MITSUBISHI ELECTRIC
Changes for the Better

PLC

PLC MELSEC Process Control System Basics

Welcome to the Basic Course on MELSEC Process Control System.
This is a tutorial for beginners of the MELSEC process control system.

L(NA)00122ENG

Introduction Purpose of the Course

This training course is designed for those who wish to build MELSEC process control systems for the first time. You will learn the features of the MELSEC modules and PX Developer and how to use them.

Introduction Course Structure

The contents of this course are as follows.
We recommend that you start from Chapter 1.

Chapter 1 - What is the MELSEC Process Control System?

You will learn about the features of the modules and software of the MELSEC process control system.

Chapter 2 - System Configuration

You will learn about the configuration of the process control system on which the course is based.

Chapter 3 - FBD Programming

You will learn about FBD programming using the PX Developer programming tools, with exercises including FBD programming, parameter setting, and writing to programmable controller CPUs.

Chapter 4 - Program Monitoring and Tuning

You will learn about program monitoring and tuning using the PX Developer programming and monitoring tools.

Chapter 5 - Final Test

Passing grade: 60% or higher.

Introduction How to Use This e-Learning Tool

Go to the next page		Go to the next page.
Back to the previous page		Back to the previous page.
Move to the desired page		"Table of Contents" will be displayed, enabling you to navigate to the desired page.
Exit the learning		Exit the learning. Window such as "Contents" window and the learning will be closed.

Introduction Cautions for Use

Safety precautions

Before using the physical hardware please read the Safety Precautions in the corresponding manuals and follow the relevant safety information contained therein.

Chapter 1 What is the MELSEC Process Control System?

In this chapter, you will learn the features of key modules and software of a MELSEC process control system.

1.1 Outline of MELSEC Process Control System

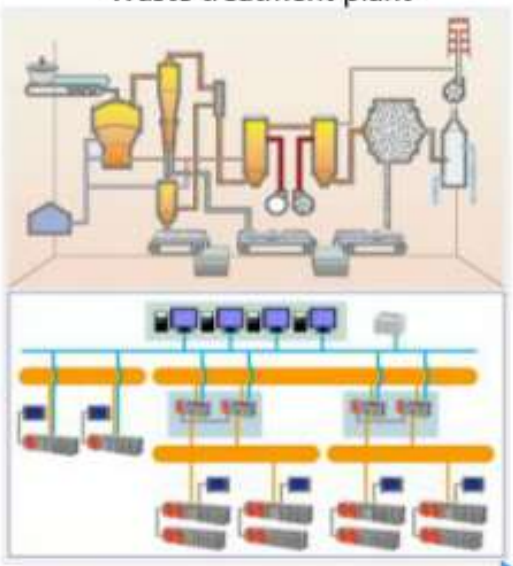
A MELSEC process control system is designed for process control applications (control of temperature, flow rate, pressure, level, etc.), and consists primarily of the following MELSEC-Q Series modules and software.

- **Process CPU** for high-speed loop and sequence control
- **Analog module with isolated channels** that can be directly connected to sensor, control valve or other input/outputs
- **PX Developer**, an FBD software package for process control system
  - **Programming tool**, with which even complex loop control can be programmed easily
  - **Monitoring tool**, with which loop control monitoring and tuning can be performed easily
- **Redundant CPUs** to ensure uninterrupted system operation in the event of a sudden failure

1.2 Range of Application of MELSEC Process Control Systems

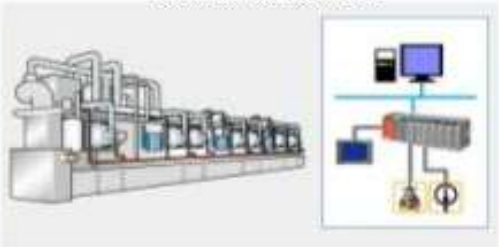
MELSEC process control systems are used in a wide range of fields and applications, from device to plant control, and from continuous to batch to discrete processes.

Waste treatment plant



Food processing machine

Industrial furnace



Device control — Plant control

Fine chemical plant

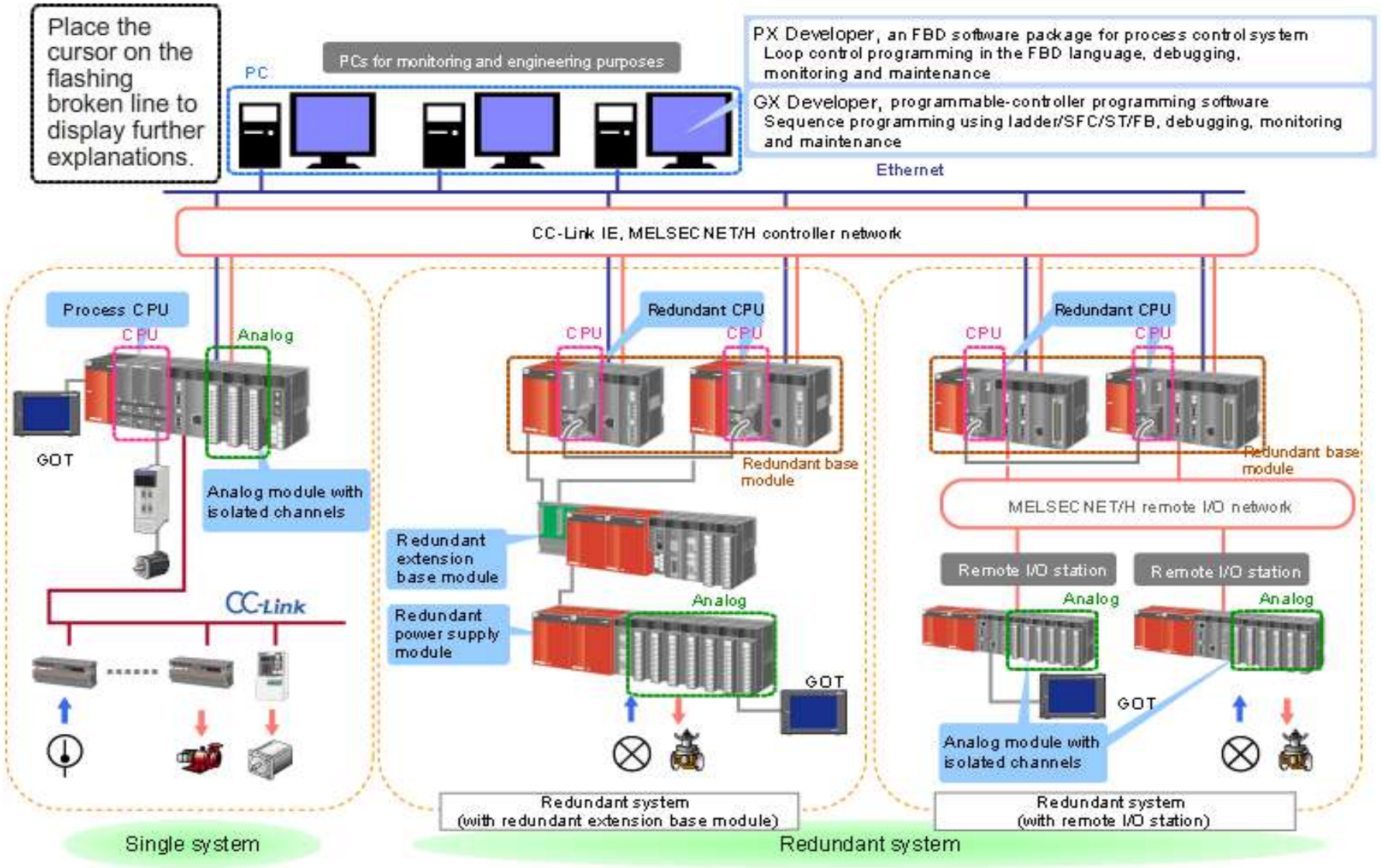
Sewage treatment plant

Continuous process — Batch process — Discrete process

Applicable fields	Food, Medicine, Chemical/fine chemical, Steel, Industrial furnaces, Environment, Water supply and sewerage, Paper/pulp, Semiconductors, Building/Air conditioning, Marine vessels

1.3 System Components and Features

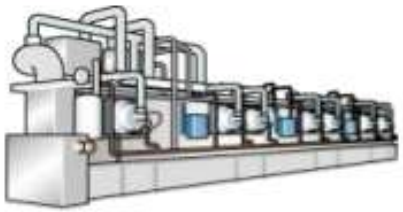
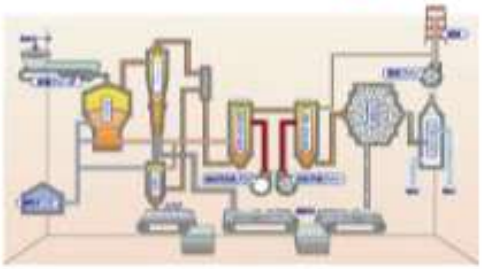
MELSEC process control systems can be configured to meet various individual needs, as a single system, as a redundant system or as a network of single/dual subsystems. The following figures show typical examples of MELSEC process control systems.



1.4 MELSEC Process Control System Lineup

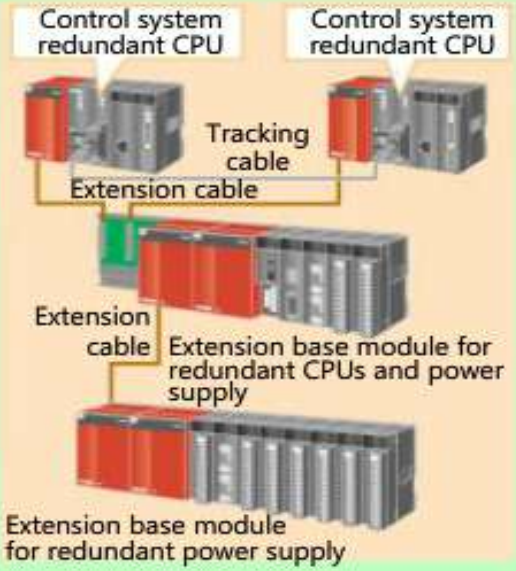
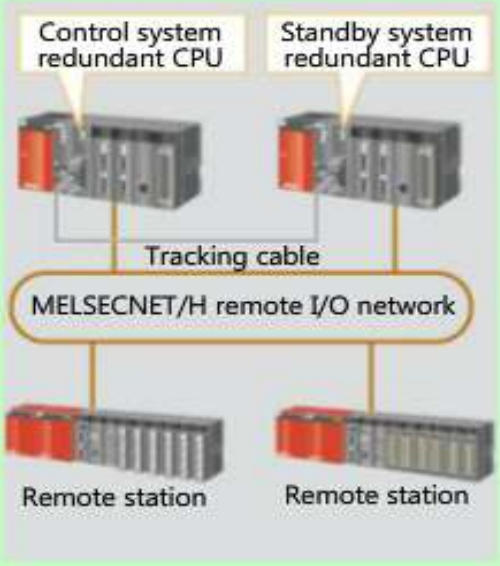
1.4.1 Process CPU

A range of process CPUs are available that all offer high-speed loop (400 μs/PID loop) and sequence control. Simply choose the most suitable one for your specific application, devices and plant environment.

Model	Q02PHCPU	Q06PHCPU	Q12PHCPU	Q25PHCPU
Process CPU				
Capacity for programming	28K steps	60K steps	124K steps	252K steps
Applicable fields	Small ← System size → Large Device — Plant    Food processing machines, industrial furnaces, air conditioning/heat source systems and other applications Plants for water treatment, chemicals, environment, steel and other applications			

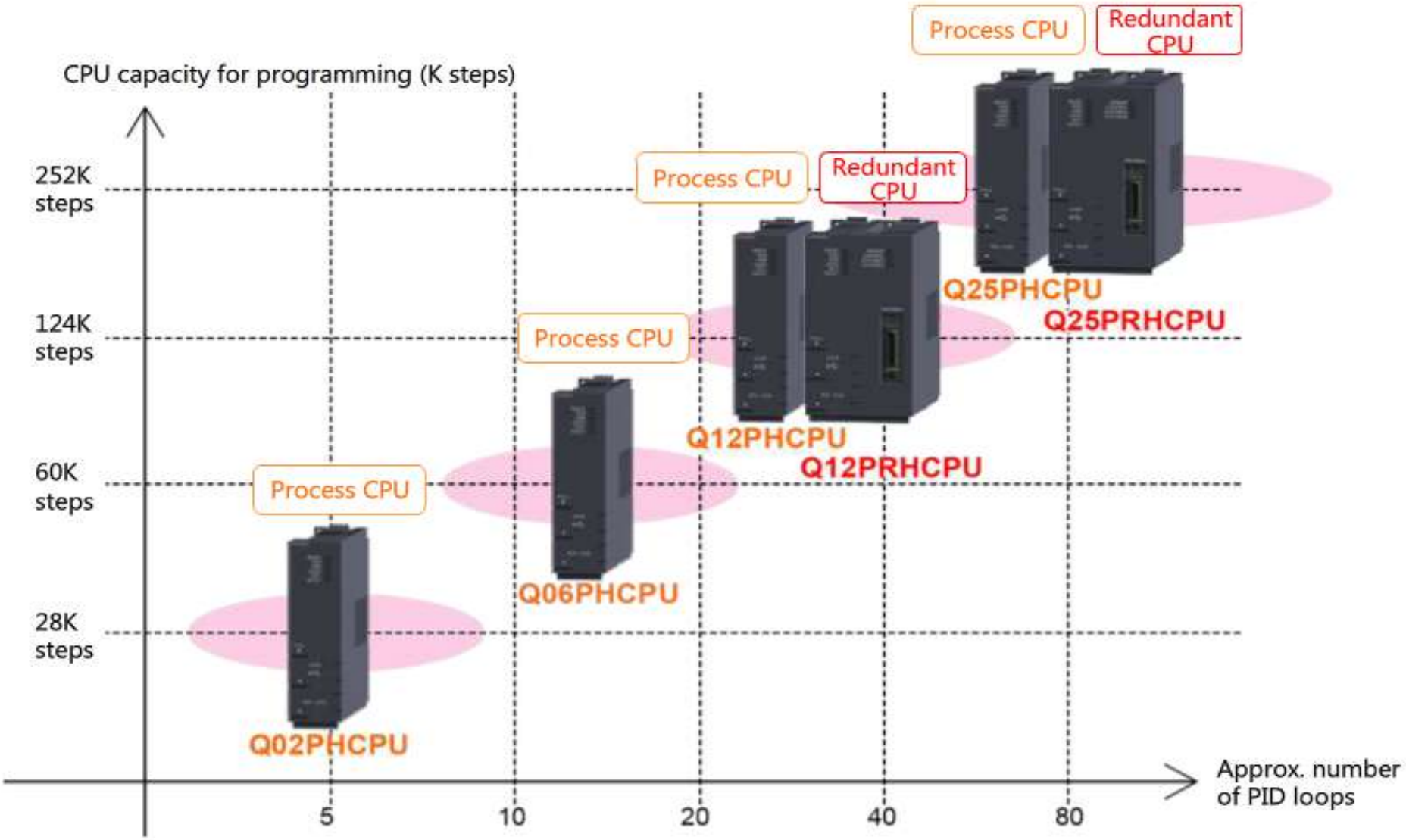
1.4.2 Redundant CPU

A redundant system offers highly reliable high-speed loop and sequence control by means of redundant CPUs, networks and power supplies. Choose the extension base module type or the remote I/O station type to suit your specific needs.

Model		Q12PRHCPU	Q25PRHCPU
Redundant CPU			
Capacity for programming		124K steps	252K steps
System structure		Extension base module type	Remote I/O station type
Application	[Extension base module type] Recommended where high-speed response is required.		
	[Remote I/O station type] Recommended where multiple remote stations are installed in the system.		

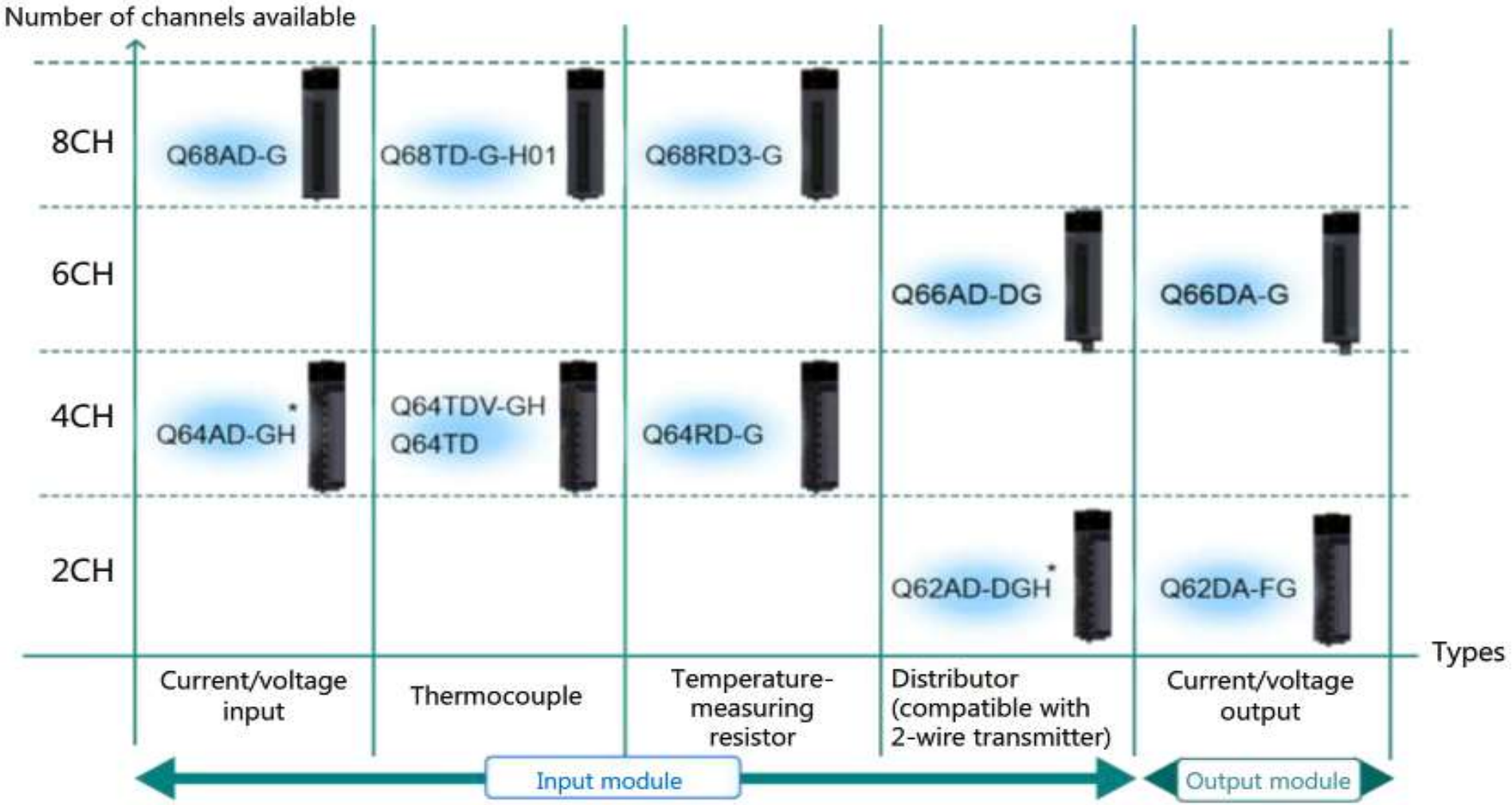
1.4.3 CPU Lineup for Systems of Any Size

From among the CPU lineup, you can select the right one for the size of your system, whether it is for device process control with several loops or for plant process control with several dozen loops.



1.4.4 Analog Module with Isolated Channels

Every analog module is equipped with channels that are isolated from each other. Besides saving space, these modules are available in a variety of specifications including high-precision high-resolution models and multi-channel (6 and 8 channels) versions.

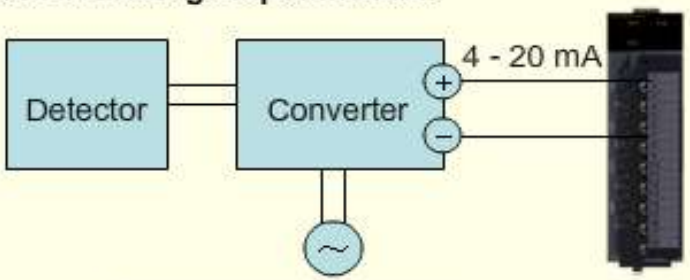


*: High-precision high-resolution type

1.4.4 Additional Information - Analog Module with Isolated Channels

The following is additional information on analog input modules with isolated channels.

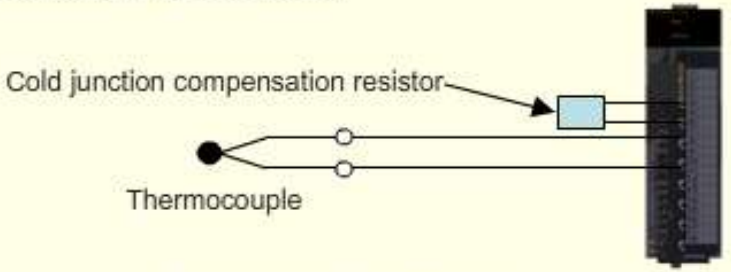
Current/voltage input module



Example of connection to current/voltage input module

The input module is designed to receive current signals of 4 - 20 mA and voltage signals of 1 - 5 V from a converter.

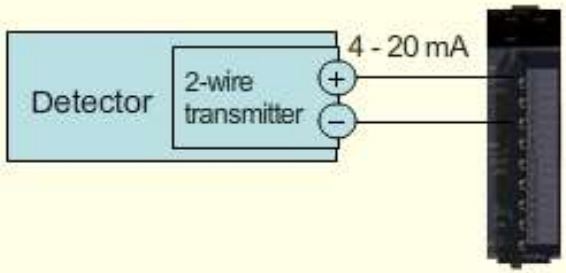
Thermocouple input module



Example of connection to thermocouple input module

Signal lines from a thermocouple can be directly connected to the input module.

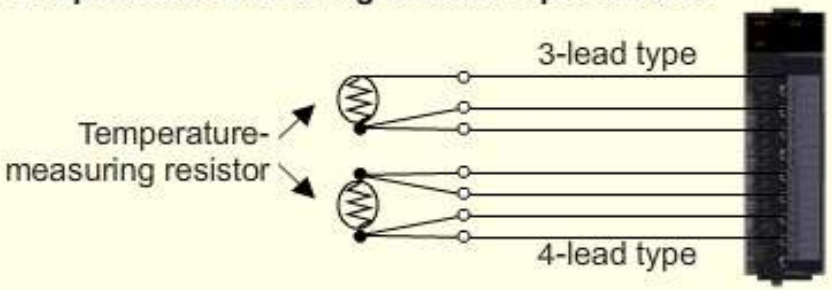
Distributor



Example of connection to distributor module

The distributor is designed to supply driving voltage via signal lines to a 2-wire transmitter.

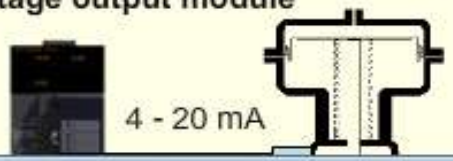
Temperature-measuring resistor input module



Example of connection to temperature-measuring resistor input module

Signal lines from a platinum/nickel temperature-measuring resistor can be connected directly to the input module.

Current/voltage output module



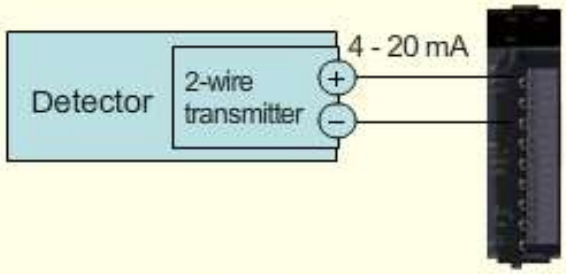
1.4.4 Additional Information - Analog Module with Isolated Channels

The following is additional information on analog input modules with isolated channels.

from a converter.

the input module.

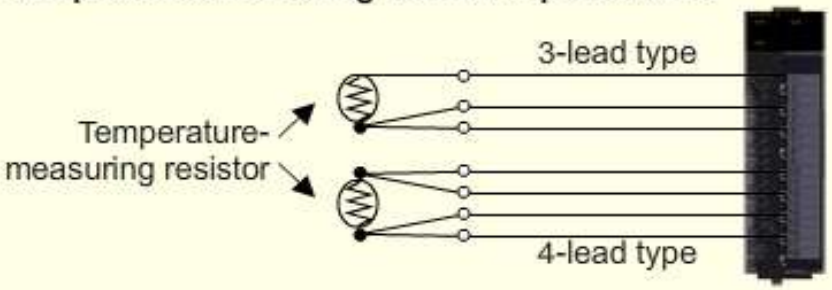
Distributor



Example of connection to distributor module

The distributor is designed to supply driving voltage via signal lines to a 2-wire transmitter.

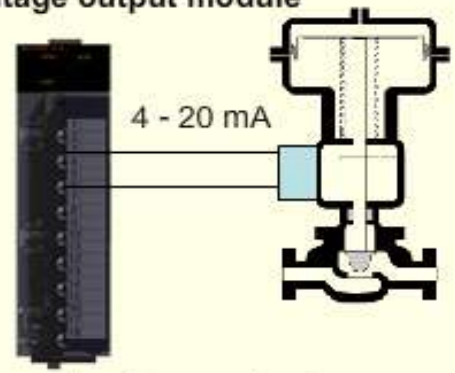
Temperature-measuring resistor input module



Example of connection to temperature-measuring resistor input module

Signal lines from a platinum/nickel temperature-measuring resistor can be connected directly to the input module.

Current/voltage output module

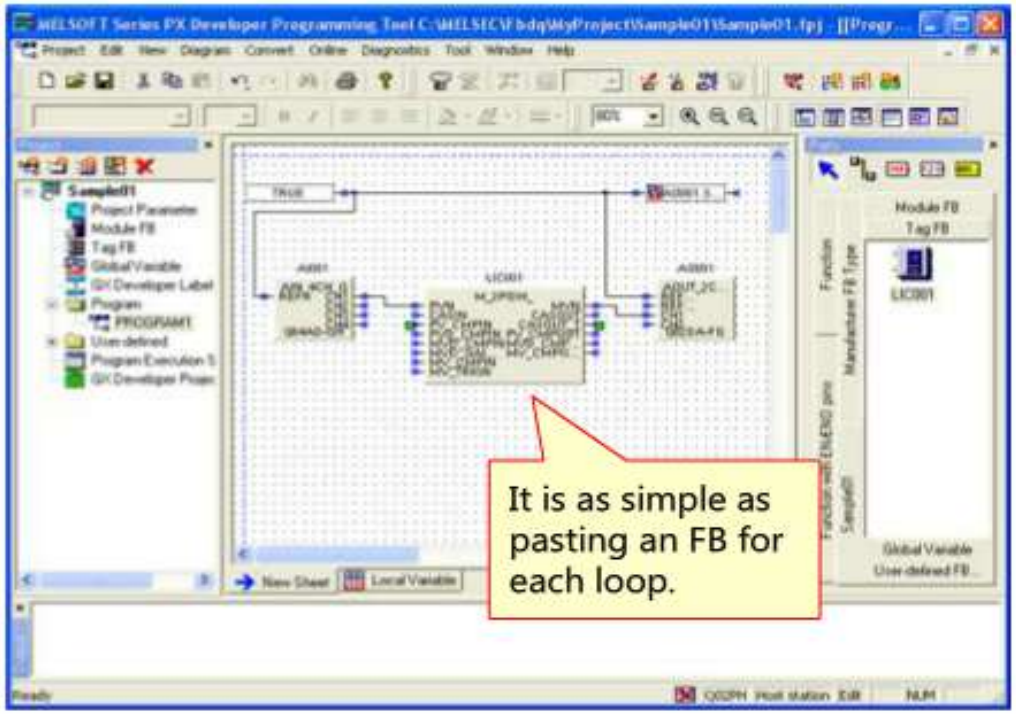


Example of connection to current/voltage output module

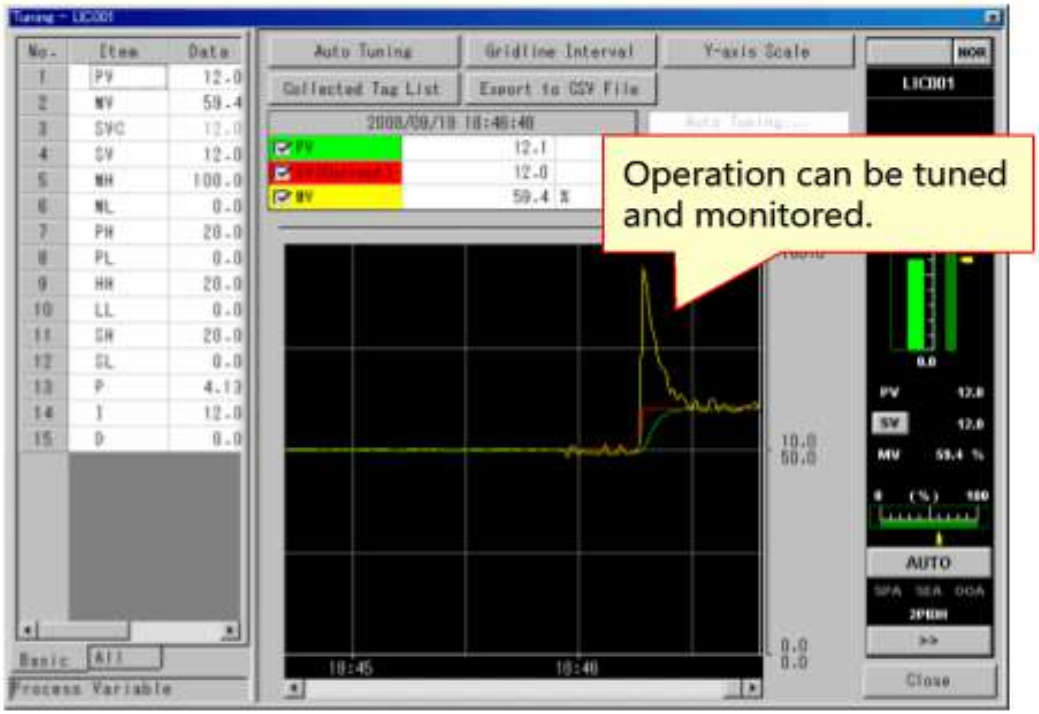
The output module is designed to send current signals of 4 - 20 mA and voltage signals of 1 - 5 V to a valve or other outputs.

1.4.5 PX Developer Software Package for Process Control System

- With PX Developer's programming tool, which meets the IEC61131-3 standard, loop control can be programmed easily, simply by pasting FBs and connecting wires. This reduces the time taken to build a process control system.
- The monitoring tool comes as standard with frequently-used features such as tuning, control panel, trend graph and a warning list. Once programming is complete, you can immediately move on to adjustment, startup and operation.



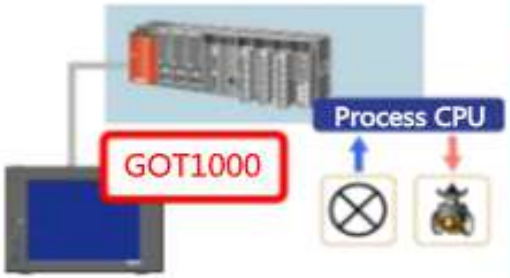
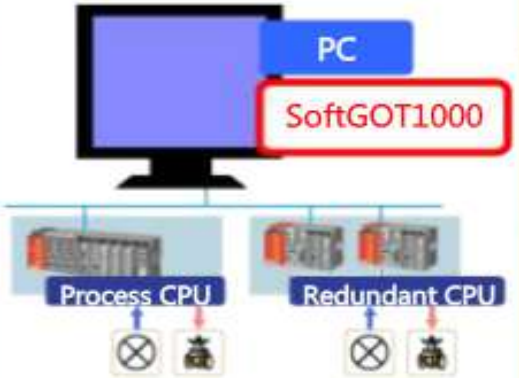
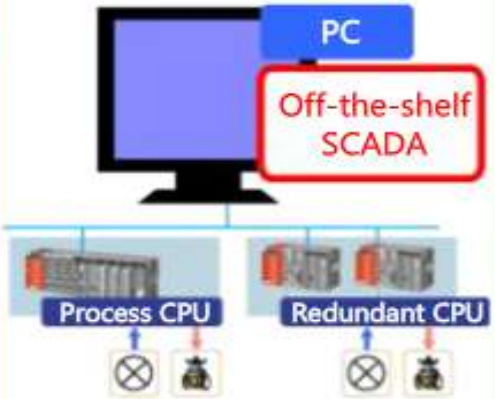
Programming tool



Monitoring tool

1.4.6 Process Control System Monitoring

The MELSEC process control system offers a range of monitoring solutions to suit all possible system sizes, whether it is just one device or a whole plant.

Type		Device/site monitoring solution	Facility/plant monitoring solution	Plant monitoring solution
Structure		GOT imaging function Coordinated indicator monitoring 	PC monitoring by coordination between PX Developer monitoring tool and SoftGOT1000 	PC monitoring by coordination between PX Developer monitoring tool and off-the-shelf SCADA 
Function	Graphic screen image	GOT1000 drawing software [GT Designer2]		Off-the-shelf SCADA
	Standard screen image	Automatically generated by GOT imaging function	Generated by PX Developer monitoring tool	Available using ActiveX faceplate components on the off-the-shelf SCADA

*1 Faceplates, tuning screen and other images of the PX Developer monitoring tool are automatically converted into GT Designer2 image data. This data can then be used for GOT without further processing.
*2 Images are available by pasting ActiveX faceplate components onto SCADA graphic screen images.

Chapter 2 System Configuration

In this chapter, you will look at a process control system that controls the water level of a tank, and explore the required configuration and software of the programmable controller.

2.1 System Building Procedures

In this example we will build a process control system that maintains the water level in a tank.

Step 1: System designing

Select a CPU and input and output modules that are suitable for the application, and design control algorithm..

Step 2: FBD programming

Write an FBD program using PX Developer.

Step 3: FB property setting

Compile the FBD program and write the program and properties to the programmable controller CPU.

Step 4: Program compilation and writing to programmable controller CPU

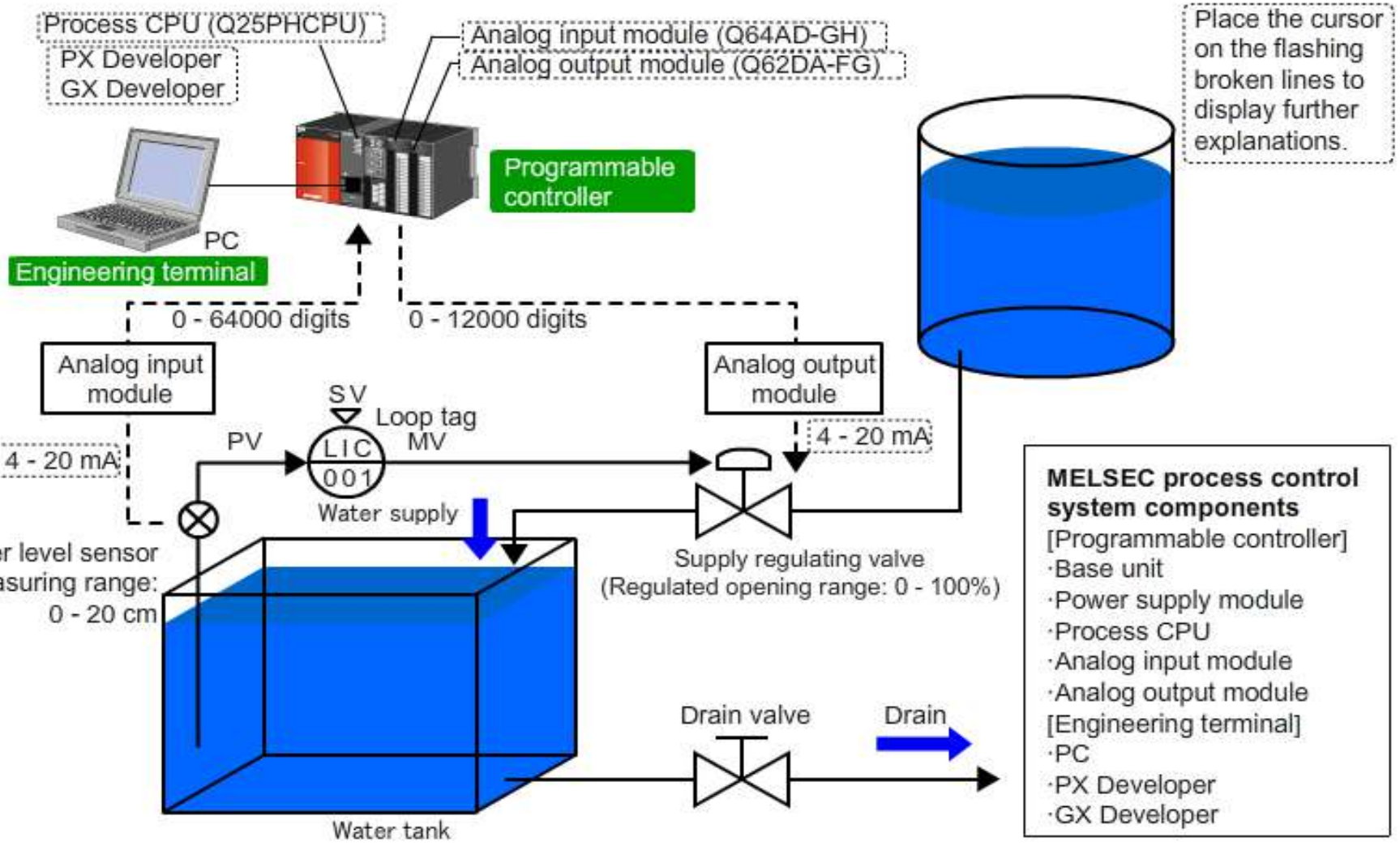
Compile the FBD program and write the program and properties to the programmable controller CPU.

Step 5: System tuning and test operation

Tune and test-drive the system (PID control).

2.2 System Structure

You will build the MELSEC process control system shown below to keep the water level of a tank at a predetermined level. When the water level in the tank drops due to the drain valve being activated the drop in the water level is detected by a water level sensor. The PID control program responds by activating the supply regulating valve. The structure of this MELSEC process control application is as follows.



2.3 Additional Information - Loop Control Tag Number

Tags are allocated to the components and functions of the process control system to identify control loop process characteristics. These tags are called loop control tag numbers.

Example: T I C 0 0 1

Variable code

- Represents process variables.
  Example: T = temperature

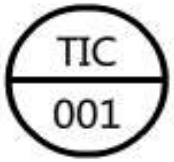
Function code

- Represents instruction, adjustment, warning and other functions.
  Example: I = instruction;
  C = adjustment

Individual number

- Used to identify measurements and control loops.
  Example: 001

"TIC001" indicates loop number 001 for temperature instruction and adjustment.



Symbol for TIC001

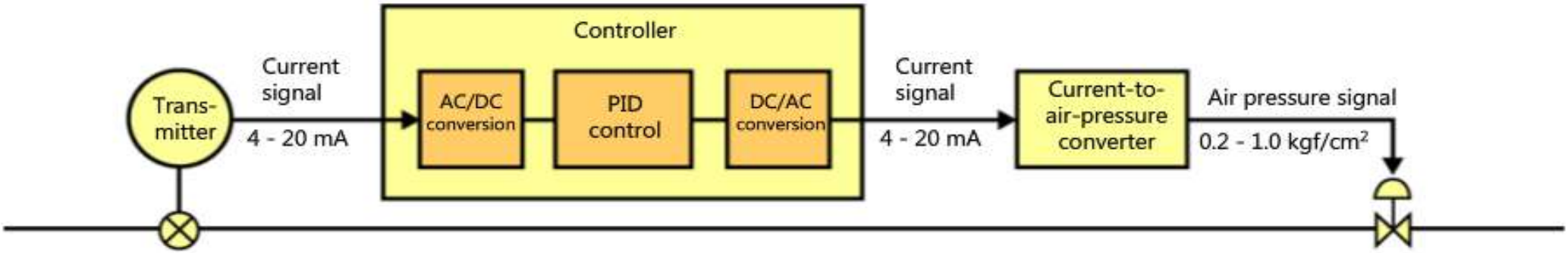
	Variable code	Function code
A		Warning
C		Adjustment
D	Density, Specific gravity	
F	Instantaneous flow rate	
G	Position, Length	
H	Manual operation	
I		Instruction
K	Time	
L	Fluid and other levels	
M	Humidity, Moisture content	
P	Pressure, Vacuum	
Q	Quality (composition, concentration)	Integration
R	Radiation	Record
S	Velocity, Speed, Frequency	Switch
T	Temperature	Transmission
V	Viscosity	
W	Mass, Force	
Z		Safety, Emergency

Frequently used code

2.4 Additional Information - Standardized Signals

Input and output signals for process control systems, such as measurements and actuation commands, are standardized (typically 4 - 20 mA DC). These signals are called **standardized signals**.

Signal type	Signal range
Current	4 - 20 mA DC
Voltage	1 - 5 V DC
Air pressure	0.2 - 1.0 kgf/cm^2



2.5 Input and Output Modules

The input and output modules for the process control system are shown in the following table. This information is required in Step 2 "FBD programming" and Step 3 "FB property setting."

Module/Device	Slot	Head I/O address	Connection	Range
Analog (current/voltage) input module (Q64AD-GH)	I/O 0	0000	The input signal line from the water level sensor is connected to the channel 1 (CH1) input terminal of the module.	Analog input signal range: 4 - 20 mA Digital output signal range: 0 - 64000
Analog (current/voltage) output module (Q62DA-FG)	I/O 1	0010	The output signal line to the supply regulating valve is connected to the channel 1 (CH1) output terminal of the module.	Digital input signal range: 0 - 12000 Analog output signal range: 4 - 20 mA

Place the cursor on the flashing broken lines to display an arrow.

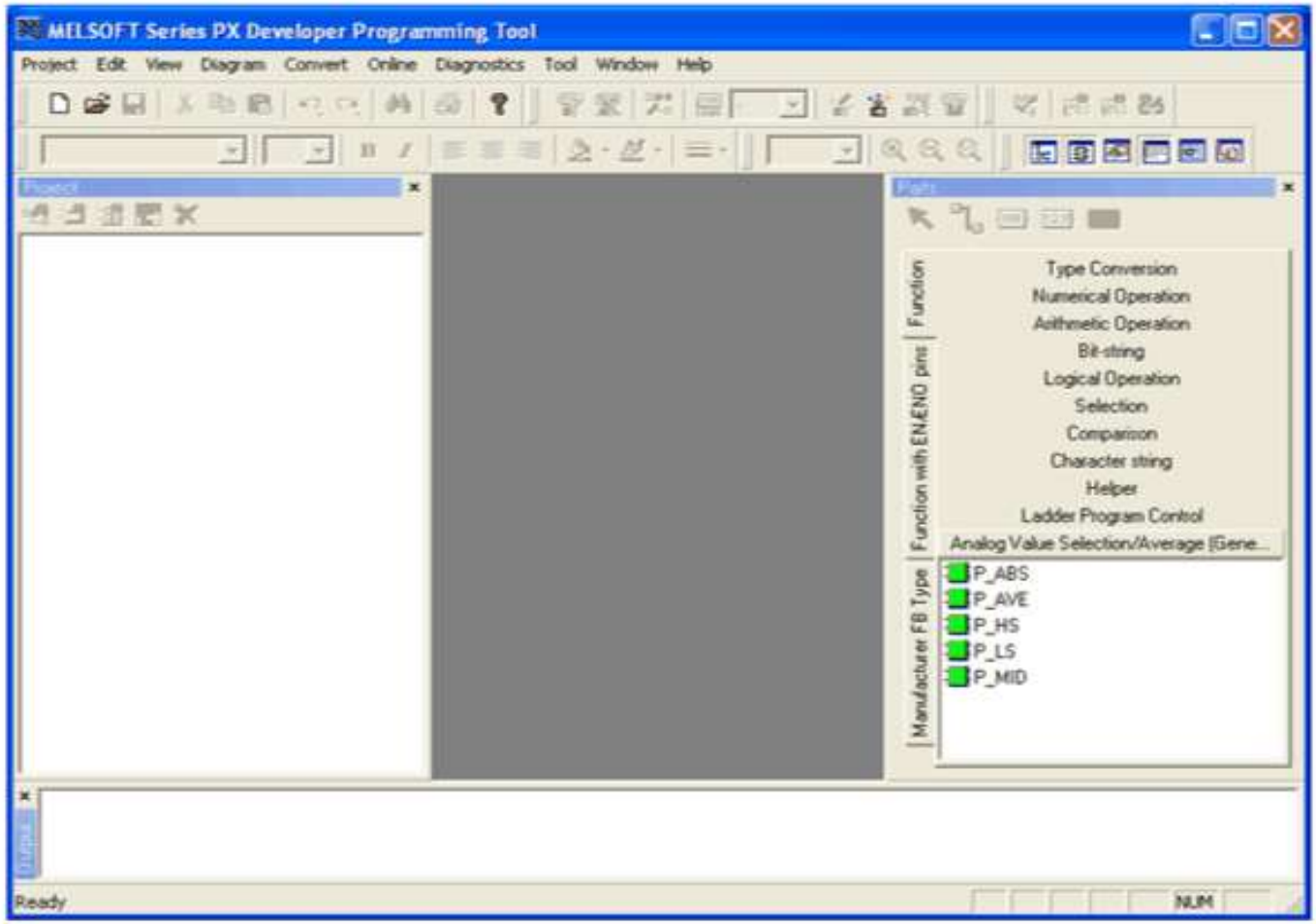
Chapter 3 FBD Programming

In this chapter, you will write FBD programs using the PX Developer programming tool.

3.1 Starting the PX Developer Programming Tool

To perform FBD programming, start the PX Developer programming tool, which is application software.

Click the **Start** menu of Windows, **All Programs** and then **PX Developer Programming Tool** to start the application software.



3.2 Creating New Projects

To write programs using the programming tool, you need to create a project.
This requires you to set certain items.

(1) Programmable controller model
Specify a programmable controller CPU.
It can be a process CPU or a redundant CPU.
In this training course, you will configure the system with a process CPU (Q25PH).
Select **Q25PH**.

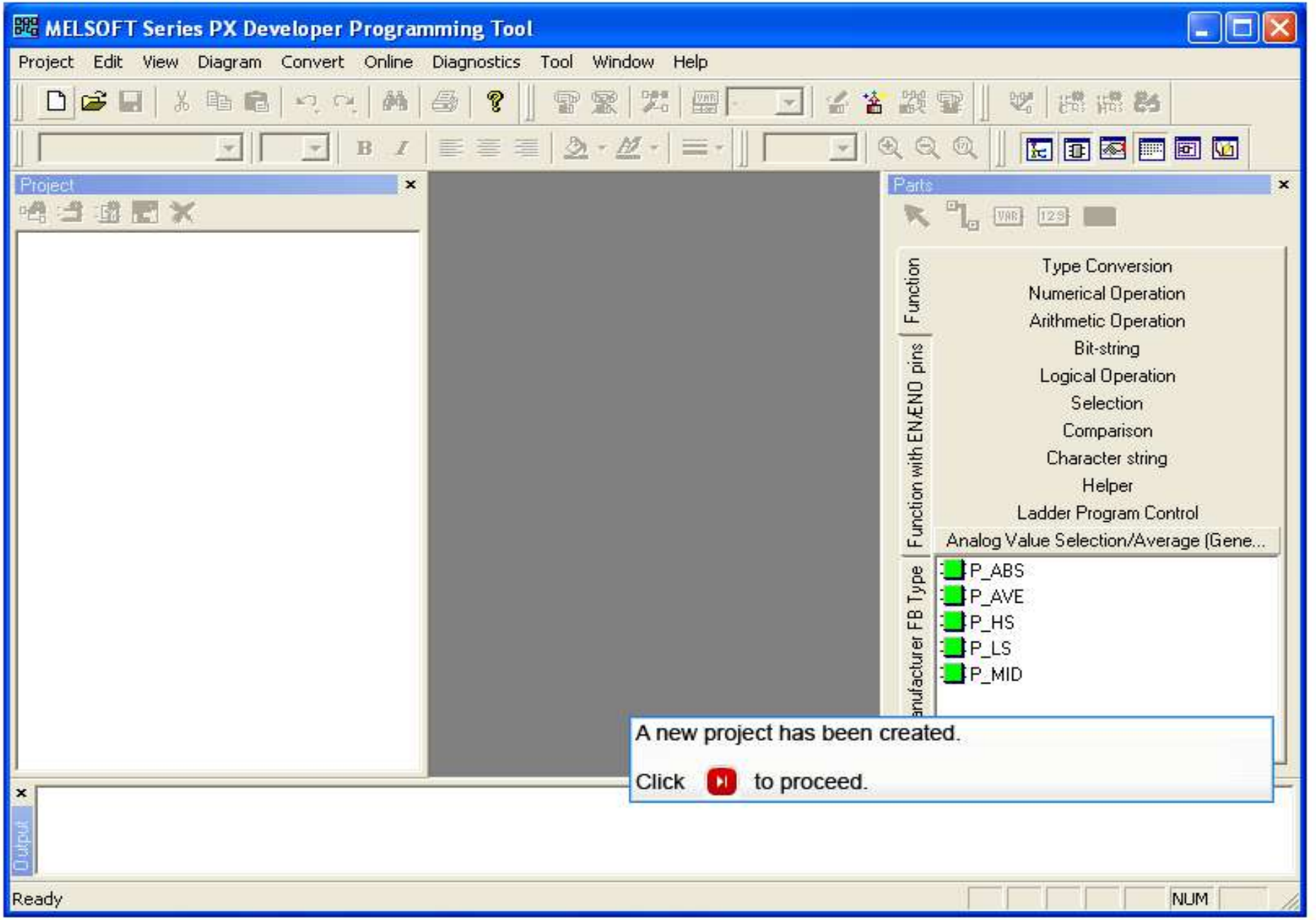
CPU type	Programmable controller model
Q02PH	Process CPU
Q06PH	
Q12PH	
Q25PH	
Q12PRH	Redundant CPU
Q25PRH	

(2) Project name
Specify the drive/path to which you want to save the project file and the project name.
In this training course, enter the following.

Drive/Path: c:\MELSEC\Flodq\MyProject
Project name: Sample01

* When a project name is specified, a folder with the project name is automatically created in the specified drive/path.

3.2 Creating New Projects



3.3 Screen Layout of the PX Developer Programming Tool

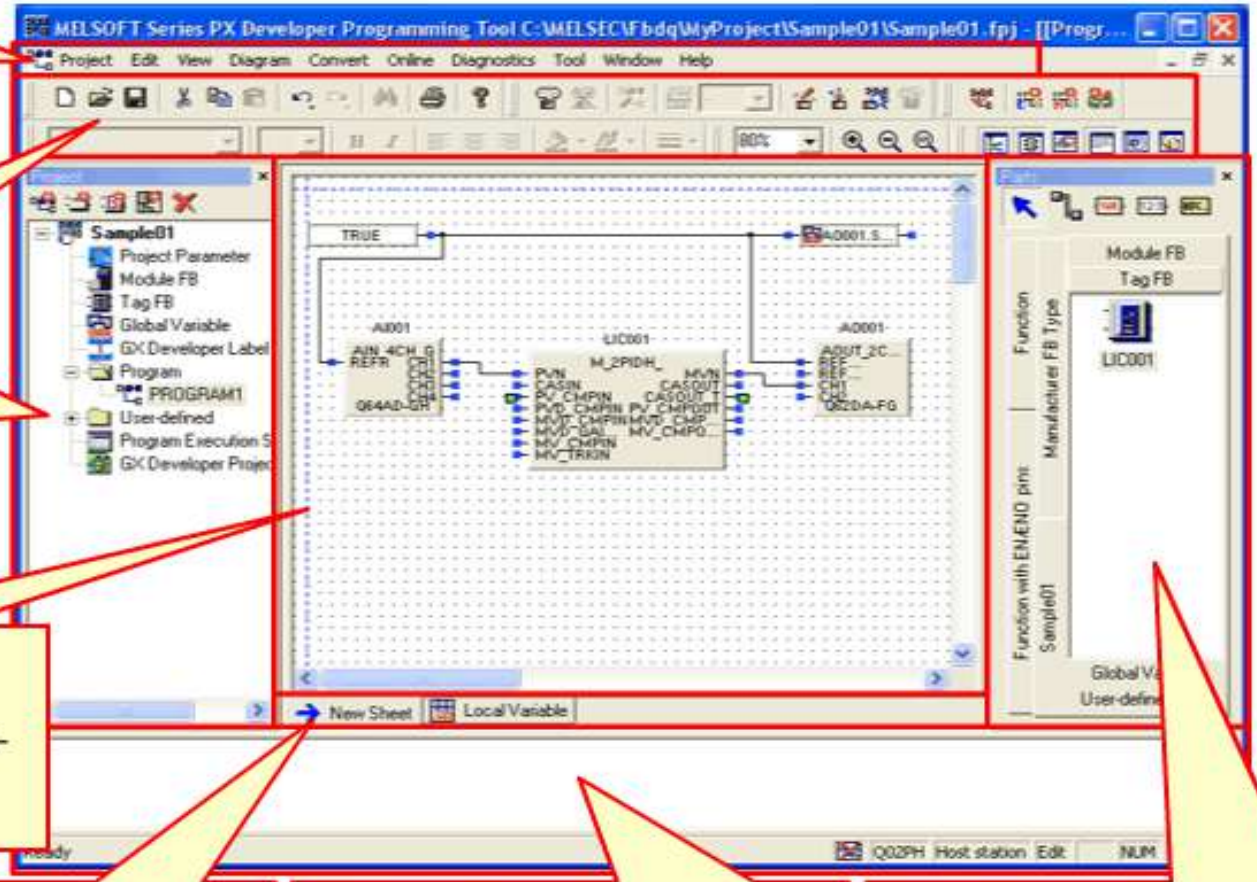
The screen of the PX Developer programming tool is laid out as shown below.

[Menu Bar]
To execute functions/operations.

[Tool Bar]
To execute frequently used functions/operations.

[Project Window]
To set project parameters, modules, tags, program names, program run frequencies, etc.

[Programming Window]
To create programs and user-defined FBs. Up to 200 process-description programs can be created.



[Sheet Tab]
To select sheets. Up to 32 sheets can be created per program.

[Output Window]
To display information such as the progress of process, errors and warnings during compilation and execution of other commands.

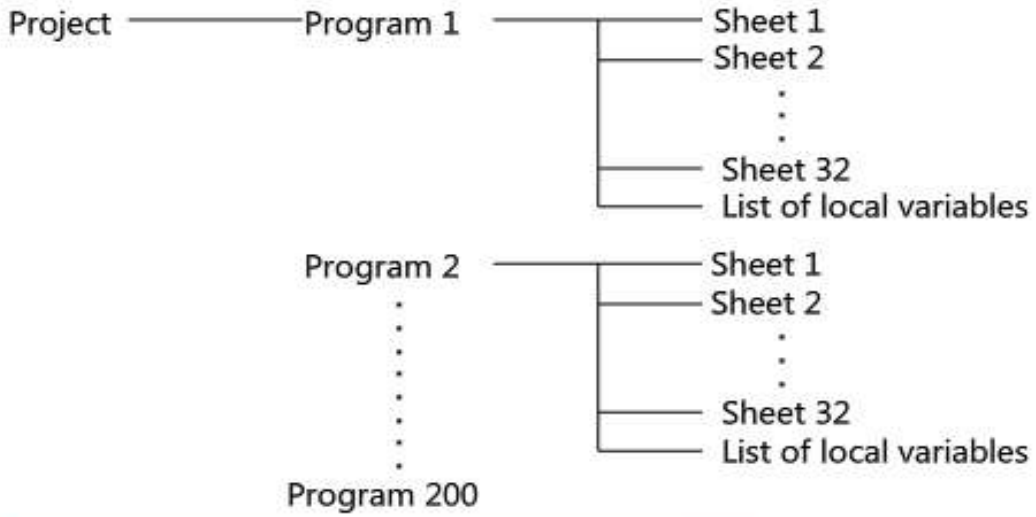
[Parts Window]
To show a list of FBs/functions that can be pasted to programs and user-defined FBs.

3.3.1 Additional Information - FBD Program Structure and Processing Sequence

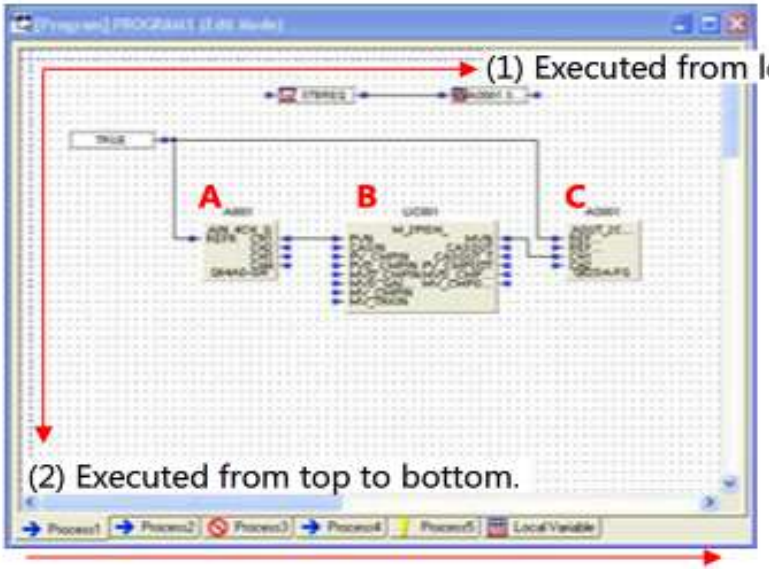
The following shows the structure of FBD programs and the processing sequence that are available with PX Developer.

As shown on the right, multiple programs can be created for a project, with each program being able to accommodate a maximum of 32 program sheets.

(For details, please see the user's manual for PX Developer.)



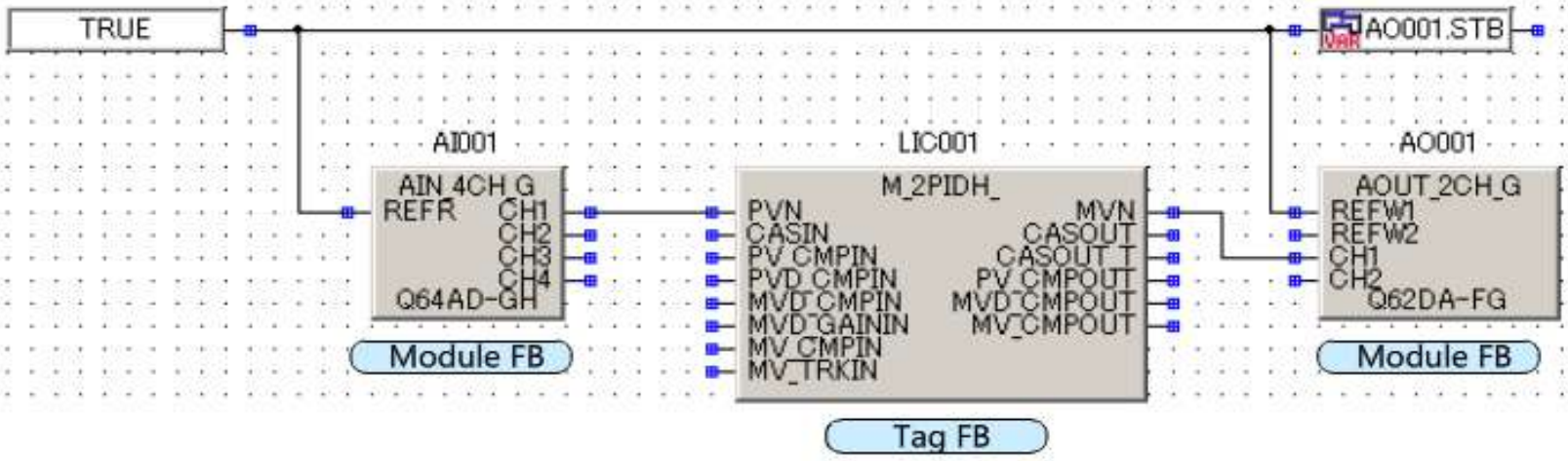
FBD parts that are inserted, arranged and connected on the sheet are executed in the order of (1), (2) and (3) as shown in the illustration on the right.
The FBD parts shown in the illustration are executed in the order of A, B and C.



3.4 Creating FBD Programs

3.4.1 Program to Be Created

In this training course, the following water level control program will be created.



FBD program

PV (process variable) is sent from a module FB representing the analog input module (Q64AD-GH) to a tag FB, which then performs computation. The results of the computation, or MV (manipulated variable), are sent out to a module FB representing the analog output module (Q62DA-FG).

The program loop tag is a 2-degree-of-freedom high-performance PID control tag FB (M_2PIDH_), which can accommodate a wide range of applications with its rich range of functions.

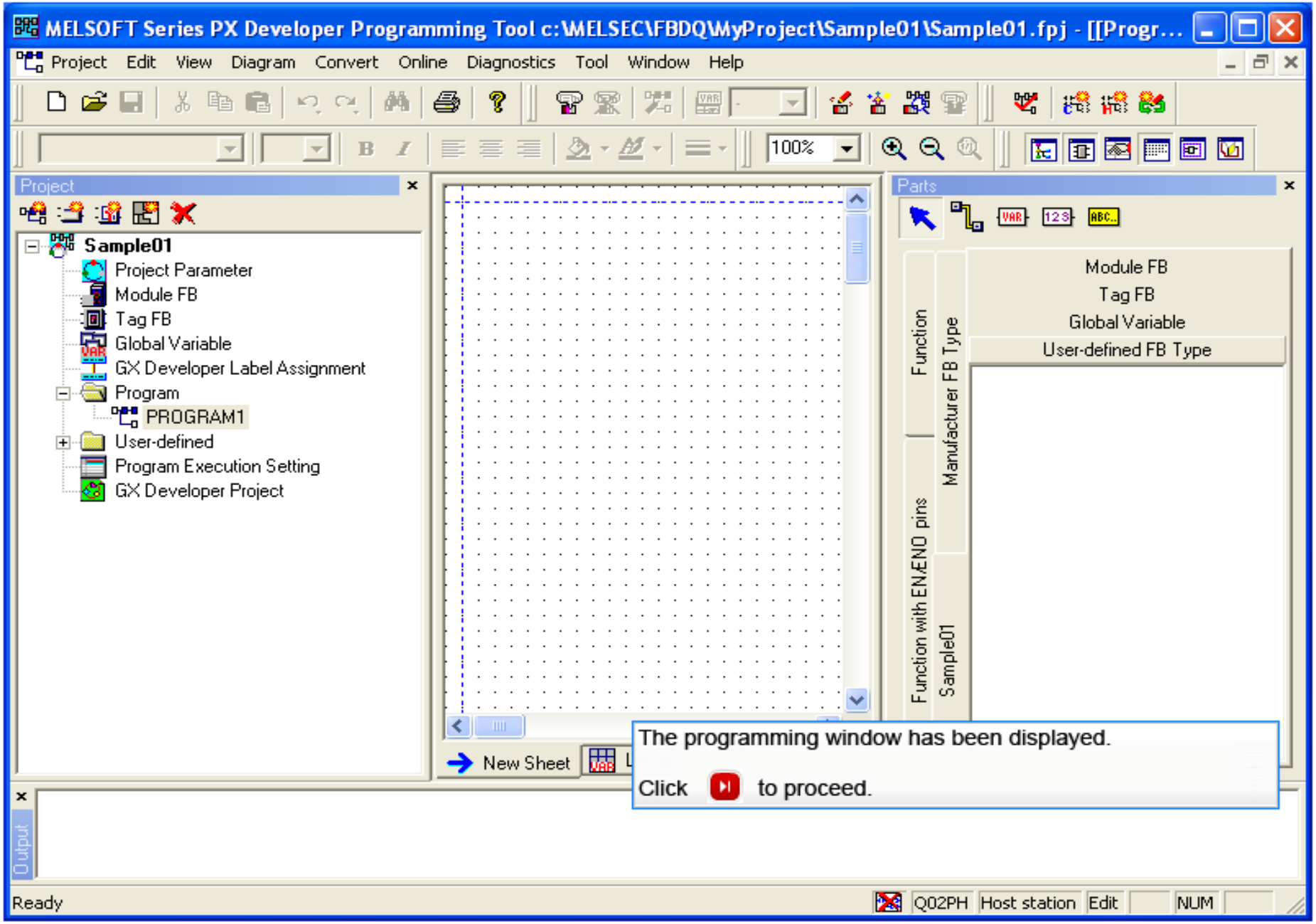
3.4.2 Displaying the Programming Window

To create an FBD program, you need to display the programming window.
In this training course, you will create an FBD program on the following sheet.

Program name: Program 1
Sheet name: New sheet

* Program 1 and a new sheet will be automatically created as you create a new project.

3.4.2 Displaying the Programming Window



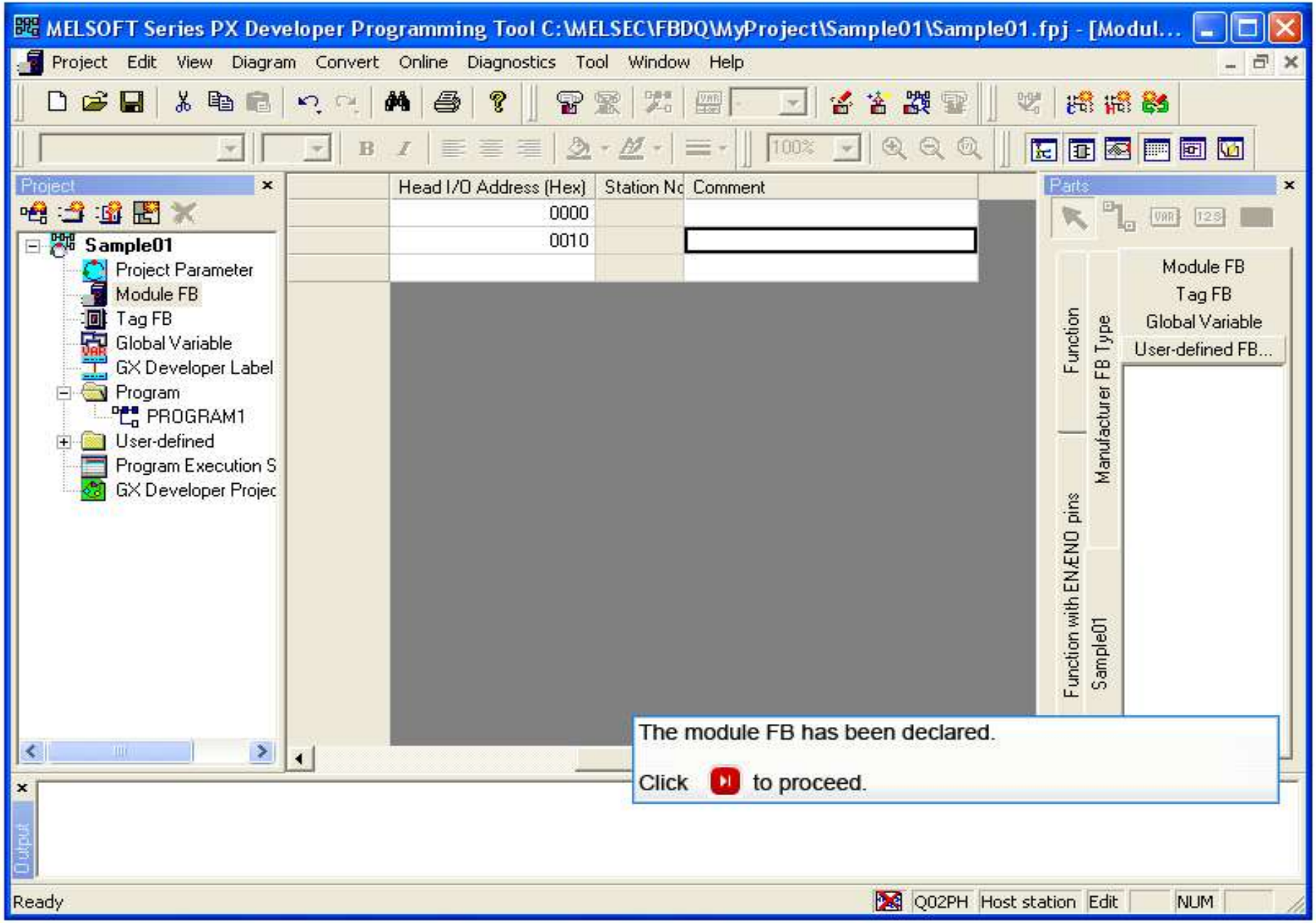
3.4.3 Declaring Module FBs

To access the input and output modules (Q64AD-GH and Q62DA-FG) from the program, declare (register) module FBs representing these modules in the module FB declaration window.
In the module FB declaration window, set the following items.

Module FB variable name	Module model	Module FB type	Start I/O address
AI001	Q64AD-GH	AIN_4CH	0000
AO001	Q62DA-FG	AOUT_2CH	0010

* Selecting a module model automatically sets a corresponding module FB type.

3.4.3 Declaring Module FBs

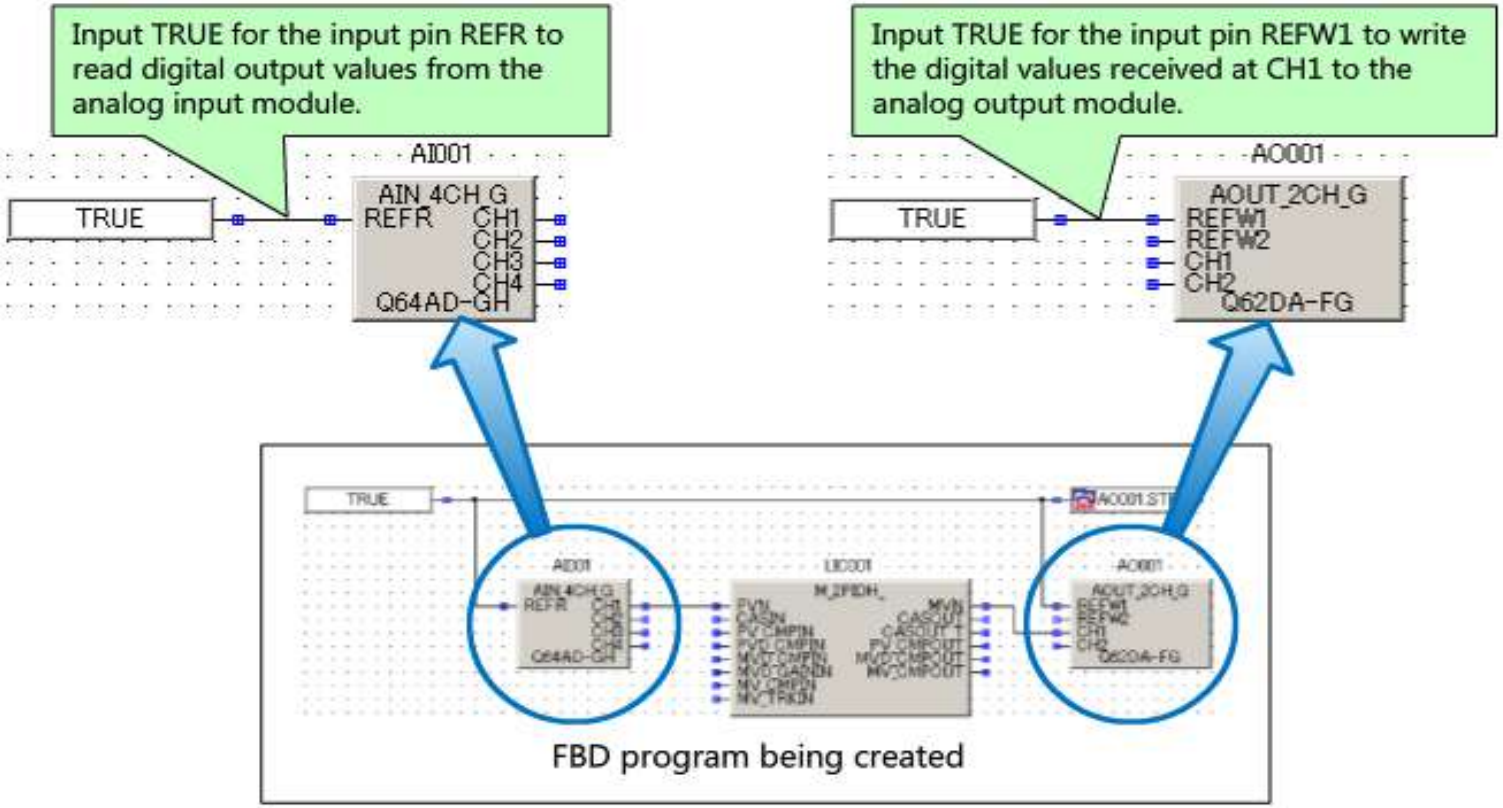


3.4.4 Pasting Module FBs

The module FBs (AI001 and AO001) that have been declared in the module FB declaration window need to be pasted to the programming window. After that, follow the procedure below to enable the module FBs.

(1) Enabling the AI001 (Q64AD-GH) for output and the AO001 (Q62DA-FG) for input

Input TRUE for REFR and REFW1 to enable the AI001 output pin and the AO001 input pin on the FBD program.



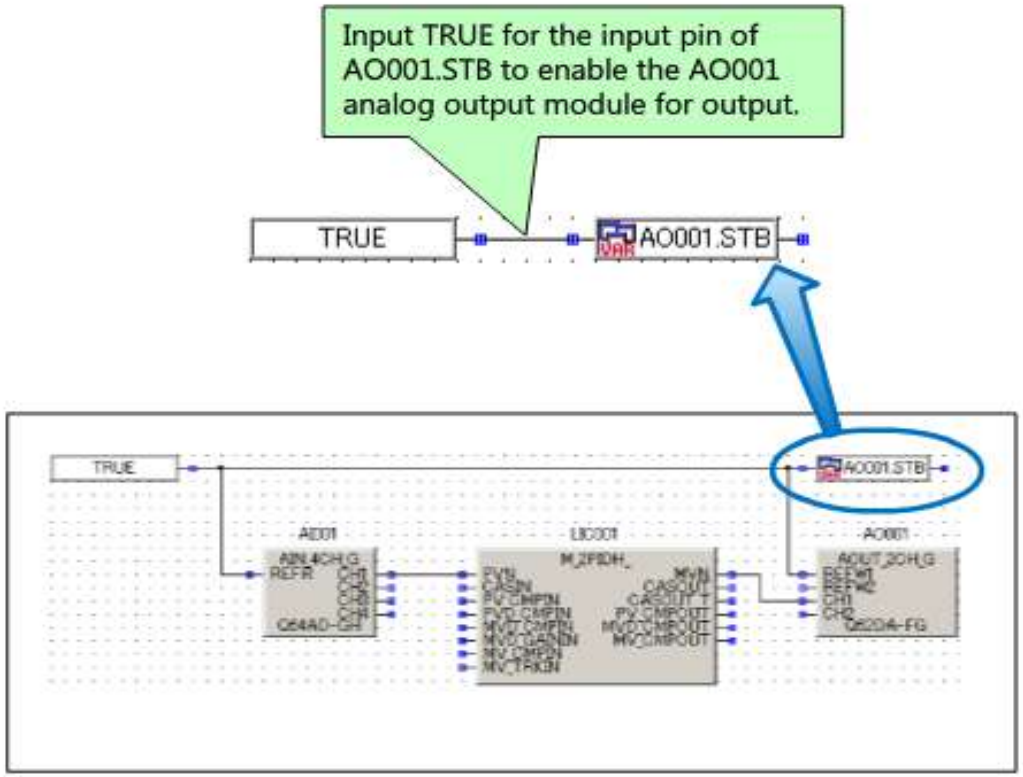
In order to achieve the above, paste TRUE constants to the programming window and connect them to the following two input variables (pins).

Module FB	Variable name	Variable type	Data type	Description
AI001	REFR	Input variable	BOOL	Output condition signal. Executed by TRUE.
AO001	REFW1	Input variable	BOOL	Input condition signal for CH1. Executed by TRUE.

3.4.4 Pasting Module FBs

(2) Enabling the AO001 (Q62DA-FG) for output

Input TRUE to AO001.STB, which is a public variable, to enable the analog output module FB AO001 for analog output.

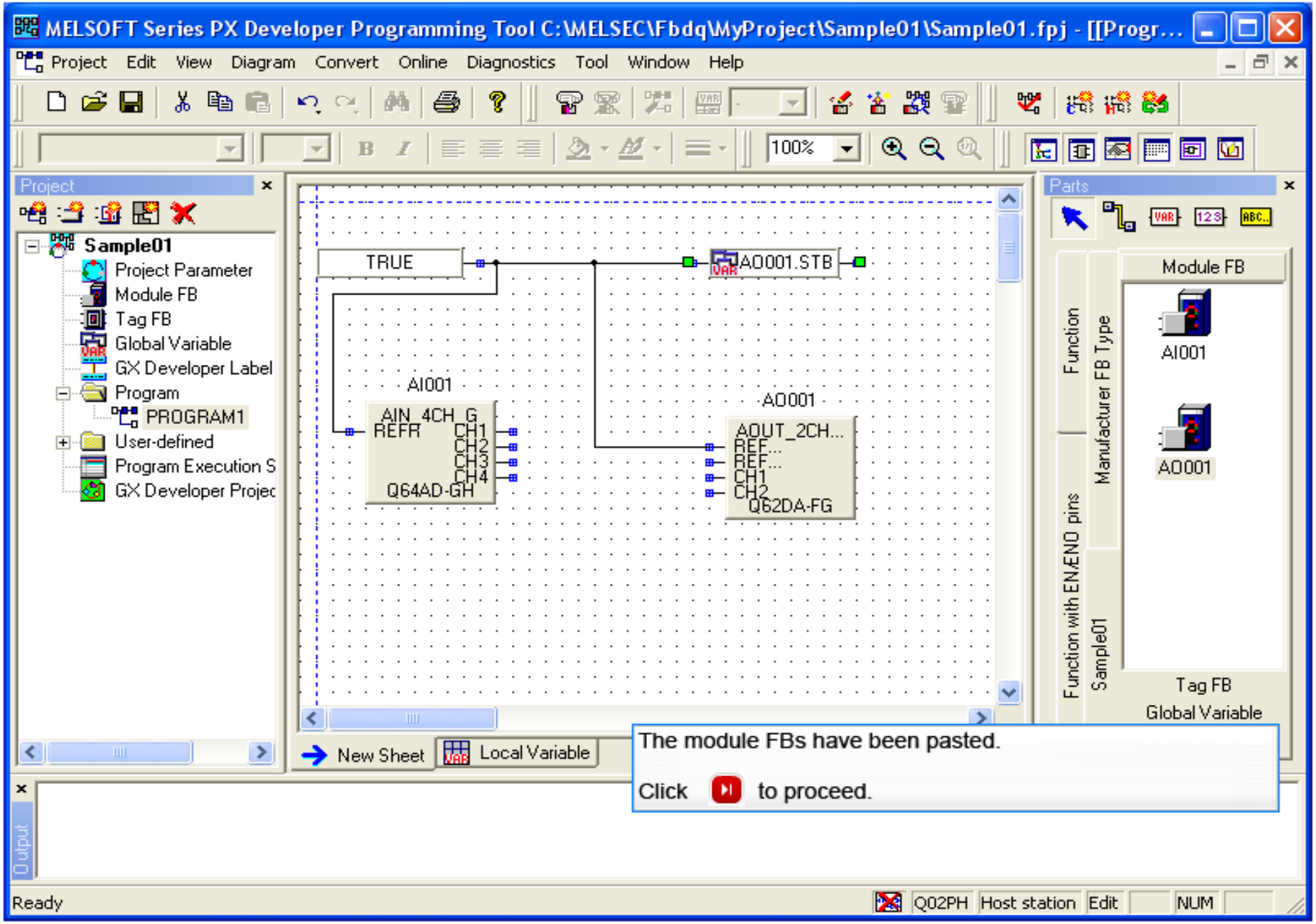


In order to achieve the above, paste TRUE constants to the programming window and connect them to the following two input variables (pins).

Variable name	Variable type	Data type	Description
AO001.STB	Public variable	BOOL	Operating condition setting request Executes D/A conversion enable/disable setting upon switching from FALSE to TRUE.

*Because AO001.STB is the public variable of the AO001, there is no need to specify the variable type when creating the variable.

3.4.4 Pasting Module FBs



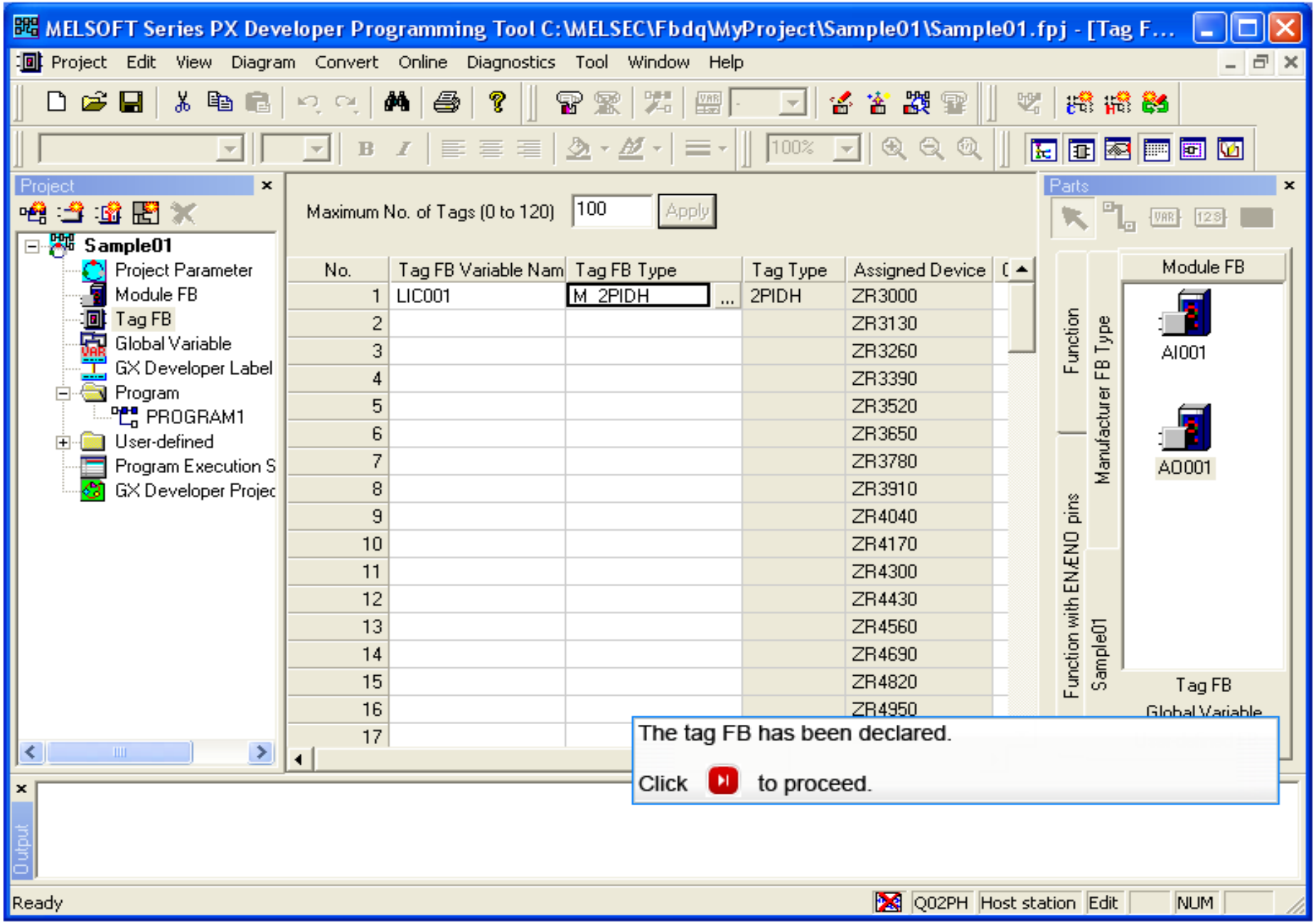
3.4.5 Declaring Tag FBs

A 2-degree-of-freedom high-performance PID control tag FB (M_2PIDH_) needs to be registered in the tag FB declaration window to enable PID control. In the tag FB declaration window, set the following items. Because the tag FB is used to give instructions and control water level, the tag FB variable is named LIC001.

Tag FB variable name	Tag FB type	Tag type
LIC001	M_2PIDH_	2PIDH

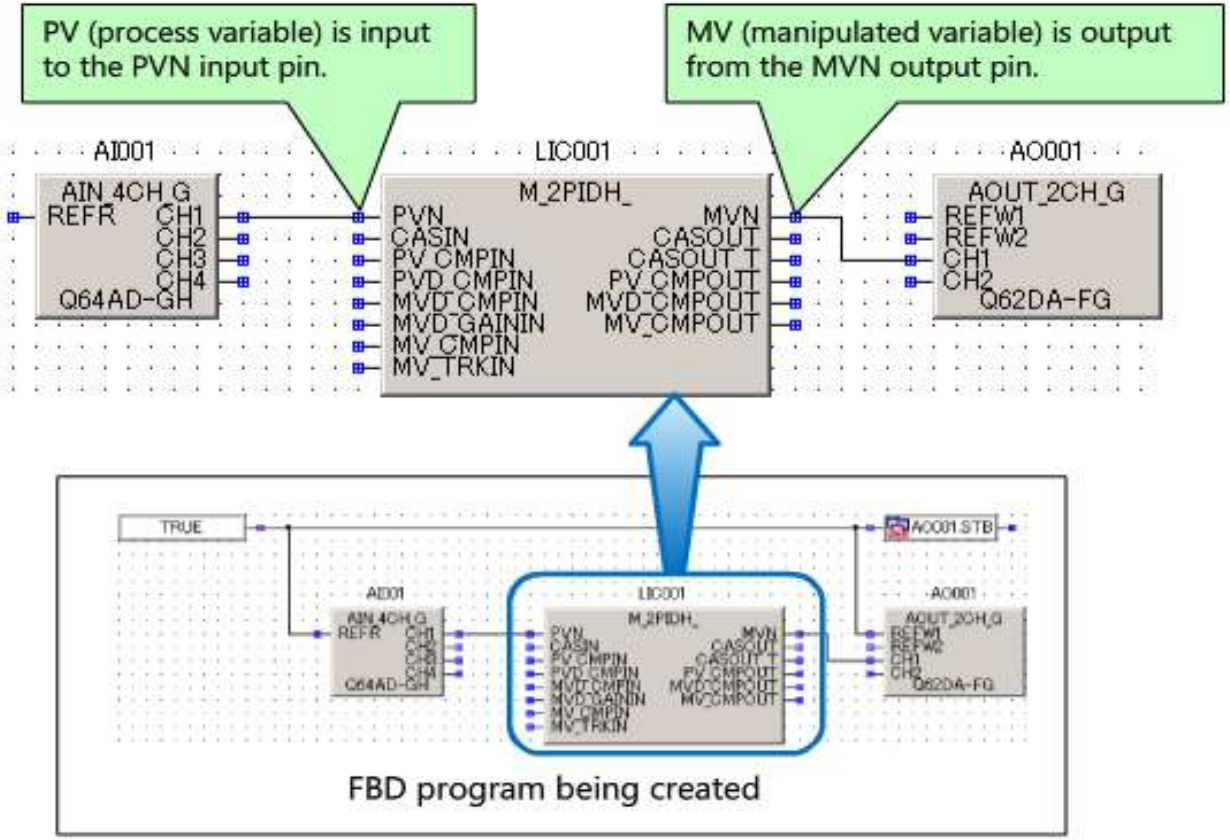
* The tag type is automatically set.

3.4.5 Declaring Tag FBs



3.4.6 Pasting Tag FBs

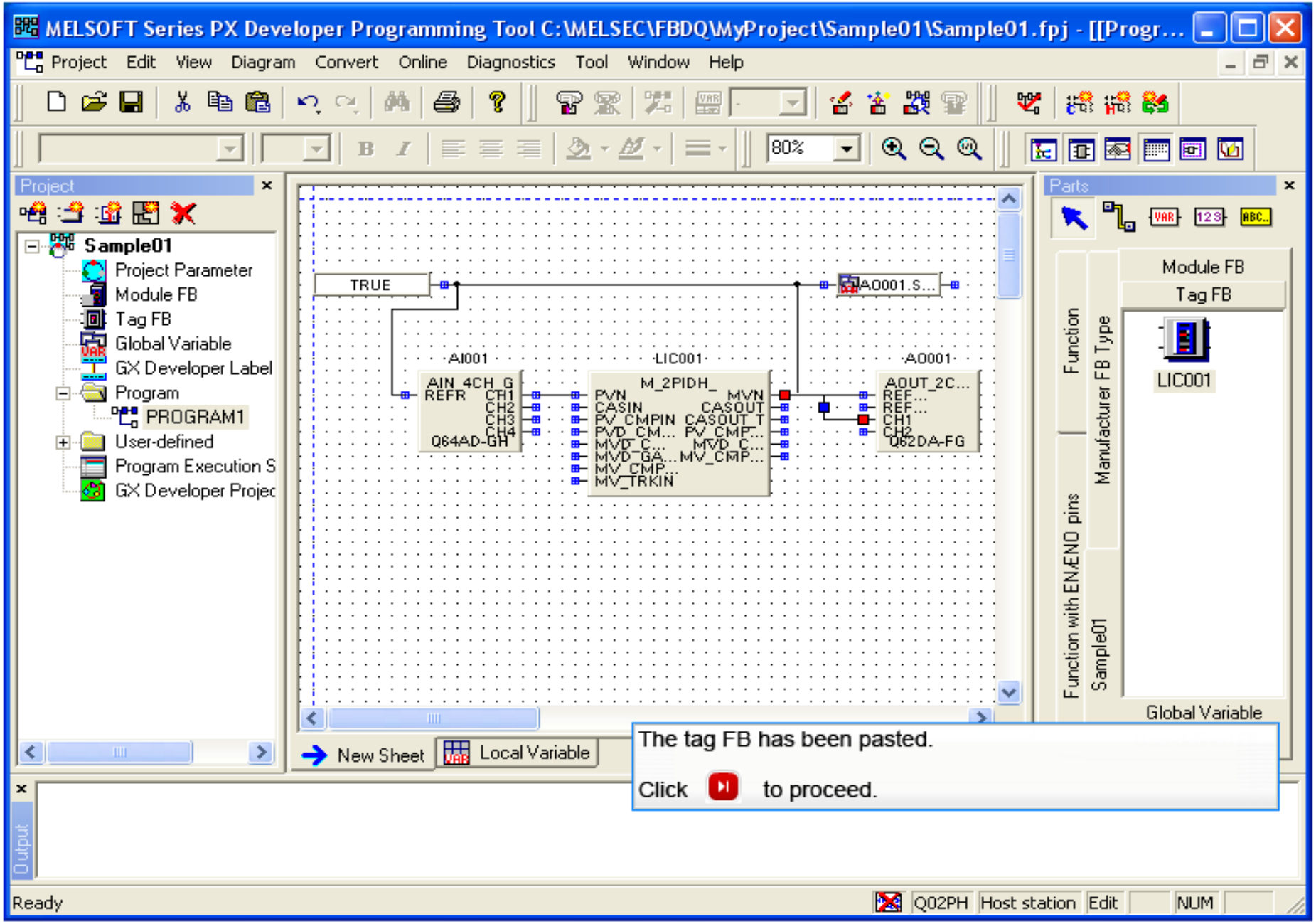
Paste the tag FB (LIC001), which has been declared in the tag FB declaration window, to the programming window. Connect the PVN pin for process variable input and the MVN pin for manipulated variable output to the input/output pins of the two module FBs that have been pasted to the window.



As shown below, connect CH1 of the analog input module to PVN and CH1 of the analog input module to MVN.

Output pin			Input pin	
Tag/module variable name	Pin name		Tag/module variable name	Pin name
AI001	CH1	→	LIC001	PVN
LIC001	MVN	→	AO001	CH1

3.4.6 Pasting Tag FBs

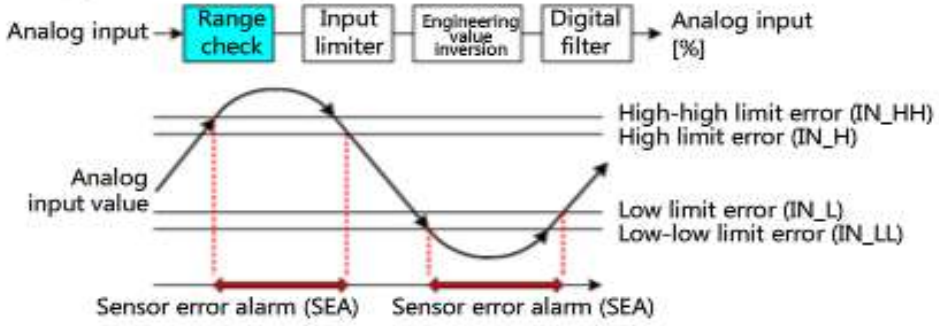


3.4.7 Setting the FB Property Initial Values

Set the initial values such as for input and output ranges of tag FBs according to the input/output characteristics of a controlled device.

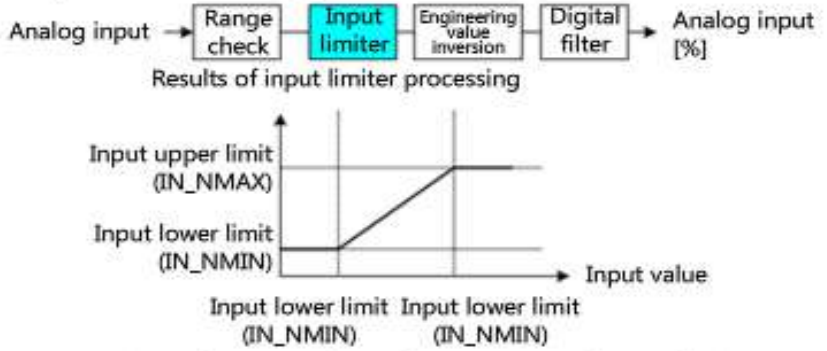
First, the setting methods for the range check, which detects errors of analog sensor input from a detector, and input limiter are described.

[Range check setting]



Range errors occur in the sensor error alarm areas.

[Input limiter setting]



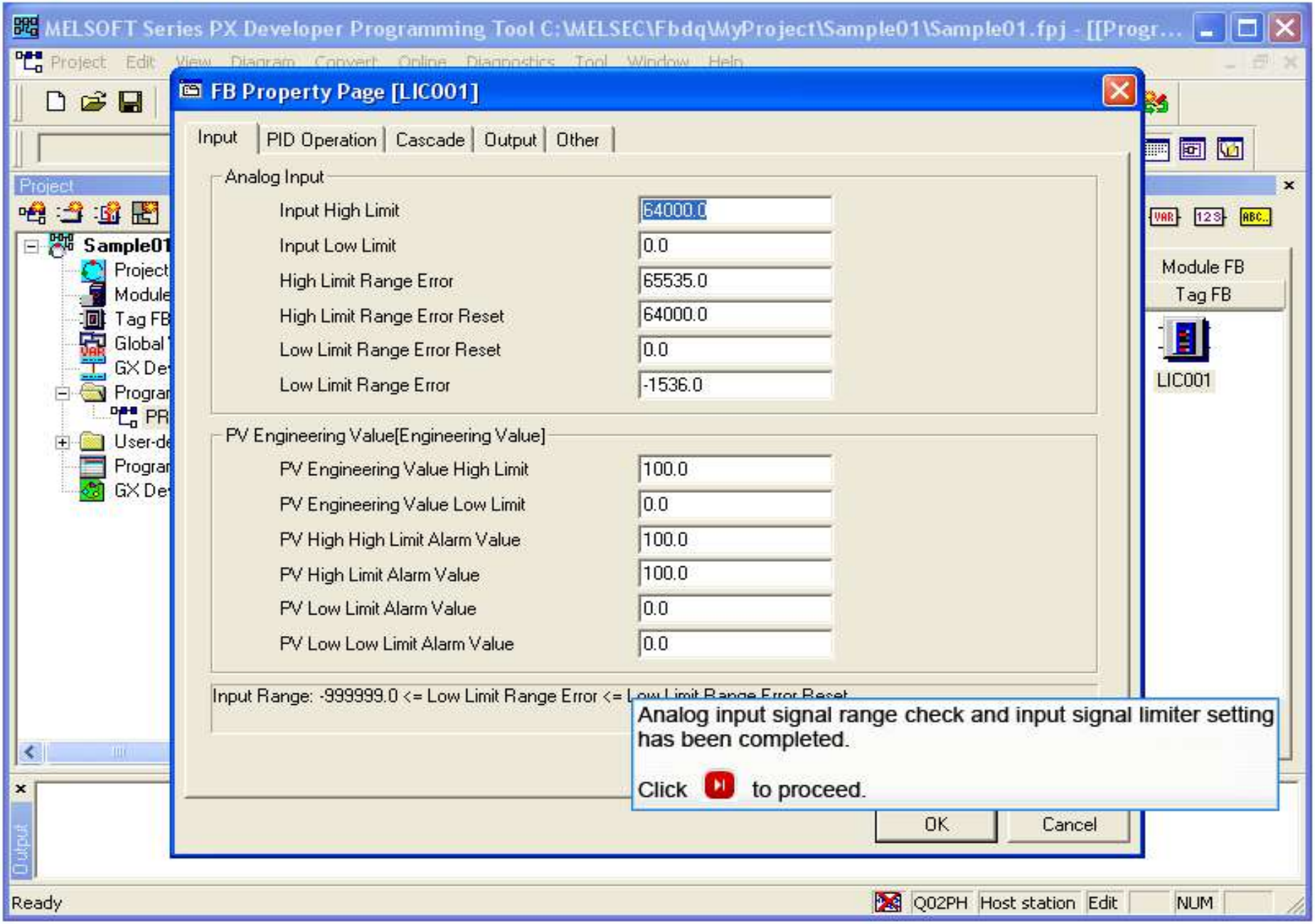
Input signals exceeding the upper or lower limit are eliminated by the input limiter.

Because the analog input module used in this course has a digital output range of 0 to 64000, the upper and lower limits of the limiter are set at 64000 and 0 respectively.

Setting item for analog input	Setting value	Description
High-high limit error	65535.0	An error occurs when the analog input value reaches 65535 or above.
High limit error	64000.0	The normal state is restored when the analog input value drops to 64000 or below.
Low limit error	0.0	The normal state is restored when the analog input value rises to 0 or above.
Low-low limit error	-1536.0	An error occurs when the analog input value drops to -1536 or below such as when the sensor circuit opens.
Input upper limit	64000.0	The Q64AD-GH module has a digital output range of 0 to 64000 for conversion of the analog input range of 4 to 20 mA.
Input lower limit	0.0	

* Out-of-range error thresholds, or setting values, vary depending on the module type.

3.4.7 Setting the FB Property Initial Values



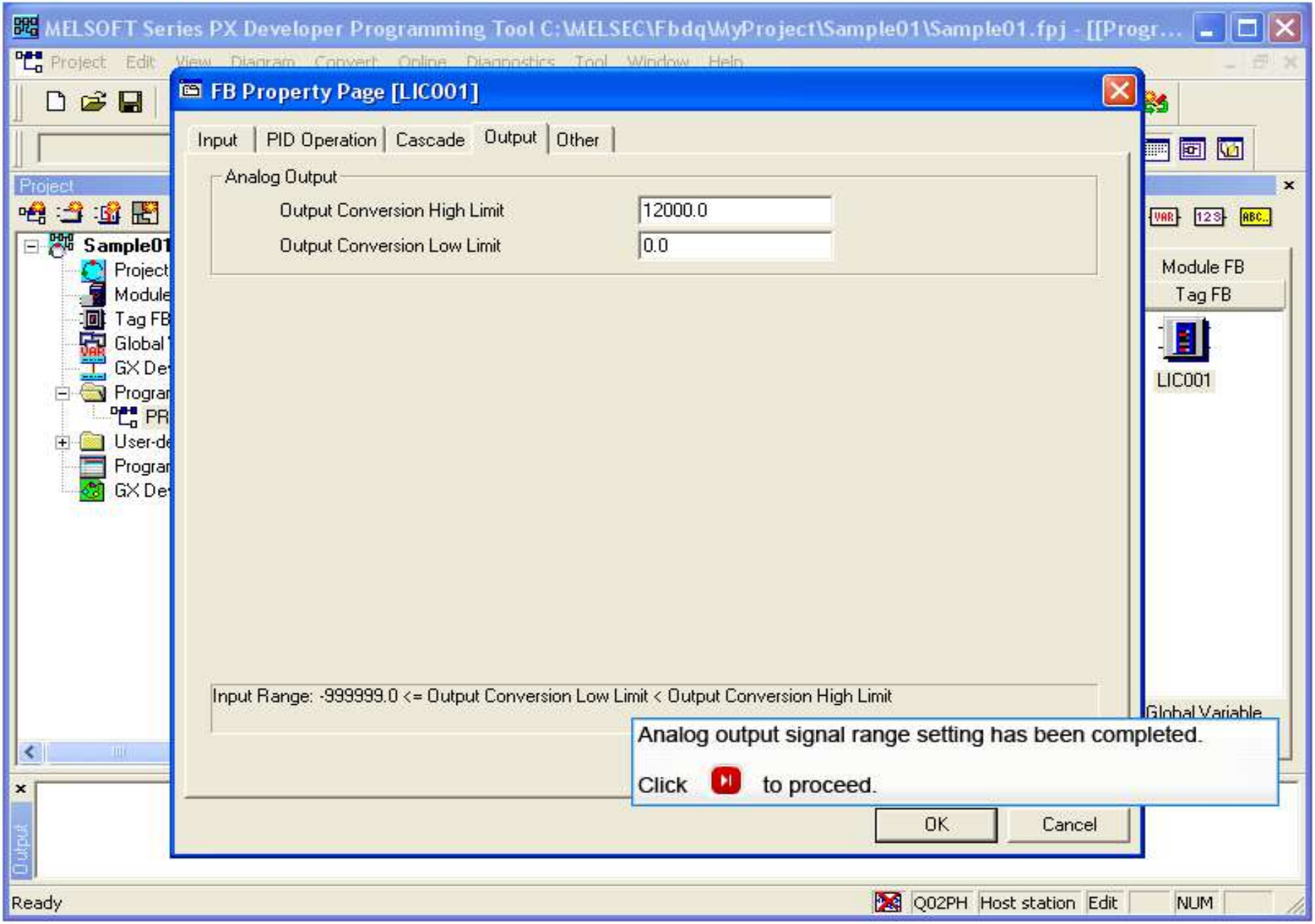
3.4.7 Setting the FB Property Initial Values

The next setting concerns the range of analog output to the final controlling element.

Because the analog output module used in this course has a digital input range of 0 to 12000, the upper and lower limits of the limiter are set at 12000 and 0 respectively.

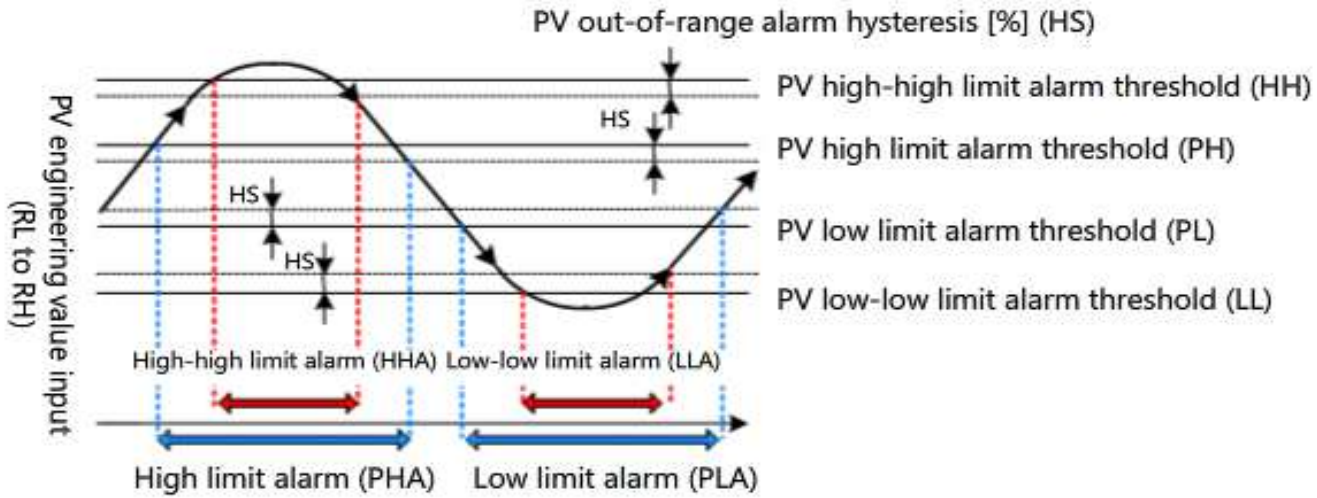
Setting item for analog output	Setting value	Description
Output conversion upper limit	12000.0	The Q62DA-FG module has a digital input range of 0 to 12000 for conversion into the analog output range of 4 to 20 mA.
Output conversion lower limit	0.0	

3.4.7 Setting the FB Property Initial Values



3.4.7 Setting the FB Property Initial Values

The next setting concerns the display of water level and related alarms.

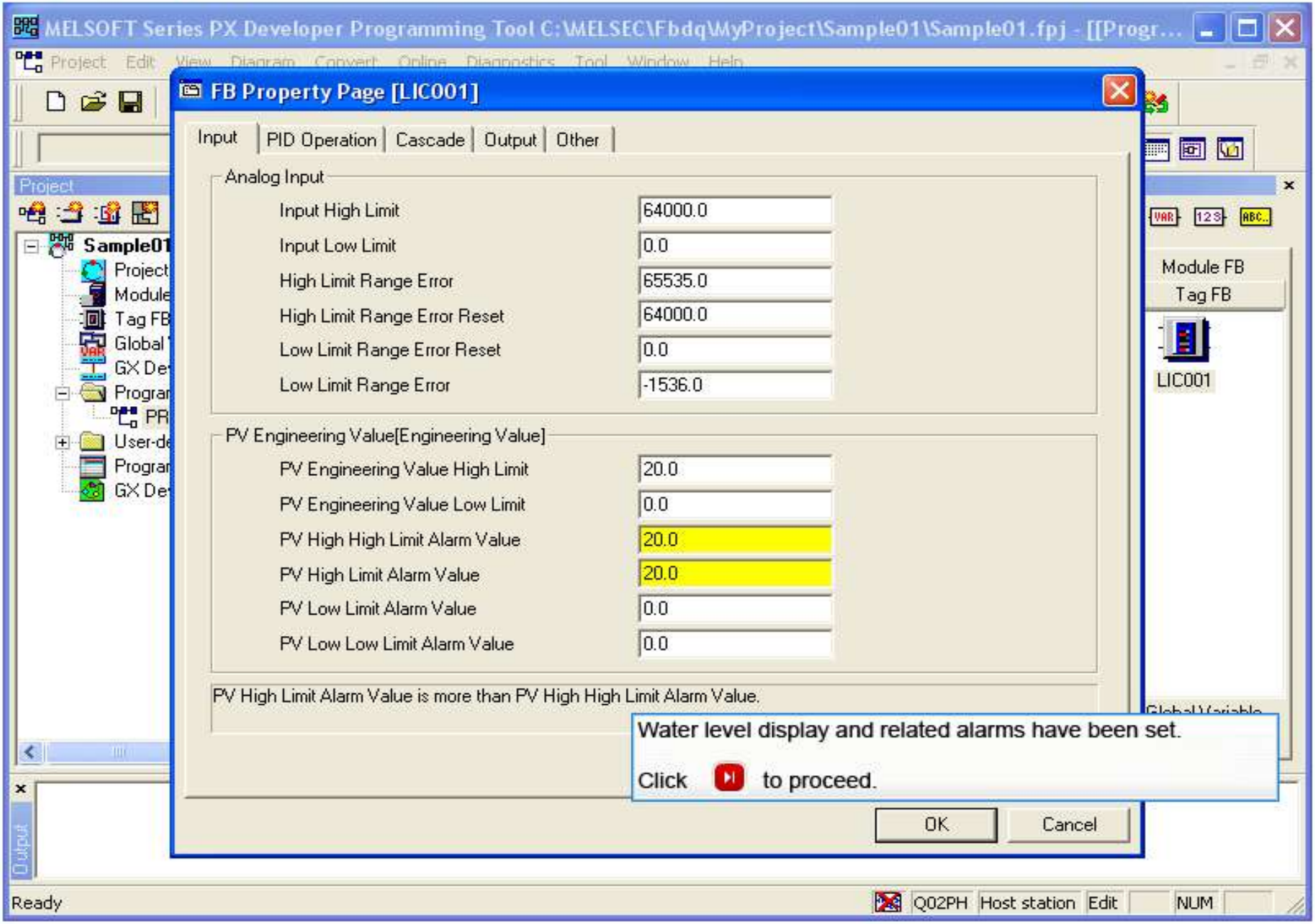


Alarms are issued when input exceeds the alarm thresholds.

The following items need to be set in accordance with the tank's upper and lower water-level limits, which are 20 and 0 respectively in this course.

Setting item	Setting value	Description
PV upper limit engineering value	20.0	The upper water-level limit of the tank is 20. Therefore, the high and low limits of the PV (process variable) range are set at 20 and 0 respectively. High and low limit alarm thresholds are also set at 20 and 0 respectively.
PV low limit engineering value	0.0	
PV high-high limit alarm threshold (HH)	20.0	
PV high limit alarm threshold (PH)	20.0	
PV low limit alarm threshold (PL)	0.0	
PV low-low limit alarm threshold (LL)	0.0	

3.4.7 Setting the FB Property Initial Values



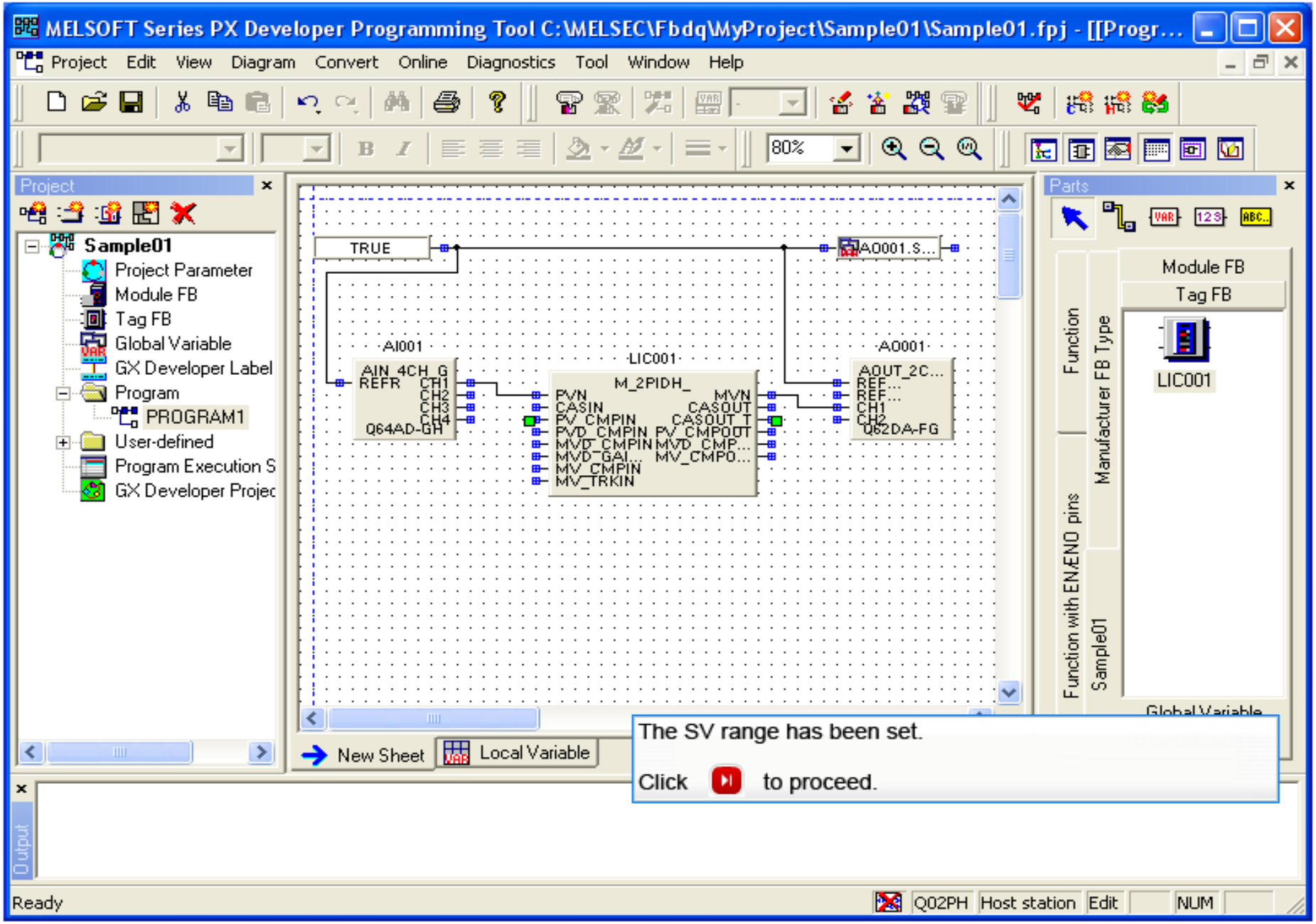
3.4.7 Setting the FB Property Initial Values

Finally, set the SV range of the tank water level for PID calculation.

The range here is defined with an upper limit of 20 and a lower limit of 0.

Setting item	Setting value	Description
SV upper limit	20.0	Set the range of tank water level.
SV low limit	0.0	

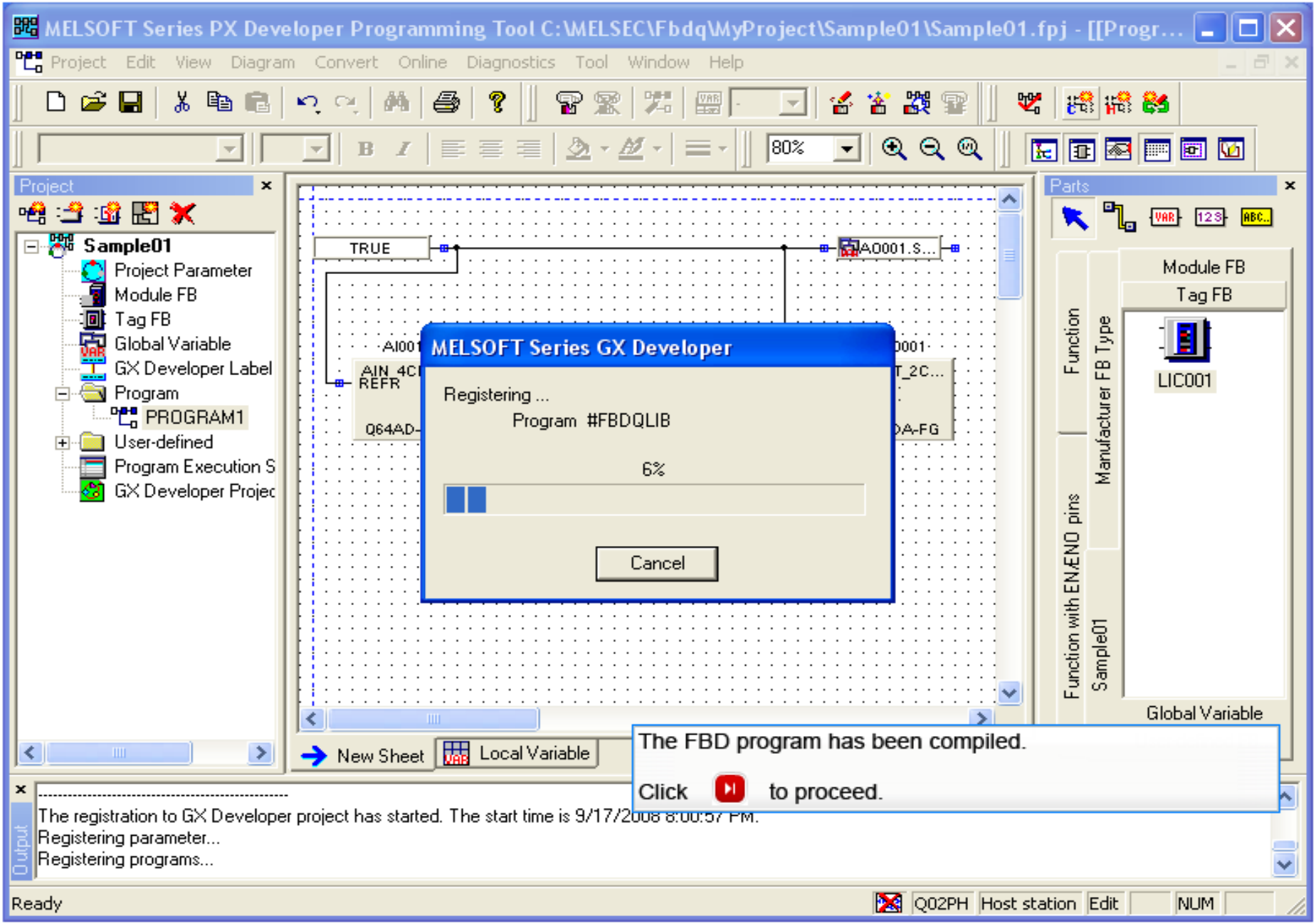
3.4.7 Setting the FB Property Initial Values



3.5 Compiling Programs

Compile the created FBD program to write it to the programmable controller.
The status of the compilation process is shown in the output window. Check the window to confirm that the compilation process has been successfully completed.

3.5 Compiling Programs

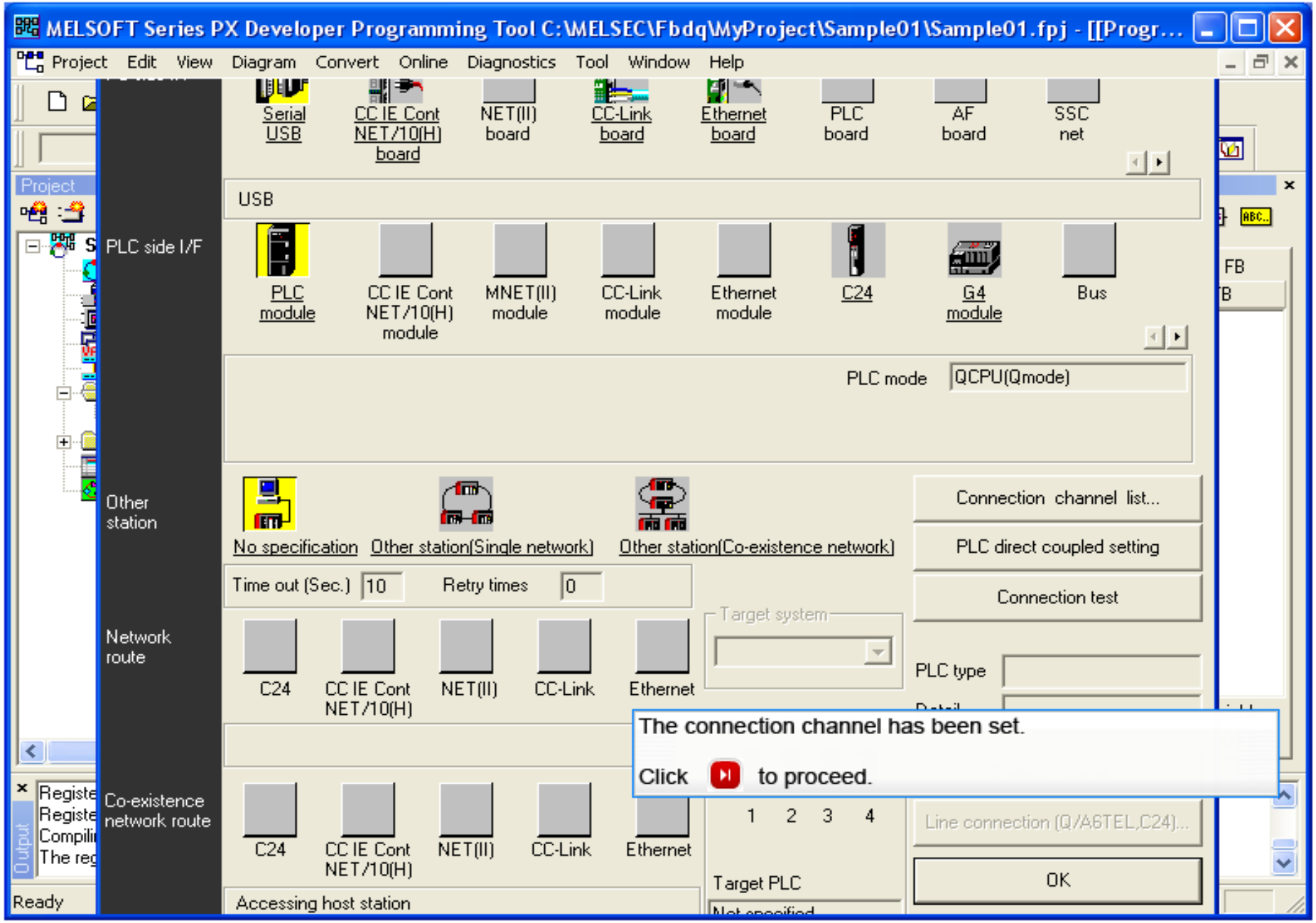


3.6 Writing Programs to the Programmable Controller CPU

3.6.1 Transfer Setup

Specify a connection channel to write the compiled program to the programmable controller CPU. Here, the PC and programmable controller CPU will be directly connected using a USB cable.

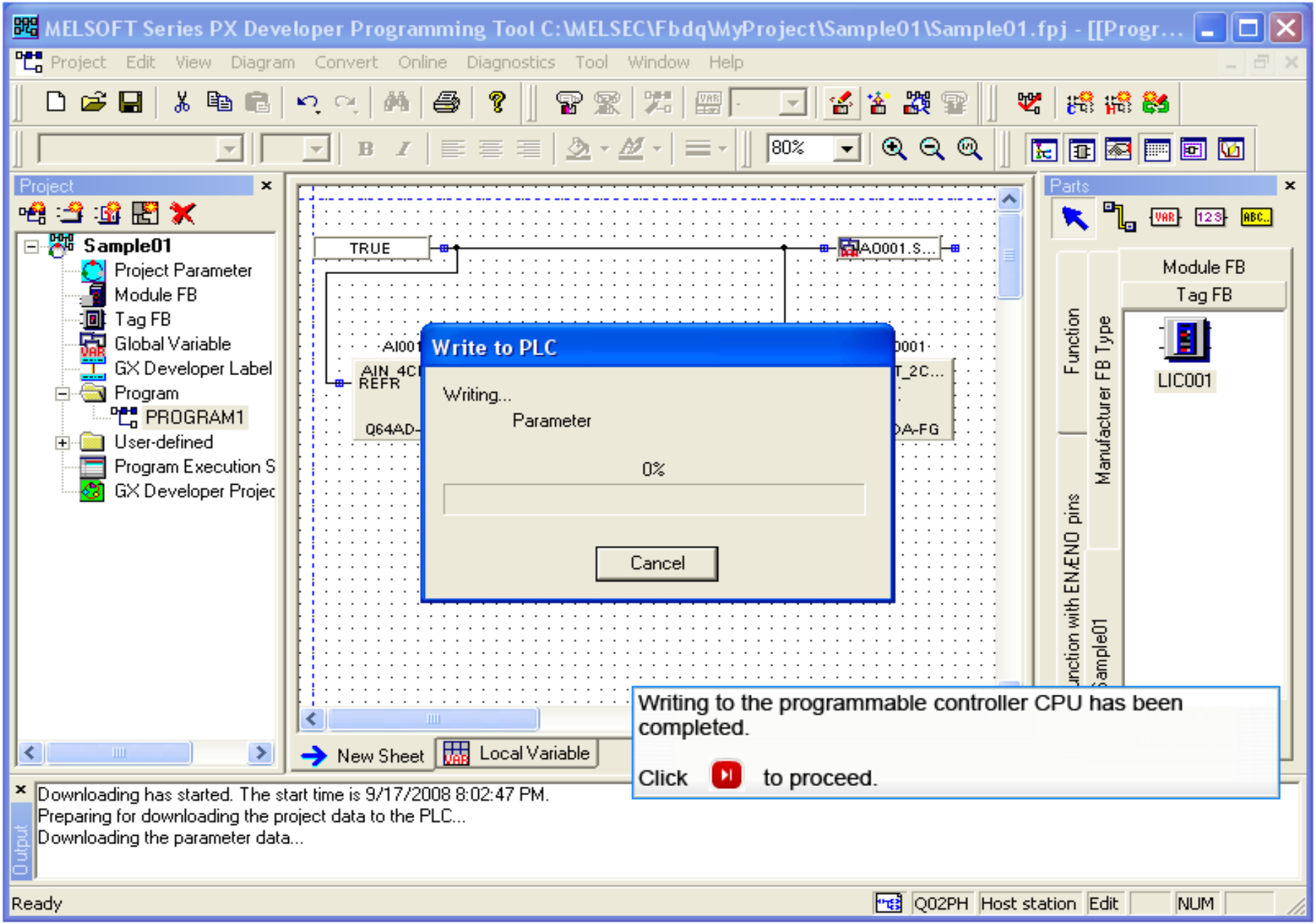
3.6.1 Transfer Setup



3.6.2 Writing to the Programmable Controller

Write the program to the programmable controller CPU.

3.6.2 Writing to the Programmable Controller



Chapter 4 Program Monitoring and Tuning

This chapter explains how to check that programs work correctly and how to tune PID control using the PX Developer programming and monitoring tools.

4.1 Starting the PX Developer Monitoring Tool

Start the PX Developer monitoring tool to monitor the operation of the FBD program that has been created.
Enter the engineer mode, which enables you to set up the monitoring tool.

The monitoring tool has the following modes.

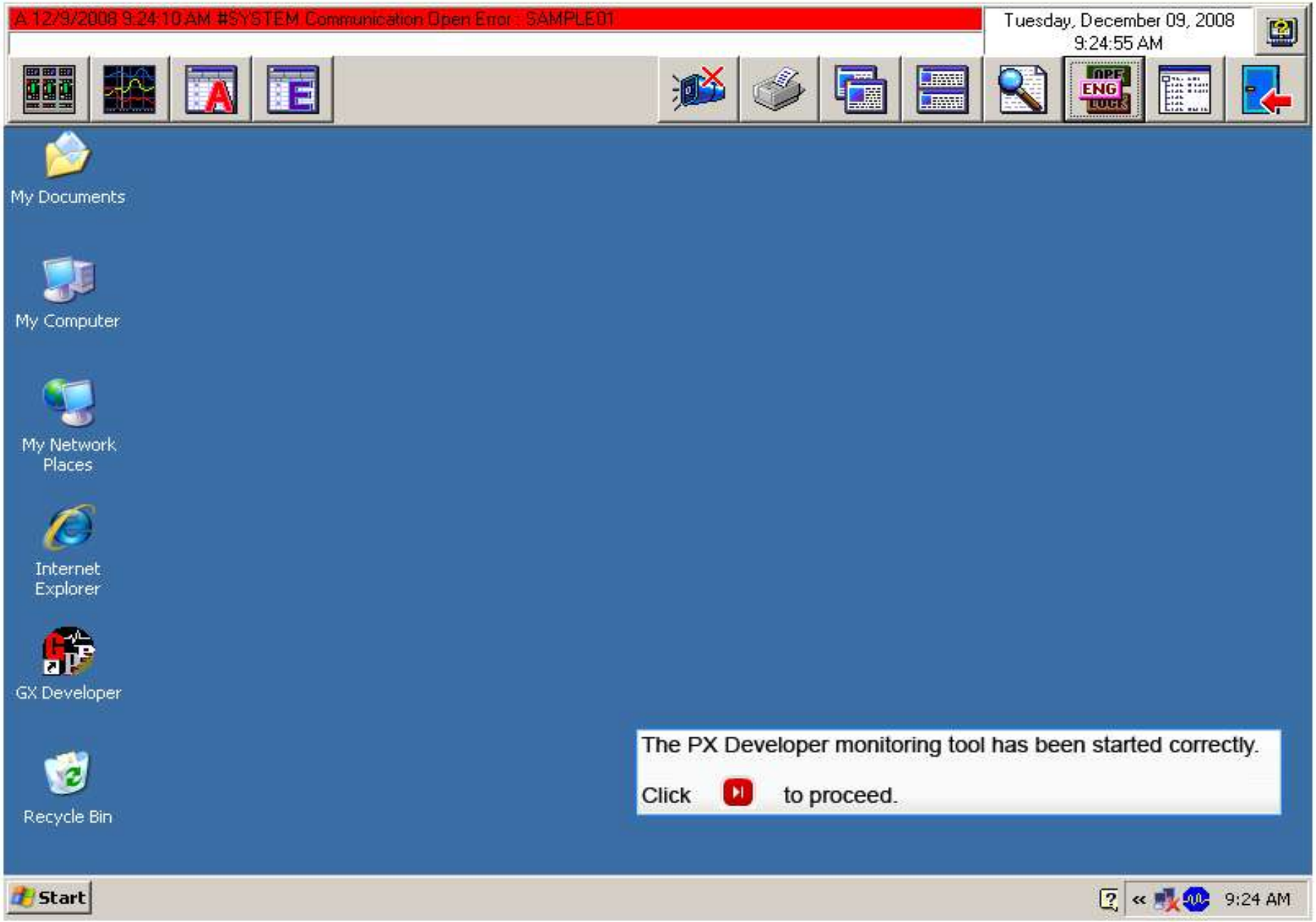
Mode name	Description
Engineer mode (for designing and administration)	In this mode, all of the monitoring tool functions can be used. This mode is used when making initial settings and changing settings.
Operator mode (for monitoring)	In this mode, general monitoring functions can be used while operating conditions and other settings of the functions cannot be changed. The system normally operates in this mode.
Lock mode	This mode blocks an attempt to change the operating conditions and other settings of the functions and to use tags for these purposes.

The engineer mode can be entered by clicking the mode switching button and entering the following user name and password for engineering authorization.

User name: admin
Password: admin

(The user name and password that have been entered can be changed later.)

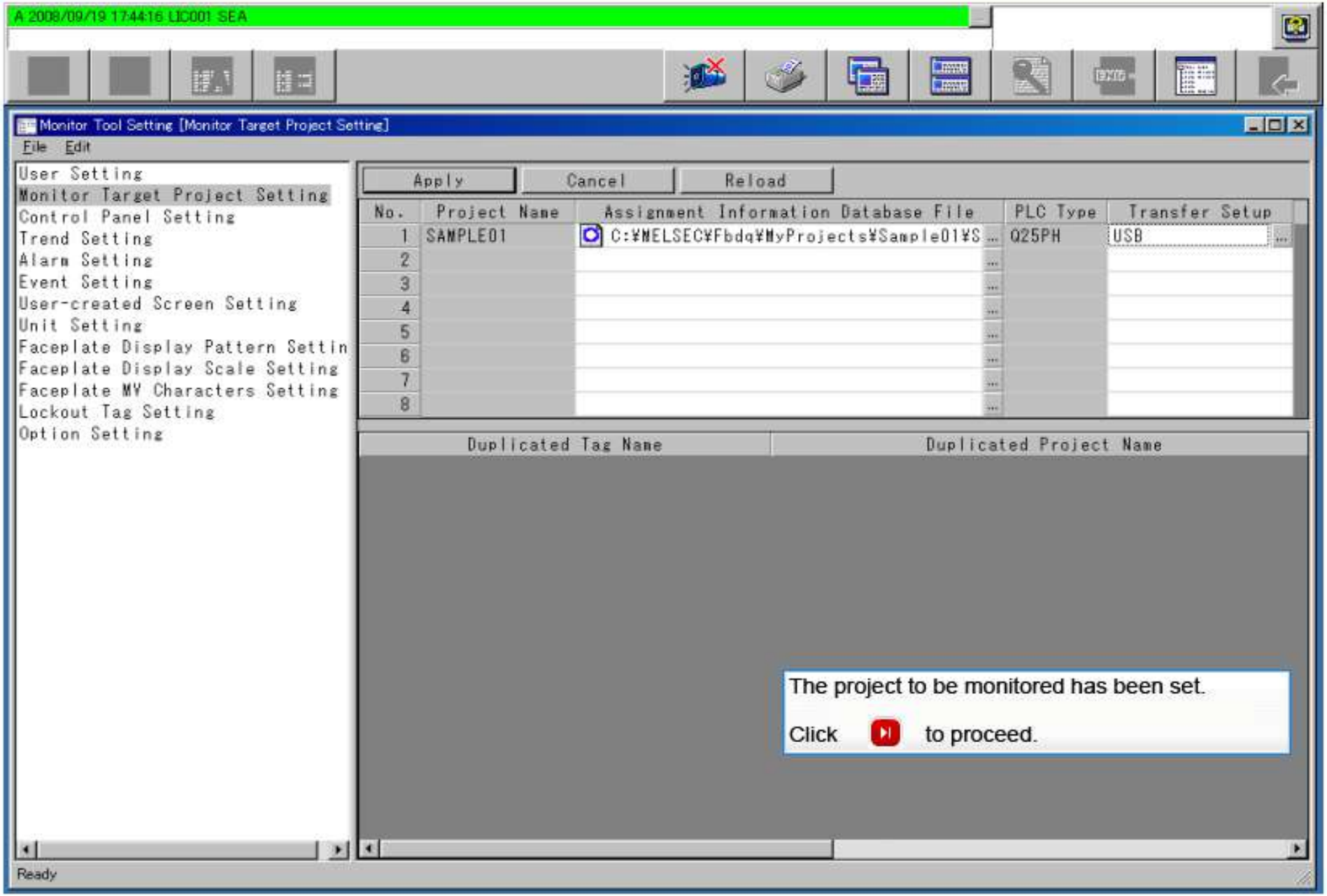
4.1 Starting the PX Developer Monitoring Tool



4.2 Setting a Monitored Project

Set a project being monitored by the PX Developer monitoring tool.
You will set the Sample01 project that has been created using the PX programming tool as the project for monitoring.

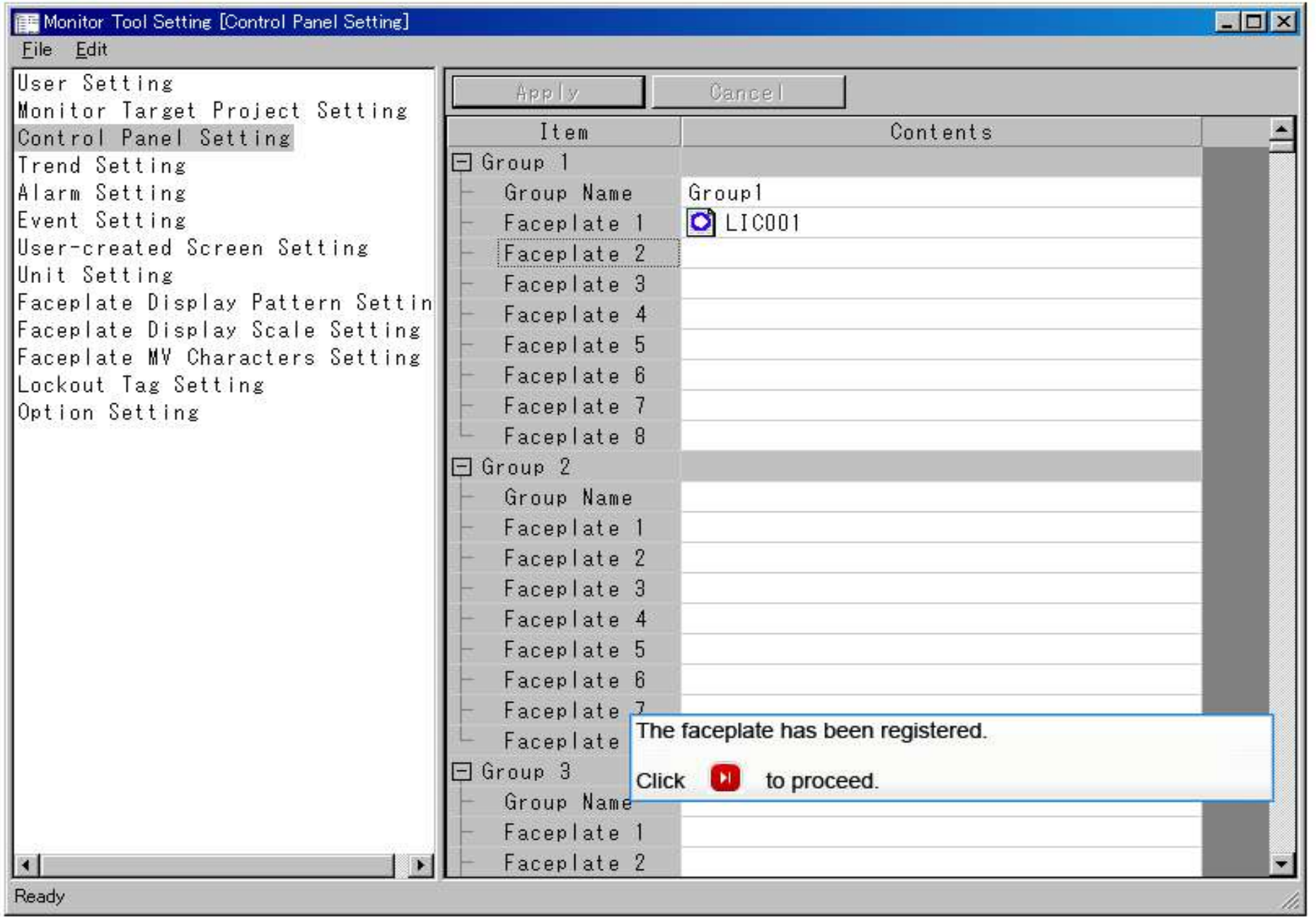
4.2 Setting a Monitored Project



4.3 Registering a Faceplate in Control Panel Setup

The PX Developer monitoring tool offers a control panel setup feature whereby up to eight faceplates, which resemble actual controllers, can be arranged on the same screen.
In this section, you will register a faceplate for the LIC001 tag FB variable that has been created in the program.

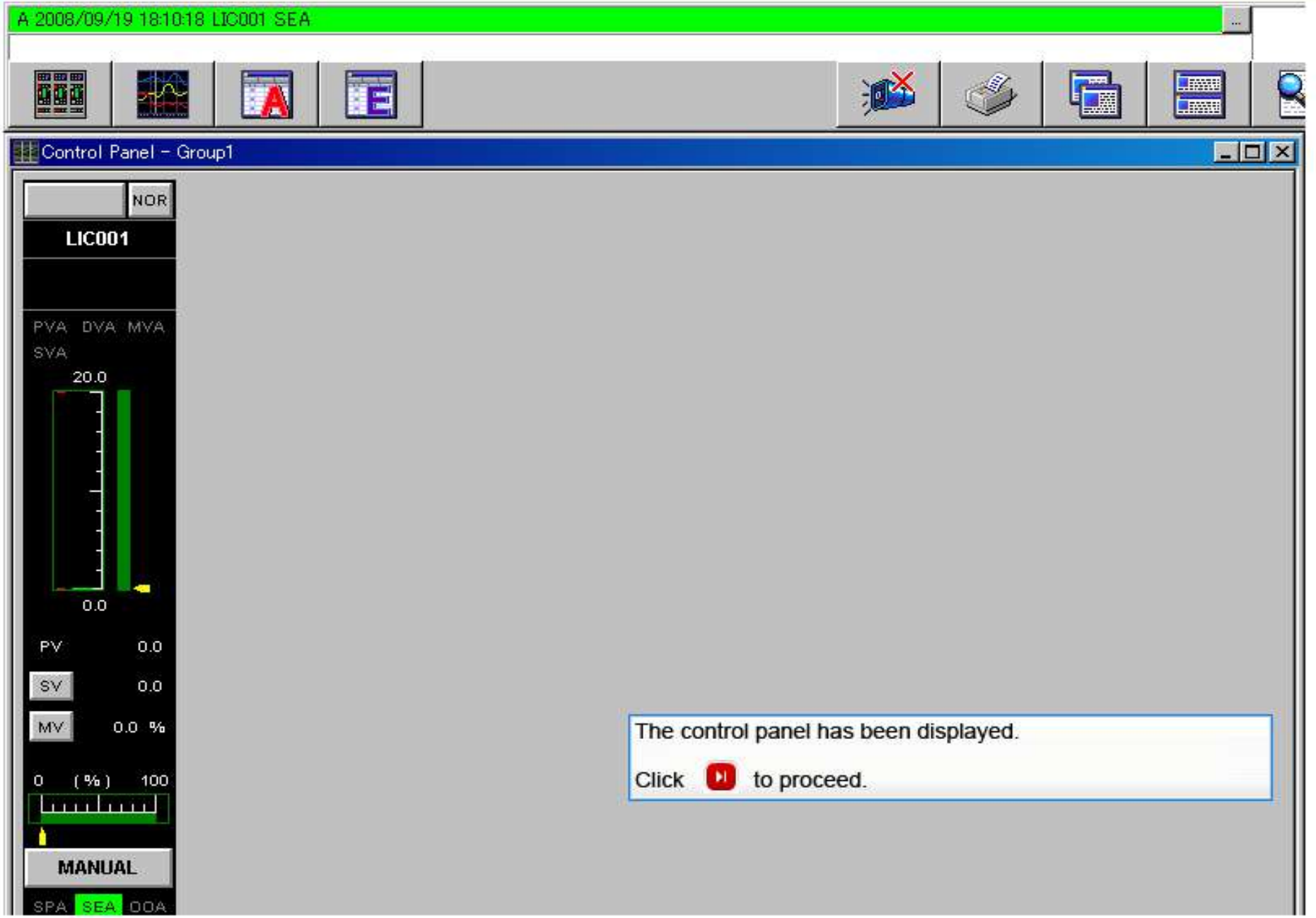
4.3 Registering a Faceplate in Control Panel Setup



4.4 Displaying the Control Panel

You will now display the control panel to check whether it contains the LIC001 faceplate that has been registered.

4.4 Displaying the Control Panel



4.5 Tuning the PID Control Loop

Click the **Details** button on the faceplate to open the **Tuning** window, and identify PID constants by auto-tuning.

4.5.1 Additional Information - Auto-tuning

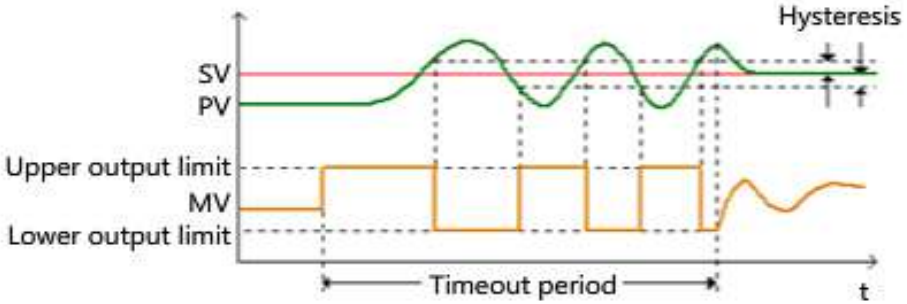
The high-performance PID control tag FB (M_2PIDH_) offers a choice of two auto-tuning methods to meet a variety of applications: limit cycle and step response.

Characteristics of limit cycle and step response methods

The limit cycle method has a minimal noise impact on PV values during the identification of PID constants, offering stable PID constants. The step response method is suitable for control systems that require non-fluctuating MV and PV values.

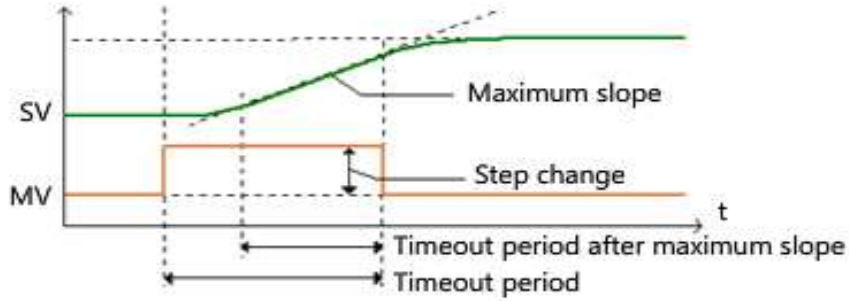
Limit cycle method

A two-position (ON/OFF) operation cycle of MV output is repeated three times to temporarily oscillate the controlled system while the amplitude and cycle of PV values are measured to calculate the most appropriate PID constant.

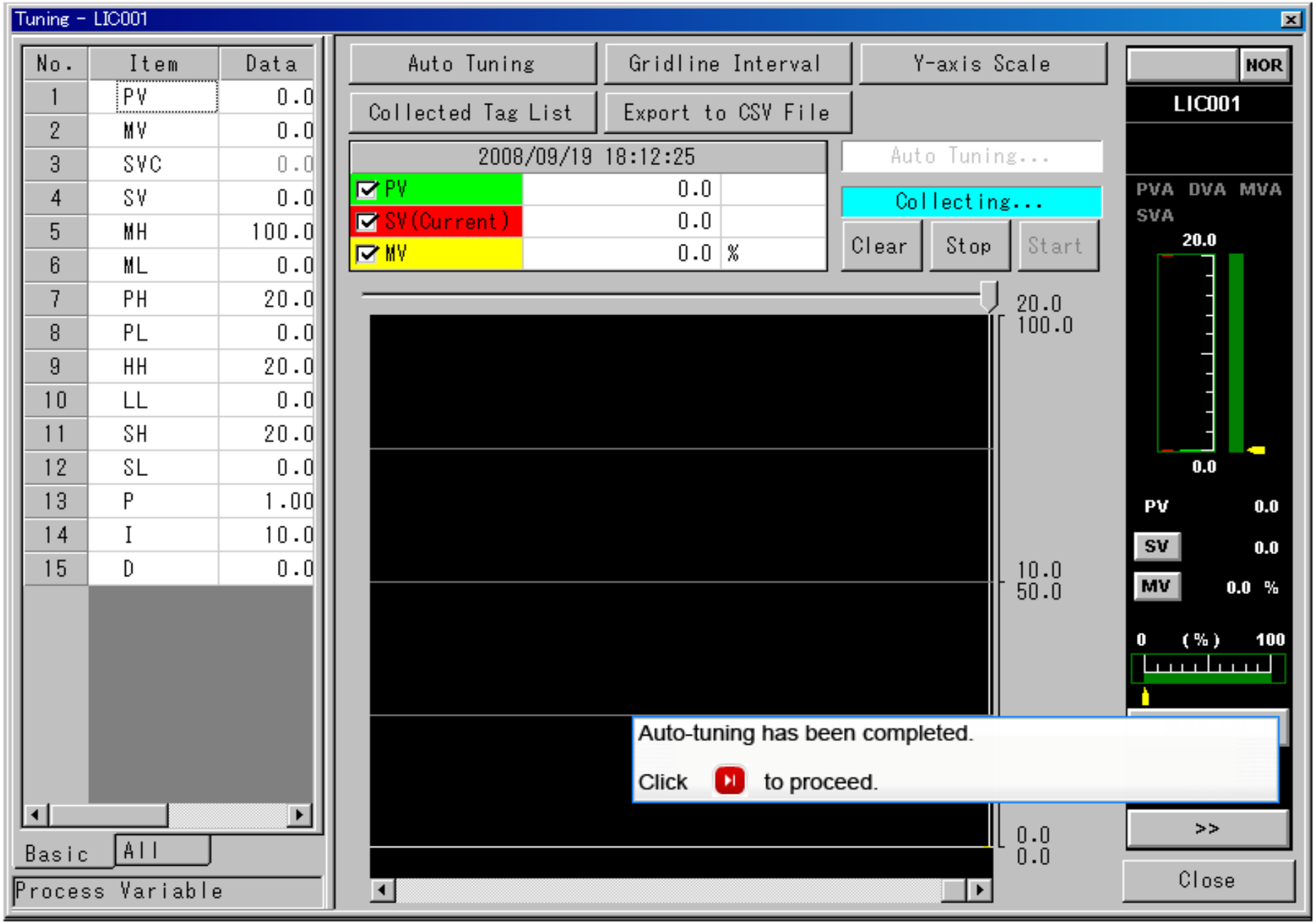


Step response method

While generating step changes in MV output, changes in PV values are measured to calculate the most suitable PID constant.



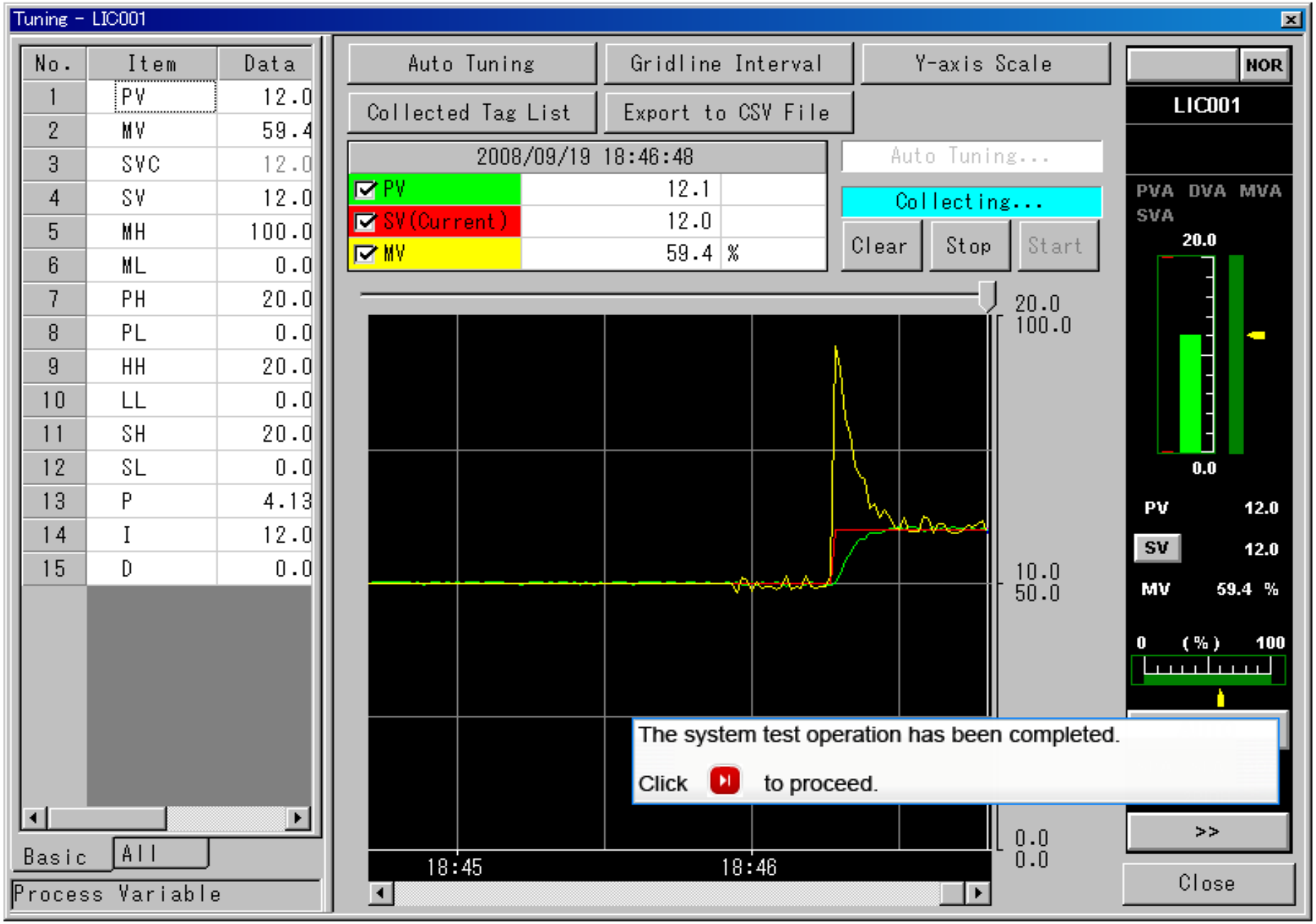
4.5.1 Additional Information - Auto-tuning



4.6 Test Operation of the System

Test-operate the system for PID loop automatic control using the PID constants identified by auto-tuning, and check whether measured PV values converge onto the target SV value.

4.6 Test Operation of the System



Test Final Test

Now that you have completed all of the lessons of the PLC MELSEC Process Control System Basics Course, you are ready to take the final test. If you are unclear on any of the topics covered, please take this opportunity to review those topics.
There are a total of 5 questions (19 items) in this Final Test.
You can take the final test as many times as you like.

How to score the test
After selecting the answer, make sure to click the **Answer** button. Your answer will be lost if you proceed without clicking the Answer button. (Regarded as unanswered question.)

Score results
The number of correct answers, the number of questions, the percentage of correct answers, and the pass/fail result will appear on the score page.

Correct Answers : 2

Total Questions : 9

Percentage : 22%

To pass the test, you have to answer **60%** of the questions correct.

Proceed Review Retry

- Click the **Proceed** button to exit the test.
- Click the **Review** button to review the test. (Correct answer check)
- Click the **Retry** button to retake the test again.

Test Final Test 1

MELSEC process control system modules/software
For each of the descriptions, choose the corresponding module/software from the list.

Description	Module/Software
An FBD software package for process control system	--Select--
A module designed to receive current/voltage signals of typically 4-20 mA/1-5 V from a converter	--Select--
A CPU module that ensures uninterrupted system operation in the event of a control system malfunction by automatically switching control to the standby system	--Select--
An analog module compatible with two-wire transmitters	--Select--
A module to which signal lines from a platinum/nickel temperature-measuring resistor can be directly connected	--Select--
A module that offers high-speed loop and sequence control and the possibility of developing a multiple-CPU system	--Select--

Answer Back

Test Final Test 2

PX Developer programming tool functions
For each of the FB descriptions, choose the corresponding PX Developer programming tool function from the list.

Description	Function
An FB designed to receive and send analog/digital signals like analog modules and I/O modules do	--Select--
An FB designed to accommodate controllers for PID and other control	--Select--

Answer Back

Test Final Test 3

PX Developer monitoring tool functions
For each of the screen descriptions, choose the corresponding PX Developer monitoring tool function from the list.

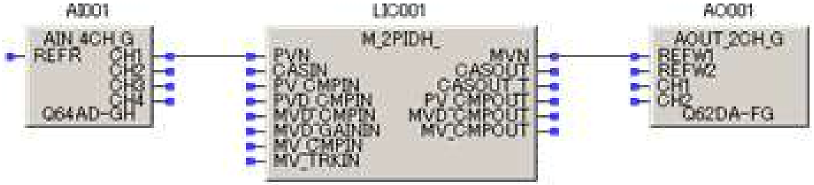
Description	Function
A setting screen for displaying faceplates by the group	--Select--
A screen to identify PID constants by step response and limit cycle methods	--Select--

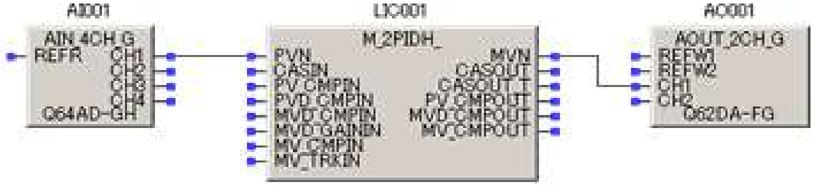
Answer Back

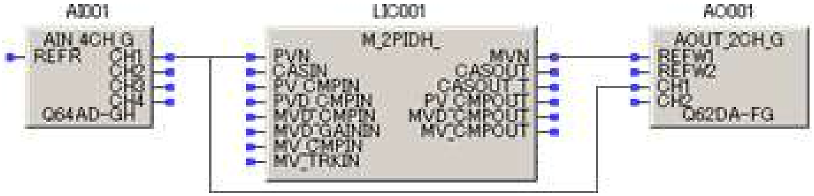
Test | Final Test 4

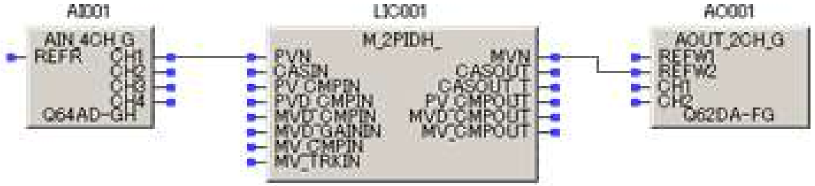
FBD programming

The following figures show connections between the module FBs representing current/voltage input and output modules and the tag FB for PID control. Choose the one that correctly shows the connection.









Answer

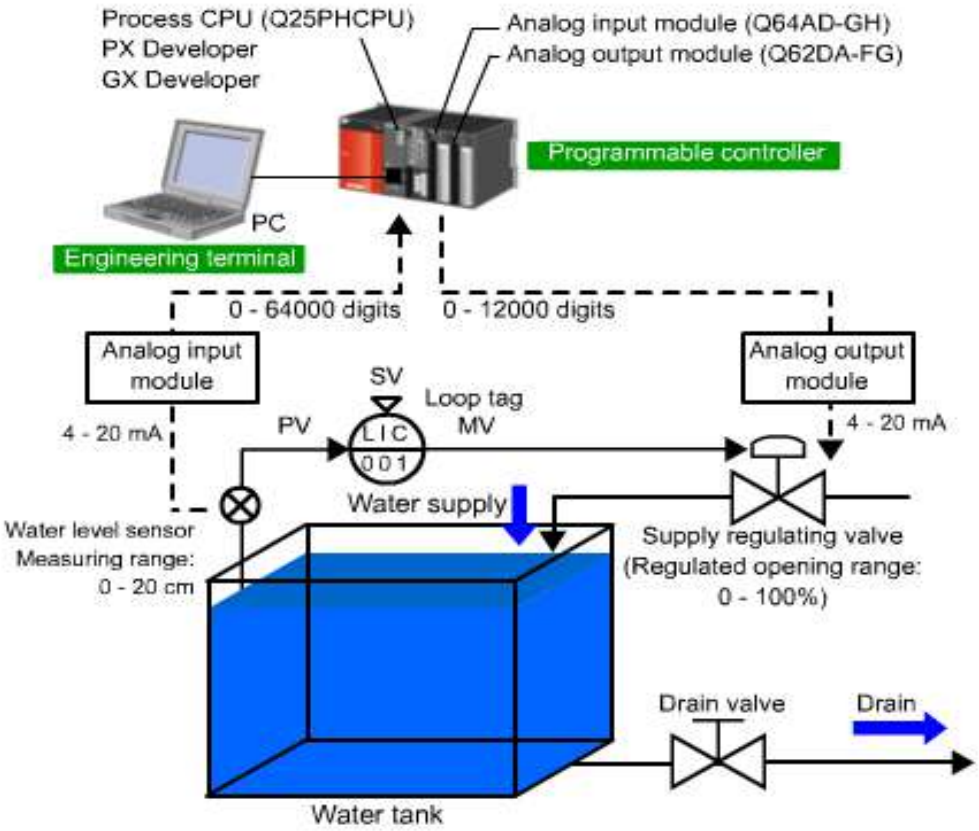
Back

Test Final Test 5

FB property

Set properties for a tag FB (M_2PIDH_) representing the loop tag LIC001 in the figure below.

Select the correct value for each of the eight setting items.



FB property setting item	Options
Analog input	
Input signal upper limit	
Input signal lower limit	
Analog output	
Output conversion upper limit	
Output conversion lower limit	
PV engineering value	
PV engineering value upper limit	
PV engineering value lower limit	
PID calculation	
SV upper limit	
SV lower limit	

Answer Back

Test Score

You have completed the Final Test. You results area as follows.
To end the Final Test, proceed to the next page.

Correct answers : 5

Total questions : 5

Percentage : 100%

Proceed Review

Congratulations. You passed the test.

You have completed the PLC MELSEC Process Control System Basics Course.

Thank you for taking this course.

We hope you enjoyed the lessons and the information you acquired in this course will be useful in the future.

You can review the course as many times as you want.

Review Close