MITSUBISHI

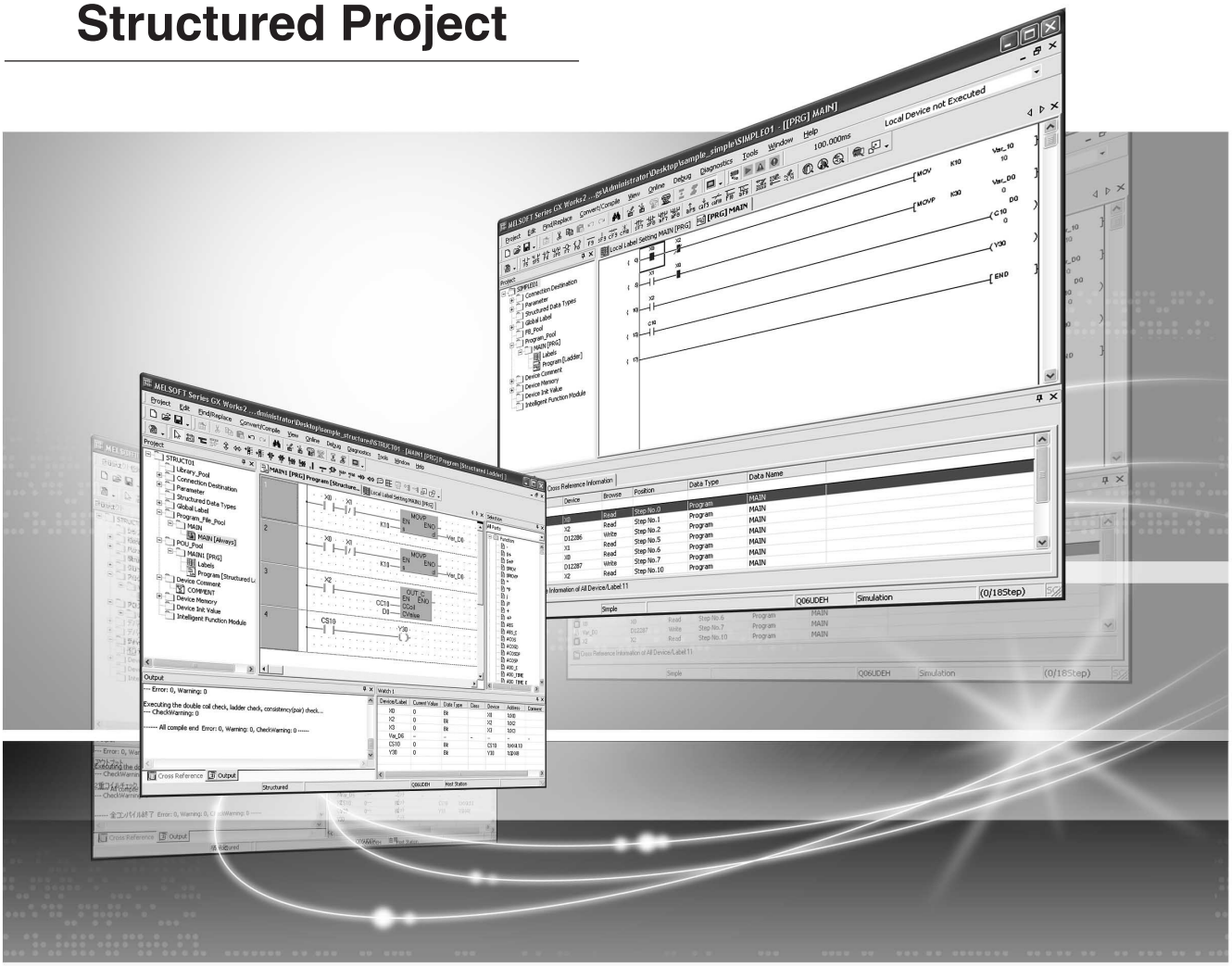
MELSOFT

Integrated FA Software

GX Works2 Version 1

Operating Manual

Structured Project



MELSOFT
Integrated FA Software

SW1DNC-GXW2-E

● SAFETY PRECAUTIONS ●

(Always read these instructions before using this product.)

Before using this product, thoroughly read this manual and the relevant manuals introduced in this manual and pay careful attention to safety and handle the products properly.
The precautions given in this manual are concerned with this product. For the safety precautions of the programmable controller system, refer to the User's Manual for the CPU module.
In this manual, the safety precautions are ranked as "DANGER" and "CAUTION".

DANGER	Indicates that incorrect handling may cause hazardous conditions, resulting in death or severe injury.
CAUTION	Indicates that incorrect handling may cause hazardous conditions, resulting in minor or moderate injury or property damage.

Note that the CAUTION level may lead to serious consequences according to the circumstances. Always follow the precautions of both levels because they are important for personal safety.

Please save this manual to make it accessible when required and always forward it to the end user.

[Design Instructions]

DANGER

- **When data change, program change, or status control is performed from a personal computer to a running programmable controller, create an interlock circuit outside the programmable controller to ensure that the whole system always operates safely.**
  Furthermore, for the online operations performed from a personal computer to a programmable controller CPU, the corrective actions against a communication error due to such as a cable connection fault should be predetermined as a system.

[Startup/Maintenance Instructions]

CAUTION

- **The online operations performed from a personal computer to a running programmable controller CPU (Program change when a programmable controller CPU is RUN, operating status change such as RUN-STOP switching, and remote control operation) have to be executed after the manual has been carefully read and the safety has been ensured.**
  When changing a program while a programmable controller CPU is RUN, it may cause a program corruption in some operating conditions. Fully understand the precautions described in GX Works2 Version1 Operating Manual (Common) before use.

REVISIONS

The manual number is written at the bottom left of the back cover.

Print date	Manual number	Revision
Jul., 2008	SH(NA)-080781ENG-A	First edition

Japanese Manual Version SH-080732-B

INTRODUCTION

Thank you for purchasing the Mitsubishi integrated FA software, MELSOFT series.
Before using this product, thoroughly read this manual to develop full familiarity with the functions and performance to ensure correct use.

CONTENTS

SAFETY PRECAUTIONS ..... A - 1
REVISIONS ..... A - 2
INTRODUCTION ..... A - 3
CONTENTS ..... A - 3
MANUALS ..... A - 19
GENERIC TERMS AND ABBREVIATIONS IN THIS MANUAL ..... A - 24

1 OVERVIEW 1 - 1 to 1 - 16

1.1 What is Structured project? 1 - 2
1.2 Features of Structured Project 1 - 2
1.3 List of Functions 1 - 5
1.3.1 List of functions common to Simple project and Structured project ..... 1 - 5
1.3.2 List of functions for editing in ladder language ..... 1 - 13
1.3.3 List of functions for editing SFC diagrams ..... 1 - 13
1.3.4 List of functions for editing SFC block list ..... 1 - 13
1.3.5 List of functions for editing in ST language ..... 1 - 14
1.3.6 List of functions for editing in structured ladder language ..... 1 - 15

2 SCREEN CONFIGURATION 2 - 1 to 2 - 2

2.1 Overview of Screen Configuration 2 - 2

3 PROGRAMMING PROCEDURE 3 - 1 to 3 - 4

3.1 Creating Programs 3 - 2

4 PROGRAM CONFIGURATIONS 4 - 1 to 4 - 16

4.1 Program Configurations of Structured Project 4 - 2
4.2 Creating Program Files and Tasks 4 - 3
4.2.1 Procedure for creating program files and tasks ..... 4 - 3
4.2.2 Registering program blocks to tasks ..... 4 - 4
4.2.3 Setting executing conditions ..... 4 - 6
4.3 Creating POUs 4 - 9
4.3.1 Procedure for creating POUs ..... 4 - 9
4.3.2 Setting properties of functions and function blocks ..... 4 - 10
4.4 Using POUs 4 - 13

4.4.1 Using functions and function blocks ..... 4 - 14

5 SETTING LABELS 5 - 1 to 5 - 20

5.1 Label Setting Screens 5 - 2
5.2 Setting Global Labels 5 - 3
5.3 Setting Local Labels for Program Blocks 5 - 5
5.4 Setting Labels for Functions and Function Blocks 5 - 7
5.5 Common Operations for Setting Labels 5 - 9
5.5.1 Selecting data types ..... 5 - 9
5.5.2 Editing lines ..... 5 - 12
5.6 Setting Structured Data Type Labels 5 - 14
5.6.1 Setting data type of structure ..... 5 - 14
5.6.2 Setting data type as structure ..... 5 - 14
5.6.3 Assigning devices to structured data type labels ..... 5 - 15
5.6.4 Assigning devices to structured array type labels ..... 5 - 17
5.7 Setting Ranges for Devices Assigned Automatically 5 - 20

6 EDITING PROGRAMS 6 - 1 to 6 - 50

6.1 Types of Program Editor 6 - 2
6.1.1 Available programming languages ..... 6 - 3
6.1.2 Available comment formats ..... 6 - 4
6.2 Common Operations of Program Editors 6 - 5
6.2.1 Using POUs in a program (Function Block Selection window) ..... 6 - 5
6.2.2 Using templates ..... 6 - 9
6.2.3 Using labels in the program ..... 6 - 10
6.2.4 Editing networks ..... 6 - 12
6.2.5 Undo and redo ..... 6 - 17
6.2.6 Displaying grid ..... 6 - 17
6.2.7 Changing display size of networks ..... 6 - 18
6.2.8 Opening label setting and program screens for selected POU ..... 6 - 19
6.2.9 Opening label setting screen of program editor being edited ..... 6 - 20
6.2.10 Setting display contents of tool hint ..... 6 - 21
6.3 Editing ST Programs 6 - 22
6.3.1 Displaying applicable label names ..... 6 - 23
6.3.2 Automatic indention ..... 6 - 23
6.3.3 Setting tab length ..... 6 - 24
6.3.4 Using bookmarks ..... 6 - 24
6.4 Editing Structured Ladder Programs 6 - 25
6.4.1 Selecting editing modes ..... 6 - 26
6.4.2 Entering network elements ..... 6 - 27
6.4.3 Moving network elements ..... 6 - 29
6.4.4 Connecting lines ..... 6 - 32
6.4.5 Inserting and deleting rows and columns ..... 6 - 34

6.4.6 Changing type of contact and coil ..... 6 - 35
6.4.7 Changing the number of arguments for functions and function blocks ..... 6 - 37
6.4.8 Adding variables automatically when entering functions and function blocks ..... 6 - 38
6.4.9 Editing variable names and instance names ..... 6 - 39
6.4.10 Changing display format of variables ..... 6 - 40
6.4.11 Editing comments ..... 6 - 44
6.4.12 Editing in Guided mode ..... 6 - 46

7 COMPILING PROGRAMS 7 - 1 to 7 - 14

7.1 Compiling All Programs 7 - 2
7.2 Compiling Only Modified Programs 7 - 4
7.3 Precautions on Compilation 7 - 6
7.4 Executing Online Change Simultaneously with Compilation 7 - 7
7.5 Checking for Errors and Warnings when Compiling Programs 7 - 7
7.6 Changing Operating Conditions of Compilation 7 - 8
7.6.1 Changing the number of errors and warnings that stops compilation ..... 7 - 8
7.6.2 Hiding warning messages ..... 7 - 9
7.6.3 Using same name for global label and local label ..... 7 - 10
7.6.4 Using same lower case label name as device ..... 7 - 10
7.6.5 Connecting PLS and/or PLF outputs directly to other inputs with lines ..... 7 - 11
7.6.6 Assigning individual devices to EN/ENO function outputs ..... 7 - 12
7.6.7 Retaining bit type outputs of functions ..... 7 - 13

8 WRITING/READING DATA TO/FROM PROGRAMMABLE CONTROLLER CPU 8 - 1 to 8 - 4

8.1 Writing/Reading Data to/from Programmable Controller CPU 8 - 2

9 MONITORING 9 - 1 to 9 - 8

9.1 Starting and Stopping Monitoring of Program 9 - 2
9.2 Starting and Stopping Monitoring of Function Blocks 9 - 3
9.3 Changing Operating Conditions of Monitoring 9 - 4
9.3.1 Changing current value display format (decimal/hexadecimal) of word type variable ..... 9 - 4
9.3.2 Setting the number of displaying characters for character strings ..... 9 - 5
9.4 Monitoring Programs in ST Editor 9 - 6
9.5 Monitoring Programs in Structured Ladder Editor 9 - 7

10 USING LIBRARIES 10 - 1 to 10 - 18

10.1 Types of Library 10 - 2
10.2 Procedure for Using User Libraries 10 - 4
10.3 Operating User Libraries 10 - 5
10.3.1 Creating libraries ..... 10 - 6

10.3.2 Enabling editing of library data ..... 10 - 7
10.3.3 Disabling editing of library data ..... 10 - 8
10.3.4 Saving libraries to library files under specified name ..... 10 - 9
10.3.5 Saving library files ..... 10 - 10
10.3.6 Installing library data to project ..... 10 - 11
10.3.7 Updating libraries ..... 10 - 12
10.3.8 Deleting libraries from project ..... 10 - 13
10.3.9 Deleting library files ..... 10 - 14
10.3.10 Renaming libraries ..... 10 - 15
10.3.11 Setting library passwords ..... 10 - 16
10.3.12 Displaying library help ..... 10 - 17

11 SETTING OPTIONS 11 - 1 to 11 - 4

11.1 Basic Operations 11 - 2

11.2 Option Setting List 11 - 3

APPENDIX App - 1 to App - 14

Appendix 1 List of Toolbars and Shortcut Keys App - 2

Appendix 1.1 Common toolbars and shortcut keys ..... App - 2
Appendix 1.2 Toolbar icons and shortcut keys for setting labels ..... App - 5
Appendix 1.3 Toolbar icons and shortcut keys for setting device memory ..... App - 6
Appendix 1.4 Toolbar icons for executing sampling trace ..... App - 7
Appendix 1.5 Toolbar icons and shortcut keys in program editors ..... App - 8
Appendix 1.6 Shortcut key for editing intelligent function module data ..... App - 13

INDEX Index - 1 to Index - 3

CONTENTS

SAFETY PRECAUTIONS
REVISIONS
INTRODUCTION
CONTENTS
MANUALS
GENERIC TERMS AND ABBREVIATIONS IN THIS MANUAL

1 OVERVIEW

1.1 Product Overview
1.2 Features
1.3 List of Functions
1.3.1 List of functions common to Simple project and Structured project
1.3.2 List of functions for editing in ladder language
1.3.3 List of functions for editing SFC diagrams
1.3.4 List of functions for editing SFC block list
1.3.5 List of functions for editing in ST language
1.3.6 List of functions for editing in structured ladder language

2 SYSTEM CONFIGURATION

2.1 Connection from USB/Serial Port
2.1.1 USB cables (compatible with QCPU (Q mode))
2.1.2 RS-232 cables (compatible with QCPU (Q mode))
2.1.3 Connection via GOT
2.2 Connection from I/F Boards
2.2.1 I/F boards
2.3 Interfaces and Connectable Modules
2.4 System Configuration with Memory Card in Personal Computer
2.5 Supported Programming Languages

3 SCREEN CONFIGURATION AND BASIC OPERATIONS

3.1 Starting and Exiting GX Works2
3.2 Screen Configuration and Basic Operations
3.2.1 Main frame configuration
3.2.2 Toolbars
3.2.3 Work windows
3.2.4 Docking windows
3.2.5 Project window
3.2.6 Status bar
3.2.7 Changing colors and fonts
3.3 Help Function

3.3.1 Checking CPU error contents
3.3.2 Checking special relay/register contents
3.3.3 Displaying operating manuals
3.3.4 Displaying changes from GX Developer
3.3.5 Connecting to MELFANSweb website
3.3.6 Checking version of GX Works2

4 PROJECT MANAGEMENT

4.1 Project Operations
4.1.1 Creating projects
4.1.2 Opening existing projects
4.1.3 Saving projects
4.1.4 Deleting projects
4.1.5 Closing projects
4.1.6 Verifying project data
4.1.7 Changing the programmable controller type of projects
4.2 Operations of Programmable Controller CPU Data
4.2.1 Adding new data to the project
4.2.2 Copying/pasting data in projects
4.2.3 Changing project data names
4.2.4 Deleting project data
4.2.5 Displaying/editing properties
4.3 Operations of Intelligent Function Module Data
4.3.1 Adding intelligent function module data
4.3.2 Setting intelligent function module data
4.3.3 Deleting intelligent function modules
4.3.4 Displaying properties of intelligent function modules
4.3.5 Checking/changing the number of settable intelligent function module parameters
4.3.6 Checking device duplication of auto refresh setting
4.4 Changing Project Types
4.5 Managing Project Revisions
4.5.1 Registering revision information (creating backup)
4.5.2 Displaying revision list
4.5.3 Restoring backup projects
4.5.4 Deleting revision information
4.5.5 Verifying revisions
4.6 Utilizing Projects and Data in Other Formats
4.6.1 Opening projects in other formats
4.6.2 Reading GX IEC Developer data
4.6.3 Reading project files created with GX Configurator-QP
4.6.4 Saving projects in other formats
4.7 Setting Security for Projects
4.7.1 Setting/resetting security of projects
4.7.2 Managing (adding/deleting/changing) users
4.7.3 Logging in projects
4.7.4 Changing access authority of access level
4.7.5 Precautions on using the security function
4.8 Protecting POUs in Projects

4.8.1 Managing (setting/unlocking/deleting) block passwords
4.8.2 Setting/changing block passwords
4.8.3 Unlocking block passwords

5 EDITING PROGRAMS

5.1 Programming
5.2 Converting/Compiling Projects

6 SETTING PARAMETERS

6.1 Setting PLC Parameter
6.1.1 PLC parameter item list
6.2 Setting Network Parameter
6.2.1 Network parameter item list
6.2.2 Supplementary explanation for Network parameter setting
6.3 Setting Remote Password
6.4 Checking Parameters

7 SETTING DEVICE MEMORY

7.1 Device Memory
7.1.1 Features of device memory
7.1.2 List of devices selectable on the Insert Devices screen
7.2 Setting Device Memory
7.2.1 Setting device values in units of points
7.2.2 Setting device values by setting device range
7.2.3 Setting character strings
7.2.4 Changing display format
7.2.5 Changing the number of rows/columns in device memory editor
7.3 Searching Devices and Device Values
7.4 Writing/Reading Device Memory Data
7.4.1 Writing/reading data to/from programmable controller CPUs
7.4.2 Writing/reading data to/from Excel files

8 SETTING DEVICE INITIAL VALUES

8.1 Device Initial Values
8.1.1 Feature of device initial values
8.1.2 List of settable devices for device initial values
8.1.3 Procedure for setting device initial values
8.2 Setting Device Initial Values

9 SETTING DEVICE COMMENTS

9.1 Device Comments
9.1.1 Global comments/local comments

9.1.2 List of comment settable devices
9.2 Creating Device Comments
9.2.1 Creating comments on the device comment editor
9.2.2 Creating comments on the ladder editor
9.3 Deleting Device Comments
9.3.1 Deleting comment data of all devices
9.3.2 Deleting comment data of device being displayed
9.4 Utilizing Sample Comments
9.4.1 Utilizing sample comments of special relays/special registers
9.4.2 Utilizing sample comments of intelligent function modules

10 SEARCH/REPLACE

10.1 Searching/Replacing Character Strings
10.1.1 Searching/replacing character strings
10.1.2 Displaying results and logs
10.2 Cross Reference
10.2.1 Creating/displaying cross reference information
10.2.2 Filtering display
10.3 Displaying device usage list

11 SETTING PROGRAMMABLE CONTROLLER CPU CONNECTION TARGET

11.1 Setting Connection Targets
11.1.1 Transfer setup screen
11.1.2 Specifying connection target for regular use
11.2 Accessing Programmable Controller CPU Directly Coupled
11.2.1 Connecting to QCPU (Q mode)
11.2.2 Accessing Built-in Ethernet port QCPUs
11.3 Accessing Programmable Controller CPU Via Networks
11.4 Accessing Programmable Controller CPU in Multiple CPU System
11.5 Accessing Programmable Controller CPU via Ethernet Board
11.6 Accessing Programmable Controller CPU via CC-Link G4 or G4-S3 Module
11.7 Setting for Access via Serial Communication Module
11.7.1 Connection on a 1:1 basis
11.7.2 Connection on a 1:n basis
11.8 Setting for Access via GOT (GOT Transparent Function)
11.8.1 Accessing programmable controller CPU via GOT
11.8.2 Accessing programmable controller CPU via serial communication module/modem interface module
11.9 Precautions on Communication with Programmable Controller CPU

12 WRITING/READING DATA

12.1 Writing or Reading Data to/from Programmable Controller CPU
12.1.1 Setting write/read range of program (program file)
12.1.2 Setting write/read range of device data

12.1.3 Setting comment write/read ranges
12.1.4 Writing/reading intelligent function module data
12.1.5 Precautions on writing/reading data to/from programmable controller CPU
12.2 Verifying Data at Programmable Controller CPU against Data at Personal Computer
12.3 Deleting Programmable Controller CPU Data
12.4 Copying Program Memory Data to ROM
12.5 Writing/Reading/Deleting PLC User Data
12.6 Transferring Data to Program Memory in a Batch
12.7 Latch Data Backup
12.7.1 Backing up latch data
12.7.2 Deleting backup data
12.8 Online Program Change
12.8.1 Performing Online program change during conversion/compile
12.8.2 Performing Online program change in units of files
12.8.3 Precautions on Online program change of rise, fall or SCJ instruction
12.8.4 Precautions on Online program change of SFC programs
12.8.5 Precautions on each programmable controller series
12.9 Writing/Reading Data to/from Memory Cards
12.9.1 Writing/Reading Data to/from ATA/SRAM Cards

13 PROTECTING PROGRAMMABLE CONTROLLER CPU DATA

13.1 Registering/Changing Password
13.2 Cancelling Password
13.3 Unlocking Password Temporarily

14 MONITORING

14.1 Monitoring Programs
14.1.1 Monitoring Programs
14.1.2 Monitor status
14.2 Registering and Monitoring Devices
14.2.1 Registering and monitoring devices
14.2.2 Registering devices/labels using shortcut menu from program editor/label editor
14.2.3 Registering devices/labels using the drag-and-drop operation from the program editor/label editor
14.2.4 Changing display format (decimal/hexadecimal) of current values
14.3 Monitoring Devices/Buffer Memory in a Batch
14.4 Starting/Stopping Monitor of All Windows

15 DEBUGGING PROGRAMS

15.1 Changing Current values
15.2 Sampling Trace
15.2.1 Opening sampling trace screen
15.2.2 Setting executing condition of sampling trace

15.2.3 Registering devices/labels
15.2.4 Executing sampling trace
15.2.5 Saving trace data
15.2.6 Applicable devices/labels for sampling trace
15.2.7 Precautions when performing sampling trace
15.2.8 Compatibility with sampling trace data in GX Developer
15.3 Debugging Programs Offline
15.3.1 Starting simulation
15.3.2 Checking unsupported instructions/devices

16 OPERATING PROGRAMMABLE CONTROLLER CPU

16.1 Remote Operation of Programmable Controller CPU
16.2 Formatting Programmable Controller CPU Memory
16.3 Clearing Programmable Controller Memory
16.4 Arranging Programmable Controller Memory
16.5 Setting Clock in Programmable Controller CPU

17 DIAGNOSING PROGRAMMABLE CONTROLLER STATUS

17.1 Diagnosing Programmable Controller CPU
17.1.1 Diagnosing QCPU (Q mode)
17.2 Diagnosing MELSECNET
17.2.1 Network test
17.2.2 Loop test
17.2.3 Setup confirmation test
17.2.4 Station order check test
17.2.5 Communication test
17.2.6 Error history monitoring
17.2.7 Network Monitor Details
17.2.8 Monitoring other station information
17.3 Diagnosing CC-Link IE Controller Network
17.3.1 Communication test
17.3.2 Link start/stop
17.3.3 Logging
17.4 Diagnosing CC-Link and CC-Link/LT
17.4.1 Monitoring line (host station)
17.4.2 Loop test
17.4.3 Monitoring line (other stations)
17.5 Diagnosing Ethernet
17.5.1 Q series-compatible E71 module
17.5.2 Built-in Ethernet port QCPU connection
17.5.3 PING test
17.5.4 Loopback test
17.6 System Monitor

18 PRINTING

18.1 Setting up Printers
18.2 Previewing Printing Images
18.3 Printing
18.3.1 Printing screen
18.3.2 Printing examples

19 SETTING OPTIONS

19.1 Basic Operations
19.2 Option Setting List

APPENDIX

Appendix 1 List of Toolbars and Shortcut Keys
Appendix 1.1 Common toolbars and shortcut keys
Appendix 1.2 Toolbar icons and shortcut keys for setting labels
Appendix 1.3 Toolbar icons and shortcut keys for setting device memory
Appendix 1.4 Toolbar icons for executing sampling trace
Appendix 1.5 Toolbar icons and shortcut keys in program editors
Appendix 1.6 Shortcut key for editing intelligent function module data
Appendix 2 Simulation Function
Appendix 2.1 Available functions
Appendix 2.2 Supported devices
Appendix 2.3 Supported instructions
Appendix 2.4 Enable/disable setting of parameter items
Appendix 2.5 Restrictions and precautions
Appendix 3 ASCII Code Table
Appendix 4 Precautions on GX Works2 and its differences with GX Developer
Appendix 4.1 Before using GX Works2
Appendix 4.2 Supported CPU modules
Appendix 4.3 Unsupported features
Appendix 4.4 Supported project types
Appendix 4.5 Programming languages supported by each project type
Appendix 4.6 Using device comments
Appendix 4.7 Using device memory
Appendix 4.8 Using device initial value
Appendix 4.9 Using online function
Appendix 4.10 Using monitor/debug function
Appendix 4.11 Using printing function
Appendix 4.12 Copying saved project data
Appendix 4.13 Compatibility with GX Developer
Appendix 4.14 Compatibility with GX IEC Developer
Appendix 4.15 Key operation
Appendix 5 Compatibility with Projects Created with Existing Applications
Appendix 5.1 Application compatibility
Appendix 6 Compatibility of Data Read from Programmable Controller CPU
Appendix 6.1 Application compatibility

Appendix 6.2 Data compatibility
Appendix 7 Character Strings that cannot be Used for Label Names and Data Names
Appendix 8 Restrictions when Changing Programmable Controller Types
Appendix 8.1 Common restrictions when changing programmable controller series/types
Appendix 8.2 Restrictions when changing to Universal model QCPUs (excluding Q02U)
Appendix 8.3 Restrictions when changing to Universal model QCPUs (Q02U)
Appendix 8.4 Restrictions when changing to Built-in Ethernet port QCPUs
Appendix 1 Restrictions when saving projects in GX Developer format
INDEX

CONTENTS

SAFETY PRECAUTIONS
REVISIONS
INTRODUCTION
CONTENTS
MANUALS
GENERIC TERMS AND ABBREVIATIONS IN THIS MANUAL

1 OVERVIEW

1.1 What is Simple Project?
1.2 Features of Simple Project
1.3 List of Functions
1.3.1 List of functions common to Simple project and Structured project
1.3.2 List of functions for editing in ladder language
1.3.3 List of functions for editing SFC diagrams
1.3.4 List of functions for editing SFC block list
1.3.5 List of functions for Structured project

2 SCREEN CONFIGURATION

2.1 Overview of Screen Configuration
2.2 Ladder Editor
2.2.1 Editing screen
2.2.2 Changing display size of editing screen
2.2.3 Changing text size on editing screen
2.2.4 Displaying/hiding comments
2.2.5 Setting number of rows and columns for displaying comments
2.2.6 Setting number of contacts to be displayed in ladder programs
2.2.7 Switching display between label names and devices
2.3 SFC Editor
2.3.1 Editing screen
2.3.2 Changing display size of editing screen
2.3.3 Displaying/hiding SFC step/transition comments
2.3.4 Setting number of columns for SFC diagram display

3 PROGRAMMING PROCEDURE

3.1 Creating Programs

4 SETTING LABELS

4.1 Label Setting Screens

4.2 Setting Global Labels
4.3 Setting Local Labels for Programs
4.4 Setting Local Labels for Function Blocks
4.5 Common Operations for Setting Labels
4.5.1 Selecting data types
4.5.2 Editing lines
4.6 Setting Structured Data Type Labels
4.6.1 Setting data type of structure
4.6.2 Setting data type as structure
4.6.3 Assigning devices to structured data type labels
4.6.4 Assigning devices to structured array type labels
4.7 Setting Ranges for Devices Assigned Automatically

5 EDITING PROGRAMS

5.1 Editing Ladder Programs
5.1.1 Creating ladders
5.1.2 Entering contacts, coils, and application instructions
5.1.3 Using function blocks
5.1.4 Drawing lines
5.1.5 Deleting contacts, coils, and application instructions
5.1.6 Deleting lines
5.1.7 Inserting and deleting rows and columns
5.1.8 Batch inserting and deleting NOP instructions
5.1.9 Cutting, copying, and pasting ladders
5.1.10 Canceling the previous operation
5.1.11 Restoring ladders to the status of after conversion
5.1.12 Precautions on editing ladders
5.2 Editing SFC Programs
5.2.1 List of SFC symbols
5.2.2 Creating SFC diagrams
5.2.3 Deleting SFC diagrams
5.2.4 Changing SFC step attributes
5.2.5 Cutting, copying, and pasting SFC diagrams
5.2.6 Sorting SFC step/transition numbers of SFC diagram
5.2.7 Redisplaying SFC diagrams
5.2.8 Creating operation outputs and transition conditions
5.2.9 Setting block information
5.2.10 Displaying SFC block list
5.2.11 Setting parameters for SFC programs

6 SEARCH AND REPLACE

6.1 Searching for and Replacing Data in Ladder Programs
6.1.1 Searching for, replacing, and batch replacing devices
6.1.2 Searching for and replacing instructions
6.1.3 Searching for contacts and coils
6.1.4 Jumping to the specified step number

6.1.5 Changing contacts between open contact and close contact
6.1.6 Changing the module start I/O numbers
6.2 Searching for and Replacing Data in SFC Programs
6.2.1 Jumping to specified SFC step number/transition number on SFC diagram
6.2.2 Jumping to specified SFC step number/block number on SFC diagram
6.2.3 Replacing SFC step number on SFC diagram
6.2.4 Searching for blocks on SFC block list
6.2.5 Searching for devices on SFC block list

7 EDITING TEXTS

7.1 Editing Device Comments
7.2 Editing Statements and Notes
7.2.1 Statements and notes
7.2.2 Entering statements
7.2.3 Modifying and deleting statements
7.2.4 Entering notes
7.2.5 Modifying and deleting notes
7.3 Batch Editing Statements and Notes
7.4 Changing the Type (PLC/Peripheral) of Statement/Note
7.5 Merge Process when Reading Programs from Programmable Controller CPU
7.5.1 Merge process
7.5.2 Performing merge process
7.6 Editing SFC Comments

8 CONVERTING PROGRAMS

8.1 Converting Ladder Blocks
8.2 Converting SFC Diagrams
8.3 Checking Programs
8.4 Compiling All Programs
8.5 Compiling Only Modified Programs
8.6 Precautions on Compilation
8.7 Executing Online Change Simultaneously with Ladder Conversion/Compilation
8.8 Checking for Errors and Warnings when Compiling Programs
8.9 Changing Operating Conditions of Compilation
8.9.1 Changing the number of errors and warnings that stops compilation
8.9.2 Hiding warning messages

9 WRITING/READING DATA TO/FROM PROGRAMMABLE CONTROLLER CPU

9.1 Writing/Reading Data to/from Programmable Controller CPU

10 MONITORING

10.1 Starting and Stopping Monitoring of Programs

10.2 Starting and Stopping Monitoring of Function Blocks

10.3 Changing Operating Conditions of Monitoring

10.3.1 Changing current value display format (decimal/hexadecimal) of word type variable

10.3.2 Monitoring buffer memory and link memory

10.3.3 Displaying/hiding current values

10.4 Monitoring Ladder Programs

10.5 Monitoring SFC Programs

10.5.1 Monitoring SFC diagrams

10.5.2 Monitoring transitions

10.5.3 Monitoring operation outputs and transition conditions

10.5.4 Batch monitoring all blocks, monitoring active steps

10.5.5 Monitoring SFC block list

11 SETTING OPTIONS

11.1 Basic Operations

11.2 Option Setting List

APPENDIX

Appendix 1 List of Toolbars and Shortcut Keys

Appendix 1.1 Common toolbars and shortcut keys

Appendix 1.2 Toolbar icons and shortcut keys for setting labels

Appendix 1.3 Toolbar icons and shortcut keys for setting device memory

Appendix 1.4 Toolbar icons for executing sampling trace

Appendix 1.5 Toolbar icons and shortcut keys in program editors

Appendix 1.6 Shortcut key for editing intelligent function module data

INDEX

■ MANUALS

Related manuals are separately issued according to the purpose of their functions in GX Works2.

● Related manuals

The manuals related to this product are shown below.
Refer to the following tables when ordering required manuals.

1) Operation of GX Works2

Manual name	Manual number (Model code)
GX Works2 Version1 Operating Manual (Common) Explains the system configuration of GX Works2 and the functions common to a Simple project and Structured project such as parameter setting, operation method for the online function. (Sold separately)	SH-080779ENG (13JU63)
GX Works2 Version1 Operating Manual (Simple Project) Explains operation methods such as creating and monitoring programs in Simple project of GX Works2. (Sold separately)	SH-080780ENG (13JU64)
GX Works2 Beginner's Manual (Simple Project) Explains fundamental operation methods such as creating, editing, and monitoring programs in Simple project for users inexperienced with GX Works2 (Sold separately)	SH-080787ENG (13JZ22)
GX Works2 Beginner's Manual (Structured Project) Explains fundamental operation methods such as creating, editing, and monitoring programs in Structured project for users inexperienced with GX Works2. (Sold separately)	SH-080788ENG (13JZ23)

2) Structured Programming

Manual name	Manual number (Model code)
QCPU Structured Programming Manual (Fundamentals) Explains the programming methods, types of programming languages, and other information required to create structured programs. (Sold separately)	SH-080782ENG (13JW06)
QCPU Structured Programming Manual (Common Instructions) Explains the specifications and functions of sequence instructions, basic instructions, and application instructions that can be used in structured programs. (Sold separately)	SH-080783ENG (13JW07)
QCPU Structured Programming Manual (Application Functions) Explains the specifications and functions of application functions that can be used in structured programs. (Sold separately)	SH-080784ENG (13JW08)
QCPU Structured Programming Manual (Special Instructions) Explains the specifications and functions of instructions for network modules, intelligent function modules, and PID control functions that can be used in structured programs. (Sold separately)	SH-080785ENG (13JW09)

Point

The Operating Manual is included in the CD-ROM with the software package.
Manuals in printed form are sold separately for single purchase. Order a manual by quoting the manual number (model code) listed in the table above.

● Purpose of this manual

This manual explains the operations for creating sequence programs in Structured project using the functions supported with GX Works2.
Manuals for reference are listed in the following table according to their purpose.
For information such as the contents and number of each manual, refer to the list of 'Related manuals'.

1) Operation of GX Works2

Purpose		GX Works2 Installation Instructions	GX Works2 Beginner's Manual		GX Works2 Version1 Operating Manual		
							
		-	Simple Project	Structured Project	Common	Simple Project	Structured Project
Installation	Learning the operating environment and installation method	Details					
Operation of Simple project	Learning the basic operations and operating procedures		Details		Outline	Outline	
	Learning the functions and operation methods for programming				Outline	Details	
	Learning all functions and operation methods except for programming				Details		
Operation of Structured project	Learning the basic operations and operating procedures			Details	Outline		Outline
	Learning the functions and operation methods for programming				Outline	Details	Details
	Learning all functions and operation methods except for programming				Details		

2) Programming

Purpose		QCPU Structured Programming Manual				QCPU(Q mode)/QnACPU Programming Manual		User's Manual for intelligent function module/ Reference Manual for network module
								
		Fundamentals	Common Instructions	Special Instructions	Application Functions	Common Instructions	PID Control Instructions	-
Programming in Simple project	Learning the types and details of common instructions, descriptions of error codes and special relays, and special registers					 Details		
	Learning the types and details of instructions for intelligent function modules							 Details
	Learning the types and details of instructions for network modules							 Details
	Learning the types and details of instructions for the PID control function						 Details	
Programming in Structured project	Learning the fundamentals for creating a Structured program for the first time	 Details						
	Learning the types and details of common instructions		 Details					
	Learning the types and details of instructions for intelligent function modules			 Details				 Details
	Learning the types and details of instructions for network modules			 Details				 Details
	Learning the types and details of instructions for the PID control function			 Details			 Details	
	Learning the descriptions of error codes, special relays, and special registers					 Details		
	Learning the types and details of application functions				 Details			

● How to read this manual

Supported CPU

Supported programmable controller CPUs are shown in icons under the section title.

Screen display

Describes the screen display procedure.
Follow the ⇒ and select [(menu)] to open the screen.
*Screen display may depend on the CPU.
In that case, typical example is described.

Operating procedure

Describes the operating procedure of the function.

Reference location

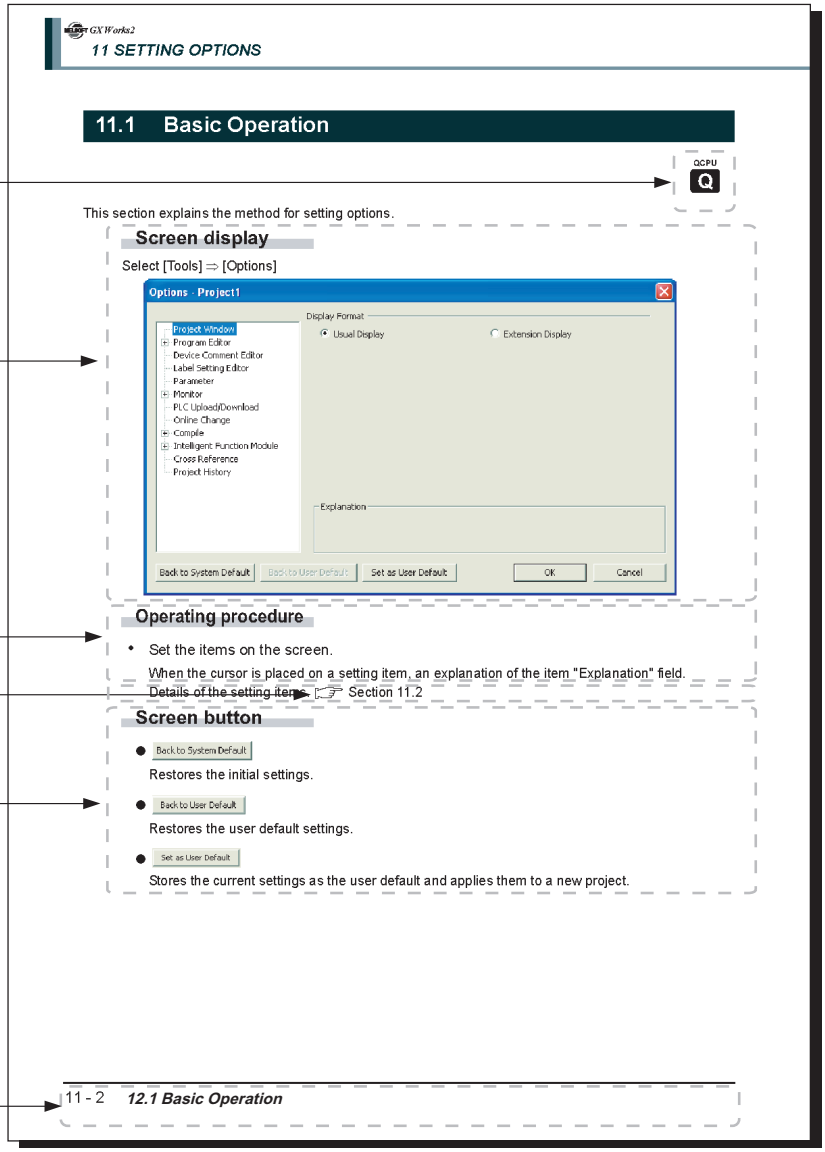
leads to the reference location and reference manual.

Screen button

Describes the buttons in the screen.

Section title

Clarifies the section of currently open page.

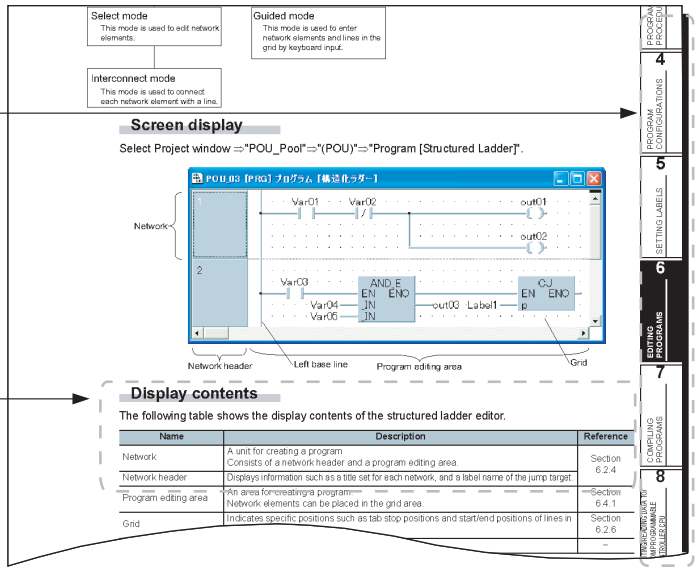


Chapter heading

Index on the right of the page number clarifies the chapter of currently open page.

Display contents

Describes the display contents in the screen.



This manual also uses the following columns:

Point

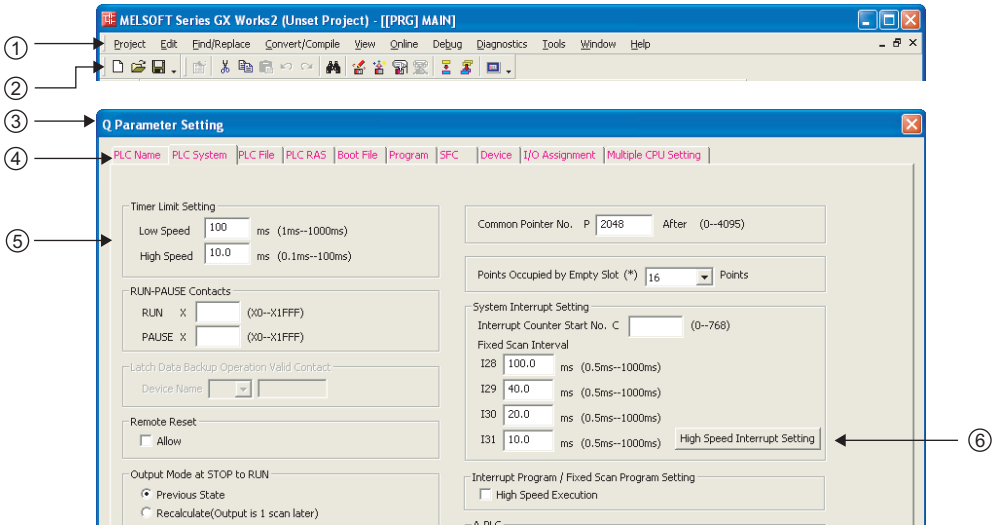
This explains notes for requring attention or useful functions relating to the information given on the same page.

Restrictions

This explains restrictions relating to the information given on the same page.

● Symbols used in this manual

The following shows the symbols used in this manual with descriptions and examples.



No.	Symbol	Description	Example
①	[ ]	Menu name on a menu bar	[Project]
②		Toolbar icon	
③	(Underline)	Screen name	Q Parameter screen
④	<< >>	Tab name in a screen	<<PLC System>>
⑤	" "	Item name in a screen	"Timer Limit Setting"
⑥		Button on a screen	 button
–		Keyboard key	Ctrl

■ GENERIC TERMS AND ABBREVIATIONS IN THIS MANUAL

This manual uses the generic terms and abbreviations listed in the following table to discuss the software packages and programmable controller CPUs. Corresponding module models are also listed if needed.

Generic term and abbreviation		Description
GX Works2		Generic product name for the SWnDNC-GXW2-E (n: version)
Existing application	GX Developer	Generic product name for the SWnD5C-GPPW-E, SWnD5C-GPPW-EA, SWnD5C-GPPW-EV, and SWnD5C-GPPW-EVA (n: version)
	GX IEC Developer	Generic product name for the SWnD5C-MEDOC3 (n: version)
Personal computer		Generic term for personal computer on which Windows® operates
High Performance model QCPU		Generic term for the Q02, Q02H, Q06H, Q12H, and Q25H
Universal model QCPU		Generic term for the Q02U, Q03UD, Q03UDE, Q04UDH, Q04UDEH, Q06UDH, Q06UDEH, Q13UDH, Q13UDEH, Q26UDH, and Q26UDEH
QCPU (Q mode)		Generic term for the High Performance model QCPU and the Universal model QCPU
Simple project		Generic term for projects created by using the ladder/SFC language
Structured project		Generic term for projects created by using the ladder/SFC/ST/structured ladder language

1 OVERVIEW

This manual explains specific operating procedures using Structured project.
For the full product features and functions of GX Works2, refer to the following manual.
☞GX Works2 Version1 Operating Manual (Common)

1.1 What is Structured project? 1-2
1.2 Features of Structured Project. 1-2
1.3 List of Functions 1-5

1.1 What is Structured project?

In Structured project, programs can be created using the structured programming.
By segmenting the control functions and creating components from commonly used parts of programs, this type of programming (structured programming) is easy to understand visually, and created components are highly reusable to other programs.

1.2 Features of Structured Project

This section explains the features of Structured project.

■ Various programming languages are available

Various programming languages are available for structured programs. Users can combine these languages by selecting the most appropriate programming language for each purpose to create programs.

Table 1.2-1Programming languages supported by Structured project

Type	Name	Description
Ladder	Ladder diagram	A graphic language using ladders composed of contacts and coils. Can be operated in a similar way to existing GX Developer operation.
ST	Structured text	A high-level structured text language with grammatical structure similar to C language or PASCAL.
SFC	Sequential function chart	A graphic language in which executing orders and executing conditions of programs are defined.
Structured ladder	Ladder diagram	A graphic language using ladders composed of contacts and coils, which can be inserted flexibly.

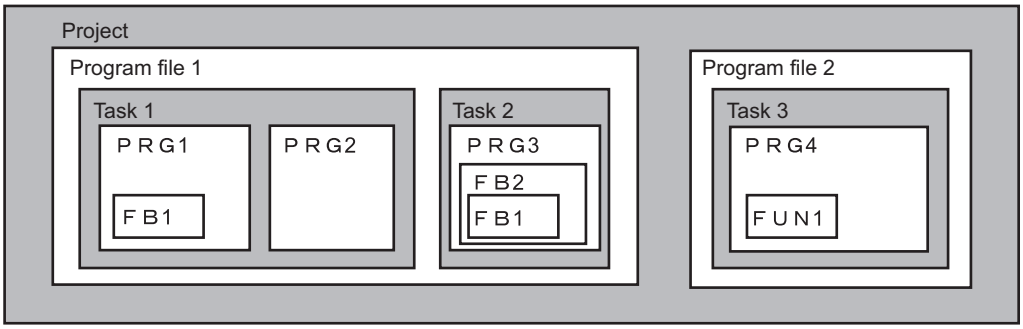
■ Common program editors for each programmable controller CPU

The program editors of Structured project can be used for any type of programmable controller CPU supported by GX Works2. Users can select the desired programming languages regardless of the target programmable controller CPU.

For details of the programmable controller CPUs supported by GX Works2, refer to the following manual.
☞GX Works2 Version1 Operating Manual (Common)

■ Creating large programs efficiently with structured programming

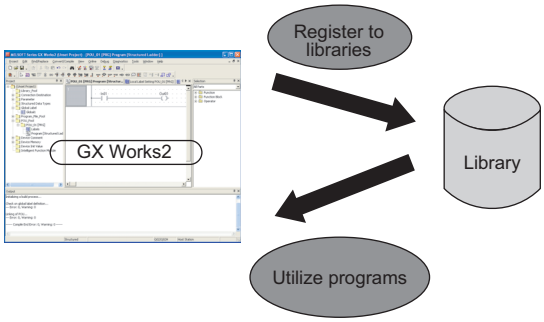
In Structured project, programs can be easily structured by managing program components in a hierarchy. *1
Programs that are easy to maintain and highly reusable can be created efficiently by structuring programs. This is suitable for developing large programs.



*1 : A program is created in units of POUs (abbreviation for Program Organization Units) such as program blocks (PRGs), functions (FUNs), and function blocks (FBs). Function blocks can be nested in a hierarchy. Created programs are registered to the tasks of program files.

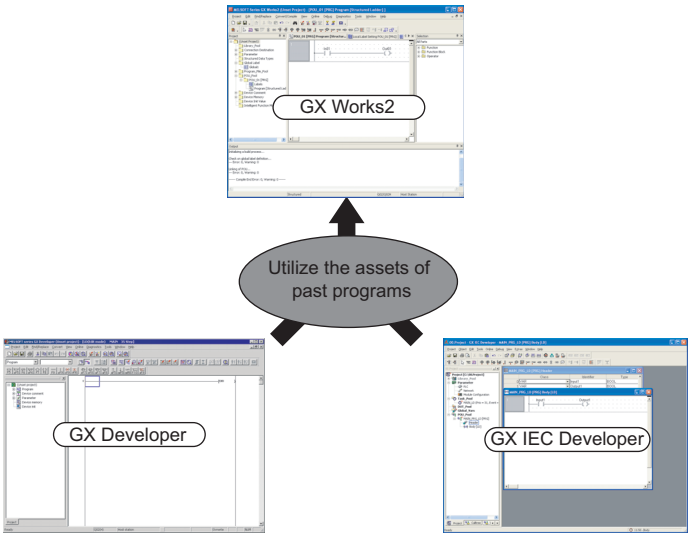
■ Assets of programs are utilized as libraries

In Structured project, sequence program components can be saved as libraries which can be utilized as assets of programs in multiple projects.



■ Assets of past programs can be utilized

In GX Works2, sequence programs created with previous versions of GX Developer and GX IEC Developer can be imported, which enables efficient utilization of program assets.



1.3 List of Functions

This section shows the list of functions of GX Works2.
The functions are divided into common functions available (Project, Online, Debug, Diagnostics, Tools, Window, and Help) and functions used for each editing and setting target (Edit, Find/Replace, Convert/Compile, and View).
For (Common) and (Simple) indicated in the Reference column, refer to the following manuals respectively:
(Common).....GX Works2 Version1 Operating Manual (Common)
(Simple).....GX Works2 Version1 Operating Manual (Simple Project)

1.3.1 List of functions common to Simple project and Structured project

This section explains the functions common to a Simple project and a Structured project.

■ List of common functions

The following tables show functions that are always available regardless of the type of editing or setting target.

Project (common function)			Reference
New		Creates a new project.	(Common)
Open		Opens an existing project.	
Close		Closes an open project.	
Save		Saves a project.	
Save As		Names and saves a project.	
Delete		Deletes an existing project.	
Verify		Verifies between two project data.	
Project Revision			-
	Revision Entry	Registers the project change history.	(Common)
	Revision List	Displays the list of project change history.	
Change PLC Type		Changes the programmable controller CPU type.	
Change Project Type		Changes the project type from a Simple project to a Structured project.	
Object			-
	New	Adds data to a project.	(Common)
	Rename	Renames the selected data.	
	Delete	Deletes the selected data.	
	Data Copy	Copies the selected data.	
	Data Paste	Pastes the copied data.	
	Set as Default Connection	Specifies data in selected connection target as a connection target for regular use.	
	Property	Displays the selected data properties.	

Project (common function)			Reference
Intelligent Function Module			-
	Add New Module	Adds new intelligent function module data.	(Common)
	Delete Module	Deletes intelligent function module data.	
	Property	Displays properties of an intelligent function module data.	
	Intelligent Function Module Parameter List	Displays a list of set/unset parameters of an intelligent function module.	
Open Other Data			
	Open Other Project	Opens a project created with GX Developer.	
	Read GX IEC Developer Data	Reads a project created with GX IEC Developer.	
	Read GX Configurator-QP Data	Reads a project created with GX Configurator-QP.	
Export to GX Developer Format File		Saves an open project in GX Developer format.	

Project (common function)			Reference
Library			Chapter 10
	Create	Creates a new library.	
	Install	Installs a created library to the project.	
	Deinstall	Deletes a library from the project.	
	Update Library	Updates the library imported to the project.	
	Rename	Renames a library.	
	Open	Enables editing of a library.	
	Close	Disables editing of a library.	
	Change Password	Sets a password for a library.	
	Save as	Names and saves a project.	
	Save	Saves a library file.	
	Delete	Deletes a library file.	
	Help	Displays help information of the library.	
Security Level			-
	Change Password	Changes the password of the current login user.	(Common)
	User Management	Manages user information of the project. Adds/deletes a user or changes the user information.	
	Data Security Setting	Sets the access authorization of each user related to reading/writing data.	
Print Window		Prints the open screen.	
Print Window Preview		Displays the print preview of the open screen.	
Printer Setup		Changes the printer settings.	
The Latest File		Displays the recently used GX Works2 project path and opens the selected project.	
Quit		Exits GX Works2.	

Edit (common function)		Reference
Cut	Cuts the selected data.	-
Copy	Copies the selected data.	
Paste	Pastes the cut or copied data at the cursor position.	
Delete	Deletes the selected data.	

Find/Replace (common function)		Reference
Find	Searches for a character string.	(Common)
Replace	Replaces the character string.	
Cross Reference	Displays the usage of selected device or label.	
List of Used Device	Displays the usage of devices.	

Convert/Compile (common function)		Reference
Build	Compiles (converts) the program being edited.	Section 7.2
Compile + Online Change	Writes a sequence program to the programmable controller CPU after compilation.	(Common)
Rebuild All	Compiles (converts) all programs in the project.	Section 7.1
Change SFC All	Converts all SFC programs in the project.	(Simple)

View (common function)			Reference
Toolbar			-
	Toolbar name	Displays/hides the tool bar.	Appendix 1
Statusbar		Displays/hides the statusbar.	(Common)
Colors		Sets the font color such as labels, device comments on the work window.	
Docking Window			-
	Project Window	Displays/hides the Project window. In the Project window, the data in the open project is displayed in a list.	(Common)
	Function Block Selection Window	Displays/hides the Function Block Selection window. Select a part such as FB or function on the window for utilizing it to a program.	Section 6.2.1
	Output Window	Displays/hides the Output window. In the window, the conversion (compilation) result is displayed.	Section 7.5
	Cross Reference Window	Displays/hides the Cross Reference window. In the Cross Reference window, devices/labels used in the project are displayed.	(Common)
	List of Used Device Window	Displays/hides the Device List window. In the Device List window, the usage of selected device is displayed.	
	Watch Window 1 to 4	Displays/hides the Watch window. In the Watch window, the monitoring result is displayed.	
	Find/Replace Window	Displays/hides the Find/Replace window. In the Find/Replace window, the search/replace result is displayed.	

Online (common function)			Reference
Read from PLC		Reads data from the programmable controller CPU.	(Common)
Write to PLC		Writes data to the programmable controller CPU.	
Verify with PLC		Verifies a project being edited against the data in the programmable controller CPU.	
Start/Stop PLC		Remotely controls RUN/PAUSE/STOP of the programmable controller CPU from GX Works2.	
PLC Keyword			-
	New	Sets a password to the programmable controller CPU.	(Common)
	Delete	Cancels the password set to the programmable controller CPU.	
	Disable	Temporarily unlocks the password set to the programmable controller CPU.	
PLC Memory Operation			
	Format PLC Memory	Formats the programmable controller CPU memory.	
	Clear PLC Memory	Clears the programmable controller CPU memory.	
	Arrange PLC Memory	Arranges the programmable controller CPU memory.	
Delete PLC Data		Deletes data in the programmable controller CPU.	
PLC User Data			
	PLC User Data Read	Reads the programmable controller CPU user data.	
	PLC User Data Write	Writes the programmable controller CPU user data.	
	PLC User Data Delete	Deletes the programmable controller CPU user data.	
Export to ROM Format		Copies the program memory data in the programmable controller CPU to ROM.	
Program Memory Batch Download		Transfers the contents of program cashe memory to program memory in a batch.	
Latch Data Backup			-
	Backup	Backs up device memory/file register/error history data in the Universal model QCPU to the standard ROM.	(Common)
	Delete Backup Data	Deletes the backup data in the Universal model QCPU.	
Set Clock		Sets the clock in the programmable controller CPU.	

Online (common function)			Reference
Monitor			-
	Start Monitoring (All Windows)	Starts monitoring of all open windows.	(Common)
	Stop Monitoring (All Windows)	Stops monitoring of all open windows.	
	Start Monitoring	Starts monitoring of the open window.	Section 9.1
	Stop Monitoring	Stops monitoring of the open window.	
	Start Watch	Starts monitoring the current values of registered devices/ labels.	(Common)
	Stop Watch	Stops monitoring the current values of registered devices/ labels.	
	Change Value Format (Decimal)	Displays the current device value in decimal in program monitoring.	Section 9.3.1
	Change Value Format (Hexadecimal)	Displays the current device value in hexadecimal in program monitoring.	
	Device/Buffer Memory Batch Monitor	Monitors device/buffer memory in a batch.	(Common)
	Change Instance (Function Block)	Selects an instance of the function block to be monitored.	Section 9.2
	SFC All Block Batch Monitor	Batch monitors all blocks in the SFC program.	(Simple)
Register to Watch Window		Registers the selected devices to the Watch window.	(Common)
Modify Value		Changes the ON/OFF status and values of devices and labels used in the program.	

Debug (common function)			Reference
Start/Stop Simulation		Starts/stops simulation.	(Common)
Show Restricted Instructions		Displays a list of the instructions and devices used in the program unsupported for the simulation function.	
Sampling Trace			-
	Open Sampling Trace	Executes sampling trace.	(Common)

Diagnostics (common function)		Reference
PLC Diagnostics	Diagnoses the operating status of the programmable controller CPU.	(Common)
Ethernet Diagnostics	Diagnoses the Ethernet	
CC-Link IE Control Diagnostics	Diagnoses the CC-Link IE controller network.	
MELSECNET Diagnostics	Diagnoses the MELSECNET/10(H).	
CC-Link / CC-Link/LT Diagnostics	Diagnoses the CC-Link and CC-link/LT.	
System Monitor	Monitors the system status of the programmable controller CPU.	

Tools (common function)			Reference
IC Card			-
	Read IC Card	Reads data from the IC memory card.	(Common)
	Write IC Card	Writes data to the IC memory card.	
Program Check		Checks programs and displays errors.	(Simple)
Parameter Check		Checks parameters and displays errors.	(Common)
Options		Sets various options.	Chapter 11
System Label Setting		Sets device range to be automatically assigned to a label.	Section 5.7
Block Password		Sets a block password to the data.	Common
Intelligent Function Module Parameter Check			-
	Auto Refresh Duplication Check	Checks the duplication of devices set in the Auto refresh function and displays the result.	(Common)

Window (common function)		Reference
Cascade	Tiles windows in overlapping display.	(Common)
Tile Vertically	Tiles windows vertically.	
Tile Horizontally	Tiles windows horizontally.	
Arrange Icons	Arranges the icons at the bottom of the window.	
Close All	Closes all open windows.	
(Switch to Other Window)	Displays the open window.	
Window	Displays a list of open windows. Also opens or arranges specified windows.	

Help (common function)		Reference
PLC Error	Displays the explanation for each CPU error code.	(Common)
Special Relay/Register	Displays the explanation for each special relay and special register.	
Changes from GX Developer	Displays the changes from GX Developer to GX Works2.	
Operating Manual		-
GX Works2 Beginner's Manual (Simple Project)	Displays the operating manuals.	(Common)
GX Works2 Beginner's Manual (Structured Project)		
GX Works2 Operating Manual (Common)		
GX Works2 Operating Manual (Simple Project)		
GX Works2 Operating Manual (Structured Project)		
Connect to MELFANSweb	Connects to the MELFANSweb website.	
About GX Works2	Displays product information such as the version.	

■ List of functions for setting labels

The following table shows the functions available for setting and editing labels.

Edit (function for label setting)		Reference
Select All	Selects all items.	Section 5.5.2
Expand Declaration	Expands the collapsed items.	
Collapse Declaration	Collapses the expanded items.	
New Declaration (Before)	Adds a line above the cursor position.	
New Declaration (After)	Adds a line below the cursor position.	
Delete Line	Deletes a line at the cursor position.	

■ List of functions for editing device comments

The following table shows functions available for editing device comments.

Edit (function for editing device comments)		Reference
Undo	Restores the previous processing status.	(Common)
Redo	Restores the processing deleted with [Undo].	
Select All	Selects all data being displayed.	
Import from Sample Comment		
Special Relay/Register	Utilizes sample comments of SM/SD.	
Intelligent Function Module	Utilizes sample comments of intelligent function module device.	
Clear All	Deletes all device comment data.	

■ List of functions for setting device memory

The following tables show the functions available for setting device memory.

Edit (function for setting device memory)		Reference
Paste Text	Enters a character string.	(Common)
Insert Row	Inserts a row at the cursor position.	
Insert Device	Enters a device.	

Find/Replace (function for setting device memory)		Reference
Find Device	Searches for a device.	(Common)

View (function for setting device memory)			Reference
Display Mode			-
	Binary	Displays data in binary.	(Common)
	Octal	Displays data in octal.	
	Decimal	Displays data in decimal.	
	Hexadecimal	Displays data in hexadecimal.	
	Float	Displays data in real number.	
	ASCII	Displays data in character string.	
Register			-
	16-bit	Displays data in units of words.	(Common)
	32-bit	Displays data in units of double words.	
	64-bit	Displays data in units of 64 bits.	
Setup		Changes the editor size.	

Tools (function for setting device memory)		Reference
Upload Device Memory from PLC	Reads device memory data from the programmable controller CPU.	(Common)
Download Device Memory to PLC	Writes device memory data to the programmable controller CPU.	
Import from Excel File	Reads data from an Excel file.	
Export to Excel File	Writes data to an Excel file.	

■ List of functions for executing sampling trace

The following tables show the functions available for executing sampling trace.

View (function for executing sampling trace)			Reference
Result Display Position			(Common)
	Move to Trigger Position	Displays the trigger position.	
Switching Display Items			-
	Device	Display/hides the display item titles.	(Common)
	Address		
	Comment		
	Data Types		
	Radix		
Timing Chart Scale			-
	Narrow Scale	Changes the display width of the timing chart scale.	-
	Wide Scale		
Trend Graph Scale			
	Narrow Scale	Changes the display width of the trend graph scale.	(Common)
	Wide Scale		
	Initial Display		
Additional Information			-
	Past Time	Display/hides the additional information.	(Common)
	Program Name		

Debug (function for executing sampling trace)			Reference
Sampling Trace			-
	Trace Settings	Displays the screen on which the sampling trace conditions are set.	(Common)
	Start Trace	Starts sampling trace.	
	Stop Trace	Suspends sampling trace.	
	Execute Manual Trigger	Generates a trigger at any given timing.	
	Register Trace	Writes trace settings to the programmable controller CPU. Execute this to control the sampling trace start by a sequence program. Sampling trace starts when the Trace start command (SM801) is executed after executing "Register Trace".	
	Forced Execution Registration Valid	Enables to execute the sampling trace from the peripherals for the Universal model QCPUs.	
	Display Trace Buffer Condition	Displays trace data storage status.	
	Export CSV Data	Saves the trace data (trace settings + results) to a personal computer in CSV file format.	
	Upload from PLC	Writes the sampling trace data (trace settings + results) to the programmable controller CPU.	
	Download to PLC	Reads the trace data (trace settings + results) from the programmable controller CPU.	
	Delete All Data	Deletes all information including the device data registered and the sampling trace result displayed on the Sampling Trace screen.	

1.3.2 List of functions for editing in ladder language

For the functions available for editing in the ladder language, refer to the following manual.
☞GX Works2 Version1 Operating Manual (Simple Project)

1.3.3 List of functions for editing SFC diagrams

For the functions available for editing SFC diagrams, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Simple Project)

1.3.4 List of functions for editing SFC block list

For the functions available for editing SFC block list, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Simple Project)

1.3.5 List of functions for editing in ST language

The following tables show the functions available for editing with the ST editor.

Edit (function for editing in ST language)		Reference
Undo	Restores the previous processing status.	Section 6.2.5
Redo	Restores the processing deleted with [Undo].	
List Operands	Displays the screen for inserting a label by selecting an existing label.	Section 6.2.3
New Label	Displays the screen for inserting a label by adding a new label setting.	
Create Template	Displays a template of argument defined for each instruction/ function.	Section 6.2.2

Find/Replace (function for editing in ST language)			Reference
Bookmark			-
	Toggle Bookmark	Sets a bookmark at the cursor row. The bookmark is deleted when one is already set at the cursor row.	Section 6.3.4
	Next Bookmark	Displays the next bookmark position.	
	Previous Bookmark	Displays the previous bookmark position.	
	Delete All Bookmarks	Deletes all bookmarks.	

View (function for editing in ST language)			Reference
Zoom			-
	50%	Displays the screen zoomed out to 50%.	Section 6.2.7
	75%	Displays the screen zoomed out to 75%.	
	100%	Displays the screen at the normal size.	
	150%	Displays the screen zoomed in to 150%.	
Zoom Header/Body			-
	Header	Opens the label setting screen in the selected POU.	Section 6.2.8
	Body	Opens the program editor in the selected POU.	
Open Header		Opens the label setting screen for the program being edited.	Section 6.2.9

Online (function for editing in ST language)			Reference
Monitor			Section 9.4
	Start Monitoring	Starts monitoring.	
	Split Window	Starts monitoring in the split window format that displays the monitor data of numerical values and character strings.	

1.3.6 List of functions for editing in structured ladder language

The following tables show the functions available for editing with the structured ladder editor.

Edit (function for editing in structured ladder language)			Reference
Undo		Restores the previous processing status.	Section 6.2.5
Redo		Restores the operation deleted with [Undo].	
Select Mode		Changes to the contact and coil input mode.	Section 6.4.3
Interconnect Mode		Changes to the line drawing mode.	Section 6.4.4
Auto Connect		Specifies and connects the start and end points to draw a line.	Section 6.4.3 Section 6.4.4
Guided Mode			-
	Guided Editing	Changes to the keyboard input mode.	Section 6.4.12
	Overwrite Mode	Overwrites the network element entered in Guided editing at the cursor position.	
	Insert Mode	Inserts the network element entered in Guided editing at the cursor position.	
	Line Mode	Changes the input format to draw lines in Guided editing.	
	Auto Comment	Adds a comment entry field at the start of the network added in Guided editing.	
Recalculate Line		Arranges a line automatically to redraw it.	Section 6.4.4
Insert Row		Inserts a row in a ladder being edited.	Section 6.4.5
Insert Column		Inserts a column in a ladder being edited.	
New Network			-
	Top	Inserts a new network at the start of all networks.	Section 6.2.4
	Before	Inserts a new network in front of a network being edited.	
	After	Inserts a new network after a network being edited.	
	Bottom	Inserts a new network at the end of all networks.	
Network Elements			-
	Contact	Inserts at the cursor position.	Section 6.4.2
	Contact Negation	Inserts at the cursor position.	
	Coil	Inserts at the cursor position.	
	Jump	Inserts at the cursor position.	
	Return	Inserts at the cursor position.	
	L-Connect Contact	Inserts at the cursor position.	
	L-Connect Contact Negation	Inserts at the cursor position.	
	Input Variable	Inserts at the cursor position.	
	Output Variable	Inserts at the cursor position.	
	Horizontal Line	Inserts at the cursor position.	
	Vertical Line	Inserts at the cursor position.	
	Comment	Inserts a comment entry field at the cursor position.	
	Network Label	Displays the Network Header screen.	

Edit (function for editing in structured ladder language)			Reference
List Operands		Displays the screen for inserting a label by selecting an existing label.	Section 6.2.3
New Label		Displays the screen for inserting a label by adding a new label setting.	
Number of Pins			-
	Increment	Adds the number of arguments of functions and function blocks.	Section 6.4.7
	Decrement	Deletes the number of arguments of functions and function blocks.	
Network List		Displays a list of networks in a program.	Section 6.2.4
Signal Configuration			-
	Configure	Sets the type of a contact and a coil.	Section 6.4.6
	Toggle	Changes the contact and coil type per execution in the following order. • Contact → Contact negation • Coil → Reversal coil → Set → Reset	

View (function for editing in structured ladder language)			Reference
View Mode			-
	Label	Displays a variable by its entered format.	Section 6.4.10
	Device	Displays a device/address in device format. The variable is displayed as an assigned device in device format.	
	Address	Displays a device/address in address format. The variable is displayed as an assigned device in address format.	
	Comment	Displays a label comment.	
	Change Display Label-Device-Address	Switches the display format in order (label, device, address).	
	Change Display Label-Comment	Switches the display between label and comment.	
Grid		Shows a grid on the screen being edited to display the start/end positions of a line.	Section 6.2.6
Zoom			-
	50%	Displays the screen zoomed out to 50%.	Section 6.2.7
	75%	Displays the screen zoomed out to 75%.	
	100%	Displays the screen at the normal size.	
	150%	Displays the screen zoomed in to 150%.	
Zoom Header/Body			-
	Header	Opens the label setting screen in the selected POU.	Section 6.2.8
	Body	Opens the program editor in the selected POU.	
Open Header		Opens the label setting screen for a program being edited.	Section 6.2.9

2 SCREEN CONFIGURATION

This chapter explains the screen configuration of GX Works2.

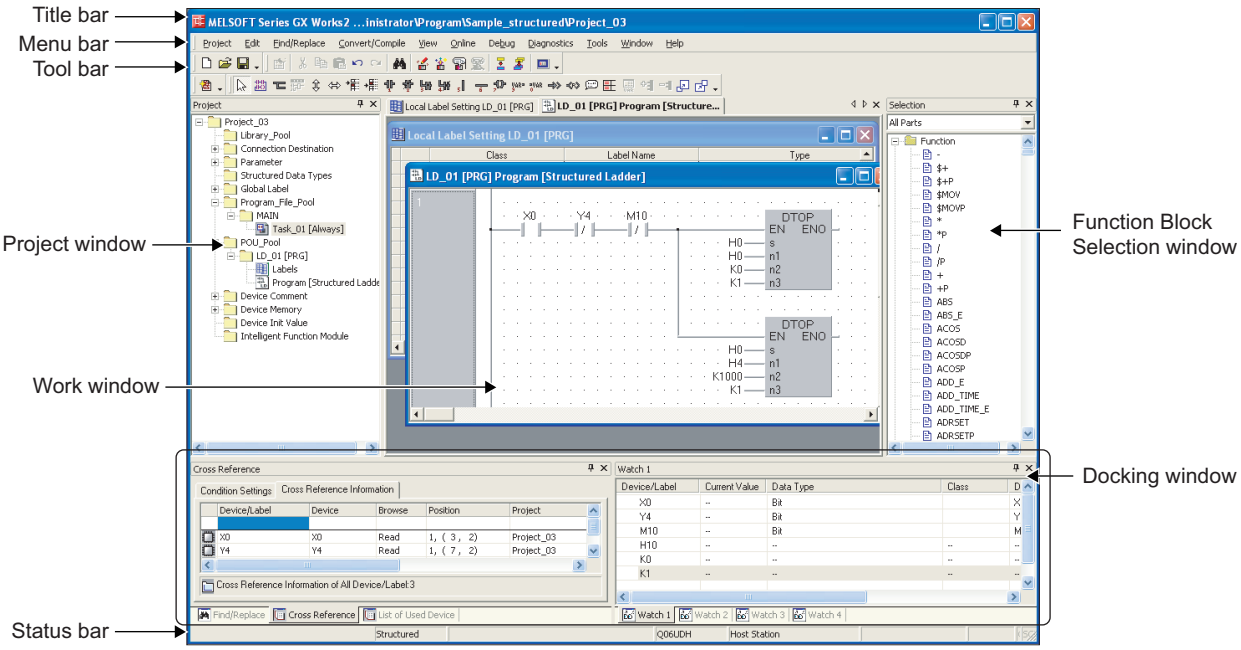
2.1 Overview of Screen Configuration 2-2

2.1 Overview of Screen Configuration

QCPU
Q

This section explains the main frame (basic screen) of GX Works2 that is displayed when it is started up.
The following shows the configuration of the main frame.

Screen display



Display contents

Name	Description	Reference
Title bar	Displays a project name.	–
Menu bar	Displays menu options for executing each function.	–
Toolbar	Displays tool buttons for executing each function.	Appendix 1
Work window	A main screen used for operations such as programming, parameter setting, and monitoring.	GX Works2 Version1 Operating Manual (Common)
Docking window	A sub screen to support operations performed on a work window.	
Project window	Displays contents of a project in tree format.	Section 4.1
Function Block Selection window	Displays a list of functions (such as function blocks) used for programming.	Section 6.2.1
Output window	Displays compilation and check results (errors and warnings).	Section 7.5
Cross Reference window	Displays cross reference results.	GX Works2 Version1 Operating Manual (Common)
List of Used Device window	Displays the device usage list.	
Watch window 1 to 4	A screen used for monitoring and changing current device values.	
Find/Replace window	A screen used for searching and replacing character strings in the project.	
Status bar	Displays information about a project being edited.	

3 PROGRAMMING PROCEDURE

This chapter explains the method for creating programs in Structured project.

3.1 Creating Programs 3-2

3.1 Creating Programs

QCPU
Q

This section explains the general operating steps from the creation of a program in Structured project to the execution of the created program in the programmable controller CPU.

1. Creating a new project

Procedure	Reference
Start up GX Works2.	GX Works2 Version1 Operating Manual (Common)
Create a new Structured project. To reuse an existing Structured project, open that Structured project.	

2. Setting parameters

Procedure	Reference
Set parameters.	GX Works2 Version1 Operating Manual (Common)

3. Configuring programs

Procedure	Reference
Create program files.	Chapter 4
Create tasks.	
Create POUs.	
Register POUs (program blocks) to the tasks.	

4. Setting labels

Procedure	Reference
Define global labels.	Chapter 5
Define local labels.	

(To the next page)

(From the previous page)

5. Editing programs

Procedure	Reference
Edit programs of each POU.	Chapter 6

6. Compiling programs

Procedure	Reference
Compile all programs.	Chapter 7

7. Connecting a personal computer to the programmable controller CPU

Procedure	Reference
Connect a personal computer to the programmable controller CPU.	GX Works2 Version1 Operating Manual (Common)
Set the connection target.	

8. Writing data to the programmable controller CPU

Procedure	Reference
Write parameters to the programmable controller CPU.	Chapter 8
Write sequence programs to the programmable controller CPU.	

9. Confirming the operation

Procedure	Reference
Monitor the execution status of sequence programs.	Chapter 9

10. Ending the project

Procedure	Reference
Save the project.	GX Works2 Version1 Operating Manual (Common)
Exit GX Works2.	

MEMO

4 PROGRAM CONFIGURATIONS

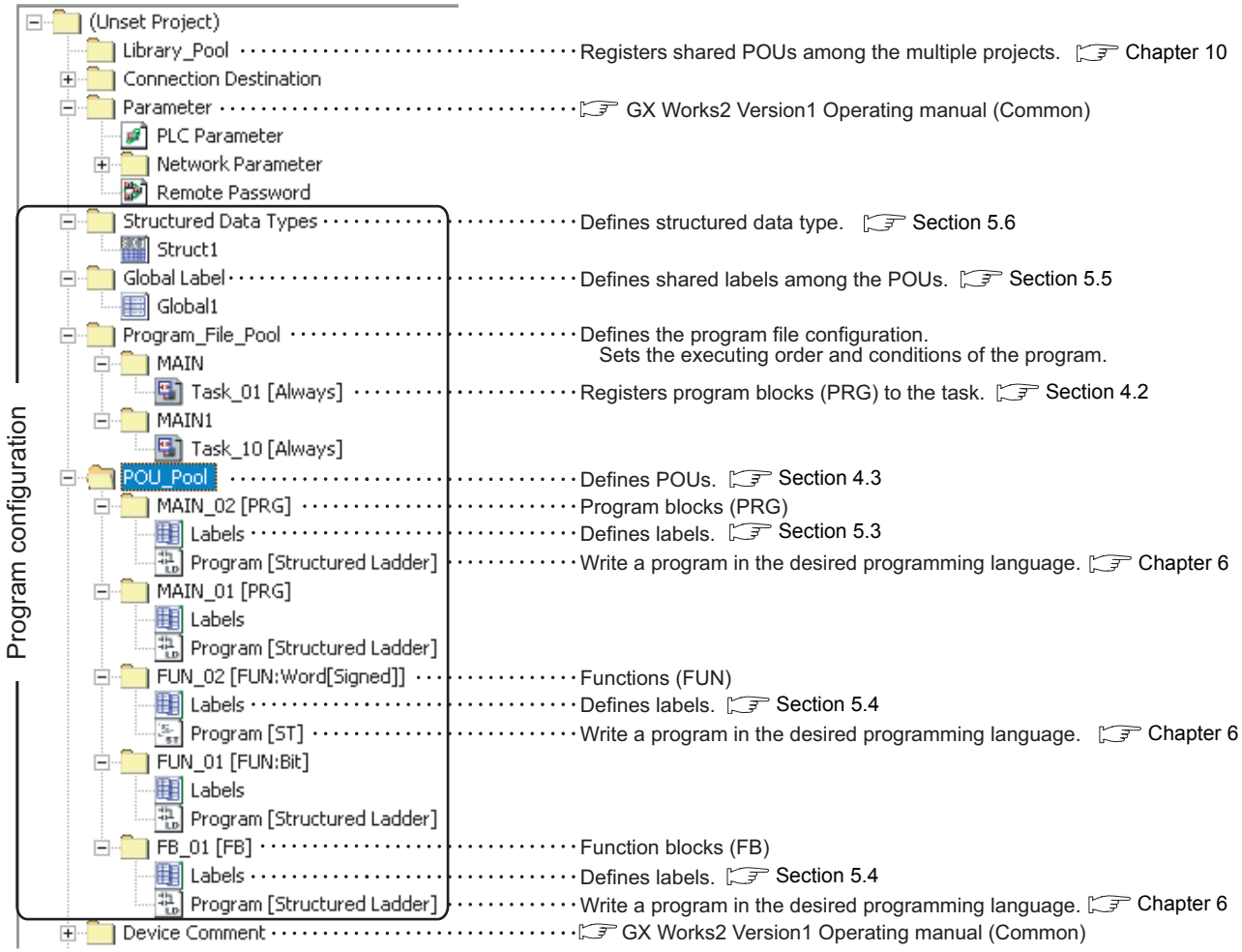
In Structured project, sequence programs are configured by using tasks and POUs.
This chapter explains the program configurations of Structured project.

4.1 Program Configurations of Structured Project 4-2
4.2 Creating Program Files and Tasks. 4-3
4.3 Creating POUs 4-9
4.4 Using POUs 4-13

4.1 Program Configurations of Structured Project

QCPU
Q

This section explains the configurations of Structured project that are displayed in a tree format in the Project window.



4.2 Creating Program Files and Tasks

QCPU
Q

This section explains the method for registering program blocks (PRGs) executed on the programmable controller CPU and setting the executing conditions of programs using program files and tasks.

4.2.1 Procedure for creating program files and tasks

This section explains the procedure for creating a program file and task.

1. Creating a program file and task data

Procedure	Reference
Create a program file. • Select [Project] ⇒ [Object] ⇒ [New].	GX Works2 Version1 Operating Manual (Common)
Create a task. • Select the program file folder and perform the following operation. Select [Project] ⇒ [Object] ⇒ [New].	

⇩

2. Registering program blocks to the task

Procedure	Reference
Display the Task Setting screen. • Select Project window ⇒ "Program_File_Pool" ⇒ "(program file)" ⇒ "(task)".	GX Works2 Version1 Operating Manual (Common)
Set program blocks to be registered to the task.	Section 4.2.2

⇩

3. Setting the executing condition

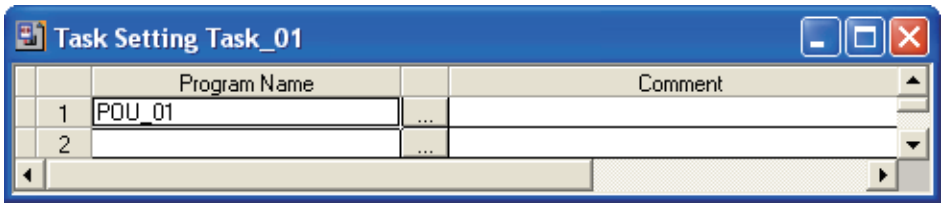
Procedure	Reference
Set the executing condition of the program file on the <<Program>> tab of PLC parameter.	GX Works2 Version1 Operating Manual (Common)
Set the executing condition on the Property screen of the task. • Select the task data and perform the following operation. Select [Project] ⇒ [Object] ⇒ [Property].	Section 4.2.3

4.2.2 Registering program blocks to tasks

This section explains the method for registering program blocks to a task and specifying the executing order. Only the program blocks that are registered to the task are compiled.
(Compiling programs ☞ Chapter 7)

Screen display

Select Project window ⇒ "Program_File_Pool" ⇒ "(program file)" ⇒ "(task)".



Operating procedure

- Set the items on the screen.

Item	Description	Number of characters
Program Name	Enter a program block name. Click [...] to set a program block name on the Program Selection screen.	0 to 32 characters
Comment	Enter comments for the program block. A new line can be inserted in a cell by pressing the Ctrl + Enter keys.	0 to 1024 characters

Point

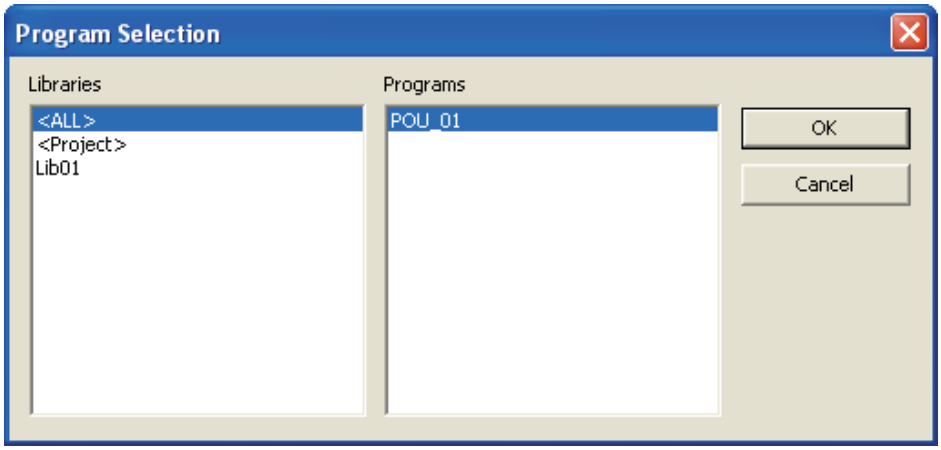
- **Program blocks that can be registered**
  - A single program block can be registered to any one of tasks for only once. Only the program blocks that are not registered to any of the tasks are displayed on the Program Selection screen.
- **When registering a ladder program to a task**
  - Multiple tasks cannot be created for a single program file, and multiple ladder programs cannot be registered to a task.
- **When registering a SFC program to a task**
  - Multiple tasks cannot be created for a single program file. Only program blocks of SFC program can be registered to a task.
- **Executing order of program blocks registered to a task**
  - Structured ladder or ST programs are executed in the order they are registered to a task.
    SFC program is executed in the order of block number, disregarding the order they are registered to the task.
- **Maximum number of program blocks to be registered**
  - The maximum number of program blocks that can be registered to a single task is 320.
  - The maximum number of tasks that can be created for a single project is 124.
  - The maximum number of program blocks that can be registered to tasks for a single project is 800.

■ Setting program names on the program selection screen

The following explains the method for setting a program block on the Task Setting screen using the Program Selection screen.

Screen display

On the Task Setting screen, click [...] next to the "Program Name" entry field.



Operating procedure

1. **Select the reference source of the program block from the "Libraries" field.**

Libraries	Description
<ALL>	Browses all program blocks defined in the project and libraries.
<Project>	Browses program blocks defined in the project.
(Library name)	Browses program blocks defined in the specified library.

2. **Select a program block from the "Programs" field.**

3. **Click the [OK] button when the setting is completed.**
   The selected program block is displayed on the Task Setting screen.

4.2.3 Setting executing conditions

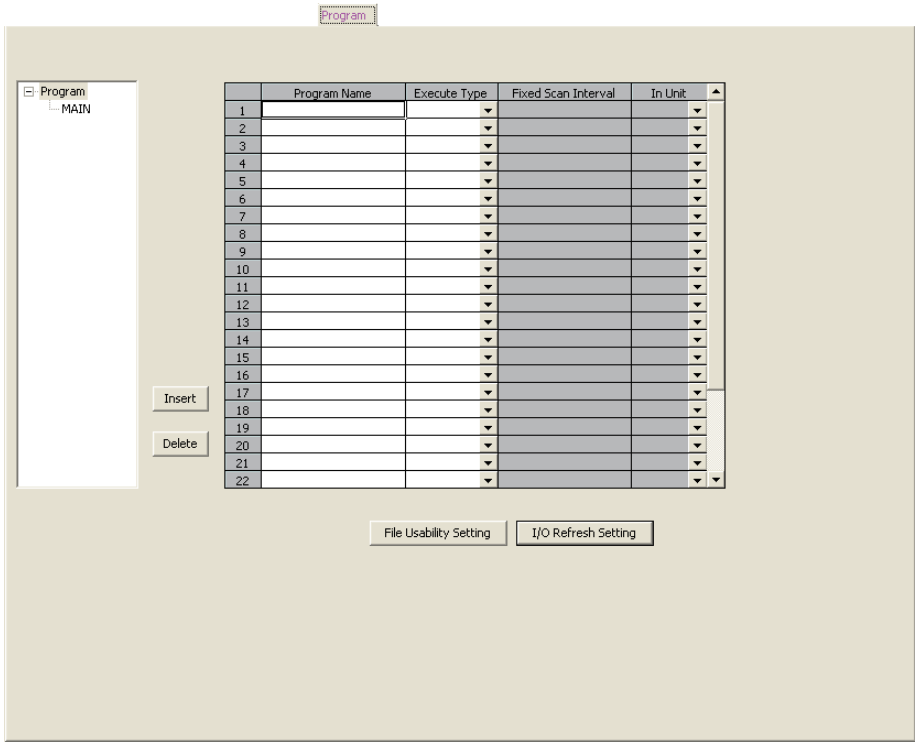
This section explains the method for setting executing conditions for program files and tasks.

■ Executing conditions for program files

The executing conditions for program files are set on the <<Program>> tab of PLC parameter.

Screen display

Select Project window ⇒ "Parameter" ⇒ "PLC Parameter" ⇒ <<Program>>.



For details of the program executing conditions, refer to the following manual.

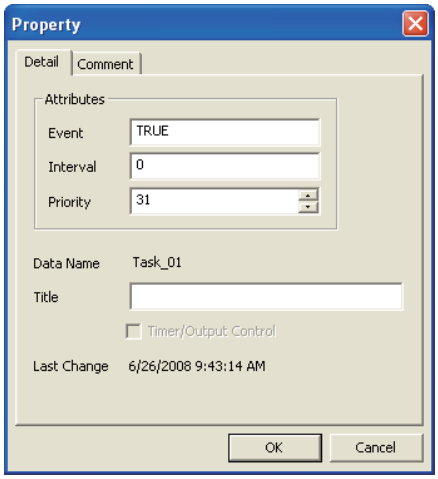
☞ QCPU User's Manual (Function Explanation, Program Fundamentals)

■ Executing conditions for tasks

The executing conditions for tasks are set on the Property screen of the task.
Select the task to be set in the Project window in advance.

Screen display

Select [Project] ⇒ [Object] ⇒ [Property].



Operating procedure

- Set the items on the screen.

Item			Description	Execution type
Attributes	Event	TRUE	Executes the task in every scan.	Scan execution
		FALSE	Executes the task in the cycle specified in the "Interval" field.	Fixed scan execution
		Device or label name	Executes the task when the specified device or label is TRUE.	Event execution
	Interval		Set the cycle time to execute the task in a constant cycle. (Enter FALSE in the "Event" field.) Set the cycle time in time type format (Ex.: #T100ms, #T24d20h31m23s647ms). QCPU Structured Programming Manual (Application Functions) Do not set shorter time than the scan time of sequence program.	
	Priority		Set the task execution priority in the range from 0 to 31. Tasks with smaller values are executed by priority. Tasks of the same priority are executed in the alphabetical order of task data names.	
Title			Enter a title of the task.	
Timer/ Output Control		Checked	The outputs and the current values of timers in the task are reset when the task is not executed.	
		Not checked	The outputs and the current values of timers in the task are retained when the task is not executed.	

Point

- **Interruption start**
  To enable an interruption start, set an interrupt pointer number (I0 to I31) in the "Event" field. In this case, create a program using the basic instruction EI (Enable Interrupt) for the program blocks registered to scan execution tasks.
- **When the program block registered to the task is created in the ladder language**
  When the program block registered to the task is created in the ladder language, items of "Attributes" cannot be set.
- **When the program end instruction is used**
  When the program end instruction such as the FEND or GOEND instruction is used, program blocks and/or tasks following the program end instruction are not executed.

The following tables show the task executing order according to the set execution type (scan execution (priority: high/low), fixed scan execution or event execution) in the task property setting.

Task name	Task attributes			Execution type
	Event	Interval	Priority	
Task1	TRUE	0	31	Scan execution (priority: low)
Task2	M0	0	31	Event execution
Task3	FALSE	T#100ms	31	Fixed scan execution
Task4	TRUE	0	30	Scan execution (priority: high)

Executing order	Task name	Execution timing
1	Task4	Executed in every scan
2	Task1	Executed in every scan
3	Task2	When M0 turned ON
4	Task3	In every 100ms

4.3 Creating POUs

QCPU
Q

This section explains the method for creating POUs.
A POU (abbreviation for Program Organization Unit) is a unit of program defined for each function.

4.3.1 Procedure for creating POUs

This section explains the procedure for creating a POU.

1. Creating a new POU

Procedure	Reference
Create a POU. • Select [Project] ⇒ [Object] ⇒ [New].	GX Works2 Version1 Operating Manual (Common)

⇩

2. Setting local labels

Procedure	Reference
Display the Local Label Setting screen or Function/FB Label Setting screen. • Select Project window ⇒ "POU_Pool" ⇒ "(POU)" ⇒ "Labels".	Chapter 5
Set labels that are used only within the POU.	

⇩

3. Editing the program

Procedure	Reference
Display the program editor. • Select Project window ⇒ "POU_Pool" ⇒ "(POU)" ⇒ "Program".	Chapter 6
Edit the program in the specified programming language.	

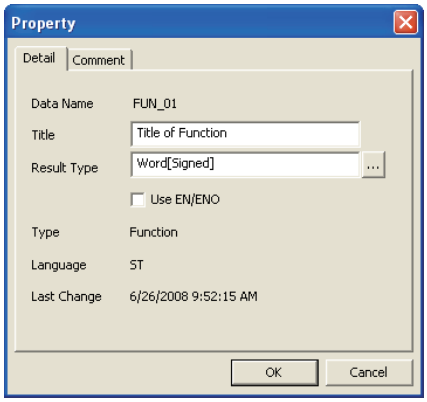
4.3.2 Setting properties of functions and function blocks

This section explains the method for setting the items on the Property screen of a function and function block.
Select a function or function block to be set in the Project window in advance.

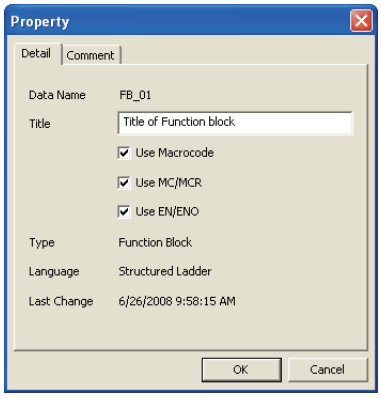
Screen display

Select [Project] ⇒ [Object] ⇒ [Property].

<For function>



<For function block>



Operating procedure

- Set the items on the screen.

Item		Description
Result Type		Set the data type of return value of the function. Click ▼ and select a data type from the list.
Use Macrocode	Checked	Applies function block program codes to each location where instances are used at compilation. Even if EN of the function block is OFF, the output variable is retained at ON. To turn OFF the output variable when EN is OFF, select the "Use MC/MCR" check box below.
	Not checked	Creates function block program codes for each function block at compilation. Codes are created to call function blocks for each location where instances are used. To call function blocks, use automatically assigned devices. (☞ Section 5.7)
Use MC/MCR	Checked	Applies function block program codes using the MC/MCR instruction to each location where instances are used at compilation.
	Not checked	Applies function block program codes using the CJ instruction to each location where instances are used at compilation.
Use EN/ENO	Checked	Becomes a function or function block with EN/ENO.
	Not checked	Becomes a function or function block without EN/ENO.

Point

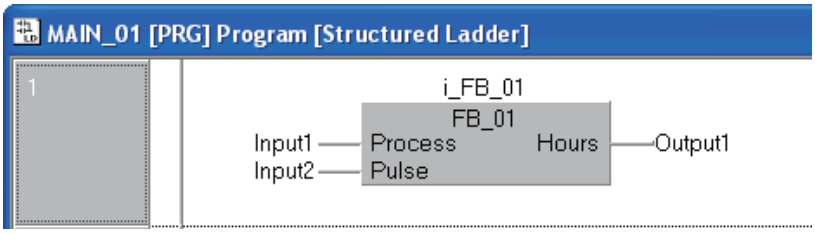
● **"Use MC/MCR"**
- This check box can be checked only if both of the "Use EN/ENO" and "Use Macrocode" check boxes are checked.
- If the MC/MCR instruction is used and the function block is not executed, the outputs and the current values of timers in the function block are reset. (If the instruction is not used, the current values are retained).

● Applying function blocks to locations where instances are used

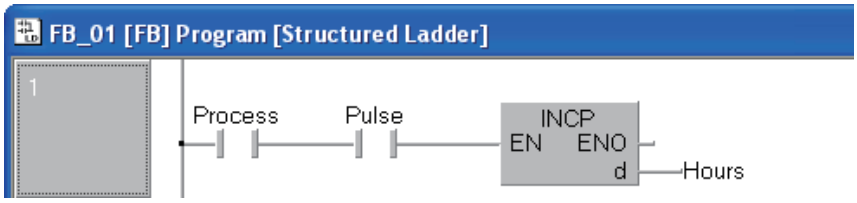
The following shows the example of codes when the following programs are compiled. Note that devices are assigned as follows.

- Input1 :X0
- Input2 :X11
- Output1 :D10

<The program at the location of use>



<Function block>



Use Macrocode	Code to be created in the program at the location of use	Function block code
Applied (The check box is selected.)	LD X0 AND X11 INCP D10	None
Not applied (The check box is not selected.)	LD X0 OUT M4096 LD X11 OUT M4097 LD SM400 CALL P2048 LD SM400 MOV D6144 D10	P2048 LD M4096 AND M4097 INCP D6144 RET

Point

● **When the "Use Macrocode" check box is not checked**

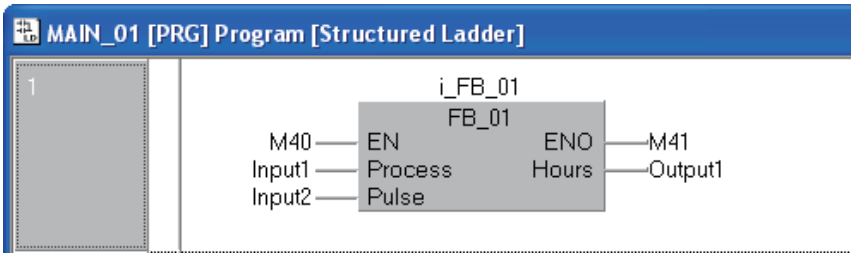
Since pointers of automatically assigned devices are used, the number of function instances that can be created is limited to the number of pointers or less specified on the System Label Setting screen. (☞ Section 5.7)
Check the "Use Macrocode" check box, if the number of set pointers or more instances are required.

● Using the MC/MCR instruction for EN control

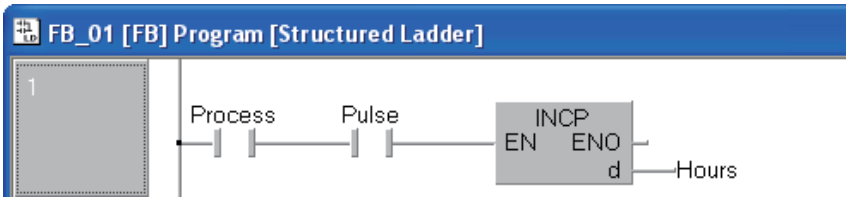
The following shows the example of codes when the following programs are compiled.
Note that devices are assigned as follows.

- Input1:X0
- Input2:X11
- Output1:D10

<The program at the location of use>



<Function block>



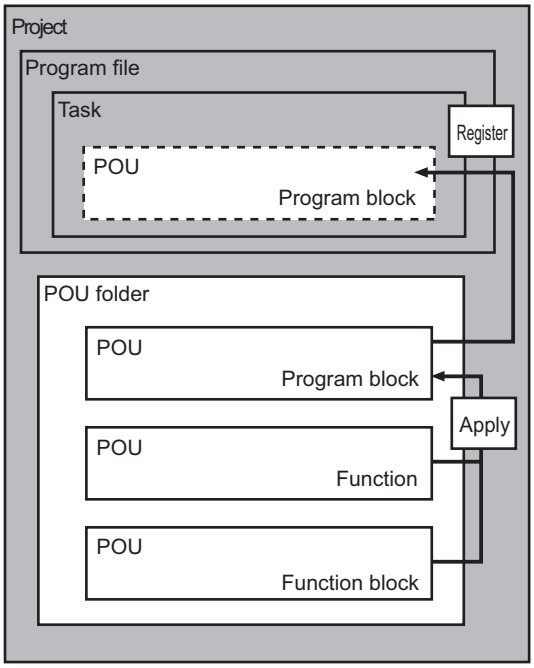
Use MC/MCR	Code to be created in the program at the location of use
MC/MCR is used (The check box is checked.)	LD M4 OUT M41 MC N0 M4096 LD X0 AND X11 INCP D10 MCR N0
MC/MCR is not used (The check box is not checked.)	LD M4 OUT M41 LDI M4 CJ P2050 LD X0 AND X11 INCP D10 P2050

4.4 Using POUs

QCPU
Q

This section explains the method for using the created POUs.

- Program blocks are used by registering them to tasks. (☞ Section 4.2.2)
- Functions and function blocks are used in program blocks. (☞ Section 4.4.1)

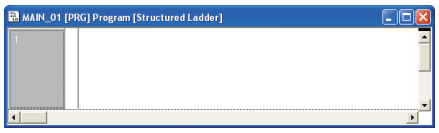


4.4.1 Using functions and function blocks

This section explains the method for selecting a function or function block from the Function Block Selection window and inserting it to a program.
(Function Block Selection window ☞ Section 6.2.1)

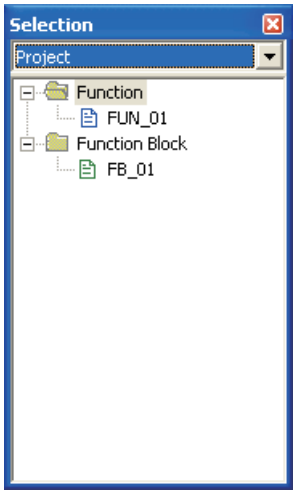
Operating procedure

1. **In the Project window, select ⇒ "POU_Pool" ⇒ "(POU)" ⇒ "Program".**
   The program editor screen is displayed.



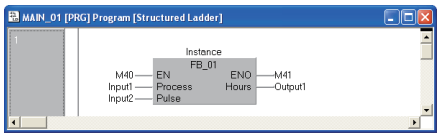
2. **Select [View] ⇒ [Docking Window] ⇒ [Function Block Selection Window].**
   The Function Block Selection window is displayed.

3. **Select "Project" from the combo box.**
   The functions and function blocks defined in the project are displayed in tree format.

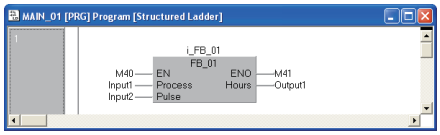


4. **Select a function or function block and then drag and drop it into the program editor.**
   The selected function or function block is inserted to the specified position in the editor.
   In the structured ladder editor, the cursor position at the time of the drop operation becomes the position of the input pin at the upper left of the function or function block.

5. **Set an argument.**
   Enter a device or label name for the argument defined in the function or function block.



6. **Set an instance.**
   For a function block, set an instance name.



■ Using function blocks created in other programming languages

This section explains the method for using function blocks created in other programming languages. The following table shows the function blocks that can be called from the program/function block.

		Function blocks to be called		
		Ladder	Structured ladder	ST
Program/function block that calls function blocks	Ladder*1	○	×	○
	Structured ladder	×	○	○
	ST	○	○	○

*1:Function blocks in the ladder language cannot call function blocks. Only programs can call function blocks.

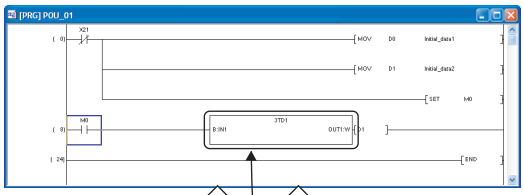
The following explains the precautions for calling function blocks from ladder to ST, or and from ST to Ladder language.

1) Check "Enable to Call Function Block from Ladder to ST, or from ST to Ladder" under [Tools] ⇒ [Options] ⇒ "Compile" ⇒ "Basic Setting". (☞ Section 11.2)

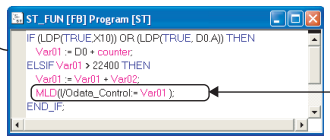
   Note that the program becomes in the uncompiled status when the option setting is changed. Compile the program again.

2) The following requirements must be satisfied when using function blocks created in the ST language in the ladder program.
   - Data type of the input/output label is any of the following types: bit, word (signed), double word (signed), single-precision real, double-precision real, string
   - The number of the input/output labels is less than 24
   - The number of characters of the input/output label is less than 16
   - The "Use EN/ENO" item is not checked in the option setting of the function blocks
   - Function block created in the structured ladder language is not used for the function block in the ST language.

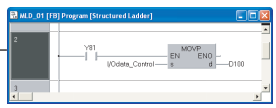
<Ladder program that calls the function block>



<Function block from structured ladder program used in ST program>



<Function block in structured ladder program>



MEMO

5 SETTING LABELS

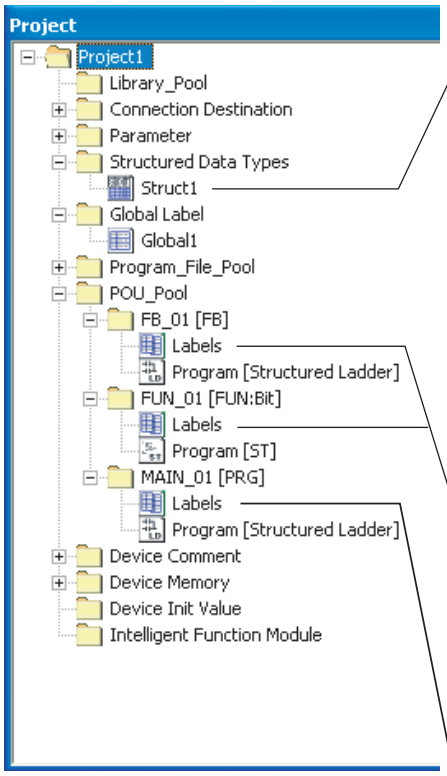
This chapter explains the method for setting labels.

5.1 Label Setting Screens 5-2
5.2 Setting Global Labels 5-3
5.3 Setting Local Labels for Program Blocks 5-5
5.4 Setting Labels for Functions and Function Blocks 5-7
5.5 Common Operations for Setting Labels 5-9
5.6 Setting Structured Data Type Labels 5-14
5.7 Setting Ranges for Devices Assigned Automatically 5-20

5.1 Label Setting Screens

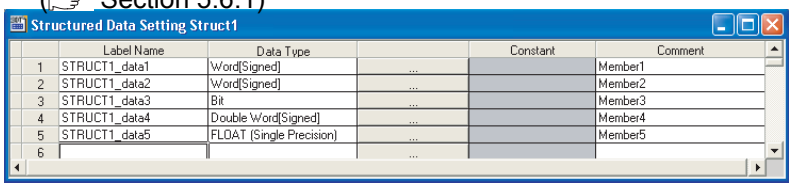
QCPU
Q

This section shows label setting screens. Labels are set on each screen below according to the type of label.



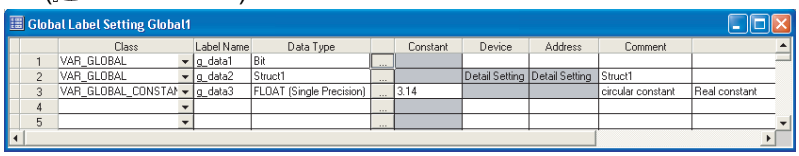
Structured Data Setting screen

Defines structured data types to be used in the label setting. (☞ Section 5.6.1)



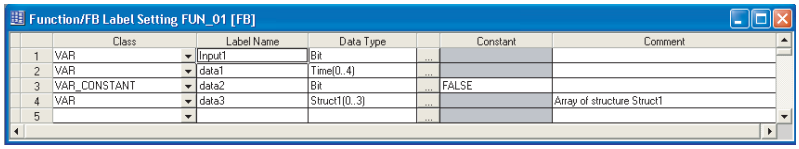
Global Label Setting screen

Defines labels that can be used for all POUs in the project. (☞ Section 5.2)



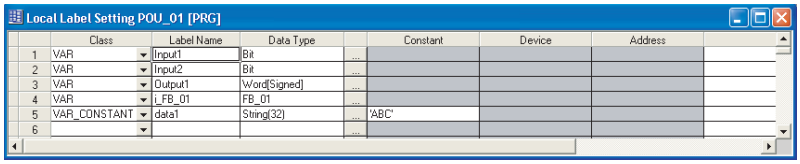
Function/FB Label Setting screen

Defines labels that can be used only for each specified POU (function/function block) in the project. (☞ Section 5.4)



Local Label Setting screen

Defines labels that can be used only for each specified POU (program block) in the project. (☞ Section 5.3)



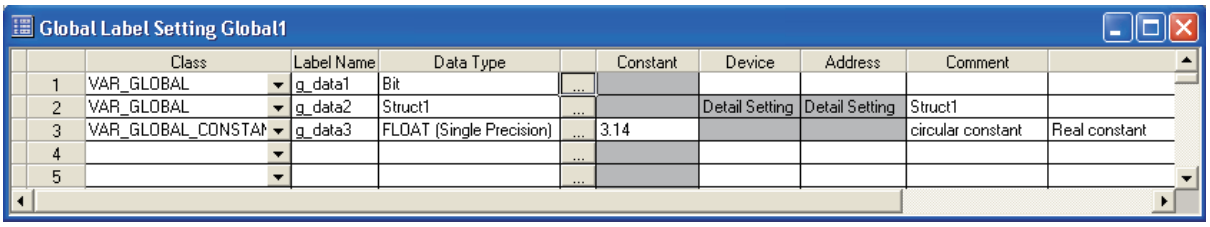
5.2 Setting Global Labels

QCPU

This section explains the method for setting global labels.

Screen display

Select Project window ⇒ "Global Label" ⇒ "(Global label)".



Operating procedure

- Set the items on the screen.

Item	Description	Number of characters
Class	Select a class name of label from the list displayed by clicking ▼. Class: VAR_GLOBAL — A common label that can be used for POUs. Class: VAR_GLOBAL_CONSTANT — A common constant that can be used for POUs.	-
Label Name	Enter a desired label name.	0 to 32 characters
Data Type	Enter the data type of label which can be set on the Type Selection screen displayed by clicking [...]. (☞ Section 5.5.1)	0 to 128 characters
Constant	Displays the constant value of the selected data type. When VAR_GLOBAL_CONSTANT is selected for "Class" and simple type is selected for "Data Type", the constant value can be set.	0 to 128 characters
Device Address	Set a device name to be assigned in the device or address format. (When a device name is entered in either column, it is displayed in the respective format automatically.) • When VAR_GLOBAL is selected for Class, the cell is left blank. • When the data type is Structure, click the "Detail Setting" cell and set devices on the Structured Data Device Setting screen. (☞ Section 5.6.3)	0 to 50 characters
Comment	Enter comments. A new line can be inserted in a cell by pressing the Ctrl + Enter keys. To display comments on the program editor, select "Device Comment" in "Comment Display Items".	0 to 1024 characters
Remark	Enter supplementary information for comments. A new line can be inserted in a cell by pressing the Ctrl + Enter keys.	0 to 1024 characters

Point

● **Characters that can be used for label names**
Any of the following label names causes an error at compilation.
- A label name that includes a space.
- A label name that begins with a numeral.
- A label name which is same as the one used for devices. (Lower case label names can be used by setting the option. ☞ Section 7.6.4)

For more details, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

● **Number of characters that can be entered for a label name**
The maximum number of characters that can be entered for a global label is 32.

● **Specifying devices/addresses**
The digit specification for bit devices (K4M0) or bit specification for word devices (D0.1) can be specified for devices/addresses.

● **Specifying timer/counter devices**
When "Data Type" is a bit type, specified devices are treated as contacts (TS, STS, CS).
When "Data Type" is a word type, specified devices are treated as current values (TN, STN, CN).

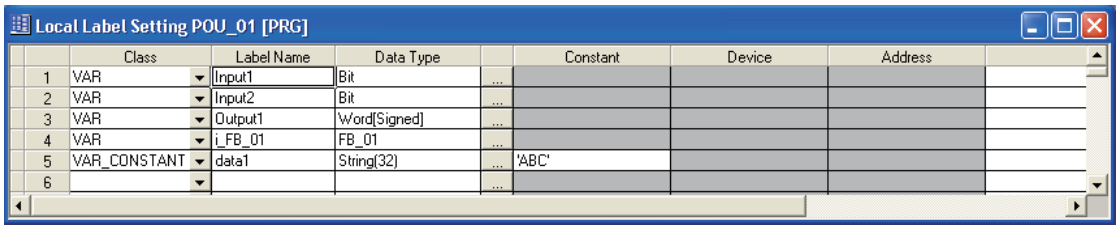
5.3 Setting Local Labels for Program Blocks

QCPU
Q

This section explains the method for setting local labels used for each program block.

Screen display

Select Project window ⇒ "POU_Pool" ⇒ "(Program block)" ⇒ "Labels".



Operating procedure

- Set the items on the screen.

Item	Description	Number of characters
Class	Select a class name of label from the list displayed by clicking ▼. **Class** / **Description** VAR: A label that can be used within the range of declared POUs. Cannot be used for other POUs. VAR_CONSTANT: A constant that can be used within the range of declared POUs. Cannot be used for other POUs. VAR_RETAIN: A latch type label that can be used within the range of declared POUs. Cannot be used for other POUs.	-
Label Name	Enter a desired label name.	0 to 32 characters
Data Type	Enter the data type of label which can be set on the Type Selection screen displayed by clicking [...]. (☞ Section 5.5.1)	0 to 128 characters
Constant	Displays the constant value of the selected data type. When VAR_CONSTANT is selected for "Class" and simple type is selected for "Data Type", the constant value can be set.	0 to 128 characters
Device Address	When the data type is Structure, click the "Detail Setting" cell and set devices on the Structured Data Device Setting screen. (☞ Section 5.6.3)	-
Comment	Enter comments. A new line can be inserted in a cell by pressing the Ctrl + Enter keys. To display comments on the program editor, select "Device Comment" in "Comment Display Items".	0 to 1024 characters

Point

● **Assigning devices automatically**
Devices are automatically assigned to labels when a program is compiled. The device assignment range can be changed on the System Label Setting screen. (☞ Section 5.7)

● **Characters that can be used for label names**
Any of the following label names causes an error at compilation.
- A label name that includes a space.
- A label name that begins with a numeral.
- A label name which is same name as the one used for devices. (Lower case label names can be used by setting the option. ☞ Section 7.6.4)

For more details, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

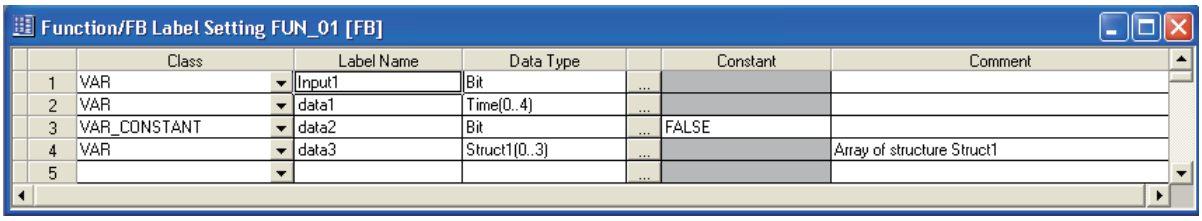
5.4 Setting Labels for Functions and Function Blocks

QCPU
Q

This section explains the method for setting local labels for each function or function block.
Create a new function or function block in advance.
For the method for creating new functions or function blocks, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

Screen display

Select Project window ⇒ "POU_Pool" ⇒ "(Function/Function block)" ⇒ "Labels".



Operating procedure

- Set the items on the screen.

Item	Description	Number of characters
Class	Select a class name of label from the list displayed by clicking ▼.	-

Class	Description
VAR	A label that can be used within the range of declared functions/function blocks. Cannot be used for other functions/function blocks.
VAR_ CONSTANT	A constant that can be used within the range of declared functions/function blocks. Cannot be used for other functions/function blocks.
VAR_INPUT	A label that receives a value. Cannot be changed in a function/function block.
VAR_OUTPUT	A label that outputs data from a function block.
VAR_IN_OUT	A label that receives a value and outputs data from a function block. Can be changed in a function block.
VAR_RETAIN	A latch type label that can be used within the range of declared POUs. Cannot be used for other POUs.

Label Name	Enter a desired label name.	0 to 32 characters
Data Type	Enter the data type of label which can be set on the Type Selection screen displayed by clicking [...]. (☞ Section 5.5.1)	0 to 128 characters
Constant	Displays the constant value of the selected data type. When VAR_CONSTANT is selected for "Class" and simple type is selected for "Data Type", the constant value can be set.	0 to 128 characters
Comment	Enter comments. A new line can be inserted in a cell by pressing the Ctrl + Enter keys. To display comments on the program editor, select "Device Comment" in "Comment Display Items".	0 to 1024 characters

Point

● **Maximum number of labels**
- In the ladder language, 1 to 24 labels can be set for inputs (input variables and I/O variables), and 1 to 24 labels can be set for outputs (output variables and I/O variables). The I/O variable is counted as '1 input + 1 output'.
- In languages other than the ladder language, the maximum number of labels that can be set for a single POU is 5120.

● **Assigning devices automatically**
Devices are automatically assigned to labels when a program is compiled. The device assignment range can be changed on the System Label Setting screen. (☞ Section 5.7)

● **Characters that can be used for label names**
Any of the following label names causes an error at compilation.
- A label name that includes a space.
- A label name that begins with a numeral.
- A label name which is same name as the one used for devices. (Lower case label names can be used by setting the option. ☞ Section 7.6.4)

For more details, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

5.5 Common Operations for Setting Labels

QCPU

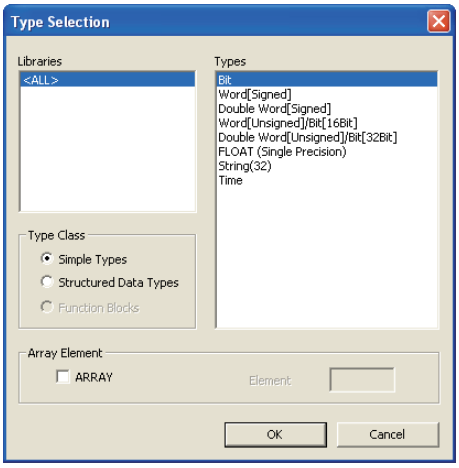
This section explains the common operations for each label setting screen.

5.5.1 Selecting data types

To define a label, a data type must be specified by directly entering the type in text or selecting it on the Type Selection screen.

Screen display

Click [...] in the data type entry field on each label setting screen.



Operating procedure

1. **Select a type in the "Type Class".**

Item	Description
Simple Types	Specify this to select a data type from basic types such as bit and word.
Structured Data Types	Specify this to select a data type from the defined structures.
Function Blocks	Specify this to select a data type from the defined function blocks. (Not displayed for the label settings of functions)

2. **In the "Libraries" field, select the reference source such as the defined structure that is used as the data type.**

Item	Description
<ALL>	Browses data types and structures/function blocks defined in the project, and all libraries.
<Project>	Browses structures/function blocks defined in the project.
(Library name)	Browses structures/function blocks defined in the specified library.

3. **In the "Types" field, select the data type, structure or function block name.**

4. **Click the [OK] button when the setting is completed.**

   The settings are displayed in the "Data Type" column on the label setting screen.

Point

● **Data types used only in a specific programming language**
Data types that can be used for local labels differ in each programming language; ladder, SFC, ST, and structured ladder.
The following shows the data types that can be used for each programming language.

Programming language	Data type
ST/structured ladder	• Word (unsigned)/16-bit string • Double word (unsigned)/32-bit string • Time
Ladder/SFC	• Timer • Counter • Retentive timer • Pointer

● **Method for displaying the Type Selection screen**
The Type Selection screen can be opened by any of the following operations when [...] is in the selected status.

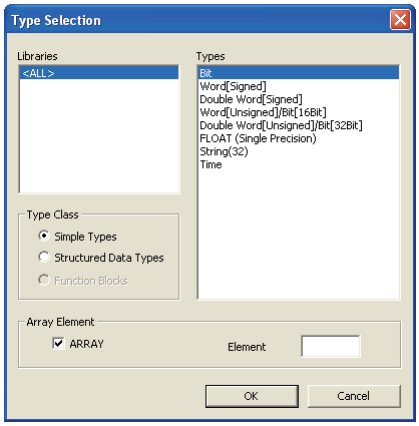
- Press the [Enter] key.
- Press the [Space] key.
- Press the [F2] key.

■ Setting arrays for data type

The following explains the method for defining a data type as an array.
To define a data type as an array, set the items in the "Array Element" field on the Type Selection screen.

Operation

1. Click [...] in the data type entry field on each label setting screen.
2. Check the check box in the "Array Element" field.
3. Enter the number of elements in the "Element" field.
4. Set the data type of the array element in the same manner as setting the normal data type.



● To change the offset

To change the offset ([Array start value]..[Array end value]) to a value other than 0, edit the offset value by directly entering the array type declaration in text on each label setting screen.

	Class	Label Name	Data Type
1	VAR	data1	Bit(0..2)
2	VAR	data2	Word[Unsigned]/Bit[16Bit](1..2)
3	VAR	data3	Struct1(0..4)

● To change the array to a two- or three-dimensional array

Edit two- or three-dimensional array by directly entering the array type declaration in text on each label setting screen.

For details on array type declarations, refer to the following manual.

☞ QCPU Structured Programming Manual (Fundamentals)

	Class	Label Name	Data Type
1	VAR	data1	Bit(0..2,0..4)
2	VAR	data2	Word[Unsigned]/Bit[16Bit](0..4,0..0,1..2)
3	VAR	data3	Struct1(1..5,0..1)

Point

● **When a constant type is set for Class**
For labels whose classes are set as VAR_CONSTANT or VAR_GLOBAL_CONSTANT, arrays cannot be set for the data type. If they are set, an error occurs at compilation.

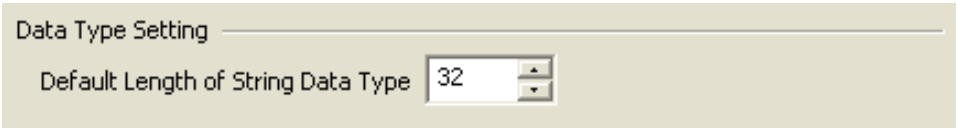
● **Offset values**
Minus values can be set for offset values.

■ Setting data length of the character string data type

The following explains the method for setting the data length of the character string data type.
To change the data length of the character string data type, directly edit the data length on each label setting screen.
The data length of the character string data type is set in the option setting.

Operation

• Select [Tools] ⇒ [Options] ⇒ "Label Setting Editor" ⇒ "Default Length of String Data Type", and set the data length.



5.5.2 Editing lines

This section explains the method for editing lines on the label setting screen.

■ Adding lines

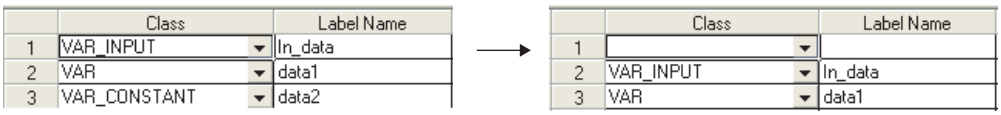
The following explains the method for adding a line on the label setting screen.

● New Declaration (Before)

This function inserts a line above the selected line.

Operation

• Select [Edit] ⇒ [New Declaration (Before)].

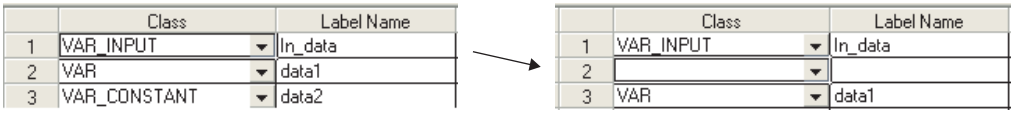


● New Declaration (After)

This function inserts a line below the selected line.

Operation

• Select [Edit] ⇒ [New Declaration (After)].

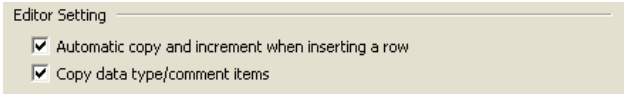


Point

● **Adding lines**

When using the New Declaration (After) function, label names, data types or other attributes can be set automatically to the added lines.

Select [Tools] ⇒ [Option] ⇒ "Label Setting Editor".



■ Deleting lines

The following explains the method for deleting a line on the label setting screen.

Operation

• Select [Edit] ⇒ [Delete Line].

	Class	Label Name
1	VAR_INPUT	In_data
2	VAR	data1
3	VAR_CONSTANT	data2

→

	Class	Label Name
1	VAR	data1
2	VAR_CONSTANT	data2
3	VAR	data3

■ Displaying comments and remarks of the first line only or all lines

For the "Comment" and "Remark" columns, data can be entered in multiple lines. The expand declaration and collapse declaration functions are provided to select whether to display all lines or only the first line by double clicking " + " or " - ".

● Expand declaration

If a cell is expanded, all lines of information are displayed in the "Comment" and "Remark" columns.

Operation

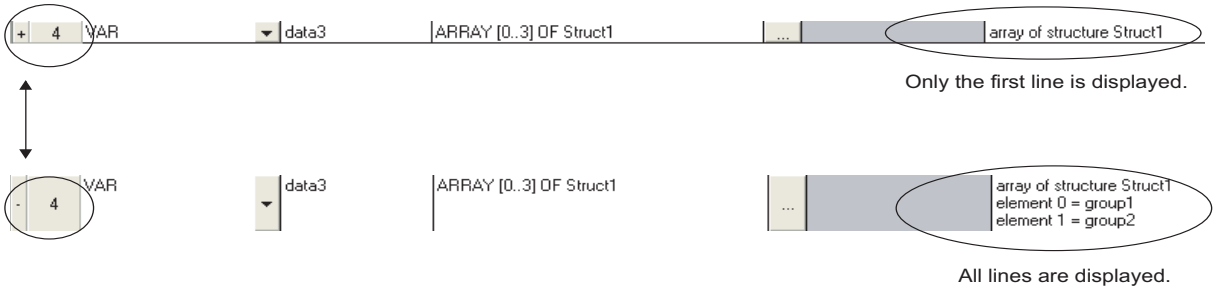
• Double click " + " in the line number cell of the table.

● Collapse declaration

If a cell is collapsed, only the first line of information is displayed in the "Comment" and "Remark" columns.

Operation

• Double click " - " in the line number cell of the table.

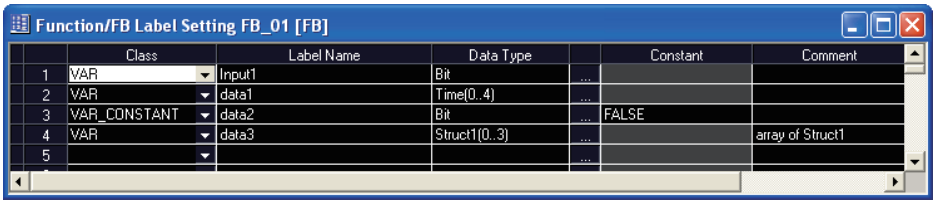


■ Selecting all lines

All lines can be selected by the following operation.

Operation

• Select [Edit] ⇒ [Select All].



5.6 Setting Structured Data Type Labels

QCPU
Q

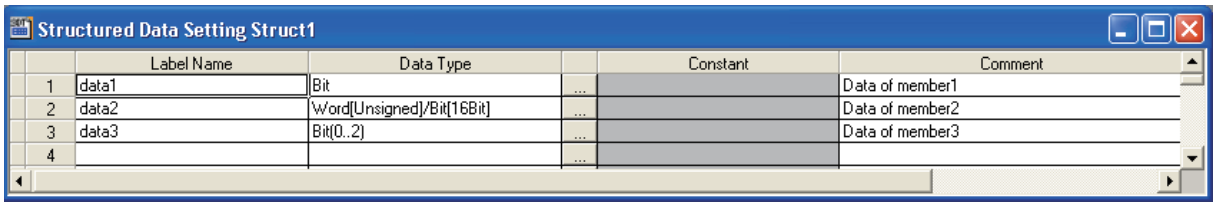
This section explains the method for setting structured data type labels.

5.6.1 Setting data type of structure

Set the elements of the structure on the Structured Data Setting screen.

Screen display

Select Project window ⇒ "Structured Data Types" ⇒ "(Structure)".



Operating procedure

- Set the items on the screen.

Item	Description	Number of characters
Label Name	Enter a desired label name.	0 to 32 characters
Data Type	Enter the data type of label which can be set on the Type Selection screen displayed by clicking [...]. (☞ Section 5.5.1)	0 to 128 characters
Constant	Displays the constant value of the selected type.	0 to 128 characters
Comment	Enter comments. A new line can be inserted in a cell by pressing the Ctrl + Enter keys. To display comments on the program editor, select "Device Comment" in "Comment Display Item".	0 to 1024 characters

Point

● **Creating new structured data**
- Create structured data using the Project window.
  (☞ GX Works2 Version1 Operation Manual (Common))

5.6.2 Setting data type as structure

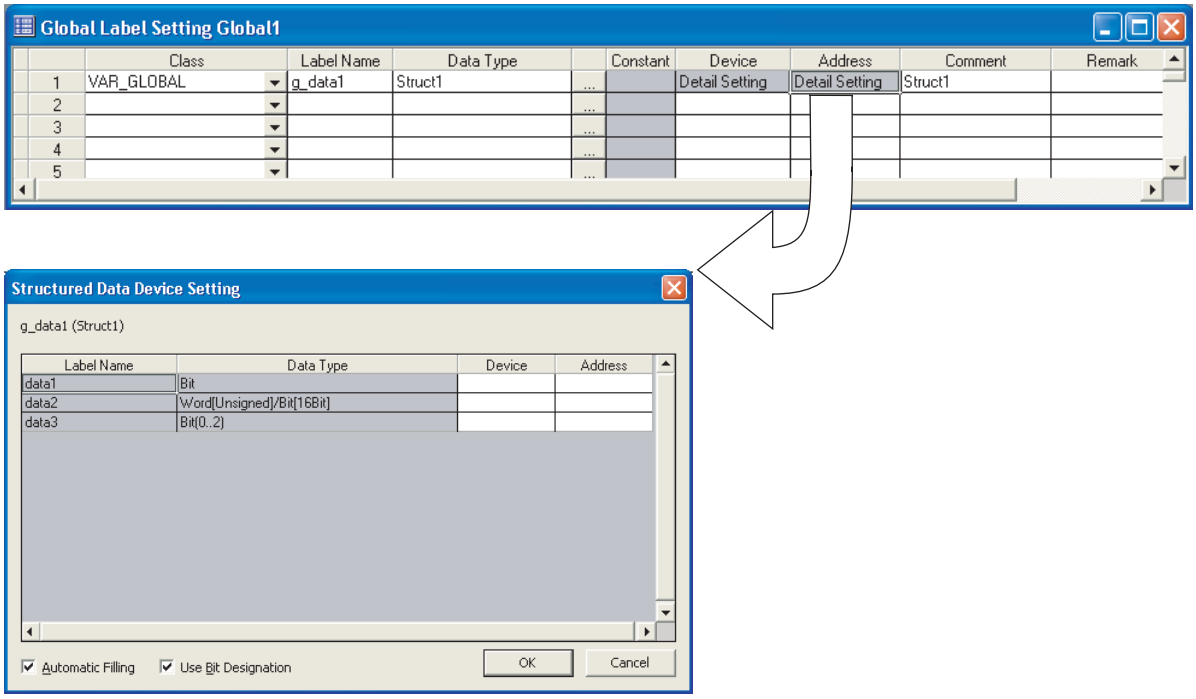
To define the data type of label as a structure, set a structure in the data type entry field on each label setting screen. In addition to direct text entry, a structure can be selected on the Type Selection screen (☞ Section 5.5.1).

5.6.3 Assigning devices to structured data type labels

Devices of the structured data type global labels are set on the Structured Data Device Setting screen. If the structure is set for data type on the Global Label Setting screen, "Detail Setting" is displayed in the "Device" and "Address" columns. Click "Detail Setting" to display the Structured Data Device Setting screen.

Screen display

Click "Detail Setting" on the label setting screen.



Operating procedure

- Set the items on the screen.

Item		Description
Label Name		Displays label names defined as structure.
Data Type		Displays data types set to data names.
Device name	Device	Set device names to be assigned in the "Device" or "Address" column. (When a device name is entered in either column, it is displayed in the respective format automatically.)
	Address	
Automatic Filling		Check this to set devices automatically to cells in which devices are not set with the same data type.
Use Bit Designation		Check this to enter bit devices automatically by using bit specification for word devices.

Point

● **Display on the Local Label Setting screen**
- For the Structured Data Device Setting screen, the read-only screen is displayed.

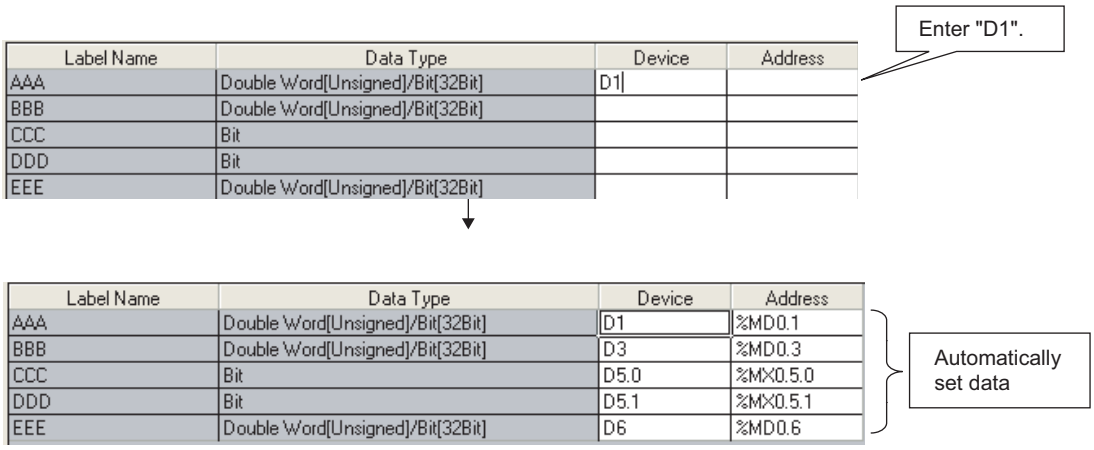
■ Entering device names automatically

The Automatic filling function can be used when setting series of devices for the same data type data on the Structured Data Device Setting screen.
When using the Automatic filling function, whether to use bit specification for word devices can be selected.

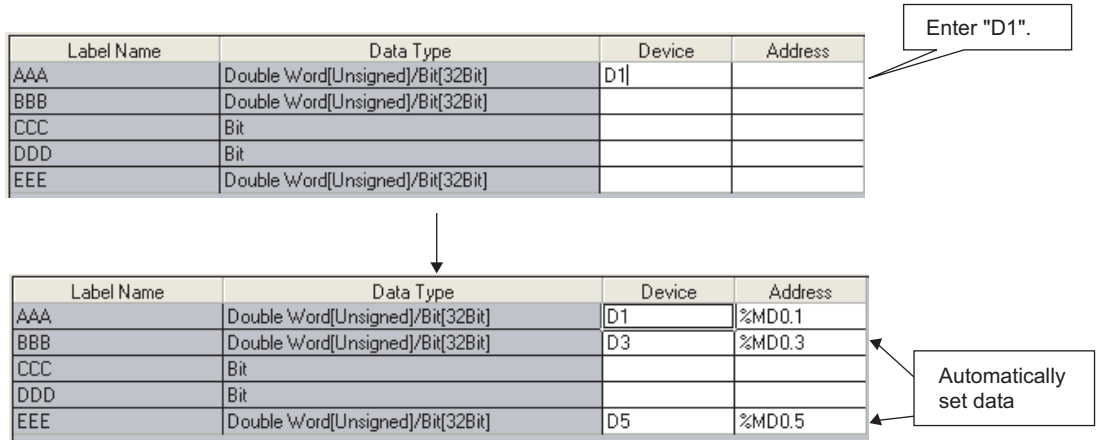
Operation

1. Click "Detail Setting" on the label setting screen.
   The Structured Data Device Setting screen is displayed.
2. Enter the device name to the Device/Address column.
   Series of devices are automatically set in the cells following the one entered, skipping the cells where devices are already set.

● When using bit specification



● When not using bit specification



Point

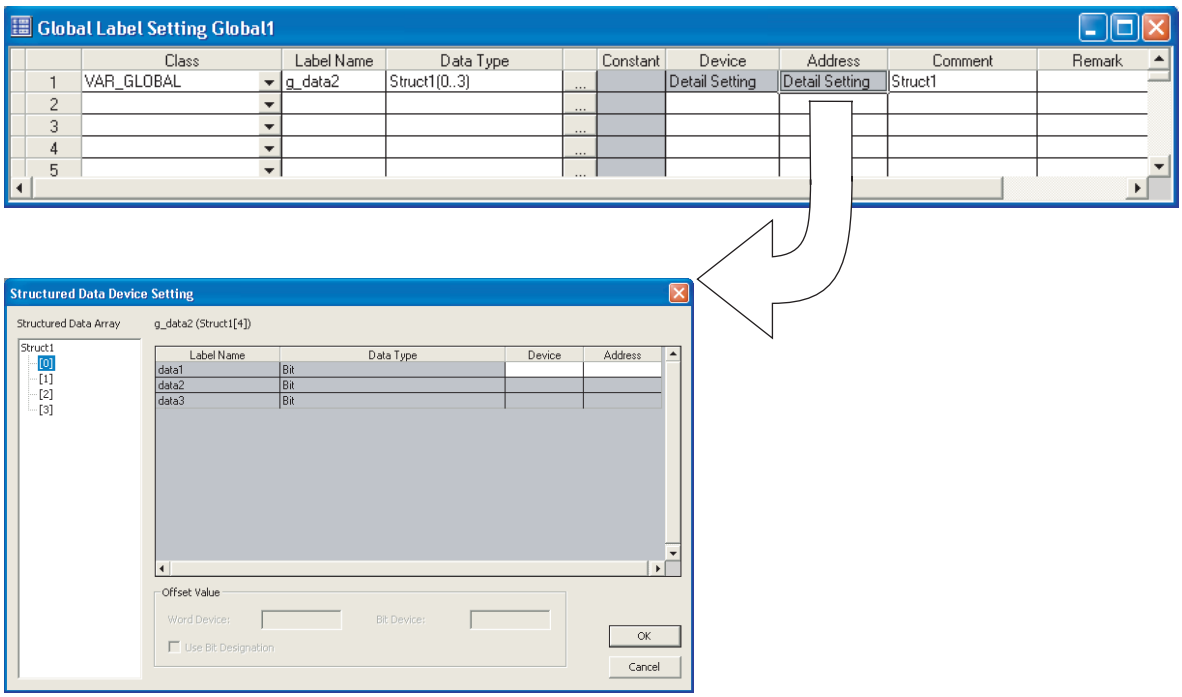
● **When not using bit specification**
- When the bit specification is not used, uncheck the "Use Bit Designation" item on the Structured Data Device Setting screen.

5.6.4 Assigning devices to structured array type labels

This section explains the method for setting the devices for the structured array type global labels on the Structured Data Device Setting screen.
When VAR_GLOBAL is set for "Class" and the structured array is set for "Data Type" on the Global Label Setting screen, "Detail Setting" is displayed in the "Device" and "Address" columns. Click "Detail Setting" to display the Structured Data Device setting screen.

Screen display

Click "Detail Setting" on the label setting screen.



Operating procedure

- Set the items on the screen.

Item		Description
Structured Data Array		Displays elements of the structured array in tree format. The device setting of the element selected in the tree is displayed in the right area of the screen.
Label Name		Displays label names defined as the structure.
Data Type		Displays data types set to data names.
Device name	Device	Set device names to be assigned in the "Device" or "Address" column. (When a device name is entered in either column, it is displayed in the respective format automatically.) Device names can be entered for the start array element only. For the subsequent array elements, device names offset from the device number set for the start element are automatically set.
	Address	
Offset Value	Word Device/Bit Device	Specify the device offset value to be set to the same data type cells in the array element.
	Use Bit Designation	Check this to enter bit devices automatically by using bit specification for word devices.

Point

● **Display on the Local Label Setting screen**
- For the Structured Data Device Setting screen, the read-only screen is displayed.

■ Entering device names automatically by setting offset values

On the Structured Data Device setting screen, device numbers to be entered automatically can be set in fixed spacing for each array element of the structured array. Specify the difference in device numbers between the array elements as an offset value for the start device of the same data type.

Operation

1. Click "Detail Setting" on the label setting screen.
   The Structured Data Device Setting screen is displayed.
2. Set the "Offset Value".

3. Set the device name for the start element data of the array.
   Devices are set to the same data type of the array elements in the screen.

・Offset value is not specified.

< Array [0] >

Label Name	Data Type	Device	Address
AAA	Double Word[Unsigned]/Bit[32Bit]	D1	%MD0.1
BBB	Double Word[Unsigned]/Bit[32Bit]	D3	%MD0.3
CCC	Bit		
DDD	Double Word[Unsigned]/Bit[32Bit]	D5	%MD0.5

Enter "D1".

< Array [1] >

Label Name	Data Type	Device	Address
AAA	Double Word[Unsigned]/Bit[32Bit]	D7	%MD0.7
BBB	Double Word[Unsigned]/Bit[32Bit]	D9	%MD0.9
CCC	Bit		
DDD	Double Word[Unsigned]/Bit[32Bit]	D11	%MD0.11

Devices starting from "D7" are assigned.

・Offset value is 10.

< Array [0] >

Label Name	Data Type	Device	Address
AAA	Double Word[Unsigned]/Bit[32Bit]	D1	%MD0.1
BBB	Double Word[Unsigned]/Bit[32Bit]	D3	%MD0.3
CCC	Bit		
DDD	Double Word[Unsigned]/Bit[32Bit]	D5	%MD0.5

Enter "D1".

< Array [1] >

Label Name	Data Type	Device	Address
AAA	Double Word[Unsigned]/Bit[32Bit]	D11	%MD0.11
BBB	Double Word[Unsigned]/Bit[32Bit]	D13	%MD0.13
CCC	Bit		
DDD	Double Word[Unsigned]/Bit[32Bit]	D15	%MD0.15

Devices starting from "D11" are assigned.

● Using bit specification

Devices can be set using bit specification for word devices when entering devices automatically.

Operation

1. Click "Detail Setting" on the label setting screen.
   The Structured Data Device Setting screen is displayed.
2. Check the "Use Bit Designation" item.
3. Set the device name for the start element data of the array.
   Devices are set to the same data type or bit device data type.

Point

- **Data for which device name can be entered**
  For the structured array type, a device name can be entered only for the start element of array.
  For the subsequent array elements, device names offset from the device number set for the start element are automatically set.
- **Offset values**
  - An expression such as '1\2' can be specified for an incremental value of device such as U0\G0.
  - When 0 is specified for the incremental value, the device number same as the data set to the start of devices is set to all data in the array.

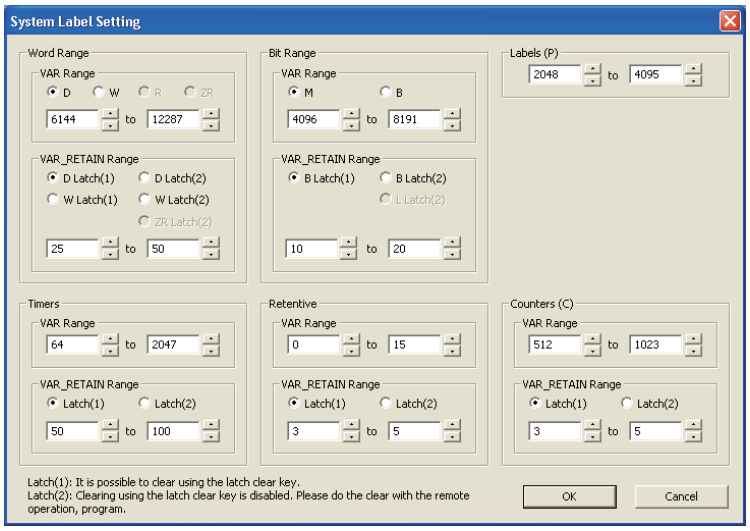
5.7 Setting Ranges for Devices Assigned Automatically

QCPU
Q

This section explains the method for setting the ranges for devices that are automatically assigned to local labels.

Screen display

Select [Tools] ⇒ [System Label Setting].



Operating procedure

- Select the device type, and set the start and end addresses to be assigned.

Point

● **Device types**
The types of automatically assigned devices differ by the type of CPU.

● **Setting ranges**
- The setting ranges depend on the number of device points set in PLC parameter.
For details of PLC parameter, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

● **Assigning devices automatically**
Word devices, bit devices, and timer/counter devices are automatically assigned in the order from larger device number within the device range set on the System Label Setting screen.

6 EDITING PROGRAMS

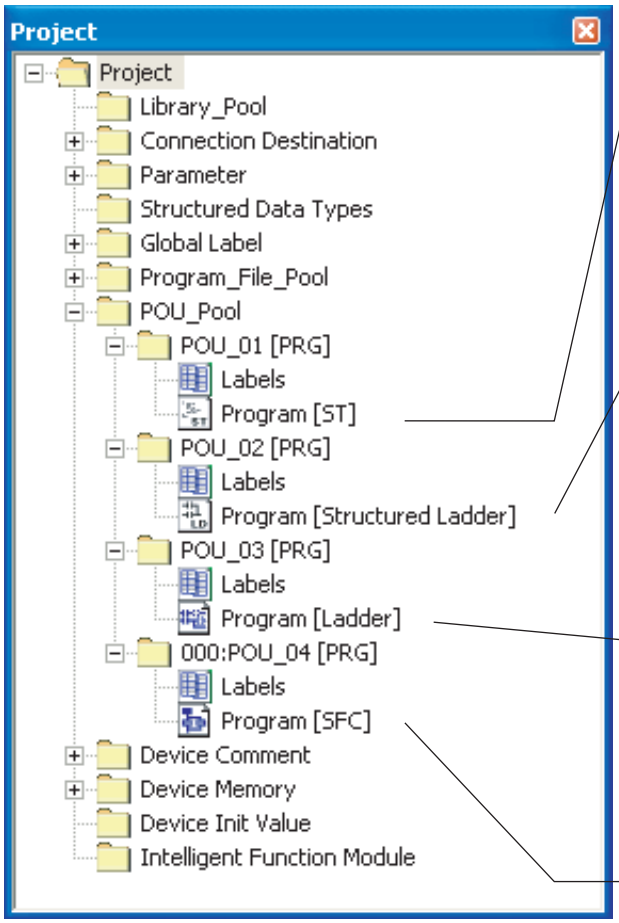
This chapter explains the functions of the program editors used for editing sequence programs.

6.1 Types of Program Editor 6-2
6.2 Common Operations of Program Editors 6-5
6.3 Editing ST Programs 6-22
6.4 Editing Structured Ladder Programs 6-25

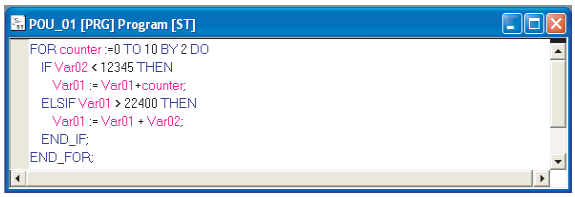
6.1 Types of Program Editor

QCPU
Q

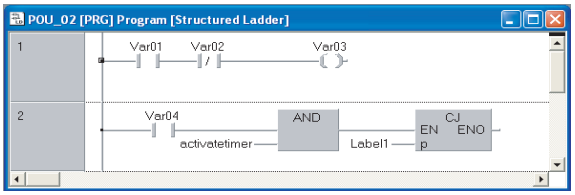
This section explains the method for editing programs according to the programming language.



ST editor (☞ Section 6.3)



Structured ladder editor (☞ Section 6.4)



Ladder editor

For the operating method of the ladder editor, refer to the following manual.

☞ GX Works2 Version1 Operating Manual (Simple Project)

SFC editor

For the operating method of the SFC editor, refer to the following manual.

☞ GX Works2 Versio1 Operating Manual (Simple Project)

Point

● **Creating new program data**
Program data are created in the project window (☞ GX Works2 Version1 Operating Manual (Common)).
Program data are created paired with a label editor when a new POU is created.

6.1.1 Available programming languages

The following table shows the programming languages that can be used for each POU.

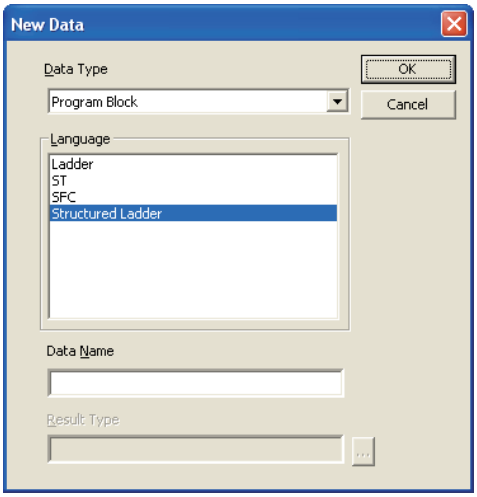
Table 6.1.1-1 Programming languages that can be used for each POU

POU	Programming language			
	ST	Structured ladder	Ladder	SFC
Functions	○	○	×	×
Function blocks	○	○	○	×
Program blocks	○	○	○	○

○: Available ×: Not available

A programming language is specified when creating a new POU.
For the method for creating new data, refer to the following manual.

☞ GX Works2 Version1 Operating Manual (Common)



Point

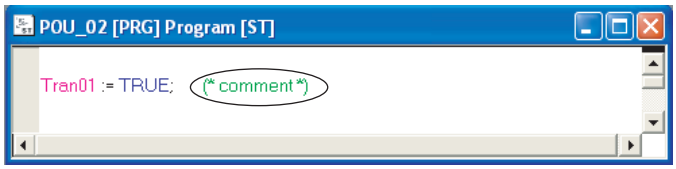
● **Changing programming languages**
The programming language that is selected when creating a program, cannot be changed later.
To change to another programming language, create a new POU.

6.1.2 Available comment formats

The following explains the comments that can be used for each program editor.

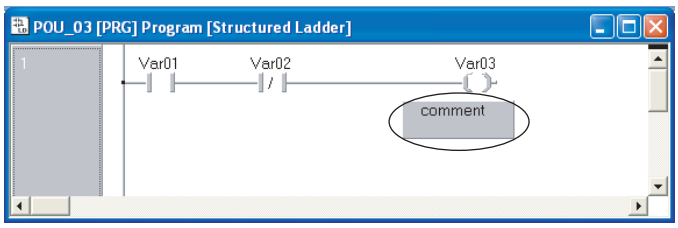
■ Comments in the ST editor

In the ST editor, comments are enclosed by '(*' and '*)'.

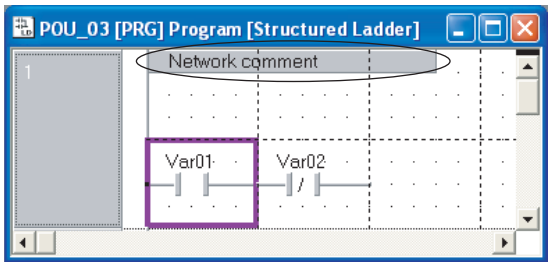


■ Comments in the structured ladder editor

In the structured ladder editor, comments can be pasted anywhere in the same manner as network elements.
(☞ Section 6.4.11)

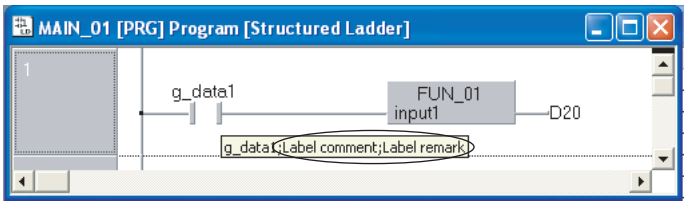


In the Guided mode, comments can be added at the start of a network.
(☞ Section 6.4.12)



■ Label comments

Label comments and remarks set at label setting can be displayed on tool hints.
(☞ Section 6.2.10).



6.2 Common Operations of Program Editors

QCPU
Q

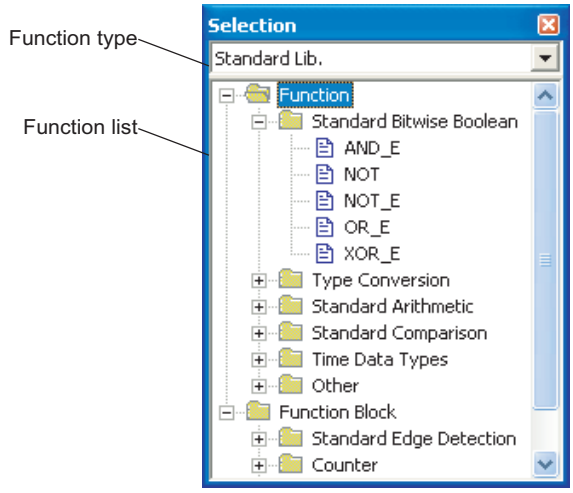
This section explains the common operations of program editors.

6.2.1 Using POUs in a program (Function Block Selection window)

This section explains the method for selecting functions or function blocks in the Function Block Selection window.
(The Function Block Selection window is one of the docking windows.)

Screen display

Select [View] ⇒ [Docking Window] ⇒ [Function Block Selection Window].



Operating procedure

1. **Set the items on the screen and select a function.**

Item		Description
Function type	All Parts	Displays all functions (including function blocks and operators).
	Sequence Instruction, Basic Instruction, Application Instruction, PID Control Instruction, Module Dedicated Instruction	Displays instructions as functions in alphabetic order.
	Standard Lib.	Displays classified functions and function blocks of the application function such as 'Type conversion' and 'Arithmetic operation'.
	(User library name)	Displays functions of the selected user library.
	Project	Displays functions and function blocks defined in the project.
	Operator	Displays operators of the selected language.
Function list		Displays selected functions alphabetically in tree format. (The displayed functions vary according to the active work window.)

2. **Enter the function in the program editing screen.**

Double click the function name, and the function is entered at the cursor position in the editor.

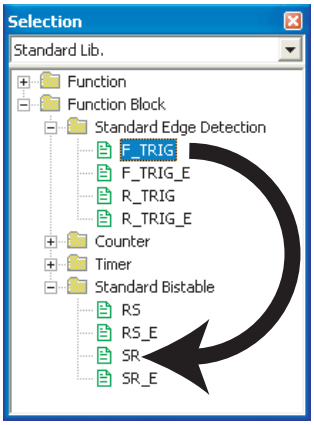
■ Searching for a function name by its initial letter

The following explains the method for searching for a particular item among those displayed in the function list tree by its initial letter.

Operating procedure

- In the function list field, enter the initial letter of the function to be searched for.

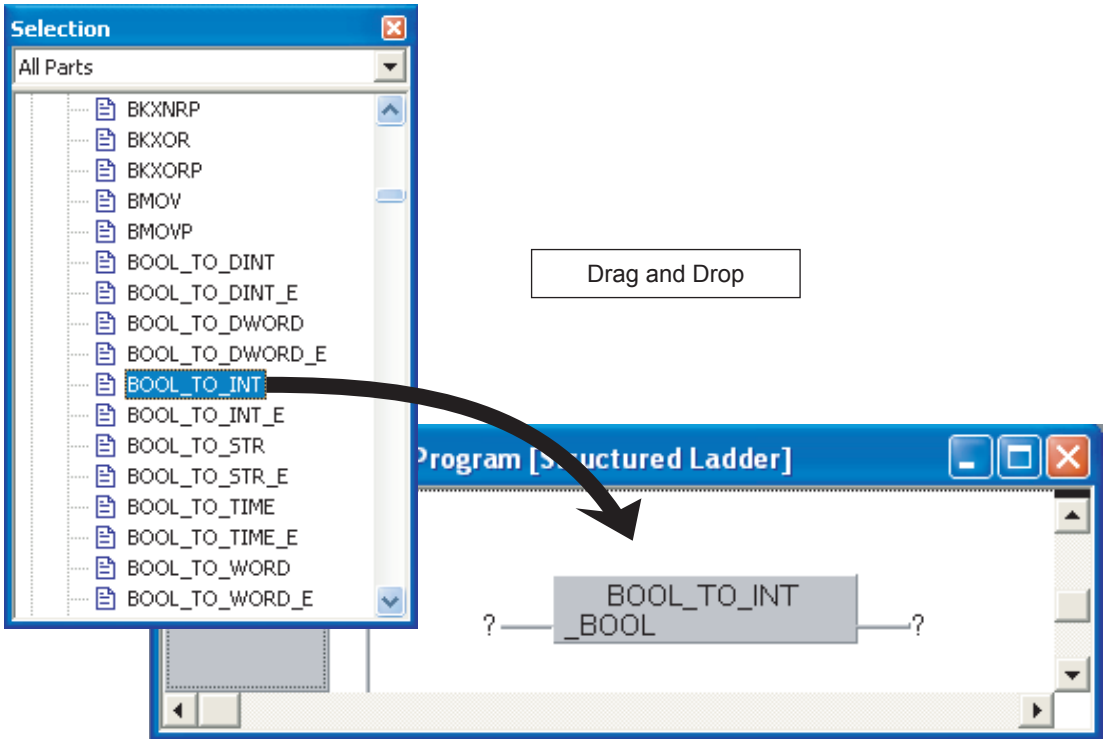
  If there is no function with the entered initial letter, the selection status does not change.



In the case shown on the left, entering [S] key changes the selection status from 'F_TRIG' to 'SR'.

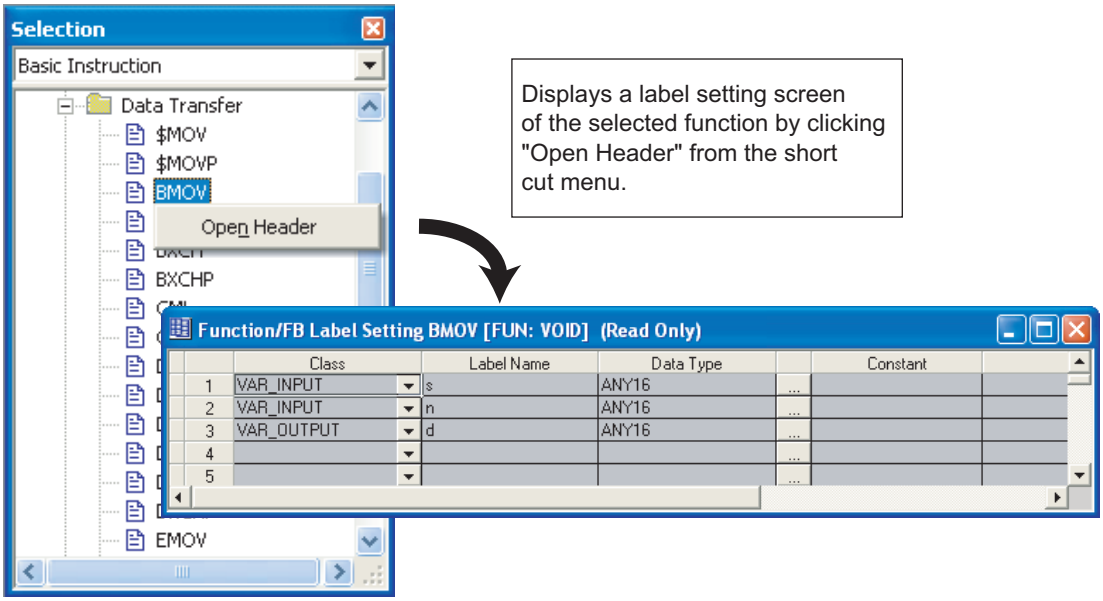
■ Pasting a function by the drag-and-drop operation

A function can be dragged and dropped at the specified position in the program editor where the function is used.



■ Displaying label settings

A label setting screen can be displayed by right clicking a function or function block.



Point

● **Operators**
Since label settings are not available for operators, short cut menu is not displayed.

6.2.2 Using templates

This section explains the method for using different templates according to the instructions/functions to edit sequence programs in the ST editor.
A template is a basic format such as an argument type, defined for each instruction/function.
For details of data types displayed in the templates, refer to the following manual.
☞ QCPU Structured Programming Manual (Fundamentals)

The following shows the method for using a template.

Operating procedure

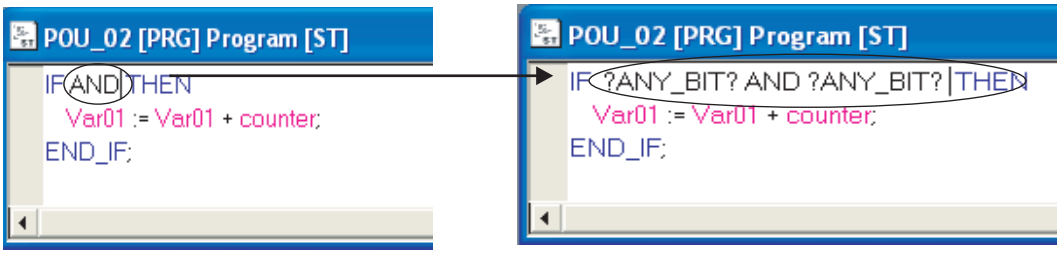
1. **Move the cursor to the instruction/function.**

2. **Select [Edit] ⇒ [Create Template].**

3. **Enter variables according to the displayed template.**

   Delete the data type name enclosed by "?", and enter a label name or device that is equivalent to the data type.

<ST editor>

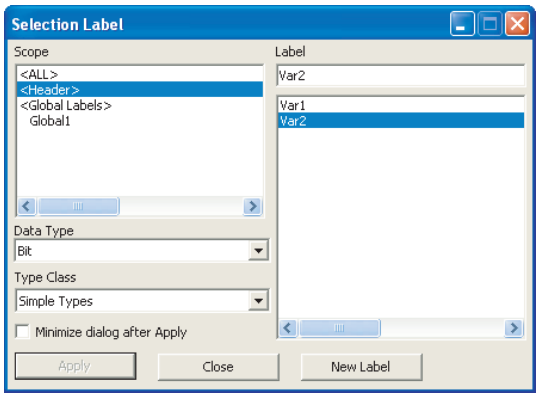


6.2.3 Using labels in the program

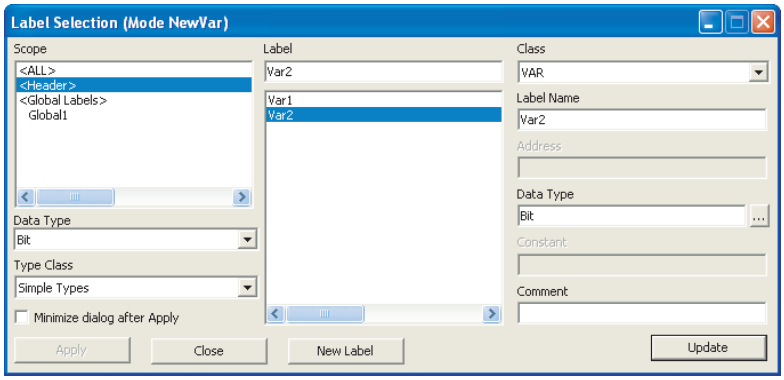
This section explains the method for entering preset labels selected on the Label Selection screen.
The settings of a new label can be added on the Label Selection screen while creating a program.

Screen display

Select [Edit] ⇒ [List Operands] ( ).



Select [Edit] ⇒ [New Label].



Operating procedure

- Set the items on the screen.

Item		Description
Scope	<ALL>	Displays all labels that can be used in the project.
	<Header>	Displays labels set in the local label settings.
	<Global Labels>	Displays labels set in the global label settings.
	(User library name)	Displays labels set in each user library.
Type Class	Simple Types	Specify this to select a data type from all types except for arrays, structures, and function blocks.
	Structure	Specify this to select a data type from existing structures.
	Function block	Specify this to select a data type from existing function blocks.
Data Type	Select a data type that belongs to the class selected in the "Type Class" drop-down list.	
Label	Displays the list of labels selected in the "Scope" and "Data Type" drop-down list. The selected label appears in the top field. Click the Apply button to insert the labels to the program editor.	
Minimize dialog after Apply	Check this to minimize the screen after the Apply button is clicked.	

Screen button

The following explains the buttons on the screen.

● Apply

Enters the selected label name in the program editor.

● Close

Disables the settings and closes the screen.

● New Label

Switches display/hide of the label setting items.
The following table shows the label setting items.

Item	Description
Class	Click and select a label class name from the list.
Label Name	Enter a desired label name.
Address	Set a device name.
Type	Enter a data type of the label. A data type can be set on the Type Selection screen displayed by clicking ... .
Constant	Displays an constant value of the selected data type. If a constant type is set for "Class" and simple type is selected for "Data Type", an constant value can be set.
Comment	Enter comments of the label.

● Define, Update

These buttons are used to create or modify labels. The settings are added to the label setting.
Click the Define button to create a new label and click the Update button to modify an existing label.
These buttons become applicable when the set contents are correct.

6.2.4 Editing networks

This section explains the method for creating sequence programs in units of networks using the following program editor.

Table 6.2.4-1 Program editor for editing in units of networks

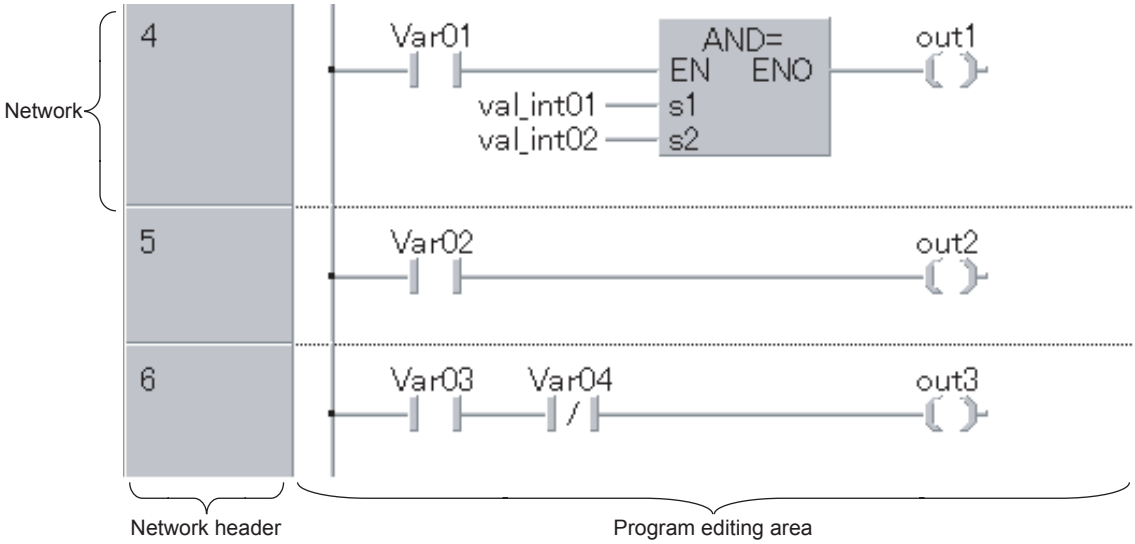
Program editor	With or without networks
ST	×
Structured ladder	○

○: With networks ×: Without networks

Networks are displayed as follows.

Screen display

Select Project window ⇒ "POU_Pool" ⇒ "(POU)" ⇒ "Program".



Display contents

The following table shows the components of a network.

Name	Description
Network	A unit for creating a program
Network header	Displays information such as a title set for each network, and a label name of the jump target.
Program editing area	An area for creating a program.

■ Inserting networks

A new network can be added or inserted by one of the following operations.

● Inserting a network at the start

Operation

• Select [Edit] ⇒ [New Network] ⇒ [Top].
A new network is inserted at the start of all networks.

● Inserting a network in front of the current network

Operation

• Select [Edit] ⇒ [New Network] ⇒ [Before] ( ).
A new network is inserted in front of the network being edited.

● Inserting a network in back of the current network

Operation

• Select [Edit] ⇒ [New Network] ⇒ [After] ( ).
A new network is inserted in back of the network being edited.

● Inserting a network at the end

Operation

• Select [Edit] ⇒ [New Network] ⇒ [Bottom].
A new network is inserted at the end of all networks.

■ Deleting networks

The following explains the method for deleting a selected network.

Operating procedure

• Select [Edit] ⇒ [Delete].

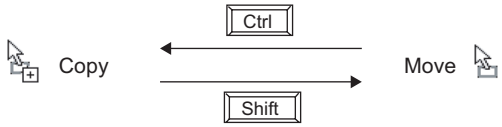
■ Copying and moving networks

The following explains the method for copying/moving networks within the editor or to another program editor which uses the same programming language on the screen.

● Copying/moving networks by the drag-and-drop operation

Networks can be copied/moved by clicking the network header and the drag-and-drop operation.

- To cancel the copy/move operation, press the Esc key during the drag-and-drop operation.
- To switch the copy/move operations, use the Ctrl / Shift key as follows:
  (Different icons are displayed for the copy and move operations.)



● Copying/moving networks using the clipboard

Networks can be copied by the general menu options or shortcut keys using the clipboard.

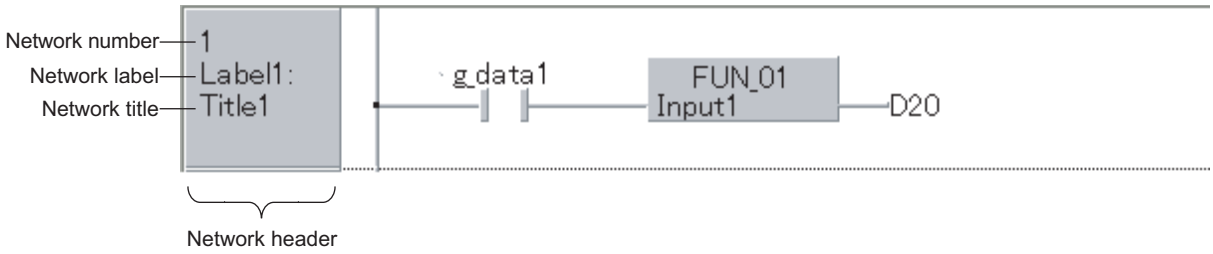
Operation

1. Select a network by clicking the network header.
2. Select [Edit] ⇒ [Cut] ( )/[Copy] ( ).
3. Select [Edit] ⇒ [Paste] ( ).

■ Editing network headers

A network header is an area that displays information set for each network.
In a network header, a network label and network title can be entered.

Screen display



Display contents

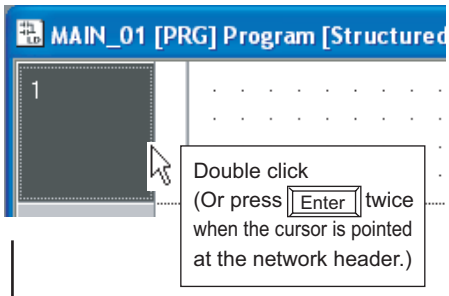
The following table shows the display contents of a network.

Name	Description
Network number	A number automatically assigned from the start.
Network label	Enter a character string to be specified as the jump target when using the jump instruction. A colon (:) needs to be entered at the end of the character string. The maximum number of characters that can be entered is 7.
Network title	Enter a meaning of the program as a title when necessary. The maximum number of characters that can be entered is 20.

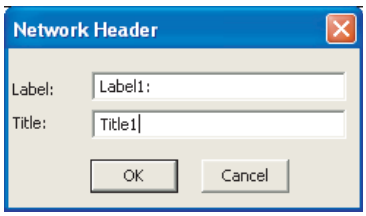
Operating procedure

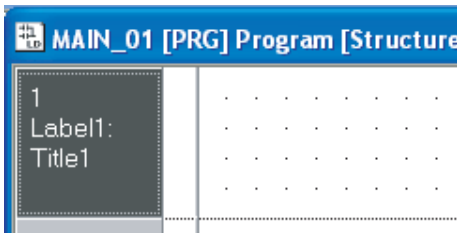
1. **Double click the network header.**

The Network Header screen is displayed.



2. **Enter the network label and network header.**





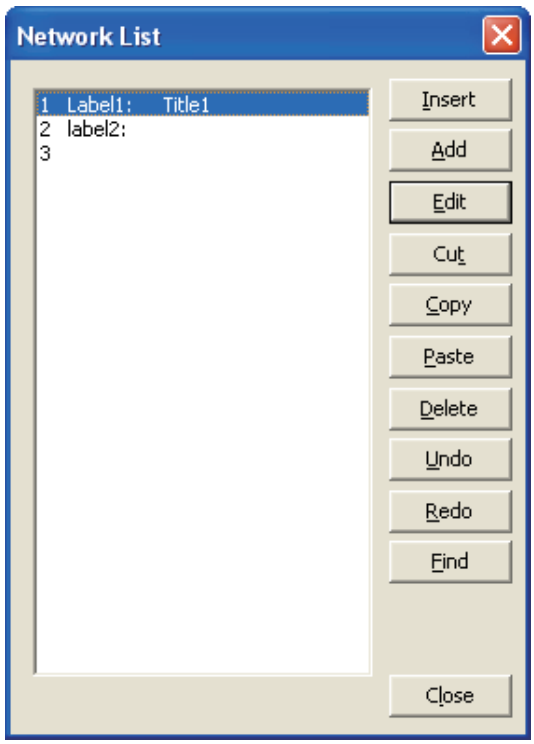
■ Editing listed networks

The following explains the method for using the Network List screen to edit listed networks in the program.
The following information of network header is displayed in the list.

- Network number
- Network label
- Network title

Screen display

Select [Edit] ⇒ [Network List].



Screen button

- Insert

  Inserts a new network in front of the selected network.

- Append

  Inserts a new network at the end of the existing networks.

- Edit

  Displays an editing screen of the network header.

- Cut , Copy , Paste , Delete , Undo , Redo

  Same functions as the buttons in program editors.

- Find

  Searches for a network based on the network label.

- Close

  Closes the Network List screen.

6.2.5 Undo and redo

This section explains the method for restoring or redoing the previous editing status.

● Undo

Operation

• Select [Edit] ⇒ [Undo] ( ).
The last edited operation is deleted.

● Redo

Operation

• Select [Edit] ⇒ [Redo] ( ).
The operation deleted by clicking the [Undo] button is recovered.

6.2.6 Displaying grid

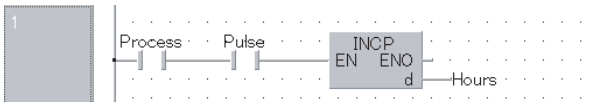
This section explains the method for displaying the grid which indicates specific positions, such as tab stop positions and the start/end position of a line, in a program editor.
To switch the grid display in a program editor, perform the following operation.

Operating procedure

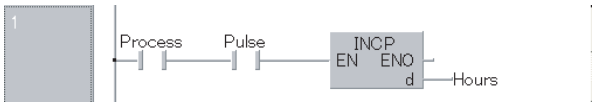
• Select [View] ⇒ [Grid].

Example: Structured ladder editor

<With grid>



<Without grid>



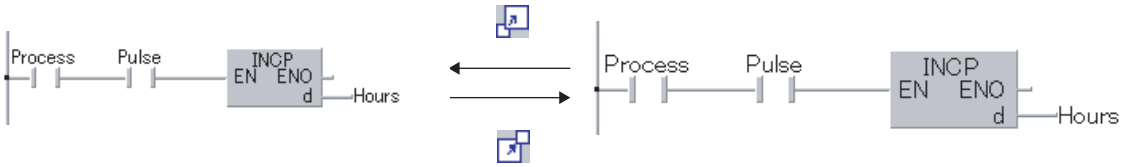
6.2.7 Changing display size of networks

This section explains the method for changing the display size of networks in the program editor by the following magnifications.
(Clicking one of the items on the toolbar raises or lowers the magnification one step at a time.)

- 50%
- 75%
- 100%
- 150%

Operating procedure

- Select [View] ⇒ [Zoom] ⇒ (Desired magnification) (/).



6.2.8 Opening label setting and program screens for selected POU

This section explains the method for displaying a label setting or program of the function or function block selected in the program editor.

● Displaying a label setting screen

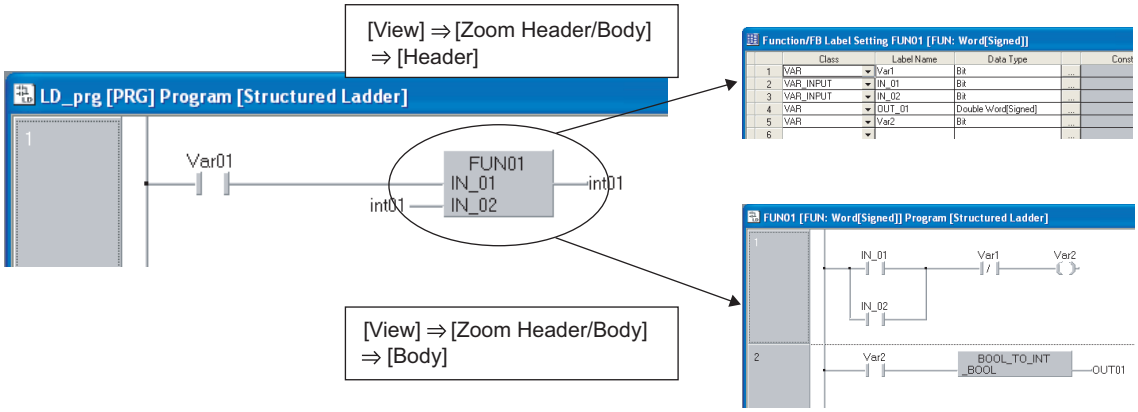
Operation

• Select [View] ⇒ [Zoom Header/Body] ⇒ [Header].

● Displaying a program editor

Operation

• Select [View] ⇒ [Zoom Header/Body] ⇒ [Body].



Point

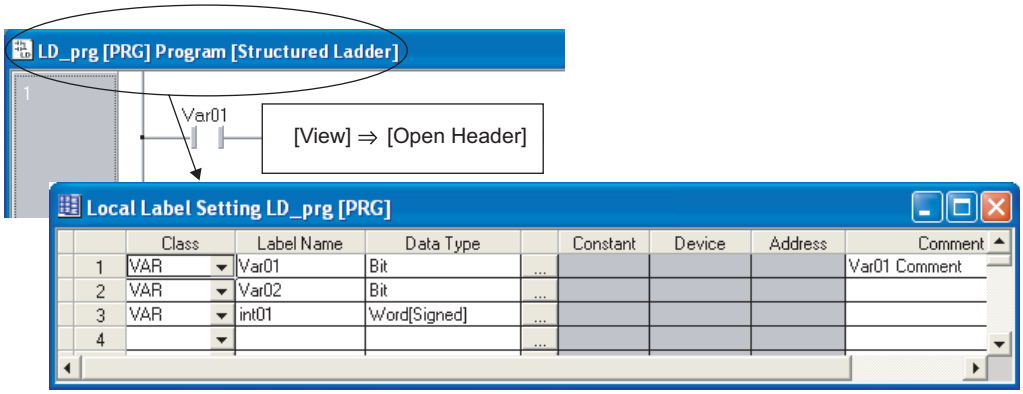
● **Displaying by double clicking the function or function block**
Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "FB/FUN" and set "Double clicking opens header" or "Double clicking opens body".

6.2.9 Opening label setting screen of program editor being edited

This section explains the method for displaying a label setting screen of the program editor being edited.

Operating procedure

- Select [View] ⇒ [Open Header].



6.2.10 Setting display contents of tool hint

This section explains the method for displaying contents defined on the label setting screen as tool hints by placing the cursor on a label name in the program editor.

The contents excluding label names displayed in the tool hints can be set by the following operation.

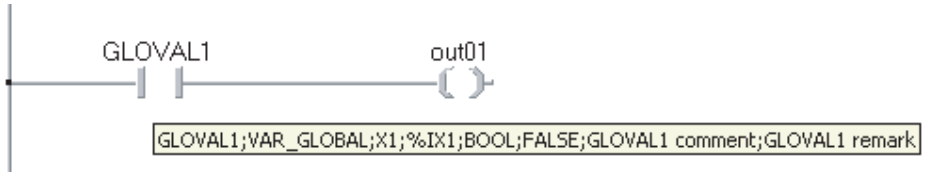
Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder/ST" ⇒ "Tool Hint".**

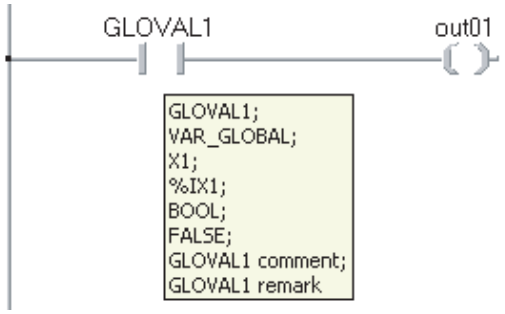
2. **Set the following items:**

Item		Description	Example
Tool Hint Display Items	Monitored Value	Displays a monitoring result.	(Displayed only during monitoring)
	Class	Displays a label class.	VAR_GLOBAL
	Device	Displays a device assigned to the label in the device format.	X1
	Address	Displays a device assigned to the label in the address format.	%IX1
	Type	Displays a data type of the label.	Bit
	Constant Value	Displays a constant value of the label.	FALSE
	Comment	Displays label comments.	GLOBAL1 comment
	Remark	Displays a remark of the global label.	GLOBAL1 remark
Tool Hint Display Format	Single Line	Displays tool tips in a single line.	
	Multi Line	Displays tool tips in multiple lines.	

<Single line>



<Multiple lines>



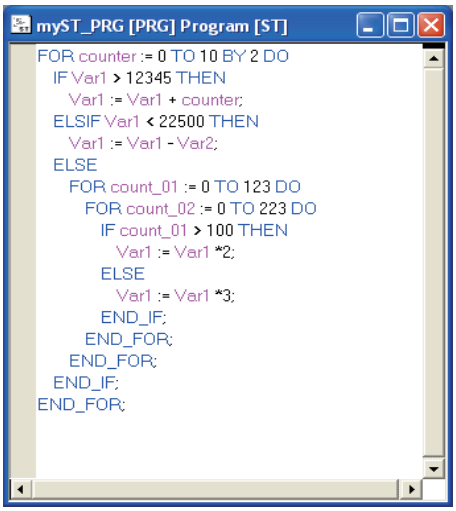
6.3 Editing ST Programs

QCPU
Q

This section explains the method for editing programs in the ST editor.
The ST editor is a language editor in text format for creating programs in the ST language.
The operating method is the same as that of general text editor. Spaces, tabs and line feeds can be entered between keywords and variable names of the ST control syntax.

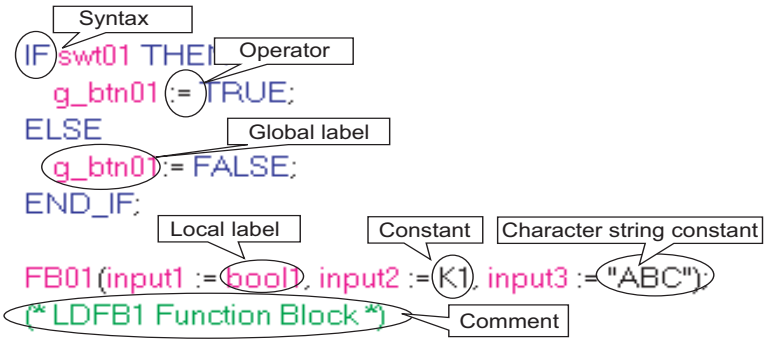
Screen display

Select Project window ⇒ "POU_Pool" ⇒ "(POU)" ⇒ "Program [ST]".



Point

- **Networks**
  Networks are not available in the ST editor.
- **Display colors**
  In the ST editor, the following display colors can be specified by selecting [View] ⇒ [Colors].
  The colors in the parentheses indicate the default colors.
  - Syntax (Blue)
  - Operator (Black)
  - Comment (Dark green)
  - Constant (Black)
  - Character string constant (Black)
  - Global label (Magenta)
  - Local label (Magenta)



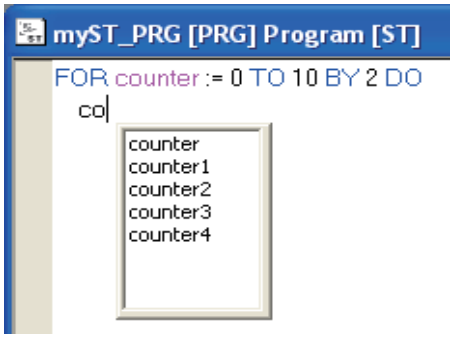
For details, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

6.3.1 Displaying applicable label names

The list of applicable label names that start with entered characters are displayed when entering a label name.

Operating procedure

1. **Enter an initial character (or character string) of a label name.**
   A list of labels that begin with the entered character(s) is displayed.
   If only one label is listed, the label name is automatically inserted to the program.



2. **Select a label name from the list.**
   The selected label name is inserted to the program.

Point

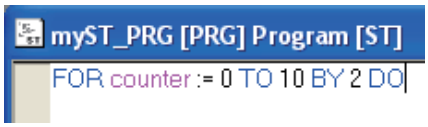
- **Displayed labels**
  - Global labels and local labels of the corresponding POUs are displayed.

6.3.2 Automatic indention

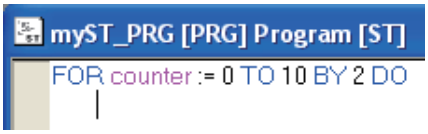
This section explains the method for automatically inserting a tab as the indention at the head of a new line when a line feed is inserted during program editing.
The indention can be set in the option setting.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "ST".**



2. **Select "Auto Indention".**



3. **In the ST editor, enter the ST control statement (such as a FOR statement).**
   When a line feed is inserted, the next line is indented based on the contents of the statement.

6.3.3 Setting tab length

This section explains the method for setting a tab length in the option setting.

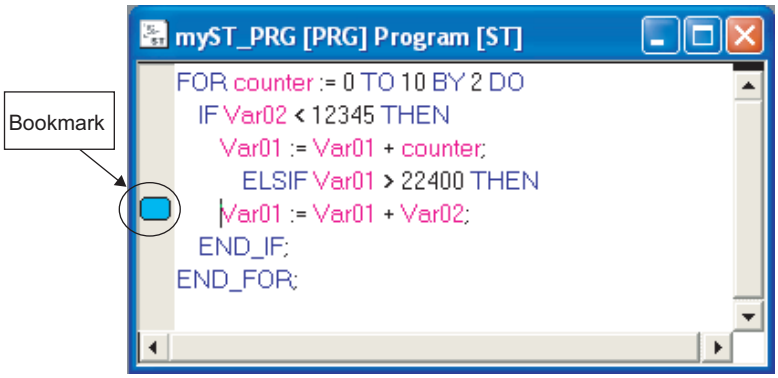
Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "ST".**

2. **Set the "Tabulator Length".**

6.3.4 Using bookmarks

This section explains the method for using bookmarks that are placed at positions in a program for frequent reference.
Bookmarks allow quick reference to certain positions in a program.



● Setting/canceling a bookmark

Operation

- Select [Find/Replace] ⇒ [Bookmark] ⇒ [Toggle Bookmark] ( ).
  A new bookmark is set at, or the existing bookmark is cleared from, the line where the cursor is placed.

● Moving to the next bookmark

Operation

- Select [Find/Replace] ⇒ [Bookmark] ⇒ [Next Bookmark] ( ).
  The cursor moves to the start of the line where the next bookmark is placed.

● Moving to the previous bookmark

Operation

- Select [Find/Replace] ⇒ [Bookmark] ⇒ [Previous Bookmark] ( ).
  The cursor moves to the start of the line where the preceding bookmark is placed.

● Deleting all bookmarks

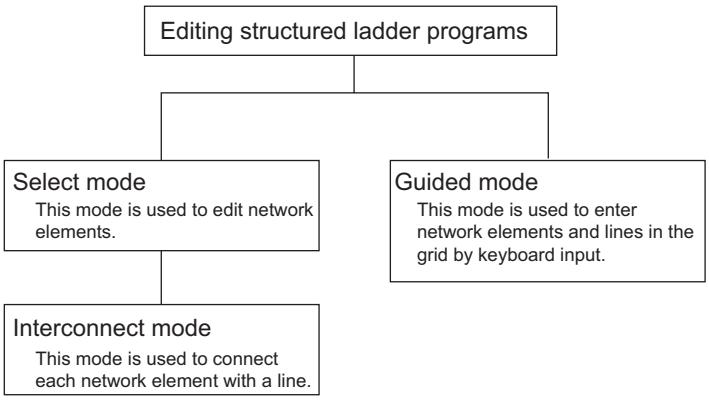
Operation

- Select [Find/Replace] ⇒ [Bookmark] ⇒ [Delete All Bookmarks] ( ).
  All bookmarks are deleted.

6.4 Editing Structured Ladder Programs

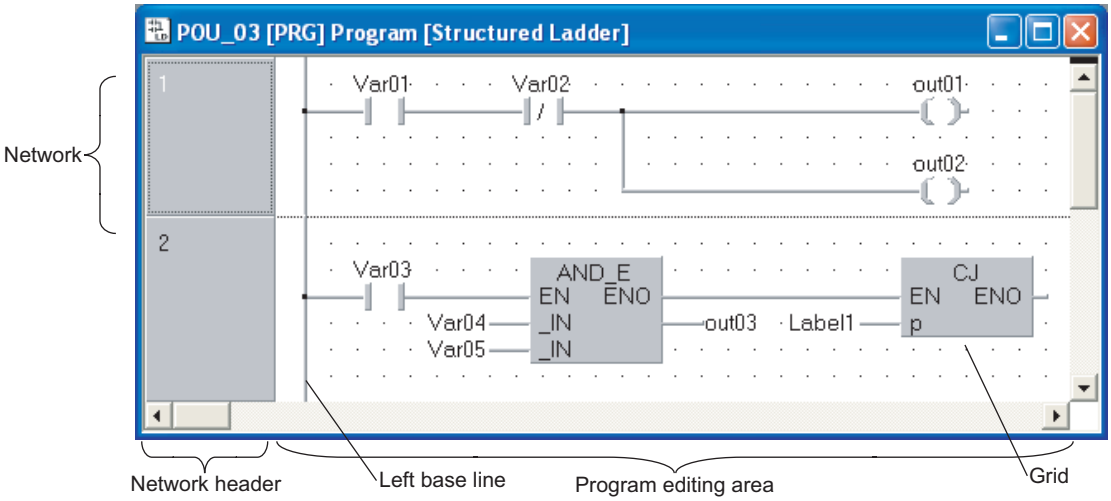
QCPU Q

This section explains the method for editing programs in the structured ladder editor.
The structured ladder editor is a language editor in graphic format for creating programs in the structured ladder language.
Network elements such as contacts, coils, and functions/function blocks are used to create structured ladder programs. Each network element is connected with a line.
The suitable editing mode can be selected according to the program entry method and editing target. (☞ Section 6.4.1)



Screen display

Select Project window ⇒ "POU_Pool" ⇒ "(POU)" ⇒ "Program [Structured Ladder]".



Display contents

The following table shows the display contents of the structured ladder editor.

Name	Description	Reference
Network	A unit for creating a program Consists of a network header and a program editing area.	Section 6.2.4
Network header	Displays information such as a title set for each network, and a label name of the jump target.	
Program editing area	An area for creating a program. Network elements can be placed in the grid area.	Section 6.4.1
Grid	Indicates specific positions such as tab stop positions and start/end positions of lines in a program editor.	Section 6.2.6
Left base line	A start position of the structured ladder program.	–

6.4.1 Selecting editing modes

This section explains the method for selecting a sequence program editing mode in the structured ladder editor.
The followings are the available modes.

● Select mode

Network elements such as contacts and coils can be placed as desired in the network. (☞ Section 6.4.3)
This mode is used for editing programs mainly with the mouse.

Operation

• Select [Edit] ⇒ [Select Mode] ( ).

● Interconnect mode

Network elements can be connected with lines. (☞ Section 6.4.4)
This mode is used for editing programs mainly with the mouse.

Operation

• Select [Edit] ⇒ [Interconnect Mode] ( ).

● Guided mode

Network elements such as contacts, coils, and instructions are placed within the grid frames. (☞ Section 6.4.12)
Lines can also be edited.
This mode is used for editing programs mainly with the keyboard.

Operation

• Select [Edit] ⇒ [Guided Mode] ⇒ [Guided Editing] ( ).

■ Displays in different editing modes

The following table shows different appearances of the mouse cursor and grid in different editing modes.

Table 6.4.1-1 Displays in different editing modes for the structured ladder editor

Mode type	Menu and toolbar	Auto connect	Mouse cursor	Grid display
Select mode	[Edit] ⇒ [Select Mode]	OFF		
		ON		
Interconnect mode	[Edit] ⇒ [Interconnect Mode]	OFF		
		ON		
Guided mode	[Edit] ⇒ [Guided Mode] ⇒ [Guided Editing]	OFF		

6.4.2 Entering network elements

This section explains the method for entering network elements such as contacts and coils.
The following table shows the network elements that can be selected for creating programs.

Table 6.4.2-1 Network elements used in the structured ladder editor

Network element		Toolbar	Shortcut key
Contact	Contact		1
	Contact Negation		2
	L-Contact		3
	L-Contact Negation		4
Line	Vertical Line		5
	Horizontal Line		6
Coil			7
Input Variable			9
Output Variable			0
Jump			Ctrl + J
Return			Ctrl + R
Comment			Ctrl + M

Operating procedure

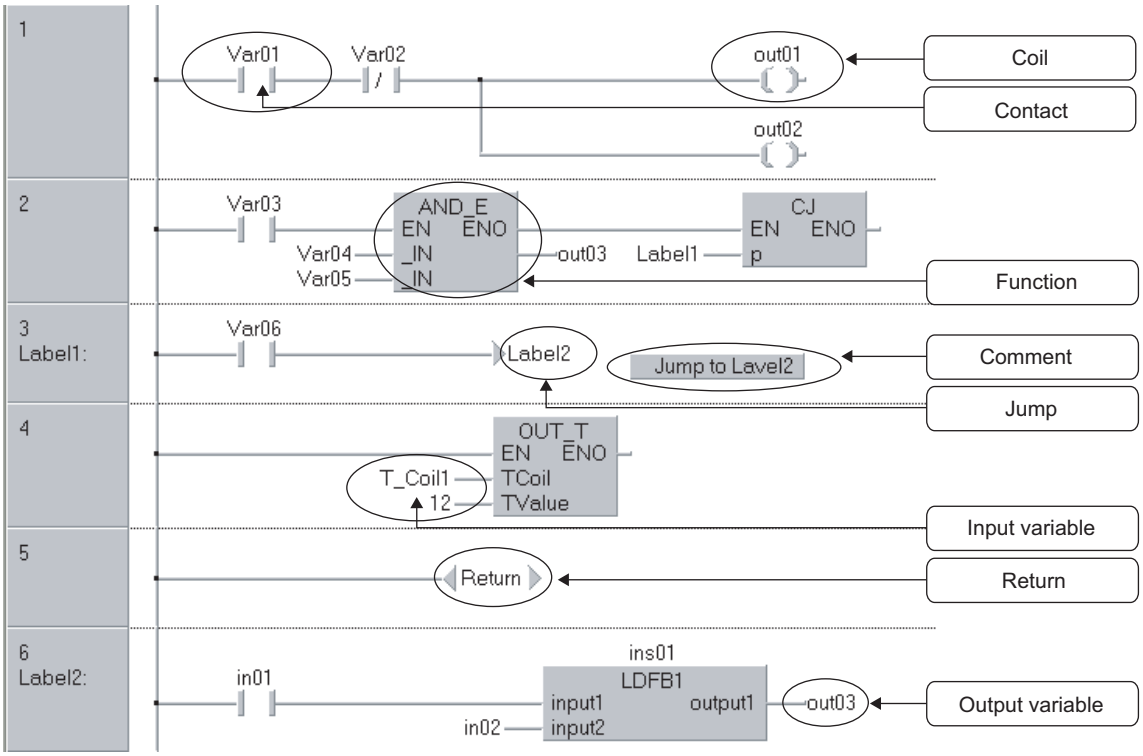
1. **Select [Edit] ⇒ [Network Elements] ⇒ [(A network element to be entered)].**
   A graphical image of the selected network element is appended to the mouse cursor.

2. **Click the position where the network element is entered.**
   The selected network element is entered.

Point

● **Entering POUs**
Use the Function Block Selection window to enter operators, functions or function blocks. (☞ Section 6.2.1)

<Display example of network elements>

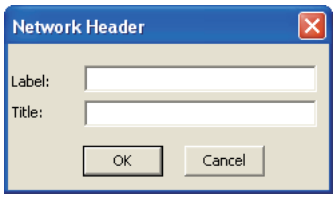


■ Setting network labels/titles

The following explains the method for setting a network label/title on the Network Header screen.

Operating procedure

1. **Select [Edit] ⇒ [Network Element] ⇒ [Network Label].**



2. **Set the network label and network title, and click the** OK **button.**

6.4.3 Moving network elements

This section explains the method for moving network elements such as contacts and coils.

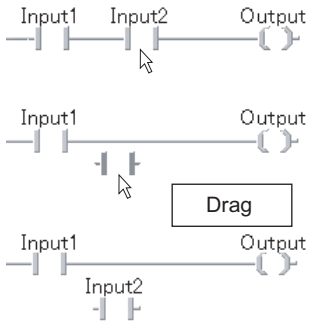
Operating procedure

1. **Select [Edit] ⇒ [Select Mode] ( ).**

   The cursor changes to .

2. **Click a network element.**

3. **Drag it to the desired position.**



■ Moving network elements with lines connected

The following explains the method for moving a network element with the lines connected by activating the Auto connect function of lines in the Select mode.

Operating procedure

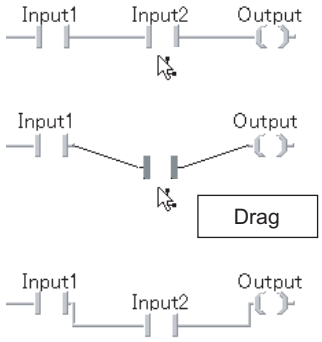
1. **Select [Edit] ⇒ [Auto Connect].**

   The cursor changes from to .

2. **Click a network element.**

3. **Drag it to the desired position.**

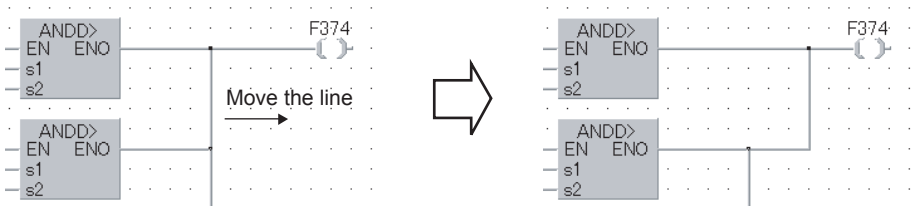
   The network element moves with the lines connected.



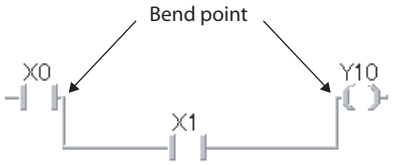
Point

● **Moving lines when the Auto connect function is deactivated.**
Some specific lines can be moved with lines connected even when the Auto connect function is deactivated.

Example) In the case of program shown below, a line can be moved with lines connected.

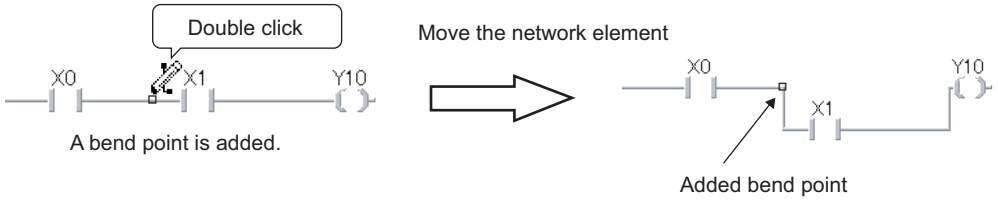


● **Bend points of the line**
The bend points of the line when the network element is moved, are at the adjacent points of network elements.

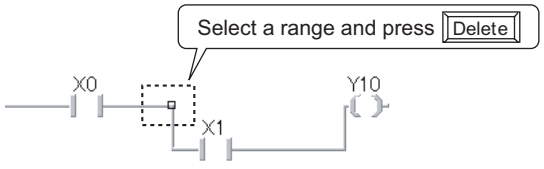


A bend point can be changed to any desired point by adding a bend point on the line as shown below.
When adding a bend point, select [Edit] ⇒ [Interconnect Mode] ( ).

[Interconnect Mode] ( ) [Select Mode] ( )



Note that when deleting the added bend point, delete not only the line but the bend point as well as shown below.



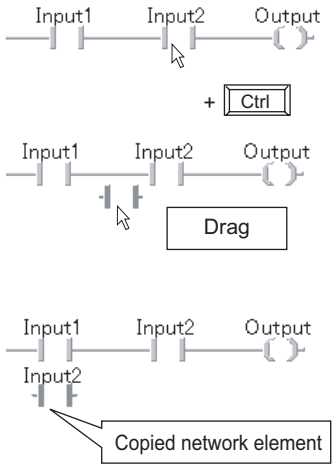
■ Copying network elements

The following explains the method for copying a network element within the editor or to another structured ladder editor displayed on the screen.

● Copying network elements by the drag-and-drop operation
Network elements can be copied within the editor.

Operation

1. Hold down the Ctrl key and click the network element.
2. Drag it to the desired position.



● Copying network elements using the clipboard

Network elements can be copied within the same editor or to another structured ladder editor displayed on the screen by the basic menu options using a clipboard or shortcut keys.

Operation

1. Select [Edit] ⇒ [Cut] ( )/[Copy] ( ).
2. Select [Edit] ⇒ [Paste] ( ).
3. Click the desired position. The upper left of the pasted position is the position to be clicked.

Point

- **Copying network elements to other networks**
  Network elements cannot be copied or moved to other networks by the drag-and-drop operation when the Auto connect function is activated.
  In order to copy or move network elements, deactivate the Auto connect function or use the clipboard.
- **Copying network elements by the drag-and-drop operation when the Auto connect function is activated**
  The line of the copied network element is connected automatically when the Auto connect function is activated. If the line is not connected properly or network elements are overlapped, deactivate the Auto connect function and correct the line.

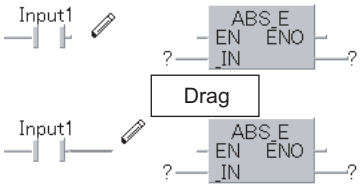
6.4.4 Connecting lines

This section explains the method for connecting network elements with a line.

Operating procedure

1. **Select [Edit] ⇒ [Interconnect Mode] ( ).**

   The cursor changes to .



2. **Click the start position of the line.**

3. **Drag the line to its end position.**

   A line is drawn in a vertical or horizontal straight line.

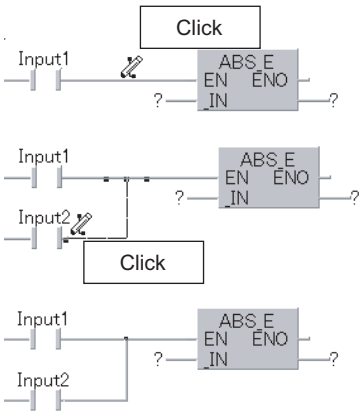
■ Connecting lines automatically

The following explains the method for connecting a line automatically using the Auto connect function by specifying the start and end points.

Operating procedure

1. **Select [Edit] ⇒ [Auto Connect].**

   The cursor changes from to .



2. **Click the start position.**

3. **Click the end position.**

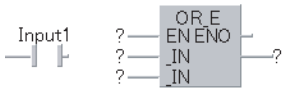
   The line is automatically routed.

■ Overwriting input/output variable with lines

The following explains the method for overwriting an input/output variable of a function or function block with a line.

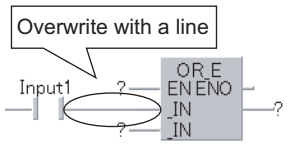
Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "FB/FUN".**



2. **Select "Pin overwrites".**

3. **Select [Edit] ⇒ [Auto Connect].**



4. **Draw a line over the input or output variable.**

   The existing input/output variable is automatically overwritten with the line.

■ Rearranging lines automatically

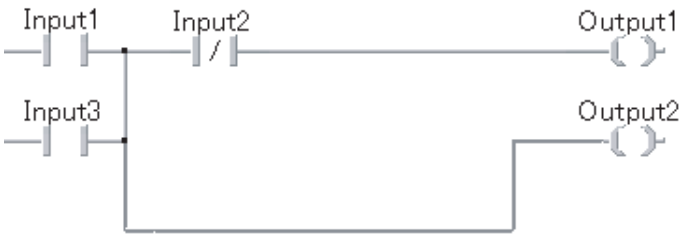
The following explains the method for rearranging and redrawing a line when the Auto connect function is activated.

Operating procedure

1. **Select the line to be rearranged.**

2. **Select [Edit] ⇒ [Recalculate Line].**

<Before automatic arrangement execution>



<After automatic arrangement execution>

6.4.5 Inserting and deleting rows and columns

■ Inserting rows and columns

The following explains the method for inserting a row or column when editing a program.

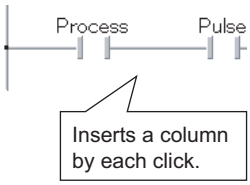
Operating procedure

1. **Select [Edit] ⇒ [Insert Row] ( )/[Insert Column] ( ).**

   The cursor changes to or .

2. **Click the position where a row or column is inserted.**

   A row or column is inserted by each click.



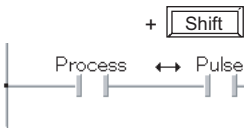
■ Deleting rows and columns

The following explains the method for deleting a row or column when editing a program.

Operating procedure

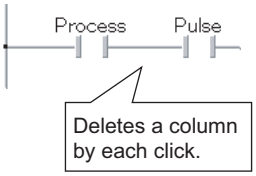
1. **Select [Edit] ⇒ [Insert Row] ( )/[Insert Column] ( ).**

   The cursor changes to or .



2. **Hold down the Shift key and click the position where a row or column is deleted.**

   A row or column is deleted by each click.



6.4.6 Changing type of contact and coil

This section explains the method for changing the type of contact and coil.

■ Setting the type of contact and coil

The following table shows the types that can be set for contacts and coils.

Table 6.4.6-1 Type of contact and coil

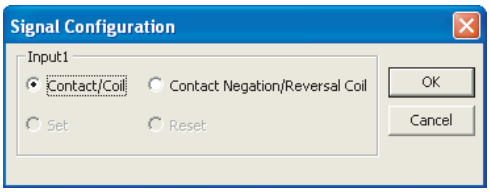
Type		Network element
Contact	Normal	-| |-
	Negation	-|/|-
Coil	Coil	-( )-
	Reversal coil	-(/)-
	Set	-(S)-
	Reset	-(R)-

The following explains the method for setting the type of contact or coil.

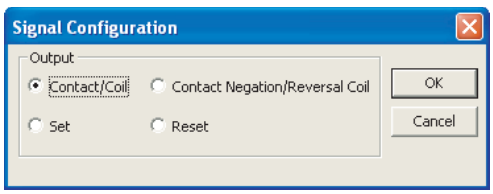
Operating procedure

1. **Select a contact or coil whose type is changed.**
2. **Select [Edit] ⇒ [Signal Configuration] ⇒ [Configure].**
3. **Select the type on the Signal Configuration screen.**
4. **Click the OK button.**
   The selected contact or coil is set.

<Contact selected>



<Coil selected>



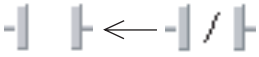
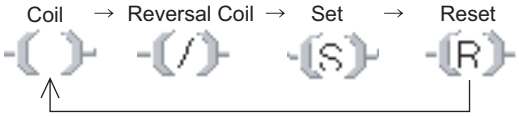
Point

● **Setting the type of contact and coil**
The type of contact or coil can be set by double clicking a contact or coil.

■ Changing the type of contact and coil in the predetermined order

The following explains the method for changing the type of contact and coil in the predetermined order.
The following table shows the changing order of the type of contact and coil.

Table 6.4.6-2 The changing order of the type of contact and coil

Type	Change sequence
Contact	Contact → Contact Negation 
Coil	Coil → Reversal Coil → Set → Reset 

The following shows the method for changing the type of contact or coil.

Operating procedure

1. **Select a contact or coil whose type is changed.**

2. **Select [Edit] ⇒ [Signal Configuration] ⇒ [Toggle].**

6.4.7 Changing the number of arguments for functions and function blocks

The number of arguments can be changed for some functions and function blocks of the application functions.
For details on the application functions, refer to the following manual.
☞ QCPU Structured Programming Manual (Application Functions)

To add or delete an argument, select the target and proceed the following operation.

● Adding an argument

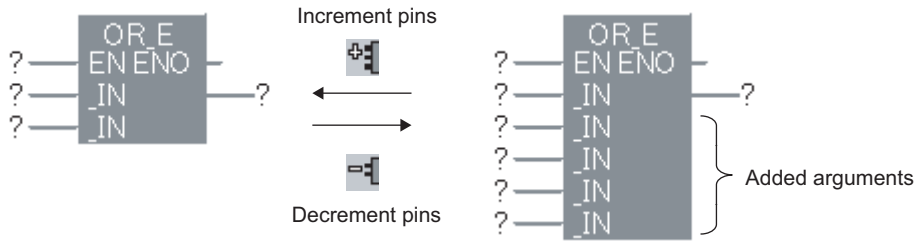
Operation

• Select [Edit] ⇒ [Number of Pins] ⇒ [Increment] ( ).

● Deleting an argument

Operation

• Select [Edit] ⇒ [Number of Pins] ⇒ [Decrement] ( ).



Point

● **Adding and deleting arguments**
Arguments can also be added or deleted using the following method.
• Using the [+] or [−] key
• Changing the height of the network element using the mouse

6.4.8 Adding variables automatically when entering functions and function blocks

This section explains the method for configuring the option setting in order to add input/output variables automatically when functions or function blocks are entered. (This setting is enabled only when the Auto connect function is activated.)

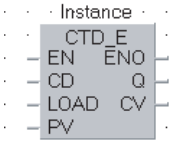
Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "FB/FUN".**

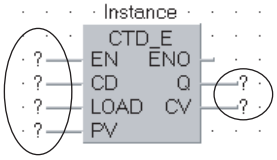
2. **Select "Automatic Input/Output variables".**
   To add an output variable to ENO, select "Automatic ENO variables".

3. **Select [Edit] ⇒ [Auto Connect].**

<Without adding input/output variables>



<Adding input/output variables>



6.4.9 Editing variable names and instance names

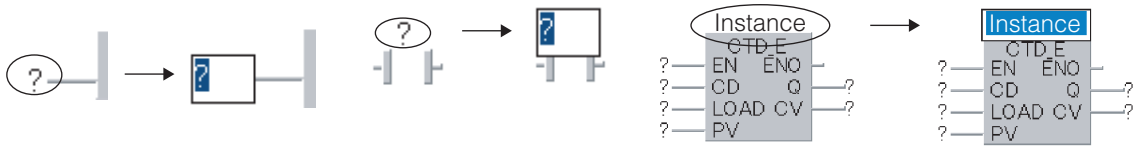
This section explains the method for editing a variable name or function block instance name.

Operating procedure

1. **Click the entry field of a network element such as a contact, coil, or input/output label.**
   The entry field is enabled for editing.

2. **Edit the variable name or instance name.**

<Input variable> <Contact> <Function block>

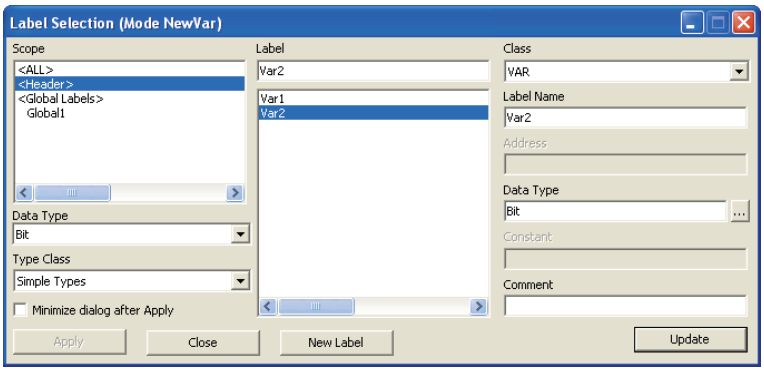


Point

- **Using set label names**
  Right click the entry field when it is enabled for editing, and select the label name on the Label Selection screen. (☞ Section 6.2.3)

■ Adding new label settings when entering labels

The following explains the method for adding a new label setting. When a label name not yet set is entered, the Label Selection screen (☞ Section 6.2.3) is displayed where label setting can be newly added.



Point

- **Displaying the Label Selection screen**
  The operations at the entry of labels can be changed by the following setting.
  Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "Label", and set the "Declare new label name".

6.4.10 Changing display format of variables

This section explains the method for changing the display format of variables in the editor.

■ Displaying variable names in multiple lines

The following explains the method for selecting a view mode to display variable names of contacts and coils in multiple lines by setting the option.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "Label".**
2. **Select "Display label name/comment of contact or coil in multiline".**
3. **Set the values of "Display Lines" and "Visible Characters per Line".**

<Single line> <Multiple lines>

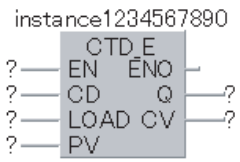
■ Displaying instance names in multiple lines

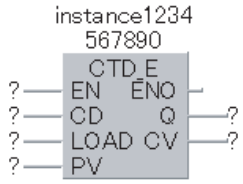
The following explains the method for changing a view mode to display wrapped function block instance name within the width of function block by setting the option.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "FB/FUN".**
2. **Select "Wrap instance name for function block".**

<Single line> <Wrapping a line>





■ Specifying the number of displaying characters for variable names

The following explains the method for changing the number of displaying characters for variable names of functions and function blocks by setting the option.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "FB/FUN".**

2. **Select "Specify the number of enable characters for label name/comment".**

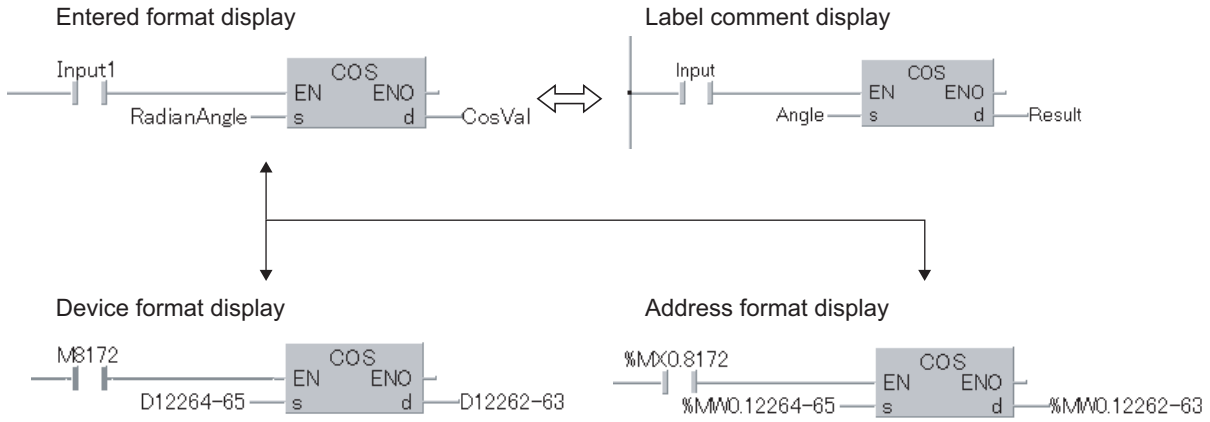
3. **Set the number of characters (2 to 255).**

<16 characters>

<8 characters>

■ Changing display format of variable

The following explains the method for changing the display format of variable to the device/address format.



● Displaying in the entered format

Operation

- Select [View] ⇒ [View Mode] ⇒ [Label].
  Variables are displayed in the format as they are entered.

● Displaying in the device format

Operation

- Select [View] ⇒ [View Mode] ⇒ [Device].
  Devices are displayed in the device format.
  As for labels, the assigned devices are displayed in the device format.

● Displaying in the address format

Operation

- Select [View] ⇒ [View Mode] ⇒ [Address].
  Devices are displayed in the address format.
  As for labels, the assigned devices are displayed in the address format.

● Displaying label comments

Operation

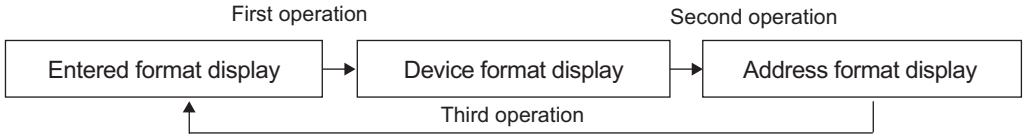
- Select [View] ⇒ [View Mode] ⇒ [Comment].
  Label comments are displayed.
  Device comments are not displayed.

● Switching label/device/address display

Operation

- Select [View] ⇒ [View Mode] ⇒ [Change Display Label-Device-Address].
  The display is switched in order (label, device, address) by selecting the menu each time.

[Change Display Label-Device-Address]

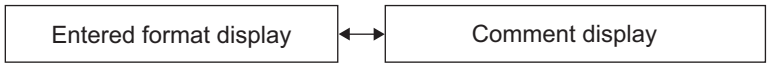


● Switching label/comment display

Operation

• Select [View] ⇒ [View Mode] ⇒ [Change Display Label-Comment].
The display is switched between label and comment by selecting the menu each time.

[Change Display Label-Comment]

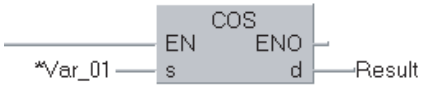


Point

● **Displaying uncompiled labels and label comments**
An asterisk (*) is prefixed to each label name of uncompiled labels to which devices are not assigned when the display format is changed to the device format or address format.
An asterisk (*) is prefixed to each label name of labels to which label comments are not set when the display format is changed to the comment display.

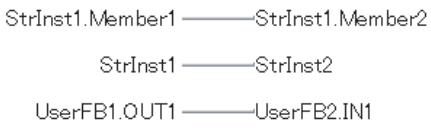
<Device format display>

<Comment display>

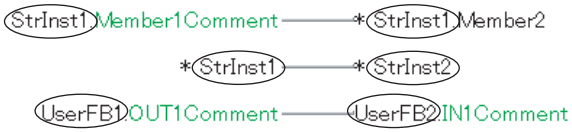


● **Structured data type/function block labels**
Structured data type/function block labels are displayed as label names even when the display format is changed to the comment display.

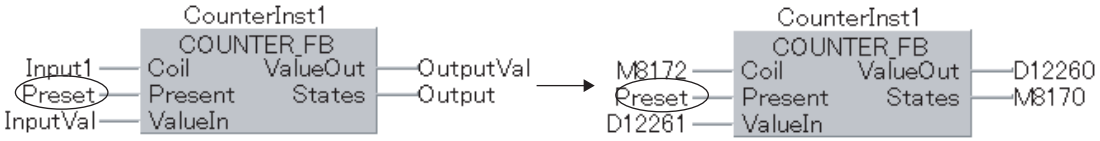
<Device format display>



<Comment display>



● **Constant type labels**
Constant type labels are displayed as label names even when the display format is changed to the device/address format.



● **Restrictions on operations when changing the display format**
Program editing and search/replace functions are disabled in the device/address format or comment display. To operate these functions, return the setting to [Standard].

6.4.11 Editing comments

This section explains the method for editing comments which can be entered same as other network elements. (☞ Section 6.4.2)

Operating procedure

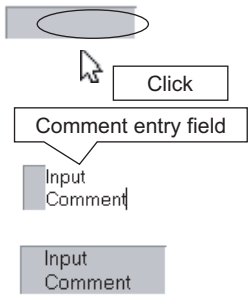
- Select [Edit] ⇒ [Network Elements] ⇒ [Comment]( ).

■ Editing comments

The following explains the method for editing a comment.

Operating procedure

1. **Click the end (right side) of the comment.**
   The comment entry field is enabled.
2. **Enter a comment.**

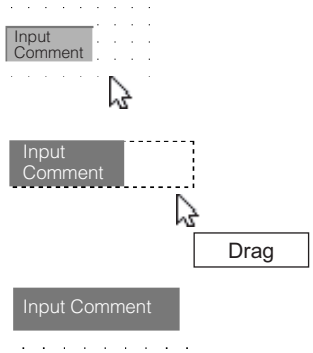


■ Changing the size of comments

The following explains the method for changing the size of a comment using a basic drag operation.

Operating procedure

1. **Click the bottom right corner of the comment.**
2. **Drag the corner to adjust the size.**

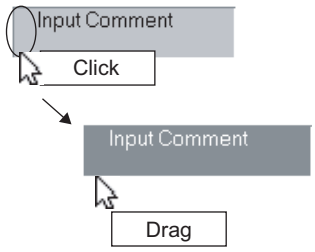


■ Moving comments

The following explains the method for moving a comment.

Operating procedure

1. **Click the head (left end) of the comment.**

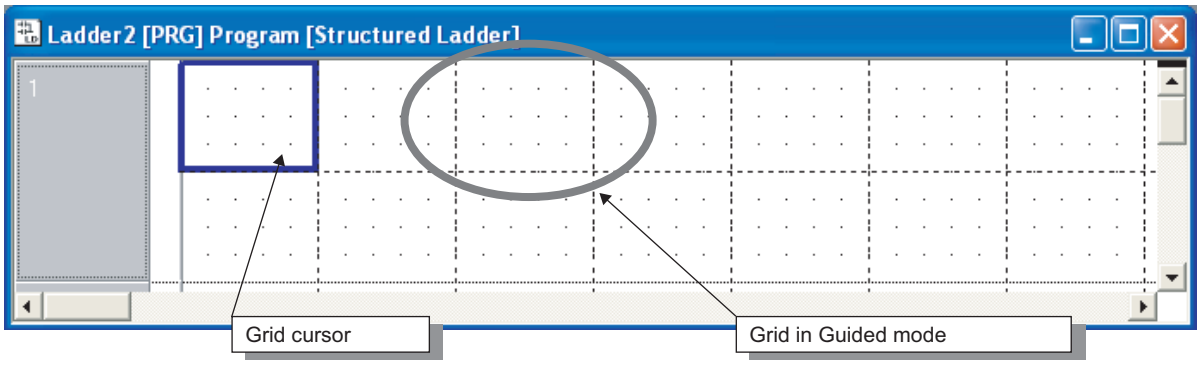


6.4.12 Editing in Guided mode

This section explains the Guided mode, which is convenient when data is entered mainly from the keyboard.

Screen display

Select [Edit] ⇒ [Guided Mode] ⇒ [Guided Editing] ( ).



■ Entering network elements

The following explains the method for entering a network element in the Guided mode.
In the Guided mode, one network element can be entered per one grid.
The network elements that can be entered in the Guided mode are the same as those in other modes. (☞ Section 6.4.2)

Operating procedure

- Select [Edit] ⇒ [Network Elements] ⇒ [(A network element to be entered)].
  The selected network element is entered at the grid cursor position.

■ Entering instructions

The following explains the method for entering instructions in the Guided mode.

Operating procedure

1. **Move the grid cursor to the position where an instruction is entered, and press the Enter key.**
   The Input Instruction screen is displayed.

2. **Enter the instruction and device, and click the Enter button.**
   The entered instruction is displayed at the grid cursor position.

■ Switching between overwrite and insert modes

The following explains the two entry methods, overwrite and insert modes, that can be selected in the Guided mode.

Table 6.4.12-1 Difference of the entering methods in the Guided mode

Menu	Description	Grid cursor color
[Edit] ⇒ [Guided Mode] ⇒ [Overwrite Mode]	Overwrites the existing network element at the cursor position with the entered network element.	Red
[Edit] ⇒ [Guided Mode] ⇒ [Insert Mode]	Moves the existing network element at the cursor position to the right and inserts a network element.	Blue

For switching between overwrite and insert modes, follow the operation below.

Operating procedure

- Press the Insert key.

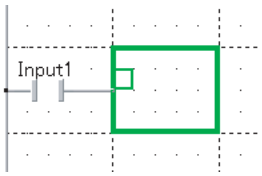
■ Connecting lines

The following explains the method for connecting lines in the Guided mode.

Operating procedure

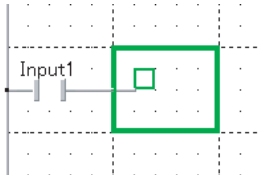
1. **Select [Edit] ⇒ [Guided Mode] ⇒ [Line Mode].**

   The thickness of grid cursor is doubled. The normal grid in the Guided mode is divided into 4×5 parts of smaller grids to specify a position.



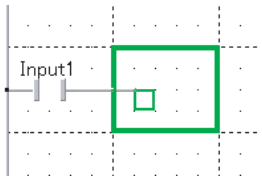
2. **Press the 6 button (Horizontal line).**

   A horizontal line is drawn by the length of one divided grid.



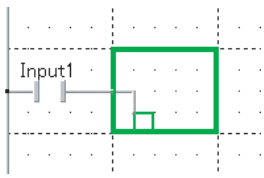
3. **Press the ↓ key.**

   The cursor moves to the next divided grid.



4. **Press the 5 button (Vertical line).**

   A vertical line is drawn by the length of one divided grid.



■ Drawing lines

The following explains the method for drawing a line.

Operating procedure

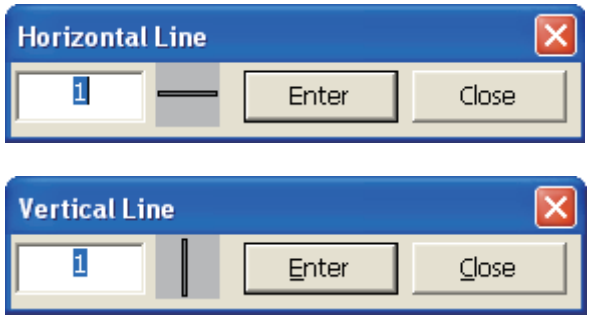
1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "Guided".**

2. **Select "Allow hotkey repeater".**

The following shows the method for drawing a line using the Horizontal Line or Vertical Line screen.

Operating procedure

1. **Press the [6] (Horizontal line) or [5] (Vertical line) button.**
   The Horizontal Line or Vertical Line screen is displayed.



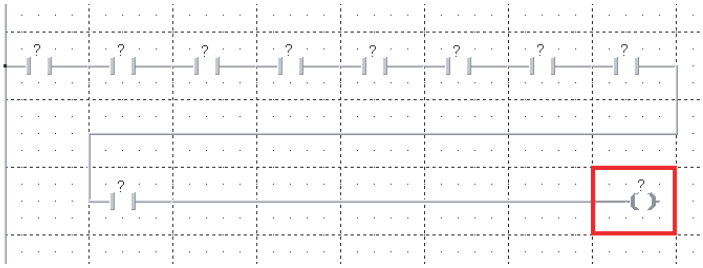
2. **Set the length of the line to be drawn using the grid width/height as the unit of length.**
   A line of the specified length is displayed.

■ Wrapping ladder

The following explains the method for wrapping a ladder if it does not fit in one line.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "Guided".**



2. **Select "Wrap ladder".**

Point

- **Target of wrapping ladder to be displayed**
  Ladders edited after setting the above option are wrapped. (The display of edited ladders before setting the above option is not changed.)

■ Setting Guided mode as the default editing mode

The following explains the method for setting the Guided mode as the default editing mode of the structured ladder editor.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "Guided".**

2. **Select "Set guided mode as default editing mode".**

■ Entering variable names after contacts and coils

The following explains the method for entering variable names after contacts and coils without selecting the entry field in the Guided mode.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "Guided".**

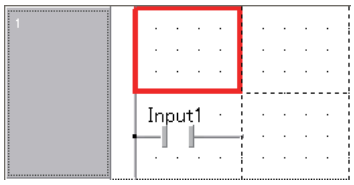
2. **Select "Enter variable names after contacts and coils".**

■ Inserting comment entry fields to networks to be added

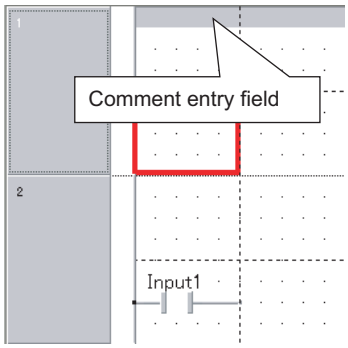
The following explains the method for inserting a comment entry field at the top of a network in the Guided mode.

Operating procedure

1. **Select [Edit] ⇒ [Guided Mode] ⇒ [Auto Comment] ( ).**

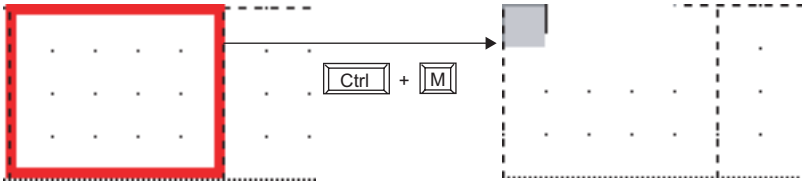


2. **Select [Edit] ⇒ [New Network] ⇒ [Before] ( ).**
   A network with a comment entry field is added.



Point

- **Width of automatically inserted comment**
  The width of automatically inserted comment can be changed by the following operation.
  Select [Tools] ⇒ [Options] ⇒ "Program Editor" ⇒ "Structured Ladder" ⇒ "Guided", and set the "Auto Comment Block Width".
- **Entering a comment in the desired grid**
  Hold down the Ctrl key and press the M key, and a comment can be entered to the grid at the cursor position.



7 COMPILING PROGRAMS

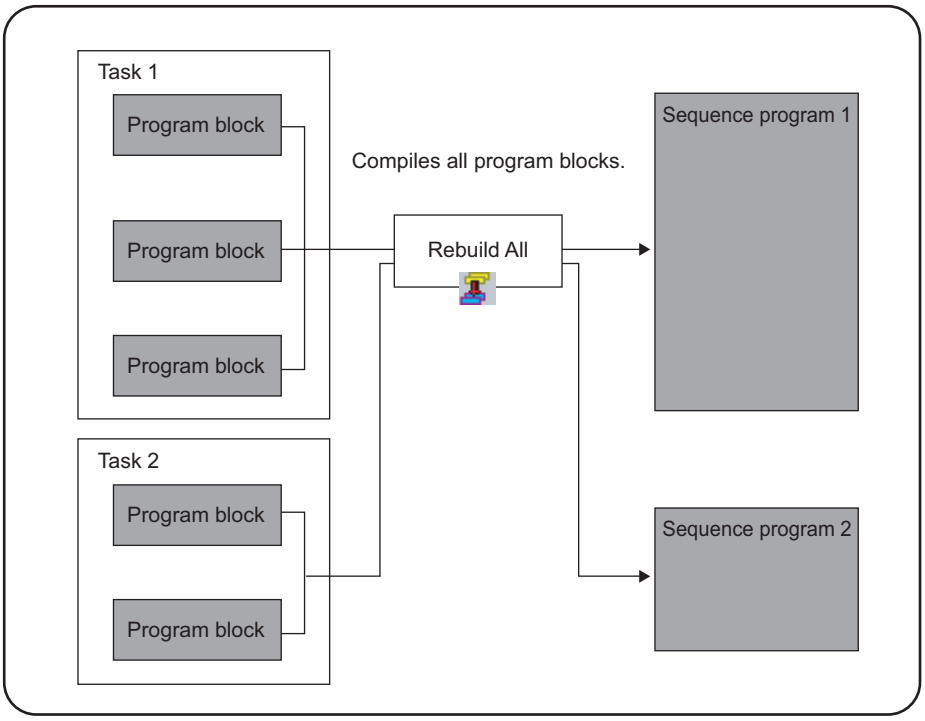
This chapter explains the method for compiling programs to sequence programs that can be executed on the programmable controller CPU.

7.1 Compiling All Programs 7-2
7.2 Compiling Only Modified Programs 7-4
7.3 Precautions on Compilation. 7-6
7.4 Executing Online Change Simultaneously with Compilation 7-7
7.5 Checking for Errors and Warnings when Compiling Programs . . . 7-7
7.6 Changing Operating Conditions of Compilation 7-8

7.1 Compiling All Programs

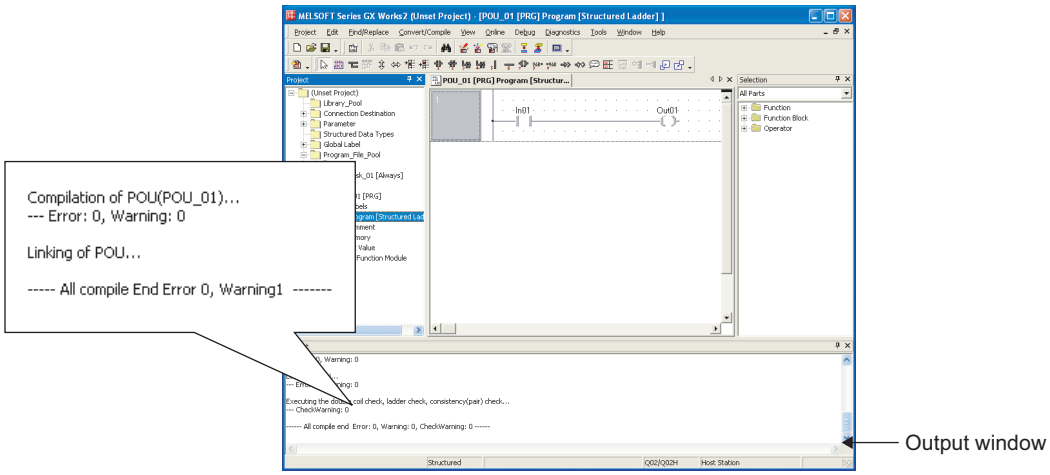
QCPU
Q

This section explains the method for batch compiling all program blocks that are registered to tasks.
Programs are compiled to sequence programs that can be executed on the programmable controller CPU.



Operating procedure

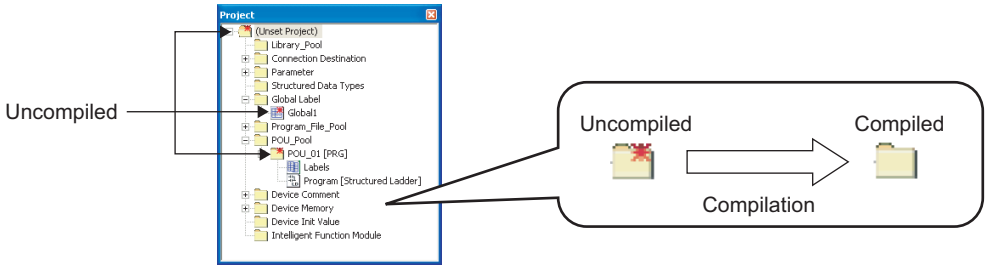
- Select [Convert/Compile] ⇒ [Rebuild All] ( ).
  All programs are compiled and the result is displayed in the Output window.



Point

● **Checking the compilation status**

The compilation status can be checked in the Project window. A symbol ( ) is appended to programs in uncompiled status.



● **Checking errors and warnings when compiling programs**

For checking errors and warnings when compiling programs, refer to the following section.

(☞ Section 7.5)

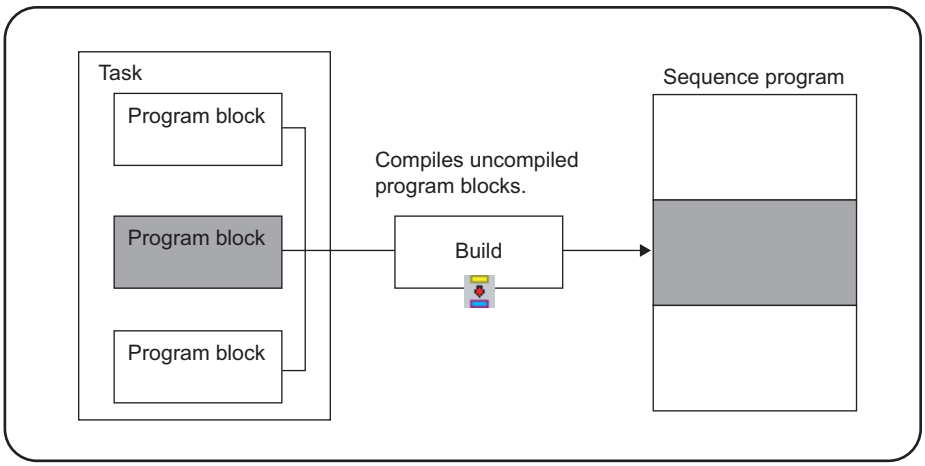
● **Label assignment when compiling all programs**

When all programs are compiled, devices are reassigned to labels of all POUs.

7.2 Compiling Only Modified Programs

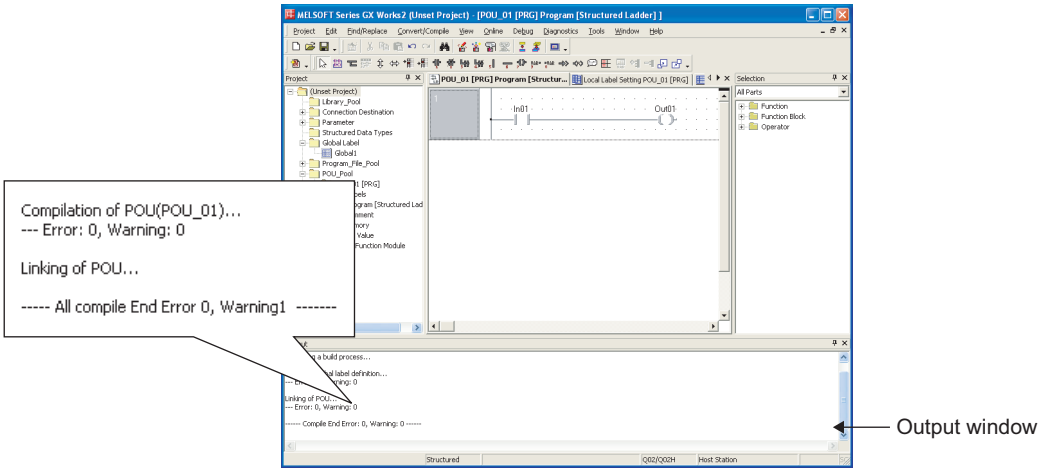
QCPU
Q

This section explains the method for batch compiling the uncompiled program blocks among all program blocks that are registered to tasks.
Programs are compiled to sequence programs that can be executed on the programmable controller CPU.
Since this process compiles only uncompiled programs, the compiling time can be reduced.



Operating procedure

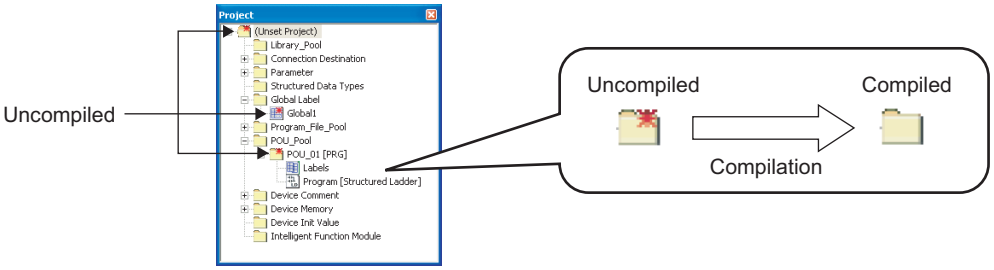
- Select [Convert/Compile] ⇒ [Build] ( ).
  Programs are compiled and the result is displayed in the Output window.



Point

● **Checking the compilation status**

The compilation status can be checked in the Project window. A symbol ( ) is appended to programs in uncompiled status.



● **Checking errors and warnings when compiling programs**

For checking errors and warnings when compiling programs, refer to the following section.

(☞ Section 7.5)

● **Label assignment when compiling programs**

When programs are compiled, devices are reassigned to labels of uncompiled POUs only.

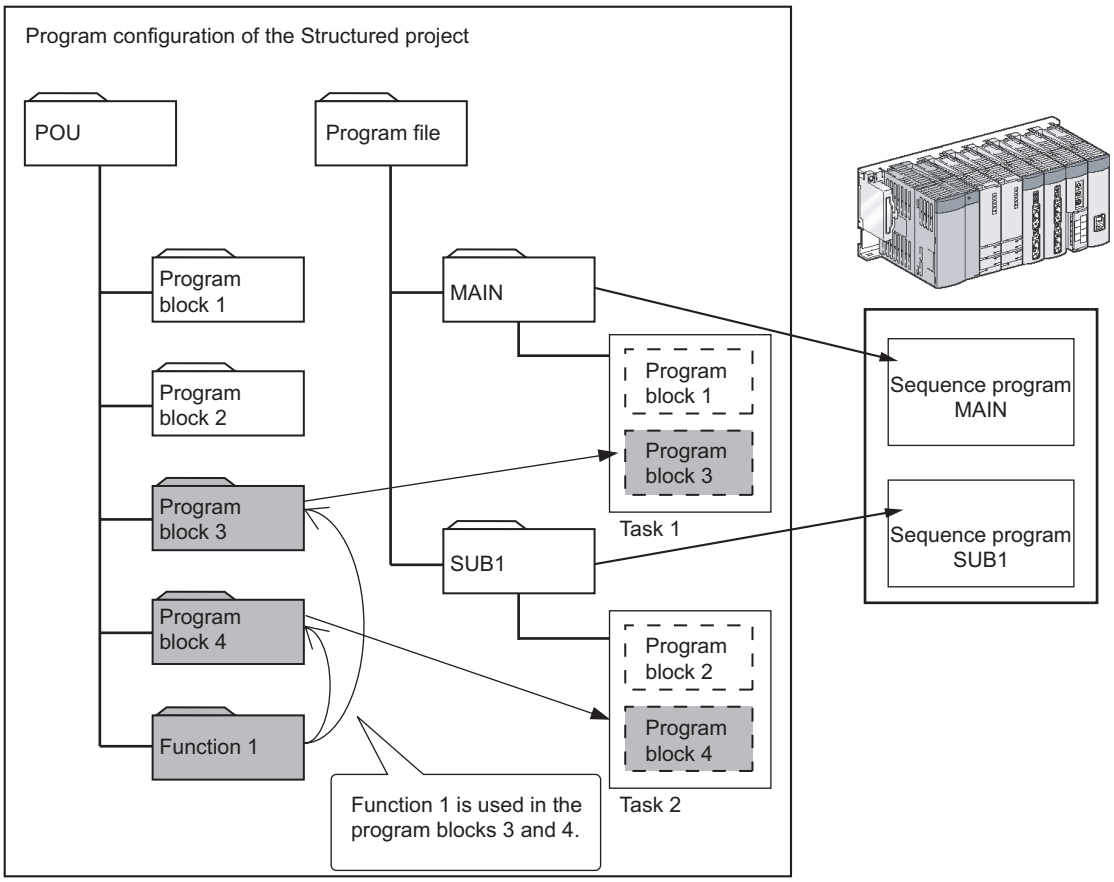
7.3 Precautions on Compilation

QCPU
Q

● Assigning devices automatically
When a program is compiled or all programs are compiled, the assignment of devices to labels is changed.
Therefore, the programmable controller CPU must be reset after writing programs to it.

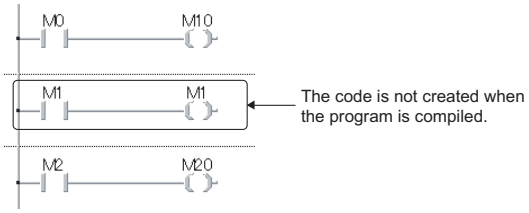
● Modification of global labels and functions/function blocks
If global labels and/or functions/function blocks are modified, multiple program blocks become the compile targets. Write all program files of the compile targets to the programmable controller CPU to apply the modifications.

Example) Compiling programs after modifying Function 1 in the following program configuration.
Program blocks 3 and 4 are compiled and the program files MAIN and SUB1 are modified.



● Compiling structured ladder/ST programs
The code is not created when the program as shown below is compiled in the structured ladder/ST program.

Example) The same device is used for a contact and coil



7.4 Executing Online Change Simultaneously with Compilation

For the operation for executing the Online change simultaneously with the compilation, refer to the following manual.

☞ GX Works2 Version1 Operating Manual (Common)

7.5 Checking for Errors and Warnings when Compiling Programs

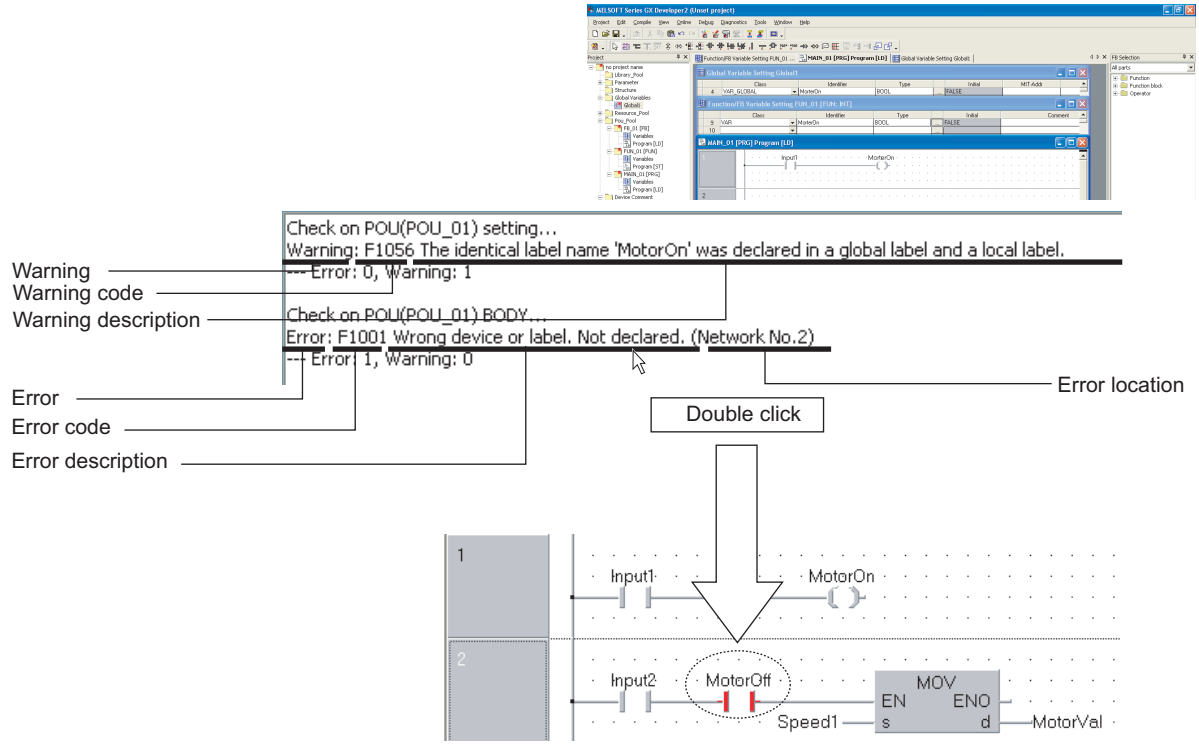
QCPU
Q

This section explains the method for checking errors/warnings when compiling programs.
When all or uncompiled programs are compiled, the target programs and label settings are checked and the results are displayed in the Output window.
The following explains the checking method and corrective action to be taken on occurrence of errors and warnings.

Operating procedure

1. **Double click the error/warning message displayed in the Output window.**

   The corresponding error location in the program is displayed.



2. **Check the corresponding error location and correct the program as instructed by the error/warning message.**

7.6 Changing Operating Conditions of Compilation

QCPU
Q

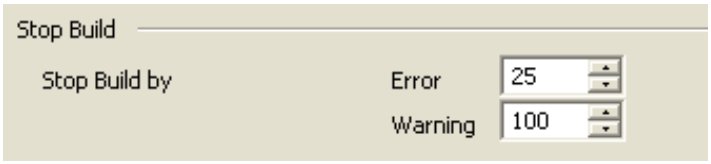
This section explains the method for changing the operating conditions of compilation.

7.6.1 Changing the number of errors and warnings that stops compilation

This section explains the method for changing the number of errors and warnings that stops compilation.
If the number of errors and warnings that occurred during compilation reaches the specified value, the compilation is aborted.

Screen display

Select [Tools] ⇒ [Options] ⇒ "Compile" ⇒ "Output Result".



Display contents

- Set the items on the screen.

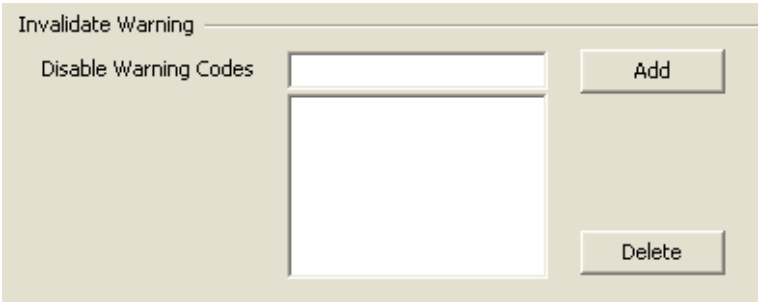
Item	Description
Error	Sets the number of errors that stops compilation (1 to 9999).
Warning	Sets the number of warnings that stops compilation (1 to 9999).

7.6.2 Hiding warning messages

This section explains the method for hiding messages displayed in the Output window when compiling programs.
The registered warnings are not displayed in the Output window.

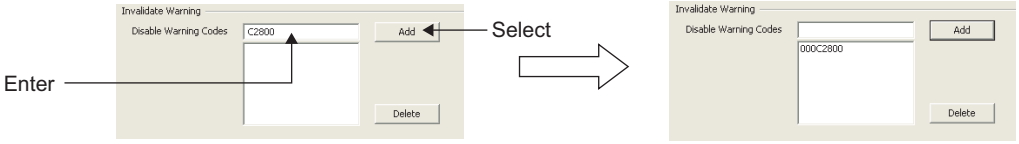
Screen display

Select [Tools] ⇒ [Options] ⇒ "Compile" ⇒ "Output Result".

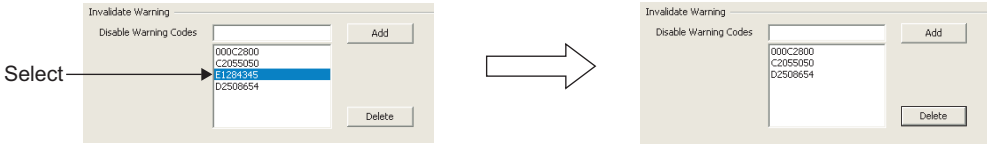


Operating procedure

- To register the warnings, enter the warning code and click the Add button.
  The registered warnings are not displayed in the Output window.

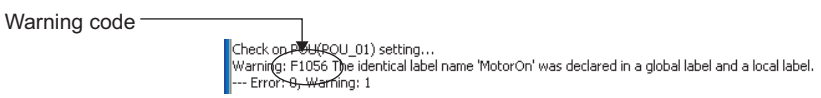


- To cancel the registration, select the warning code and click the Delete button.
  The unregistered warnings are displayed in the Output window.



Point

● **Warning codes**
Warning codes and their contents can be checked in the Output window when compiling programs.
(☞ Section 7.5)



● **Copying warning codes**
Warning codes displayed in the Output window can be copied (Ctrl + C) and pasted (Ctrl + V) to the "Disable Warning Codes" entry field.

● **Maximum number of warnings to be invalidated**
The maximum number of warnings to be invalidated is 100.

7.6.3 Using same name for global label and local label

This section explains the method for using a same name for a global label and a local label.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Compile" ⇒ "Structured Ladder/ST" ⇒ "Compile Condition1".**

2. **Check the "Use the same label name in global label and local label" check box.**
   - If this check box is not checked, an error occurs when compiling programs.
   - If this check box is checked, a warning is displayed when compiling programs (Same label names can be used).

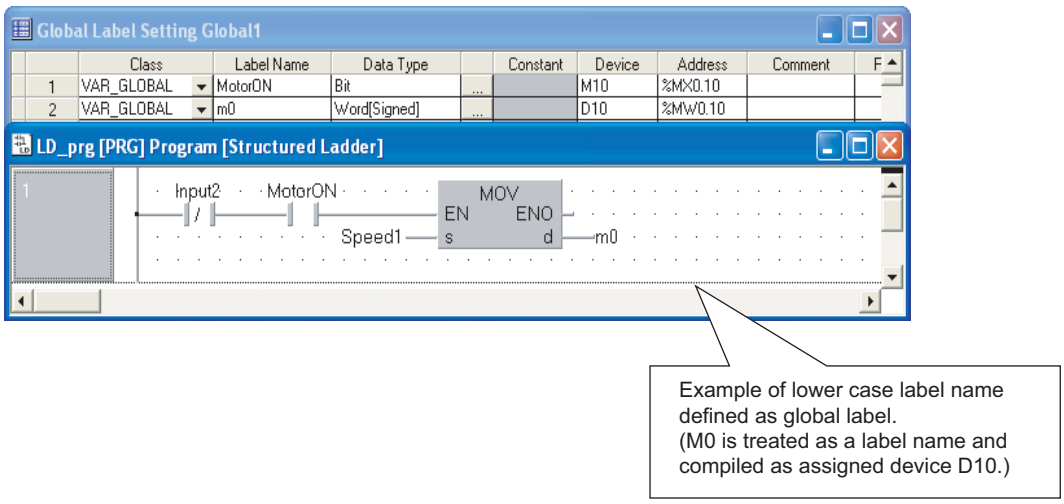
7.6.4 Using same lower case label name as device

This section explains the method for using a label name (for example: m0, x1F) that contains a lower case of alphabetic part of the device.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Compile" ⇒ "Structured Ladder/ST" ⇒ "Compile Condition1".**

2. **Check the "Use lower-case device names as labels" check box.**
   - If this check box is not checked, an error occurs when compiling programs.
   - If this check box is checked, a warning is displayed when compiling programs. (Label names that contain a same lower case as the device can be used.)



7.6.5 Connecting PLS and/or PLF outputs directly to other inputs with lines

This section explains the method for connecting outputs of the selected function(s) (PLS and/or PLF) directly to other inputs with lines.

Operating procedure

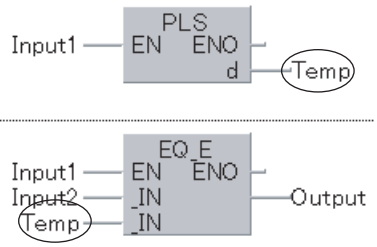
1. **Select [Tools] ⇒ [Options] ⇒ "Compile" ⇒ "Structured Ladder/ST" ⇒ "Compile Condition1".**

2. **In "Function Output Setting", check the function(s) (PLS and/or PLF) to be set.**
   - If any of these check boxes are not checked, an error occurs when compiling programs.
   - If any of these check boxes are checked, errors and warnings are not displayed when compiling programs.

Example) Creating a program using the PLS output

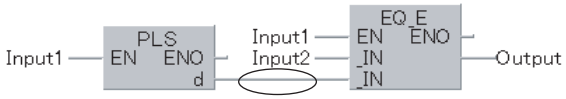
<Check box is not checked>

The output is required to be retained at the label or device.



<Check box is checked>

The output can be connected directly to another input with a line.



7.6.6 Assigning individual devices to EN/ENO function outputs

This section explains the method for assigning individual devices to each function, when system devices are automatically assigned to EN/ENO function outputs while editing in the structured ladder editor.

Operating procedure

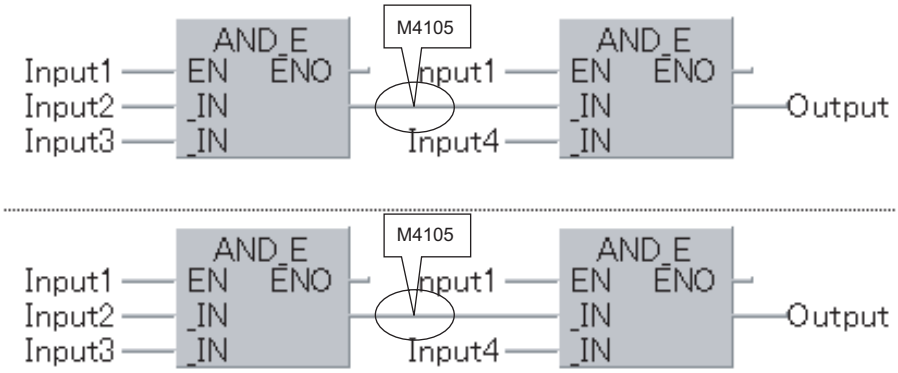
1. **Select [Tools] ⇒ [Options] ⇒ "Compile" ⇒ "Structured Ladder/ST" ⇒ "Compile Condition2".**

2. **Check the "Assign each system device to output of functions that maintains bit type" check box.**
   - If the check box is not checked, the same device is assigned to the outputs.
   - If the check box is checked, the individual devices are assigned to each function.

Example) Devices automatically assigned to EN/ENO function outputs

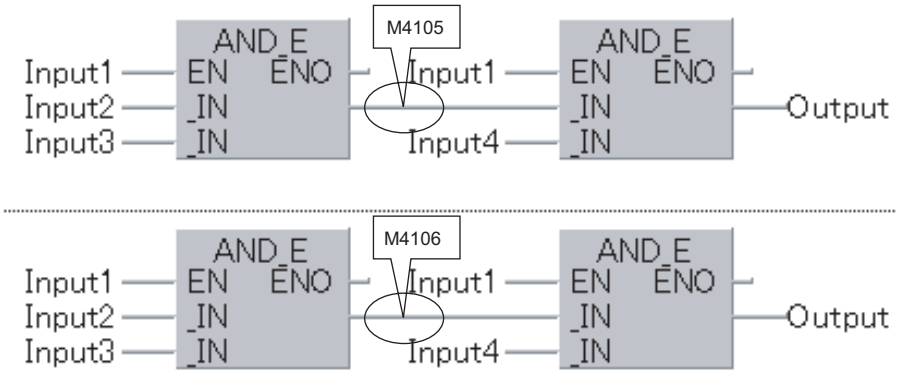
<Check box is not checked>

The same device is assigned.



<Check box is checked>

The individual devices are assigned.



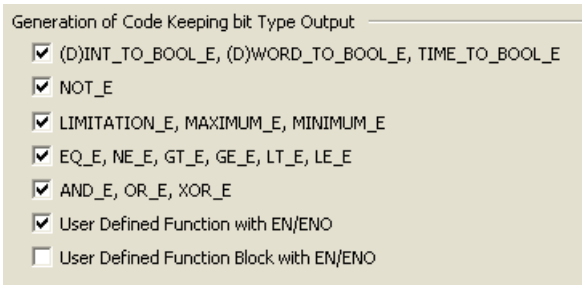
7.6.7 Retaining bit type outputs of functions

This section explains the method for retaining bit type outputs of the target function(s).

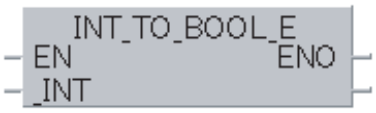
Operating procedure

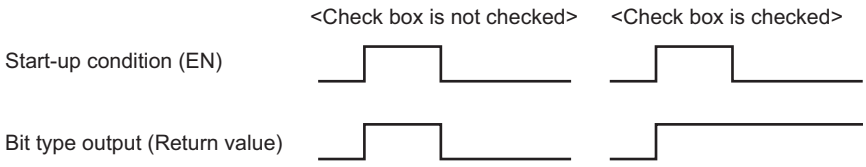
1. **Select [Tools] ⇒ [Options] ⇒ "Compile" ⇒ "Structured Ladder/ST" ⇒ "Compile Condition3".**

2. **Check the function(s) to be set.**
   - If any of these check boxes are not checked, bit type outputs of function(s) are not retained. (An operation equivalent to the OUT function).
   - If the check box(es) is checked, bit type outputs of the function(s) are retained. (An operation equivalent to the SET function).



Example) Retaining the INT_TO_BOOL_E output





MEMO

8 WRITING/READING DATA TO/ FROM PROGRAMMABLE CONTROLLER CPU

This chapter explains the method for writing/reading sequence programs to/from the programmable controller CPU or memory card.

For the overview of the data write/read operation, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

8.1 Writing/Reading Data to/from Programmable Controller CPU 8-2

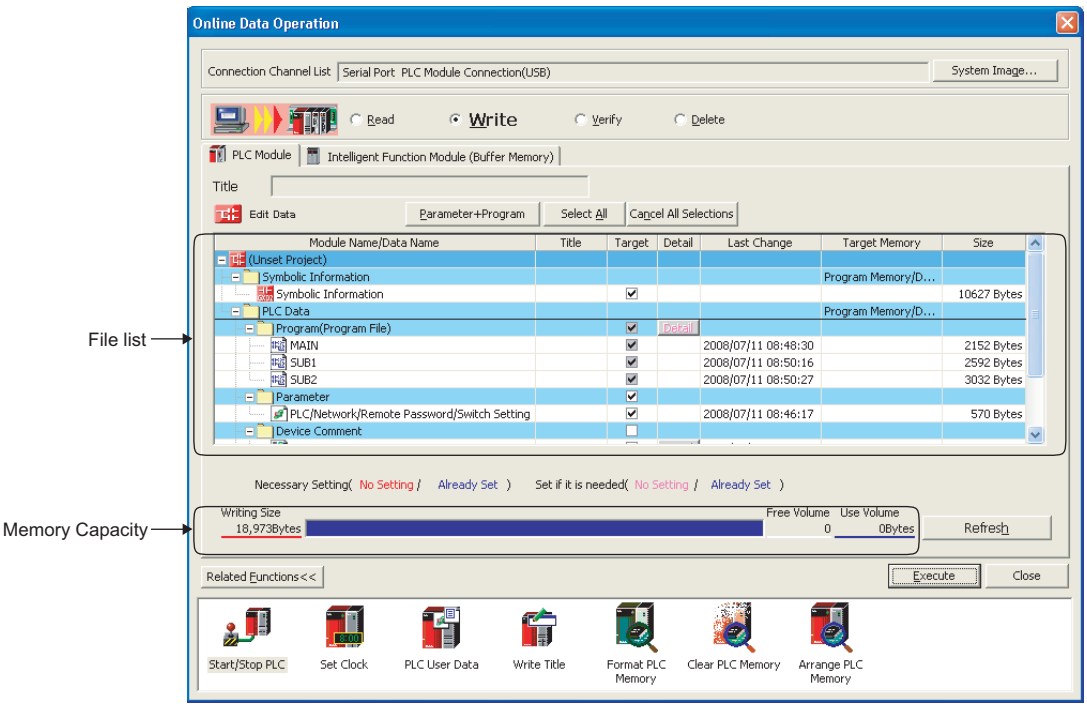
8.1 Writing/Reading Data to/from Programmable Controller CPU

QCPU

This section explains the method for writing the Structured project data to the programmable controller CPU or memory card, and the method for reading the data from the programmable controller CPU or memory card to the project.

Screen display

Select [Online] ⇒ [Write to PLC] ( )/[Read from PLC]( ).



Operating procedure

1. Set the items on the screen.

Item		Description
Connection Channel List		Displays the information on the set connection target.
Title		Displays the title of the target memory by clicking the Refresh button.
File list		-
	Target	Select the data to be written/read.
	Target Memory	Select the memory from the list ( ) by clicking the cell under “Target Memory”. For details of the memory card application, refer to the following manual. ☞ QCPU User's Manual (Hardware Design, Maintenance and Inspection)
Memory capacity		-
	Writing Size	Displays the total size of written data checked in “Target”.
	Free Volume	Displays the free space volume of the target memory.
	Use Volume	Displays the used space volume of the target memory.

When a program (program file), device comment or device memory is selected, the range can be set by clicking the Detail / Detail button.
When reading data from the programmable controller CPU, and the symbolic information or device memory is selected, the detailed settings are required.
For details, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

2. Click the Execute button.

When the Write to PLC function is executed, the specified data are written to the target memory.
When the Read from PLC function is executed, the specified data are read from the target memory.

Screen button

- System Image...

  Displays the illustration of the connection channel list.

- Parameter + Program

  Selects the parameters and all programs displayed in the list.

- Select all

  Selects all data displayed in the list.

- Cancel all selections

  Cancels the selection status of all data displayed in the list.

- Related functions>> / Related functions<<

  Switches display/hide of the Related functions button.
  For details of the related functions, refer to the following manual.
  ☞ GX Works2 Version1 Operating Manual (Common)

- Get Project Name of Source Information (Read from PLC, Delete PLC Data only)

  Displays the project name of the symbolic information in the Title/Project Name column.

- Refresh

  Updates data list, writing size, free space volume, and used space volume on the Online Data Operation screen.
  When multiple personal computers are connected to the programmable controller CPU, update the target memory before reading data from the programmable controller CPU.

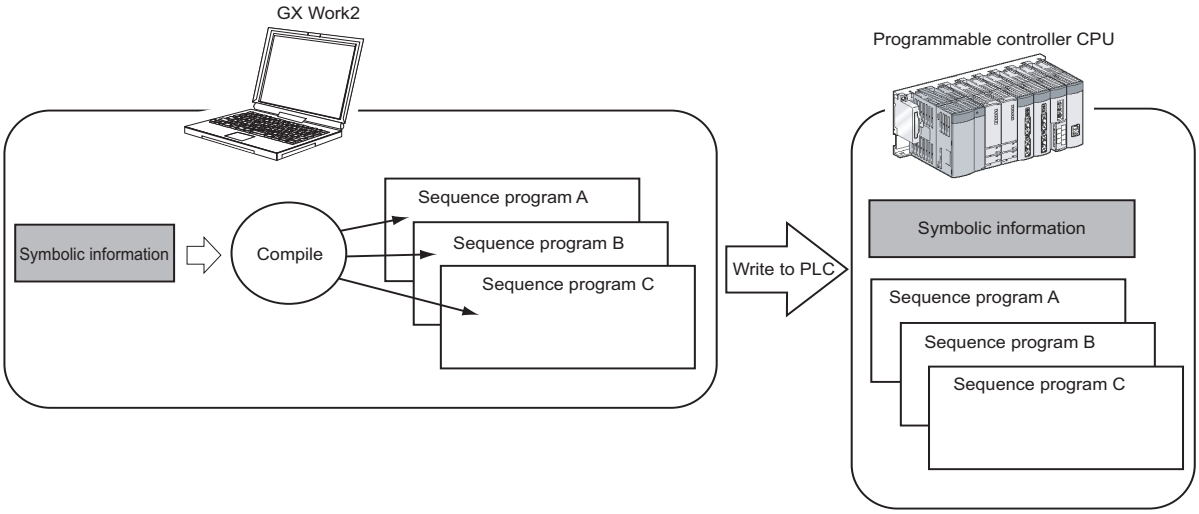
■ Symbolic information

The symbolic information is data that store the program configuration such as structures and labels. To restore these data included in the symbolic information when reading a program from the programmable controller CPU, write/read the symbolic information to/from the programmable controller CPU.
Data such as structures and labels included in the symbolic information cannot be restored if only sequence programs are read. (Data are treated as the ladder language in Structured projects.)

The following table shows the program data included in the symbolic information.

Table 8.1-1Data included in the symbolic information

Item	Included data	
Symbolic information	User libraries	
	Structures	
	Global labels	
	Program_File_Pool	
		Tasks
	POU_Pool	
		Local labels
		Programs
		Function blocks
		Functions



Point

- **Compile status when the symbolic information is read**
  - When the symbolic information and parameters are read simultaneously, and the data in the symbolic information match with parameters and programs (program files) in the programmable controller CPU, the read data are in compiled status. When only the symbolic information is read, the read data are in uncompiled status.
  - When the symbolic information of GX Developer or GX IEC Developer is read, the read data are in uncompiled status. Compile the program again after executing the Read from PLC function.
- **Precautions for reading symbolic information**
  For the precautions for reading label programs (symbolic information) of the existing application using GX Works2, or reading label programs (symbolic information) of GX Works2 using the existing application, refer to the following manual.
  (☞ GX Works2 Version1 Operating Manual (Common))

9 MONITORING

This chapter explains the method for monitoring the program execution status of the programmable controller CPU using the program editor.
For the overview of monitoring, refer to the following manual.
☞GX Works2 Version1 Operating Manual (Common)

9.1 Starting and Stopping Monitoring of Program 9-2
9.2 Starting and Stopping Monitoring of Function Blocks 9-3
9.3 Changing Operating Conditions of Monitoring 9-4
9.4 Monitoring Programs in ST Editor 9-6
9.5 Monitoring Programs in Structured Ladder Editor 9-7

9.1 Starting and Stopping Monitoring of Program

QCPU
Q

This section explains the method for monitoring a program in the program editor.
Open the program editor to be monitored in advance.

■ Starting monitoring of programs

The following explains the method for starting monitoring of the program.

Operating procedure

- Select [Online] ⇒ [Monitor] ⇒ [Start Monitoring] ( ).
  The monitoring starts.

■ Stopping monitoring of programs

The following explains the method for stopping monitoring of the program.

Operating procedure

- Select [Online] ⇒ [Monitor] ⇒ [Stop Monitoring] ( ).
  The monitoring stops.

Point

● **Changing current values**
During monitoring, bit devices can be forced ON/OFF, and the current values of devices, buffer memories, and labels can be changed.
For forcing bit devices ON/OFF and changing current values, refer to the following manual.
☞ GX Works2 Version1 Operating Manual (Common)

9.2 Starting and Stopping Monitoring of Function Blocks

QCPU Q

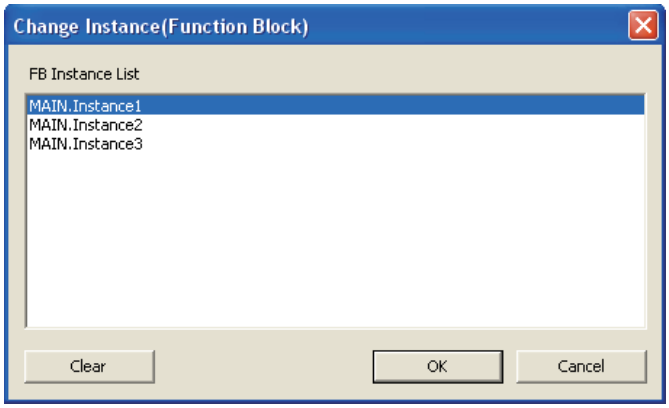
This section explains the method for monitoring function block programs.
Open the function block program to be monitored in advance.

■ Starting monitoring of function blocks

The following explains the method for starting monitoring of the function block.

Operating procedure

1. **Select [Online] ⇒ [Monitor] ⇒ [Change Instance (Function Block)].**
   The Change Instance (Function Block) screen is displayed.



2. **Select the FB instance to be monitored.**
3. **Click the OK button.**
4. **Select [Online] ⇒ [Monitor] ⇒ [Start Monitoring] ( ).**
   The monitoring starts.
   The display and executable operations during monitoring are the same as those of the program monitoring.

Screen button

● Clear
Cancels the selected status of the FB instance and stops monitoring.

■ Stopping monitoring of function blocks

The following explains the method for stopping monitoring of the function block.

Operating procedure

• Select [Online] ⇒ [Monitor] ⇒ [Stop Monitoring] ( ).
The monitoring stops.

9.3 Changing Operating Conditions of Monitoring

QCPU
Q

This section explains the method for changing the operating conditions of monitoring.

9.3.1 Changing current value display format (decimal/ hexadecimal) of word type variable

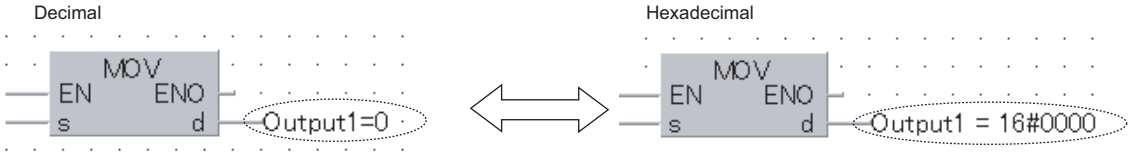
This section explains the method for changing the current value display format of word and double word type variable displayed during monitoring.

■ Changing display format during monitoring

The following explains the method for changing the current value display format of word and double word type variable during monitoring.

Operating procedure

- Select [Online] ⇒ [Monitor] ⇒ [Change Value Format (Decimal)] / [Change Value Format (Hexadecimal)].

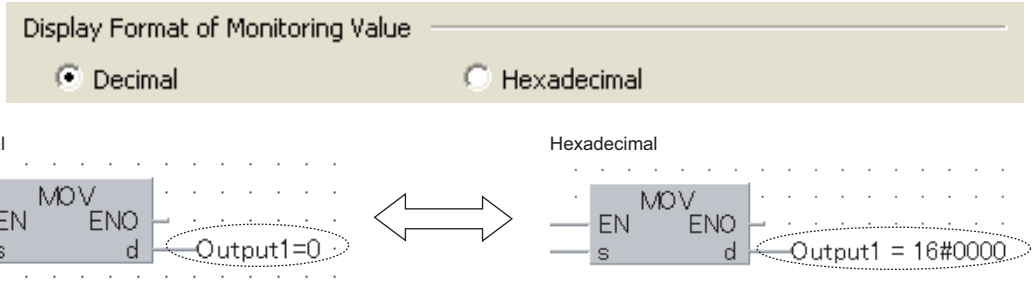


■ Changing display format before starting monitoring

The following explains the method for changing the current value display format by setting the option. The monitoring starts with the set display format.

Operating procedure

- Select [Tools] ⇒ [Options] ⇒ "Monitor" ⇒ "Structured Ladder/ST" ⇒ "Display Format of Monitoring Value" ⇒ "Decimal"/"Hexadecimal".



9.3.2 Setting the number of displaying characters for character strings

The following explains the method for setting the number of displaying characters for the monitoring result of character strings by setting the option in the structured ladder and ST editors.

Operating procedure

1. **Select [Tools] ⇒ [Options] ⇒ "Monitor" ⇒ "Structured Ladder/ST" ⇒ "Character String Monitor Setting".**

2. **Set the number of displaying characters within the range from 2 to 50.**

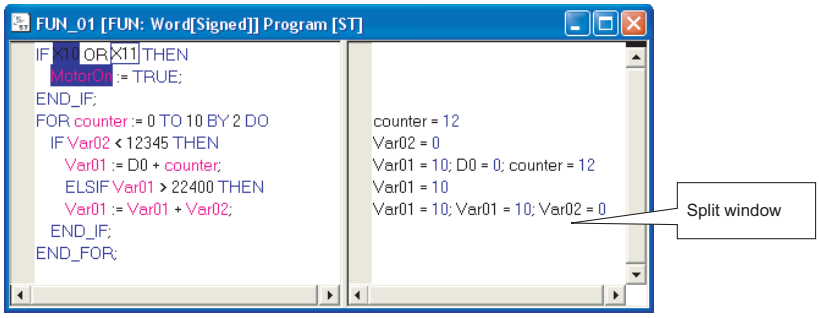
9.4 Monitoring Programs in ST Editor

This section explains the display of the ST editor at the monitoring.

Screen display

Select [Online] ⇒ [Monitor] ⇒ [Start Monitoring].

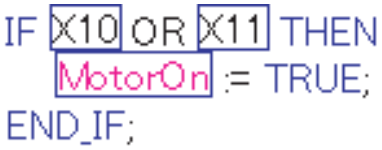
Select [Online] ⇒ [Monitor] ⇒ [Split Window].



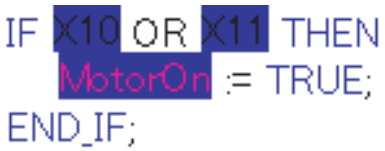
■ Monitoring bit type labels and bit devices

The following shows the display of bit type labels and bit devices at the monitoring.

<FALSE>

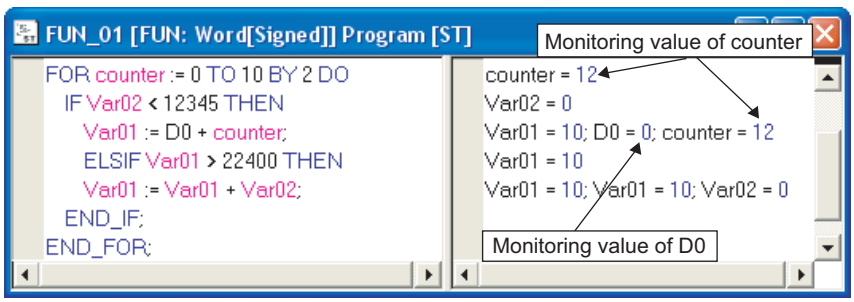


<TRUE>



■ Monitoring labels other than bit type labels and word devices

The monitoring values of labels other than bit type labels and word devices are displayed on the right side of the split window.

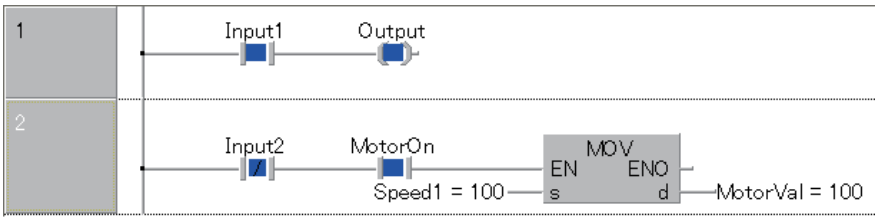


9.5 Monitoring Programs in Structured Ladder Editor

This section explains the display of the structured ladder editor at the monitoring.

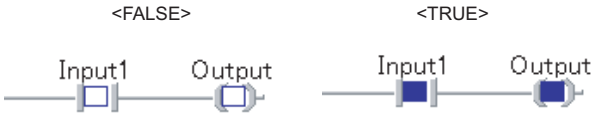
Screen display

Select [Online] ⇒ [Monitor] ⇒ [Start Monitoring].



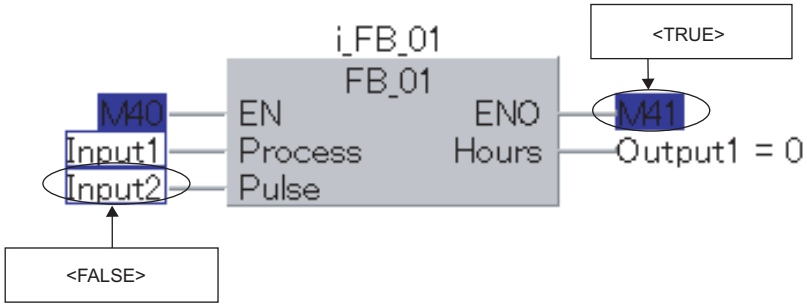
■ Monitoring contacts and coils

The following shows the display of contacts and coils at the monitoring.



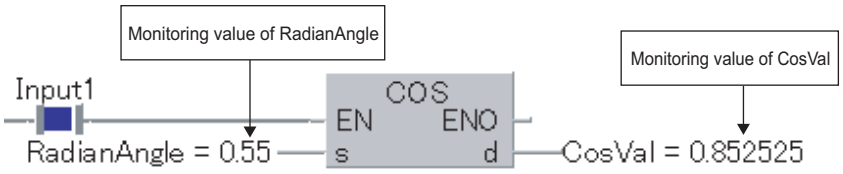
■ Monitoring bit type labels and bit devices

The following shows the display of bit type labels and bit devices at the monitoring.



■ Monitoring labels other than bit type labels and word devices

The monitoring values of labels other than bit type labels and word devices are displayed to the right of the corresponding labels or devices.



Point

● **Monitoring values of array**
The monitoring values are not displayed when labels are used for array elements.

10 USING LIBRARIES

In Structured project, the libraries are shared with multiple projects as program assets.
This chapter explains the types of library and their operating methods.

10.1 Types of Library 10-2
10.2 Procedure for Using User Libraries 10-4
10.3 Operating User Libraries 10-5

10.1 Types of Library

QCPU
Q

This section explains the library, a format that can be used commonly in multiple projects. Library is data consist of component-based programs.

The followings are the three types of library.

- Sequence instructions, basic instructions, application instructions, PID control instructions, module dedicated instructions

  Instructions used in the ST/structured ladder editor to perform the same functions as instructions in the ladder editor. Functions such as common processes for the intelligent function modules are used as components.

  ☞ QCPU Structured Programming Manual (Common Instructions)
  QCPU Structured Programming Manual (Special instructions)

- Application functions

  Functions compliant with the IEC61131-3 standards.
  Functions such as standard arithmetic processes independent of user programs are used as components.

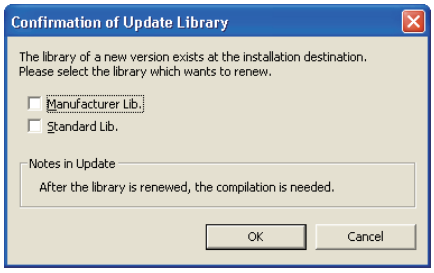
  ☞ QCPU Structured Programming Manual (Application Functions)

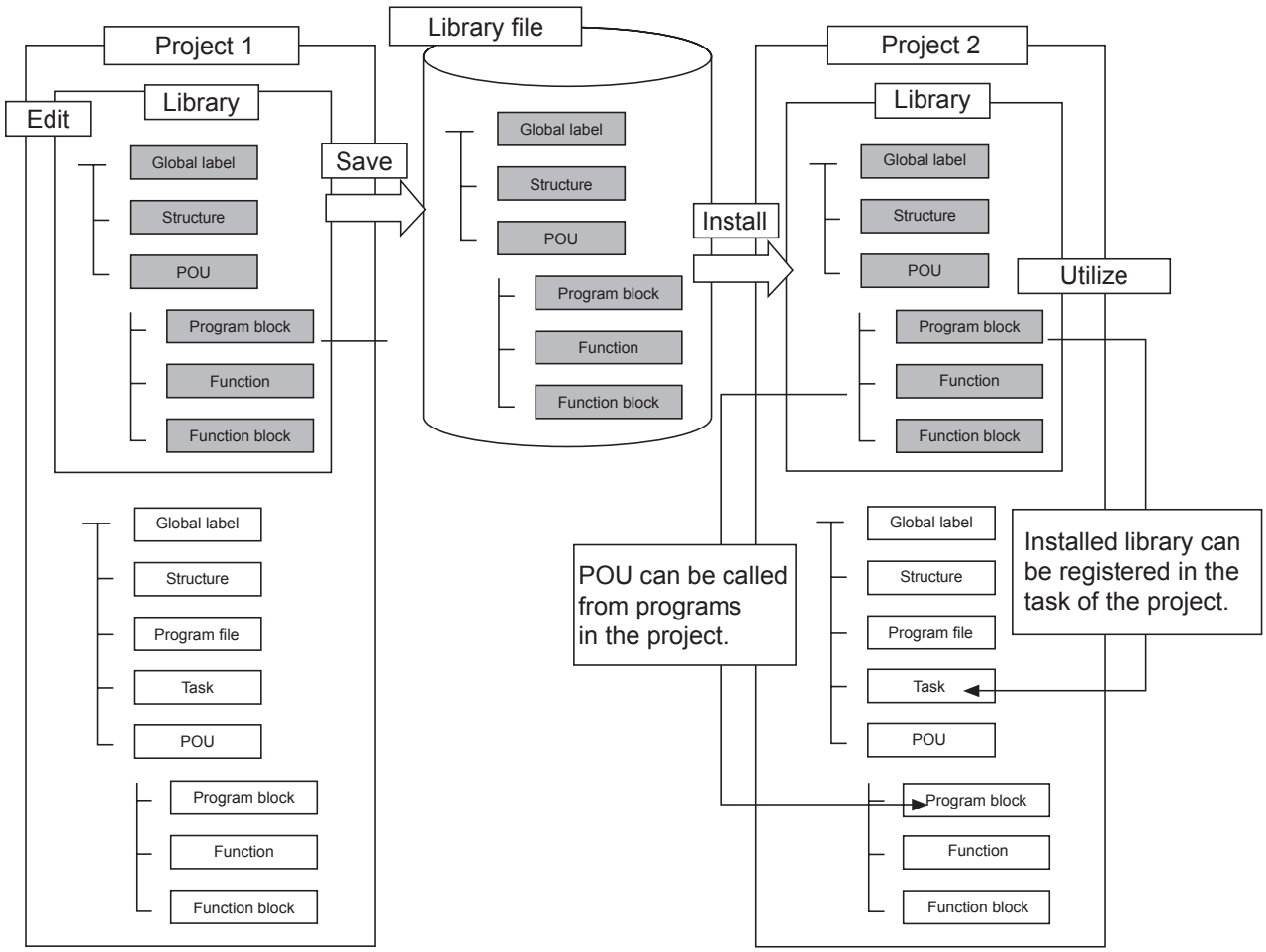
- User library

  A library created by user. User libraries created for a certain project can be shared with other projects by saving libraries in library files.
  The following data can be registered in a user library.
  - Global labels
  - Structures
  - POUs (program blocks, functions, and function blocks)

Point

- **Opening a project whose libraries of instructions and/or application functions are old.**
  When a project is saved, the libraries of instructions and/or application functions are also saved in the project. Therefore, if the Structured project is opened in new version of GX Works2, the Confirmation of Update Library screen may be displayed.
  To update the libraries in the project, select the corresponding check box(es).




Project 1
Edit
Library
Global label
Structure
POU
Program block
Function
Function block
Global label
Structure
Program file
Task
POU
Program block
Function
Function block
Save
Library file
Global label
Structure
POU
Program block
Function
Function block
Install
Project 2
Library
Global label
Structure
POU
Program block
Function
Function block
Utilize
Global label
Structure
Program file
Task
POU
Program block
Function
Function block
POU can be called from programs in the project.
Installed library can be registered in the task of the project.

10.2 Procedure for Using User Libraries

QCPU
Q

This section explains the method for utilizing the created user libraries to other projects.

1. Create a library.

Procedure	Reference
Create a library. • Select Library_Pool folder in the Project window, then select [Project] ⇒ [Library] ⇒ [Create].	Section 10.3.1
Register data such as elements of program configuration to the created library. • Select [Project] ⇒ [Object] ⇒ [New].	GX Works2 Version1 Operating Manual (Common)
Edit the data such as elements of program configuration registered to the library.	Chapter 5, Chapter 6
Set a password to the library created in the project. • Select [Project] ⇒ [Library] ⇒ [Change Password].	Section 10.3.11
Set a help file to the library. • Select a library data in the Project window, then select [Project] ⇒ [Object] ⇒ [Property].	Section 10.3.12

2. Utilize the library.

Procedure	Reference
Save the library data in a library file. • Select [Project] ⇒ [Library] ⇒ [Save as].	Section 10.3.4
Install the library file data to another project. • Select [Project] ⇒ [Library] ⇒ [Install].	Section 10.3.6

3. Update the library data.

Procedure	Reference
Enter the password to enable editing data in the library. • Select [Project] ⇒ [Library] ⇒ [Open].	Section 10.3.2
After editing the data in the library, save the library file. • Select [Project] ⇒ [Library] ⇒ [Save].	Section 10.3.5
Update the library data in another project that shares the data. • Select [Project] ⇒ [Library] ⇒ [Update Library].	Section 10.3.7

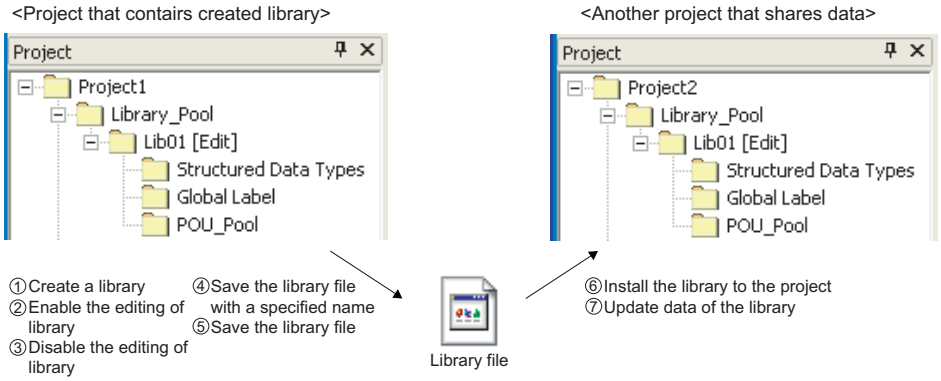
10.3 Operating User Libraries

QCPU
Q

This section shows the functions related to user library operations.

Table 10.3-1Library operating functions

Name	Description	Reference
Create	Creates a new library.	10.3.1
Open	Enables/disables editing of a library.	10.3.2
Close		10.3.3
Save as	Names and saves a project.	10.3.4
Save	Saves a library file.	10.3.5
Install	Installs a created library to the project.	10.3.6
Update Library	Updates the library imported to the project.	10.3.7
Deinstall	Deletes a library from the project.	10.3.8
Delete	Deletes a library file.	10.3.9
Rename	Renames a library.	10.3.10
Change Password	Sets a password for a library.	10.3.11
Help	Displays help information of the library.	10.3.12

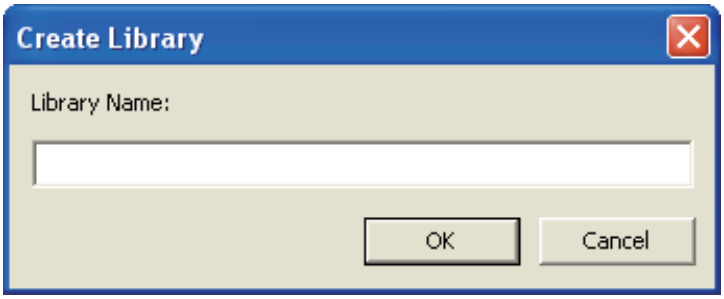


10.3.1 Creating libraries

This section explains the method for creating a user library.
Select the Library_Pool folder and follow the operation below.

Screen display

Select [Project] ⇒ [Library] ⇒ [Create].



Operating procedure

1. **Set a library name.**

2. **Click the** OK **button.**
   A library is created and displayed in the Project window.

Point

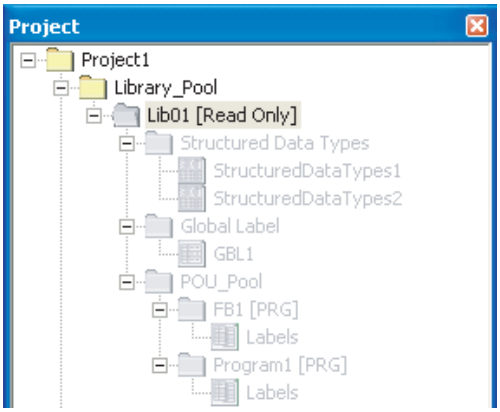
- **Maximum number of user libraries**
  - The maximum number of user libraries that can be created in a project is 800.
- **Number of characters for a library name**
  - The maximum number of characters that can be entered for a library name is 32.
- **Invalid characters for library names**
  - For invalid characters for library names, refer to the following manual.
    ☞ QCPU Structured Programming Manual (Fundamentals)

10.3.2 Enabling editing of library data

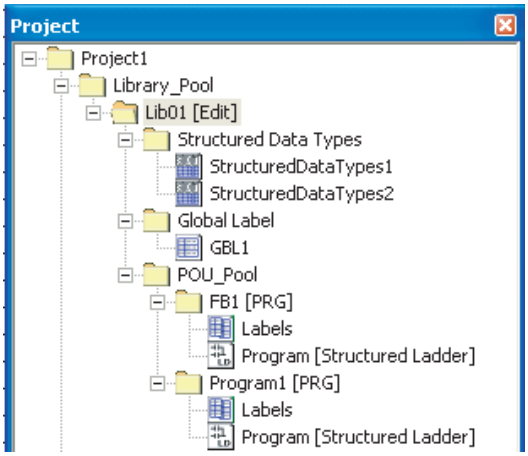
This section explains the method for enabling the editing of user library data.
To edit user library data, the library needs to be set in the editing status.

Screen display

<Read-only status>



<Editing status>



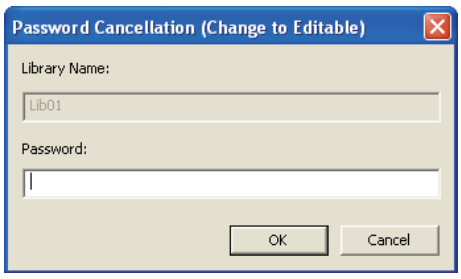
Operating procedure

1. **Select a library to edit in the Project window. (Multiple libraries can be selected.)**

2. **Select [Project] ⇒ [Library] ⇒ [Open].**
   The editing of the library data is enabled.

● Library password

If the library is protected with a password, the password needs to be entered to open the library for editing.
When the password entry screen is displayed, enter the library password.



Point

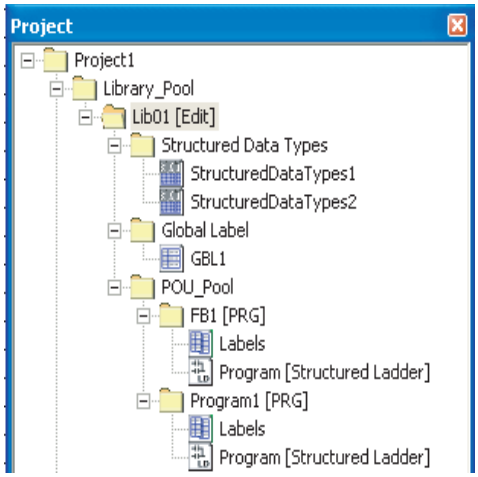
● **Editing status when opening a project**
When a project is opened, all libraries are in the read-only status.
(Even if a project is closed with the editing status, it is set in the read-only status when reopening the project.)

10.3.3 Disabling editing of library data

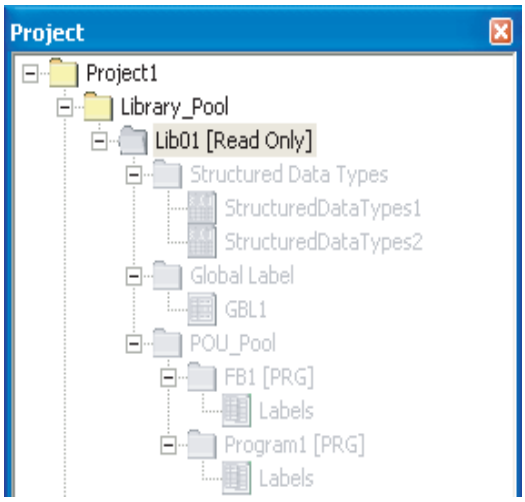
This section explains the method for disabling the editing of user library data.

Screen display

<Editing status>



<Read-only status>



Operating procedure

1. **Select a library to edit in the Project window. (Multiple libraries can be selected.)**

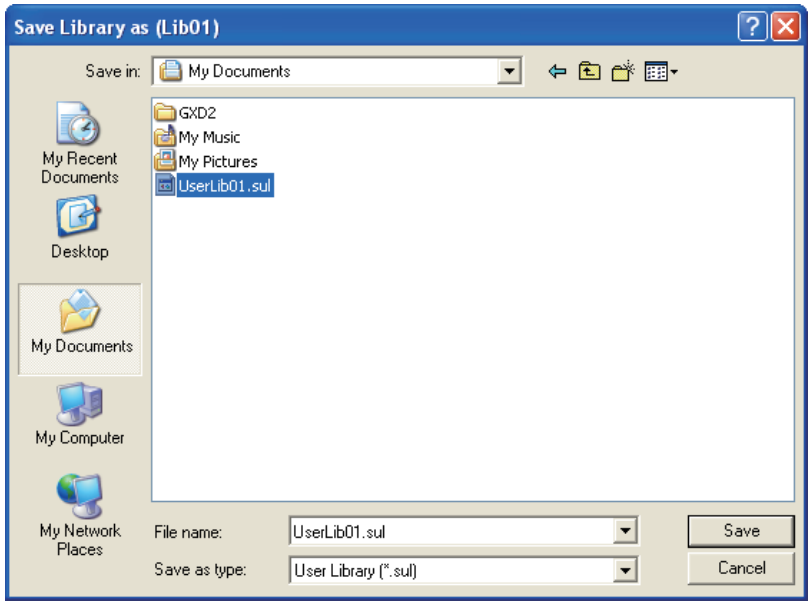
2. **Select [Project] ⇒ [Library] ⇒ [Close].**
   The editing of the library data is disabled.

10.3.4 Saving libraries to library files under specified name

This section explains the method for saving a user library to the library file.
A library file can be also saved with a different library name.

Screen display

Select [Project] ⇒ [Library] ⇒ [Save as].



Operating procedure

1. **Select a library name to be saved in the Project window. (Multiple libraries can be selected.)**

2. **Select [Project] ⇒ [Library] ⇒ [Save as].**
   The Save Library as screen is displayed.

3. **Set a file name.**

4. **Click the** Save **button.**
   The library data is saved to the library file.

10.3.5 Saving library files

This section explains the method for saving a user library in the library file.

Operating procedure

1. **Select a library name to be saved in the Project window. (Multiple libraries can be selected.)**

2. **Select [Project] ⇒ [Library] ⇒ [Save].**
   The library data is saved in the library file.

Point

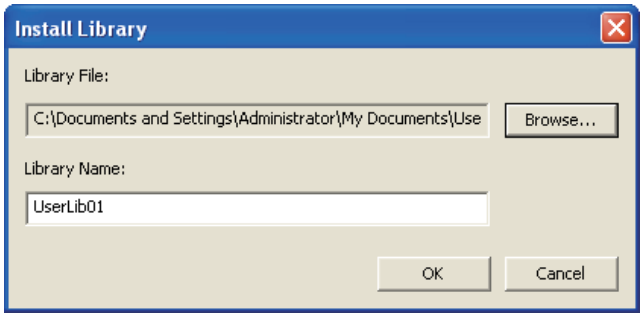
- **Identifying the saved library file**
  A saved library file is identified with the path and file name which are specified when the file was saved with a specified name previously or installed to the project.
- **Applying file changes to other projects**
  To apply the changes to library files in other projects, the libraries need to be updated in each project.
  (☞ Section 10.3.7)

10.3.6 Installing library data to project

This section explains the method for installing library data from the existing library file to the project.

Screen display

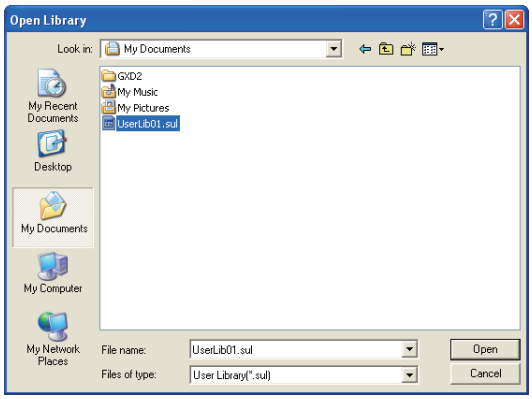
Select [Project] ⇒ [Library] ⇒ [Install].



Operating procedure

1. **Click the [Browse...] button.**

   The Open Library screen is displayed.



2. **Select a file to be installed and click the [Open] button.**

   The path and name of the selected library file are displayed on the Install Library screen.

3. **Set a library name.**

   A name different from the library file can be set.

4. **Click the [OK] button.**

   The selected library file data are installed and the library name is displayed in the Project window.

Point

- **Maximum number of user libraries**
  The maximum number of user libraries that can be created in a project is 800.

10.3.7 Updating libraries

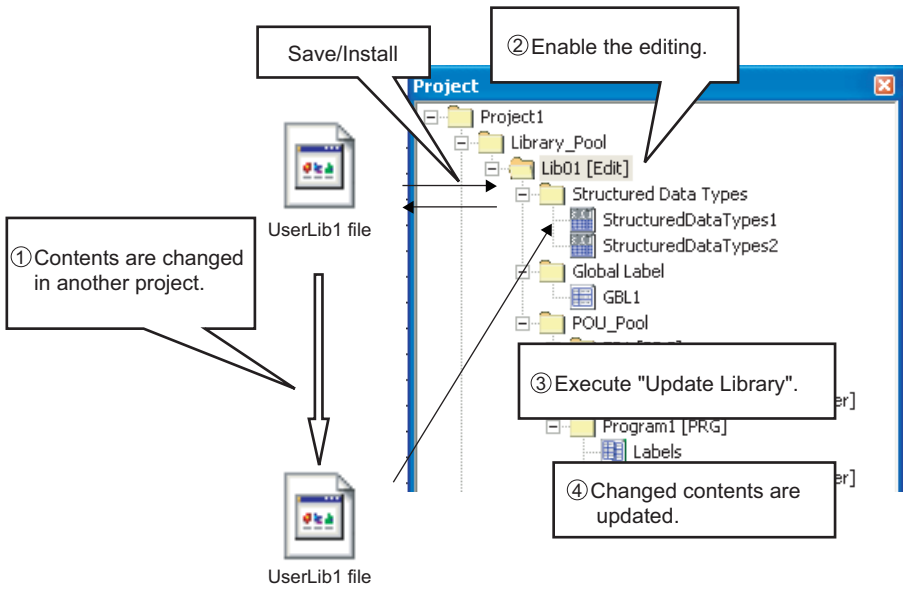
This section explains the method for rereading data in a library file.

Operating procedure

1. **Select a library name to be updated in the Project window. (Multiple libraries can be selected.)**

2. **Select [Project] ⇒ [Library] ⇒ [Update Library].**
   The data in the library file is read and the display contents in the Project window are refreshed.

To apply the library file changes made in another project (① in the figure below) to the current project, the library needs to be updated (③ and ④ in the figure below).



Point

- **Compilation status of POUs**
  When a library is updated, all POUs using the updated library become in the uncompiled status.
- **Editing status**
  - Before using this function, enable the editing of the library data. (☞ Section 10.3.2)
  - When this function is completed, the editing of the library data is disabled.

10.3.8 Deleting libraries from project

This section explains the method for deleting a user library from the project.

Screen display

Operating procedure

1. **Select a library name to be deleted in the Project window. (Multiple libraries can be selected.)**

2. **Select [Project] ⇒ [Library] ⇒ [Deinstall].**
   The library is deleted from the Project window.

Point

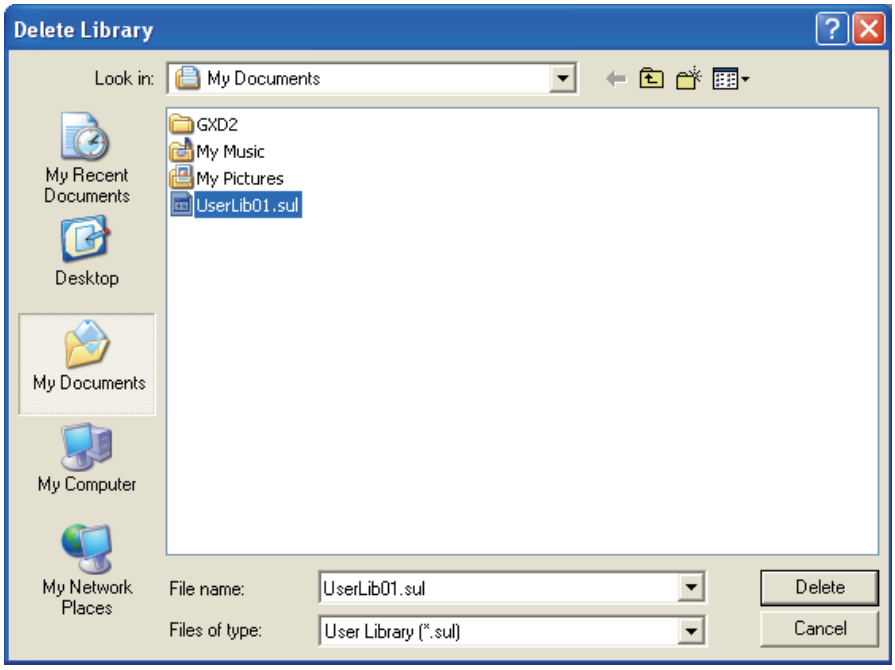
- **Library files**
  This function deletes a library from the project, but does not delete the library file that contains the deleted library data. (Deleting library files ☞ Section 10.3.9)
- **Compilation status of POUs**
  When a library is deleted from the project, all POUs using the deleted library become in the uncompiled status.

10.3.9 Deleting library files

This section explains the method for deleting a library file.

Screen display

Select [Project] ⇒ [Library] ⇒ [Delete].



Operating procedure

1. **Select a file to be deleted on the screen shown above.**

2. **Click the** [Delete] **button.**
   The library file is deleted.

Point

● **Deleting library files**
A library file can be deleted regardless of the library password setting and library editing status.
(A library file can be deleted even when the library data are being edited in the project).
Library files must be managed carefully if they are shared with multiple projects.

10.3.10 Renaming libraries

This section explains the method for renaming a user library.

Operating procedure

1. **Select a library name to be renamed in the Project window.**

2. **Select [Project] ⇒ [Library] ⇒ [Rename].**
   The editing of the library name in the project tree is enabled.

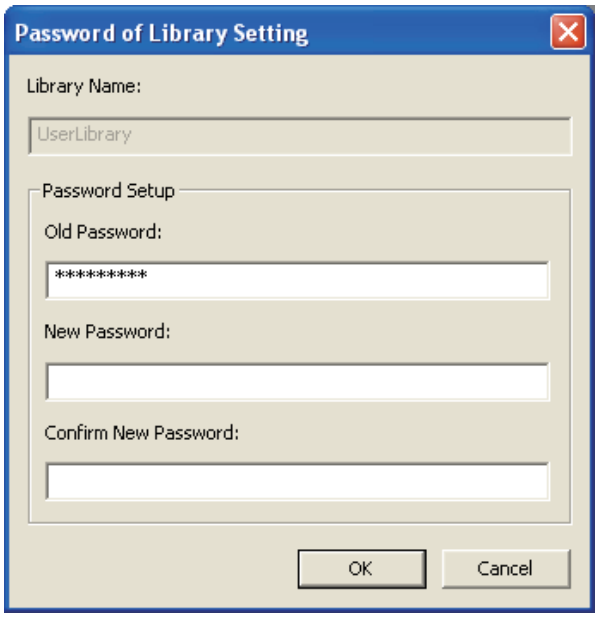
3. **Set the library name.**
   Press the Enter key to confirm the new library name.

10.3.11 Setting library passwords

This section explains the method for setting a password for the library.
Data in the user library can be protected by setting a password. Once the password is set, programs in the POUs are hidden on the screen (except for structures, global labels, and local labels).

Screen display

Select [Project] ⇒ [Library] ⇒ [Change Password].



Operating procedure

1. **Enter the old library password.**

   Enter the current library password.
   If this is the first time for a library password to be set, leave this field blank.

2. **Enter a new library password.**

   Enter the desired library password.
   The password must be a combination of alphanumeric characters and symbols corresponding to ASCII codes from 20_H to $7E_H$ within 14 characters.
   For details of ASCII codes, refer to the following manual.
   ☞ GX Works2 Version1 Operating Manual (Common)

3. **Confirm the new library password.**

   Enter the new library password again.

Point

- **Libraries saved in library files**
  If a library is saved in the library file, the library password setting is also saved.
  Library files must be managed carefully if they are shared with multiple projects.
- **Editing status**
  Before using this function, enable the editing of the library. (☞ Section 10.3.2)

10.3.12 Displaying library help

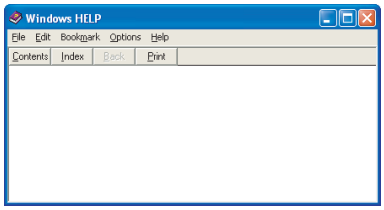
This section explains the method for displaying a help for the library.
Specified files created in the following formats can be set as help data.

- Windows help file (*.hlp)
- HTML help file (*chm)

Screen display

Select [Project] ⇒ [Library] ⇒ [Help].

Windows® help file (*.hlp)



HTML help file (*.chm)

Operating procedure

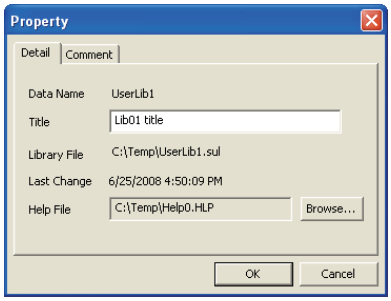
1. **Select a library name to be displayed as a library help in the Project window.**

2. **Select [Project] ⇒ [Library] ⇒ [Help].**
   A library help screen is displayed.

■ Setting help files

The following explains the method for setting the help data to be displayed on the Property screen of each user library.

Screen display



Select [Project] ⇒ [Object] ⇒ [Property].

Operating procedure

1. **Click the** Browse... **button.**
   The Open a help file screen is displayed.

2. **Select a file to be set.**
   The path and name of the selected help file are displayed on the Property screen.

3. **Click the** OK **button when the setting is completed.**
   The specified help file is set as the user library help.

MEMO

11 SETTING OPTIONS

This chapter explains the functions of options for setting screen display format or detailed operations of each function.

11.1 Basic Operations 11-2
11.2 Option Setting List 11-3

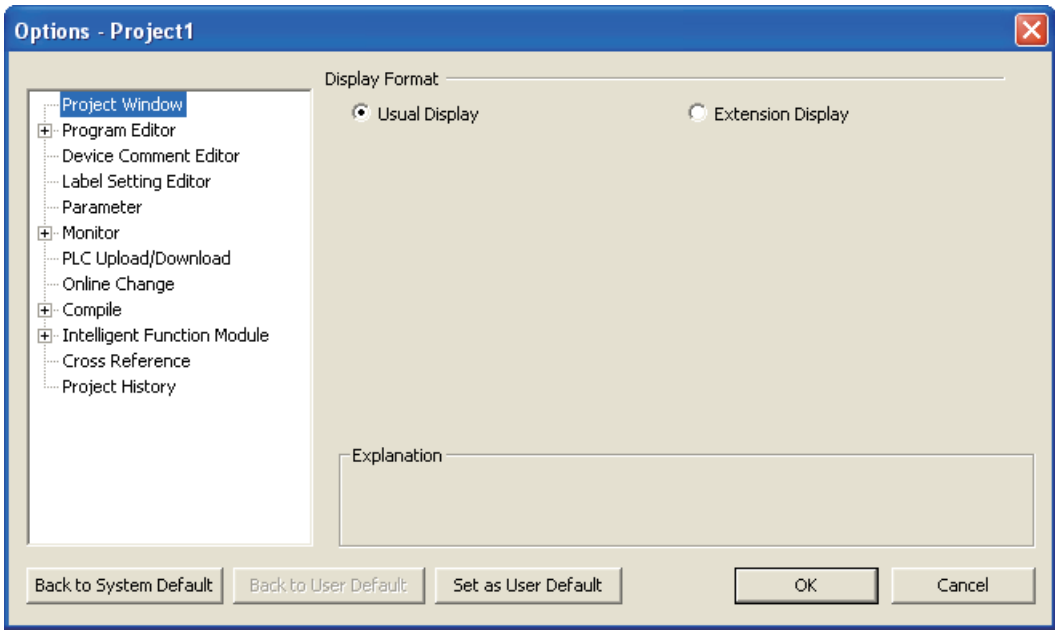
11.1 Basic Operations

QCPU
Q

This section explains the method for setting options.

Screen display

Select [Tools] ⇒ [Options].



Operating procedure

- **Set the items on the screen.**

  When the cursor is placed on a setting item, an explanation of the item is displayed in the "Explanation" field.

  Details of the setting items ☞ Section 11.2

Screen button

● Back to System Default

Restores the initial settings.

● Back to User Default

Restores the user default settings.

● Set as User Default

Stores the current settings as the user default and applies them to a new project.

11.2 Option Setting List

QCPU Q

The following table shows the option setting items.

Table 11.2-1Option setting items (1/2)

Option tree item			Item	Description	Reference
Project Window			☞ GX Works2 Version1 Operating Manual (Common)		
Program Editor	Structured Ladder/ST	Tool Hint	Tool Hint Display Items	Select items displayed on tool hint.	Section 6.2.10
			Tool Hint Display Format	Select display lines on tool hint.	
	Structured Ladder	Label	Display label name/ comment of contact or coil in multiline	Set display lines and characters per line.	Section 6.4.10
			Declare new label name	Select display/non-display assignment dialog when declaring new label names.	Section 6.4.9
		FB/FUN	Wrap instance name for function block	Select whether to wrap function block instance name at the function block width.	Section 6.4.10
			Specify the number of enable characters for label name/comment	Specify the number of display characters of function or function block label.	
			Automatic input/output variables	Select whether to add input/output variable when function block or function is pasted.	Section 6.4.8
			Automatic ENO variables	Select whether to add ENO output variable when function block or function is pasted.	
			Pin overwrites	Select whether to overwrite I/O variable with a grid line when ruling a grid line over function or function block.	Section 6.4.4
			Double clicking opens header	Select whether a label editor is displayed when function or function block is double clicked.	Section 6.2.8
			Double clicking opens body	Select whether a program editor is displayed when function or function block is double clicked.	
		Guided	Wrap Ladder	Select whether to wrap ladder. The edited ladder shall be object after changing the setting.	Section 6.4.12
			Set guided mode as default editing mode	Set the default at opening structured ladder editor as guided mode.	
			Allow hotkey repeater	Select whether to display grid line ruling dialog at drawing lines.	
			Enter variable names after contacts and coils	Select whether to input label names or devices after entering contacts and coils.	
			Auto Comment Block Width	Set comment length added in network with grid numbers.	
	ST		Auto indention	Select whether to use the automatic indent when entering ST control syntax, such as IF or FOR, and starting a new line.	Section 6.3.2
			Label Name Prediction	Select whether to display label name by List starting with the input character at entering label name.	Section 6.3.1
			Tabulator Length	Set the number of characters at entering tabulator.	Section 6.3.3
	Ladder/SFC		☞ GX Works2 Version1 Operating Manual (Common)		
	Ladder		☞ GX Works2 Version1 Operating Manual (Simple Project)		
	SFC				

Table 11.2-1 Option setting items (2/2)

Option tree item			Item	Description	Reference
Device Comment Editor			☞ GX Works2 Version1 Operating Manual (Common)		
Label Setting Editor			Automatic copy and increment when inserting a row.	Select whether to copy the texts in the upper row after incrementing it when inserting a row.	Section 5.5.2
			Copy data type/comment items	Select if the data type, comment, and remark shall be object for auto copy.	
			Default Length of String Data Type	Set the default string length for string data type.	Section 5.5.1
Parameter			☞ GX Works2 Version1 Operating Manual (Common)		
Monitor	Structured Ladder/ST		Display Format of Monitoring Value	Select whether to display monitored value in Decimal or Hexadecimal.	Section 9.3.1
			Character String Monitor Setting	Set the number of displayable characters to monitor character-string data.	Section 9.3.2
	Ladder		☞ GX Works2 Version1 Operating Manual (Simple Project)		
	SFC				
PLC Read/Write			☞ GX Works2 Version1 Operating Manual (Common)		
Online Change			☞ GX Works2 Version1 Operating Manual (Common)		
Compile	Basic Setting		Call Function Block	A function block will be enabled to call from ladder to ST or form ST to ladder. And the steps after compiling is reduced when a function block is called.	Section 4.4.1
			Execution of Program Check	Set when you do not execute the program check after the build, compile+online change. This setting can shorten the compile time.	-
	Output Result		Stop Build	Set the number of error and warning to stop the compile.	Section 7.6
			Disable Warning Codes	Register warning codes to invalidate. The registered warning codes shall not be displayed in output window.	
	Structured Ladder/ST	Compile Condition 1	Use the same label name in global label and local label	Select whether to use the same label name in global label and local label.	
			Use lower-case device names as labels	Select whether to use device names typed with lower case as labels.	
			Function Output Setting	Select whether to connect directly from the objective function output to other input.	
		Compile Condition 2	Structured Ladder	Select whether to assign each system device to output of functions that maintains bit type.	
		Compile Condition 3	Generation of Code Keeping bit Type Output	Select whether to generate a code to keep bit type output of objective function.	
Intelligent Function Module			☞ GX Works2 Version1 Operating Manual (Common)		
Cross Reference					
Project History					

APPENDIX

Appendix 1 List of Toolbars and Shortcut Keys. App-2

Appendix 1 List of Toolbars and Shortcut Keys

QCPU
Q

This section shows the list of toolbars and shortcut keys that can be used in Structured project.

For (Common) and (Simple) indicated in the Reference column, refer to the following manuals respectively:
(Common).....GX Works2 Version1 Operating Manual (Common)
(Simple).....GX Works2 Version1 Operating Manual (Simple Project)

Appendix 1.1 Common toolbars and shortcut keys

The following explains the toolbars that are available regardless of the editing target and the corresponding shortcut keys.

■ "Standard" toolbar icons

The following table shows the "Standard" toolbar icons and the corresponding shortcut keys.

Table App.1.1-1 "Standard" toolbar icons and shortcut keys

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
	Ctrl + N	New	Creates a new project.	(Common)
	Ctrl + O	Open	Opens an existing project.	
	Ctrl + S	Save	Saves the project.	

■ "New Object" toolbar icon

The following table shows "New Object" toolbar icon.

Table App.1.1-2 "New Object" toolbar icon

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
	–	New Object	Adds data to a project.	(Common)

■ "Program Common" toolbar icons

The following table shows the "Program Common" toolbar icons and the corresponding shortcut keys.

Table App.1.1-3 "Program Common" toolbar icons and shortcut keys

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
	-	Property	Displays properties of the selected data.	(Common)
	Ctrl + X	Cut	Cuts the selected data.	-
	Ctrl + C	Copy	Copies the selected data.	
	Ctrl + V	Paste	Pastes the cut/copied data at the cursor position.	
	Ctrl + Z	Undo	Cancels the previous operation.	Section 6.2.5
	Ctrl + Insert + F	Find	Searches for a character string.	(Common)
	-	Write to PLC	Writes data to the programmable controller CPU.	Chapter 8
	-	Read from PLC	Reads data from the programmable controller CPU.	
	F3	Start Monitoring	Starts monitoring the window being operated.	Section 9.1
	Alt + F3	Stop Monitoring	Stops monitoring the window being operated.	
	Ctrl + Insert + B	Build	Compiles uncompiled programs.	Section 7.2
	-	Rebuild All	Compiles all programs.	Section 7.1
	-	Start/Stop Simulation	Starts/stops simulation.	(Common)

■ Other shortcut keys

The following table shows other shortcut keys that are available regardless of the editing target.

Table App.1.1-4 Other shortcut keys common to various programs

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Alt + 7	-	Switches the display of the project data list and the uppermost work window.	-
-	Ctrl + F4	-	Closes the window displayed in the uppermost position.	-
-	Ctrl + F6	-	Moves to the next uppermost window.	-
-	Alt + F4	Quit	Closes the project being edited and exits GX Works2.	(Common)
-	Ctrl + Shift + C	Copy	Copies data in the project.	
-	Ctrl + Shift + V	Paste	Pastes the copied data to a folder.	
-	F2	Rename	Changes the name of the selected data or library in the Project window.	
-	Ctrl + Shift + E	Add New Module	Adds the intelligent function module data to the project being edited.	
-	Ctrl + Shift + H	Replace	Replaces the character string.	
-	Ctrl + E	Cross Reference	Creates the cross reference information.	
-	Ctrl + D	Device List	Displays the device usage list.	
-	Alt + Ctrl + ↓	Down	Searches for a device in the downward direction.	
-	Alt + Ctrl + ↑	Up	Searches for a device in the upward direction.	
-	Ctrl + Shift + F4	Online Program Change	Compiles the program and executes online program change.	
-	Shift + F3	Start Watching	Starts monitoring devices/labels registered to the Watch window.	
-	Shift + Alt + F3	Stop Watching	Stops monitoring devices/labels registered to the Watch window.	

Appendix 1.2 Toolbar icons and shortcut keys for setting labels

The following explains the toolbar icons and the corresponding shortcut keys used for setting labels.

■ "Label" toolbar icons

The following table shows the "Label" toolbar icons and the corresponding shortcut keys.

Table App.1.2-1 "Label" toolbar icons and shortcut keys

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
	Shift + Insert	New Declaration (Before)	Adds a line above the cursor position.	Section 5.5.2
	-	New Declaration (After)	Adds a line below the cursor position.	
	Shift + Delete	Delete Line	Deletes the line at the cursor position.	

■ Other shortcut keys

The following table shows other shortcut keys that are available for setting labels.

Table App.1.2-2 Other shortcut keys for setting labels.

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Ctrl + +	Expand Declaration	Displays the comment and remarks of the selected line in multiple lines.	Section 5.5.2
-	Ctrl + −	Collapse Declaration	Displays only the first line of the comment and remarks of the selected line.	
-	Ctrl + A	Select All	Selects all lines.	
-	Tab	-	Moves the cursor one cell to the right.	-
-	Shift + Tab	-	Moves the cursor one cell to the left.	-
-	Back Space	-	Changes the mode to the text entry mode if the cell is text entry enabled.	-
-	Shift + Space / Shift + Home / Shift + End	-	Selects cells in one line.	-
-	Select the button and press the Enter / Space / F2 key	-	The related setting screen is displayed.	-
-	Ctrl + Enter	-	Allows a line feed while entering a text in the "Comment" and "Remark" items.	Section 5.2 Section 5.3 Section 5.4 Section 5.6.1

Appendix 1.3 Toolbar icons and shortcut keys for setting device memory

The following explains the toolbar icons and the corresponding shortcut keys used for setting device memory.

■ "Device Memory" toolbar icons

The following table shows the "Device Memory" toolbar icons and the corresponding shortcut keys.

Table App.1.3-1 "Device Memory" toolbar icons and shortcut keys

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
	-	Display Mode/Binary	Changes display format to binary.	(Common)
	-	Display Mode/Octal	Changes display format to octal.	
	-	Display Mode/Decimal	Changes display format to decimal.	
	-	Display Mode/ Hexadecimal	Changes display format to hexadecimal.	
	-	Display Mode/Float	Changes display format to real numbers.	
	-	Display Mode/ASCII	Changes display format to ASCII.	
	-	Register/16-bit	Displays data in units of words.	
	-	Register/32-bit	Displays data in units of double words.	
	-	Register/64-bit	Displays data in units of 64 bits.	
	-	Upload Device Memory from PLC	Reads data in device memory from a programmable controller CPU.	
	-	Download Device Memory to PLC	Writes data in device memory to a programmable controller CPU.	
	-	Import from Excel File	Reads data from an Excel file.	
	-	Export to Excel File	Writes data to an Excel file.	
	Ctrl + I	Insert Device	Enters a device.	

■ Other shortcut keys

The following table shows other shortcut keys available for setting device memory.

Table Appendix.1.3-2 Other shortcut keys for setting device memory

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Insert	Insert Row	Inserts a row at the cursor position.	(Common)
-	Ctrl + G	Find Device	Searches for a device.	

Appendix 1.4 Toolbar icons for executing sampling trace

The following table shows the toolbar icons used for executing sampling trace.

Table Appendix.1.4-1 "Sampling Trace" toolbar icons

Toolbar icon	Shortcut key	Corresponding menu		Description	Reference
	-	Trace Setting		Displays the Trace Setting screen.	(Common)
	-	Trace Start/Stop		Starts or stops sampling trace. To start sampling trace, the trace ready signal (SM800) must be ON.	
	-	Display Trace Status		Displays the Trace Data Storage Status screen.	
	-	Monitor Status		Displays the current sampling trace status	-
	-		Execution Failed	Indicates the sampling trace stop status, or the status sampling trace has not started.	
	-		Before Trigger	Indicates the status sampling trace is executed but a trigger is not generated.	
	-		After Trigger	Indicates the status sampling trace is executed and a trigger is generated.	
	-		Stop	Indicates the status sampling trace is interrupted.	
	-		Completion	Indicates the status sampling trace has ended normally by acquiring trace data up to the specified total number of samplings after generation of a trigger.	
	-		Error	Indicates the status a sampling trace error occurred during sampling trace.	
	-	Buffer Status		Displayed when the trace data have been acquired up to the specified total number of samplings.	
	-	Trigger Occurrence		Displayed when a trigger is generated during sampling trace.	
	-	Zoom Out Timing Chart		Zooms the scale of timing chart	(Common)
	-	Zoom In Timing Chart			
	-	Zoom Out Trend Graph		Zooms the scale of trend graph	
	-	Zoom In Trend Graph			
	-	Chart/Switching Detail		Switches the trace result display on the Sampling Trace screen between the timing chart (graph) and the detailed data (value).	

Appendix 1.5 Toolbar icons and shortcut keys in program editors

The following explains the toolbar icons and the corresponding shortcut keys used for editing programs in each program editor.

■ "Ladder" toolbar icons and shortcut keys

The following table shows the toolbar icons and the corresponding shortcut keys available in the ladder editor.

Table App.1.5-1 "Ladder" toolbar icons and shortcut keys (1/2)

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
	F5	Open Contact	Inserts an open contact at the cursor position.	(Simple)
	Shift + F5	Open Branch	Inserts an open branch at the cursor position.	
	F6	Close Contact	Inserts a close contact at the cursor position.	
	Shift + F6	Close Branch	Inserts a close branch at the cursor position.	
	F7	Coil	Inserts a coil at the cursor position.	
	F8	Application Instruction	Inserts an application instruction at the cursor position.	
	F9	Horizontal Line	Inserts a horizontal line at the cursor position.	
	Shift + F9	Vertical Line	Inserts a vertical line at the cursor position.	
	Ctrl + F9	Delete Horizontal Line	Deletes the horizontal line at the cursor position.	
	Ctrl + F10	Delete Vertical Line	Deletes the vertical line at the cursor position.	
	Shift + F7	Rising Pulse	Inserts a rising pulse at the cursor position.	
	Shift + F8	Falling Pulse	Inserts a falling pulse at the cursor position.	
	Alt + F7	Rising Pulse Branch	Inserts a rising pulse branch at the cursor position.	
	Alt + F8	Falling Pulse Branch	Inserts a falling pulse branch at the cursor position.	
	Alt + F5	Operation Result Rising Pulse	Inserts an operation result rising pulse at the cursor position.	
	Alt + Ctrl + F5	Operation Result Falling Pulse	Inserts an operation result falling pulse at the cursor position.	
	Alt + Ctrl + F10	Invert Operation Results	Inserts an operation result inversion at the cursor position.	
	F10	Edit Line	Inputs a line at the cursor position.	
	Alt + F9	Delete Line	Deletes the line at the cursor position.	
	-	Device Comment	Edits device comments.	
	-	Statement	Edits the ladder statement at the cursor position.	
	-	Note	Edits the note at the cursor position.	
	Ctrl + F	Find Device	Searches for a device.	
	-	Find Instruction	Searches for an instruction.	
	-	Address Display	Displays a device actually assigned with compilation.	
	-	Zoom	Changes the display magnification ratio of the ladder.	

Table App.1.5-1 "Ladder" toolbar icons and shortcut keys (2/2)

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Shift + Insert	Insert Row	Inserts a row at the cursor position.	(Simple)
-	Shift + Delete	Delete Row	Deletes the row at the cursor position.	
-	Ctrl + Insert	Insert Column	Inserts a column at the cursor position.	
-	Ctrl + Delete	Delete Column	Deletes the column at the cursor position.	
-	F4	Ladder Conversion	Converts the program being edited.	
-	Ctrl + ↑ / ↓ / → / ←	-	Moves the cursor on the editing screen while the Enter Symbol screen is displayed.	
-	Ctrl + G	Jump	Displays the specified row.	
-	Ctrl + F5	Comment	Displays device comments.	
-	Ctrl + F7	Statement	Displays statements.	
-	Ctrl + F8	Note	Displays notes.	
-	Ctrl + R	Back to SFC Block	Opens the SFC diagram corresponds to the program on the Zoom screen.	

■ "ST" toolbar icons and shortcut keys

The following table shows the toolbar icons and the corresponding shortcut keys available in the ST editor.

Table App.1.5-2 "ST" toolbar icons and shortcut keys

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Ctrl + Y	Redo	Redoes the operation canceled with <<Undo>>.	Section 6.2.5
VAR	F2	List Operands	Displays the screen (label list screen) for selecting an existing label to be inserted.	Section 6.2.3
-	Alt + N	New Label	Displays the screen (label list screen for creating new label) for adding a new label to be inserted.	
-	Ctrl + F1	Create Template	Displays the template of argument determined for each instruction/function.	Section 6.2.2
	Ctrl + F7	Toggle Bookmark	Set a bookmark at the cursor line. If a bookmark has already been set, deletes the set bookmark.	Section 6.3.4
	F7	Next Bookmark	Displays the next bookmark position.	
	Shift + F7	Previous Bookmark	Displays the previous bookmark position.	
	Ctrl + Shift + F7	Delete All Bookmarks	Deletes all bookmarks.	
	-	Zoom In	Zooms in the screen being edited one level.	Section 6.2.7
	-	Zoom Out	Zooms out the screen being edited one level.	
-	Ctrl + F8	Split Window	Starts monitoring in the split window format that displays the monitor data of numerical values and character string.	Section 9.4

■ "Structured Ladder" toolbar icons and shortcut keys

The following table shows the toolbar icons and the corresponding shortcut keys available in the structured ladder editor.

Table App.1.5-3 "Structured Ladder" toolbar icons and shortcut keys (1/2)

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Ctrl + Y	Redo	Redoes the operation canceled with <<Undo>>.	Section 6.2.5
	Ctrl + A	Select Mode	Changes the input format for positioning contacts/coils.	Section 6.4.3
	Ctrl + T	Interconnect Mode	Changes the input mode for drawing lines.	Section 6.4.4
-	Ctrl + B	Auto Connect	Connects the start and end points to draw a line.	
	Ctrl + G	Guided Mode/Guided Editing	Changes the input format for entry with keyboard.	Section 6.4.12
-	Insert	Guided Mode /Overwrite, Insert Mode	Switches the entry mode between Overwrite/ Insert in the guided mode.	
-	Ctrl + L	Guided Mode/Line Mode	Changes the input mode for drawing lines in Guided editing.	
	Alt + Shift + A	Guided Mode/Auto Comment	Adds a comment entry field at the start of the network added in Guided editing.	
	Ctrl + W	Insert Row	Inserts a row in the ladder being edited.	Section 6.4.5
	Ctrl + U	Insert Column	Inserts a column in the ladder being edited.	
	Alt + B	New Network (Before)	Inserts a new network in front of the network being edited.	Section 6.2.4
	Alt + A	New Network (After)	Inserts a new network in back of the network being edited.	
	1	Contact	Inserts a contact at the cursor position.	Section 6.4.2
	2	Contact Negation	Inserts a contact negation at the cursor position.	
	7	Coil	Inserts a coil at the cursor position.	
	Ctrl + J	Jump	Inserts a jump at the cursor position.	
	Ctrl + R	Return	Inserts a return at the cursor position.	
	3	L-Connect Contact	Inserts an L-connect contact at the cursor position.	
	4	L-Connect Contact Negation	Inserts an L-connect contact negation at the cursor position.	
	9	Input Variable	Inserts an input variable at the cursor position.	
	0	Output Variable	Inserts an output variable at the cursor position.	
	5	Vertical Line	Inserts a vertical line at the cursor position.	
	6	Horizontal Line	Inserts a horizontal line at the cursor position.	
	Ctrl + M	Comment	Inserts a comment entry field.	
	Ctrl + Shift + L	Network Label	Displays the Network Header screen.	
	F2	List Operands	Displays the screen (label list screen) for selecting an existing label to be inserted.	Section 6.2.3
-	Alt + N	New Label	Displays the screen (label list screen for creating new label) for adding a new label setting to be inserted.	

Table App.1.5-3 "Structured Ladder" toolbar icons and shortcut keys (2/2)

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
	+	Increment Pins	Increments the number of arguments of functions and function blocks.	Section 6.4.7
	−	Decrement Pins	Decrements the number of arguments of functions and function blocks.	
-	Alt + Ctrl + S	Signal Configuration/ Configure	Sets the type of contact and coil.	Section 6.4.6
-	Alt + Ctrl + C	Signal Configuration/ Toggle	Changes the type of contact and coil in the order shown below: • Contact → Contact negation • Coil → Reversal coil → Set → Reset	
-	Ctrl + Shift + M	Change Display Label-Device-Address	Switches the display format in order (label, device address).	Section 6.4.10
-	Ctrl + Shift + K	Change Display Label-Comment	Switches the display between label and comment.	
	-	Zoom In	Zooms in the screen being edited one level.	Section 6.2.7
	-	Zoom Out	Zooms out the screen being edited one level.	
-	Shift + F2	Zoom Header/Body/ Header	Opens the label setting screen for the selected POU.	Section 6.2.8

■ "SFC Symbol" toolbar icons and shortcut keys

The following table shows the toolbar icons and the corresponding shortcut keys available in the SFC editor.

Table App.1.5-4 "SFC Symbol" toolbar icons and shortcut keys (1/2)

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Shift + Insert	Insert Row	Inserts a row at the cursor position.	(Simple)
-	Shift + Delete	Delete Row	Deletes a row at the cursor position.	
-	Ctrl + Insert	Insert Column	Inserts a column at the cursor position.	
-	Ctrl + Delete	Delete Column	Deletes a column at the cursor position.	
	F5	Step	Inserts at the cursor position.	
	F6	Block START Step - with END check	Inserts at the cursor position.	
	Shift + F6	Block START Step - with no END check	Inserts at the cursor position.	
	F8	Jump	Inserts at the cursor position.	
	F7	END Step	Inserts at the cursor position.	
	Shift + F5	Dummy Step	Inserts at the cursor position.	
	F5	Transition	Inserts at the cursor position.	
	F6	Selection Divergence	Inserts a selection divergence.	
	F7	Simultaneous Divergence	Inserts a simultaneous divergence.	
	F8	Selection Convergence	Inserts a selection convergence.	
	F9	Simultaneous Convergence	Inserts a simultaneous convergence.	
	Shift + F9	Vertical Line	Inserts a vertical line.	
	Ctrl + 1	Normal	Sets the step attribute to Normal.	
	Ctrl + 2	Coil Saving	Sets the step attribute to Stored Coil.	

Table App.1.5-4 "SFC" toolbar icons and shortcut keys (2/2)

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
	Ctrl + 3	Action Saving - with no transition check	Sets the step attribute to Action Saving (SE).	(Simple)
	Ctrl + 4	Action Saving - with transition check	Sets the step attribute to Action Saving (ST).	
	Ctrl + 5	Reset	Resets the step attribute.	
	Alt + F5	Vertical Line (Draw Line)	Inserts  at the cursor position.	
	Alt + F7	Selection Divergence (Draw Line)	Inserts  at the cursor position.	
	Alt + F8	Simultaneous Divergence (Draw Line)	Inserts  at the cursor position.	
	Alt + F9	Selection Convergence (Draw Line)	Inserts  at the cursor position.	
	Alt + F10	Simultaneous Convergence (Draw Line)	Inserts  at the cursor position.	
	Ctrl + F9	Delete Line	Deletes a line at the cursor position.	
	Alt + S	SFC Step/Edit Step Comment	Edits the SFC step/transition comments.	
	-	SFC Step No. Sort	Sorts the SFC block step numbers.	
	-	SFC All Block Batch Monitor	Batch monitors all blocks in the SFC program.	
	-	SFC Auto Scroll Monitor	Scrolls the screen to display active steps automatically when they are out of the screen during monitoring.	
	-	Zoom	Changes the display magnification ratio of the ladder.	
-	Alt + Ctrl + F4	Build All SFC Program	Converts all SFC programs in the project.	
-	Ctrl + F5	SFC Step/Transition Comment	Displays the SFC step/transition comments.	
-	Ctrl + L / Ctrl + double click	Open Step/Transition	Displays the Zoom screen or the start destination block.	
-	Space	-	Displays the start destination block.	
-	Ctrl + R	Back to SFC Block	Displays the start source block.	

■ SFC block list shortcut keys

The following table shows the shortcut keys available in the SFC block list.

Table App.1.5-5 SFC block list shortcut keys

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Ctrl + G	Jump	Jumps to the specified block number/ data name.	(Simple)
-	Numeric key	-	Jumps to the selected block number.	
-	Alt + Ctrl + F4	Build All SFC Program	Converts all SFC programs in the project.	
-	Ctrl + F5	SFC Block List Comment	Displays comments of the SFC block list.	

■ Other shortcut keys

The following table shows other shortcut keys available in program editors.

Table App.1.5-6 Other shortcut keys available in program editors

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	↑ / ↓ / → / ←	-	Moves the cursor in the arrow direction.	-
-	Space	-	Operates in the same way as the mouse button click during editing.	
-	PgUp	-	Scrolls up the screen.	
-	PgDown	-	Scrolls down the screen.	
-	Ctrl + PgUp	-	Scrolls the screen to the right.	
-	Ctrl + PgDown	-	Scrolls the screen to the left.	
-	Home	-	Moves the cursor to its leftmost position in the row.	
-	End	-	Moves the cursor to its rightmost position in the row.	
-	Ctrl + Home	-	Moves the cursor to the start of the program.	
-	Ctrl + End	-	Moves the cursor to the bottom of the program.	
-	Shift + ↑ / ↓ / → / ←	-	Sets a range.	
-	Ctrl + Shift + Home	-	Selects the range from the current position up to the start of the program.	
-	Ctrl + Shift + End	-	Selects the range from the current position down to the end of the program.	
-	Delete	-	Deletes the selected target.	

Appendix 1.6 Shortcut key for editing intelligent function module data

The following explains the shortcut key used for editing intelligent function module data.

■ Shortcut key for editing type QD75 positioning module data

The following table shows the shortcut key used for editing type QD75 positioning module data.

Table App.1.6-1 Shortcut key for editing type QD75 positioning module data

Toolbar icon	Shortcut key	Corresponding menu	Description	Reference
-	Ctrl + A	-	Selects all data on the Positioning Data and Block Start Data screens.	-

MEMO

INDEX

[A]

address 6-42
Application functions 10-2
array 5-10
Assigning devices automatically 5-20, 7-6
Auto connect 6-29, 6-32

[B]

bit specification for word devices 5-16
bookmarks 6-24
Build 7-4

[C]

check 7-7
Class 5-3, 5-5, 5-7
Close Contact 6-35
Coil 6-27, 6-35
Comment 6-27
comment 5-3, 5-5, 5-7, 5-13, 6-4
comments 6-27, 6-44
Common Instructions 10-2
Compilation 7-7
Compile condition 11-4
Conditions of Monitoring 9-4
Contact 6-27
Contact Negation 6-27
current value display 9-4
current values 9-2

[D]

Data Name 5-15, 5-17
data type 5-9
decimal and hexadecimal 9-4
device 5-3, 6-42
display format 6-42
Docking window 2-2

[E]

errors 7-7, 7-8
executing conditions 4-6

[F]

function 4-10, 4-13, 6-3
function block 4-10, 4-13, 6-3
Function Block Selection Window 2-2, 4-14, 6-5

[G]

Global label 5-3
grid 6-17, 6-46
Guided mode 6-26, 6-46

[I]

indention 6-23
Initial 5-3, 5-5, 5-7, 5-14
Input Variable 6-27
Instance 4-14, 6-39, 6-40, 9-3
Interconnect mode 6-26

[J]

Jump 6-27

[L]

Label Name 5-3, 5-5, 5-7
Label Selection 6-10
label setting 5-2, 5-9
Labels 5-2
Ladder 1-2
Ladder diagram 1-2
Library 10-2
library help 10-17
library passwords 10-16
line 6-27, 6-32, 6-47
List of Functions 1-5
local labels 5-5, 5-7

[M]

main frame 2-2
menu 2-2

[N]

network elements 6-27
network header 6-15
Network label 6-15
Network List 6-16
Network title 6-15
networks 6-12

[O]

Open Contact 6-35
Operator 6-5
Output Variable 6-27
Output window 2-2, 7-7
output window 7-7

[P]

POUs 4-9, 4-13
Priority 4-7
Program 4-6, 6-2

Program blocks 4-4, 4-13, 6-3
Program Configuration 4-2
Program Editor 6-2
program files 4-3
programming language 1-2, 6-3
Project window 2-2, 4-2
Property 4-7, 4-10

[R]

reading symbolic information from programmable controller CPU 8-4
Rebuild All 7-2
Remark 5-3, 5-13, 6-21
Reset 6-35
Return 6-27
Reversal coil 6-35

[S]

Select mode 6-26
Set 6-35
Setting Ranges for Devices Assigned Automatically 5-20
SFC 6-3
Shortcut Keys App-2
Simple project A-24
Special instructions 10-2
ST 1-2, 6-22
structure 5-14
Structured Data Setting 5-14
structured data type 5-14
structured ladder 1-2, 6-25
Structured project 1-2
Structured text 1-2
Symbolic information 8-4

[T]

tabulator 6-24
tasks 4-3
Templates 6-9
tool hint 6-21
toolbar 2-2, App-2

[U]

user library 10-2, 10-4

[V]

variable 6-39, 6-40, 6-49

[W]

warning 7-7, 7-8, 7-9
word device 9-4
Work window 2-2
writing symbolic information from programmable controller CPU 8-4

GX Works2 Version1
Operating Manual
(Structured Project)

MODEL	GXW2-VER1-O-KP-E
MODEL CODE	13JU65
SH(NA)-080781ENG-A(0807)KWIX	

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
NAGOYA WORKS : 1-14 , YADA-MINAMI 5-CHOME , HIGASHI-KU, NAGOYA , JAPAN