OMRON

Programmable Terminal

NA-series

Software

User's Manual

NA5-15□101□ (-V□)
NA5-12□101□ (-V□)
NA5-9□001□ (-V□)
NA5-7□001□ (-V□)
NA-RTLD□□

SYSMAC
always in control

V118-E1-28

NOTE

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, mechanical, electronic, photocopying, recording, or otherwise, without the prior written permission of OMRON.

No patent liability is assumed with respect to the use of the information contained herein. Moreover, because OMRON is constantly striving to improve its high-quality products, the information contained in this manual is subject to change without notice. Every precaution has been taken in the preparation of this manual. Nevertheless, OMRON assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained in this publication.

Trademarks

- Sysmac and SYSMAC are trademarks or registered trademarks of OMRON Corporation in Japan and other countries for OMRON factory automation products.
- Microsoft, Windows, Microsoft Edge, Excel, and Visual Basic are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.
- EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
- ODVA, CIP, CompoNet, DeviceNet, and EtherNet/IP are trademarks of ODVA.
- The SD and SDHC logos are trademarks of SD-3C, LLC.
- Portions of this software are copyright 2014 The FreeType Project (freetype.org). All rights reserved.
- Celeron, Intel, Intel Core and Intel Atom are trademarks of Intel Corporation in the U.S. and / or other countries.

Other company names and product names in this document are the trademarks or registered trademarks of their respective companies.

Copyrights

- Microsoft product screen shots used with permission from Microsoft.
- This product incorporates certain third party software. The license and copyright information associated with this software is available at https://www.fa.omron.co.jp/product/tool/na5-info/index_en.html.

Introduction

Thank you for purchasing an NA-series Programmable Terminal.

This manual contains information that is necessary to use the NA-series Programmable Terminal. Please read this manual and make sure you understand the functionality and performance of the NA-series Programmable Terminal before you attempt to use it in a control system.

Keep this manual in a safe place where it will be available for reference during operation.

Intended Audience

This manual is intended for the following personnel, who must also have knowledge of electrical systems (an electrical engineer or the equivalent).

- Personnel in charge of introducing FA systems.
- Personnel in charge of designing FA systems.
- Personnel in charge of installing and maintaining FA systems.
- Personnel in charge of managing FA systems and facilities.

Applicable Products

This manual covers the following products.

- NA-series Programmable Terminals[*1]

*1. Unless otherwise specified, the descriptions for the NA5-□□W□□□□ apply to the NA5-□□U□□□□ as well.

Relevant Manuals

The basic information required to use an NA-series PT is provided in the following manuals.

- *NA-series Programmable Terminal Hardware User's Manual* (Cat. No. V117)
- *NA-series Programmable Terminal Hardware(-V1) User's Manual* (Cat. No. V125)
- *NA-series Programmable Terminal Hardware(-V2) User's Manual* (Cat. No. V128)
- *NA-series Programmable Terminal Software User's Manual* (Cat. No. V118)
- *NA-series Programmable Terminal Device Connection User's Manual* (Cat. No. V119)
- *NA-series Programmable Terminal Soft-NA User's Manual* (Cat. No. V126)

Operations are performed from the Sysmac Studio Automation Software.

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

Other manuals are necessary for specific system configurations and applications.

The following manual is also available to walk you through installations and operations up to starting actual operation using simple examples.

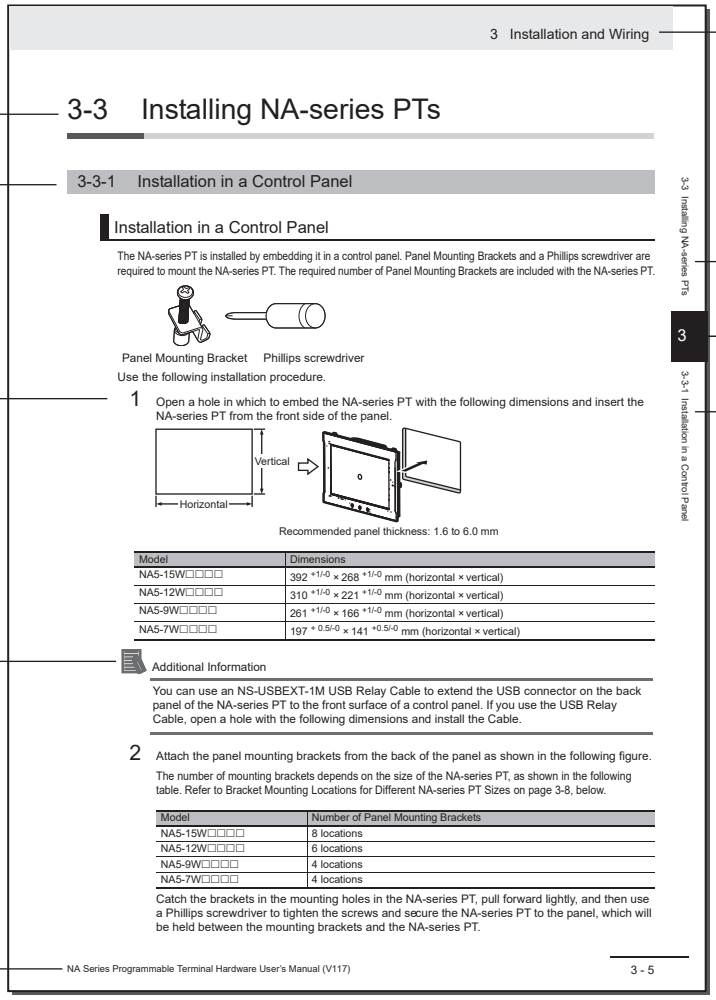
Refer to it as required.

- *NA-series Programmable Terminal Startup Guide Manual* (Cat. No. V120)

Manual Structure

Page Structure and Markings

The following page structure is used in this manual.



Note This illustration is provided only as a sample. It may not literally appear in this manual.

Special Information

Special information in this manual is classified as follows:

Precautions for Safe Use

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

Precautions for Correct Use

Indicates precautions on what to do and what not to do to ensure proper operation and performance.

Additional Information

Additional information to read as required.

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

Version Information

Information on differences in specifications and functionality with different versions is given.

Sections in this Manual

1 Introduction to the NA-series Programmable Terminals

2 Basic Sysmac Studio Operations

3 HMI Configuration and Setup

4 Creating the HMI Application

5 Objects

6 Connecting to the HMI

7 Debugging

8 Transfer

9 Reusing Objects

10 Connecting to HMIs from External Devices

11 Other Functions

A Appendices

I Index

CONTENTS

Introduction **1**

Relevant Manuals **2**

Manual Structure **3**

Sections in this Manual **5**

Terms and Conditions Agreement **13**

Safety Precautions **15**

Precautions for Safe Use **18**

Related Manuals **20**

Terminology **30**

Revision History **31**

Section 1 Introduction to the NA-series Programmable Terminals

1-1 NA-series Programmable Terminals **1-2**
1-1-1 Features 1-2

1-2 How HMIs Operate **1-4**
1-2-1 HMI Software Configuration 1-4
1-2-2 HMI Projects 1-4
1-2-3 Pages 1-4
1-2-4 Objects 1-5
1-2-5 Memory Specifications for Connected Devices 1-6
1-2-6 Events 1-7
1-2-7 Subroutines 1-8
1-2-8 Functions Shared by the Entire HMI Project 1-9
1-2-9 Data That Retained When Power Is Turned OFF 1-9

1-3 Operating Procedure for HMIs **1-10**
1-3-1 Overall Procedure 1-10
1-3-2 Procedure Details 1-11

Section 2 Basic Sysmac Studio Operations

2-1 Parts of the Sysmac Studio Window **2-2**
2-1-1 Application Window 2-2

2-2 Menu Command Structure **2-7**

2-3 Basic Editing Operations **2-10**

2-4 Sysmac Studio Settings and Operations 2-12
2-4-1 Setting Parameters 2-12
2-4-2 Programming 2-12
2-4-3 Library Functions 2-13
2-4-4 Operations for Debugging 2-13
2-4-5 Communications 2-13
2-4-6 Security Measures 2-14
2-4-7 Online Help 2-14
2-4-8 Project Management Functions 2-14

2-5 Basic Operations for HMI Projects 2-15
2-5-1 Creating a Project File from the Start Page 2-15
2-5-2 Adding an HMI to an Existing Project 2-16
2-5-3 Changing Devices 2-17
2-5-4 Importing and Exporting Devices 2-19

Section 3 HMI Configuration and Setup

3-1 Outline of Configurations and Setup 3-2
3-1-1 Connected Device Registration and Variable Mapping 3-2

3-2 Device References 3-3
3-2-1 Types of Connected Devices 3-3
3-2-2 Connected Devices in the Current Project 3-3
3-2-3 Registering External Connected Devices 3-4

3-3 Mapping Variables 3-7
3-3-1 Mapping Variables 3-7
3-3-2 Opening the Variable Mapping Tab Page and Tab Page Parts 3-7
3-3-3 Variable Mapping Methods 3-8
3-3-4 When a Variable Mapping Error Cannot be Resolved 3-9

3-4 HMI Settings 3-11
3-4-1 HMI Settings 3-11
3-4-2 Device Settings 3-12
3-4-3 TCP/IP Settings 3-14
3-4-4 FTP Settings 3-15
3-4-5 NTP Settings 3-16
3-4-6 FINS Settings 3-17
3-4-7 VNC Settings 3-18
3-4-8 Printing Settings 3-19
3-4-9 Serial Port Settings 3-20

3-5 Security Settings 3-21

3-6 Troubleshooter 3-22

3-7 Language Settings 3-23

3-8 Operation Log Settings 3-25

3-9 HMI Clock 3-28

3-10 Updating the HMI Name 3-29

3-11 Write Protecting the HMI 3-30

3-12 Clear All Memory 3-31

3-13 Resetting the HMI 3-32

Section 4 Creating the HMI Application

4-1 Registering Variables 4-3
4-1-1 Variables 4-3
4-1-2 Registering Global Variables 4-4
4-1-3 Registering External Variables 4-6
4-1-4 Searching for Unused Variables 4-11
4-1-5 Attributes and Entry Methods for Global Variables 4-12
4-1-6 System-defined Variables 4-16
4-1-7 Subroutine Variables 4-20

4-2 Registering Data Types 4-21

4-3 Creating Pages 4-22
4-3-1 Displaying Pages 4-23
4-3-2 Registering Pages 4-23
4-3-3 Page Property Settings 4-25
4-3-4 Editing Pages 4-27

4-4 Setting Common Object Functions 4-35
4-4-1 Registering User Alarms 4-35
4-4-2 Setting Controller Events 4-36
4-4-3 Registration for Data Logging 4-39
4-4-4 Registering Data Groups 4-40
4-4-5 Registering Recipes 4-42
4-4-6 Registering Custom Keypads 4-44
4-4-7 Setting Global Events and Corresponding Actions 4-47
4-4-8 Registering Global Subroutines 4-49
4-4-9 Setting Up Resources 4-50
4-4-10 Setting Up IAG Resources 4-52
4-4-11 Registering Scaling 4-53

4-5 Subroutines 4-56
4-5-1 Subroutine Execution 4-57
4-5-2 Precautions on Internal Processing 4-61
4-5-3 Code Editor 4-62
4-5-4 Differences in Language Specifications 4-62

4-6 Search and Replace 4-63

4-7 Font Replacement 4-64

4-8 IAG Replacement 4-73

4-9 Device Replacement 4-76

4-10 Cross References 4-79
4-10-1 Cross References 4-79

4-11 Building 4-80
4-11-1 Building 4-80
4-11-2 Build Operation 4-81

4-12 Offline Comparison 4-82

4-13 Resource Usage 4-83

Section 5 Objects

5-1 Objects 5-2
5-1-1 Object List 5-2
5-1-2 Object Attributes 5-4
5-1-3 Using Objects 5-8

5-2 Examples of Using Objects 5-12
5-2-1 Displaying a PDF File 5-12
5-2-2 Displaying a User Alarm 5-14
5-2-3 Inserting a Variable Value in a User Alarm 5-16
5-2-4 Displaying a Trend Graph 5-18
5-2-5 Displaying a Broken-line Graph 5-21
5-2-6 Using a Recipe 5-24
5-2-7 Setting the Order of Automatic Move of Input Focus 5-28
5-2-8 Displaying Text Strings by Indirect Addressing 5-31
5-2-9 Creating Buttons with the Lamp Function 5-33
5-2-10 Creating Buttons to Output Operation Log Files 5-35

Section 6 Connecting to the HMI

6-1 Introduction 6-2

6-2 Going Online with an HMI 6-3
6-2-1 Methods for Going Online with an HMI 6-3
6-2-2 Setting the Connection Method 6-4
6-2-3 Online Connection 6-5
6-2-4 Going Online after Checking the Connection Method 6-6
6-2-5 Going Offline 6-6
6-2-6 Confirming Serial IDs 6-7

Section 7 Debugging

7-1 HMI Debugging Functions 7-2
7-1-1 Watch Tab Page 7-3
7-1-2 Breakpoints 7-3
7-1-3 Step Execution 7-5
7-1-4 Simulator Functions 7-8
7-1-5 Offline Debugging with Only the HMI Simulator 7-9
7-1-6 Offline Debugging with the Controller Simulator 7-10

Section 8 Transfer

8-1 Transfer 8-2

8-2 Download Function 8-3

8-3 Upload Function 8-4

8-4 Relinking Internal Devices 8-5

8-5 Synchronization 8-8
8-5-1 Downloading While Online 8-12
8-5-2 Using Storage Media for Downloading 8-13
8-5-3 Uploading Projects Online 8-16
8-5-4 Uploading with Storage Media 8-18

Section 9 Reusing Objects

9-1 Reusing Objects 9-2

9-2 IAGs 9-3
9-2-1 Differences when an IAG Project Is Selected 9-3
9-2-2 Creating an IAG 9-10
9-2-3 Using IAGs 9-13

9-3 Custom Objects 9-16
9-3-1 Objects That You Can Register as Custom Objects 9-16
9-3-2 Creating Custom Objects 9-16
9-3-3 Deleting Custom Objects 9-19
9-3-4 Using Custom Objects 9-20

Section 10 Connecting to HMIs from External Devices

10-1 Accessing an HMI from an External Device 10-2
10-1-1 VNC 10-2
10-1-2 FTP 10-3

Section 11 Other Functions

11-1 Sysmac Studio Option Settings 11-2
11-2 Printing 11-5
11-2-1 Printable Items 11-5
11-3 Image File Output 11-6
11-4 Import/Export User Alarm 11-7
11-4-1 Importing User Alarms 11-7
11-4-2 Exporting User Alarms 11-9
11-4-3 File Format 11-11
11-5 Import/Export Resources 11-13
11-5-1 Importing Resources 11-13
11-5-2 Exporting Resources 11-16
11-5-3 File Format 11-19
11-6 Import/Export Object Properties 11-20
11-6-1 Importing Object Properties 11-20
11-6-2 Exporting Object Properties 11-21
11-6-3 File Format 11-22
11-7 Importing/Exporting Pages 11-24
11-7-1 Importing Pages 11-24
11-7-2 Exporting Pages 11-25

Appendices

A-1 Events and Actions A-2
A-2 Supported Formats A-5
A-3 Specifications of User Alarm Log Files A-7
A-4 Specifications of Data Log Files A-8
A-5 Specifications of Operation Log Files A-9
A-6 Differences of NA5-□□W□□□□-V2 ..... A-12
A-7 Differences between the Physical HMI and Simulator A-15
A-8 Version Upgrade History A-16
A-8-1 Version Upgrade History for Sysmac Studio and Runtime A-16
A-8-2 Version Upgrade History for Sysmac Studio Only A-20
A-8-3 Sysmac Studio Corresponding Versions A-24
A-8-4 Runtime Support Limit A-25
A-9 Precautions for Version Upgrades A-26

Index

Terms and Conditions Agreement

Warranty, Limitations of Liability

Warranties

- **Exclusive Warranty**

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

- **Limitations**

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

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

- **Buyer Remedy**

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

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

Limitation on Liability; Etc

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

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

Application Considerations

Suitability of Use

Omron Companies shall not be responsible for conformity with any standards, codes or regulations which apply to the combination of the Product in the Buyer's application or use of the Product. At Buyer's request, Omron will provide applicable third party certification documents identifying ratings and limitations of use which apply to the Product. This information by itself is not sufficient for a complete determination of the suitability of the Product in combination with the end product, machine, system, or other application or use. Buyer shall be solely responsible for determining appropriateness of the particular Product with respect to Buyer's application, product or system. Buyer shall take application responsibility in all cases.

NEVER USE THE PRODUCT FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY OR IN LARGE QUANTITIES WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE OMRON PRODUCT(S) IS PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.

Programmable Products

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

Disclaimers

Performance Data

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

Change in Specifications

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

Errors and Omissions

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

Safety Precautions

Definition of Precautionary Information

The following notation is used in this manual to provide precautions required to ensure safe usage of the NA-series Programmable Terminal. The safety precautions that are provided are extremely important to safety. Always read and heed the information provided in all safety precautions.

The following notation is used.

WARNING	Indicates a potentially hazardous situation which, if not avoided, could result in mild or moderate injury or at the worst, serious injury or death. Additionally, there may be severe property damage.
Caution	Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage.

Precautions for Safe Use
Indicates precautions on what to do and what not to do to ensure safe usage of the product.

Precautions for Correct Use
Indicates precautions on what to do and what not to do to ensure proper operation and performance.

Symbols

	The circle and slash symbol indicates operations that you must not do. The specific operation is shown in the circle and explained in text. This example indicates prohibiting disassembly.
	The triangle symbol indicates precautions (including warnings). The specific operation is shown in the triangle and explained in text. This example indicates a general precaution.
	The filled circle symbol indicates operations that you must do. The specific operation is shown in the circle and explained in text. This example shows a general precaution for something that you must do.

Warnings

⚠ WARNING

Always ensure that the personnel in charge confirm that installation, inspection, and maintenance were properly performed for the NA Unit. “Personnel in charge” refers to individuals qualified and responsible for ensuring safety during machine design, installation, operation, maintenance, and disposal.	⚠
Ensure that installation and post-installation checks are performed by personnel in charge who possess a thorough understanding of the machinery to be installed.	⚠
Do not use the input functions such as the touch panel or function keys of the NA Unit, in applications that involve human life, in applications that may result in serious injury, or for emergency stop switches.	⊘
Never press two points or more on the touch panel of the NA Unit at a time. Touching two points or more interrupts normal touch panel operations.	⊘

Security Measures

Anti-virus protection

Install the latest commercial-quality antivirus software on the computer connected to the control system and maintain to keep the software up-to-date.

Security measures to prevent unauthorized access

Take the following measures to prevent unauthorized access to our products.

- Install physical controls so that only authorized personnel can access control systems and equipment.
- Reduce connections to control systems and equipment via networks to prevent access from untrusted devices.
- Install firewalls to shut down unused communications ports and limit communications hosts and isolate control systems and equipment from the IT network.
- Use a virtual private network (VPN) for remote access to control systems and equipment.
- Adopt multifactor authentication to devices with remote access to control systems and equipment.
- Set strong passwords and change them frequently.
- Scan virus to ensure safety of USB drives or other external storages before connecting them to control systems and equipment.

Data input and output protection

Validate backups and ranges to cope with unintentional modification of input/output data to control systems and equipment.

- Checking the scope of data
- Checking validity of backups and preparing data for restore in case of falsification and abnormalities
- Safety design, such as emergency shutdown and fail-soft operation in case of data tampering and abnormalities

Data recovery

Backup data and keep the data up-to-date periodically to prepare for data loss.

When using an intranet environment through a global address, connecting to an unauthorized terminal such as a SCADA, HMI or to an unauthorized server may result in network security issues such as spoofing and tampering. You must take sufficient measures such as restricting access to the terminal, using a terminal equipped with a secure function, and locking the installation area by yourself.

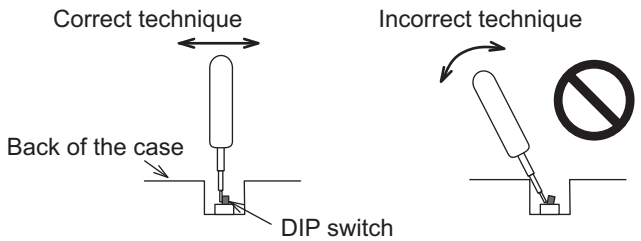
When constructing an intranet, communication failure may occur due to cable disconnection or the influence of unauthorized network equipment. Take adequate measures, such as restricting physical access to network devices, by means such as locking the installation area.

When using a device equipped with the SD Memory Card function, there is a security risk that a third party may acquire, alter, or replace the files and data in the removable media by removing the removable media or unmounting the removable media.

Please take sufficient measures, such as restricting physical access to the Controller or taking appropriate management measures for removable media, by means of locking the installation area, entrance management, etc., by yourself.

Precautions for Safe Use

- Turn OFF the power supply before connecting or disconnecting cables.
- Be certain to use the cables with lock mechanism such as serial cable or the Ethernet cable after confirming if it is securely locked.
- Confirm the safety of the system before turning ON or OFF the power supply, or pressing the reset switch.
- The whole system may stop depending on how the power supply is turned ON or OFF. Turn ON/OFF the power supply according to the specified procedure.
- Operate DIP switch according to the following way.



  The DIP switch may break if it is levered with a tool against the case as shown in the figure.
- Once the DIP switch settings are changed, reset by pressing the reset switch, or restart the power supply.
- Initialize the project, after confirming that existing project is backed up at the Sysmac Studio.
- When changing the password, do not reset or turn OFF the power supply until the writing is completed. A failure to store the password may cause the project to fail to function.
- While uploading or downloading a project or a system program, do not perform the operations as follows. Such operations may corrupt the project or the system program:
  - Turning OFF the power supply of the NA Unit
  - Resetting the NA Unit.
  - Removing the USB devices or SD card.
  - Disconnecting the cable between a support tool and the NA Unit.
- Dispose of the NA Units and batteries according to local ordinances as they apply.

- The following precaution must be displayed on all products containing lithium primary batteries with a perchlorate content of 6 ppb or higher when exporting them to or shipping them through California, USA.

  Perchlorate Material - special handling may apply.

  See www.dtsc.ca.gov/hazardouswaste/perchlorate

  The Battery is a primary lithium battery with a perchlorate content of 6 ppb or higher. Label all packing and shipping containers appropriately when shipping finished products that contain a NA Unit with a mounted Battery to the State of California, USA.
- Do not connect the USB devices in the environment subject to the strong vibration.
- Use a USB memory device for temporary purposes such as data transfer.
- Do not connect USB devices which are not allowed to connect to NA Unit.
- Start actual system application only after checking normal operation of the system including storage devices such as USB memory and SD card.
- When connecting peripheral devices which do not meet the performance level of the NA Unit for noise and static electricity, ensure sufficient countermeasures against noise and static electricity during installation of the peripheral devices to the NA Unit.

- Do not carry out the following operations when accessing USB devices or SD card:
  - Turning OFF the power supply of the NA Unit
  - Press the Reset switch of the NA Unit
  - Pull out the USB devices or SD card
- To ensure the system's safety, make sure to incorporate a program that call periodically signals during the operation at connected device side and can confirm the normal functionality of the NA Unit before running the system.
- Start actual system application only after sufficiently checking project, subroutine and the operation of the program at the connected device side.
- To execute a subroutine with multiple threads, fully check the operation of the program that takes multithreads into consideration, before starting actual system application.
- To use numeric input functions safely, always make maximum and minimum limit settings.
- The deterioration over time may cause the touch points to move on the touch panel. Calibrate the touch panel periodically.
- A touch position detection error of approximately 20 pixels may occur due to the precision of the touch panel. Always take this into account when positioning objects on the panel so adjoining objects will not be activated by mistake.
- Confirm the safety of the system before pressing the touch panel.
- Do not accidentally press the touch panel when the backlight is not lit or when the display does not appear or is too dark to identify visually.
- You can change the brightness by changing the setting such as in the system menu or by downloading project.
  If the brightness is set to very dark, it causes flickering or unreadable screen. Additionally, the brightness can be restored by transferring the project again after setting the property of the brightness appropriately.
  In a case of the applications where end users can control the brightness, create the applications so as keeping on operations by such as assigning the function which restores the brightness to one of function keys, if necessary.
- Signals from the touch panel may not be entered if the touch panel is pressed consecutively at high speed. Make sure to go on the next operation after confirming that the NA Unit has detected the input of the touch panel.

Related Manuals

The following manuals are related to the NA-series PTs. Use these manuals for reference.

Manual name	Cat. No.	Models	Applications	Description
NA-series Programmable Terminal Hardware User's Manual	V117	NA5-□□W□□□□	Learning the specifications and settings required to install an NA-series PT and connect peripheral devices.	Information is provided on NA-series PT specifications, part names, installation procedures, and procedures to connect an NA Unit to peripheral devices. Information is also provided on maintenance after operation and troubleshooting.
NA-series Programmable Terminal Hardware(-V1) User's Manual	V125	NA5-□□W□□□□-V1 (-P)	Learning the specifications and settings required to install an NA-series PT and connect peripheral devices.	Information is provided on NA-series PT specifications, part names, installation procedures, and procedures to connect an NA Unit to peripheral devices. Information is also provided on maintenance after operation and troubleshooting.
NA-series Programmable Terminal Hardware(-V2) User's Manual	V128	NA5-□□W□□□□-V2	Learning the specifications and settings required to install an NA-series PT and connect peripheral devices.	Information is provided on NA-series PT specifications, part names, installation procedures, and procedures to connect an NA Unit to peripheral devices. Information is also provided on maintenance after operation and troubleshooting.
NA-series Programmable Terminal Software User's Manual	V118	NA5-□□W□□□□ (-V□)	Learning about NA-series PT pages and object functions.	NA-series PT pages and object functions are described.
NA-series Programmable Terminal Device Connection User's Manual	V119	NA5-□□W□□□□ (-V□)	Learning the specifications required to connect devices to an NA-series PT.	Information is provided on connection procedures and setting procedures to connect an NA-series PT to a Controller or other device.
NA-series Programmable Terminal Soft-NA User's Manual	V126	NA-RTLD□□	Learning about the procedure to install the Soft-NA and differences from the NA5 series.	Information is provided on the specifications of the Soft-NA and differences from the NA5 series. Information is also provided on maintenance after operation and troubleshooting.
NA-series Programmable Terminal Startup Guide	V120	NA5-□□W□□□□	Learning in concrete terms information required to install and start the operation of an NA-series PT.	The part names and installation procedures are described followed by page creation and transfer procedures with the Sysmac Studio. Also operation, maintenance, and inspection procedures after the project is transferred are described. Sample screen captures are provided as examples.

Manual name	Cat. No.	Models	Applications	Description
NX-series CPU Unit Hardware User's Manual	W535	NX701-□□□□	Learning the basic specifications of the NX-series CPU Units, including introductory information, designing, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NX-series system is provided along with the following information on the CPU Unit. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection Use this manual together with the *NJ/NX-series CPU Unit Software User's Manual* (Cat. No.W501).
NJ-series CPU Unit Hardware User's Manual	W500	NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning the basic specifications of the NJ-series CPU Units, including introductory information, designing, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NJ-series system is provided along with the following information on a Controller built with a CPU Unit. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Inspection and maintenance Use this manual together with the *NJ-series CPU Unit Software User's Manual* (Cat. No. W501).
NJ/NX-series CPU Unit Built-in EtherNet/IP™ Port User's Manual	W506	NX701-□□□□ NX502-□□□□ NX102-□□□□ NX1P2-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Using the built-in EtherNet/IP port on an NJ/NX-series CPU Unit.	Information on the built-in EtherNet/IP port is provided. Information is provided on the basic setup, tag data links, and other features.
NJ/NX-series CPU Unit Software User's Manual	W501	NX701-□□□□ NX502-□□□□ NX102-□□□□ NX1P2-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning how to program and set up an NJ/NX-series CPU Unit. Mainly software information is provided.	Provides the following information on a Controller built with an NJ/NX-series CPU Unit. • CPU Unit operation • CPU Unit features • Initial settings • Programming based on IEC 61131-3 language specifications
NJ/NX-series Instructions Reference Manual	W502	NX701-□□□□ NX502-□□□□ NX102-□□□□ NX1P2-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning detailed specifications on the basic instructions of an NJ/NX-series CPU Unit.	The instructions in the instruction set (IEC 61131-3 specifications) are described.

Manual name	Cat. No.	Models	Applications	Description
NJ/NX-series Troubleshooting Manual	W503	NX701-□□□□ NX502-□□□□ NX102-□□□□ NX1P2-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning about the errors that may be detected in an NJ/NX-series Controller.	Concepts on managing errors that may be detected in an NJ/NX-series Controller and information on individual errors are described.
CJ Series Programmable Controllers Operation Manual	W393	CJ1H-CPU□□H-R CJ1G/H-CPU□□H CJ1G-CPU□□P CJ1M-CPU□□ CJ1G-CPU□□	Learning the basic specifications of the CJ-series PLCs, including introductory information, designing, installation, and maintenance.	The following information is provided on a CJ-series PLC. • Introduction and features • System configuration design • Installation and wiring • I/O memory allocation • Troubleshooting Use this manual together with the *Programming Manual* (Cat. No. W394).
CS/CJ/NSJ Series Programmable Controllers Operation Manual	W394	CS1G/H-CPU□□H CS1G/H-CPU□□-V1 CS1D-CPU□□H CS1D-CPU□□S CJ1H-CPU□□H-R CJ1G/H-CPU□□H CJ1G-CPU□□P CJ1M-CPU□□ CJ1G-CPU□□ NSJ□-□□□□(B)-G5D NSJ□-□□□□(B)-M3D	Learning about the functions of the CS/CJ-series and NSJ-series PLCs.	The following information is provided on a CS/CJ-series or NSJ-series PLC. • Programming • Master function • File memory • Other functions Use this manual together with the *Operation Manual* (CS-series PLCs: W339, CJ-series PLCs: W393).
CS/CJ/NSJ-series Instructions Reference Manual	W340	CS1□-CPU-□□□-□□ CJ1□-CPU-□□□-□□□ CJ2H-CPU-□□□-□□□ NSJ□□-□□□□□-□□□	Learning detailed information on programming instructions.	Instructions are described in detail. When programming, use this manual together with the *Operation Manual* (CS-series PLCs: W339, CJ-series PLCs: W393) and the *Programming Manual* (W394).
CS/CJ Series Programming Consoles Operation Manual	W341	CQM1H-PRO01 CQM1-PRO01 C200H-PRO27 +CS1W-KS001	Learning the operating procedures of the Programming Consoles.	The operating procedures of the Programming Consoles are described. When programming, use this manual together with the *Operation Manual* (CS-series PLCs: W339, CJ-series PLCs: W393), the *Programming Manual* (W394), and the *Instructions Reference Manual* (W340).

Manual name	Cat. No.	Models	Applications	Description
CS/CJ/NSJ Series Communications Commands Reference Manual	W342	CS1G/H-CPU□□H CS1G/H-CPU□□-V1 CS1D-CPU□□H CS1D-CPU□□S CS1W-SCU□□-V1 CS1W-SCB□□-V1 CJ1G/H-CPU□□H CJ1G-CPU□□P CJ1M-CPU□□ CJ1G-CPU□□ CJ1W-SCU□□-V1	Learning detailed specifications on the communications instructions addressed to CS/CJ-series CPU Units and NSJ-series PLCs.	1) C-mode commands and 2) FINS commands are described in detail. Refer to this manual for information on communications commands (C-mode commands and FINS commands) addressed to CPU Units. Note This manual describes communications commands that are addressed to a CPU Unit. The communications path is not relevant. (The communications commands can be sent through the serial communications port of the CPU Unit, the communications port of a Serial Communications Board/Unit, or a communications port on another Communications Unit.)
CJ-series CJ2 CPU Unit Hardware User's Manual	W472	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□	Learning the hardware specifications of CJ2 CPU Units.	The following information is provided on a CJ2 CPU Unit. • Introduction and features • Basic system configuration • Part names and functions • Installation and setting procedures • Troubleshooting Use this manual together with the *Software User's Manual* (Cat. No. W473).
CJ-series CJ2 CPU Unit Software User's Manual	W473	CJ2H-CPU6□-EIP CJ2H-CPU6□ CJ2M-CPU□□	Learning the software specifications of CJ2 CPU Units.	The following information is provided on a CJ2 CPU Unit. • CPU Unit operation • Internal memory • Programming • Settings • Functions built into the CPU Unit Use this manual together with the *Hardware User's Manual* (Cat. No. W472).
Ethernet Units Operation Manual Construction of Networks	W420	CS1W-ETN21 CJ1W-ETN21	Learning how to use an Ethernet Unit.	Information is provided on the Ethernet Units. Information is provided on the basic setup and FINS communications. Refer to the *Communications Commands Reference Manual* (Cat. No. W342) for details on FINS commands that can be sent to CS/CJ-series CPU Units when using the FINS communications service.
Ethernet Units Operation Manual Construction of Applications	W421	CS1W-ETN21 CJ1W-ETN21	Learning how to use an Ethernet Unit.	Information is provided on constructing host applications, including functions for sending/receiving mail, socket service, automatic clock adjustment, FTP server functions, and FINS communications.

Manual name	Cat. No.	Models	Applications	Description
CS/CJ-series EtherNet/IP™ Units Operation Manual	W465	CJ2H-CPU6□-EIP CJ2M-CPU3□ CS1W-EIP21 CJ1W-EIP21	Learning how to use the built-in EtherNet/IP port of the CJ2 CPU Units.	Information is provided on the built-in EtherNet/IP port and EtherNet/IP Units. Basic settings, tag data links, FINS communications, and other functions are described.
Sysmac Studio Version 1 Operation Manual	W504	SYSMAC-SE2□□□	Learning about the operating procedures and functions of the Sysmac Studio.	The operating procedures of the Sysmac Studio are described.
CX-Programmer Operation Manual	W446	CXONE-AL□□C-V4 CXONE-AL□□D-V4	Learning about the CX-Programmer except for information on function blocks, ST programming, and SFC programming.	The operating procedures of the CX-Programmer are described.
NY-Series Industrial Box PC User's Manual	W553	NYB□□-□1□□□	Learning the basic specifications of the NY-series Industrial Box PCs, including introductory information, designing, installation, and maintenance.	An introduction to the entire NY-series system is provided along with the following information on the Industrial Box PC. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection
NY-Series Industrial Panel PC User's Manual	W555	NYP□□-□1□□□-□□WC100□	Learning the basic specifications of the NY-series Industrial Panel PCs, including introductory information, designing, installation, and maintenance.	An introduction to the entire NY-series system is provided along with the following information on the Industrial Panel PC. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection
NY-Series IPC Machine Controller Industrial Box PC Hardware User's Manual	W556	NY512-□□□□	Learning the basic specifications of the NY-series Industrial Box PCs, including introductory information, designing, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NY-series system is provided along with the following information on the Industrial Box PC. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection

Manual name	Cat. No.	Models	Applications	Description
NY-Series IPC Machine Controller Industrial Panel PC Hardware User's Manual	W557	NY532-□□□□	Learning the basic specifications of the NY-series Industrial Panel PCs, including introductory information, designing, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NY-series system is provided along with the following information on the Industrial Panel PC. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection
NY-Series IPC Machine Controller Industrial Panel PC / Industrial Box PC Software User's Manual	W558	NY532-□□□□ NY512-□□□□	Learning how to program and set up the Controller functions of an NY-series Industrial PC.	The following information is provided on the NY-series Controller functions. • Controller operation • Controller features • Controller settings • Programming based on IEC 61131-3 language specifications
NY-Series Instructions Reference Manual	W560	NY532-□□□□ NY512-□□□□	Learning detailed specifications on the basic instructions of an NY-series Industrial PC.	The instructions in the instruction set (IEC 61131-3 specifications) are described.
NY-Series Troubleshooting Manual	W564	NY532-□□□□ NY512-□□□□	Learning about the errors that may be detected in an NY-series Industrial PC.	Concepts on managing errors that may be detected in an NY-series Controller and information on individual errors are described.
NX-series NX1P2 CPU Unit Hardware User's Manual	W578	NX1P2-□□□□	Learning the basic specifications of the NX-series NX1P2 CPU Units, including introductory information, designing, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NX1P system is provided along with the following information on the NX1P2 CPU Unit. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection

Manual name	Cat. No.	Models	Applications	Description
NX-series NX1P2 CPU Unit Built-in I/O and Option Board User's Manual	W579	NX1P2-□□□□	Learning about the details of functions only for an NX-series NX1P2 CPU Unit and an introduction of functions for an NJ/NX-series CPU Unit.	Of the functions for an NX1P2 CPU Unit, the following information is provided. • Built-in I/O • Serial Option Boards • Analog Option Boards An introduction of following functions for an NJ/NX-series CPU Unit is also provided. • Motion control functions • EtherNet/IP communications functions • EtherCAT communications functions
NX-series NX102 CPU Unit Hardware User's Manual	W593	NX102- □□□□	Learning the basic specifications of NX102 CPU Units, including introductory information, design, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NX102 system is provided along with the following information on the CPU Unit. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection
NX-series Safety Control Unit / Communication Control Unit User's Manual	Z395	NX-SL5□□□ NX-SI□□□□ NX-SO□□□□ NX-CSG□□□	Learning how to use the NX-series Safety Control Units and Communications Control Units.	Describes the hardware, setup methods, and functions of the NX-series Safety Control Units and Communications Control Units.
NX-series Communication Control Unit Built-in Function User's Manual	Z396	NX-CSG□□□	Learning about the built-in functions of an NX-series Communications Control Unit.	Describes the software setup methods and communications functions of an NX-series Communications Control Unit.
NJ-series Robot Integrated CPU Unit User's Manual	O037	NJ501-R□□□	Using the NJ-series Robot Integrated CPU Unit.	Describes the settings and operation of the CPU Unit and programming concepts for OMRON robot control.
CS Series Programmable Controllers Operation Manual	W339	CS1G-CPU□□H CS1H-CPU□□H	Learning the basic specifications of the CS-series PLCs, including introductory information, designing, installation, and maintenance.	The following information is provided on a CS-series PLC. • Introduction and features • System configuration design • Installation and wiring • I/O memory allocation • Troubleshooting Use this manual together with the Programming Manual (Cat. No. W394).

Manual name	Cat. No.	Models	Applications	Description
CS Series Duplex System Operation Manual	W405	CS1D-CPU□□H(A) CS1D-CPU□□S(A)	Learning the basic specifications of the CS-series Duplex System, including introductory information, designing, installation, and maintenance.	The following information is provided on a CS-series Duplex System. • Introduction and features • System configuration design • Installation and wiring • I/O memory allocation • Troubleshooting Use this manual together with the Programming Manual (Cat. No. W394).
CP Series CP1H CPU Unit Operation Manual	W450	CP1H-□□□□□-□	Learning the basic specifications of the CP1H CPU Unit, including introductory information, designing, installation, and maintenance.	Provide the following information on the CP Series: • Overview, design, installation, maintenance, and other basic specifications • Features • System configuration • Mounting and wiring • I/O memory allocation • Troubleshooting Use this manual together with the CP1H Programmable Controllers Programming Manual (W451).
CP Series CP1L CPU Unit Operation Manual	W462	CP1L-L□□□□-□ CP1L-M□□□□-□	Learning the basic specifications of the CP1L CPU Unit, including introductory information, designing, installation, and maintenance.	
CP Series CP1L-EL/EM CPU Unit Operation Manual	W516	CP1L-E□□□□-□	Learning the basic specifications of theCP1L-EL/EM CPU Unit, including introductory information, designing, installation, and maintenance.	Provides the following information on the CP Series: • Overview, design, installation, maintenance, and other basic specifications • Features • System configuration • Mounting and wiring • I/O memory allocation • Troubleshooting Use this manual together with the CP1L Programmable Controllers Programming Manual (W451).
CP Series CP1H and CP1L CPU Unit Programming Manual	W451	CP1H-□□□□□-□ CP1L-□□□□□-□	Learning about the functions of the CP Series CP1H CPU Units.	Provides the following information on the CP Series: • Programming instructions • Programming methods • Tasks • File memory • Functions Use this manual together with the CP Series CP1H CPU Units Operation Manual (W450) and CP Series CP1L CPU Units Operation Manual (W462).

Manual name	Cat. No.	Models	Applications	Description
CP Series CP2E CPU Unit Hardware User's Manual	W613	CP2E-□□□□□-□	To learn the hardware specifications of the CP Series CP2E CPU Unit.	Describes the following information for CP2E PLCs. • Overview and features • Basic system configuration • Part names and functions • Installation and settings • Troubleshooting Use this manual together with the CP2E CPU Unit Software User's Manual (Cat. No. W614) and Instructions Reference Manual (Cat. No.W483).
CP Series CP2E CPU Unit Software User's Manual	W614	CP2E-□□□□□-□	To learn the software specifications of the CP Series CP2E CPU Unit.	Describes the following information for CP2E PLCs. • CPU Unit operation • Internal memory • Programming • Settings • CPU Unit built-in functions • Interrupts • High-speed counter inputs • Pulse outputs • Serial communications • Ethernet • Other functions Use this manual together with the CP2E CPU Unit Hardware User's Manual (Cat. No. W613) and Instructions Reference Manual (Cat. No.W483).
CP Series CP1E/CP2E CPU Unit Instructions Reference Manual	W483	CP2E-□□□□□-□	To learn programming instructions in detail.	Describes each programming instruction indetail. When programming, use this manual together with the CP2E CPU Unit Software User's Manual (Cat. No. W614).
CS/CJ Series Serial Communications Boards/Units Operation Manual	W336	CS1W-SCB□1-V1 CS1W-SCU□1-V1 CJ1W-SCU□1-V1 CJ1W-SCU□2	To learn the specifications of the hardware and serial communication mode of a serial communication board/unit.	Describes the use of Serial Communications Unit and Boards to perform serial communications with external devices, including the use of standard system protocols for OMRON products. Note Refer to the CS/CJ Series Communications Commands Reference Manual (W342) for details on sending commands in host link mode from a Serial Communications Board or Unit's port.

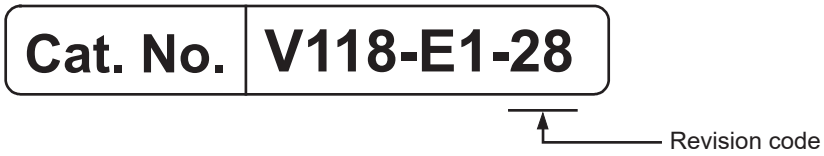
Manual name	Cat. No.	Models	Applications	Description
NX-series NX502 CPU Unit Hardware User's Manual	W629	NX502- □□□□	Learning the basic specifications of NX502 CPU Units, including introductory information, design, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NX502 system is provided along with the following information on the CPU Unit. • Features and system configuration • Introduction • Part names and functions • General specifications • Installation and wiring • Maintenance and inspection

Terminology

Term	Description
HMI	A general term for interface devices that indicates both hardware and software elements. In this manual, “HMI” refers to an OMRON Sysmac-brand product unless otherwise specified.
PT	The hardware elements of the HMI.
NA Series	The NA Series of Programmable Terminals and peripheral devices.
NA5 Series	NA5-□□W□□□□ (-V□).
HMI Project	A Sysmac Studio project for an HMI.
NA Unit	An NA-series Programmable Terminal.
Download	Transferring data from the Sysmac Studio to an HMI.
Upload	Transferring the project from an HMI to the Sysmac Studio.
IAG collection	When you provide IAGs, you provide them as IAG collections. IAGs are also imported as IAG collections. An IAG collection contains one or more IAGs.

Revision History

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



Revision code	Date	Revised content
01	June 2014	Original production
02	October 2014	Made revisions accompanying version upgrade.
03	April 2015	Made revisions accompanying version upgrade.
04	October 2015	Made revisions accompanying version upgrade.
05	December 2015	Made revisions accompanying version upgrade.
06	April 2016	Made revisions accompanying version upgrade.
07	July 2016	Made revisions accompanying version upgrade.
08	October 2016	Made revisions accompanying support of NX1/NY series.
09	February 2017	Made revisions accompanying version upgrade.
10	October 2017	Made revisions accompanying version upgrade.
11	April 2018	Made revisions accompanying support of NX102 series.
12	July 2018	Made revisions accompanying version upgrade.
13	October 2018	Made revisions accompanying version upgrade.
14	January 2019	Made revisions accompanying version upgrade.
15	April 2019	Made revisions accompanying version upgrade.
16	April 2020	Made revisions accompanying version upgrade and the addition of units.
17	July 2020	Made revisions accompanying version upgrade.
18	October 2020	Made revisions accompanying version upgrade.
19	April 2021	Made revisions accompanying version upgrade.
20	July 2021	Made revisions accompanying version upgrade.
21	October 2021	Made revisions accompanying version upgrade.
22	July 2022	Made revisions accompanying support for secure communication with the NJ/NX series.
23	October 2022	Revisions for adding safety precautions regarding security.
24	April 2023	Made revisions accompanying version upgrade.
25	July 2023	Made revisions accompanying version upgrade.
26	October 2025	Made revisions accompanying version upgrade.
27	January 2026	Made revisions accompanying support of NA5-□□W□□□□-V2.
28	July 2026	Made revisions accompanying version upgrade.

1

Introduction to the NA-series Programmable Terminals

This section describes the features, basic system configuration, specifications, and overall operating procedure of the NA-series Programmable Terminals.

1-1 NA-series Programmable Terminals 1-2

1-1-1 Features 1-2

1-2 How HMIs Operate 1-4

1-2-1 HMI Software Configuration 1-4

1-2-2 HMI Projects 1-4

1-2-3 Pages 1-4

1-2-4 Objects 1-5

1-2-5 Memory Specifications for Connected Devices 1-6

1-2-6 Events 1-7

1-2-7 Subroutines 1-8

1-2-8 Functions Shared by the Entire HMI Project 1-9

1-2-9 Data That Retained When Power Is Turned OFF 1-9

1-3 Operating Procedure for HMIs 1-10

1-3-1 Overall Procedure 1-10

1-3-2 Procedure Details 1-11

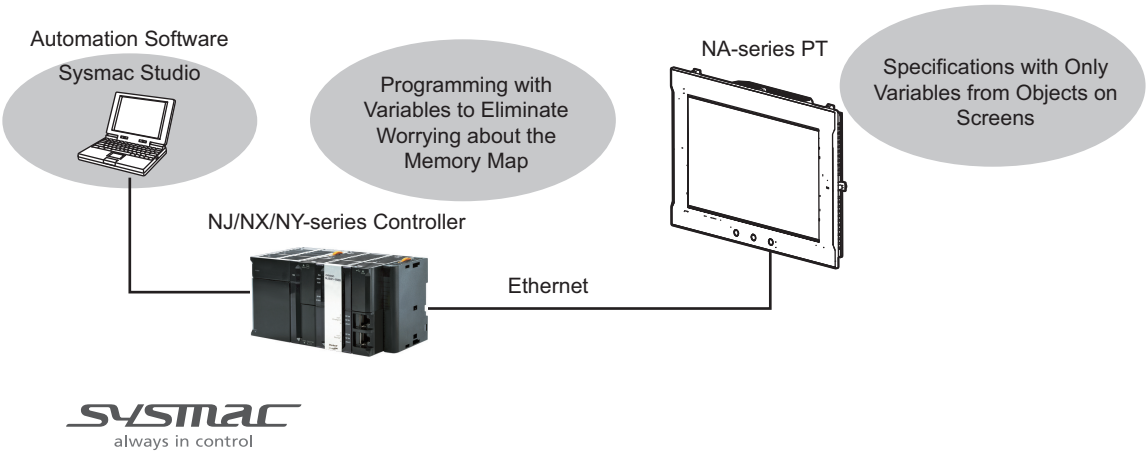
1-1 NA-series Programmable Terminals

The NA-series Programmable Terminals represent the next generation of HMIs for industrial applications. They display information on FA manufacturing sites and function as control interfaces while providing safety, reliability, and maintainability. They provide all of the functions of traditional programmable terminals with a clearer, easy-to-use interface.

OMRON offers the new Sysmac Series of control devices designed with unified communications specifications and user interface specifications.

The NA-series Programmable Terminals are Sysmac devices that you can use together with the NJ/NX/NY-series Machine Automation Controllers and the Sysmac Studio Automation Software to achieve optimum functionality and ease of operation.

If you connect an NA-series Programmable Terminal to an NJ/NX/NY-series Controller, all you have to do to specify memory in the Controller is to specify the Controller variables for the objects on the Programmable Terminal screens. This allows you to create screens without being concerned with the memory map of the Controller.



1-1-1 Features

Hardware Features

● High-resolution Display Panels

High-resolution display panels are used to more clearly display large amounts of information than was possible with previous OMRON products.

● Two Ethernet Ports (Standard Feature)

You can use both Ethernet ports to separate the segment attached to control devices from the segment attached to maintenance devices. Access is possible from both segments at the same time.

You can connect the following devices.

- NJ/NX/NY-series Controllers
- PLCs
- Computers
- Sysmac Studio

Standard-feature SD Memory Card Slot

You can use an SD Memory Card inserted in the NA Unit to transfer the project you created on the Sysmac Studio to the NA Unit, to update the system program in the NA Unit, or to save the log data from the NA Unit.

Software Features

Specifications with Variables for Superior Reusability

If you connect to an NJ/NX/NY-series Controller, all you have to do to specify memory in the Controller is to specify the Controller variables. This allows you to create objects that are not dependent on specific devices or memory maps. This in turn makes the objects much more reusable than they were with previous PTs.

Program with Visual Basic

You can use Microsoft's Visual Basic to program advanced functions that you cannot achieve with standard objects.

A Wealth of Security Features

The many security features of the NA-series PTs include operation authority settings and execution restrictions with IDs.

Use the Integrated Development Environment of Sysmac Studio Automation Software

You use the Sysmac Studio to create applications for the NA-series Programmable Terminals.

The Sysmac Studio provides an integrated development environment that covers not only the NA-series Programmable Terminal, but also the Controller and devices on EtherCAT as well.

You can use consistent procedures for all devices regardless of differences in the devices. The Sysmac Studio supports all phases of Controller application, from page creation and sequence design through debugging, simulations, commissioning, and changes during operation.

A Wealth of Simulation Features

You can perform simulations using a virtual HMI on the Sysmac Studio. And you can also perform online debugging with a virtual NJ/NX/NY-series Controller.

1-2 How HMIs Operate

This section describes how the HMI operates.

1-2-1 HMI Software Configuration

An HMI consists of the following software.

- System Program

  The system program is required to start the HMI and execute the runtime. For details, refer to the following manuals.

  • *NA-series Programmable Terminal Hardware User's Manual* (V117)

  • *NA-series Programmable Terminal Hardware(-V1) User's Manual* (V125)

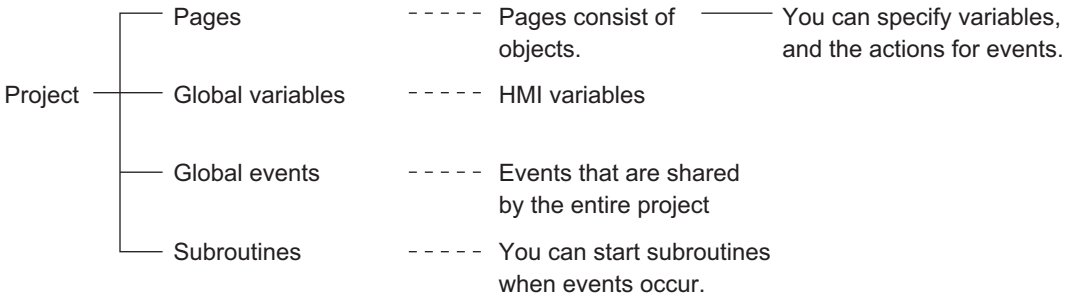
  • *NA-series Programmable Terminal Hardware(-V2) User's Manual* (V128)
- Runtime

  The runtime is the middleware that executes the project. The runtime is started by the system program and it manages execution of the project.
- Project

  You use the Sysmac Studio to create your applications. The applications are executed on the runtime.

1-2-2 HMI Projects

An HMI project contains mainly the following data.

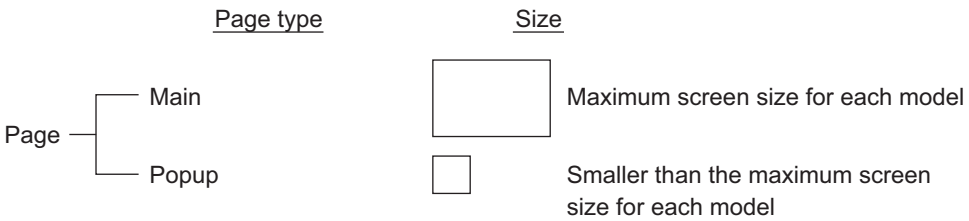


In addition, there is data that is shared by the entire project, such as user alarms, data logging, recipes, and resources.

1-2-3 Pages

One HMI screen is called a page.

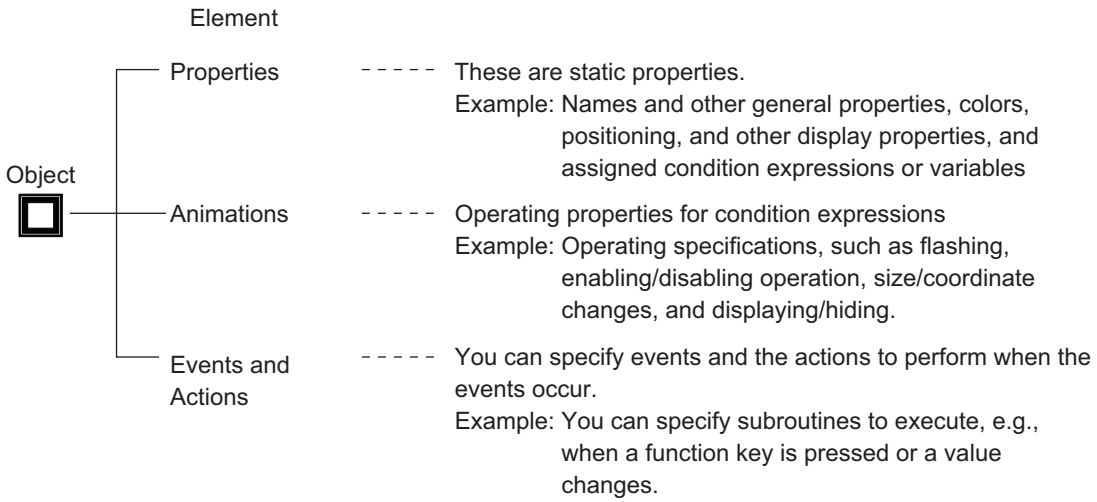
There are the following two types of pages.



You paste objects on the pages.

1-2-4 Objects

The objects that you paste on HMI pages consist of the following three elements.



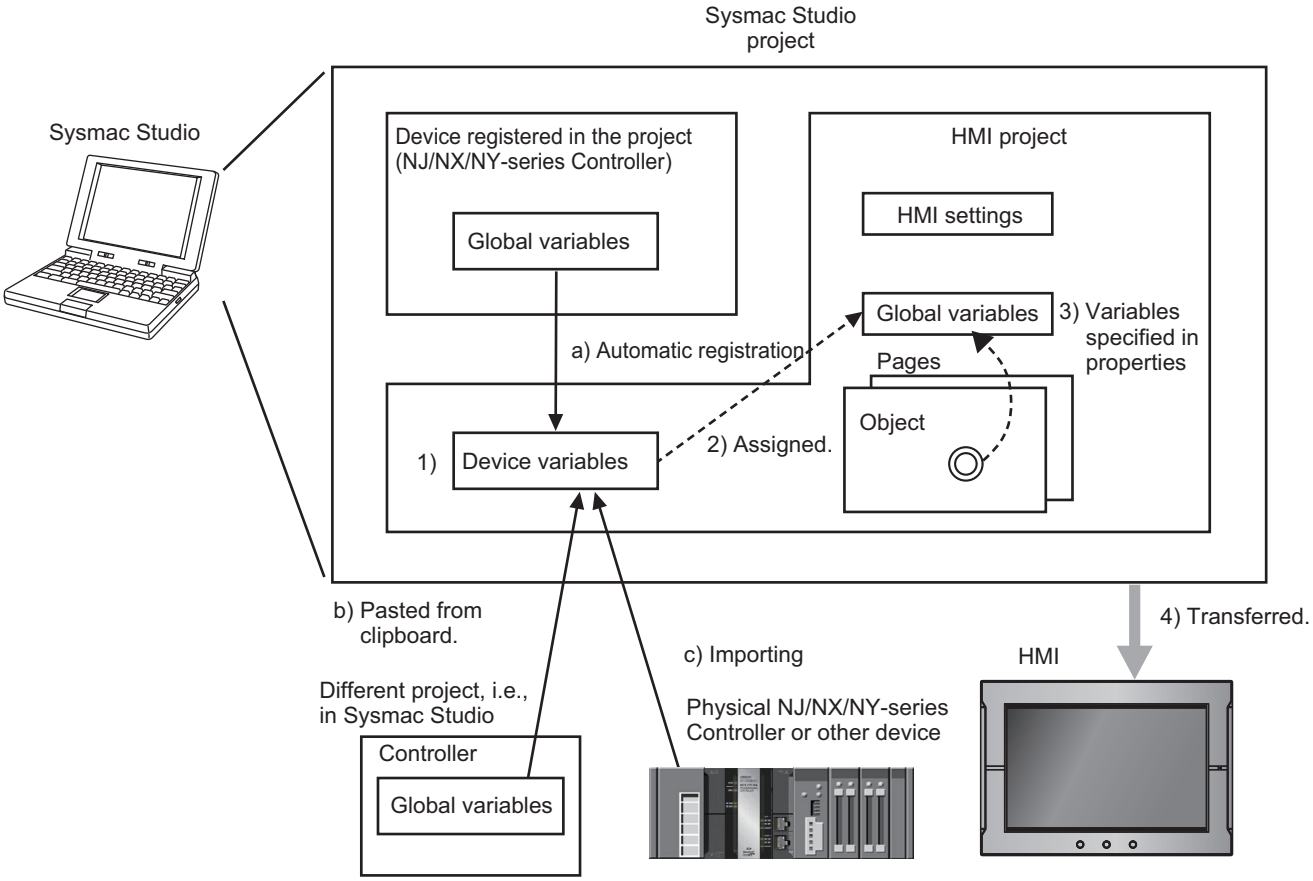
1-2-5 Memory Specifications for Connected Devices

Overview

You use HMI global variables to specify memory in a Controller or PLC.

You assign HMI global variables to connected device variables in advance to map them.

1) Variables for connected devices are registered to device variables in the HMI project with one of the following methods.
   a) Variables for connected devices that are registered in the same project are registered automatically.
   b) You can copy and paste variables from another project using the clipboard.
   c) You can import variables from the external connected device.
2) Devices variables are assigned to HMI global variables.
3) The assigned HMI global variables are specified in the properties of the objects.
4) Then, you transfer the project that you created to the HMI.



Refer to *4-1 Registering Variables* on page 4-3 for the details on HMI variables.

1-2-6 Events

Events are triggers that activate actions.[*1]

*1. Actions are various operations that can be directly assigned to events.

Events occur when the common page status or object status meets certain conditions.

Events are classified into three groups as shown below.

	Group	Description
Events	Global events	Events that occur for shared project status.
	Page and object events	Events that occur for specific page or object status.
	User alarm events	Events that occur for user alarm status.

1-2-7 Subroutines

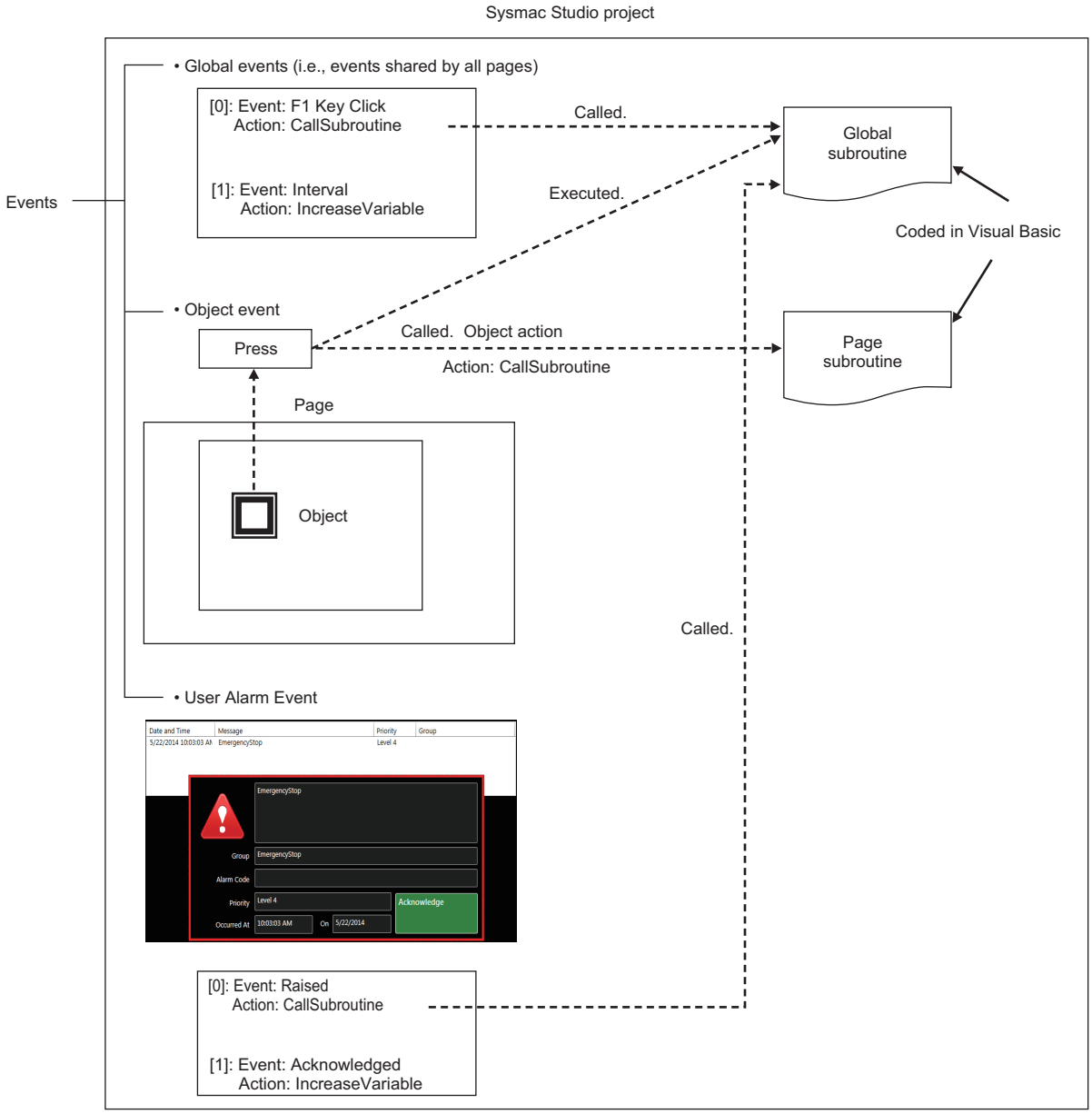
You can execute user-created subroutines in the HMI based on the following three types of conditions.

- When global events occur
- When events occur on pages or for objects
- When user alarm events occur

There are two types of subroutines that you can create.

- Global Subroutines
  You create these subroutines under the global subroutine item of the HMI project.
- Page Subroutines
  You create these subroutines with the page code editor.

You can use Visual Basic to write both the global subroutines and the page subroutines.



- You can call a global subroutine by executing the CallSubroutine action when a global event occurs.

- You can call a global subroutine or page subroutine by executing the CallSubroutine action when an event occurs on the page or for an object.
- You can call a global subroutine by executing the CallSubroutine action when a user alarm event occurs.

You can also call a global subroutine from another global subroutine or a page subroutine.

1-2-8 Functions Shared by the Entire HMI Project

In addition to global events, the following functions are shared by the entire HMI project.

Alarms

Alarms notify the user when certain conditions are met in the HMI.

The following alarms are supported.

- User alarms

Data Logging

You can log data to store the changes in the values of specified variables over time.

You can display the saved data with Trend Graph objects. You can also save this data to external files.

Recipes

A recipe is used to write data (numeric data or text strings) that was set in advance in the project to all of the specified variables as a group or to read all of the specified variables as a group.

You can manipulate the registered recipe data with Recipe Viewer objects.

Resources

You can manage resources, such as the text strings, movies, still images, and documents that are displayed for objects and alarms on pages.

1-2-9 Data That Retained When Power Is Turned OFF

The following data is retained when the power supply is turned OFF.

With No Battery or Low Battery Voltage

- Project data
- Log data that is not written to the SD Memory Card
- User alarm history
- Values of variables with Retain attribute
- Calibration information for touch panel

With Good Battery (in addition to the above)

- Clock information

1-3 Operating Procedure for HMIs

This section gives the operating procedure for an HMI and then describes it in more detail.

1-3-1 Overall Procedure

The overall procedure to use an HMI is given below.

STEP1

System Configuration and Project Design

Design the system configurations and project.

STEP 1-1 Designing the System Configurations
STEP 1-2 Designing the Project (Pages, Variables, Subroutines, etc.)

STEP2

Software Settings (Configurations and Setup) and Creating the HMI Application

Create the system configurations that you designed in step 1 on the Sysmac Studio. Also create the project (pages, variables, subroutines, etc.), build the project, and debug it with simulation and other functions.

Determining the Connected Device Variables and Mapping HMI Variables to Them (We recommend this as the basic procedure.)	Setting HMI Variables First and Then Mapping Them to Connected Device Variables
STEP 2-1 Starting the Sysmac Studio and Creating a Project	STEP 2-1 Starting the Sysmac Studio and Creating a Project
STEP 2-2 Software Settings (Configurations and Setup)	STEP 2-2 Creating the HMI Application
STEP 2-3 Creating the HMI Application	STEP 2-3 Software Settings (Configurations and Setup)
STEP 2-4 Building the HMI	STEP 2-4 Building the HMI
STEP 2-5 Offline Debugging	STEP 2-5 Offline Debugging

STEP3

Mounting and Wiring

Mount the HMI.

Connect the connected device and computer (Sysmac Studio) to the HMI.

STEP4

Confirming Operation and Starting Actual System Operation

Download the project from the Sysmac Studio.

Make the settings on the System Menu, check operation on the physical devices, and start operation.

1-3-2 Procedure Details

STEP1 System Configuration and Project Design

Step	Description	Reference
STEP 1-1 Designing the System Configurations	• Connect an HMI to the external device. • Connect an HMI to the Sysmac Studio.	*Section 2 Configuration Units* in *NA Series Programmable Terminal Hardware User's Manual* (V117) *Section 2 Configuration Units in NA Series Programmable Terminal Hardware(-V1) User's Manual* (V125) *Section 2 Configuration Units in NA-series Programmable Terminal Hardware(-V2) User's Manual* (V128) *NA-series Programmable Terminal Device Connection User's Manual* (V119)
STEP 1-2 Designing the Project (Pages, Variables, Subroutines, Etc.)	Design the project as given below.	*NA Series Programmable Terminal Software User's Manual* (V118)
1) Designing the Pages and Subroutines	• Design the contents to display on the pages (the pages and objects to use). • Design the execution methods and contents of the subroutines.	
2) Designing Items Shared by All Pages	• Design the global events. • Design the alarms, recipes, data logging, and other functions.	
3) Variable Design	• HMI external variable design: Design the mappings between the connected device variables and the HMI global variables. • Design the HMI internal variables and subroutine variables. • Define the attributes of the above variables, such as the Data Type, Name, and Retain attributes.	

STEP2 Software Settings (Configurations and Setup) and Creating the HMI Application

Step	Description	Sysmac Studio operations	Reference
STEP 2-1 Starting the Sysmac Studio and Creating a Project	1. Start the Sysmac Studio and create a project. 2. Insert the HMI.	Press the **New Project** Button. Use ***HMI*** on the Insert Menu.	*Sysmac Studio Version 1 Operation Manual* (W504) *Section 2 Basic Sysmac Studio Operations*

Determining the Connected Device Variables and Mapping HMI Variables to Them

We recommend this procedure as the basic procedure.

- Connecting to a Connected Device Registered in the Current Project

STEP 2-2 Software Settings (Configurations and Setup)	Make the initial software settings on the Sysmac Studio.	Use **Configurations and Setup** in the Multiview Explorer of the Sysmac Studio.	*Section 3 HMI Configuration and Setup*
• **Mapping Variables**	• Assign global variables to connected device variables (mapping variables).	Map the variables under **Configurations and Setup** – **Variable Mapping**.	*3-3 Mapping Variables* on page 3-7
• **HMI Settings**	• Set the parameters related to the HMI. Startup Page, Brightness Settings, IP Address, FTP Settings, NTP Settings, FINS Settings, VNC Settings, etc.	Make the settings under **Configurations and Setup** – **HMI Settings**.	*3-4 HMI Settings on page 3-11*
• **Security Settings and Language Settings**	• Set the operation rights to the HMI and the language to display on the HMI.	Make the settings under **Configurations and Setup** – **Security Settings** and **Configurations and Setup** – **Language Settings**.	*3-5 Security Settings on page 3-21* *3-7 Language Settings on page 3-23*

- Connecting to a Connected Device Not Registered in the Current Project

STEP 2-2 Software Settings (Configurations and Setup)	Make the initial software settings on the Sysmac Studio.	Use **Configurations and Setup** in the Multiview Explorer of the Sysmac Studio.	*Section 3 HMI Configuration and Setup*
• **Device Settings**	• Register the external connected devices.	Add the connected device under **Configurations and Setup** – **Device References**.	*3-2 Device References* on page 3-3
	• You can do either of the following. a) Importing Variables from the Actual Connected Device: Place the Sysmac Studio online with the connected device and import the variables from the connected device. b) Importing Variables from Another Project: Copy the variable table in the other project and paste it in the variable table for the connected device to import the variables. c) Importing Variables from a File: Import the information on the variables of the connected device from a file, such as an Excel file.	Set up communications and import the variables. Copy the variable table from another project using the clipboard. Import variable information from a file.	

• **Mapping Variables**	Assign global variables to connected device variables (mapping variables).	Map the variables under **Configurations and Setup** – **Variable Mapping**.	*Section 3 HMI Configuration and Setup* and *3-3 Mapping Variables* on page 3-7
• **HMI Settings**	• Set the parameters related to the HMI. Startup Page, Brightness Settings, IP Address, FTP Settings, NTP Settings, FINS Settings, VNC Settings, etc.	Make the settings under **Configurations and Setup** – **HMI Settings**.	*3-4 HMI Settings on page 3-11*
• **Security Settings and Language Settings**	• Set the operation rights to the HMI and the language to display on the HMI.	Make the settings under **Configurations and Setup** – **Security Settings** and **Configurations and Setup** – **Language Settings**.	*3-5 Security Settings on page 3-21* *3-7 Language Settings on page 3-23*

STEP 2-3 Creating the HMI Application	Create the application (pages, variables, subroutines, etc.) with the Sysmac Studio.	Use **HMI** in the Multiview Explorer of the Sysmac Studio.	*Section 4 Creating the HMI Application*
1) Registering Variables	• Register the variables in the HMI global variable table with the Sysmac Studio. Note: Variables that were mapped in step 2-2 are automatically registered in the HMI global variables table.	Use the editor for **HMI** – **Data** – **Global Variables**	*4-1 Registering Variables on page 4-3*
2) Creating Pages	• Paste the objects on each page and set the object properties and other settings.	Use the editor for **HMI** – **Pages**.	*4-3 Creating Pages on page 4-22* *Section 5 Objects*
3) Creating Subroutines	• Create the subroutines. You can create the following. • Global subroutines • Page subroutines	Subroutines shared by the entire project: Select **HMI** – **Global Subroutine**. Page subroutines: Use **HMI** – **Pages | Page Name** and select **View Code Editor** from the individual pages.	*4-5 Subroutines on page 4-56*
4) Settings Shared by All Pages	• Make the settings that are shared by the project: alarms, recipes, data logging, global events, etc.	Use **HMI** – **User Alarms**, **HMI** – **Recipes**, etc.	*4-4 Setting Common Object Functions on page 4-35*

STEP 2-4 Building the HMI	• Convert the HMI project into a form that the HMI can execute.	Use ***Build HMI*** on the Project Menu.	*4-11 Building on page 4-80*

STEP 2-5 Offline Debugging	• Check the operation of the pages on the Simulator (a virtual HMI).	Use ***Start NA Simulation*** or ***Run with Controller Simulator*** on the Simulation Menu.	*Section 7 Debugging*

Setting HMI Variables First and Then Mapping Them to Connected Device Variables

STEP 2-2 Creating the HMI Application	Create the application (pages, variables, subroutines, etc.) with the Sysmac Studio.	Use **HMI** in the Multiview Explorer of the Sysmac Studio.	*Section 4 Creating the HMI Application*
1) Registering Variables	Register the variables in the HMI global variable table with the Sysmac Studio.	HMI – Data – Global Variables Table	*4-1 Registering Variables on page 4-3*
2) Creating Pages	Paste the objects on each page and set the object properties.	Use the editor for **HMI** – **Pages**.	*4-3 Creating Pages on page 4-22* *Section 5 Objects*
3) Creating Subroutines	Create the subroutines. You can create the following. • Global subroutines • Page subroutines	Subroutines shared by the entire project: Select **HMI** – **Global Subroutine** – **Add**. Page subroutines: Use **HMI** – **Pages** and select **View Code Editor** from the individual pages.	*4-5 Subroutines on page 4-56*
4) Settings Shared by All Pages	Make the settings that are shared by the project: alarms, recipes, data logging, global events, etc.	Use **HMI** – **User Alarms**, **HMI** – **Recipes**, etc.	*4-4 Setting Common Object Functions on page 4-35*

- Connecting to a Connected Device Registered in the Current Project

STEP 2-3 Software Settings (Configurations and Setup)	Make the initial software settings on the Sysmac Studio.	Use **Configurations and Setup** in the Multiview Explorer of the Sysmac Studio.	*Section 3 HMI Configuration and Setup*
• **Mapping Variables**	• Assign global variables to connected device variables (mapping variables).	Map the variables under **Configurations and Setup** – **Variable Mapping**.	*3-3 Mapping Variables* on page 3-7
• **HMI Settings**	• Set the parameters related to the HMI. Startup Page, Brightness Settings, IP Address, FTP Settings, NTP Settings, FINS Settings, VNC Settings, etc.	Make the settings under **Configurations and Setup** – **HMI Settings**.	*3-4 HMI Settings on page 3-11*
• **Security Settings and Language Settings**	• Set the operation rights to the HMI and the language to display on the HMI.	Make the settings under **Configurations and Setup** – **Security Settings** and **Configurations and Setup** – **Language Settings**.	*3-5 Security Settings on page 3-21* *3-7 Language Settings on page 3-23*

- Connecting to a Connected Device Not Registered in the Current Project

STEP2-3 Software Settings (Configurations and Setup)	Make the initial software settings on the Sysmac Studio.	Use **Configurations and Setup** in the Multiview Explorer of the Sysmac Studio.	*Section 3 HMI Configuration and Setup*
• **Connected Device Settings**	• Register the external connected devices.	Add the connected device under **Configurations and Setup** – **Device References**.	*3-2 Device References* on page 3-3
	• You can do either of the following. a) Importing Variables from the Unit Connected Device: Place the Sysmac Studio online with the connected device and import the variables from the connected device. b) Importing Variables from Another Project: Copy the variable table in the other project and paste it in the device variable table. c) Importing Variables from a File: Import the information on the variables of the connected device from a file, such as an Excel file.	Set up communications and import the variables. Copy the variable table from another project using the clipboard. Import variable information from a file.	*3-2-2 Connected Devices in the Current Project on page 3-3*
	• Assign global variables to connected device variables (mapping variables).	Map the variables under **Configurations and Setup** – **Variable Mapping**.	*3-3 Mapping Variables on page 3-7*
• **HMI Settings**	• Set the parameters related to the HMI. Startup Page, Brightness Settings, IP Address, FTP Settings, NTP Settings, FINS Settings, VNC Settings, etc.	Make the settings under **Configurations and Setup** – **HMI Settings**.	*3-4 HMI Settings on page 3-11*
• **Security Settings and Language Settings**	• Set the operation rights to the HMI and the language to display on the HMI.	Make the settings under **Configurations and Setup** – **Security Settings** and **Configurations and Setup** – **Language Settings.**	*3-5 Security Settings on page 3-21* *3-7 Language Settings on page 3-23*

STEP 2-4 Building the HMI	• Convert the HMI project into a form that the HMI can execute.	Use *Build HMI* on the Project Menu.	*4-11 Building on page 4-80*

STEP2-5 Offline Debugging	• Check the operation of the pages on the Simulator (a virtual HMI).	Use ***Start NA Simulation*** or ***Run with Controller Simulator*** on the Simulation Menu.	*Section 7 Debugging*

STEP3 Mounting and Wiring

Step	Description	Reference
1) Mounting	• Mount the HMI to the panel. • Wire Power Supply.	*3-3 Installing NA Units* in *NA Series Programmable Terminal Hardware User's Manual* (V117) *3-3 Installing NA Units in NA Series Programmable Terminal Hardware(-V1) User's Manual* (V125) *3-3 Installing NA Units in NA-series Programmable Terminal Hardware(-V2) User's Manual* (V128)
2) Wiring the Ethernet Cable to the Connected Device	• Wire the Ethernet cables.	*3-4 Wiring Methods* in *NA Series Programmable Terminal Hardware User's Manual* (V117) *3-4 Wiring Methods in NA Series Programmable Terminal Hardware(-V1) User's Manual* (V125) *3-4 Wiring Methods in NA-series Programmable Terminal Hardware(-V2) User's Manual* (V128) *NA-series Programmable Terminal Device Connection User's Manual* (V119)
3) Connecting the Computer (Sysmac Studio)	• Wire the USB cable. or • Wire the Ethernet cable.	*2-4 Support Software* in *NA Series Programmable Terminal Hardware User's Manual* (V117) *2-4 Support Software in NA Series Programmable Terminal Hardware(-V1) User's Manual* (V125) *2-4 Support Software in NA Series Programmable Terminal Hardware(-V2) User's Manual* (V128)

STEP4 Confirming Operation and Starting Actual System Operation

Step	Description	Sysmac Studio operations	Reference
1) Online Connection to Sysmac Studio and Project Download	• Turn ON the power supply to the HMI and place the Sysmac Studio online. Then, download the project.*1 *1.Use the Synchronize operation of the Sysmac Studio to download the project.	Use ***Communications Setup*** on the HMI Menu. Use ***Synchronization*** on the HMI Menu.	*Section 6 Connecting to the HMI* and *Section 8 Transfer*
2) Operation Check on NA Unit	• Integrate the NA Unit into the actual system, manipulate the project that you created and confirm the following: that correct values are written to the connected device, that the pages change correctly, and that values set at the connected device are updated.	---	*Section 7 Debugging*
3) Actual System Operation	• Start actual operation.	---	---

2

Basic Sysmac Studio Operations

This section describes basic operations on the Sysmac Studio.

2-1 Parts of the Sysmac Studio Window 2-2
2-1-1 Application Window 2-2
2-2 Menu Command Structure 2-7
2-3 Basic Editing Operations 2-10
2-4 Sysmac Studio Settings and Operations 2-12
2-4-1 Setting Parameters 2-12
2-4-2 Programming 2-12
2-4-3 Library Functions 2-13
2-4-4 Operations for Debugging 2-13
2-4-5 Communications 2-13
2-4-6 Security Measures 2-14
2-4-7 Online Help 2-14
2-4-8 Project Management Functions 2-14
2-5 Basic Operations for HMI Projects 2-15
2-5-1 Creating a Project File from the Start Page 2-15
2-5-2 Adding an HMI to an Existing Project 2-16
2-5-3 Changing Devices 2-17
2-5-4 Importing and Exporting Devices 2-19

2-1 Parts of the Sysmac Studio Window

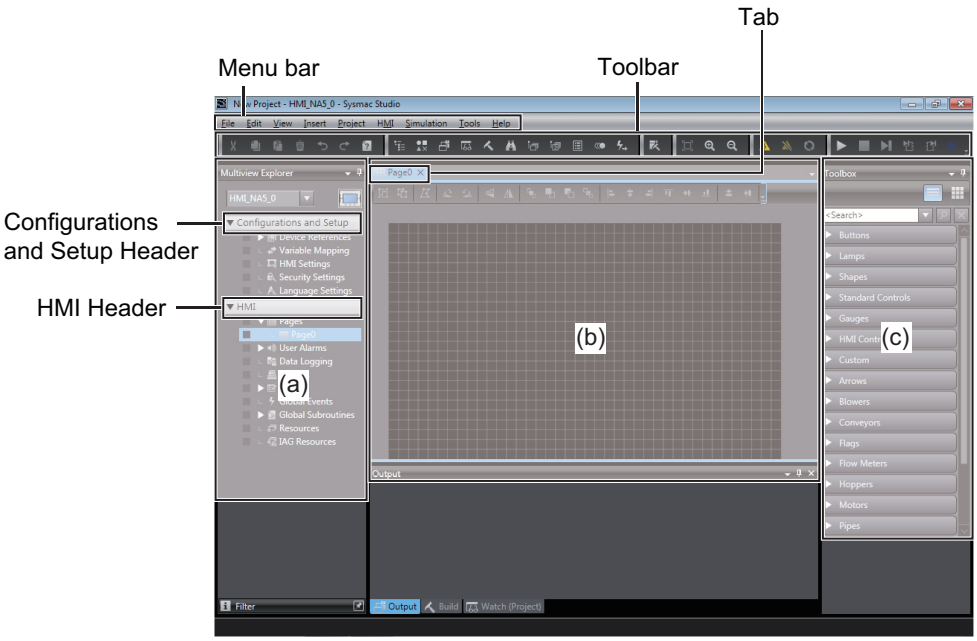
This section gives the names and functions of the parts of the Sysmac Studio Window.

This manual describes only functions that apply when an HMI is selected for the device. For information on Sysmac Studio functions not described in this manual, refer to the *Sysmac Studio Operation Version 1 Manual* (Cat. No. W504).

Precautions for Correct Use

When you use the Sysmac Studio, use the standard Windows desktop theme. If you do not use the standard Windows desktop theme, part of the display may not be correct.

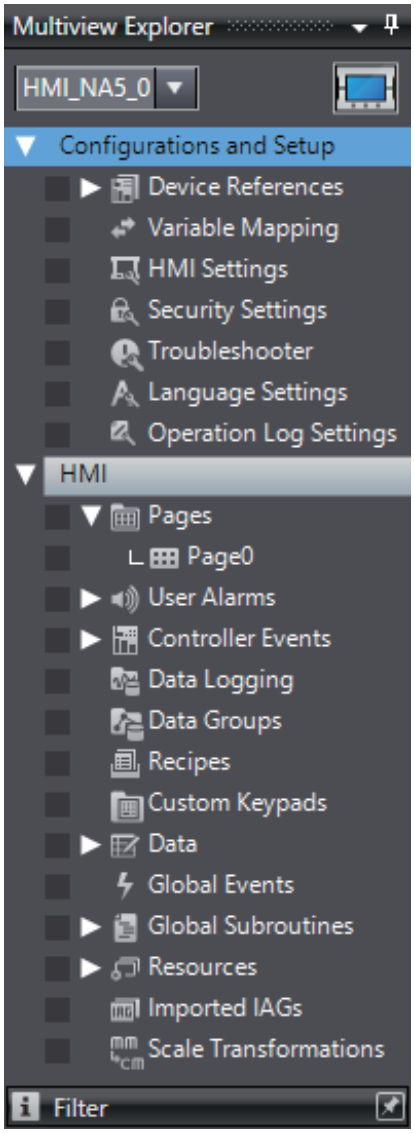
2-1-1 Application Window



Number	Name
(a)	Multiview Explorer
(b)	Edit Pane
(c)	Toolbox

The functions of these parts are described starting on the next page.

Multiview Explorer (a)



- This pane is your access point for all Sysmac Studio data. When an HMI is selected, it is divided into a Configurations and Setup Layer and an HMI Layer.
- You can also display the Page Explorer to display lists of objects on pages or the Code Explorer to display lists of subroutines.

Layers and Items in the Multiview Explorer

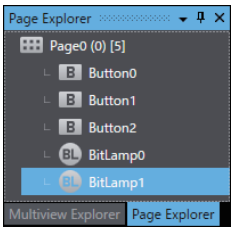
Configurations and Setup
Device References
Variable Mapping
HMI Settings
Security Settings
Troubleshooter
Language Settings
Operation Log Settings

HMI
Pages Page
User Alarms Group
Controller Events User Events
Data Logging DataSet
Data Groups DataGroup
Recipes Recipe
Custom Keypads Group
Data Data Types Global Variables
Global Events
Global Subroutines SubroutineGroup
Resources Root
Imported IAGs IAG
Scale Transformations

Precautions for Correct Use

You cannot download the data to the HMI if an error icon is displayed.

Page Explorer



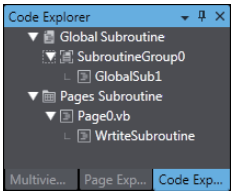
The Page Explorer displays a list of objects on a page. If you click an object in the Page Explorer, the object will be selected on the Edit Pane.

To change the attributes for grouped objects or for individual objects in IAGs, select the individual objects on the Page Explorer.

The page index and number of objects in the page are displayed at the topmost part of the tree in the following format.

Name of page (page index) [number of objects]

Code Explorer

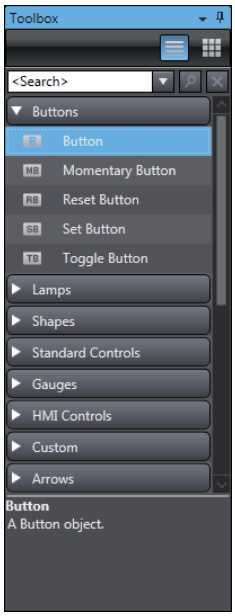


The Code Explorer displays the subroutines in the project. You can double-click a subroutine to edit it. You can also search for the place where a subroutine is used by right-clicking a subroutine name and selecting **Search** from the menu.

Edit Pane (b)

The basic Sysmac Studio operations on the Edit Pane generally apply to HMIs. If an HMI is selected, you can edit pages and set up the HMI.

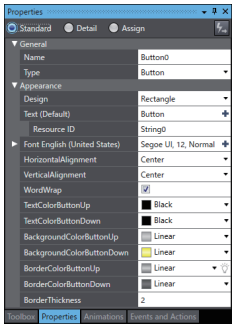
Toolbox (c)



- The Toolbox shows the objects that you can use to edit the page that is displayed in the Edit Pane.

  You can also display the Properties Window, Animations Window and the Events and Actions Window to make the settings of the objects.

Properties Window (c)



- The Properties Window allows you to set the static operations for the selected objects and pages. For items that have a number of properties that fluctuate according to settings such as a text, a setting dialog box is displayed by clicking ✚.

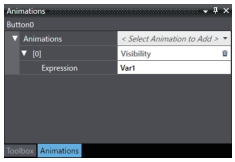
  The Properties Window offers two modes.

  The Standard mode displays only the most commonly used properties.

  The Detail mode displays all the properties.

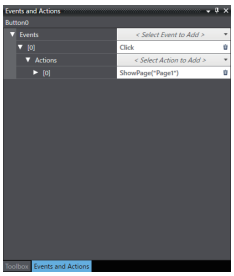
  In the assignment mode, only variables and resource assignments are displayed.

Animations Window (c)



- The Animations Window allows you to set the conditions for dynamically changing the selected objects.

Events and Actions Window (c)



- The Events and Actions Window allows you to set the actions linked to events generated by the selected objects and pages.

2-2 Menu Command Structure

The menu commands that are displayed when an HMI is selected as the device are listed below.

Menu	Submenu/command	
File	Close	—
	Save	
	Save As	
	Save As New Number	
	Import	
	Export	
	Page Settings	
	Print	
	Image File Output	
	Project Properties	
	Exit	
Edit	Undo	—
	Redo	
	Cut	
	Copy	
	Paste	
	Delete	
	Select All	
	Search and Replace	
	Find Previous	
	Find Next	
	Replace Font	
	IAG Replace	
	Device Replace	
	Jump to Multiview Explorer	
	Search unused variables	
View	Multiview Explorer	—
	Project Shortcut View	
	Toolbox	
	3D Visualizer	
	Output Tab Page	
	Watch Tab Page	
	Cross Reference Tab Page	
	Build Tab Page	
	Search and Replace Results Tab Page	
	Page Explorer	
	Code Explorer	
	Properties	
	Animations	
	Events and Actions	
	Smart Project Search	—
	Recently Closed Windows	—
	Clear Recently Closed Windows History	—
	Zoom	Zoom In
		Zoom Out
		Zoom to Fit
		Zoom Reset

Menu	Submenu/command	
View	Manage Window Layout Templates	—
	Reset Window Layout	—
Insert	Controller	NJ101
		NJ301
		NJ501
		NX1P2
		NX102
		NX502
		NX701
		NY512
		NY532
	Application Manager	
	Safety Network Controller	NX
	Drive	
	HMI	NA5
		NA
	Measurement Sensor	ZW
	Vision Sensor	FH
		FHV
		FQ-M
	Slave Terminal	EtherNet/IP Coupler
		PROFINET Coupler
	External Device	—
	Page	
	Page Group	
	User Alarm Group	
	Data Set	
	Recipe	
	Global Subroutine Group	
Project	Build HMI	—
	IncrementalBuild HMI	
	Abort Build	
	Resource Usage	
	Reset Default Value	
	IAG Collections Manager	
HMI	Communications Setup	—
	Change Device	
	Online	
	Offline	
	Synchronization	NA Device
		Media Device
	Transfer	Transfer From Device
		Transfer To Device
	HMI Clock	—
	Update HMI Name	
	Security	HMI Write Protection
		HMI Source Code Protection
	Clear All Memory	—
	Reset HMI Device	

Menu	Submenu/command	
Simulation	Start NA Simulation	—
	Stop NA Simulation	
	Step Execution	
	Step In	
	Step Out	
	Continue	
	Jump to Current Position	
	Set/Clear Breakpoint	
	Enable/Disable Breakpoint	
	Clear All Breakpoints	
	Run with Controller Simulator	
Tools	Import Object Properties	—
	Export Object Properties	
	Set Shortcut Key	
	Option	
Window	Close	—
	Float	
	Dock	
	New Horizontal Tab Group	
	New Vertical Tab Group	
	Move to Next Tab Group	
	Move to Previous Tab Group	
	Close All But This	
	Close All Except Active Device	
	Close All	
	Close Tab	
	Open Next Tab	
	Open Previous Tab	
Help	Help Contents	—
	Instruction Reference	
	Keyboard Mapping Reference	
	Manual Download	
	Video Manual	
	Upgrading Information	
	Rating & Feedback	
	Online Registration	
	About Sysmac Studio	

2-3 Basic Editing Operations

This section describes differences in basic Sysmac Studio operation when an HMI is selected as the device.

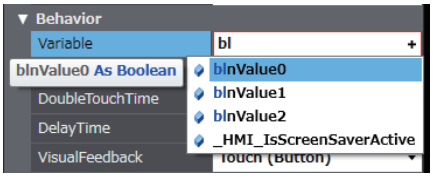
Entry Assistance

There are some differences in the standard operation of the Sysmac Studio when an HMI is selected as the device. This section describes those differences.

● Entering Variable Names and Data Types

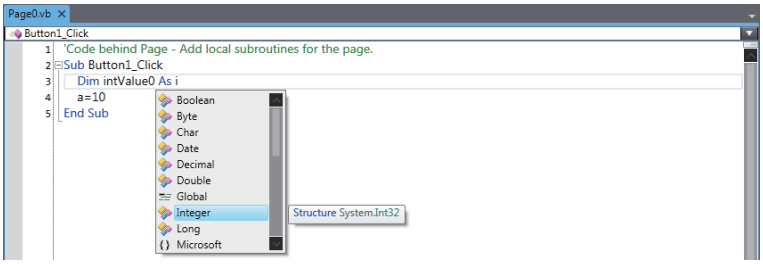
- Entering variable names, e.g., when setting properties
- Entering data types in variable tables

Example: When you enter a variable name as a property, the variable names that you can enter are displayed in a list.

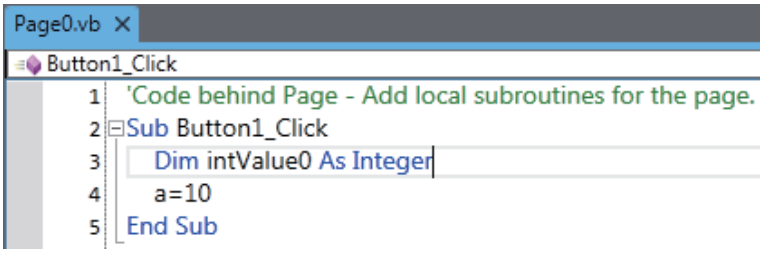


Entering Text in the Code Editor

- When you enter text in the Code Editor, the cursor moves to the first item in the list that starts with the character that you entered.

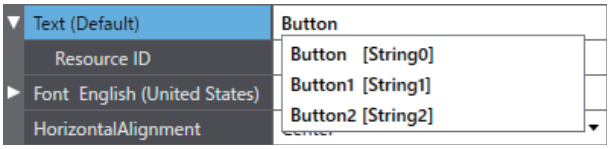


- When you press the **Tab** Key after entering the first part of the keyword (“in” in this example), the rest of the keyword is automatically entered.



Entering Text and Resource ID

- Entering text and resource ID when setting properties
  Example: When you enter text as a property, the texts that you can enter are displayed in a list. Texts and resource IDs are displayed at intervals.



Additional Information

For text which has multiple lines, line feed codes are replaced with spaces in the resource ID list.

2-4 Sysmac Studio Settings and Operations

This section lists the operations of the Sysmac Studio that can be used only when an HMI is selected.

2-4-1 Setting Parameters

Item	Description	Reference
HMI Settings	You can make settings for an HMI.	*Section 3 HMI Configuration and Setup*
Device References	If you connect an HMI to a device (e.g., Controller or PLC) that is not registered in the current HMI project, the connected external device will be added.	*3-2 Device References* on page 3-3
Internal Device	Controllers registered in the project are displayed.	
External Device	You can set up communications and import variables for connected devices that have been added.	
Variable Mapping	You can associate variables in the connected devices with variables in the HMI.	*3-3 Mapping Variables* on page 3-7

2-4-2 Programming

Item	Description	Reference
Toolbox	The Toolbox displays a list of the objects that you can use. You can search for the required objects and drag them to the Page Editor to position the objects.	*Section 5 Objects*
Properties	You can set the static attributes of the pages and objects.	
Animations	You can set the operations for object condition expressions.	
Events and Actions	You can set the actions to perform when events occur.	
Page Explorer	The Page Explorer displays a list of objects on a page. You can select objects or change the order of the display.	
Code Explorer	The Code Explorer displays lists of subroutines in the project. You can double-click a subroutine to edit it.	
Page Editor	You can position objects and create pages. You can also use the Page Editor to make settings for objects.	*4-3 Creating Pages* on page 4-22
Code Editor	You can use Visual Basic to create subroutines.	*4-5 Subroutines* on page 4-56

Item		Description	Reference
Search and Replace		You can search and replace strings in project data.	*4-6 Search and Replace* on page 4-63
Replace Font		Replaces all specified fonts with other fonts.	*4-7 Font Replacement* on page 4-64
IAG Replace		Replaces all specified IAGs with other IAGs.	*4-8 IAG Replacement* on page 4-73
Device Replace		Replaces devices assigned with a variable, etc.	*4-9 Device Replacement* on page 4-76
Cross Reference		Searches the area where the search target is allocated, and displays a list.	*4-10 Cross References* on page 4-79
Build	Build HMI	Convert the project into a form that the HMI can execute.	*4-11 Building* on page 4-80
	IncrementalBuild HMI	Builds only parts that were changed since the last build.	
	Abort Build	You can abort a build operation.	
Resource Usage		You can display the resource usage.	

2-4-3 Library Functions

Item	Description	Reference
Toolbox	You can register objects that you have created and then reuse them.	*Section 9 Reusing Objects*
IAG	You can output an IAG that you created in an IAG project as an IAG collection, to use it in another project.	

2-4-4 Operations for Debugging

Item	Description	Reference
Monitoring	You can monitor variables during project execution. You can monitor the present values of HMI global variables. You use the Watch Tab Page for monitoring.	*Section 7 Debugging*
Changing the Present Values of Variables	You can change the present values of global variables and system-defined variables. You can do this on a Watch Tab Page.	*Section 7 Debugging*
Controlling Execution with Breakpoints and Step Execution	You can control simulation execution to monitor the program or to check operation. Step execution and pausing are also possible.	*Section 7 Debugging*

2-4-5 Communications

Item	Description	Reference
Going Online with an HMI	You can place the computer online with an HMI to synchronize the project.	*Section 6 Connecting to the HMI*

2-4-6 Security Measures

Item		Description	Reference
Prevention of Incorrect Connections	Confirming HMI device Names and Serial IDs	If the device name or the serial ID is different between the project and the HMI when an online connection is established, a confirmation dialog box is displayed.	*6-2-6 Confirming Serial IDs* on page 6-7
Preventing Incorrect Operation	Operation Authority Verification	You can use operation authorities to restrict the ability to perform operations or display data.	*3-5 Security Settings* on page 3-21
Prevention of the Theft of Assets	Password Protection for Project Files	You can set password protection for project files to protect your assets.	
	HMI Source Code Protection	You can set a project password on the HMI device source code to protect your assets.	
Recording operations	Operation Log	You can record logs when operations specified on the NA unit are executed.	*3-8 Operation Log Settings* on page 3-25

2-4-7 Online Help

Item	Description	Reference
Sysmac Studio Help System	You can access Sysmac Studio operating procedures.	
Keyboard Mapping Reference	You can display a list of convenient shortcut keys that you can use on the Sysmac Studio.	

2-4-8 Project Management Functions

Item	Description	Reference
Image File Output	You can output a page as an image file. You can set any page to be output.	*11-3 Image File Output* on page 11-6
Import/Export User Alarm	You can import or export user alarms.	*11-4 Import/Export User Alarm* on page 11-7
Import/Export Resources	You can import or export resource general strings and alarm strings.	*11-5 Import/Export Resources* on page 11-13
Import/Export Object Properties	You can import or export object text and variables on pages, and expressions.	*11-6 Import/Export Object Properties* on page 11-20
Import/Export Pages	You can import or export pages and page-related settings.	*11-7 Importing/Exporting Pages* on page 11-24

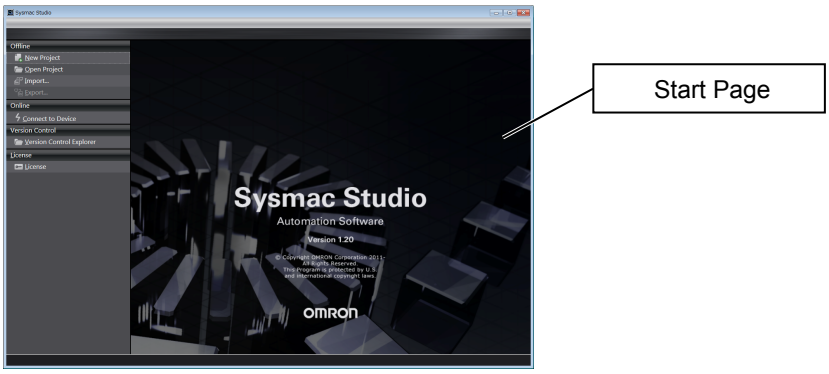
2-5 Basic Operations for HMI Projects

This section describes how to create and save projects and perform other basic operations to use HMIs.

2-5-1 Creating a Project File from the Start Page

Use the following procedure to create a project file from the Start Page.

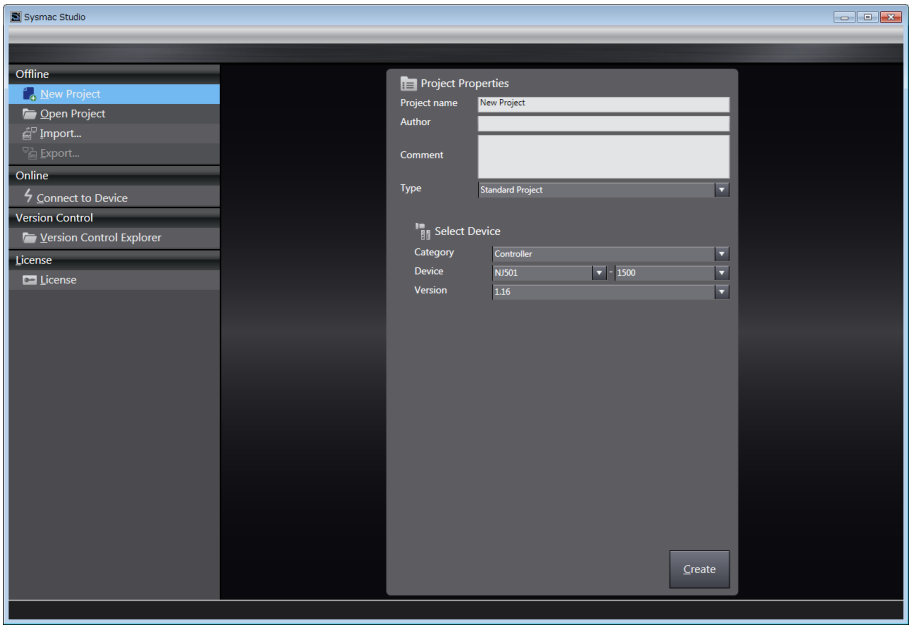
1 Click the **New Project** Button in the Start Page.



The Project Properties Dialog Box is displayed. The following table gives the functions of the buttons.

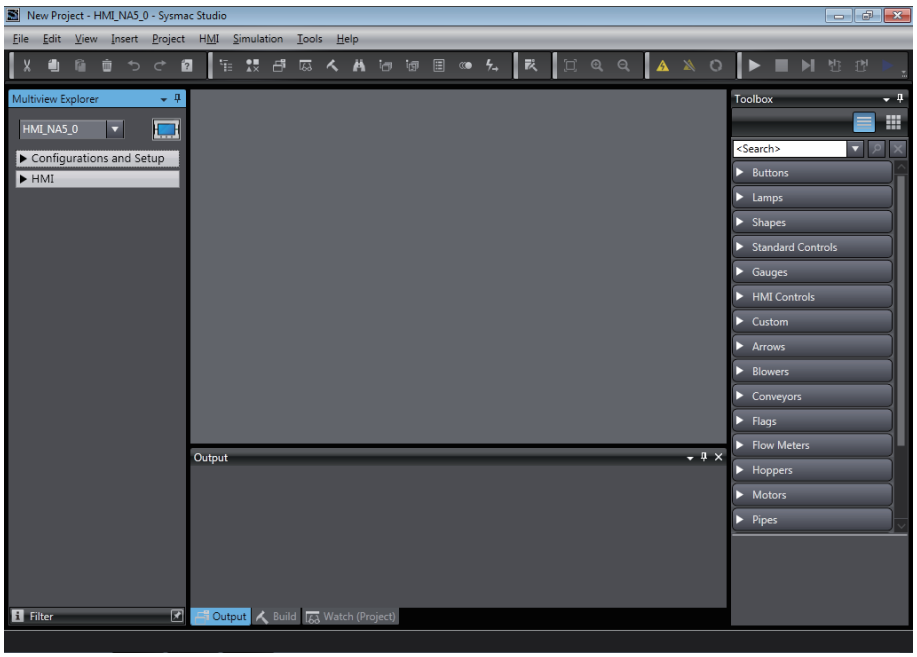
Menu command	Description
New Project Button	Creates a project file.
Open Project Button	Opens an existing project file.

2 Enter the project name, author, and comment in the Project Properties Dialog Box, select *HMI* from the device category, and then click the **Create** Button. (Only the project name is required.)

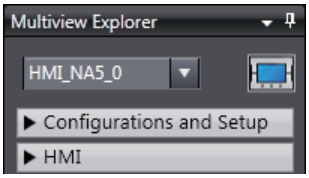


You can change the properties later. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504).

A project file is created and the following window is displayed.

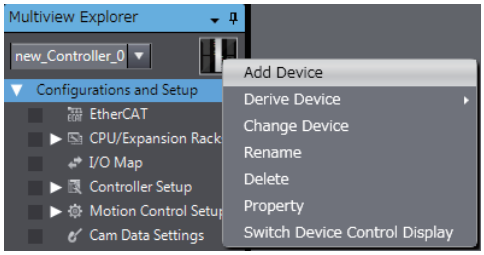


A project file is created with the specified device already inserted.



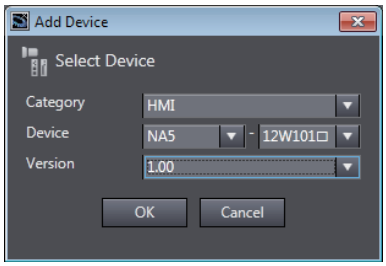
2-5-2 Adding an HMI to an Existing Project

Right-click the Controller Icon and select ***Add Device*** from the menu.

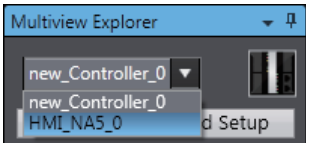


Or, select the device directly from the Insert Menu.

Example: ***HMI - NA5***: The Add Device Dialog Box is displayed.



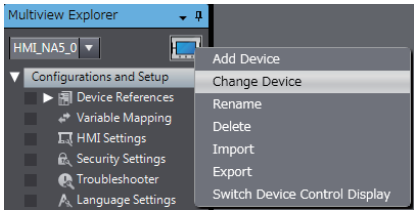
Select the device and then click the **OK** Button. The device is added to the project.



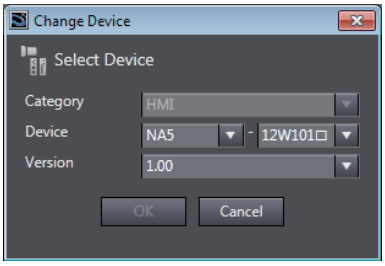
To change the target device, select a device from the list.

2-5-3 Changing Devices

Right-click the HMI Icon and select ***Change Device*** from the menu. Or, select ***Change Device*** from the HMI Menu.



The Change Device Dialog Box is displayed.



Select the device and then click the **OK** Button. The device is changed.

Precautions for Correct Use

- If you change the device, the settings for functions that are not supported by the new model will be lost.
- If you change to a model that has a different display size, the objects will be enlarged or reduced according to the new display size. However, elements other than objects, such as font sizes, will not change.

HMI Versions

Set the version when you create a new HMI project or when you add an HMI to an existing project.

You can set the version to the runtime version of the HMI that you are using. You can program and make settings within the ranges that are supported for the runtime version. If you attempt to use functions that are not supported by the runtime version that you set, you will not be able to use them or errors will occur.

Additional Information

When you open the project that was created in past versions of the Sysmac Studio version 1.11 or higher, it will be converted as follows.

Runtime version before conversion	Sysmac Studio																		
	V1.10	V1.11, V1.12	V1.13	V1.14	V1.15	V1.16	V1.17	V1.18 to V1.22	V1.23	V1.24 to V1.26	V1.27 to V1.30	V1.31	V1.32 to V1.42	V1.43 to V1.44	V1.45 to V1.49	V1.50 to V1.53	V1.51 to V1.63	V1.64 to V1.66	V1.67 or later
1.00	1.00	1.01*1	1.02*1	1.03*1	1.03*1*2	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1
1.01	—	1.01	1.02*1	1.03*1	1.03*1*2	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1
1.02	—	—	1.02	1.03*1	1.03*1*2	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1
1.03	—	—	—	1.03	1.03*2	1.03	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1	1.03*1
1.04	—	—	—	—	—*2	1.04	1.04	1.04	1.04	1.04	1.04	1.04	1.04	1.04	1.04	1.04	1.04	1.04	1.04
1.05	—	—	—	—	—	1.05	1.05	1.05	1.05	1.05	1.05	1.05	1.05	1.05	1.05	1.05	1.05	1.05	1.05
1.06	—	—	—	—	—	—	1.06	1.06	1.06	1.06	1.06	1.06	1.06	1.06	1.06	1.06	1.06	1.06	1.06
1.07	—	—	—	—	—	—	1.07	1.07	1.07	1.07	1.07	1.07	1.07	1.07	1.07	1.07	1.07	1.07	1.07
1.08	—	—	—	—	—	—	—	1.08	1.08	1.08	1.08	1.08	1.08	1.08	1.08	1.08	1.08	1.08	1.08
1.09	—	—	—	—	—	—	—	—	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09
1.10	—	—	—	—	—	—	—	—	—	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10	1.10
1.11	—	—	—	—	—	—	—	—	—	—	1.11	1.11	1.11	1.11	1.11	1.11	1.11	1.11	1.11
1.12	—	—	—	—	—	—	—	—	—	—	—	1.12	1.12	1.12	1.12	1.12	1.12	1.12	1.12
1.13	—	—	—	—	—	—	—	—	—	—	—	—	1.13	1.13	1.13	1.13	1.13	1.13	1.13
1.14	—	—	—	—	—	—	—	—	—	—	—	—	—	1.14	1.14	1.14	1.14	1.14	1.14
1.15	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1.15	1.15	1.15	1.15	1.15
1.16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1.16	1.16	1.16	1.16
1.17	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1.17	1.17	1.17
1.18	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1.18	1.18
1.19	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	1.19

*1. The runtime version will be converted.

*2. The runtime version will be converted to 1.04 if the HMI Extended Option is enabled.

2-5-4 Importing and Exporting Devices

Right-click the HMI Icon and select **Import** or **Export** from the menu.

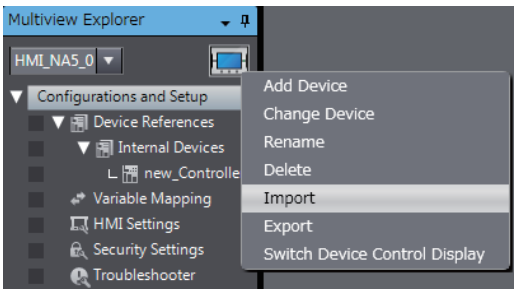
Additional Information

You cannot execute exporting devices if an error exists in the project.

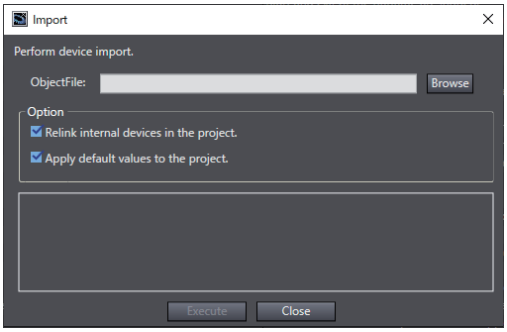
Importing Devices

Use the following procedure to import a device.

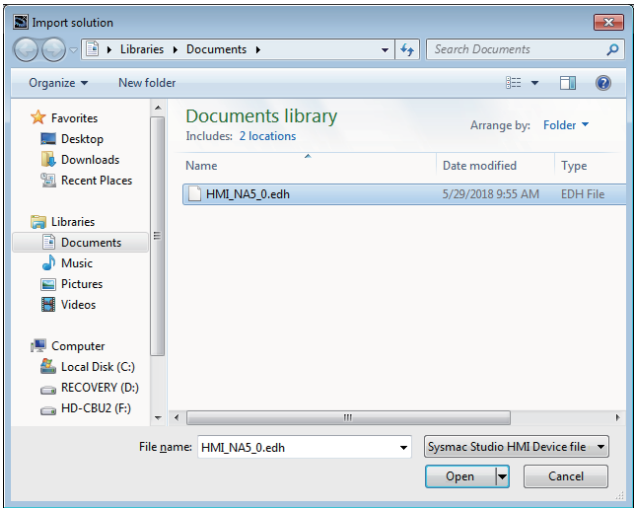
1 Right-click the HMI Icon and select **Import** from the menu.



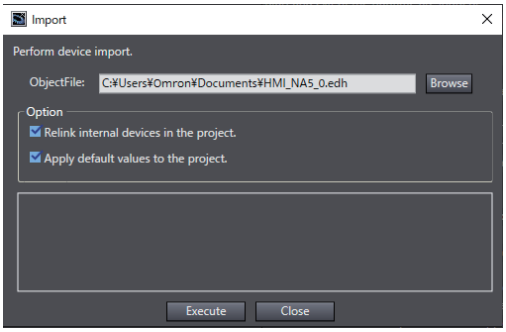
2 Click **Browse**.



3 Select the file and click **Open**.



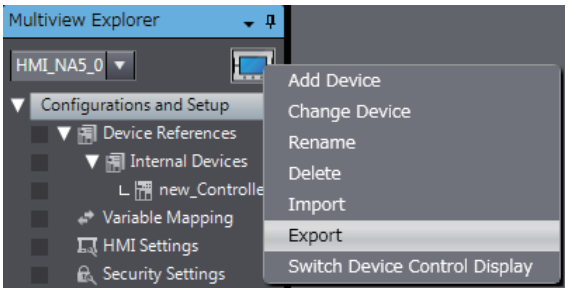
4 Set the options as required and click **Execute**.



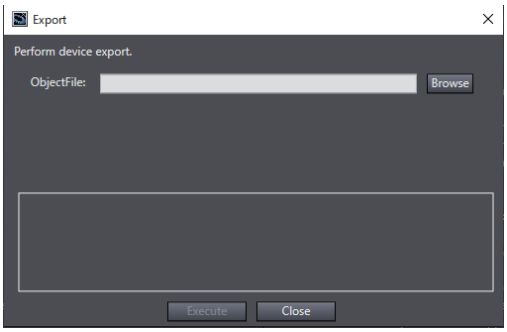
Exporting Devices

Use the following procedure to export a device.

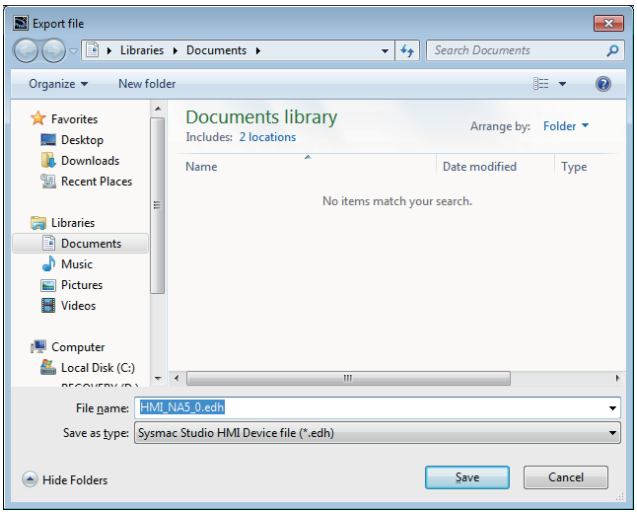
1 Right-click the HMI Icon and select **Export** from the menu.



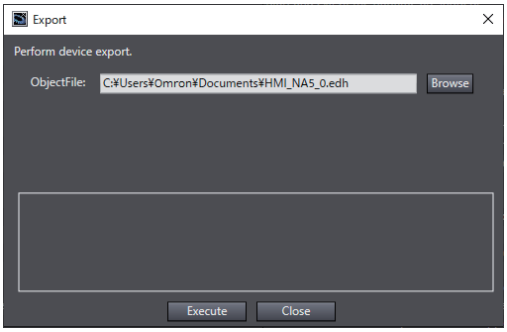
2 Click **Browse**.



3 Select the file and click **Save**.



4 Click **Execute**.



3

HMI Configuration and Setup

This section describes how to configure and set up HMIs on the Sysmac Studio, including mapping variables with connected devices and HMI settings.

3-1 Outline of Configurations and Setup 3-2
3-1-1 Connected Device Registration and Variable Mapping 3-2
3-2 Device References 3-3
3-2-1 Types of Connected Devices 3-3
3-2-2 Connected Devices in the Current Project 3-3
3-2-3 Registering External Connected Devices 3-4
3-3 Mapping Variables 3-7
3-3-1 Mapping Variables 3-7
3-3-2 Opening the Variable Mapping Tab Page and Tab Page Parts 3-7
3-3-3 Variable Mapping Methods 3-8
3-3-4 When a Variable Mapping Error Cannot be Resolved 3-9
3-4 HMI Settings 3-11
3-4-1 HMI Settings 3-11
3-4-2 Device Settings 3-12
3-4-3 TCP/IP Settings 3-14
3-4-4 FTP Settings 3-15
3-4-5 NTP Settings 3-16
3-4-6 FINS Settings 3-17
3-4-7 VNC Settings 3-18
3-4-8 Printing Settings 3-19
3-4-9 Serial Port Settings 3-20
3-5 Security Settings 3-21
3-6 Troubleshooter 3-22
3-7 Language Settings 3-23
3-8 Operation Log Settings 3-25
3-9 HMI Clock 3-28
3-10 Updating the HMI Name 3-29
3-11 Write Protecting the HMI 3-30
3-12 Clear All Memory 3-31
3-13 Resetting the HMI 3-32

3-1 Outline of Configurations and Setup

This section describes how to set up HMIs and connected devices, such as Controllers and PLCs. The following items are provided in the HMI Configurations and Setup.

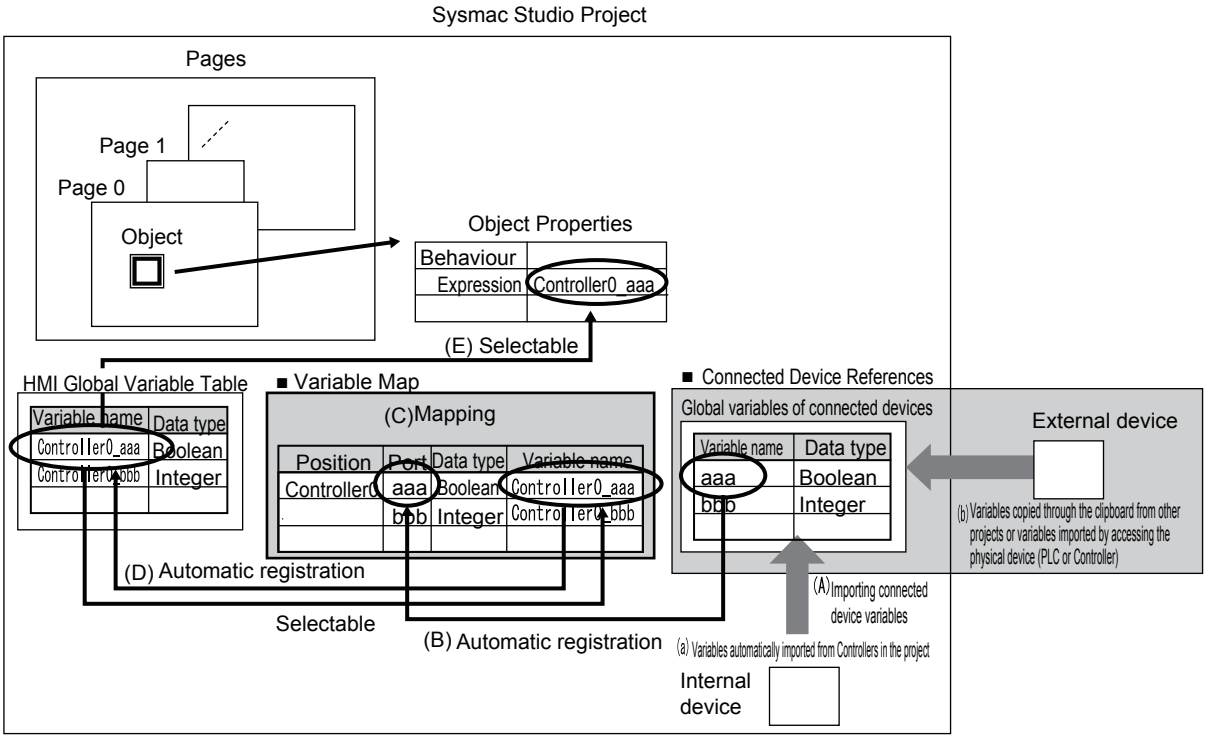
Item	Description
Device References	You can set up connected devices and import variables.
Variable Mapping	You can assign HMI variables to the variables in the connected devices.
HMI Settings	These are the parameters related to the HMI.
Security Settings	You can set up restrictions to operations on HMIs.
Troubleshooter	You can set parameters for the Troubleshooter.
Language Settings	You can make settings for multi-language projects.
Operation Log Settings	These are used for an Operation Log.

3-1-1 Connected Device Registration and Variable Mapping

Device references must be set only to connect to external devices that are not registered in the current project. They are not necessary to connect to a Controller that is registered in the current project.

The following figure shows the relationship between connected device references and variable mapping.
The HMI global variables are mapped to the connected device variables.

To access variables in the connected devices from an HMI, you must map the variables.



(A) Connected device variables are a) automatically imported from the same project or b) copied from another project or manually imported from an external device or a file, such as an Excel file.
(B) The connected device variables are automatically registered in the variable mappings.
(C) The HMI global variables are mapped to the connected device variables.
(D) The mapped HMI global variables are automatically registered in the global variable table of the HMI.
(E) You specify HMI global variables in the object properties.

3-2 Device References

This section describes how to set up HMIs and connected devices, such as Controllers and PLCs.

3-2-1 Types of Connected Devices

Different operations are used to connect to Controllers that are registered and Controllers that are not registered in the current project.

- Controllers that are already registered in the current project are automatically registered in the HMI project as internal connected devices.
- To connect to a device that is not registered in the current HMI project, you must register the device as an external connected device.

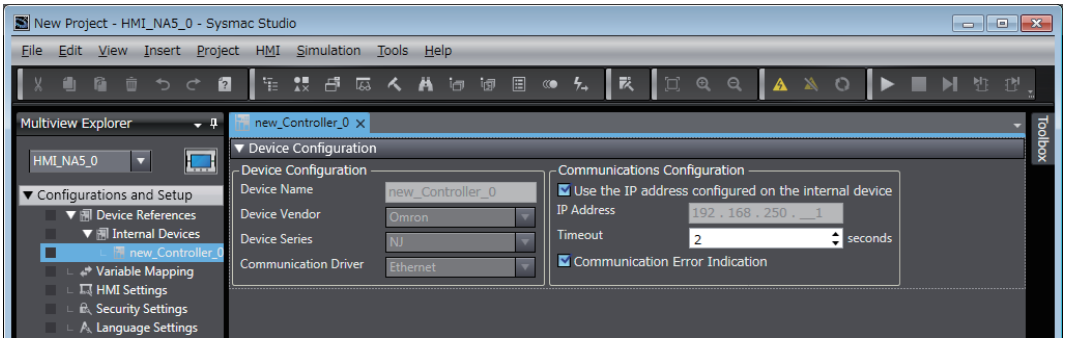
Additional Information

If you upload a project that includes Controllers registered as internal connected devices to a project in which the Controllers are not registered, the devices are registered as internal connected devices that do not have links to Controllers. Refer to *8-1 Transfer* on page 8-2 for details.

3-2-2 Connected Devices in the Current Project

Controllers that are registered in the current project are displayed as connected devices. Use the following procedure to display the device settings if you need to check them.

1. Click **Device References** under **Configurations and Setup** in the Multiview Explorer.
2. The Controllers that are registered in the current project are displayed under **Internal Device**.
3. Double-click the project to display the following Device Configuration Tab Page.



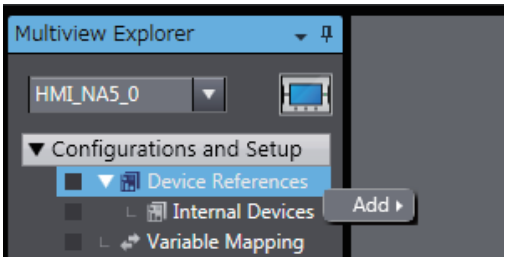
3-2-3 Registering External Connected Devices

To connect the HMI to a device that is not registered in the current HMI project, you must register the device as an external connected device. The procedures to register and set up external connected devices are given below.

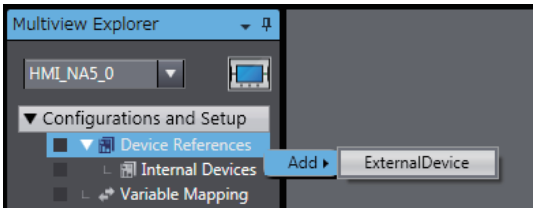
Registering and Setting Up External Connected Devices

This section describes how to register and set up external connected devices.

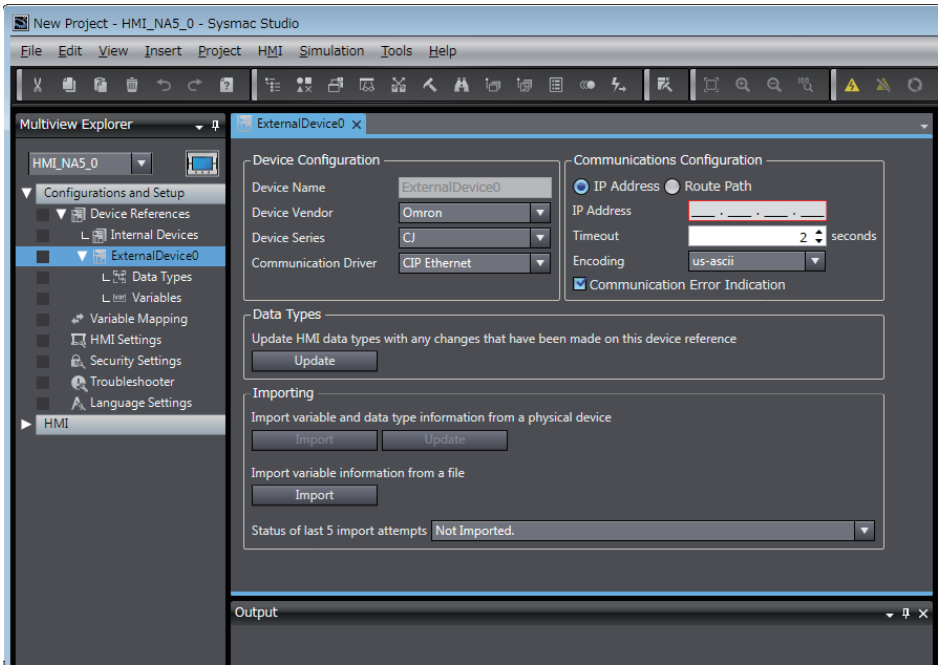
1 Right-click **Device References** under **Configurations and Setup** in the Multiview Explorer.



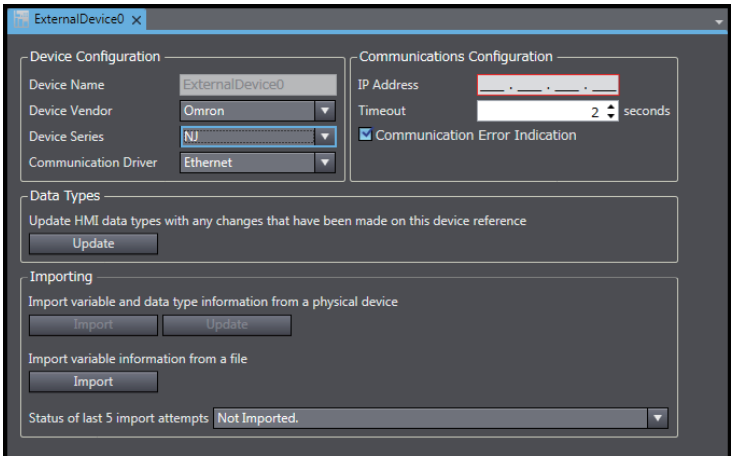
2 Select ***Add - ExternalDevice***. The device is added as ExternalDevice☐, where ☐ is a serial number starting from 0.



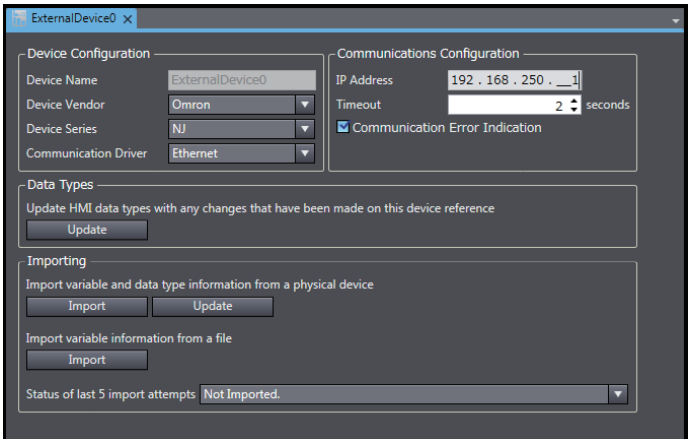
3 Double-click the new ExternalDevice☐.



4 Select the vendor of the required device in the *Device Vendor* Box under **Device Configuration**. The device series of the selected vendor is displayed. Make the selections for the required device. The device communications drivers of the vendor selected for the device series are displayed. Select the communications driver for the required device.



5 Make the required settings in the *Communications Configuration* Area. Refer to the *NA-series Programmable Terminal Device Connection User's Manual* (Cat. No. V119) for details.



Importing External Connected Device Variables

To connect the HMI to a device that is not registered in the current HMI project, you must import the variables from the external connected device.

There are three ways to import external connected device variables.

- Importing device variables online from the actual external connected device
- Copying variables from the variable table in another project
- Importing variables from an Excel or CXT file

Importing Device Variables Online from the Actual External Connected Device

Click the **Import Variables** Button. The variables are imported from the external connected device.

Refer to the *NA-series Programmable Terminal Device Connection User's Manual* (Cat. No. V119) for details.

Copying Variables from the Variable Table in Another Project

You can use the clipboard to copy the required variables from the Support Software for the connected device and paste them in the device variables table for the external connected device. However, you cannot copy connected device variables if they are structure variables.

Refer to the *NA-series Programmable Terminal Device Connection User's Manual* (Cat. No. V119) for details.

Importing Variables from an Excel or CXT File

Click the **Import from File** Button. The variables are imported for the external connected device. Refer to the *NA-series Programmable Terminal Device Connection User's Manual* (Cat. No. V119) for details.

Updating Device Variables

If you change the variables on a device, update the device variables in the HMI project as required.

There are three ways to update device variables.

- Updating device variables online from the actual external connected device
- Copying variables from the variable table in another project
- Importing variables from an Excel or CXT file

Updating Device Variables Online from the Actual External Connected Device

Click the **Update Variables** Button. The differences between the variables on the external connected device and the device variables in the HMI project are displayed. Select the variables to update.

Copying Variables from the Variable Table in Another Project

You can use the clipboard to copy the required variables from the Support Software for the connected device and paste them in the device variables table for the external connected device.

Importing Variables from an Excel or CXT File

Click the **Import from File** Button. The differences between the variables in the Excel file and the device variables in the HMI project are displayed. Select the variables to be updated.

3-3 Mapping Variables

This section describes the settings required to access variables in connected devices through HMI global variables.

3-3-1 Mapping Variables

Mapping variables refers to assigning variables in devices connected to the HMI (called device variables) to global variables in the HMI. Device variables are used on the HMI by assigning them to HMI global variables.

Therefore, mapping variables is required. Not accessing device variables directly allows you to reuse projects simply by changing the variable mappings.

Global variables that are assigned to device variables are called external variables.

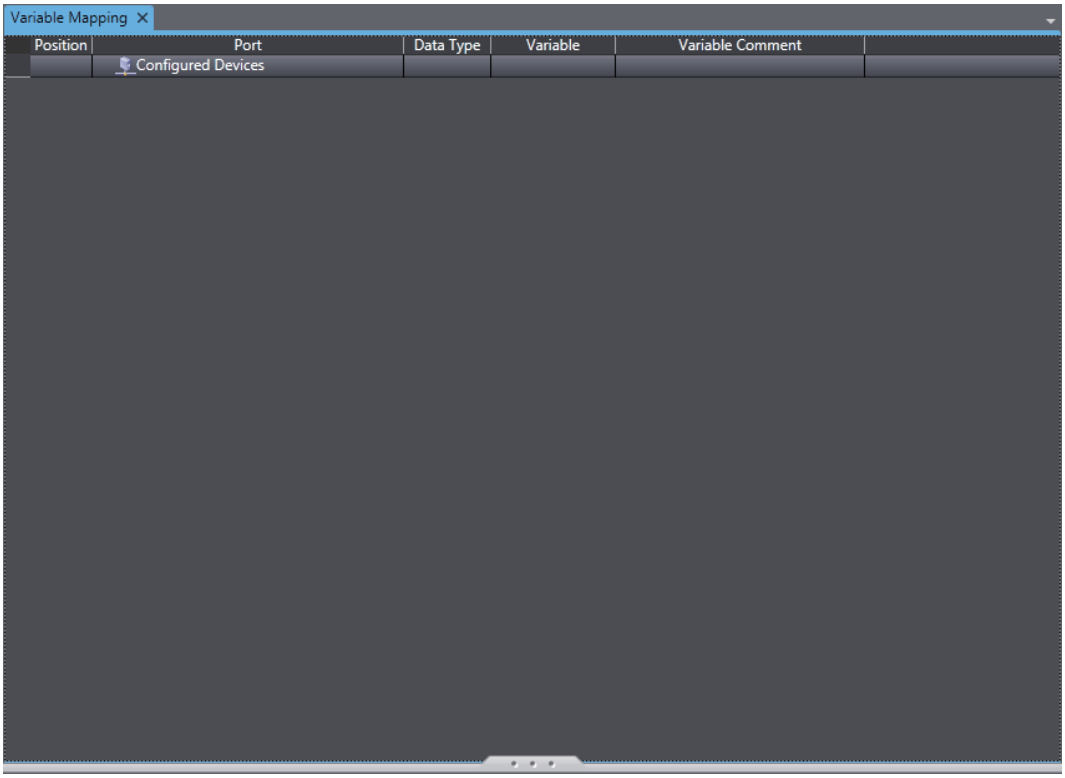
3-3-2 Opening the Variable Mapping Tab Page and Tab Page Parts

Additional Information

You can also use a setting to map variables automatically. For details, refer to *11-1 Sysmac Studio Option Settings* on page 11-2.

1 Double-click **Variable Mapping** under **Configurations and Setup**.

The Variable Mapping Tab Page is displayed.



Parts of the Window

No	Item	Description
1	Position	Displays the IP addresses of the connected devices.
2	Port	Displays the connected devices and device variables in a tree structure.
3	Data Type	The data types of the device variables are displayed.
4	Variable	You can set the name of a HMI global variable. You can use entry assistance to select from a list of previously registered HMI global variables.
5	Variable Comment	You can set comments for the HMI global variables. These comments are also applied to the global variable table. If comments are set for the device variable, Comment*1 for the device variable is automatically applied.

*1. If there are multiple comments registered in internal variables, Comment 1 will be applied.

3-3-3 Variable Mapping Methods

To map variables, you can either create new global variables and assign them or you can assign previously created global variables.

To increase the reusability of the project, create the global variables first and then assign them.

However, if a device variable is a structure, you must create a new external variable during variable mapping.

Creating New External Variables

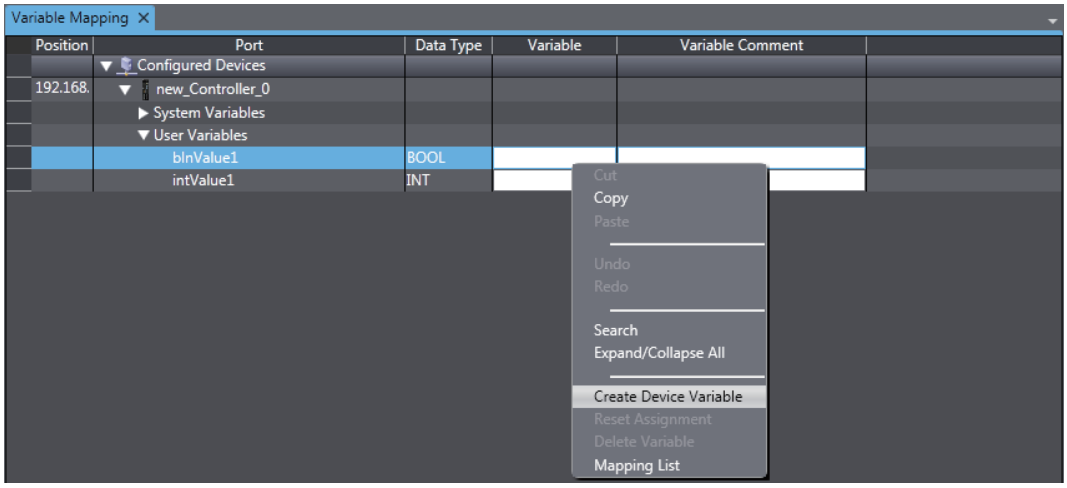
You can create a new global variable and assign it to a device variable.

When you create an external variable, you can either have the name generated automatically or you can create it manually.

● Automatically Creating New Variable Names

Use the following procedure.

1 Select one or more device variables in the variable mappings, right-click, and select ***Create Device Variable*** from the popup menu.



Automatically generated variable names are registered in the global variable table according to the following rule.

Automatic generation rule: The device variable name is added after the controller name and separated with an underline.

- **Manually Entering New Variable Names**

  Select the device variable and directly enter the variable name in the *Variable* column.

Selecting Previously Registered Global Variables and Mapping Them

You can select global variables that are already registered in the global variable table and assign them to device variables.

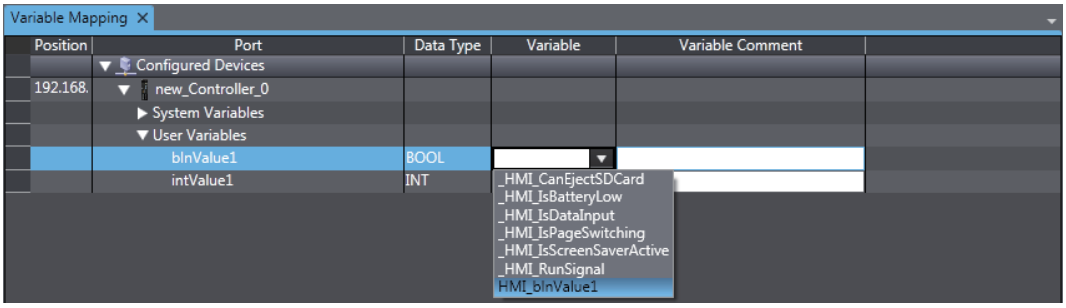
For example, this method can be used to map external variables in the following cases.

- Setting an HMI global variable first and then assigning it to a device variable
- Creating a common project first and specifying connected devices later

Use the following procedure.

1 Register the global variables in the global variable table in advance.

2 When you map variables, you can select global variables from lists of variables that are already registered in the global variable table and assign them to device variables.



Additional Information

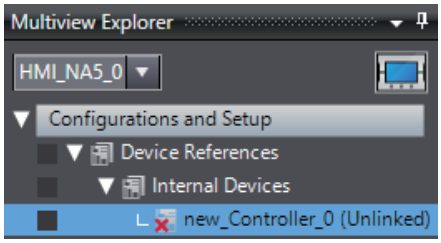
When the variables are mapped, the device variable comments are automatically set as global variable comments.

3-3-4 When a Variable Mapping Error Cannot be Resolved

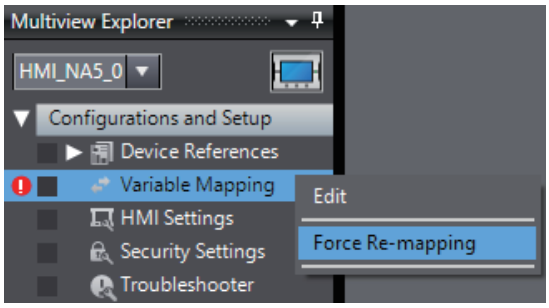
Follow the procedure below if a variable mapping error cannot be resolved.

1 If an unlinked internal device exists, relink it.

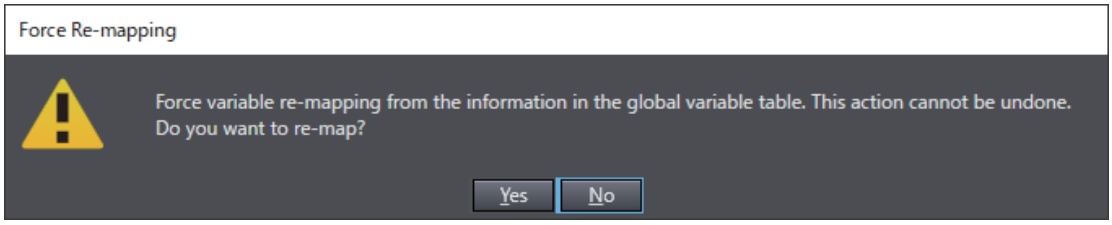
Refer to *8-4 Relinking Internal Devices* on page 8-5 for information on how to relink internal devices.



2 Right-click **Variable Mapping** under **Configurations and Setup** in the Multiview Explorer, and select **Force Re-mapping** in the menu that appears.



3 Click **Yes**.



Additional Information

When forced remapping is executed, if the variables were deleted on the controller side, the mapping is canceled and internal variables are used. As the controller user-defined data type cannot be used for internal variables, a build error occurs if the build is executed in this state. Make appropriate corrections such as re-registering the variables on the controller side.

Similarly, if the data type is changed on the controller side, an error may occur in the object properties. Make appropriate corrections such as changing the variable assignments.

Note that all items other than the name and AT will be initialized. If you have changed these items, make sure to reconfigure them.

3-4 HMI Settings

This section describes the HMI settings.

3-4-1 HMI Settings

You can make settings for an HMI.

The following table lists the setting items.

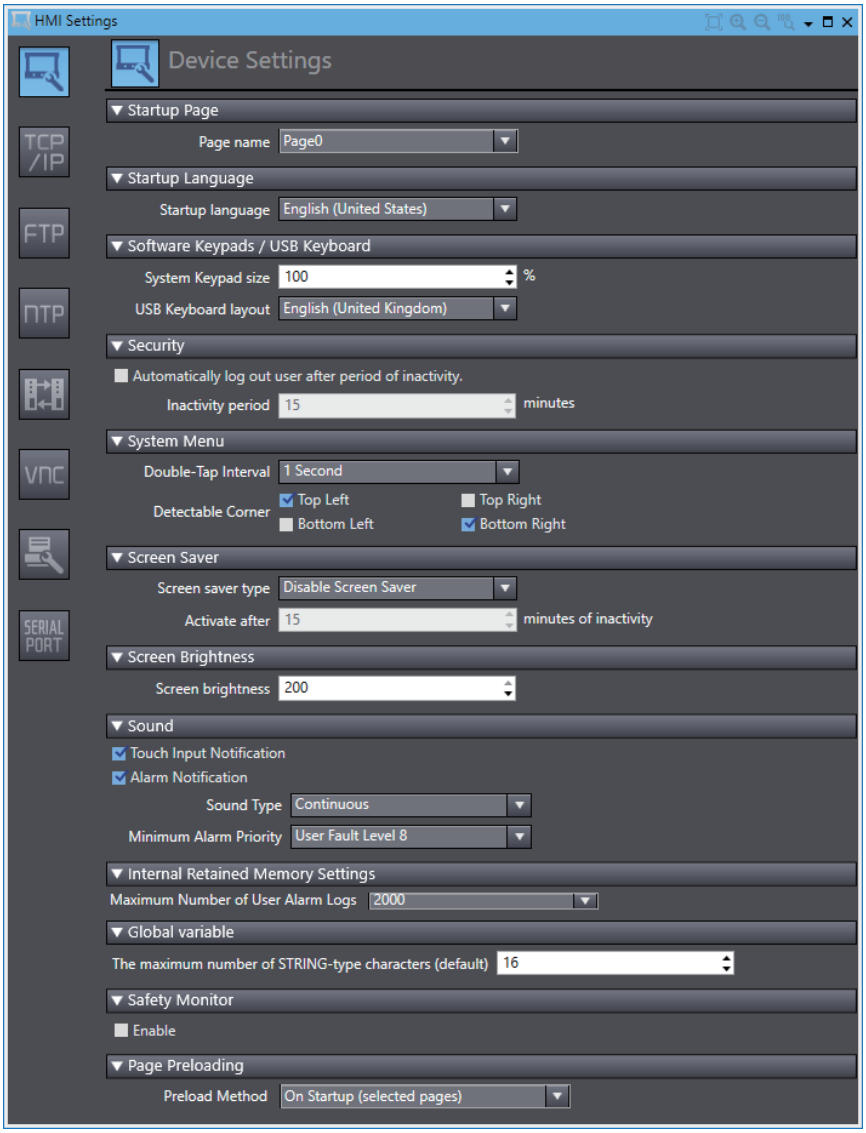
Item	Icon	Description	When setting is required
Device Settings		There are page, screen saver, brightness settings, and other settings.	These settings are always required.
TCP/IP Settings		These are the Ethernet settings for Ethernet ports 1 and 2.	These settings are always required.
FTP Settings		These are the FTP server settings.	These settings are required to use the FTP server.
NTP Settings		These are the NTP client settings.	These settings are required when you use an NTP client.
FINS Settings		These are the settings for FINS communications.	These settings are required when using FINS communications with a CS/CJ/CP-series PLC.
VNC Settings		These are the VNC settings.	These settings are required to use VNC.
Print Settings		These are the settings for printing and capturing screens on the HMI.	These settings are required to print or capture screens.
Serial port Settings		These are serial port settings.	When using a serial port to connect with the CS/CJ/CP series.

Setting Procedure for HMI Settings

1. Double-click **HMI Settings** under **Configurations and Setup** in the Multiview Explorer. The HMI Settings Tab Page is displayed in the Edit Pane.
2. Click the icons on the left to display the corresponding dialog boxes.

3-4-2 Device Settings

These are the device settings.

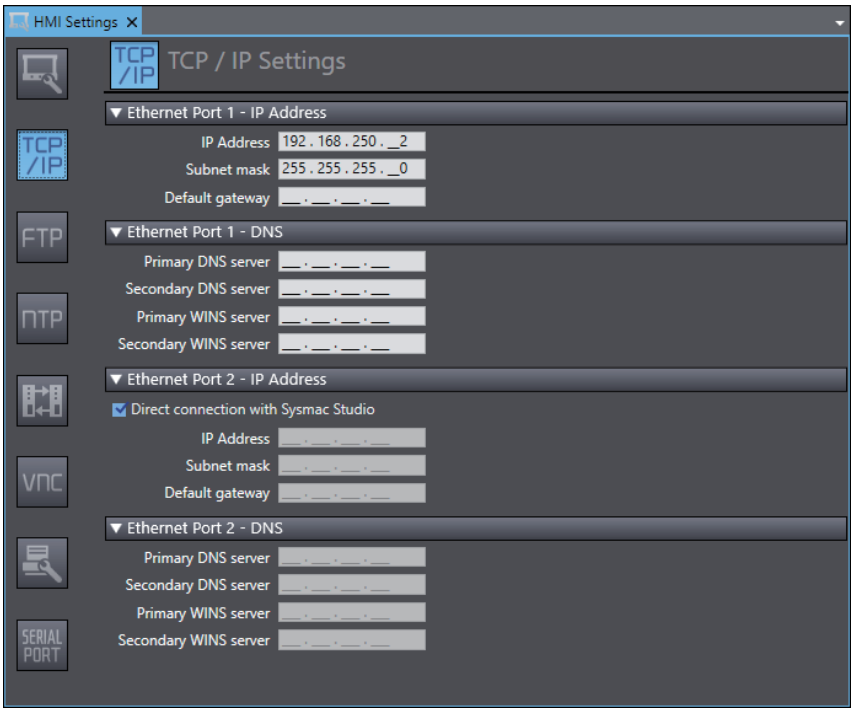


Item	Description
Startup Page	
Page name	Sets the page to display first when the HMI is started.
Startup Language	
Startup language	Sets the project language to use when the HMI is started.
Software Keypads/USB Keyboard	
System Keypad Size	Sets the display size of the system keypad.
USB Keyboard layout	Sets the layout of the USB keyboard.
Security	
Automatically log out user after period of inactivity.	Select this check box to automatically log out the user after a specified period of inactivity.
Inactivity period	Specify the time of inactivity before the user is logged out automatically.

Item	Description
System Menu	
Double-tap Interval	Sets the interval to use to detect double taps.
Detectable corner	Select the corners in which to detect the operation to display the System Menu.
Screen Saver	
Screen saver type	Sets the type of screen saver.
Activate after	Sets the time after the screen is touched before the screen saver is started.
Screen Brightness	
Screen brightness	Sets the brightness of the screen.
Sound	
Touch Input Notification	Sets whether to beep when an input is accepted.
Alarm Notification	Sets the buzzer that is beeped when an alarm occurs, specifically, the type of buzzer and the user alarm level.
Internal Retained Memory Settings	
Maximum Number of User Alarm Logs	Sets the maximum number of alarm logs to save in the NA.
Global Variables	
The maximum number of STRING-type characters (default)	Sets the maximum number of characters allocated for a String Retain variable when the maximum number of characters is not set.
Safety Monitor	
Enable	Enables the Safety Monitor function.
Page Preloading	
Preload Method	Select an option to speed up page transition on the NA Unit. This setting is only available on the NA5-□□□□□□□-V2. • On Startup (selected pages): Preloads only complex pages on startup. Startup time may be extended depending on page content. (Models other than the NA5-□□□□□□□-V2 operate in this mode.) • Delay Execution (selected pages): Begins preload in the background after the startup page is displayed. The applicable pages are the same as “On Startup (selected pages)”. • Delay Execution (all pages): Begins preload in the background after the startup page is displayed. Preloads all pages.

3-4-3 TCP/IP Settings

These are the settings for TCP/IP.



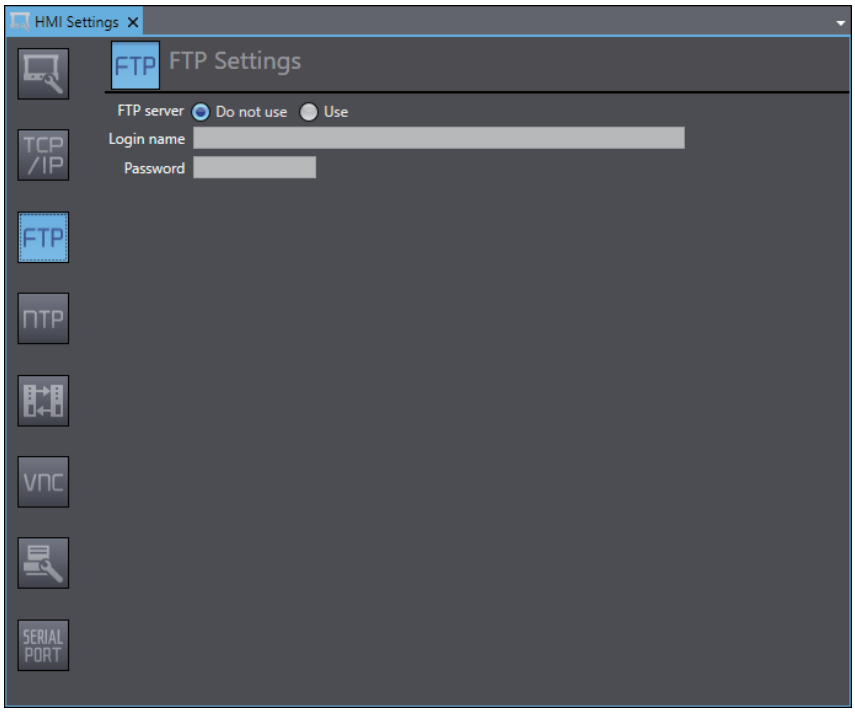
The following settings are provided for Ethernet port 1 and Ethernet port 2.

Item	Description
Ethernet Port 1 - Settings	
IP Address	Sets the local IP address.
Subnet mask	Sets the subnet mask.
Default gateway	Sets the IP address of the default gateway. This setting is not required when a default gateway is not used.
Primary DNS server	Sets the IP address of the primary DNS server.
Secondary DNS server	Sets the IP address of the secondary DNS server.
Primary WINS server	Sets the IP address of the primary WINS server.*1
Secondary WINS server	Sets the IP address of the secondary WINS server.*1
Ethernet Port 2 - Settings	
Direct connection with Sysmac Studio	Select this check box to connect Ethernet port 2 directly to the Sysmac Studio without going through an Ethernet switch. If you select this check box, the IP addresses and other settings for Ethernet port 2 are ignored.
IP Address	Sets the local IP address.
Subnet mask	Sets the subnet mask.
Default gateway	Sets the IP address of the default gateway. This setting is not required when a default gateway is not used.
Primary DNS server	Sets the IP address of the primary DNS server.
Secondary DNS server	Sets the IP address of the secondary DNS server.
Primary WINS server	Sets the IP address of the primary WINS server.*1
Secondary WINS server	Sets the IP address of the secondary WINS server.*1

*1. Not displayed for the NA5-□□W□□□□-V2.

3-4-4 FTP Settings

These are the FTP server settings.

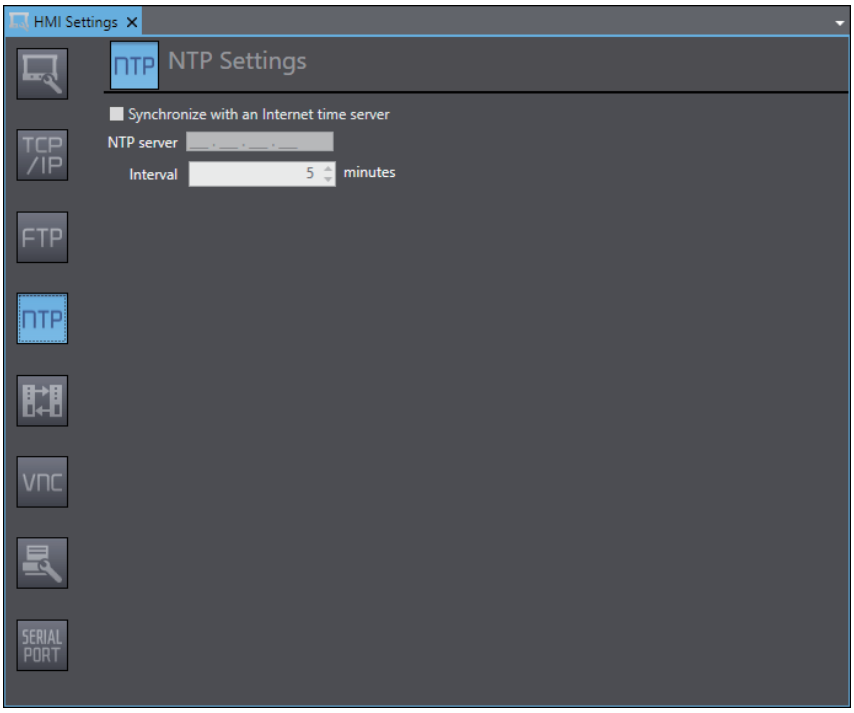


Item	Description
FTP Settings	
FTP server	Specifies whether to use the FTP server of the HMI.
Login name	Sets the login name to externally connect to Ethernet port 1 or 2 on the HMI via FTP. You can use up to 130 alphanumeric characters.
Password	Sets the password to use to externally connect to Ethernet port 1 or 2 via FTP. You can use up to 15 alphanumeric characters.

3-4-5 NTP Settings

These are the settings for an NTP (Network Time Protocol) client.

The HMI gets the clock information from the specified NTP server at the specified interval and updates the built-in clock information.



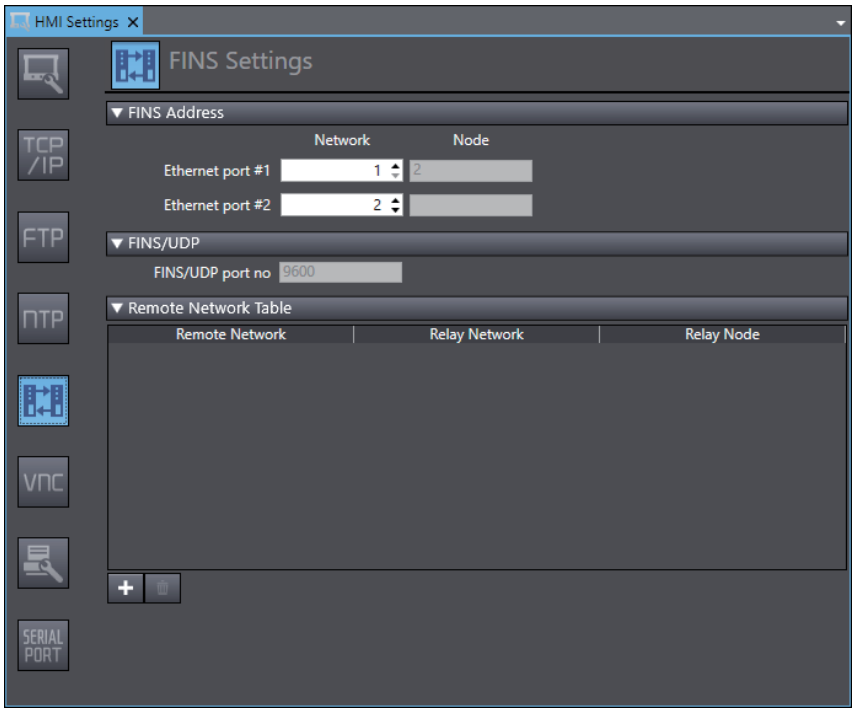
Item		Description
NTP Settings		
	Synchronize with an Internet time server	Select this check box to synchronize the built-in clock in the HMI with the clock information from the NTP server.
	NTP server	Sets the IP address of the NTP server.
	Interval	Sets the interval at which to get the clock information from the NTP server. *1

*1. Communication with the server is based on these settings. However, if the displacement between the time obtained from the server and the HMI built-in clock exceeds 24 hours, the built-in clock will not be corrected.

3-4-6 FINS Settings

These are the settings for FINS communications.

These settings are required when FINS communications are used between the HMI and a CS/CJ/CP-series PLC.



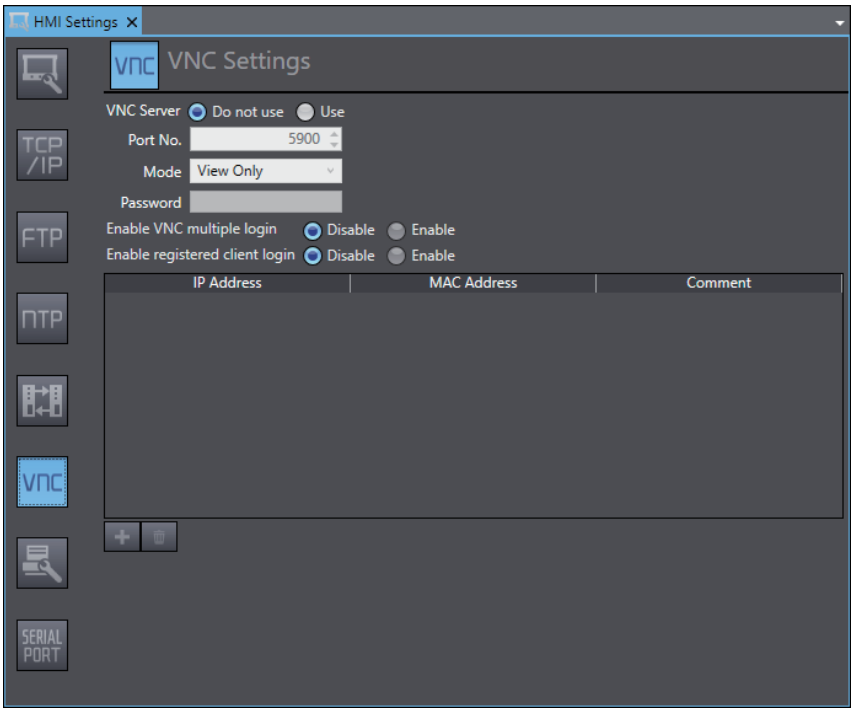
Item	Description
FINS Address	
Ethernet port #1	Sets the FINS network address of Ethernet port 1. The FINS node address is automatically created from the IP address.
Ethernet port #2	Sets the FINS network address of Ethernet port 2. The FINS node address is automatically created from the IP address.
FINS/UDP	
FINS/UDP port no	Displays the port number used for FINS/UDP.
Remote Network Table	Sets the routing table.

3-4-7 VNC Settings

These are the settings for VNC (Virtual Network Computing).

VNC implements a remote desktop to allow remote control of a computer located on a network.

You must enable the server functionality on the HMI to control. You can start the client software on the computer from which to perform remote control, connect to the HMI, and then control it remotely.

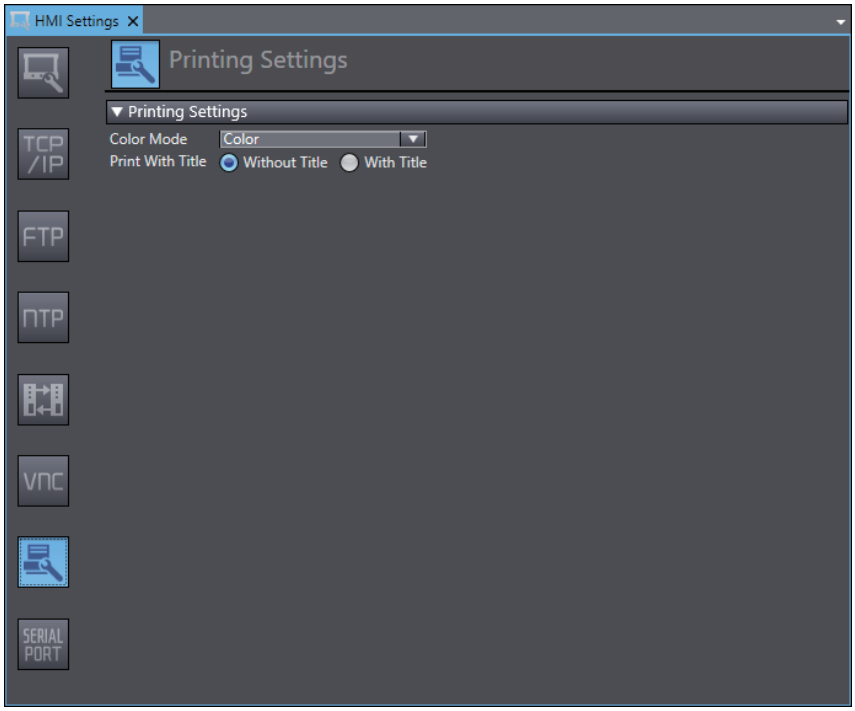


Item	Description
VNC Settings	
VNC Server	Specifies whether to use VNC.
Port No.	Sets the port number.
Mode	Sets the operations to enable from the VNC client.
Password	Sets the password.
Enable VNC multiple login	Specifies whether to allow multiple clients to log in.
Enable registered client login	Specifies whether to allow only certain clients to log in.
IP Address	Registers the IP addresses of the client devices to be allowed to log in.
MAC Address	Registers the MAC addresses of the client devices to be allowed to log in. (e.g., 3D:F2:C9:A6:B3:4F).
Comment	Specifies a comment.

3-4-8 Printing Settings

These are the settings for printing and capturing screens on the HMI.

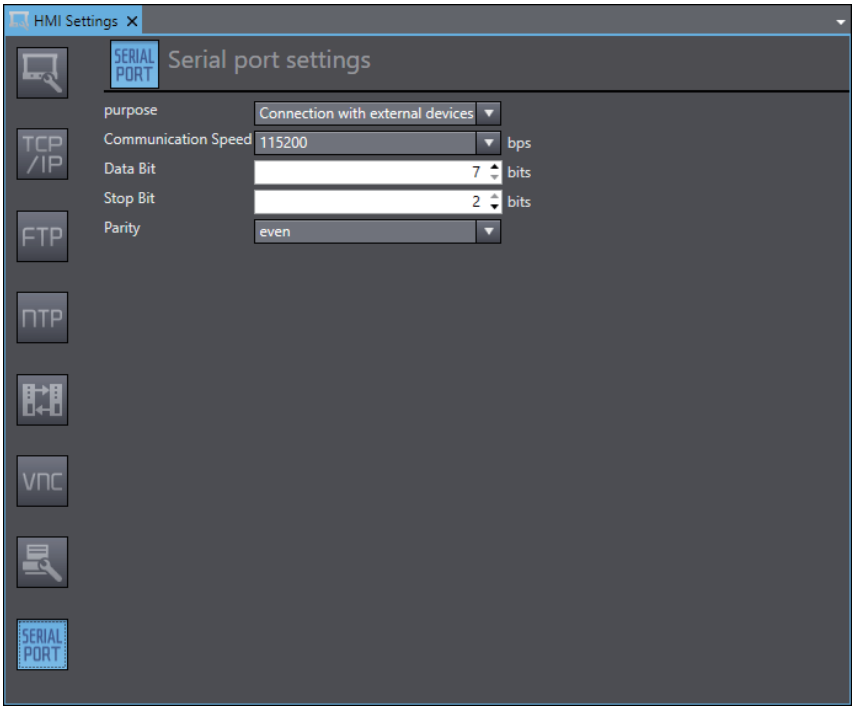
You can capture screens displayed on the HMI.



Item	Description
Printing Settings	
Color Mode	Sets one of the following options for the number of colors or use of highlighting: • Color • Grayscale • Reverse Grayscale
Print With Title	Sets whether to print the page title.

3-4-9 Serial Port Settings

These settings are for the serial port.



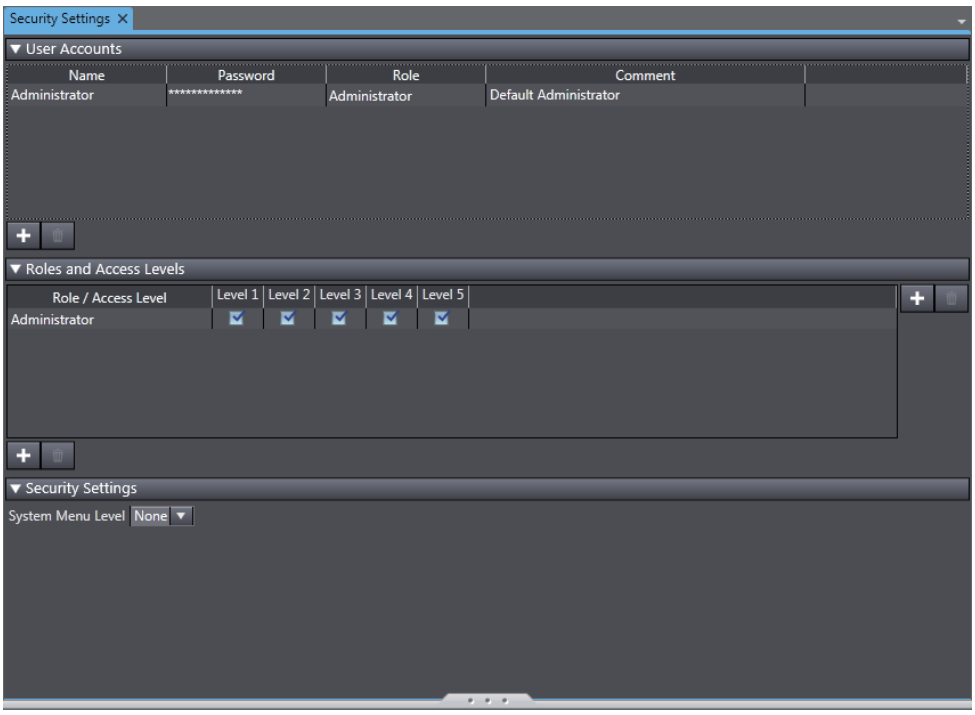
Item	Description
Serial port settings	
purpose	Select the purpose of the serial port. Select **Connection with external devices**.
Communication Speed	Select the communication speed of the serial port.
Data Bit	Set the data bit.
Stop Bit	Set the stop bit.
Parity	Select the parity.

3-5 Security Settings

These settings are used to restrict the operations that can be performed on the HMI and register accounts.

Setting Procedure for Security Settings

1 Double-click **Security Settings** under **Configurations and Setup** in the Multiview Explorer.
The Security Settings Tab Page is displayed in the Edit Pane.



Item	Description
User Accounts	Registers user accounts.
Roles and Access Levels	Sets the access level for each role.
Security Settings	Sets the level of operations to permit from the System Menu.

3-6 Troubleshooter

These settings are used for the troubleshooter for the NJ/NX/NY series.

For details, refer to following manuals.

- *NA-series Programmable Terminal Hardware User's Manual* (Cat. No. V117)
- *NA-series Programmable Terminal Hardware(-V1) User's Manual* (Cat. No. V125)
- *NA-series Programmable Terminal Hardware(-V2) User's Manual* (Cat. No. V128)

3-7 Language Settings

These settings are used for multi-language projects.

Project Language and System Language

There are two languages on the HMI.

- Project Language
  The project language is displayed for the project that you create.
- System Language
  The system language is displayed for the System Menu and for error messages displayed by the system.

The two language settings are managed as pairs. If you change the project language, the paired system language will also change.

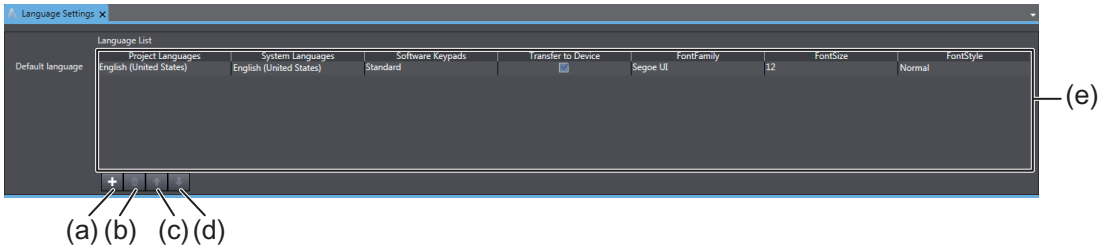
Additional Information

Changing the project language settings will affect the positions below. For items that have an affect only in subroutines, refer to the technical information on the effects of culture in Microsoft's Visual Basic.

- Date and time display format
- Decimal point and thousands separator
- Currency symbols
- Character string sort order

Language Setting Procedure

1 Double-click **Language Settings** under **Configurations and Setup** in the Multiview Explorer. The Language Settings Tab Page is displayed in the Edit Pane.



Number	Item		Description
(a)			Adds a language to the language list.
(b)			Deletes the selected language from the language list.
(c)			Moves the selected language one position up in the language list.
(d)			Moves the selected language one position down in the language list.
(e)	Language List		Displays a list of the languages. The language at the top of the list is treated as the default language.
		Project Languages	Sets the project language.
		System Languages	Sets the system language.
		Software Keypads	Sets the software keypad to be used.
		Transfer to Device	Sets whether or not the language is transferred to the HMI.
		FontFamily *1	Sets the font that is set by default when an object is created.
		FontSize *1	Sets the font size that is set by default when an object is created.
		FontStyle *1	Sets the font style that is set by default when an object is created.

*1. Only appears when the Runtime version is 1.11 or higher.

Additional Information

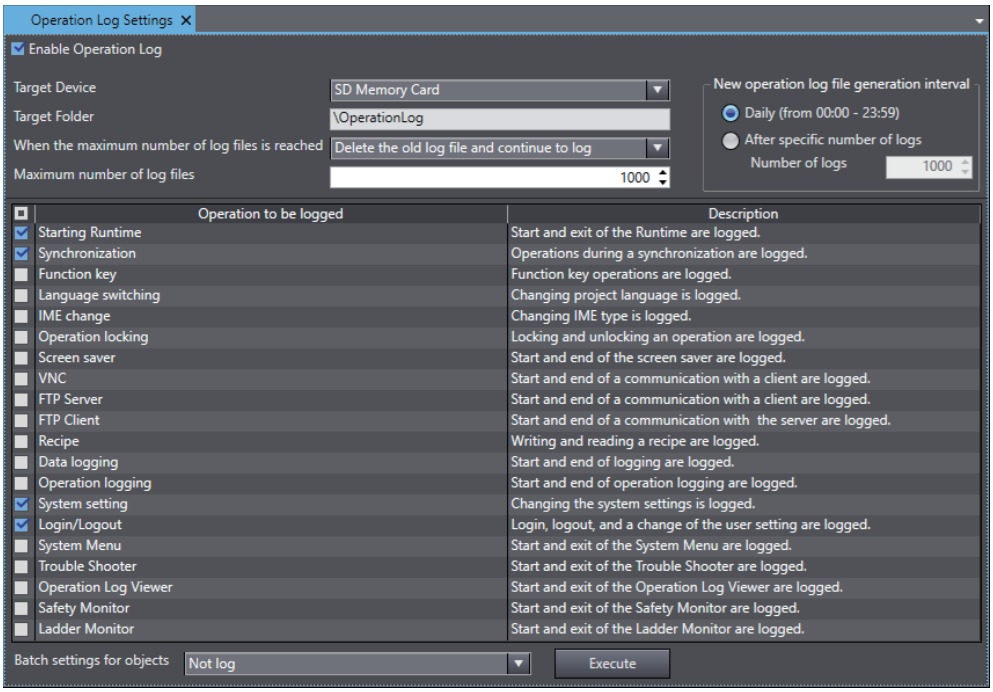
You can clear the selection of the *Transfer to Device* Check Box so that the language files are not transferred to the HMI. You can use this to delete unnecessary languages depending on the destination of the HMI.

3-8 Operation Log Settings

These settings are used for an Operation Log.

Additional Information

To use the Operation Log, the media to save data specified in the Operation Log Settings must be installed on the NA unit in a state where data can be written to it.



Item	Description
Start Operation Log	Activates or deactivates operation logging.
Target Device	Specifies the media to save operation logs.
Target Folder	Specifies the folder to save operation logs.
When the maximum number of log files is reached	Specifies the operation to execute when the number of log files reaches its upper limit.
Maximum number of log files	Specifies the upper limit of the number of log files.
New operation log file generation interval	Specifies the conditions for generating new log files.
Daily (from 00:00 - 23:59)	Generates a new log file daily at 00:00.
After specific number of logs	Generates a new log file when the number of logs in one file exceeds the number set for the number of logs (100 to 65000).
Operations to be logged	Sets the operations to be logged.
Batch settings for objects	Sets all “Operation Log” of the properties of the object.

Subject Operations of Operation Log

The following system operations of the NA unit are subject to operation logging. Select the Operation to be logged Check Box in Operation Log Settings to record these to Operation Log.

Operation to be logged	Operation	Description
Starting Runtime	Startup	When the Runtime starts up.
	Exit	When the Runtime stops.
Synchronization	Start Project Transfer	When a project data transfer with the NA unit starts.
	Complete Project Transfer	When a project data transfer with the NA unit completes.
Function key	Function Key operation	When a Function key is operated.
Language switching	Set language	When the project language is changed.
IME change	IME change	When the IME type is changed.
Operation locking	Disable touch input	When the input operation from the touch panel is disabled.
	Enable touch input	When the input operation from the touch panel is enabled.
Screen saver	Start screensaver	When the screen saver starts.
	End screensaver	When the screen saver is exited.
VNC	Connect to VNC server	When a user logs in from a VNC client.
	Disconnect VNC server	When a VNC client logs out.
FTP Server	Connect to FTP server	When a user logs in from an FTP client.
	Disconnect FTP server	When an FTP client logs out.
FTP Client	Connect with FTP Client	When a user logs into the FTP server using the FTP client function.
	Disconnect with FTP Client	When a user logs out from the FTP server using the FTP client function.
Recipe	Write recipe	When recipe data is written to the controller.
	Read recipe	When recipe data is loaded from the controller.
Data logging	Start Data logging	When data logging starts.
	Stop Data logging	When data logging stops.
Operation Logging	Start Operation logging	When operation logging starts.
	Stop Operation logging	When operation logging stops.
System setting	Change System settings	When settings for the following items of the NA unit are changed. • Time • Time zone • NTP Settings • System Languages • Ethernet • FTP Server • VNC Server • FINS • Screen Brightness
Login/Logout	Login	When a user logs in in a state where there is no user currently logged in.
	Logout	When a user logs out.
	Switch user	When a user logs in in a state where there is a user currently logged in.
System Menu	Show System Menu	When the System Menu starts up.
	Exit System Menu	When the System Menu is exited.
Trouble Shooter	Show Troubleshooter	When the Troubleshooter starts up.
	Exit Troubleshooter	When the Troubleshooter is exited.
Operation log Viewer	Show Operation Log Viewer	When the Operation Log Viewer starts up.
	Exit Operation Log Viewer	When the Operation Log Viewer is exited.

Operation to be logged	Operation	Description
Safety Monitor	Show Safety Monitor	When the Safety Monitor starts up.
	Exit Safety Monitor	When the Safety Monitor is exited.

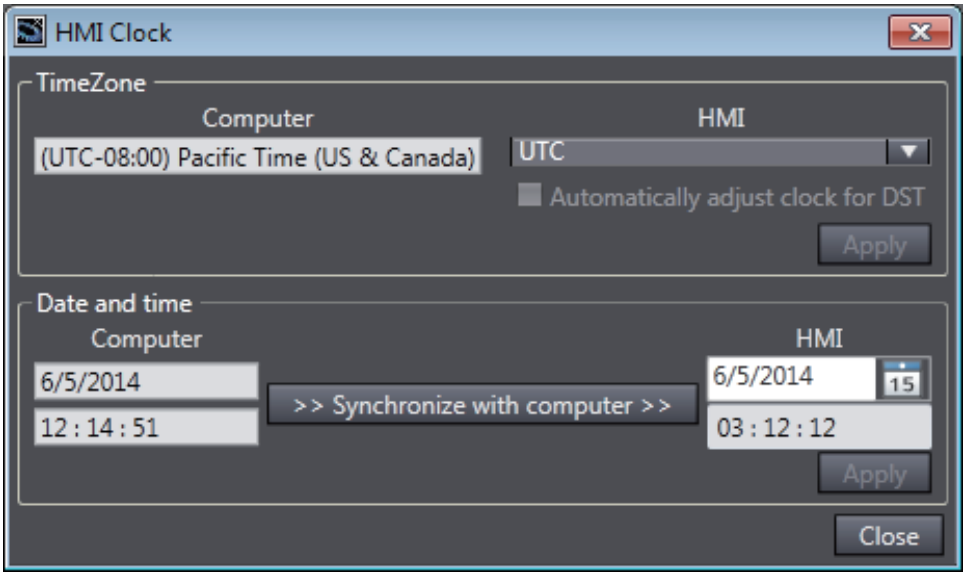
Subject Objects of Operation Log and Events to be Recorded

The subject objects of Operation Log and events to be recorded are shown below. Select the Operation Log Check Box in Properties of each object to record these to Operation Log.

Classification	Object	Events to be Recorded
Button	Button object	Click/Press/Release
	Momentary Button object	Click/Press/Release
	Set Button object	Click/Press/Release
	Reset Button object	Click/Press/Release
	Toggle Button object	Click/Press/Release
Standard controls	CheckBox object	Click/Checked/Unchecked
	Data Display object	Press/Release
	Data Edit object	Release
	DateTime object	Press/Release
	DropDown object	SelectionChanged
	Image object	Press/Release
	Label object	Press/Release
	ListBox object	SelectionChanged
	Radio Button object	Press/Checked/Unchecked
	Slider object	Release/LostMouseCapture/KeyUp/ValueChanged
	Tab Control object	SelectionChanged
	Text Box object	Press/Release
Shapes	Curve object	Press/Release
	Ellipse object	Press/Release
	Line object	Press/Release
	Polygon object	Press/Release
	Polyline object	Press/Release
	Rectangle object	Press/Release
	Triangle object	Press/Release

3-9 HMI Clock

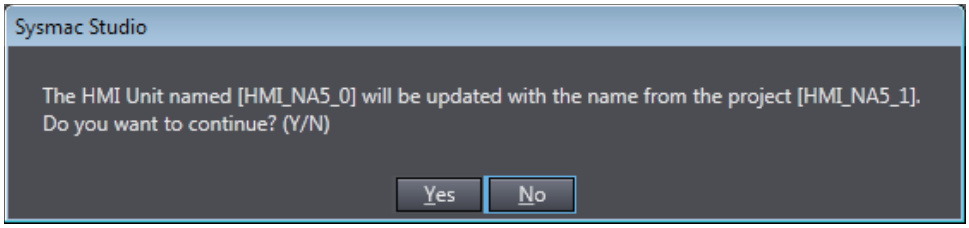
To set the clock in the HMI, select ***HMI Clock*** from the HMI Menu when you are online. Set the required items and click the **Apply** Button to update the information in the HMI.



Item	Description
Time Zone	
Computer	Displays the time zone of the computer.
HMI	Sets the time zone of the HMI
Automatically adjust clock for DST	Select this check box to enable automatically adjusting for daylight savings time.
Date and time	
Computer	Displays the current date and time on the computer.
Synchronize with computer	Updates the clock information on the HMI with the clock information from the computer.
HMI	Set the clock information on the HMI.

3-10 Updating the HMI Name

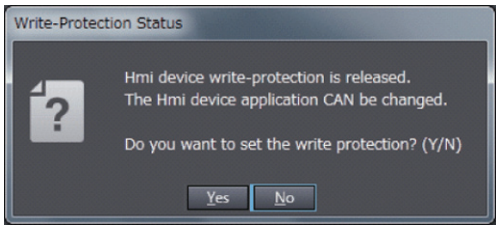
To change the HMI name, select ***Update HMI Name*** from the HMI Menu when you are online.
A confirmation dialog box is displayed. To update the HMI name that is set in the project, click the **Yes** Button.



3-11 Write Protecting the HMI

To write-protect the HMI, select ***Security - HMI Write Protection*** from the HMI Menu when you are online.

A confirmation dialog box is displayed. Click the **Yes** Button to write-protect the HMI.



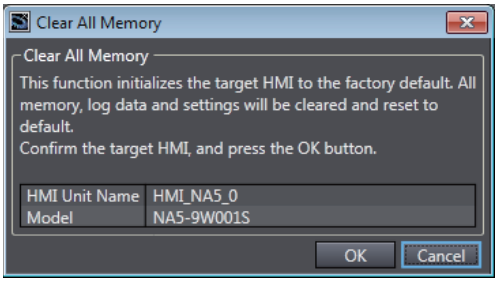
Only the projects contained within the NA Unit and in files used for transferring data from a media device are write protected. The project can still log data and write other data. This function cannot be used to set a password when data is write protected.

3-12 Clear All Memory

You can initialize the HMI.

When you perform the Clear All Memory operation, all data is cleared except for time data. To clear all memory in the HMI, select ***Clear All Memory*** from the HMI Menu when you are online.

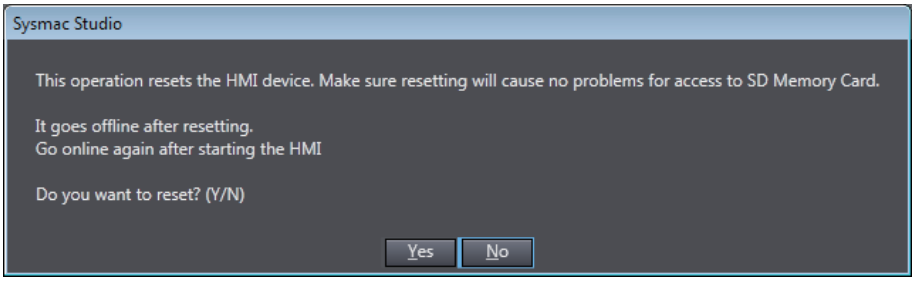
A confirmation dialog box is displayed. Click the **OK** Button to clear all memory.



3-13 Resetting the HMI

You can reset the HMI.

To reset the HMI, select ***Reset HMI Device*** from the HMI Menu when you are online. A confirmation dialog box is displayed. Click the **Yes** Button to reset the HMI.



4

Creating the HMI Application

This section describes how to create the HMI application (pages, variables, subroutines, etc.) with the Sysmac Studio.

4-1 Registering Variables 4-3
4-1-1 Variables 4-3
4-1-2 Registering Global Variables 4-4
4-1-3 Registering External Variables 4-6
4-1-4 Searching for Unused Variables 4-11
4-1-5 Attributes and Entry Methods for Global Variables 4-12
4-1-6 System-defined Variables 4-16
4-1-7 Subroutine Variables 4-20

4-2 Registering Data Types 4-21

4-3 Creating Pages 4-22
4-3-1 Displaying Pages 4-23
4-3-2 Registering Pages 4-23
4-3-3 Page Property Settings 4-25
4-3-4 Editing Pages 4-27

4-4 Setting Common Object Functions 4-35
4-4-1 Registering User Alarms 4-35
4-4-2 Setting Controller Events 4-36
4-4-3 Registration for Data Logging 4-39
4-4-4 Registering Data Groups 4-40
4-4-5 Registering Recipes 4-42
4-4-6 Registering Custom Keypads 4-44
4-4-7 Setting Global Events and Corresponding Actions 4-47
4-4-8 Registering Global Subroutines 4-49
4-4-9 Setting Up Resources 4-50
4-4-10 Setting Up IAG Resources 4-52
4-4-11 Registering Scaling 4-53

4-5 Subroutines 4-56
4-5-1 Subroutine Execution 4-57
4-5-2 Precautions on Internal Processing 4-61
4-5-3 Code Editor 4-62
4-5-4 Differences in Language Specifications 4-62

4-6 Search and Replace 4-63
4-7 Font Replacement 4-64
4-8 IAG Replacement 4-73
4-9 Device Replacement 4-76
4-10 Cross References 4-79
4-10-1 Cross References 4-79
4-11 Building 4-80
4-11-1 Building 4-80
4-11-2 Build Operation 4-81
4-12 Offline Comparison 4-82
4-13 Resource Usage 4-83

4-1 Registering Variables

4-1-1 Variables

A variable is a ‘container’ that holds data.

The HMI supports the following types of variables.

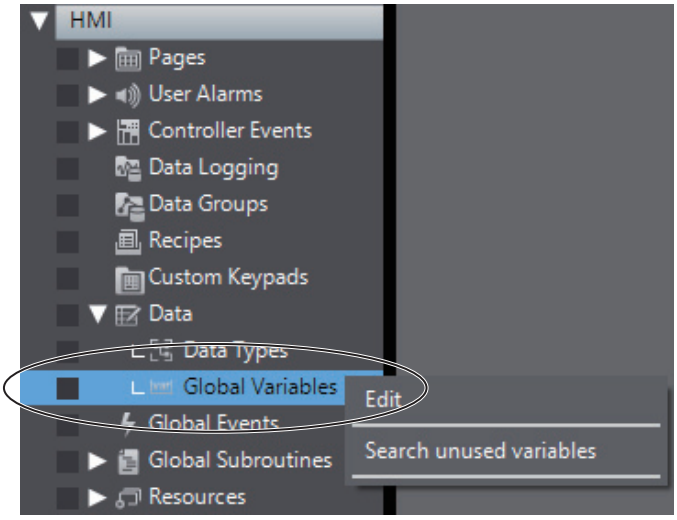
Variable type	Description
Global variables	Global variables are defined by the user and can be accessed from anywhere in the project. You can register up to 35,000 total in the entire project and up to 30,000 for each connected device.*1 Global variables are declared in the HMI global variable table. Global variables include external variables and internal variables, which are described below.
External variables	External variables are global variables that are used to access data in Controllers and other connected devices. External variables are assigned to device variables in the variable mapping.
Internal variables	An internal variable can be used only within the HMI. All global variables that are not external variables are internal variables.
System-defined variables	System-defined variables are provided in advance in the HMI. The names and all attributes are defined by the system. They have specific functions. You cannot change the variable names or any other attributes of these variables.
Subroutine variables	Subroutine variables are defined by the user and are used only within subroutines. Subroutine variables are declared in Dim statements in page subroutines or global subroutines. You can use all of the data types that are supported by Visual Basic.

*1. For the Runtime version 1.02 or lower, the registerable numbers are up to 35,000 in the entire project, and up to 20,000 for each connected device.

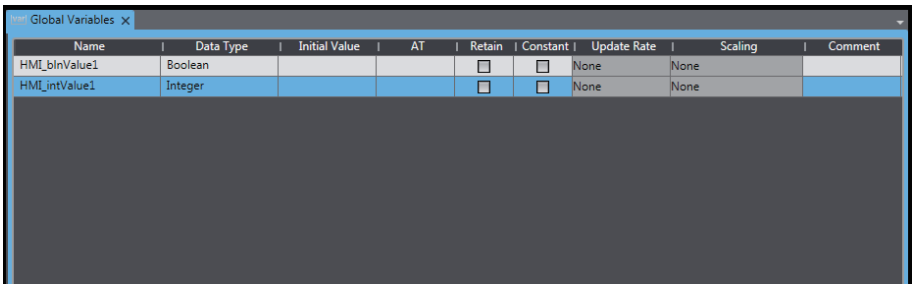
4-1-2 Registering Global Variables

● Opening the Global Variable Tab Page

Double-click **Global Variables** under **HMI - Data** in the Multiview Explorer. Or, right-click **Global Variables** under **HMI - Data** and select ***Edit*** from the menu.



The global variable table is displayed in the Edit Pane.



The basic Sysmac Studio operations for the global variable table generally apply to HMIs. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for the basic operations.

However, the following items are different. Refer to *4-1-5 Attributes and Entry Methods for Global Variables* on page 4-12.

Item	Description
Name	There are specific prohibited characters that apply to HMIs.
Data Type	There are specific data types for HMIs.
AT	You can set the device variables to which to assign the global variables. This setting is not used for internal variables.
Update Rate	You can select the interval at which to get values from the connected device. This setting is used only for external variables.
Scaling	You can select a conversion expression for the value obtained from the connected device. This setting is used only for an external variable.

Additional Information

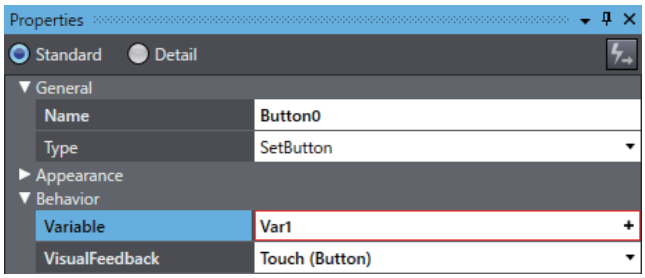
Even if you change the name of a previously registered global variable, the name of the variable accessed from objects or subroutines will not change.

Registering Global Variables from the Properties Window

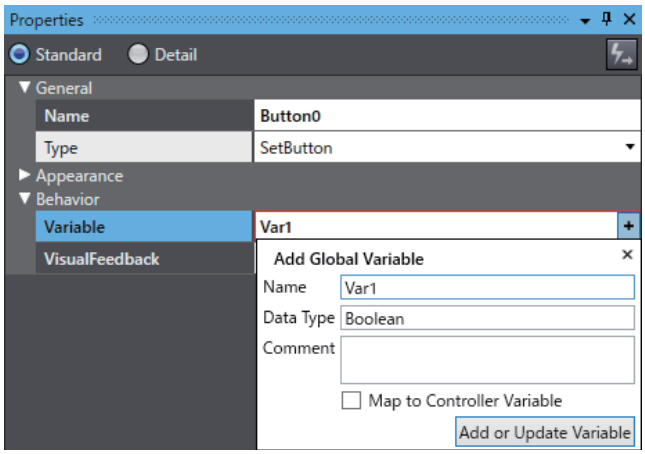
Use the following procedure to add a global variable from the Properties Window for an object.

Example: Using the Properties Window for a Set Button

1 Open the Properties Window for the object, and enter a variable name in the ***Variable*** Box under ***Behavior***.



2 Click the **+** Button to the right of the variable name that you entered.



A dialog box appears to add a global variable.

3 Set a data type in ***Data Type*** Box.

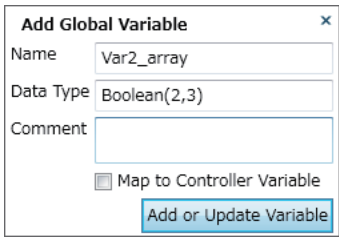
4 Enter a comment in the ***Comment*** Box.

5 Click the ***Add Variable*** Button.

This completes adding a variable.

Additional Information

- You can also create an array variable by specifying the data type as follows:



- You can add a variable the same way for an object for which an expression is specified under ***Behavior*** in the Properties Window.

4-1-3 Registering External Variables

External Variables

External variables are global variables that are used to access data in connected devices.

Creating External Variables

The following two methods can be used to create external variables.

- In the variable mapping, map a global variable to a connected device variable.
- When creating a global variable, also register a variable in the device.

When using the variable mapping, you can create a new variable or select a previously created global variable.

If you create a new external variable in the variable mapping, it will automatically be registered as a global variable.

Refer to *3-3 Mapping Variables* on page 3-7 for details on the variable mapping.

Attributes of External Variables

If you create a new external variable in the variable mapping, the following attributes are registered for it in the global variable table.

Refer to *4-1-6 System-defined Variables* on page 4-16 for details on the attributes of variables.

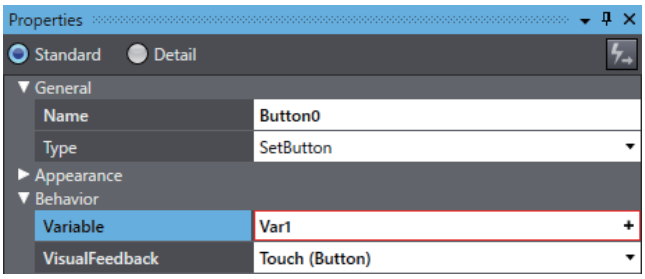
Attribute	Setting	Changes to settings
Name	Specified variable name	Possible.
Data Type	Refer to the *NA-series Programmable Terminal Device Connection User's Manual* (Cat. No. V119) for details on data type for the HMI global variables based on data types for device variable.	Possible.
Initial Value	None	Not possible.
AT	*Device_name.Device_variable_name*	Possible.
Retain	Non-retained	Not possible.
Constant	None	Not possible.
Update Rate	500 ms	Possible.
Scaling	None	Possible.
Comment	Specified comment	Possible.

Registering External Variables from the Properties Window

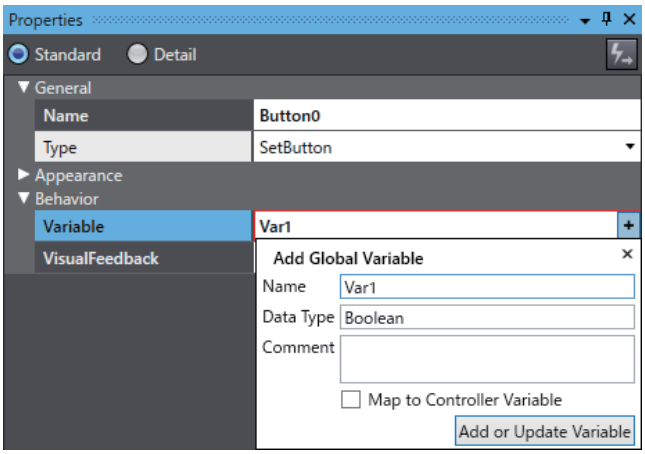
Use the following procedure to add an external variable that is mapped to a Controller variable from the object Properties Window.

Example: Using the Properties Window for a Set Button

1 Open the Properties Window for the object, and enter a variable name in the ***Variable*** Box under ***Behavior***.



2 Click the + Button to the right of the variable name that you entered.

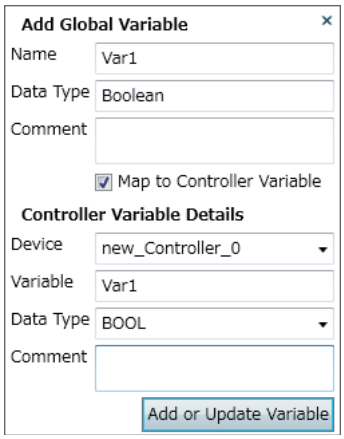


A dialog box appears to add a global variable.

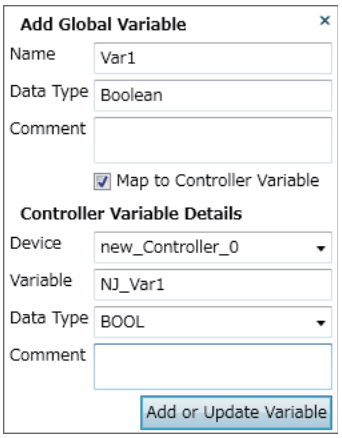
3 Set a data type in ***Data Type*** Box.

4 Enter a comment in the ***Comment*** Box.

5 Select the ***Map to Controller Variable*** Check Box.



6 Change the items in Controller Variable Details as needed.

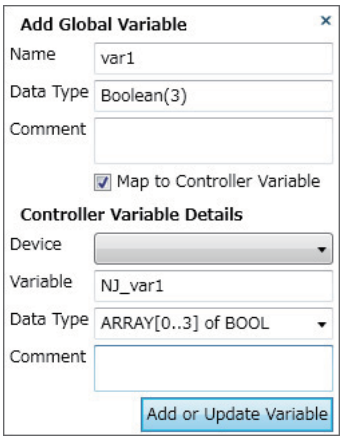


7 Click the add or update the global variable button.

Variables are added to the device and HMI, and are automatically mapped.

Additional Information

- If an Array[0..3] of BOOL-type NJ_var1 array variable has already been registered in the Controller, you can map the variable as an array variable by specifying the data type as follows:

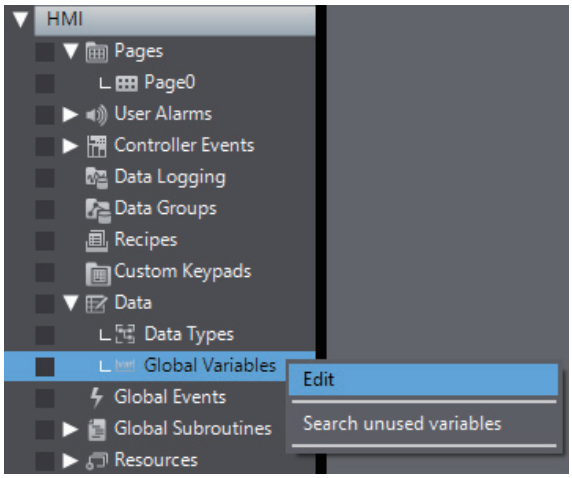


- You can map a variable in the same way for an object for which an expression is specified under ***Behavior*** in the Properties Window.
- When registering variables from properties, comments entered in the dialog box are registered into variables of each device.

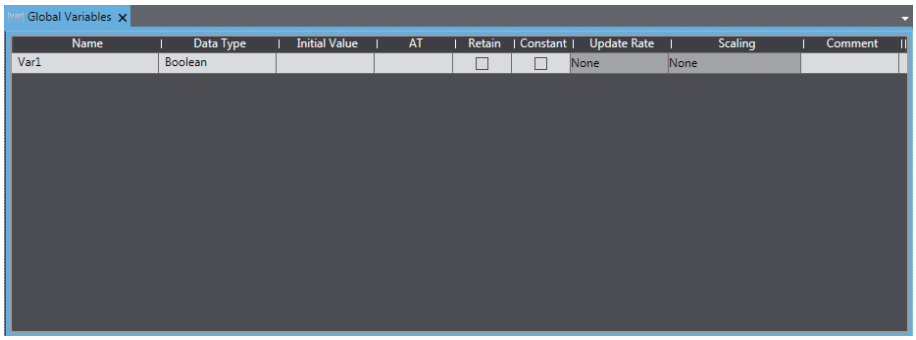
Registering External Variables from the Global Variable Table

Use the following procedure to add an external variable that is mapped to a Controller variable from the Global Variable Table.

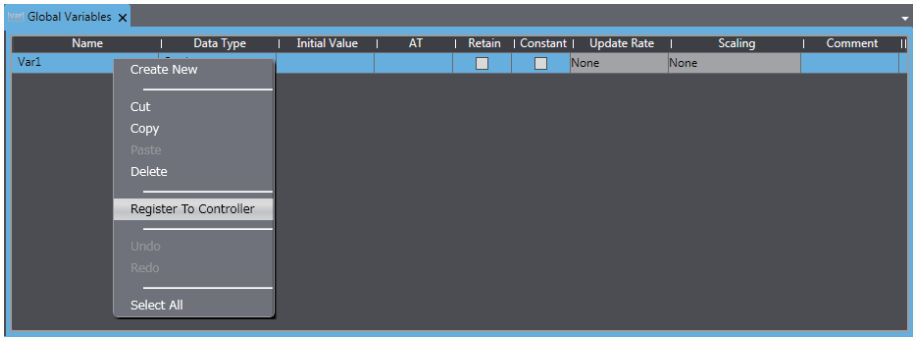
1 Double-click **Global Variables** under **HMI - Data** in the Multiview Explorer. Or, right-click **Global Variables** under **HMI - Data** and select **Edit** from the menu.



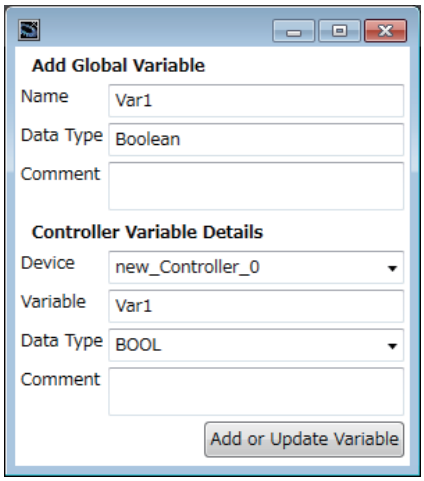
2 Add a variable.



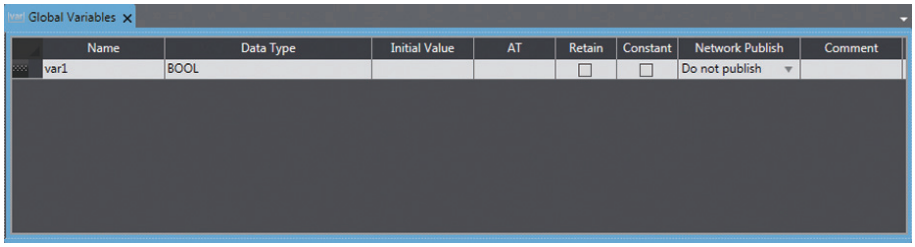
3 Right-click the variable and select **Register To Controller** from the menu.



4 Change the items in Controller Variable Details as needed, and click **Add or Update Variable**.



5 The global variable is registered in the controller.



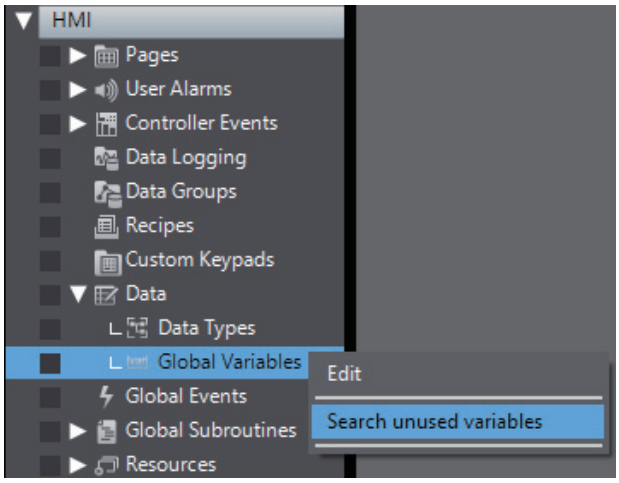
Additional Information

- You cannot register multiple variables at once.
- You cannot register structure, union, or enumeration variables that use data types defined on the device.
- When registering variables from the Global Variable Table, comments entered in the dialog box are registered into variables of each device.

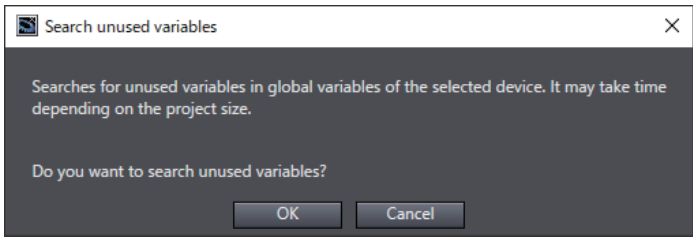
4-1-4 Searching for Unused Variables

To search for unused variables, right-click **HMI - Data - Global Variables**, and select **Search unused variables** from the menu that appears. Alternatively, select **Edit - Search unused variables** from the menu.

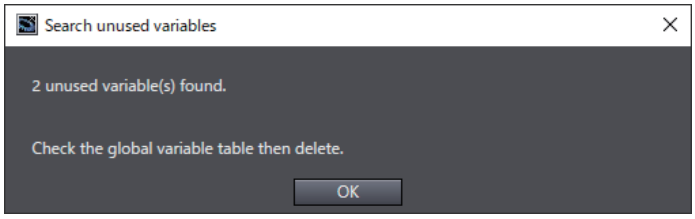
1 Right-click **Global Variables** under **HMI - Data** and select **Search unused variables** from the menu that appears.



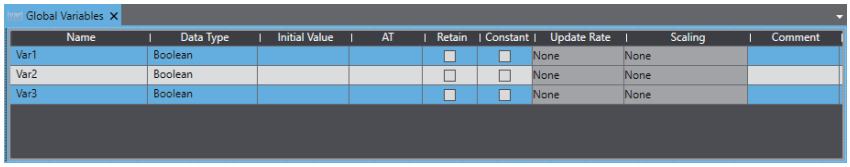
2 Click **OK**.



3 Click **OK**.



4 The unused variables are selected in the global variable table.



Additional Information

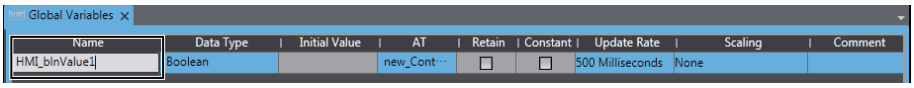
If the variable name of the subroutine or action parameter is enclosed in double quotations and specified as a text string, it will be handled as an unused variable.

4-1-5 Attributes and Entry Methods for Global Variables

This section describes attributes and entry methods for global variables.

Attributes of Global Variables

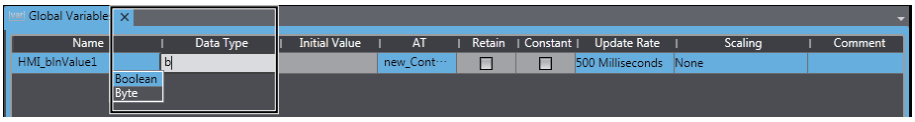
● Name Attribute



Enter a text string for the name. Observe the following restrictions.

- You cannot use any symbols except for underscores.
- The first letter cannot be a number.
- The names are not case sensitive.
- You cannot use names that start with “_HMI”, “_eHMI_”, “_sHMI_”, “_uHMI_”, or “_SyxSPC_”.
- You cannot use names that are reserved for the Sysmac Studio or Visual Basic.
- You cannot use more than 127 characters.
- You cannot use three underscores in a row.
- You cannot redundantly use names that are already used for a device name, global subroutine, keypad, page, or object name.

● Data Type Attribute



Enter a data type directly or select one from the list.

You can use the following data types, which are supported by Visual Basic. Arrays of the data types that are supported by Visual Basic are also supported. If you specify an array, specify the element number in parentheses after the data type in the same way as in Visual Basic.

Classification	Data type	Data type name	Size	Range of values
Basic data types	Boolean	Boolean	—	True or False
	Byte	Byte	1 byte	0 to 255 (unsigned)
	Char	Char	2 bytes	0 to 65,535 (unsigned)
	Date	Date	8 bytes	0:00:00 (midnight) on January 1, 0001 through 11:59:59 PM on December 31, 9999
	Decimal	Decimal	16 bytes	0 through +/-79,228,162,514,264,337,593,543,950,335 (+/-7.9...E+28) with no decimal point; 0 through +/-7.9228162514264337593543950335 with 28 places to the right of the decimal; smallest nonzero number is +/-0.0000000000000000000000000001 (+/-1E-28)
	Double	Double	8 bytes	-1.79769313486231570E+308 through -4.94065645841246544E-324 for negative values; 4.94065645841246544E-324 through 1.79769313486231570E+308 for positive values
	Integer	Integer	4 bytes	-2,147,483,648 through 2,147,483,647 (signed)

Classification	Data type	Data type name	Size	Range of values
Basic data types	Long	Long	8 bytes	-9,223,372,036,854,775,808 through 9,223,372,036,854,775,807 (9.2...E+18) (signed)
	SByte	SByte	1 byte	-128 through 127 (signed)
	Short	Short	2 bytes	-32,768 through 32,767 (signed)
	Single	Single	4 bytes	-3.4028235E+38 through -1.401298E-45 for negative values; 1.401298E-45 through 3.4028235E+38 for positive values
	String	String	Variable length*1	0 to approximately 2 billion Unicode characters (UTF-16)
	UInteger	UInteger	4 bytes	0 through 4,294,967,295 (unsigned)
	ULong	ULong	8 bytes	0 through 18,446,744,073,709,551,615 (1.8...E+19) (unsigned)
	UShort	UShort	2 bytes	0 to 65,535 (unsigned)
Derivative data types	TimeSpan	Structure that gives a time interval		
	Structures/ Unions/ Enumerations	This is a user-defined data type that integrates different data types into a single group. You can name the data type. Specify the name of the variables that use the user-defined data type, and one of the member variables included in the group.		

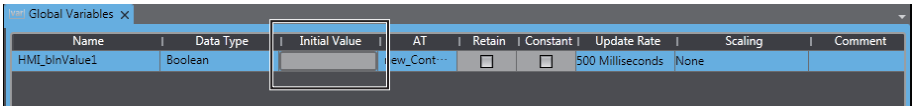
*1. The length is fixed only when a Retain variable is specified. The maximum number of characters is specified with the following notation.

String[**maximum_number_of_characters*][(*number_of_array_elements*)]

Values in square brackets [ ] may be omitted. If a Retain variable is specified and the maximum number of characters is not set, the maximum number of variables specified under **HMI Settings - Device Settings - Global Variables** applies.

Refer to the *NA-series Programmable Terminal Device Connection User's Manual* (Cat. No. V119) for information on which of the above HMI variable data types can be assigned to the data types of connected device variables for different connected devices.

Initial Value Attribute



Specify a value for the variable for one of the following situations when the Retain attribute is not specified.

- When the power supply is turned ON
- When you specify to initialize the value when the project is transferred

Enter a value directly or select an item from the list (the values in the list depend on the data type).

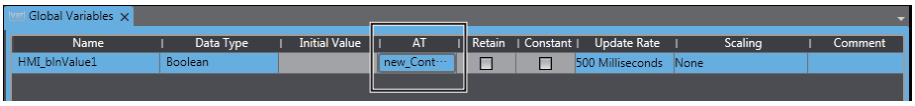
Select *None* for no initial value. You cannot specify the Initial Value attribute for an external variable.

Additional Information

If the box is left blank, the initial value of the variable will be as follows:

Data type	Initial value
Boolean, Byte, Char, Decimal, Double, Integer, Long, SByte, Short, Single, UInteger, ULong, or UShort	0
String	Blank
Date	January 1, 0001, 0:00:00 (AM 0:00)

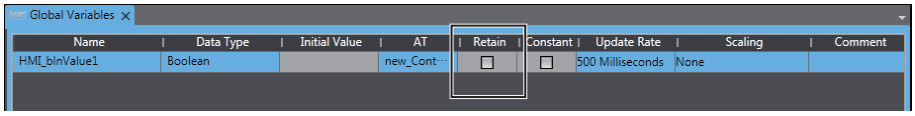
AT Attribute



This attribute is specified for external variables. Use the following format.

Device_name.Device_variable_name

Retain Attribute



Specify whether to retain the value of the variable in the following cases.

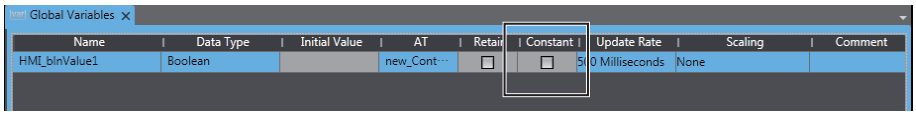
- When the power supply is turned ON
- When you do not specify to initialize the value when the project is transferred

You cannot specify the Retain attribute for an external variable, structure variable, or union variable.

You can specify up to 5,000 Retain variables (36,864 bytes). For version 1.08 or higher, a building error occurs if the maximum number or maximum size is exceeded. The size depends on the data type, as shown below. For String arrays, one element is counted as one variable. For example, Integer variable a(10) is counted as one variable, but String variable b(10) is counted as 10 variables.

Data type	Non-volatile memory consumption (bytes)
Boolean	1
Byte	1
Char	2
Date	8
Decimal	16
Double	8
Integer	4
Long	8
SByte	1
Short	2
Single	4
String	2 × Number of characters + 3
UInteger	4
ULong	8
UShort	2
TimeSpan	8
Structure	Cannot be set.
Union	Cannot be set.
Enumeration	4

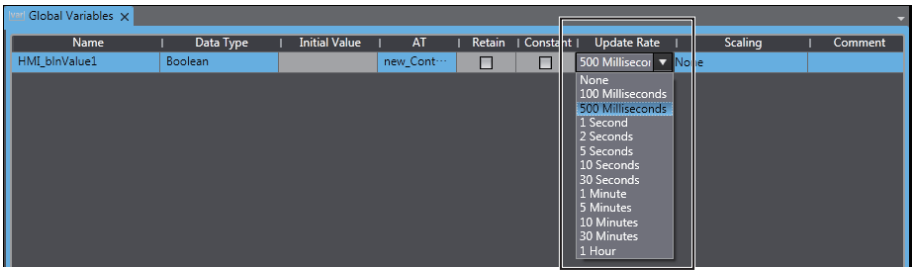
Constant Attribute



If you set the Constant attribute, you can set the initial value of the variable when the project is downloaded, but you cannot overwrite the value afterward.

You cannot specify the Constant attribute for an external variable.

● Update Rate Attribute



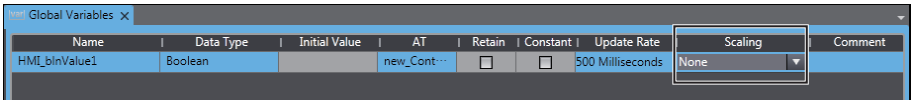
The update interval with the connected device is specified for external variables. If *None* is set for an external variable, communications are only performed with the connected device, such as when a value is written to the variable. Periodic communications are not performed. This attribute is not valid for internal variables.

● Scaling

Specify a conversion expression to apply to the values of the connected device for external variables. The external variable for which scaling is set will be cast as the Double type. Therefore, it is necessary to set Double in Data Type regardless of the data type of the device variable.

For information on setting the conversion expression, refer to *4-4-11 Registering Scaling* on page 4-53.

● Comment Attribute



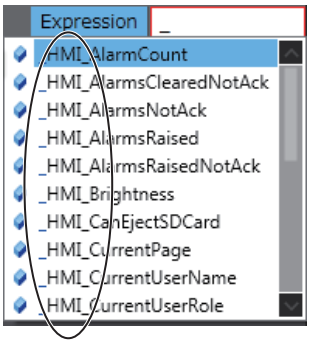
Enter a comment. When the device variable comment is changed for an external device, the change is automatically applied.

4-1-6 System-defined Variables

System-defined Variables

System-defined variables are internal variables that are pre-defined by the system. All system-defined variable attributes are fixed. The names and all other attributes of these variables are defined, and special functions are assigned to them. You cannot change the names or any other attributes.

- You can use the system-defined variables for an HMI as soon as you register an HMI in the project. It is not necessary to register system-defined variables in the global variable table.
- System-defined variables that are related to an HMI start with "_HMI_".



System-defined Variables

System-related Variables

Variable name	Meaning	Description	Data type	R/W
_HMI_Brightness	Brightness	Sets the brightness of the screen.	Integer	R/W
_HMI_CanEjectSDCard	Can Eject SD Card	Tells whether you can remove the SD Memory Card. True: Ejection is possible, False: Ejection is not possible.	Boolean	R
_HMI_ConnectedVNCClientCount	Connected number of VNC Client	Gives number of client via VNC.	Integer	R
_HMI_CurrentPage	Current Page Name	Sets the name of the currently displayed page.	String	R/W
_HMI_CurrentPageIndex	Current Page Number	Sets the number of the currently displayed page.	Integer	R/W
_HMI_DateTime	System Time	Gives the system clock time as the local time.	Date	R
_HMI_Hour	Current Hour	Gives the hour in the current time.	Integer	R
_HMI_IsBatteryLow	Low Battery Voltage	Gives True if the battery voltage has dropped below a specific level.	Boolean	R
_HMI_IsDataInput	Data Entry in Progress	Gives True when a data entry object is selected.	Boolean	R
_HMI_IsPageSwitching	Page Switching in Progress	Gives True while page switching processing is in progress and False after processing is completed.	Boolean	R
_HMI_IsScreenSaverActive	Screen Saver Status	Tells whether the screen saver is active. True: Active, False: Not active.	Boolean	R/W
_HMI_ManagedRAMInUse	Usage of Managed RAM	Gives the total bytes of managed RAM that is currently allocated to some process.	ULong	R
_HMI_Millisecond	Current Milliseconds	Gives the milliseconds in the current time.	Integer	R
_HMI_Minute	Current Minutes	Gives the minutes in the current time.	Integer	R
_HMI_RAMInUse	Usage of RAM	Gives the total bytes of RAM that is currently allocated to some process.	ULong	R
_HMI_RAMTotal	Total RAM	Gives the total bytes of RAM that the system is using or can use. This is not the currently usable amount of RAM.	ULong	R
_HMI_RunSignal	Run Signal	The value changes approximately every second while the HMI is operating.	Boolean	R
_HMI_Second	Current Seconds	Gives the seconds in the current time.	Integer	R

Alarm-related Variables

Variable name	Meaning	Description	Data type	R/W
_HMI_AlarmCount	Alarm Count	Gives the number of alarms retained in internal retained memory. The value is initialized when the alarm history is cleared.	Integer	R
_HMI_AlarmsClearedNotAck	Cleared Unacknowledged Alarms	Gives the number of alarms that are cleared but not acknowledged.	Integer	R
_HMI_AlarmsNotAck	Unacknowledged Alarm Count	Gives the number of alarms that are not acknowledged.	Integer	R
_HMI_AlarmsRaised	Current Alarm Count	Gives the number of current alarms.	Integer	R
_HMI_AlarmsRaisedNotAck	Unacknowledged Current Alarm Count	Gives the number of current alarms that are not acknowledged.	Integer	R

User and Security

Variable name	Meaning	Description	Data type	R/W
_HMI_CurrentUserName	Current User Name	Gives the name of the user that is currently logged in.	String	R
_HMI_CurrentUserRole	Current User Authority	Gives the authority (role) of the user that is currently logged in.	String	R

System Enumeration Variable Table

The following table shows the enumeration variables that are used to control the HMI.

Enumerators are specified for input enumeration variables instead of numerical values.

Data type name	Description
_eHMI_ALARM_STATUS	Specifies the alarm status. Acknowledged: Acknowledged All: All Cleared: Already cleared. ClearedAckNotRequired: Already cleared and no need to acknowledge. ClearedAcknowledged: Already cleared and acknowledged. ClearedUnacknowledged: Already cleared but not yet acknowledged. Raised: Raised. RaisedAckNotRequired: Raised and no need to acknowledge. RaisedAcknowledged: Raised and already acknowledged. RaisedUnacknowledged: Raised but not yet acknowledged. Unacknowledged: Not yet acknowledged.
_eHMI_BUZZER_TYPE	Specifies the buzzer mode. ContinuousBuzzer: Continuous buzzer IntermittentLongPulse: Intermittent long buzzer IntermittentShortPulse: Intermittent short buzzer

Data type name	Description
_eHMI_COLUMN	Specifies the column for sorting or filtering the user alarms viewer. AlarmCode: Alarm code All: All DateAndTime: Date and time Group: Group LoggedInUser: Logged-in user Message: Message Name: Name Priority: Priority Status: Status ElapsedTime: Elapsed time AdditionalInformation: Additional Information
_eHMI_EXPOSETTYPE	Specifies the search target of the Safety Monitor. All: All variables InputVariable: Input to Safety CPU OutputVariable: Output from Safety CPU
_eHMI_FTPFileType	Specifies the target when using an FTP client. All: All File: Files only Directory: Folders only
_eHMI_GRAPH_CURSOR	Specifies the cursor for graphs. Cursor1: Cursor 1 Cursor2: Cursor 2
_eHMI_GRAPH_DIRECTION	Specifies the direction of graphs. Left: Left Right: Right
_eHMI_GRAPH_ZOOM_TARGET	Specifies the expansion mode for graphs. Both: Both Horizontal: Horizontal direction only Vertical: Vertical direction only
_eHMI_IME_LANGUAGE	Specifies the IME input mode. CurrentSystemLanguage: System language Japanese: Japanese Korean: Korean SimplifiedChinese: Simplified Chinese TraditionalChinese: Traditional Chinese
_eHMI_KNOWN_PATH	Specifies the medium. SDCard: SD card USBMemory: USB memory

Data type name	Description
_eHMI_PRIORITY	Specifies the priority of the items to be displayed in the user alarms viewer. All: All UserFaultLevel1: User alarm level 1 UserFaultLevel2: User alarm level 2 UserFaultLevel3: User alarm level 3 UserFaultLevel4: User alarm level 4 UserFaultLevel5: User alarm level 5 UserFaultLevel6: User alarm level 6 UserFaultLevel7: User alarm level 7 UserFaultLevel8: User alarm level 8 UserInformation: Information
_eHMI_SERVICE_STATE	Specifies the state of the service. Running: Running Suspended: Suspended
_eHMI_SERVICE_TYPE	Specifies the type of the service. VNCServer: VNC server
_eHMI_SORT_ORDER	Specifies the sort order of the user alarms viewer. Ascending: Ascending Descending: Descending
_eHMI_TROUBLESHOOTER_PAGE	Specifies the screen displayed for the troubleshooter. ActiveControllerEvents: Controller error list screen ActiveUserEvents: User error list screen ControllerEventLogs: Controller error log list screen UserEventLogs: User error log list screen

System Structure Variable Table

The following table shows the structure variables that are used to control the HMI.

Data type name	Variable	Data type	Description
_sHMI_VNC_CLIENT_INFO	Comment	String	Specifies a comment.
	IPAddress	String	Specifies an IP address.
	MACAddress	String	Specifies a MAC address.

4-1-7 Subroutine Variables

Subroutine Variables

Subroutine variables can be used only with subroutines. Subroutine variables conform to Visual Basic specifications.

Refer to the *NA-series Subroutine Reference* for details.

4-2 Registering Data Types

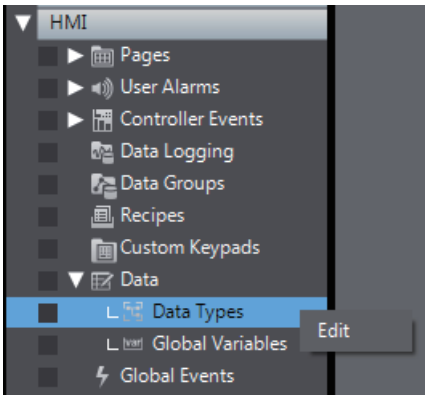
Registering Data Types

You can combine existing data types to define and register a new data type.

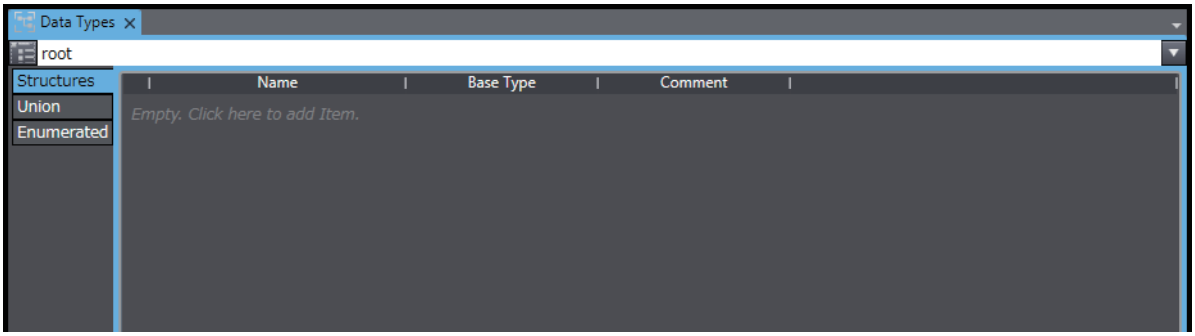
To use a data type that was registered with the Data Type Editor, just select it from the data types in the Variable Editor. You can define data types that are structures or enumerations.

Opening the Data Types Tab Page

1 Double-click **Data Types** under **HMI - Data** in the Multiview Explorer, or right-click **Data Types** under **HMI - Data** and select Edit from the menu.



The Data Type Editor is displayed.



The basic operating procedures for the Data Type Editor are the same throughout the Sysmac Studio. For basic usage instructions, refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504). The differences are as follows:

- You cannot register unions.
- You cannot set the offset type, offset bytes, or offset bits.

Additional Information

You cannot refer to data type of the connected device by subroutine or other means. If you want to use enumerations defined by the connected device, use variables mapped as enumerated variables.

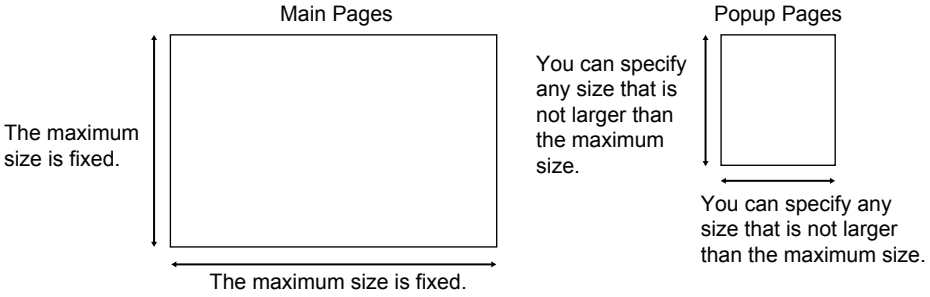
4-3 Creating Pages

One page represents one screen in the HMI project.

You can arrange various objects on a page to achieve the required functions.

There are two types of pages, as described in the following table.

Page Type in the Page Behavior	Sizes (width and height)	Application
Main	Fixed for each model	Basic pages Pages can be layered. You can specify a transparent color and specify a background page.
Popup	You can set the page sizes as long as they are not larger than the main page size.	Warning dialog boxes and other applications



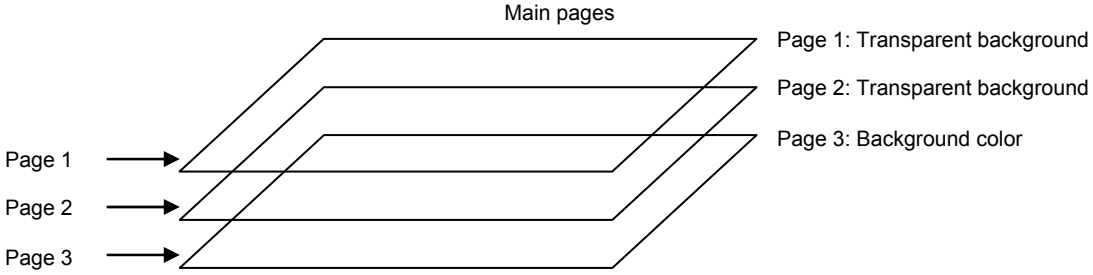
● Main Pages

The main pages are the basic pages. You can create applications to call popup pages from main pages or move to other main pages. You can place up to 450 objects on one main page. You cannot place more than one Trend Graph, Broken-line Graph, or Media Player object on one page. Note that in Soft-NA, if Runtime version 1.16 or higher is used, up to a total of two graphs, a Trend Graph and a Broken-line Graph, can be arranged on one page.

You can create layers of main pages.

You can specify a background page to make the top layer transparent and place a specified page underneath it. You can layer up to five pages including the main pages.

You can create a common page, e.g., with a toolbar, to display with all other pages and then specify it as the background page so that you do not have to create it more than once.



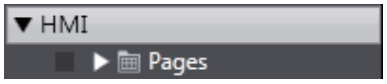
● Popup Pages

Popup pages are called from main pages. Use them to display warnings and other information. You can place up to 450 objects on one popup page. The following limitations apply to popup pages.

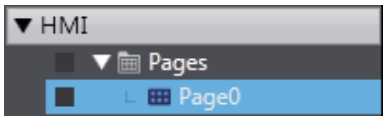
- You cannot display multiple popup pages at the same time.
- You cannot place Trend Graph, Broken-line Graph, Tab Control, or Media Player objects on a popup page.
- The background color Alpha is disregarded. For this reason, if a transparent color is set, it will be drawn as a non-transparent color.

4-3-1 Displaying Pages

Click **Pages** under **HMI** in the Multiview Explorer.



The pages are displayed under **Pages**.

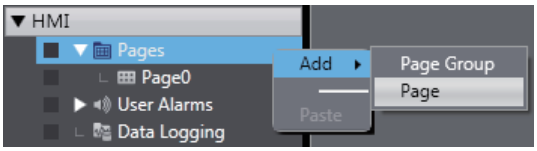


4-3-2 Registering Pages

The project contains one page by default, and you can add more pages as required.

Registering New Pages

1 Right-click **Pages** under **HMI** and select ***Add - Page*** from the menu.

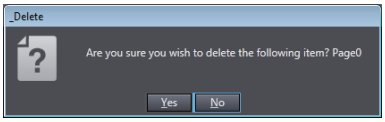


A new page, Page1, is added under **Pages**.

Deleting Pages

1 Right-click the page to delete in the Multiview Explorer and select ***Delete*** from the menu.

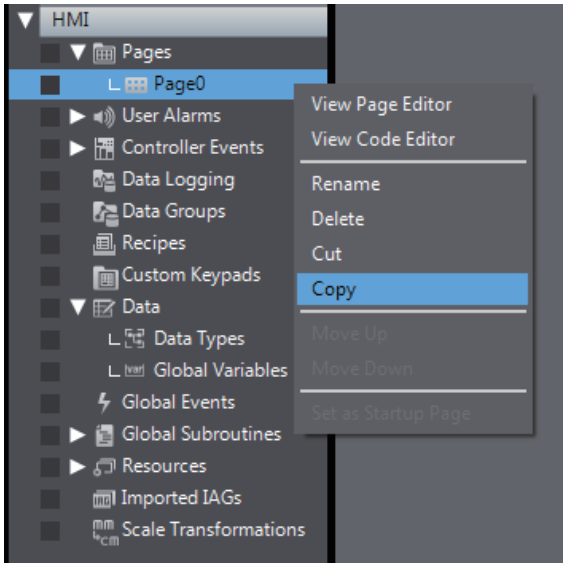
A deletion confirmation dialog box is displayed.



Click the **Yes** Button. The page is deleted.

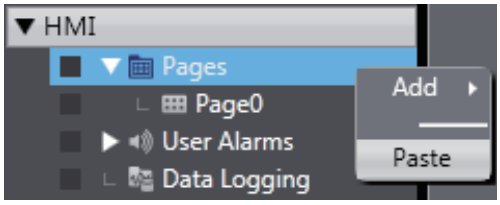
Copying and Pasting Pages

1 Select the page to copy in the Multiview Explorer and press the **Ctrl + C** Keys. Or, right-click the page and select ***Copy*** from the menu.

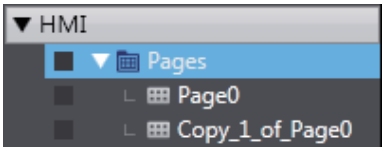


The page is copied.

2 Select **Pages** in the Multiview Explorer and press the **Ctrl + V** Keys. Or, right-click **Pages** and select ***Paste*** from the menu.



The copied page is registered with “Copy_1_of_” added to the front of the name of the page that was copied. All of the objects on the page are also copied.



Additional Information

- You can change the names of pages. Right-click the page and select ***Rename*** from the menu.
- If you copy a page from another product and a page with the same name already exists, the page will be named as a copy.
- When the page name is changed, the change is executed across the entire project. The replace results are displayed in the Search and Replace Results Tab Page.

4-3-3 Page Property Settings

You can set properties for the pages.

Setting Properties

1 Right-click in a location in the Edit Pane where there is no object and select ***Properties***.

2 Set the properties for the page in the Properties Window that is displayed.

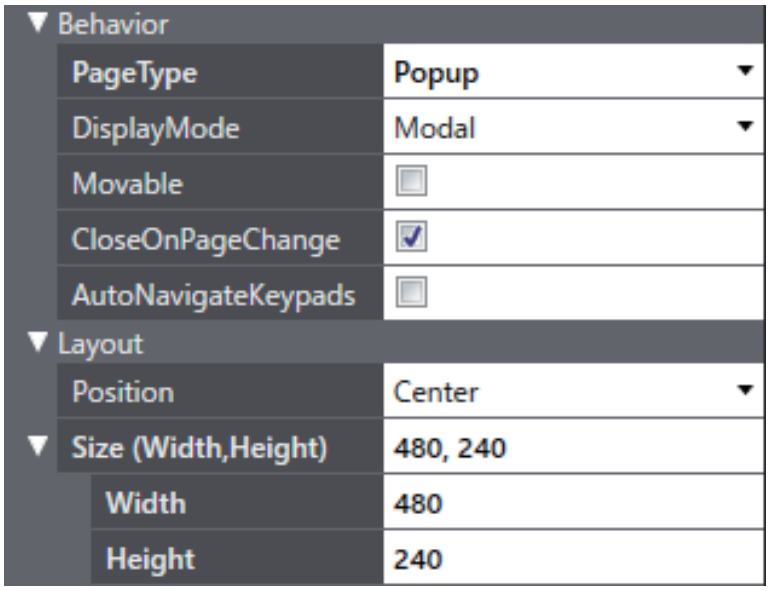
Item classification	Property	Description	Main pages	Popup pages
General	Name	Sets the name of the page. All names must be unique.	OK	OK
	Type	The object type is displayed. **Page** is displayed for a page.	Display only	Display only
Appearance	Background Color	Sets the background color. A transparent page will automatically be set if a background page is specified.	OK	OK
Behavior	PageType	Sets the type of page.	OK	OK
	BackgroundPage	Sets a page that is overlapped as the background.	OK	---
	AutoNavigateKeypads	If you select this check box, the keypad is automatically displayed according to the Data Input Order settings.	OK	OK
	DisplayMode	If you select **Modal**, no operations are possible on the main page while a popup page is displayed. If you select **Modeless**, main page operations are possible while a popup page is displayed.	---	OK
	Moveable	If you select this check box, the popup page can be moved.	---	OK
	CloseOnPageChange	If you select this check box, the popup page is automatically closed when the page is changed.	---	OK

Item classification	Property	Description	Main pages	Popup pages
Layout	Position (Let, Top)	Sets the display position of the popup page. If you select **Custom**, the coordinates of the top left corner of the page are specified with (Left,Top).	---	OK
	Size (Width, Height)	Sets the size of the page. The maximum size is the physical size of the screen.	Display only	OK

OK: Can be edited, Display only: Cannot be edited, ---: Property does not exist.

Click the arrowhead ( ) at the right of each item to make selections.

If you select a **Popup** for the **Behavior** setting, you can increase and decrease the **Size (Width, Height)** of the **Layout** with the arrowheads on the right side of the box after selecting the Detail mode and the size to change (left: width, right: height).



4-3-4 Editing Pages

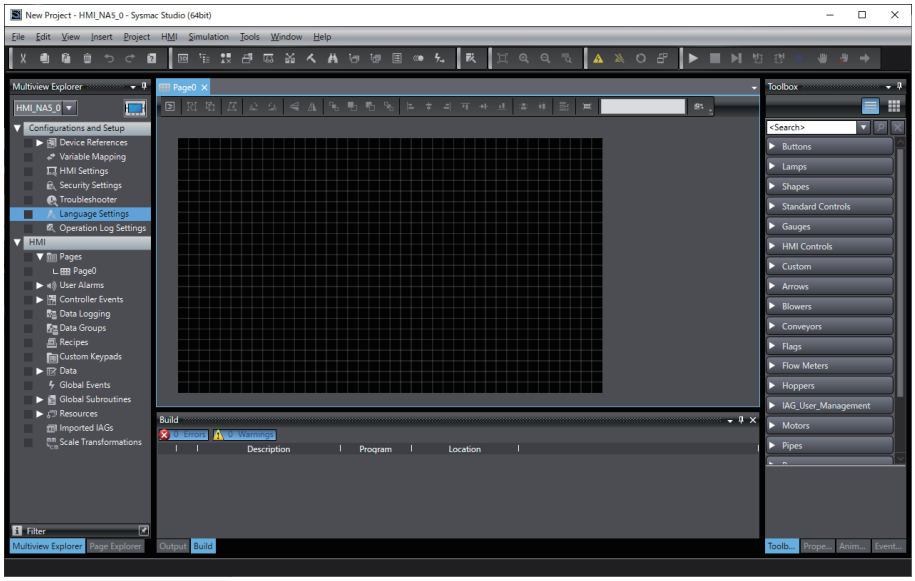
You can arrange objects on the pages to achieve the required functions.

Additional Information

Sysmac Studio may end abnormally if many objects grouped together are copied and pasted. Take care to prevent the number of objects from excessively exceeding the maximum number per page by pasting.

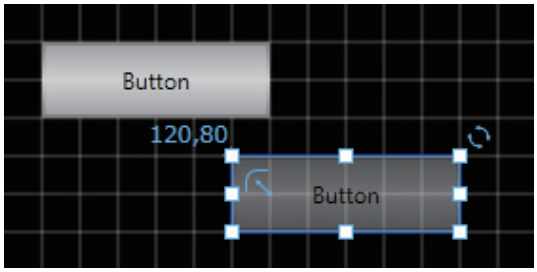
Positioning Objects

Drag objects from the Toolbox to position them on a page.



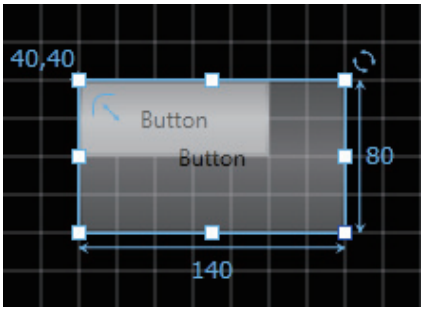
Moving Objects

To move an object, click the object and drag it while the cross cursor is displayed.



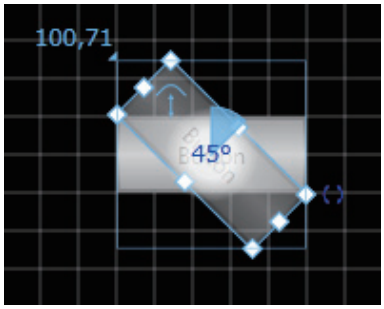
Resizing Objects

Click one of the resize handles around the object and drag it while the resizing cursor is displayed.



Rotating Objects

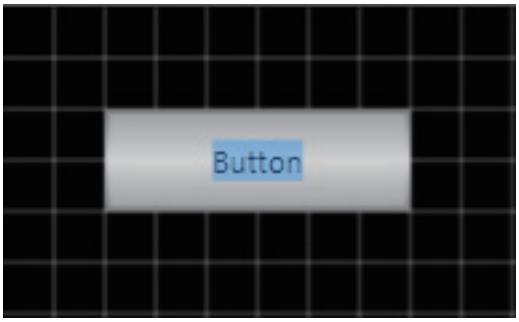
Click the rotation handle to the upper right of the object and drag it while the angle is displayed in the center of the object.



Changing Object Labels

You can directly change the label of an object by selecting it and performing one of the following operations.

- Press the F2 Key.
- Double-click the left mouse button.
- Right-click and select ***Edit***.
- Click the ***Edit*** Button on the Toolbar.



The following objects are supported for this function:

- All the button objects including Buttons, Momentary Buttons, Set Buttons, Reset Buttons, and Toggle Buttons
- All the lamp objects including Bit Lamps and Data Lamps
- Label objects
- Text Box objects
- Check Box objects
- Radio Button objects

Duplicating Objects

You can create the specified number of duplicate objects in the vertical and/or horizontal direction based on the selected object or objects.

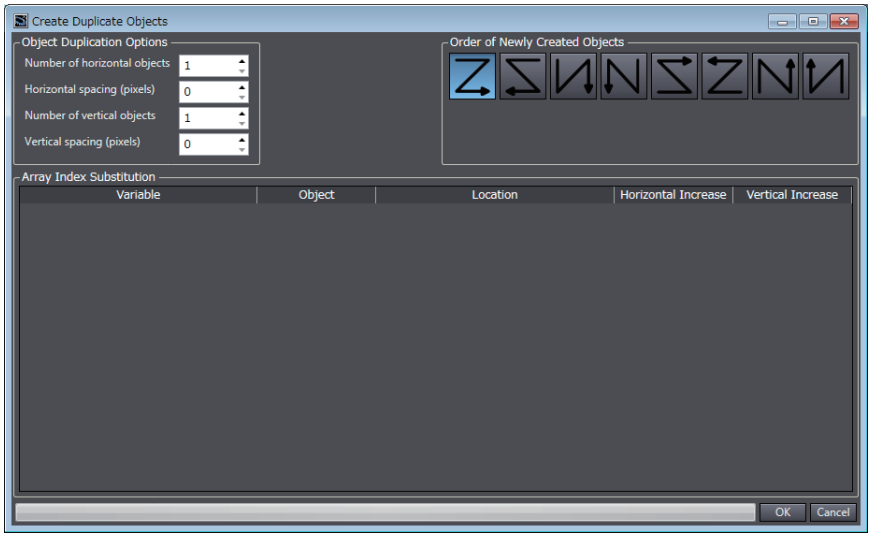
You can also duplicate the object or objects by specifying an offset value for an array variable.

1 Select the object to duplicate.

To duplicate multiple objects at the same time, select all the objects to be duplicated.

2 Right-click the selected object and select ***Create Duplicate Objects***… from the popup menu.

A dialog appears in the Edit Pane to specify object duplication. Specify the number of duplicate objects and the direction.



If a Value Other Than 0 Is Specified in the ***Horizontal Increase*** and ***Vertical Increase*** Box:

If an array variable has been specified for the object, you can duplicate the object by adding the specified offset value to the element number.

Set As Default

Changes the initial value of the object's properties.

Default value registration applies only to frequently used objects, such as buttons. Properties that have a different setting for each object, such as variables, cannot be registered as a default value.

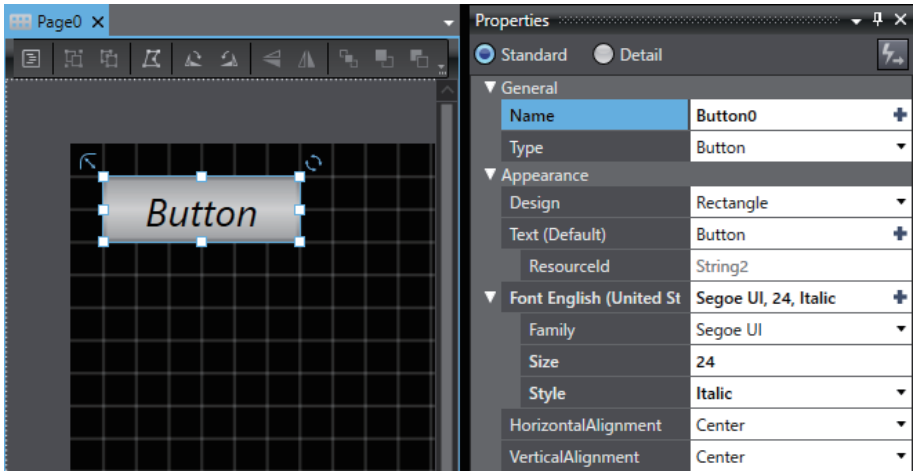
Additional Information

- The object default value is registered for each object type and managed as a project.
- The default values that have been set in a project property will be applied when the object properties are set to initial values. For example, when an object is copied from one project to another project, the default values set in the destination project will be applied to the initial values of the object properties.
- In the case of version control, the default values are out of scope of the version control. Although the functions of the default values can be used, history control is not supported.

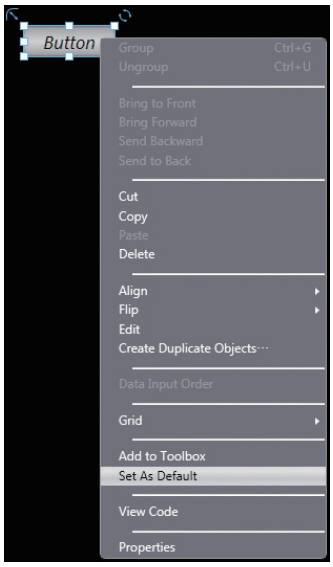
Changing default value

Follow the procedure below to set default value. The default value will be applied to the object's properties if it has been set to the property details.

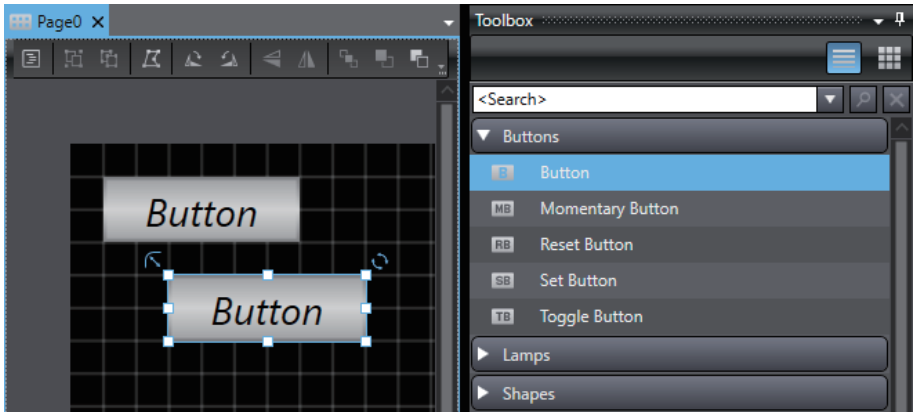
1 Set the object properties to the state you wish to set as default value.



2 Right-click the object and select ***Set As Default*** in the menu that appears.



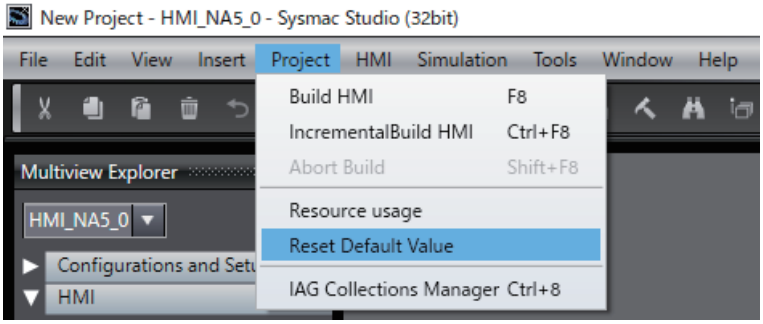
3 An object with the same settings can be created by dragging and dropping the object from the tool box.



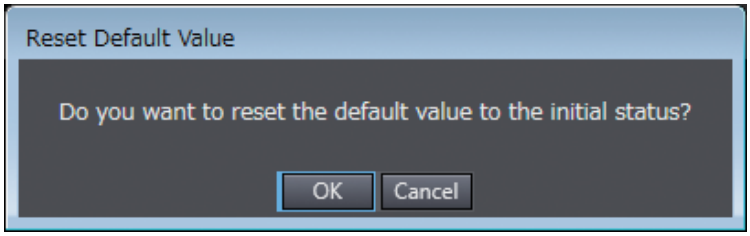
● Returning the default value to the initial state

Follow the procedure below to return the default value to the initial state. Note that this procedure returns the default value of all objects to the initial state.

1 Select ***Reset Default Value*** in the ***Project*** menu.



2 A confirmation dialog box appears. Click ***OK***.



Editing with the Toolbar

The functions that are commonly used to edit pages are provided in the Toolbar.

Code View

This button displays the Code Editor.

Group

This button creates a group of objects.

Ungroup

This button ungroups previously grouped objects.

Edit

This button is used to edit graphic objects, labels of Button or Lamp objects, and other objects.

Rotate Right 90 Degrees

This button rotates an object 90° clockwise.

Rotate Left 90 Degrees

This button rotates an object 90° counterclockwise.

Flip Vertical

This button flips a graphic object vertically.

Flip Horizontal

This button flips a graphic object horizontally.

Bring to Front

This button moves an object to the front.

Bring Forward

This button moves an object toward the front.

Send Backward

This button moves an object toward the back.

Send to Back

This button moves an object to the back.

Align Left

This button aligns the left edges of the selected objects.

Align Center Horizontal

This button aligns the centers of the selected objects vertically.

Align Right

This button aligns the right edges of the selected objects.

Align Top

This button aligns the top edges of the selected objects.

Align Center Vertical

This button aligns the centers of the selected objects horizontally.

Align Bottom

This button aligns the bottom edges of the selected objects.

Distribute Horizontally

This button positions the centers of the objects at equal distances horizontally.

Distribute Vertically

This button positions the centers of the objects at equal distances vertically.

To Previous Tab Page

This button displays the previous tab page.

To Next Tab Page

This button displays the next tab page.

Data Input Order Setting

This button changes the mode to the one to set data input order, and specifies the input order of objects.

Variable Display Mode

This button switches to the mode that displays the variables or expressions set for the object.

Variable Name

In the variable display mode, the variable name or expression is displayed for the selected object.

Simulation Mode

This button changes the mode to the one to confirm object state.

0 Simulation Index

This button switches the index and lets you confirm the display state of objects.

English (United States) Change Language

This box changes the project language that is displayed in the Edit Pane. When Display AllLanguages is selected, text in all languages is displayed at once.

4-4 Setting Common Object Functions

This section describes the settings for functions that are shared by the entire HMI project.

The following functions are provided.

- User alarms
- Controller events
- Data logging
- Data groups
- Recipes
- Custom keypads
- Global events
- Global subroutines
- Resources
- Imported IAGs
- Scale Transformations

Additional Information

If the text string contains a control code, unexpected behavior may occur when it is used in an object. Be careful that a control code is not included when copying and pasting from other applications.

4-4-1 Registering User Alarms

User Alarms

You can specify conditions for specified variables to display user messages when the conditions are met.

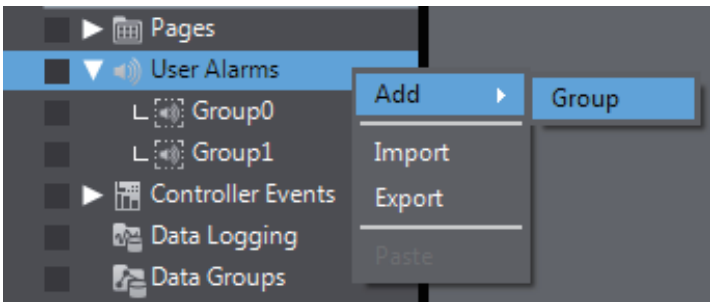
You can use this to record information when errors occur, when operation is started, etc.

Registering a New User Alarm

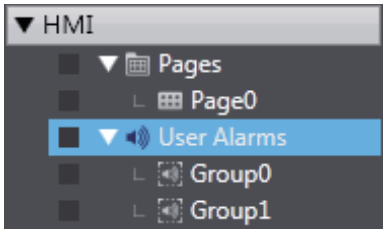
You manage user alarms by group.

The groups are displayed in the User Alarms Viewer and are helpful in organizing information.

1 Right-click **HMI** under **User Alarms** in the Multiview Explorer and select ***Add - Group*** from the menu.

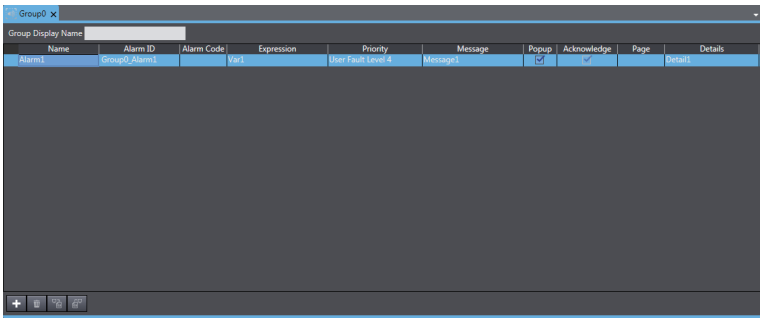


Group1 is added under **User Alarms.**



2 Double-click the new group.

A tab page to edit the group is displayed in the Edit Pane so that you can register user alarms.



Additional Information

You can import or export the user alarm settings using the Excel workbook format. Refer to *11-4 Import/Export User Alarm* on page 11-7.

Deleting, Copying, and Pasting Groups

You can delete, copy, and paste groups using the same procedures as those that you use for pages.

4-4-2 Setting Controller Events

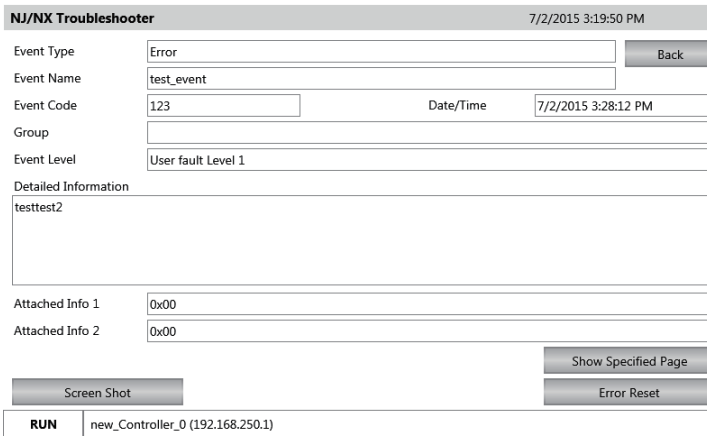
Events

Events are the errors, changes of states, or user-defined occurrences that occur in the controller.
For details, refer to the *NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501)*, *NJ/NX-series Troubleshooting Manual* (Cat. No. W503), and *NY-series Troubleshooting Manual* (Cat. No. W564-E1).

User Events

User events include the errors and information defined by users, and can be triggered by commands from the controller.

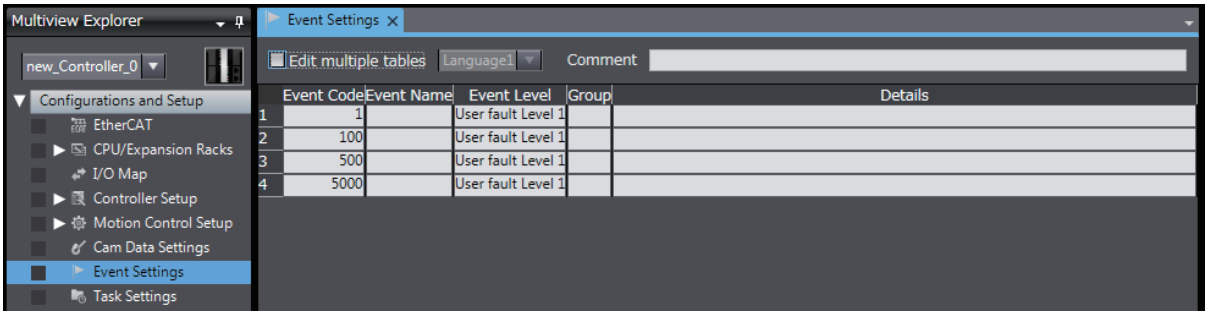
You can specify on the NA's NJ/NX/NY-series troubleshooter, the user screens to be switched to from the troubleshooter screen when a detailed user event of the controller is displayed.



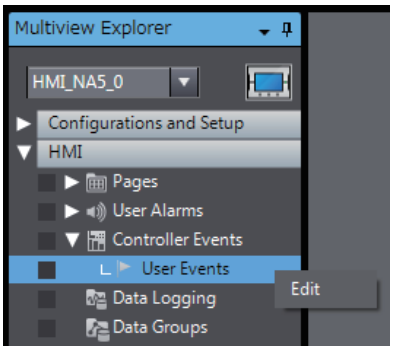
Registering User Events

To switch a detailed user event screen of the troubleshooter to a user screen, you must specify the event table for the controller and the transition destination screen that corresponds to each event.

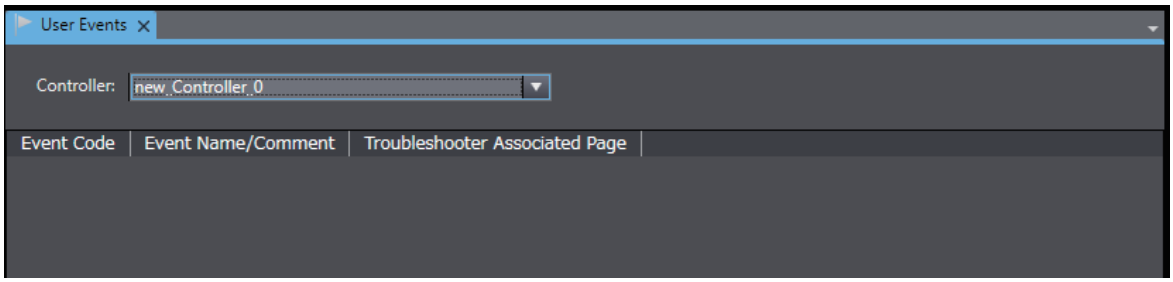
1 Specify the event table of the user events in the project for the controller.



2 Double-click **User Events** under **HMI - Controller Events** in the Multiview Explorer of the project for the HMI, or right-click **User Events** under **HMI - Controller Events** and select ***Edit*** from the menu.

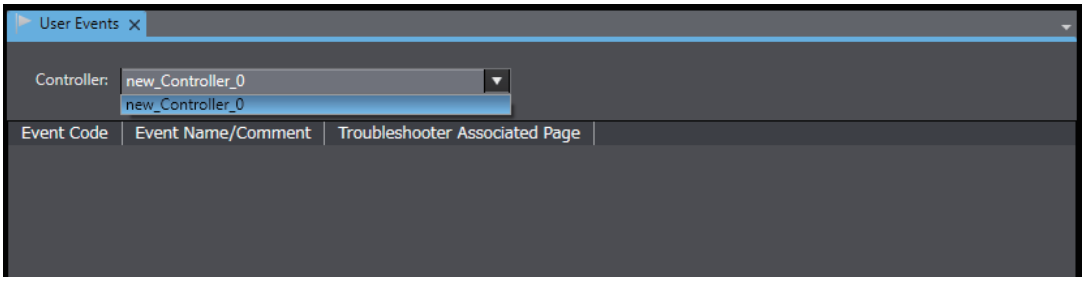


The User Event Editor is displayed.

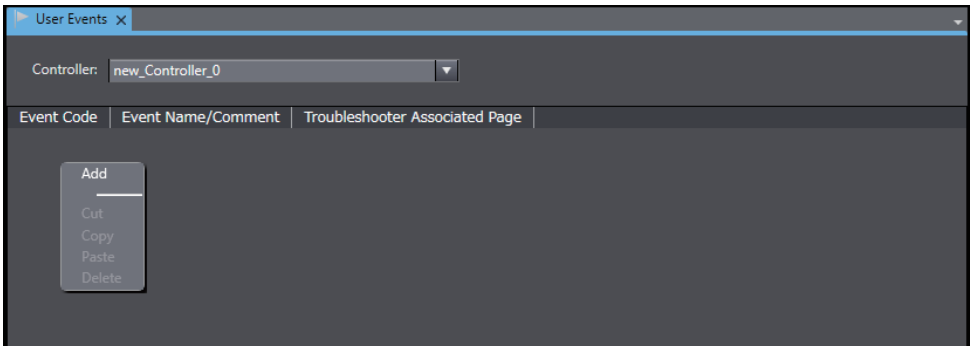


3 Select the controller for which to perform the settings.

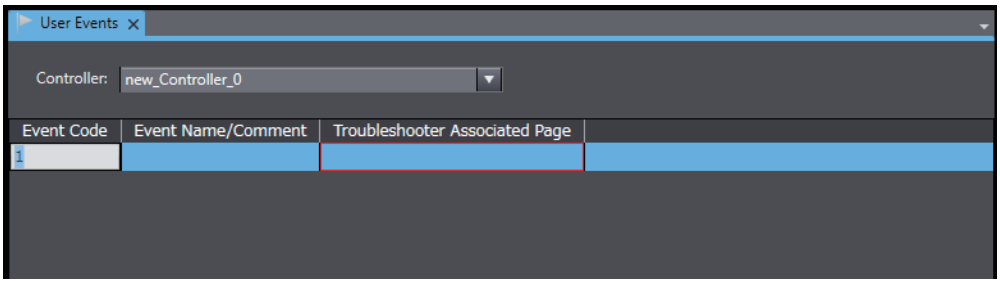
Select the target controller from the drop-down list at the top of the Edit Pane.



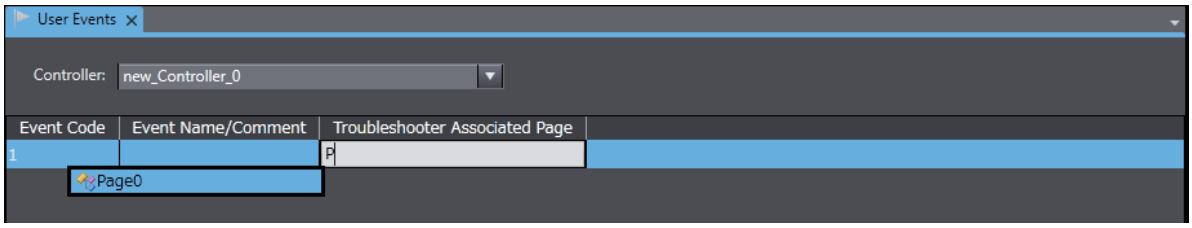
4 Right-click on the grid and select ***Add*** from the menu.



5 Specify the event codes for the user events.



6 Specify the user screens to use as the transition destinations.



Deleting, Copying, and Pasting User Events

You can delete, copy, and paste user events using the same procedures as those that you use for pages.

4-4-3 Registration for Data Logging

Data Logging

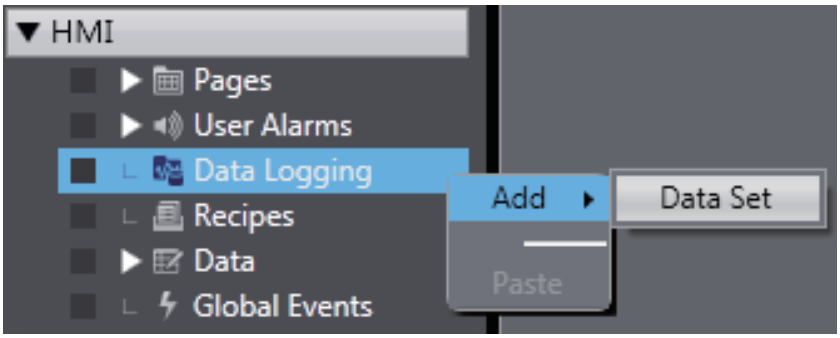
You can use data logging to store the changes in the values of specified variables over time.

You can display the saved data with Trend Graph objects. You can also save this data to external files.

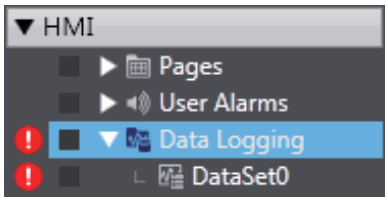
Registering a New Data Set

To log data, you must create one or more data sets. You can create different data sets to change the location where the data is saved.

1 Right-click **HMI - Data Logging** in the Multiview Explorer and select ***Add - Data Set*** from the menu.

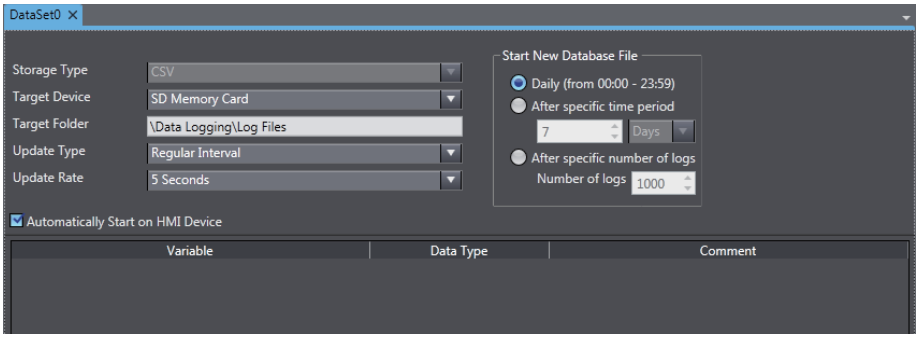


DataSet0 is added under **Data Logging.**



2 Double-click the new data set.

A tab page to make settings for the data set is displayed in the Edit Pane. You can set the variables to log, the storage locations, and other parameters.



Additional Information

You can copy the data set settings to Excel or another spreadsheet application to edit the settings.

Deleting, Copying, and Pasting Data Sets

You can delete, copy, and paste data sets using the same procedures as those that you use for pages.

4-4-4 Registering Data Groups

Data Groups

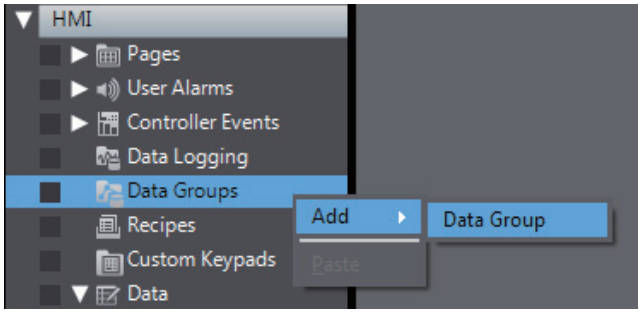
You can register the data groups that are displayed as broken-line graphs.

You can register multiple series of data as a data group. You can also register multiple variables or array variables in each data series.

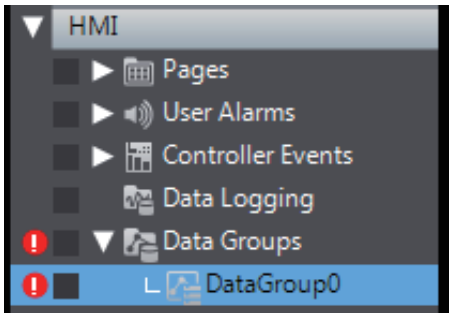
Registering a New Data Group

To display a broken-line graph, you must create a data group. A maximum of 32 groups can be registered. You can specify up to 20 series of data for a single group.

1 Right-click **HMI - Data Groups** in the Multiview Explorer and select ***Add - Data Groups*** form the menu.

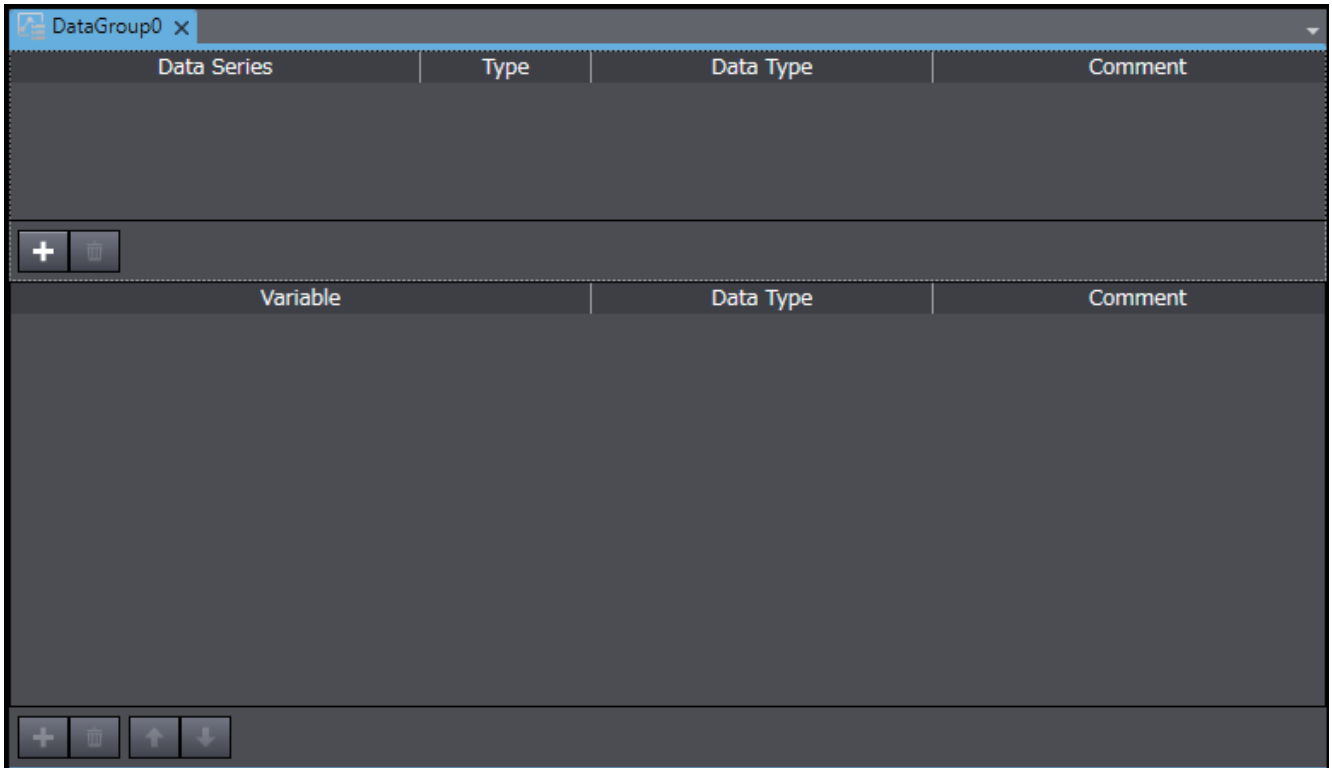


DataGroup0 is added under **Data Groups**.



2 Double-click the new data group.

A tab page to perform settings for the data group is displayed in the Edit Pane. You can specify variables to be displayed, as well ad other parameters.



Deleting, Copying, and Pasting Data Groups

You can delete, copy, and paste data groups using the same procedures as those that you use for pages.

4-4-5 Registering Recipes

Recipes

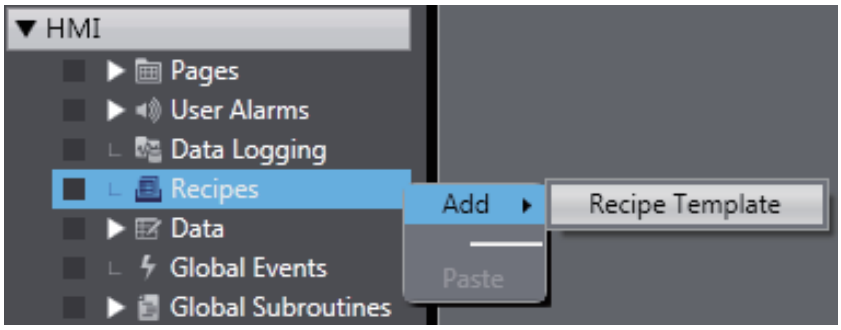
A recipe is used to write data (numeric data or text strings) that was set in advance in the HMI to all of the specified variables as a group or to read all of the specified variables as a group.

You can manipulate the registered recipe data with Recipe Viewer objects.

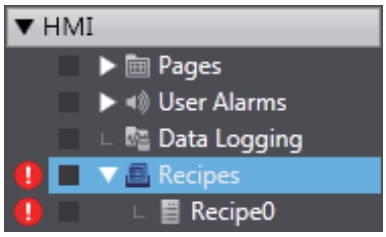
Registering a New Recipe

To use recipes you must create them.

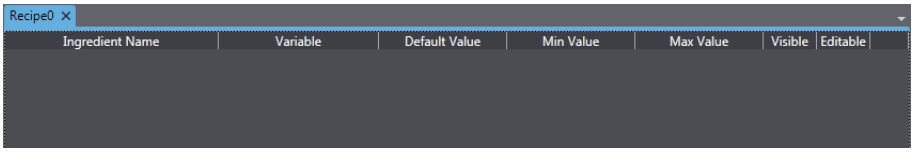
1 Right-click **HMI** - **Recipe** in the Multiview Explorer and select ***Add - Recipe Template*** from the menu.



Recipe0 is added under **Recipes**.



2 Double-click the new **Recipe0**. A tab page to make settings for Recipe0 is displayed in the Edit Pane. You can set the variables to use and other parameters.



Additional Information

- You can copy the recipe settings to Excel or another spreadsheet application to edit the settings.
- Restrictions relating to text that can be used in recipe names are as follows.

  Restrictions Common to All Runtime Versions
  - Names must begin with an alphabet character or an underscore (_).
  - Alphabet characters, numbers and underscores can be used.
  - When beginning a name with an underscore, at least one alphabet character or number must be used.
  - Names beginning with _HMI, _eHMI_, _sHMI_, _uHMI_, and _SyxSPC_ cannot be used.
  - Names that are reserved for Sysmac Studio or Visual Basic cannot be used.
  - Names must be within 127 characters.

  Restrictions in Runtime Version 1.16 or Earlier
  - Symbols except for underscores cannot be used.

  Restrictions in Runtime Version 1.17 or Later
  - Symbols except for comma (,) and double quotation marks (") can be used.

Deleting, Copying, and Pasting Recipes

You can delete, copy, and paste recipes using the same procedures as those that you use for pages.

4-4-6 Registering Custom Keypads

Custom Keypads

You can specify the colors, layouts, and other properties of buttons to create keypads.

You can manage custom keypads by creating groups in which to register multiple keypads including Boolean Keypad, Numeric Keypad, Hexadecimal Keypad, QWERTY Keypad, DateTime Keypad, and TimeSpan Keypad.

Additional Information

The keypad displayed with a data input object and EditVariable is dependent on the data type of the variable assigned to the corresponding location. The following table shows the data types and displayed keypads. If a custom keypad is specified, the custom keypad for the data type is displayed. If a custom keypad that corresponds to a data type does not exist, the system keypad is displayed.

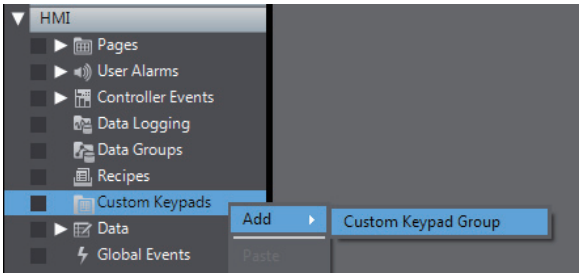
In addition, if scaling is set, the Numeric Keypad is displayed because the variable is handled as the Double type.

Data type name	Displayed keypad
Boolean	Boolean Keypad
Byte UInteger ULong UShort	Hexadecimal Keypad
Decimal Double Integer Long SByte Short Single String (when Numeric Keypad is specified)	Numeric Keypad
String (when QWERTY Keypad is specified)	QWERTY Keypad
Date	DateTime Keypad
TimeSpan	TimeSpan Keypad

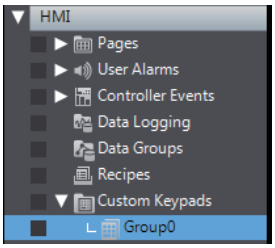
Registering a New Custom Keypad

To create a custom keypad, you must first create a custom keypad group. You can register up to 50 groups.

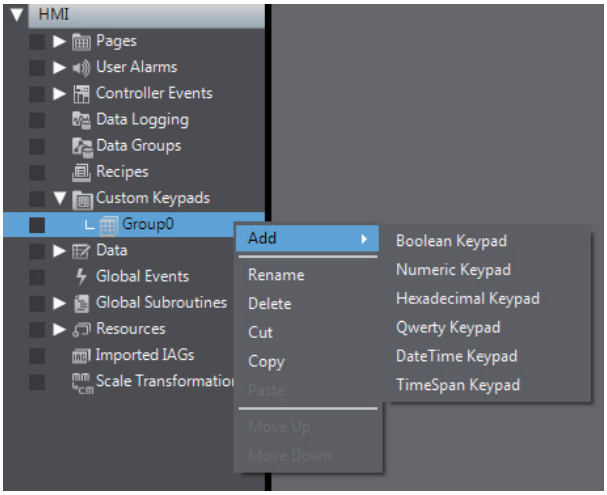
1 Right-click **HMI - Custom Keypads** in the Multiview Explorer and select ***Add - Custom Keypad Group*** from the menu.



"Group0" is added under **Custom Keypads**.

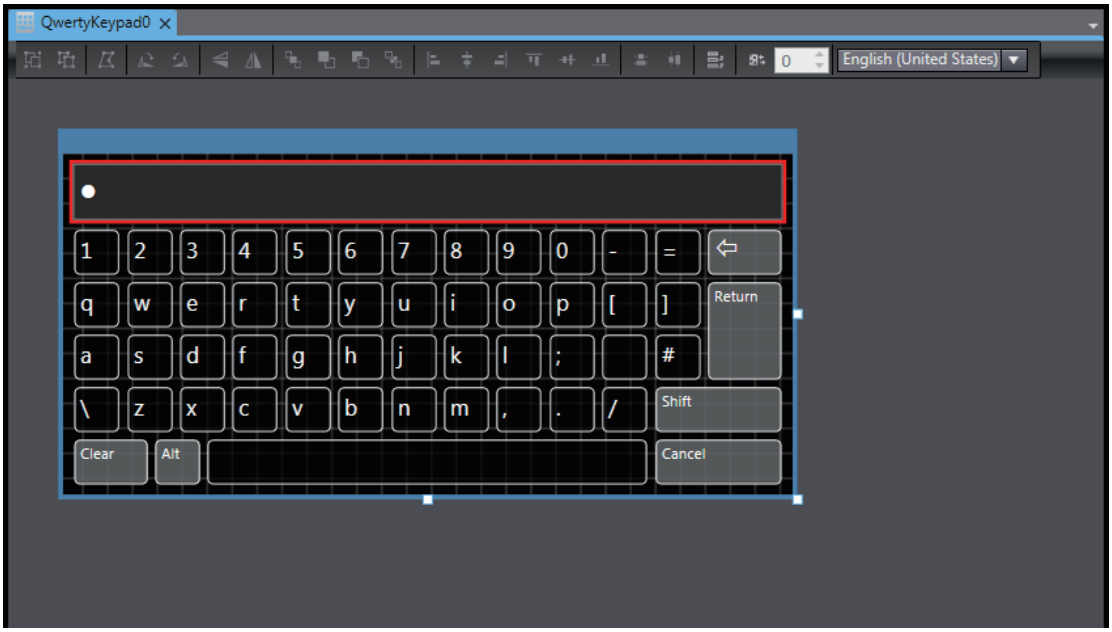


2 Right-click the new custom keypad group, and select ***Add*** and then select a custom keypad from the menu.



3 Double-click the new custom keypad.

A tab page to edit the custom keypad is displayed in the Edit Pane. You can set the layout, design, and other parameters for the objects.

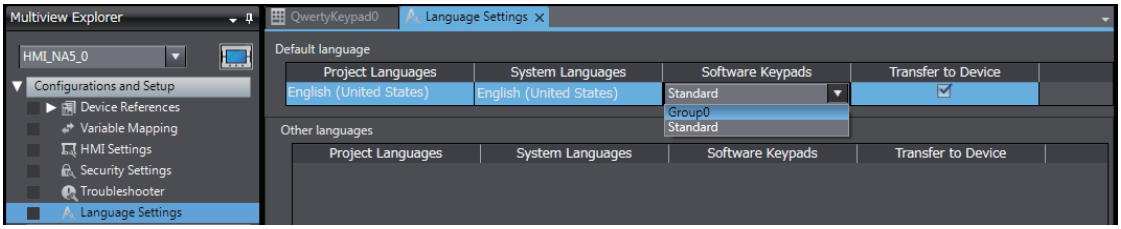


Applying a Custom Keypad

To use a custom keypad, you must specify the custom keypad group to use on the Language Settings Tab Page.

1 Double-click Language Settings under Configuration and Setup in the Multiview Explorer.

2 Select a custom keypad group from the drop-down list of the Software Keypads in the Edit Pane.



Deleting, Copying, and Pasting Custom Keypads or Custom Keypad Groups

You can delete, copy, and paste custom keypads or custom keypad groups using the same procedures as those that you use for pages.

4-4-7 Setting Global Events and Corresponding Actions

Global Events

Global events occur at the project level and do not belong to any specific page.

When a global event occurs, the action that is assigned to the event is executed.

● Global Events

Global events include function key operations, changes in the values of variables, starting the project, etc. A list of the events that you can set is provided in *A-1 Events and Actions* on page A-2.

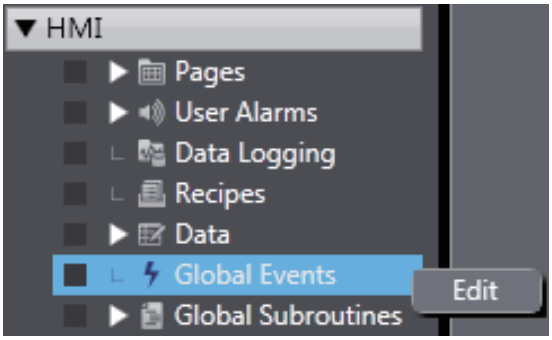
● Actions

You can specify the action to perform when a global event occurs from a list of predefined actions. Actions include executing global subroutines and other system-defined operations. A list of the actions that you can set is provided in *A-1 Events and Actions* on page A-2.

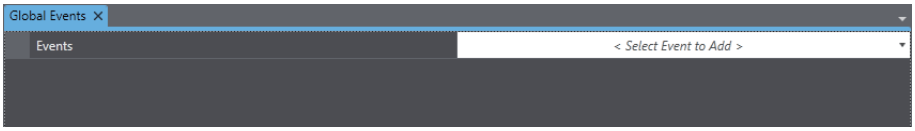
Setting Up Global Events

To set up a global event, you select the global event and then set the action to execute when the event occurs.

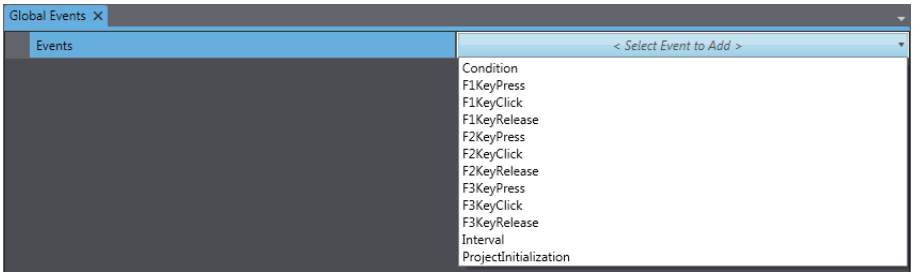
1 Right-click **HMI - Global Events** in the Multiview Explorer and select ***Edit*** from the menu. Or, double-click **HMI - Global Events**.



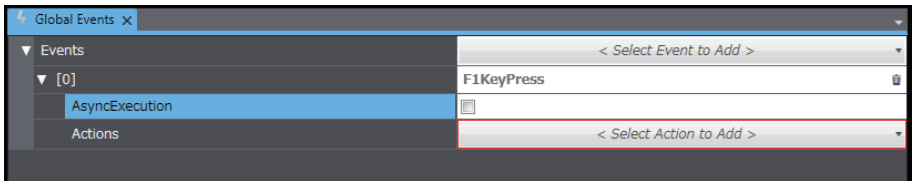
2 A tab page to make settings for the global event is displayed in the Edit Pane.



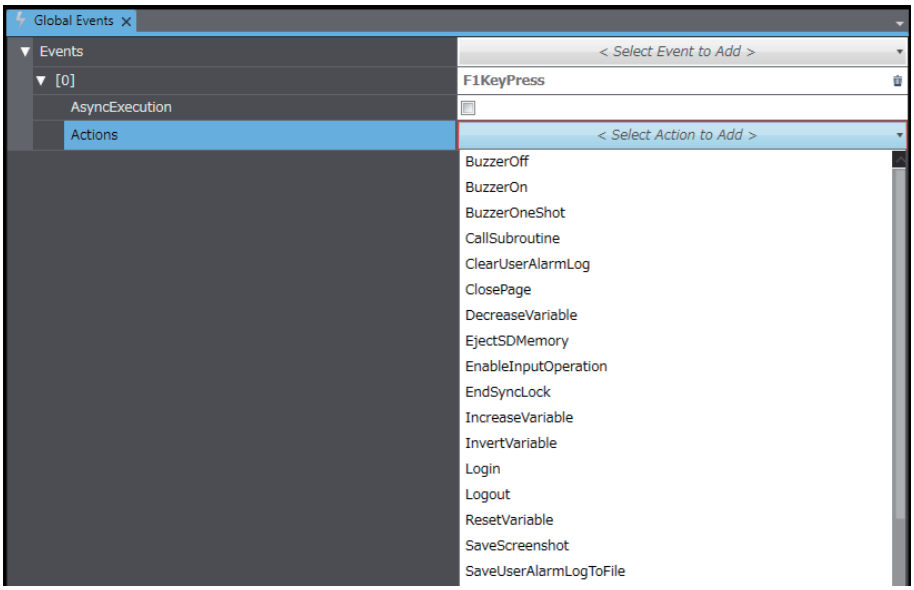
3 Click in the column on the right to select the event to set from the event list.



A global event is added.



4 In the **Actions** row below the new global event, click in the right column and select the action to perform for the event from the action list.



5 When you add global events, they are numbered serially from 0 in the order that you add them.

Deleting Global Events

To delete all of the settings for global events, right-click the **Events** header at the top and select ***Reset***.

To delete an individual event, click the Button on the right edge of the event.

4-4-8 Registering Global Subroutines

Global Subroutines

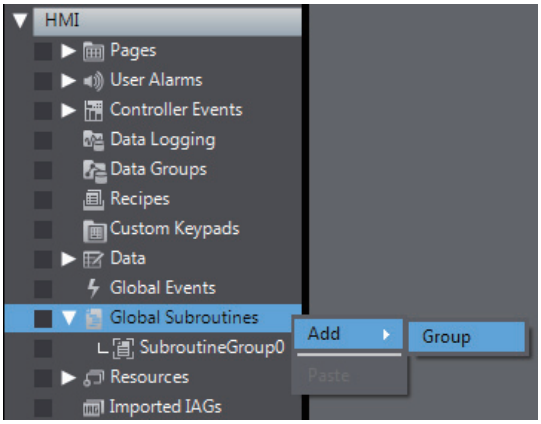
You can register global subroutines, which you can then call from anywhere in the project.

You can register common subroutines that do not rely on page conditions to make subroutines easier to maintain.

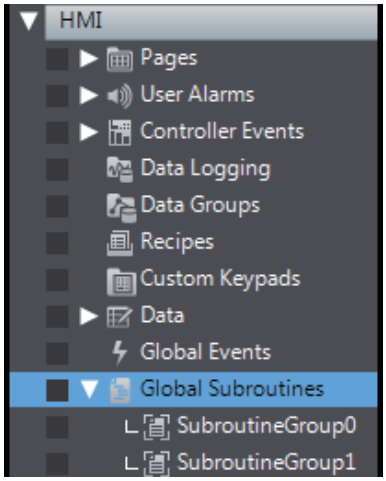
Registering a New Subroutine Group

To register global subroutines, you must create one or more subroutine groups. You can use subroutine groups to separate subroutines by purpose.

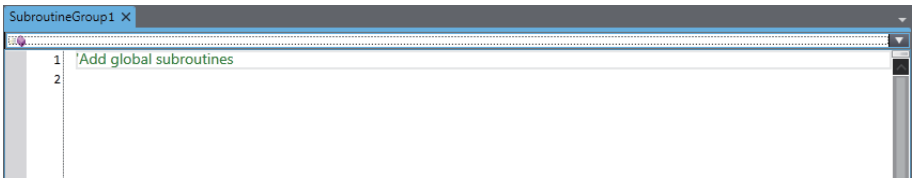
1 Right-click **HMI - Global Subroutines** in the Multiview Explorer and select ***Add -Group*** from the menu.



SubroutineGroup1 is added under **Subroutine Groups**.



2 Double-click the new subroutine group. A tab page for the Code Editor is displayed in the Edit Pane.



Deleting, Copying, and Pasting Subroutine Groups

You can delete, copy, and paste subroutine groups using the same procedures as those that you use for pages.

4-4-9 Setting Up Resources

Resources

Resources are the text strings, movies, still images, and documents that are displayed for objects and alarms on user pages.

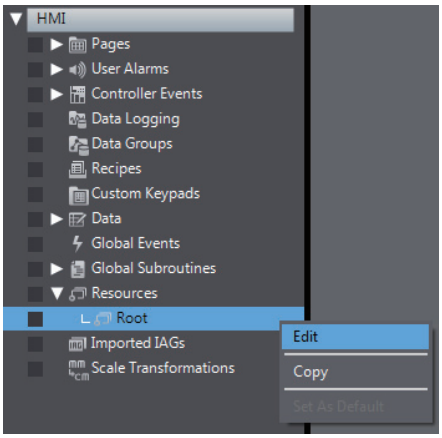
You can use **Resources** to manage all of the text strings, images, files, and other resources that you use in a project.

For multi-language projects, you can set resources for each project language.

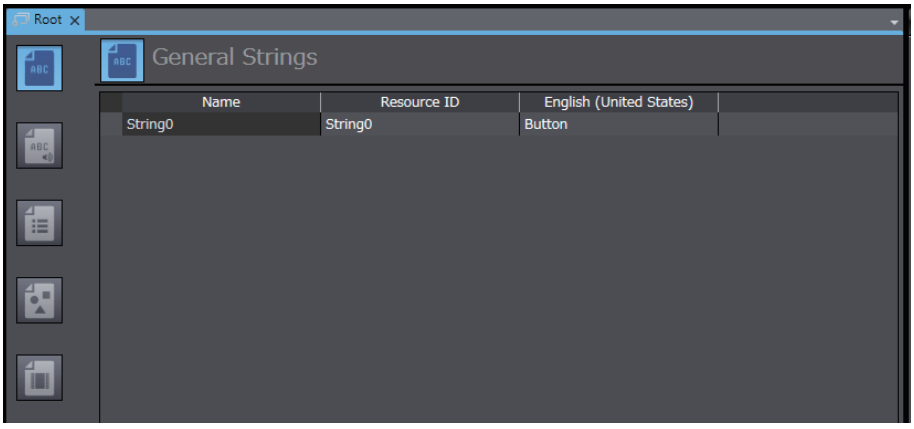
Setting Up Resources

To set up a resource, select the resource to set up and then make the settings.

1 Right-click **HMI - Resources - Root** in the Multiview Explorer and select ***Edit*** from the menu. Or, double-click **HMI - Resources - Root**.



2 A tab page to make settings for resources is displayed in the Edit Pane. You can select the resource to set and make the settings.



Additional Information

- You can import or export resource settings using the Excel workbook format. Refer to *11-5 Import/Export Resources* on page 11-13.
- To save the selected file, right-click on the grid and select **Save file** from the menu.

Deletion of Unused Resources

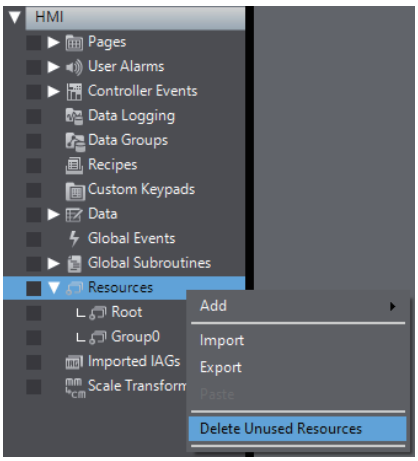
Resources that are not used can be deleted by using the following procedure. The resources to be deleted are general strings only.

Additional Information

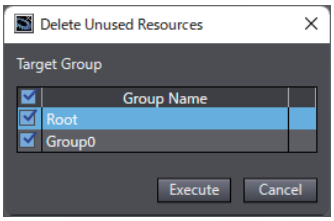
In deletion of unused resources, resources expressly assigned with a resource ID in the project are recognized as used resources. For this reason, resources for which the resource ID is known only at execution, such as dynamic links, are handled as unused resources. As a function for individually selecting deletion targets is not provided, make use of groups to avoid the deletion of resources that must not to be deleted.

Moreover, as unused resources cannot be restored once they have been deleted, backing up the project before executing a deletion is recommended.

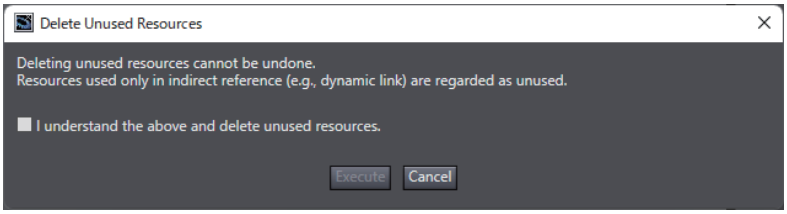
1 Right-click **HMI - Resources** in the Multiview Explorer and select **Delete Unused Resources** from the menu.



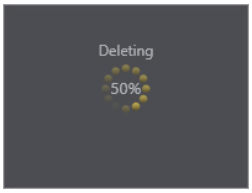
2 Select the check box of the group to execute deletion, and click **Execute**.



3 Check the text in the dialog box, then click the check box and click **Execute**.



4 When the dialog box below disappears, this indicates that deletion is completed.



4-4-10 Setting Up IAG Resources

IAG Resources

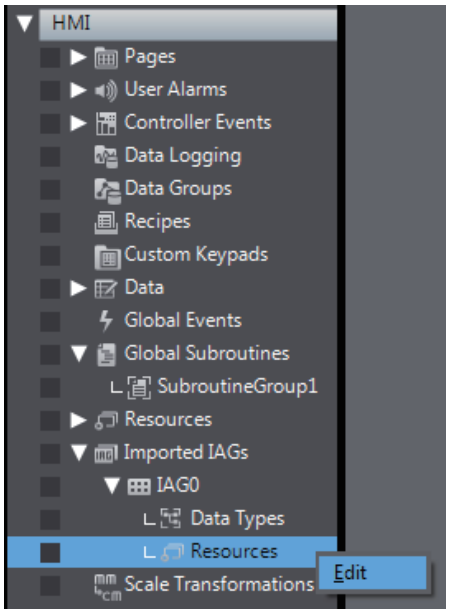
IAG resources are used within IAGs on user pages. When you place an IAG on a page, the resources for the IAG are automatically registered.

The languages that are displayed by the IAG resources are determined by the project languages. The resources that are set in advance for the IAGs are displayed for the IAG resources. If a language that is not included in an IAG is set as a project language, the resources will be blank by default.

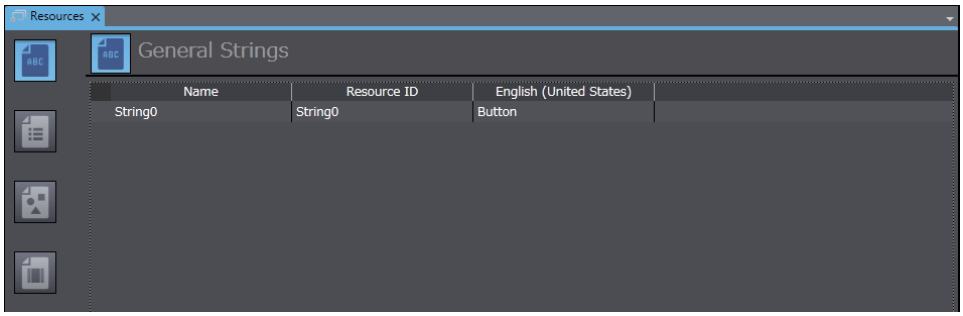
Setting Up Resources

To set up an IAG resource, select the IAG resource to set up and then make the settings.

1 Right-click the IAG resource to edit under **HMI - Imported IAGs** in the Multiview Explorer and select ***Edit*** from the menu. Or, double-click the resource.



2 A tab page to make settings for IAG resources is displayed in the Edit Pane. The procedures are the same as for other resources, but you can change only the contents of the resources that are in the IAGs.



Additional Information

- To delete all Imported IAGs not used in the project, right-click [Imported IAGs] and select [Delete not used IAG] from the menu.
- To save the selected file, right-click on the grid and select **Save file** from the menu.

4-4-11 Registering Scaling

Scaling

Scaling is used to transform displayed numeric values by specifying a multiplier and offset value.

To perform the settings for each object, open the object properties and set the parameters for scaling under **Behavior**.

You can perform the settings for individual variables on the Global Variables Table.

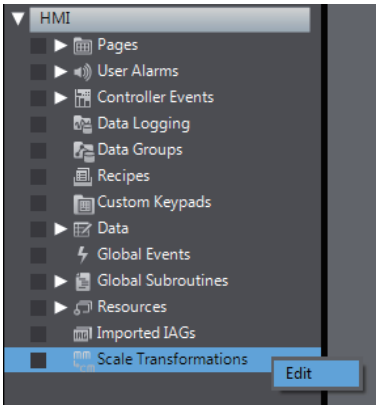
Example: In this example, a multiplier of 100 and an offset of 50 are specified for an integer variable (intSample):

Value of intSample	Calculation formula	Displayed value
0	50+0×100	50
1	50+1×100	150
2	50+2×100	250
100	50+100×100	10050

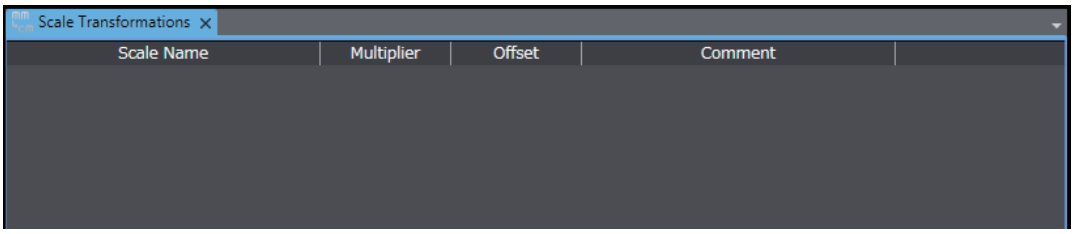
Registering a New Scaling

You can register a maximum of 1,000 scaling settings.

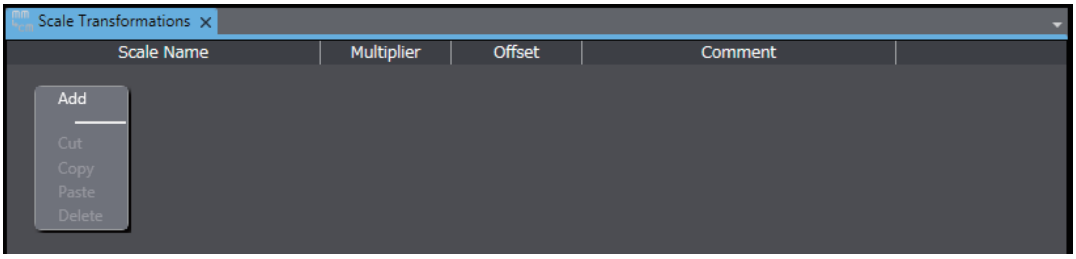
1 Right-click **HMI - Scale Transformations** in the Multiview Explorer, and select ***Edit*** from the menu. Or, double-click **HMI - Scale Transformations**.



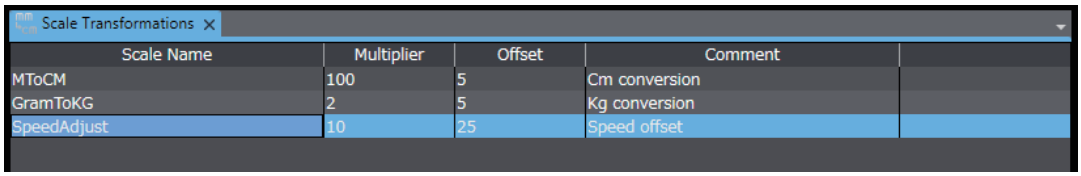
2 A tab page to perform settings for the scaling is displayed in the Edit Pane.



3 Right-click in the grid and select ***Add*** from the menu.



4 Specify the scale name, multiplier, offset, and other items.



Deleting, Copying, and Pasting Scalings

To delete all of the settings for scaling, select all the items and then right-click and select ***Delete*** from the menu.

To delete an individual scaling, click the button at the bottom left of the setting pane.

Precautions for Correct Use

When rounding values to the nearest integer, ".5" is rounded to the nearest even-number integer, as shown in the following examples.

Value before rounding	Value after rounding
1.2	1
1.5	2
1.8	2
2.5	2 (not 3)

4-5 Subroutines

Subroutines

Subroutines are Visual Basic programs that the user can create. You create subroutines under **Subroutines** in the HMI project.

Precautions for Correct Use

This manual describes only aspects that are different from the specifications standardized by Microsoft Corporation. For any specifications not given in this manual, refer to commercially available reference materials.

Subroutines

There are global subroutines and page subroutines, as described below.

Global Subroutines

Global subroutines are shared by the entire project.

You create global subroutines under **Global Subroutines** in the HMI project. You set CallSubroutine as the action in a global event, object event, or user alarms event to call a global subroutine.

You can also call a global subroutine from a page subroutine or from another global subroutine.

Page Subroutines

Page subroutines are specific to one page.

You create page subroutines with the Code Editor for a page in the HMI project.

You can directly create the subroutines to execute in the events for individual objects on pages.

4-5-1 Subroutine Execution

You can execute subroutines in the following ways.

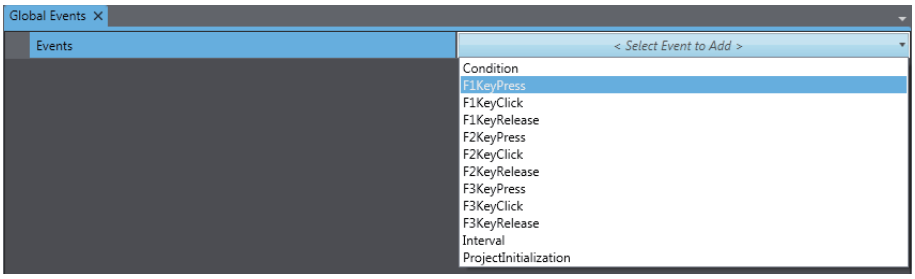
- Execution from Global Events
  You can execute a global subroutine from a global event.
- Execution from Page and Object Events
  You can execute a page subroutine or global subroutine from a page or object event.
- Execution from User Alarm Events
  You can execute a global subroutine from a user alarm event.

Execution from Global Events

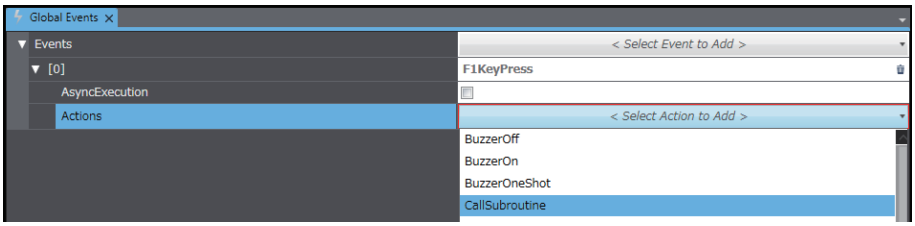
The following example shows how to use a global event to execute a global subroutine.

In this example, settings are made to execute the global subroutine when the F1 Key is pressed.

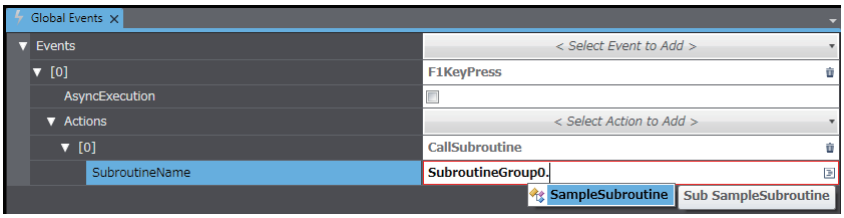
1 Display the global events and select *F1KeyPress* as the event.



2 Select *CallSubroutine* as the action.



3 For the subroutine name, specify the name of a previously created subroutine in the following format: *Subroutine_group.Subroutine_name*

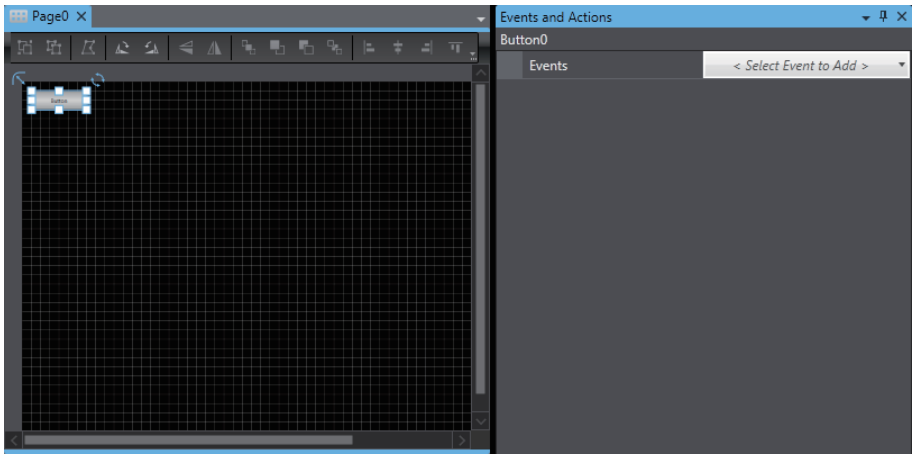


Execution from Objects

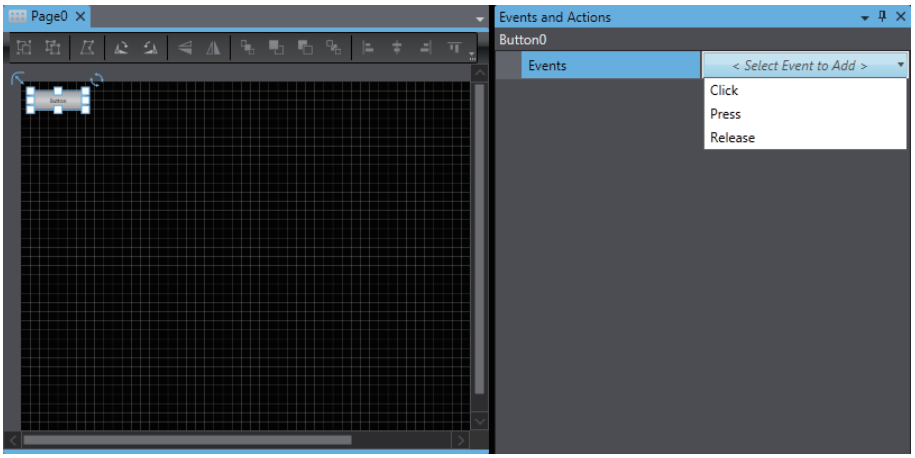
The following example shows how to use an object event to execute a subroutine.

In this example, settings are made to execute the subroutine when a Button object is pressed.

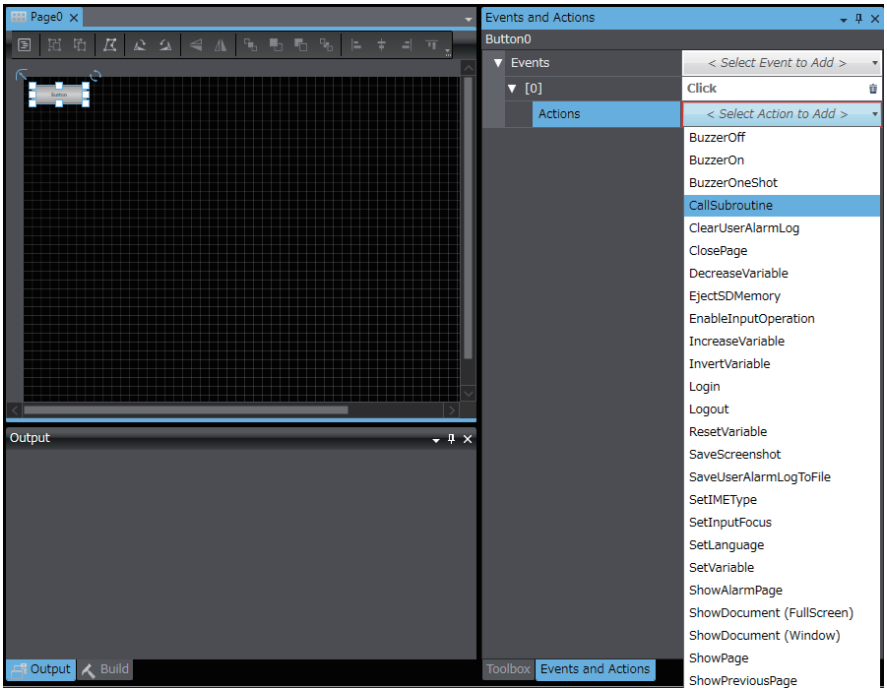
1 Select the Button object and display the events and actions.



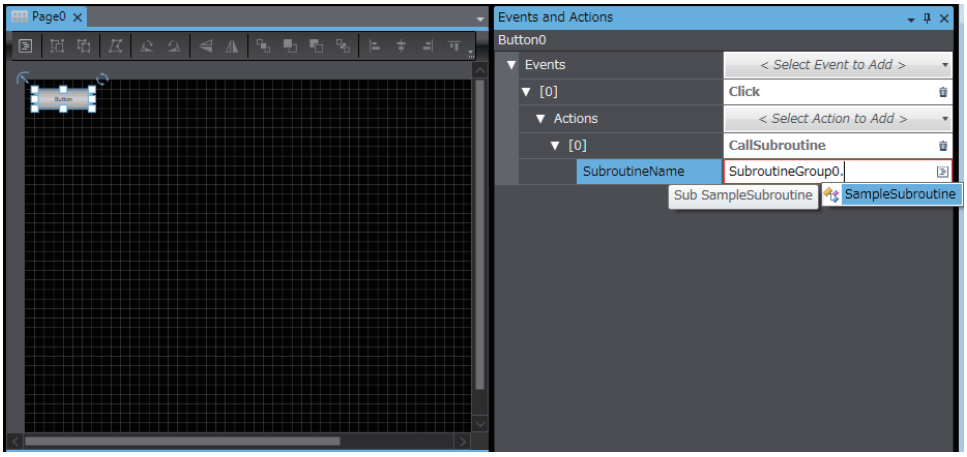
2 Select *Click* as the event.



3 Select *CallSubroutine* as the action.



4 Specify the subroutine to execute in the following format: *Subroutine_group.Subroutine_name*

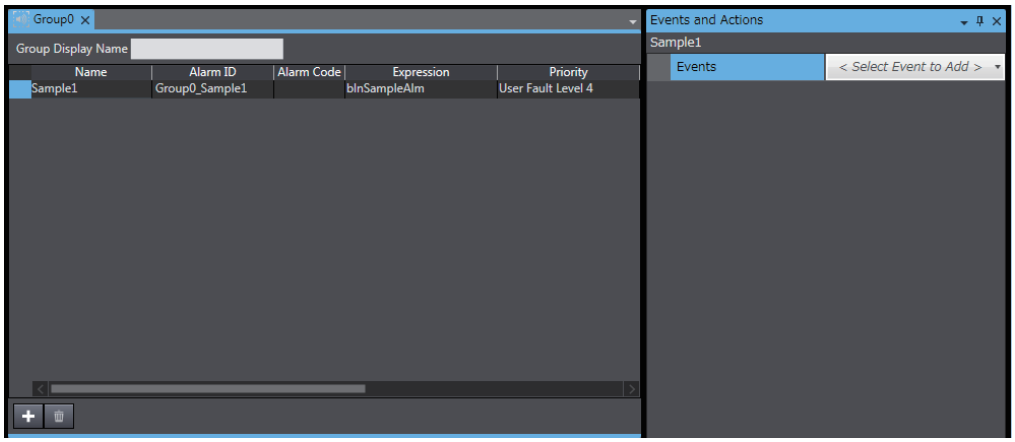


Execution from User Alarms

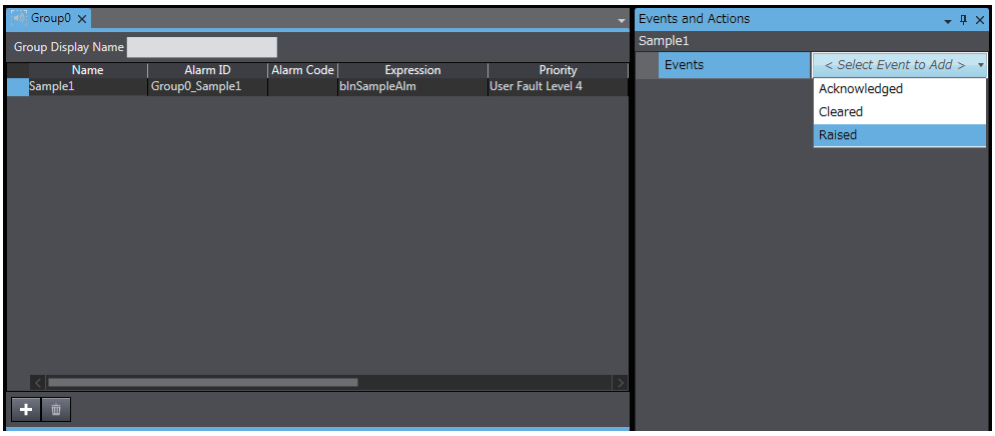
The following example shows how to use a user alarm event to execute a subroutine.

In this example, settings are made to execute the subroutine when the user alarm occurs.

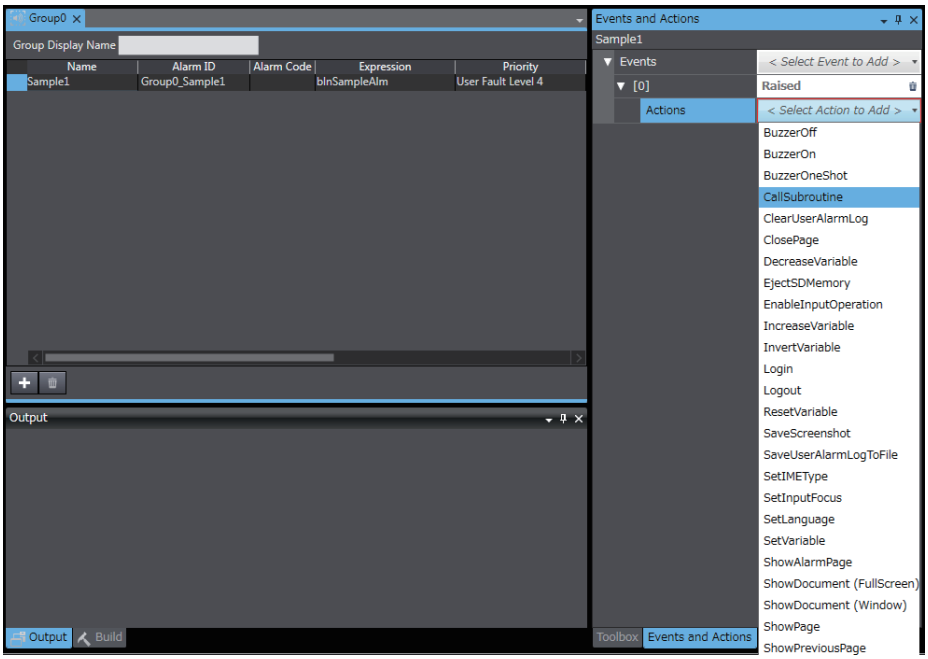
1 Select the user alarm and display the events and actions.



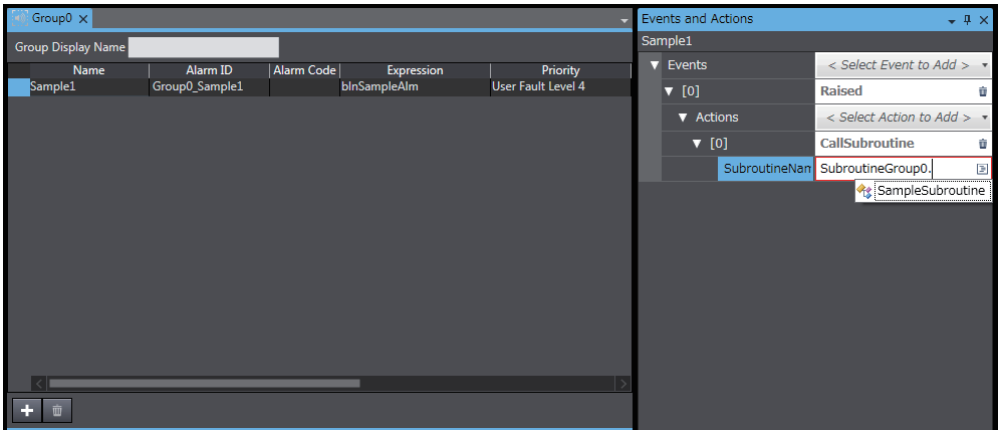
2 Select *Raised* as the event.



3 Select *CallSubroutine* as the action.



4 Specify the subroutine to execute in the following format: *Subroutine_group.Subroutine_name*



4-5-2 Precautions on Internal Processing

● Handling of Variables

If the value of an external variable is changed in a subroutine, the change is immediately updated at the connected device. Therefore, if you frequently change the value of an external variable inside a subroutine, the performance of that subroutine will be reduced.

● Processing during Subroutine Execution

The touch panel and function keys will not respond during execution of a page subroutine. If you execute processing that requires time, the HMI will not perform other operations until the processing is completed. Consider the execution time when you create subroutines. However, processing will continue for background operations, such as communications, and for page refreshing.

● Simultaneous Execution of Subroutines

It is possible that a subroutine for a global event and a page subroutine will be executed simultaneously. If both subroutines manipulate the same variable, implement exclusive control or other suitable measures.

Background Processing of Subroutines for Global Events

If you select the *AsyncExecution* Check Box, a subroutine process and a drawing update process for that global event are asynchronously executed. Up to five subroutines can be executed simultaneously. If you attempt to execute more than that, the processes enter standby for execution. If the number of subroutines waiting for execution exceeds 100, a warning dialog box appears on the screen.

When you asynchronously execute subroutines, the execution timing is controlled by the OS. Therefore, if you do not synchronize subroutines, you may get unexpected results.

Start actual system application only after sufficiently checking the timing of the execution of the operations of the programs.

4-5-3 Code Editor

Subroutines are edited with the Code Editor.

Starting the Code Editor

- **Global Subroutines**

  Double-click a previously registered subroutine under **HMI - Global Subroutines** in the Multiview Explorer. Or, right-click the subroutine and select ***Edit*** from the menu.

- **Page Subroutines**

  Right-click a previously registered page name under ***HMI - Pages*** in the Multiview Explorer and select ***View Code Editor*** from the menu.

Code Editor Features

The Code Editor provides functions equivalent to those of a standard text editor. It also provides functions that are optimized for Visual Basic, such as keyword highlighting, entry assistance, and collapsing Sub statements.

4-5-4 Differences in Language Specifications

Although subroutines are used in Visual Basic, some of the functions are restricted for HMIs. There are also extensions that are provided for use with HMIs. Refer to the *NA-series Subroutine Reference* for details.

4-6 Search and Replace

You can search and replace strings in the data of an HMI project. The basic Sysmac Studio operations for searching and replacing generally apply to HMIs. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for details.

Differences When an HMI Is Selected

The following differences apply when an HMI is selected.

- You can select only *All* for the *Look at* Box.

Precautions for Correct Use

You can use an action name as a search string, but you cannot replace it with another action name.

Example: You can search for the objects that specify the Login action. However, you cannot replace the Login action settings with the Logout action settings using the replacement function.

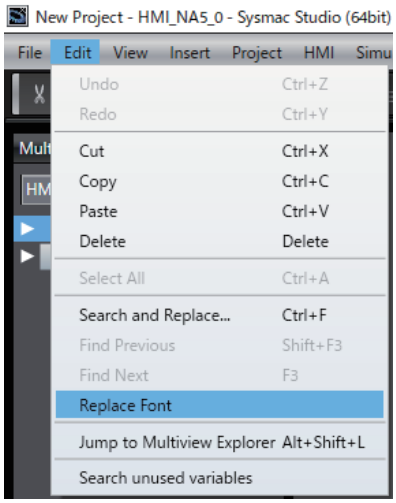
4-7 Font Replacement

Replaces the font settings set for objects in the device with the specified conditions.

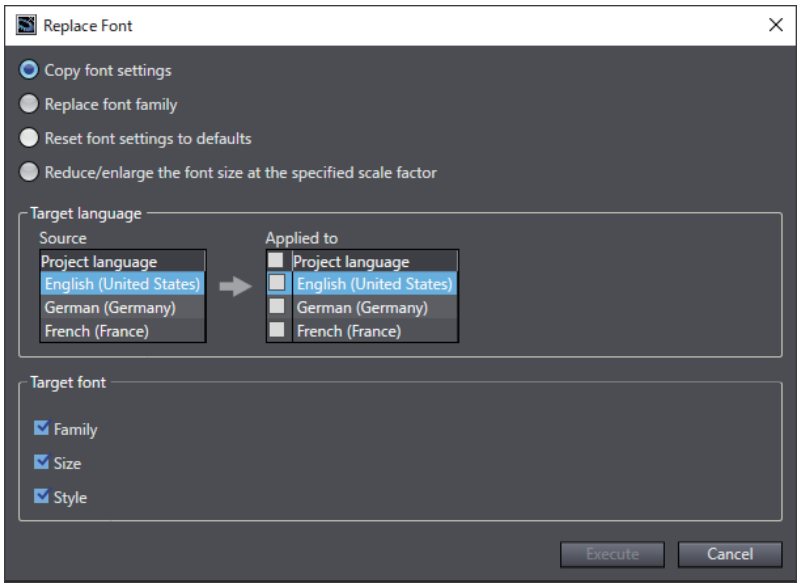
Font Replacement Procedure

Use the following procedure to replace the fonts.

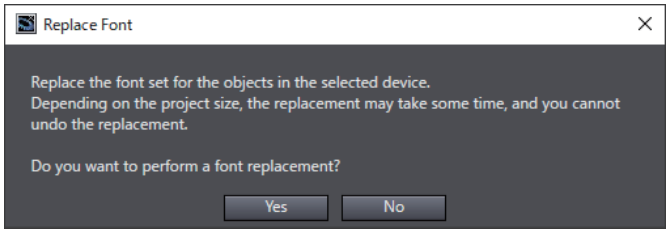
1 From the main menu, select **Replace Font** under the **Edit**.



2 Select the replacement condition and click **Execute**.



3 Click **Yes**.

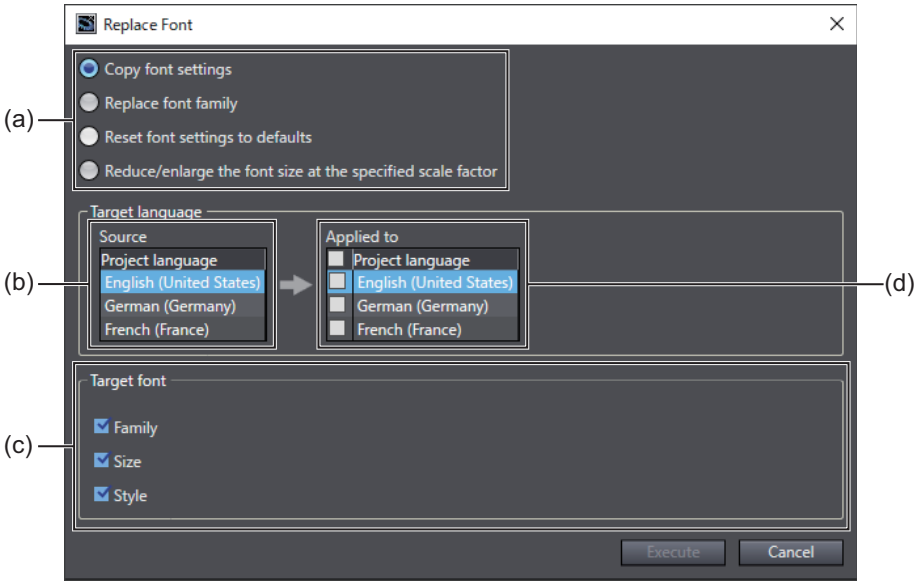


Additional Information

In the case of a project in which a large number of objects are arranged, if this function is executed immediately after opening the project or immediately after a large number of changes are made, it may take a very long time for the function to be executed. Run after confirming from the Task Manager that the CPU utilization rate of Sysmac Studio is low.

Copying the Font Settings

Copy the font settings for the language specified in the **Source** to the font settings for the language specified in the **Applied to**.



Symbol	Item	Description
(a)	Replacement method	Select the replacement method.
(b)	Source	Set the project language to be copied from.
(c)	Target font	Select the setting item of the font to be replaced.
(d)	Applied to	Select the project language to be replaced.

Example: The result of performing font replacement under the conditions below is as follows.

Project languages: English (United States), German (Germany), French (France)

Source: **English (United States)**

Applied to: only **German (Germany)** selected

Target font: all selected

● Before execution

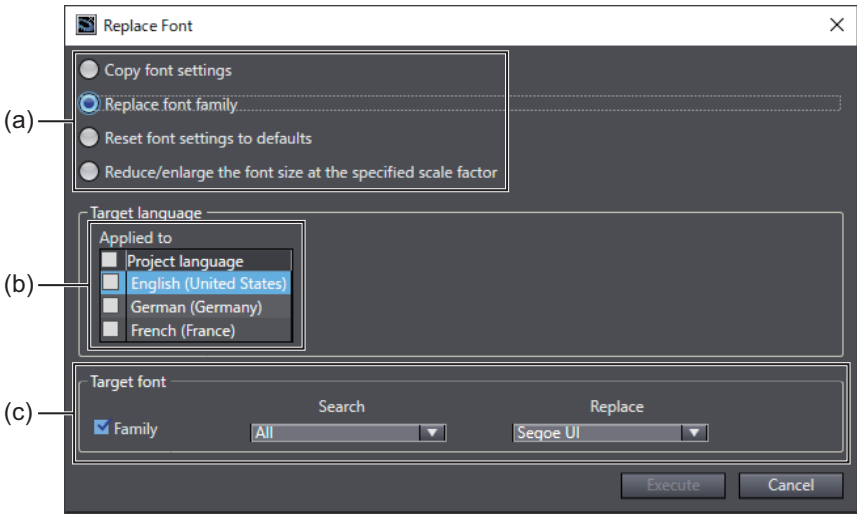
Project Languages	Family	Size	Style	View
English (United States)	Tahoma	20	Bold	English
German (Germany)	Segoe UI	12	Normal	Englisch
French (France)	Segoe UI	12	Normal	Anglais

● After execution

Project Languages	Family	Size	Style	View
English (United States)	Tahoma	20	Bold	English
German (Germany)	Tahoma	20	Bold	Englisch
French (France)	Segoe UI	12	Normal	Anglais

Replacing the Font Family

Replace the font family with the specified font family.



Symbol	Item	Description
(a)	Replacement method	Select the replacement method.
(b)	Applied to	Select the project language to be replaced.
(c)	Target font	Use **Search** to specify the font family to be replaced, and **Replace** to specify the font family to replace it with.

Example: The result of performing font replacement under the conditions below is as follows.

Project languages: English (United States), German (Germany), French (France)

Applied to: only **German (Germany)** selected

Target font: selected. Search set to **All** and Replace set to **Arial**.

● Before execution

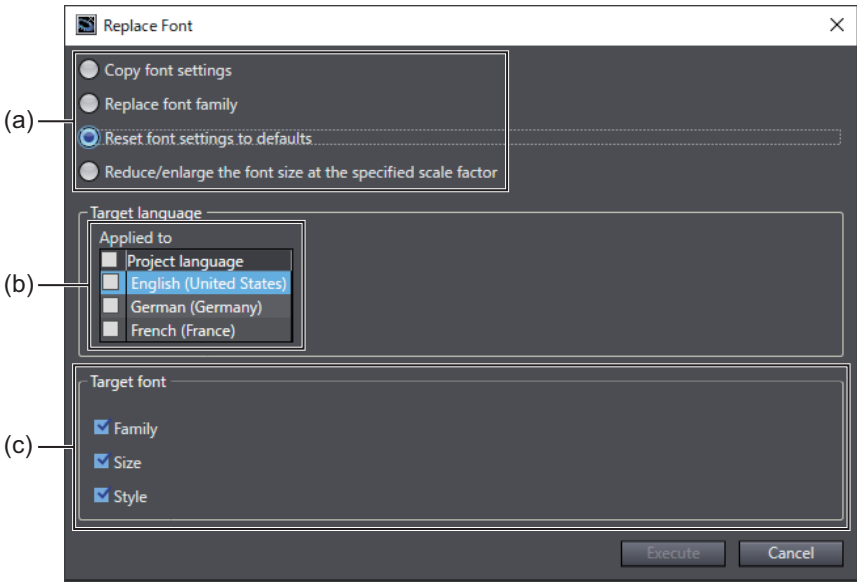
Project Languages	Family	Size	Style	View
English (United States)	Tahoma	12	Normal	English
German (Germany)	Times New Roman	20	Bold	Englisch
French (France)	Courier New	8	Italic	Anglais

● After execution

Project Languages	Family	Size	Style	View
English (United States)	Tahoma	12	Normal	English
German (Germany)	Arial	20	Bold	Englisch
French (France)	Courier New	8	Italic	Anglais

Resetting the Font Settings to the Default Settings

Change the font settings set for all objects to the font settings set in the language settings.



Symbol	Item	Description
(a)	Replacement method	Select the replacement method.
(b)	Applied to	Select the project language to be replaced.
(c)	Target font	Select the setting item of the font to be replaced.

Example: The result of performing font replacement under the conditions below is as follows.

Project languages: English (United States), German (Germany), French (France)

Applied to: select all languages

Target font: all selected

● Language settings

Project Languages	Family	Size	Style
English (United States)	Segoe UI	12	Normal
German (Germany)	Arial	12	Normal
French (France)	Arial	12	Normal

● Before execution

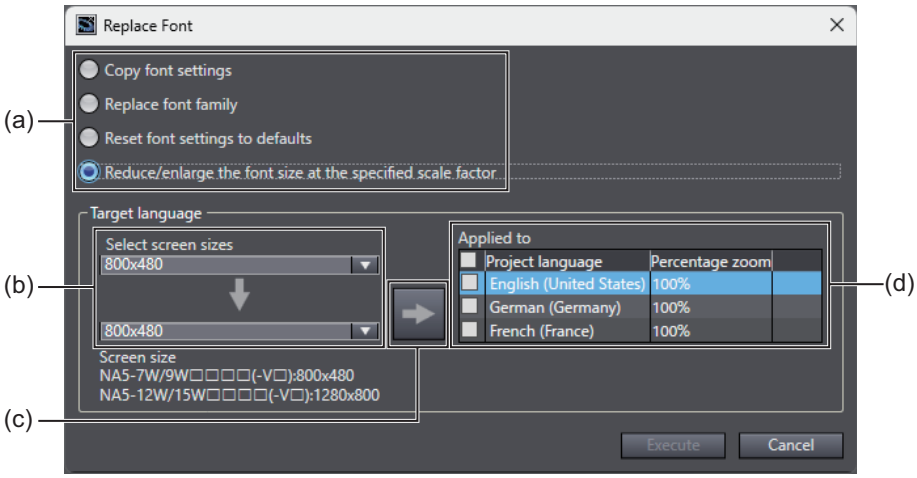
Project Languages	Family	Size	Style	View
English (United States)	Tahoma	12	Normal	English
German (Germany)	Times New Roman	20	Bold	Englisch
French (France)	Courier New	8	Italic	Anglais

● After execution

Project Languages	Family	Size	Style	View
English (United States)	Segoe UI	12	Normal	English
German (Germany)	Arial	12	Normal	Englisch
French (France)	Arial	12	Normal	Anglais

Scaling the Font Size to the Specified Magnification

Increase or decrease the font size set for all objects to the specified magnification.



Symbol	Item	Description
(a)	Replacement method	Select the replacement method.
(b)	Select screen sizes	Specify the screen resolution to specify the magnification according to the NA Series screen size.
(c)		Set the magnification value according to the details specified in **Select screen sizes**.
(d)	Applied to	Specify the project language and magnification to be replaced.

Example: The result of performing font replacement under the conditions below is as follows.

Project languages: English (United States), German (Germany), French (France)

Applied to: select all languages, specify 60% magnification

● Before replacement

Project Languages	Family	Size	Style	View
English (United States)	Tahoma	20	Normal	English
German (Germany)	Times New Roman	18	Normal	Englisch
French (France)	Courier New	22	Normal	Anglais

● After replacement

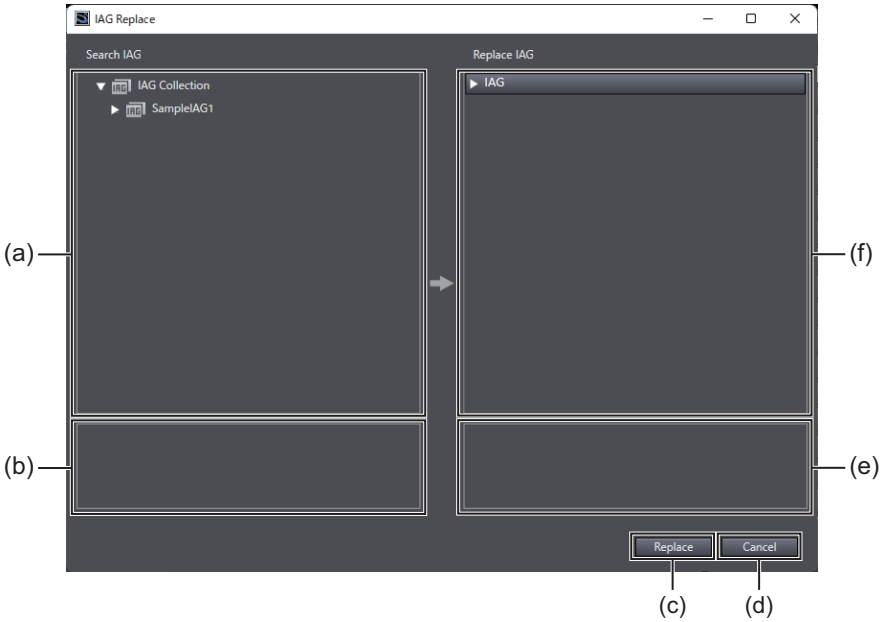
Project Languages	Family	Size	Style	View
English (United States)	Tahoma	12	Normal	English
German (Germany)	Times New Roman	10	Normal	Englisch
French (France)	Courier New	13	Normal	Anglais

4-8 IAG Replacement

IAG objects placed in a page in the currently selected device are replaced with other IAGs.

IAG Replace Dialog Box

In the IAG Replace dialog box, select the IAG to replace and execute replace.



Symbol	Item	Description
(a)	Search IAG	Select the IAG to be replaced. All IAGs in the device are displayed.
(b)	Contents of IAG to search	The contents of the IAG currently selected at **Search IAG** are displayed.
(c)	Replace	Executes the replace.
(d)	Cancel	Closes the dialog box.
(e)	Contents of IAG to replace	The contents of the IAG currently selected at **Replace IAG** are displayed.
(f)	Replace IAG	Select the IAG after replacement. The displayed IAG is the same as the one displayed in the Toolbox.

Targeted Objects

With IAG replacement, IAG objects in the currently selected device are targeted for replacement. In addition, properties belonging to the IAG objects are replaced as follows.

- Properties of IAG objects

  Values are transferred to the same properties. For this reason, when a property name that can be changed by a user belonging to **Behavior(Input)**, etc. is changed, values are not transferred. Configure the settings again after the replacement is complete.
- Animations of IAG objects

  The same animation is added to the IAG objects after replacement.
- Events and actions of IAG objects

  When the same events exist in the IAG objects after replacement, the same events and actions are added.
- Properties of objects in IAG objects

  The default values of the IAG after replacement are set. Set the values again as required.

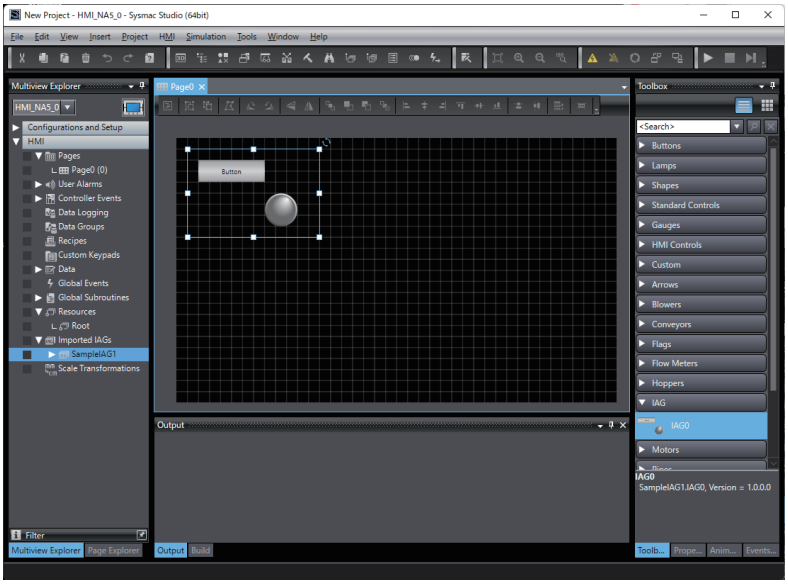
Procedure for Replacing IAGs

The following procedure describes an example of replacing IAG0 with IAG1.

1 Create a new IAG project, and create an IAG collection containing IAG0.

2 Create a new standard project, and display the IAG collection created in step 1 in the Toolbox.

3 Place IAG0 on the page.

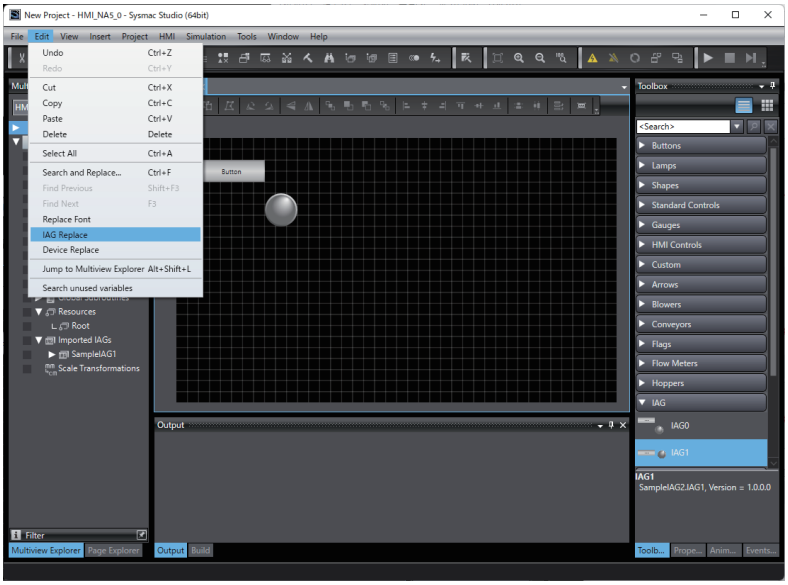


4 Save and close the project.

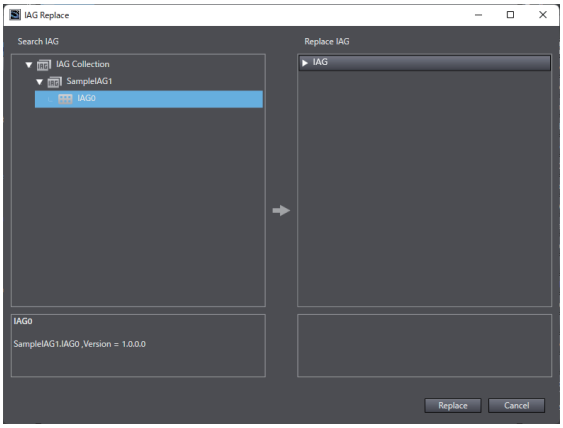
5 Open the project created in step 1, add changes to IAG0, and rename it as IAG1. Then, create the IAG collection.

6 Open the project created in step 2, and display the IAG collection created in step 5 in the Toolbox.

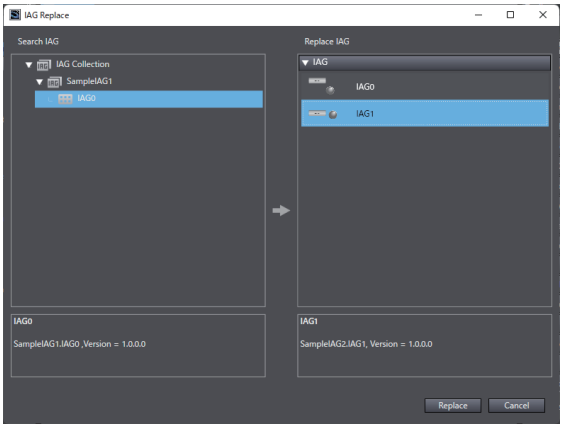
7 From the main menu, select **Edit - IAG Replace**.



8 From **Search IAG**, select IAG0 of the IAG to be replaced.

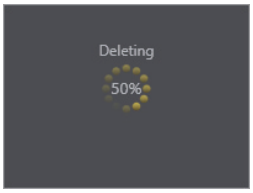


9 From **Replace IAG**, select IAG1 of the IAG after replacement.



10 Click **Replace**.

11 When the dialog box below disappears, this indicates that replacement is completed. In the IAG Replace dialog box, click **Cancel**.
The IAG object that has been replaced is displayed in the Search and Replace Results Tab Page. Use this page for confirmation, etc.

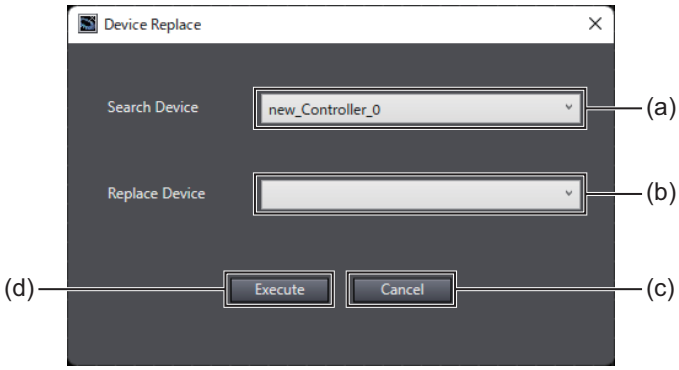


4-9 Device Replacement

Assignments such as the mapping information onto the devices of variables in the currently selected device are replaced.

Device Replace Dialog Box

In the Device Replace dialog box, select the device to replace and execute replace.



Symbol	Item	Description
(a)	Search Device	Select the device to be replaced. All internal and external devices in the current device are displayed.
(b)	Replace Device	Select the device after replacement. All internal and external devices in the current device are displayed. When **Create a new external device** is selected, a new external device is created, and device variables, etc. are registered to that device.
(c)	Cancel	Closes the dialog box.
(d)	Execute	Executes the replace.

Targeted Objects

By device replacement, the following are replaced. Even in the case of disagreement, for example, when multiple global variables are assigned to the same device variable in the device after replacement, replacement is executed and this results in an error. When an error occurs, correct the settings as appropriate.

The following are targeted for replacement.

- Data type and assignment destination of global variables

  The name of user-defined type devices set to the data type of global variables and the names of devices set to assignment destinations are replaced.
- User events

  All user events registered to devices to be replaced are moved to devices specified at Replace Device.
- Action

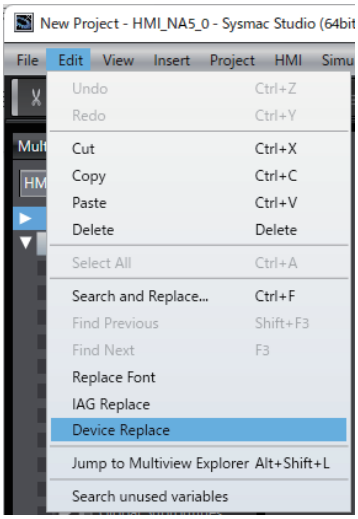
  Device names included in the settings of the following actions are replaced:
  - **Controller Name** of **ShowTroubleshooter**
  - **Look in** of **StartSafetyMonitor**

- Subroutines
  The following function arguments are replaced:
  - RetrieveRange of StartSafetyMonitor(VariableName, RetrieveRange, RetrievePurpose)
  - ControllerName of ShowTroubleshooter(ControllerName, StartPage)
- Troubleshooter
  The following settings are replaced:
  - Device of **Launch on System Event** and **Launch on User Event**
  - Device of Language mapping

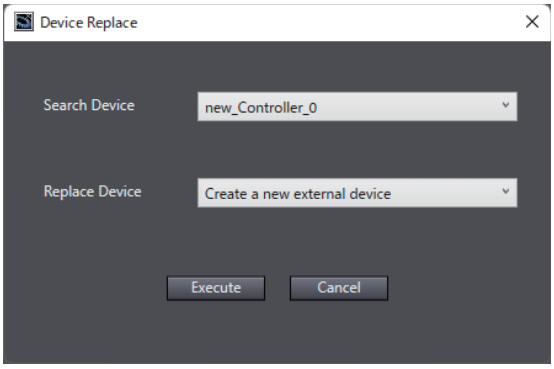
Procedure for Replacing Devices

The following procedure describes an example of replacing information assigned to new_Controller_0 by creating a new external device in a project containing the new_Controller_0 and the HMI of an internal device.

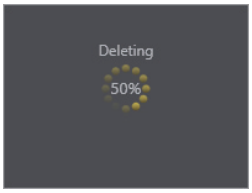
1 From the main menu, select **Edit - Device Replace**.



2 At **Search Device**, select **new_Controller_0** and at **Replace Device**, select **Create a new external device**, and click **Execute**.



3 When the dialog box below disappears, this indicates that replacement is completed. In the Device Replace dialog box, click **Cancel**.
The location that has been replaced is displayed in the Search and Replace Results Tab Page. Use this page for confirmation, etc.



4-10 Cross References

4-10-1 Cross References

Cross references allow you to confirm in a list format where an element[*1] that is part of a project is used within the project.

On the list that shows the locations where an element is used, you can access the locations where the element is used.

*1. You can cross-reference the following elements.

- Variables
- Data types
- Pages
- Resources
- Data groups
- Data sets
- Recipes

The basic operating procedures are the same as for the rest of the Sysmac Studio. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for details.

Additional Information

When a variable in the properties of an object or a variable set in an expression is selected, the corresponding row in the Cross Reference window is highlighted. Only Variable in the properties and variables set in Expression in the properties are compatible with this function.
Note that even if the same variable set in other than the properties is selected while a row is highlighted, the highlighted row on the Cross Reference window will remain the same as in the previous selection.

Displaying and Manipulating Cross References

1 From the main menu, select ***View - Cross Reference Tab Page***.

The Cross Reference Tab Page is displayed. You can check a list of the locations where the selected element is used.

4-11 Building

4-11-1 Building

The project must be built to convert it into a form that the HMI can execute.

During the building process, subroutines and variables are checked.

If there are any errors, the build operation is not performed and is displayed next to the program or variable where the error occurred in the Multiview Explorer.

You can confirm the errors on the Build Tab Page.

Additional Information

In a project using Runtime Ver. 1.10 or later, detailed error checks of the expressions are made during the building process and the results are displayed in the Build Tab Page.

The detailed error checks of the expressions are not displayed unless compilation is performed. Therefore, these error checks will not be performed if errors remain that are checked in real time. Also, as compilation is run on a module-by-module basis and ends when an error is found, all the errors may not be displayed with a single build. Repeat building and error correction until no errors remain.

Note that expressions whose type is only fixed during execution will not cause an error during building. For example, if a String type is assigned in a position where a Boolean type is required, some values assigned to the String type variable can be cast to a Boolean type so that no error occurs during building.

If a single build has 100 or more errors, some errors will not be displayed.

Some of error messages displayed here will be displayed in a language selected in Windows' language settings.

4-11-2 Build Operation

Use the main menu to execute the build operation. HMI projects are not built automatically even if no operations are performed for 5 seconds.

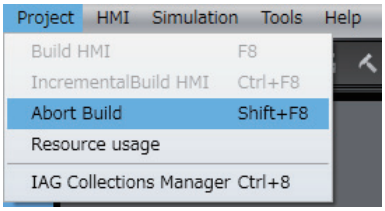
Executing a Build Operation

1 Select ***Build HMI*** from the Project Menu.

Aborting a Build Operation

No functions can be executed while building the project is in progress. If necessary, you can abort building to check the project.

1 Select ***Abort Build*** from the Project Menu.



2 The build operation is aborted and a message that says it was aborted is displayed in the Output Tab Page.

Additional Information

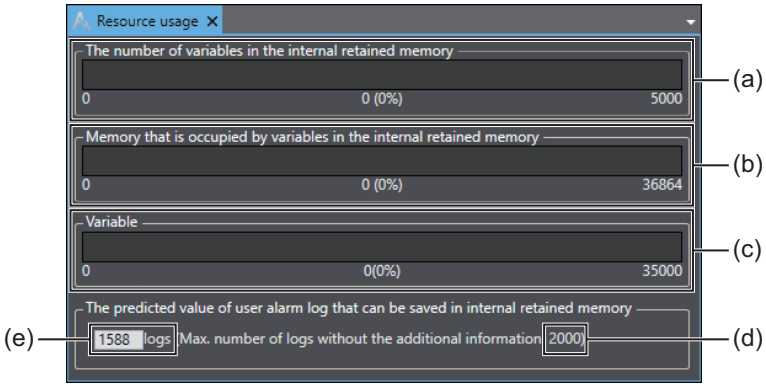
You can execute IncrementalBuild HMI to build only parts that were changed since the last build. This shortens the build time, but dependency relations or other conditions may sometimes prevent the build from completing normally. In this case, repeat Build HMI.

4-12 Offline Comparison

HMI projects are not included in offline comparisons. Even if you perform an offline comparison for a project that contains an HMI as a device, information on the HMI is not included in the comparison results.

4-13 Resource Usage

The resource usage status in a device can be displayed.



Symbol	Item	Description
(a)	The number of variables in the internal retained memory	Displays the number of variables set to be retained in the form of a graph.
(b)	Memory that is occupied by variables in the internal retained memory	Displays the size of the memory used for variables set to be retained in the form of a graph.
(c)	Variable	Displays the total number of global variables in the form of a graph. The number of variables for each individual device can be checked by hovering the mouse over the graph.
(d)	Maximum number of user alarm logs	The value set at **Internal Retained Memory Settings** - **Maximum Number of User Alarm Logs** at the device settings is displayed.
(e)	User alarm log predicted value	When additional information is used by user alarms, the predicted number of user alarms that can be saved to internal retained memory is displayed.

5

Objects

This section describes the objects that are provided as standard features.

5-1 Objects 5-2

5-1-1 Object List 5-2

5-1-2 Object Attributes 5-4

5-1-3 Using Objects 5-8

5-2 Examples of Using Objects 5-12

5-2-1 Displaying a PDF File 5-12

5-2-2 Displaying a User Alarm 5-14

5-2-3 Inserting a Variable Value in a User Alarm 5-16

5-2-4 Displaying a Trend Graph 5-18

5-2-5 Displaying a Broken-line Graph 5-21

5-2-6 Using a Recipe 5-24

5-2-7 Setting the Order of Automatic Move of Input Focus 5-28

5-2-8 Displaying Text Strings by Indirect Addressing 5-31

5-2-9 Creating Buttons with the Lamp Function 5-33

5-2-10 Creating Buttons to Output Operation Log Files 5-35

5-1 Objects

Basic objects are provided in the Sysmac Studio as standard features. You can use these objects to easily create pages.

5-1-1 Object List

The following objects are provided.

Functional Objects

A functional object provides some sort of function by itself.

Classification	Object name	Description
Buttons*1	Button object	Used to execute an action without writing a value to a variable.
	Momentary Button object	Used to change the value of the specified variable to True only while the object is being touched.
	Set Button object	Used to change the value of the specified variable to True when the object is touched.
	Reset Button object	Used to change the value of the specified variable to False when the object is touched.
	Toggle Button object	Used to toggle the value of the specified variable between True and False when the object is touched.
Lamps	Bit Lamp object	Lit while the result of the condition expression is True.
	Data Lamp object	Used to change the color of the lamp according to the value of a condition expression.

Classification	Object name	Description
Standard controls	CheckBox object	Used to change a variable to True or False depending on whether the check box is selected. Also, the check box can be displayed when the value of the specified variable changes.
	Data Display object	Used to display numeric values or text strings.
	Data Edit object	Used to display and enter numeric values or text strings.
	DateTime object	Used to display the value of a variable as a date and time.
	DropDown object*2	Used to store the value that corresponds to the item selected in a drop-down list in a variable. Also, the item that corresponds to the specified variable value is displayed.
	Image object	Used to display an image. Refer to *A-2 Supported Formats* on page A-5 for the supported formats.
	Label object	Used to display a fixed text string. Also, if you use DisplayValue for the animation property, you can display the value of the variable.
	ListBox object	Used to store the value that corresponds to the item selected in a list in a variable. Also, if the value of the specified variable is changed, you can move the focus to an item that corresponds to the new value of the variable.
	Radio Button object	Used to set the specified variable to the value that was set for the selected button. Only one of the specified group of buttons can be selected. Also, the option button can be displayed when the value of the specified variable changes.
	Slider object	Used to set the specified variable to the value that corresponds to the position of the slider. Also, if the value of the specified variable is changed, you can move the slider to the position that corresponds to the new value.
	Tab Control object	Used to partially change pages within the screen. Several tab pages are used to change the displayed contents by changing the tab page. You can create a maximum of 64 tab pages for a Tab Control object, and up to 10 Tab Control objects on a screen.
	TextBox object	Used to display a fixed text string. A TextBox object differs from a Label object in that the object can be enabled or disabled in the properties. Normally use Label objects that operate faster.
Gauges*3	Gauge Object	There are several types of gauges provided, such as one with a needle that rotates in a circle and one with a needle that moves in a straight line.
HMI controls	Broken-line Graph object	Used to display broken-line graphs. There is no limit to the number of traces, but a maximum of 16 is recommended.*4
	Media Player object	Used to create an object that plays video. Refer to *A-2 Supported Formats* on page A-5 for the supported formats.
	Recipe Viewer object	Used to display the contents of a recipe.
	Trend Graph object	Used to display data from data logging as a graph. Up to 8 analog traces and 16 digital traces are supported.*4*5
	User Alarms Viewer object	Used to display a user alarm.

*1. You can change the object type with the Properties setting.

*2. The background color differs between Sysmac Studio and the NA Unit.

*3. The size of the font of the gauge object scale may vary between Windows 10 and other operating systems.

*4. The color Alpha is disregarded for these objects. For this reason, if a transparent color is set, it will be drawn as a non-transparent color. A combined total of up to 8 Broken-line Graph objects and Trend Graph objects can be placed in one project.

*5. A maximum of 25,000 points can be drawn in Trend Graph objects. Any part that exceeds 25,000 points will not be drawn.

Precautions for Correct Use

Do not switch input control while operating a slider object or the scroll bar of an object.
The object may not be operated from then onward.

Graphic Objects

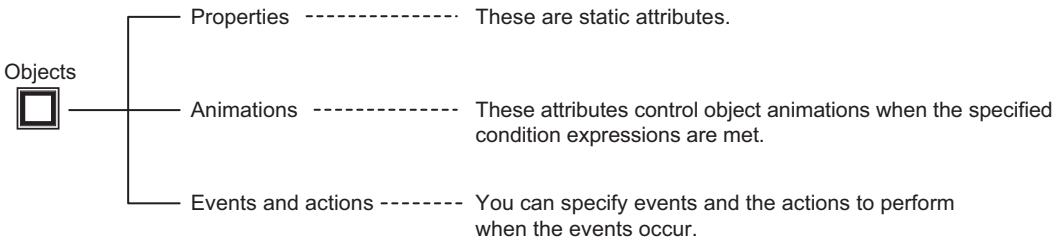
A graphic object does not provide any specific function by itself. You must add functions by using events and actions. Many different graphic objects are available. Some typical ones are described in the following table.

Classification	Object name	Description
Shapes	Curve object	Used to draw a curved line. You can double-click the graphic object to edit it.
	Ellipse object	Used to draw a circle or ellipse.
	Line object	Used to draw a straight line. You can double-click the graphic object to edit it.
	Polygon object	Used to draw a polygon. You can double-click the graphic object to edit it.
	Polyline object	Used to draw connected straight lines. You can double-click the graphic object to edit it.
	Rectangle object	Used to draw a rectangle.
	Triangle object	Used to draw a triangle. You can double-click the graphic object to edit it.

5-1-2 Object Attributes

There are the following three types of attributes for objects.

- Properties
- Animations
- Events and actions



Properties

Properties

Properties are the static attributes of an object.

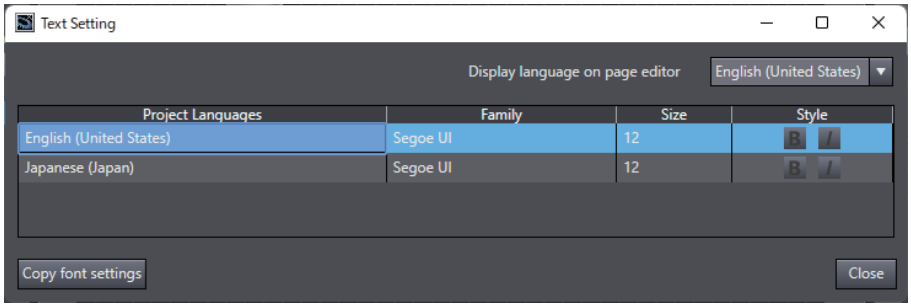
These include settings for the names and other general properties, colors, positioning, and other display properties, and assigned condition expressions or variables, as described in the following table.

Properties

Properties		Description
General		You can set the name of the object and check the object type.
Appearance		You can set the object color, shape, label, etc.
Behavior		You can set condition expressions, variables, and delay times to assign to the object.
	Expression	You can specify a condition expression that uses variables.
	Variable	You can specify a variable.
	IsEnabled, DoubleTouch-Time, OnDelayTime, etc.	You can make settings to enable the object, determine the double-touch interval, set the ON-delay time, etc.
Layout		You can set the position and size of the object.
Security		You can set security for the object.
Pointer Marker		You can set the needle marker for a gauge.
Ranges		You can set the range for a gauge.
Scale Bar		You can set the scale bars for a gauge.
Tick Label		You can set scale labels for a gauge.
Tick Major		You can set the major scale division labels for a gauge.
Tick Minor		You can set the minor scale division labels for a gauge.
Needle		You can set the needle for a gauge.
Needle Cap		You can set the needle cap for a gauge.
Data		You can set the display target of a Trend Graph or Broken-line Graph.
Left Axis		You can set the left axis for a graph.
Right Axis		You can set the right axis for a graph.
Time Scale		You can set the time axis for a graph.

Additional Information

- **Encoding** of data display objects is used when a numeric type array variable is assigned to **Expression**. To designate character encoding of String type variables, use **Communications Configuration - Encoding** in Device References.
- To copy settings related to the font of objects to other languages in projects set with multiple languages, open the font setting in Properties, and click **Copy font settings**. Settings related to the font of the default language are copied to other languages.



Notation for *Expression*

If you specify a BOOL variable, e.g., for a Lamp object, you can specify an expression for the Behavior property.

Examples of the expression notations are given below.

Example 1: Executing a Function when a Boolean Variable (*blnSample* in this Example) Is True

blnSample=True

Example 2: Executing a Function when an Integer Variable (*intSample* in this Example) Is Less Than 20

intSample<20

Example 3: Executing a Function when a Boolean Variable (*blnSample* in this Example) Is True and an Integer Variable (*intSample* in this Example) Is Less Than 20.

(blnSample=True) AND (intSample<20)

Example 4: Setting a Value by Adding 100 to an Integer Variable (*intSample* in this Example)

intSample+100

Notation for *CustomDisplayFormat*

If you set *ValueFormat* to *Custom* for a Data Display object, you must set the custom display format (*CustomDisplayFormat*). The basic format is 0:*****, where ***** is replaced with the result of *Expression*.

This format follows the specifications of custom numeric format strings in Visual Basic. For details, refer to the Microsoft website or to commercially available reference materials.

Example 1: Display When Result of Expression is 1.234 and {0:00.0000} Is Specified

01.2340

Example 2: Display When Result of Expression is 1.234 and {0:##.####} Is Specified

1.234

Example 3: Display When Result of Expression is 1.234 and X={0:##.####} Is Specified

X=1.234

Animations

Animations

You can use animations to change the status of the object according to *Expression* (condition expression).

As described below, you can specify the status when the condition expression is met, such as flashing, enabling/disabling operation, size/coordinate changes, and displaying/hiding the object.

Animation List

Animation name	Description
Blink	When the condition expression is met, the object flashes in the specified color.
ColorChange(Analog)	Changes the color of the object according to a value.
ColorChange(Boolean)	Changes the color of the object according to True/False status.
DisplayValue	Displays a value based on a condition expression.
Enable	Enables operating the object when the condition expression is met.
Move	Changes the coordinates of the object according to specified condition expressions.
PercentageFill	Fills a graphic figure based on a condition expression and a percentage between the upper and lower limits.*1
ResizeHeight	Changes the height of the object according to a specified condition expression.
ResizeWidth	Changes the width of the object according to a specified condition expression.
Rotate	Rotates a graphic object based on a condition expression.
Visibility	Displays the object when a condition expression is met.

*1. Be sure to set the background color of objects to Solid Color. Operation when a background color other than Solid Color is set is not guaranteed.

Events and Actions

Events and Actions

You can make settings for object events and corresponding actions.

You can specify events and the actions to perform when the events occur.

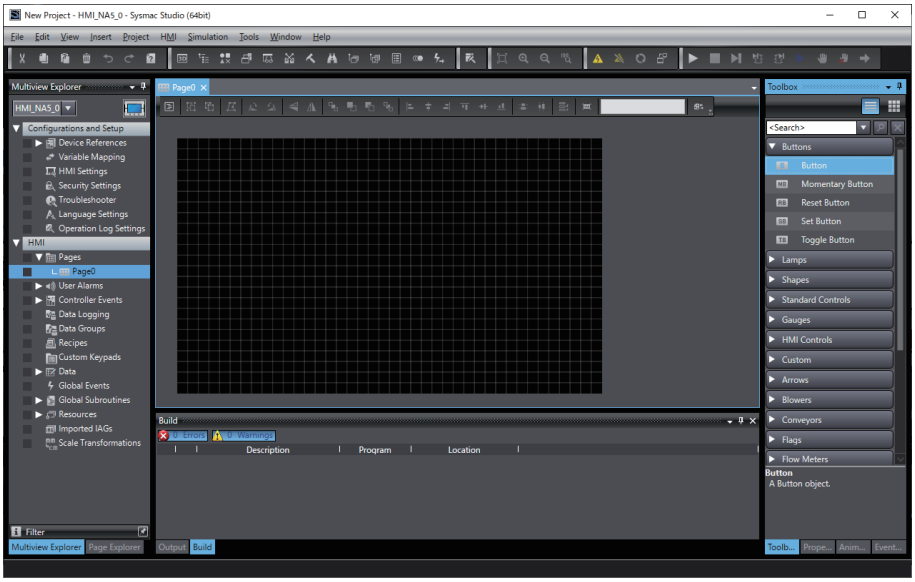
You can specify the required conditions and operations, such as executing a specified subroutine when a function key is touched.

The events and actions are listed in *A-1 Events and Actions* on page A-2.

5-1-3 Using Objects

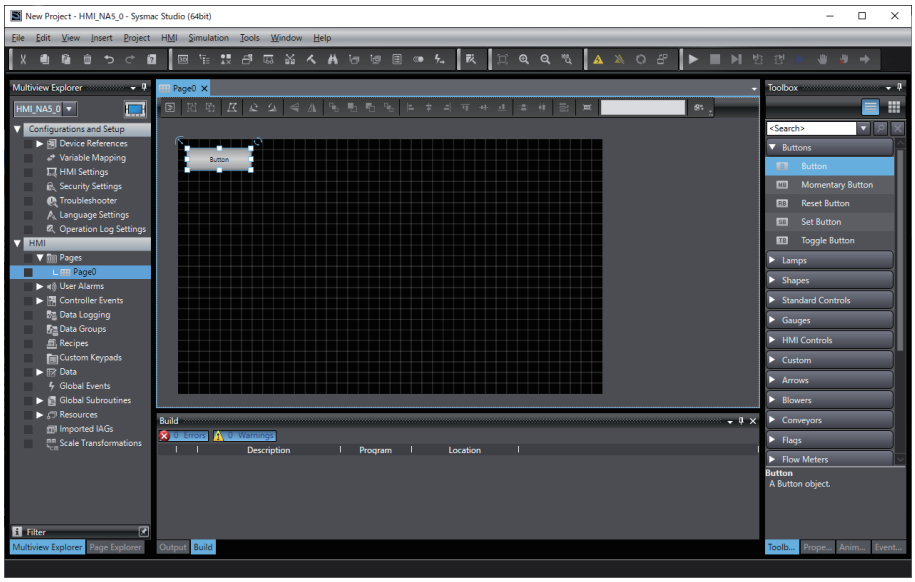
The objects are provided in the Toolbox on the right side of the window.

You can create them on pages and set the properties and animations and also the events and actions.



Creating Objects

1 Drag the objects from the Toolbox and drop them on the page.



Additional Information

- You can drag and drop a variable registered in the controller's global variable table or a contact or coil placed in the ladder editor onto the page editor to automatically register the controller variable as an HMI-mapped global variable, and create an object corresponding to the variable type.
- When pasting an array variable to such as an HMI property variable, brackets are automatically converted to "()" which are used in HMI. Same conversion is performed when pasting in the opposite way.

Setting Properties

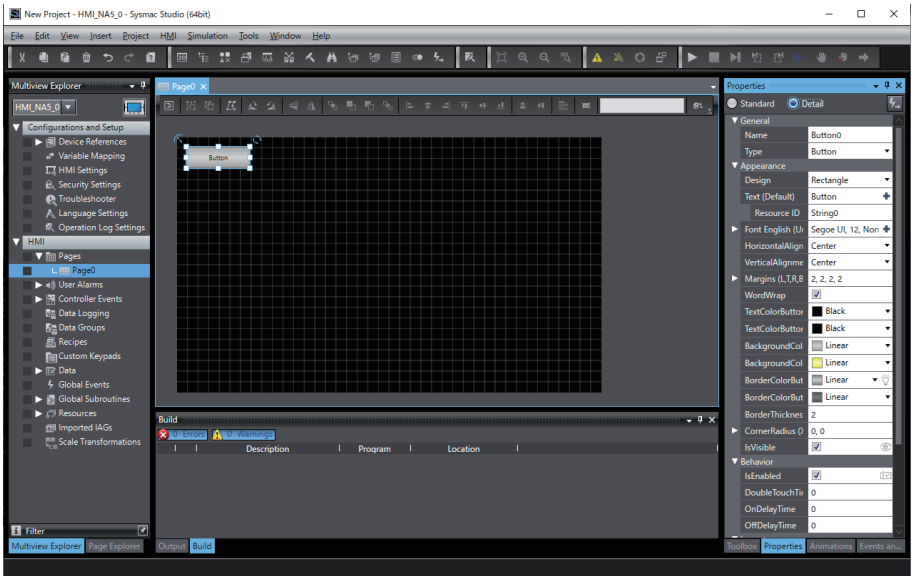
1 Double-click the object.

Additional Information

You can also use the following methods to display the properties.

- Select ***Properties*** from the View Menu.
- Right-click the object and select ***Properties*** from menu.

2 The properties are the static attributes of the objects.



Make the following settings as required.

Setting Animations

You can set animations by clicking the icon displayed for the property on which the animation acts. The procedure shown here is for the example of Blink in a button object.

The animation icon has three states.

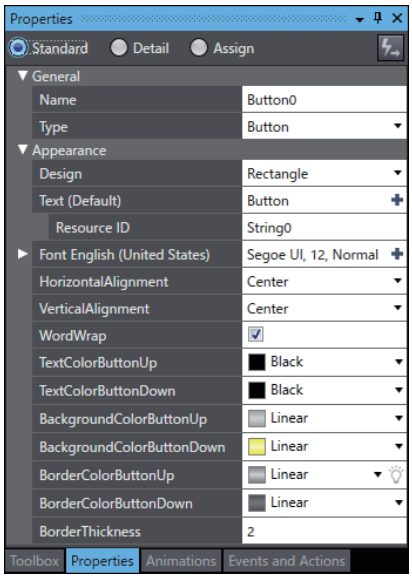
: Set. No error.

: Not set.

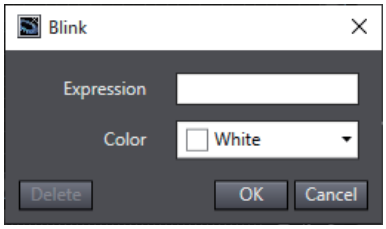
: Set. Error.

1 Double-click the button object to display the Properties Window.

2 Click for **BorderColorButtonUp**.



3 Set the expression and color, and click **OK**.



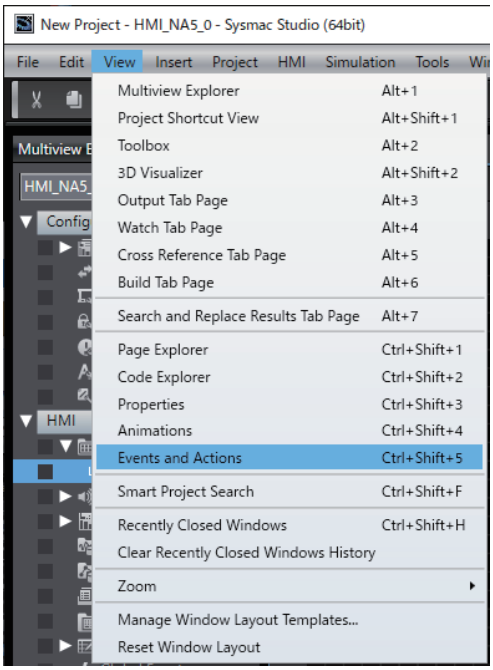
Additional Information

Click **Delete** to delete the animation settings.

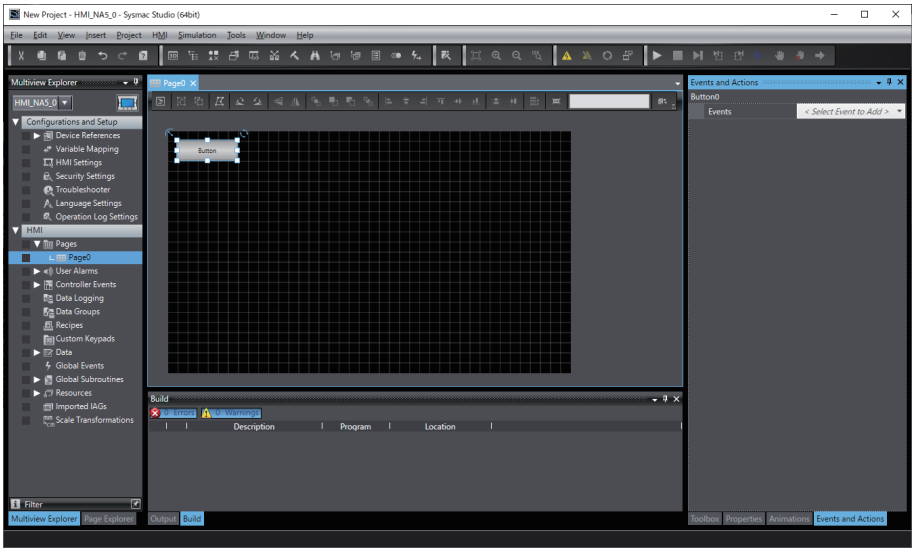
Animations can also be set from the Animations Window.

Setting Events and Actions

1 Select ***Events and Actions*** from the View Menu.



2 You can set the actions for events in the Events and Actions Window.



Additional Information

The Events and Actions Window can also be displayed using in the Toolbar or Properties Window.

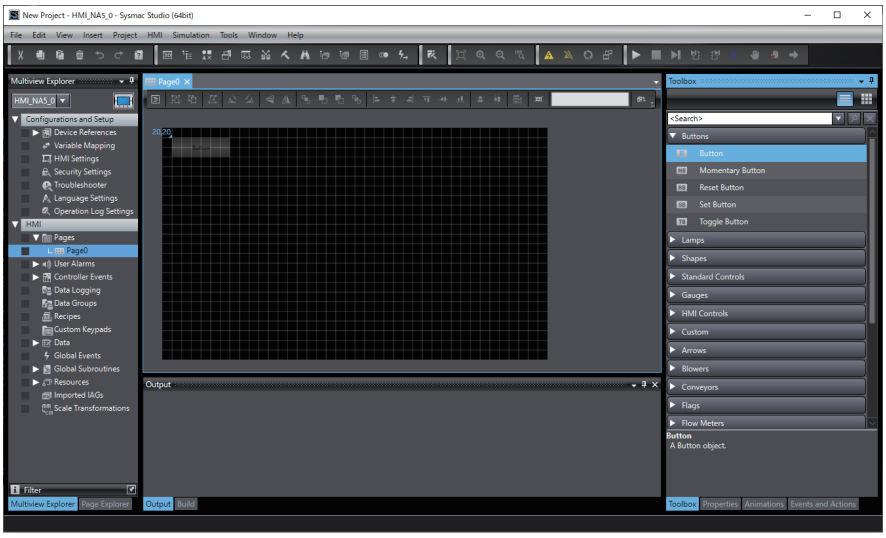
5-2 Examples of Using Objects

This section provides examples of using objects.

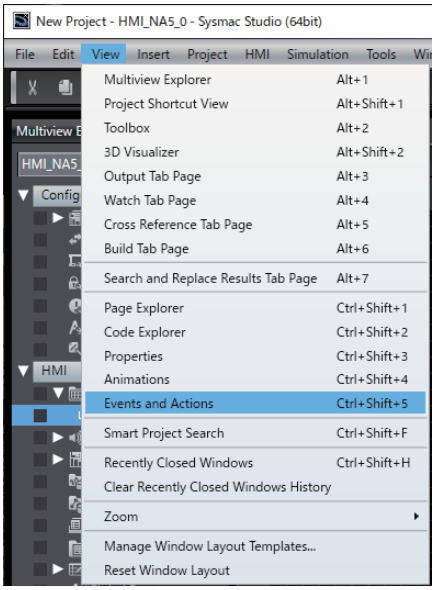
5-2-1 Displaying a PDF File

The following example shows how to display a PDF file full screen when a button is touched.

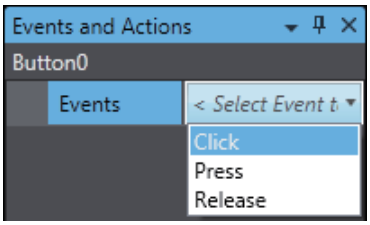
1 Place a button on the page.



2 Select ***Events and Actions*** from the View Menu.

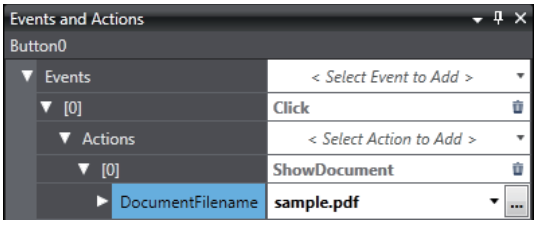


3 Select *Click* from **Events**.



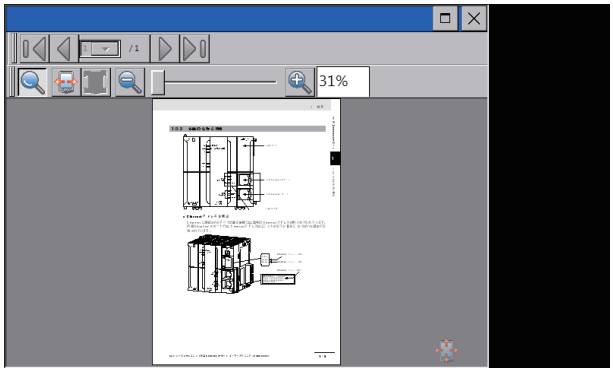
4 Select *ShowDocument (FullScreen)* from **Actions**.

5 Set **DocumentFileName** to the name of the file to display.



Transfer the project to the HMI and confirm the operation.

The PDF file should be displayed when the button is touched.

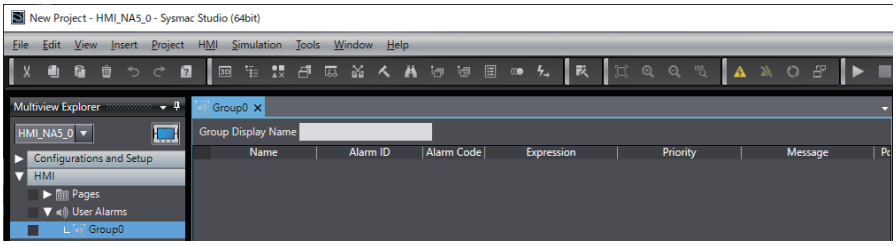


5-2-2 Displaying a User Alarm

The following example shows how to register a user alarm and display a message when the user alarm occurs.

This example creates a user alarm that displays the message *Alarm1* in a confirmation dialog box when the *blnAlarm1* variable changes to True.

1 Double-click **Group0** under **HMI** – **User Alarms** in the Multiview Explorer.



2 Right-click in the user alarm table for the new group and select ***Add*** from the menu.

3 A row is added. Make the following settings in the new row. Use the default values for settings that are not specified.

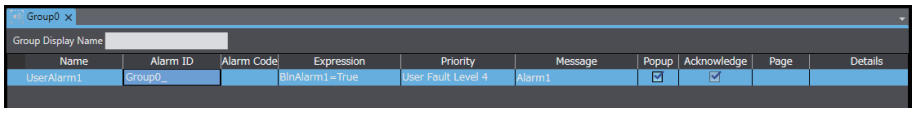
Name: UserAlarm1

Expression: blnAlarm1=True

Message: Alarm1

Popup: Selected

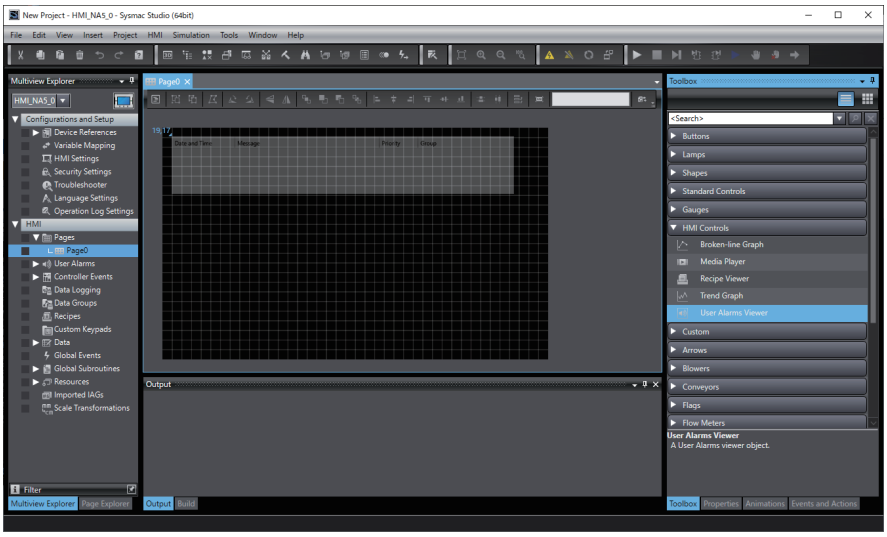
Acknowledge: Selected



Additional Information

If you set a page name for the Page parameter, the screen is switched to the specified page when an alarm occurs. Select an alarm displayed on the User Alarms Viewer, and execute the ShowAlarmPage action/function. The screen is changed to the specified page.

4 Double-click **HMI - Pages - Page0** in the Multiview Explorer. Drag a User Alarms Viewer object from **HMI Controls** in the Toolbox to the page.

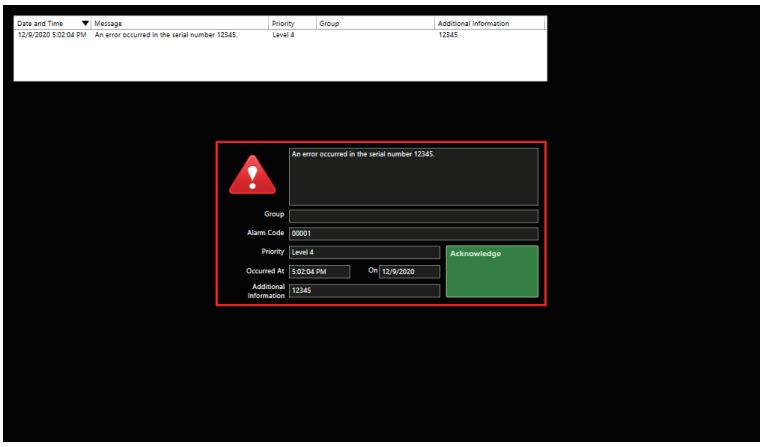


Additional Information

With Runtime Ver. 1.14 or later, the elapsed time since an alarm occurred can be displayed. Select **Elapsed Time** in the property column. The elapsed time is displayed in units of minutes.

Transfer the project to the HMI and confirm the operation.

When the *bInAlarm1* variable changes to True, the contents that was set for the User Alarms Viewer object are displayed in a confirmation dialog box.



Additional Information

You can sort the displayed items by touching a column of the User Alarms Viewer.

5-2-3 Inserting a Variable Value in a User Alarm

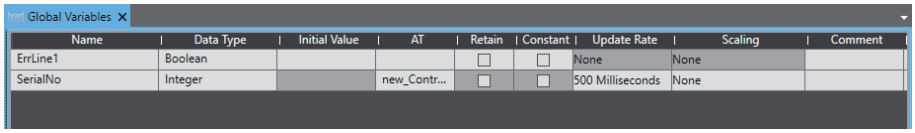
The following example shows how to insert a specific variable value into the user alarm message at the time the user alarm occurred. Refer to *5-2-2 Displaying a User Alarm* on page 5-14 for the basic setting procedure.

Additional Information

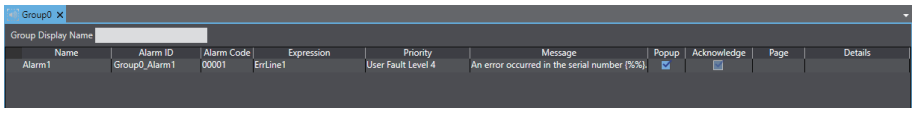
Additional information is retained together with the information of user alarms in internal retained memory. For this reason, when additional information is set, the number of user alarms that can actually be retained is less than the number set in **HMI Settings - Maximum Number of User Alarm Logs**.

For a rough estimate of the number of user alarms that can be retained in internal retained memory when additional information is used, check at **Resource Usage**.

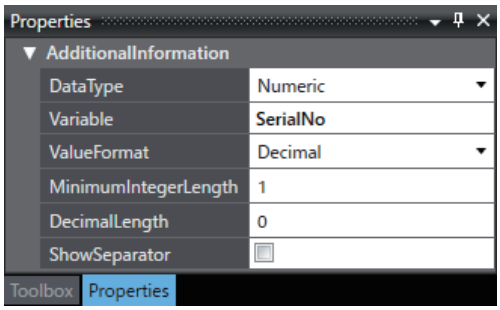
1 Create the SerialNo variable to contain the serial number.



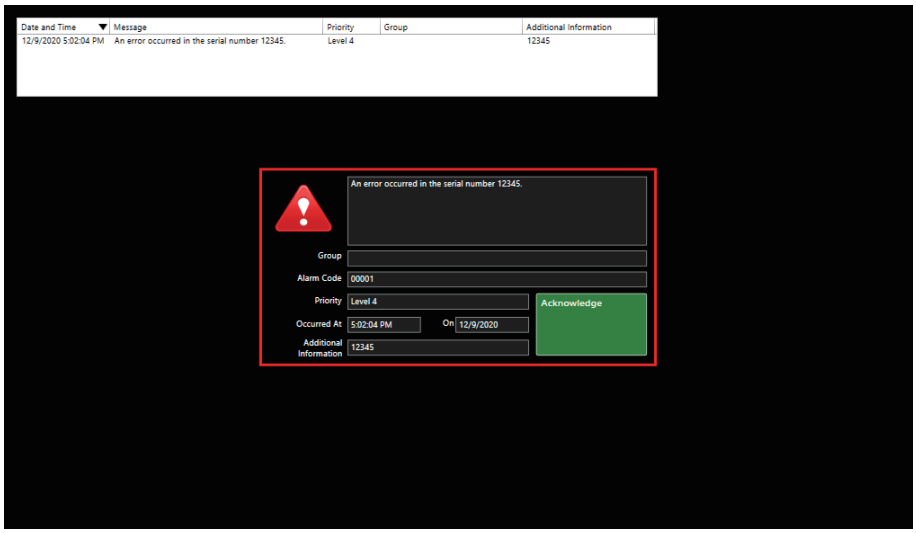
2 Set the user alarm as follows.
Add {%%} at the position in the user alarm message where you want to insert the variable.



3 With the user alarm created at Step 2 selected, display the Properties Window and set SerialNo as the variable.



Transfer the project to the HMI and confirm the operation. After setting SerialNo to 12345, the alarm message appears as follows when Alarm1 is generated.



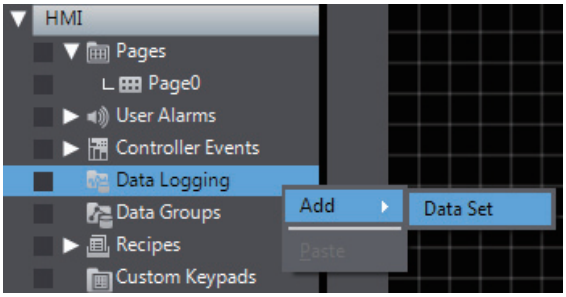
5-2-4 Displaying a Trend Graph

The following example shows how to register a data set for data logging and display a trend graph that accesses it.

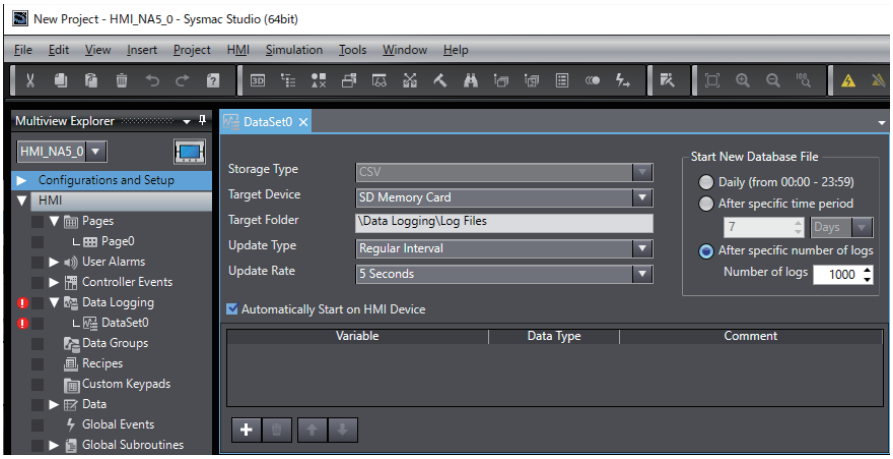
This example records log data continuously every 5 seconds and saves it in a separate file for each 24-hour period.

1 Register an integer variable called *intDatalog1* in the global variable table.

2 Right-click **HMI - Data Logging** in the Multiview Explorer and select ***Add - Data Set*** from the menu.

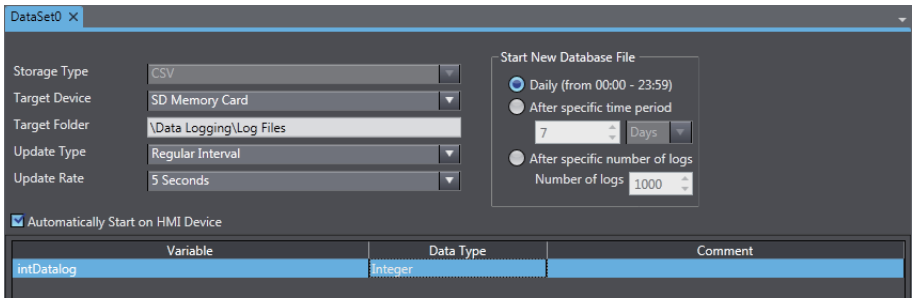


3 Double-click the new data set.

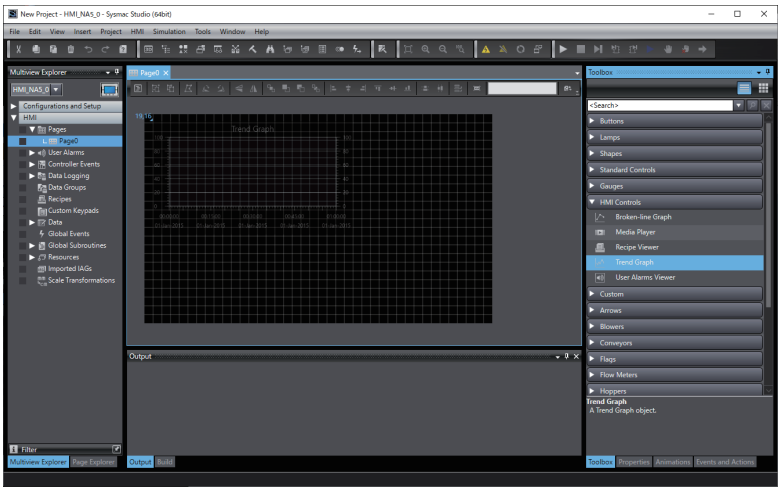


4 Right-click in the new data set grid and select ***Add*** from the menu.

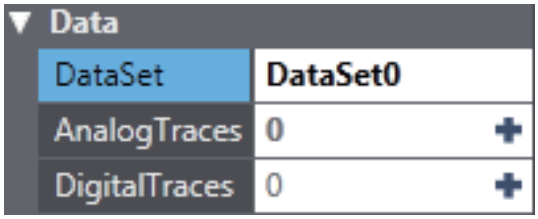
5 A row is added. Set *intDatalog1* in the *Variable* column of the new row. Use the default values for other settings.



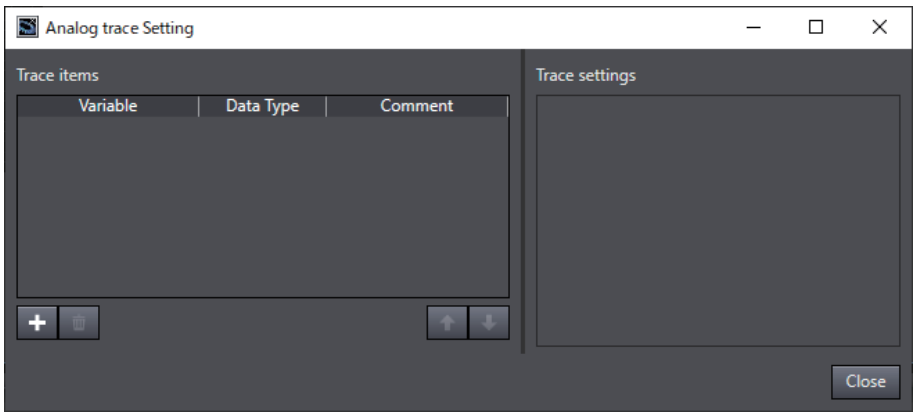
6 Double-click **Page0** under **HMI - Pages** in the Multiview Explorer. Drag a Trend Graph object from **HMI Controls** in the Toolbox to the page.



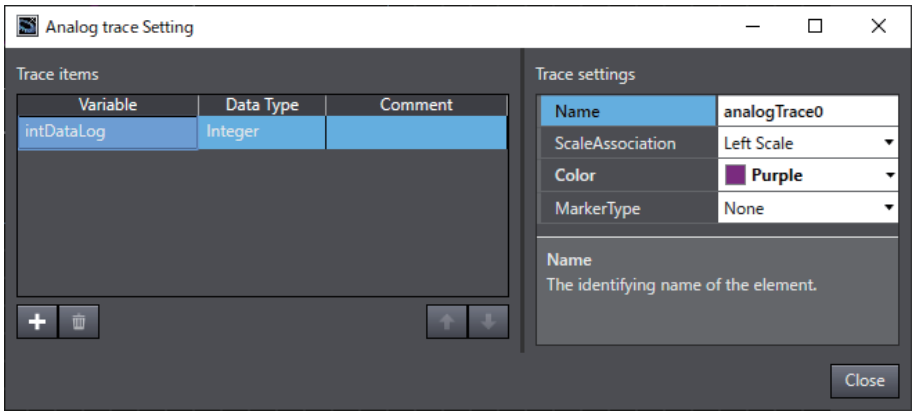
7 Set the name of the data set that you created as the data set in the properties.



8 Click ✚ for *Analog Traces* to open the Analog trace Setting.



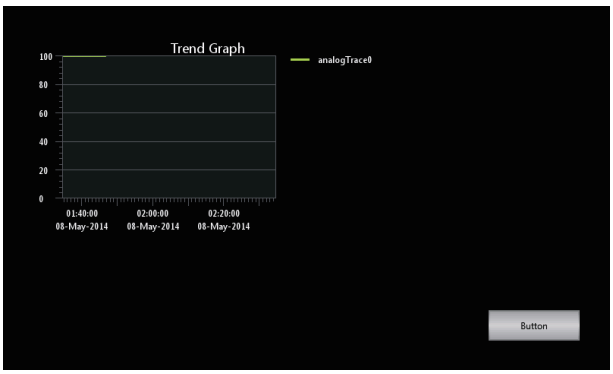
9 Click + to add a trace and set the variable to display on the graph.



10 Click **Close**.

Insert an SD Memory Card into the HMI, transfer the project to the HMI, and confirm the operation.

Every 5 seconds, the value of *intDatalog1* should be sampled and displayed on the graph.

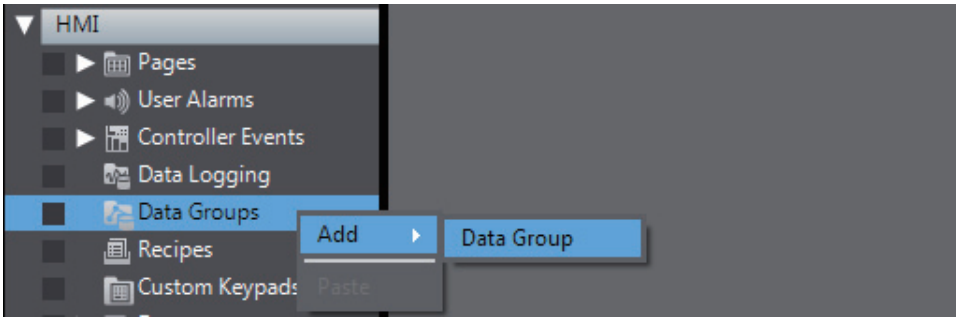


5-2-5 Displaying a Broken-line Graph

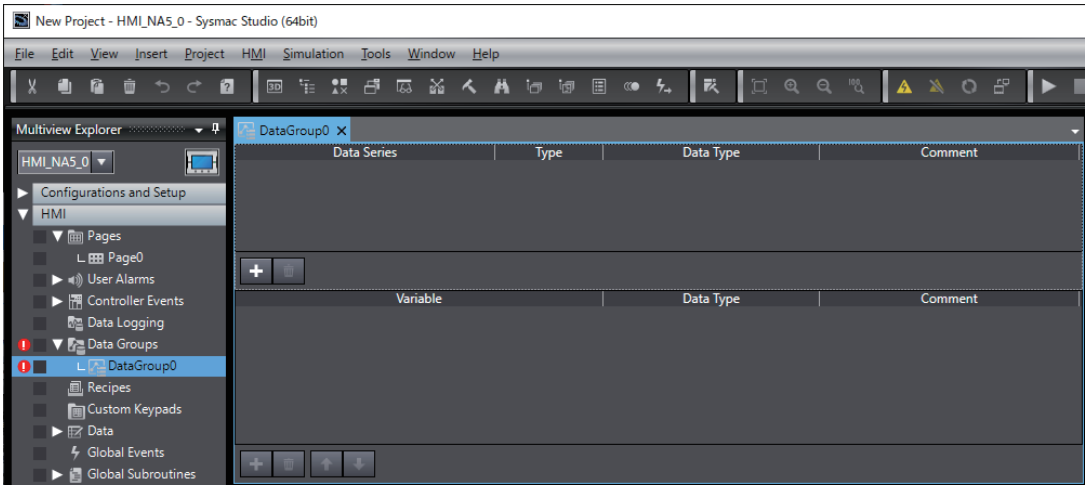
The following example shows how to use a broken-line graph.

1 Register an integer array variable called *intData(10)* in the global variable table.

2 Right-click **HMI - Data Groups** in the Multiview Explorer and select ***Add - Data Group*** from the menu.

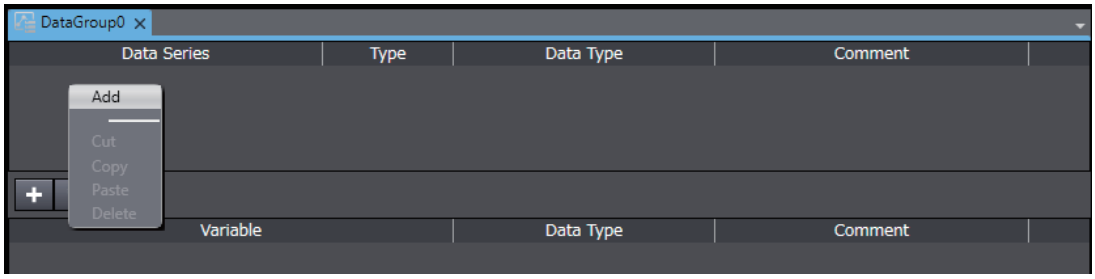


3 Double-click the new data group.

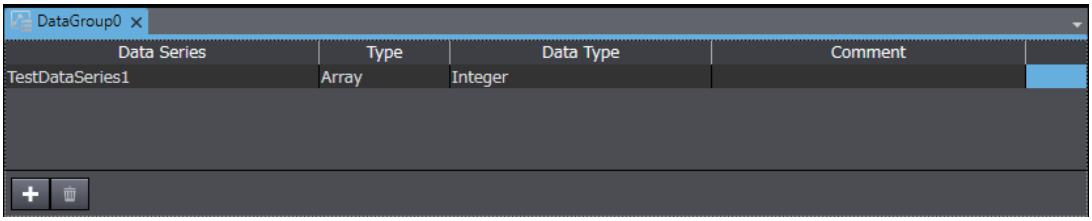


4 Register a data series in the new DataGroup0.

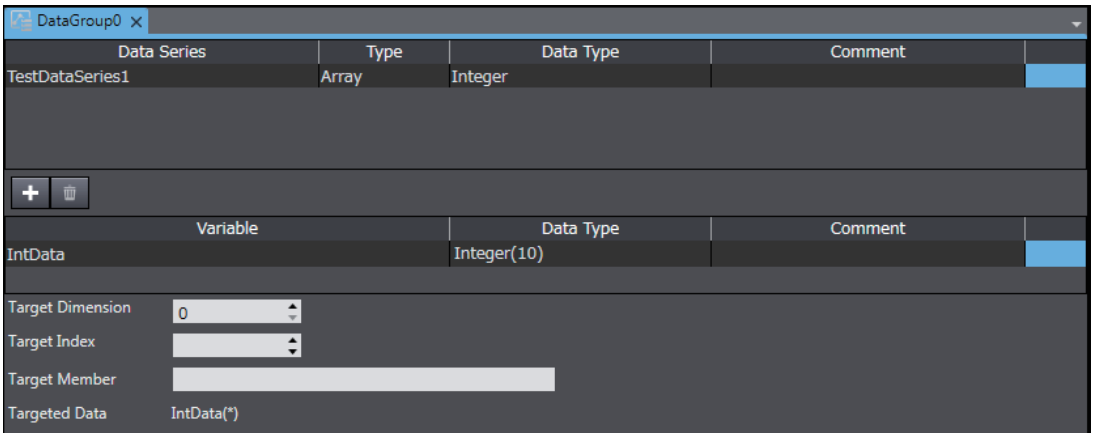
Right-click in the grid and select ***Add*** from the menu.



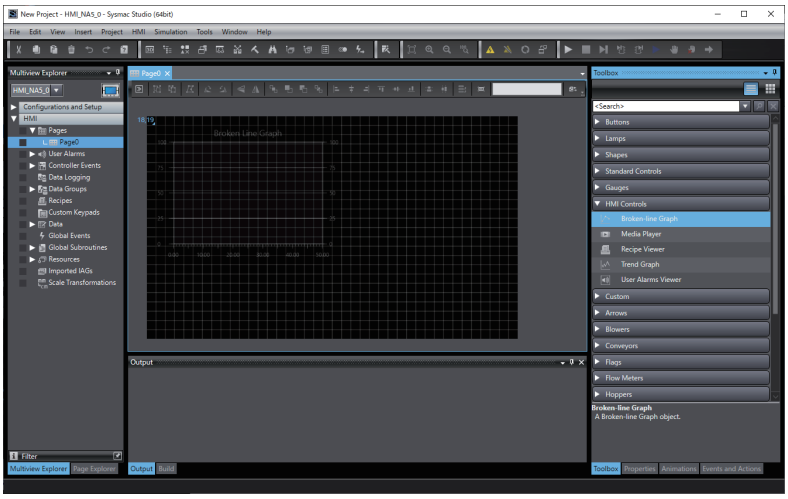
5 A row is added. Perform the following settings in the new row. Use the default values for all other settings.



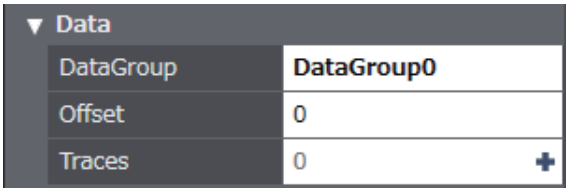
6 Set an array variable called *intData* in the added data series. Use the default values for all other settings.



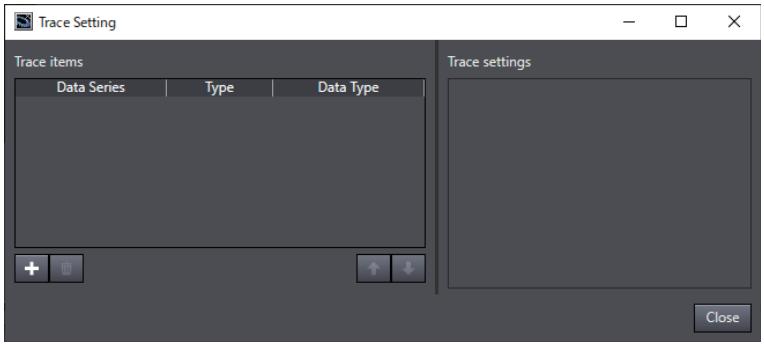
7 Go back to the Page Edit Pane. Double-click Page0 under **HMI - Pages** in the Multiview Explorer. Drag a Broken-line Graph object to the page from **HMI Controls** in the Toolbox.



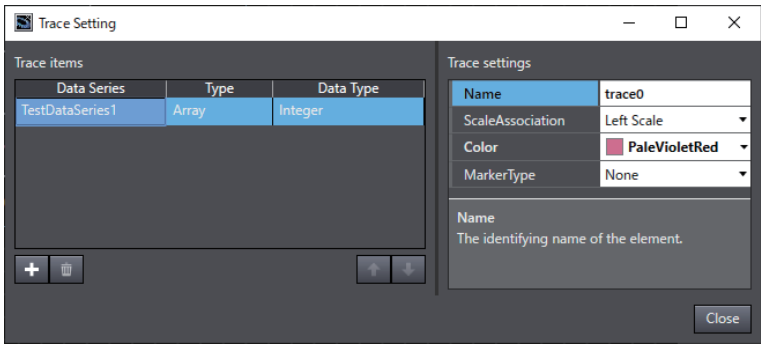
8 Set the name of the data group that you created as the data group in the properties.



9 Click ✚ for *Traces* to open the Trace Setting.



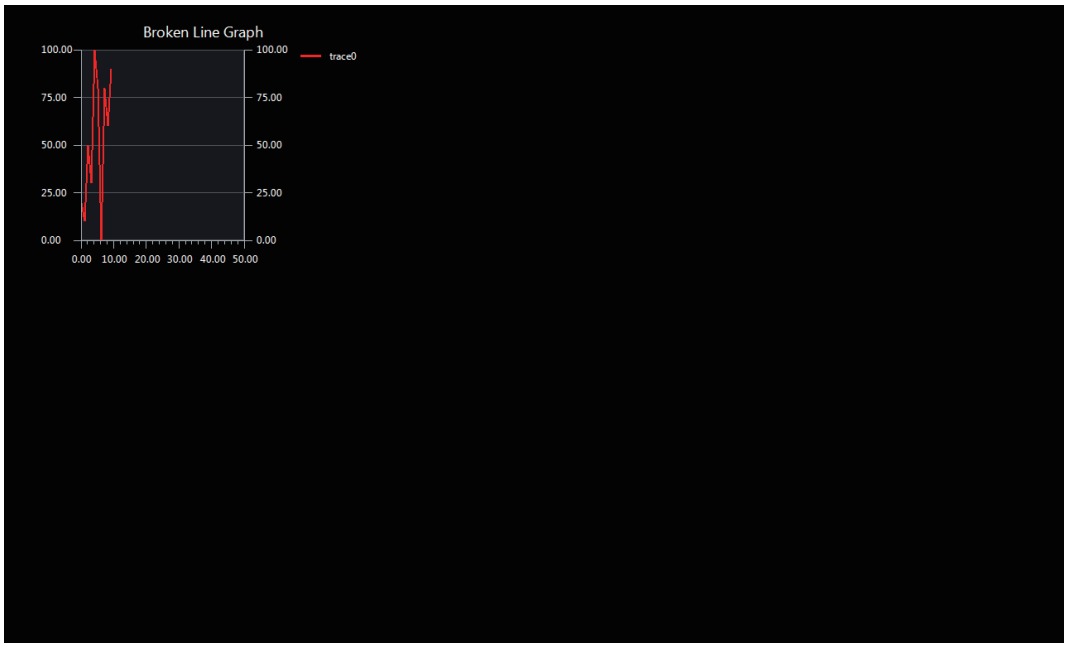
10 Click ✚ to add a trace and set the data series to display on the graph.



11 Click **Close**.

Transfer the project to the HMI, and confirm the operation.

The value of each element of *intData(10)* should be displayed on the broken-line graph.

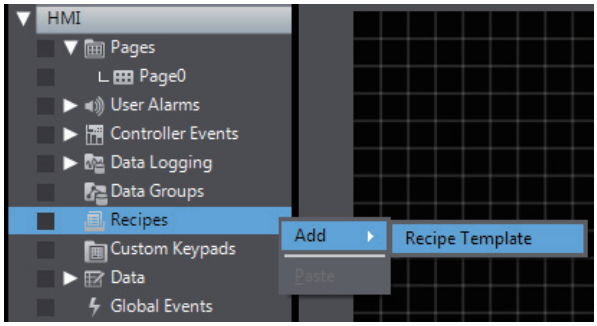


5-2-6 Using a Recipe

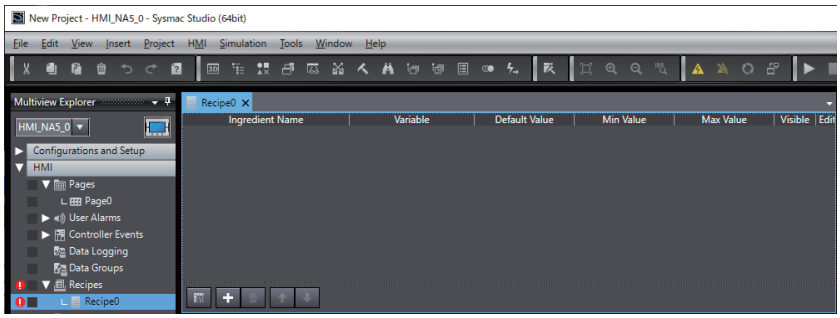
The following example shows how to use a recipe.

1 Register integer variables called *intRecipe1* and *intRecipe2* in the global variable table.

2 Right-click **HMI - Recipes** in the Multiview Explorer and select ***Add - Recipe Template*** from the menu.

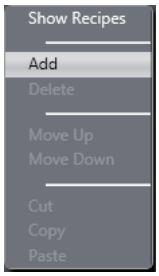


3 Double-click the new recipe template.



4 Register ingredients in the recipe template for the new recipe, Recipe0.

Right-click in the grid and select ***Add*** from the menu.



Additional Information

A recipe template is a defined data structure for a recipe.

5 A row is added. Make the following settings in the new row. Use the default values for other settings.

Ingredient Name: Recipe1

Variable: intRecipe1

Default Value: 10

Min Value: 0

Max Value: 100

Set Recipe2 as follows:

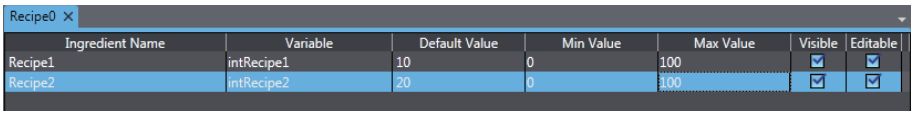
Ingredient Name: Recipe2

Variable: intRecipe2

Default Value: 20

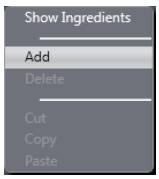
Min Value: 0

Max Value: 100



6 Right-click in the grid and select ***Show Recipes*** from the menu.

7 Right-click in the grid and select ***Add*** from the menu.



Additional Information

Recipes make settings in advance that are actually set according to the data structure.

8 A row is added. Make the following settings in the new row. Use the default values for other settings.

Recipe Name: RecipeSample1

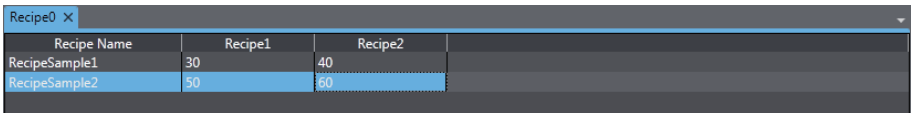
Recipe1: 30

Recipe2: 40

Recipe Name: RecipeSample2

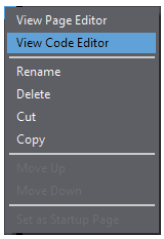
Recipe1: 50

Recipe2: 60

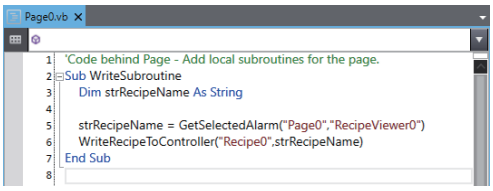


9 Create a button to transfer the recipe.

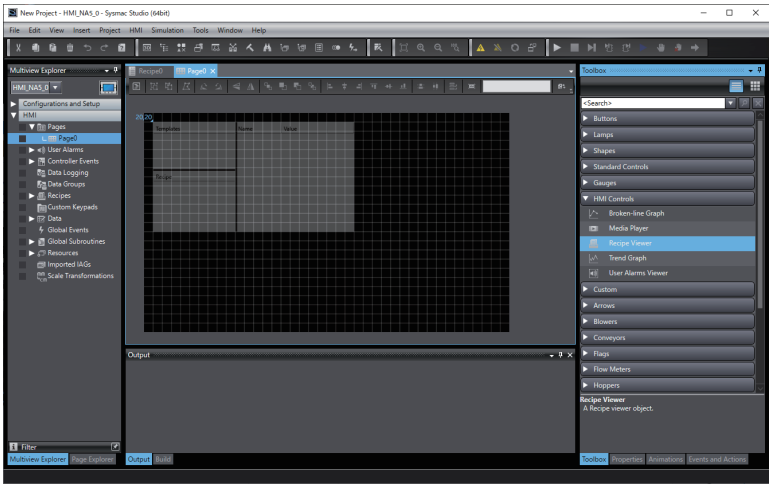
Right-click **HMI - Pages- Page0** in the Multiview Explorer and select ***View Code Editor*** from the menu.



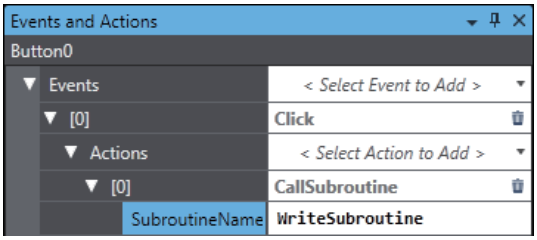
10 Create the following subroutine.



11 Double-click **HMI - Pages - Page0** in the Multiview Explorer. Drag a Recipe Viewer object from **HMI Controls** in the Toolbox to the page.

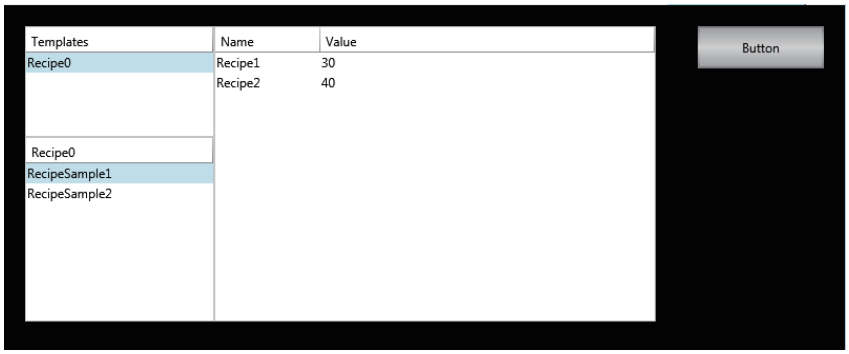


12 Then drag a Button object from **Buttons** in the Toolbox to the page and set the following event and action.



Transfer the project to the HMI and confirm the operation.

When the button is touched, the values that are set for the specified recipe should be written to the connected device.



5-2-7 Setting the Order of Automatic Move of Input Focus

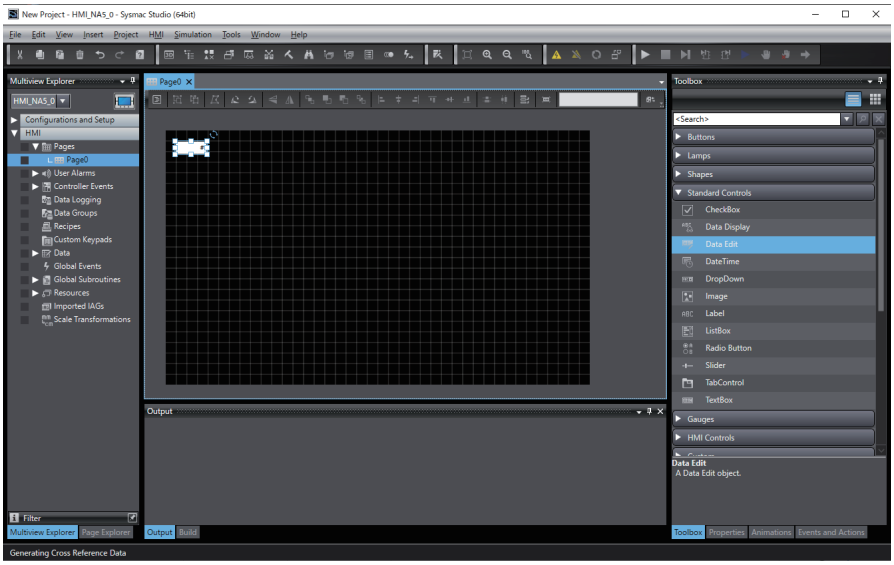
The following example shows how to consecutively enter numeric values in the multiple data input objects placed on a page.

Additional Information

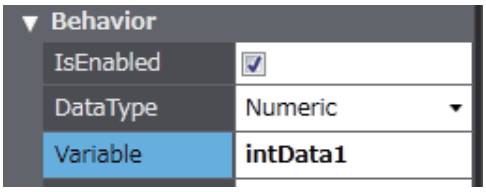
The order of automatic move of tab control objects is not supported. When the order of automatic move is set to tab control objects, the setting is ignored, and the tab moves to the object that should be moved next among the objects not placed on tab control.

1 Register an integer variable called *intData1* in the global variable table.

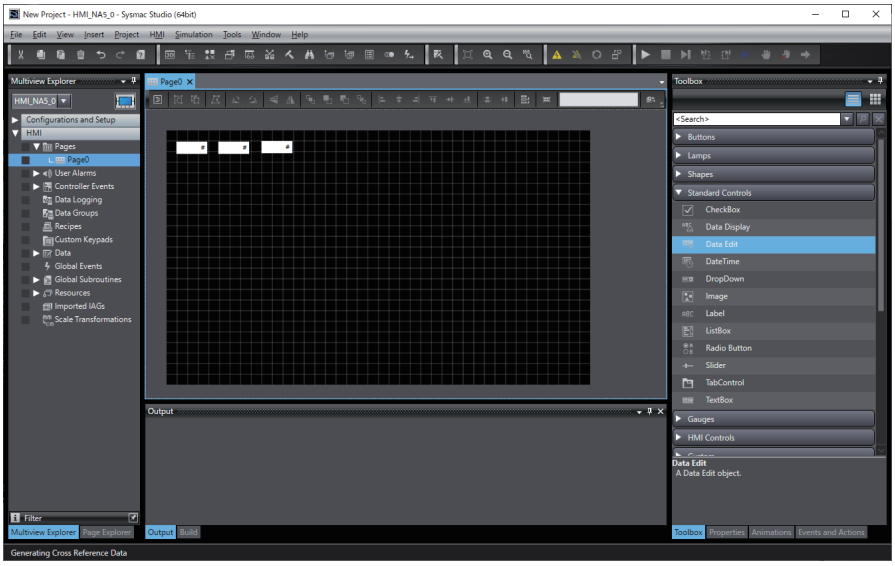
2 Place a data input object on the page.



3 Set *intData1* in **Variable** of the properties for the data input object.

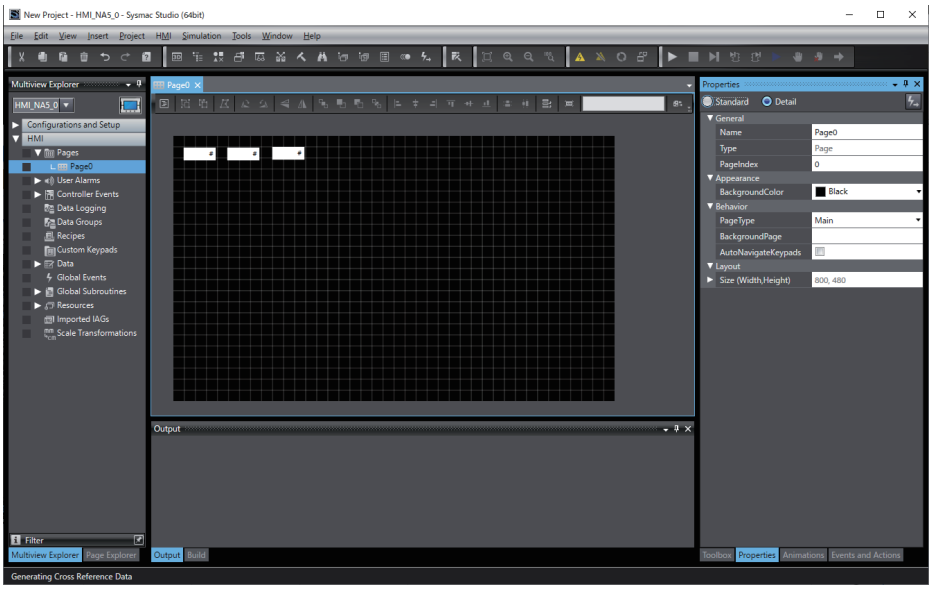


4 Place the multiple data input objects on the page by copying and pasting one.

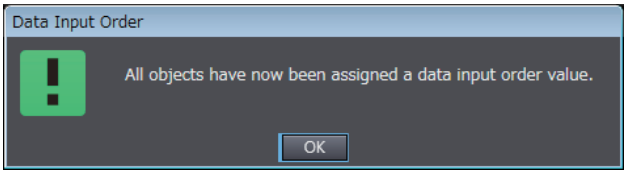


5 Click the **Data Input Order Setting** Button on the Toolbar to switch to the mode for setting data input order.

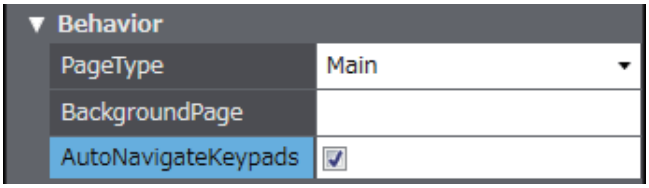
6 Click the data input objects in the order in which to move the input focus.



7 When you complete the settings, the following dialog box appears.



8 Select the *AutoNavigateKeypads* Check Box under **Behavior** in the properties of the page.



Transfer the project to the HMI, and confirm the operation.

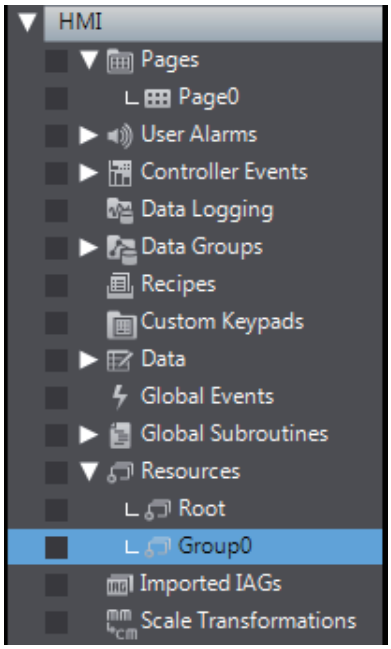
When you press a data input object, a keypad is displayed. After you complete an input, the input focus is automatically moved from one data input object to the next.

5-2-8 Displaying Text Strings by Indirect Addressing

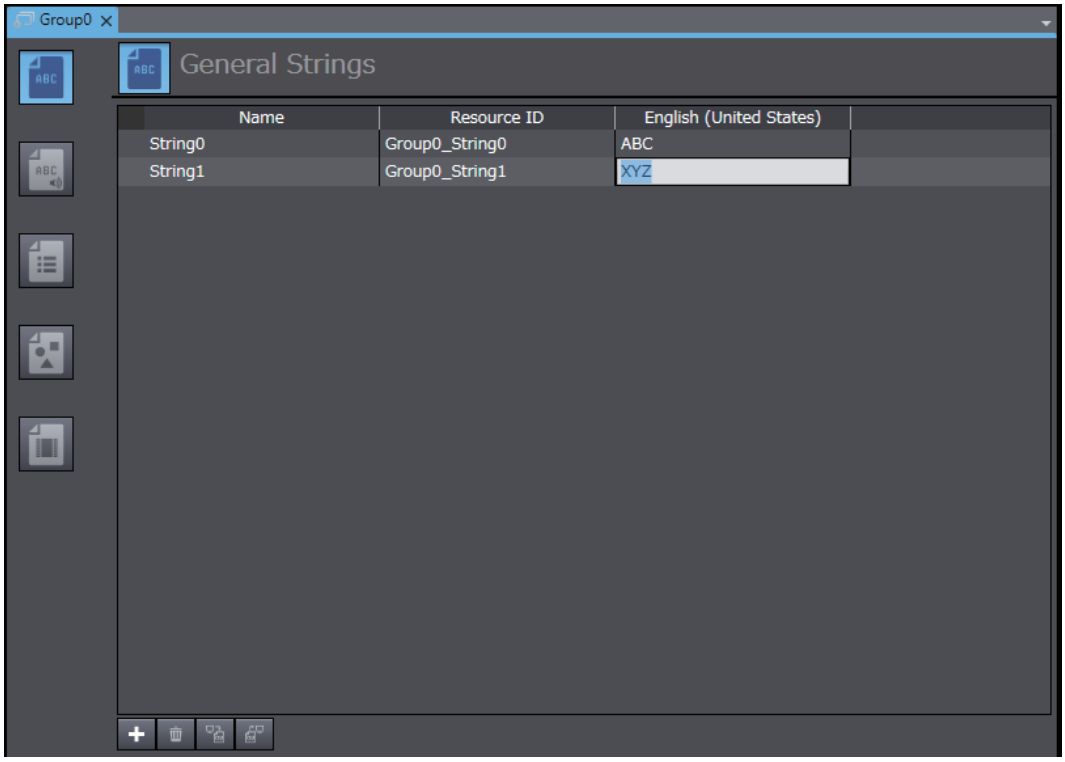
You can dynamically change text strings that are displayed on label objects or text box objects. The following example shows how to switch a text string displayed on an object by using a value specified in a variable.

1 Register an integer variable called *intData1* in the global variable table.

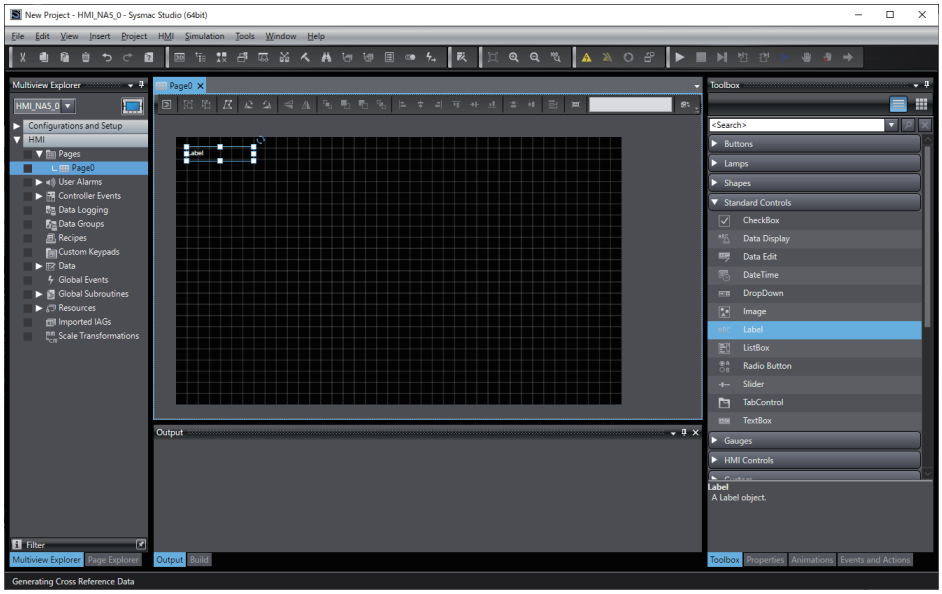
2 Register a new resource group in a Resources table.



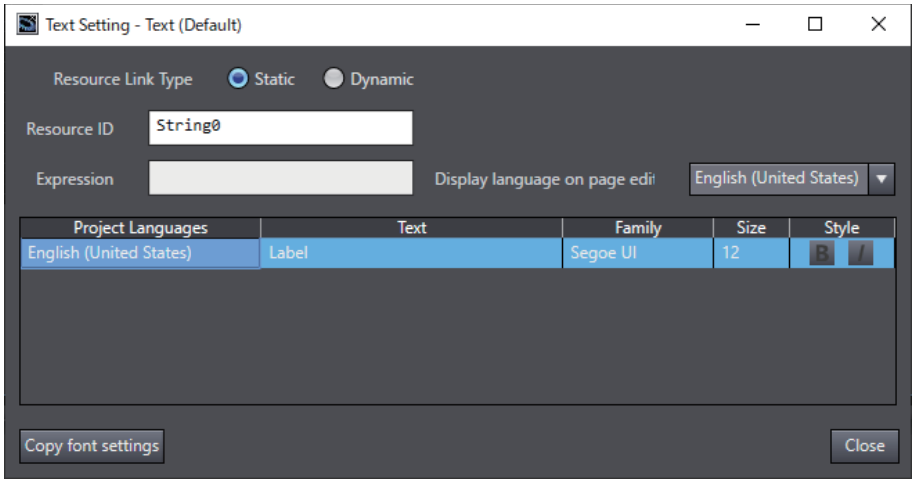
3 Open the added Resources table, and register a text string to display on an object.



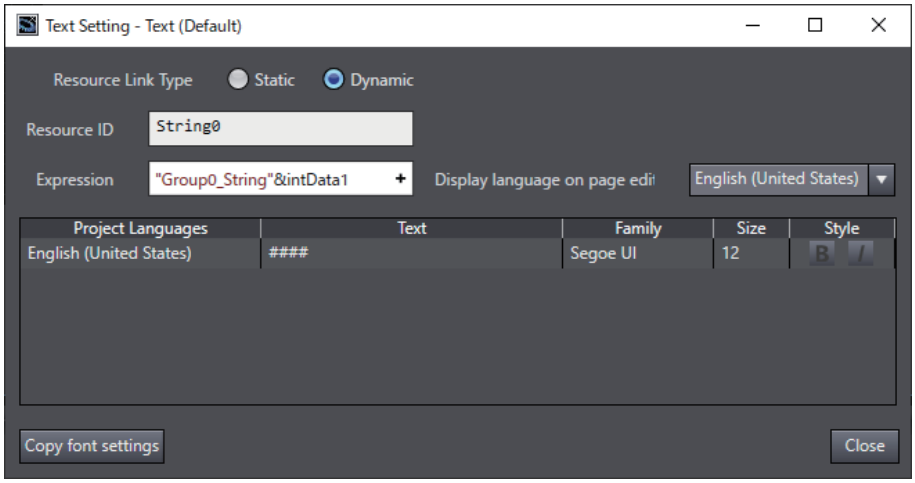
4 Open the Page Editor and create a Label object.



5 Open the properties of the Label object, and click ✚ under **Appearance - Text (Default)** to open the Text Setting.



6 Select **Dynamic** in **Resource Link Type** and set "Group0_String"& intData1 in **Expression**.



7 Click **Close**.

8 On the page, create the data input object that inputs values to intData1.

Transfer the project to the HMI and confirm the operation.

When the value in intData1 is 0, the Label object displays “ABC”.

When the intData1 value is changed to 1, the object displays “XYZ”.

Additional Information

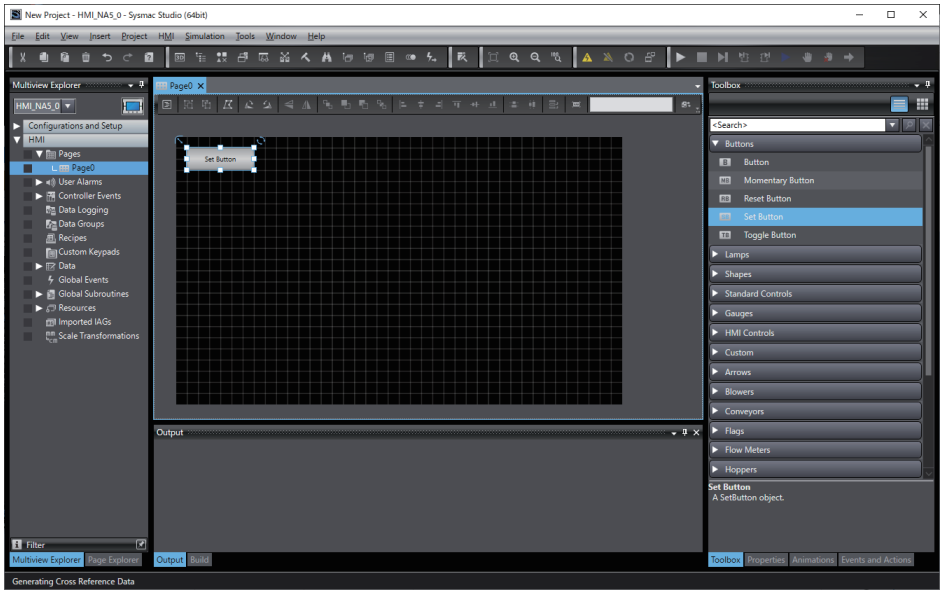
In Expression, set the text string that is displayed for the Resource ID on the Resources table.

5-2-9 Creating Buttons with the Lamp Function

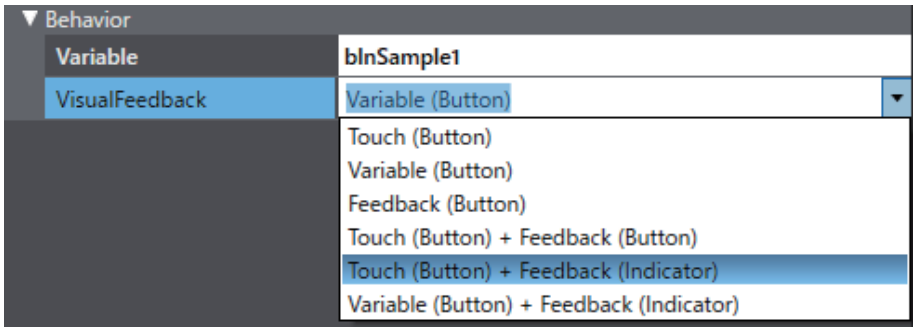
The following example shows how to create a Button that, when touched, allows the entire color to be changed as Variable 1 changes between TRUE and FALSE, and the inside indicator color changes as Variable 2 changes between TRUE and FALSE.

1 Register Boolean variables called *blnSample1* and *blnSample2* in the global variable table.

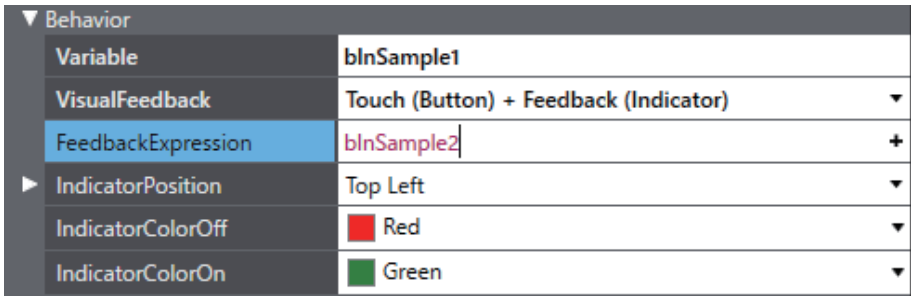
2 Place a Set Button object on the page.



3 Open the ***Properties*** Window for the Set Button object. Set blnSample1 in the ***Variable*** Box, and select Touch (Button) + Feedback (Indicator) in the ***VisualFeedback*** Box.



4 Set *bInSample2* in the **FeedbackExpression** Box.



With the above settings, pressing the Set Button object changes *bInSample1* to TRUE, which changes the entire button color.

The inside indicator color is changed as *bInSample2* changes between TRUE and FALSE.

Additional Information

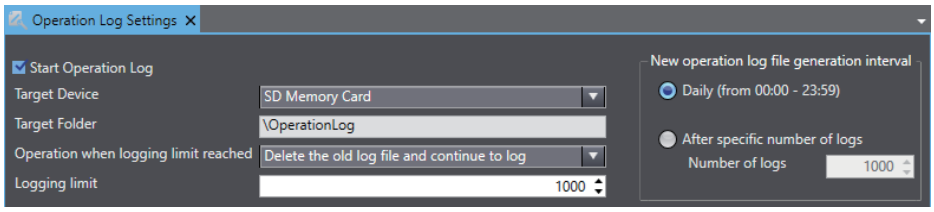
The position, size, and color of the indicators can be changed as required.

5-2-10 Creating Buttons to Output Operation Log Files

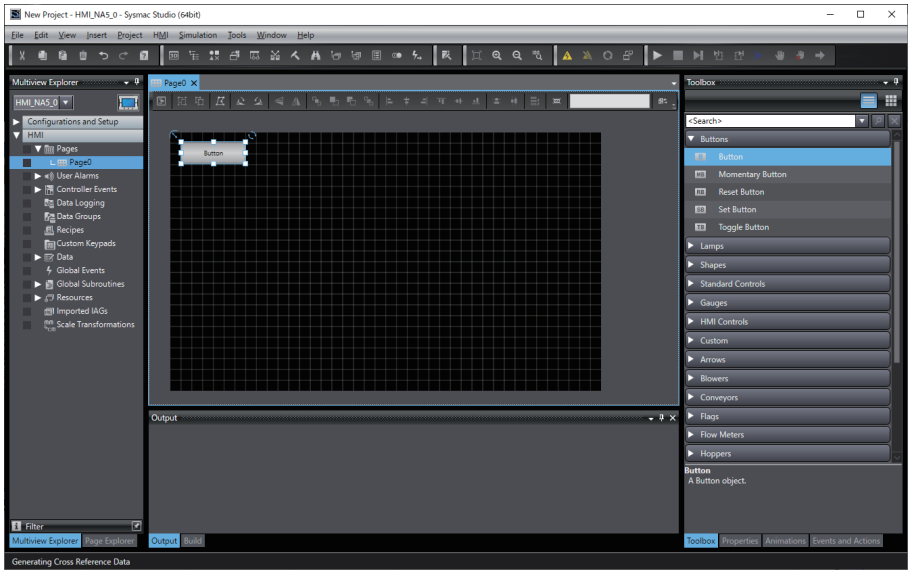
The following example shows how to create buttons to output operation log files to a SD memory card.

1 Click *Configuration and Setup - Operation Log Settings* in the Multiview Explorer.

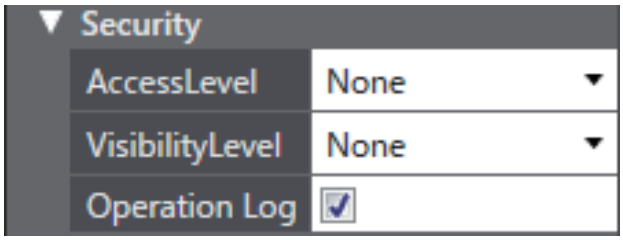
2 Select the *Start Operation* Log Check Box.



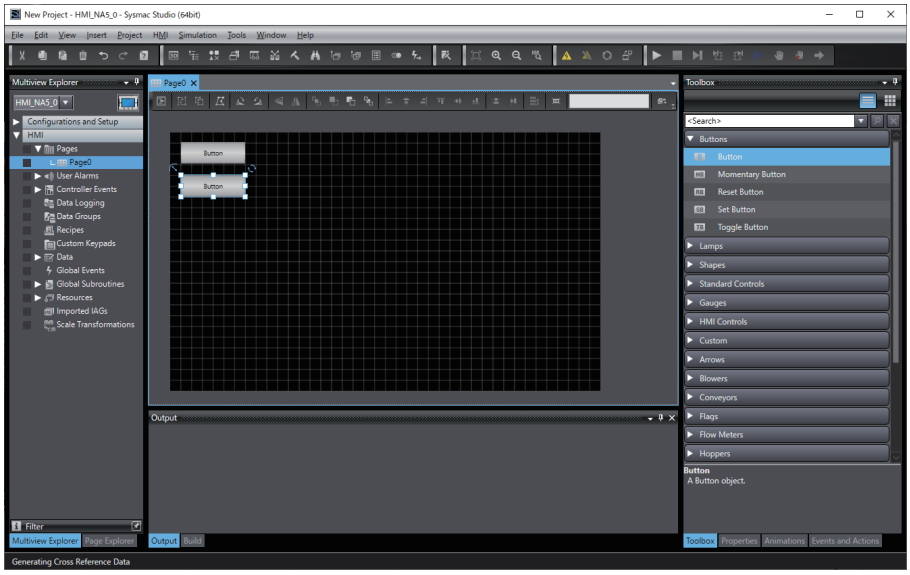
3 Place a button object on the page.



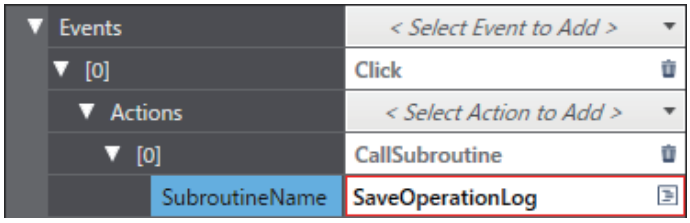
4 Select the *Operation Log* Check Box.



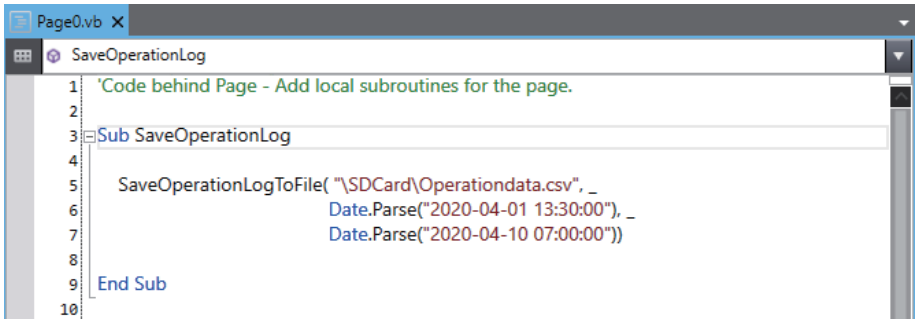
5 Place a button object on the page.



6 Set the following event and action and click .



7 Create the following subroutine.



8 Transfer To HMI.

Operation logging starts when the HMI starts.

Tap the button to record logs created in 2 to start leaving records in the Operation Log.

Tap the button created in 4 to save the operation log from 13:30:00 on April 1 2020 to 7:00:00 on April 10 2020 in the SD memory card under the file name Operationdata.csv. Refer to *A-5 Specifications of Operation Log Files* on page A-9 for the specifications of the files to be saved.

6

Connecting to the HMI

This section describes how to go online with an HMI.

6-1 Introduction 6-2

6-2 Going Online with an HMI 6-3

6-2-1 Methods for Going Online with an HMI 6-3

6-2-2 Setting the Connection Method 6-4

6-2-3 Online Connection 6-5

6-2-4 Going Online after Checking the Connection Method 6-6

6-2-5 Going Offline 6-6

6-2-6 Confirming Serial IDs 6-7

6-1 Introduction

Connecting to the HMI

You must go online with the HMI or connect to the Simulator to communicate with it from the Sysmac Studio.

The Sysmac Studio supports the following online connections for different applications.

Online connection	Connection made to	Application
Online connection	The actual HMI	To perform debugging, startup, or normal maintenance, the same project as in the actual HMI is opened on the Sysmac Studio and then an online connection is made. An online connection is made based on the Communications Setup in the project.
Simulator connection	HMI Simulator	The Simulator is used to debug the program offline. The Communications Setup in the project is not used.

Additional Information

Refer to *7-1-5 Offline Debugging with Only the HMI Simulator* on page 7-9 for information on connecting to the Simulator and debugging operations.

6-2 Going Online with an HMI

You can simultaneously go online with more than one HMI in a project from the Sysmac Studio. The operations that are described in this section apply to the currently selected HMI. If there is more than one HMI registered in the project, confirm the HMI to operate before connecting to it.

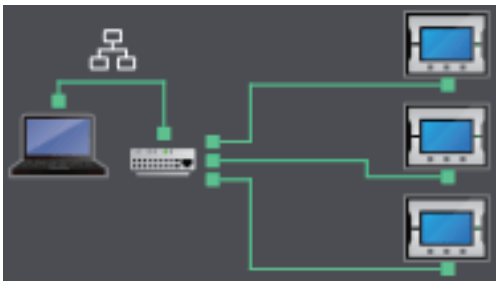
Precautions for Correct Use

Do not reconnect the USB cable for at least 10 seconds after you disconnect it. After you connect the USB cable, do not disconnect it until Windows detects the connection.

It may become impossible for Windows to detect when the cable is connected or disconnected, which would effectively disable the USB port. If that occurs, restart the computer. Otherwise, you will not be able to use the USB device until Windows recovers from the suspended status.

6-2-1 Methods for Going Online with an HMI

To go online with an HMI, there are three connection methods that can be used. These are described in the following table.

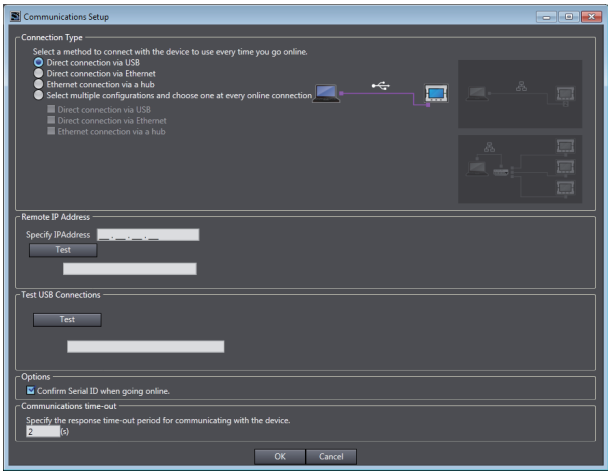
Connection method	Connection diagram	Description
Direct connection via USB		The USB port on the computer is connected directly to the USB Type-B port on the HMI. This is the default connection configuration.
Direct connection via Ethernet		The Ethernet port on the computer is connected directly to Ethernet port 2 on the HMI.
Ethernet connection via hub		The Ethernet port on the computer is connected through the Ethernet network to an Ethernet port on the HMI.

6-2-2 Setting the Connection Method

You must set the connection method, IP address to connect to, and other parameters for communications between the computer and HMI.

1 Select ***Communications Setup*** from the HMI Menu.

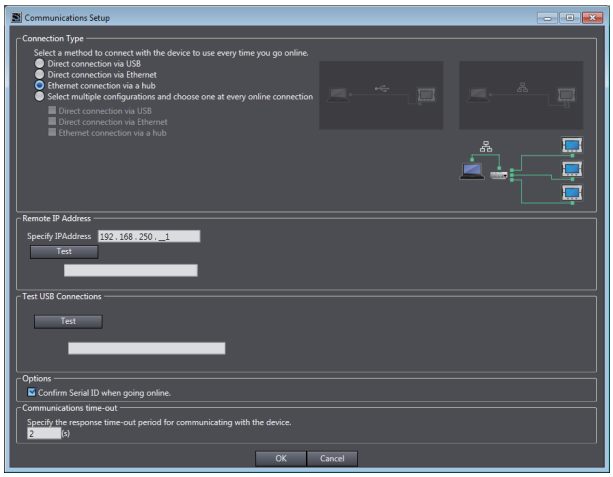
The Communications Setup Dialog Box is displayed.



2 Select the connection method for the connection configuration in the *Connection Type* Field.

For an Ethernet connection via a hub, enter the IP address of the HMI to which you need to connect in the Remote IP Address Area. Select any required options and enter the communications time-out time if required.

Note Refer to *Communications Setup Dialog Box Settings* on page 6-5 on the next page for information on the settings.



3 Click the **OK** Button.

This concludes the settings.

● Communications Setup Dialog Box Settings

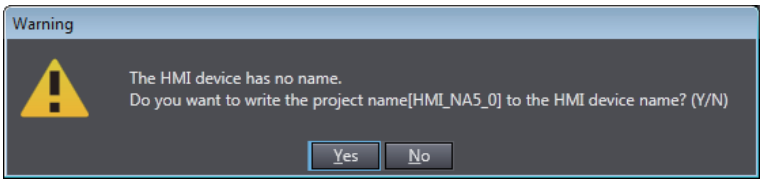
Item	Description	
Connection Type	Select the connection port to use to go online. The selected method is normally used to go online.	
Remote IP Address	If you specify an Ethernet hub connection, set the IP address of the HMI that you will connect to. Also, you can click a test button to perform a communications test.	
Test USB Connections	Click the test button to perform a USB communications test.	
Options	Confirm serial ID when going online.	If you select this option, the names and serial IDs are compared between the project and the HMI when you go online to make sure that a connection is made to the intended HMI.
Communications time-out	You can set the response monitoring time for communications with the HMI. An error is displayed if a response is not received before this time expires. *1. The time can be set to between 1 and 3,600 s.	

6-2-3 Online Connection

Use the following procedure to place the Sysmac Studio online with the HMI.

1 Select ***Online*** from the HMI Menu. Or, click the **Go Online** Button ( ) in the Toolbar.

The following message is displayed the first time you go online. After you write the project name, this message is not displayed.

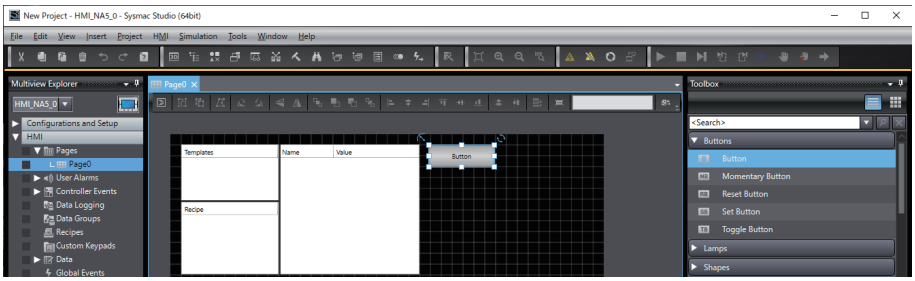


Additional Information

The same message will be displayed the next time you connect if the HMI device name that is set in the project is different from the HMI device name that is set in the HMI.

2 Click the **Yes** Button.

The Sysmac Studio goes online and the color of the bar under the Toolbar changes to yellow.

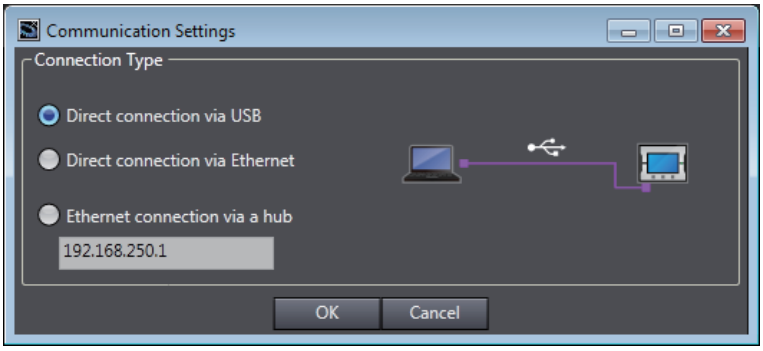


6-2-4 Going Online after Checking the Connection Method

Use the following procedure to go online if you selected the option to select the connection method whenever you connect the computer with the HMI in the Communications Setup Dialog Box.

1 Select ***Online*** from the HMI Menu. Or, click the **Go Online** Button ( ) in the Toolbar.

The Communications Setup Dialog Box is displayed.



The IP address that is set in the Communications Setup Dialog Box is displayed below the *Ethernet connection via a hub* Option.

2 Select the connection method and then click the **OK** Button.

The Sysmac Studio goes online.

6-2-5 Going Offline

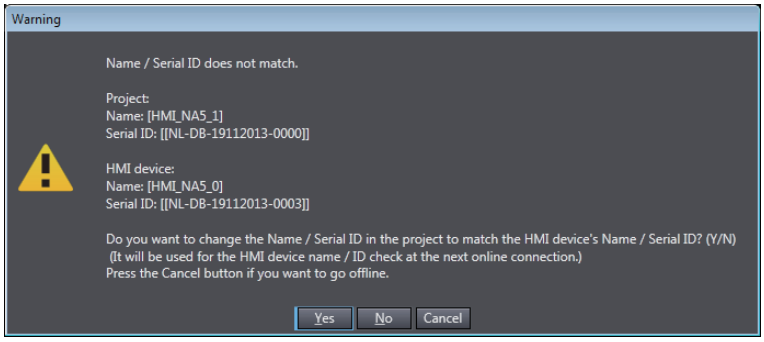
Use the following procedure to go offline.

Select ***Offline*** from the HMI Menu. Or, click the **Go Offline** Button ( ) in the Toolbar.

The Sysmac Studio goes offline.

6-2-6 Confirming Serial IDs

The serial ID is verified when going online if the option to do so was selected in the Communications Setup Dialog Box. If the serial ID of the project on the Sysmac Studio is different from that of the HMI, the following confirmation dialog box is displayed when you attempt to go online.



1 Click the **Yes** Button.

The serial ID of the project on the Sysmac Studio is rewritten to the same value as that of the HMI.

Precautions for Correct Use

- If a USB connection is used, an Ethernet IP address of 192.168.255.xxx is used internally. When using a USB connection, do not use an IP address of 192.168.255.xxx for the Ethernet interface card in the computer.
- Socket ports 9600 and 2224 are used for Ethernet UDP/IP communications. When using Ethernet UDP/IP communications, do not use these ports for any other application.
- When using a direct Ethernet connection and there is more than one Ethernet interface card mounted in the computer, you must select the Ethernet interface card to use. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for specific selection procedures.

7

Debugging

This section describes offline debugging using the HMI Simulator.

7-1 HMI Debugging Functions 7-2

7-1-1 Watch Tab Page 7-3

7-1-2 Breakpoints 7-3

7-1-3 Step Execution 7-5

7-1-4 Simulator Functions 7-8

7-1-5 Offline Debugging with Only the HMI Simulator 7-9

7-1-6 Offline Debugging with the Controller Simulator 7-10

7-1 HMI Debugging Functions

The operations you can use for debugging on the Sysmac Studio are listed below. The HMI Simulator is used for offline debugging. You can use the HMI Simulator by itself, or you can connect it to the Controller Simulator to debug the entire system.

Caution

Although the Simulator simulates the operation of the HMI, there are differences from the HMI in operation and timing. After you debug operation with the Simulator, always check operation on the actual Controller and HMI before you use them in the actual system. Accidents may occur if the controlled system performs unexpected operation. Refer to *A-7 Differences between the Physical HMI and Simulator* on page A-15 for details on differences in operation.	!
Although offline debugging simulates the integrated operation of the HMI and Controller, there are differences in operation and timing in comparison with combining the actual HMI and Controller. After you debug operation with the simulation, always check operation on the actual Controller and HMI before you use them in the actual system. Accidents may occur if the controlled system performs unexpected operation. Refer to *A-7 Differences between the Physical HMI and Simulator* on page A-15 for details on differences in operation.	!
To run the simulator, administrator rights are required.	!

The following three functions are provided for debugging during HMI simulations. These functions are the same regardless of whether they are used in offline debugging.

- Watch Tab Page
- Breakpoints
- Step execution

Additional Information

The following functions are disabled while the HMI simulator is running.

Changes made while the simulator is running will not be applied until the simulator is turned off.

- Build
- Online and functions that require online (synchronization, etc.)
- Editing subroutines
- Editing devices
- Editing variables and data types

7-1-1 Watch Tab Page

The basic Sysmac Studio operations of the Watch Tab Page apply to HMIs.

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

However, the following differences exist for HMI projects.

- You can monitor the following variables on the Watch (Project) Tab Page.

  Standard projects: Only global variables can be registered. You can continuously monitor any of the global variables. Register the variable names as they are.

  IAG projects: You cannot monitor variables in the IAG project.

- The Watch Tab Page functions only for the Simulator. Even if you are online with the physical HMI, you cannot monitor the variables in the physical HMI.

7-1-2 Breakpoints

You can set breakpoints in the source code of a subroutine. You can use breakpoints to pause the execution of a subroutine at any desired point.

Setting and Clearing Breakpoints

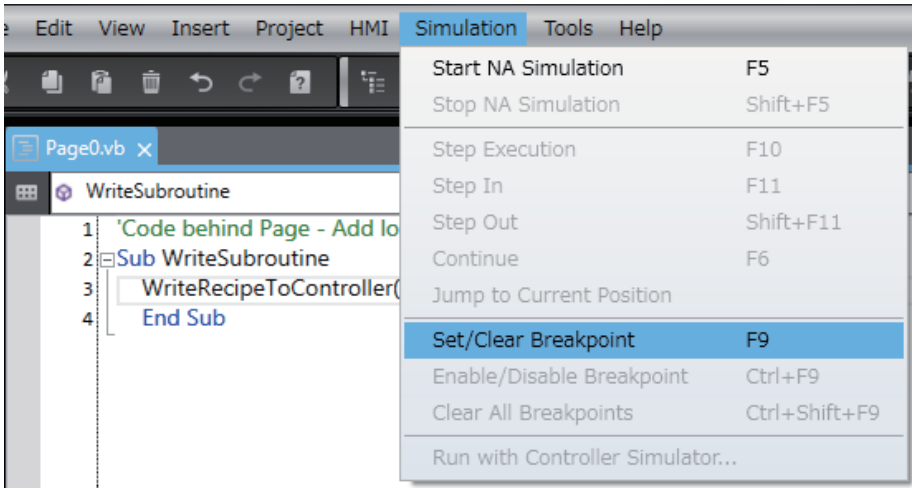
● Setting Breakpoints

You can set breakpoints to pause execution of a simulation, e.g., to see the status after a subroutine is executed.

● Procedure

To set a breakpoint, move the cursor to the line in the subroutine where you want to set the breakpoint and select ***Set/Clear Breakpoint*** from the Simulation Menu.

If you repeat this operation, the breakpoint will be cleared.



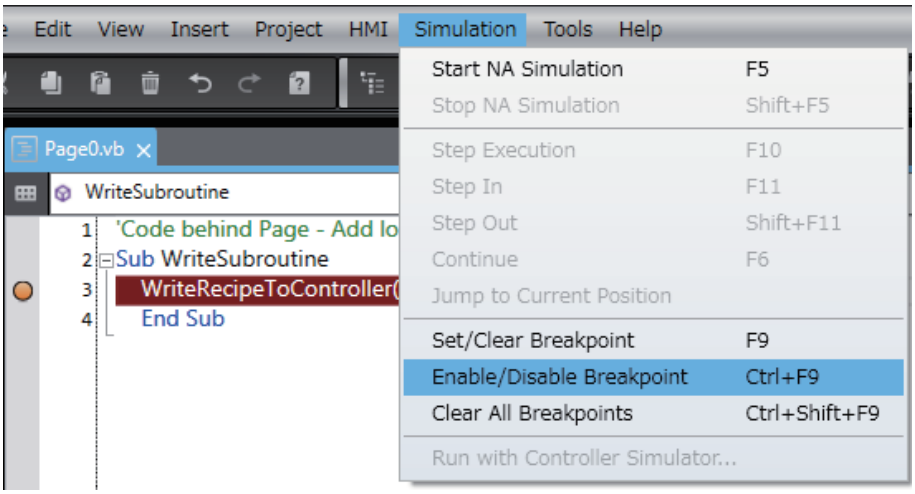
Enabling and Disabling Breakpoints

Disabling a Breakpoint

If you do not need a breakpoint temporarily during debugging, you can disable the breakpoint but still leave it in the program.

Procedure for Disabling Breakpoints

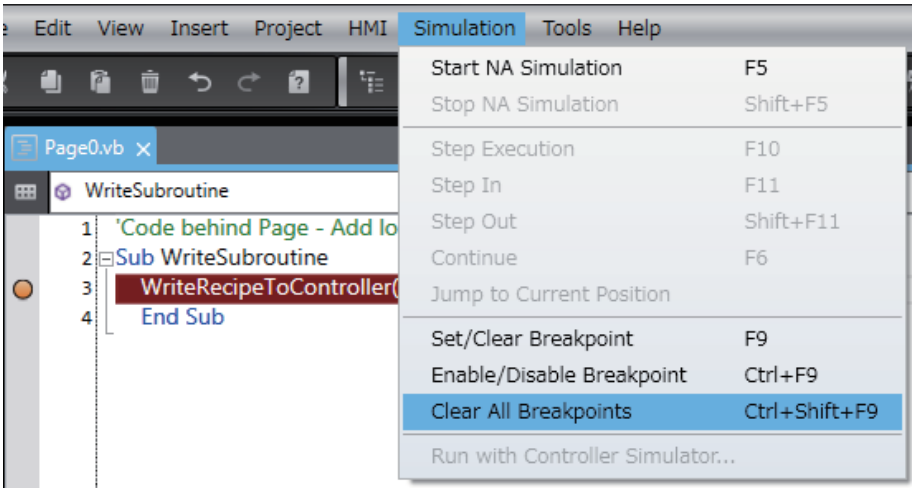
To disable a breakpoint, move the cursor to the line of the subroutine that has the breakpoint that you want to disable, and select ***Enable/Disable Breakpoint*** from the Simulation menu. If you perform the same steps for a breakpoint that is disabled, the breakpoint will be enabled.



Clearing All Breakpoints

You can clear all of the breakpoints at the same time.

Select ***Clear All Breakpoints*** from the Simulation Menu to clear all of the breakpoints.



7-1-3 Step Execution

You can use step execution when subroutine execution is paused at a breakpoint to trace operation one line of the source code at a time.

Step Execution

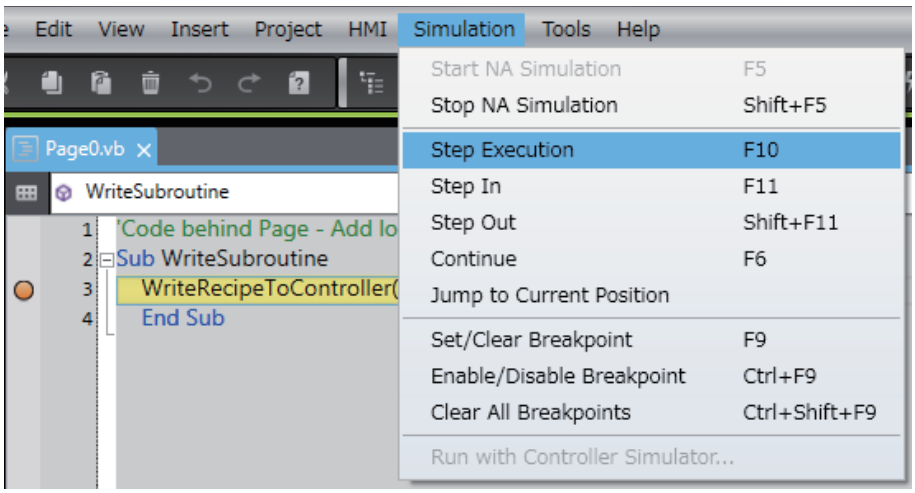
You can use step execution to execute the source code of a subroutine one line at a time.

If you select ***Step In*** and other commands from the Simulation Menu while paused at a breakpoint during a simulation, you can control execution one line or one function at a time.

Step Execution

The ***Step Execution*** command executes one function at a time.

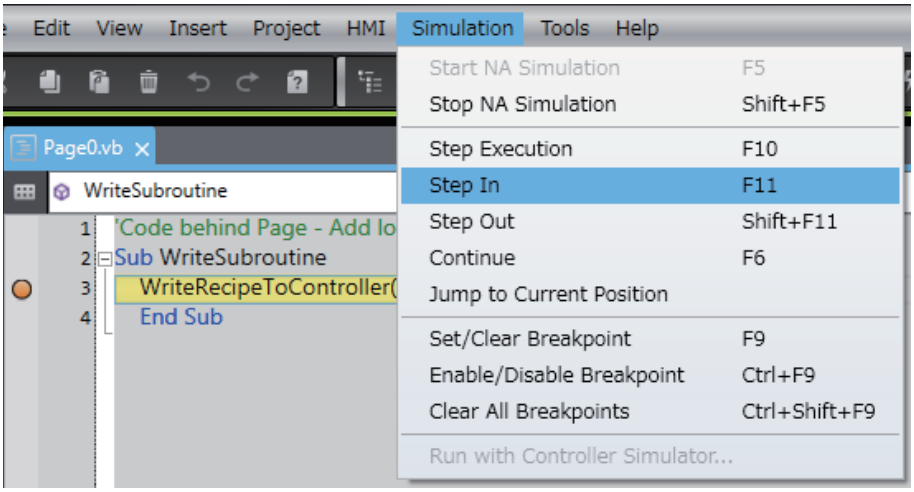
Select ***Step Execution*** from the Simulation Menu when subroutine execution is paused during step-in execution.



Step-in Execution

Step-in execution performs step execution for the source code of a subroutine.

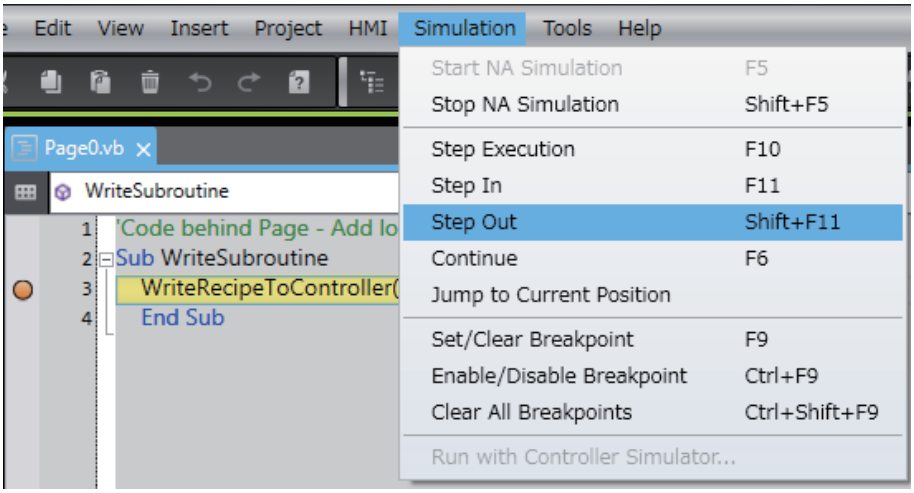
When you execute a program through the Simulator, select ***Simulation - Step In*** when the subroutine is stopped at a breakpoint, paused, or stopped during step execution.



Step-out Execution

Use the ***Step Out*** command to exit a subroutine during step-in execution.

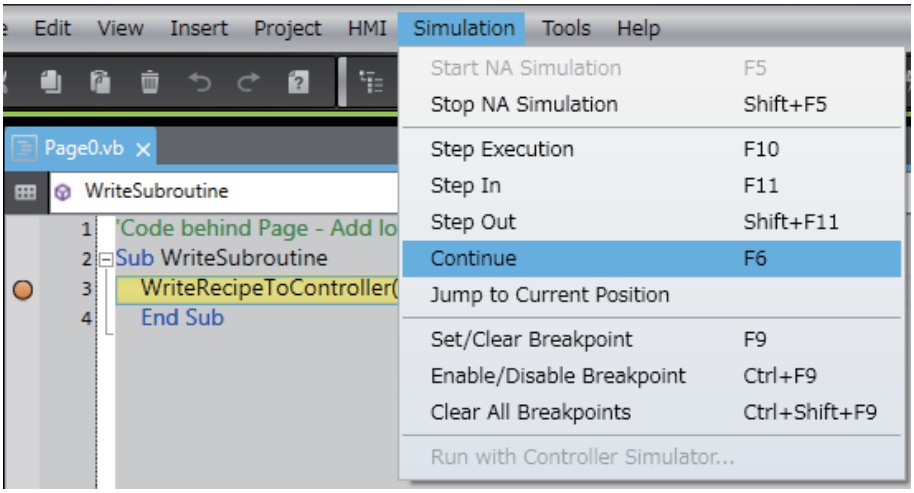
Select ***Step Out*** from the Simulation Menu during step-in execution.



Continue

The ***Continue*** command executes the subroutine to the next breakpoint.

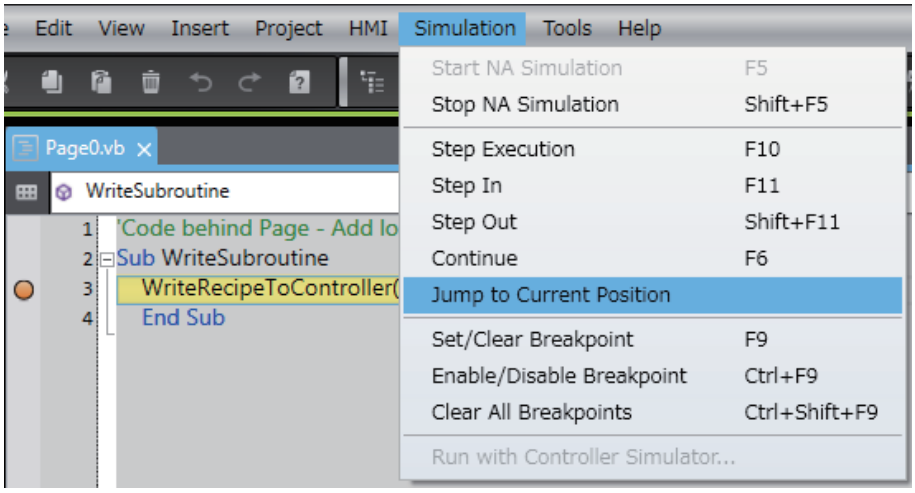
Select ***Continue*** from the Simulation Menu when subroutine execution is paused during step-in execution.



Jump to Current Position

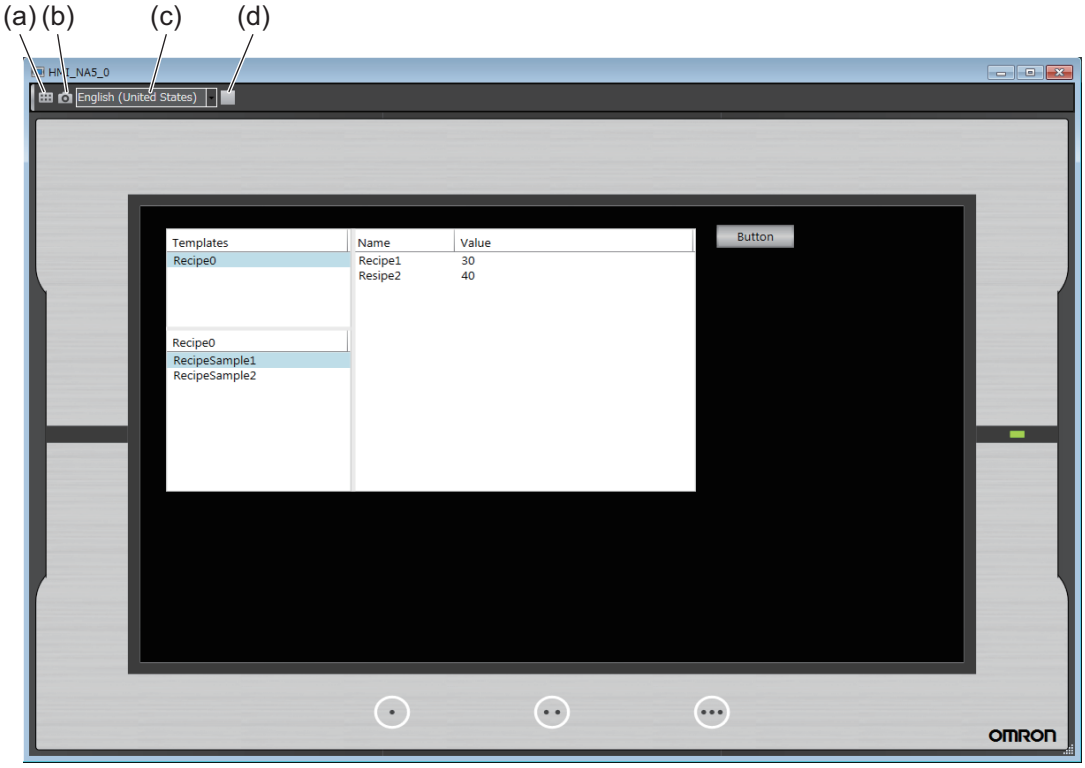
Use this command to jump to the breakpoint at which the program is currently paused in the code editor.

Select ***Jump to Current Position*** from the Simulation menu when subroutine execution is paused during step-in execution.



7-1-4 Simulator Functions

You can use functions such as page changing in Simulator if the Runtime version is 1.08 or higher.



Symbol	Item	Description
(a)		You can select a page to be displayed in Simulator.
(b)		You can output the currently displayed page to the clipboard.
(c)	Language Selection	You can select the project language to be displayed.
(d)		Exit Simulator

7-1-5 Offline Debugging with Only the HMI Simulator

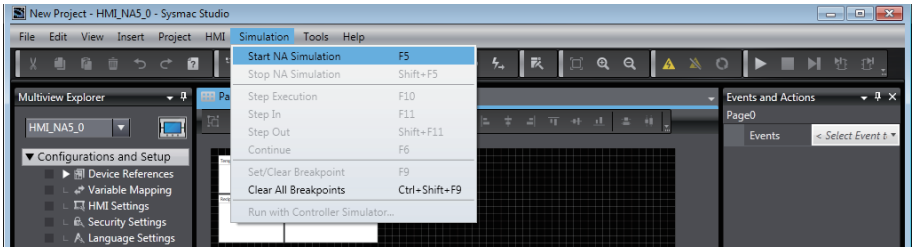
Before you check the entire system, you can perform debugging with the HMI Simulator alone.

When you create a project and want to debug it, always check operation with only the HMI Simulator first. The Sysmac Studio comes with a Simulator that simulates HMI functions.

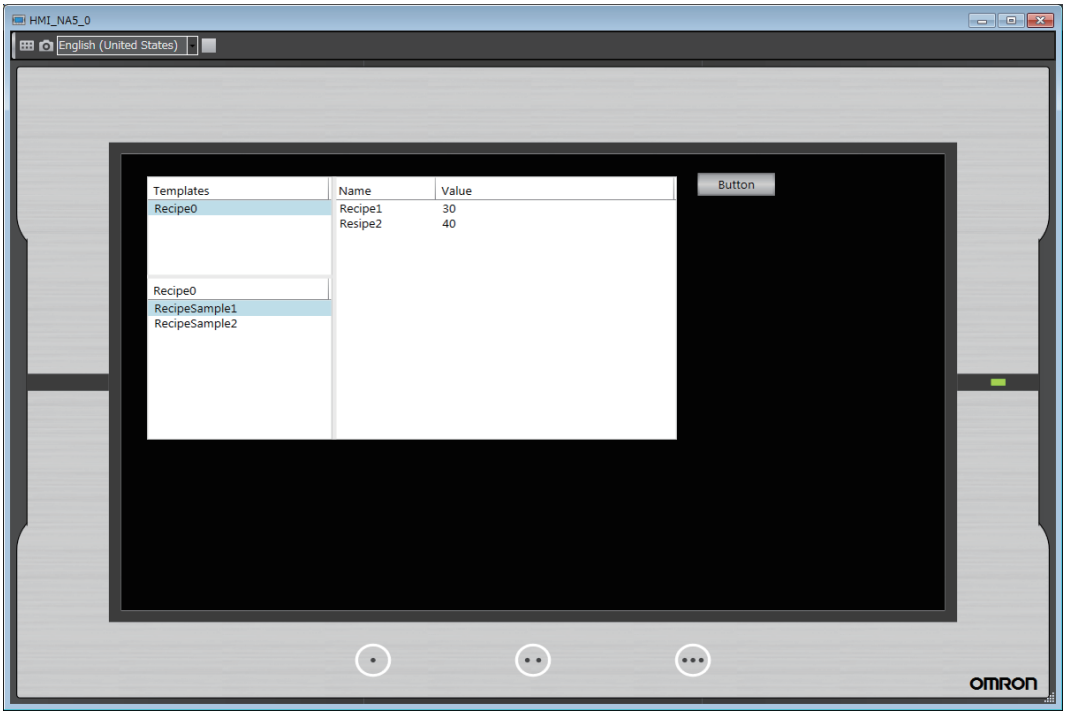
Procedure

Use the following procedure to start the Sysmac Studio and perform debugging.

1. Start the Sysmac Studio and create a project.
2. Create the HMI application.
3. Select ***Build HMI*** from the Project Menu to build the project.
4. Select ***Start NA Simulation*** from the Simulation Menu.



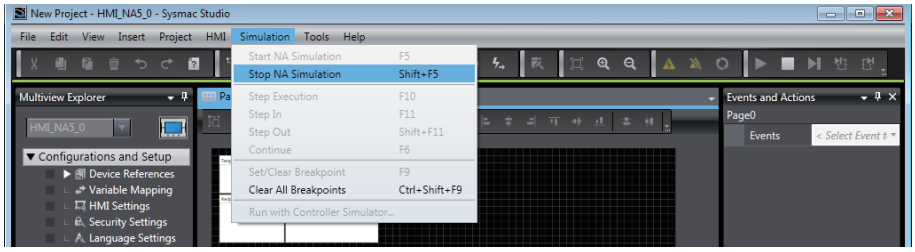
The Simulator starts.



Additional Information

If the project is not yet built, it is built automatically.

5 To stop the simulation, click ***Stop NA Simulation*** from the Simulation Menu. The HMI Simulator is ended.



7-1-6 Offline Debugging with the Controller Simulator

You can perform debugging with the HMI Simulator connected online to the Controller Simulator. This allows you to debug the project while the Controller program is actually running.

After you complete checking operation with the HMI Simulator alone, check operation that includes the Controller program. The Sysmac Studio enables integrated debugging of the Controller program and the HMI application by connecting the Controller Simulator to the HMI Simulator.

Procedure

Use the following procedure to start the Sysmac Studio and perform debugging.

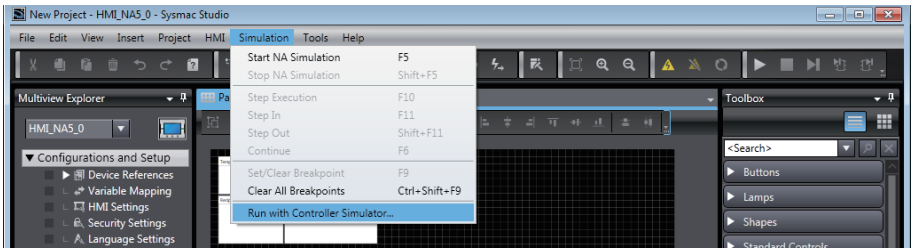
1 Start the Sysmac Studio and create a project.

2 Create the Controller program.

3 Create the HMI application.

4 Select ***Build HMI*** from the Project Menu to build the project.

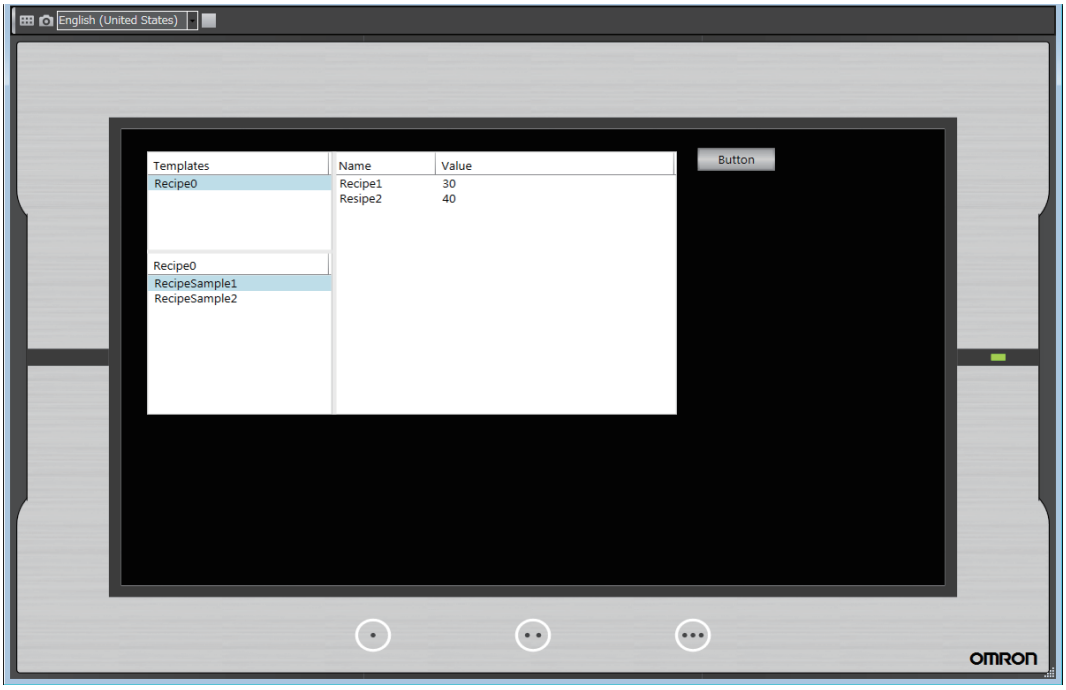
5 Select ***Run with Controller Simulator*** from the Simulation Menu.



Additional Information

- If there is more than one Controller in the project, the window will be displayed for the Controller that is currently connected. If you select a Controller for which variables are not mapped, an error will occur and starting the Simulator will be canceled.
  This procedure does not support offline debugging with the NX-series Safety Network Controller.
- When the simulator is started under the **Run with Controller Simulator** setting, E_COM_002 may occur. This is because the controller simulator is always operated with the latest unit version, because of which it may differ from the definition of the system variables of the set unit version. This problem can be avoided by changing the set unit version of the controller to the latest version.

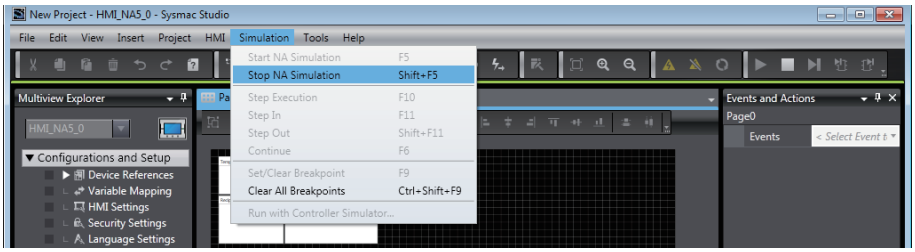
The Controller Simulator is started and the HMI Simulator is displayed.



Additional Information

If the project is not yet built, the entire project including the Controller is built automatically. If a building error occurs for the Controller, a dialog box is displayed to notify you of the error. Change the device to the relevant Controller, build the project, and check the error.

6 To stop the simulation, click ***Stop NA Simulation*** from the Simulation Menu. The Controller Simulator and the HMI Simulator are stopped.



Additional Information

You can also start online debugging when the Controller is selected in the project. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for details.

8

Transfer

This section describes how to transfer the projects between the HMI and the Sysmac Studio.

8-1 Transfer 8-2

8-2 Download Function 8-3

8-3 Upload Function 8-4

8-4 Relinking Internal Devices 8-5

8-5 Synchronization 8-8

8-5-1 Downloading While Online 8-12

8-5-2 Using Storage Media for Downloading 8-13

8-5-3 Uploading Projects Online 8-16

8-5-4 Uploading with Storage Media 8-18

8-1 Transfer

The download function and upload function are used to transfer projects on the Sysmac Studio and the HMI. Use synchronization to check the differences in projects before transfer.

⚠ WARNING

Sufficiently check the operation of any project that you create before you start actual system operation.

Precautions for Safe Use

Unexpected operation may result if you set inappropriate network configuration settings. Even if appropriate network configuration settings are set, confirm that the controlled system will not be adversely affected before you transfer the data.

Precautions for Correct Use

Do not perform any other operations on the Sysmac Studio while the download function, upload function, or synchronization is being executed. An error will occur and synchronization will fail.

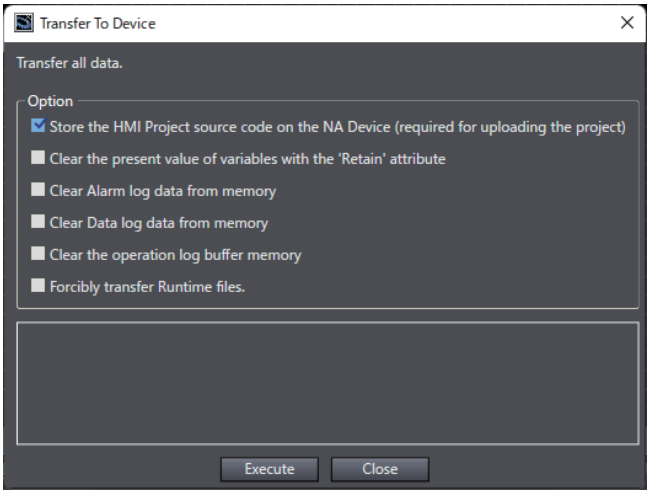
8-2 Download Function

The download function is a specialized downloading function that makes downloading easier.

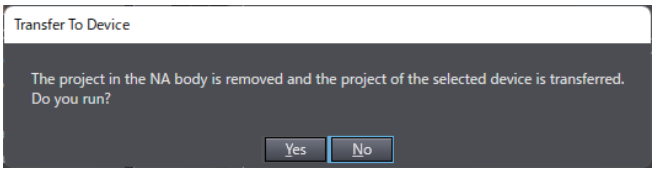
Procedure

1 Click on the Toolbar. Or, select **HMI - Transfer - Transfer To Device**.

2 Select the options as required and click **Execute**.



3 The following confirmation message is displayed. Click **Yes**.



Additional Information

The download function differs from online downloading in the following ways:

- Cannot be run when online.
- System program updates are automatically executed, followed by execution of the project download.
- The Synchronization Window is not displayed.
- When a device name is not set to HMI, the device name set in the project is unconditionally set.
- Although the following options are forcibly executed in accordance with conditions, they are not reflected in the check boxes of the dialog box.
  - Clear the present values of variables with Retain attribute
  - Clear alarm log data from memory
  - Clear data log data from memory
  - Clear the operation log buffer memory

When downloading to NA5-□□W□□□□-V2, system program updates are not supported under a 32-bit Windows environment. When downloading, first perform a system recovery and update the system program to an appropriate version. For details on system recovery, refer to *4-2 System Recovery in NA-series Programmable Terminal Hardware(-V2) User's Manual* (V128).

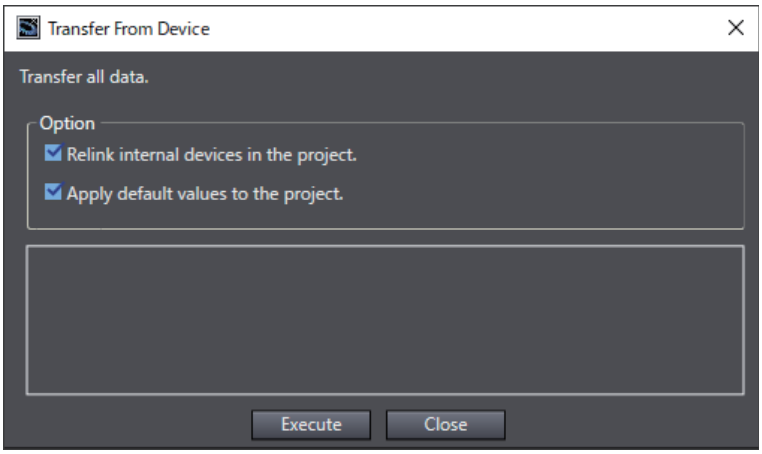
8-3 Upload Function

The upload function is a specialized uploading function that makes uploading easier.

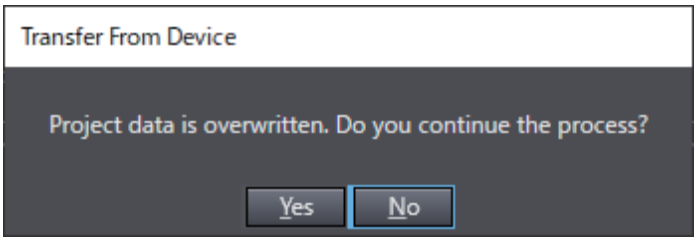
Procedure

1 Click on the Toolbar. Or, select ***Transfer*** – ***Transfer From Device*** from the HMI Menu.

2 Select the options and click **Execute**.



3 The following confirmation message for overwriting is displayed. Click the **Yes** Button.



Additional Information

The upload function differs from online uploading in the following ways:

- Cannot be run when online.
- System program updating is not run.
- Building is not performed before the upload.
- Uploading is run even if errors remain.
- The Synchronization Window is not displayed. Therefore, the upload always applies to all upload items.

8-4 Relinking Internal Devices

When a project is uploaded, the internal devices will not be linked and there will be restrictions on operations related to the Controller, such as variable mapping. This section describes how to relink a Controller in the project.

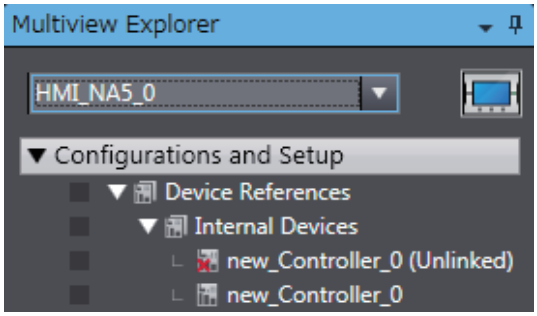
Relinking while Uploading

If you select the *Relinking to Internal Devices* Check Box and upload the project, the internal devices in the project that was uploaded are automatically relinked to the Controller of the same name. If there is no Controller of the same name in the project, the Controller is registered as an unlinked Controller.

During relinking, the variables that are registered in the Controller in the project are compared with the variables that are not registered in the Controller in the project. If no errors are detected, relinking is completed.

Relinking after Uploading

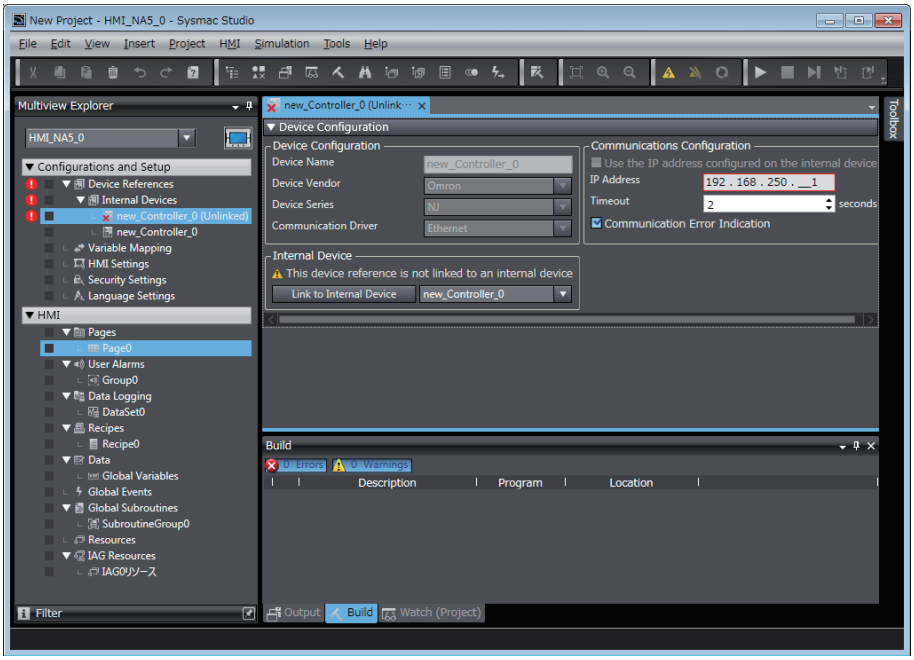
If you do not select the *Relinking to Internal Devices* Check Box and upload the project, the internal devices in the project that was uploaded will not be linked to the Controller's project.



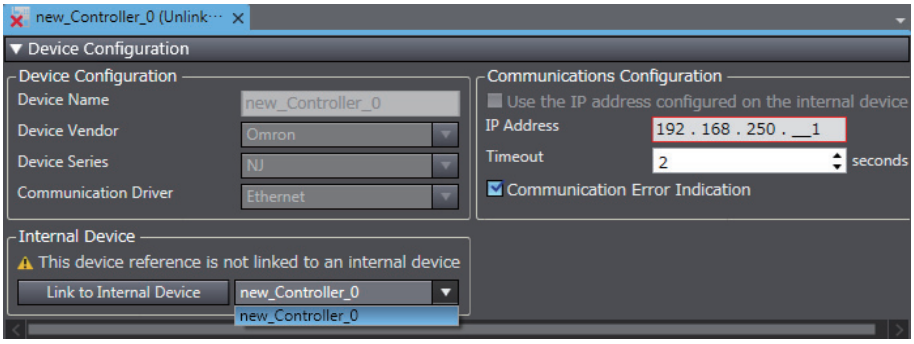
Use the following procedure to link the unlinked Controller with the Controller in the project.

1 Place the Sysmac Studio offline from the HMI. Refer to *Section 6 Connecting to the HMI* for details.

2 Open **Device References** under **Configurations and Setup** in the Multiview Explorer and double-click the unlinked Controller.



3 Select the Controller to link in the *Internal Devices* Area and click the **Link to Internal Device** Button.

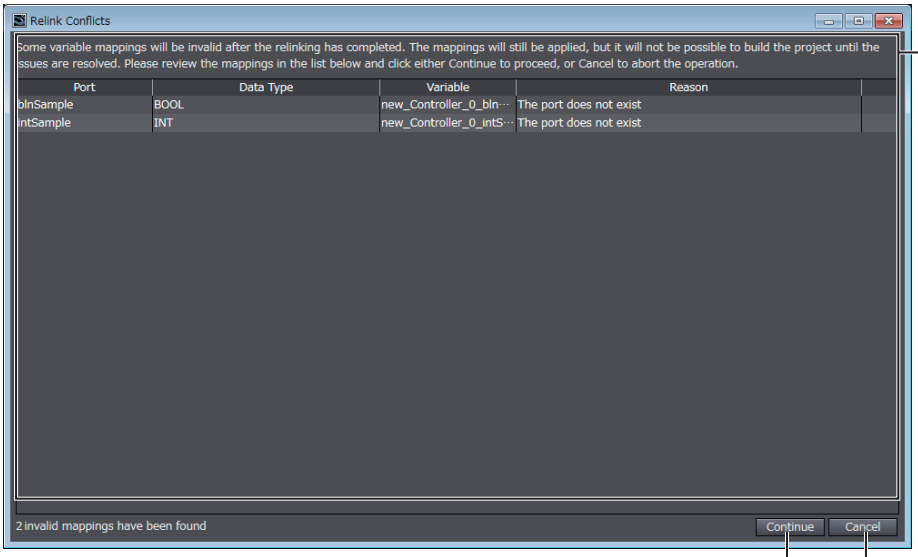


4 The variables that are registered to the specified Controller are compared with the variables that are registered to the unlinked Controller. If no errors are detected, relinking is completed.

Additional Information

If any variables that are registered to the destination Controller do not agree with the variables that are registered to the unlinked Controller when relinking is executed, the following dialog box is displayed.

Modify the global variables in the Controller and in the HMI according to the information provided in the dialog box.



(a)

(c) (b)

Letter	Item	Description	Remarks
(a)	Conflict list	Lists all of the conflicts.	
(b)	Cancel	Cancels relinking.	
(c)	Continue	Leaves the conflicts and completes relinking.	

8-5 Synchronization

The basic Sysmac Studio operations for synchronization generally apply to HMIs. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for details.

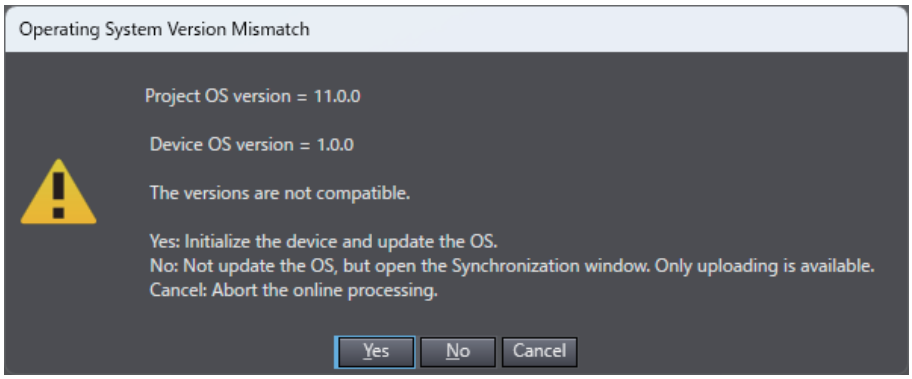
Functional Differences When an HMI Is Selected

The following differences exist in synchronization when an HMI is selected.

- Detailed comparison is not supported.
- Algorithm checking is not supported. Therefore, there is no way to prevent mismatches in build dates that result from rebuilding.
- If the project version specified in the project is different from the version of the project in the HMI, the runtime files will always be transferred.
- The following synchronization options are provided.

Option	Default	When transfer is enabled	Description
Transfer source code to the NA device.	Selected.	Values written to HMI	Select this check box to transfer the source code.
Relink internal devices in the project (valid for Transfer From Device)	Selected.	Values written from HMI	If this check box is selected, global variables are automatically mapped to the Controller that is registered in the project when the project is uploaded.
Clear the present values of variables with Retain attribute.	Not selected.	Values written to HMI	If this check box is selected, the present values of Retain variables are cleared after the data is downloaded. If you have changed variable-related settings, you cannot clear the selection of this check box.
Clear alarm log data from memory	Selected.	Values written to HMI	If this check box is selected, the alarm log in non-volatile memory is cleared. If you have changed the user alarm settings, you cannot clear the selection of this check box.
Clear Data log data from memory	Selected.	Values written to HMI	If this check box is selected, the data log data in non-volatile memory is cleared. If you have changed the data logging settings, you cannot clear the selection of this check box.
Clear the operation log buffer memory.	Selected.	Values written to HMI	If this check box is selected, the operation log in non-volatile memory is cleared. If you have changed the operation logging settings, you cannot clear the selection of this check box.
Apply default values to the project. (Upload)	Selected.	Values written from HMI	Overwrite the default values of the NA unit project with the default values of the currently open project when the project is uploaded.
Forcibly transfer Runtime files. (Downloading)	Not selected.	Values written to HMI	Downloads the Runtime files forcibly.

- The following dialog box is displayed when you connect to the HMI if the version of the system program in the HMI is not compatible with the system program in the Sysmac Studio. You must update the system program before you can perform synchronization. Update the system program according to the displayed instructions.



Additional Information

System program 1.0.0 is installed in the NA unit in the factory default state. The system program will always be updated when synchronization is performed in this state.

Precautions for Correct Use

- Projects that are downloaded with Sysmac Studio version 1.10 cannot be uploaded.
- Projects that are downloaded when the *Transfer source code to the NA device* Check Box is not selected cannot be uploaded.
- If a project with a different system version from the HMI is downloaded, the values of Retain variables, the user alarm log, and the data log buffer memory will be initialized.
- Changing the settings with the HMI may result in a discrepancy at the comparison result, even if the settings are the same.
- When version 1.10 or earlier of IAG is used in a project with Runtime Ver. 1.11 or later, the page in which IAG is used may be different. When using Runtime Ver. 1.11 or later, use version 1.11 or later of IAG in order to eliminate the differences.
- When using the Safety Monitor, perform safety validation on the safety controller before executing synchronization.
- For a project downloaded to the NA unit with Sysmac Studio Ver. 1.31 or earlier, differences may occur if synchronization is performed using Sysmac Studio Ver. 1.40 or later. This is caused by a difference in the internal data structure and thus there is no problem on the project itself. To eliminate the differences, download with Sysmac Studio Ver. 1.40 or later.
- When a custom keypad is used, the language setting and custom keypad may become inconsistent.
- If using Sysmac Studio Ver. 1.40 or earlier, VNC server is set to use, and a password is not set, differences may occur if synchronization is performed using Sysmac Studio Ver. 1.41 or later. This is caused by a difference in the internal data structure and thus there is no problem on the project itself. To eliminate the differences, download with Sysmac Studio Ver. 1.41 or later.
- For a project downloaded with Sysmac Studio (32-bit edition), differences may occur if synchronization is performed using Sysmac Studio (64-bit edition). (Same goes with the opposite case.) This is caused by a difference of the values lower than the significant digits of internal data. This difference does not affect the project operation, since this is only a difference in the internal data structure.
- Differences may occur if a download is executed after a subroutine group is deleted from a global subroutine, and synchronization is executed again without closing the project. When a subroutine group has been deleted, first close the project before executing synchronization.
- When connecting to NA5-□□W□□□□-V2 online, system program updates are not supported under a 32-bit Windows environment. When downloading, first perform a system recovery and update the system program to an appropriate version. For details on system recovery, refer to *4-2 System Recovery in NA-series Programmable Terminal Hardware(-V2) User's Manual* (V128).

Differences in Comparison Results

The following differences apply to the comparison results.

Verification Results Example

The verification results are displayed as shown in the following example.

Column	Item
Source	Project name on the Sysmac Studio
Source Modified Date	The last time that the project was built on the Sysmac Studio
Target Modified Date	The last time that the HMI project was built on the Sysmac Studio
Target	HMI project name
Detailed Comparison	This column is not used when an HMI is selected.

Verification Units

The units for comparison that are shown in the Synchronization Window are listed in the following table.

Synchronization data name	Level	Qty	Detailed comparison	Remarks
HMI name	1	1	None	
Project	2	1	None	
Pages	3	1	None	
Page*	4	N	None	
Subroutines	3	1	None	
SubroutineGroup*	4	N	None	
Variables	3	1	None	
Global Events	3	1	None	
Alarms	3	1	None	
Data Groups	3	1	None	
Data logging	3	1	None	
Recipe Template	3	1	None	
Project Security	3	1	None	
Troubleshooter	3	1	None	
Languages	3	1	None	
*	4	N	None	
Operation Log Settings	3	1	None	
Settings	2	1	None	
HMI Settings	3	1	None	
User Data	2	1	None	
User Accounts	3	1	None	
Recipe Instances	3	N	None	
Resource Files	2	1	None	
Documents	3	1	None	
.	4	N	None	
Images	3	1	None	
Videos	3	1	None	
.	4	N	None	
Safety Data	2	1	None	
*	3	N	None	

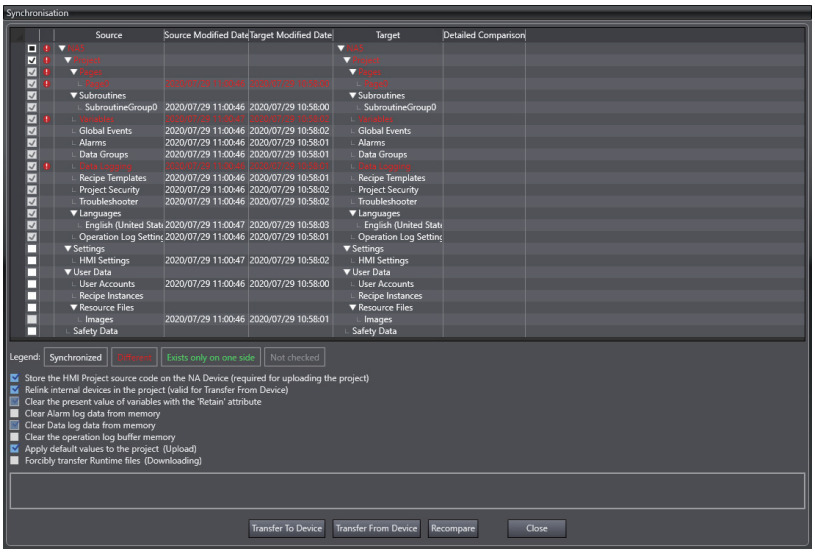
8-5-1 Downloading While Online

You can use the following procedure to download the project while the Sysmac Studio is online.

1 Place the Sysmac Studio online with the HMI. Refer to *Section 6 Connecting to the HMI* for details.

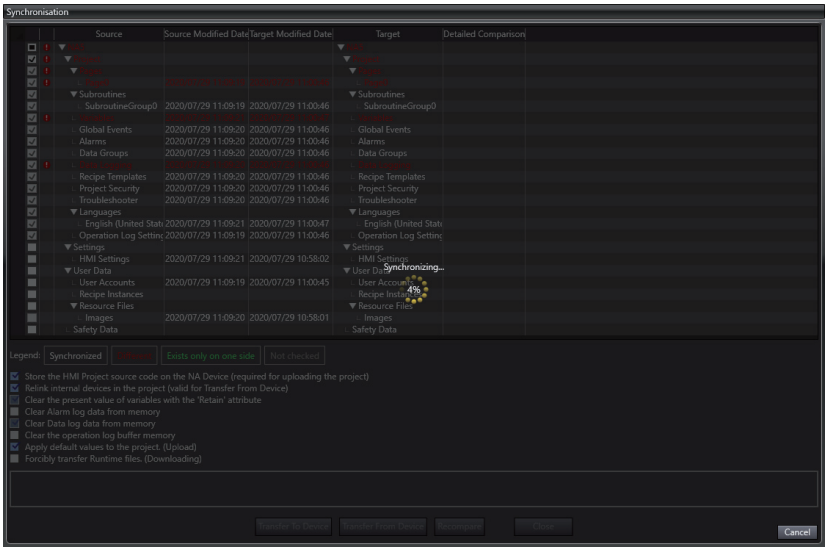
2 Click on the Toolbar. Or, select ***Synchronization*** – ***NA Device*** from the HMI Menu.

3 The project on the Sysmac Studio is compared with the project in the HMI and the Synchronization Window is displayed.



4 Select the items to download and select the options as required.

Click the **Transfer to Device** Button. The data is downloaded and the HMI is restarted.



Precautions for Correct Use

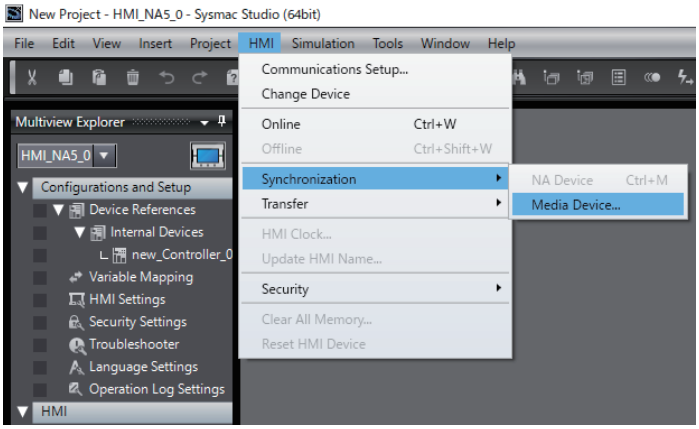
If you download a project simultaneously with PUT processing of a file from a client device using the FTP server, the FTP-transferred file may be lost or downloading may fail.

8-5-2 Using Storage Media for Downloading

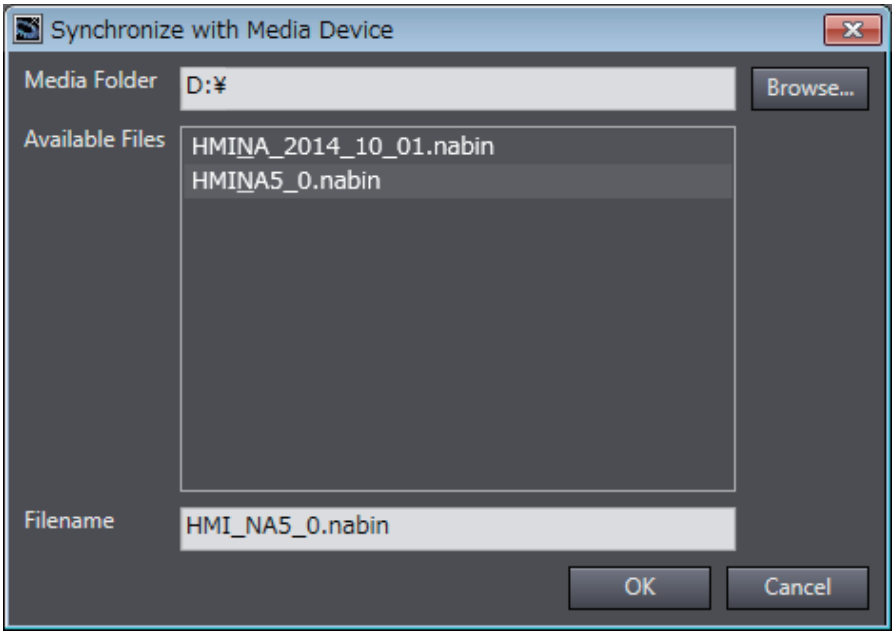
With an HMI, you can perform product downloads with storage media.

1 Insert an SD Memory Card or USB memory device to use for the download into the computer.

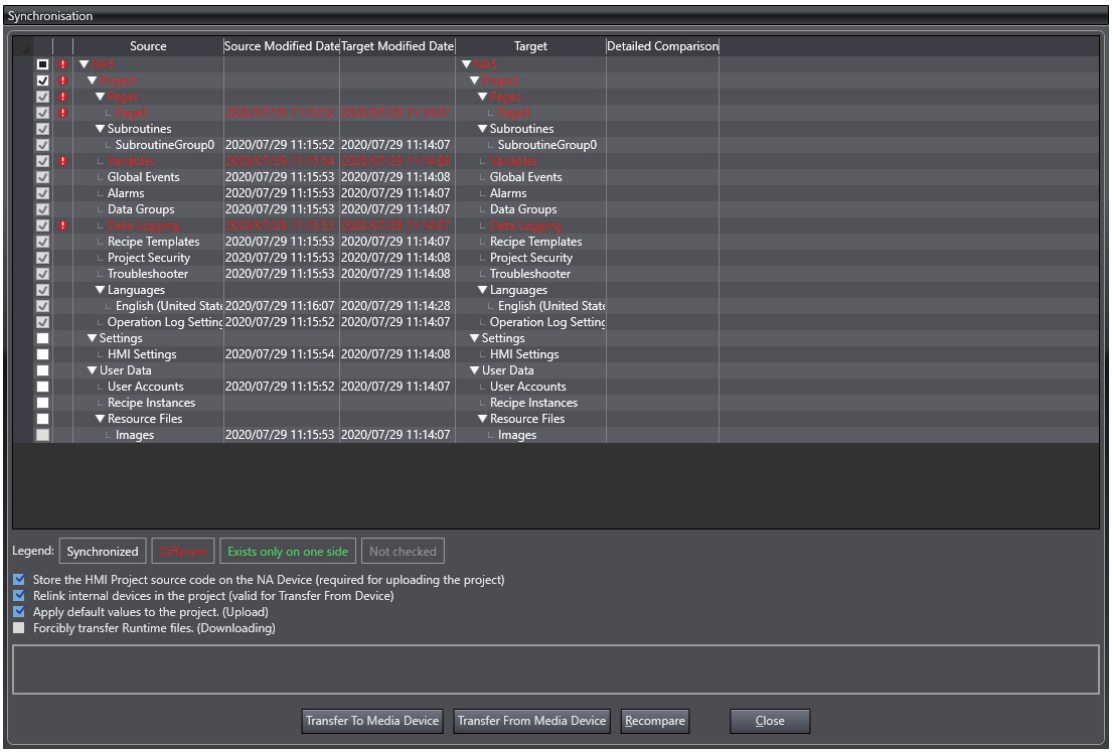
2 Select ***Synchronization*** – ***Media Device*** from the HMI Menu.



3 Select the file to use for the download and click the **OK** Button.

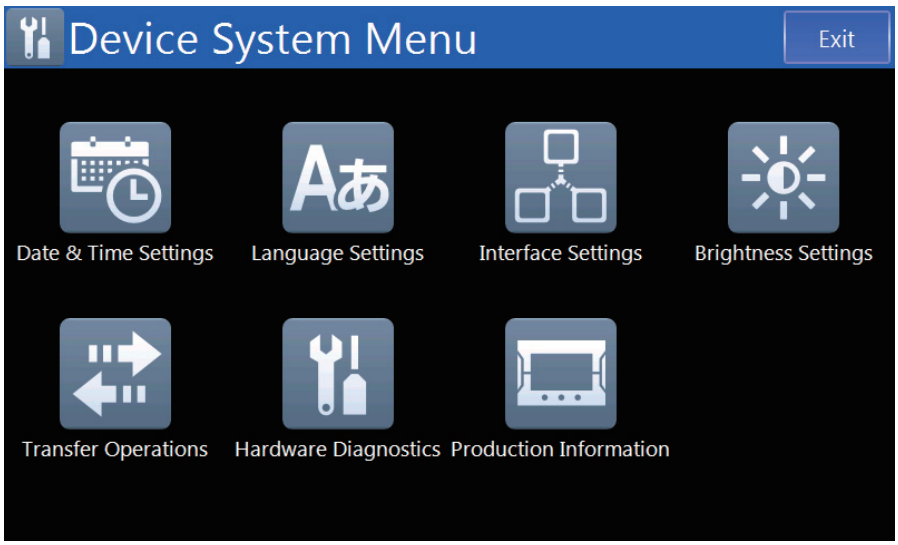


4 The project on the Sysmac Studio is compared with the project in the storage media and the Synchronization Window is displayed.

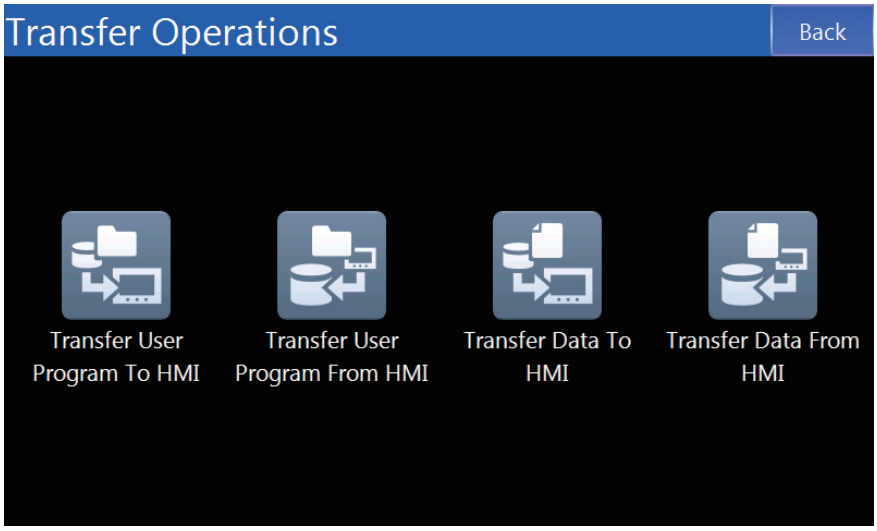


5 Click the **Transfer to Media Device** Button to transfer the project to the storage media.

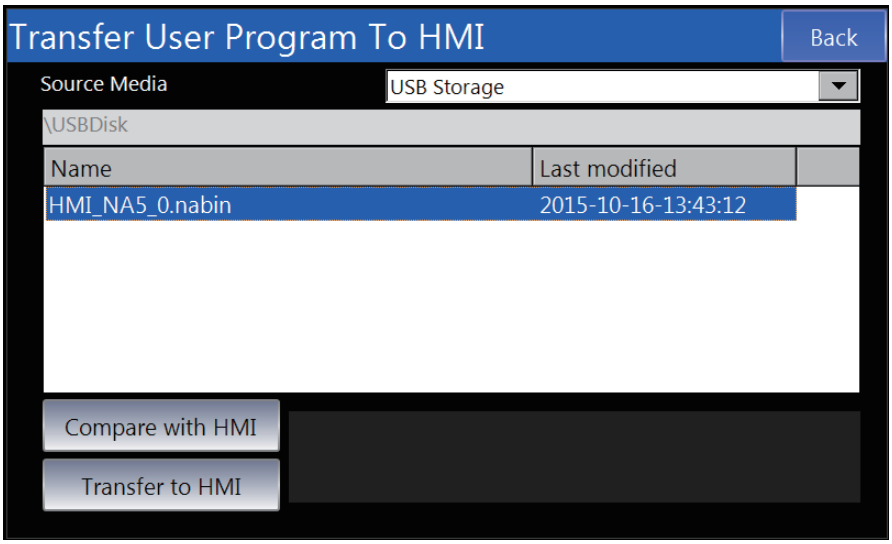
6 Insert the storage media into the HMI, display the Device System Menu, and touch the **Transfer Operations** Button.



7 Touch the **Transfer User Program To HMI** Button.



8 Select the project to transfer and touch the **Transfer to HMI** Button to download the selected project to the HMI.



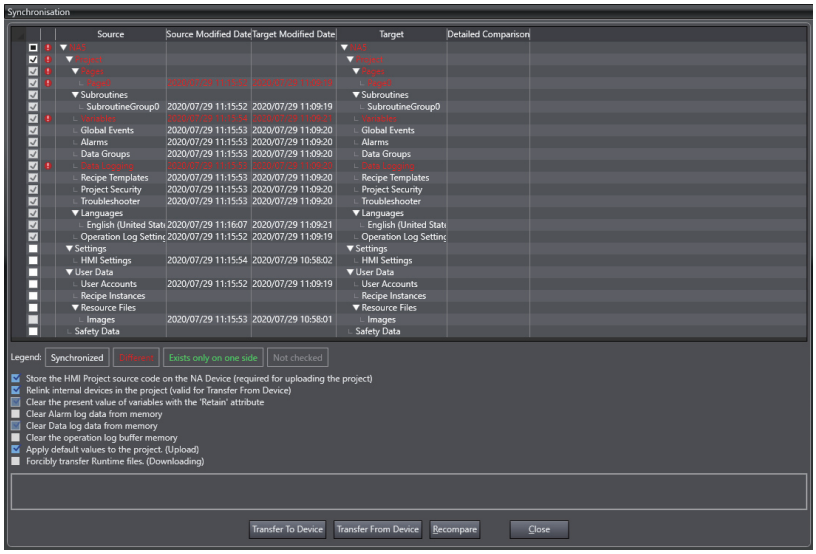
8-5-3 Uploading Projects Online

You can use the following procedure to upload a project while the Sysmac Studio is online.

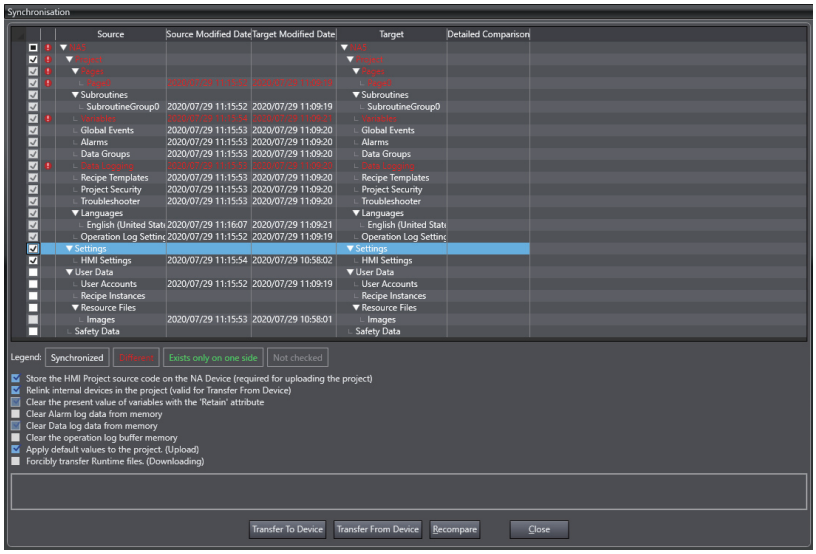
1 Place the Sysmac Studio online with the HMI. Refer to *Section 6 Connecting to the HMI* for details.

2 Click the (Synchronize) Button on the Toolbar. Or, select ***Synchronization*** – ***NA Device*** from the HMI Menu.

3 The project on the Sysmac Studio is compared with the project in the HMI and the Synchronization Window is displayed.



4 Select the items to upload and select the options as required.

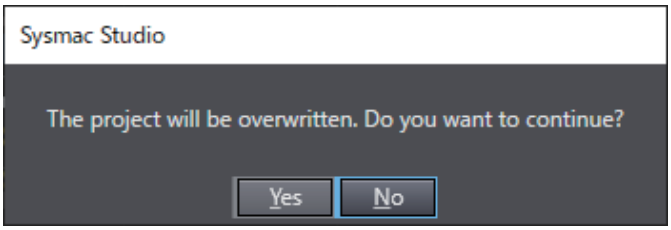


Precautions for Correct Use

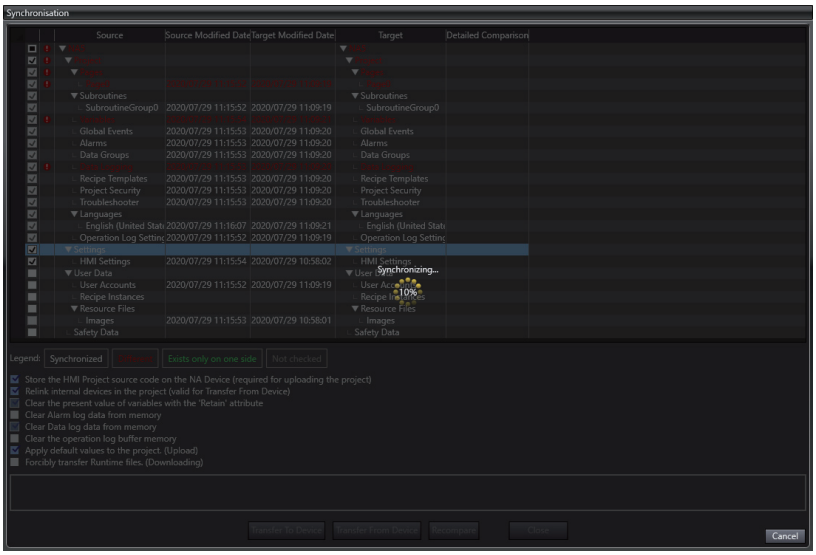
When a project is uploaded, the internal devices are not linked. In this condition, you will not be able to perform offline debugging with the Controller Simulator, or variable mapping.

Refer to *8-4 Relinking Internal Devices* on page 8-5 for information on how to relink internal devices.

5 Click the **Transfer From Device** Button. The following confirmation message for overwriting is displayed. Click the **Yes** Button.



The project is uploaded and the changes are applied to the open project.

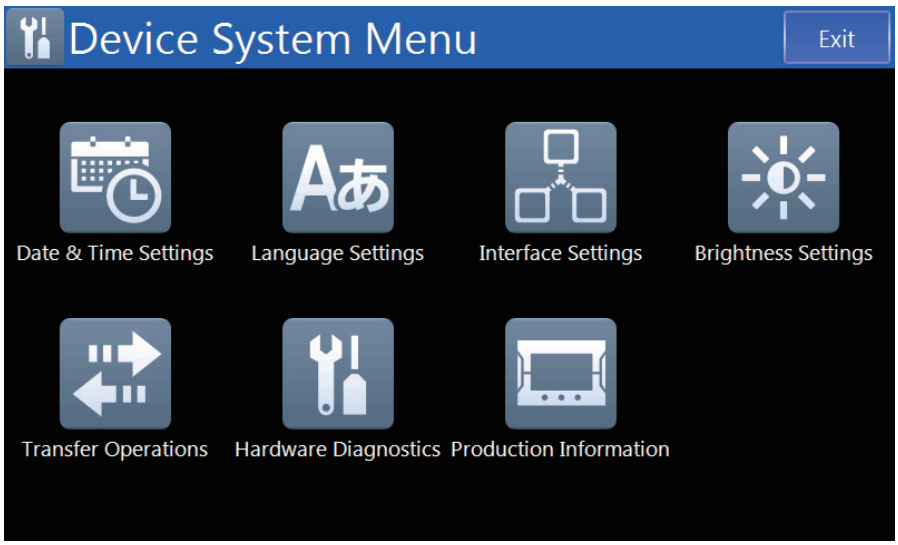


8-5-4 Uploading with Storage Media

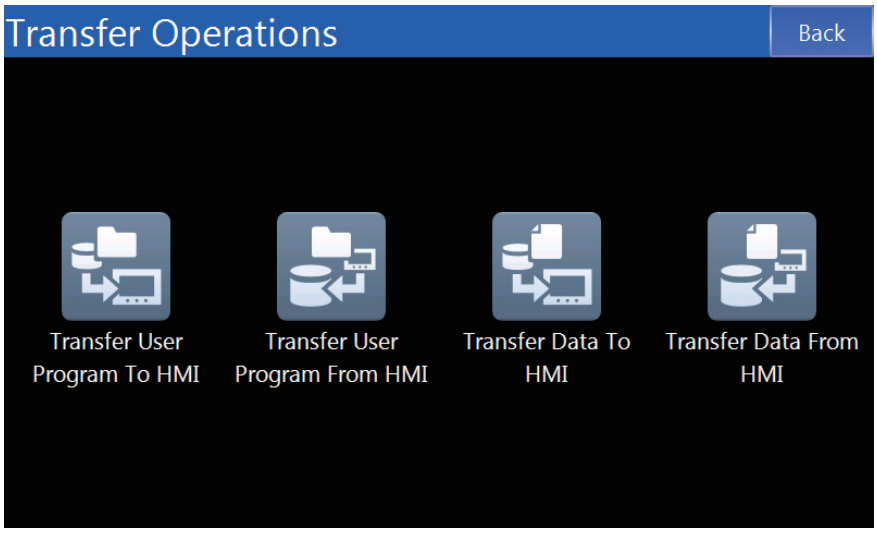
With an HMI, you can perform project synchronization with storage media.

1 Insert the SD Memory Card or USB memory device to use for the upload into the computer.

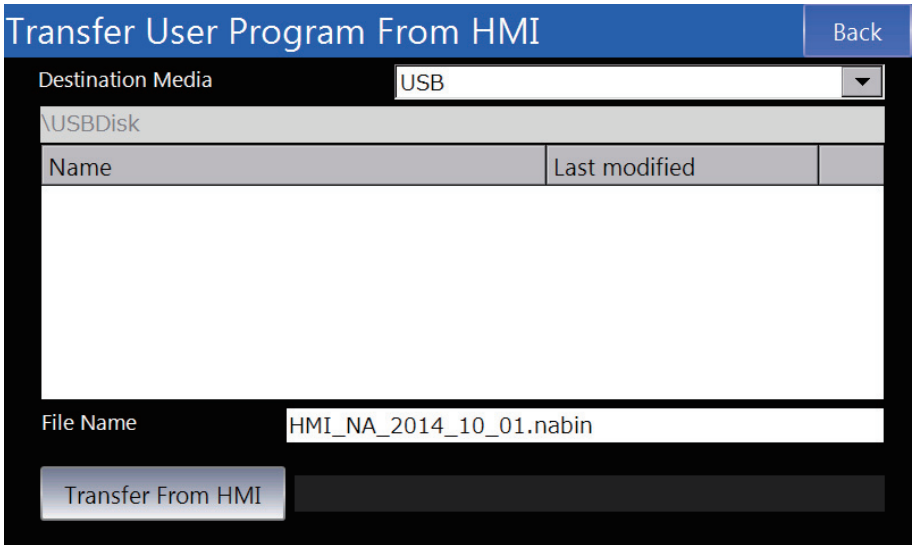
2 Display the Device System Menu and touch the **Transfer Operations** Button.



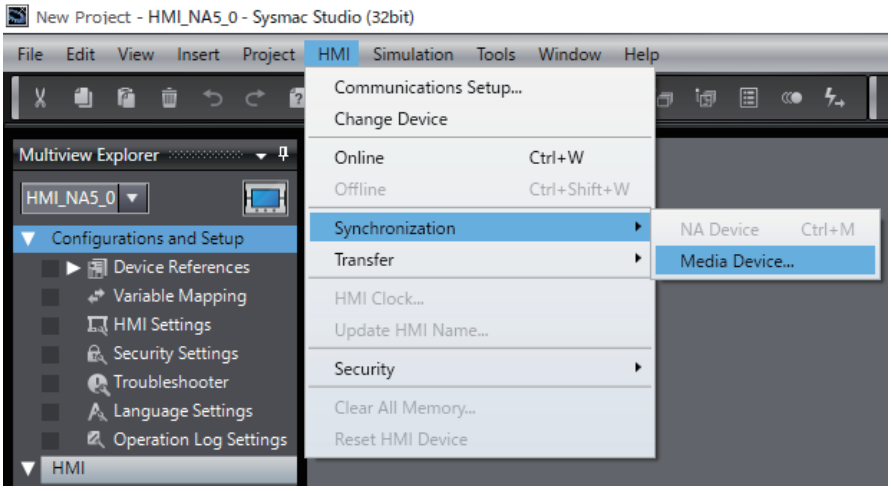
3 Touch the **Transfer User Program From HMI** Button.



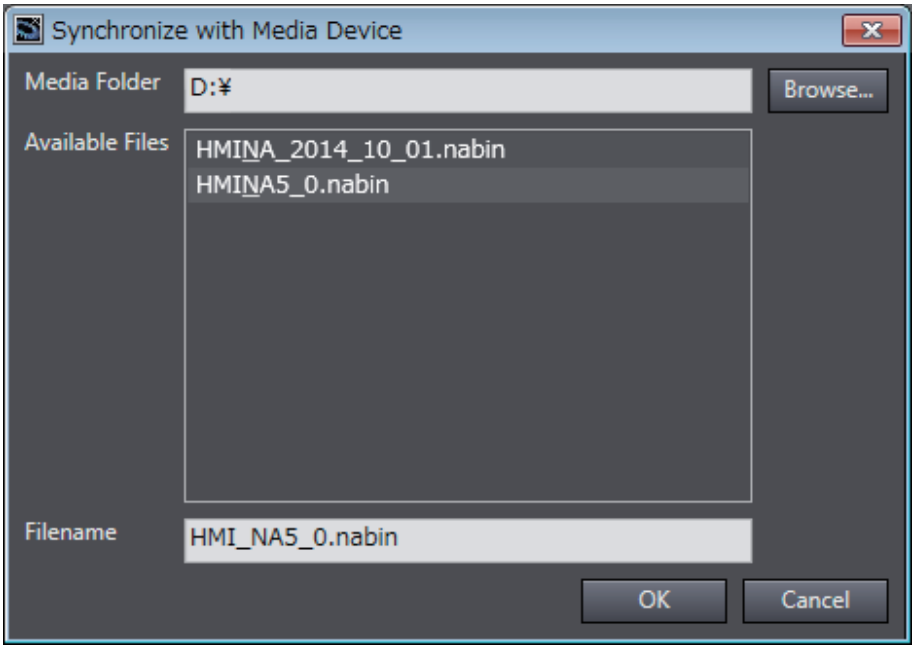
4 Specify the destination media and file name and touch the **Transfer From HMI** Button. The project is uploaded to the specified media.



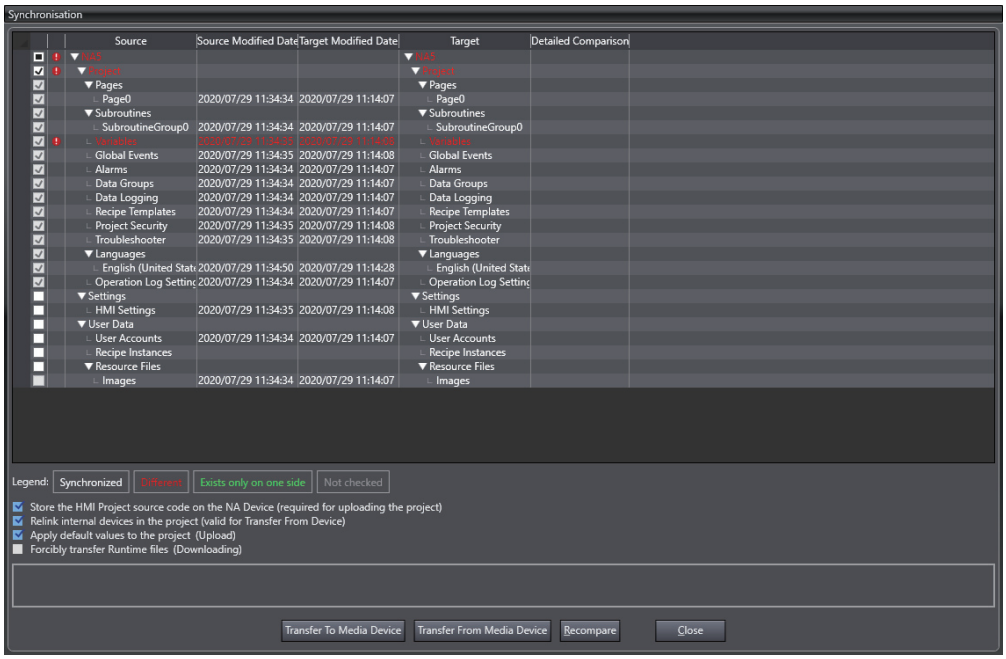
5 Insert an SD Memory Card or USB memory device into the computer and select ***Synchronization - Media Device*** from the HMI Menu.



6 Specify the file name on the media specified for the upload and click the **OK** Button.



7 The project on the Sysmac Studio is compared with the project in the storage media and the Synchronization Window is displayed.

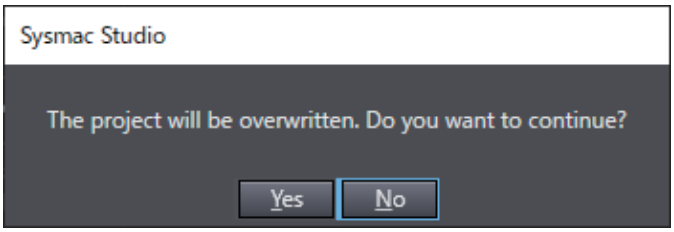


Precautions for Correct Use

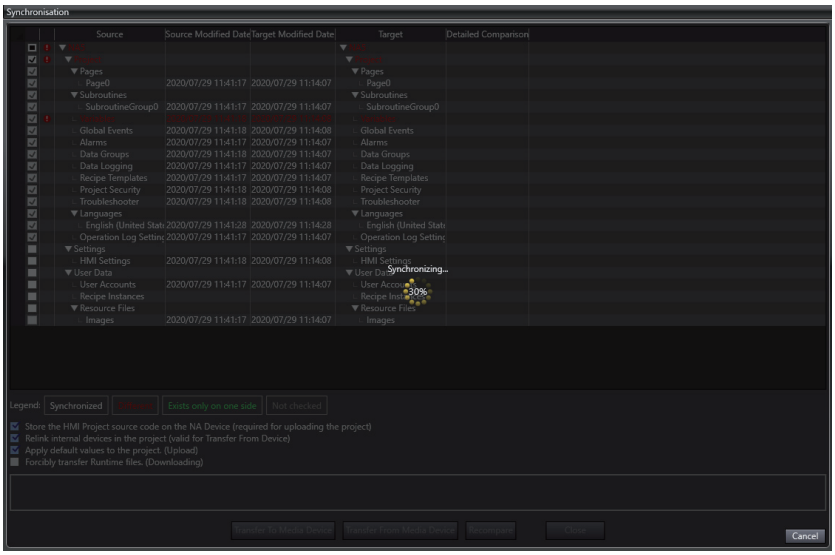
When a project is uploaded, the internal devices are not linked. In this condition, you will not be able to perform offline debugging with the Controller Simulator or variable mapping.

Refer to *8-4 Relinking Internal Devices* on page 8-5 for information on how to relink internal devices.

8 Click the **Transfer From Media Device** Button. The following confirmation message for overwriting is displayed. Click the **Yes** Button.



The project is uploaded and the changes are applied to the open project.



9

Reusing Objects

This section describes how to reuse objects.

9-1 Reusing Objects 9-2

9-2 IAGs 9-3

9-2-1 Differences when an IAG Project Is Selected 9-3

9-2-2 Creating an IAG 9-10

9-2-3 Using IAGs 9-13

9-3 Custom Objects 9-16

9-3-1 Objects That You Can Register as Custom Objects 9-16

9-3-2 Creating Custom Objects 9-16

9-3-3 Deleting Custom Objects 9-19

9-3-4 Using Custom Objects 9-20

9-1 Reusing Objects

The Sysmac Studio provides the following two functions to simplify and increase the speed of the development of HMI applications.

- Intelligent application gadgets (IAGs)
- Custom objects

IAGs and custom objects are different in the following ways.

- You can distribute IAGs to other parties. You can use custom objects only on the Sysmac Studio.
- You must treat IAGs as IAGs even after you place them on pages. You treat custom objects like any other objects after you place them on pages.
- After you place an IAG on a page, you can change only the properties of the IAG objects. You can change any of the attributes of custom objects.
- You can hide the contents of subroutines in IAGs. Therefore, you can provide IAGs to other parties without disclosing technology.

You can customize objects yourself and create reusable objects that combine other objects.

9-2 IAGs

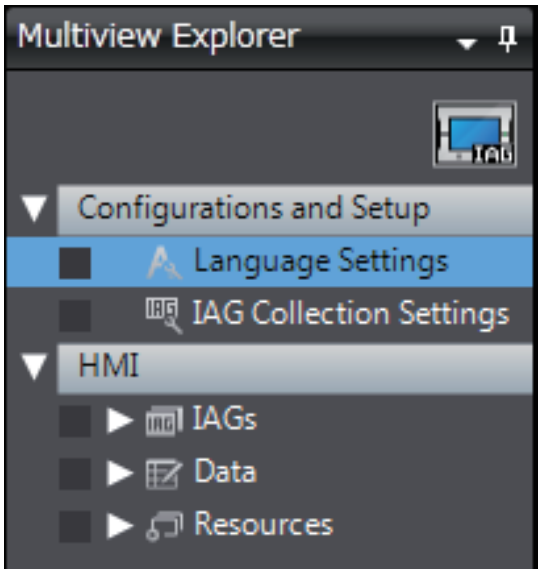
An IAG is a library object that you can distribute to other parties. To create a new IAG, you must create it in an IAG project. This section describes the procedures to create and use IAGs.

9-2-1 Differences when an IAG Project Is Selected

Even when an IAG project is selected, basic operations are generally the same as for a standard project. However, the following items are different.

Items in the Multiview Explorer

The structure of the Multiview Explorer when you select an IAG project is shown below. You can edit the user-defined variables, subroutines, and other resources in an IAG from the menu that is displayed when you right-click the IAG.



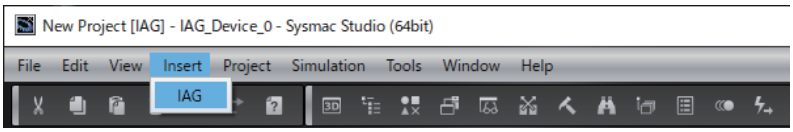
Configurations and Setup
Language Settings
IAG Collection Settings

HMI
IAGs
Data
Resources

Toolbar

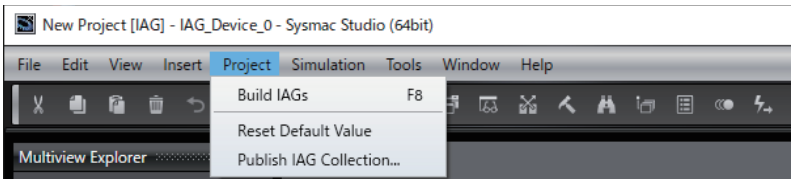
This section describes the differences in the Toolbar compared with a standard project.

● Insert Menu



Item	Description	Remarks
IAG	Inserts a new IAG in **IAGs.**	

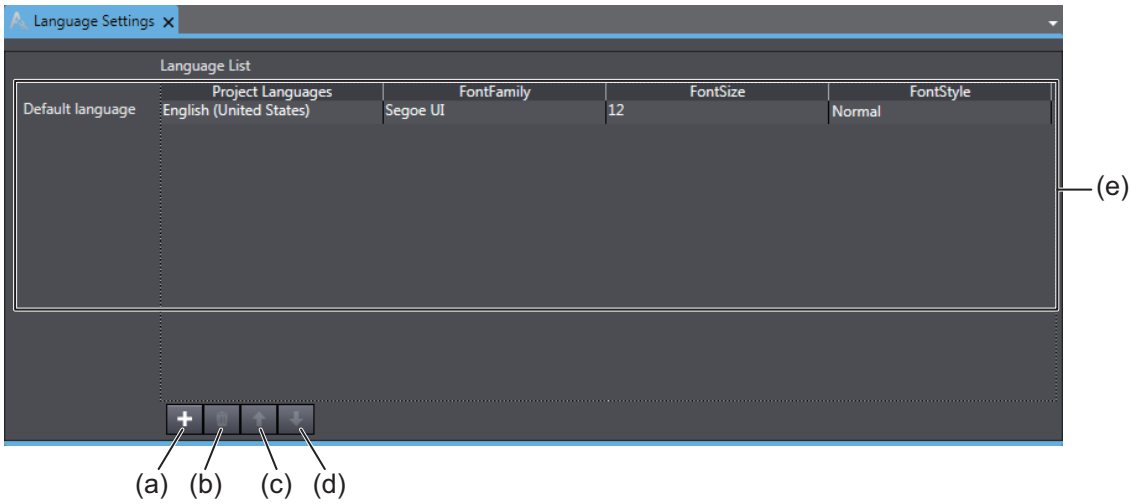
● Project Menu



Item	Description	Remarks
Build IAGs	Builds all of the IAGs.	
Publish IAG Collection	Saves an IAG collection in a file format that you can use in a standard project.	

Language Settings

You can set the languages to use in the IAG.



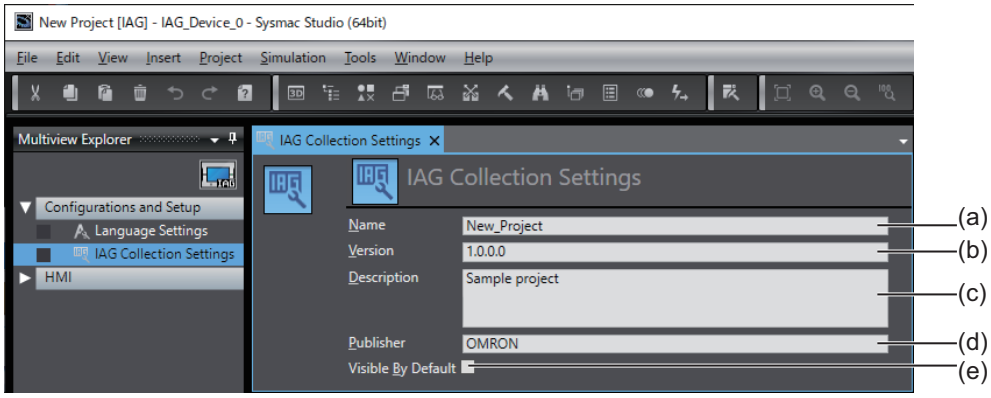
Symbol	Item	Description
(a)	+	Adds a language to the language list.
(b)		Deletes the selected language from the language list.
(c)		Moves the selected language one position up in the language list.
(d)		Moves the selected language one position down in the language list.

Symbol	Item		Description
(e)	Language List		Displays a list of the languages. The language at the top of the list is treated as the default language.
		Project Languages	Sets the project language.
		FontFamily *1	Sets the font that is set by default when an object is created.
		FontSize *1	Sets the font size that is set by default when an object is created.
		FontStyle *1	Sets the font style that is set by default when an object is created.

*1. Only appears when the Runtime version is 1.11 or higher.

IAG Collection Settings

You can make the following settings for an IAG collection.

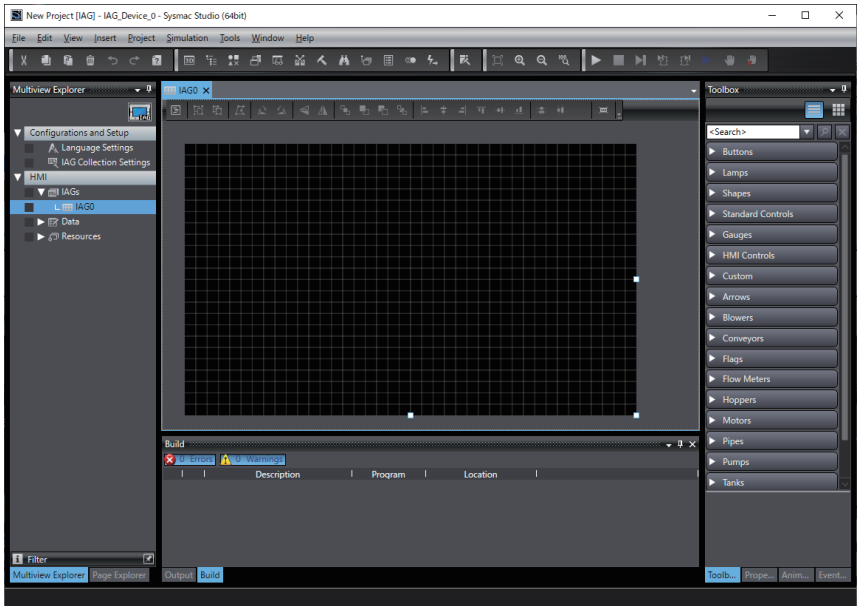


Symbol	Item	Description	Remarks
(a)	Name	Sets the name.	
(b)	Version	Sets the version.	
(c)	Description	Sets a description.	
(d)	Publisher	Sets the publisher.	
(e)	Visible By Default	Select this check box to display the IAG collection in the Toolbox when the IAG collection is imported into the Sysmac Studio.	

IAGs

The following tab page is used to create an IAG. The procedures are generally the same as for standard projects, except for the following differences.

- There are no global variables. Only the user variables for each IAG can be used.
- You cannot set page animations.
- You cannot use functions that cannot be placed on a page, such as global events and recipes.
- You cannot use Trend Graph or Broken-line Graph objects and user-created IAGs.



Additional Information

You cannot use names that are reserved by the system for user event names.

User Variables

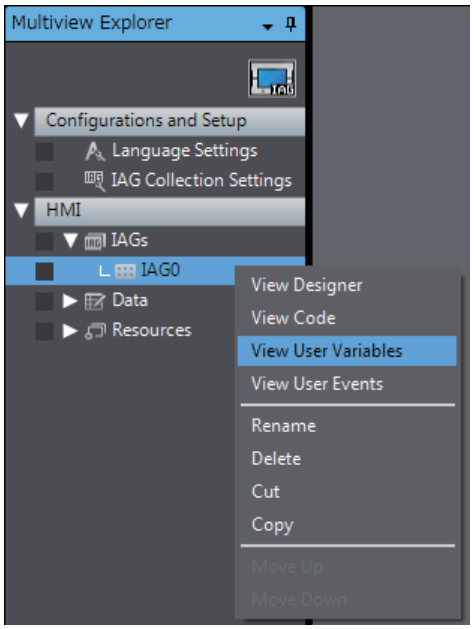
For IAG, only the variables for each IAG can be used. The variables that are defined for each IAG are called user variables.
You can create user variables that are internal variables and user variables that are In/Out variables.

Additional Information

Registering different structure variables with the same name to different IAG user variables may cause errors to occur when assigning them to objects, etc. When variable names are the same, assign the same structure even if they are different IAG user variables.

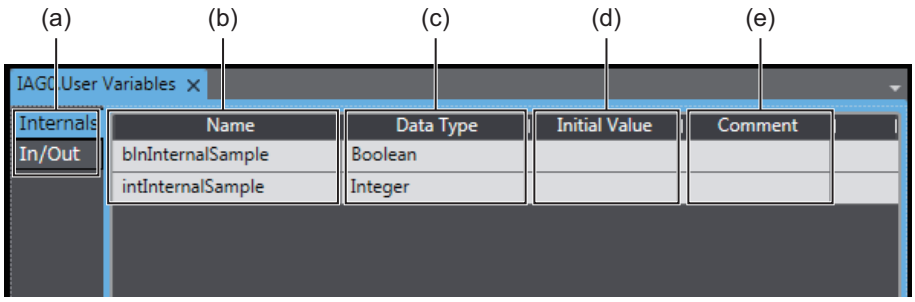
Editing User Variables

To edit user variables, right-click **IAG** and select ***View User Variables*** from the menu.



Internal Variables

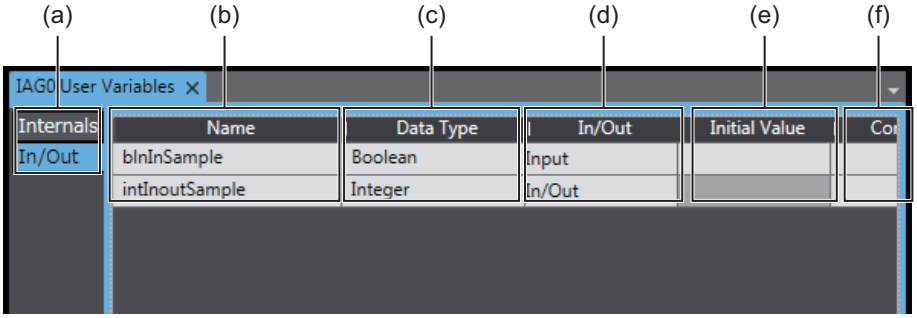
The internal variables are used only within the IAG. The internal variables are displayed in the user variable table, but you cannot access them when you use the IAG.



Letter	Item	Description	Remarks
a	Internals or In/Out	Changes the display between internal variables and In/Out variables.	
b	Name	Sets the variable name.	
c	Data Type	Sets the data type.	
d	Initial Value	Sets the initial value.	
e	Comment	Sets a comment.	

● In/Out Variables

In/Out variables are exposed externally. You can access the In/Out variables when you use the IAG.

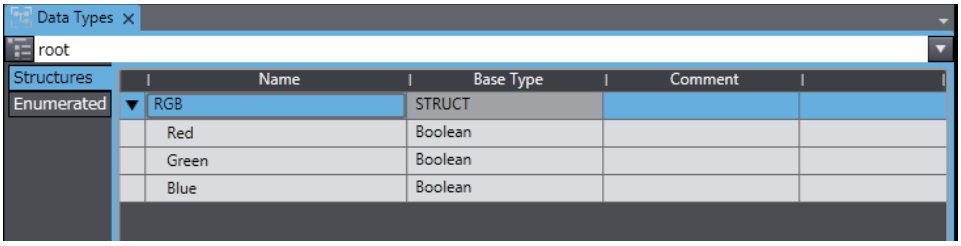


Letter	Item	Description	Remarks
a	Internals or In/Out	Changes the display between internal variables and In/Out variables.	
b	Name	Sets the variable name.	
c	Data Type	Sets the data type.*1	
d	In/Out	If you set an In/Out variable, you can read and write the variable when you use the IAG. If you set an input variable, you can only write the variable when you use the IAG.	
e	Initial Value	Sets the initial value.	
f	Comment	Sets a comment.	

*1. Enumeration type is not supported.

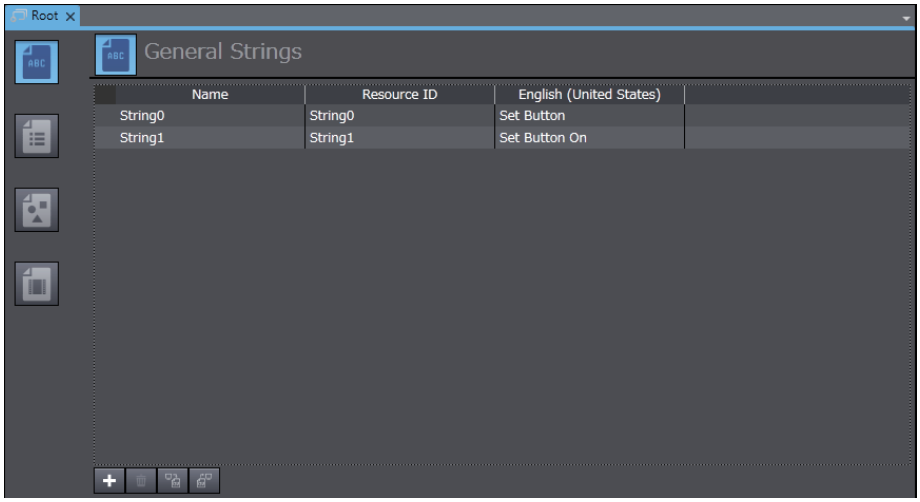
Data Types

You can set the data types to use in the IAG. The procedures are the same as for a standard project.



Resources

You can set the resources to use in the IAG. The procedures are the same as for a standard project.

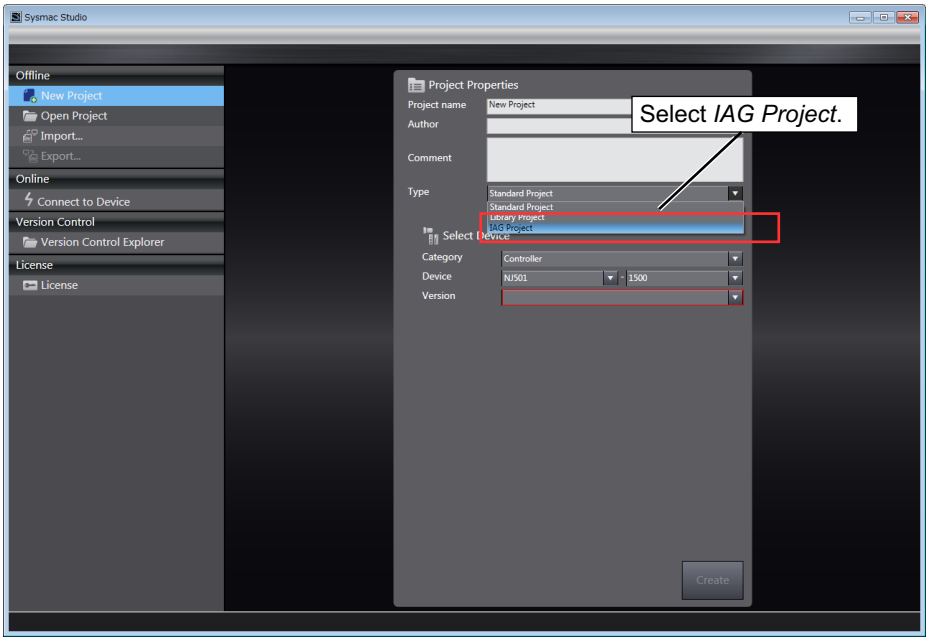


9-2-2 Creating an IAG

This section describes how to create an IAG.

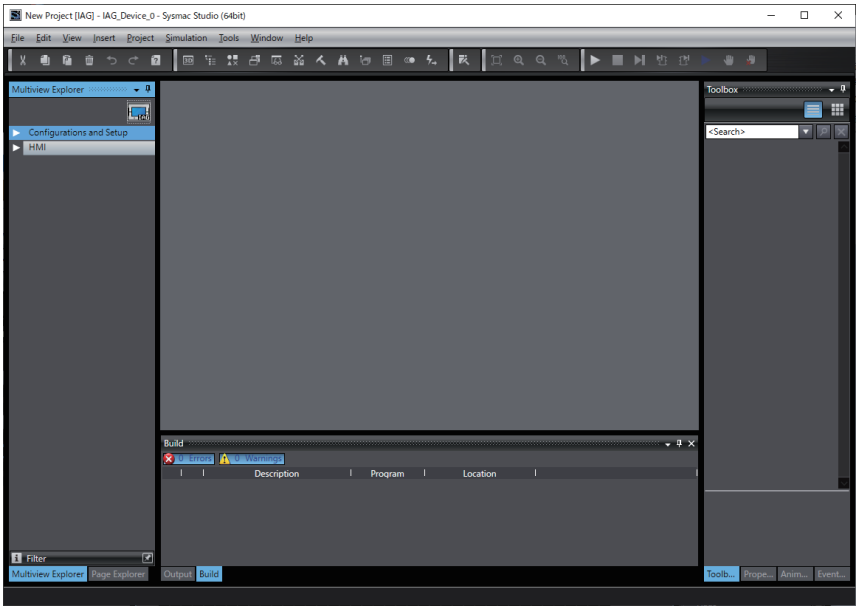
The size of the IAG will be the size of the page when the IAG is created. Any objects that are positioned outside of the page are ignored.

1 Create a new project and set the project type to *IAG Project*.



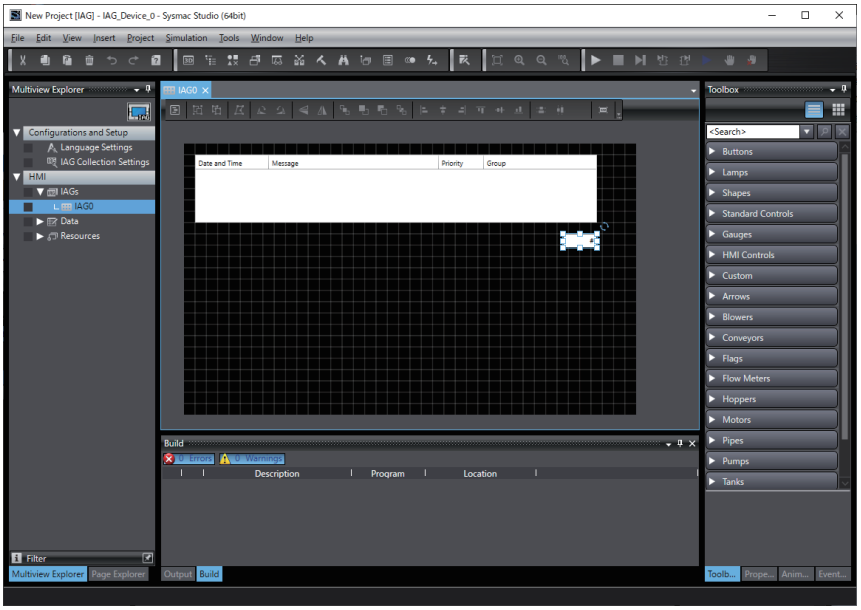
2 Click the **Create** Button.

An IAG project is created and the following window is displayed.

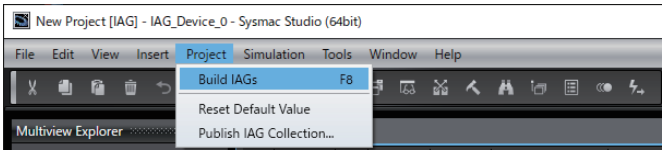


3 Right-click the IAG, select ***View User Variables*** from the menu, and then register the required user variables.

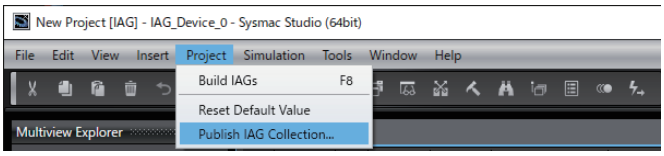
4 Create the IAG with the same methods as for a normal page.



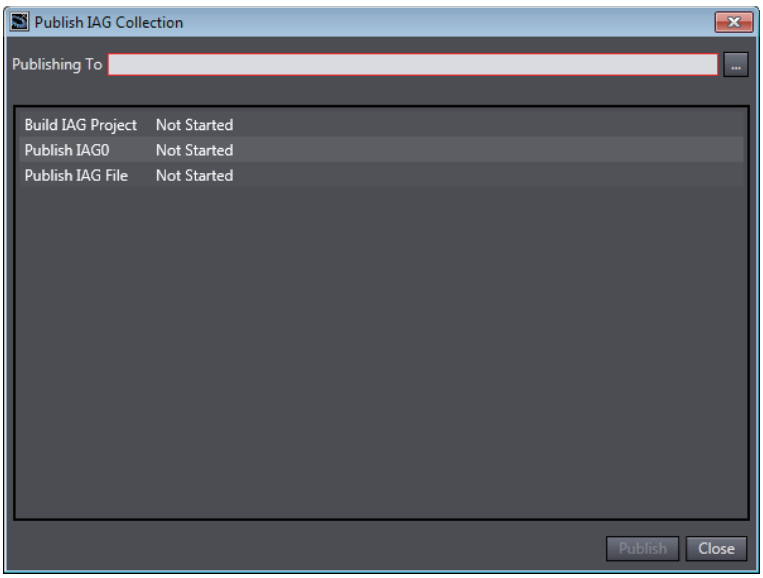
5 Select ***Build IAGs*** from the Project Menu.



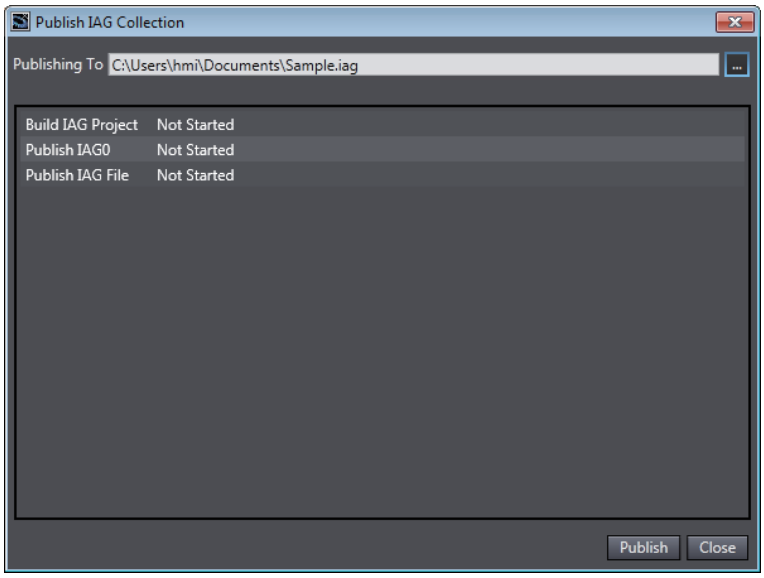
6 Select ***Publish IAG Collection*** from the Project Menu.



7 Click the **Browse** Button (...) and specify where to save the collection.



8 Click the **Publish** Button. The IAG collection is created.



9-2-3 Using IAGs

This section describes how to use the IAGs that you create when you edit a standard project.

To use IAGs, you must first register them in the Toolbox. After you register them in the Toolbox, you can drag them to the page to use them in the same way as for other objects.

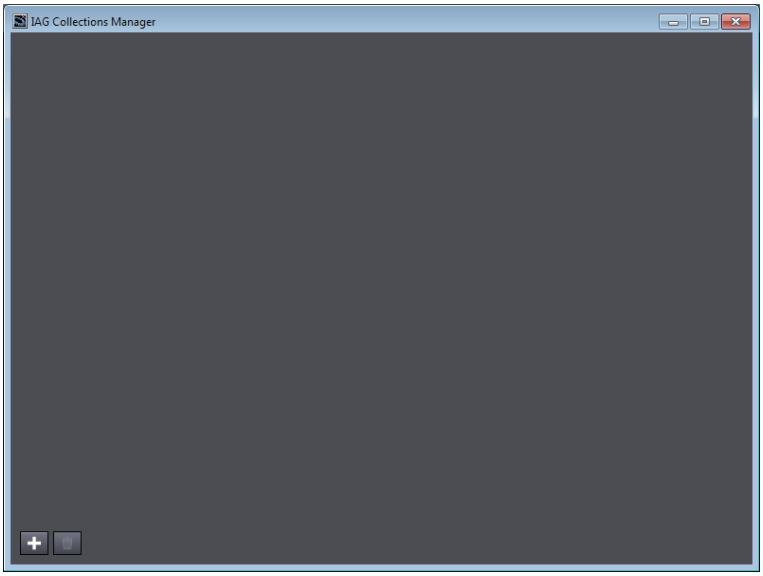
Additional Information

When copy-and-pasting pages using an IAG to different projects, one of the following conditions must be satisfied.

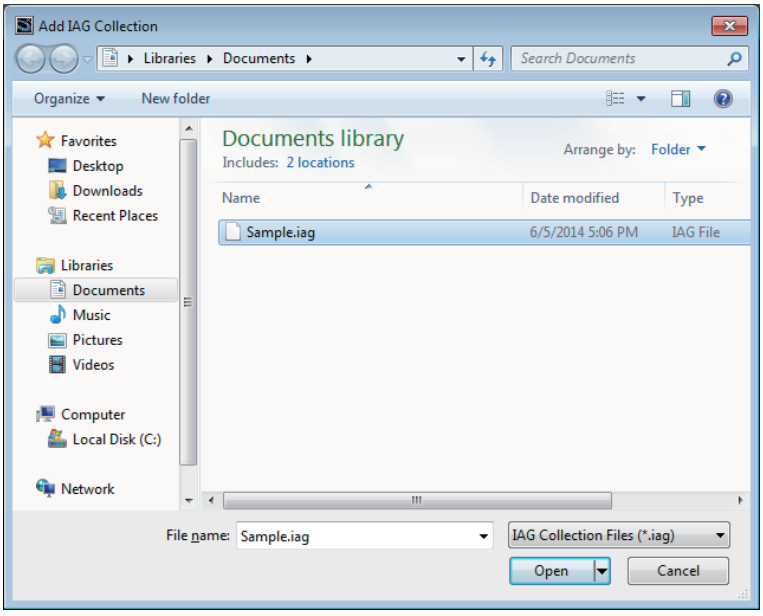
- The same IAG is registered in an IAG resource in the project being pasted.
- An IAG collection containing the same IAG is registered in the Sysmac Studio IAG Collections Manager.

Registering IAGs

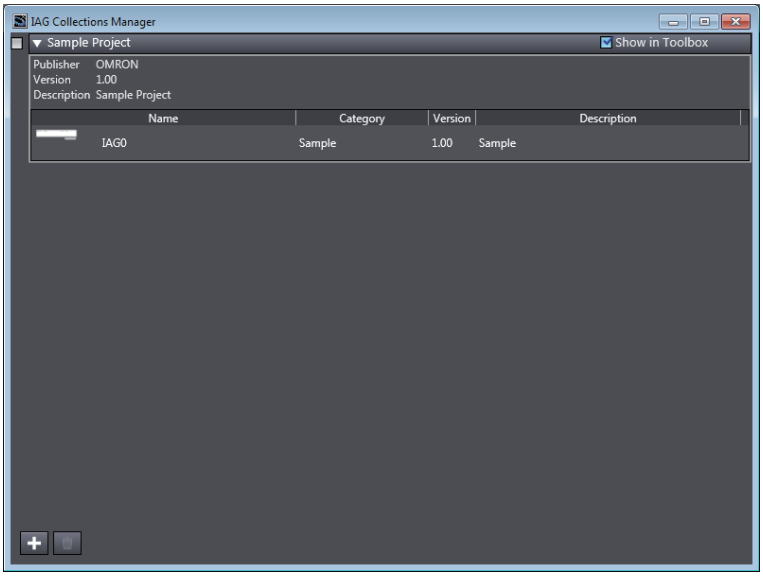
1 Select ***IAG Collections Manager*** from the Project Menu.



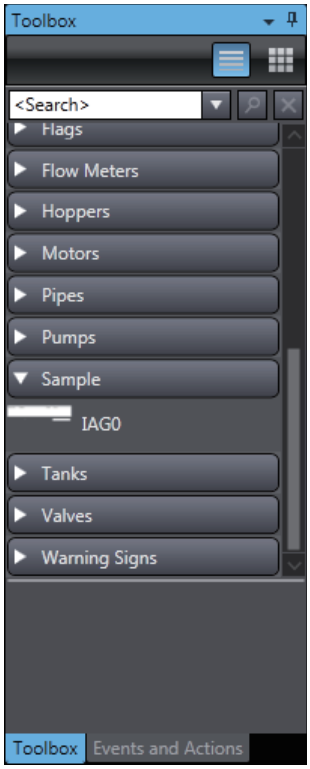
2 Click the **+** Button. Select the IAG file and click the **Open** Button.



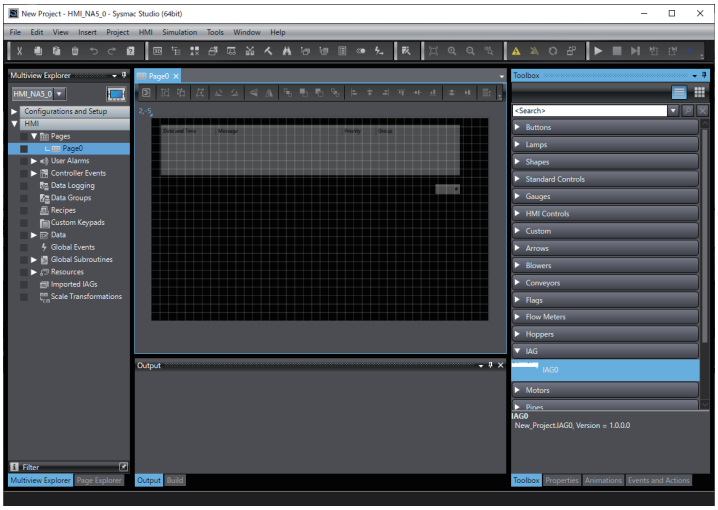
3 Select the *Show in Toolbox* Check Boxes for the IAG collections that you want to display in the Toolbox.



4 The IAGs for the selected check boxes are displayed in the Toolbox.

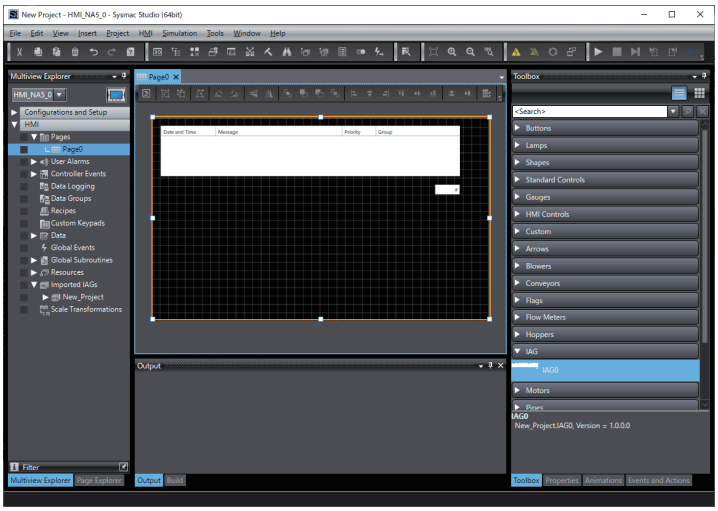


5 Drag the IAG to the page.



6 Double-click the IAG and set the properties.

Set global variables or constants for the variables that are set as In/Out variables.



9-3 Custom Objects

You can register the objects that you use most frequently to increase your productivity. This section describes the procedures to create and use custom objects.

9-3-1 Objects That You Can Register as Custom Objects

An object must meet the following conditions before you can register it as a custom object.

- You can register only individual objects.

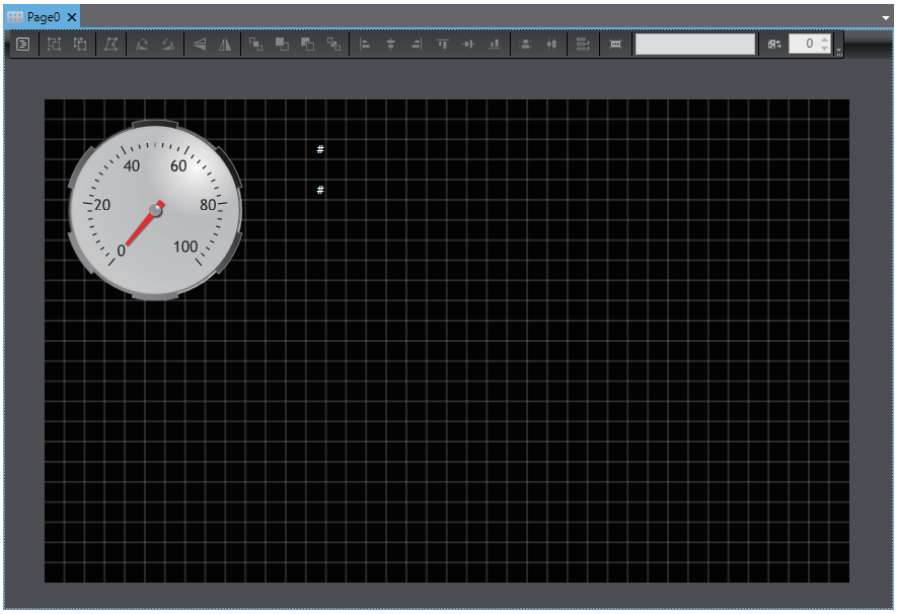
  You cannot register more than one object as a custom object. If you want to register more than one object as a custom object, group the objects into one object first.
- The object cannot be a user-defined IAG.

  You cannot register a user-defined IAG or a group that contains a user-defined IAG as a custom object.

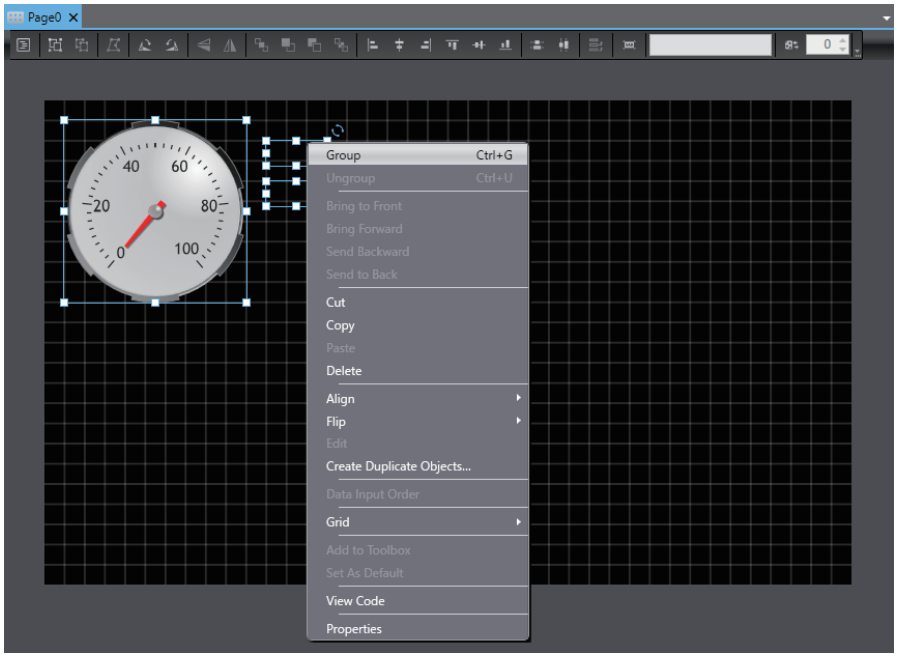
9-3-2 Creating Custom Objects

Use the following procedure to create a custom object.

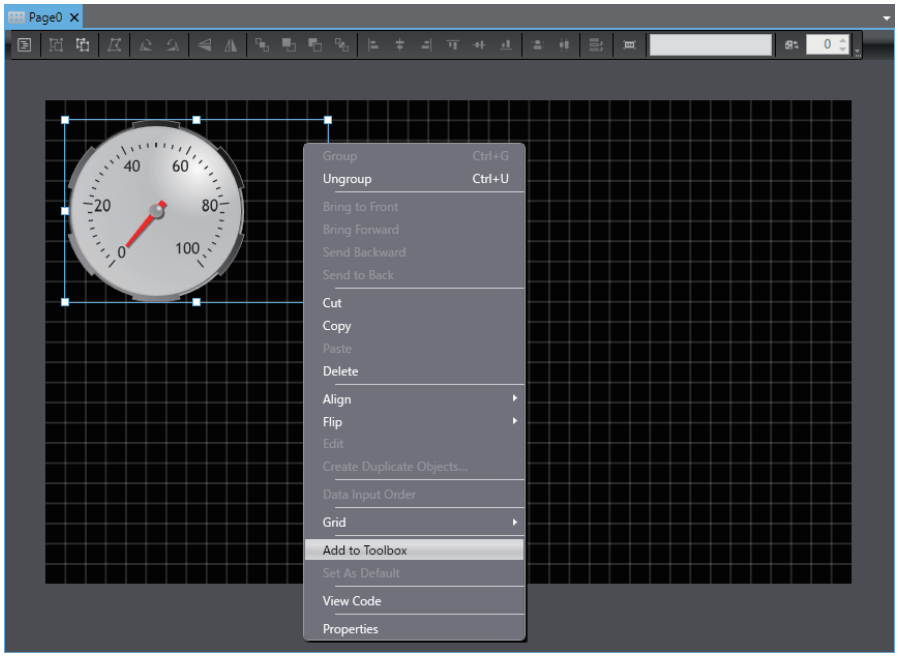
1 Create a standard project and place the objects to register as a custom object on the page.



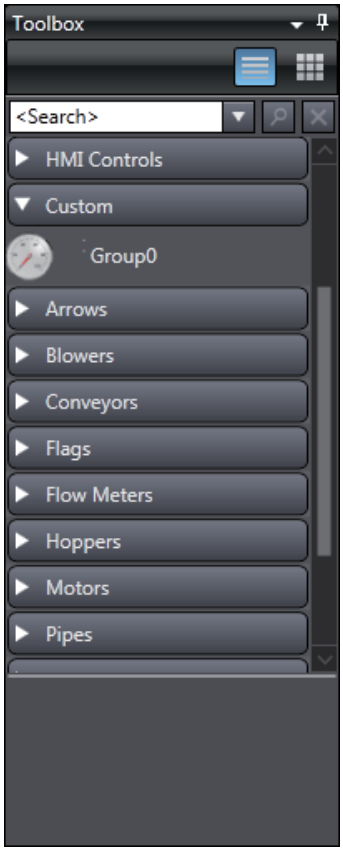
2 Group the objects to register the group as a custom object.



3 Right-click the group of objects and select ***Add to Toolbox*** from the menu.



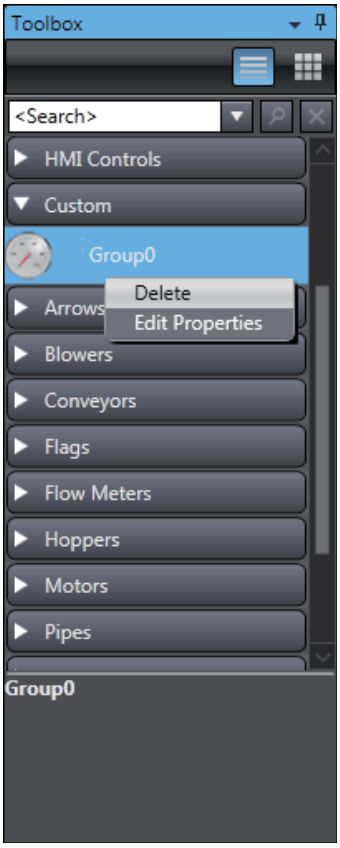
4 The group is added as a custom object under **Custom** in the Toolbox. The displayed name is the name of the registered group or object.



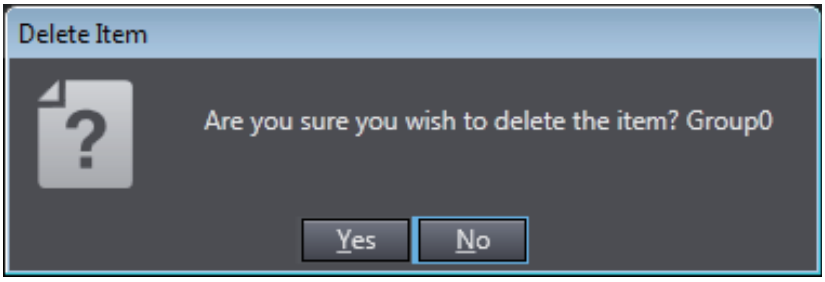
9-3-3 Deleting Custom Objects

Use the following procedure to delete a registered custom object.

1 Right-click the custom object to delete from the Toolbox. Select ***Delete*** from the menu.

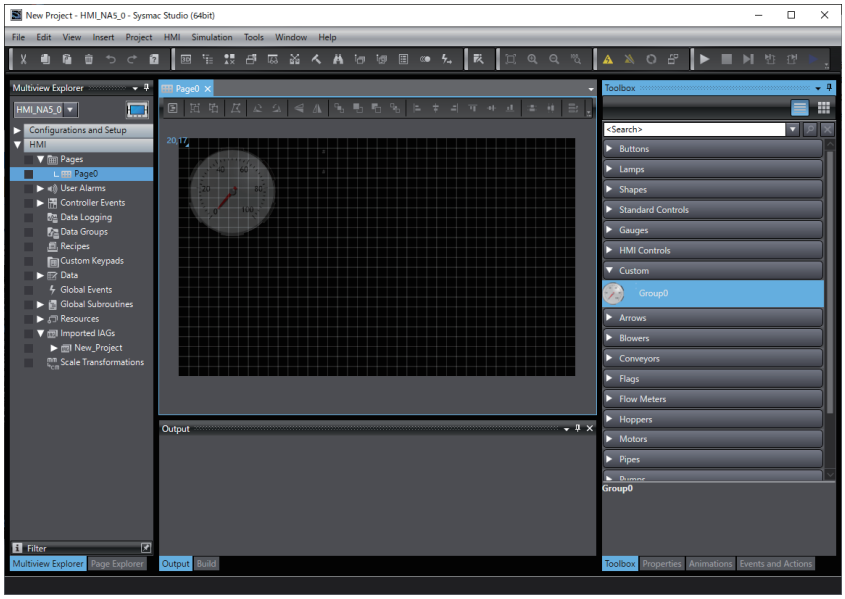


2 A confirmation dialog box is displayed. Click the **Yes** Button.



9-3-4 Using Custom Objects

It is very easy to use a custom object. Just select the desired custom object in the Toolbox and drag it to the page. You can handle the custom objects on pages in the same ways as you handle normal objects.



10

Connecting to HMIs from External Devices

This section describes how to connect to an HMI from an external device.

10-1 Accessing an HMI from an External Device 10-2
10-1-1 VNC 10-2
10-1-2 FTP 10-3

10-1 Accessing an HMI from an External Device

You can use the following two methods to access an NA-series Programmable Terminal from an external device.

- Remote monitoring and control with VNC
- File operations with FTP

Precautions for Correct Use

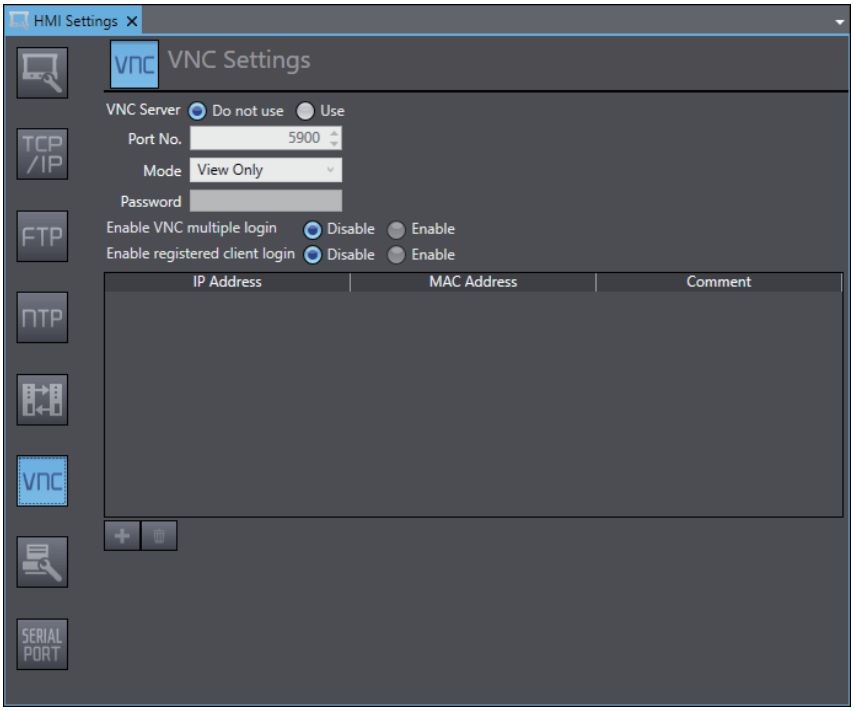
- Only password security is provided for using VNC and FTP. Sufficiently consider the network configuration in terms of security and implement any required measures to prevent unauthorized access.
- Use the same keyboard layout settings for the VNC client computer and the HMI. If the settings are not the same, different characters may results from the characters input from the VNC client.

10-1-1 VNC

You can enable VNC to use a VNC client to monitor and control HMI pages. You can also use a mode setting to prohibit controlling operation from a VNC client and allow only monitoring.

Setting Method

Double-click **HMI Settings** under **Configurations and Setup**. Click the **VNC Settings** Button.



Select the *Use* Option for *Enable VNC* and change the setting of the *Mode* Box as required. Set a text string in the *Password* Box.

After you complete the settings, select ***Build HMI*** from the Project Menu. When building the project is completed, download the project to the HMI.

After the download is completed, you can access the HMI from a VNC client.

Additional Information

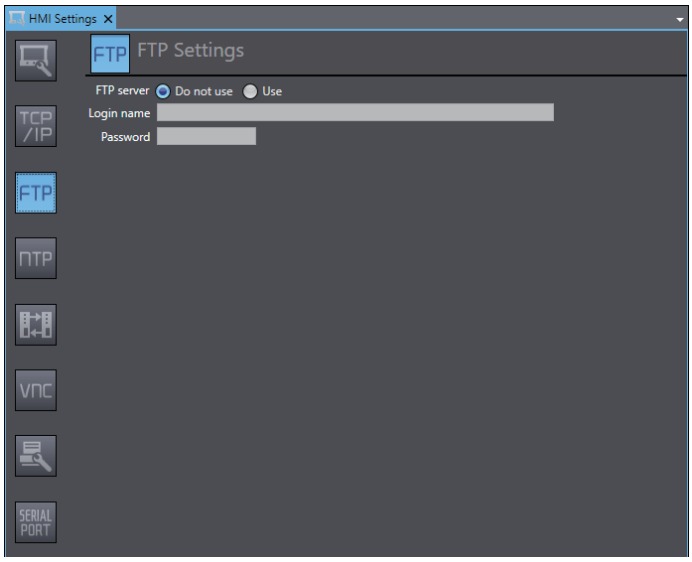
Refer to the relevant software manuals for the operating procedures of the VNC client.

10-1-2 FTP

You can enable the FTP server to use an FTP client to access files in the HMI. However, you can access only folders prepared for FTP in the NA Unit. Folders storing user projects and the SD memory card, etc. cannot be accessed.

Setting Method

Double-click **HMI Settings** under **Configurations and Setup**. Click the **FTP Settings** Button.



Select the *Use* Option. Set text strings for the login name and password.

After you complete the settings, select ***Build HMI*** from the Project Menu. When building the project is completed, download the project to the HMI.

After the download is completed, you can access the HMI from an FTP client.

Additional Information

Refer to the relevant software manuals for the operating procedures of the FTP client.

11

Other Functions

This section describes other functions that the Sysmac Studio provides for HMIs.

11-1 Sysmac Studio Option Settings 11-2

11-2 Printing 11-5

11-2-1 Printable Items 11-5

11-3 Image File Output 11-6

11-4 Import/Export User Alarm 11-7

11-4-1 Importing User Alarms 11-7

11-4-2 Exporting User Alarms 11-9

11-4-3 File Format 11-11

11-5 Import/Export Resources 11-13

11-5-1 Importing Resources 11-13

11-5-2 Exporting Resources 11-16

11-5-3 File Format 11-19

11-6 Import/Export Object Properties 11-20

11-6-1 Importing Object Properties 11-20

11-6-2 Exporting Object Properties 11-21

11-6-3 File Format 11-22

11-7 Importing/Exporting Pages 11-24

11-7-1 Importing Pages 11-24

11-7-2 Exporting Pages 11-25

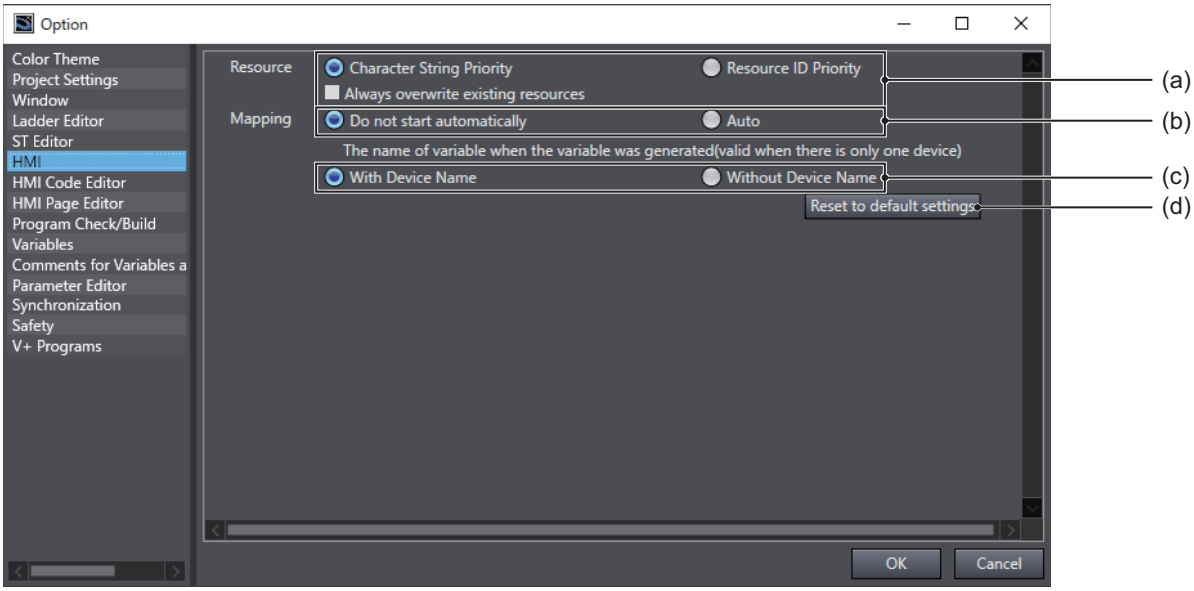
11-1 Sysmac Studio Option Settings

The following Sysmac Studio option settings are related to HMIs.

- HMI
- HMI Code Editor
- HMI Page Editor

HMI

In HMI, you can configure settings for all operation of the HMI.



Symbol	Item	Description	Remarks
(a)	Resource	Select the priority if the same resource exists when copying and pasting. Character String Priority: If the character string is the same, you can change the resource ID before pasting. Resource ID Priority: If the same resource ID exists, pasting overwrites the resource. When the check box in [Always overwrite existing resources] is selected, if the resource ID is the same, the existing resource is overwritten when the text is changed in the object properties.	
(b)	Mapping	Select Auto to automatically map a variable when added in an internal device. This setting is only valid for an internal device.	

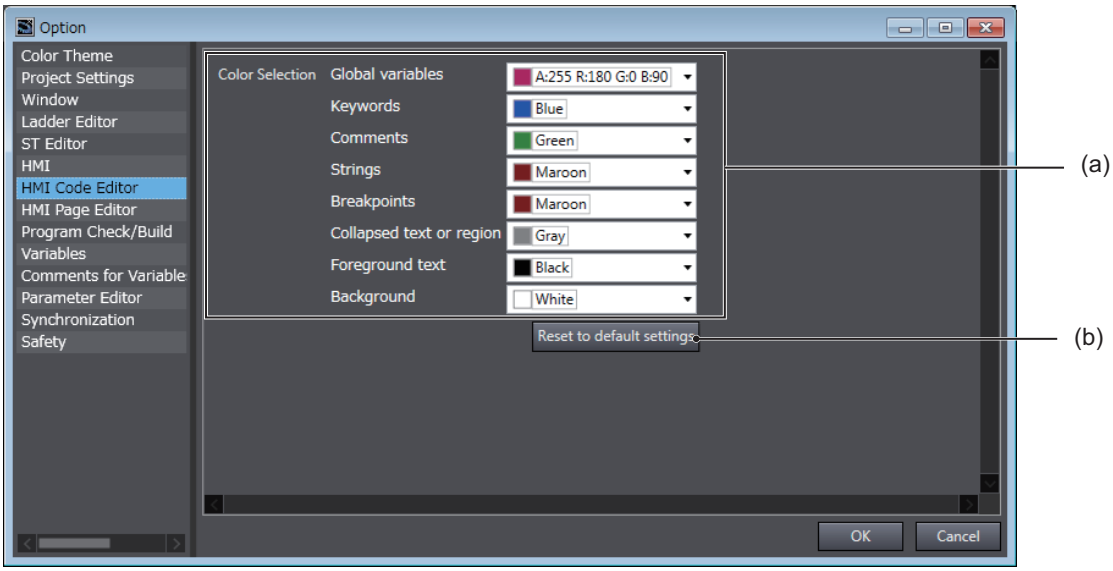
Symbol	Item	Description	Remarks
(c)	The name of variable when the variable was generated	Set whether or not the device name is added to the name of a device variable when variables are mapped automatically. This setting is only valid when there is only one controller in the project.	
(d)	Reset to default settings	You can reset the settings to the default settings.	

Additional Information

- Even when variable mapping is set to auto, variable mapping will not be executed when the following functions are executed.
  - Uploading by Going Online with an HMI
  - Uploading Using HMI Media
  - Importing an HMI Device
- When variable mapping is set to auto, variable mapping is executed when the device is changed, so changing takes longer.

HMI Code Editor

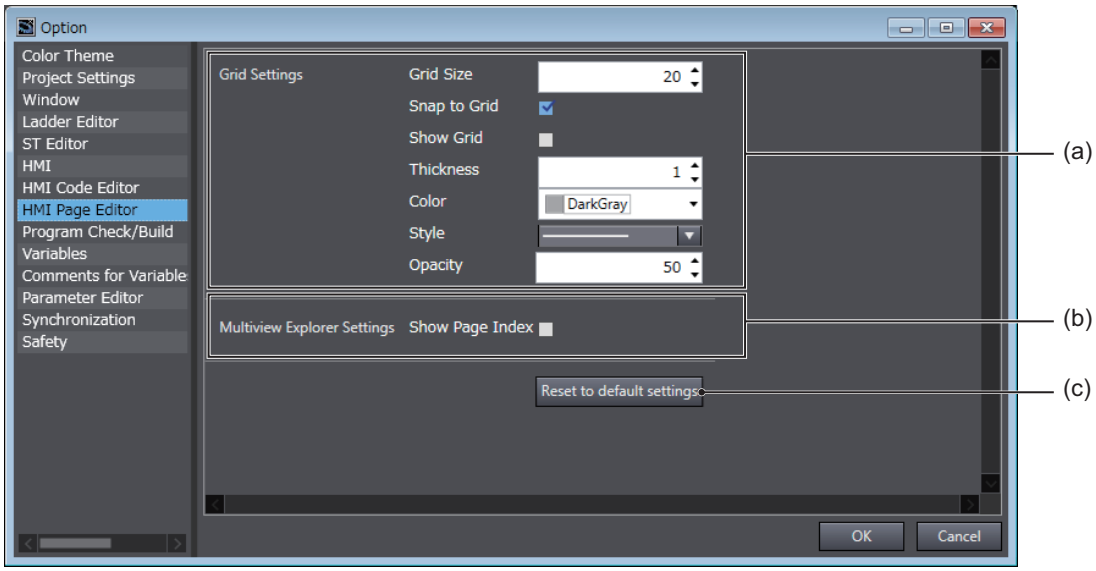
You can make settings to highlight text in the HMI Code Editor.



Symbol	Item	Description	Remarks
(a)	Color Selections	Select the color for each item.	
(b)	Reset to default settings	This button resets the colors to the default settings.	

HMI Page Editor

You can make settings for the HMI Page Editor.



Symbol	Item	Description	Remarks
(a)	Grid Settings	Make the settings for the grid.	
(b)	Multiview Explorer Settings	Make the settings for Multiview Explorer.	
(c)	Reset to Defaults	This button restores the default settings.	

11-2 Printing

This section describes the printing feature.

11-2-1 Printable Items

The following items can be printed. You can select the items to print.

Items	Category of project	
	Standard	IAG
Device References	✓	
Variable Mapping	✓	
Pages	✓	
Page Subroutines	✓	
User Alarms	✓	
HMI Settings	✓	
Controller Events	✓	
Data Logging	✓	
Security Settings	✓	
Troubleshooter	✓	
Language Settings	✓	✓
Recipes	✓	
Data	✓	✓
Global Events	✓	
Global Subroutines	✓	
Resources	✓	✓
IAG Collection Settings		✓
IAGs		✓

Additional Information

For printing features that are unique to devices other than the HMI, refer to the relevant manuals.

For details on the settings and the procedure for printing, refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504).

11-3 Image File Output

This section explains the image file output function.

You can output any page in a project as an image file. Supported file types are BMP, JPEG, and PNG.

11-4 Import/Export User Alarm

This section describes importing and exporting of user alarms.

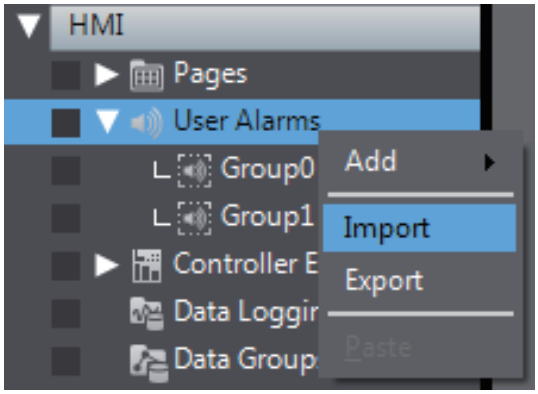
11-4-1 Importing User Alarms

You can either import all user alarms at once, or import only a specific group of user alarms.

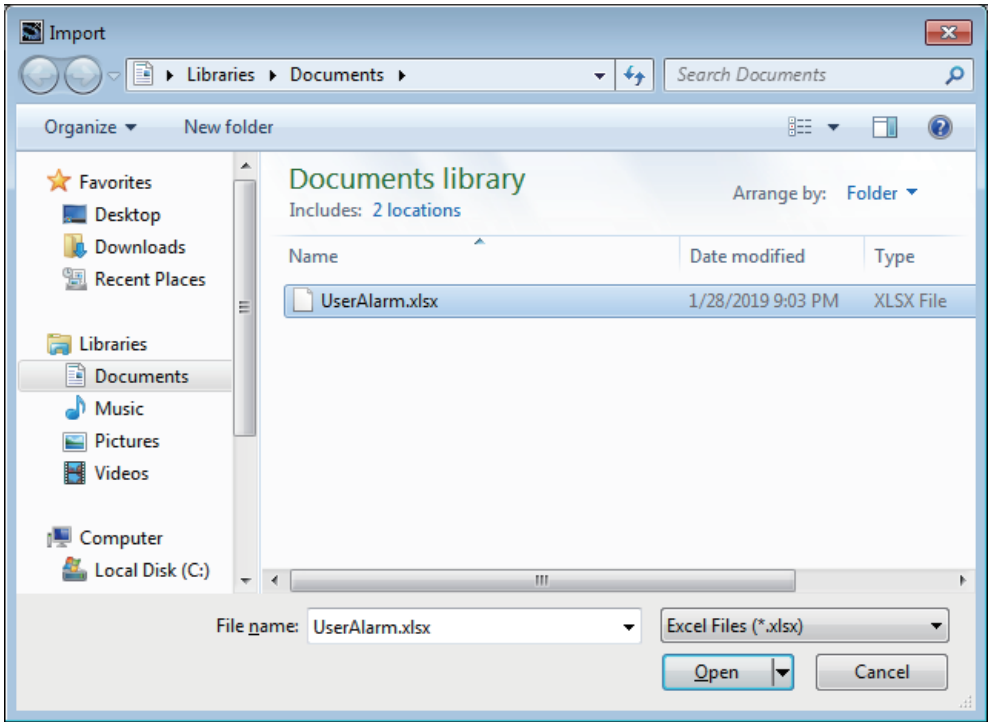
An import is executed based on resource IDs. This means a user alarm will be overwritten when the same resource ID is used in the open project and the file to import. Event and Action settings retain the state they had before importing.

Importing All User Alarms at Once

1 Right-click User Alarms. Select **Import** from the menu.

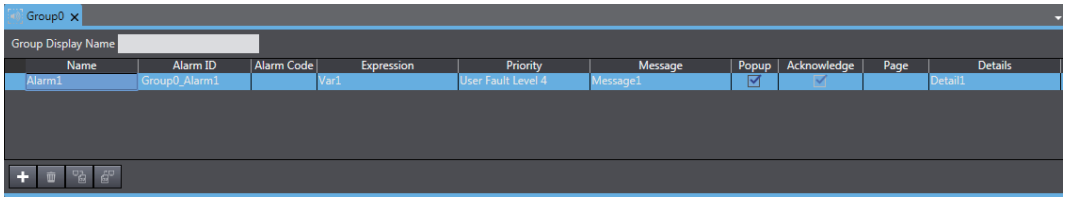


2 Select the file and click **Open**.



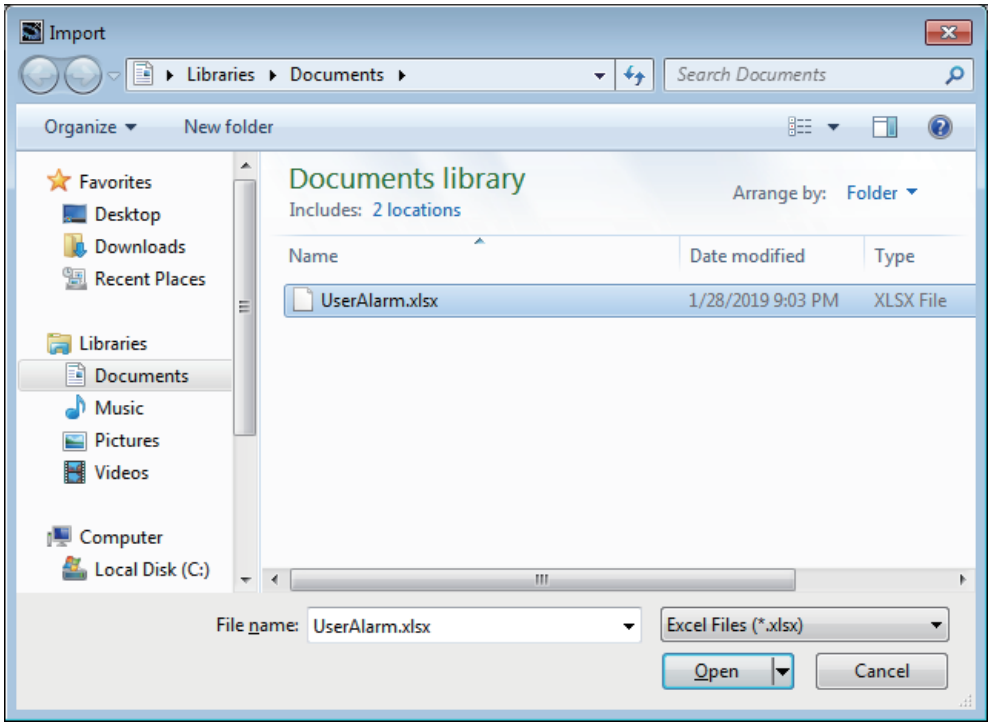
Importing a Specific Group

1 Open the group you want to import.



2 Click .

3 Select the file and click **Open**.



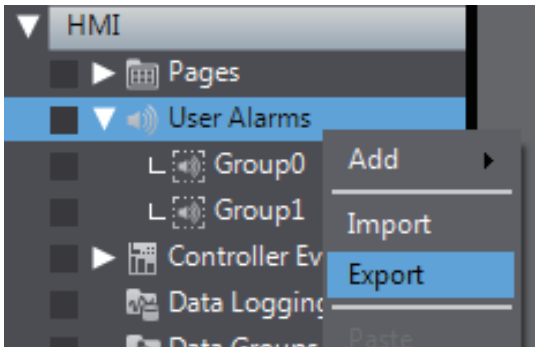
11-4-2 Exporting User Alarms

You can either export all user alarms at once, or export only a specific group of user alarms.

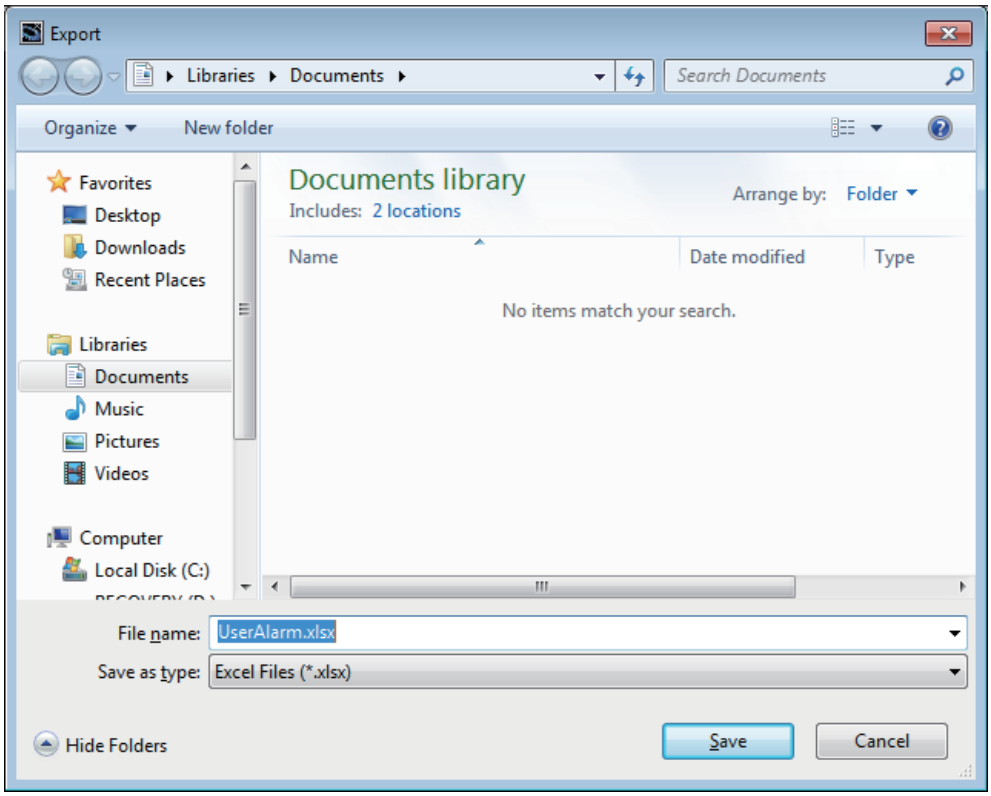
Event and Action settings are not exported.

Exporting All User Alarms at Once

1 Right-click User Alarms. Select **Export** from the menu.

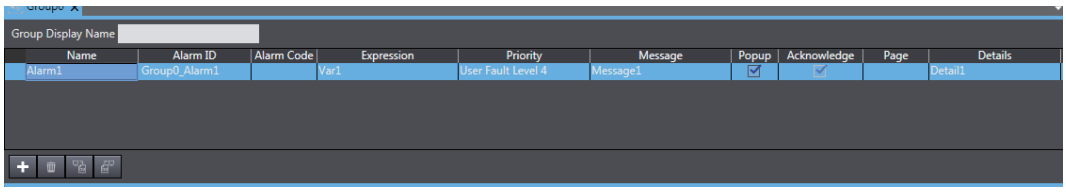


2 Set a file name and click **Save**.



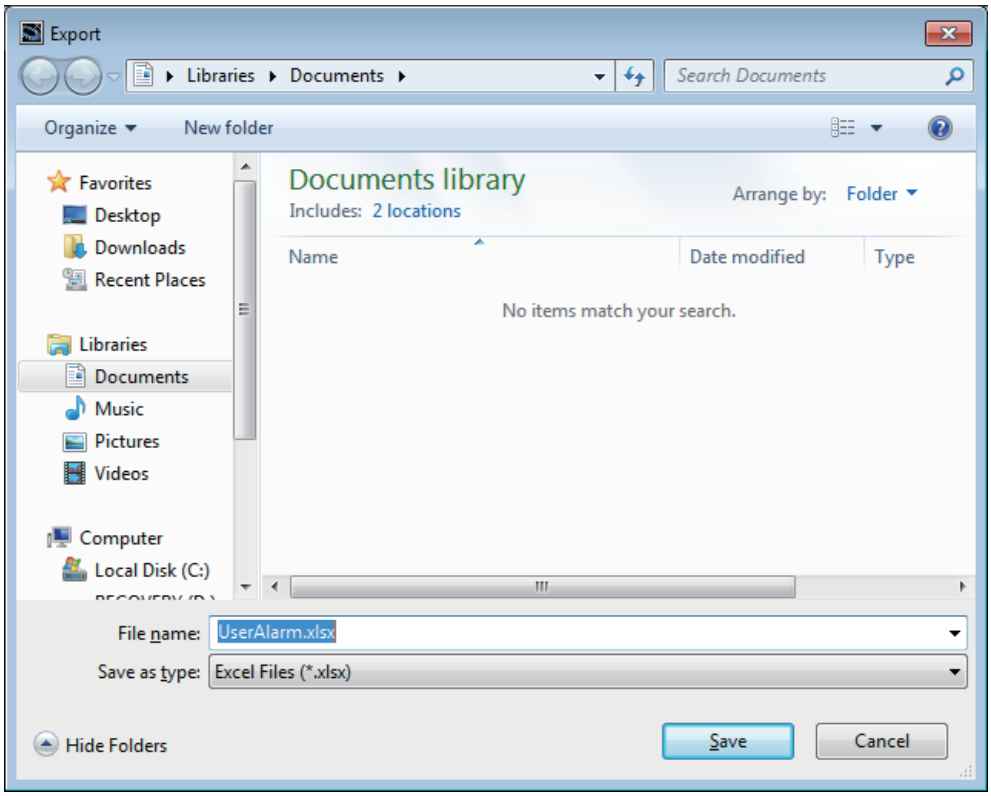
Exporting a Specific Group

1 Open the group you want to export.



2 Click .

3 Set a file name and click **Save**.



11-4-3 File Format

An Excel workbook is used to import or export user alarms.
The workbook contains the following sheets. Do not change the sheet name.

Sheet name	Editing	Description
UserAlarm	Yes	Set the user alarm description.
AlarmStrings	Yes	Set the alarm strings used in the UserAlarm sheet.
Information	No	Used by the system. Do not edit.

UserAlarm sheet

Set the user alarm information in the UserAlarm sheet. The first line is a header, and the 2nd and following lines contain the user alarm information. Do not edit the header line.

<table>
<tr><th>Line</th><th>Setting Item</th><th>Description</th></tr>
<tr><td>A</td><td>Group Name 1</td><td>1st group name.</td></tr>
<tr><td>B</td><td>Group Name 2</td><td>2nd group name.</td></tr>
<tr><td>C</td><td>Group Name 3</td><td>3rd group name.</td></tr>
<tr><td>D</td><td>Alarm ID</td><td>ID.</td></tr>
<tr><td>E</td><td>Alarm Code</td><td>Alarm code.</td></tr>
<tr><td>F</td><td>Expression</td><td>Expression.</td></tr>
<tr><td>G</td><td>Priority</td><td>Order of priority. The levels indicated in Sysmac Studio correspond to the levels indicated on the sheet as follows.
<table>
<tr><th>Sysmac Studio</th><th>Sheet</th></tr>
<tr><td>User Alarm Level 1</td><td>UserFaultLevel1</td></tr>
<tr><td>User Alarm Level 2</td><td>UserFaultLevel2</td></tr>
<tr><td>User Alarm Level 3</td><td>UserFaultLevel3</td></tr>
<tr><td>User Alarm Level 4</td><td>UserFaultLevel4</td></tr>
<tr><td>User Alarm Level 5</td><td>UserFaultLevel5</td></tr>
<tr><td>User Alarm Level 6</td><td>UserFaultLevel6</td></tr>
<tr><td>User Alarm Level 7</td><td>UserFaultLevel7</td></tr>
<tr><td>User Alarm Level 8</td><td>UserFaultLevel8</td></tr>
<tr><td>User Information</td><td>UserInformation</td></tr>
</table></td></tr>
<tr><td>H</td><td>Message</td><td>Message. Set the resource ID of the set alarm string in the message.</td></tr>
<tr><td>I</td><td>Popup</td><td>Popup. The levels indicated in Sysmac Studio correspond to the status indicated on the sheet as follows.
<table>
<tr><th>Sysmac Studio</th><th>Sheet</th></tr>
<tr><td>Selected</td><td>True</td></tr>
<tr><td>Not selected</td><td>False</td></tr>
</table></td></tr>
<tr><td>J</td><td>Acknowledge</td><td>Check. The levels indicated in Sysmac Studio correspond to the status indicated on the sheet as follows.
If anything other than False is set in the popup, this setting will be disregarded and always treated as if True were set.
<table>
<tr><th>Sysmac Studio</th><th>Sheet</th></tr>
<tr><td>Selected</td><td>True</td></tr>
<tr><td>Not selected</td><td>False</td></tr>
</table></td></tr>
<tr><td>K</td><td>Page</td><td>Page.</td></tr>
<tr><td>L</td><td>Details</td><td>Details. Set the resource ID of the alarm string set in Details.</td></tr>
</table>

Line	Setting Item	Description
M*1	DataType	Data type of user alarm properties. The levels indicated in Sysmac Studio correspond to the levels indicated on the sheet as follows. Sysmac Studio: Boolean → Sheet: Boolean Sysmac Studio: Numeric → Sheet: Numeric Sysmac Studio: Text → Sheet: Text
N*1	Variable	User alarm property variable.
O*1	ValueFormat	Property storage format of user alarm. The levels indicated in Sysmac Studio correspond to the levels indicated on the sheet as follows. Sysmac Studio: Decimal → Sheet: Decimal Sysmac Studio: Hexadecimal → Sheet: Hexadecimal
P*1	IntegerLength	MinimumIntegerLength of the user alarm property.
Q*1	DecimalLength	DecimalLength of the user alarm property.
R*1	ShowSeparator	ShowSeparator of the user alarm property. Sysmac Studio: Selected → Sheet: True Sysmac Studio: Not selected → Sheet: False
S*1	TextLength	TextLength of the user alarm property.

*1. Only appears in Runtime version 1.15 or later.

AlarmsStrings sheet

In the AlarmStrings sheet, enter the alarm strings used in the UserAlarm sheet. The first line is a header, and the 2nd and following lines contain the alarm string. Do not edit the header line. The AlarmStrings sheet is output based on the UserAlarm sheet, so some of the content may duplicate previous content. When imported, the lowest duplicate content on the sheet is given priority.

The information entered in each line is described below. Lines A and B are for the purpose of identifying where a string is used during editing. These are disregarded when imported.

Line	Setting Item	Description
A	Alarm ID	Alarm ID on UserAlarm sheet.
B	Type	Indicates whether an alarm is used in Message or Details on the UserAlarm sheet.
C	Resource Group Name	Group name of resource. For the root that exists by default, enter [root].
D	Resource ID	Resource ID of alarm string.
E and following	Language name Culture	Alarm string of each language.

Information sheet

The information sheet is used by the system. Do not edit.

11-5 Import/Export Resources

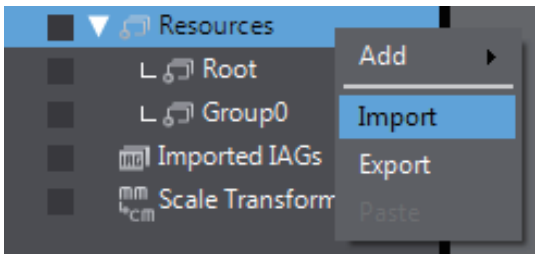
This section describes importing and exporting of resources. Only general strings and alarm strings can be imported and exported.

11-5-1 Importing Resources

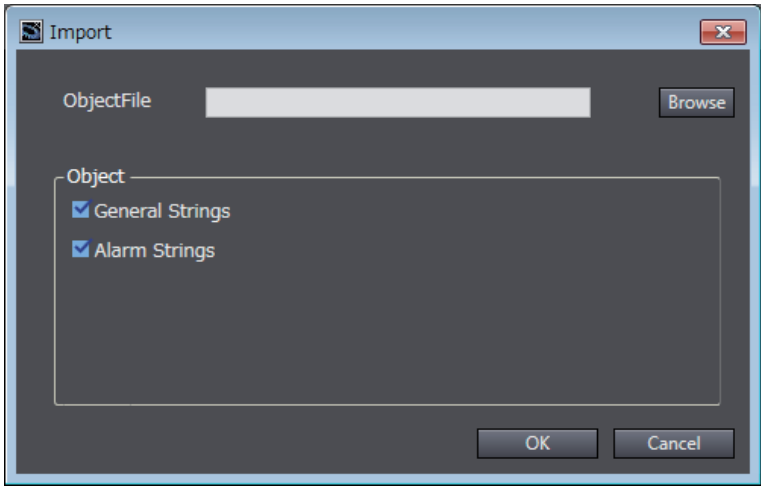
You can import all resources at once, or import only a specific group of resources.

Importing All Resources at Once

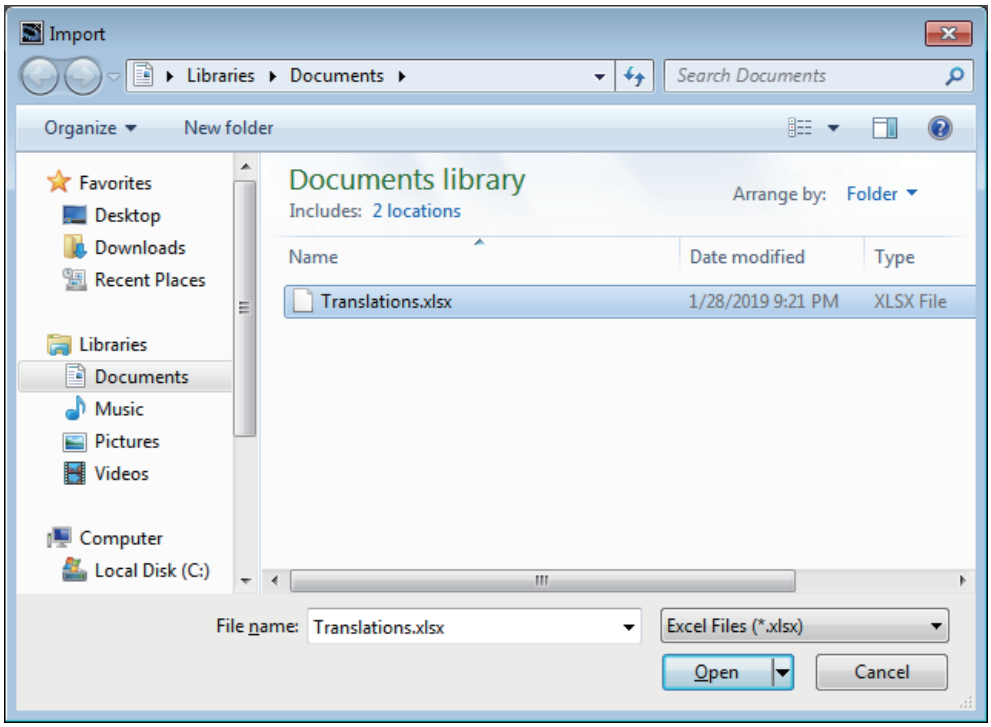
1 Right click a resource. Select **Import** from the menu.



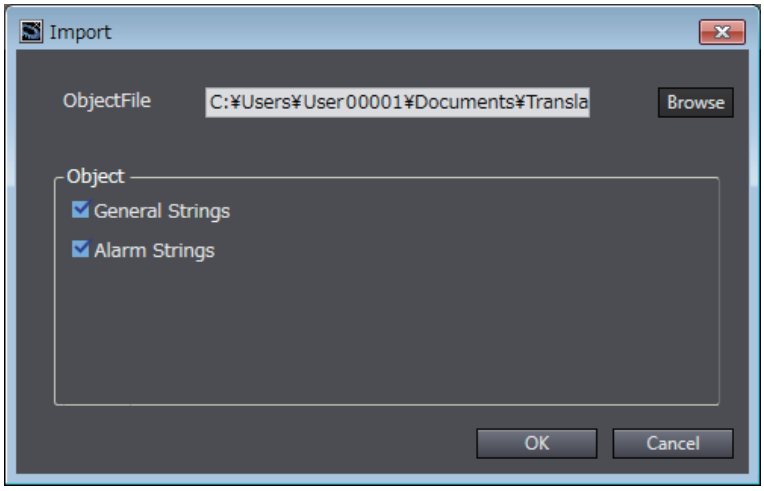
2 Click **Browse**.



3 Select the file and click **Open**.

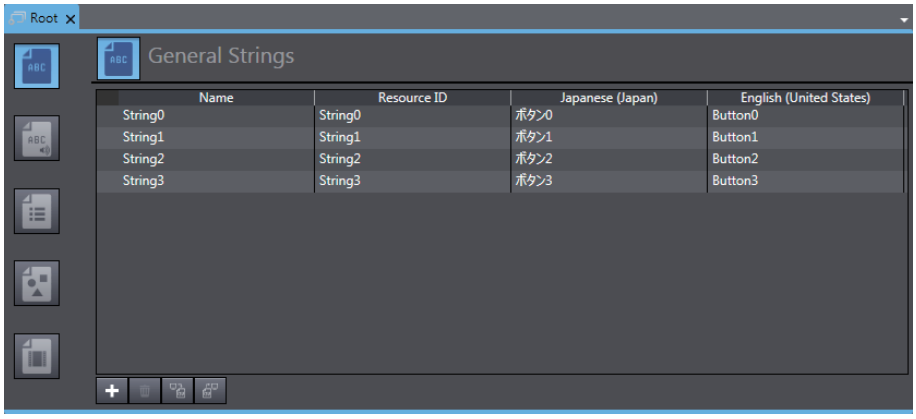


4 Select the resources you want to import, and click **OK**.



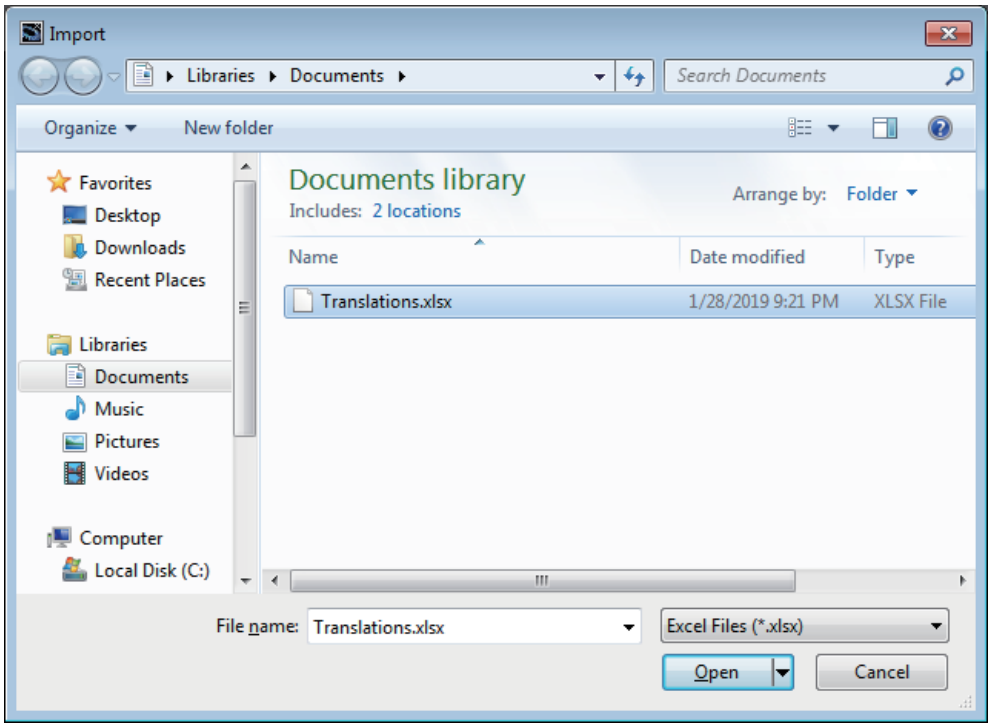
Importing a Specific Group

1 Open the group you want to import.



2 Click .

3 Select the file and click **Open**.

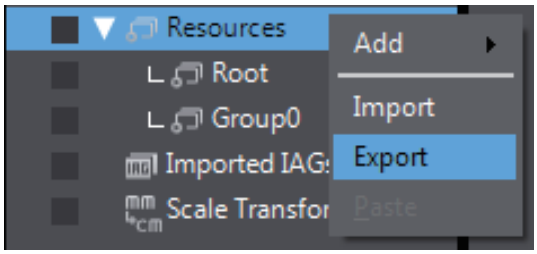


11-5-2 Exporting Resources

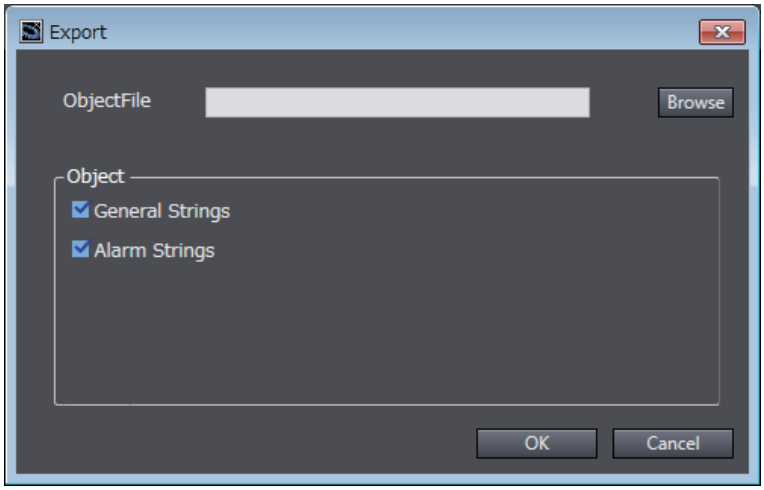
You can export all resources at once, or export only a specific group of resources.

Exporting All Resources at Once

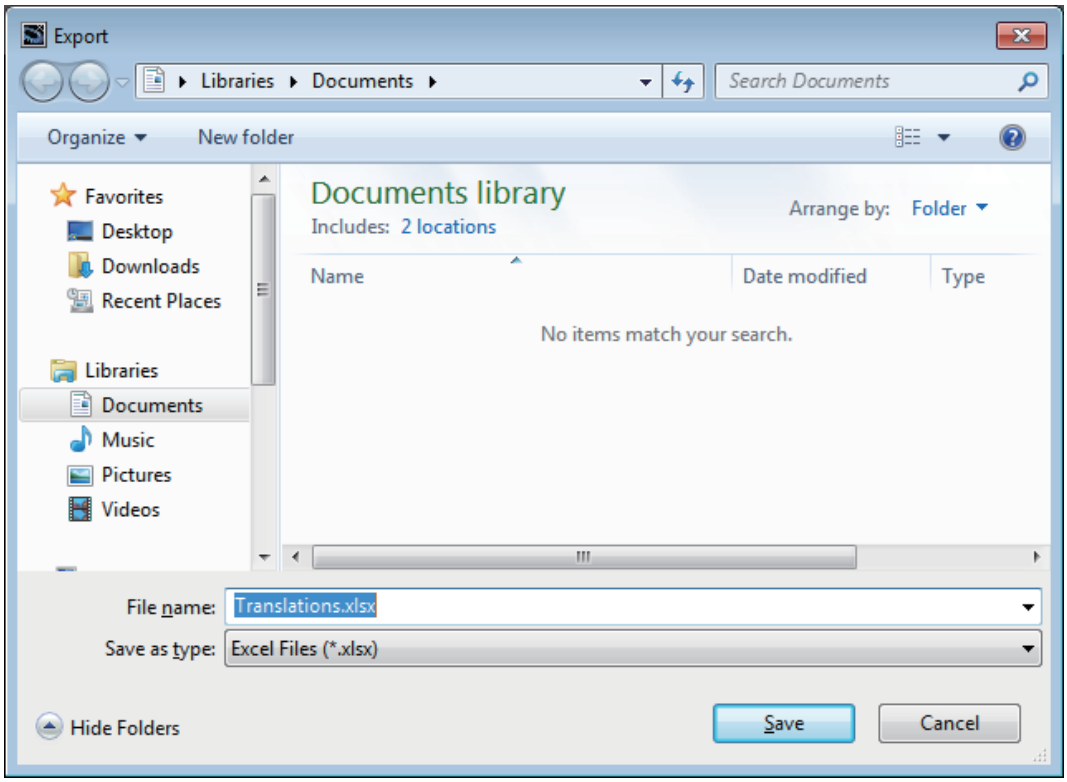
1 Right click a resource. Select **Export** from the menu.



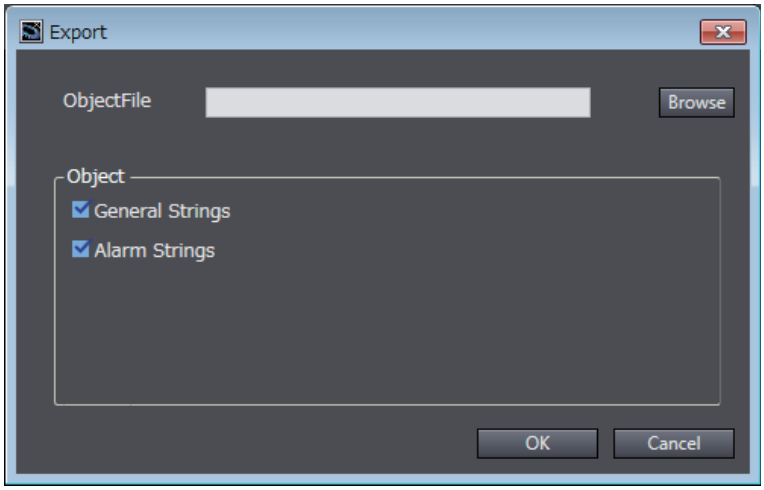
2 Click **Browse**.



3 Set a file name and click **Save**.

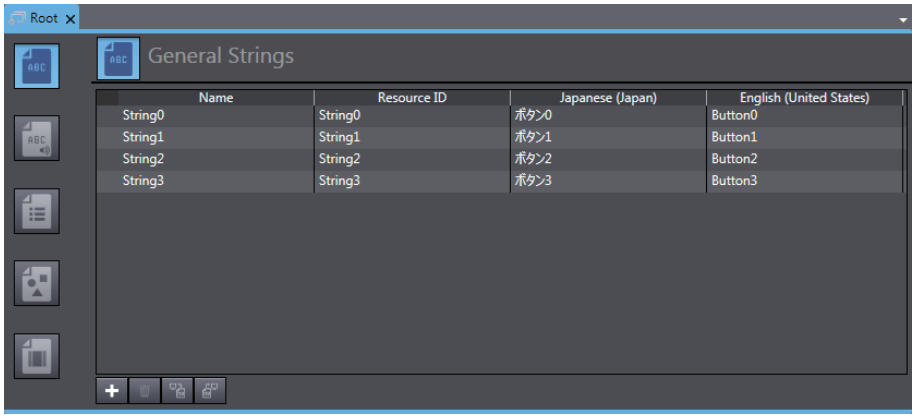


4 Select the resources you want to export, and click **OK**.



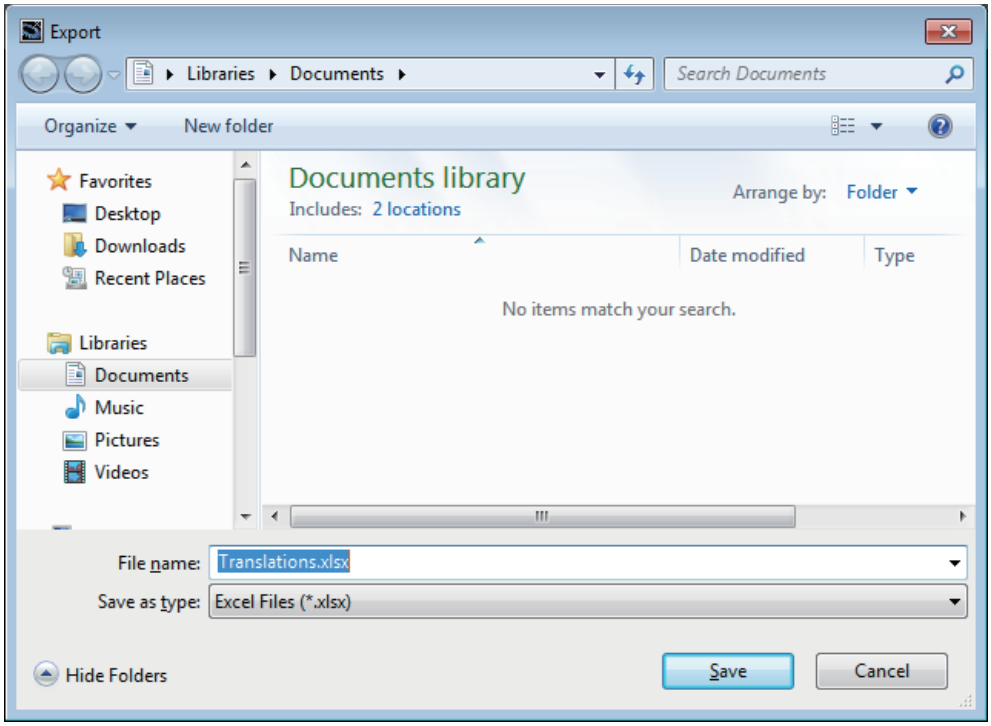
Exporting a Specific Group

1 Open the group you want to export.



2 Click .

3 Set a file name and click **Save**.



11-5-3 File Format

An Excel workbook is used to import or export resources.

The workbook contains the following sheets. Do not change the sheet name.

Sheet name	Editing	Description
GeneralStrings	Yes	Set general strings.
AlarmStrings	Yes	Set alarm strings.
Information	No	Used by the system. Do not edit.

GeneralStrings sheet

Set general strings in the GeneralStrings sheet. The first line is a header, and the 2nd and following lines contain the general strings. Do not edit the header line.

Line	Setting Item	Description
A	Group Name	Group name. For the root that exists by default, enter [root].
B	Resource ID	Resource ID.
C and following	Language name [Culture]	General strings of each language.

AlarmsStrings sheet

Set alarm strings in the AlarmStrings sheet. The first line is a header, and the 2nd and following lines contain the user alarm information. Do not edit the header line.

Line	Setting Item	Description
A	Group Name	Group name. For the root that exists by default, enter [root].
B	Resource ID	Resource ID.
C and following	Language name [Culture]	Alarm string of each language.

Information sheet

The information sheet is used by the system. Do not edit.

11-6 Import/Export Object Properties

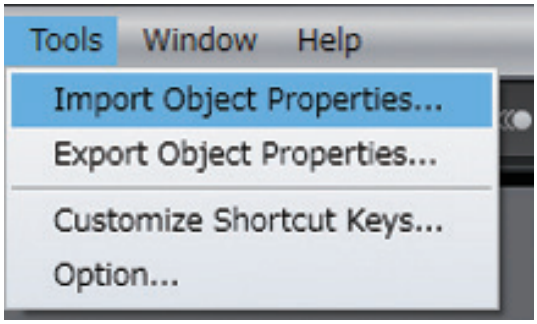
This section describes importing and exporting of object properties. Only text, variables, and expressions of some objects can be imported and exported. Use this when using the same project to create a derivative project, with changes to only the text and variable assignments.

11-6-1 Importing Object Properties

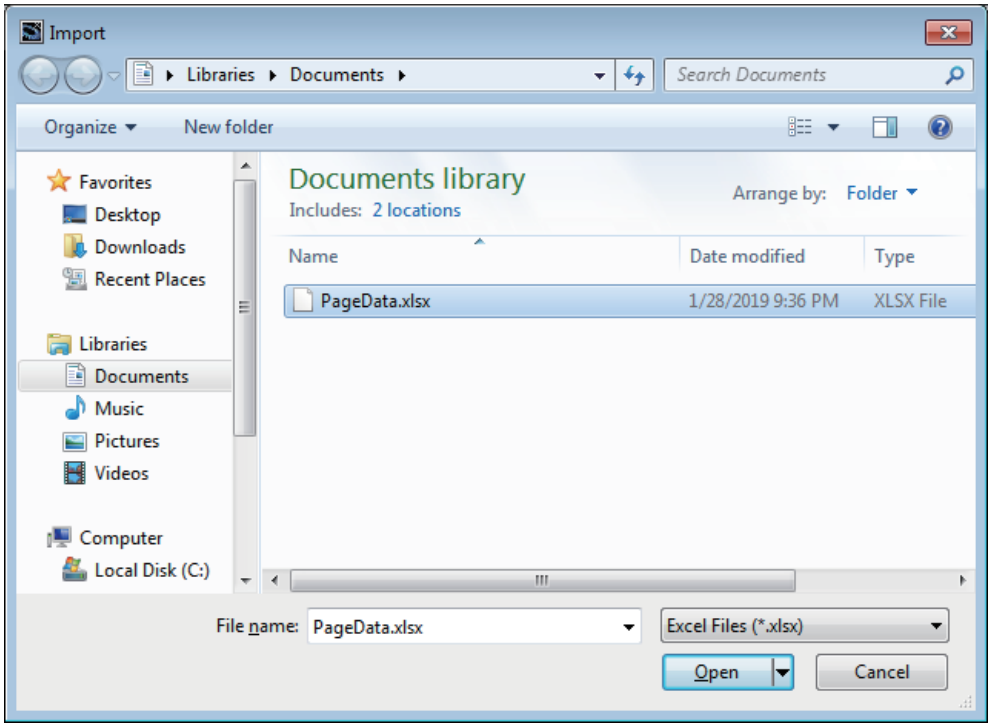
This section explains how to import object properties.

When the same settings already exist, they are overwritten. Settings that do not exist in the workbook retain the state they had before importing.

1 Select **Tool** - **Import Object Properties** from the menu.



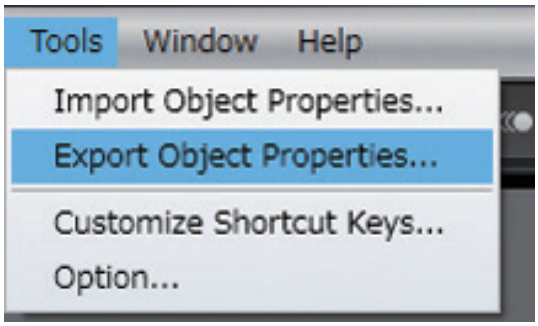
2 Select the file and click **Open**.



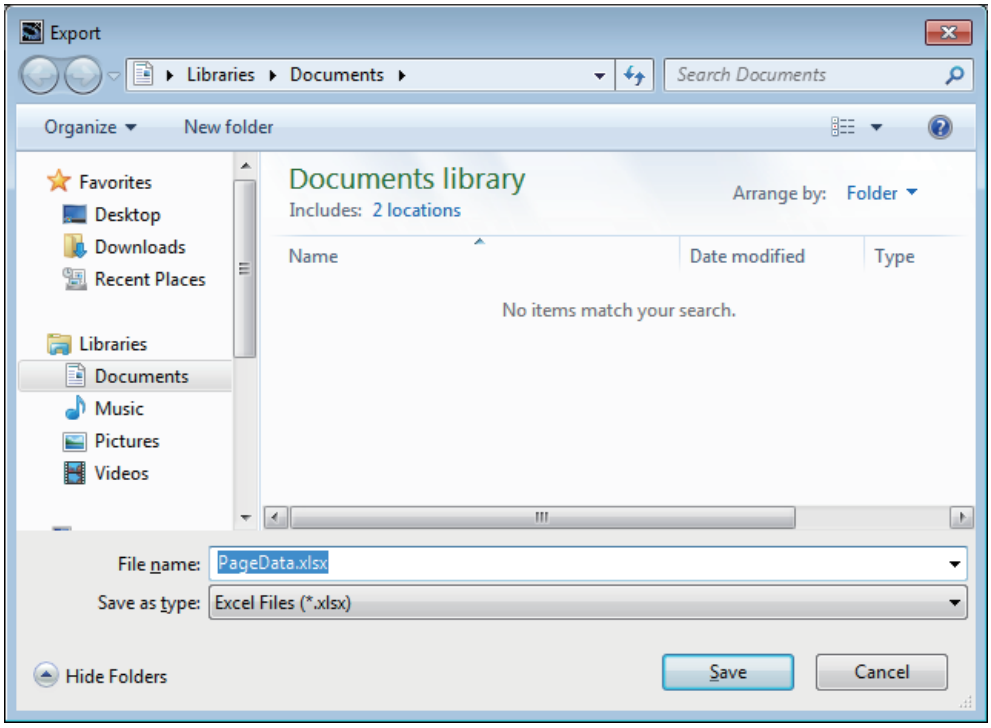
11-6-2 Exporting Object Properties

This section explains how to export object properties.

1 Select **Tool** - **Export Object Properties** from the menu.



2 Set a file name and click **Save**.



11-6-3 File Format

An Excel workbook is used to import or export object properties. The workbook contains the following sheets. Do not change the sheet name.

Sheet name	Editing	Description
Settings	Yes	Set the properties of the object.
GeneralStrings	Yes	Set general strings that are used in the Settings sheet.
Information	No	Used by the system. Do not edit.

Settings sheet

In the Setting sheet, set the properties of the object placed on the page. The first line is a header, and the 2nd and following lines contain the object properties. Do not edit the header line.

Line	Setting Item	Description
A	Page Name	Name of the page.
B	Object Name	Name of the object. Objects included in IAG have the following names: IAG object name.object name
C	Property Name	Name of the property.
D	Setting	Property setting. For text, set the resource ID. For a variable or an expression, set the variable or expression.

Additional Information

When editing texts allocated to objects which are included in IAG, note the following.

IAG resources are allocated by default to the texts allocated to the objects included to IAG. They are displayed in the Setting column as resource IDs like [iag]XXX. To edit text of an object included in a specific IAG object, select a normal resource which does not have [iag] on its head. IAG resources of the corresponding IAG and normal resources are applicable to use within IAG. IAG resources for other IAGs are not applicable.

The correspondences between the property type of the corresponding object and the character string set in Property Name are described below. Objects included in IAG are excluded.

Object Type	Property Name in Sysmac Studio	Property Name on sheet
Button	General - Type	ResourceKey
	Appearance - Text (Default) - Resource ID	Text
Set Button	General - Type	ResourceKey
	Appearance - TextButtonUp (Default) - Resource ID	Text
	Appearance - TextButtonDown (Default) - Resource ID	TextOn
	Behavior - Variable	Variable*1
	Behavior - Feedback Expression	FeedbackExpression*1
Toggle Button	General - Type	ResourceKey
	Appearance - TextButtonUp (Default) - Resource ID	Text
	Appearance - TextButtonDown (Default) - Resource ID	TextOn
	Behavior - Variable	Variable*1
	Behavior - Feedback Expression	FeedbackExpression*1

Object Type	Property Name in Sysmac Studio	Property Name on sheet
Momentary Button	General - Type	ResourceKey
	Appearance - TextButtonUp (Default) - Resource ID	Text
	Appearance - TextButtonDown (Default) - Resource ID	TextOn
	Behavior - Variable	Variable*1
	Behavior - Feedback Expression	FeedbackExpression*1
Reset Button	General - Type	ResourceKey
	Appearance - TextButtonUp (Default) - Resource ID	Text
	Appearance - TextButtonDown (Default) - Resource ID	TextOn
	Behavior - Variable	Variable*1
	Behavior - Feedback Expression	FeedbackExpression*1
Bit Lamp	General - Type	ResourceKey
	Appearance - TextOff (Default) - Resource ID	Text
	Appearance - TextOn (Default) - Resource ID	TextOn
	Behavior - Expression	Expression*1
Data Lamp	General - Type	ResourceKey
	Appearance - DefaultText (Default) - Resource ID	Text
	Behavior - Expression	Expression*1
	Behavior - ColorRanges - [*] - Text (Default)	colorRange*.Text*1
Text Box	General - Type	ResourceKey
	Appearance - Text (Default) - Resource ID	Text
	Appearance - Text (Default) - Expression	Expression
Label	General - Type	ResourceKey
	Appearance - Text (Default) - Resource ID	Text
	Appearance - Text (Default) - Expression	Expression
IAG	General - Type	ResourceKey
	Behavior(Input) - User-Defined Input Variable	Input.User-Defined Input Variable
	Behavior(In/Out) - InOut.User-Defined I/O Variable	InOut.User-Defined I/O Variable

*1. Objects included in IAG are not output.

GeneralStrings sheet

Set general strings in the GeneralStrings sheet. The first line is a header, and the 2nd and following lines contain the general strings. Do not edit the header line. This is output based on the Settings sheet, so some of the content may duplicate previous content. When imported, the lowest duplicate content on the sheet is given priority.

The information entered in each line is described below. Lines A to C are for the purpose of identifying where a string is used during editing. These are disregarded when imported.

Line	Setting Item	Description
A	Page Name	Name of the page.
B	Object Name	Name of the object.
C	Property Name	Name of the property.
D	Resource Group Name	Name of resource group. For the root that exists by default, enter [root]. For an IAG resource, the name is [iag] IAG collection name.IAG name,{GUID}.
E	Resource ID	Resource ID.
F and following	Language name Culture	General strings of each language.

Information sheet

The information sheet is used by the system. Do not edit.

11-7 Importing/Exporting Pages

You can import/export pages often used by specifying each page.

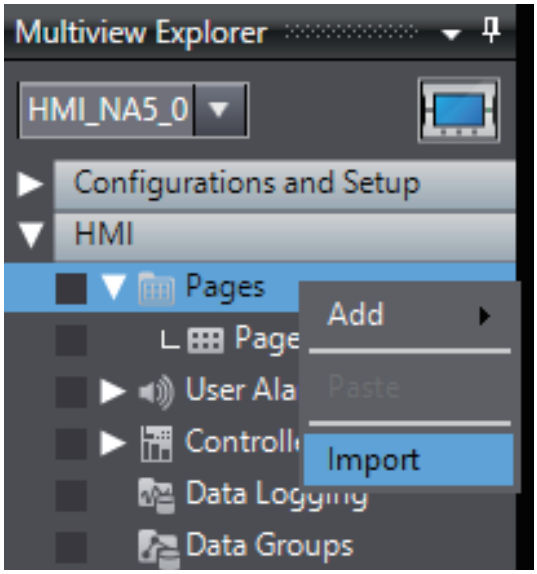
11-7-1 Importing Pages

Use the following procedure to import a page. When Import is executed, variables as well as resources used in the page are also imported in the same way as Paste.

Additional Information

An Import can be only executed when the version of Runtime on the device to execute Import is later than that of the version of Runtime used when the data to be imported was created.

1 Right-click on Page Group. Select **Import** from the menu.



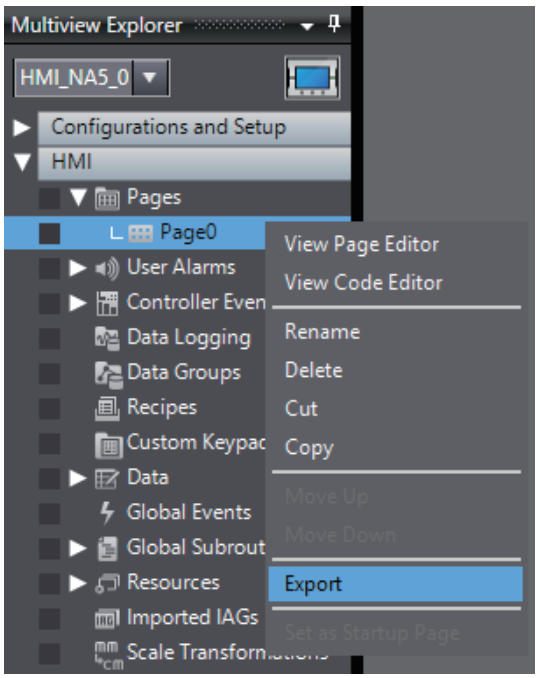
2 Select the file to import and click **Open**.

3 The page is imported.

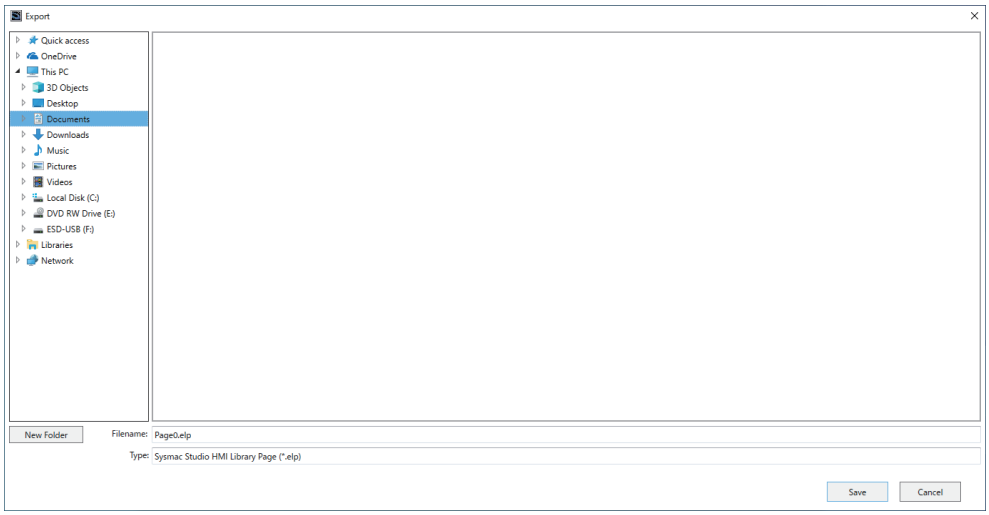
11-7-2 Exporting Pages

Use the following procedure to export pages. When Export is executed, variables as well as resources used in the page are also exported in the same way as Copy.

1 Right-click on Page. Select **Export** from the menu.



2 Select the folder to export the page to, specify the file name, and click **Save**.



3 The page is exported.

A

Appendices

The appendices provide information on supported file formats and other information.

A-1 Events and Actions A-2
A-2 Supported Formats A-5
A-3 Specifications of User Alarm Log Files A-7
A-4 Specifications of Data Log Files A-8
A-5 Specifications of Operation Log Files A-9
A-6 Differences of NA5-□□W□□□□-V2 . . . . . A-12
A-7 Differences between the Physical HMI and Simulator A-15
A-8 Version Upgrade History A-16
A-8-1 Version Upgrade History for Sysmac Studio and Runtime A-16
A-8-2 Version Upgrade History for Sysmac Studio Only A-20
A-8-3 Sysmac Studio Corresponding Versions A-24
A-8-4 Runtime Support Limit A-25
A-9 Precautions for Version Upgrades A-26

A-1 Events and Actions

Events

Name	Description	Global events	Page	Objects	User alarms	IAG
Acknowledged	The event occurs when the user alarm is acknowledged.	—	—	—	Yes	—
Checked	The event occurs when the check box is selected.	—	—	Yes	—	—
Cleared	The event occurs when the user alarm is cleared.	—	—	—	Yes	—
Click*1	The event occurs when the object is tapped.	—	—	Yes	—	—
Condition	The event occurs when the set condition is met.	Yes	Yes	—	—	Yes
F1 Key Click*1 *2	The event occurs when the **F1** Key is tapped.	Yes	Yes*3	—	—	—
F1 Key Press*2	The event occurs while the **F1** Key is held down.	Yes	Yes*3	—	—	—
F1 Key Release*1 *2	The event occurs when the **F1** Key is released.	Yes	Yes*3	—	—	—
F2 Key Click*1 *2	The event occurs when the **F2** Key is tapped.	Yes	Yes*3	—	—	—
F2 Key Press*2	The event occurs while the **F2** Key is held down.	Yes	Yes*3	—	—	—
F2 Key Release*1 *2	The event occurs when the **F2** Key is released.	Yes	Yes*3	—	—	—
F3 Key Click*1 *2	The event occurs when the **F3** Key is tapped.	Yes	Yes*3	—	—	—
F3 Key Press*2	The event occurs while the **F3** Key is held down.	Yes	Yes*3	—	—	—
F3 Key Release*1 *2	The event occurs when the **F3** Key is released.	Yes	Yes*3	—	—	—
Interval	The event occurs at the specified interval.	Yes	—	—	—	—
Page Displayed	The event occurs while the page is displayed.	—	Yes	—	—	—
Page Hidden	The event occurs when the page is hidden.	—	Yes	—	—	—
Press	The event occurs while the object is held down.	—	—	Yes	—	—
Project Initialization*4	The event occurs when the project is initialized.	Yes	—	—	—	—
Raised	The event occurs when the user alarm occurs.	—	—	—	Yes	—
Release*1	The event occurs when the object is released.	—	—	Yes	—	—
Selection Changed	The event occurs when the item selected in the list changes.	—	—	Yes	—	—
Unchecked	The event occurs when the check box selection is cleared.	—	—	Yes	—	—

*1. For both *Click* and *Release*, the event occurs when the object is released, but the operation when the page is changed is different. If the page changes when an object set for *Click* is touched but not yet released, the event does not occur. If the page changes when an object set for *Release* is touched but not yet released, the event does occur.

*2. The event is enabled only while a user-created page is displayed. The event is disabled while functions built into the system such as the System Menu are operating.

*3. These events are ignored for popup pages.

*4. The results of accessing external variables during project initialization immediately after startup are not always dependable. Do not access external variables during project initialization.

Actions

Action	Description	Global events	Page	Objects	User alarms	IAG
BuzzerOff	Turns OFF the buzzer.	Yes	Yes	Yes	Yes	Yes
BuzzerOn	Turns ON the buzzer.	Yes	Yes	Yes	Yes	Yes
BuzzerOneShot *1	Turns ON the buzzer that beeps for only 0.25 second.	Yes	Yes	Yes	Yes	Yes
CallSubroutine	Executes a subroutine registered as a global subroutine or page subroutine.	Yes*2	Yes	Yes	Yes*2	Yes
ClearUserAlarmLog	Clears the user alarm log.	Yes	Yes	Yes	Yes	Yes
ClosePage	Closes the specified page.	Yes	Yes	Yes	Yes	Yes
DecreaseVariable	Subtracts the specified value from the specified variable.	Yes	Yes	Yes	Yes	Yes
EjectSDMemory	Enables removing the SD Memory Card.	Yes	Yes	Yes	Yes	Yes
EnableInputOperation	Enables or disables inputs on the touch panel.	Yes	Yes	Yes	Yes	Yes
EndSyncLock*3	Ends a critical section.	Yes	---	---	---	---
IncreaseVariable	Adds the specified value to the specified variable.	Yes	Yes	Yes	Yes	Yes
InvertVariable	Inverts the value of the specified Boolean variable.	Yes	Yes	Yes	Yes	Yes
Login	Displays the login page.	Yes	Yes	Yes	Yes	Yes
Logout	Logs out the user.	Yes	Yes	Yes	Yes	Yes
ResetVariable	Changes the value of the specified Boolean variable to False.	Yes	Yes	Yes	Yes	Yes
SaveScreenshot	Captures the screen displayed on the HMI.	Yes	Yes	Yes	Yes	Yes
SaveUserAlarmLog-ToFile	Saves the user alarm log to a file.	Yes	Yes	Yes	Yes	Yes
SetIMEType	Changes the IME language settings.	Yes	Yes	Yes	Yes	Yes
SetInputFocus	Sets the data input focus.	---	Yes	Yes	---	Yes
SetLanguage	Changes the project language settings.*4	Yes	Yes	Yes	Yes	Yes
SetVariable	Sets the value of the specified variable to a specified value.	Yes	Yes	Yes	Yes	Yes
ShowAlarmPage	Displays the page set in a specified User Alarm.	Yes	Yes	Yes	Yes	---
ShowDocument(FULL SCREEN)	Displays a PDF or other file full screen.	Yes	Yes	Yes	Yes	Yes
ShowDocument(Window)	Displays a PDF or other file in a window.	Yes	Yes	Yes	Yes	Yes
ShowPage*5	Displays a page.	Yes	Yes	Yes	Yes	Yes
ShowPreviousPage*5	Displays the most recently displayed page.	Yes	Yes	Yes	Yes	Yes
ShowSystemMenu	Displays the System Menu.	Yes	Yes	Yes	Yes	Yes
ShowTroubleshooter	Displays the NJ/NX/NY Troubleshooter.	Yes	Yes	Yes	Yes	---

Action	Description	Global events	Page	Objects	User alarms	IAG
StartDataLogging	Starts data logging.	Yes	Yes	Yes	Yes	Yes
StartSafetyMonitor	Displays the Safety Monitor.	Yes	Yes	Yes	Yes	Yes
StopDataLogging	Stops data logging.	Yes	Yes	Yes	Yes	Yes
ShowOperation-LogViewer	Displays the Operation Log Viewer.	Yes	Yes	Yes	Yes	Yes
StartOperationLog	Starts operation logging.	Yes	Yes	Yes	Yes	Yes
StopOperationLog	Stops operation logging.	Yes	Yes	Yes	Yes	Yes
SyncLock*3	Starts a critical section.	Yes	---	---	---	---
TriggerEvent	Sets the UserEvents for IAG objects.	---	---	---	---	Yes

*1. When multiple actions attempt to simultaneously beep buzzers, the last buzzer demand overwrites the previous demands.

*2. You cannot execute page subroutines for global events or user alarms.

*3. The SyncLock and EndSyncLock actions must be specified as a pair.

*4. Asynchronous execution is not supported.

*5. If you execute an action with a Momentary Button, use a Click or Release event.

Additional Information

If the same variable might be accessed simultaneously from different locations, such as for simultaneous execution of subroutines, perform exclusive control to prevent simultaneous access.

For a section where exclusive control is required, specify the start of the section with the SyncLock action and the end of the section with the EndSyncLock action. This defines the section between these two actions as a critical section. Until the execution of the critical section is completed, other processing will not be executed, therefore achieving exclusive control.

A-2 Supported Formats

The HMI objects support the following formats. However, it may not be possible to display some files even if the file format is supported.

Additional Information

Use images and videos that are roughly the same size as the object. Using extremely large size images or videos may result in improper functioning. Note that image files that are 2.9 MB or larger in size cannot be registered.

Image Files

The following formats are supported.

Format name	Extension	Specifications
Microsoft Windows Bitmap Image	BMP	1, 4, 8, 16, 24, or 32-bit Uncompressed or RLE compression
Graphics Interchange Format	GIF	1, 4, or 8-bit Interlaced GIF is supported. Transparent color (Runtime version 1.08 or later) is supported. Animation GIFs are displayed as still images.
JFIF or EXIF	JPG	8-bit gray scale 24-bit/basic DCT or progressive DCT
Portable Network Graphics	PNG	1, 4, 8, or 24-bit Transparency is supported only when the α channel is saved for each pixel.
Scalable Vector Graphics	SVG	SVG 1.0 is supported.
Extensible Application Markup Language	XAML	
Drawing Exchange Format	DXF	R12 and older are supported.

Video

The following formats are supported. However, only progressive formats are supported. The maximum size of the file that can be registered in a project in Sysmac Studio is 100 MB.

Format name	Extension	Specifications	Maximum resolution
MPEG-1	mpg	The MPEG-1 Video Stream format is not supported.	768×480
MPEG-2	mpg	Main Profile Low, Main, and High 1440 MPEG-2 TS is not supported.	1280×720
MPEG-4 Part 2*1	mp4	Simple Profile L0, L1, L2, and L3 Advanced Simple Profile L0, L1, L2, L3, L4, and L5 Global motion compensation is not supported.	1280×720
MPEG-4 Part 10 (H.264)	mp4	Baseline Profile L1, L1.2, L1.3, L2, L2.2, and L3 Main Profile L1, L1.2, L1.3, L2, L2.2, L3, L3.1, L3.2, and L4.1 High Profile L1, L1.2, L1.3, L2, L2.2, L3, L3.1, L3.2, and L4.1	1280×720
Windows Media Video	wmv	WMV9	240×160

*1. Not supported on the NA5-□□W□□□□-V1.

Files Supported by Document Viewer

The Document Viewer can display the following files. The maximum size of file that can be registered in a project in Sysmac Studio is 16 MB.

Format name	Extension	Specifications
Adobe Acrobat Document	PDF	PDF 1.5 and later are not supported.
Microsoft Excel Books*1	xlsx	
Microsoft Excel 97-2003 Books*1	xls	
Microsoft Word Documents*1	docx	
Microsoft Word 97-2003 Documents*1	doc	

*1. Not supported on the NA5-□□W□□□□-V2.

A-3 Specifications of User Alarm Log Files

The specifications of user alarm log files are shown below.

Specifications of Files

File format: CSV file
Encoding method: UTF-8 (with BOM (Runtime version 1.15 or later), without BOM (Runtime version 1.14 or earlier))
Delimiter character: , (0x2c)
File name: File name specified by the user in **SaveUserAlarmLogToFile** of the action or at SaveUserAlarmLogToFile

Recording Specifications

Recording specifications are shown below.

Column	Item	Description
1	Date and Time	The last update date/time (UTC) of a record is output in the date/time format in accordance with the currently selected project language.
2	Name	The name of the user alarm is output.
3	Alarm Code	The alarm code of the user alarm is output.
4	Message	The message of the user alarm is output.
5	Group Display Name	The group display name of the user alarm is output.
6	Order of Priority	The order of priority of the user alarm is output.
7	Status	The status at the time the user alarm was saved is output.
8	Logged-in User	The name of the currently logged-in user at the time the user alarm was saved is output.
9	Additional Information	The additional information of the user alarm is output.

A-4 Specifications of Data Log Files

The specifications of data log files are shown below.

Specifications of Files

File format: CSV file
Encoding method: UTF-8 (with BOM)
Delimiter character: , (0x2c)
File name: File name naming rules are shown below.

\Data Logging\Log Files\[data set name]\[file generation year/month/day]\[file generation hour/minute]_[index].csv

- [file generation year/month/day]: YYYYMMDD
- [file generation hour/minute]: hhmm*1 *2
- [index]: 3-digit serial number (default: 000) Duplicate values are incremented by “1”.

*1. Fixed at “0000” when the new data log file generation interval is **Daily**.

*2. The mm section is fixed at “00” when **Hours** is selected for the unit when the new data log file generation interval is **After specific time period**.

Recording Specifications

Recording specifications are shown below. The first line is a header and the data is recorded in the second and following lines.

The number of columns from the second column onwards is variable depending on the contents of the data set.

Column	Item	Description
1	Date/time that the record was recorded	The date/time (UTC) that the record was recorded is output. The output format is fixed at YYYY-MM-DD hh:mm:ss.fff.
2	Value of 1st variable	The value of the 1st variable in the corresponding data set is output.
3	Value of 2nd variable	The value of the 2nd variable in the corresponding data set is output.
• • •	• • •	• • •
n	Value of nth variable	The value of the nth variable in the corresponding data set is output.

A-5 Specifications of Operation Log Files

Specifications of operation log file are shown below.

Specifications of files

File format: CSV file
Encoding method: UTF-8 (with BOM)
Delimiter character: , (0x2c)
File name: File name specified at SaveOperationLogToFile by the user

Recording specifications

Recording specifications are shown below. The first line is a header and the data in the second and following lines are recorded.

Information entered in each column differs according to the recording operation.

Column	Item	Description
1	Date and Time	Outputs the time the operation occurred in UTC. The output format according to Locale is used.
2	User	Outputs the user account that was logged in when the operation was executed.
3	Category	Outputs the operation category.
4	Operation	Outputs the name of the operation.
5	Operation Information1	Outputs detailed information of the operation.
6	Operation Information2	
7	Page	
8	Action	Outputs the action name.
9	Action Information1	Outputs detailed information of the action.
10	Action Information2	
11	Value before	
12	Value after	

Output specifications of each operation

Columns 1 to 7 output the description that correspond with each operation. The output specifications are shown below.

Operation	Date and Time	User	Category	Operation Information1	Operation Information2	Page
Startup	Date and Time	User Accounts	System			
Exit						
Start transfer/synchronisation				Transfer type		
Finish transfer/synchronisation						
Execute Project Synchronization						
Show System Menu						
Exit System Menu						
Show Troubleshooter				Controller name	Page name	
Exit Troubleshooter						
Show Safety Monitor						
Exit Safety Monitor						
Show Operation Log Viewer						
Exit Operation Log Viewer						
Disable touch input						
Enable touch input						
Start screensaver						
End screensaver						
Connect to VNC server				IP address		
Disconnect VNC server				IP address		
Connect to FTP server				IP address		
Disconnect FTP server				IP address		
Connect with FTP Client				Host	User name	
Disconnect with FTP Client				Host	User name	
Change System settings				Setting item name		
Login			User	User name	Login result	
Logout						
Switch user				User name	Login result	
Set language*1			Locale			
IME change*1						
Write recipe			Recipe	Template name	Recipe name	
Read recipe				Template name	Recipe name	
Start Data logging			Data Logging	Data set name		
Stop Data logging				Data set name		
Start Operation logging			Operation Log			
Stop Operation logging						
Object operation			User Operation	Object name	Name	Page name
Function Key operation				Function Key name	Name	Page name

*1. A value only for *Set language* and *IME change* is output to Value before and Value after in columns 11 to 12.

Operation	Value before	Value after
Set language	Languages before change	Languages after change
IME change	Type before change	Type after change

Output specifications of action-related columns

Values that correspond to actions associated with the operation are recorded in columns 8 to 12. The output specifications are shown below.

Action executed	Action	Action Information1	Action Information2	Value before	Value after
BuzzerOff	BuzzerOff				
BuzzerOn	BuzzerOn				
BuzzerOneShot	BuzzerOneShot				
Subroutines	Subroutine name				
ClearUserAlarmLog	ClearUserAlarmLog				
ClosePage	ClosePage	Page name			
DecreaseVariable	DecreaseVariable	Variable name		Value before	Value after
EjectSDMemory	EjectSDMemory				
EnableInputOperation	EnableInputOperation				
IncreaseVariable	IncreaseVariable	Variable name		Value before	Value after
InvertVariable	InvertVariable	Variable name		Value before	Value after
Login	Login				
Logout	Logout				
ResetVariable	ResetVariable	Variable name		Value before	Value after
SaveScreenshot	SaveScreenshot				
SaveUserAlarmLogToFile	SaveUserAlarmLogToFile				
SetIMEType	SetIMEType				
SetInputFocus	SetInputFocus				
SetLanguage	SetLanguage				
SetVariable	SetVariable	Variable name		Value before	Value after
ShowPage (user alarm)	ShowPage (user alarm)	Page name	Viewer name		
ShowDocument (FULL SCREEN)	ShowDocument (FULL SCREEN)	Document name			
ShowDocument (Window)	ShowDocument (Window)	Document name			
ShowPage	ShowPage	Page name			
ShowPreviousPage	ShowPreviousPage				
ShowSystemMenu	ShowSystemMenu				
ShowTroubleshooter	ShowTroubleshooter				
StartSafetyMonitor	StartSafetyMonitor				
ShowOperationLogViewer	ShowOperationLogViewer				
StartDataLogging	StartDataLogging				
StopDataLogging	StopDataLogging				
StartOperationLogging	StartOperationLogging				
StopOperationLogging	StopOperationLogging				
Change value using the keypad	EditVariable				
Selection changed using the radio button					
Selection changed in a drop button/list box					
Change tab page					
State changed in a toggle button/check box					
Value changed using a slider					

A-6 Differences of NA5-□□W□□□□-V2

The main differences between NA5-□□W□□□□-V2 and other models are as follows. For details, refer to *Replace Guide From NA5-V1 to V2* (V472).

- Folder name delimiter

  There are differences in the folder name delimiter. You may specify either one for the action parameter or the function argument provided by OMRON. For other functions, use appropriate delimiters according to the model.

  NA5-□□W□□□□-V2: / (0x2f)

  Other models: \ (0x5c)

- Operations according to project language

  The currency symbols, date display format, etc., will be changed according to the project language, however, there are operation specification differences in some project languages.

- Floating point precision

  NA5-□□W□□□□-V2 specifications are in stricter compliance with IEEE754. This means that differences may occur in values after the decimal point.

- Supported file formats

  Some file formats are not supported. For details, refer to *A-2 Supported Formats*.

- Objects

  There are minor differences in how objects appear and text string spacing.

- Simulator

  Simulator display specifications are the same as NA5-□□W□□□□-V1. For this reason, objects may appear different than the NA Unit.

- Fonts

  Available fonts differ. A correspondence table is given below: If the model has been converted, the font will be automatically replaced based on the correspondence table.

NA5-□□W□□□□ **NA5-□□W□□□□-V1** **Soft-NA**	**NA5-□□W□□□□-V2**
Arial	Liberation Sans
Arial Black	Liberation Sans Black
Arial Narrow	Liberation Sans Narrow
Comic Sans MS	Omron NA Comic Sans
Courier New	Omron NA Mono
DF7segHMI	DF7segHMI
Ebrima	Omron NA Sans African
Estrangelo Edessa	Omron NA Sans Syriac
Euphemia	Omron NA Sans Canadian Aboriginal
Gautami	Omron NA Sans Telugu
Georgia	Source Serif Pro
Gulim	Omron NA Sans G KR
GulimChe	Omron NA Sans G Che KR
Impact	Anton
Iskoola Pota	Omron NA Sans Sinhala
Kalinga	Omron NA Sans Oriya
Kartika	Omron NA Sans Malayalam
Khmer UI	Omron NA Sans Khmer
Lao UI	Omron NA Sans Lao
Latha	Omron NA Sans Tamil

NA5-□□W□□□□ NA5-□□W□□□□-V1 Soft-NA	NA5-□□W□□□□-V2
Lucida Sans Unicode	Omron NA Sans Unicode
Malgun Gothic	Omron NA Sans KR
Mangal	Omron NA Sans Devanagari
Meiryo	Omron NA Sans JP
Meiryo UI	Omron NA Sans UI JP
Microsoft Himalaya	Omron NA Serif Tibetan
Microsoft JhengHei	Omron NA Sans TC
Microsoft New Tai Lue	Omron NA Sans New Tai Lue
Microsoft PhagsPa	Omron NA Sans PhagsPa
Microsoft Sans Serif	Liberation Serif
Microsoft Tai Le	Omron NA Sans Tai Le
Microsoft Uighur	Omron NA Serif Old Uyghur
Microsoft YaHei	Omron NA Sans SC
Microsoft Yi Baiti	Omron NA Sans Yi
MingLiU	Omron NA Serif TC*1
MingLiU_HKSCS	
MingLiU_HKSCS-ExtB	
MingLiU-ExtB	
Mongolian Baiti	Omron NA Sans Mongolian
MS Gothic	Omron NA Gothic JP
MS Mincho	Omron NA Mincho JP
MS PGothic	Omron NA PGothic JP
MS PMincho	Omron NA PMincho JP
MS UI Gothic	Omron NA UI Gothic JP
MV Boli	Omron NA Sans Thaana
NSimSun	Omron NA Serif SC
Nyala	Omron NA Sans Ethiopic
Plantagenet Cherokee	Omron NA Sans Cherokee
PMingLiU	Omron NA PSerif TC*2
PMingLiU-ExtB	
Raavi	Omron NA Sans Gurmukhi
Segoe UI	Omron NA Sans
Segoe UI Light	Omron NA Sans Light
Segoe UI Semibold	Omron NA Sans SemiBold
Shruti	Omron NA Sans Gujarati
SimSun	Omron NA Serif SC*3
SimSun-ExtB	
Sylfaen	Omron NA Sans Georgian Condensed
Symbol	Omron NA Symbol
Tahoma	IBM Plex Sans
Times New Roman	Omron NA Serif
Trebuchet MS	Alatsi
Tunga	Omron NA Sans Kannada
Verdana	Anuphan
Vrinda	Omron NA Sans Bengali
Webdings	Omron NA Web
Wingdings	Omron NA Wing

*1. If the model has been converted to another model from NA5-□□W□□□□-V2, the font will be converted to MingLiU.

*2. If the model has been converted to another model from NA5-□□W□□□□-V2, the font will be converted to Omron NA PSerif TC.

*3. If the model has been converted to another model from NA5-□□W□□□□-V2, the font will be converted to SimSun.

A-7 Differences between the Physical HMI and Simulator

The following major differences exist between the physical HMI and Simulator. In addition, the display by the simulator is not guaranteed to be completely compatible with the physical HMI.

- Trend Graph and Broken-line Graph Objects

  The Simulator does not update graphic displays. Fixed still images are displayed.
- ShowDocument Action and ShowDocument Function

  The Simulator ignores parameters that specify the display positions and sizes for documents. To display documents, there must be Microsoft Edge*1, such as Adobe Reader. The application used to display a document will not be exited when the Simulator is exited or the page is changed after the document is displayed. Exit the application manually.

  *1. On rare occasions, the program may not run correctly due to Windows Update or for other reasons. In these cases, the program may run correctly if you update Sysmac Studio to the latest version.
- ShowTroubleshooter Action and ShowTroubleshooter Function

  The Simulator cannot display the NJ/NX/NY Troubleshooter.
- _HMI_Brightness Variables

  The brightness of the screen does not change for the Simulator.
- Media Player Object

  There may be some differences in the behavior or timing when a video is replayed. The Simulator does not execute actions/functions that operate a Media Player object.
- Operation Log

  The Simulator cannot run functions related to Operation Log.
- Safety Monitor

  The Simulator cannot run functions related to Safety Monitor.

A-8 Version Upgrade History

This section describes the additions and improvements that were made during version upgrades.

A-8-1 Version Upgrade History for Sysmac Studio and Runtime

Sysmac Studio Version 1.10 and Runtime Version 1.00

Item	Description
First release	Support added for NA-series Programmable Terminals.

Sysmac Studio Version 1.11 and Runtime Version 1.01.

Item	Description
Uploading	Added support for uploading.
Expansion of popup page functionality	Expanded functionality for popup pages, such as the addition of display position specifications.
Expansion of IAG functionality	Expanded functionality for IAG variables. Also added support for a Condition event.
Expansion of settings for object appearance	Expanded settings related to appearance, such as adding vertical text for some objects.
Addition of data logging	Added Data Log and Trend Graph objects.

Sysmac Studio Version 1.13 and Runtime Version 1.02.

Item	Description
Support for NX7-series Controllers and NJ1-series Controllers	Added support for connection to NX7-series Controllers and NJ1-series Controllers.
Addition of NJ/NX Troubleshooter	Added support for the NJ/NX Troubleshooter.
Addition of system variables for VNC function	Added _HMI_ConnectedVNCClientCount system variable that represents the number of clients that are connected to the HMI via VNC.

Sysmac Studio Version 1.14 and Runtime Version 1.03.

Item	Description
Addition of broken-line graph function	Added support for the broken-line graph function.
Expansion of NJ/NX-series troubleshooter functionality	Expanded functionality to display/delete event logs of the EtherCAT slave units. Also added support that allows the troubleshooter to switch to a user screen.
Support for union/enumeration variables	Enabled handling of union and enumeration variables of NJ/NX-series Controllers, as well as inputting and displaying of them on HMIs.
Support for creating new structure variables	Enabled creating new structure variables in the HMI.
Expanded maximum number of variables registered in a connected device	The number of global variables that can be registered in a connected device was expanded to 30,000.
Expansion of IAG functionality	Enabled settings of user events for IAG.
HMI Main Unit screen capture functionality	Enabled capturing screens displayed on the HMI.
Custom keypad functionality	Enabled creation of system keypads with user-specified sizes. Also added support for editing then to make a user-specific colored/shaped keypad.
Tabbing order functionality	Enabled settings of order in which to tab around the input focus during data input.

Item	Description
Scale conversion functionality for numeric values	Enabled scale conversion of numeric values.
Indirect addressing of text strings	Enabled indirect addressing of text strings that are displayed on objects.
Buzzer function	Enabled sounding a buzzer from the HMI when a button is pressed, and when an alarm occurs, or for other events.
Page jump from an alarm message to a specified screen	Enabled a page jump from an alarm screen specified in the Alarm Viewer to a specified screen.
Enhanced VNC functionality	Support for accesses control of VNC clients for various conditions.
Addition of tab control objects	Added tab control objects.
Support for background processing of subroutines	Added support for background processing of subroutines to enable parallel processing with updating of the display.
Input of Japanese and Asian languages	Enabled input of Japanese, Chinese (traditional and simplified), and Korean in data input objects.

Sysmac Studio Version 1.15 and Runtime Version 1.04.

Item	Description
Expansion of button object functionality	Enabled combinations of buttons with lamp functionality using variable values.
Indirect specification of upper/lower limit values	The upper/lower limit values for an object such as Data Edit can now be dynamically changed by using variables.
Function to specify a page number	Numerical values can now be assigned to pages, allowing the pages to be switched by using the assigned values.
Sorting and filtering of displayed items in User Alarms Viewer	Enabled setting for sorting displayed items by default. Enabled filtering of displayed alarms by a keyword for each item.
Improved operability of some objects	Enabled changing the size of the CheckBox, Slider, and Radio Button objects.

Sysmac Studio Version 1.16 and Runtime Version 1.05.

Item	Description
Improved operability of User Alarm Viewer objects	Enabled displaying the filtering state of User Alarm Viewer objects.

Sysmac Studio Ver.1.17 Runtime Ver.1.06

Item	Description
Support for NY-series	Enabled connection to NY-series.

Sysmac Studio Ver.1.17 Runtime Ver.1.07

Item	Description
Support for NX1-series	Enabled connection to NX1-series CPU Unit.

Sysmac Studio Ver.1.18 Runtime Ver.1.08

Item	Description
Improved functionality for Retain variables	The number of characters can now be set for String-type Retain variables.
Expanded functionality for data display objects and data input objects	The character encoding method can now be converted for data display objects and data input objects.

Sysmac Studio Ver.1.23, Runtime Ver.1.09

Item	Description
Support for NX102 Series	Connection to an NX102 Series CPU Unit is now possible.

Sysmac Studio Ver.1.24, Runtime Ver.1.10

Item	Description
Support for NX-CSG320	Connection to an NX-CSG320 CPU Unit is now possible.

Sysmac Studio Ver.1.27, Runtime Ver.1.11

Item	Description
Support for fonts in multiple languages	Object font settings can now be set in each language.

Sysmac Studio Ver.1.31, Runtime Ver.1.12

Item	Description
FTP Client	FTP Client functions are now supported.

Sysmac Studio Ver.1.40, Runtime Ver.1.11

Item	Description
Support for Soft-NA	Soft-NA is now supported.

Sysmac Studio Ver.1.40, Runtime Ver.1.13

Item	Description
Support for Operation Log	Operation Log is now supported.
Support for Safety Monitor	Safety Monitor is now supported.

Sysmac Studio Ver.1.40, Runtime Ver.1.08 to Ver.1.13

Item	Description
Support for NA5-□□W□□□□-V1	The NA5-□□W□□□□-V1 is now supported.

Sysmac Studio Ver.1.42, Runtime Ver.1.13

Item	Description
Support for NJ501-R□00 controller	Connection to NJ501-R□00 is now possible.

Sysmac Studio Ver.1.43, Runtime Ver.1.14

Item	Description
Support for alarm time elapsed	Display of the elapsed time after an alarm occurs in the user alarm object is now supported.

Sysmac Studio Ver.1.45, Runtime Ver.1.15

Item	Description
Support for user alarm attached information	Added support for inserting a variable value in the message when a user alarm occurs.

Sysmac Studio Ver.1.50, Runtime Ver.1.16

Item	Description
Supported secure communication	Support for secure communication with the NJ/NX series.

Sysmac Studio Ver.1.54, Runtime Ver.1.16

Item	Description
Support for display of multiple graphs	The number of graph objects that can be placed in a single page in Soft-NA has been increased to two.
Support for secure communication	In Soft-NA, secure communication with the NJ/NX series is supported.

Sysmac Studio Ver.1.54, Runtime Ver.1.17

Item	Description
Support for CS series, CP series	Connection to the CS series and CP series is now enabled.
Support for serial connection	Serial connection with the CS/CJ/CP series is now enabled.
Support for NX502-series	Connection to an NX502-series CPU Unit is now possible.

Sysmac Studio Ver.1.64, Runtime Ver.1.18

Item	Description
Supported Runtime version	Support for Runtime Ver.1.18.

Sysmac Studio Ver.1.65, Runtime Ver.1.18

Item	Description
Support for NA5-□□W□□□□-V2	The NA5-□□W□□□□-V2 is now supported.

Sysmac Studio Ver.1.67, Runtime Ver.1.19

Item	Description
Supported Runtime version	Support for Runtime Ver.1.19.
Preload method setting supported	The preload method can now be selected.

A-8-2 Version Upgrade History for Sysmac Studio Only

Version 1.11

Item	Description
Support for copying settings	Added support to copy settings for user alarm tables, data sets, and recipe templates and paste them into external applications.
Improvement to property display method	Enabled displaying properties by double-clicking objects.

Version 1.13

Item	Description
Support to copy CJ/NJ variable settings	Added support to copy and paste variables directly from the Sysmac Studio and CX-Programmer.

Version 1.14

Item	Description
Enhanced Search and Replace functionality	Expanded target for the Search and Replace function to all the texts within a project.
Support for the simulation mode	Enabled confirmation of indications such as the ON/OFF status of Lamps and other objects.
Sysmac Studio printing functionality	Enabled printing of information including project settings with HMI selected.
Direct text editing	Enabled direct editing of Labels on the Page Editor.
Importing of vector graphic files	Enabled selecting of vector graphic files (e.g.,.XAML files) for the Image files of objects.
Improved Watch Tab Page	Enabled constant monitoring of global variables.
Cross reference function	Enabled cross reference when the HMI is selected.

Version 1.15

Item	Description
Function to change Button type	The Button Type such as Set or Momentary can now be changed by the setting of properties.
Improved procedures to add variables	A variable can now be created and added to the variable table directly from the variable setting field.
Improved operation of specifying subroutines	The function has been expanded to open the Code Editor (for editing subroutines). A subroutine can now be created from the Properties Window.
Japanese translation of items in Properties	The items in the Properties Window are now shown in Japanese under a Japanese environment.
Serial creation of multiple objects	It is now possible to create more than one object with the same appearance and settings at the same time.

Version 1.16

Item	Description
Version control	Enabled version control.

Version 1.18

Item	Description
Improved functionality for Retain variables	Validation is now executed for the total number and total size of Retain variables when a project is built.
Resource usage status	You can now check the usage of Retain variables.

Version 1.20

Item	Description
Enhanced workability between devices	Enhanced workability between devices, such as opening windows of different devices at the same time.
Improved copy and paste	The global variables and subroutines referenced by objects during object copy and paste are automatically copied.
Increased cross reference range	Added data groups, recipes, and data sets as cross reference targets.
Image preview	When selecting an image to be pasted on an object, a preview of the image is displayed.
Incremental build	Added support for incremental build.
Deletion of all Imported IAGs	Added support for a function that deletes all Imported IAGs that are not used.

Version 1.22

Item	Description
Improvement of Code Editor Performance	Performance when editing a large amount of source code has been improved.
Improvement of user alarm operability	Grid sorting when configuring group settings is supported.
Improvement of resource operability	"Undo" is supported in the resource editing software.
Image file output of pages	A function for saving a page as an image file is supported.

Version 1.24

Item	Description
Support for subroutine searches	Added support for a function that searches for the place where a subroutine is used from the Code Explorer.
Importing and exporting of devices	Added support for importing and exporting devices.
Support for editing Resource IDs	Added support for editing a Resource ID set for an object.

Version 1.25

Item	Description
Set as default	Added support for setting initial value of object properties.

Version 1.26

Item	Description
Simulator	Properties and other items can be edited while the simulator is running.

Version 1.27

Item	Description
Import/Export User Alarm	Importing and exporting of user alarms is supported.
Importing and exporting of all resources	Importing and exporting of all resources at once is supported.
Importing and exporting of object properties	Importing and exporting of object properties is supported.
Simulator	• A function for displaying any page is supported. • A function for switching the project language is supported. • A screen shot function is supported.
Language settings	• The default language can now be changed. • Default font settings can now be set.
Variable mapping	A function that automatically executes variable mapping is supported.

Version 1.40

Item	Description
Support for specifying the size for the Image File Output	The size can now be specified when outputting image files.
Support for exporting resource files	Exporting resource files is supported.
Support for importing and exporting pages	Importing and exporting pages is supported.
Support for replacing the name of pages	Replacement is now executed when the name of a page is changed.
Support for displaying property by default	Property window and other items are now displayed by default when such as creating a new project.

Version 1.41

Item	Description
Improvement of copy-and-paste function	Brackets are now automatically replaced with appropriate ones when copying and pasting array variables between the controller and HMI.
Improvement of text input	Texts and resource IDs are now displayed in the drop-down list which is displayed when entering texts and resource Ids from the property.

Version 1.43

Item	Description
Automatic application of variable comment	Device variable comments are automatically applied to HMI variables.
Support for upload function	A function specializes in uploading is supported.
Improvement of resource editing	A mode to always overwrite existing resources when editing text in properties is now supported.

Version 1.45

Item	Description
Improvement of performance	Overall performance when using HMI has been improved.
Improvement of operability	The general user interface of the Properties Window has been improved. Animation is integrated into the Properties Window.
Support for displaying variables	A function for displaying the variables assigned on objects in the page editor is now supported.
Improvement of jump function after search	Improvements have been made so that the focus moves to the appropriate position when jumping to a resource after a search.
Support for searching unused variables	Searching unused variables in the global variable table is now supported.
Improvement of build errors	The error details displayed during building have been improved.

Version 1.46

Item	Description
Support for Animations Window	The Animations Window is once again supported.
Support for changing the size of the Animations dialog box	Changing the size of the Animations dialog box is supported.

Version 1.47

Item	Description
Support for font replacement	A function to replace the font-related settings under specified conditions is supported.
Support for forced remapping	A function that rebuilds the variable mapping information is supported.
Improvement of user interface	Made improvements, such as supporting the copying of multiple objects in the font settings.

Version 1.54

Item	Description
Support for download function	A specialized function for downloading is now supported.
Support for number of objects display	The number of objects is now displayed in Page Explorer.
Support for deletion of unused resources	A function for deleting unused resources is now supported.
Support for IAG replacement	A function for replacing IAGs is now supported.
Improvement of Resource Usage	The number of global variables can now be checked in Resource Usage.
Support for device replacement	A function for replacing device assignments is now supported.
Support for assignment mode	A mode that displays only the variables and resource assignments in the Properties Window has been added.

A-8-3 Sysmac Studio Corresponding Versions

The table below shows the latest Runtime and system program versions included with each version of Sysmac Studio.

Sysmac Studio	NA5-□□□□□□		NA5-□□W□□□□-V1*1		NA5-□□W□□□□-V2*2		Soft-NA	
	Runtime	System program	Runtime	System program	Runtime	System program	Runtime	Soft-NA
1.10	1.00	1.1.0	-	-	-	-	-	-
1.11 to 1.12	1.01*3	2.0.0	-	-	-	-	-	-
1.13	1.02*4	3.0.3	-	-	-	-	-	-
1.14	1.03*5	4.3.3	-	-	-	-	-	-
1.15	1.03*5*6	4.4.1	-	-	-	-	-	-
1.16	1.05*5	5.0.3	-	-	-	-	-	-
1.17	1.07*5	5.1.5	-	-	-	-	-	-
1.18	1.08*5	5.1.5	-	-	-	-	-	-
1.19	1.08*5	5.2.2	-	-	-	-	-	-
1.20	1.08*5	6.0.2	-	-	-	-	-	-
1.23	1.09*5	7.0.5	-	-	-	-	-	-
1.24	1.10*5	7.1.2	-	-	-	-	-	-
1.24.2	1.10*5	7.2.1	-	-	-	-	-	-
1.26	1.10*5	7.3.0	-	-	-	-	-	-
1.27	1.11*5	7.3.0	-	-	-	-	-	-
1.31	1.12*5	7.4.0	-	-	-	-	-	-
1.40	1.13*5	8.0.0	1.08	6.0.2	-	-	1.11	1.01
			1.09	7.0.5	-	-		
			1.10	7.4.0	-	-		
			1.11	7.4.0	-	-		
			1.12	7.4.0	-	-		
			1.13	8.0.0	-	-		
1.43	1.14*5	8.0.0	1.14	8.0.0	-	-	1.11	1.01
1.44	1.14*5	8.1.0	1.14	8.1.0	-	-	1.11	1.01
1.45	1.15*5	9.0.0	1.15	9.0.0	-	-	1.12	1.02
							1.13	
							1.14	
							1.15	
1.50	1.16*5	10.1.0	1.16	10.1.0		-	-	-
1.52	1.16*5	10.2.0	1.16	10.2.0	-	-	-	-
1.54	1.17*5	11.0.0	1.17	11.0.0	-	-	1.16	1.02
							1.17	
1.64	1.18*5	11.0.0	1.18	11.0.0	-	-	1.18	1.04
1.65	1.18*5	11.0.0	1.18	11.0.0	1.18	11.0.0	1.18	1.04
1.67	1.19*5	11.0.0	1.19	11.0.0	1.19	12.0.0	1.19	1.04

*1. NA5-□□W□□□□-V1 supports only Runtime Ver.1.08 or higher.

*2. NA5-□□W□□□□-V2 supports only Runtime Ver.1.18 or higher.

*3. The projects created in version 1.00 are converted to version 1.01.

*4. The projects created in either version 1.00 or 1.01 are converted to version 1.02.

*5. Projects created in version 1.00, 1.01, or 1.02 are converted to version 1.03.

*6. The runtime version will be converted to 1.04 if the HMI Extended Option is enabled.

Additional Information

Older versions of Runtime can be used with Sysmac Studio, but the internal version may have changed due to bug fixes. The version of the system program required at that time may also have changed. The required version of the system program is supplied with Sysmac Studio. Follow the procedure to update it.

A-8-4 Runtime Support Limit

Support of Runtime is scheduled to end as follows. In principle, support ends three years after release of the successor version.

In Sysmac Studio after Runtime support has ended, new projects can no longer be created by the corresponding Runtime version. When a project in the corresponding Runtime version has been loaded, the project will be converted to the lowest version of currently supported Runtime versions.

Runtime	Release	Status	End of support (scheduled)
1.00	June 2014	End of support	September 2015
1.01	October 2014	End of support	September 2015
1.02	April 2015	End of support	September 2015
1.03	September 2015	Supported	April 2024
1.04	April 2016	Supported	April 2024
1.05	July 2016	Supported	April 2024
1.06	August 2016	Supported	April 2024
1.07	October 2016	Supported	April 2024
1.08	February 2017	Supported	April 2024
1.09	April 2018	Supported	April 2024
1.10	July 2018	Supported	April 2026
1.11	April 2019	Supported	April 2026
1.12	December 2019	Supported	April 2026
1.13	April 2020	Supported	April 2026
1.14	October 2020	Supported	April 2026
1.15	April 2021	Supported	April 2026
1.16	June 2022	Supported	April 2026
1.17	April 2023	Supported	October 2028
1.18	October 2025	Supported	July 2029
1.19	July 2026	Supported	-

A-9 Precautions for Version Upgrades

Observe the following precautions for version upgrades.

Sysmac Studio Version 1.14 and Runtime Version 1.03

- User-defined enumerations on external devices are not assigned during version upgrades. You must redefine them.
- Some data type definition names are reserved for the data definitions of structures, unions, and numerations on the HMI. An error will occur if a reserved name is used, in which case you must change the name.

Sysmac Studio Version 1.18 and Runtime Version 1.08

- The values of Retain variables are cleared when they are downloaded to NA Units with runtime version 1.07 or lower. When downloading is completed, the NA Unit will be restarted and the Retain variable area will be rebuilt, so approximately 10 seconds longer than for a normal restart will be required. Do not turn OFF the power supply to the NA Unit while the Retain variable area is being rebuilt.

Sysmac Studio Version 1.27 and Runtime Version 1.11

- If you convert to Runtime Version 1.11, expansion of font settings may cause processing to take 10 minutes or longer, depending on the project size.

I

Index

I

A

actions 4-47, A-3
adding an HMI 2-16
alarms 1-9
animations 5-7

B

Battery 1-9
breakpoints 7-3
building 4-80

C

Clear All Memory 3-31
Code Editor 4-62
Code Explorer 2-4
Communications Setup 6-4
comparison results 8-11
connected devices 3-3
Continue 7-6, 7-7
Copying the Font Settings 4-65
creating a project file 2-15
creating pages 4-22
custom objects 9-16

D

data logging 1-9, 4-39
debugging functions 7-2
Deletion of Unused Resources 4-51
device references 3-3
Device Replacement 4-76
device settings 3-12
differences between the physical HMI and Simulator .. A-15

E

entry assistance 2-10
entry methods for global variables 4-12
events 1-7, A-2
events and actions 5-7
examples of using objects 5-12
Exporting Object Properties 11-21
Exporting Pages 11-25
Exporting Resources 11-16
Exporting User Alarms 11-9
external 3-3
external connected devices 3-3
external variables 3-8, 4-3, 4-6

F

FINS settings 3-17
Font Replacement Procedure 4-64
FTP 10-3
FTP settings 3-15
functional objects 5-2

G

global events 4-47
global subroutines 4-49, 4-56
global variables 4-3
graphic objects 5-4

H

HMI clock 3-28
HMI Code Editor 11-3
HMI Name 3-29
HMI projects 1-4
HMI software configuration 1-4
HMI versions 2-17
HMI Write 3-30

I

IAG collection settings 9-5
IAG Replacement 4-73
IAG resources 4-52
IAGs 9-3, 9-6
importing device variables 3-6
Importing Object Properties 11-20
Importing Pages 11-24
Importing Resources 11-13
Importing User Alarms 11-7
internal connected devices 3-3
internal variables 4-3

L

language settings 3-23, 9-4
language specifications 4-62

M

main pages 4-22
mapping variables 3-7
memory specifications 1-6
menu 2-7
Multiview Explorer 2-3

N

NA-series Programmable Terminals 1-2
new user alarm 4-35
notation for CustomDisplayFormat 5-6
notation for expression 5-6
NTP settings 3-16

O

object attributes 5-4
object list 5-2
objects 1-5, 5-2
offline comparison 4-82
offline debugging 7-9, 7-10
online connection 6-2
operating procedure 1-10
operation log 5-35
Operation Log Settings 3-25
option settings 11-2

P

Page Explorer 2-4
page subroutines 4-56
pages 1-4
PDF file 5-12
popup pages 4-23
properties 5-5

R

recipe 5-24
recipes 1-9, 4-42
registering IAGs 9-13
Replacing the Font Family 4-67
reset 3-32
Reset Default Value 4-32
Resetting the Font Settings to the Default Settings 4-69
Resource Usage 4-83
resources 1-9, 4-50, 9-9

S

Scaling the Font Size to the Specified Magnification 4-71
search and replace 4-63
security settings 3-21
serial IDs 6-7
set as default 4-30, 4-31
simulator connection 6-2
step execution 7-5
step-in execution 7-5
step-out execution 7-6
storage media 8-13
subroutine execution 4-57
subroutine group 4-49
subroutine variables 4-3, 4-20
subroutines 1-8, 4-56
supported formats A-5
system-defined variables 4-3, 4-16, 4-17

T

TCP/IP settings 3-14
Toolbar 4-33, 9-4
Toolbox 2-5
Transfer 8-2
trend graph 5-18

U

updating device variables 3-6
user alarm 5-14

V

variables 4-3
VNC 10-2
VNC settings 3-18

W

Watch Tab Page 7-3

OMRON Corporation **Industrial Automation Company**

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

Regional Headquarters

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

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

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

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

Authorized Distributor:

©OMRON Corporation 2014-2026 All Rights Reserved.
In the interest of product improvement,
specifications are subject to change without notice.

Cat. No. V118-E1-28 0726