INOVANCE

InoTouchPad

User Guide

Preface

Introduction

InoTouchPad is HMI configuration software developed by Inovance, which provides powerful functions and friendly user interface.

This user guide describes how to use InoTouchPad to configure an HMI project and run it on a PC or embedded hardware. This guide assists you in building communication systems, creating new projects, configuring projects, and applying projects to various platforms. This guide applies to InoTouchPad v0.9.0.0 and above.

Other documents

Name	Description
InoTouchPad User Guide (this guide)	introduces the installation, basic operation, project configuration and function applications.
IT7000 Series HMI Application Cases	Introduces the application cases of InoTouchPad, concerning project management, controls, alarm, communication, tags, reporting, script functions, and recipes to guide users to customize project configuration.
IT7000 Series HMI Quick Start	Introduces the quick commissioning and basic operations of the product.
IT7000 Series HMI Product Selection Guide	Describes the technical specifications and dimensions of products, as well as the detailed specifications and selection of optional accessories (installation accessories, cables, and so on).
IT7150E HMI User Guide	Introduces the technical specifications, installation dimensions, installation method and commissioning of IT7000 series 15-inch products.
IT7100 HMI User Guide	Introduces the technical specifications, installation dimensions, installation method and commissioning of IT7000 series 10.1-inch products.
IT7070 HMI User Guide	Introduces the technical specifications, installation dimensions, installation method and commissioning of IT7000 series 7-inch products.

Description

This guide introduces the installation, basic operation, project configuration and function applications of the software. The structure of the guide is as follows:

Overview: This section describes the installation and uninstallation of InoTouchPad.

Quick Start: This section guides you to configure the system, and build, compile and simulate and download a project.

Project Management: This section describes project compiling, uploading and downloading, simulation, pass-through, packaging, firmware updating, and language font settings.

Communication: This section introduces communication principles, settings and protocol options in HMI project configuration.

Tags: This section introduces the definition, categories, setting and use of tags in HMI projects.

Creating a Screen: This section guides users how to create screens and use objects.

Controls: This section describes common operations and property setting of all the simple controls and enhanced controls provided by InoTouchPad.

User Management: This section describes the management of user permissions in HMI projects.

Creating an Alarm System: This section guides users to create an alarm system to visualize processes and alarms.

Creating a Recipe: This section guides users to create, use, import and export recipes.

Historical Data: This section describes how to set up, export data logs, and the specific data contained in historical data.

System Functions and Scripts: This section describes all system functions that InoTouchPad supports, scripting principles, and how to call functions in scripts.

Project Language: This section describes how to set up languages in HMI projects.

HMI Settings: This section describes the hotkeys, scheduler, project version, and project settings in InoTouchPad.

Reporting: This section guides users with reporting features through examples.

Major Functions and Applications: This section introduces the key functions in InoTouchPad, including drag and drop, table operation, import and export , printing and SCADA applications.

Data Service: This section describes the use of MQTT data service.

Intended Readers

This guide is intended for operators and configuration engineers who use InoTouchPad to communicate, configure, commissioning, install and service IT7000 series products.

Revision History

Date	Version	Description
2021–08	A01	Minor corrections.
2021–06	A00	First issue.

Access to the Guide

This guide is not shipped with the product. If you need to obtain the electronic PDF file, you can obtain it in the following ways:

Visit *www.inovance.com*, go to Services and Support→Download.

Table of Contents

1 Overview

1.1 Introduction

InoTouchPad is a configuration screen development system for Inovance InoTouch series HMIs. It features an integrated development environment and rich and powerful functions.

This software is suitable for Inovance IT 7000 series HMI products, and can also runs independently on a PC as the software for a small SCADA.

Software

InoTouchPad is a programming tool developed by Inovance. For the latest version, contact your HMI supplier or download it from our website (http://www.inovance.com).

Requirements on Computer Specifications

CPU: Intel or AMD, 2 GHz or faster

Memory: 1 GB or above

Hard disk: at least 1 GB free space

Display: color display with resolution of 1024 x 768 or above

Communication port: Ethernet port or USB port

Operating system: Windows 7 or Windows 10

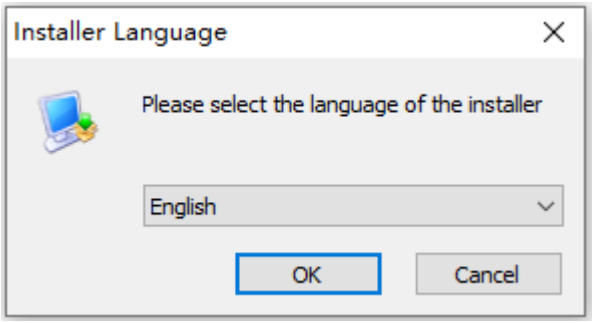
1.2 Installation and Uninstallation

1.2.1 Installing InoTouchPad

The steps to install the InoTouchPad software are as follows:

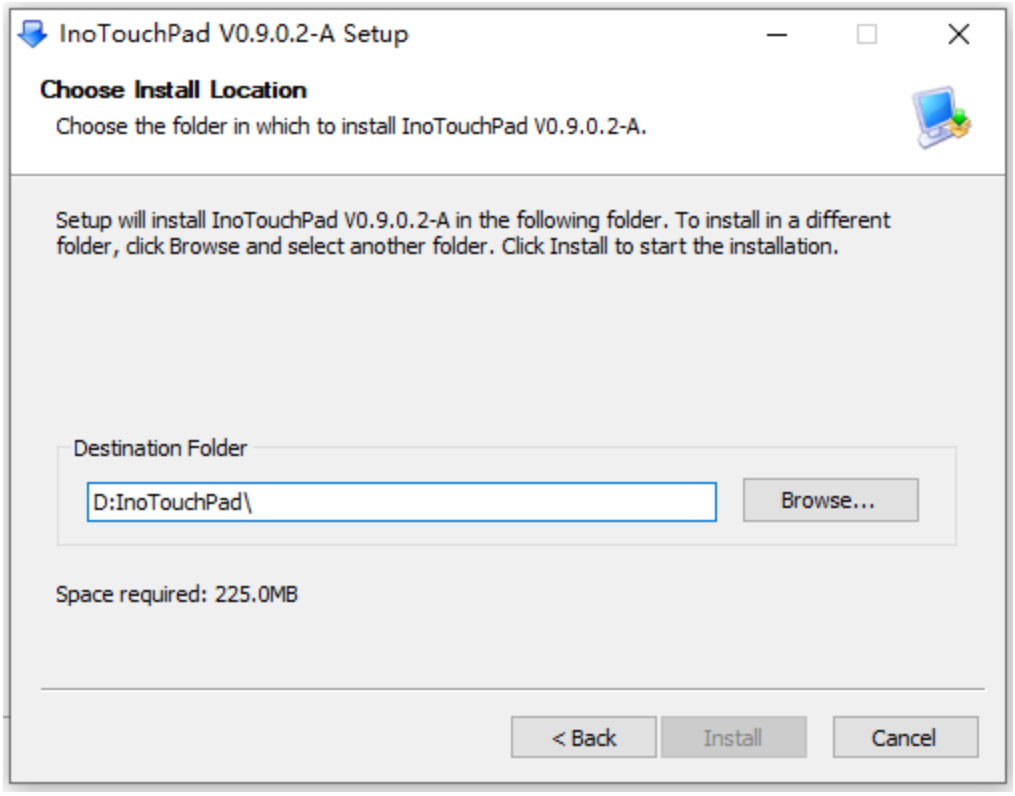
1. Double-click the setup program 

.

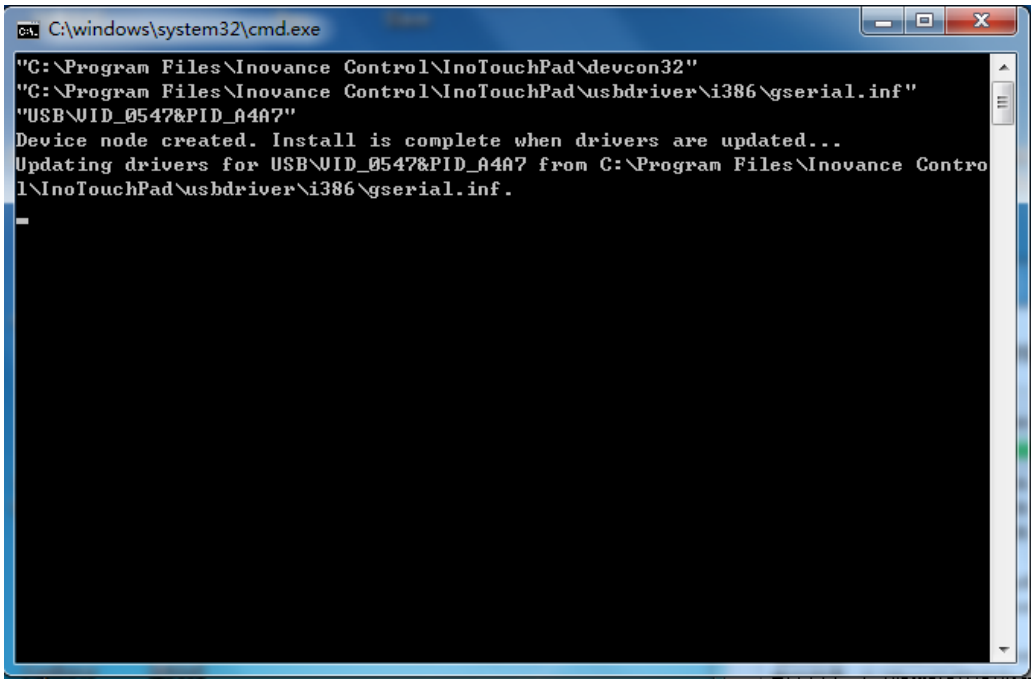


2. Select the installation language and click OK. Click Next in the pop-up window.

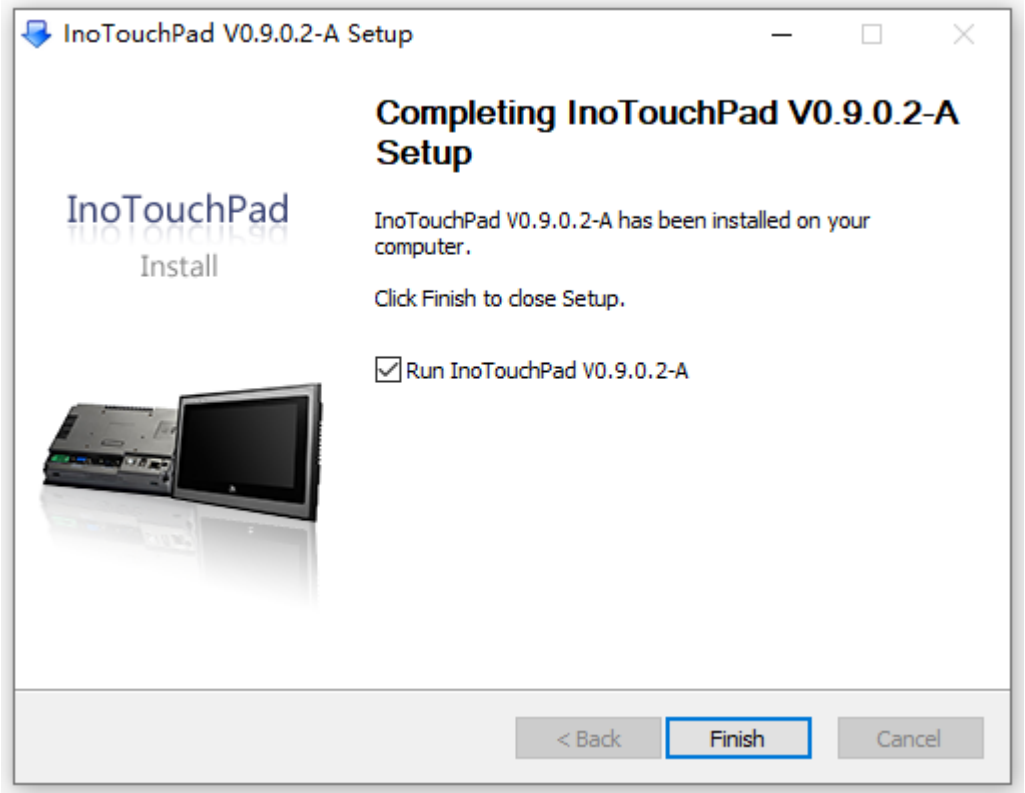
3. Select the installation path. Click Install to start the installation.



4. During installation, the USB driver is also installed.



5. The installation takes about one minute, and then the following window will pop up. If you want to run InoTouchPad immediately after the installation is completed, check Run InoTouchPad1.0. 0 (R), and then click the Finish button to complete the installation and start the software; If you do not check the option, you will exit the installation wizard.



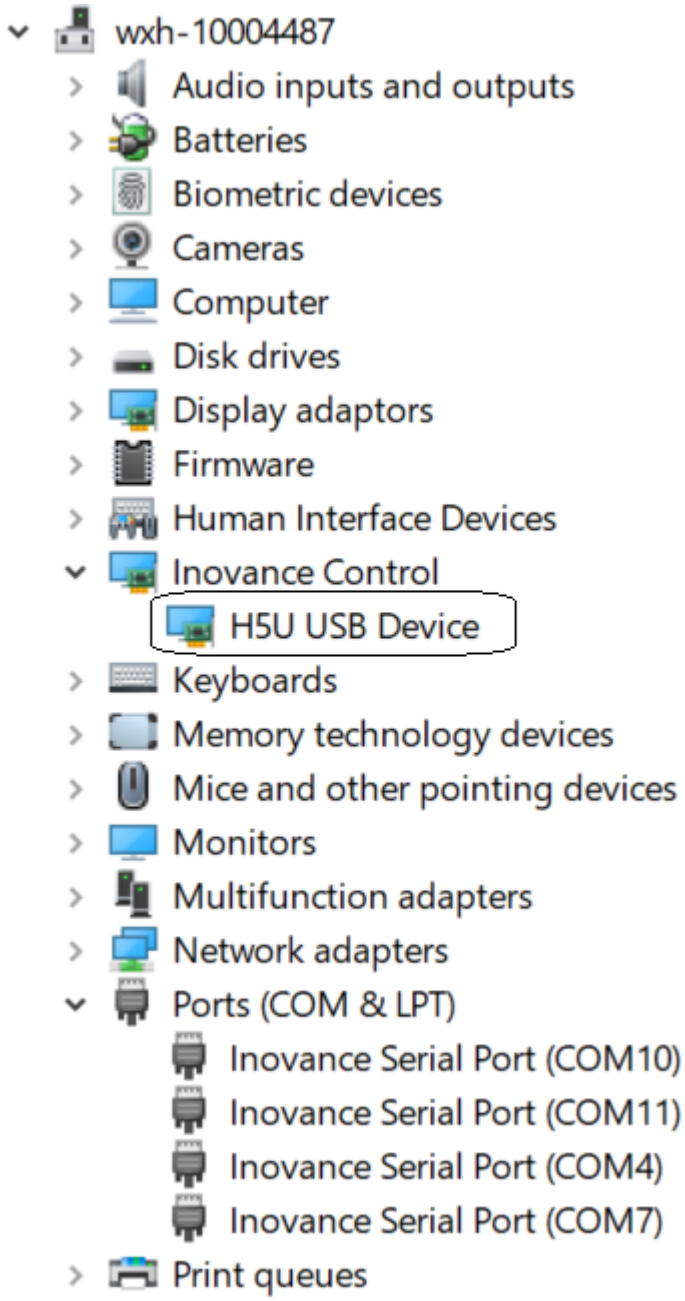
Note

If the USB driver cannot be installed automatically during software installation, you can install it manually.

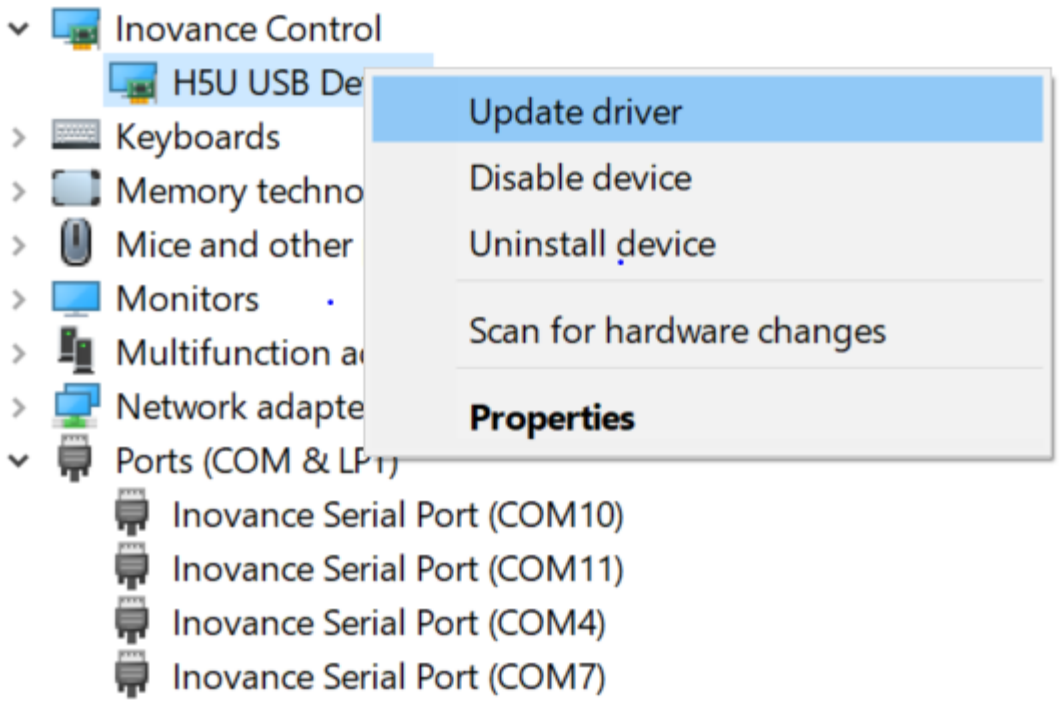
1.2.2 Installing USB Driver Manually

1. Find the USB device first.

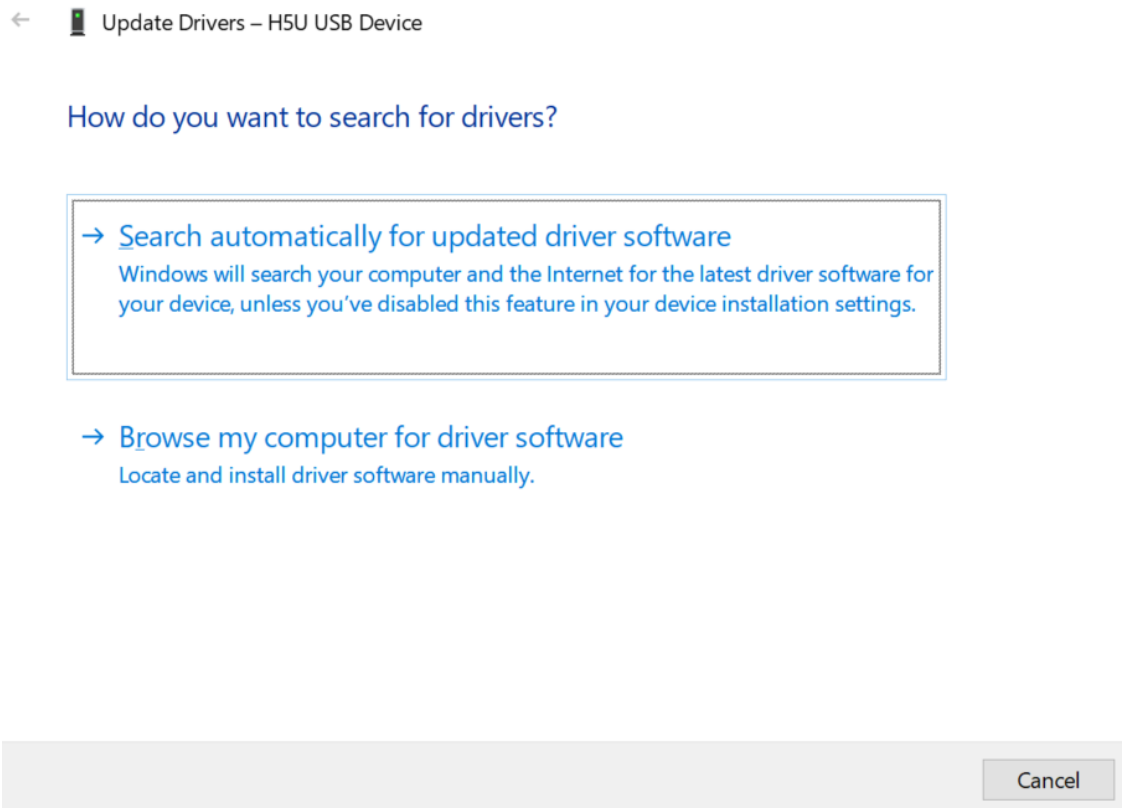
- For Windows 7, go to Control Panel→Administrative Tools→Computer Management→Device Manager→Other devices;
- For Windows 10, go to Control Panel→Administrative Tools→Device Manager→USB controllers;

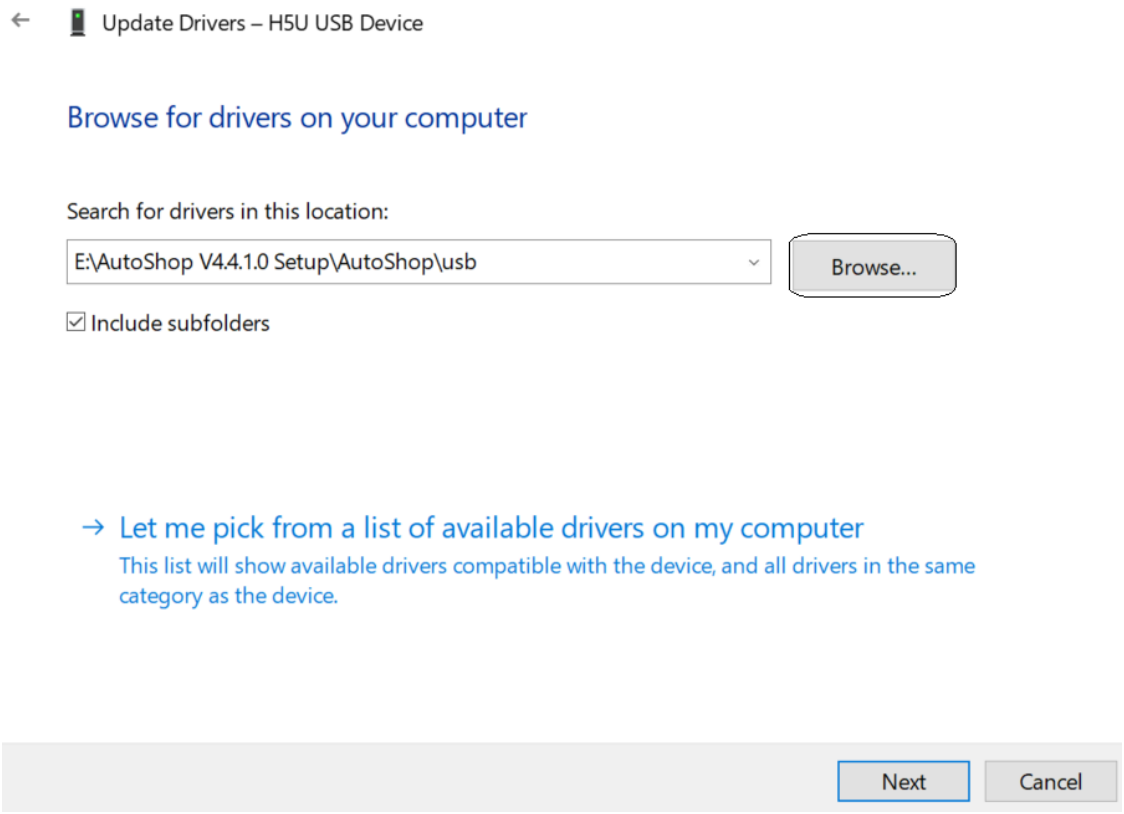


2. Right-click the device and select "Update Driver Software".



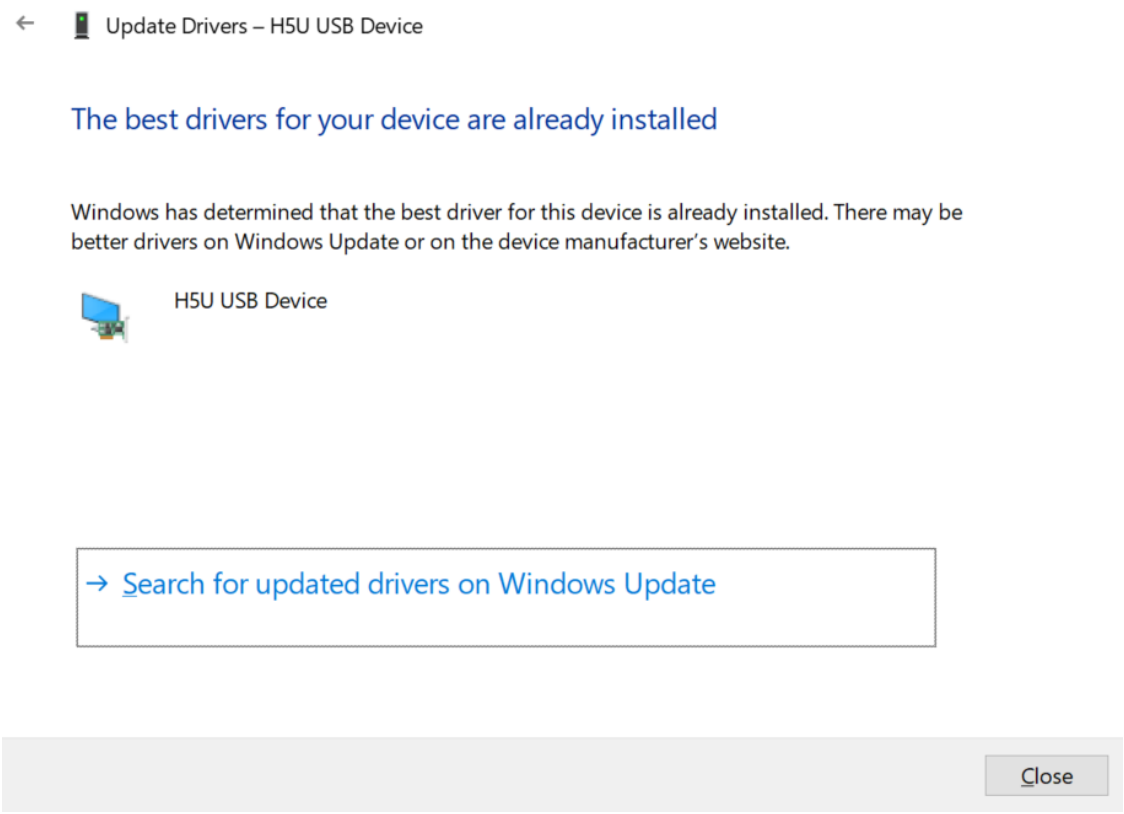
3. Select "Browse my computer for driver software", set the search location to the InoTouchPad software installation path, and check "Include subfolders", as shown in the following figure:





4. Click Next and select "Install this driver software anyway".

5. After the driver is installed, the following window appears. Click Close to finish the installation.

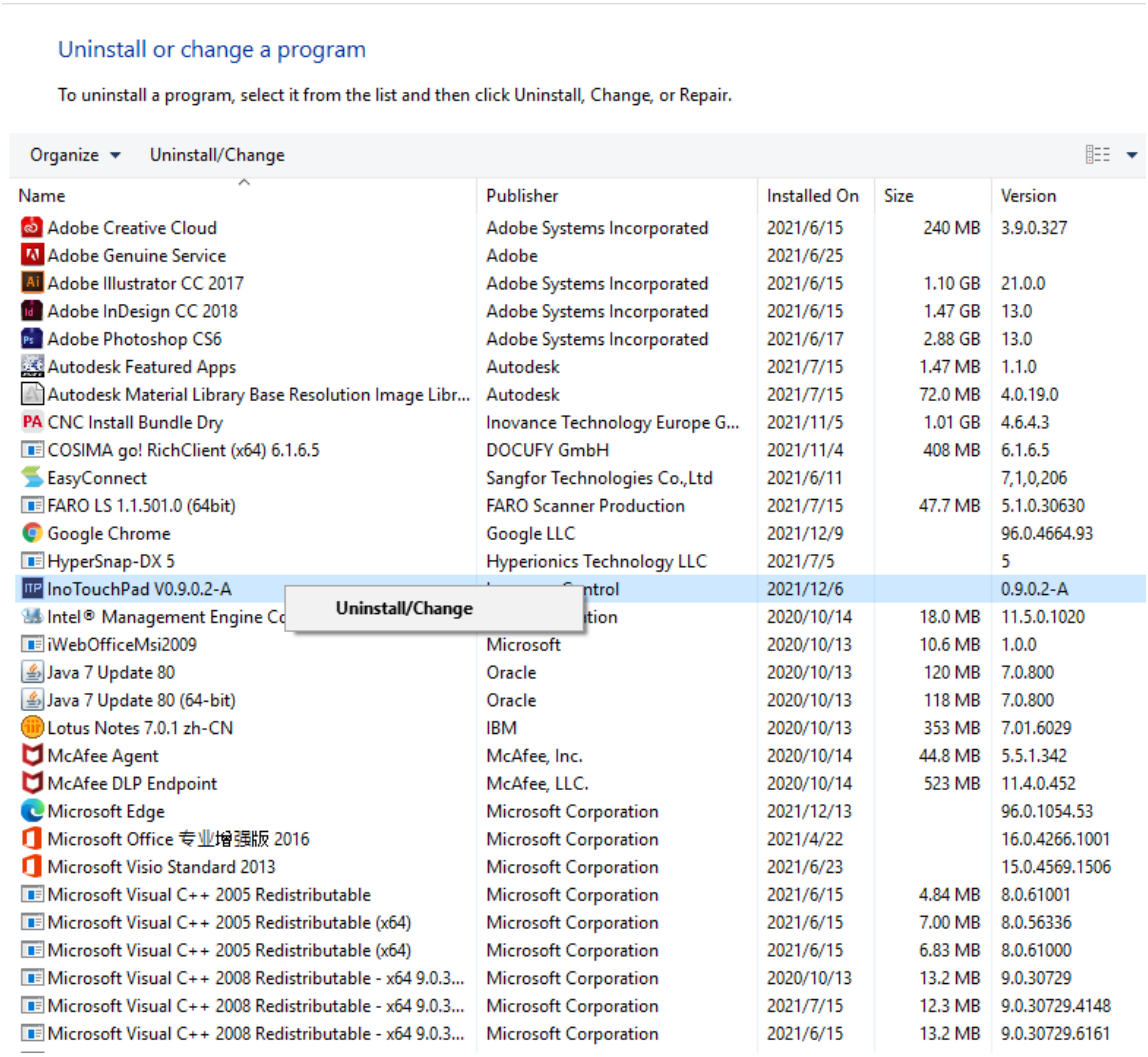


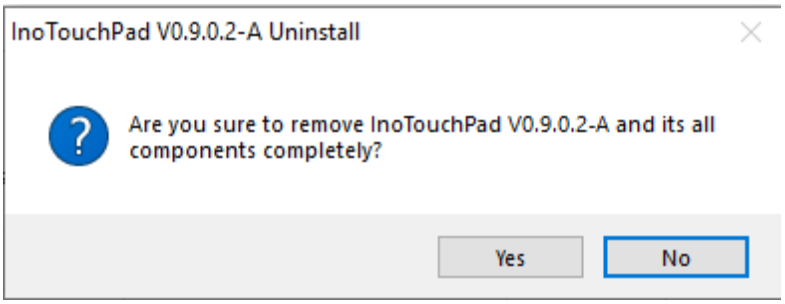
1.2.3 Uninstalling InoTouchPad

You can uninstall InoTouchPad in Programs and Features or with the uninstaller in the InoTouchPad software installation directory.

With Programs and Features

Go to Control Panel->Programs and Features, find InoTouchPad in the list, right-click and select Uninstall/Change, and confirm to completely remove all its components.



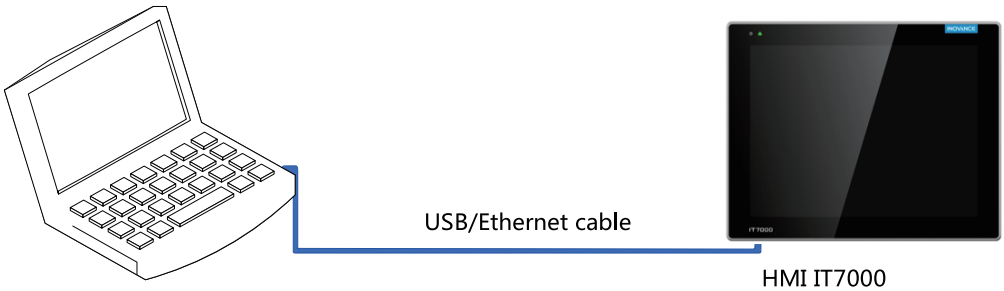


With the uninstaller

Double-click the uninstaller uninst.exe in the installation directory to uninstall InoTouchPad.

1.3 Connection of Host Controller and HMI

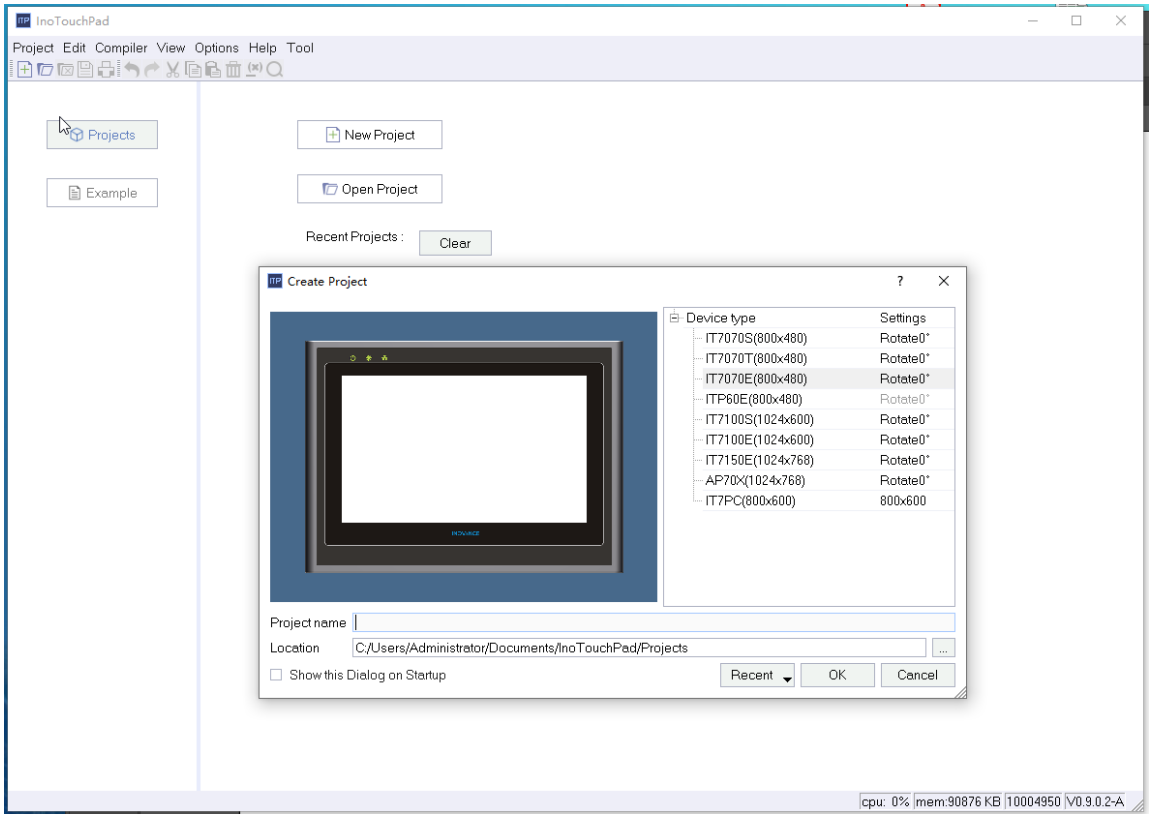
A PC installed with InoTouchPad is connected with IT7000 through a USB/Ethernet cable. After a project is finished in InoTouchPad, it can be downloaded to IT7000 for commissioning.


USB/Ethernet cable
PC with InoTouchPad installed
HMI IT7000

2 Quick Start

2.1 User Interface

When you run the InoTouchPad software for the first time, the Create New Project dialog box will pop up. You can create a new project or open a project or open an Example Project to enter the main interface of InoTouchPad.



The user interface includes: menu bar, toolbar, Project tree view, Detailed view, screen editor, right sidebar, Properties view and Output view, as shown in the following figure:

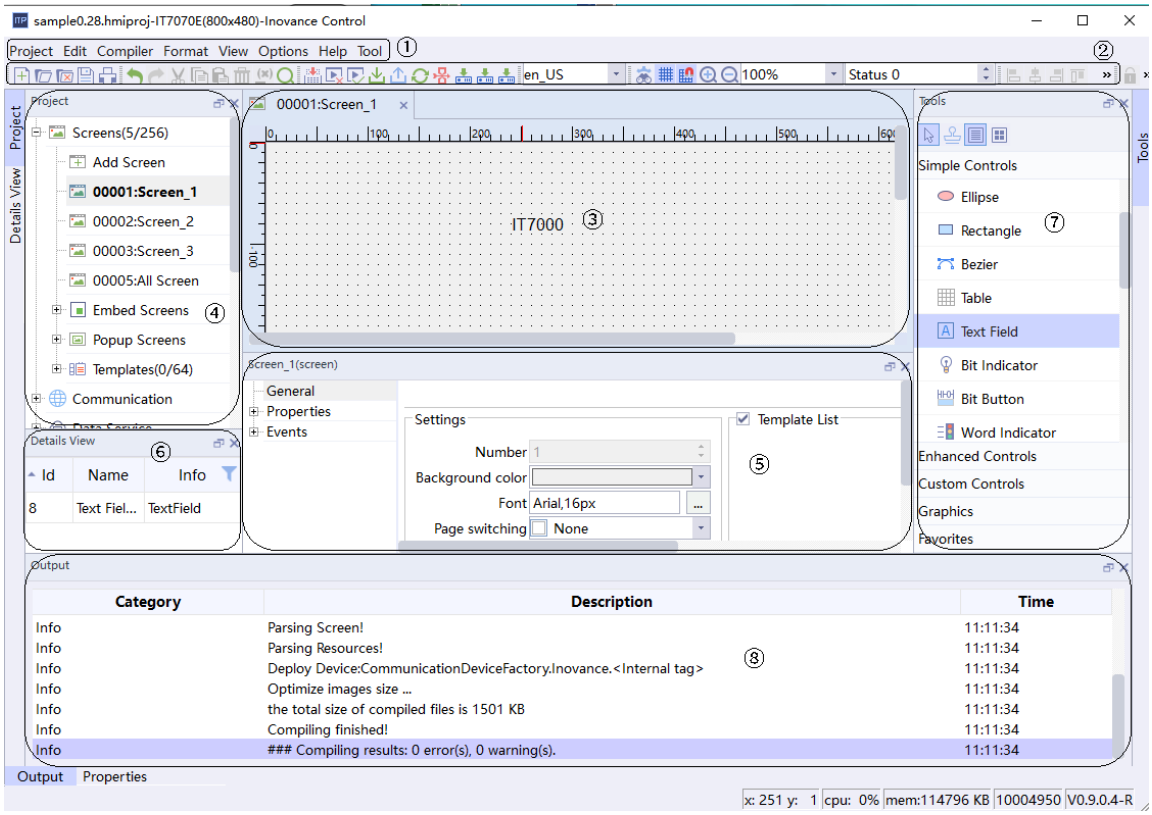


Table 2–1 Layout

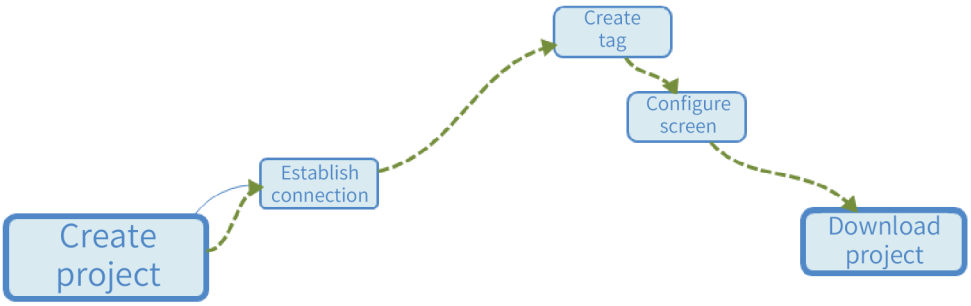
Area		Description	Description
(1)	Menu bar	The framework only provides 5 menus including Edit, View, Options and Help Other menus belong to functional modules	Menus
(2)	Toolbar	The framework only provides three toolbars: Project, edit and compiler Other toolbars belong to functional modules	Buttons for main operations
(3)	Screen editor	Pages Page-specific properties window	You can open up to 20 tab pages
(4)	Project tree view	The tree view includes Screens, Communication, Alarm Management, Recipes, Scripts, and Historical Data	N/A
(5)	Object box	During screen configuration, you can drag an item from Screens, Tags, Recipes, Text Lists and Graphics Lists to the screen to generate a control	Only for Screens, Tags, Recipes, Text Lists, Graphics Lists and Reports
(6)	Details view	If you select a screen item in the Project view, all configured control objects will be listed in the Details view (if there is an object group in the screen, you can select any control in the group in the Details view, and then edit the properties of the control separately in the Properties view). If you select a tag group item, all the tags in the group are listed in the view where you can drag and drop the tags onto the screen	N/A

Area		Description	Description
(7)	Right sidebar	For Screens: control list For Scripts: function list wizard and code template wizard For Communication: protocol list	N/A
(8)	Output view	Displays compiling information Displays operation information of function modules Displays status information	N/A

2.2 Project Configuration

2.2.1 Overview

Typical configuration process:



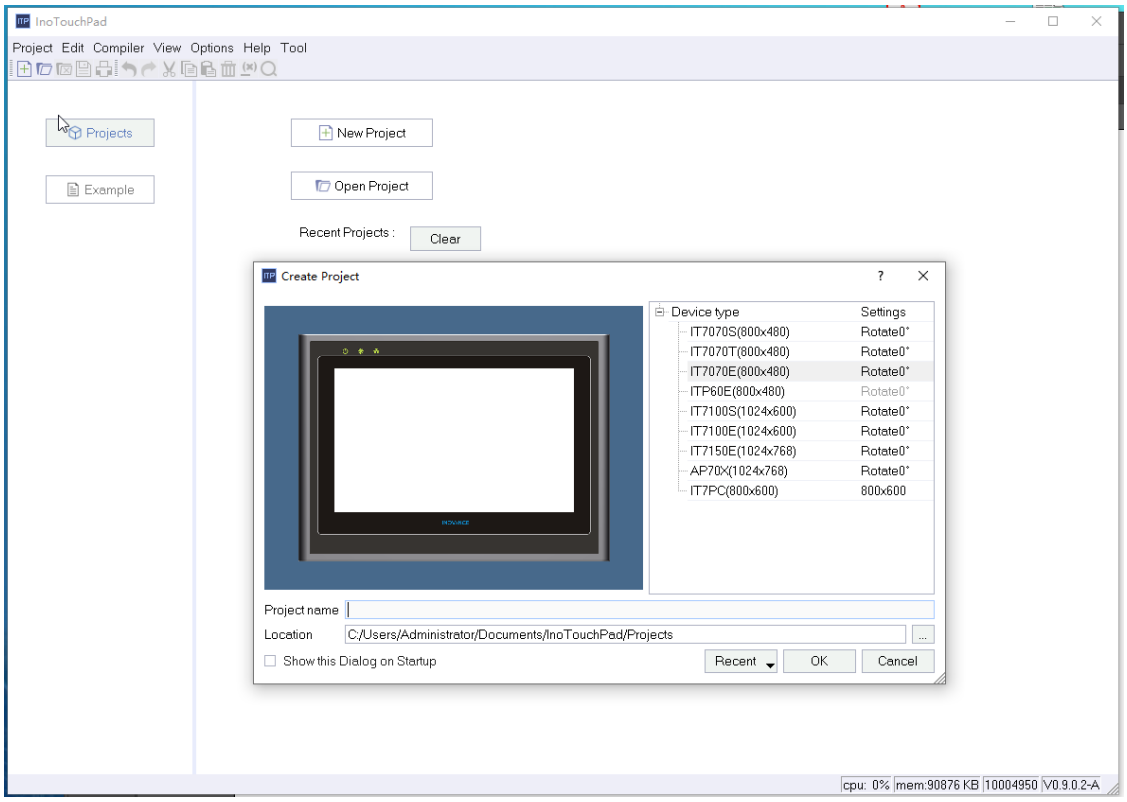
The following is a brief description of the typical operation steps to configure a project in InoTouchPad.

2.2.2 Creating a Project

Follow these steps to create a HMI configuration project:

1. Double-click the InoTouchPad software icon 

on the desktop to open the software, and click New Project, and the following dialog box will pop up:

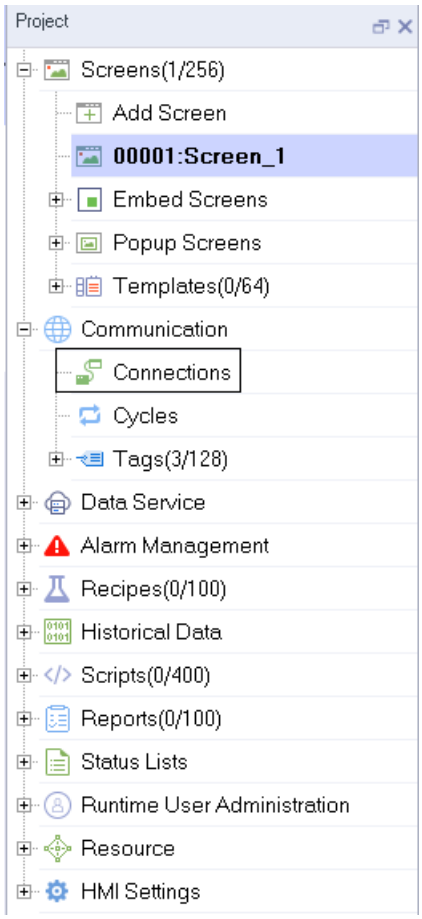


2. Select the desired touch screen equipment (IT7070E is selected in this example), then enter "Indicator Light" as the project name and select the saving location of the project, and click OK to create the project.

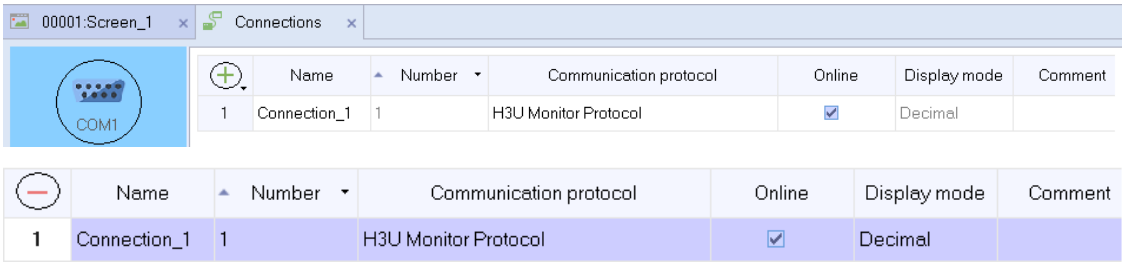
2.2.3 Establishing a Connection

Connection refers to the communication mode between the host controller software and the target device, and the specific creation steps are as follows:

1. Double-click or right-click Connection in the Communication tree of the project window to open the connection editor.



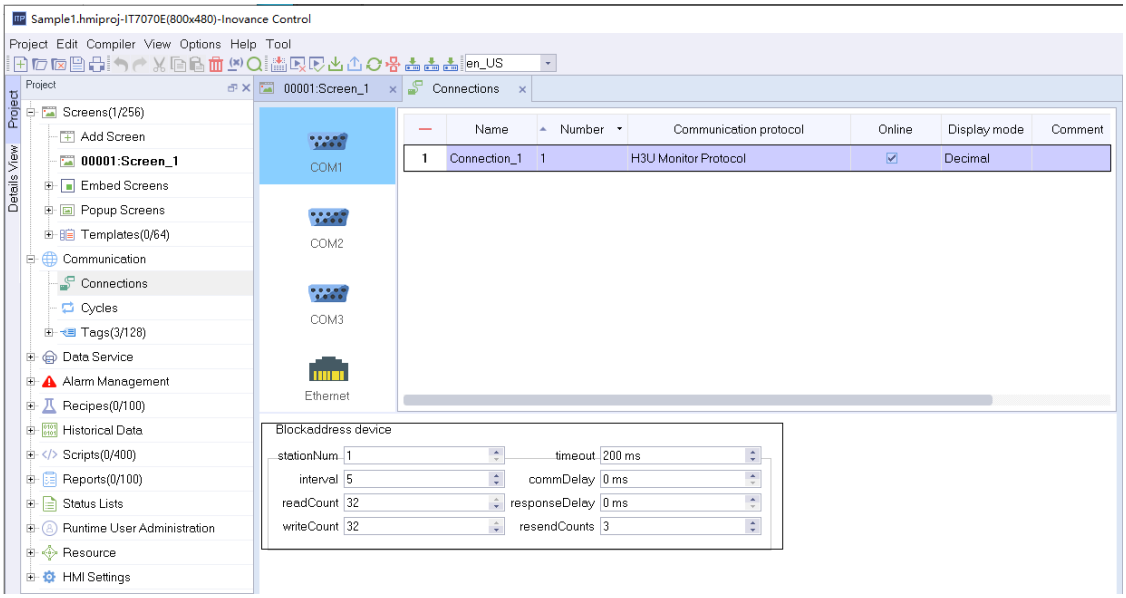
2. Select the physical port type of communication, then configure the connection, and click the + button at the top of the connection table to add a new connection. (Click − to delete the selected connection.)



	Name	Number	Communication protocol	Online	Display mode	Comment
1	Connection_1	1	H3U Monitor Protocol	☑	Decimal	

According to the specific communication mode, you can select the corresponding protocol in Communication Protocol. See Supported Protocols for details.

3. In the parameter window, you can set the corresponding parameters of the selected communication protocol, establish a connection, and select the default parameter settings in Parameter Settings, as shown in the following figure:



Note

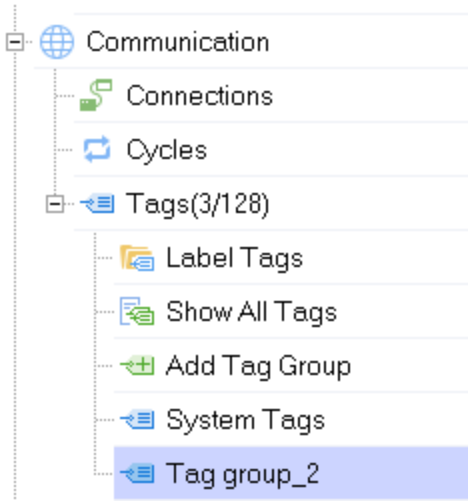
During Ethernet communication, the HMI and PLC must be in the same LAN.

2.2.4 Creating a Tag

External tags created in an HMI project can be transmitted to PLC, which is convenient for data exchange between them. In the tag module, tags include internal tags, system tags and external tags. The steps of creating a tag are as follows:

1. In the project view, go to Communication→Tags.

2. In the Tags tree, there is already a Tag Group_2, and you can also add more tag groups as needed. Double-click a variable group to open the tag editor.

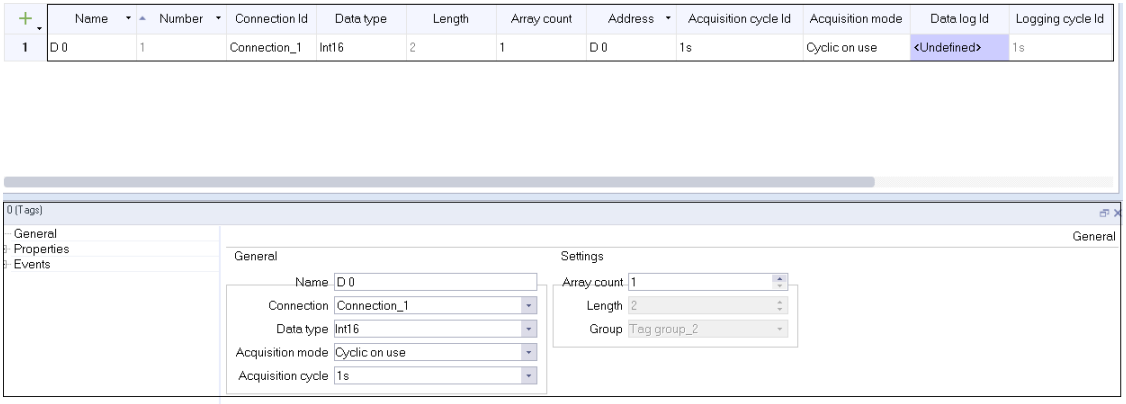


3. Click the “+” button in the workspace of the tag editor to create a tag.

	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycle Id	Acquisition mode	Data log Id	Logging cycle Id
1	D 0	1	Connection_1	Int16	2	1	D 0	1s	Cyclic on use	<Undefined>	1s

You can right-click the header to show hidden fields.

4. In the properties view, you can set other properties of the tag as needed.



	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycle Id	Acquisition mode	Data log Id	Logging cycle Id
1	D 0	1	Connection_1	Int16	2	1	D 0	1s	Cyclic on use	<Undefined>	1s

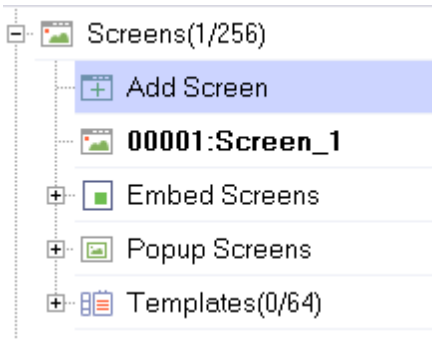
5. Repeat the above steps to configure other tags or tag groups.

2.2.5 Configuring a Screen

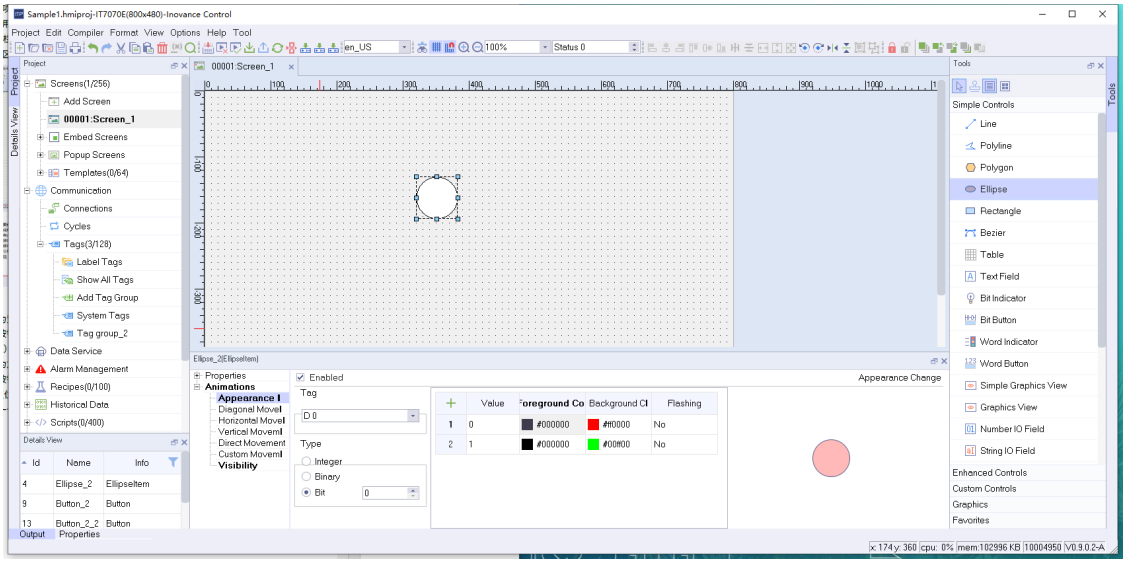
The screen is the main element of an HMI project. It enables the interaction between the HMI and the users.

See the previous sections to configure communication connections and related tags, and then create controls on the screen (indicator lights and control buttons) in the following steps:

1. Open the project and enter the default screen (Screen_1), or open Screen_1 from the Project tree view:

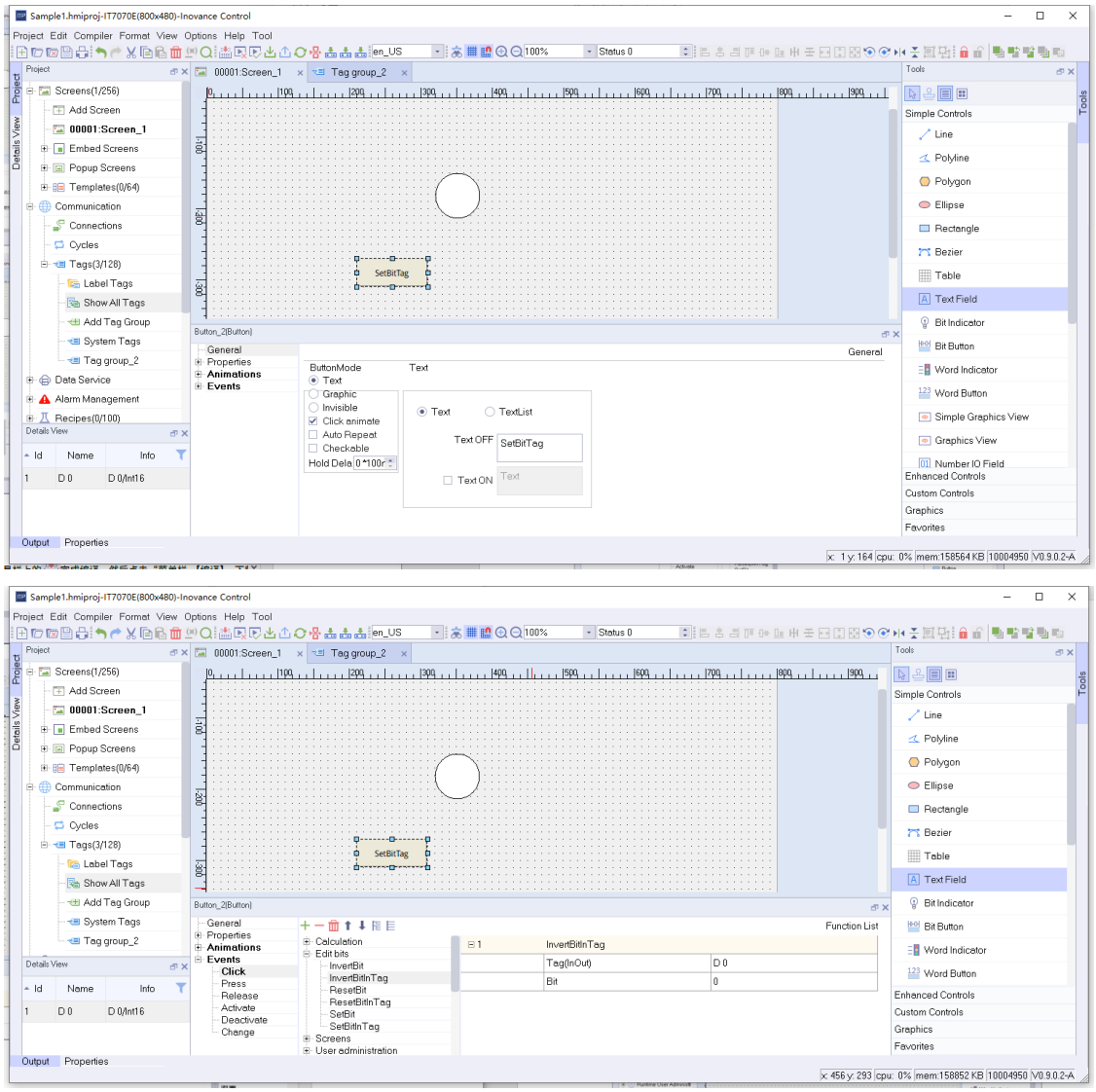


2. Expand Simple Controls and drag and drop Ellipse in the Tools window onto the screen. Under its Animations property, click Appearance Change and check Enabled.
   As shown in the following figure, set the tag to D0, and the type to Bit. Click the "+" button in the table to add two bit numbers. Set the background color of bit 0 row to #ff0000, and that of the bit 1 row to #00ff00. (The background colors correspond to different states of the indicator light, which is easy to distinguish. You can set the colors as needed.)



3. Drag and drop a button control onto the screen to control the flashing of the indicator.

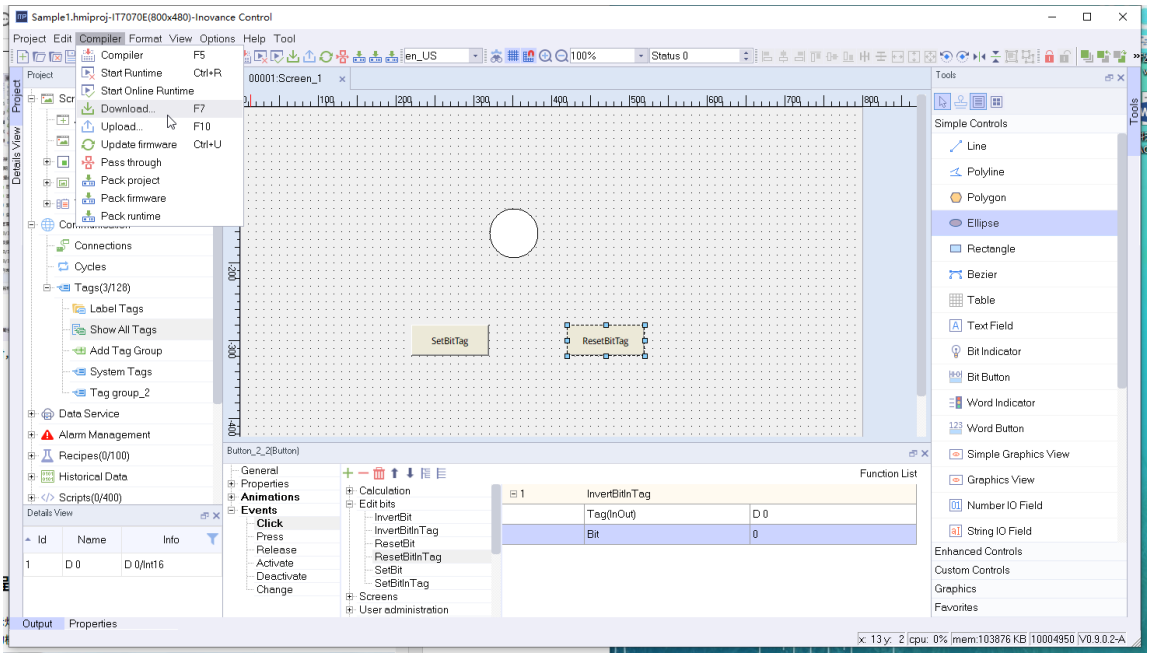
- To set the button function to Set, in General property setting of the button, enter "Set" as the Text OFF text. Then select the button trigger mode in Events and associate the system function SetBitInTag with the button event;
- To set the button function to Reset, in General property setting of the button, enter "Reset" as the Text OFF text. Then select the button trigger mode in Events and associate the system function ResetBitInTag with the button event;
- The button is associated with the same tag D0 as the ellipse.



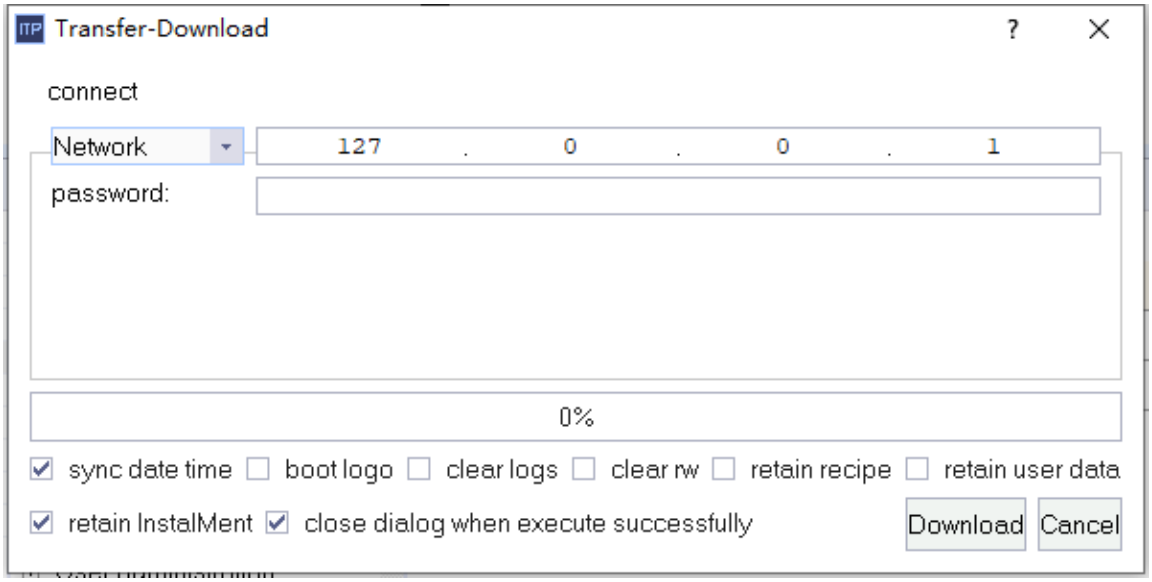
2.3 Downloading a Project

After creating the project, click Compiler→Compiler on the menu bar or on the toolbar to compile the project, and then click Compile→Download or on the toolbar or press F7 to download the project to the HMI.

You can download the project through Ethernet and USB. (Note: The project size cannot exceed 30 M).



During downloading, if the user has set the download password, you must first enter the correct password.

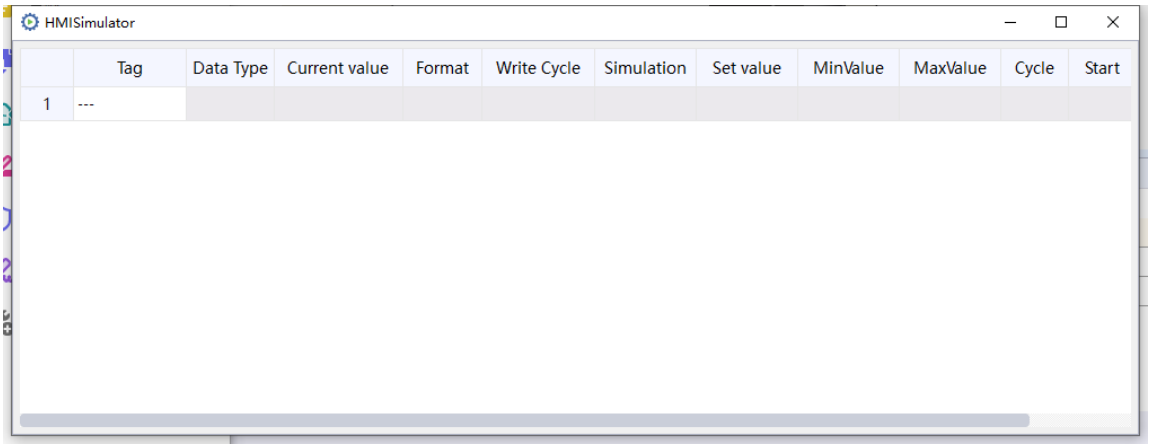


Note

The default IP address of the IT7000 series HMI is 192.168.1.100.

2.4 Simulation and Run of a Project

When the project configuration is completed, select Compiler → Start Runtime/Start Online Runtime from the menu bar or click the button in the toolbar to run the project. As shown in the following figure, you can change the setting value in the simulator to observe the dynamic change of the object in the screen.



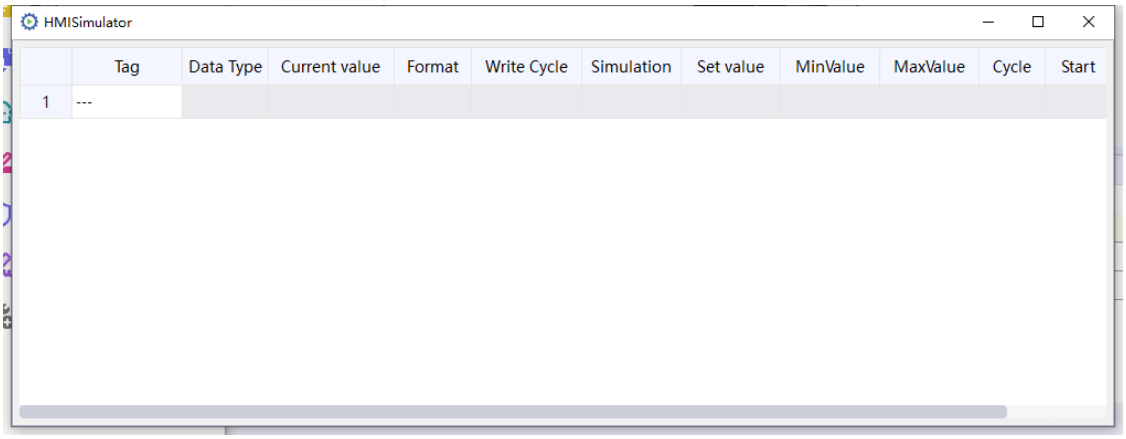
InoTouchPad offers an offline simulator and an online emulator. The two simulators allow you to run a project offline and online respectively. The simulator is a stand-alone application to test a configured project.

Starting the offline simulator

Note: The offline simulator does not include the controller.

Steps:

1. Create a project.
2. Save and compile the project.
3. Select Compiler → Start Runtime from the menu bar or click the button in the toolbar to open the simulation table and display the screen. When the project is simulated for the first time, a blank simulation table is created, as shown below.
4. Open the configuration tag in the simulation table.
5. Now you can manipulate the tags and values of the project in the simulation table. Observe the changes of configured objects in the simulator.



Result:

When you click the Set button, the indicator light turns green and when you click the Reset button, and the indicator light turns red.

Starting the online simulator

Note: The online simulator includes the controller. Therefore, the external tag is only effective when the programming device is connected to the controller. By connecting the PG/PC to the controller, you can perform online simulation of the configured HMI device in the programming device.

Steps:

1. Create a project.
2. Save and compile the project.
3. Select Compiler → Start Online Runtime from the menu bar or click the button in the toolbar to run the project.
4. Now you can manipulate the tags and values of the project in the simulator and observe the change of the screen objects. Similarly, start online simulation of the previously configured project.

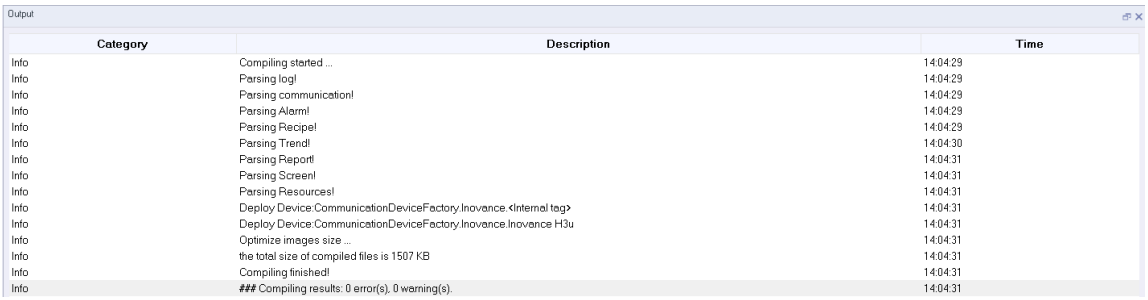
Result: When you click the Set button, the indicator light turns green and when you click the Reset button, and the indicator light turns red.

3 Project Management

3.1 Compiling a Project

After you have designed a project, you need to compile the project without any error before running it.

Select Compiler→Compiler in the menu bar or click the button in the toolbar or press F5 to compile the configured project. After the compilation is completed, the output view shows the result, as shown in the following figure:



The result provides three types of information: description, warning and error.

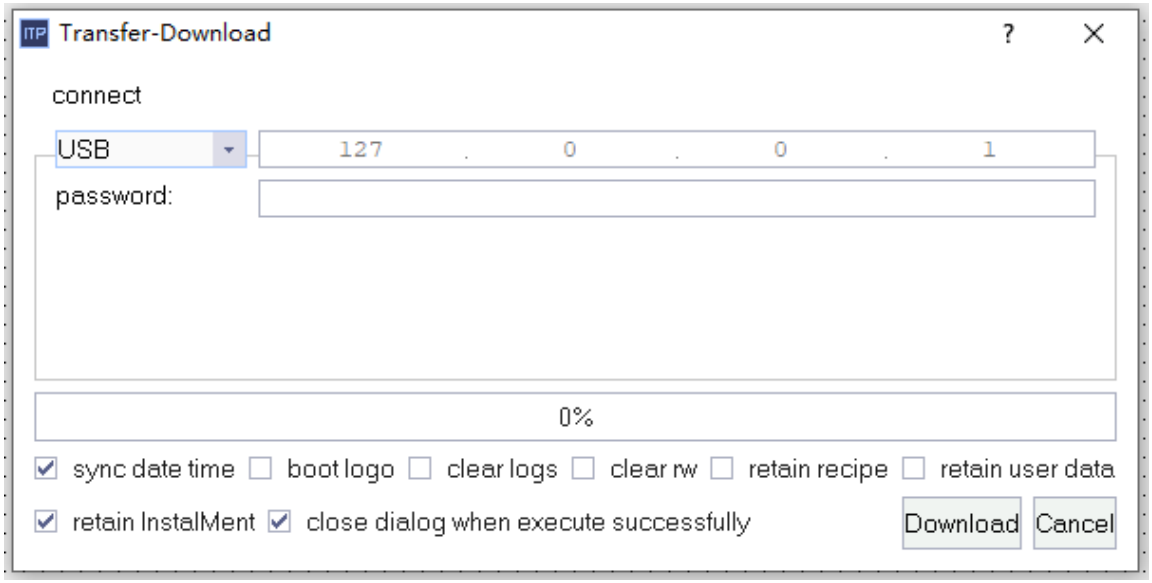
If some control properties are set incorrectly in the configuration, warnings or errors will be reported at compile time. You can locate those warnings and errors by clicking an item in the output window, and then modify the project to eliminate the warnings and errors.

3.2 Uploading and Downloading a Project

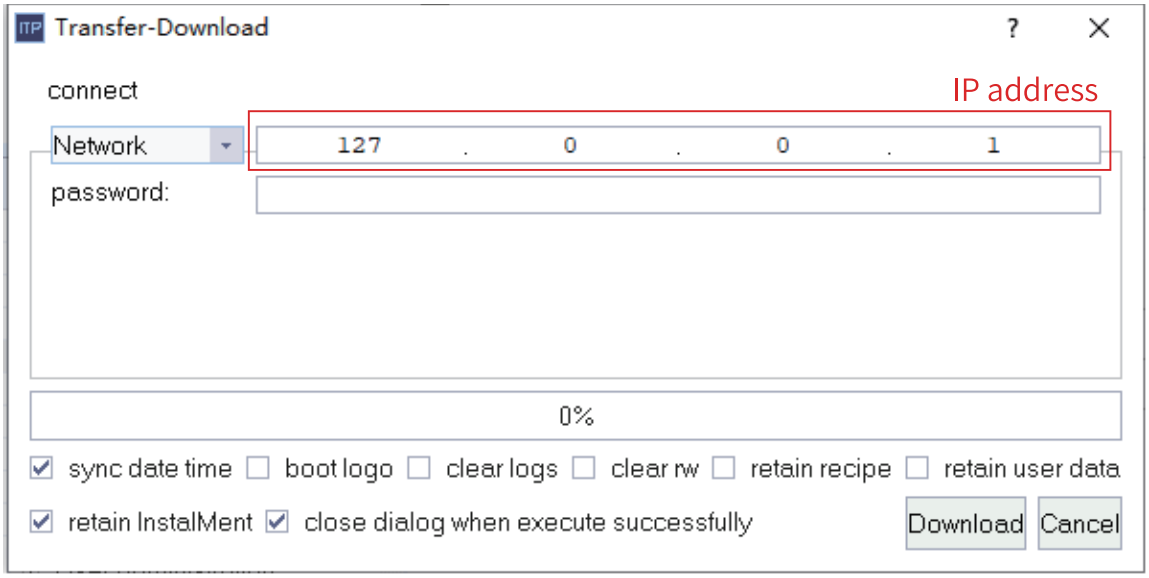
3.2.1 Download

If you want to run your project on the HMI, you need to download the project to the HMI.

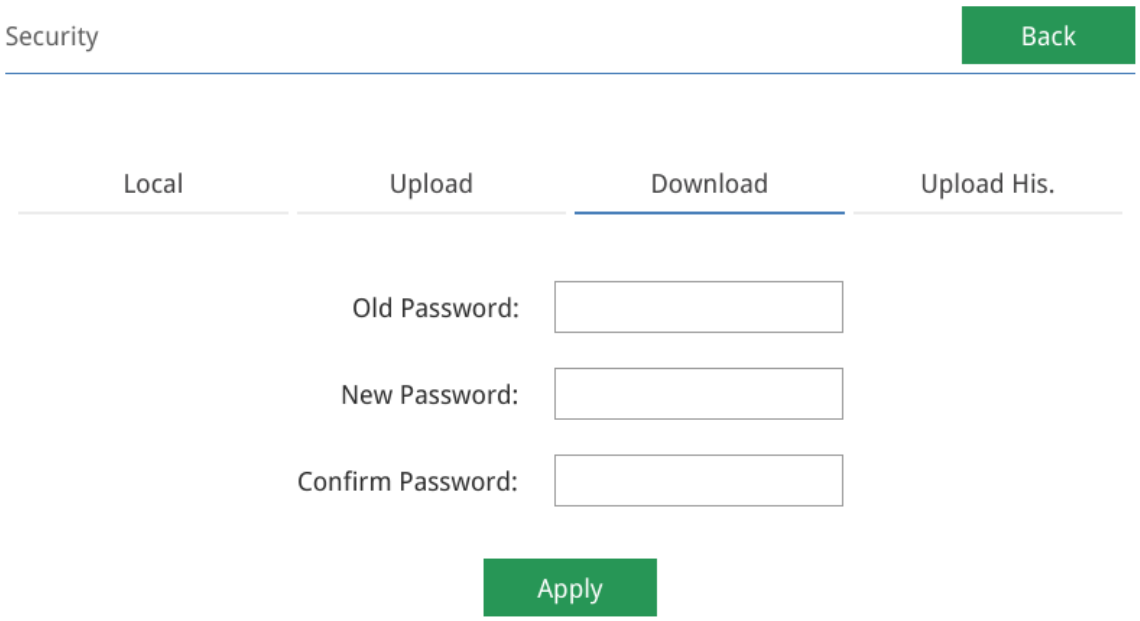
Select Compiler→Download in the menu bar or click the button in the toolbar or press F7, and the following window pops up:



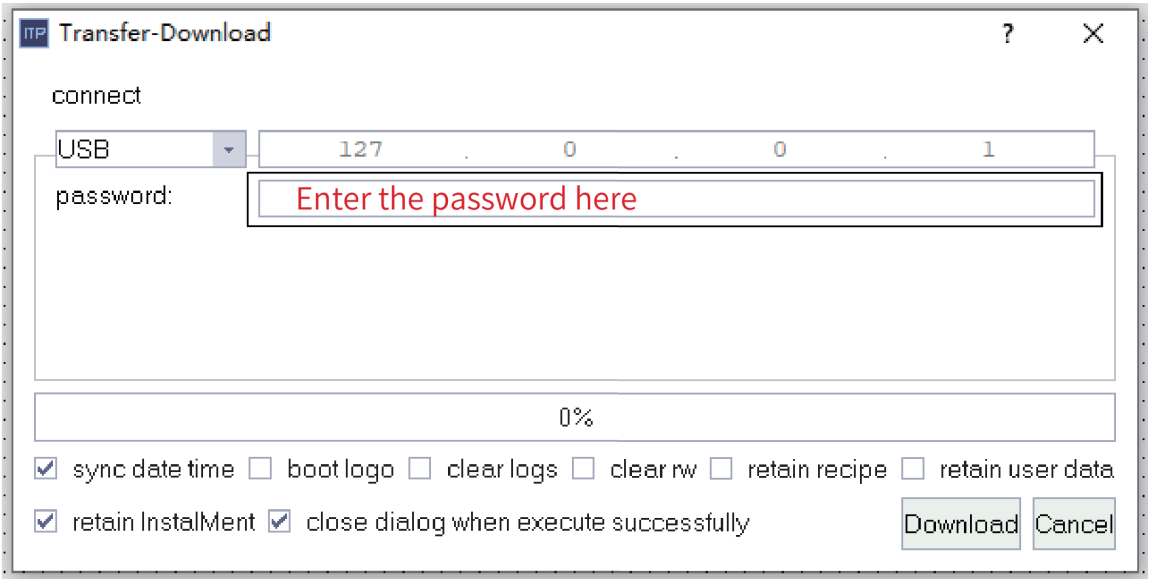
Select the connection method and input the IP address of the connected HMI:



By default, downloading requires no password. If you have set the password in Control Panel->Security of the HMI:

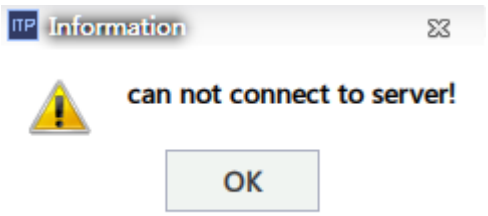


When you download the project, you must enter the password first.



In addition, in the download window, You can also check "sync date time" to synchronize the HMI time with current PC system time, check "boot logo" to customize the boot screen, check "clear logs" to clear data logs, alarm logs and operation logs and check "clear rw" to clear RW internal tag data. Other options include retaining recipes, user data and installment.

Click the Download button to begin downloading. If any error occurs during downloading, you will be prompted. For example, when the PC is not connected with the HMI, the following message box shows:

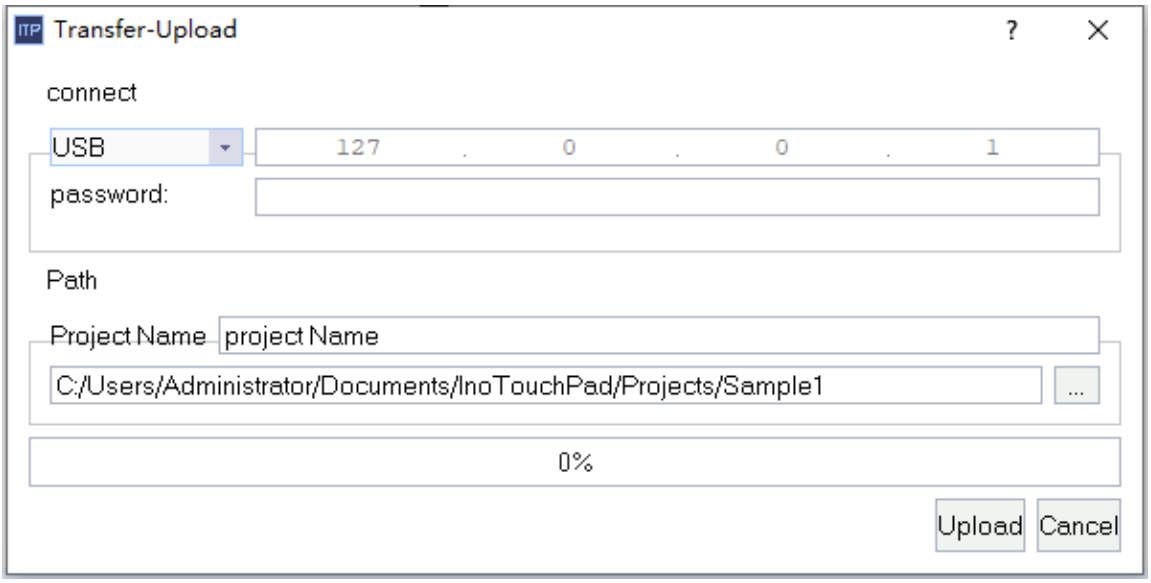


The project to be downloaded must not exceed 30 M, otherwise it cannot be compiled and downloaded.

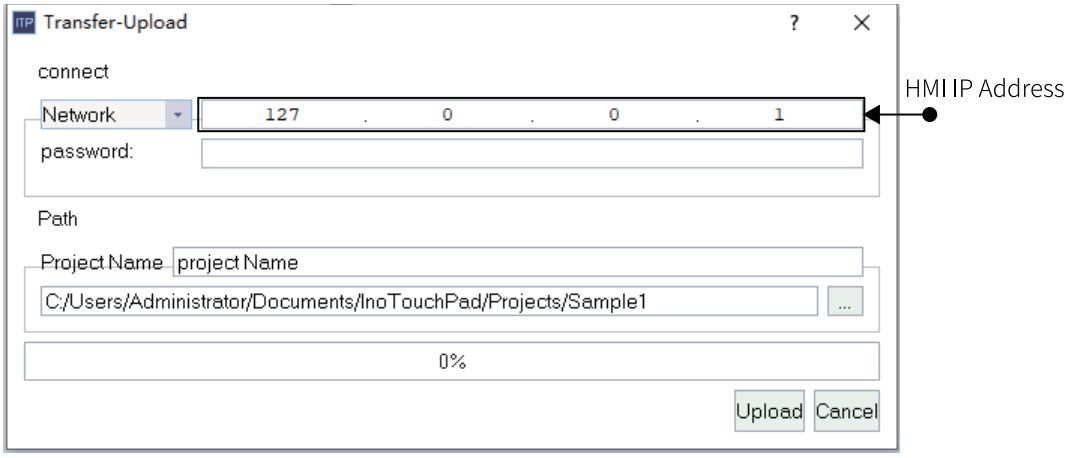
3.2.2 Upload

To upload the project running on the HMI to the PC, you can use the upload function in the software tool (note: the project size cannot exceed 30 MB).

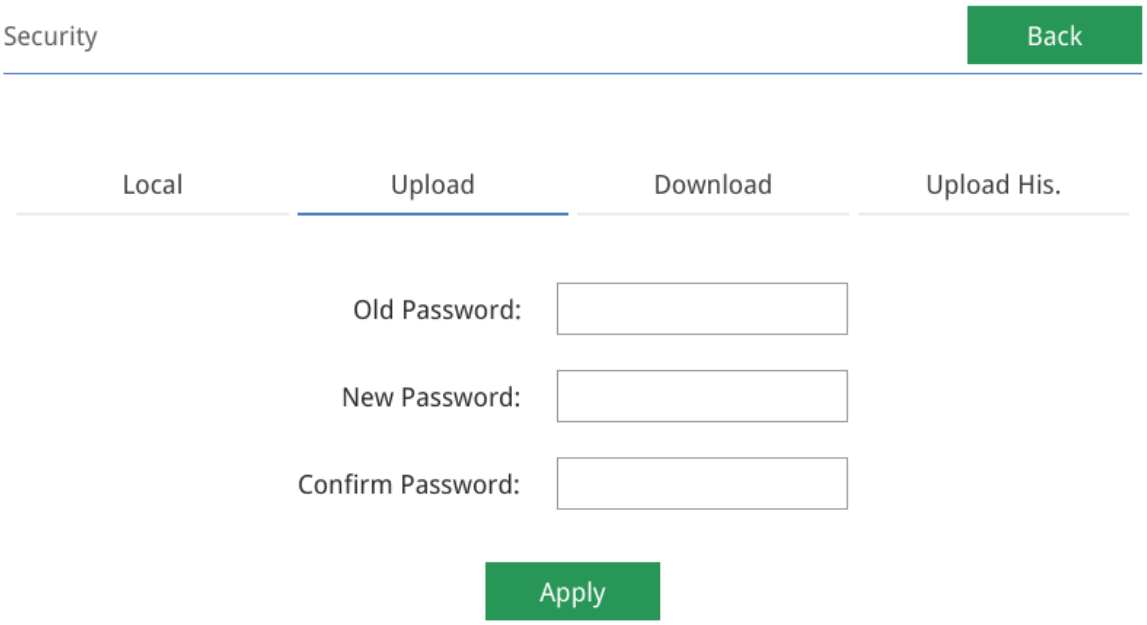
Select Compiler→Upload in the menu bar or click the button in the toolbar or press F10, and the following window pops up:



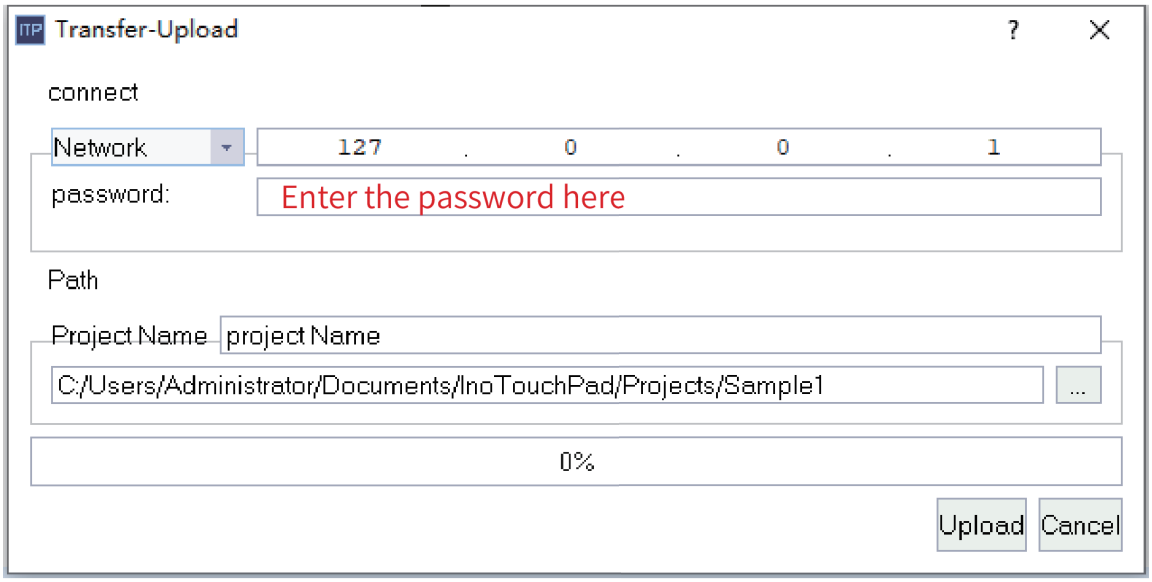
Select the connection method and input the IP address of the connected HMI:



Uploading requires no password by default. If you have set the upload password in Control Panel->Security of the HMI:



When uploading a project, you must enter the password.



In addition, you need to set the project name and select the path of the project.

Click Upload. When the progress bar reaches 100%, the upload is successful, and the currently uploaded project will be opened. If any other project is opened before uploading, you will be prompted to save that project.

3.3 Project Simulation

3.3.1 Off-line Simulation

To display the off-line simulation diagram of a new project, select Compiler→Start Runtime, or click the  button in the toolbar , or press ctrl + R. The off-line simulation includes HMI display simulation

and PLC communication simulation. The following is an off-line simulation example of the communication project between HMI and Inovance H2U, as shown in the following figure

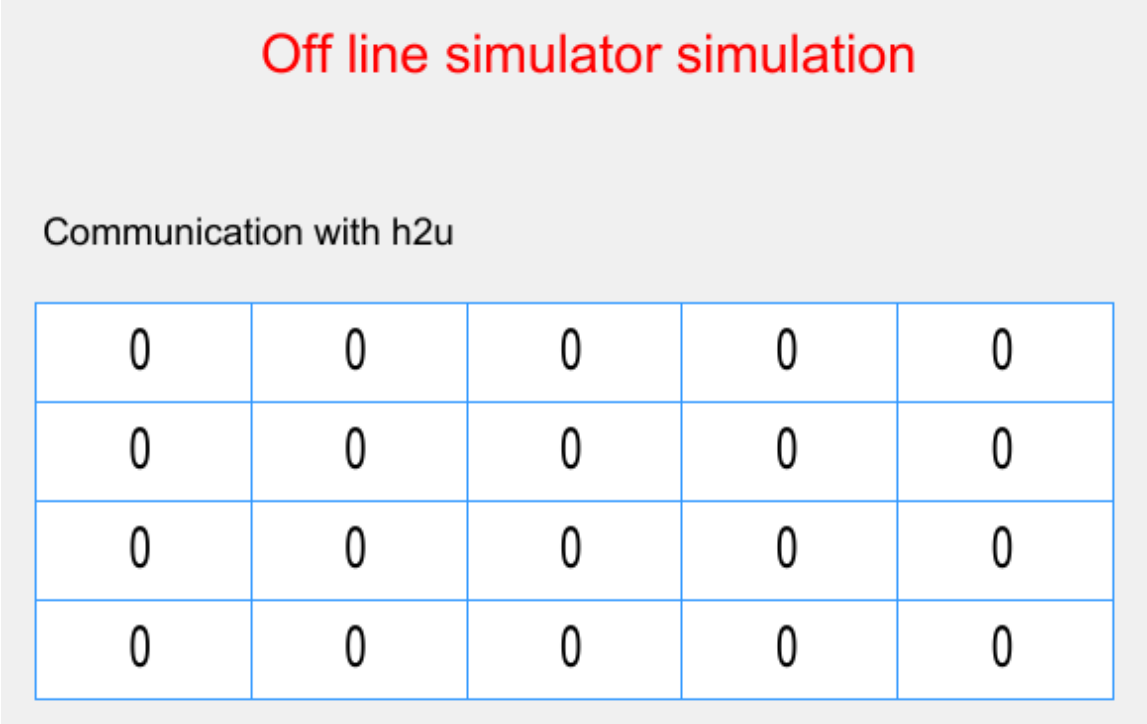


Figure 3-1 HMI simulation

	Tag	Data Type	Curren...	Format	Write Cy...	Simulati...	Set value	MinValue	MaxValue	Cycle	Start
2	D 1	Int16	0	DEC	1	display		-32768	32767		
3	D 2	Int16	0	DEC	1	display		-32768	32767		
4	D 3	Int16	0	DEC	1	display		-32768	32767		
5	D 4	Int16	0	DEC	1	display		-32768	32767		
6	D 5	Int16	0	DEC	1	display		-32768	32767		
7	D 6	Int16	0	DEC	1	display		-32768	32767		
8	D 7	Int16	0	DEC	1	display		-32768	32767		
9	D 8	Int16	0	DEC	1	display		-32768	32767		
10	D 9	Int16	0	DEC	1	display		-32768	32767		

Figure 3-2 PLC simulation

Users can input any value of the address of the configured register in the PLC simulation window to test offline simulation.

3.3.2 On-line Simulation

On the menu bar, select Compiler → Start Online Runtime or click the  button in the toolbar to start the simulation of the current project. The simulation is displayed as an HMI front panel. The following is an online simulation example of the communication between the HMI and Inovance H2U:

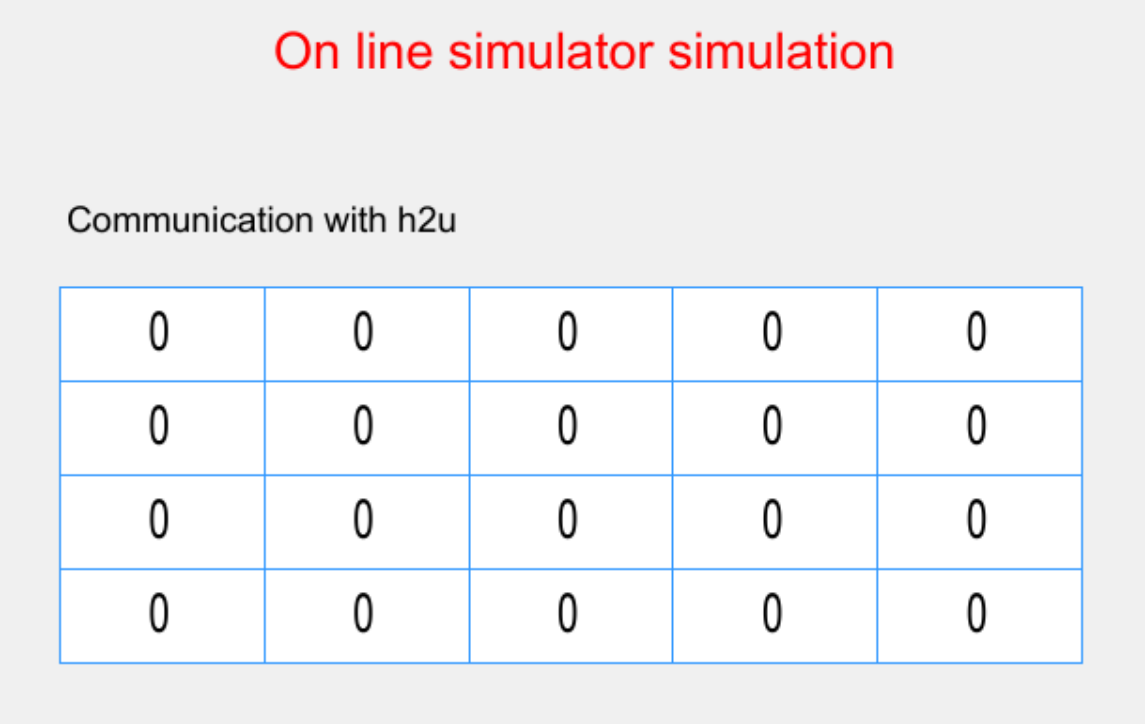


Figure 3-3 HMI simulation

You can use online simulation to simulate an HMI device on a PC and connect the HMI to a real PLC, which facilitates screen testing because you need not to download the screen program to the HMI every time. Before simulation, connect the PC and PLC through a serial port or network port communication cable. Since all the serial ports of the computer are of the RS232 type, an adapter may be needed. However, a download cable is usually used to connect the PC with the PLC.

Before connecting the PLC, you must set all communication parameters of the PLC correctly.

If you use a COM1 port to connect the PLC in HMI software settings, you must also connect the PC with the PLC through the COM1 port of the PC.

3.4 Pass-through

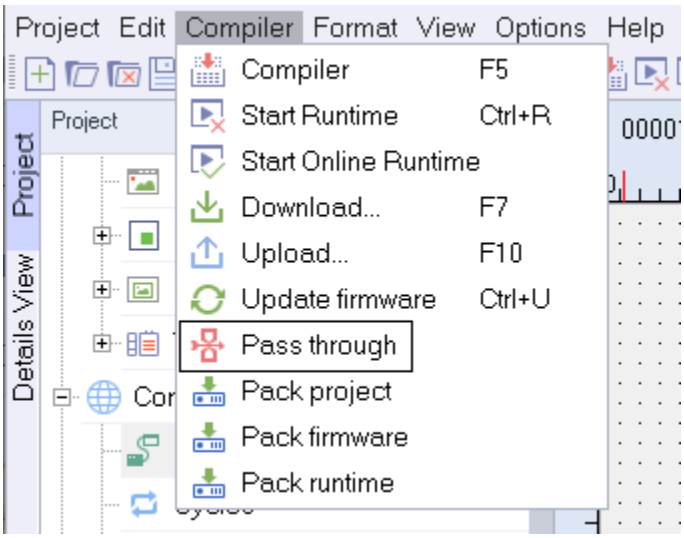
3.4.1 Application

The penetration function is specially designed for Inovance H1U, H2U and H3U. This function enables intermediate connection between the PC and PLC through the HMI.

The penetration function is mainly used in applications where the PLC is enclosed and thus can only be indirectly connected to the PC.

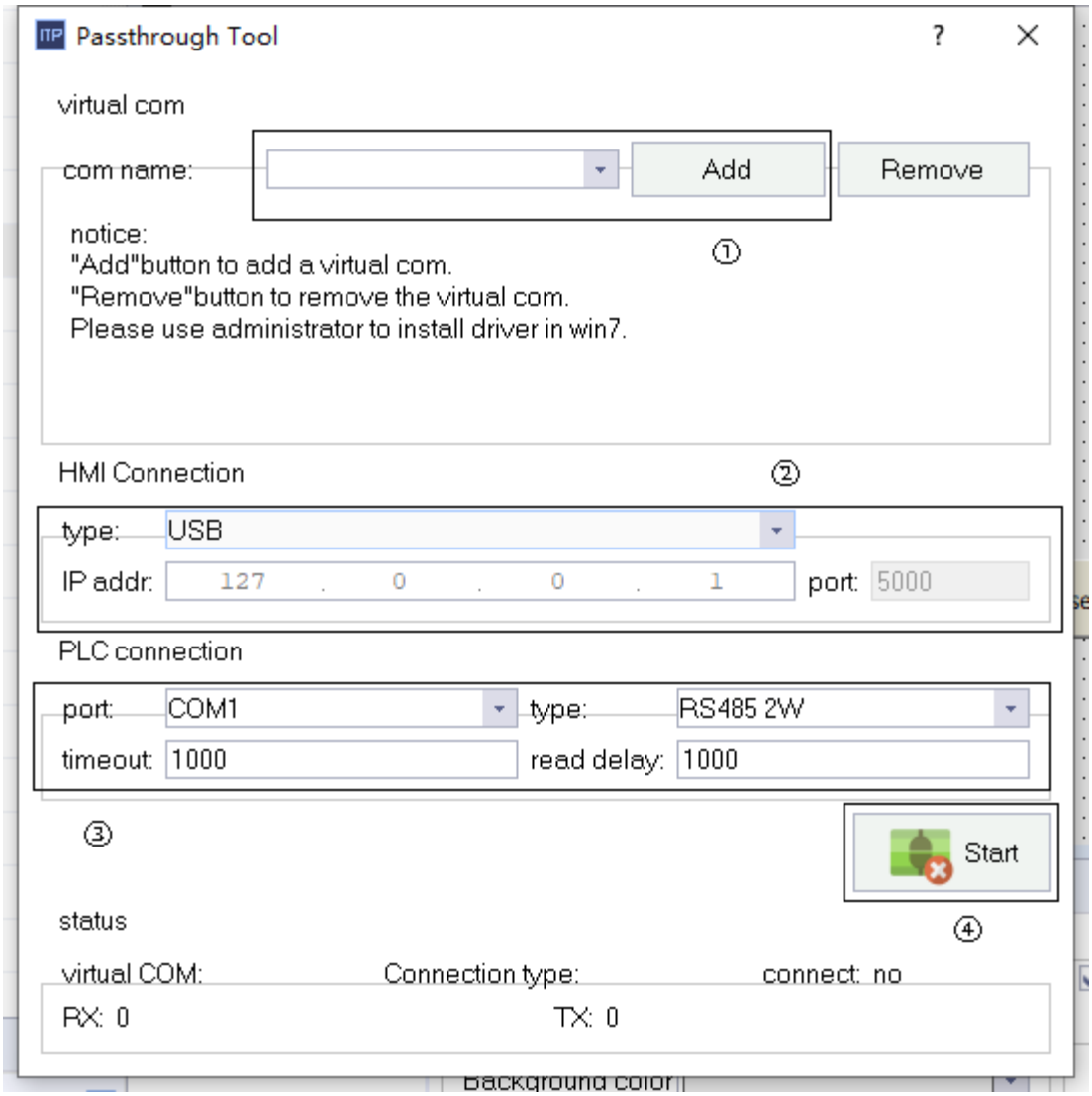
3.4.2 Steps

First, select Compiler→Pass through in the menu bar.

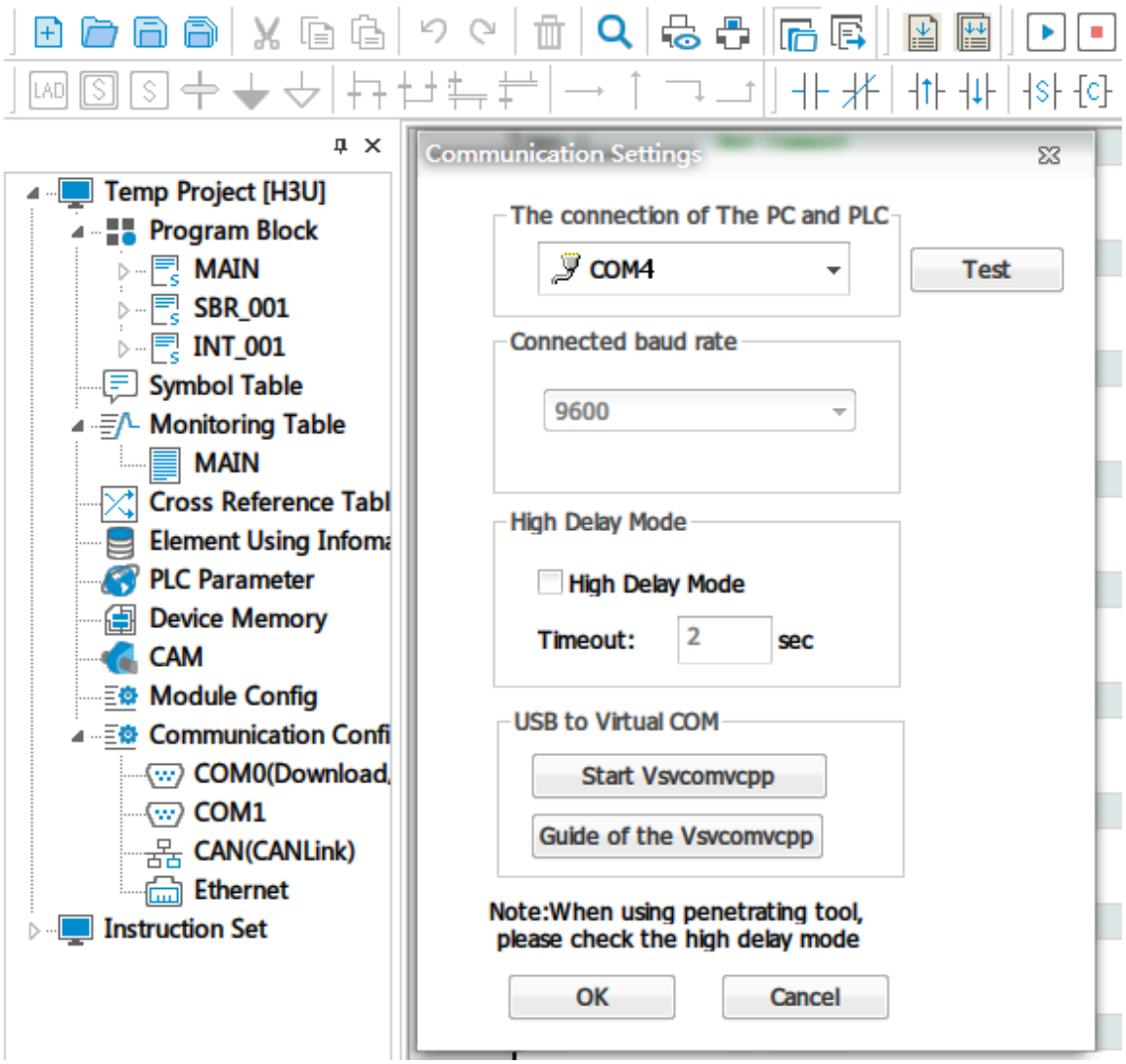


In the pop-up window, set the parameters:

1. Add a virtual serial port to connect with Inovance PLC software tool AutoShop;
2. Under HMI Connection, set the connection type between PC and HMI (if you select Net, the IP address should be HMI IP);
3. Under PLC Connection, choose the port for HMI and PLC connection and set other parameters;
4. Click Start.



After the parameters are set and the pass-through function is enable, you need to start AutoShop. As shown in the following figure, set the serial port to COM4 in Communication Settings (consistent with the virtual serial port set in the pass-through tool).



3.5 Packaging Project Files

3.5.1 Packaging a Project

In addition to InoTouchPad, you can also update an HMI project through a USB drive or SD card. To do so, you need to use InoTouchPad to package the project and transfer it to the USB drive or SD card which will be connected with the HMI.

Select Compiler→Pack Project in the menu bar or click the button in the toolbar (note the tooltip) to package the project to the specified location. After clicking OK, the current project will be packaged into an HMI project update package with a fixed name (project.zip). The packaging window is as follows:

Note that when packaging the project, check the items to be kept or cleared, including boot logo, clear logs, clear rw, retain recipe, retain record and retain Instalment. None of them are checked by default.

3.5.2 Packaging Firmware

In addition to InoTouchPad, you can also update the firmware of the HMI through a USB drive or SD card. Select "Compile" → "Packaging Firmware" in the menu bar or click the button in the toolbar (pay attention to the tooltip), and the Packaging firmware window will show:

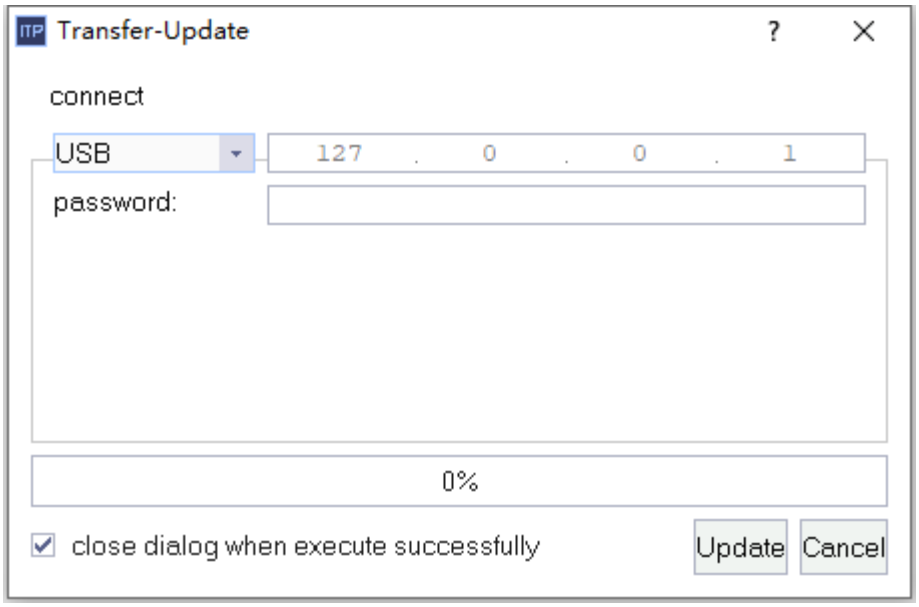
3.5.3 Packaging HMIRuntime

InoTouchPad allows HMIRuntime packaging through a USB drive or SD card. Select "Compile" → "Packaging HMIRuntime" in the menu bar or click the button in the toolbar (pay attention to the tooltip), and the Packaging HMIRuntime window will show. You can choose what to keep or clear. For details, see the project packaging instructions.

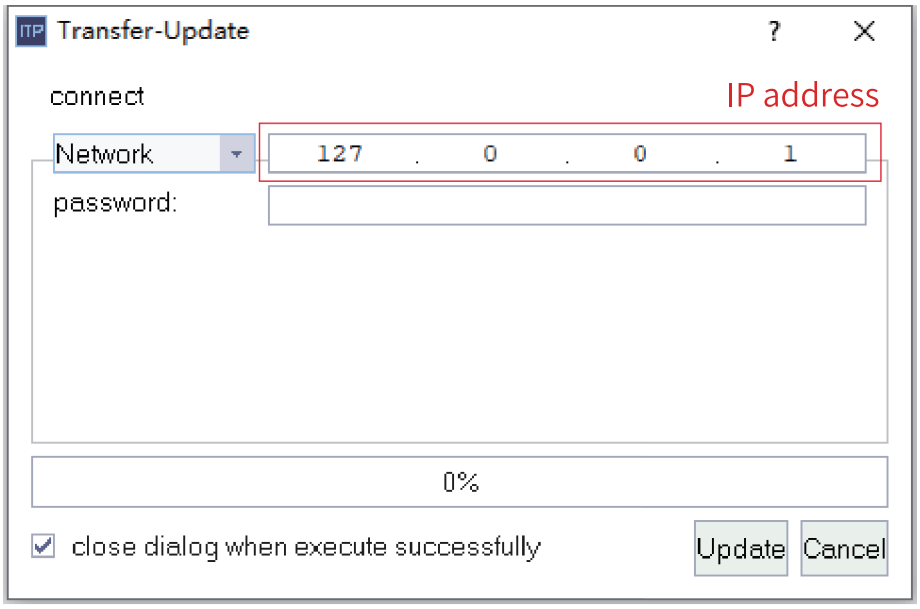
3.6 HMI Firmware Update

When you have modified the HMI firmware to adjust the functions of the HMI, you need to update the modified firmware to the HMI. Generally, the firmware update occurs after InoTouchPad update is released, and the version of the update must be higher than the old firmware.

Select Compiler→Update firmware in the menu bar or click the button in the toolbar or press ctrl +U, and the following window pops up:



Select the connection method and input the IP address of the connected HMI:



The update requires no password by default. If you have set the update password in the control panel of the HMI (the download password is used as the update password), you need to enter the correct password before updating.

In addition, you also needs to set the storage path of the firmware (usually the installation path of InoTouchPad).

Click Update. When the progress bar reaches 100%, the firmware update is successful and the HMI will be started automatically. If any error occurs, you will be prompted.

3.7 Font Setting

See *"13.3 Language and Font Setting" on page 289*.

3.8 Notes for the Use of Peripherals

The HMI allows you to manage project files with a USB drive or SD card. The supported file systems are FAT32, NTFS and exFAT. This section briefly describes their OS compatibility and features.

File System	FAT32	NTFS	exFAT
Compatibility (R/W)	Windows, Mac OS, Linux	Windows	Windows, Mac OS
Features	Low disk utilization, an individual file cannot be larger than 4 GB, best compatibility	Good security and stability, but less favorable compatibility	Support for flash memory and large files

To sum up, it is recommended to use a USB drive or SD card in FAT32 format to manage HMI project files.

3.9 HMI Common Errors

When you use InoTouchPad to download a project or update the firmware, the following errors may occur.

Type	Error Code and Description
HMI communication	10004: write error
	10005: read error
	10006: device start error
	10007: timeout
	10008: device failed to start. For 10004–10008, you need to check the connection at the USB or network port
HMI operation	10003: device type mismatch
	10002: incorrect password
	10001 or 205: incorrect HMI version
	100: incorrect password (when you try to use pass-through)
	102: HMI is operated by another user (when you try to use pass-through)
	3: Please check that you have the proper permissions
	4: Insufficient resources, more flash space needs to be released
	200–202: HMI upload error. You need to check whether the connection is unstable
	302, 303, 203, 204: CRC32 error. Check the file
	301, 400, 401, 10009: Failed to unzip or open the file. More flash space is needed.

4 Communication

4.1 Introduction

4.1.1 Basic information

Data exchange between devices is called communication. Devices can be networked and interconnected through communication media such as serial communication cables, network cables and wireless modules.

Communication device

A communication device is any device that can access the communication network and exchange data, which is usually called a communication node. In InoTouchPad, an HMI or PLC is generally considered a communication node.

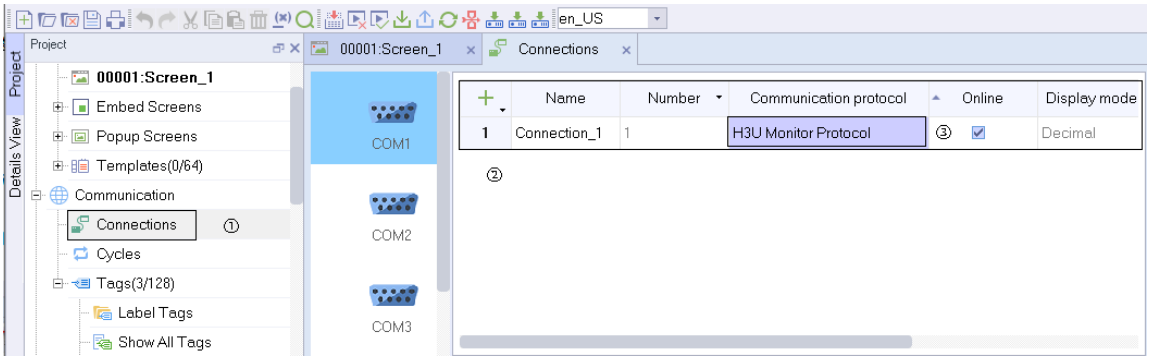
Use of communication data

Data transferred between devices can be used for project control, process data collection, process status reporting, and process data recording.

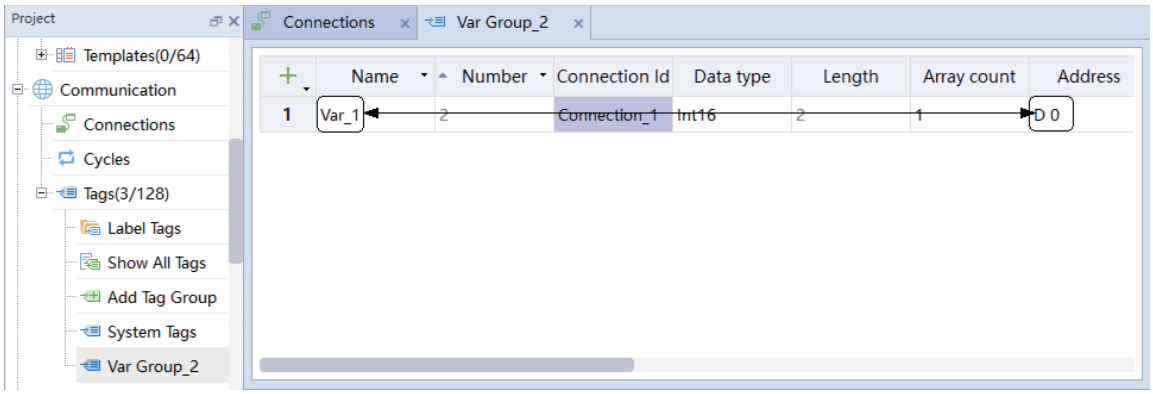
4.1.2 Communication Principle

InoTouchPad controls communication between HMI and PLC or between HMIs by mapping device register addresses to communication variables through the selected protocol.

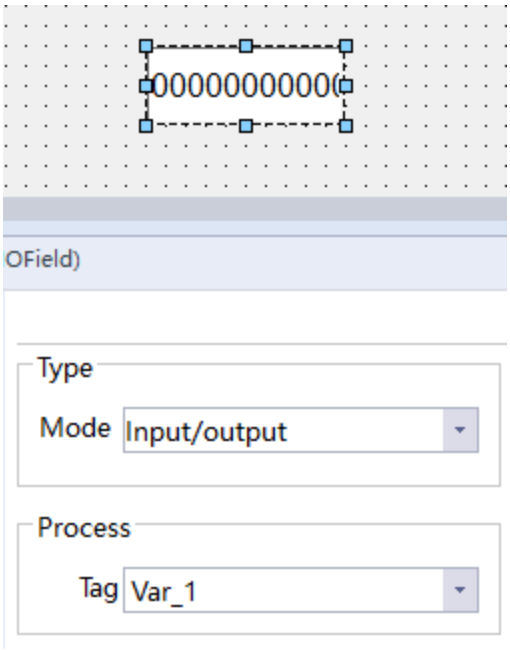
For example, in InoTouchPad, go to Connections, create a connection_1 connection, and configure an inovance H3U monitoring protocol.



Then, in Tag group_2 of Tags, create a tag, configure the newly created connection_1, and map the Inovance H3U register D 0 address to this tag.



In the screen, create a number IO field, configure tag_1, and download it to the HMI. Then connect the H3U device. The number IO field can access the value of the register with the address of D 0 through the tag name.



4.1.3 Communication Tag

Communication tags, which are managed in the tag editor, include internal tags and external tags. Internal tags are defined memory images on an HMI, while external tags are defined memory images on a PLC, all of which can be used for communication. During communication, these defined memory images can be read and written by devices. This is usually done in a periodic or event-triggered manner. See *"5.3.1 Using Tags in Communication" on page 69* for details.

After you create an address tag, the HMI can read the value from the address and display the value. You can also write the tag value to the specified address on the HMI.

4.1.4 Protocol

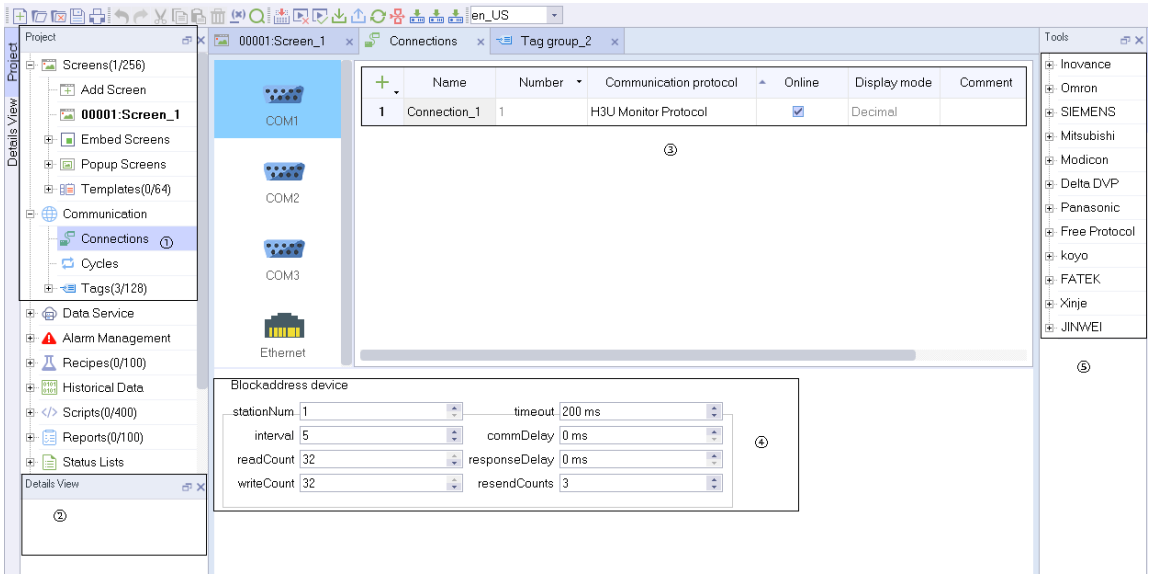
Different drivers enable different device connections. Users can choose a specific communication interface, configuration and transmission speed for an individual device. For a list of supported protocols, see *"4.3.1 Supported Protocols" on page 56*.

4.2 Basic Setting

4.2.1 Connection Configuration

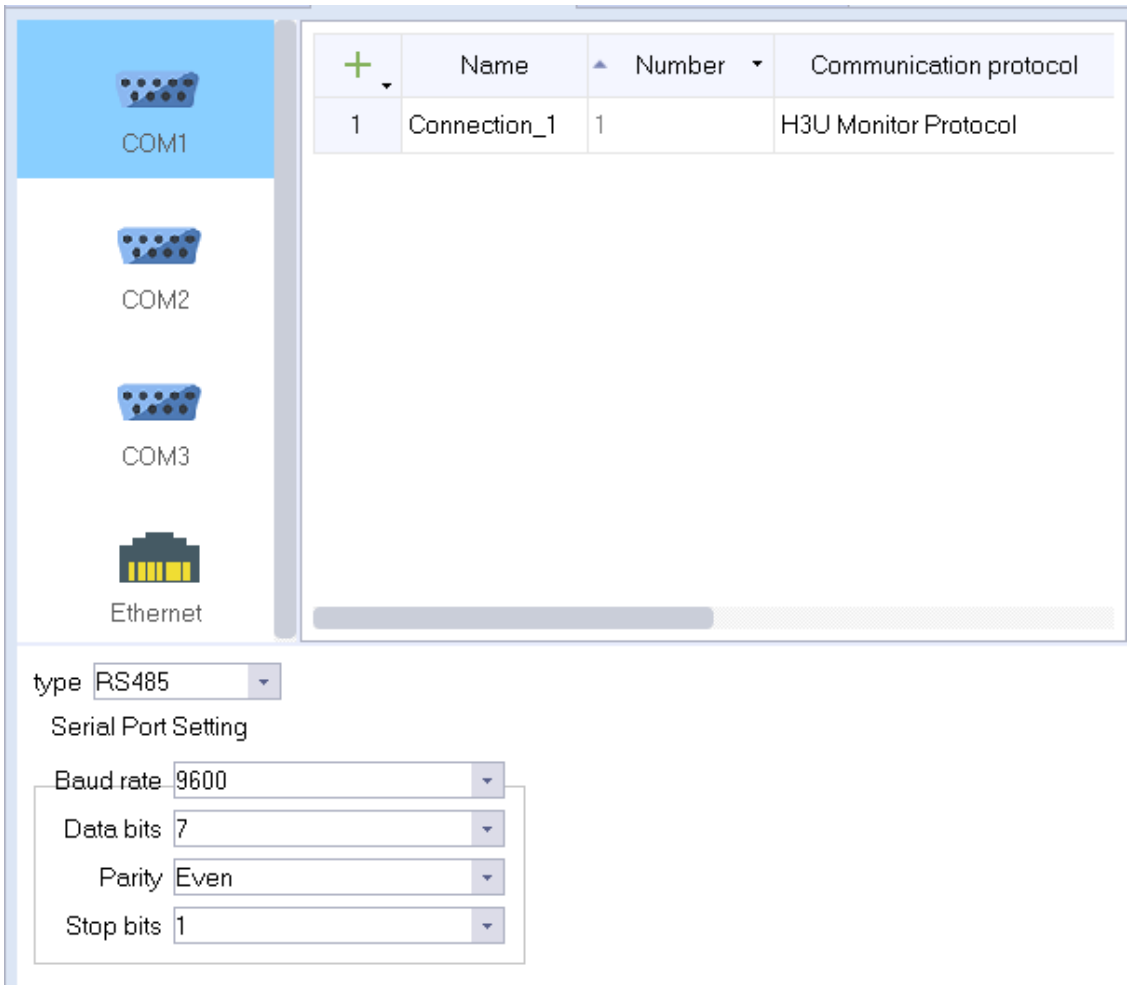
The configuration screen of a HMI project is as shown in the following figure:

1-project view; 2-detail view; 3-workspace; 4-parameter setting area; 5-Tool area



Steps:

1. Expand Communication in the project view.
2. Double-click Connections to open the connection editor.
3. Click the + button in the workspace to create a connection.
4. You can customize the parameters as needed. If you need to set the physical serial port communication terminal, you can click a port to show the serial port setting.

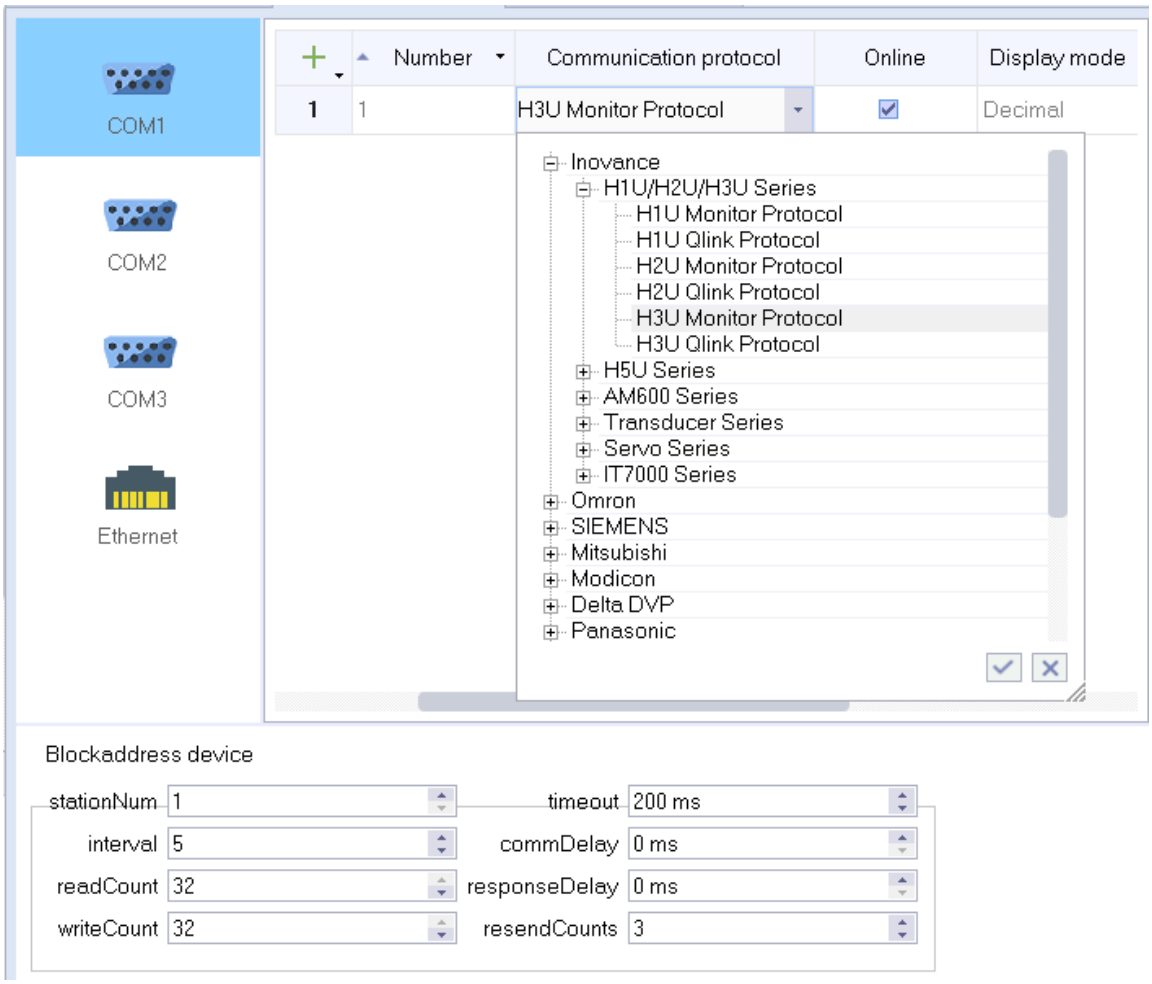


5. In the communication protocol, select the communication protocol driver suitable for your PLC.
6. Then set the parameters accordingly. By default, the communication parameters, timeout, communication delay and response delay, are auto-tuned during communication.
7. Save the project to finish the connection configuration.

4.2.2 Connection Editor

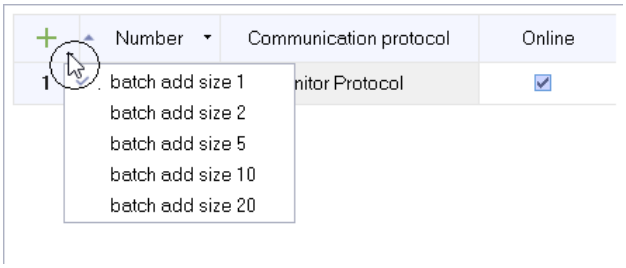
In the connection editor, you can create and configure connections. With the tabular connection editor, you can set the connection name, communication protocol, default status of the protocol, and comments. After selecting the communication protocol, you can edit the related attributes.

Click the + button in the upper left corner to create a connection, and then select the protocol in the Communication Protocol field, as shown in the following figure. Note that the drop-down list box and the communication protocol tree in the right sidebar support filtering according to the current physical port.



You can also directly select the protocol in the right sidebar, and then double-click the protocol to create a connection, as shown in the following figure:

If you want to create multiple connections at one time, you can also click the button in the upper right corner to select a batch operation in the drop-down menu, as shown in the following figure:



For example, if you select "batch add size 5", and then click the button, the result is as follows:

Connections

	Name	Number	Communication protocol	Online	Display mode
1	Connection_1	15	H3U Monitor Protocol	☑	Decimal
2	Connection_2	16	H3U Monitor Protocol	☑	Decimal
3	Connection_3	17	H3U Monitor Protocol	☑	Decimal
4	Connection_4	18	H3U Monitor Protocol	☑	Decimal
5	Connection_5	19	H3U Monitor Protocol	☑	Decimal

Connection state

在线

You can check the check box ☑ in the Online field to activate the connection after the HMI starts. The connection status can be changed by system functions.

4.2.3 Connection Parameter

After selecting a protocol in the connection editor, you can configure the connection parameters of HMI and communication device.

Supported interfaces:

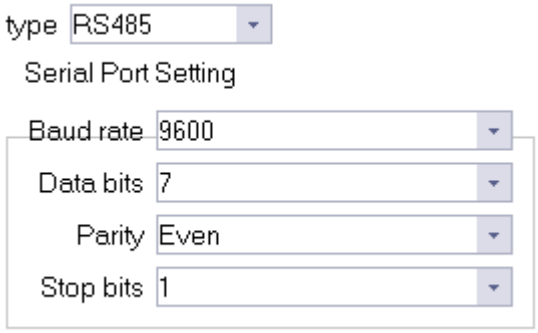
Interface	Type	Description
COM1	RS485	Up to 32 protocols (of the same family) can be configured, but they cannot cross.
	RS422	
COM2	RS232	Only 1 protocol can be configured
COM3	RS485	Up to 32 protocols (of the same family) can be configured, but they cannot cross.
Ethernet	Ethernet	TCP/IP protocol supports configuration, provided that IP + Port is unique.

4.2.4 HMI as a Master

When an HMI device needs to access the values of registers of other devices, use a master protocol (all are master protocols except Modbus Slave and Modbus TCP Slave protocols).

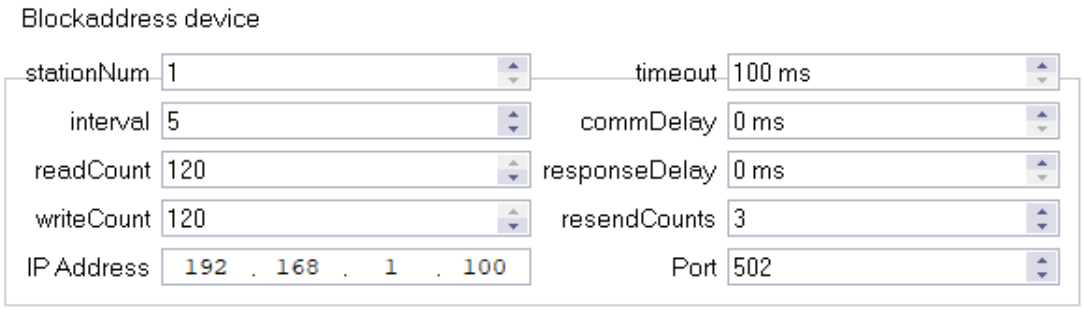
1. For communication through a serial port, you can configure the serial port and the communication parameters of the connected devices as shown in the following figure.

   - Configuration interface for physical port



- Configuration interface for communication protocol

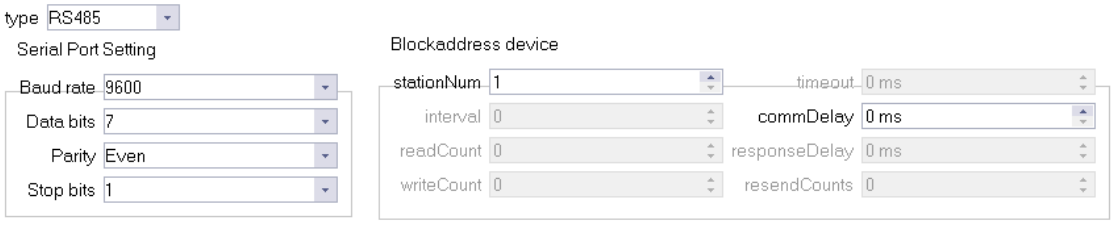
2. For communication through Ethernet, you only need to configure the parameters of the connected devices as shown in the following figure.



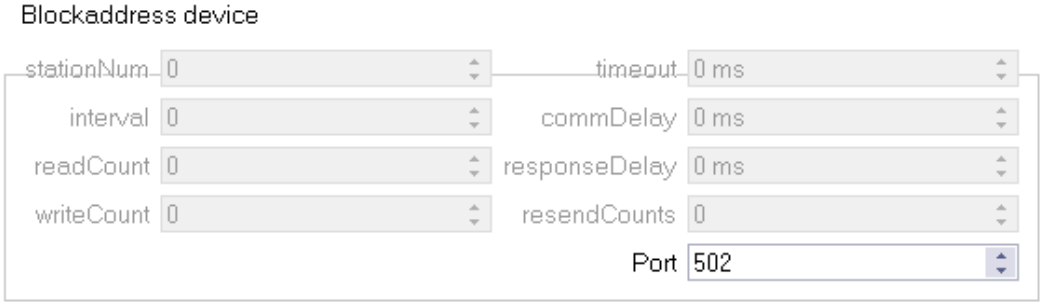
4.2.5 HMI as a Slave

When the internal image address of the HMI needs to be accessed by other devices, use a slave protocol (Modbus Slave and Modbus TCP Slave are both slave protocols).

1. For communication through a serial port, you can configure the serial port and slave parameters as shown in the following figure.



2. For communication through Ethernet, use the default settings in the following figure.



4.3 Supported protocol

4.3.1 Supported Protocols

At present, InoTouchPad supports all Inovance PLC protocols, standard modbus protocol, and Omron FINS protocol. In addition, InoTouchPad also supports user-defined protocol (free protocol supporting serial port and Ethernet communication), which makes communication more flexible. The protocol applies to all devices that support serial port and Ethernet communication. The protocol is as follows:

Manufacturer	Protocol		Description
Inovance	H1U/H2U/H3U series	H1U monitoring protocol	Monitoring protocol, supporting Inovance H1U devices
		H1U Qlink	A variant of Modbus family, supporting Inovance H1U devices
		H2U monitoring protocol	Monitoring protocol, supporting Inovance H2U devices
		H2U Qlink	A variant of Modbus family, supporting Inovance H2U devices
		H3U monitoring protocol	Monitoring protocol, supporting Inovance H3U devices
		H3U Qlink	A variant of Modbus family, supporting Inovance H3U devices
		H3U Qlink TCP	A variant of Modbus family, supporting Inovance H3U devices
		H3U Modbus TCP	A variant of Modbus family, supporting Inovance H3U devices
	H5U Series	H5U Qlink TCP	A variant of Modbus family, supporting Inovance H5U devices
	AM600 Series	AM600 Modbus	A variant of Modbus family, supporting Inovance AM600 devices
		AM600 Modbus TCP	A variant of Modbus family, supporting Inovance AM600 devices
		AM600 Qlink TCP	A variant of Modbus family, supporting Inovance AM600 devices
	AC810 Series	AC810 Qlink TCP	A variant of Modbus family, supporting Inovance AC810 devices
	AC drives	MD series AC drive Modbus protocol	A variant of Modbus family, supporting Inovance AC drives
	Servos	IS series Servo Modbus protocol	A variant of Modbus family, supporting Inovance Servo devices
	IT7000 Series	Modbus slave	Modbus slave, used when an Inovance HMI is used as a slave
		Modbus TCP slave	Modbus TCP slave, used when an Inovance HMI is used as a slave
Omron	FINS	FINS serial port protocol	FINS protocol, supporting Omron FINS-enabled devices
		FINS TCP	FINS protocol, supporting Omron FINS-enabled devices
	Hostlink	Hostlink	

Manufacturer	Protocol		Description
Siemens	Siemens Ethernet series	S7-200 SMART (Ethernet) protocol	
		S7-300 (Ethernet) protocol	
		S7-1200\1500 (Ethernet) protocol	
	Siemens serial port protocol	S7-200	
		S7-200 Smart protocol	
Mitsubishi	Mitsubishi Ethernet series	MelSec-Q\L (Binary)	
		MelSec-Q\L (ASCII)	
		FX5U (ASCII)	
	Mitsubishi serial port series	FX0SFX0NFX1SFX1NFX2	
		FX2N	
		FX3U\FX3G	
		FX5U	
Modicon	Modbus		A variant of Modbus family, supporting MODBUS-enabled devices
	Modbus TCP		A variant of Modbus family, supporting MODBUS-enabled devices
Delta	Delta DVP RTU		
	Delta DVP ASCII		
Panasonic	Panasonic FP		
Free Protocol	Free protocol		
	Free TCP protocol		
Koyo	SN series	SN K series	
Yudian	Serial Port	AIBus	
Fatek	FB/FBS series	FATEK Fbs	
Xinjie	XC series protocol		
	XD\XL series protocol		
Keyence	Network port	KV-L120\700\1000\3000 \5000\7500\Nano	
Yaskawa	Network port	Yaskawa MP23XX	
Kingway	Kingway 984		

The communication error codes are as follows:

Protocol	Error Code and Description
Inovance monitoring protocol	0x10: check error
	0x11: incorrect data length
	0x12: unknown operation
	0x15: 15 error
Modbus family protocols	0x01: illegal function
	0x02: Illegal data address
	0x03: Illegal data value
	0x04: slave device failure
	0x05: Acknowledge
	0x07: slave busy
	0x08: storage parity error
	0x0a: unavailable gateway path
	0x0b: gateway target device failed to respond

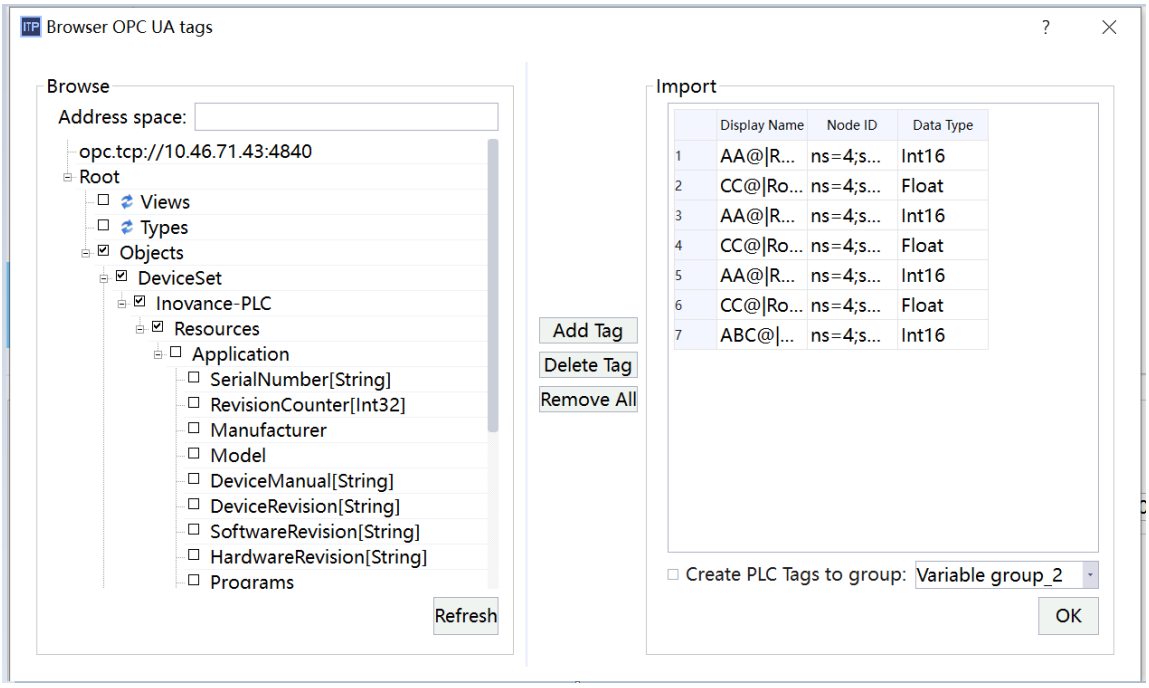
Protocol	Error Code and Description
FINS protocol	See the user manuals of Omron series PLCs.

The system communication error codes are as follows:

Errors	Error Code and Description
System communication error	0x10001: device failed to start
	0x10002: device write error
	0x10003: read error
	0x10004: read timeout
	0x10006: bus busy
	0x10007: serial port recognition error
	0x10008: offline
	0x10009: error in writing command once
	0x1000a: error in reading command once

4.3.2 OPC UA

OPC Unified Architecture is a new technology created by OPC Foundation, which is more secure, reliable and neutral (independent of suppliers), and transmits raw data and preprocessing information from the manufacturing field to the production planning or enterprise resource planning (ERP) system. The Inovance IT7000 HMI is OPC UA-enabled.



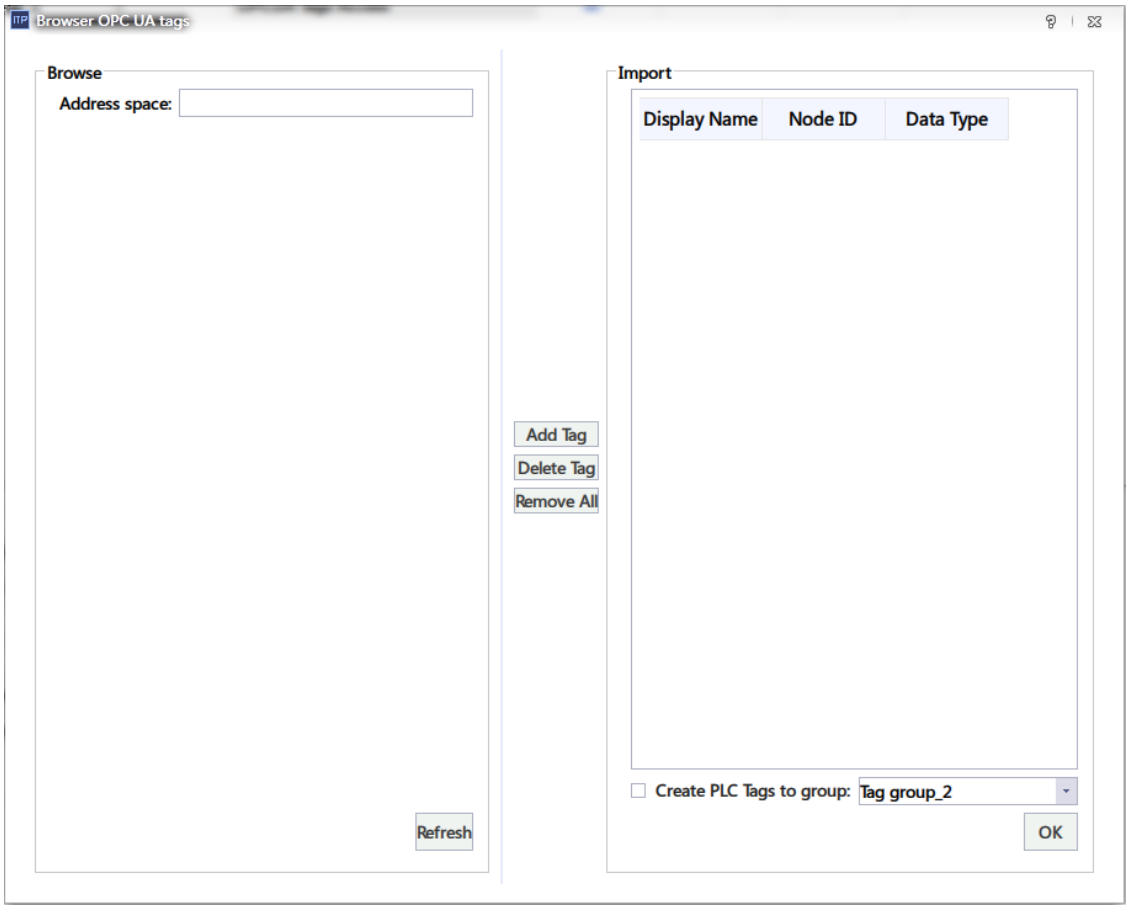
1. General settings of OPC UA protocol communication:

Name			Description	Remarks
General	Connection	IP Address	The address of the opc server.	
		Port Number	The port of the opc server.	Default: 4840
		Timeout Time	The timeout time for connection to the server, which defaults to 1000 ms	in ms
		Interval	Communication interval, which defaults to 100 ms	in ms
		Session Name	The session name is automatically generated and not modifiable. It is used by the server to identify the connection of a client.	Read only
		Security Policies	The security policies include None, Basic128Rsa15, Basic256, and Basic256Sha256	The default is None
		Data Privacy	Data privacy modes include None, Sign, and SignAndEncrypt	The default is None
	Label	Browser Tags	Click the button to pop up the Browse Tag window for browsing the tags on the server side	With the correct IP and port number, you can browse the online server and view and select tags

2. Authentication and read mode:

Name			Description	Remarks
Authenti-cation	Authentica-tion Mode	Anonymous	Check this to use anonymous authentication	
		Use Account	Check this to use account authentication	
		User name	User name	Only effective when "User Name/ Password Login" is checked
		Password	Password	
Read Mode	Read mode	Initiative	Reads tags actively	
		Monitor	Reads tags through monitoring	
		Monitor time	Monitor time, in ms	Only effective when Monitor is selected

3. The Browse Tag window is as shown in the following figure:



Operation	Description	Remarks
Add Tag	Adds the selected tags the left column to the import table	
Delete Tag	Deletes the selected items in the import table	
Delete All	Deletes all items in the import table	
Create PLC tags to groups	Check this and select a tag group to save tags to the tag group.	After adding tags to the tag group, you can perform configuration

5 Tag

5.1 Introduction

5.1.1 Number of Tags

In an HMI project created by InoTouchPad, you can create up to 64 tag groups, and the total number of tags saved by multiple tag groups does not exceed 65535 (excluding system tags).

5.1.2 External Tag

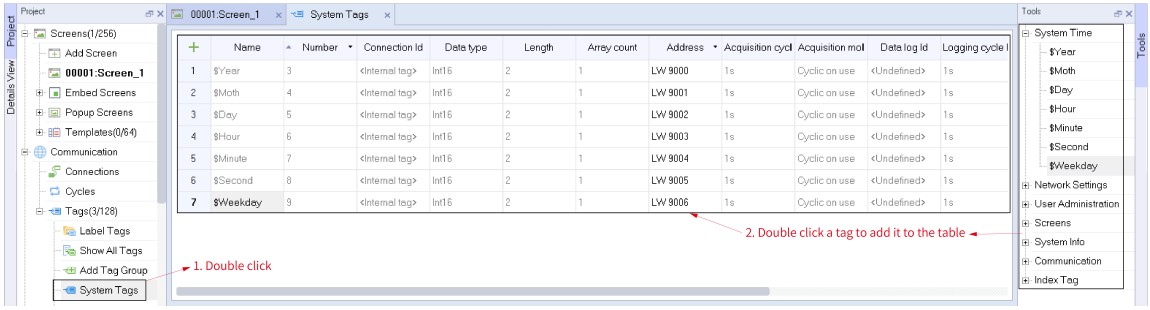
Configured connected tags, called external tags, are images of some defined storage locations in PLC. These areas can be accessed by HMI and PLC together (that is, read and write are allowed). The data types supported by these areas depend entirely on the PLC. For example, Inovance AC drives support int16 and unint16 data types, while Inovance H3Us support int16, UInt16, UInt32, int32, unint32, float, Bool and string.

5.1.3 Internal Tag

A tag without a configuration connection, called an internal tag, is an image of some defined storage locations in the HMI, and these areas can only be accessed by the HMI itself. when the HMI is a slave, the storage area in which the internal tags are located can also be accessed by other HMIs: For another HMI device with which it communicates, these internal tags act as external tags.

5.1.4 System Tag

Internal tags ranging from LW 9000 to LW 9982 are used to display some system information, and are called system tags. The addresses occupied by system tags are generally no longer used as ordinary internal tags, and the tags in this area are read-only. Note that when configuring internal tags, do not associate the address area of system tags. If you need to view specific system information, you can add corresponding system tags in the system tag group, as shown in the following figure:



1. Double click System Tags to open the System Tags tab.

2. The added system tags are displayed in the system tag table, and you can use the IO field to configure the system tags in the system tag table to obtain the corresponding system information.

3. In the toolbar on the right, double-click thesystem tag you want to add.

The system tags are described as follows:

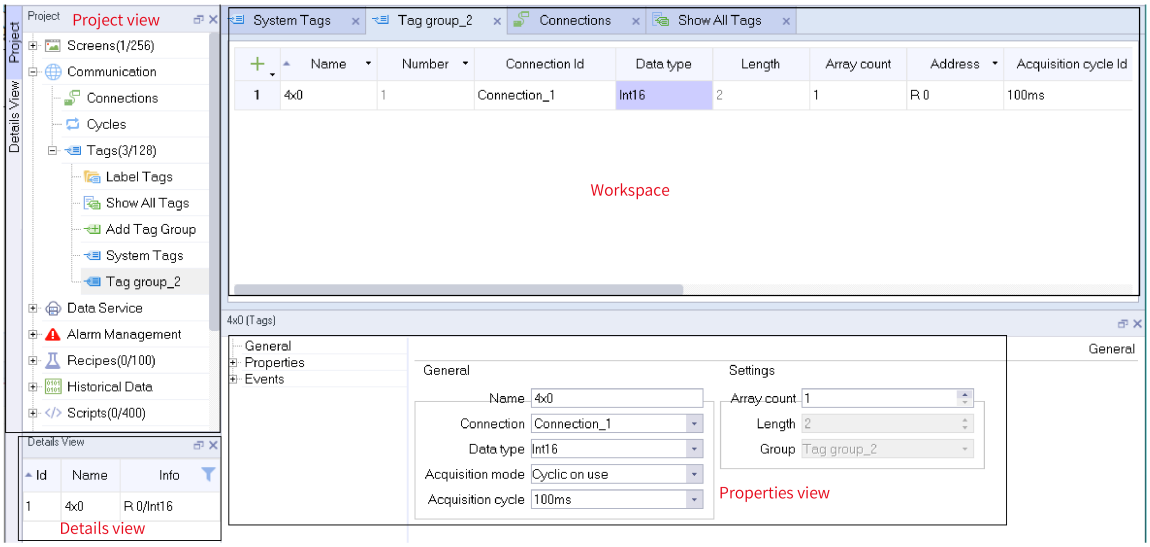
Category	Name	Description	Data type	Address
System time	$Year	System time: Year	Int16	LW 9000
	$Moth	System time: Month	Int16	LW 9001
	$Day	System time: Day	Int16	LW 9002
	$Hour	System time: Hour	Int16	LW 9003
	$Minute	System time: Minute	Int16	LW 9004
	$Second	System time: Second	Int16	LW 9005
	$Weekday	System time: Weekday	Int16	LW 9006
Network setting	$MACAddress	MAC address of target	String	LW 9007
	$IPAddress	IP address of target	String	LW 9019
	$Netmask	Subnet mask of target	String	LW 9031
	$Gateway	Gateway of target	String	LW 9043
	$DNS1	Primary DNS server address	String	LW 9055
	$DNS2	Secondary DNS server address	String	LW 9067
User Management	$CurrentUserName	Runtime user name	String	LW 9079
	$CurrentUserGroupId	Runtime user Group ID	Int32	LW 9099
	$CurrentUserAuthority	Runtime user permission	Int32	LW 9101
Screen	$CurrentScreenId	Runtime current screen ID	Int16	LW 9103
	$StartScreenId	Runtime start screen ID	Int16	LW 9104
System information	$HMIDeviceType	HMI Device type	Int16	LW 9105
	$HMISerialNumber	HMI serial number	String	LW 9106
	$HMIAutorunVersion	HMIAutorun version number	String	LW 9126
	$HMIRuntimeVersion	HMIRuntime version number	String	LW 9138
	$IOTSDKVersion	IoT SDK version number	String	LW 9150
	$ProjectVersion	Project version number	String	LW 9162
	$CPUUsage	CPU usage	Int16	LW 9174
	$MemoryUsage	Memory usage	Int16	LW 9175
	$Uptime	Device uptime	Int32	LW 9176
	$CurrentLanguage	Current language	String	LW 9178
	$ScreenSaverTime	Backlight timeout	Int16	LW 9190
	$CloseBacklightTime	Backlight timeout	Int16	LW 9191
	$Brightness	Backlight brightness	Int16	LW 9192
	$UsbUnmountCount	USB device unmount count	Int16	LW 9193
	$UsbMountCount	USB device mount count	Int16	LW 9194
	$SDCardUnmount-Count	SD card unmount count	Int16	LW 9195
	$SDCardMountCount	SD card mount count	Int16	LW 9196

Category	Name	Description	Data type	Address
Communication	$COM1Mode	COM1 mode	Int16	LW 9197
	$COM1BaudRate	COM1 Baud rate	Int32	LW 9198
	$COM1DataBits	COM1 data bits	Int16	LW 9200
	$COM1Parity	COM1 parity check	Int16	LW 9201
	$COM1StopBits	COM1 stop bits	Int16	LW 9202
	$COM1ErrorStationId	COM1 wrong station ID	Int16	LW 9203
	$COM1ErrorCode	COM1 error code	Int32	LW 9205
	$COM1ErrorCount	COM1 single connection error count	Int16	LW 9204
	$COM1ErrorTotal	COM1 total error count	Int32	LW 9207
	$COM2Mode	COM2 mode	Int16	LW 9209
	$COM2BaudRate	COM2 Baud rate	Int32	LW 9210
	$COM2DataBits	COM2 data bits	Int16	LW 9212
	$COM2Parity	COM2 parity check	Int16	LW 9213
	$COM2StopBits	COM2 stop bits	Int16	LW 9214
	$COM2ErrorStationId	COM2 wrong station ID	Int16	LW 9215
	$COM2ErrorCode	COM2 error code	Int32	LW 9217
	$COM2ErrorCount	COM2 single connection error count	Int16	LW 9216
	$COM2ErrorTotal	COM2 total error count	Int32	LW 9219
	$COM3Mode	COM3 mode	Int16	LW 9221
	$COM3BaudRate	COM3 Baud rate	Int32	LW 9222
	$COM3DataBits	COM3 data bits	Int16	LW 9224
	$COM3Parity	COM3 parity check	Int16	LW 9225
	$COM3StopBits	COM3 stop bits	Int16	LW 9226
	$COM3ErrorStationId	COM3 wrong station ID	Int16	LW 9227
	$COM3ErrorCode	COM3 error code	Int32	LW 9229
	$COM3ErrorCount	COM3 single connection error count	Int16	LW 9228
	$COM3ErrorTotal	COM3 total error count	Int32	LW 9231
	$TCPErrorCode	TCP error code	Int32	LW 9244
	$TCPErrorCount	TCP single connection error count	Int16	LW 9243
	$TCPErrorConnectionId	TCP wrong station ID	Int16	LW 9242
	$TCPErrorIP	TCP erroneous IP	String	LW 9233
	$TCPErrorPort	TCP wrong port number	Int16	LW 9241
	$TCPErrorTotal	TCP total error count	Int32	LW 9246
	$SelfAdaptionOvertime	Adaptation timeout	Int16	LW 9500
	$SelfAdaptionIntervalTime	Adaptation interval	Int16	LW 9501
IoT	$SIMStatus	SIM state	Int16	LW 9300
	$SIMSignal	SIM signal value	Int16	LW 9301
	$IOTLoginStatus	IoT login status	Int16	LW 9302
	$IOTLoginType	IoT login type	Int16	LW 9303
	$IOTLoginTime	IoT login time	String	LW 9304
	$GPS	GPS	String	LW 9314

Category	Name	Description	Data type	Address
Index tag	16-bit	If an ordinary tag is associated with an index tag, the address of the ordinary tag can be offset by modifying the value of the index tag. 128 16-bit index tags are supported	Int16	LW9600 address start
	32-bit	128 32-bit index tags are supported	Int32	LW9728 address start

5.2 Basic Setting

5.2.1 Configuring a Tag

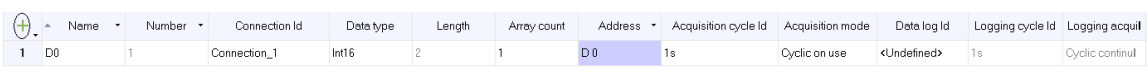


Steps:

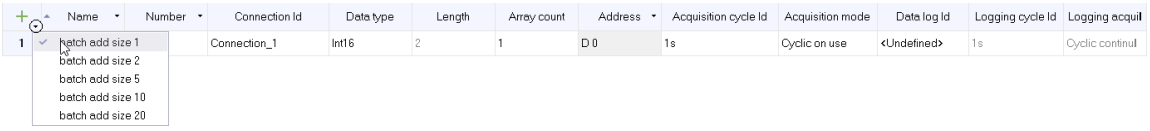
1. Expand Communication in the project view.
2. Expand the sub node Tags, and double click Tag Group_2 to open the tag editor.
3. Click the + button in the workspace of the tag editor to create a tag.
4. Then you can select the connection and data type and address of the driver of the PLC to be connected. If you want to use internal tags, just select <Internal tag> in the Connection ID field, and then select the data type and address.
5. Other properties of a tag can be set as needed.
6. Save the project to finish the tag configuration.

5.2.2 Tag Editor

In the tag editor, you can create tags. The tag editor is table as shown below. You can edit the properties of all created tags here.



Click the + button in the upper left corner to create a tag.



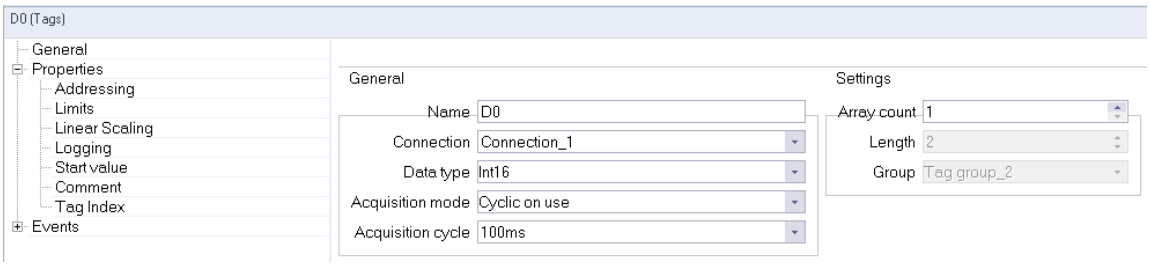
If you want to create multiple tags at one time, you also click the button in the upper right corner to select a batch operation.

Right-click the header to show hidden fields and set their values as needed.

Category	Name	Description	Description
General	Name	Variable	Cannot be duplicate and null; Limit: 128 characters
	Connection ID	Connection of the tag	Values: Internal tag, external tag Default: Internal tag
	Data type	Data type of the tag	Value: constrained by connection; Default: Int
	Acquisition mode	The acquisition mode of the specified tag	Values: on demand, cyclic continuous, cyclic on use On demand: The tag is updated by calling the system function UpdateTag; Cyclic continuous: The tag is updated all the time; Cyclic on use: The tag is updated only when the screen uses the tag;
	Acquisition cycle	How often the tag is acquired	The default is 1s
Settings	Tag group	The tag group that the tag belongs to	It is read only for the recipe tag group
	Array count	Array count	Only editable fo the recipe tag group Value range: 1 (default) to 2000.
	Length	Count of byte * array	It is read only
Address	Address	PLC Address	Not available for internal tags; Restricted by the protocol;
Log	Data Record	Data log of the tag	Defaulted to null
	Log acquisition mode	Log acquisition mode	Values: on change, on demand and cyclic continuous. Default: cyclic continuous On change: The tag is logged when its value changes; On demand: The tag is logged by executing the system function LogTag. Cyclic continuous: The tag is logged at fixed time intervals;
	Logging cycle	How often the tag is acquired	Effective only when 'Cyclic continuous' is selected; Default is null

Category	Name	Description	Description
Limit	Upper limit	Upper limit of the tag	Values: constant, tag Constant value range: data type range
	Upper limit alarm	Upper limit alarm	A Rising edge analog alarm saying 'The value is higher than the upper limit'
	Lower limit	Lower limit of the tag	Values: constant, tag Constant value range: data type range
	Lower limit alarm	Lower limit alarm	A falling edge analog alarm saying 'The value is lower than the lower limit'
Linear scaling	Linear scaling	Enables/disables linear scaling	Only effective for external tags. Ineffective for arrays. Values: on and off. The default value is off. You can also use InverseLinearScaling and LinearScaling
	Upper PLC scaling value	Upper PLC scaling value	Default: 10
	PLC linear scaling Lower limit	PLC lower scaling value	Default: 0
	HMI linear scaling Upper limit	Upper HMI scaling value	Default: 100
	HMI linear scaling Lower limit	HMI lower scaling value	Default: 0
Start value	Start value	The start/default value of the tag	Constant value range: data type range which is null by default
Comment	Comment	Comment of the tag	Limit: 500 characters
Index variable	Index variable	Associating an Index Tag	Default is null

5.2.3 Properties View of the Tag



The properties view of a tag consists of a tree view on the left and an editing area on the right. After you have edited the properties in the editing area, the modifications will be synchronized to the tag table editor. Similarly, modifications made in the tag table editor will also be synchronized to the Properties view.

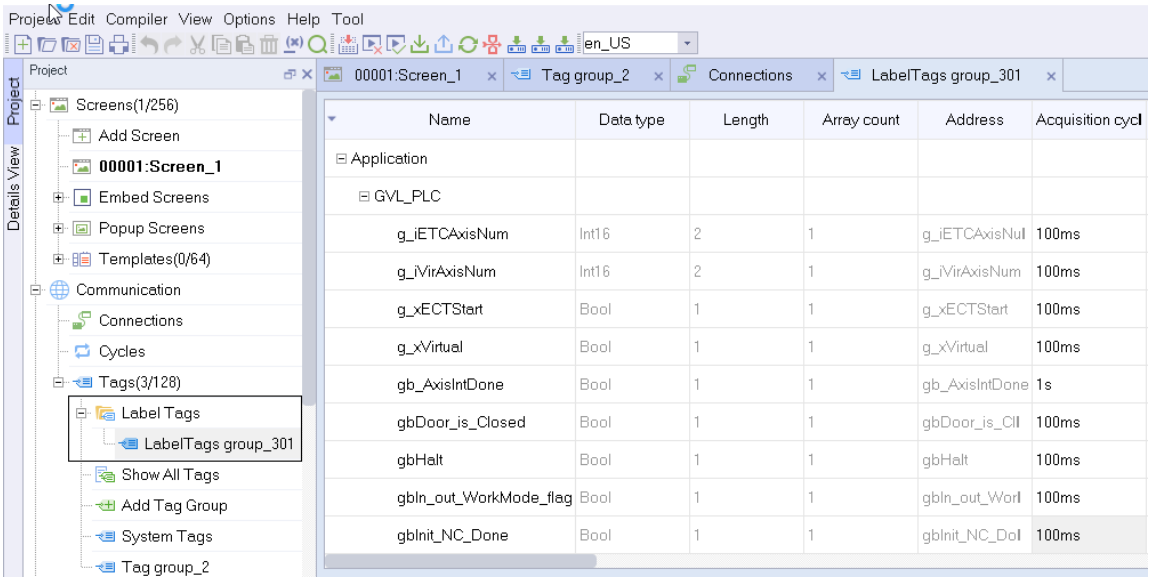
Property	Property	Description	Description
General	Name	Tag name	Cannot be duplicate and null;
			Limit: 128 characters
	Connection ID	Connection of the tag	Values: Internal tag, external tag
			Default: Internal tag
	Data type	Data type of the tag	Values: Constrained by connection;
			Default: Int16
	Acquisition mode	The acquisition mode of the specified tag	Values: on demand, cyclic continuous, cyclic on use
			On demand: The tag is updated by calling the system function UpdateTag;
			Cyclic continuous: The tag is updated all the time;
			Cyclic on use: The tag is updated only when the screen uses the tag;
	Acquisition cycle	How often the tag is acquired	The default is 1s
Settings	Tag group	The tag group that the tag belongs to	It is read only
	Array count	Array count	Value range: 1 (default) to 2000.
	Length	Count of byte * array	It is read only
Address	Address	Address	In the case of an Internal tag, you can edit the internal address of the HMI;
			In the case of an external tag, the address is restricted by the protocol;
Log	Data log	Data log of the tag	Defaulted to null
	Log acquisition mode	Log acquisition mode	Values: on change, on demand and cyclic continuous. Default: cyclic continuous
			On change: The tag is logged when its value changes;
			On demand: The tag is logged by executing the system function LogTag.
			Cyclic continuous: The tag is logged at fixed time intervals;
	Logging cycle	How often the tag is acquired	Effective only when 'Cyclic continuous' is selected;
			Default is null
Limits	Upper limit	Upper limit of the tag	Values: constant, tag
			Constant value range: data type range
	Upper limit alarm	Upper limit alarm	A Rising edge analog alarm saying 'The value is higher than the upper limit'
	Lower limit	Lower limit of the tag	Values: constant, tag
			Constant value range: data type range
	Lower limit alarm	Lower limit alarm	A falling edge analog alarm saying 'The value is lower than the lower limit'

Property	Property	Description	Description
Linear scaling	Linear scaling	Enables/disables linear scaling	Only effective for external tags. Ineffective for arrays.
			Values: on and off. The default value is off.
			You can also use InverseLinearScaling and LinearScaling
	Upper PLC scaling value	Upper PLC scaling value	Default: 10
	PLC linear scaling	PLC lower scaling value	Default: 0
	Lower limit		
	HMI linear scaling	Upper HMI scaling value	Default: 100
	Upper limit		
	HMI linear scaling	HMI lower scaling value	Default: 0
	Lower limit		
Start value	Start value	The start/default value of the tag	Constant value range: data type range which is null by default
Comment	Comment	Comment of the tag	Limit: 500 characters
Index variable	Index variable	The associated Index tag	It is not associated by default, and needs to be configured in system tags

In InoTouchPad, you can configure specific properties for each tag that determine the usage of the tag.

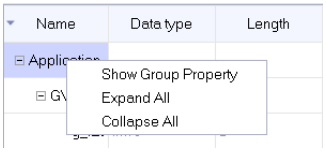
5.2.4 Label Tag

Label communication, as a special communication protocol, can import label tags and display them in a tree view. As shown in the following figure, after you create a label communication, a corresponding label tag group will be added under Label Tags. For property setting of the tag, see the contents of the above two sections. Note that some general properties cannot be modified due to restrictions.



Tag tree view

You can collapse and expand the tree with one click. For example, you can right-click ⊟ Application as shown above and select Expand All and Collapse All.



Alarm codes

If there is something wrong with the label tag, an alarm message will show. The alarm codes are shown in the following table:

Alarm Code	Description
0x10000100	Failed to initialize client communication
0x10000101	Communication not initialized
0x10000102	Number of connections exceeds the limit
0x10000103	Communication connection failed, which may be due to network disconnection
0x10000104	Communication is disconnected and needs to be reconnected
0x10000105	Gateway not connected or disconnected and needs to be reconnected
0x10000200	Timed out, communication may be disconnected
0x10000302	PLC manufacturer does not match
0x10000501	Registered tags do not match symbols in PLC
0x10000502	Invalid communication client, please reconnect
0x10000503	Invalid tag definition, please redefine the tag
0x10000504	Cannot write referenced type tag
0x10000505	Tag has no access
0x10000506	Too many symbol tags
0x10000507	The definition of symbol tag has changed and needs to be redefined
0x10000508	Too many values are written
0x10000400	Symbol profile is not XML
0x10000401	Symbol profile does not have symbol information start identification
0x10000402	Symbol profile does not have a header
0x10000403	Invalid label handle
0x10000410	Incorrect symbol configuration data type
0x12000000	Symbol profile line error. The last 24 bits indicate the line number

5.3 Usage

5.3.1 Using Tags in Communication

External tags, which are storage location images defined in the PLC, are used for data exchange between the HMI and PLC. In InoTouchPad, to create a tag connected to the PLC, you must specify the

connection, the tag data type, and the address so that the HMI and the PLC can access the same location.

An internal tag can also be used for communication when the HMI is used as a slave, provided that data are exchanged between HMIs. Internal tags are storage location images defined in the HMI. In InoTouchPad, if you want to create a tag connected to the HMI, you must create a slave protocol in Connections, and then set the HMI connected to it as the master. Use the standard Modbus protocol and specify the data type and address, then the master HMI can access the address area of the internal tag of the slave HMI. Since the slave device being read and written is an HMI, the correspondence is as follows:

Reading/Writing 0x/1x(0–12000)	Corresponds to reading/writing LB(0–11999)
Reading/Writing 3x/4x/5x(0–10000)	Corresponds to reading/writing LW(0–8999)
Reading/Writing 3x/4x/5x(10000–65535)	Corresponds to reading/writing RW(0–55535)

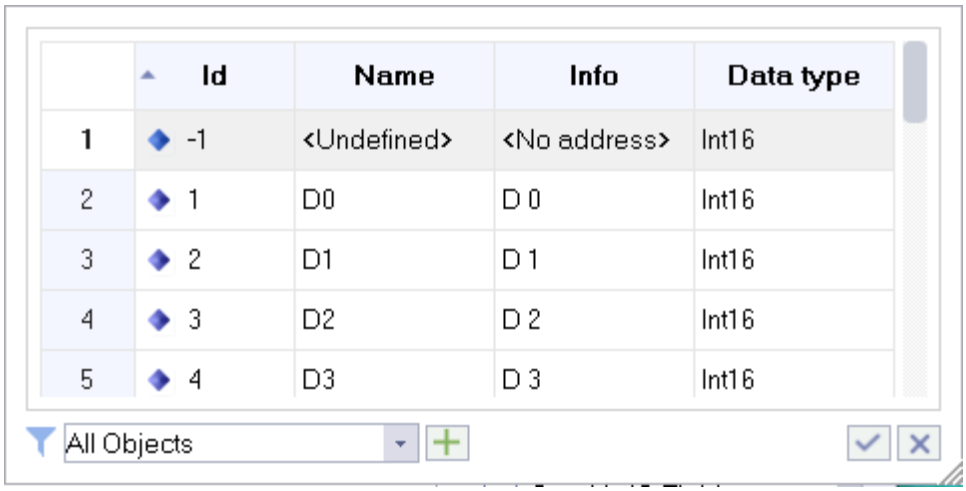
You can set an acquisition period to make the HMI read the value of an external tag in the specified period. As long as the tag is displayed or recorded in the screen, the value will be updated regularly. The time interval of regular update is determined by the acquisition period, which can be the default period or customized.

5.3.2 Changing Tag Configuration

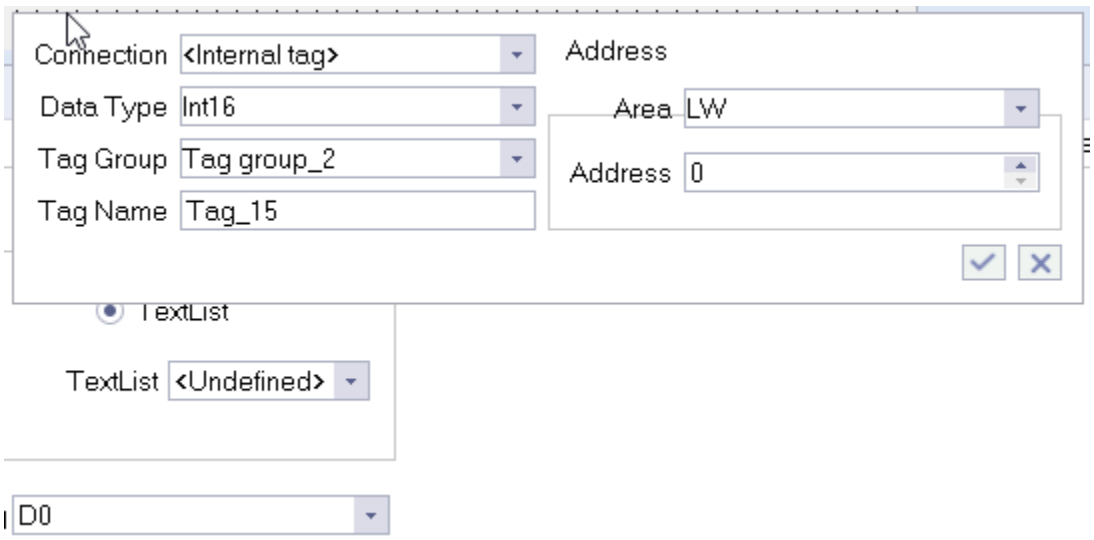
Users can change the settings of tags as needed.

	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycle Id	Acquisition mode	Data log Id
1	D0	1	Connection_1	Int16	2	1	D 0	1s	Cyclic on use	<Undefined>
2	D1	2	Connection_1	Int16	2	1	D 1	1s	Cyclic on use	<Undefined>
3	D2	3	Connection_1	Int16	2	1	D 2	1s	Cyclic on use	<Undefined>
4	D3	4	Connection_1	Int16	2	1	D 3	1s	Cyclic on use	<Undefined>
5	D4	5	Connection_1	Int16	2	1	D 4	1s	Cyclic on use	<Undefined>
6	D5	6	Connection_1	Int16	2	1	D 5	1s	Cyclic on use	<Undefined>
7	D6	7	Connection_1	Int16	2	1	D 6	1s	Cyclic on use	<Undefined>
8	D7	8	Connection_1	Int16	2	1	D 7	1s	Cyclic on use	<Undefined>
9	D8	9	Connection_1	Int16	2	1	D 8	1s	Cyclic on use	<Undefined>
10	D9	10	Connection_1	Int16	2	1	D 9	1s	Cyclic on use	<Undefined>
11	D10	11	Connection_1	Int16	2	1	D 10	1s	Cyclic on use	<Undefined>
12	D11	12	Connection_1	Int16	2	1	D 11	1s	Cyclic on use	<Undefined>

When you need to edit multiple tags, you can use the tag editor. In the tag editor, you can compare and adjust the properties of multiple tags, or sort by tag properties.



To create a tag directly at the point of use, click the button in the object list to open the tag properties edit window, as shown in the following figure.



Then edit and configure directly in the window.

	Data log Id	Logging...	Logging ac...	Upper limit	Lower limit	Linear scali...	Upper PLC sc...
1	<Undefined>	1s	Cyclic conti...	1	100	Off	10
2	<Undefined>	1s	Cyclic conti...	<No limit>	<No limit>	Off	10
3	<Undefined>	1s	Cyclic conti...	<No limit>	<No limit>	Off	10
4	<Undefined>	1s	Cyclic conti...	<No limit>	<No limit>	Off	10
5	<Undefined>	1s	Cyclic conti...	<No limit>	<No limit>	Off	10
6	<Undefined>	1s	Cyclic conti...	<No limit>	<No limit>	Off	10

During configuration, illegal items will be highlighted in red. For example, it is illegal to set the upper limit value less than the lower limit value.

5.3.3 Limit Value

The tag limit value is only for tags of numeric type (except bool type). If the process value is above or below the limit range, this can be configured to trigger analog alarm messages or events. When an operator enters a value that exceeds the limit, the value is rejected and is not saved.

5.3.4 Initial Value of a Tag

The initial value of a tag is the value that will be preset when the running system starts. An initial value can be preset for all data types. Setting the initial value can ensure that the tag is in a user-defined state when the project starts.

5.3.5 Updating Tag Values at Run Time

At run time, tag values can usually be changed in the following ways:

1. The operator change the tag value directly in IO field.
2. Using system functions, such as "SetValue".

3. Assigning a value to the tag in the script, for example: SmartTags ("D 0") = 1;
4. Through communication. (When updating external tags, the updating mode depends on the acquisition mode. If you use the on demand mode, a command triggers a tag update; If you use the recycling mode, as long as the tag is displayed in the current screen, it will be updated after an acquisition period, which can be customized; If you use cyclic continuous mode, the tag will be updated once after an acquisition period, no matter whether the tag is displayed in the current screen. Because frequent reading operations will increase communication load, recycling mode is preferred. with this mode, only when the tag is displayed and used on the current screen, it will be updated periodically, and cyclic continuous mode is generally only used for some tags that must be updated continuously.)

5.3.6 Data Record

Configuration data records are tag values stored for data statistics, analysis and calculation. There are three data recording methods. With the value change mode, the tag value will be recorded once every time it changes; with the on demand mode, the tag value is recorded every time the command (LogTag) is triggered; with cyclic continuous mode, the value is recorded based on the recording period, which is customizable and determines the recording frequency.

The data records that can be configured by tags are created in Historical Data, see section Historical Data for details.

5.3.7 Linear Scaling of Tags

Linear scaling applies to numeric data. Linear scaling maps the values of external tags to a value range in an InoTouchPad project.

To perform linear scaling, you must first enable linear scaling of the tag in the tag editor, and then set the upper and lower limits of HMI linear scaling and PLC linear scaling. After that, the value ranges will be linearly mapped with each other.

	Name	Number	Connection Id	Address	Linear scaling	Upper PLC scaling ...	Lower PLC scaling...	Upper HMI scalin...	Lower HMI scalin...
1	Var_1	1	Connection_1	D 0	On	10	0	100	0

For example, create an external tag "Tag_1", set its connection to H3U monitor protocol, the address to D 0, enable linear scaling, set the upper and lower limits of PLC to 10 and 0 respectively, and set the upper and lower limits of HMI to 100 and 0 respectively. When the HMI communicates with H3U, if you set the value of Tag_1 to 50 on the HMI, the value will be converted according to the linear relationship set before, and then written to the D 0 register in H3U (that is, the value written to the D 0 register is actually 5). Similarly, if the value read from the D 0 register is 8, the displayed value of Tag_1 on the HMI after linear scaling is 80.

Through the above procedure, the range [0–10] on the PLC can be mapped to the range [0–100] on the HMI.

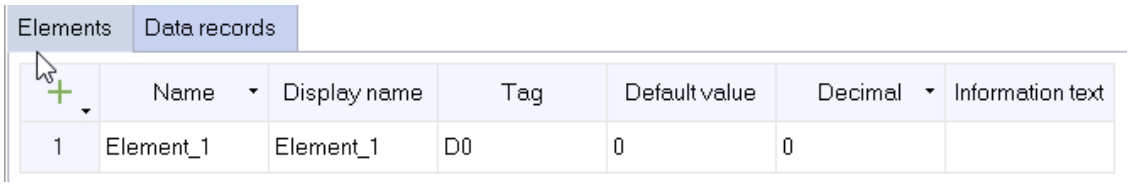
In addition, you can also use system functions LinearScaling() and InverseLinearScaling() to achieve linear scaling of tags, as detailed in *"12.3.2 Calculation" on page 251*.

5.3.8 Array Tag

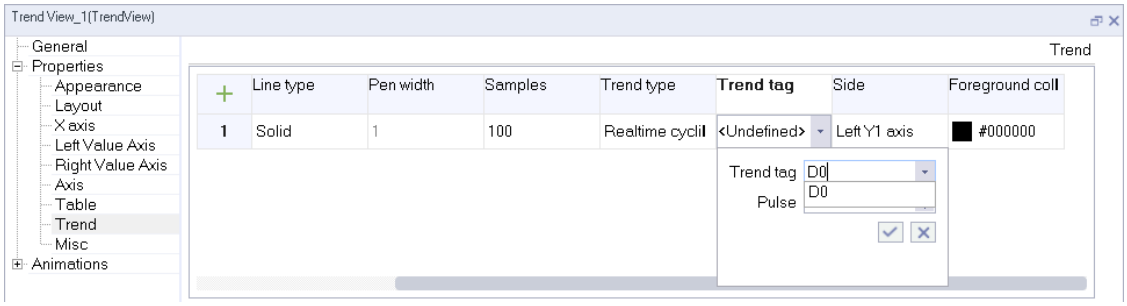
An array tag is a tag whose element count is greater than 1. It takes the specified address as its first address, and the address space is continuous. Each array element has the same data type, that is, the data type specified by the tag.

Multiple array elements with the same attributes can be addressed by a single array tag name, and then, in configuration, each array element can be used individually. Array tags are not available everywhere in an InoTouchPad project:

1. Ingredient tags can be used in the recipe editor.

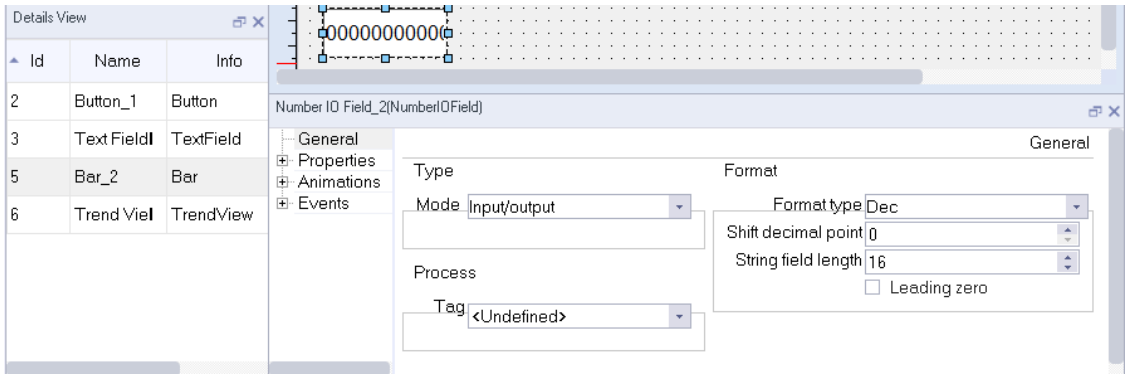


2. In trend view, trend buffer tags can be used.



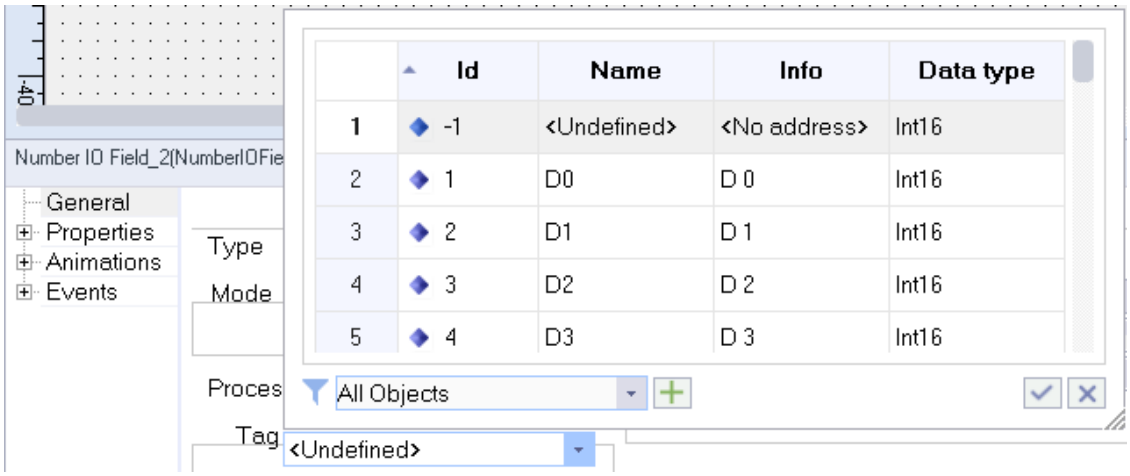
5.3.9 Using Tags

1. Dragging tag group to the screen: Click the tag group to display the detailed view and drag the selected tag to the screen. A control is automatically generated.
2. Dragging a tag to the tag selection box:
   Click on the control to be configured, take IO domain as an example.



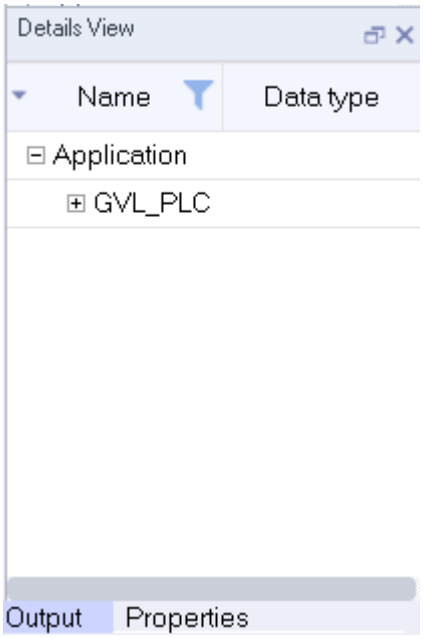
   Drag the tag to the tag selection box to finish configuration.

3. This is the most common way.



4. Configuring by inputting name in the control's tag selection box.
   Note: Some special tags may be configured differently. For example, the detailed view of tag communication is a tree structure, as shown in the following figure:

   Label communication allows drag and drop of tags, but does not support name input and auto completion.

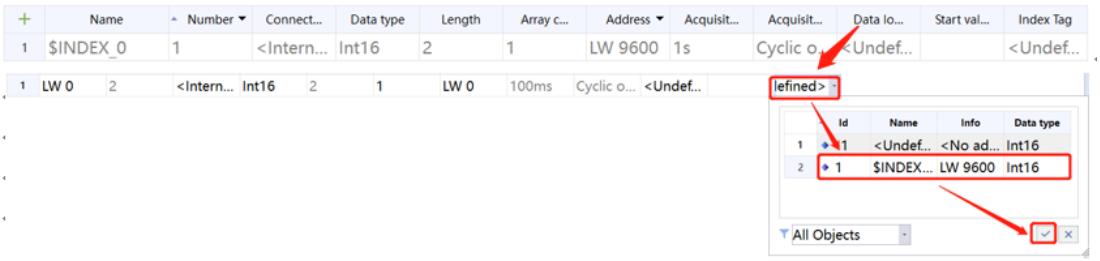


5.3.10 Associating an Index Tag

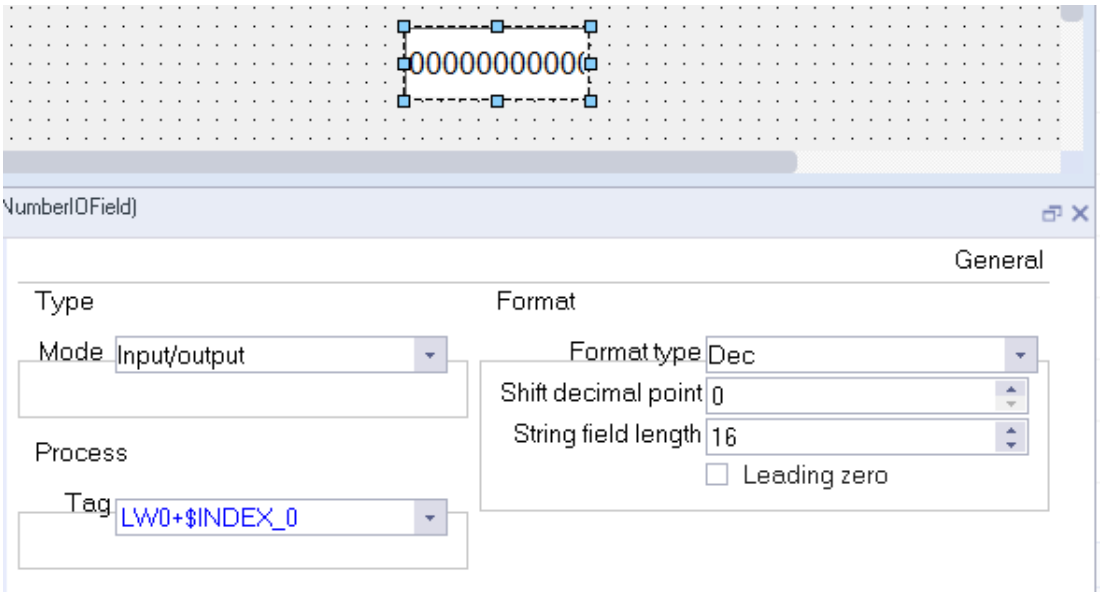
There are 128 16-bit and 32-bit index tags in the system tags of IT7000. You can access multiple address spaces with one control by associating an index tag.
The index tag is used as follows:

1. Go to Communication->Tags->System tags to create a 16-bit $INDEX_0.
2. There are two ways to associate the index tag:

   a. Select the index tag in the tag editor, as shown in the following figure.



b. Drag a new control to manually enter the association in the format of "tag name+$INDEX_index tag number". For example, LW0+$INDEX_0. As shown in the following figure, press Enter after the association text.



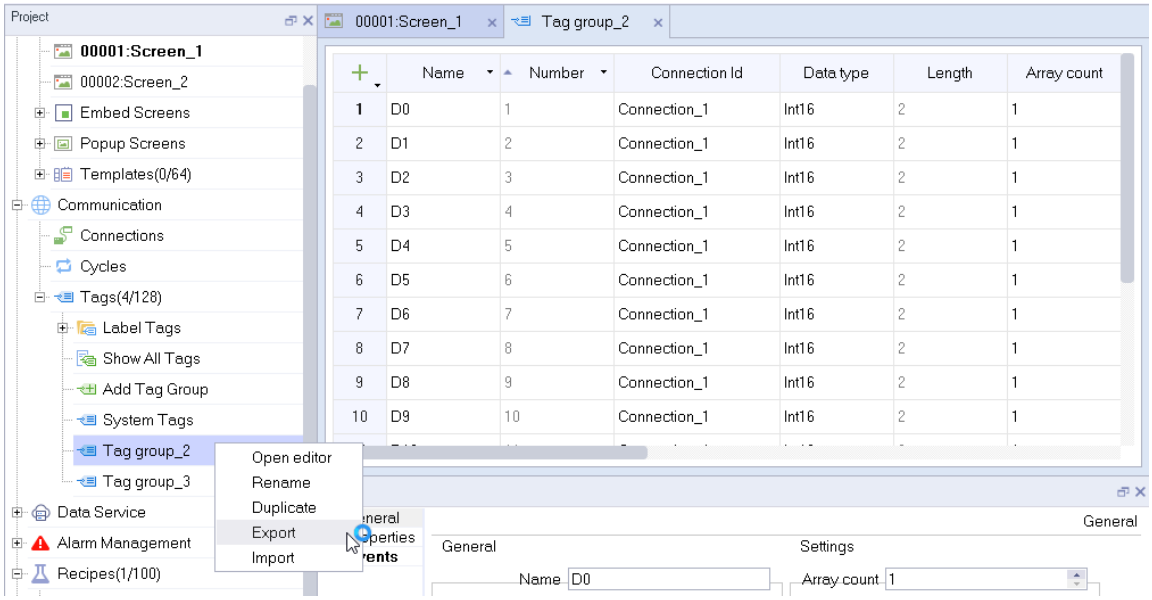
5.4 Importing and Exporting Tags

5.4.1 Overview

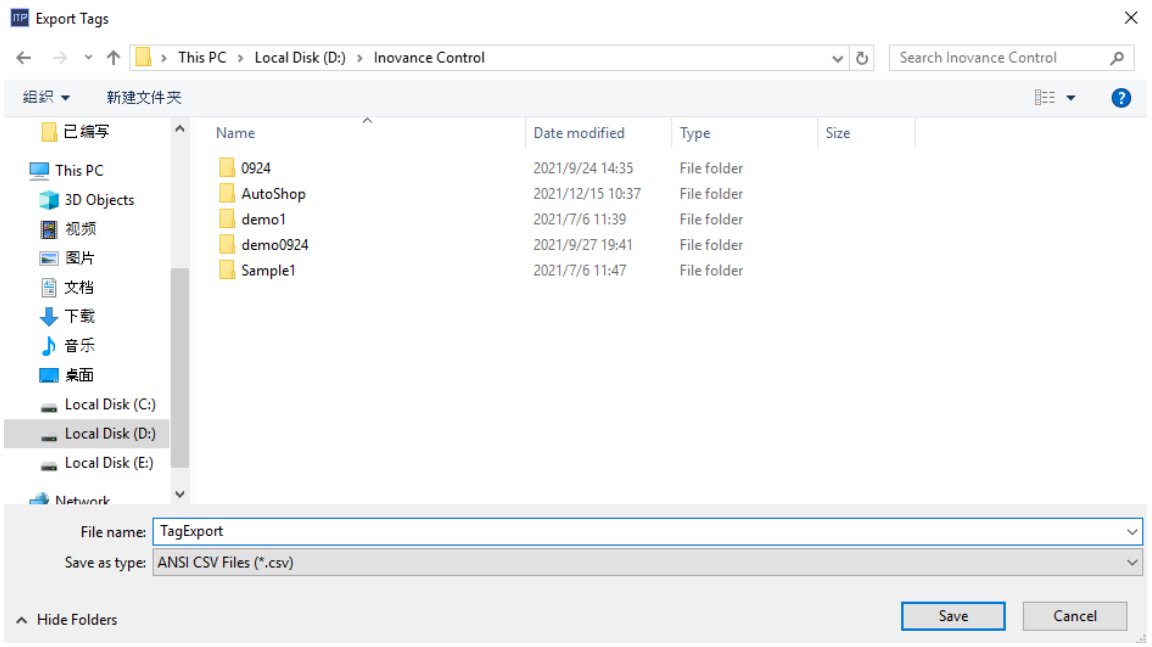
InoTouchPad allows exporting variables from one project and then importing them into another project, which can greatly reduce the workload of creating new projects.

5.4.2 Exporting Tags

Right click any tag group in the tree view on the left side of the project (system tags cannot be imported or exported), such as "Tag group 2", and click Export in the menu.

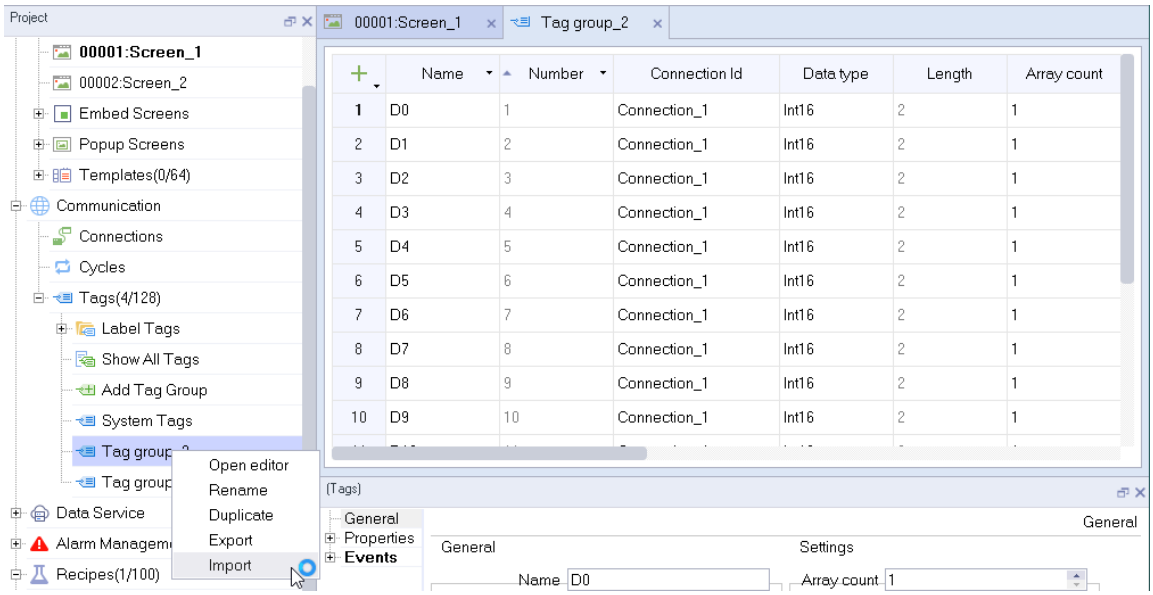


Then, specify the path and file name of the exported CSV file in the pop-up window and click Save. All tags in tag group 2 will be exported to the *. CSV file as specified. You can view and edit the exported CSV file by Excel.



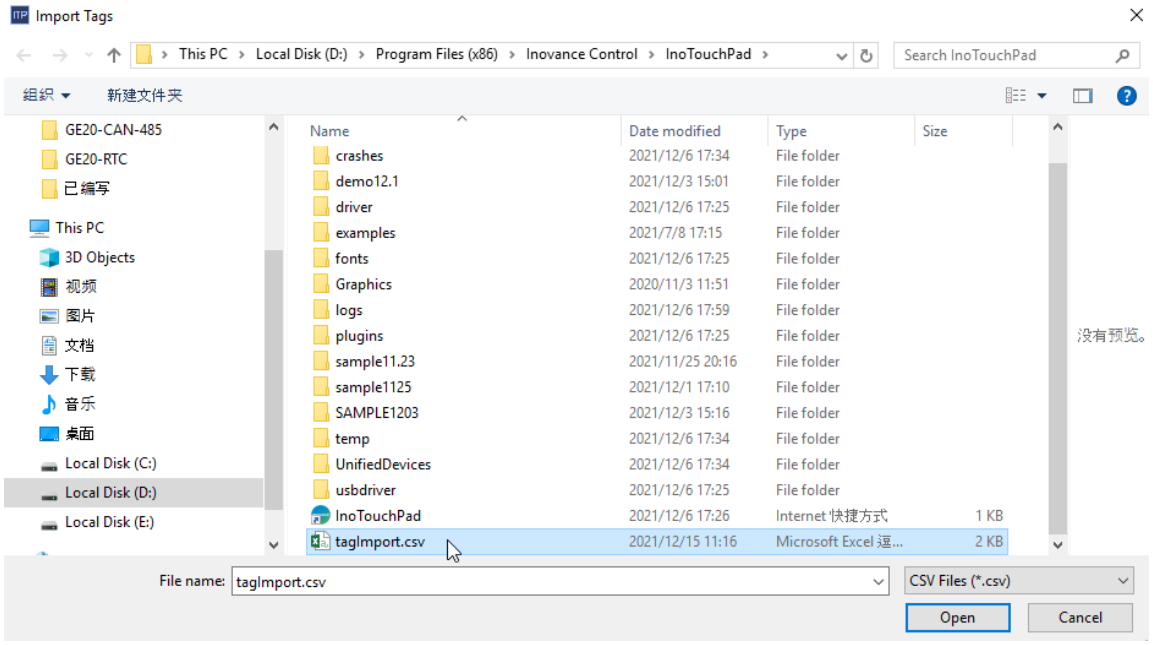
5.4.3 Importing Tags

Right click any tag group in the tree view on the left side of the project (system tags cannot be imported or exported), such as "Variable group 2", and click Import in the menu.



Then, a prompt will pop up, warning you that if a tag name has already existed in the project, it will be overwritten.

Click Yes to find the *. csv file to be imported through the pop-up dialog.



5.4.4 Tag Data File Format

The data file used for tag import and export must be a comma-separated CSV file.

During export, the tag information is written to a *. csv file. The first 18 rows of the exported text are explanatory information and headers of the tag list, and the remaining rows are the exported data, which are in 13 columns: name, connection ID, data type, length, array count, address, acquisition cycle ID, acquisition mode, maximum, minimum, start value, comment and group Id. The fields are separated by ",". When a field is not empty, the contents are enclosed by double quotation marks. If the contents contain double quotation marks, the " will be escaped by another ", that is ""..."". (Note:

The "," used to separate fields and the double quotation marks enclosing the contents of fields are English characters).

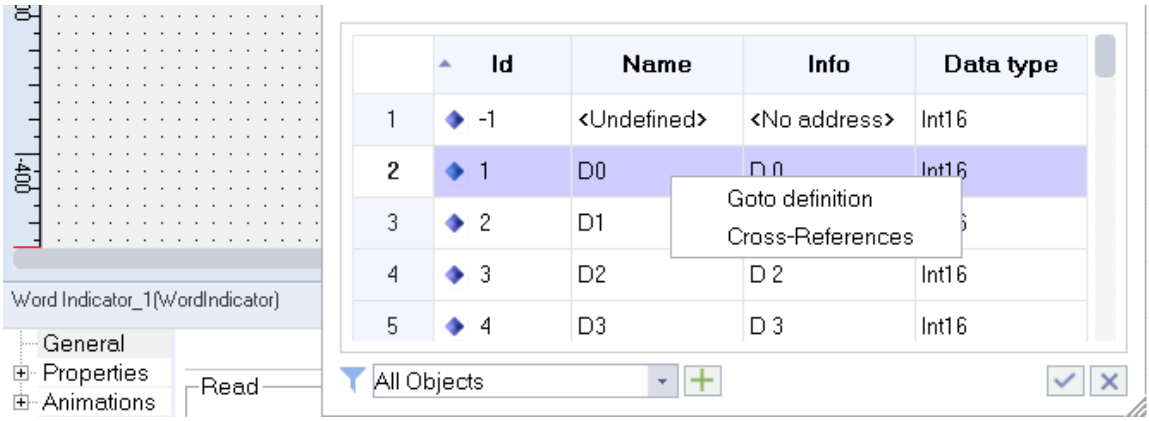
During import, the first 18 lines of the import text are in a fixed format, and can not be added, deleted or changed. Only variable data can be edited. In the import file, each variable occupies an independent line, and each variable contains the following 13 fields which cannot be added or deleted:

Item	Function	Description
Name	The tag name	If the name is empty, the line of data will not be imported. If the name exists in the project database, the imported data will overwrite the data in the database, otherwise, new data is added.
Connection	Connection	It can be empty or the connection created in the current project (except for the connection with slave protocol including MODBUS Slave and MODBUS TCP/IP Slave). If the connection does not exist in the current project, the data will not be imported
Data type	Data type of the tag	1. It cannot be null. 2. When the connection is an internal tag, any data type that is not supported by an internal tag cannot be imported. 3. For an existing connection, any data type that is not supported by the connection protocol cannot be imported. (note: Strings are case sensitive and strictly match the type in the variable editor. Invalid strings will be filtered).
Length	The length of the tag, which is equal to array count * data type length,	1. It cannot be null. 2. When it is of any type other than Wstring or String: The length must be consistent with the calculated value. 3. When the data type is Wstring or String: The value must be within the specified range. Note: length of Bool type is in bit and that of other types is in byte: UInt16 (length is 2), Int16 (length is 2), UInt32 (length is 2), Int32 (length is 4), Float (length is 4), Double (length is 8), Bool (length is 1), String (length is 12–255), WString (length is 4–512), DataTime (length is 8), UIntUInt64 (length is 8), and Int64 (length is 8).
Array count	the number of elements in the array	1. It cannot be null. 2. When the data type is Wstring or String: The value must be 1. 3. When the data type is of any type other than Wstring or String: The value must be an integer within 1–2000.
Address	The address of the specified tag	1. It cannot be null. 2. The address is invalid and is not imported if not as specified by the protocol or the internal tag. 3. The address value should be formatted correctly, such as D 0 instead of D0, where the space cannot be omitted.
Acquisition cycle	The acquisition cycle of the specified tag	1. It cannot be null. 2. Any period that is not 1h, 1min, 10s, 5s, 2s, 1s, 500ms, or 100ms, or that is not added in InoTouchPad is invalid.
Acquisition mode	The acquisition mode of the specified tag	1. It cannot be null. 2. The value must be 0, 1, or 2. (0-on demand, 1-cyclic on use, and 2-cyclic continuous)

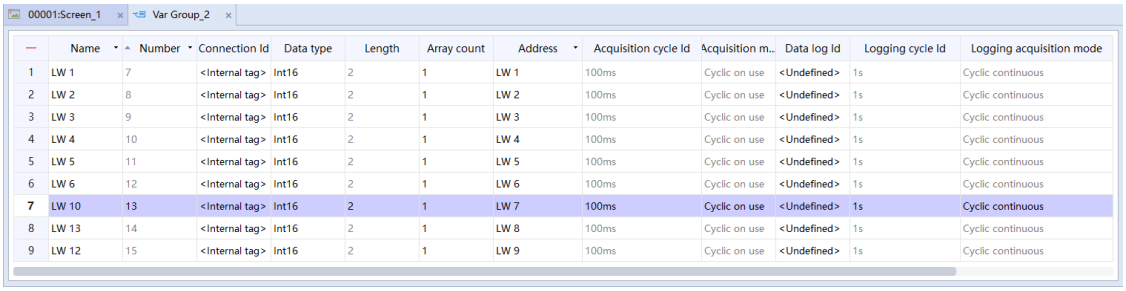
Item	Function	Description
Upper limit	Restricts objects to numeric type	1. When the import data is of UInt16, Int16, UInt32, Int32, Float, or Double type, any value beyond the limit is invalid (it can be void, meaning that there is no limit). 2. For Bool, WString, String and DataTime, the value must be null. 3. When the data is a numeric value, the upper limit must be greater than the lower limit. 4. Non-numeric strings, including valid tag names, are invalid. (The value is not exported if it is a tag).
Lower limit	Restricts objects to numeric type	1. When the import data is of UInt16, Int16, UInt32, Int32, Float, or Double type, any value beyond the limit is invalid (it can be void, meaning that there is no limit). 2. For Bool, WString, String and DataTime, the value must be null. 3. When the data is a numeric value, the upper limit must be greater than the lower limit. 4. Non-numeric strings, including valid tag names, are invalid. (The value is not exported if it is a tag).
Start value	The start value of the tag	1. When the import data is of UInt16, Int16, UInt32, Int32, Float, or Double type, any value beyond the limit is invalid (it can be void, meaning that there is no limit). 2. For WString and String types, the value is invalid if the string size is greater than the tag length. 3. For the DataTime type, the time value must be correct and in the format of "yyyy-MM-dd hh:mm:ss" or "yyyy/MM/dd hh:mm:ss" (note: the hour, minute and second are not necessary, but the year, month and day must be provided)
Comment	The comment of the tag	No restrictions
Group ID	The tag group to which the tag belongs	Only valid for importing all tags
Index Tag	The name of the associated index tag	It can be null and <undefined>;

5.4.5 Tag Reference Lookup

In the selection box of project configuration, you can find the control and screen that uses the tag according to the tag. Right-click the corresponding control and select Back to definition or Cross-reference



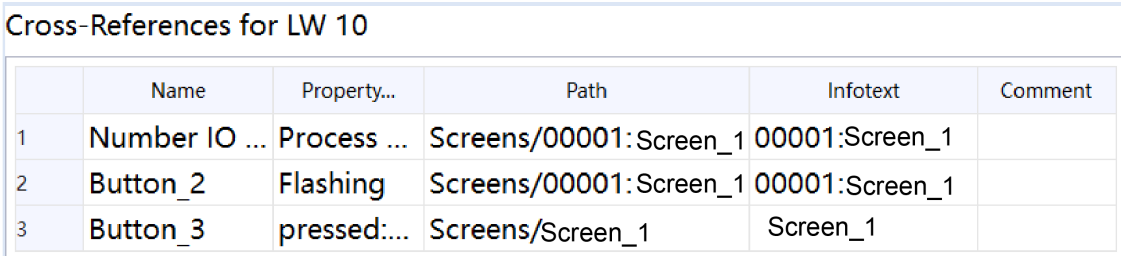
1. Back to definition: You will go back to the tag group where the tag is located, as shown in the following figure.



00001:Screen_1 | Var Group_2

	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycle Id	Acquisition m..	Data log Id	Logging cycle Id	Logging acquisition mode
1	LW 1	7	<Internal tag>	Int16	2	1	LW 1	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous
2	LW 2	8	<Internal tag>	Int16	2	1	LW 2	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous
3	LW 3	9	<Internal tag>	Int16	2	1	LW 3	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous
4	LW 4	10	<Internal tag>	Int16	2	1	LW 4	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous
5	LW 5	11	<Internal tag>	Int16	2	1	LW 5	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous
6	LW 6	12	<Internal tag>	Int16	2	1	LW 6	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous
7	LW 10	13	<Internal tag>	Int16	2	1	LW 7	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous
8	LW 13	14	<Internal tag>	Int16	2	1	LW 8	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous
9	LW 12	15	<Internal tag>	Int16	2	1	LW 9	100ms	Cyclic on use	<Undefined>	1s	Cyclic continuous

2. Cross-reference: The Cross-reference page will show, listing all controls or system functions that refer to the tag.



Cross-References for LW 10

	Name	Property...	Path	Infotext	Comment
1	Number IO ...	Process ...	Screens/00001:Screen_1	00001:Screen_1	
2	Button_2	Flashing	Screens/00001:Screen_1	00001:Screen_1	
3	Button_3	pressed:...	Screens/Screen_1	Screen_1	

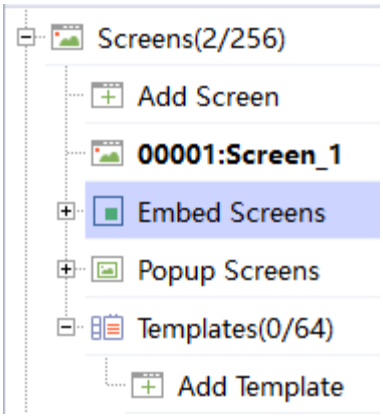
6 Creating a Popup Screen

6.1 Introduction

6.1.1 Overview

As the most basic element of configuration, the screen is used to present the contents of configuration. By creating pages, you can easily control and monitor the process and data of machinery and equipment. Screens include ordinary screens, embed screens, pop-up screens and template screens.

- Ordinary screen: a screen displayed on the display
- Pop-up screen: a screen pops up on the ordinary screen
- Template screen: a screen that can be referenced by the ordinary screen



An ordinary screen is added by double-clicking Add Screen

An embed screen is added by double-clicking Add Embed Screen

A pop-up screen is added by double-clicking Add Popup Screen

A template screen is added by double-clicking Add Template

A screen can contain static and dynamic elements:

- Static elements: They do not change their state at run time. Such as text or graphic objects.
- Dynamic elements: They change their state as needed. For example:
  - Display the output read from PLC memory
  - Display the process values output from the memory of the HMI in the form of letters and numbers, trend and bar charts

Caution

The input field on the HMI device can also be used as a dynamic element. You can switch the process value and operator input value between the controller and the HMI device through variables.

6.1.2 Screen Association

The type of the HMI device determines how the project is displayed in InoTouchPad and the functions of the editor.

When creating a project, you must select the corresponding HMI device type for the project. The " *“14.4 Project Settings” on page 316*" in the project view can be used to change the HMI device type.

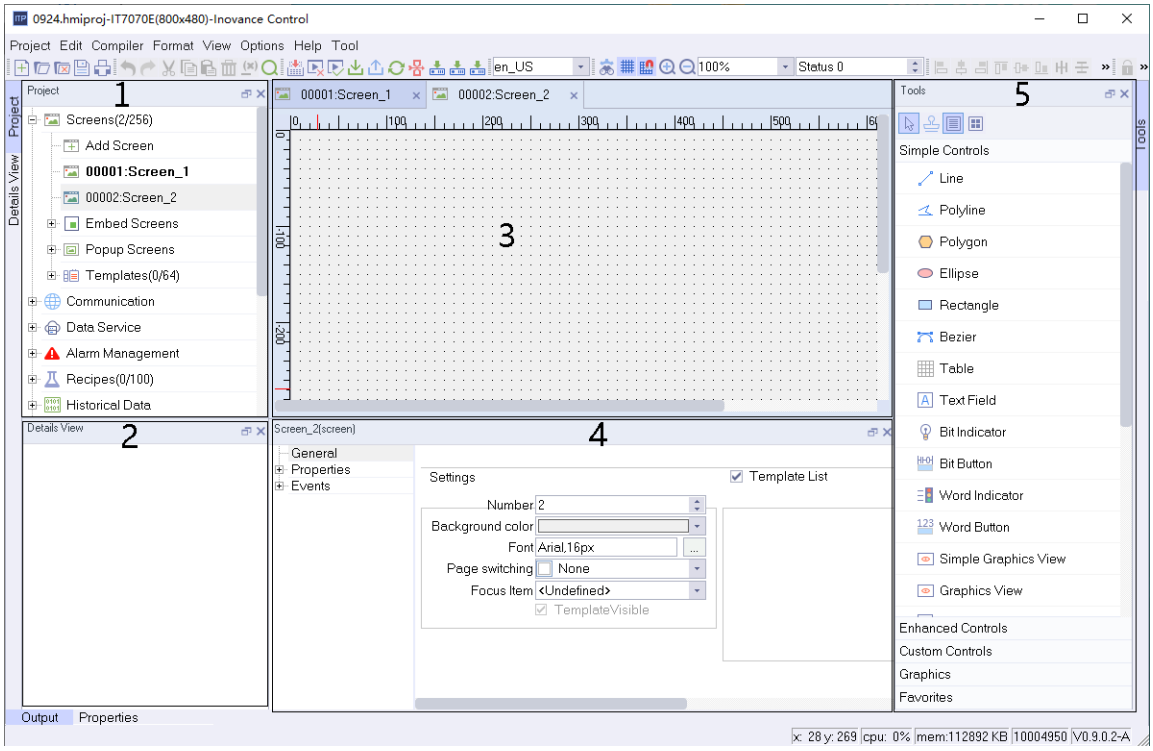
The following screen properties are determined by the type of HMI selected:

- Resolution
- Available objects

6.1.3 Screen Editor

When you create or open a project, the screen editor also opens You can also open the screen editor by double-clicking a screen or double-clicking Add Screen under Screens of the project view. The screen editor is as follows:

1-project view; 2-detail view; 3-workspace; 4-Properties view; 5-Tool window



Project view

Displays the functional modules that can be used in the current project. You can open a module for editing by double-clicking an item.

Workspace

You can configure the screen in the workspace.

Tool window

The tool window contains simple and complex controls that you can add to the screen, such as the string IO field control and the alarm view control. You can use the Graphics and Favorites to add items to the screen.

Properties view

The content in the properties view depends on the object currently selected in the workspace.

- The properties of the selected object can be browsed and edited in the Properties dialog box.
- If you do not select an object in the active screen, the properties of this screen are displayed and can be edited in properties view.

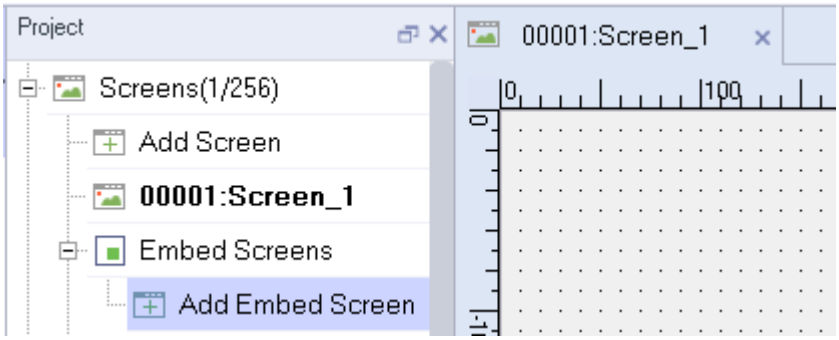
Detail view

Displays the control information in the current screen

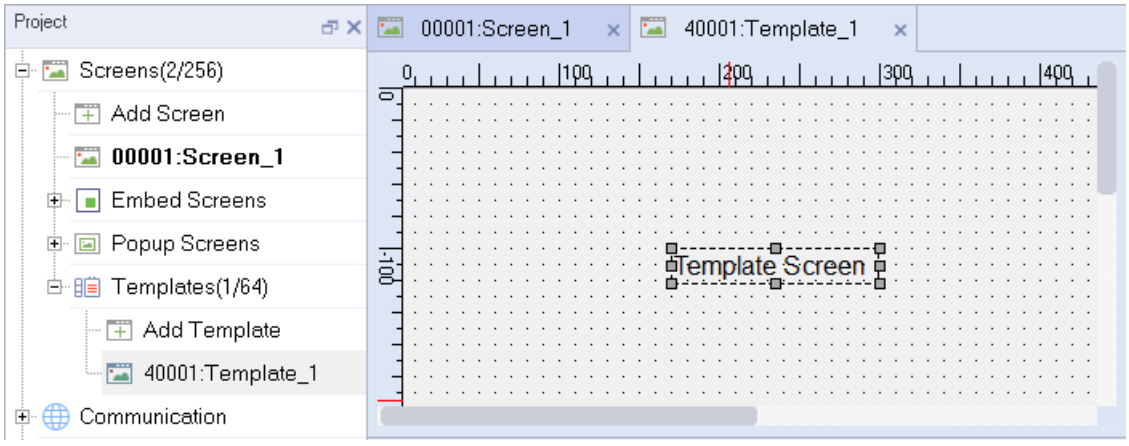
6.1.4 Using Templates and Popup Screens

Using a Template

1. Creating a template
   In the Project view, go to Screen→Templates, and double-click Add Template:

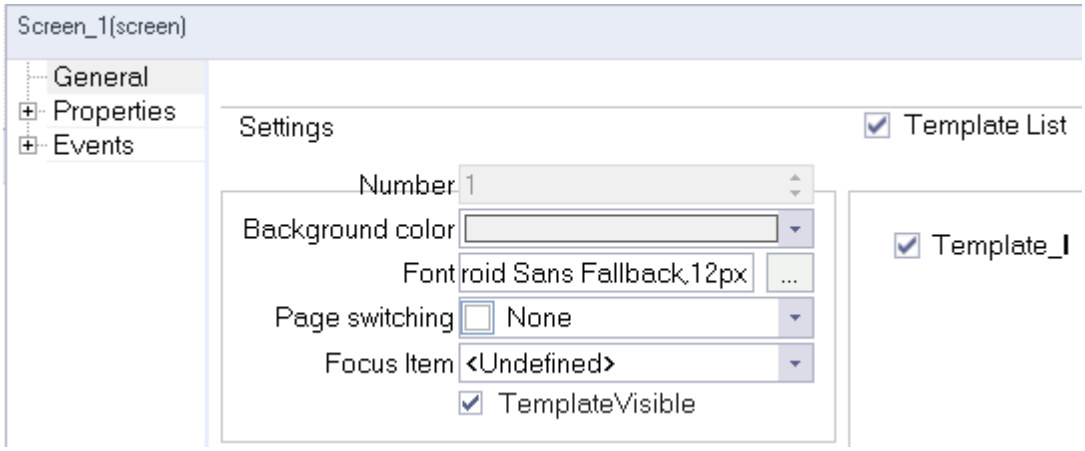


To facilitate the demonstration, first add a text field named Template Screen in the template screen:

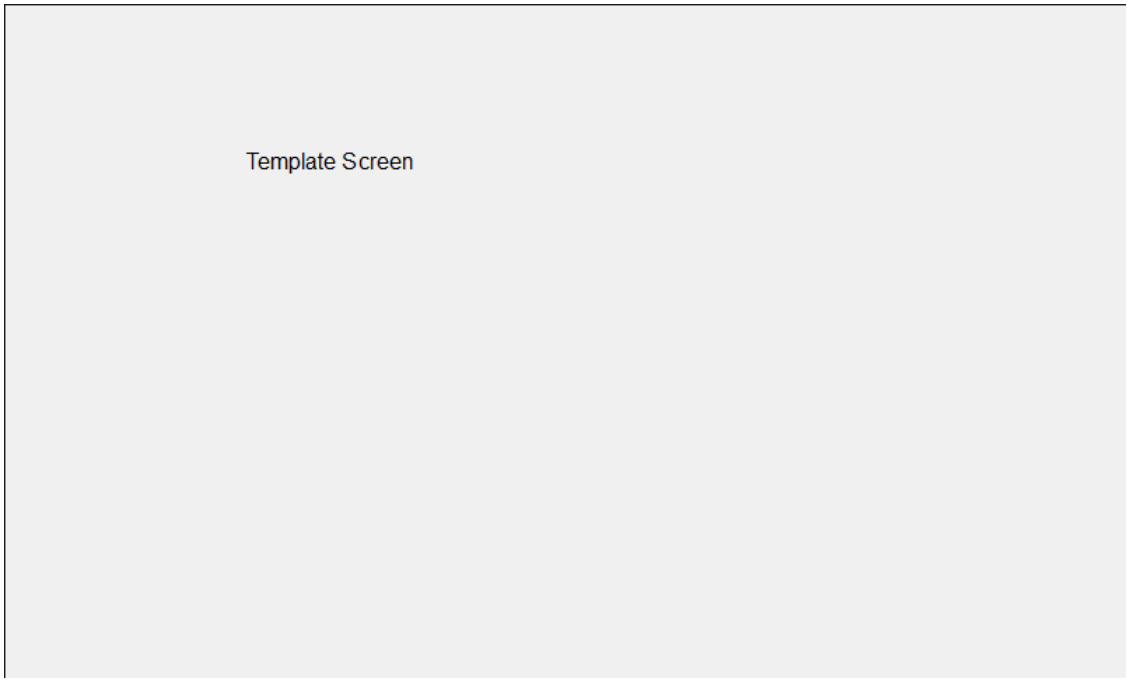


2. Using the template

In the Project view, expand Screens and double-click Screen_1, select the template you created under Template List:

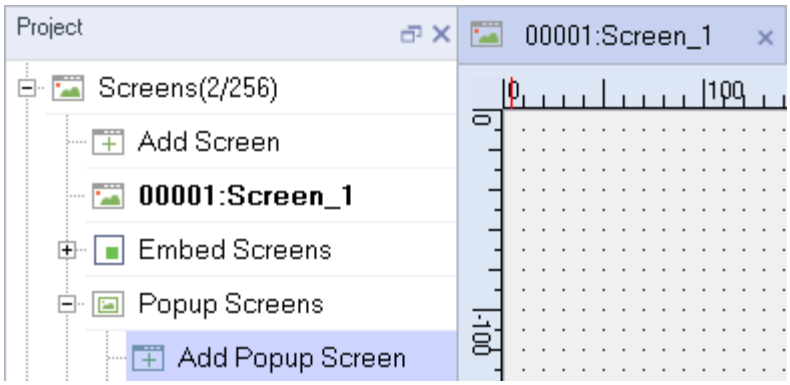


Run the project and the template screen should be applied to Screen_1:

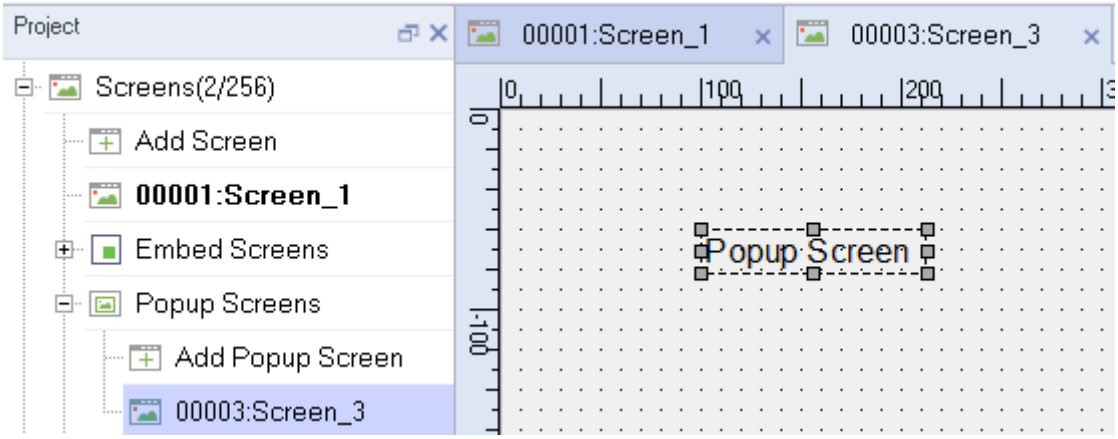


Using a popup screen

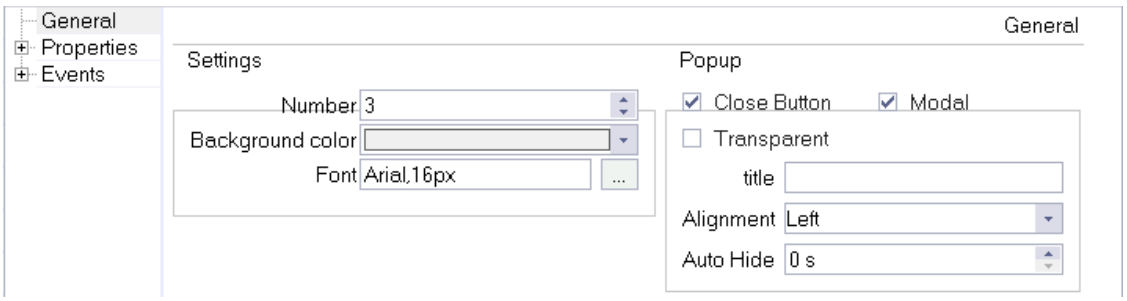
1. Creating a popup screen
   In the Project view, go to Screens→Popup Screens, and double-click Add Popup Screen:



To facilitate the demonstration, first add a text field named Popup Screen in the popup screen:



Set the properties as follows:



Settings

a. Number: The id number of the popup screen.
b. Background color.

Popup

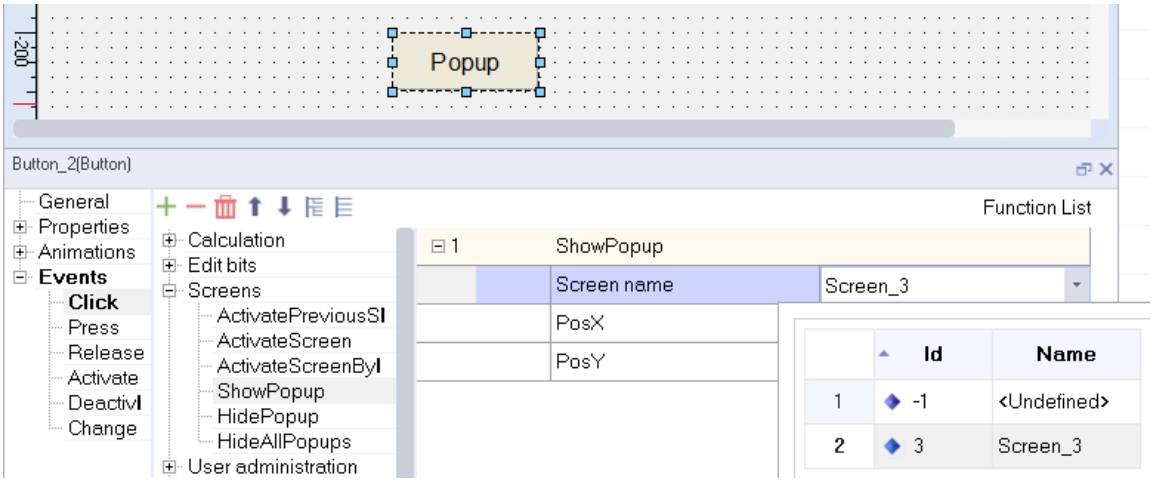
a. Select Close Button to display a close button on the screen. Select Modal to display the popup window in a modal style. Select Transparent to display a transparent popup screen.
b. You can also set the title and alignment of the popup screen.

2. Using the popup screen

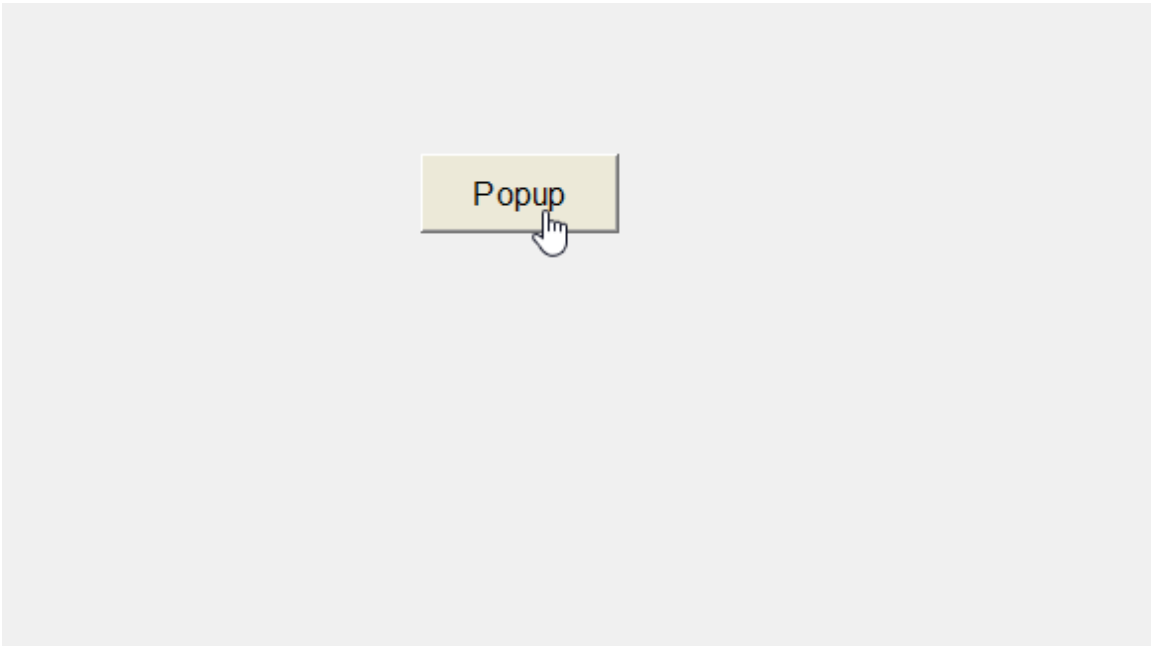
In the Project view, expand Screens and double-click Screen_1. Create a button named "Pop-up" on the screen.

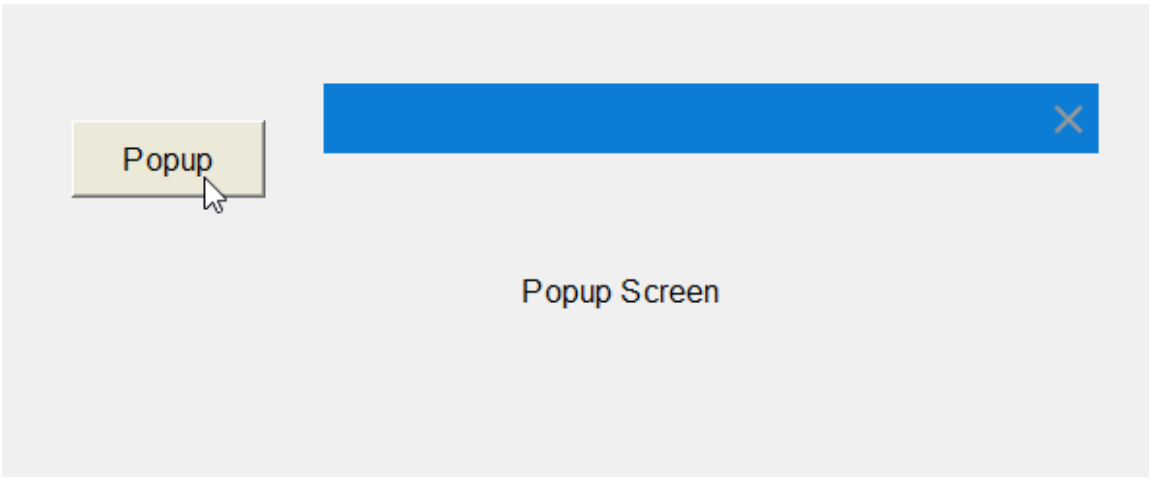
Designate the popup screen to the ShowPopup function:

Click the Popup button and select Events→Click→Screens→ShowPopup in the Properties view. Select the popup screen in the Screen name field.



Run the project and the popup screen should display when you click the Popup button:

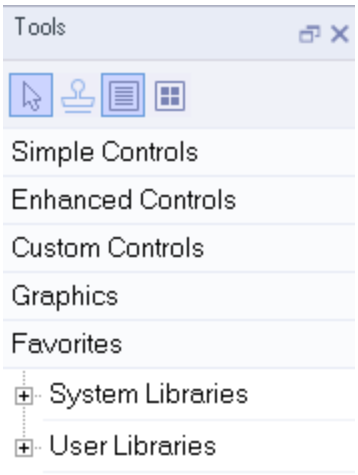




6.2 Using Screen Objects

6.2.1 Overview

The graphic elements used in project design are called "objects", which are presented in object groups in the Tools sidebar. As shown in the following figure.



Simple controls

Contains graphic objects such as polyline, polygon, and ellipses, as well as standard controls such as number IO field or button.

Enhanced controls

Contains control objects that provide enhancements. For example, the bar chart and progress bar can be used to display a process dynamically. The alarm view and data view can be used for data query.

Graphics

You can use it to browse and preview the image objects in the local computer, and add them to a project.

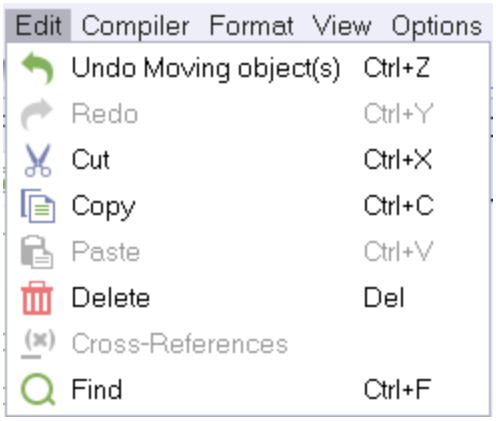
Favorites

Contains object templates, such as images of pipes, pumps, or default buttons. You can integrate multiple library object instances into a project without reconfiguration.

6.2.2 Editing Operations for Screen Objects

You can edit objects with the following operations:

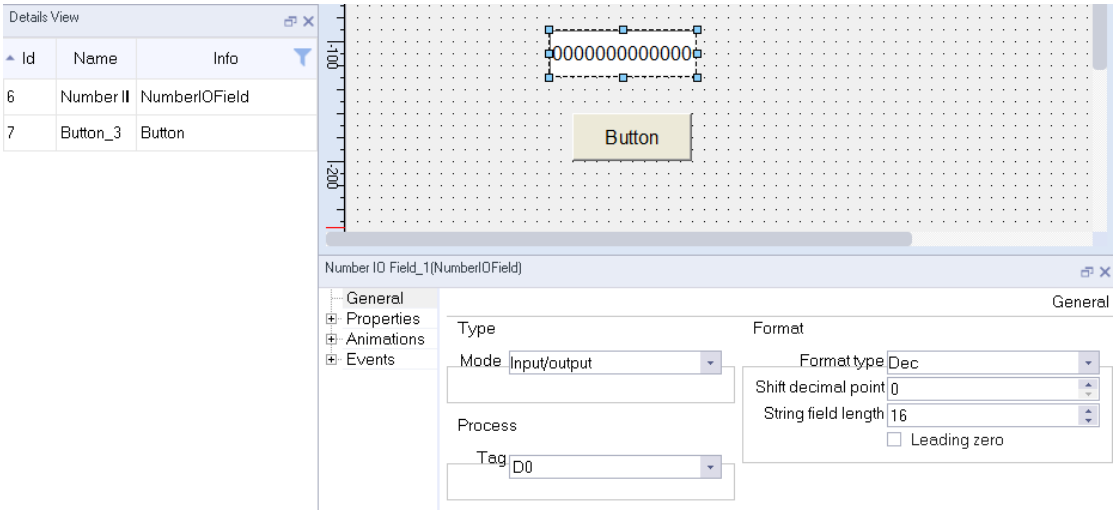
- Cut, copy, insert, and delete objects.
  To perform these operations, you can execute commands through the Edit menu, right-click menu or shortcut keys:



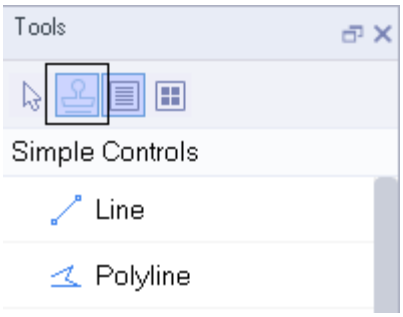
Caution

Because the system requires that the names of all objects in the screen or template must remain unique, if an object is copied to the screen, but the screen already includes an object with the same name, the name of the copied object will be automatically changed.

- Change the properties of an object, such as the size.
  Modify the properties in the property panel.
- Locate an object
  Enter the screen editor and click an object item in the details view. The corresponding object on the screen will be selected.



- Move an object in front of or behind another object.
  Right-click the object and choose an operation in the Arrange menu item. You can also use the arrangement buttons on the tool bar.
- Rotate an object
  Click the button on the tool bar to rotate the object.
- Stamp You can insert multiple objects of the same type by selecting the stamp in the Tool window.



- Select multiple objects at the same time
  Hold down Shift and click on the objects; Or click and drag the mouse in the blank space of the screen, and select multiple objects by using the selection box.
- Relocate multiple objects
  After selecting multiple objects, you can adjust all objects through the Adjust menu in the menu bar.

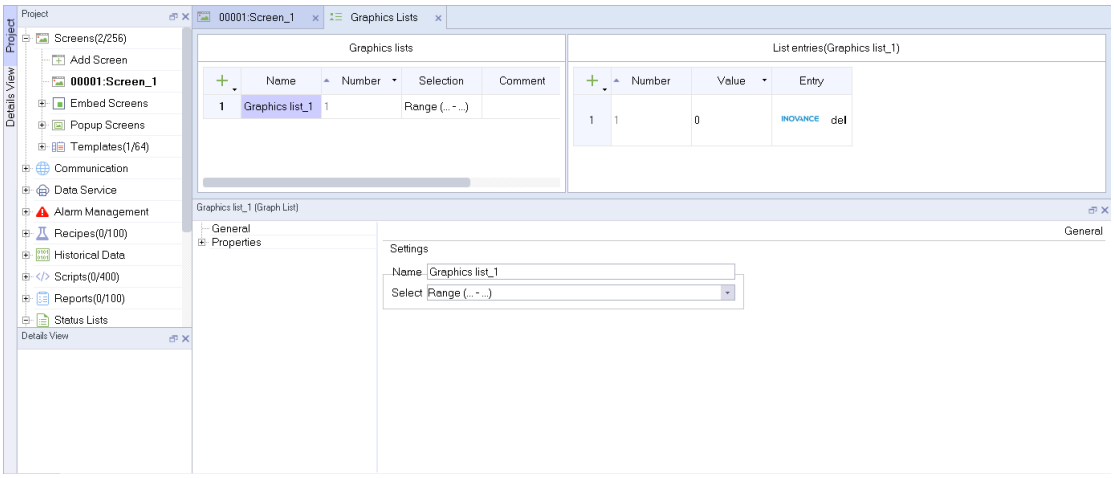
- Group
  Several objects can be combined to a group. You can edit groups in the same way as you edit any other object. To group multiple objects, select and right-click them and then select Group→Group in the context menu or use the group function in the menu bar.

6.2.3 External Graphics

In InoTouchPad, you can use images created by an external graphic editor. To use these graphics, you must store them in the InoTouchPad project's image browser. You can use images in the following formats: *. bmp, *. dib, *. ico, *. emf, *. wmf, *. gif, *. tif, *. jpeg, or *. jpg.

You can save images in the Image Browser as follows:

- Drag and drop a graphic object from Graphics in the tool window to the screen workspace, and the object will be automatically stored in the image browser.
- Go to Graphics List in Status List in project view to create a graphic list in the workspace first, then add a list entry on the right side of the workspace. Click Entry and the image browser will pop up. Click Open in the browser, and the system file browsing window will pop up. Select the graphic file to be added in the browsing window.

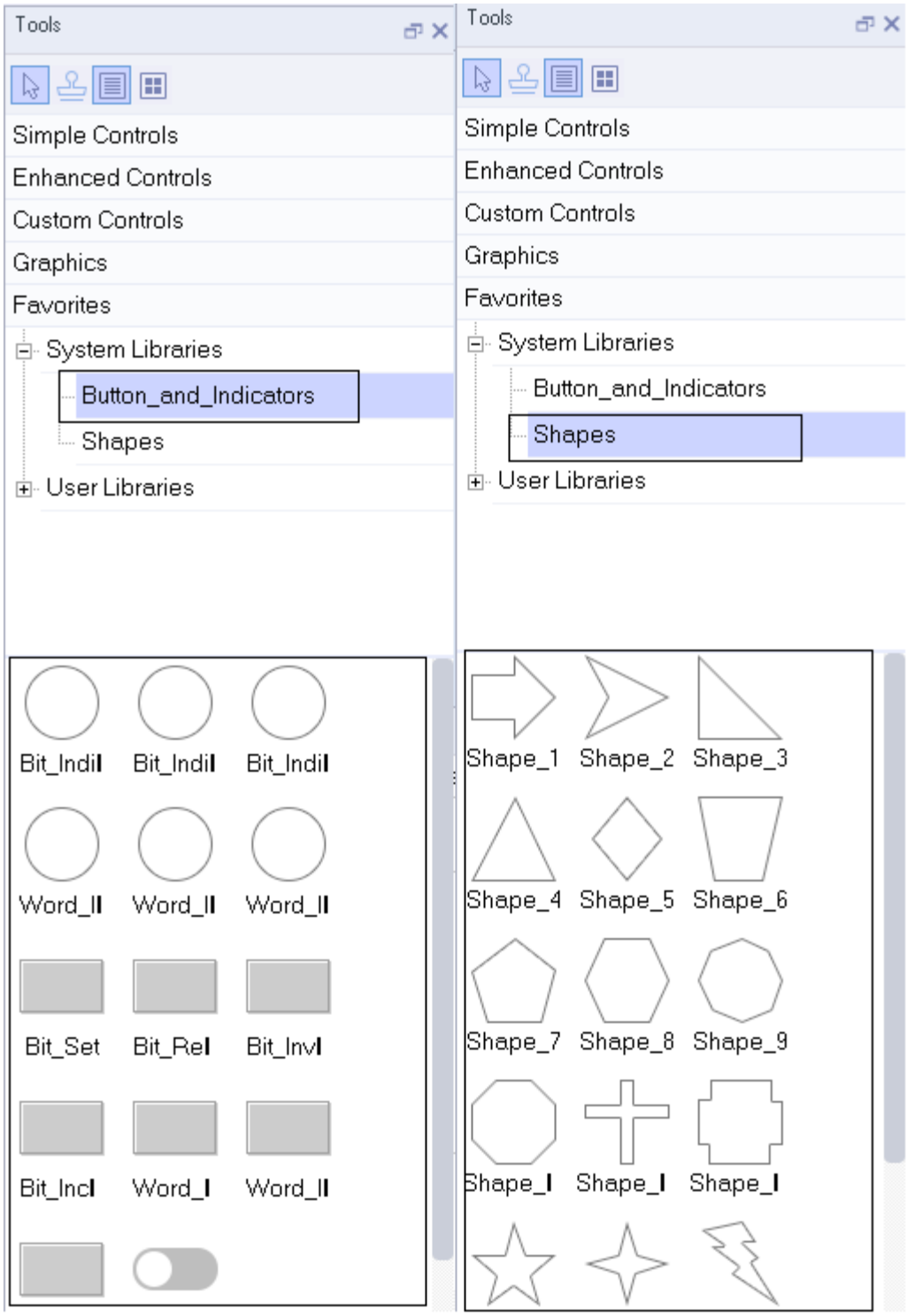


6.2.4 Favorites

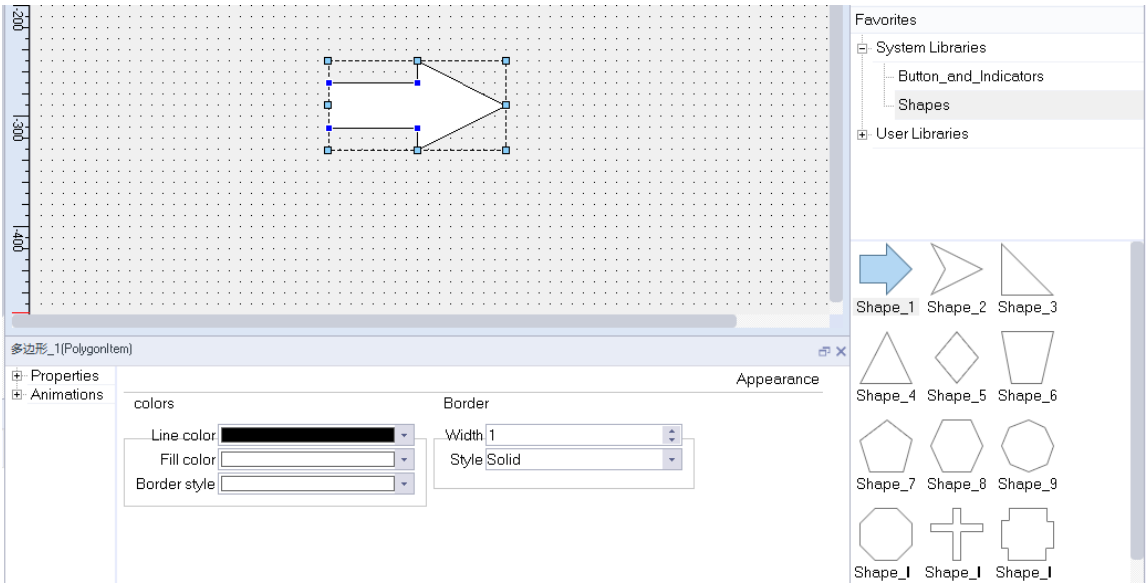
A library is a collection of screen object templates. Library objects can always be reused without reconfiguration, thus improving design efficiency. InoTouchPad includes system libraries and user libraries, and you can browse and view their contents through Favorites in the Tools window.

System libraries

They are provided by InoTouchPad and cannot be modified.



The controls in the libraries are the same as ordinary controls. You can select them in the toolbar, then put them on the screen or drag them directly to the screen for configuration, as shown in the following figure:

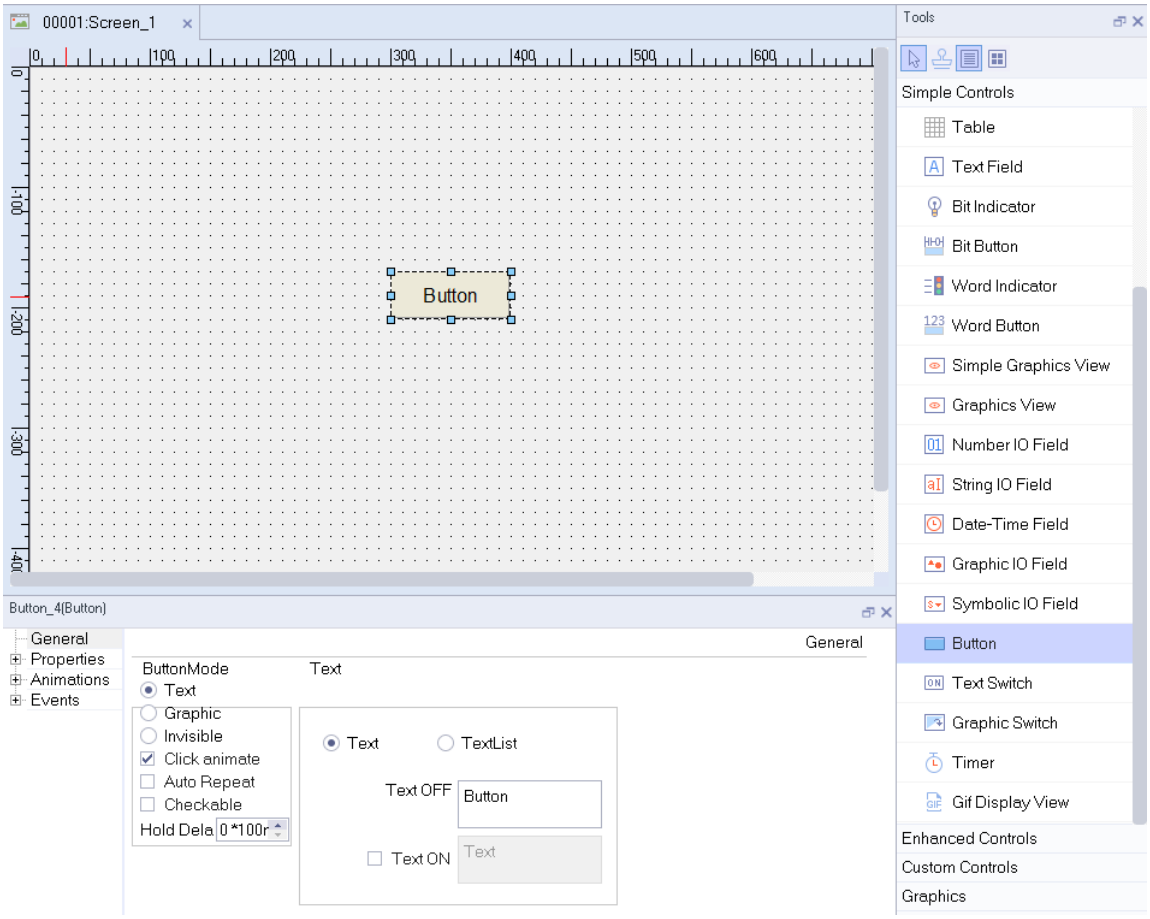


User libraries

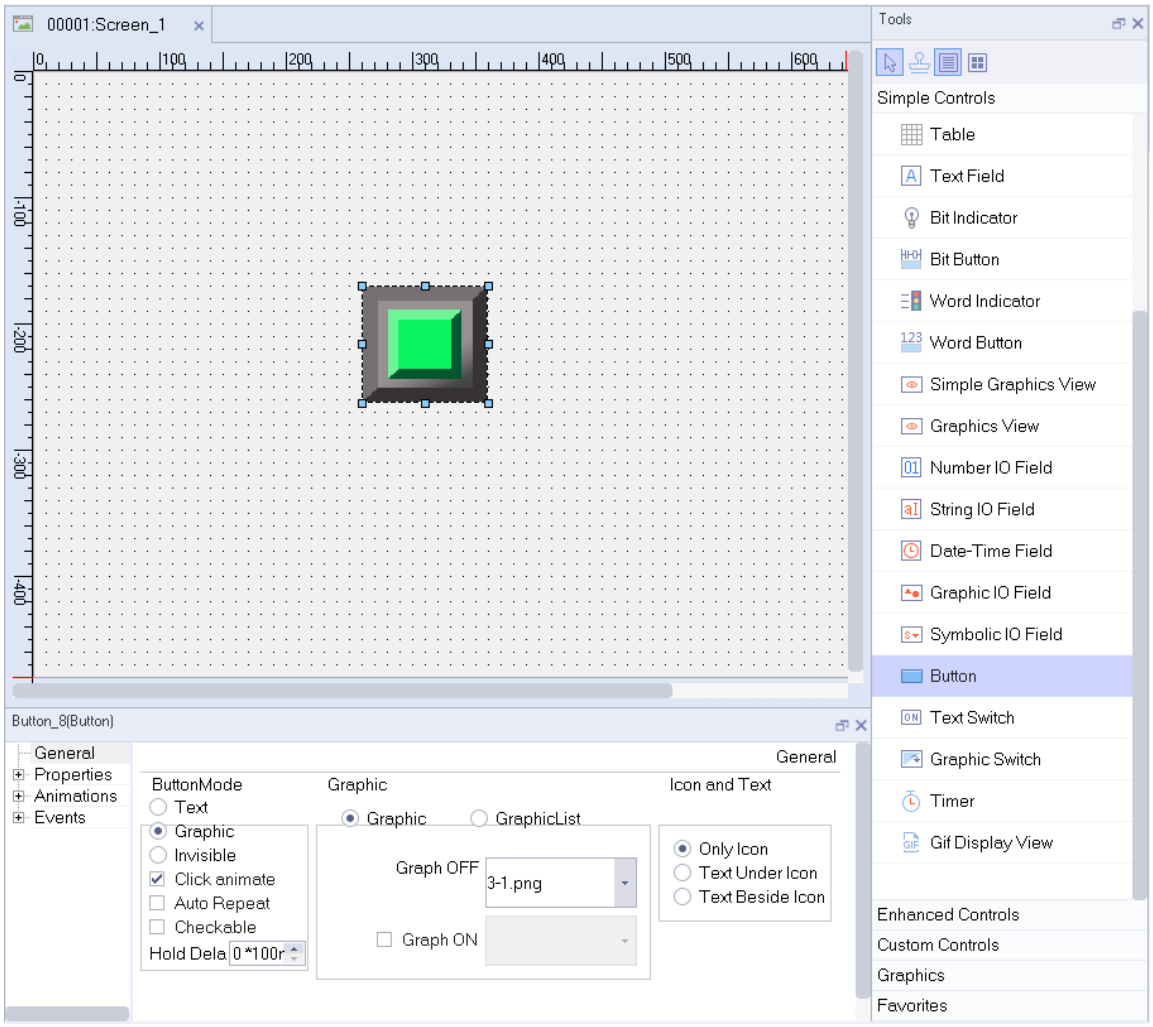
The user library is a collection of user-defined controls, which allows users to customize the form of controls, and then collect them for the next project. The collected controls exist in the user library, as shown in the figure:

The following is a detailed explanation of the collection and use of a custom button control:

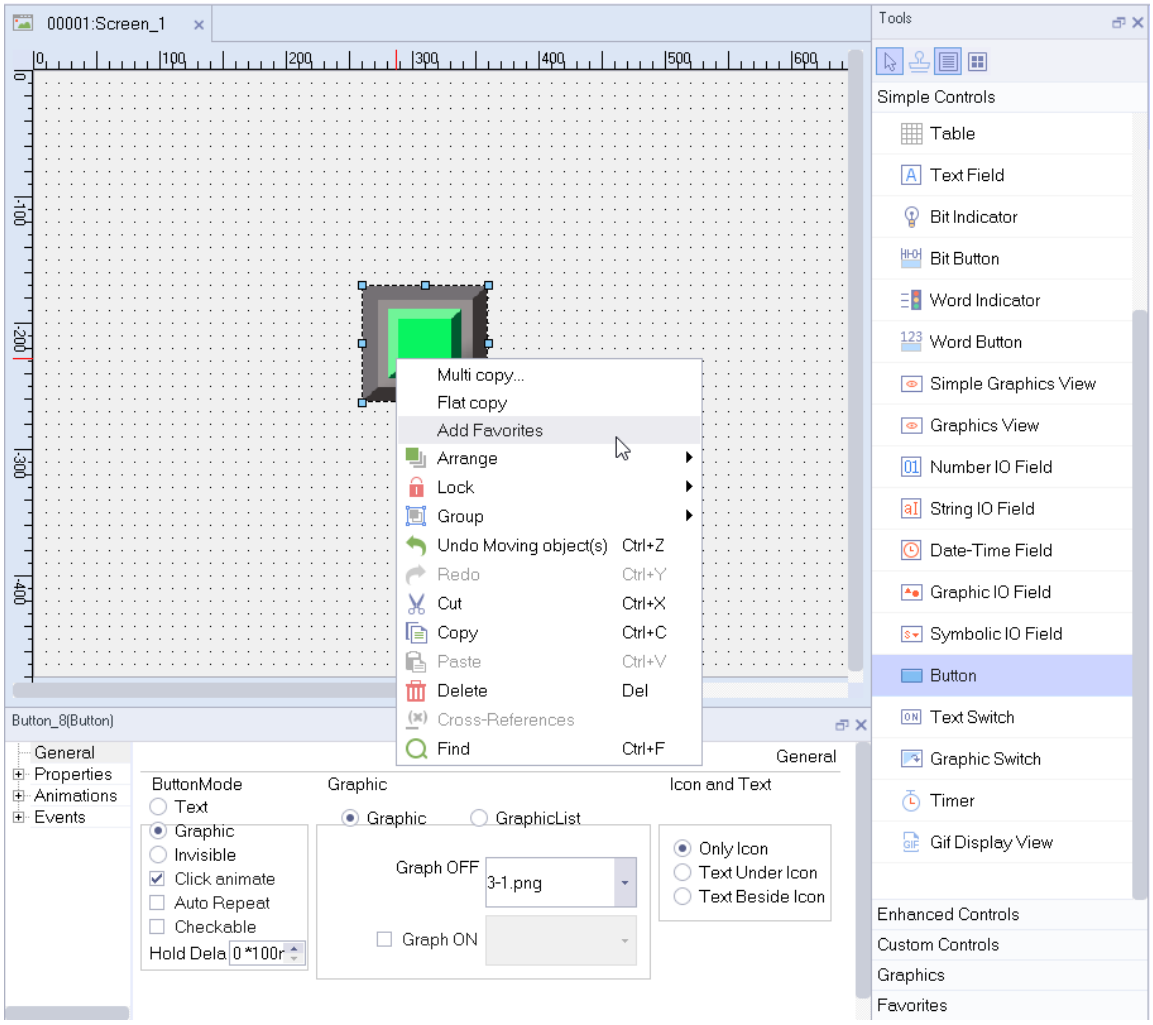
First, drag a button control from the toolbar into the screen.



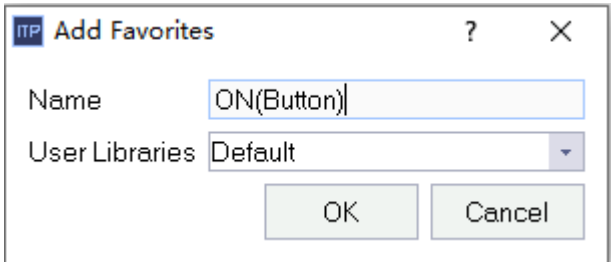
Then the button mode to Grapic and select the button image.



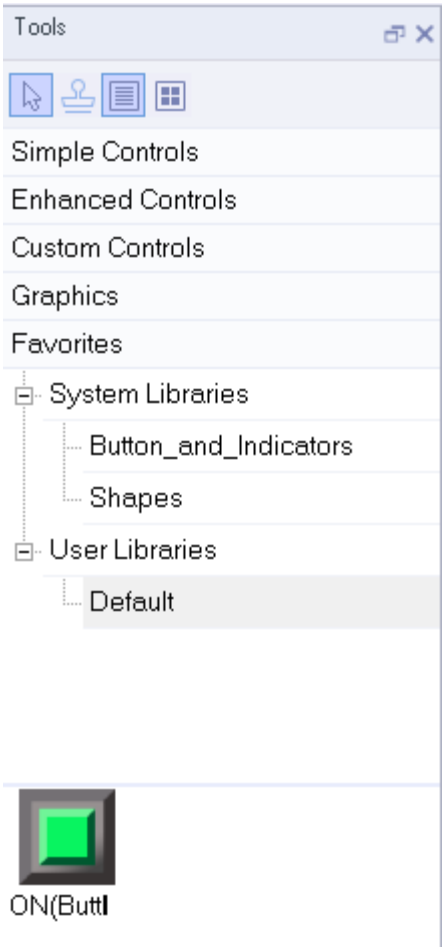
After configuring the button, right-click the button and select Add Favorites in the context menu.



You can rename the button in the popup window.



After you click OK, the button is added in Favorites → User Libraries.



If you want to use this button in this project or other projects later, you only need to drag it from the user library to the screen without reconfiguration.

In addition, if a recipe, data record, alarm log, script, report, text list and/or graphic list is configured in a custom control, when you use the control in other projects, these associated data will be automatically created. If the control is associated with tags, these tags and corresponding connections are also automatically created.

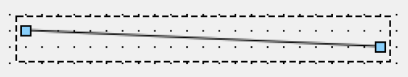
7 Controls

7.1 Simple controls

7.1.1 Line

The line is an open object. The length and slope of a line are defined by the height and width of the bounding box.

There is a blue control point at each end of the line. You can drag them to move them.



1. In the Properties dialog box in the Properties view, you can set the corresponding properties.

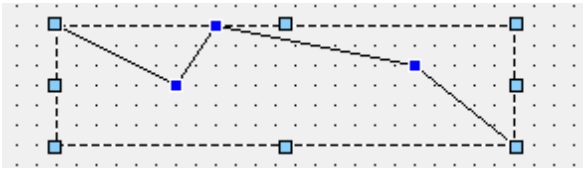
Name			Description	Remarks
Appear-ance	Color	Line color	Color of the line	
		Fill color	Fill color of the line	It only works when the blinking function is configured, blinking between Line Color and Fill Color
	Border	Width	Width of the line	Value range: 1 (default) to 30.
		Style	Style of the line. There are five styles: Solid, dash, dot, dash-dot and dash-dot-dot	The default is solid
		Line start	There are two types: Standard and Arrow	The default is standard
		Line end	There are two types: Standard and Arrow	The default is standard
		Line end shape	There are two types: Square and Round	Default is square
Layout	Start Point	X	Horizontal distance from the start point of the line to the left edge of the screen.	In pixels
		Y	Vertical distance from the start point of the line to the top edge of the screen.	In pixels
	End Point	X	Horizontal distance from the end point of the line to the left edge of the screen.	In pixels
		Y	Vertical distance from the end point of the line to the top edge of the screen.	In pixels
Flashing	Runtime Appear-ance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1

Name			Description	Remarks
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

2. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.2 Polyline

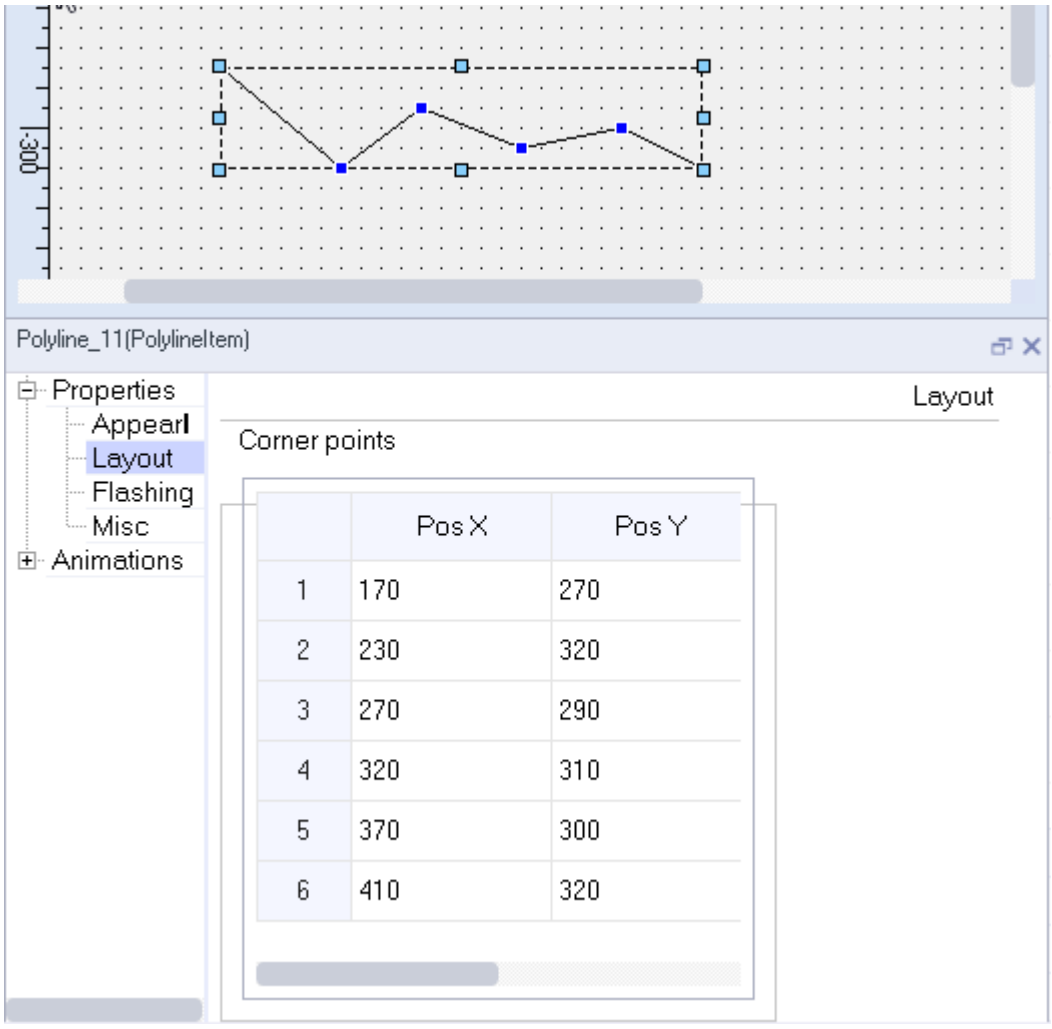
A polyline consists of connected segments and can have indefinite number of control points (numbered in the order in which they were created). The control points are blue squares which can be directly dragged to change their position. The polylie is an open object. Although the start and end points may coincide in the same coordinate, the area enclosed by the segments cannot be filled.

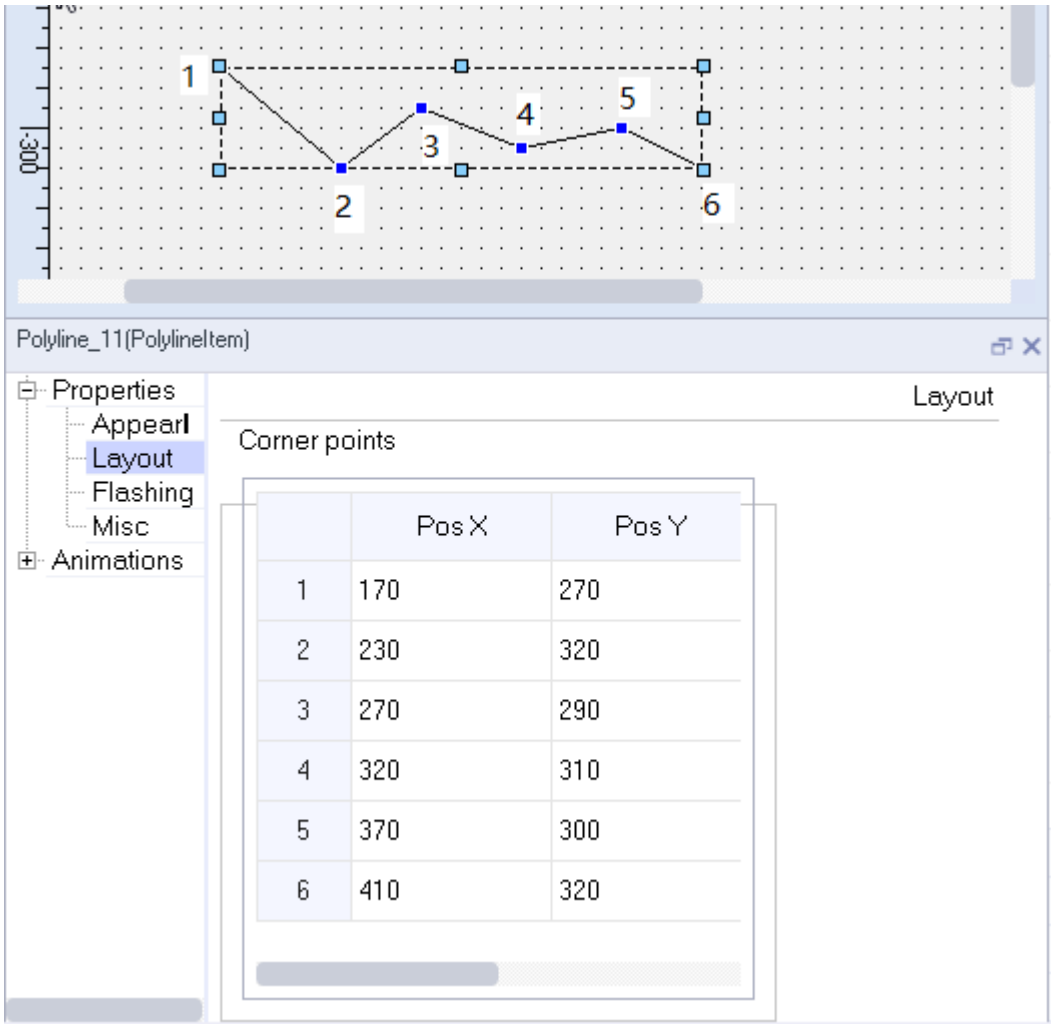


1. In the Properties dialog box in the Properties view, you can set the corresponding properties

Name			Description	Remarks
Appearance	Color	Line color	Line color of the polyline	
		Fill color	Border color	It only works when the blinking function is configured, blinking between Line Color and Fill Color
	Border	Width	Width of the border line	
		Style	There are five styles:	The default is solid
			Solid, dash, dot, dash-dot and dash-dot-dot	
Layout	Corner points	Pos X	Horizontal distance from the control point to the left edge of the screen	In pixels
		Pos Y	Vertical distance from the control point to the top edge of the screen	In pixels
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1

Name			Description	Remarks
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque



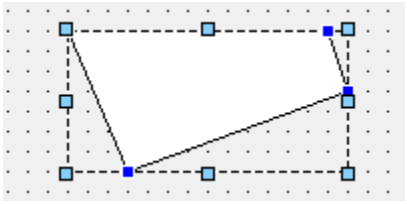


[Note] the five points in Properties-Layout-Corner points

2. For animation properties, see *"7.4.1 Animation" on page 189*.

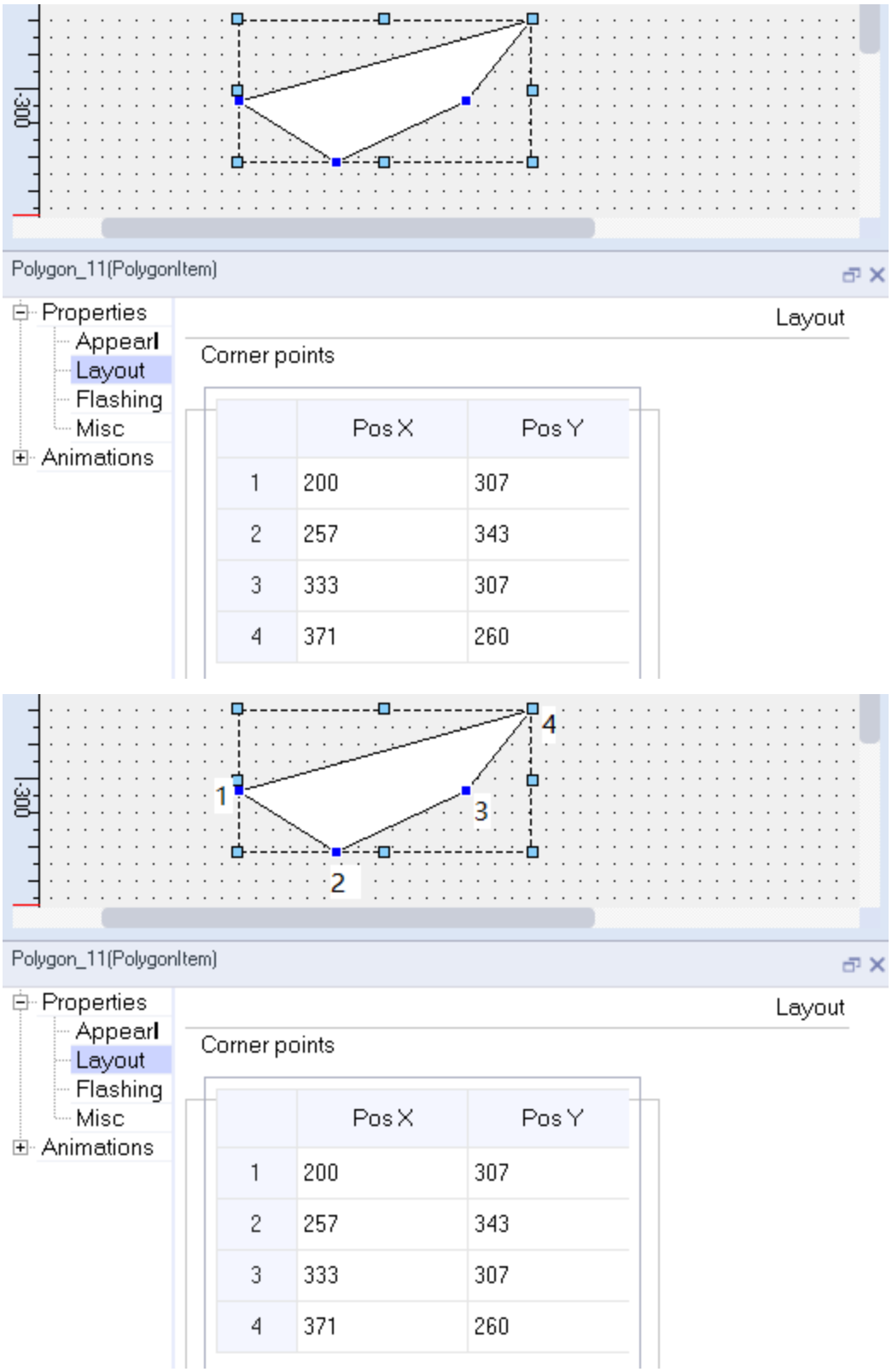
7.1.3 Polygon

The polygon is a closed object that can be filled with a background color. The control points of the polygon are numbered in the order in which they are created. To create a polygon, click the Polygon control under Simple Controls, click the screen to draw lines, and right-click to stop drawing. The control points are blue squares which you can drag to change their position.



1. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Color	Line color	Line color of the polygon	
		Fill color	Fill color of the enclosed shape	
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
	Border	Width	Width of the border line	
		Style	There are five styles: styles:	The default is solid
			Solid, dash, dot, dash-dot and dash-dot-dot	
Layout	Corner points	X-axis position	Horizontal distance from the control point to the left edge of the screen	In pixels
		Y-axis position	Vertical distance from the control point to the top edge of the screen	In pixels
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

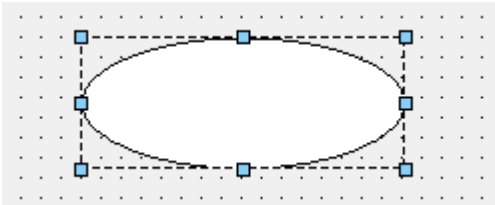


[Note] the four points in Properties-Layout-Corner points

2. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.4 Ellipse

The ellipse is a closed object that can be filled with a background color.



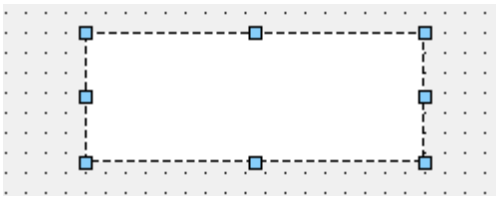
1. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Color	Border color	Border color of the ellipse	
		Fill color	Fill color of the ellipse	
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
	Border	Width	Width of the border line	
		Style	There are five styles:	The default is solid
			Solid, dash, dot, dash-dot and dash-dot-dot	
Layout	Position	X	Horizontal distance from the center of the ellipse to the left edge of the screen	In pixels
		Y	Vertical distance from the center of the ellipse to the top edge of the screen	In pixels
	Size	Width	Width of the ellipse	In pixels
		Height	Height of the ellipse	In pixels
Shape	Shape	Shape	The shapes include the pie, arc and chord.	
		Start	Starting at the rightmost point of the ellipse, offsets counterclockwise	
		Span	The circumference of the shape, which is an ellipse at 360 degrees.	
Flashing	Runtime Appear-ance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

2. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.5 Rectangle

The rectangle is a closed object that can be filled with a background color.



1. In the Properties dialog box in the Properties view, you can set the corresponding properties.

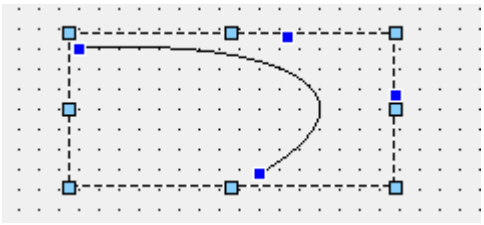
Name			Description	Remarks
Appearance	Color	Border color	Border color of the rectangle	
		Fill color	Fill color of the rectangle	
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
	Border	Width	Width of the border line	
		Style	There are five styles:	The default is solid
			Solid, dash, dot, dash-dot and dash-dot-dot	
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Shape	Shape	Shape	You can choose "rect" and "roundRect"	
	Corner Radius	Horizontal	Relationship between radius of horizontal fillet and half length of horizontal side	In pixels
		Vertical	Relationship between the radius of the vertical fillet and half the length of the vertical side	If you check Percent, the corner radius is the percentage of the half length of the corresponding side. If you do not, half of the input pixel value represents the corner radius
Flashing	Runtime Appear-ance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1

Name			Description	Remarks
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

2. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.6 Bezier

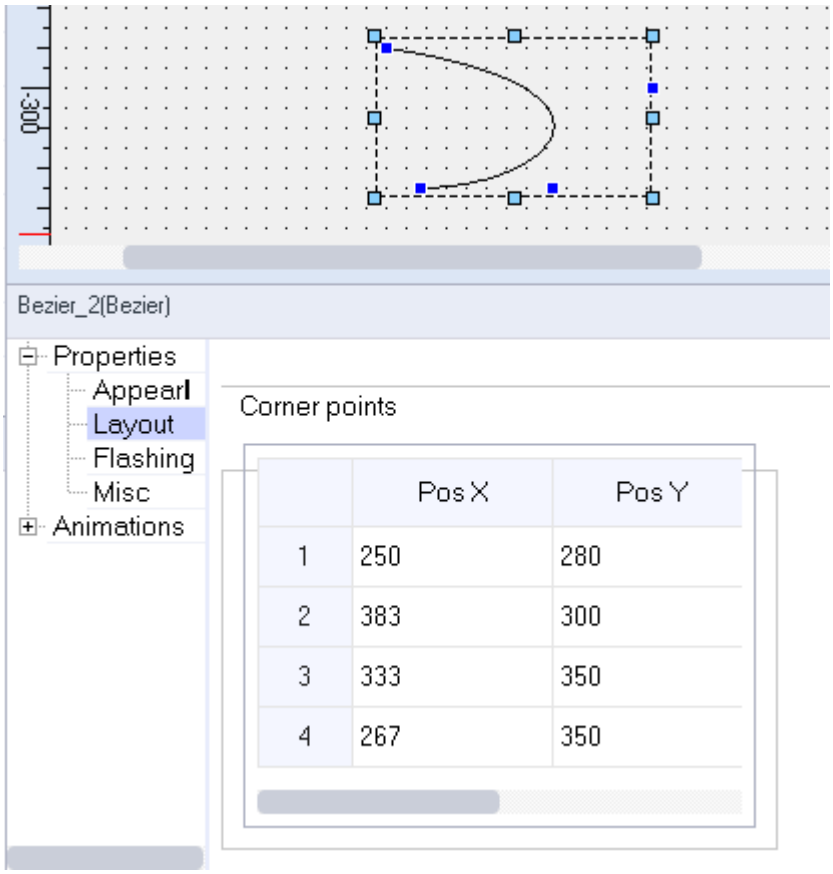
Bezier displays a Bezier curve. The control points of the curve are numbered in the order in which they are created. The control points are blue squares which you can drag to change their position.

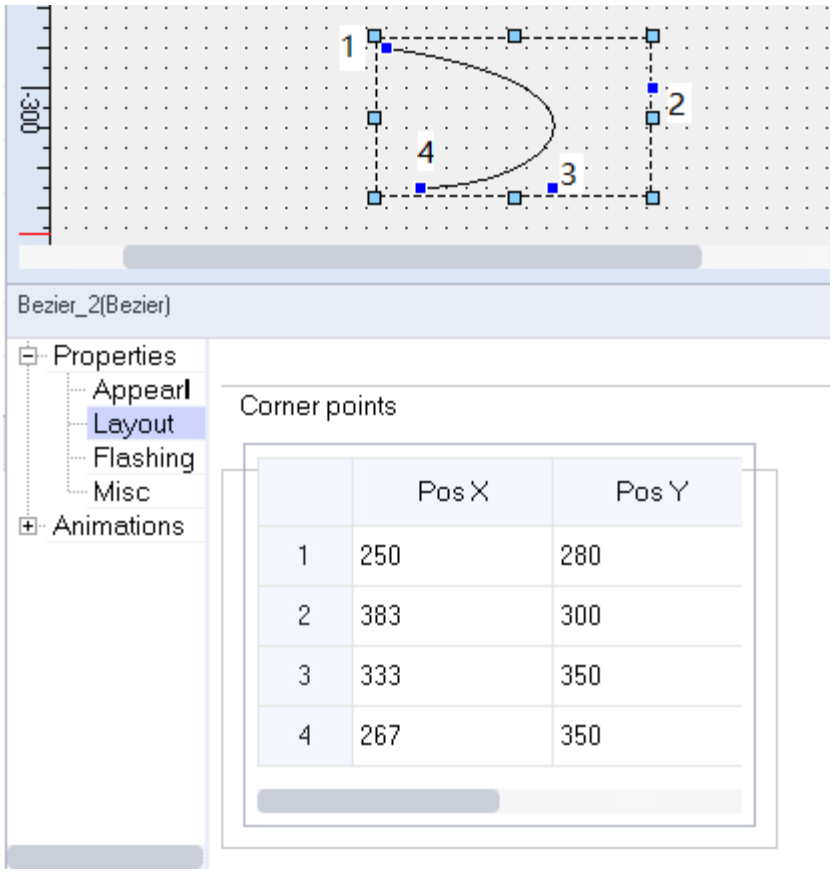


1. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appearance	Color	Border color	Border color of the rectangle	
		Fill color	Fill color of the enclosed shape	It only works when the blinking function is configured, blinking between Line Color and Fill Color
	Border	Width	Width of the border line	
		Style	There are five styles: styles:	The default is solid
			Solid, dash, dot, dash-dot and dash-dot-dot	
Layout	Corner points	Pos X	Horizontal distance from the point of the curve to the left edge of the screen	In pixels
		Pos Y	Vertical distance from the point of the curve to the top edge of the screen	In pixels
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1

Name			Description	Remarks
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque



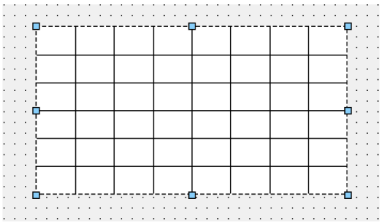


[Note] the four points in Properties-Layout-Corner points

2. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.7 Table

The table control displays a table.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Settings	Row count	The number of rows in the table	Ranges from 1 to 100, and the default value is 6
		Column count	The number of columns in the table	Ranges from 1 to 100, and the default value is 8

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

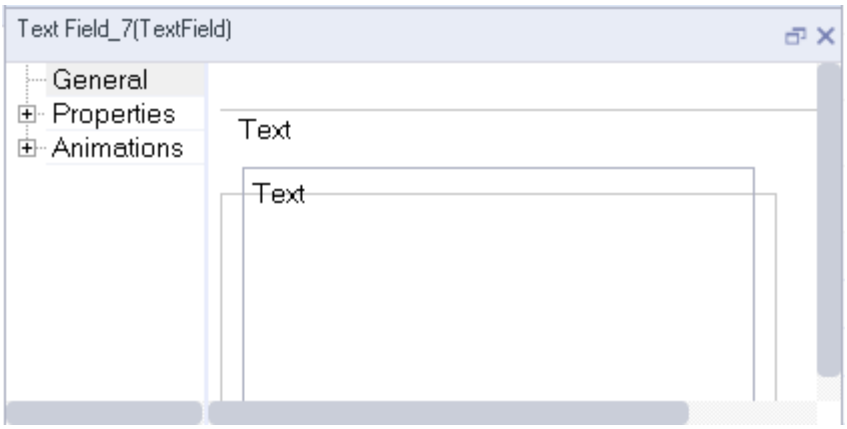
Name			Description	Remarks
Appear-ance	Color	Border color	The color of table borders	
		Fill color	The fill color of the table	
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
	Border	Width	The width of the table border	
		Style	There are five styles:	The default is solid
			Solid, dash, dot, dash-dot and dash-dot-dot	
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.8 Text Field

The text field is a closed object in which you can enter multi-line styled text. You can use it to enter text in any configured language.

1. 1. In General of the Properties view, you can enter the text of the text field. Although the length of the text can be up to 1024, it is restricted by the size of the object.



2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

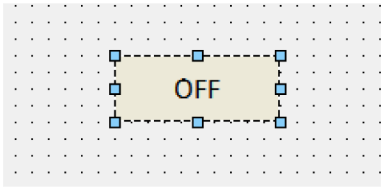
Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
	Sizing	Auto-sizing	Check this to enable auto-sizing	Auto-resizing automatically resizes the text field according to its content so that the text is displayed completely
Text	Font	Font	Text font	The font size is in pixels (px)
	Alignment	Horizontal	The horizontal alignment of text: left, center, or right	
		Vertical	The vertical alignment of text: top, middle, or bottom	
Whirl	Whirl	Speed	The text scrolling speed	In pixels
		Direction	You can set 5 directions: no direction, left to right, right to left, top to bottom, or bottom to top	
	Process	Tag	Tag to control a process	In pixels
		Value ON	The value of the tag that controls a process	Text scrolls when the tag value reaches the value of Value ON

Name			Description	Remarks
Flashing	Runtime Appear-ance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Style	Style	Style	Sets the text display style	Three default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

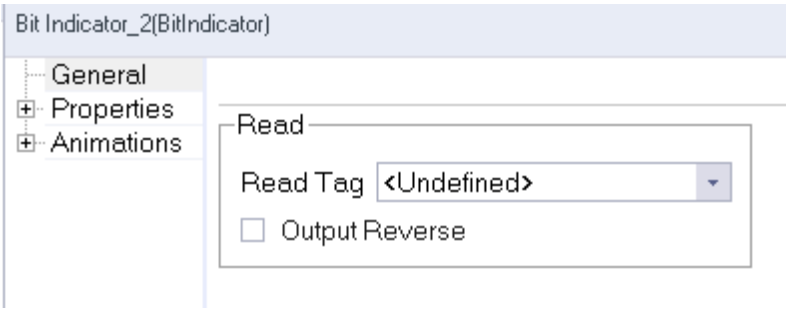
3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.9 Bit Indicator

The bit indicator is used to display the bit status of a tag. Drag the control to the screen:



1. In General of the Properties view, you can set the tag by drag-and-drop. If you check Output Reverse, when the tag is 0, the status is 1, and when the tag is 1, the status is 0.



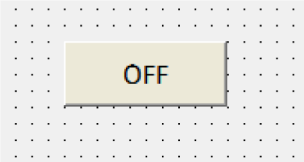
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Status	Status		There are two statuses, status 0 indicates the status when the tag is 0. Status 1 indicates the status when the tag is 1.	If you check Output Reverse, the status is opposite to the tag value.
	Graphics		Different images can be displayed in different statuses	
	Text		Different text can be displayed in different statuses	
	Font		Text font	The font size is in pixels (px)
	Position		The text alignment of the control	
	Text color		Text color	
	Background color		Background color of the text	
	Flashing		Enable/disabling flashing for a status	
	Preview		Simple preview of the control	
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authoriza-tion	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input Otherwise, input is disabled
		Visibility controlled by authoriza-tion	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.10 Bit Button

You can use the bit button to perform bit operation on the bound tag.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Read	Read Tag	The tag to be read	
		Output Reverse	Check this to reverse the read tag value.	
	Write	Read/Write Tag Same	If it is checked, it means that the tag to be written is the same as the tag to be read. If not, you need to set the write tag.	
		Mode	Set	Set the tag to 1 when the button is released.
			Reset	Set the tag to 0 when the button is released.
			Invert	Writes the inverted value.
			MomentaryON	When the button is pressed, the tag is set to 1. When the button is released, the tag is set to 0.
			MomentaryOFF	When the button is pressed, the tag is set to 0. When the button is released, the tag is set to 1.
	Press	Click animation	Check this to enable animation when you click the button	
		Auto Repeat	When the button is pressed, the event can be repeatedly triggered	
		Hold Delay	The operation is triggered with delay	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

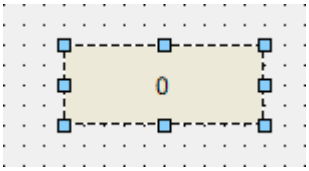
Name			Description	Remarks
State	State		There are two statuses, status 0 indicates the status when the tag is 0. Status 1 indicates the status when the tag is 1.	If you check Output Reverse, the status is opposite to the tag value.
	Graphics		Different images can be displayed in different statuses	
	Text		Different text can be displayed in different statuses	
	Font		Text font	The font size is in pixels (px)
	Position		The text alignment of the control	
	Text color		Text color	
	Background color		Background color of the text	
	Flashing		Enables/disables flashing for a status	
	Preview		Simple preview of the control	
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Style	Style	Style	Sets the button display style.	Seven default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authoriza-tion	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input Otherwise, input is disabled
		Visibility controlled by authoriza-tion	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

Name			Description	Remarks
Operation Record	Record Tag		You can enable or disable the operation record here	The operation record is enabled if the tag value is not 0. It is also enabled if no tag is configured.

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.11 Word Button

The word button is used to display the word status of a tag. Drag the control to the screen:



1. In General of the Properties view, you can set the tags by drag-and-drop.

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

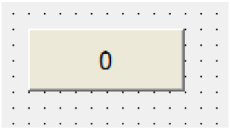
Name			Description	Remarks
Status	Status		There is one default status, and you can add more. Each status is in a different range	
	Graphics		Different images can be displayed in different statuses	
	Text		Different text can be displayed in different statuses	
	Font		Text font	The font size is in pixels (px)
	Position		The text alignment of the control	
	Text color		Text color	
	Background color		Background color of the text	
	Flashing		Enable/disabling flashing for a status	
	Preview		Simple preview of the control	
Appearance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles

Name			Description	Remarks
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.12 Word Button

You can use the word button to perform word operation on the bound tag.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Read	Read Tag	The tag to be read	
	Write	Read/Write Tag Same	If it is checked, it means that the tag to be written is the same as the tag to be read. If not, you need to set the write tag.	
		Mode	Increase	Status-independent. The increment can be a constant value or tag value.
			Decrease	Status-independent. The decrement can be a constant value or tag value.
			Cycle Increase	Status-dependent. the current value is the minimum value of the next status, and the status value is reset to the minimum after reaching the maximum value. The default status is not involved in the cycle.
			Cycle Decrease	Status-dependent. the current value is the minimum value of the previous status, and the status value is reset to the maximum after reaching the minimum value. The default status is not involved in the cycle.
			Status Increase	Status-dependent. The current value is the minimum value of the next status. After reaching the maximum value, the status stops increasing.
			Status Decrease	Status-dependent. The current value is the minimum value of the previous status. After reaching the minimum value, the status stops decreasing.
		Constant	Constant value	It is only effective in the increase and decrease mode, and cannot be used with the tag at the same time.
		Tag	Tag value	It is only effective in the increase and decrease mode, and cannot be used with the constant at the same time.
	Press	Click animation	Check this to enable animation when you click the button	
		Auto Repeat	When the button is pressed, the event can be repeatedly triggered	
		Hold Delay	The operation is triggered with delay	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

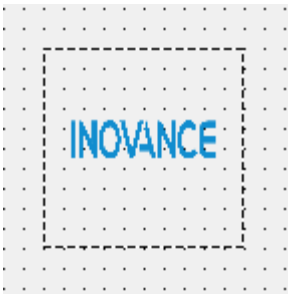
Name			Description	Remarks
State	State		There can be a variety of statuses, and one default status is provided. The value ranges of these statuses are different and you can set the status range.	
	Graphics		Different images can be displayed in different statuses	
	Text		Different text can be displayed in different statuses	
	Font		Text font	The font size is in pixels (px)
	Position		The text alignment of the control	
	Text color		Text color	
	Background color		Background color of the text	
	Flashing		Enables/disables flashing for a status	
	Preview		Simple preview of the control	
Appearance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Style	Style	Style	Sets the button display style.	Seven default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

Name			Description	Remarks
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control
		Confirm Before Operation	Check this to pop up a confirm window before operation	When this is checked, you need to confirm the system security first
Operation Record	Record Tag		You can enable or disable the operation record here	The operation record is enabled if the tag value is not 0. It is also enabled if no tag is configured.

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.13 Simple Graphics View

The simple graphic view is used to display graphics.



1. In General of the properties view, you can set the image to be displayed.

Open the file browser to import images

Delete the selected image

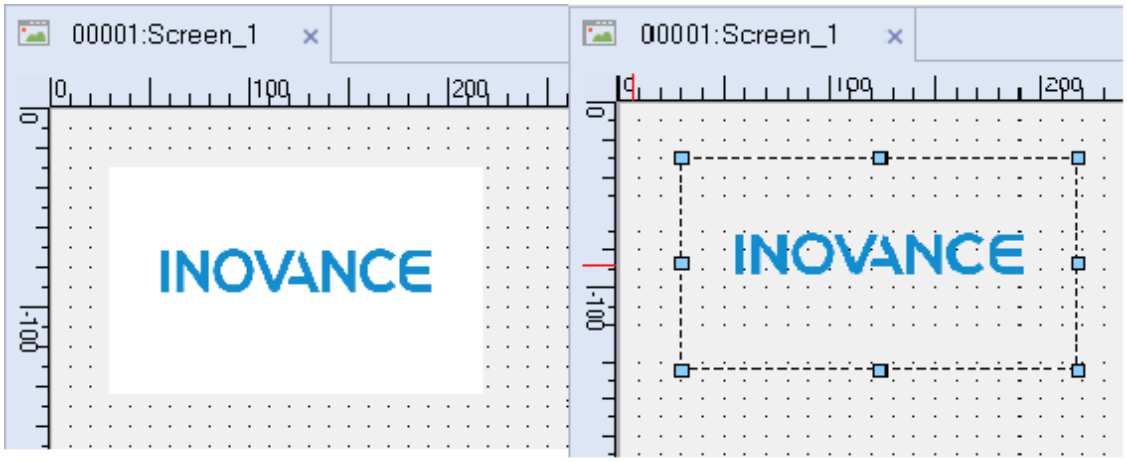
Show the images in thumbnails

Show the images in a list

The Set button sets the image to be displayed

The Clear button clears the displayed image

The Transparent button sets the image background transparent. Take the default image as an example:



The button on the right of the Transparent button is used to set the image background color

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

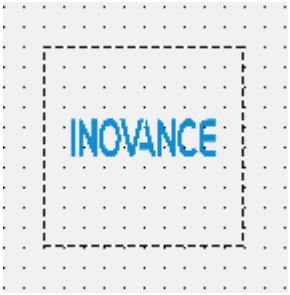
Name			Description	Remarks
Appear-ance	Fill	Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels, can be a tag value.
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels, can be a tag value.
	Size	Width	Width of the rectangle	In pixels, can be a tag value.
		Height	Height of the rectangle	In pixels, can be a tag value.
	Rotate	Rotate	Rotates the image by a given angle	Rotates the image by an angle based on the tag value.
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1

Name			Description	Remarks
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

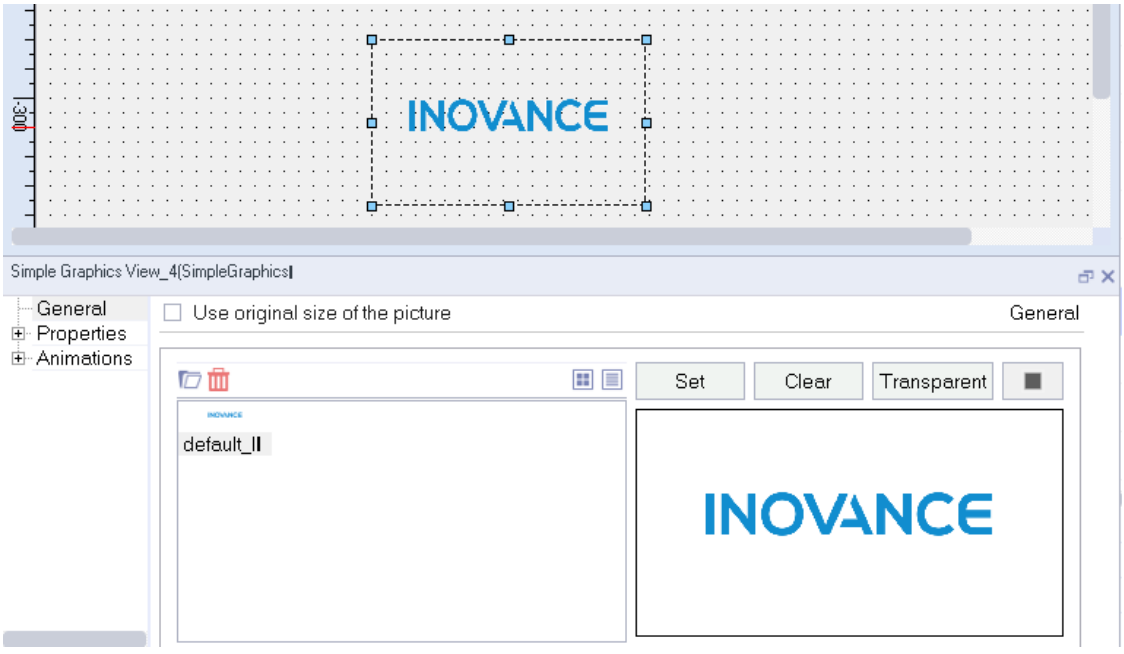
3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.14 Graphics View

The graphics view is used to display graphics.



1. In General of the properties view, you can set the image to be displayed.



Open the file browser to import images

Delete the selected image

Show the images in thumbnails

Show the images in a list

The Set button sets the image to be displayed

The Clear button clears the displayed image

The Transparent button is used to remove the background color of the currently displayed image and make it transparent

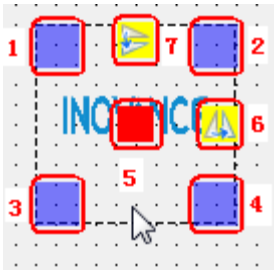
The button is used to set the background color of the image

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appearance	Fill	Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

In addition, you can also set the graphics view by manipulating the 7 control points on the view:



Point 1: You can drag this point to resize or rotate the view. Hold down shift while dragging to only adjust the size and hold down ctrl while dragging to only rotate the view. Double-click the point to undo the rotation.

Points 2, 3 and 4 are the same as point 1.

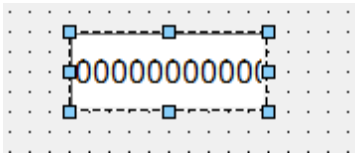
Point 5: You can drag this point to rotate the view around the X or Y axis. Double-click the point to undo the rotation.

Point 6: Click this point to flip the view horizontally.

Point 7: Click this point to flip the view vertically.

7.1.15 Number IO field

The number IO field is used to enter and display process values.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Type	Mode	There are three modes: input, input/output, and output	
	Process	Tag	The tag to display a process	
	Format	Format type	Bin, dec, hex, and BCD	
		Shift decimal point	The position of decimal point: 0–16	The shift range is determined by the length of the string field
		String field length	String length: 1–64	
		Leading zero	Check this to pad the number with leading zeros	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
	Size	Auto-sizing	Check this to enable auto-sizing	Auto-resizing automatically resizes the control according to the text length
Text	Font	Font	Text font	The font size is in pixels (px)
	Align-ment	Horizontal	The horizontal alignment of text: left, center, or right	
		Vertical	The vertical alignment of text: top, middle, or bottom	
Flashing	Runtime Appear-ance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Scaling	Gain	Static	Gain between screen display value and PLC value	Value written to PLC = (Input value - Offset)/Gain;
	Offset	Static	Offset between screen display value and PLC value	Value displayed on HMI = (Output Value * Gain) + Offset
		Tag	Process tag	Process tag associated with gain and offset settings
Limits	Color	Color above upper limit	The color displayed when upper limit is exceeded	
		Color below lower limit	The color displayed when lower limit is exceeded	
Style	Style	Style	Sets the display style.	Two default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

Name			Description	Remarks
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control
		Confirm Before Operation	Check this to pop up a confirm window before operation	When this is checked, you need to confirm the system security first
		SoftwareInput Enable	Enables the on-screen keyboard	Check this to use the on-screen keyboard when inputing the password
		Password input	Password display	Check this to display the password with asterisks
Operation Record	Record Tag		You can enable or disable the operation record here	The operation record is enabled if the tag value is not 0. It is also enabled if no tag is configured.

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.16 String IO field

The string IO field is used to enter and display process values.

1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Type	Mode	There are three modes: input, input/output, and output	
	Process	Tag	The tag to dispaly a process	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

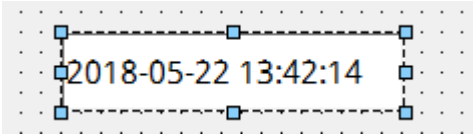
Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
	Size	Auto-sizing	Check this to enable auto-sizing	Auto-resizing automatically resizes the control according to the text length
Text	Font	Font	Text font	The font size is in pixels (px)
	Alignment	Horizontal	The horizontal alignment of text: left, center, or right	
		Vertical	The vertical alignment of text: top, middle, or bottom	
Whirl	Whirl	Speed	Sets the text scrolling speed	The speed unit is 1
		Direction	Sets the text scrolling direction	You can set 5 directions: no direction, left to right, right to left, top to bottom, or bottom to top
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Style	Style	Style	Sets the display style.	Two default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

Name			Description	Remarks
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control
		SoftwareInput Enable	Enables the on-screen keyboard	Check this to use the on-screen keyboard when inputing the password
		Password input	Password display	Check this to display the password with asterisks

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.17 Date-Time Field

The date-time field is used to enter and display time and date.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Type	Mode	There are two modes: input/ output and output	
	Process	Display system time	Check this to display system time	
		Use tag	Check this to display the time set in the tag	
		Tag	The tag to be used	Only enabled when you check Use tag
	Format	Display date	Check this to display date	Check at least one of the three. "Display weekday" is available only when "Display system time" is checked.
		Display time	Check this to display time	
		Display weekday	Check this to display the day of the week	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

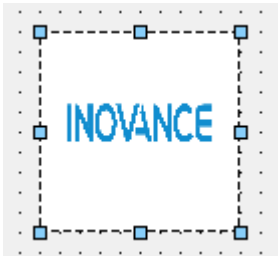
Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
	Size	Auto-sizing	Check this to enable auto-sizing	Auto-resizing automatically resizes the control according to the text length
Text	Font	Font	Text font	The font size is in pixels (px)
	Alignment	Horizontal	The horizontal alignment of text: left, center, or right	
		Vertical	The vertical alignment of text: top, middle, or bottom	
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Style	Style	Style	Sets the display style.	Two default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Operation Record	Record Tag		You can enable or disable the operation record here	The operation record is enabled if the tag value is not 0. It is also enabled if no tag is configured.

Name			Description	Remarks
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control
		SoftwareInput Enable	Enables the on-screen keyboard	Check this to use the on-screen keyboard when inputing the password
		Password input	Password display	Check this to display the password with asterisks

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.18 Graphic IO Field

The graphic IO field object can be used to display many different states for multiple images in the graphic list.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Settings	Mode	Mode	Values: output, input, input/output
				Default: input/output
	Process	Tag	Process tag	Defaults to null, types: int16, UInt16, int32, UInt32, float, double, bool, Int64, and UInt64, array is not supported
		Bit number	Bit number of the process tag	Only when you select "Bit (0, 1)" in Graphic list can it be edited;
				Value range: Word: 0-15 DWord: 0-63
				Default: 0
	Display	Graphic List	Graphic list	Default is null
		Scroll bar orientation	Scroll bar orientation	Values: vertical and horizontal
				Default: vertical

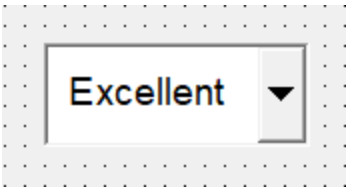
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Background color	Background color	
		Fill style	Fill style	
Layout	Position	X	X coordinate of the object	In pixels. Note: The object cannot move out of the screen
		Y	Y coordinate of the object	
	Size	Width	Object width	
		Height	Object height	
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authoriza-tion	Runtime Security	Default is null
				Not configurable in output mode
				Only authorized users can operate
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authoriza-tion	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.1.19 Symbolic IO Field

The symbolic IO field is a dropdown list of input and output text.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Settings	Mode	There are three modes: input, input/output, and output	
	Process	Tag	Tag to control the displayed information	
		Bit number	Bit number of the tag	
	Display	Text List	The text list to be linked to the object	
		Number of visible items	Number of visible items	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appearance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Text	Font	Font	Text font	The font size is in pixels (px)
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Limits	Colors	Color above upper limit	The color displayed when upper limit is exceeded	
		Color below lower limit	The color displayed when lower limit is exceeded	
StyleSheet	StyleSheet	StyleSheet	Sets the display style	Two default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

Name			Description	Remarks
Security	Runtime Security	Authorization	Runtime Security	Default is null
				Not configurable in output mode
				Only authorized users can operate
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.20 Button

The button is used to execute a command at run time.

The button can be displayed as text, a graphic or invisible:

1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Button Mode	Text	Text/Text List	When you select Text, you can edit and set Text OFF/ON. When you select Text List, you can control the display of the specified text in the text list through process tags.
		Graphic	Graphic/Graphic List	When you select Text, you can edit and set Graph OFF/ON. When you select Graphic List, you can control the display of the specified graphic in the graphic list through process tags. In addition, you can also set the icon text display style.
		Invisible	Check this to hide the button during operation	

In the General page, you can also set Click Animation and Auto Repeat. If you check Click Animation, you can play an animation when clicking the button. If you check Auto Repeat, one button click can trigger the event repeatedly. If you tick Checkable, the button keeps pressed or released when you click it, just like a check box.

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

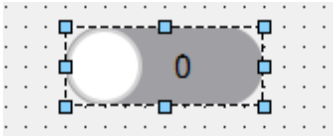
Name			Description	Remarks
Appearance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
	Size	Auto-sizing	Check this to enable auto-sizing	Auto-resizing automatically resizes the control according to the text length
Text	Font	Font	Text font	The font size is in pixels (px)
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Style	Style	Style	Sets the button display style.	Seven default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control
		Confirm Before Operation	Check this to pop up a confirm window before operation	When this is checked, you need to confirm the system security first

3. For animation properties, see *"7.4.1 Animation" on page 189*.

4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.21 Text Switch

The text switch is a switch which changes state at runtime. You can visualize the current state of the switch through text.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Text	Text ON	The text displayed when the status is ON	
		Text OFF	The text displayed when the status is OFF	
		Switch	Check this to use the control with dynamic effects. When it is not checked, the switch becomes a button.	
		Click animation	Enables animation when you click the switch	
	Process	Tag	Tag to control a process	
		Value ON	The state is ON when the tag reaches the set value	The default is 1, which cannot be changed

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

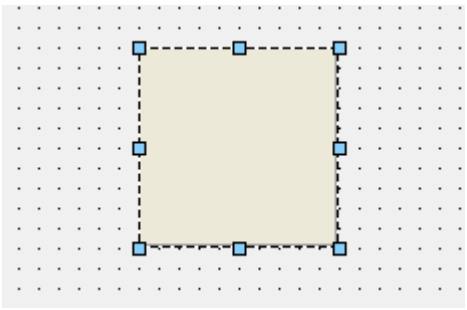
Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
	Size	Auto-sizing	Check this to enable auto-sizing	Auto-resizing automatically resizes the control according to the text length
Text	Font	Font	Text font	The font size is in pixels (px)

Name			Description	Remarks
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Limits	Color	Color above upper limit	The color displayed when upper limit is exceeded	
		Color below lower limit	The color displayed when lower limit is exceeded	
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.22 Graphic Switch

The graphic switch is a switch which changes state at runtime. You can visualize the current state of the switch through graphics.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Graphics	Graphic ON	The image displayed when the status is ON	
		Graphic OFF	The image displayed when the status is OFF	
		Click animation	Enable animation when you click the switch	
	Process	Tag	Tag to control a process	
		Value ON	The state is ON when the tag reaches the set value	The default is 1, which cannot be changed

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

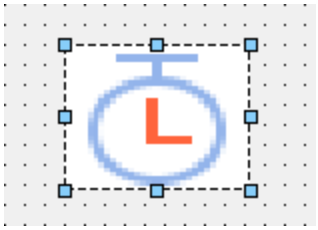
Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.23 Timer

The timer is used to trigger events at intervals. Events can be triggered once or in a loop. The configured timer only takes effect in the configuration screen, and only when the configuration screen

displays, the event can be triggered. If you need to trigger event globally, you can set a timing job in the scheduler.



1. In General of the properties view, you can set its properties.

Name			Description	Remarks
General	Settings	Interval	Time interval of the timer, ranging from 1–600, in 100 ms	
		Single shot	Check this to enable one-time triggering.	
	Process	Tag	Tag to control a process	
		Value ON	Starts timing when the tag reaches the set value	The default is 1, which cannot be changed

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.1.24 GIF Display View

The GIF display view is used to display GIF images.

1. In General of the properties view, you can set its properties:
   Select the GIF file in the GIF selection box, and click the Set button to display the selected GIF image on the control. For details, see related instructions on the graphics view.

Name			Description	Remarks
General	Process	Tag	Tag to control a process	
		Value ON	Display the image when the tag reaches the set value	The default is 1, which cannot be changed
		Picture File	Sets the associated string tag to dynamically switch the images	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

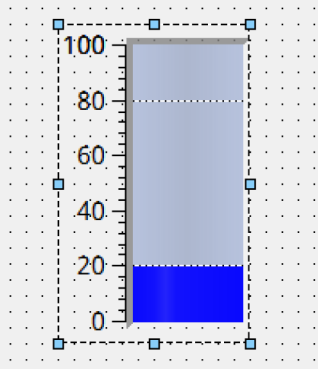
3. For animation properties, see *"7.4.1 Animation" on page 189*.

4. For the event property, see *"7.4.2 Event" on page 203*.

7.2 Enhanced controls

7.2.1 Bar

The bar control is used to monitor process values in a predefined range.



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Scale	Max	The maximum value of the scale	The value can be static or a tag. If you use the tag, the static setting will be disabled
		Min	The Minimal value of the scale	The value can be static or a tag. If you use the tag, the static setting will be disabled
	Process	Tag	The process tag to be displayed	
	Scale Colors & pos	BarBack color	The background color of the bar	
		Normal color	The color of normal process values	
		Scale pos	The position of tick marks: none, left, right, top and bottom	
		Major Scale Count	Number of divisions of the large scale	This item cannot be edited when Scale pos is set to No
		Minor Scale Count	Number of divisions in a large scale division	
		Decimal Digits	The decimal places of the scale values	
		Divisions	Number of divisions of the large scale	This item cannot be edited when Scale pos is set to No
		Inverted	Check this to reverse the direction of the scale	This item cannot be edited when Scale pos is set to No
		Display mark labels	Check this to display the scale text	This item cannot be edited when Scale pos is set to No

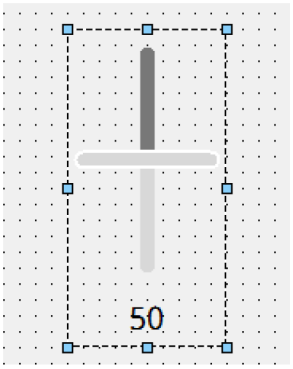
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appearance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Text	Font	Font	Text font	The font size is in pixels (px)
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Limits	Color	Color above upper limit	The color displayed when upper limit is exceeded	
		Color below lower limit	The color displayed when lower limit is exceeded	
	Settings	Display limit lines	Check this to display limit lines	
		Display limit markers	Check this to display limit markers	
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.2.2 Slider

The slider is used to monitor and adjust a process value within a defined range.



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Scale	Max	The maximum value of the slider	The value can be static or a tag. If you use the tag, the static setting will be disabled
		Min	The minimal value of the slider	The value can be static or a tag. If you use the tag, the static setting will be disabled
	Process	Tag	The process tag to be displayed	
	Ticks pos	Orientation	Direction of the progress bar: vertical or horizontal	
		Ticks pos	Position of the tick marks: NoTicks, TicksLeft, TicksRight, and TicksBothSides	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

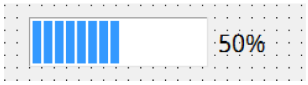
Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
	Slider	Slider color	The color of the slider thumb	
		SliderFrame color	The border color of the slider thumb	
		BarBack color	Background color of the slider	
		Bar color	Color of the slider	
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Text	Font	Font	Text font	The font size is in pixels (px)

Name			Description	Remarks
Limits	Color	Color above upper limit	The color displayed when upper limit is exceeded	
		Color below lower limit	The color displayed when lower limit is exceeded	
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authorization	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authorization	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.2.3 Progress Bar

The progress bar is used to display the proportional value of a process.



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Scale	Max	The maximum value of the progress bar	The value can be static or a tag. If you use the tag, the static setting will be disabled
		Min	The minimum value value of the progress bar	The value can be static or a tag. If you use the tag, the static setting will be disabled
	Process	Tag	The process tag to be displayed	
	Ticks pos	Orientation	Direction of the progress bar: vertical or horizontal	
		Text visible	Check this to show the percentage text	
		Inverted	Check this to reverse the direction of the progress bar	

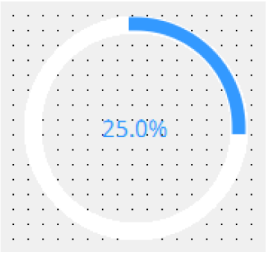
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Style-Sheet	Style-Sheet	StyleSheet	Sets the display style.	Five default styles are provided
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.4 Round Progress Bar

The round progress bar is used to display the proportional value of a process



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Scale	Max	The maximum value of the progress bar	The value can be static or a tag. If you use the tag, the static setting will be disabled
		Min	The minimum value value of the progress bar	The value can be static or a tag. If you use the tag, the static setting will be disabled
	Process	Tag	The process tag to be displayed	
	Ticks pos	Text visible	Check this to show the percentage text	
		Clockwise	Check this to reverse the direction of the progress bar	
		Outline pen width	The width of the bar	
		Data pen width	The width of the indicator	
		Decimal point	The number of decimal places of the percentage text	
		Style	The style of the percentage text	

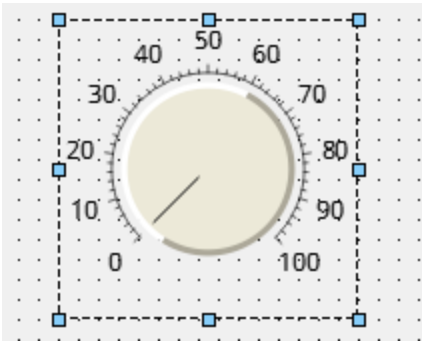
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Color	Text color	Text color	
		Background color	Fill color of the bar	
Layout	Position	X	Horizontal distance from the upper left corner of the bounding box to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the bounding box to the top edge of the screen	In pixels
	Size	Width	Width of the bounding box	In pixels
		Height	Height of the bounding box	In pixels
Text	Font	Font style	Font style	
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.5 Knob

The knob is used to set a process value.



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Scale	Max	The maximum value of the scale	–
		Min	The minimal value of the scale	–
	Process	Tag	The process tag to be displayed	–
	Color	Needle color	The color of the pointer	–
		Arc Left Color	The color of the left arc of the knob	–
		Arc Right Color	The color of the right arc of the knob	–

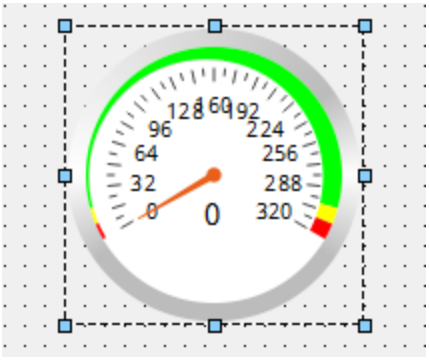
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Text	Font	Font	Text font	
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.6 Gage

The gage. displays the analog value of a process tag.



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	**General**	Label	The text displayed under the dial	
		Unit	The text displayed on the control to indicate the unit	
		Needle color	The color of the pointer	
		Scale color	The Color of the scale marks and the scale text	
	Fonts	Title	The font of the title	The title Includes the label, unit, and measured value
		Digit	The font of the scale text	
	Process	Tag	The process tag to be displayed	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

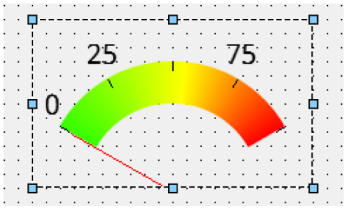
Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Scale	Range	MinValue	Min. value of the scale	
		lowError	The error lower limit	
		lowWarning	The warning lower limit	
		highWarning	The warning upper limit	
		highError	The error upper limit	
		Max	Max. value of the scale	
	Color	Color	The color of normal values	
		warningColor	The color of warning values	
		errorColor	The color of error values	
	Display	Display mark labels	Check this to display the scale text	

Name			Description	Remarks
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.
4. For the event property, see *"7.4.2 Event" on page 203*.

7.2.7 Meter

The meter displays the analog value of a process tag.



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Scale	Max	The maximum value of the scale	The value can be static or a tag. If you use the tag, the static setting will be disabled
		Min	The minimal value of the scale	The value can be static or a tag. If you use the tag, the static setting will be disabled
		Scale font	The font of the scale text	
		Display mark labels	Check this to display the scale text	
	Process	Tag	The process tag to be displayed	
	Color	Needle color	The color of the pointer	
		Scale color	The color of the scale text	
		Text	The text displayed at the center of the meter	

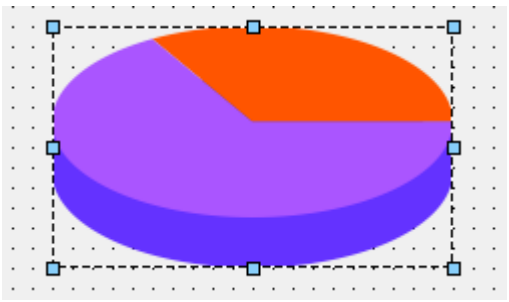
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Text	Font	Font	Text font	
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.8 3D-Pie

The 3D-PIe is used to display the proportion of multiple process tags.



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Data Display	Style	Sets the text to be displayed on the pie chart. Options: none, numeric value, percent	
		Channel number	Sets the number of slices of the pie chart	Ranges from 2 to 8

The Process property is used to set tags bound to the pie chart and channels;

You can click the + button to add channels. After adding a channel, you can set the channel name, the bound tag and the color.

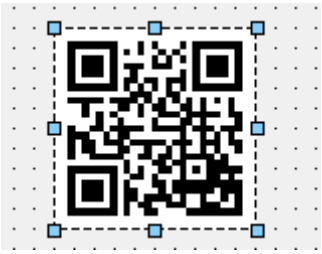
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.9 QR Code

The QR code represents a string.



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Settings	Level	Sets the calibration level to L, M, Q or H	The higher the level, the higher the tolerance
	Process	Tag	The process tag to be displayed	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.10 Bar Code

The bar code represents a number.

1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Settings	Code system	Bar code symbology	Code 128 is supported
		Hide text	Check this to hide the number text	
	Process	Tag	The process tag to be displayed	

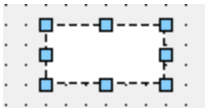
2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Color	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the line is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance of the line	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.11 Canvas

Canvas is used to display graphics drawn with scripts.



1. In the Properties dialog box in the Properties view, you can set the corresponding properties.

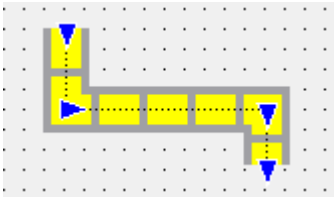
Name			Description	Remarks
Appear-ance	Color	Border color	Sets the border color of the bounding rectangle	
		Fill color	Fill color of the rectangle	
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
	Border	Width	Width of the border line	
		Style	There are five styles:	The default is solid
			Solid, dash, dot, dash-dot and dash-dot-dot	
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels

Name			Description	Remarks
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

2. For the event property, see *"7.4.2 Event" on page 203*.

7.2.12 Flow Block

The flow block is used to dynamically display the flow of fluid. The blue arrows indicate the direction of flow:



1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Flow	Reverse	Check this to reverse the flow direction	
		Dynamic Speed	Check this to enable dynamic speed. When it is not checked, You can set static speed	
	Process	Tag	The process tag for flow velocity	This setting is visible when Dynamic Speed is checked

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name				Description	Remarks
Appear-ance	Conduit	Background		Check this to show the background	
		Width		The width of the conduit	
		Color		The background color	This setting is visible when Background is checked
		Border		Check this to show the border	When Border is checked, Background will be also checked
		Border	Width	Width of the border	
			Color	Color of the border	
	Slider	Width		The width of the flow block	
		Length		The length of the flow block	
		Interval		The interval of between two flow blocks	
		Color		The color of the flow block	
Misc	Misc	Layer		The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value		z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity		Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.13 Alarm Bar

The alarm bar is used to display the active alarm.

1. In General of the properties view, you can set its properties:

Name			Description	Remarks
General	Display	Move Speed	The speed at which the alarm text moves, in pixels	
		Move Direction	The scrolling direction of the alarm text: from right to left and from left to right	
		Alarm Order	The alarm display order: time sequence and time reversal	
		Time	Check this to show the time in the alarm text	
		Date	Check this to show the date in the alarm text	
		Message	Check this to show message in the alarm text	
		Auto hide	Check this to display the alarm bar only when there is alarm information	
		Errors	Check this to display error type alarm text	
		Warnings	Check this to display warning type alarm text	
		System	Check this to display system type alarm text	

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Text color	Text color	
		Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
		Fill style	Provides a variety of fill styles	Click the drop-down list to browse the styles
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Flashing	Runtime Appearance	Flashing	You can choose "standard" or "none".	This setting is only effective when no process tag is selected.
	Process	Tag	Process tag	Process tag, flashes when the value is 1
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

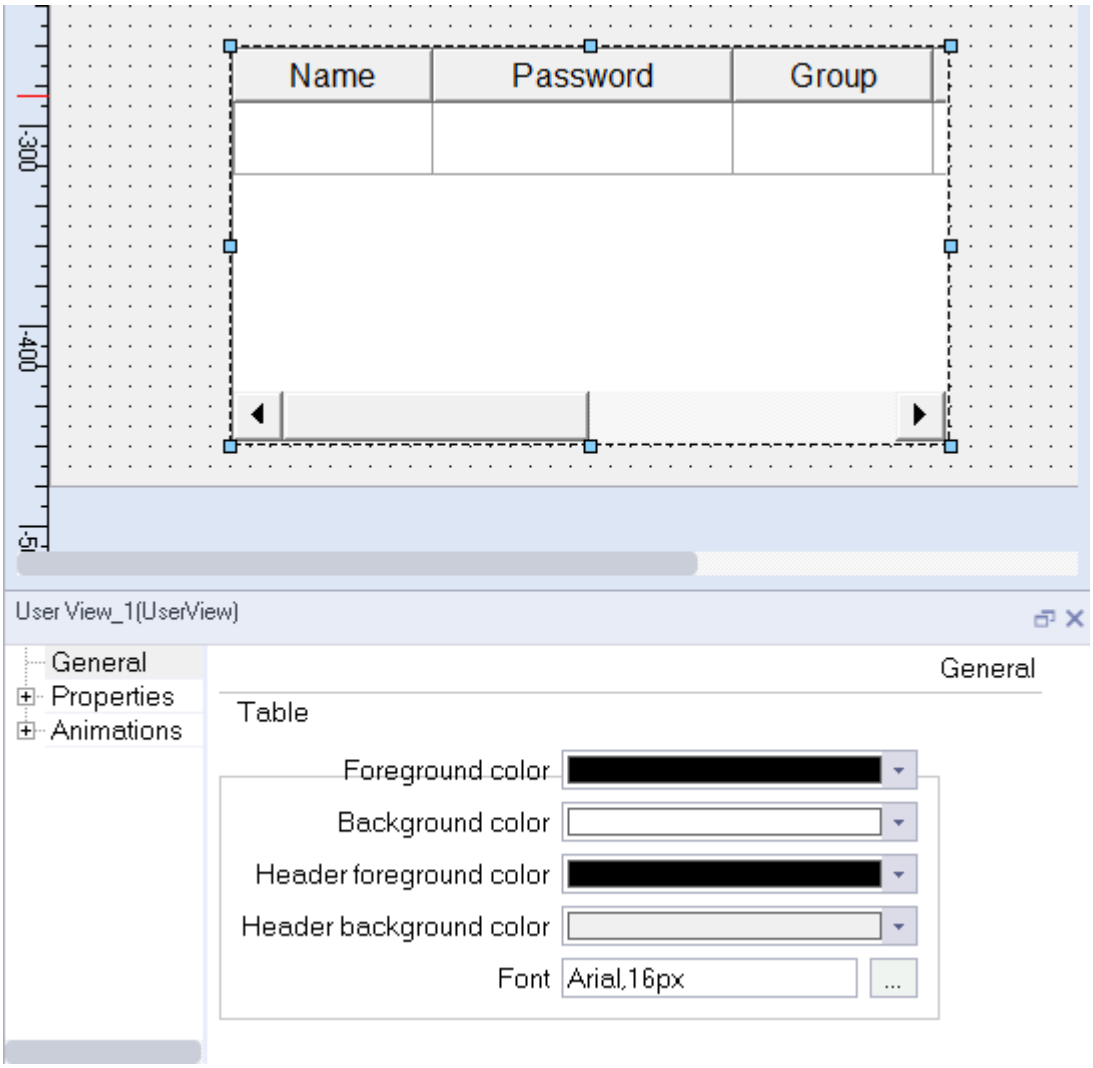
3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.14 User View

The User view is used to manage users on the runtime.

Configuring the user view

1. Adding a User view to the screen
   As shown in the following figure, click "Enhanced Controls-> User View" in the toolbox on the right and drag it to the screen.



2. Property setting
   The properties of the User view include general, properties and animations. For animation setting, see *"7.4.1 Animation" on page 189*.

General setting

Property	Description
Foreground color	The foreground color of the user view table
Background color	Background color of the user view table

Property	Description
Header foreground color	The foreground color of the user view table header
Header background color	The background color of the user view table header
Font	The style and size of table user view table text

Properties setting

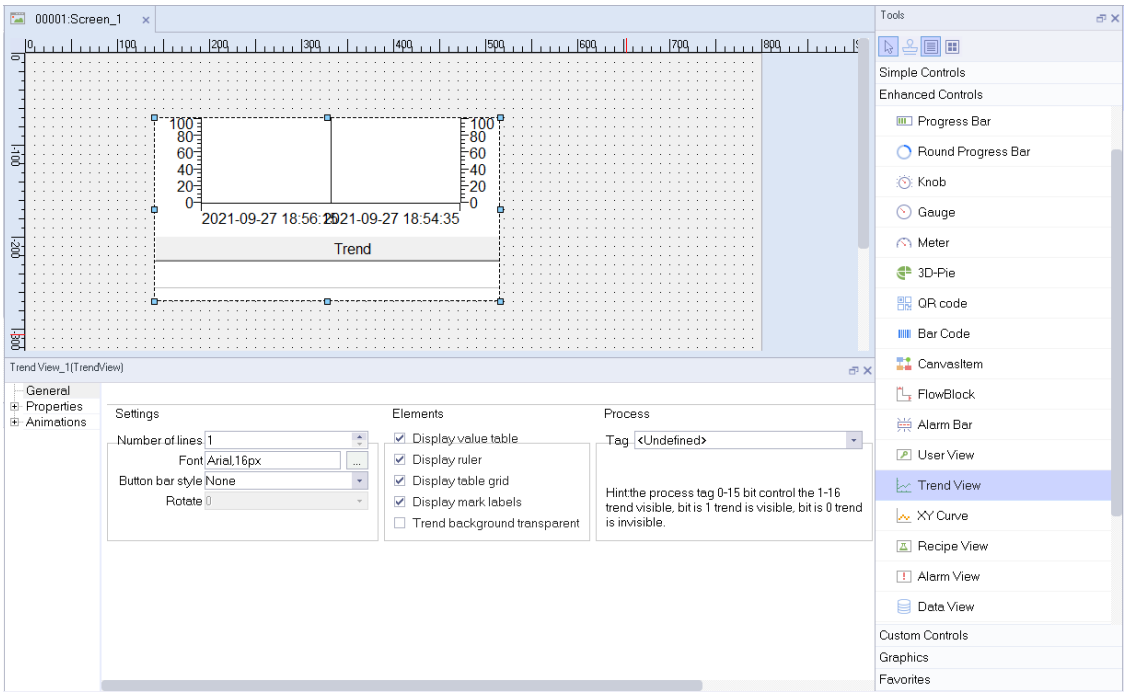
Name			Description	Remarks
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

7.2.15 Trend View

The trend view is a curve management control that displays the data collected in the current process or recorded in the data record in a curve graph.

Configuring a trend view

1. Add a trend view to the screen.
   As shown in the following figure, click Enhanced Controls→Trend View in the toolbox on the right and drag it to the screen.



2. Set its properties.

   The properties of the trend view include general, properties and animations. For animation setting, see *"7.4.1 Animation" on page 189*.

General setting

Property	Description
Number of lines	The number of rows of the value table, which is only effective when Display value table is selected.
Font	style and size of the scale text.
Button bar style	Displays or hides buttons. For details, see Description of buttons in the following section.
Rotate	Rotates the trend view control for a certain angle. The control is rotated after you download the project to the target device. When you have selected Display value table, this setting is unavailable.
Display value table	Check this to display a value table.
Display ruler	The ruler is a vertical line in the trend view. When the ruler shows, the value of the point where the trend curve intersects with the ruler is displayed in the value table, and when the ruler does not show, the current value is displayed instead.
Ruler movement	You can use system functions or the ruler moving button or click a point on the trend view to move the ruler.
Display table grid	Check this to display the grid lines of the value table.
Display mark labels	Check this to show the scale value text.
Trend background transparent	Check this to make the background of the graph transparent.
Process tag	Visibility of curves 1–16 is controlled by bits 0-15 of the process tag. The curve is visible when its corresponding bit is 1 and invisible when the bit is 0.
Ruler Tag	You can use the tag to control the position and width of the ruler and enables/disables movement of the ruler.

Description of buttons

Button	Function
	Pause trend update
	Start trend update
	Show current trend
	Move the trend graph to the left half of the display width
	Move the trend graph to the right half of the display width
	Zoom in the trend curve
	Zoom out the trend curve
	Move the ruler to the left
	Move the ruler to the right

Properties setting

In Properties, you can set appearance, layout, X axis, left value axis, right value axis, axis, table, trend and misc settings. The layout and misc settings are general for all controls and you can refer to the settings of the user view. For the other settings, see the following tables:

Appearance

Attribute	Description
Background color	Background color of the view
Ruler color	Ruler color
Axis color	Axis color

X axis color

Property	Description
Mode	Modes include: Time: The time interval between every two sampling points. When selecting the time mode, you need to set the time interval. Points: With this mode, the configured points are used as coordinates. Tag/constant: the range scale value of the axis. When selecting this mode, you need to set "Axis begin", "Axis end" and "Number of points".
New values from	Sets where the data is from
Time Display Format	Sets how time is displayed on the view. Available only for the time mode.
Start Time	Sets the time for the sampling start point. Available only for the time mode.

Property	Description
Oscillograph display	When the control tags of oscilloscope display are not configured, check this to display the trend according to oscilloscope display mode. Available only for the time mode. The source of the new value is fixed to the right. When the first page is displayed, the time axis remains unchanged, and the new value is drawn from left to right in turn. After drawing a full screen, the whole curve moves to the left
Axis begin	Start value of the X axis. Only for the tag/constant mode.
Axis end	End value of the X axis. Only for the tag/constant mode.
Time Interval (Sec.)	Available only for the time mode
Number of points	Only for the points and tag/constant modes. The number of points on the X axis.
Tag	The control tag for oscilloscope display. The trend is displayed by oscilloscope if the tag value is 1.

Left Value Axis

Property	Description
Axis begin	Beginning of the left axis (Y1, Y3), which can be a tag or constant. If the tag value is greater than the setpoint of axis end, the axis end value will be changed to tag value + 1.
Axis end	End of the left axis (Y1, Y3), which can be a tag or constant. If the tag value is less than setpoint of axis begin, the axis begin value will be changed to tag value - 1.

Right Value Axis

Property	Description
Axis begin	Beginning of the right axis (Y2, Y4), which can be a tag or constant. If the tag value is greater than the setpoint of axis end, the axis end value will be changed to tag value + 1.
Axis end	End of the right axis (Y2, Y4), which can be a tag or constant. If the tag value is less than setpoint of axis begin, the axis begin value will be changed to tag value - 1.

Axis

Property	Description
Axis label	Check this to display the corresponding axis
Increments	The interval between scale marks
Marks	The number of marks on the axis
Mark line	Check this to add mark lines on the axis
Lines count	The number of mark lines on the axis. Only available when "Mark line" is checked.

Table

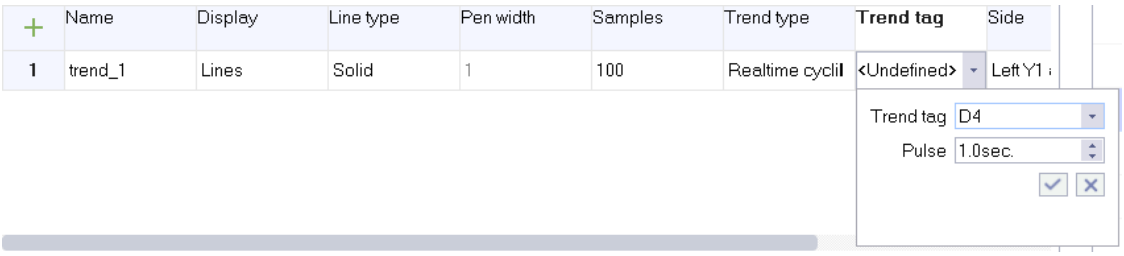
Property	Description
Foreground color	The foreground color of the table
Background color	The background color of the table
Header foreground color	The foreground color of the table header
Header background color	The background color of the table header
Font	Style and size of table text

Trend

Property	Description
Name	The name of the trend curve.
Display	The display style of the curve. Options include Bars, Steps, Point, Lines and X Area
Line Type	Available only for a line curve.
Pen Width	Available only for line and bar curves.
Samples	Maximum number of trend curve sampling points.
Trend Type	The trend types include real-time cyclic trigger, real-time bit trigger, buffer bit trigger, and log.
Trend Tag	The tag associated with the trend. The function of the tag depends on the trend type.
Side	Sets which scale prevails
Foreground color	The color of the curve

Description of trend types

1. Real-time cyclic triggered

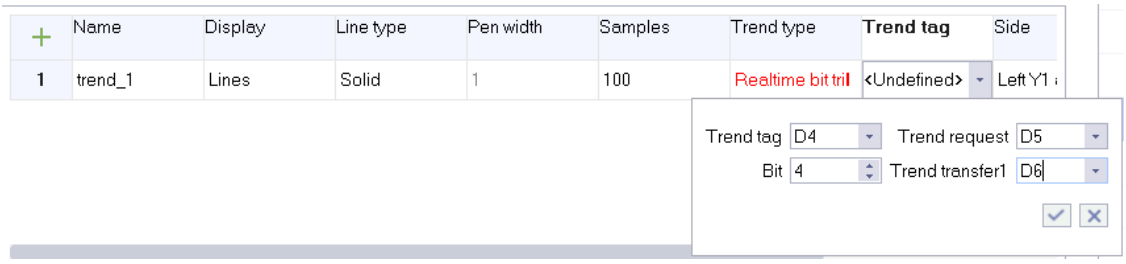


The trend collects the values of the specified tag in a fixed period and displays them in real time.

Specified tag: refers to the tag set in the Trend Tag column.

Period: It is determined by the pulse of the trend tag, and the minimum unit is 0.1s. The shorter the period, the higher the occupancy rate of the CPU. It is not recommended to use a large number of real-time sampling with short periods in a project.

2. Real-time bit triggered



It is for the point-type X axis.

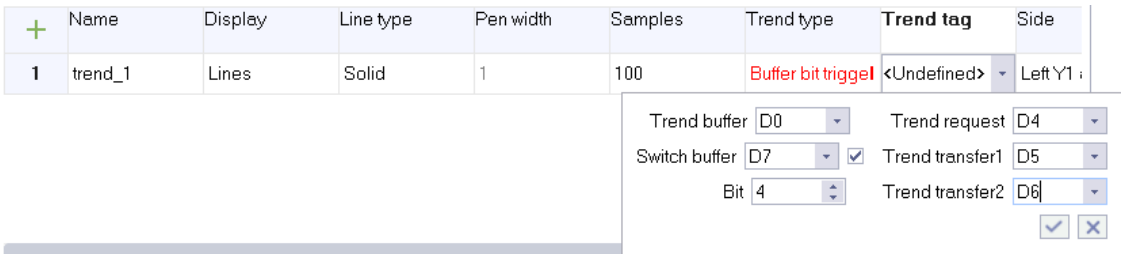
The trend samples the values of the specified tag according to the change of the trend send tag bit and displays them in real time.

Trend send tag: refers to the tag specified in trend send area 1. Trend sampling occurs only when the highest bit and trigger bit of the trend send tag are both 1. After sampling, the highest bit and trigger bit are set to zero.

Trend request: auxiliary tag, enters the screen containing this trend, and sets the trigger bit; Leaves the screen and resets the trigger bit.

Trigger bit: The trigger bit of the trend send tag and the trigger bit of the trend request are both specified by the "bit" in the trend tag.

3. Buffer bit triggered



It is for the point-type X axis.

The trend samples the values of the specified array according to the change of the trend send tag bit and displays them in a trend curve.

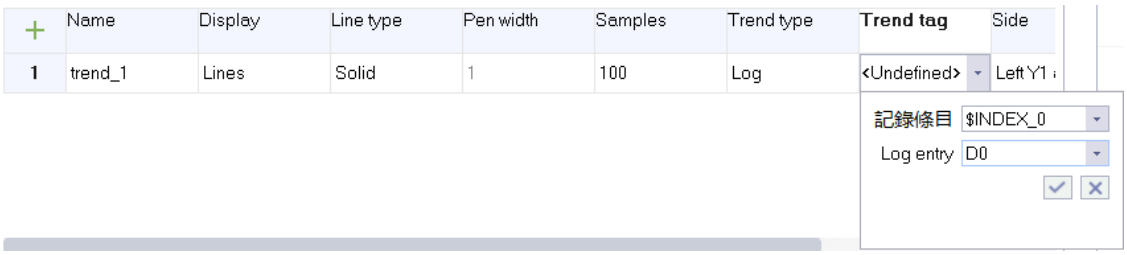
Trend send tag: refers to the tag specified in trend send area 1. Trend sampling occurs only when the highest bit and trigger bit of the trend send tag are both 1. After sampling, the highest bit and trigger bit are set to zero.

Trend request: auxiliary tag, enters the screen containing this trend, and sets the trigger bit; Leaves the screen and resets the trigger bit.

Trigger bit: The trigger bit of the trend send tag and the trigger bit of the trend request are both specified by the "bit" in the trend tag.

Switch buffer: If you check this, the trend array corresponds to the array specified in the switch buffer, and the trend send tag corresponds to the tag specified in trend send area 2.

4. Log



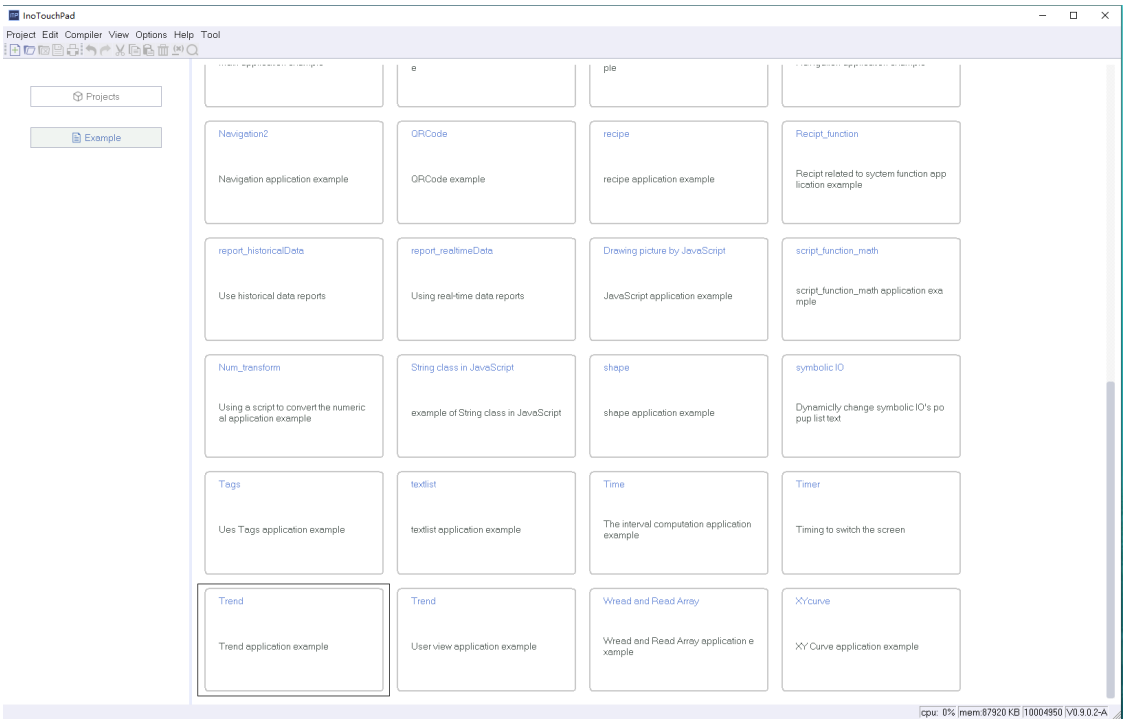
It is for the time-type X axis.

The trend is used to display the logged values of the specified tag in a data record for a certain period of time.

Data log: selected from the logs in the trend tag.

Specified tag: selected from the log entries in the trend tag. As shown in the above figure, you can only select the tag that logs data in data log_1.

For details, see the example project "Trend View".



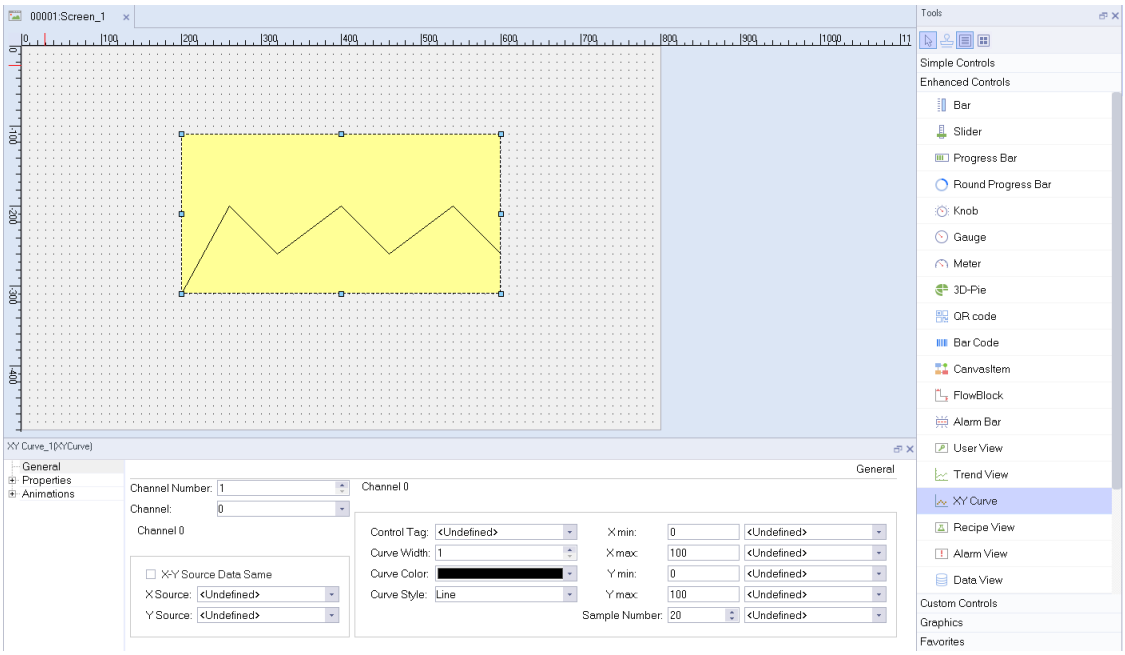
7.2.16 XY Curve

If you take the data in two groups or a group of continuous registers as coordinates on X and Y axes respectively, the points/figure formed by these coordinates generate an XY curve which cannot be edited or zoomed.

Configuring an XY curve

1. Add an XY curve to the screen.

   As shown in the following figure, click Enhanced Controls→XY Curve in the toolbox on the right and drag it to the screen.



2. Property setting.
   The properties of the XY Curve include General, Properties and Animations. For animation setting, see *"7.4.1 Animation" on page 189*.

   General setting

   Channel Number: the number of curves contained in the control, which can be 1–15.

   Channel: Displays the current curve number, which can be switched.

   X-Y Source Source Data Same: If you check this, you only need to configure the data source address for the X axis.

   X Source: the X axis data source address of the points on the curve. It can only be an array tag, with address types: Int16, UInt16, Int32, UInt32, Float, and Double.

   Y Source: the Y axis data source address of the points on the curve. It can only be an array tag, with address types: Int16, UInt16, Int32, UInt32, Float, and Double.

   Control Tag: The curve display control address. When the address value changes to 1, the curve is generated, and when it changes to 2, the curve is cleared. Address types: Int16, UInt16, Int32, UInt32.

   Curve Width: The line width of the curve.

   Curve Color: The line color of the curve.

   Curve Style: The style of the curve, including Points, Line (default), Projection on to X axis and Projection on to Y axis.

   X min: The lower limit of the X axis. Can be a constant or tag. The tag address types are Int16, UInt16, Int32, UInt32, Float, Double.

   X max: The upper limit of the X axis. Can be a constant or tag. The tag address types are Int16, UInt16, Int32, UInt32, Float, Double.

   X min: The lower limit of the Y axis. Can be a constant or tag. The tag address types are Int16, UInt16, Int32, UInt32, Float, Double.

   X max: The upper limit of the Y axis. Can be a constant or tag. The tag address types are Int16, UInt16, Int32, UInt32, Float, Double.

   Sample Number: The points of the curve (0–1000), and the default is 20. Can be a constant or tag. The tag address types are Int16, UInt16, Int32, and UInt32.

   Properties setting: This setting includes Appearance, Guides, Layout and Misc. For Layout and Misc, see the related description for the user view. For the other settings, see the following tables:

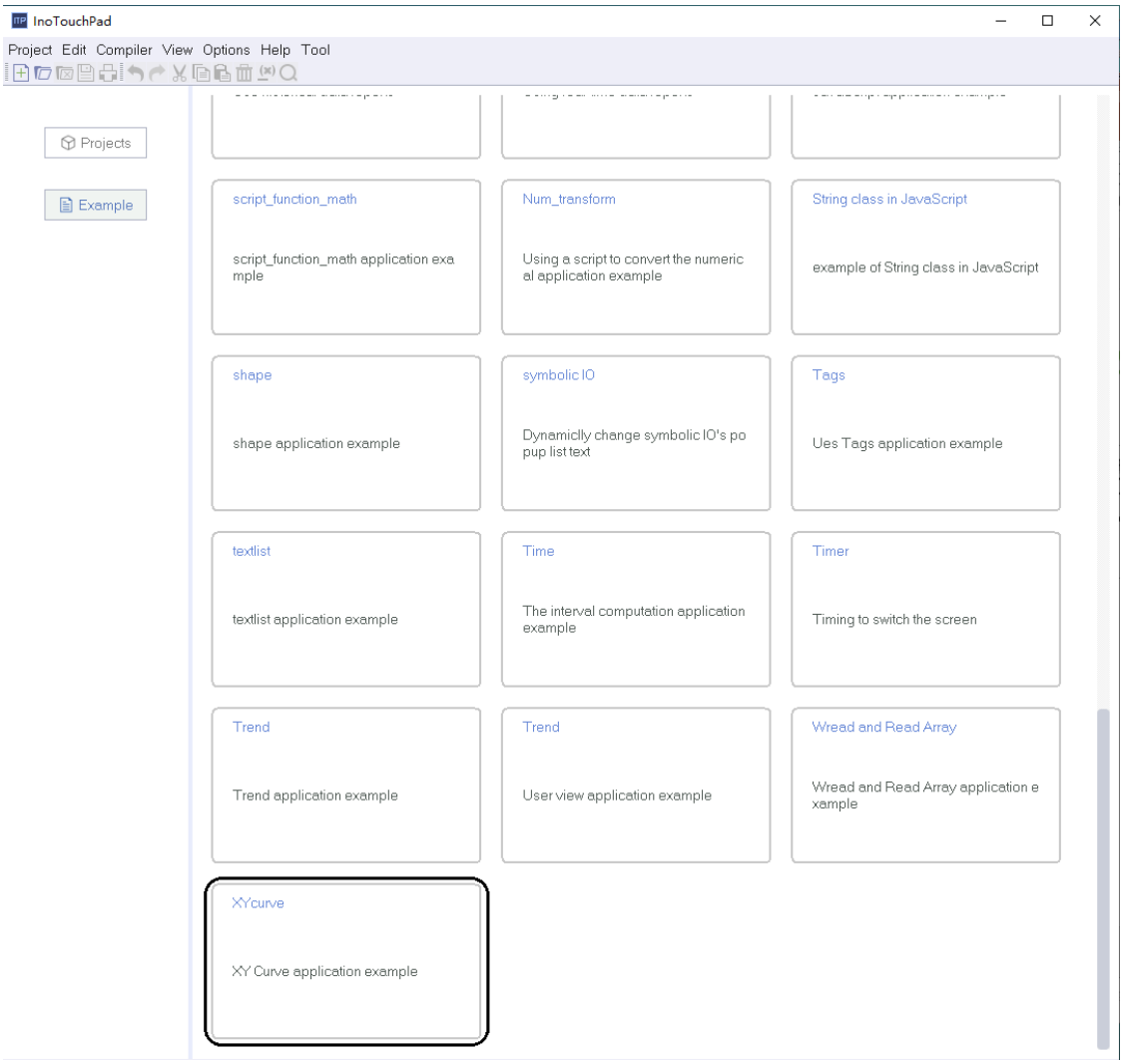
   Appearance

Attribute	Description
Text color	Text color
Background color	Background color of the curve
Fill style	Fill style of the curve background

   Guides:

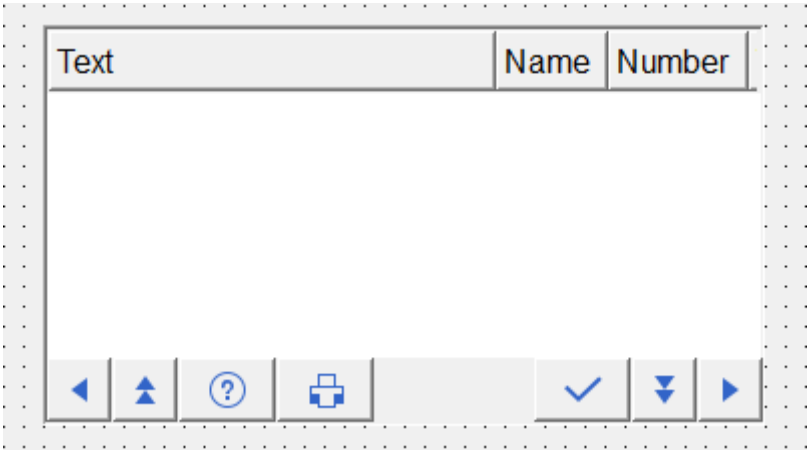
Attribute	Description
X Guides	The number of vertical guides, which defaults to 0 and ranges from 0 to 99
Y Guides	The number of horizontal guides, which defaults to 0 and ranges from 0 to 99
Guides Color	Guides Color

3. For details, see the example project "XY Curve ".



7.2.17 Alarm View

The alarm window is used to dynamically display the specified information in the runtime. It is always the topmost object on the screen and you can close it any time.



Description of buttons

Button	Function
	Move the table to the left. You can use this button to scroll the table to view information when the display is out of the view port
	Page up
	Click this button to display help information and other configured text
	Printing alarm
	Alarm acknowledgment
	Page up
	Move the table to the right

1. In General of the properties view, you can set its properties:
   You can set the display properties in General. Display properties are used to define the categories and display mode of all displayed alarms in the Alarm view.

   The alarms categories include "Error", "Warning" and "System".

   The following table describes how the alarm view displays the alarm with different display modes and alarm categories:

Name		Description
Alarm	Pending alarms	Displays an activated but not deactivated and acknowledged alarm. Displays activated alarms in all categories, and activated and acknowledged (at the same time) alarms in the error category
	Unacknowl-edged alarms	Displays activated alarms, and activated and deactivated alarms in the error category
Alarms events		Displays the message events Displays each alarm. There are "activated", "(activated) acknowledged" (deactivated after activated), "activated and acknowledged" (simultaneously), "(activated and acknowledged) deactivated" (deactivated after activated and acknowledged) and "(activated and deactivated) acknowledged".
Alarms log		You must only select an alarm log among configured alarm logs
Show grid		Check this to display grid lines in the view
Display by date		This option is available when you check "Alarm log". Check this to display the alarm information by date
Auto Display		Only available if you do not check Alarm Log. If you check this option, t will be automatically displayed in the template
Background transparent		Check this to make the background transparent
Default Style		Check this to apply the default style to the selected row

2. In the Properties dialog box in the Properties view, you can set the corresponding properties.

Name			Description	Remarks
Appear-ance	Fill	Background color	Fill color of the rectangle	It only works when the blinking function is configured, blinking between Line Color and Fill Color
Layout	Position	X	Horizontal distance from the upper left corner of the rectangle to the left edge of the screen	In pixels
		Y	Vertical distance from the upper left corner of the rectangle to the top edge of the screen	In pixels
	Size	Width	Width of the rectangle	In pixels
		Height	Height of the rectangle	In pixels
Text	Fonts	Fonts	Text font	

Name			Description	Remarks
Columns	Visible columns	Name	Check this to display the alarm category	
		Number	Check this to display the alarm number	
		Date Time	Check this to display time	
		Date	Check this to display date	
		Status	Check this to display the state	
		GR	Check this to display the ack group	
		PLC	Check this to use a PLC	
	Column properties	Headings	Check this to display the headings	
		Column ordering	Check this to enable column ordering	
		Check this to sort by date/time	Check this to sort by date/time	
		Text across columns	Check this to display text across columns	
		Time in milliseconds	Check this to display time in milliseconds	
	Sort	Oldest alarm first	Displays the oldest alarm first	
		Newest alarm first	Displays the newest alarm first	
Misc	Misc	Layer	The layer in which the object is currently located	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Display clearance	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque
Security	Runtime Security	Authoriza-tion	Assigns users with permissions	
	Operation	Enabled	Enables the security	You must check Enabled to enable input
				Otherwise, input is disabled
		Visibility controlled by authoriza-tion	Check this to control visibility through authorization	If this is checked, only authorized users can see the control

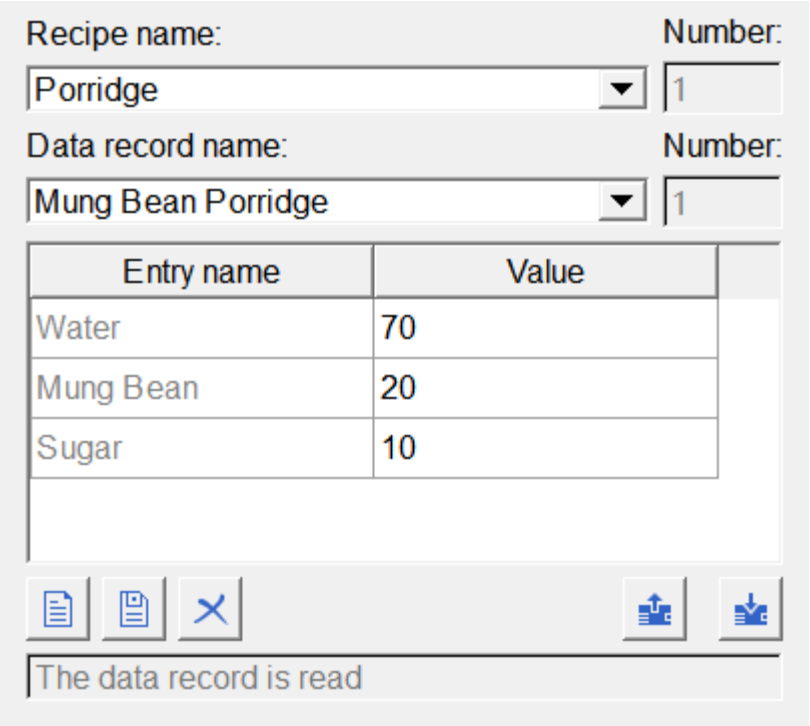
3. For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.18 Recipe view

The Recipe view is a general-purpose enhanced screen object for displaying, editing and managing recipe data records. It presents data records in tables. In the recipe view, the values displayed and entered can be stored in data records, which exchange data with the PLC through system functions.

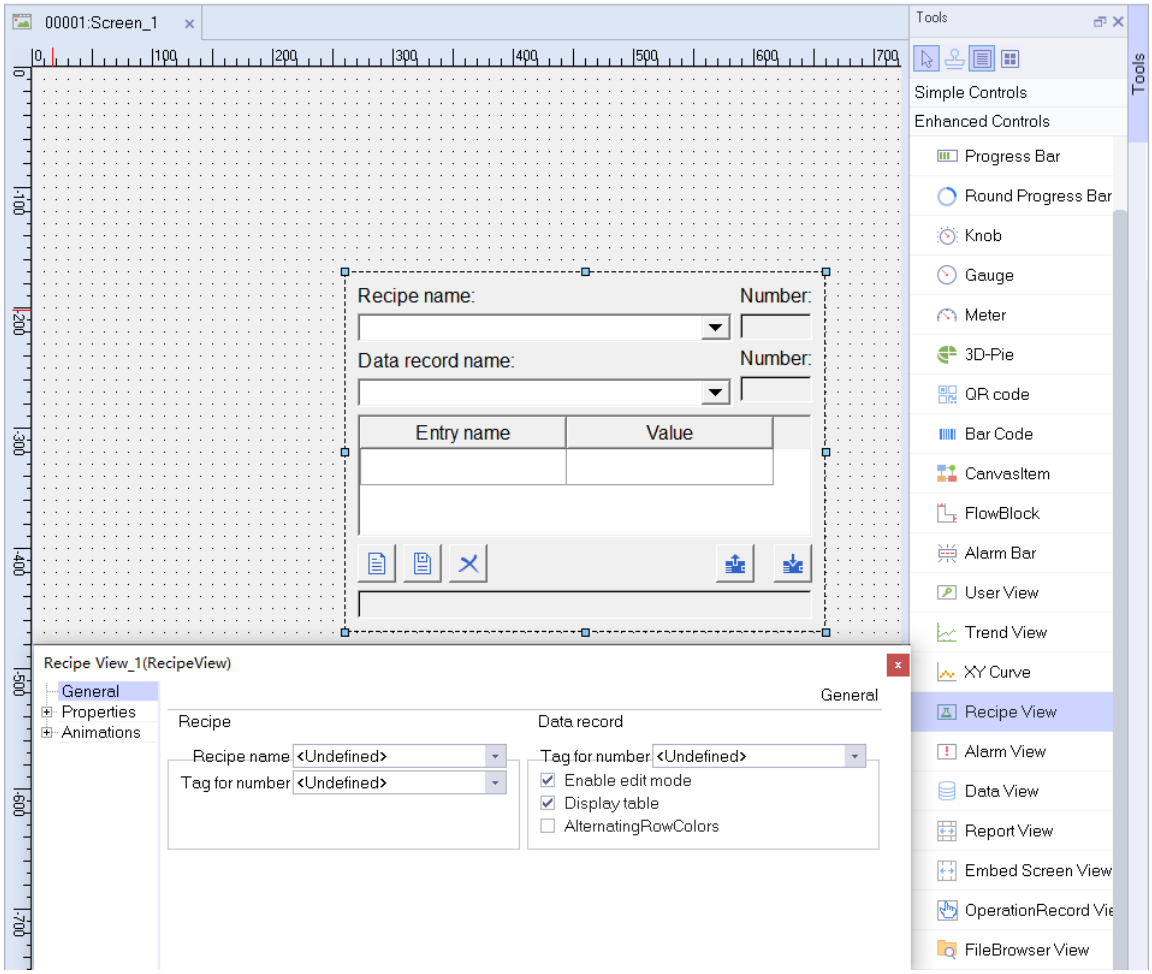
The recipe view mainly consists of five modules, namely:

1. Recipe list
2. Data record list
3. Component table
4. Button
5. Operation prompt box



Configuration of Recipe View

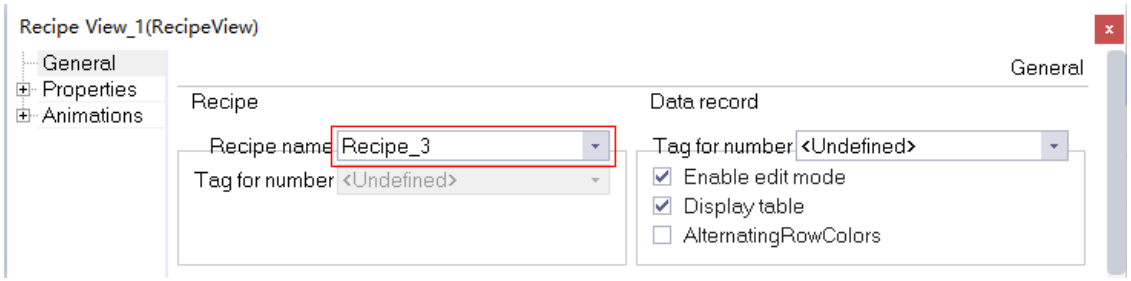
Select Recipe View in the screen to configure the recipe view in the Recipe View Properties window.



General setting

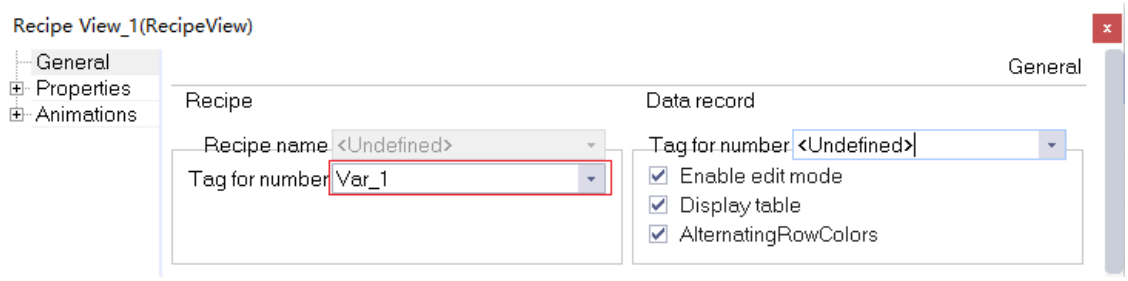
1. Displays a recipe by the name.

   The user can specify the recipe name to be displayed in the runtime through Recipe name in the general setting column. After setting the recipe name, the tag option for numbering will be disabled, and only one way can be specified.



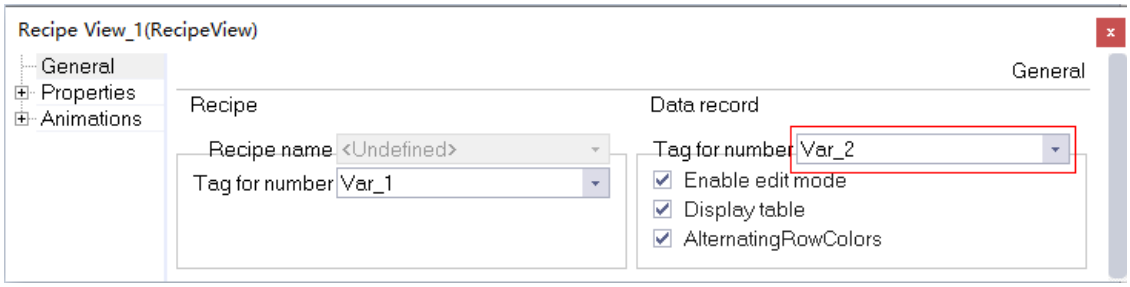
2. Displays a recipe by the number.

   The user can also set specific tags in the tags used for numbering the recipe, and display the recipe corresponding to the numbered recipe through the tag value. When the tag value is a nonexistent recipe number, the recipe view is displayed as empty.



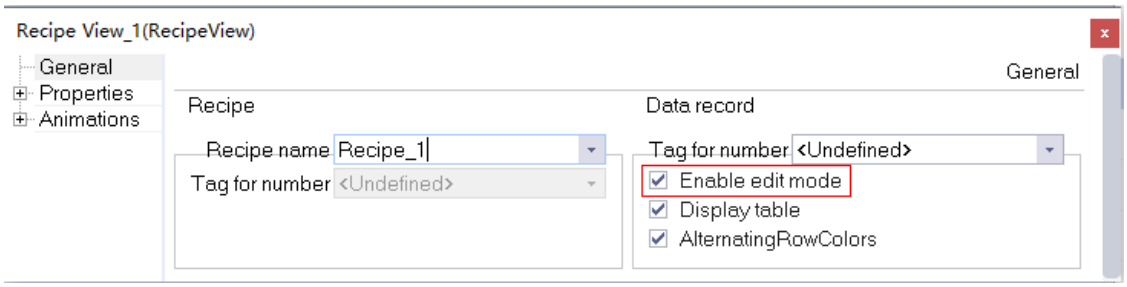
3. Displays recipe data record by the number.

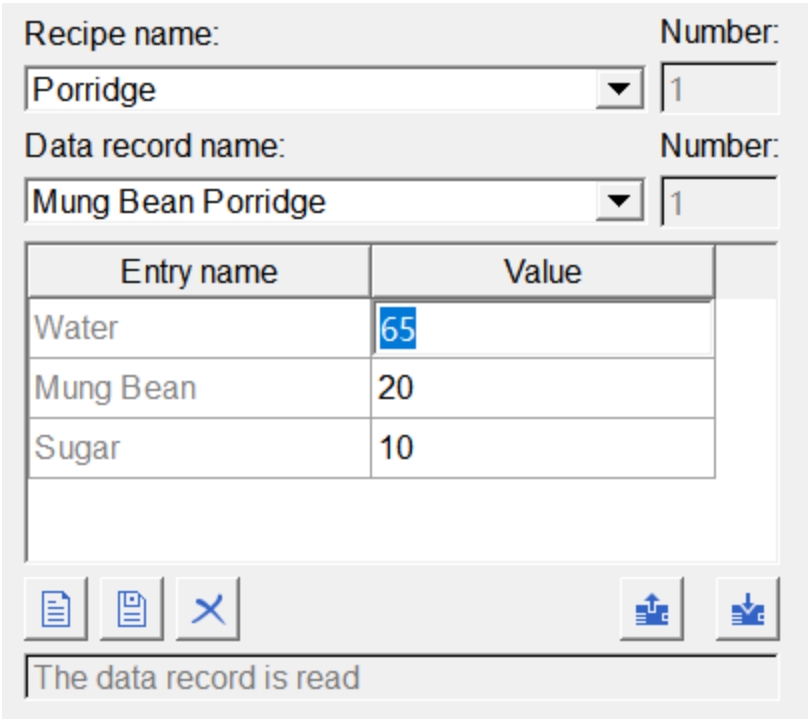
   The specified tag is set in the tag for numbering of the data record, and the recipe data record corresponding to the number is displayed by the tag value. When the tag value is a nonexistent recipe data record number, the data record and the ingredient table of the recipe view are displayed as empty.

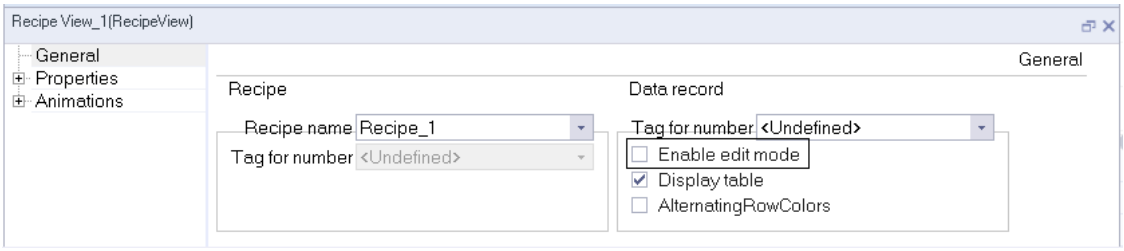


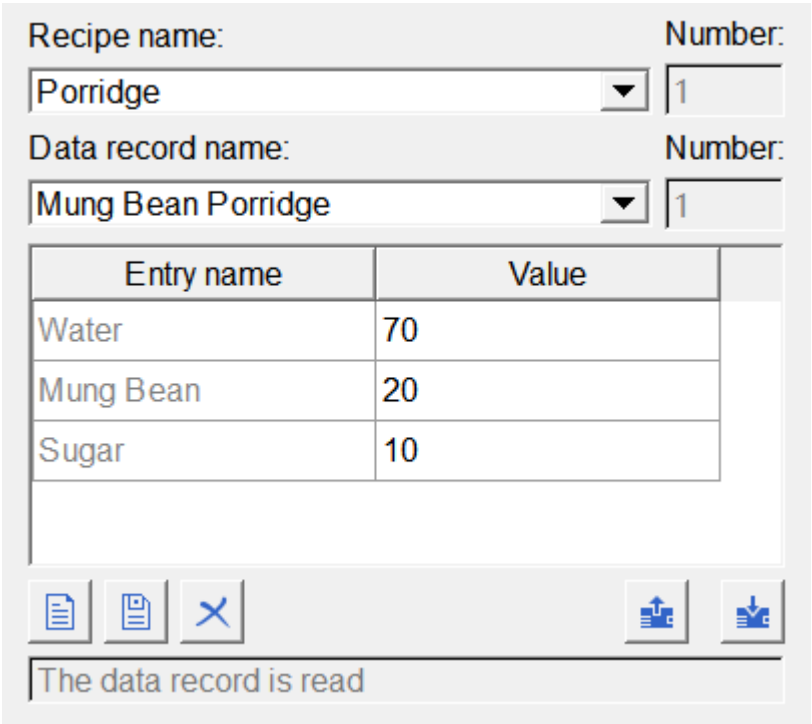
4. Enable edit mode.

   "Enable edit mode" is checked by default and therefore you can modify data records. If you only wants to view recipe records but does not want to modify them, do not check this option.



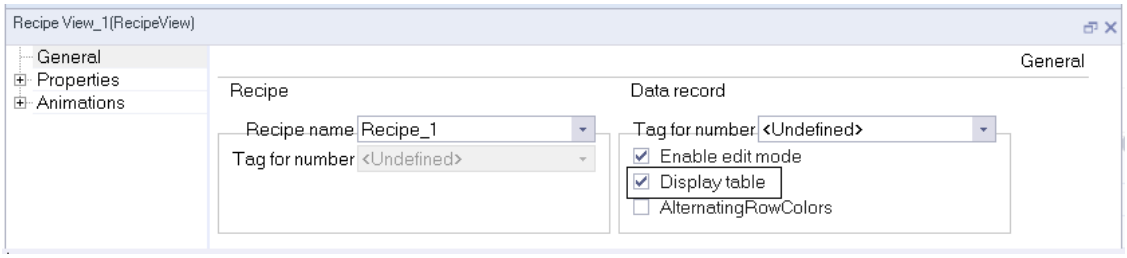


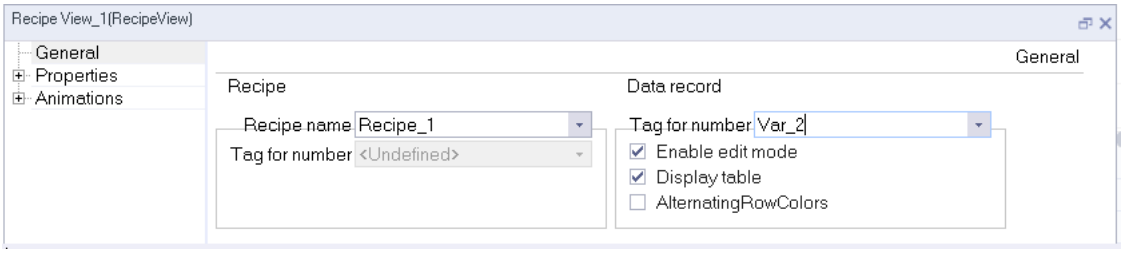


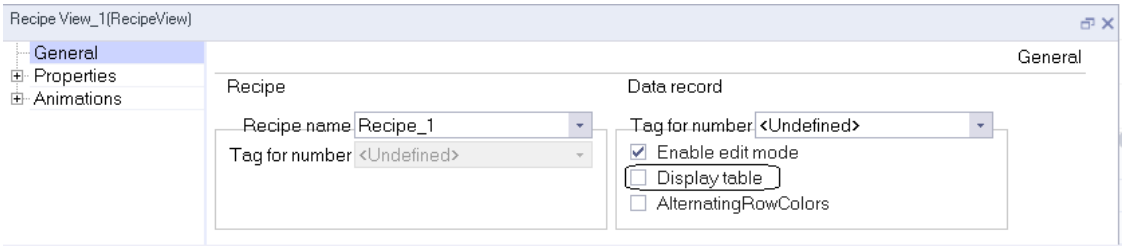


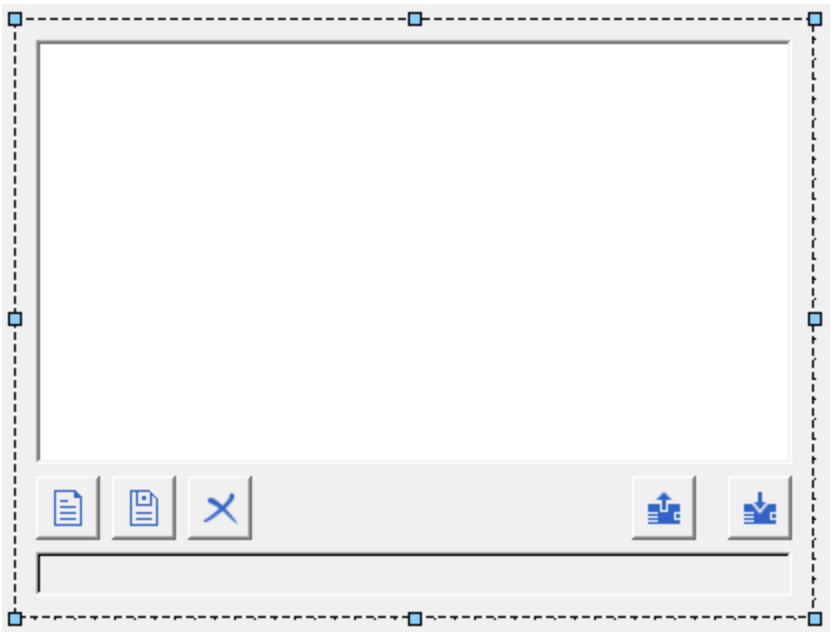
5. Display table.

If you check this option, the recipe view will be displayed as a table, otherwise, it will be displayed as a simple view.









During operation, the simple view directly displays all data records of the recipe selected by the user, and the user can directly select it for operation.

6. Alternating row colors.
   If you check this, the rows of the table will be displayed alternately with the background color, as shown in the following figure.

Entry name	Value
Water	0
Mung Bean	0
Sugar	0

Property setting

Name			Description
Appearance	Fill	Text color	The fill color of the text, excluding the data table area
		Background color	The background color, excluding the data table area
Layout	Position	X	Horizontal distance, in pixels, from the origin of recipe view to the left edge
		Y	Horizontal distance, in pixels, from the origin of recipe view to the upper edge
	Size	Width	The width of the recipe view, in pixels
		Height	The height of the recipe view, in pixels
Text	Font		Defines the style and size of the recipe view text font, default is Droid Sans Fallback, font size 12.
Buttons	General commands/menu entries		You can select the buttons to display
Table	Table	Foreground color	The color of table text
		Background color	The background color of the table
		Header foreground color	The color of the table header text
		Header background color	Header background color
Security	Runtime Security	Authorization	In the recipe view, the authorization of the operator control
	Operation	Enabled	Is the recipe view can be used in the runtime
			Checked: Yes
			Unchecked: No
		Visibility controlled by authorization	Is the recipe view visible under the current user permission
			Checked: only visible under current permission
			Unchecked: always visible
Miscellane-ous	Miscellaneous	Layer	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

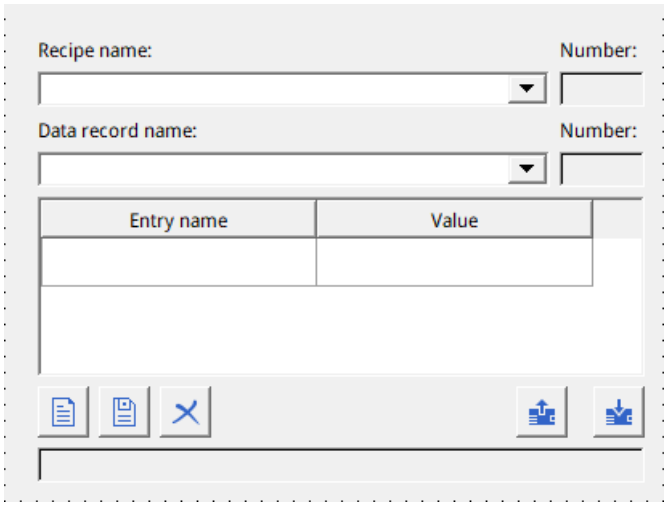
For animation properties, see *"7.4.1 Animation" on page 189*.

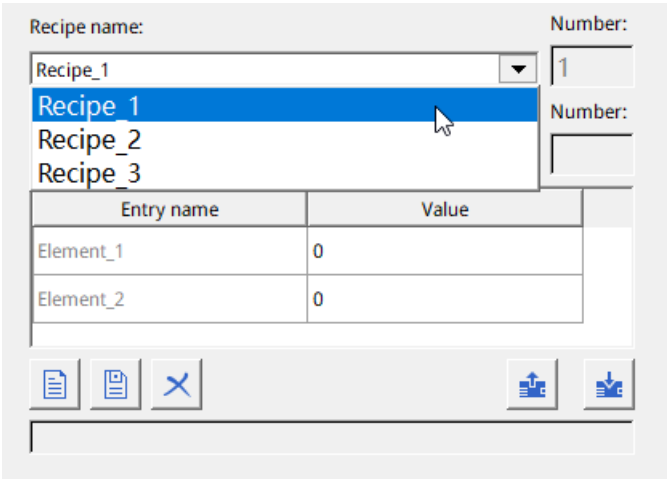
2 Features of runtime recipe view

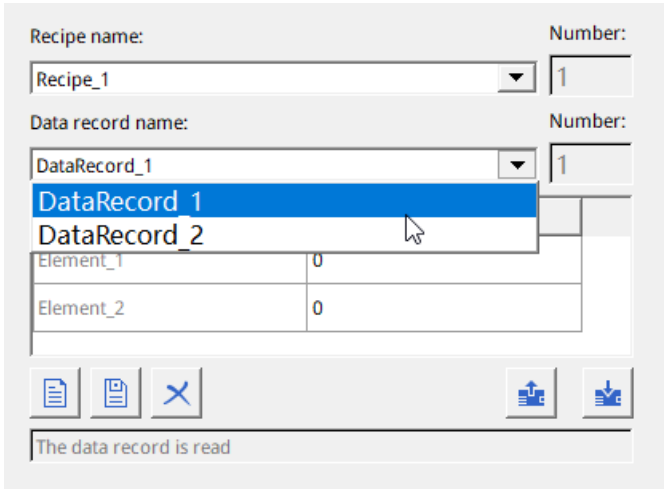
Initial display

In the runtime, the recipe view appears in the following initial displays:

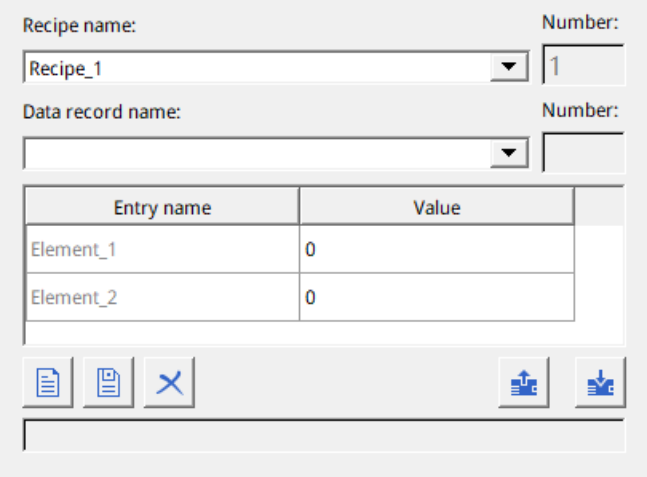
1. When the user does not configure any recipes for the recipe view, the initial display is empty, but the ingredient data that the user needs to view and modify can be displayed by selecting the recipe name and recipe record.



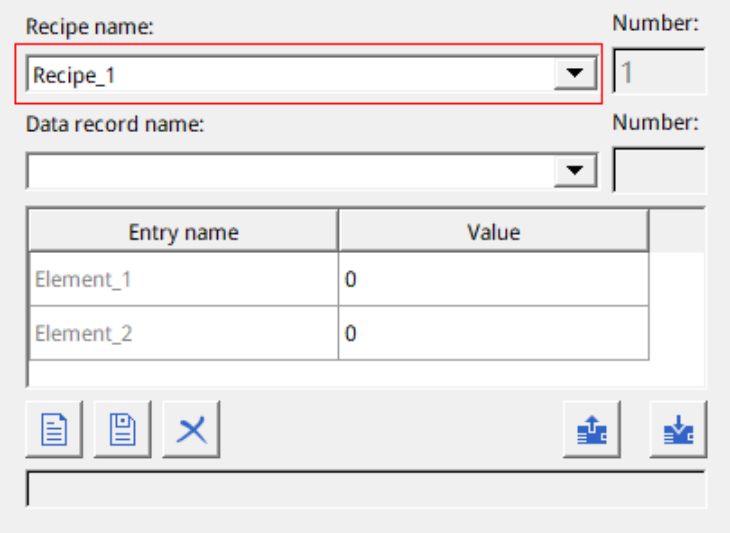




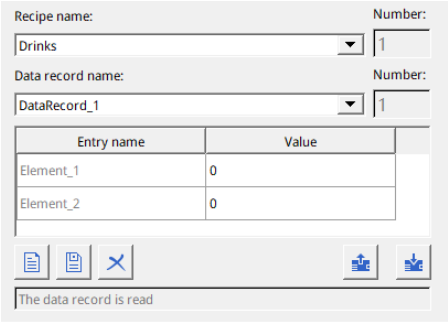
2. The user configures the recipe through the recipe name to the recipe view. Without the configuration data record, the configuration recipe and the default score ratio of the recipe are initially displayed. At this time, the recipe view can also display the ingredient data that the user needs to view and modify by selecting the recipe name and recipe record.

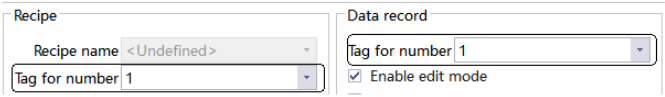


3. The user configures a recipe for the recipe view through the tag used for numbering. Without the configuration data record, the recipe of the number corresponding to the tag value is initially displayed and the default score ratio of the recipe. At this time, you cannot select any other recipe to display on the recipe view, but can only display the recipe by changing the tag value.



4. The user configures a recipe for the recipe view through the tag for numbering, and configures data record through the tag for numbering, and the ratio of the recipe with the number corresponding to the tag value and the data record ingredient value corresponding to the number corresponding to the recipe is initially displayed. At this time, other recipes can only be displayed by changing the tag value, while the data record can be specified by the number or directly selected.





Operations available on the recipe view

Depending on the configuration, users can make the following operations on the recipe view:

1. Create, copy, modify, delete, rename and print recipe record

2. Read the recipe data record from the PLC or create the value of the recipe record to the PLC

3. Import and export the recipe

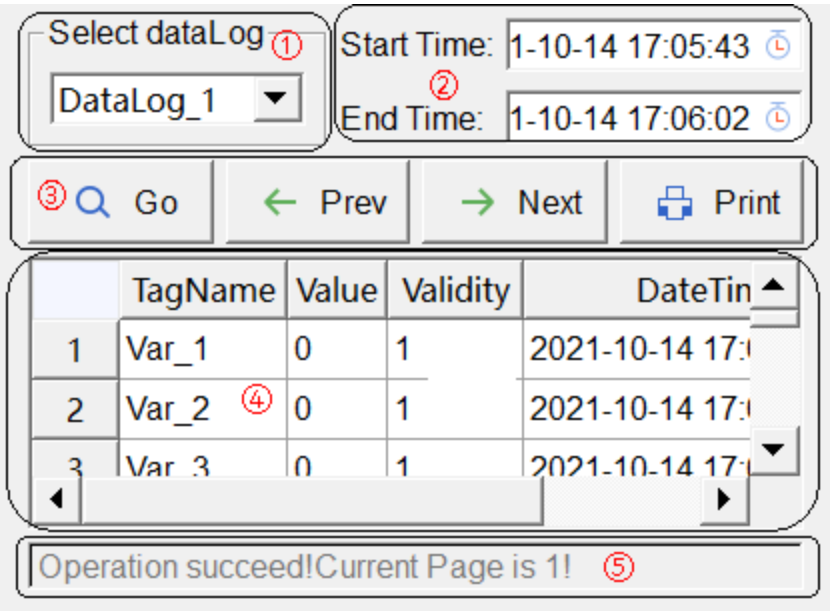
Button	Function
New	Create a recipe data record
Delete	Delete the currently displayed data record
Save As	Save the selected data record with a different name for replication.
Rename	Rename the currently displayed data record.
Save	Save the modifications made on the currently displayed data record.
Print	Print the currently displayed data record
To PLC	Transferring the value of the selected data record from the HMI device to the PLC
From PLC	Display the recipe value in PLC on HMI
Synchronize tag	Synchronize the tag values corresponding to the recipe ingredients to the recipe record. Note: This feature is active only if the synchronization tag is enabled in the recipe configuration.
Info text	Displays information text for recipe view configuration

7.2.19 Data View

The data view is a general-purpose enhanced screen object for displaying, editing and managing data records. It presents data records in tables. In the view, you can search for values of historical data.

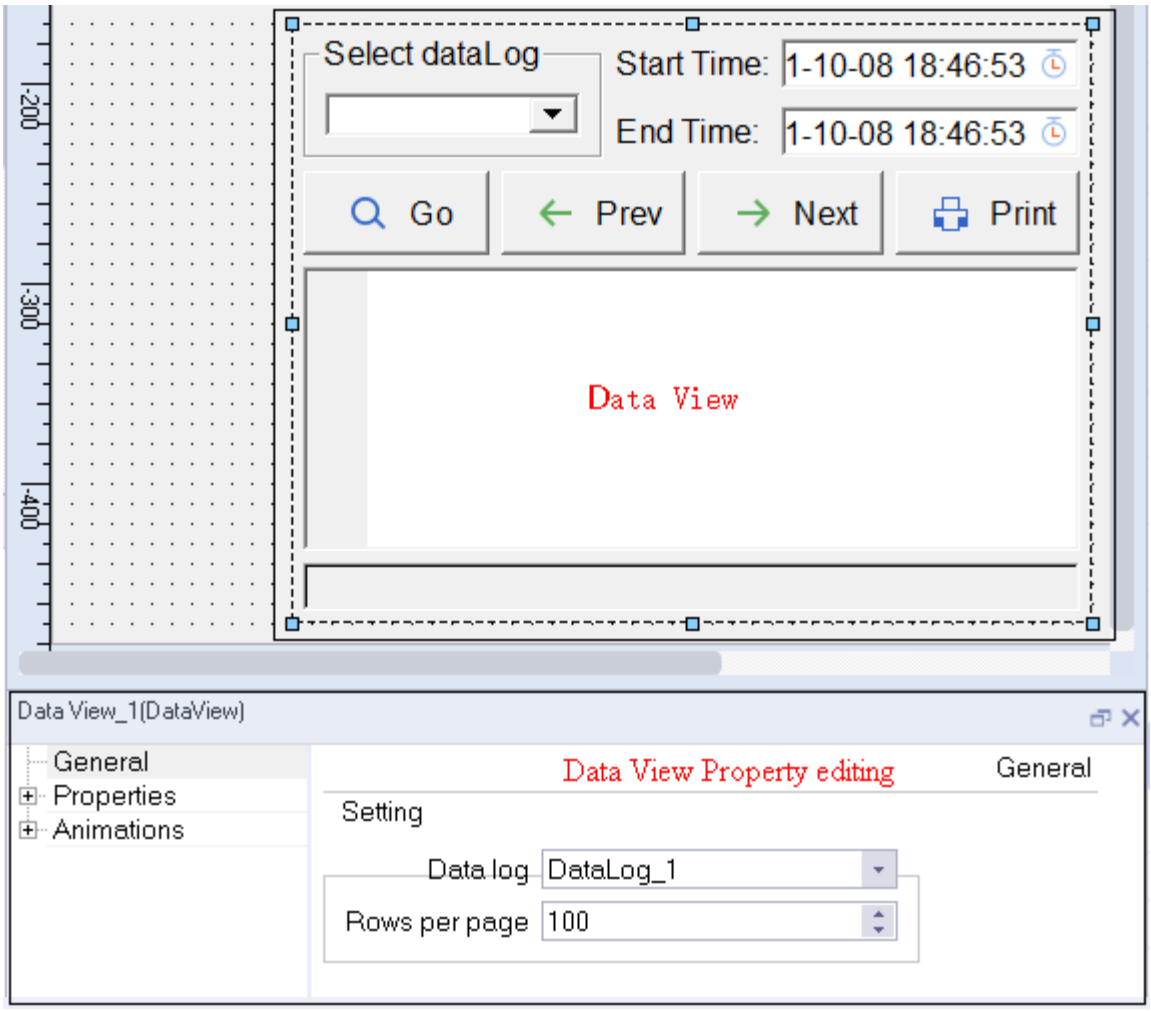
The data view mainly consists of five areas:

1. Data log list
2. Time editing area
3. Button
4. Data log table
5. Operation prompt box



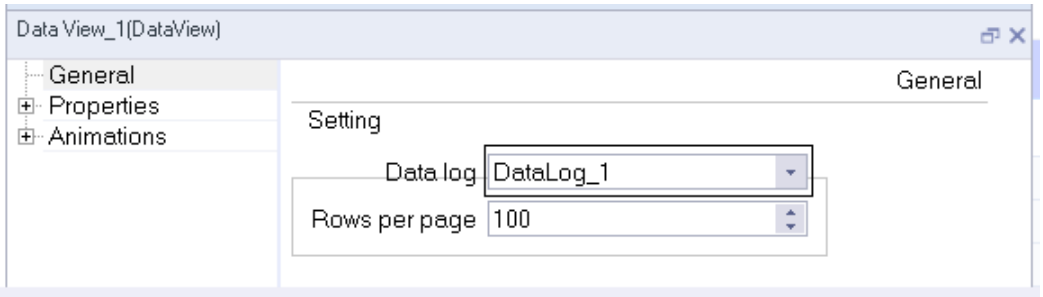
Configuring a data view

Select the data view in the screen and configure it through the Properties window.



General setting

1. Displays a specific data log.
   You can specify the name of the data log to be displayed when the runtime starts.



2. The maximum number of rows per page of the data log table.
   The table displays 100 rows by default.

Property setting

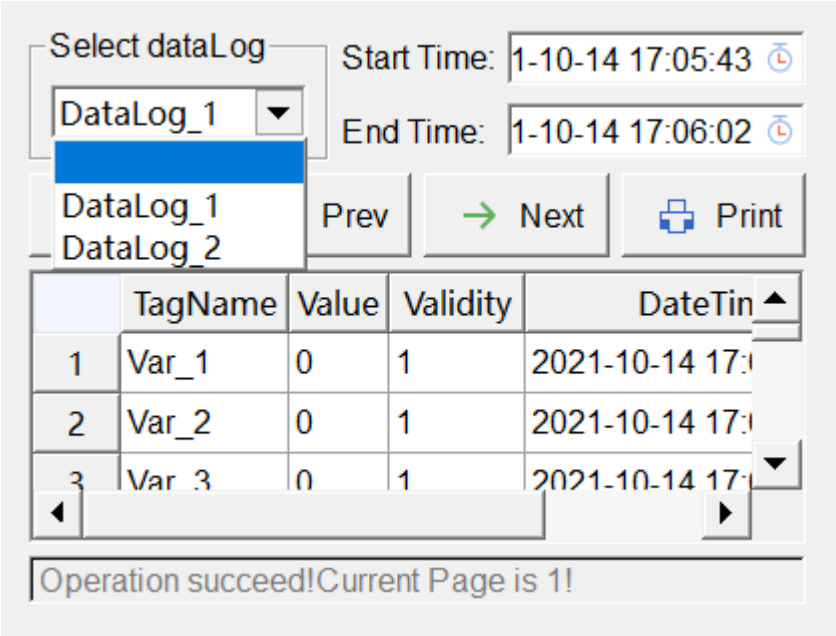
Name			Description
Appearance	Fill	Text color	The color of the text in the view, excluding the data table
		Background color	The background color of the view, excluding the data table
Layout	Position	X	Horizontal distance, in pixels, from the origin of the data view to the left edge
		Y	Vertical distance, in pixels, from the origin of the data view to the top edge
	Size	Width	The width of the data view,, in pixels
		Height	The height of the data view, in pixels
Text	Font		Defines the style and size of the data view text font, default is Droid Sans Fallback, font size 12
Buttons	General commands/menu entries		Includes Search, Update, Print, Last Page, and Next Page
Table	Table	Foreground color	The color of table text
		Background color	The background color of the table
		Header foreground color	The color of the table header text
		Header background color	Header background color
Miscellaneous	Miscellaneous	Layer	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

Animation

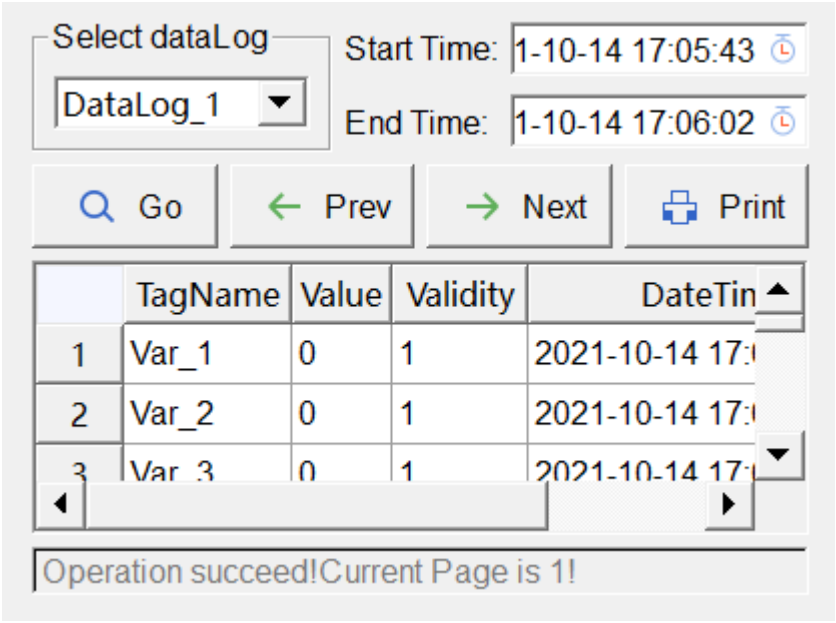
See *"7.4.1 Animation" on page 189*.

Using the data view

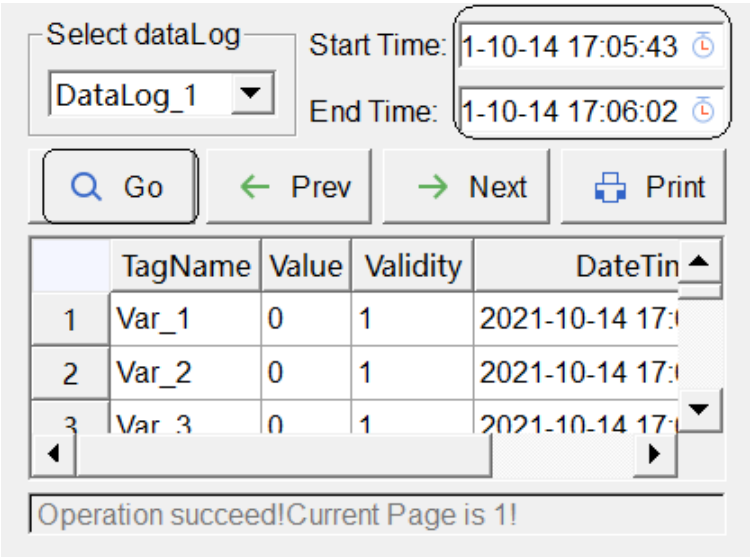
In the runtime, you can view data log information through the data log table.



For example, select data record _ 1, This is to display all the data recorded in Data Record 1 to the table (and the query start time and end time are divided into the time of the first data recorded and the time of the last data recorded, respectively, and the start and end time will be reloaded every time the data is switched). The historical data of the record (showing the variable name, value, active, and date and time of the record) will be displayed:

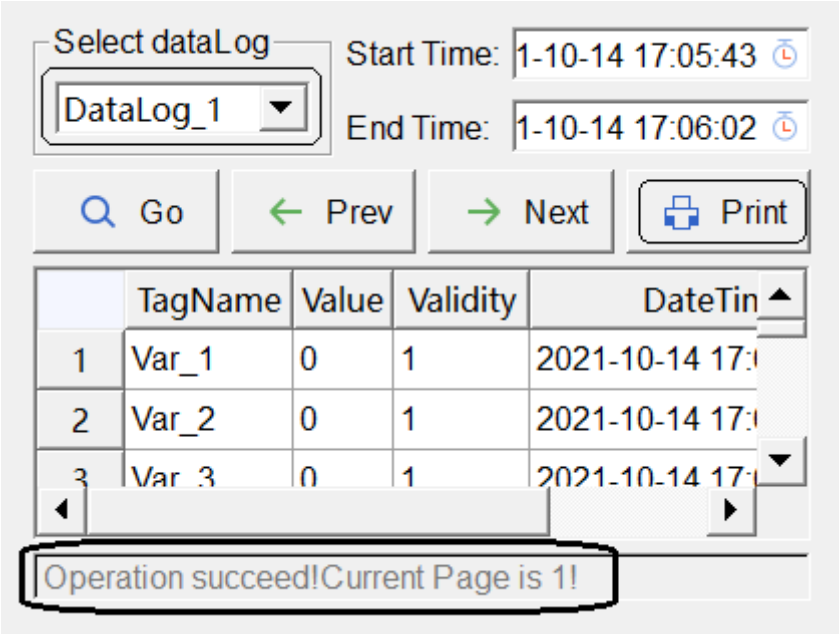


At this time, the user can edit the record data of the period to be searched in the query time editing area, for example, the record data in the interval from 2019-01-26 09:30:23 to 2019-01-26 09:31: 55 needs to be searched, enter these two time points at the start time and end time of the data view, and then click Find to find the records of the required interval:



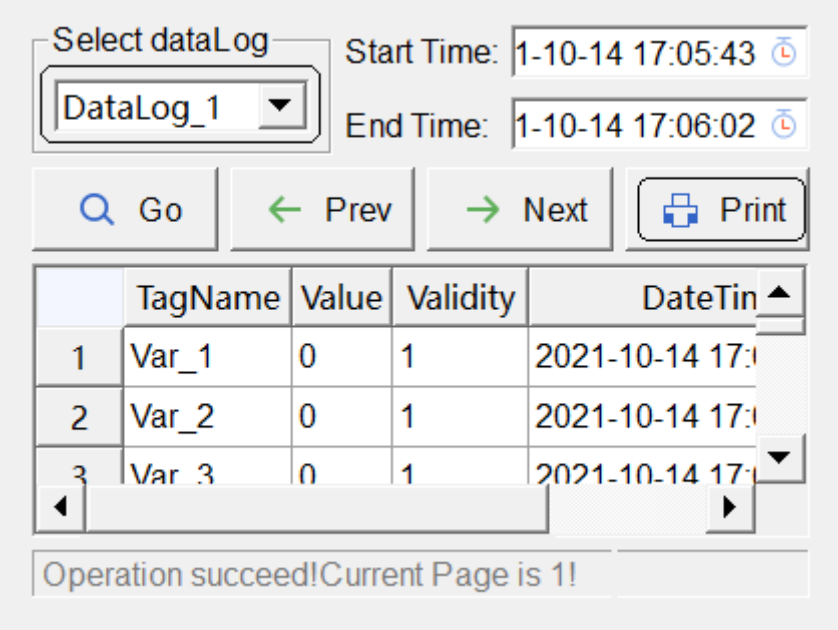
Note: "valid" refers to whether the data value of the recorded row is valid. 1 means valid and 0 mean invalid. For example, when an external tag is logged and the connection is disconnected, the logged value is actually valid, while the validness of the logged value is 0.

You can set the number of rows displayed on each page of the data log. When there are a lot of data, you can click the Pre and Next buttons to navigate the pages. The message bar will also show the current page number.



Printing

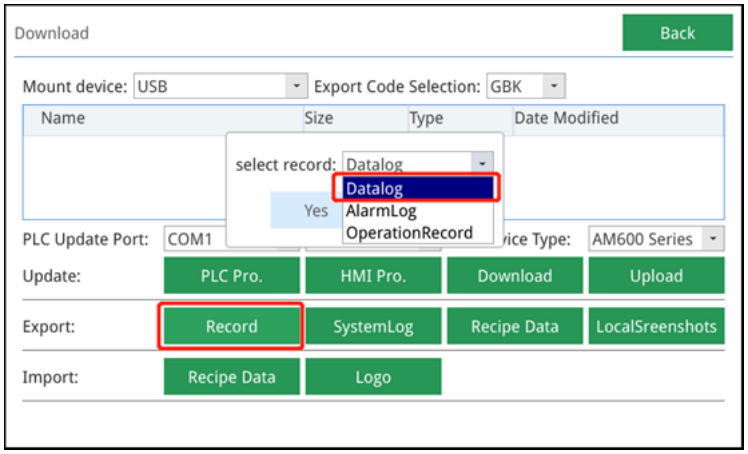
If you need to print the data log information, you can select the data log and click the Print button.



In the screen, you can use related system functions to manipulate the data logs.

Exporting data logs

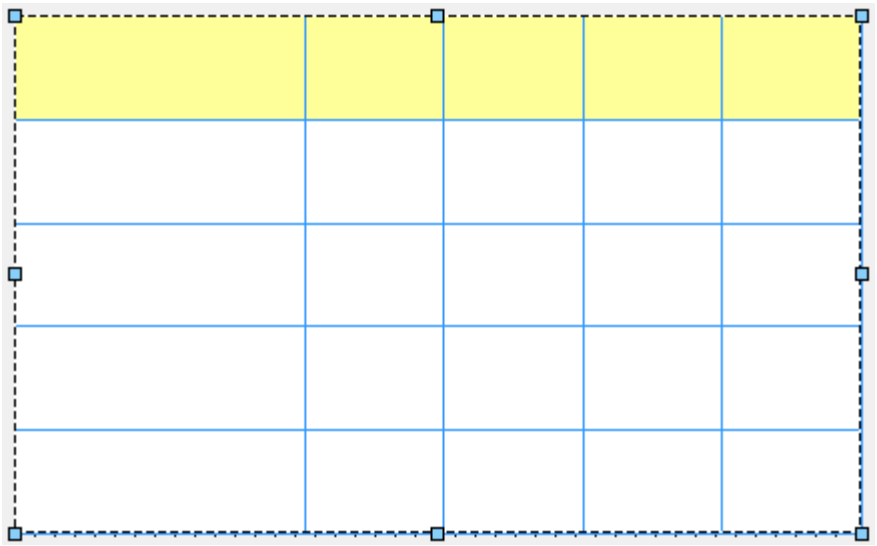
You can export the data logs to a USB drive or SD card in the Download page of the control panel.



During export, all the data logs in the project will be exported to the specified path and saved to a *. CSV file (each data log is saved as a file separately).

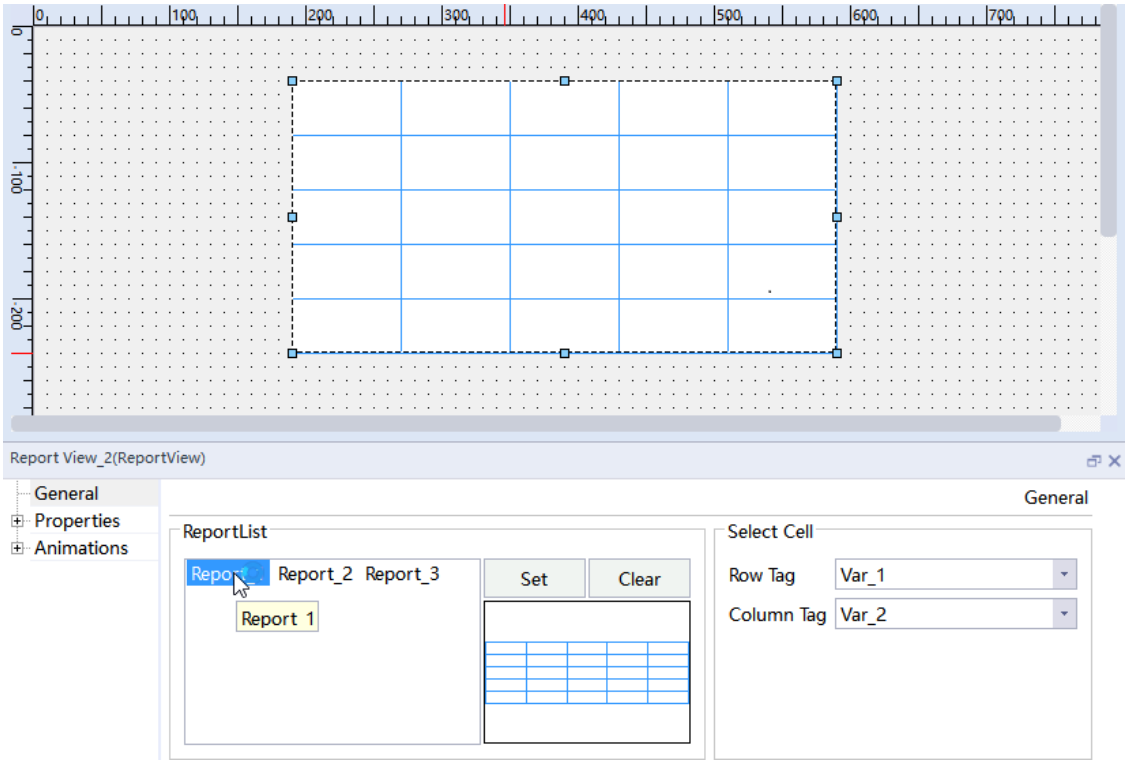
7.2.20 Report View

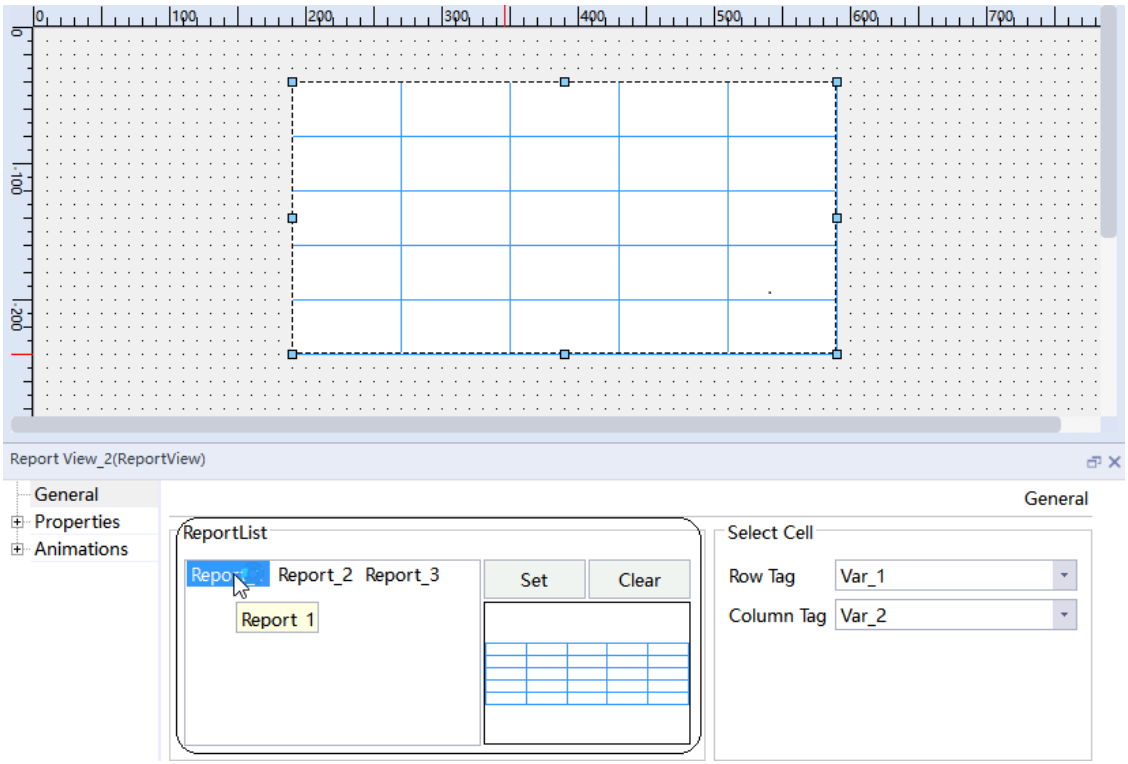
The report view is a general-purpose enhanced screen object for displaying and managing reports. It presents report data in a table. In the report view, you can edit and view report data. The following is a report view in the screen:



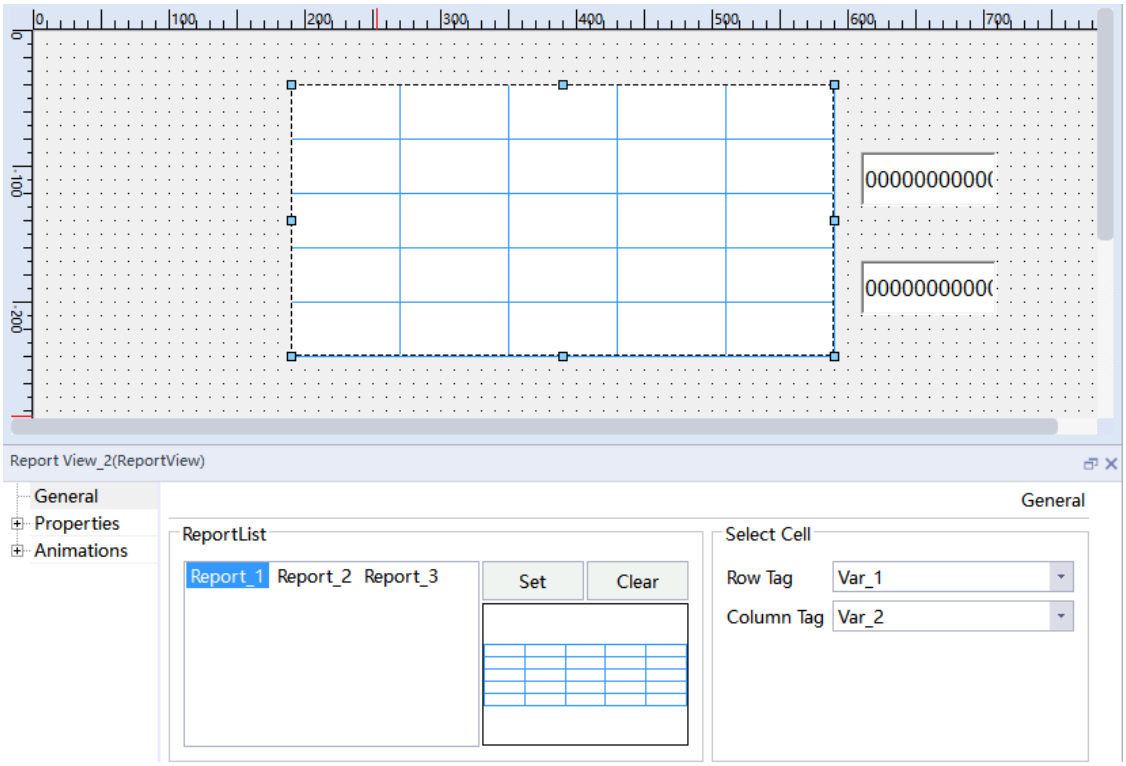
General setting

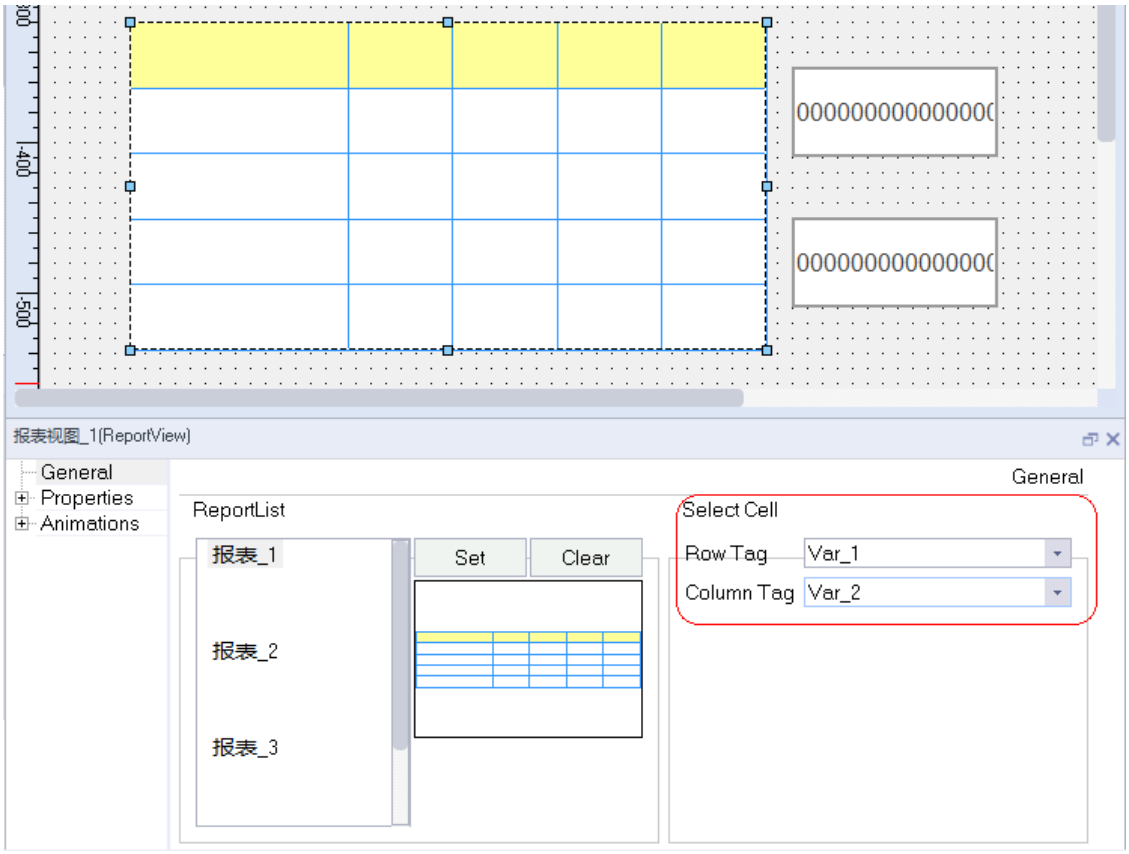
1. Report List: Lists all reports created by the user, where you can specify which report to display.





2. Select Cell: row tag, column tag





The row tag and column tag can be used to display the index of the selected cell in the report view. In addition, you can control the selected cell by changing the values of row and column tags.

Property setting

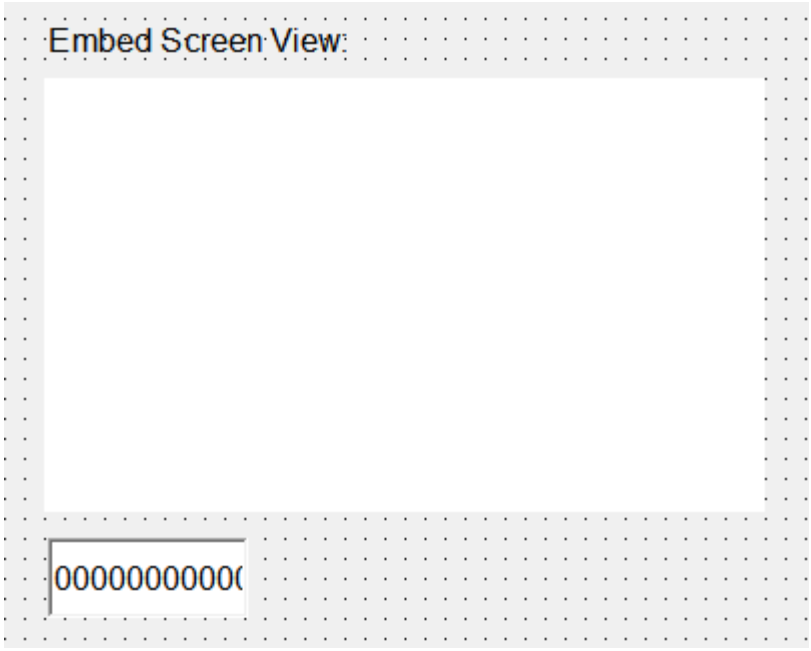
Name			Description
Layout	Position	X	Horizontal distance, in pixels, from the origin of the report view to the left edge
		Y	Vertical distance, in pixels, from the origin of the report view to the top edge
	Size	Width	The width of the report view, in pixels
		Height	The Height of the report view, in pixels
Buttons	Size, Operation		Set the buttons on the report view. See *"15.3.3 Property Settings of Report History Cells" on page 335*
Miscella-neous	Miscella-neous	Layer	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

For animation properties, see *"7.4.1 Animation" on page 189*.

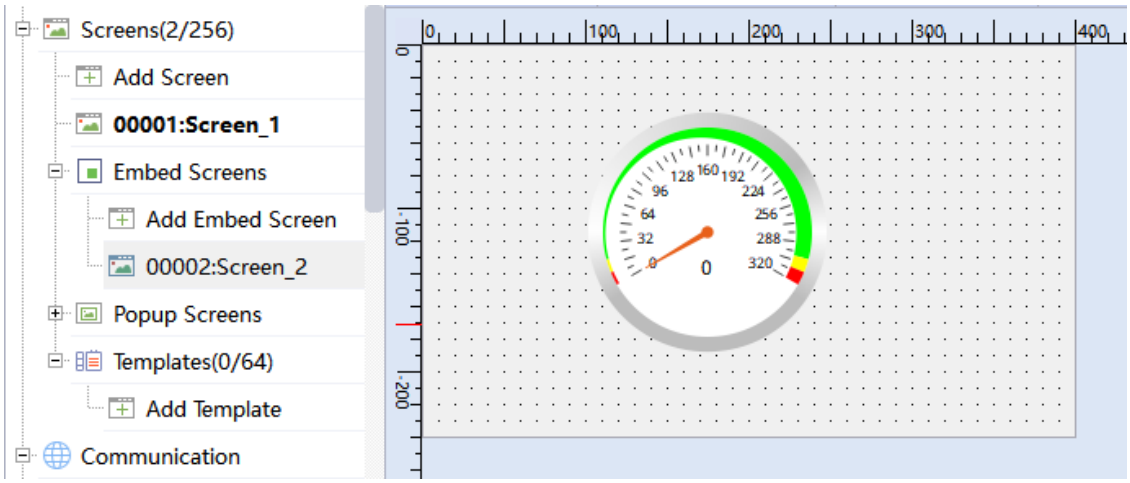
7.2.21 Picture-in-picture View

The picture-in-picture view shows an additional screen in the current screen. The screen can be shown and hidden through tag configuration.

1. Drag the picture-in-picture view.



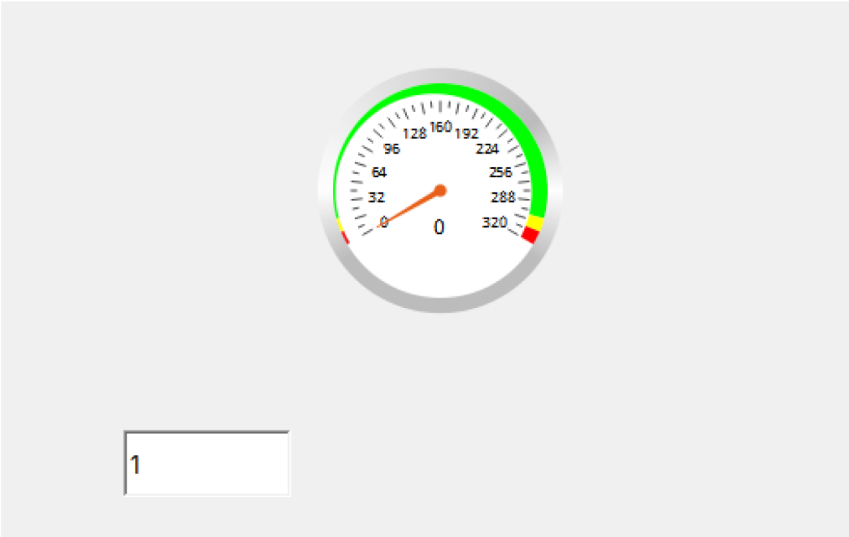
2. Configure the picture-in-picture screen.



Property setting:

General setting:

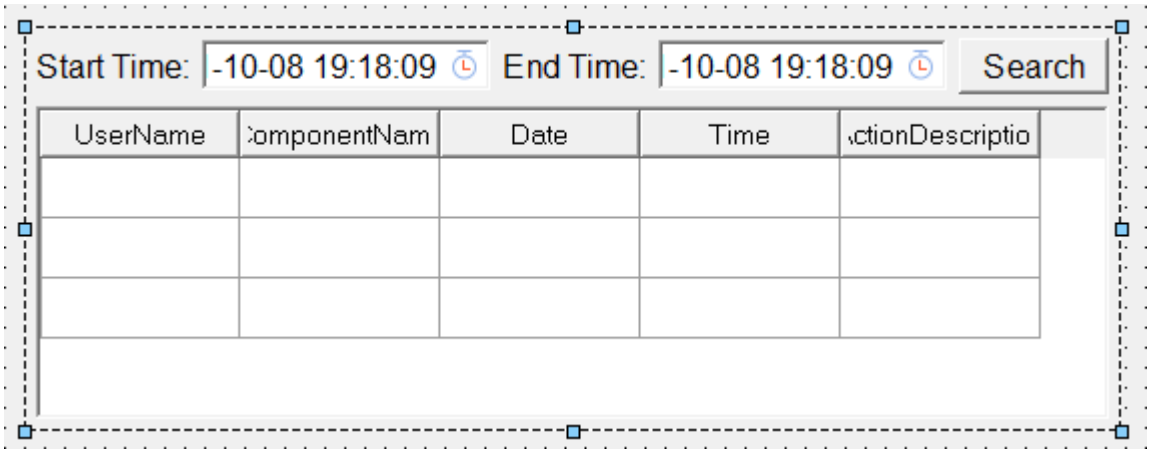
- Screen list: You can configure the picture-in-picture view.
- Process tags: You can control the display of the picture-in-picture screen through tags.



By manipulating process tags, you can display the picture-in-picture screen.

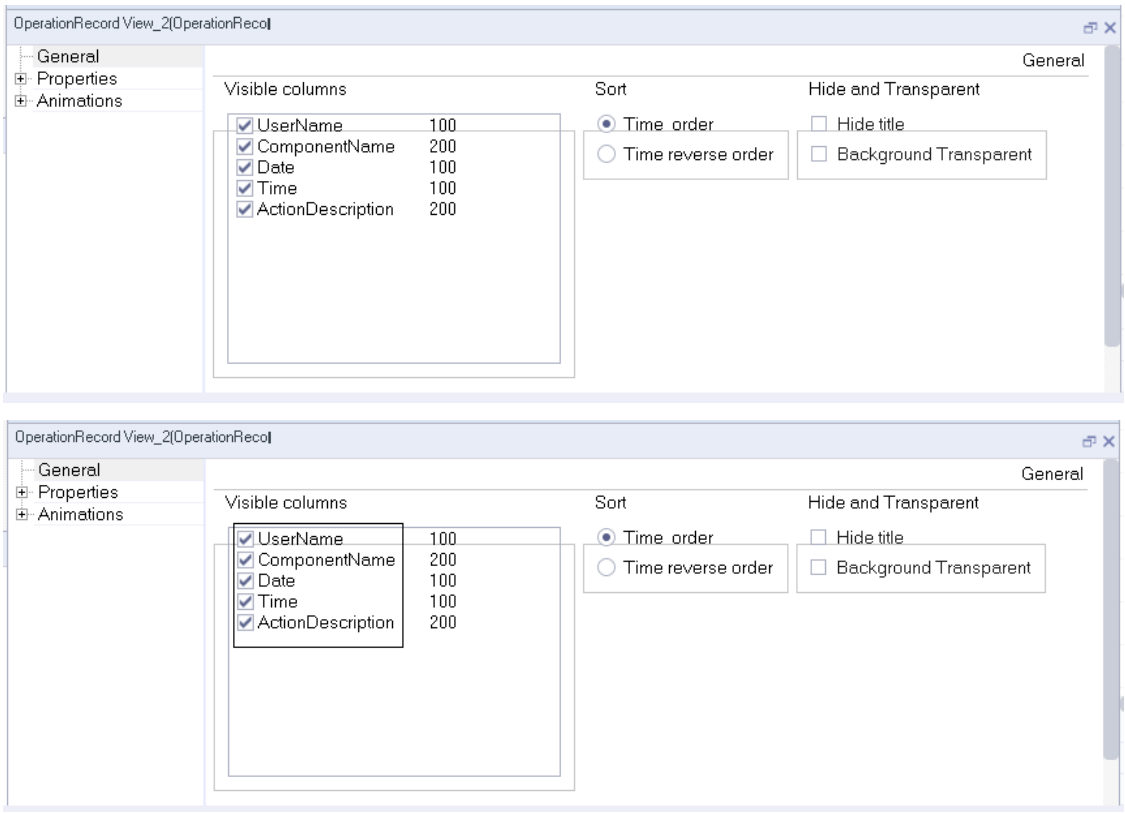
7.2.22 OperationRecord View

The OperationRecord view is a general enhanced screen object used to display operation records in a table. To use this view, you need to check Enable Operation Record in HMI Settings→Project Settings. In the Operation Record view, you can search and view the operation records of operators in a certain period of time. The following is the Operation Record view interface:



General setting

1. Visible columns: There are five columns: user name, component name, date, time and action description. Check a column to display it in the view, and all columns are checked by default, as shown in the following figure:



2. Sort: You can sort operation records by time in chronological (default) or reverse chronological order.

Property setting

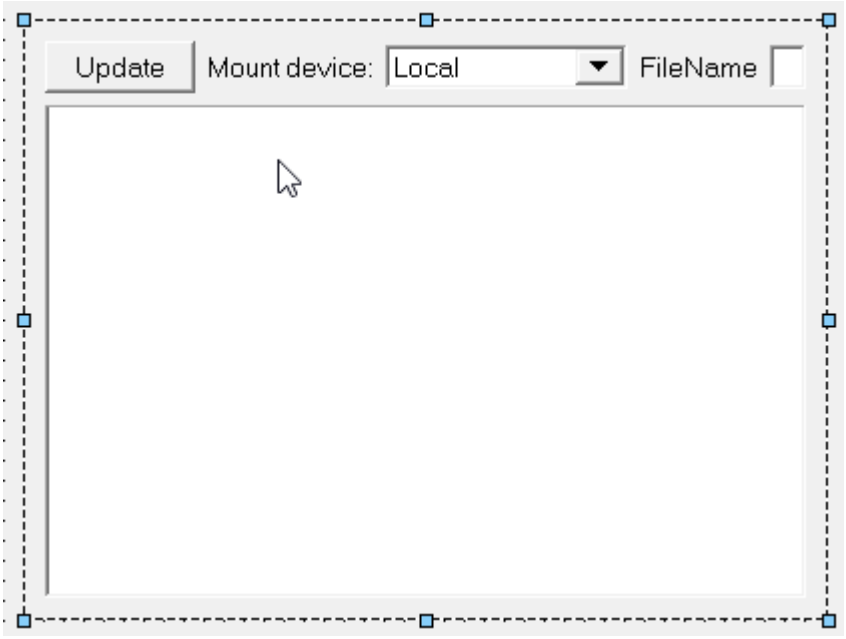
Name			Description
Layout	Position	X	Horizontal distance, in pixels, from the origin of the Operation Record view to the left edge
		Y	Vertical distance, in pixels, from the origin of the Operation Record view to the top edge
	Size	Width	The width of the view, in pixels
		Height	The height of the view, in pixels
Table	Table	Foreground color	The color of table text
		Background color	The background color of the table
		Header foreground color	The color of the table header text
		Header background color	Header background color
		Font	style and size of table text
Miscellane-ous	Miscellaneous	Layer	The screen layer in which the object is located, the value range is 0–31, and the default value is 0
		z Value	The position of the control in the z axis. A control with a larger value covers a control with a lower one.
		Opacity	Value range: 0–100, default: 100, 0 for transparent, 100 for opaque

Name			Description
Security	Runtime Security	Authorization	Assigns users with permissions
	Operation	Enabled	Enables the security
		Visibility controlled by authorization	Check this to control visibility through authorization

For animation properties, see *"7.4.1 Animation" on page 189*.

7.2.23 File Browser View

The file browser view is a general enhanced screen object, which is used for displaying files in devices such as a USB drive.



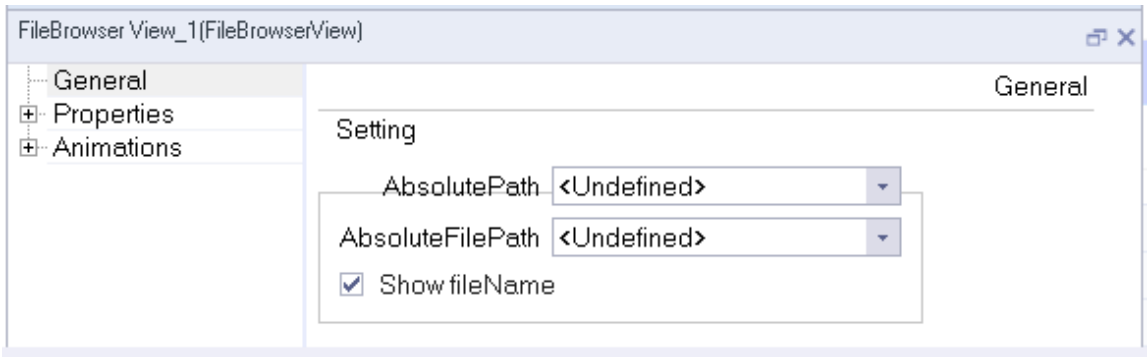
1. Update: refreshes the list.
2. Mount device: the device whose files you want to browse.
3. File name: displays the file name.

General setting:

Setting and configuration items are absolute path and absolute file path, and you can also configure whether the file name is displayed.

Absolute path: the absolute path of the file after the tag is configured.

Absolute file path: the path of the file after the tag is configured.



7.3 Plug-ins

Menu bar/**toolbar**

The menu bar 'format' is produced and managed inside the screen module, and the main frame only provides docking function.

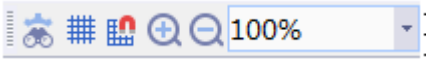
Property			Description	Restrictions
Align-ment	Horizontal	Top	Align the object with the top edge of the reference object	At least 2 objects should be selected
		Center	Aligns the centers of all selected objects horizontally	
			Objects are centered along the common horizontal center axis	
		Bottom	Aligns the object with the bottom of the reference object	
	Vertical	Left	Aligns the object with the left edge of the reference object	
		Middle	Aligns the centers of all selected objects vertically	
			Objects are aligned along the common vertical center axis	
		Right	Aligns the object with the right edge of the reference object	
Distribu-tion	Horizontal distribute		Makes the horizontal spacing between controls equal	At least 3 objects should be selected
	Vertical distribute		Makes the vertical spacing between controls equal	
Size	Equal width		Adjusts the width of the object according to the reference object	At least 2 objects should be selected. The last selected object is the reference
	Equal height		Adjusts the Height of the object according to the reference object	
	Equal size		Adjusts the width and height of the object according to the reference object	
Lock	Lock		Lock the object so that it cannot be moved	Lock and Unlock are mutually exclusive: Unlock is enabled after Lock is executed and Lock is enabled after Unlock is executed
	Unlock		The object can be moved after unlocking	

Property		Description	Restrictions
Arrange-ment	Bring to front	Move an object to the top position	The Z value changes
	Bring forward	Move the object forward by one object	Stack is different from a layer
	Bring backward	Move the object backward by one object	
	Send to back	Move an object to the bottom position	
	Same Z value	Place the selected objects on the same position	At least 2 objects should be selected
Group	Group	Groups the selected objects	At least 2 objects should be selected
	Ungroup	Ungroup the grouped objects	
Selection	Select All	Select all controls on the current screen	Press shift + left mouse button to select multiple objects
	Select None	Deselects all selected controls	
	Invert Selection	Inverts selection	
Rotate		Rotate the selected object	You must only select one object at a time

The alignment functions in the toolbar are the same as the commands in the Format menu;

The arrangement functions in the toolbar are the same as the commands in the Format menu;

Toolbar: Zoom in and out. Zooms in or out the screen. Show Grid and Snap to Grid are the same as the settings in Options→Settings→Screen options. You can select Show Grid to show gridlines on the screen, and Snap to Grid to snap an object to the gridlines when you move it.



You can click Address Table to show all the used addresses.

7.4 Animation and Event Properties

7.4.1 Animation

Animation is used to control whether an object is visible in runtime, can be operated, whether it can be moved, how to move, etc.; Different objects have different animation attributes, and there are nine basic animation attributes:

Animation	Description
Appearance Change	Defines the foreground color, background color and flashing of dynamic control objects in runtime.
Object Enable	Defines whether objects are enabled in runtime.
Diagonal Movement	Defines movement path of a dynamic object.
Horizontal Movement	Defines movement path of a dynamic object.

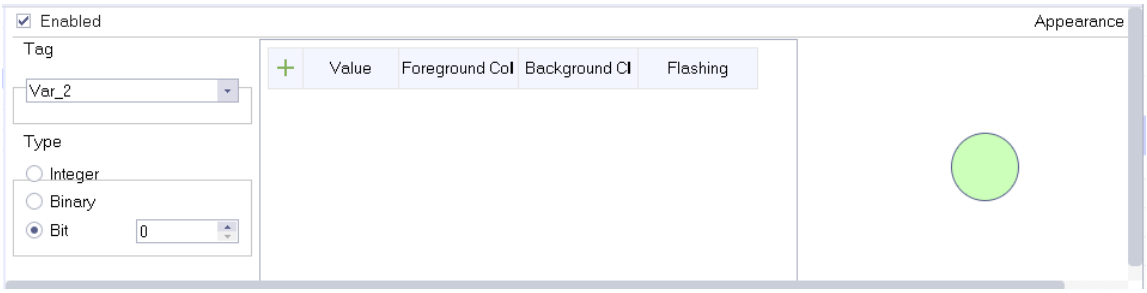
Animation	Description
Vertical Movement	Defines movement path of a dynamic object.
Direct Movement	The object moves a number of pixels along the X and Y coordinates.
Visibility	Enables the object in the scope of a tag.
Custom Movement	The object moves along custom coordinates.
Dynamic Movement	Configures the values of tags X1, X2, Y1 and Y2 to control the dynamic position change of a line object.

You can only configure one of the six movement modes at the same time.

Appearance Change

You can dynamically control the appearance of objects, including foreground color, background color and flashing, by changing the value of the tag, If you configure a tag for object appearance, when the tag value is within the configured range, the object is displayed as configured, otherwise the settings in Appearance take effect.

The following is the Appearance Change setting page of an object:



As shown in the above figure, the configuration consists of:

"Enable": Check this to enable the appearance change setting.

"Tag": The tag to control the appearance of the object.

"Type": The type of the tag value. See the following table.

Type	Description
Integer	The appearance of the object is controlled by the value or range of the tag. You can define any number of values or ranges, and each definition represents an appearance. You cannot define duplicate values or ranges.
Binary	Different bits of the tag correspond to different appearances. You can define the number of appearances according to the tag type. For example, a BYTE type tag has 8 bits, which can define 8 appearances. You cannot define duplicate bits.
Bit	The appearance of the object is controlled by a bit in the tag or a bit tag. There are only two appearances.

"Value": the value of the tag.

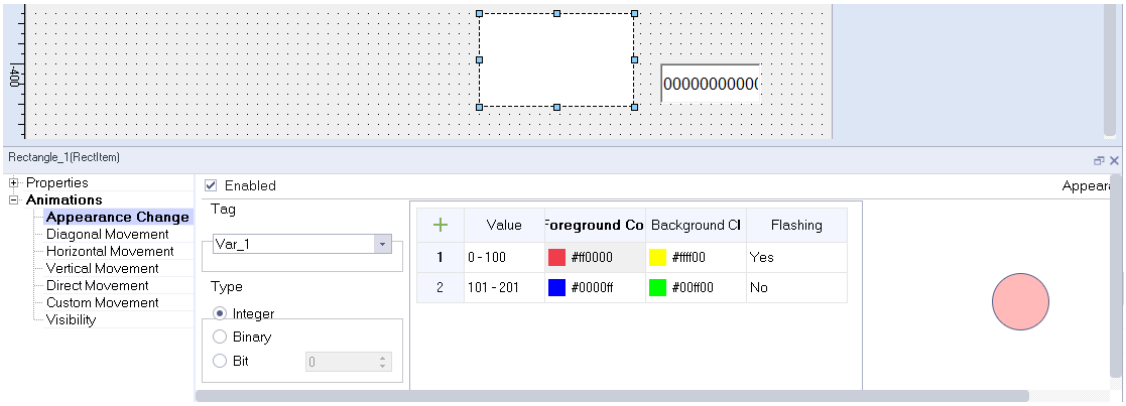
"Foreground Color": the foreground color of the object displayed on the runtime.

"Background Color": the background color of the object displayed on the runtime.

In the following example, the appearance change of a rectangle object is configured:

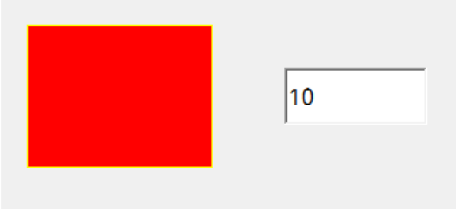
1. Select the rectangle object, then expand the Animation property in its properties view, and select Appearance Change.
2. Check Enabled.

3. Configure the tag. Click the drop-down list box under Tag to open the tag list and select a tag. Select the data type of the tag (in this example, Tag_1 is of Int16 type).
4. Configure the type. Choose one of the three types as needed. "Integer" is selected in this example.
5. Configure the appearance in the table. Click the + button on the upper left corner of the table to add appearances and you can set the foreground color, background color and flashing as needed.
6. The configuration is as follows:

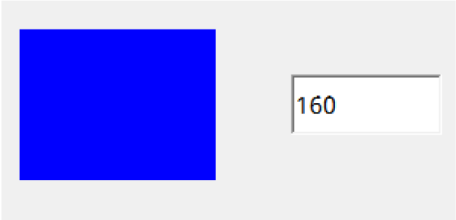


The simulation is as follows:

In the IO Field, when you enter "10", which is between 0 and 100, the rectangle becomes red and flashes.



In the IO Field, when you enter "160", which is between 101 and 201, the rectangle becomes blue and stops flashing.

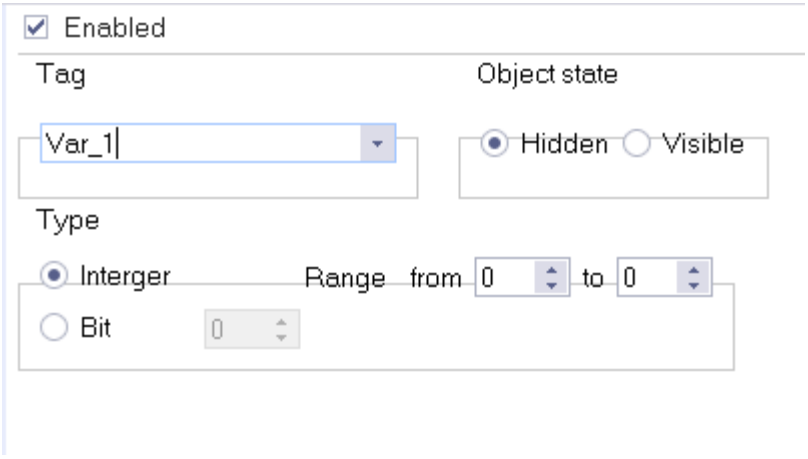


2 Enable object

The "enable object" configuration can dynamically control the availability of an object.

In the runtime, the object becomes enabled/disabled according to the tag value.

The following is the "enable object" setting page of an object:



As shown in the above figure, the configuration consists of:

"Enable": When you tick the check box, the object is enabled when the tag value is within the configured range.

"Tag": The tag to control the availability of the object. The object availability changes according to the value or value range of the tag.

"Type": The type of the tag value. See the following table.

Type	Description
Integer	If the object state is Enabled, the object is enabled when the tag value is within the range, and disabled otherwise. The opposite is true when the object state is Disabled.
Bit	If the object state is Enabled, the object is enabled as long as the bit value is 1, and disabled as long as the bit value is 0. The opposite is true when the object state is Disabled.

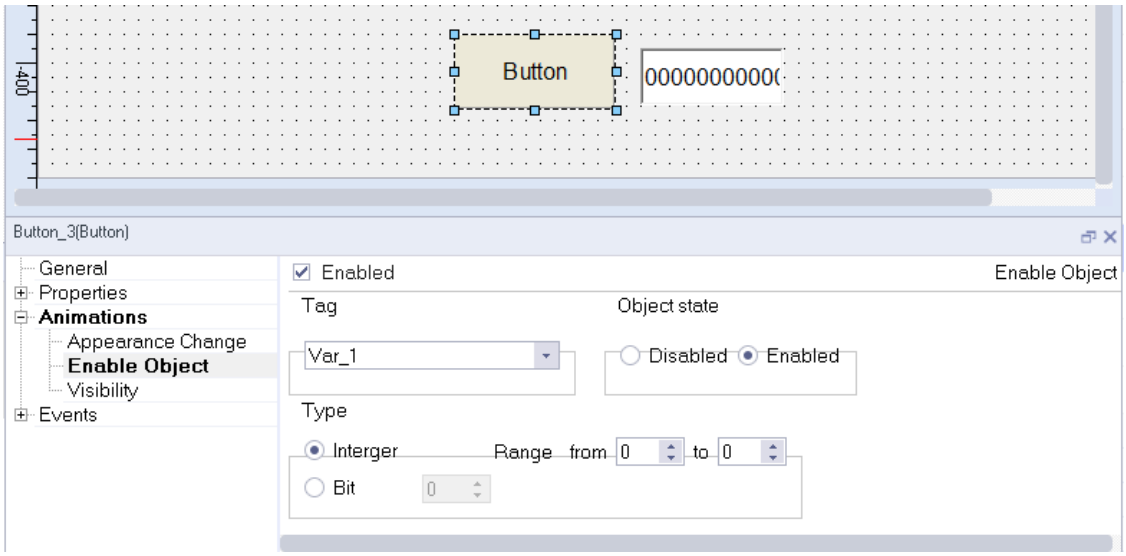
"Object state": Determines the availability of the object.

States: "Enabled" and "Disabled". See the following table.

Operation	Description
Enabled	The object is enabled when the value of the tag is within the range.
Disabled	The object is disable when the value of the tag is within the range.

In the following example, the Enable Object of a button object is configured:

1. Select the button object, then expand the Animation property in its properties view, and select Enable Object.
2. Check Enabled.
3. Configure the tag. Click the drop-down list box under Tag to open the tag list and select a tag. Select the data type of the tag (in this example, Tag_1 is of Int16 type).
4. Configure the type. Choose one type as needed. "Bit" is selected in this example.
5. Select a bit in the spin box (such as bit '0').
6. Select Enabled in Object State.
   The configuration is as follows:



The simulation is as follows:

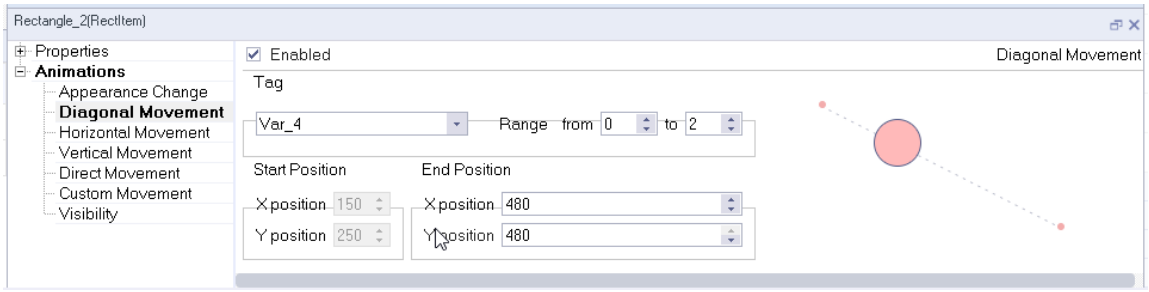
When you enter "1" in the IO field, the button is enabled.

When you enter "0" in the IO field, the button is disabled.

3 Diagonal Movement

The object can be configured to move along a particular path or to move a particular number of pixels relative to the current position. Movement is controlled by a tag. You can move a proportional distance diagonally according to the value range of the tag.

The following is the Diagonal Movement setting page of an object:



As shown in the above figure, the configuration consists of:

"Enable": Tick this check box to enable the Diagonal Movement setting. The object is enabled when the value is within the range.

"Tag": The tag determines the position of an object on the path.

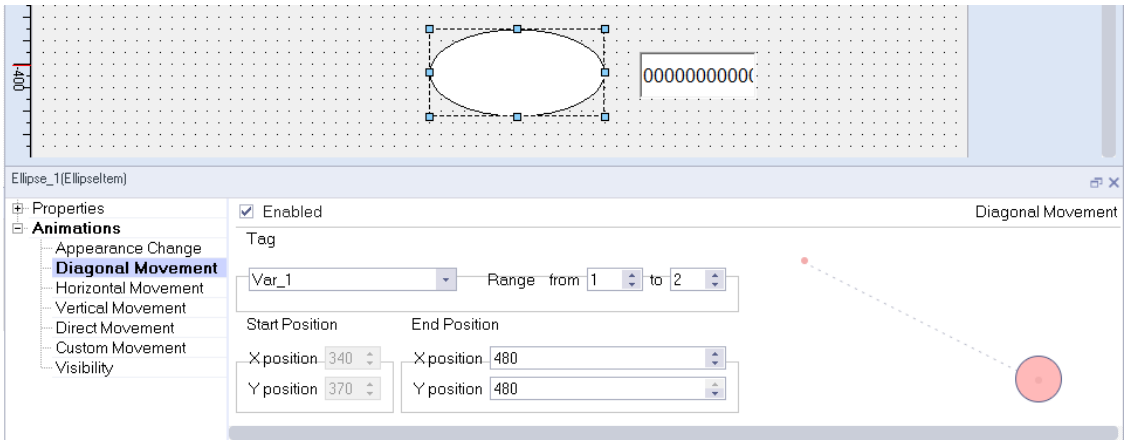
"Range": The range of the tag.

Start Position: X, the start coordinates of the path.

End Position: the end coordinates of the path.

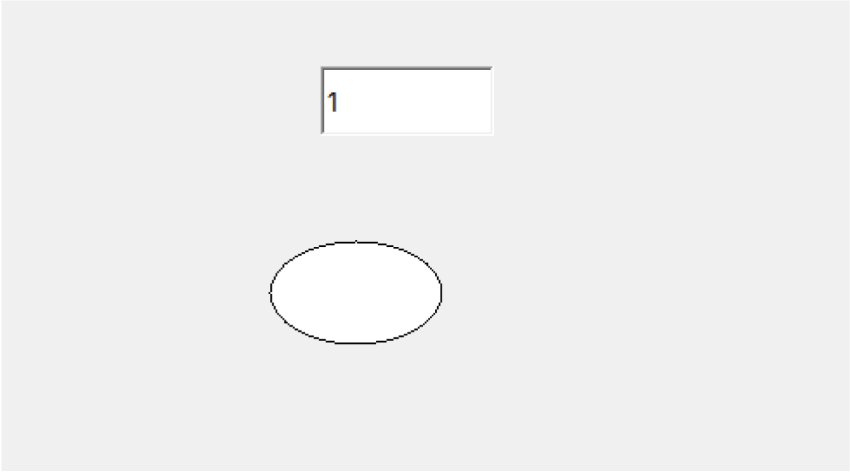
In the following example, the Diagonal Movement of an ellipse object is configured:

1. Select the ellipse object, then expand the Animation property in its properties view, and select Diagonal Movement.
2. Configure the tag. Click the drop-down list box under Tag to open the tag list and select a tag. Select the data type of the tag (in this example, Tag_1 is of Int16 type).
3. Configure the Range, Start Position, and End Position.
   The configuration is as follows:



The simulation is as follows:

When you enter "1" in the IO field, the ellipse moves to 1/2 of the diagonal path.

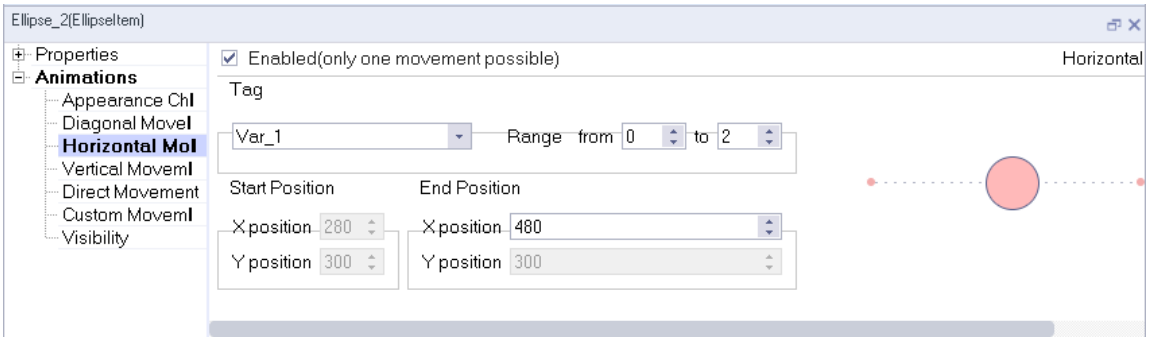


4 Horizontal Movement

The object can be configured to move along a particular path or to move a particular number of pixels relative to the current position. Movement is controlled by a tag.

The object can move a distance horizontally proportional to the value range of the tag.

The following is the Horizontal Movement setting page of an object:



As shown in the above figure, the configuration consists of:

"Enable": Tick this check box to enable the Diagonal Movement setting. The object is enabled when the value is within the range.

"Tag": The tag determines the position of an object on the path.

"Range": The range of the tag.

"Start Position": The start coordinates of the object, which is not editable.

"End Position": The end coordinates of the object. The Y coordinate is not editable.

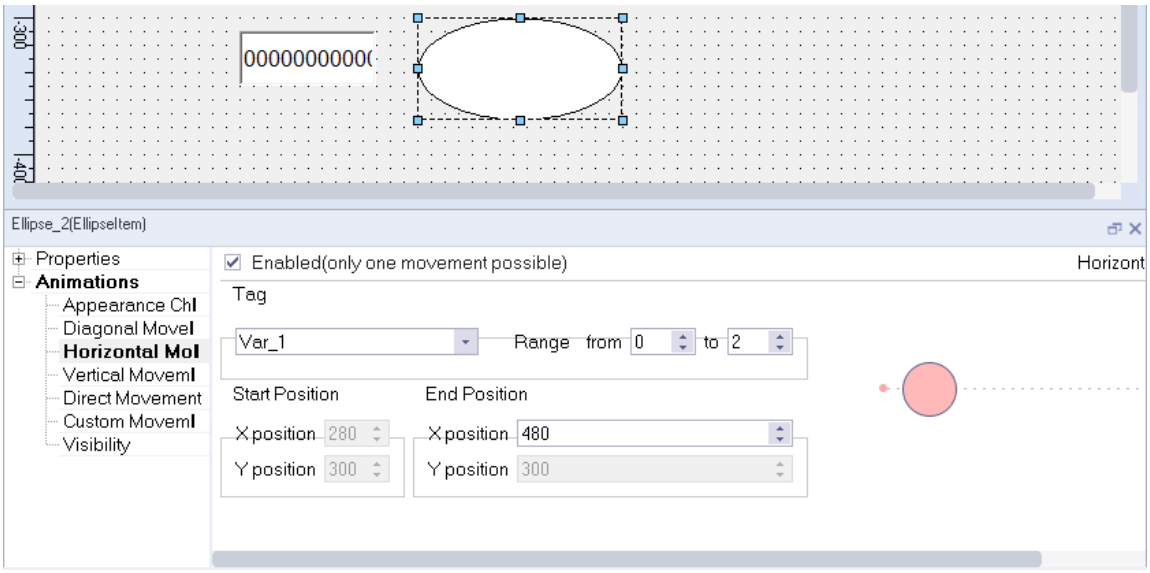
If the end X position is greater than the start X position, the object moves to the right, and if the end X position is less than the start X position, the object moves to the left.

X offset for every 1 change of the tag within the range = (X end - X start)/(max. range value - min. range value). If the tag value is greater than the max. range value, the object moves directly to the end position. The object will never move outside the screen. If the tag value is less than the min. range value, the object moves directly to the start position. The object cannot move outside the screen.

In the following example, the horizontal movement of an ellipse object is configured:

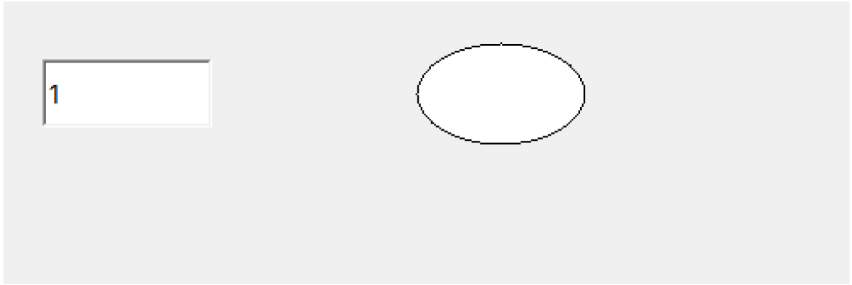
1. Select the ellipse object, then expand the Animation property in its properties view, and select Horizontal Movement.
2. Configure the tag. Click the drop-down list box under Tag to open the tag list and select a tag. Select the data type of the tag (in this example, Tag_1 is of Int16 type).
3. Configure the Range, Start Position, and End Position.

The configuration is as follows:



The simulation is as follows:

When you enter "1" in the IO field, the ellipse moves to 1/2 of the path.

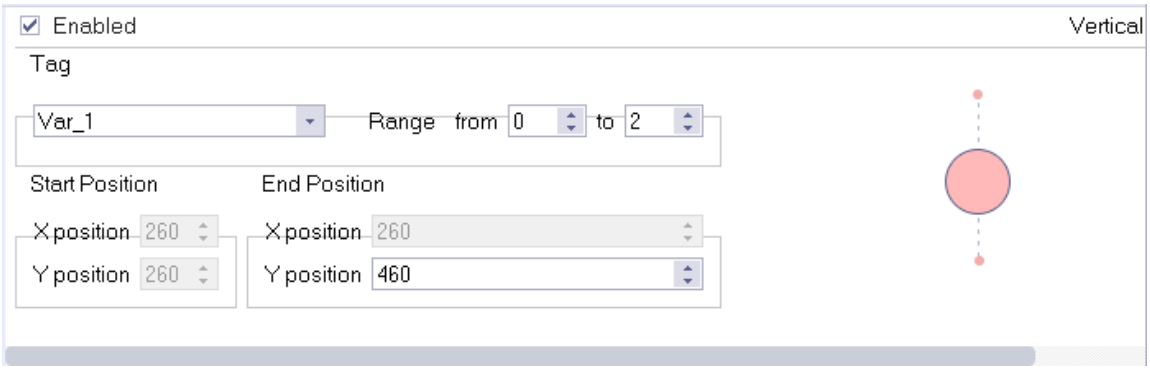


5 Vertical Movement

The object can be configured to move along a particular path or to move a particular number of pixels relative to the current position. Movement is controlled by a tag.

The object can move a distance vertically proportional to the value range of the tag.

The following is the Vertical Movement setting page of an object:



As shown in the above figure, the configuration consists of:

"Enable": Tick this check box to enable the Diagonal Movement setting. The object is enabled when the value is within the range.

"Tag": The tag determines the position of an object on the path.

"Range": The range of the tag.

"Start Position": The start coordinates of the object, which is not editable.

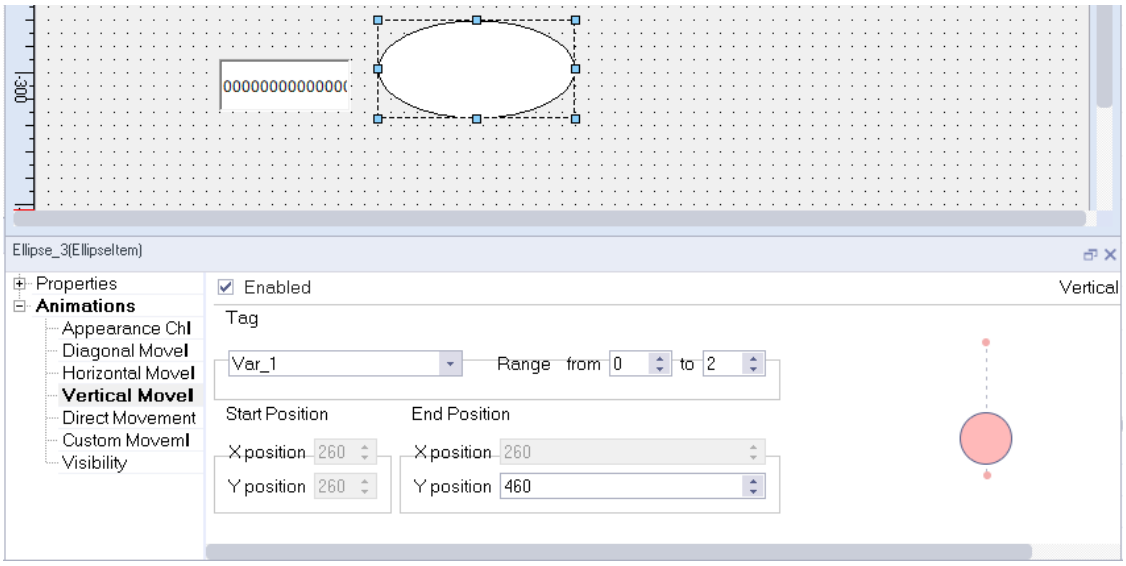
"End Position": The end coordinates of the object. The X coordinate is not editable.

If the end Y position is greater than the start Y position, the object moves downwards, and if the end Y position is less than the start Y position, the object moves upwards.

X offset for every 1 change of the tag within the range = (Y end - Y start)/(max. range value - min. range value). If the tag value is greater than the max. range value, the object moves directly to the end position. The object will never move outside the screen. If the tag value is less than the min. range value, the object moves directly to the start position. The object cannot move outside the screen.

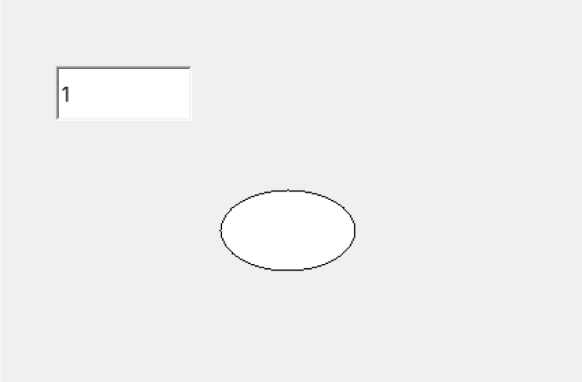
In the following example, the vertical movement of an ellipse object is configured:

1. Select the ellipse object, then expand the Animation property in its properties view, and select Vertical Movement.
2. Configure the tag. Click the drop-down list box under Tag to open the tag list and select a tag. Select the data type of the tag (in this example, Tag_1 is of Int16 type).
3. Configure the Range, Start Position, and End Position.
   The configuration is as follows:



The simulation is as follows:

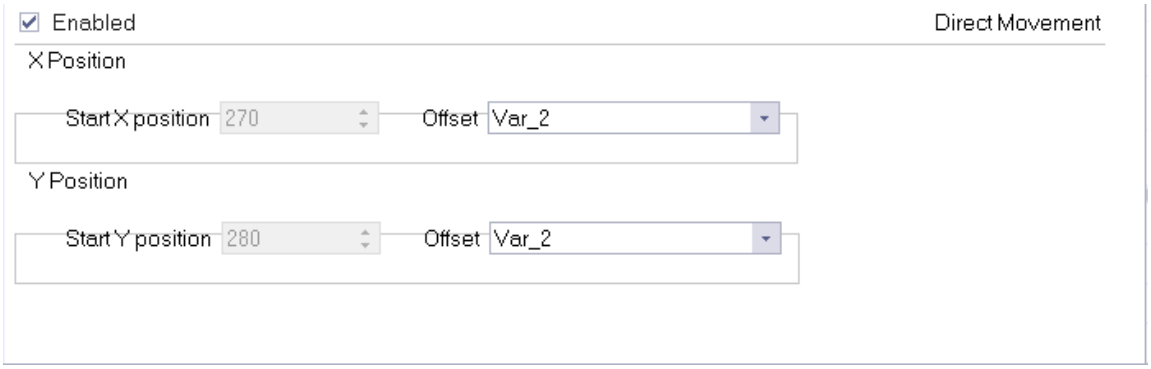
When you enter "1" in the IO field, the ellipse moves to 1/2 of the path.



6 Direct Movement

The object moves a number of pixels along the X and Y coordinates. The amount of movement is defined by the absolute values of two tags.

The following is Direct Movement setting page of an object:

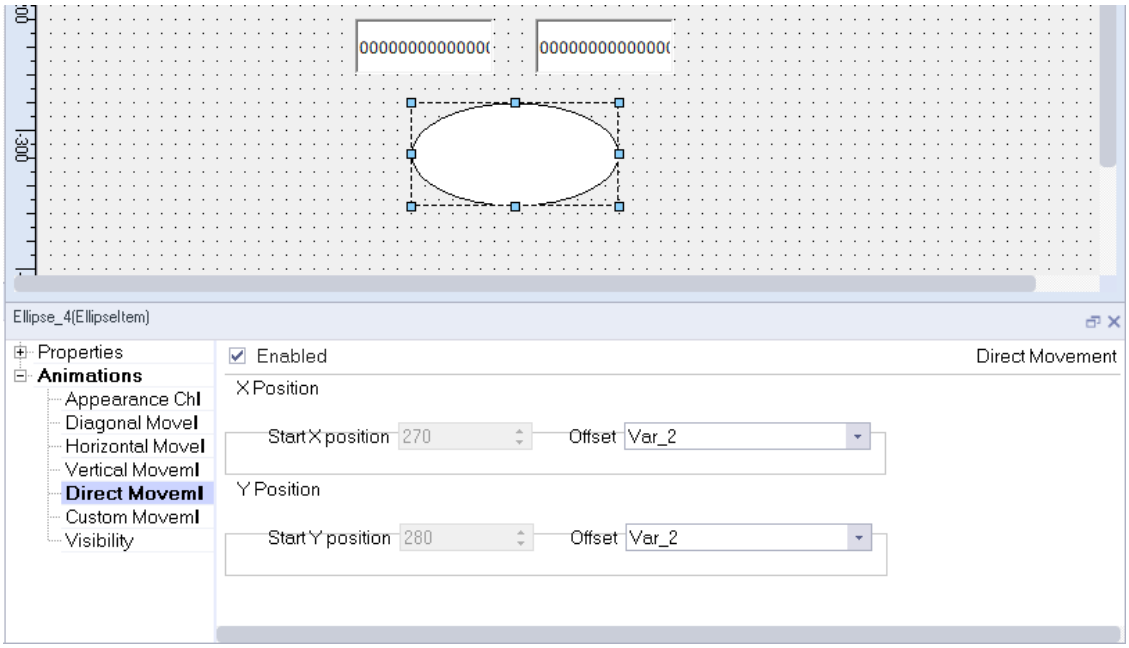


See the following table:

Property Name		Description
Enabled		Enables the Vertical Movement setting.
X position	Start X position	The start position of the control, which cannot be edited.
	Offset	The tag that configures the offset of the object on the X axis. If the value increases, the object moves to the right, and if the value decreases, the object moves to the left.
Y position	Start Y position	The start position of the control, which cannot be edited.
	Offset	The tag that configures the offset of the object on the Y axis. If the value increases, the object moves downwards, and if the value decreases, the object moves upwards.

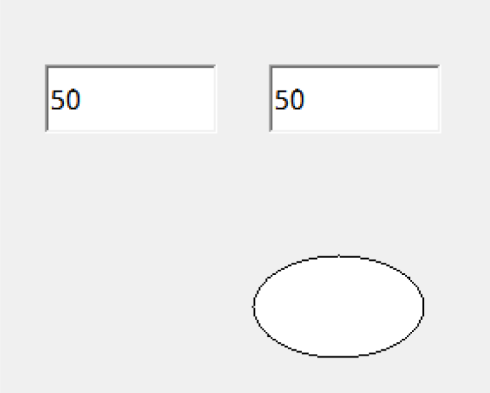
In the following example, the direct movement of an ellipse object is configured:

1. Select the object, then expand the Animation property in its properties view, and select Direct Movement.
2. Check Enabled.
3. Configure the X offset. Click the drop-down list to select a tag to control the offset.
4. Configure the Y offset. Click the drop-down list to select a tag to control the offset.
   The configuration is as follows:



The simulation is as follows:

When you enter "50" and "50" In the IO fields, the ellipse moves to the specified position.



7 Custom Movement

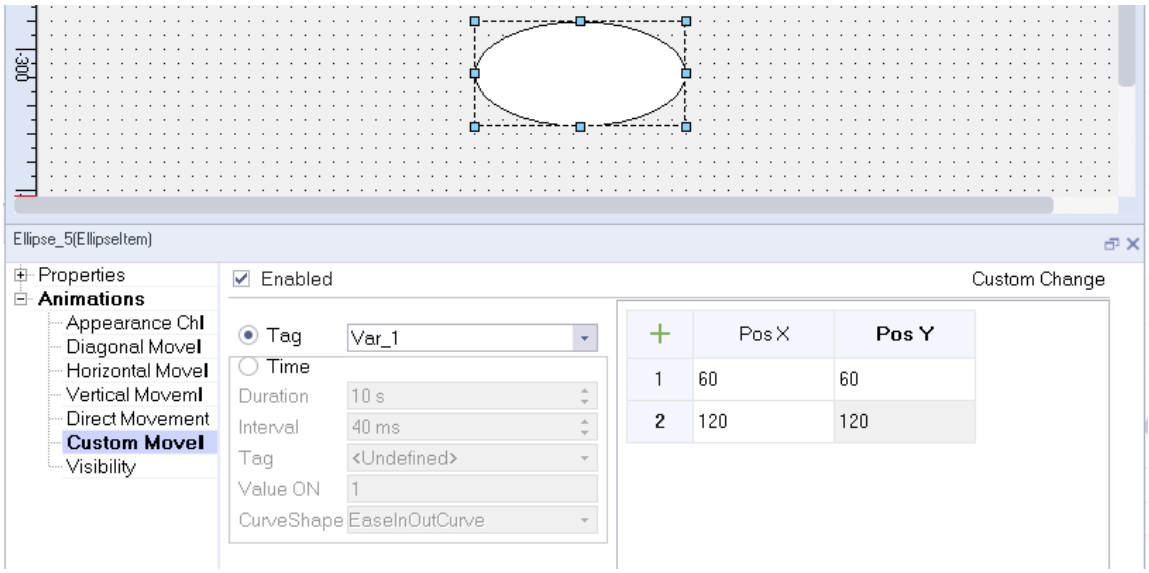
Custom movement refers to the movement of an object along custom coordinates. It can be controlled by a tag or time. In the tag mode, the object moves according to X and Y coordinates corresponding to the tag values. In the time mode, the object moves for a duration or at intervals. When the object moves for a duration, you can select the shape of its movement curve.

Up to 32 positions can be configured.

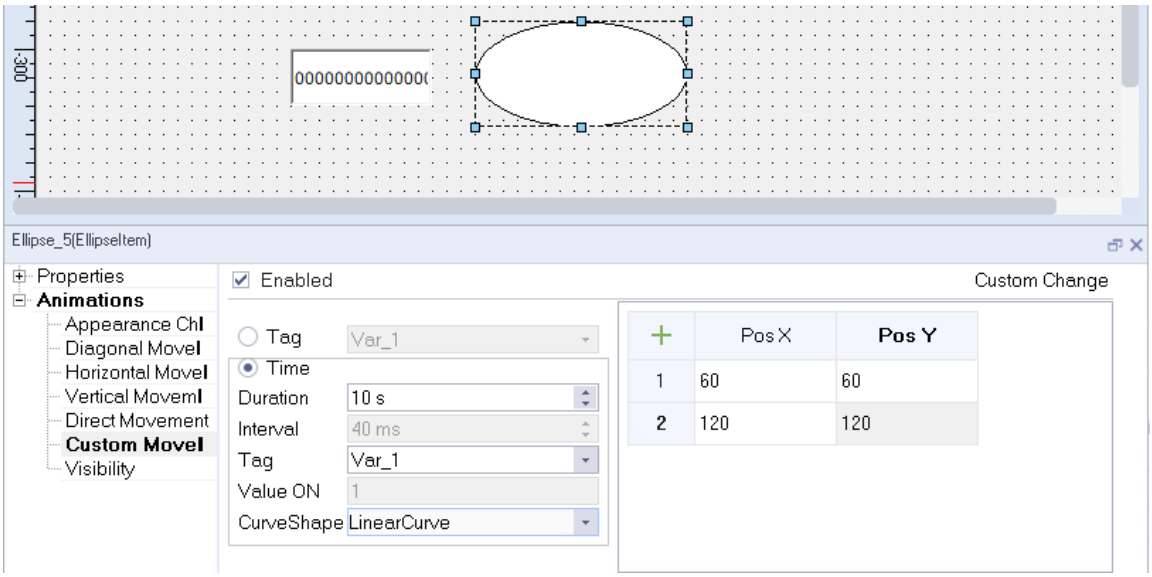
In the following example, the custom movement of an ellipse object is configured:

Select the ellipse object, then expand the Animation property in its properties view, and select Custom Movement.

Set the tag and coordinates, as shown in the following figure:



Set the time and coordinates, as shown in the following figure:



In the time mode, you can set the duration within 1s–60s. The following curve shapes are available:

Curve Shape	Description
Ease in curve	The object accelerates along the defined coordinates.
Ease out curve	The object decelerates along the defined coordinates.
Ease-in-out curve	The object accelerates and then decelerates along the defined coordinates.
Linear curve	The object moves at a uniform speed along the defined coordinates.
Sin curve	The object moves with a sine curve along the defined coordinates.
Cos curve	The object moves with a cosine curve along the defined coordinates.

In the time mode, set the tag type to bool to start and stop the movement. When the tag value is 1, the object moves along the defined coordinates, and when the tag value is 0, the object stops.

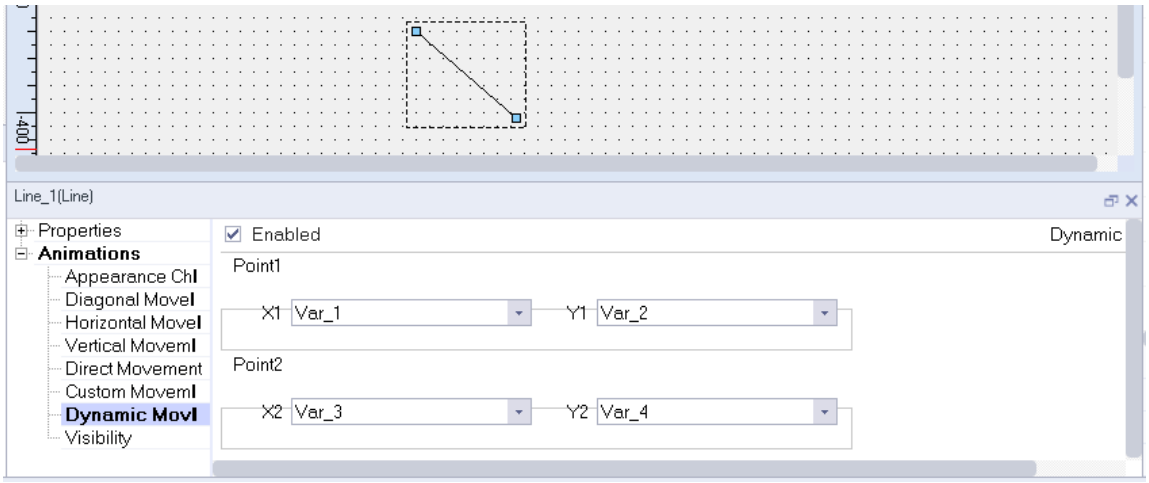
8 Dynamic Movement

The object moves according to the values of tags X1, X2, Y1 and Y2. (The movement only applies to the line object).

In the following example, the dynamic movement of a line object is configured:

Select the object, then expand the Animation property in its properties view, and double-click Dynamic Movement.

Configure the tags X1, X2, Y1 and Y2, as shown in the following figure:

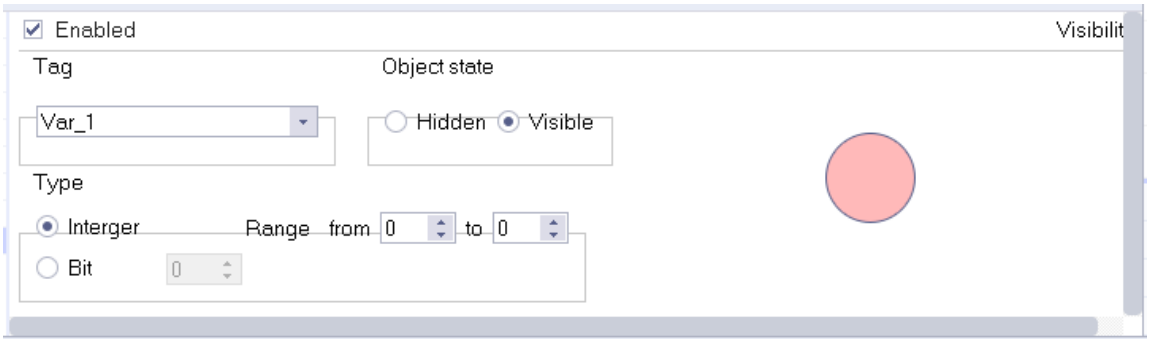


Note: (X1, Y1) is not necessarily the starting point of the line, but may be the end point.

9 Visibility

You can dynamically control the visibility of objects by changing the value of the tag, Whether an object is displayed depends on its state and the tag value. The object shows if the tag reaches a specific value or is in a specific range.

The following is Visibility setting page of an object:



As shown in the above figure, the configuration consists of:

"Enable": Check this to enable the Visibility setting.

"Tag": The tag to control the visibility of the object.

"Object state": the visibility state of the object. If you select “Visible", the object is visible when the tag value is in the configured range. If you select “Hidden", the object is hidden when the tag value is in the configured range.

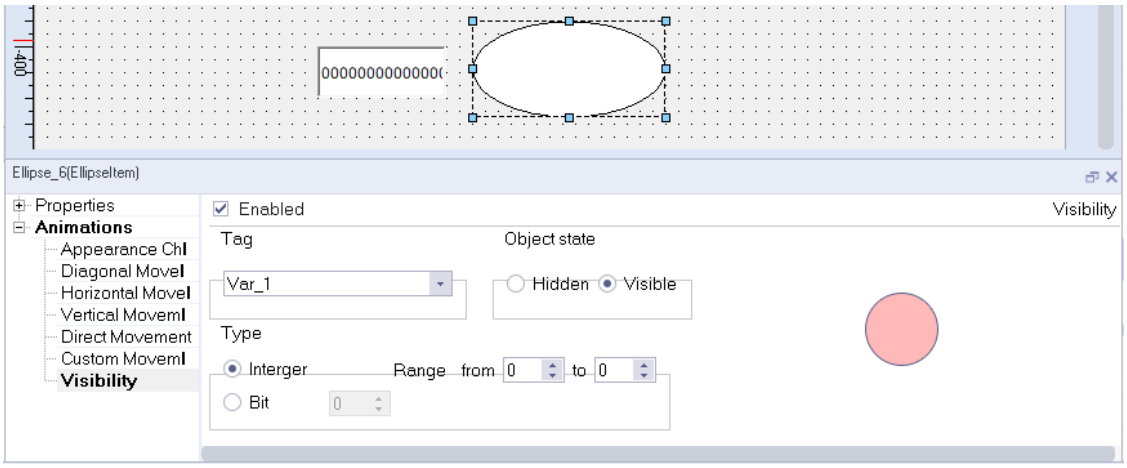
"Type": the type of the tag value. See the following table.

Property Name	Description
Integer	The visibility of the object is controlled by the value or range of the tag. If the object state is Visible, the object is visible when the tag value is in the configured range, and otherwise is hidden. The opposite is true when the object state is Hidden.
Bit	The visibility of the object is controlled by the bits of the tag. If the object state is Visible, the object is visible as long as the bit value is 1, and hidden as long as the bit value is 0. The opposite is true when the object state is Hidden.

In the following example, the visibility of an ellipse object is configured:

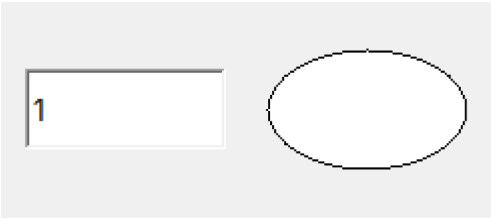
1. Select the ellipse object, then expand the Animation property in its properties view, and select Visibility.
2. Check Enabled.
3. Under "Object state", select Hidden or Visible as needed.
4. Configure the type. Choose one type as needed. "Bit" is selected in this example.
5. Select a bit in the spin box (such as bit '1').

   The configuration is as follows:



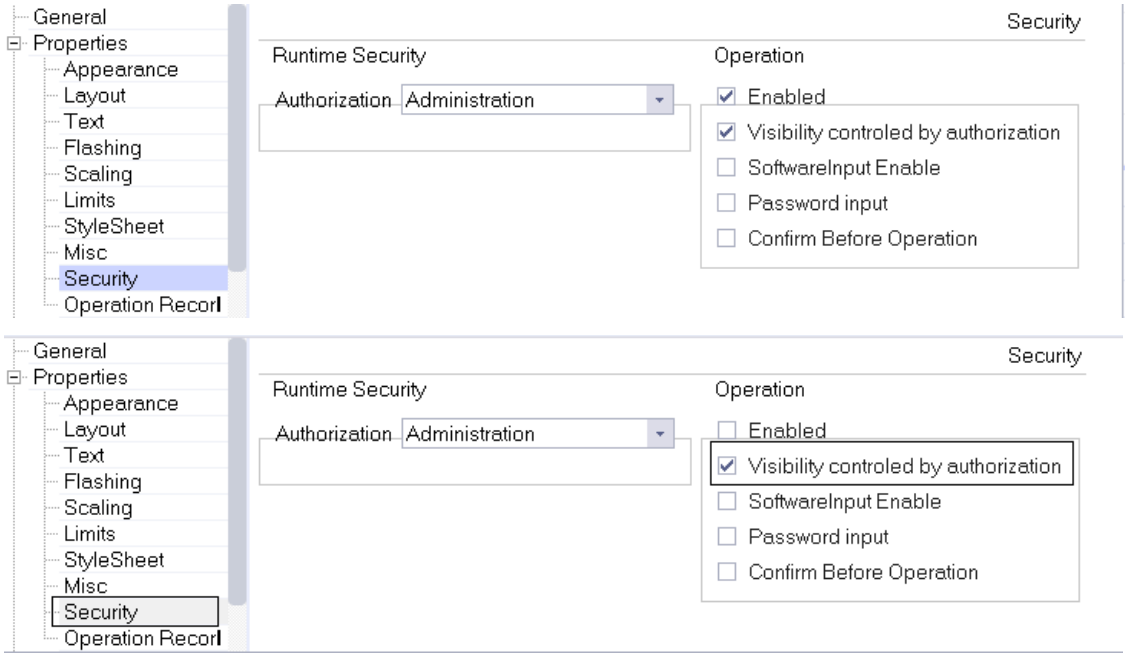
The simulation is as follows:

When you enter "1" in the IO field, the ellipse shows.



When you enter "0" in the IO field, the ellipse is hidden.

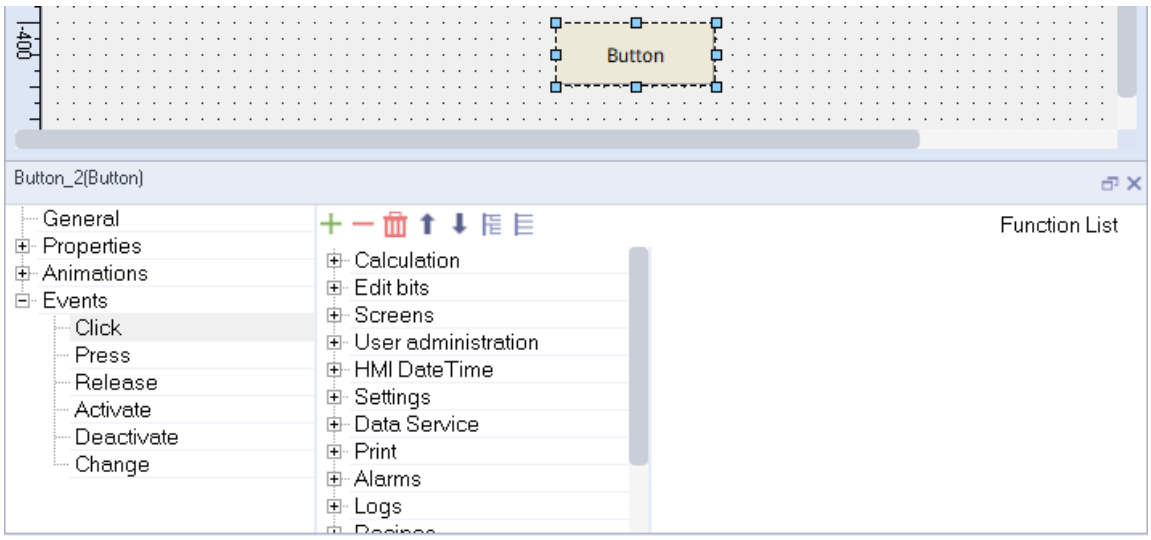
Note: If "visibility controlled by authorization" is checked in the Security property of the object, the visibility set in the Animation property is ineffective and non-editable.



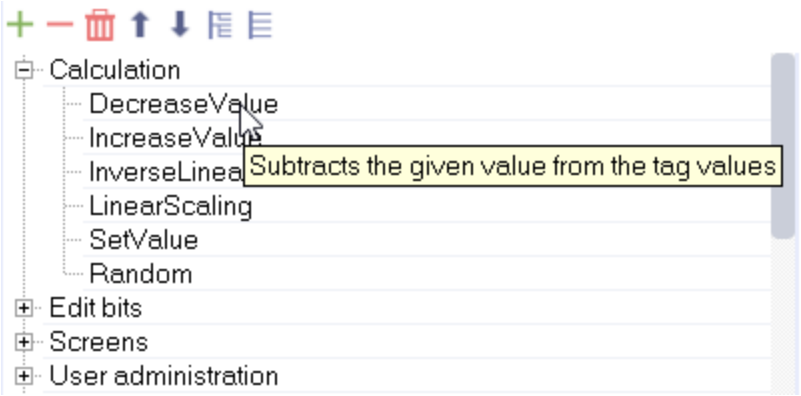
7.4.2 Event

An event is a condition that triggers the execution of a related task, such as the click of a button that triggers the execution of a task (system function or script) assigned to the event. A list of events and related functions, as well as user-created scripts, can be configured in the Properties dialog box. Which events can actually be used depends on the object type.

For example, a button has three events: click, press and release, as shown in the following figure:



When the user configures the system function, the help information of the function will be displayed when the mouse hovers on the function name, as shown in the figure:



For information about system functions associated with the events, see section System Functions and Running Scripts.

The events are described as follows:

Name	Description	Object
Loaded	Triggered when a screen or canvas is loaded.	Screen
Cleared	Triggered when a screen or canvas is cleared.	Canvas
Activate	Triggered when an analog or discrete alarm is triggered.	Analog Alarm Discrete alarm
Deactivate	Triggered when an alarm is cleared.	
Acknowledge	Triggered when you acknowledge an alarm.	
ReturnPressed	Triggered when you finish typing in a control.	Number IO field String IO field Date-Time Field
Change	Triggered when the state of an object changes.	Symbolic IO field Graphic switch Text switch Slider Gage
Switch off	Triggered when you set a switch in the OFF position.	Text switch Graphic switch
Switch on	Triggered when you set a switch in the ON position.	
Click	Triggered when a button is clicked (the cursor must on the button when you press and release the button).	Buttons
Press	Triggered when a button is pressed.	
Release	Triggered when a button is released.	
Upper limit	Triggered when the upper limit of a tag is exceeded.	Non-Bool tag
Lower limit	Triggered when the low limit of a tag is exceeded.	
Change value	Triggered by a PLC or the user when the value of a tag changes	Tag
True	Triggered when a Bool tag is true.	Bool tags
False	Triggered when a Bool tag is true.	
Timeout	Triggered when the time set by a timer is reached.	Timer

Name	Description	Object
Once	Triggered once when the system time reaches a certain time set by the user. When no timer is selected, the event is triggered when the time set by the user is reached. When there is a timer, the event is triggered when the system time reaches the time set in the timer.	Scheduler
X seconds	Triggered every 0.1–60 seconds	
1 min	Triggered every minute	
1 h	Triggered once at the nth minute in every hour. The value of n ranges from 0 to 59. For example, if the time is set to 10, the event is triggered once in the 10th minute of every hour.	
Daily	Triggered once at a specific time (HH:mm) in every day. For example, if you set the time to 10:22, the event is triggered at 10:22 every day. If you have set a timer, the event is triggered when the time set by the timer is reached.	
Every week	Triggered once at a specific time (dddd HH:mm) in every day. For example,, if you set the time to Tuesday 10:22, the event is triggered at 10:22 on Tuesday every week.	
Monthly	Triggered once at a specific time (d HH:mm) in every day. For example, if you set the time to 2 10:22, the event is triggered at 10:22 on the second day of the month.	
Yearly	Triggered once at a specific time (MM-dd HH:mm) in every day. For example, if you set the time to 02-02 10:22, the event is triggered at 10:22 on February 2 every year. If you have set a timer, the event is triggered when the time set by the timer is reached.	
Screen switching	Triggered once at screen switching	
User switching	Triggered once when the user logs out.	
Login	Triggered once when the user logs in.	
Alarm buffer overflow	Triggered once when the alarm buffer overflows. The alarm buffer is limited to a specified size (512) and is not involved in configuration.	
Start	Triggered once when the runtime starts.	
Screensaver	Triggered once after screensaver is activated	

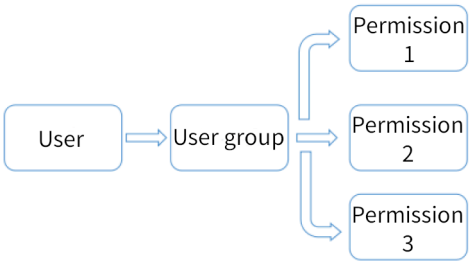
8 User Management

8.1 Overview

To prevent misopeation, different permission levels must be set. In configuration software, user management focuses on the protection of data and functions.

After creating users and user groups and assigning permissions, you can configure permissions for objects in the screen. After they are transferred to the HMI, all screen objects configured with permissions will be protected, and users will only be able to access them if they log in successfully and are authorized.

8.2 Description



In user management, permissions are not assigned directly to users, but to user groups. For example, a user named Tom is assigned to the "Commissioning Engineer" user group and granted corresponding permissions. You need not to assign permissions to each user individually.

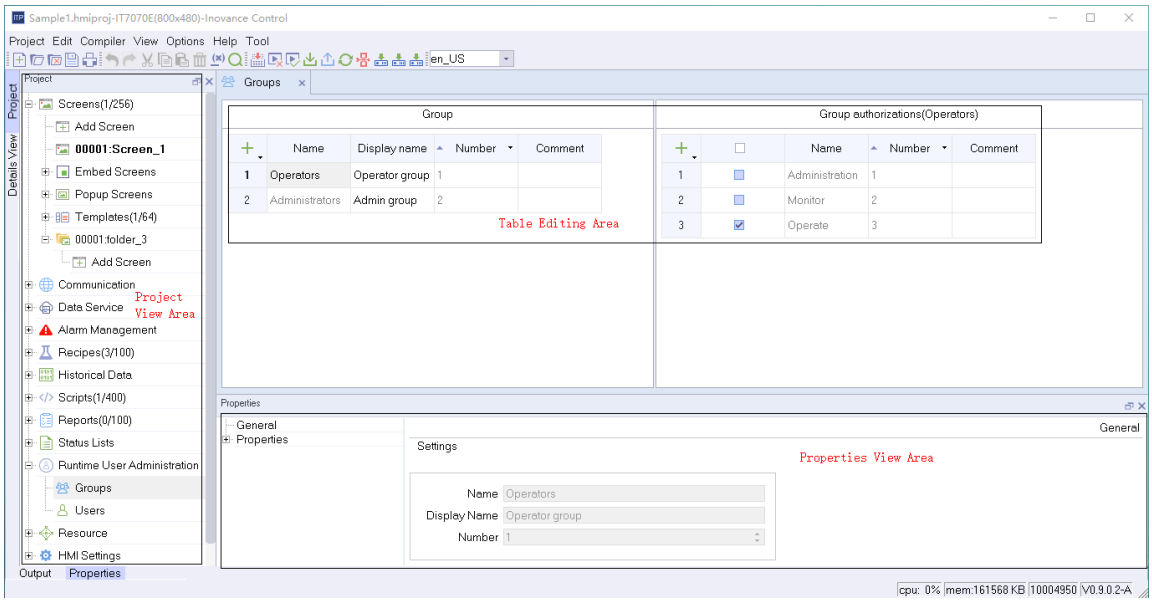
8.3 Basic Setting

8.3.1 Group Management

The group editor displays a list of user groups and their permissions. You can manage user groups and assign them permissions.

Opening the group editor

As shown in the following figure, expand Runtime User Administration and double-click Groups to open the user group editor.



Editor area

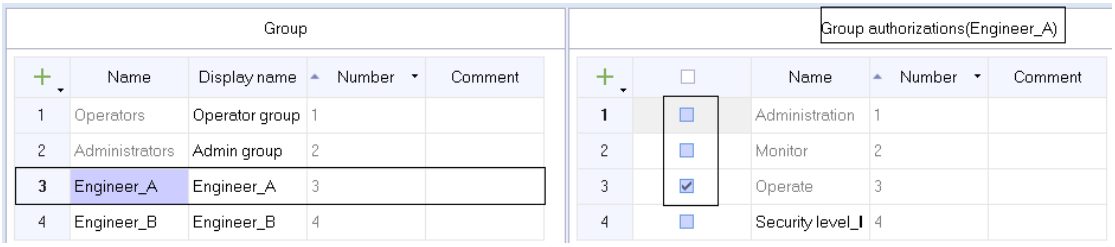
In the editor area, you can display and edit user groups and their permissions. For details on editing, see *" Use of the editors" on page 219*.

Properties view

The properties view provides the same property setting function as the editor area.

Configuration

- Groups and group permissions displayed in gray in the group editor view are automatically generated by the system and cannot be edited by users.
- The system provides two default groups: administrators and operators. An administrator has all permissions by default, and operator permissions can be set as needed.
- If you check a permission in Group Authorizations, this permission is added to that group. As shown in the following figure. The Commissioning Engineer_A group is assigned with administration, monitor and operate permissions.

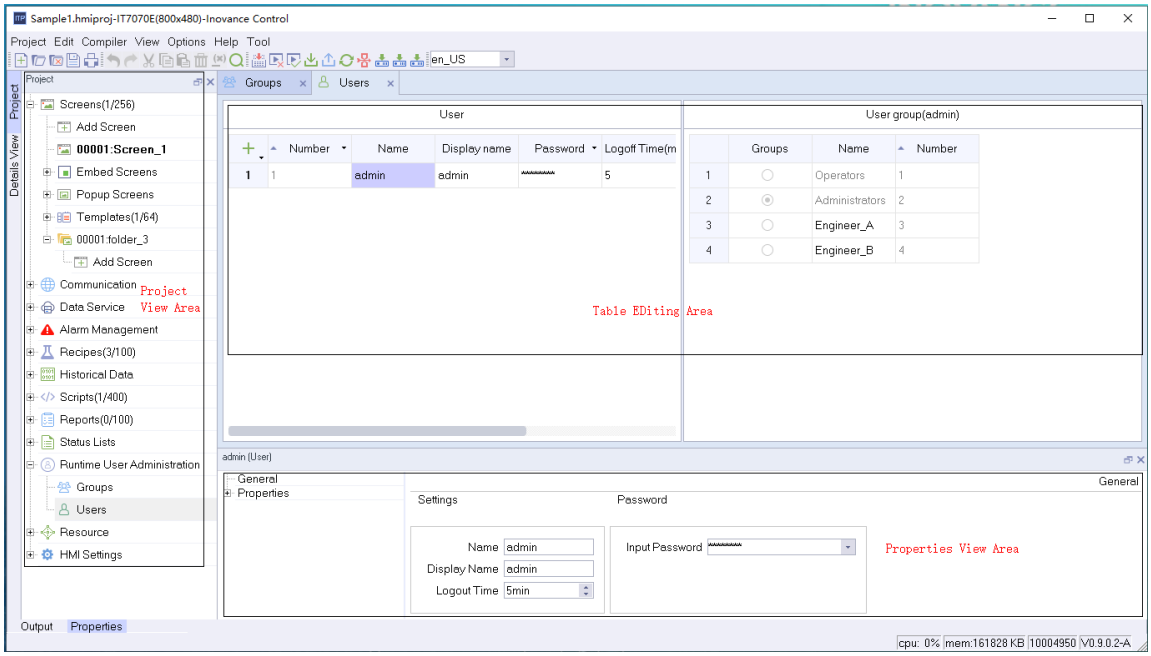


8.3.2 User Management

Users and user groups are listed in a table in the user administration editor. You can manage users and assign them to user groups here.

Opening the user administration editor

As shown in the following figure, expand Runtime User Administration and double-click Users to open the editor.



Editor area

In editor area, you can display and edit users and their groups. For details on editing, see *" Use of the editors" on page 219*.

Properties view

The properties view provides the same property setting function as the editor area.

Configuration

By default, the system has created an "admin" user which is in the administrator group (the grayed users and user groups are created by the system and cannot be edited).

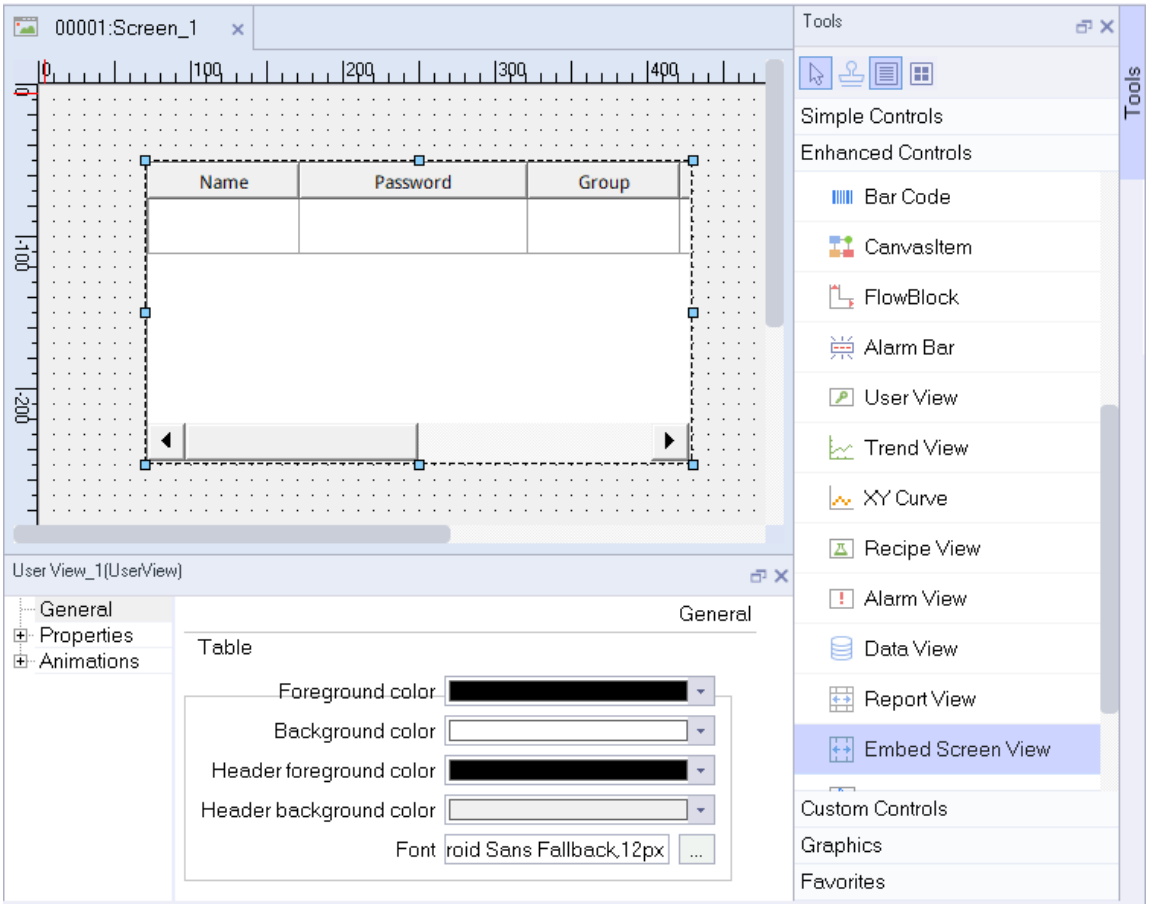
Password: The login password on the HMI. The password can be displayed as plain text or cipher text, as shown in the following figure.

Logoff Time (min): If the the user does not perform any operation after this time, the user will be automatically logged out (the defalut is 5 min). If you set it to 0, automatic logout is disabled.

8.4 Application of User View in User Management

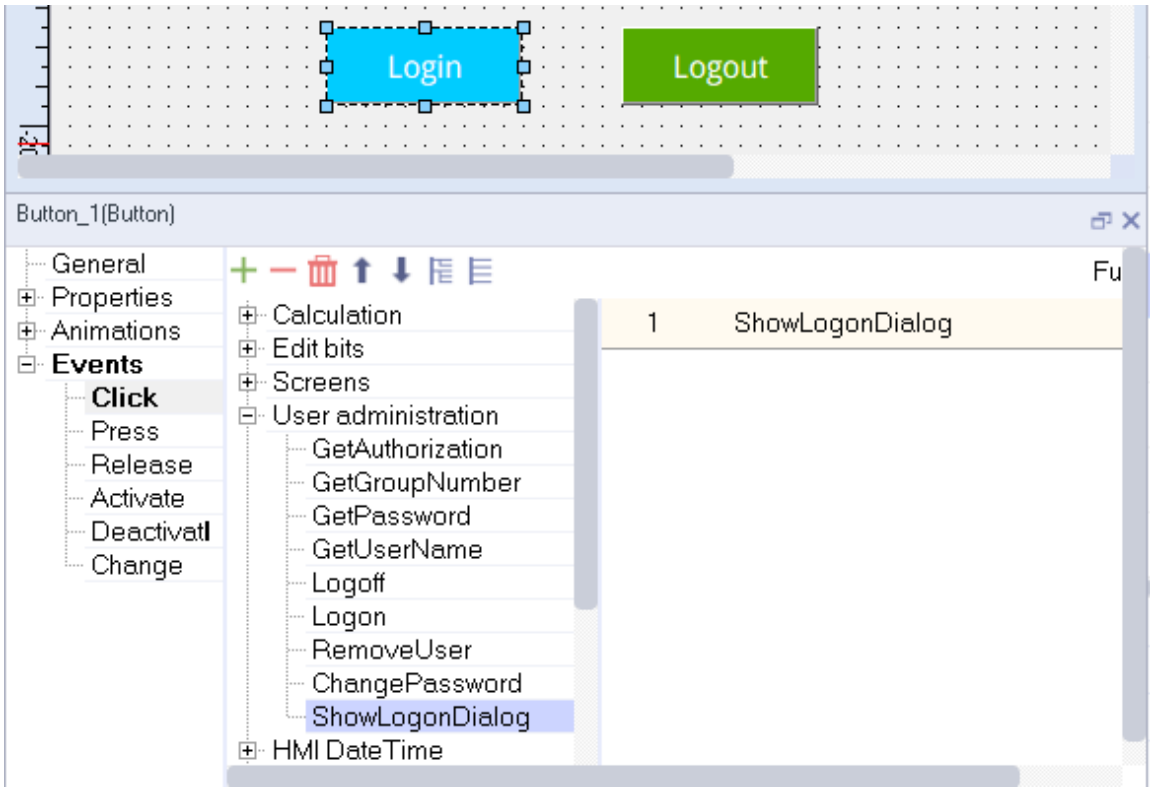
8.4.1 Creating a User View

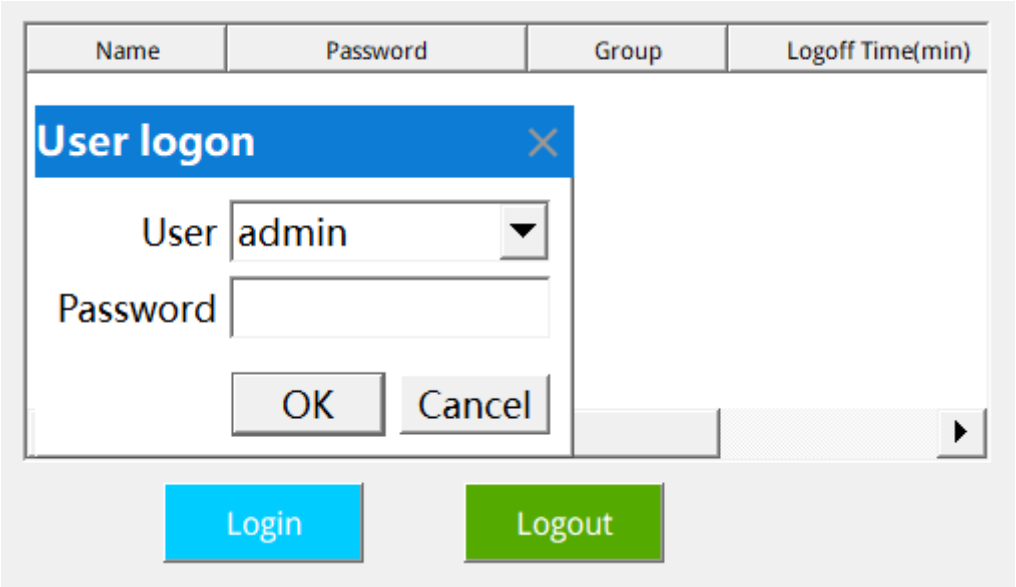
The User View is an edit window that manages users in HMI. Users with Manage permission can manage all users in User View, while other restricted users can only manage themselves.
As shown in the following figure, click "Enhanced Controls-> User View" in the toolbox on the right and drag it to the screen.



8.4.2 Configuring System Function for User Login and Logout

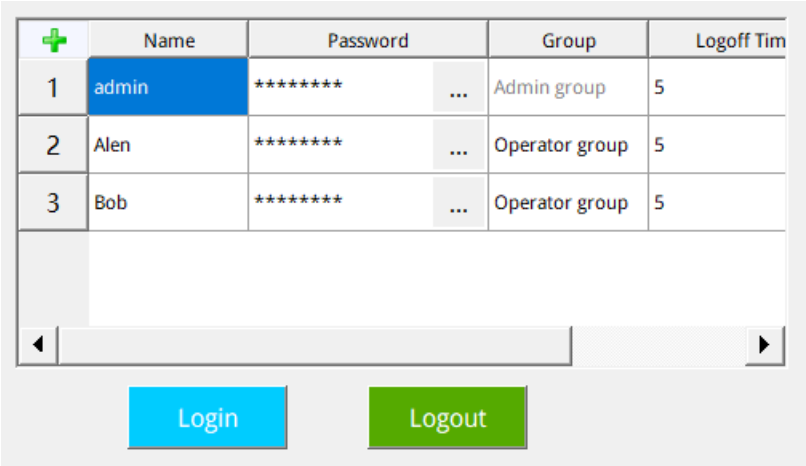
As shown in the following figure, select the Login button, and click Events->Click->User administration in the Properties view. Double-click the ShowLogonDialog function to add it to the button's click event. Then, you can click the Login button to execute the system function and display the login dialog box; In the same way, you can associate the Logoff function with the Logoff button so that you can click the button to log out.



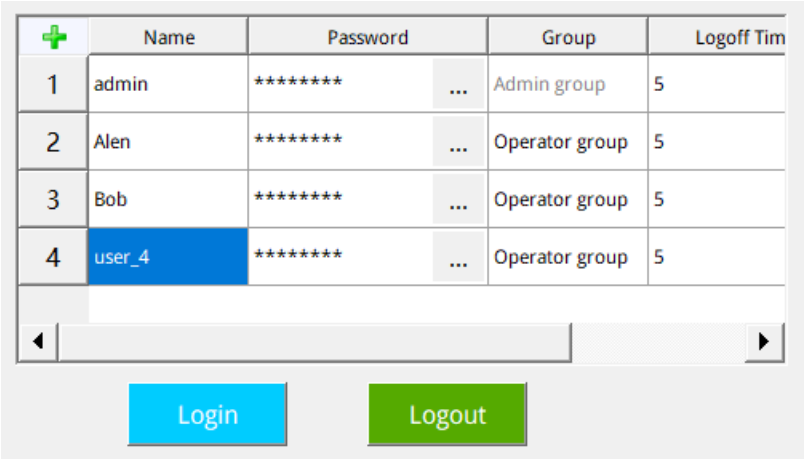


8.4.3 Using User View in HMI

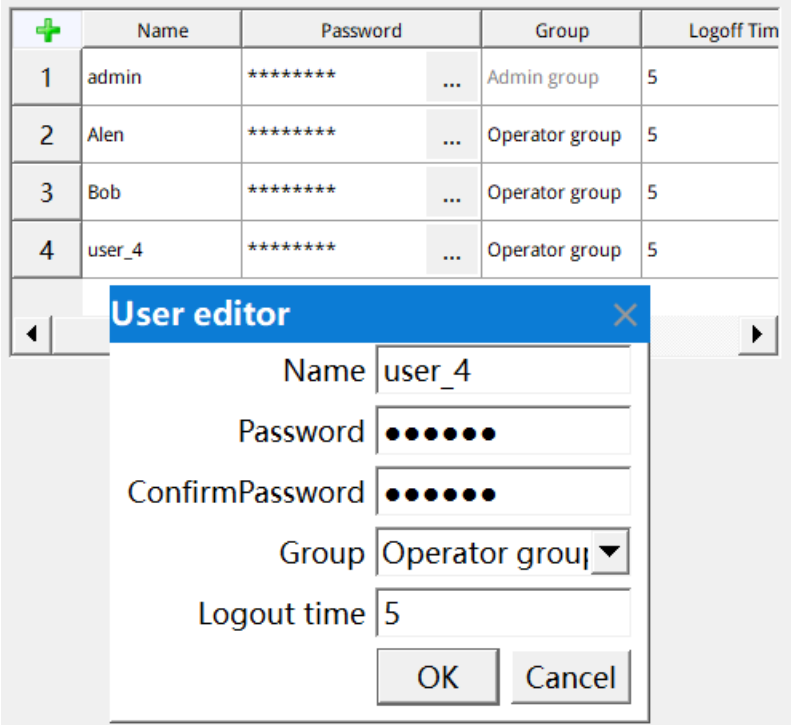
1. After configuring user management, download it to HMI. There is no data when the user view is initialized. Click the Login button, and select the designated user to log in in the pop-up User Login dialog box. After logging in, you can see your own information in the user view. After a user with administrative rights logs in, all user information will be displayed.
2. See the following figure.

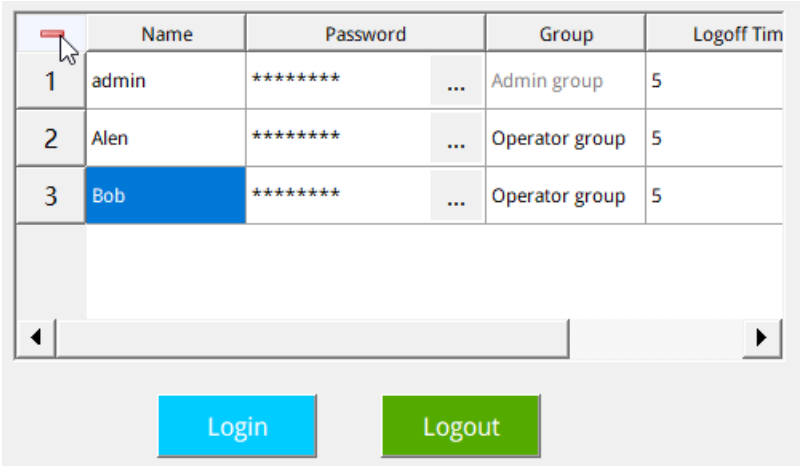


Double-click the button in the upper left corner (there is no such button if you logged in as a non-administrator) to create a new user, such as "user4".



You can double-click any non-gray cell to pop up an editing window, as shown in the following figure.

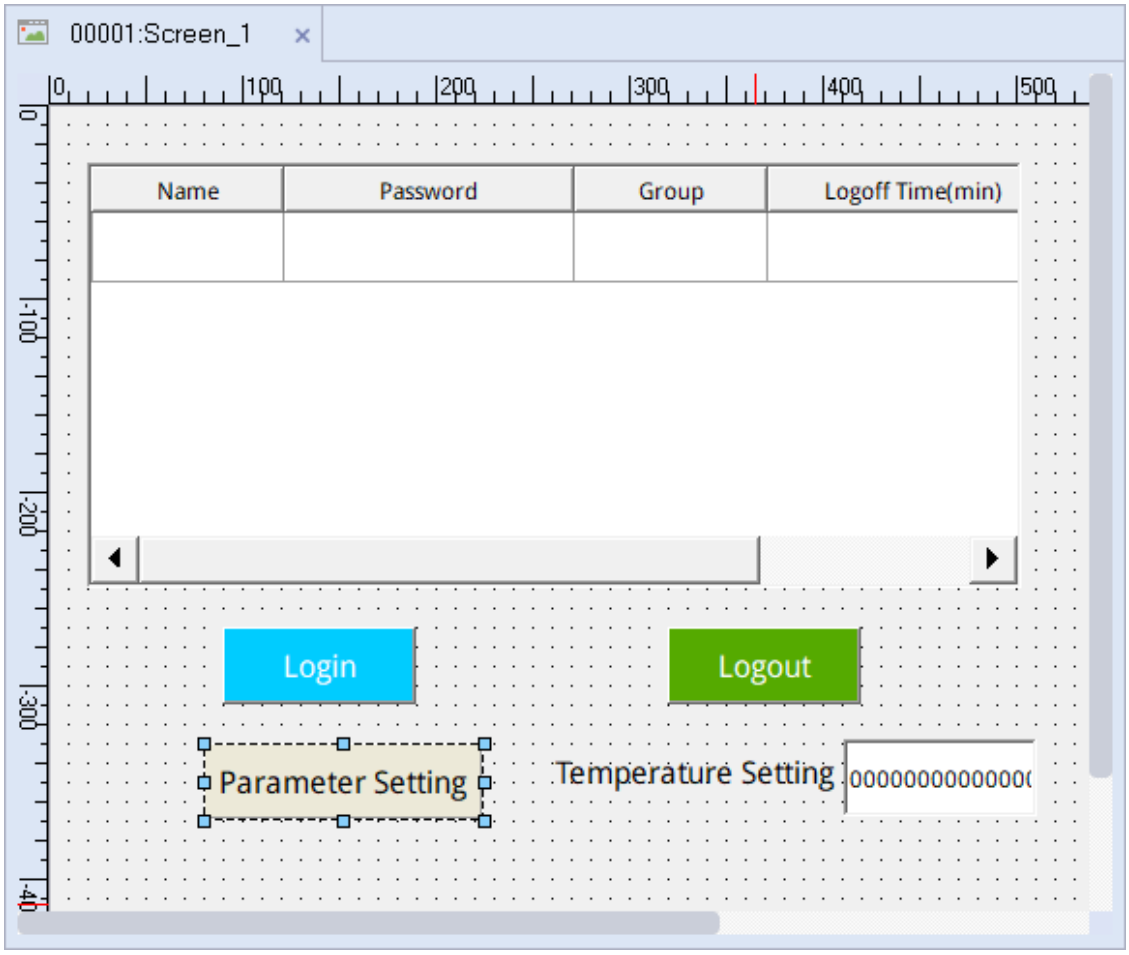




You can select a non-administrator user and double-click the button in the upper left corner to delete this user.

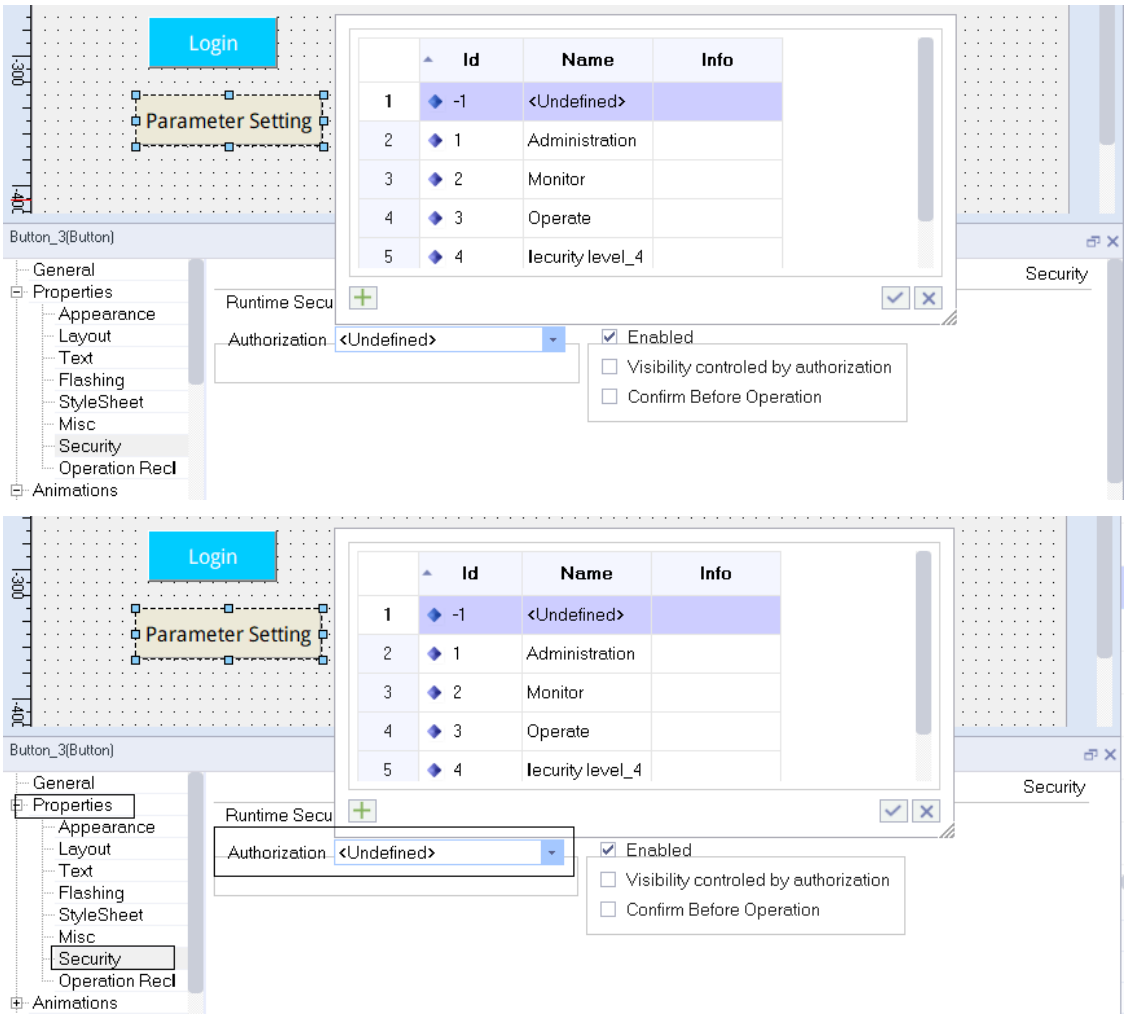
8.5 Application

1. ON Screen _ 1, add a Parameter Setting button, Temperature Setting text field and Number IO Field control, as shown in the following figure.



2. Setting permission for the Parameter Setting button and the Number IO Field control.

Select the Parameter Setting button. In the property view, go to Properties→Security, and set the authorization to Administration under Runtime Security (see *" " on page*); Similarly, set the authorization of the Digital IO Field control to Operate;



3. Turn to the HMI.

   After the project is downloaded to the HMI, when the user clicks on the controls without logging in or authorization, a login dialog box will pop up.

   For the Parameter Setting button and the Temperature Setting IO field, if you log in as an authorized user, the login dialog box will not pop up.

   Click the Logout button, and then log in as another user. If this user is only assigned the Operate permission, you are allowed to modify the temperature value. However, when you click the Parameter Setting button, the login dialog box will pop up again.

9 Alarm System

9.1 Overview

9.1.1 Visualization of Process and System Alarm

- User-defined Alarm
  - The alarm is configured to display process status on the HMI device or to measure and report process data received from the PLC.
- System alarm
  System alarms are predefined on these devices to show specific system statuses in the HMI device.

  User-defined alarms and system alarms can be triggered by a HMI device or PLC, and can be displayed on the HMI device.

9.1.2 User-defined Alarm

Available alarm process

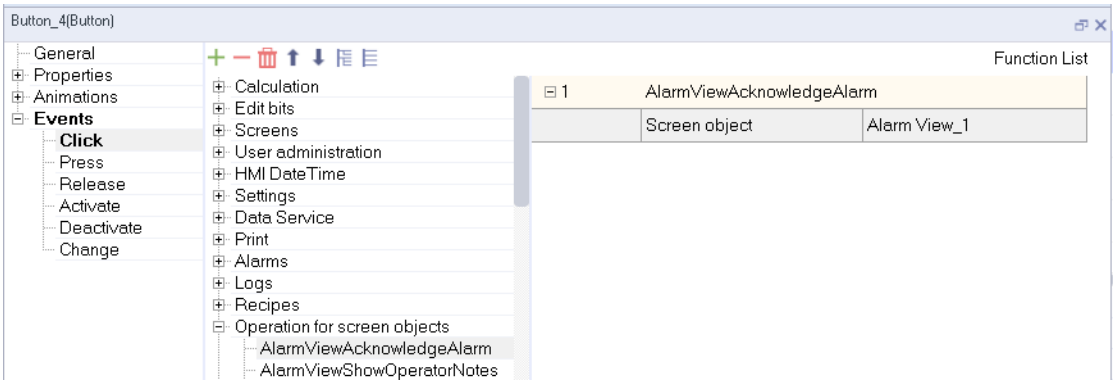
- Analog Alarm
  If the value of a tag exceeds the limit value, the HMI device triggers an alarm.
- Discrete alarm
  If the specified bit of a tag is set, the HMI device triggers an alarm.

Alarm acknowledgment

For discrete and analog alarms indicating critical or dangerous operation and process states, the equipment operator may be required to acknowledge the alarms.

Acknowledging methods

- Using the Acknowledge button, the button in the alarm view.
- Configuring the system function in the event of the screen control, as shown in the following figure:



- For discrete alarms, you can also acknowledge them by setting the specific bit in the tag.

Alarm category

Alarm category mainly determines the display mode and acknowledgment behavior of alarms on the HMI.

Predefined alarm categories

- "Error": used for discrete and analog alarms to indicate emergency or dangerous operation and process status. This kind of alarms must be acknowledged.
- "Warning": used for discrete and analog alarms, indicating general operation status, process status and process sequence. This kind of alarms needs not to be acknowledged.
- “System”: used for system alarms, prompting you about the the operating status of the HMI and PLC. This alarm category cannot be used for user-defined alarms.

9.1.3 System alarm

System alarm prompts about the operating status of the HMI and PLC. System alarm consists of an alarm number and alarm text. The alarm text accurately describes the cause.

The following table lists all the predefined system alarms:

	Text	Alarm classes	Event n...	Enabled
1	Log %1 is full.	System	100001	☑
2	Log %1 is %2 percent full.	System	100002	☑
3	Connection failure: %1, station %2.	System	100003	☑
4	Connection successful: %1, station %2.	System	100004	☐
5	Invalid input of date/time.	System	100005	☑
6	Overflow range,the valid range is [%...	System	100006	☑
7	The Medium of Log is full .	System	100007	☑
8	Tag %1 can not write to PLC.	System	100008	☑
9	Invalid PLC job number: %1.	System	100009	☑
10	No other screens can be selected. No...	System	100010	☑
11	SIM status: %1.	System	100011	☑
12	IOT status: %1.	System	100012	☑
13	Connection off line: %1, station %2.	System	100013	☑
14	%1 read error: %2.	System	100014	☑
15	%1 write error: %2.	System	100015	☑
16	SD card not been detected.	System	100016	☑
17	SD card has been detected.	System	100017	☑
18	SD card has been eject.	System	100018	☑
19	USB not been detected.	System	100019	☑
20	USB card has been detected.	System	100020	☑
21	USB has been eject.	System	100021	☑

9.1.4 Displaying Alarm

Displaying alarm on HMI

- Alarm view
  Depending on the configuration size of the alarm view, multiple alarms can be displayed at the same time. You can select the alarm category to display in the alert view, and you can configure multiple alarm views for different alarm categories.
- Alarm bar
  The alarm bar can display all error alarms.
- System alarm window
  The system alarm window is an alarm window provided by the runtime to display system events.

Logging alarms

After alarm logging is configured for an alarm category, all alarms of this alarm category will be recorded in the specified alarm log.

System functions for editing alarms

AlarmViewAcknowledgeAlarm	Acknowledge the alarm selected in the specified alarm view
AlarmViewShowOperatorNotes	Display the info text of the alarm selected in the specified alarm view
ClearAlarmBuffer	Delete alarms from the HMI alarm buffer
ClearActionedAlarmBuffer	Delete alarms that have been dealt with from HMI alarm buffer
ShowAlarmView	Show\Hide the specified alarm view control
Beep	Play a buzzer

9.2 Properties and Basic Settings

9.2.1 Alarm Properties

Alarm Properties

- Alarm text
  Alarm text is the description of the alarm. You can configure a tag in the text through the drop-down box or drag-and-drop of a tag.
- Number
  The alarm number is used to identify an alarm, and is unique in the following alarm types.
  - Analog Alarm
  - Discrete alarm
  - System alarm
- Alarm class
  The alarm class determines whether the alarm requires acknowledgment. It also determines the display mode of alarms on the HMI.

  It can also determines whether to log the alarm or not.
- Trigger tag
  - For analog alarm: triggers an alarm when the tag value reaches the limit value
  - For discrete alarm: triggers an alarm when a bit in the tag is set

Optional alarm properties

- Alarm group
  Handled altogether. That is, if the alarm belongs to a alarm group, all the alarms in the group can be acknowledged when you acknowledge it.
- Info text
  Additional information about the alarm. The text will not be displayed automatically. You must select the alarm and press the button in the alarm view to display it.

Analog Alarm

- Limits
  The value used to trigger the tag is compared with this value to determine whether to trigger the alarm
- **Trigger mode**
  The trigger modes include >, <, ==, =, and <=, which are used as the condition for comparing the value of the trigger tag with the limit value
- Hysteresis
  Hysteresis value, ranging from 0 to 100, used with percentage.
- Hysteresis in percent
  When it is on, the hysteresis value becomes percentage, which is equal to limit value x hysteresis%

  When it is off, the hysteresis value = the value in the Hysteresis field
- Use
  On "activated": value for triggering alarm = limit value + hysteresis value

  On "deactivated": value for triggering alarm = limit value – hysteresis value

  The sign of the hysteresis value is determined by the trigger mode:

  When the trigger mode is >, =, and ==, the hysteresis value is positive.

  When the trigger mode is< and ==, the hysteresis value is positive.
- Delay
  When the trigger condition is met, the analog alarm is triggered after the specified time (in 100 ms). For example, if you set the delay to 10, the delay time = 10 * 100 ms.

Discrete alarm

- Trigger bit
  The state of a bit of a trigger tag is used as the trigger condition. The trigger bit can be the same regardless of the trigger mode for the trigger tag.
- **Trigger mode**
  The trigger modes include 1->0, 0->1, ==0 and ==1
- Ack PLC tag and Ack PLC bit
  The PLC bit number designates the n-th bit of the PLC tag as the acknowledgment bit. You can reset a bit of the PLC tag to acknowledge the corresponding alarm. The Ack PLC tag can only be the same as the trigger tag of the corresponding alarm, with different bit numbers.
- Ack HMI tag and Ack HMI bit
  The HMI bit number designates the n-th bit of the HMI tag as the acknowledgment bit. When the alarm is triggered, the specified bit of the HMI tag is reset. When the corresponding discrete alarm is acknowledged, the specified bit of the HMI tag is set.

  Note: The Ack PLC tag and Ack HMI tag can only be used when the discrete alarm triggers the external tag configured by the trigger tag.

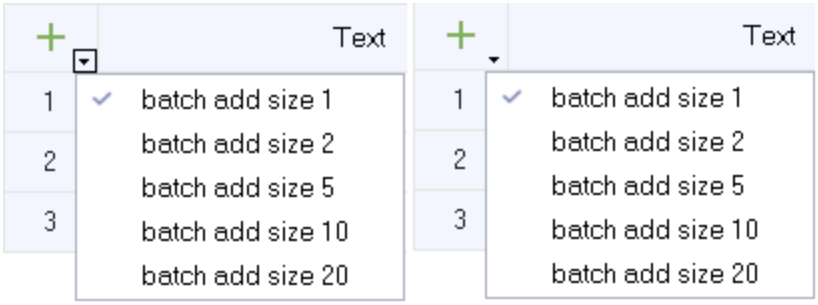
9.2.2 Alarm Configuration Editor

Editor types

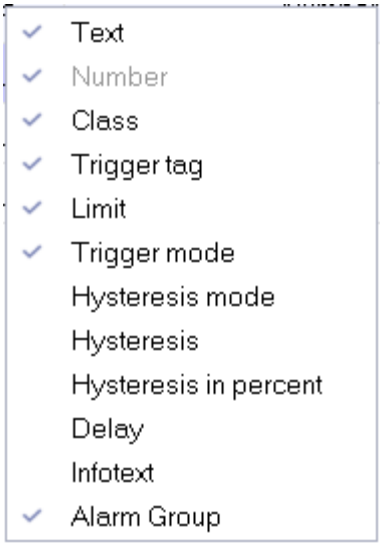
- Discrete alarm editor for creating and modifying discrete alarms
- Analog alarm editor for creating and modifying analog alarms
- System alarm editor for modifying the alarm text of system alarms
- Alarm class editor for creating and changing alarm categories
- Alarm group editor for creating and modifying alarm groups

Use of the editors

- Creating an object: Click + in the upper left corner of the table;
- Batch creating: Click the triangle under the Add button and select an operation in the drop-down menu. Then click the Add button;



- Deleting an object: Click the row number of the alarm row to be deleted, and the Add button becomes the Delete button −. Click the button to delete the row;
- Hiding and displaying column fields: Right-click the column header, and select the fields to be hidden or displayed in the drop-down menu.



- Column sort: Click the column header

9.2.3 Analog Alarm Editor

You can create analog alarms and specify their properties in the analog alarm editor. For property setting, see *"9.2.1 Alarm Properties" on page 217*.

Opening the analog alarm editor

Click Alarm Management in the project view as shown below, and then double-click Analog Alarms to open the editor.

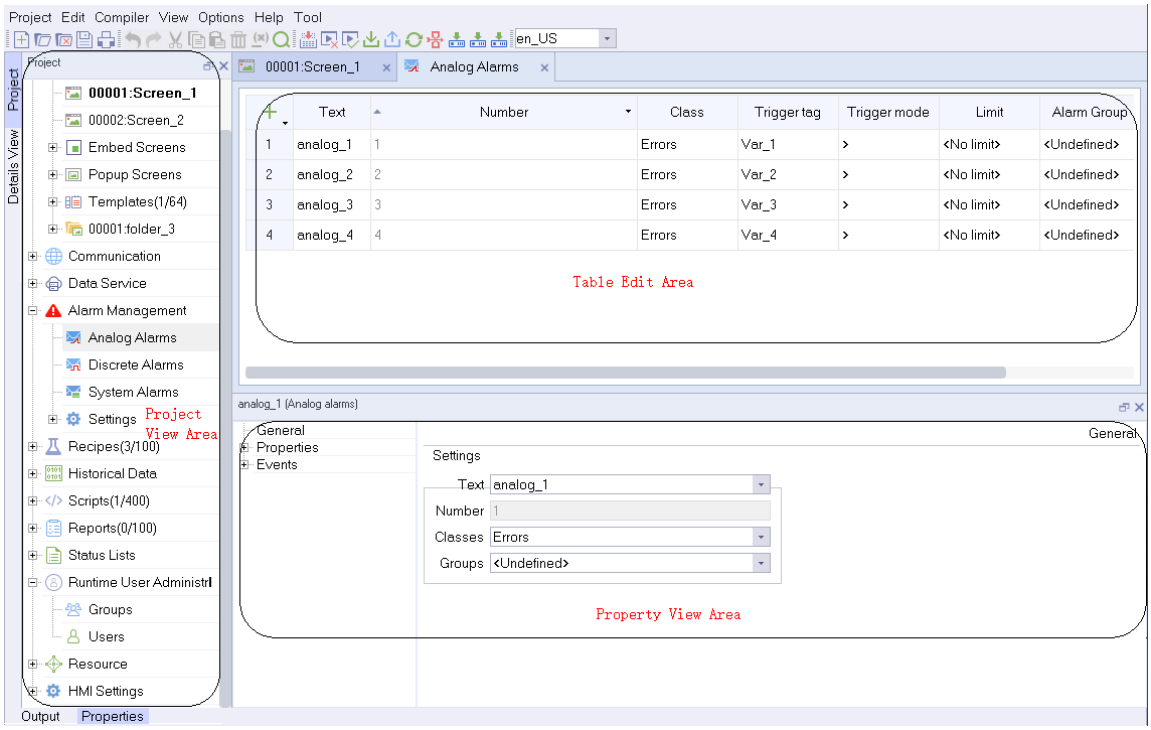


Figure 9-1 Analog alarm editor

Editor area

The editor area displays all created analog alarms and their property settings in a table. For table editing, see *" Use of the editors" on page 219*.

Properties view

The properties view provides the same property setting function as the editor area. In addition, It also provides event editing.

Description of event

Activate: When the alarm occurs, the script function configured in the 'Activate' event will be called.

Deactivate: When the alarm is deactivated, the script function configured in the 'Deactivate' event will be called.

Acknowledge: When the alarm is acknowledged, the script function configured in the 'Acknowledge' event will be called.

9.2.4 Discrete Alarm Editor

You can create discrete alarms and specify their properties in the discrete editor. For property setting, see *"9.2.1 Alarm Properties" on page 217*.

Opening the discrete alarm editor

Click Alarm Management in the project view as shown below, and then double-click Discrete Alarms to open the editor.

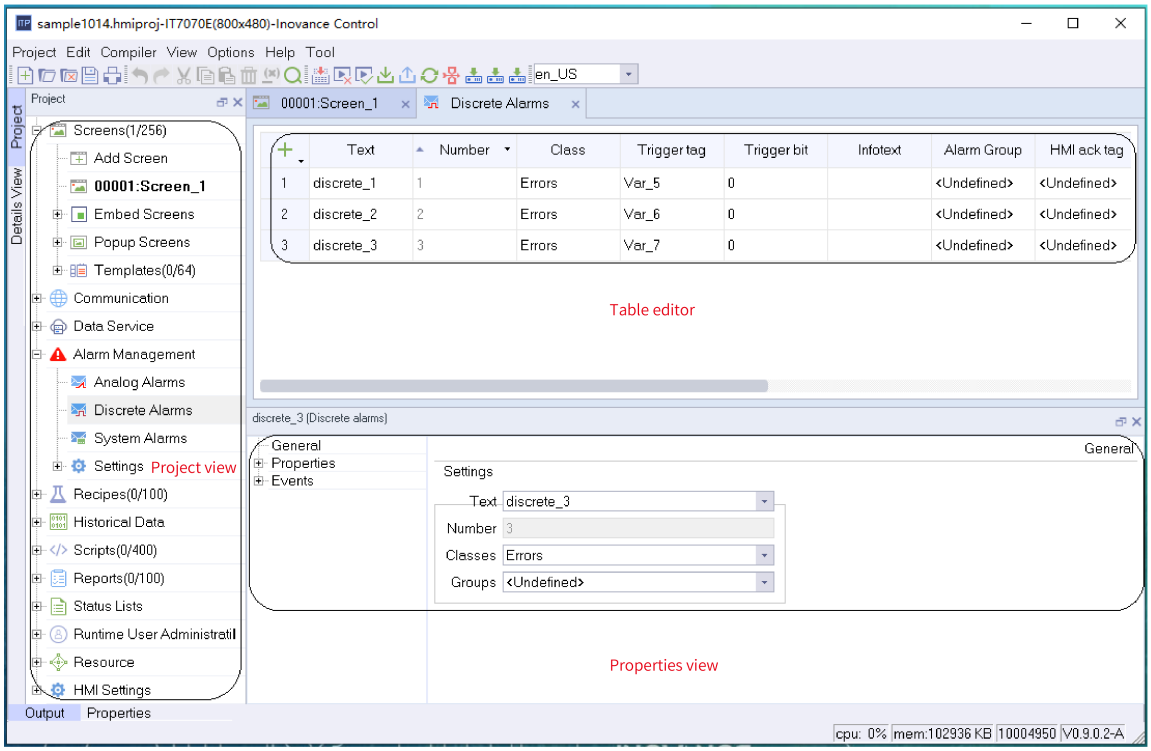


Figure 9-2 Discrete alarm editor

Editor area

The editor area displays all created discrete alarms and their property settings in a table. For table editing, see *" Use of the editors" on page 219*.

Properties view

The properties view provides the same property setting function as the editor area. In addition, It also provides event editing.

Description of event

Activate: When the alarm occurs, the script function configured in the 'Activate' event will be called.

Deactivate: When the alarm is deactivated, the script function configured in the 'Deactivate' event will be called.

Acknowledge: When the alarm is acknowledged, the script function configured in the 'Acknowledge' event will be called.

9.2.5 System Alarm Editor

In the system alarm editor, you can only view all defined system alarm information.

Opening the system alarm editor

Click Alarm Management in the project view as shown below, and then double-click System Alarms to open the system alarm editor.

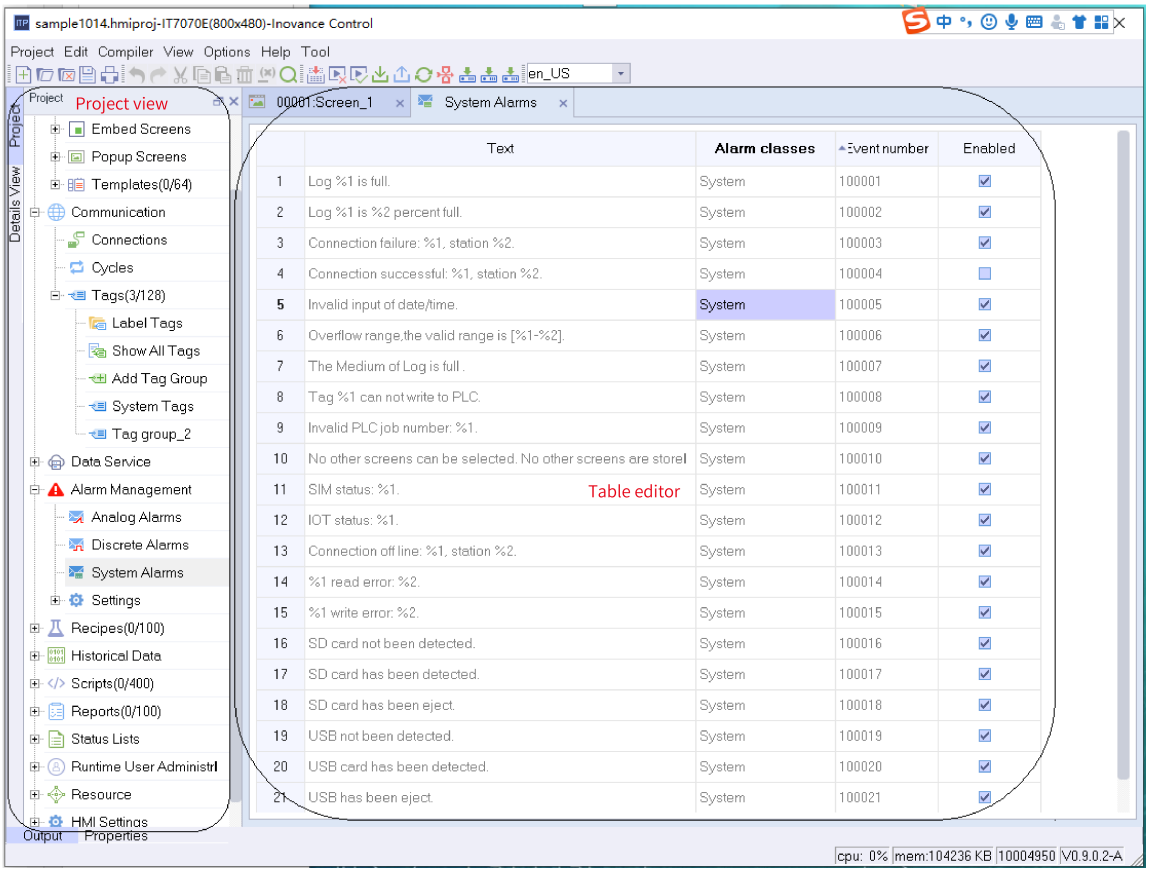


Figure 9-3 System alarm editor

Editor area

All defined system alarm information is displayed in the editor area. You cannot modify the system alarm attribute, but you can decide whether to enable this alarm by ticking the check box in the Enabled column.

9.2.6 Alarm Classes Editor

You can create alarm categories and specify their properties in the editor.

Click Alarm Management→Settings→Alarm Classes as shown in the following figure to open the editor.

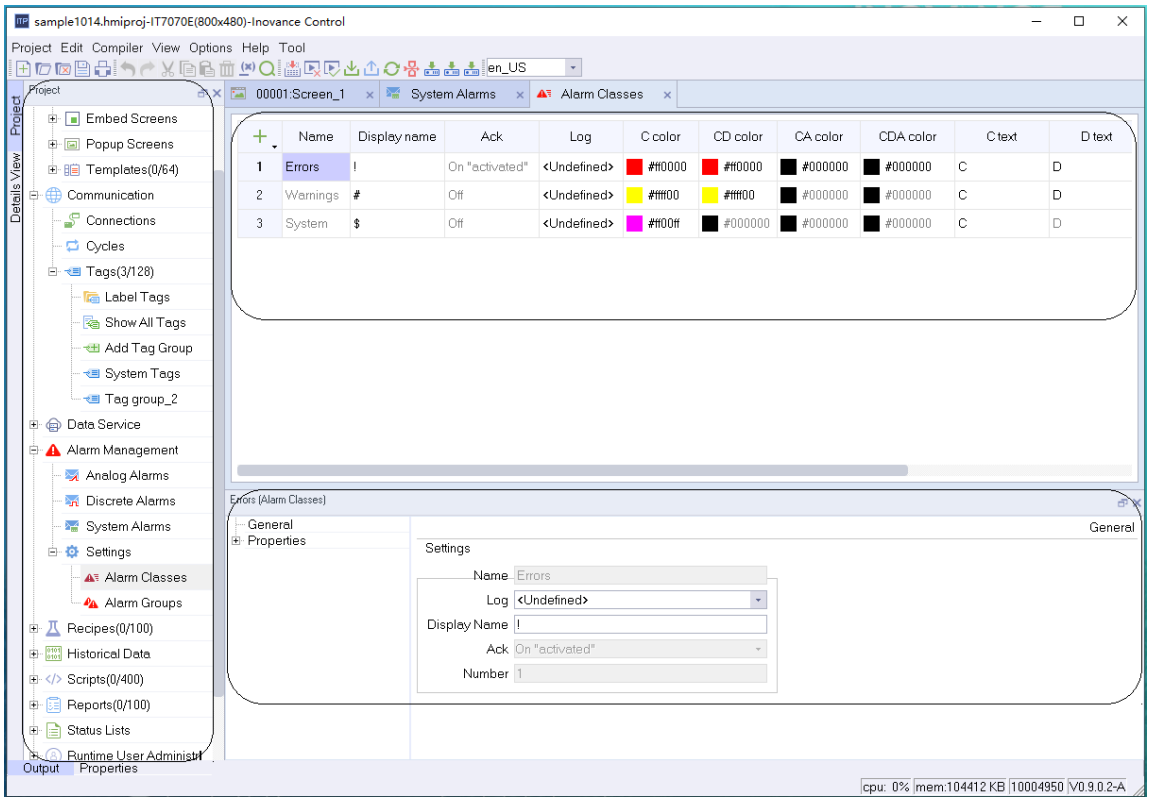


Figure 9-4 Alarm classes editor

Editor area

The editor area displays all created alarm categories and their property settings. You can edit the property in the cell.

Properties view

The properties view provides the same property setting function as the editor area

Properties

- Name: category name. The name is unique.
- Display name: The mark displayed under the Type of the alarm view control when the alarm is triggered. Different alarm categories can have the same display name.
- Ack: Sets whether the alarm needs to be acknowledged.
- Log: Sets the location of alarms and logs, and only the logs in Alarm Logs can be configured.
- C color: When the alarm text is displayed in this color, it indicates that the alarm is active.
- CD color: When the alarm text is displayed in this color, it indicates that the alarm has occurred and been eliminated.
- CA color: When the alarm text is displayed in this color, it indicates that the alarm has occurred and been acknowledged.
- CDA color: When the alarm text is displayed in this color, it indicates that the alarm has occurred and been acknowledged and eliminated.
- C text: The alarm text displayed after an alarm occurs.
- D text: The alarm text displayed after an alarm is eliminated.

- A text: The alarm text displayed after an alarm is acknowledged.

Caution

The alarm status text is composed of "Arrived Text", "Departed Text" and "Confirmed Text". As shown in Figure 6-2-6 above, when an alarm occurs, the status text of the alarm is displayed as "C", and if the alarm is confirmed, the status text is displayed as "CA".

9.2.7 Alarm Group Editor

In the alarm group editor, you can create alarm groups and specify their attributes. Alarm groups are mainly used to classify multiple alarms into one group. In the alarm view, select any alarm in the alarm group to confirm, and all the alarms generated by this group can be confirmed.

Opening the alarm group editor

Click Alarm Management→Settings→Alarm Groups as shown in the following figure to open the system alarm editor.

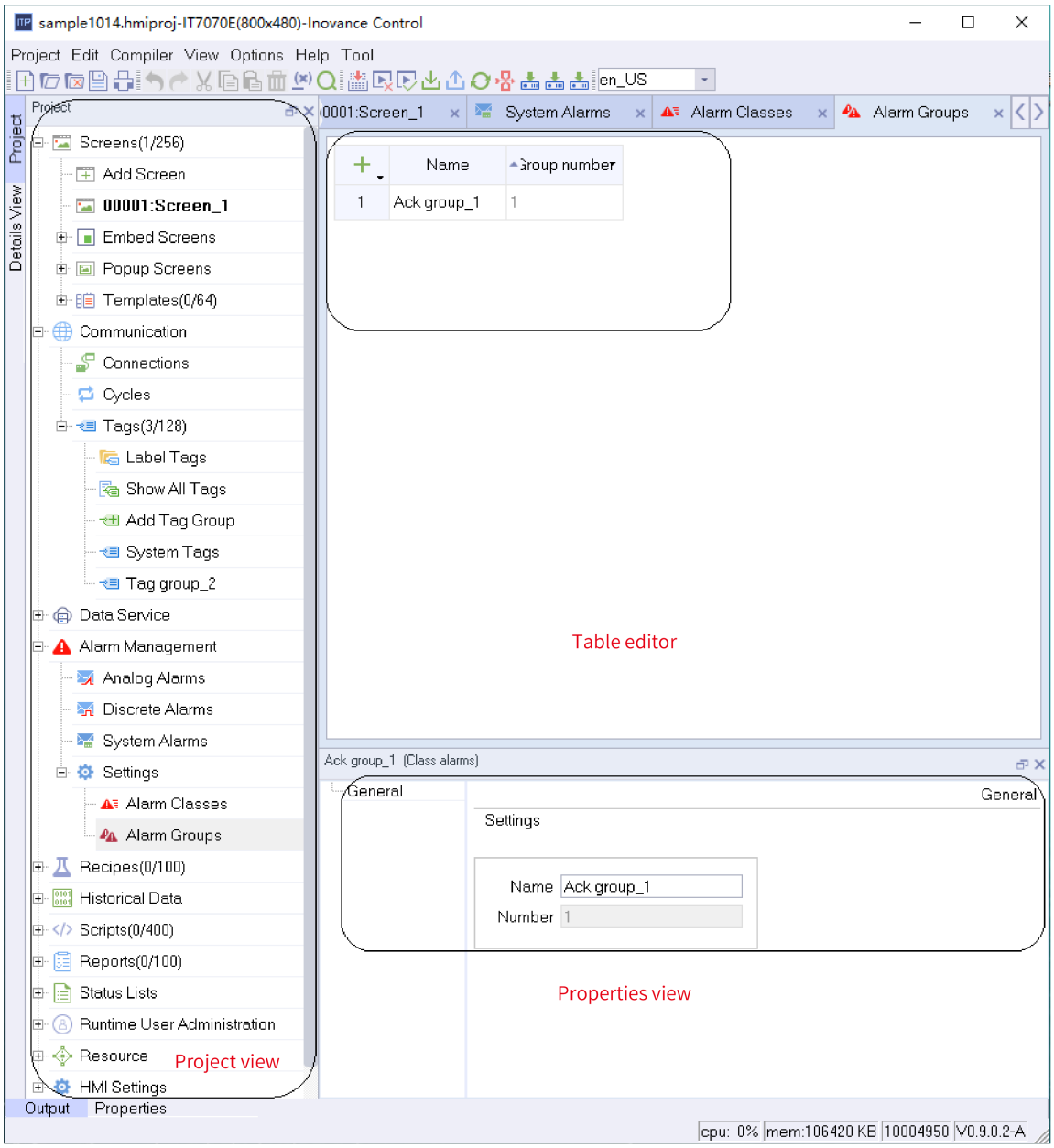


Figure 9-5 Alarm group editor

Editor area

The editor area displays all created alarm groups and their property settings. You can edit the property in the cell.

Properties view

The properties view provides the same property setting function as the editor area.

9.3 Alarms log

9.3.1 Description

The alarm record saves the alarm information generated during the project operation for later problem analysis.

9.3.2 Contents of Alarm Record

The alarm to be recorded is assigned to the alarm record by alarm category. When the alarm status changes, the alarm record will hold the following information for the corresponding alarm:

- Alarm text
- Alarm category
- Alarm No.
- Date and time
- Alarm state
- Confirmation group
- PLC: alarm trigger tag connection name

An alarm record will be generated when the alarm occurs, the alarm is confirmed and the alarm is eliminated.

9.3.3 Alarm Log Editor

You can create and edit the properties of an alarm log in the alarm log editor. For details, see *"9.3.4 Basic Settings of Alarm Log" on page 228*.

Opening the alarm log editor

To open the alarm log editor, expand Historical Data, and double-click Alarm Logs.

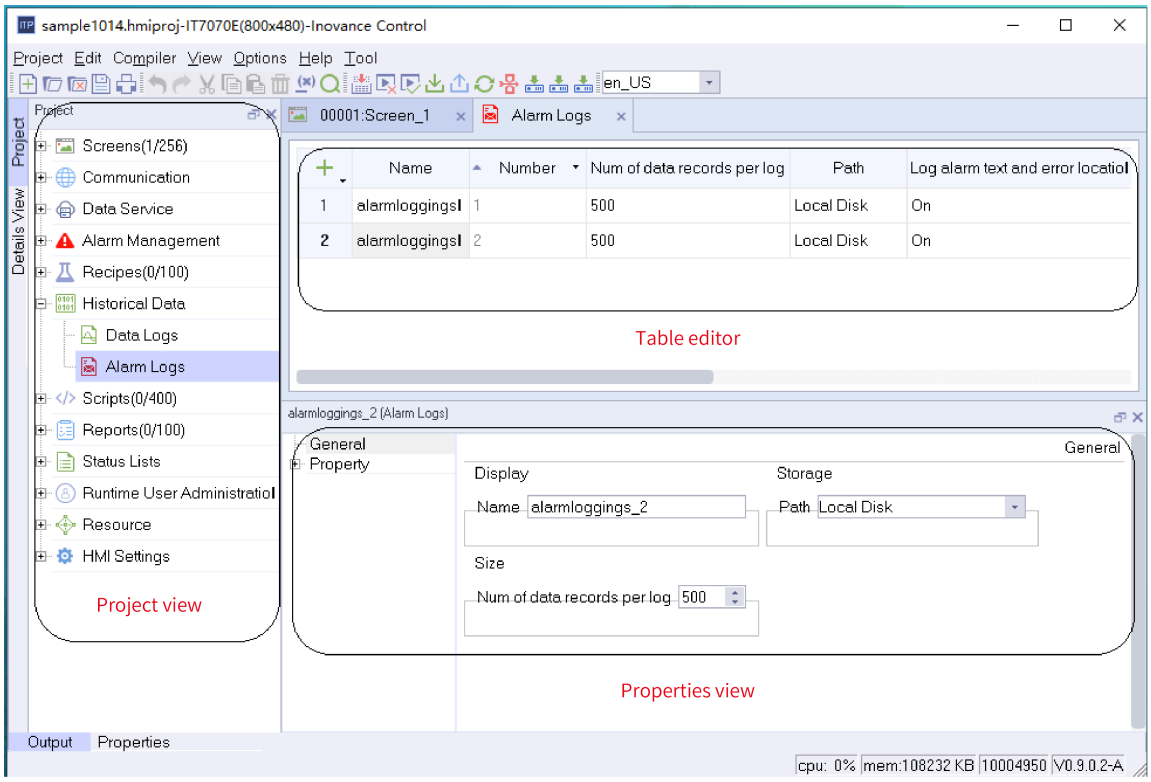


Figure 9-6 Alarm log editor

Editor area

- Creating an object: Click + in the upper left corner of the table;
- Batch creating: Click the triangle under the Add button and select an operation in the drop-down menu. Then click the Add button;

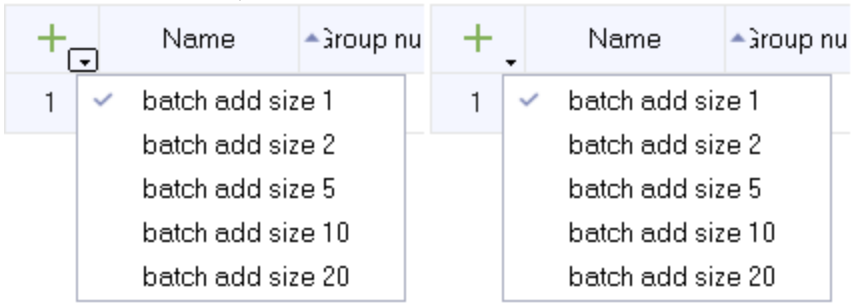


Figure 9-7 Batch creating options

- Deleting an object: Click the row number of the alarm row to be deleted, and the Add button becomes the Delete button −. Click the button to delete the row;
- Hiding and displaying column fields: Right-click the column header, and select the fields to be hidden or displayed in the drop-down menu.

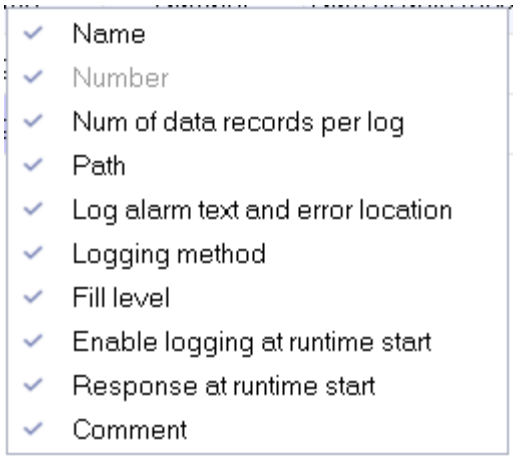


Figure 9-8 Column attributes

- Column sort: Click the column header

Properties view

The properties view provides the same property setting function as the editor area. In addition, It also provides event editing.

Description of event

Overflow event: The script function that triggers configuration when the alarm logs overflow.

Event items can only be used when triggering events in the Logging Method configuration.

9.3.4 Basic Settings of Alarm Log

Property setting

- Name
  Name of the alarm log
- Num of data records per log
  The maximum number of alarms allowed to be recorded per alarm log.
- Path
  The storage location of the alarm log, which can be the HMI or a USB drive or SD card.
- Log alarm text and error location
  Choose whether to save the alarm text and the connection name used by the alarm trigger tag.
- Logging method
  - Circular log: When the table is full, the earliest log will be overwritten.
  - Raise event: When the table is full, the script function configured in the overflow event is triggered.
  - Display system alarm: When the number of recorded alarms reaches the specified maximum number, the system alarm is displayed.
- Fill level

The maximum number of logs in percentage to display the system alarm.

- Enable logging at runtime start
  Choose whether the alarm logging function is enabled when HMI is started.
- Response at runtime start
  - Append data to existing log: Keep the previous alarm logs when HMI starts.
  - Reset log: Clear the previous alarm logs when HMI starts.
- Comment
  Description of the log

Displaying Alarm Log

To display alarm logs on the HMI, you need to select the data log to be displayed in the setting of the alarm view control. As shown in the following figure.

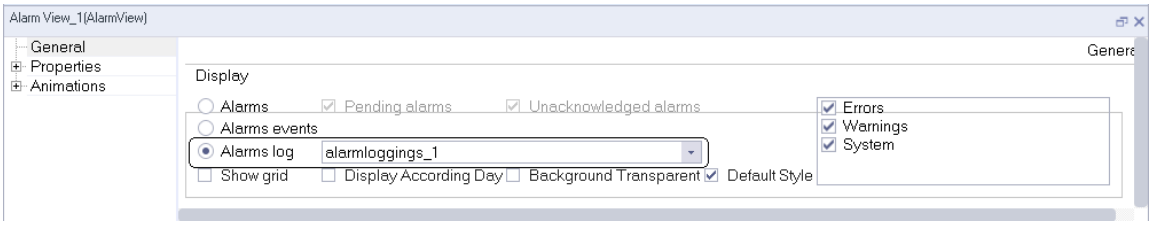
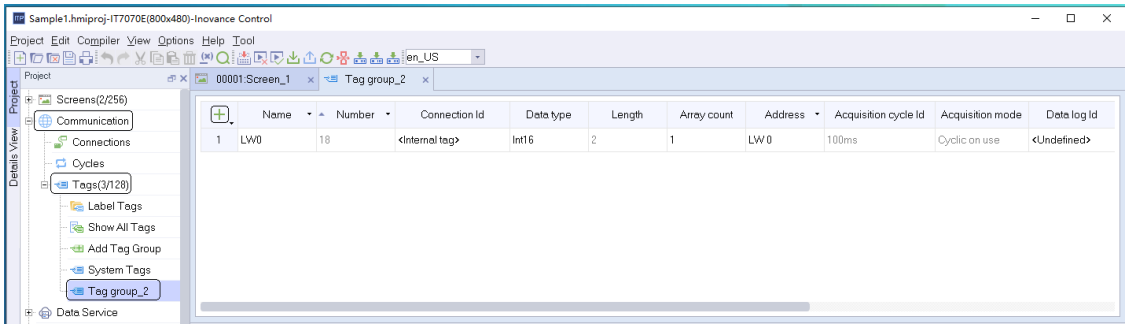


Figure 9-9 Alarm log configuration

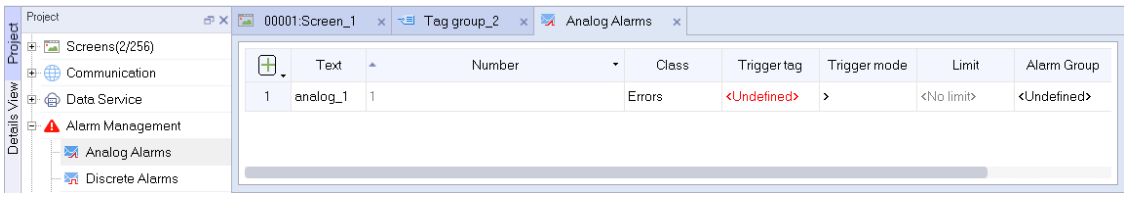
9.4 Using Alarms

9.4.1 Creating an Analog Alarm

1. Add a numeric tag to tag group_2.



2. Create an analog alarm item.



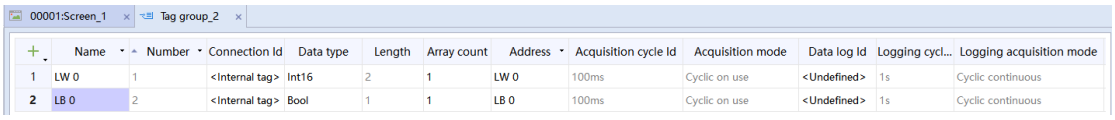
3. Set the trigger tag to LW 0, limit to 100, and trigger mode to >.

	Text	Number	Class	Trigger tag	Trigger mode	Limit	Alarm Group
1	analog_1	1	Errors	LW_0	>	<No limit>	<Undefined>

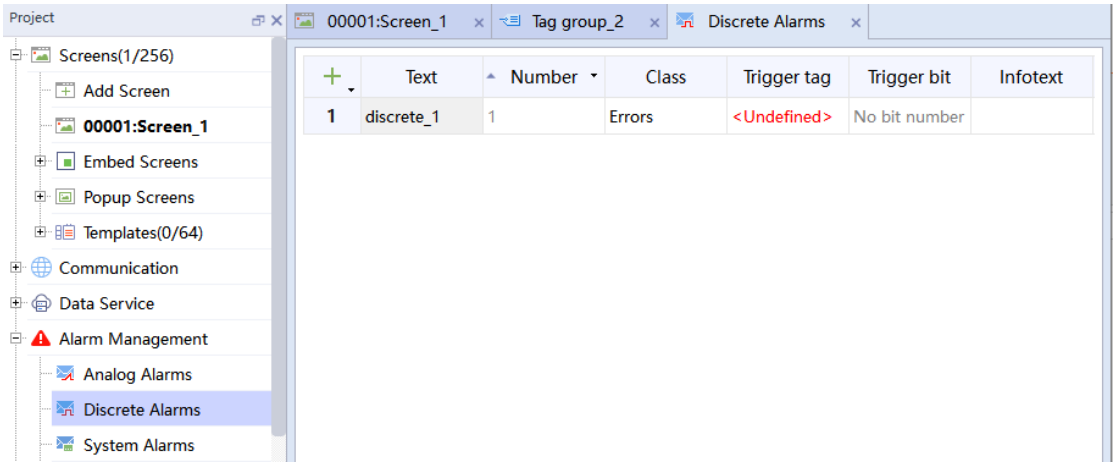
During operation if you set LW 0 to a value greater than 100, the analog_1 alarm will be triggered.

9.4.2 Creating a Discrete Alarm

- Add a Boolean tag to tag group_2.



- Create a discrete alarm item.



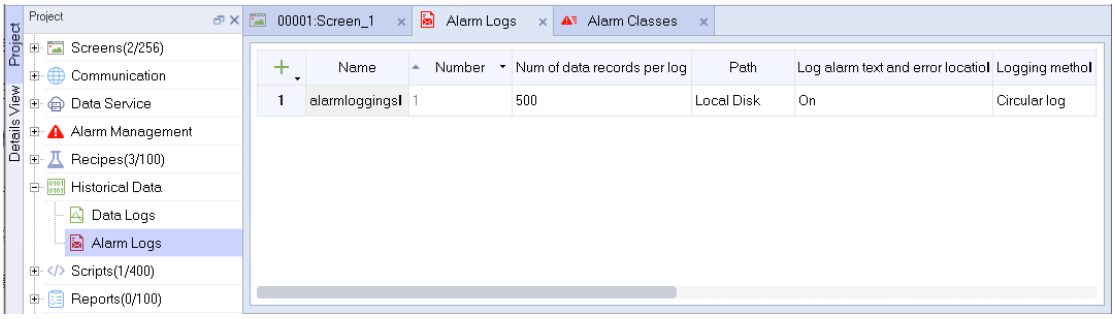
- Set trigger tag to LB 0.

	Text	Number	Class	Trigger tag	Trigger bit	Infotext
1	discrete_1	1	Errors	LB 0	0	

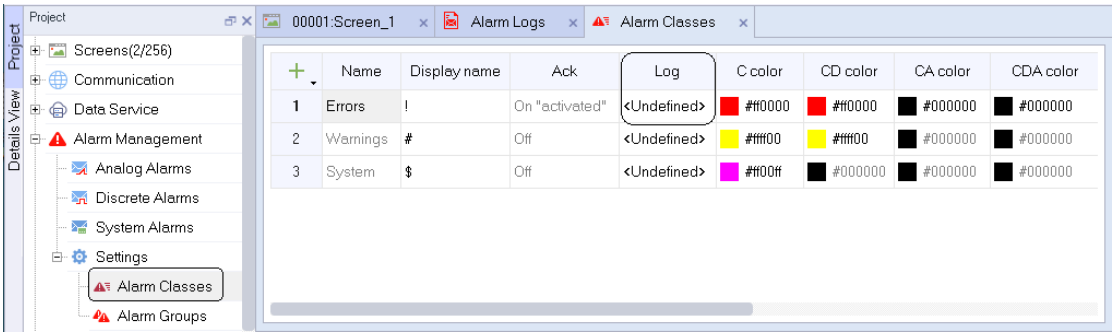
During operation, if you set LB 0, the discrete_1 alarm will be triggered.

9.4.3 Using Alarm Record

- Creating an alarm record



- Using Alarm Record
  Select the alarm information of the specified category in the alarm category and record it in the alarm record.



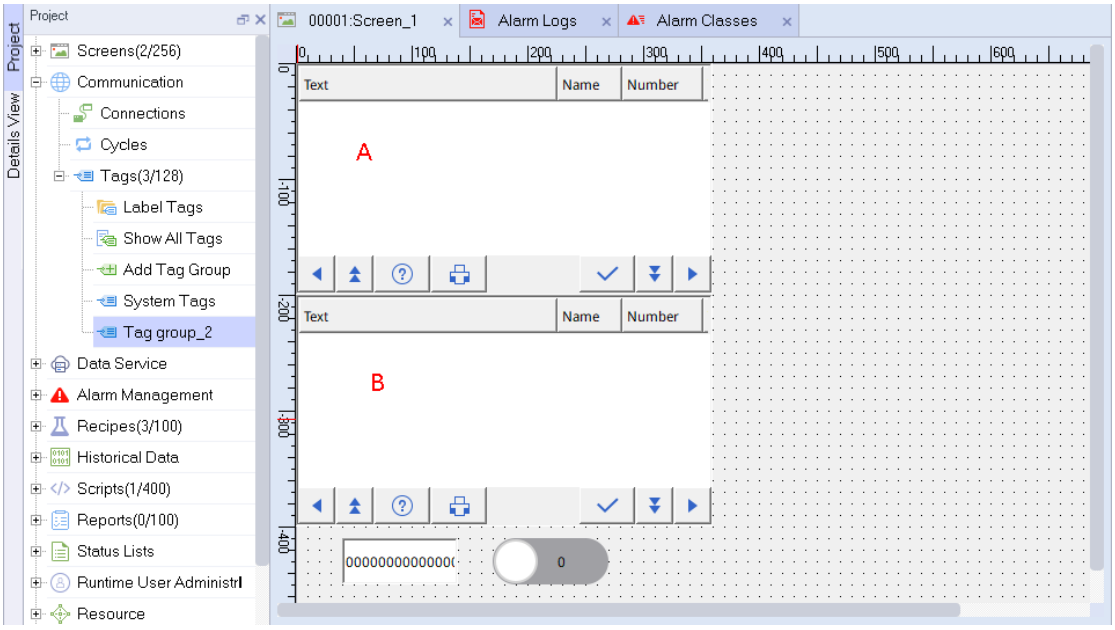
In the following figure, an "error" alarm is set to be recorded in alarm record_1.

	Name	Display name	Ack	Log	C color	CD color	CA color	CDA color
1	Errors	!	On "activated"	alarmloggin	#ff0000	#ff0000	#000000	#000000
2	Warnings	#	Off	<Undefined>	#ffff00	#ffff00	#000000	#000000
3	System	$	Off	<Undefined>	#ff00ff	#000000	#000000	#000000

Whenever an "error" alarm occurs, it will be kept in alarm record_1.

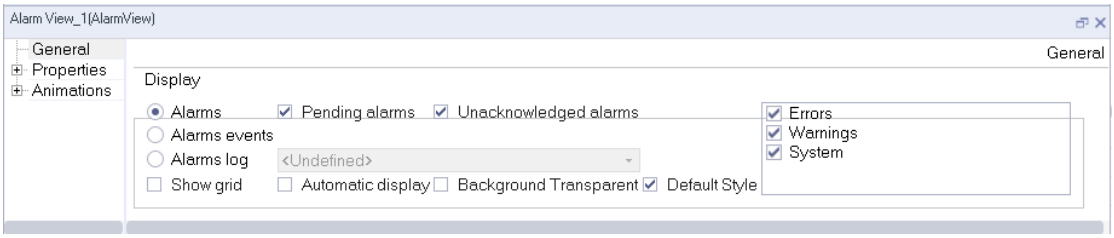
9.4.4 Displaying Alarm

- After configuring the alarm, add an alarm view control in the screen to display the alarm.

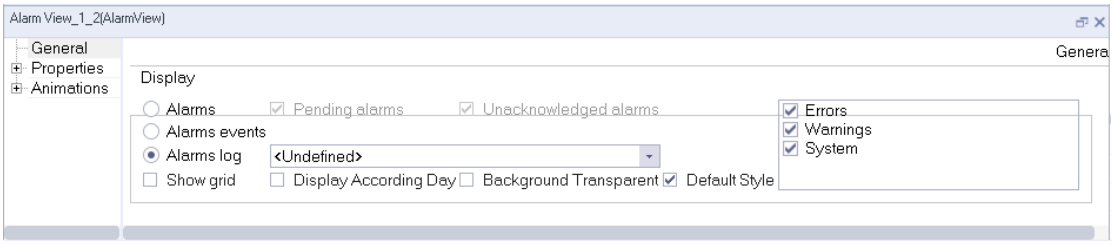


As shown in the figure, add two alarm views A and B.

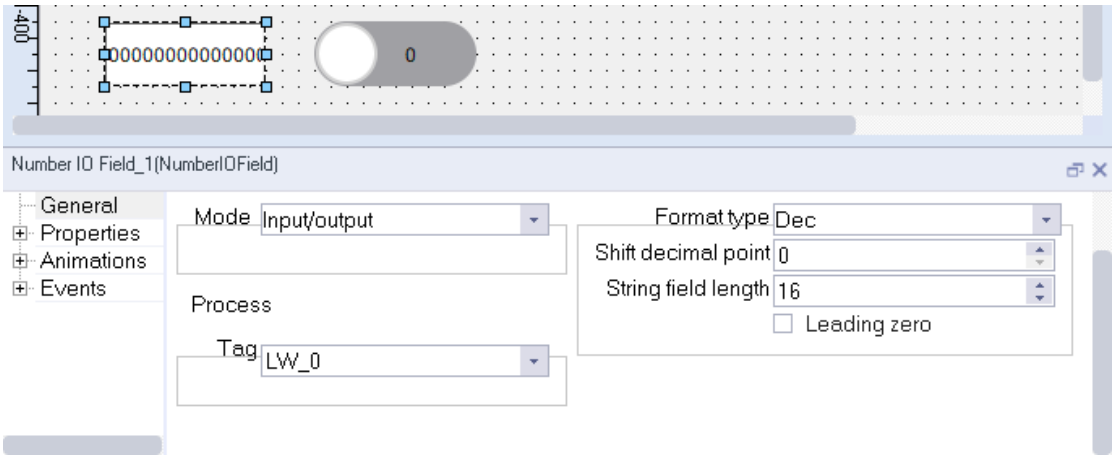
The properties of alarm view A are as follows: displays pending and unacknowledged error, warning, and system alarms.



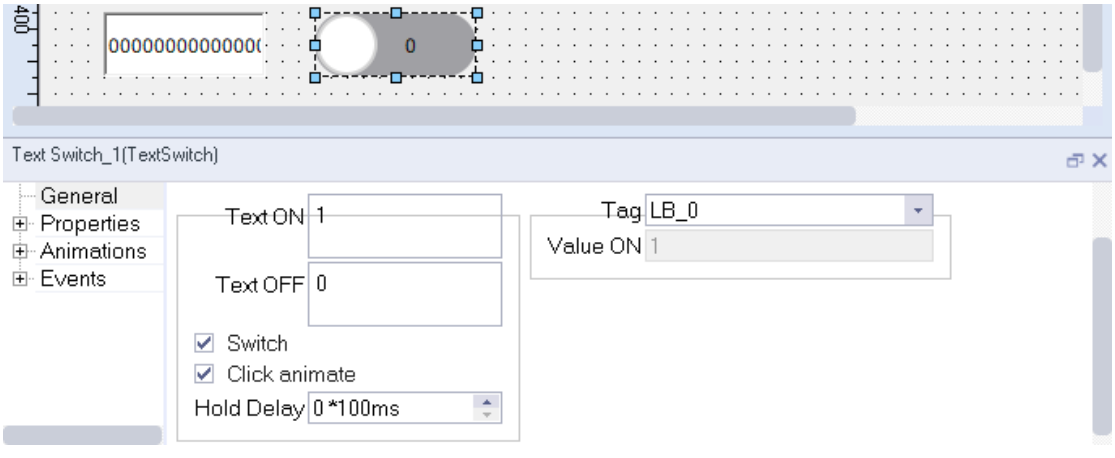
The properties of alarm view B are as follows: displays error, warning, and system alarms in Alarm log_1.



The number IO field process tag in the figure is LW 0:



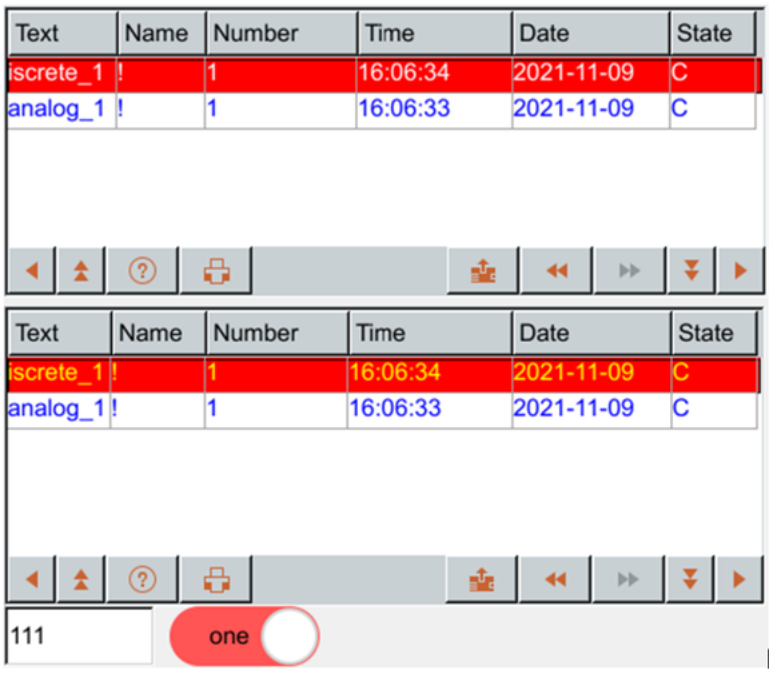
The text switch process tag is LB 0:



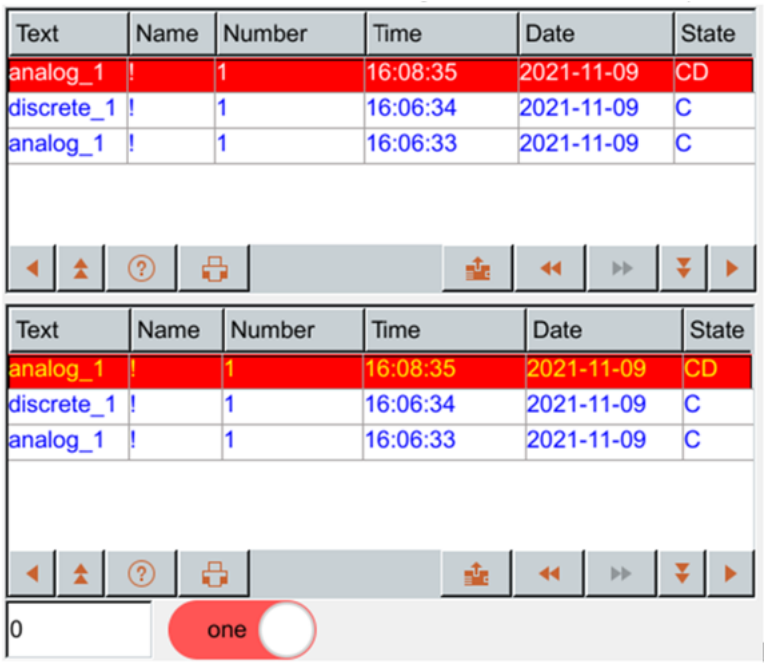
- Download and run

  After the configuration is completed, you can download and run the project.

  1. Input a value so that LW 0 is greater than 100 to trigger the Analog_1 alarm.
  2. Set LB 0 to trigger discrete_1 alarm. The following alarm message shows:



3. Enter the value so that LW 0 is less than 100. As shown in the following figure, the alarm status of Analog_1 in alarm view A changes to CD, while the new alarm information status in alarm view B (alarm log) is also CD. The alarm log records all the alarm information (the occurrence, acknowledgment and resolution of the alarm will produce a record).



10 Recipe

10.1 Overview

10.1.1 Description

A recipe is a set of data of the same type, and has a fixed data structure. A recipe can contain several data records, which have the same structure but are different in data. A recipe can be stored in a HMI or an external storage medium.

A project can configure up 100 recipes and a recipe can configure up to 32767 components. A recipe data record can configure up to 1000 records.

10.1.2 Display of Recipe

The recipe can be displayed through the recipe view control in the screen. The recipe view control is as shown in the following figure.

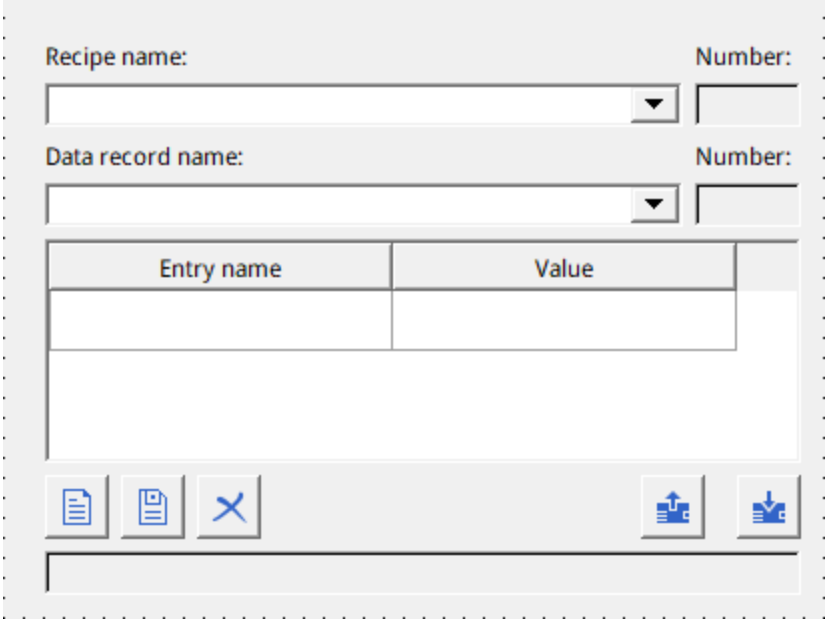


Figure 10-1 Recipe view control

10.1.3 System Functions for Editing Recipes

See *"12.3.12 Recipe" on page 260*.

10.2 Properties and Basic Settings

10.2.1 Recipe Editor

You can create, configure and edit recipes in the recipe table editor. You can also set element in the recipe data log.

Opening the recipe editor

Expand Recipes and double-click Recipe _ 1 in the project view as shown in the following figure to open the recipe editor.

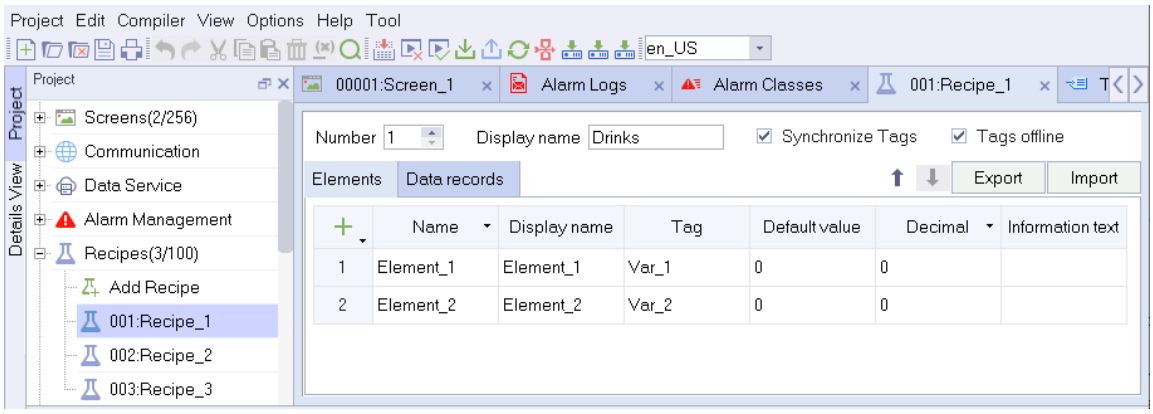


Figure 10-2 Recipe editor

Editor area

- Creating an object: Click + in the upper left corner of the table. A recipe can have up to 32767 elements and 1000 data logs.
- Batch creating: Click the triangle under the Add button and select an operation in the drop-down menu. Then click the Add button;

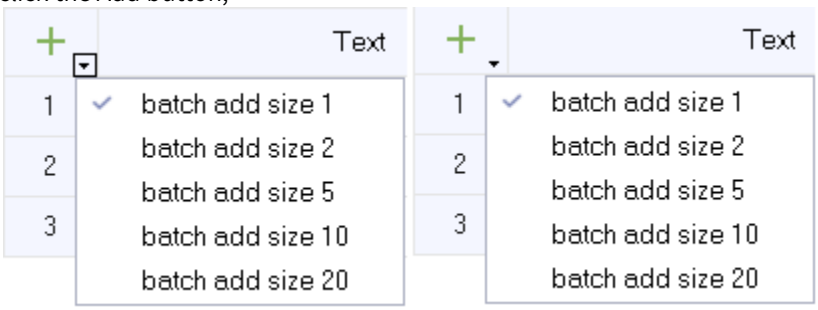


Figure 10-3 Batch creating options

- Deleting an object: Click the row number of the alarm row to be deleted, and the Add button becomes the Delete button −. Click the button to delete the row;
- Hiding and displaying column fields: Right-click the column header, and select the fields to be hidden or displayed 2/3/2010 in the drop-down menu.

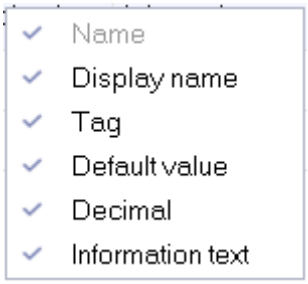


Figure 10-4 Column attributes

10.2.2 Recipe Property

- Number
  Unique identification of a recipe in HMI.
- Display Name
  In the runtime, the name of the recipe that is displayed when the recipe is selected.
- Synchronize Tags
  Synchronize the value in the recipe log with the value of the recipe tag. To use the Synchronize button  in the recipe view, you must check Synchronize Tags.
- Tags Offline
  When this is checked, the tags on the HMI will be disconnected from the corresponding PLC addresses. This option can be enabled when Synchronize Tags is checked.

Caution

For the use of synchronize tag and offline tag, see *"10.4.2 Applications of the Recipe" on page 241*.

- Export
  You can export the elements in the table to the specified. csv file, as well as data logs.
- Import
  Select the specified recipe file to import it to the current recipe table, as well as data logs.
- Name
  Unique identifier inside the recipe.
- Display Name
  The name of the entry that appears in recipe view when the HMI is running.
- Tag
  The HMI tag corresponding to the recipe element. Array tags can also be used here when element tags are contiguous addresses.
- Default Value
  Default value for recipe data.
- Decimal point
  The number of decimal places displayed by recipe data at run time.

- Info text
  A description of a recipe item that can be viewed when the HMI device is running by selecting a recipe item and clicking  on the recipe view.

10.2.3 Recipe Data Record

The value in the recipe data record, which is the preset element value. During HMI operation, you can modify the HMI tag value of an element by switching recipe data records.

Recipe data record editor

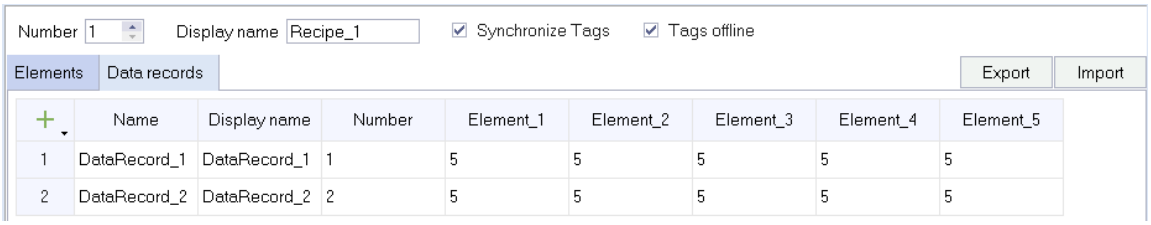


Figure 10-5 Recipe data record editor

Properties

- Name
  The recipe data record name uniquely identifies the recipe data record in the recipe.
- Display Name
  The name of the recipe data record that displays in the recipe view. You can configure the display name in multiple languages. You can also specify a descriptive name or a mark directly related to the product, such as a product number. The display name can be duplicate.
- Number
  The number uniquely identifies the recipe data record in the recipe.
- Element_1 (n)
  It is a preset setting of the tag value in Element_1. "n" is the sub-index of the element array tag. The default value in the element is used as the preset value if it is not provided.

10.3 Viewing and Editing Recipe at Run Time

10.3.1 Recipe view

You can view and edit recipes on the recipe view.

10.3.2 View and Edit Recipes in Recipe View

1. Recipe configuration and recipe data record are as follows:

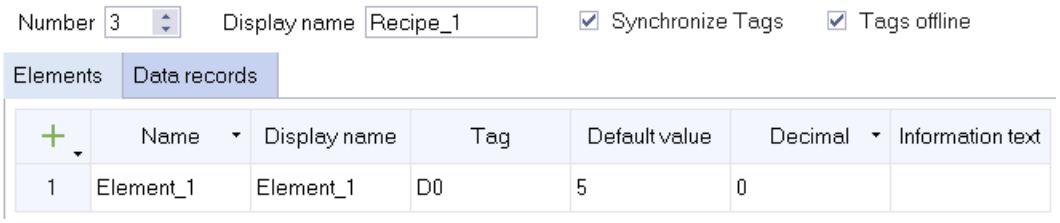


Figure 10-6 Recipe ingredient

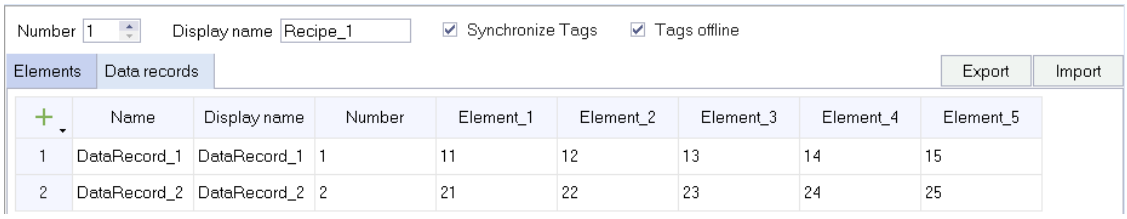


Figure 10-7 Recipe Data Record

2. In the runtime, a recipe is displayed in recipe view as shown in the following figure:
   The ingredient values in the figure are all 5 which is the default value.

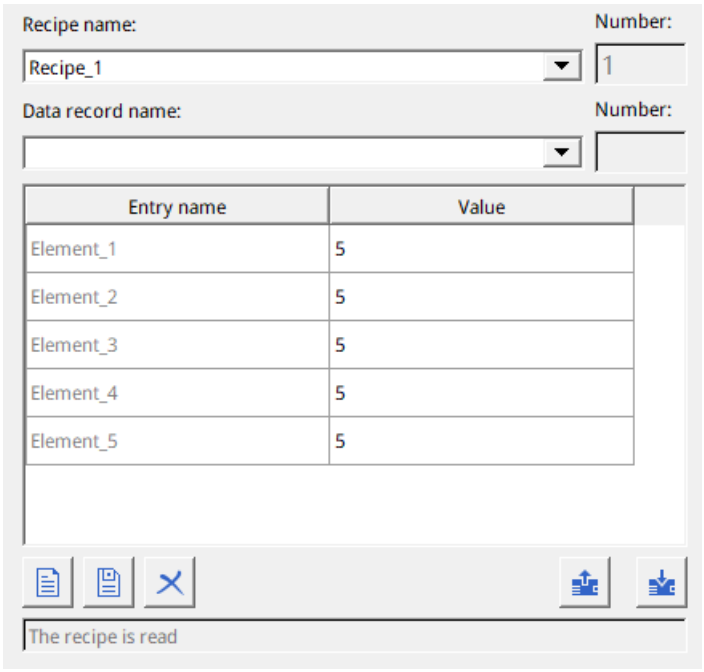


Figure 10-8 Recipe view control

3. Viewing a recipe
   Click on the recipe view shown in the following figure to select the recipe you want to view. you can view any configured recipe in the recipe view.

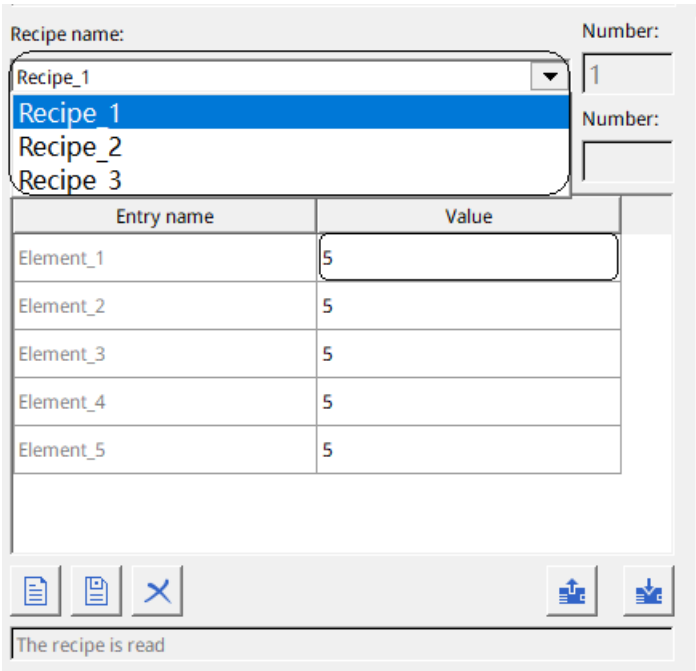


Figure 10-9 Recipe view

4. Editing a recipe at run time

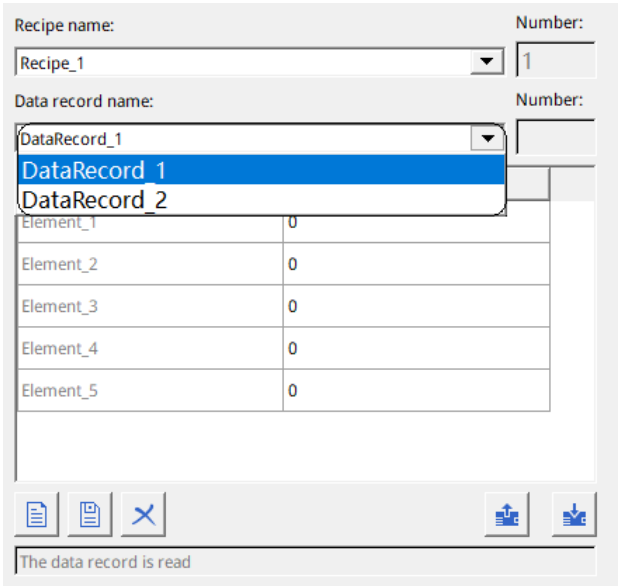
As shown in *"Figure 10–9 Recipe view" on page 239*, you can double-click Value to edit the recipe ingredient value, but since no data record is selected at this time, the edited value cannot be saved.

10.3.3 Viewing and Editing Recipe Data Records at Run Time

For recipe configuration, see *"10.3.2 View and Edit Recipes in Recipe View" on page 237*.

Viewing recipe data record

After selecting a recipe, you can select a recipe data record that belong to it in the recipe view control. As shown in the following figure.



Here, we select Data record_1, as shown in the following figure. The element values in the figure are modified to the default values preset in Data record_1. The message box also shows "The data record is read".

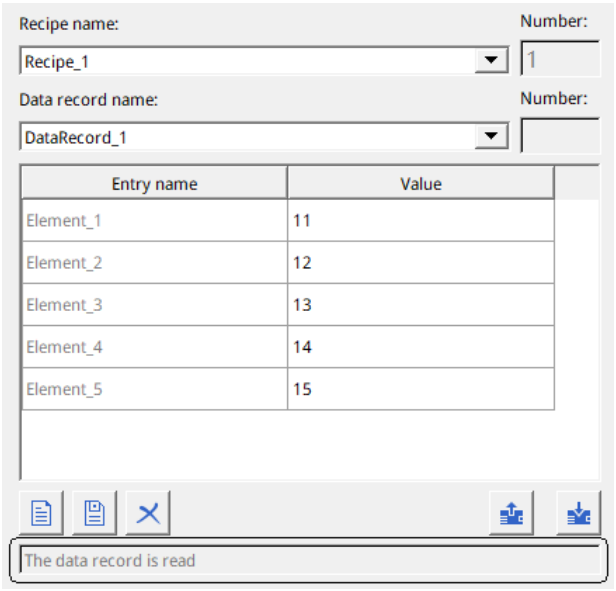


Figure 10-10 Recipe view

Editing recipe data record

- Editing data record value
  After selecting the recipe data record, you can double-click the value in the table to edit the element value. Different from recipe editing, after editing the value, you need to click the Save button on the recipe view control to write the modified value to the corresponding recipe data record.
- Creating a data record
  Click the New button in the recipe view to pop up the New data record dialog box as shown in Figure 10-3-8. Enter the data record name to create the data record. After creating the data record, you can select it to edit it.

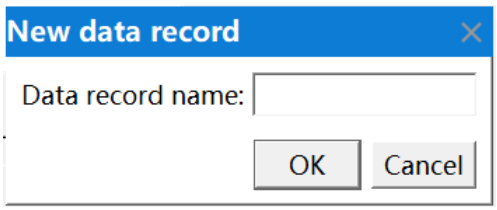


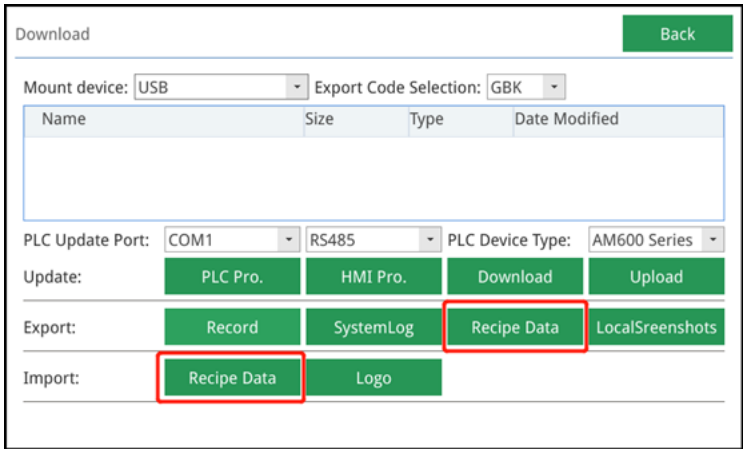
Figure 10-11 Creating a data record

- Deleting a data record
  After selecting the specified data record, click the Delete button in the recipe view to delete it.

10.4 Using Recipes

10.4.1 Importing and Exporting Recipe Data Records

In the recipe screen, configuration *"12.3.12 Recipe" on page 260* can be used to control the recipe. The system functions for loading, saving, and transferring recipe data records and recipes are located in the "recipe" of the event.
You can import and export the recipe data from/to a USB drive or SD card in the Download page of the control panel.



During export, all the recipe data records in the project will be exported to the specified path and saved to a *. CSV file (each recipe is saved as a file separately), and the saved contents are the name of the recipe data record and the element values.

When you import a recipe data record, the recipe will be updated (only one recipe can be updated at a time).

- In the exported file, only the component data can be modified. The identification text in the first line and data record name cannot be changed, otherwise the modified file will fail to be imported.
- Any data record will be discarded when it is imported if any of its component values or data types are incorrect.

10.4.2 Applications of the Recipe

In recipe applications, Synchronize Tags and Tags Offline can be used as follows:

Application	Synchronize Tags	Tags Offline
1	×	×
	✓	✓
2	✓	×
✓-Checked; ×-Unchecked		

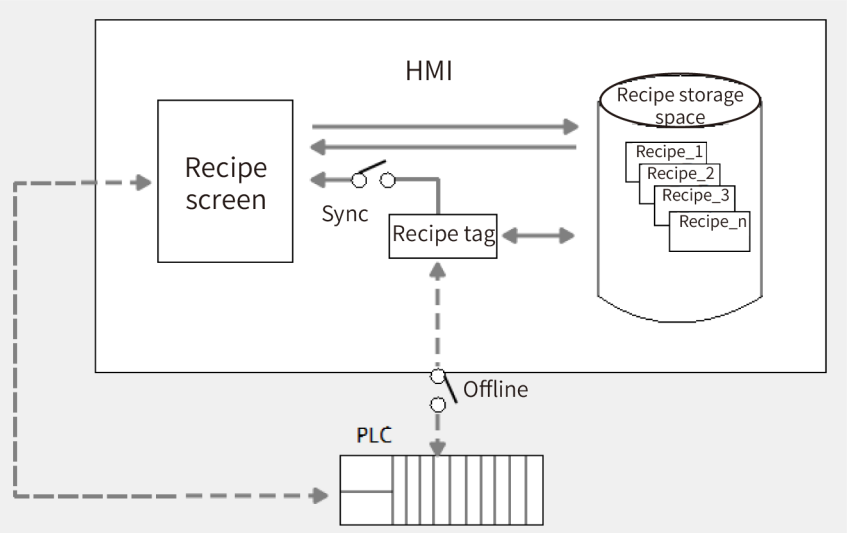
Application 1: Entering recipe data at run time

Objective: To input production data on an HMI device without interfering with the ongoing process.

Requirements for recipe configuration:

1. Synchronize Tags is not checked, both Synchronize Tags and Tags Offline are checked

2. The screen contains a recipe view
3. There is an element storing data records



When Synchronize Tags is not checked:

1. Enter product data into the table in the recipe view.
2. In the recipe view, save the modified recipe data record to the recipe storage space (you can save this data record directly or save it with another name)
3. In the recipe view, transmit the recipe data to the PLC through the To PLC button.

When both are checked:

You can use the same steps mentioned above. Or:

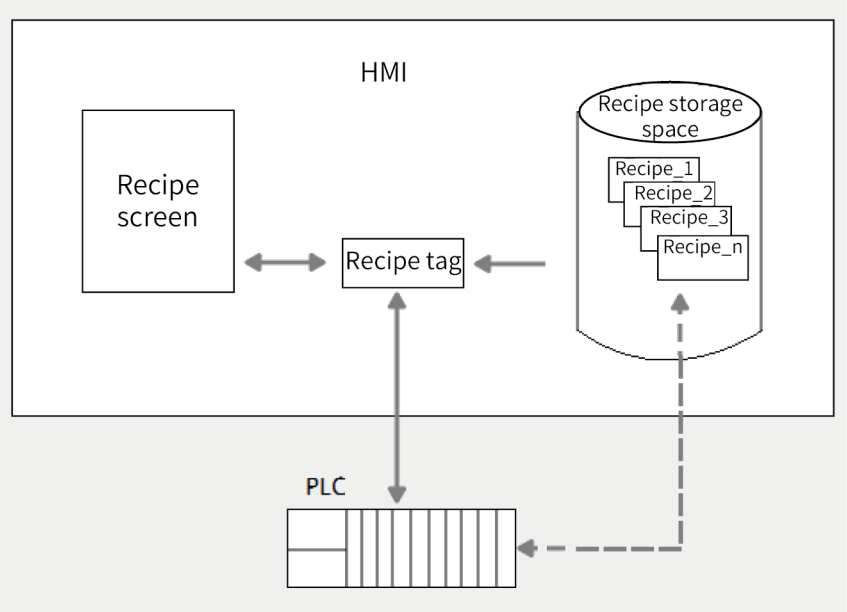
1. Modify the value of the numeric IO field where the tag is.
2. In the recipe view, synchronize the value of the IO field to the recipe table with the Synchronize Tag button, and then save the modified value to the recipe storage space with the Save button on the recipe view. The two operations are equivalent to the system function SaveDataRecord().
3. In the recipe view, transmit the recipe data to the PLC through the To PLC button. You can also transmit the recipe data from the recipe storage space to the PLC through the system function SetDataRecordToPLC().

Application 2: Manual production sequence

Objective: To display recipe data in real time on the recipe screen, and correct the transferred production data online.

Requirements for recipe configuration:

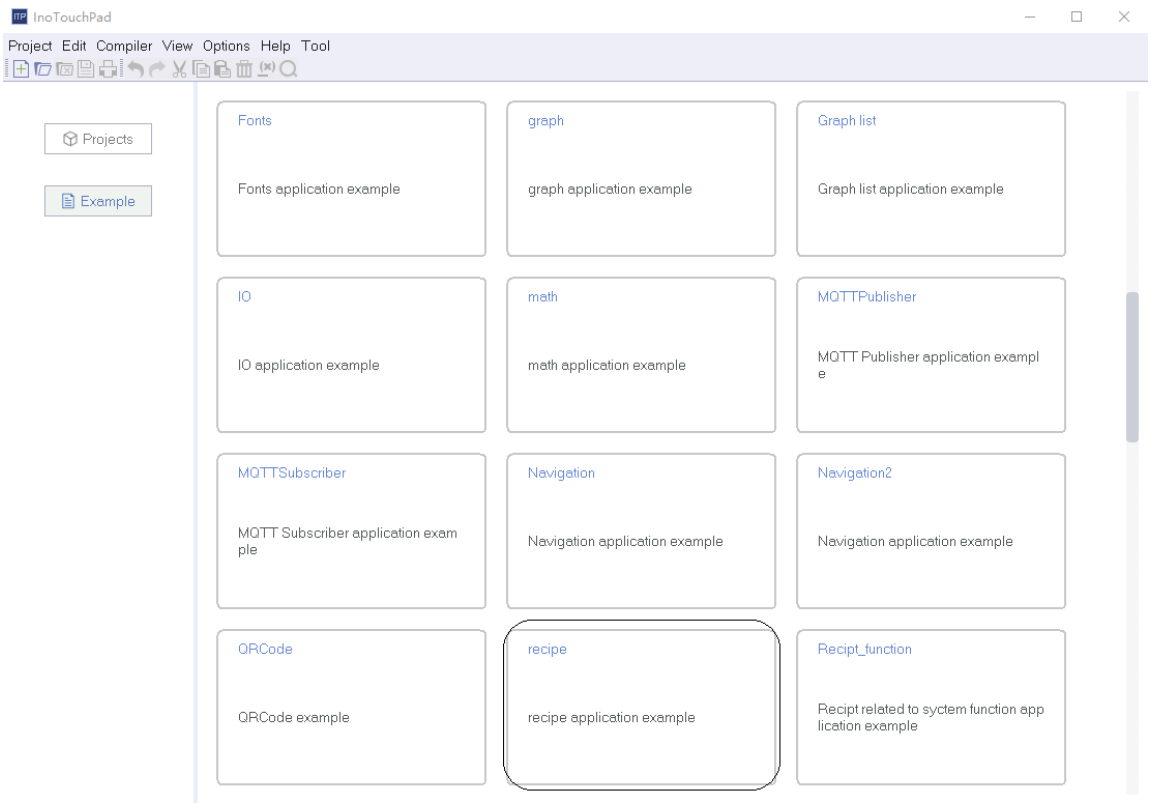
1. Synchronize Tag is checked and Tags Offline is not checked
2. The screen contains a recipe view
3. There is an element storing data records



Steps:

1. View and modify the element values in the recipe view. Transmit the recipe data in the PLC directly to the recipe view. Transmit the modified recipe data to the PLC.
2. Use the system function SetDataRecordToPLC() to transfer the data from the recipe storage space to the PLC, use the system function GetDataRecordFromPLC() to transfer the data from the PLC to the recipe storage space, and use the system function LoadDataRecord() to transfer the data from the recipe storage space to the recipe tag.

For details, see the example project "recipe" in InoTouchPad:



11 Historical Data

11.1 Overview

The record keeps tag values and alarm values generated previously. The history data function block includes data recording and alarm recording functions, which are managed by different editors and attribute views respectively.

11.2 Data log

11.2.1 Data Log Editor

You can create new data logs and specify their properties in the data log editor.

Opening the data log editor

Expand Historical Data, and double-click Alarm Logs.

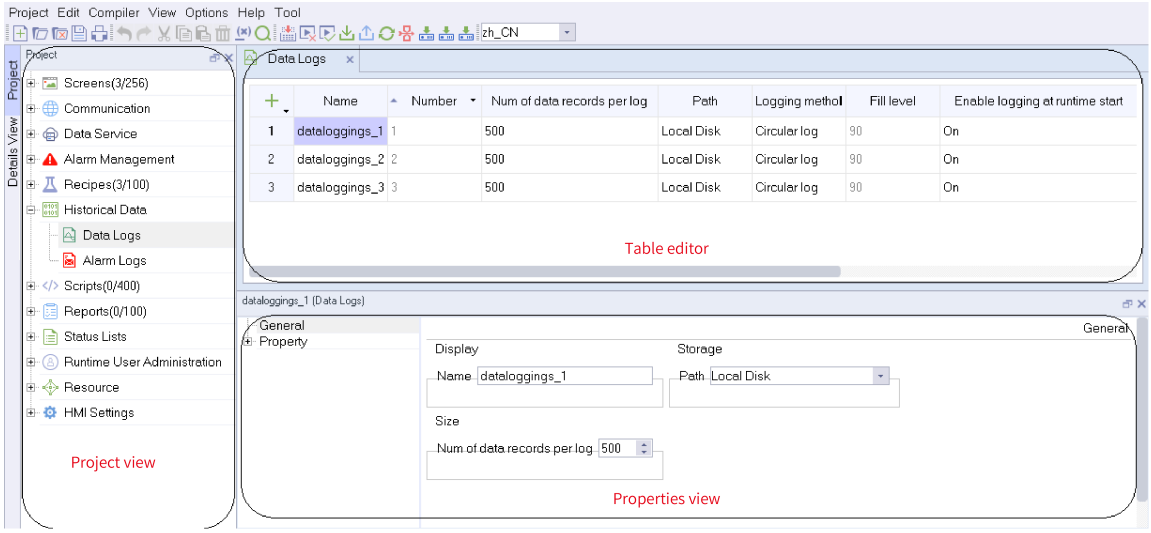


Figure 11-1 Data log editor

Editor area

- Creating an object: Click + in the upper left corner of the table;
- **Batch creating:** Click the triangle under the Add button and select an operation in the drop-down menu. Then click the Add button;

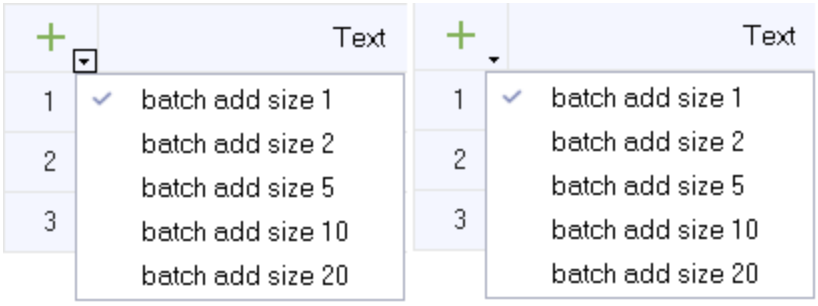


Figure 11-2 Batch creating options

- **Deleting an object:** Click the row number of the data log row to be deleted, and the Add button becomes the Delete button . Click the button to delete the row;
- **Hiding and displaying column fields:** Right-click the column header, and select the fields to be hidden or displayed in the drop-down menu.

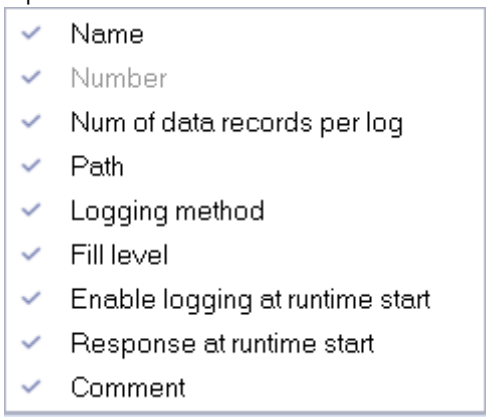


Figure 11-3 Column attributes

- **Column sort: Click the column header**

Properties view

The properties view provides the same property setting function as the editor area. In addition, It also provides event editing.

Description of event

- Overflow event: The script function that triggers configuration when the data logs overflow.
- Event items can only be used when triggering events in Logging Method configuration.

11.2.2 Basic Setting

Property setting

- Name
  Name of the data log
- Num of data records per log
  The maximum number records allowed by an alarm log

- Path
  The storage location of the data log, which can be the HMI or a USB drive or SD card
- Logging method
  - Circular log: When the table is full, the earliest log will be overwritten.
  - Raise event: When the table is full, the script function configured in the overflow event is triggered.
  - Display system alarm: When the number of records reaches the specified maximum number, the system alarm is displayed.
- Fill level
  The maximum number of logs in percentage to display the system alarm.
- Enable logging at runtime start
  Choose whether the data logging function is enabled when HMI is started.
- Response at runtime start
  - Append data to existing log: Keep the previous data logs when HMI starts.
  - Reset log: Clear the previous data logs when HMI starts.
- Comment
  Description of the log

11.2.3 Logging Tag Values

Logging configuration data

Open the Data Log Editor and add a data log.

Logging configuration tag

As shown in the tags editor in the following figure.

	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycle Id	Acquisition mode	Data log Id	Logging cycle Id	Logging acquisition mode
1	LW0	1	<Internal tag>	Int16	2	1	LW 0	100ms	Cyclic on use	DataLog_1	1s	Cyclic continuous

Figure 11-4 Tags editor

- **Data log ID:** the data log which stores the tag value.
- **Logging cycle ID:** The logging interval. The logging cycle is related to the acquisition mode. Only when the acquisition mode is "cyclic continuous", the cycle is effective.
- **Acquisition mode:**
  - When changing: The current value is recorded when the tag value changes.
  - On demand: The tag value is recorded by executing the "LogTag" system function.
  - Cyclic continuous: Acquisition is done based on the logging cycle.

11.2.4 Outputing Logged Data

Viewing the tag values in the specified data log on the data view

As shown in the following figure, configure a data view control in the screen, and specify the data log. Then you can see all the recorded tag values in the specified time period on the HMI.

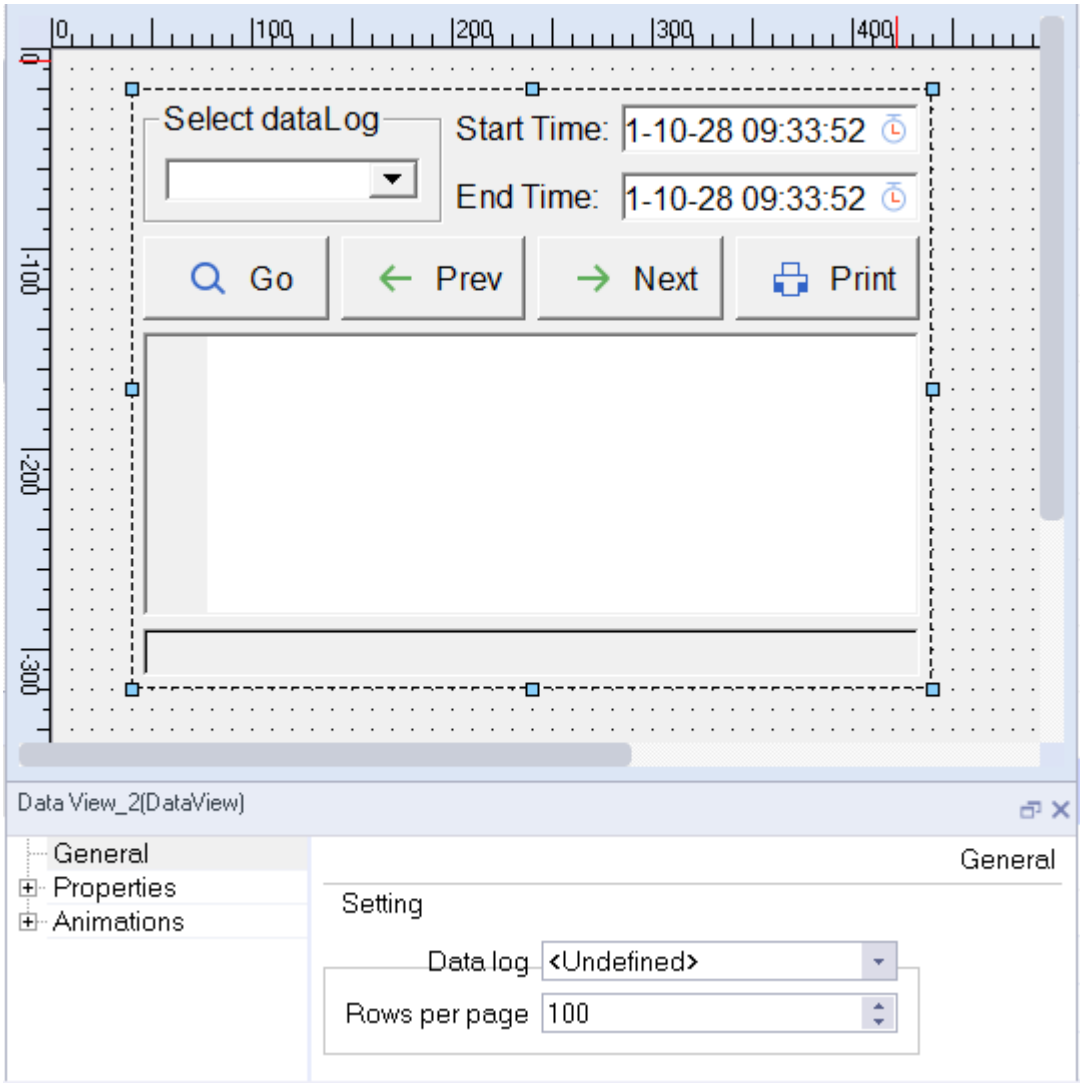


Figure 11-5 Configuring a data view

Viewing the tag values in the specified data log on the trend view

As shown in the following figure, configure a trend view control in the screen. In Trend, set the Trend Type as Log, configure the related tag, and then the historical graph of the specified tag can be displayed in the trend view.

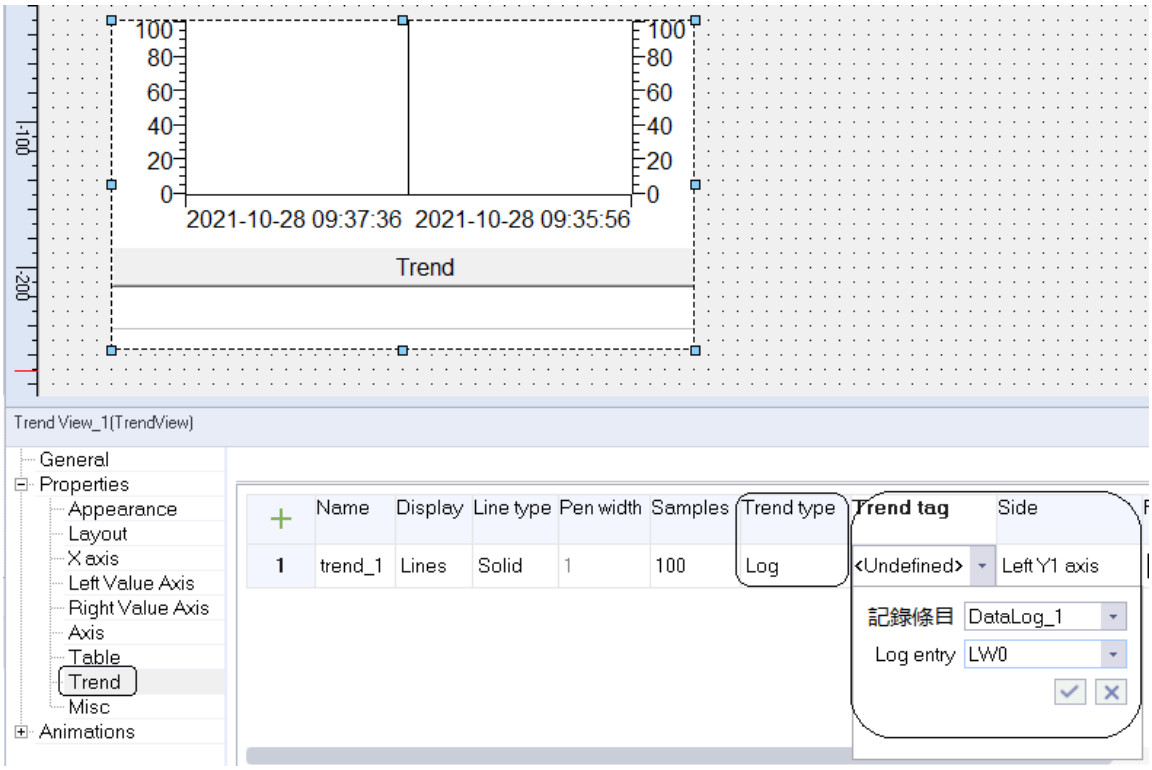


Figure 11-6 Displaying data log in a trend chart

Exporting data log from the HMI

Insert an SD card on the HMI, and then, as shown in the following figure, go to Control Panel->Download->Export->Logs->Data Log. A *. csv file is exported.

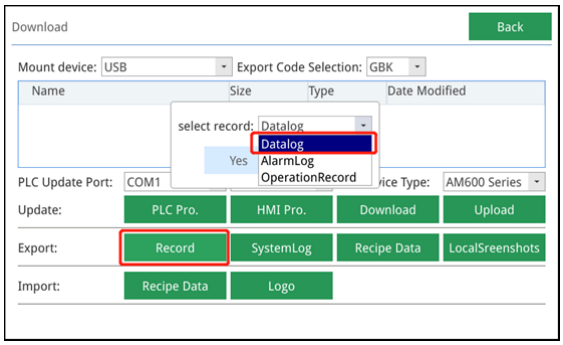


Figure 11-7 Control panel view

Caution

If you cannot enter the download page, check whether the USB drive or SD card is properly connected.

- **Access to the control panel:**
  - When HMI is running, if you insert a USB drive or SD card, the control panel screen will pop up automatically.
  - Configuring the system function OpenControlPanel in the screen.

11.3 Alarm Record

See *"9.3.1 Description" on page 226*.

12 System Functions and Scripts

12.1 Overview

InoTouchPad provides predefined system functions for general configuration tasks. You can use them to accomplish many tasks in the running system without programming skills.

You can also solve complex problems by scripts. The script has a programming interface, which can access some project data in the running system. Scripting is for project designers with knowledge of JavaScript.

12.2 Use of System Functions

You can configure system functions on all objects that can respond to events. System functions are mainly used in function lists and scripts to control processes.

- Function list
  The system functions in the function list are processed in order.
- Script
  In a script, you can use system functions related to commands and requirements in your code. In this way, scripts can be executed according to the specified system state.

12.3 Function Types

12.3.1 Introduction

System functions are predefined functions that can be used to perform many tasks at run time without even requiring any programming knowledge, such as:

- Calculation. for example, they can be used to increase the value of a variable by a specific value or a variable value.
- Recording. They can be used to record the starting process value.
- Setting. They can be used to replace PLC or set bits in PLC.
- Alarm. For example, they can be used to issue an alarm when another user logs in.

12.3.2 Calculation

No.	Function		Description	Script Calling
1	DecreaseValue (Tag, Value, Reset)		Tag = Tag – Value	Supported, for example, DecreaseValue (SmartTags ('tag_1'), 1, 0);
	Parameter	Tag(InOut)	A tag defaulted to null	
		Value(In)	A constant/tag defaulted to 1	
		Reset	Reset. Options include Yes and No (default). If Yes is selected, the tag will be reset to the upper limit after being reduced to the lower limit.	
2	IncreaseValue (Tag, Value, Reset)		Tag = Tag + Value	Supported, for example, IncreaseValue(SmartTags('tag_1'),1,1);
	Parameter	Tag(InOut)	A tag defaulted to null	
		Value(In)	A constant/tag defaulted to 1	
		Reset	Reset. Options include Yes and No (default). If Yes is selected, the tag will be reset to lower limit after being increased to upper limit.	
3	InverseLinearScaling (X, Y, b, a)		X = (Y - b)/a, X and Y cannot be the same	Supported, for example, InverseLinearScaling(SmartTags('tag_1'),SmartTags('tag_2'),2,2);
	Parameter	X(Out)	A tag defaulted to null	
		Y(In)	A tag defaulted to null	
		b(In)	A constant/tag defaulted to 0	
		a(In)	A constant/tag defaulted to 1	
4	LinearScaling(Y, X, a, b)		Y= (a *X) + b, X and Y cannot be the same	Supported, for example, LinearScaling(SmartTags('tag_1'),3,SmartTags('tag_2'),3);
	Parameter	Y(Out)	A tag defaulted to null	
		a(In)	A constant/tag defaulted to 1	
		X(In)	A tag defaulted to null	
		b(In)	A constant/tag defaulted to 0	
5	SetValue (Tag, Value)		Tag = Value	Supported, for example, SetValue(SmartTags('tag_1'),22);
	Parameter	Tag(InOut)	A tag defaulted to null	
		Value(In)	A constant/tag defaulted to 0	
6	Random (Tag)		Take a random value in the upper and lower limits of a tag	Supported, for example: Random (SmartTags('tag_1'));
	Parameter	Tag(Out)	A tag defaulted to null	

12.3.3 Bit Operations

No.	Function		Description	Script Calling
1	InvertBit (Tag)		Inverts a Bool tag	Supported, for example, InvertBit(SmartTags('tag_1'));
	Parame-ter	Tag(InOut)	A tag defaulted to null	
2	InvertBitInTag (Tag, Bit)		Inverts the bit of a tag	Supported, for example, InvertBitInTag(SmartTags('tag_1'),1,0);
	Parame-ter	Tag(InOut)	A tag defaulted to null	
		Bit(In)	Bit number, ranging from 0 to 31 and defaulted to 0	

No.	Function		Description	Script Calling
3	ResetBit (Tag)		Set the value of a Bool tag to 0	Supported, for example, ResetBit(SmartTags('tag_1'));
	Parame-ter	Tag(InOut)	A tag defaulted to null	
4	ResetBitInTag(Tag, Bit)		Sets a bit in a tag to 0	Supported, for example, ResetBitInTag(SmartTags('tag_2'),0);
	Parame-ter	Tag(InOut)	A tag defaulted to null	
		Bit(In)	Bit number, ranging from 0 to 31 and defaulted to 0	
5	SetBit(Tag)		Set the value of a Bool tag to 1	Supported, for example, SetBit(SmartTags('tag_1'));
	Parame-ter	Tag(InOut)	A tag defaulted to null	
6	SetBitInTag(Tag, Bit)		Sets a bit in a tag to 1	Supported, for example, SetBitInTag(SmartTags('tag_2'),0);
	Parame-ter	Tag(InOut)	A tag defaulted to null	
		Bit(In)	Bit number, ranging from 0 to 31 and defaulted to 0	

12.3.4 Screen

No.	Function		Description	Script Calling
1	ActivateScreen(ScreenName)		Switches to the specified screen by screen name	Supported, for example: ActivateScreen ('Screen_1');
	Parameter	ScreenName	Screen name	
2	ActivateScreenByNumber (ScreenId)		Switches to the specified screen by screen ID	Supported, for example: ActivateScreenByNumber (SmartTags('Tag_1'));
	Parameter	ScreenId	Screen ID	
3	ActivatePreviousScreen		Switches to the previous screen	Supported, for example: ActivatePreviousScreen();
4	ShowPopup		Displays a pop-up screen at the specified position	Supported, for example: ShowPopup ('Screen_2', 100, 200);
	Parameter	ScreenName	The name of the pop-up screen	
		posX	X coordinate of the pop-up screen, which is screen center by default. Constant/tag	
		posY	Y coordinate of the pop-up screen, which is screen center by default. Constant/tag	
5	HidePopup		Hides a pop-up screen	Supported, for example: HidePopup('Screen_2');
	Parameter	ScreenName	The name of the pop-up screen	
6	HideAllPopups		Hides all pop-up screens	Supported, for example: HideAllPopups();

12.3.5 User Management

No.	Function		Description	Script Calling
1	GetAuthorization		Writes the permissions of the group to which the currently logged-in user belongs to the specified tag	Not supported
	Parameter	Tag(Out)	writes a tag. Default is null. The tag must be of Int32 type.	
2	GetGroupNumber (Tag)		Writes the group number of the currently logged-in user to the specified tag	
	Parameter	Tag(Out)	Writes a tag. Default is null.	
3	GetPassword (Tag)		Writes the current user password to the specified tag	
	Parameter	Tag(Out)	Writes a tag. Default is null.	
4	GetUserName (Tag)		Writes the current user name to the specified tag	
	Parameter	Tag(Out)	Writes a tag. Default is null.	
5	Logoff		Logs off	
6	Logon (UserName, Password)		Logs on	
	Parameter	UserName(In)	User name. Default is null. It must be of string type.	
		Password(In)	Password. Default is null. It must be of string type.	
7	RemoveUser (UserName)		Administrator group users can use this to delete a user (except admin users) by user name	
	Parameter	UserName(In)	User name. Default is null.	
8	ShowLogonDialog		Pops up the logon dialog box	
9	Parameter	ChangePass-word (UserName, Password)	Modify the password of the specified user	
		UserName(In)	User name. Default is null. It must be of string type.	
		Password(In)	Password. Default is null. It must be of string type.	

12.3.6 Date and Time in HMI

No.	Function		Description	Script Calling
1	SetHMITime(Tag, Tag,Tag)		Sets the current HMI time: hours, minutes and seconds.	Not supported
	Parameter	Tag(In)	Reads the value of the bound tag and converts the value into a time in hours, minutes and seconds, which is then written into the HMI.	
2	SetHMIDate(Tag,Tag,Tag)		Sets the current HMI time: year, month and day.	
	Parameter	Tag(In)	Reads the value of the bound tag and converts the value into a time in year, month and day, which is then written into the HMI. (You need to check whether the time is valid. The system does not regard February 30th wrong.)	
3	SetHMIDateTime(Tag,Tag, Tag, Tag, Tag,Tag)		Sets the current HMI time: year, month, day, hours, minutes and seconds.	
	Parameter	Tag(In)	Reads the value of the bound tag and converts the value into a time in year, month, day, hours, minutes and seconds, which is then written into the HMI. (You need to check whether the time is valid. The system does not regard February 30th wrong.)	
4	UpdateTagDateTime (Tag)		Reads the current HMI time value and writes it to the specified tag	
	Parameter	Tag(Out)	Tag is of DateTime type.	

12.3.7 Settings

No.	Function		Description	Script Calling
1	SetConnectionMode (Connection, Mode)		Connect or disconnect a specified connection	Supported, for example, SetConnectionMode ("Connection_1", 0);
	Parameter	Connection (In)	Connection Name	
		Mode(In)	Value: online/offline	
2	ChangeConnection (Connection, Station address)		Change the slave device address of the specified connection	Supported, see SetConnectionMode
	Parameter	Connection (In)	Connection Name	
		Station address(In)	Slave device address	
3	GetConnectionModeState (Connection)		Gets the state of a specified connection. This function is used in scripts	
	Parameter	Connection (In)	Connection Name	

No.	Function		Description	Script Calling
4	ChangeConnectionIpAddress (Connection, IpAddress)		Change the IP address of the connected slave	Not supported
	Parameter	Connection (In)	Connection Name	
		IpAddress (In)	IP address, with string tag	
5	UpdateTag(Tag)		Get the latest value of a tag	
	Parameter	tag	Tag	
6	SetLanguage (Language)		Set the language of the HMI	
	Parameter	language(In)	Language abbreviation You can select Toggle or language abbreviation; Toggle: switch to the next language	
7	SetStyle (styleSheet)		Set the global style of the HMI	
	Parameter	styleSheet (In)	You can select Toggle or a style name; Toggle: switch to the next style	
8	SetBlacklightMode (BlacklightMode)		Set backlight mode	
	Parameter	Blacklight-Mode (In)	Off/ON, default is ON	

No.	Function		Description	Script Calling
9	SetBacklight（LightValue）		Set the screen brightness	Not supported
	Parameter	LightValue (In)	0 to 100. Default value: 90	
10	SetBacklightTime（TimeValue）		Sets backlight timeout, in minutes	
	Parameter	TimeValue (In)	0 to 255. Default value: 5	
11	SetScreenSaverTime（TimeValue）		Sets screensaver timeout, in minutes	
	Parameter	TimeValue (In)	0 to 255. Default value: 3	
12	SetSoftwareInputEnabled（Status）		Disabled/enabled soft keypad	
	Parameter	Status(In)	Value: enabled/disabled	
13	OpenControlPanel		Open the HMIAutorun setup interface	
14	CalibrateTouchScreen		Open the calibration interface	
15	RestoreFactorySettings		Restores factory settings	
16	Sync		Synchronize the data in memory to disk. Before exiting/restarting, you need to call this function to save all the data to the disk.	
17	Reboot		Restart the operating system that hosts HMIRuntime	
18	Shutdown		Shut down the HMI	
19	UmountDisk(TmpType, Tag)		Unmount a storage device	
	Parameter	TmpType(In)	Value: SD card/USB drive, default is SD card	
		Tag(Out)	Process status output value, which is empty by default	
20	TrigInstalMentEarly(Tag)		Trigger the next installment in advance	
	Parameter	Tag(In)	Installment trigger tag, Bool type	
21	UnlockCurrentInstalMent		Unlock the current installment	

12.3.8 Data Service

No.	Function		Description	Script Calling
1	PublisheTag(Publisher)		Send the specified publish variable information to the server	Not supported
	Parame-ter	Publisher(In)	Publisher list ID	
2	WriteTagValuetoRemoteDB(Group)		Send a command to the device to start sampling data to the database	
	Parame-ter	Group(In)	Data group No.	
3	ReadTagValuefromRemoteDB(Group)		Send a command to the device to sample data from the database	
	Parame-ter	Group(In)	Query group No.	
4	SendMail(MailId)		Send an e-mail message from the HMI to a designated recipient	
	Parame-ter	MailId(In)	E-mail No.	

12.3.9 Printing

No.	Function		Description	Script Calling
1	PrintScreen (path)		Prints the current screen, and saves the screenshot to a USB drive when the printer is not connected	Not supported
	Parame-ter	Path	Screenshot storage directory: Local HMI/Storage Card/USB drive	
2	PrintAlarm(Screen object)		Prints the contents of an alarm view	
	Parame-ter	Screen object(In)	Screen object. You can only select the alarm view control.	
3	PrintLog (Screen object)		Prints the contents of a data view	
	Parame-ter	Screen object (In)	Screen object. You can only select the data view control.	
4	PrintRecipe (Screen object)		Prints the contents of a recipe view	
	Parame-ter	Screen object (In)	Screen object. You can only select the recipe view control.	
5	PrintBitMap (…)		Prints images in BMP format	
	Parame-ter	BmpStartX(In)	X coordinate of the starting point of the screenshot	
		BmpStartY(In)	Y coordinate of the starting point of the screenshot	
		BmpWide(In)	Width of the screenshot	
		BmpHigh(In)	Height of the screenshot	
		LabelStartX(In)	X coordinate of the starting point of the image on the label paper	
		LabelStartY(In)	Y coordinate of the starting point of the image on the label paper	

12.3.10 Alarm

No.	Function		Description	Script Calling
1	ClearAlarmBuffer		Delete alarms from the HMI alarm buffer Unacknowledged alarms are also deleted	Not supported
2	ShowAlarmWindow (ObjectName, DisplayMode)		Hide/show the alarm window on HMI	
	Parameter	ObjectName	Alarm window object name	
		DisplayMode	No: Hide the alarm screen Yes: Display the alarm screen Toggle: Switch between two modes	
3	ClearActionedAlarmBuffer		Delete alarms that have been dealt with from HMI alarm buffer	
4	Beep		Play a buzzer	

12.3.11 Log

No.	Function		Description	Script Calling
1	ClearLog (LogType, Log)		Deletes all records in a log	Supported, for example, ClearLog('data log_1');
	Parameter	LogType(In)	Data log/alarm log. Default: data log	
		Log(In)	Log name. Default is null.	
2	CloseAllLogs		Disconnects the runtime from all logs You must first stop the logging function before closing the log. Use the system function StopLogging.	Not supported
3	OpenAllLogs		Establishes a connection between the runtime and logs. You can proceed with logging. Use StartLogging to restart logging.	Not supported
4	LogTag (Tag)		Saves the value of a tag in a data log; Used for tag logging mode "on demand"	Supported, for example, LogTag(SmartTags('tag_1'));
	Parameter	Tag(In)	The tag whose value to be logged. Default: null	
5	StartLogging (LogType, Log)		Start logging	Not supported
	Parameter	LogType(In)	Data log/alarm log. Default: data log	
		Log(In)	Log name. Default is null.	
6	StopLogging (LogType, Log)		Stop logging	Not supported
	Parameter	LogType(In)	Data log/alarm log. Default: data log	
		Log(In)	Log name. Default is null.	

No.	Function		Description	Script Calling
7	LogTagGroup (Tag)		Saves the tag values in a tag group in a data log. Used by tag group logging mode "on demand"	Supported, for example, LogTagGroup ('tag group_2');
	Parameter	Group(In)	The tag group whose values to be logged. Default: null	
8	ClearOperationRecord		Clears operation records	Not supported
9	ExportDataLogs (PathType, EncodeType, Tag)		Exports all data logs	
	Parameter	PathType(In)	Export location. Default: local drive	
		EncodeType(In)	Encoding of the exported text. Default: GBK	
		Tag(out)	Operation status output. Default: null	
10	ExportAlarmLogs (PathType, EncodeType, Tag)		Exports all alarm logs	
	Parameter	PathType(In)	Export location. Default: local drive	
		EncodeType(In)	Encoding of the exported text. Default: GBK	
		Tag(out)	Operation status output. Default: null	
11	ExportOperationRecord(PathType, EncodeType, Tag)		Exports an operation record	
	Parameter	PathType(In)	Export location. Default: local drive	
		EncodeType(In)	Encoding of the exported text. Default: GBK	
		Tag(out)	Operation status output. Default: null	
12	ExportDataLogsByTime(PathType, EncodeType, Tag, StartTime, EndTime)		Exports data logs by time	
	Parameter	PathType(In)	Export location. Default: local drive	
		EncodeType(In)	Encoding of the exported text. Default: GBK	
		Tag(out)	Operation status output. Default: null	
		StarTime(In)	Start time, of DateTime type	
		EndTime(In)	End time, of DateTime type	
13	ExportDataLogsSplitDate(PathType, DirName， EncodeType, Tag)		Splits exported data logs by date	
	Parameter	PathType(In)	Export location. Default: local drive	
		DirName(In)	Directory name, can be a constant or a string tag	
		EncodeType(In)	Encoding of the exported text. Default: GBK	
		Tag(out)	Operation status output. Default: null	

12.3.12 Recipe

No.	Function		Description	Script Calling
1	DeleteDataRecord (…)		Deletes a recipe data record	Not supported
	Parameter	Recipe number/name(In)	Recipe number/name	
		Data record number/name (In)	Data record number	
		Confirmation(In)	Confirm, values: Yes and No	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
2	GetDataRecordFromPLC (…)		Transfers a given recipe data record from the PLC to a storage medium of the HMI	
	Parameter	Recipe number/name(In)	Recipe number/name	
		Data record number/name (In)	Data record number	
		Overwrite(In)	Overwrite, values: Yes, No and with confirmation	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
3	GetDataRecordName (…)		Writes the name of the given recipe and recipe data record in the a tag	
	Parameter	Recipe number(In)	Recipe number/name	
		Data record number(In)	Data record number	
		Recipe name(Out)	Recipe name	
		Data record name(Out)	Data record name	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
4	GetDataRecordTagsFromPLC (…)		Transfers the value of the recipe data record loaded into the PLC to a recipe tag	
	Parameter	Recipe number/name(In)	Recipe number/name	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
5	LoadDataRecord (…)		Transfers a given recipe data record from the storage medium of the HMI to a recipe tag	
	Parameter	Recipe number/name(In)	Recipe number/name	
		Data record number/name (In)	Data record number	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
6	SaveDataRecord (…)		Stores the current value of a recipe tag to the storage medium of the HMI as a data record	
	Parameter	Recipe number/name(In)	Recipe number/name	
		Data record number/name (In)	Data record number	
		Overwrite(In)	Overwrite, values: Yes, No and with confirmation	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
7	SetDataRecordTagsToPLC (…)		Sends the value of a recipe tag to the PLC. The recipe tag contains the value of the data record displayed on the HMI.	
	Parameter	Recipe number/name(In)	Recipe number/name	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	

No.	Function		Description	Script Calling
9	SetRecipeTags (…)		Sets the online/offline status of a recipe tag	Not supported
	Parameter	Recipe number/name(In)	Recipe number/name	
		Status (In)	Status, values: online and offline	
		Output status message(In)	Output status message, values: yes and no	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
10	ExportDataRecords		Exports data records	
	Parameter	Recipe number/name(In)	Recipe number/name	
		Path	Path	
		Code	Encoding, GBK	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
11	ImportDataRecords		Imports data records	
	Parameter	Path	Path	
		File name	File name (file: aaa, note that the file name here is different from the export function. For example, when the path is a local drive, the file name is RecipeDataRecords/aaa.csv)	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
12	ExportSingleDataRecords		Exports a single data record	
	Parameter	Recipe number/name(In)	Recipe number/name	
		Data number	Data record number	
		Path	Path: local disk, SD card, USB drive	
		Code	Encoding, GBK and UTF-8	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	
		ExportFileName	Name of the exported file	
13	ImportSingleDataRecords		Imports a single data record	
	Parameter	Recipe number/name(In)	Recipe number/name	
		Path	Path: local disk, SD card, USB drive	
		ImportFileName	Name of the Imported file	
		Processing status(Out optional)	Processing status, 2: Ready 4: Successful 12: Failed	

12.3.13 Operations on Screen Objects

No.	Function		Description	Script Calling
1	AlarmViewAcknowledgeAlarm (Screen object)		Acknowledges the alarm in the specified alarm view	Not supported
	Parameter	Screen object (In)	Screen object. You can only select the alarm view control.	
2	AlarmViewShowOperatorNotes (Screen object)		Displays the operator instructions in the specified alarm view. These instructions are the information text configured in analog/discrete alarms	
	Parameter	Screen object(In)	Screen object. You can only select the alarm view control.	
3	RecipeViewDeleteDataRecord (Screen object)		Deletes the the data record displayed in a recipe view	
	Parameter	Screen object (In)	Screen object. You can only select the recipe view control.	
4	RecipeViewGetDataRecordFromPLC (Screen object)		Sends the data record loaded in the PLC to the HMI and displays it in the recipe view.	
	Parameter	Screen object (In)	Screen object. You can only select the recipe view control.	
5	RecipeViewNewDataRecord (Screen object)		Creates a new data record in the given recipe view.	
	Parameter	Screen object (In)	Screen object. You can only select the recipe view control.	
6	RecipeViewSaveAsDataRecord (Screen object)		Saves the data record currently displayed in recipe view with a new name.	
	Parameter	Screen object (In)	Screen object. You can only select the recipe view control.	
7	RecipeViewSaveDataRecord (Screen object)		Saves the the recipe data record displayed in the recipe view.	
	Parameter	Screen object (In)	Screen object. You can only select the recipe view control.	
8	RecipeViewSetDataRecordToPLC (Screen object)		Sends the the recipe data record displayed in the recipe view to the PLC.	
	Parameter	Screen object (In)	Screen object. You can only select the recipe view control.	
9	RecipeViewShowOperatorNotes (Screen object)		Displays configured information text of a specified recipe view.	
	Parameter	Screen object (In)	Screen object. You can only select the recipe view control.	
10	RecipeViewSynchronizeDataRecordWith-Tags (Screen object)		Synchronizes the element tag value to the data record currently displayed in the recipe view	
	Parameter	Screen object (In)	Screen object. You can only select the recipe view control.	
11	TrendViewBackToBeginning (Screen object)		Goes back to the beginning of the trend record in the trend view, where the starting value of the trend record will be displayed.	
	Parameter	Screen object (In)	Screen object. You can only select the trend view control	
12	TrendViewRulerBackward (Screen object) -262-		Move the ruler backward in the trend view	
	Parameter	Screen object (In)	Screen object. You can only select the trend view control	
13	TrendViewRulerForward (Screen object)		Move the ruler forward in the trend view	

12.3.14 Free Protocol

No.	Function		Description	Script Calling
1	FreeIn (Connection)		Receives data and return an array	Supported, for example, FreeIn ('Connection_1');
	Parameter	Connection (In)	The name of the free protocol	

2	FreeOut(Connection , Array)		Sends data in an array	Supported, for example,
				var a = new Array();
				a[0] = 1;
				FreeOut('Connection_1',a);

	Parameter	Connection (In)	The name of the free protocol	
	Parameter	Array (In)	The data array to be sent	

3	AddCRC16 (Array)		Adds CRC16 check data to an array	Supported, for example,
				var a = new Array();
				a[0] = 1;
				a[2] = 2;
				AddCRC16(a);

	Parameter	Array	The array to be CRC checked	

4	AddCheckSum(Array)		Adds checksum to an array	Supported, for example,
				var a = new Array();
				a[0] = 1;
				a[2] = 2;
				AddCheckSum(a);

	Parameter	Array	The array to which the checksum to be added	

12.3.15 Timer

No.	Function		Description	Script Calling

1	SetInterval(expression,value)		Sets a timer to execute a function repeatedly	Supported, for example,
				function add()
				{
				var val=SmartTags("tag_1");
				SmartTags("tag_1")=val+1;
				}
				SetInterval(add , 1000);

	Parameter	expression	The function to be executed	
	Parameter	Value	The interval, in ms	
2	ClearInterval(timer)		Clears a timer, which must be used with SetInterval	Supported, for example, var timer=SetInterval(add , 1000); ClearInterval(timer);
	Parameter	Timer	The timer created by SetInterval	

3	SetTimeout(expression,value)		Sets the timer to execute a function only once	Supported, for example,
				function add()
				{
				var val=SmartTags("tag_1");
				SmartTags("tag_1")=val+1;
				}
				SetTimeout (add , 1000);

	Parameter	expression	The function to be executed	
	Parameter	Value	The interval, in ms	

No.	Function		Description	Script Calling
4	ClearTimeout(timer)		Clears a timer, which must be used with SetTimeout	Supported, for example, var timer= SetTimeout (add , 1000); ClearTimeout (timer);
	Parame-ter	The timer created by SetInterval	The timer created by SetTimeout	

12.3.16 Data Operation

No.	Function		Description	Script Calling

1	SwapByte(value)		Swaps high and low bytes	Supported, for example,
				var tag=-32768;
				tag =SwapByte(tag);
				Tag output value: 128

	Parame-ter	Value(In)	Object to be operated	

2	SwapWord(value)		Swaps high and low words	Supported, for example,
				var tag= -2147483648;
				tag =SwapWord(tag);
				Tag output value: 32768

	Parame-ter	Value(In)	Object to be operated	

3	LowByte(value)		Takes the value of the low byte	Supported, for example,
				var tag= -32640;
				tag =LowByte (tag);
				Tag output value: 128

	Parame-ter	Value(In)	The object to be operated	

4	HighByte(value)		Takes the value of the high byte	Supported, for example,
				var tag= -32640;
				tag =HighByte (tag);
				Tag output value: -32768

	Parame-ter	Value(In)	Object to be operated	

5	LowWord(value)		Takes the value of the low word	Supported, for example,
				var tag= -2147450880;
				tag =LowWord (tag);
				Tag output value: 32768

	Parame-ter	Value(In)	Object to be operated	

6	HighWord(value)		Takes the value of the high word	Supported, for example,
				var tag= -2147483648;
				tag =HighWord (tag);
				Tag output value: -2147483648

	Parame-ter	Value(In)	Object to be operated	

12.3.17 Reading and Writing an Array

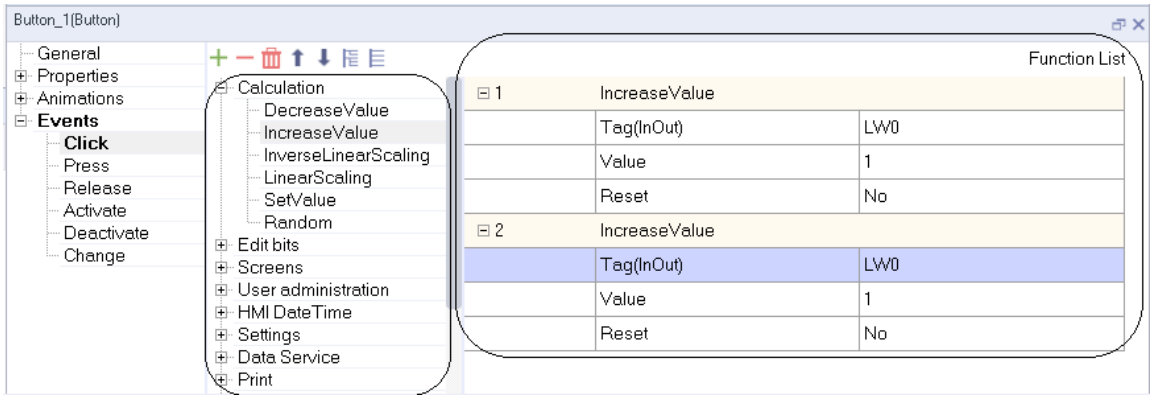
No.	Function		Description	Script Calling
1	ReadArray(tagSource, index, size, tagTarget)		Starting from "index", reads a certain number ("size") of elements from the array tag "tagSource" to the tag "tagTarget", and returns the number of successfully read elements	Supported, for example, SmartTags('result') = ReadArray(SmartTags('plc_array'),SmartTags('index'), SmartTags('count'), SmartTags ('local_ array')); Starting from "index", reads "count" elements from "plc_array" to "local_array" and saves the result to "result"
	Parameter	tagSource(In)	The array which stores the source data	
		Index(In)	Start index	
		size(In)	The number of elements	
		tagTarget(In)	The target array	
2	WriteArray(tagSource, index, size, tagTarget)		Starting from "index", writes "count" elements of "tagSource" to "tagTarget"	Supported, for example, SmartTags('result') = WriteArray(SmartTags('local_array'),SmartTags('plc_array'), SmartTags('index'),SmartTags ('co
	Parameter	tagSource(In)	The array which stores the source data	
		Index(In)	Start index	
		size(In)	The number of elements	
		tagTarget(In)	The target array	

12.4 Using Function list

12.4.1 Description

When a configuration event occurs, multiple system functions and scripts can be executed through the function list.

For example, using a function list in an event of a button:



Button	Function
+	Adds the selected function from the function selection list to the function execution list
−	Deletes the selected function from the execution list
🗑	Deletes all functions from the execution list

Button	Function
	Moves up the selected function in the execution list
	Moves down the selected function in the execution list
	Expands the execution list
	Collapses the execution list

12.4.2 Properties of Function List

Different HMI devices can be used for the same project. When you change an HMI device in a project, all system functions and scripts that are not supported by the selected HMI device are marked yellow. Unsupported system functions cannot be executed at run time.

The system functions and scripts in the function list are processed at run time from top to bottom. To avoid latency, system functions that require longer runtime (such as file manipulation) are processed simultaneously. For example, even if the previous system function is not completed, you can still execute the next system function.

12.5 Use of Scripts

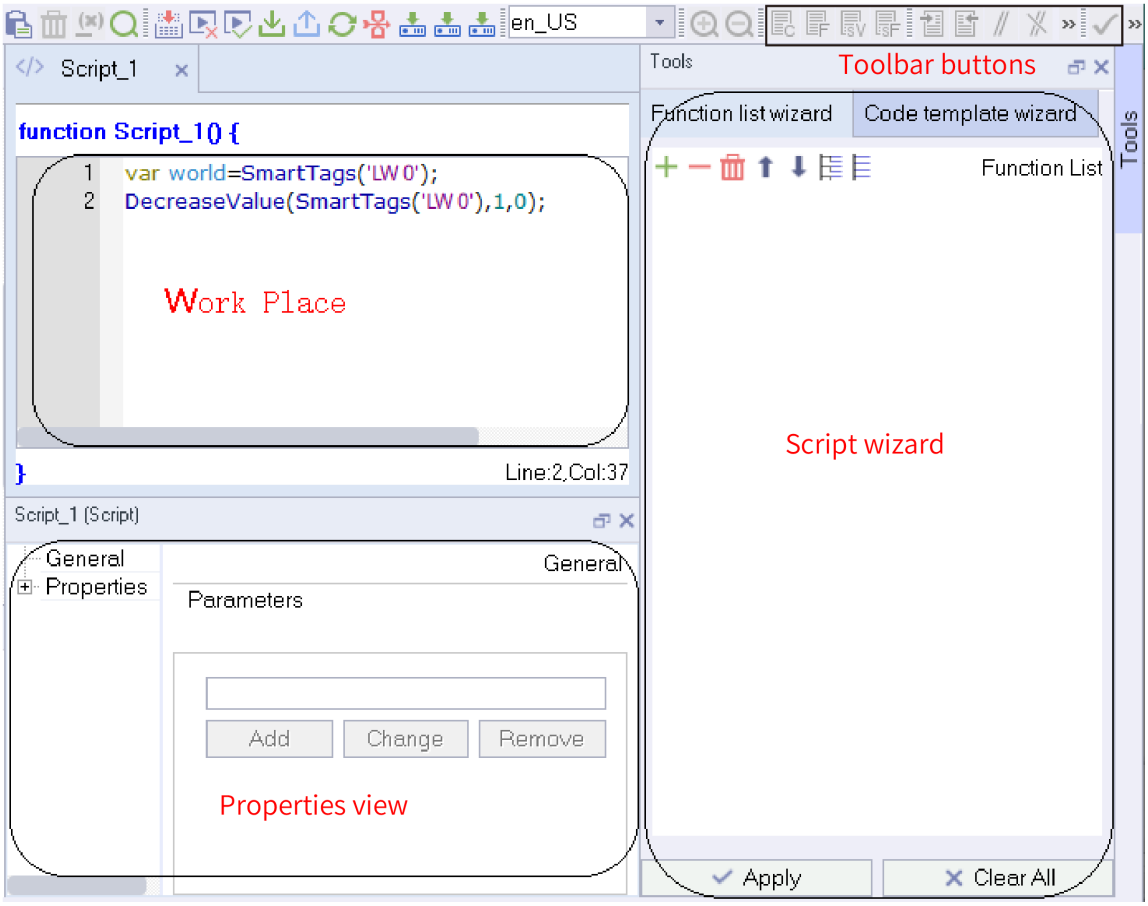
Using scripts in the runtime, you can achieve a single solution in a project, such as:

- Configuring advanced function list
  By calling system functions and other scripts in a script, you can use the script just like using a function list. Execute system functions and scripts in a script according to conditions, or repeat them. Then add the script to the function list.
- Creating a function
  Scripts are available throughout the project. You can use scripts as system functions. You can define send parameters and return values for these scripts. For example, you can use scripts to convert numeric values.

12.6 Script Editor

12.6.1 Overview

Creating a new script or opening an existing script will open the script editor.

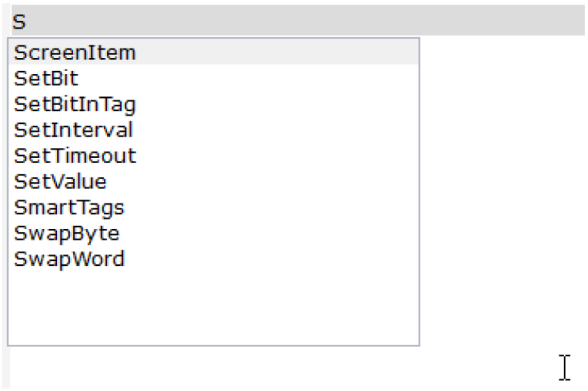


- Script toolbar
  Commands for synchronizing objects and tags and checking script syntax are located in the Scripts toolbar. Move the mouse to the corresponding tool icon and a tooltip will show. For example, if you click , the prompt "Use basic syntax" shows, it will become . In this mode, the script compiler does not strictly check the syntax and you can write classes and need not to define variables first.
- Workspace
  You can create and edit scripts in the workspace.
- Properties view
  You can configure scripts in the properties view. You can determine whether the script is a process or a function, and can also declare parameters for the script. In addition, to ensure that multiple scripts can run stably, IT7000 provides thread numbering function. If you want to run a large script, it is recommended to put it into a single thread.
- Script wizard
  In the script wizard, you can create system functions and scripts using the assigned parameters as in the function list. Archived system functions and scripts can also be transferred from the script wizard to active scripts. In this way, you only need to perform parameter allocation once.

12.6.2 Features

- Autocomplete

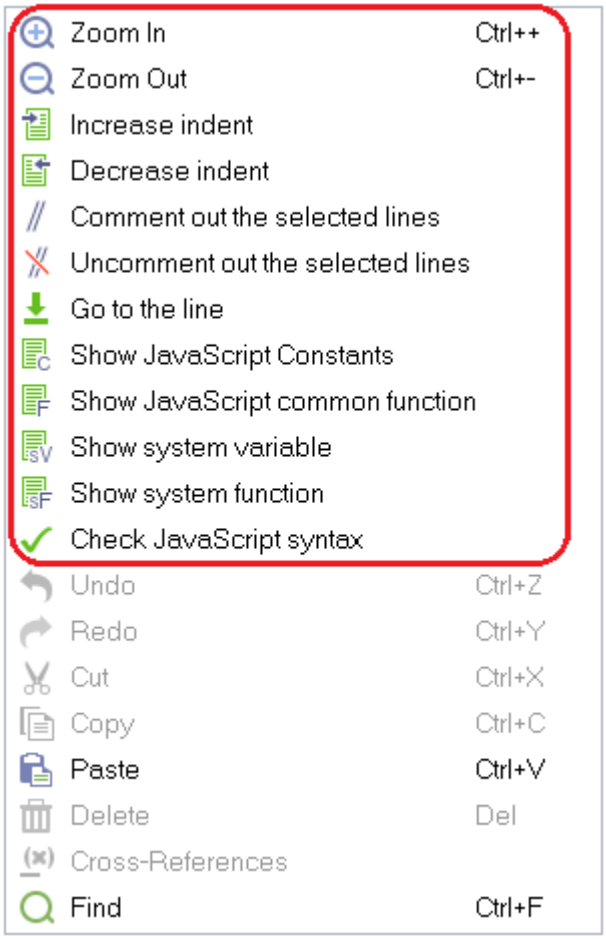
Autocomplete automatically lists all matching system functions in JS object model while you are typing:



- Syntax highlighting
  In the script editor, keywords are highlighted with different colors:

```
2  var arr = new Array(3);
3  arr[0] = "a"
4  arr[1] = "b"
5  arr[2] = "c"
6  arr.push( "d" );
7  arr.pop();
8  arr.toString(); //"a,b,c"
9  arr.reverse(); //arr[0]=c, arr[1]=b, arr[2]=a
10
```

- Object list
  You can use the context menu in the workspace to display script objects in a list.



12.7 Syntax

12.7.1 Overview

InoTouchPad uses JavaScript for coding to achieve higher flexibility and operation logic.

JavaScript is the most popular scripting language on the Internet. It is widely used in servers, PCs, laptops, tablets and smart phones. In a project, up to 100 scripts can be created, and scripts cannot be called each other.

12.7.2 Literal

In programming languages, fixed values are generally called literals:

The number literal can be an integer or a decimal, or a scientific notation (e), for example: 3.14, 1001, 123e5.

The **string literal can** use single or double quotation marks: "John Doe", 'John Doe'.

The **expression literal is** used for calculation: 5 + 6, 5 * 10.

The **array literal** defines an array: [40, 100, 1, 5, 25, 10].

The **object literal defines** an object: {firstName: "John", lastName: "Doe", age: 50, eyeColor: "blue"}.

The **function literal** defines a function: function myFunction (a, b) {return a * b;}.

12.7.3 Variable

In programming languages, variables are used to store data values. JavaScript uses the keyword "var" to define variables and the equal sign to assign a value to variables:

var x, length;

x = 5;

length = 6;

You can declare multiple variables in a single statement. The statement starts with "var" and separates variables with a commas:

var lastname="Doe", age=30, job="carpenter";

The declaration can be in multiple lines:

var lastname="Doe",

age=30,

job="carpenter";

Variables can be declared without a value. For these variables, their value is actually "undefined".

Variables can be accessed by the variable name. In instruction-style languages, variables are usually variable. A literal is a constant value. The variable name can be short (such as x and y) or more descriptive (such as age, sum and totalvolume).

Variables can start with a letter or with a $ and _ sign (though this is not recommended), and variable names are case sensitive (y and Y are different).

12.7.4 Operators

Table 12–1 Arithmetic operators

Operator	Description	Example
+	Addition	x=y+2 The + operator can be used to add up (concatenate) text values or string variables, for example: txt1= "what a"; txt2 = "nice day"; txt3=txt1txt2; The value of txt3 is "What a nice day". In addition, the + operator can also add strings and numbers, for example: z = "Hello" +5; The value of Z is "Hello5".
-	Subtraction	x=y-2
*	Multiplication	x=y*2
/	Division	x=y/2
%	Remainder	x=y%2

Operator	Description	Example
++	Increment	x=++y x=y++
–	Decrement	x=–y x=y–

Table 12–2 Assignment operators

Operator	Description	Example
=	Assignment	x=y
+=	Addition assignment	X+=y, that is x=x+y
-=	Subtraction assignment	X-=y, that is x=x-y
*=	Multiplication assignment	X*=y, that is x=x*y
/=	Division assignment	X-=y, that is x=x-y
%=	Remainder assignment	X%=y, that is x=x%y

Table 12–3 Comparison operators

Operator	Description	Comparison	Return Value
==	Equal	x==5	TRUE
		x==8	FALSE
===	Strict equal (both value and type are the same)	x===5	TRUE
		x===” 5”	FALSE
!=	Not equal	x!=8	TRUE
!==	Strict not equal (the operands are of the same type but not equal, or are of different type)	x!==5	FALSE
		x!==” 5”	TRUE
	Greater than	x>8	FALSE
<	Less than	x<8	TRUE
=	Greater than or equal	x>=8	FALSE
<=	Less than or equal	x<=8	TRUE

Table 12–4 Logical operators

Operator	Description	Example
&&	Logical AND	(x < 10 & & y > 1) is true
||	Logical OR	(x==5 || y==5) is false
!	Logical NOT	!(x==y) is true

Table 12–5 Conditional operators

Operator	Description	Example
? :	Conditional assignment	voteable = (age < 18)?"Too young":"The age is acceptable"; If the value of the variable "age" is less than 18, "Too young" is assigned to the variable "voteable", otherwise "The age is acceptable" is assigned.

12.7.5 JavaScript Statement Separation

JavaScript statements are separated by the semi colon. For example: x = 5 + 6;y = x * 10;

12.7.6 JavaScript Keywords

JavaScript keywords are used to identify the action to be performed, and like any other programming language, JavaScript reserves some keywords for its own use, as shown in the following figure:

abstract	else	instanceof	super
boolean	enum	int	switch
break	export	interface	synchronized
byte	extends	let	this
case	false	long	throw
catch	final	native	throws
char	finally	new	transient
class	float	null	true
const	for	package	try
continue	function	private	typeof
debugger	goto	protected	var
default	if	public	void
delete	implements	return	volatile
do	import	short	while
double	in	static	with

12.7.7 Comments

Comments are not executed. You can add comments to describe the code, or to improve the readability of the code. A single-line comment begins with //. For example:

//Title text

document.getElementById ("myH1"). innerHTML= "Welcome to my home page";

//Paragraph text

document.getelementbyid ("myP"). innerHTML= "This is my first paragraph. " ;

A multi-line comment begins with /* and ends with */. For example:

/* The following code will output a title and a paragraph which represent the beginning of the home page */

document.getElementById ("myH1"). innerHTML= "Welcome to my home page";

document.getelementbyid ("myP"). innerHTML= "This is my first paragraph. " ;

You can comment a statement so that it will not be executed. For example:

// document.getElementById ("myH1"). innerHTML= "Welcome to my home page";

document.getelementbyid ("myP"). innerHTML= "This is my first paragraph. " ;

You can put comments at the end of a line. For example:

Var x = 5; //declares variable x and assigns 5 to it

Var y = x + 2; //declares variable y and assigns x + 2 to it

12.7.8 Data Types

- **Data Types**

  1. Value types (primitive types): string, number, boolean, null, undefined, and symbol.
  2. Reference data types: object, array, and function.

  JavaScript is dynamically typed, which means that the same variable can be of different types:

  var x; //x is undefined

  var x = 5; //x is now an integer

  var x = "John" //x is now a string

- **Declaration**
  When you declare a new variable, you can use the keyword "new" to declare its type:

  var carname=new String;

  var x= new Number;

  var y= new Boolean;

  var cars= new Array;

  var person= new Object;

- **Number**
  There is only one number type in JavaScript. Numbers may or may not have a decimal point:

  Var x1 = 34.00; //with a decimal point

  Var x2 = 34; //without a decimal point

- **Boolean**
  A Boolean (logical) variable can have only two values: true or false.

  var x=true;

  var y=false;

- **String**
  A string can be any text in quotation marks. You can use single or double quotation marks:

  var carname="Volvo XC60";

  var carname='Volvo XC60';

- **Array**
  The following code creates an array named "cars":

  var cars=new Array();

  cars[0]="Saab";

  cars[1]="Volvo";

  cars[2]="BMW";

- **Object**
  An object is created by figure brackets. Inside the brackets, the properties of the object are defined in the form of "name:value". Properties are separated by a comma:

  var person={firstname:"John", lastname:"Doe", id:5566};

  The object (person) in the above example has three properties: firstname, lastname, and id. Spaces and line breaks do not matter. The declaration can be in multiple lines:

  var person={

  firstname : "John",

  lastname : "Doe",

  id : 5566

  };

  The object property can be accessed through the following ways:

  name=person.lastname;

  name=person["lastname"];

- **Undefined and Null**
  An "undefined" variable does not have a value. You can empty a variable by setting its value to null:

  cars=null;

  person=null;

12.7.9 Function

A function is a block of code wrapped in curly braces, preceded by the keyword function. When a function is called, the code block inside the function will be executed. A function can be called directly when an event occurs (such as when the user clicks a button) and can be called anywhere by JavaScript. JavaScript is case-sensitive. The keyword "function" must be lowercase. When you call the function, you must type the function name in the correct case.

- **Calling a function with arguments**
  When you call a function, you can pass values to it, which are called arguments. These parameters can be used in the function. You can send as many arguments as you like. These arguments are separated by commas (,): myFunction(argument1, argument2)

  When you declare a function, declare the arguments as variables:

  function myFunction(var1,var2){

```
//code block
}
```

Variables and arguments must appear in the same order. The first variable is the value of the first argument passed, and so on.

- **Functions with a return value**
  Sometimes, we want the function to return the value to the place where it was called. This can be achieved by using the return statement. When the return statement is used, the function stops executing and returns the specified value.

```
function myFunction(){
var x=5;
return x;
}
```

Note that the entire script does not stop executing. It continues from where the function was called.

- **Function expression**
  A function can be defined by an expression, and the function expression can be stored in a variable:

```
var x = function (a, b) {return a * b};
```

After the function expression is stored in the variable, the variable can also be used as a function:

```
var x = function (a, b) {return a * b};
var z = x(4, 3);
```

In addition, functions can also be defined through the built-in function constructor (Function ()):

```
var myFunction = new Function("a", "b", "return a * b");
var x = myFunction(4, 3);
```

12.7.10 Scope of Variables

- **Local**
  Variables which are declared within a function are local variables. Local variables: accessible only within a function. Because local variables are only effective within a function, different functions can use variables with the same name. Local variables are created when the function executes, and are automatically destroyed after the function is executed.

```
//The carName variable cannot be accessed here
function myFunction() {
var carName = "Volvo";
//The carName variable can be accessed here
}
```

- **Global variable**

Variable which are defined outside a function are global variables. The scope of global variables is global and all scripts and functions can use them.

```
var carName = " Volvo";

//The carName variable can be accessed here

function myFunction() {

//The carName variable can be accessed here

}
```

- **Variable lifecycle**
  The lifecycle of a JavaScript variable is initialized when it is declared. Local variables are destroyed after the function is executed, and global variables are destroyed after the page is closed.

12.7.11 Conditional Statements

In programming, you often need to perform different actions for different conditions. You can use conditional statements in your code to achieve that. JavaScript provides the following conditional statements:

If statement: The code is executed only if the condition is true

If... else statement: The if block is executed when the condition is true and the else block is executed when the condition is false

If.... else if.... else statement: Use this statement when there are more than two decisions

Switch statment: Use this statement when there are more than two decisions

- **If statement**: The code is executed only if the condition expression is true
  ```
  if (condition){

  //code block to be executed

  }
  ```
- If... else statement: The if block is executed when the condition is true and the else block is executed when the condition is false
  ```
  if (condition){

  //code block to be executed

  }

  else{

  //code block to be executed}
  ```
- If.... else if.... else statement: Use this statement when there are more than two decisions
  ```
  if (condition1){

  //code block to be executed}

  else if (condition2){

  //code block to be executed
  ```

}

else{

//code block to be executed

}

- **Switch statment**: Use this statement when there are more than two decisions
  switch(n) {

  case 1: //code block to be executed when n = 1

  break;

  case 2: //code block to be executed when n = 2

  break;

  default:

  //code to be executed when n is not equal to any case

  }

12.7.12 Loop Statement

If you want to run the same code over and over again, with different values each time, you should use the loop statement. JavaScript supports different types of loops:

The for loop can run a block of code for a number of times

The for/in loop can iterates the properties of an object

The while loop can repeat the specified block of code when the specified condition is true

The do/while loop can repeat the specified block of code when the specified condition is true

- **For statement**: runs a block of code for a number of times
  for (Statement 1; Statement 2; Statement 3) {

  Block of code to be executed}

  Statement 1 is executed before the code block, statement 2 defines the conditions for the loop, and statement 3 is executed after the code block is executed.

- **for/in statement:** Iterating the properties of an object
  var person={fname:"John",lname:"Doe",age:25};

  For (x in person)//x is the property name {

  txt=txt + person[x];

  }

- **While statement: A** while loop repeats a block of code when the specified condition is true
  While (condition) {

  Block of code to be executed

  }

- **do/while statement: The** do/while loop is a variant of the while loop. The loop executes the code block once regardless of the condition, and then repeats the code block if the condition is true
  do{

  Block of code to be executed

  }

  while (condition);

12.7.13 Break and Continue Statements

- **Break statement** can be used to jump out of a loop and continue to execute the code after the loop, if any.
  for (i=0;i<10;i++) {

  if (i==3) break;

  x=x + "The number is " + i + "<br>";

  }
- **Continue statement** can be used to force to execute the next iteration of the loop if the specified condition is met
  for (i=0;i<10;i++) {

  if (i==3) continue;

  x=x + "The number is " + i + "<br>";

  }

12.7.14 Letter Case

JavaScript is case-sensitive, so when writing JavaScript statements, pay attention to the CapsLock key. The function getElementById is different from getElementbyID, and the variable myVariable is also different from MyVariable.

12.7.15 JavaScript Character Set

JavaScript uses the Unicode character set, which covers all characters, including punctuation.

12.8 Creating a Script

12.8.1 Access to Tags

You can access external and internal tags created in the project in scripts. Tag values can be read or changed at run time. In addition, a local tag can be created in a script as a counter or buffer memory. The script gets the value of an external tag from the runtime memory. When runtime is started, the

actual value will be read from PLC and written into runtime memory. The tag value is then updated to the set period time. The script first accesses the tag value read from the PLC at the checkpoint of the previous scan period.

SmartTags ('tag name ') is used in the script to access a tag in the project:

```
var a = SmartTags( 'LW 0' );

SmartTags( 'LW 0' ) = 2018;
```

12.8.2 Calling System Functions in a Script

You can insert system functions into the script from the Script Wizard or the context menu List All System Functions, or drag tags directly into the script from the Details view.

- **Example 1: Decreasing the value of tag LW 0 by 1 by calling DecreaseValue**
  DecreaseValue(SmartTags('LW 0'),1);
- **Example 2: Setting a 1s timer to increase LW 0 by 1**
  function add()

  {

  var val= SmartTags('LW 0');

  SmartTags('LW 0')=val+1;

  }

  SetInterval(add , 1000);
- **Example 3: Swapping the high and low bytes of LW 0, which is of type Int16**
  var tag= SmartTags('LW 0');

  SmartTags('LW 0')=SwapByte(tag);

  For more system functions that can be called by a script, see section Categories of System Functions.

  The tag supports synchronous mode, and allows reading and writing of single tag of array.
- **Example 4**: The tag supports synchronous mode, and allows reading and writing of single tag of array.
  var i=0;

  for(i=0;i<10;i++)

  {

  SmartTags("LW 0",i,true)=i;

  }

12.8.3 Example of Dynamic Drawing

With the script, you can draw graphics on the canvas control.

- **Example 1: Drawing a clock**

Script:

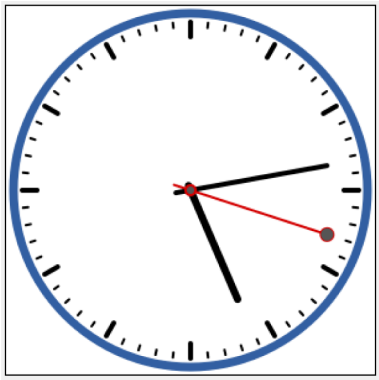
```
function clock(){
var now = new Date();
Var canvas = ScreenItem("Screen _1", "Canvas_1");
if( canvas.draw2d )
{
var ctx = canvas.draw2d(); var w = ctx.width();
var h = ctx.height();
var r = ( ( w > h ) ? h : w ) / 2 - 14;
ctx.manualUpdate = true;
ctx.save();
ctx.clearRect(0,0,w,h);
ctx.translate(w/2,h/2);
ctx.rotate(-Math.PI/2);
ctx.strokeStyle = "black";
ctx.fillStyle = "white";
ctx.lineWidth = 4;
ctx.lineCap = "round";

// Hour marks
ctx.save();
ctx.beginPath();
var i;
for (i=0;i<12;i++){
ctx.rotate(Math.PI/6);
ctx.moveTo(r-12,0);
ctx.lineTo(r,0);
}
ctx.stroke();
ctx.restore();
// Minute marks
ctx.save();
ctx.lineWidth = 2;
```

```
ctx.beginPath();
for (i=0;i<60;i++){
if (i%5!=0) {
ctx.moveTo(r-4,0);
ctx.lineTo(r,0);
}
ctx.rotate(Math.PI/30);
}
ctx.stroke();
ctx.restore();
var sec = now.getSeconds();
var min = now.getMinutes();
var hr = now.getHours();
hr = hr>=12 ? hr-12 : hr;

ctx.fillStyle = "black";

// write Hours
ctx.save();
ctx.rotate( hr*(Math.PI/6) + (Math.PI/360)*min + (Math.PI/21600)*sec );
ctx.lineWidth = 6;
ctx.beginPath();
ctx.moveTo( r - 140,0);
ctx.lineTo( r-40,0);
ctx.stroke();
ctx.restore();

// write Minutes
ctx.save();
ctx.rotate( (Math.PI/30)*min + (Math.PI/1800)*sec );
ctx.lineWidth = 4;
ctx.beginPath();
ctx.moveTo( r - 148,0);
```

```
ctx.lineTo( r - 24,0);
ctx.stroke();
ctx.restore();

// Write seconds
ctx.save();
ctx.rotate(sec * Math.PI/30);
ctx.strokeStyle = "#D40000";
ctx.fillStyle = "#D40000";
ctx.lineWidth = 2;
ctx.beginPath();
ctx.moveTo( r - 150,0);
ctx.lineTo( r - 20,0);
ctx.stroke();
ctx.beginPath();
ctx.arc(0,0,5,0,Math.PI*2,true);
ctx.fill();
ctx.beginPath();
ctx.arc(r - 20,0,5,0,Math.PI*2,true);
ctx.stroke();
ctx.fillStyle = "#555";
ctx.arc(0,0,3,0,Math.PI*2,true);
ctx.fill();
ctx.restore();

ctx.beginPath();
ctx.lineWidth = 7;
ctx.strokeStyle = '#325FA2';
ctx.arc(0,0,r + 7,0,Math.PI*2,true);
ctx.stroke();

ctx.restore();
ctx.doUpdate();
```

```
}
}
SetInterval( clock,1000 );
```

The result is as follows:



- **Example 2: Drawing a heart shape**

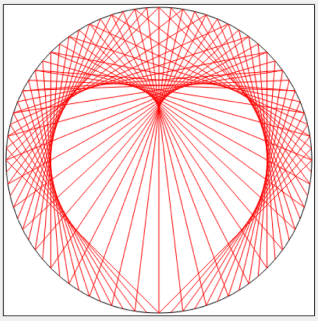
  Script:

```
Var canvas = ScreenItem("Screen _3", "Canvas_5");
var ctx = canvas.draw2d();
var sleep = 0;
var count = 0;
var rx = 10;
var ry = 10;
var radius = 10;
function deg2rad(x)
{
return x *Math.PI/180;
}

function drawCircle( x, y, r )
{
ctx.beginPath();
ctx.arc(x,y,r,0,Math.PI*2,true);
ctx.stroke();
}

function drawLine( x1, y1, x2, y2 )
{
```

```
ctx.beginPath();
ctx.moveTo(x1, y1);
ctx.lineTo(x2, y2);
ctx.stroke();
if (count++ == 82) {
count = 0;
sleep = 0;
}
}
this.line = drawLine;
function drawSide(begin1, end1, step1, begin2, step2 )
{
var d1, d2, x1, y1, x2, y2;

while ( begin1 != end1+step1 ) {
d1 = deg2rad(begin1);
d2 = deg2rad(begin2);
x1 = rx + radius * Math.cos(d1);
y1 = ry - radius * Math.sin(d1);
x2 = rx + radius * Math.cos(d2);
y2 = ry - radius * Math.sin(d2);
SetTimeout( 'line(' + x1 + ',' + y1 + ',' + x2 + ',' + y2 + ')', sleep*100 );
sleep++;
begin1 += step1;
begin2 += step2;
}
}
function drawHeart(){
Var canvas = ScreenItem("Screen _3", "Canvas_5");
if( canvas.draw2d )
{
ctx = canvas.draw2d();
var w = ctx.width();
```

```
var h = ctx.height();
ctx.clearRect(0,0,w,h);
rx = w / 2;
ry = h / 2;
radius = ( ( rx > ry ) ? ry : rx ) - 4;
ctx.strokeStyle = "black";
drawCircle( rx, ry, radius );
ctx.strokeStyle = "red";
drawSide(-90, 0, 4.5, 0, 4.5 );
drawSide(-90, -180, -4.5, -180, -4.5 );
drawSide(0, 180, 4.5, 90, 9 );
}
}
drawHeart();
```

The result is as follows:



12.8.4 Assigning Values to Array with Script

You can use SmartTags() in a for or while loop to assign values to an array

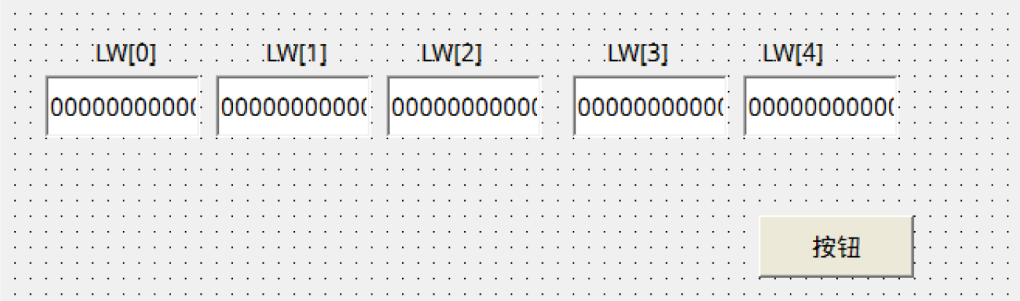
For example:

```
var i=0;
for ( i = 0; i < 5; ++i )
{
SmartTags( 'LW 0' ,i,true) = i;
}
```

While loop:

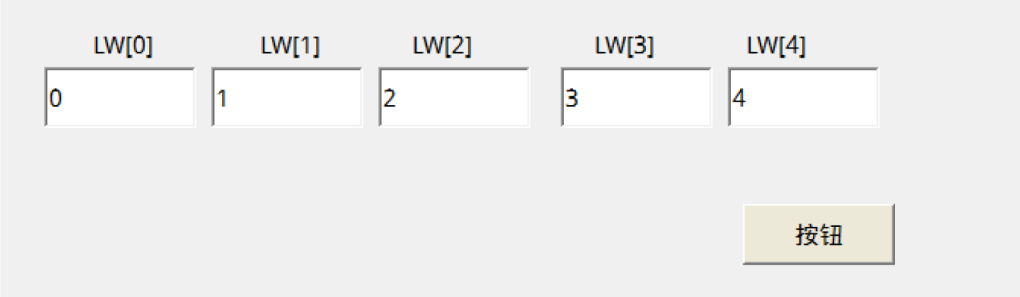
```
var j = 0;
```

```
while(j < 5)
{
SmartTags( 'LW 0', j,true) = j;
++j;
}
```



The configuration is shown in the example. The button executes the script mentioned above;

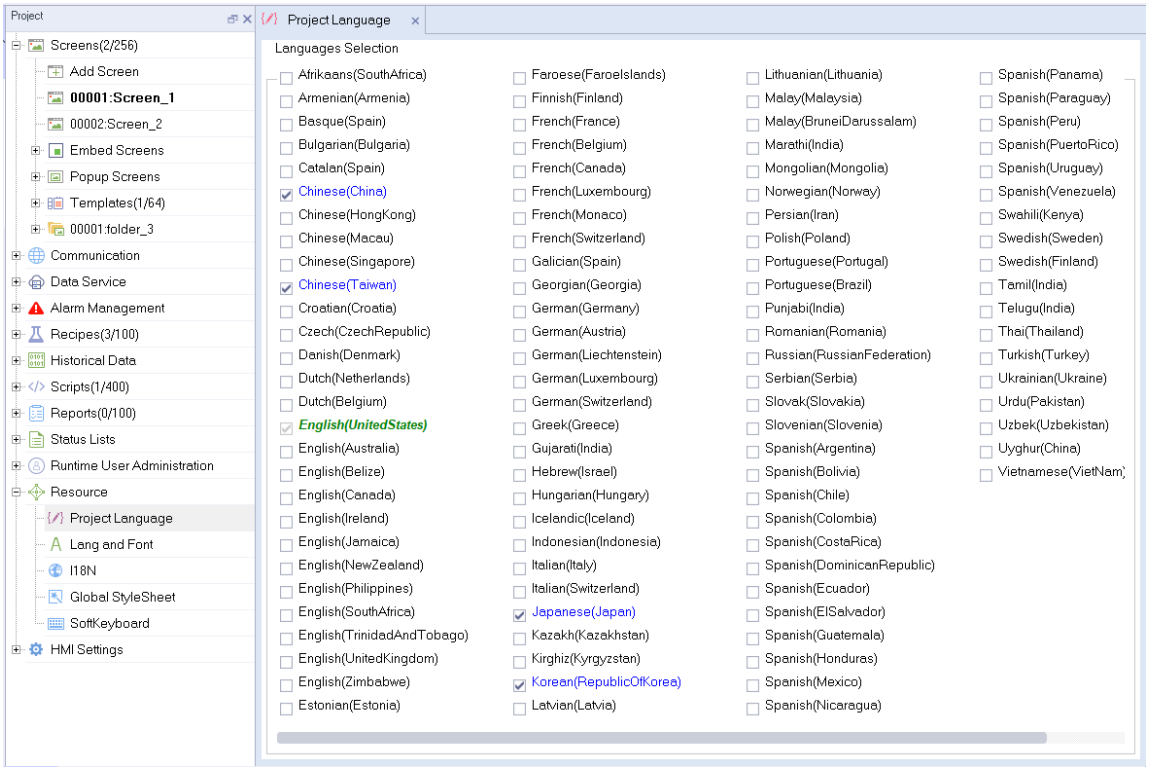
The display is as below:

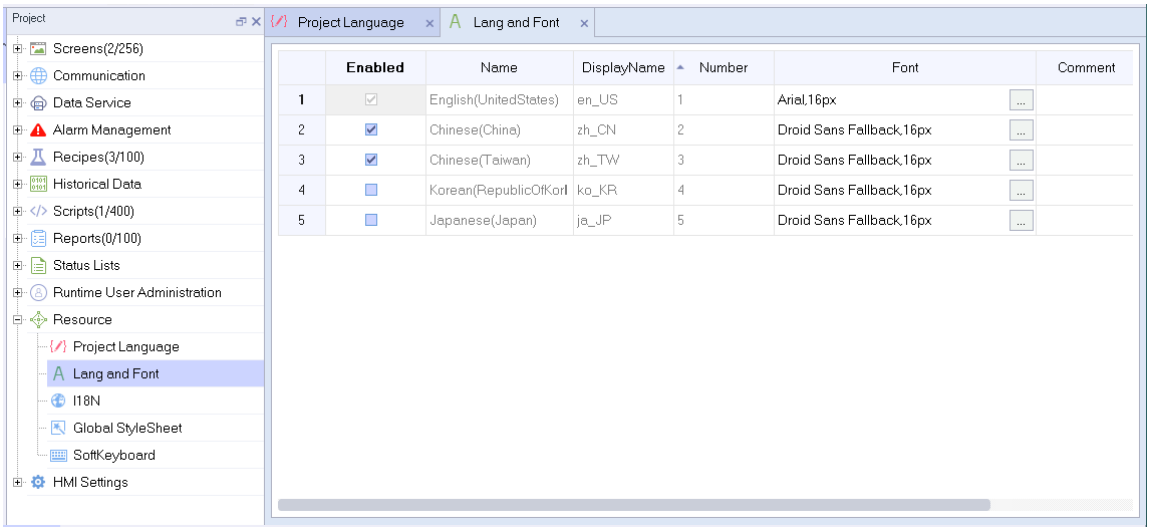


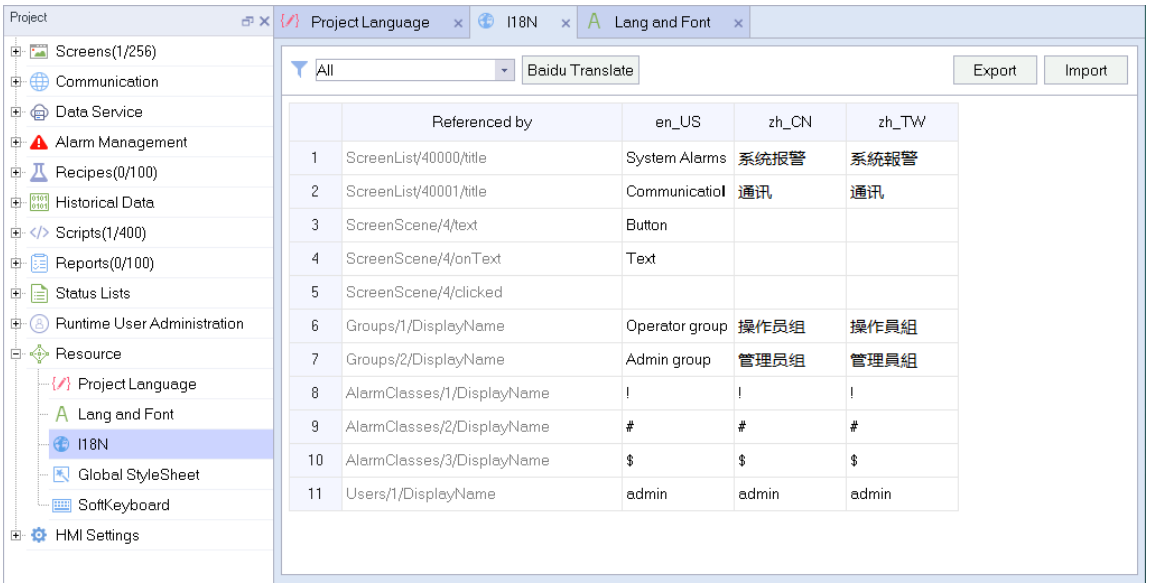
13 Multi-lingual Project

13.1 Description

In InoTouchPad, you can set up a project in different languages through machine translation, so that users in different countries can use the same project, which can greatly reduce the development period. To enable this multilingual function, you need to set the project language, language and font, and internationalization of the resource module.

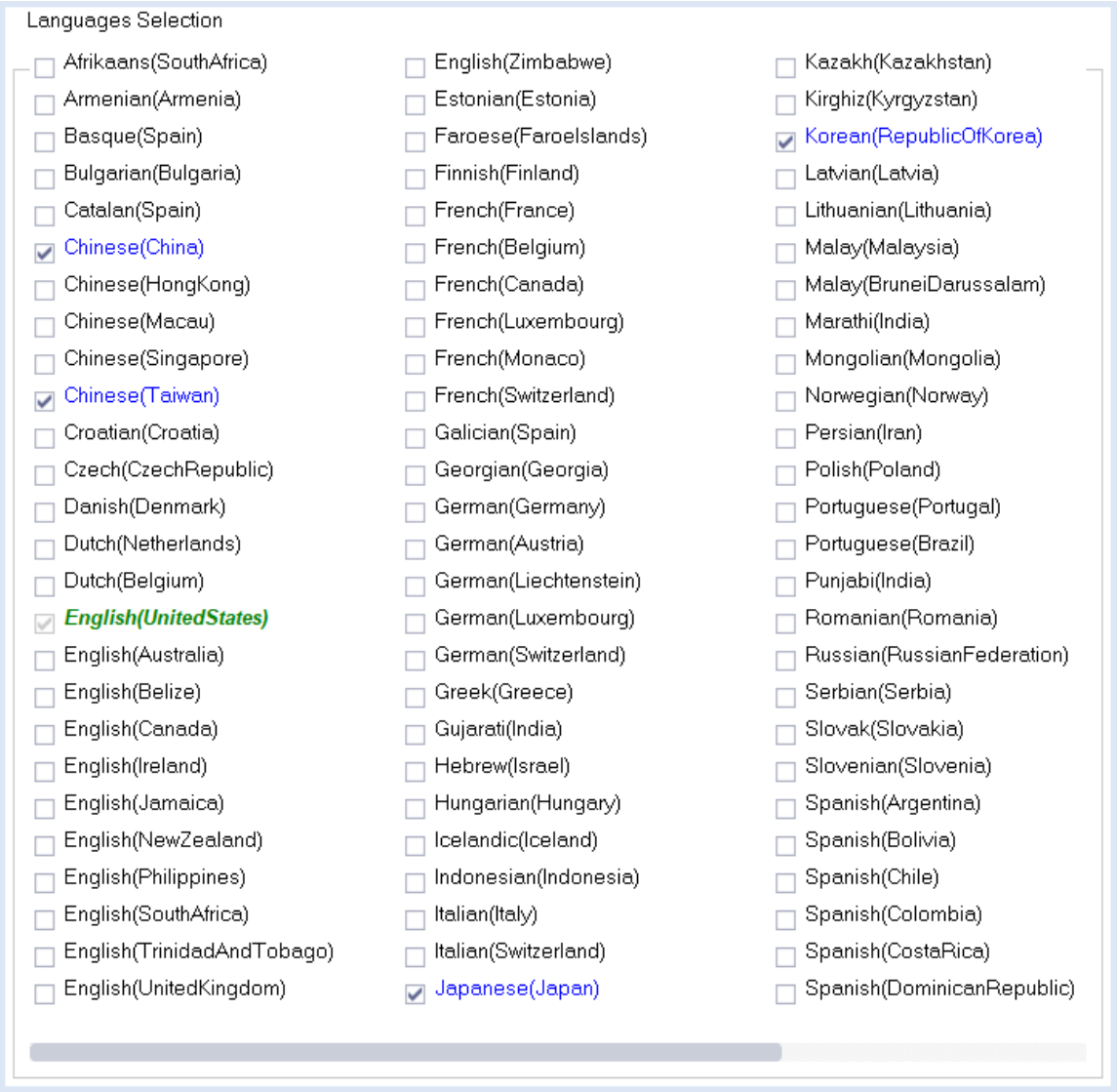






13.2 Project Language

The project language setting provides over a hundred languages of your choice, as shown in the following figure:



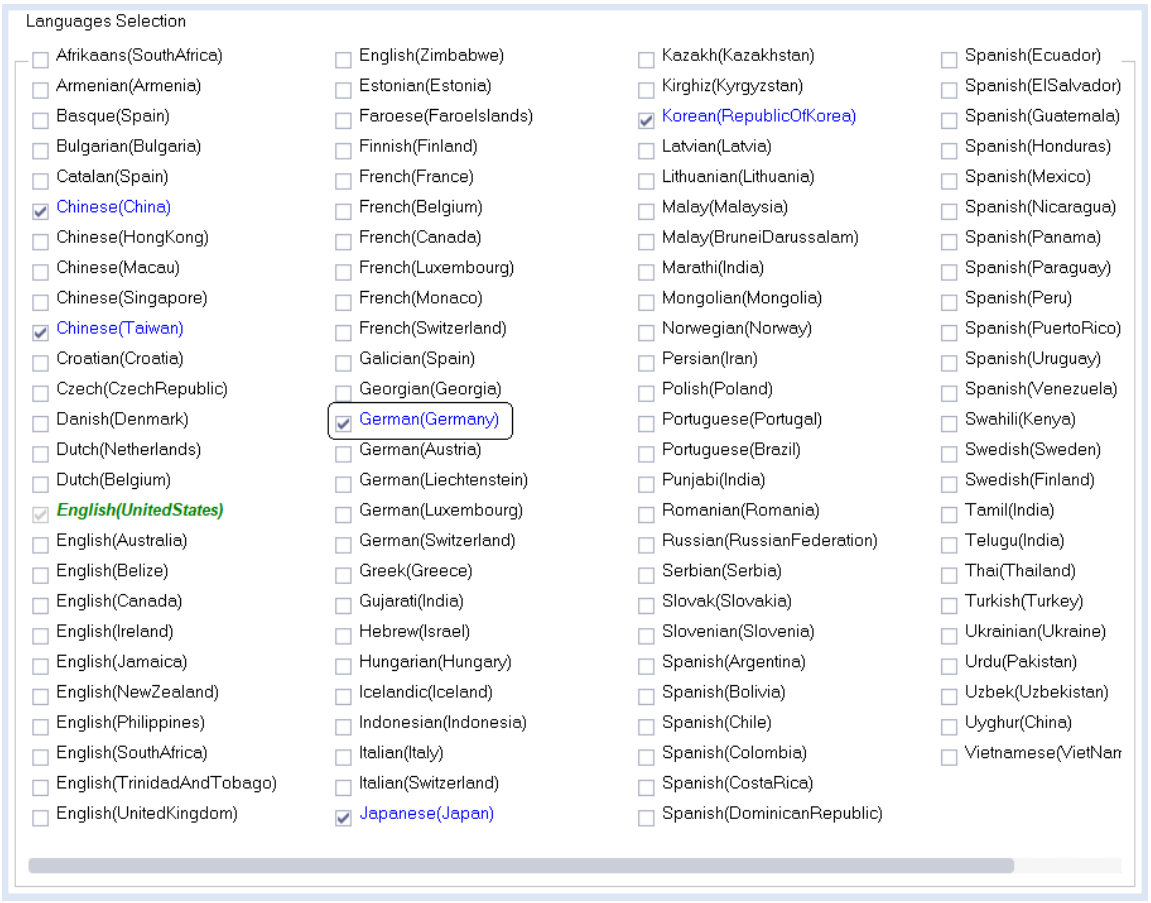
Five languages, Chinese (China), Chinese (Taiwan), English (United States), Japanese (Japan) and Korean (Korea), are checked by default, which indicates that these languages are activated. The detailed settings of a language can be found in the Language and Font editor. If you need to develop a project for a specified language, you can configure the text of that language in I18N by choosing that language and enabling it in the Language and Font editor (see *"13.3 Language and Font Setting" on page 289* for details).

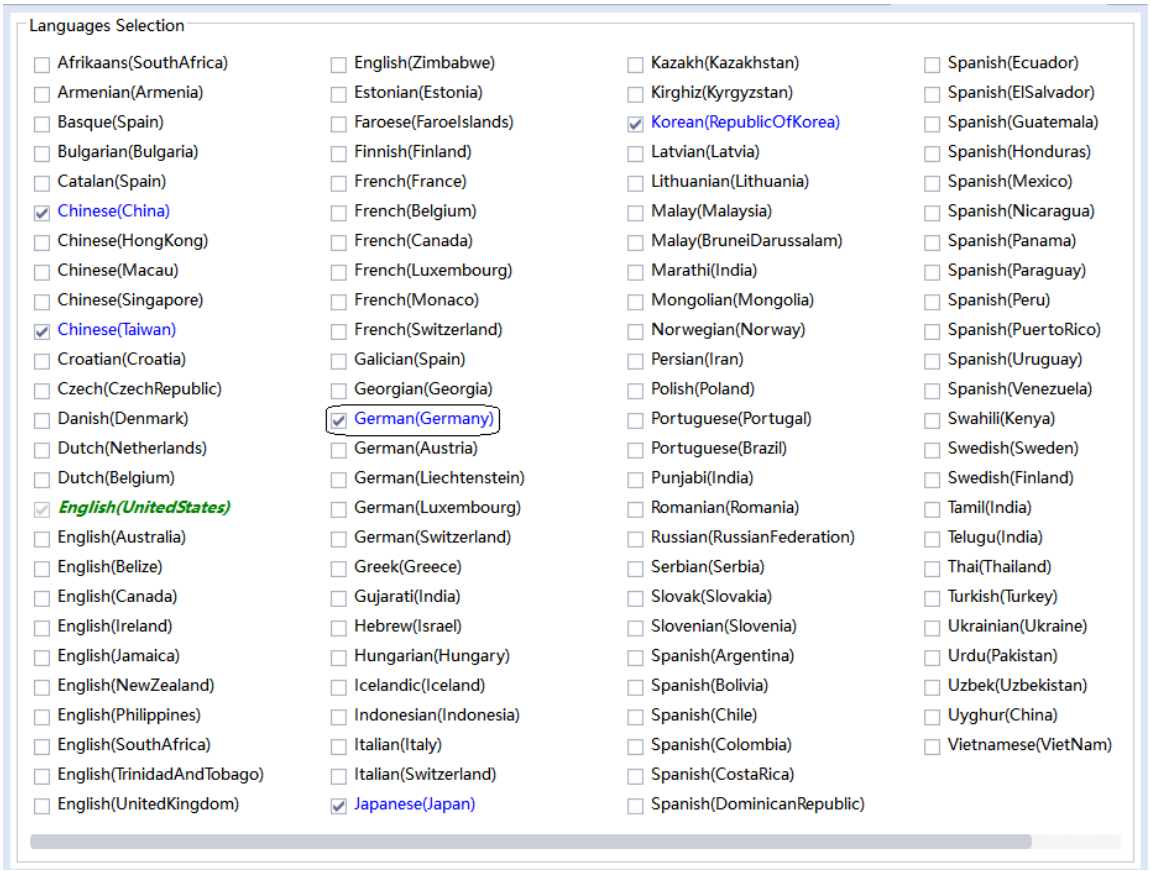
13.3 Language and Font Setting

The language and font workspace is used to set the properties of the selected project language. Double-click Lang and Font in Resource to open the language and font workspace, as shown in the following figure:

	Enabled	Name	DisplayName	Number	Font	Comment
1	☑	English(UnitedStates)	en_US	1	Arial,16px	
2	☑	Chinese(China)	zh_CN	2	Droid Sans Fallback,16px	
3	☑	Chinese(Taiwan)	zh_TW	3	Droid Sans Fallback,16px	
4	☐	Korean(RepublicOfKorl	ko_KR	4	Droid Sans Fallback,16px	
5	☐	Japanese(Japan)	ja_JP	5	Droid Sans Fallback,16px	

If you need to add a language to this workspace, you can check the language in Project Language and then the language will be automatically added to the language and font workspace. Take adding German (Germany) as an example:





In Project Language, check German (Germany), and a new line for German (German) is added in the workspace. If you want to remove the language, uncheck it in Project Language.

	Enabled	Name	DisplayName	Number	Font	Comment
1	☑	English(UnitedStates)	en_US	1	Arial,16px	
2	☑	Chinese(China)	zh_CN	2	Droid Sans Fallback,16px	
3	☑	Chinese(Taiwan)	zh_TW	3	Droid Sans Fallback,16px	
4	☐	Korean(RepublicOfKorl	ko_KR	4	Droid Sans Fallback,16px	
5	☐	Japanese(Japan)	ja_JP	5	Droid Sans Fallback,16px	
6	☑	German(Germany)	de_DE	6	Droid Sans Fallback,16px	

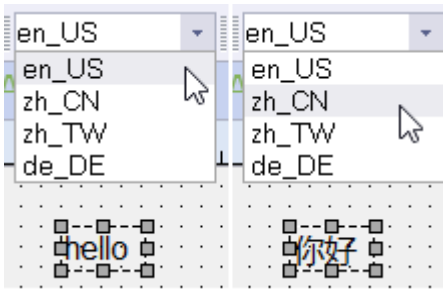
You can enable/disable the languages and set their font and comment. But the name and display name are predefined and cannot be edited. After setting the required language attributes, you need to check Enable to use the language in the project.

The name is the name displayed in Project Language, and the display name is the abbreviation of the name, which is mainly used by internationalization, internationalization switchover preview and project start language setting:

1. Internationalization table

	Referenced by	en_US	zh_CN	zh_TW	de_DE
1	ScreenList/40000/title	System Alarms	系统报警	系統報警	
2	ScreenList/40001/title	Communicatiol	通讯	通訊	
3	Groups/1/DisplayName	Operator group	操作员组	操作員組	
4	Groups/2/DisplayName	Admin group	管理员组	管理員組	

2. Internationalization switchover preview

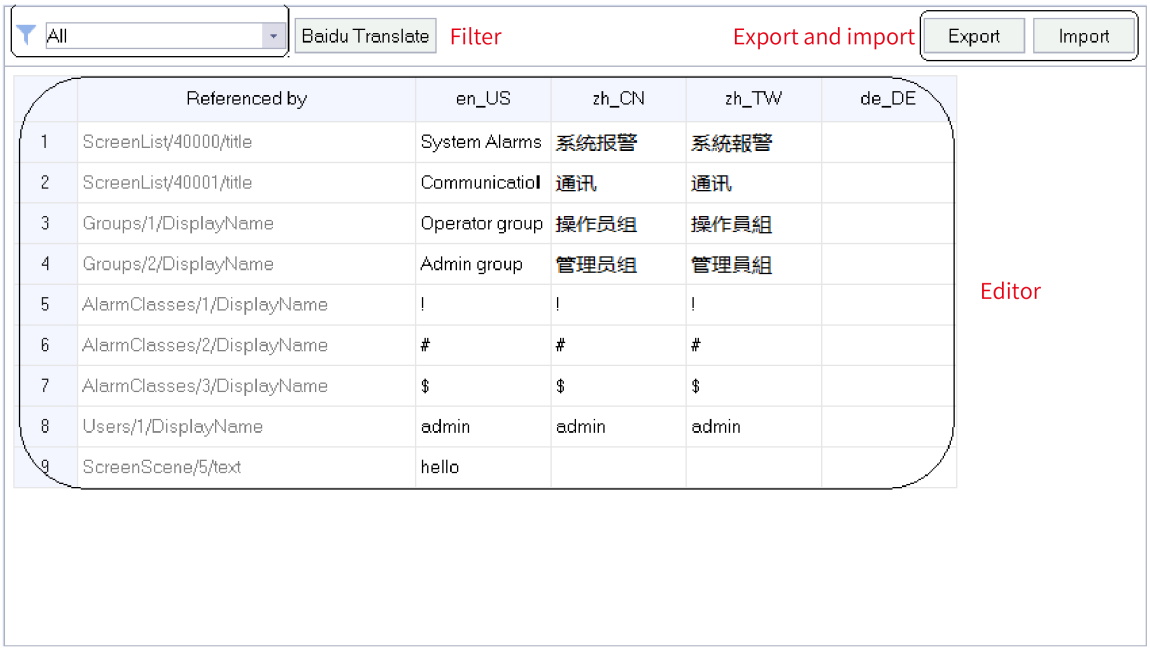


3. Project start language setting

13.4 Internationalization

13.4.1 Overview

The Internationalization workspace is mainly used to edit project texts that need to be translated into multiple languages. Go to Resources→Internationalization and double click to open the workspace, as shown in the following figure:

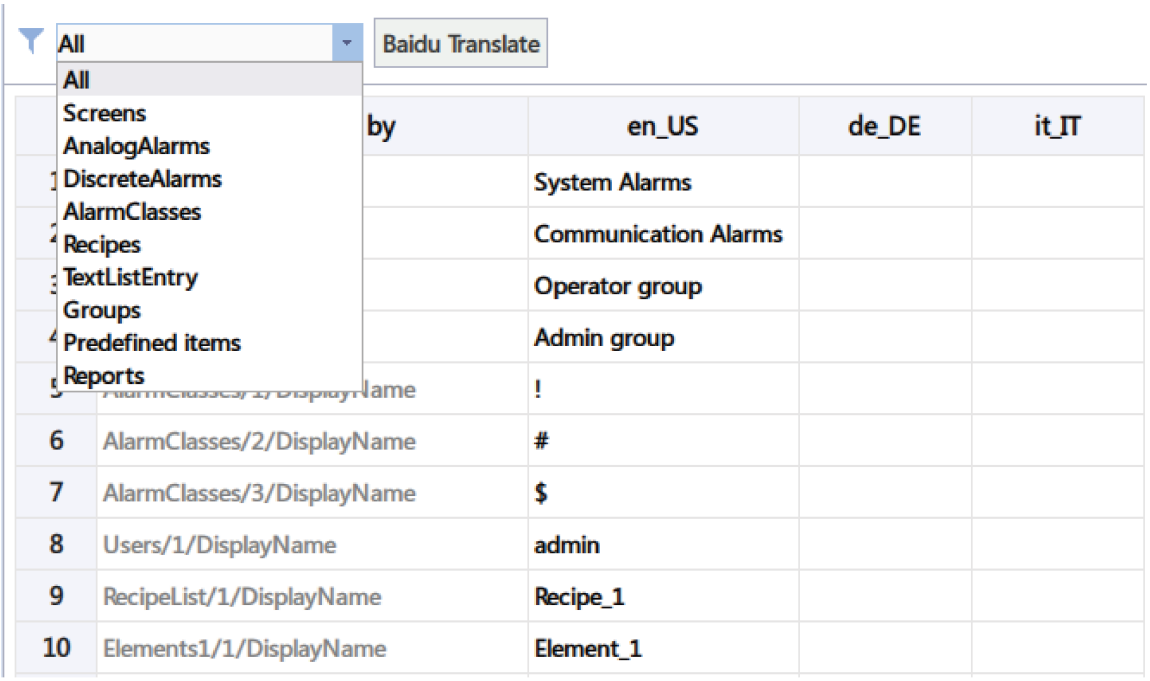


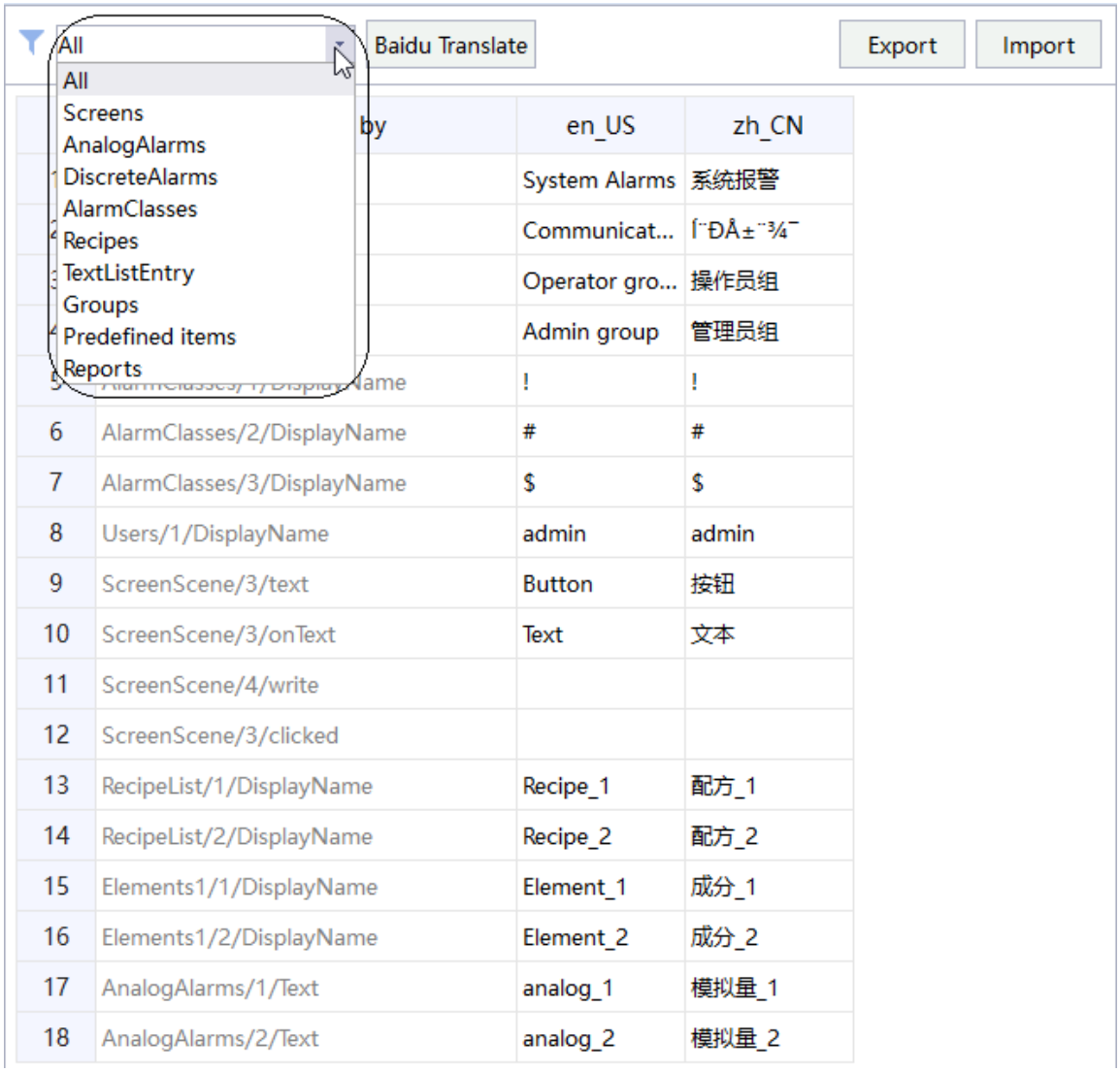
	Referenced by	en_US	zh_CN	zh_TW	de_DE
1	ScreenList/40000/title	System Alarms	系统报警	系统報警	
2	ScreenList/40001/title	Communicatiol	通讯	通讯	
3	Groups/1/DisplayName	Operator group	操作员组	操作員組	
4	Groups/2/DisplayName	Admin group	管理员组	管理員組	
5	AlarmClasses/1/DisplayName	!	!	!	
6	AlarmClasses/2/DisplayName	#	#	#	
7	AlarmClasses/3/DisplayName	$	$	$	
8	Users/1/DisplayName	admin	admin	admin	
9	ScreenScene/5/text	hello			

This workspace can display all the text that needs to be translated in the project, and the language that can be translated is determined by "Project Language" and "Language and Font" in the resource. For details, see *"13.3 Language and Font Setting" on page 289*.

13.4.2 Filter

There is a filter in the upper left corner of the I18N editor, which classifies all the texts that can be translated in the project into screens, analog alarms, discrete alarms, recipes, list entries, user groups, predefined items and reports, as shown in the following figure:





	Referenced by	en_US	zh_CN
		System Alarms	系统报警
		Communicat...	Ï¨ÐÅ±¨¾¯
		Operator gro...	操作员组
		Admin group	管理员组
5	AlarmClasses/1/DisplayName	!	!
6	AlarmClasses/2/DisplayName	#	#
7	AlarmClasses/3/DisplayName	$	$
8	Users/1/DisplayName	admin	admin
9	ScreenScene/3/text	Button	按钮
10	ScreenScene/3/onText	Text	文本
11	ScreenScene/4/write		
12	ScreenScene/3/clicked		
13	RecipeList/1/DisplayName	Recipe_1	配方_1
14	RecipeList/2/DisplayName	Recipe_2	配方_2
15	Elements1/1/DisplayName	Element_1	成分_1
16	Elements1/2/DisplayName	Element_2	成分_2
17	AnalogAlarms/1/Text	analog_1	模拟量_1
18	AnalogAlarms/2/Text	analog_2	模拟量_2

When you select All, all the translated text in the modules except the predefined items is displayed:

	Referenced by	en_US	de_DE	it_IT
1	ScreenList/40000/title	System Alarms		
2	ScreenList/40001/title	Communication Alarms		
3	Groups/1/DisplayName	Operator group		
4	Groups/2/DisplayName	Admin group		
5	AlarmClasses/1/DisplayName	!		
6	AlarmClasses/2/DisplayName	#		
7	AlarmClasses/3/DisplayName	$		
8	Users/1/DisplayName	admin		
9	RecipeList/1/DisplayName	Recipe_1		
10	Elements1/1/DisplayName	Element_1		

All | Baidu Translate

	Referenced by	en_US	zh_CN
1	ScreenList/40000/title	System Alarms	系统报警
2	ScreenList/40001/title	Communicat...	Í¨ÐÅ±¨¾¯
3	Groups/1/DisplayName	Operator gro...	操作员组
4	Groups/2/DisplayName	Admin group	管理员组
5	AlarmClasses/1/DisplayName	!	!
6	AlarmClasses/2/DisplayName	#	#
7	AlarmClasses/3/DisplayName	$	$
8	Users/1/DisplayName	admin	admin
9	ScreenScene/3/text	Button	按钮
10	ScreenScene/3/onText	Text	文本
11	ScreenScene/4/write		
12	ScreenScene/3/clicked		
13	RecipeList/1/DisplayName	Recipe_1	配方_1
14	RecipeList/2/DisplayName	Recipe_2	配方_2
15	Elements1/1/DisplayName	Element_1	成分_1
16	Elements1/2/DisplayName	Element_2	成分_2
17	AnalogAlarms/1/Text	analog_1	模拟量_1
18	AnalogAlarms/2/Text	analog_2	模拟量_2

When you select Screens, only the text of the screen module that needs to be translated is displayed:

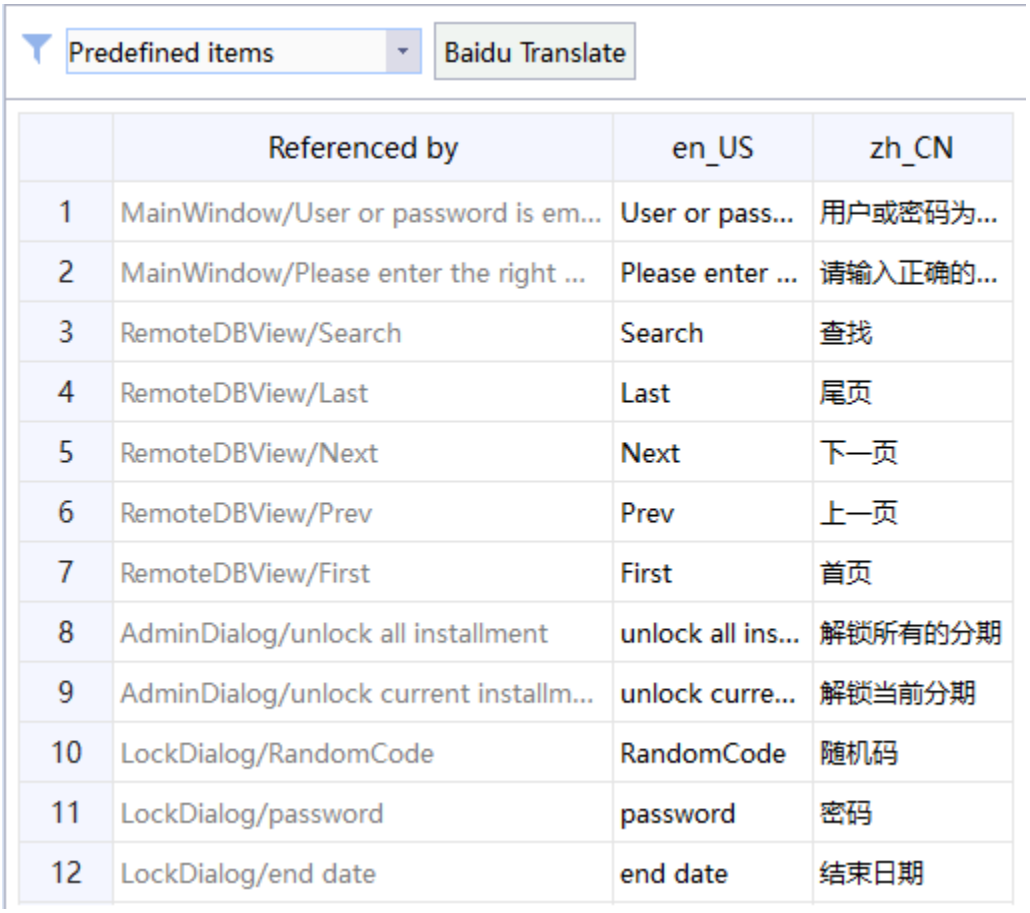
Screens | Screen_1---1 | Baidu Translate

	Referenced by	en_US	zh_CN	zh_TW
1	ScreenScene/2/text	Use internationalization	使用国际化	使用國際化

Screens | Screen_1---1 | Baidu Translate

	Referenced by	en_US	zh_CN	zh_TW
1	ScreenScene/16/text	Use I18N	使用国际化	使用國際化

When you select predefined items, the predefined items are displayed (predefined items can also be modified):

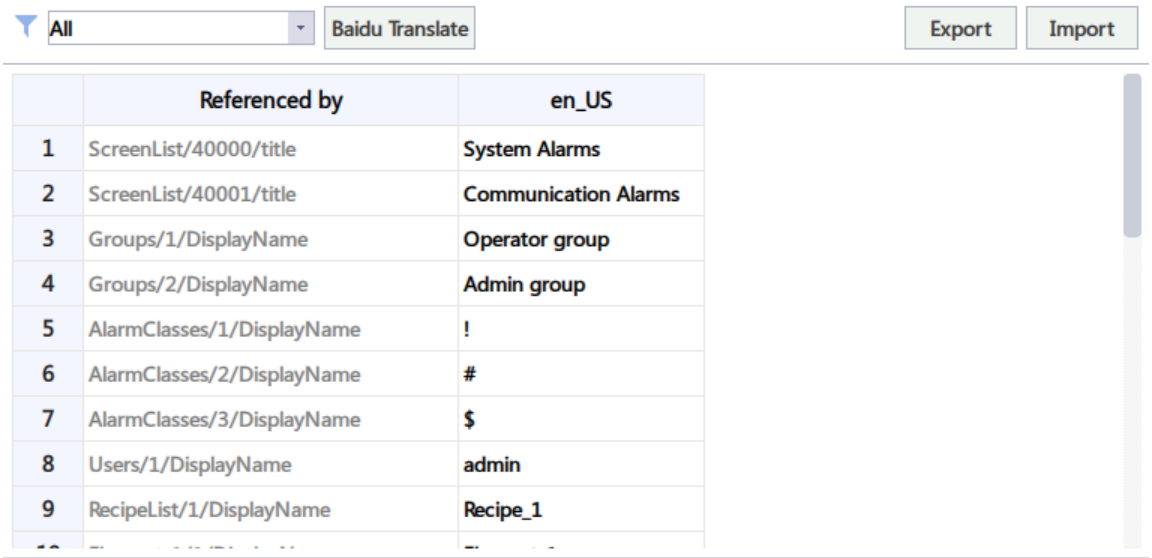


	Referenced by	en_US	zh_CN
1	MainWindow/User or password is em...	User or pass...	用户或密码为...
2	MainWindow/Please enter the right ...	Please enter ...	请输入正确的...
3	RemoteDBView/Search	Search	查找
4	RemoteDBView/Last	Last	尾页
5	RemoteDBView/Next	Next	下一页
6	RemoteDBView/Prev	Prev	上一页
7	RemoteDBView/First	First	首页
8	AdminDialog/unlock all installment	unlock all ins...	解锁所有的分期
9	AdminDialog/unlock current installm...	unlock curre...	解锁当前分期
10	LockDialog/RandomCode	RandomCode	随机码
11	LockDialog/password	password	密码
12	LockDialog/end date	end date	结束日期

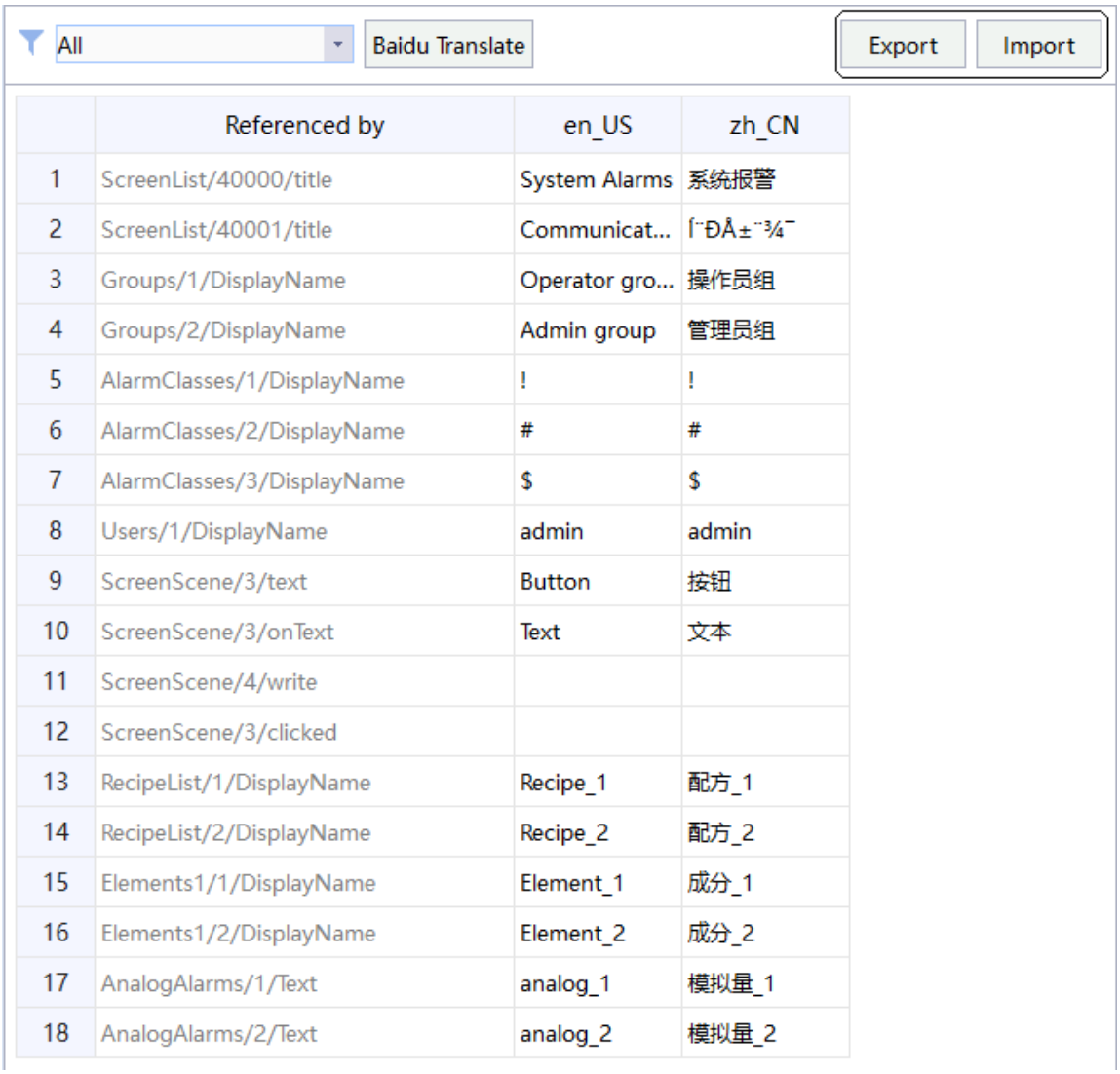
The other modules are similar. You only need to select a module to show the translation text involved in this module.

13.4.3 Importing and Exporting Internationalization Text

You can import and export translation text in the upper right corner of the I18N editor:

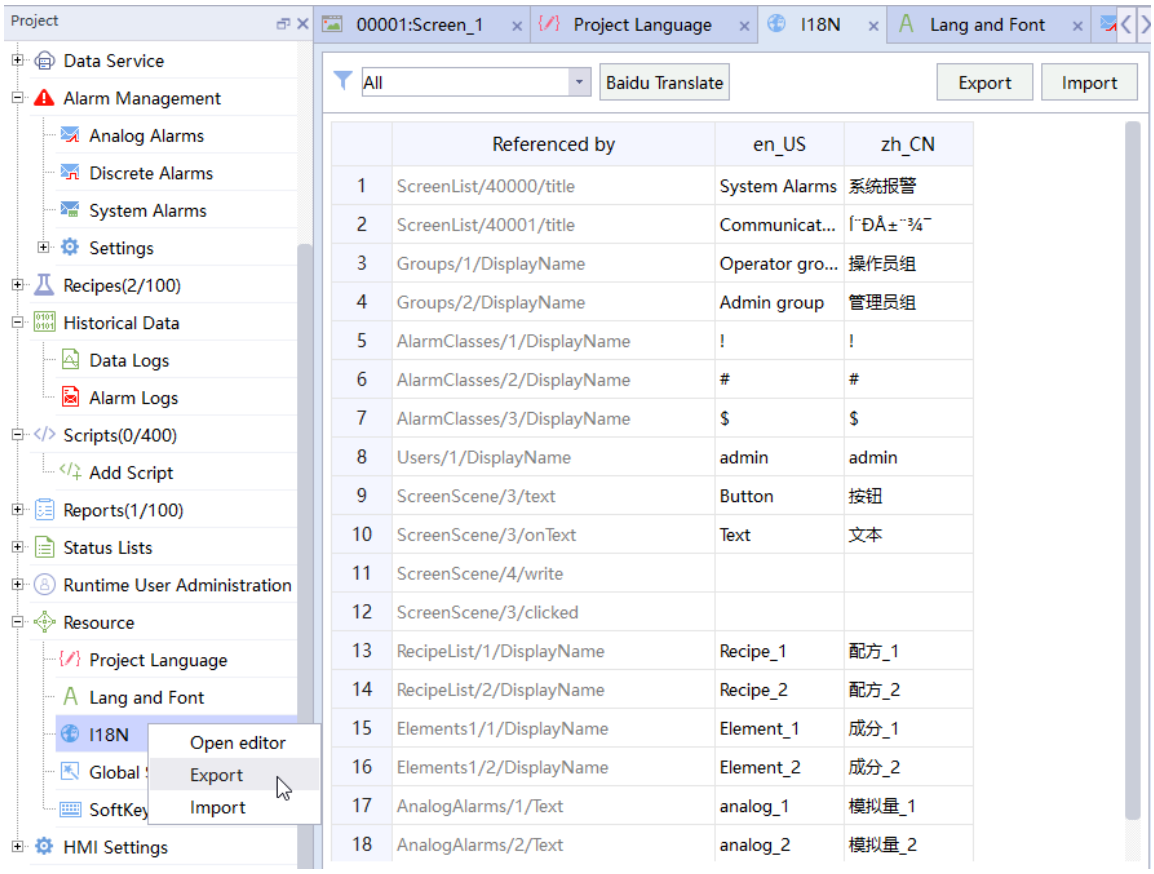


	Referenced by	en_US
1	ScreenList/40000/title	System Alarms
2	ScreenList/40001/title	Communication Alarms
3	Groups/1/DisplayName	Operator group
4	Groups/2/DisplayName	Admin group
5	AlarmClasses/1/DisplayName	!
6	AlarmClasses/2/DisplayName	#
7	AlarmClasses/3/DisplayName	$
8	Users/1/DisplayName	admin
9	RecipeList/1/DisplayName	Recipe_1

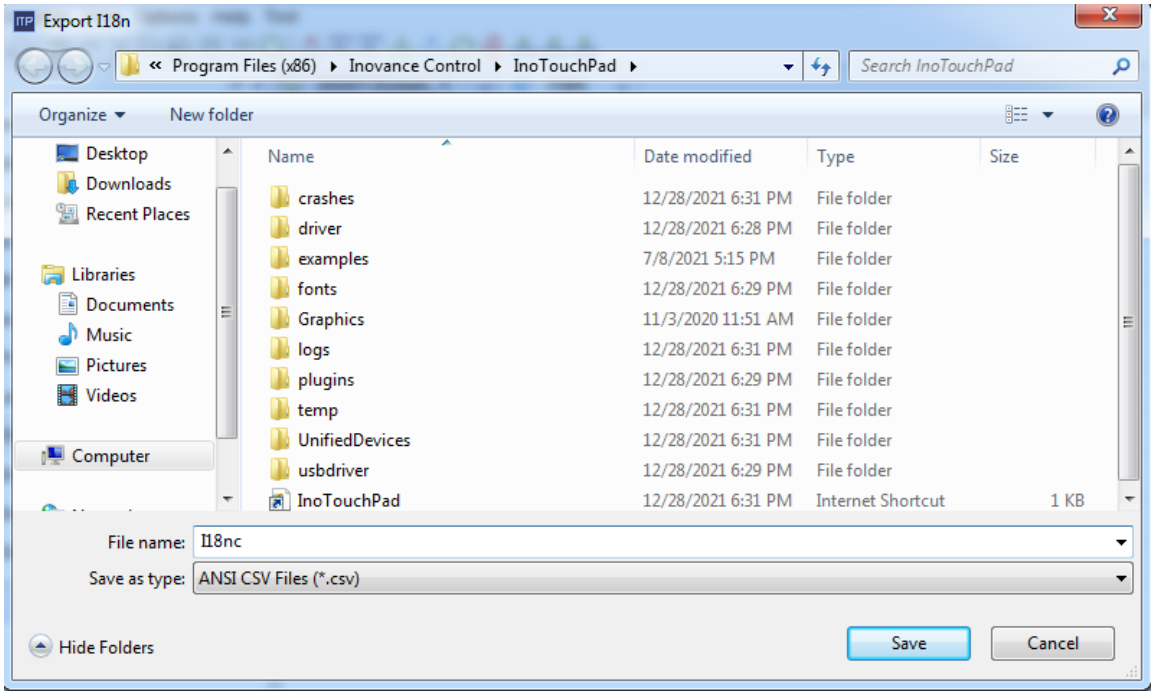


	Referenced by	en_US	zh_CN
1	ScreenList/40000/title	System Alarms	系统报警
2	ScreenList/40001/title	Communicat...	Í¨ÐÅ±¨¾¯
3	Groups/1/DisplayName	Operator gro...	操作员组
4	Groups/2/DisplayName	Admin group	管理员组
5	AlarmClasses/1/DisplayName	!	!
6	AlarmClasses/2/DisplayName	#	#
7	AlarmClasses/3/DisplayName	$	$
8	Users/1/DisplayName	admin	admin
9	ScreenScene/3/text	Button	按钮
10	ScreenScene/3/onText	Text	文本
11	ScreenScene/4/write		
12	ScreenScene/3/clicked		
13	RecipeList/1/DisplayName	Recipe_1	配方_1
14	RecipeList/2/DisplayName	Recipe_2	配方_2
15	Elements1/1/DisplayName	Element_1	成分_1
16	Elements1/2/DisplayName	Element_2	成分_2
17	AnalogAlarms/1/Text	analog_1	模拟量_1
18	AnalogAlarms/2/Text	analog_2	模拟量_2

You can also right-click I18N in the project tree view and select Import or Export:

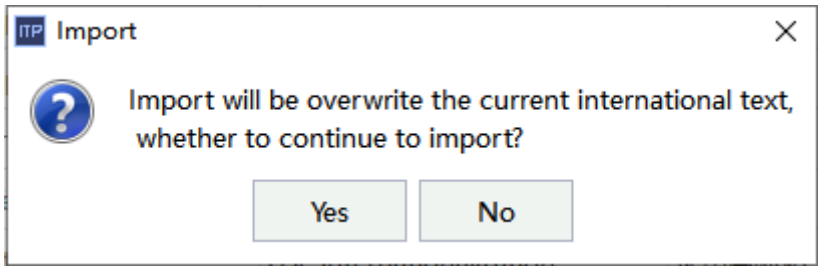


After selecting the module in the filter, click Export to pop up the following window:

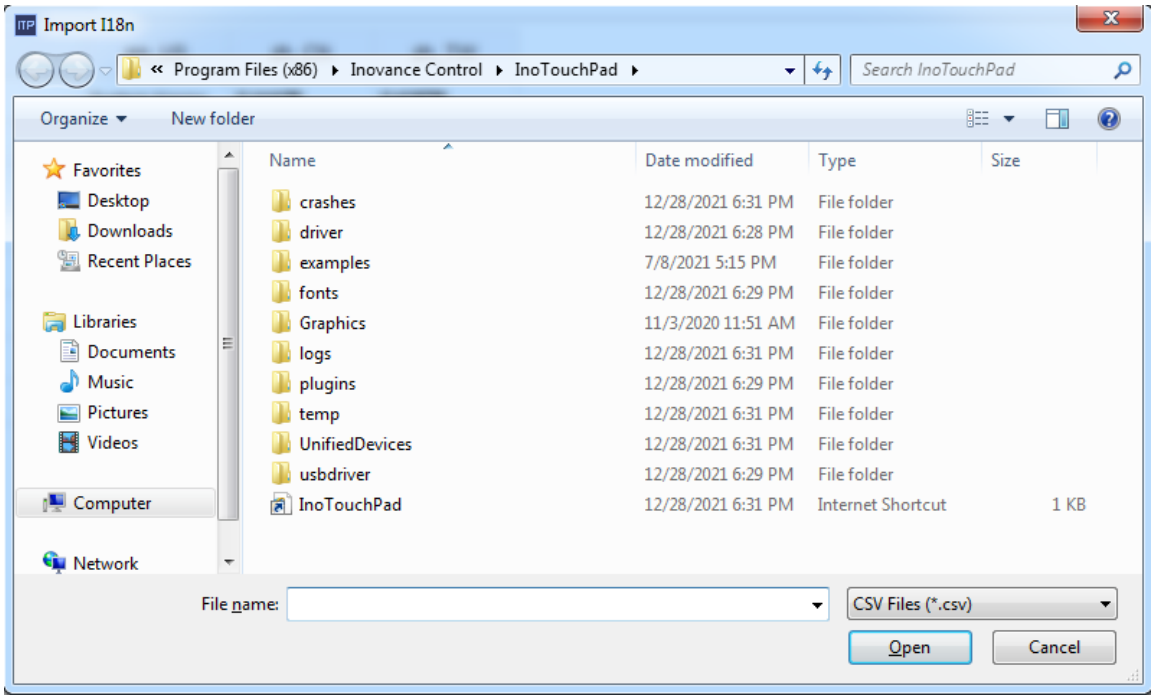


Enter the file name, and then click Save to export the translation text displayed in the current module to a *. csv file.

The exported file is generally used for external editing. After the file is edited, click Import, and the following message box will appear:



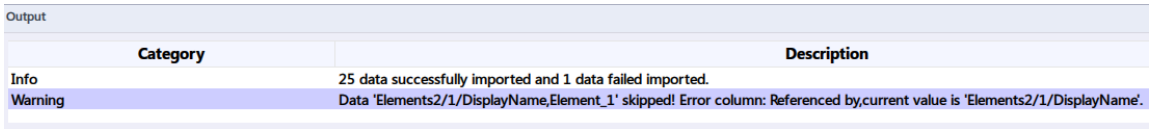
Click Yes, and the file import window will pop up:



Select the csv file to be imported and click Open. Notes:

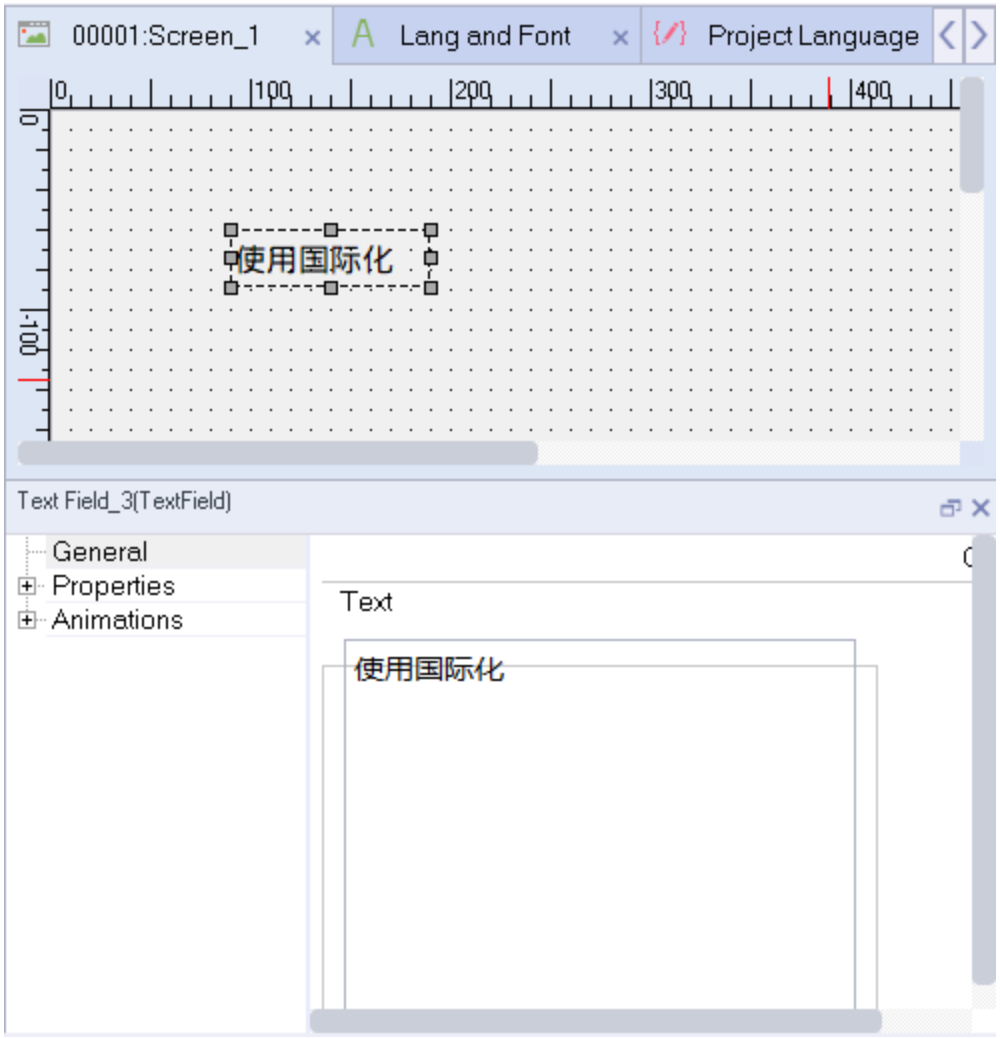
1. The target module should be consistent with the current module selected by the filter, otherwise the import will fail;
2. You cannot import "indexed" fields that do not exist in the project, that is, you must not add or modify "indexed" fields when editing the file externally.

When incorrect data appears in the imported text, a warning message will appear in the output window. The following figure shows the warning when an "indexed" field does not exist:



13.4.4 Internationalization of a Project

In a project, any element that contains translatable text will be automatically added to the internationalization table. The "Referenced by" column shows the unique identification of the element, and the second column shows the text that needs to be translated. For example, in InoTouchPad, you can add a text field in Screen_1 and change the text to "Using internationalization".

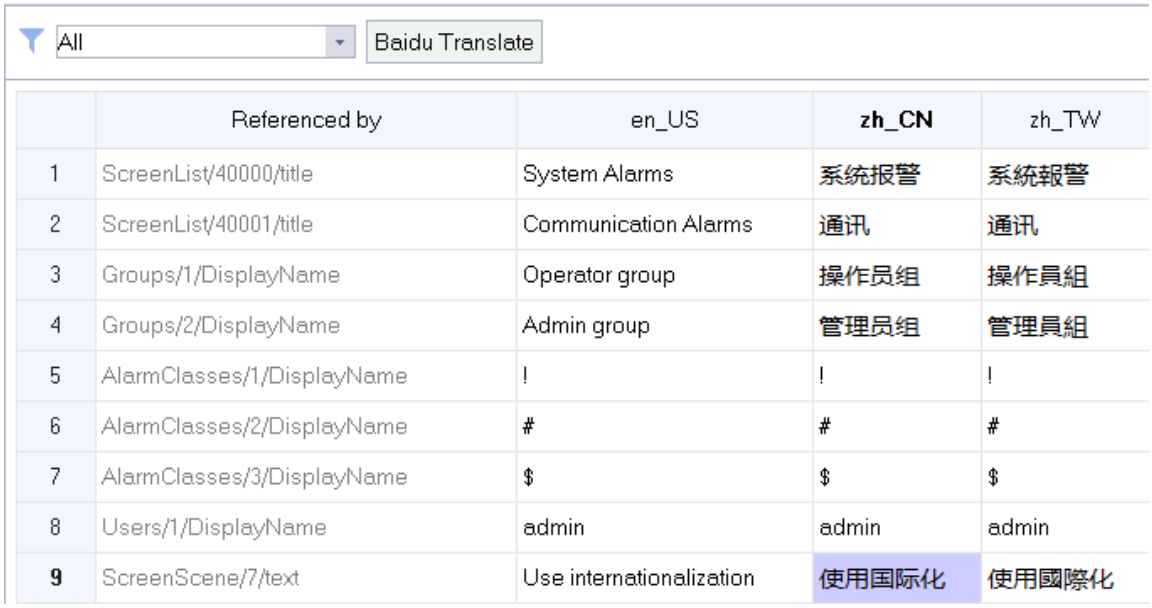


Then a new line for the text field is added at the end of the I18N table.

All | Baidu Translate

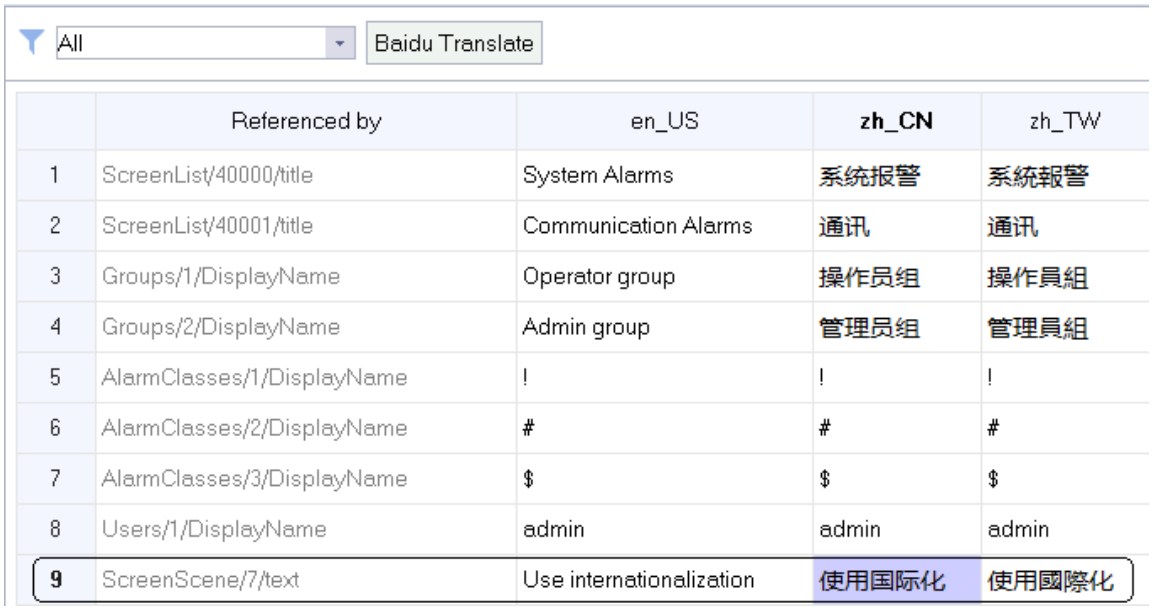
	Referenced by	en_US	zh_CN	zh_TW
1	ScreenList/40000/title	System Alarms	系统报警	系统報警
2	ScreenList/40001/title	Communication Alarms	通讯	通讯
3	Groups/1/DisplayName	Operator group	操作员组	操作員組
4	Groups/2/DisplayName	Admin group	管理员组	管理員組
5	AlarmClasses/1/DisplayName	!	!	!
6	AlarmClasses/2/DisplayName	#	#	#
7	AlarmClasses/3/DisplayName	$	$	$
8	Users/1/DisplayName	admin	admin	admin
9	ScreenScene/7/text	Use internationalization	使用国际化	

Now you can add translation in other languages for the text.



All | Baidu Translate

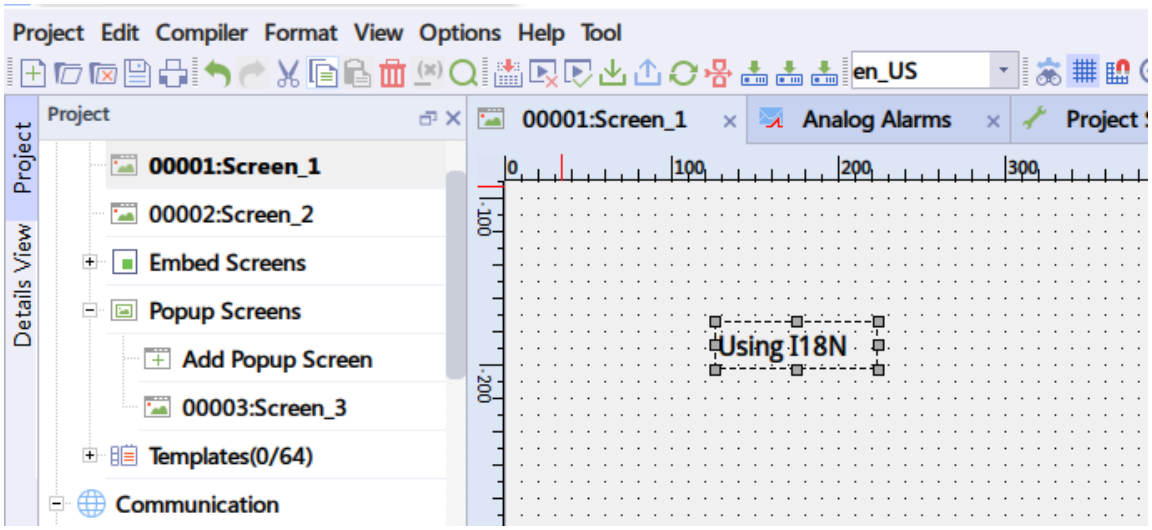
	Referenced by	en_US	zh_CN	zh_TW
1	ScreenList/40000/title	System Alarms	系统报警	系统報警
2	ScreenList/40001/title	Communication Alarms	通讯	通讯
3	Groups/1/DisplayName	Operator group	操作员组	操作員組
4	Groups/2/DisplayName	Admin group	管理员组	管理員組
5	AlarmClasses/1/DisplayName	!	!	!
6	AlarmClasses/2/DisplayName	#	#	#
7	AlarmClasses/3/DisplayName	$	$	$
8	Users/1/DisplayName	admin	admin	admin
9	ScreenScene/7/text	Use internationalization	使用国际化	使用國際化



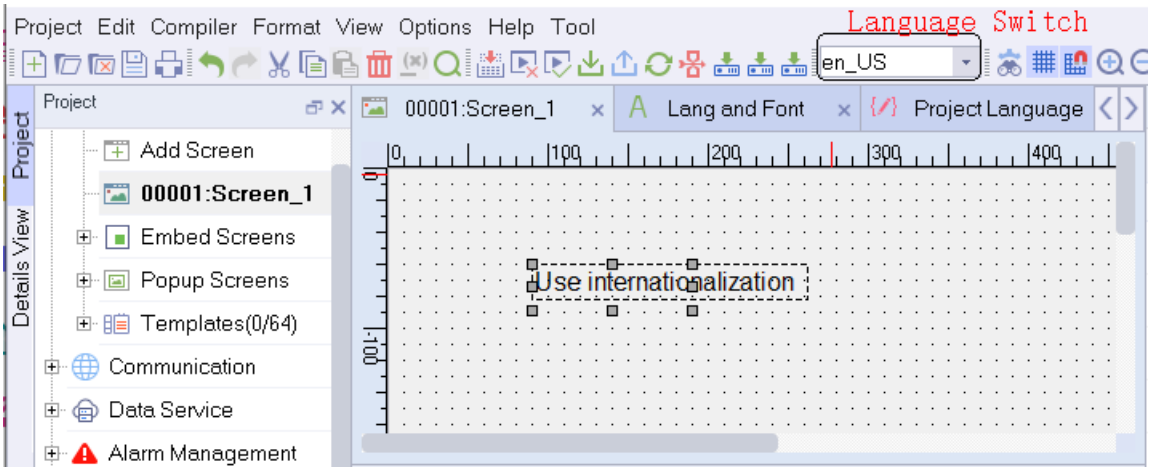
All | Baidu Translate

	Referenced by	en_US	zh_CN	zh_TW
1	ScreenList/40000/title	System Alarms	系统报警	系统報警
2	ScreenList/40001/title	Communication Alarms	通讯	通讯
3	Groups/1/DisplayName	Operator group	操作员组	操作員組
4	Groups/2/DisplayName	Admin group	管理员组	管理員組
5	AlarmClasses/1/DisplayName	!	!	!
6	AlarmClasses/2/DisplayName	#	#	#
7	AlarmClasses/3/DisplayName	$	$	$
8	Users/1/DisplayName	admin	admin	admin
9	ScreenScene/7/text	Use internationalization	使用国际化	使用國際化

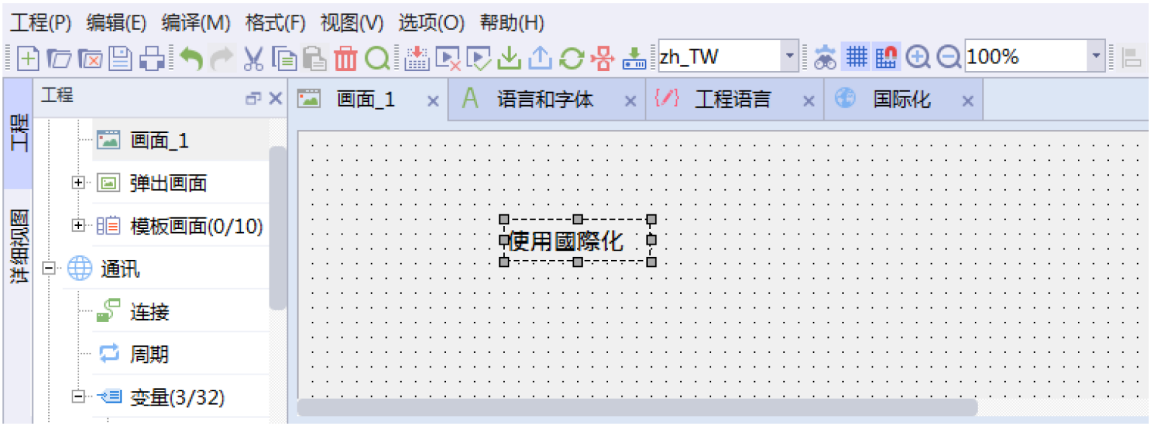
After that, the translation is ready for use. To preview the translation, go back to Screen_1.



On the tool bar, switch the language to en_US, as shown below:



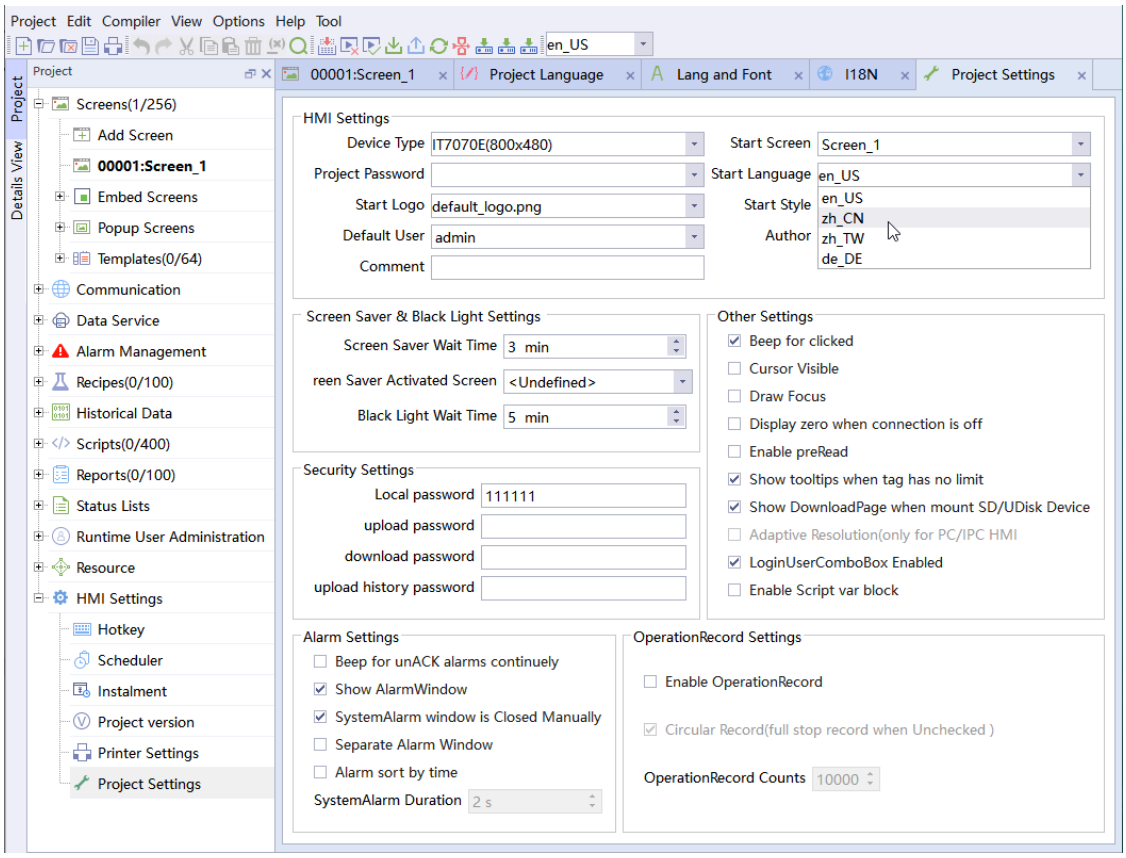
Switch the language to zh_TW, as shown below:



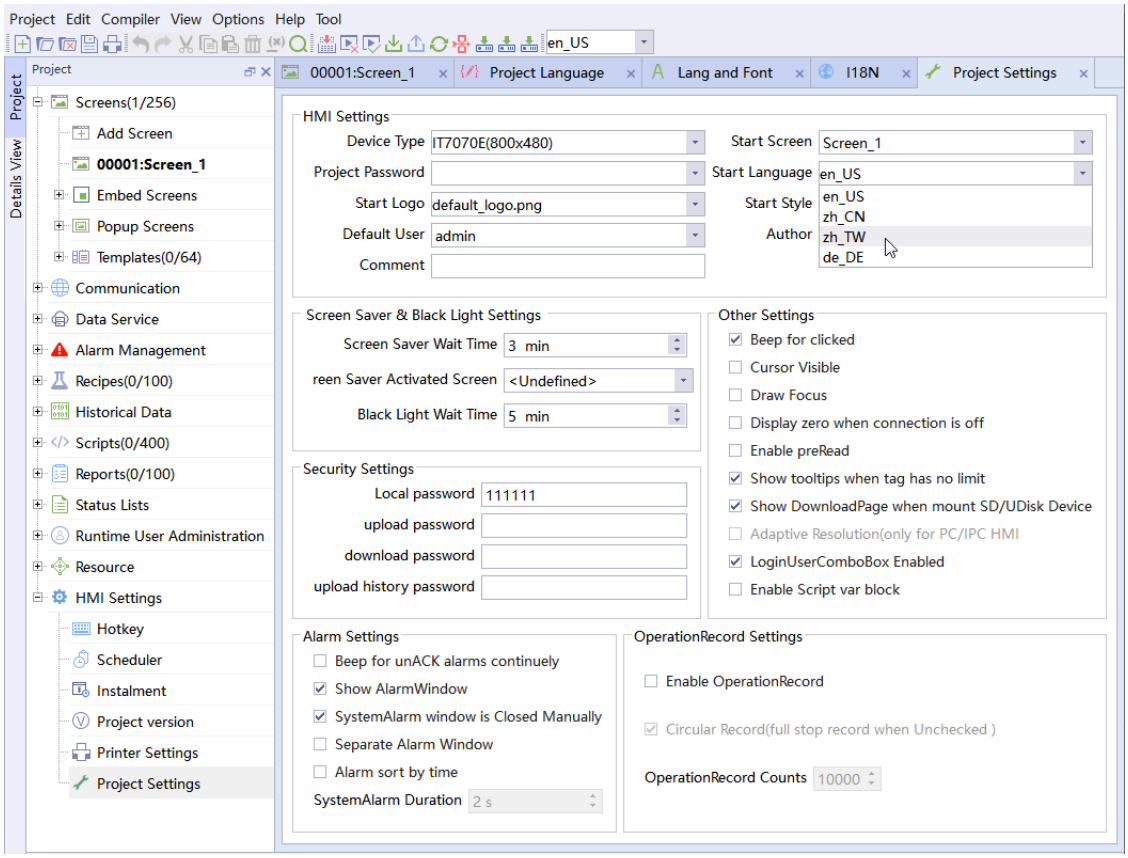
Up to 32 languages can be configured in a project.

After the translation of a project is completed, you can specify a start language for the project. For example:

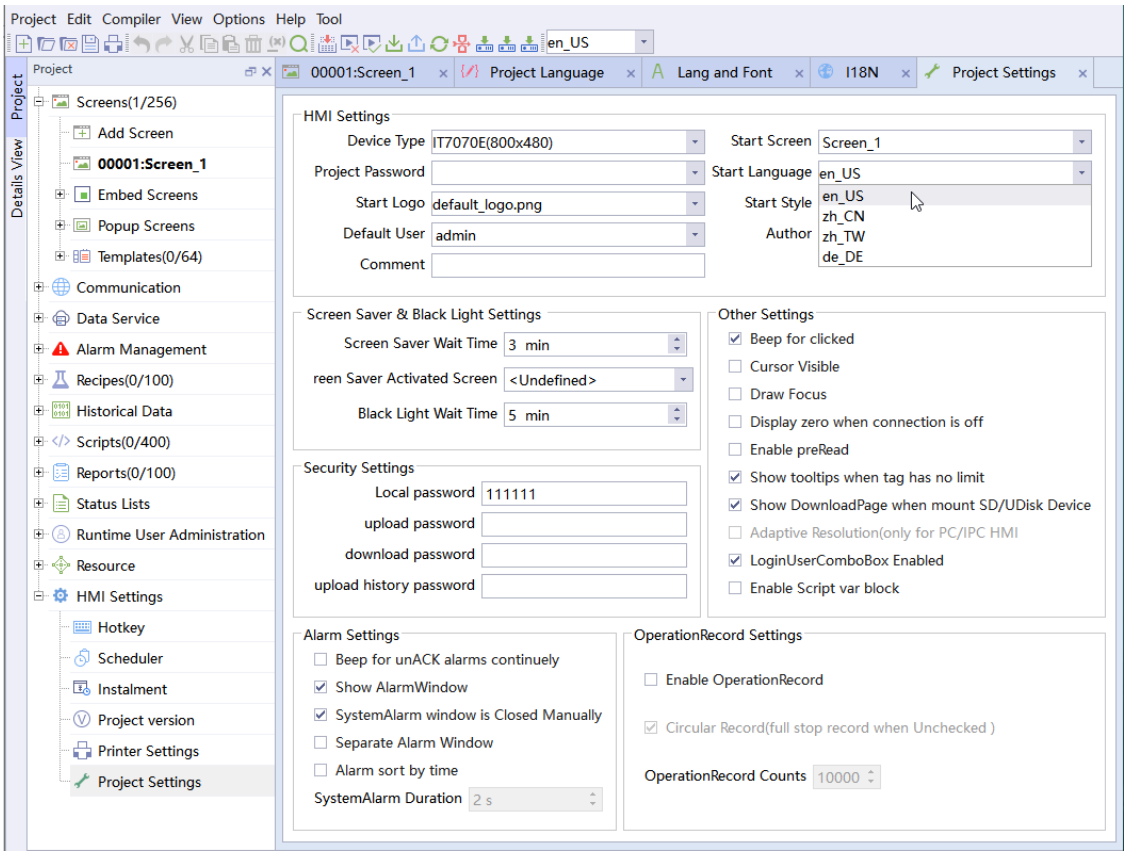
1. If the project is applied in Chinese mainland, set the start language to Simplified Chinese in HMI settings.



2. If the project is applied in Taiwan, set the start language to Traditional Chinese in HMI settings.



3. If the project is applied in the USA, set the start language to English in HMI settings.

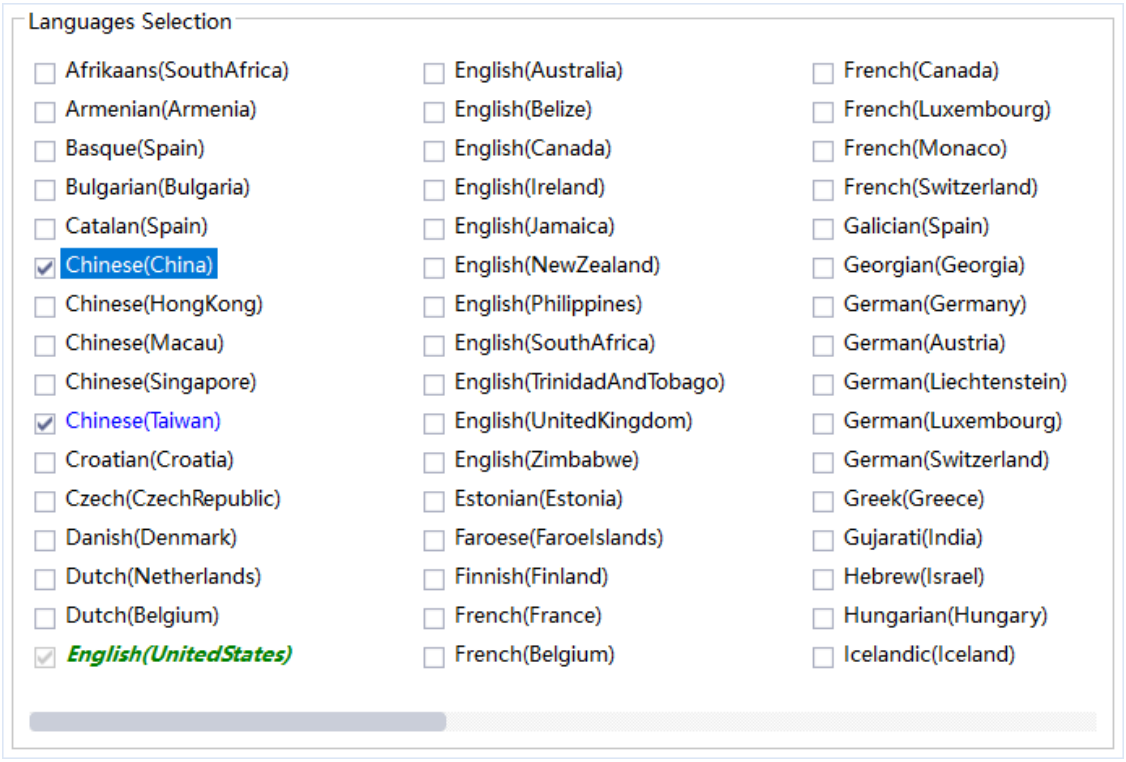


13.4.5 Translation Service

You can translate the project text to any other language through online machine translation.

Steps:

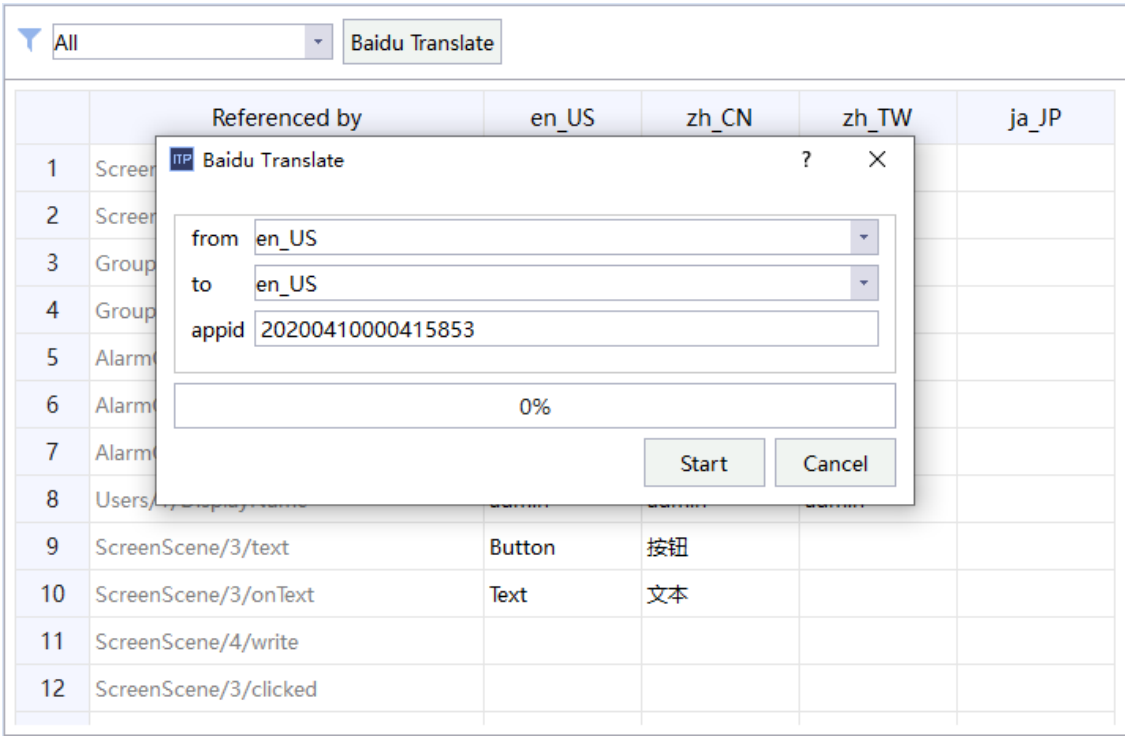
1. Select the target languages you want to use in Resources→Project Language.



2. Select the target languages you want to use in Resources→Language and Font.

	Enabled	Name	DisplayName	Number	Font	Comment
1	☑	English(UnitedStates)	en_US	1	Arial,16px ...	
2	☑	Chinese(China)	zh_CN	2	Droid Sans Fallback,16px ...	
3	☑	Chinese(Taiwan)	zh_TW	3	Droid Sans Fallback,16px ...	
4	☐	Korean(RepublicOfK...	ko_KR	4	Droid Sans Fallback,16px ...	
5	☑	Japanese(Japan)	ja_JP	5	Droid Sans Fallback,16px ...	

3. Click Translate in Resources→Internationalization.



Source language: The language to be translated.

Target language: The language into which the source language to be translated.

Caution

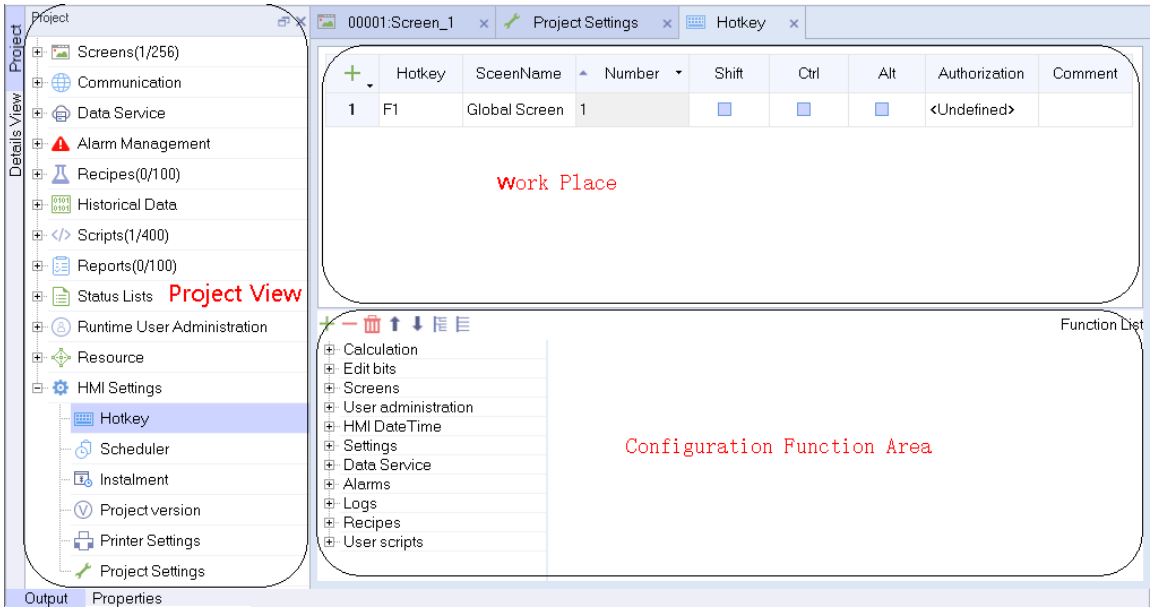
- You need network connection to use Baidu translation service.
- If you have set the target language, this will be skipped during translation.

14 HMI settings

14.1 Hotkeys

14.1.1 Shortcut Keys

The user can use the shortcut keys to configure shortcut key combinations, and bind a system function or script to a shortcut key. If a key is configured for both a global picture and a specific picture, this picture takes precedence over the global picture in the execution of the key. The shortcut key function is only applicable to AP70X series devices, and the shortcut key editing interface is as follows:

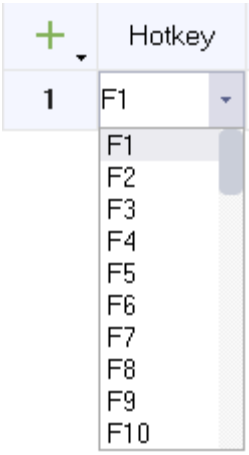


14.1.2 Hotkey Editor

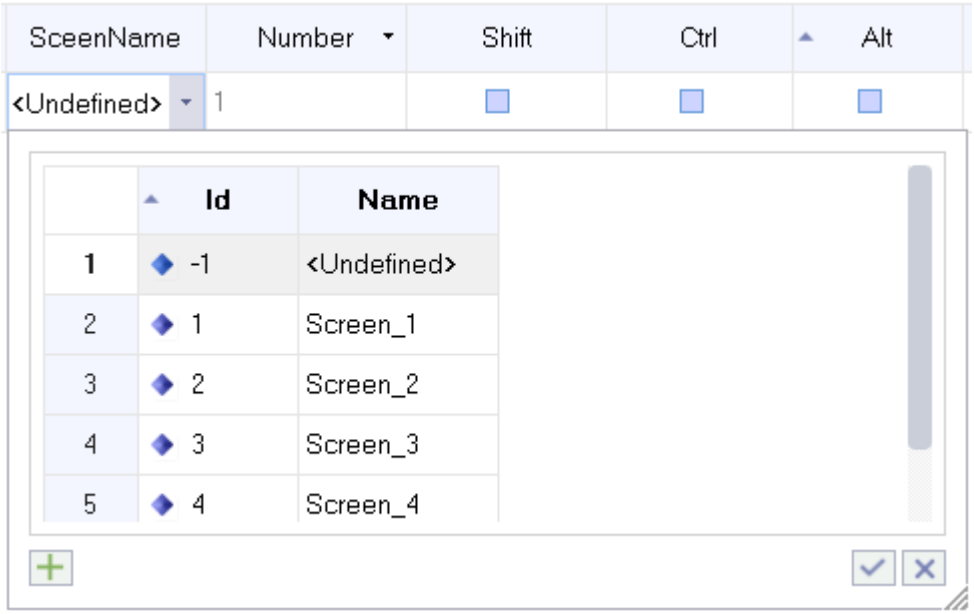
In the hotkey editor, you can add hotkeys and bind functions from the function list to them. A hotkey can be a single key or a key combination. For the use of functions, see *"12.2 Use of System Functions" on page 250*.

The hotkey editor table has eight fields, which are Hotkey, Screen Name, Number, Shift, Ctrl, Alt, Authorization and Comment.

Hotkey includes common keys (except numbers) on a keyboard, as shown in the following figure:

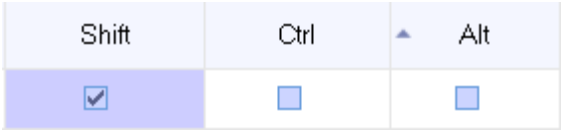


Screen Name includes the global screen and all created screens (except the template) in a project. When you select the global screen, the hotkey is effective in all screens. If you select any other screen, the hotkey can only take effect on that screen.



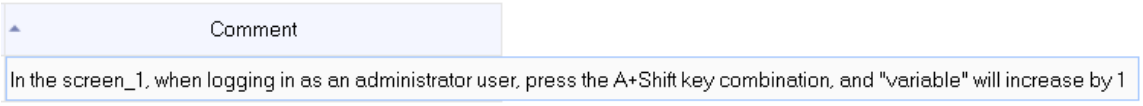
Number is the number of a hotkey.

Shift, Ctrl, and Alt are used to create key combinations.



Authorization assigns the permission to execute the hotkey to a user group.

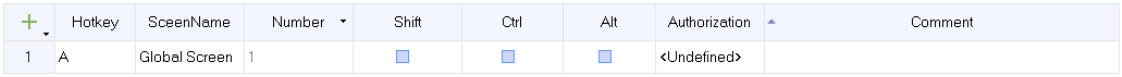
Comment describes the function of a hotkey.



14.1.3 Hotkey

1. In the project view, go to HMI Settings→Hotkey. Double-click Hotkey to open the hotkey editor.
2. Adding a custom hotkey

   Click + to add a hotkey, for example, A, as shown in the following figure.

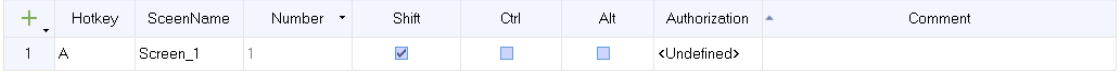


3. Configuring the effective screen

   Click the Screen Name column to show the drop-down symbol. Click the symbol and select the screen in the drop-down list. You can also select Undefined, that is, Global Screen.

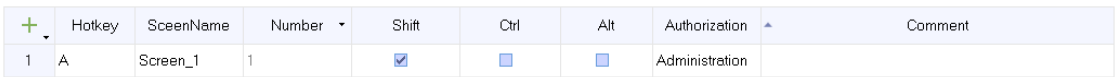
4. Configuring key combination

   If you check Shift, you enable the A + Shift key combination.



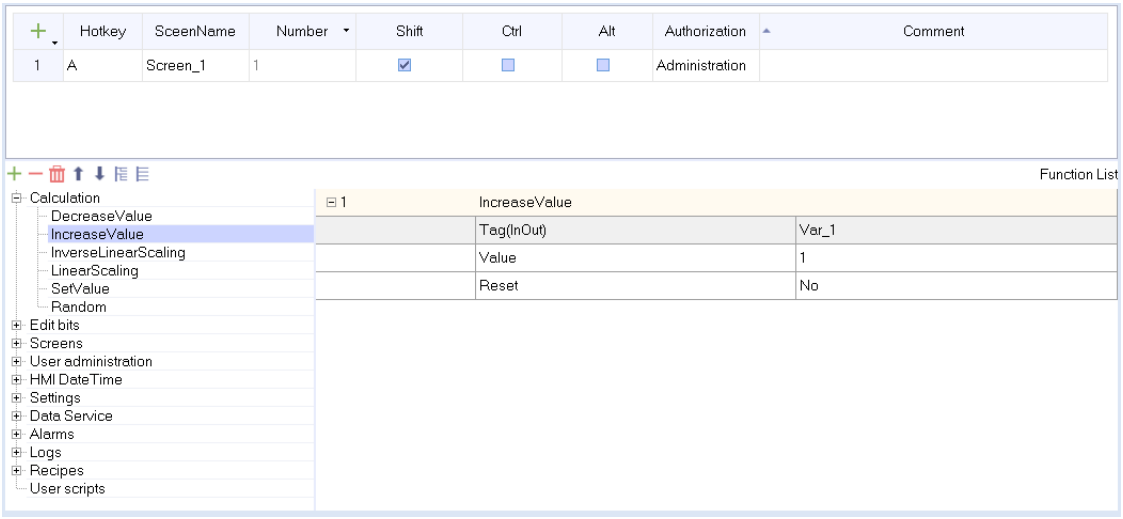
5. Configuring authorization

   Authorization is configured in the Authorization field.



6. Configuring system function

   In the workspace, select the configured hotkey, and bind a system function in the function list.



7. Commenting

   After defining the hotkey and configuring the system function, you should write comments about the function of the hotkey.

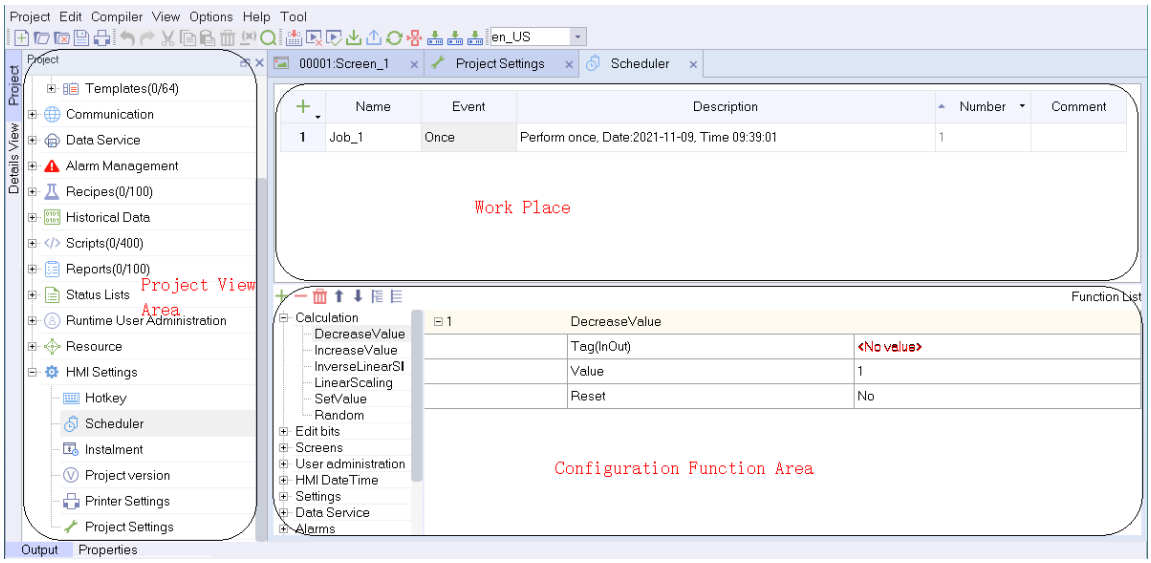
   That is the complete configuration process of a hotkey. In the example above, when the runtime displays screen_1, if you have logged in as an administrator and press A + Shift, the value of tag_1 will be incremented by 1.

14.2 Scheduler

14.2.1 Scheduler Editor

In the scheduler, you can link a system function to a job. When an event occurs, the linked function is called. The scheduler is used to automatically execute jobs controlled by events. For the use of functions, see *"12.2 Use of System Functions" on page 250* for details.

In the scheduler editor, you can schedule jobs by configuring a list of functions for events. You can add jobs in the workspace. The added jobs are globally effective in the runtime.



The workspace displays scheduled jobs, and the job list displays jobs, events, descriptions, numbers, and comments. You can assign labels and comments and select events. Write the scheduler's description of the job, and configure the function or script to be executed in the job in the configuration function area. In the workspace, you can configure multiple jobs. During operation, the system schedules these jobs according to job event conditions.

14.2.2 Event

When a scheduler event is triggered, the associated system function is executed immediately. The following table shows all scheduler events and their descriptions:

Type	Trigger	Description	Restrictions
Time	Once	Triggered once when the system time reaches a certain time set by the user. When no timer is selected, the event is triggered when the time set by the user is reached. When there is a timer, the event is triggered when the system time reaches the time set in the timer.	-
	Every X seconds	Triggered every 0.1–60 seconds	-
	Every minute	Triggered every minute	-
	Every hour	Triggered once at the nth minute in every hour. The value of n ranges from 0 to 59. For example, if the time is set to 10, the event is triggered once in the 10th minute of every hour.	-
	Every day	Triggered once at a specific time (HH:mm) in every day. For example, if you set the time to 10:22, the event is triggered at 10:22 every day. If you have set a timer, the event is triggered when the time set by the timer is reached.	-
	Every week	Triggered once at a specific time (dddd HH: mm) in every day. For example,, if you set the time to Tuesday 10:22, the event is triggered at 10:22 on Tuesday every week.	-
	Every month	Triggered once at a specific time (d HH:mm) in every day. For example, if you set the time to 2 10:22, the event is triggered at 10:22 on the second day of the month.	-
	Every year	Triggered once at a specific time (MM-dd HH: mm) in every day. For example, if you set the time to 02-02 10:22, the event is triggered at 10:22 on February 2 every year. If you have set a timer, the event is triggered when the time set by the timer is reached.	-
Event	Screen switching	Triggered once at screen switching	This event can only be configured once for a project
	User switching	Triggered once when the user logs out.	This event can only be configured once for a project
	Login	Triggered once when the user logs in.	This event can only be configured once for a project
	Alarm buffer overflow	Triggered once when the alarm buffer overflows. The alarm buffer is limited to a specified size (512) and is not involved in configuration.	This event can only be configured once for a project
	Start	Triggered once when the runtime starts.	This event can only be configured once for a project
	Screensaver	Triggered once after screensaver is activated	This event can only be configured once for a project

14.2.3 Job

1. In the project view, expand HMI Settings and double-click Scheduler to open the editor.
2. Adding or deleting jobs

Click + to add a job. Select the job you want to delete, and click — to delete it.

	Name	Event	Description	Number	Comment
1	Job_1	Once	Perform once, Date:2021-11-09, Time 09:39:01	1	
2	Job_2	Once	Perform once, Date:2021-11-09, Time 09:46:03	2	

- Configuring the event

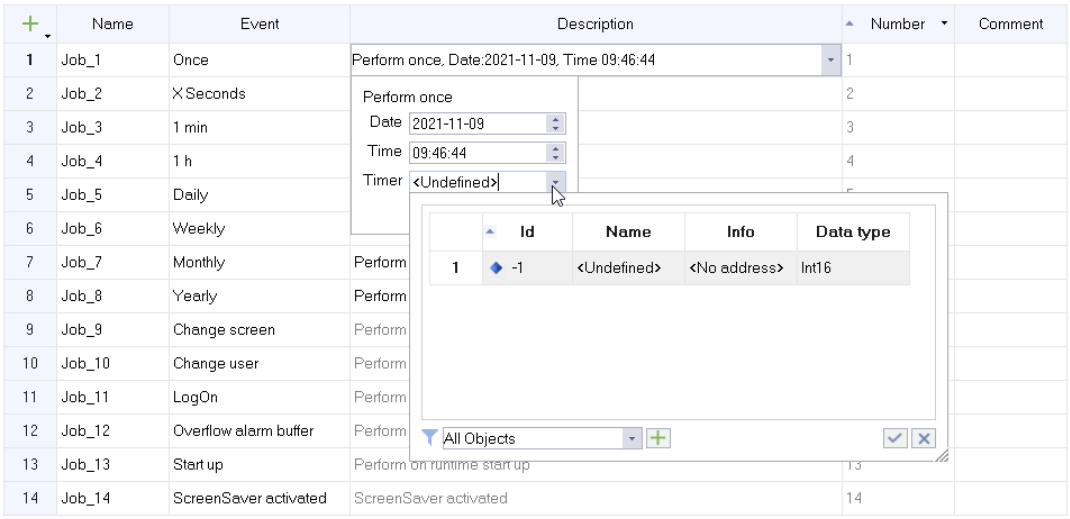
  Click the Event column and click to show the the drop-down list. You can select one as needed from the 14 events.

- Configuring job description

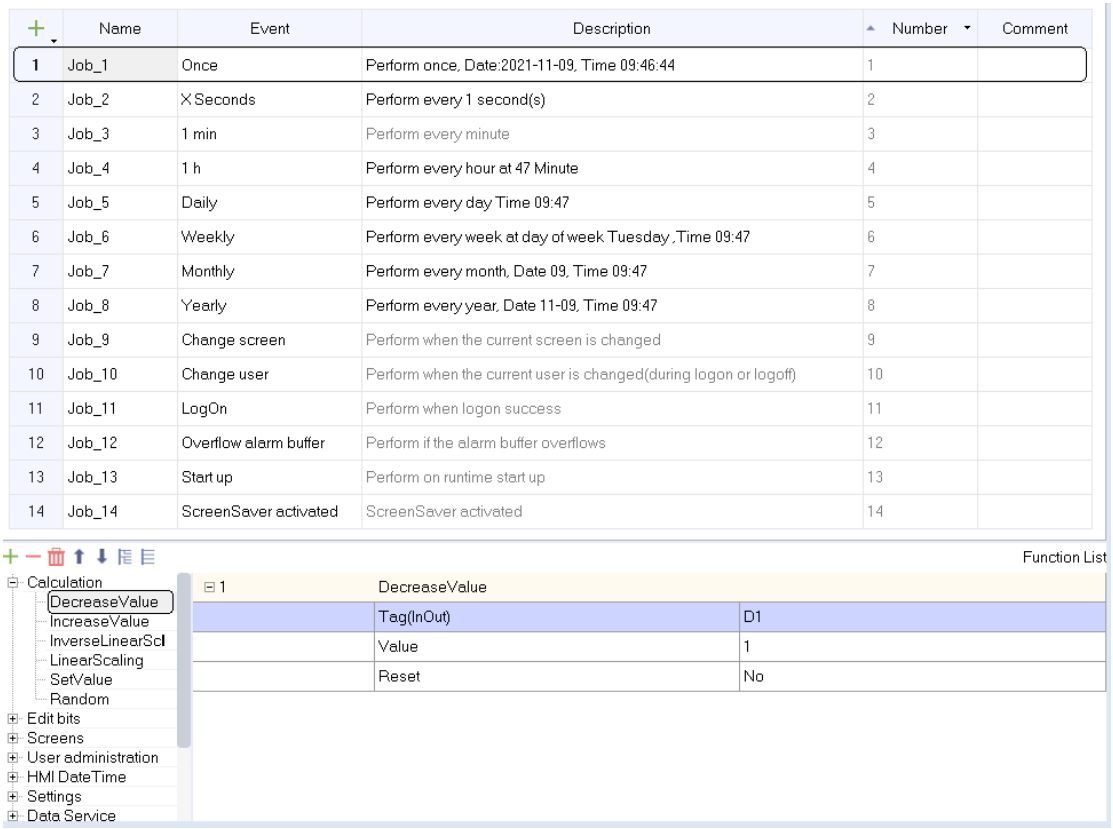
  When creating a job, you need to configure the event description.

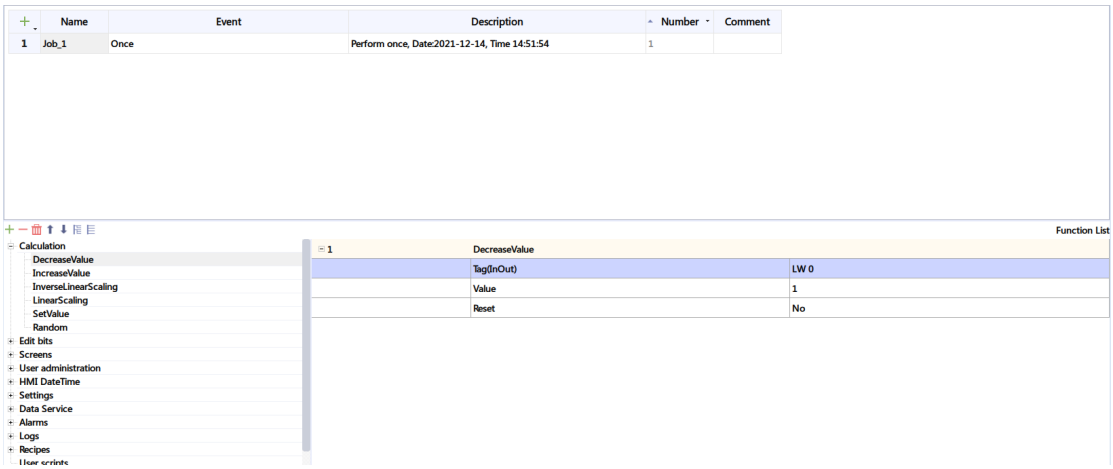
	Name	Event	Description	Number	Comment
1	Job_1	Once	Perform once, Date:2021-11-09, Time 09:46:44	1	
2	Job_2	X Seconds	Perform every 1 second(s)	2	
3	Job_3	1 min	Perform every minute	3	
4	Job_4	1 h	Perform every hour at 47 Minute	4	
5	Job_5	Daily	Perform every day Time 09:47	5	
6	Job_6	Weekly	Perform every week at day of week Tuesday ,Time 09:47	6	
7	Job_7	Monthly	Perform every month, Date 09, Time 09:47	7	
8	Job_8	Yearly	Perform every year, Date 11-09, Time 09:47	8	
9	Job_9	Change screen	Perform when the current screen is changed	9	
10	Job_10	Change user	Perform when the current user is changed(during logon or logoff)	10	
11	Job_11	LogOn	Perform when logon success	11	
12	Job_12	Overflow alarm buffer	Perform if the alarm buffer overflows	12	
13	Job_13	Start up	Perform on runtime start up	13	
14	Job_14	ScreenSaver activated	ScreenSaver activated	14	

  - When the event is "1 min", "change screen", "change user", "overflow alarm buffer", "start up" or "screensaver activated", the default description is used.
  - To dynamically modify the configured startup time of a daily, annual, or one-time event at run time, select an internal tag as the timer. The tag value determines the start time of the job at run time. The tag must be of the Date Time type.
  - For system events, only one job can be configured and executed on each HMI device.



- Comment
  The comment is a description of the job.
- Configuring system function
  When an event occurs, the linked function is called.





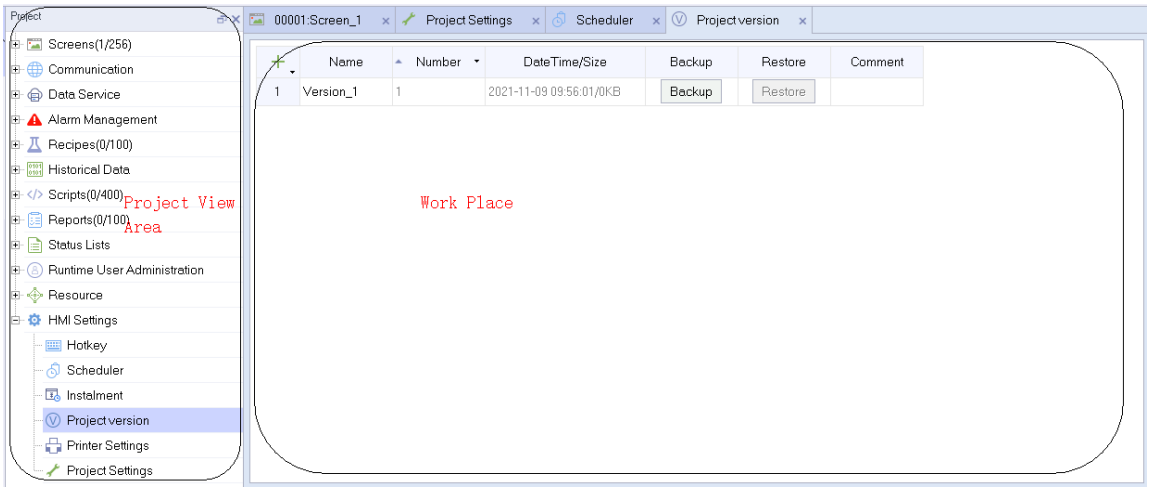
Select a job, select a function in the function list, and configure the tag and value. When an event occurs, the linked function is called.

14.3 Project Version

14.3.1 Project Version Editor

The project version is mainly used for the backup and recovery of a project. Up to 16 projects can be backed up.

The project version editor is as follows. The editor table has six fields, namely name, number, date and time, backup, restore and comment:



Name: the version of the project backup.

Number: the number of the project backup.

Date and time: the time when the project was backed up.

Backup: Click the Backup button to back up the project.

Restore: Click the Restore button to restore the project backup.

Comment: description of the project backup.

14.3.2 Using Project Version Management

During project configuration, when some independent functions are configured, and the modification is huge, it is often necessary to back up the project. If later configuration is inappropriate or need to be re-edited, you can go back to a backed up version.

1. Expand HMI Settings and double-click Project version to open the editor.
2. Add a backup/restore entry.

   Click + .

+	Name	Number	DateTime/Size	Backup	Restore	Comment
1	Version_1	1	2021-11-09 09:56:01/0KB	Backup	Restore	

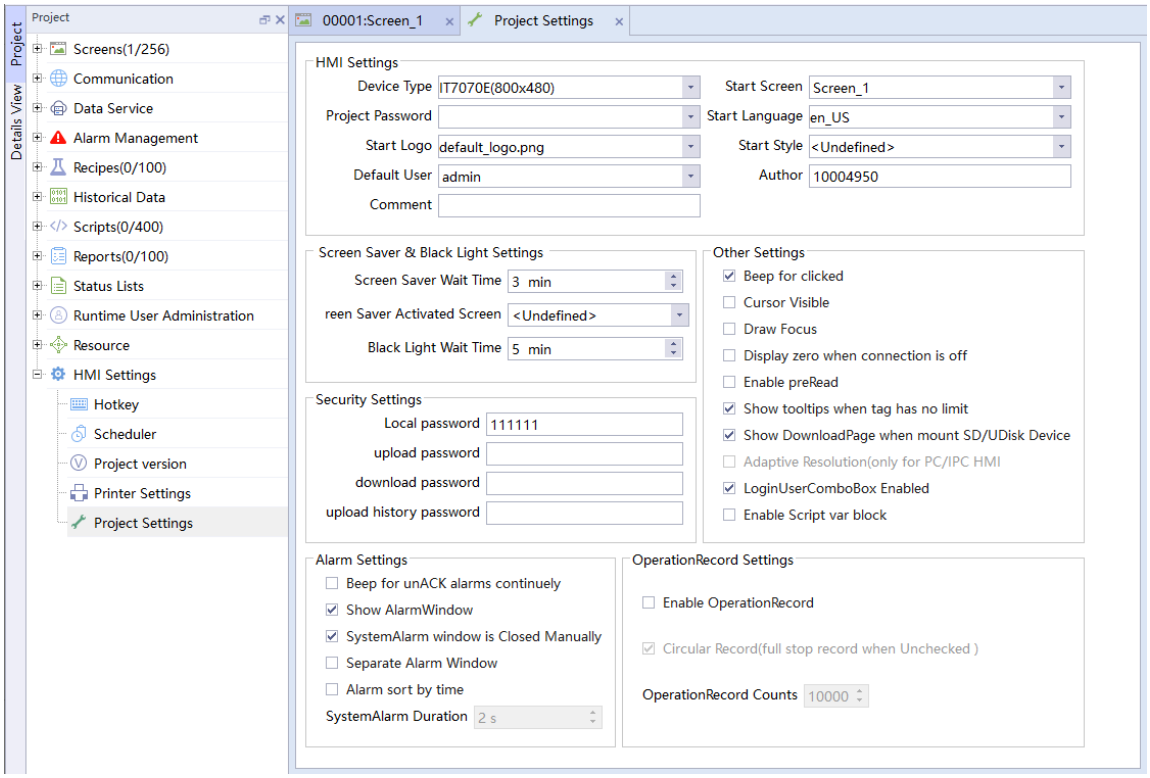
3. Click the Backup button in the Backup button to back up the current project, and you will find that the Restore button is enabled.

+	Name	Number	DateTime/Size	Backup	Restore	Comment
1	Version_1	1	2021-11-09 09:59:49/18KB	Backup	Restore	

4. After backup, in the subsequent configuration, if you need to re-edit a configured function module, just find the backup/restore entry created last time in the project version workspace, and click the Restore button in this entry.

14.4 Project Settings

Project settings include HMI settings, screen saver and backlight settings, alarm settings, operation record settings, other settings and security settings. The setting page is as follows:



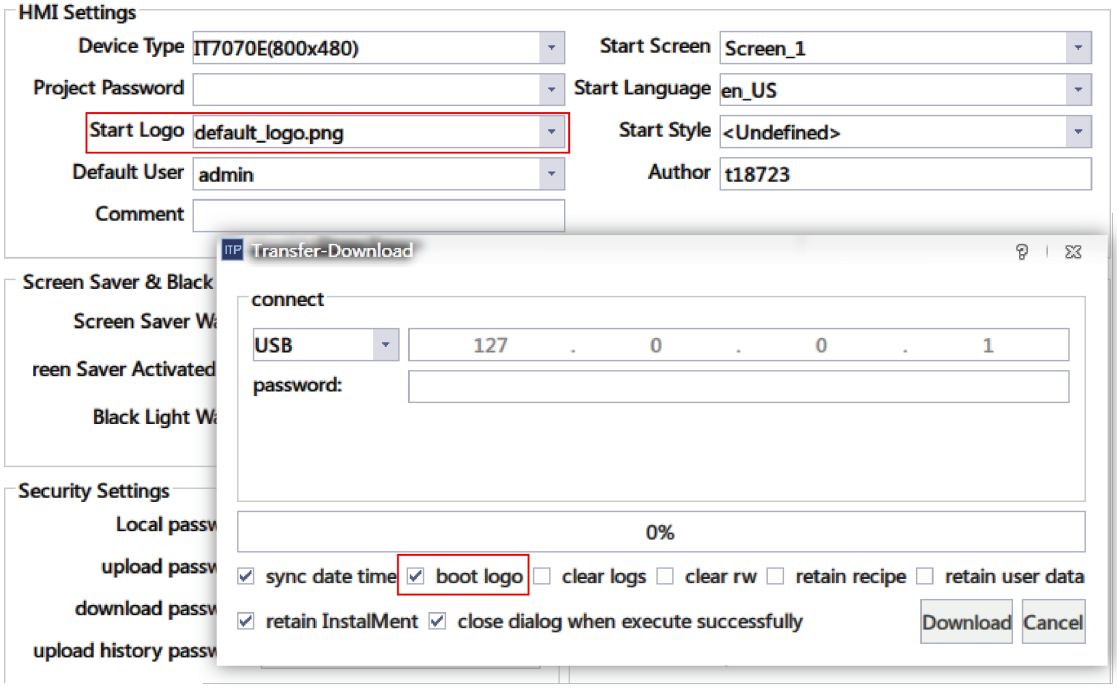
HMI settings

Device Type: You can choose the device type used in the project here.

Project Password: To improve the security of the project, you can encrypt the project. If you set the password, you must input the password before editing the project.

Start Logo: You can choose the boot image to be displayed when the device is started. Note: You must also check "boot logo" in the Download window when you download the project.

Default User: Default user on the device.

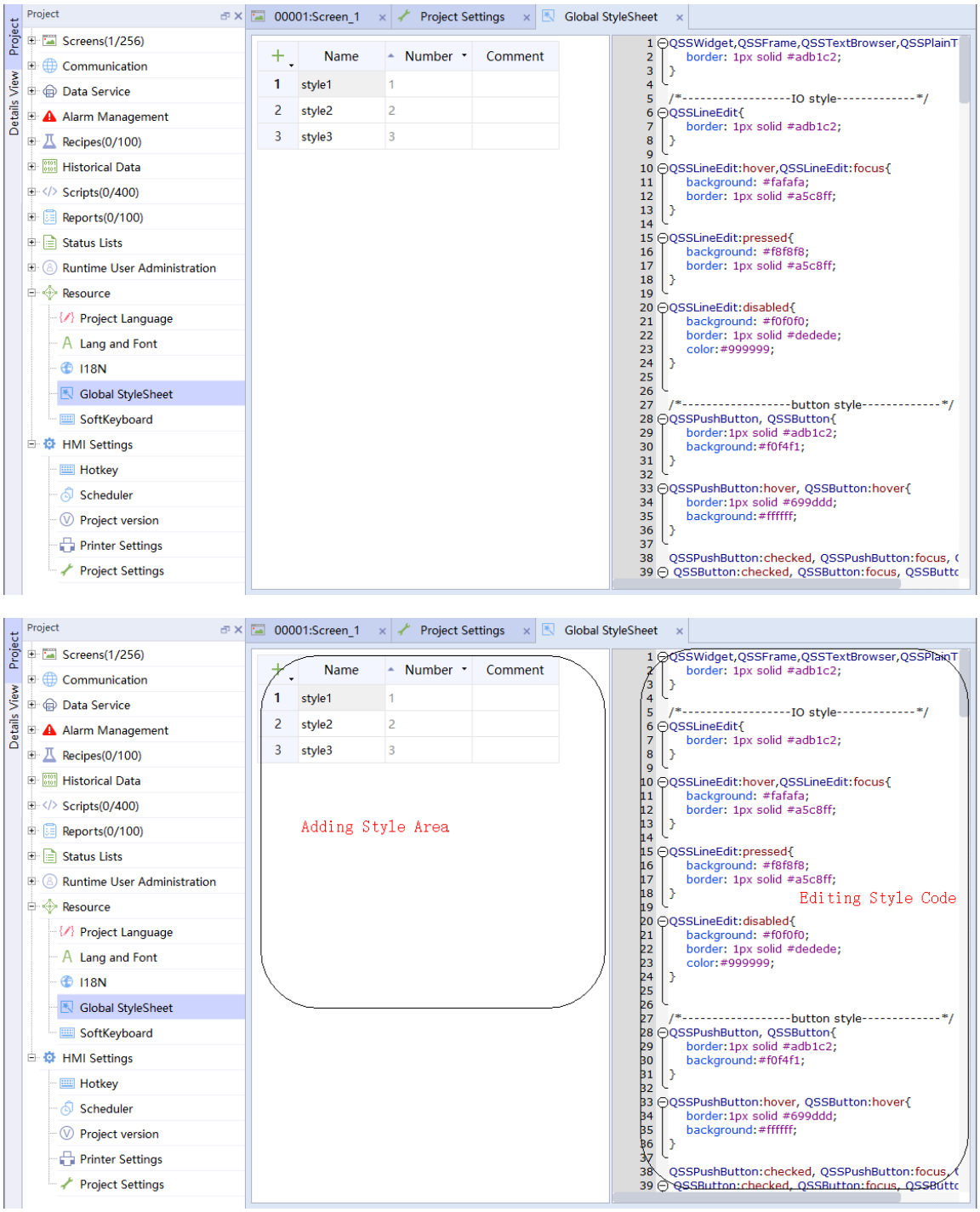


Author: The project creator.

Start Screen: The initial screen displayed after the runtime starts.

Start Language: The default language of the runtime.

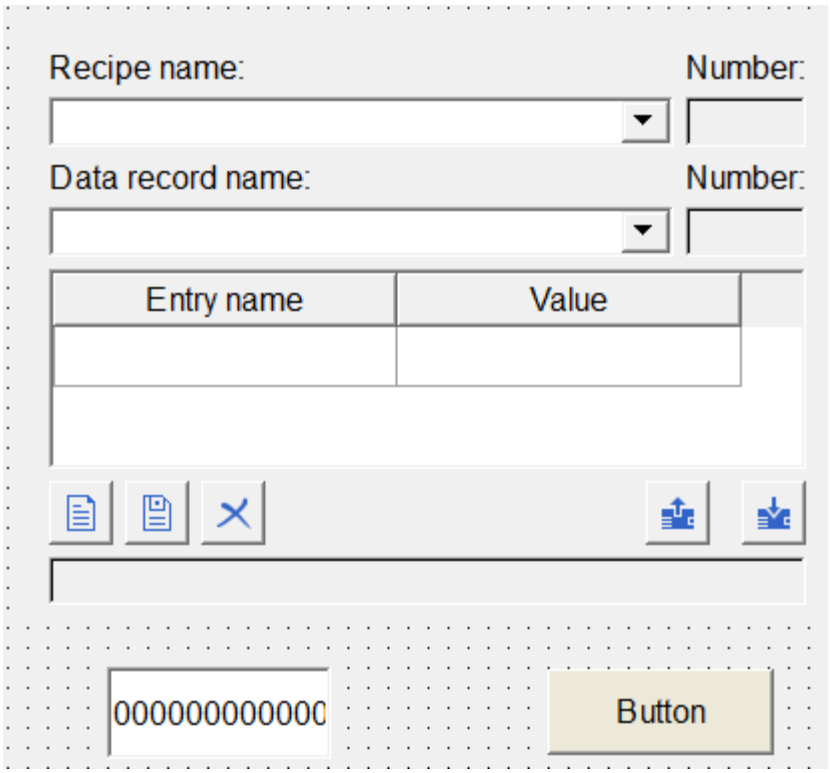
Start Style: The global style initially applied after the runtime is started. InoTouchPad provides three global styles, which you can edit in Resource→Global StyleSheet, as shown in the following figure:



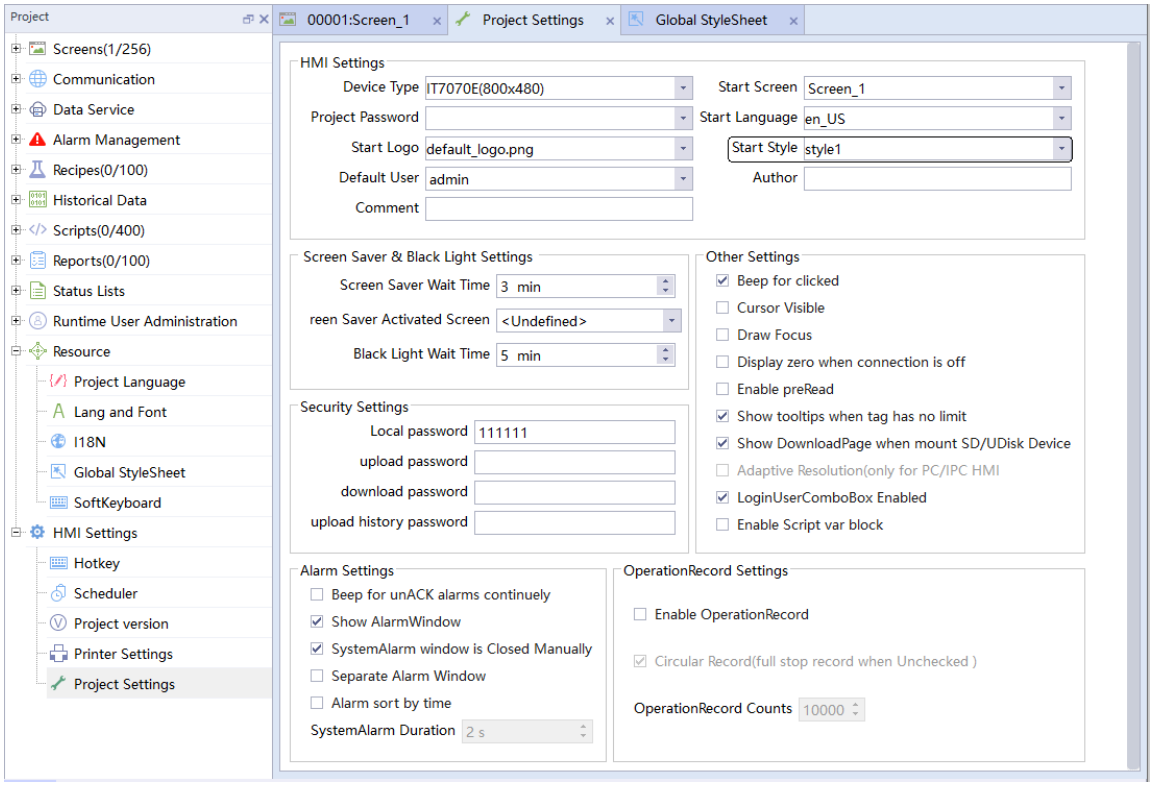
The global styles apply to IO fields, buttons, text, progress bars, drop-down list controls, scroll bars, tables, and table headers.

You can add and edit new styles in the page and select one of them as the start style.

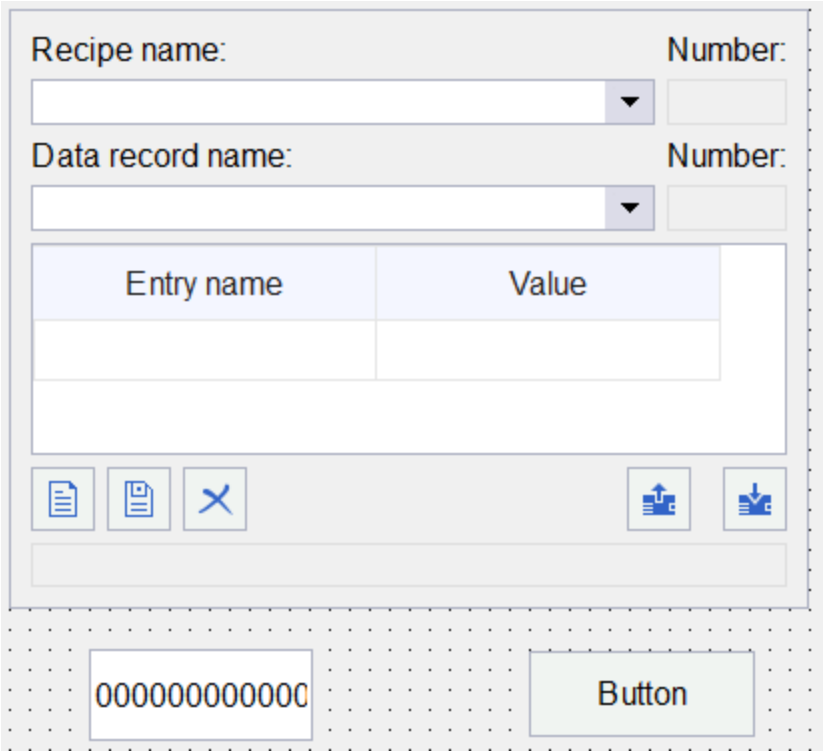
For example, add a recipe view, an IO field and a button to the screen. The following figure shows the appearance of the screen before the global style is applied:



Then, select the global style style1:



Style1 is applied to the objects on the screen:



Comment: Description of the project.

Screen saver & backlight settings

Screen Saver Wait Time: The time before the screen saver is activated when there is no operation. The default is 3 minutes. Set it to 0 to disable the screen saver.

Screen Save Activated Screen: The screen to be switched to when the screen saver is activated.

Backlight Wait Time: The time before the backlight is off when there is no operation. The default is 5 minutes. Set it to 0 to never turn off the backlight.

Caution

The wait time for the screensaver and the wait time for turning off the backlight are independent of each other, and both start when no operation is detected on the system. If the wait time for turning off the backlight is less than that for the screensaver, the backlight will be turned off first, so you cannot see the screensaver.

Alarm settings

Beep for unACK alarms continuously: If this is checked, when there is an alarm, the buzzer will continue to sound to remind the user that there is a fault. By default, the option is not selected.

Show Alarm Window: When this is checked, the system alarm window will pop up when there is a system alarm. By default, the option is selected.

Close System Alarm Window Manually: This is available when Show Alarm Window is checked. If this is checked, the operator needs to manually close the system alarm window. If not, the system alarm window will be automatically closed according to the system alarm duration. By default, the option is selected.

System Alarm Duration: This is available when Close System Alarm Window Manually is not checked. The system alarm window is automatically closed when this time is reached. The default is 2 seconds.

Separate Alarm Window: If this is checked, a separate communication alarm window will pop up if there is a communication alarm.

Alarm sort by time: when this is checked, alarms are sorted by time.

Operation record settings

Enable Operation Record: When this is checked, the runtime will record the operations of the operator.

Circular Record: If this is checked, when the number of records reaches the operation record count, earlier records will be erased. If this is not checked, the recording will stop after the number of records reaches the operation record count.

Operation Record Count: This is available when Circular Record is checked. It is the total number of operation records.

Other settings

Beep for Clicking: If this is checked, the buzzer will sound when the operator operates in the runtime. If not, there is no sound.

Cursor Visible: When this is checked, the runtime will display the cursor. If not, the cursor is not displayed.

Draw Focus: If this is checked, dotted border will be drawn on a control when it is focused. If not, no border is drawn.

IT7PC/AP701 Adaptive Resolution: It is only available when the device type is IT7PC or AP70X series. If this is checked, the interface will be adapted to the screen resolution.

Security Settings

Local Password: The password required to access the HMI setting page.

Upload Password: The password required to upload a project.

Download Password: The password required to download a project or firmware.

Upload History Password: The password required to upload historical data.

14.5 Installment Payment

14.5.1 Basic information

For customers who want to try our products, we offer an installment payment solution. This solution also benefits HMI manufacturers when they want to extend the trial period, maintain multiple clients in a project, or customize the screen displayed after unlocking.
Major features:

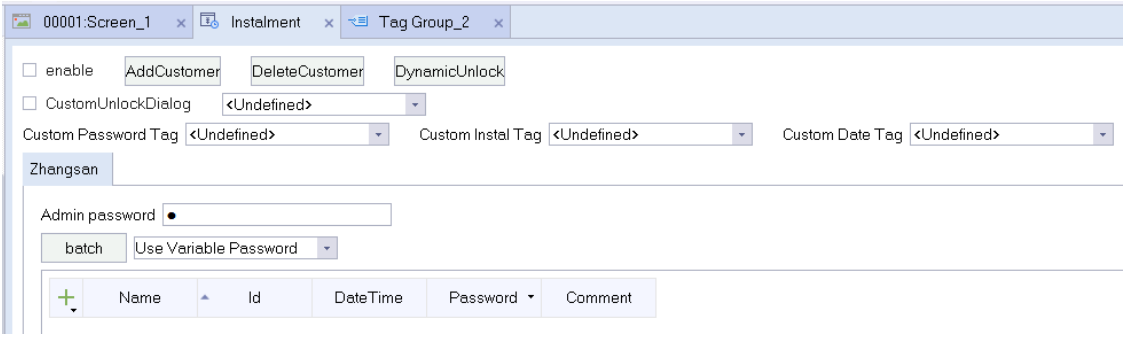
1. Up to 10 customers can be created in a project.
2. Three installment payment methods:

   a. Static password
   b. Variable password
   c. Dynamic password

3. Customizable pop-up screen after unlocking.
4. Option to keep the original installment information during downloading.
5. Trigger the next installment in advance.
6. Display remaining time before next installment.
7. Unlocking with the administrator password.

14.5.2 Overview

1. The installment page is as follows:

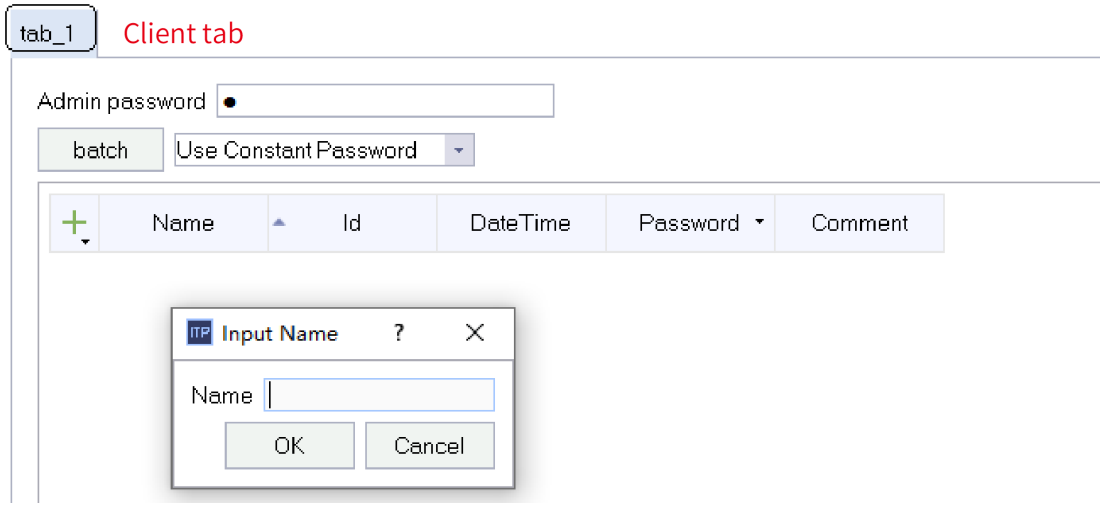


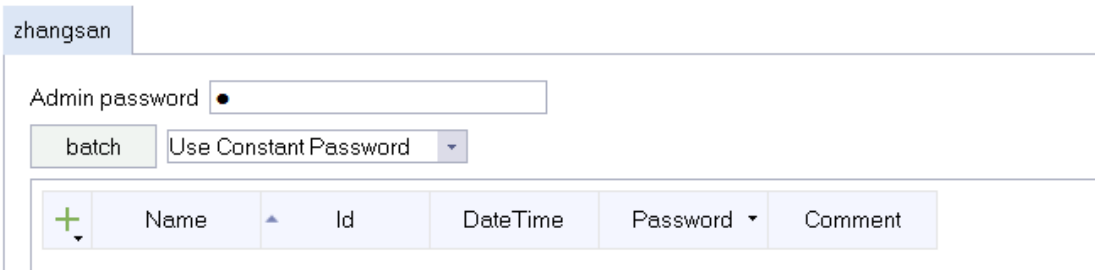
2. Using constant password

   In the constant password mode, installment time and password are fixed, and the result depends on configuration design.

   a. You can change the customer name

      As shown in the following figure, double-click the customer name tab to pop up the customer name input window and enter "Tom" to modify the name:

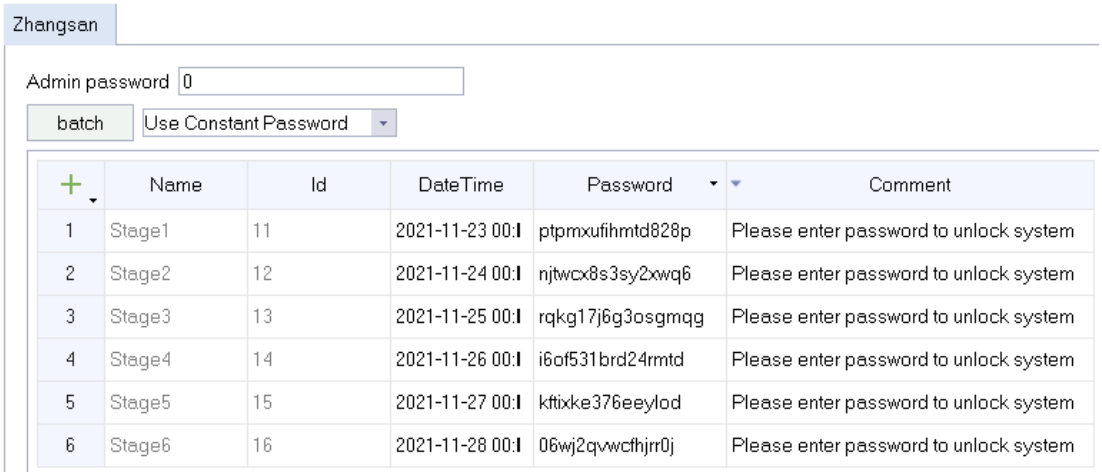




b. Using constant password

Select "Use Constant Password".

c. Creating an installment

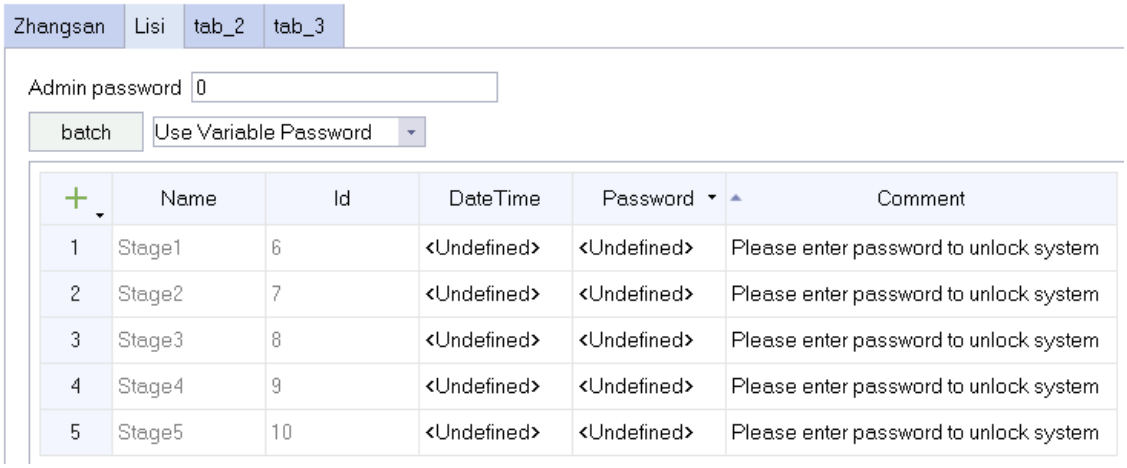


As shown in the above figure, you can create new installment through the "Batch" button or the "+" button in the upper left corner of the table. Among them, the administrator password of the customer can be modified, and the "time" and "password" of a certain period can be selected for modification.

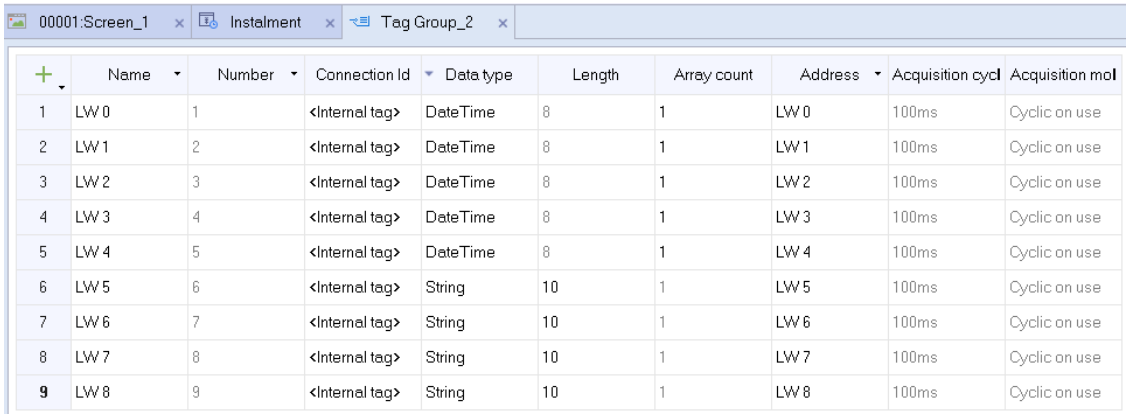
d. Enabled

After the new installment is completed, you can check "Enable" in the upper left corner of the page, and download the project to the HMI.

3. Using variable password

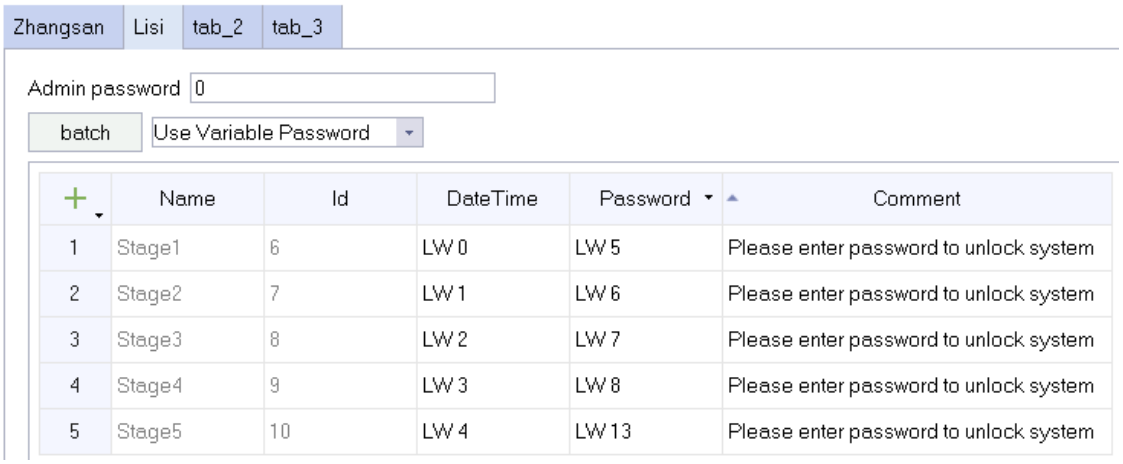


As shown in the above figure, add a new customer "Alfred", select "Use Variable Password", and create new installments. According to the newly created 5 periods, create 5 tags of DateTime and String types in the tag group, and assign an initial value to them, as shown in the following figure.



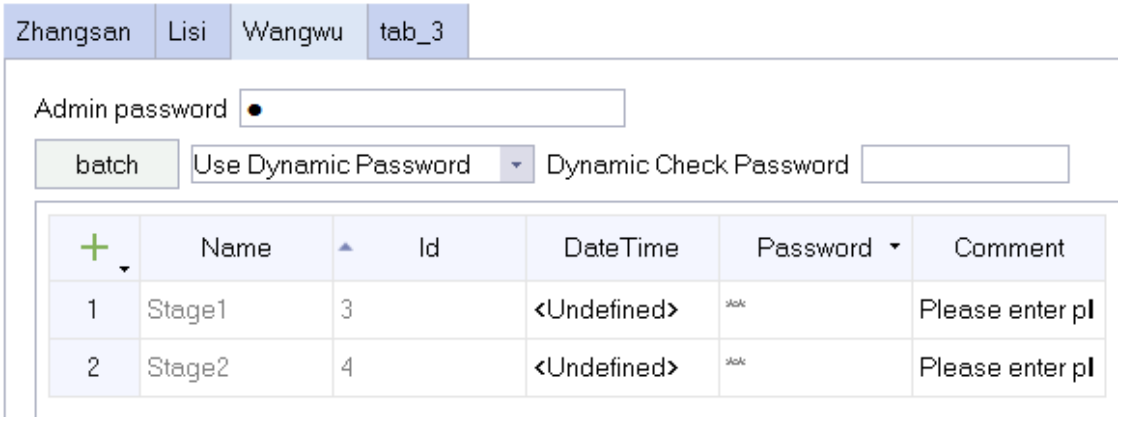
	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycl	Acquisition mol
1	LW 0	1	<Internal tag>	DateTime	8	1	LW 0	100ms	Cyclic on use
2	LW 1	2	<Internal tag>	DateTime	8	1	LW 1	100ms	Cyclic on use
3	LW 2	3	<Internal tag>	DateTime	8	1	LW 2	100ms	Cyclic on use
4	LW 3	4	<Internal tag>	DateTime	8	1	LW 3	100ms	Cyclic on use
5	LW 4	5	<Internal tag>	DateTime	8	1	LW 4	100ms	Cyclic on use
6	LW 5	6	<Internal tag>	String	10	1	LW 5	100ms	Cyclic on use
7	LW 6	7	<Internal tag>	String	10	1	LW 6	100ms	Cyclic on use
8	LW 7	8	<Internal tag>	String	10	1	LW 7	100ms	Cyclic on use
9	LW 8	9	<Internal tag>	String	10	1	LW 8	100ms	Cyclic on use

After that, bind the time and password, and the administrator password can be modified, as shown in the following figure. When the HMI is running, you can change the value of a bound tag. Note that the unlocked stage can be adjusted by modifying the time tag, but after all the stages are unlocked, modifying the time will not trigger installment again.



	Name	Id	DateTime	Password	Comment
1	Stage1	6	LW 0	LW 5	Please enter password to unlock system
2	Stage2	7	LW 1	LW 6	Please enter password to unlock system
3	Stage3	8	LW 2	LW 7	Please enter password to unlock system
4	Stage4	9	LW 3	LW 8	Please enter password to unlock system
5	Stage5	10	LW 4	LW 13	Please enter password to unlock system

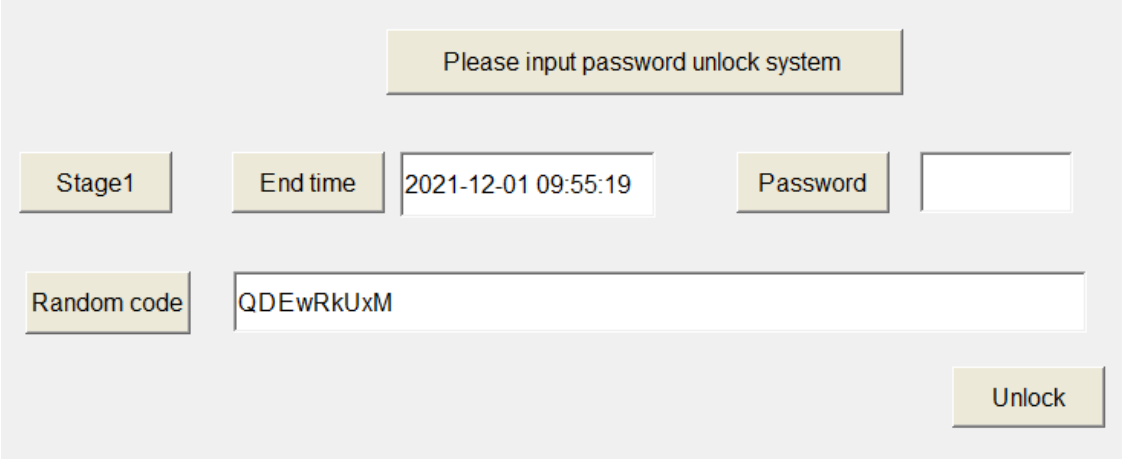
4. Using dynamic password



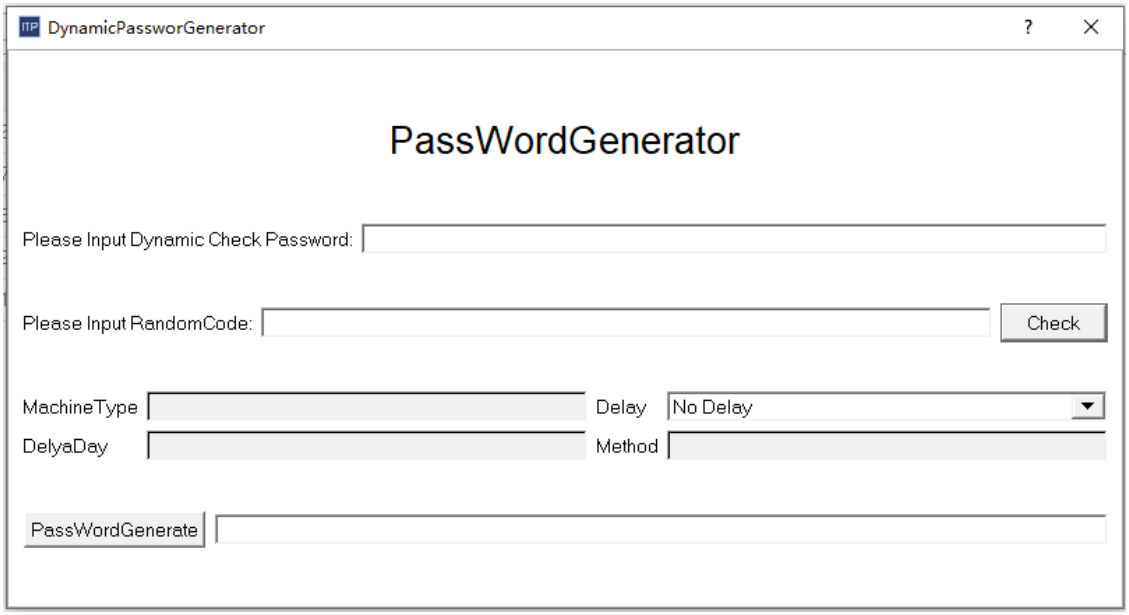
	Name	Id	DateTime	Password	Comment
1	Stage1	3	<Undefined>		Please enter pl
2	Stage2	4	<Undefined>		Please enter pl

As shown in the above figure, dynamic password is similar to variable password, but you only need to configure time tags, and the installment expiration page is shown in the following figure. The

dynamic check password is for password protection of the password generator. It can contain two numeric or alphabetic characters.



After the dynamic password expires, a random code will be generated. You can click the "Dynamic Unlock" button in the configuration page to enter the random code and dynamic check password as shown in the following figure. Click the Check button. After check, you need to set the delay. If you select Delay, you need to enter the delay days (1-99 days). After that, click the "Generate Password" button, and the generated dynamic password can be input into the password input box of the unlocking screen, thus achieving online extension.

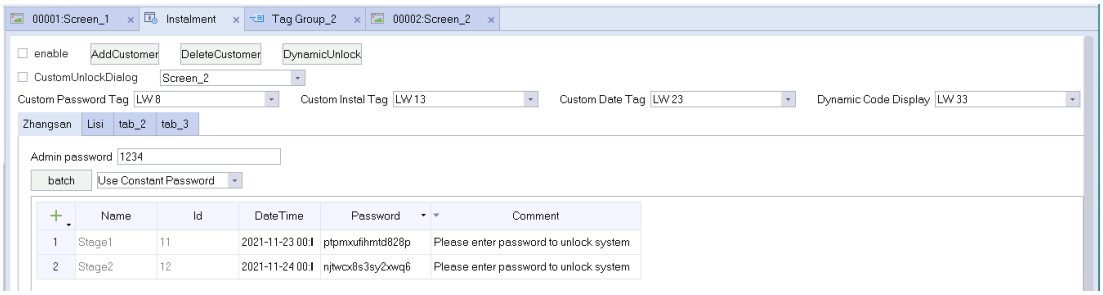


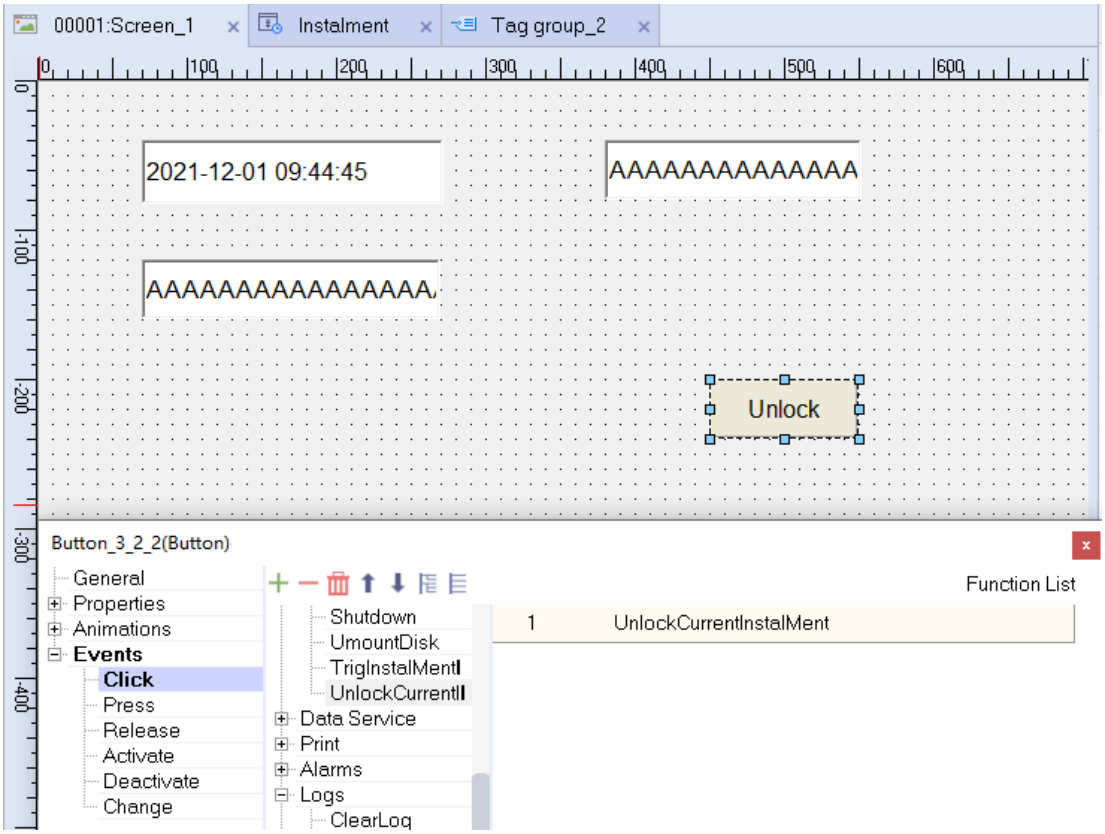
5. Customizing the unlocking screen

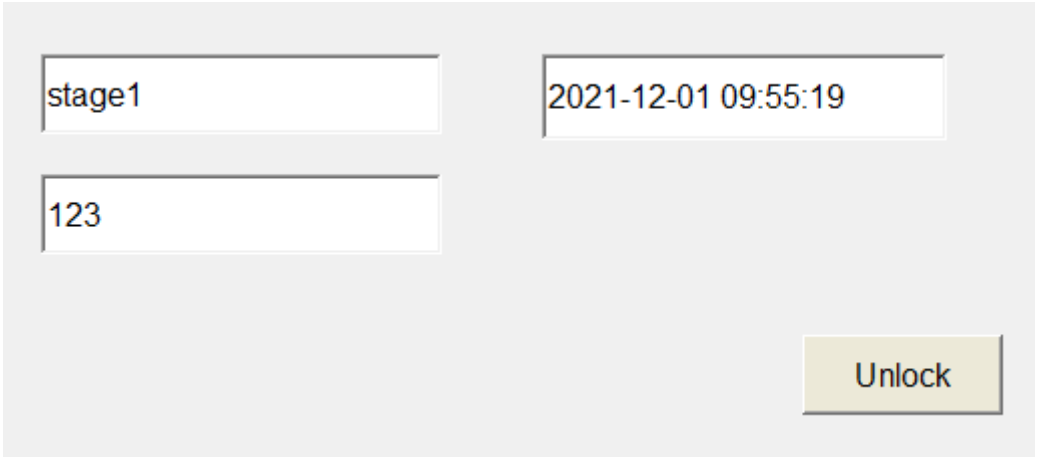
   If you are not satisfied wit the installment unlocking screen coming with the system, you can customize the unlocking screen, as shown in the following figure.

   a. First check Custom Unlock Dialog, select a pop-up screen which will pop up when the installment arrives, and then set the custom password tag, custom instal tag and custom date tag, which are all of string type. The password tag is used to obtain the input password after the installment arrives, the installment tag is used to display the name of the installment after the installment arrives, and the date tag is used to display the date of the installment after the installment arrives.

b. As shown in the following figure, add three string IO fields in the pop-up screen, bind a tag to them, and add a button and bind the UnlockCurrentInstalMent function to it. Remember to set the pop-up screen to modal type.

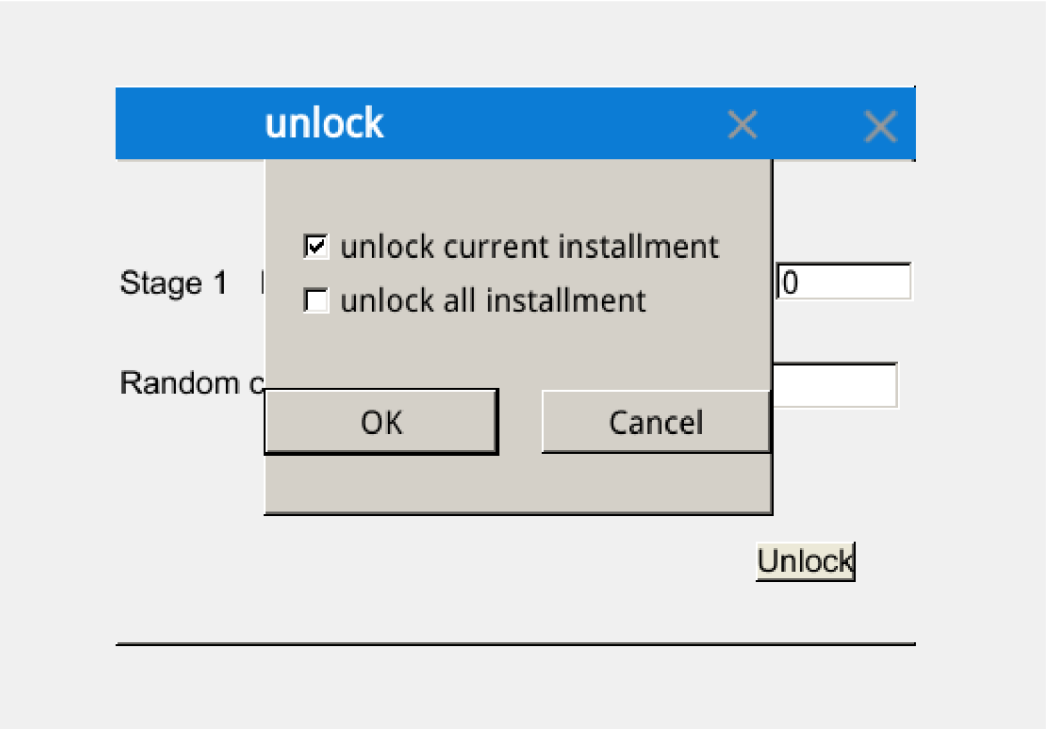






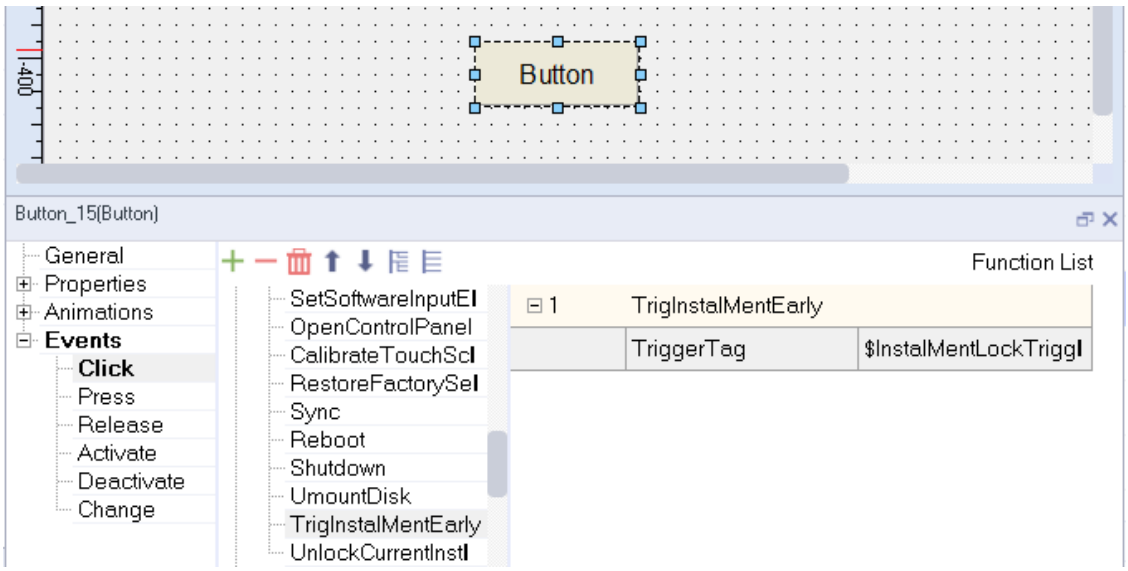
6. Unlocking with the administrator password

In addition to the installment password, you can also enter the administrator password to unlock. After you enter the administrator password, the following dialog box will pop up, and you can choose to unlock the current stage or unlock all stages.



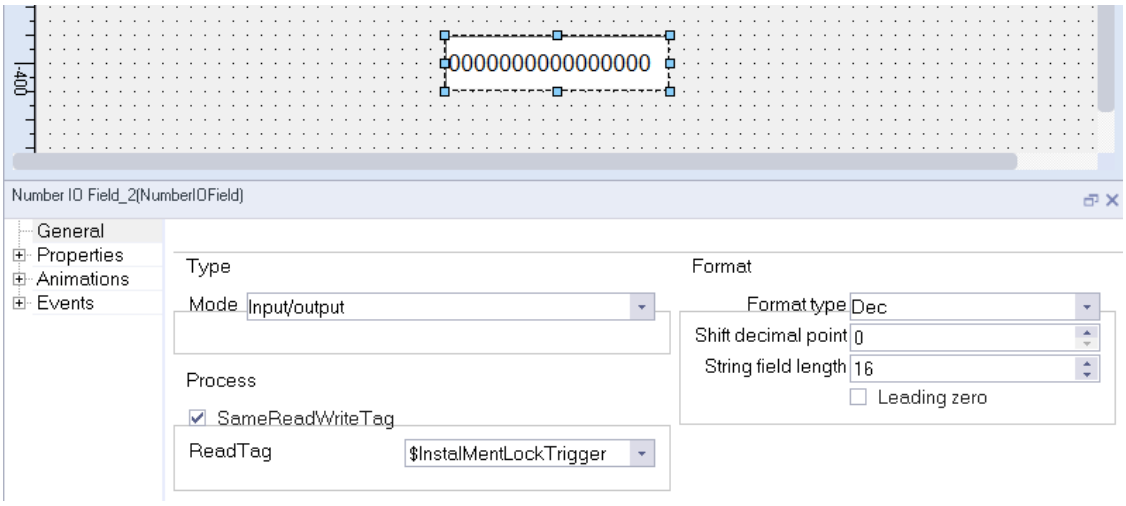
7. Trigger the installment in advance

You can trigger the next stage in advance as shown in the following figure. The button is bound to the system function TrigInstalMentEarly, and the tag is bound to the system tag $InstalMentLockTrigger. If there are multiple stages, it is necessary to trigger the next stage in advance by clicking the button.



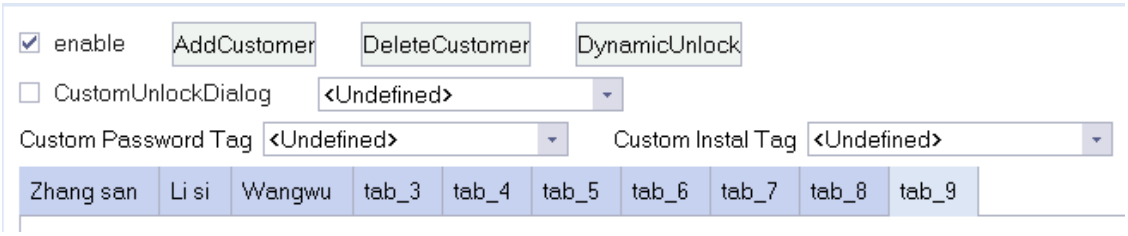
8. Remaining time of next installment

As shown in the following figure, you can use $InstalMentLockRemainTime to display the remaining arrival time for the next installment (in minutes).



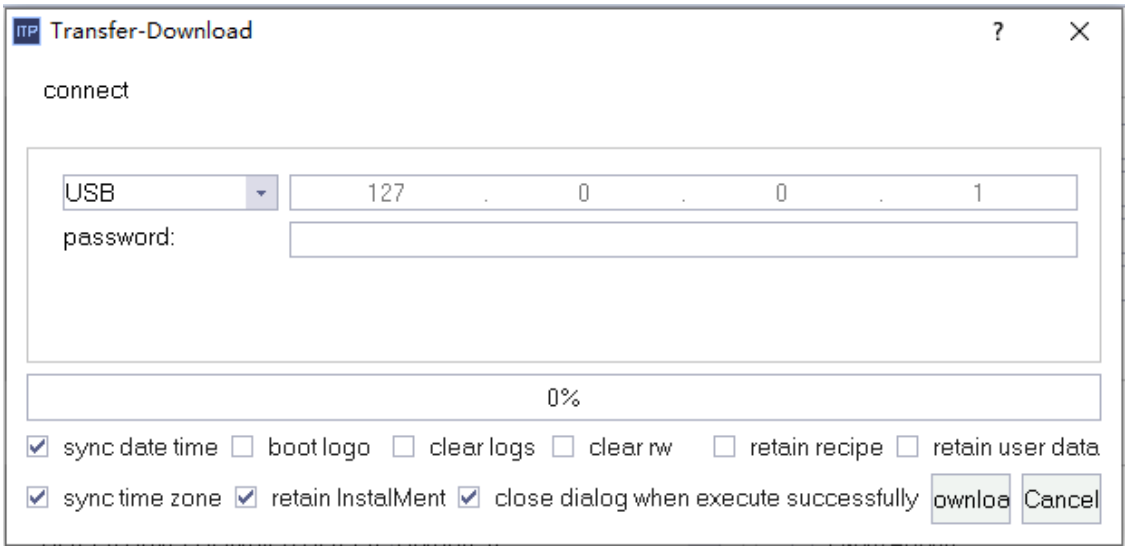
9. Creating multiple customers

As shown in the following figure, you can maintain multiple customers by adding or deleting. The customer selected during interface configuration is the installment downloaded to the HMI. The "tab_9" customer is effective at this time in the following figure.



10. Retaining installment

As shown in the following figure, you can decide whether to retain the previous installment during downloading. If it is retained, the installment configured in the project is ignored. If not, the previous information is overwritten by the downloaded installment.



15 Report

15.1 Basic information

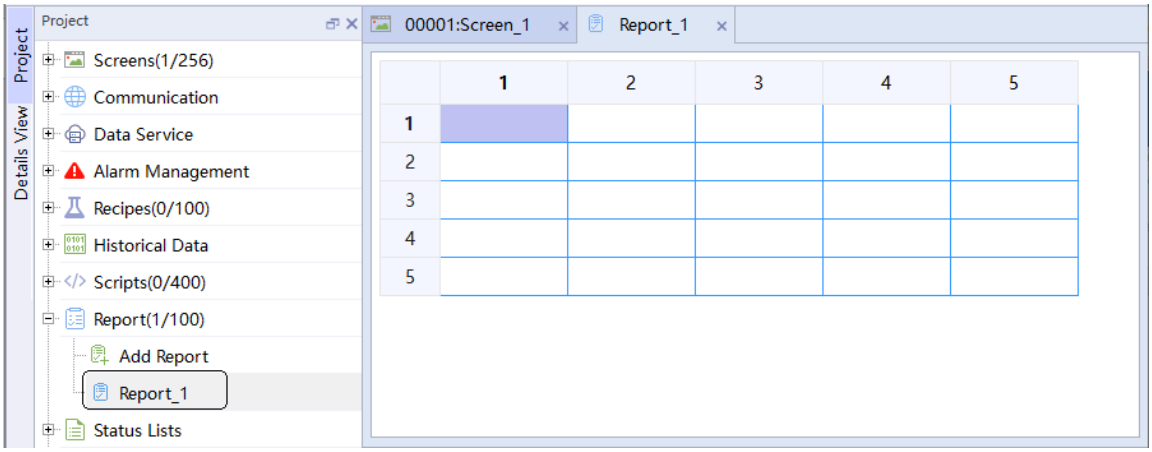
In applications, Most monitoring systems need to save and analyze the data collected by data acquisition equipment, and print a data report. The data report is a formated report containing analyzed data, which can be used for recording and analyzing the status of a monitored system object.

Data report is essential in the industrial control system, and is the final output of the whole system. Common report forms include real-time data report and historical data report (shift report, daily report and monthly report), which serve the following purposes:

- Displays static data, real-time data, historical records in a historical database and their statistical results;
- Displays all type of reports quickly;
- Easy query and statistics;
- Modifies data and write the table data into the specified variable;
- Displays multipage reports.
  For the creation and use of projects, see the sample projects: "Report_Historical Data Display" and "Report_Real-Time Data Display".

15.2 Creating a Report

In the project tree, double-click Report->Add Report to create a report as shown in the following figure:

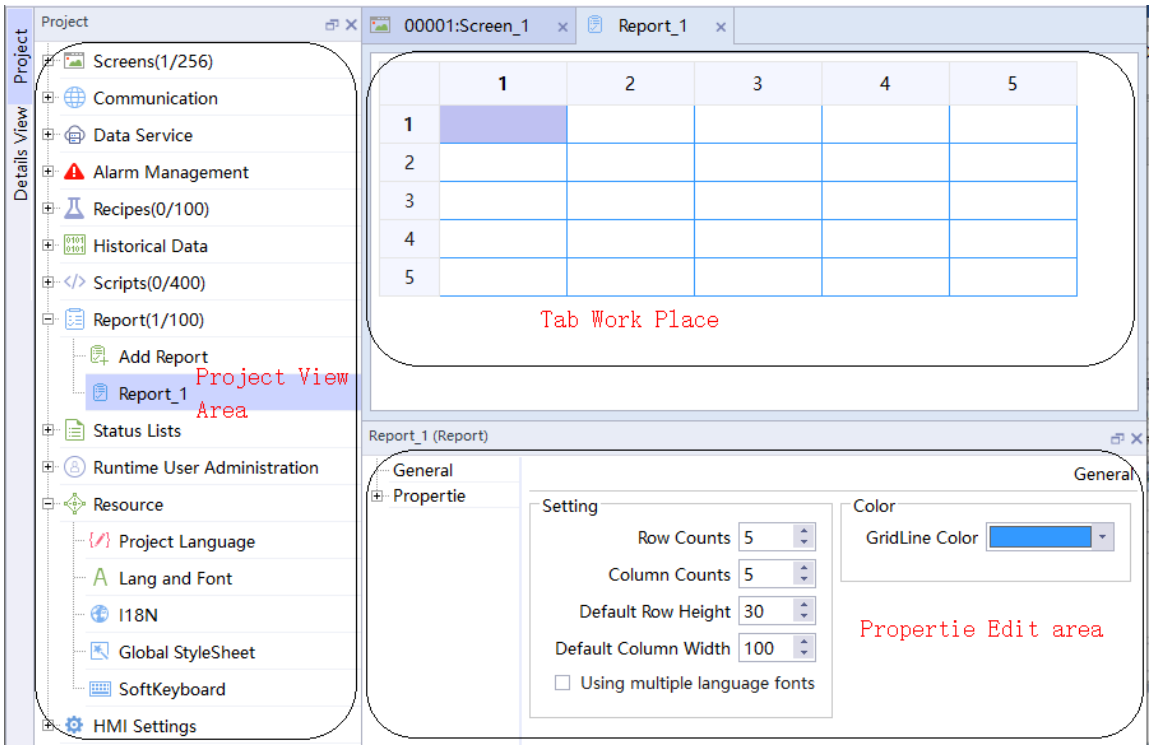


15.3 Properties and Basic Settings

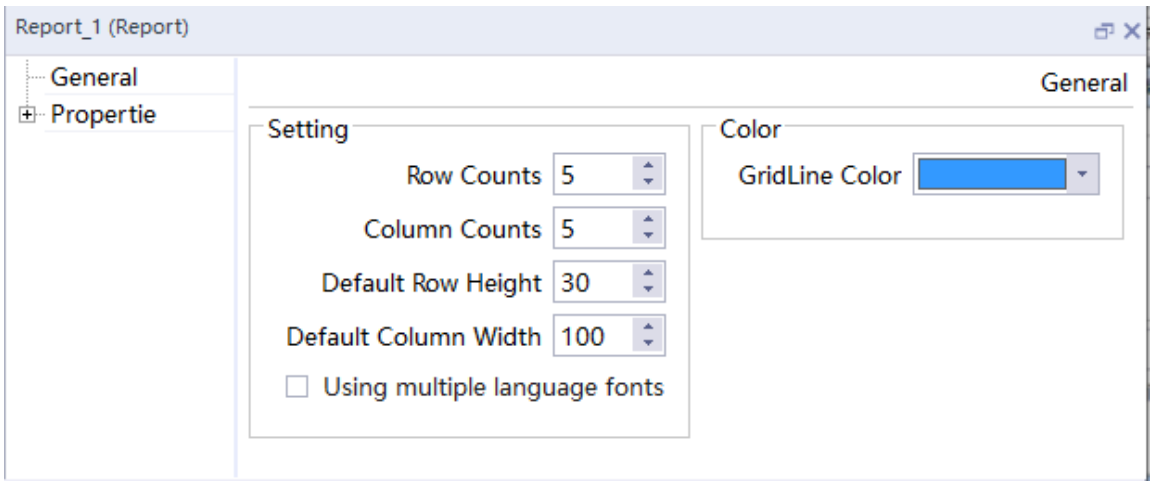
15.3.1 Report Properties

In a report, you can set report properties, report cell properties, and report history cell properties.

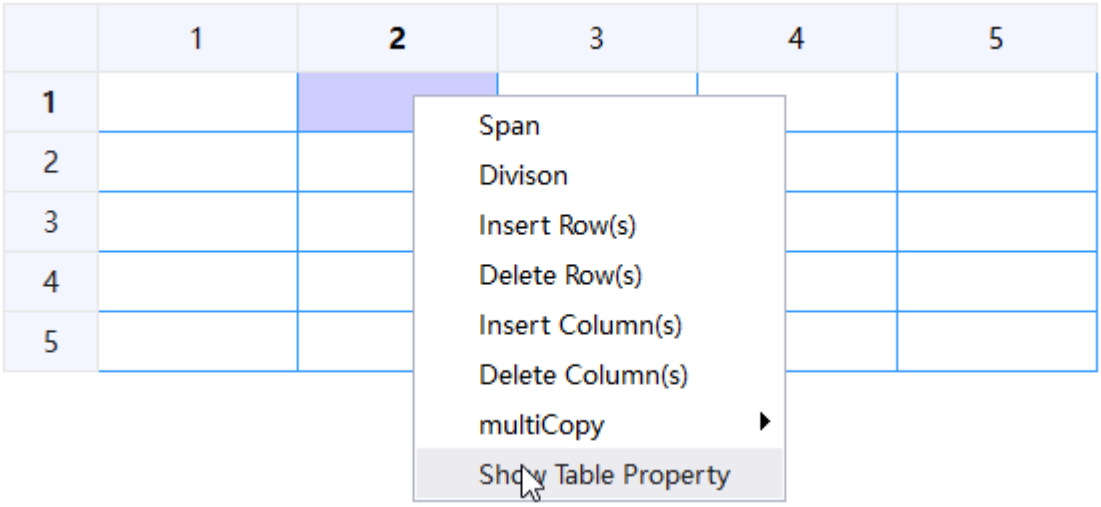
The report property setting page is as follows:



Report property setting:



- To enter the report property setting page:
  You can click the area outside the report table to display the setting page. You can also right click the table and select Show Table Property, as shown in the following figure:



- See the following table for the detailed description of the General property settings:

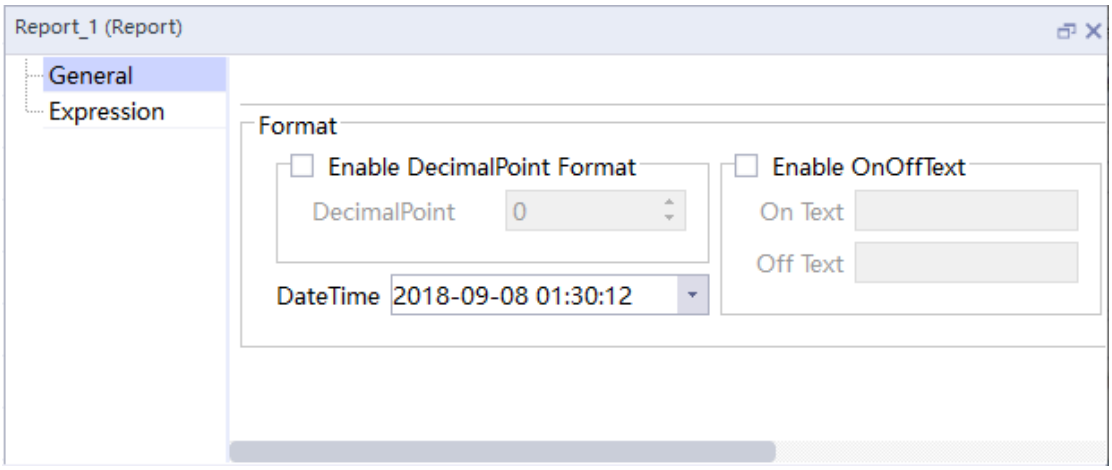
Name		Description
Settings	Row Count	The number of rows of the report table, ranging from 1 to 40, and the default is 5
	Column Count	The number of column of the report table, ranging from 1 to 30, and the default is 5
	Default Row Height	The default row height of the report table, ranging from 10 to 200.
	Default Column Width	The default column height of the report table, ranging from 10 t0 200.
	Using multiple language fonts	Check this to use different fonts for different languages in a cell. (No matter whether it is checked or not, different cells can have different font settings)
Color	Gridline Color	The border color of the table

- Comment: You can add comments for the report.

15.3.2 Property Settings of Report Cells

The real-time data report is to display and print the value of current data object according to a certain report format (user configuration) in real time, which is a reflection of instantaneous quantity. Real-time data reports consist of ordinary cells. Report normal cell property setting is the setting of the selected cell property (which can be one or more cells).

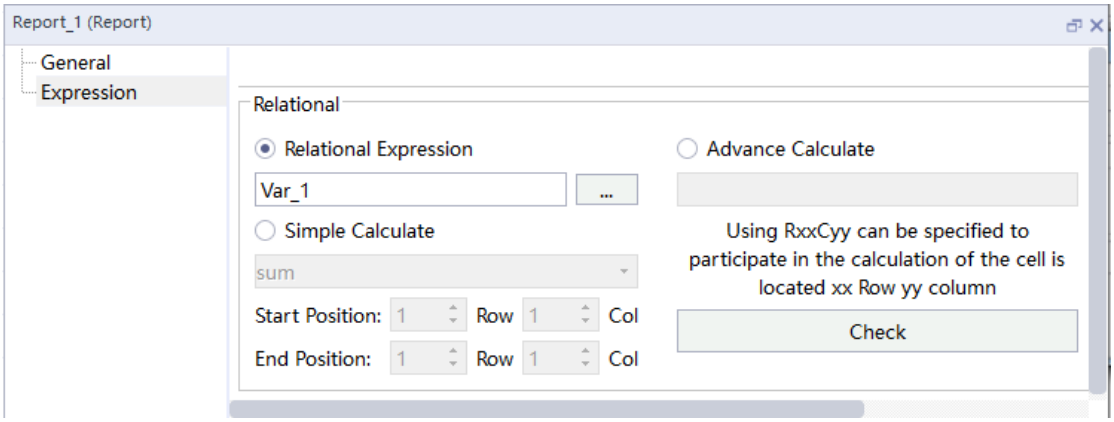
- Click the selected cell to enter cell property editing. The editing interface is as follows:



- See the following table for the detailed description of the General property settings:

Name		Description
Format	Enable Decimal Point Format	When this option is checked, the decimal point can be edited. In the decimal point, you can set the display decimal number specified by the cell display floating point number. The decimal point setting range is 0-13, and the default value is 0. In runtime, when a cell displays a floating-point number, the output is displayed in the specified decimal place.
	Enable OnOff Text	When this option is checked, the ON and OFF text can be edited, and the text corresponding to the status can be customized in the ON and OFF text editing box. In runtime, when the cell value is 0, the off text is displayed, and when the cell value is not 0, the on text is displayed.
	Date Time	You can choose the time display format in the drop-down list. In the runtime, the cell date and time will be displayed in the user-specified format.
Content of cell	Enable Edit	When this option is checked, you can edit the selected cell in the runtime. If it is not checked, the cell is read only.
	Enable Write Tag	When this option is checked, you can choose to write tags. In the runtime, you can write the values displayed by cells to the specified tags.

- Expression property setting:

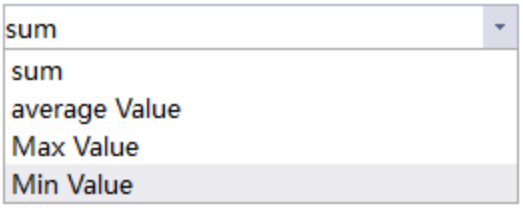


1. **Relational Expression:**
   If the user selects an relational expression, the cell is output according to the value of the expression, which can be a single tag (equivalent to binding a tag) or a single constant:

It can also be an expression of tag and tag, tag and constant, or constant and constant:

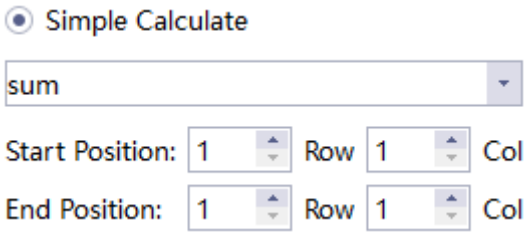
2. **Simple Calculation:**

   If the user selects Simple Calculation, the cell displays the calculation result. Simple calculation provides four operation modes, as shown in the following figure:



   They are sum, average, max and min.

   Cell configuration is as follows:

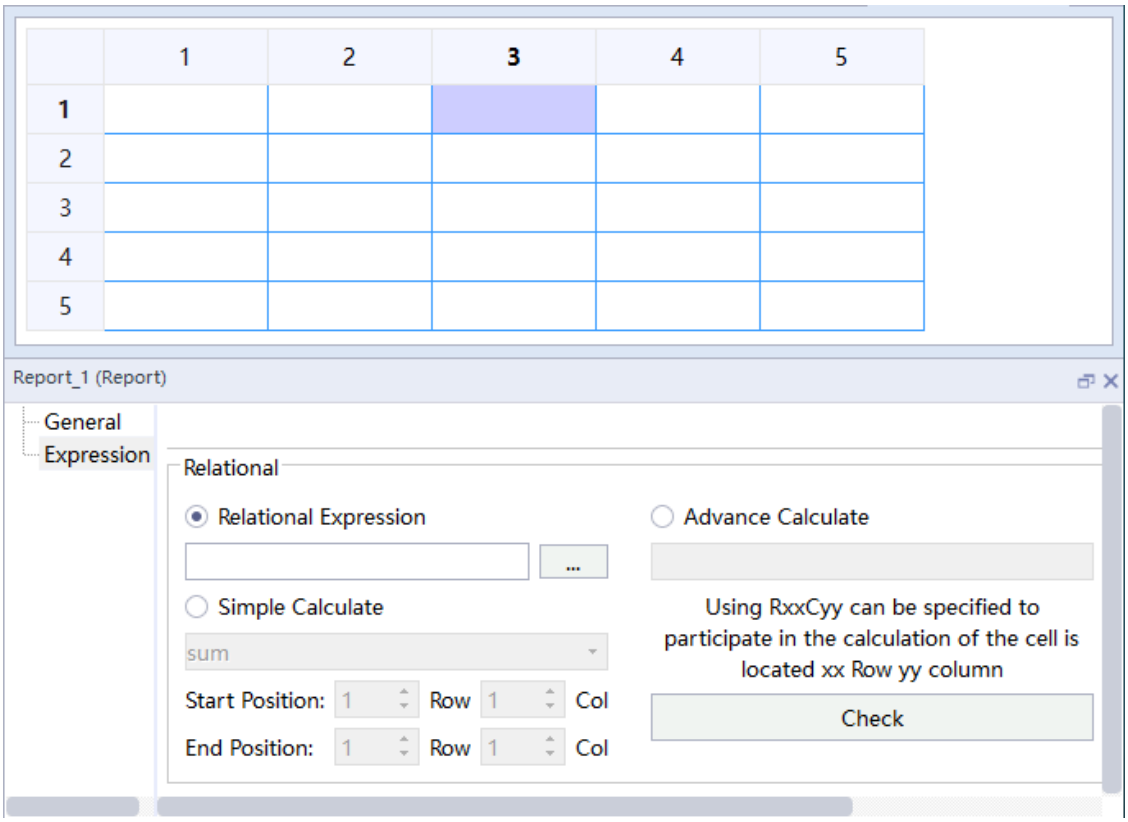


   With the setting in the above figure, the cells involved in the calculation are R1C1, R1C2, R2C1 and R2C2, where "R" means "row" and "C" means "column".

3. **Advanced Calculation**

   If the user selects Advanced Calculation, the cell displays the calculation result.

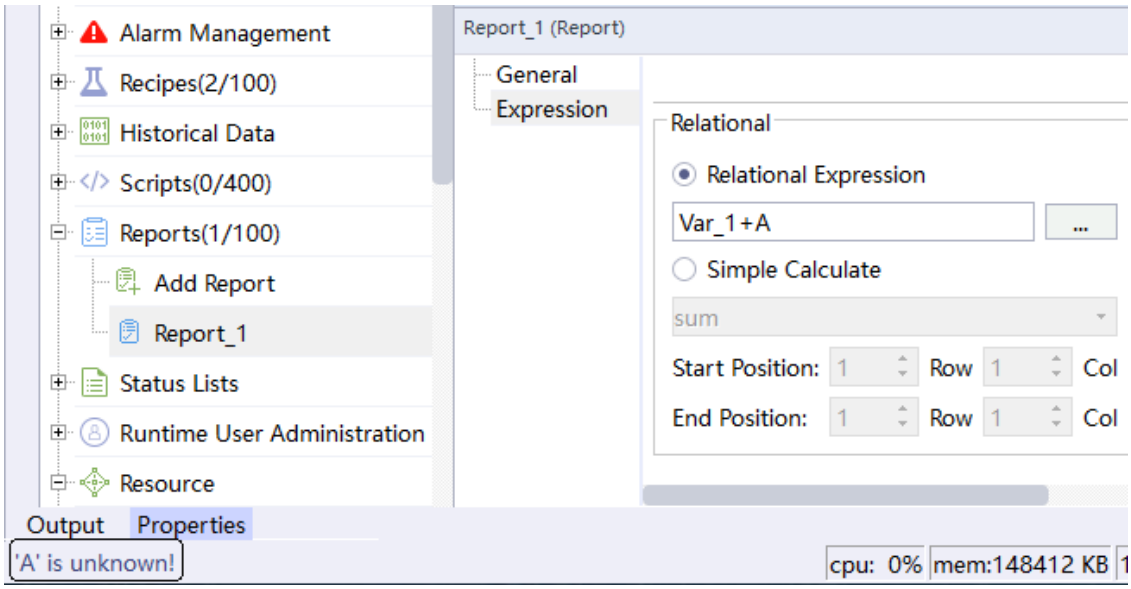
   For example, to add the value of the first cell of the first row to the value of the second cell of the first row, and display the result in the third cell of the first row, you can:



Select the cells in the third column of the first row and enter the expression R1C1+R1C2.

The rows are indicated with "R", and the columns are indicated with "C", which are case insensitive. The index starts from 1, for example, R1C1 represents the first cell of the first row.

For rational expression or advanced calculation, you can check the expressions through the Check button after editing. If the expressions are wrong, an error message will be output in the status bar, as shown in the following figure:



4. **Supported operators:**

   +-*/: four arithmetic operators

   (): brackets

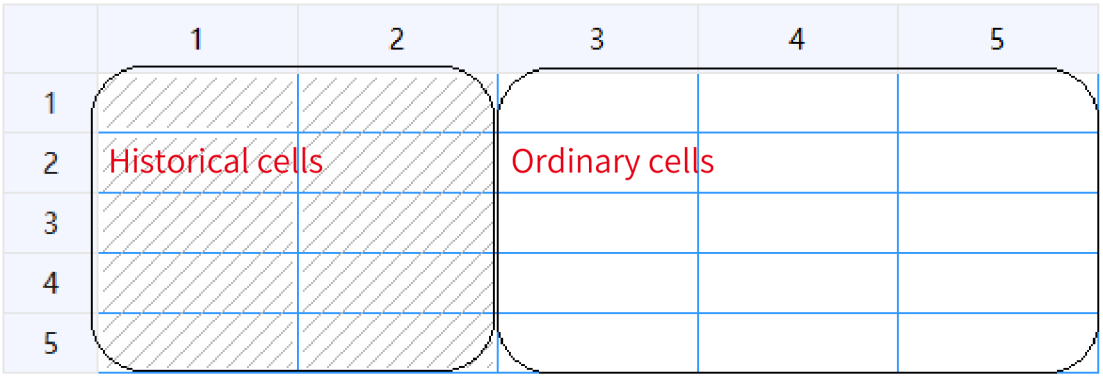
pow(x,y): power function

sqrt(): square root function

Sin, cos, tan: trigonometric functions

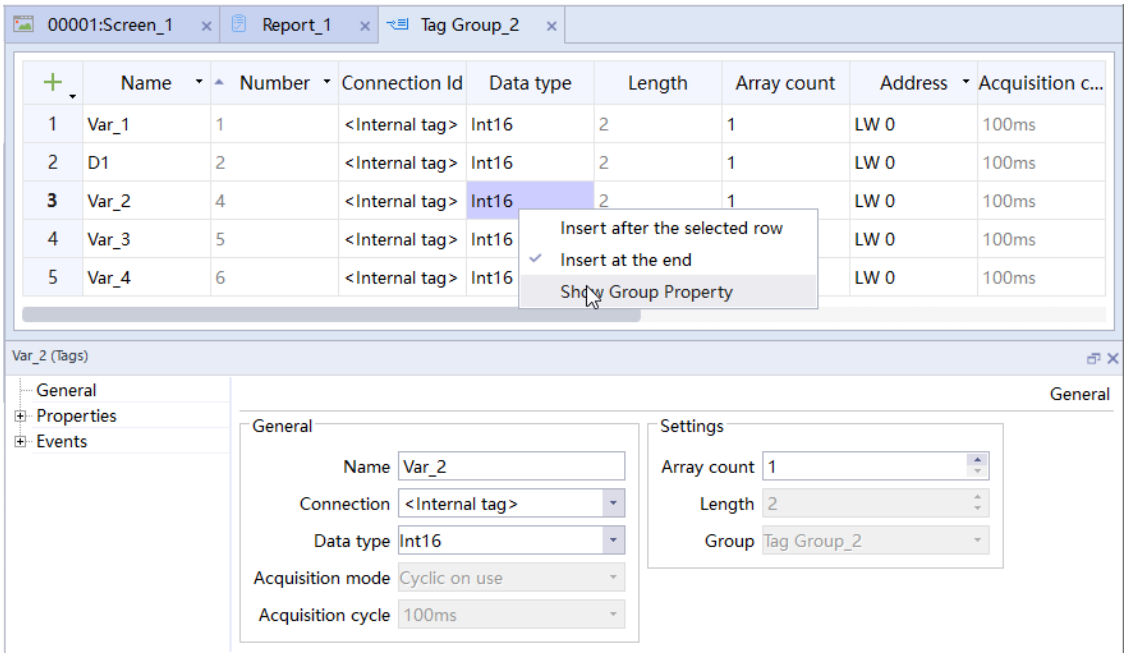
15.3.3 Property Settings of Report History Cells

Historical data report is to extract and save data records from historical database, and display and print historical data in a certain format. The historical data report is composed of historical cells. After selecting ordinary cells, click Group , and the selected cells will become historical cells, and the historical cells will be shaded, as shown in the following figure:

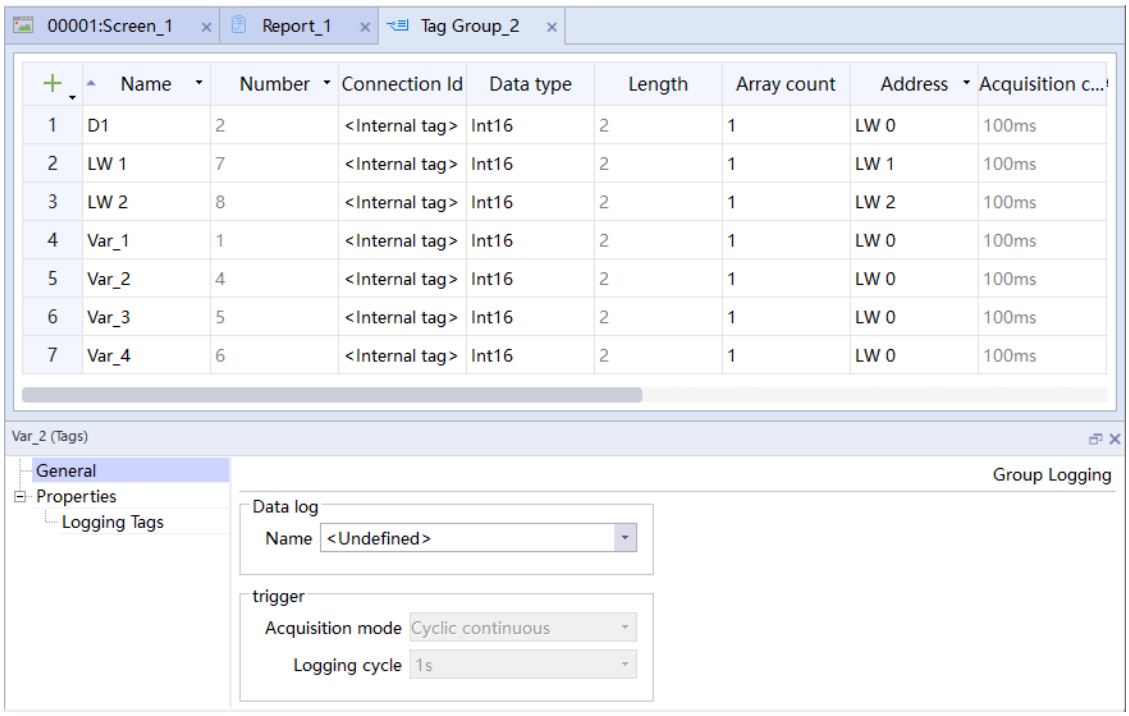


History cells are mainly used to display historical data of tags in group records. To display the specified tag in the report, you need to first add the tag to the group record. Here, take the group record in tag group 2 as an example:

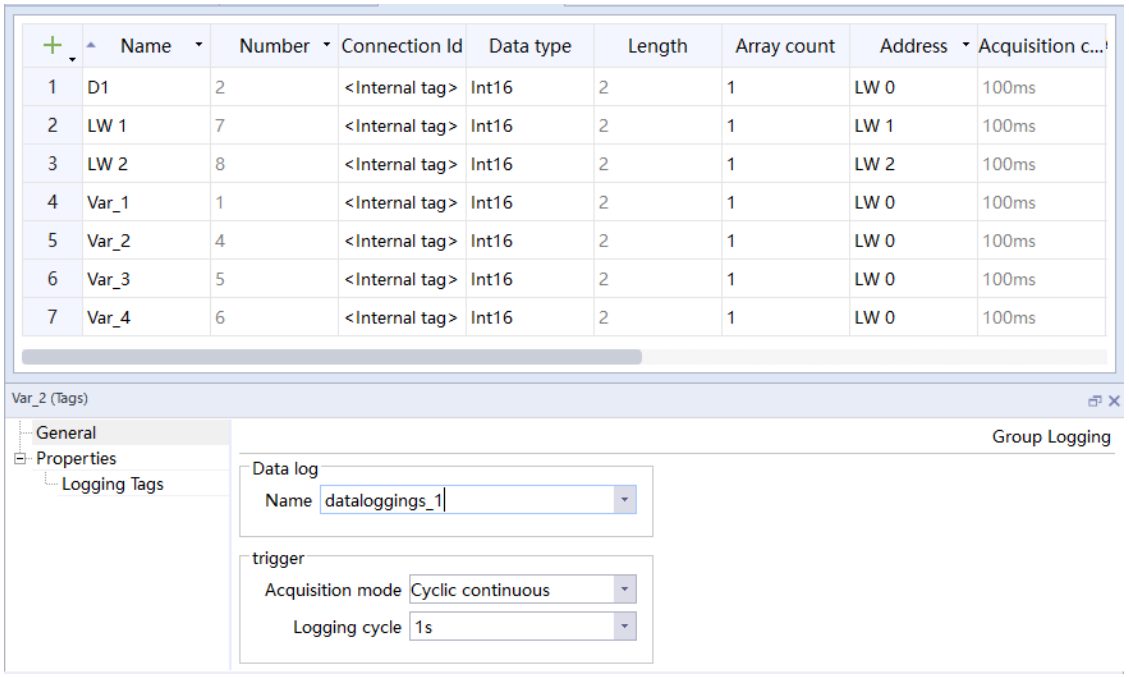
1. First, you need to enter the workspace of tag group 2 and right click the table area.



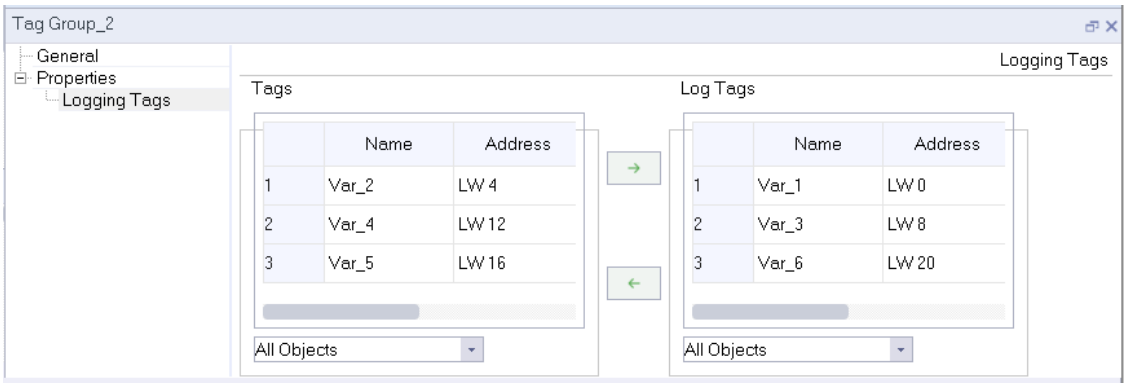
2. Select Display Group Properties, and you can see the property settings of the tag group, as shown in the following figure:



3. Then, in General, add a data log to the group, and set the trigger mode (cyclic continuous or on demand) and set the acquisition period (the trigger mode is not required for the on demand mode), as shown in the following figure:

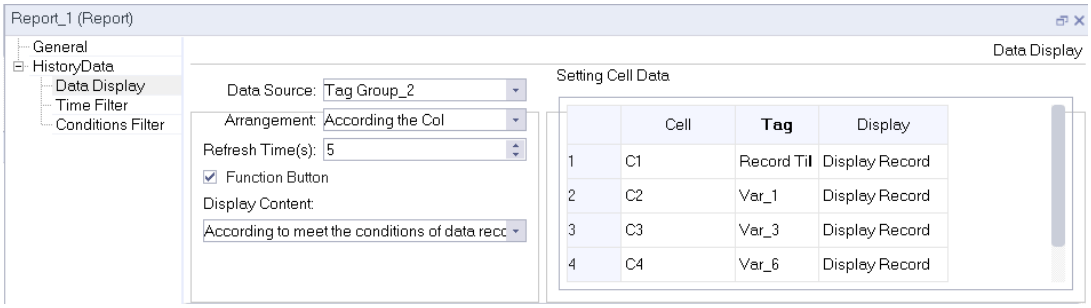


4. Finally, in Properties→Logging Tags, select the tags to be logged in the group log as the data sources that can be displayed in the history table, as shown in the following figure:



The historical cell properties are set as follows:

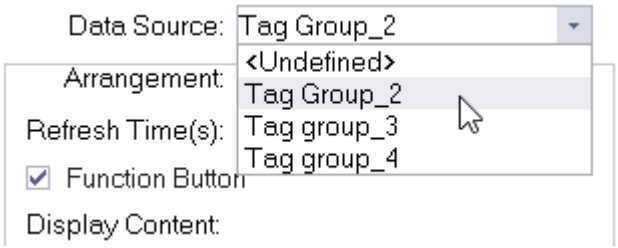
- General
  The general property settings of history cells are consistent with those of ordinary cells, see *"15.3.2 Property Settings of Report Cells" on page 331*.
- Data Display



The Data Display property of the historical cell is described as follows:

Data Source

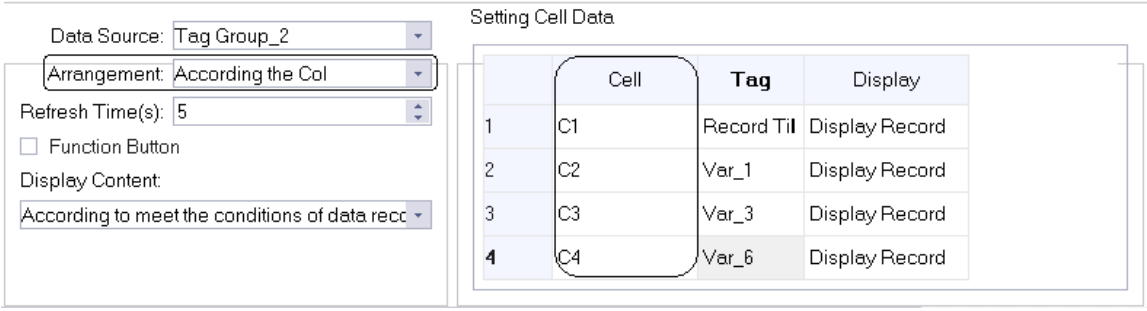
The history cell displays the source of historical data, and the optional object is the tag group configured by the user:



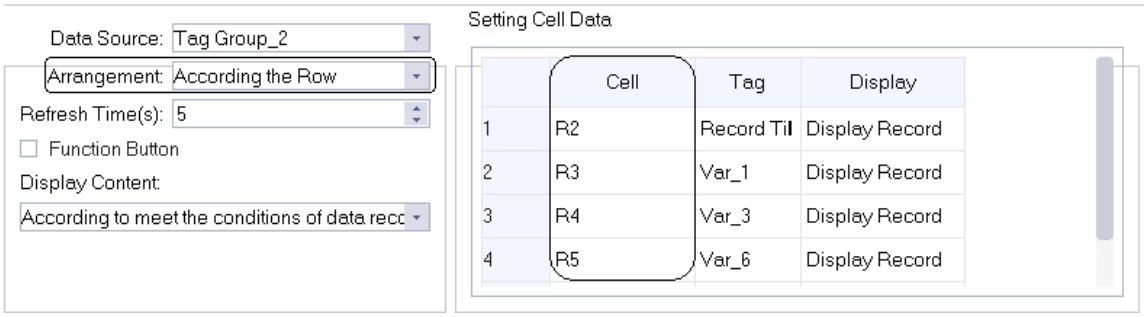
Arrangement

You can choose by column/by row, and the default is by column.

When you select by column, set the cells in the cell data table to be listed by columns of the report, and display historical data in columns:



When you select by row, set the cells in the cell data table to be listed by rows of the report, and display historical data in rows:

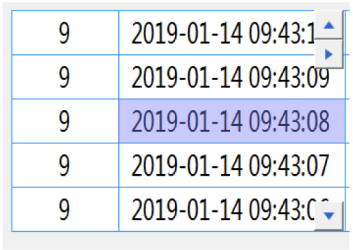


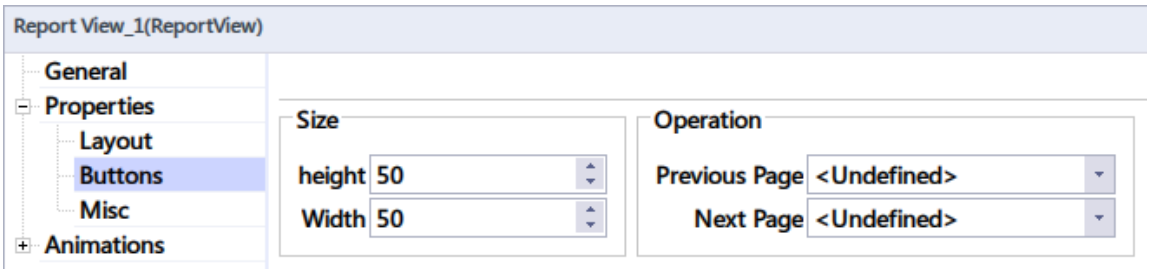
Refresh Time

The time to update the historical data, ranging from 0-65535, and the default value is 0. When the refresh time value is set to 0, the history table is not refreshed (it is recommended not to refresh frequently when the amount of data is large).

Function Button

Check the Function button and click the history table in runtime, you can navigate the history data in the current history table, and you can set the page navigation button in the report attributes in the screen.



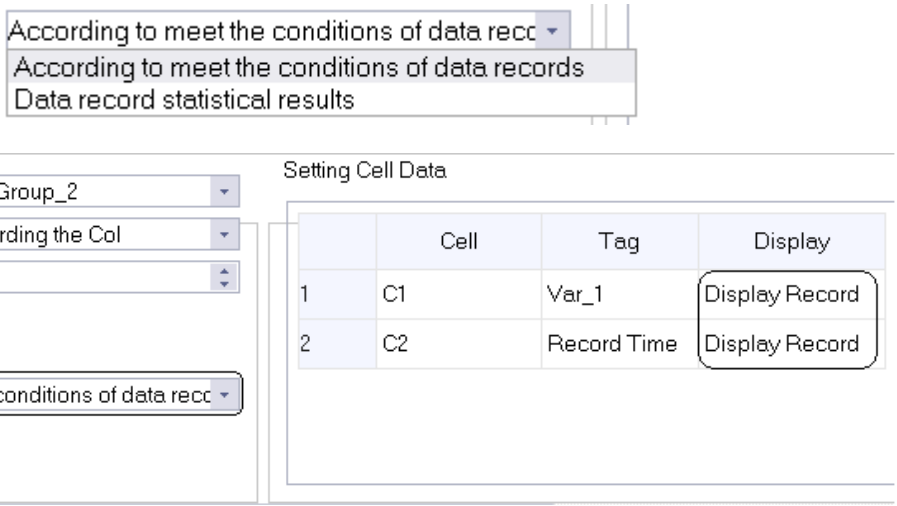


Attribute	Description
Height/width	The size of the button is set to 50 by default, and the range is 30–100
Previous/next page	They are associated with a Bool tag and navigate pages according as the tag value changes. They are null by default

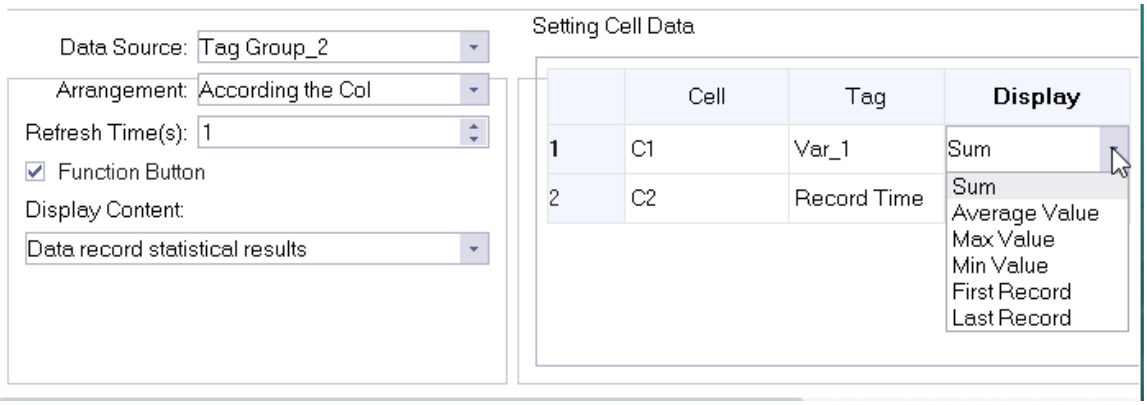
If this is not checked, the latest historical data can only be updated and displayed according to the refresh time, and the button prosperities are ineffective. It is not checked by default.

Display Content

There are two options, one is to display the data record based on the filter result, which corresponds to the display record of the selected object in the display settings table:



The other is to display statistical results. Select to display statistical results. You can select the following statistical methods in the display columns in the table:



Sum: The recorded tags are summed, and the result is displayed in the historical cell.

Average: The recorded tags are summed, and the result is displayed in the historical cell.

Max: The recorded tags are compared, and the maximum value is displayed in the historical cell.

Min: The recorded tags are compared, and the minimum value is displayed in the historical cell.

First Record: Displays the data of the first record.

Last Record: Displays the data of the last record.

Setting cell data:

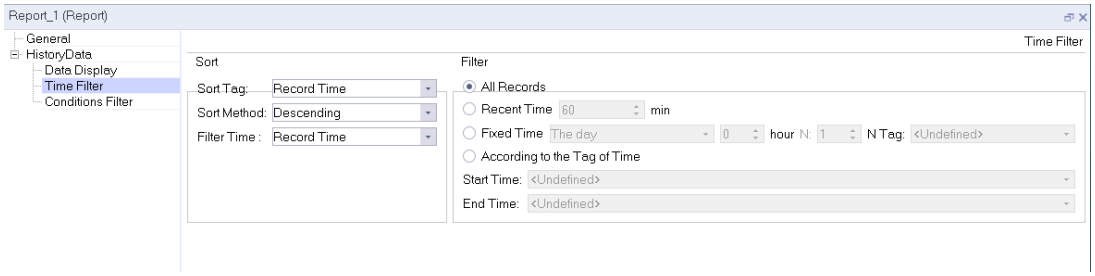
Cell: Displays report rows or columns according to the sort method. It is read only.

Tag: You can select the record tag or record time in the selected group of records in Data Source.

Display: It is decided by different ways of displaying contents. If you select the way of displaying data records according to filtering results, the history table will display recorded data; If you select the way of displaying statistical results, the history table will display corresponding statistical results.

- Time Filter

1. The Timer Filter setting page is as follows:



2. The Timer Filter settings described as follows:

Sort:

1. Sort TagThe object for sorting, which can be the record tag selected in the data source array or the record time.
2. Sort MethodSorts in ascending or descending order according to the value of the sort tag.
3. Filter TimeSpecifies the record time to be the object to be filtered.

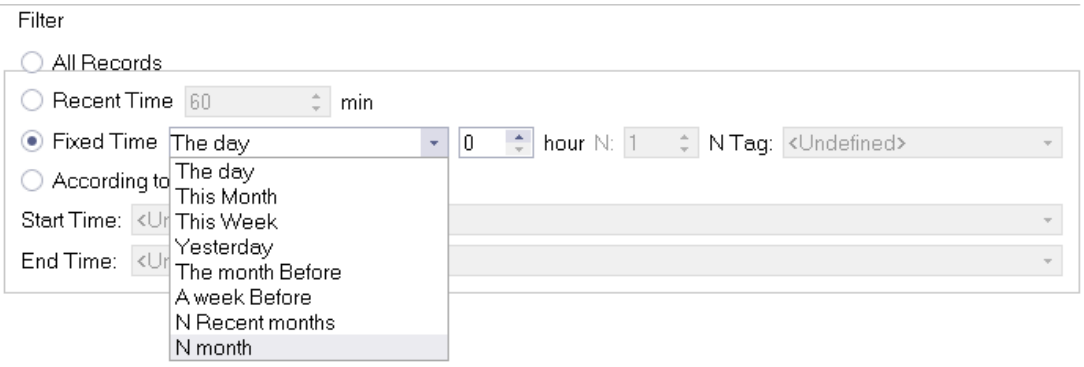
Filter:

1. All Records

   If you select this mode, the data of all records in the specified record group are filtered.
2. Recent Time

   Select this mode to filter the data recorded in the specified group of records in the last X minutes according to the record time. X ranges from 0 to 2147483647 minutes. The default value is 60 minutes.
3. Fixed Time

   If you select this mode, you can filter data through one of the following criteria:



The day: You can select a certain time of the day, such as 6:00, and the data recorded from 6:00 to the current time will be filtered from the historical records.

This month: You can select a certain time of the month, such as 6:00, and the data recorded from 6:00 on the first day of the month to the current time will be filtered from the historical records.

This week: You can select a certain time of the week, such as 6:00, and the data recorded from 6:00 on Monday of the week to the current time will be filtered from the historical records.

Yesterday: You can select a certain time of yesterday, such as 6:00, and the data recorded from 6:00 of yesterday to 6:00 today will be filtered from the historical records.

A week before: You can select a certain time of the last week, such as 6:00. Assume that today is Wednesday, and the data recorded from 6:00 of last Wednesday to 6:00 today will be filtered from the historical records.

The month before: You can select a certain time of the last month, such as 6:00. Assume that today is October 16, and the data recorded from 6:00 of of September 16 to 6:00 today will be filtered from the historical records.

N recent months: You can select a certain time of last N months, such as 6:00 of last 2 months. Assume that today is October 16, and the data recorded from 6:00 of of August 16 to 6:00 today will be filtered from the historical records. N can be a constant or a tag.

Nth month: You can select a certain time of the Nth moth, such as 6:00 of the 10th month. The data recorded from 6:00 of October 1 to 6:00 on the first day of the next month will be filtered from the historical records. N can be a constant or a tag.

4. According to the Tag of Time

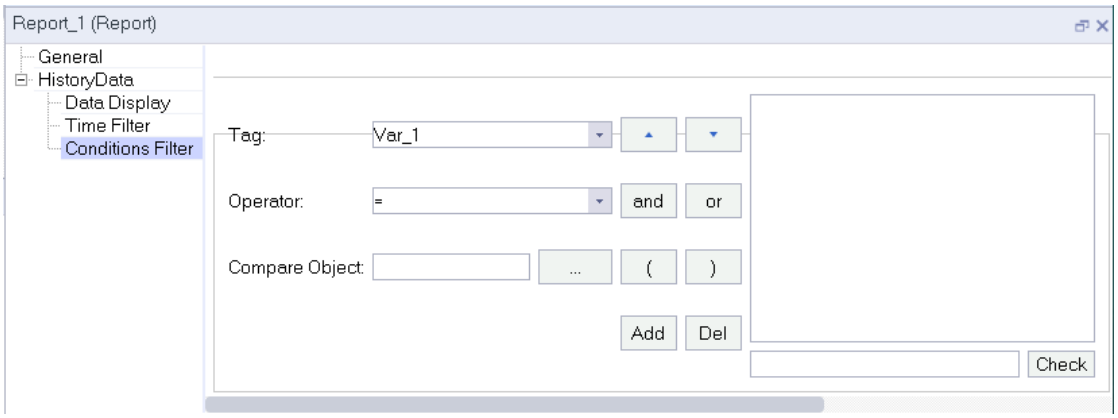
   If you select this mode, you can filter the data recorded from the start time to the end time.

   The start time and end time must be a datetime type tag.

- Conditions Filter

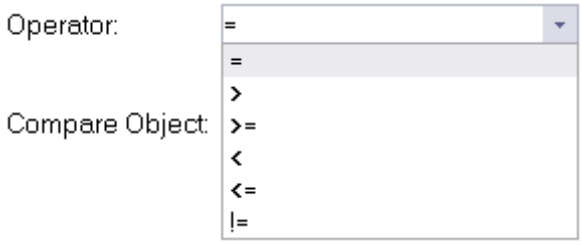
  Through conditional filtering, you can filter the data records that meet the conditions from the group records.

The Conditions Filter property settings are described as follows:

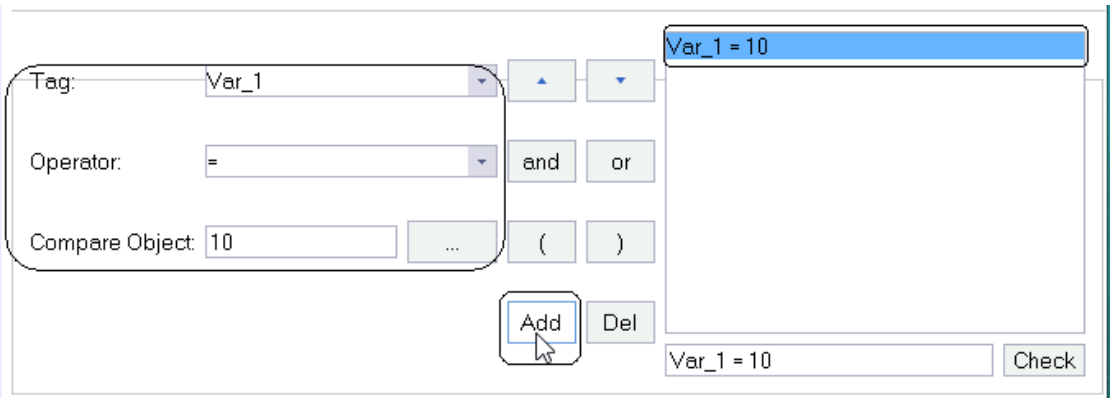
Tag: You can only select a tag recorded in the data source.

Operator: The comparison operators include =, >, >=, <, <= and !=.



Comparison Object: It can be any created tag, constant or string (the string needs to be enclosed in single quotation marks, and must not contain any operator mentioned above).

After you set the tag, operator and compare object, click Add to add the condition statement to the condition box, as shown in the following figure:

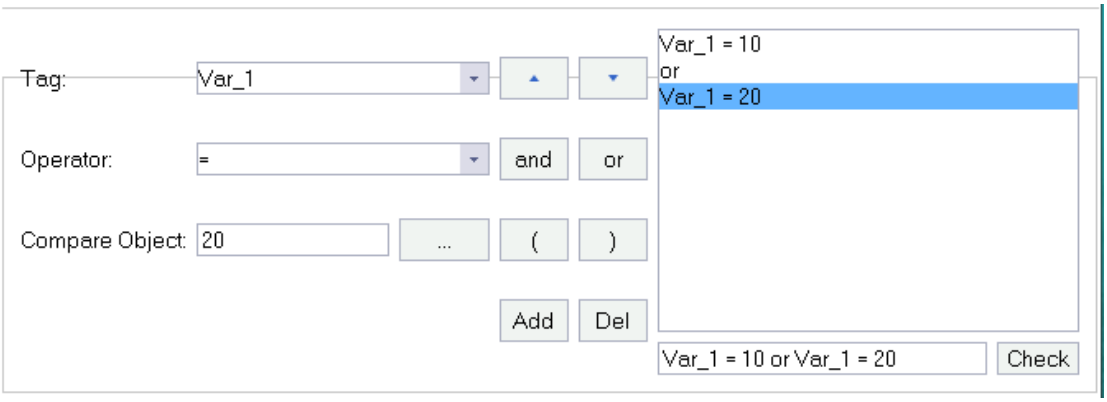


You can add multiple conditions to the condition box. You can use the "and" and "or" operators to join them logically.

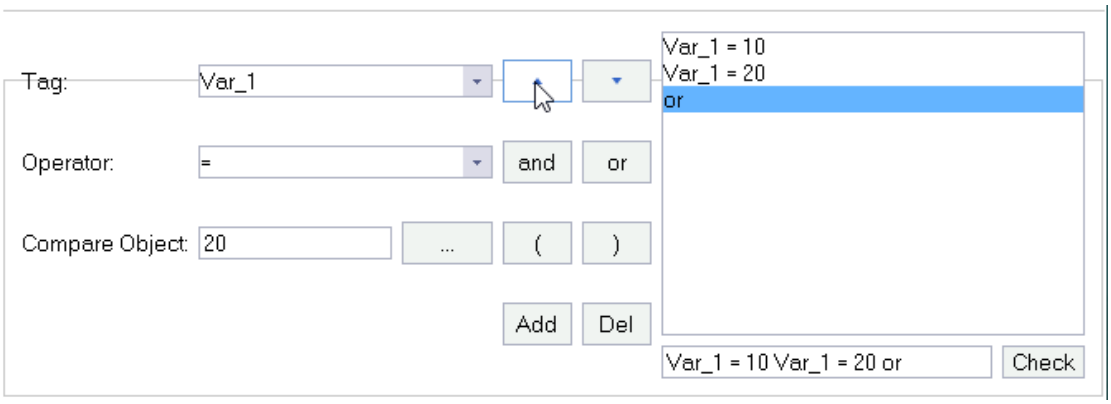
and: To filter the data, all the conditions joined by the "and" operator must be true.

or: To filter the data, any of the conditions joined by the "or" operator must be true.

For example, if you want to filter historical data with tag_1 being 10 or 20, you can set the filter condition as follows:



You can also move the statement up and down in the condition box by Move Up and Down buttons:

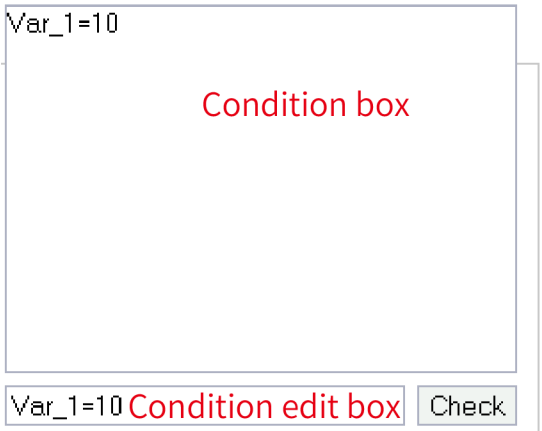


In the above figure, the third statement is moved up one position.

In addition, you can change the priority of a condition with parentheses.

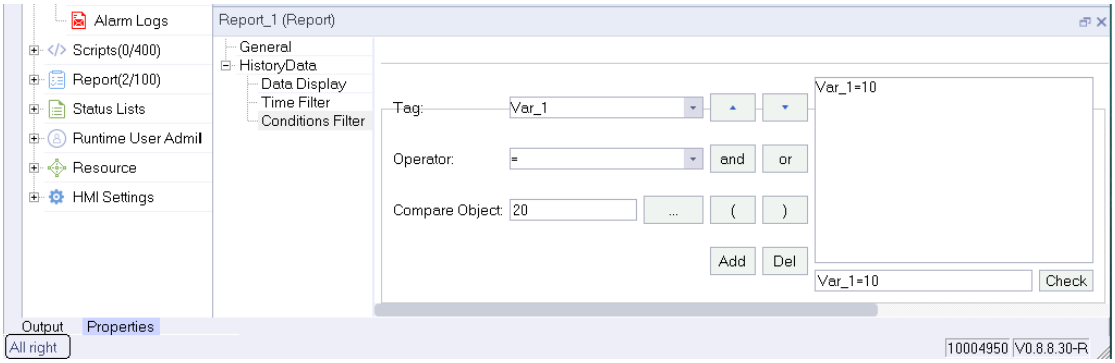
To delete a condition, select the condition statement you want to delete in the box, and then click the Delete button.

You can also edit the conditions directly in the condition edit box.

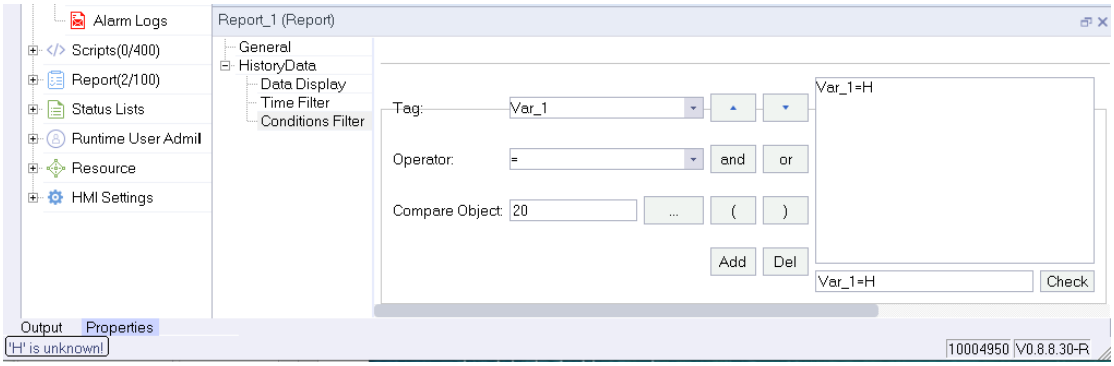


Then press Enter or move the focus away from the box to synchronize the changes to the condition box.

After editing all the conditions, click the Check button to check the written condition statement. If the condition statement is fine, the following prompt will appear in the status bar in the lower left corner:



If the statement is written incorrectly, an error message will be prompted in the status bar to help the user correct it:



15.4 Common Operations of the Report

- **Toolbar**

  The toolbar enables you to perform the following operations on the report editor:

  微软雅黑 Font: Sets the text font of a single or multiple cells.

  14 Font size: Sets the text font size of a single or multiple cells.

  Bold: Bolds the text of a single or multiple cells.

  Italic: Italicizes the text of a single or multiple cells.

  Underline: Underlines the text of a single or multiple cells.

  Left align: Left aligns the text of a single or multiple cells.

  Center align: Center aligns the text of a single or multiple cells.

  Right align: Right aligns the text of a single or multiple cells.

  Foreground and background colors

  Foreground color: Sets the foreground color of a single or multiple cells.

  Background color: Sets the background color of a single or multiple cells.

  Size objects horizontally equal: Changes the widths of all selected columns to the width of the last column in the selection.

  Size objects vertically equal: Changes the heights of all selected rows to the height of the last row in the selection.

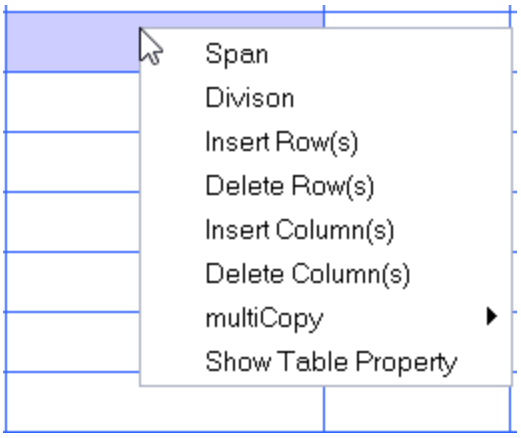
  Size objects size equal: Changes the heights and widths of all selected cells to the height and width of the last cell in the selection.

  Group: Changes the selected cells into historical cells.

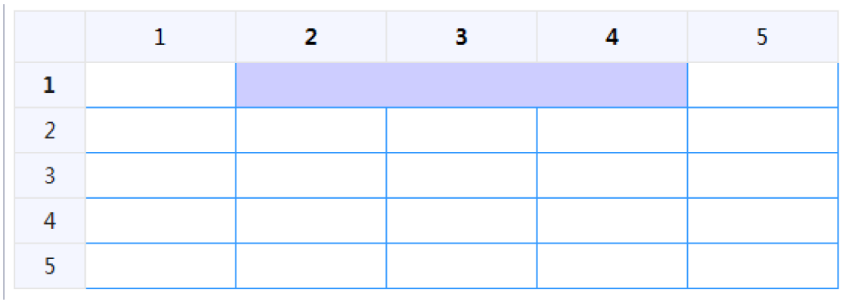
  Ungroup: Changes the selected historical cells into ordinary cells.

- **Context menu**

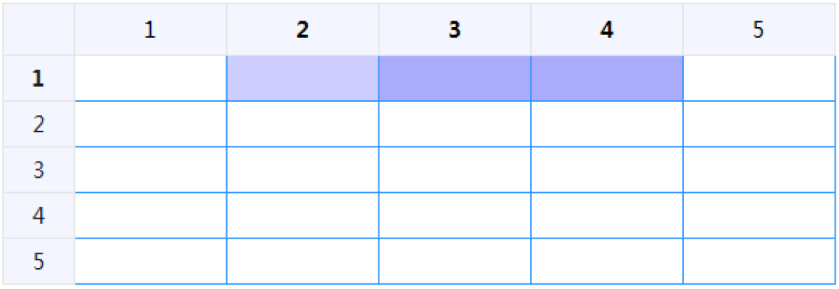
  If you right click the report editor, the following menu will pop up:



- **Span**
  Merges all selected cells.



- **Split**
  Restores the merged cells.



- **Insert Row(s)**
  If you select a number of rows, this command inserts the same number of new rows before the first row in the selection.
- **Delete Row(s)**
  Deletes all selected rows.
- **Insert Column(s)**
  If you select a number columns, this command inserts the same number of new columns before the first column in the selection.
- **Delete Column(s)**
  Deletes all selected columns.
- **MultiCopy**
  If you select multiple cells, this command provides the following options:

  1. Expression Tag

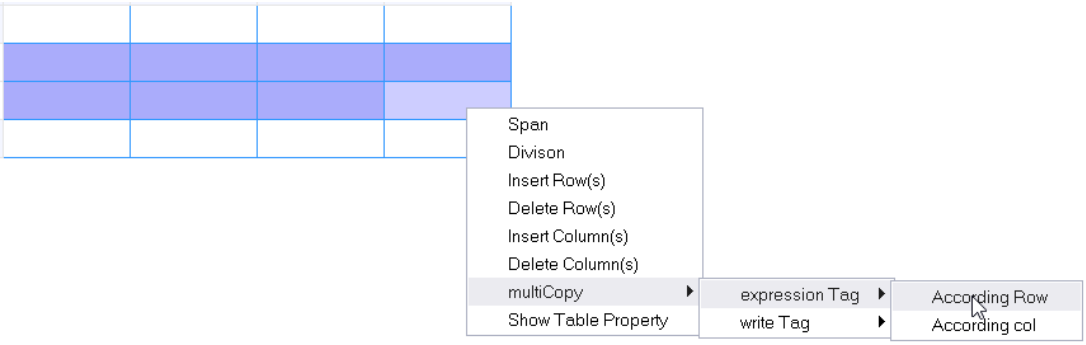
According Row: If you choose row-based multiple copy, the selected cells are based on the tags associated with the last selected cell. From the tag table, select the reference tag number to start, find the tags of the same type, and automatically fill them into the associated expressions of cells in row order. If the number of tags of the same type queried in the tag table is not enough to fill the cells, then fill the remaining cells with the reference tags after filling the tags of the same type.

According Column: If you choose column-based multiple copy, the selected cells are based on the tags associated with the last selected cell. From the tag table, select the reference tag number to start, find the tags of the same type, and automatically fill them into the associated expressions of cells in column order. If the number of tags of the same type queried in the tag table is not enough to fill the cells, then fill the remaining cells with the reference tags after filling the tags of the same type.

2. Write Tag

According Row: If you choose row-based multiple copy, the selected cells are based on the tag of the last selected cell. From the tag table, select the reference tag number to start, find the tags of the same type, and automatically fill them into the tag in row order. If the number of tags of the same type queried in the tag table is not enough to fill the cells, then fill the remaining cells with the reference tags after filling the tags of the same type.

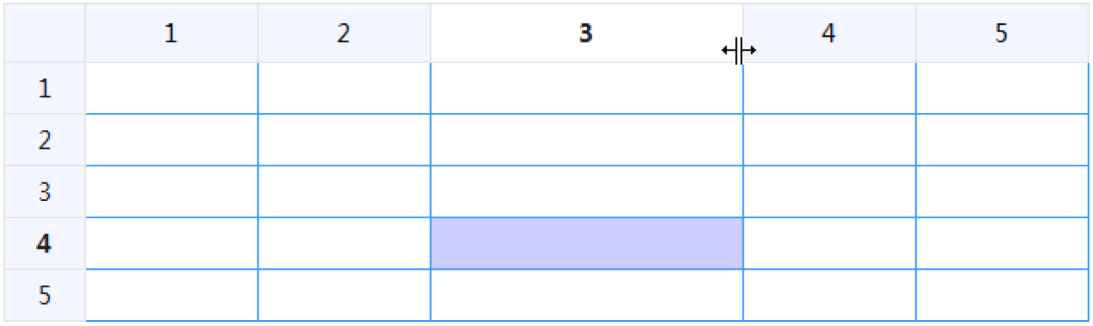
According Column: If you choose row-based multiple copy, the selected cells are based on the tag of the last selected cell. From the tag table, select the reference tag number to start, find the tags of the same type, and automatically fill them into the tag in column order. If the number of tags of the same type queried in the tag table is not enough to fill the cells, then fill the remaining cells with the reference tags after filling the tags of the same type.



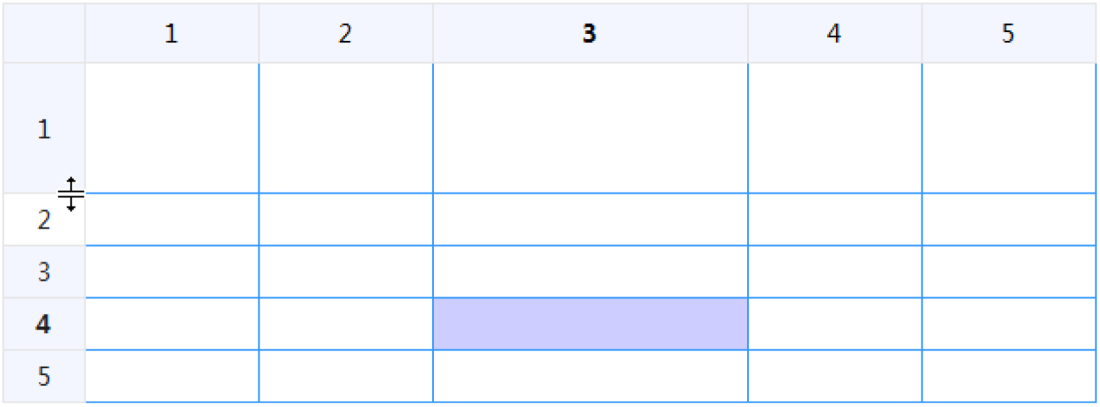
Show Table Property: Right-click the report editor and select Show Table Property. The Properties window will show the properties of the report table.

- **Adjusting row height and column width**

  You can drag the right boundary of a column heading to change the column width, as shown in the following figure:



You can drag the bottom boundary of a row heading to change the row height, shown in the following figure:



15.5 Examples

Example 1: Creating a Daily Output Report of a production line through the report function

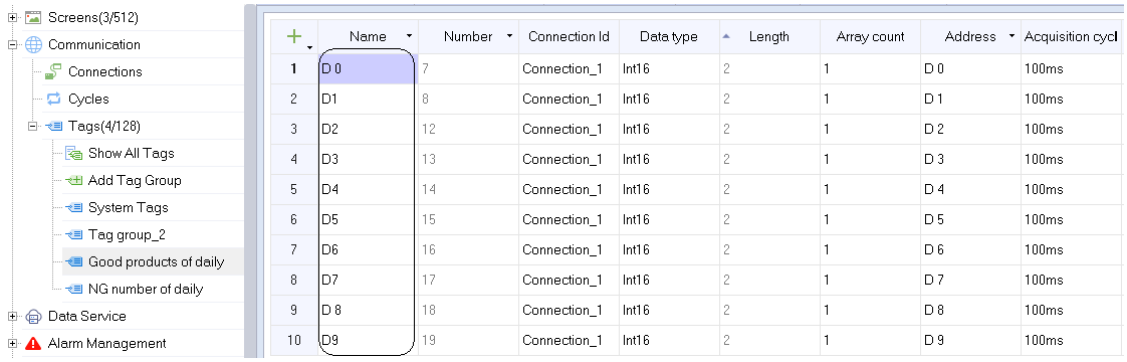
Visual design:

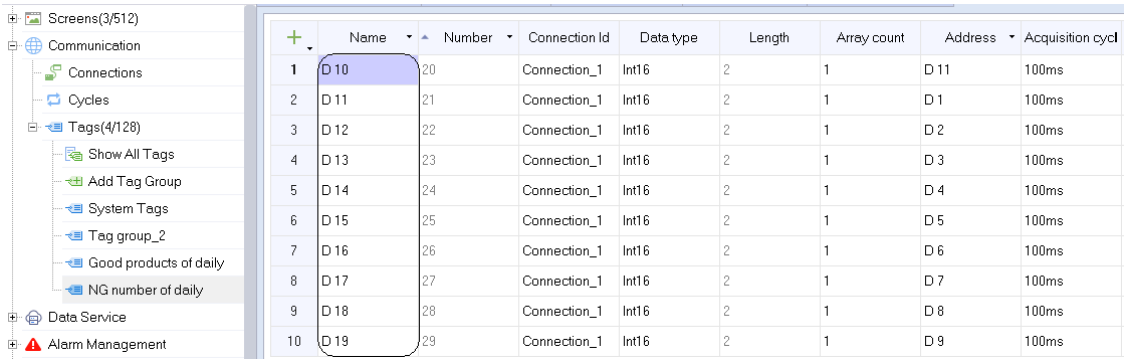
1. In the Project view, create a report.
2. In the report editor, adjust the table to 13 rows and 5 columns.
3. Adjust the size of rows and columns and straddle cells as needed.
4. Input static text, such as the report title, item names, and periods, directly in the corresponding cells.
5. Set the background color. For example, you can set the background color of the header to yellow, and the Total row to light blue as follows:

	1	2	3	4	5
1	XXXStatistics of daily output of production line				
2	Period	Total output	Good products number	NG number	Pass rate
3	8:00~9:00				
4	9:00~10:00				
5	10:00~11:00				
6	11:00~12:00				
7	12:00~13:00				
8	13:00~14:00				
9	14:00~15:00				
10	15:00~16:00				
11	16:00~17:00				
12	Total				

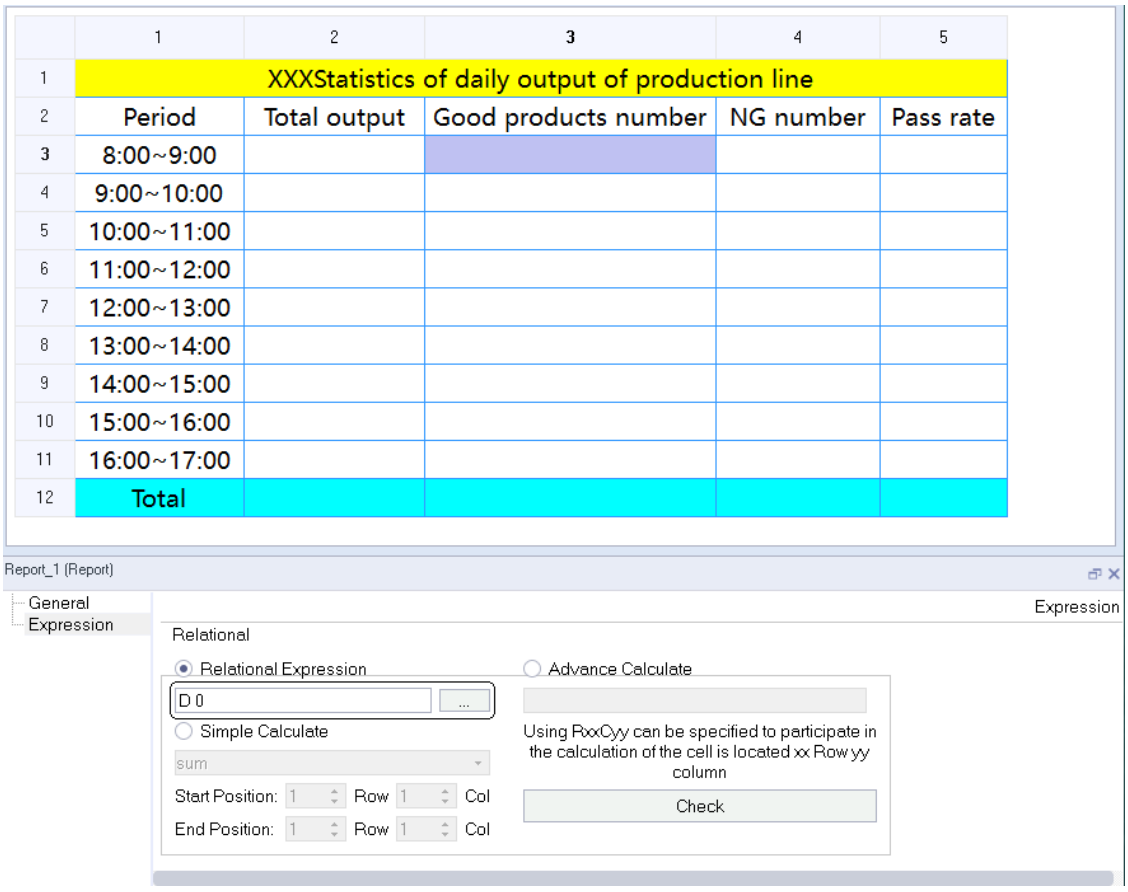
Function implementation

1. Create a tag group for good products of the day and NG products of the day respectively, and add statistical tags in the groups, as shown in the following figure:

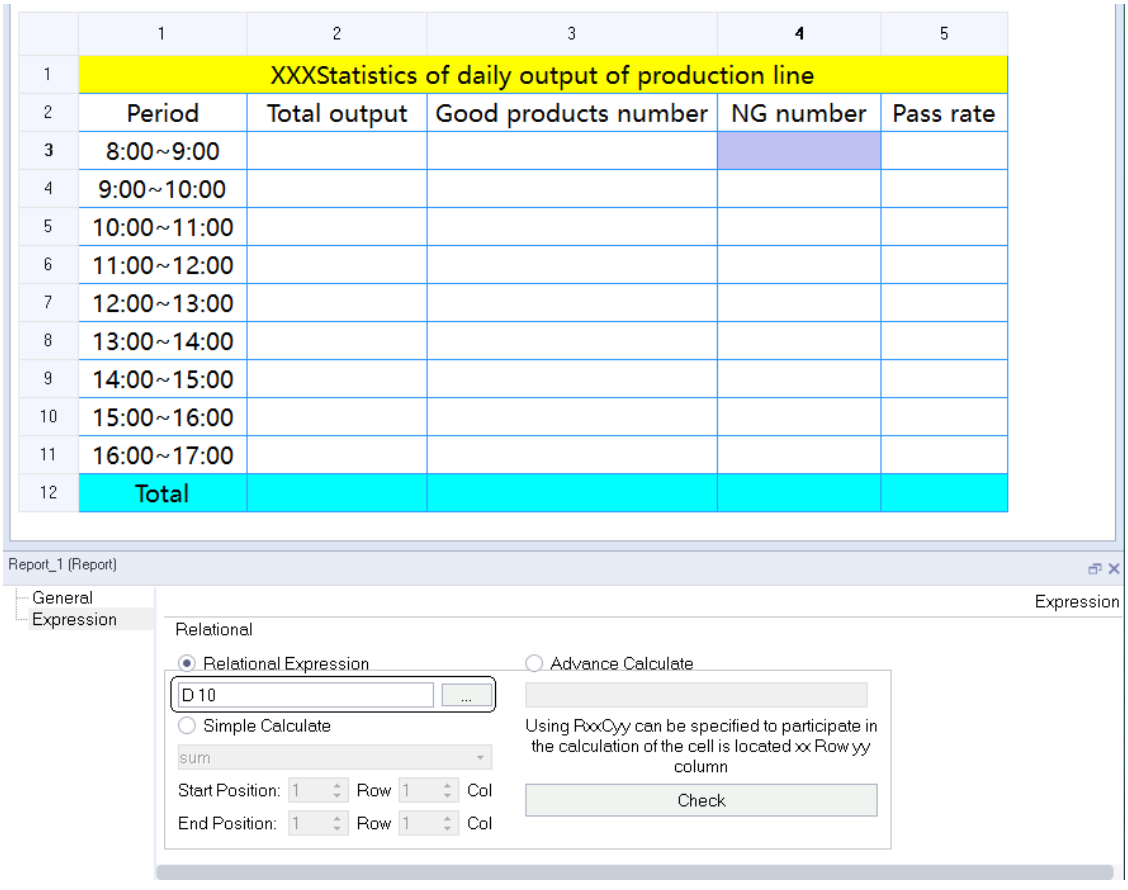




2. In the report, the period cells in the "Good Product of the Day" column and the "NG Product of the Day" column are associated with these statistical tags to display real-time data. For details, see the following example in which the period cells of 8:00–9:00 are associated with tags:



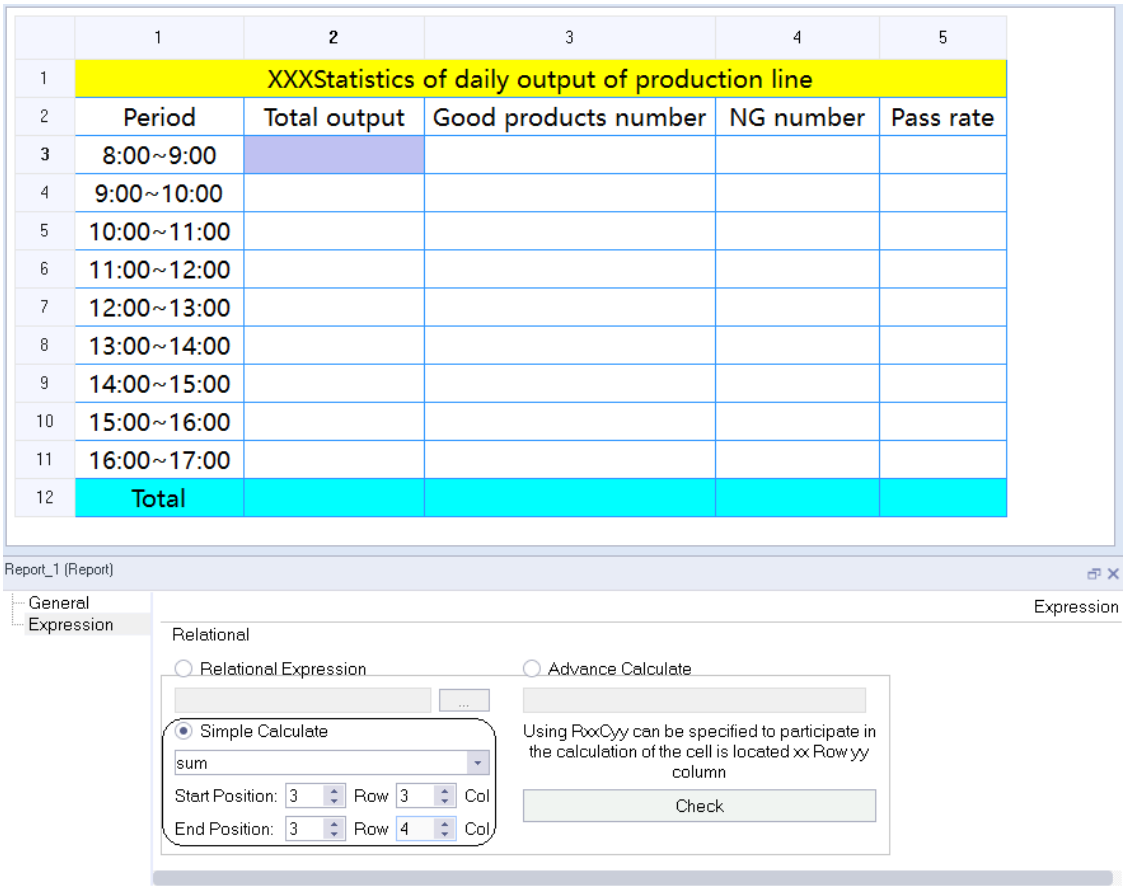
	1	2	3	4	5
1	XXXStatistics of daily output of production line				
2	Period	Total output	Good products number	NG number	Pass rate
3	8:00~9:00				
4	9:00~10:00				
5	10:00~11:00				
6	11:00~12:00				
7	12:00~13:00				
8	13:00~14:00				
9	14:00~15:00				
10	15:00~16:00				
11	16:00~17:00				
12	Total				

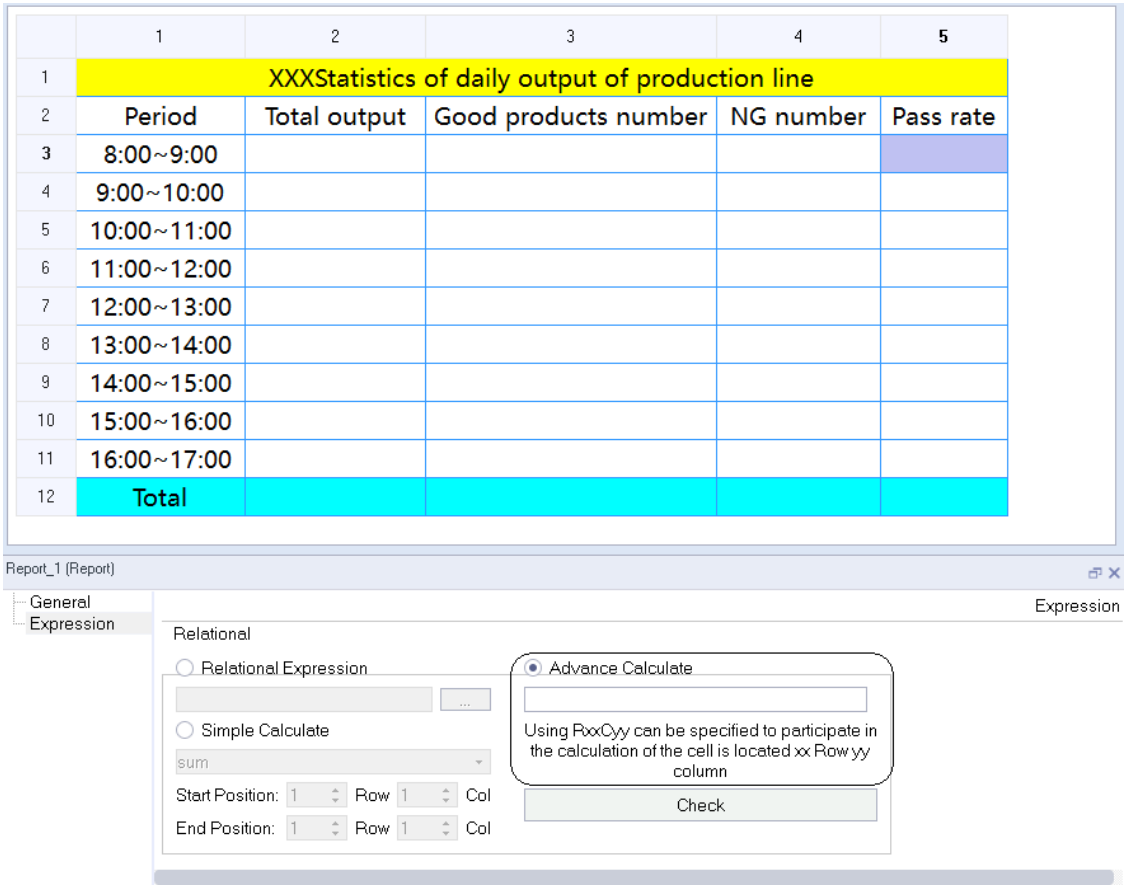


	1	2	3	4	5
1	XXXStatistics of daily output of production line				
2	Period	Total output	Good products number	NG number	Pass rate
3	8:00~9:00				
4	9:00~10:00				
5	10:00~11:00				
6	11:00~12:00				
7	12:00~13:00				
8	13:00~14:00				
9	14:00~15:00				
10	15:00~16:00				
11	16:00~17:00				
12	Total				

The association of other period cells is the same.

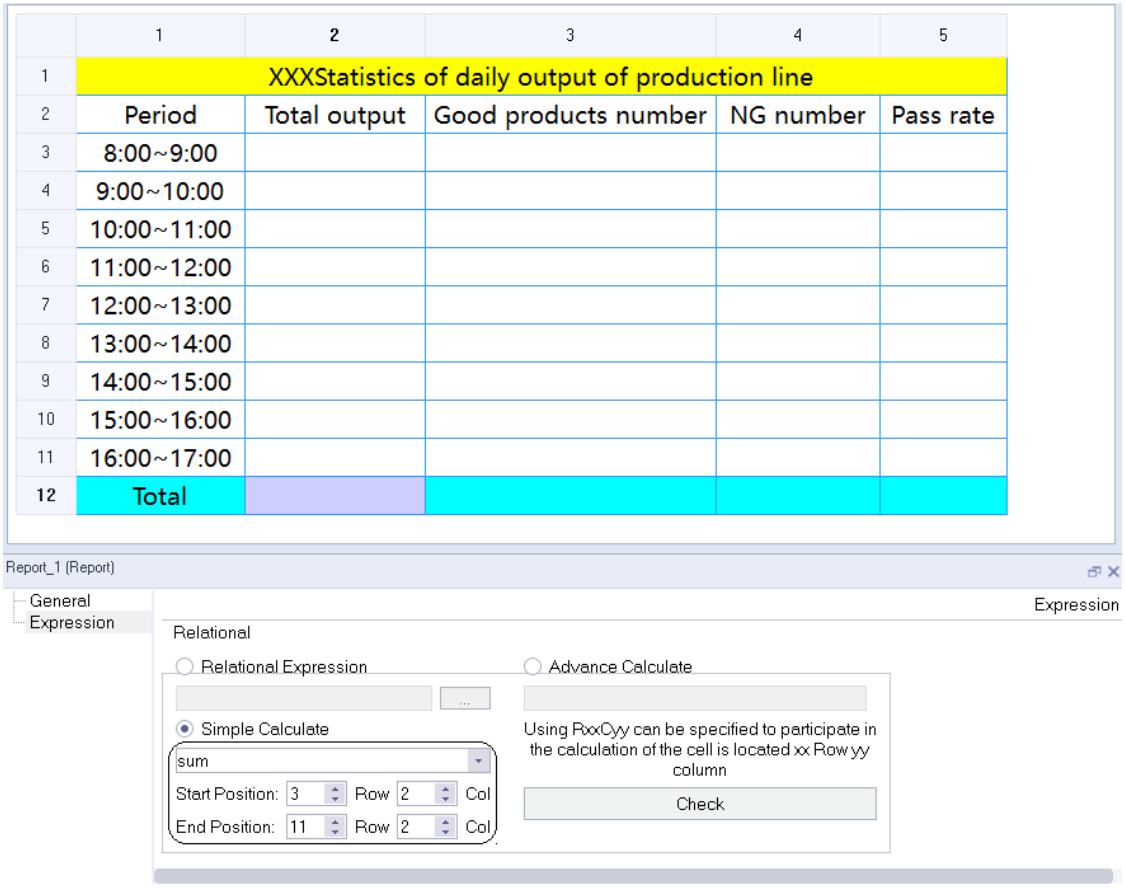
3. Set the calculation mode and expression for the cells in the "Total Output of the Day" column and "Qualified Rate of the Day" column respectively, as shown in the following figure:





The setting also applies to other period cells.

4. Set the calculation method in each cell of the Total row, as shown in the following figure:



This applies to all Total row cells.

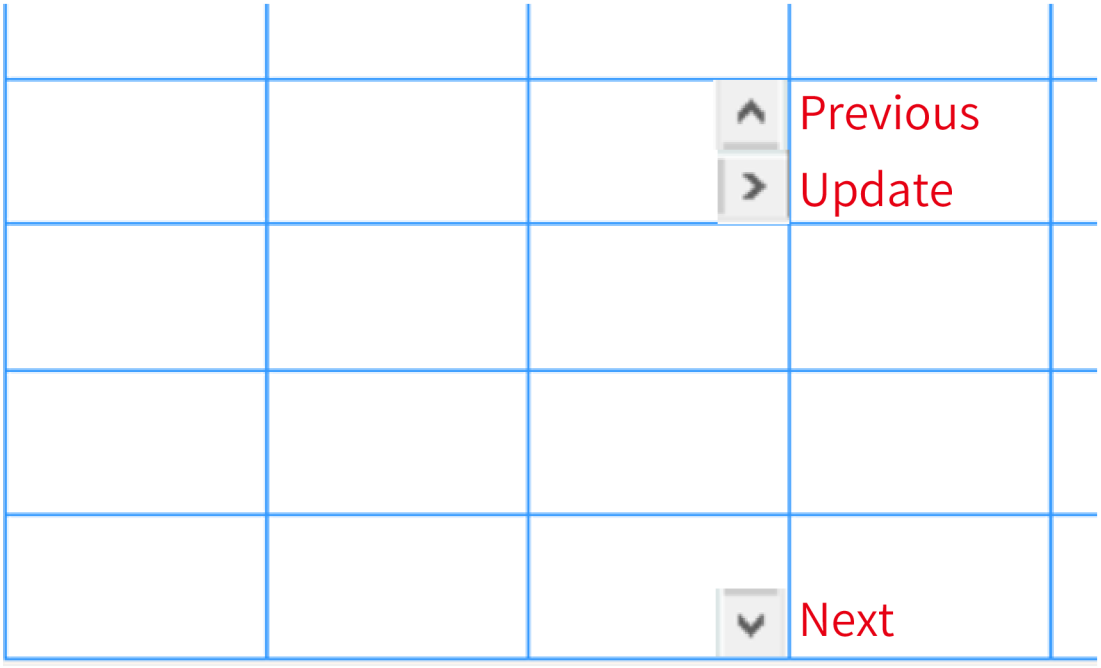
Now, you can configure the statistical report with a report view on the screen. In the runtime, the number of good products and NG products is transmitted by tags, as shown in the following figure:

	1	2	3	4	5
1	XXXStatistics of daily output of production line				
2	Period	Total output	Good products number	NG number	Pass rate
3	8:00~9:00	11	10	1	90.9
4	9:00~10:00	11	9	2	81.81
5	10:00~11:00	10	10	0	100
6	11:00~12:00	13	12	1	92.3
7	12:00~13:00	16	13	3	81.25
8	13:00~14:00	12	12	0	100
9	14:00~15:00	10	9	1	90
10	15:00~16:00	16	10	6	62.5
11	16:00~17:00	9	9	0	100
12	Total	108	94	14	88.75

15.6 Features of the Runtime

After you complete the configuration described in *"15.5 Examples" on page 347*, run the project, and you can see the following properties of the Report view:

- **Display static data:** It can display the contents of table elements (table cells) edited by users in configuration environment. This function is generally used to complete the header of a report or other fixed contents, and is effective only when the table elements are not connected with tags and data sources.
- **Edit data in the runtime environment:** The data of the table element can be edited in the runtime environment and the edited results can be output to the corresponding tags. This function is generally used to manually modify the current data of the report.
- **Display dynamic data:** It connects related tags in table elements, and dynamically displays the values of real-time tags at runtime.
- **Display history:** It connects the record tag in the database in the table element, dynamically displays the value of the saved records in the saved data source at runtime. It can display multiple pages and display the fields in the historical data table by row or column.
- **Page navigation buttons:** If you click a multipage-enabled historical report, the following buttons show:



- **Page-up button**: Click this to show the previous history record.
- **Refresh start-stop button:** Click this to refresh the history automatically. When the refresh time is set to 0, clicking this button refreshes the history only once.
- **Page-down button**: Click this to show the next history record. Automatic refresh will stop when you use this feature.
- **Hide function button:** When you click other positions on the screen, it will be automatically hidden.
- **Display statistical results of historical records:** There are two ways to display the statistical results. One is to count the data of other real-time table elements in the table, such as the total of the table. The other is the statistics of the records in the historical database, which connects the

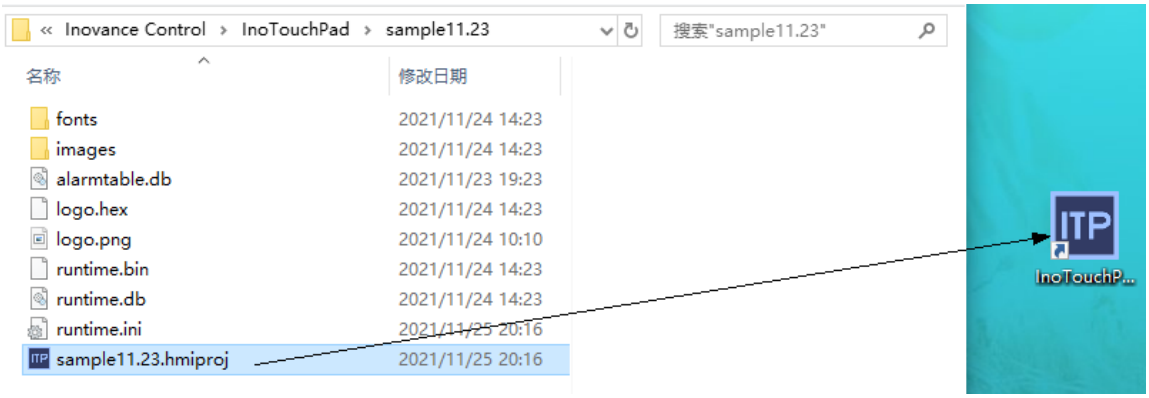
saved data source in the table element, and dynamically displays the statistical results of the saved records in the saved data source at runtime.

16 Typical Functions

16.1 Drag and Drop

16.1.1 Screen Drag-and-drop

To open an existing project, you can directly double-click the *. hmiproj project file or find and open the project in the software tool. In addition to that, you can also drag the project file directly to the software icon to open the project, as shown in the following figure:

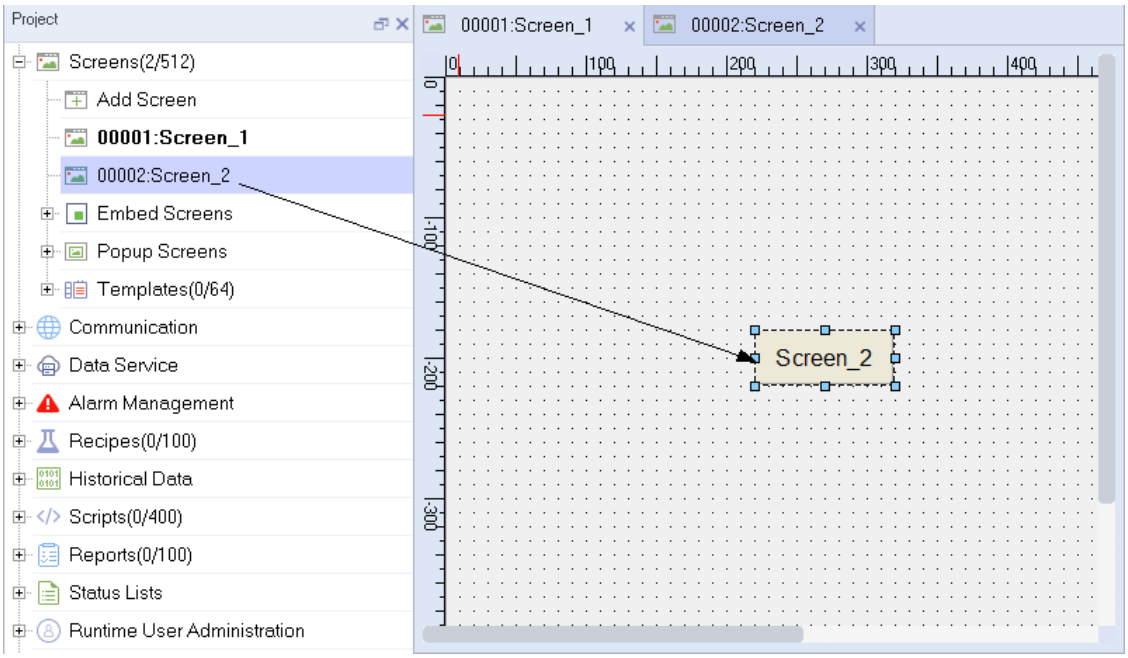


During project configuration, you often need to use a button to jump to a screen, which can be achieved through the following steps:

1. Add a button control to the current screen.
2. Rename the button with the name of the screen that you want to jump to.
3. Bind the screen jump function to the button.

   Then you can jump to the designated screen by clicking the button.

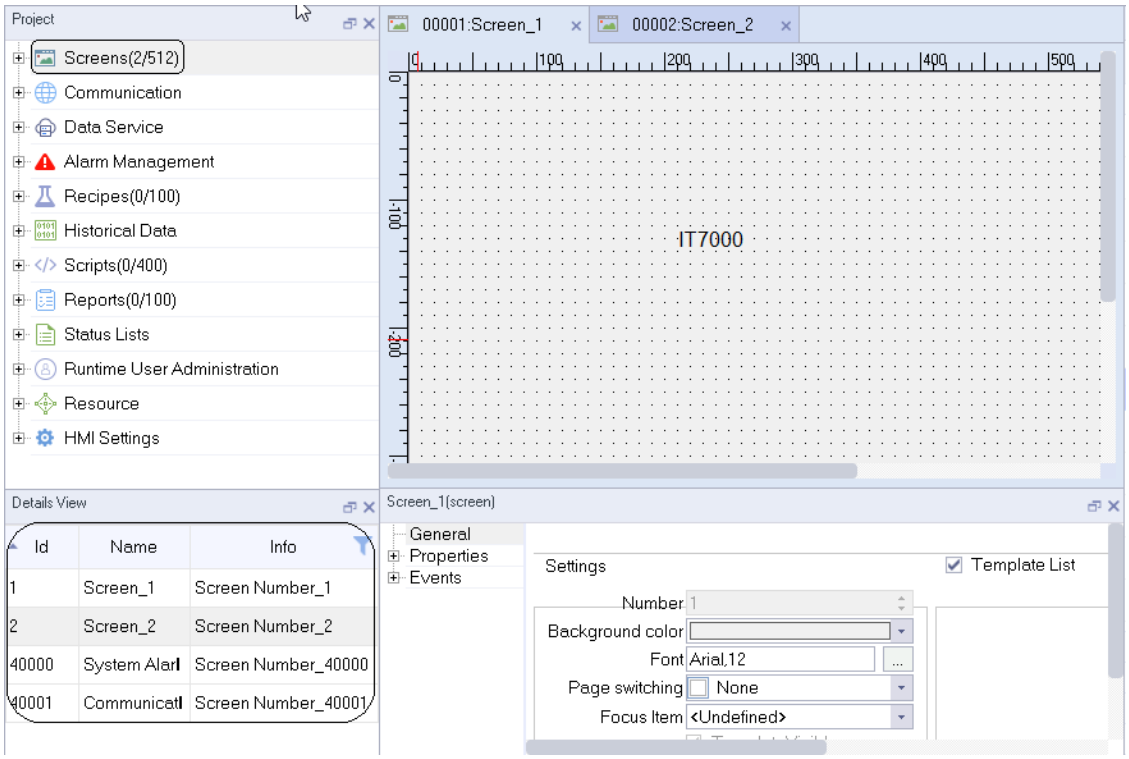
   However, a quick way to achieve this is using screen drag-and-drop, as shown in the following figure:



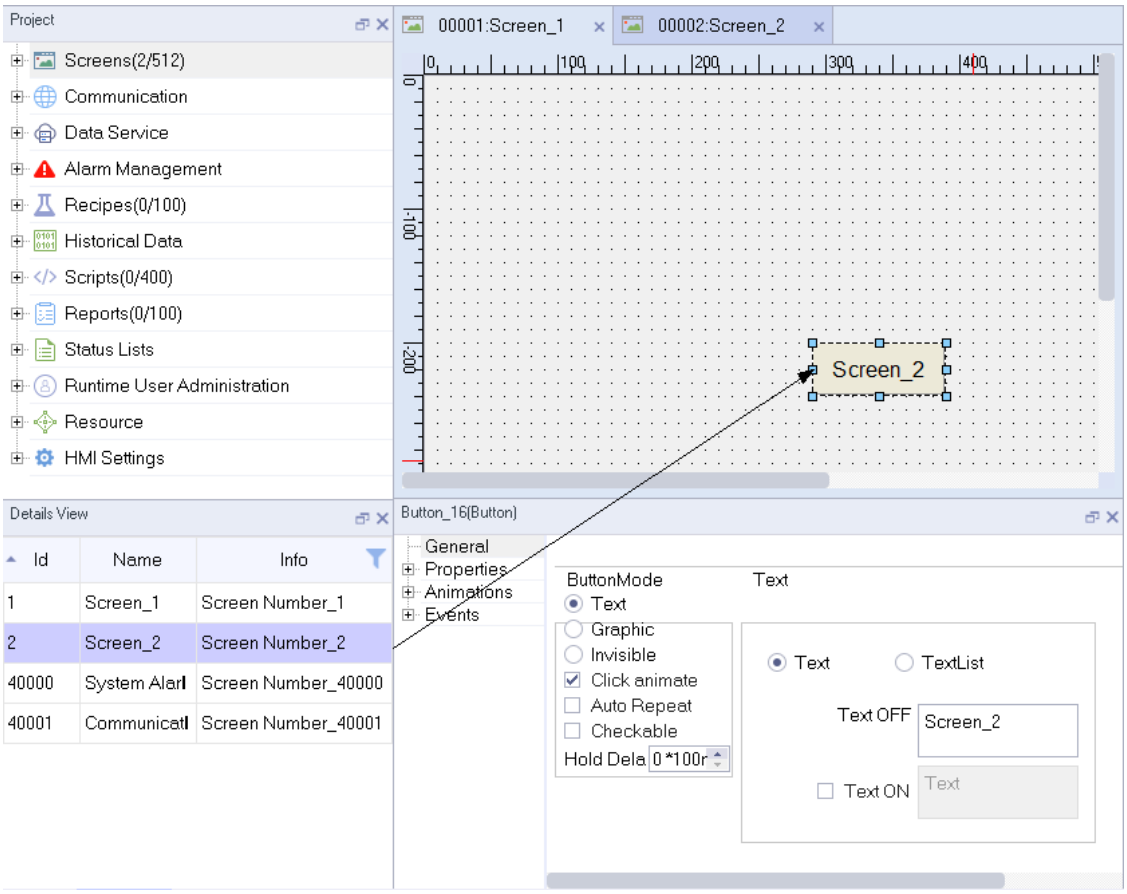
Screens supporting drag-and-drop: Common screens and popup screens.

In addition, you can also drag and drop the screen to be jumped to directly into the screen in the Details view. The steps are as follows:

In the Project view, select Screens and the Details view will show all the created common and popup screens, as shown in the following figure:



In the Details view, directly drag the screen you want to jump to into the current screen to automatically generate the jump button, as shown in the following figure:



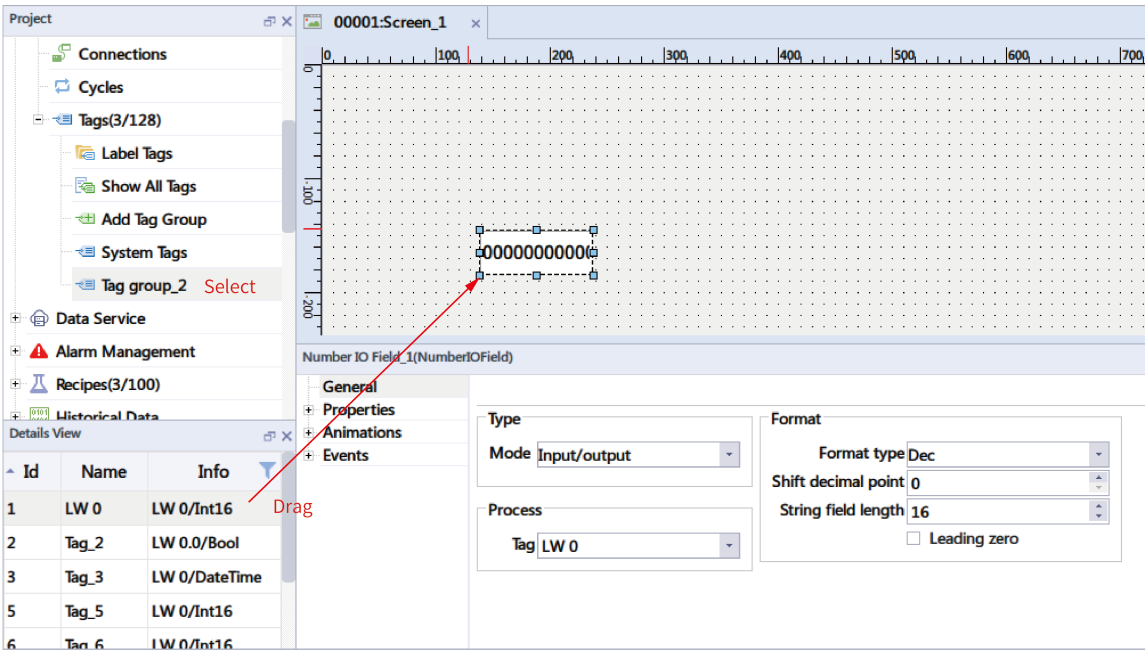
16.1.2 Dragging and Dropping a Tag

During project configuration, you often need to use IO domain to read and write tag values. You can achieve that through the following steps:

1. Add an IO field to the current screen.
2. In the properties bar of the IO field, select the specified process tag.

Then you are able to read and write specified project tags with IO domain.

However, a quick way to achieve this is using tag drag-and-drop, as shown in the following figure:



All tags except array tags allow drag-and-drop.

The detailed view of tag variables is slightly different from that of ordinary variables. See *"5.3.9 Using Tags" on page 73* for details.

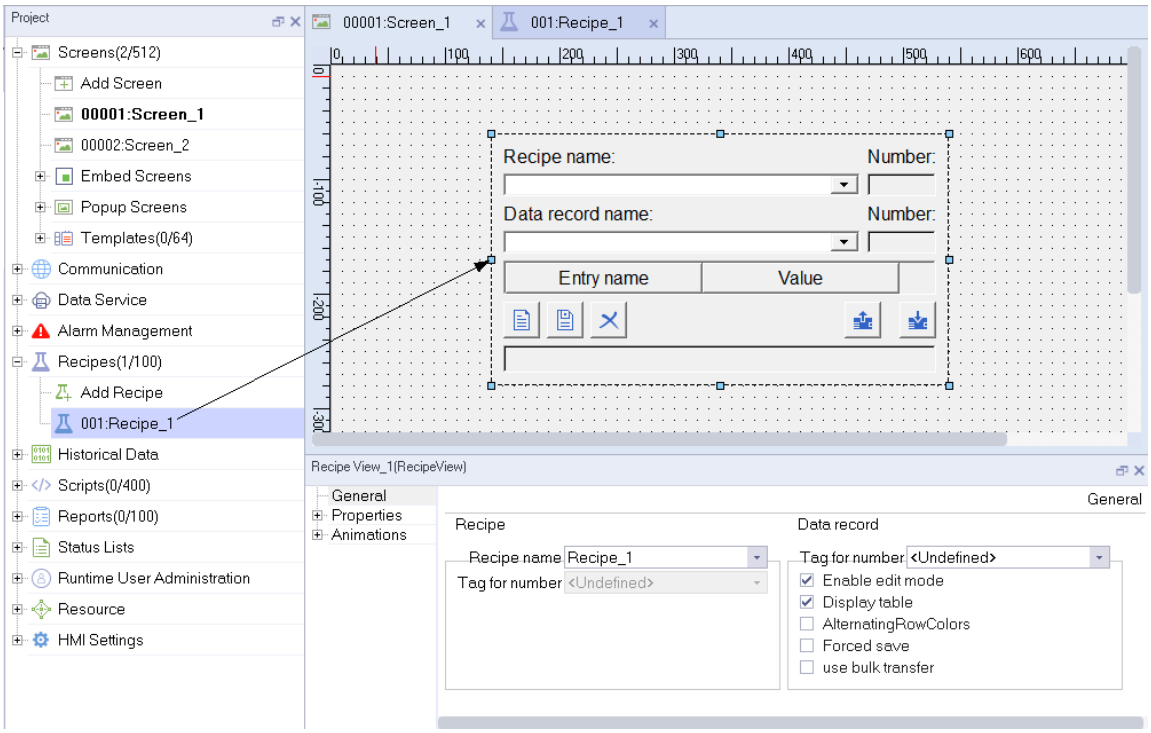
16.1.3 Dragging and Dropping Recipes

When you configure a project, you can display a recipe in the recipe view in the following steps:

1. Add a recipe view to the screen.
2. In the General property setting page of the recipe view, select the recipe you want to display under "Recipe name".

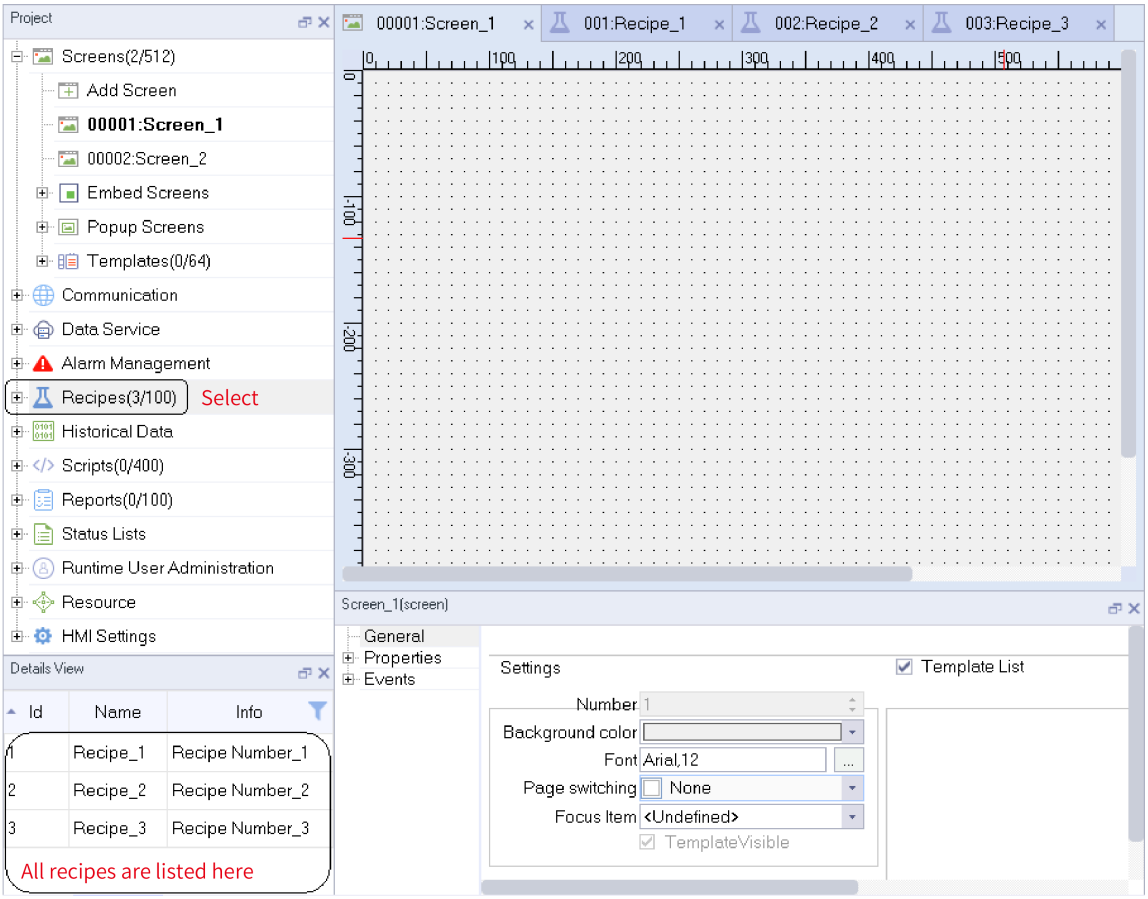
Then the specified recipe is displayed in the recipe view.

However, a quick way to achieve this is using drag-and-drop, as shown in the following figure:

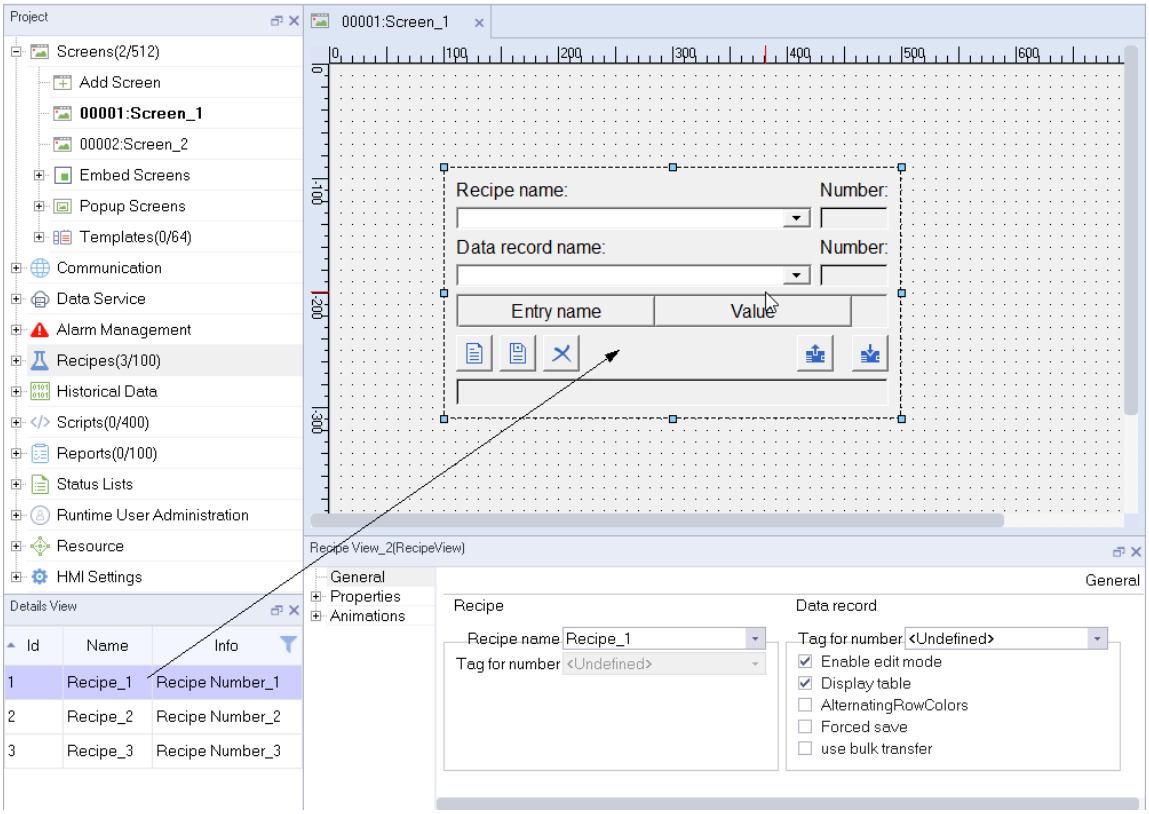


In addition, you can also drag and drop a recipe in the detailed view directly into the screen. The steps are as follows:

First, in the project tree on the left, expand Recipes to list all the created recipes in the detailed view, as shown in the following figure:



Then, in the Detailed view, directly drag a recipe to the screen to generate a recipe view, as shown in the following figure:

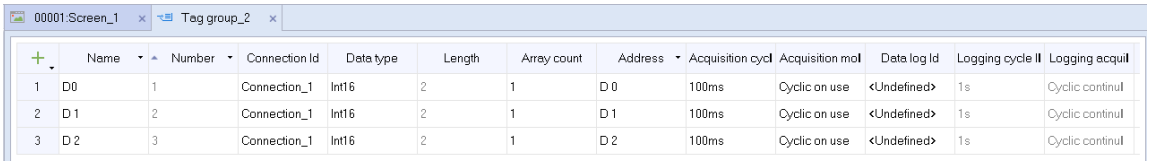


In the recipe editor, you can add elements as follows:

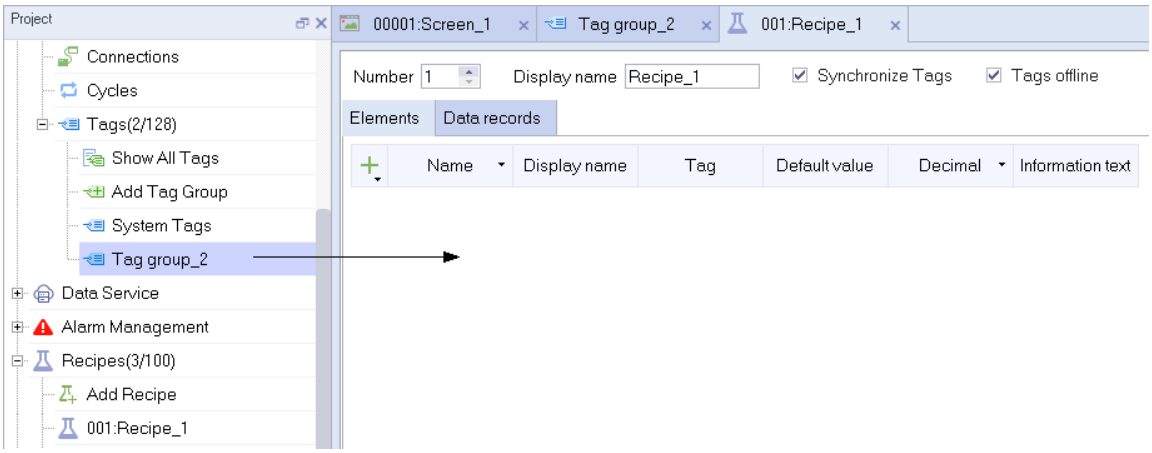
1. In the recipe editor, click the Add button to add an element
2. Then, you can set the element tag in the Tag field.

When there are lots of elements, the operation will be cumbersome. InoTouchPad provides a a quick way, with which you can drag the specified tag group to the recipe element editor to automatically create elements. The tags in the tag group are also added into the Tag field in sequence. See the following example:

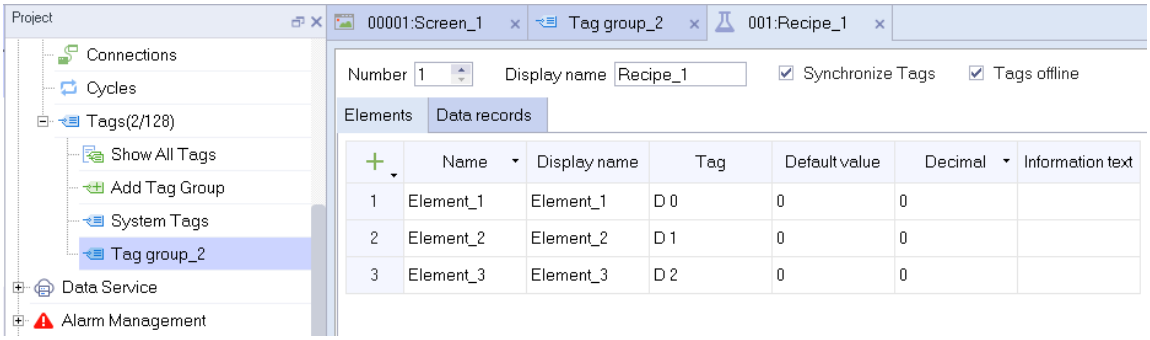
First, create three tags in tag group_2: an array tag and two common tags, as shown in the following figure:



Then open recipe_1 editor and drag Tag group_2 in the tree view to the editor, as shown in the following figure:



Three elements are automatically created, and the tags in the tag group are bound with these elements in sequence, as shown in the following figure:



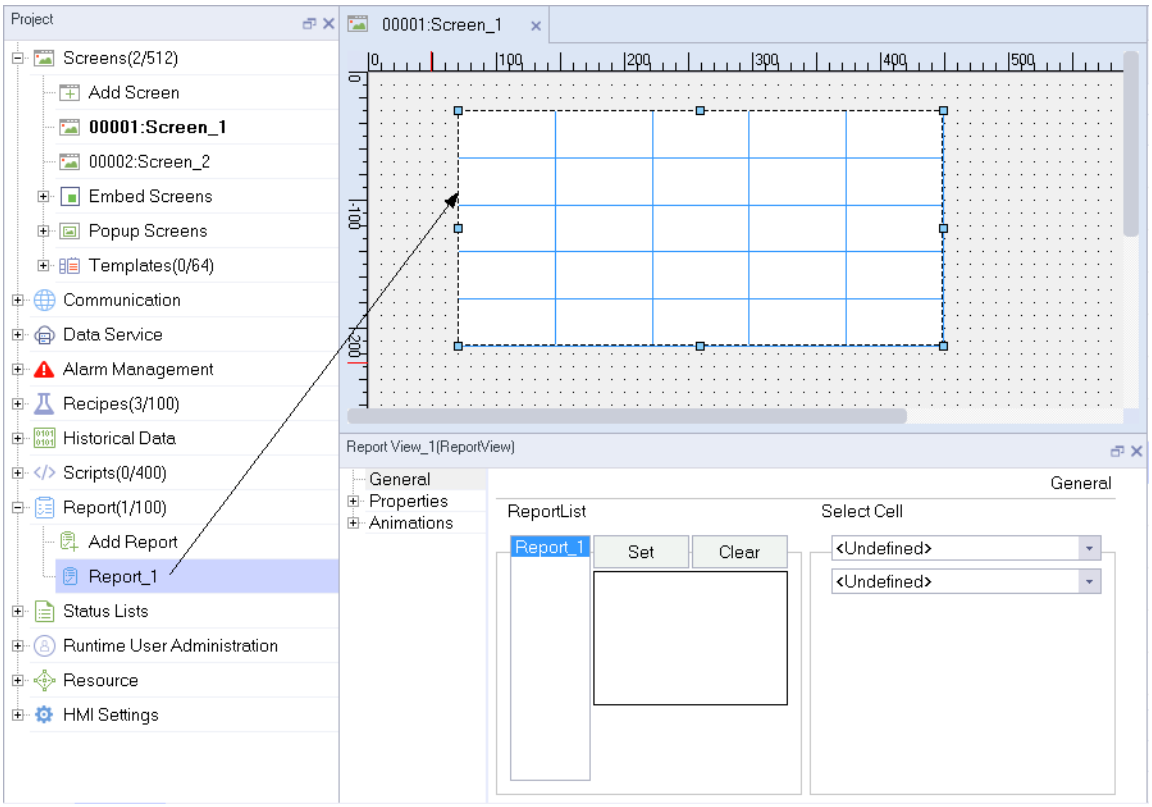
16.1.4 Dragging and Dropping the Report

When you configure a project, you can display a report in the report view in the following steps:

1. Add a report view to the screen.
2. In the Properties bar of the report view, locate the report list and select the report you want to display.

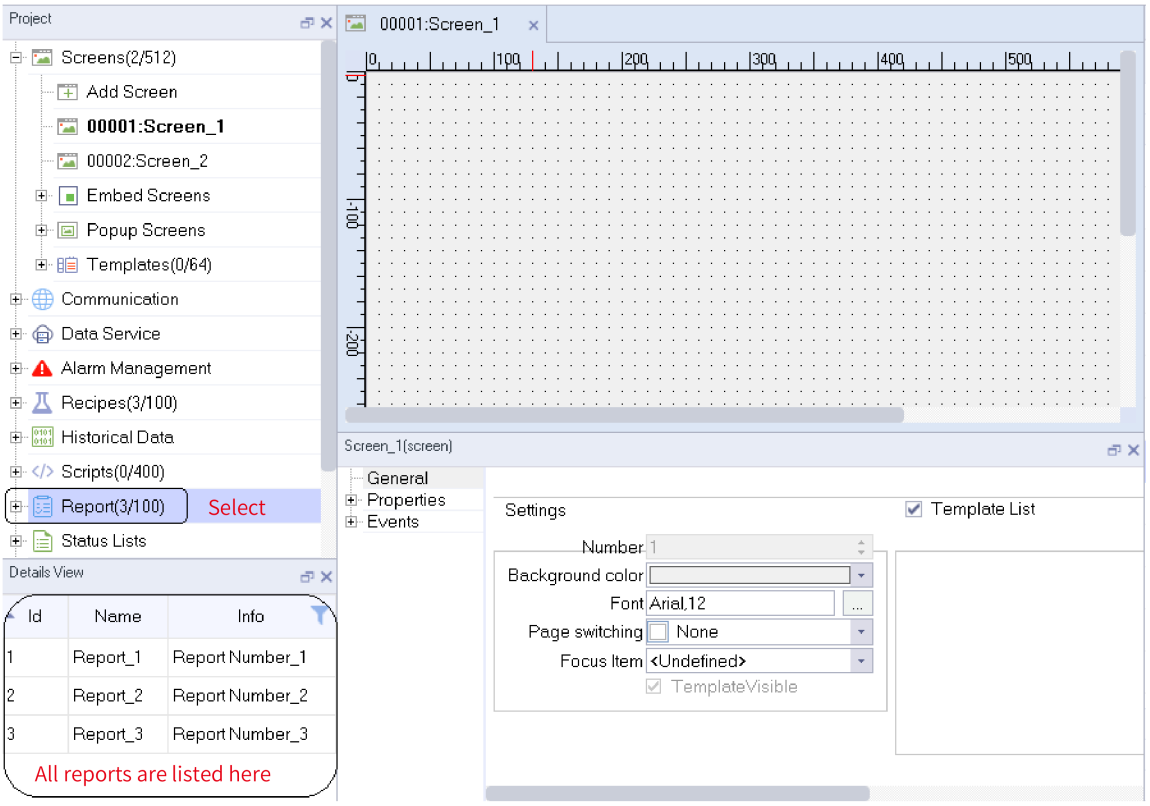
Then the specified report is displayed in the report view.

However, a quick way to achieve this is using report drag-and-drop, as shown in the following figure:

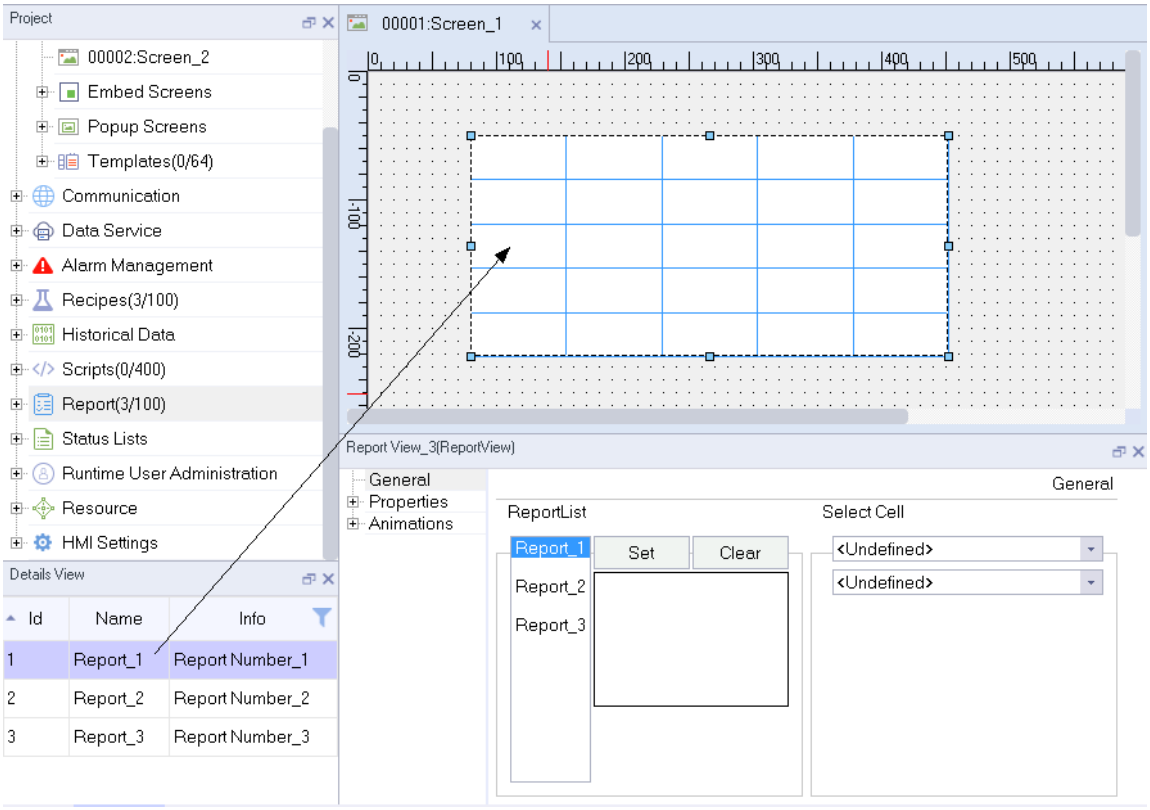


In addition, you can also drag and drop the report to be displayed directly into the screen in the detailed view. The steps are as follows:

First, in the project tree on the left, select the root node of the report module to list all the created reports in the detailed view, as shown in the following figure:

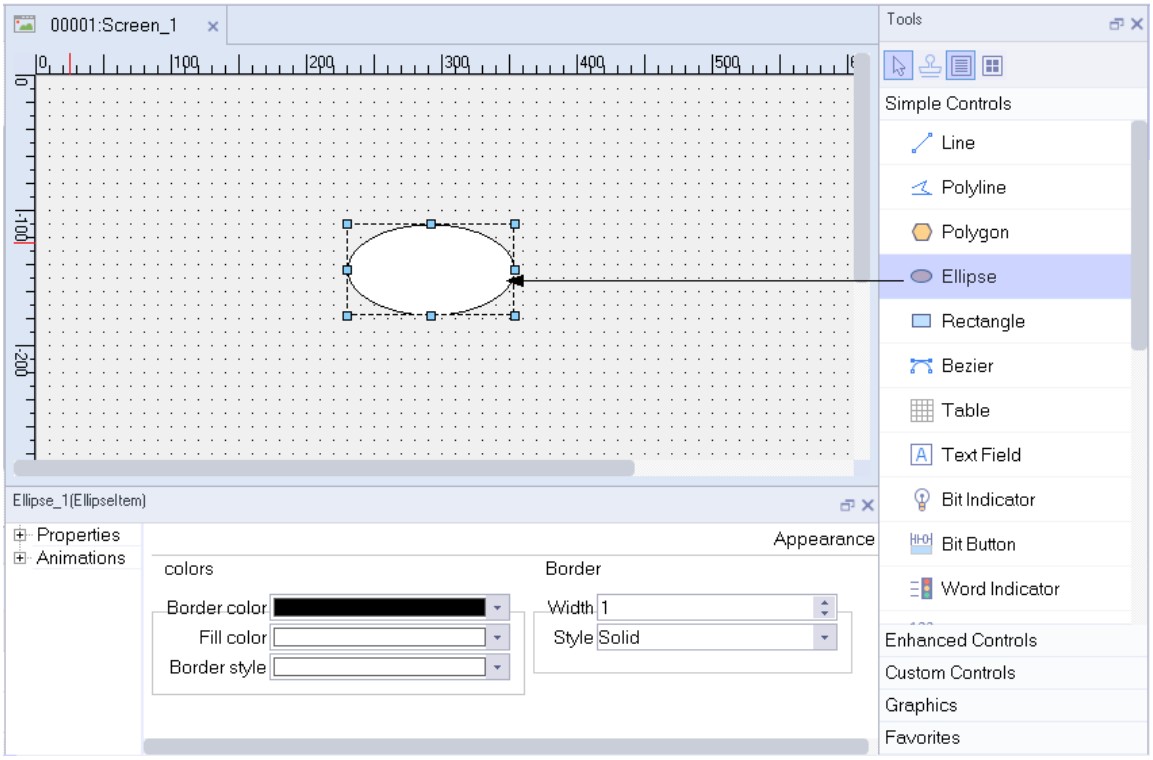


Then, in the detailed view, directly drag the report to be displayed to the screen to generate a report view displaying the report in the screen, as shown in the following figure:



16.1.5 Dragging a Control

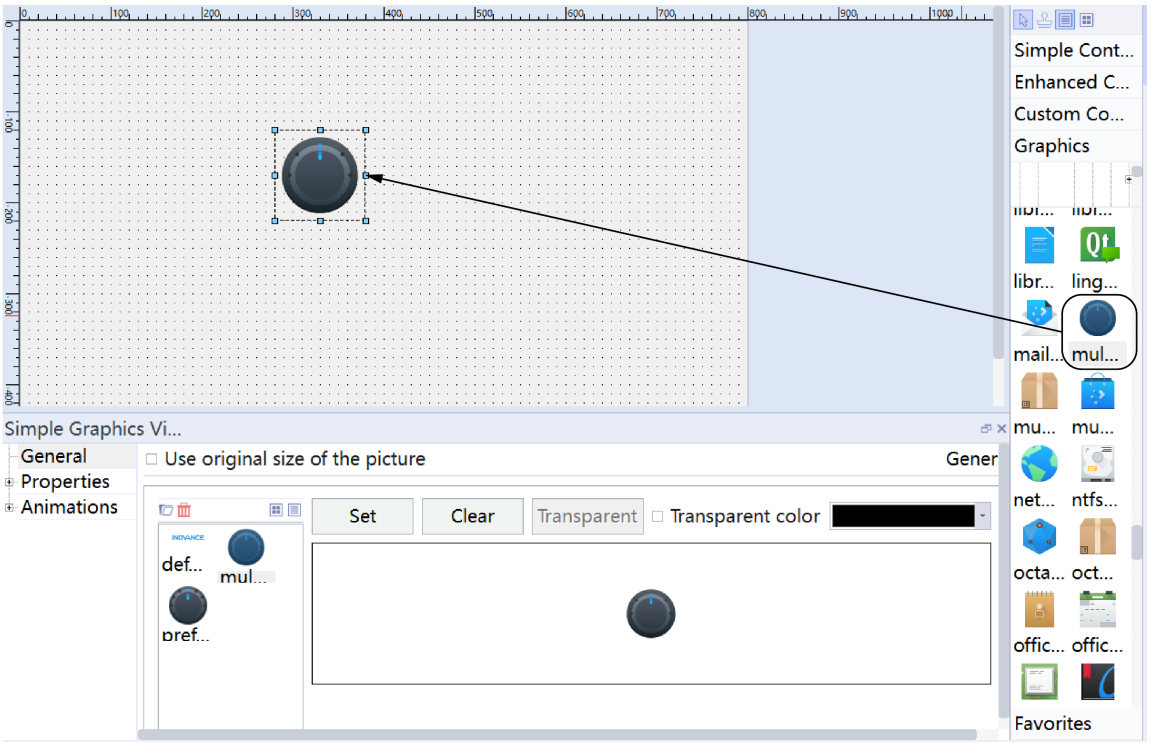
When configuring a project, you can add a control by selecting the control first, and then click on the screen. You can also add a control through drag and drop, as shown in the following figure:



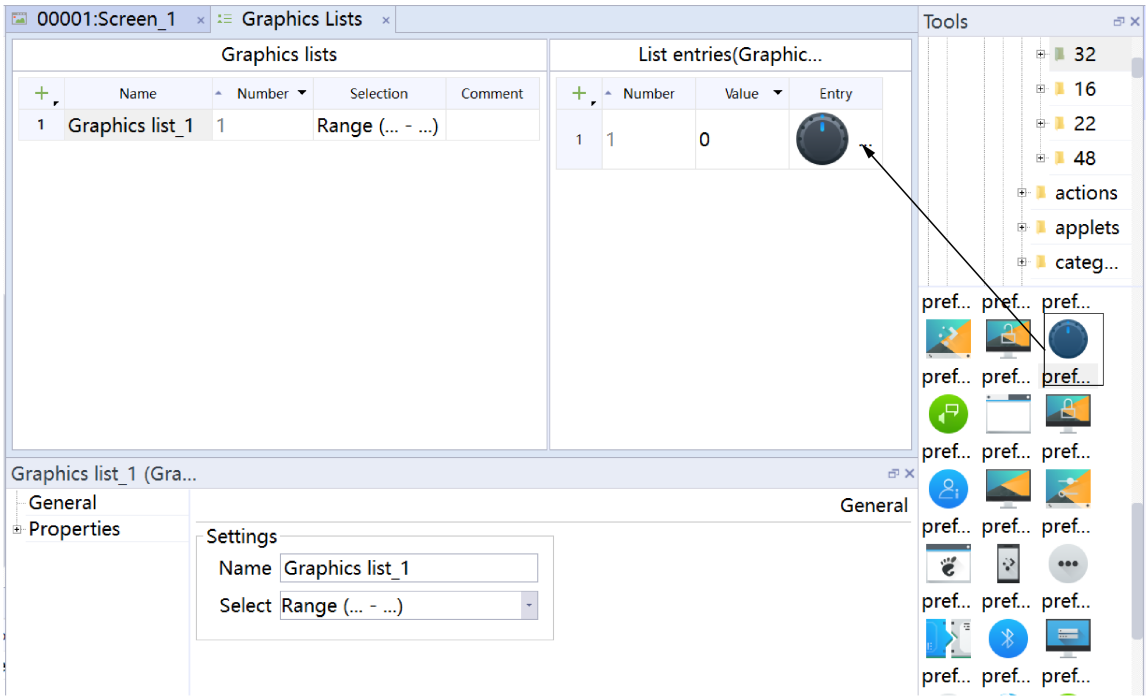
The drag-and-drop feature applies to all controls, including simple controls, enhanced controls, and favorite controls.

16.1.6 Dragging an Image

In the gallery, you can select any picture and drag it directly to the screen to add a graphical view control configured with the picture, as shown in the following figure:



In addition, you can also drag and drop pictures directly into the list entry of the graphic list to add pictures, as shown in the following figure:



16.1.7 Dragging-and-dropping Historical Data

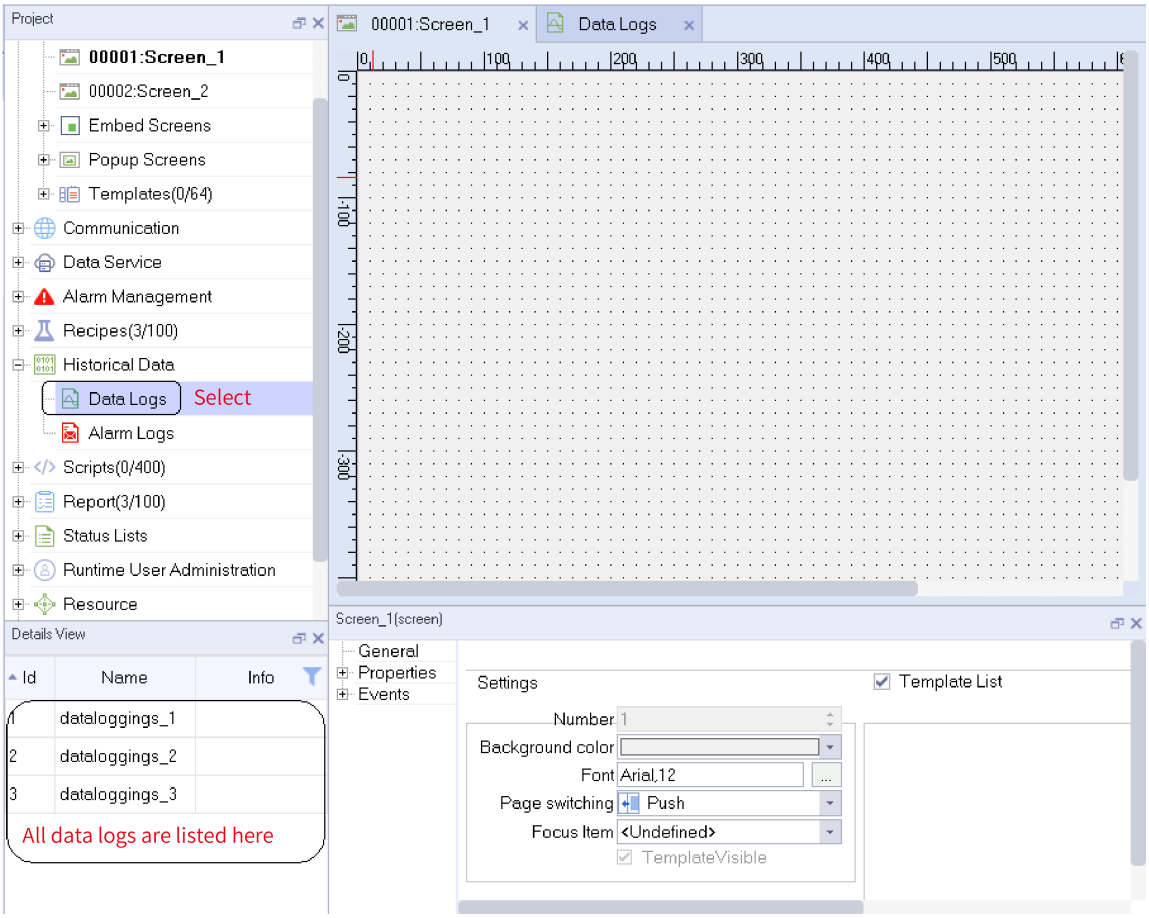
When you configure a project, you can display a data log in the data view in the following steps:

1. Add a data view to the screen.
2. In the general settings of the data view, select the data log you want to display.

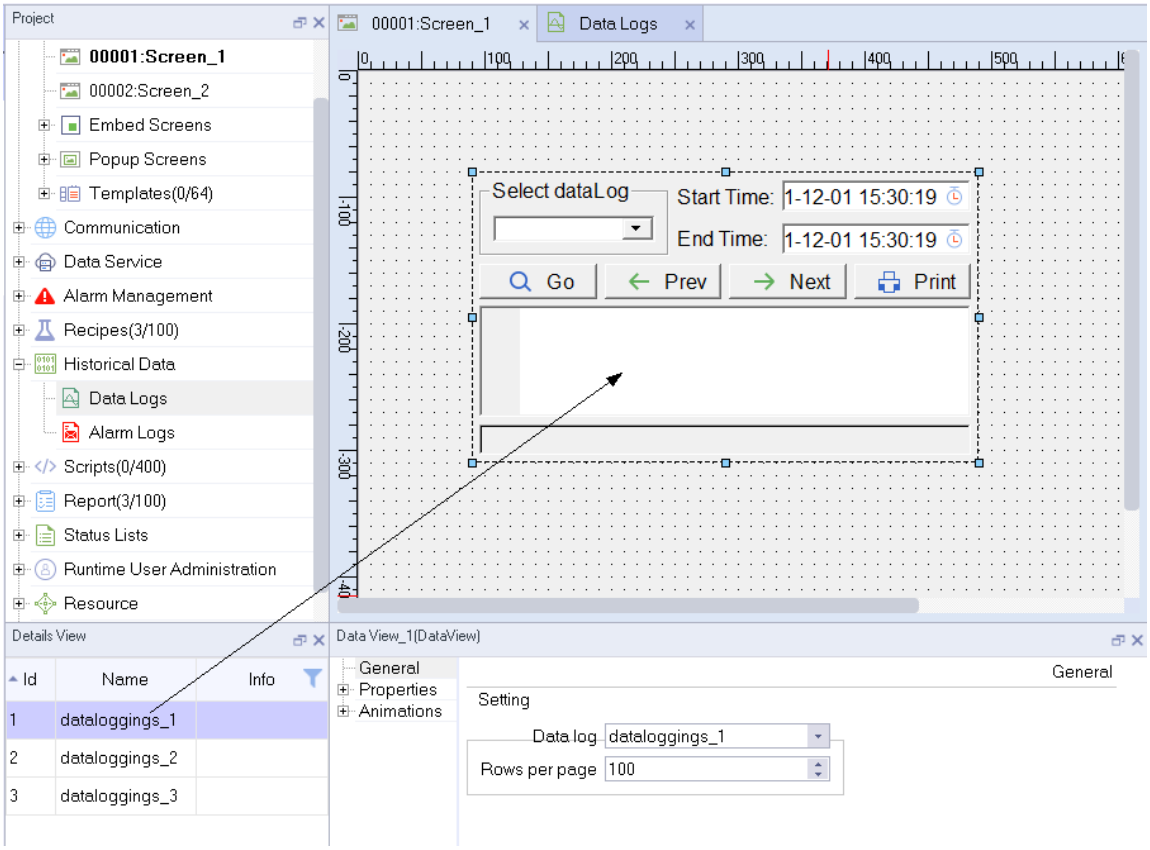
Then the specified data log is displayed in the data view.

However, a quick way to achieve this is using data log drag-and-drop in the details view.

First, in the screen editing view, select Data Logs in Historical Data to show all created data logs the details view:



Then, in the details view, directly drag the data log to be displayed to the screen to generate a data view displaying the data log in the screen, as shown in the following figure:



Likewise, you can also using drag-and-drop to display a alarm log in an alarm view.

16.1.8 Dragging and Dropping the Status List

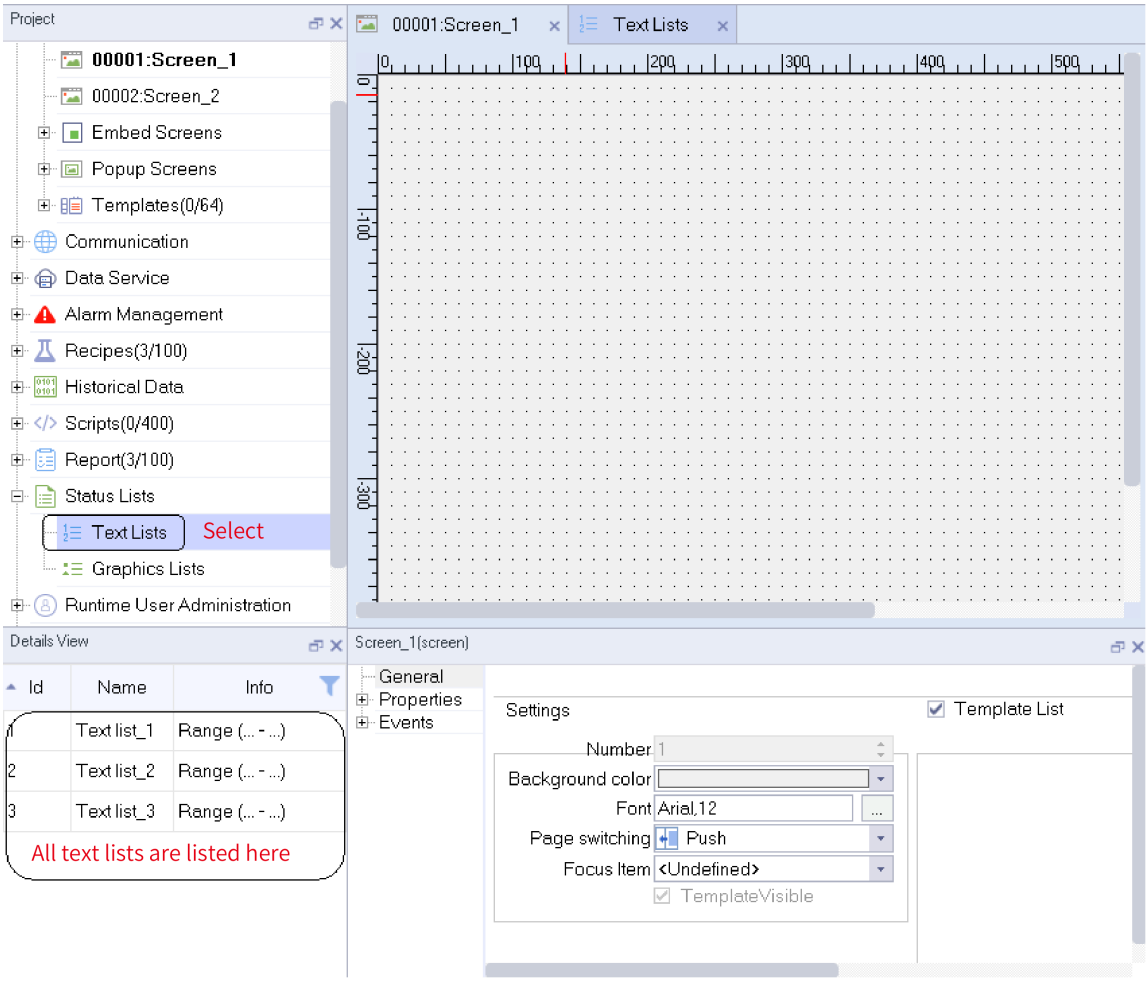
When you configure a project, you can display a text list in a symbolic IO field in the following steps:

1. Add a symbolic IO field to the screen.
2. In the General settings, select the text list you want to display under Display.

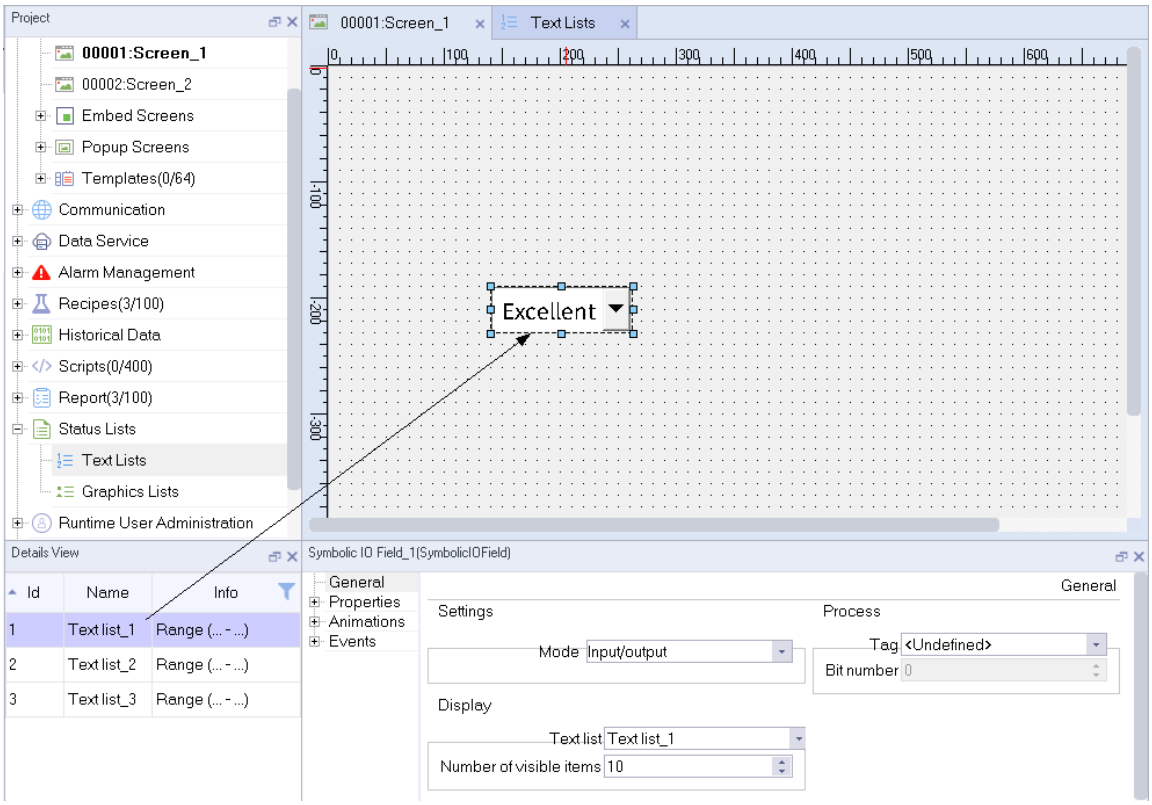
After these two steps, the symbol IO field can be bound to a text list.

However, a quick way to achieve this is using text list drag-and-drop in the details view.

First, in the screen editing view, select Text Lists in Status Lists to show all created text lists in the details view:



Then, directly drag the text list to be displayed to the screen to generate a symbolic IO field in the screen, as shown in the following figure:



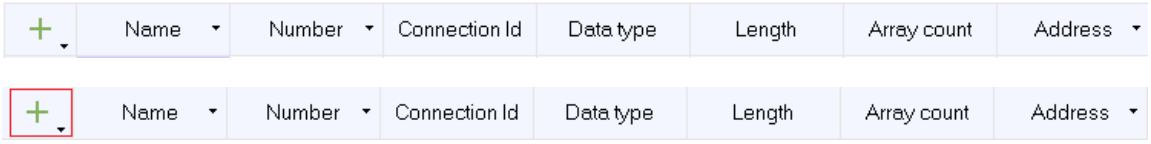
Likewise, you can also using drag-and-drop to display a graphics list in a grapic IO field.

16.2 Table operations

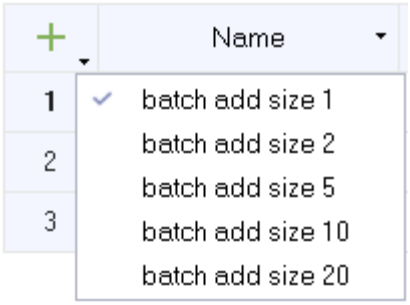
16.2.1 Add

Most modules of InoTouchPad, such as connection, tag, and alarm, are edited in a table. Some common operations of the table include add, delete, hide, and sort, which are described in detail in this section.

For tables where rows can be added, such as tables in tag groups, you can use the Add icon + in the upper left corner of the table to add data.



You can click on the arrow icon in the lower right corner of the Add icon to set how many rows the Add icon can add at one time. The default is one row:



After you select, for example, batch add size 5, and click +, five rows will be added at one time.

16.2.2 Delete

You can delete data from a table.

Select the row where the data to be deleted is located.

—	Name	Number	Connection Id	Data type	Length	Array count	Address
1	LW 0	5	<Internal tag>	Int16	2	1	LW 0
2	LW1	6	<Internal tag>	Int16	2	1	LW 1

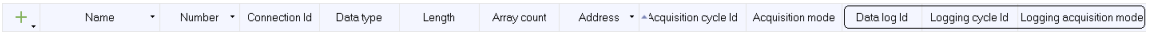
The Add icon + becomes — the Delete icon. Click the Delete icon to delete that row. You can also delete the row by pressing the Delete key.

If you need to delete multiple rows at a time, select the rows, and then click the Delete icon — or press the Delete key.

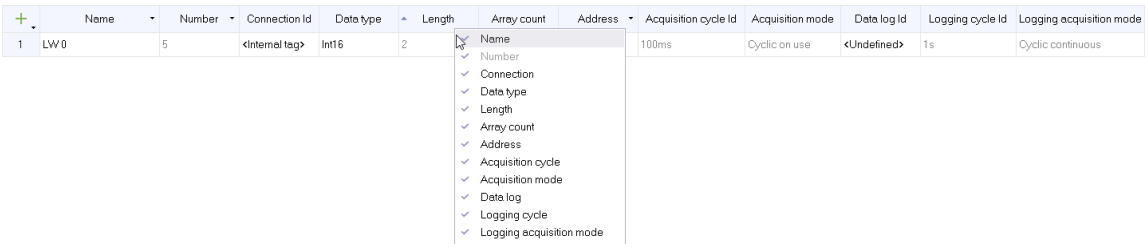
If you need to delete all the data in the table, press Ctrl+A to select all the rows, and then click the Delete icon — or press the Delete key.

16.2.3 Hiding a Column

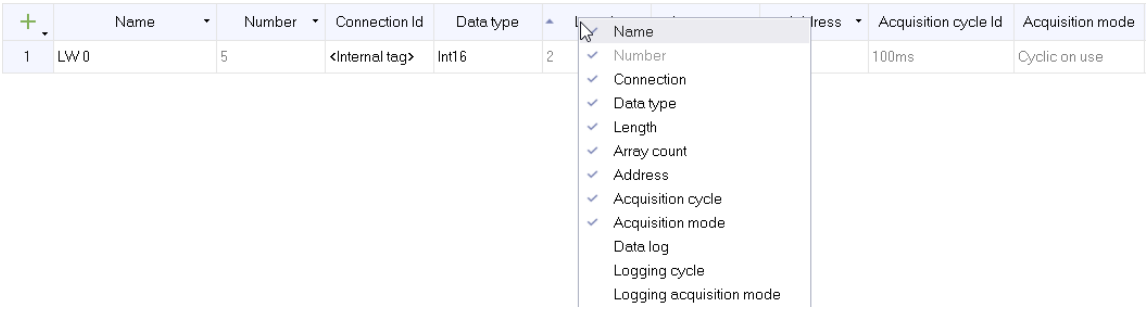
A table typically has multiple fields, corresponding to some attributes of a module. Since some of these attributes are optionally editable, If you need not to use them, you can hide them. For example, in the tag table, for the internal tag LW 0, no data logs are used, then Data log , Logging cycle and Logging acquisition mode can be hidden.



Right-click the column header as shown in the following figure:

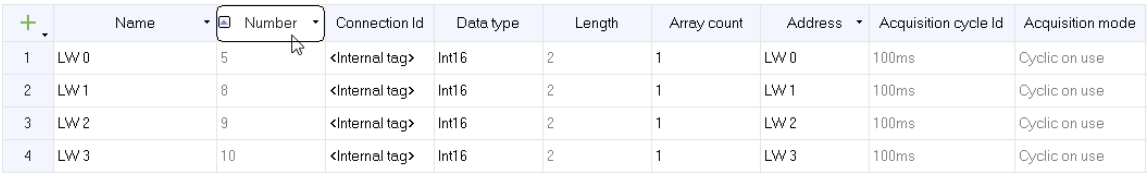


Uncheck Data log, Logging cycle and Logging acquisition mode to hide them, as shown in the following figure:



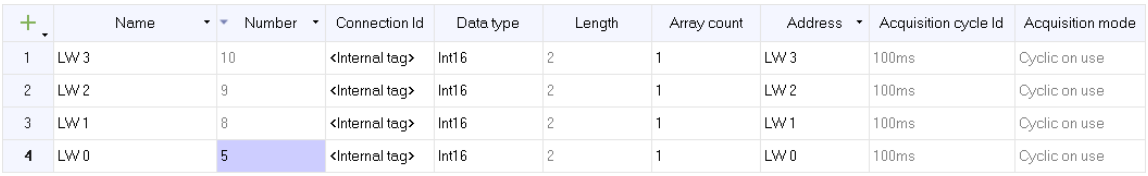
16.2.4 Sorting a Column

For fields that can be sorted, when you click the header of the list, a small sorting icon will appear on the leftmost side of the header. For example, in the variable table, click the header of the numbered list, and the sorting icon on the left side will appear.



	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycle Id	Acquisition mode
1	LW 0	5	<Internal tag>	Int16	2	1	LW 0	100ms	Cyclic on use
2	LW 1	8	<Internal tag>	Int16	2	1	LW 1	100ms	Cyclic on use
3	LW 2	9	<Internal tag>	Int16	2	1	LW 2	100ms	Cyclic on use
4	LW 3	10	<Internal tag>	Int16	2	1	LW 3	100ms	Cyclic on use

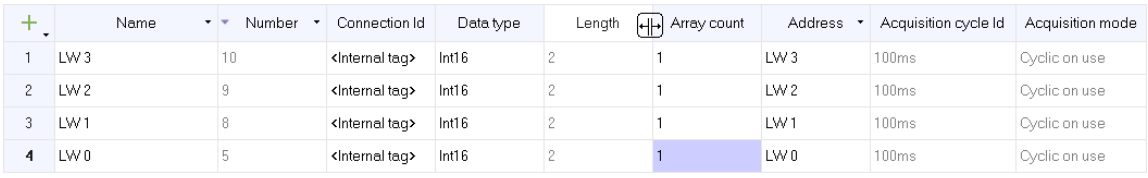
Click the sort icon, and the whole table will be sorted according to the column data. The sorted table is as follows:



	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycle Id	Acquisition mode
1	LW 3	10	<Internal tag>	Int16	2	1	LW 3	100ms	Cyclic on use
2	LW 2	9	<Internal tag>	Int16	2	1	LW 2	100ms	Cyclic on use
3	LW 1	8	<Internal tag>	Int16	2	1	LW 1	100ms	Cyclic on use
4	LW 0	5	<Internal tag>	Int16	2	1	LW 0	100ms	Cyclic on use

16.2.5 Column Width Adjustment

For tables that allow adjusting column width, you can adjust the size by dragging the borders of columns with the mouse. For example, in the variable group table, you can adjust column width as shown in the following figure:



	Name	Number	Connection Id	Data type	Length	Array count	Address	Acquisition cycle Id	Acquisition mode
1	LW 3	10	<Internal tag>	Int16	2	1	LW 3	100ms	Cyclic on use
2	LW 2	9	<Internal tag>	Int16	2	1	LW 2	100ms	Cyclic on use
3	LW 1	8	<Internal tag>	Int16	2	1	LW 1	100ms	Cyclic on use
4	LW 0	5	<Internal tag>	Int16	2	1	LW 0	100ms	Cyclic on use

16.3 Import and Export

16.3.1 Import and Export Rules

Import and export rules are mainly aimed at five modules: tags, text lists (including text lists and list entries), alarms (including analog alarms and discrete alarms), resource (i18n) and recipes (data records of recipes), which support import and export functions.

The general rule for importing is: for the tag, text list and recipe, when an identification name already exists, the corresponding row is overwritten, otherwise, a new row is added; For the alarm, because the alarm names can be the same, a new line is always added. When these four modules exceed the maximum number of rows that the list can hold, the extra rows are discarded (a tag list can accommodate 32,767 rows, a text list and list entry can accommodate 256 rows, and analog alarm and discrete alarm can accommodate 2000 rows). If a field is illegal, the row is filtered during import and a prompt about the discarded contents and filtered field is displayed; For the internationalization of the resource module, only the text is updated (that is, the items that exist in the "reference" column are updated, while the items that do not exist are directly filtered out, and a prompt about the discarded information is displayed).

The export and import of tags, text lists, alarms, internationalization, and recipes are as follows:

Tag

During export, the tag information is written to a *. csv file. The first 18 rows of the exported text are explanatory information and headers of the tag list, and the remaining rows are the exported data, which are in 13 columns: name, connection ID, data type, length, array count, address, acquisition cycle ID, acquisition mode, maximum, minimum, start value, comment and group Id. The fields are separated by ",". When a field is not empty, the contents are enclosed by double quotation marks. If the contents contain double quotation marks, the " will be escaped by another ", that is ""..."". (Note: the "," used for separating the fields and the double quotation marks enclosing the contents of the fields are all English characters).

During import, the first 18 lines of the import text are in a fixed format, and can not be added, deleted or changed. Only variable data can be edited. In the import file, each variable occupies an independent line, and each variable contains the following 13 fields which cannot be added or deleted:

Name: The tag name. If a name is empty, the line of data will not be imported. If the name exists in the project database, the imported data will overwrite the data in the database, otherwise, new data is added.

Connection: The name of the configured connection, which can be empty or the connection created in the current project (except for the connection with slave protocol including MODBUS slaves and MODBUS TCP slaves). If the connection does not exist in the current project, the data will not be imported.

Data type: the data type of the tag:

1. It cannot be null.
2. When the connection is an internal tag, any data type that is not supported by an internal tag cannot be imported.

3. For an existing connection, any data type that is not supported by the connection protocol cannot be imported.

Strings are case sensitive and strictly match the type in the variable editor. Invalid strings will be filtered.

Length: Length = Array count * Data type length, specifies tag length (the length of Bool type is in bit and that of other types is in byte), Length = Array count * Data type length, UInt16 (length is 2), Int16 (length is 2), UInt32 (length is 4), Int32 (length is 4), Float (length is 4), Double (length is 8), Bool (length is 1), String (length is 2–255), WString (length is 4–80), DataTime (length is 8).

1. It cannot be null.
2. When the data type is Wstring or String: The value is only valid when it is 1.
3. When the the data type is of any type other than Wstring or String: The value is an integer within 1–1600.

Array count: the number of elements in an array:

1. It cannot be null.
2. When the data type is Wstring or String: The value is only valid when it is 1.
3. When the the data type is of any type other than Wstring or String: The value is an integer within 1–1600.

Address The address of the specified tag

1. It cannot be null.
2. The address is invalid and is not imported if not as specified by the protocol or the internal tag.
3. The address value should be formatted correctly, such as D 0 instead of D0, where the space cannot be omitted.

Acquisition cycle: The acquisition period of a specified tag:

1. It cannot be null.
2. Any period that is not 1h, 1min, 10s, 5s, 2s, 1s, 500ms, or 100ms, or that is not added in InoTouchPad is invalid.

Acquisition mode: The acquisition mode of a specified tag:

1. It cannot be null.
2. The value must be 0, 1, and 2 (0-on demand, 1-cyclic on use, and 2-cyclic continuous).

Maximum: restricts objects to numeric type:

1. When the import data is of UInt16, Int16, UInt32, Int32, Float, or Double type, any value beyond the limit is invalid (it can be void, meaning that there is no limit).
2. For Bool, WString, String and DataTime, the value must be void.
3. Maximum must be greater than minimum.
4. Non-numeric strings, including valid tag names, are invalid (The value is not exported if it is a tag).

Minimum: restricts objects to numeric type:

1. When the import data is of UInt16, Int16, UInt32, Int32, Float, or Double type, any value beyond the limit is invalid (it can be void, meaning that there is no limit).

2. For Bool, WString, String and DataTime, the value must be void.
3. Maximum must be greater than minimum.
4. Non-numeric strings, including valid tag names, are invalid (The value is not exported if it is a tag).

Start value: the start value of the specified tag:

1. When the import data is of UInt16, Int16, UInt32, Int32, Float, or Double type, any value beyond the limit is invalid (it can be void, meaning that there is no limit).
2. For Wstring and String types, the value is invalid if the string size is greater than the tag length.
3. For the DataTime type, the time value must be correct and in the format of "yyyy-MM-dd hh:mm:ss" or "yyyy/MM/dd hh:mm:ss" (note: the hour, minute and second are not necessary, but the year, month and day must be provided).

Comment: The comment of a tag.

Group ID: the tag group to which the tag should be imported. This is only effective when you import tag under Show All Tags, and the group ID must also exist.

Text List

During export, the text list information is written to a *. csv file. The first 10 rows of the exported text are explanatory information and headers of the Text list, and the remaining rows are the exported data, which are in 5 columns: name, selection, comment, value, and entry. The fields are separated by ",". When a field is not empty, the contents are enclosed by double quotation marks. If the contents contain double quotation marks, the " will be escaped by another ", that is ""..."". (Note: The "," used to separate fields and the double quotation marks enclosing the contents of fields are English characters)

During import, the first 10 lines of the import text are in a fixed format, and can not be added, deleted or changed. In the import file, each entry occupies an independent line containing 5 fields which cannot be added or deleted:

Name: The name of the text list. If a name is empty, the line of data will not be imported. If the name exists in the project database, the imported data will overwrite the data in the database, otherwise, new data is added.

Selection:

1. It cannot be null.
2. Values other than 0, 1, and 2 are invalid. (0-bit, 1-bit number, 2-range value)

Comment: no restrictions.

Value:

1. 0 (bit):

   a. Values other than 0 or 1 are invalid.
   b. The value that already exists in the database is invalid.

2. 1 (bit number):

   a. Values other than an integer in 0–31 are invalid.
   b. The value that already exists in the database is invalid.

3. 2 (range), single value:

a. Values other than an integer in -2147483648–2147483647 are invalid.
b. The value that already exists in the database is invalid.

4. 2 (range), single value:

a. Except for valid single-valued integers, all values that are not in the format of "number - number" are invalid (note that there must be one, and only one, space before and after "-"). It is invalid if the minimum value is greater than or equal to the maximum value, or the value is beyond the range.

Entry: no restrictions.

Analog Alarm

During export, the alarm list information is written to a *. csv file. The first 16 rows of the contents are explanatory information and headers of the text list, and the remaining rows are the exported data, which are in 11 columns: text, alarm class, trigger tag, limit, trigger mode, hysteresis mode, hysteresis, hysteresis in percent, delay, infotext and alarm group. The fields are separated by ",". When a field is not empty, the contents are enclosed by double quotation marks. If the contents contain double quotation marks, the " will be escaped by another ", that is ""..."". (Note: The "," used to separate fields and the double quotation marks enclosing the contents of fields are English characters)

The first 16 lines of the import text are in a fixed format, and can not be added, deleted or changed. In the import file, each alarm occupies an independent line containing 11 fields which cannot be added or deleted:

Text: no restrictions.

Alarm class:

1. It cannot be null.
2. Any class that is not in the alarm class is invalid.
3. Cannot be the system alarm.

Trigger tag:

1. Tag names that are not of type UInt16, Int16, UInt32, Int32, Float, or Double are invalid.
2. Tag names that do not exist in the tag list are invalid.
3. Tag names whose array count is not 1 are invalid.

Limit:

1. When there is no trigger tag, the limit must be null.
2. When there is a trigger tag, the limit must not exceed the tag range.
3. Tag names cannot be imported. (Note: Constants can be imported, but tags cannot.)

Trigger mode: any value except 0, 1, 2, 3, and 4. It cannot be null. (0: less than 1: greater than 2: equal to 3: less than or equal to 4: greater than or equal to)

Hysteresis mode: any value except 0, 1, 2, and 3. It cannot be null. (0: Off; 1: On "activated"; 2: On "activated" and "deactivated"; 3: On "deactivated")

Hysteresis:

1. It cannot be null.
2. It can only be 0 when the hysteresis mode is off.

3. When the hysteresis mode is not off, it must be an integer in 0–100.

Hysteresis percent:

1. It cannot be null.
2. It can only be 0 when the hysteresis mode is off.
3. When the hysteresis mode is not off, it must be 0 or 1.

Delay:

1. It cannot be null.
2. Values other than an integer in 0–10000 are invalid.

Infotext: No restrictions.

Alarm group: Any name that is not in the alarm group is invalid. It can be null.

Discrete alarm

During export, the alarm list information is written to a *. csv file. The first 16 rows of the contents are explanatory information and headers of the alarm list, and the remaining rows are the exported data, which are in 11 columns: text, alarm class, trigger tag, Trigger bit, Info text, Alarm Group, HMI ack tag, HMI ack bit, PLC ack tag, PLC ack bit and trigger mode. The fields are separated by ",". When a field is not empty, the contents are enclosed by double quotation marks. If the contents contain double quotation marks, the " will be escaped by another ", that is ""..."". (Note: The "," used to separate fields and the double quotation marks enclosing the contents of fields are English characters)

The first 16 lines of the import text are in a fixed format, and can not be added, deleted or changed. In the import file, each alarm occupies an independent line containing 11 fields which cannot be added or deleted:

Text: no restrictions.

Alarm class:

1. It cannot be null.
2. Any class that is not in the alarm class is invalid.
3. Cannot be the system alarm.

Trigger tag:

1. The tag must be of Uint16, Int16 and Bool type.
2. The tag name must exist in the tag list.
3. The array count of the trigger tag in the tag list must be 1. (note: it can be null.)

Trigger bit:

1. When there is no trigger tag, the trigger bit cannot be null.
2. It can only be 0 when the trigger tag is of Bool type.
3. It must be an integer in 0–15 when the the trigger tag is of Int16 or UInt16 type.
4. When there is a trigger tag, the trigger bit cannot be null.
5. In the discrete alarm database, the bit number of the tag whose name is identical to the trigger tag must be different. (That is, the same tag cannot have the same bit number in the whole alarm list)

Infotext: No restrictions.

Alarm group: Any name that is not in the alarm group is invalid. It can be null.

HMI ack tag:

1. When the trigger tag is of Bool type, it must be null.
2. When the trigger tag is an internal tag, it must be null.
3. When the trigger tag has an external connection:

   a. The HMI ack tag must be of Int16 or Uint16 type.
   b. The HMI ack tag and the trigger tag must be different tags.
   c. The HMI ack tag name must exist (can be null).

4. When the trigger tag is null:

   a. The HMI ack tag must be of Int16 or Uint16 type.
   b. The data type of The HMI ack tag must exist (can be null).

HMI ack bit:

1. It must be an integer in 0–15 when the HMI ack tag is of Int16 or UInt16 Type.
2. When the HMI ack tag is null, it must be null.
3. The same HMI ack tag must not has the same bit.

PLC ack tag:

1. When the trigger tag is of Bool type, it must be null.
2. When the trigger tag is an internal tag, it must be null.
3. When the trigger tag has an external connection:

   a. The PLC ack tag must be of Int16 or Uint16 type.
   b. The PLC ack tag and the trigger tag must be the same.
   c. The PLC ack tag name must exist (can be null).

4. When the trigger tag is null, the PLC ack tag must be null.

PLC ack bit:

1. It must be an integer in 0–15 when the PLC ack tag is of Int16 or UInt16 type.
2. It must be null when the PLC ack tag is null.
3. In the discrete alarm database, the bit number of the tag whose name is identical to the PLC ack tag must be different.

Trigger mode: any value except 0, 1, 2, and 3. It cannot be null. (0: 0->1 1: 1->0; 2: ==0; 3: ==1)

Internationalization

During exporting, the translation text information is written to a *. csv file. The first 3 lines, which are in a fixed format, are some explanatory information of the tag list and the fourth line is the header of the list. The number of exported columns is determined by the start column of language and font. The contents after the fourth line are exported data. The fields are separated by ",". When a field is not empty, the contents are enclosed by double quotation marks. If the contents contain double quotation marks, the " will be escaped by another ", that is ""..."". (Note: The "," used to separate fields and the double quotation marks enclosing the contents of fields are English characters)

During importing, the first 3 lines of the import text are in a fixed format, and can not be added, deleted or changed.

Referenced by: 1. It cannot be null. 2. Users cannot define new reference information.

There are no restrictions for other columns.

Recipe

During exporting, the recipe data record list information is written to a *. csv file. The first line is in a fixed format, which is some description information of the recipe data record list. The second line is the header of the list. The number of exported columns is determined by the number and name of data records. The first column is the element name of the recipe, and the content after the second line (except the first column) is the exported data. The fields are separated by ",". When a field is not empty, the contents are enclosed by double quotation marks. If the contents contain double quotation marks, the " will be escaped by another ", that is ""..."". (Note: The "," used to separate fields and the double quotation marks enclosing the contents of fields are English characters)

During importing, the first 1 line of the import text are in a fixed format, and can not be added, deleted or changed.

Name: The name of an element. Generally, it will not be modified when exported. It is used to modify the recipe with data records.

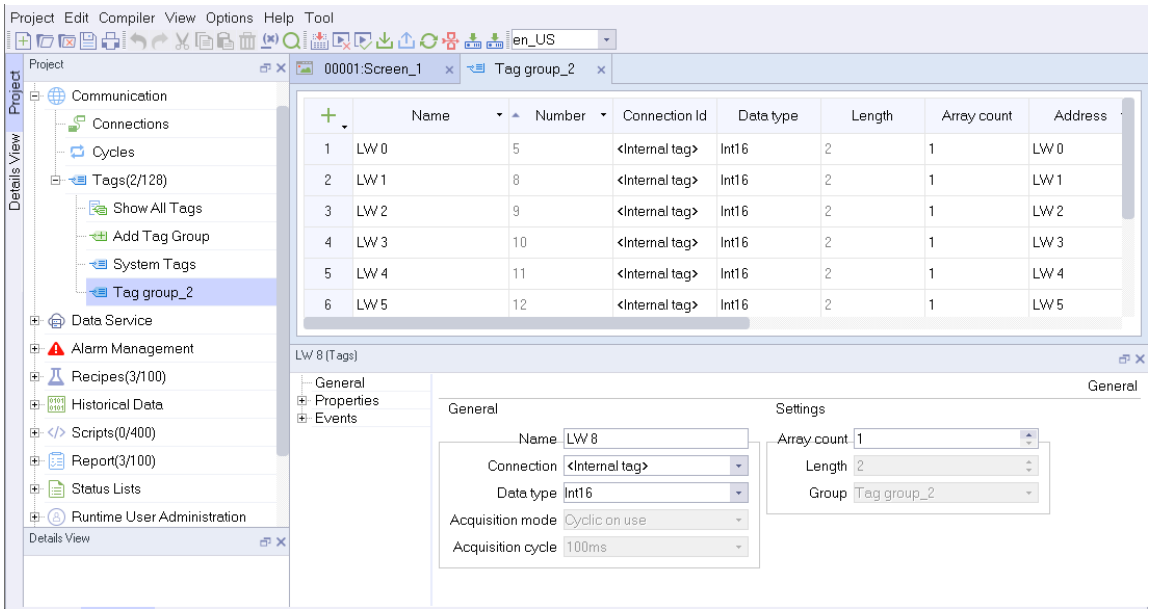
Data record name column: numerical data corresponding to each element:

1. If any data in this column is null, the data record is discarded during import.
2. If any data in this column does not match the element tag type, the data record is discarded during import.
3. If any data in this column exceeds element tag type upper limit, the data record is discarded during import.

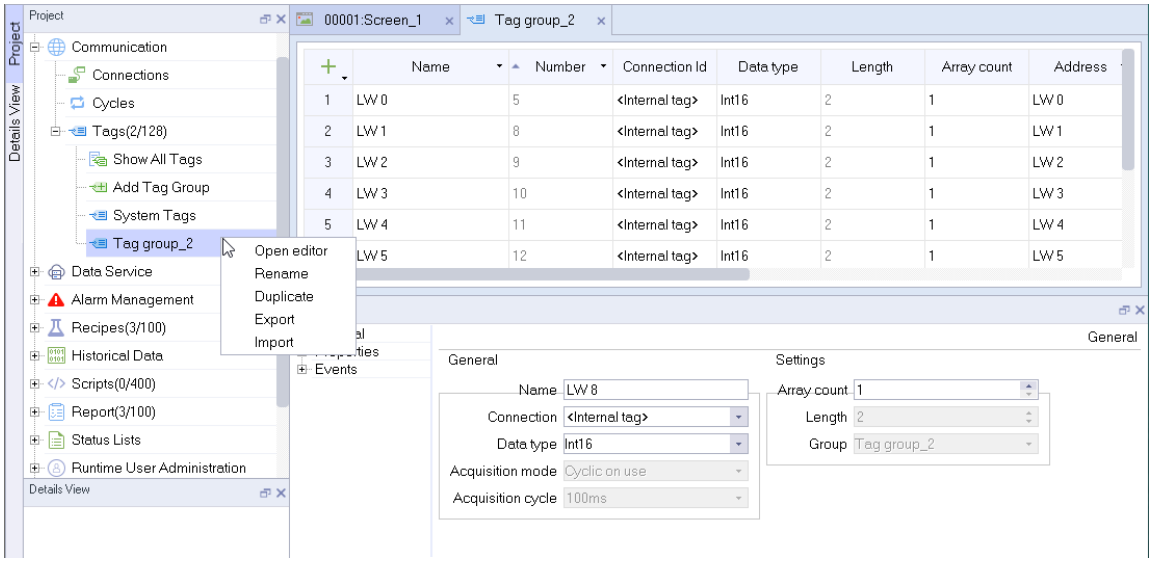
When editing the exported *. csv file, you cannot add new rows, that is, you cannot add components externally. If you add any new row, you cannot import the whole table because the number of components is incorrect. However, you can add new columns, that is, a new data record. The data record name in the table header must not be empty, otherwise, the data in this column will be filtered out during importing. In addition, if the data record name of the newly added column already exists, the data record data with the same name will be overwritten during importing; If the data record name of the new column does not exist, it will be imported into the software tool as a new data record. This also applies to importing and exporting in HMIAutorun.

16.3.2 Importing and Exporting Tags

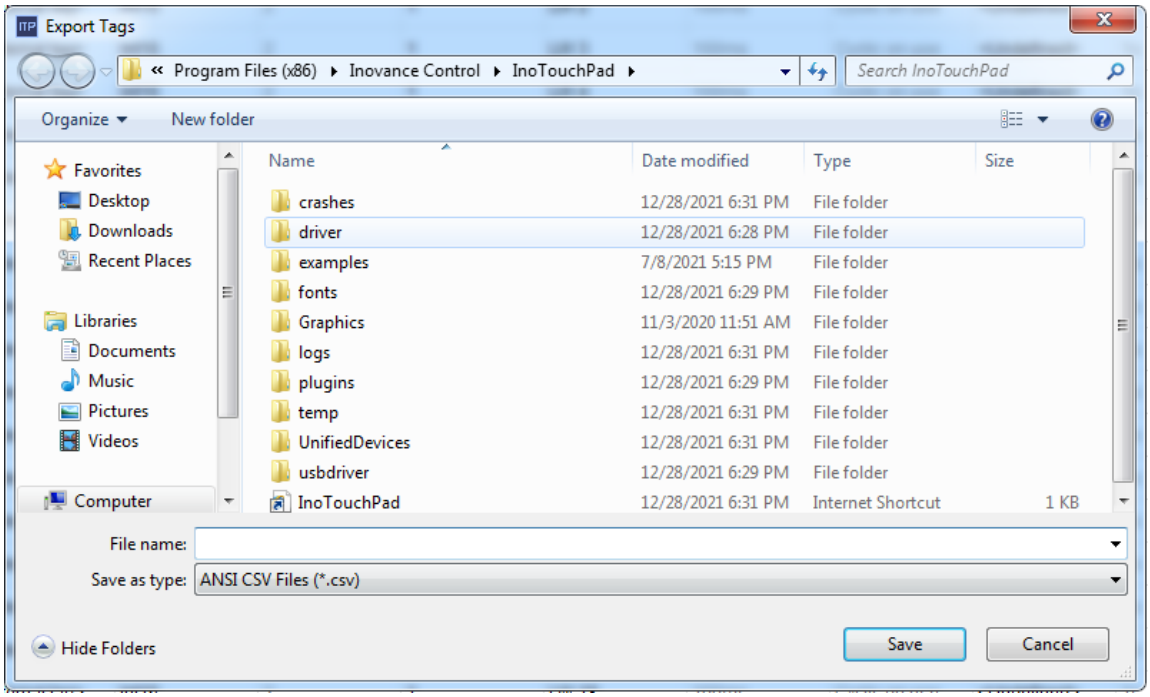
You can import and export a tag group or all tags. To export a tag group, expand Tags in the Project view, and then double-click the tag group (for example, Tag group_2) to open the editor.



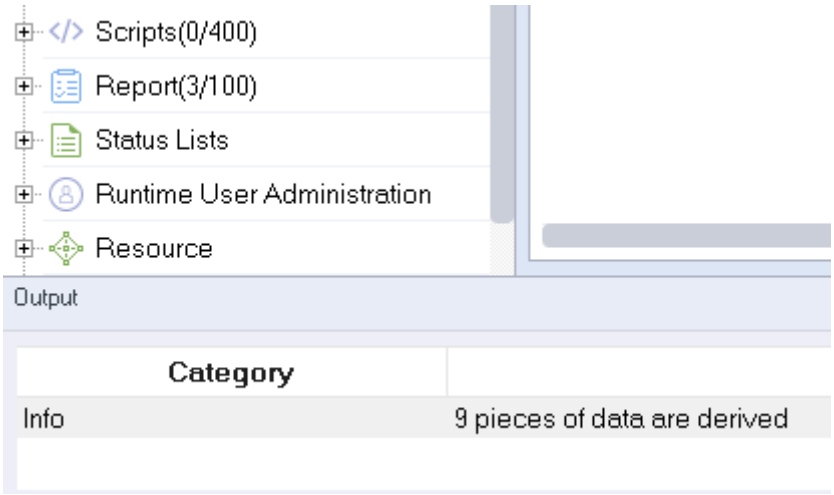
Right-click the tag group to show the following context menu:



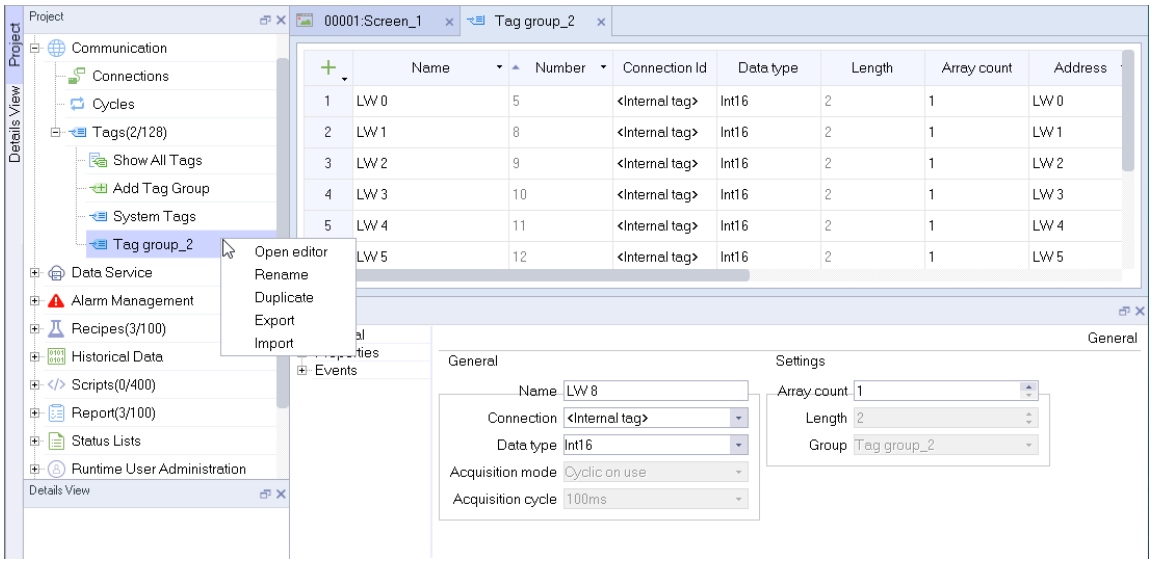
Click Export, and the export window pops up:



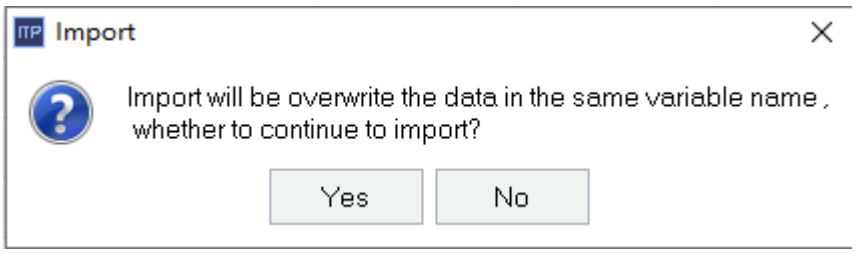
Click Save to export the file in CSV format, which can be viewed and edited by Excel. At the same time, the status bar shows the following message:



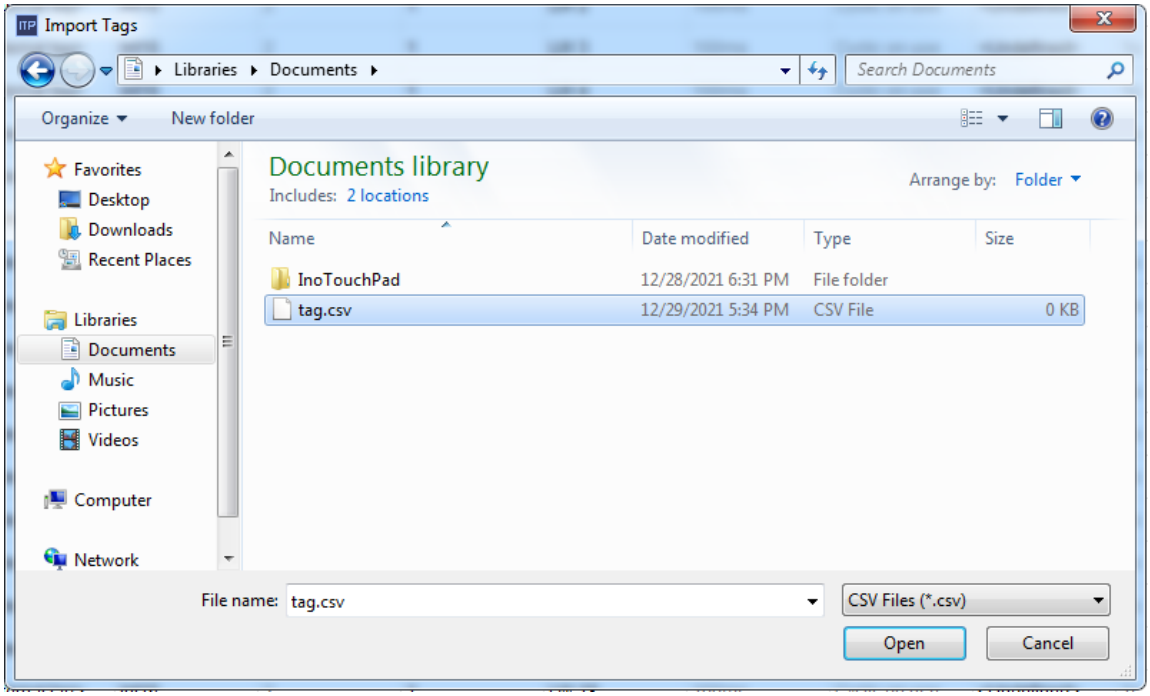
For import, similarly, also right-click the tag group to show the context menu:



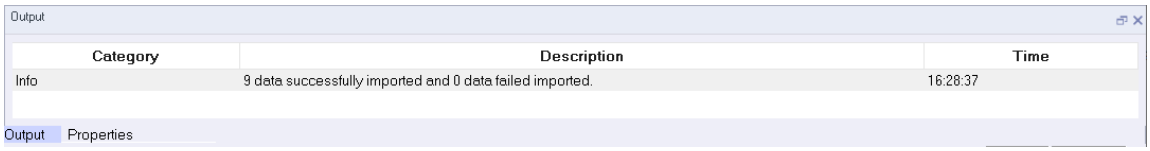
Click Import, and the following message box pops up:



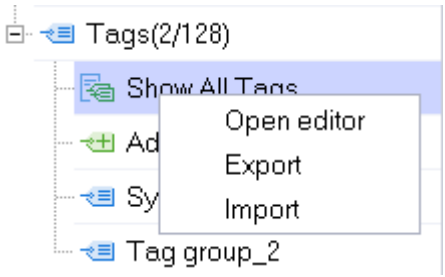
Click No to cancel importing and Yes to continue: The import window pops up:



Select the file to be imported in the pop-up window and click Open. after the import is completed, the output window will show a message about the result.



For import and export of all tags, you can double click Show All Tags and perform the above operations.



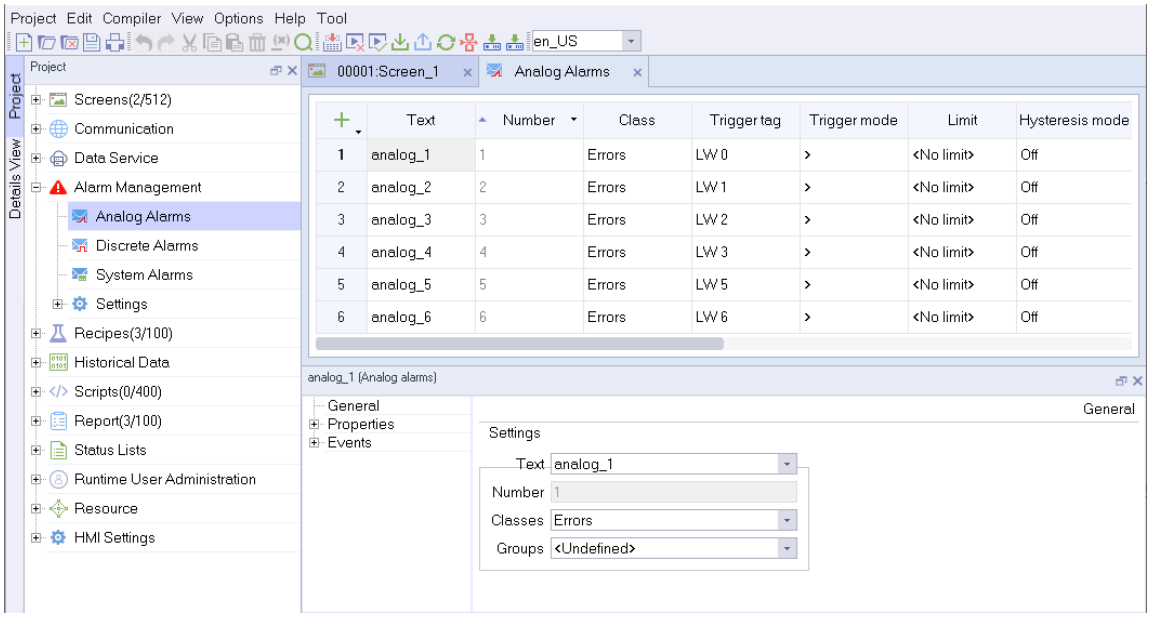
Caution

System Tag Group cannot be imported or exported.

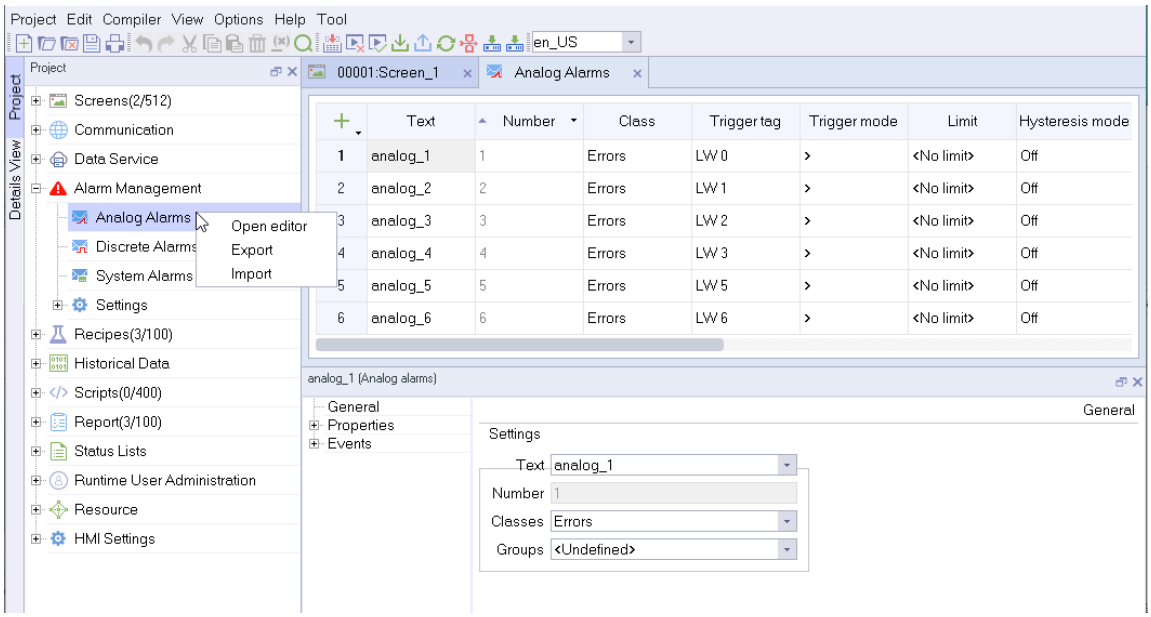
16.3.3 Importing and Exporting Alarms

You can import and export analog alarms and discrete alarms. This section instructs you how to import and export analog alarms.

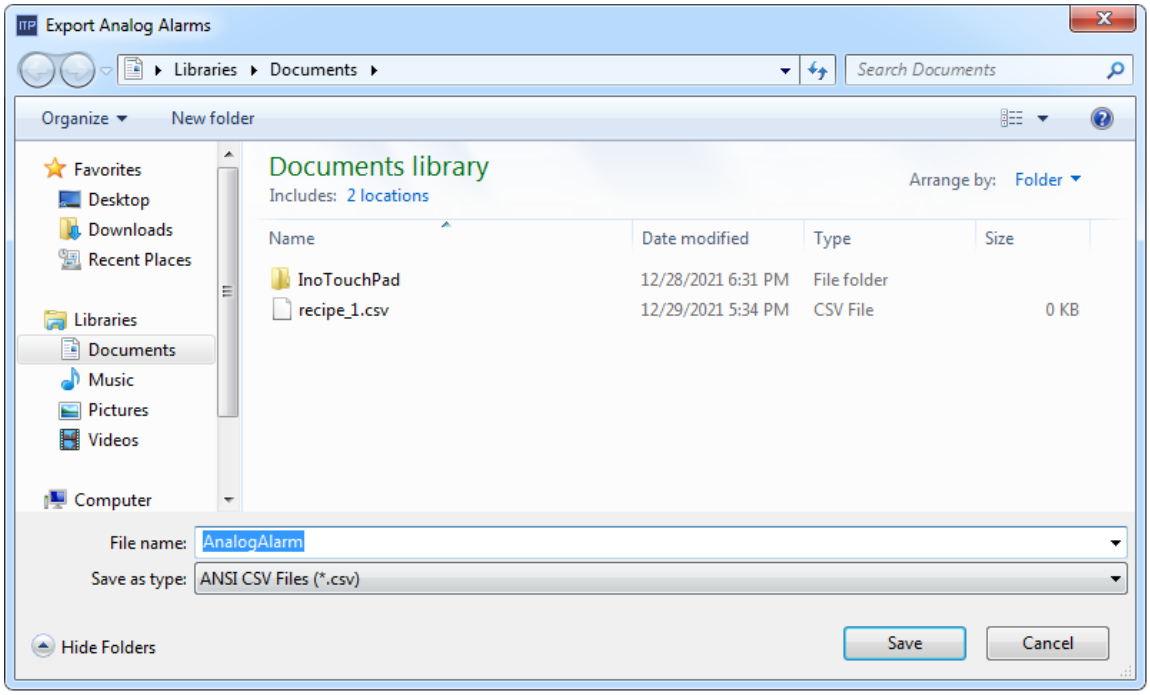
In the Project view, expand Alarm Management and then double-click Analog Alarms to open the editor.



Right-click Analog Alarms to show the following context menu:

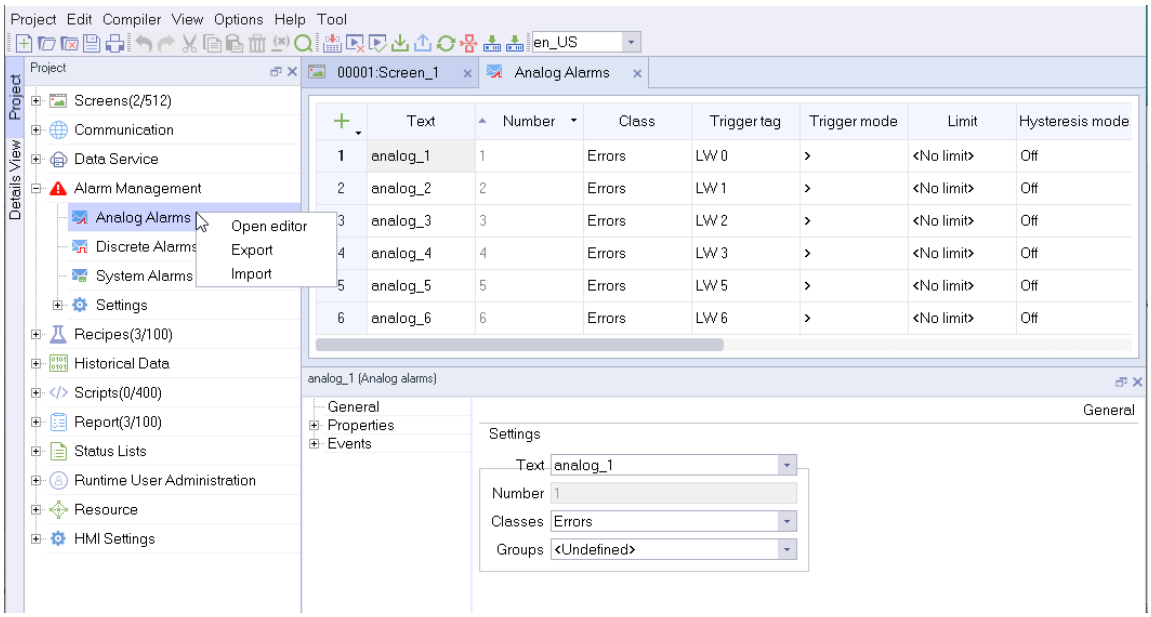


Click Export, and the export window pops up:

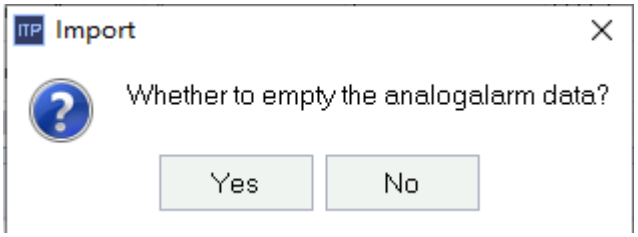


Click Save to export the file in CSV format, which can be viewed and edited by Excel. At the same time, the status bar shows the following message:

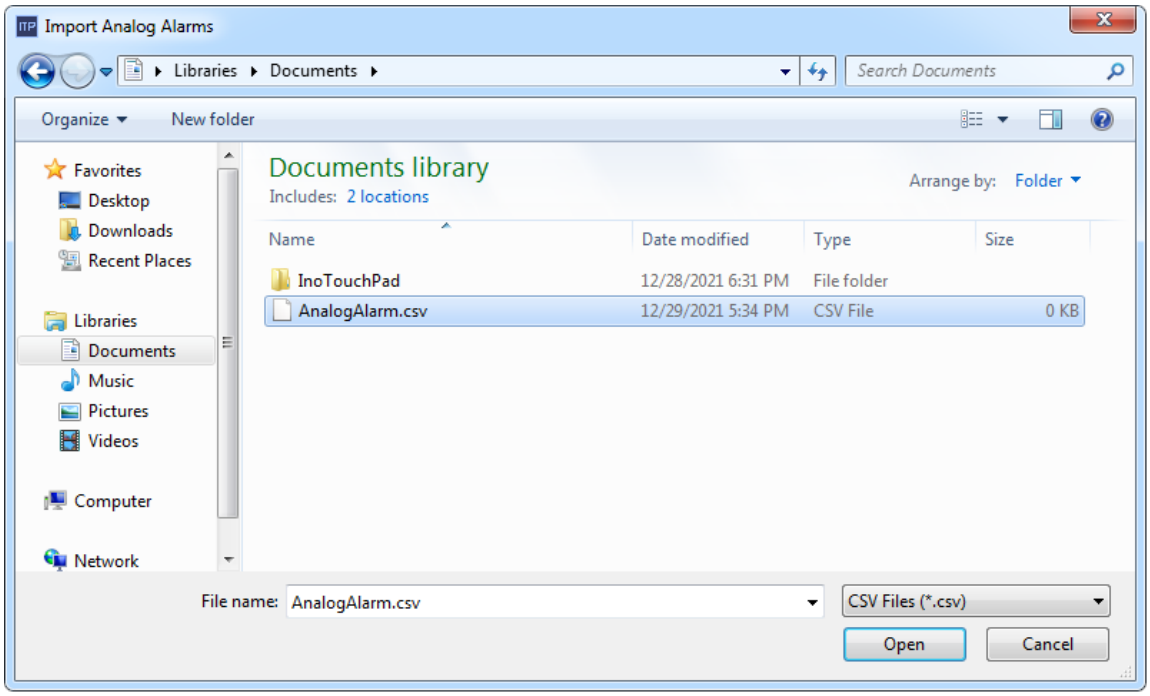
For import, similarly, also right-click Analog Alarms to show the context menu:



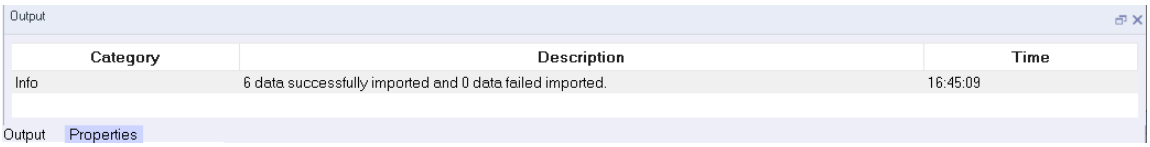
Click Import, and the following message box pops up:



The following dialog box shows to ask you whether to empty the analog alarm data before importing. If you only want to export the data to modify them and then import them again, it is recommended to click Yes and the following window will pop up:



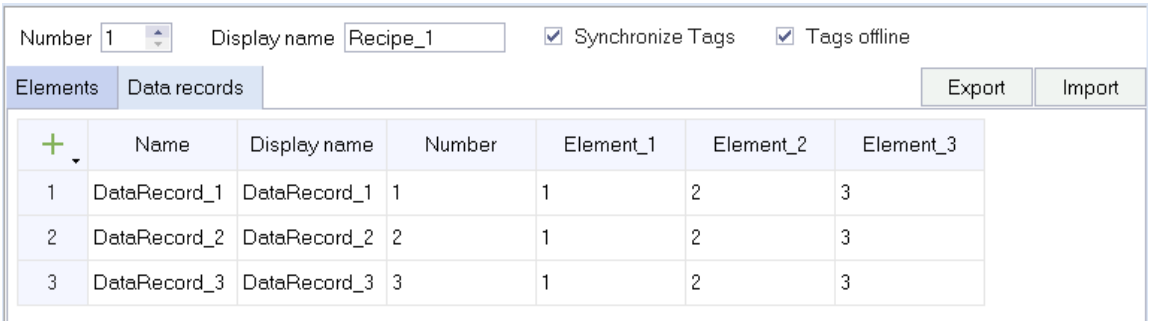
Select the file to be imported in the pop-up window and click Open. after the import is completed, the output window will show a message about the result.



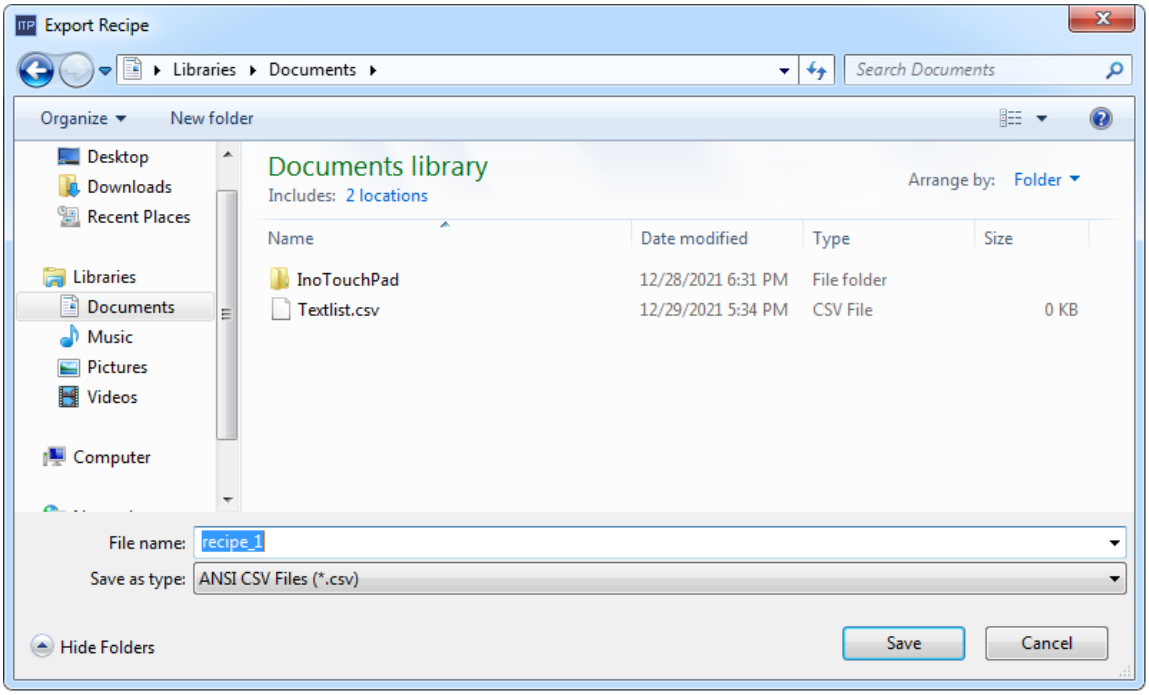
For import and export of discrete alarms, you can double click Discrete Alarms and perform the above operations.

16.3.4 Importing and Exporting the Recipe

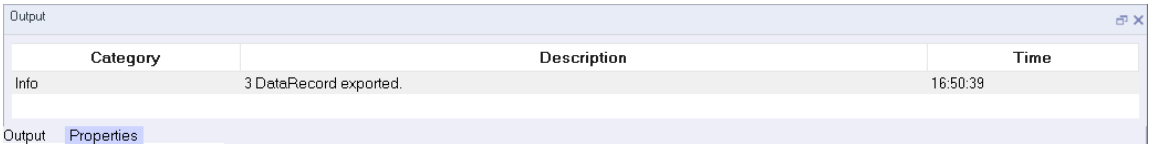
You can import and export the data logs of a recipe. Enter the data log workspace of recipe_1, as shown in the following figure:



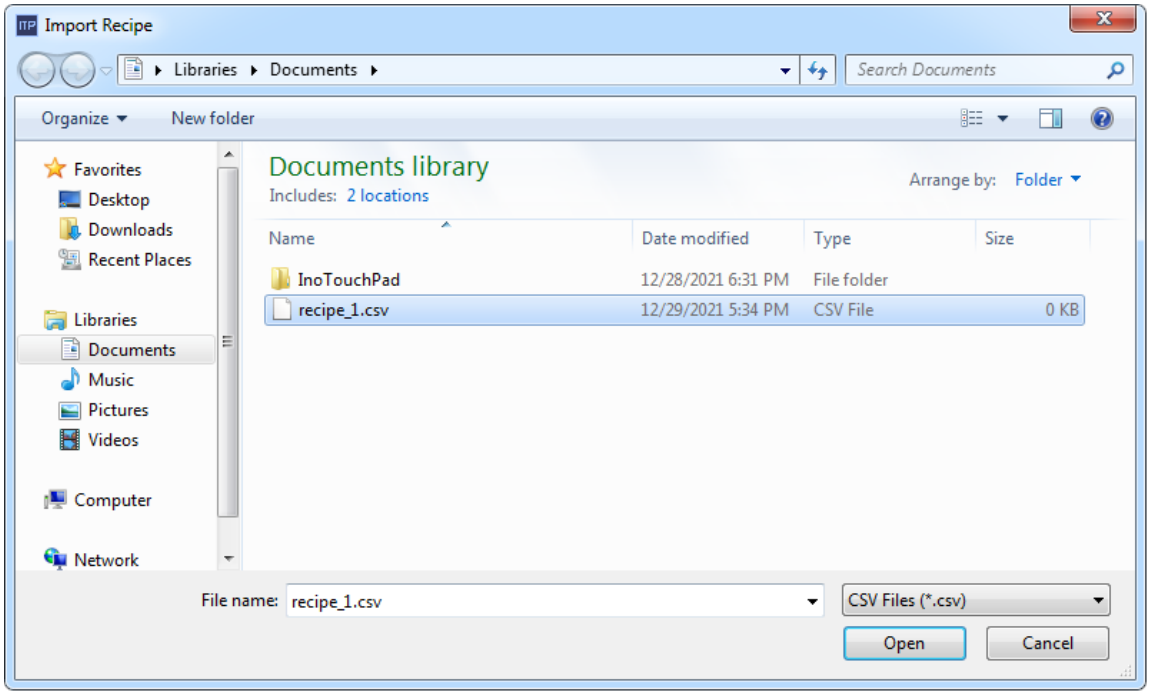
Then click the Export button in the upper right corner. Input the name of the exported file in the pop-up window.



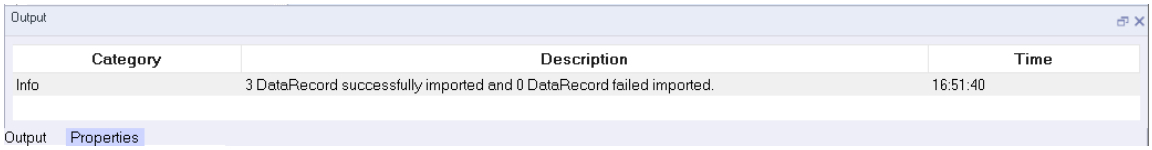
Click Save to export the file in CSV format, which can be viewed and edited by Excel. At the same time, the output window will also show a message.



Likewise, you can click the Import button in the upper right corner to import a file.

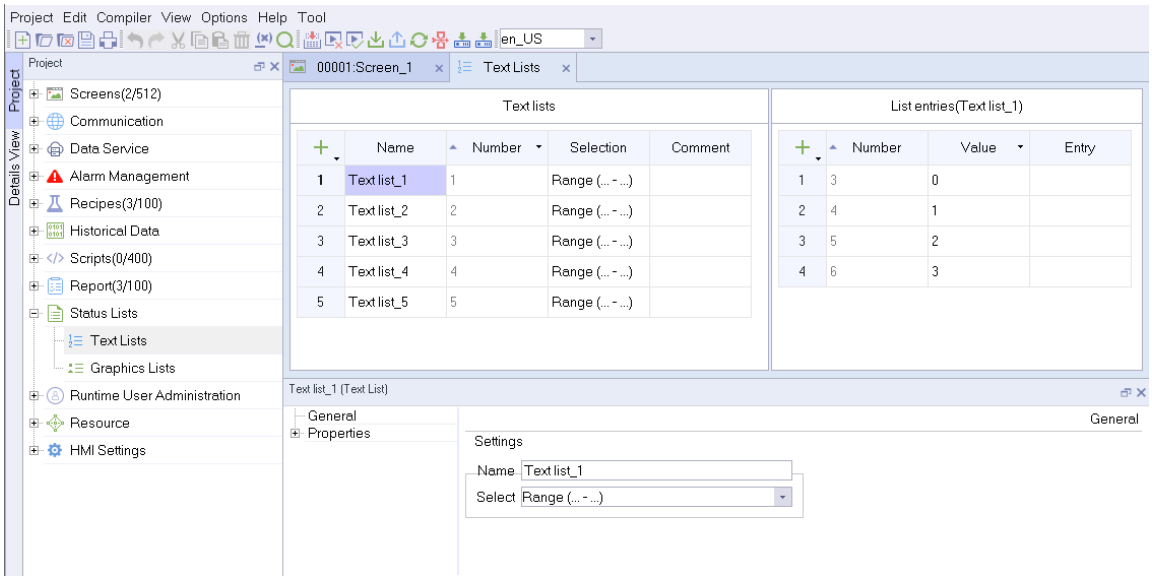


Select the file to be imported in the pop-up window and click Open. after the import is completed, the output window will show a message about the result.

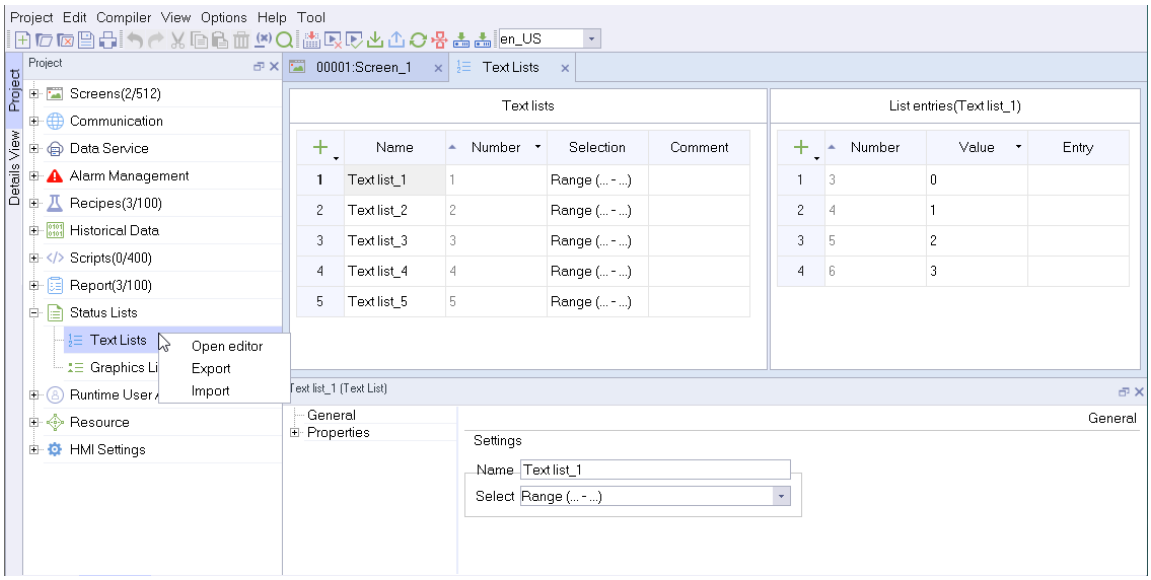


16.3.5 Importing and Exporting Text List

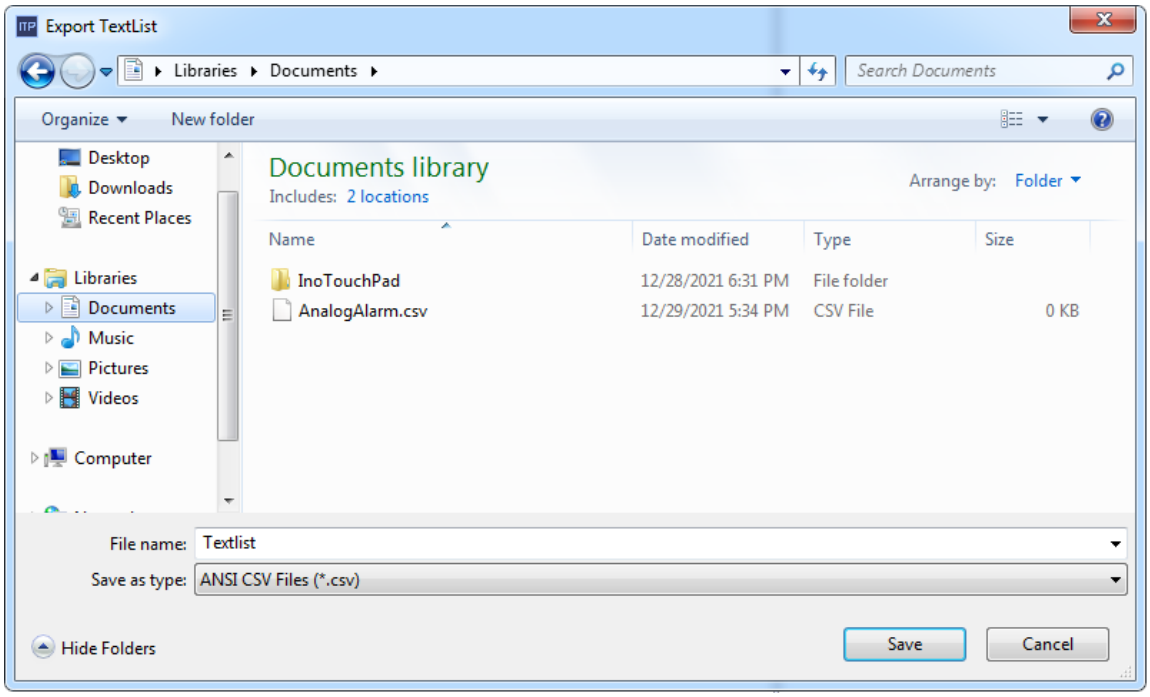
To import and export a text list, go to Status Lists→Text Lists, and then double-click Text Lists to open the text list editor. The workspace displays all the created text lists:



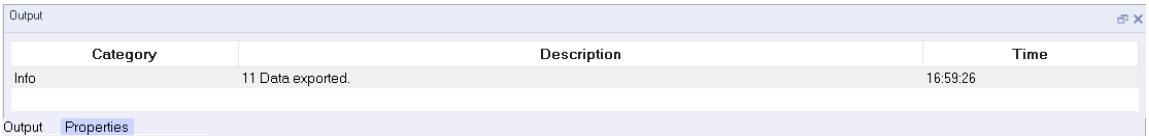
Right-click on the Text Lists:



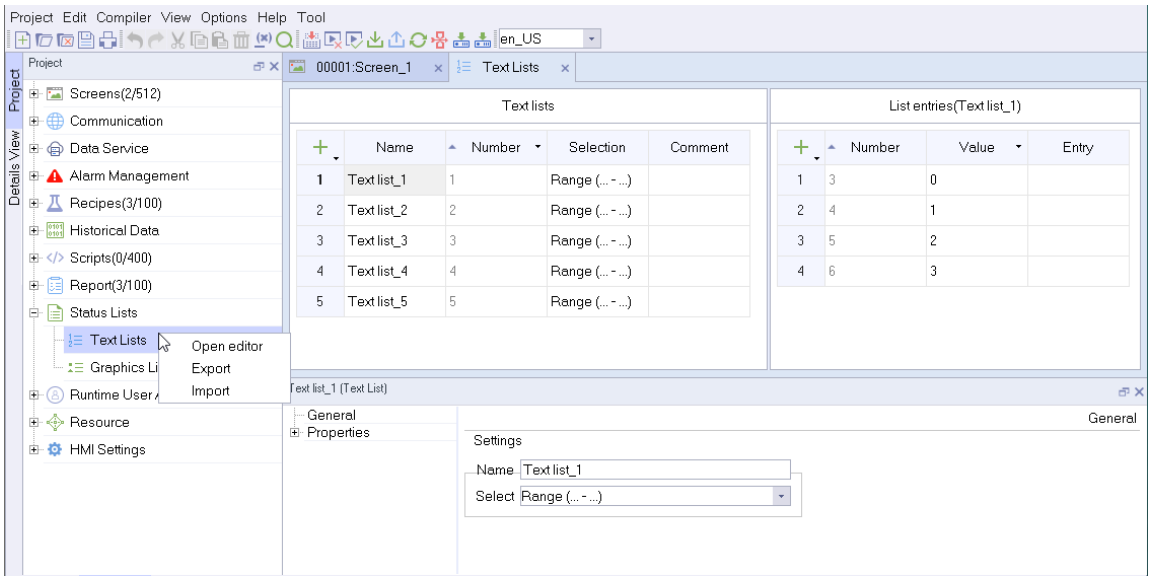
Click Export, and the export window pops up:



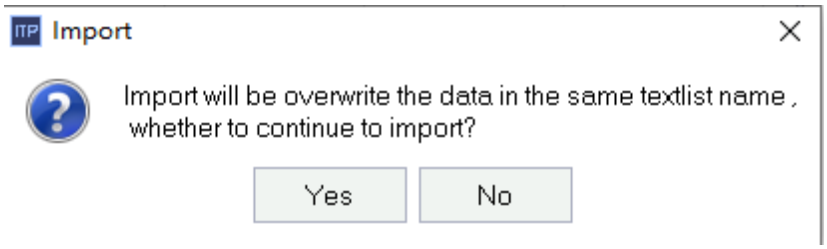
Click Save to export the file in CSV format, which can be viewed and edited by Excel. Note that the data of text lists and list entries are merged. At the same time, the status bar shows the following message:



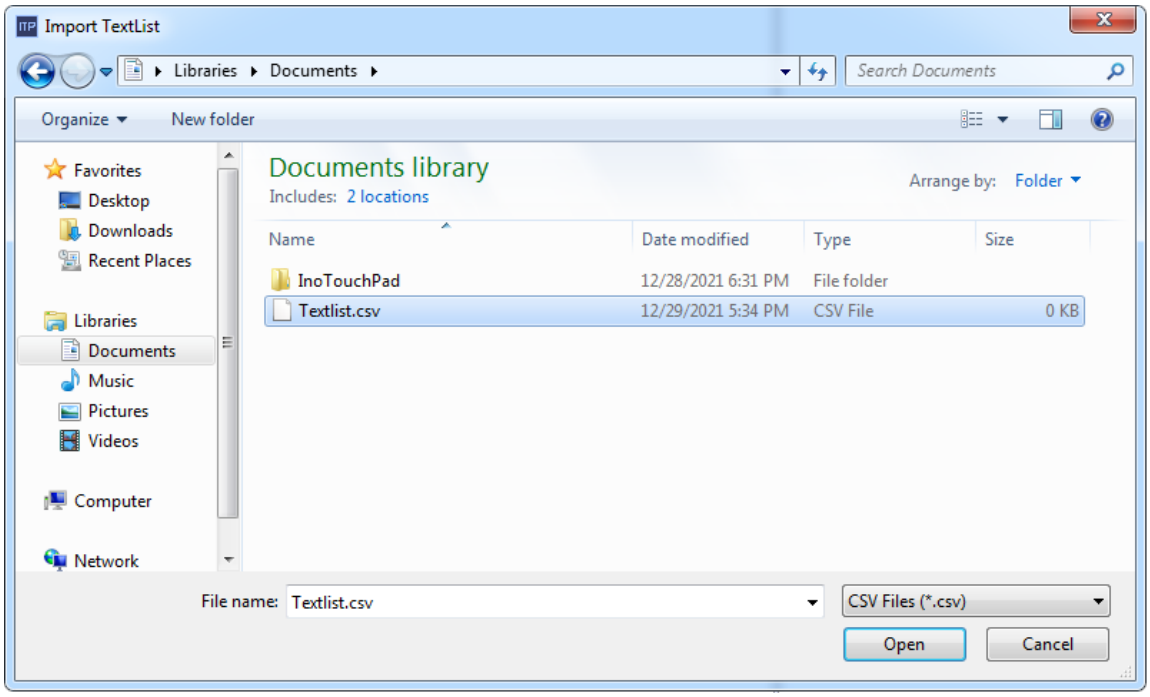
For import, similarly, also right-click Text Lists:



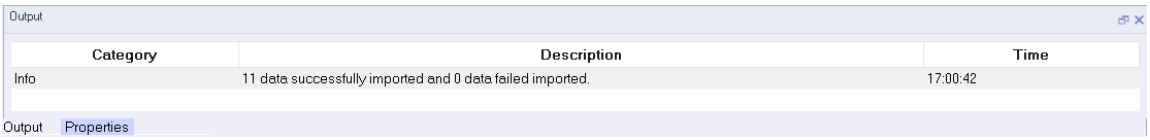
Click Import, and the following message box pops up:



Click No to cancel importing and Yes to continue: The import window pops up:



Select the file to be imported in the pop-up window and click Open. after the import is completed, the output window will show a message about the result.

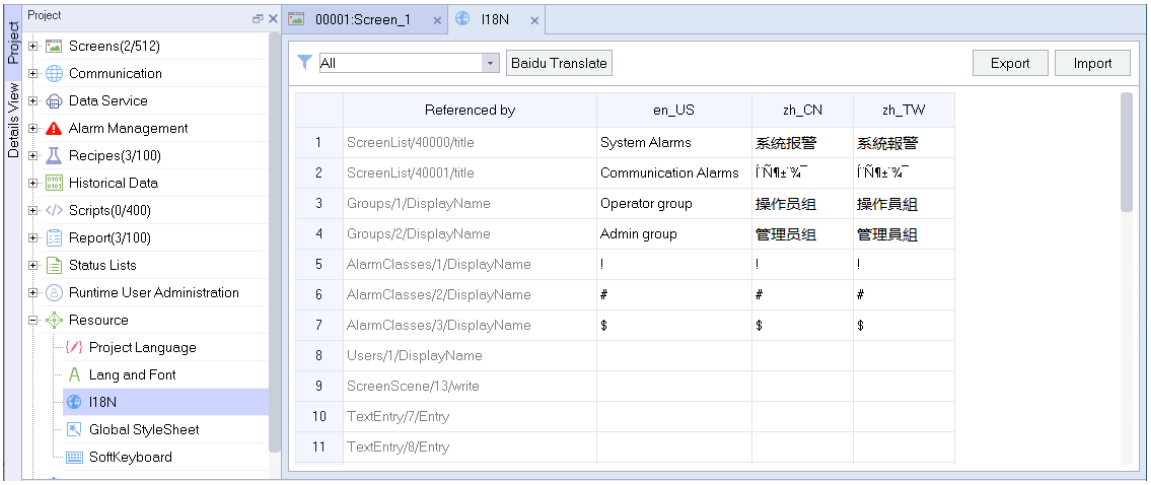


16.3.6 Importing and Exporting Translation Text

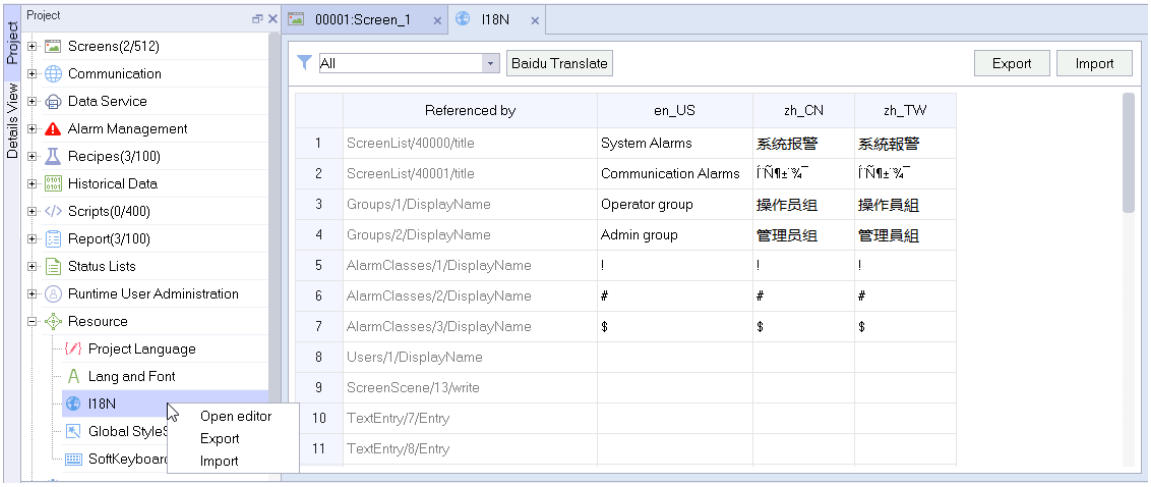
The export function enables you to export project data for viewing, editing and modifying;

The import function enables you to import the modified exported data, and then apply these data in a project.

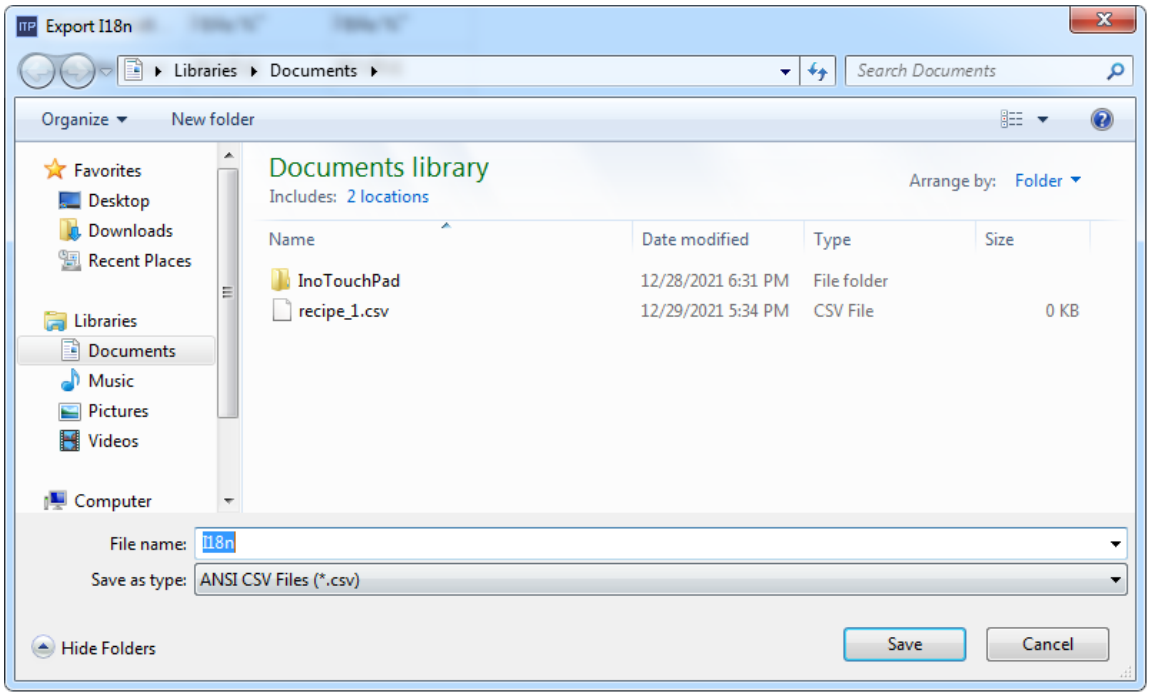
In the Project view, expand Resource and then double-click I18N to open the editor.



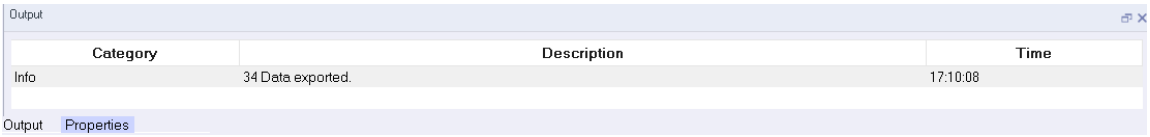
Right-click on I18N.



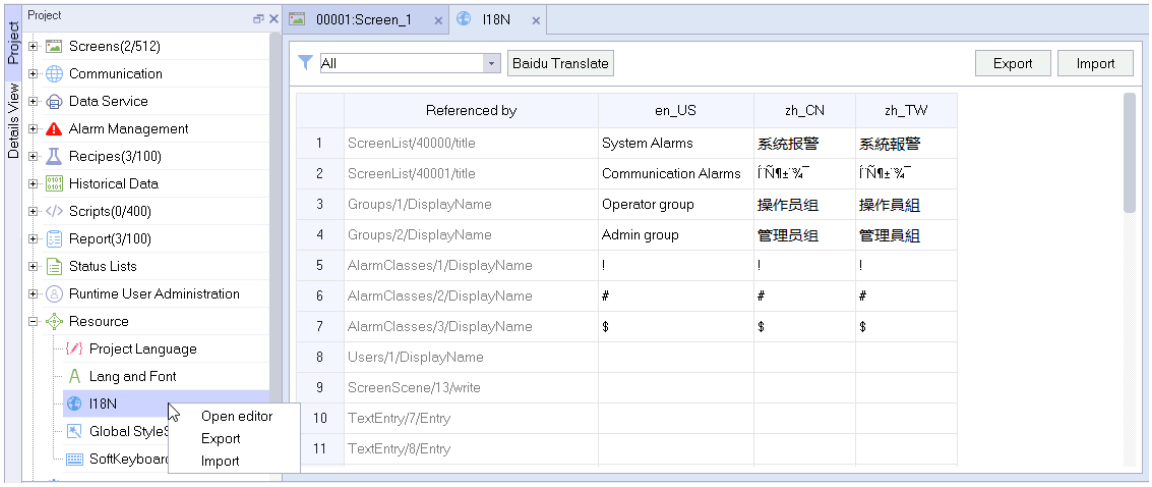
Click Export in the context menu (or click the Export button in the upper right corner). Input the name of the exported file in the pop-up window.



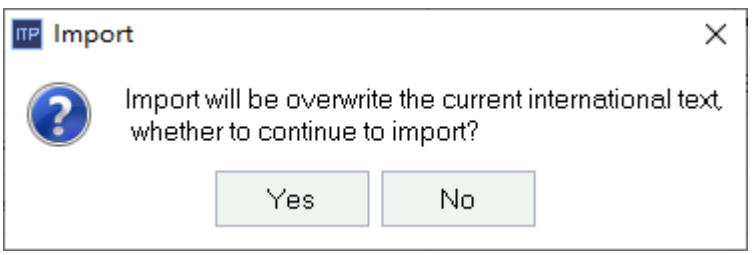
Click Save to export the file in CSV format, which can be viewed and edited by Excel. At the same time, the status bar shows the following message:



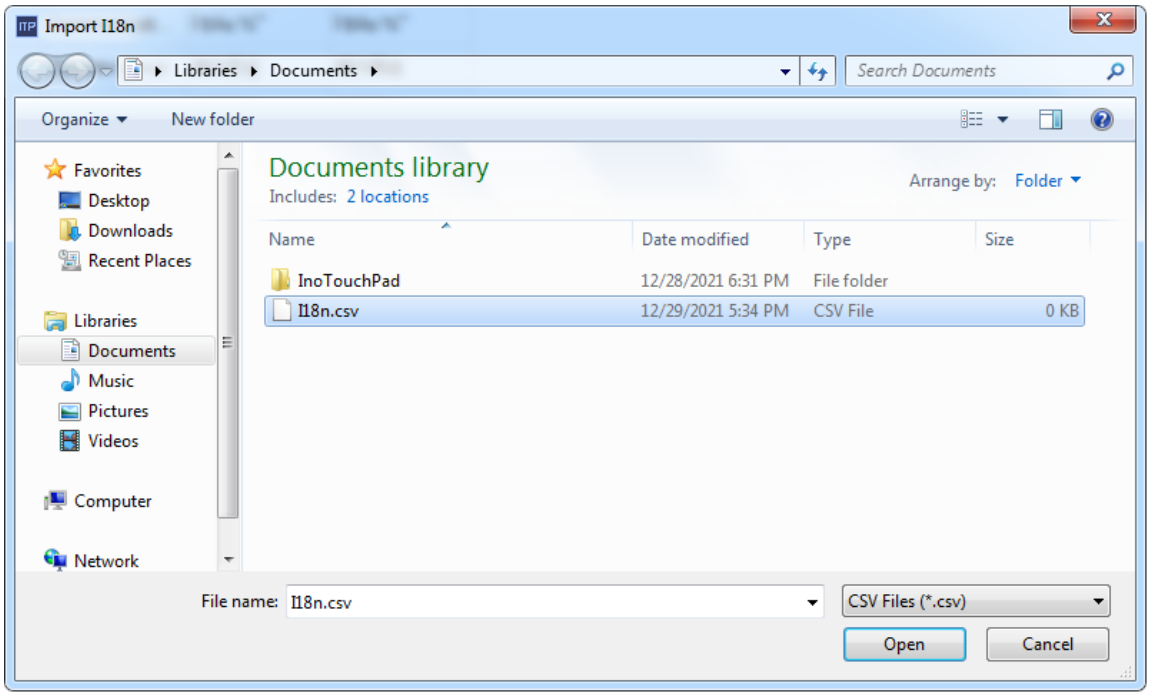
For import, similarly, also right-click I18N:



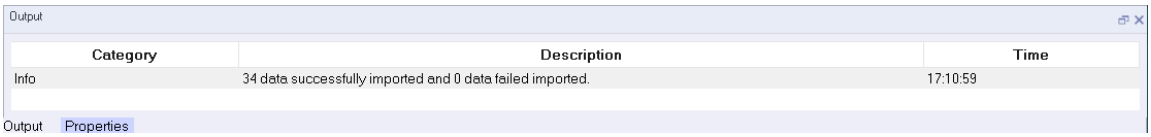
Click Import, and the following message box pops up:



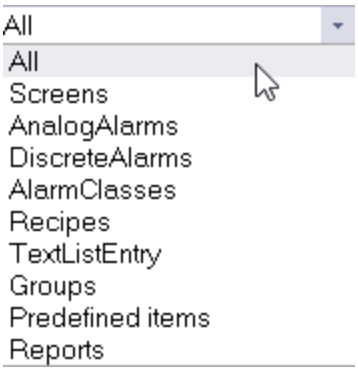
Click No to cancel importing and Yes to continue: The import window pops up:



Select the file to be imported in the pop-up window and click Open. after the import is completed, the output window will show a message about the result.



You can use the filter to only export translation text in a specific category:

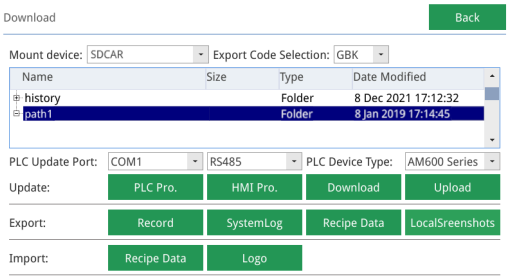


During importing, the imported text and the text displayed by the current filter must be of the same category. Otherwise, the import will fail. For example, if you import translation text in the Screens category while the filter condition is Recipes,warnings will display in the Output window and no item is imported.

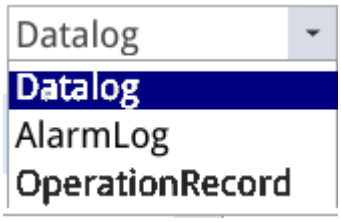
16.3.7 Exporting Data Logs

If a project has configured data logging, after the runtime runs for a period of time, you can export the logged data in the control panel of HMI.

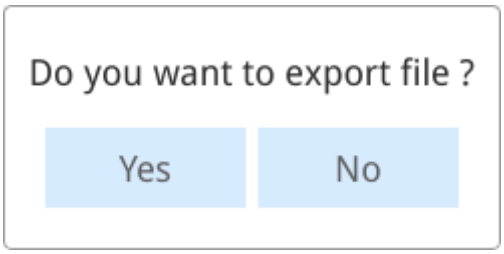
Insert a USB drive or SD card into the HMI, enter the download page of the control panel, and select the directory where the log to be exported, as shown in the following figure:



Then click the Log button in the Export field, and the following dialog box will pop up:



Select Data Log and click Yes.

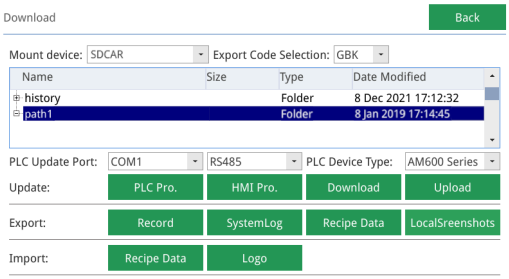


Click Yes in the message box to start export. After the progress data reaches 100%, you will be prompted if the export is successful or failed. If the user did not select an export path, it is exported to the root directory by default. You can open the exported csv file with Excel.

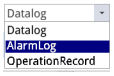
16.3.8 Exporting Alarm Logs

If a project has configured alarm logging, after the runtime runs for a period of time, you can export the logged alarm data in the control panel of HMI.

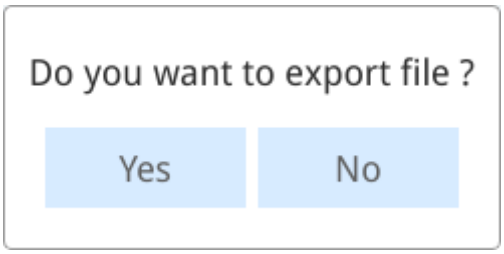
Insert a USB drive or SD card into the HMI, enter the download page of the control panel, and select the directory where the log to be exported, as shown in the following figure:



Then click the Log button in the Export field, and the following dialog box will pop up:



Select Alarm Log and click Yes.

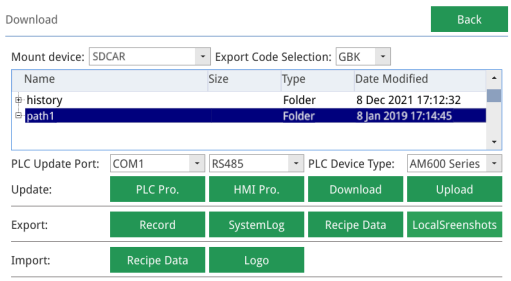


Click Yes in the message box to start export. After the progress bar reaches 100%, you will be prompted if the export is successful or failed. If the user did not select an export path, it is exported to the root directory by default. You can open the exported csv file with Excel.

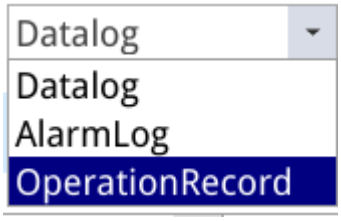
16.3.9 Exporting Operating Log

If a project has enabled operation logging, after the runtime runs for a period of time, you can export the logged operation data in the control panel of HM.

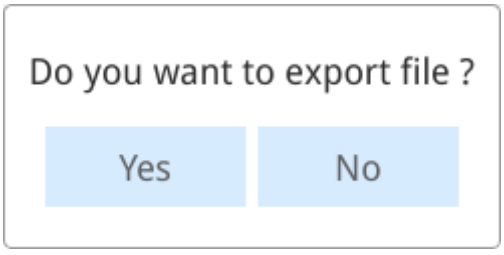
Insert a USB drive or SD card into the HMI, enter the download page of the control panel, and select the directory where the log to be exported, as shown in the following figure:



Then click the Log button in the Export field, and the following dialog box will pop up:



Select Operation Log and click Yes.

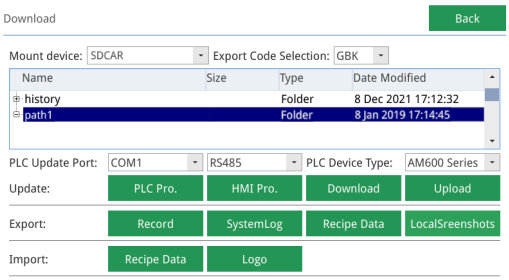


Click Yes in the message box to start export. After the progress bar reaches 100%, you will be prompted if the export is successful or failed. If the user did not select an export path, it is exported to the root directory by default. You can open the exported csv file with Excel.

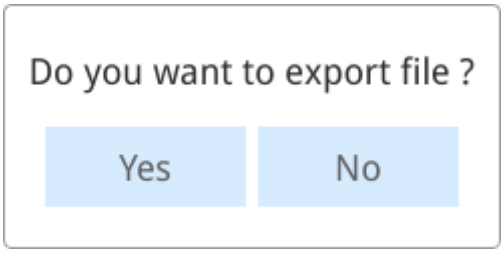
16.3.10 Exporting the Log

The log is some system output information recorded by runtime, which is stored in *. txt format. The export steps are as follows:

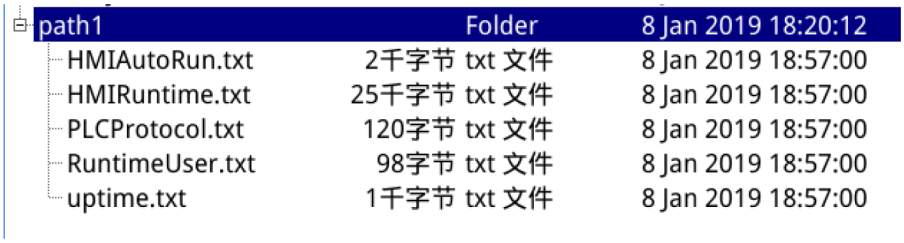
Insert a USB drive or SD card into the HMI, enter the download page of the control panel, and select the directory where the log to be exported, as shown in the following figure:



Then click the "Log" button in the Export field, and the following dialog box will pop up:



Click Yes to start export. After the progress bar reaches 100%, you will be prompted if the export is successful or failed. The export file is in *. txt format and stored in the path specified by the user. The export file is as follows:



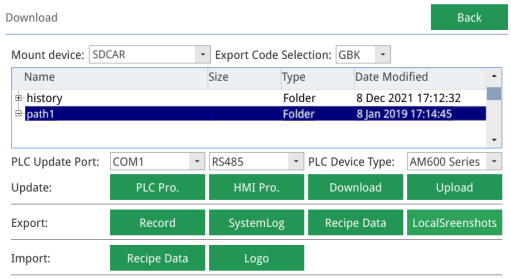
When the user does not select a path, it is exported to the root directory by default.

16.3.11 Importing and Exporting the Recipe

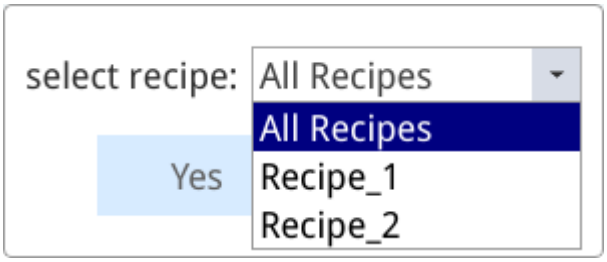
You can export the data record of a configured recipe in the control panel of the HMI.

The steps are as follows:

Insert a USB drive or SD card into the HMI, enter the download page of the control panel, and select the directory where the log to be exported, as shown in the following figure:



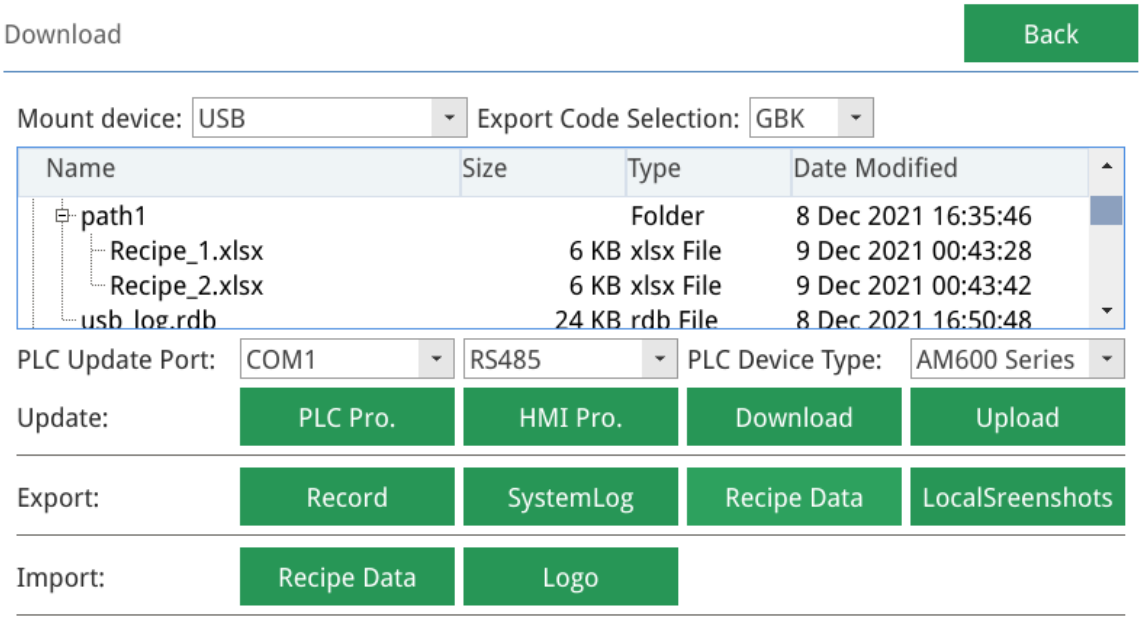
Then click the Recipe Data button in the Export field, and the following dialog box will pop up:



The dialog box will list all configured recipes. You can export the data record of a specified recipe, or all the recipes. Click Yes in the message box to start export. After the progress reaches 100%, you will be prompted if the The export is successful or failed. If the user did not select an export path, it is exported to the root directory by default. You can open and edit the exported csv file with Excel. You can also import the edited csv file back into the runtime.

The steps are as follows:

Insert a USB drive or SD card into the HMI, enter the download page of the control panel, and select the directory where the file to be exported, as shown in the following figure:



Then click the Recipe Data button in the Import field to import the file. A popup window will prompt you if the import is successful or failed.

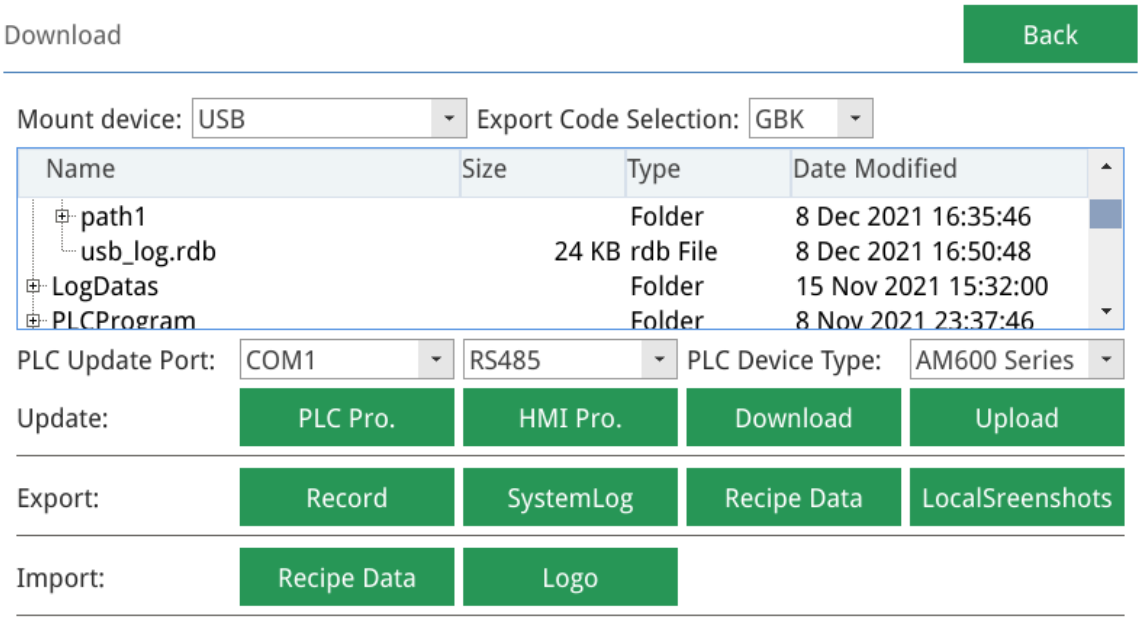
Caution

When importing recipes, if you select a directory, all recipe files in the selected directory will be imported into the project. This method can be used to import all modified recipe files at one time.

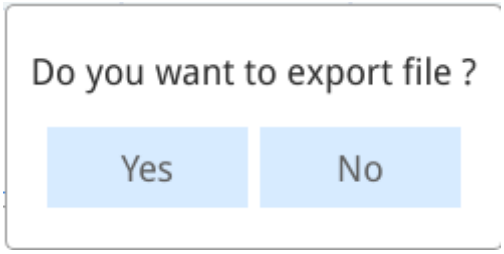
16.3.12 Exporting Local Screenshots

In the runtime, if you configured the screen printing system function, the printed screens will be saved to the local HMI path. At this time, the user can obtain these printed screens through local screenshot export. The steps are as follows:

Insert a USB drive or SD card into the HMI, enter the download page of the control panel, and select the directory where the screenshot to be exported, as shown in the following figure:



Then click the Local Screenshot button in the Export field, and the following dialog box will pop up:

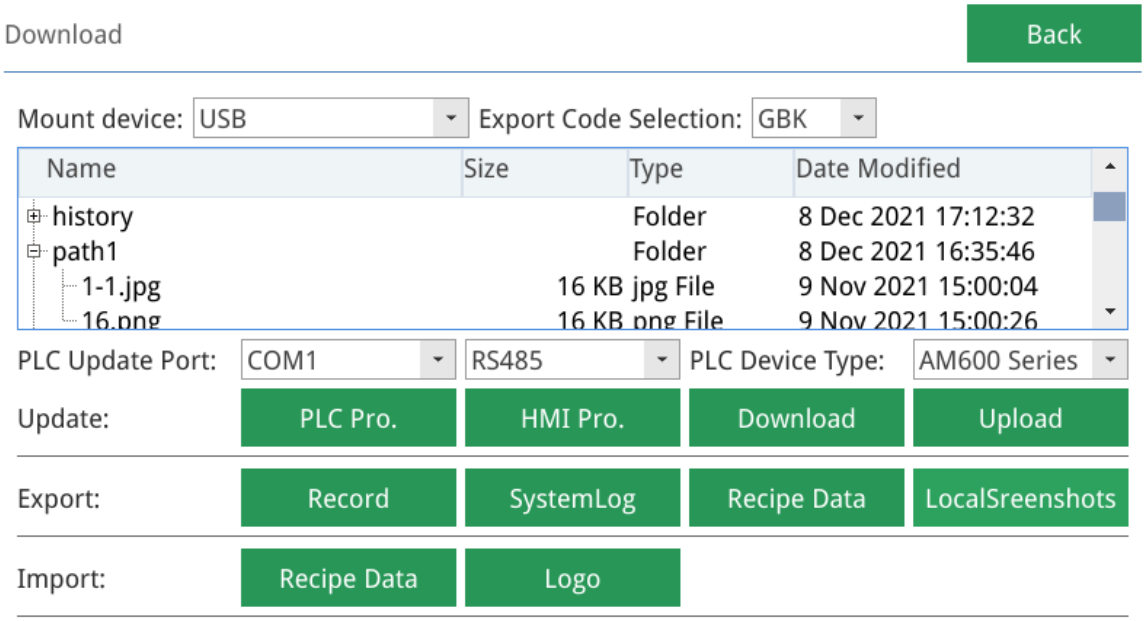


Click Yes to start export. After the progress bar reaches 100%, you will be prompted if the export is successful or failed. If the user did not select an export path, it is exported to the root directory by default.

16.3.13 Importing a Startup Screen

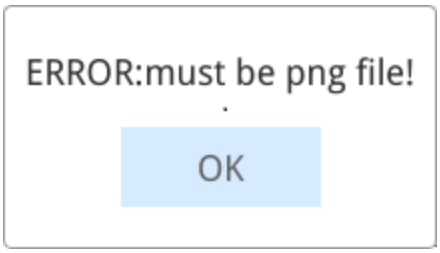
If you want to change the startup screen of the HMI, you can import your own picture as the startup screen as follows:

Insert a USB drive or SD card into the HMI, enter the download page of the control panel, and select the image to be imported (only PNG format is supported), as shown in the following figure:



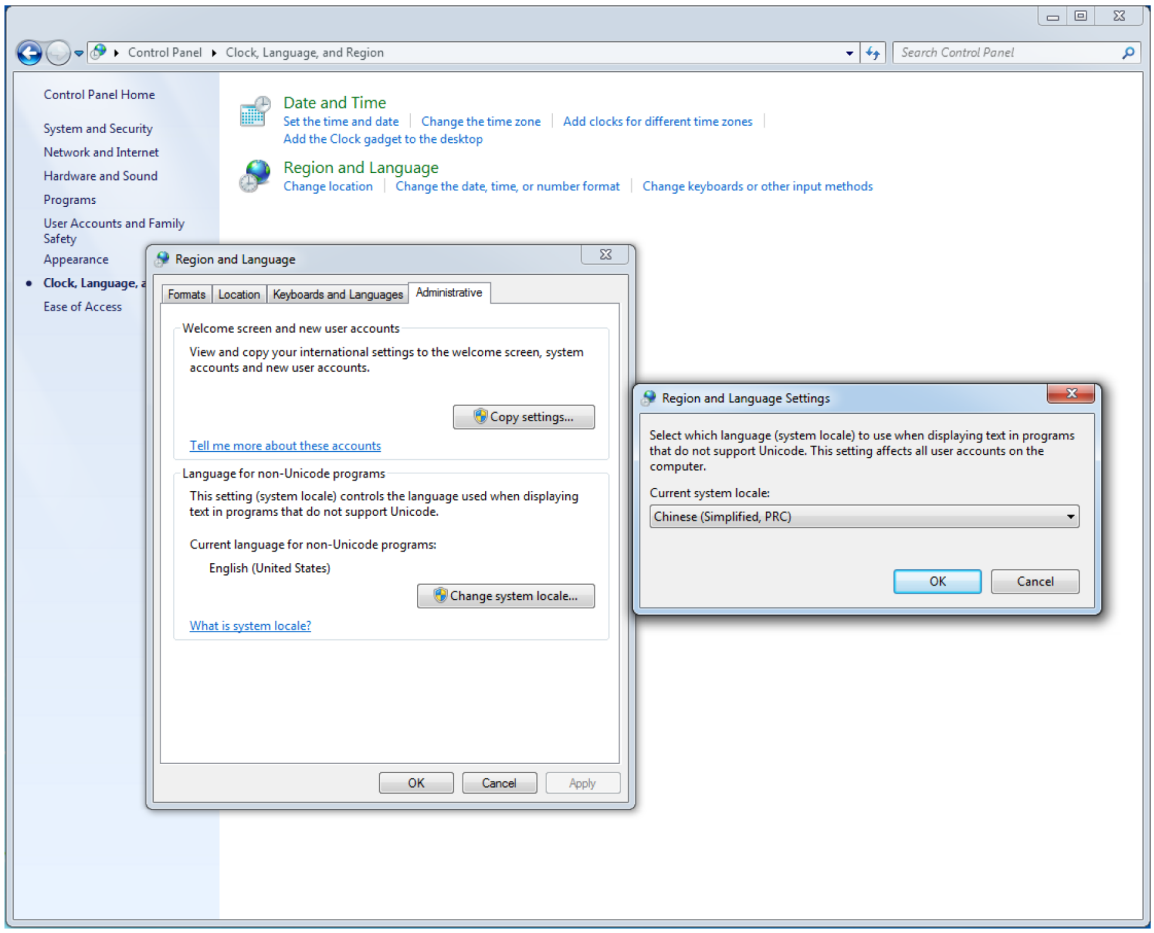
Then click the Startup Screen button in the Import field to import the startup screen. A popup window will prompt you if the import is successful or failed.

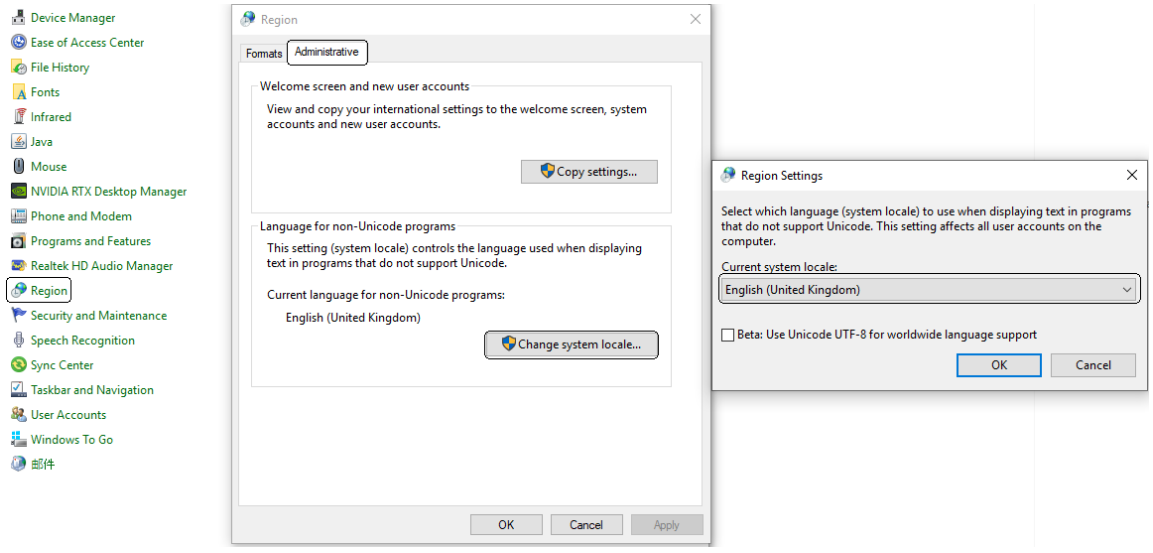
If you select a picture in a format other than PNG, the following prompt will pop up:



16.3.14 Important Notes about Import and Export of Chinese Characters

In the software tool, you can export the file as UTF8 or ANSI encoded, but in a Chinese Windows OS, you should only choose ANSI (that is, GBK). Because when you open the exported file with Excel, UTF8 encoded Chinese characters become gibberish. If you perform import and export on an English Windows OS, you must change the time zone to China. Otherwise, the exported Chinese characters become gibberish. Because the default encoding of the system is ASCII, which does not support Chinese characters. Even if you export the file as UTF-8 encoded, it is still saved as ASCII encoded after you edit it with Excel. The steps of setting China time zone in an English system are as follows:





When importing and exporting edited text, special symbols are supported, including: /, //, ', '', [, /[,], /], %, /%, &, /&, _, /_, (, /(,), /), "", <.

16.4 Printing

16.4.1 InoTouch Spooler

16.4.1.1 Printable Items

The printer can print the following:

1. The current screen.
2. Alarms in the Alarm view.
3. Recipe data in the Recipe view.
4. Data records in the Data view.

16.4.1.2 Requirements on the Printer

- With USB ports.
- HP LaserJet series printers (HP LaserJet 1000, HP LaserJet 1005, HP LaserJet 1018, HP LaserJet 1022nw, HP LaserJet 1022n, HP LaserJet 1020, HP LaserJet 1022) are recommended.
- Support for A4 paper.

16.4.1.3 Printing Format

You can print information in reports and images.

1. Report printing includes:

   a. Alarms in the Alarm view.
   b. Recipe data in the Recipe view.
   c. Data records in the Data view.

   Printing format:

   For the top line of the header, such as "Alarm", use 16 point regular font formatting.

   For the second line of the header, such as the recipe name or log name, use 14 point regular font formatting.

   For the attribute names of the recipe (such as Name, Path, Entry name, Value), use 10 point non-bold italic font formating; for the value of the recipe attribute name, use 10 point regular font formatting.

   Only the Droid Sans Fallback font can be used.

2. Image printing: You can print the current screen.

16.4.1.4 Report Printing

Printing alarm view data

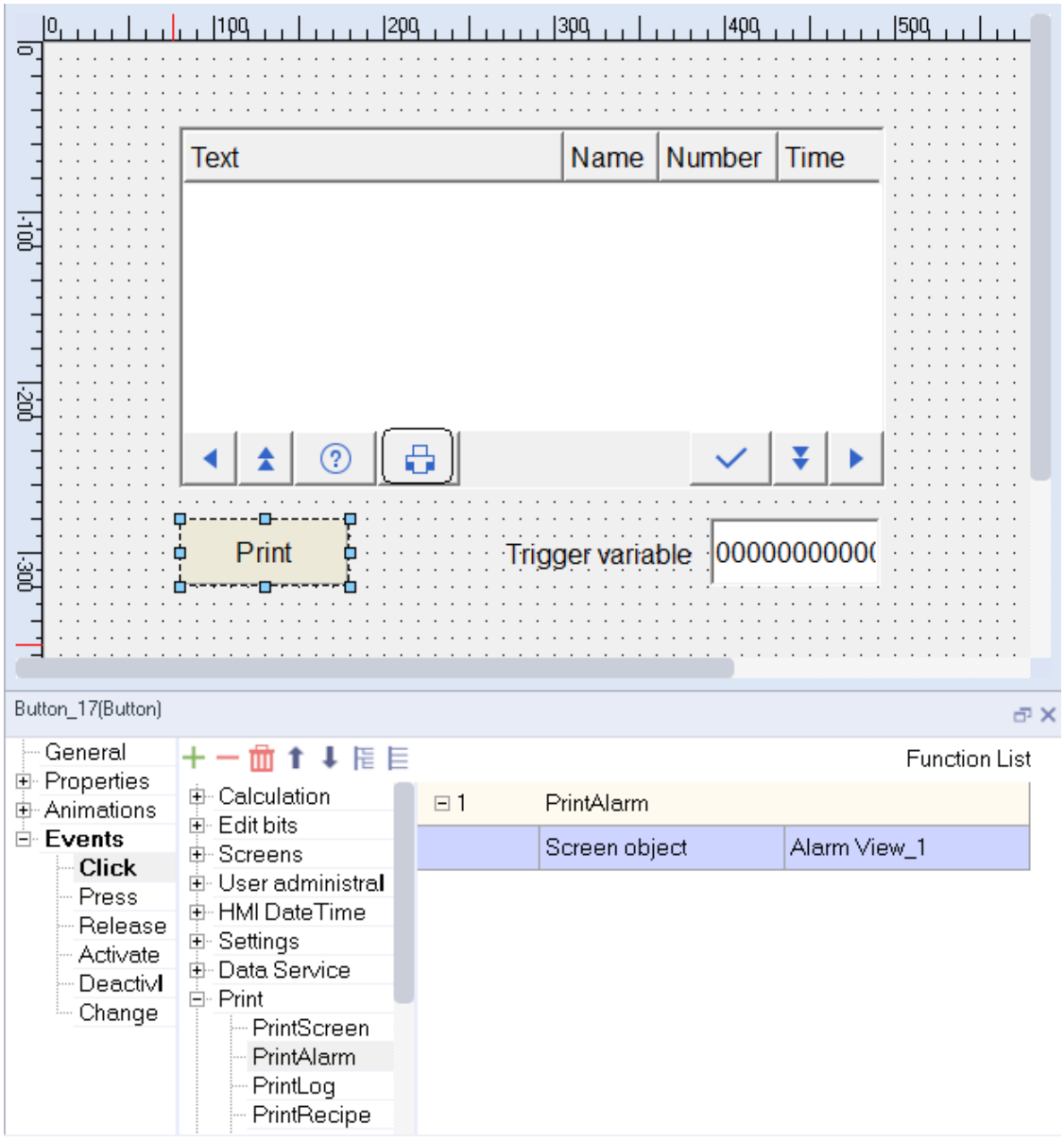
You can use the system function PrintAlarm to print an alarm view in the screen.

It prints the alarm information or alarm record of the alarm view, starting from the latest alarm or the line where the cursor is.

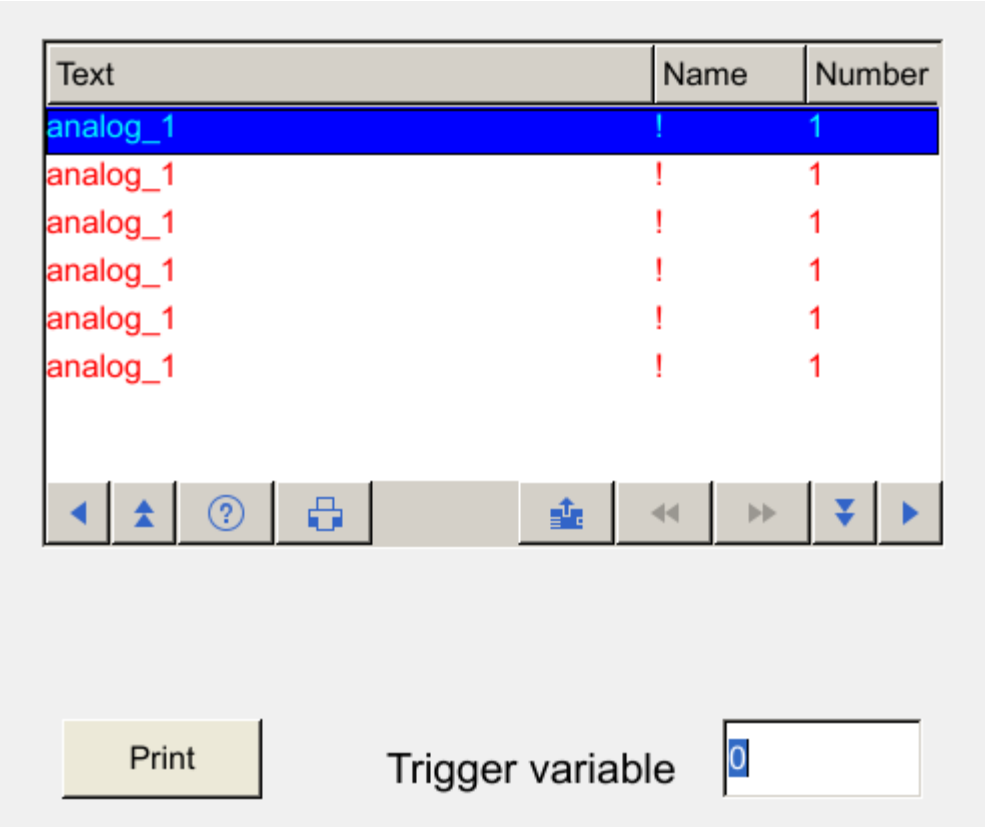
Since a USB drive must use the same USB port as the printer, you can only print the alarm record stored in a USB drive by simulation.

During simulation, if the computer is not connected with a printer, you can print the current alarm information and alarm records with the virtual printer and store them in a directory in the form of printing files, and the printing format is basically the same as that printed by the target device.

1. Create a new project, and configure an alarm view, a print button and numeric IO field in the screen. You can also use the print button on the alarm view, as shown in the figure:



2. The simulation is as follows:



3. The printing view is as follows:

14	RowCount	Text	Number	Time	Date	State	Category	ClassId	GroupId	ConnectName
15	1	analog_1	1	17:29:07	2021/12/8	1	1	1	-1	Connect_1
16	2	analog_1	1	17:29:16	2021/12/8	0	1	1	-1	Connect_1
17	3	analog_1	1	17:29:21	2021/12/8	1	1	1	-1	Connect_1
18	4	analog_1	1	17:29:51	2021/12/8	0	1	1	-1	Connect_1
19	5	analog_1	1	17:29:55	2021/12/8	1	1	1	-1	Connect_1
20	6	analog_1	1	17:31:08	2021/12/8	1	1	1	-1	Connect_1
21	7	analog_1	1	17:34:41	2021/12/8	0	1	1	-1	Connect_1

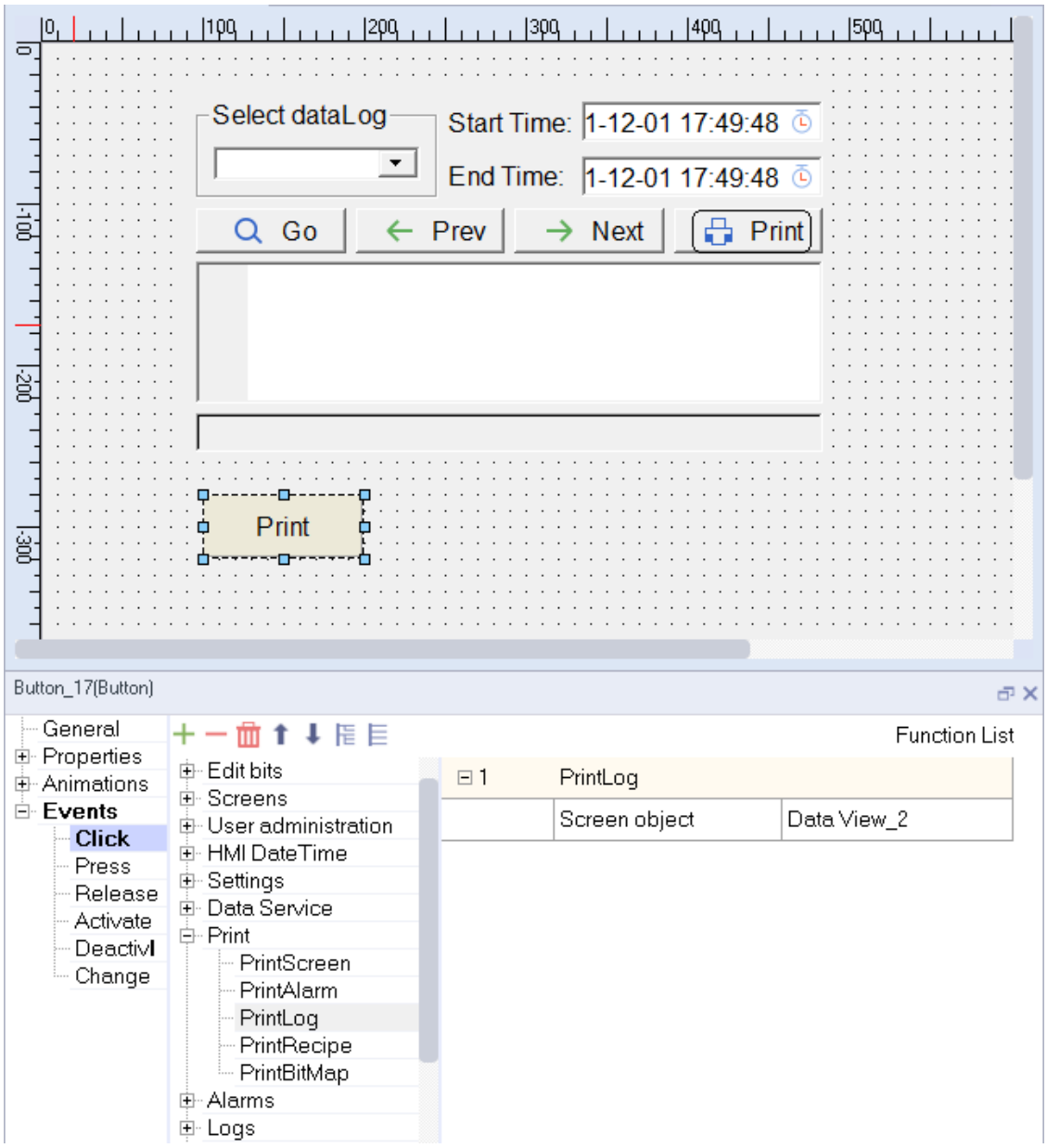
Printing data view data

You can use the system function PrintLog to print an data view in the screen.

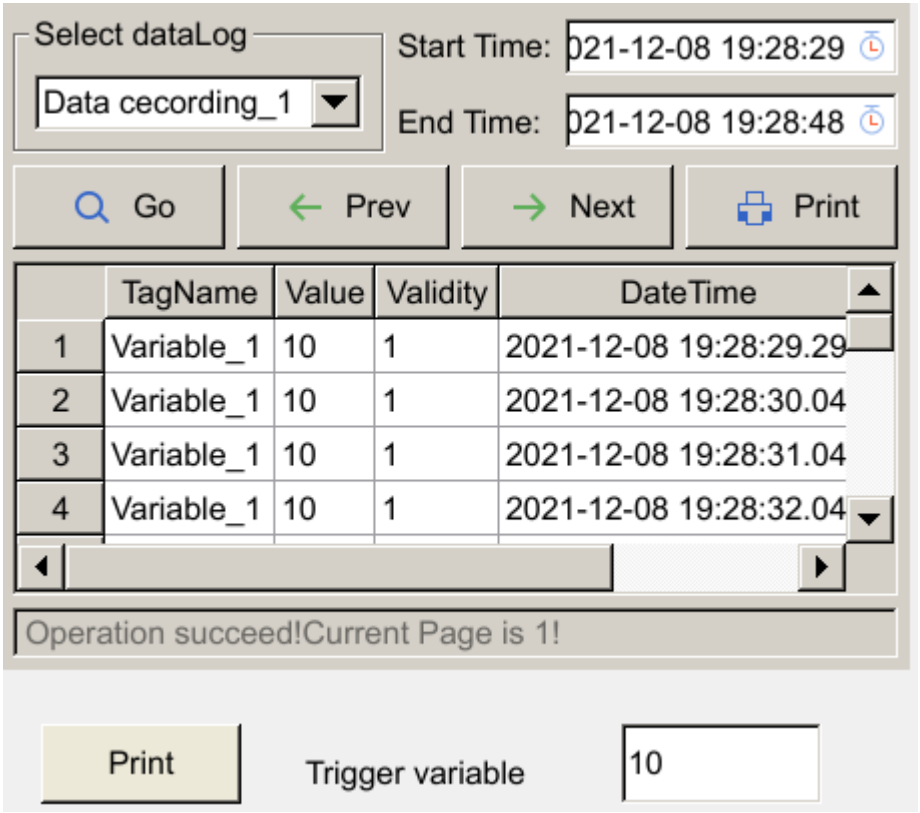
Since a USB drive must use the same USB port as the printer, you can only print the alarm record stored in a USB drive by simulation.

During simulation, if the computer is not connected with a printer, you can print the current data log with the virtual printer and store them in a directory in the form of printing files, and the printing format is basically the same as that printed by the target device.

1. Create a new project, and configure a data view, a print button and numeric IO field in the screen. You can also use the print button on the data view, as shown in the figure:



2. The simulation is as follows:



3. The printing view is as follows:

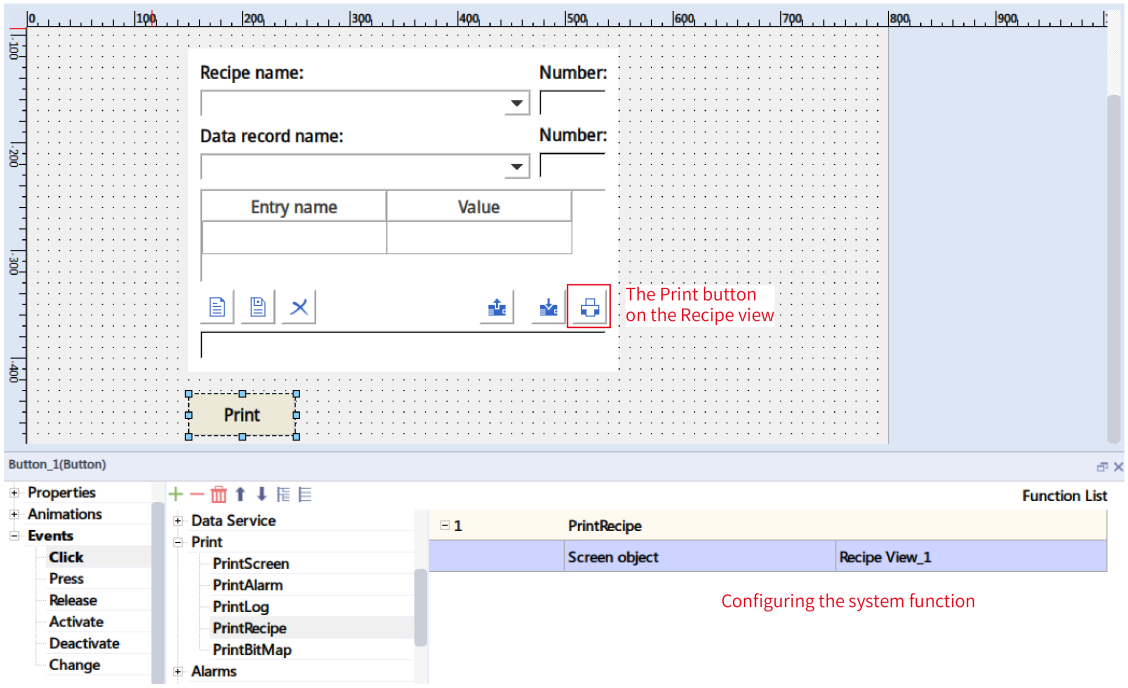
10	RowCount	TagName	Value	Validity	DateTime	ValId
11	1	Variable_1	10	1	2021/12/8 19:28	1
12	2	Variable_1	10	1	2021/12/8 19:28	1
13	3	Variable_1	10	1	2021/12/8 19:28	1
14	4	Variable_1	10	1	2021/12/8 19:28	1
15	5	Variable_1	10	1	2021/12/8 19:28	1
16	6	Variable_1	10	1	2021/12/8 19:28	1
17	7	Variable_1	10	1	2021/12/8 19:28	1
18	8	Variable_1	10	1	2021/12/8 19:28	1
19	9	Variable_1	10	1	2021/12/8 19:28	1
20	10	Variable_1	10	1	2021/12/8 19:28	1
21	11	Variable_1	10	1	2021/12/8 19:28	1
22	12	Variable_1	10	1	2021/12/8 19:28	1
23	13	Variable_1	10	1	2021/12/8 19:28	1
24	14	Variable_1	10	1	2021/12/8 19:28	1
25	15	Variable_1	0	1	2021/12/8 19:28	1
26	16	Variable_1	0	1	2021/12/8 19:28	1
27	17	Variable_1	10	1	2021/12/8 19:28	1
28	18	Variable_1	10	1	2021/12/8 19:28	1
29	19	Variable_1	10	1	2021/12/8 19:28	1
30	20	Variable_1	10	1	2021/12/8 19:28	1
31	21	Variable_1	10	1	2021/12/8 19:28	1
32	22	Variable_1	10	1	2021/12/8 19:28	1
33	23	Variable_1	10	1	2021/12/8 19:28	1
34	24	Variable_1	10	1	2021/12/8 19:28	1
35	25	Variable_1	10	1	2021/12/8 19:28	1
36	26	Variable_1	10	1	2021/12/8 19:28	1

Printing recipe view data

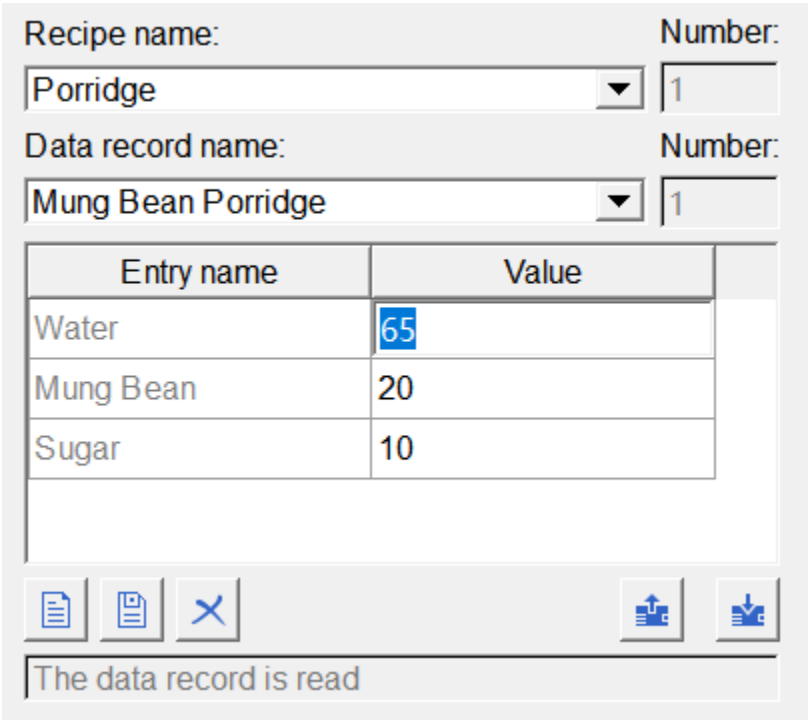
You can use the system function PrintRecipe to print an recipe view in the screen.

During simulation, if the computer is not connected with a printer, you can print the current recipe data with the virtual printer and store them in a directory in the form of printing files, and the printing format is basically the same as that printed by the target device.

1. Create a new project, and configure a recipe view and a print button. You can also use the print button on the recipe view, as shown in the figure:



2. The simulation is as follows:



3. The printing view is as follows:

	A	B
1	Recipe name	
2	Recipe_1	
3	Data record name	
4	Datarecord_1	
5	Entry name	Value
6	water	50
7	sugar	20
8	spice	30

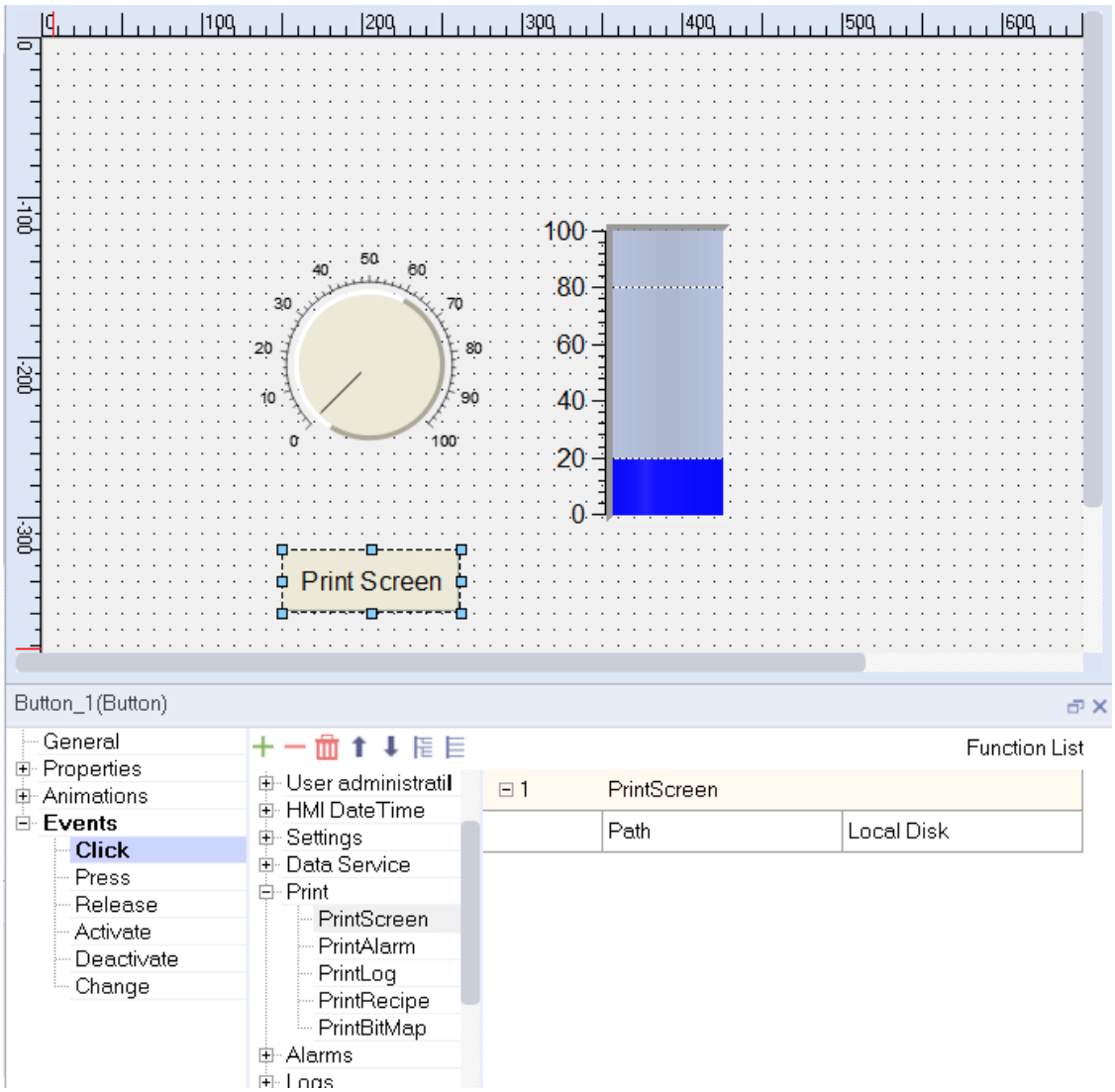
"Synchronize tags" and "Tags offline" in the above table are the properties of the recipe view. The values of the two properties are both True.

16.4.1.5 Image Printing

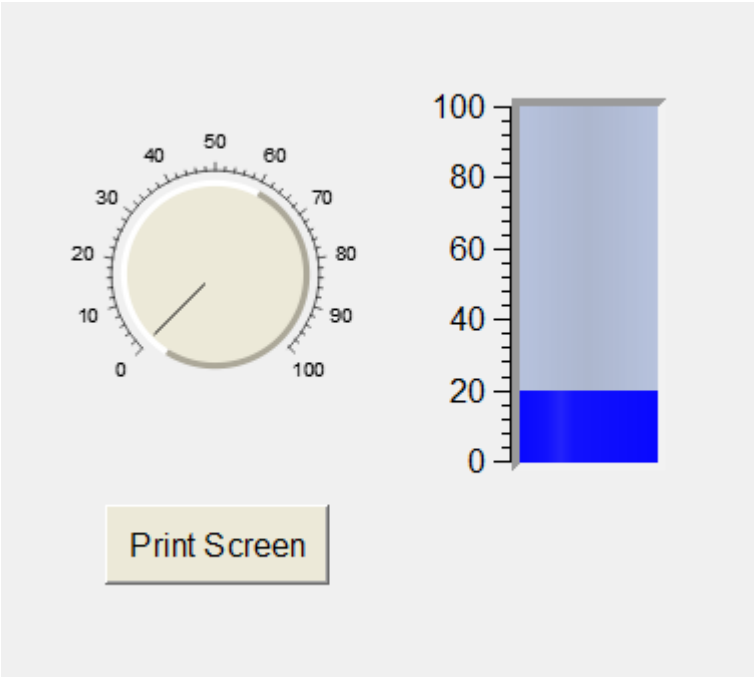
You can use the system function PrintScreen to print the current screen.

Title and time will not be added to the image; The print size of the image is adaptive to the width of the paper. In simulated printing, the printed image is stored in PNG format under the path C:\Users \<user name>\Documents\InoTouchPad\ ScreenShots, and the printing format is basically the same as that in the target machine.

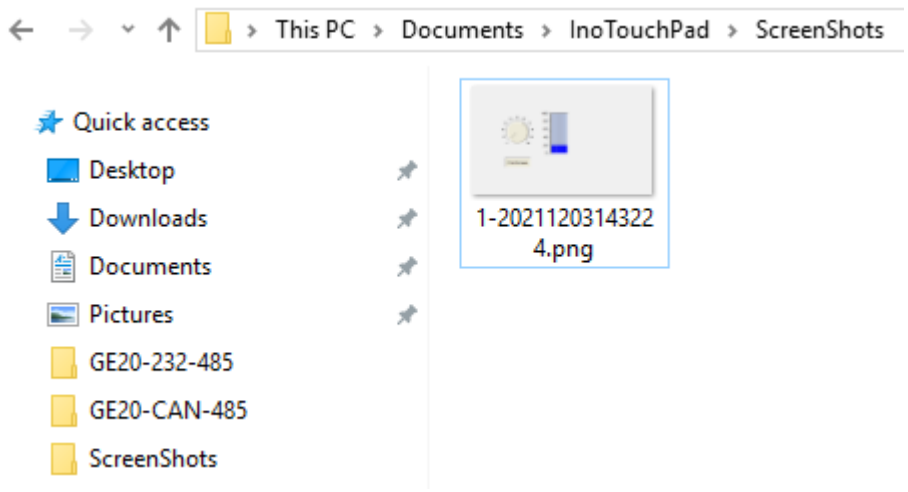
1. Create a project and configure a bar diagram, knob and a Print Screen button in the screen to print the current screen, and set the storage location to local HMI, as shown in the following figure:



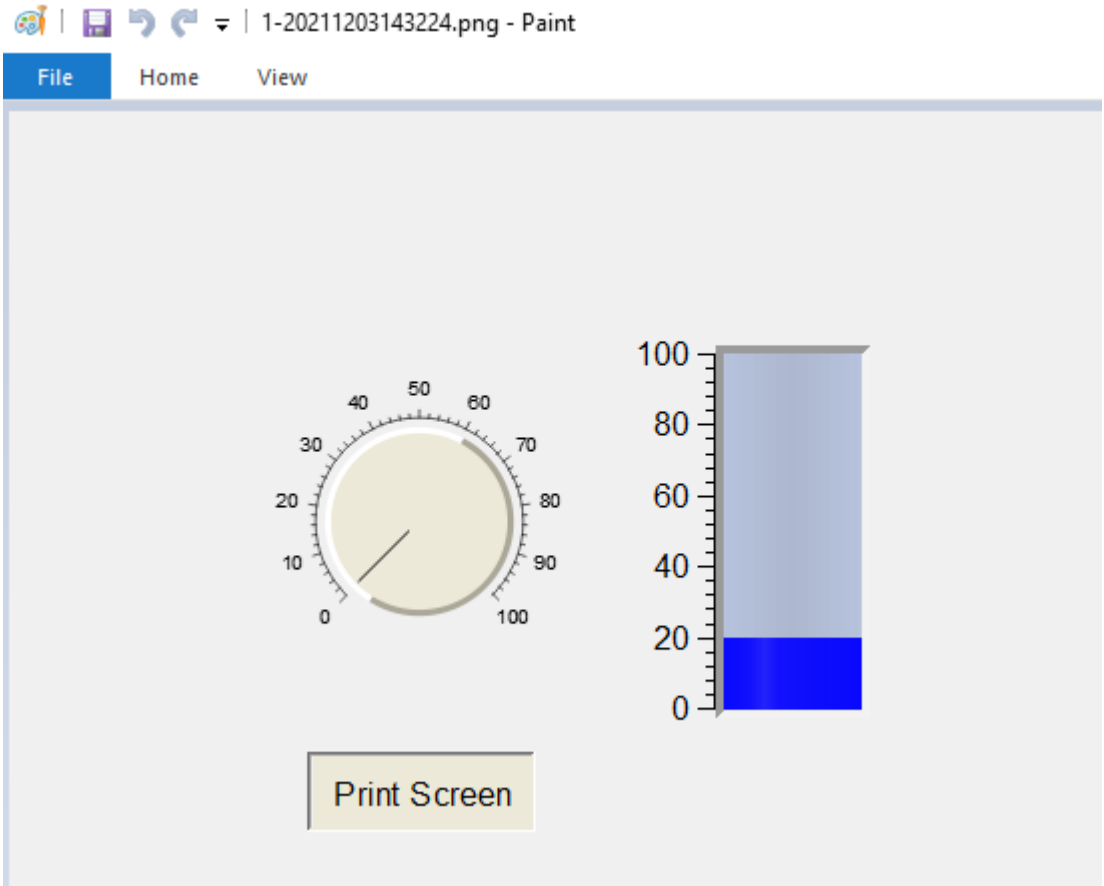
2. The simulation is as follows:



3. The printed image is saved in the following directory:



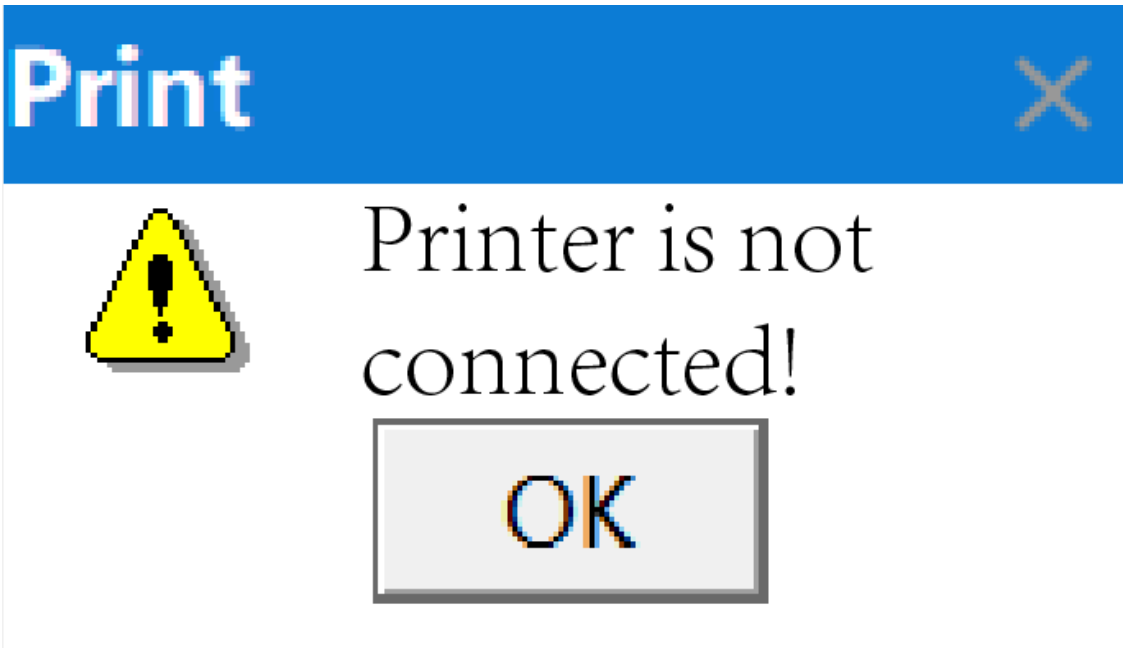
The screenshot is as follows:



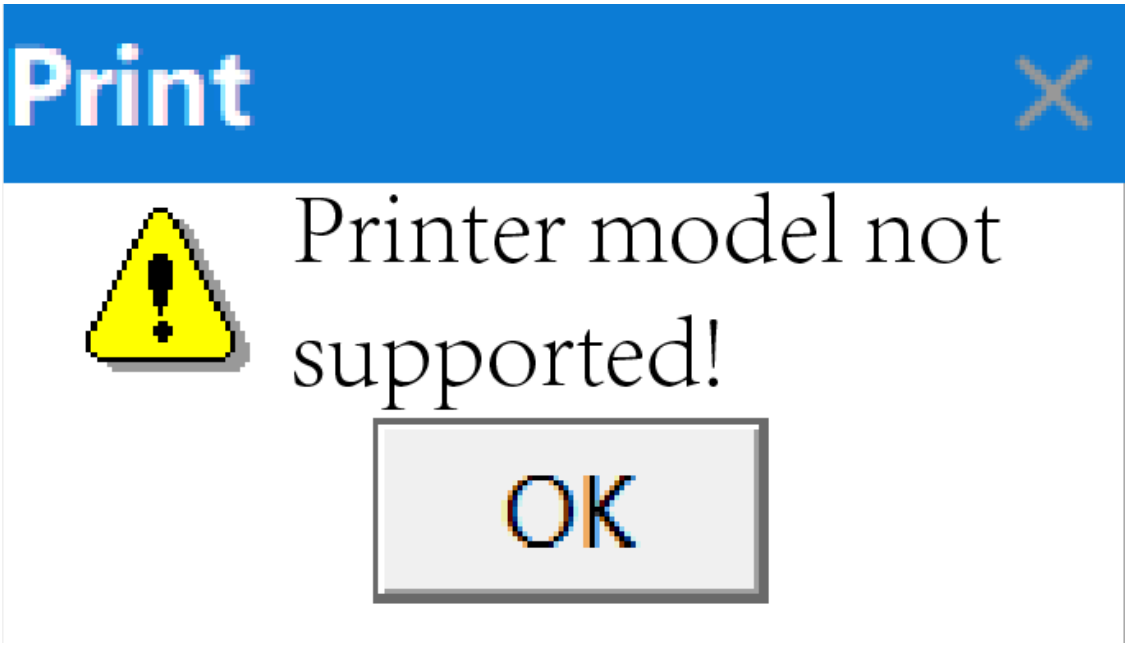
If the screen is printed on the target machine, the screenshot image is stored in the screenshots folder under the root directory of the storage medium (local drive, USB drive or SD card).

16.4.1.6 Messages

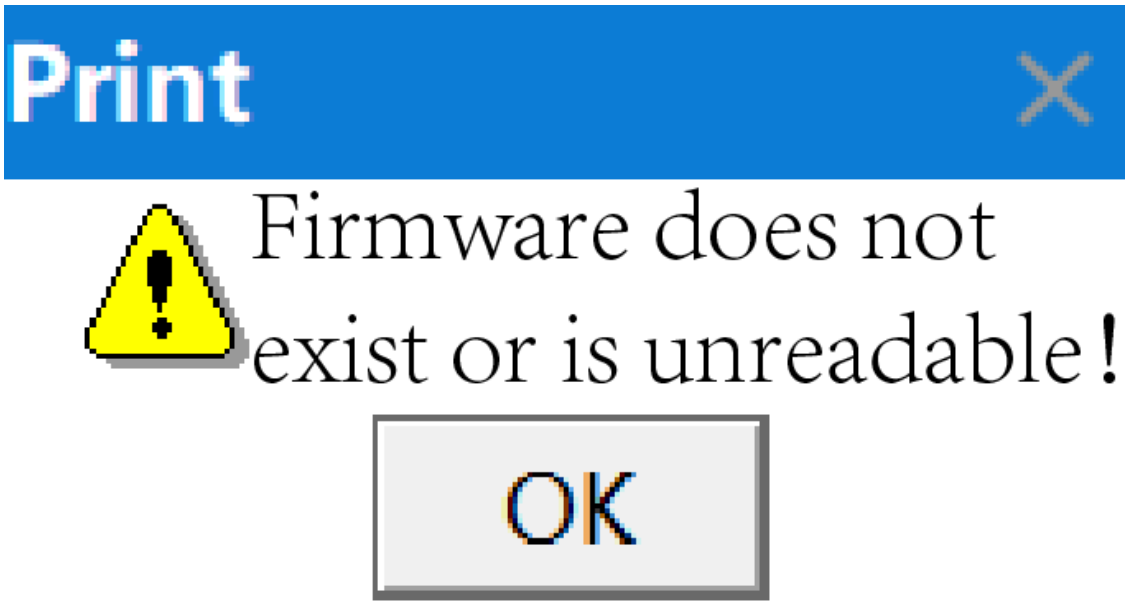
1. When you print a report, if the printer is not connected, the following window will pop up:



2. When you print a report, if the printer is incompatible, the following window will pop up:

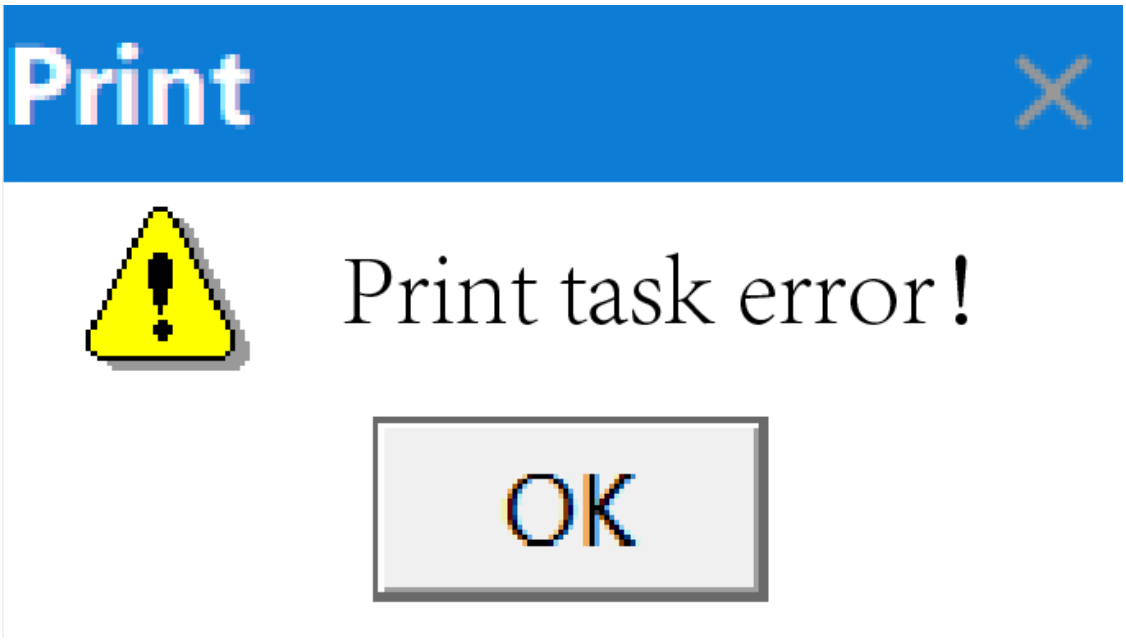


3. When you print a report, if access to the printer firmware failed, the following window will pop up:

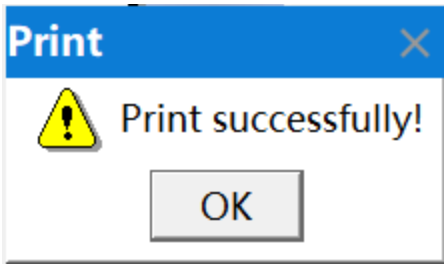


4. When you print a report, if nothing is provided for printing, the following window will pop up:

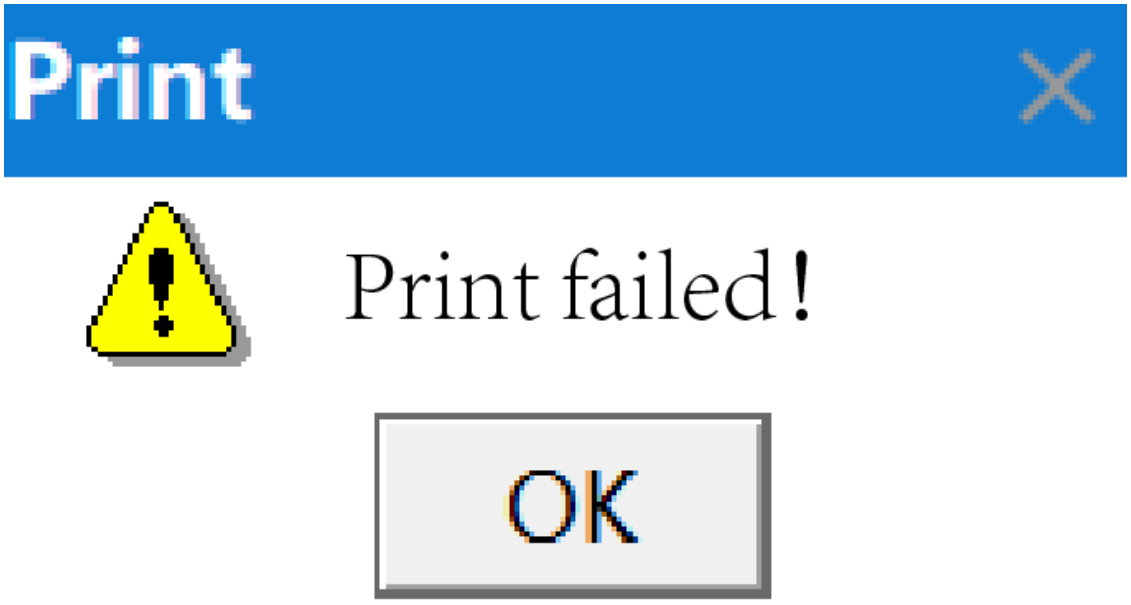
5. When you print a report, if the process is interrupted, the following window will pop up:



6. When you print an image, if the printing is successful, the following window will pop up:



7. When you print an image, if the printing failed, the following window will pop up:



16.4.2 HMI Screen Printing

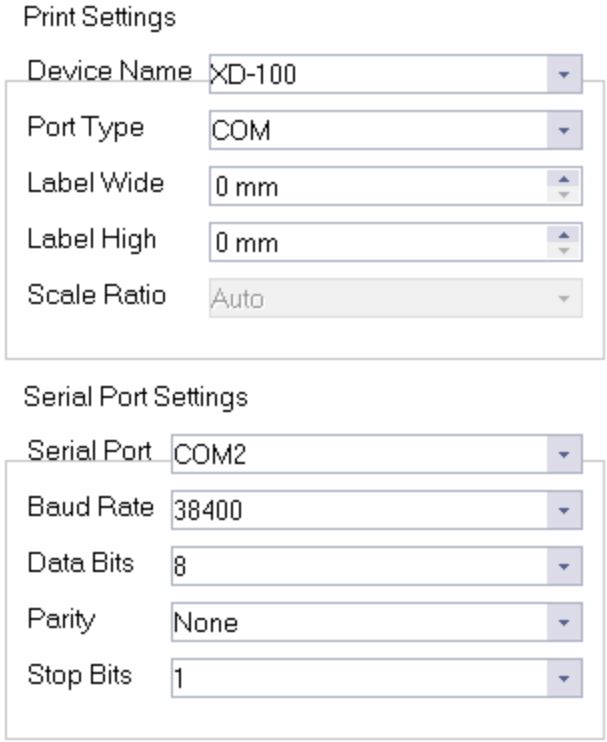
16.4.2.1 Printing Mode

To print QR codes and bar codes, IT7000 allows you take a screenshot of the screen and print it. You can configure a QR code output control in the screen, set the printing function according to the screen coordinates of the control, and then set the printing mode. Note that the screenshot is taken from the current screen. If you switch the screen, the printing result will not be what you want.

1. Auto printing: The printing function is bound to a tag and is triggered through the PLC.
2. Manual printing: The printing function is bound to a button in the screen, and you can press the button to trigger the printing.

16.4.2.2 Configuration of the Printer

On the left tree view of the software, click HMI Settings→Printer Settings. As shown in the following figure.

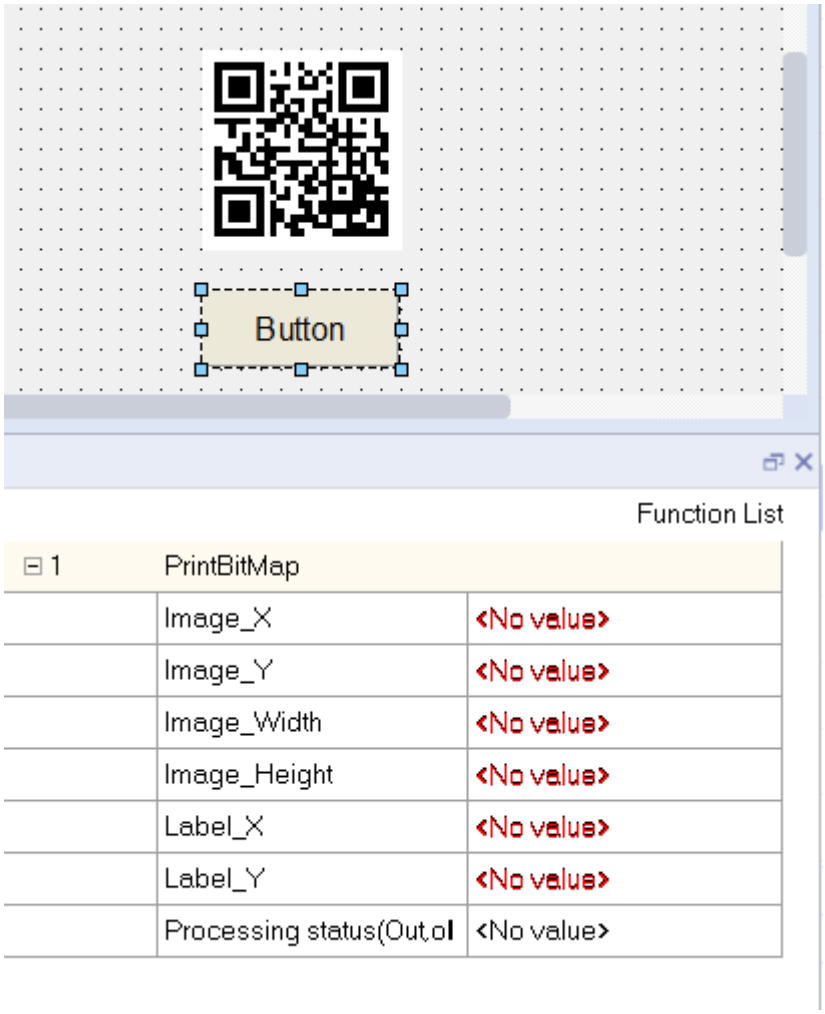


Attribute	Description
Device Name	Supported printer
Port type	Printer connection method
Label Width	The width of the label paper
Label Height	The height of the label paper
Scale ratio	The screen size to be printed will be scaled according to the size of label paper, which is not enabled by default
Serial Port	The serial port you want to use
Baud rate	Baud rate: 4800/9600/19200/38400/57600/115200
Data Bits	Data bit: 7 or 8

Attribute	Description
Parity	Parity check type: none/odd/even
Stop Bits	Stop bit: 1 or 2

16.4.2.3 Print Function

As shown in the following figure, the button is bound to the printing system function PrintBitMap. For details, see *"12.3.9 Printing" on page 257*.

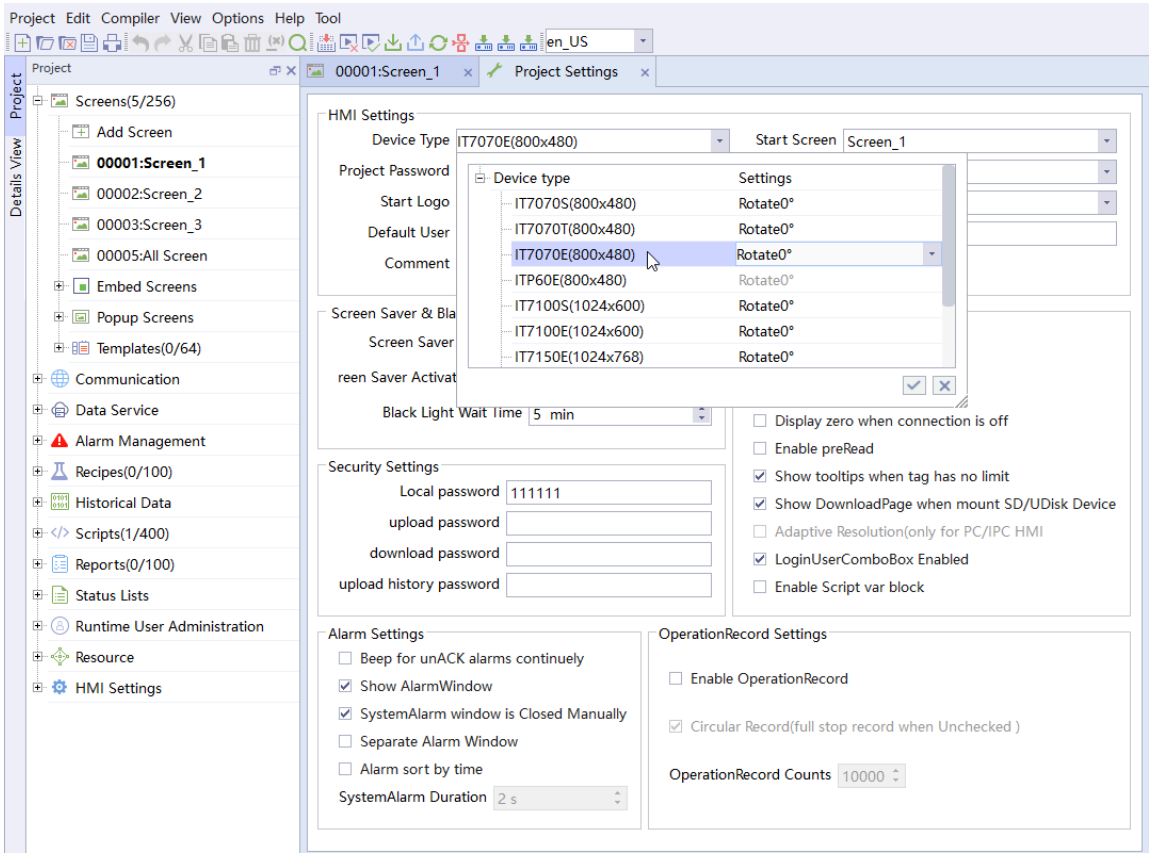


16.5 Setting and Application of SCADA

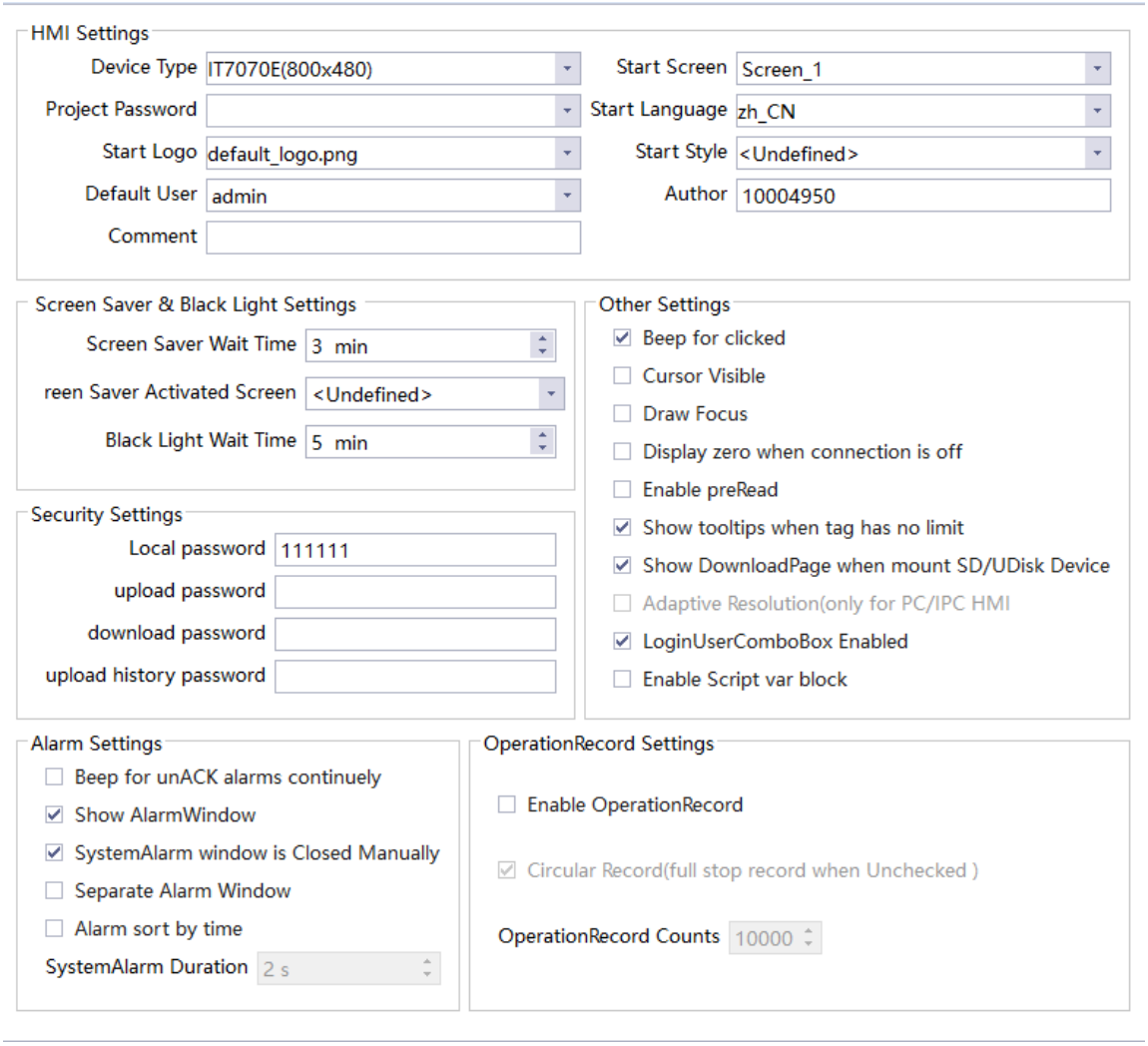
The SCADA system is a computer-based DCS and power automated monitoring system. It is widely used in fields like electric power, metallurgy, petroleum, chemical industry, gas, and railway, for data acquisition, supervision, and process control. InoTouchPad runs on a PC and can used as a SCADA system application.

The following steps instructs you to configure a SCADA system with InoTouchPad:

First, go to HMI Settings→Project Settings, and set the device type as ITPC (800x600), that is, to configure a runtime project running on a PC.



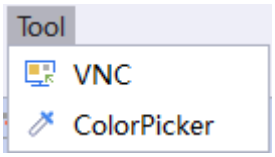
Then, in Other Settings, check Adaptive Resolution (only for PC/IPC HMI), so that the configured runtime can be displayed in full screen on the PC.



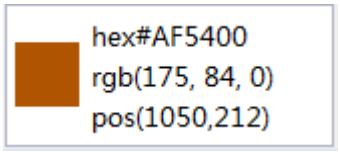
After that, you can configure the functions required by users. After the project is completed, the runtime runs on the PC and the PC is connected with field devices through buses, forming a SCADA system with which you can monitors and controls field devices on the PC.

16.6 Color Picker

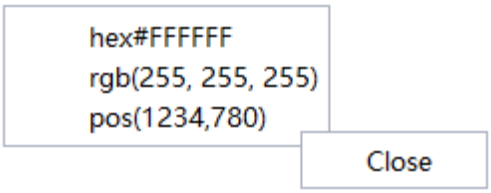
The color picker can obtain the screen coordinates and color value of the position where the cursor is. You can start it from the toolbar.



As shown in the following figure, a floating window will show when the tool is running.



Right-clicking this window will pop up the Close button.



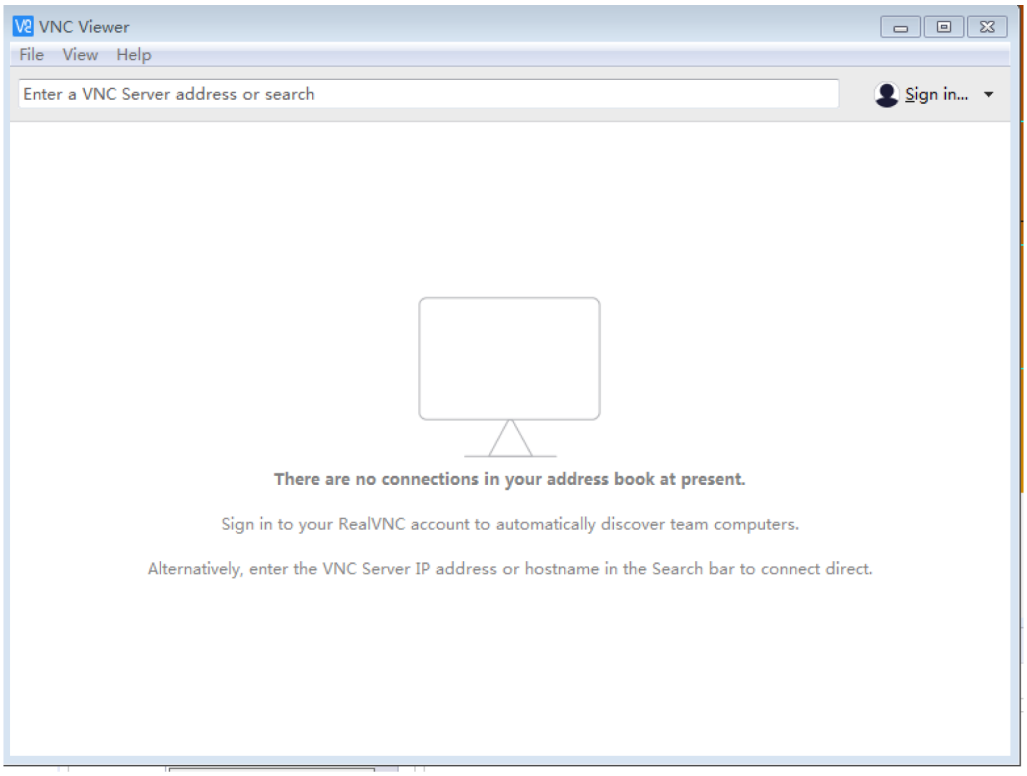
Parameter	Description
Hex#AF5400	HEX value of the color, which is converted from the RGB value
Rgb(175,84,0)	RGB value of the color
Pos(1050，212)	Screen coordinates of the position where the cursor is

16.7 VNC Tool

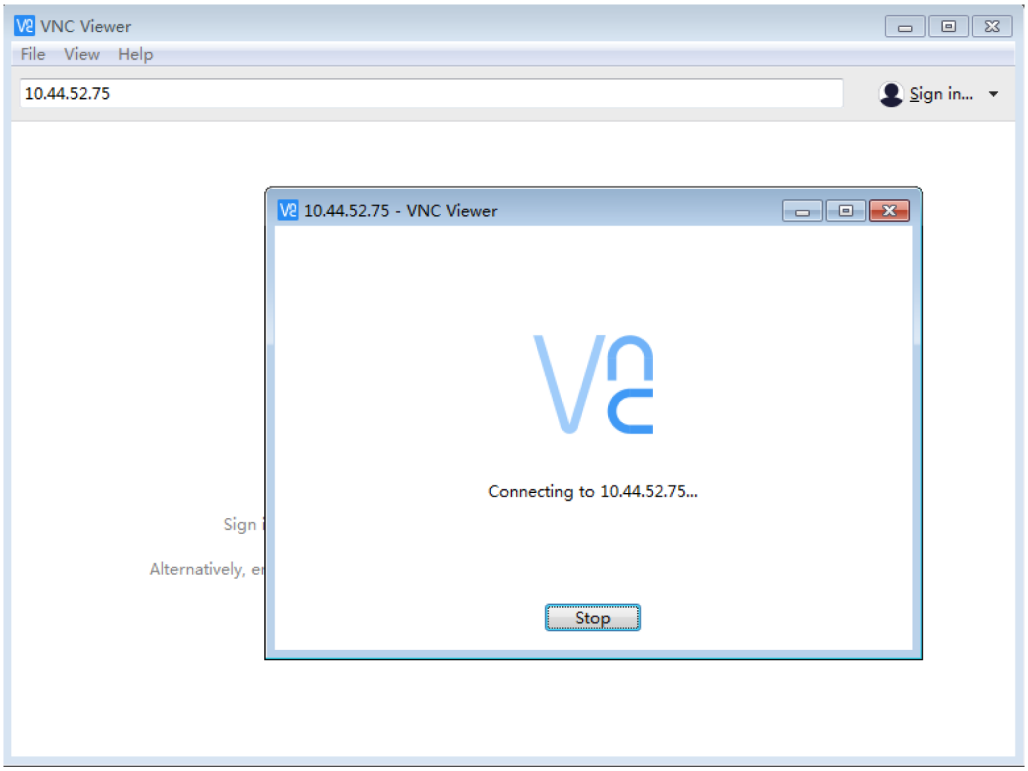
IT7000 comes with a VNC tool to help you monitor the HMI easily. In the same LAN, you can log in to the IP address of an HMI through a PC or mobile device, and then view the screen of the HMI and control it.

To use the VNC tool, follow the steps as follows:

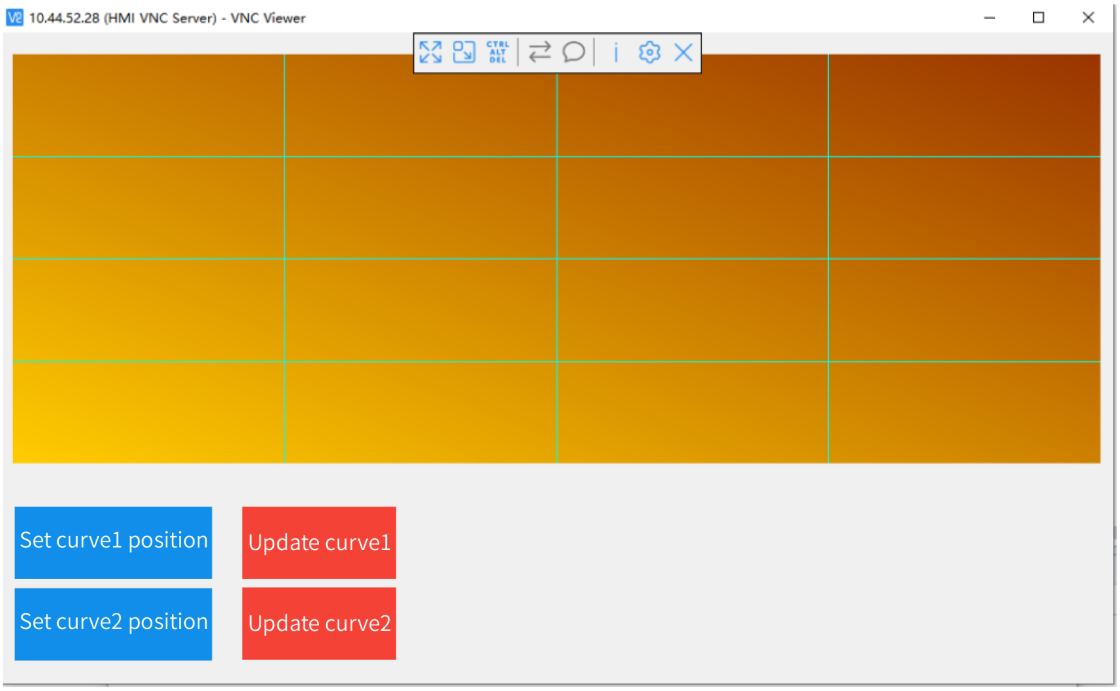
1. Click on the toolbar of the IT7000 software tool, select VNC, and the following window pops up.



2. Ensure that the HMI in the same LAN as the PC. Enter the IP address of the HMI to be monitored in the input box, and then press Enter, as shown in the following figure.



3. After logging into the HMI, you can view the screen and click on the screen for control. Click this button to enter full screen. The window size is adapted to the screen size by default. You can unlock the window by the button and change its size through dragging.



16.8 Global Style

You can use the css-based global style to customize the user interface. Go to Resource→Global Style, and you can customize the style of buttons and interfaces.

	名称	编号	注释
1	style1	1	
2	style2	2	
3	style3	3	

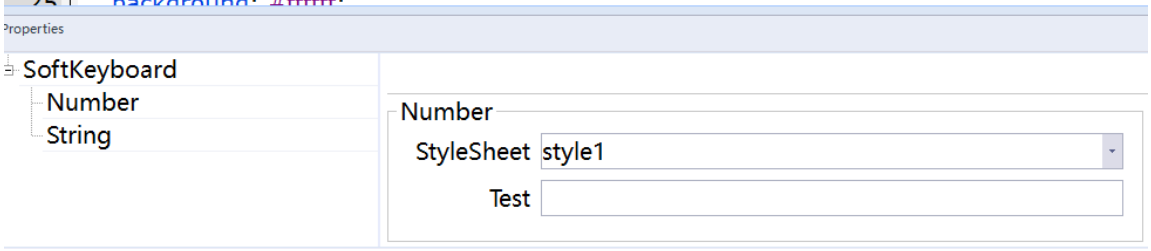
```
    border: 1px solid #adb1c2;
}

/*----------------IO style----------*/
QSSLineEdit{
    border: 1px solid #adb1c2;
}

QSSLineEdit:hover,QSSLineEdit:focus{
    background: #fafafa;
    border: 1px solid #a5c8ff;
}

QSSLineEdit:pressed{
    background: #f8f8f8;
    border: 1px solid #a5c8ff;
}

QSSLineEdit:disabled{
    background: #f0f0f0;
    border: 1px solid #dedede;
    color:#999999;
}


/*----------------button style----------*/
QSSPushButton, QSSButton{
    border:1px solid #adb1c2;
    background:#f0f4f1;
}

QSSPushButton:hover, QSSButton:hover{
    border:1px solid #699ddd;
    background:#ffffff;
}

QSSPushButton:checked, QSSPushButton:focus, QSSPushButton:pressed,
QSSButton:checked, QSSButton:focus, QSSButton:pressed{
    border:1px solid #699ddd;
    background:#ccd4ff;
}

QSSPushButton:disabled, QSSButton:disabled{
    border:1px solid #adb1c2;
    background:#f0f0f0;
}

/*----------------text style----------*/
QSSLabel:pressed{
    color:#1e1e1e;
    background: transparent;
}
```

The software provides three default styles: Style1, Style2 and Style3, which can be edited and duplicated. The styles can be switched through variable configuration and scripting.

16.9 Soft Keypad

Similar to *"16.8 Global Style" on page 418*, the soft keypad can also be styled.

```
NumberInputPanel {
    background: #a8b0bc;
}

/*-----------------Context book display-------------*/
QLabel#limitLabel {
    border: 0px;
    margin: 0px;
    background: #ffffdc;
    font:16px;
}

/*-----------------Input box-------------*/
QLineEdit#inputEdit {
    border: 0px;
    margin: 0px;
    background: #ffffff;
    font:21px;
}

/*-----------------Ordinary button-------------*/
KeyButton {
    border: 0px;
    margin: 0px;
    background: #ffffff;
```



There are three default styles:

You can use css to develop your own style.

16.10 HMI Control Panel

After starting the IT7000 HMI, long press the screen to enter the control panel. You can use the control panel to maintain the HMI. Major functions:

1. Time/Date: You can view and set the local time of the HMI.

2. Security: You can set the local password, upload password, download password and upload history password.

   a. The local password is used to access the setting page locally.
   b. The upload password is required when you upload a project.
   c. The download password is required when you download a project or firmware.
   d. The upload history password is used to access historical upload data.

3. Backlight: You can set the backlight brightness of the HMI. The default value is 90.
4. Buzzer: You can enable/disable button press sound and continuous beep alarm for an unacknowledged alarm.
5. Download: When a USB drive or SD card is connected with the HMI, you can press the Download icon to manage project files. Generally, it takes about 2 seconds to recognize an inserted external device. If the project page is active, the download page will pop up automatically. The download page is described as follows:

   a. Upgrade

Item	Description
PLC program	You can update PLC programs
HMI program	You can upgrade HMI user projects and firmware. Note that the file name of the user project must be project.zip and that of the firmware must be firmware.zip.
Gcode	You can d upgrade the PLC program in Gcode

   b. Export: You can export system logs, recipe data, local screenshots, and logs (data logs, alarm logs, and operation logs). For details, see section 16.3 Import and Export.
   c. Import: You can import recipe data and boot screen. The imported boot screen image will be automatically scaled to the resolution of the screen.

6. Factory reset: This restores the HMI, including the firmware and user projects, to the original state, and clears all user files and log files. The buzzer will beep at the beginning of reset, and will long beep again after reset is completed.
7. Start screen: You can set the start screen of a user project.
8. Language: You can set the display language of the HMI, including Simplified Chinese, Traditional Chinese and English.
9. Calibrate: You can calibrate the screen in the Calibrate page.
10. Screen saver: You can set the screen saver timeout which ranges from 0 to 255 minutes. 0 means the screen saver is disabled, and 3 is the default value. You can set the the backlight timeout which ranges from 0 to 255 minutes. 0 means the backlight is disabled, and 5 is the default value.
11. Network: You can view and set the IP address of the HMI. The IP address can be obtained automatically or static. When the IP address is set to be obtained automatically, the HMI obtains the

current network environment information and assigns an available IP address. Otherwise, you can set a fixed IP address as needed.

12.  Restart: Restarts the HMI.

16.11 Status List

16.11.1 Overview

The status list is divided into text list and graphic list, which is used to make controls and options more recognizable. For example, for the button for controlling motors, "OFF" or "ON" is more readable than "0" or "1". The status list associates the control with the corresponding list. When the control performs a switching action and the state changes, if the variable value has been assigned a specific entry, the control will be displayed as readable text or graphics.

16.11.2 Text List

The text list can be configured in the control symbol IO field/button. Example:

1. Bit (0,1): True when tag value is 1, and false when tag value is 0;

Text lists

	Name	Number	Selection
1	TextList_1	1	Bit (0, 1)
2	Text list_2	2	Bit number (...
3	Text list_3	3	Range (... - ...)

List entries(TextList_1)

	Number	Value	Entry
1	8	0	false
2	9	1	true

2. Bit number (0-31): Use the 0-31 bits of the tag, and display entries with a value of 0 when the 0-th bit of the tag is 1 and only the 0-th bit is 1, and so on.

Text lists

	Name	Number	Selection
1	TextList_1	1	Bit (0, 1)
2	Text list_2	2	Bit number (...
3	Text list_3	3	Range (... - ...)

List entries(Text list_2)

	Number	Value	Entry
1	1	0	a
2	5	1	b
3	6	2	c
4	7	3	d

3. Range (...-...): Displays the corresponding entry when the range of tag value falls within the range.

Text lists

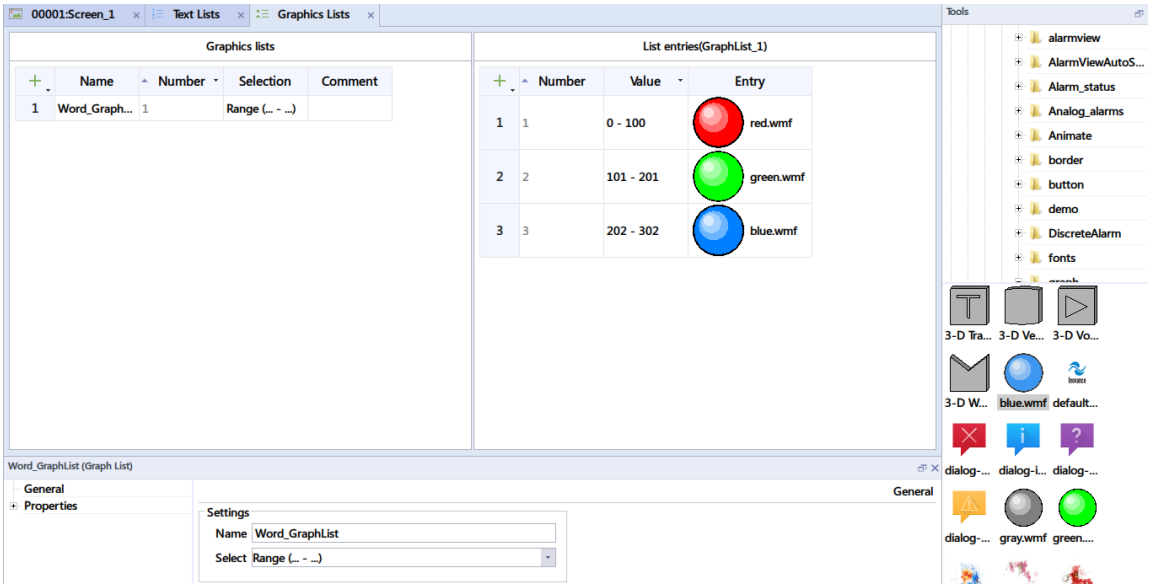
	Name	Number	Selection
1	TextList_1	1	Bit (0, 1)
2	Text list_2	2	Bit number (...
3	Text list_3	3	Range (... - ...)

List entries(Text list_3)

	Number	Value	Entry
1	2	0 - 59	Failure
2	3	60 - 79	Good
3	4	80 - 100	Excellent

16.11.3 Graphic List

The graphic list can be configured in the control graphic IO field/button. For the usage of the items "selected" in the list, refer to the above text list. In the following figure, the range (...-...) is configured, and you can drag items from the gallery on the right into the table entry.

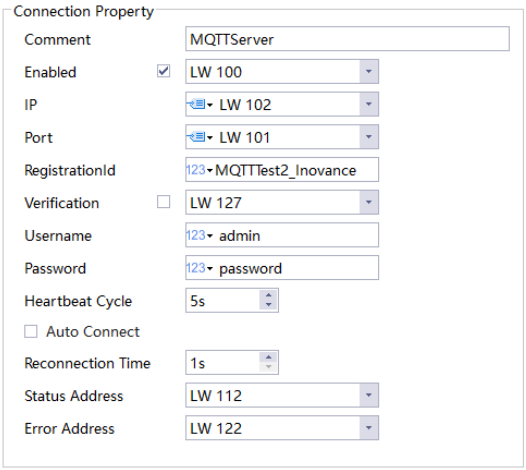


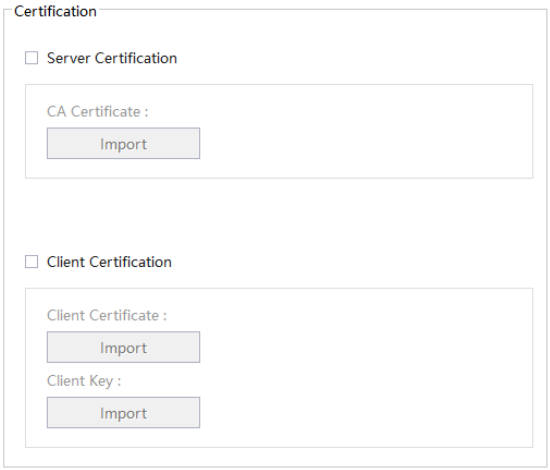
17 Data Service

17.1 MQTT

17.1.1 MQTT Server

MQTT, fully known as Message Queuing Telemetry Transport, is a lightweight, publish-subscribe network protocol that transports messages between devices. MQTT can be interpreted as a low overhead instant messaging protocol with low bandwidth occupation. It can provide real-time reliable message service for connecting remote devices with a small code footprint and minimal network bandwidth. It is suitable for remote devices with low hardware performance and bad network environment. Therefore, MQTT protocol has a wide range of applications in IoT (Internet of Things), small device applications, and mobile applications. At present, the HMI is only used as a client, which can subscribe or publish JSON data. The HMI can be both a subscriber and a publisher. The topic, which is the key for MQTT, is a tag that connects to an application message and matches a subscription on the sever. The server sends message to each client that subscribes to the matching tag.
When the HMI is used as a client connecting to the specified server, the settings are as follows:





1. Description of settings

Name	Description	Limits	Supported Data Types
Comment	Description of the configuration	128 characters	/
Enabled	Enables/disables MQTT protocol connection		Int16,UInt16,Int32,UInt32 0: Disabled non-0: Enabled
IP	MQTT server IP	IP or domain name, up to 256 characters, automatically recognized	String
Port	MQTT server port number		Int16,UInt16,Int32,UInt32
Registration ID	The name of the HMI when it logs on to the server	Up to 256 characters	String

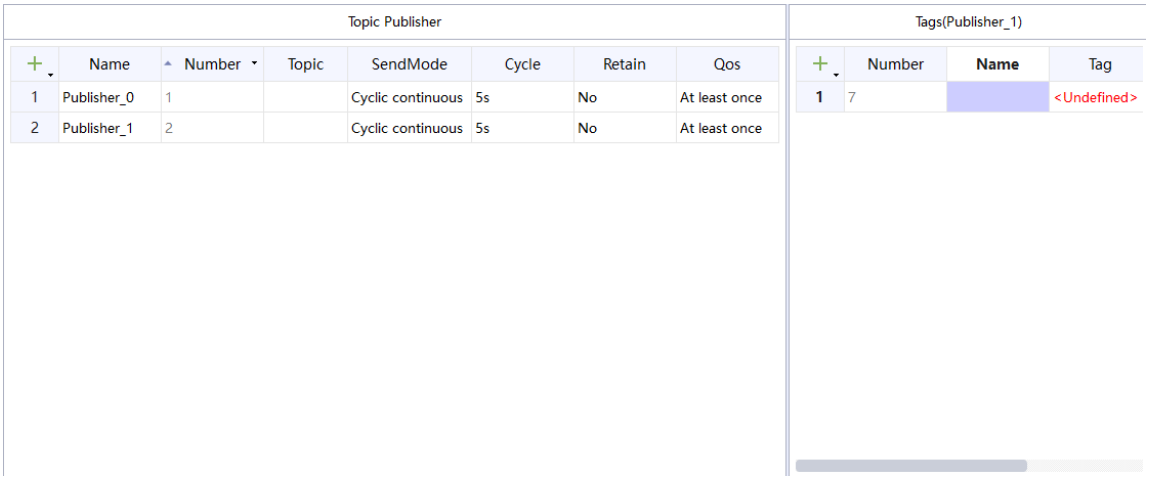
Name	Description	Limits	Supported Data Types
Verification	Enables logon verification	If you have enabled logon verification on the server, you must also enable verification on the HMI. Otherwise the connection will fail	Int16,UInt16,Int32,UInt32 0: Disabled non-0: Enabled
User name	The user name used to log on the server	Up to 256 characters	String
Password	The password used to log on the server	Up to 256 characters	String
Heartbeat Cycle	Use this to send a heartbeat message to the server regularly. If the server does not receive a message from the client within one and a half heartbeat cycle, it will disconnect from the client	2s-32767s	/
Auto Connect	If you check this, the client will be disconnected automatically if the reconnection time is reached, and will not be connected until the next time the data needs to be updated		/
Reconnection Time	The idle time before automatic disconnection	1s-32767s	/
Status Address	Indicates the connection state, which can be null	0: Stopped, 1: Connected, 2: Disconnected	Int16,UInt16,Int32,UInt32
Error Address	Error code output address, which can be null	0: No error, non-0: error is present (for details on the error code, see the following text)	Int16,UInt16,Int32,UInt32

2. Error messages

```
enum ClientError{
        UnknownError = 0,
        SocketConnectionRefusedError,
        SocketRemoteHostClosedError,
        SocketHostNotFoundError,
        SocketAccessError,
        SocketResourceError,
        SocketTimeoutError,
        SocketDatagramTooLargeError,
        SocketNetworkError,
        SocketAddressInUseError,
        SocketAddressNotAvailableError,
        SocketUnsupportedSocketOperationError,
        SocketUnfinishedSocketOperationError,
        SocketProxyAuthenticationRequiredError,
        SocketSslHandshakeFailedError,
        SocketProxyConnectionRefusedError,
        SocketProxyConnectionClosedError,
        SocketProxyConnectionTimeoutError,
        SocketProxyNotFoundError,
        SocketProxyProtocolError,
        SocketOperationError,
        SocketSslInternalError,
        SocketSslInvalidUserDataError,
        SocketTemporaryError
};
```

17.1.2 Publishing MQTT Topics

The publisher integrates the specified address data into json data and published to the server. As shown in the following figure, you can create up to 256 new topic publications. You can edit the properties in a table.



1. The properties are described as follows:

Property	Description	Remarks
Name	The name of this topic publication, which cannot be duplicate	Up to 128 characters
Topic	The topic received by the server (For topic configuration, see *"17.1.4 Topic Wildcards" on page 428*	Up to 512 characters
Send Mode	Sets the mode of topic publishing, which is cyclic continuous by default	
Cycle	Effective when the send mode is cyclic continuous	1s–32767s, default is s
Retain	If you select Yes, the MQTT server will retain the last message in the topic, which can be received after the subscriber client connects to the server	Default is No
Qos	Message quality	"At most once": Message delivery depends entirely on the underlying TCP/IP network. There is no definition of reply and retry in the protocol, and the message will either only arrive at the server once or not at all. "At least once": The message received by the server is confirmed by PUBACK message. If the communication link or sending device fails, or the confirmation message is not received within the specified time, the sender will resend the message with DUP bit set in the message header. "Exactly once": This is the highest level. Message loss and duplication are unacceptable, and there is additional overhead to use this level.

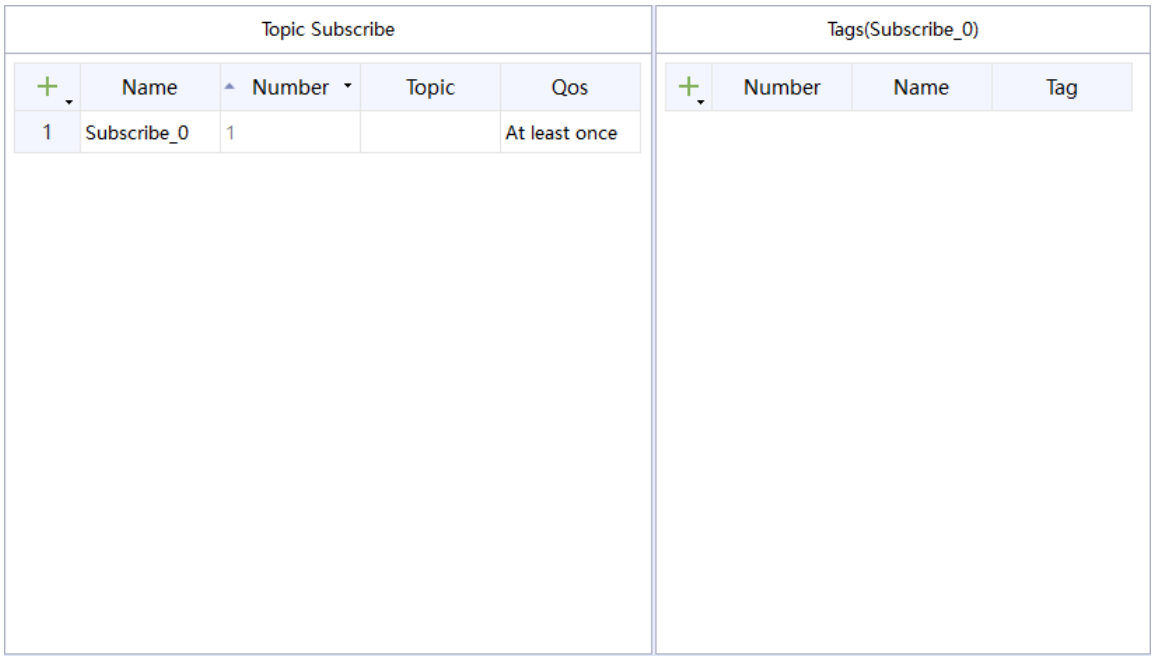
2. Add tag addresses.

   If you have configured tag addresses, during publishing, the addresses added in the topic will be integrated into json data and sent to the server. One topic allows up to 1024 rows of tags.

Property	Description	Remarks
Name	The name used in json data format during publishing. You can use a tag name. It will be automatically changed to the tag name when you select a tag.	Up to 128 characters
Tag	The tag used in json data	

17.1.3 MQTT Topic Subscription

The receiver receives the data published by the publisher and parses the data to the specified address. If the name in the data does not exist in the subscribed tag, the data is discarded. As shown in the following figure, you can create up to 256 new topic subscriptions. You can edit the properties in a table.



1. The properties are described as follows:

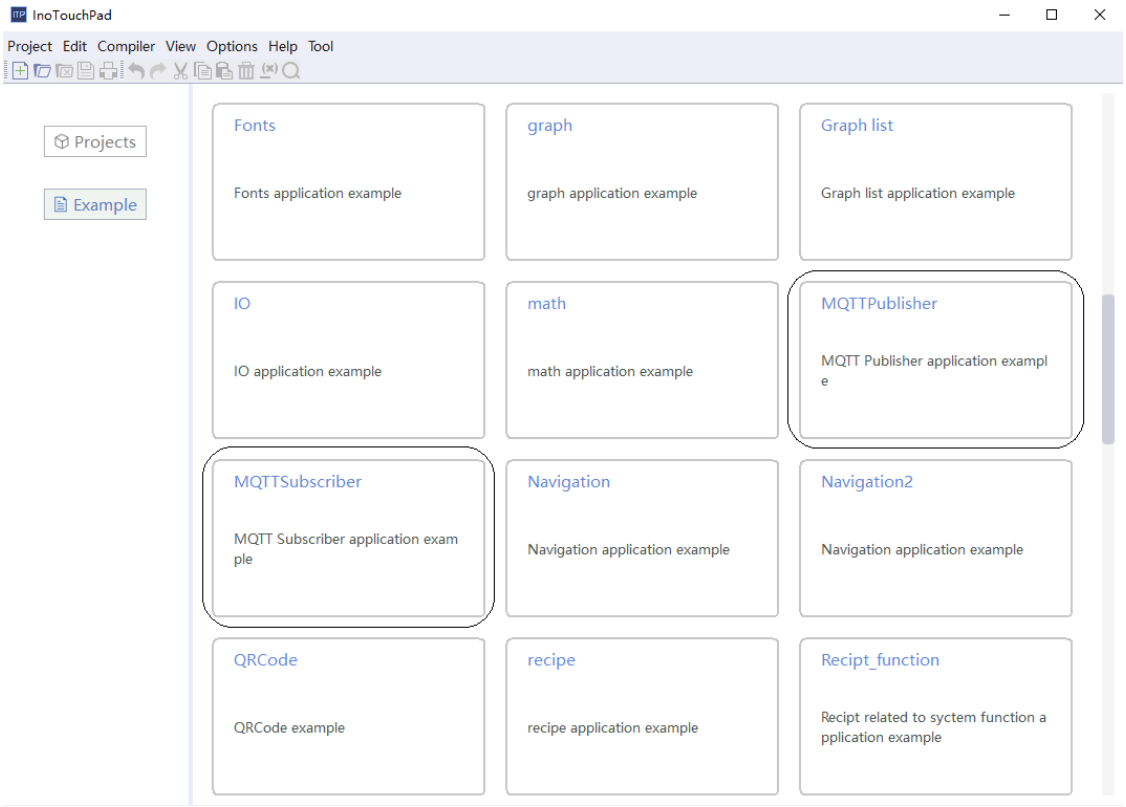
Attribute	Description	Description
Name	The name of the topic subscription, which cannot be duplicate	Up to 128 characters
Topic	The subscribed topic (For topic configuration, see *"17.1.4 Topic Wildcards" on page 428*)	Up to 512 characters
Qos	Message quality	"At most once": Message delivery depends entirely on the underlying TCP/IP network. There is no definition of reply and retry in the protocol, and the message will either only arrive at the server once or not at all. "At least once": The message received by the server is confirmed by PUBACK message. If the communication link or sending device fails, or the confirmation message is not received within the specified time, the sender will resend the message with DUP bit set in the message header. "Exactly once": This is the highest level. Message loss and duplication are unacceptable, and there is additional overhead to use this level.

2. Add tag addresses

   After you configure the tag addresses for topic subscription, the received data is parsed into each address in json data format. The value of the corresponding tag is updated when a name in the tag address table matches the name in the json data. One topic subscription allows up to 1024 lines of tags.

Attribute	Description	Description
Name	The name used in json data format during publishing. You can use a tag name. It will be automatically changed to the tag name when you select a tag.	Up to 128 characters
Tag	The tag used in json data	N/A

3. For details, see the example projects "MQTTPublisher" and "MQTTSubscriber" that come with InoTouchPad:



17.1.4 Topic Wildcards

When establishing a connection, especially in the application which involves a server and multiple clients, you need to subscribe to the topics of all devices. Then you need to know how to use the MQTT wildcards. A subscription may contain special characters and allow multiple topics to be defined at once. The topic level separator is used to introduce hierarchies into topics. Multi-level wildcard and single-level wildcars can be used, but they cannot be used for publisher messages.

1. Topic level separator "/"
   "/" is used to divide each level of the topic tree and provide a hierarchical structure for the topic space, so that you can know which levels of messages you can receive according to the separator in the wildcard characters.

2. Multi-level wildcard "#"
   "#" is used to match multiple levels in a topic. For example, if a client subscribes to "finance/stock/ibm/#", it can receive the following topic messages.

   finance/stock/ibm

finance/stock/ibm/closingprice

finance/stock/ibm/currentprice

As described above, the "#" wildcard character can represent the topic at the current or next level, so "finance/#" can also match the "finance" level. Note that the "#" must be the last character in the topic tree. For example, "#" and "finance/#" are valid, but "finance/#/closingprice" and "finance#" are invalid.

3. Single-level wildcard "+"
"#" is used to match a level in a topic. For example, "finance/stock/+" can match "finance/stock/ibm" and "finance/stock/xyz", but not "finance/stock/ibm/closingprice" and "finance/stock". A single-level wildcard can be used in combination with a multi-level wildcard, but it must be used only after the topic-level separator unless it matches itself. So "+", "finance/+" and "finance/+/ibm" are all valid, while "finance+" is invalid.

4. Notes

a. The topic must have at least one character.
b. The topic name is case sensitive. For example, ACCOUNTS and Accounts are two different topics.
c. A topic name can contain spaces. For example, "Accounts Payable" is a valid topic.
d. A topic beginning with "/" will become a new topic. For example, "/finance" is different from "finance", that is, "/finance" can match "+/+" and "/+".
e. Do not include characters such as null (Unicode\ x0000) in topic names.

Shenzhen Inovance Technology Co., Ltd.
Address: Building E, Hongwei Industrial Park, Liuxian 2nd Road, Baocheng 70th Zone,

Bao'an District, Shenzhen 518101, P.R.China
Tel: (0755) 2979 9595
Fax: (0755) 2961 9897
http://www.inovance.com

销售服务联络地址

Suzhou Inovance Technology Co., Ltd.
Address: No.16 Youxiang Road, Yuexi, Wuzhong District, Suzhou 215104

Tel: (0512) 6637 6666

Fax: (0512) 6285 6720

http://wwww.inovance.com

Copyright©Shenzhen Inovance Technology Co.,Ltd.