Input 8 channels
AGM1DA4	Panasonic Corporation	1.0.0.0	Analog-Output 4 channels
AGM1PG04L	Panasonic Corporation	1.0.0.0	Line driver Pulse output 4 channels
AGM1PG04T	Panasonic Corporation	1.0.0.0	Transistor Pulse output 4 channels
AGM1X64D2	Panasonic Corporation	1.0.0.0	24V-DC Input 64points,MIL-connector
AGM1XY64D2P	Panasonic Corporation	1.0.0.0	24V-DC Input 32points,Transistor output source(PNP) 32
AGM1XY64D2T	Panasonic Corporation	1.0.0.0	24V-DC Input 32points,Transistor output sink(NPN) 32po
AGM1Y64P	Panasonic Corporation	1.0.0.0	Transistor output source(PNP) 64points,MIL-connector
AGM1Y64T	Panasonic Corporation	1.0.0.0	Transistor output sink(NPN) 64points,MIL-connector

☑ Group by category ☐ Display all versions (for experts only) ☐ Display outdated versions

Name: Slot1_Pulse_4Axes

Step 5

Click **OK** to close the pop-up window.

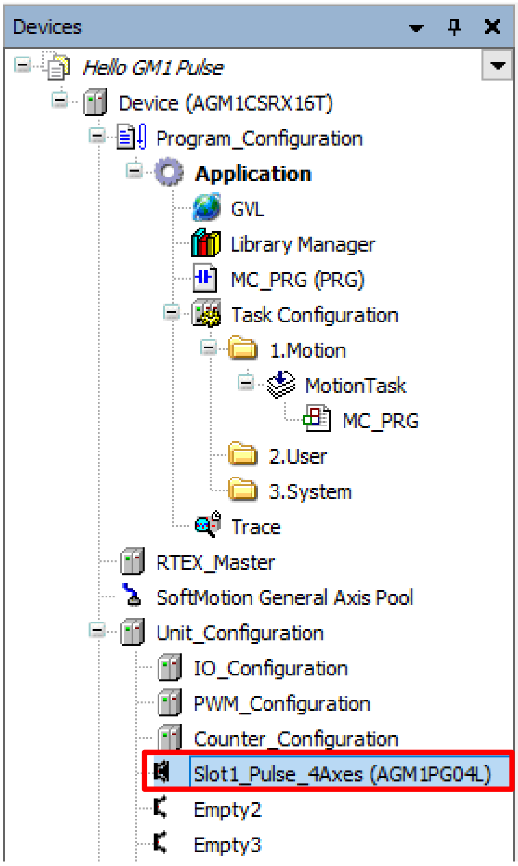


Unit Management

Device:Unit Configuration

No.	Name	Product number
1	Slot1_Pulse_4Axes	AGM1PG04L
2	Empty	Empty
3	Empty	Empty
4	Empty	Empty
5	Empty	Empty
6	Empty	Empty
7	Empty	Empty
8	Empty	Empty
9	Empty	Empty
10	Empty	Empty
11	Empty	Empty
12	Empty	Empty
13	Empty	Empty
14	Empty	Empty
15	Empty	Empty

Check that **Empty1** has been changed to **slot1_Pulse_4Axis(AGM1PG01L)** as shown below.



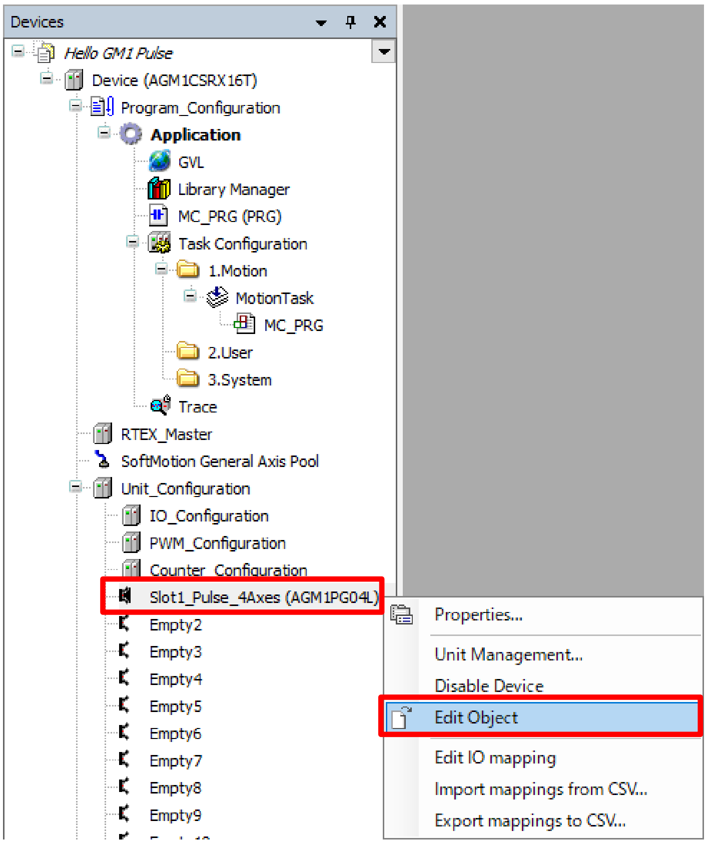
```
Devices
Hello GM1 Pulse
  Device (AGM1CSRX16T)
    Program_Configuration
      Application
        GVL
        Library Manager
        MC_PRG (PRG)
        Task Configuration
          1.Motion
            MotionTask
              MC_PRG
          2.User
          3.System
        Trace
    RTEX_Master
    SoftMotion General Axis Pool
    Unit_Configuration
      IO_Configuration
      PWM_Configuration
      Counter_Configuration
      Slot1_Pulse_4Axes (AGM1PG04L)
      Empty2
      Empty3
```

1.3 Setting up a Pulse Output Unit

This section explains the parameter setting procedure and parameter settings for the pulse output unit.

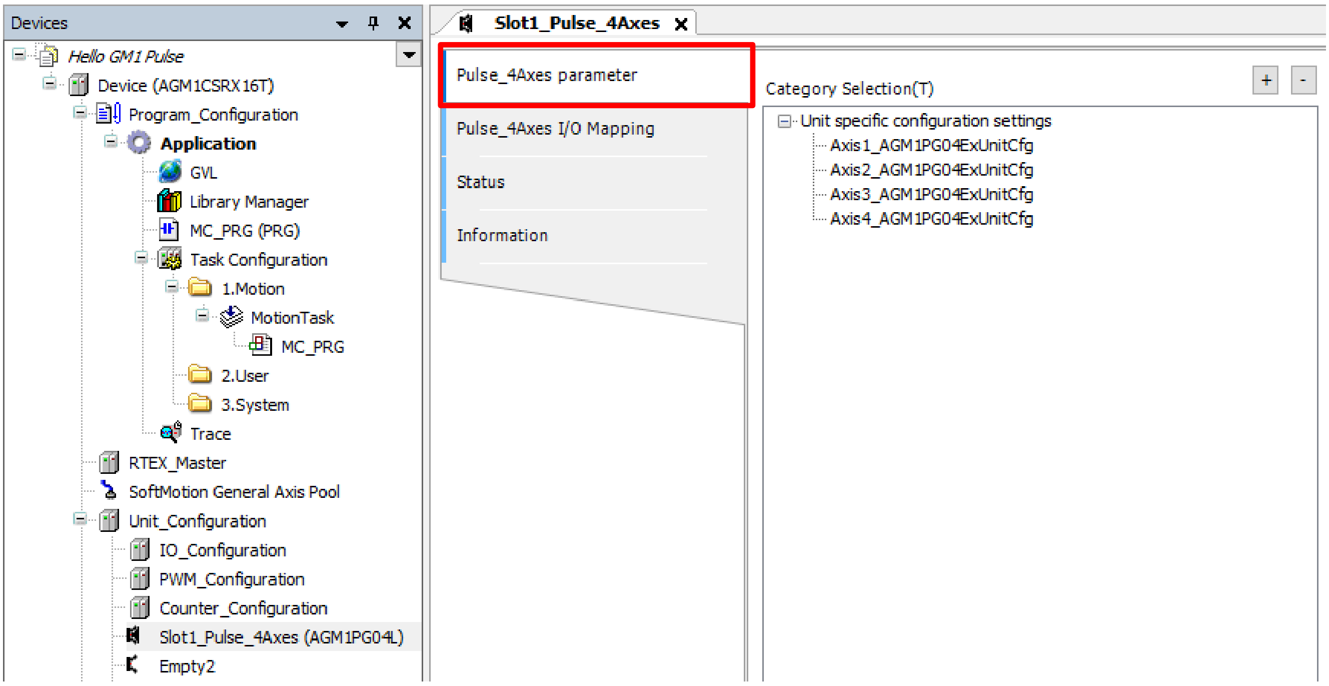
Step 1

Right-click **Slot1_Pulse_4Axes(AGM1PG04L)**, which is the expansion unit that has been registered previously, and select **Edit Object**.



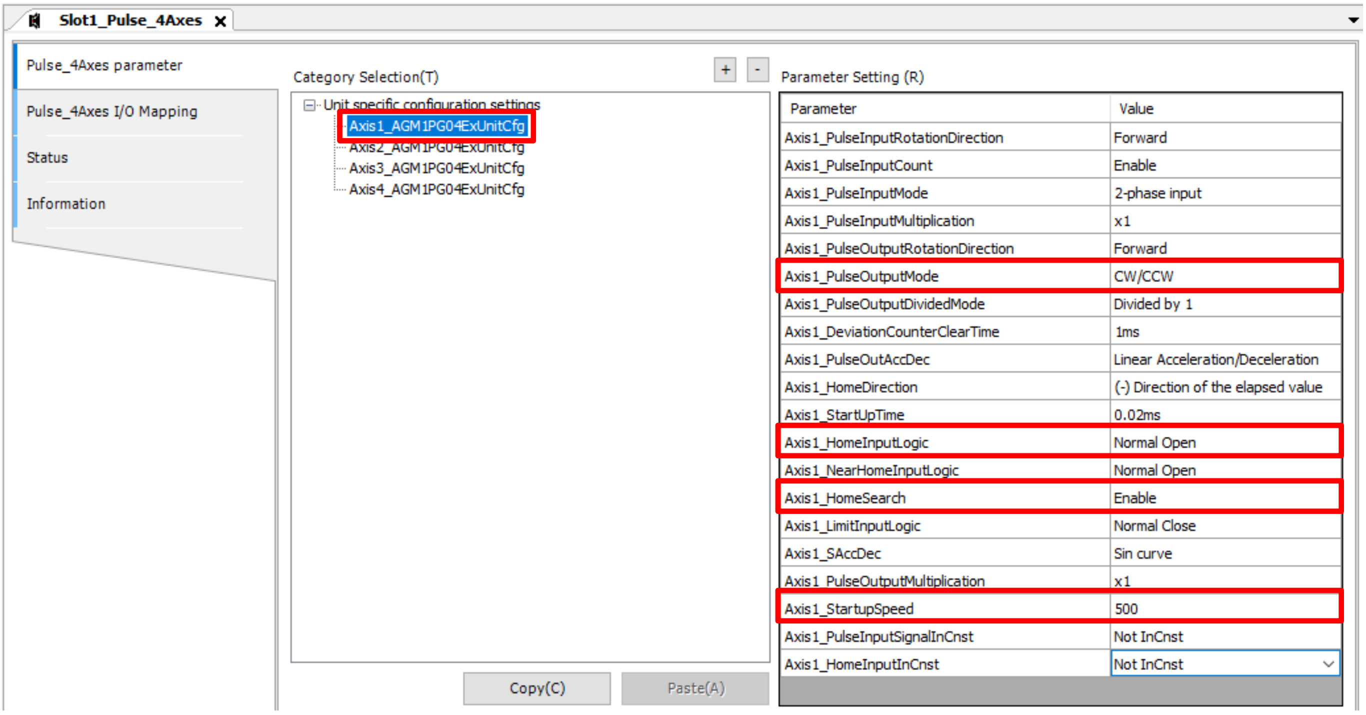
Step 2

A setting window will open as the main pane. Select the **Pluse_4Axes parameter** tab.



Step 3

This time, we will use Axis 1 of the pulse output unit, so click **Axis1_AGM1PG04ExUnitCfg** and set up parameters.



Parameter	Value
Axis1_PulseInputRotationDirection	Forward
Axis1_PulseInputCount	Enable
Axis1_PulseInputMode	2-phase input
Axis1_PulseInputMultiplication	x1
Axis1_PulseOutputRotationDirection	Forward
Axis1_PulseOutputMode	CW/CCW
Axis1_PulseOutputDividedMode	Divided by 1
Axis1_DeviationCounterClearTime	1ms
Axis1_PulseOutAccDec	Linear Acceleration/Deceleration
Axis1_HomeDirection	(-) Direction of the elapsed value
Axis1_StartUpTime	0.02ms
Axis1_HomeInputLogic	Normal Open
Axis1_NearHomeInputLogic	Normal Open
Axis1_HomeSearch	Enable
Axis1_LimitInputLogic	Normal Close
Axis1_SAccDec	Sin curve
Axis1_PulseOutputMultiplication	x1
Axis1_StartupSpeed	500
Axis1_PulseInputSignalInCnst	Not InCnst
Axis1_HomeInputInCnst	Not InCnst

This time, set up parameters as below.

Axis1_PulseOutputMode	CW/CCW
Axis1_HomeInputLogic	Normal Open
Axis1_HomeSearch	Enable
Axis1_StartupSpeed	500

INFO

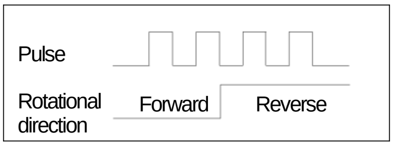
If there is no limit input sensor, set **Axis1_ LimitInputLogic** to **Normal Open**. If **Axis1_ LimitInputLogic** is set to **Normal Close**, a limit error will occur.

Set **Axis1_HomeSearch** to **Disable**.

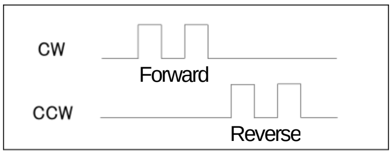
Column ④: Pulse output mode (Pulse/Sign and CW/CCW)

The pulse input mode for the motor driver is divided into two types: Pulse/Sign mode and CW/CCW mode. Therefore, an appropriate pulse output mode must be selected according to the driver specification.

- Pulse/Sign mode
  In this mode, the rotational direction of the motor is identified according to the high or low level of the rotational direction signal when a pulse is input into the pulse input terminal of the motor driver.



- CW/CCW mode
  In this mode, the rotational direction of the motor is identified according to the pulse input terminal of the motor driver (CW input or CCW input) into which a pulse is input.



Column ⑤: Home search valid mode and home search invalid mode

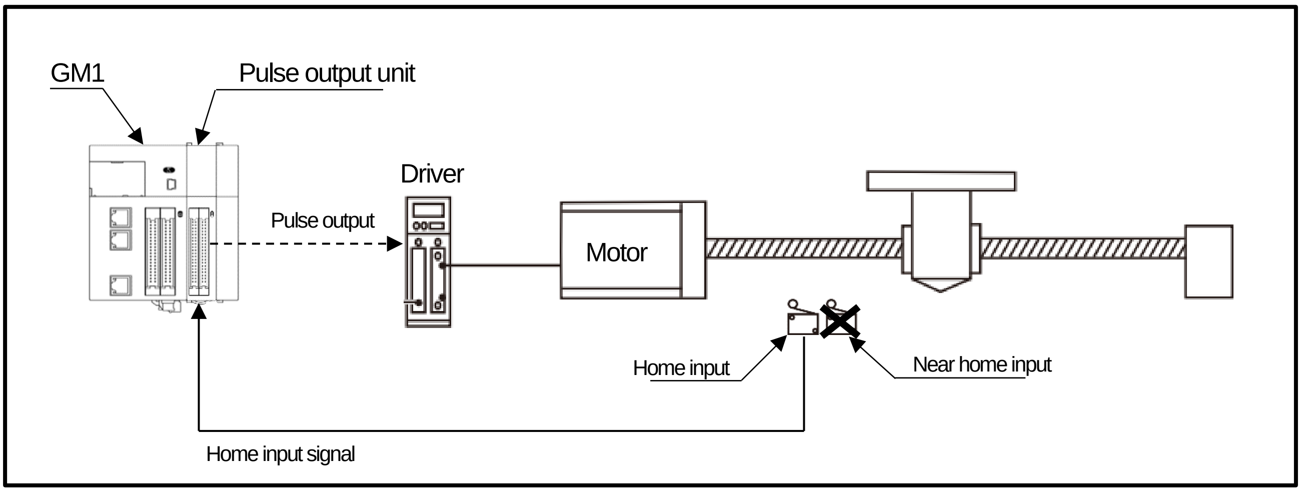
Home search valid mode

- If there is a near home signal when the object is moving in the home return direction, the object will slow down near the home position and stop at the home input position.
- If there is no home input signal when the object is moving in the home return direction, the object will reverse automatically when a limit input is detected. If the near home signal turns ON, the object will reverse again and then stop when the home input signal turns ON.
- If the near home signal is input during acceleration, the object will reverse automatically and then return to the home position after the change of the near home signal state from ON to OFF is detected.

Home search invalid mode

- If there is a near home signal when the object is moving in the home return direction, the object will slow down near the home position and stop at the home input position.
- If there is no home input signal when the object is moving in the home return direction, the object will stop when a limit input is detected.

Column ⑥: Home return parameters using only home input signals



- Axis1_HomeInputLogic: Normal Open
- Axis1_NearHomeInputLogic: Normal Close
- Axis1_HomeSearch: Disable

The above settings cause the near home input to be already enabled after home return operation is started.
Then, the home return operation stops when the home input signal turns ON.
These parameters can be used only in the home search invalid mode.

2 Programming

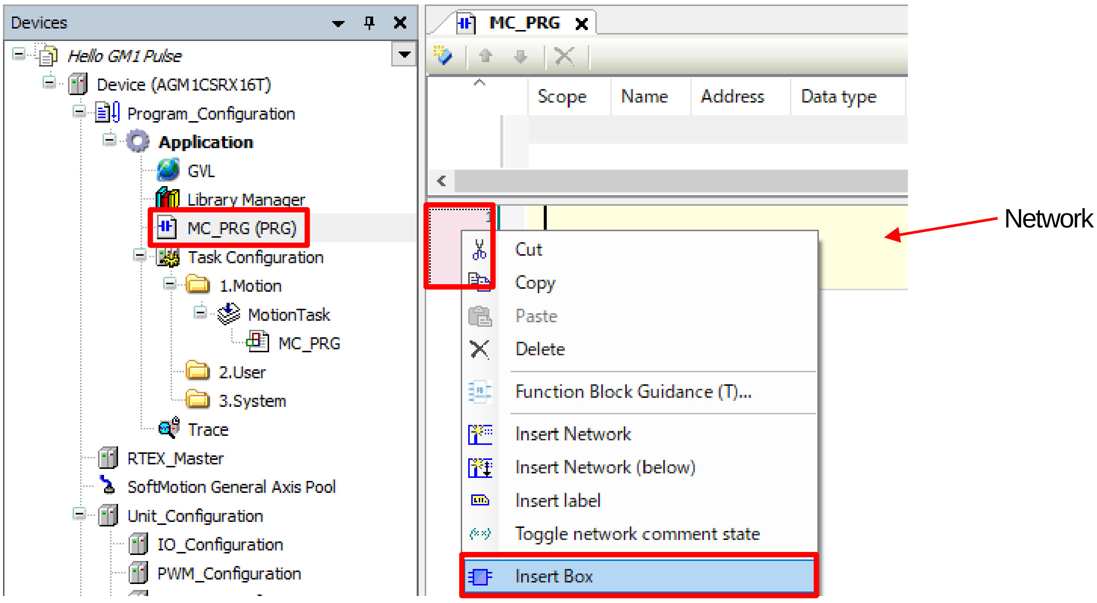
2.1 Creating Servo ON Program

Step 1

Code a program within MC_PRG.

Double-click **MC_PRG** to open the program pane.

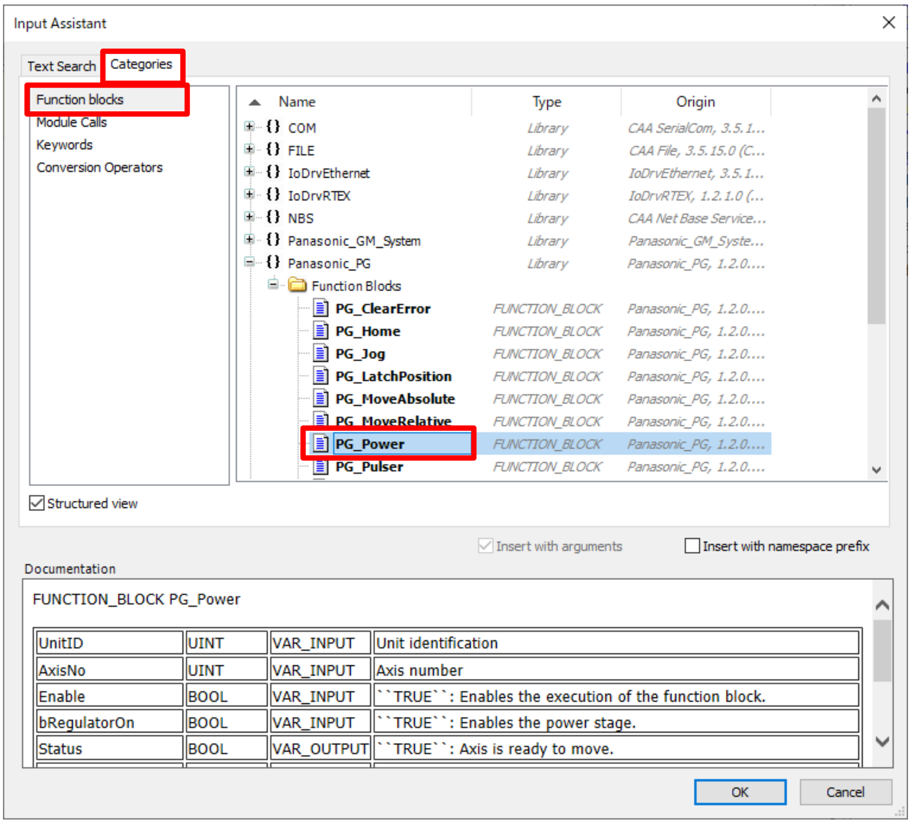
Right-click the leftmost section of the network (the red section in the figure below) and select **Insert Box**.



Step 2

The **Input Assistant** dialog box will open. In the **Categories** tab, select **Function blocks**.

In the right pane, select **{ } Panasonic_PG**, **Function Blocks**, and then **PG_Power** and click **OK**.



Name	Type	Origin
{} COM	Library	CAA SerialCom, 3.5.1...
{} FILE	Library	CAA File, 3.5.15.0 (C...
{} IoDrvEthernet	Library	IoDrvEthernet, 3.5.1...
{} IoDrvRTEX	Library	IoDrvRTEX, 1.2.1.0 (...
{} NBS	Library	CAA Net Base Service...
{} Panasonic_GM_System	Library	Panasonic_GM_Syste...
{} Panasonic_PG	Library	Panasonic_PG, 1.2.0....
Function Blocks		
PG_ClearError	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Home	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Jog	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_LatchPosition	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveAbsolute	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveRelative	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Power	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Pulser	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....

FUNCTION_BLOCK PG_Power

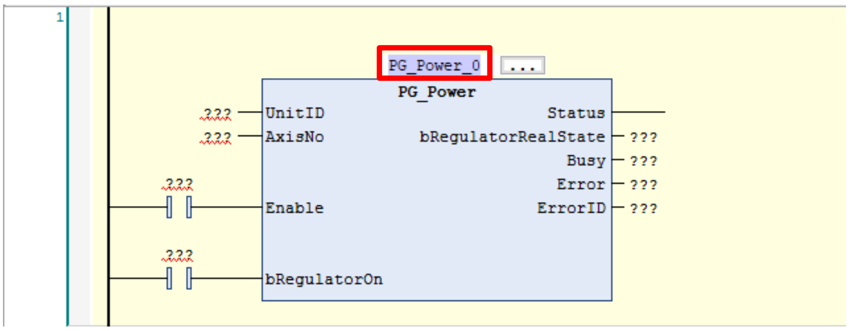
UnitID	UINT	VAR_INPUT	Unit identification
AxisNo	UINT	VAR_INPUT	Axis number
Enable	BOOL	VAR_INPUT	``TRUE``: Enables the execution of the function block.
bRegulatorOn	BOOL	VAR_INPUT	``TRUE``: Enables the power stage.
Status	BOOL	VAR_OUTPUT	``TRUE``: Axis is ready to move.

Column ⑦: List of function blocks for pulse output unit

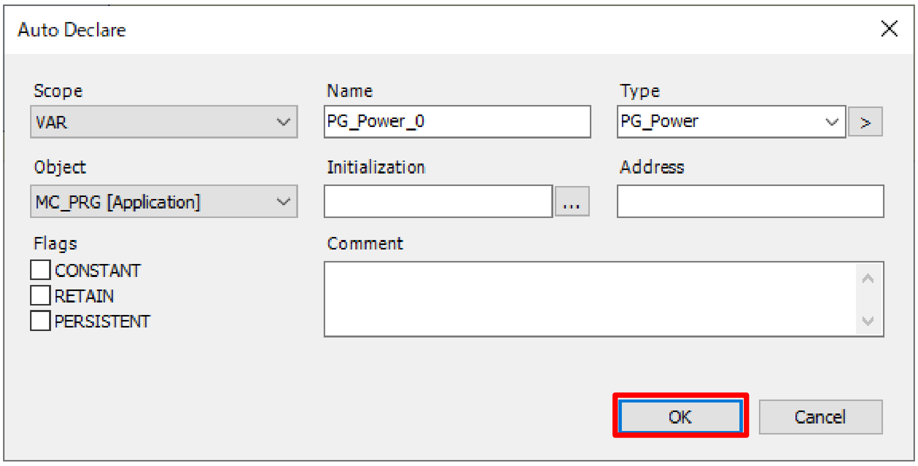
Function block	Outline
PG_Power	This is a function block (FB) that performs servo ON/OFF control on the axis of the pulse output unit.
PG_Jog	This is a function block (FB) that causes the axis of the pulse output unit to keep traveling in a forward or backward direction.
PG_MoveAbsolute	This is a function block (FB) that causes the axis of the pulse output unit to travel to a position specified as an absolute position.
PG_Move_Relative	This is a function block (FB) that causes the axis of the pulse output unit to travel to a position specified as a relative position.
PG_LatchPosition	This is a function block (FB) that causes the axis of the pulse output unit to travel to a position specified by an external signal input as a relative position.
PG_Pulser	This is a function block (FB) that enables constant speed operation for the axis of the pulse output unit using external pulse input.
PG_Stop	This is a function block (FB) that causes the axis of the pulse output unit to make a forced stop or deceleration stop.
PG_Home	This is a function block (FB) that performs home return of the pulse output unit.
PG_SetPosition	This is a function block (FB) that sets the elapsed value and the feedback counter of the pulse output unit to desired values.
PG_WriteParameter	This is a function block (FB) that writes the parameters to the pulse output unit.
PG_ReadParameter	This is a function block (FB) that reads the parameters of the pulse output unit.
PG_ClearError	This is a function block (FB) that clears the limit error or the set value error of the pulse output unit.
PG_ReadStatus	This is a function block (FB) that reads the status of the pulse output unit.

Step 3

Make sure that the cursor stays on **PG_Power_0** above the **PG_Power** function block that has been added, and then press the Enter key.



Variables will be automatically declared for PG_Power and the **Auto Declare** dialog box will be displayed. Check the contents of the dialog box and then click **OK**.


Auto Declare

Scope: VAR

Name: PG_Power_0

Type: PG_Power

Object: MC_PRG [Application]

Initialization:

Address:

Flags

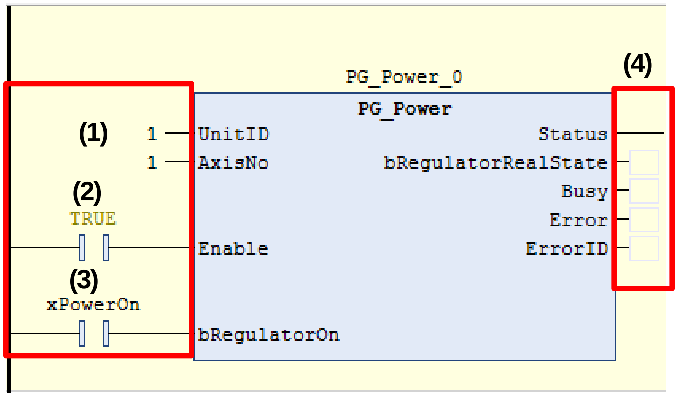
☐ CONSTANT

☐ RETAIN

☐ PERSISTENT

Comment:

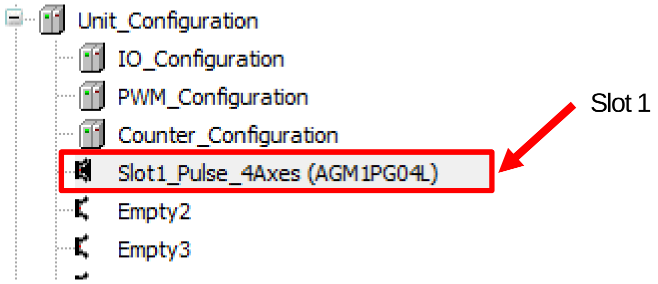
Complete the function block as show below.



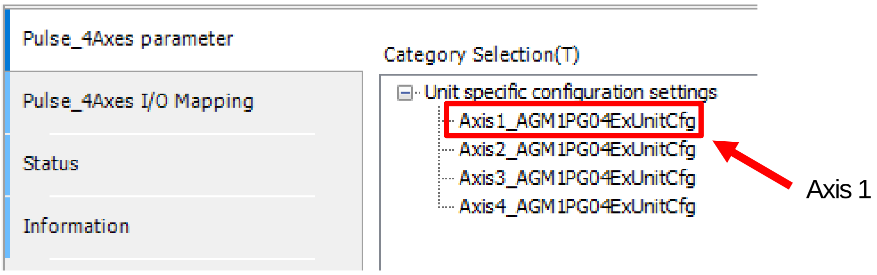
	Type	Parameter name	Set value	Description
(1)	Input	UnitID	1	Specifies the unit ID.
		AxisNo	1	Specifies the axis No.
(2)		Enable	TRUE	TRUE: FB ready for execution
(3)		bRegulatorOn	xPowerON	TRUE: Servo lock FALSE: Servo lock released
(4)	Output	Status	Delete ???	TRUE: Axis ready for operation
		bRegulatorRealState	Delete ???	TRUE: FB ready for execution
		Busy	Delete ???	TRUE: Execution of the FB is incomplete
		Error	Delete ???	TRUE: Error has occurred within FB
		ErrorID	Delete ???	An error ID is output

INFO

For the unit ID, specify the number of the slot in which the pulse output unit is mounted.



For Axis No., specify an axis number of the pulse output unit.



Column ⑧: Variables

Space and special characters (such as !, ", $, %, @, or &) cannot be used and a numeric character cannot be used as the first character.
Underscores cannot be used consecutively.

･Variable naming examples (naming conventions used in this textbook)

Variable names are assigned as English letters.
Each variable name starts with a prefix that indicates the type and scope of the variable.

A prefix is followed by a meaningful word name.
Only the first character of each word name is capitalized (to make it easy to understand the border between the prefix and the word).
Global variables must start with "g_".

Examples) BOOL type: xServoON
REAL type: rVelocity

BOOL type (Global variable): g_xPowerON
REAL type (Global variable): g_rPosition

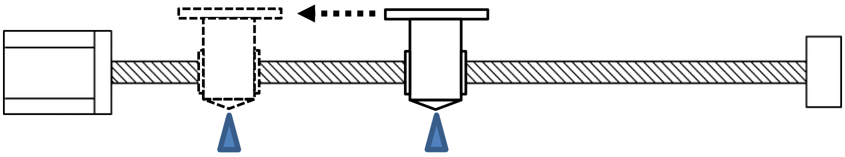
List of data types

Data type	Prefix	Remarks
BOOL	X or b	
BYTE	by	Not used for arithmetic operations
WORD	w	Not used for arithmetic operations
DWORD	dw	Not used for arithmetic operations
LWORD	lw	Not used for arithmetic operations
SINT	si	
USINT	usi	
INT	i	
UINT	ui	
DINT	di	
UDINT	udi	
LINT	li	
ULINT	uli	
REAL	r	
LREAL	lr	
STRING	s	
WSTRING	ws	
TIME	tim	
LTIME	ltim	
TIME_OF_DAY	tod	
DATE_AND_TIME	dt	
DATE	date	
POINTER	p	
ARRAY	a	
ENUM	e	
SCOPE		A prefix to identify the scope of a variable
VAR_GLOBAL VAR CONSTANT VAR_GLOBAL CONSTANT	g_ c_ gc_	For global variables, a concrete example is g_uiTest; For local constants, a concrete example is c_uiTest:INT; For global constants, a concrete example is gc_uiTest:INT;

2.2 Creating Home Return Program

Insert PG_Home to execute home return.

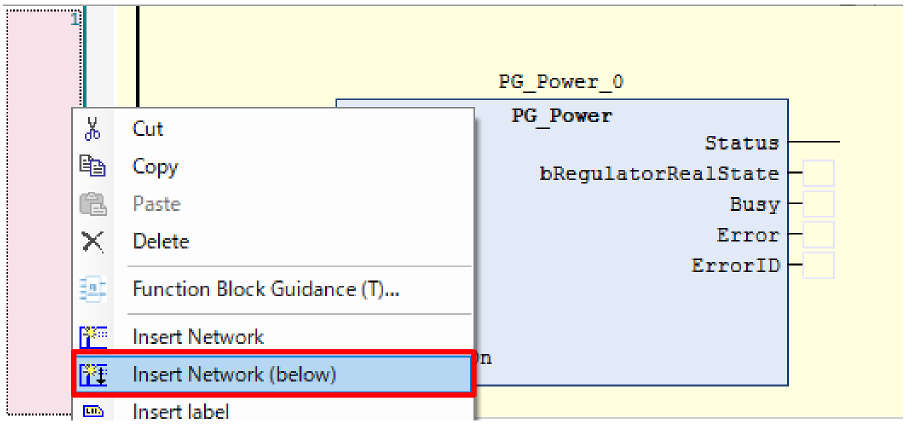
Operation 1) The object is moved to its home position (0 mm). (Home return)



Home position
(0)

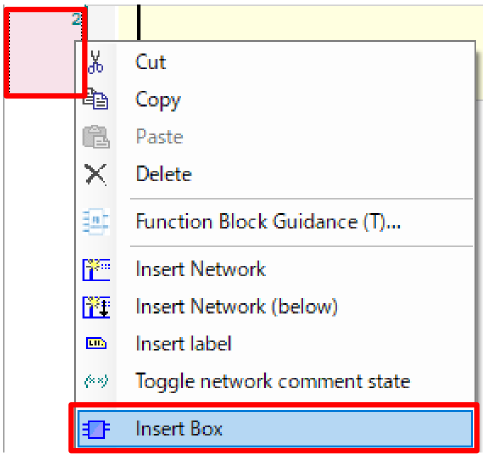
Step 1

Insert a new network. Right-click in the network and select **Insert Network (below)**.



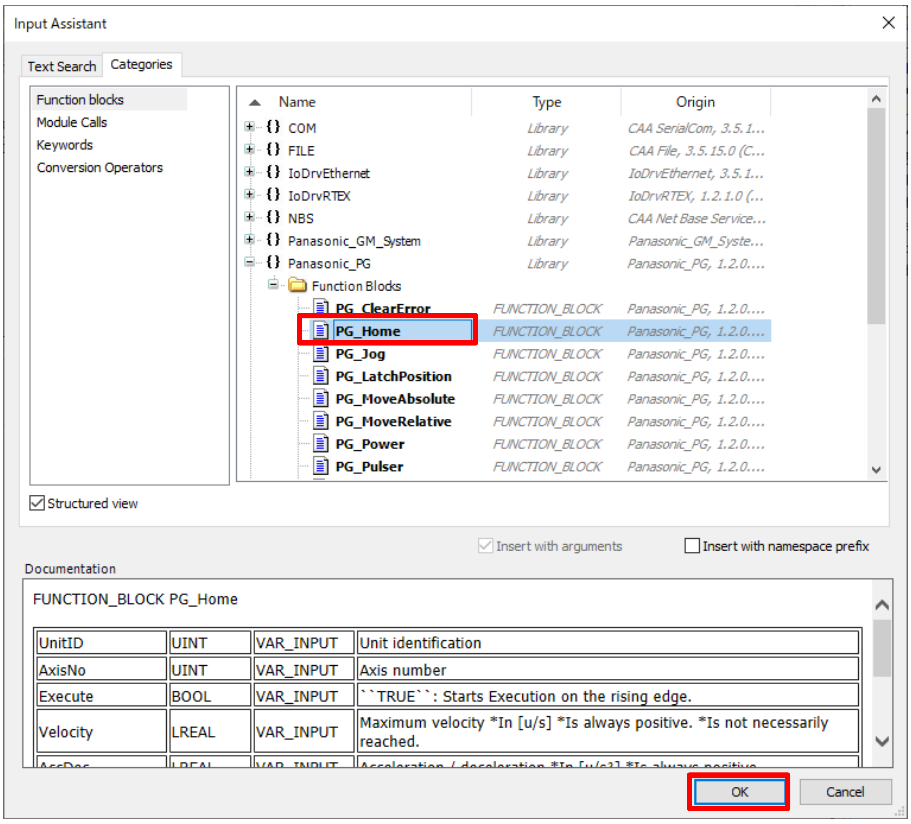
Step 2

Right-click the leftmost section of the new network and select **Insert Box**.



Step 3

The **Input Assistant** dialog box will open. In the **Categories** tab, select **Function blocks**.
In the right pane, select **{ } Panasonic_PG**, **Function Blocks**, and then **PG_Home** and click **OK**.



Input Assistant

Text Search | Categories

- Function blocks
- Module Calls
- Keywords
- Conversion Operators

Name	Type	Origin
{} COM	Library	CAA SerialCom, 3.5.1....
{} FILE	Library	CAA File, 3.5.15.0 (C...
{} IoDrvEthernet	Library	IoDrvEthernet, 3.5.1...
{} IoDrvRTEX	Library	IoDrvRTEX, 1.2.1.0 (...
{} NBS	Library	CAA Net Base Service...
{} Panasonic_GM_System	Library	Panasonic_GM_Syste...
{} Panasonic_PG	Library	Panasonic_PG, 1.2.0....
Function Blocks		
PG_ClearError	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Home	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Jog	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_LatchPosition	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveAbsolute	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveRelative	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Power	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Pulser	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....

☑ Structured view

☑ Insert with arguments ☐ Insert with namespace prefix

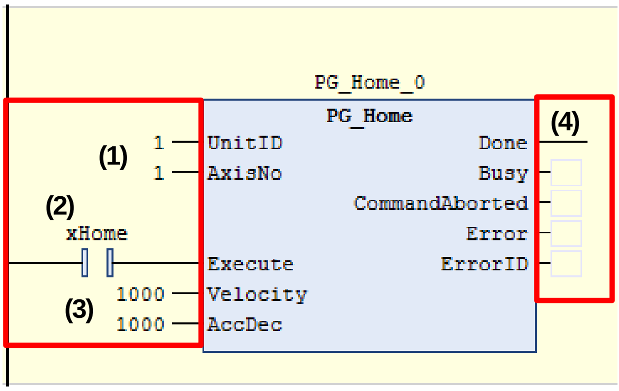
Documentation

FUNCTION_BLOCK PG_Home

UnitID	UINT	VAR_INPUT	Unit identification
AxisNo	UINT	VAR_INPUT	Axis number
Execute	BOOL	VAR_INPUT	``TRUE``: Starts Execution on the rising edge.
Velocity	LREAL	VAR_INPUT	Maximum velocity *In [u/s] *Is always positive. *Is not necessarily reached.
AccDec	LREAL	VAR_INPUT	Acceleration / deceleration *In [u/s2] *Is always positive

Step 4

Complete the function block as show below.



	Type	Parameter name	Set value	Description
(1)	Input	UnitID	1	Specifies the unit ID.
		AxisNo	1	Specifies the axis No.
(2)		Execute	xHome	Starts execution at the rising edge.
(3)		Velocity	1000	Specifies the maximum velocity (u/s).
		AccDec	1000	Specifies the acceleration/deceleration (u/s2).
(4)	Output	Done	Delete ???	TRUE: FB operation is completed
		Busy	Delete ???	TRUE: FB operation in progress
		CommandAborted	Delete ???	TRUE: FB operation is interrupted
		Error	Delete ???	TRUE: FB is abnormally completed
		ErrorID	Delete ???	An error ID is output

2.3 Creating JOG Operation Program

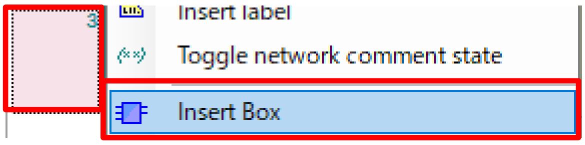
Create a program for JOG operation.

Step 1

Insert a new network using the same procedure as in "2.2 Creating Home Return Program".
Right-click in the network and select **Insert Network (below)**.

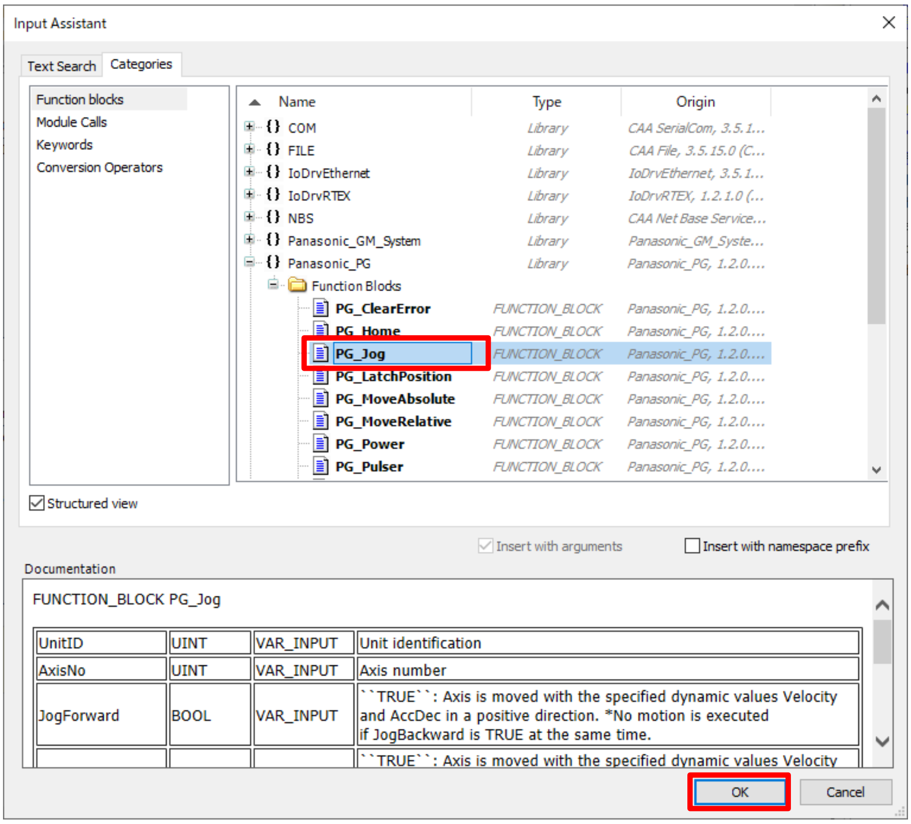
Step 2

Right-click the leftmost section of the new network and select **Insert Box**.



Step 3

The **Input Assistant** dialog box will open. In the **Categories** tab, select **Function blocks**.
In the right pane, select **{ } Panasonic_PG**, **Function Blocks**, and then **PG_Jog** and click **OK**.



Name	Type	Origin
{} COM	Library	CAA SerialCom, 3.5.1...
{} FILE	Library	CAA File, 3.5.15.0 (C...
{} IoDrvEthernet	Library	IoDrvEthernet, 3.5.1...
{} IoDrvRTEX	Library	IoDrvRTEX, 1.2.1.0 (...
{} NBS	Library	CAA Net Base Service...
{} Panasonic_GM_System	Library	Panasonic_GM_Syste...
{} Panasonic_PG	Library	Panasonic_PG, 1.2.0....
Function Blocks		
PG_ClearError	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Home	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Jog	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_LatchPosition	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveAbsolute	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveRelative	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Power	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Pulser	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....

☑ Structured view

☑ Insert with arguments □ Insert with namespace prefix

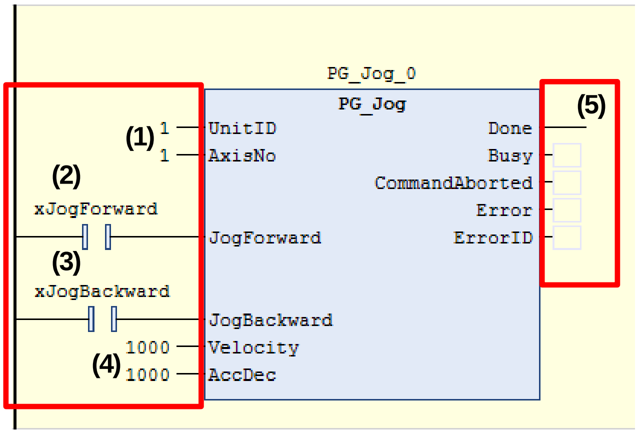
Documentation

FUNCTION_BLOCK PG_Jog

UnitID	UINT	VAR_INPUT	Unit identification
AxisNo	UINT	VAR_INPUT	Axis number
JogForward	BOOL	VAR_INPUT	``TRUE``: Axis is moved with the specified dynamic values Velocity and AccDec in a positive direction. *No motion is executed if JogBackward is TRUE at the same time.
			``TRUE``: Axis is moved with the specified dynamic values Velocity

Step 4

Complete the function block as show below.



	Type	Parameter name	Set value	Description
(1)	Input	UnitID	1	Specifies the unit ID.
		AxisNo	1	Specifies the axis No.
(2)		JogForward	xJogForward	TRUE: Travels in a forward direction
(3)		JogBackWard	xJogBackward	TRUE: Travels in a backward direction
(4)		Velocity	1000	Specifies the target speed (u/s).
		AccDec	1000	Specifies the acceleration/deceleration (u/s2).
(5)	Output	Done	Delete ???	TRUE: FB operation is completed
		Busy	Delete ???	TRUE: FB operation in progress
		CommandAborted	Delete ???	TRUE: FB operation is interrupted
		Error	Delete ???	TRUE: FB is abnormally completed
		ErrorID	Delete ???	An error ID is output

INFO

Set either JogForward or JogBackward to TRUE to execute the function block.
Depending on the direction in which the axis is to be moved, set either JogForward or JogBackward to TRUE.
If JogForward and JogBackward are simultaneously executed, a PG_JOG_INVALID_REQUEST error will occur.
To execute the function block again after the occurrence of the error, set both JogForward and JogBackward to FALSE once beforehand.

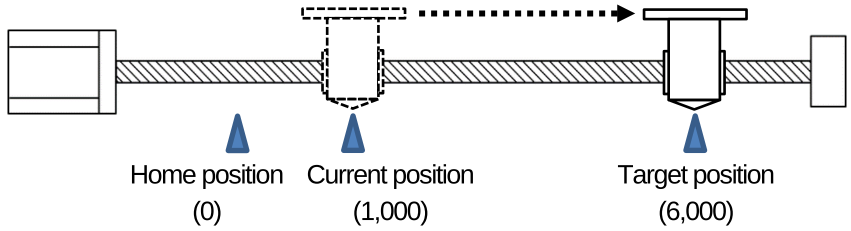
Column ⑨: Difference between absolute value positioning and relative value positioning

･Absolute value positioning

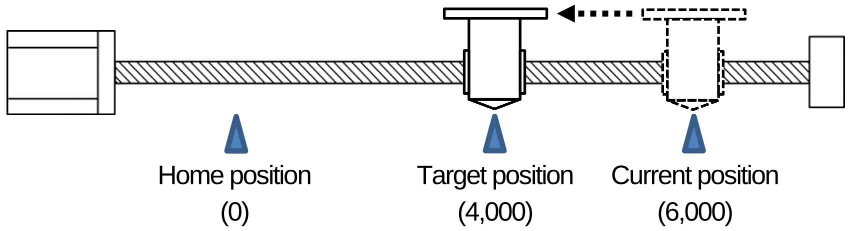
This is a positioning control method in which the target position is specified as an absolute position from the home position.

<Example>

Operation 1) Moving the object to position 6,000 (command value + 6,000)



Operation 2) Moving the object to position 4,000 (command value + 4,000)

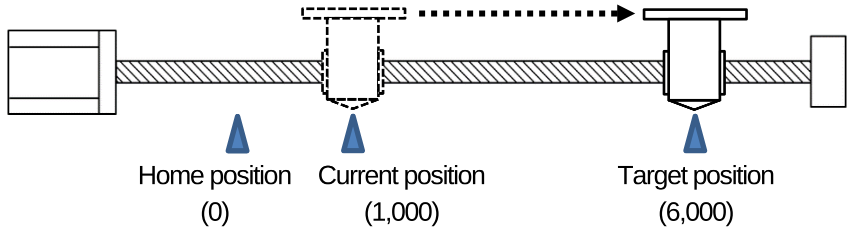


･Relative value positioning

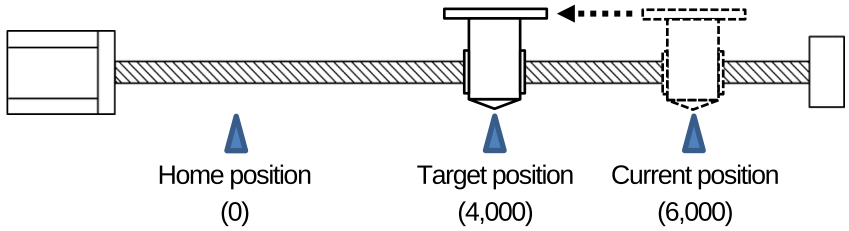
This is a positioning control method in which the target position is specified as a movement distance from the current position.

<Example>

Operation 1) Moving the object to position 6,000 (command value + 5,000)



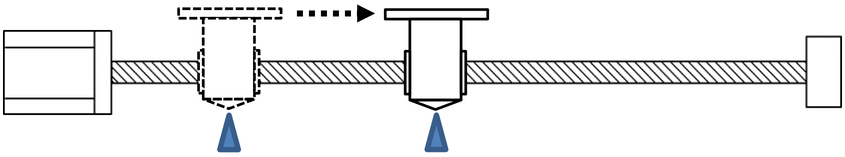
Operation 2) Moving the object to position 4,000 (command value - 2,000)



2.4 Creating Positioning Operation Program (Absolute Value Positioning)

Create a program for absolute value positioning. (PG_MoveAbsolute instruction)

Operation 1) Moving the object from the home position (0) to target position ① (5000) (Absolute value positioning)

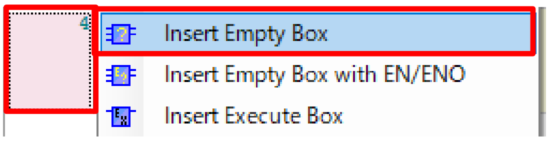


Step 1

Insert a new network using the same procedure as in "2.2 Creating Home Return Program".
Right-click in the network and select **Insert Network (below)**.

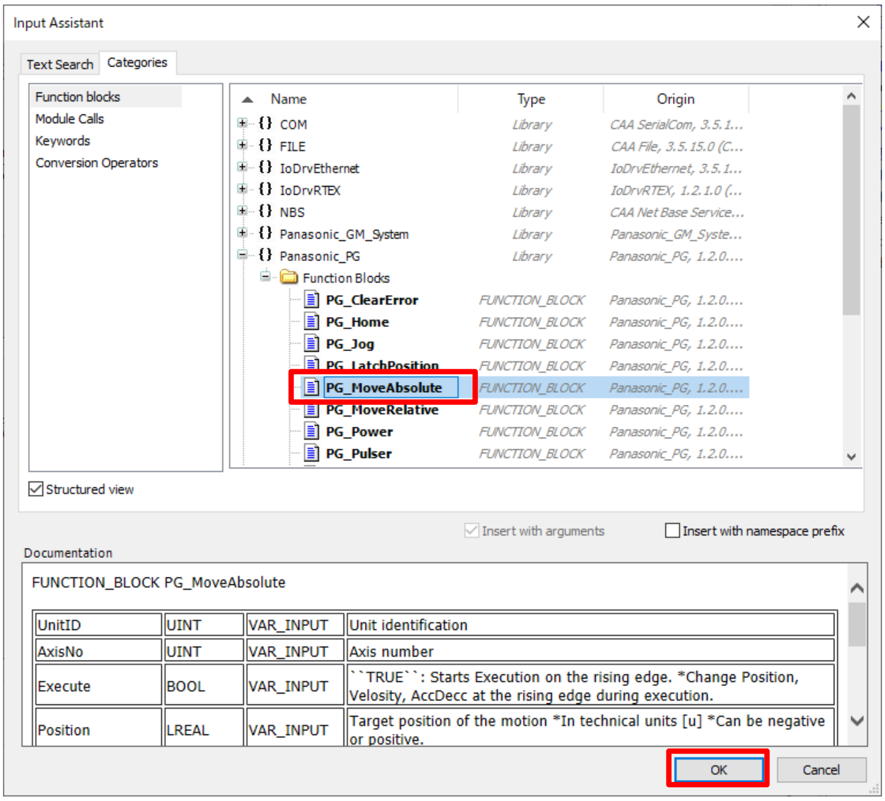
Step 2

Right-click the leftmost section of the new network and select **Insert Box**.



Step 3

The **Input Assistant** dialog box will open. In the **Categories** tab, select **Function blocks**.
In the right pane, select **{ } Panasonic_PG**, **Function Blocks**, and then **PG_MoveAbsolute** and click **OK**.



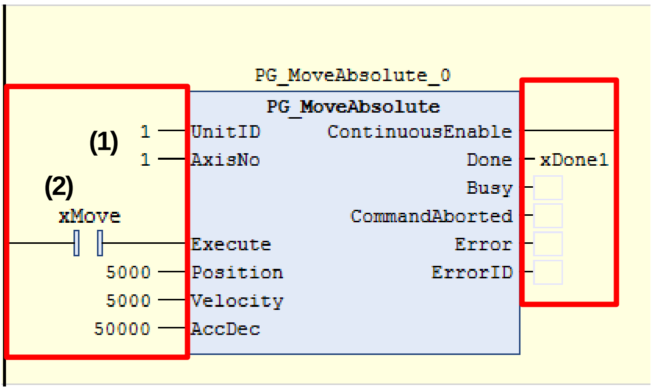
Name	Type	Origin
{} COM	Library	CAA SerialCom, 3.5.1...
{} FILE	Library	CAA File, 3.5.15.0 (C...
{} IoDrvEthernet	Library	IoDrvEthernet, 3.5.1...
{} IoDrvRTEX	Library	IoDrvRTEX, 1.2.1.0 (...
{} NBS	Library	CAA Net Base Service...
{} Panasonic_GM_System	Library	Panasonic_GM_Syste...
{} Panasonic_PG	Library	Panasonic_PG, 1.2.0....
Function Blocks		
PG_ClearError	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Home	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Jog	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_LatchPosition	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveAbsolute	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveRelative	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Power	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Pulser	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....

FUNCTION_BLOCK PG_MoveAbsolute

UnitID	UINT	VAR_INPUT	Unit identification
AxisNo	UINT	VAR_INPUT	Axis number
Execute	BOOL	VAR_INPUT	``TRUE``: Starts Execution on the rising edge. *Change Position, Velocity, AccDecc at the rising edge during execution.
Position	LREAL	VAR_INPUT	Target position of the motion *In technical units [u] *Can be negative or positive.

Step 4

Complete the function block as show below.

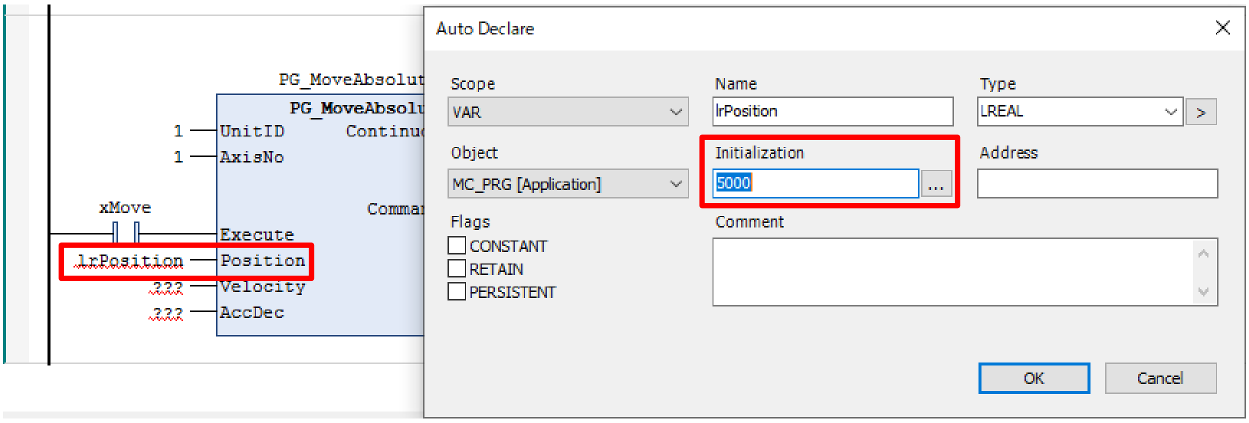


	Type	Parameter name	Set value	Description
(1)	Input	UnitID	1	Specifies the unit ID.
		AxisNo	1	Specifies the axis No.
(2)		Execute	xMove	Starts execution at the rising edge.
(3)		Position	5000	Specifies the target position (u).
		Velocity	5000	Specifies the maximum velocity (u/s).
		AccDec	50000	Specifies the acceleration/deceleration (u/s2).
(4)	Output	ContinuousEnable	Delete ???	TRUE: Position, Velocity, and AccDec can be changed.
		Done	xDone1	TRUE: FB operation is completed
		Busy	Delete ???	TRUE: FB operation in progress
		CommandAborted	Delete ???	TRUE: FB operation is interrupted
		Error	Delete ???	TRUE: FB is abnormally completed
		ErrorID	Delete ???	An error ID is output

Column ⑩: Assigning numerical variables

In the previous example, fixed value "5000" is substituted for the "Position" input parameter, but you can also declare a variable and set "5000" as the default value.

Declaring a variable enables you to change the value flexibly.



Auto Declare

Scope: VAR

Name: lrPosition

Type: LREAL

Object: MC_PRG [Application]

Initialization: 5000

Address:

Flags

□ CONSTANT

□ RETAIN

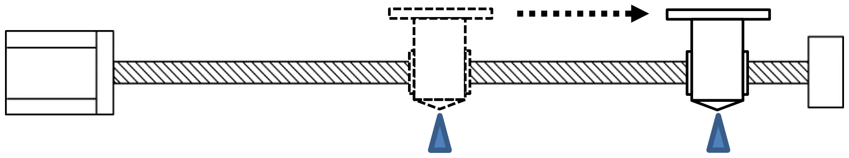
□ PERSISTENT

Comment:

2.5 Creating Positioning Operation Program (Relative Value Positioning)

Create a program for relative value positioning. (MC_MoveRelative instruction)

Operation 3) Moving the object from target position ① (5000) to target position ② (15000) (Relative value positioning)

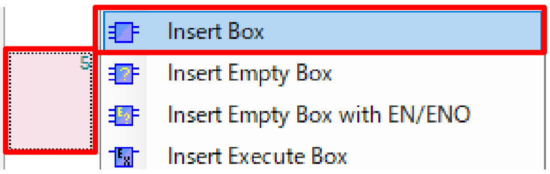


Step 1

Insert a new network using the same procedure as in "2.2 Creating Home Return Program".
Right-click in the network and select **Insert Network (below)**.

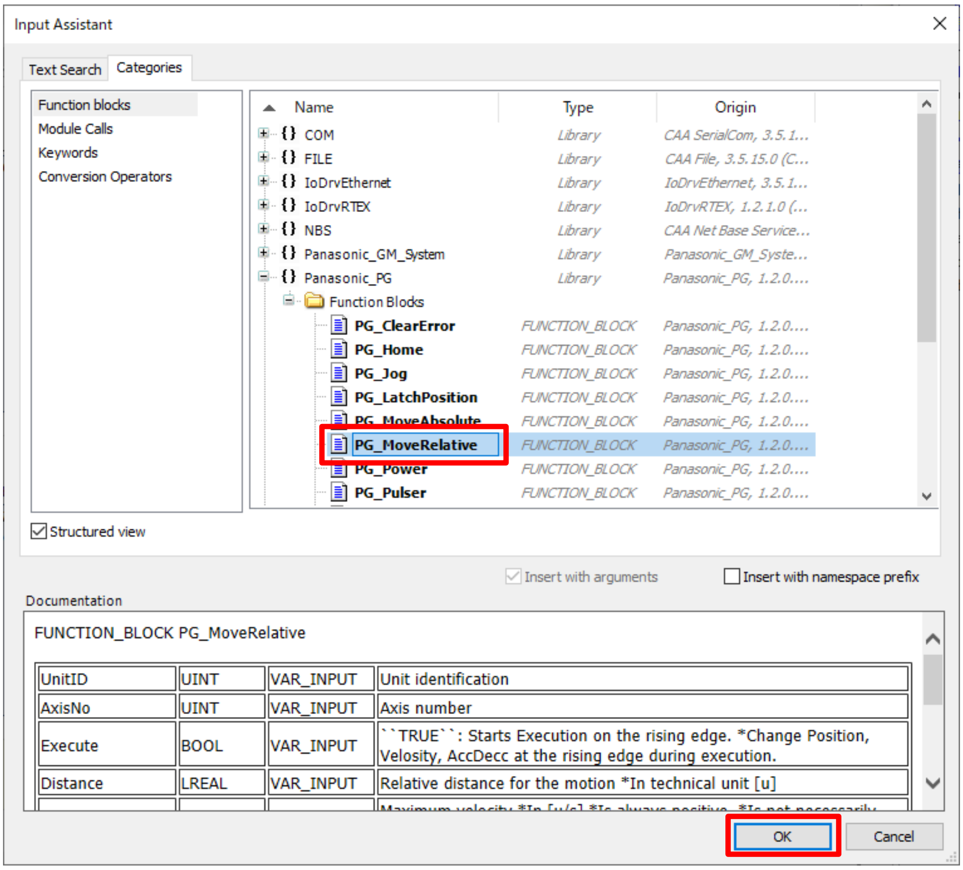
Step 2

Right-click the leftmost section of the new network and select **Insert Box**.



Step 3

The **Input Assistant** dialog box will open. In the **Categories** tab, select **Function blocks**.
In the right pane, select **{ } Panasonic_PG**, **Function Blocks**, and then **PG_MoveRelative** and click **OK**.



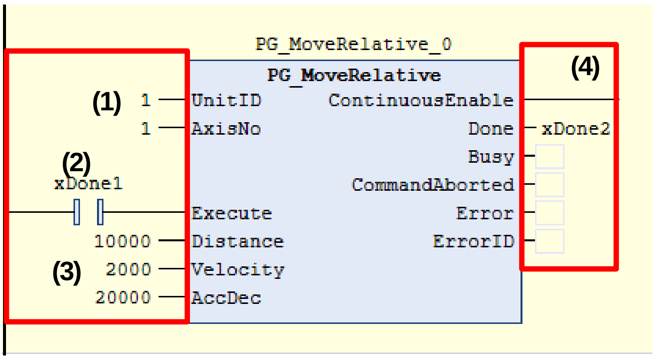
Name	Type	Origin
{} COM	Library	CAA SerialCom, 3.5.1...
{} FILE	Library	CAA File, 3.5.15.0 (C...
{} IoDrvEthernet	Library	IoDrvEthernet, 3.5.1...
{} IoDrvRTEX	Library	IoDrvRTEX, 1.2.1.0 (...
{} NBS	Library	CAA Net Base Service...
{} Panasonic_GM_System	Library	Panasonic_GM_Syste...
{} Panasonic_PG	Library	Panasonic_PG, 1.2.0....
Function Blocks		
PG_ClearError	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Home	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Jog	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_LatchPosition	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveAbsolute	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveRelative	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Power	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Pulser	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....

FUNCTION_BLOCK PG_MoveRelative

UnitID	UINT	VAR_INPUT	Unit identification
AxisNo	UINT	VAR_INPUT	Axis number
Execute	BOOL	VAR_INPUT	``TRUE``: Starts Execution on the rising edge. *Change Position, Velosity, AccDecc at the rising edge during execution.
Distance	LREAL	VAR_INPUT	Relative distance for the motion *In technical unit [u]
			Maximum velocity *In [u/s] *Is always positive. *Is not necessarily

Step 4

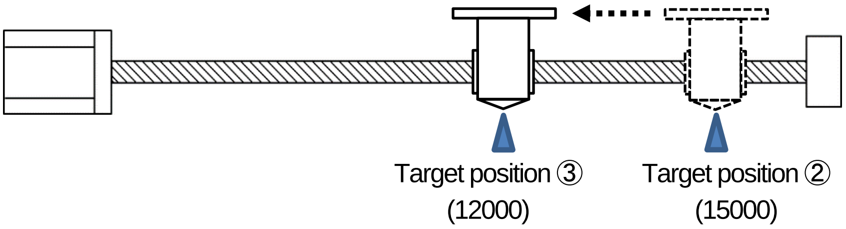
Complete the function block as show below.



	Type	Parameter name	Set value	Description
(1)	Input	UnitID	1	Specifies the unit ID.
		AxisNo	1	Specifies the axis No.
(2)		Execute	xDone1	Starts execution at the rising edge.
(3)		Position	10000	Specifies the target position (u).
		Velocity	2000	Specifies the maximum velocity (u/s).
		AccDec	20000	Specifies the acceleration/deceleration (u/s2).
(4)	Output	ContinuousEnable	Delete ???	TRUE: Position, Velocity, and AccDec can be changed.
		Done	xDone2	TRUE: FB operation is completed
		Busy	Delete ???	TRUE: FB operation in progress
		CommandAborted	Delete ???	TRUE: FB operation is interrupted
		Error	Delete ???	TRUE: FB is abnormally completed
		ErrorID	Delete ???	An error ID is output

Create a second program for relative value positioning. (MC_MoveRelative instruction)

Operation 4) Moving the object from target position ② (15000) to target position ③ (12000) (Relative value positioning)

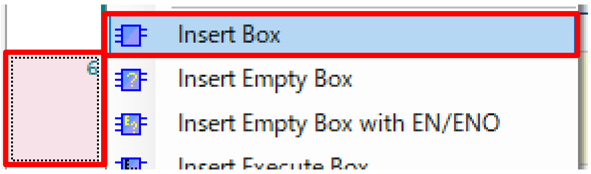


Step 1

Insert a new network using the same procedure as in "2.2 Creating Home Return Program".
Right-click in the network and select **Insert Network (below)**.

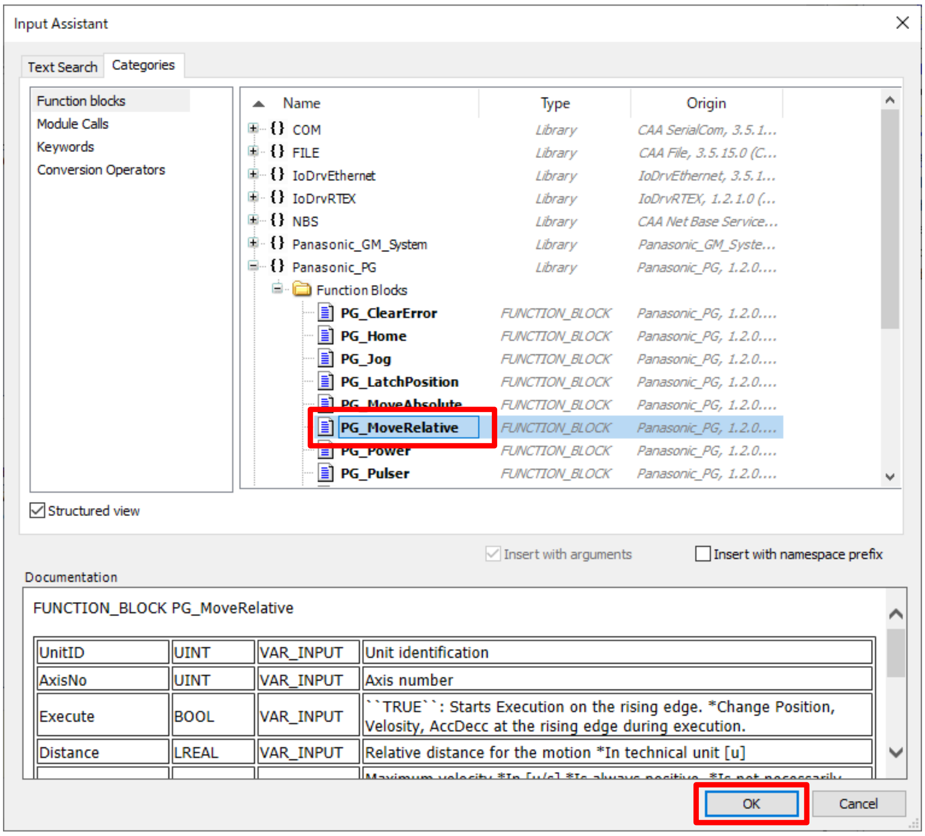
Step 2

Right-click the leftmost section of the new network and select **Insert Box**.



Step 3

The **Input Assistant** dialog box will open. In the **Categories** tab, select **Function blocks**.
In the right pane, select **{ } Panasonic_PG**, **Function Blocks**, and then **PG_MoveRelative** and click **OK**.



Input Assistant

Text Search | Categories

Function blocks
Module Calls
Keywords
Conversion Operators

Name	Type	Origin
{} COM	Library	CAA SerialCom, 3.5.1...
{} FILE	Library	CAA File, 3.5.15.0 (C...
{} IoDrvEthernet	Library	IoDrvEthernet, 3.5.1...
{} IoDrvRTEX	Library	IoDrvRTEX, 1.2.1.0 (...
{} NBS	Library	CAA Net Base Service...
{} Panasonic_GM_System	Library	Panasonic_GM_Syste...
{} Panasonic_PG	Library	Panasonic_PG, 1.2.0....
Function Blocks		
PG_ClearError	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Home	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Jog	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_LatchPosition	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveAbsolute	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_MoveRelative	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Power	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....
PG_Pulser	FUNCTION_BLOCK	Panasonic_PG, 1.2.0....

☑ Structured view

☑ Insert with arguments ☐ Insert with namespace prefix

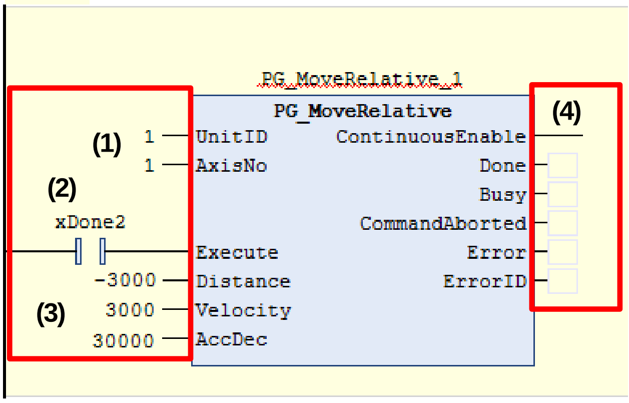
Documentation

FUNCTION_BLOCK PG_MoveRelative

UnitID	UINT	VAR_INPUT	Unit identification
AxisNo	UINT	VAR_INPUT	Axis number
Execute	BOOL	VAR_INPUT	``TRUE``: Starts Execution on the rising edge. *Change Position, Velosity, AccDecc at the rising edge during execution.
Distance	LREAL	VAR_INPUT	Relative distance for the motion *In technical unit [u]
			Maximum velocity *In [u/s] *Is always positive. *Is not necessarily

Step 4

Complete the function block as show below.



PG_MoveRelative_1
PG_MoveRelative

```
1      UnitID     ContinuousEnable
1      AxisNo     Done
                  Busy
xDone2            CommandAborted
       Execute    Error
-3000  Distance   ErrorID
3000   Velocity
30000  AccDec
```

	Type	Parameter name	Set value	Description
(1)	Input	UnitID	1	Specifies the unit ID.
		AxisNo	1	Specifies the axis No.
(2)		Execute	xDone2	Starts execution at the rising edge.
(3)		Distance	-3000	Specifies the target position (u).
		Velocity	3000	Specifies the maximum velocity (u/s).
		AccDec	30000	Specifies the acceleration/deceleration (u/s2).
(4)	Output	ContinuousEnable	Delete ???	TRUE: Position, Velocity, and AccDec can be changed.
		Done	Delete ???	TRUE: FB operation is completed
		Busy	Delete ???	TRUE: FB operation in progress
		CommandAborted	Delete ???	TRUE: FB operation is interrupted
		Error	Delete ???	TRUE: FB is abnormally completed
		ErrorID	Delete ???	An error ID is output

3 Operation Check

3.1 Network Scanning

The PC communicates with the GM1 controller using GM Programmer and LAN ports.
The IP address (default) of LAN port 1 is set as below.
An IP address on the same network that is different from the IP address of LAN port 1 must be set for the PC.

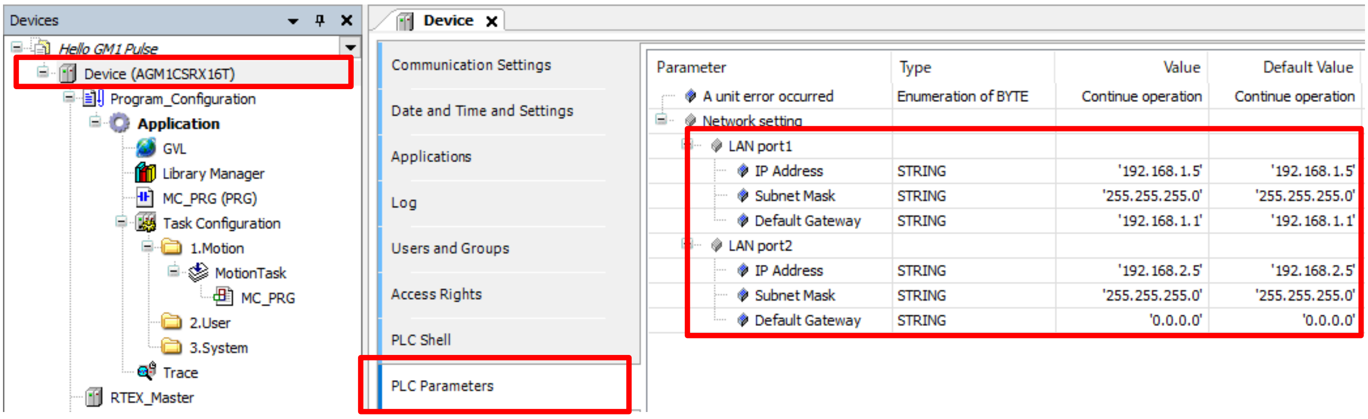
Interface: LAN port 1

IP address:	192.168.1.5
Subnet mask:	255.255.255.0
Default gateway:	192.168.1.1

Interface: PC

IP address:	192.168.1.10
Subnet mask:	255.255.255.0
Default gateway:	192.168.1.1

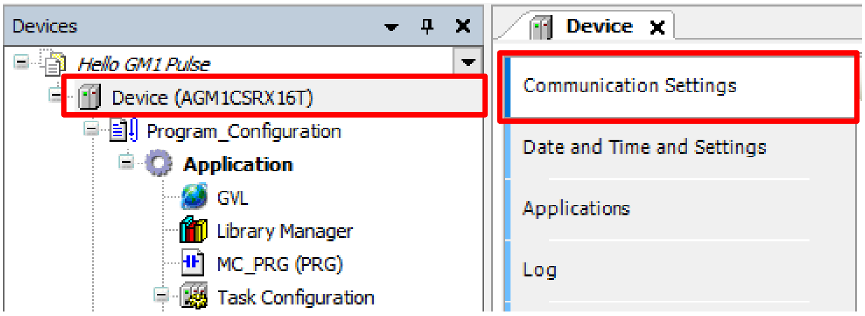
* The IP address of the GM1 controller can be checked and set in the main pane displayed by selecting **PLC Parameters** in the **Device** tab.



Parameter	Type	Value	Default Value
A unit error occurred	Enumeration of BYTE	Continue operation	Continue operation
Network setting			
LAN port1			
IP Address	STRING	'192.168.1.5'	'192.168.1.5'
Subnet Mask	STRING	'255.255.255.0'	'255.255.255.0'
Default Gateway	STRING	'192.168.1.1'	'192.168.1.1'
LAN port2			
IP Address	STRING	'192.168.2.5'	'192.168.2.5'
Subnet Mask	STRING	'255.255.255.0'	'255.255.255.0'
Default Gateway	STRING	'0.0.0.0'	'0.0.0.0'

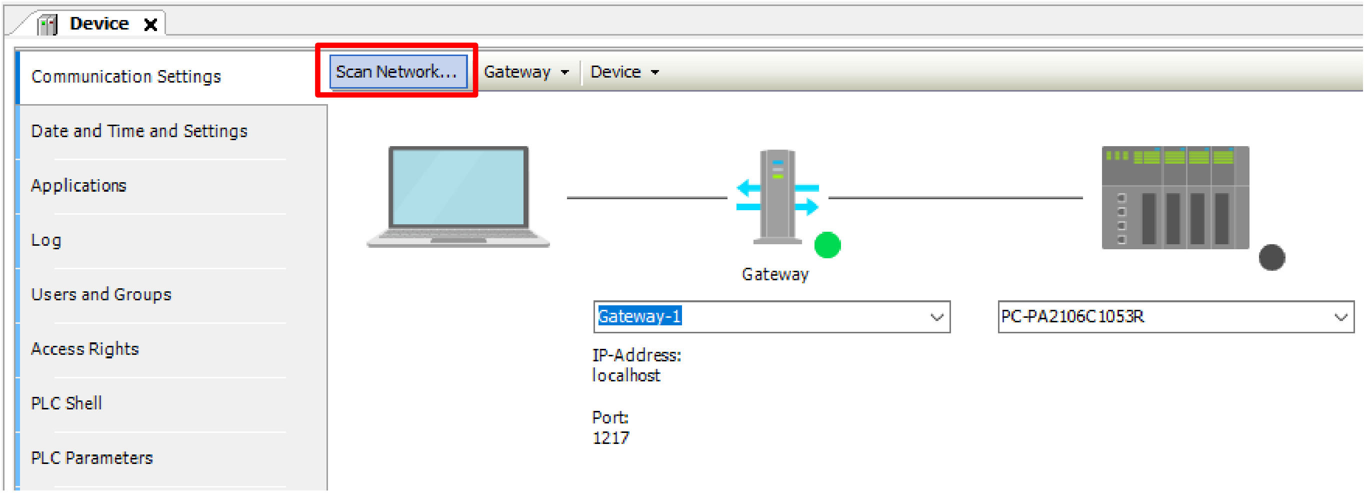
Step 1

Double-click the **Device** object and open the **Communication Settings** tab.



Step 2

Click **Scan Network**, select a controller to be connected, and click **OK**.

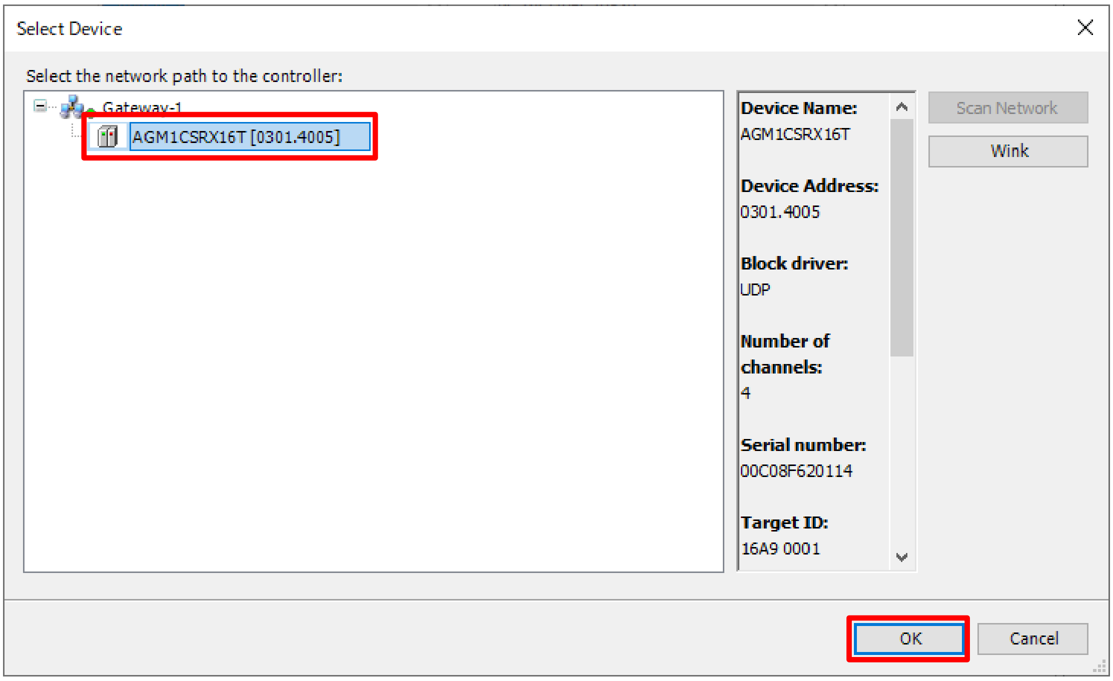


Gateway-1

IP-Address:
localhost

Port:
1217

PC-PA2106C1053R



Select the network path to the controller:

- Gateway-1
  - AGM1CSRX16T [0301.4005]

Device Name:
AGM1CSRX16T

Device Address:
0301.4005

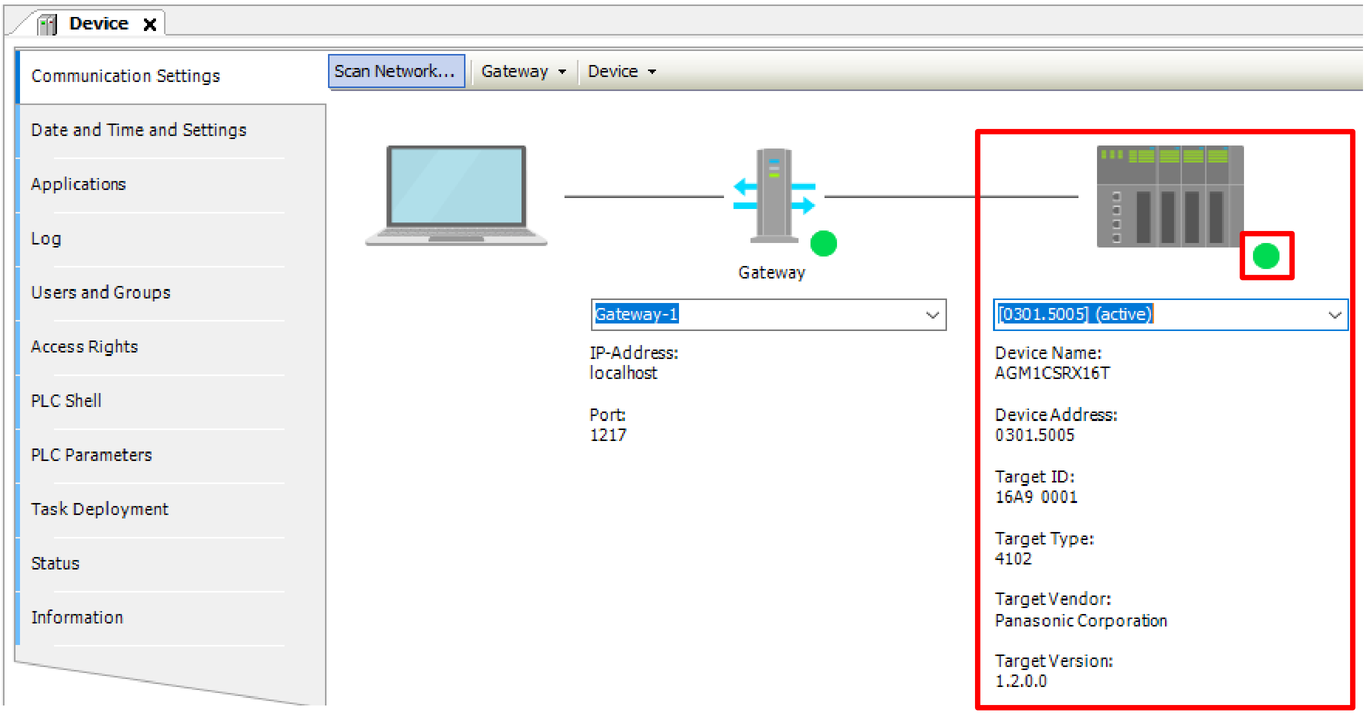
Block driver:
UDP

Number of channels:
4

Serial number:
00C08F620114

Target ID:
16A9 0001

When a connection is completed, the lamps light up in green (●).
If the display appears as shown in the figure below, it is OK.



Gateway-1

IP-Address:
localhost

Port:
1217

[0301.5005] (active)

Device Name:
AGM1CSRX16T

Device Address:
0301.5005

Target ID:
16A9 0001

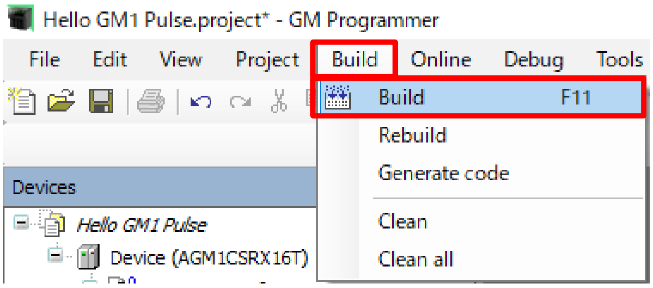
Target Type:
4102

Target Vendor:
Panasonic Corporation

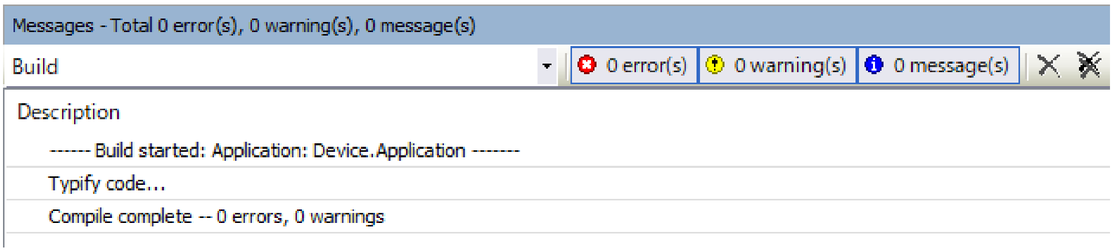
Target Version:
1.2.0.0

Step 3

Select **Build** from the **Build** menu to execute build.



If the processing is normal, "**Compile complete -- 0 errors, 0 warnings**" will be displayed as shown below.



```
------ Build started: Application: Device.Application -------
Typify code...
Compile complete -- 0 errors, 0 warnings
```

INFO

When a created program or settings are subject to a build process, objects in the application are compiled.
If codes are generated after the build process is executed, an application to be downloaded to the GM1 controller will be generated.

The syntax of all objects is verified when the build process is executed for the first time.
The syntax of only differences is verified when the build process is executed a second time and thereafter. No application code will be generated.

Column ⑪: Communication using USB ports

USB ports can be set as a communication interface between the GM1 controller and tool software such as GM Programmer or PANATERM Lite for GM, as below.

(1) Select **Add USB Port** from the **Online** menu.

(2) The **Add USB Port** dialog box will be displayed.

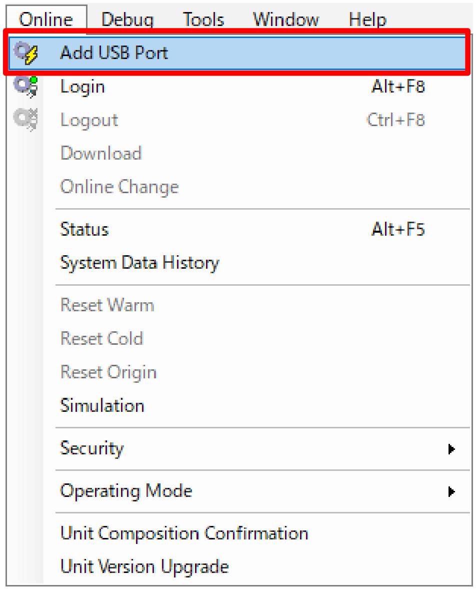
(3) Click **OK** to display a dialog box that restarts the gateway.

(4) Click **OK** to display the **Select Device** dialog box.

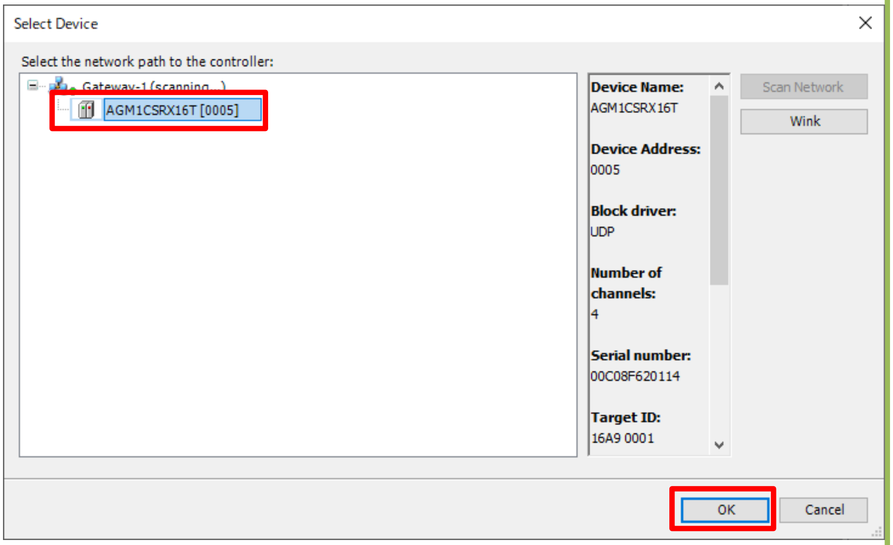
(5) Select a GM1 controller that you want to connect and click **OK**.

(6) When a connection is established, USB ports are added as a communication interface between the PC and GM1 controller.

(1)



(5)


Select Device

Select the network path to the controller:

Gateway-1 (scanning...)

AGM1CSRX16T [0005]

Device Name: AGM1CSRX16T

Device Address: 0005

Block driver: UDP

Number of channels: 4

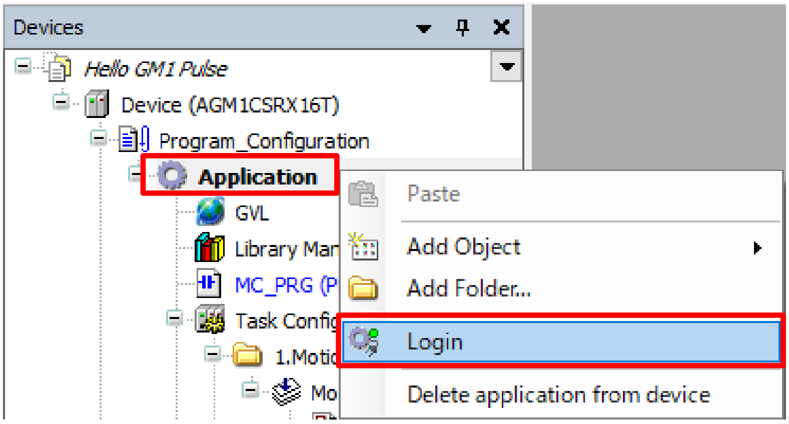
Serial number: 00C08F620114

Target ID: 16A9 0001

3.2 Login and Download

Step 1

Right-click the **Application** object and select **Login** to execute a download.



INFO

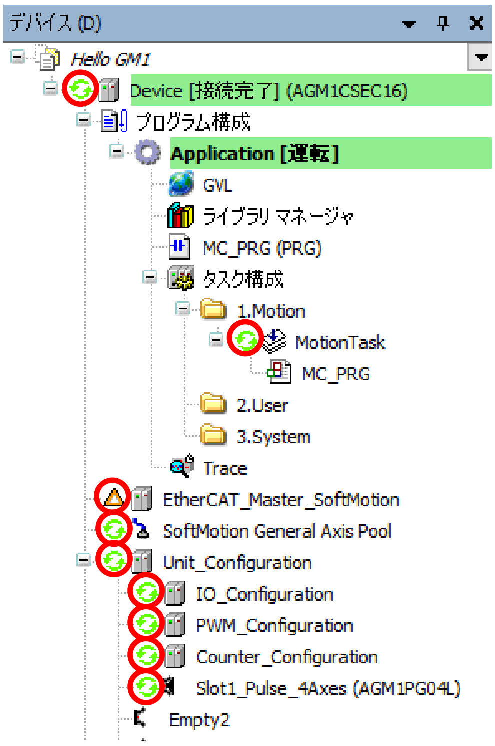
The **Device** and **Application** objects are displayed in the green background, indicating that they are in an online (connected) state.

⟳ mark appears on the left side of the **Device** object, indicating that the device is connected to the real machine.

[stop] appears on the right side of the **Application** object, indicating that the application is not running.

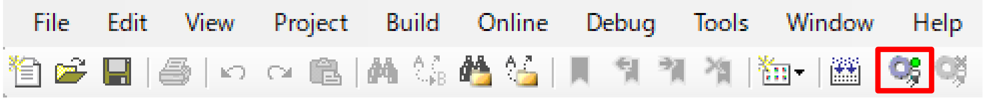
⟳ mark also appears on the left side of each device that is operating normally.

⚠ mark appears on the left side of each device that is not connected.



INFO

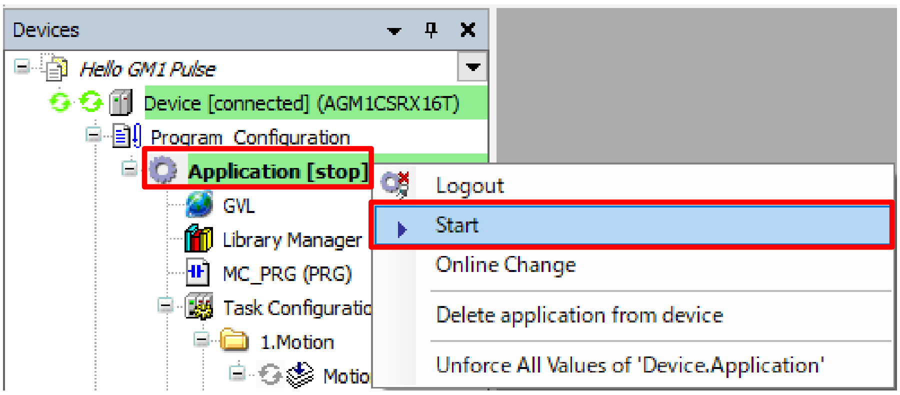
You can also log in from the toolbar.

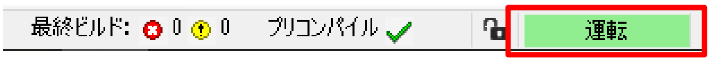


Step 2

Right-click the **Application [stop]** object and select **Start**.

When the application enters a running state, **RUN** appears in the status area on the bottom of the GM Programmer window.





INFO

At the time of login (connection), there are two states: STOP and RUN.

"STOP" indicates that the program is not running, and "RUN" indicates that the program is running.

3.3 Executing Commissioning and Home Return

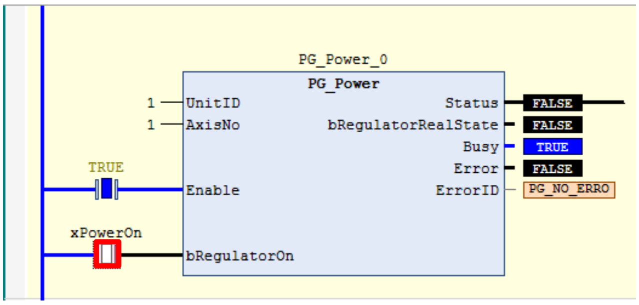
Step 1

Execute Servo ON.

To execute Servo ON, input **bRegulatorOn** in the **PG_Power** section must be set to TRUE.

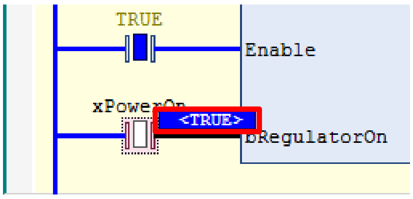
xServoON is set to FALSE by default and the inside of the contact is displayed in white, indicating that the servo is OFF.

Double-click the inside of the **xServoON** contact (the portion surrounded by the red frame in the figure below).

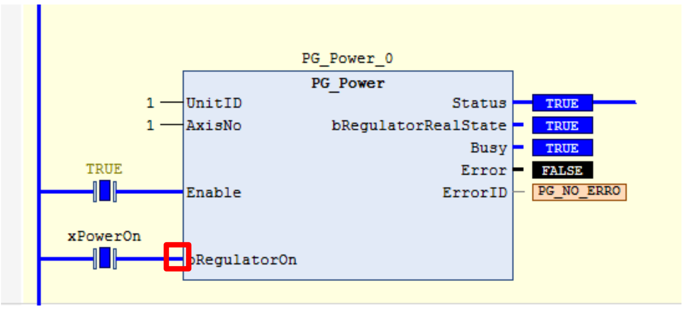


Double-clicking the above portion displays the set value <TRUE> for the variable on the right side of the contact.

At this stage, the value has not yet been written to perform operation.



To write input status change values such as TRUE and FALSE, **in this state**, press the F7 key while holding down the Ctrl key.

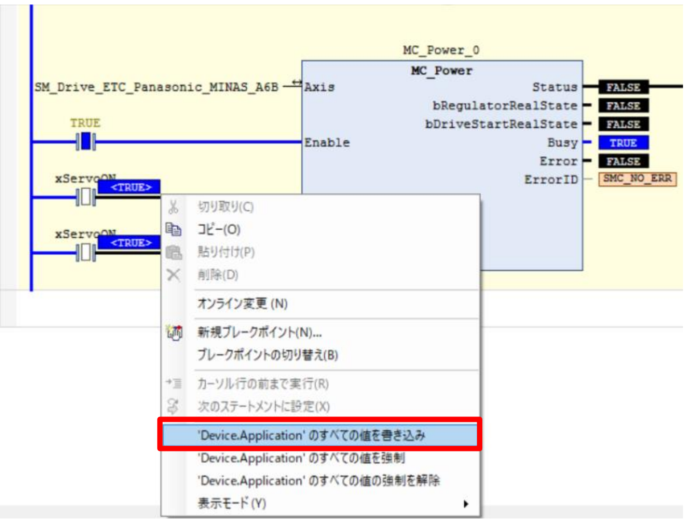


When **xServoON** is set to ON, the inside of the contact is displayed in blue, indicating that the servo is ON.

"**Status**" and "**bRegulatorRealState**" outputs are also set to TRUE , indicating that the servo is ON.

Column ⑫: How to write values

After displaying <TRUE> by double-clicking the relevant portion, you can also write values by right-clicking and selecting **Write All Values of 'Device.Application'**.

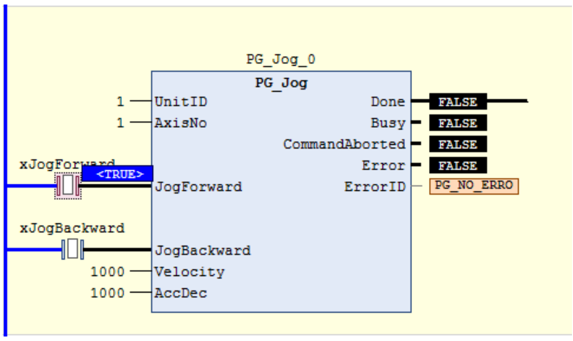

MC_Power_0
MC_Power
SM_Drive_ETC_Panasonic_MINAS_A6B
Axis
Status
bRegulatorRealState
bDriveStartRealState
Enable
Busy
Error
ErrorID
TRUE
FALSE
FALSE
FALSE
TRUE
FALSE
SMC_NO_ERR
xServoON
<TRUE>
xServoON
<TRUE>
切り取り(C)
コピー(O)
貼り付け(P)
削除(D)
オンライン変更 (N)
新規ブレークポイント(N)...
ブレークポイントの切り替え(B)
カーソル行の前まで実行(R)
次のステートメントに設定(X)
'Device.Application' のすべての値を書き込み
'Device.Application' のすべての値を強制
'Device.Application' のすべての値の強制を解除
表示モード (Y)

Step 2

When servo ON is complete, perform commissioning on the motor in JOG operation mode.

Double-click the startup contact of **JogForward** or **JogBackWard** for the **PG_JOG** instruction (JOG operation). When <TRUE> is displayed, press the Ctrl+F7 keys to execute JOG operation.

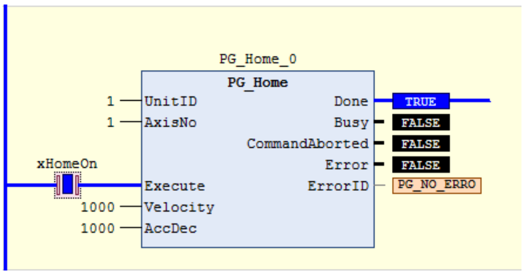
The PG_Jog instruction runs the motor while the startup contact is set to TRUE. To stop the motor, double-click the startup contact that is set to TRUE. When <FALSE> is displayed, press the Ctrl+F7 keys.



Step 3

When the operation check in JOG operation mode is complete, perform home return operation as below.

Double-click the startup contact of the **PG_Home** instruction (home return). When <TRUE> is displayed, press the Ctrl+F7 keys to execute home return operation.

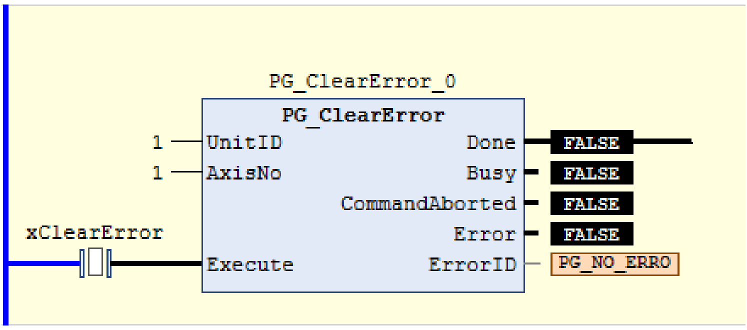


The motor runs and home return operation is executed.

When the home return operation is completed, **Done** is set to TRUE.

Column ⑬: What to do when a limit error occurs

If the over-limit input signal turns ON during motor operation, a limit error will occur, causing the ERR LED on the pulse output unit to light up.
In this case, use the PG_ClearError instruction to reset the error state and the PG_Home or PG_Jog instruction to move the object to a position where it is not detected by the over-limit input function.



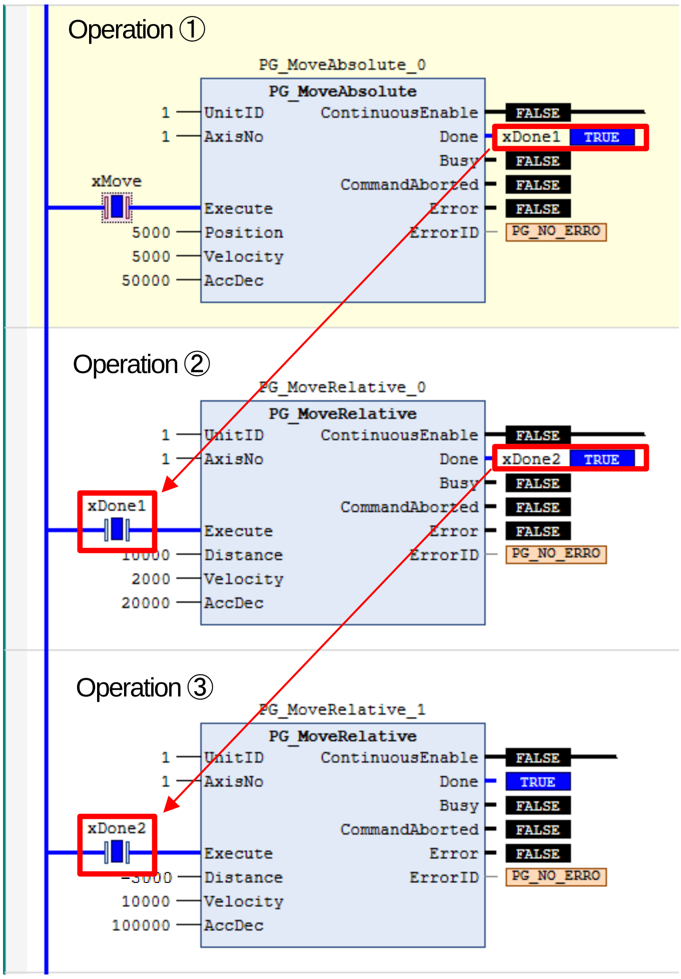
3.4 Performing Positioning Operation

Step 1

Reset **xMove** to TRUE.

After **MC_MoveAbsolute_0** operation is complete, **xDone1** is set to TRUE and **MC_MoveRelative_0** operates.

After **MC_MoveRelative_0** operation is complete, **xDone2** is set to TRUE and **MC_MoveRelative_1** operates.



Operation ①

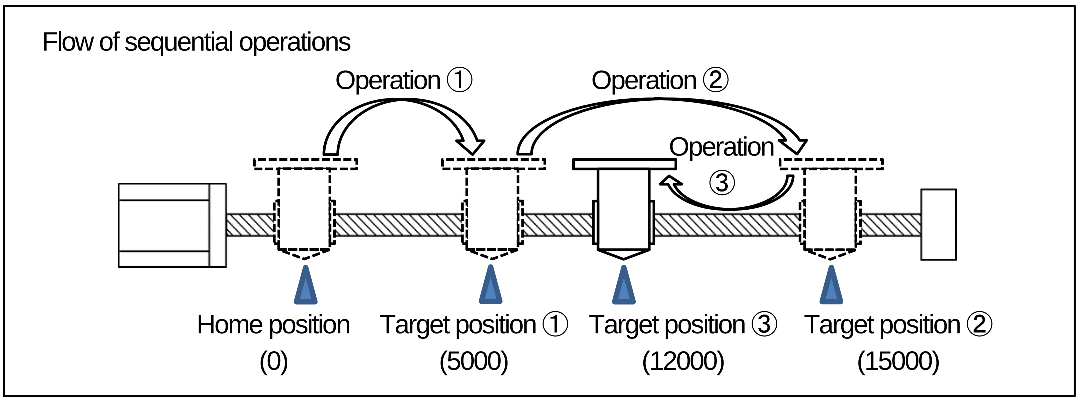
PG_MoveAbsolute_0 (PG_MoveAbsolute)
- UnitID: 1
- AxisNo: 1
- Execute: xMove
- Position: 5000
- Velocity: 5000
- AccDec: 50000
- ContinuousEnable: FALSE
- Done: xDone1 TRUE
- Busy: FALSE
- CommandAborted: FALSE
- Error: FALSE
- ErrorID: PG_NO_ERRO

Operation ②

PG_MoveRelative_0 (PG_MoveRelative)
- UnitID: 1
- AxisNo: 1
- Execute: xDone1
- Distance: 10000
- Velocity: 2000
- AccDec: 20000
- ContinuousEnable: FALSE
- Done: xDone2 TRUE
- Busy: FALSE
- CommandAborted: FALSE
- Error: FALSE
- ErrorID: PG_NO_ERRO

Operation ③

PG_MoveRelative_1 (PG_MoveRelative)
- UnitID: 1
- AxisNo: 1
- Execute: xDone2
- Distance: -3000
- Velocity: 10000
- AccDec: 100000
- ContinuousEnable: FALSE
- Done: TRUE
- Busy: FALSE
- CommandAborted: FALSE
- Error: FALSE
- ErrorID: PG_NO_ERRO



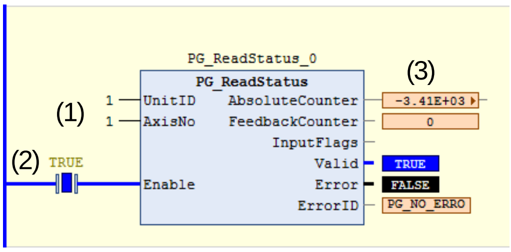
Step 2

Upon completion of operation check, stop the operation and log out.

4 Monitoring

4.1 Monitoring the Pulse Output Unit

The PG_ReadStatus instruction can be used to monitor the status of each axis.



	Type	Parameter name	Set value	Description
(1)	Input	UnitID	1	Specifies the unit ID.
		AxisNo	1	Specifies the axis No.
(2)		Enable	TRUE	TRUE: Enables execution of the FB
(3)	Output	AbsoluteCounter	Delete ???	Read value (Elapsed value)
		FeedbackCounter	Delete ???	Read value (Feedback value)
		InputFlag	Delete ???	Read value content (Input flag)
		Valid	Delete ???	TRUE: Enables use of parameters
		Error	Delete ???	TRUE: FB is abnormally completed
		ErrorID	Delete ???	An error ID is output

For details on InputFlag for the PG_ReadStatus instruction, see the following table.

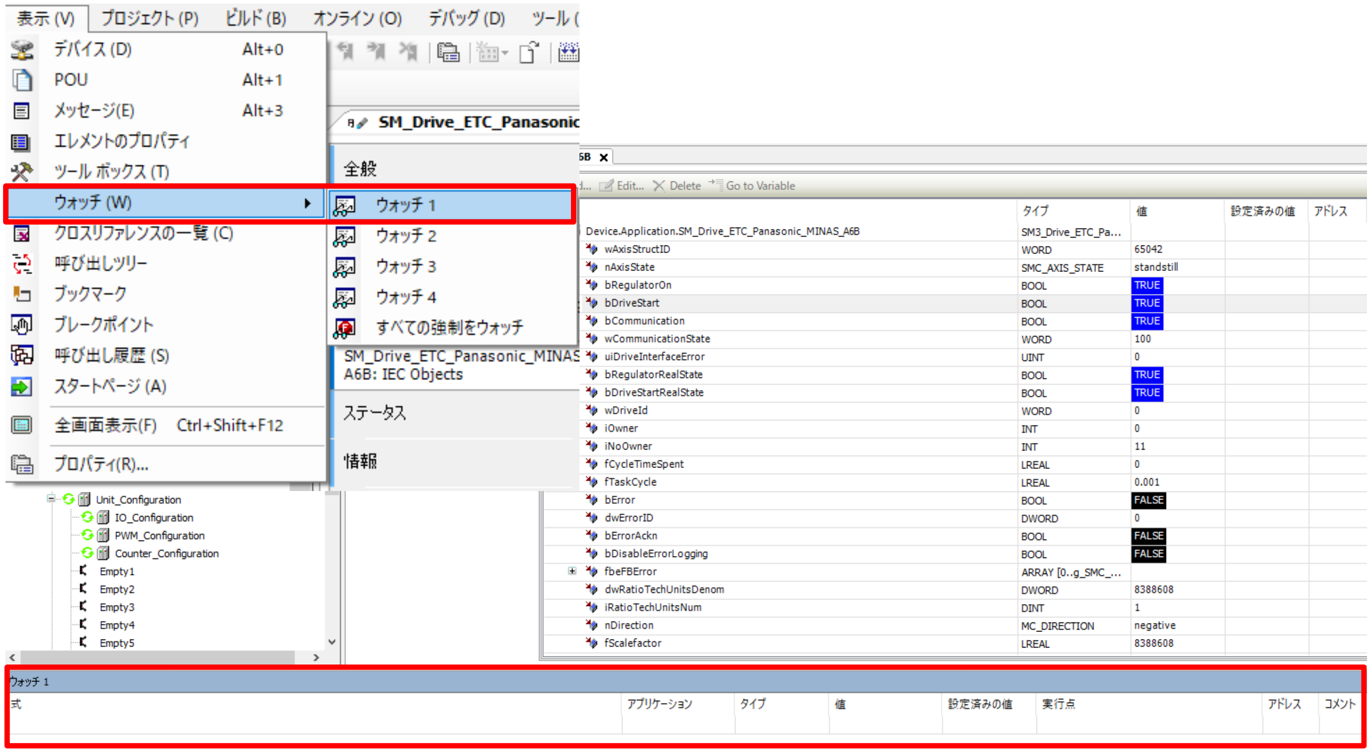
Parameter	Name	Description
PulseOutputBusy	Pulse output busy	TRUE when pulse output is in progress
PulseOutputDone	Pulse output done	TRUE when pulse output is completed
AccelerationZone	Acceleration zone	TRUE when in the acceleration zone
ConstantSpeedZone	Constant speed zone	TRUE when in the constant speed zone
DeceleartionZone	Deceleration zone	TRUE when in the deceleration zone
RotationDirection	Rotation direction	Elapsed value of monitoring in the rotation direction TRUE when the elapsed value is incrementing
HomeInput	Home input	Monitors the home input signal TRUE when the home input is enabled
NearHomeInput	Near home input	Monitors the near home input signal TRUE when the near home input is enabled
HomingDone	Home return done	TRUE when the home return is completed
OutputStopError	Output stop error	TRUE when an error occurs in the pulse output unit and output is stopped
SetValueChangeConfirmation	Set value change	Used to check rewriting of the set value during P-point control
OverPositiveLimitInput	Over limit (+)	TRUE when monitor contact limit input (+) for the over-limit input (+) signal is enabled
OverNegativeLimitInput	Over limit (-)	TRUE when monitor contact limit input (-) for the over-limit input (-) signal is enabled
TRUE TimingInputMonitor	Timing input monitor	TRUE when monitor contact position control start input for position control start input (timing input) is enabled
SetValueError	Set value error	TRUE when a set value error occurs
TRUE LimitError	Limit error	TRUE when an over-limit input signal is input during operation or at startup
TRUE ServoOnOutputState	Servo ON output state	TRUE when servo ON is output

4.2 Registering in Watch

By registering variables in the watch view, you can perform variable value management such as checking or changing variable values. This time, register each contact and elapsed values of pulse output.

Step 1

From the **View** menu, select **Watch** and then **Watch 1**. **Watch 1** will be displayed on the bottom of the main window.

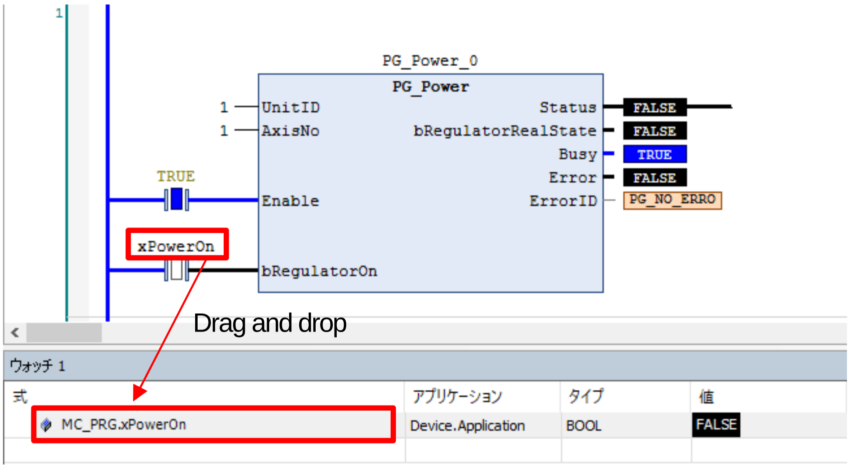


	タイプ	値	設定済みの値	アドレス
Device.Application.SM_Drive_ETC_Panasonic_MINAS_A6B	SM3_Drive_ETC_Pa...			
wAxisStructID	WORD	65042		
nAxisState	SMC_AXIS_STATE	standstill		
bRegulatorOn	BOOL	TRUE		
bDriveStart	BOOL	TRUE		
bCommunication	BOOL	TRUE		
wCommunicationState	WORD	100		
uiDriveInterfaceError	UINT	0		
bRegulatorRealState	BOOL	TRUE		
bDriveStartRealState	BOOL	TRUE		
wDriveId	WORD	0		
iOwner	INT	0		
iNoOwner	INT	11		
fCycleTimeSpent	LREAL	0		
fTaskCycle	LREAL	0.001		
bError	BOOL	FALSE		
dwErrorID	DWORD	0		
bErrorAckn	BOOL	FALSE		
bDisableErrorLogging	BOOL	FALSE		
fbeFBError	ARRAY [0..g_SMC_...			
dwRatioTechUnitsDenom	DWORD	8388608		
iRatioTechUnitsNum	DINT	1		
nDirection	MC_DIRECTION	negative		
fScalefactor	LREAL	8388608		

式	アプリケーション	タイプ	値	設定済みの値	実行点	アドレス	コメント

Step 2

The variables to be monitored can be registered in **Watch 1** by dragging and dropping them into the **Watch 1** pane.
Drag and drop **xPowerOn** to register it.

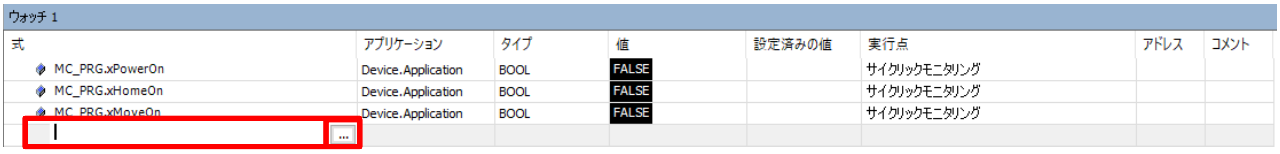


式	アプリケーション	タイプ	値
MC_PRG.xPowerOn	Device.Application	BOOL	FALSE

Register xHomeOn and xMoveOn in the same way.

Step 3

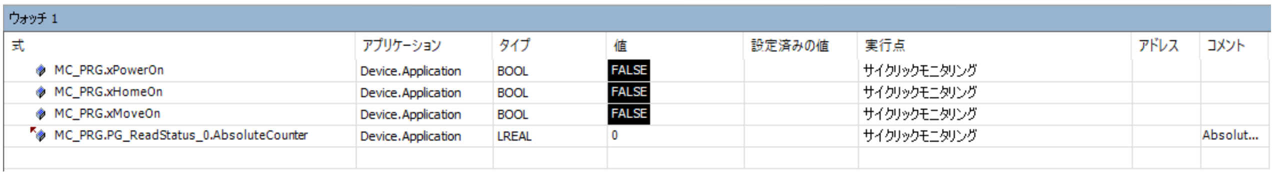
Register a variable to watch the elapsed value of pulse output.
Double-click in the empty column and then click … .



ウォッチ 1

式	アプリケーション	タイプ	値	設定済みの値	実行点	アドレス	コメント
MC_PRG.xPowerOn	Device.Application	BOOL	FALSE		サイクリックモニタリング		
MC_PRG.xHomeOn	Device.Application	BOOL	FALSE		サイクリックモニタリング		
MC_PRG.xMoveOn	Device.Application	BOOL	FALSE		サイクリックモニタリング		
I …							

Select **Device**, **Application**, and then **MC_PRG**, select **PG_ReadStatus_0** and then**AbsoluteCounter**, and click **OK**.

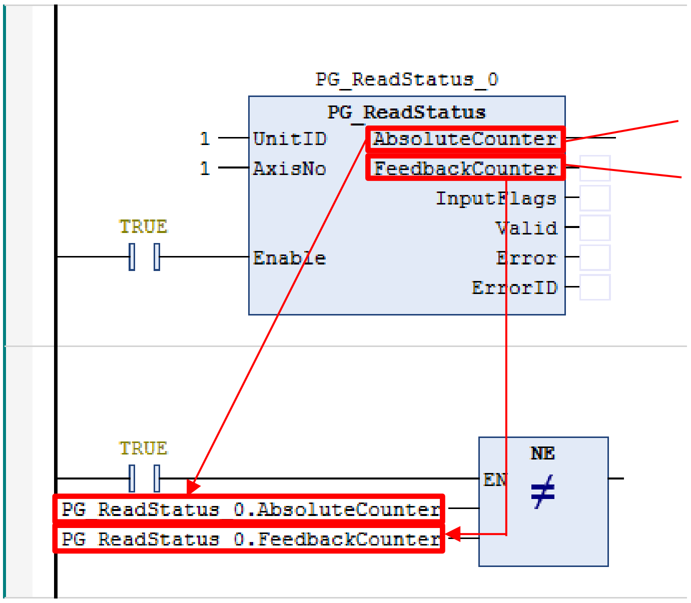


ウォッチ 1

式	アプリケーション	タイプ	値	設定済みの値	実行点	アドレス	コメント
MC_PRG.xPowerOn	Device.Application	BOOL	FALSE		サイクリックモニタリング		
MC_PRG.xHomeOn	Device.Application	BOOL	FALSE		サイクリックモニタリング		
MC_PRG.xMoveOn	Device.Application	BOOL	FALSE		サイクリックモニタリング		
MC_PRG.PG_ReadStatus_0.AbsoluteCounter	Device.Application	LREAL	0		サイクリックモニタリング		Absolut...

Column⑭: Feedback counter

The pulse output unit has a function to count pulse signals from external inputs such as encoders.
The number of pulses from external inputs is stored in the feedback counter and can be read using the PG_ReadStatus instruction. Stepping motor step-out can be detected by comparing feedback values with the elapsed values of pulse output.

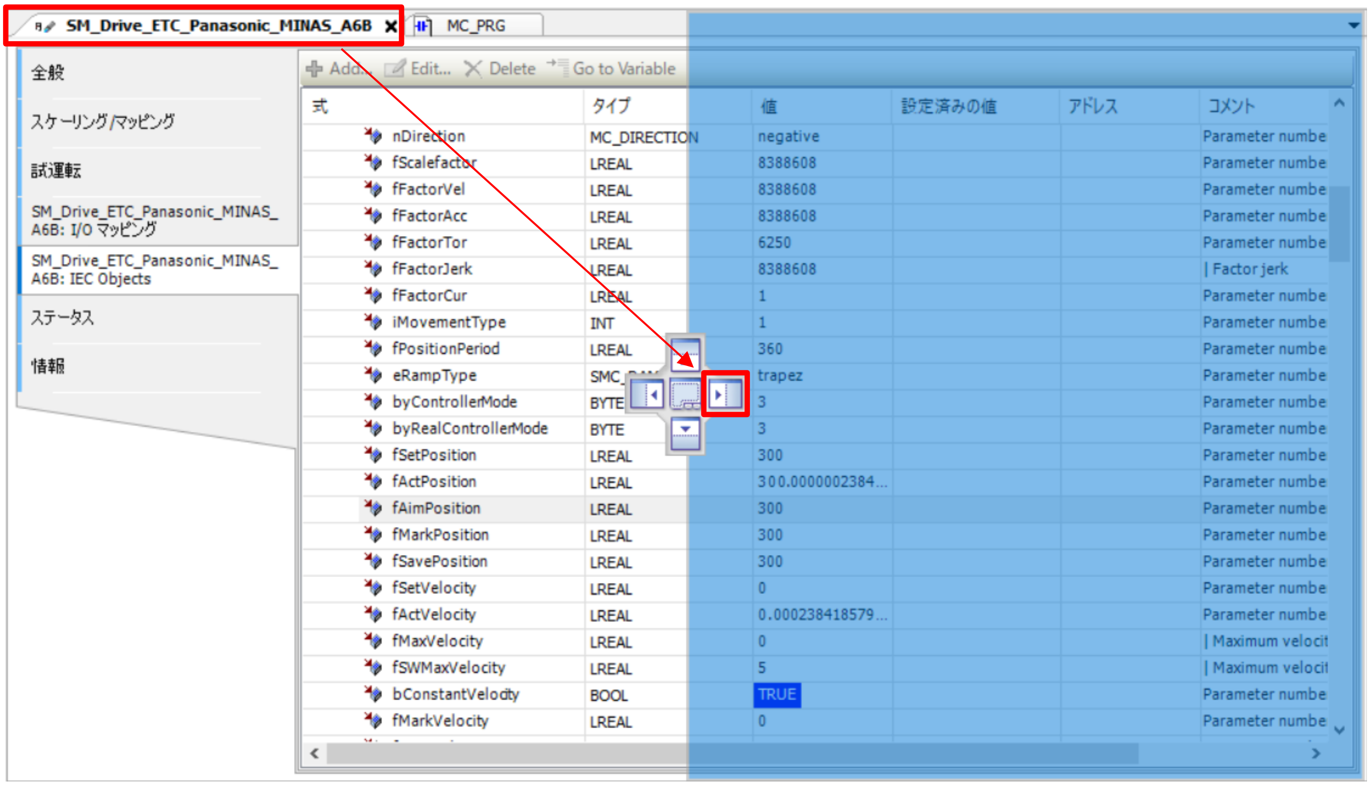


Elapsed value of pulse output

Feedback counter

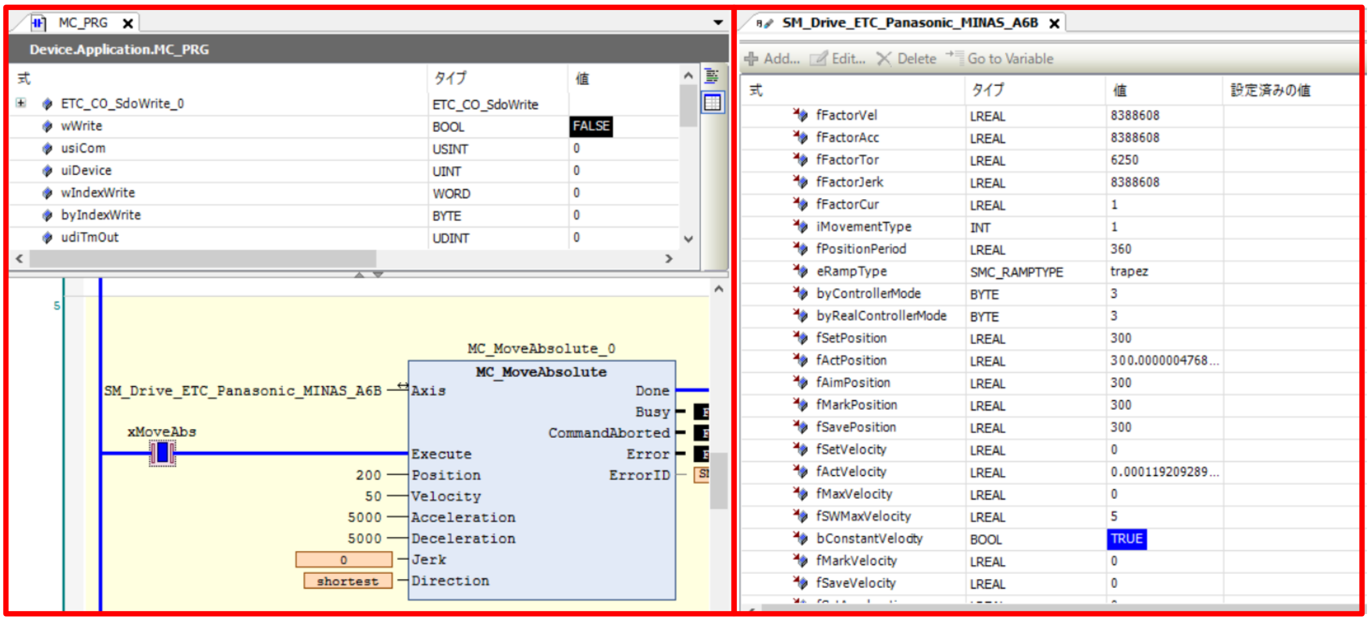
Column ⑮: How to separate tabs

Each tab of the main view can be separately displayed by dragging and dropping it.



SM_Drive_ETC_Panasonic_MINAS_A6B
MC_PRG
全般
スケーリング/マッピング
試運転
SM_Drive_ETC_Panasonic_MINAS_A6B: I/O マッピング
SM_Drive_ETC_Panasonic_MINAS_A6B: IEC Objects
ステータス
情報
Add...
Edit...
Delete
Go to Variable

式	タイプ	値	設定済みの値	アドレス	コメント
nDirection	MC_DIRECTION	negative			Parameter numbe
fScalefactor	LREAL	8388608			Parameter numbe
fFactorVel	LREAL	8388608			Parameter numbe
fFactorAcc	LREAL	8388608			Parameter numbe
fFactorTor	LREAL	6250			Parameter numbe
fFactorJerk	LREAL	8388608			| Factor jerk
fFactorCur	LREAL	1			Parameter numbe
iMovementType	INT	1			Parameter numbe
fPositionPeriod	LREAL	360			Parameter numbe
eRampType	SMC_	trapez			Parameter numbe
byControllerMode	BYTE	3			Parameter numbe
byRealControllerMode	BYTE	3			Parameter numbe
fSetPosition	LREAL	300			Parameter numbe
fActPosition	LREAL	300.0000002384...			Parameter numbe
fAimPosition	LREAL	300			Parameter numbe
fMarkPosition	LREAL	300			Parameter numbe
fSavePosition	LREAL	300			Parameter numbe
fSetVelocity	LREAL	0			Parameter numbe
fActVelocity	LREAL	0.000238418579...			Parameter numbe
fMaxVelocity	LREAL	0			| Maximum velocit
fSWMaxVelocity	LREAL	5			| Maximum velocit
bConstantVelocity	BOOL	TRUE			Parameter numbe
fMarkVelocity	LREAL	0			Parameter numbe



式	タイプ	値
ETC_CO_SdoWrite_0	ETC_CO_SdoWrite	
wWrite	BOOL	FALSE
usiCom	USINT	0
uiDevice	UINT	0
wIndexWrite	WORD	0
byIndexWrite	BYTE	0
udiTmOut	UDINT	0

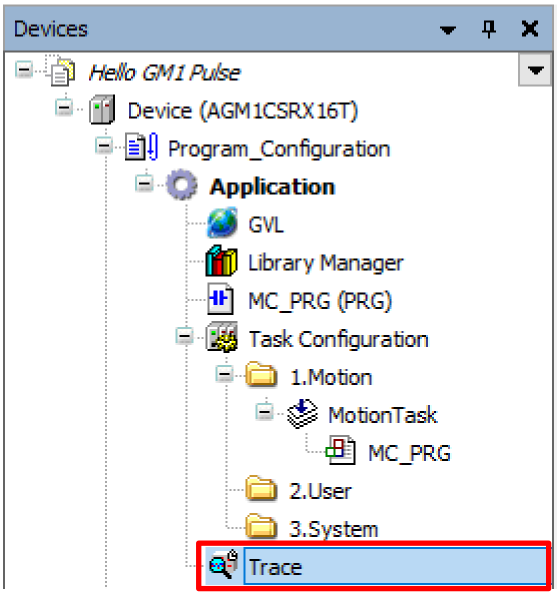
式	タイプ	値	設定済みの値
fFactorVel	LREAL	8388608	
fFactorAcc	LREAL	8388608	
fFactorTor	LREAL	6250	
fFactorJerk	LREAL	8388608	
fFactorCur	LREAL	1	
iMovementType	INT	1	
fPositionPeriod	LREAL	360	
eRampType	SMC_RAMPTYPE	trapez	
byControllerMode	BYTE	3	
byRealControllerMode	BYTE	3	
fSetPosition	LREAL	300	
fActPosition	LREAL	300.0000004768...	
fAimPosition	LREAL	300	
fMarkPosition	LREAL	300	
fSavePosition	LREAL	300	
fSetVelocity	LREAL	0	
fActVelocity	LREAL	0.000119209289...	
fMaxVelocity	LREAL	0	
fSWMaxVelocity	LREAL	5	
bConstantVelocity	BOOL	TRUE	
fMarkVelocity	LREAL	0	
fSaveVelocity	LREAL	0	

4.3 Adding Trace

The trace function allows the variable value histories in the GM1 controller to be checked in GM Programmer.

Step 1

Double-click **Trace**.



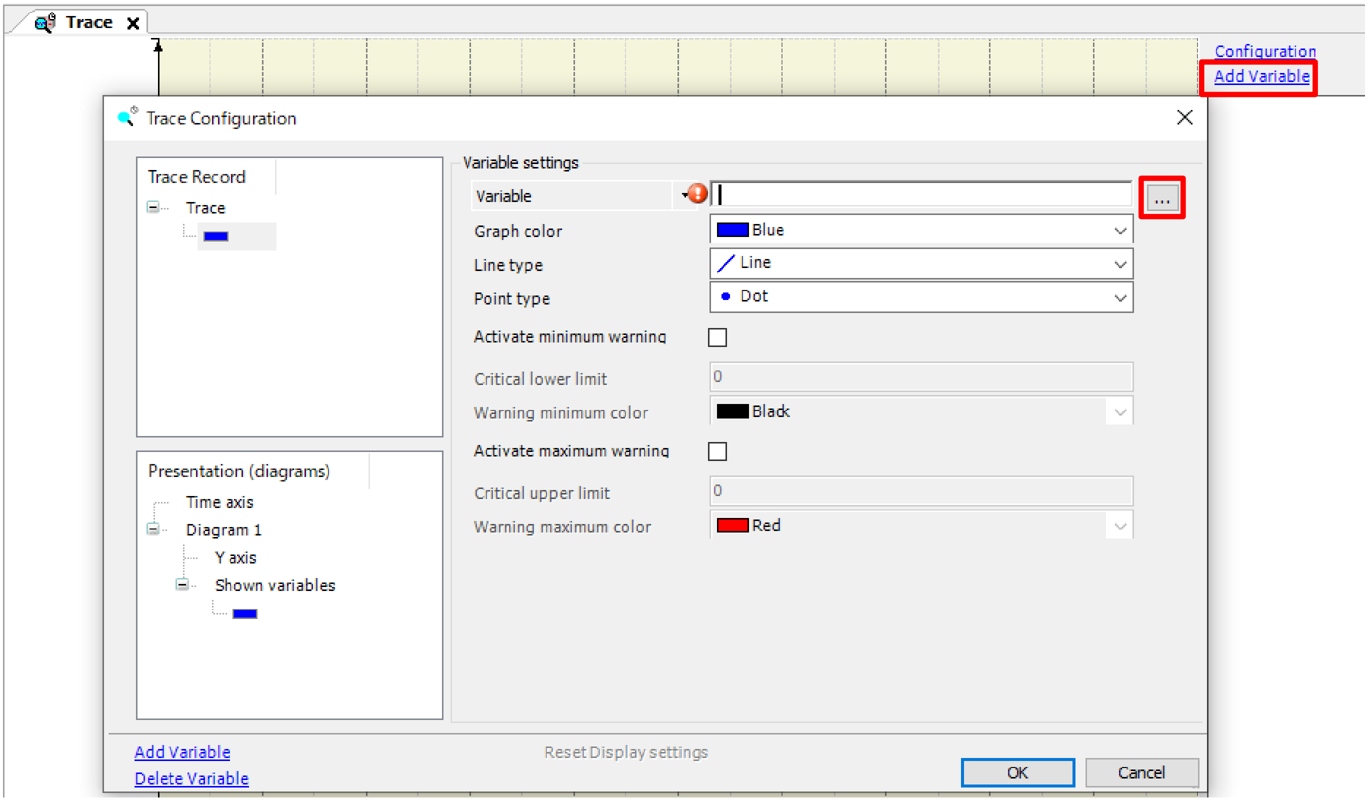
Step 2

Register variables to be traced. You can register and trace global variables and variables within function blocks.

Clicking **Add Variable** on the top right of the **Trace** window displays the **Trace Configuration** dialog box.

Click ... on the right side of the **Variable** field to display the **Input Assistant** dialog box.

*** ❗ indicates required input items. "Variable" is a required input item.**

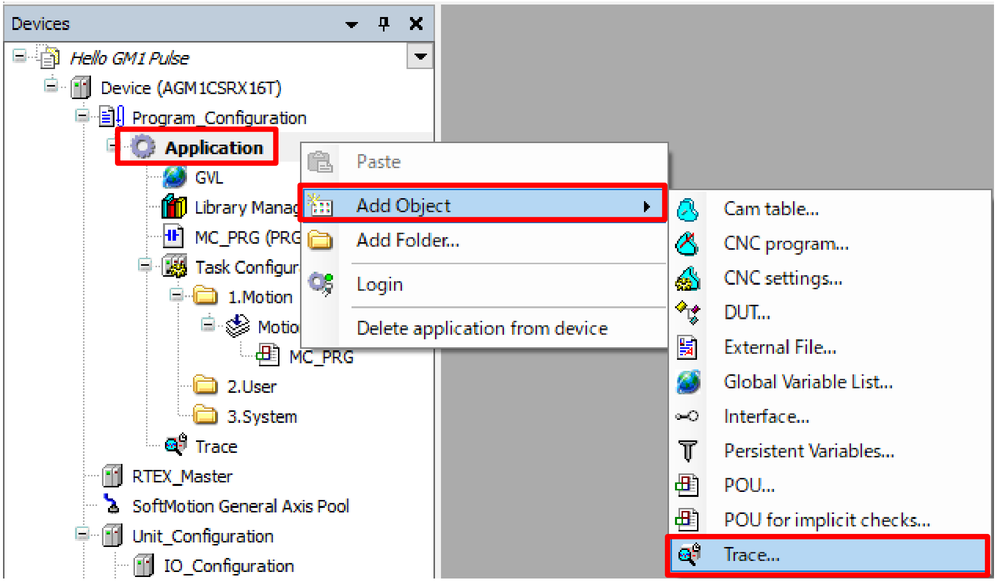


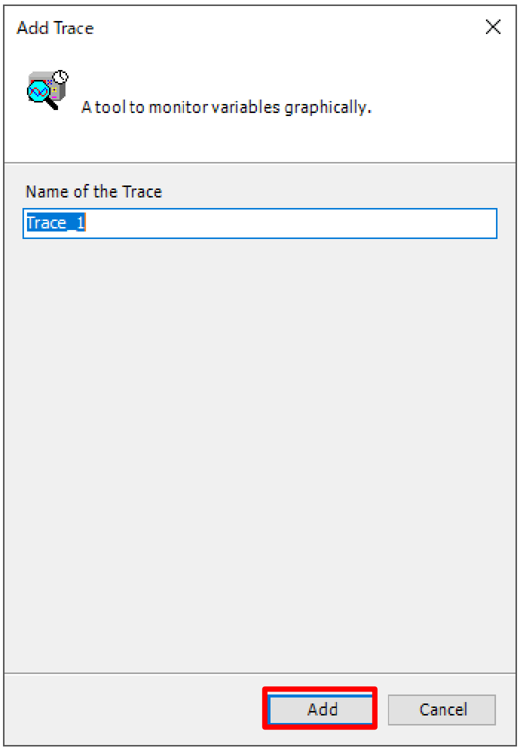
Variable settings

Variable ❗	
Graph color	Blue
Line type	Line
Point type	● Dot
Activate minimum warning	□
Critical lower limit	0
Warning minimum color	Black
Activate maximum warning	□
Critical upper limit	0
Warning maximum color	Red

Column ⑯: Adding objects (trace)

Right-click **Application** and select **Add Object** and then **Trace**.
The **Add Trace** dialog box will be displayed. Enter a trace name and click **Add**.



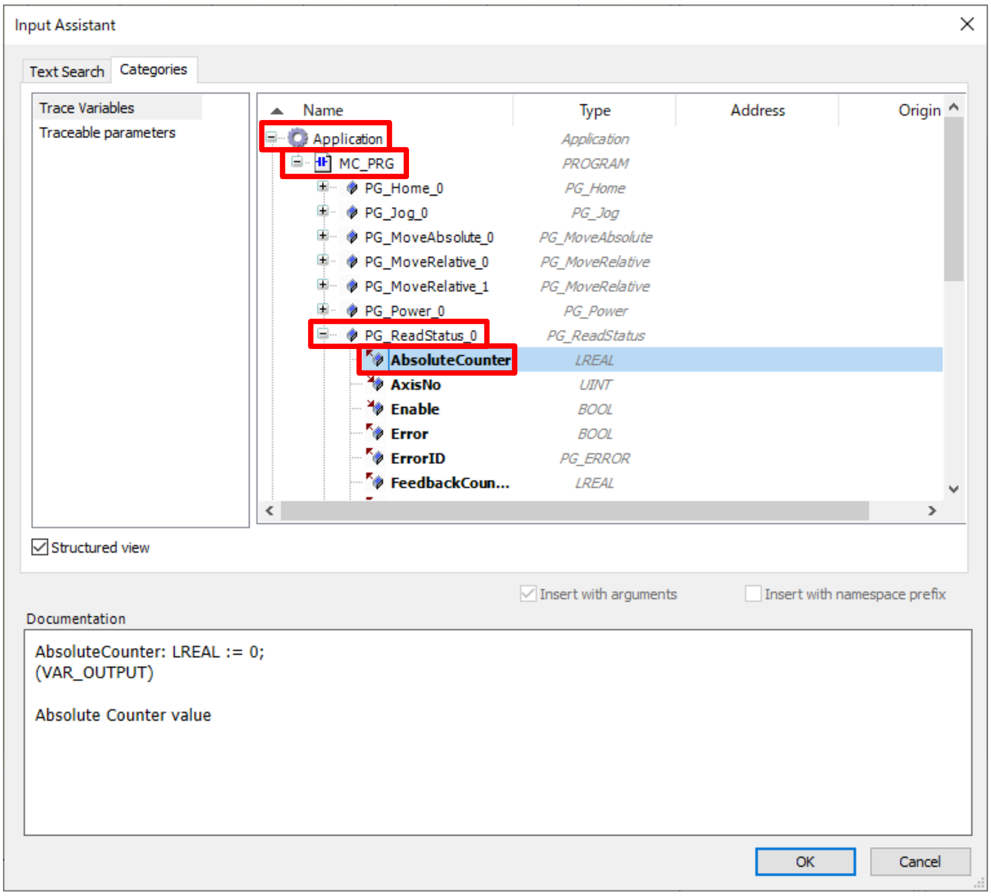


A tool to monitor variables graphically.

Name of the Trace: Trace_1

Step 3

In the **Input Assistant** window, select a variable to be traced and click **OK**, as below.
Select **Application**, **MC_PRG**, **PG_ReadStatus_0**, and then **AbsoluteCounter**.



Input Assistant

Text Search | Categories

Trace Variables
Traceable parameters

Name	Type	Address	Origin
Application	Application		
MC_PRG	PROGRAM		
PG_Home_0	PG_Home		
PG_Jog_0	PG_Jog		
PG_MoveAbsolute_0	PG_MoveAbsolute		
PG_MoveRelative_0	PG_MoveRelative		
PG_MoveRelative_1	PG_MoveRelative		
PG_Power_0	PG_Power		
PG_ReadStatus_0	PG_ReadStatus		
AbsoluteCounter	LREAL		
AxisNo	UINT		
Enable	BOOL		
Error	BOOL		
ErrorID	PG_ERROR		
FeedbackCoun...	LREAL		

☑ Structured view

☑ Insert with arguments ☐ Insert with namespace prefix

Documentation

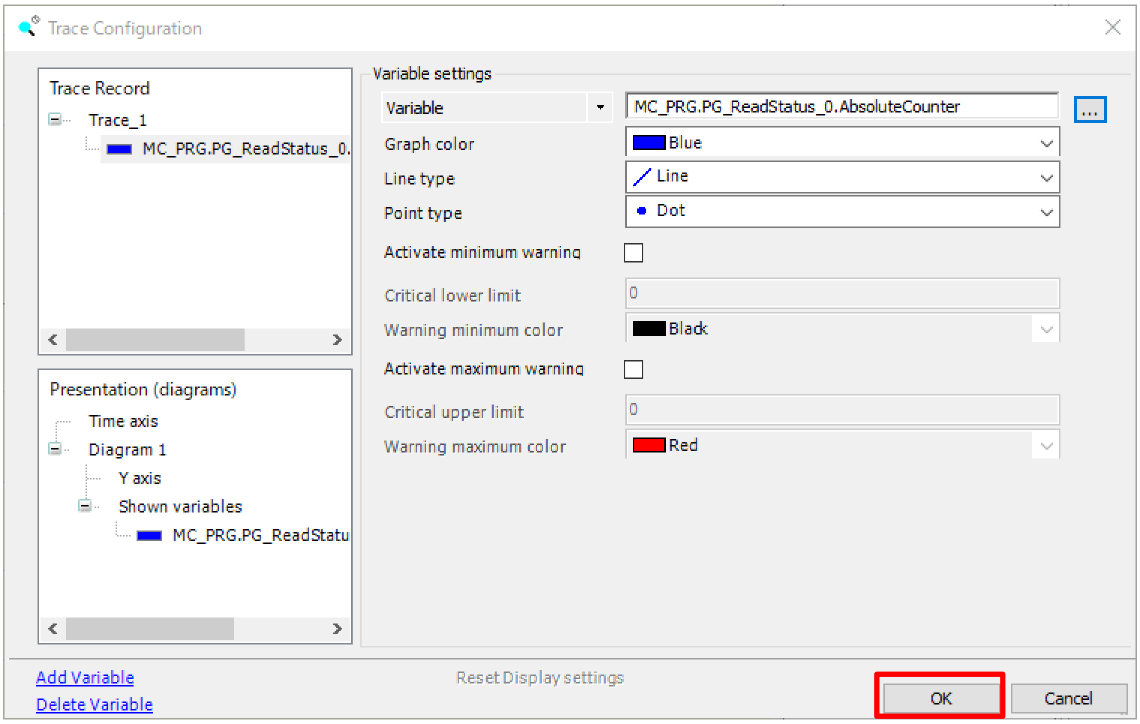
AbsoluteCounter: LREAL := 0;
(VAR_OUTPUT)

Absolute Counter value

OK Cancel

Step 4

Click **OK**.



Trace Configuration

Trace Record
Trace_1
MC_PRG.PG_ReadStatus_0.

Presentation (diagrams)
Time axis
Diagram 1
Y axis
Shown variables
MC_PRG.PG_ReadStatu

Variable settings

Variable	MC_PRG.PG_ReadStatus_0.AbsoluteCounter
Graph color	Blue
Line type	Line
Point type	Dot
Activate minimum warning	☐
Critical lower limit	0
Warning minimum color	Black
Activate maximum warning	☐
Critical upper limit	0
Warning maximum color	Red

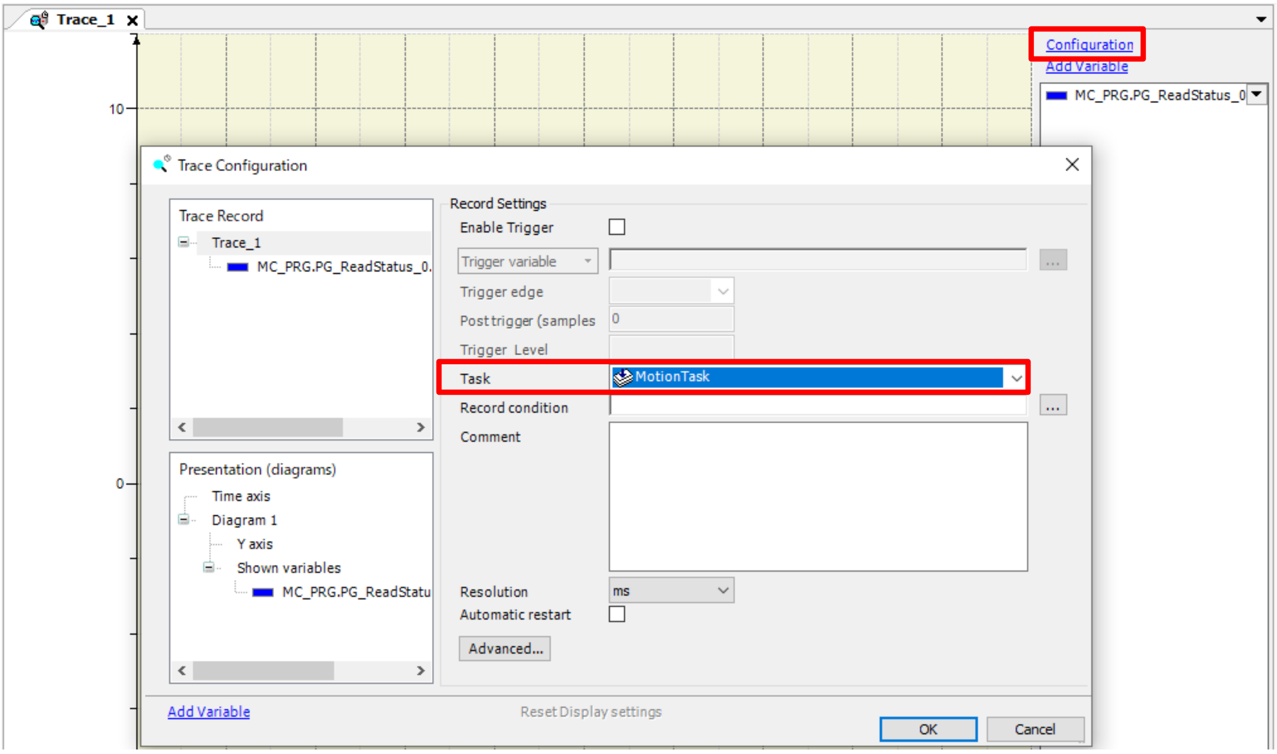
Add Variable
Delete Variable

Reset Display settings

OK Cancel

Step 5

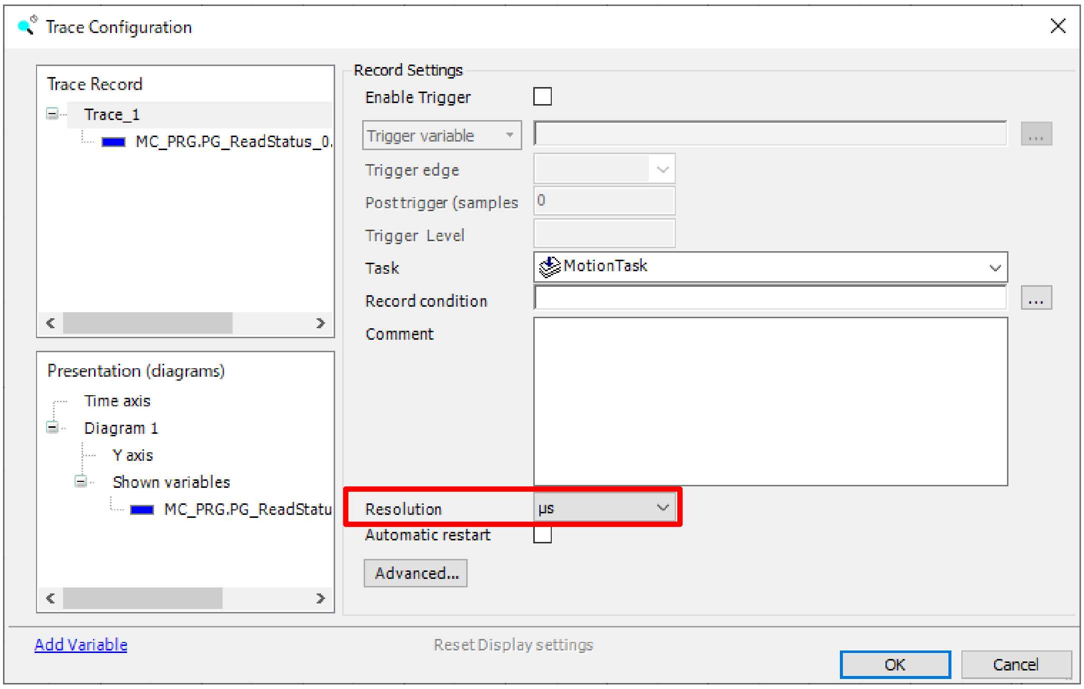
Clicking **Configuration** on the top right of the **Trace** window displays the **Trace Configuration** dialog box.
Open the drop-down list of **Task** (required input item) and select **MotionTask**.



Enable Trigger □
Trigger variable
Trigger edge
Posttrigger (samples 0
Trigger Level
Task MotionTask
Record condition
Comment
Resolution ms
Automatic restart □

Step 6

When a task is selected, **Resolution** becomes a required input item. Open the drop-down list of **Resolution** and select **μs**.



Enable Trigger □
Trigger variable
Trigger edge
Posttrigger (samples 0
Trigger Level
Task MotionTask
Record condition
Comment
Resolution μs
Automatic restart □

INFO

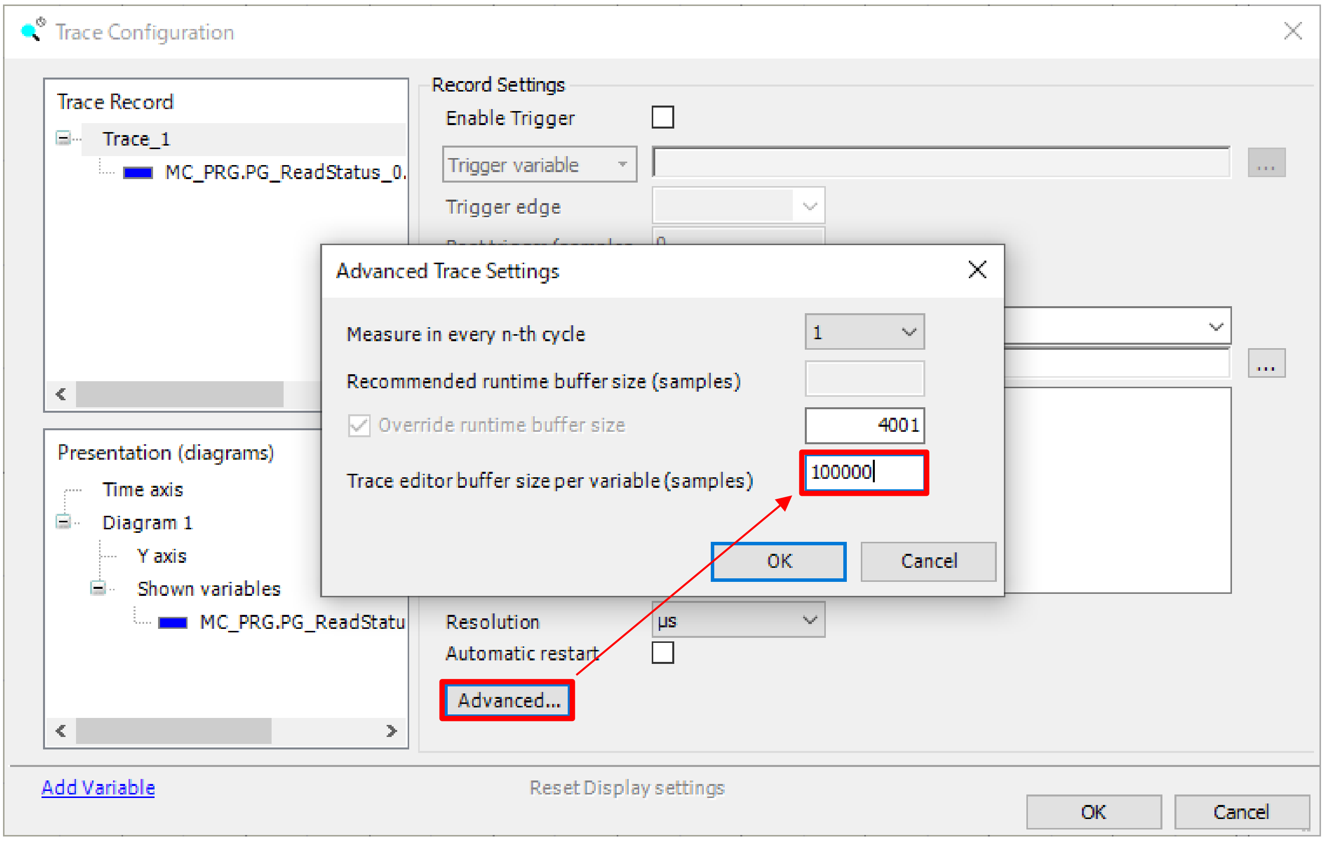
Task	When tracing a variable related to motion control, select **MotionTask**.
Resolution	The Motion Task interval is 1 ms. Therefore, if **ms** (default) is selected in the **Resolution** drop-down list, a caution message will be output, indicating that sampling resolution is too coarse. As a guide, sampling resolution should be half the interval.

Step 7

After selecting resolution, click **Advanced**.

Change the setting of **Trace editor buffer size per variable (samples)** to 100000.

Trace editor buffer size per variable (samples): Data display area viewed from the horizontal axis (time) of the **Trace** window



Advanced Trace Settings

Measure in every n-th cycle: 1

Recommended runtime buffer size (samples):

☑ Override runtime buffer size: 4001

Trace editor buffer size per variable (samples): 100000

Record Settings

Enable Trigger □

Resolution: µs

Automatic restart □

When **Trace editor buffer size per variable (samples)** is set to 100000, the buffer size becomes 50 times the recommended runtime buffer size.

Data equivalent to 2 s x 50 = approx. 100 s can be displayed in the **Trace** window.

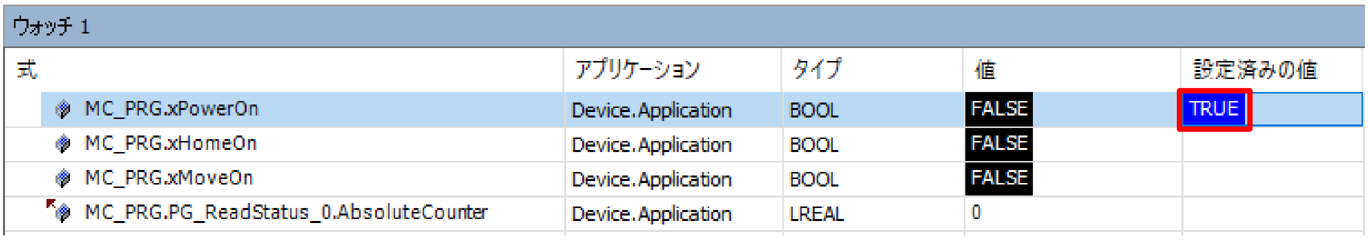
* This buffer size depends on the PC specifications. Therefore, note that if too large buffer size is set, the tool behavior will become sluggish.

INFO

Measure in every n-th cycle	Measurement interval = Set task interval
Recommended runtime buffer size (samples)	Number of buffers required for the recommended runtime (2 s) To collect trace data precisely, set **Measure in every n-th cycle** to 1 (measurement at every task interval) and prepare buffers equivalent to the recommended size.
Override runtime buffer size	Specifies the size of data to be overwritten in the number of buffers that is specified above Check box cleared: Leaves histories intact without overwriting data Check box selected: Leaves the amount of data that is specified in the override specification field on the right side
Trace editor buffer size per variable (samples)	Data display area viewed from the horizontal axis (time) of the **Trace** window

Step 8

Set xPowerON registered in Watch to TRUE to execute servo ON.
Change the value in the **Prepared value** column to <TRUE> and write the setting by pressing the Ctr+F7 keys.



ウォッチ 1

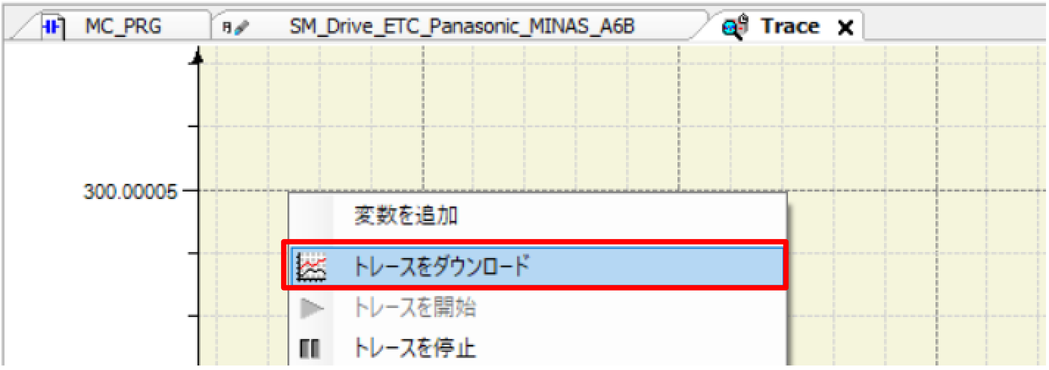
式	アプリケーション	タイプ	値	設定済みの値
MC_PRG.xPowerOn	Device.Application	BOOL	FALSE	TRUE
MC_PRG.xHomeOn	Device.Application	BOOL	FALSE	
MC_PRG.xMoveOn	Device.Application	BOOL	FALSE	
MC_PRG.PG_ReadStatus_0.AbsoluteCounter	Device.Application	LREAL	0	

Step 9

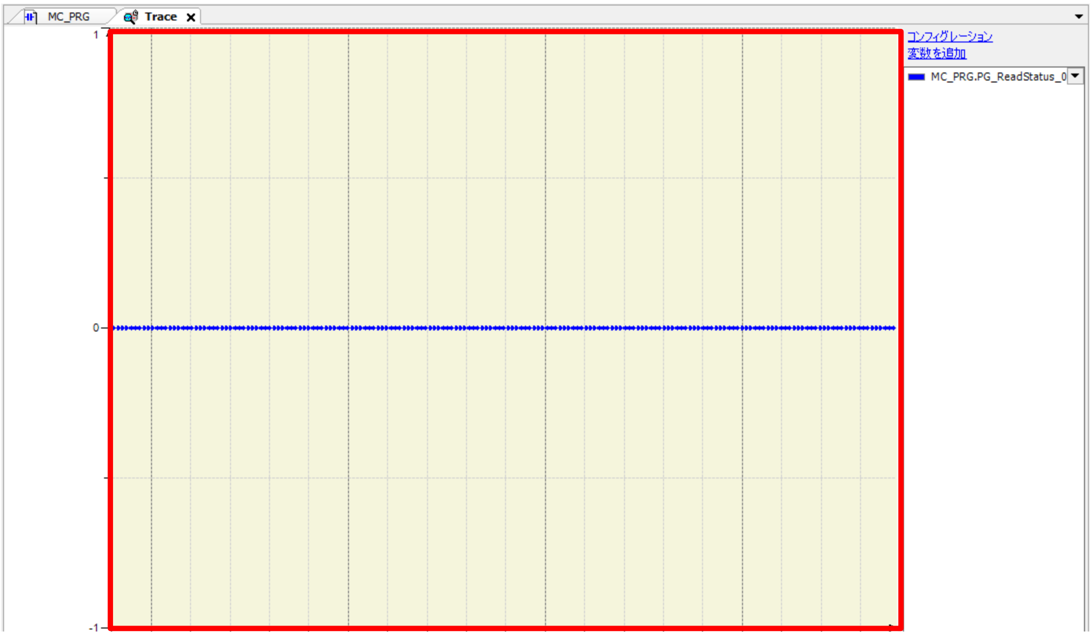
Next, change the setting of xHomeOn to TRUE.

Step 10

Before checking trace operation, right-click in the **Trace** window. Select **Download Trace**.

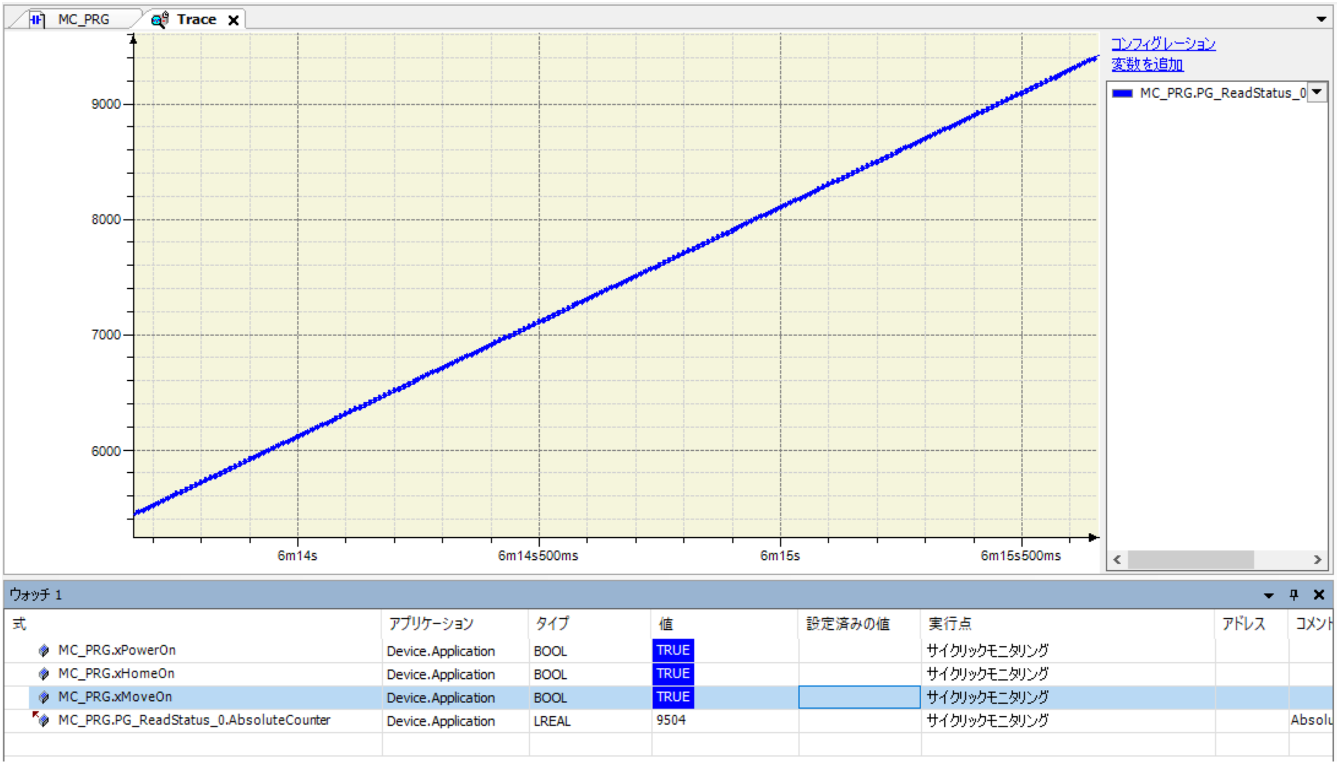


When **Download Trace** is selected, waveforms are displayed.



Step 11

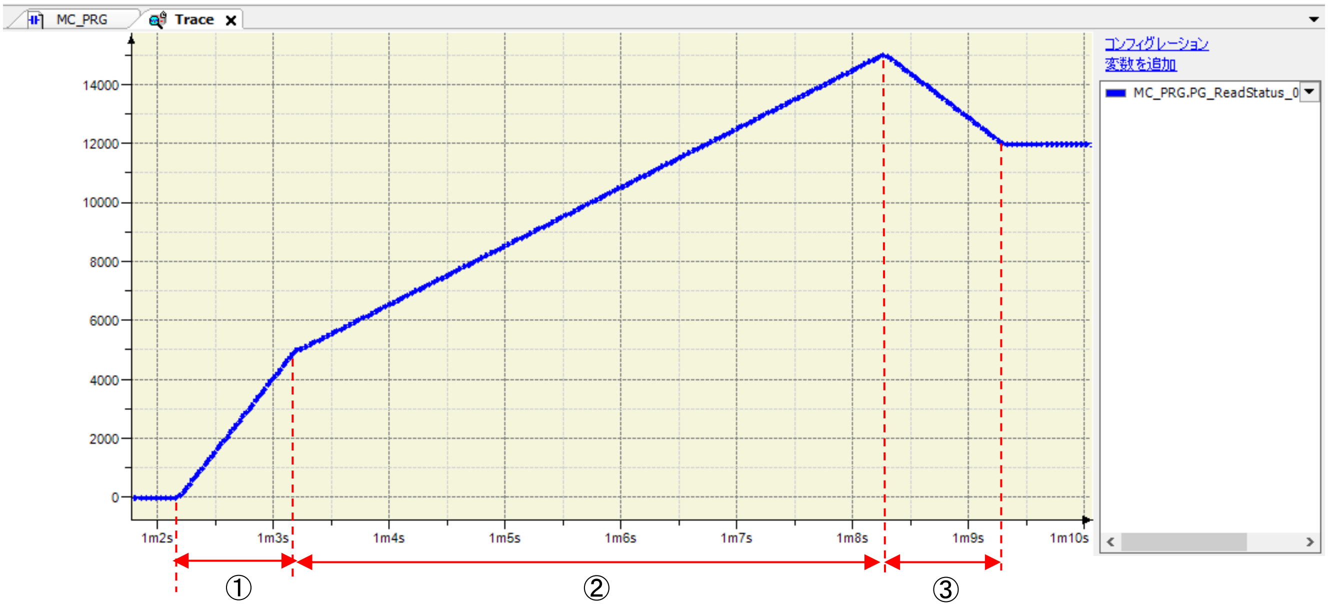
Write <TRUE> to xMove registered in Watch 1.
Check that MC_PRG.PG_ReadStatus_0.AbsoluteCounter registered in Watch 1 during positioning operation contains the current value.



式	アプリケーション	タイプ	値	設定済みの値	実行点	アドレス	コメント
MC_PRG.xPowerOn	Device.Application	BOOL	TRUE		サイクリックモニタリング		
MC_PRG.xHomeOn	Device.Application	BOOL	TRUE		サイクリックモニタリング		
MC_PRG.xMoveOn	Device.Application	BOOL	TRUE		サイクリックモニタリング		
MC_PRG.PG_ReadStatus_0.AbsoluteCounter	Device.Application	LREAL	9504		サイクリックモニタリング	Absolu	

Step 12

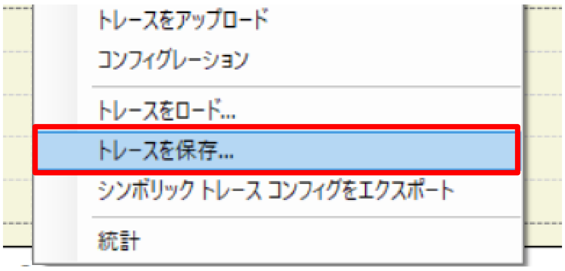
After positioning operation is complete, right-click in the **Trace** window and select **Stop Trace**.



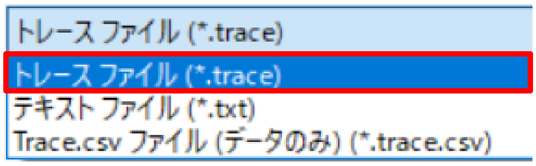
Operation ①: The object moves from the home position to target position ① (5000) (Absolute positioning)
Operation ②: The object moves from target position ① (5000) to target position ② (15000) (Relative positioning)
Operation ③: The object moves from target position ② (15000) to target position ③ (12000) (Relative positioning)

INFO

By right-clicking in the **Trace** window and selecting **Save Trace**, you can save data in XML, text, or CSV format.



Selecting **Trace file (.trace)** saves the trace setting environment and data without making any changes.
This function is convenient when you perform operation checks on devices in remote locations, for example.

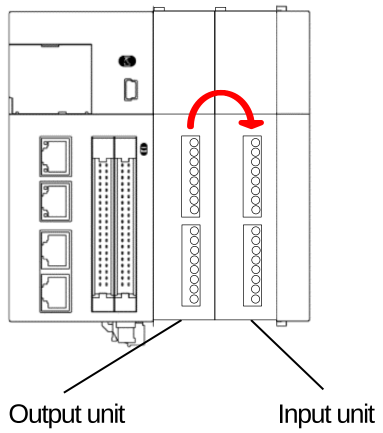

トレース ファイル (*.trace)
トレース ファイル (*.trace)
テキスト ファイル (*.txt)
Trace.csv ファイル (データのみ) (*.trace.csv)

Analog I/O Unit

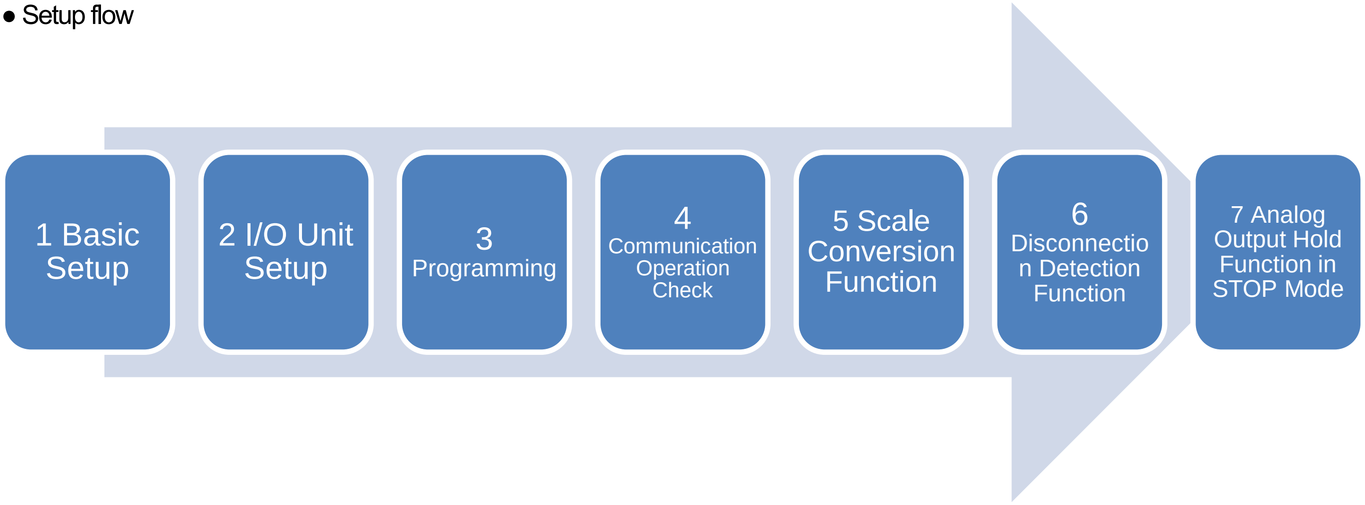
Installation Overview

- Operation images

  In this textbook, the analog values output from the GM1 analog output unit are imported into the GM1 input unit.
  CH0 in each unit uses voltage output and voltage input.



- Setup flow



1 Basic Setup

2 I/O Unit Setup

3 Programming

4 Communication Operation Check

5 Scale Conversion Function

6 Disconnection Detection Function

7 Analog Output Hold Function in STOP Mode

● Unit types

Name	Specifications		Product number
GM1 Analog Input Unit	Input 8ch	Voltage input/output range - 10 to +10 VDC (Resolution: 1/64,000) 0 to +10 VDC (Resolution: 1/32,000) -5 to +5 VDC (Resolution: 1/64,000) 0 to +5 VDC (Resolution: 1/32,000) +1 to +5 VDC (Resolution: 1/25,600) Voltage input/output range 0 to +20 mA (Resolution: 1/32,000) +4 to +20 mA (Resolution: 1/25,600)	AGM1AD8
GM1 Analog Output Unit	Output 4ch		AGM1DA4

INFO

■ Optional functions

Analog input unit

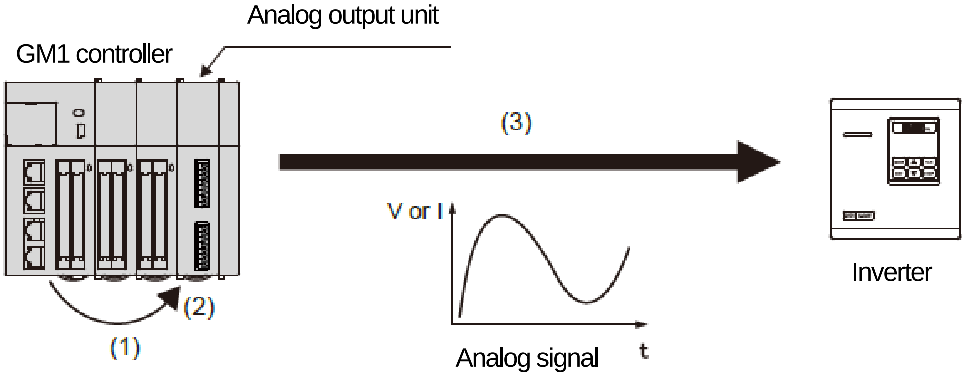
Function	Specifications
Average processing settings function	Selects average processing for the analog values captured by sampling
Offset / Gain processing function	Adjusts and corrects offset values (addition correction) or gain values (magnification correction)
Scale conversion function	Converts the scale of converted values to an easy-to-handle data range
Upper limit / lower limit comparison function	Compares acquired analog input data with the preset upper limit and lower limit values
Max./Min. hold function	Holds the maximum and minimum values of acquired data
Disconnection detection function	Sets the disconnection detection status to ON to warn of an error state when input is disconnected or unconnected

Analog output unit

Function	Specifications
Offset / Gain processing function	Adjusts and corrects offset values (addition correction) or gain values (magnification correction)
Scale conversion function	Sets the analog output range to any easy-to-handle range
Clipping function	Sets the upper limit and lower limit to analog output values in advance
Analog output hold function in STOP mode	Holds the analog output when the operation mode of the GM1 controller changes from RUN to STOP

Analog output unit

- Basic analog output operations



(1) Writing digital data

A user program is used to write digital data to the variable that corresponds to the "Ch*_OutputValue" channel in the analog output unit. The analog signal to be converted changes with the setting of the range.

(2) Analog conversion processing

The data written to the variable is automatically converted to analog signals sequentially inside the unit.

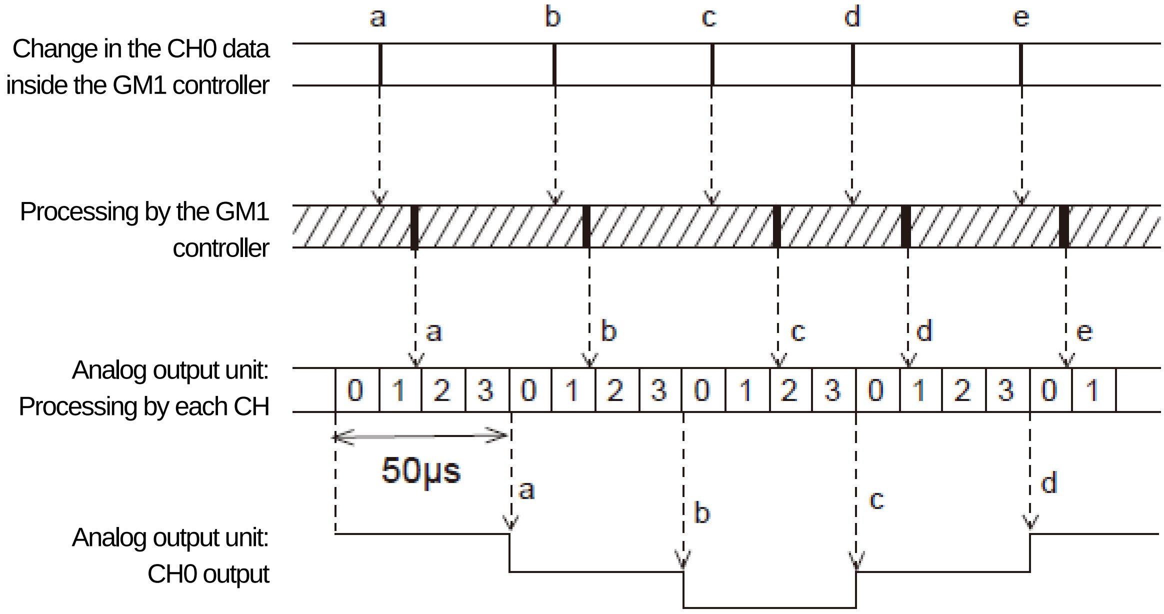
(3) Output to analog-driven device

Converted analog signals are output to analog-driven devices such as inverters.

- Timing chart of output processing

･The "Ch0 output value" written to the GM1 controller is imported into the analog output unit at the time of I/O refresh.

･The analog output unit converts the latest data imported from the GM1 controller into analog values and outputs the converted values.

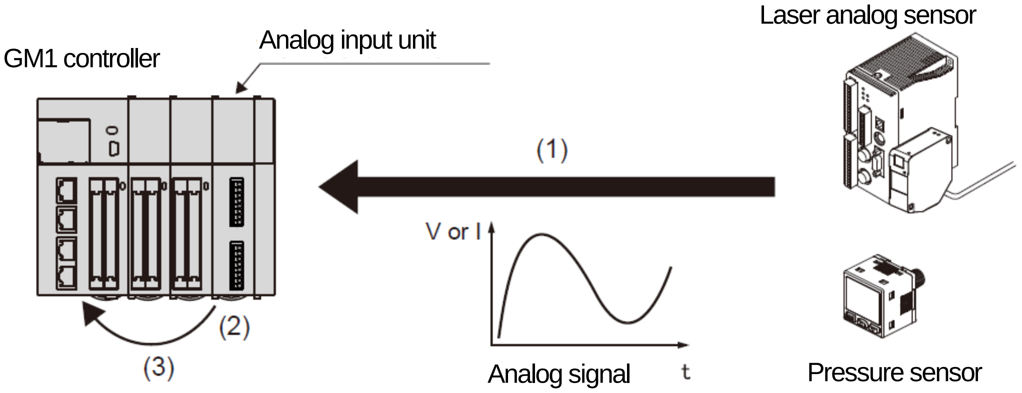

a
b
c
d
e
Change in the CH0 data inside the GM1 controller
Processing by the GM1 controller
a
b
c
d
e
Analog output unit: Processing by each CH
0 1 2 3 0 1 2 3 0 1 2 3 0 1 2 3 0 1
50μs
a
b
c
d
Analog output unit: CH0 output

(Note 1) The processing time of the analog output unit is 50 µs, regardless of the number of valid channels.

(Note 2) The unit outputs converted values when all processing is completed for CH0 to CH3.

Analog input unit

● Basic analog input operations



(1) Importing analog input signals

Analog input signals from analog devices such as laser analog sensors and pressure sensors are imported into the input section of the analog input unit.

(2) Digital conversion processing

Analog input signals imported into the unit are automatically converted to digital values sequentially inside the unit.

(3) Storing digital values

The converted digital values are stored in the channel next to "Analog_8IN I/O Mapping". The asterisk "*" below represents the channel number. (0 to 7)

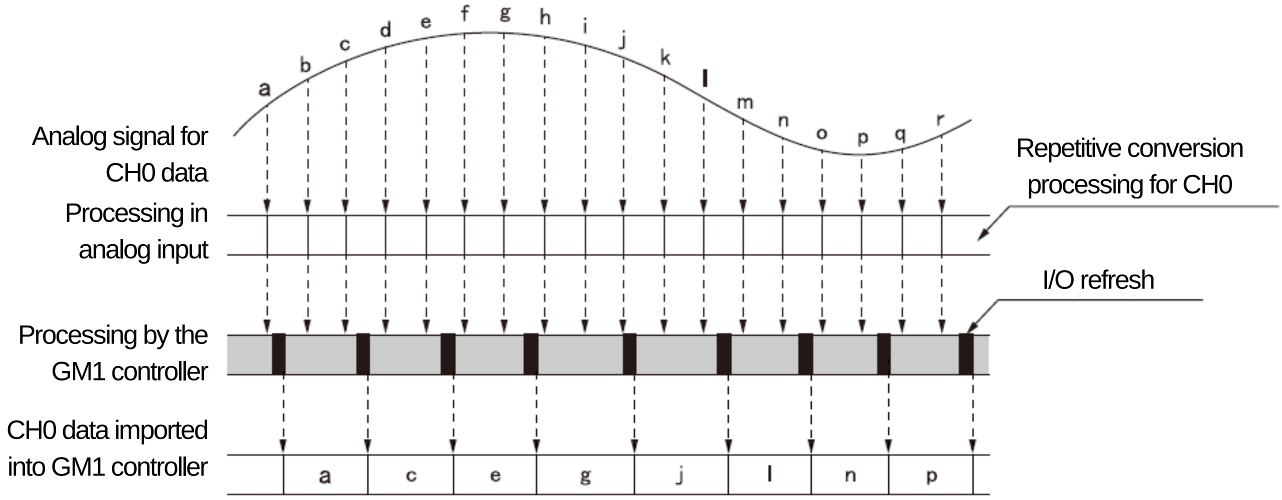
Channel	
Ch*_InputValue	Stores a value before Offset / Gain processing and scale conversion
Ch*_ConversionValue	Stores a value after Offset / Gain processing and scale conversion

● Timing chart of input processing

･The data converted by the analog input unit is imported into the GM1 controller at the time of I/O refresh.

･ When the GM1 controller performs an I/O refresh, the latest data is written to the "Ch0 input value" in the GM1 controller.

For one channel worth of data (Conversion processing time: 50 µs)


a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q
r
Analog signal for CH0 data
Processing in analog input
Repetitive conversion processing for CH0
I/O refresh
Processing by the GM1 controller
CH0 data imported into GM1 controller
a
c
e
g
j
l
n
p

INFO

The conversion processing time differs according to the settings of "Unit-specific configuration setting" for the Analog_8IN parameter.

Select **Not Execute** or **Execute** for **ConversionProcess** on a channel-by-channel basis.

This can save the conversion time for channels that do not execute conversion processing. The conversion time per channel is 50 µs.

Example: Conversion time for one channel (when channels other than CH0 are set to "Disable")

Data for only Ch0 is converted repeatedly. 1 cycle = 50 µs

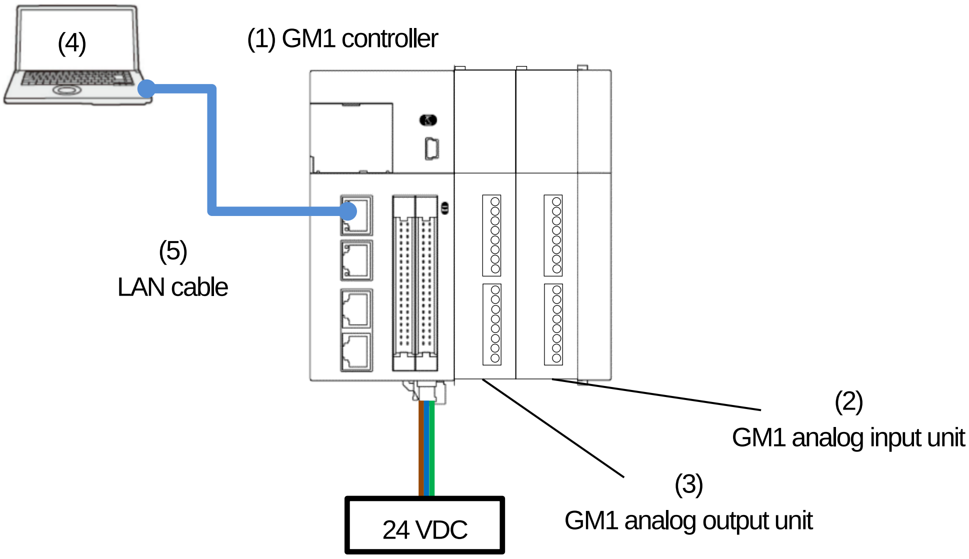
1 Basic Setup

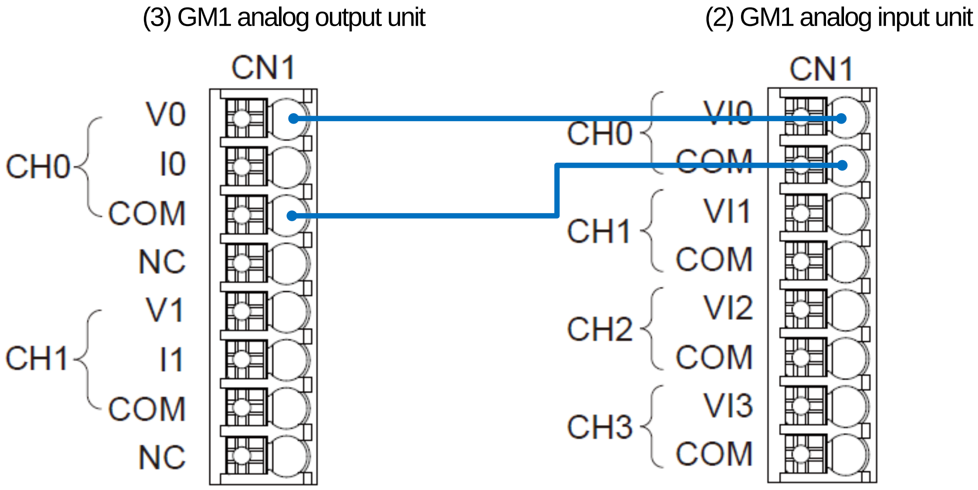
1.1 Preparing and Wiring the Required Devices

Prepare the following devices.

No.	Name
(1)	GM1 controller x 1 (RTEX type)
(2)	GM1 analog input unit
(3)	GM1 analog output unit
(4)	PC (with GM Programmer installed)
(5)	LAN cable: x 1

Wire each device as shown below.





The terminals used when the output range is set to voltage output are different from those used when it is set to current output.

- For voltage output: Vn
- For current output: In

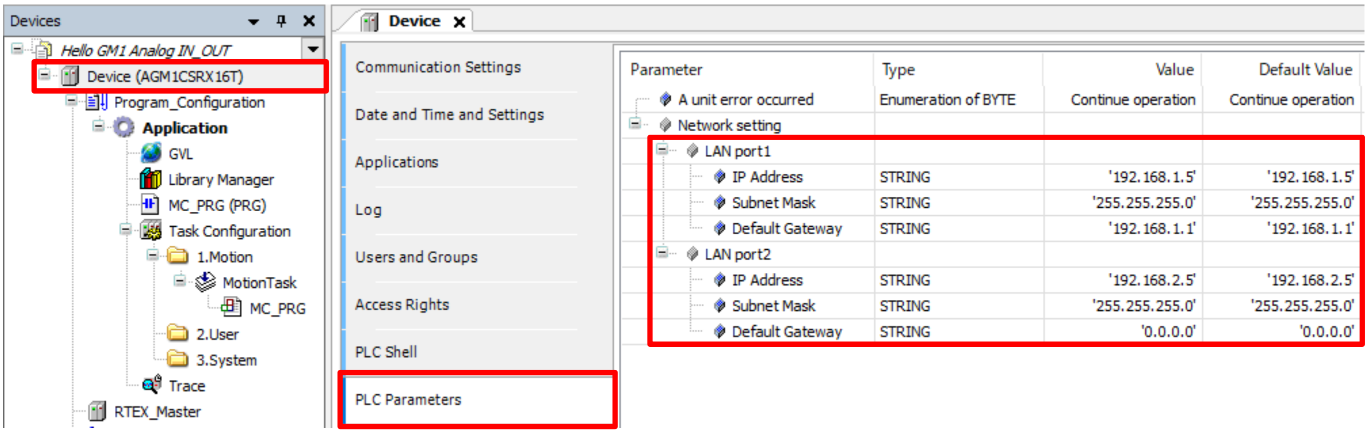
"n" indicates a channel number. (0 to 3)

Common terminals are used for analog voltage input and analog current input.

1.2 IP Address Setting to Network Scanning

Step 1

Open GM Programmer and double-click **Device**.
Select **PLC Parameters** and check the IP address of LAN port 1.



Devices

- Hello GM1 Analog IN_OUT
  - Device (AGM1CSRX16T)
    - Program_Configuration
      - Application
        - GVL
        - Library Manager
        - MC_PRG (PRG)
        - Task Configuration
          - 1.Motion
            - MotionTask
              - MC_PRG
          - 2.User
          - 3.System
        - Trace
    - RTEX_Master

Device

- Communication Settings
- Date and Time and Settings
- Applications
- Log
- Users and Groups
- Access Rights
- PLC Shell
- PLC Parameters

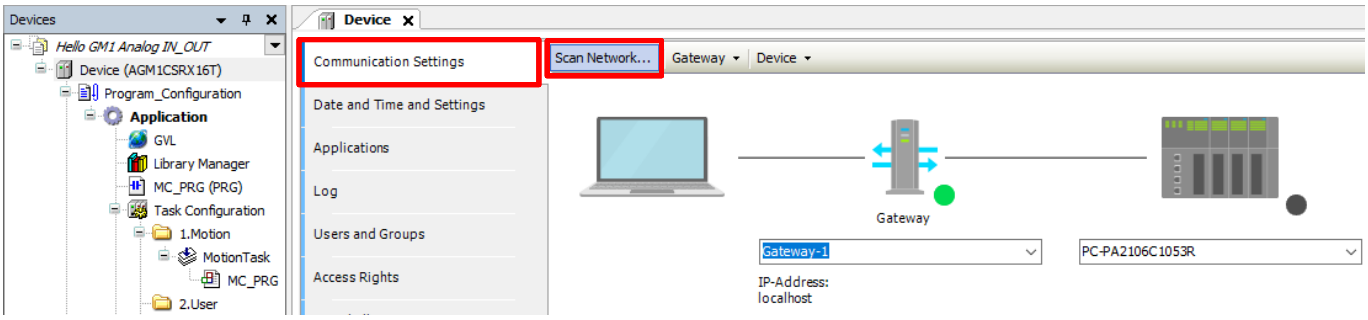
Parameter	Type	Value	Default Value
A unit error occurred	Enumeration of BYTE	Continue operation	Continue operation
Network setting			
LAN port1			
IP Address	STRING	'192.168.1.5'	'192.168.1.5'
Subnet Mask	STRING	'255.255.255.0'	'255.255.255.0'
Default Gateway	STRING	'192.168.1.1'	'192.168.1.1'
LAN port2			
IP Address	STRING	'192.168.2.5'	'192.168.2.5'
Subnet Mask	STRING	'255.255.255.0'	'255.255.255.0'
Default Gateway	STRING	'0.0.0.0'	'0.0.0.0'

LAN port 1 (default value)

IP address	192.168.1.5
Subnet mask	255.255.255.0
Default gateway	192.168.1.1

Step 2

Select **Communication Settings** and click **Scan Network**.



Devices

- Hello GM1 Analog IN_OUT
  - Device (AGM1CSRX16T)
    - Program_Configuration
      - Application
        - GVL
        - Library Manager
        - MC_PRG (PRG)
        - Task Configuration
          - 1.Motion
            - MotionTask
              - MC_PRG
          - 2.User

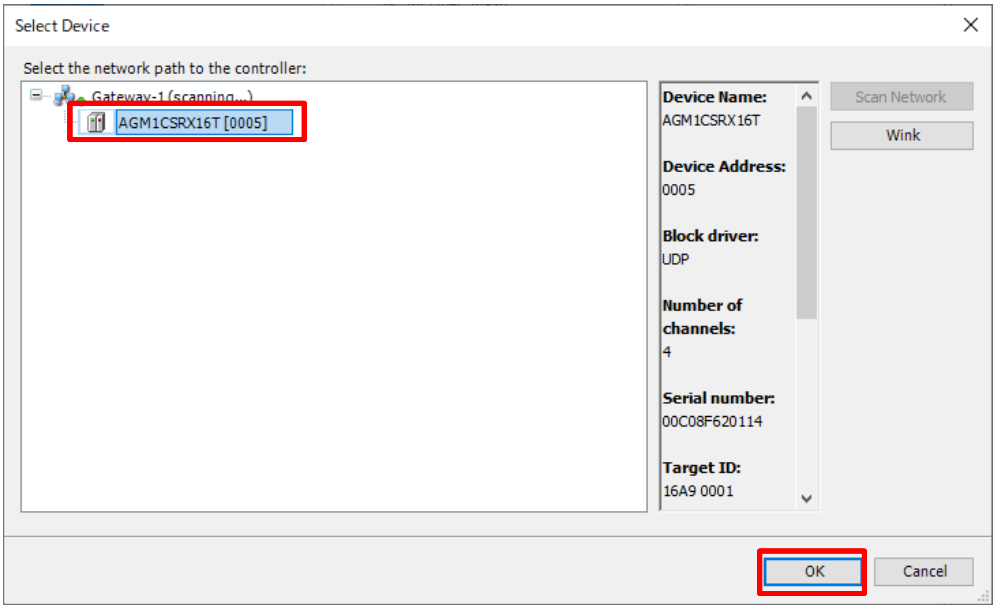
Device

- Communication Settings
- Date and Time and Settings
- Applications
- Log
- Users and Groups
- Access Rights

IP-Address:
localhost

Step 3

Select a device to be connected and click **OK**.



Select Device

Select the network path to the controller:

- Gateway-1 (scanning...)
  - AGM1CSRX16T [0005]

Device Name:
AGM1CSRX16T

Device Address:
0005

Block driver:
UDP

Number of channels:
4

Serial number:
00C08F620114

Target ID:
16A9 0001

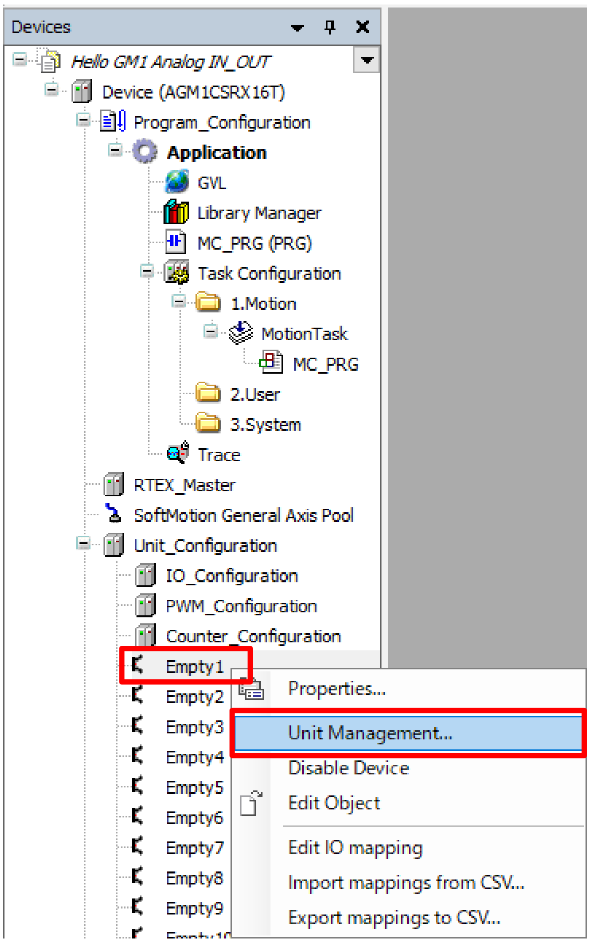
2 I/O Unit Setup

2.1 Adding I/O Units

First, add a device object for the I/O unit to the project.

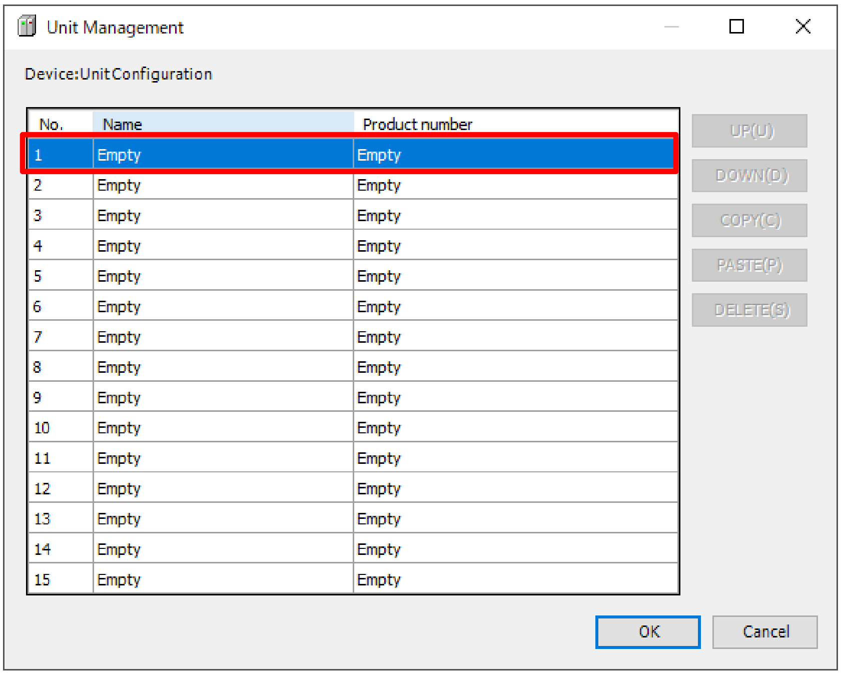
Step 1

Select **Device** and then **Unit_Configuration**, right-click **Empty1**, and select **Unit Management**.



Step 2

The **Unit Management** dialog box will be displayed.
Double-click on the No. 1 row.

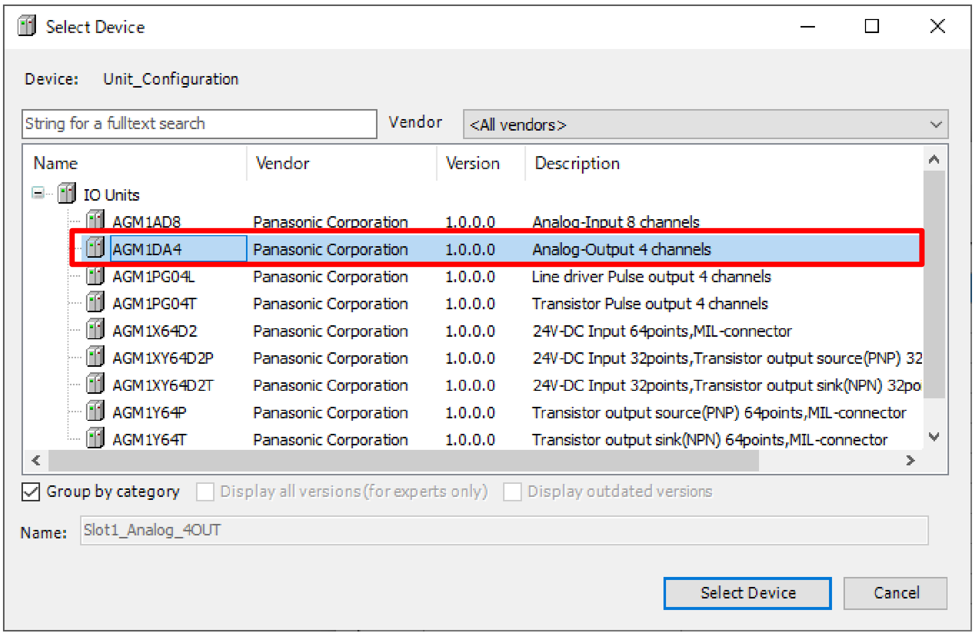


Device:UnitConfiguration

No.	Name	Product number
1	Empty	Empty
2	Empty	Empty
3	Empty	Empty
4	Empty	Empty
5	Empty	Empty
6	Empty	Empty
7	Empty	Empty
8	Empty	Empty
9	Empty	Empty
10	Empty	Empty
11	Empty	Empty
12	Empty	Empty
13	Empty	Empty
14	Empty	Empty
15	Empty	Empty

Step 3

The **Select Device** dialog box will be displayed. Select an expansion unit to be added.
In this textbook, the GM1 controller, analog output unit (AGM1DA4), and analog input unit (AGM1AD8) are arranged in this order from left to right. For this reason, register AGM1DA4 in the No. 1 row.



Device: Unit_Configuration

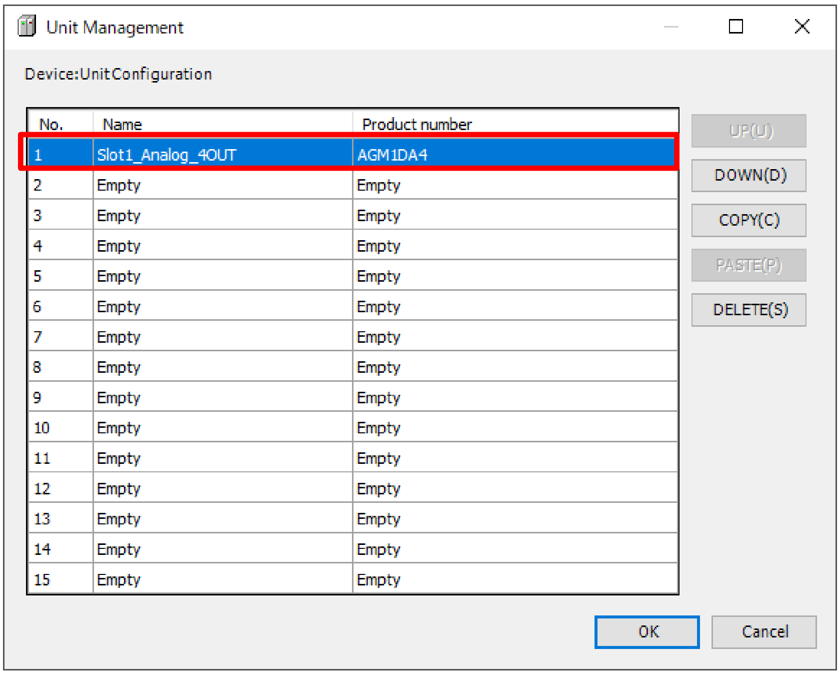
Vendor <All vendors>

Name	Vendor	Version	Description
IO Units			
AGM1AD8	Panasonic Corporation	1.0.0.0	Analog-Input 8 channels
AGM1DA4	Panasonic Corporation	1.0.0.0	Analog-Output 4 channels
AGM1PG04L	Panasonic Corporation	1.0.0.0	Line driver Pulse output 4 channels
AGM1PG04T	Panasonic Corporation	1.0.0.0	Transistor Pulse output 4 channels
AGM1X64D2	Panasonic Corporation	1.0.0.0	24V-DC Input 64points,MIL-connector
AGM1XY64D2P	Panasonic Corporation	1.0.0.0	24V-DC Input 32points,Transistor output source(PNP) 32
AGM1XY64D2T	Panasonic Corporation	1.0.0.0	24V-DC Input 32points,Transistor output sink(NPN) 32po
AGM1Y64P	Panasonic Corporation	1.0.0.0	Transistor output source(PNP) 64points,MIL-connector
AGM1Y64T	Panasonic Corporation	1.0.0.0	Transistor output sink(NPN) 64points,MIL-connector

☑ Group by category □ Display all versions (for experts only) □ Display outdated versions

Name: Slot1_Analog_4OUT

AGM1DA4 has been registered in the No. 1 row, as below.



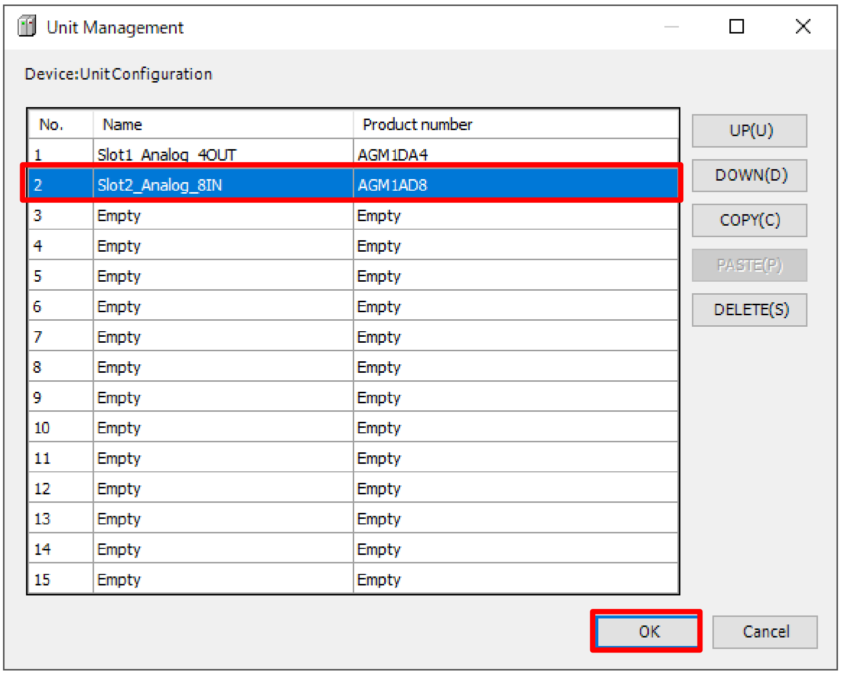
Unit Management

Device:UnitConfiguration

No.	Name	Product number
1	Slot1_Analog_4OUT	AGM1DA4
2	Empty	Empty
3	Empty	Empty
4	Empty	Empty
5	Empty	Empty
6	Empty	Empty
7	Empty	Empty
8	Empty	Empty
9	Empty	Empty
10	Empty	Empty
11	Empty	Empty
12	Empty	Empty
13	Empty	Empty
14	Empty	Empty
15	Empty	Empty

Step 4

Register AGM1AD8 in the No. 2 row in the same way as above.
When registration is completed, click **OK**.

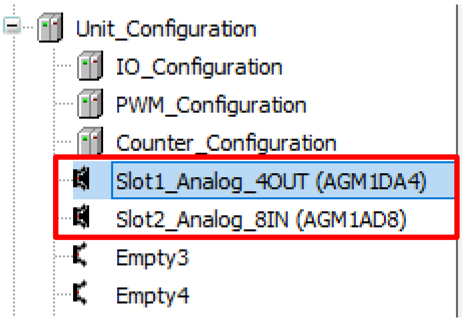


Unit Management

Device:UnitConfiguration

No.	Name	Product number
1	Slot1_Analog_4OUT	AGM1DA4
2	Slot2_Analog_8IN	AGM1AD8
3	Empty	Empty
4	Empty	Empty
5	Empty	Empty
6	Empty	Empty
7	Empty	Empty
8	Empty	Empty
9	Empty	Empty
10	Empty	Empty
11	Empty	Empty
12	Empty	Empty
13	Empty	Empty
14	Empty	Empty
15	Empty	Empty

Check that AGM1DA4 and AGM1AD8 have been registered as shown in the figure to the right.
Empty1 → Slot1_Analog_4OUT (AGM1DA4)
Empty2 → Slot2_Analog_8IN (AGM1AD8)

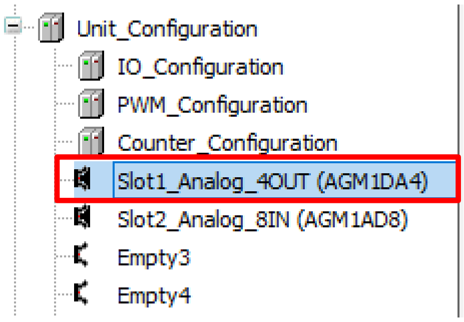


2.2 Setting up the Analog Output Unit (AGM1DA4)

Set up parameters for the analog output unit and register I/O mapping.

Step 1

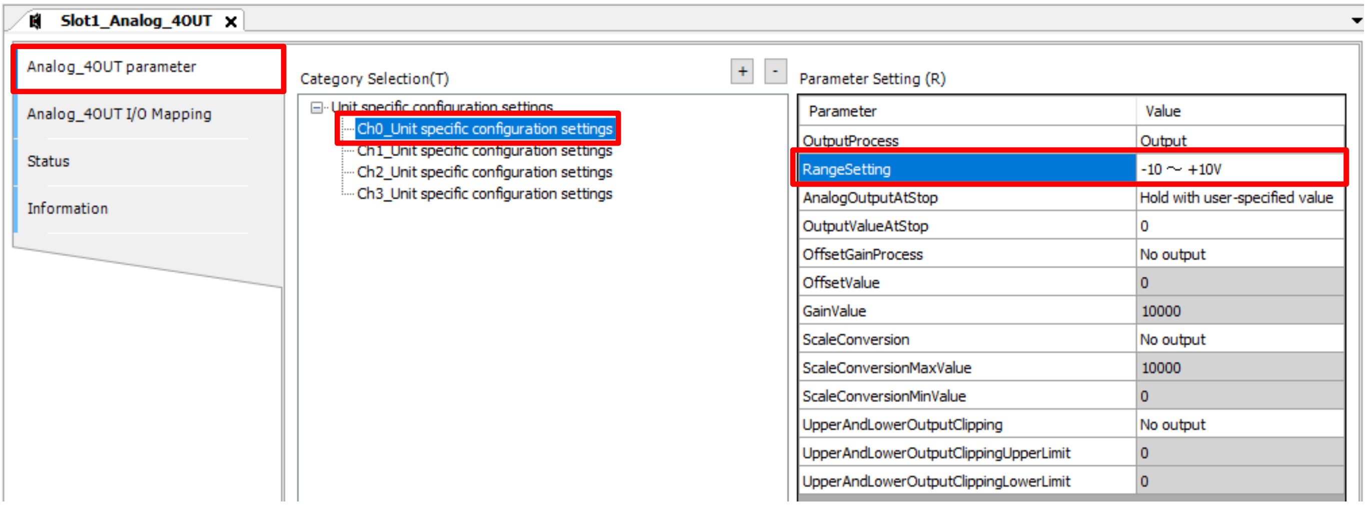
Double-click **Slot1_Analog_4OUT (AGM1DA4)**, which has been registered previously.



- Unit_Configuration
  - IO_Configuration
  - PWM_Configuration
  - Counter_Configuration
  - Slot1_Analog_4OUT (AGM1DA4)
  - Slot2_Analog_8IN (AGM1AD8)
  - Empty3
  - Empty4

Step 2

Select **Analog_4OUT Parameter** and then **Ch0_Unit specific configuration settings**.
In **Parameter Setting (R)**, select **-10 ~ +10V** (default value) in **RangeSetting**.

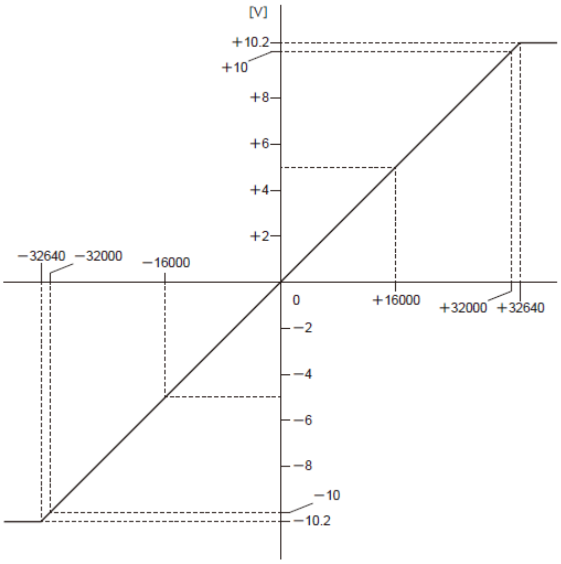


Slot1_Analog_4OUT
Analog_4OUT parameter
Analog_4OUT I/O Mapping
Status
Information
Category Selection(T)
+
-
Unit specific configuration settings
Ch0_Unit specific configuration settings
Ch1_Unit specific configuration settings
Ch2_Unit specific configuration settings
Ch3_Unit specific configuration settings
Parameter Setting (R)

Parameter	Value
OutputProcess	Output
RangeSetting	-10 ~ +10V
AnalogOutputAtStop	Hold with user-specified value
OutputValueAtStop	0
OffsetGainProcess	No output
OffsetValue	0
GainValue	10000
ScaleConversion	No output
ScaleConversionMaxValue	10000
ScaleConversionMinValue	0
UpperAndLowerOutputClipping	No output
UpperAndLowerOutputClippingUpperLimit	0
UpperAndLowerOutputClippingLowerLimit	0

INFO

Voltage output range: -10 to +10 V (Resolution: 1/64,000)



Digital input value	Analog output value (V)
+32000	+10
+25600	+8
+19200	+6
+12800	+4
+6400	+2
0	0
- 6400	- 2
- 12800	- 4
- 19200	- 6
- 25600	- 8
- 32000	- 10

When the rated range is exceeded

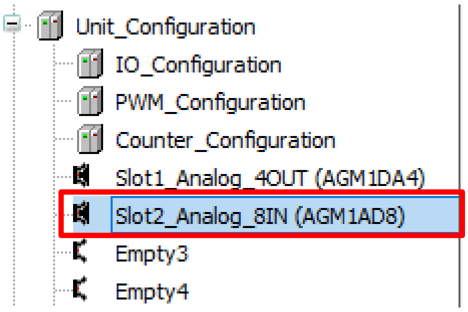
Digital input value	Analog output value (V)
+32640 or more	+10.2
-32640 or less	- 10.2

2.3 Setting up the Analog Input Unit (AGM1AD8)

Set up parameters for the analog input unit and register I/O mapping.

Step 1

Double-click **Slot2_Analog_8IN (AGM1AD8)**, which has been registered previously.

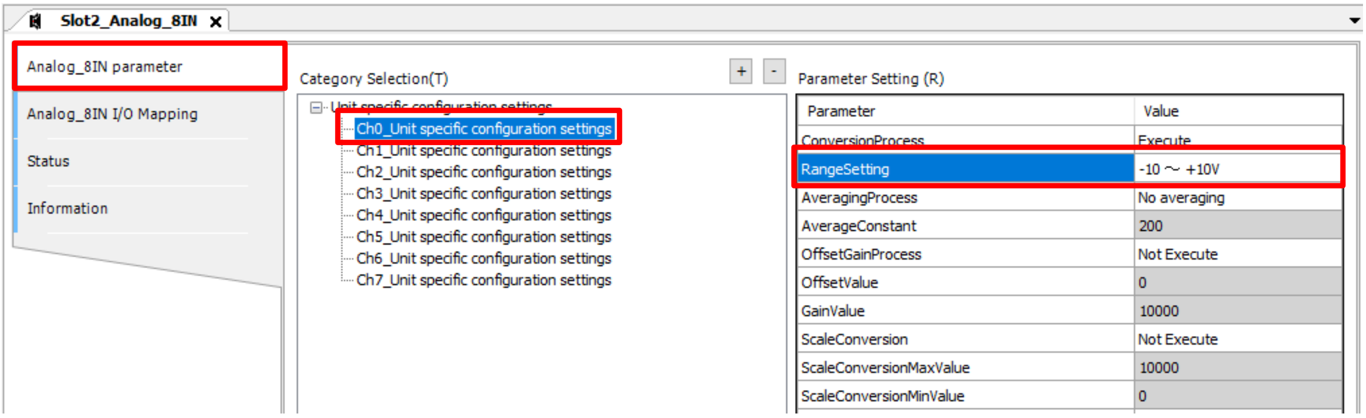


- Unit_Configuration
  - IO_Configuration
  - PWM_Configuration
  - Counter_Configuration
  - Slot1_Analog_4OUT (AGM1DA4)
  - Slot2_Analog_8IN (AGM1AD8)
  - Empty3
  - Empty4

Step 2

Select **Analog_8IN Parameter** and then **Ch0_Unit specific configuration settings**.

In **Parameter Setting (R)**, select **-10 ~ +10V** (default value) in **RangeSetting**.



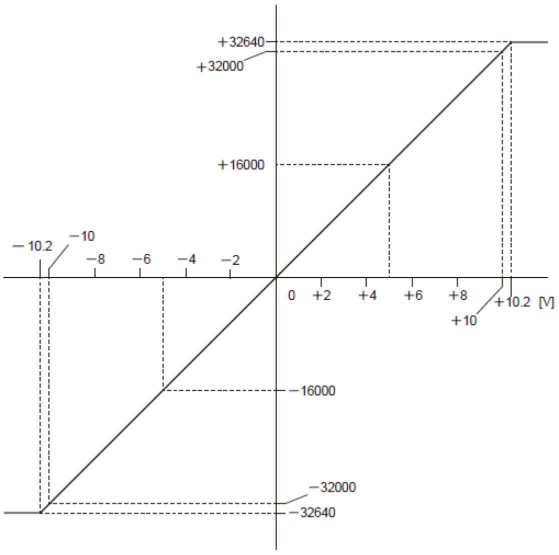
- Unit specific configuration settings
  - Ch0_Unit specific configuration settings
  - Ch1_Unit specific configuration settings
  - Ch2_Unit specific configuration settings
  - Ch3_Unit specific configuration settings
  - Ch4_Unit specific configuration settings
  - Ch5_Unit specific configuration settings
  - Ch6_Unit specific configuration settings
  - Ch7_Unit specific configuration settings

Parameter Setting (R)

Parameter	Value
ConversionProcess	Execute
RangeSetting	-10 ～ +10V
AveragingProcess	No averaging
AverageConstant	200
OffsetGainProcess	Not Execute
OffsetValue	0
GainValue	10000
ScaleConversion	Not Execute
ScaleConversionMaxValue	10000
ScaleConversionMinValue	0

INFO

Voltage output range: -10 to +10 V (Resolution: 1/64,000)



Analog input value (V)	Digital converted value
+10	+32000
+8	+25600
+6	+19200
+4	+12800
+2	+6400
0	0
- 2	- 6400
- 4	- 12800
- 6	- 19200
- 8	- 25600
- 10	- 32000

When the rated range is exceeded

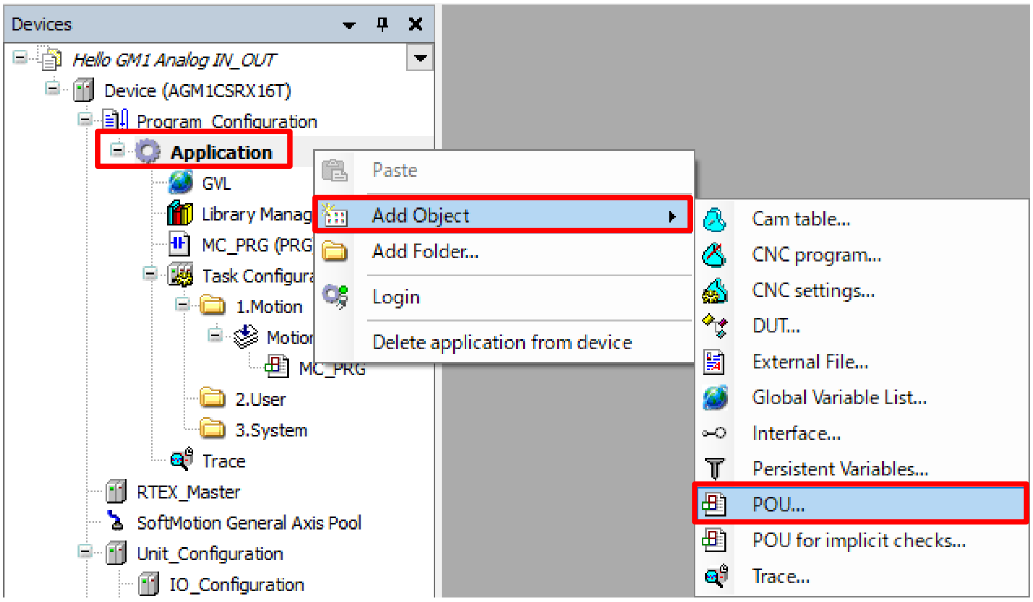
Analog input value	Digital converted value
+10.2 V or more	+32640 or more
- 10.2 V or less	-32640 or less

3 Programming

3.1 Adding New POU

Step 1

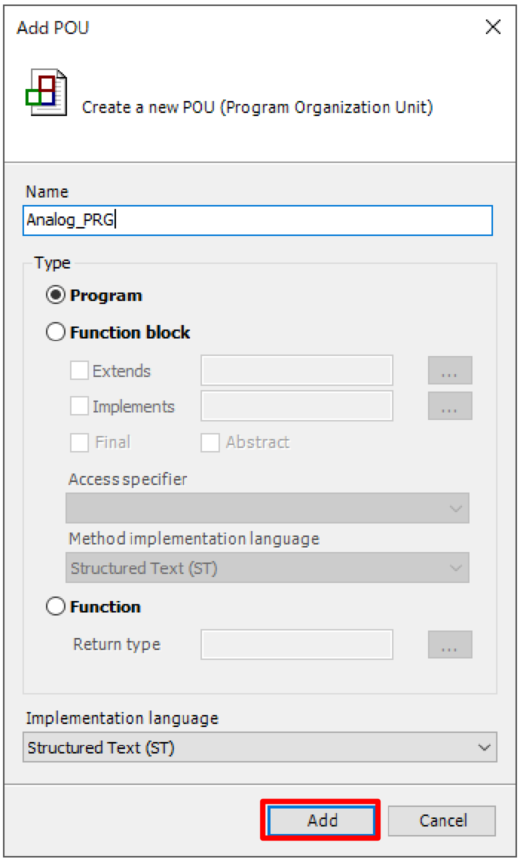
Right-click **Application** and select **Add Object** and then **POU** to create a new POU.



Step 2

In the **Add POU** dialog box, specify settings as below and click **Add**.

Name	Analog_PRG
Type	**Program**
Implementation language	**Structured Text (ST)**


Add POU

Create a new POU (Program Organization Unit)

Name: Analog_PRG

Type

● Program

○ Function block

□ Extends

□ Implements

□ Final □ Abstract

Access specifier

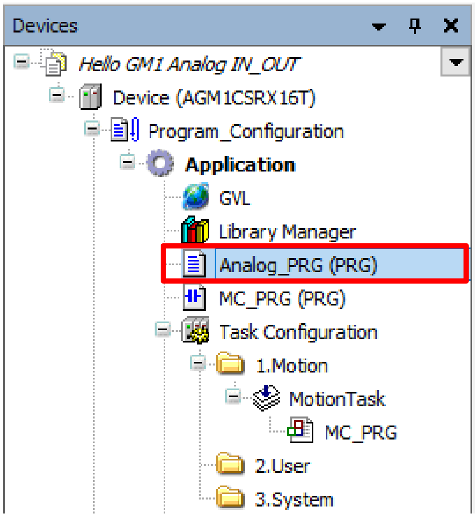
Method implementation language: Structured Text (ST)

○ Function

Return type

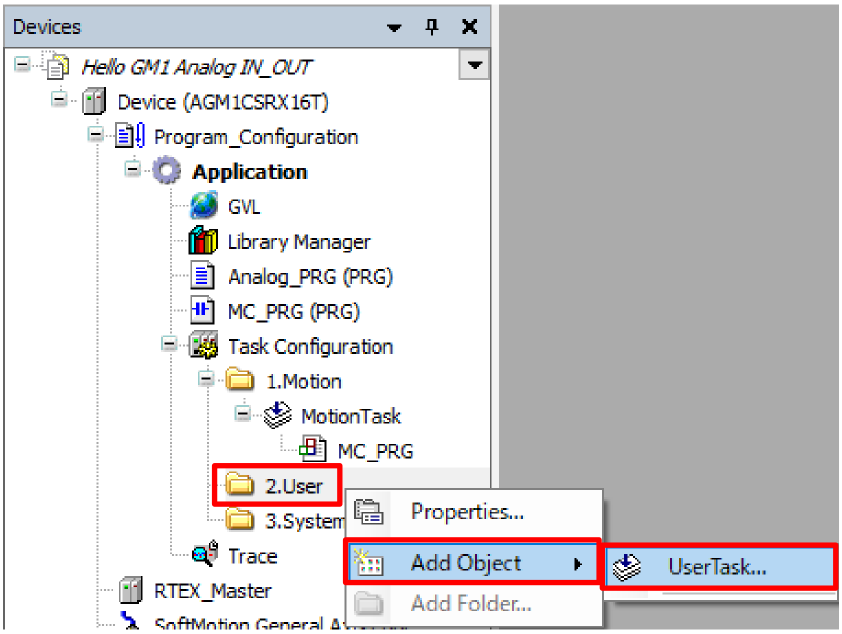
Implementation language: Structured Text (ST)

Analog_PRG (PRG) will be added to **Application**.



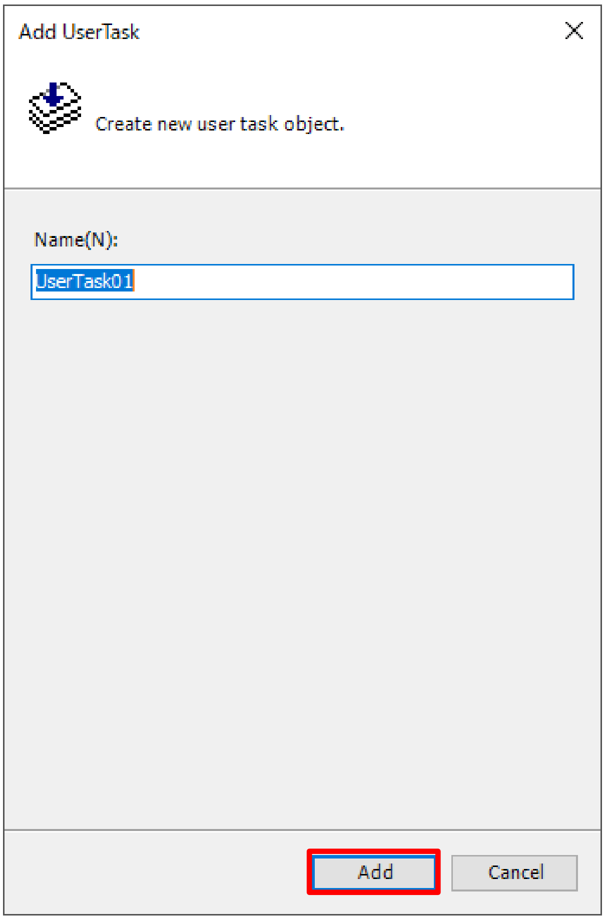
Step 3

Right-click **2.User** and select **Add Object** and then **UserTask**.



Step 4

The **Add UserTask** dialog box will be displayed. Leave **UserTask01** (default) unchanged in the **Name(N)** field and click **Add**.

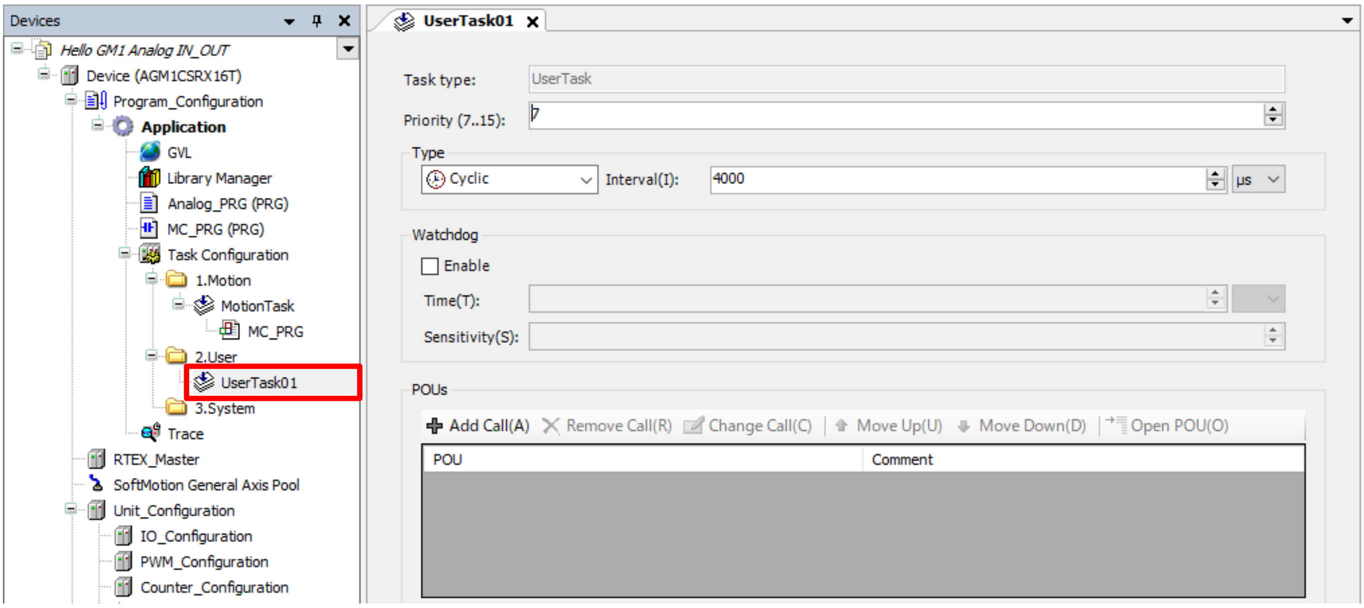


Add UserTask

Create new user task object.

Name(N): UserTask01

UserTask01 will be added.



Task type: UserTask

Priority (7..15): 7

Type: Cyclic Interval(I): 4000 µs

Watchdog

□ Enable

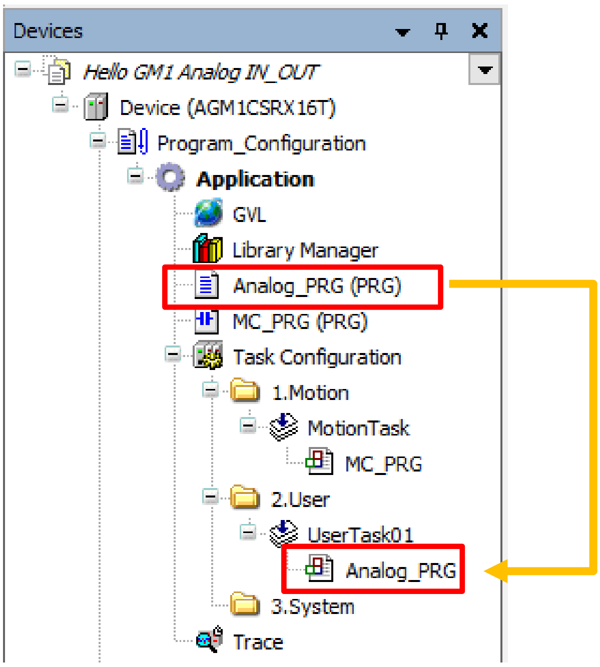
Time(T):

Sensitivity(S):

POUs

Step 5

Drag and drop **Analog_PRG (PRG)** into the **UserTask01** object, which has been added, to add it to the task.



Drag and drop

Column ⑥: Tasks

Task	Description
MotionTask	This is a user program task to perform motion control. It is given the highest priority. Only one MotionTask is allowed for each project.
UserTask	This is a user program task to perform control other than motion control. The user can set the level of priority. Up to 50 tasks can be registered in a single project.
SystemTask	This is a task that is used by the system and cannot be added by user programs. It is processed while other tasks are inactive.

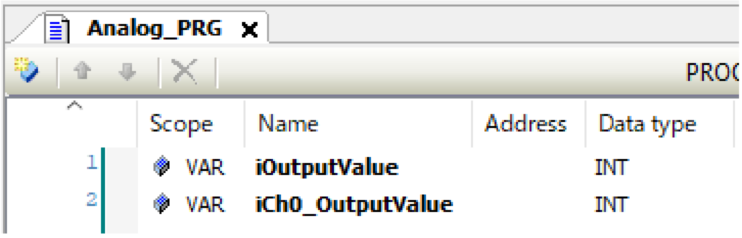
3.2 Programming

First, programming for the analog input unit (AGM1DA4) is explained.

Step 1

Open the **Analog_PRG** tab and add local variables as below.

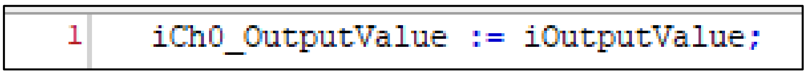
Name	Data type
iOutputValue	INT
iCh0_OutputValue	INT



	Scope	Name	Address	Data type
1	VAR	**iOutputValue**		INT
2	VAR	**iCh0_OutputValue**		INT

Step 2

Using the variables registered in Step 1 above, code a program as below.

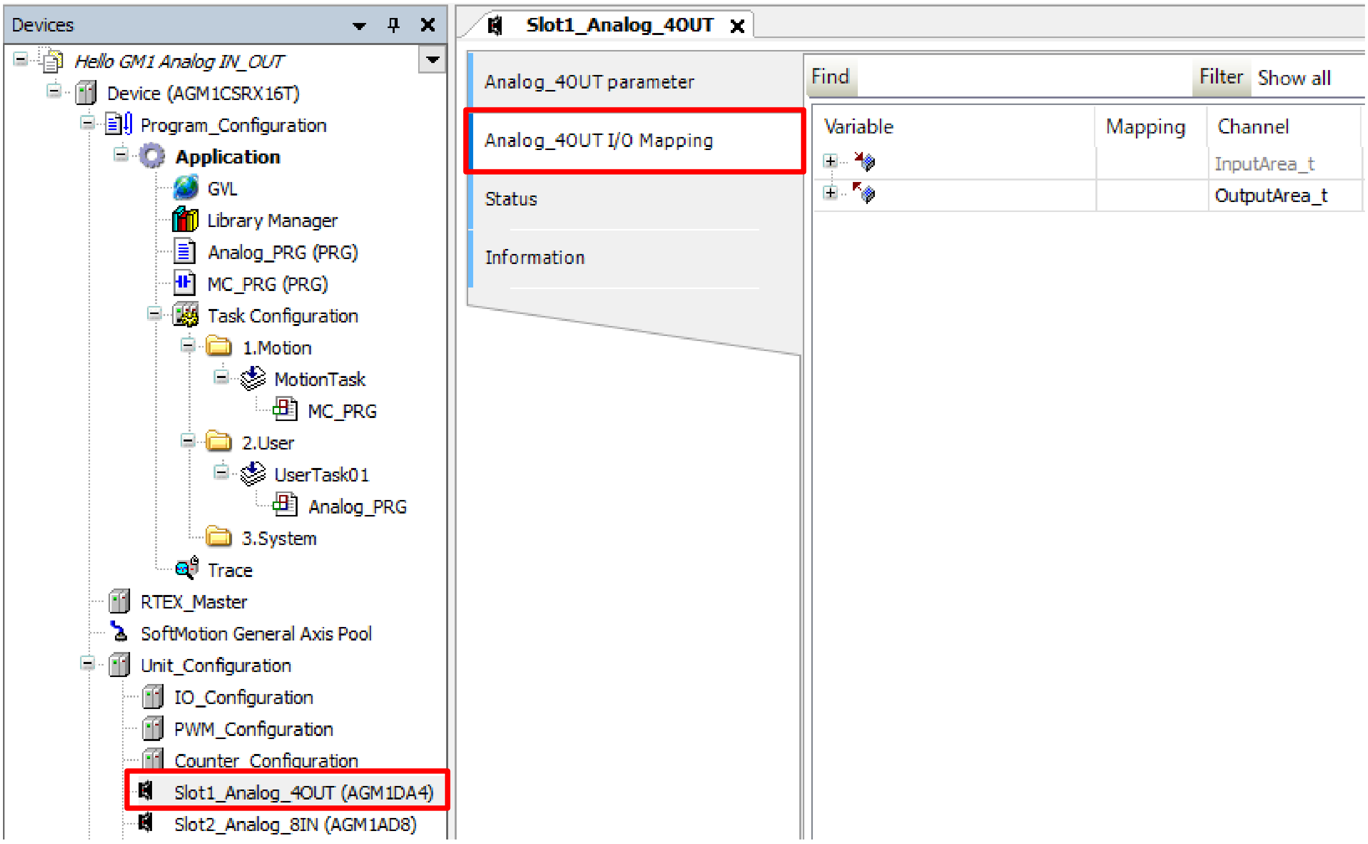


```
iCh0_OutputValue := iOutputValue;
```

Step 3

In **Analog_4OUT I/O Mapping**, link iCh0_OutputValue to the corresponding channel, as below.

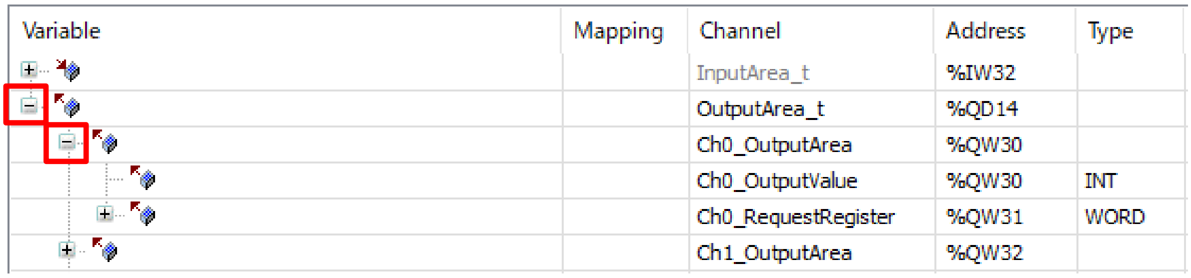
Double-click **Slot1_Analog_4OUT (AGM1DA4)** and open the **Analog_4OUT I/O Mapping** tab.



Variable	Mapping	Channel
		InputArea_t
		OutputArea_t

Step 4

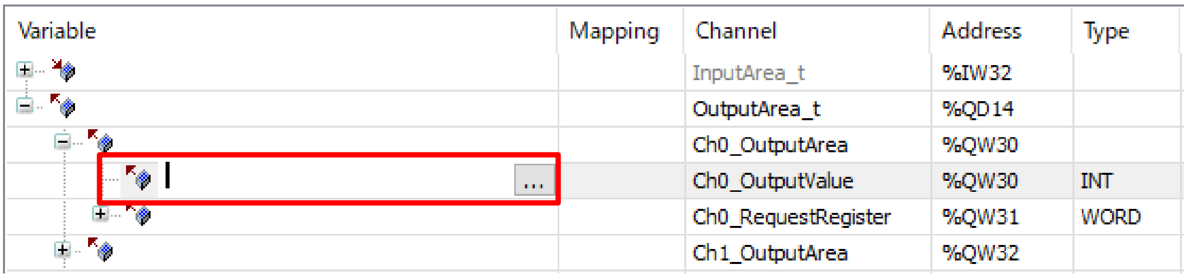
Expand **OutputArea_t** through to **Ch0_OutputArea** by clicking ⊞.



Variable	Mapping	Channel	Address	Type
⊞		InputArea_t	%IW32	
⊟		OutputArea_t	%QD14	
⊟		Ch0_OutputArea	%QW30	
		Ch0_OutputValue	%QW30	INT
⊞		Ch0_RequestRegister	%QW31	WORD
⊞		Ch1_OutputArea	%QW32	

Step 5

Double-click in the empty row corresponding to **Ch0_OutputValue** and then click ... .

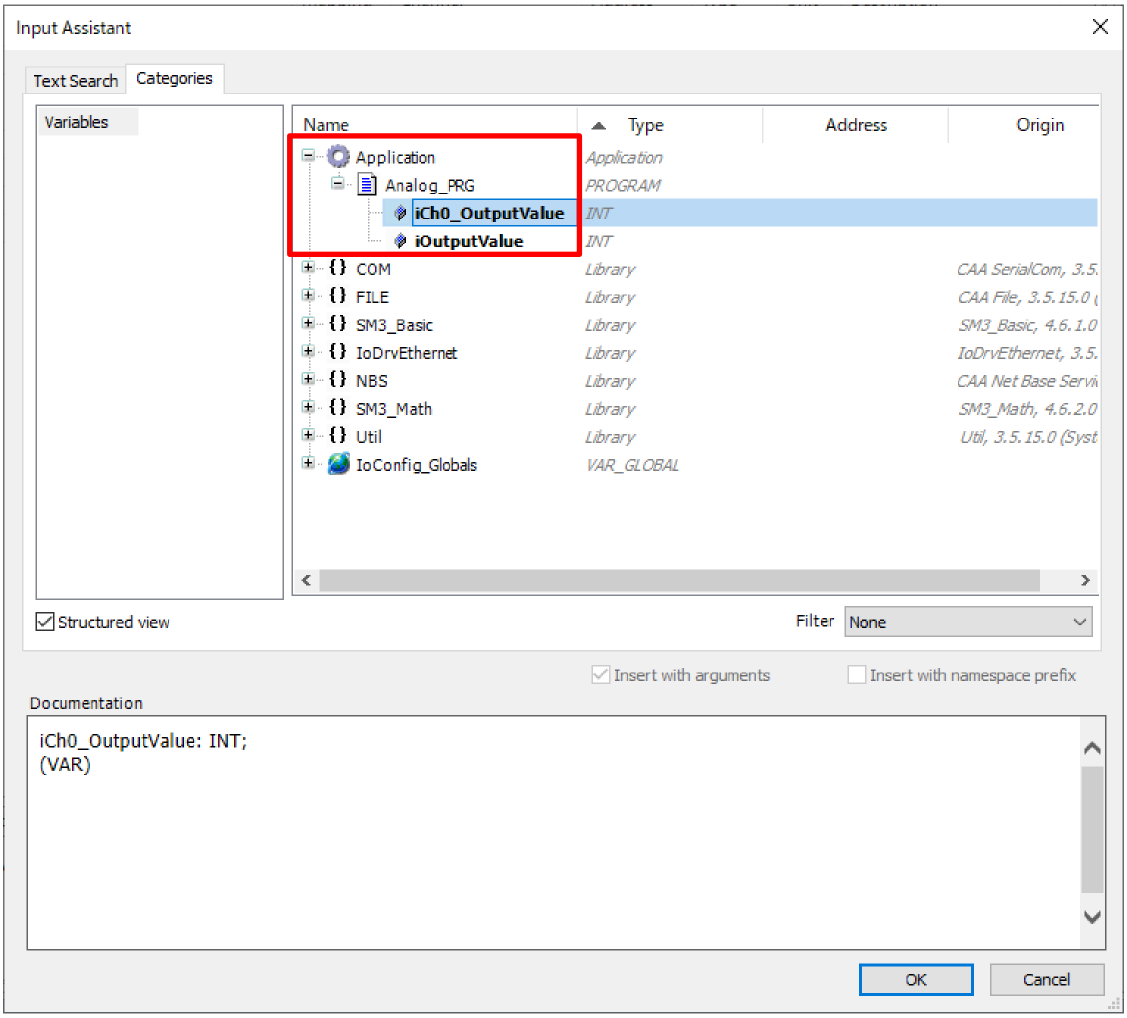


Variable	Mapping	Channel	Address	Type
⊞		InputArea_t	%IW32	
⊟		OutputArea_t	%QD14	
⊟		Ch0_OutputArea	%QW30	
...		Ch0_OutputValue	%QW30	INT
⊞		Ch0_RequestRegister	%QW31	WORD
⊞		Ch1_OutputArea	%QW32	

Step 6

The **Input Assistant** dialog box will be displayed.

Select **Application**, **Analog_PRG**, and then **iCh0_OutputValue**.



Input Assistant

Text Search | Categories

Variables

Name	Type	Address	Origin
⊟ Application	Application		
⊟ Analog_PRG	PROGRAM		
iCh0_OutputValue	INT		
iOutputValue	INT		
⊞ COM	Library		CAA SerialCom, 3.5.
⊞ FILE	Library		CAA File, 3.5.15.0 (
⊞ SM3_Basic	Library		SM3_Basic, 4.6.1.0
⊞ IoDrvEthernet	Library		IoDrvEthernet, 3.5.
⊞ NBS	Library		CAA Net Base Servi
⊞ SM3_Math	Library		SM3_Math, 4.6.2.0
⊞ Util	Library		Util, 3.5.15.0 (Syst
⊞ IoConfig_Globals	VAR_GLOBAL		

☑ Structured view

Filter: None

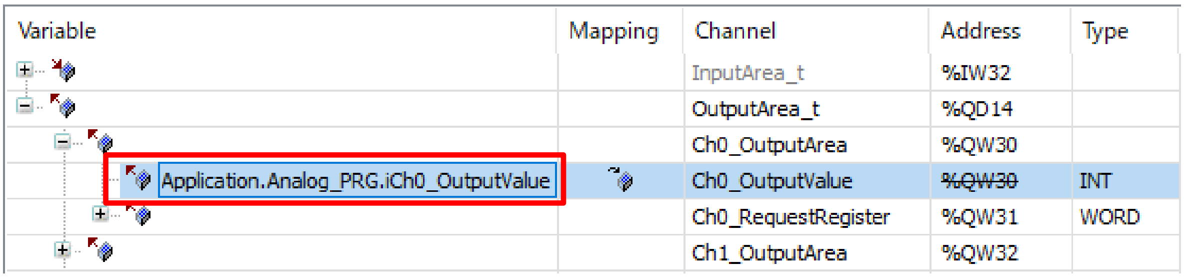
☑ Insert with arguments ☐ Insert with namespace prefix

Documentation

```
iCh0_OutputValue: INT;
(VAR)
```

OK | Cancel

iCh0_OutputValue will be linked to the corresponding channel.



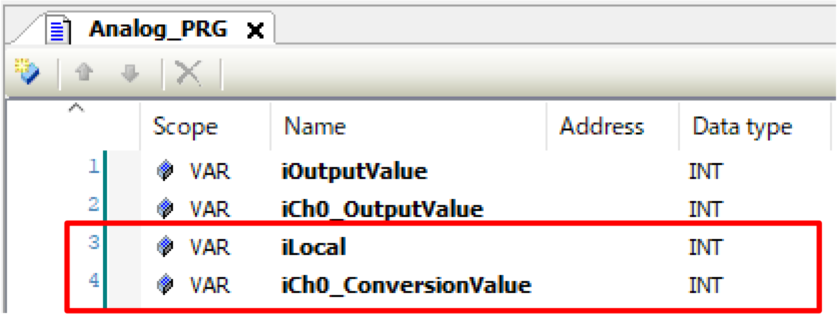
Variable	Mapping	Channel	Address	Type
		InputArea_t	%IW32	
		OutputArea_t	%QD14	
		Ch0_OutputArea	%QW30	
Application.Analog_PRG.iCh0_OutputValue		Ch0_OutputValue	~~%QW30~~	INT
		Ch0_RequestRegister	%QW31	WORD
		Ch1_OutputArea	%QW32	

Next, programming for the analog output unit (AGM1AD8) is explained.

Step 1

Double-click **Analog_PRG** to open the program that has been created previously.
Add variables as below.

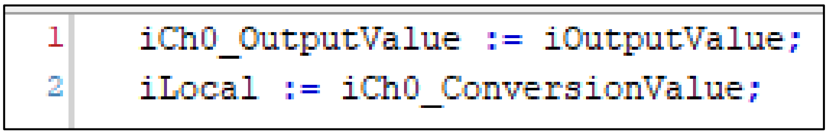
Name	Data type
iLocal	INT
iCh0_ConversionValue	INT



	Scope	Name	Address	Data type
1	VAR	iOutputValue		INT
2	VAR	iCh0_OutputValue		INT
3	VAR	iLocal		INT
4	VAR	iCh0_ConversionValue		INT

Step 2

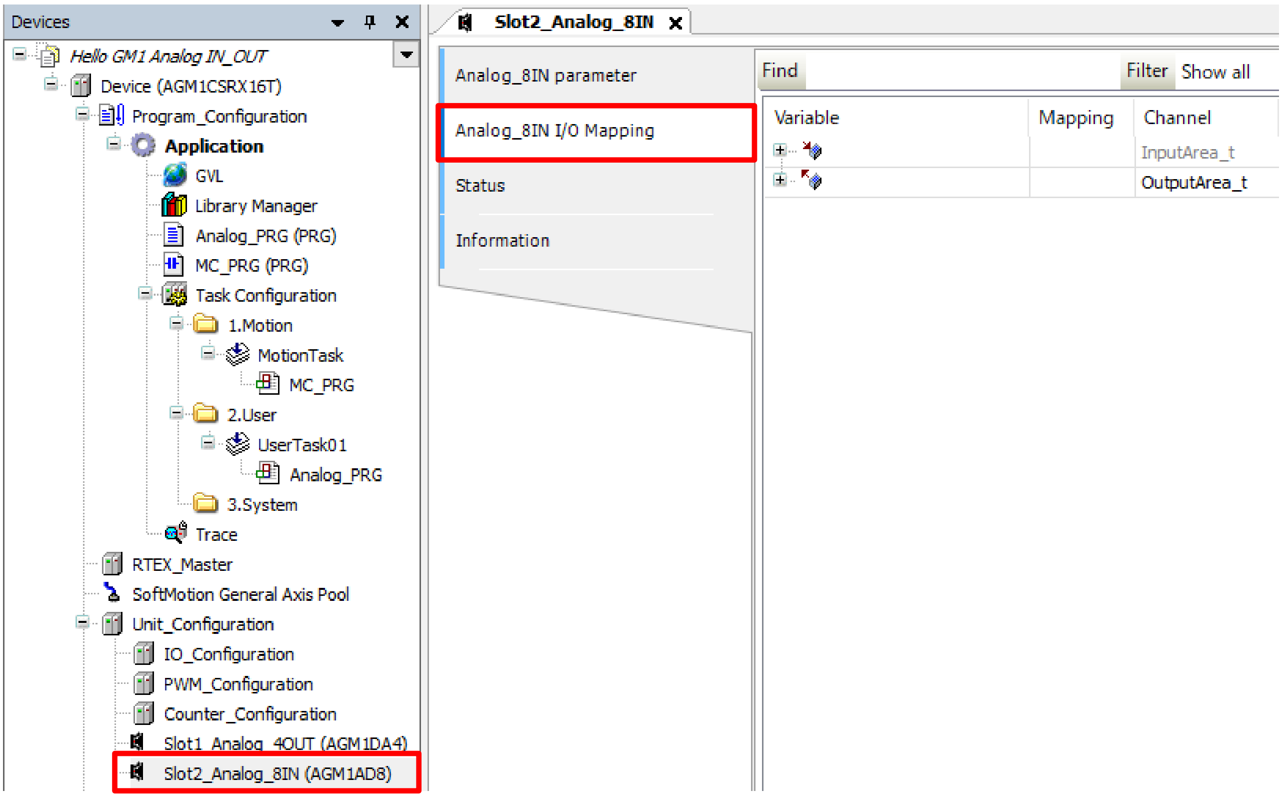
Using the variables registered in Step 1 above, code a program as below.



```
iCh0_OutputValue := iOutputValue;
iLocal := iCh0_ConversionValue;
```

Step 3

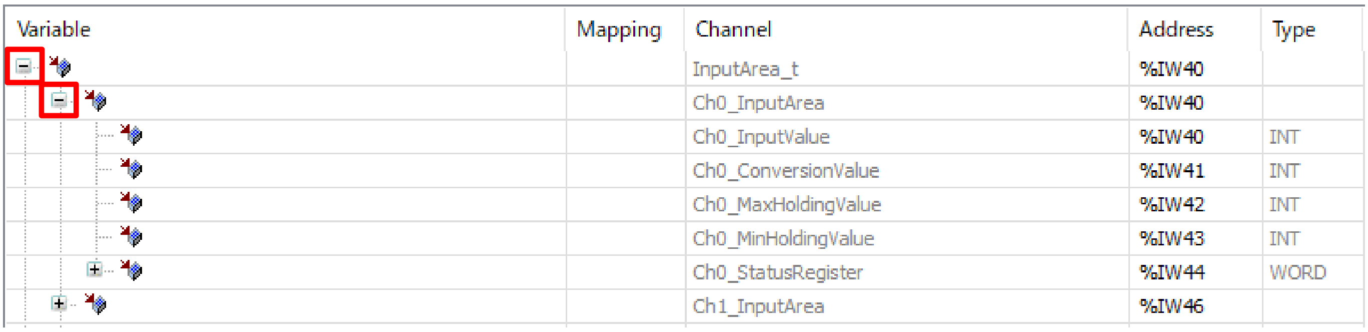
In **Analog_8IN I/O Mapping**, link iCh0_ConversionValue to the corresponding channel, as below.
Double-click **Slot2_Analog_8IN (AGM1AD8)** and open the **Analog_8IN I/O Mapping** tab.



Variable	Mapping	Channel
		InputArea_t
		OutputArea_t

Step 4

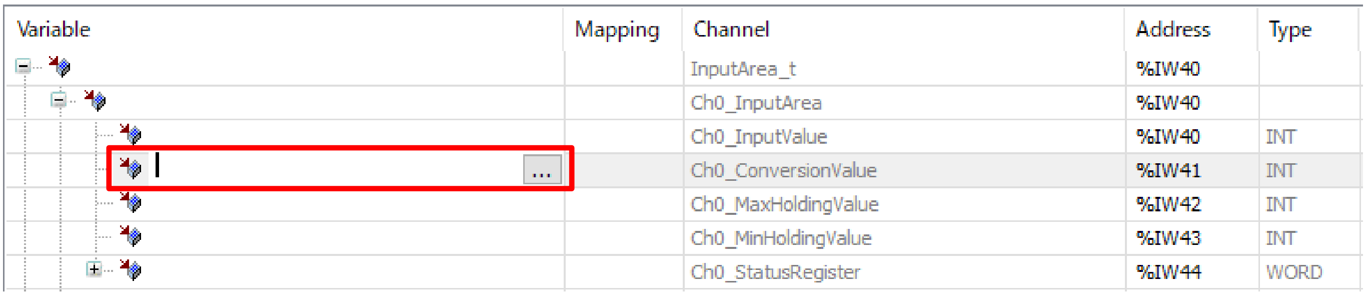
Expand **InputArea_t** through to **Ch1_InputArea** by clicking ⊞.



Variable	Mapping	Channel	Address	Type
		InputArea_t	%IW40	
		Ch0_InputArea	%IW40	
		Ch0_InputValue	%IW40	INT
		Ch0_ConversionValue	%IW41	INT
		Ch0_MaxHoldingValue	%IW42	INT
		Ch0_MinHoldingValue	%IW43	INT
		Ch0_StatusRegister	%IW44	WORD
		Ch1_InputArea	%IW46	

Step 5

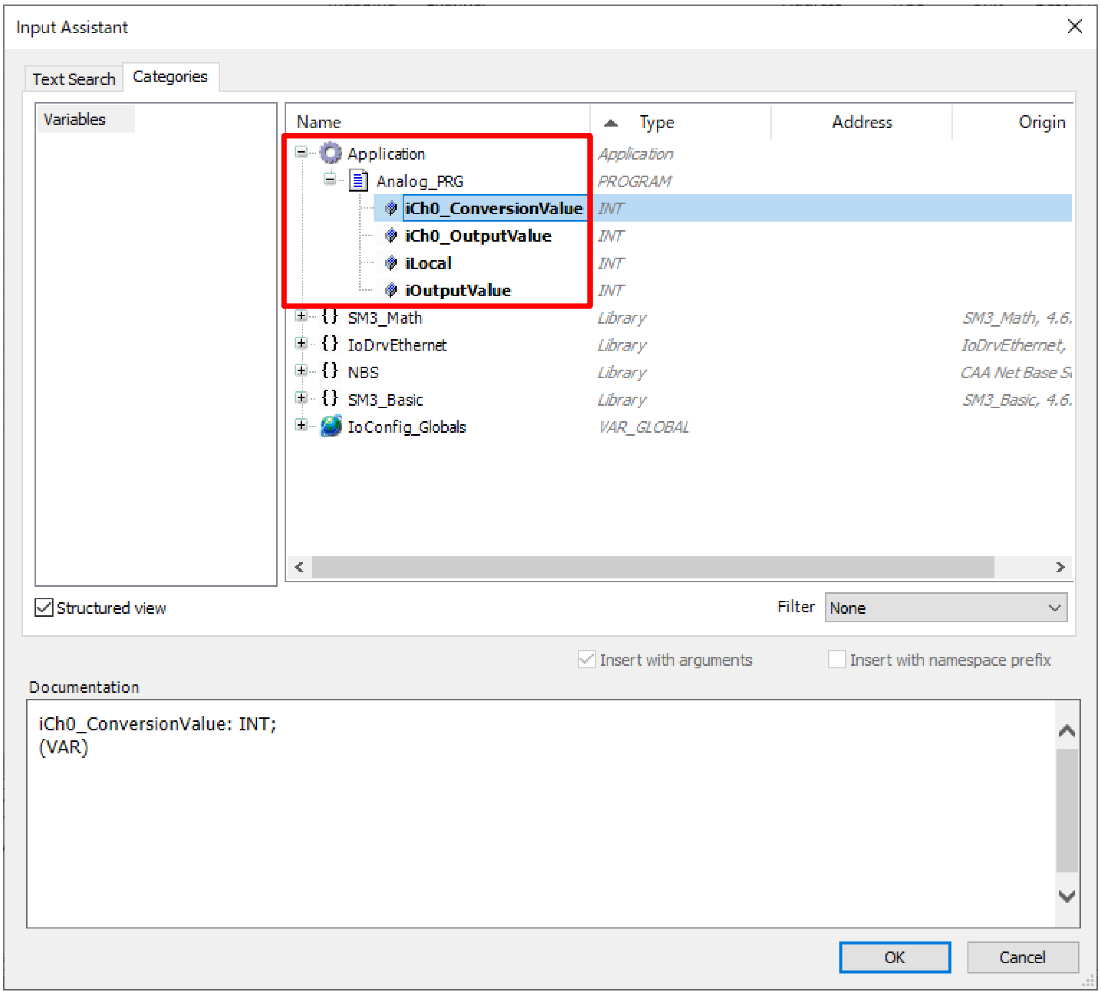
Double-click in the empty row corresponding to **Ch0_ConversionValue** and then click … .



Variable	Mapping	Channel	Address	Type
		InputArea_t	%IW40	
		Ch0_InputArea	%IW40	
		Ch0_InputValue	%IW40	INT
…		Ch0_ConversionValue	%IW41	INT
		Ch0_MaxHoldingValue	%IW42	INT
		Ch0_MinHoldingValue	%IW43	INT
		Ch0_StatusRegister	%IW44	WORD

Step 6

The **Input Assistant** dialog box will be displayed.
Select **Application**, **Analog_PRG**, and then **iCh0_ConversionValue**.



Name	Type	Address	Origin
Application	Application		
Analog_PRG	PROGRAM		
iCh0_ConversionValue	INT		
iCh0_OutputValue	INT		
iLocal	INT		
iOutputValue	INT		
SM3_Math	Library		SM3_Math, 4.6.
IoDrvEthernet	Library		IoDrvEthernet,
NBS	Library		CAA Net Base S
SM3_Basic	Library		SM3_Basic, 4.6.
IoConfig_Globals	VAR_GLOBAL		

☑ Structured view

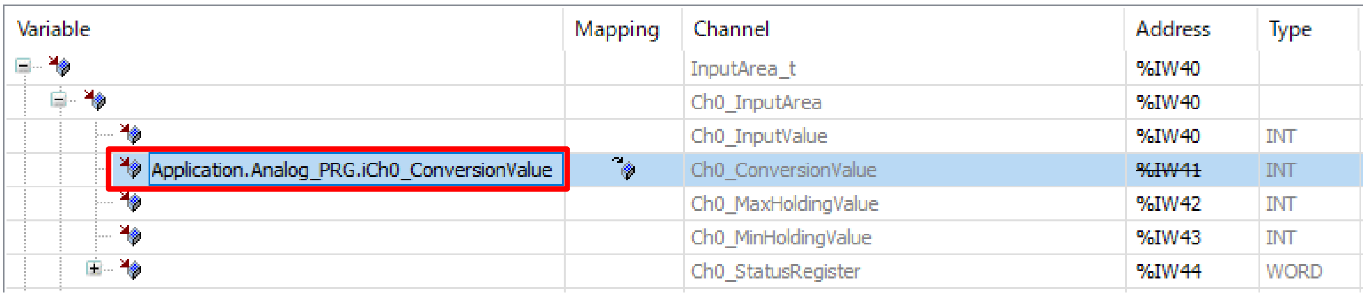
Filter: None

☑ Insert with arguments ☐ Insert with namespace prefix

Documentation

```
iCh0_ConversionValue: INT;
(VAR)
```

iCh0_ConversionValue will be linked to the corresponding channel.

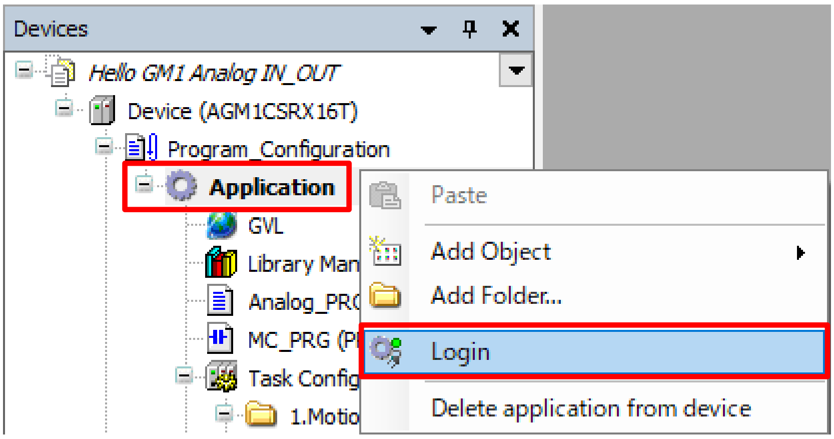


Variable	Mapping	Channel	Address	Type
		InputArea_t	%IW40	
		Ch0_InputArea	%IW40	
		Ch0_InputValue	%IW40	INT
Application.Analog_PRG.iCh0_ConversionValue		Ch0_ConversionValue	~~%IW41~~	INT
		Ch0_MaxHoldingValue	%IW42	INT
		Ch0_MinHoldingValue	%IW43	INT
		Ch0_StatusRegister	%IW44	WORD

4 Communication Operation Check

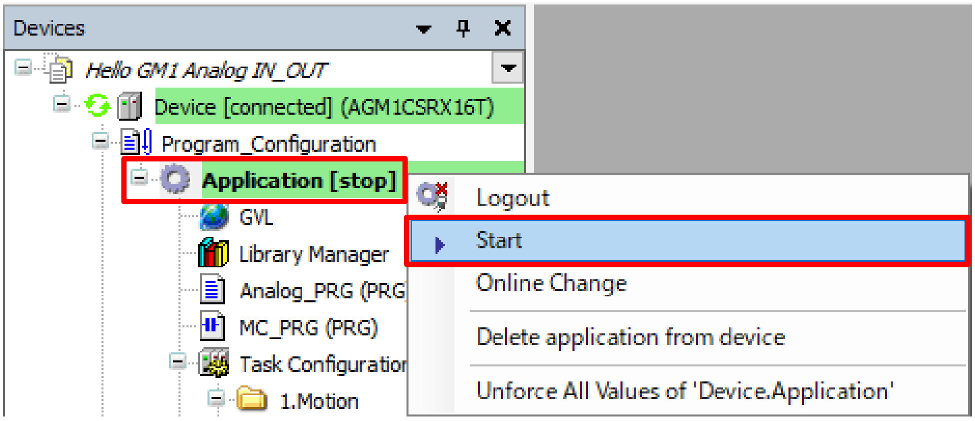
Step 1

Right-click **Application** and then select **Login**.



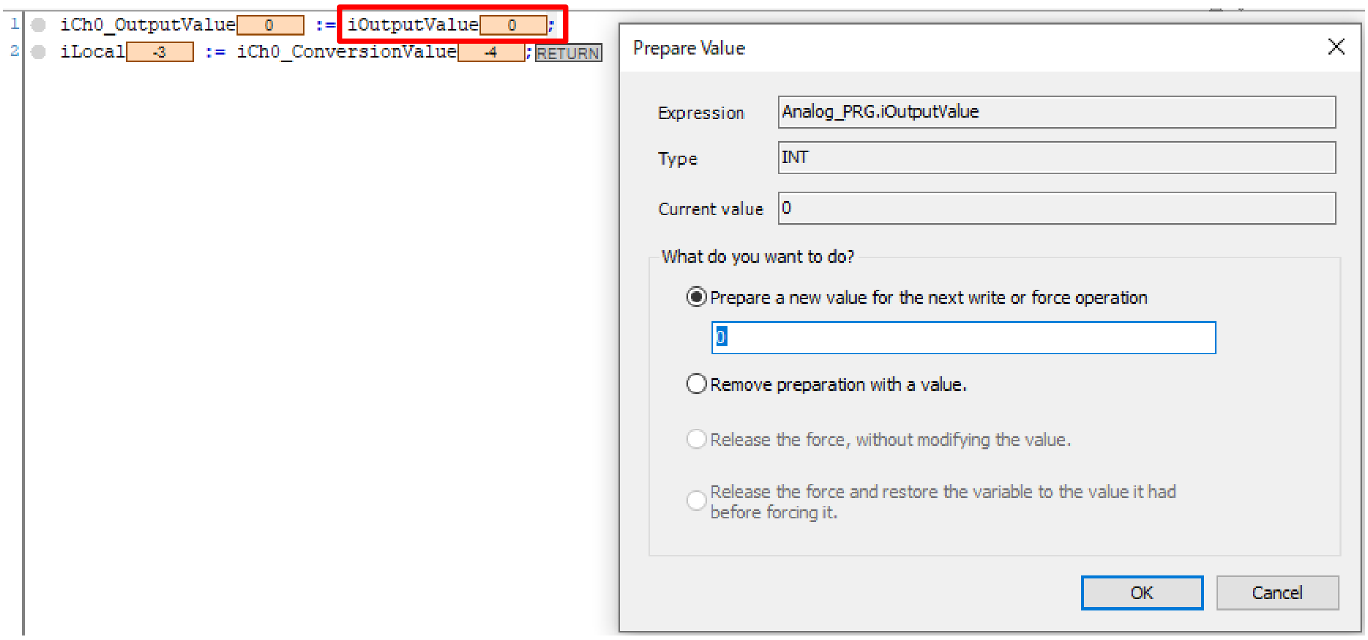
Step 2

Right-click **Application [stop]** and then select **Start** to switch the state from STOP to RUN.



Step 3

Open the **Analog_PRG** tab and then double-click [0] on the right side of **iOutputValue** to open the **Prepare Value** dialog box.



```
iCh0_OutputValue[ 0 ] := iOutputValue[ 0 ];
iLocal[ -3 ] := iCh0_ConversionValue[ -4 ]; RETURN
```

Prepare Value

Expression: Analog_PRG.iOutputValue

Type: INT

Current value: 0

What do you want to do?

● Prepare a new value for the next write or force operation

0

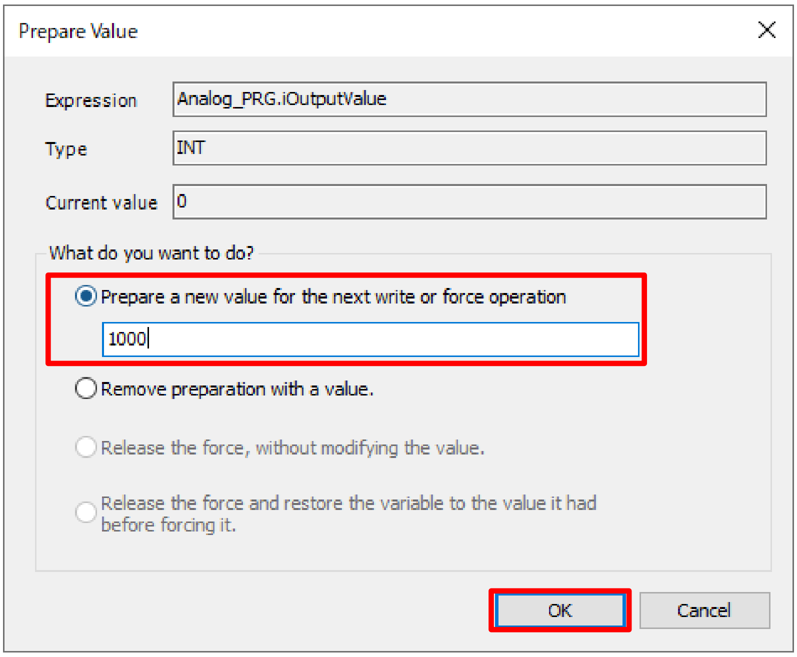
○ Remove preparation with a value.

○ Release the force, without modifying the value.

○ Release the force and restore the variable to the value it had before forcing it.

Step 4

Enter 1000 in the **Prepare a new value for the next write or force operation** field and then click **OK**.



Prepare Value

Expression: Analog_PRG.iOutputValue

Type: INT

Current value: 0

What do you want to do?

● Prepare a new value for the next write or force operation

1000

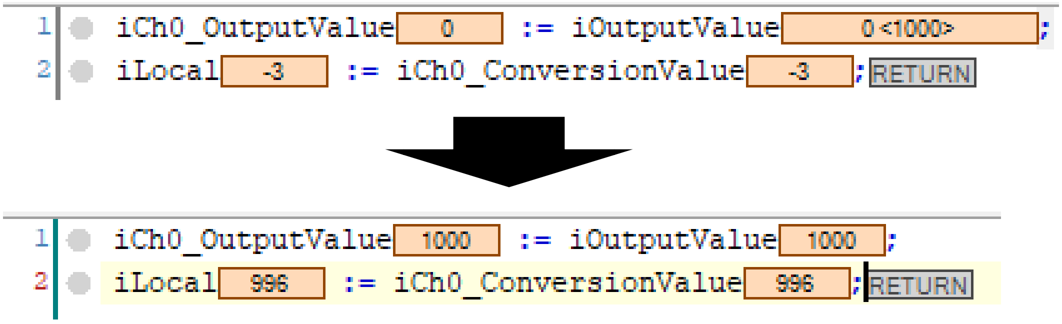
○ Remove preparation with a value.

○ Release the force, without modifying the value.

○ Release the force and restore the variable to the value it had before forcing it.

Step 5

Check that "1000" has been entered only in the **iOutputValue** variable.
In this state, press the Ctrl+F7 keys.



```
iCh0_OutputValue 0 := iOutputValue 0 <1000>;
iLocal -3 := iCh0_ConversionValue -3 ;RETURN
```

```
iCh0_OutputValue 1000 := iOutputValue 1000 ;
iLocal 996 := iCh0_ConversionValue 996 ;RETURN
```

"1000" will be entered in the **iCh0_OutputValue** variable.
At the same time, "1000" will also be entered in the **iCh0_ConversionValue** variable that has been mapped to the input unit.

In this textbook, **RangeSetting** is set to **-10 ~ +10V** (default value) for both the output unit and input unit. Therefore, if "1000" is entered in the **Prepare a new value for the next write or force operation** field, the output unit will output the corresponding voltage value and the input unit will convert the received voltage value to approximately "1000".

INFO

As shown in the figure above, when "0", "-3", and "1000" are entered, "996" is output. This is based on the specifications of the analog input and output units.
When **RangeSetting** is set to **-10 ~ +10V**, an error occurs within 0.2% of F.S. ±32,000.

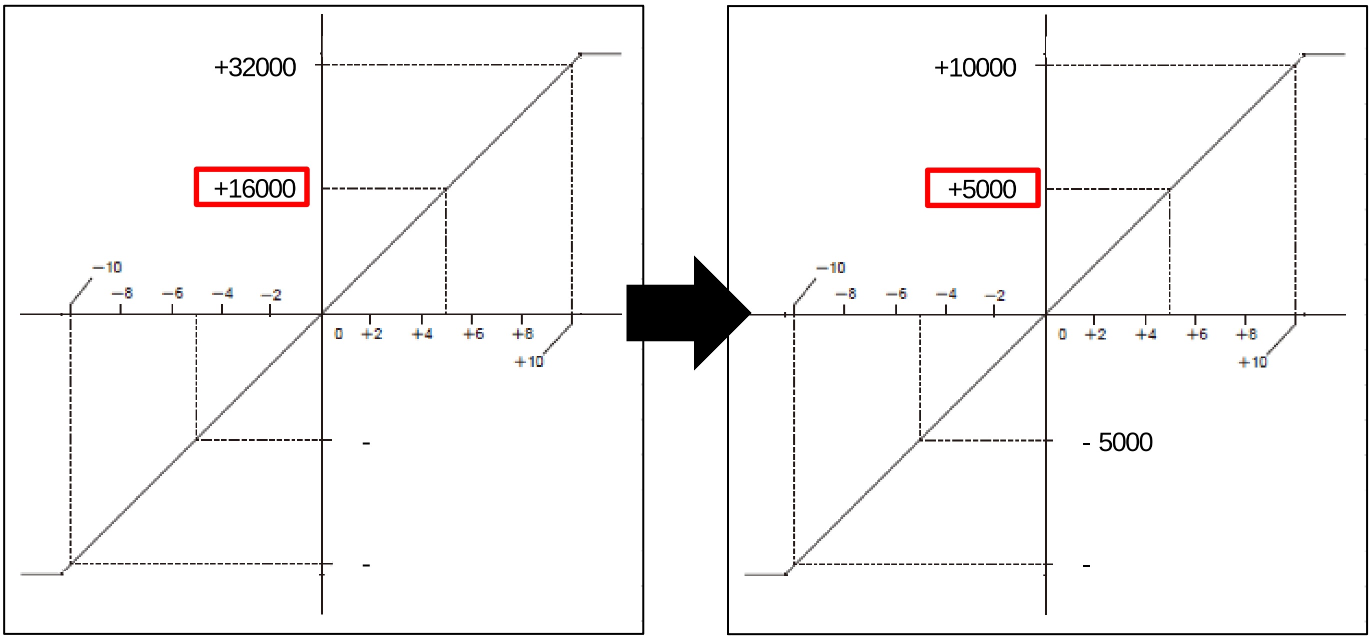
Total accuracy	±0.2%F.S. or less (at +25℃) ±0.4%F.S. or less (at 0 to +55℃)

To avoid this error, use the Offset / Gain processing function.
For details, refer to Section 5.3 or 6.2 in the GM1 Series Reference Manual (Analog I/O Unit Edition).

5 Scale Conversion Function

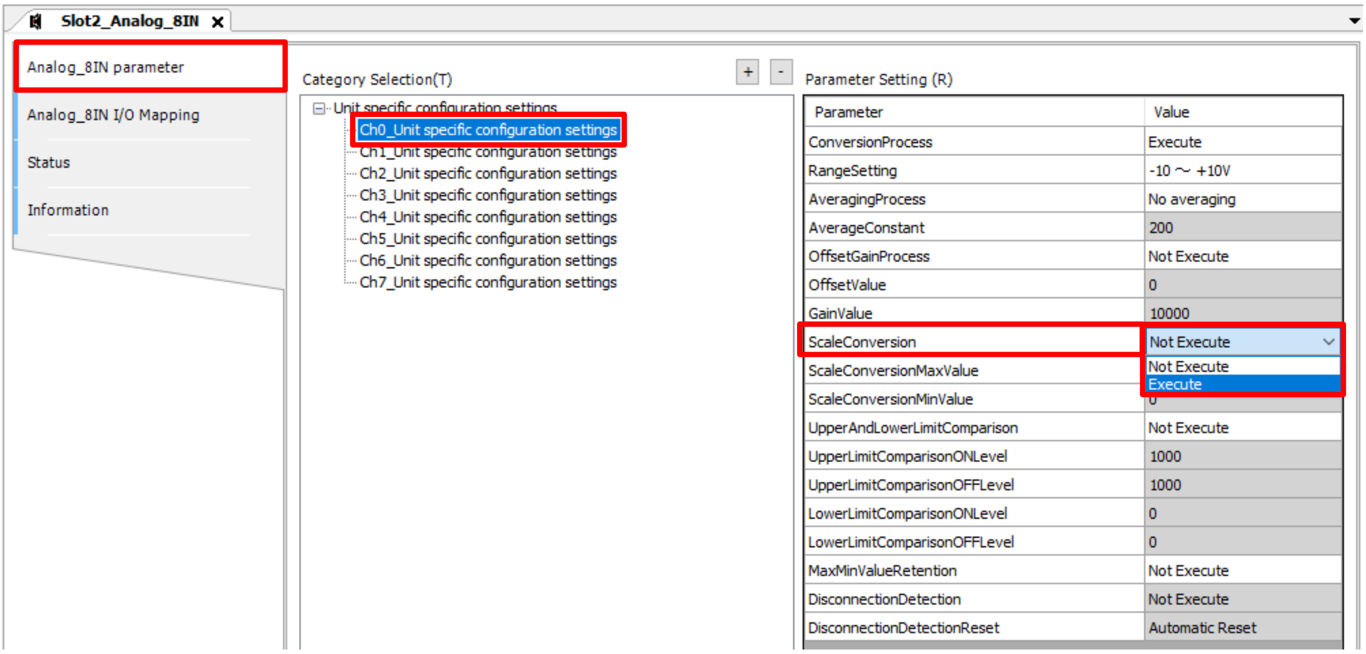
The scale conversion function can be used in both analog input and output units in the same way.
In this textbook, the scale conversion function is enabled in the analog input unit.
Check that "+16000" is output and "+5000" is input.

- This function converts the scale of analog input to an easy-to-handle data range.
- Data subjected to scale conversion is stored in the I/O mapping.
- The function can be used for unit conversion, etc.
- Scale conversion processing is executed on a channel-by-channel basis.



Step 1

Double-click **Slot2_Analog_8IN** and open the **Analog_8IN parameter** tab.
Select **Ch0_Unit specific configuration settings** and then **ScaleConversion**, and change the setting from **Not Execute** to **Execute**.


Slot2_Analog_8IN
Analog_8IN parameter
Analog_8IN I/O Mapping
Status
Information
Category Selection(T)
+
-
Unit specific configuration settings
Ch0_Unit specific configuration settings
Ch1_Unit specific configuration settings
Ch2_Unit specific configuration settings
Ch3_Unit specific configuration settings
Ch4_Unit specific configuration settings
Ch5_Unit specific configuration settings
Ch6_Unit specific configuration settings
Ch7_Unit specific configuration settings
Parameter Setting (R)

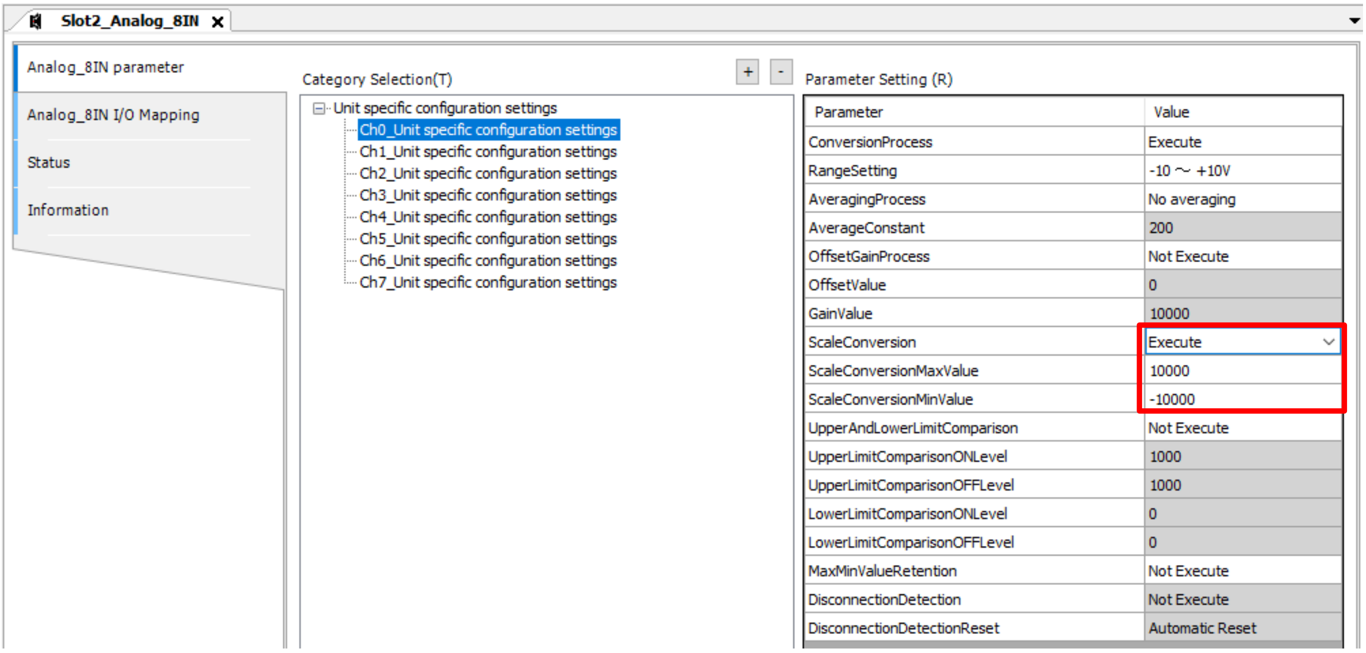
Parameter	Value
ConversionProcess	Execute
RangeSetting	-10 ～ +10V
AveragingProcess	No averaging
AverageConstant	200
OffsetGainProcess	Not Execute
OffsetValue	0
GainValue	10000
ScaleConversion	Not Execute Not Execute Execute
ScaleConversionMaxValue	
ScaleConversionMinValue	0
UpperAndLowerLimitComparison	Not Execute
UpperLimitComparisonONLevel	1000
UpperLimitComparisonOFFLevel	1000
LowerLimitComparisonONLevel	0
LowerLimitComparisonOFFLevel	0
MaxMinValueRetention	Not Execute
DisconnectionDetection	Not Execute
DisconnectionDetectionReset	Automatic Reset

Step 2

Change the settings as below.

ScaleConversionMaxValue: 10000

ScaleConversionMinValue: -10000

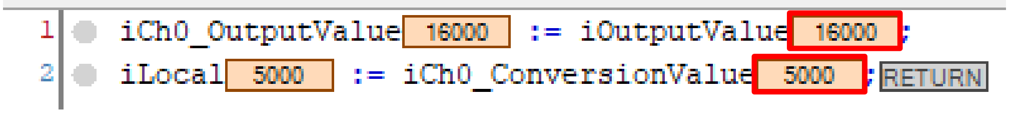


Slot2_Analog_8IN
Analog_8IN parameter
Analog_8IN I/O Mapping
Status
Information
Category Selection(T)
+
-
Unit specific configuration settings
Ch0_Unit specific configuration settings
Ch1_Unit specific configuration settings
Ch2_Unit specific configuration settings
Ch3_Unit specific configuration settings
Ch4_Unit specific configuration settings
Ch5_Unit specific configuration settings
Ch6_Unit specific configuration settings
Ch7_Unit specific configuration settings
Parameter Setting (R)

Parameter	Value
ConversionProcess	Execute
RangeSetting	-10 ～ +10V
AveragingProcess	No averaging
AverageConstant	200
OffsetGainProcess	Not Execute
OffsetValue	0
GainValue	10000
ScaleConversion	Execute
ScaleConversionMaxValue	10000
ScaleConversionMinValue	-10000
UpperAndLowerLimitComparison	Not Execute
UpperLimitComparisonONLevel	1000
UpperLimitComparisonOFFLevel	1000
LowerLimitComparisonONLevel	0
LowerLimitComparisonOFFLevel	0
MaxMinValueRetention	Not Execute
DisconnectionDetection	Not Execute
DisconnectionDetectionReset	Automatic Reset

Step 3

Right-click **Application** and select **Login**, and then enter 16000 into **iOutputValue** in the program.



```
iCh0_OutputValue 16000 := iOutputValue 16000 ;
iLocal 5000 := iCh0_ConversionValue 5000 ; RETURN
```

The output unit outputs +16000, which is equivalent to +5 V.

We can see that the input unit receives +5 V, which is then scale-converted to +5000 for input.

6 Disconnection Detection Function

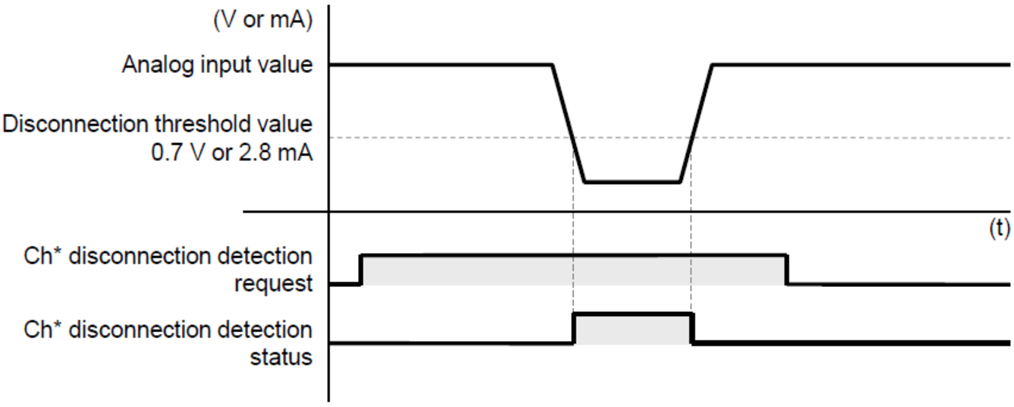
･This function sets the disconnection detection status to ON to warn of an error state when input is disconnected or unconnected.

･The disconnection detection function operates only in the following ranges.

Range	Detection level
1 to 5 V	0.7 V or less
4 to 20 mA	2.8 mA or less

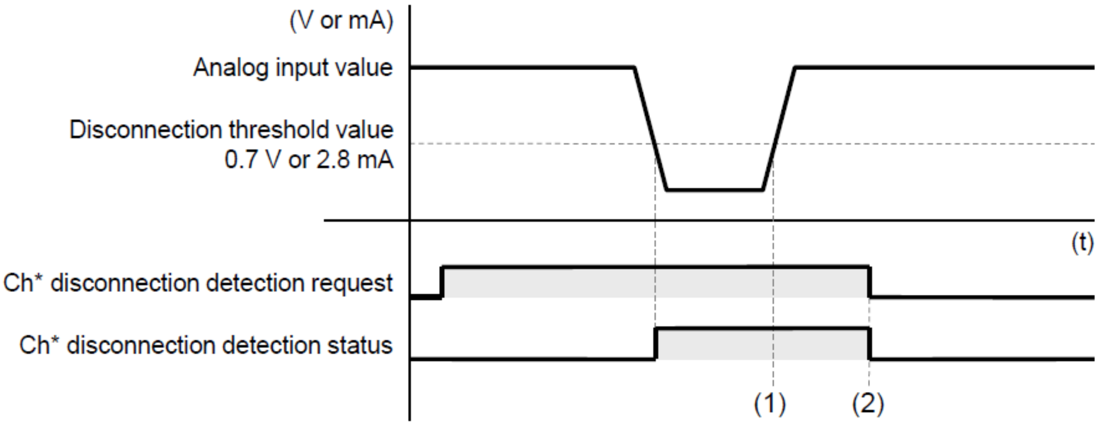
■ Behavior when "Disconnection detection reset" is set to "Auto"

When the input value exceeds the detection level while "Ch* disconnection detection request" is ON, "Ch* disconnection detection status" is automatically set to OFF.


(V or mA)
Analog input value
Disconnection threshold value
0.7 V or 2.8 mA
(t)
Ch* disconnection detection request
Ch* disconnection detection status

■ Behavior when "Disconnection detection reset" is set to "Manual"

In manual setting mode, when "Ch* disconnection detection request" is set to OFF using the user program, "Ch* disconnection detection status" is set to OFF.


(V or mA)
Analog input value
Disconnection threshold value
0.7 V or 2.8 mA
(t)
Ch* disconnection detection request
Ch* disconnection detection status
(1)
(2)

(1)	"Ch* disconnection detection status" is not set to OFF automatically even when the disconnection is restored.
(2)	When "Ch* disconnection detection request" is set to OFF, "Ch* disconnection detection status" is set to OFF.

Step 1

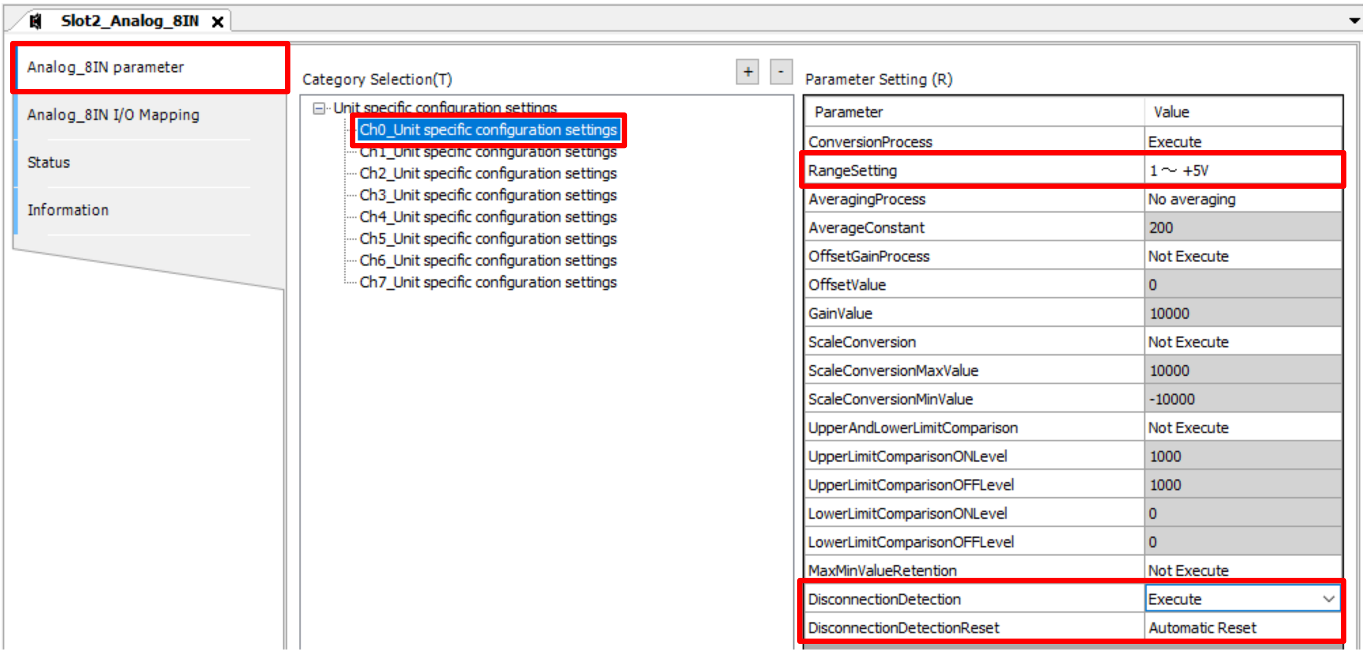
Double-click **Slot2_Analog_8IN** and open the **Analog_8IN parameter** tab.

Select **Ch0_Unit specific configuration settings** and then set the following items.

RangeSetting 1 ~ +5V

DisconnectionDetection: **Execute**

DisconnectionDetectionReset: **Automatic Reset**



Slot2_Analog_8IN

- Analog_8IN parameter
- Analog_8IN I/O Mapping
- Status
- Information

Category Selection(T)

- Unit specific configuration settings
  - Ch0_Unit specific configuration settings
  - Ch1_Unit specific configuration settings
  - Ch2_Unit specific configuration settings
  - Ch3_Unit specific configuration settings
  - Ch4_Unit specific configuration settings
  - Ch5_Unit specific configuration settings
  - Ch6_Unit specific configuration settings
  - Ch7_Unit specific configuration settings

Parameter Setting (R)

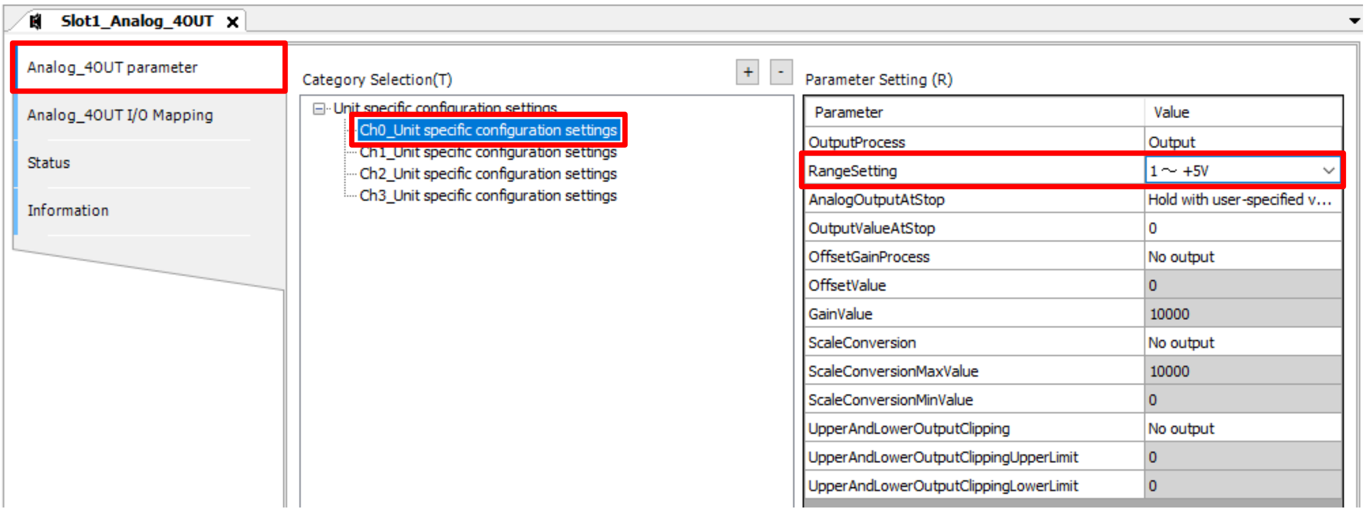
Parameter	Value
ConversionProcess	Execute
RangeSetting	1 ~ +5V
AveragingProcess	No averaging
AverageConstant	200
OffsetGainProcess	Not Execute
OffsetValue	0
GainValue	10000
ScaleConversion	Not Execute
ScaleConversionMaxValue	10000
ScaleConversionMinValue	-10000
UpperAndLowerLimitComparison	Not Execute
UpperLimitComparisonONLevel	1000
UpperLimitComparisonOFFLevel	1000
LowerLimitComparisonONLevel	0
LowerLimitComparisonOFFLevel	0
MaxMinValueRetention	Not Execute
DisconnectionDetection	Execute
DisconnectionDetectionReset	Automatic Reset

Step 2

Double-click **Slot1_Analog_4OUT** and open the **Analog_4OUT parameter** tab.

Select **Ch0_Unit specific configuration settings** and then set **RangeSetting** to the same value as for analog input, as below.

RangeSetting: 1 ~ +5V



Slot1_Analog_4OUT

- Analog_4OUT parameter
- Analog_4OUT I/O Mapping
- Status
- Information

Category Selection(T)

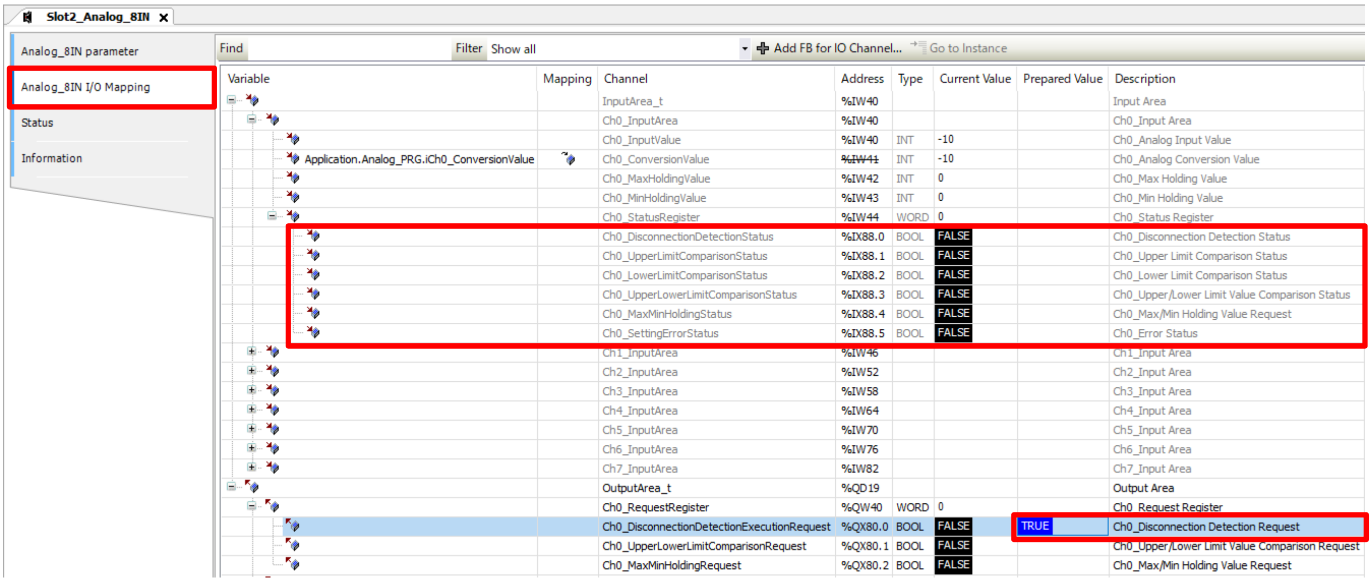
- Unit specific configuration settings
  - Ch0_Unit specific configuration settings
  - Ch1_Unit specific configuration settings
  - Ch2_Unit specific configuration settings
  - Ch3_Unit specific configuration settings

Parameter Setting (R)

Parameter	Value
OutputProcess	Output
RangeSetting	1 ~ +5V
AnalogOutputAtStop	Hold with user-specified v...
OutputValueAtStop	0
OffsetGainProcess	No output
OffsetValue	0
GainValue	10000
ScaleConversion	No output
ScaleConversionMaxValue	10000
ScaleConversionMinValue	0
UpperAndLowerOutputClipping	No output
UpperAndLowerOutputClippingUpperLimit	0
UpperAndLowerOutputClippingLowerLimit	0

Step 3

Right-click **Application** and select **Login**, and then open the **Analog_8IN I/O Mapping** tab.
Expand **Ch0_StatusRegister**. Set **Ch0_Disconnection Detection Request** to **TRUE**.



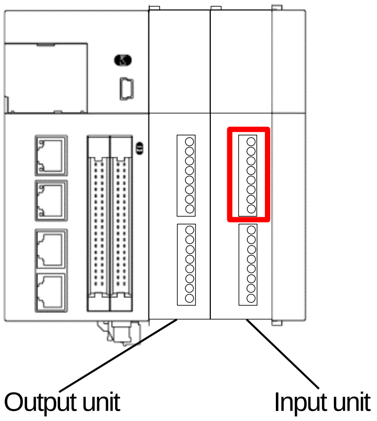
Variable	Mapping	Channel	Address	Type	Current Value	Prepared Value	Description
		InputArea_t	%IW40				Input Area
		Ch0_InputArea	%IW40				Ch0_Input Area
		Ch0_InputValue	%IW40	INT	-10		Ch0_Analog Input Value
Application.Analog_PRG.iCh0_ConversionValue		Ch0_ConversionValue	~~%IW41~~	INT	-10		Ch0_Analog Conversion Value
		Ch0_MaxHoldingValue	%IW42	INT	0		Ch0_Max Holding Value
		Ch0_MinHoldingValue	%IW43	INT	0		Ch0_Min Holding Value
		Ch0_StatusRegister	%IW44	WORD	0		Ch0_Status Register
		Ch0_DisconnectionDetectionStatus	%IX88.0	BOOL	FALSE		Ch0_Disconnection Detection Status
		Ch0_UpperLimitComparisonStatus	%IX88.1	BOOL	FALSE		Ch0_Upper Limit Comparison Status
		Ch0_LowerLimitComparisonStatus	%IX88.2	BOOL	FALSE		Ch0_Lower Limit Comparison Status
		Ch0_UpperLowerLimitComparisonStatus	%IX88.3	BOOL	FALSE		Ch0_Upper/Lower Limit Value Comparison Status
		Ch0_MaxMinHoldingStatus	%IX88.4	BOOL	FALSE		Ch0_Max/Min Holding Value Request
		Ch0_SettingErrorStatus	%IX88.5	BOOL	FALSE		Ch0_Error Status
		Ch1_InputArea	%IW46				Ch1_Input Area
		Ch2_InputArea	%IW52				Ch2_Input Area
		Ch3_InputArea	%IW58				Ch3_Input Area
		Ch4_InputArea	%IW64				Ch4_Input Area
		Ch5_InputArea	%IW70				Ch5_Input Area
		Ch6_InputArea	%IW76				Ch6_Input Area
		Ch7_InputArea	%IW82				Ch7_Input Area
		OutputArea_t	%QD19				Output Area
		Ch0_RequestRegister	%QW40	WORD	0		Ch0_Request Register
		Ch0_DisconnectionDetectionExecutionRequest	%QX80.0	BOOL	FALSE	TRUE	Ch0_Disconnection Detection Request
		Ch0_UpperLowerLimitComparisonRequest	%QX80.1	BOOL	FALSE		Ch0_Upper/Lower Limit Value Comparison Request
		Ch0_MaxMinHoldingRequest	%QX80.2	BOOL	FALSE		Ch0_Max/Min Holding Value Request

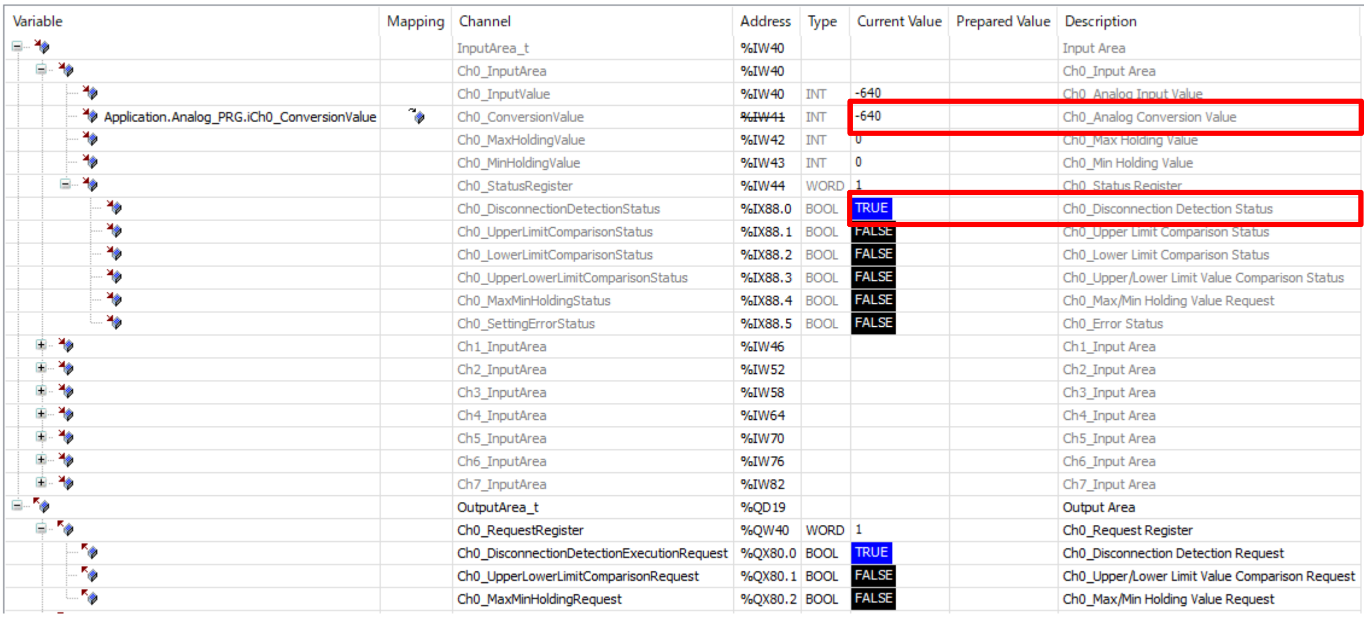
Step 4

Disconnect the connector from the input unit while checking the value in the **Current Value** column corresponding to **Ch0_InputValue**.

As soon as "-640" (equivalent to 0.7 V or less) is entered, **Ch0_DisconnectionDetectionStatus** is set to **TRUE**.

Because **DisconnectionDetectionReset** is set o **Automatic Reset**, **Ch0_DisconnectionDetectionStatus** is set to **FALSE** when the connector is reconnected.





Variable	Mapping	Channel	Address	Type	Current Value	Prepared Value	Description
		InputArea_t	%IW40				Input Area
		Ch0_InputArea	%IW40				Ch0_Input Area
		Ch0_InputValue	%IW40	INT	-640		Ch0_Analog Input Value
Application.Analog_PRG.iCh0_ConversionValue		Ch0_ConversionValue	~~%IW41~~	INT	-640		Ch0_Analog Conversion Value
		Ch0_MaxHoldingValue	%IW42	INT	0		Ch0_Max Holding Value
		Ch0_MinHoldingValue	%IW43	INT	0		Ch0_Min Holding Value
		Ch0_StatusRegister	%IW44	WORD	1		Ch0_Status Register
		Ch0_DisconnectionDetectionStatus	%IX88.0	BOOL	TRUE		Ch0_Disconnection Detection Status
		Ch0_UpperLimitComparisonStatus	%IX88.1	BOOL	FALSE		Ch0_Upper Limit Comparison Status
		Ch0_LowerLimitComparisonStatus	%IX88.2	BOOL	FALSE		Ch0_Lower Limit Comparison Status
		Ch0_UpperLowerLimitComparisonStatus	%IX88.3	BOOL	FALSE		Ch0_Upper/Lower Limit Value Comparison Status
		Ch0_MaxMinHoldingStatus	%IX88.4	BOOL	FALSE		Ch0_Max/Min Holding Value Request
		Ch0_SettingErrorStatus	%IX88.5	BOOL	FALSE		Ch0_Error Status
		Ch1_InputArea	%IW46				Ch1_Input Area
		Ch2_InputArea	%IW52				Ch2_Input Area
		Ch3_InputArea	%IW58				Ch3_Input Area
		Ch4_InputArea	%IW64				Ch4_Input Area
		Ch5_InputArea	%IW70				Ch5_Input Area
		Ch6_InputArea	%IW76				Ch6_Input Area
		Ch7_InputArea	%IW82				Ch7_Input Area
		OutputArea_t	%QD19				Output Area
		Ch0_RequestRegister	%QW40	WORD	1		Ch0_Request Register
		Ch0_DisconnectionDetectionExecutionRequest	%QX80.0	BOOL	TRUE		Ch0_Disconnection Detection Request
		Ch0_UpperLowerLimitComparisonRequest	%QX80.1	BOOL	FALSE		Ch0_Upper/Lower Limit Value Comparison Request
		Ch0_MaxMinHoldingRequest	%QX80.2	BOOL	FALSE		Ch0_Max/Min Holding Value Request

7 Analog Output Hold Function in STOP Mode

This function holds the analog output when the operation mode of the GM1 controller changes from RUN to STOP.

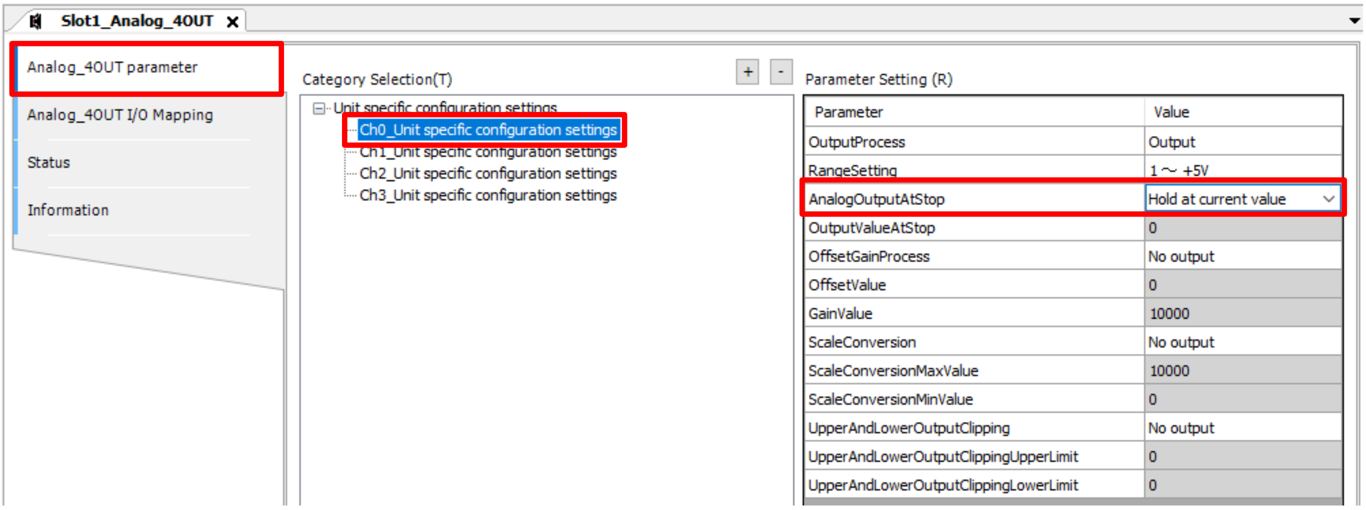
･Analog output hold settings are configured on a channel-by-channel basis.
･"Hold at any value" or "Hold at current value" can be selected for the analog output value for each channel.

INFO

･If "Hold at any value" is selected, the analog signal to be output differs according to the value of **RangeSetting**.
･When an error occurs, the output is turned OFF (0 V or 0 mA).

Step 1

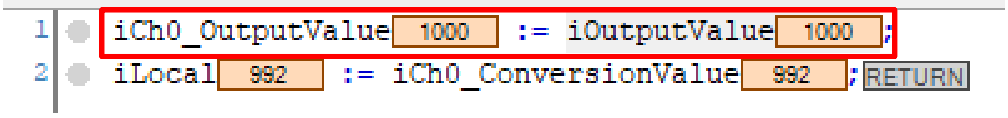
Double-click **Slot1_Analog_4OUT** and open the **Analog_4OUT parameter** tab.
Select **Ch0_Unit specific configuration settings** and then set the following item.
AnalogOutputAtStop: **Hold at current value**


Slot1_Analog_4OUT
Analog_4OUT parameter
Analog_4OUT I/O Mapping
Status
Information
Category Selection(T)
Unit specific configuration settings
Ch0_Unit specific configuration settings
Ch1_Unit specific configuration settings
Ch2_Unit specific configuration settings
Ch3_Unit specific configuration settings
Parameter Setting (R)

Parameter	Value
OutputProcess	Output
RangeSetting	1 ~ +5V
AnalogOutputAtStop	Hold at current value
OutputValueAtStop	0
OffsetGainProcess	No output
OffsetValue	0
GainValue	10000
ScaleConversion	No output
ScaleConversionMaxValue	10000
ScaleConversionMinValue	0
UpperAndLowerOutputClipping	No output
UpperAndLowerOutputClippingUpperLimit	0
UpperAndLowerOutputClippingLowerLimit	0

Step 2

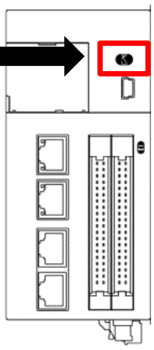
Right-click **Application** and select **Login**, and then open the **Analog_PRG** tab.
Enter "1000" into **iOutputValue** and check that "1000" has been input into **iCh0_OutputValue**.

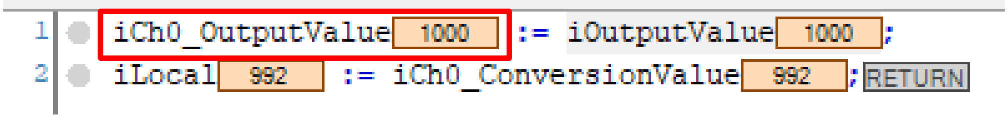


```
iCh0_OutputValue 1000 := iOutputValue 1000 ;
iLocal 992 := iCh0_ConversionValue 992 ;RETURN
```

Step 3

In this state, switch the RUN/STOP selector switch on the GM1 controller to STOP.
Check that "1000", which is the value (current value) displayed before the switch was switched to STOP, is maintained as the output value.





```
iCh0_OutputValue 1000 := iOutputValue 1000 ;
iLocal 992 := iCh0_ConversionValue 992 ;RETURN
```

Memo

Revision History

Date of issue	Manual code	Revision details
April 2022	AIM0013_01	First edition

Panasonic Industry Co., Ltd.
7-1-1 Morofuku, Daito City, Osaka, 574-0044, Japan
Industrial. Panasonic. com/ac/e

April 2022