Welcome

HMI Software

Human Machine Interface (HMI) Programming Software, It is a HMI programming tool.

This Help document will brifly introduce the functions, programming process for the software.

Software Homepage

Start software:

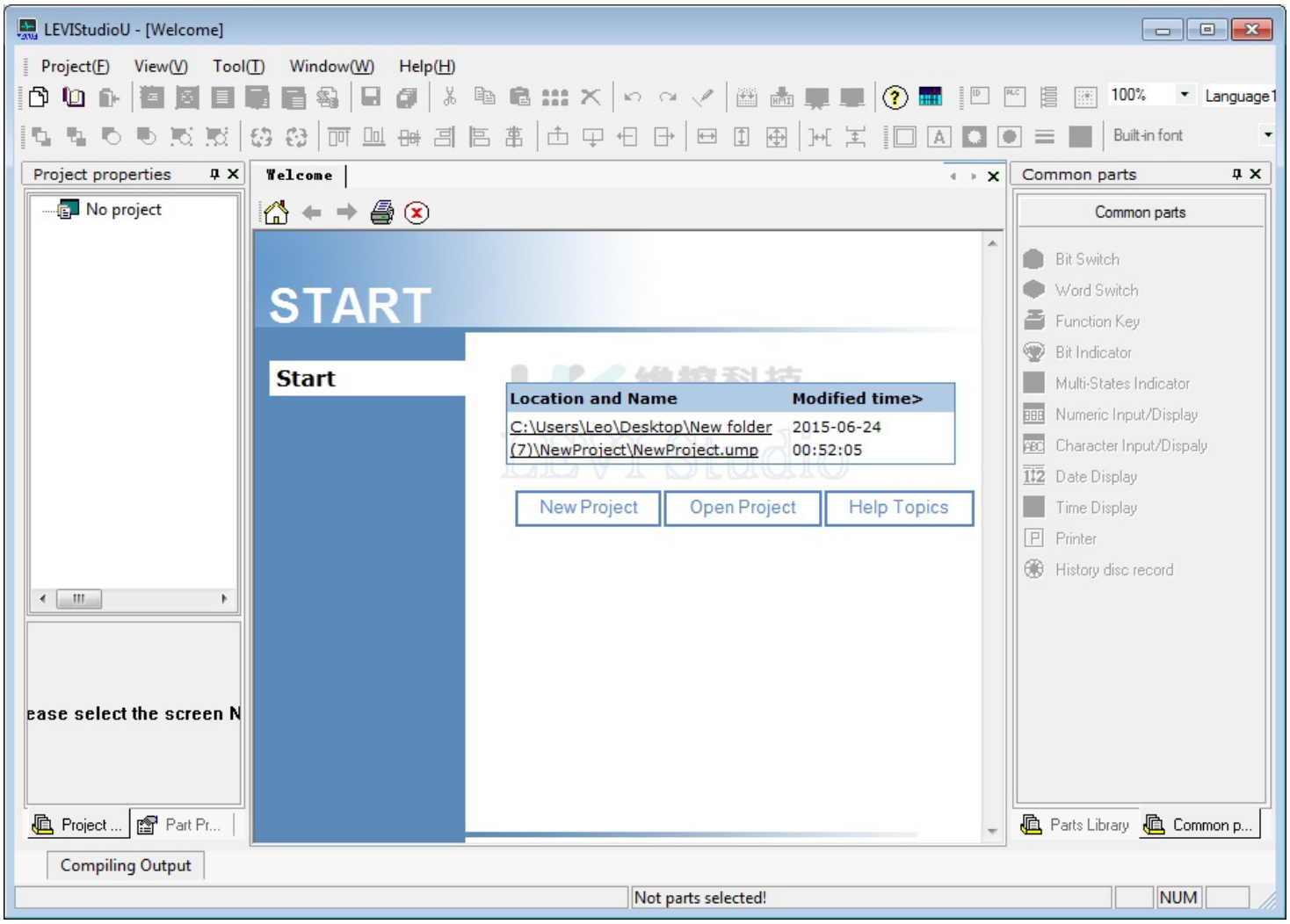


Figure 1

1. Decompile:

HMI project copied from HMI need to be de-compiled first.

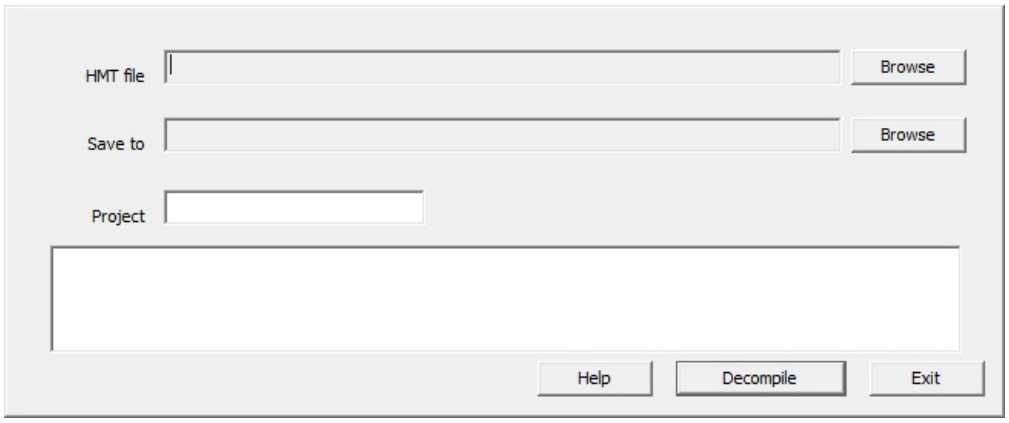


Figure 2

Property	Description
HMT File	Select existing HMT file from local folder.
Save to	Saving location of decompiled HMI project.
Project Password	Require password if the uploaded project comes with designer password.

2. Udisk Download:

Hold right top corner of HMI for 5 seconds to enter [Setup Screen].

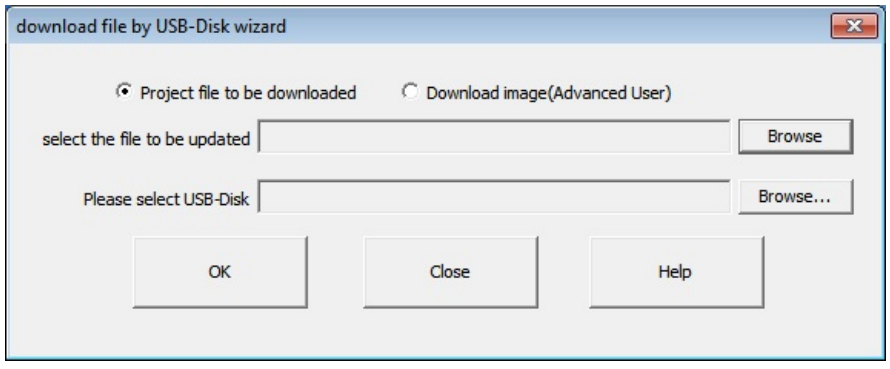


Figure 3

Property	Description
Project file	Select HMI project file which end with "HMI", this project file will be downloaded to HMI via USB disk (Insert USB Disk to HMI, then go to [Setup Screen] of HMI, then click [Copy HMT]).
Firmware file	Select Firmware file which end with "OSF",this firmware file will be downloaded to HMI via USB Disk(Insert USB Disk to HMI, then go to [Setup Screen] of HMI, then click [Update OSF]).

Demo Projects

Various small demo projects are provided to illustrate several aspects of using the HMI programming.
All of the demo projects are existing in [Demo Projects] which under software installation folder.

Functions:

1. Animation
2. Change Screen by Address
3. Data Record to UDisk
4. Discrete Recipe
5. File List Recipe
6. Simple Recipe

Script:

7. Four Arithmetic Operations
8. Timer
9. Determine

Extensions:

10. User Restriction
11. Short Message

Communication:

12. Open CAN
13. Siemens S7-1200 Ethernet
14. User-Defined Protocol
15. Modbus RTU-Client
16. Modbus RTU-Host
17. Modbus TCP_Host
18. Modbus TCP_Client
19. Multi Link_Client
20. Multi Link_Host

Start a new project

The new project can be created by selecting [Project]-[New Project] form the menu bar, or clicking the [New Project] on tool bar.

Location and Name:

Project name :“Newproject” by default, it can only contain a~z, 0~9.
Select a location to save the project files.
Choose “HMI” or “PLC” model from the list.

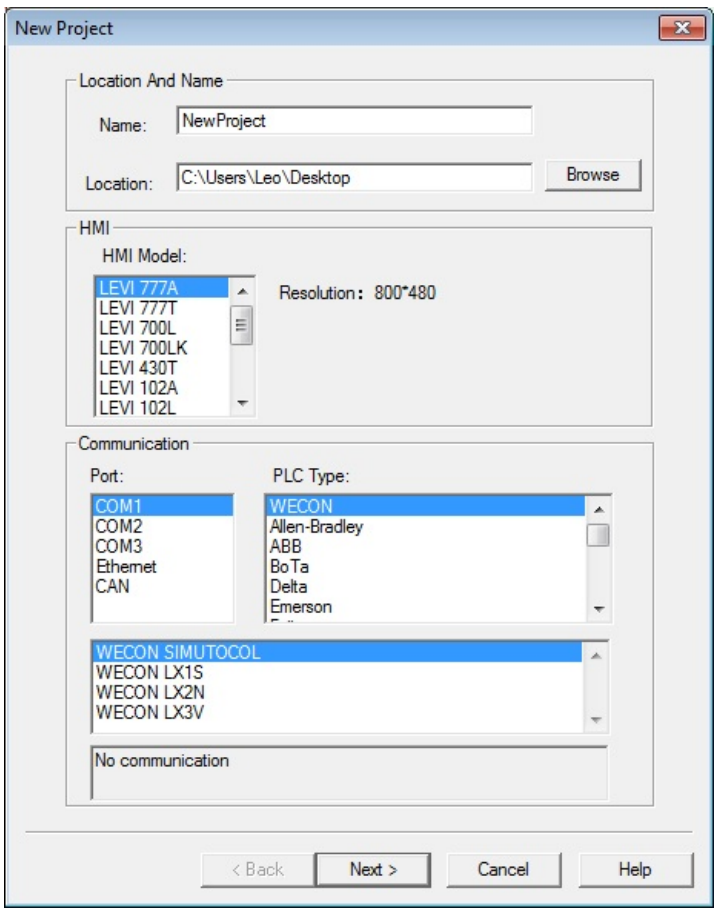


Figure 1

Project Properties

1. Select template, the project will be created as the template setting.
2. Customize the initial screen and project style.

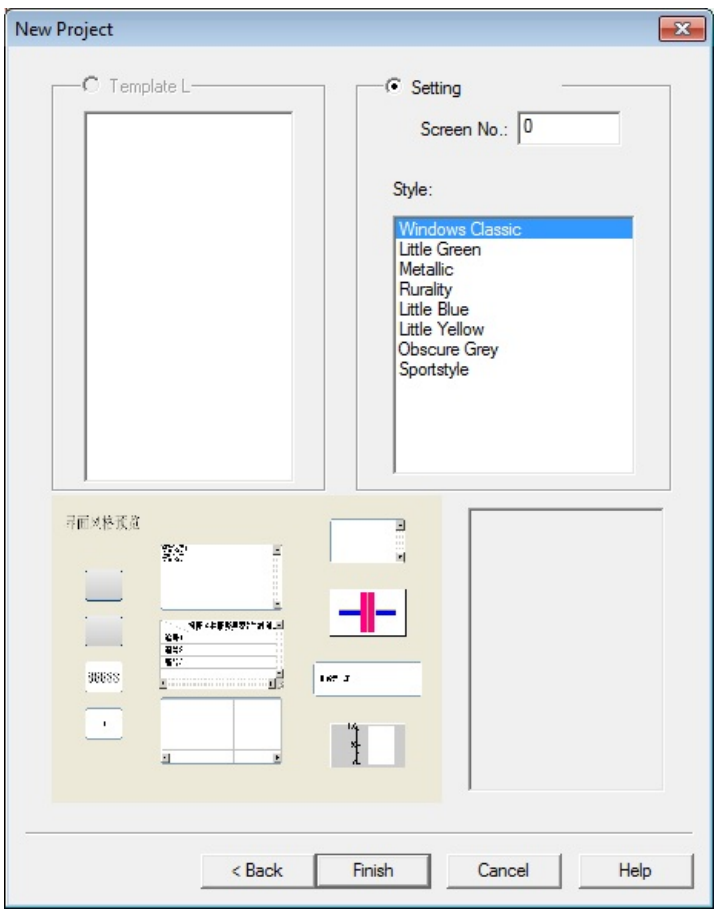


Figure 2

There are four parts in Project Manager, including Project Screen, Built-in screen, Project Setting, Library and Script.

1. Project Screen: All screens in this project.
2. Built-in scree: All built-in screens in this project, including Common Screen, Keypad Screen etc.
3. Project Properties:The project configuration information, for example, screen style, recipe, bit alarm, data record, ect.
4. Library: library information includes address, text, gallery and font.
5. Script: global script, global function and background script can be configured to achieve advanced function.

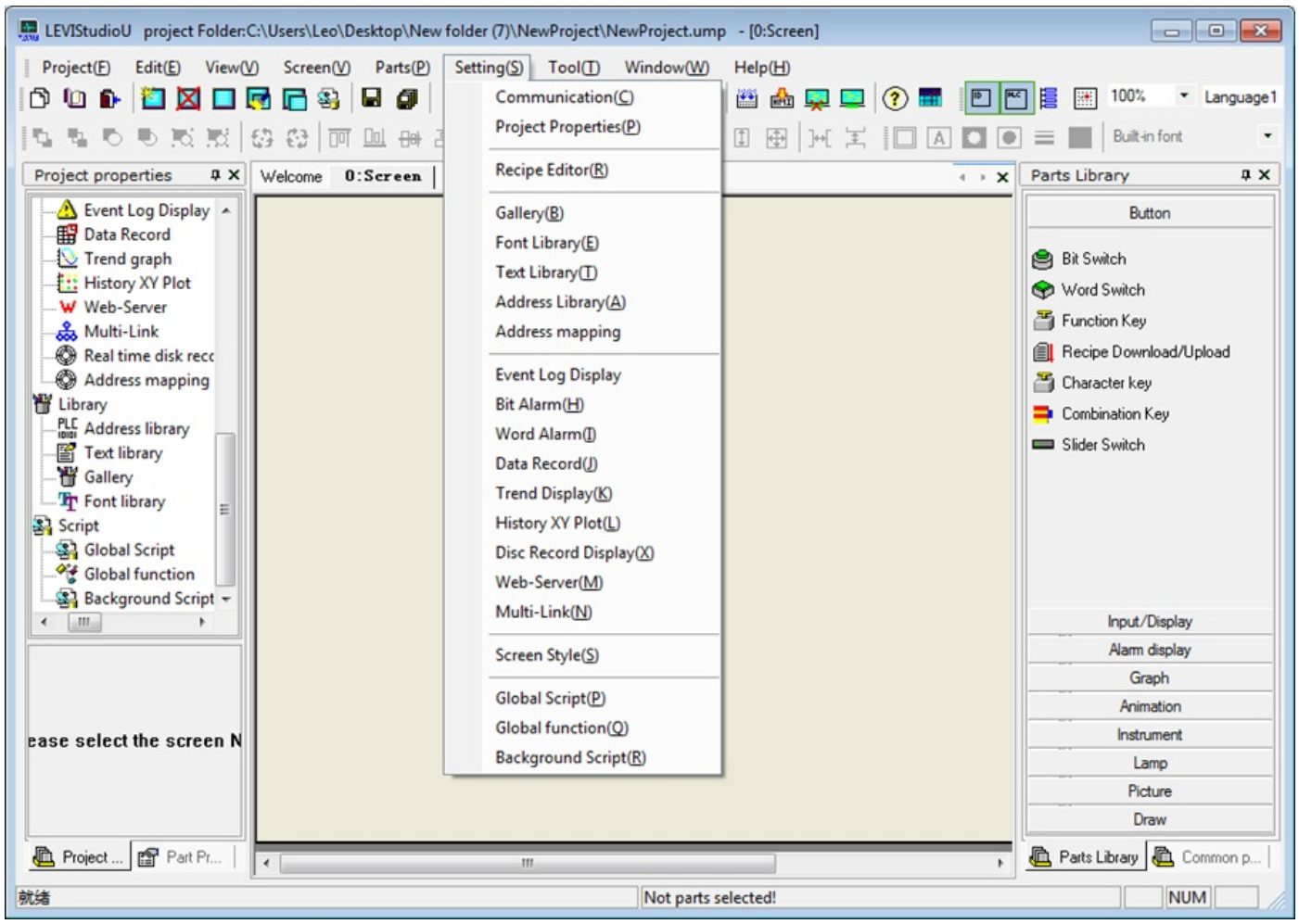


Figure 3

The HMI model can be changed by selecting [Setting]- [project properties].

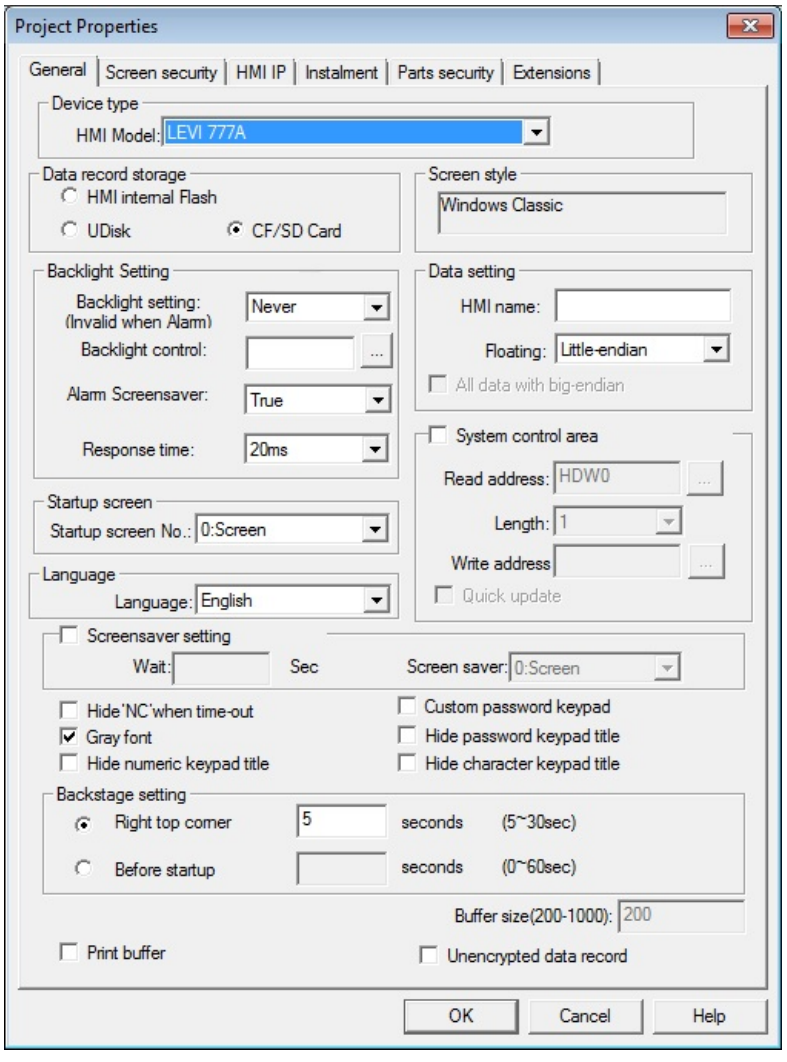

Project Properties
General
Screen security
HMI IP
Instalment
Parts security
Extensions
Device type
HMI Model:
LEVI 777A
Data record storage
HMI internal Flash
UDisk
CF/SD Card
Screen style
Windows Classic
Backlight Setting
Backlight setting:
(Invalid when Alarm)
Never
Backlight control:
Alarm Screensaver:
True
Response time:
20ms
Data setting
HMI name:
Floating:
Little-endian
All data with big-endian
System control area
Read address:
HDW0
Length:
1
Write address
Quick update
Startup screen
Startup screen No.:
0:Screen
Language
Language:
English
Screensaver setting
Wait:
Sec
Screen saver:
0:Screen
Hide 'NC' when time-out
Gray font
Hide numeric keypad title
Custom password keypad
Hide password keypad title
Hide character keypad title
Backstage setting
Right top corner
5
seconds
(5~30sec)
Before startup
seconds
(0~60sec)
Buffer size(200-1000):
200
Print buffer
Unencrypted data record
OK
Cancel
Help

Figure 4

Screen Desing

Project editor is a integrated development environment that combine the project management and screen design, user can design a new screen after the project created.

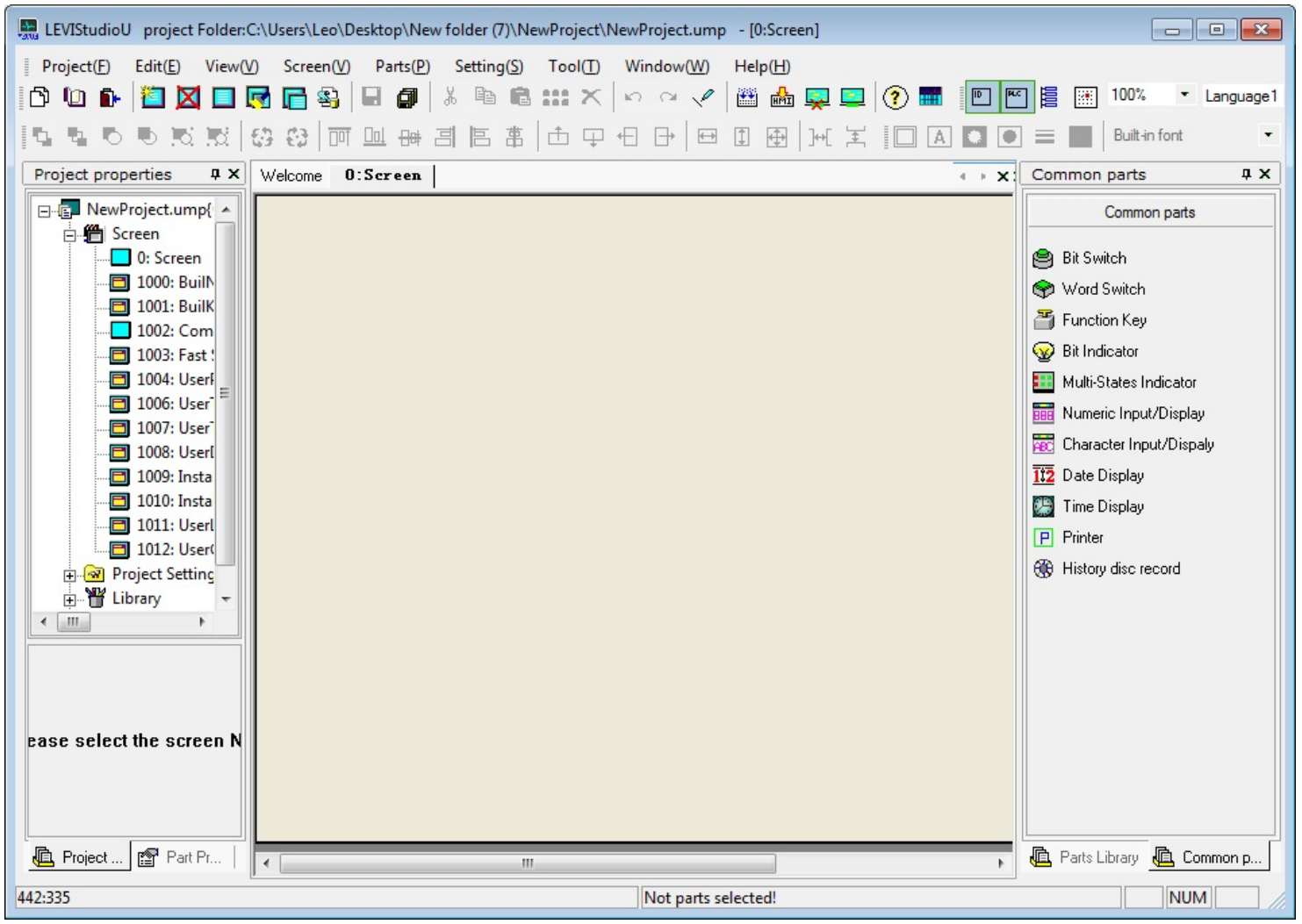


Figure 1

Add Screen

The new screen can be created by clicking “new screen” on the tool bar.

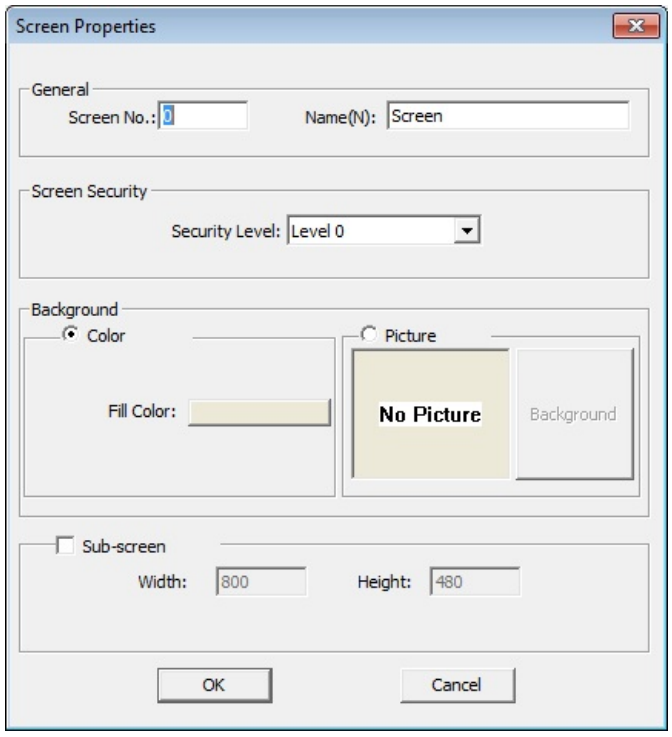


Figure 1

General:screen number and screen name (Screen No. should be unique).
Screen security: set the security level for different authorization level.
Background: screen background can be filled with single color or picture, the resolution of background picture and screen should be same .
Sub-screen: create the new screen as sub-screen.
Click [OK] to complete creating new screen.

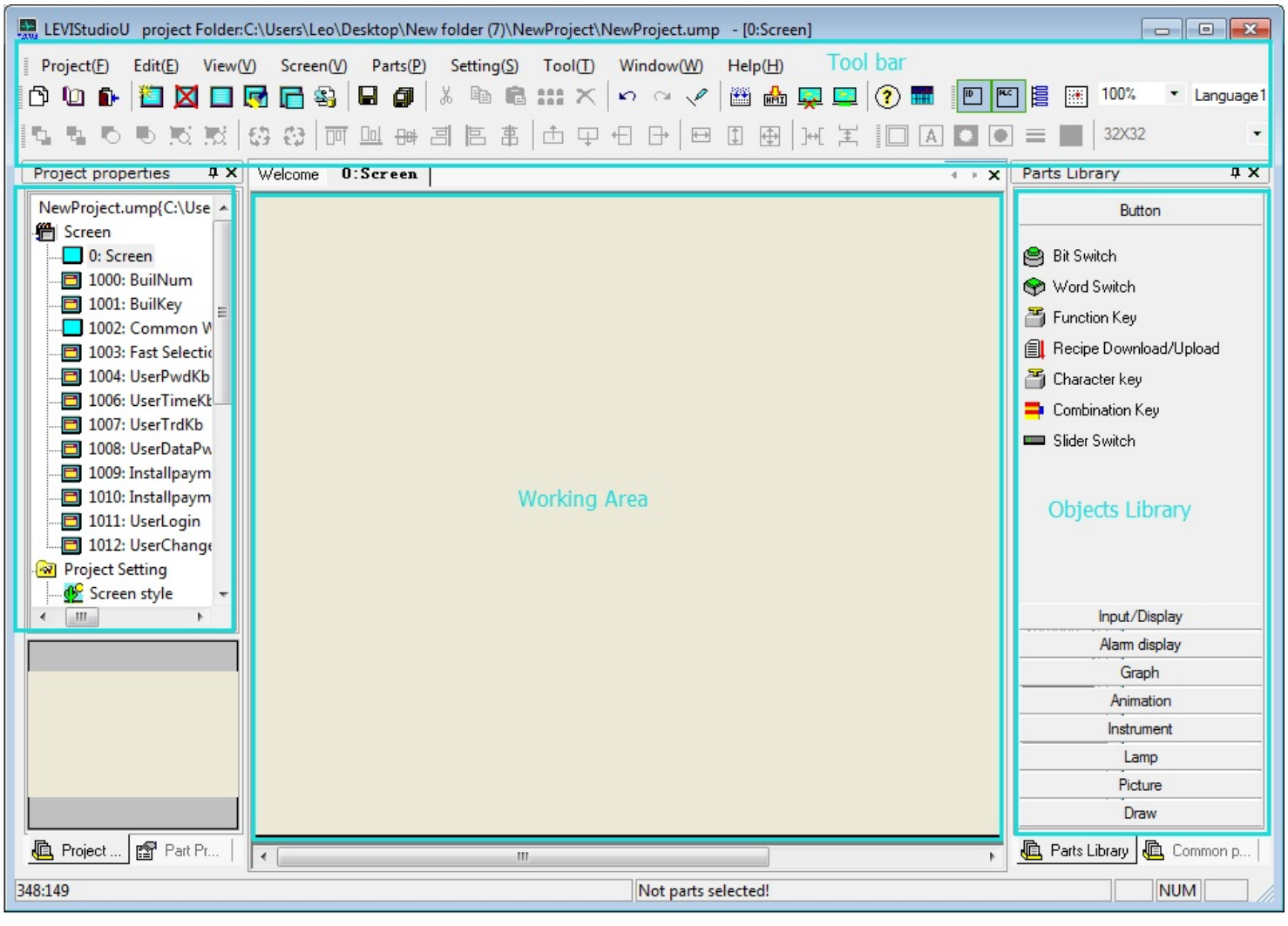


Figure 2

Add Bit Switch

Add objects to the screen, to meet the different requirement of automation control.
Select [Bit switch] from [objects Library].
Left click on screen to locate the position of "Objects" , then drag mouse to define a area for the object, left click again to finish adding the bit switch.
Double click the "Objects" to edit its properties.

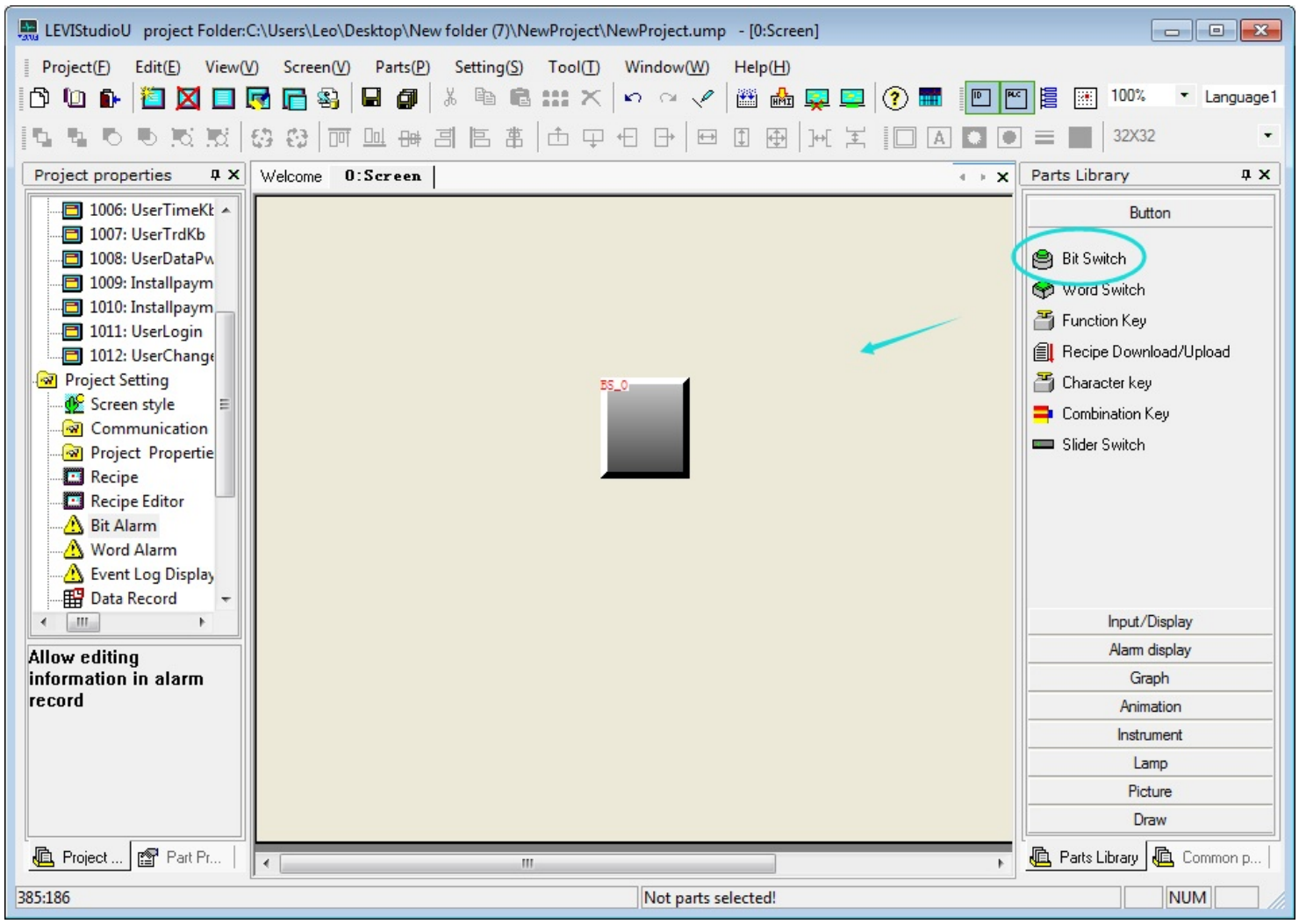


Figure 1

Address Setting

Edit the properties by double click the “Object” to open the properties window.
The address can be entered manually or edited by address editor.

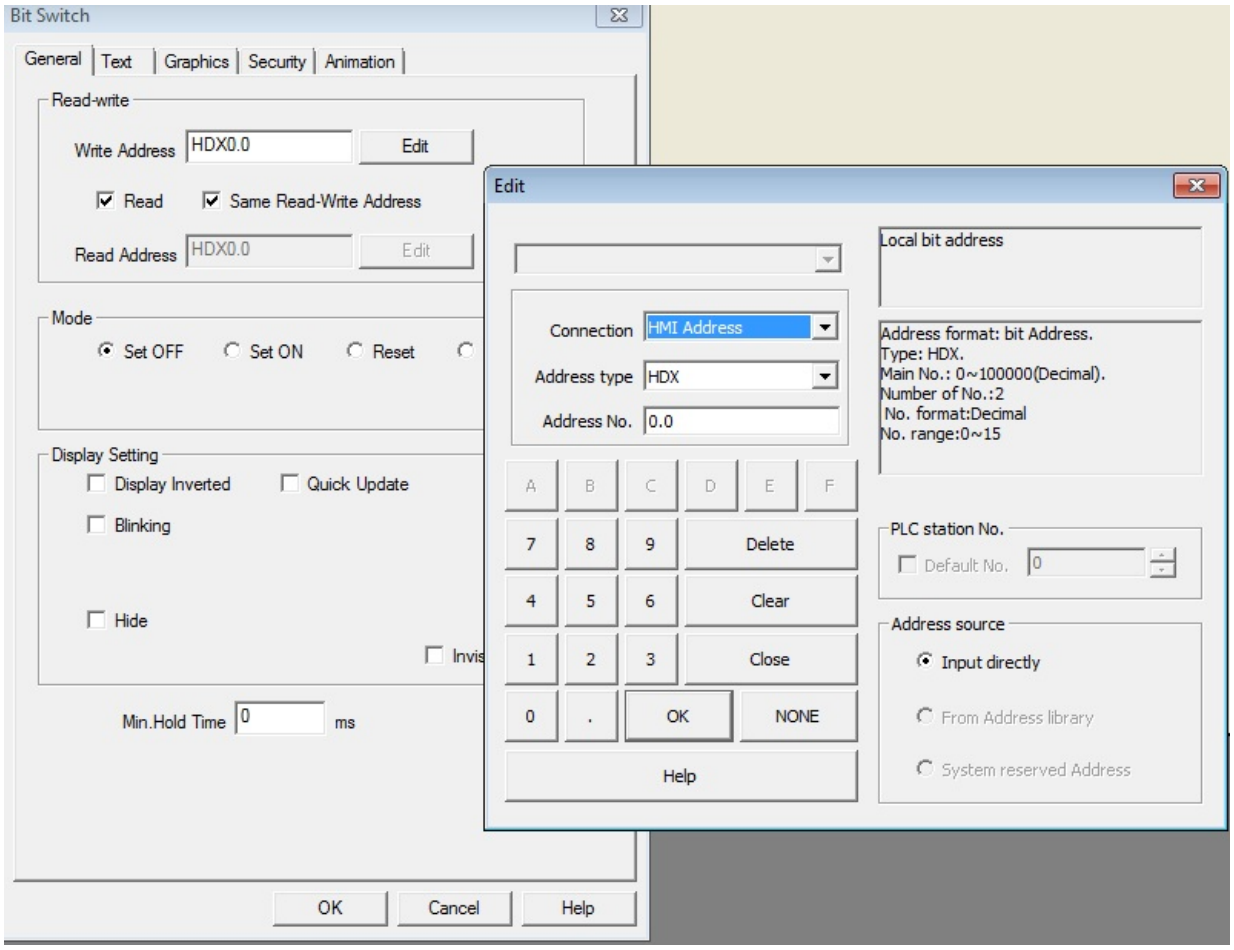


Figure 1

Appearance

The appearance setting can be found in [Object Properties] - [Graphic].
Some picture can set different colors for different states.

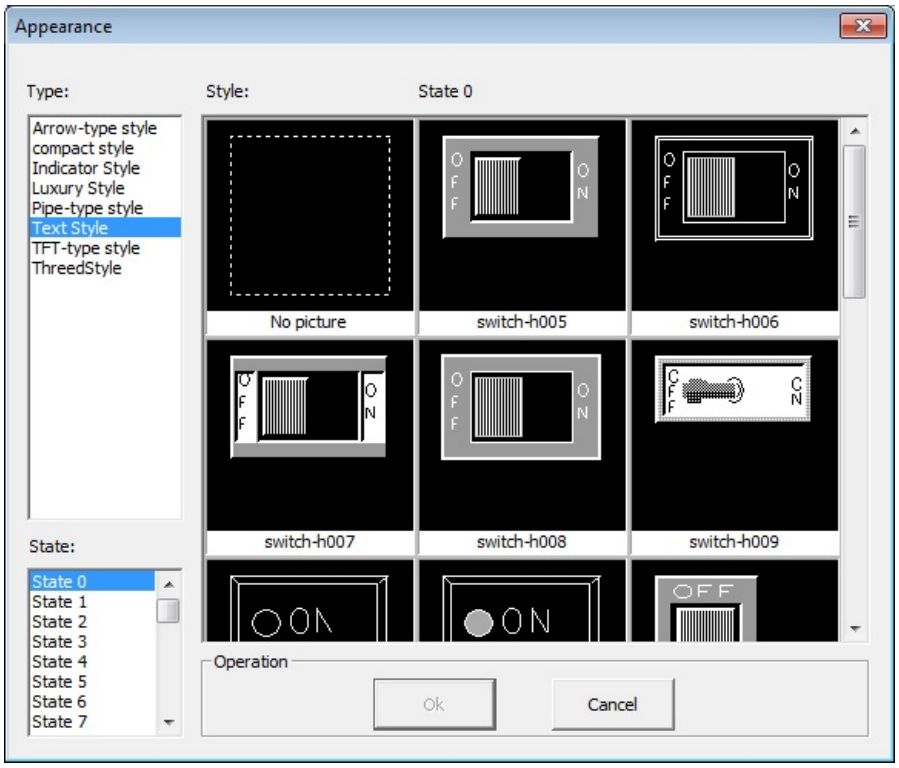


Figure 1

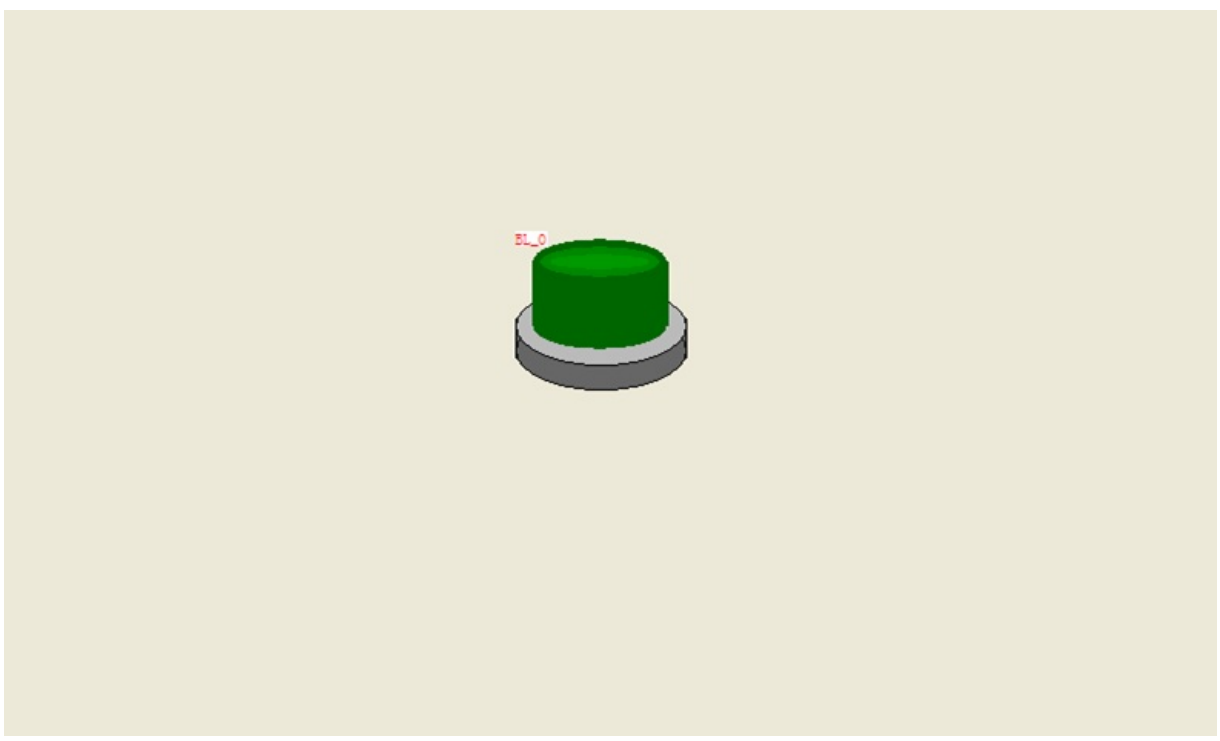


Figure 2

Gallery

Select [Use picture]- [Object Properties] - [Graphics] to customize the switch appearance.

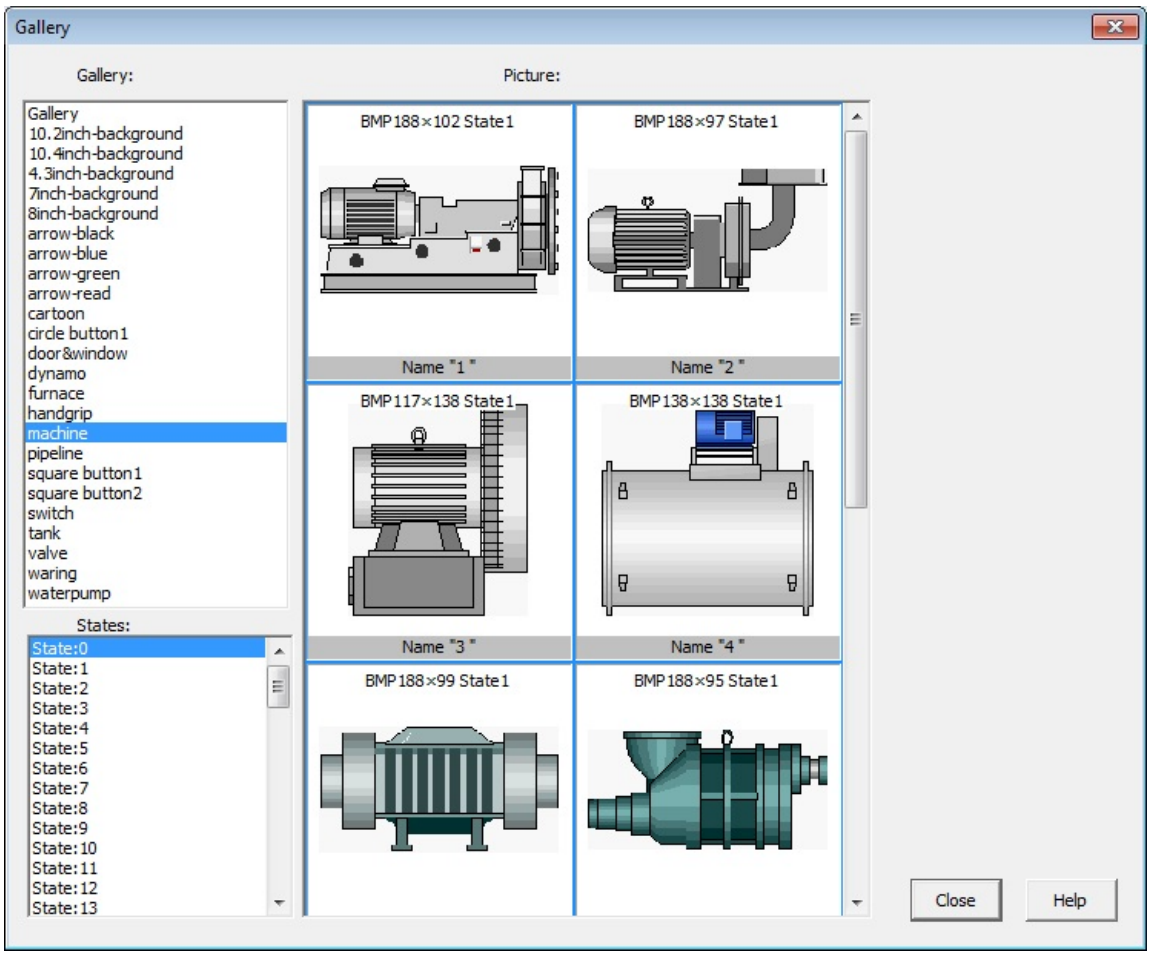


Figure 1

Double click to select the picture from the gallery.

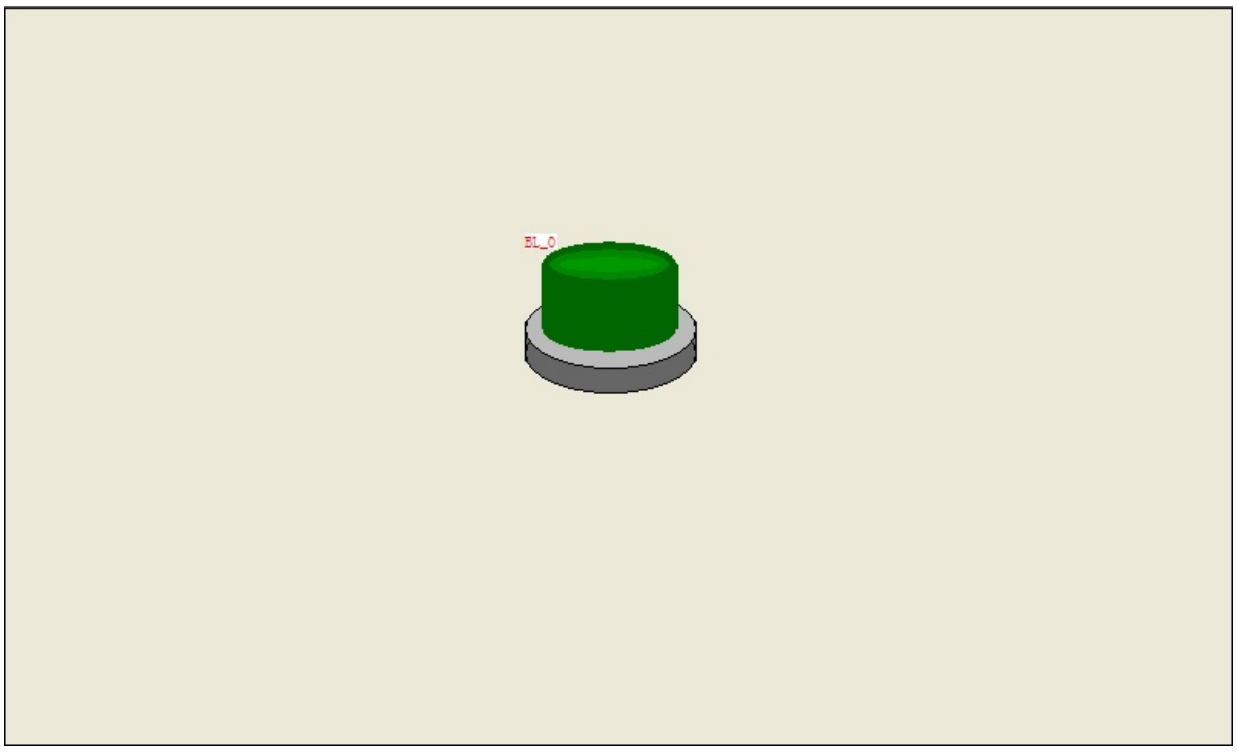


Figure 2

The pictures used in the project are all shown in the gallery.

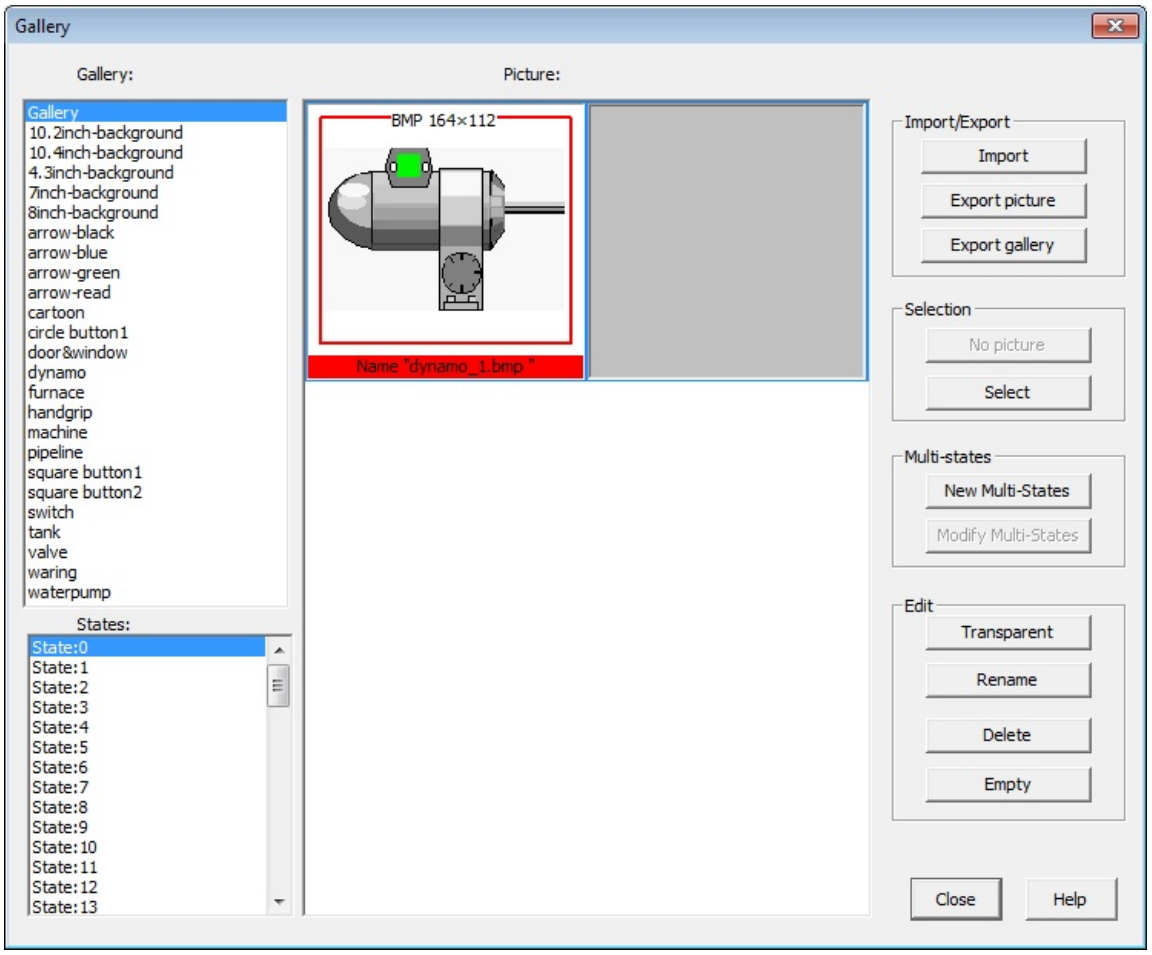


Figure 3

	Property	Description
Import/Export	Import Picture	Import the picture to the project.
	Export Picture	Export the picture from the project to local folder.
	Export Gallery	Export the entire gallery from the project.
Multi-States	New	Create a multiple state picture.
	Edit	Edit a multiple state picture.
Edit	Transparency	To make designated color transparent.

Add Bit Lamp

1. Add a bit indicator to the screen.
2. Set the read address corresponding to the device.
3. Set the suitable appearance for bit lamp.

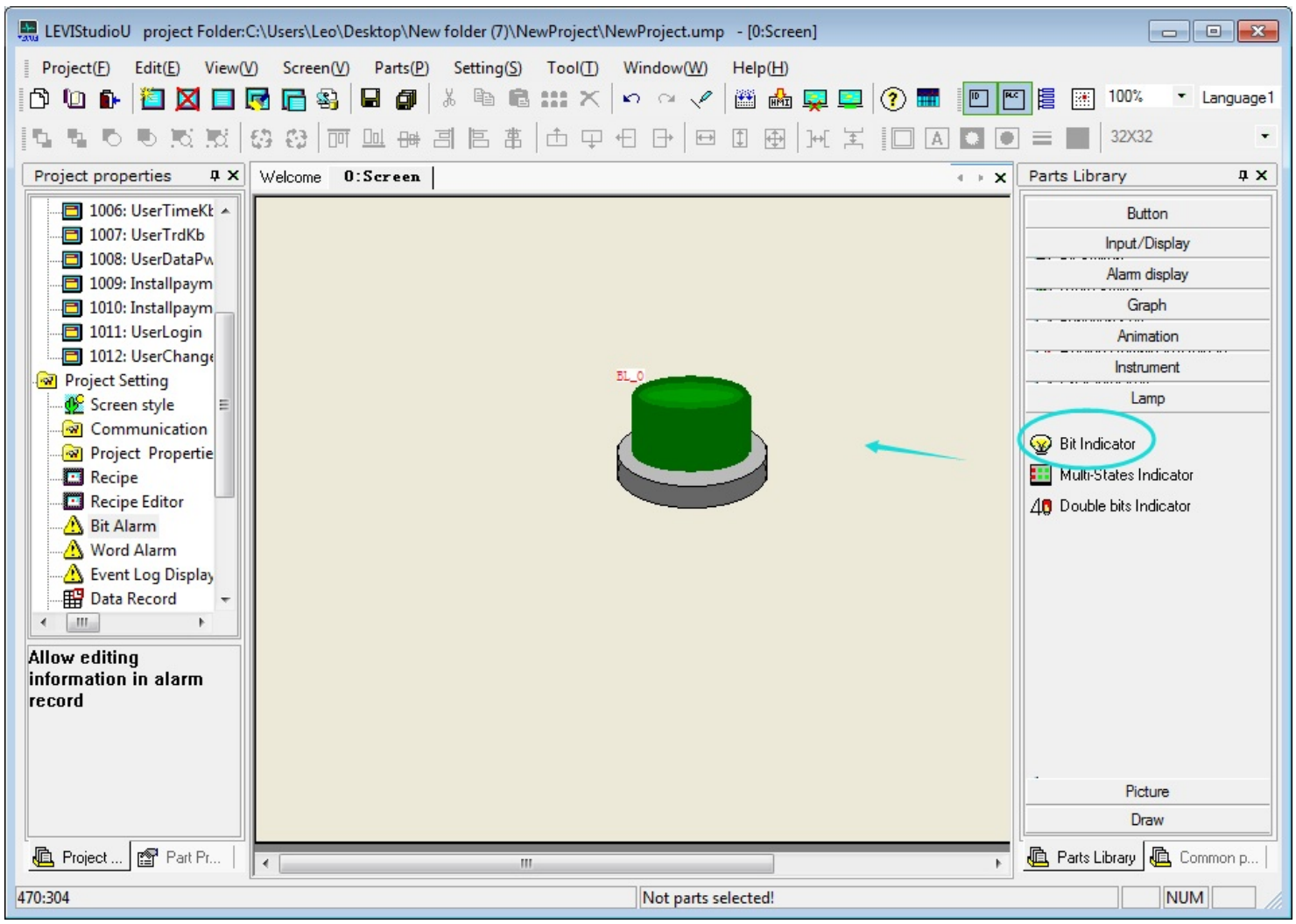


Figure 1

Add Text

1. Add text for the bit switch to distinguish from others.
2: Double click the object, enter “Hello World” in the text box.

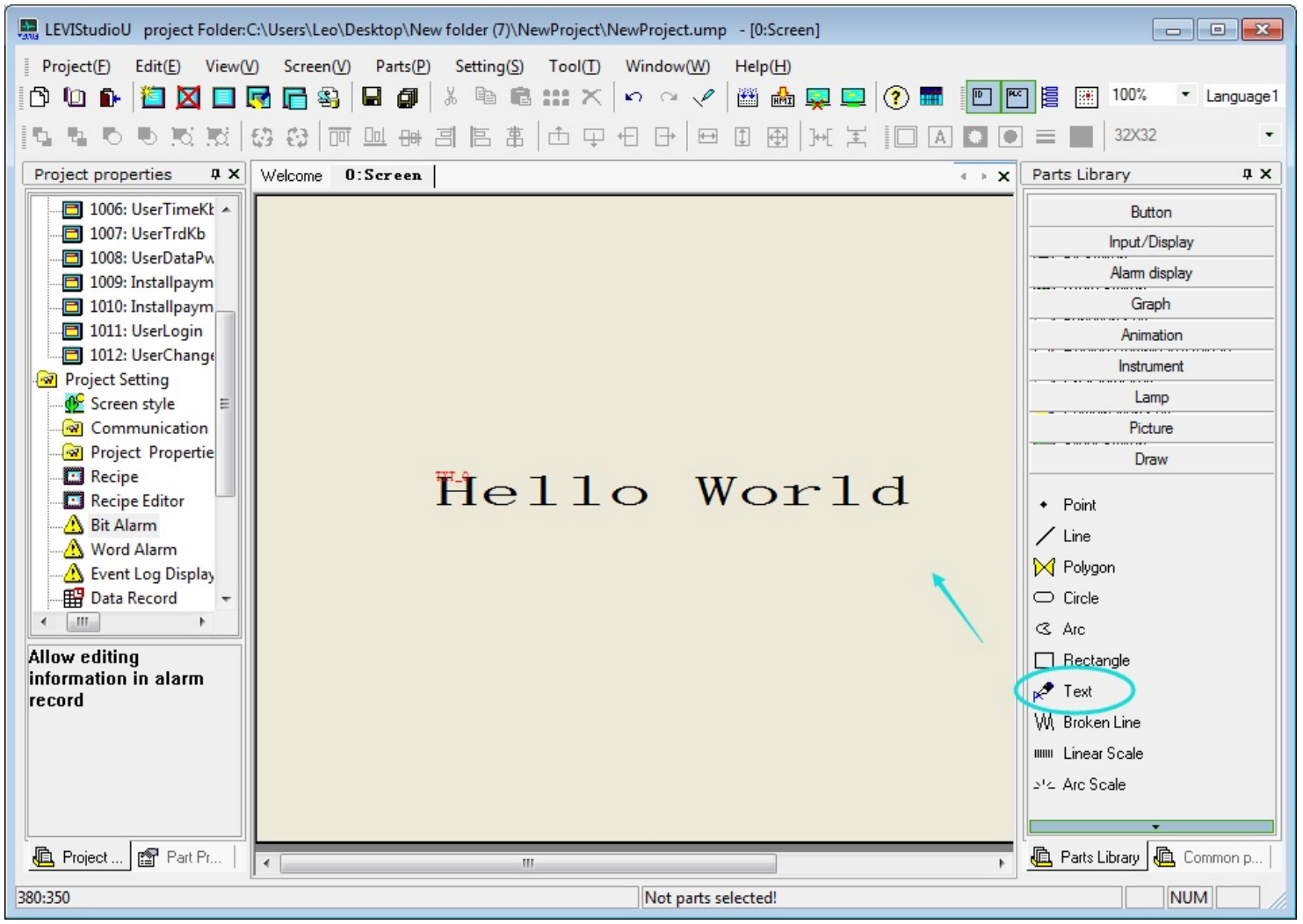


Figure 1

Add Bit Alarm

Add the Bit Alarm from [Property Setting]-[Project Setting]-[Bit Alarm] o r click [setting]-[Bit Alarm] on menu bar to add bit alarm.
Set the alarm bit address, trigger mode and other alarm information.

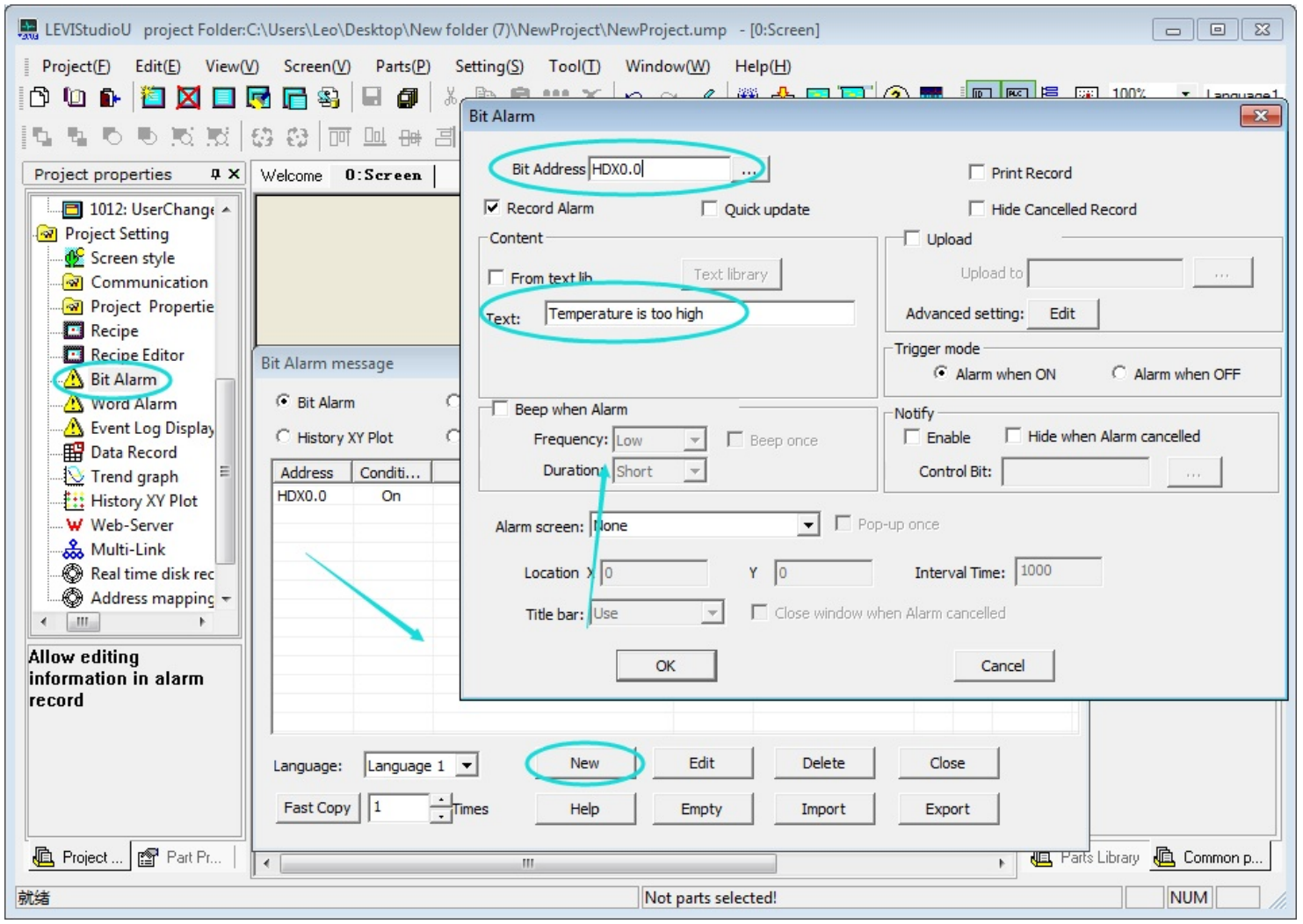


Figure 1

Place the alarm bar on the screen, the alarm text will display when the state of the bit address meet the demand of the trigger mode.

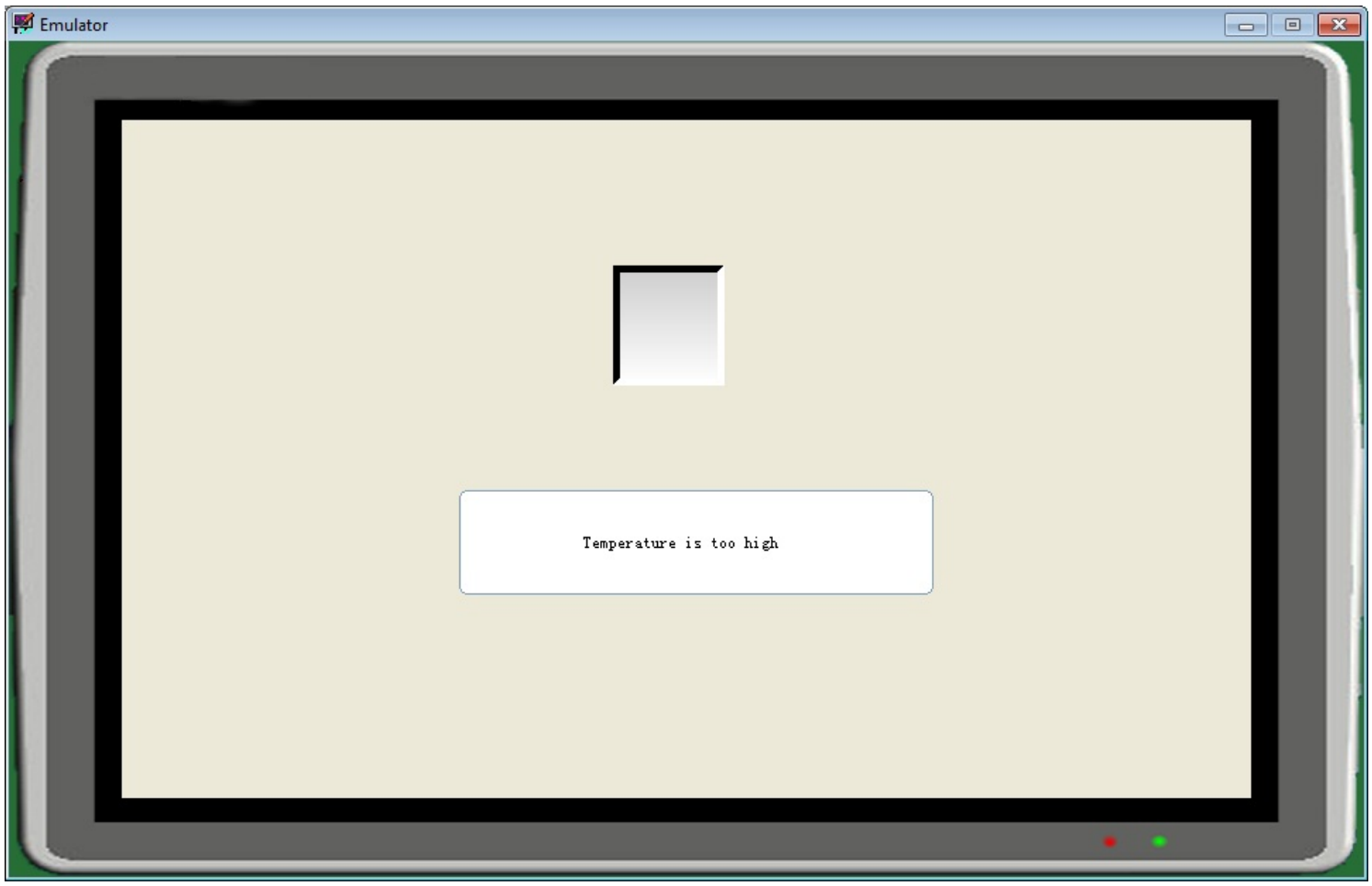


Figure 2

Compile

Click [Tool]-[Compile] or select the  icon on the toolbar to start compiling.
The project needs to be compiled before downloading or running simulator:
Checking the error existing in project.
Optimizing project to make the screen running faster on the HMI.
Processing the project fronts.
Compiling the HMI project to create project file with extension "HMT", which can be recognized by HMI.

Project Download/Upload

project download/upload is a process to transfer compiled project file to the HMI, via serial cable,USB cable or U disk.

1. Connect HMI with PC via proper download cable,before downloading.
2. HMI will enter download mode automatically,after HMI connected.
3. C lick [Tool]-[Download] on menu bar to open "Download Tool".

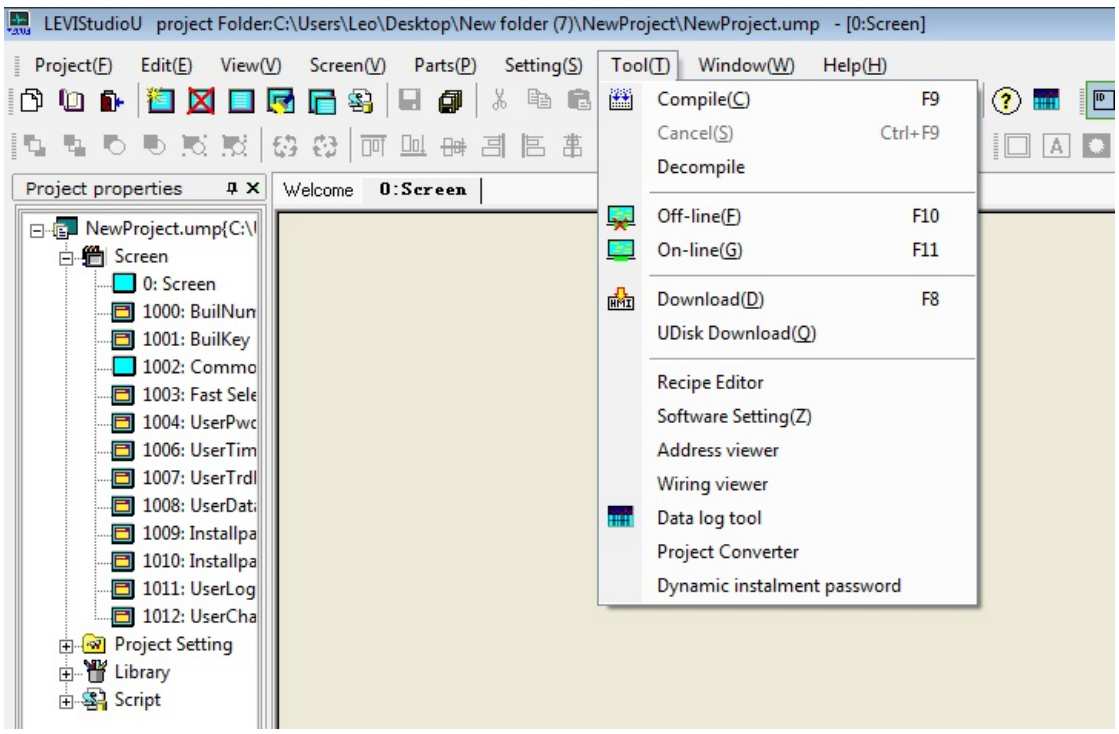


Figure 1

4. The "Download Tool" can be found on software installation folder.

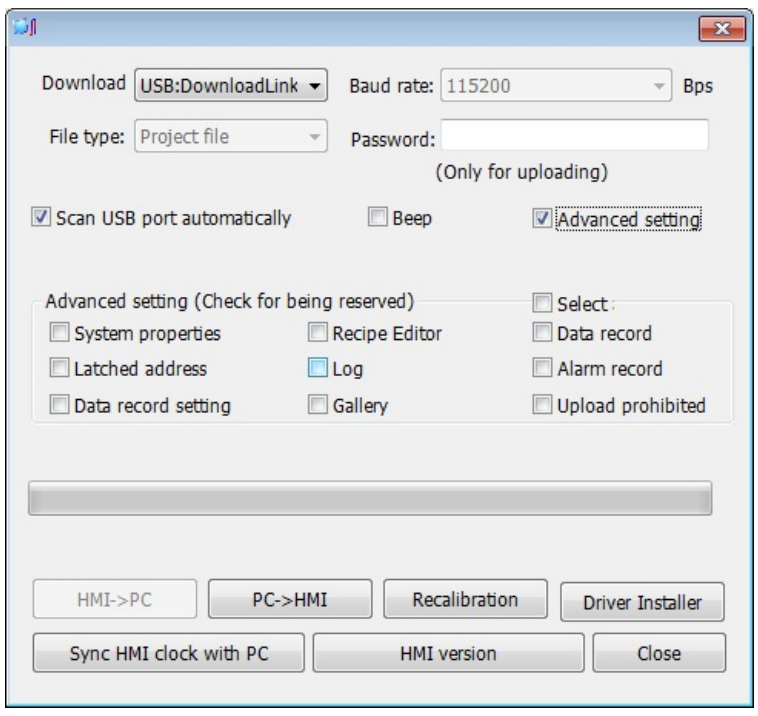


Figure 2

property	Description
Download	The port on PC used to connect with HMI (serial cable or USB cable).
Baud rate	115200 bps by default.
File type	Select a file type to download to HMI, including project file, recipe,Firmware.
Password	Uploading password. go to {Project Properties]-[Screen Security]-[Designer password]to set the password.
Advanced Setting	The selected items will not be deleted during downloading.
HMI->PC	Upload project file from HMI to PC.
PC->HMI	Download file from PC to HMI.
Recalibration	Recalibrate HMI touch screen.
Sync HMI time	Update the HMI system time with PC time.
HMI version	Check the HMI firmware version.

5. Restart HMI to take effect by clicking after HMI project downloaded.

Recipe Download/Upload

Recipe upload/download is the process to download recipe file , Select [File Type] as "Recipe File" then click [HMI->PC] or [PC->HMI].

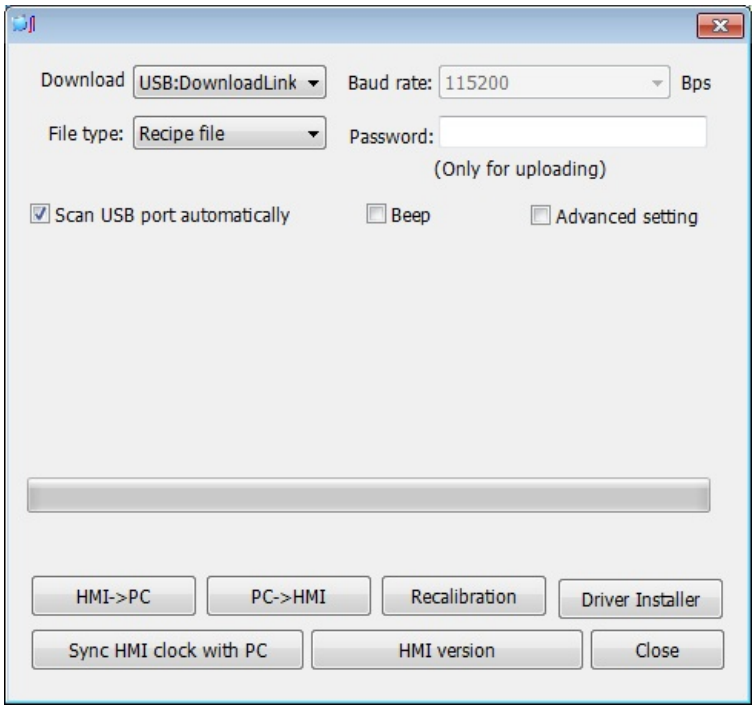


Figure 1

Simulation

HMI programming software provide two types simulation to user : Offline simulator and online simulator.

Offline simulation:[tool]-[offline simulation] 

Online simulation:[tool]-[online simulation] 

Offline Simulation

Offline simulation is the process to simulate the HMI project without any connection, the project operation can be simulated on PC before downloading to HMI.

Online Simulation

Online simulation is a process to simulate HMI project with external device connection, it enables the connection between Device and PC without connecting HMI. The project file can be edited before downloading and greatly saves time

there is a 30-minute online simulation limit.

New Screen

Click [screen]-[new screen] to add a new screen, or choose new screen from the toolbar.

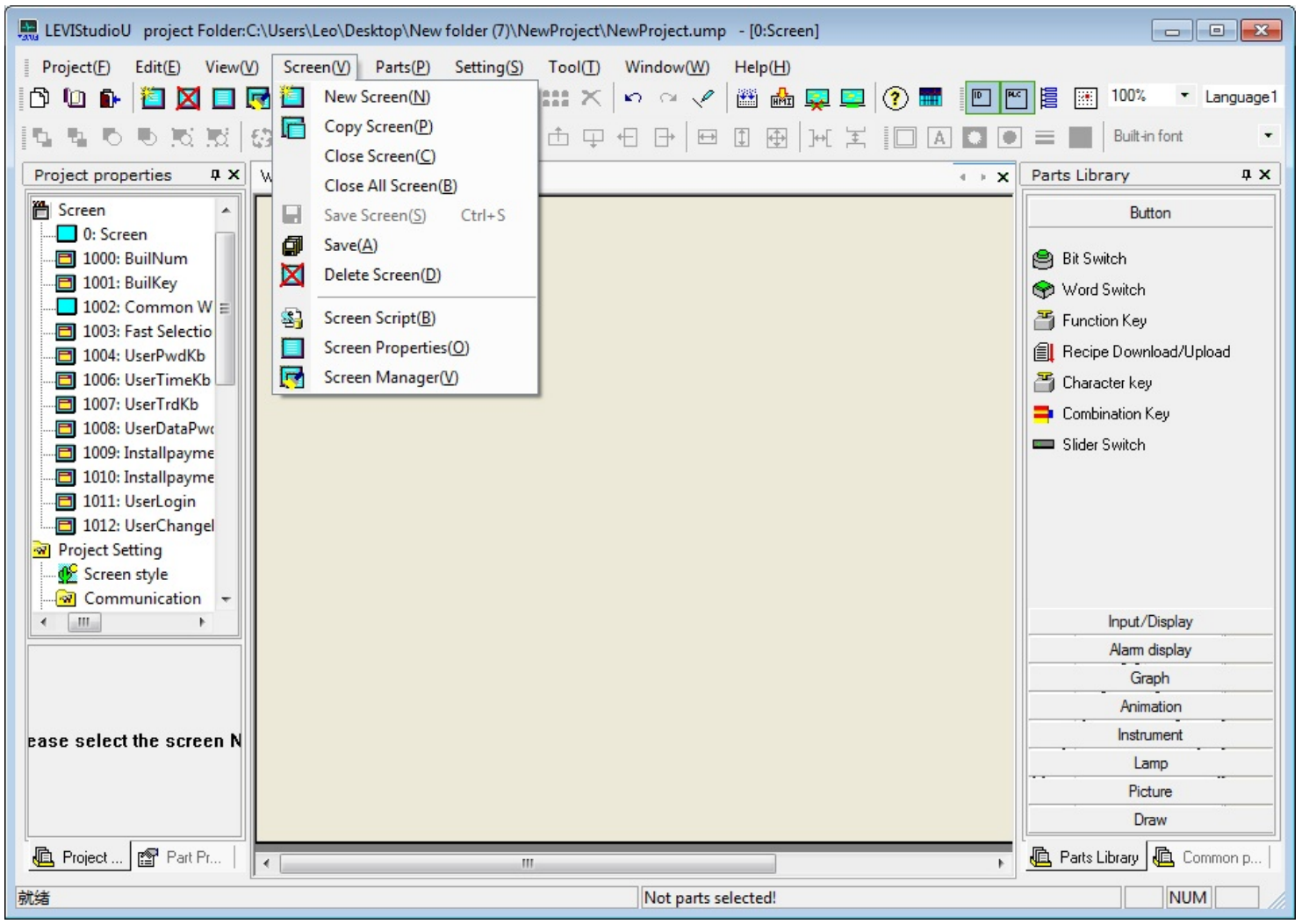


Figure 1

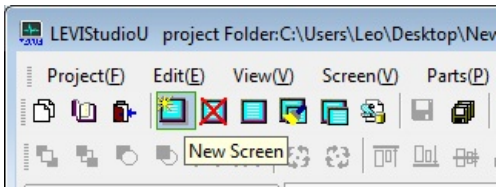


Figure 2

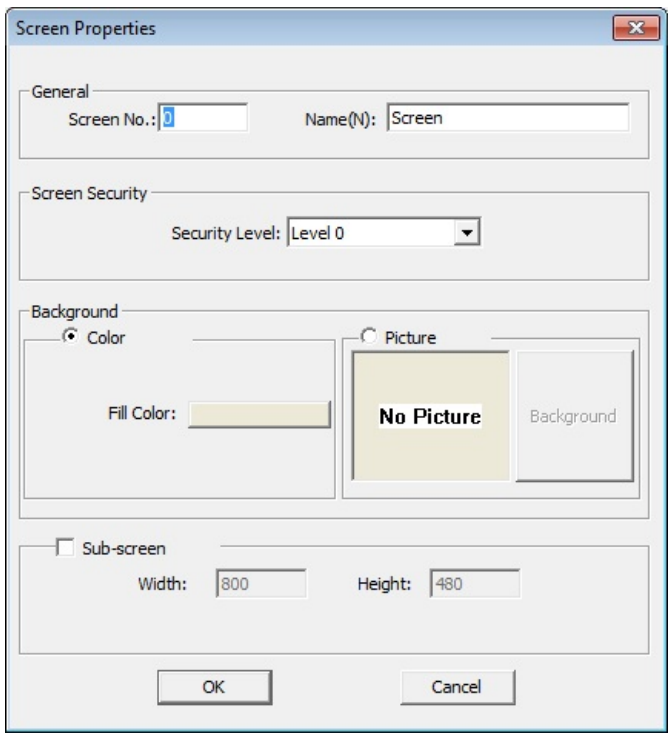


Figure 3

Property	Description
Screen No.	Screen number can only contain 0~9.
Name	Screen name can only contain 0~9, a~z.
Screen security	User can set the security level of screen.

Background	Screen background setting.
Sub-screen	Set the screen as sub-screen.

Screen and Sub-screen

Basic screen can be displayed on HMI, sub-screen can only be displayed based on basic screen.
The difference between basic screen and sub-screen:

Item	Screen	Sub-screen
Display	The screen can be changed by function Switch directly, only this kind of screen can be startup screen.	sub-screen can only be displayed based on basic screen.
Size	The full size of the touch screen.	Set the screen size according to the demand.
Display Layout	Basic screen can display on software directly.	The order of sub-screen display depends on screen objects layout.

Delete Screen

The operation will permanently deleted screen.

Figure 1

Screen Properties

Screen number should be unique.

Open screen properties dialog to edit its properties.

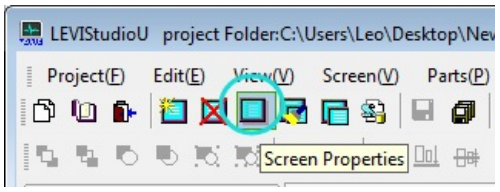


Figure 1

Right click on the working area, select [screen properties] to edit screen properties.

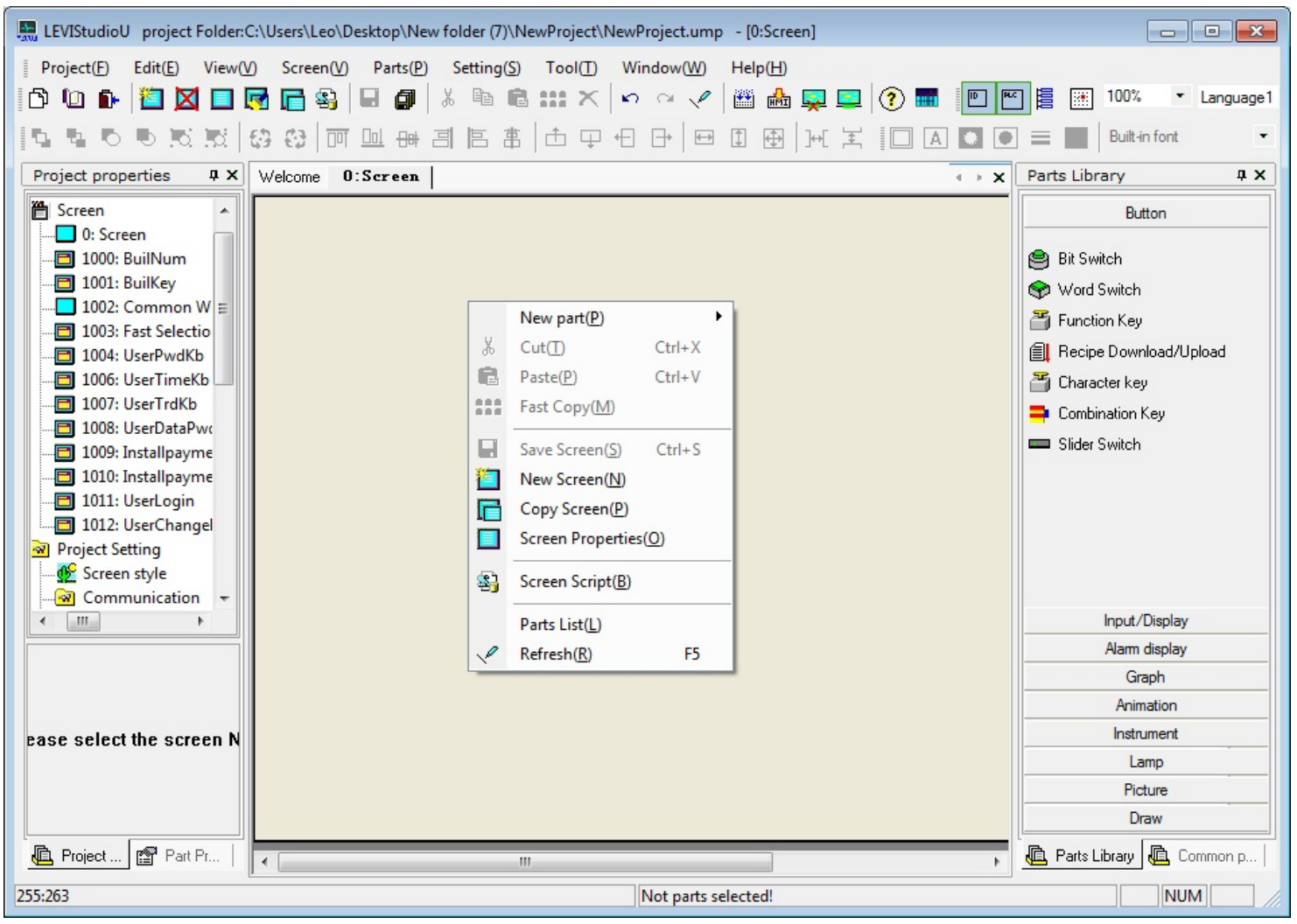


Figure 2

Copy Screen

Copy a new screen from local project or other existing project.

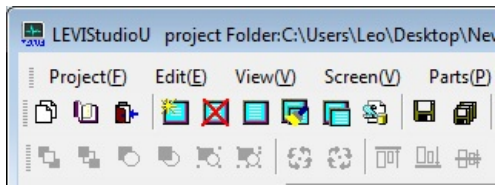


Figure 1

Notice: the picture cannot be copied automatically when copy the screen.

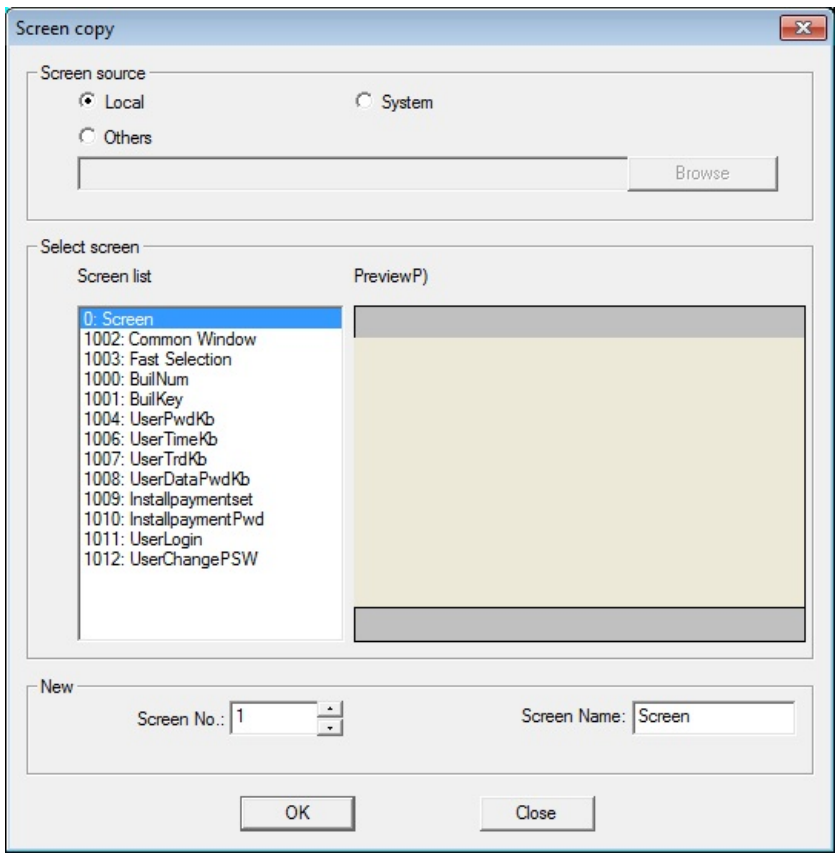


Figure 2

Property	Description
Local	Copy screen from current project.
System	Copy screen from current built-in project, like keypad screen etc.
Others	Copy the screen from other existing project.
Screen List	All of the screens in this project.
Preview	Preview of selected screen.
Screen No.	The number for copied screen, it should be unique.
Screen name	The name for copied screen.

General

Click [Setting]-[Project Properties]-[General] to edit HMI general properties, including HMI model, Startup screen, Screensaver, Screen control etc.

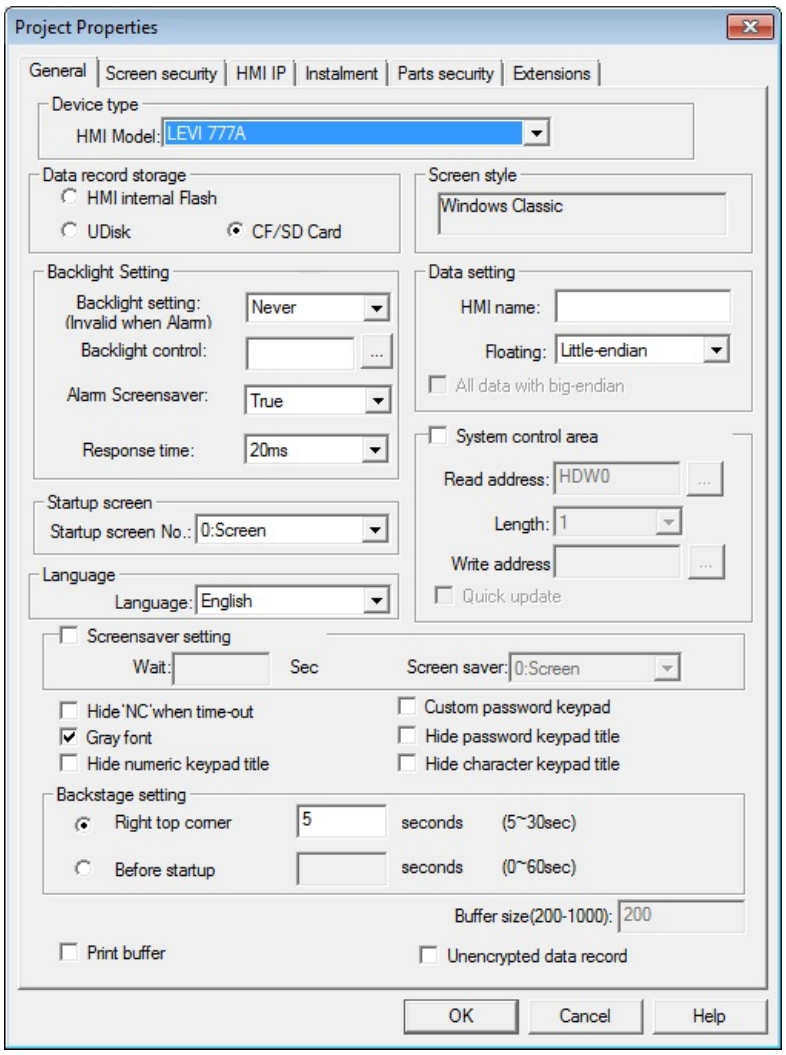


Figure 1

1. Device type:Select proper HMI Model.

2. Data record storage:Choose the storage to save data record.

Property	Description
HMI FLASH	Save data record to HMI internal FLASH, only supply 30M for data storage
CF/SD Card	Save data record to CF/SD Card (Capacity<4GB)
U Disk	Save data record to U Disk (Capacity<4GB)

3. Backlight setting:HMI backlight setting.

Property	Description
Backlight setting	Auto sleep after designated time of inactivity.
	Invalid when alarm: HMI will not sleep when alarm is ON.
Backlight control	HMI will sleep when control bit is ON.
Alarm screensaver	When backlight is enabled, HMI keep sleeping when alarm is ON.
Response time	The minimum time interval for clicking touch screen.

4. Data Setting

Property	Description
HMI name	HMI name
Floating	Floating data format setting, little-endian by default.

5. Startup Screen:The first screen displays on HMI when HMI started.

6. System control area:Use the designated address to change screen display.

Property	Description
Address	Use the designated address to control the screen display.
Length	The length address, 1 by default. The first address: control screen display. The second address: Switch display language (length=2).
Write address	Save current screen display.

7. Screensaver setting:Set wait time and screensaver.

8. Others

Property	Description	
Hide “NC” when time out	Hide “NC” when time out.	
Gray font	To make font display more clear.	
Custom password keypad	Use customized password keypad (No.:1004).	
Hide password keypad tile	Hide password keypad tile when use .	
Hide character keypad tile	Hide character keypad tile when use.	
Hide password keypad tile	Hide password keypad tile when use.	
Quick update	Quick update the address information priority.	
After startup	Keep pressing(for 5 seconds) on right top corner of screen to enter setup screen.	
Before startup	Enter setup screen before startup.	
Print buffer	Valid when “Print record” is enabled (alarm setting).	Save the alarm message to buffer which will be sent to printer later. Buffer size: 200 by default (0-1000).
Unencrypted data record	The created data record file is not encrypt.	

Communication

Click [Setting]-[Communication] to edit the communication parameters of HMI or PLC.

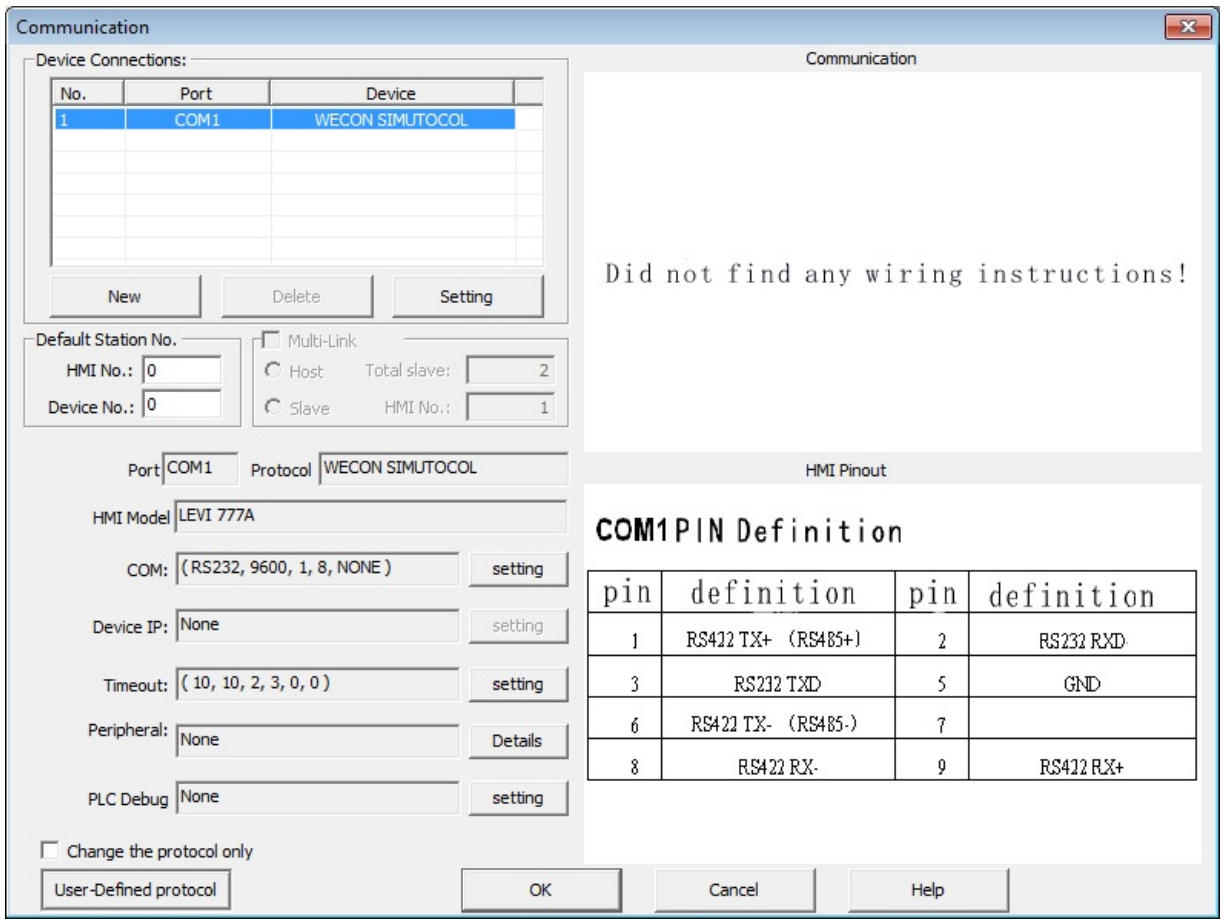


Figure 1

	Properties	Description
Device Connections	New	Add new port and protocol.
	Delete	Delete the selected protocol.
	Setting	Edit selected protocol.
Station No.	HMI No.	Set HMI station No. (0 by default).
	PLC No.	Set PLC station No. (by default).
Multi-link	Host Device	Set HMI as host device.
	Client Device	Set HMI as client device.
COM	Setting	Connect:RS232,RS485,RS422 (COM2 cannot support RS422 connection). Parity bits:EVEN ODD SPACE NONE Data bits: 7 or 8.
Device IP	Device IP	Set the device IP Address.
	Port No.	Set port number.
	Network	TCP_Client_2N:TCP protocol, PLC multi-link . UDP_Client_2N:UDP protocol,PLC multi-link . TCP_Server: TCP protocol,HMI server.
Timeout	Wait timeout(ms)	HMI wait PLC response.
	Receive timeout (ms)	The max time between HMI receive the character.
	Retry times	Retry times when communication timeout.
	Sequential Length	(0: default length 10), read or write the sequential length of device address.
Peripheral		Peripheral device which can work with HMI, including Mini printer, serial keyboard etc.
PLC Debug		PLC debugging parameters setting.
Change the protocol only		The communication parameter do not change when choose a new protocol.
User-defined protocol		Setting about User-defined protocol.
CAN		Setting about CAN protocol.

HMI IP

Click [Setting]-[Project Properties]-[HMI IP] to set HMI IP address.

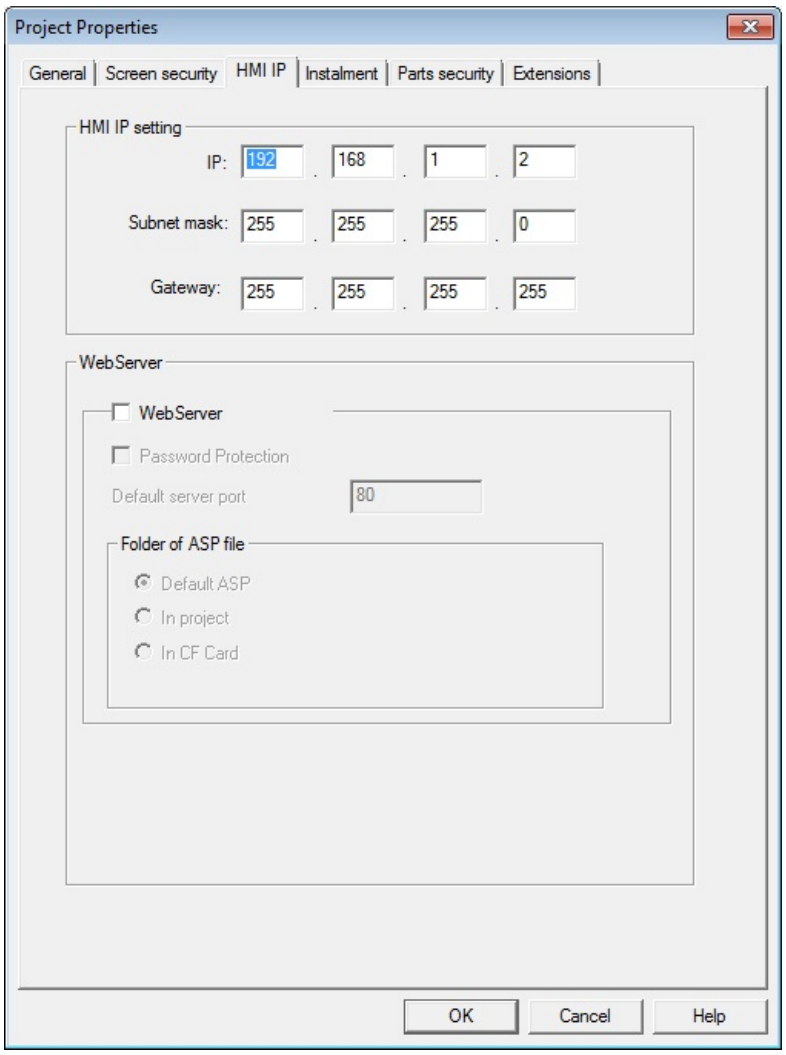


Figure 1

Properties	Description
HMI IP setting	Set HMI IP address.
WebSever	Enable webserver function.
Default server port	Default network port number.
Password protection	Enable password protection on webserver webpage, require password when user access the webpage.(Set user name and password in the UserASP/USER.ASP of software installation folder).
Folder of ASP file	The location to save the webpage file of webserver.

Instalment

The HMI requires the password when it reach expire time.
Click [Setting]-[Project Properties]-[Instalment].

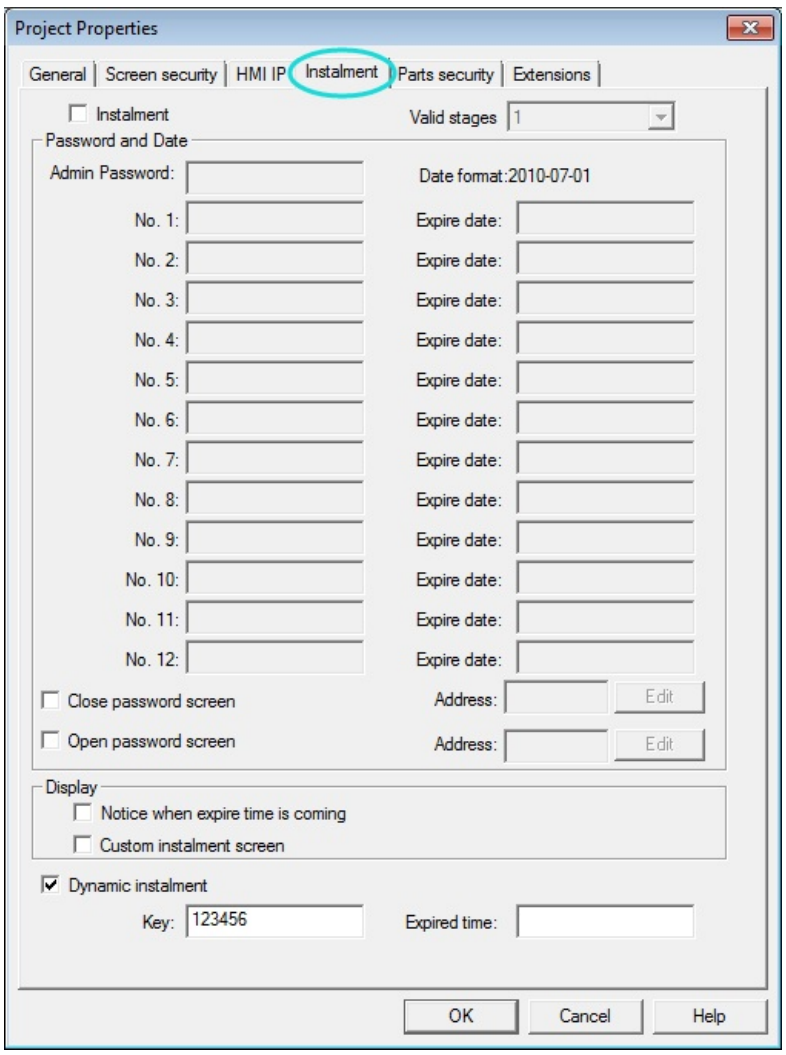


Figure 1

Properties	Description
Instalment	Enable instalment.
Max. payment	Set the maximum number of payments.
Admin. password	The admin password can be used for any payments in this project.
No.1-12	Input the password for each payment, HMI pop up password window when time has come. (Password contain at most 8 digit from a~z, 0~9.
Close password screen	Trigger to close “instalment password screen”.
Open password screen	Trigger to open “instalment password screen”.
Notice when expire time is coming	Remind before the expiration time.
Custom instalment screen	Allow setting the new password for new payment.

Instalment password screen

Display “Input password” when expire time is coming.

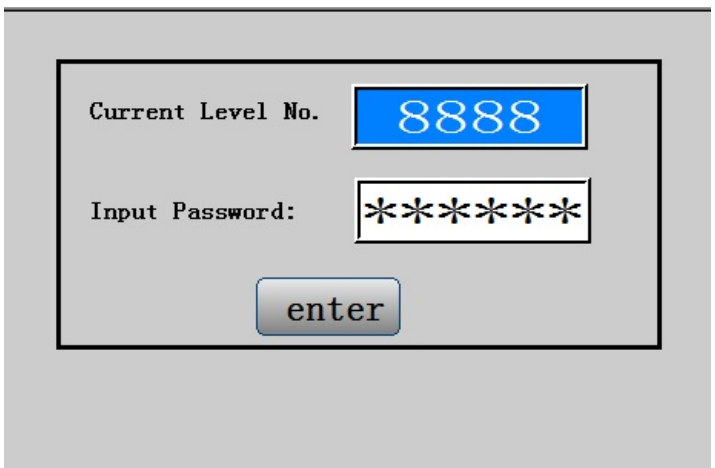


Figure 2

2.Modify installments information on HMI interface. Select “instalment”in the “Function Switch” .

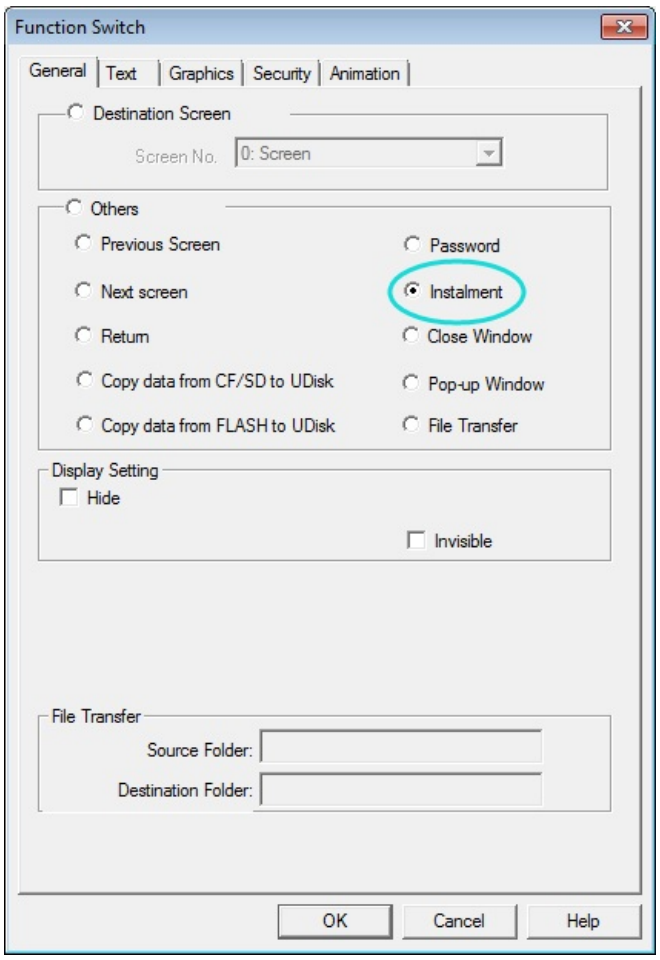


Figure 3

Display instalment screen when click the function switch.

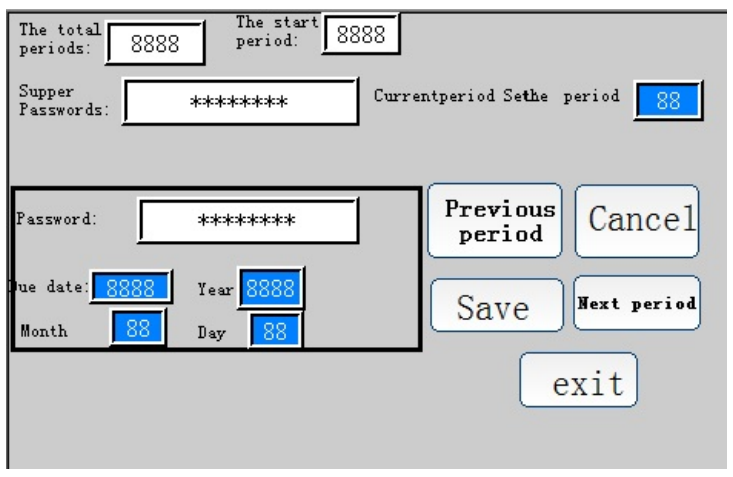


Figure 4

Custom instalment screen

Select [Instalment]-[Custom instalment screen] to enable system built-in screen No. 1009 and No. 1010.

Notice: Do not change the address of objects in “custom instalment screen”.

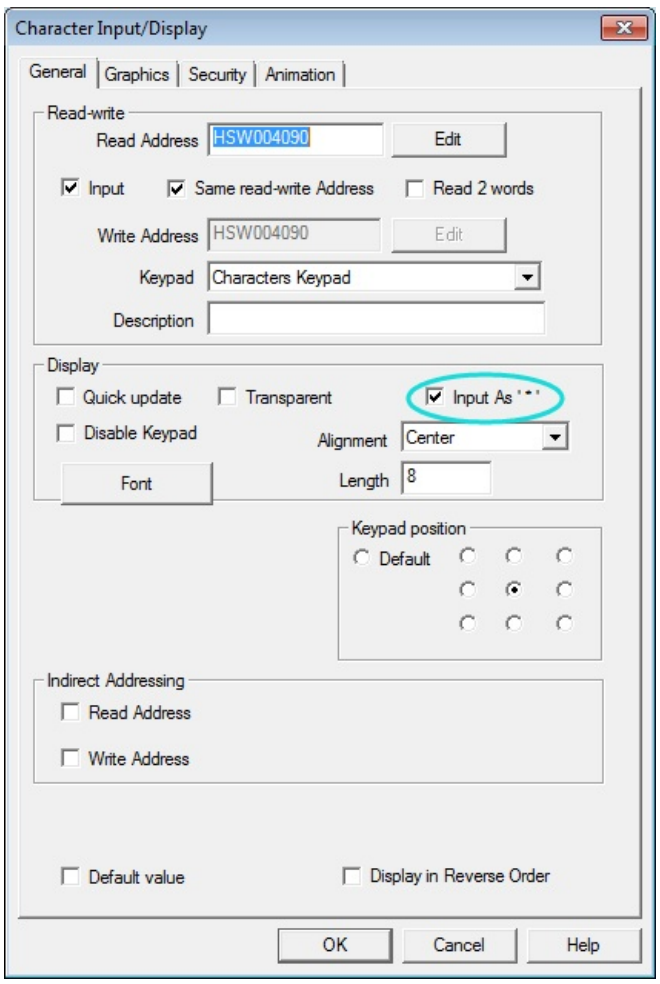


Figure 5

Dynamic Instalment Password

HMI only need “Key” (for first payment) and ”Date” to generate “Password”. So if someone open the original project of HMI, even they cannot get the right key for rest of payments.

Set the “Key” and “Date” in the project when use “dynamic instalment password”.

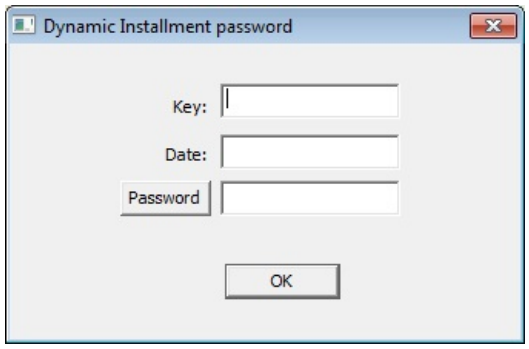


Figure 6

For example:
Date: 2018-10-10
Key: 0d809192
Next expire date: 2018-10-10 (HMI display password screen when data reach 2018-10-10).
Notice:
1. Key can only contain 6 digits, from 0~9, a~z.
2. Date format: 2014-12-12.

Security

HMI provides 3 kinds of protection: Screen Security, Button password and Parts security.

Screen security: Input the password to access the screen.

Button password: The screen is automatically locked after designated time of inactivity.

Parts security: Input password to operate the objects with corresponding level.

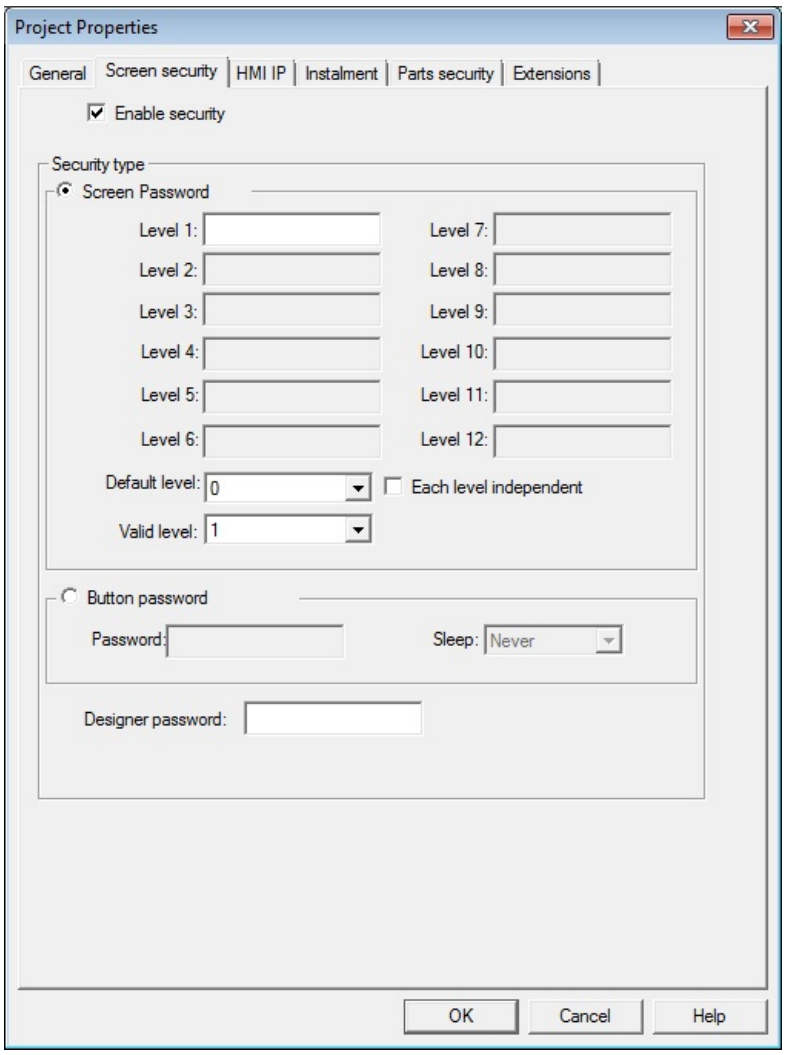


Figure 1

Parts Security

HMI require password when operating protected objects. The parts security protects the equipment from false operation without permission.

Click [setting]-[Project Properties]-[Parts security] to open the following dialog.

There are 12 security levels in the parts security; the higher level means the higher authority. In most case, HMI require password when operate object with the higher security level. HMI will always require password when select “Each level independent” .

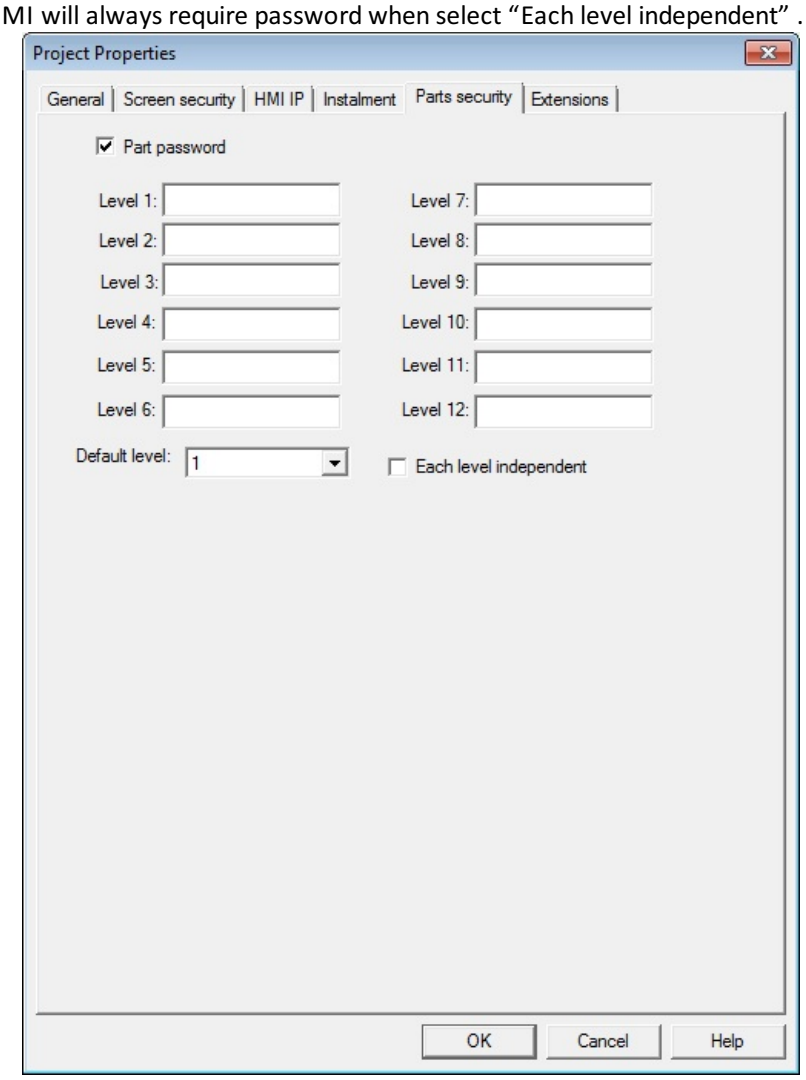


Figure 1

Properties	Description
Parts security	Enable parts security.
Level 1	Level 1 security passwords, System Address HSW452~HSW455, 8 bytes.
Level 2	Level 2 security passwords, System Address HSW456~HSW459, 8 bytes.
Level 3	Level 3 security passwords, System Address HSW460~HSW463, 8 bytes.
……	……
Initiate level	Default level when HMI startup.
Each level independent	HMI always require password when operate all objects with different part security level.
Objects properties “Always require password”	HMI always require password when operate all objects, even all objects with same part security level.

Screen Security

HMI require password when accessing protected screen with higher level. The screen security protects the screen from false operation without permission. Screen security is available for both basic screen and sub screen.

There are 12 security levels in the screen security; the higher level means the higher authority. In most case, HMI require password when access object with the higher security level. HMI will always require password when select “Each level independent”.

Click [setting]-[Project Properties]-[Screen security] to open the following dialog.

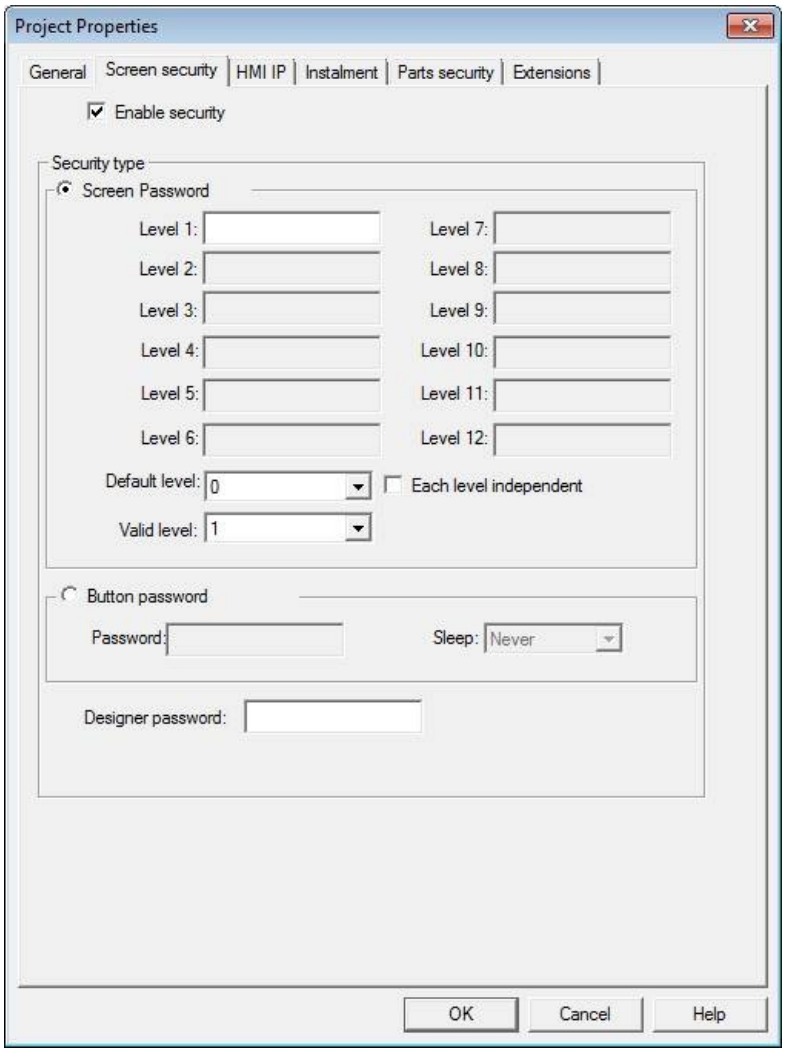


Figure 1

Properties	Description
Level 1	Level 1 security passwords ,System Address HSW404~HSW407, 8 bytes.
Level 2	Level 2 security passwords ,System Address HSW408~HSW411, 8 bytes.
Level 3	Level 3 security passwords ,System Address HSW412~HSW415, 8 bytes.
……	……
Initial level	Initial level of HMI when startup.
Valid Level	The maximum level will be enabled.
Each level independent	HMI always require password when accessing the screen with different security level.

Notice: The password should not contain more than 8 bytes.

Edit the screen properties to set its security level.

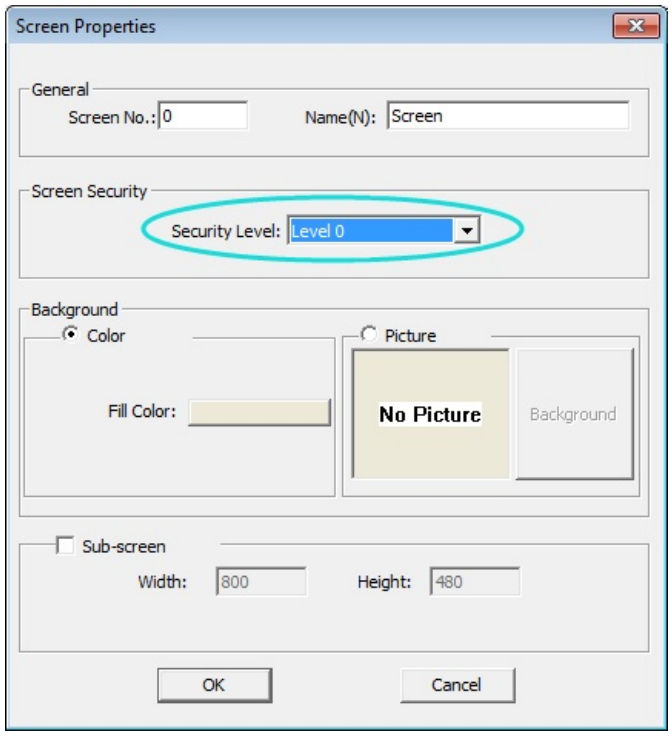

Screen Properties
General
Screen No.: 0
Name(N): Screen
Screen Security
Security Level: Level 0
Background
Color
Fill Color:
Picture
No Picture
Background
Sub-screen
Width: 800
Height: 480
OK
Cancel

Figure 2

Screen Lock

Set HMI to automatically lock itself after a specified period of inactivity. Once the set period of time expires, the computer will be locked and enter the password to unlock it.

In most case, add the “Function Switch”(with full screen size) to “Common screen”.

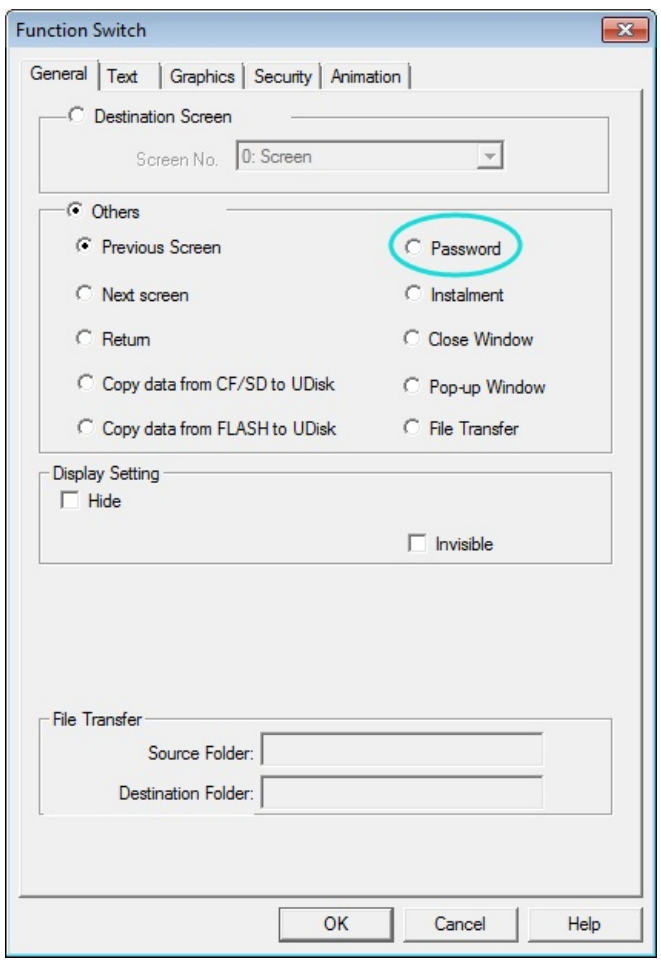


Figure 1

Set a specified period of inactivity in [Setting]-[Project Properties], enter the password and time.

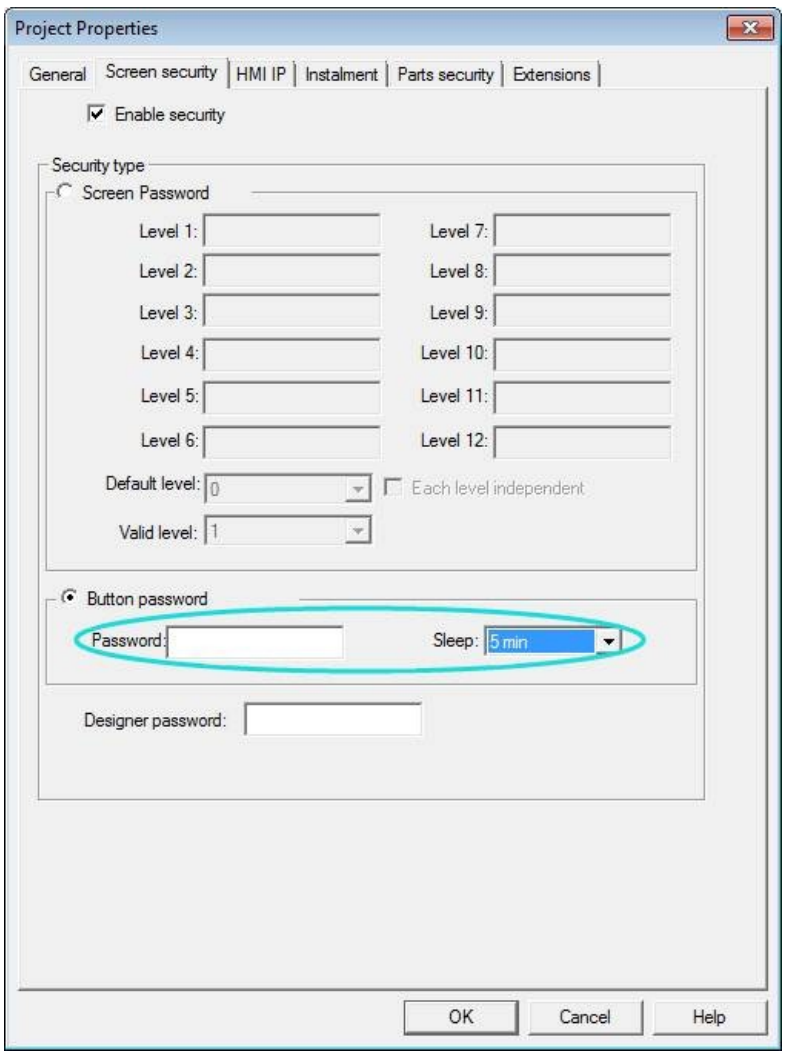


Figure 2

Extensions

HMI provides some special functions to HMI, it is optional, select the function if needed.

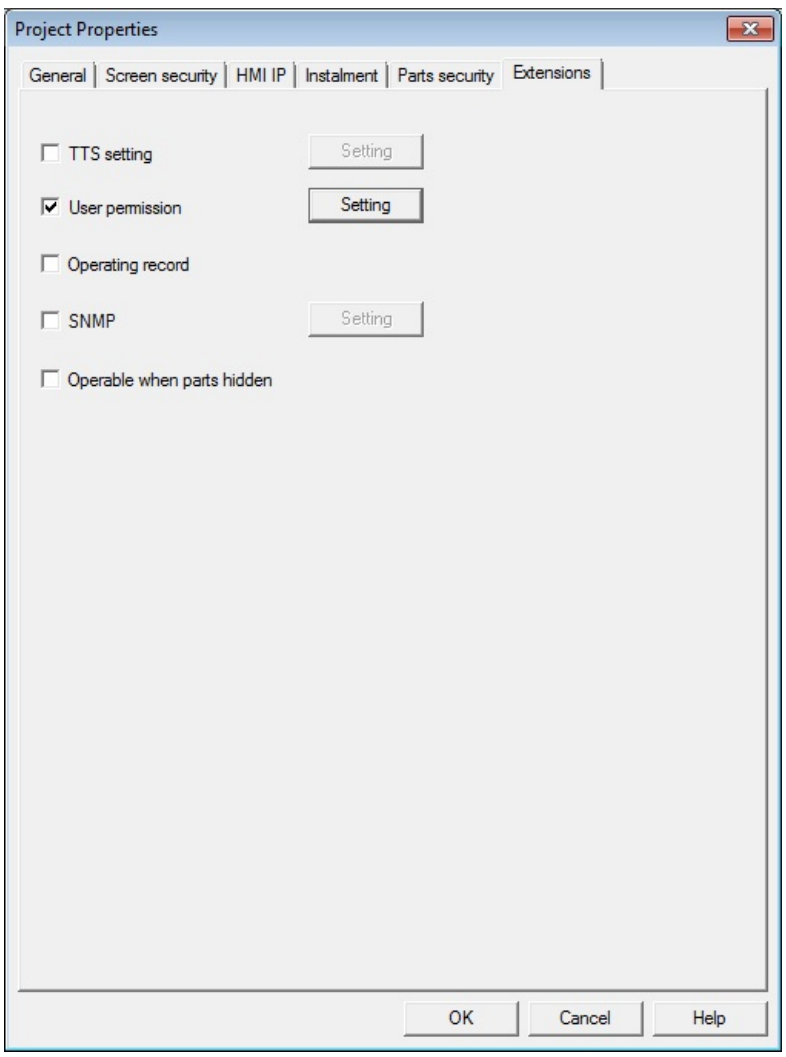


Figure 1

User Restriction

General

HMI allows accessing by multiple users. User need to set the user and group when designing project. Different group have different permission level for accessing. Each user should be added to certain group, it is possible to add one user to different group.

Notice:User permission and parts security cannot be used in same object.

Operating record: the detail information about the operation of HMI, find the record file in \\flash\\UserOperationLogs.txt (On simulator C:\\UserOperationLogs.txt).

Setting

1. New User and Group:

Click [Setting]-[Project Properties]-[Extensions] to open the following dialog.

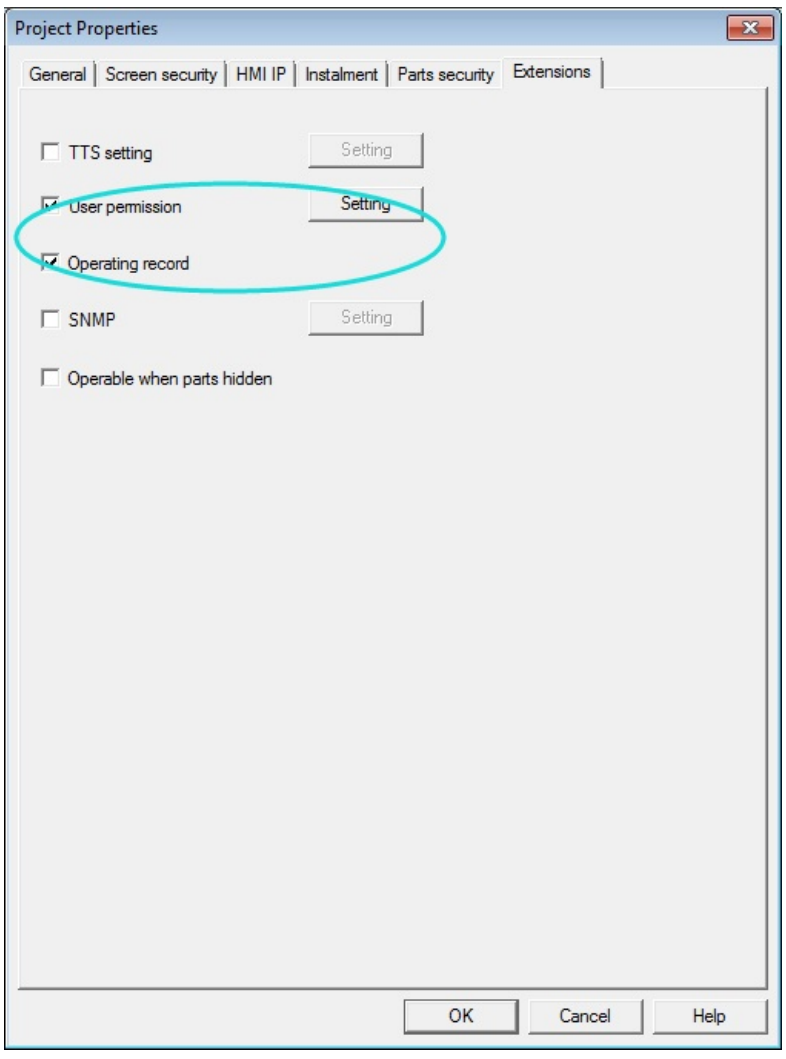


Figure 1

Add “New Group” to following dialog.

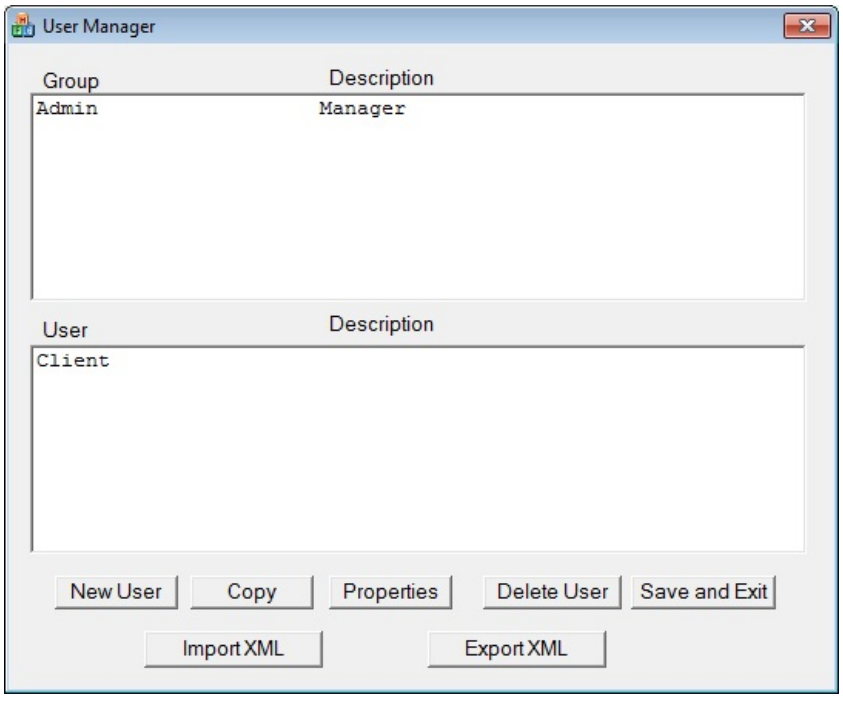


Figure 2

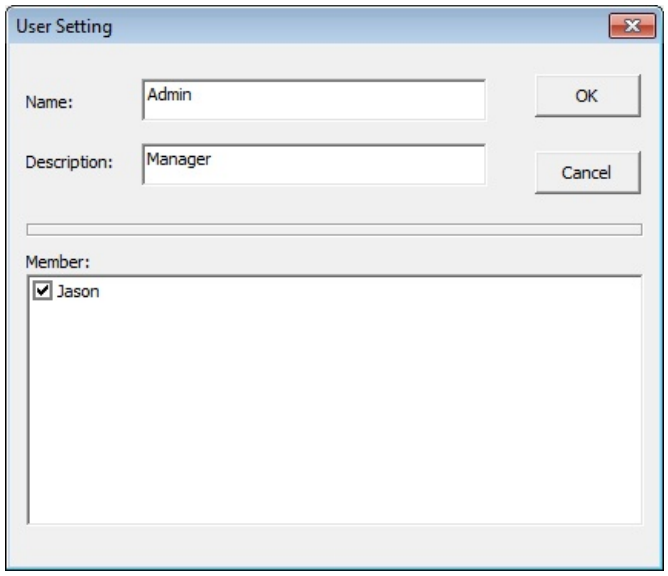


Figure 3

Set the password for each user, then add the user to existing group.

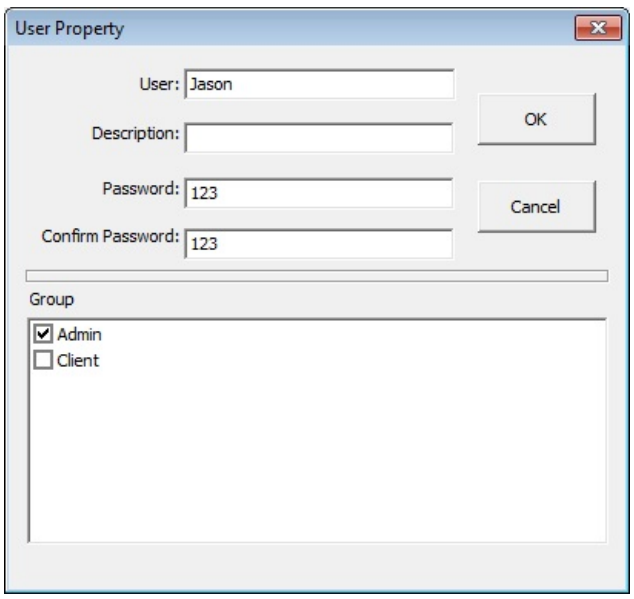


Figure 4

Or create a “New group”, then select existing user in following list.

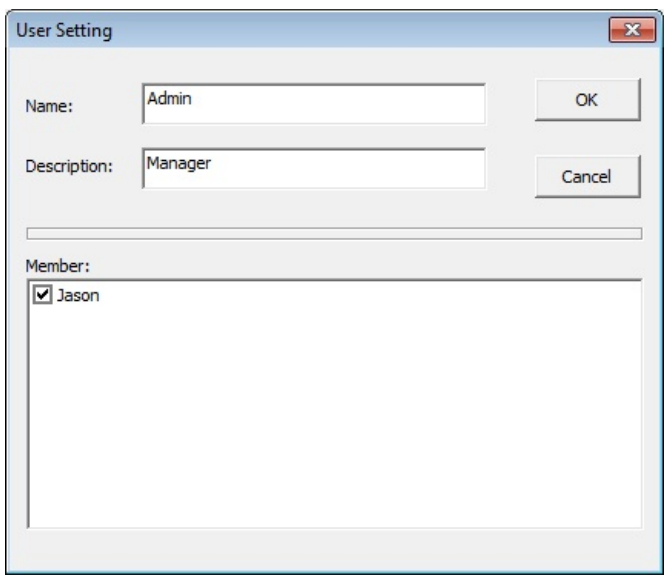


Figure 5

2. Set User permission

Double Click on objects, then edit its properties, [Security]-[User Restriction]-[User Setting]:

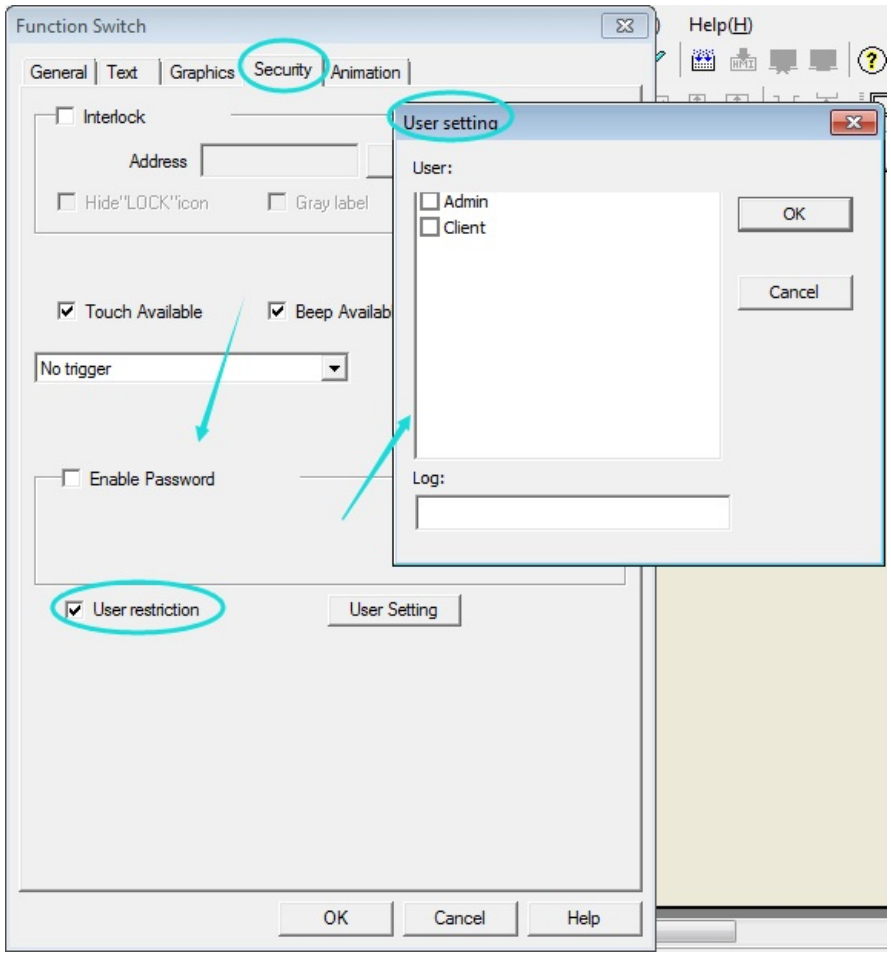


Figure 6

3. Operating Record

Record the detail information about the operation of HMI.

User Management

HMI allows managing user accounts on screen. Including adding, deleting and editing the user account. HMI provide built-in screen for “sign in “and “password”(screen No.1011 and 1012).

Function	Address	Object type	Address function
Log in	HSW10060~10148	Drop down list	User name
	HSW10002	Character input object	Password
	HSX10000.0	Bit switch(On)	OK (log in)
Change password	HSW10061	Character display object	User name
	HSW10002	Character input object	Old password
	HSW10006	Character input object	New password
	HSW10010	Character input object	Confirm password
	HSX10000.1	Bit switch(On)	OK(change password)
Log out	HSX10000.2	Bit switch(On)	Log out
New user	HSW10014	Character input object	User name
	HSW10006	Character input object	password
	HSW10010	Character input object	Confirm password
	HSW10149~10260	Drop down list	Select group
	HSX10000.3	Bit switch(On)	OK(add new user)
Delete user	HSW10060~10148	Drop down list	User name
	HSX10000.4	Bit switch(On)	OK (delete user)
Select group	HSW10060~10148	Drop down list	User name
	HSW10149~10260	Drop down list	Group name
	HSX10000.5	Bit switch(On)	OK(join)
delete	HSW10060~10148	Drop down list	User name
	HSW10149~10260	Drop down list	Group name
	HSX10000.6	Bit switch(On)	OK (remove)
Delete Profile	HSX10000.8	Bit switch(On)	OK(delete)
Export Profile	HSX10000.9	Bit switch(On)	OK(export)
Import profile	HSX10000.10	Bit switch(On)	OK(import)
Export log file	HSX10000.11	Bit switch(On)	OK(export)

Delete log file	HSX10000.12	Bit switch(On)	OK (delete)
Current system info	HSW10022~10039	Character display object	

The notification in HSW10001 shows the current state of HMI.

Value of HSW10001	Description
1	Insufficient permissions.
2	User name does not exist.
3	User name already exists.
4	Invalid password.
5	Log in successfully.
6	Password you inputted do not match.
7	Password changed.
8	User adding complete.
9	User deleting complete.
10	Maximum number of users exceeded.
11	User already exists in the group.
12	Completed adding user to group.
13	User not in the group.
14	User delete successfully.
15	Import file successfully.
16	Failed to Import file.
17	Export file successfully.
18	Export file failure.
19	Log out.
20	Delete Profile successfully.
21	Delete log file successfully.

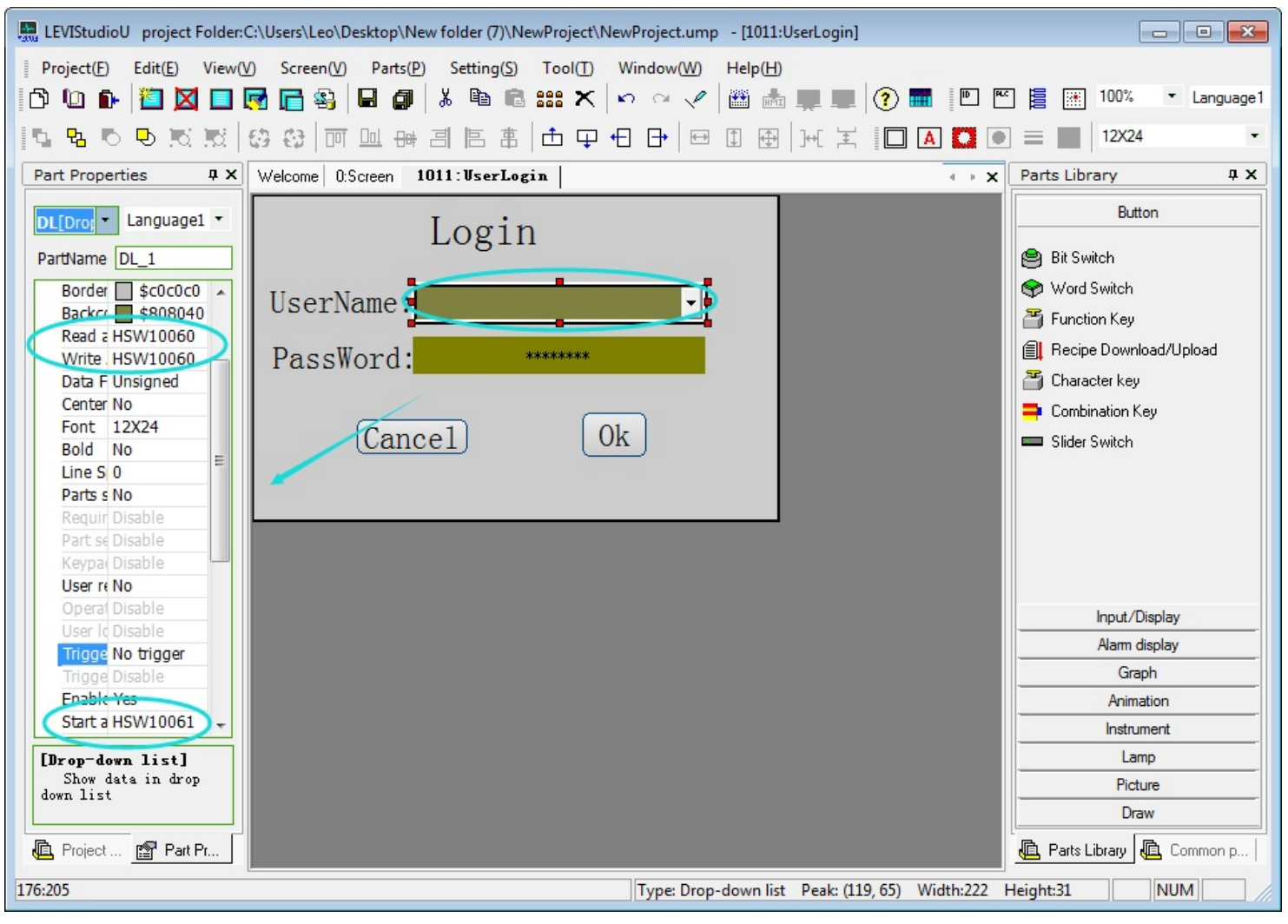


Figure 7

User Permission

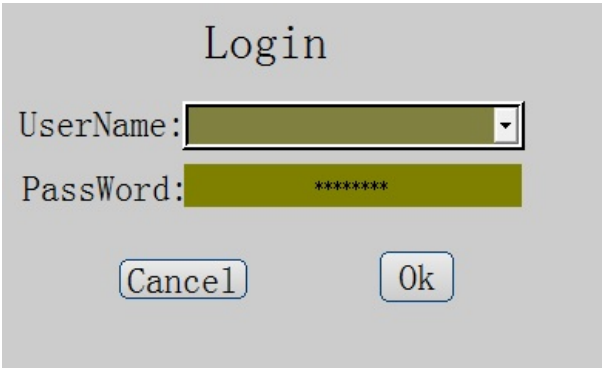


Figure 8

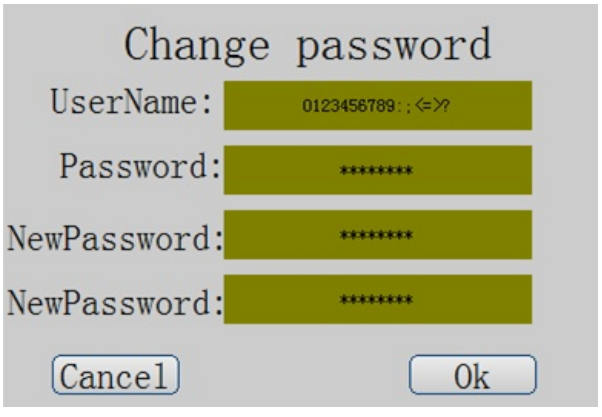


Figure 9

TTS Function

Setting

1. TTS function require additional voice module.

2. Click [Setting]-[Project Properties]-[Extensions] to enable “TTS function”.

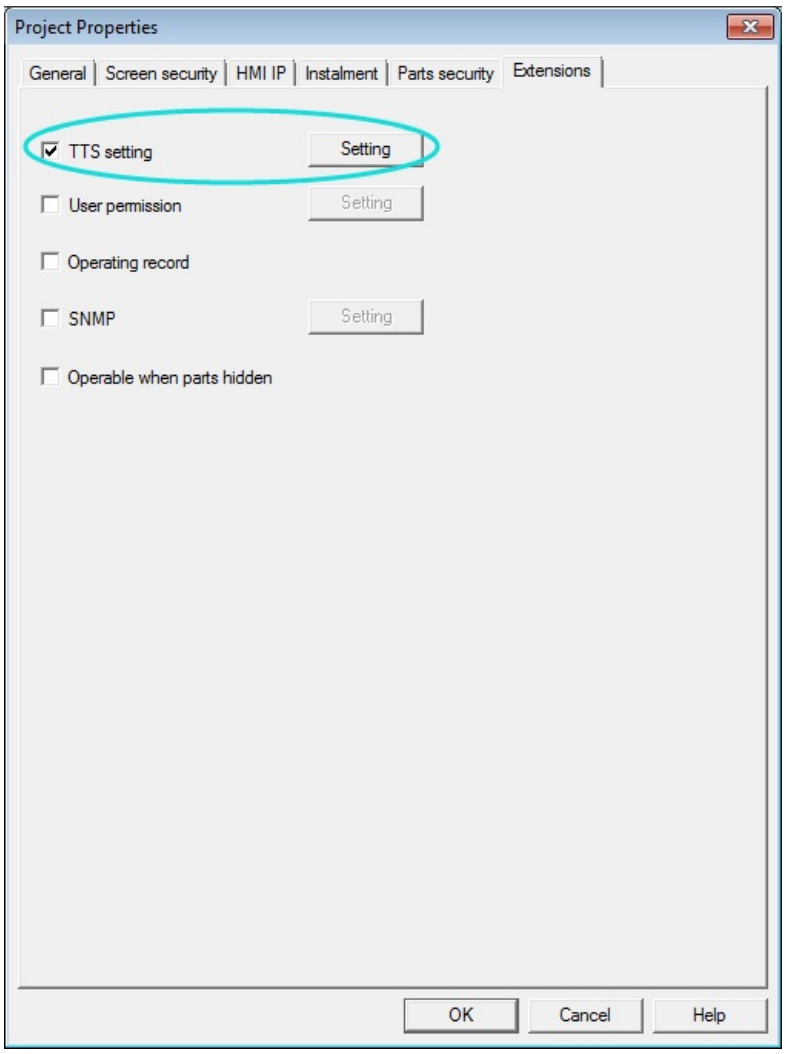


Figure 1

Audio Setting

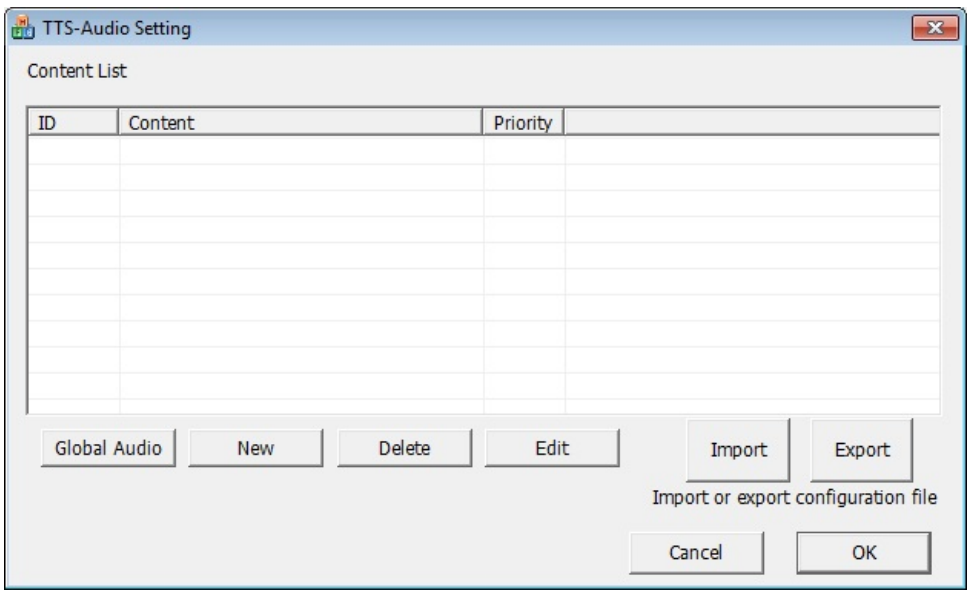


Figure 2

1. Global Audio: Audio setting.
2. Import or export configuration file.

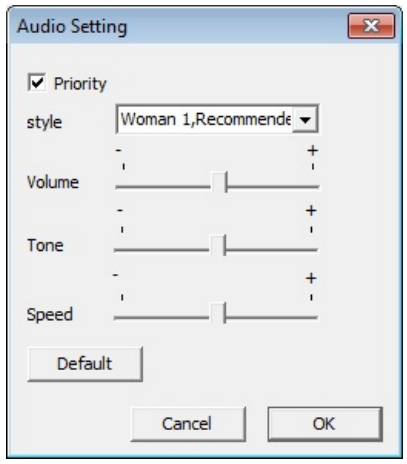


Figure 3

Audio Settings

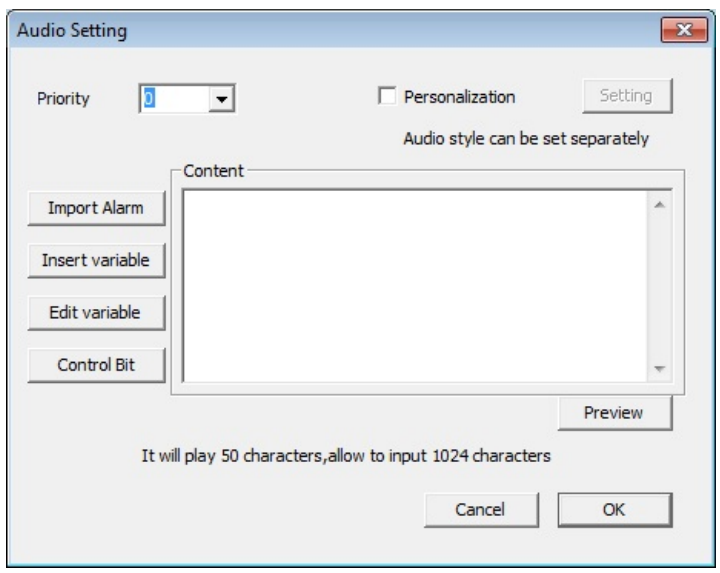


Figure 4

1. Priority: the higher priority the have the priority when multiple trigger at same time.
2. Content: input the content in the text box.
3. Personalization: Audio style can be set separately when start.
4. Preview: Listen test. (Notice:need to link the usb to interface that have TTS module, only can preview first 50 characters)
5. Control Bit: Insert the character to current text box.
6. Import alarm: Insert the alarm content.

Control bit

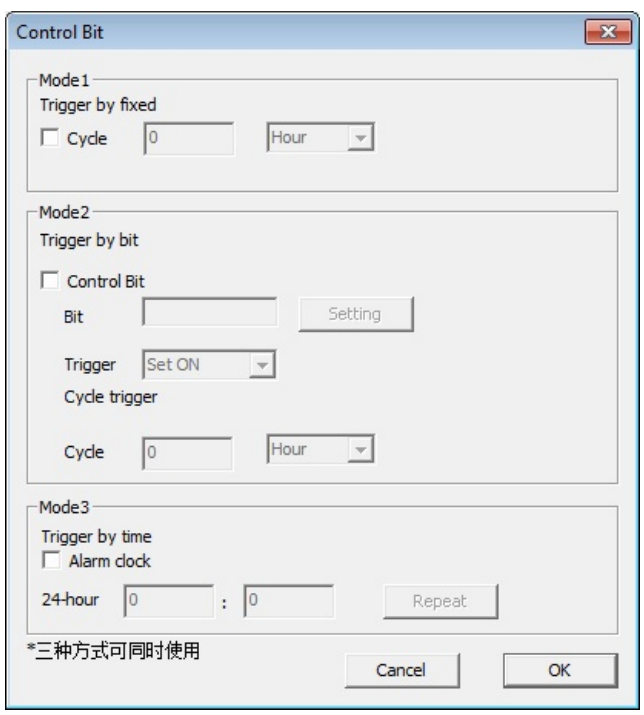


Figure 5

1. Trigger by fixed: cycle.

2. Trigger by bit: audio output when the trigger address meet demand. Cycle trigger when ON and OFF.

3. Alarm clock:24-hour,trigeer by time.

Text Label

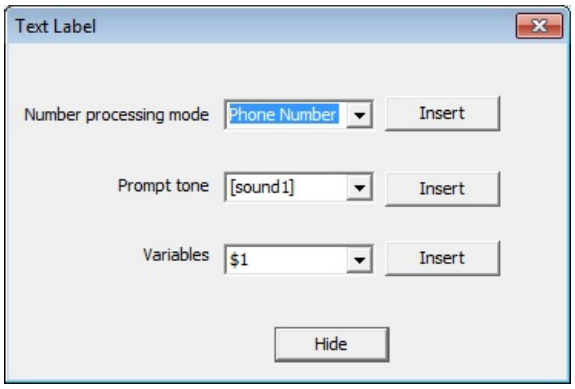


Figure 6

1. Number processing mode: the number will pronunciation after insert.

2. Prompt tone: prompt tone when come across the sign.

3. Variables: pronounce according to the variables address.

Edit variable

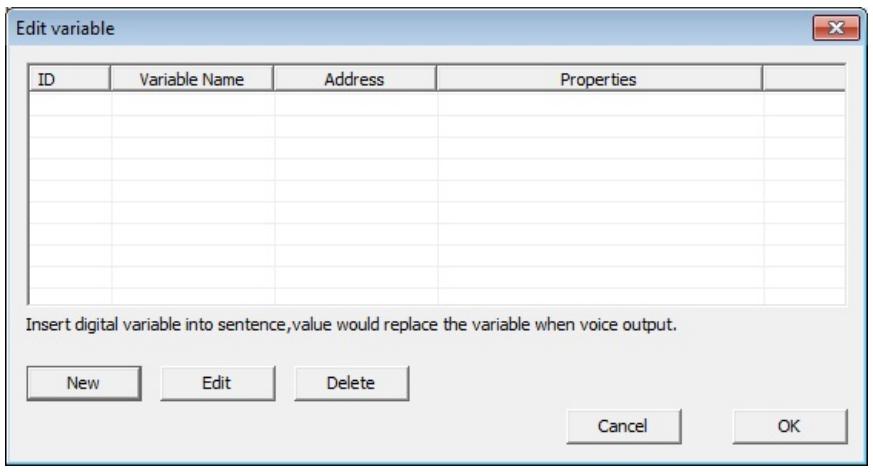


Figure 7

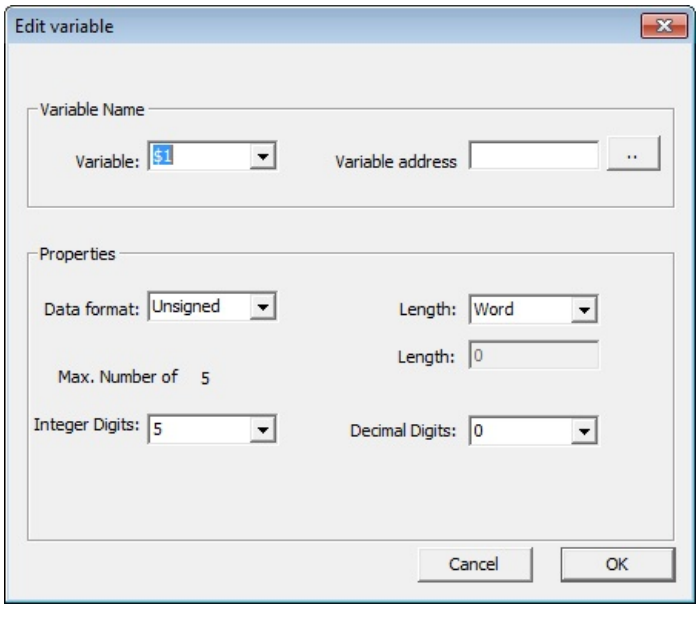


Figure 8

Picture

All pictures of project can be found in [setting]-[Gallery].

Picture could be:

1. Single state picture.
2. Multi-state picture.
3. GIF format picture.

Gallery

User could use customized pictures from “Gallery”.

Two types of pictures in Gallery:

1. User picture: Import picture by user.
2. System picture: Software provide the different kinds of picture for user programming.

Select from [setting]-[gallery].

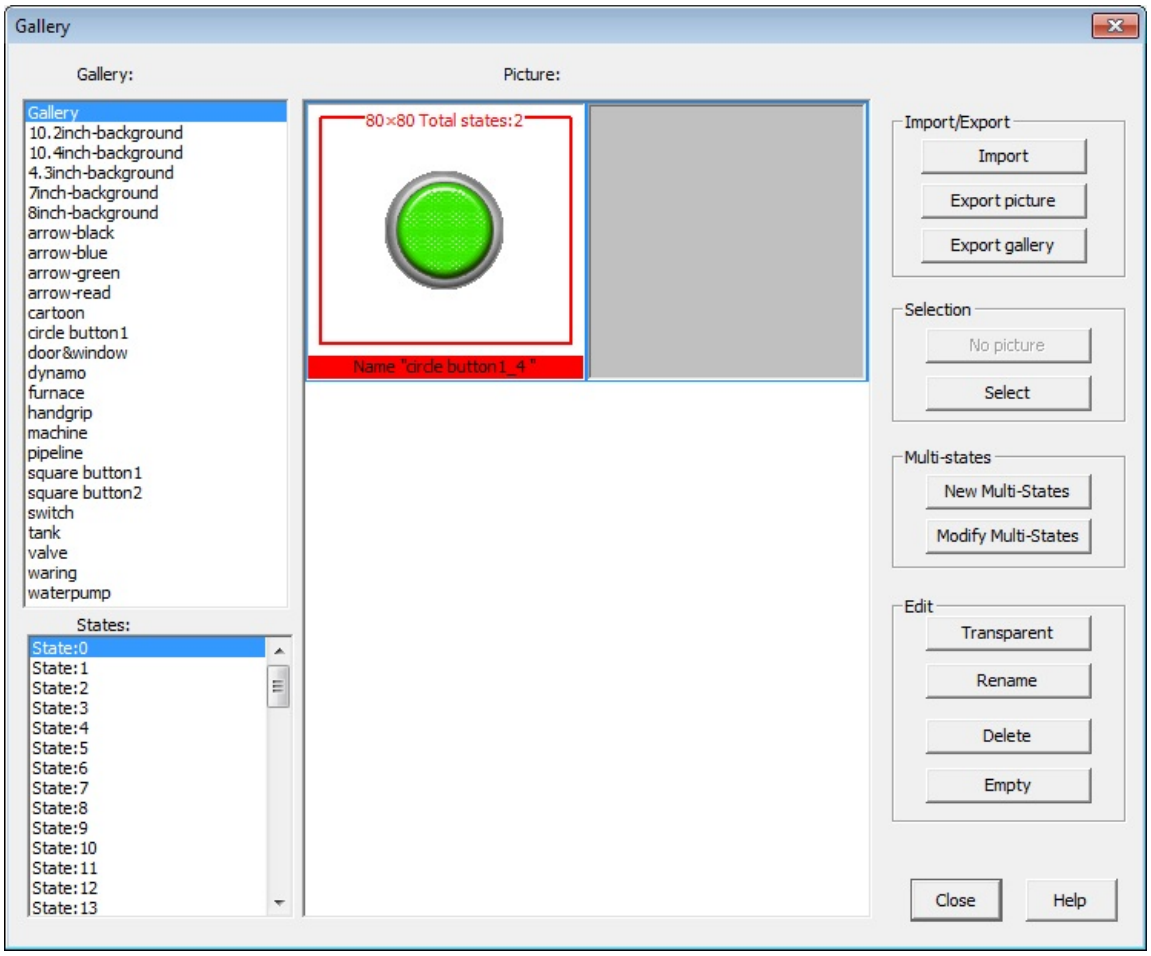


Figure 1

Properties	Description
Import	Import picture from PC.
Export	Export picture to PC.
Export gallery	Export full gallery to PC in blb format.
Multi-states	New: Create a new multi-states.
	Edit: Edit existing multi-states.
Selection	Select or unselect a picture.
Transparent	Make designated color transparent.
Rename	Rename picture (name must be unique).
Delete	Delete picture that choose currently.
Empty	Delete all of picture in Gallery.
Close	Close Gallery window.

Add Picture

Add pictures to [gallery]:

1. Import a picture from PC.

2. Create a new multi-states.

3. Select picture from software built-in gallery.

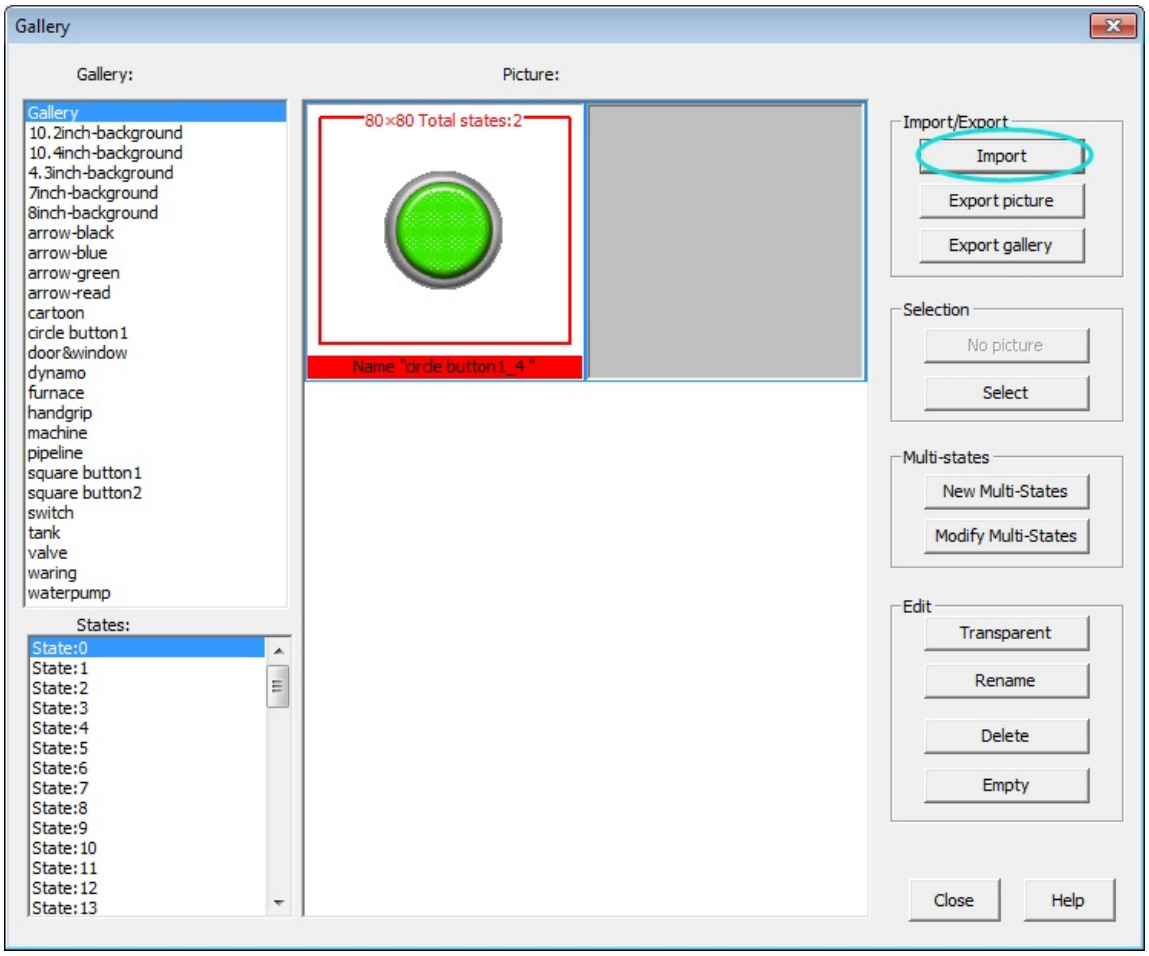


Figure 1

New Multi-states

Multi-states consist of common pictures , those pictures should be added to multi-sates in certain order.
Click [setting]-[gallery]-[new multi-states].

Notice: Gif format picture cannot be transparent.

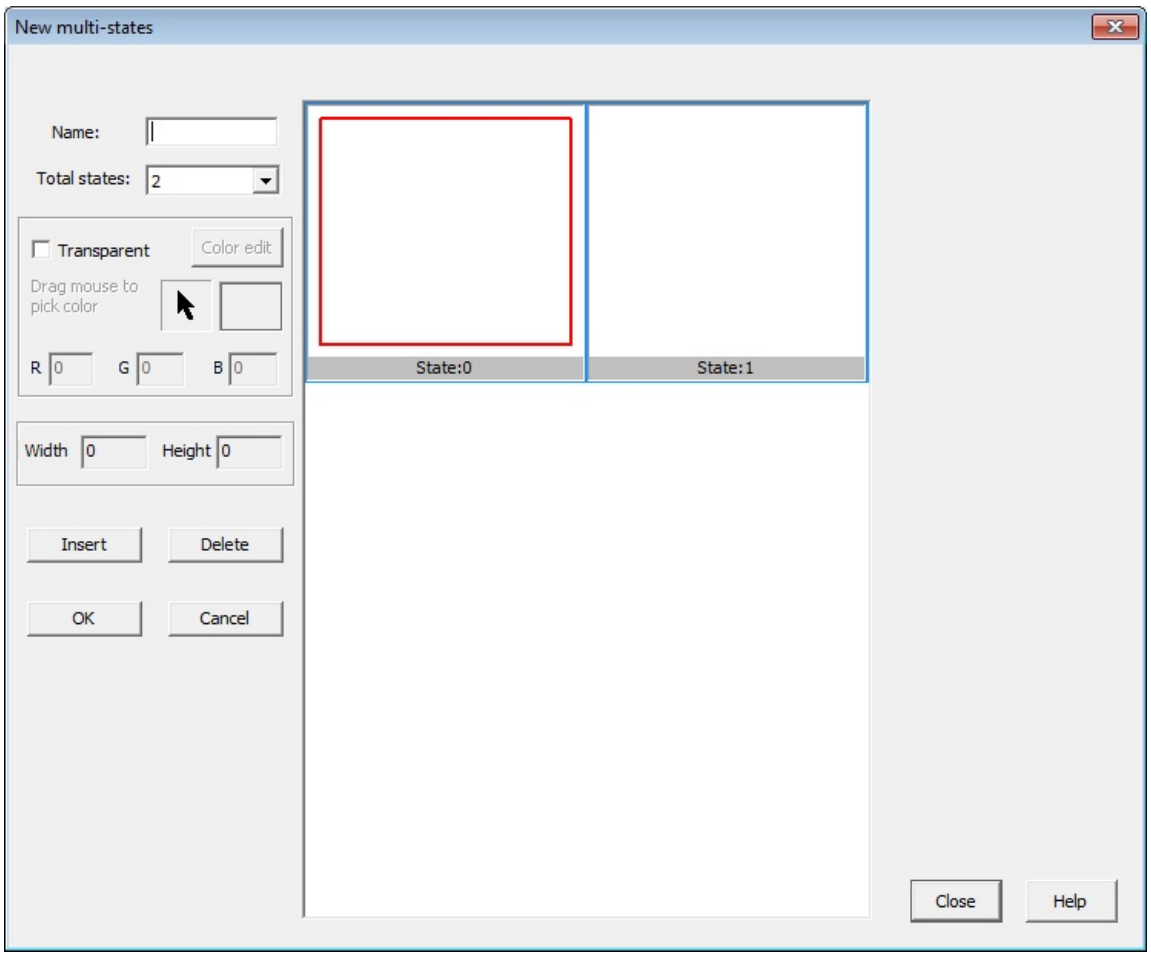


Figure 1

Property	description
Name	Name of new picture.
Total states	Total states of new picture, support up to 32 states.
Transparent	Make designated color transparent.
R/G/B	The digital of transparent color.
Drag mouse to pick color	Select the color.
Width/height	The size of selected picture.
Insert	Insert picture to multi-states.
Delete	Delete picture from multi-states.
Ok	Save changes to multi-states.
Cancel	Cancel and close the dialog.

Select Picture

Select a picture for the object from “Gallery”.

Notice: unable to select both vector and picture in one multi-states.

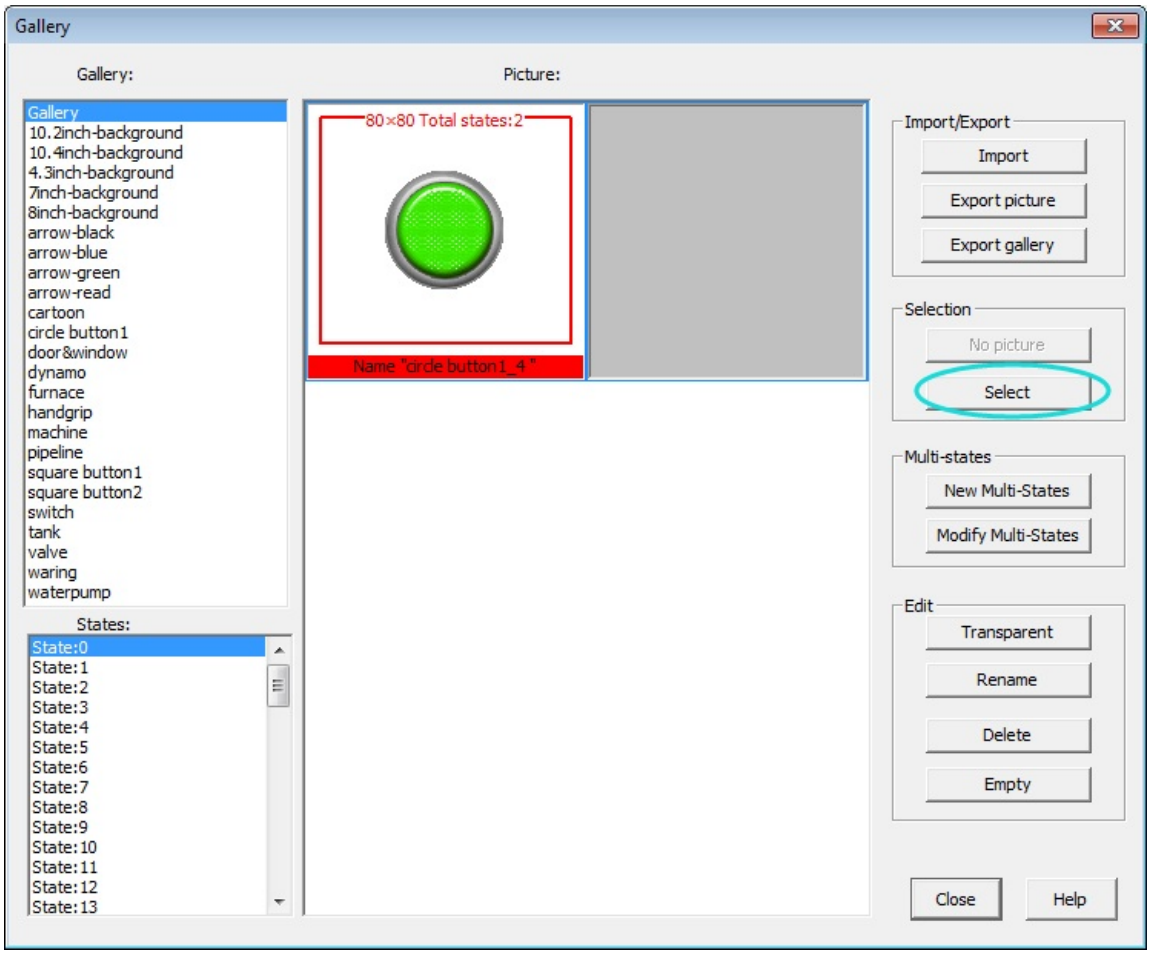


Figure 1

Address

1. Address length:

Word address: each address occupies 16 bits.
Byte address: each address occupies 1 bit.

2. Address type:

Internal address of HMI: HMI local address, including HDW/HDX,HSW/HSX and so on.
Device address: PLC address or other peripheral device.

Address Format

HMI support connecting with PLC devices by different COM port, also support one COM port to communicate with more than one device;

[Connection] #[Station No.] :[Address]
Connection: COM port number (range 1-3).
Station No. :To mark the different device in same COM port.
Address: Device address number.

Notice:
Connection and station number can be omitted.
*for examplae “A#B:C”means that connection number is “A” , station number is “B”and access address “C”.

Address Editor

Address editor is a dialog to edit address, including connection, station number and address of device.

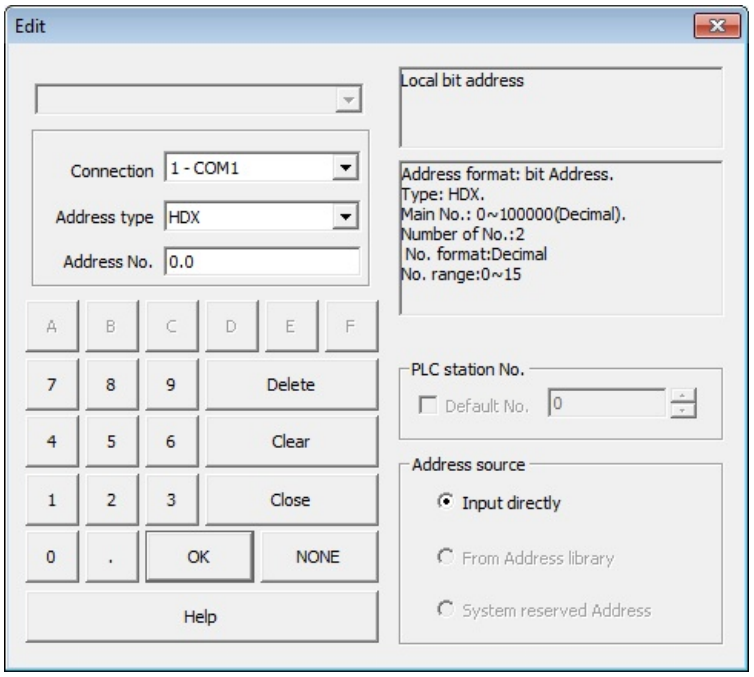


Figure 1

Property	Description
PLC station No.	Set device station number for address.
Input directly	The address is inputted by user.
From address library	Choose address from address library.
System reserved address	Choose address from the system address.
Connection	COM port number (range 1-3).
Address type	Choose address type.
Address No.	Choose address number.

Address Library

Address library contains commonly used addresses. It not only avoids setting the addresses repeatedly but also expresses the function of an address more clearly.

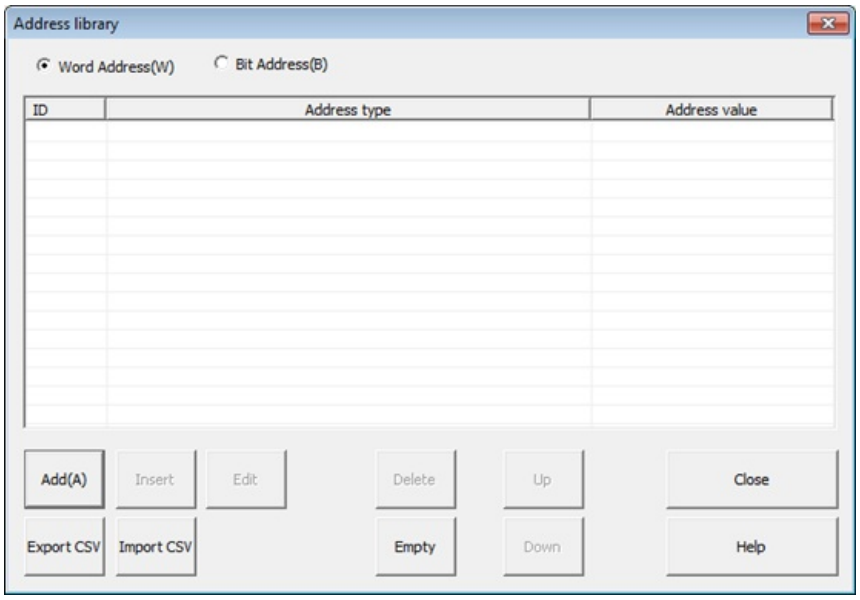


Figure 1

Property	description
Word address	List of all predefined word addresses.
Bit address	List of all predefined bit addresses.

Objects List

List of all addresses in current project, user can see the all of usage address.

Click [View]-[Objects list] from the menu bar.

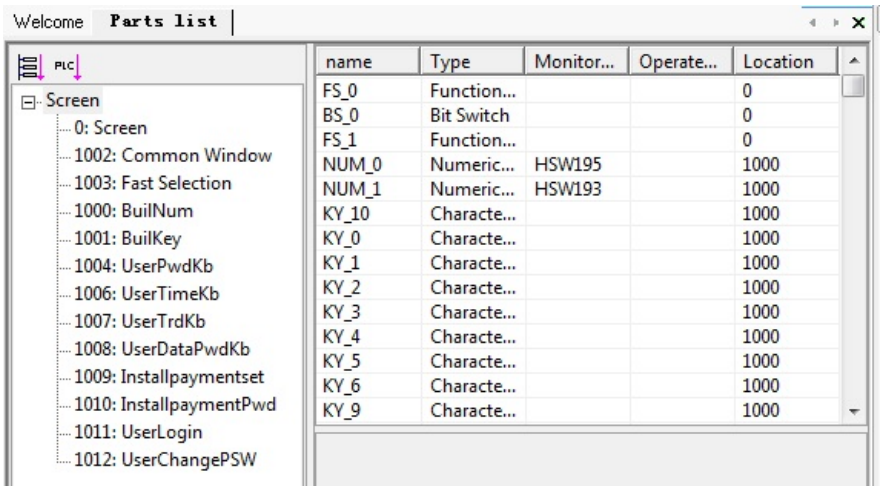


Figure 1

Address Mapping

Description:

Address mapping is the operation to build the relationship between two different addresses. Forwarding address from source address to destination address, so the value in both addresses would be same according to the predefined mapping mode. Both addresses could be from different, in this way, to make HMI processing faster.

*For example: forward “D0” to ”HDW100” (length: 10), so the address D0,D1,……D9 would be related to HDW100,HDW101……HDW109.

Function:

1. Mapping Mode(Read only): Destination address only read the value from source address.it is unable to write value to source address.

2. Mapping Mode(Read/Write): The value of source and destination address always keep same.

3.Mapping control: Set the control bit to start or stop mapping.

4. The maximum length of mapping group is 2048.

Font Library

Support all of Windows font display in HMI screen, allow adding up to 32 kinds of font display in one project.
Click [Setting]-[Font library] on the menu bar.

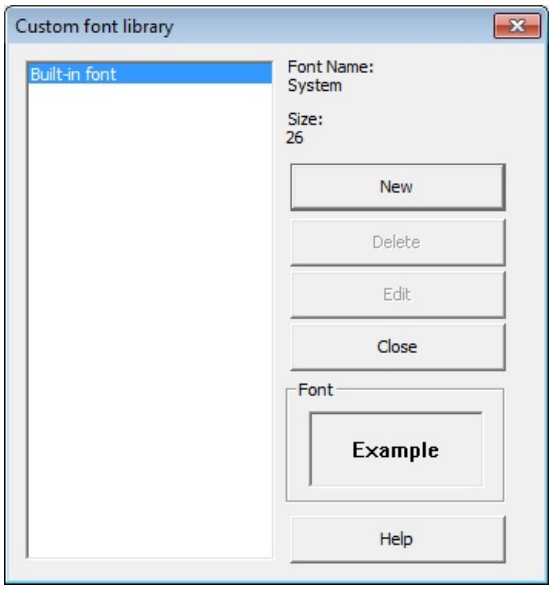


Figure 1

Property	Description
List	List of font type in font library.
Font name	Font name of selected font.
Size	Font size of selected font
Example	Preview of selected.

Custom Font

Objects can use custom font.

Click [Font] on object [text], click [new] to add a new font.

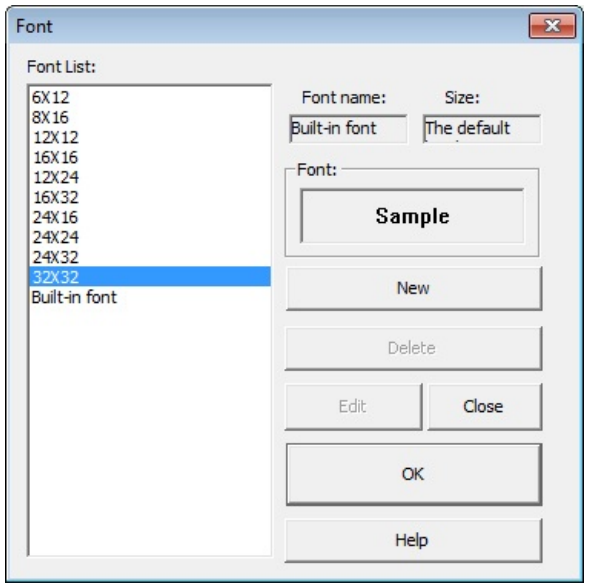


Figure [

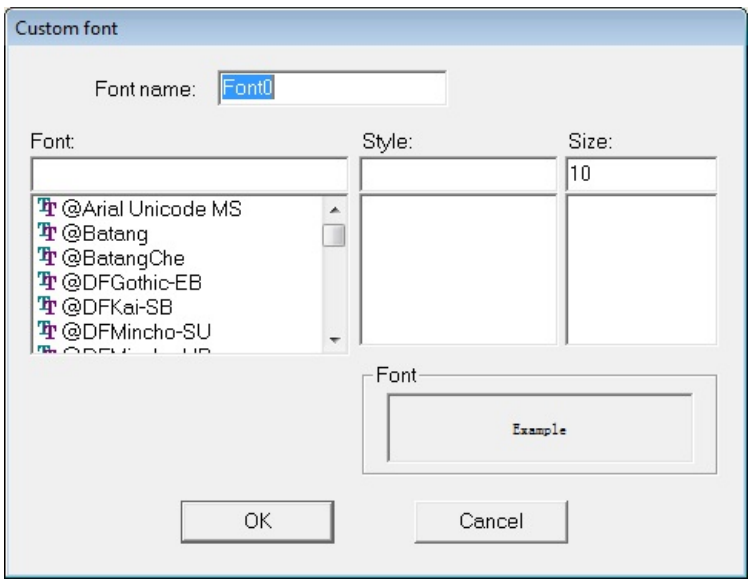


Figure 2

Property	Description
Font name	Font name (must be unique).
Font	Current selected font.
Style	Font style, including Regular, Bold, Italic, Bold Italic.
Size	Font size.

Text Library

Text library contains commonly used text. It avoids setting the text repeatedly.

Click [setting]-[Text library] to open the following dialog.

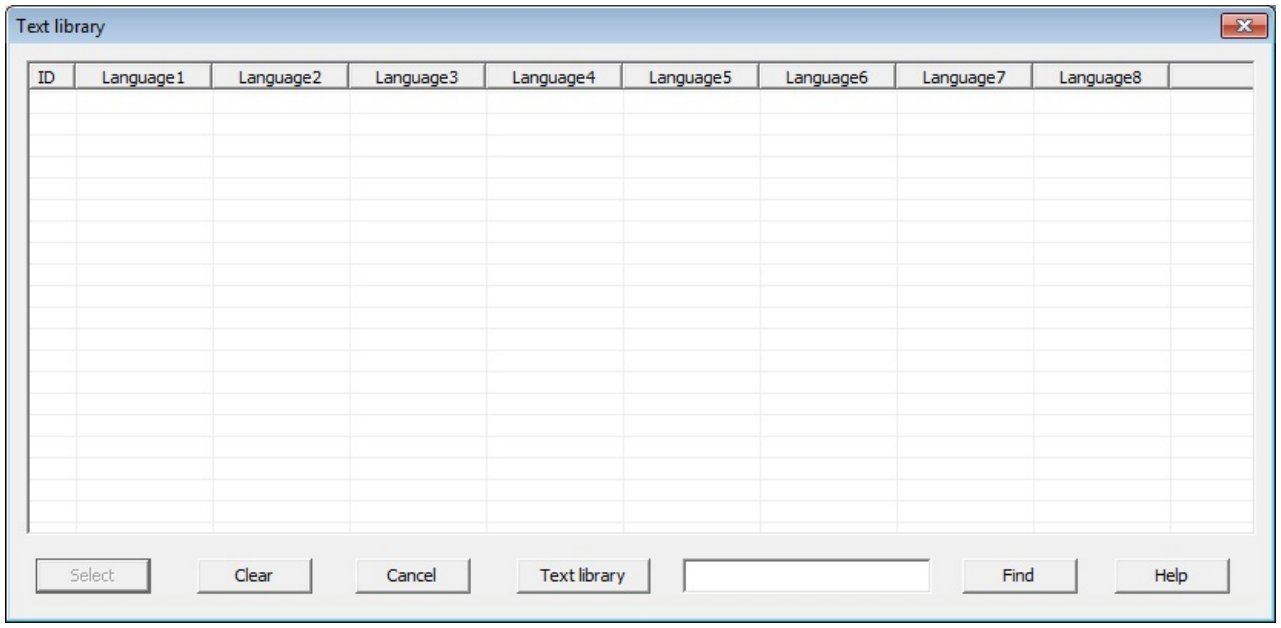


Figure 1

Property	Description
ID	Automatic ID for the predefined text.
Language1	The first language content (256 bytes).
Language2	The second language content (256 bytes).
Language3	The third language content (256 bytes).

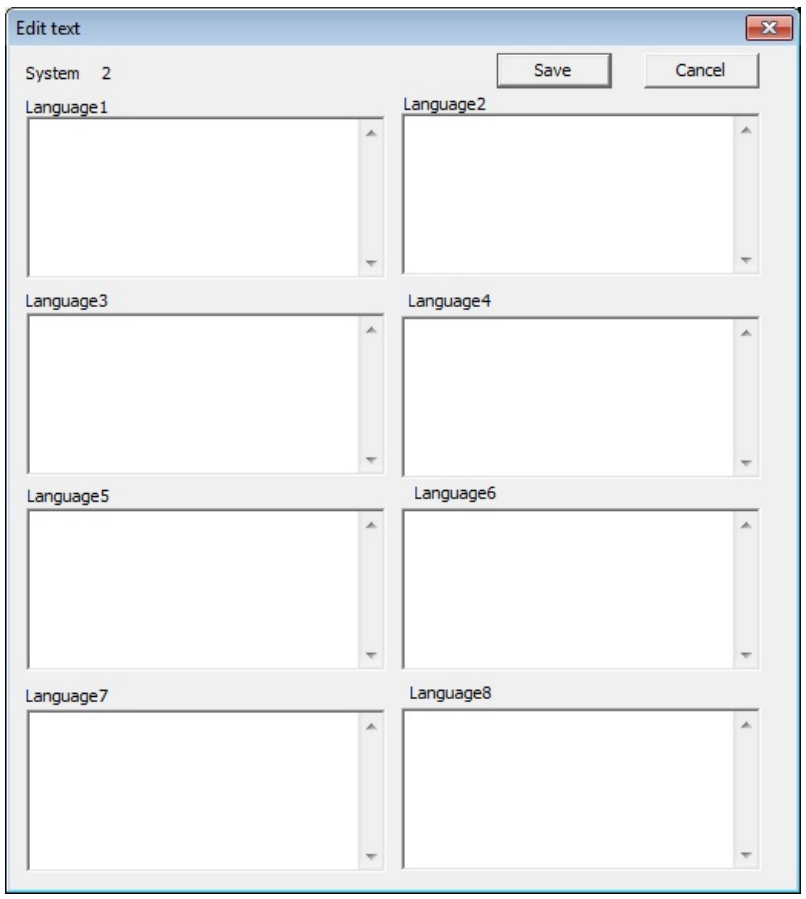


Figure 2

Current Language

Select the display of screen from the tool bar.

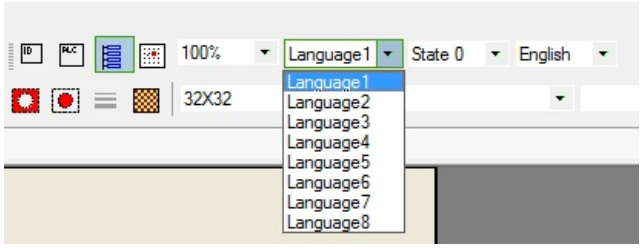


Figure 1

Use Text Library

Objects can be added text from text library.

There are two types of text library:

1. Click [setting]-[text library] to select the proper text you need;
2. Choose [text] label on bit switch, then select [from text Library]-[text library].

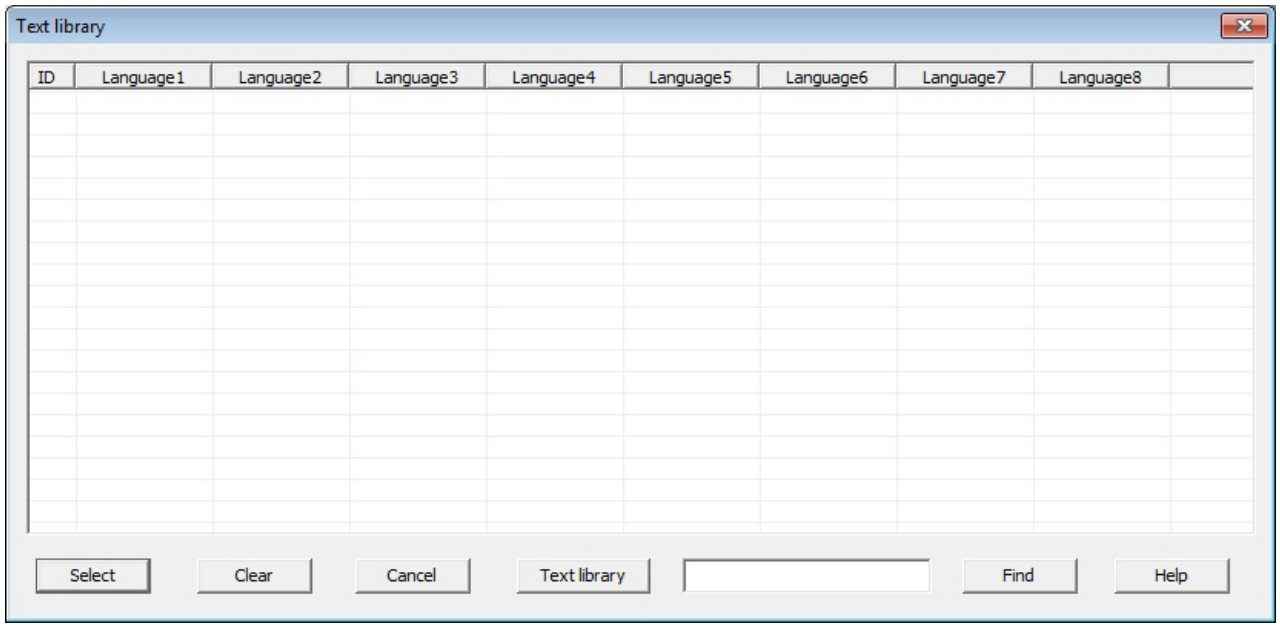


Figure 1

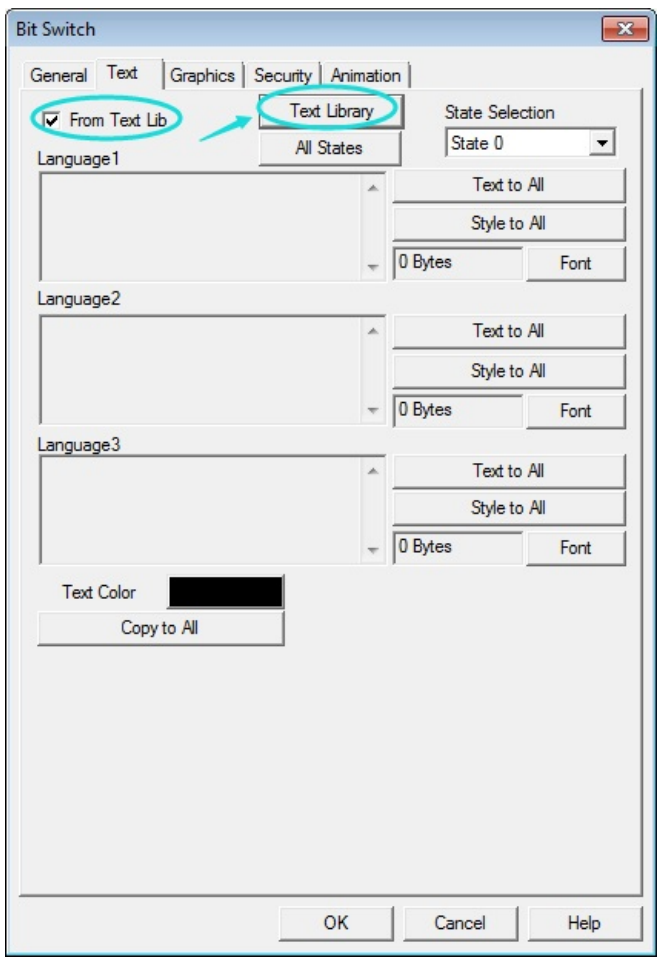


Figure 2

Bit Switch (General/Text/Graphics/Security/Animation)

General

Bit switch is to access the bit-address of the PLC/ HMI. When bit switch is triggered, the changing of [write address] depends on the mode setting. When [read] option is selected, the [read address] is editable.

*For example, if the value of [read address] is “ON”, the state of a designated bit address shows the picture and text corresponds to “ON”, in contrary, the “OFF” features are shown.

Settings

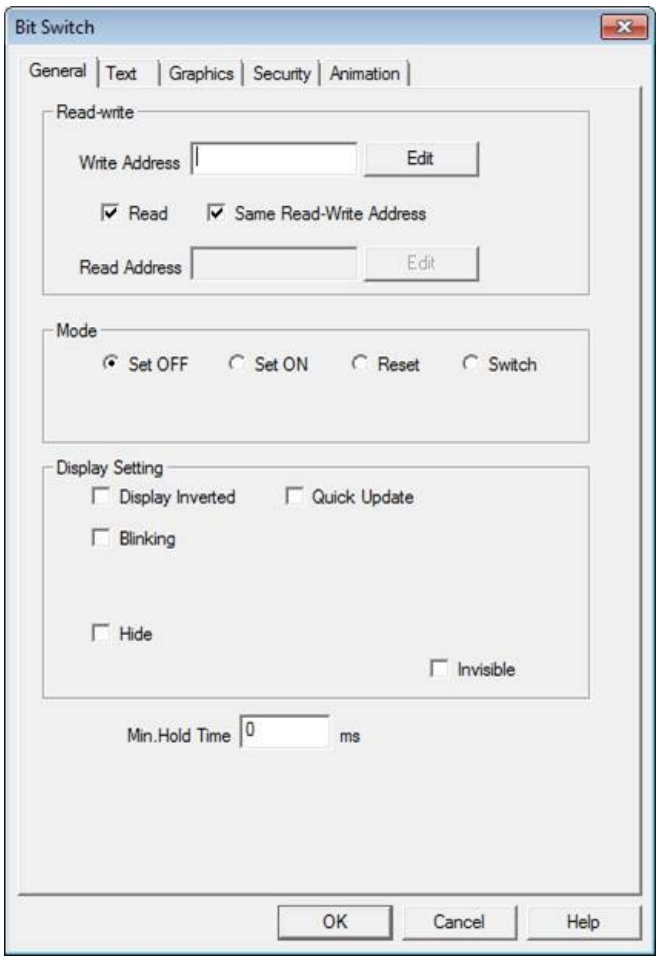


Figure 1

Classification	Property	Description
Read-Write	Write address	Edit the value of Address of PLC/ HMI.
	Read	The Address is red from [Read-Address].
	Same Read-Write Address	The value of [Read Address] is equals to [Writing Address].
	Read-address	Read the value from the set address.
Mode	Set OFF	Set OFF the [Write-Address].
	Set ON	Set ON the [Write-Address].
	Reset	Switch pressed: Set ON the [Write-Address]. Switch released: Set OFF the [Write-Address]. Reset Delay:Set OFF the [Write-Address] after [ ] (ms), when release the switch.
	Switch	Alter the [write-address] between 0 and 1, each time the Switch is triggered.
Display Setting	Display inverted	Display the picture or text inverted. *For Example, the value of [Write-Address] is set ON, the OFF mode is displayed.
	Quick Update	Increase the [Read-address] update frequency.
	Blinking	Blinking while the switch is triggered, the frequency is editable.
	Hide	Display or Hide the bit switch. *Bit switch is not operable when [hide] is selected, to set the switch operable while hide, go to [Setting]-[Project Properties] - [operability of hiding setting], and select [enable] option.
Min. Hold Time	N/A	The switch triggers after [ ](ms) in case of misoperation.

Word Switch (General/Text/Graphics/Security/Animation)

General

Change a value of the word address, when the switch is triggered the value of word address will change depends on the switch type.
*For example: Set two word switch with increase and decrease mode respectively, the value will change depends on which type of switch is operated.

Settings

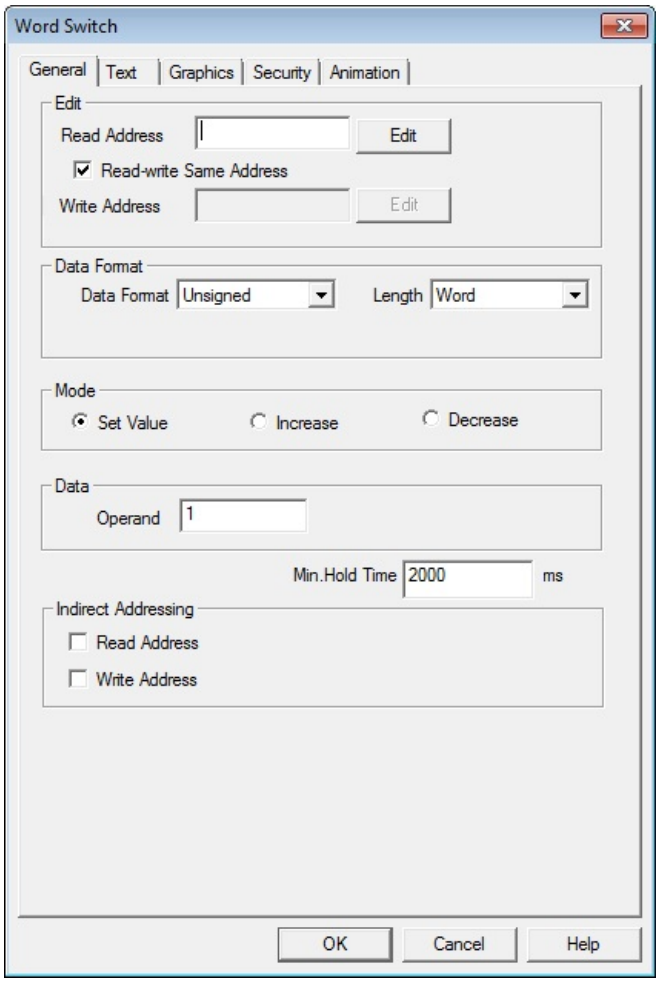


Figure 1

Classification	Properties	Description
Edit	Read Address	Read the value from set address.
	Same Read-Write Address	The value of [Read Address] is equals to [Writing Address].
	Write address	Edit the value of Address for the device.
Data Format	Data Format	The Data Format of the Object.
	Length	The data length of the Object.
Mode	Set value	Set a constant value for the word address.
	Increase	Increase the [operand] value for each time the switch triggered.
	Decrease	Decrease the [operand] value for each time the switch triggered.
	Operand	The variable for each operation.
	High Limit	The value will not increase when the value is reached.
	Low Limit	The value will not decrease when the value is reached.

Keypad Switch (General/Text/Graphic)

General

Keypad switch is a manual data input object; this part must work with the Keypad.
User can assign letters or numbers for each switch, when the switch is triggered the data will input to the textbox.

Settings

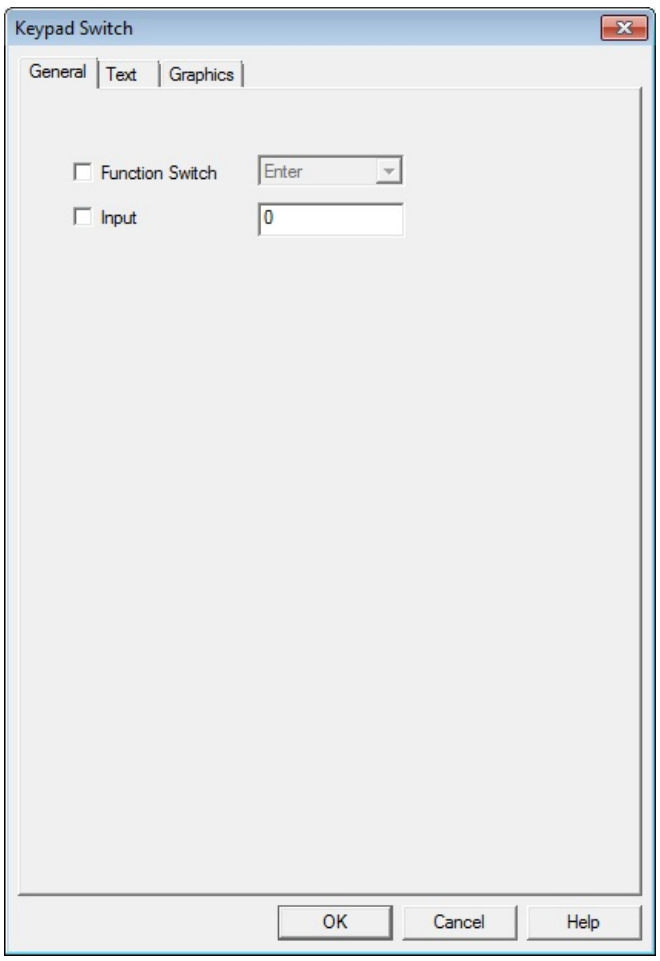


Figure 1

Classification	properties	description
Function Switch	Enter	Corresponds to [enter] on keyboard.
	Delete	Corresponds to [Delete] on keyboard.
	Clear	Corresponds to [Clear] on keyboard.
	ESC	Corresponds to [ESC] on keyboard.
	Symbol	For the symbol input.
	Switch	Change the input mode between Chinese and English.
	Pinyin UP/Down	Page up/down for Pinyin input.
	Chinese UP/Down	Page up/down for Chinese input.
	Pinyin 1~3	Select Pinyin from designated number.
	Chinese 1~6	Select Chinese from designated number.
Input	N/A	Set input value in ASCII code.

Slider Switch (General /Graphics)

General

Change the address value from the slider position.

*For example: set the high and low limit for a slider as 0-100; read-write address as HDW0, when the slider moves to the middle of the switch, the value of HDW0 is 50.

Settings

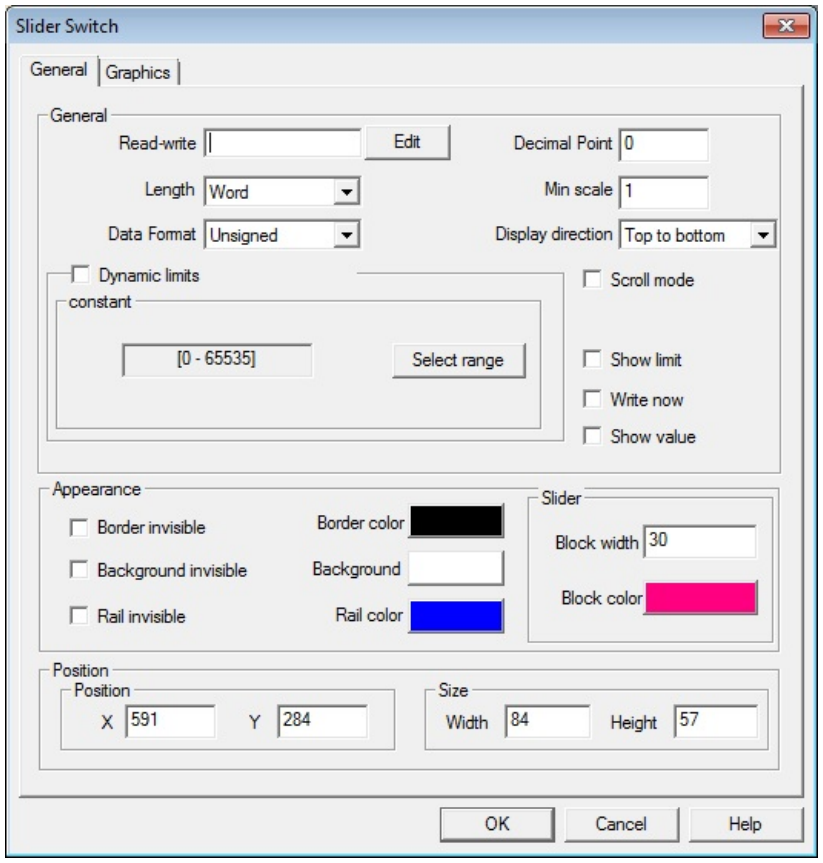


Figure 1

Classification	Properties	Description
General	Read- write	Read and write for the designated address.
	Length	The length of the read-write address.
	Data format	The data format for the read-write address.
	Dynamic limits	The limits of value will read from address.
	Decimal point	N/A
	Min scale	The minimum operand for each operation.
	Display direction	Slider movement direction.
	Scroll mode	Set constant operand for each operation. *For example, scroll value is set as 10, every time slides the switch the data value will increase or decrease a constant value of 10.
	Show limits	Show the limits on the slider
	Write now	Out the value in real time.
	Show value	Display the slide value on the left side of the switch.
Slider	Block width	Set the width of the slider.
	Block color	Set the color for the slider.

Function Key (General/Text/Graphics/Security/Animation)

General

Function key can implement functions like switch between different screens, copy data to UDisk, file transfer, ect.

Settings

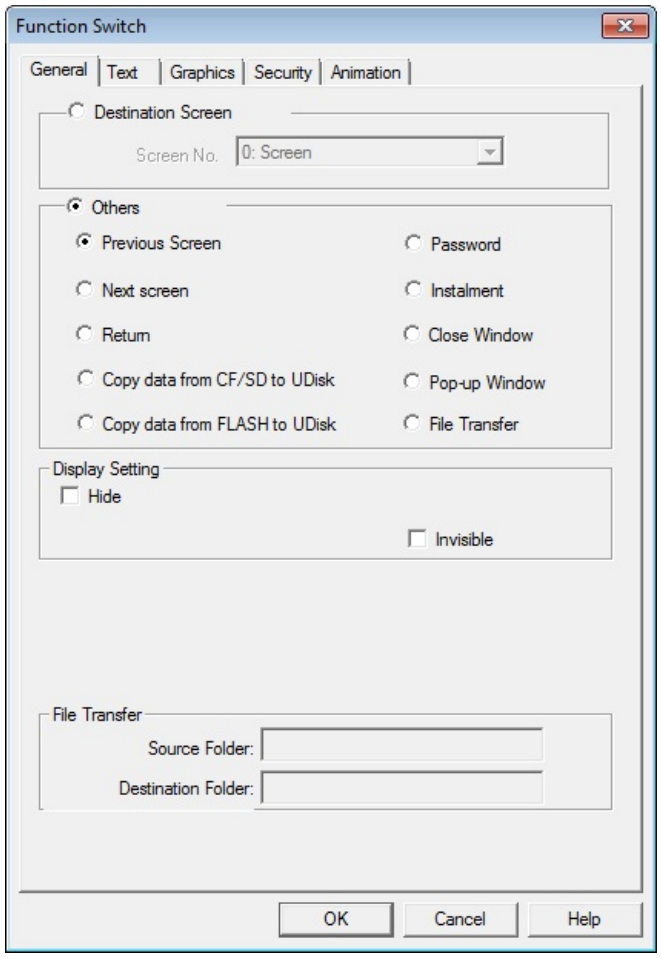
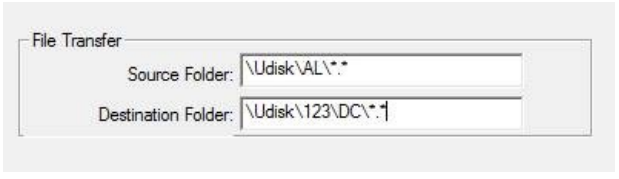
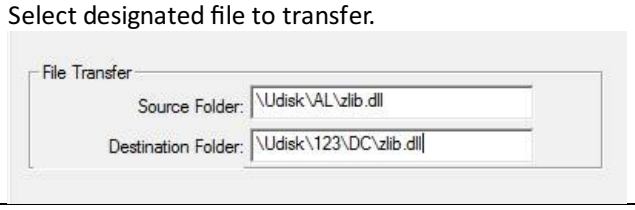


Figure 1

Classification	Properties	Description
Destination screen	Screen No.	Switch to the selected screen.
Others	Previous screen	Switch to previous screen.
	Next screen	Switch to next screen.
	Return	Return to last screen.
	File transfer	Transfer file to designated location.
	Copy data from CF/SD to UDisk	Transfer data from CF/SD to Udisk for data backup and alarm recording.
	Copy data from FLASH to UDisk	Transfer data from FLASH memory to Udisk for data backup and alarm recording.
	Pop-up window	Pop-up designated sub-screen, relevant pop-up window setting can be found underneath.
	Password	Input data from pop-up Keypad, if the physical data input is overtime.
	Instalment	Pop-up the instalment window instantly.
	Close window	Close current sub screen.
Display setting	Hide	Display or Hide the Object. *Function Key is not operable when [hide] is selected, to set the switch operable while hide, go to [Setting]-[Project Properties] – [operability of hiding setting], and select [enable] option.
	Invisible	Keep the function switch hided.
Pop-up window	Screen No.	Pop-up the designated screen.
	Title	Display or hide the title bar.
	Position	The pop-up window will display on the designated position.
	Mode	Pop-up window can close with the main screen.
	Source folder	Select the folder need to be transferred.
	Destination	Select the folder to save the file

File transfer	folder	Select the folder to save the file.
	File transfer description	Udisk :\Udisk\ Flash:\Flash\ SD: \CFDC\ C Disk: C:\ Use *.* at the end of the route to transfer all files under the folder, shown as following  Select designated file to transfer. 

Recipe Transfer (General/Text/Graphics)

General

Recipe transfer is designed for reading and writing the recipe address, over write will replace the current address.

Setting

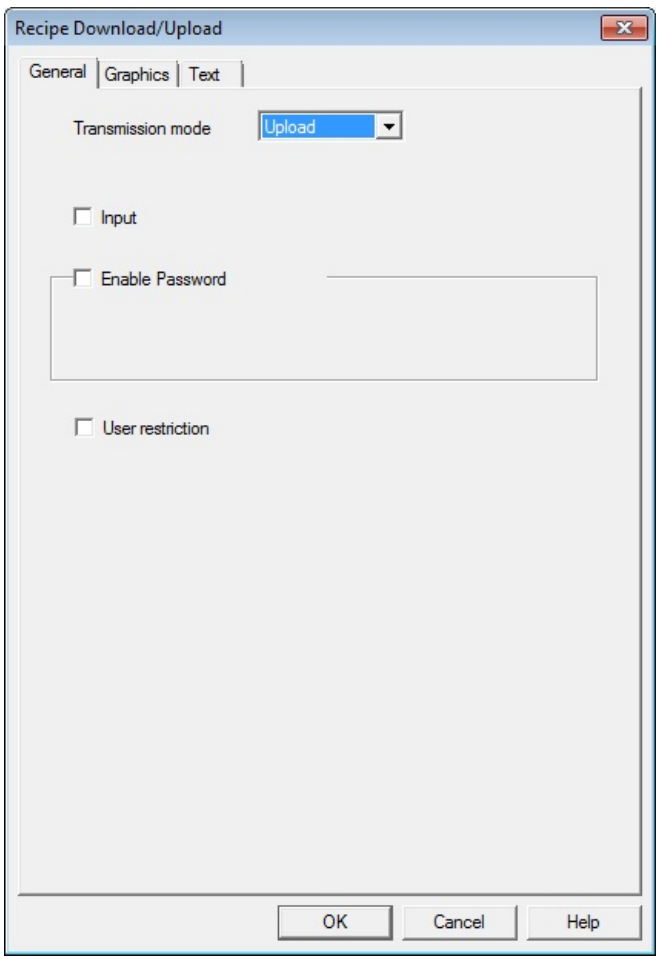


Figure 1

Classification	Properties	Description
Transmission mode	Upload	Transfer the data to the designated recipe.
	Download	Write the recipe to designated address.
Enable password	N/A	Please refer to Parts Security
User restriction	N/A	Please refer to User Restriction

Combination Switch (General/Text/Graphics/Security/Animation)

General

Individual functions can be added to one object, to implement multiple tasks at the same time.
*For example, add “set bit” and “destination screen” to a combination switch, when press the switch, it will trigger a bit switch while switch to another screen.

Settings

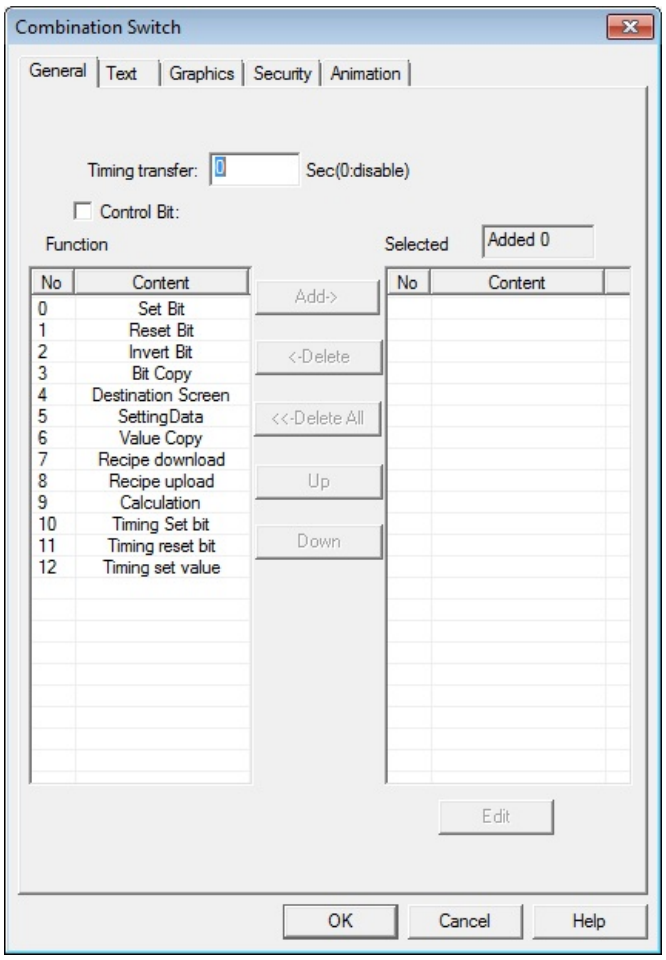


Figure 1

Classification	Properties	Description
Timing transfer	N/A	System will trigger the object periodically depends on the set time.
Control bit	N/A	Trigger the object by designated address.
Function	Set bit	Set ON the designated address.
	Reset bit	Set OFF the designated address.
	Invert bit	Invert the designated address, for each time the Switch is triggered.
	Bit copy	Copy multiple continues addresses, the address number needs to be set.
	Destination screen	Switch to designated screen.
	Setting data	Set multiple data, the data volume needs to be set.
	Value copy	Copy multiple data, the data volume needs to be set.
	Recipe download	N/A
	Recipe upload	N/A
	Calculations	Perform adding, subtraction, multiplication, division, complementation calculations to designated addresses.

Numerical input/ display (General/Graphics/Security/Advanced/Animation)

General

Numerical input/ display in designated data format, the data is scalable.

Data format	Description
Binary	Binary system is a number system which represents numeric values using two different symbols: typically 0 (zero) and 1 (one).
Octal	Octal system is a number system using data from 0 to 7.
Unsigned	Data contain from 0 to 9.
Hex	Data value expressed by symbols 0 -9 and “A,B,C,D,E,F”.
BCD	Binary-coded decimal (BCD) is a class of binary encodings of decimal numbers where each decimal digit is represented by a fixed number of bits, usually four or eight. Special bit patterns are sometimes used for a sign or for other indications.
32bits floating	Double word by default.

Settings

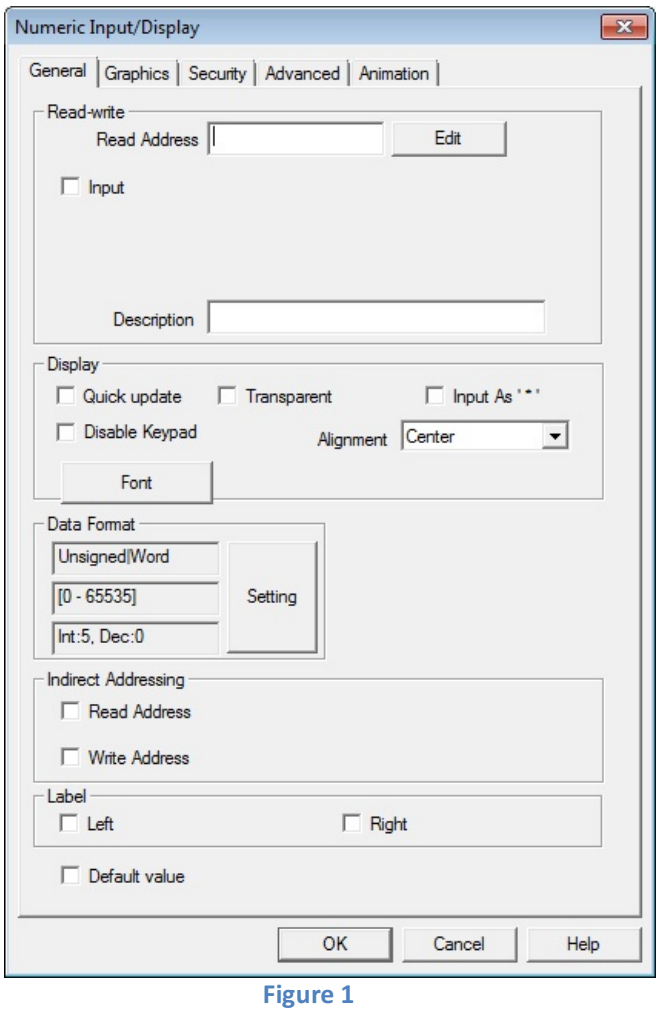


Figure 1

Classification	properties	Description
Read-write	Read address	The data from designated address will be displayed.
	Input	Initialize the keypad input and address input function.
	Same read-write address	The value of [Read Address] is equals to [Writing Address].
	Write address	Shows the state of designated address on the object.
	Keypad	Select keypad features and display condition.
	Description	N/A
Display	Quick update	Increase the data update and display fresh frequency.
	Transparent	Data input background shown transparent.
	Input as “*”	The data input shown as “*”, this function is designed for password security.
	Disable keypad	The keypad input function is not functional.
	Alignment	Input box position.
	Font	N/A
Keypad	N/A	Keypad display position

position	N/A	Keypad display position.
Indirect addressing	Read address	
	Write address	
Label	Left	The text shown on the left side of the input box.
	Right	The text shown on the right side of the input box.
Default value	N/A	Need to custom the keypad, add the key of “Default value ” to keypad scree, ID should be 99.

Character input/display (General/ Graphics / security / Animation)

General

Data from PLC is inputted / displayed in ASCII format, multiple continues data can input/ display simultaneously.

Settings

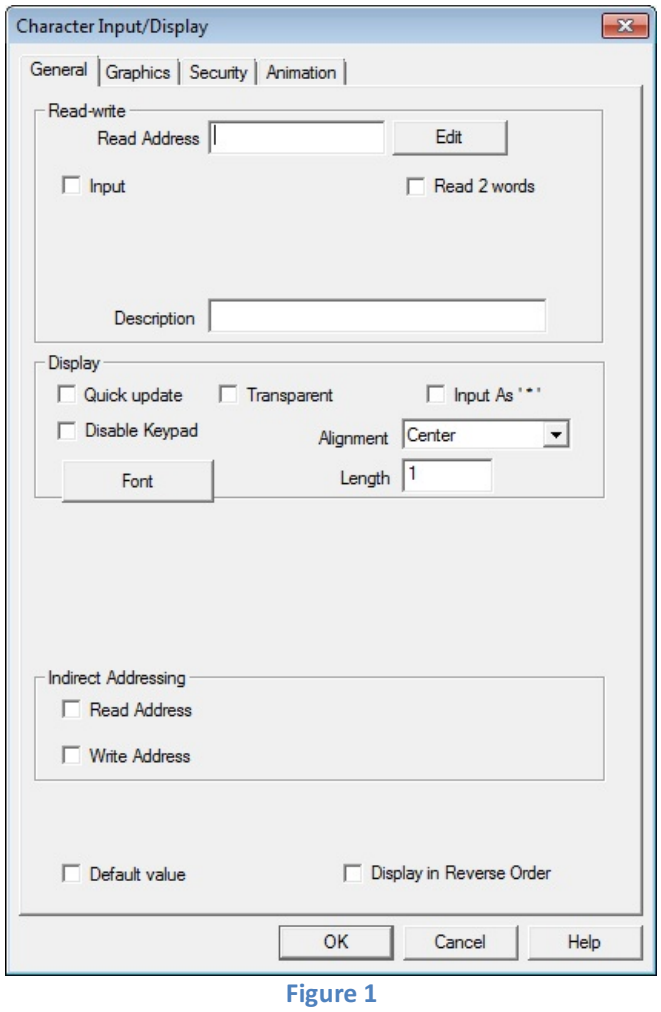


Figure 1

Classification	properties	description
Read-write	Read 2 words	Read 2 bytes data simultaneously.
Display	Length	The address length of the read or write address.
Display in reverse order	N/A	Reverse the order of high byte and low byte.
*The description of other items, please refer to “Numeric Input/Display”.		

Log Display

General

The object is designed to display and browse the operation log.

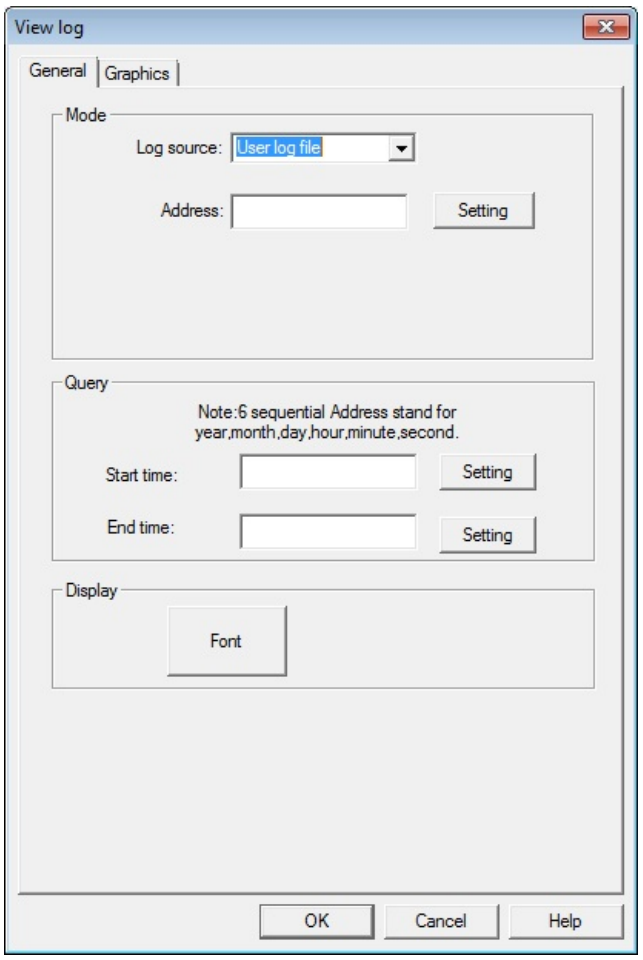


Figure 1

Settings

Classification	Properties	Description
Mode	Log source	The file location to view the log.
	Address	The address is aiming for receive function code to interact with the HMI. The function code shown as following: 1: page up 2: page down 3:front page 4:last page 5:search 6:cancel the search
Query	Start time	Set the start time for the query period. For example, set the start time as HDW10(HDW0~HDW15 will be occupied) HDW0:year,HDW11:month,HDW12:day,HDW13:hours:HDW14:minute,HDW15:second.
	End time	Set the end time for the query period.

Alarm record display (Graphics)

General

Check recorded alarm, the time span set by the user, and the alarm record will list by the time sequence.

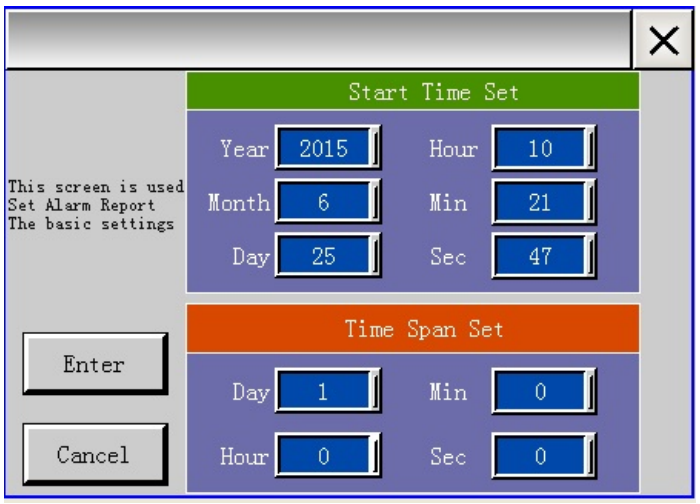


Figure 1

Settings

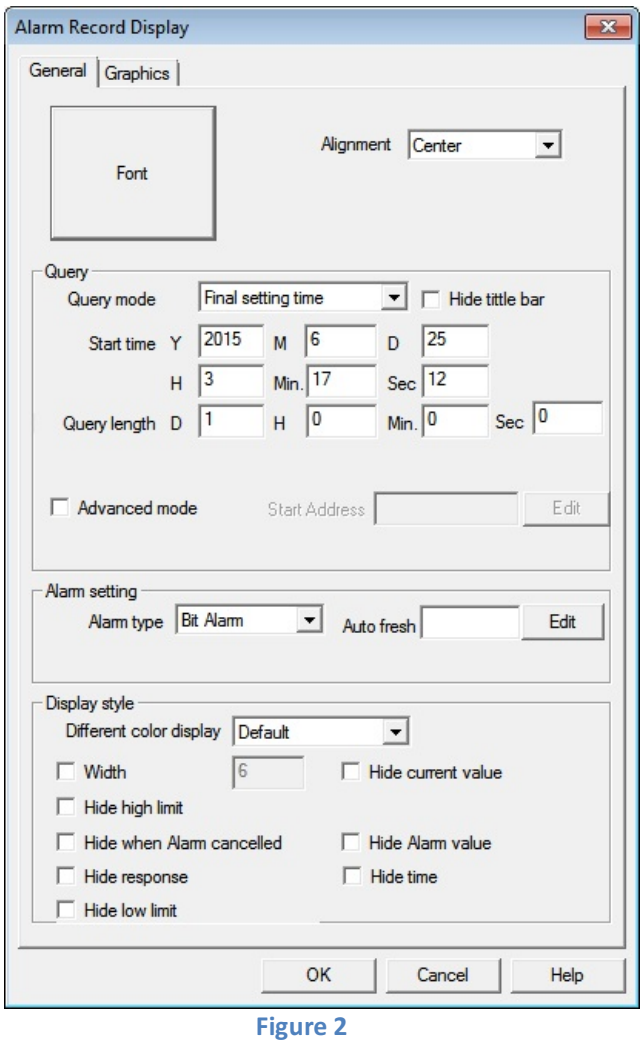


Figure 2

Classification	Properties	Description
Query	Query mode	Set query time.
	Hide title bar	N/A
	Start time	Set the start time to collect alarm data. Date format set as: Year/month/day/hour/minute/second.
	Query length	Set the time interval for query, the maximum length is 31 days. Date format: day/hour/minute/second.
Advanced mode	N/A	Refer the description down below.
Alarm setting	Alarm type	Select between bit alarm and word alarm.
	Auto fresh	Set the address for auto fresh key.

Advanced mode

Set start address, the next 15 addresses will all be occupied.

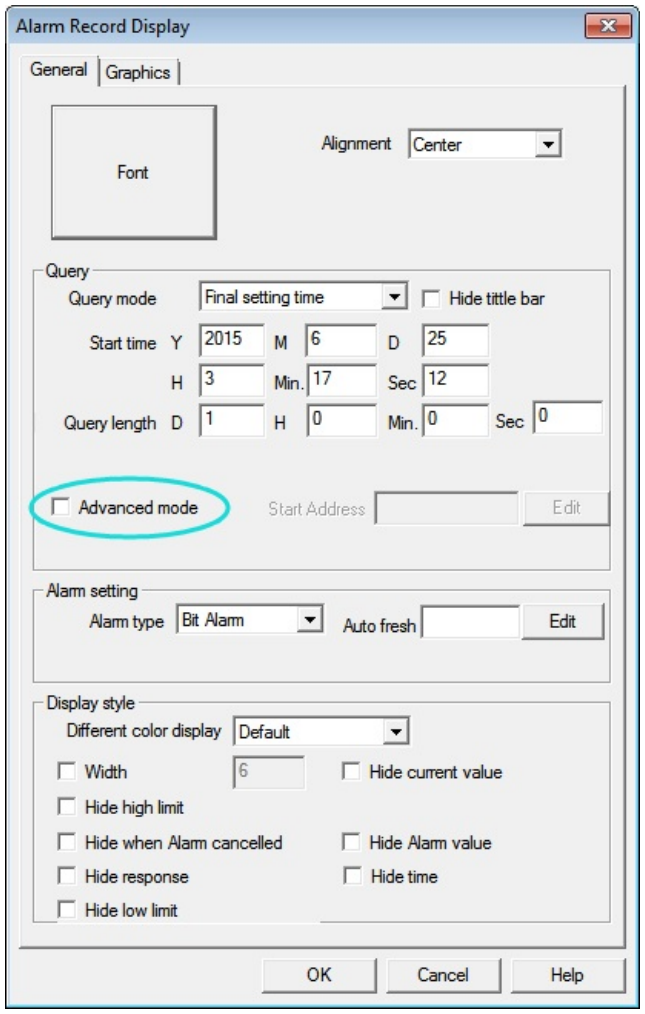


Figure 3

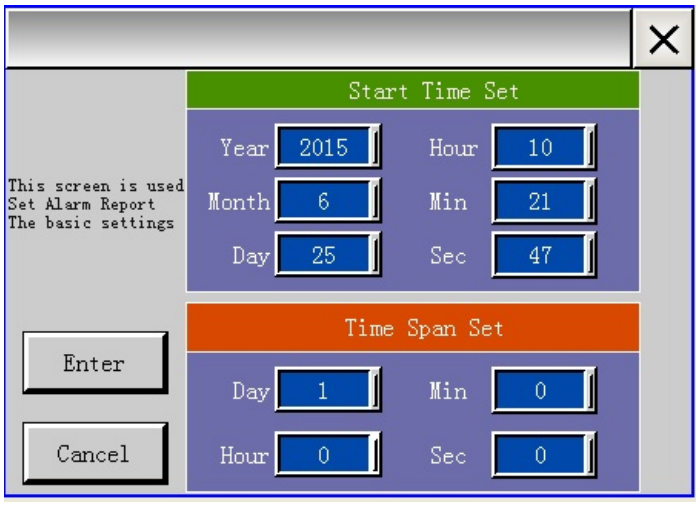


Figure 4

If the start address set as HDW100 the next 15 address perform the following functions.

Address	Address type	Functions	remark
HDW100	Word address	Start time: year	
HDW101		Start time: month	
HDW102		Start time: day	
HDW103		Start time: hour	
HDW104		Start time: minute	
HDW105		Start time: second	
HDW106		Time interval: day	
HDW107		Time interval: hour	
HDW108		Time interval: minute	
HDW109		Time interval: second	
HDX110.0	Bit address	Refresh	Auto reset
HDX110.1		ON:keep refreshing OFF:stop refreshing	
HDX111.0		Page up	Auto reset
HDX111.1		Page down	
HDX111.2		Page up in the designated range	
HDX111.3		Page down in the designated range	

History alarm (General/Graphics)

General

All alarm records are displayed, data includes alarm time, clear time. History alarm record will erased after HMI powered off.

Settings

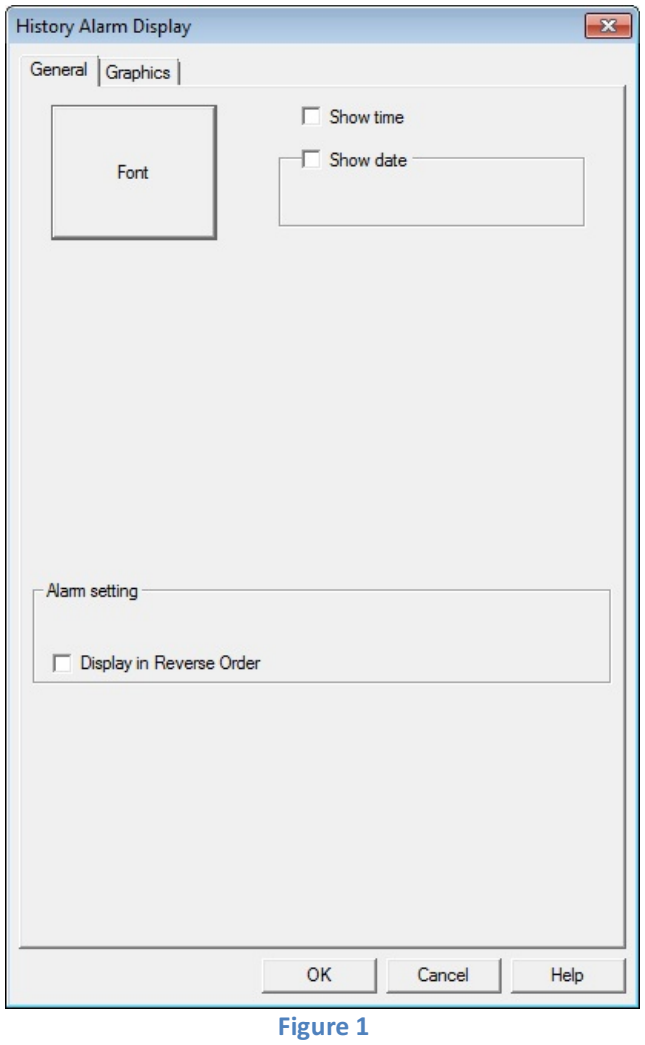


Figure 1

Classification	Properties	Description
Show time	N/A	Display alarm time and clear time.
Show date	N/A	Display alarm time and clear time by selected order.
Alarm setting	Display in reverse order	Reverse display the alarm order.

Alarm Bar

General

Scroll the alarm information on HMI.

Settings

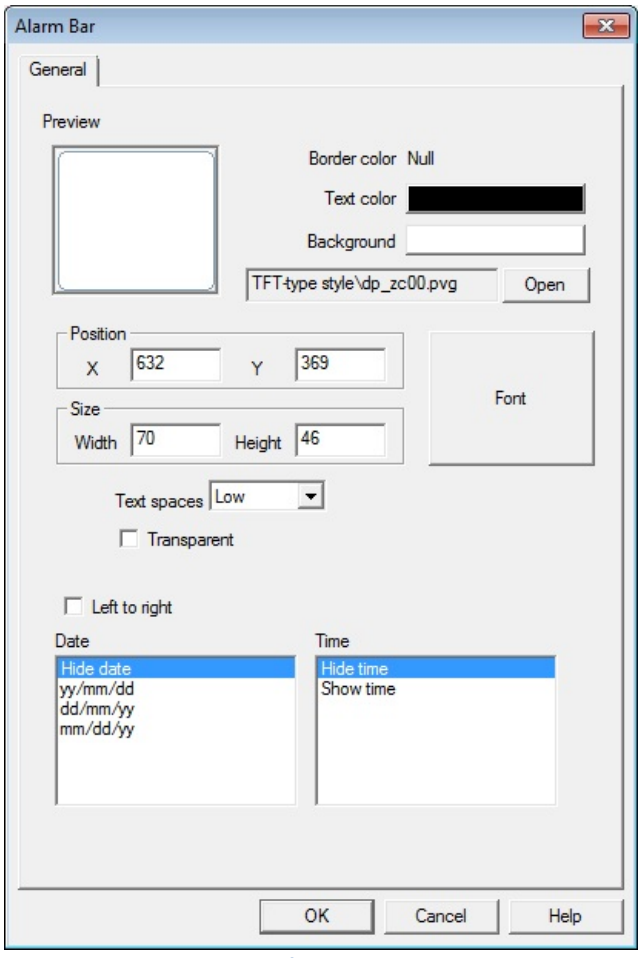


Figure 1

Classification	properties	Description
Text speed	N/A	Set the scroll speed.
Transparent	N/A	Set the display background.
Date	N/A	Date format.
Time	N/A	Time format.

Alarm display (General/Graphics)

General

This function will list the current alarm information.

Settings

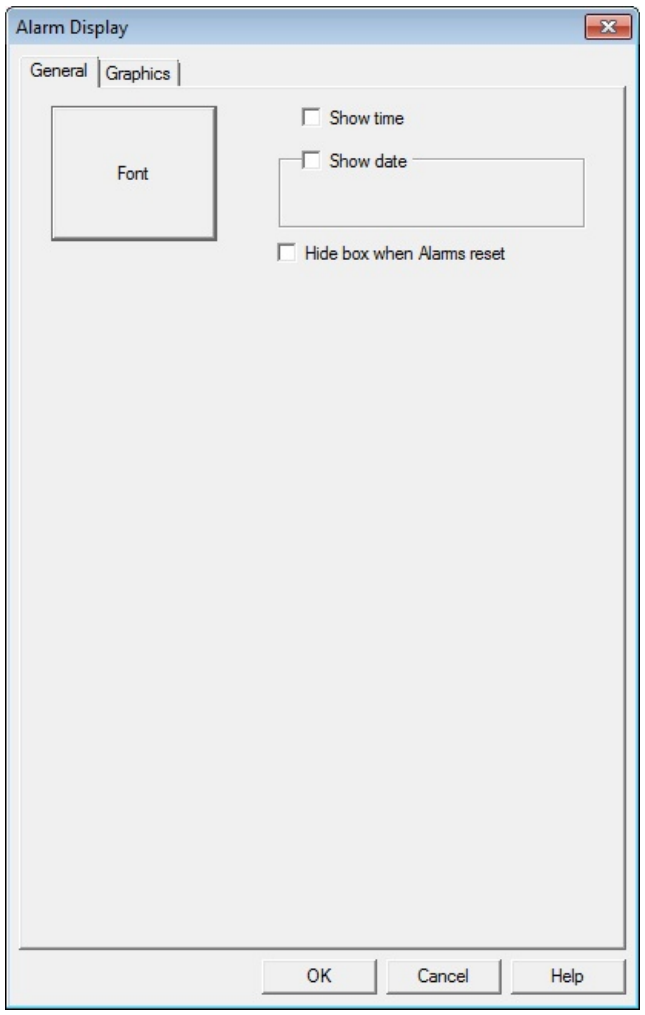


Figure 1

classification	Properties	Description
Show time	N/A	Display alarm time and clear time.
Show date	N/A	Display alarm time and clear time by selected order.
Hide box when alarms reset	N/A	Hide alarm information after the alarm reset.

History Disk Record Display

General

The data is recorded in cache and displayed in disk diagram.

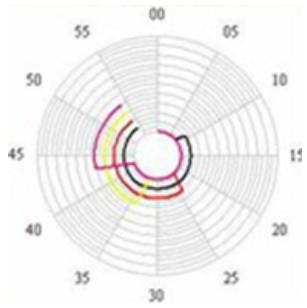


Figure 1

Settings

The history disk record is able to record only one data group, maximum 4 data type, each data type corresponds to one disk curve.

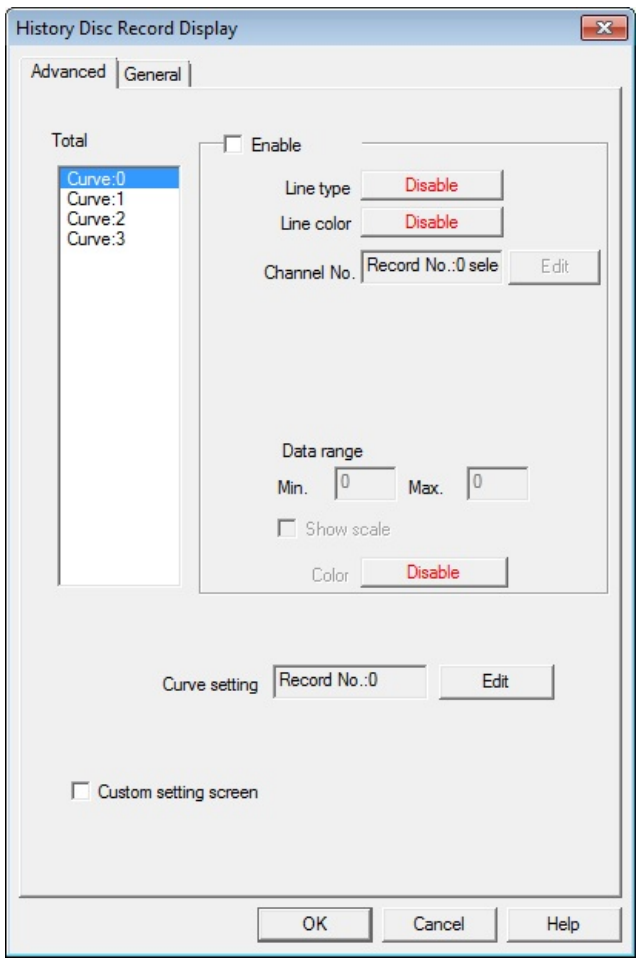


Figure 2

Classification	Properties	Description
Total	N/A	The line type and color is editable for each curve number selected.
Enable	Channel No.	Select channel number needs to be displayed.
	Data range	The record data range.
Curve setting	N/A	Select the curve number needs to be edited.

Trend Display (percent) (General/Graphics)

General

Trend graph will visualize the data as the percentage trend.

Settings

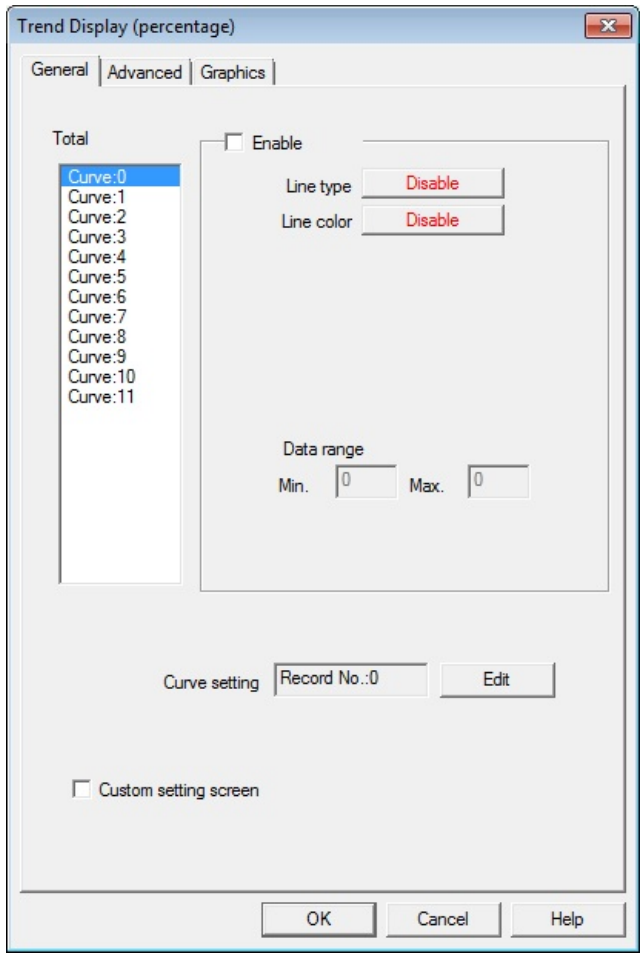


Figure 1

Classification	Properties	Description
Total	N/A	The line type and color is editable for each curve number selected.
Enable	Channel No.	Select channel number needs to be displayed.
	Data range	The record data range.
Curve setting	N/A	Select the curve number needs to be edited.

Data Record Display

General

The data recorded can be displayed, queried as commanded and shown as a table.

Setting

Select [menu bar]N/A [setting]N/A[data record] to add the data group.

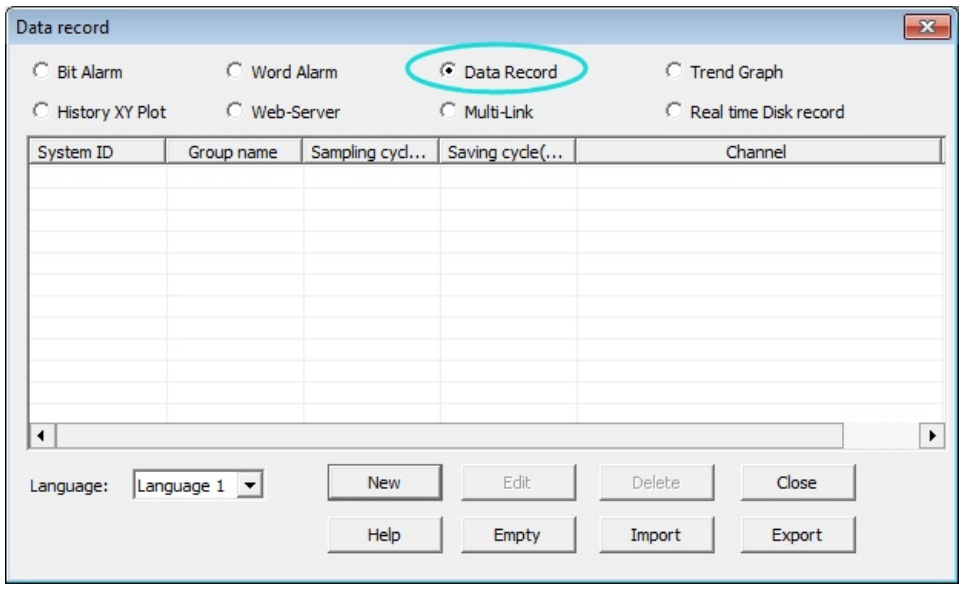


Figure 1

The record and save interval, trigger condition can be edited.

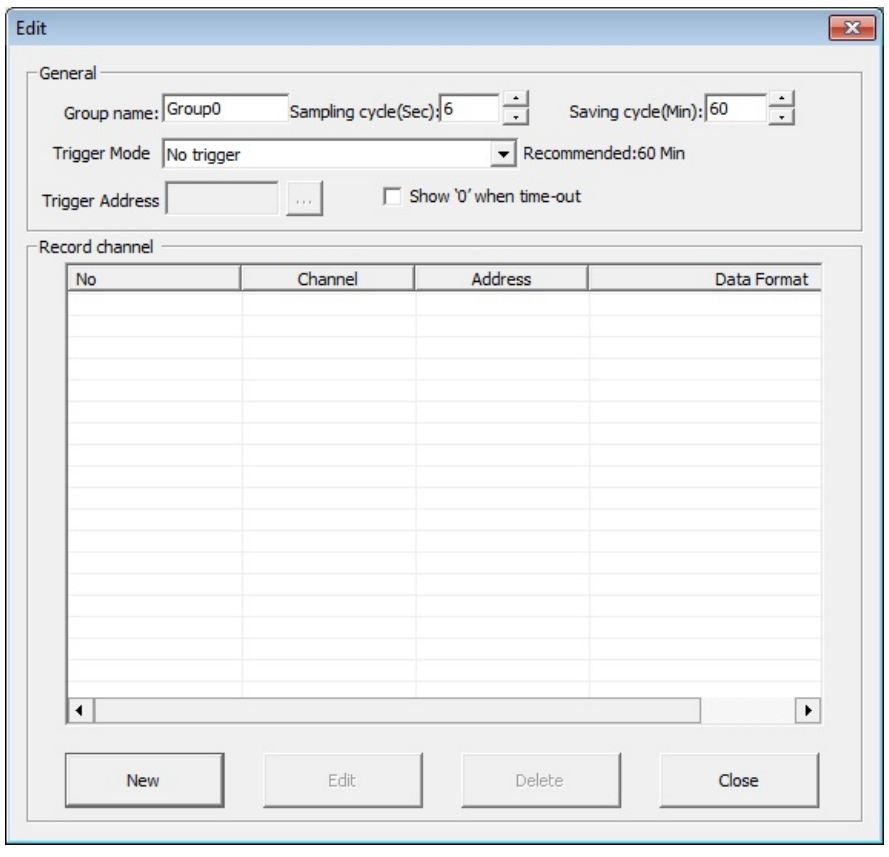


Figure 2

Advanced mode is able to change the query conditions from designated addresses, the "start address" need to be set to activate the function, the following 21 adjacent addresses will be occupied, but only 16 addresses are functional.

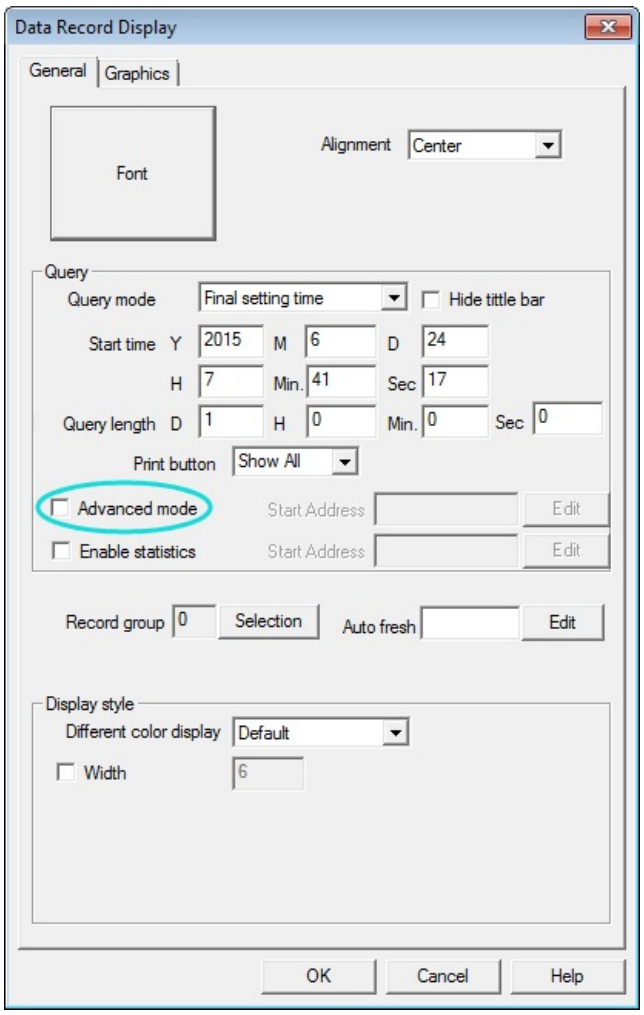


Figure 3

Address	Address type	Part type	function
HDW0	Word address	Numerical display	Start time: year
HDW1			Start time: month
HDW2			Start time: day
HDW3			Start time: hour
HDW4			Start time: minute
HDW5			Start time: second
HDW6			Time interval: day
HDW7			Time interval: hour
HDW8			Time interval: minute
HDW9			Time interval: second
HDX10.0	Bit address	Bit switch	Require auto print
HDX10.1			Confirm auto print
HDX10.2			Query for printing
HDW11	Word address	Numerical display	Time interval for auto print
HDW12			Start row for printing
HDW13			End row for printing
HDW14			Query for print interval
HDX15.0	Bit address	Bit switch	Page up for a set number
HDX15.1			Page down for a set number
HDX15.2			Page up
HDX15.3			Page down
HDW16	Word address	N/A	N/A
HDW17		N/A	N/A
HDW18		N/A	N/A
HDW19		N/A	N/A
HDW20		N/A	N/A

When the project is operating, hold the part for 2~3 seconds to activate the query time.

Classification	Properties	Description
Start time	N/A	Time format: year/month/day/hour/minute/second
Interval	N/A	Time format: day/ hour/ minute/second. The maximum time interval is 31 days

HSW257: Record time interval.
HSW948: Print time interval.
Print time must longer than the record time interval.
Print time interval better set as multiple of record time interval.
The print time interval will be set as record time interval, while the previous setting invalid.

–addresses shown as following:
HSW257 - HSW261 N/A HSW265 N/A HSW269 ....
HSW9481 - HSW9482 N/A HSW9483 N/A HSW9484....

Disk Record Display

General

The real- time data is recorded in cache and displayed in disk diagram.

Setting

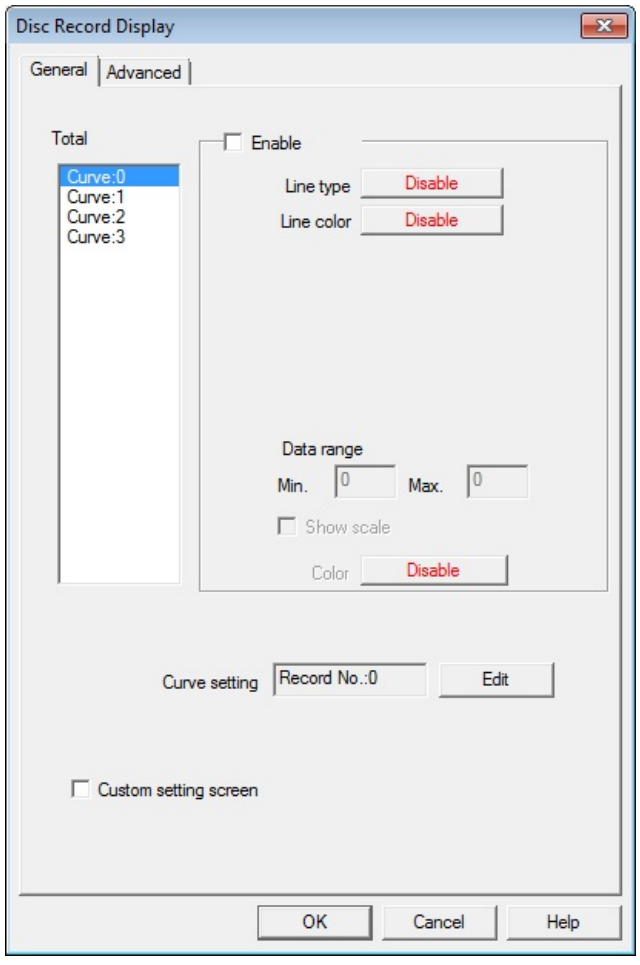


Figure 1

Classification	Properties	Description
Total	N/A	The line type and color is editable for each curve number selected.
Enable	Channel No.	Select channel number needs to be displayed.
	Data range	The record data range.
Curve setting	N/A	Select the curve number needs to be edited.

History curve display (General/Graphics)

General

The real-time data in HMI cache will display curve graph, which X axis represent as time, Y axis represent as data.

Settings

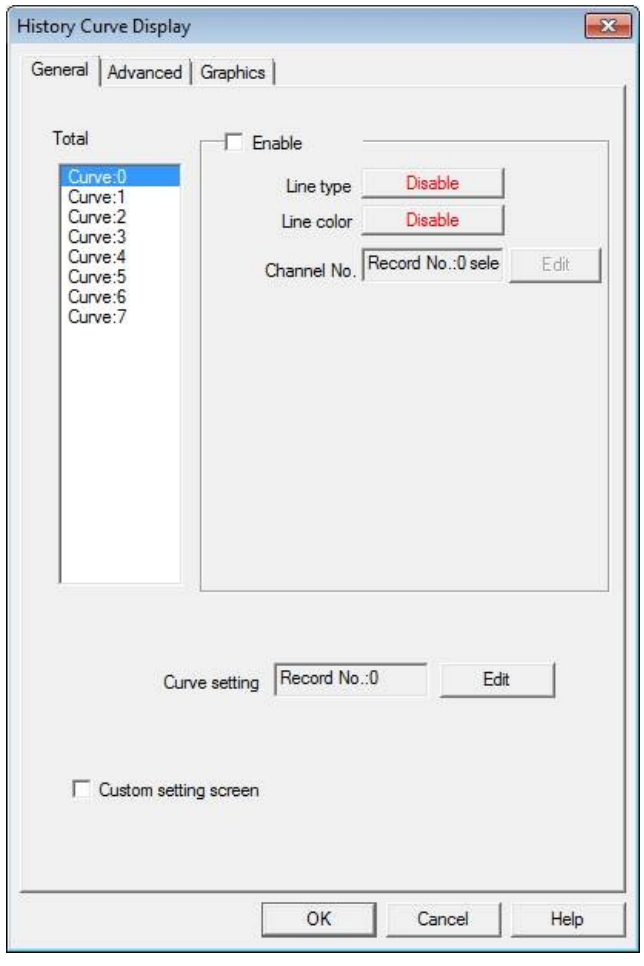


Figure 1

Classification	Properties	Description
Total	N/A	The line type and color is editable for each curve number selected.
Enable	Channel No.	Select channel number needs to be displayed.

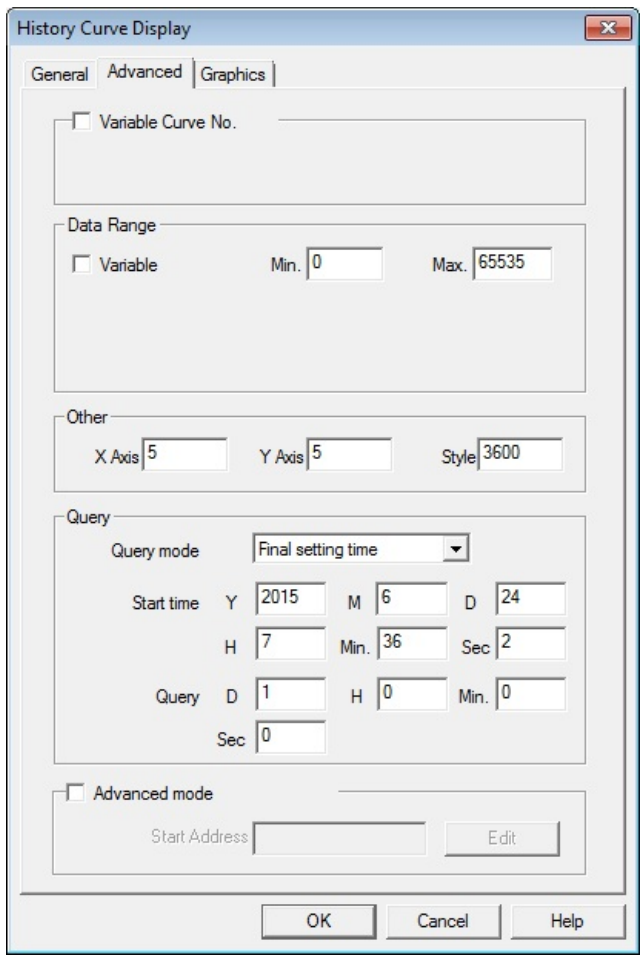


Figure 2

Classification	Properties	Description
Variable curve No.	N/A	The curve will change dynamically as the data change.
Other	X axis	The number of box on X axis.
	Y axis	The number of box on X axis.
	style	Choose the direction to move the object.
Data range	N/A	The variable range.

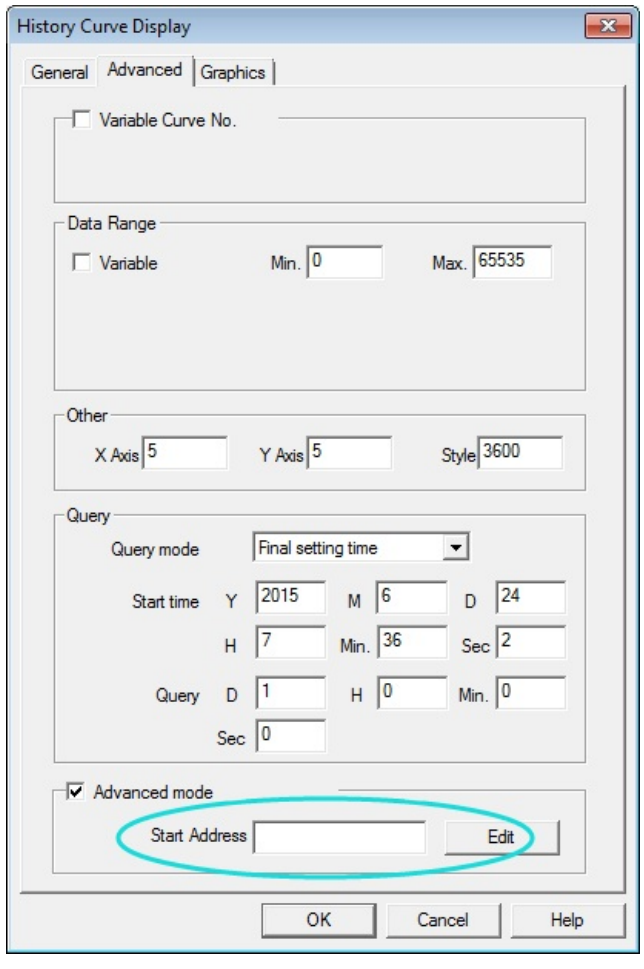


Figure 3

Advanced mode is able to change the query conditions from designated addresses, the "start address" need to be set to activate the function, the following 21 adjacent addresses will be occupied, but only 16 addresses are functional.

Figure 4

Address	Type	Part	function
HDW100	Word address	Numerical display	Start time: year
HDW101			Start time: month
HDW102			Start time: day
HDW103			Start time: hour
HDW104			Start time: minute
HDW105			Start time: second
HDW106			Time interval: day
HDW107			Time interval: hour
HDW108			Time interval: minute
HDW109			Time interval: second
HDX110.0			
HDX110.1			

HDX110.2	Bit address	Bit switch	Whether to display 0-7 the curves
HDX110.3			
HDX110.4			
HDX110.5			
HDX110.6			
HDX110.7			
HDW111	Word address	Numerical display	Maximum data range
HDW112			Minimum data range
HDW113			Movement interval
HDW114			Print percentage
HDW115.0	Bit address	Bit address	Refresh
HDW115.1			ON:keep refreshing OFF:stop refreshing

History XY Plot (General/Graphics)

General

The data in cache will display as continuous curve or separated points, each point in XY plot needs 2 addresses to set a point.

Setting

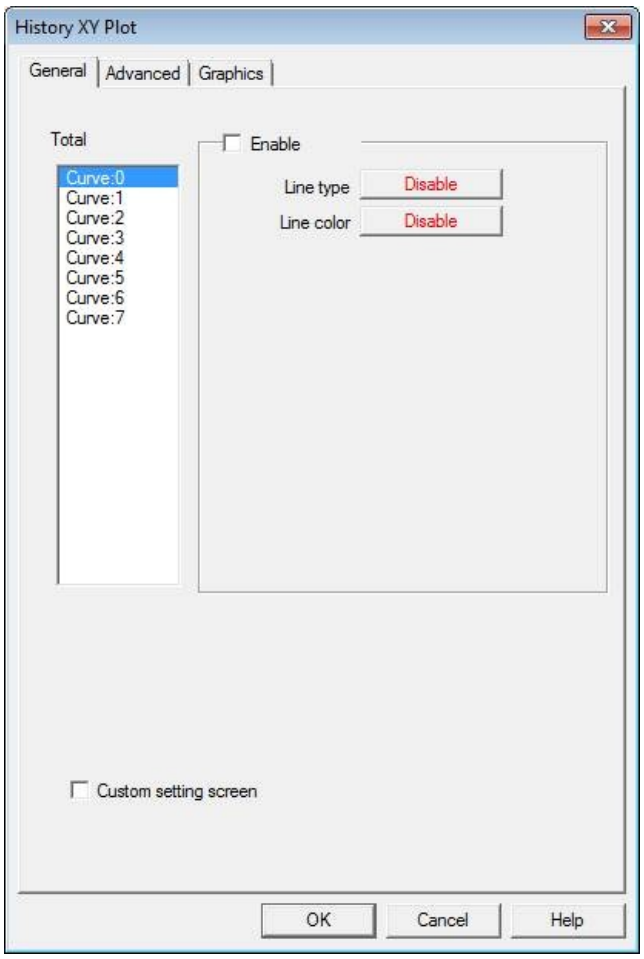


Figure 1

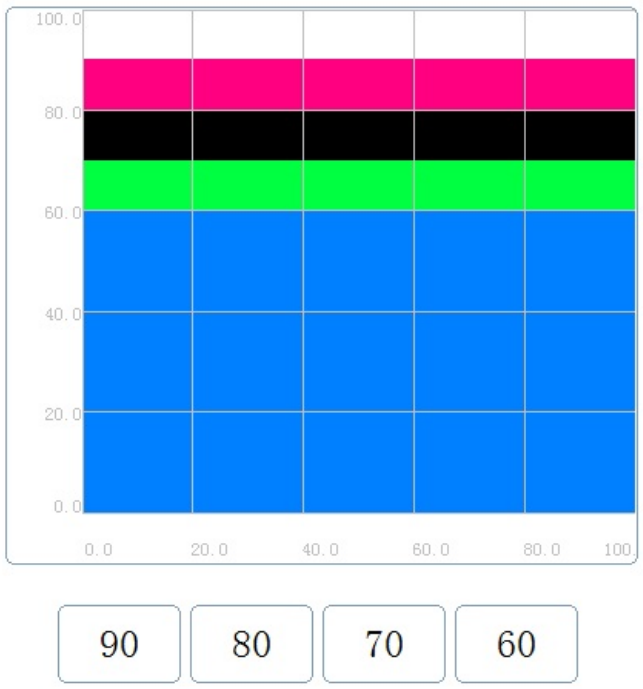


Figure 2

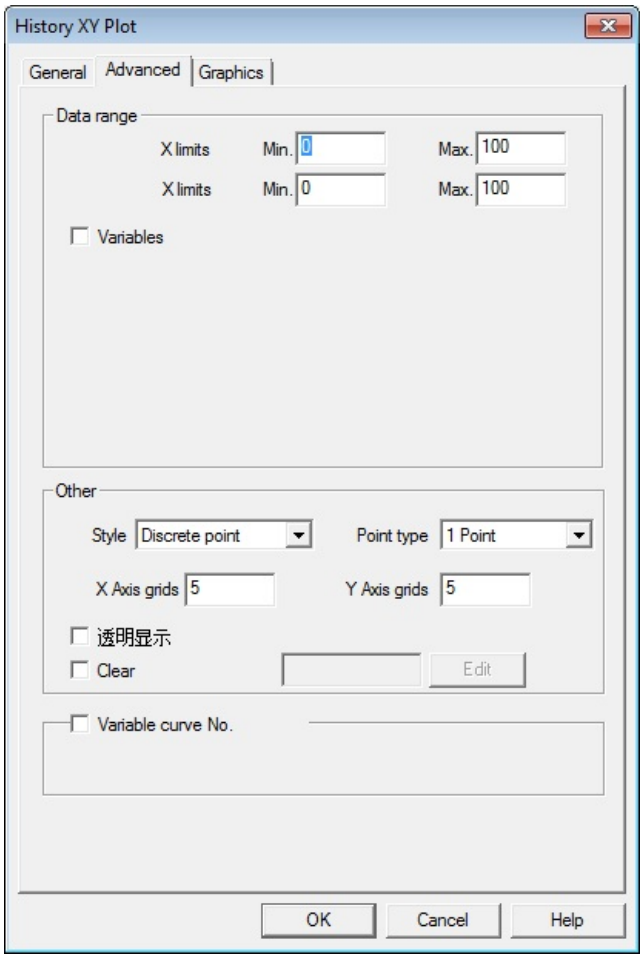


Figure 3

Classification	Properties	Description
Data range	X limits	The Maximum and minimum value displayed on X axis.
	Y limits	The Maximum and minimum value displayed on Y axis.
	X high limit	The X high limit value will determined by the designated address.
	X low limit	The X low limit value will determined by the designated address.
	Y high limit	The Y high limit value will determined by the designated address.
	Y low limit	The Y low limit value will determined by the designated address.
other	Style	Discrete point:the XY graph will displayed as separated dot. Junction:the XY graph will displayed as a curve.
	Point type	1 point:a point is displayed as 1 pixel. 2 point:a point is displayed as 2 pixel. 3 point:a point is displayed as 3 pixel. 5 point:a point is displayed as 5 pixel.
	X axis grids	The grids number on X axis.
	Y axis grids	The grids number on Y axis.
	Clear	Clear the display.

XY Plot (General/Graphics)

General

The real time data in cache will display as continuous curve or separated points, each point in XY plot needs 2 addresses to set a point.

Setting

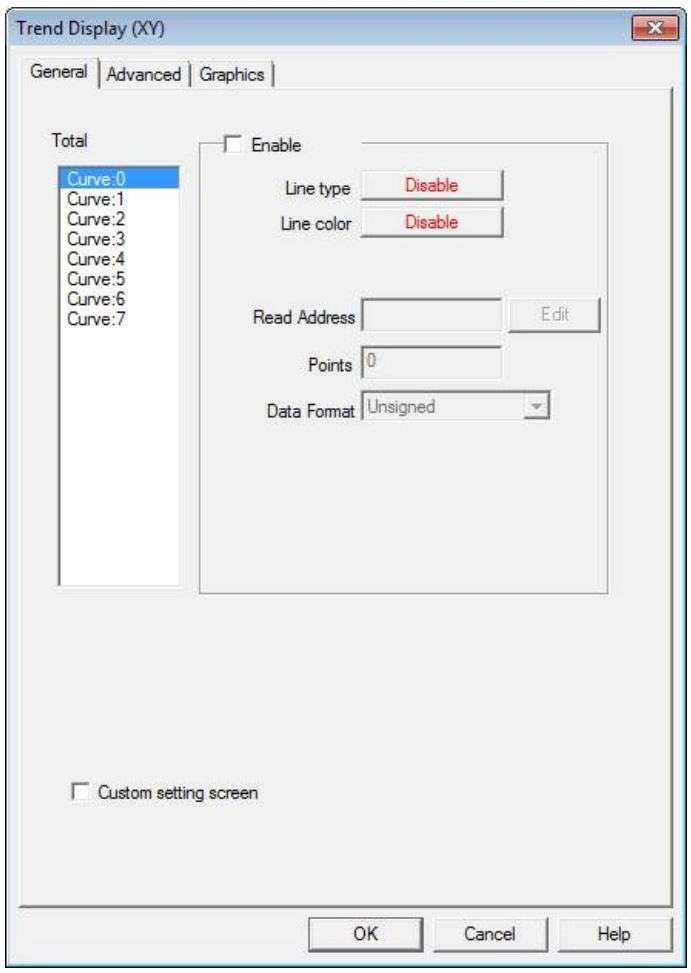


Figure 1

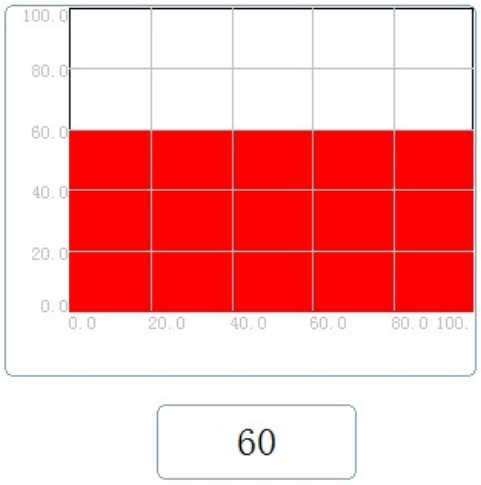


Figure 2

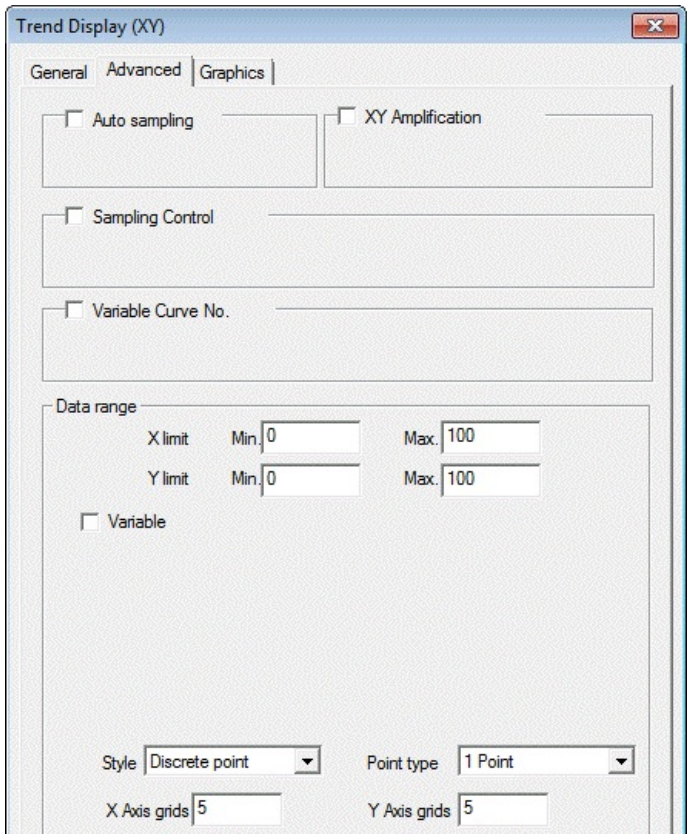


Figure 3

Classification	Properties	Description
Auto sampling	Sampling time	Auto record time interval.
XY amplification	Amplification	Enlarge the scale of the XY axis.
Sampling control	Control bit	The switch to control the sampling.
Data range	Style	Discrete point:the XY graph will displayed as separated dot. Junction:the XY graph will displayed as a curve.
	Point type	1 point:a point is displayed as 1 pixel. 2 point:a point is displayed as 2 pixel. 3 point: a point is displayed as 3 pixel. 5 point:a point is displayed as 5 pixel.
	X axis grids	The grids number on X axis.
	Y axis grids	The grids number on Y axis.

Trend Graph

General

Trend graph shows the data as a dynamic curve, the X axis represent the time, Y axis represent the data.

Figure 1

Press “S” icon or hold the graph for 2~3 seconds. The following setting will show up.

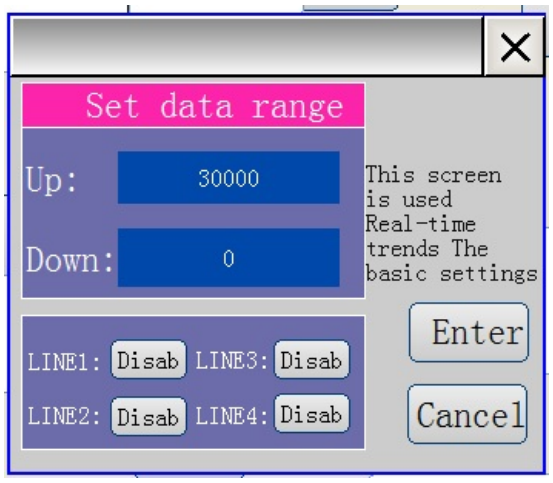


Figure 2

Setting

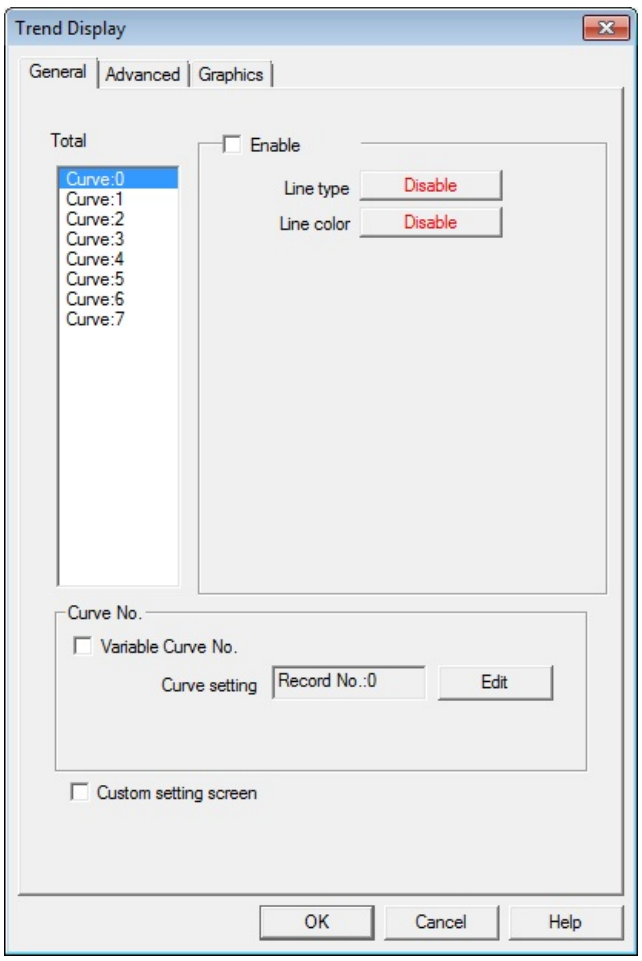


Figure 3

Classification	Properties	Description
Total	N/A	The line type and color is editable for each curve number selected.
Enable	Line type	N/A
	Line color	N/A
Curve No.	Variable curve No.	The curve number will be determined by designated address.
	Curve setting	Select the curve number needs to be edited.
Custom setting screen	N/A	User is able to change the setting UI, but the address changing is not recomanded.

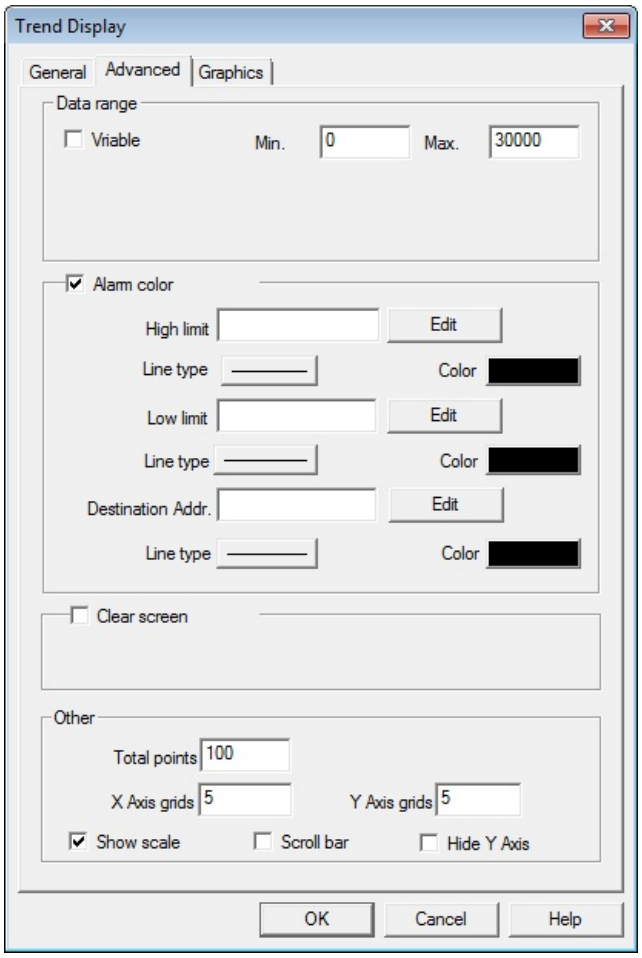


Figure 4

Classification	Properties	Description
Data range	N/A	The data range for the graph can be set as a variable.
Alarm color	N/A	The color for high and low limit, and line type can be set.
Advanced mode	N/A	Refer the table down blow.
Clear screen	N/A	Clear the display by pressing the bit switch.
Others	Total points	The maximum point shown on one graph.
	X axis	The number of box on X axis.
	Y axis	The number of box on X axis.
	Show scale	N/A
	Scroll bar	N/A
	Hide Y axis	N/A

Rotating Picture

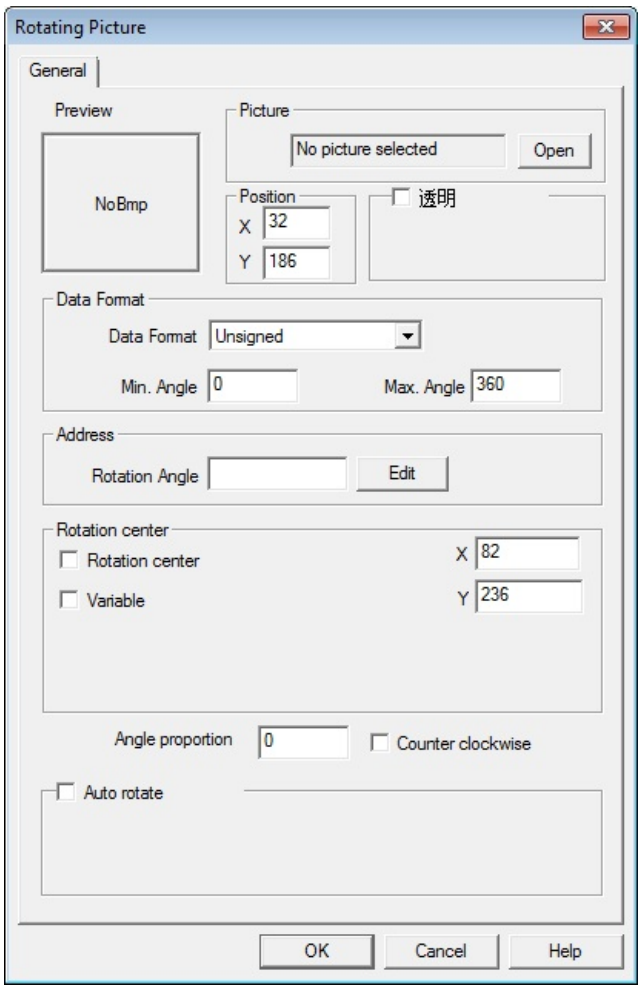
General

Display the data change by rotate the corresponding picture (only support bmp format), user can set rotation centre, direction, angle ect.

Figure 1

Settings

The picture selected cannot be scaled in this function.



?2

Classification	Properties	Description
Angle proportion	N/A	Scale the rotation angle by the set data.
Counter clockwise	N/A	N/A
Transparent	N/A	Delete the selected color in picture.
Data format	Data format	Unsigned, BCD, signed, 32bits floating.
	Min. Angle	N/A
	Max. Angle	N/A
Address	Rotation angle	Set the write address to keep the rotation angle data.
Rotation center	Rotation center	The picture rotates around the central point of the picture.
	X/Y	Set the central rotation point manually, this position may set as a variable.
Auto rotate	Cycle	The picture will rotate every [ ] (100ms). *for example, the data input is 10, the time interval will be 10*100ms=1000ms.
	Automatic angle	Rotates designated angle for each time interval.

Dynamic Picture (General/Text/Graphics)

General

The picture moves as the preset track path, the position and state of the picture is determined by the designated address.

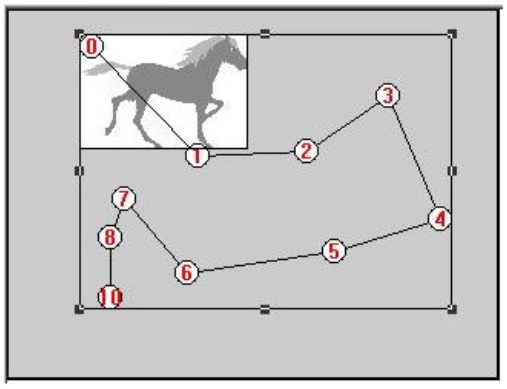


Figure 1

Settings

How to set the object:

Select the dynamic picture function in parts library.

Left-click on the start point (point 0), then drop the following track points (point 1, 2, 3…..) by left-click. The track path contains maximum 50 points.

Drag the point to change the position.

Right –click to finish the track input.

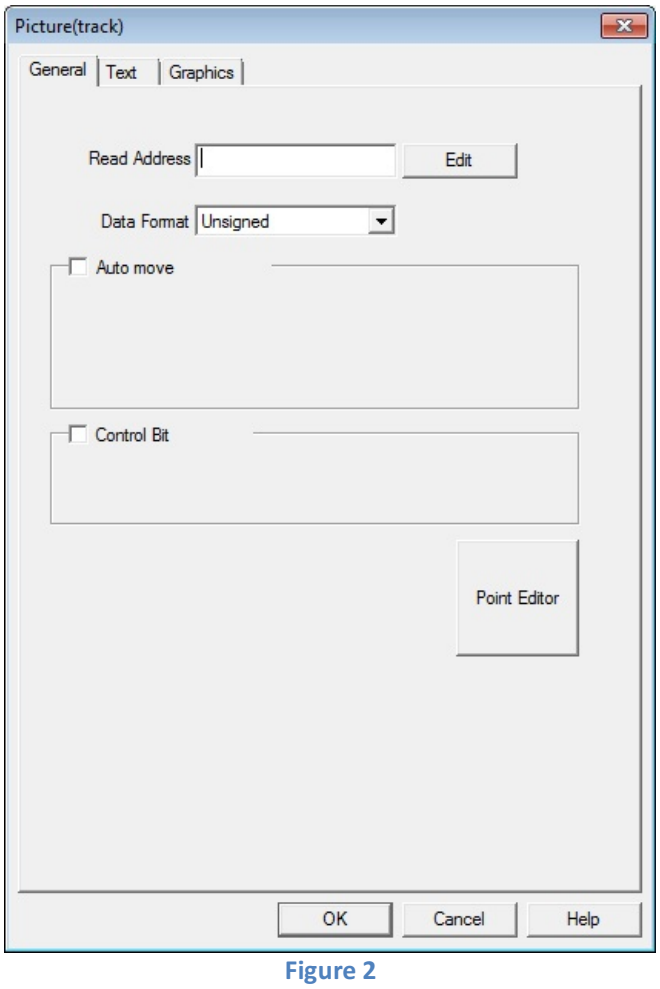


Figure 2

Classification	Properties	Description
Read address	N/A	Set designated addresses to control the state and position the picture, 2 addresses will be occupied in total. *For example set the read address as HDW0, then the addresses function as following: HDW0:state of the picture HDW1:position of the picture
Auto move	Return	After the picture reaches the last point, it will return to the start point.
	Frequency	The picture will move automatically at the designated frequency. (unit, 100ms)
	State change	Change the state by designated frequency.(unit, 100ms)
Control bit	Read address	Set designate address to trigger the movement.
Point editor	N/A	Edit the positions for the track points.

Scrolling picture

General

The picture will moves in a set movement pattern, for example, add water effect on the picture.

Settings

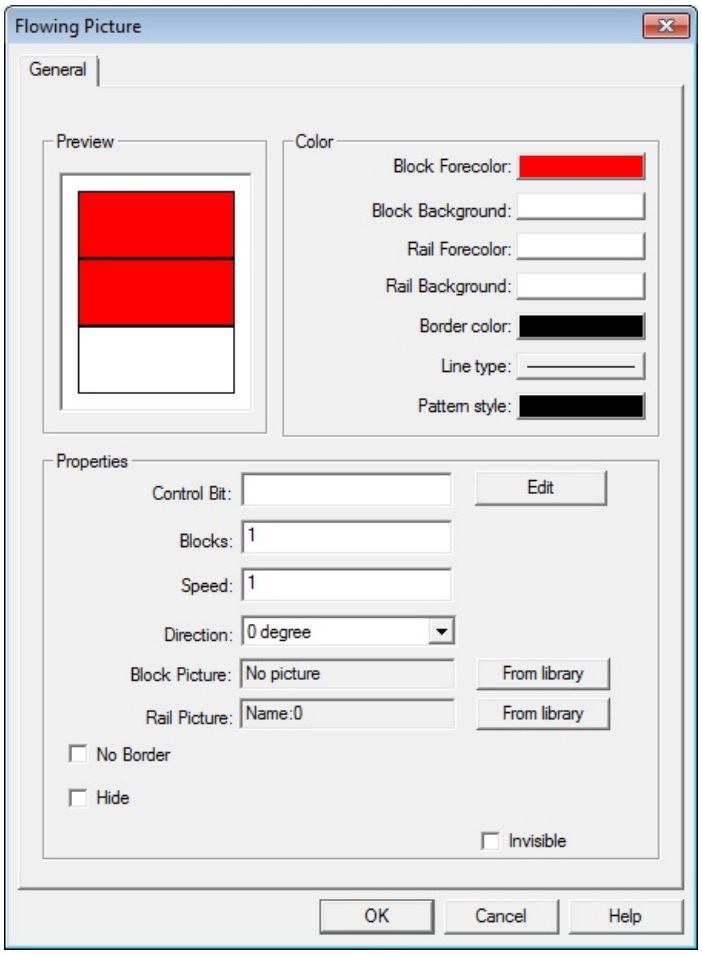


Figure 1

Classification	Properties	Description
Color	Block forecolor	N/A
	Block background	N/A
	Rail forcolor	N/A
	Rail background	N/A
	Border color	N/A
	Line type	N/A
	Pattern style	N/A
Properties	Control bit	Set the designated address to trigger the movement.
	Blocks	The number of blocks.
	Speed	Scrolling speed.
	Direction	Movement direction.
	Block picture	Customize the block.
	Rail picture	Customize the rail.
	No border	Display or hide the border.

Dynamic picture (XY axis) (General/Text/Graphics/Animation)

General

The picture will move based on the designated read address.

Settings

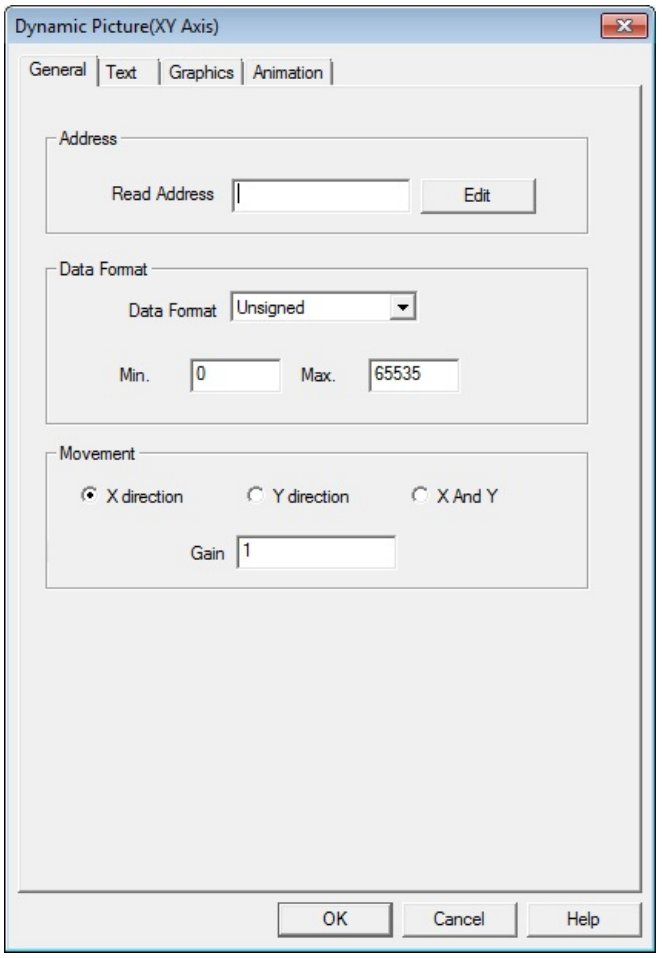


Figure 1

Dynamic Polygon

General

Connect points to construct a polygon; the points are positioned based on designated address.

Settings

Each point requires 2 addresses to locate the position.

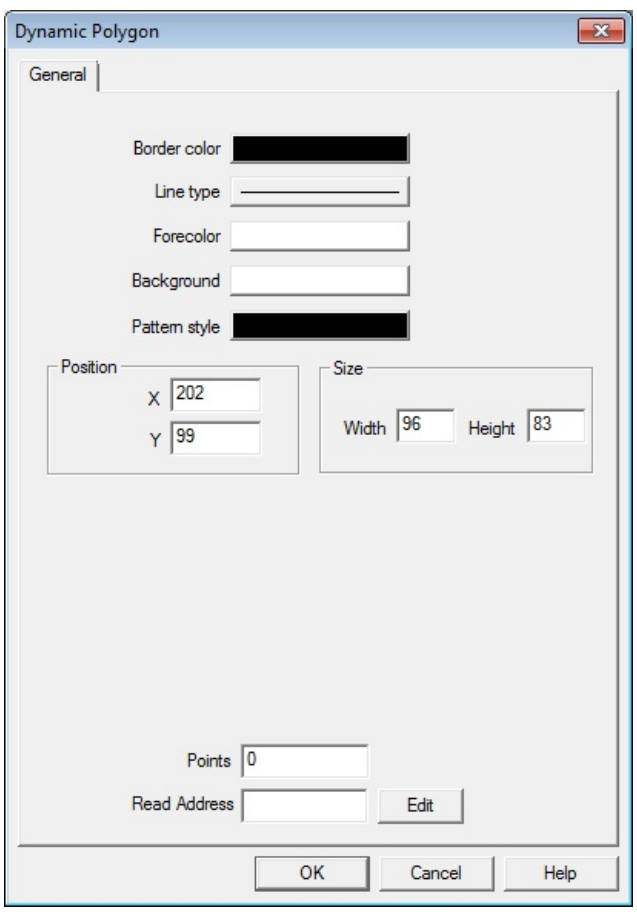


Figure 1

Properties	Description
Points	The points contained for the polygon.
Read address	Start address for the graph.

Video input display

General

Real time monitoring can be realized in this function, the video data cannot be saved.
The video input window will display on the top layer.

Settings

If the designated address set as HDW100, it will occupy the following 13 addresses.
The preset for brightness, contrast ratio, saturation and display area is required.
PAL: the maximum individual scan lines are 312 lines.
NTSC: the maximum individual scan lines are 256 lines.

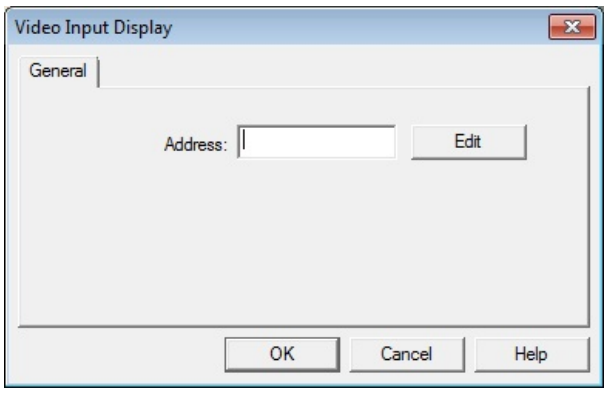


Figure 1

Address	value	Description
HDW100	0	Pause
	1	Start
	2	Stop
HDW101,HDW102, HDW103,HDW104		Set the display position
HDW105	0	Display video signal from 1 channel.
	1	Display video signal from 2 channel.
	2	Display video signal from 3 channel.
	3	Display video signal from 4 channel.
HDW106	0~255	Brightness setting, 0-minimum brightness, 255-maximum brightness.
HDW107	-128~0~127	Contrast setting:
HDW108	-128~0~127	Saturation setting.
HDW109	0~255	Color tone setting.
HDW110		Reset and initialize the video input setting.
HDW111	0	Set the display rotation direction.
	1	Rotate 90 degree.
	2	Rotate 180 degree.
	3	Rotate 270 degree.
HDW112		Reset the color setting of the video input.

Pie Graph

General

A pie chart (or a circle chart) is a circular statistical graphic, which is divided into slices to illustrate numerical proportion.
*For example: there are 3 values: 100 (HDW0), 60 (HDW1) and 40 (HDW2, 200 (100 + 60+ 40), the ratio relationship is: 50%; 30%; 20%, shown on the pie chart as figure 1.

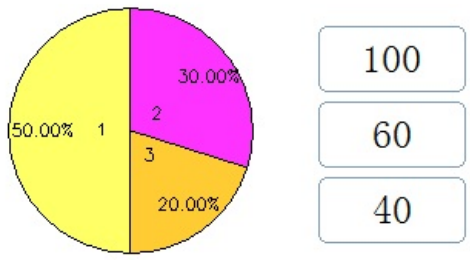


Figure 1

Setting

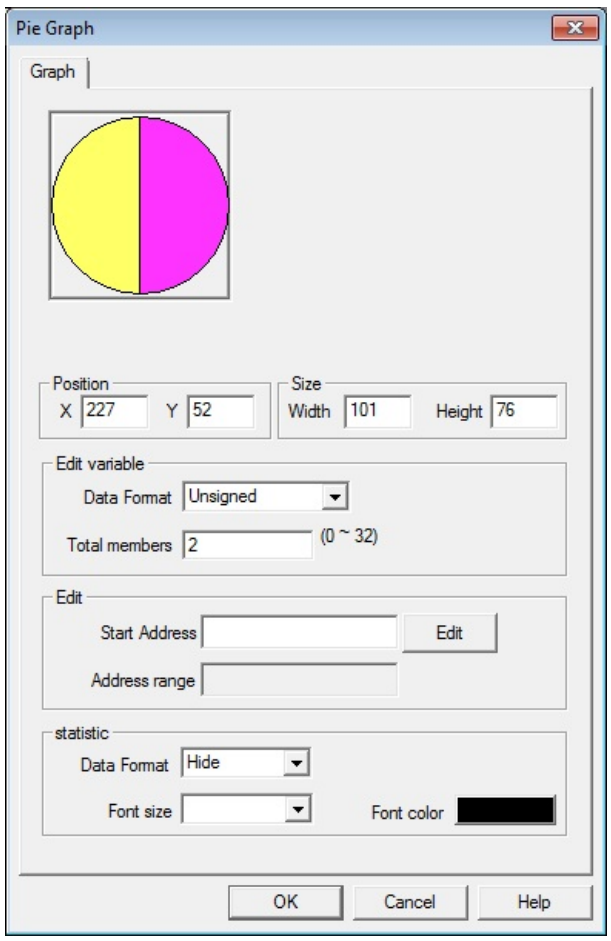


Figure 2

Classification	Properties	Description
Edit variable	Total members	The number of slices in the chart.
Edit	Start address	The pie chart value will start to read from the designated address.
	Address range	The range will calculated automatically depends on the start address and total members.
Statistic	Data format	Display format, number or percentage.
	Font size	Font size setting.

Bar graph

General

Bar graph is a chart that presents Grouped data with rectangular bars with lengths proportional to the values that they represent.

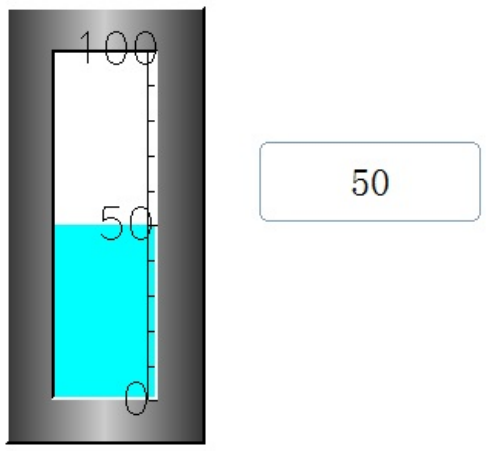


Figure 1

Settings

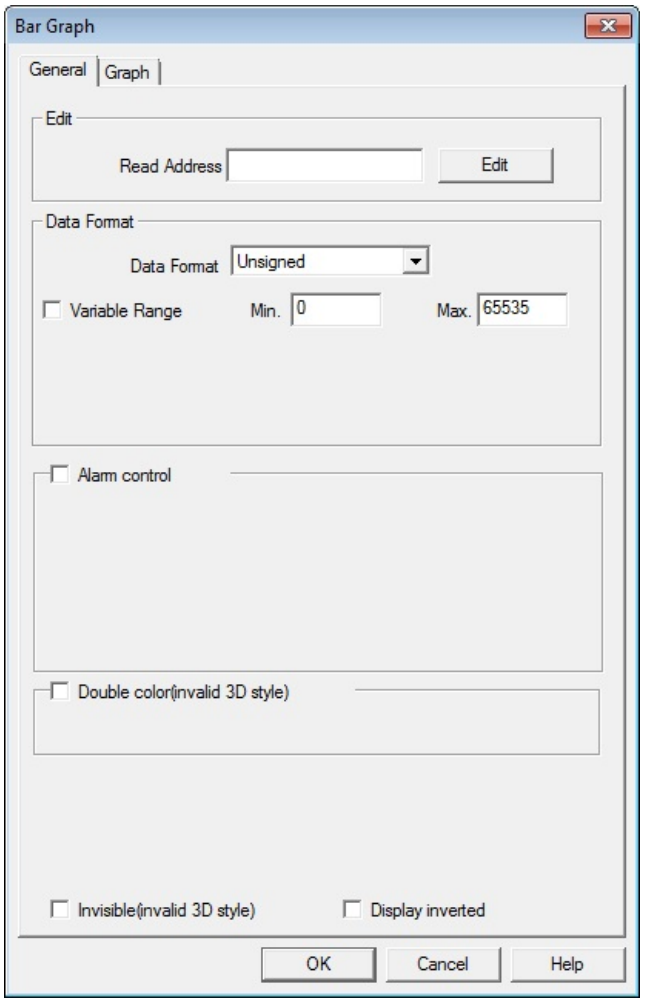
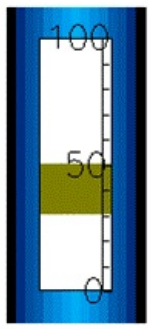


Figure 2

Classification	Properties	Description
Edit	Read address	The graph read value from the designated address.
Data format	Data format	Value format setting
	Min/Max	The data range
Alarm control	Min	When the minimum value reaches, display the alarm color.
	Max	When the maximum value reaches, display the alarm color.
		When the low limit reaches.

Double color	Read address	
Transparent	N/A	N/A
Display inverted	N/A	Invert display the graph

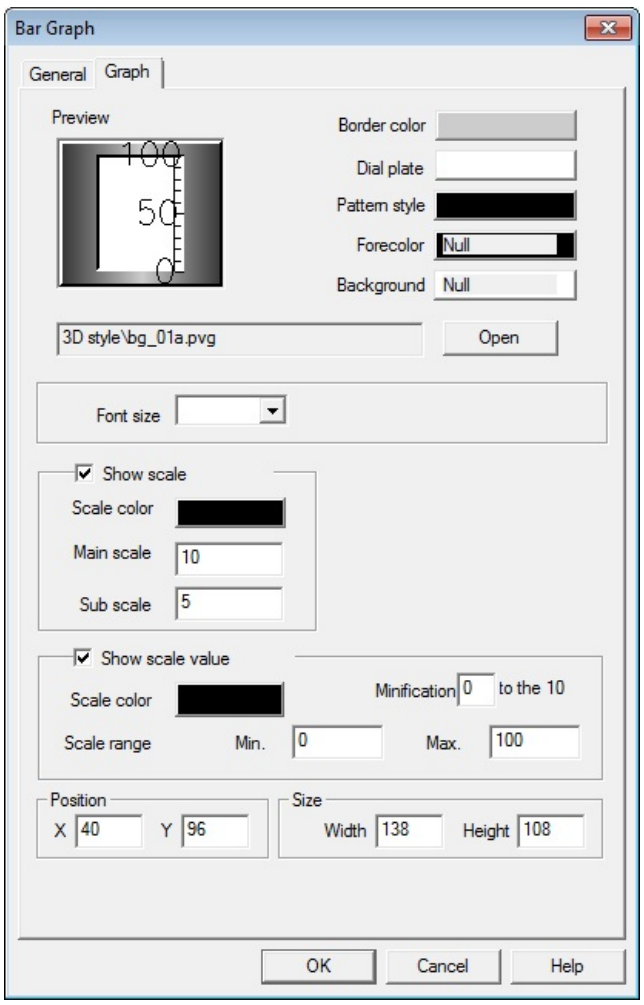


Figure 3

Classification	Properties	Description
Show scale	Main scale value	The total scale number on the bar graph.
	Sub scale value	The scale number between two main scales.
Show scale value	Minification	Display the decimal numbers.
	Scale range	The display ratio.
Font size	Style	Customize the text style.

Meter Graph

General

Meter graph display the value change in an autometer.

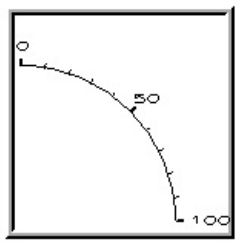


Figure 1

Settings

The settings refers to "Graph".

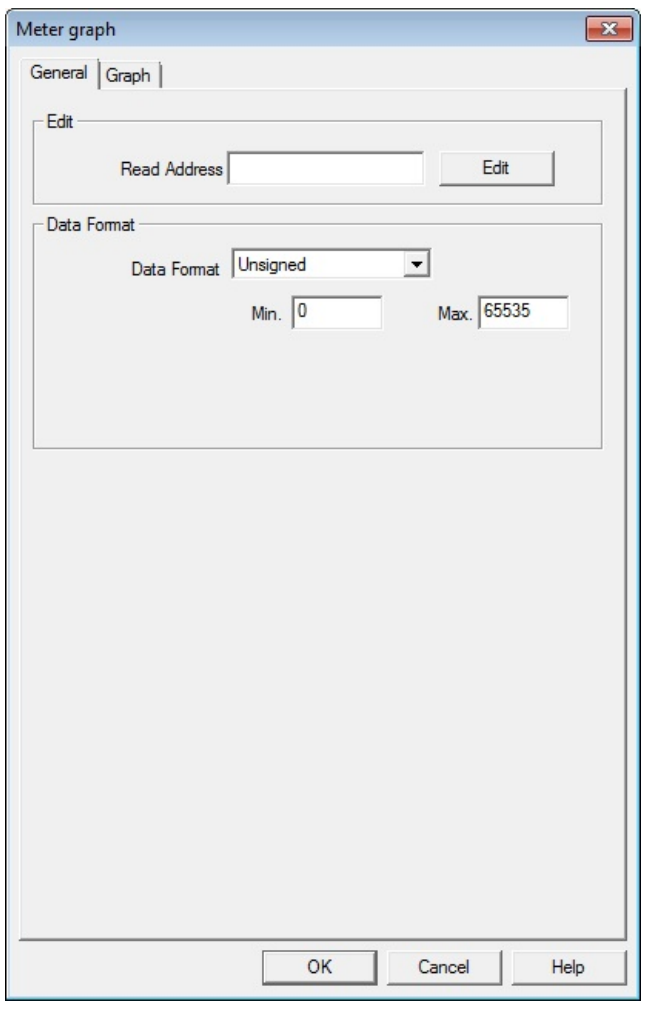


Figure 2

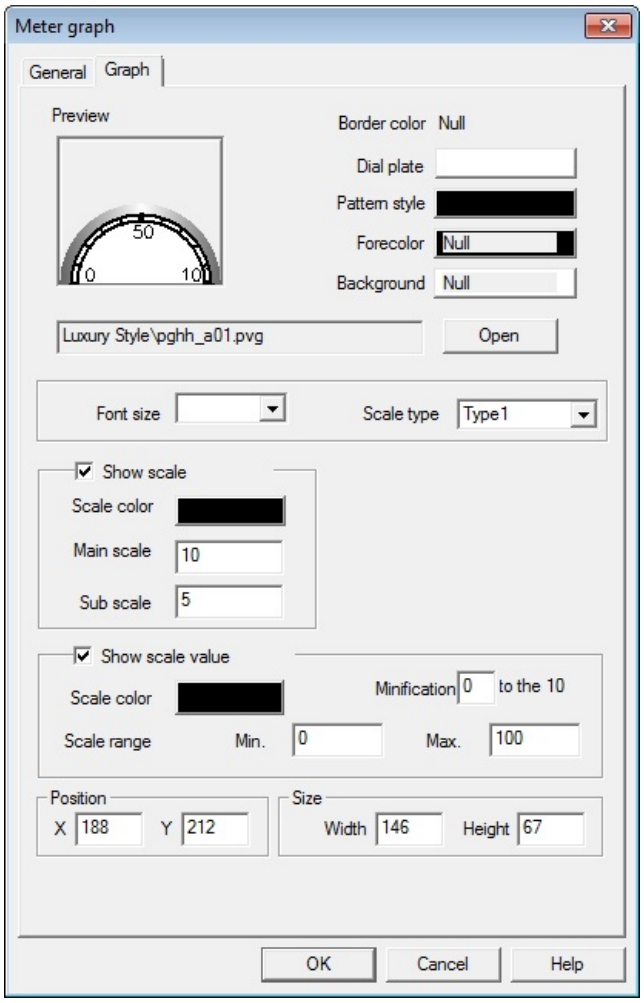

Meter graph
General
Graph
Preview
50
0
10
Border color Null
Dial plate
Pattern style
Forecolor Null
Background Null
Luxury Style\pghh_a01.pvg
Open
Font size
Scale type Type1
Show scale
Scale color
Main scale 10
Sub scale 5
Show scale value
Scale color
Minification 0 to the 10
Scale range Min. 0 Max. 100
Position
X 188 Y 212
Size
Width 146 Height 67
OK
Cancel
Help

Figure 3

Column Graph

General

Column graph reflect the data change as fluid change in the tank.

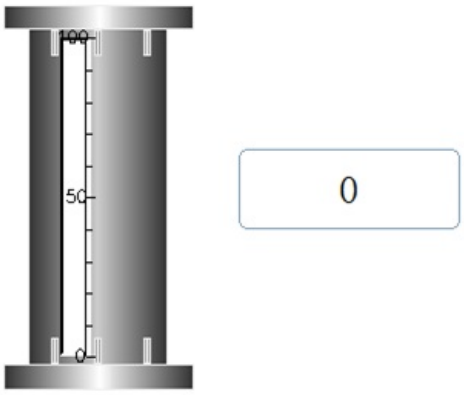


Figure 1

Settings

The settings refers to "Bar graph".

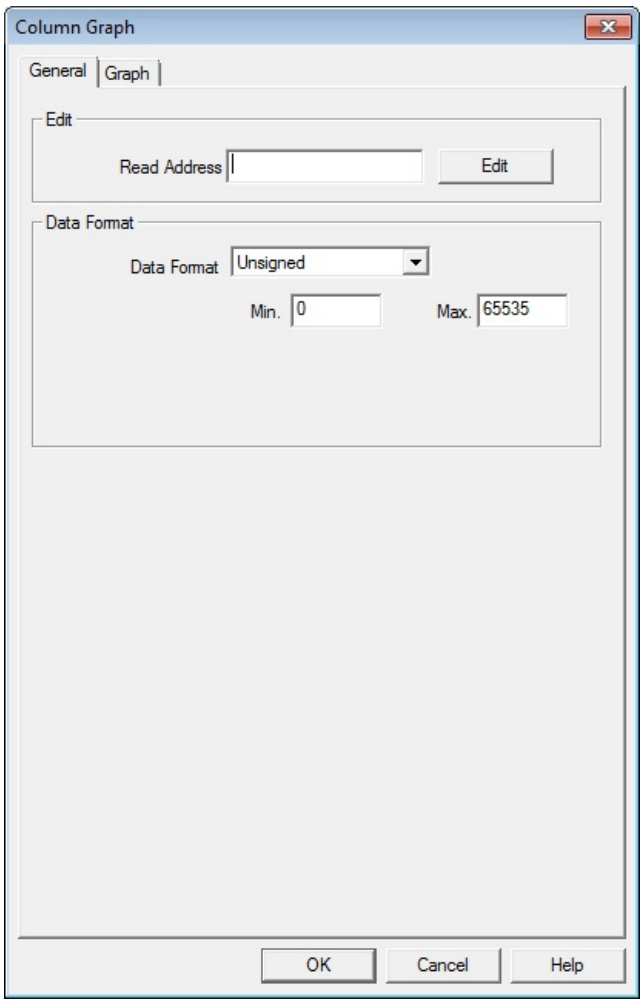


Figure 2

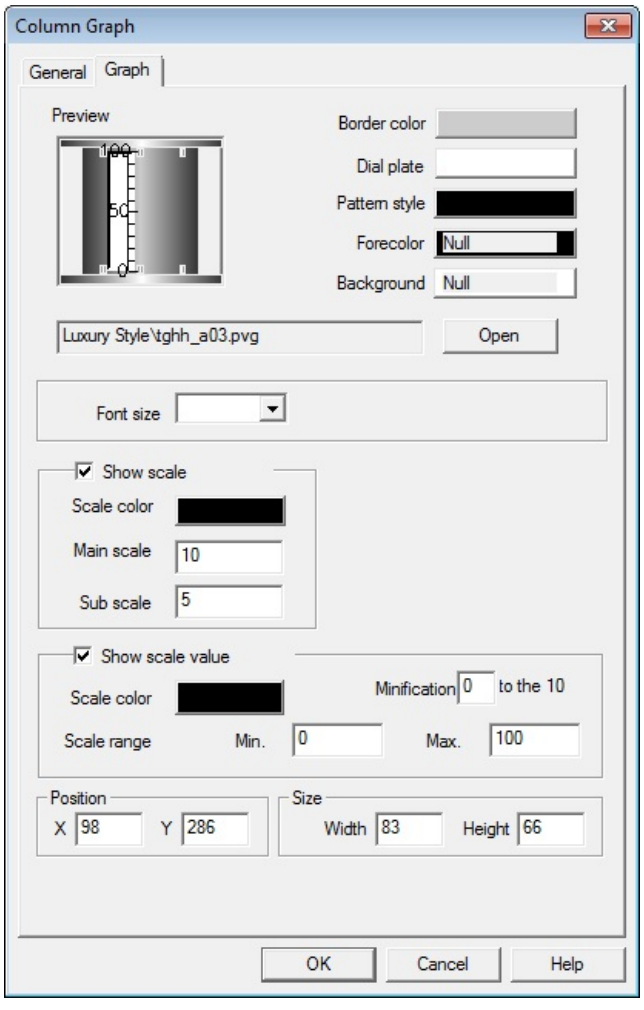

Column Graph
General
Graph
Preview
100
50
0
Border color
Dial plate
Pattern style
Forecolor
Null
Background
Null
Luxury Style\tghh_a03.pvg
Open
Font size
Show scale
Scale color
Main scale
10
Sub scale
5
Show scale value
Scale color
Minification
0
to the 10
Scale range
Min.
0
Max.
100
Position
X
98
Y
286
Size
Width
83
Height
66
OK
Cancel
Help

Figure 3

Meter (General/Text/Graph)

General

Graph shows the data change in a combination of pointer and dial.

Settings

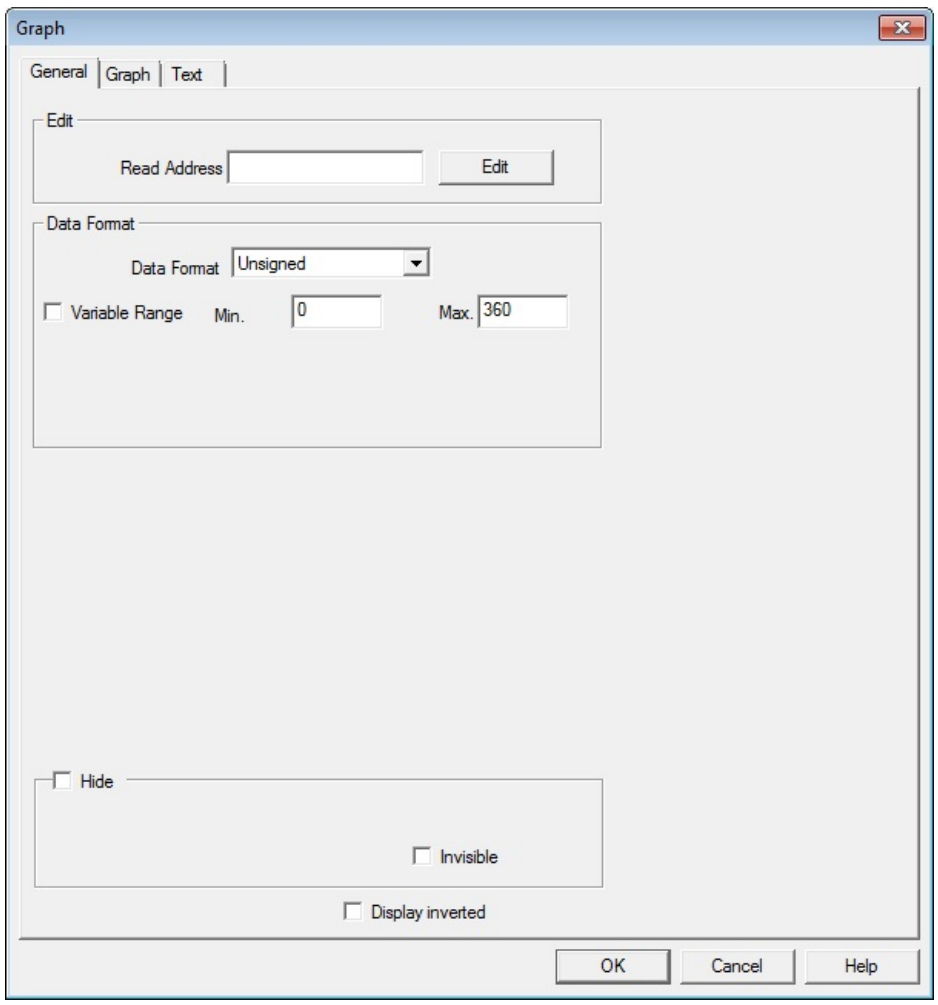


Figure 1

Classification	Properties	Description
Edit	Read address	The value is red from designated address.
Data format	Data format	The format of the data.
	Max/Min	The data range.
Display inverted	N/A	Invert the display value.

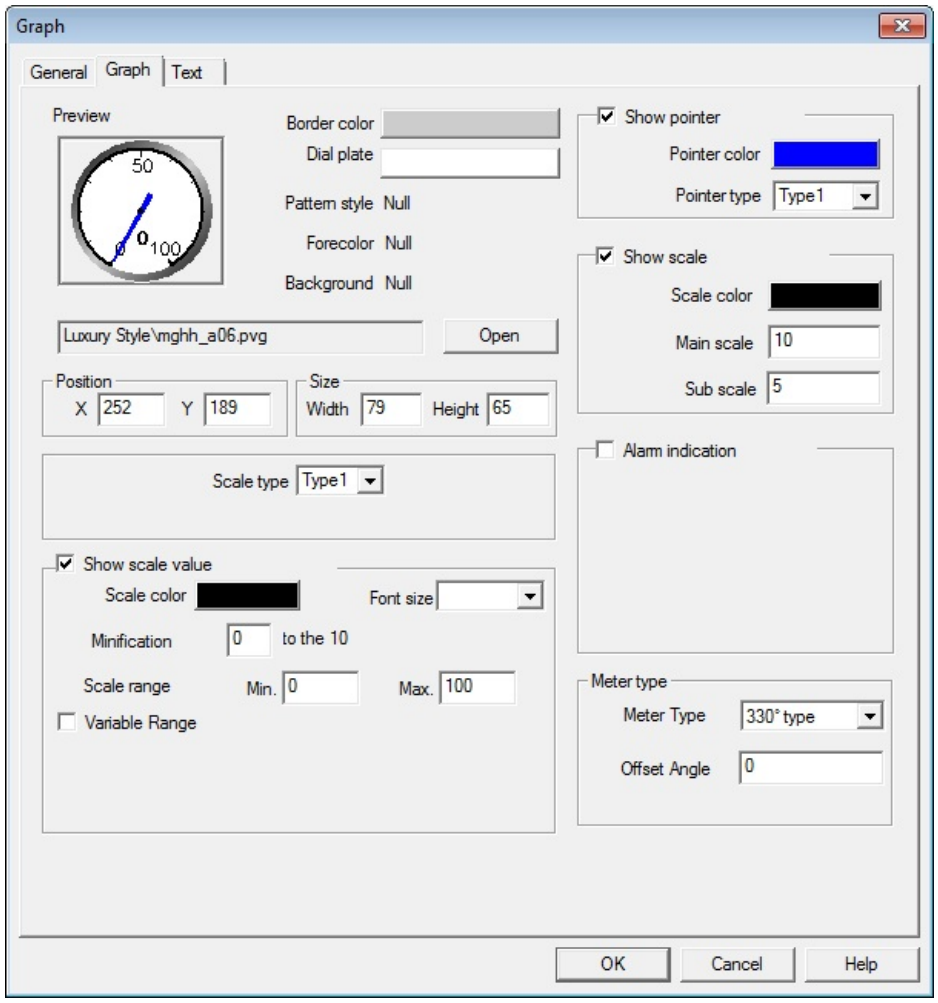


Figure 2

Classification	Properties	Description
Show scale value	Minification	Minify the scale.
	Scale range	The display ratio.
	Dynamic limit	The limits are determined by the designated addresses.
Show pointer	Pointer color	N/A
	Pointer type	N/A
Show scale	Main scale value	The total scale number on the bar graph.
	Sub scale value	The scale number between two main scales.
Abnormal indication	Normal range	The normal data range.
	Low limit color	The color of the area lower than the limit .
	High limit color	The color of the area higher than the limit.
Meter type	Circle meter type	330 degree or 360 degree.
	Starting offset angle	The “0” position.

Clock

General

The time is displayed in the clock graph

Figure 1

Settings

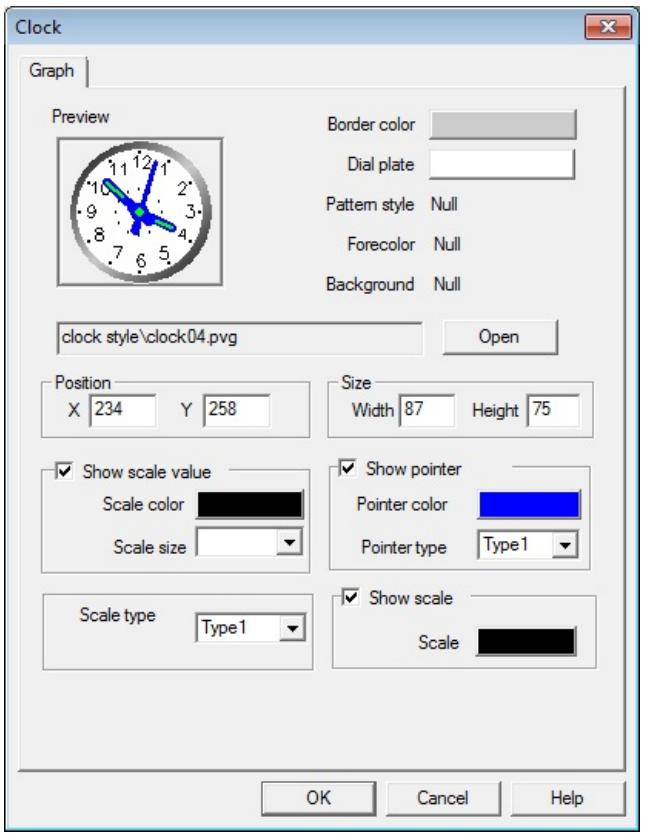


Figure 2

Classification	Properties	Description
Show scale value	Scale color	The scale(text) color.
	Scale size	N/A
Scale type	N/A	N/A
Show pointer	Pointer color	N/A
	Point type	N/A
Show scale	Scale color	N/A

Timer

General

The designated command will be executed after the timer set.

Setting

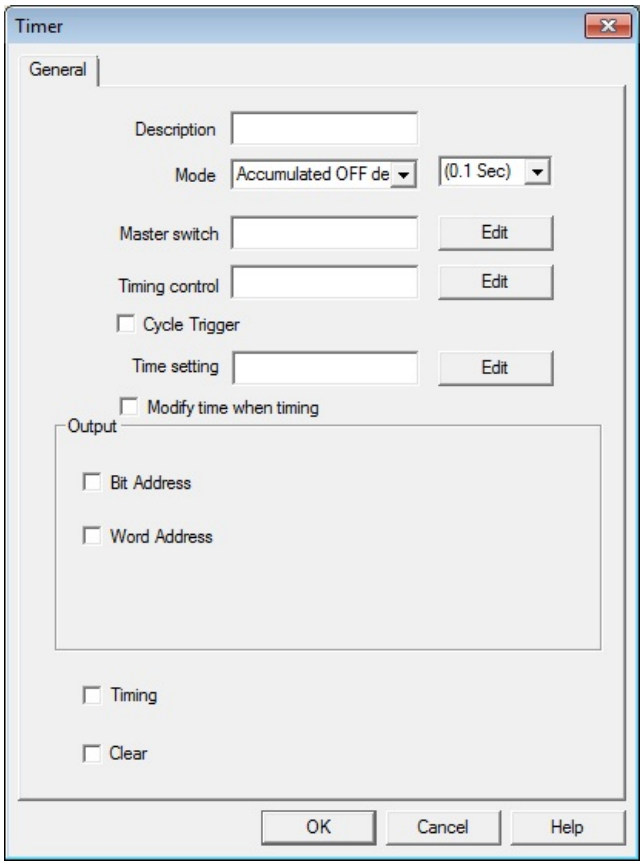


Figure 1

Classification	Properties	Description
Mode	N/A	The timer function.
Timing trigger	N/A	The timer triggers address.
Counted time	N/A	The timer will reset after the designated time reaches.
Modify time when timing	N/A	Timer data can be modified while counting, default setting is not ediable.
Output	Bit address	When the timer reaches the designated time, trigger the address.
	Word address	When the timer reaches the designated time change the value of the word address in the selected mode (set value, increase, decrease).
Timing	N/A	Numerical display the timer.
clear	N/A	Clear the timer record, when the address is triggered.

Bit Lamp

General

Bit lamp indicates the state of corresponding bit switch.

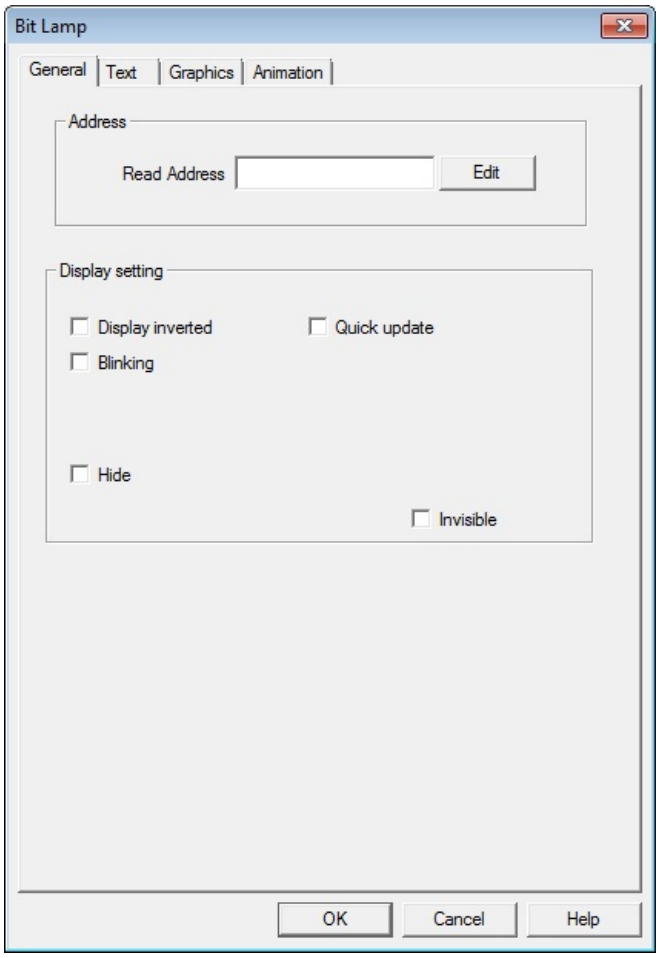


Figure 1

Classification	properties	Description
Read address	N/A	The address corresponds to the switch.
Display inverted	N/A	Display the inverted state.
Quick update	N/A	Increase the read frequency.
Blinking	Blink when ON	N/A
	Blink when OFF	N/A
	Blink alternately when ON	N/A
	Blink alternately when OFF:	N/A
	Blinking frequency (100ms)	*For example: the input value is 5, the blinking frequency will set as 500ms.
Hide	Hide when OFF/Hide when ON	N/A
	Control bit	Trigger the control to hide the indicator.

Multi-state Lamp

General

The Indicator will display different text or picture corresponds to different state.

*For example, the state of the switch is 2, the indicator shows the designated picture or text corresponds to this state.

Settings

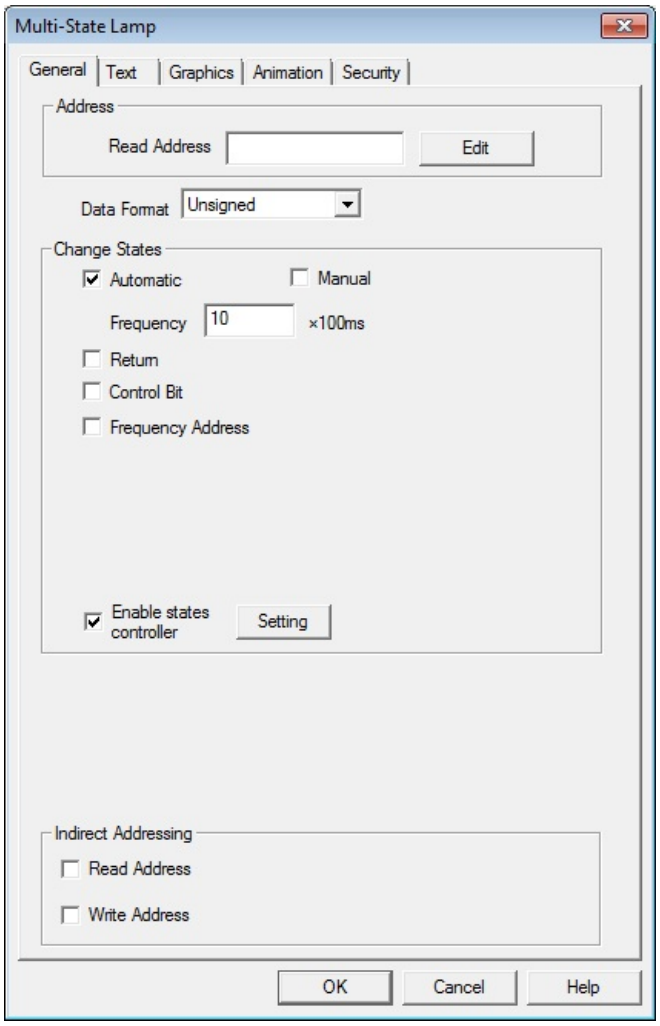


Figure 1

Classification	Properties	Description
Change states	Automatic	The states will change automatically in designated frequency.
	Frequency	The time interval to change the states, unit set as (ms). *For example, the input value is 5, the frequency will be 500ms.
	Return	Change back to the initial state after the last state reaches
	Control Bit	Set a switch to trigger the state change mode.
	Frequency Address	The frequency can set as a variable, the value will be red from designated address, unit set as (ms).
	Enable states controller	*For example, in default setting, state 0 corresponds to the value of the read address is 0, the state controller can change the state number corresponds to the designated address value.
	Manual	Change the state manually.

Double bits lamp

General

Double bits lamp read from two addresses, display the states depend on the different combination of the addresses.

Setting

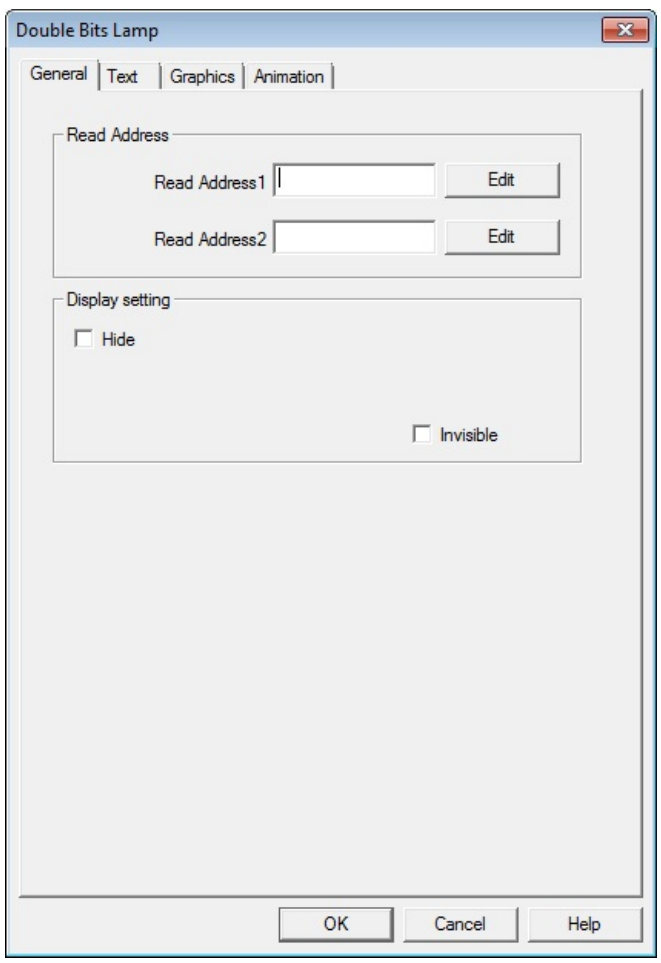


Figure 1

Classification	properties	Description			
Read Address	Read address 1	Read the value from designated address to show the state.			
	Read address 2	N/A			
States	4 states			Address 2	
				ON	OFF
		Read address 1	ON	State 1	State 2
			OFF	State 3	State 4

Indirect window (General/Animation)

General

Indirect window is designed for display designated screen, depends on the screen number input.

Settings

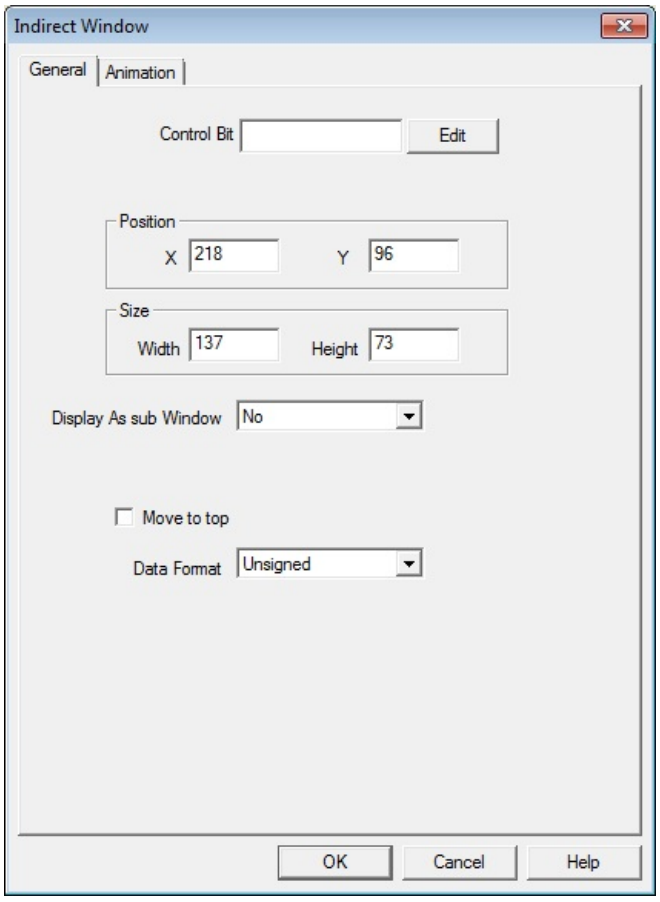


Figure 1

Classification	Properties	Description
Control Bit	N/A	The switch to trigger the window.
Display as sub	N/A	Active the control bar for the sub-window.
Move to top	N/A	Top priority on display .

Direct Window (General/Animation)

General

Direct window will display a designated sub-window when the set conditions are met.
Features of direct window:
a. Direct window only displays sub-window.
b. The sub-window displays depending on the trigger conditions.
c. The window size and position depend on the size and position of the object.

Settings

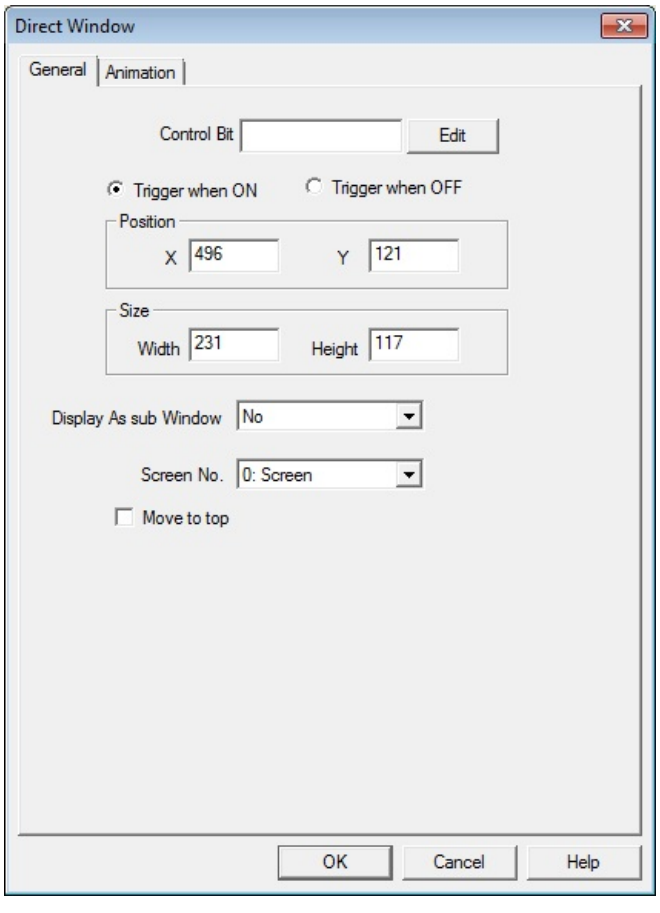


Figure 1

Classification	Properties	Description
Trigger	Control bit	The switch to trigger the window.
	Trigger when ON	The window will show up when the trigger is set ON.
	Trigger when OFF	The window will show up when the trigger is set OFF.
Display as sub window	N/A	Active the control bar for the sub-window.
Screen Number	N/A	The designated screen displays when triggered the switch.
Move to top	N/A	Top priority on display.

Drop-Down list (General/Text/Graphics)

General

Drop-Down list is designed for display the designated state by text, to select the text in the list, the corresponding state will change, and the write address will change to the designated state value. (There are 32 states in total, as “0-31”).

Settings

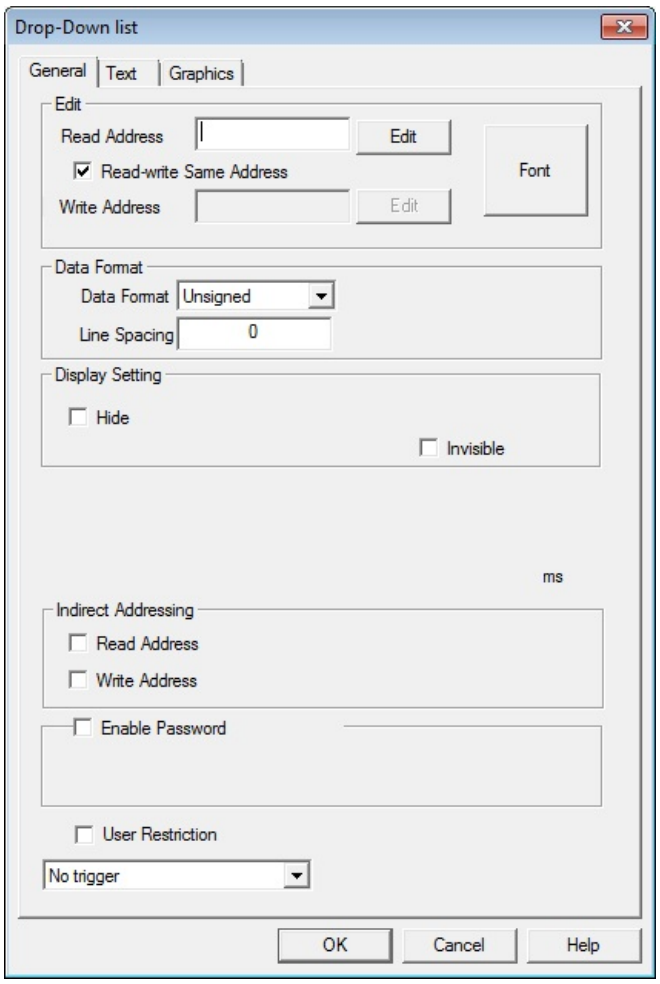


Figure 1

Classification	Description
Data format	Select the data format from Unsigned, signed and BCD.
Line Spacing	The distance between two different options in the list.

Dynamic Text (General/Text)

General

A designated Text content will loop display on the HMI. This function is mainly used for display the advertisement on the HMI.

Settings

The content needs to be set in the text option.

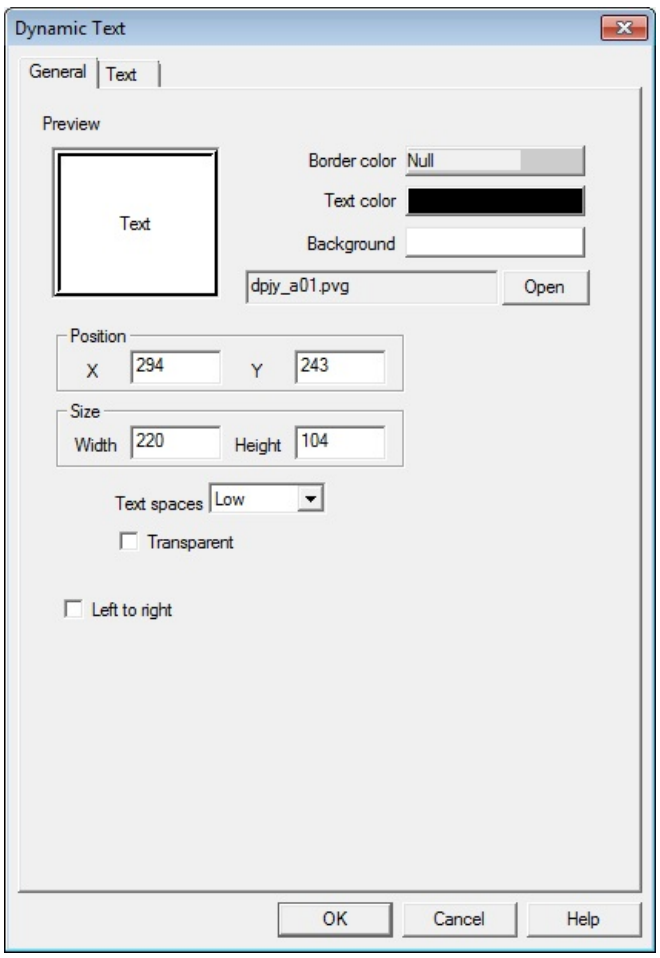


Figure 1

Classification	Description
Text speed	Select the scrolling speed for the text.
Transparent	Set the feature transparently.
Left to right	The text scrolling from left to right, default setting as right to left.

Printer

General

Printer can save the designated display area as picture, or print-out the text though the micro printer.

Settings

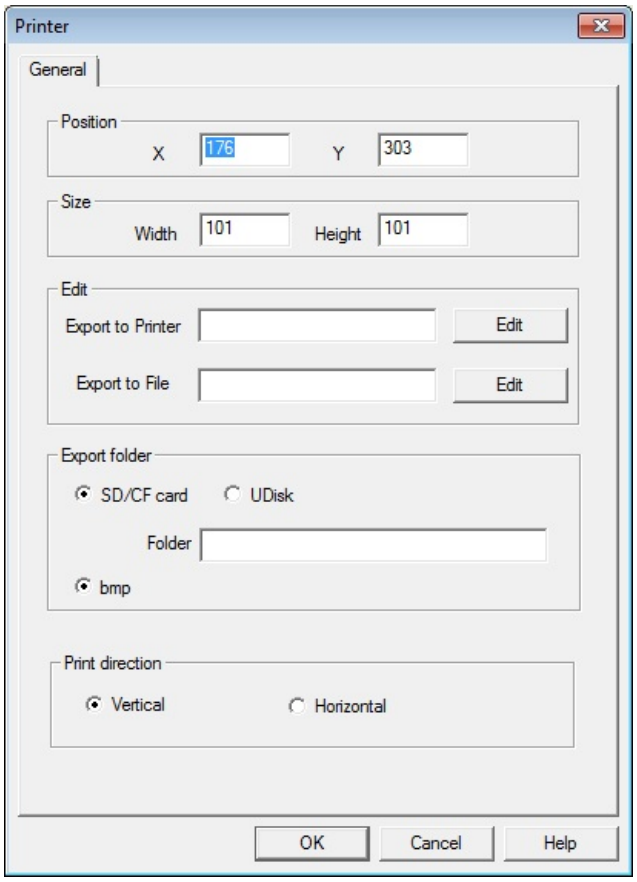


Figure 1

Classification	Properties	Description
Edit	Export to printer	The printer address to output the text content.
	Export to file	Saving location of the screen-shot on designated area.
Export folder	SD/CF card UDisk	SD/CF card or Udisk to save the screen-shot on designated area.
	Folder	Simulation: Save the picture to designated PC folder. HMI: Save the picture to SD/CF card or Udisk.
	BMP	Screen-shot format.
Print direction	Vertical Horizontal	Print-out direction, while work with the micro printer.

Date Display (General/Animation)

General

Display the date on HMI.

Settings

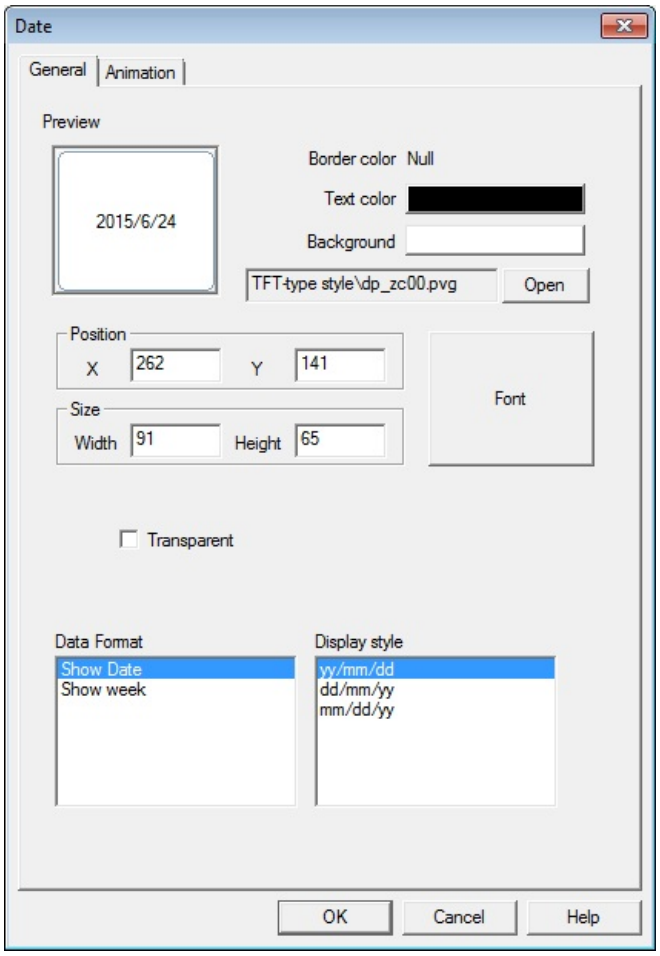


Figure 1

Classification	Properties	Description
Format	Show date	yy/mm/dd:year/month/day mm/dd/yy:month/day/year dd/mm/yy:day/month/year
	Display style	Chinese1 English1 Chinese2 English2

Time Display (General/Animation)

General

Time display shows the system time on the main screen.

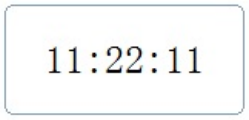


Figure 1

Settings

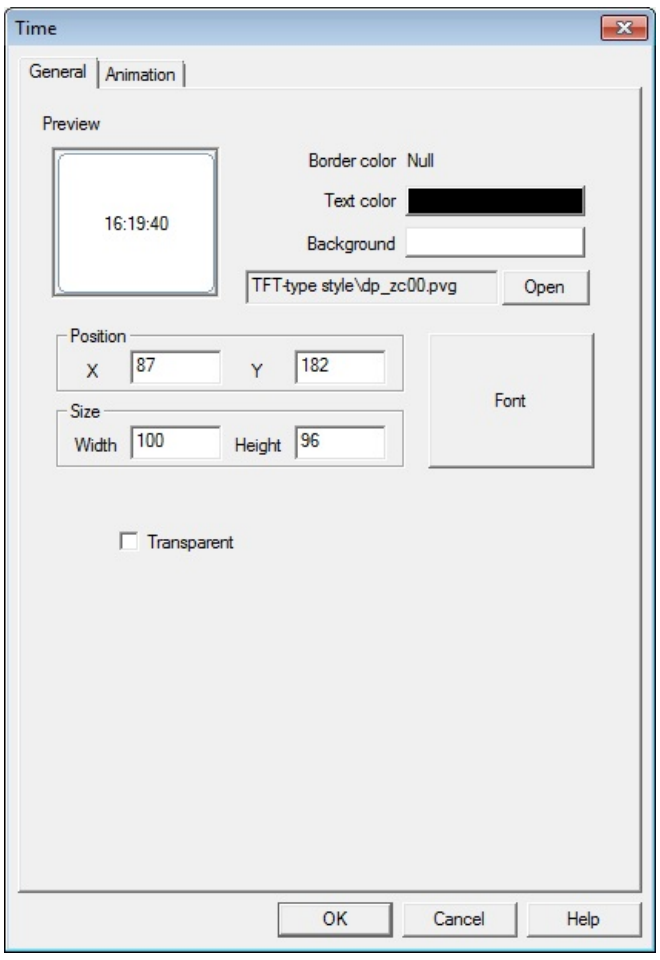


Figure 2

File List (Graphics/Query)

Query

Query is mainly used for data searching; the required data will be acquired by the set condition.

Settings

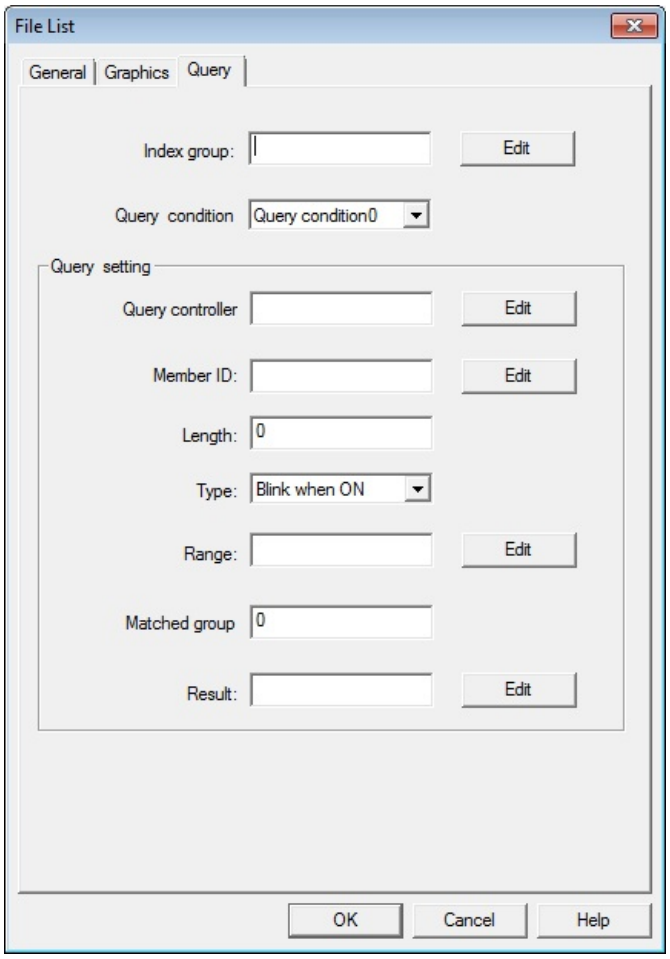


Figure 1

Classification	Properties	Description
Index group	N/A	Display a designated data group.
Query condition	N/A	The conditions set to select the required data, support maximum 10 conditions.
Query setting	Query controller	Trigger the controller to query the designated data group.
	Member ID	Query will start from the designated member ID.
	Length	The ID number needs to be queried.
	Type	Number, time, string.
	Range	The data limits.
	Matched group	Store maximum 50 groups data queried.
	Result	N/A

Recipe Display

General

This function is designed to display and modify the formula of the recipe.

Settings

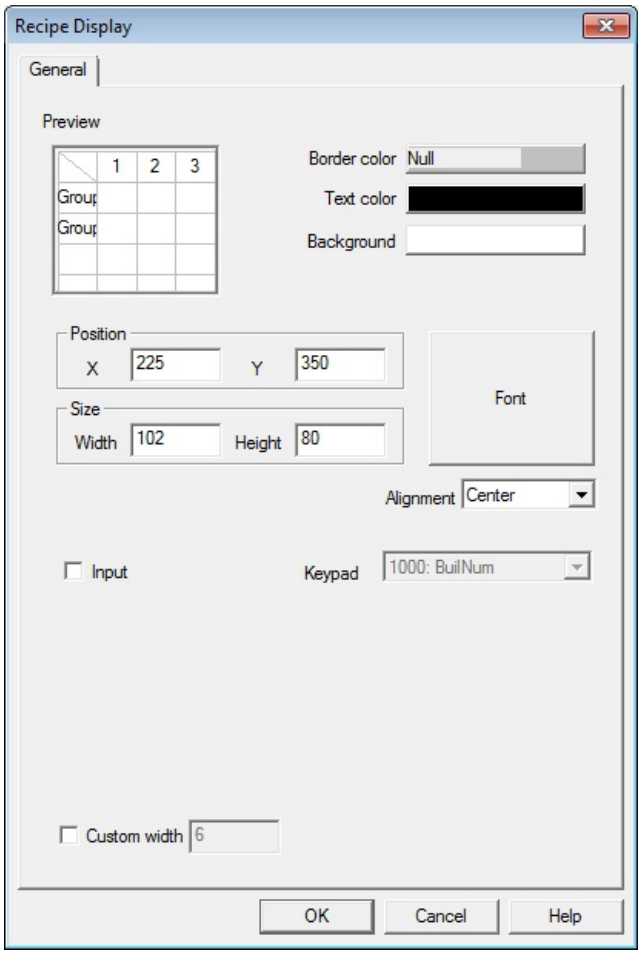


Figure 1

Classification	Description
Input	Modify the recipe is possible when input function active.
Alignment	The text position in the frame.
Custom width	N/A

List Box (General/Text/Graphics)

General

List box is able to modify the value of the designated bit address from 0- 31.

Settings

Settings refer to drop-down list.

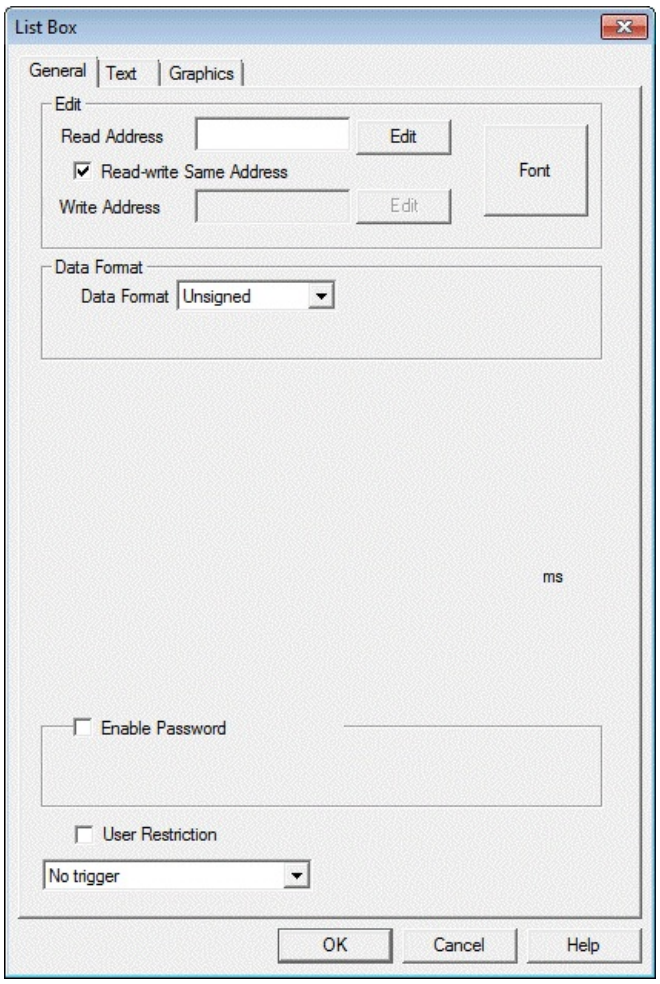


Figure 1

Draw

General

Draw function can be found in the menu bar [parts]-[draw], or in the parts library.

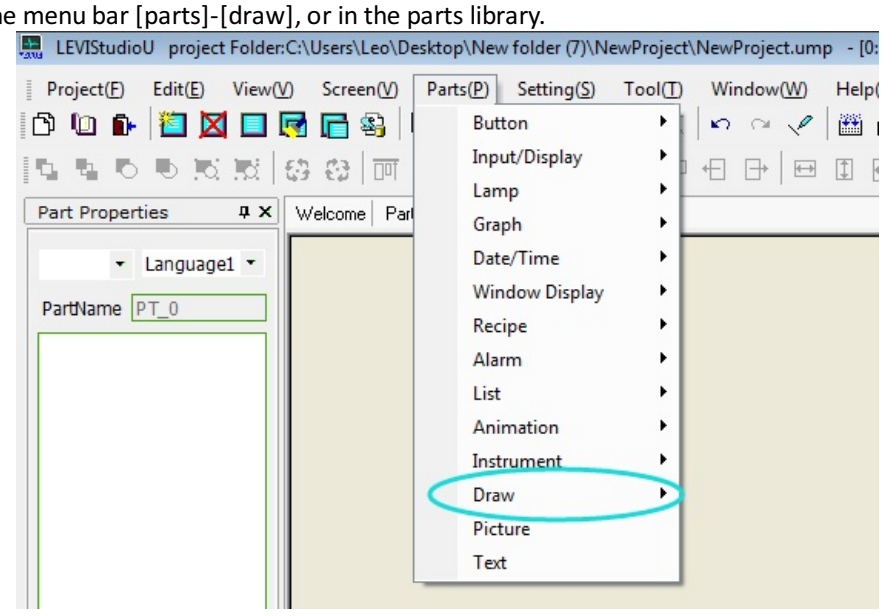


Figure 1

Settings

position: the top left position of the drawing shown in the screen.
Line color:select the color for the drawing
Line type: select the type of the drawing
size:the size of the drawing, expressed as pixel number.

Point

Settings shown as below.

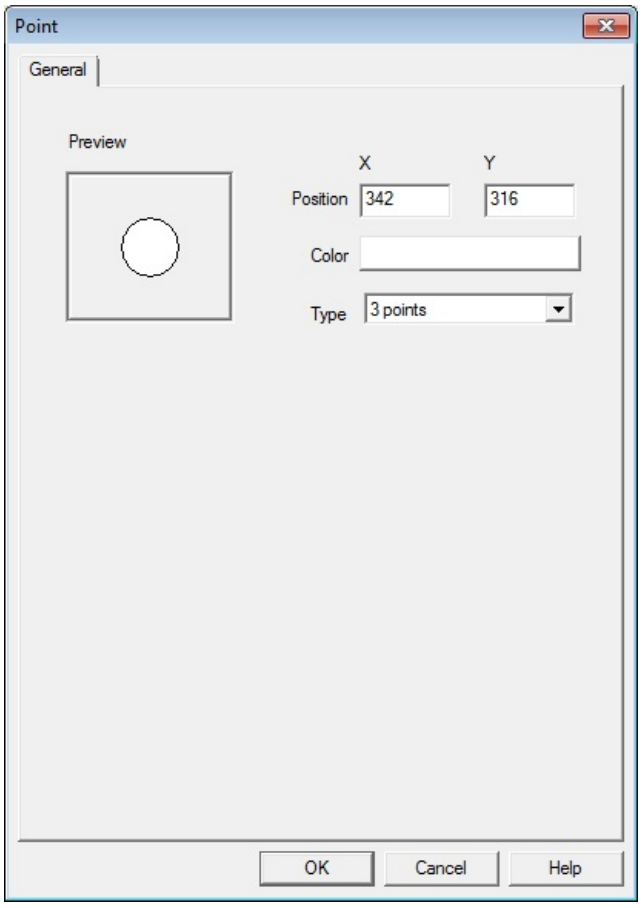


Figure 1

Line

The setting window shown as below.

Automatic adjustment range:in the range of designated value, the line will remain vertical.

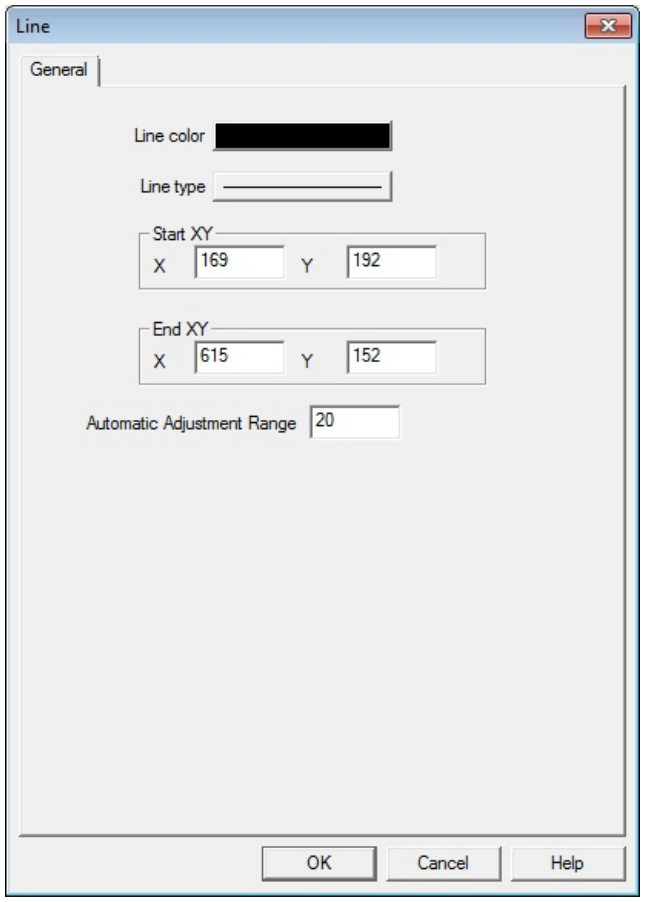


Figure 1

Polygon

New polygon

Left click the mouse to draw the profile of the polygon, right click to finish drawing.

Settings

The points of the polygon can be modified by the point Editor at polygon setting window.
Drag the point needs to be modified to the designated position.

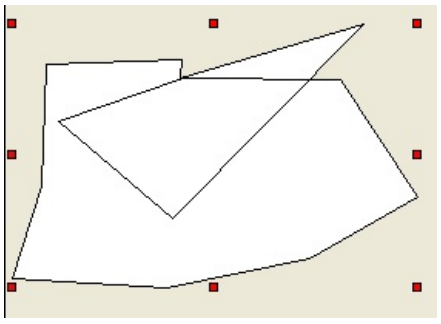


Figure 1

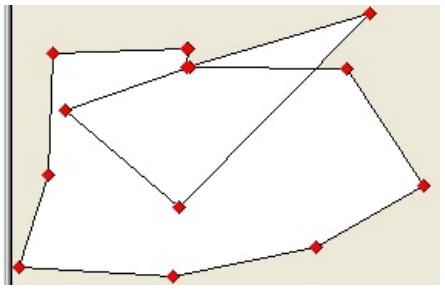


Figure 2

Cycle

Cycle setting window shown as below.

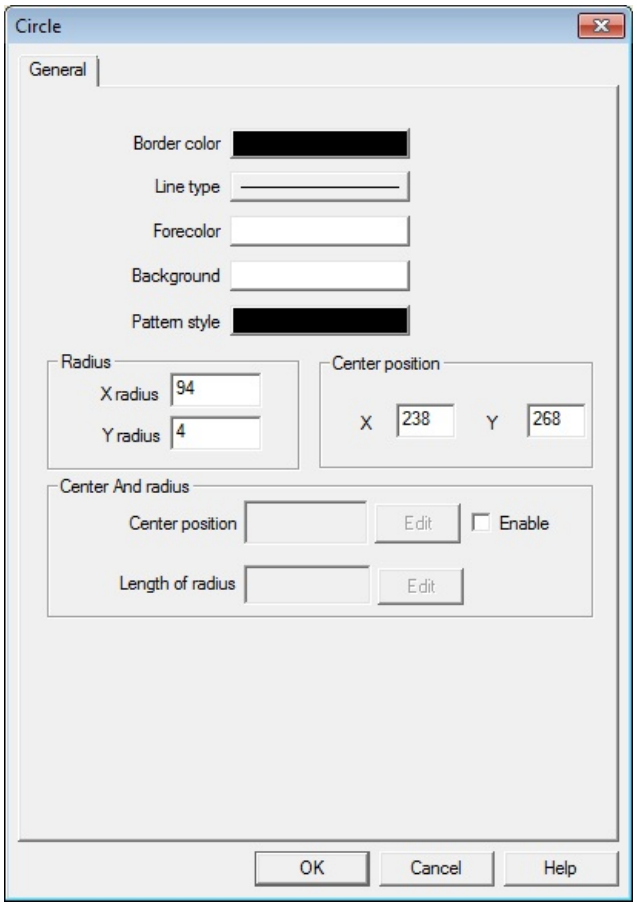


Figure 1

1. Radius: X and Y radius represents the horizontal and vertical radius respectively.
2. Dynamic cycle: The center position and X, Y radius are decided by the designated addresses.

Arc

General

Left click to drop the top right point of the arc.
Left click again and rotates clockwise to erase the redundant part of the arc, rotates counter clockwise to create the desired arc.
Left click again to finish drawing.

Settings

The profile, start point, end point can be modified by simply dragging.

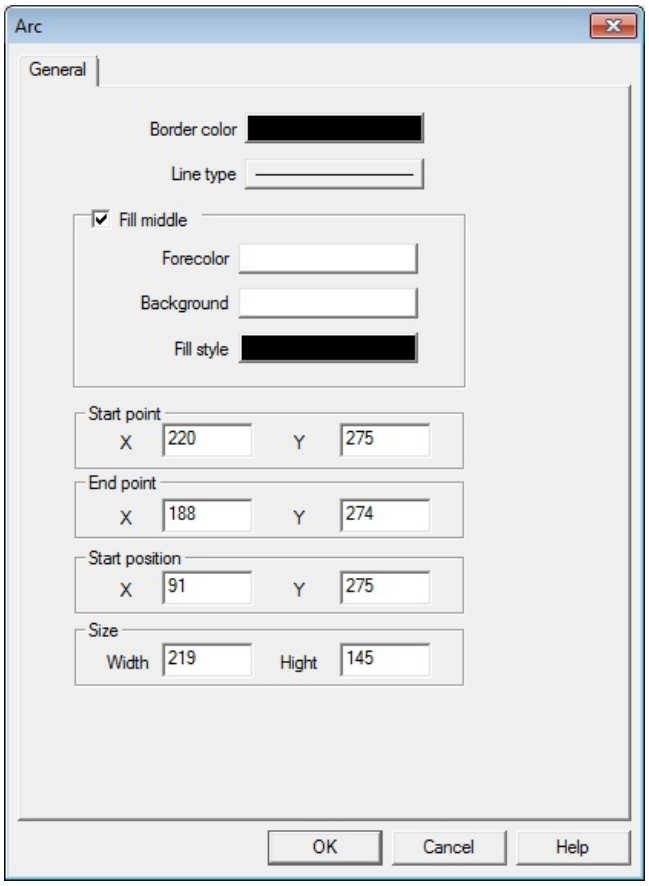


Figure 1

Rectangle

Settings

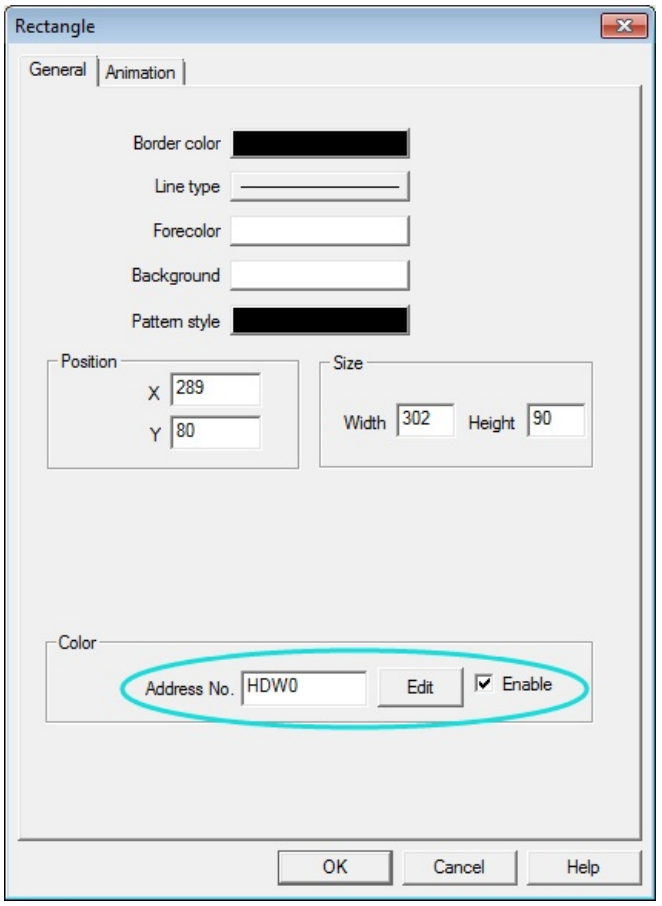


Figure 1

The color will change as the value input changes. The color setting will occupy 3 continues addresses, for example, the designated address is HDW100, the following two addresses, HDW101, HDW102 will all be occupied, the value input range is between 0-255, the higher the value the darker the color.
HDW100 represents Red;
HDW101 represents Green;
HDW102 represents Blue.

Text

Textbox display the designated text content.

Broken Line

General

Left click on the screen to drop the start point.
Drop the following points by left click on designated area.
Right click the mouse to finish the drawing.

Settings

The points of the polygon can be modified by the point Editor at polygon setting window.
Drag the point needs to be modified to the designated position.

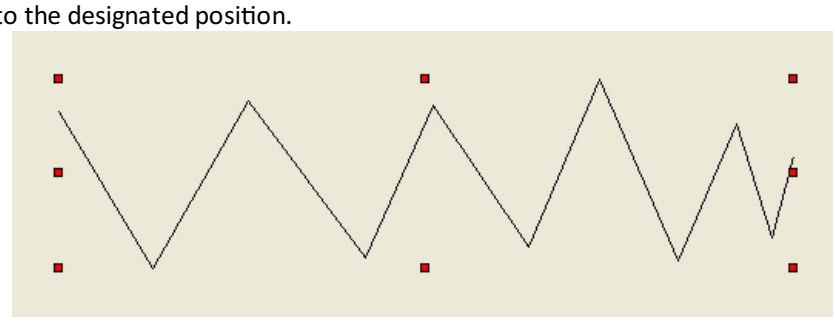


Figure 1

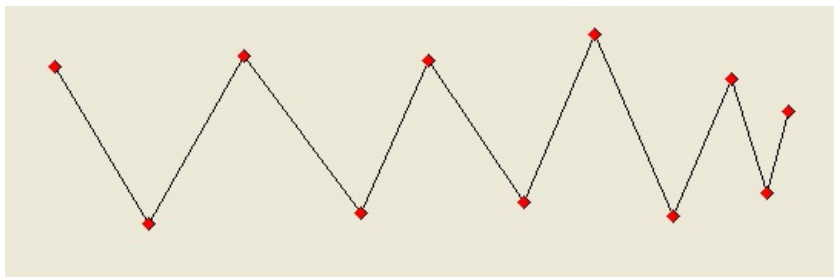


Figure 2

Linear Scale

Settings

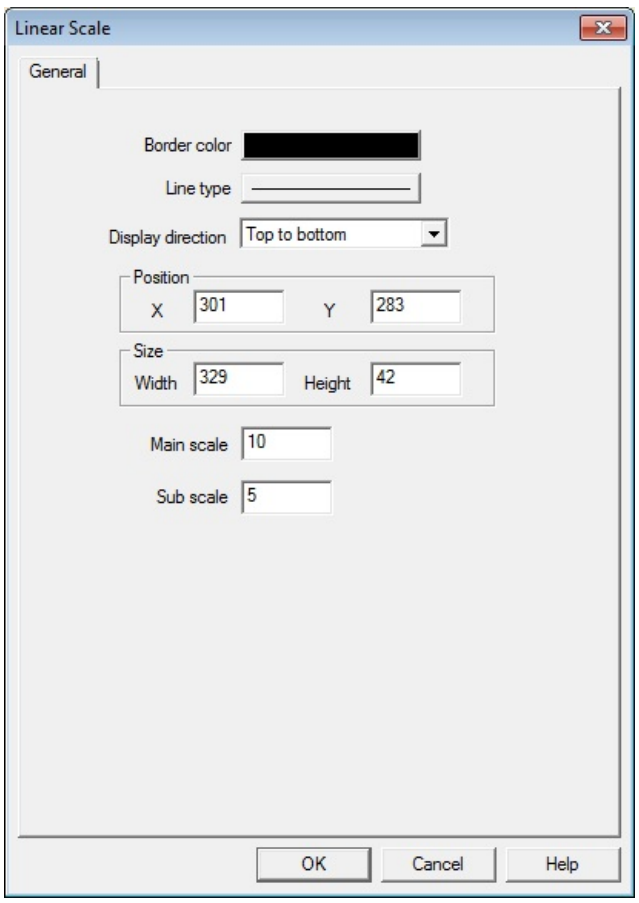
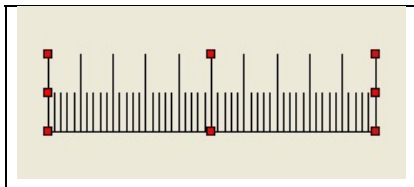
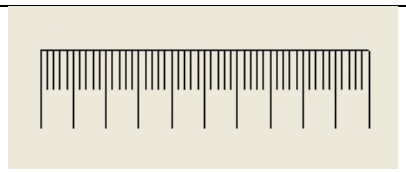
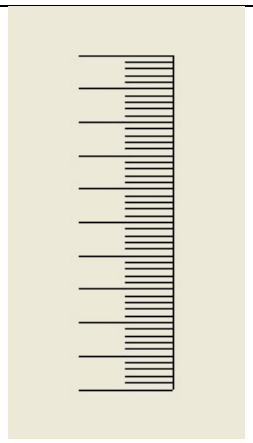
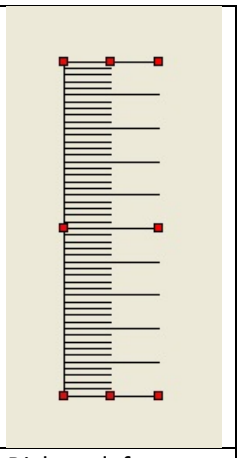


Figure 1

Display direction: There are 4 display directions list as, top to bottom, bottom to top, left to right, right to left, corresponding to the following pictures respectively.

			
Top to bottom	Bottom to top	Left to right	Right to left

1. Main scale: default main scale number is 10.
2. Sub scale:the number of the sub scale is between the adjacent main scales.

Arc Scale

General

Left click to drop the top right point of the arc scale.
Left click again and rotates clockwise to erase the redundant part of the arc, rotates counter clockwise to create the desired arc scale.
Left click again to finish drawing.

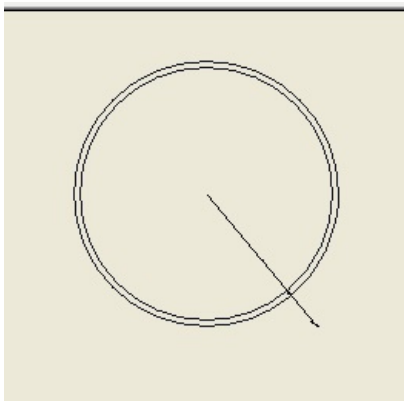


Figure 1

Settings

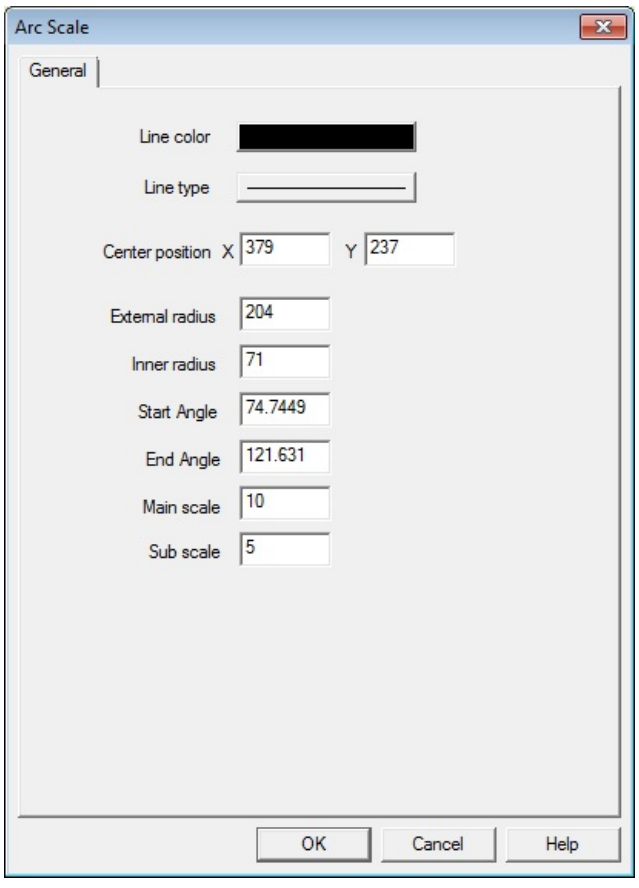


Figure 2

1. Start angle:The angle corresponds to the X axis and the first radius drew.
2. End angle:The angle corresponds to the X axis and the last radius drew.
3. Main scale:Default main scale number is 6.
4. Sub scale:The number of the sub scale is between the adjacent main scales.

Picture

Settings

The picture can be put on the screen by following steps;
Optimize the picture frame on the screen;
Double click the object and select from gallery;
To import customized picture, select [import]option;
System will auto adjust the color of the picture harmoniously with the HMI background.

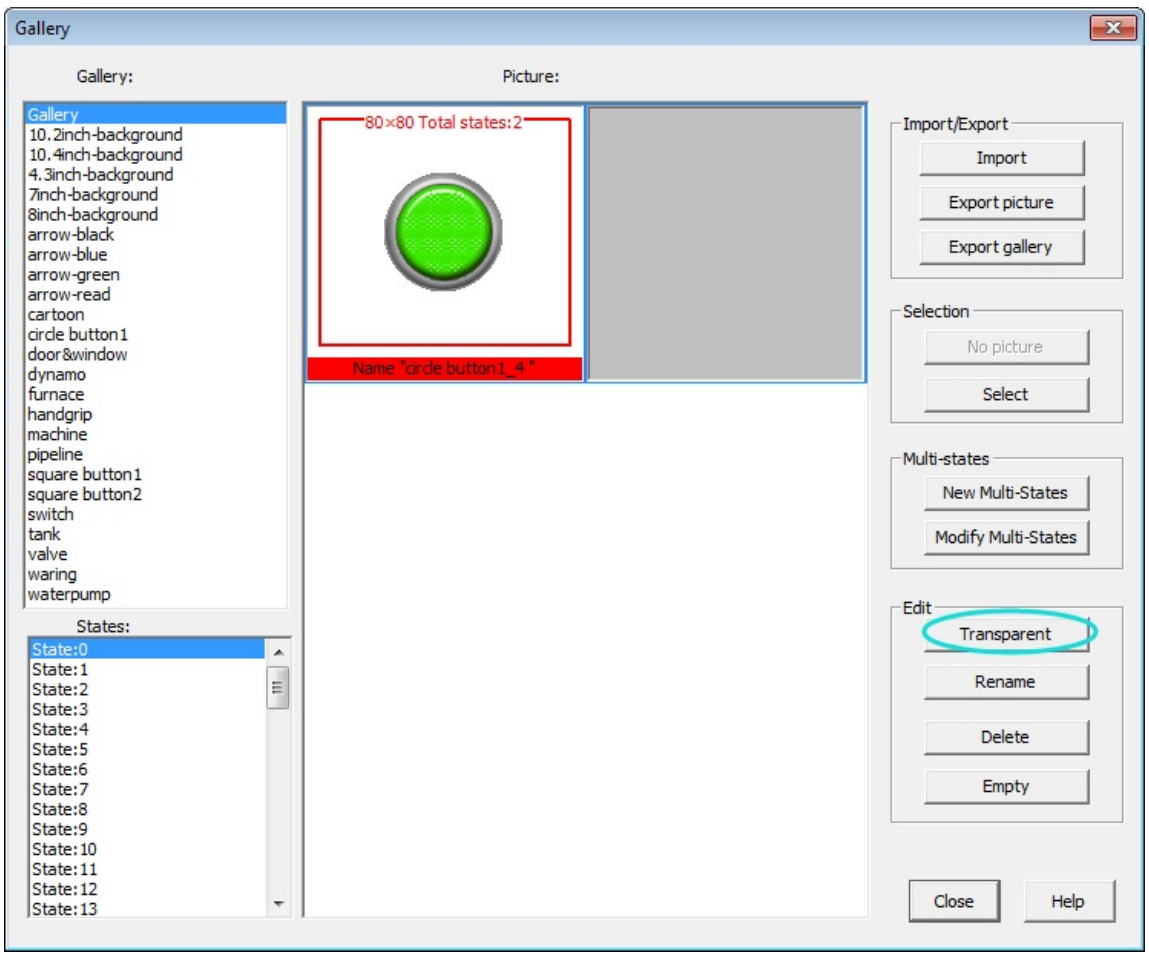


Figure 1

Table

Settings

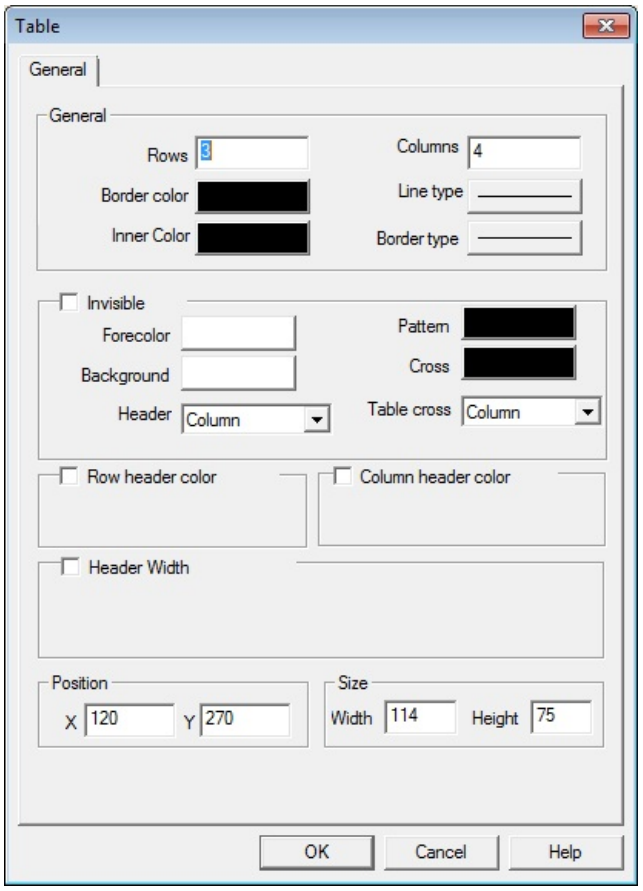
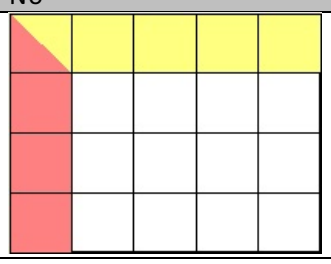
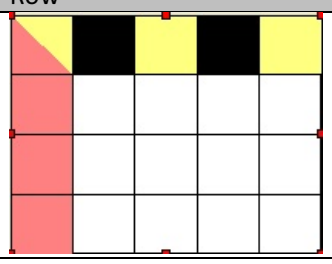
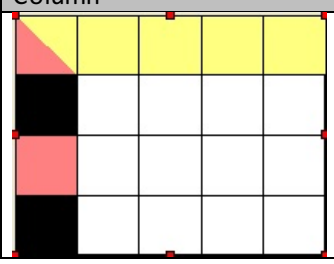
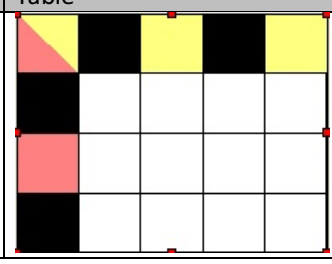
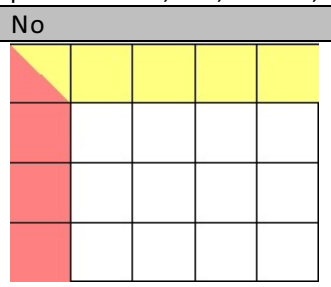
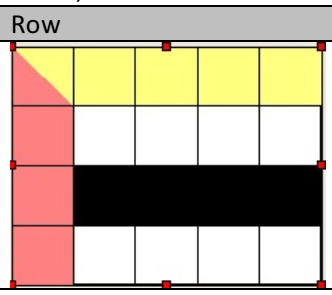
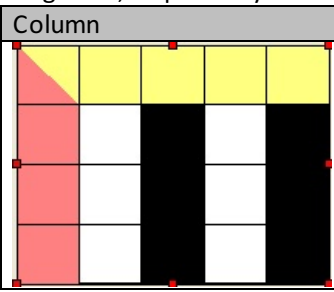
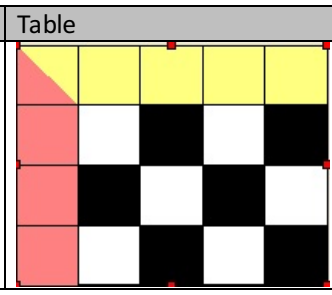


Figure 1

1. Rows:The number of textboxes list horizontally.
2. Columns:The number of textboxes list vertically.
3. Invisible:The table frame becomes invisible, excluding the header.
4. Header cross:The options for header cross are No, row, column, and table, which shown in the following table, respectively.

Header cross	No	Row	Column	Table
Effects				

5. Table cross:The options are No, row, column, and table, which shown in the following table, respectively.

Table cross	No	Row	Column	Table
Effects				

6. Row header color.
7. Column header color.
8. Header width:The header width does not affect the table size.

Vector Graphics

Vector graphics does not support custom picture.

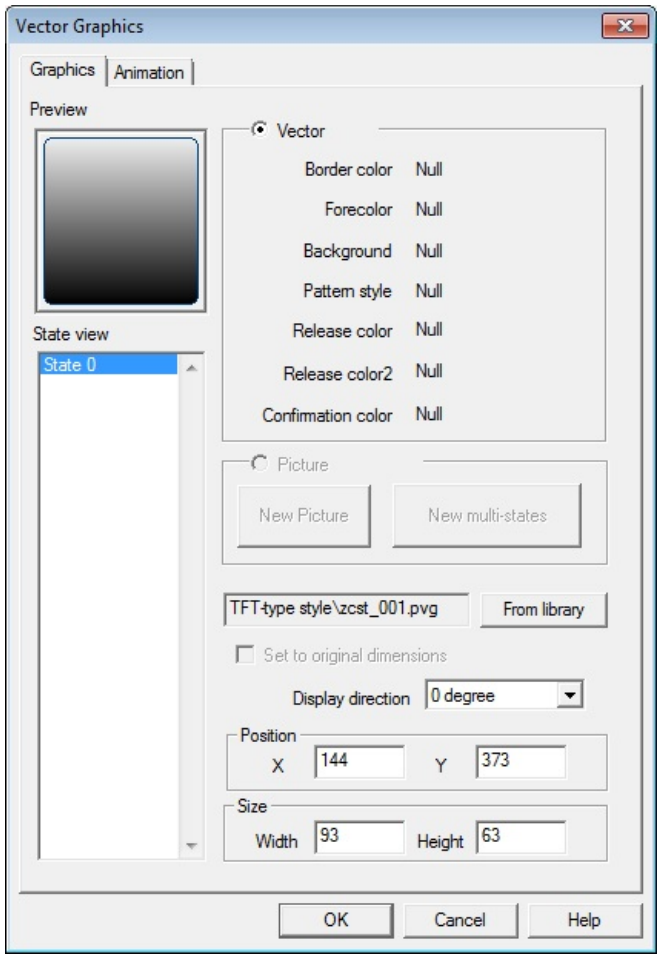


Figure 1

Fast Copy

General

When the same part is required for multiple times, click the part need to be copied, select fast copy on the menu bar, chose the rows and columns required, shown as following.

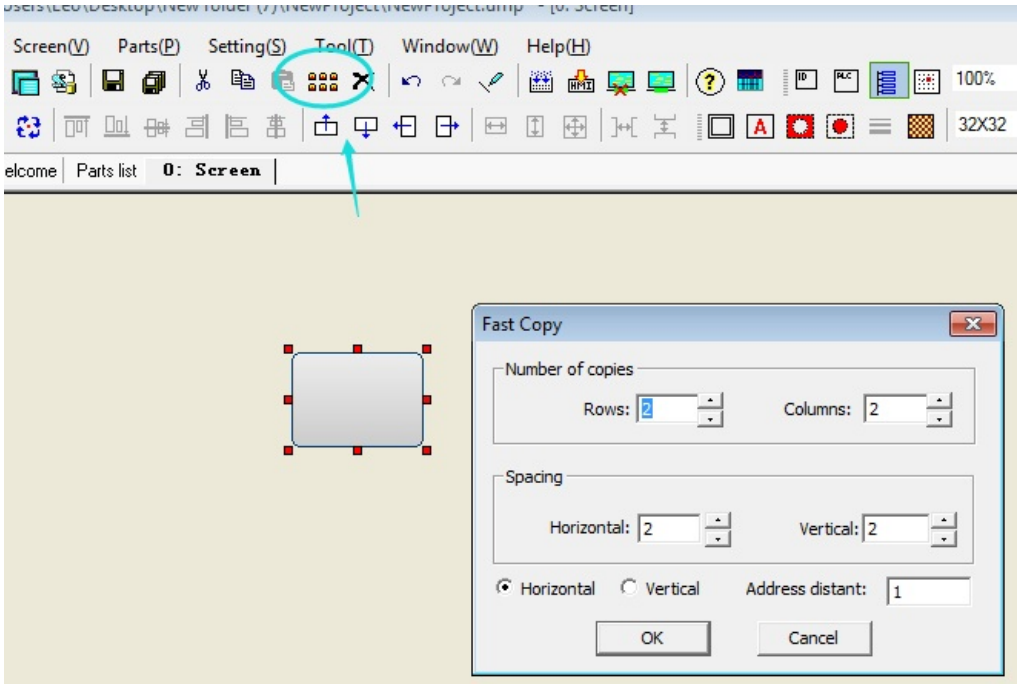


Figure 1

Settings

1. Number of copies:the numbers is expressed by the rows and columns.
2. Spacing:the space expressed as pixels between the rows and columns.
3. Direction:the addresses line up direction.
4. Address distant: the interval of the address number. For example, set the address distant as " 2", the start address is HDX10.0 then the addresses of the copied parts are named as HDX10.2, HDX10.4...

When the fast copy setting is completed the parts will list as following.

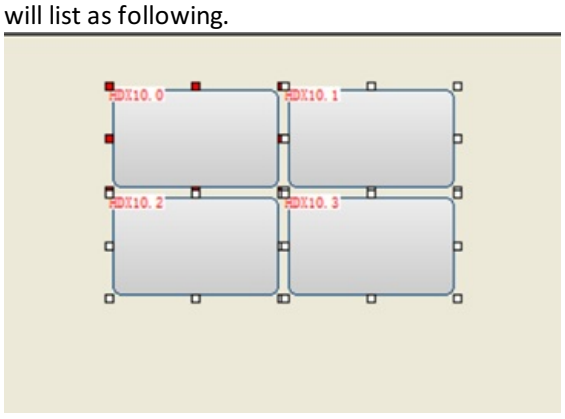


Figure 2

Bit Alarm

Bit alarm message

Add bit alarm to display alarm information on the screen.
Click [Setting]-[Bit Alarm]-[Add].

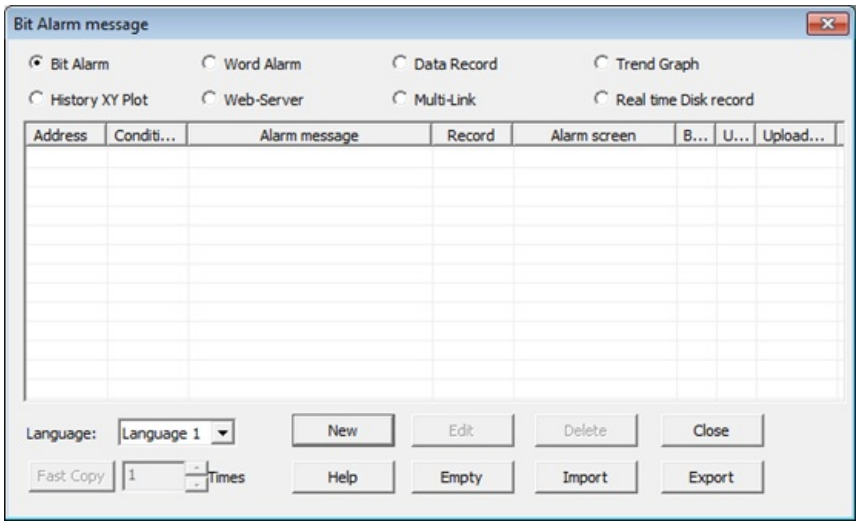


Figure 1

Bit Alarm

Add or edit bit alarm information, set bit address, trigger mode, content, and alarm screen ect.

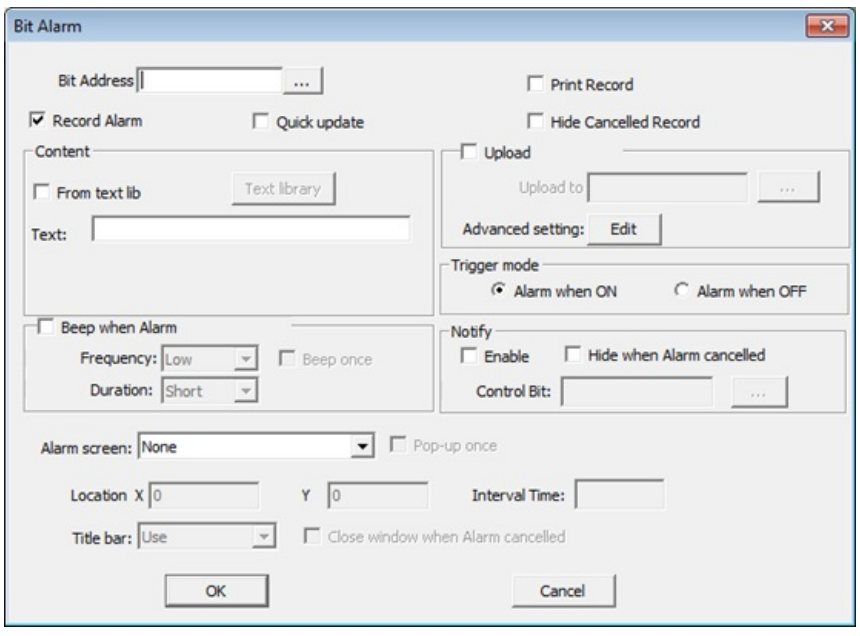


Figure 2

Property	Description
Bit address	Read address.
Record Alarm	Record history data.
Upload	Upload the alarm time and information to designed address or not. Click the ”Edit” beside “Advanced setting” to select the upload information.
Trigger mode	The condition cause alarm.
Content	Alarm message which display on screen.
Beep when alarm	Beep when alarm on.
Notify	Set control bit on when alarm on, clear alert when alarm off.
Alarm screen	Pop-up alarm screen (it must be sub-screen).
Location	The location of the screen alarm display.
Interval time	Reopen the alarm screen when alarm screen closed.
Pop-up once	Pop up alarm screen once.
Close window when Alarm cancelled	Automatically close the alarm screen when alarm off.

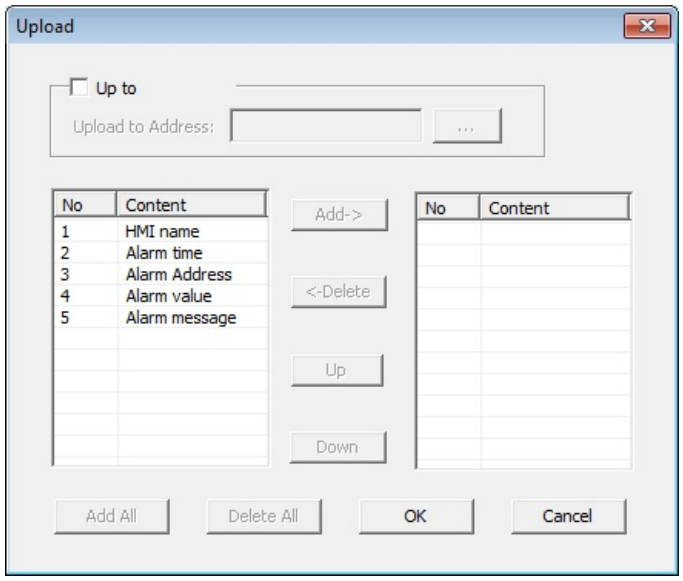

Upload
Up to
Upload to Address:
...
No
Content
1
HMI name
2
Alarm time
3
Alarm Address
4
Alarm value
5
Alarm message
Add->
<-Delete
Up
Down
No
Content
Add All
Delete All
OK
Cancel

Figure 3

Word Alarm

Word Alarm

Monitor the data, alarm is on when designated address meet the condition.

Word Alarm type:

1.High Limit Alarm: Alarm is on when it reaches high limit.

2. Low Limit Alarm: Alarm is on when it reaches low limit.

3. Range Alarm: Alarm is on when it reaches the range.

Click [Setting]-[Word Alarm].

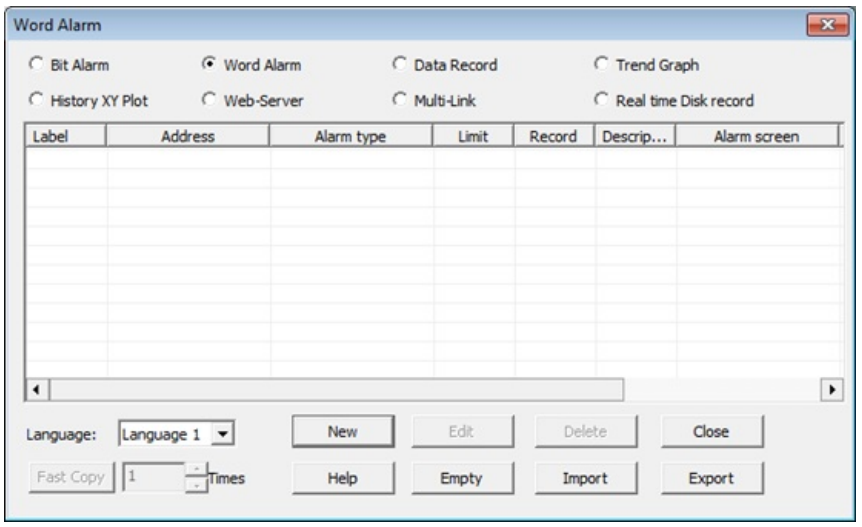


Figure 1

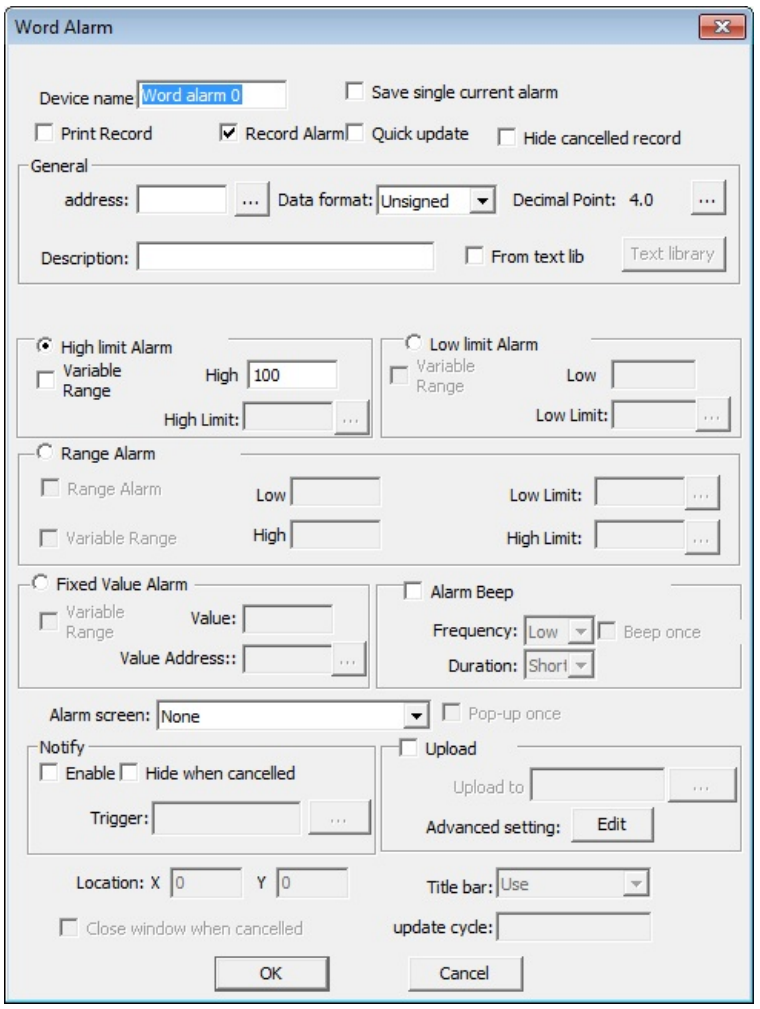
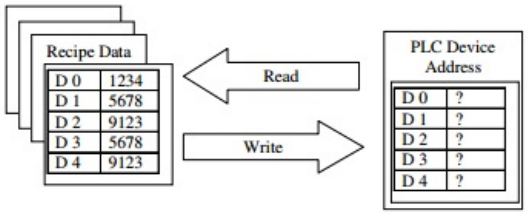


Figure 2

Properties	Description
Device name	Alarm name.
Record Alarm	Record history alarm information.
Device address	Set alarm address.
Data Format	Set data format.
Alarm description	Set alarm information when alarm is on.
High Limit Alarm	Alarm is on when it reaches high limit.
Low Limit Alarm	Alarm is on when it reaches high limit.
Range Alarm	Alarm is on when it is within the range.
Fixed value alarm	Alarm is on when it equals to a constant.

Recipe Function

The Recipe function keeps data in the HMI or other removable storage device, download the data from HMI to designated device addresses, or upload data from device addresses to HMI.



If user has several kinds of parameters and want to switch all or those parameters during production, you need to enter the parameters one by one every time, but if user can use the Recipe function, just need to enter the all parameters in HMI once and just switch the recipes.
For example: A new can needs to be printed with different color and different printing duration. So user can create a recipe to save all of those parameter.

Position	Red(KG)	Green(KG)	Blue (KG)	Duration (Second)
Car Top	2	2	1	30
Car Bottom	3	1	2	40
Car Inside	2	3	3	20

Recipe	Data 1 (Red)	Data 2 (Green)	Data 3 (Blue)	Data 4 (Duration)
Group 1 (Top)	2	2	1	30
Group 2 (Bottom)	3	1	2	40
Group 3 (Inside)	2	3	3	20

Recipe Setting

Recipe can support up to 25,000 data (Maximum total group 50, or Maximum total members 1,000).

Click [Setting]-[Recipe]

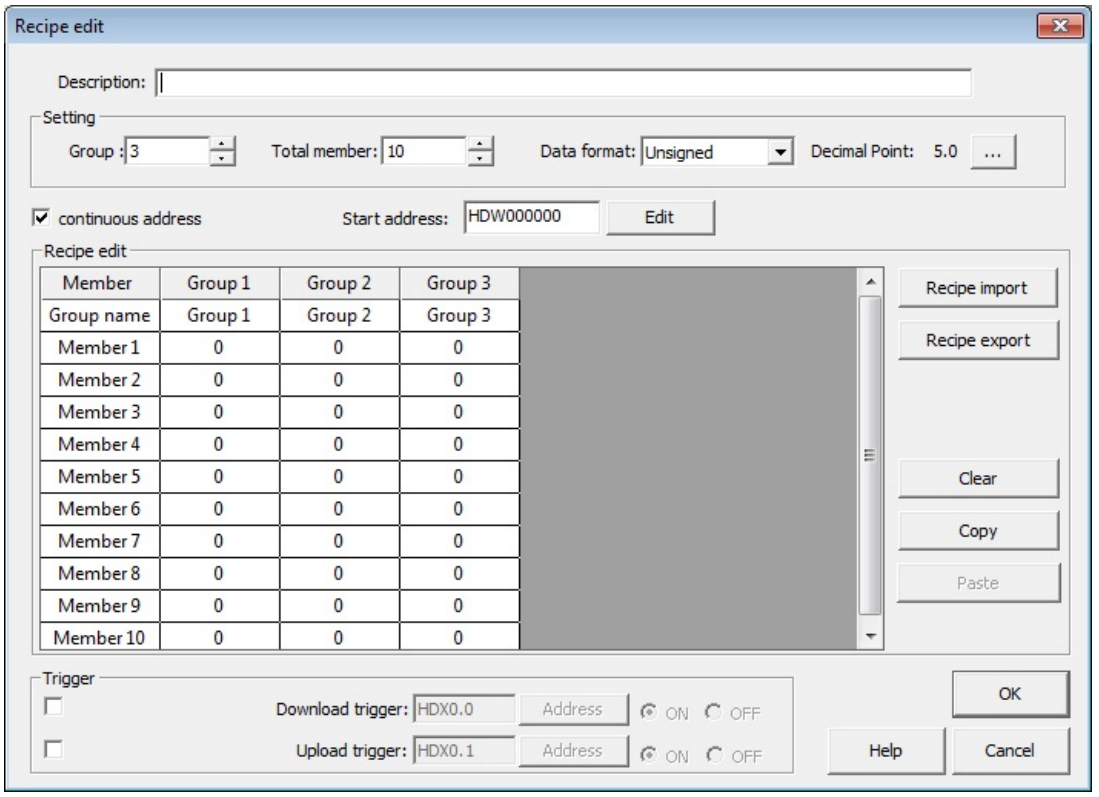


Figure 1

Properties	Description
Setting	Specify the total groups and members in recipe.
Data Format	The format data write to device or read from device addresses.
Continuous address	Use continuous address for recipe, need to specify the start address.
Trigger	Transfer recipe file when meet the control bit condition.

Recipe Display

Click [Objects]-[Recipe]-[Recipe Display].
For example: In the car paint case, it needs 3 groups. 4 numbers in each group:
Group 1: Top;
Group 2: Bottom;
Group3: Inside;

Red: HDW0;
Green: HDW1;
Blue: HDW2;
Duration: HDW3;

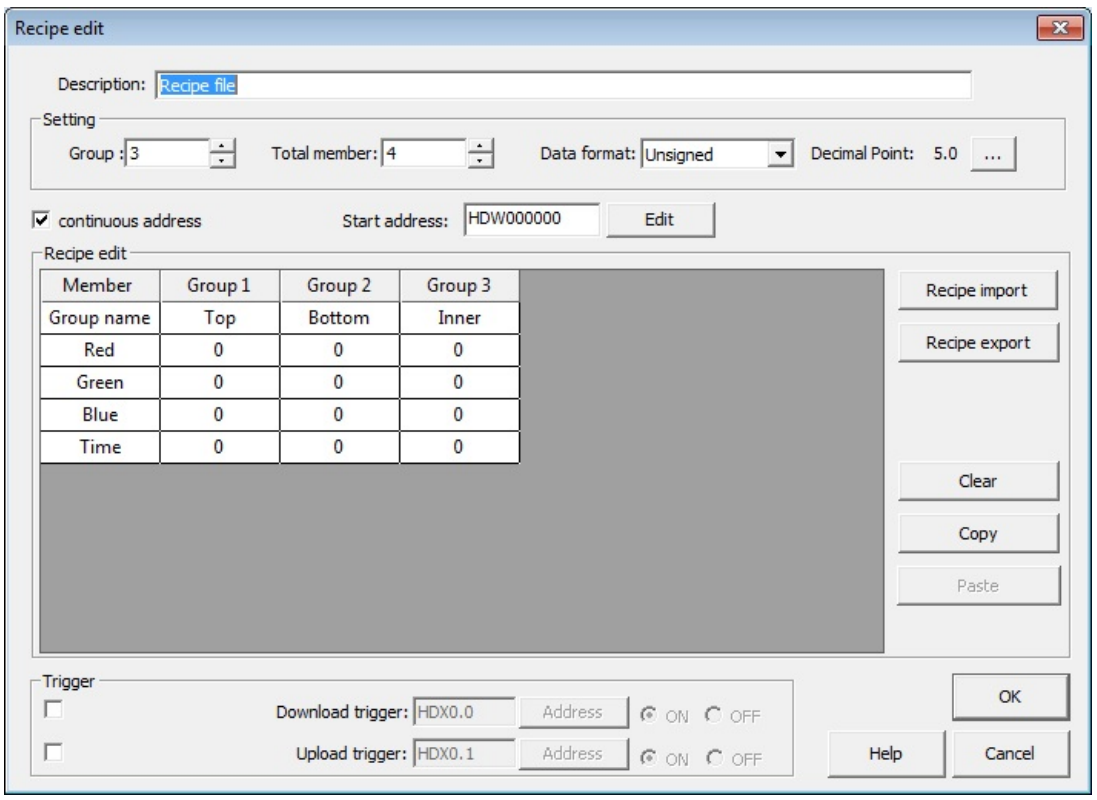


Figure 1

Add “Recipe Display” to the screen by clicking [Objects]-[Recipe]-[Recipe Display].

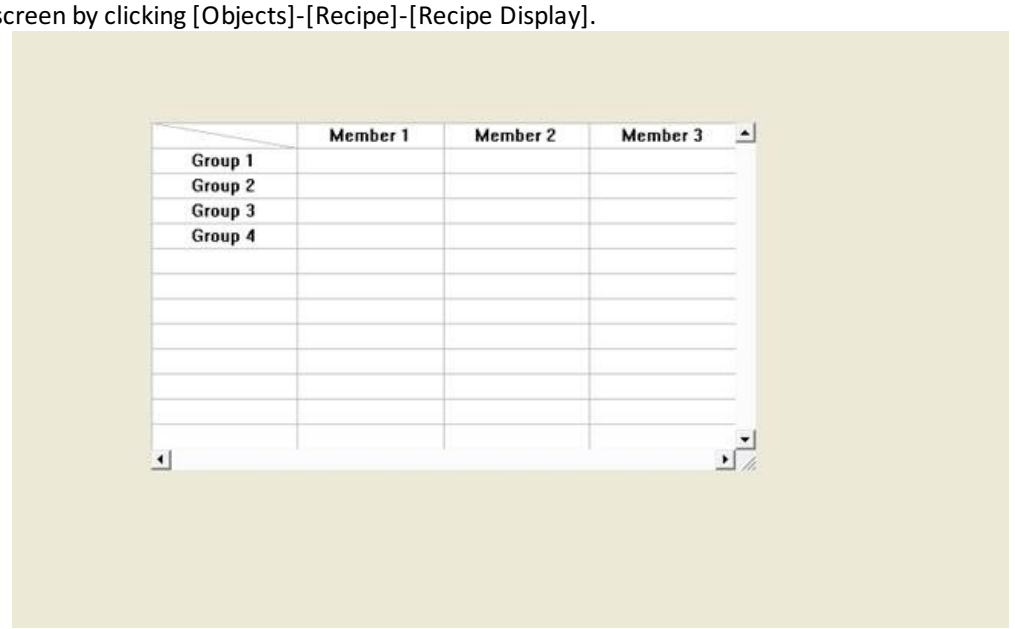


Figure 2

Recipe Transfer

Recipe transfer is designed for reading and writing the recipe address, over write will replace the current address.
Add the “Recipe Transfer button” to the screen by click [Objects]-[Button]-[Recipe Transfer] and set its properties.

Also need to add a “Numeric Input/Display” with address “HPW0”, “HPW” is used for specifying the Group number.

Classification	Properties	Description
Transmission mode	Upload	Transfer the data to the designated recipe.
	Download	Write the recipe to designated address.

Trend Chart

Trend graph is designed to input and save related trend information. It will display the curve in real time.

Click [Setting]-[Trend Display] to add the trend curve.

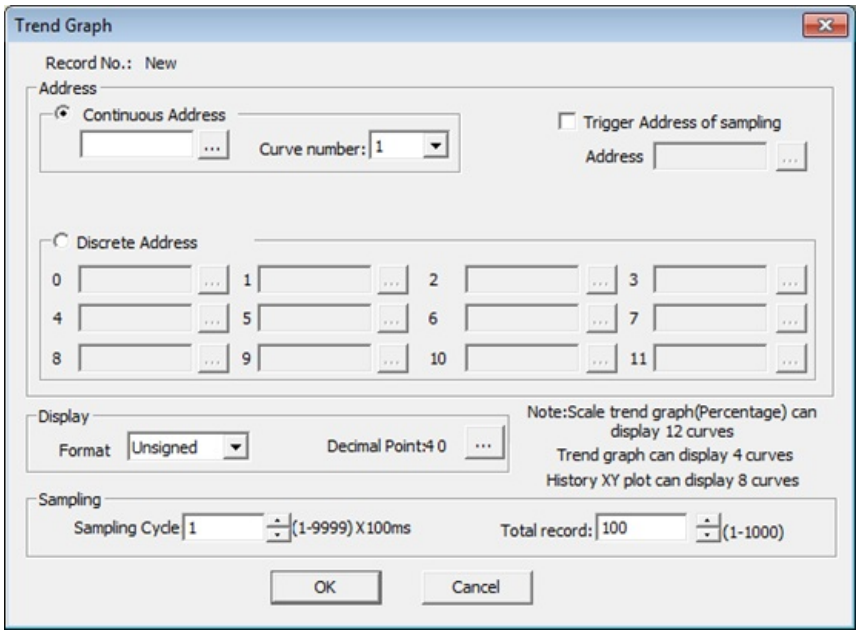


Figure 1

History XY Plot

1. A record can collect up to 12 word address.
2. Use triggers address of sampling.
3. Sampling cycle is 0.1 second.

Click [Setting]-[History XY Plot]-[Add].

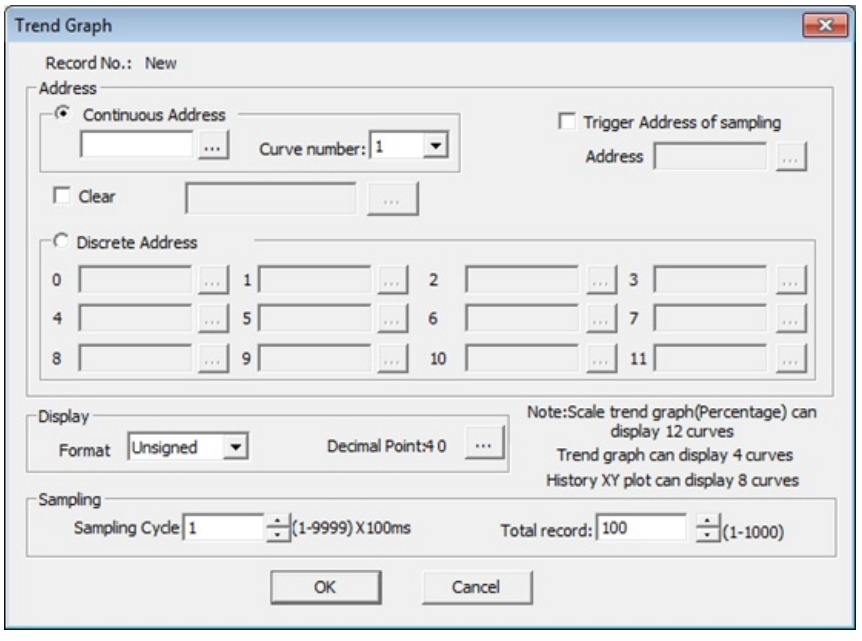


Figure 1

Web-Sever

Web-Server is designed to access HMI from PC Browser, user need to set the related address in this area.
Click [Setting]- [Web-Server]-[Add]:

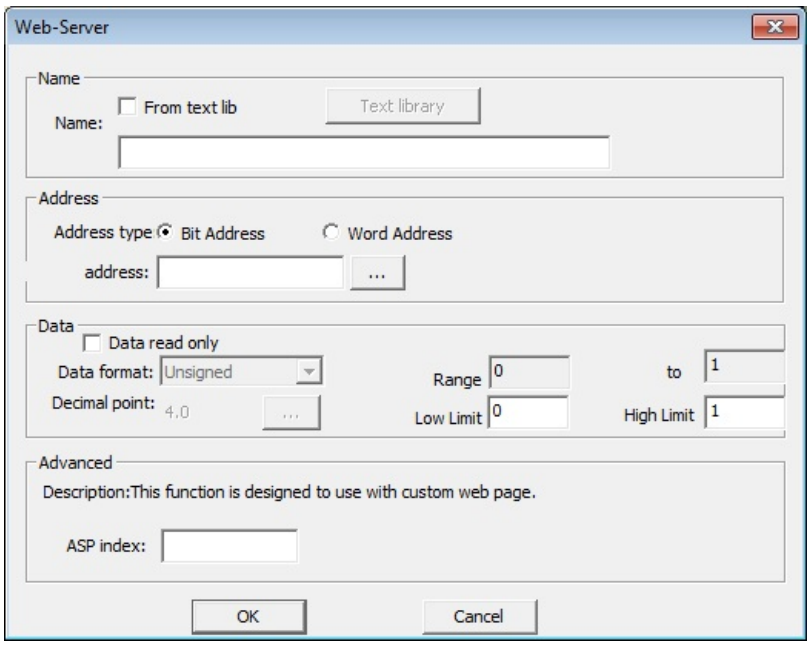


Figure 1

Properties	Description
Name	Name of each information.
Address	The address needs to be related to web ASP Index.
ASP Index	Set ASP Index to link with project address. For example, set the project object address “HDW10”, Set ASP Index” VW0”. Wherever there is “VW0”, it will be replaced with value of “HDW0”.

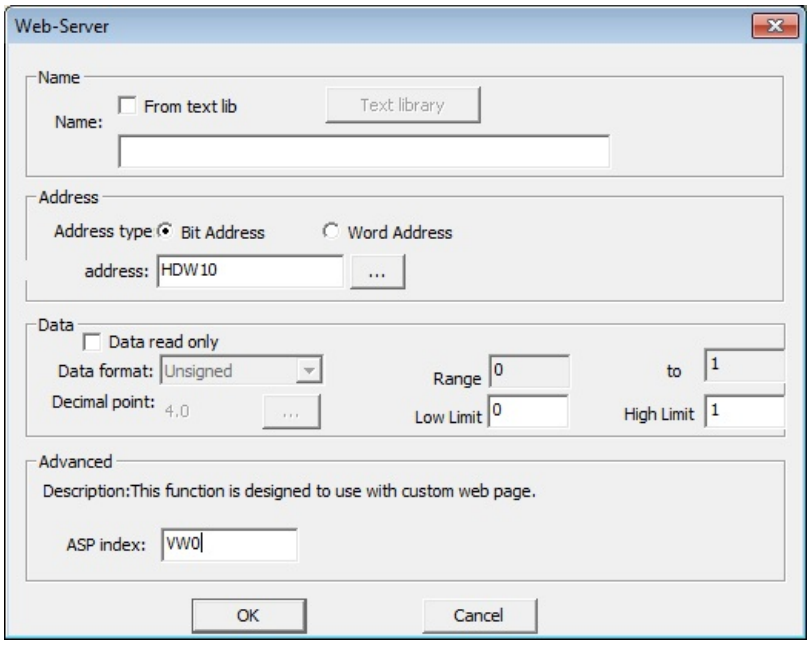


Figure 2

Multi-Link

Multiple HMI can communicate with each other through RS485 connection. one interface is HOST, the other are client. Only HOST HMI communicates with PLC, and it shares the data with other Client HMI.

Connection Type

Note: Every HMI needs to set different station number for communication. It can use RS232 connection if there is only one client.

Setting

Click [Setting]-[Communication]-[Device Type]-[Other Protocol]-[Multi-Link Protocol].

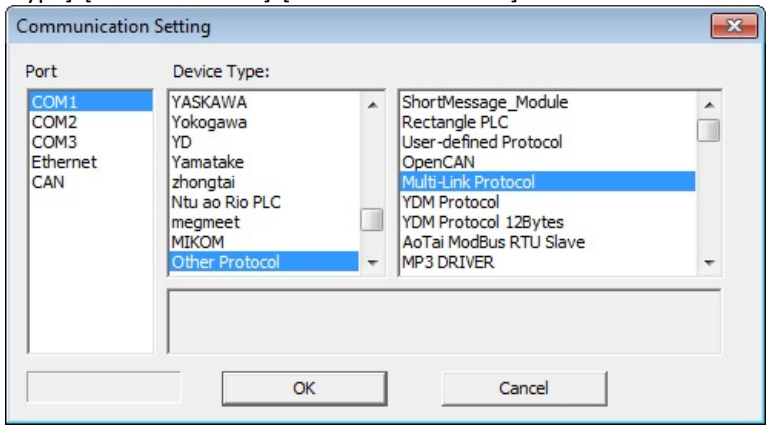


Figure 1

Notice:

Make sure using same address setting for all HMI (HOST and Client).

Need to add all addresses to [Settings]-[Multi-Link] area, if use those addresses in script.

Disc Record Display

Disc Record Display is used to input and save disc record related information. User can read from disc record display and History XY Plot.

1. A record can maximum collect 4 curves.
2. Use Trigger Address of sampling.
3. Sampling Cycle is 0.1 second,"Sampling cycle" means the time of the disc record draw a cycle.

Click [Setting]-[Disc record Display]-[Add]:

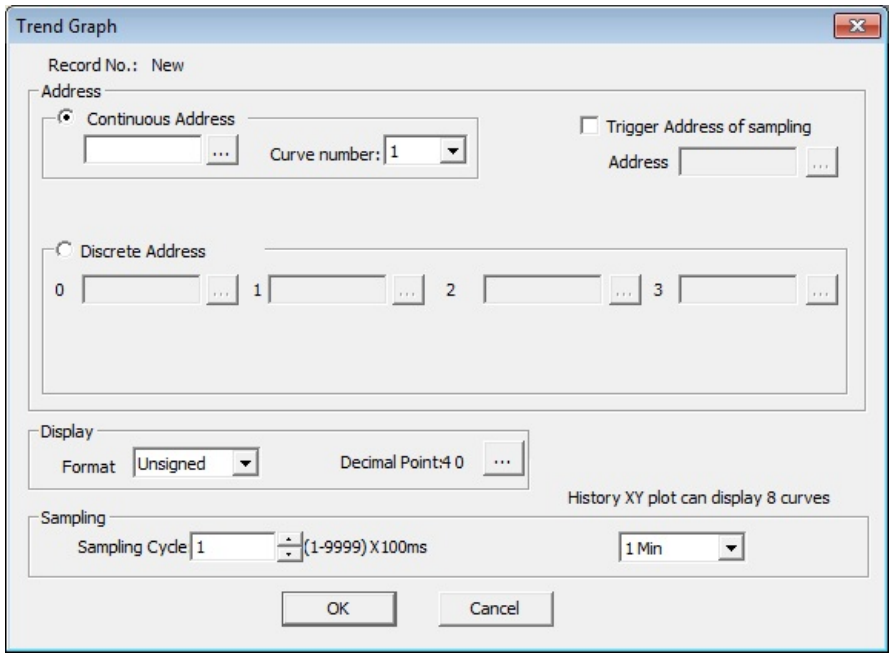


Figure 1

General

Script is applied to realize complex control functions. HMI compile software provide powerful function, simple operation, reliable script system, the features of script are list as follow:

1. Similar to BASIC grammatical structure;

BASIC work as the first computer language for the general public, it is easy and efficient to use.

2. Support all of program logic control structures;

Software script support three logic control structures: order?condition?loops. It can realize complexity procedures.

3. Powerful function;

Functions of script are divided into two types: system and custom function. System function: the functions that system has identified for customers. Custom function: users can define a function and apply to all scripts.

4. Support variety of data format;

Script support integer, floating, BCD code, byte, byte string and etc.

Hints to Use Script

Script can make project more convenient and flexible to use. Script is useful in realizing complex HMI function. If the script is used improperly, it may affect the efficiency of entire project. So pay attention to the follow issues:

1. Try not to use too much script loops, if the script loops that executes too many times, it might influence the efficiency of HMI.

2. In the cycle scripts, try not to use external register, due to the relatively slow serial communication, frequent access to external registers may cause the execution of scripts severely reduced, even influence the screen respond efficiently. There is fine to use internal register.

3. The maximum script length is 512 rows.

Script Access to Device

Software script supports an efficiency way to access the device address by using symbol @.

Writing	Meaning	Examples
@B_;@b_;	Access designated bit address	@B_I0.0:access bit address I0.0 @b_HDX0.0:access bit address HDX0.0
@W_;@w_;	Access designated word address	@W_IW0:access word address IW0 @b_HDW0:access word address HDW0
@B_(the number of protocol connection)#(station number):address	HMI connect more than automatic control devices, "#"stands for choosing number before the symbol," :" stands for accessing the station number before symbol. Access the first protocol without "# ",access default station number1 without":".	@B_2#2:I0.0:access the bit address I0.0, with the connection number 2 and station number 2; @B_I0.0:access to bit address I0.0;
@W_(the number of protocol connection)#station number):address		@W_2#2:IW0:access the bit address IW0, with the connection number 2 and station number 2;

The script can access with the device though: write and read.

For example

```
If @B_HDX0.0 = 1 then        'read the value from address HDX0.0.
@B_HDX0.0 = 0                'write 0 to address HDX0.0
Else
@B_HDX0.0 = 1                'write 1 to address HDX0.0
Endif
@W_QW0 = @W_QW0 + 1    'read data from address QW0, add 1 to this value then write to address QW0.
```

Script Type

Script divided into three categories:

1. Background script: Run independently when start project, screen updates have no influence and valid of all scripts.

2. Global script: It began to run when project start, screen update, switch will affect all of script during running.

3. Screen script: Only run under the designated screen. Screen script start running until screen is closed or switched.

Notice: background: interval unit of running time is 1 millisecond.

Edit background script: double click [background script] on project manager. Then click  to add it, set the timing interval to enter screen script. Shown as following:

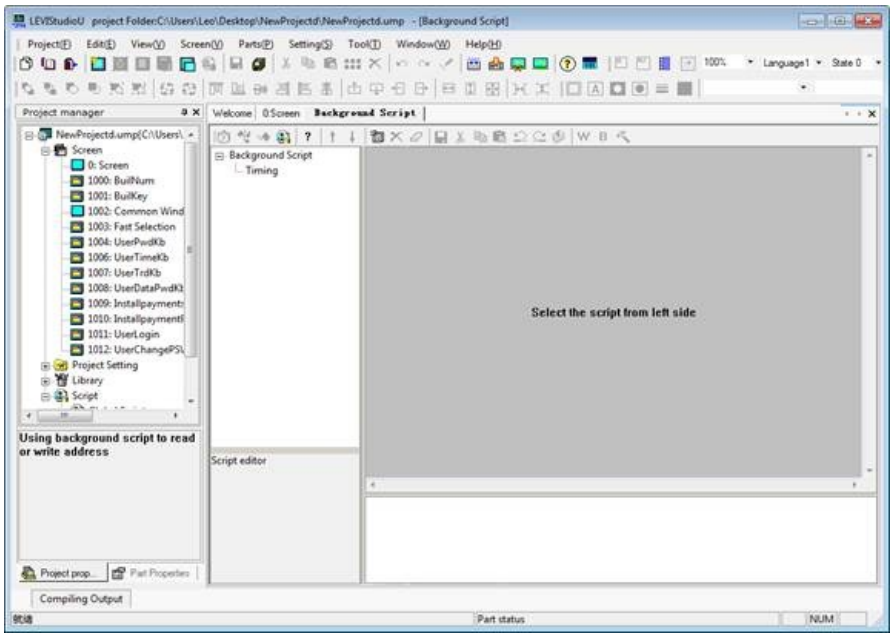


Figure 1

Global script can be divided into four categories:

Property	Description
Initialize	The script will be executed once during loading project.
Close	The script will be run once during closing HMI project.
Timing	Trigger to run once in a while until project end, during running. Each project can be set global script of more than one timing trigger.
Bit trigger	Script will be repeat executed when meet the condition of bit trigger.

Bit trigger has four conditions

Property	Description
TRUE	Detect the value of toggle bit constantly, script will always execute when the value is TRUE.
FALSE	Detect the value of toggle bit constantly, script will always execute when the value is FALSE.
Rising	Detect the value of toggle bit constantly; script will be once executed when the value changes FALSE from TRUE.
Falling	Detect the value of toggle bit constantly, script will be once executed when the value changes TRUE from FALSE.

Notice: global script of bit trigger can also have more than one in one project.

In order to edit global script, double click [global script] from project manager.

Screen script can be divided into four categories under running condition:

Property	Description
Initialize	Run under loading screen, the script will be executed once during running.
Close	Run under closing screen, the script will be executed once during running.
Timing	Trigger to run at every once until screen is closed, during running.
Bit trigger	Script will be repeat executed when meet the condition of bit trigger, under running project.

Bit trigger divided into four conditions.

Property	Description

TRUE	Detect the value of toggle bit constantly, script will always execute when the value is TRUE.
FALSE	Detect the value of toggle bit constantly, script will always execute when the value is FALSE.
Rising	Detect the value of toggle bit constantly; script will be once executed when the value changes FALSE from TRUE.
Falling	Detect the value of toggle bit constantly, script will be once executed when the value changes TRUE from FALSE.

Right click [current screen]-[screen script] to edit screen script.

Global script and screen script have the same categories from above description. The difference between screen script and global script is that screen script runs only during screen operation, and the global script runs only during project operation.

Initialization Script

Summary

Initialization divided into screen initialization script and global initialization script. screen initialization script run once when enter screen at the moment; global initialization script is the moment of loading project, it runs once and is valid of all script.

Setting

Click [global script] from project manager to enter global script editor window.

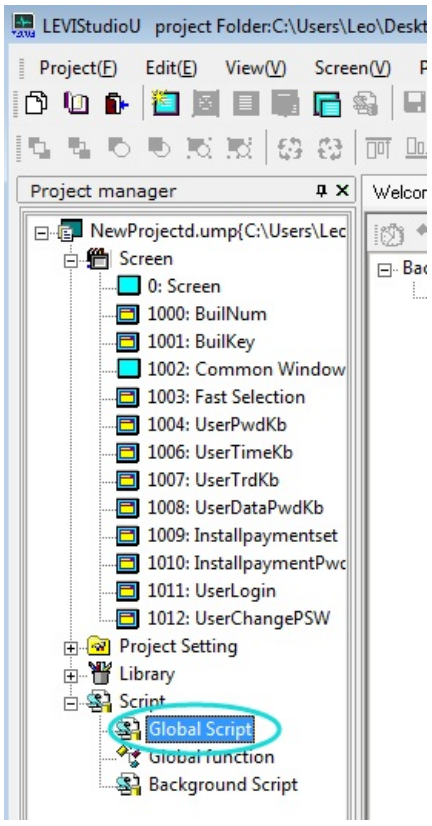


Figure 1

Double click [initialize] node from “the view of script” on screen script or global script, then edit script.

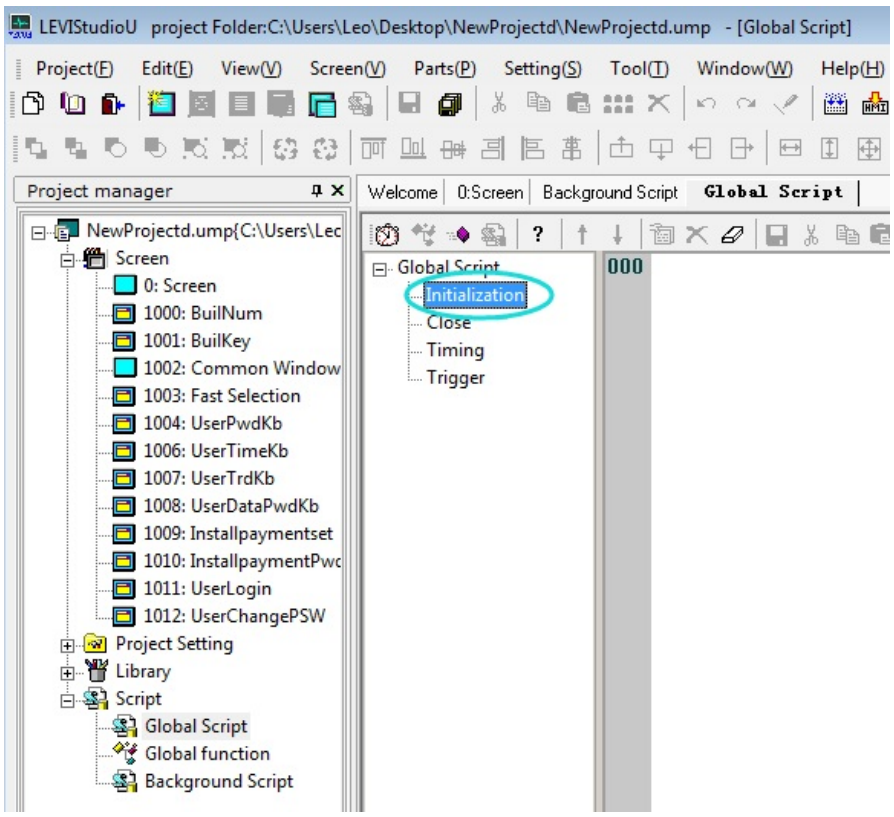


Figure 2

Trigger Control

General

Trigger control script is that software will check whether the designated bit meet trigger condition every 20ms.script execute once when condition is met until project closed.

Notice:The maximum number of trigger script for one screen is 32.

1. The script guidance.
2. New trigger control script.

Click [global script] from project manager to enter global script editor window.

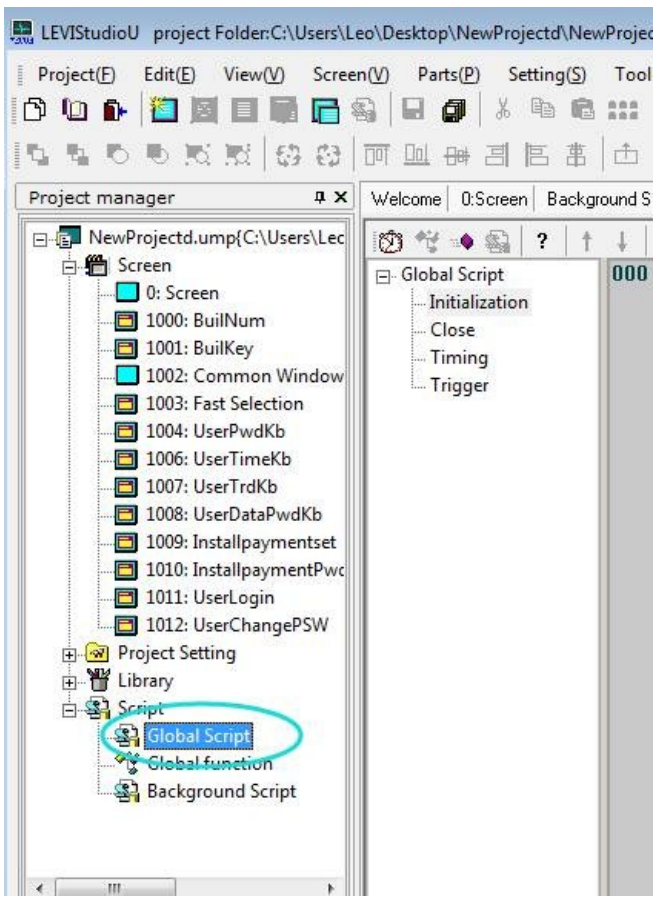


Figure 1

Double click [initialize] node from “the view of script” on screen script or global script, then edit script.

Condition	Description
TRUE	check monitor bit Every 20ms,script execute once when the bit value is TRUE;
FALSE	check monitor bit Every 20ms,script execute once when the bit value is FALSE;
Bit changed	Execute once when bit switch state.
rising	check monitor bit Every 20ms, script execute once when the bit value from FALSE to TRUE;
falling	check monitor bit Every 20ms, script execute once when the bit value from TURE to FALSE;

Delete trigger control script

Select a trigger control script from script view, and then click from tool bar, the script will be not restored after deleting.

Edit trigger control script property

Edit timing script property is to modify trigger condition and monitor bit, select script from script view then click from tool bar, click [OK] to save after modifying.

Close Script

General

Close script divide into screen close and global close. Screen close is that script will run a period when screen closed. Global close is that script run a period when project closed.

Click [global script] on project manager to open global script editor window.

Double click [close] from " script tree view", open script editor window to edit, shown as figure 1.

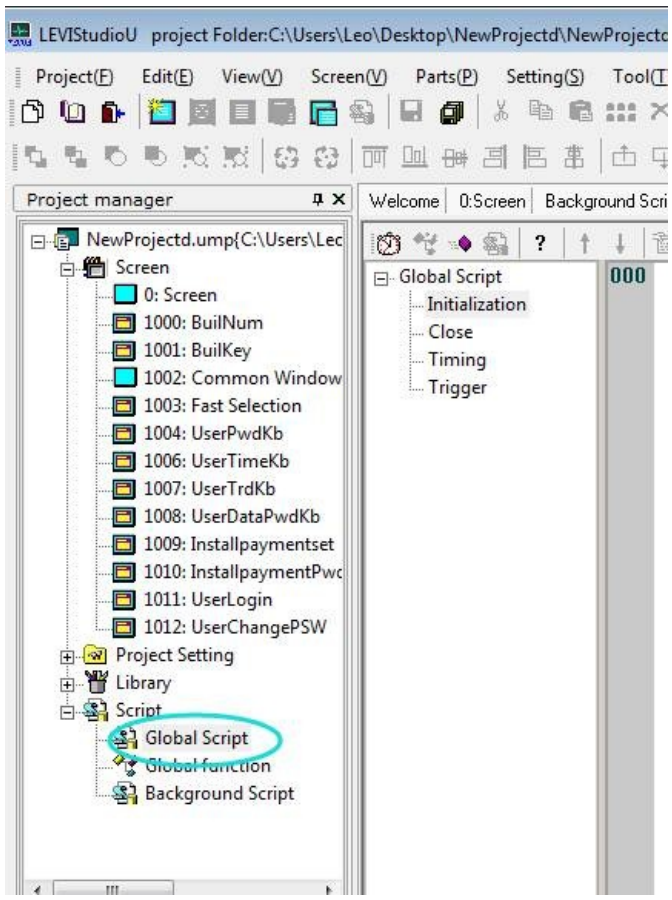


Figure 1

Timing Script

General

The script will run for a designated time interval.

Notice: each trigger script screen of timing and global timing script limit to 32.

New timing script

Click the symbol  from script editor bar, shown as figure1.

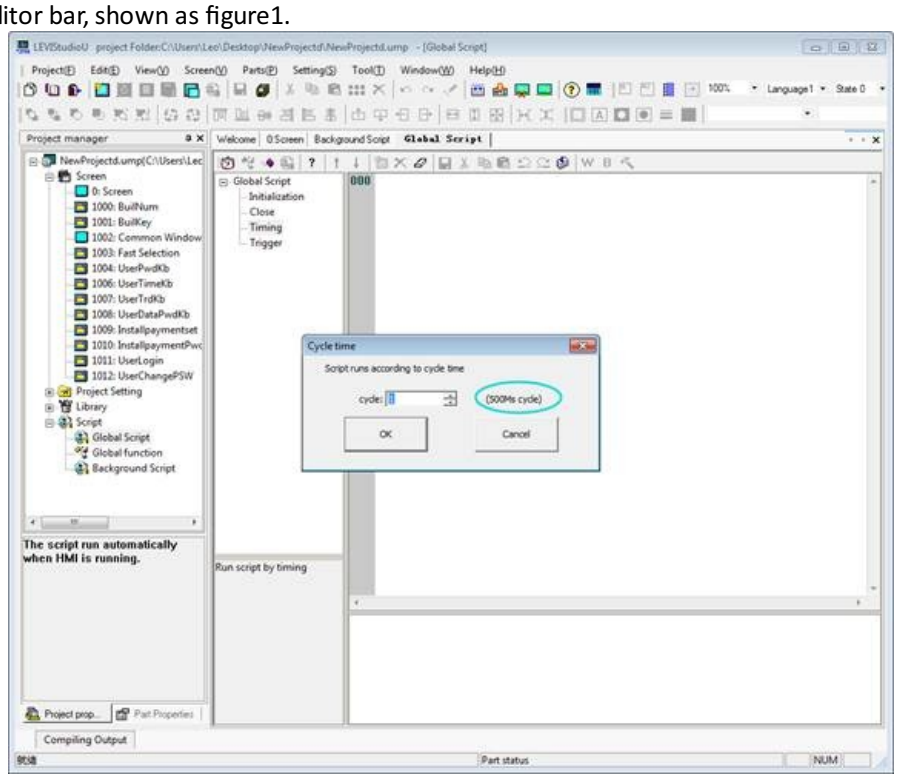


Figure 1

Property	Description
Cycle	Script runs at designated time interval, unit is 500ms. Background timing script at 1ms per cycle.
Ok	Script created.
Cancle	Cancel the current script setting.

List timing node on script view after creating timing.

Edit timing script property

Edit timing property is to modify cycle, select timing from script view then click  from tool bar, and click [OK] to save after modifying cycle, shown as figure 3.

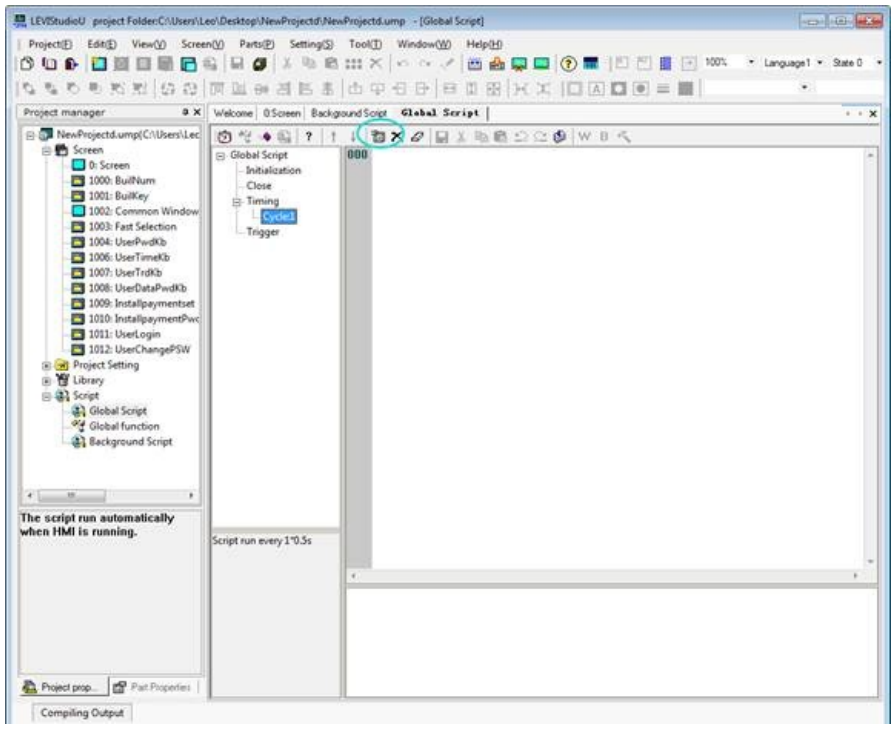


Figure 2

Global Function

General

Global function is a form of code for using,it can be called in any script.the method reference system function.

New global function

Double click [global function] in project manager.

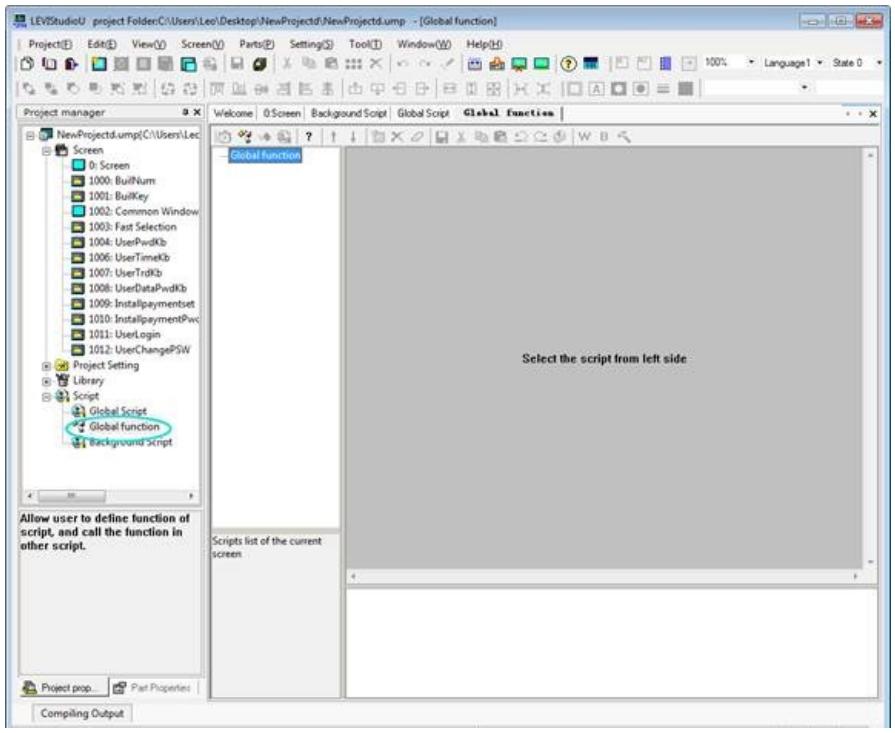


Figure 1

Edit global function parameter

Click from script tool bar, it is used for adding a global function, shown as figure 2.

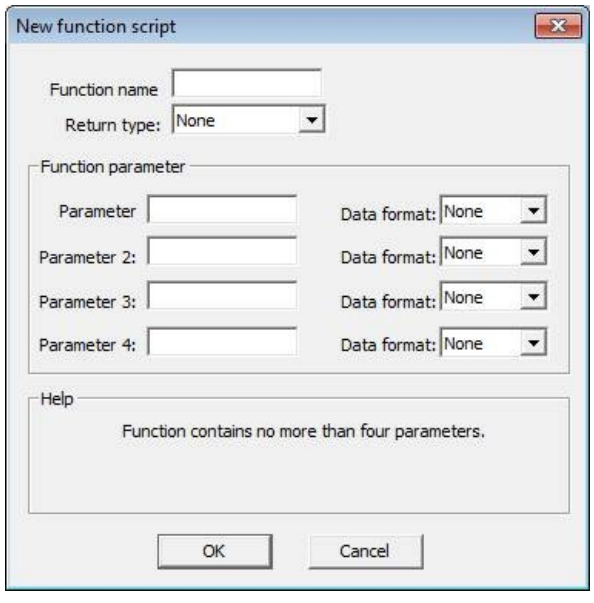


Figure 2

Property	Description
Function name	Function name cannot be the same as existing.
Return type	None, string, integer, float.
Paremeter1	The name of paremeter 1.

Notice: function contains no more than are four parameters and cannot be repeated.

Edit global function properties

[Edit] is to modify name and parameter. Click on tool bar.

Call global function

Refer to the call of system function.

Checking Grammar

Check script grammar is correct or not, error occurs when compile error.
Select  from script tool bar, system does not prompt grammar error if grammar is correct, system will list all errors for modification.

Grammar Error

In this section common grammar errors of software script are listed as following:

1. Identifier *** contains invalid characters.
2. Attempt to redeclare sub *** .
3. Attempt to redeclare function *** .
4. Attempt to use reserved word *** as identifier.
5. Attempt to use type *** as identifier.
6. Unexpected ‘)’ while parsing arguments for function *** .
7. Cannot parse expression (one of the arguments of function ***) .
8. Cannot parse arguments of *** .
9. Too many arguments for function *** .
10. Not enough arguments for function *** .
11. ‘(‘ expected after sub name *** .
12. Unexpected ’(‘ while parsing arguments for sub *** .
13. Cannot parse expression (one of the arguments of sub ***) .
14. Cannot parse arguments of *** .
15. Too many arguments for sub/function *** .
16. Not enough arguments for sub/function *** .
17. Cannot parse expression .
18. ‘(‘ expected after function name *** .
19. Unexpected use of sub *** as a part of expression .
20. Illegal statements preceding subs/functions declaration .
21. Unexpected end of file while looking for ‘endsub’ .
22. End of line expected after ‘else’.
23. End of line expected after ‘endif’.
24. End of line expected after ’next’.
25. End of line expected after ‘wend’.
26. ‘while’, ‘until’ or end of line expected afte r’do’.
27. Cannot parse expression after ‘while’.
28. Cannot parse expression after ‘until’.
29. ‘do’ without ‘loop’.
30. Sub *** contains invalid character’@’ .
31. Sub *** already declared.
32. Function *** already declared.
33. Sub name expected after ‘sub’.
34. Function name expected afte r’function’.
35. Variable name expected.
36. Argument *** contains invalid character ‘@’.
37. ‘integer’, ‘floating’ or ’string’ expected.
38. ‘‘,’ or ‘)’ expected.
39. ‘endsub’ without ’sub’.
40. ‘end function’ without ‘function’.
41. End of line expected after ’beep’.
42. ‘dim’ unexpected here.
43. Variable name expected after ’dim’.
44. ‘as’ expected after variable name.
45. ‘integer’ ‘floating’ or ‘string’ expected after ’as’.
46. ‘,’ or end of line expected after type in dim statement.

47. Cannot parse expression after ’while’.
48. End of line expected after’ while’ condition.
49. ‘while’ without ‘wend’.
50. End of line expected after ’wend’.
51. ‘wend’ without ‘while’.
52. Variable name expected after’ for’.
53. ‘=‘ expected after variable name.
54. Cannot parse expression after ’for’.
55. ‘to’ expected.
56. Cannot parse expression after ’to’.
57. Cannot parse expression after ’step’.
58. End of line expected.
59. ‘for’ without ‘next’.
60. End of line expected after ‘next’.
61. ‘Next’ without ’for’.
62. Cannot parse expression after’if’.
63. ‘then’ expected.
64. Unexpected end of file while looking for ’endif’.
65. Unexpected end of file while looking for ’else’ or ’endif’.
66. ‘else’ without ’if’.
67. ‘end if’ without ’if’.
68. Label name expected after ’goto’.
69. Unexpected end of line while looking for ‘)’ in function call.
70. ‘,’ expected.
71. Missing ‘)’.
72. Unexpected end of line in expression.
73. Unexpected end of file in expression.

A2H

Function
Val = A2H(A1)

Description
Convert string A1 to hexadecimal number.

Parameters
A1: convert first four value of the string.
Notice: string must be address(such as:@W_HDW000002).

Val: The value is hexadecimal number.

Example
@W_HDW20=A2H(@W_HDW10) ' convert the string of HDW10 to hex then save in HDW20.

Input: @W_HDW10=255
Result: @W_HDW20=255

Abs

Function
val = Abs(A1)

Description
The absolute value of A1.

Parameters
A1:the data of absolute value, must be variable.
Val:it is absolute value that is address or variable.

Example
```
Dim a as integer        'a is defined as integer
a = SignedInt16("@W_HDW0")'convert the data of @W_HDW0 into signed data.
@W_HDW1 = Abs(a)        'assign the returned absolute value to @W_HDW1
```

Input:@W_HDW0=-6,
Result:@W_HDW1=6
Notice: SignedInt16 function is designed to convert unsigned to signed.

ACOS

Funcition
val = ACos(A1)

Description
To compute the inverse cosine value of A1.

Parameters
A1: float value, can be a address or variable.
Val: return value is float, can be a address or variable.

Example
```
Dim a ,b as floating      'define two float variable a,b
a = 0.5             'assign the designated value to a
b = ACos(a)          'the inverse cosine value of “a” is a radian which assign to variable b.
'to add the following sentence if needed to view the return value:
float2d("@W_HDW200", b)   'float b written into HDW200.
```

Result: @W_HDW200=1.047
Notice: please call RadToDeg function to convert radian into angle.

AddrStringCompare

Function
val = AddrStringCompare(A1, A2, length)

Description
It is designed to compare the designated length of two character strings.the string value is 1 when the two strings are the same.

Parameters
A1, A2:character string, must be an address(such as:"@W_HDW000002")
Length:The length of character string.
Val:Return value, 0 or 1.

Example
```
if AddrStringCompare("@W_HDW10","@W_HDW0",2)=1 then 'compare the character string of HDW10 and @W_HDW0 whether value are 1.
@W_HDW20=1 '@W_HDW20 display 1
else
@W_HDW20=0 '@W_HDW20 display 0
Endif
```

Input: @W_HDW10="1a2 ",@W_HDW0="1a2 ",
result:@W_HDW20=1

Input: @W_HDW10="ab2 ",@W_HDW0="12a ",
result:@W_HDW20=0

Asc

Function
val = Asc(A1)

Description
Return the first character of the string in ASCII value.

Parameters
A1:character string, it can be an address(such as:@W_HDW000002)
val:return value,ASCII value,it can be an address or variable.

Example
```
@W_HDW10 = Asc("A")    'return the ASCII value of A to HDW10
@W_HDW11 = Asc("a")    'return the ASCII value of a to HDW11
@W_HDW12 = Asc("Apple")'return the first character A of string Apple to HDW12
@W_HDW13 = Asc("123")  'return the first character ASCII value 1 of string 123 to HDW13.
```

Result: @W_HDW10 = 65
@W_HDW11 = 97
@W_HDW12 = 65
@W_HDW13 = 49

AsFloating

Function
val = AsFloating(A1)

Description
Convert parameter A1 to a float.

Parameters
A1:integer variable.
val:return float value, can be a variable or address.

Example
```
Dim a as integer    'define a integer variable {a}.
a = @W_HDW10        'assign @W_HDW10 to a
b = AsFloating(a)  'convert integer a to float then assign to b.
b = b/1.2           'add as following sentence when need to view the return value:
Float 2D("@W_HDW11",b)'float variable b written into HDW11.
```

Input:@W_HDW10=24,
result:@W_HDW11=20.00(set two decimals)

ASin

Function
val = ASin(A1)

Description
Calculate the arcsine value of A1.

Parameters
A1:Float can be a address or variable.
Val:Return float value, can be a address or variable.

Example
```
Dim a, b as floating   'define two float variable a,b
a = 0.5              'assign the designated value to a
b=ASin(a)            'calculate the arcsine value of a ,assign the radian to b.
'Add the following command if need to view the return value:
float2d ("@W_HDW200", b) 'float variable b written into HDW200
```

Result:@W_HDW200=0.524
Notice: please call RadToDeg function to convert radian into angle.

AsInteger

Function

val = AsInteger(A1)

Description

Convert parameter A1 to integer value.
Parameters

A1:floating must be a variable.
Val:return integer value can be a variable or address.

Example

```
Dim a as floating       'define floating variable a
a = D2Float("@W_HDW0",a)      'use D2Float function to save the float date of HDW0 in a
b= AsInteger(a)        'convert the float a into integer,the return value assigned to b
@W_HDW10=b             'save b to HDW10
```

Input: @W_HDW0=20.12,
Result: @W_HDW10=20

AsString

Function
val = AsString(A1)

Description
Convert parameter A1 to a character string.

Parameters
A1:not string parameter,it can be a variable.
Val:return string value, variable or address.

Example

Script 1:
```
a=123              'assign a value to a
b=234              'assign a value to b
c=AsString(a)+AsString(b)      'convert a and b to string then add up the two strings ,assign the result to c.
@W_HDW0=c          'assign c to HDW0
d=a+b              'plus a with b
@W_HDW100=d        'assign d to (HDW100)
```

result:@W_HDW0=123234
@W_HDW100=357

Script 2:
```
W2S("@W_HDW200","@W_HDW300","02d")
W2S("@W_HDW210","@W_HDW400","02d")
W2S("@W_HDW220","@W_HDW500","02d")
@W_HDW0=AsString(@W_HDW300)+AsString(@W_HDW400)+AsString(@W_HDW500)
```

Input:@W_HDW200=12,@W_HDW210=34,@W_HDW220=56
Result:@W_HDW300=12,@W_HDW400=34,@W_HDW500=56,@W_HDW0=123456
(Ensure the data always is two bits; otherwise occur error. reference the other chapter of W2S function)

ATan

Function
var = ATan(A1)

Description
Return a arctangent value,the radian ranges -pi/2 to pi/2.

Parameters
A1:Can be float, address or variable.
Val:radian of return value.

Example
@W_HDW20= Atan (@W_HDW10) 'save the arctangent value of (HDW10) to (HDW20)

Input: @W_HDW10=1.000,@W_HDW20=0.785
Notice: please call RadToDeg function convert radian to angle.

ATan2

Function
val = ATan2(A1,A2)

Description
Return the arctangent value of A1/A2,radian range

Parameters
A1, A2: Address or variable.
Val: return value is a radian, range -pi to pi.
Notice: ATan2 use sign of two parameters to define the quadrant of return value.

Example
```
@W_HDW20= ATan2 (@W_HDW10,@W_HDW12)  'save the arctangent value of (HDW10/HDW12) to (HDW20).
```

Input:@W_HDW10=1.0,@W_HDW12=1.0,
Result: @W_HDW20=0.785
Notice: please call RadToDeg function convert radian to angle.

B2W

Function
B2W(A1, A2,length)

Description
Convert a array (begins with A2, unit: byte, to another array (begins with A1,unit:word).

Parameters
A1:Saving address after converting
Notice:must be address(e.g.:@W_HDW000002)

A2:Address of the value be converted
Notice:must be address(e.g.:@W_HDW000002)

Length:The length of conversion
Notice:It can be address or variable.

Return value:None
Notice:This is a subprogram, it has no return value.

Example
```
B2W(@W_HDW100,@W_HDW10,2)   'convert (@W_HDW10) to the length of 2,save as the result that begins with @W_HDW100.
```

Input: @W_HDW10=1A2B
Result: @W_HDW100=2B
@W_HDW101=1A

BCD

Function
val = BCD(A1)

Description
Convert A1(binary) to BCD, save the result as return value.

Parameters
A1:The binary be converted, it can be a address or variable.
Val:Return value, BCD code; it can be a address or variable.
Notice:Return value is a word; it hexadecimal corresponds to BCD code.

Example
```
@W_HDW20=BCD(@W_HDW10)   'convert HDW10 (binary) to BCD code, then save in (HDW20)
```

Input: @W_HDW10=11111111(binary),
Result: @W_HDW20=255

Beep

Function
Beep

Description
Buzzers send out beep.

Parameters
None

Example
```
if @B_HDX100.0=1 then  'beep when the bit switch HDX100.0 set ON
beep
endif
```

result: HMI beep when bit switch HDX100.0 set ON.

BIN

Function
Val = BIN(A1)

Description
Convert A1 (BCD) into binary, save the result in return value.

Parameters
A1: The BCD code is converted; it can be address or variable.
Val: Return binary value,it can be address or variable.

Example
```
@W_HDW20=BIN(@W_HDW10)  'convert HDW10(BCD) to binary, save the result in (HDW20)
```

Input: @W_HDW10=255
Result: @W_HDW20=11111111 (binary)

BMOV sub function

Function
BMOV(A1, A2,length)

Description
Copy data with a designated length from source address A2 to A1.

Parameter
destAddr: saving address
srcAddr: source address
length: data length

Example
```
@W_HDW20 = 20          'assign value to HDW20
@W_HDW21 = 21          'assign value to HDW21
@W_HDW22 = 22          'assign value to HDW22
BMOV(@W_HDW10,@W_HDW20,3) 'assign the word address of HDW20, HDW21, HDW22 to HDW10, HDW11, HDW12
```

Result:
@W_HDW10 = 20
@W_HDW11 = 21
@W_HDW12 = 22

Chr

Function
val = Chr(A1, A2, ...)

Description
Convert integer parameter into correspond ASCII character,return the character string.

Parameters
A1, A2....: converted integer; it can be a address or variable.
Val:return value, can be a address or variable.

Example
```
@W_HDW100=Chr(@W_HDW20,@W_HDW21,@W_HDW22,@W_HDW23,@W_HDW24)
'convert the value of(HDW20,HDW21,HDW22,HDW23,HDW24) to ASCII character,assign the value to (HDW100)
```

Input 72,69,76,76,79 step by step according to HDW20,HDW21,HDW22,HDW23,HDW24,result returns HELLO to (@W_HDW100).

ClrB subfunction

Function
ClrB(A1)

Description
Set the bit of A1 as FALSE(0).

Parameters
A1:System address(bit)
Notice; subprogram has no return value.

Example
```
ClrB(@B_HDX100.0)  'assign 0 to(@B_HDX100.0)
```

Constant

Description

Script supportd stable constant, users can use on script:
pi = 3.14159265358979321
TRUE = 1
FALSE = 0

Example

```
Dim a as integer ‘define integer a
a = RadToDeg(pi)  'convert radian pi to angle then assign to a,RadToDeg function is used to convert radian to angle.
@W_HDW11 = a     'assign a to (HDW11)
```

Result: @W_HDW11=180

Cos

Function
Val = Cos(A1)

Description
Return a cosine value of an angle.

Parameters
A1:a float radian of angle, it can be an address or variable.
Val:return float value, it can be a address or variable.

Example
```
Dim a, b as floating   'define float a, b
b = pi/3               'convert the value of HDW11 to float and assign to ‘b’.
a=Cos(b)          'return the cosine value of ‘b’ and assign the result to ‘a’.
'to add the following sentence if need to view the return value:
Float2D("@W_HDW20",a)  'the float value of a written into HDW13.
```

Result:@W_HDW20=0.5
Notice: please call RadToDeg function convert radian to angle.

D2F subfunction

Function
D2F (A1, A2) or A1= D2F (A1, A2)

Description
Convert the 32 bit integer format data to float then output the result.

Parameters
A1: required data, begin with“@”;
A2: source data, begin with“@”;

Example
```
D2F(@W_HDW2,@W_HDW0)            'convert the double word (HDW0) to float, save the result to (HDW2).
@W_HDW2=D2F(@W_HDW2,@W_HDW0)   'convert the double word (HDW0) to float, save the result to (HDW2).
```

Result: HDW0=100, HDW2=100

D2Float

Function
F= D2Float("A1",F)

Description
Convert the designated value to floating then assign to variable.

Parameters
A1: Must begin with address“@”;
F: Define the floating by self;

Example
```
dim F as floating            'define F as floating
F=D2Float("@W_HDW10",F)    'assign the value of (HDW10) to F in floating
Float2D("@W_HDW12",F)      'copy the floating value of F to HDW12 register,use to display result.
```

Result:HDW10=200,
HDW12=200.

DegToRad

Function
A2 = DegToRad(A1)

Description
Convert the angle into correspond radian, and display.

Parameters
A1: inputting angle supports address, other variable or floating.
A2: outputting radian supports address, other variable or floating.

Example
```
@W_HDW12=DegToRad(@W_HDW10)  ' input angle on (HDW10),convert to correspond redian and copy to (HDW12)
```

Result: HDW10=180; HDW12=3.14159

```
dim a as floating          ' set variable
dim b as floating          ' set variable
b=30                  ' input angle
a=DegToRad(b)          ' convert the length of radian and copy to variable {a}
float2d("@W_HDW0", a)    ' display the value of floating on(HDW0)
```

Result: HDW0=0.52360

DIM ... AS ...

Function
Dim “variable” as “date type”

Description
Declare a variable, stable the type of data.

Parameters
Variable: begin with letter, other character can be letter, numbers, underscores ('_'),must begin with '@' if it is address;
Data type:string,floating,integer;
Notice: use the variable of Dim during running, cannot change the type, Dim will be missed if not define the type of variable. Variable can be declared once.

Example
```
dim a as integer            'define a as integer
dim @W_HDW0 as floating   'define @W_HDW0 as floating
dim hi as string            'define "hi" as string
```

Result:
a is integer
@W_HDW0 is floating
hi is string

DO ... LOOP

Function

Do [While | Until condition]
[statements]
Loop
or
Do
[statements]
Loop [While | Until condition]

Description

Condition determent instruction.
Do while...loop executes an instruction of block repeatedly when condition is true.
Do until...loop executes an instruction of block repeatedly until condition is true.

Parameters

Condition: determine condition; obtain the expression of True or False.
Statements: execute one or more instructions repeatedly when condition is True or until condition is True.
If condition is true, all statements are executed until the Wend statement is encountered. Control then returns to the While statement and condition is again checked. If condition is still True, the process is repeated. If it is not true, execution resumes with the statement following the Wend statement.

Example

```
dim i as integer        'end DO loop when i=100
do while i<100
i=i+1
@W_HDW0=i
loop
```

Result: HDW0=100

End

Function
Terminates the script immediately.

Description
End the execution of script.

Parameters
Statement: Judging condition, use with IF together. end script when meet condition.

Example
```
If a = 10 Then End   'end script when a=10.
```

Result: End the script program.

Exp

Function
A1=Exp(A2)

Description
Returns the power value of e (natural logarithm), save the outputting result to A1, e=2.71828182846.

Parameters
A1: the goal date: the power floating value of returning, must begin with '@'(e.g.@W_HDW10);
A2: Source data, natural exponential function, must be integer or variable. Cannot not begin with the address of "@"(e.g.@W_HDW10)

Example
```
dim a as integer          'define a as integer
a = @W_HDW2          ' assign the value of (HDW2) to variable a
@W_HDW0= Exp(a)      'exponential is the value of (HDW2),save result to(HDW0)
```

Result: HDW2=2,
HDW0=7.38905600

F2D

Function
F2D (A1, A2)

Description
Convert a 32 bit floating to integer format, then output the result.

Parameters
A1:Goal date, the value can be a address(e.g.@W_HDW12).
A2:source date, it can be a address or other variable.

Example
```
F2D(@W_HDW12,@W_HDW10)  'convert the floating of (HDW10) to integer, save in (HDW12).
```

result:
HDW10=200,
HDW12=200

F2S

Function
F2S (A1,A2,s1)

Description
Output a format of floating that in the type of string.

Parameters
A1: source address, used to store floating, the value is a address(e.g.@W_HDW200);
A2: Goal address, used to store string after converted, value is a address(e.g.@W_HDW100).
S1: the format of displaying goal data. such as the format of 03.03f,f, used for outputting a single-precision in the form of decimal.m.nf:means m column and n decimals when outputting.

Example
F2S("@W_HDW200", "@W_HDW100", "03.03f") '(HDW200) is floating input,(HDW100) is text output;

result:HDW200=1.22365,
HDW100=1.224

FILL

Function
FILL (A1, A2, A3)

Description
Write the same value to designated address constantly.

Parameters
A1: The beginning address, it can be a address(e.g.@ W_ HDW25);
A2: Source data, it needs to be written in continuous value, the value can be a address, variable or constant;
A3: The number of operation, writing address number, it can be a address, variable or constant;

Example
```
FILL (@W_HDW25, 10, 3)   'At the beginning three address of @W_ HDW25 is 10.
```

result:At the beginning three address of @W_ HDW25 is 10,@W_ HDW25=10,@W_ HDW26=10,@W_ HDW27=10.

Float2D

Function
Float2D (A1,A2);

Description
Copy floating value to the address.

Parameters
A1: Goal address, the value must be address (e.g.@W_HDW102);
A2: Source data, it can be floating;

Example
```
dim f as floating          'define f as floating
f=1.1                  'assign a designated value to f
Float2D ("@W_HDW102",f)   'assign the value f to HDW102
```

Result: HDW102=1.1

For... to... step...next

Function

For counter = start to end Step
[Statements]
Next

Description

Execute a command repeatedly for designated times.

Parameters

counter:Work as a variable for loop counter;
start:The start value of counter, it can be any variable type or expression;
end:The end value of counter, it can be any variable type or expression;
step:Every loop, the changed value of counter is step value, step default if it is not designation. step default is -1,when start>end,step default is 1 when start< end. It can be any variable type or expression;
statements:Between For with Next, execute instruction set of designated times;
Set a loop of for...Next in another loop, it can nest call the loop of for...Next. Different from a while, for only search once value from end. Empty for loop will be ignored, and cannot delay time.

Example

```
'Use HDX2.0 to trigger the loop
for i=100 to 0 step -5          ' set the start is 100,end is 0,subtract 5 every time, execute 20 times totally.
@W_HDW100=@W_HDW100+1    ' execute (HDW100+1) 21 times totally, the final result is 21.
Next
@B_HDX2.0=0
```

Result: HDW100=101

Function

Function

Function name (arglist)
  statements
  name = expression
  statements
End Function

Description

Differ from internal function, need to declare the name, parameter, code of the function.

Parameters

Name:function name.
arglist:stands for the variable list of parameter, this parameter will be entered when calling function. use comma to separate.
statements:a set of code in running function body.
Notice: it can not define a function program at any other program body. Write name first and then follow with parameter list, when calling function. Declare function must before calling. in the internal function body, it can assign to a function name from return value at any place. Return value is 0 if not assign function name. Functions can recursive call, but, it may lead to stack overflow.

Example

```
Function sincos (angle as floating)
    sincos = sin(angle) + cos(angle)
End Function<
.....
@W_0002 = sincos(pi/2)
.....
```

Goto

Function
Goto label

Description
Go to the designated row unconditionally in a function body.

Parameters
Label: target character, start with letter in row label, end with(:)of any string.row label has no sensitive to the format of letter.
notice:Goto only can jump into the internal function that visible row.

Example
```
Goto sd   'go to the row which start with "sd";
```

Result: Go to sd row.

H2A

Function
A1 = H2A (A2)

Description
Convert a binary(16 bit) to hexadecimals(4 bit) of ASCII.

Parameters
A1:return value, string, it can be an address or variable.
A2:binary is needed to be converted, the value can be a address or variable.

Example
```
@W_HDW100= H2A (@W_HDW0)   ' convert the binary of (HDW0) to character and save in (HDW100).
```

Result: HDW0=200,
HDW=100

Hypot

Function

Var = Hypot (expr1, expr2)

Description

Calculate the value of the hypotenuse of a right triangle.

Parameters

expr1, expr2:source data, the two sides of right triangle. it must be address;
Var:target data,it must be address;
Notice: hypot function can support integer and floating when the format of source data and target data are the same.

Example

@W_HDW200=Hypot (@W_HDW105,@W_HDW108) ' input the value of right-angle side at (HDW105) and (HDW108),and assign the result of the hypotenuse to (HDW200).

Result: HDW105=3,

HDW108=4,

HDW200=5

IF ... THEN ... ELSE ... END IF

Function

If condition Then
Statements
[Else
elsestatements]
End If

Description

Execute correspond instruction when fulfil condition. condition will be tested when executing if. it will execute the later instruction block of then, if condition is true. otherwise, execute the later of else. complete the two instructions, next execute the later of End if.

Parameters

condition:any expression, the value can be true or false.
statement:execute the instruction block when condition is true.
else statement:execute the instruction block when condition is false.

Example

```
if @W_HDW105=200 then        ' judging condition: whether the value of (HDW105) is 200
@W_HDW108=1             'the value of (HDW108) is 1 if fulfil condition
else
@W_HDW200=1             ' the value of (HDW200) is 1 if not fulfil condition.
Endif
```

Result:
HDW105=199;
HDW108=0;
HDW200=1

InStr

Function
var = InStr (“str1”, “str2”)

Description
Return the position of str1 in str2(start with 0), set -1 if not find.

Parameters
str1: source string, it can only be string, not address;
str2: target string, it can only be string, not address;
var: return value, the format of data must be string;

Example
```
dim a as floating
a = InStr ("Hello", "o")        ' calculate the position of“o”in“hello”.
float2d ("@W_HDW0",a)      ' return value is 4.(start with 0)
```

Result: HDW0=4

InvB subfunction

Function
InvB (A1)

Description
The state of inverse bit, it is a subfunction, so has no return value. Achieve the state of switching address constantly.

Parameters
A1: it is an address.

Example
```
InvB (@B_HDX0.1)  'switch the state if (HDX0.1).
```

Result: Switch the state of (HDX0.1) constantly.

IsFloating

Function

A2=IsFloating (A1)

Description

Decide whether a parameter is floating, return true if it is floating, otherwise return FALSE.

A1: source data, variable;

A2: target data, must be variable, it cannot use system address directly.

Example

```
dim a as integer
dim b as floating
b= D2float ("@W_HDW200",b)    'assign the value of (HDW200) to b
a = IsFloating (b)            'judge whether b is floating or not
@W_HDW300=a                   'save the result to (HDW300)
```

Result: HDW300=1

IsInteger

Function
A2= IsInteger (A1)

Description
Determine whether a parameter(A1) is integer, returen TRUE if the parameter is integer, otherwise return FALSE.

Parameter
A1: Source date, it is variable or number;
A2: Target date, must be variable, it cannot use system address directly;

Example
```
dim a as integer
a = IsInteger (20)     'determine whether 20 is integer
@W_HDW300=a     ' display the result on (HDW300)
```

Result: HDW300=1

IsString

Function
val = IsString(expr)

Description
Determine whether a parameter is string, return TRUE if it is string, otherwise return FALSE.

Parameters
Expr: source string, it can be a variable or string, not address;
Val: target date, the result must be variable, cannot be a address;

Example
```
dim a as integer        'define variable, display the result;
a= isstring ("hello")      'determine whether“hello”is string;
@W_HDW0=a          'assign the result to (HDW0)
```

Result: HDW=1

LCase

Function
A2 = LCase(A1)

Description
Return converted string all parameters to low edian.

Parameters
A1: source string, it can be a address or variable;
A2: outputting string, it can be a address or variable;

Example
```
@W_HDW33 = LCase (@W_HDW25)   'input source sting on (HDW25),convert it to target string and display the result on (HDW33);
```

Result: HDW25=HELLO
HDW33=hello

Left

Function
Val =Left (String, Length)

Description
Count a string from left then return string length.

Parameters
String: source string; it can be an address or string.
Length: return the number of character. It can be an address, integer or variable. Return empty string if length<1.return the whole string if length not less than the character number of string.
Val:target string, outputting string, it can be a address or variable.

Example
```
@W_HDW30=Left (@W_HDW36, @W_HDW40)  '(HDW36) used to input source string,(HDW30) used to display the string result;
```

Result: HDW36=hello,
HDW40=2,
HDW30=he

Len

Function
Length=Len(String)

Description
Return the string length.

Parameters
String: source string, it can be a address or string;
Length: target data, return value, it can be a address, variable, integer or floating;

Example
```
    @W_HDW30=Len (@W_HDW36)    'count the character number of (HDW36),save the result to (HDW30);
```

Result: HDW36=hello
        HDW30=5

Log

Function

a= Logn (x)=Log(x)/Log(n)

Description

Log function:return the natural logarithm of the value.

Parameters

a: source date,it can be a variable, but it cannot use address directly;
x, n: source date, it can be a variable, but it cannot use address directly;

Example

```
Dim a as integer   'define a as integer;
Dim b as integer   'define b as integer;
Dim c as integer   'define c as integer;
b=@W_HDW10   'assigns a value to b
c=@W_HDW20   'assigns a value to c
a=Log (b)/Log(c)   'count logarithm
@W_HDW0
=a    'assign the result to (HDW0)
```

Result: HDW10=27,
HDW20=3,
HDW0=3

Log10

Function
a=Log10(x)= Log(x)/Log(10)

Description
Log function: return the natural logarithm.

Parameters
A: target data, result can be variable, cannot use address directly;
x: source data, it can be variable that must be the multiples of 10,cannot use address directly;

Example
```
dim a as integer    'define a as integer
dim b as integer    'define b as integer
b=
@W_HDW10    'assign a value to b
a= Log (b)/Log(10)   'result
@W_HDW0=a     'assign the result to (HDW0)
```

Result: HDW10=100,
HDW0=2

LTrim

Function
val=LTrim("string")

Description
Remove the left empty part of the string and return.

Parameters
Val: target sring, it can be either a variable or address;
string: source string, it can be either a variable or address;

Example
```
dim a as string
a=Ltrim("    hello")
@W_HDW103=a
```

Result: HDW103=hello

MAX

Function
A1=MAX(A2,A3)

Description
Compare the value of A2 and A3, assign the greater number to A1.

Parameters
A1: return value (used to store the greater number between A2 with A3).
A2: the first comparison value.
A3: the second comparison value.
Notice: A1,A2,A3 only used in unsigned integer or unsigned address.

Example
```
DIM A1 as integer
@W_HDW106=10 'assign the value to (@W_HDW106),unsigned decimal word.
@W_HDW107=5  'assign the value to (?@W_HDW107),unsigned decimal word.
A1 = Max(@W_HDW106,@W_HDW107)
@W_HDW105 = A1
```

Result: @W_HDW105 = 10

Mid

Function
A1=mid(A2,start,length)

Description
Returns a string contain a specified characters length from a string.

Parameters
A1: string contains selected characters
notice: A1 must be a string

A2: string of being selected
notice: A2 must be a variable or address

Start: the start position of string
notice: Start must be a variable or address, it means that count begin with 0.

Length: the designated length of string
notice: length no more than 127

Example
```
DIM A1 as string
A1 = Mid("hellokitty",1,2) 'select the string of in "
@W_HDW106=A1
```

Result: @W_HDW106 'display "el" on text input and output window

MIN

Function
A1=MIN(A2,A3)

Description
Compare the value of A2 and A3, assign the smaller number to A1.

Parameters
A1:return value (used to store the less number between A2 with A3).
A2:the first comparison value.
A3:the second comparison value.
Notice: A1,A2,A3 only used in unsigned integer or unsigned address.

Example
```
DIM A1 as integer
@W_HDW106=10   'assign the value to (@W_HDW106),unsigned decimal word.
@W_HDW107=5   'assign the value to (?@W_HDW107),unsigned decimal word.
A1 = Min(@W_HDW106,@W_HDW107)
@W_HDW105 = A1
```

Result: @W_HDW105 = 5

MSeconds

Function
A1=MSeconds( )

Description
A1 is used to display the current microseconds of system.

Parameters
A1:used to store the current microseconds of system.
Notice: A1 is unsigned integer variable or unsigned integer address.

Example
```
DIM A1 as integer
@W_HDW0= 10      'assign a value to (@W_HDW0), unsigned decimal word
A1=MSeconds()    'return the current microseconds of system to A1
@W_HDW0= A1      'display microseconds on screen,(HDW0) is a unsigned decimal integer address
```

Result: @W_HDW0 will generate the time value of changing microseconds unit.

NewNoAddr

Function
A1= NewNoAddr (A2, length)

Description
At the basic of source address A2,offset designated length,obtain a new address A1.

Parameters
A1:address after offsetting
Notice: A1 must be variable.
A2: source address
Notice:A2 must be address(e.g.:"@W_HDW000002")
Length:offset length

Example
```
DIM A1 as string
A1=NewNoAddr("@W_HDW0",50) '(HDW0) offsets 50 words address(16 bit),and save the result to A1
@W_HDW1=A1                 '(HDW50) save in A1
```

Result: (@W_HDW1) character input/display will show @W_HDW50

NewStatAddr

function

A1= NewStatAddr(A2,length)

description

At the basic of source address A2,offset the designated length,to obtain a new station A1.

parameters

A1: The address after offsetting

Notice: A1 must be variable.

A2: Source station address

notice:A2 must be address (e.g.:"@W_1:10").

Length: offset length

Example

```
DIM A1 as string
A1=NewStatAddr("@W_1:10",2) 'address 10 of station address1 that offset 2 station addresses,then save the result to A1
@W_HDW1=A1                  'address 3:10 is saved in A1
```

Result: @W_HDW1 character input/display will show @W_3:10

NStringCompare

Function
A1= NStringCompare(A2,A3,length)

Description
Compare whether the designated length of two strings is the same,return 1 to A1 if yes,otherwise return 0.

Parameters
A1:return value (compare the designated length of two strings,display 1 when equal,else 0).
Notice: A1 can be a address integer or variable.

A2:the address of string to be compared
Notice: A2 must be address.

A3:source string
Notice: A3 must be variable or constant string.

Length:string length to be compared

Example
```
@W_HDW1= NStringCompare("@W_HDW0","87654",5)
if @W_HDW1=1 then
@B_HDX10.0=1 'result: HDX10.0 set ON ‘when the two strings are the same.
endif
if @W_HDW1=0 then
@B_HDX10.0=0 'result:HDX10.0 set OFF ‘when not equal.
endif
```

Power

Function
var = power(expr1, expr2)

Description
The value of [expr2] to the power of [expr1] will be assigned to Var.

Parameters
var: return value.
expr 1: base number.
expr 2: power number.

Example
```
Dim a as floating
a=power (2, 3)       'the value of 3 to the power of 2 is assigned to a.
Float2D("@W_HDW10",a)   'assign the float value of a to @W_HDW10
```

Result: @W_HDW10=8

RadToDeg

Function
Var= RadToDeg(expr)

Description
Convert radiant value to degree, then assign to Var.

Parameters
Var: return degree value.
expr: input radiant value.

Example
```
Dim a as floating
a = RadToDeg(pi)     'assignt the degree value of p to a.
Float2D("@W_HDW4",a) 'assign the degree value to address "@W_HDW4".
```

Result: @W_HDW4=180

RAND

Function
Var = rand(expr1)

Description
Generate a random number.

Parameter
Var: generated random number.
Expr1: the base number.

Example
```
@W_HDW0=rand(@W_HDW0) 'Set the value of address@W_HDW0 as the base number to generate random number.
```

Result: @W_HDW0 random number.

ReadAddr

Function
Word = ReadAddr(A1)

Description
Assign the red value from A1 to word.

Parameter
Word: return value

Example
```
Dim word as integer
@W_HDW100=10
word = ReadAddr("@W_HDW100") 'Read the value of address @W_HDW100 and assign to word.
@W_HDW200=word
```

Result:@W_HDW200=10

Right

Function
val = Right(string, length)

Description
Return the designated number of string to val.

Parameter
String: the operated string.
Length: the designated number of byte required to return, count from the right side.

Example
```
@W_HDW103= Right("Hello", 3) 'return "llo"
```

Result:@W_HDW103="llo"

RTrim

Function
val = RTrim(str)

Description
Clear the empty part on the right side of string [str], then assign the empty part to val.

Parameter
val: return value.
str: the string need to be operated.

Example
```
@W_HDW0 = RTrim("  -Hell  o-    ") '  retrun"  -Hell  o-    "
```

Result: @W_HDW0 'display " -Hell o-"

SetB sub

Function
SetB(A1)

Description
Set the bit A1 ON.

Parameters
A1:Bit address

Example
```
SetB(@B_HDX100.0) 'Set the address {@B_HDX100.0} ON
```

Result: @B_HDX100.0=1

SignedInt16

Function
val = SignedInt16(A1)

Description
Assign the value to {val} from address A1 which is signed integer.

Parameters
A1:contain signed integer as "@W_HDW000002"
Val:return value

Example
```
Dim a as integer          'Integer variable a
a = SignedInt16("@W_HDW0") 'read signed integer from HDW0 addresses and assign the value toa
@W_HDW2=a                 'assign the value a to HDW2
```

Input: @W_HDW0=-2:
Result: @W_HDW2=-2.

SignedInt32

Function
val = SignedInt32 (A1)

Description
Assign the value to {val} from address A1 which is signed even integer.

Parameters
A1: the address contains signed even integer
Val: Return value

Example
```
Dim a as integer          'define {a} as a integer
a = SignedInt32("@W_HDW0") 'read signed even integer from HDW0, then assign this value to a.
@W_HDW2=a                 'assign the value of a to HDW2
@W_HDW13=a>>16
```

Input: @W_HDW0=-2
Result: @W_HDW2=-2
@W_HDW13=-1

Sin

Function
val = Sin(A1)

Description
Implement sin calculation to computing the value in A1 then assign to val.

Parameters
A1: A1 needs to be a angle.
Val: Return value.

Example
```
Dim a as floating    'floating variable a,b
a=sin(pi/6)          'return sinb to a
Float2D("@W_HDW13",a)  'assign the value of the floating variable a to address HDW13.
```

Result: @W_HDW13=0.5

SleepA

Function
SleepA(T)

Description
Wait time T(ms).

Parameters
T: wait time, the unit is [ms]
Return value: not available.

Example
```
SleepA(10) 'wait 10ms
```

Result: When the script runs to SleepA(10) then wait 10ms to continue running.

Sqr

Function
val = Sqr(A1)

Description
Assign a square root value of A1 to val.

Parameters
A1:the data need to be operated
Val:Return value

Example
```
@W_HDW0 = Sqr(4)  'calculate the square root of HDW0
```

Result: @W_HDW0=2

Sub

Function

Sub name (arglist)
  statements
End Sub

Description

Declare the name, parameters and codes of the Sub (sub function)

Parameters

Name:naming rules refer to variable.
Arglist:variable list.
Statements: the code set of the sub function.

Example

```
sub samesub(a,b as integer) ' samesub and integer variable a,b
   c=a+b
   @W_HDW0=c
endsub
samesub(1,12)          'call function samesub
```

Result: @W_HDW0=13

SWAP subfunction

Function
SWAP(A1,length)

Description
Swap the High endian with the low endian from address A1, swap length is adjustable.

Parameters
A1:the swapped high edian, must be a address as HDW_000002.
Length:swap length.
Return value:Not available.

Example
```
 @W_HDW103=0x1234  'assign value to HDW103
@W_HDW104=0x2345  'assign value to HDW104
@W_HDW105=0x2565  'assign value to HDW105
@W_HDW106=0x2675  'assign value to HDW106
SWAP(@W_HDW103,4) 'swap the high and low endian for the 4 adjacent addresses start with HDW103.
```

Result:
@W_HDW103=0x3412
@W_HDW104=0x4523
@W_HDW105=0x6525
@W_HDW106=0x7526

Tan

Function
val = Tan(A1)

Description
Implement tan calculation to computing the value in A1 then assign to val.

Parameters
A1:A1 needs to be an angle.
Val:return value.

Example
```
Dim a as floating      'define a floating variable a
a=TAN(pi/3)     'calculate the tangent value of pi/3 and assign to a
Float2D("@W_HDW16",a)  'assign the value of a to HDW16
```

Result: @W_HDW13=1.732

Trim

Function
val = Trim(A1)

Description
Return a value of a address without empty string next to it.

Parameters
A1:The operated string
val:return value

Example
```
@W_HDW1=Trim(" ab  ")
```

Result: @W_HDW1="ab"

UCase

Function
val = UCase(A1)

Description
Capitalize the string data, and then assign the value to val.

Parameters
A1:operated string, address or variable.
Val:Return value

Example
```
@W_HDW1=ucase("abcd") 'Capitalize abcd then assign the value to HDW1
```

Result: @W_HDW1="ABCD"

Variable

Description

A variable is any factor, trait, or condition that can exist in differing amounts or types.

Define variable

Use Dim to define variable in script. The variable can be string, floating, integer.

Example:

```
Dim a as floating ’define variable {a} as a floating.
Dim b,c,d as integer’define variable {b,c,d} as integer
```

Naming rules

The first letter must be English letter.
No symbols.
Maximum length is 15 words.

W2B

Function
W2B(A1, A2, A3)

Description
Replace the high endian of [A2]+1 with the high endian of A2.

Parameters
A1: operated address.
A2: source address.
A3: the conversion length.
Return value: not available.

Example
```
@W_HDW0 = 4660 'assign 16bit value 1234 to HDW0.
@W_HDW1=0x5678 ''assign 16bit value 5678 to HDW1.
@W_HDW2 = 0x2425 ''assign 16bit value 2425 to HDW1.
@W_HDW3 = 0x3536 ''assign 16bit value 3536 to HDW0.
@W_HDW4 = 0x1415 ''assign 16bit value 1415 to HDW0.
W2B(@W_HDW20,@W_HDW0, @W_HDW10)
@W_HDW10=1: save the high endian {34} of HDW0 to HDW20.
```

Result: @W_HDW20=0x34, @W_HDW21=0, @W_HDW22=0

W2D

Function
W2D(A1, A2)

Description
Convert the unsigned Word to unsigned Dword and save the result in A1.

Parameters
A1: operated address.
A2: source address.
Return value.

Example
unsigned decimal word

```
@W_HDW0 = 1234 'assign 1234 to HDW0.
W2D(@W_HDW2, @W_HDW0) 'convert unsigned word {1234} from HDW0 to Dword and save in HDW2
```

Result: @W_HDW0=12345,@W_HDW2=12345,@W_HDW3=0

signed decimal word

```
@W_HDW0 = -12345 'assign value to HDW0: convert {-12344} to unsigned decimal word is {53191}.
W2D(@W_HDW2, @W_HDW0) 'save unsigned Dword to HDW0
```

Result: @W_HDW0=-12345,@W_HDW2=53191,@W_HDW3=0

W2F

Function
A1 = W2F (A2)

Description
Convert a 16bit integer to a 32bit floating, and then save to the next word of A1.

Parameters
A1: operated address.
A2: source address.
Return value: not available.

Example
1. A1, A2 are addresses

```
@W_HDW0 = 1234 ' assign unsigned word {1234} HDW0 @W_HDW1=W 2F(@W_HDW0) '  Convert {1234} to a 32bit floating and then save to HDW1, HDW2.
```

Result: @W_HDW1=1234'32bit floating

2. A1 is an address,A2 is variable

```
dim a as integer
a=134 'define a integer 134 to a,
@W_HDW2=W2F (a) 'convert to 32bit floating save to HDW1, HDW2.
```

Result:@W_HDW1=134' 32bit floating

W2S

Function
W2S(A1,A2,S1)

Description
Convert integer word in address A1 text as S1 format, and then save to A2.

Parameters
A1: operated address.
A2: source address.
S1: saving format.

1. d format:Decimal format.
   d:real data length.
   Md:designated data length.
   0md:designated data length if the length is shorter than m add 0 at the left.
2. oformat:Unsigned octonary format. Mo and 0mo is also applied.
3. x format:unsigned Hex integer format?Mx and 0mx is also applied.
4. c format: ASCII format.

Example
1. Decimal format

@W_HDW1=1456 'assign value {1456} to HDW1.
W2S("@W_HDW1", "@W_HDW10", "6d") ' convert{1456} to decimal text and save to HDW10.

Result: @W_HDW10 shown "1456"

2. 0md

@W_HDW1=1456 ' assign value {1456} to HDW1
W2S("@W_HDW1", "@W_HDW10", "06d") ' convert{1456} to integer decimal text and add 2 {0} on the left of the data then save to HDW10.

Result: @W_HDW10 show text "001456"

WHILE ... WEND

Function

While condition
[statements]
Wend

Description

If the condition is true, then all the commands before Wend in the statement will be executed then recheck the condition, if the condition is false, the command after Wend will be executed.

Parameters

Condition:Number or string, the result represent as True or False.
Return value: not available.

Example

```
while @W_HDW1>50 'the condition is the value of HDW1 bigger than 50.
@W_HDW1=@W_HDW1-1 'when the condition is true, execute subtract 1 from 1HDW.
wend
@W_HDW2=@W_HDW2+1 ' when the condition is false, execute add 1 from 1HDW.
```

Execute: If HDW1=60,after executed; HDW1=50, if the condition is true.

WriteAddr

Function
WriteAddr(A1,A2)

Description
Assign the value from A2 to address A1.

Parameters
A1: operated address
A2: source address
Return value: not available.

Example
```
dim f as integer  ' integer f
f=13 ' assign the value 13 to f
WriteAddr("@W_HDW1",f) ' write the value to HDW1.
WriteAddr("@W_HDW10",@W_HDW2) ' write the value from HDW2 to HDW10.
```

Result:
HDW1=13
HDW10= HDW2'IF HDW2=1456,Then HDW10=1456;IF HDW2=-123,Then HDW10=-123

HMI Address

HMI provide four types of HMI address for user:

System Parameters (HSW/HSX)

The address starting with “HSW” are reserved by HMI system, define those address to save the state or parameter of current system. Like system properties, communication parameters etc.

User Data (HDW/HDX):

The address starting with “HDW” are provided for user, save data to this type of address. Provide holding addresses (HDW8000~HDW30000) as well.
Special Data(HPW/HPX):
Special address reserved by HMI.
Recipe Query(RPW):
Accessing recipe data by reading or writing those addresses.
The address format RPW**####; ** means group number; #### means data number; (e.g. RPW010002 means the second data in first group)

Notice:
HSW, HDW, HPW, RPW are word address, HSX, HDX, HPX are bit address;
HSW is related to HSX, (e.g. one word address HSW0 contains 16 bits from HSX0.0 to HSX0.15);
HDW is related to HDW, (e.g. one word address HDW0 contains 16 bits from HDX0.0 to HDX0.15);
HPW is related to HPX, (e.g. one word address HPW0 contains 16 bits from HPX0.0 to HPX0.15);
RPW has no related bit address.

Word Address	Address Format	Range
HSW	HSWn	HSW0~16000
HDW	HDWn	HDW0~30000
HPW	HSWn	HPW0~8191
RPW	RPWn	RPW000000~491000

Bit Address	Address Format	Range
HSX	HSXn.m	HSX0.00~16000.15
HDX	HDXn.m	HDX0.00~30000.15
HPX	HPXn.m	HPX0.00~8191.15

HPW Address		
Address	Meaning	Description
HPW0	Group number of recipe	Need to specify the group number.(value of HPW0) when you download or upload the data to recipe.
HSW Address		
Address	Meaning	Parameter
HSW00000	Save preferences setting	Save the parameter of project when HSW000000=1
HSW00001	Cancel parameter Settings	Cancel the parameter of project when HSW000001=1
HSW00002	HMI station No.	Reserved
HSW00003	PLC station No.	Reserved
HSW00004	Type of COM1 serial port	0: RS232 1: RS485 2: RS422
HSW00005	COM No.	0: COM1 1: COM2
HSW00006	COM1 baud rate (BPS)	0: 2400 1: 4800 2: 9600 3: 19200 4: 38400 5: 57600 6: 115200
HSW00007	COM1 data bits	0: 7 1: 8
HSW00008	COM1 parity	0: None 1: Odd 2: Even 3:SPACE
HSW00009	COM1 stop bit	0: 1 Bit 1: 2 Bit
HSW00010	COM1 flow control	0: None 1: Software 2: Hardware
HSW00011	COM1 number of retied	
HSW00012	COM1 waiting timeout	Unit: millisecond
HSW00013	COM1 receiving timeout	Unit: millisecond
HSW00014	Default screen No.	The first screen after starting the HMI
HSW00015	Display language	0: language 1 1: Language 2 2: Language 3
HSW00016	Font	Reserved
HSW00017	Font size	Reserved
HSW00018	Font quality	Reserved
HSW00019	Touch buzzer	0: True 1: False
HSW00024	Restart HMI	0: False 1: True
HSW00026	Printer type	Reserved
HSW00027	Print port	Reserved
HSW00028	Local time year	Range: 0~999
HSW00029	Local time - month	Range: 01~12
HSW00030	Local time - Day	Range: 01~31
HSW00031	Local time - Hour	Range: 0~23
HSW00032	Local time - Minute	Range: 0~59
HSW00033	Local time - Second	Range: 0~59
HSW00034	Upload project	1: Upload (Reserved)
HSW00035	Download project	1: Download (Reserved)
HSW00036	Type of COM2 serial port	0: RS232 1: RS485
HSW00037	COM2 baud rates	0: 2400 1: 4800 2: 9600 3: 9600 4: 19200 5: 38400 6: 57600 7: 115200

HSW00038	COM2 data bits	0: 7 1: 8
HSW00039	COM2 parity	0: None 1: Odd 2: Even 3: SPACE
HSW00040	COM2 stop	0:1 Bit 1:2 Bit
HSW00041	COM2 flow control	0: None 1: Software 2: Hardware
HSW00042	COM1 number of retied	
HSW00043	COM1 waiting timeout	Unit: millisecond
HSW00044	COM1 receiving data timeout	Unit: millisecond
HSW00045	Characters keypad display	Characters keypad display
HSW00112	Security protection	1: Enable security protection
HSW00113	Security level	1: level 1 2: level 2 3: level 3 .....
HSW00114	Current security level	1: level 1 2: level 2 ..... 11: level 11 12: level 12
HSW00126	Local time - week	0:Sunday 1:Monday 2:Tuesday 3:Wednesday 4:Thursday 5:Friday 6:Saterday
HSW00127	Beep	0: normal beeping 1: keep beeping (unlatched)
HSW00128	Script running cycle	
HSW00136~139	HMI IP address 1 (left to right)	
HSW00140~143	HMI subnet mask 1 (left to right)	
HSW00144~147	HMI default gateway 1 (left to right)	
HSW00148	HMI port No.	
HSW00157	Security level password independent	0: False 1: True
HSW00160~163	Password	8 bytes
HSW00169	COM 1 timeout retried	Range: 0~65535
HSW00170	COM 2 timeout retried	Range: 0~65535
HSW00173	COM1 delay read or write	Range: 0~65535
HSW00174	COM2 delay read or write	Range: 0~65535
HSW00175	COM3 delay read or write	0~65535
HSW00176	Empty history XY plot	0: not heavy painting empty 1: heavy painting empty
HSW00189	Display an error message box	0: False 1: True
HSW00191	Empty record data files,	0: False 1: True
HSW00192	Empty system configuration files,	0: False 1: True
HSW000200	The number of current bit alarm	
HSW000201	The number of current word alarm	
HSW000202	Empty bits alarm data	1: True 0: False
HSW000203	Empty words alarm data	1: True 0: False
HSW000205	Clear file under “\ Flash \ Data”	1: clear all file under” \ \ Flash \ \ Data “ 2: clear all file under”\ \ Flash \ \Data \ \ DL “ 3: clear all file under”\ \ Flash \ \ Data \ \ AL “ 4: clear all file under”\ \ CFDC \ \ Data “ 5: clear all file under”\ \ CFDC \ \Data \ \ DL “ 6: clear all file under”\ \ CFDC \ \Data \ \ AL “ 7: clear all file under”\ \ Udisk \ \ Data” 8: clear all file under”\ \ Udisk \ \Data \ \ DL " 9: clear all file under” \ \ Udisk \ \ Data \ \ AL “

HSW000214	Close the pop up window	1 :True
HSW000216	Keyboard case switching	0:capital 1:lowercase
HSW000217	Clear multi-language setting	0:False 1:True
HSW000218	Settings screen saver switch	0:False 1:True
HSW000219	Start time of screen saver countdown	Unit: second
HSW000220	Screen saver countdown	Unit: second
HSW000221	Screen sleep	Unit: second
HSW000223	Beep when alarm	0:True 1:False
HSW000224	PLC station No. Of COM2	Range: 0~255
HSW000225	HMI station No. Of COM2	Range: 1~255
HSW000226	PLC station No. Of COM2	Range: 0~255
HSW000227	HMI station No. Of COM2	Range: 1~255
HSW000230	Save CF/SD card data to U disk	0:False 1:True
HSW000233	Read data from PLC	0: False (from ache) 1: True (from PLC)
HSW000238	Bit alarm flag	1: True
HSW000239	Word alarm flag	1: True
HSW000243	Group number of Discrete recipe	Range: 1~3
HSW000244	Recipe upload or download	1: upload 2: download
HSW000245	Clear alarm record	Clear alarm record
HSW000247	Type of COM1 serial port	0: RS232 1: RS422 2: RS485
HSW000248	COM3 baud rates	0: 2400 1: 4800 2: 9600 3: 9600 4: 19200 5: 38400 6: 57600 7: 115200
HSW000249	COM3 data bits	0: 7 1: 8
HSW000250	COM3 parity	0: None 1: Odd 2: Even 3: SPACE
HSW000251	COM3 stop bit	0: 1 Bit 1: 2 Bit
HSW000252	COM3 flow control	0: None 1: Software 2: Hardware
HSW000253	COM3 number of reties	
HSW000254	COM3 waiting timeout	Unit: millisecond
HSW000255	COM3 receiving data timeout	Unit: millisecond
HSW000256	Save parameters of group No.0 data record	0:True 1:False
HSW000257	Sampling period of group No.0 data record	Unit: second (0 ~ 65535)
HSW000258	Autosave period of group No.0 data record	Unit: minutes (0 ~ 65535)
HSW000259	The type of saving data of group No.0 data record	0:According to setting to save data 1:save data to CF/SD card, data in HMI will be cleared
HSW000260	Save parameters of group No.1 data record	0:True 1:False
HSW000261	Sampling period of group No.1 data record	Unit: second (0 ~ 65535)
HSW000262	Autosave period of group No.1 data record	Unit: minutes(0 ~ 65535)
HSW000263	The type of saving data of group No.1 data record	0:According to setting to save data 1:save data to CF/SD card, data in HMI will be cleared

…	…	…
HSW000296	Save parameters of group No.10 data record	0:True 1:False
HSW000297	Sampling period of group No.10 data record	Unit: second (0 ~ 65535)
HSW000298	Autosave period of group No.10 data record	Unit: minutes(0 ~ 65535)
HSW000501	Default level of parts security	
HSW000502	Current level of parts security	
HSW000503	Import group No. of recipe	
HSW000504	export group No. of recipe	
HDW25000	Import file name of file list	
HDW25001	Export file name of file list	
HSW000506	Clear the current event list	
HSW000507	Clear history event list	
HSW000508	Delete EVReg.dat files	
HSW000509	Parts security password separated	
HSW000500	Whether opening parts password	
HSW000543	X position of touching	
HSW000544	Y position of touching	
HSW000546	U disk state for print part	HSW546=1
HSW000547	State of saving picture	HSW547=1:Saving HSW547=2:Saved
HSW000545	Screen No. of screen saver	
HSW000661-666	Year, month, day, hour,minute,second for File list	
HSW000667	File list import export state	HSW667=10000:importing HSW667=10000+group:Import success HSW667=20000:Import failure HSW667=30000:Exporting HSW667=30000+group:Export success HSW667=40000+group:Export failure
HSW000299	10 sets of data recording parameters Settings Immediately store data records This setting can power lost preservation	0:According to the engineering of setup time preservation 1:Immediately storage to CF card, within the system will be reset
HSW000744-HSW001255	Latched address	Read or write data by this area may damage FLASH
HSW000542	=1	Memory is full already
HDW8000-HDW30000	Latched address	Total addresses length: 16 K + 2000 words, Read or write data by this area may damage FLASH
Picture level password		
HSW000404	Safety level 1 password	Total 8 bytes
HSW000408	Safety level 2 password	Total 8 bytes
HSW000412	Safety level 3 password	Total 8 bytes
HSW000416	Safety level 4 password	Total 8 bytes
HSW000420	Safety level 5 password	Total 8 bytes
HSW000424	Safety level 6 password	Total 8 bytes
HSW000428	Safety level 7 password	Total 8 bytes
HSW000432	Safety level 8 password	Total 8 bytes
HSW000436	Safety level 9 password	Total 8 bytes
HSW000440	Safety level 10 password	Total 8 bytes
HSW000444	Safety level 11 password	Total 8 bytes
HSW000448	Safety level 12 password	Total 8 bytes
Parts level password		
Password using the internal parts HSW address words address type		

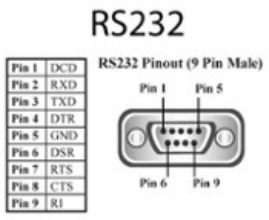
safety level 1	Total 8 bytes	HSW452
safety level 2	Total 8 bytes	HSW456
safety level 3	Total 8 bytes	HSW460
safety level 4	Total 8 bytes	HSW464
safety level 5	Total 8 bytes	HSW468
safety level 6	Total 8 bytes	HSW472
safety level 7	Total 8 bytes	HSW476
safety level 8	Total 8 bytes	HSW480
safety level 9	Total 8 bytes	HSW484
safety level 10	Total 8 bytes	HSW488
safety level 11	Total 8 bytes	HSW492
safety level 12	Total 8 bytes	HSW496

open parts level password	HSW500
The default parts password level	HSW501
Current part password level	HSW502

RS 232C/RS422/RS485 Serial Interface

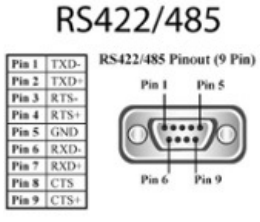
RS-232 serial interface

RS232 was first introduced in 1962, This is the most common type of serial interface, it was the standard communication before the PS2 and USB become popular in the computer industry, you use to connect mouse, modem and printer to RS232 serial interface. RS232 only allows for one transmitter and one receiver on each line. RS232 also use a Full-Duplex transmission method. RS232 can transmit up to 1Mbps with maximum distance up to 50 feet.



RS-422 serial interface

RS422 is an improved version of RS232, it uses twisted pair cable to reduce the noise, and it uses signaling balancing to transmit data, so what is signal balanced – It uses a voltage-difference between the two lines as an indication of the signal value, with this method the data is able to transmit for longer distance with faster data rates, with RS422 the data can transmit up to 10 Mpbs at 50 feet or 100 Kbps at 4000 feet. RS422 is capable of multi-drop capability, it limits up to 10 slaves in the data line.



RS485 serial interface

RS485 is an improved version of RS422, it expands on the capabilities, the major change is to have multi-drop Limitation of RS422, it allowing up to 32 devices to communicate through the same data line. Any of the slave devices on an RS-485 bus can communicate with all the slave within the data line without going through master device.

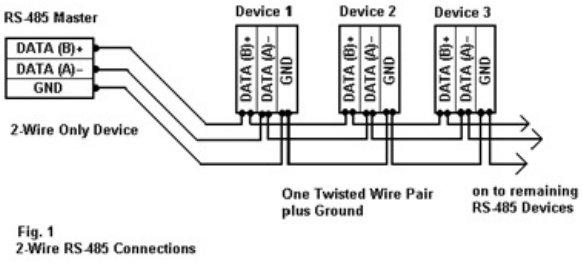


Fig. 1
2-Wire RS-485 Connections

Item	RS232	RS422	RS485
Cabling	Single ended	Single ended multi-drop	Multi-drop
Number of Devices	1 transmit 1 receive	1 transmitters 10 receivers	32 transmitters 32 receivers
Communication Mode	Full duplex	Full duplex, half duplex	Full duplex, half duplex
Max Distance	50 feet at 19.2 Kbps	4000 feet at 100 Kbps	4000 feet at 100 Kbps
Max. Data Rate	1Mpbs for 50 feet	10 Mpbs for 50 feet	10 Mpbs for 50 feet
Signaling	Unbalanced	Balanced	Balanced
Mark (data 1)	-5V min. -15V max.	2V min. (B>A) 6V max. (B>A)	5V max. (B>A)
Space (data 0)	5V min. 15V max.	2V min. (A>B) 6V max. (A>B)	5V max. (A>B)
Input Level Min.	±3V	0.2V difference	0.2V difference

Data Record

Collect data in designated time period, and save the data to storage device, The record data file can be saved in the SD card, U disk or external storage.

Data Record

Click [Setting]-[Data record] to edit its properties:

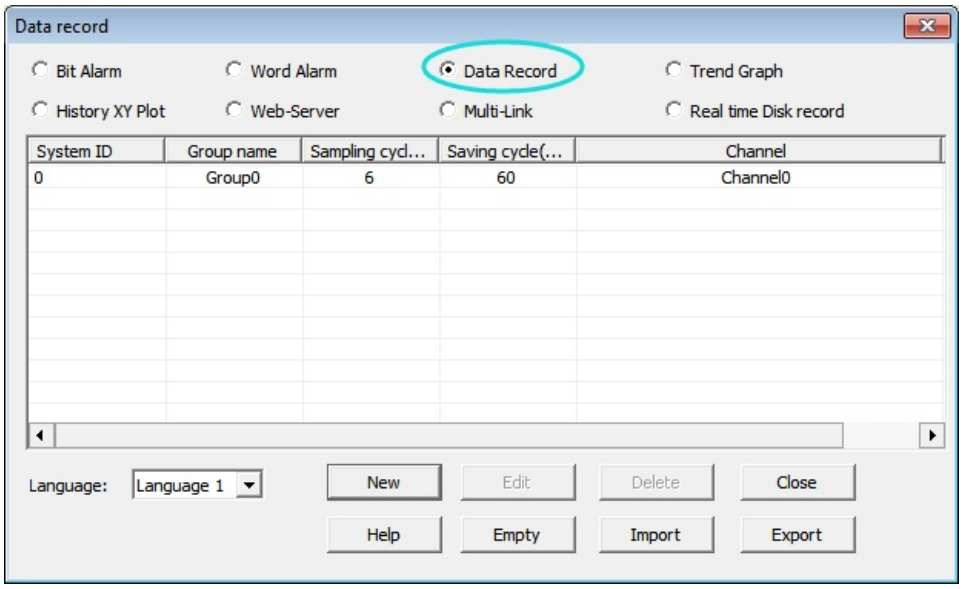


Figure 1

1. Language: choose display language.
2. Import: import new record group from CSV file.
3. Export: export the current record group to CSV file.

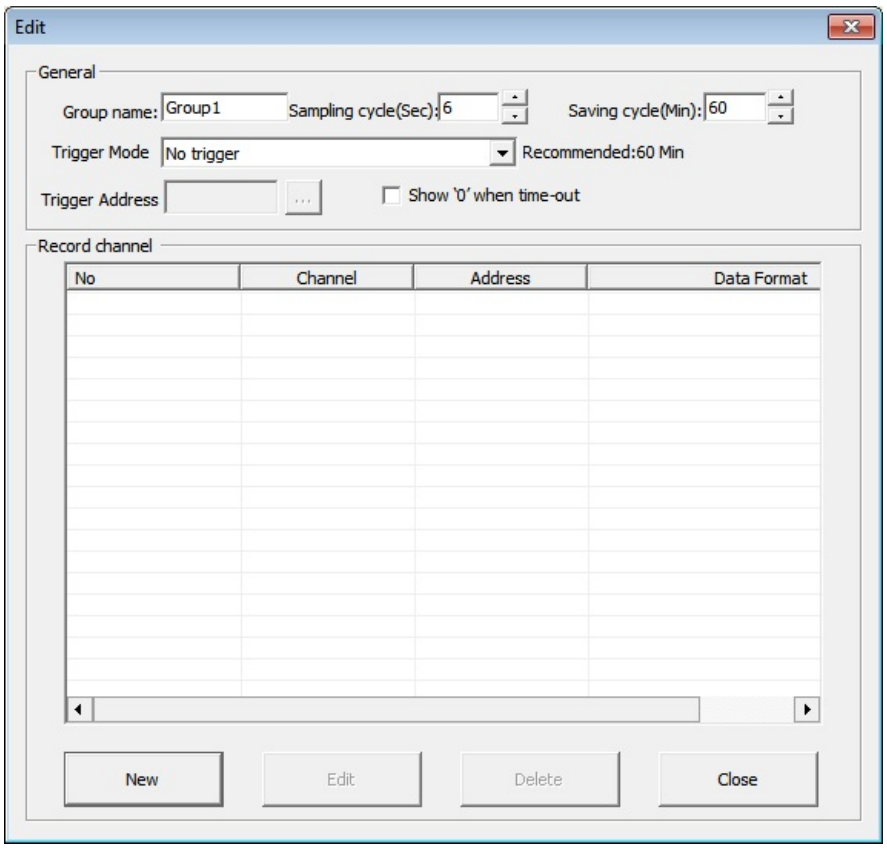


Figure 2

1. Group name: group name, should be unique;
2. Sampling cycle: the sampling cycle of the group;
3. Saving cycle: the saving cycle of the group, in minute;
4. Trigger mode: Trigger to record the ;
5. Trigger address: Control to collect the record;
6. Show “0” when time out: the data record will display 0 when channel failed to get the data.

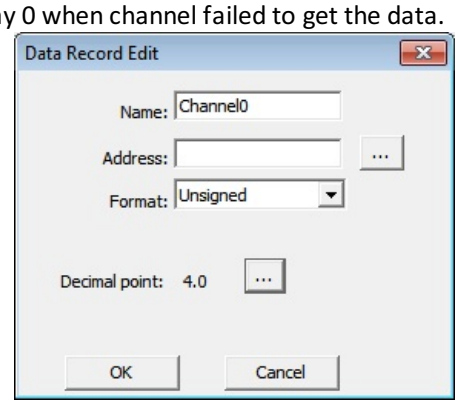


Figure 3

1. Channel name: group name, it is sole, can not repeat;

2. Address: the word address need to be record, it can be external register address, also can be HMI register address;

3. Display format: Data transfer format, current support binary system, octonary number system, unsigned, signed, hexadecimal, BCD,32 floating,32 unsigned,32 signed, character;

4. Decimal point: whether to display the decimal point and the number after the decimal point in the data storage medium.

Data record file

Click the “Export” in the “Data record” window after set, then can export an CSV file.

1. The first line of the CSV: STUDIO, DataLog;

2. The second line of the CSV: Language of group 1, Language of group 2, Language of group 3, Sampling cycle(in second),save cycle(in minute),trigger mode, trigger address,

Channel language 1, Channel language 2, Channel language,3 address, data format, data length, the place of decimal point.

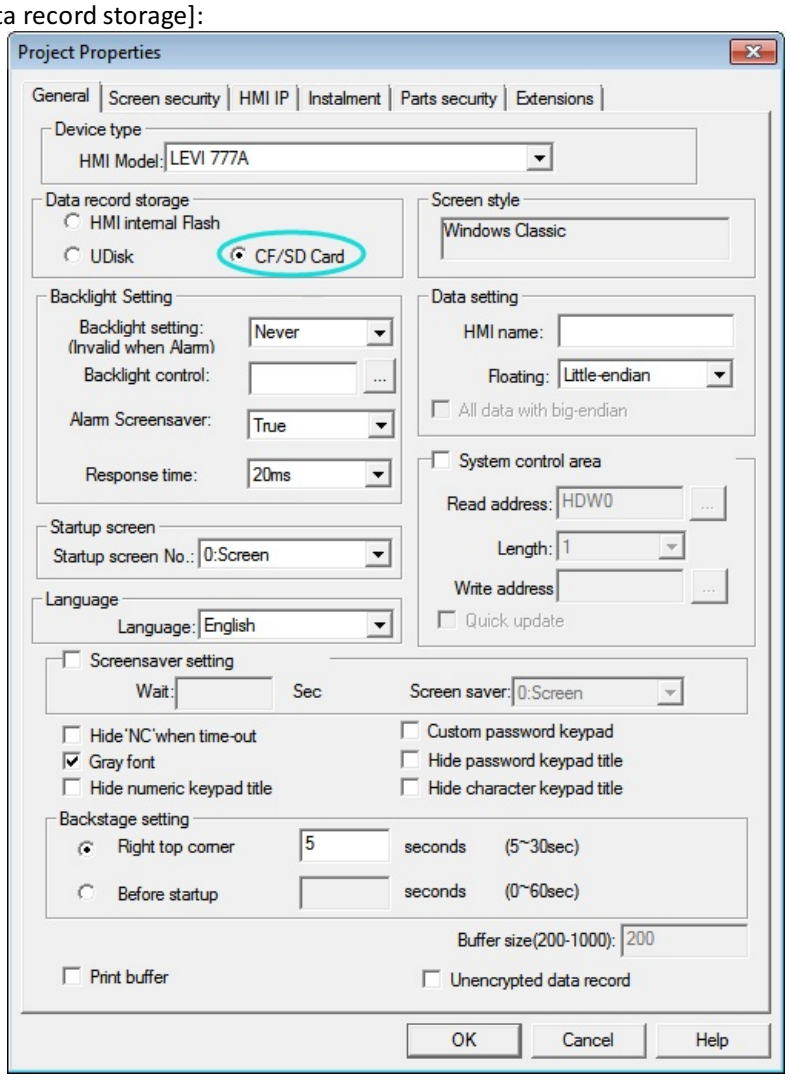
3. The last line of CSV:END;

4. Data format: binary system (0), octonary number system(1), unsigned(2), signed(3),BCD(4),32 floating (5),signed(6),32 unsigned(8),32 signed(9),character(11)

Data Record Storage

Storage

Data record file can be saved in different storage medium according to customer requirement.
Click [Setting]-[project Properties]-[Data record storage]:



At most 30Mb space for data storage. If user needs to record large data, it is suggested to storage in U disk or SD card. The data may be lost if removable storage is larger than 4 GB.

Save cycle of the data	Interval of generate file
0-30 minutes	1 hour generate a file
30 minutes -12hours	1 day generate a file
12hours-15days	1 month generate a file
More than 15days	1 year generate a file

Data Record Tool

View data from “Data record display”.

06-06 11:07:25	26	351	3276	3751	1434	5385	3520	6060
06-06 11:07:19	25	325	2925	0475	3219	3951	8135	2540
06-06 11:07:13	23	276	2300	4950	5194	9060	3452	0221
06-06 11:07:07	22	253	2024	2650	244	3866	4392	2305
06-06 11:07:01	20	210	1540	8855	2504	6028	2440	7387
06-06 11:06:55	19	190	1330	7315	3649	3524	1948	4947
06-06 11:06:49	17	153	969	4845	0349	9077	8549	4575
06-06 11:06:43	16	136	816	3876	5504	4264	9472	1562
06-06 11:06:37	14	105	560	2380	8568	7132	1984	6882
06-06 11:06:31	13	91	455	1820	6188	8564	0388	0434
06-06 11:06:25	11	66	286	1001	3003	8008	9448	3758
06-06 11:06:19	10	55	220	715	2002	5005	1440	4310
06-06 11:06:13	8	36	120	330	792	1716	3432	6435
06-06 11:06:07	7	28	84	210	462	924	1716	3003
06-06 11:06:01	5	15	35	70	126	210	330	495
06-06 11:05:55	4	10	20	35	56	84	120	165
06-06 11:05:49	3	6	10	15	21	28	36	45
06-06 11:05:43	1	1	1	1	1	1	1	1

Figure 1

Click “S” in the left side of the data record display object, display the setting dialog box.

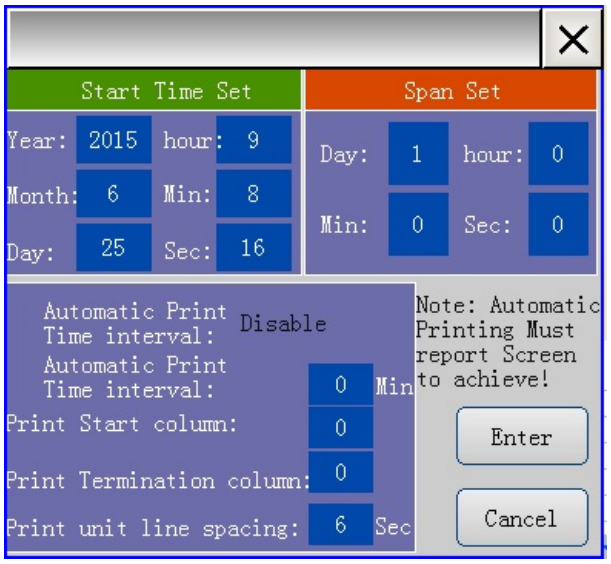


Figure 2

Data log tool

The software record the file in CSV format, The record data exported form HMI is encrypted by default. So need to use the “data log tool”.

Click [Tool]-[Data Log Tool]

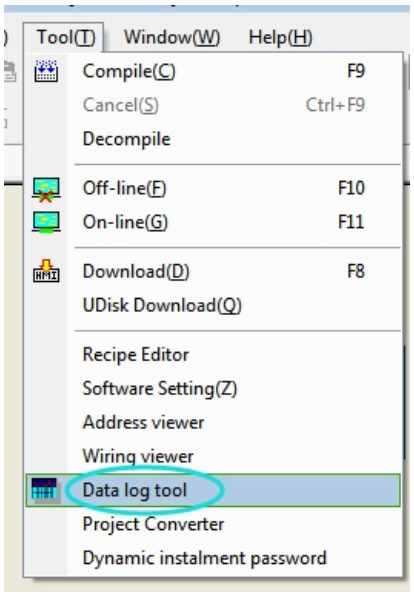


Figure 3

Data Record Transfer

Add “Function Switch” to the screen, and double click to edit its properties.

The record data stored in HMI FLASH or SD card can be transferred to Udisk by function switch.

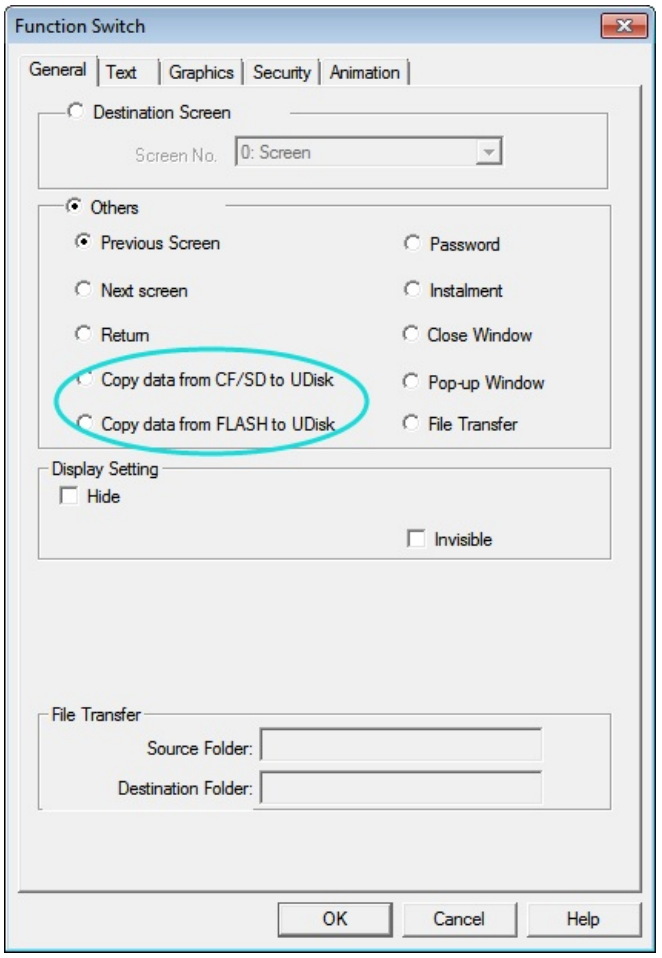


Figure 1

PLC Debugging

The PLC programming software(Running on PC) can be communicated with remote PLC over HMI, to download, upload and monitor the PLC.

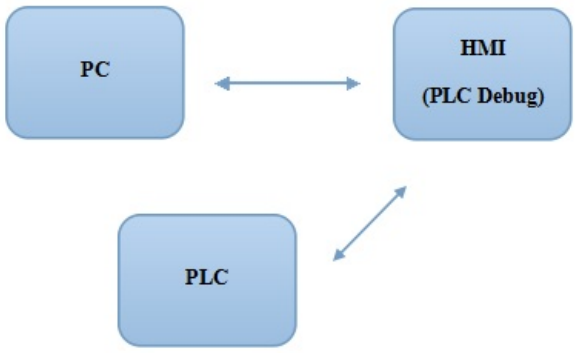


Figure 1

Setting

Requirement:

1. At least 2 COM port on HMI, one is for PLC connection, the other one is for PLC debugging.
2. PLC programming software support communicating with PLC over COM port.

For Example

Mitsubishi PLC:

1. Connect HMI(COM1) with PLC.
2. Connect HMI (COM2) with PC (PLC programming software), the communication parameters of both COM ports should be same.
3. Set the wait timeout.

Notice: It may be disconnected during communication, when PLC debugging is enabled.

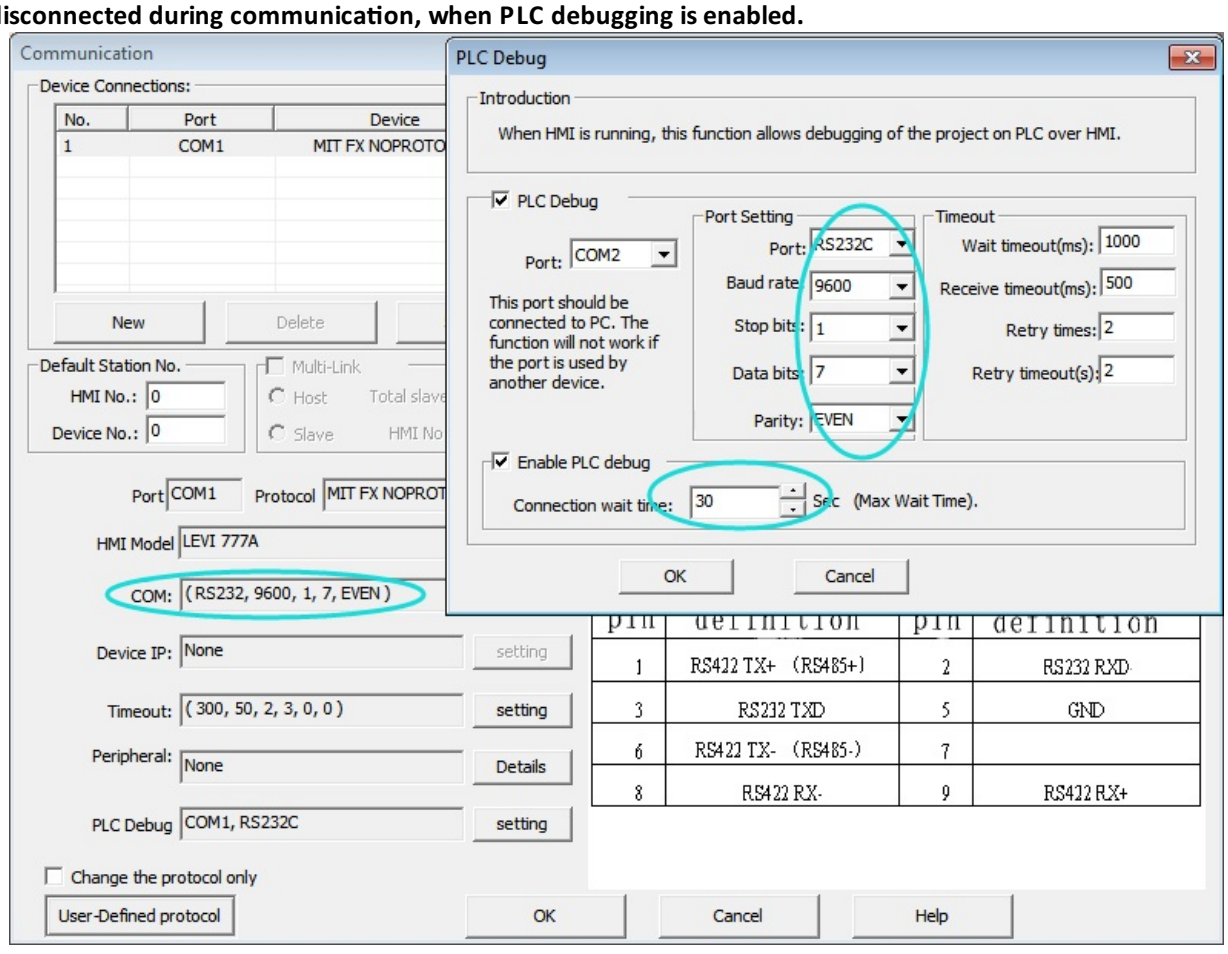


Figure 2

Siemens S7-200 Smart Ethernet

General

The driver is designed for Siemens S7-200 Smart series PLC.

Addresses

Type	HMI address	PLC address	Format	Range	note
Word address	IW	I	IWn	0 ~ 99999	
	QW	Q	QWn	0 ~ 99999	
	AIW	AI	AIWn	0 ~ 99999	
	VB	AQ	VBn	0 ~ 99998	
	VW	V	VWn	0 ~ 99998	VW0=VB(0~1) VW2=VB(2~3) Even address
	VD	V	VDn	0 ~ 99998	VD0=VB(0~3) VD4=VB(4~7) The addresses need to be the multiply of 4
	MB	M	MBn	0 ~ 99999	
	MW	M	MWn	0 ~ 99999	MW=MB(0~1) MW2=MB(2~3) Even addresses
	MD	M	MDn	0 ~ 99999	MD0=MB(0~3) MD4=MB(4~7) The addresses need to be the multiply of 4
	TW	T	TWn	0 ~ 99999	Current value on timer
	CW	C	CWn	0 ~ 99999	Current value on counter
	SW	W	SWn	0 ~ 99999	
Bit address	I	I	In.x	0.0 ~ 99999.7	
	Q	Q	Q	0.0 ~ 99999.7	
	VW-BIT	V	VW-BITn.x	0.0 ~ 99999.7	
	V	V	Vn.x	0.0 ~ 99998.7	
	M	M	Mn.x	0.0 ~ 99999.7	
	SM	SM	SMn.x	0.0 ~ 99999.7	
	S	S	Sn.x	0.0 ~ 99999.7	Read only
	T	T	Tn	0 ~ 99999	Timer state, Read-only
	C	C	Cn	0 ~ 99999	Counter state, Read-only

Setting

Choose the correspond protocol for PLC, at [Setting]-[Communication]-[setting].

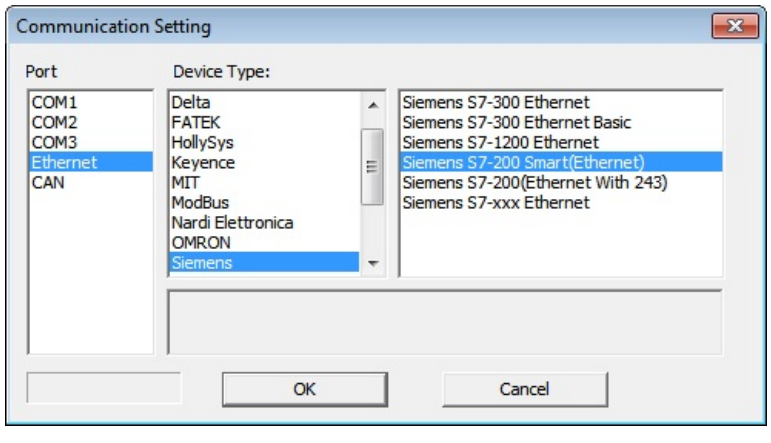


Figure 1

Set the Ethernet data.

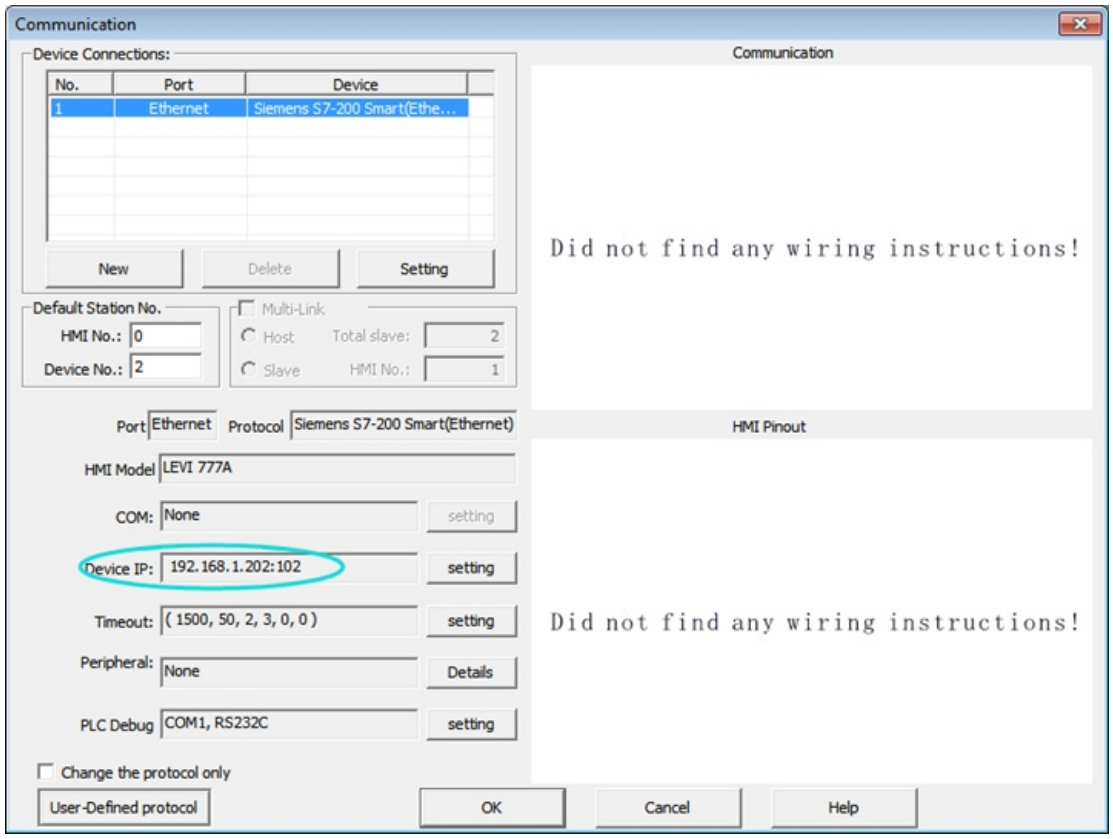


Figure 2

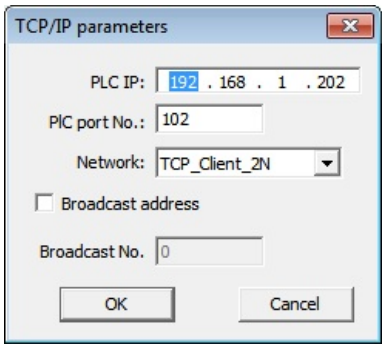


Figure 3

PLC IP: device IP

PLC port No.:102(fixed)

Network:TCP_Client_2N (fixed)

Wait time:the internet status(=1500ms)

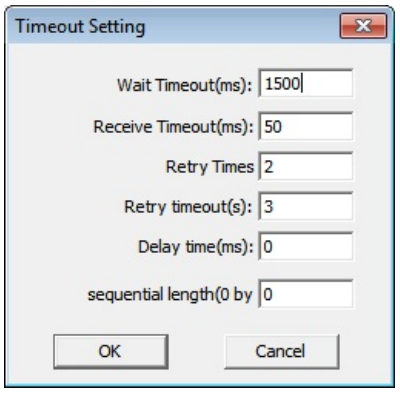


Figure 4

Siemens S7-300 MPI protocol

General

The RS485 cable connection is shown as figure 1. Communication parameter: 185000,1,8,Even; Station No.: 2.

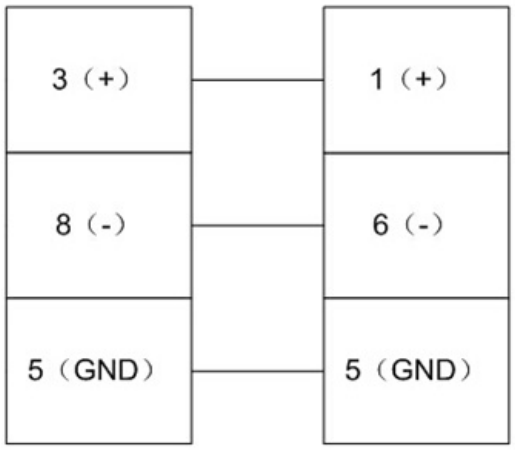


Figure 1

Address

IW address format is shown as following, QW and MW address setting is the same.
Word: IW0, IW2, IW4, IW6, IW8…… corresponds to: I0.0~I1.7(IW0),I2.0~I3.7(IW2),I4.0~I5.7(IW4),….
Dword: IW0, IW4, IW8, IW12, IW16…

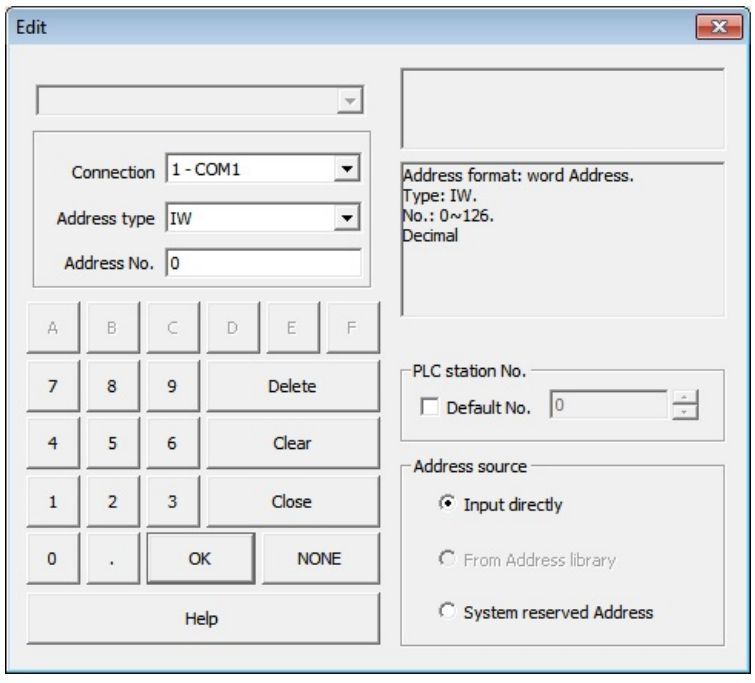


Figure 2

DBn Address:
BBB
BD data (0~255);
DDDD=
Corresponds to BD addresses (0~8192);
Shown as following BBBDDD = 1000095, 100 is data block, 0095 is address.

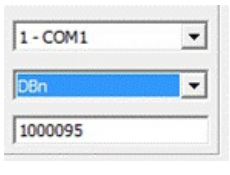


Figure 3

The data filled with zeros shown as 0120005 which represent the address 0005 in data block 012.

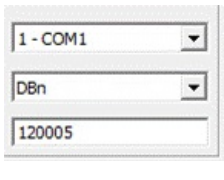


Figure 4

DB##DBWxxxxxx:## is data block number (0~255), xxxxx represent data block range (0~8192), shown as figure 5, 01 is data block, 4 is the address correspond to it.

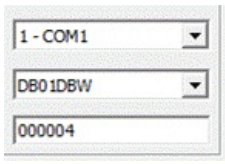


Figure 5

Siemens S7-300 Ethernet

Type	HMI address	PLC address	Format	Range	Note
Word address	IW	I	IWn	0 ~ 99999	
	QW	Q	QWn	0 ~ 99999	
	MB	M	MBn	0 ~ 99999	
	MW	M	MWn	0 ~ 99999	MW=MB(0~1) MW2=MB(2~3) Even address
	MD	M	MDn	0 ~ 99999	MD0=MB(0~3) MD4=MB(4~7) The addresses need to be the multiply of 4
	DBxDBB	DB0.DB~DB99.DB	DBxDBBnnxxxx	000000~999999	nn: block number, xxxx: address
	DBxDBW	DB0.DB~DB99.DB	DBxDBBnnxxxx	000000~999999	nn: block number, xxxx: address
	DBxDBD	DB0.DB~DB99.DB	DBxDBBnnxxxx	000000~999999	nn: block number, xxxx: address
Bit address	I	I	In.x	0.0 ~ 99999.7	
	Q	Q	Q	0.0 ~ 99999.7	
	M	M	Mn.x	0.0 ~ 99999.7	
	DBxDB	DB0.DB~DB99.DB	DBxDBnnxxxx.y	0.0 ~ 99999.7	nn: block number, xxxx: address

Driver selection.

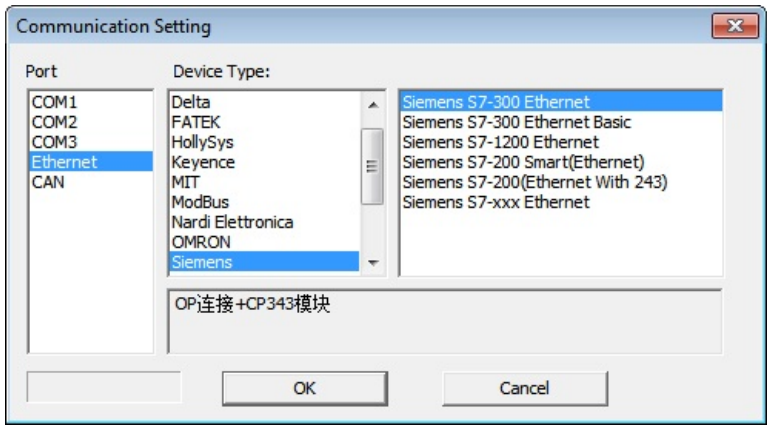


Figure 1

Internet parameter.

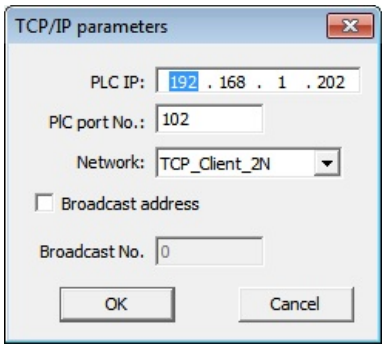


Figure 2

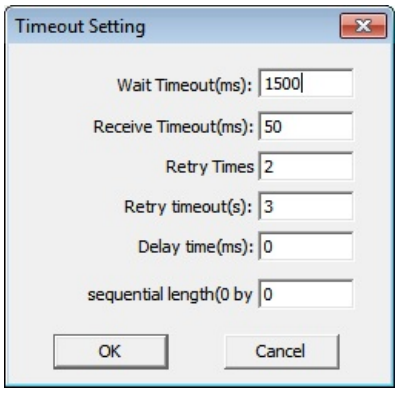

Timeout Setting
Wait Timeout(ms): 1500
Receive Timeout(ms): 50
Retry Times 2
Retry timeout(s): 3
Delay time(ms): 0
sequential length(0 by 0
OK
Cancel

Figure 3

Siemens S7-1200 Ethernet

Settings

Select [Siemens S7-1200 Ethernet] in [setting]-[communication]- [Setting].

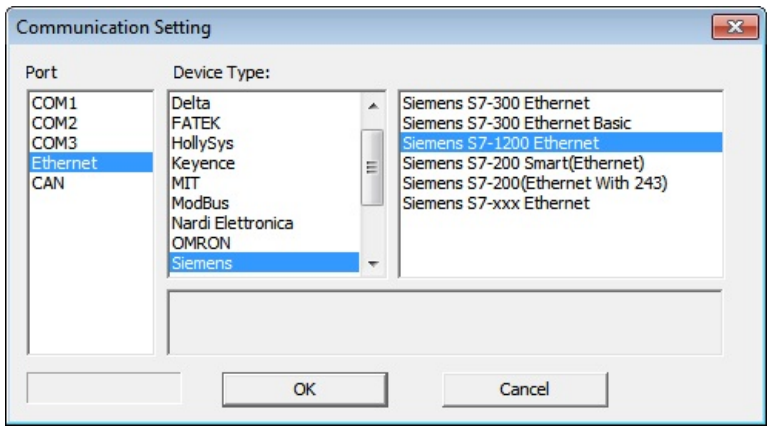


Figure 1

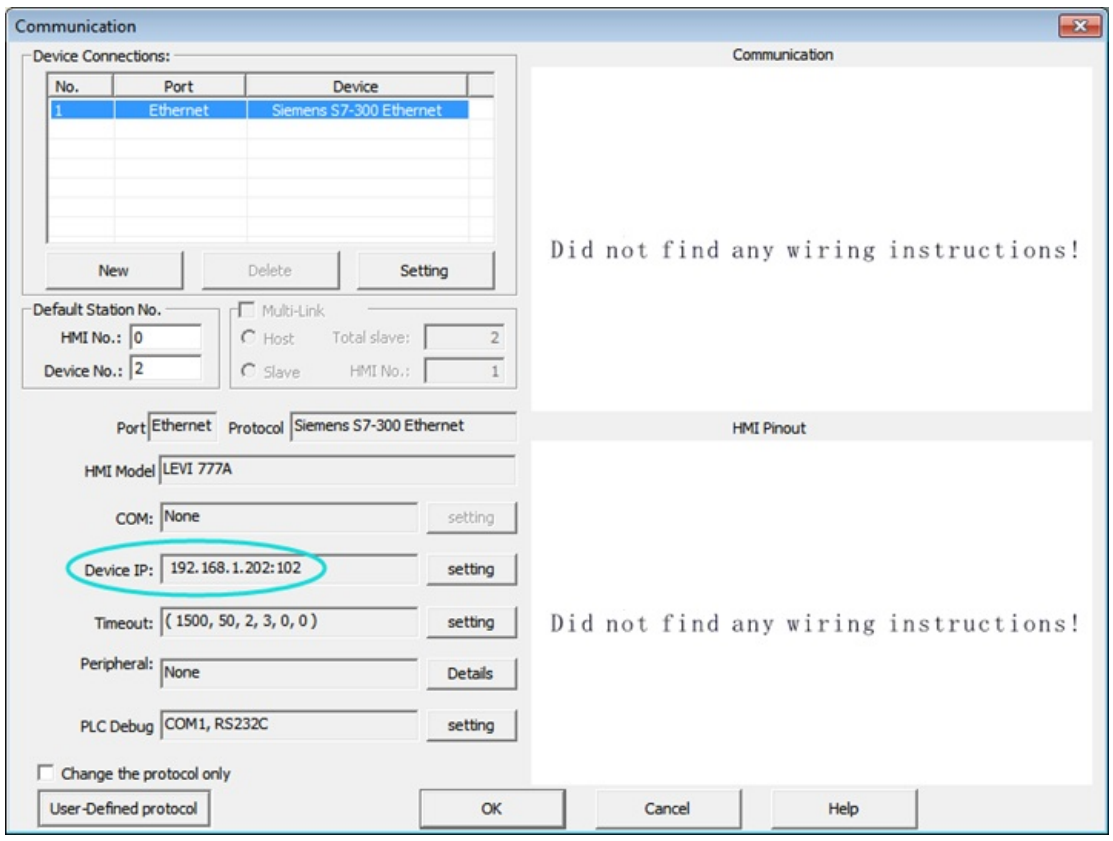


Figure 2

Connection

S7-1200 support maximum 3 device connection.

HMI access with PLC with S7 protocol, TSAP is 02.01.

DB access

Offset is DB2 address

To access the B1 data in DB2, use addresses DBxDBB2xxxx, DBxDBW2xxxx, DBxDBD2xxxx.

Represent block: 2xxxx

For example: DBxDBB20000 = DD2.DBB0

DBxDBB20001 = DD2.DBB1

User-defined Protocol

Write and Read

Application 1: according to Modbus protocol, HMI send 01 03 00 00 00 04 44 09, 01: station number, 03: write, 00 00: index address, 00 04: data length, 44 09: CRC code.
The device will return 01 03 08 00 24 00 00 00 06 00 07 50 16, 01: station number, 03: write, 08: valid data length, 50 16: CRC code.
The specific settings are shown as following.

Step 1: Add checking, in [check] setting.

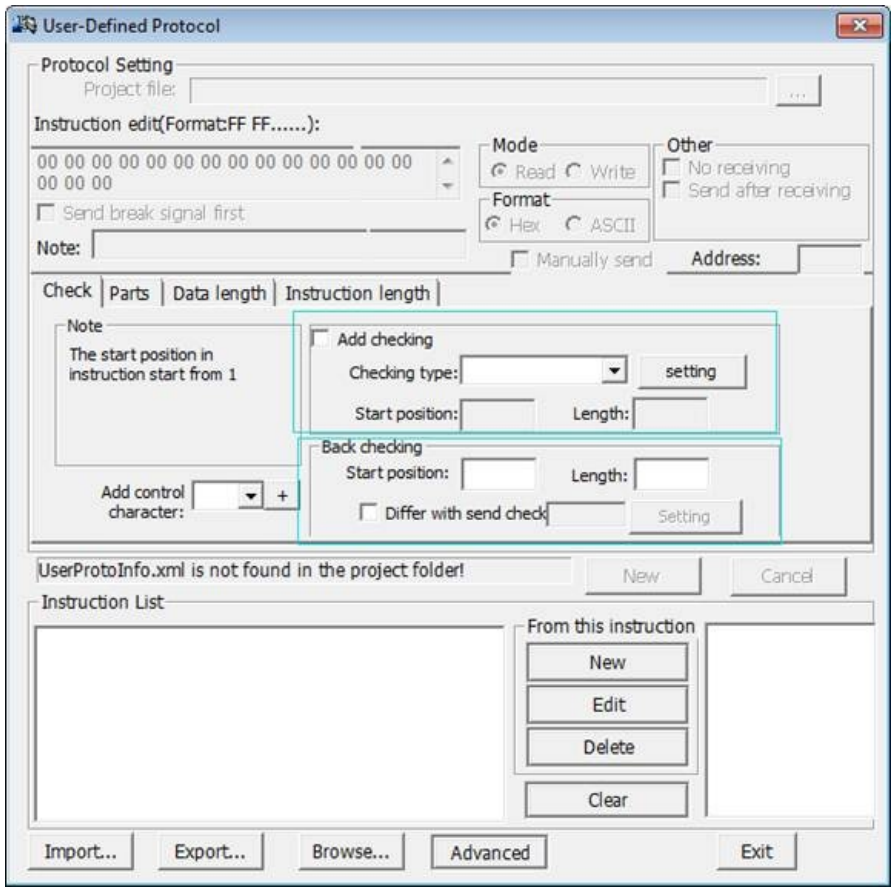


Figure 1

Step 2: set address details in [part] setting.

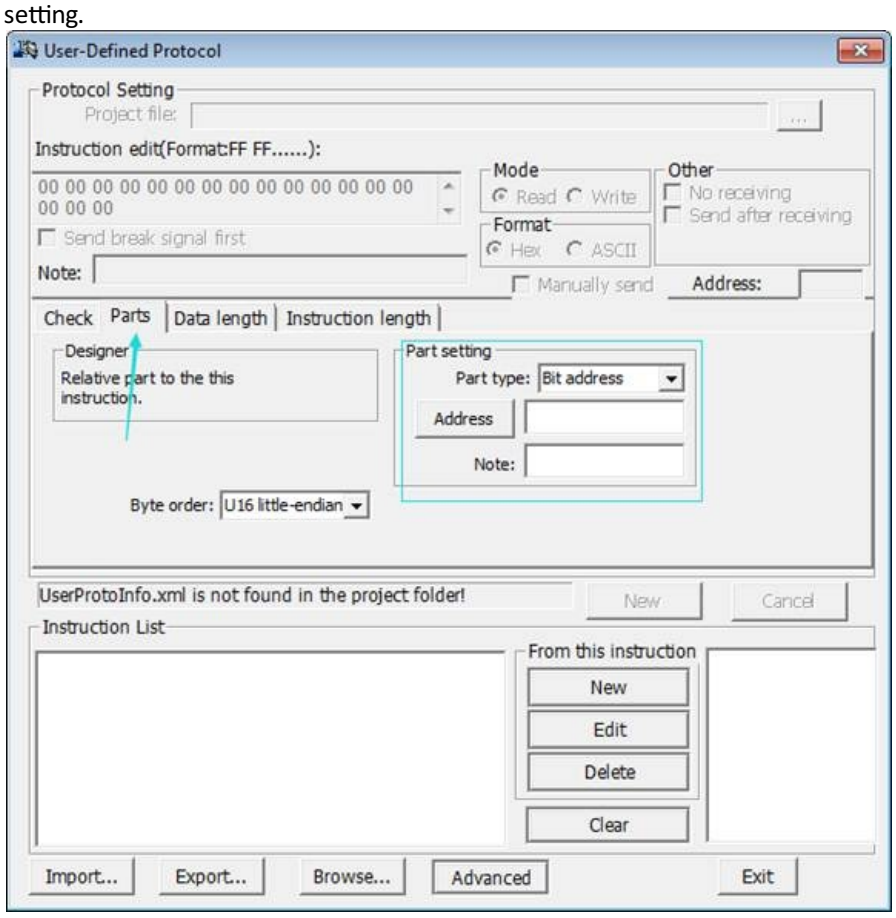


Figure 2

Step 3:set return data position and length in [data length].

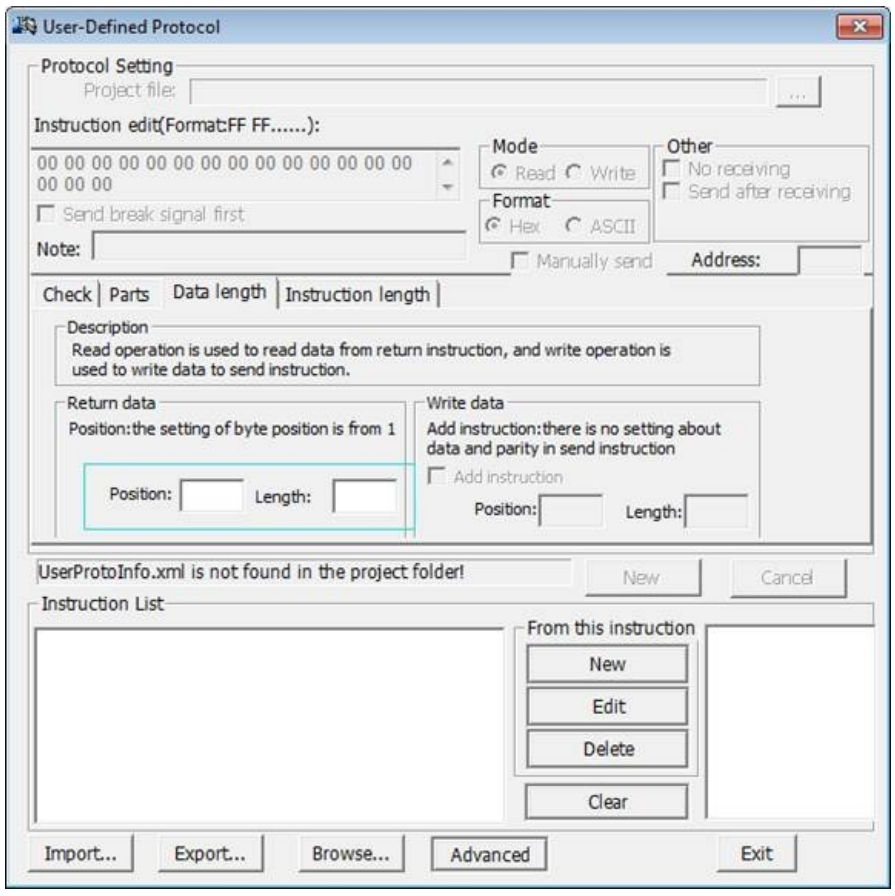


Figure 3

Step 4:set the return data length in [instruction length].

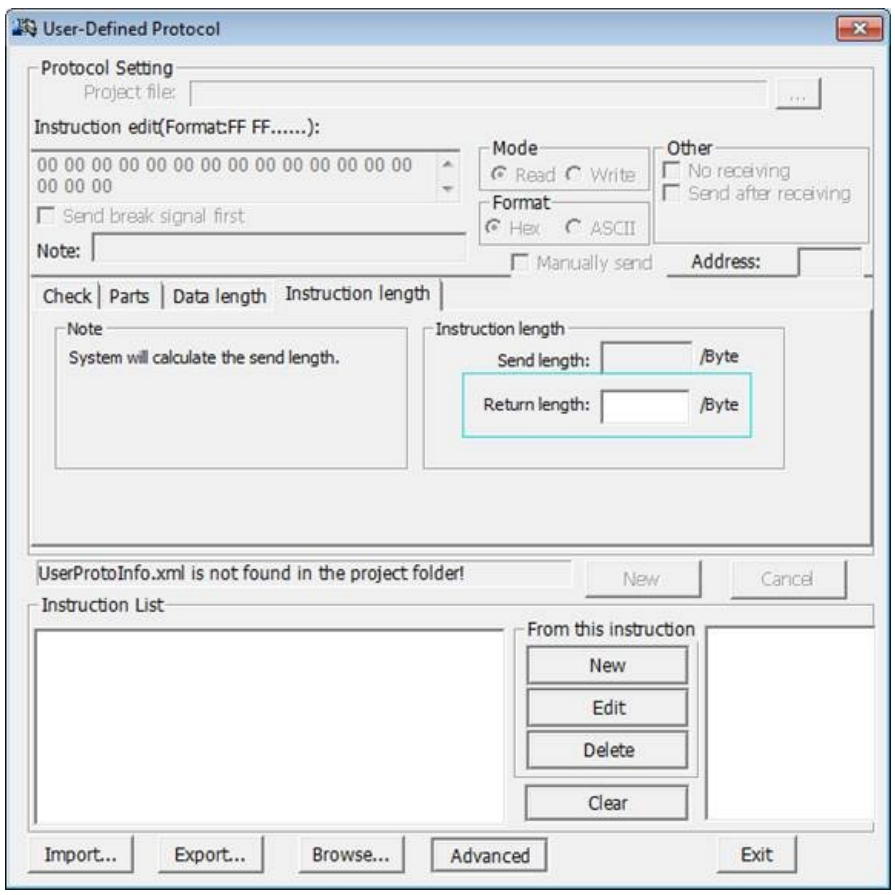


Figure 4

Case 2: Based on Modbus protocol, HMI send: 01 06 00 01 00 38 D9 D8. 01: station number, 06: write, 00 01 index address, 00 38: acquired data, D9 D8: CRC code.

Device responds same command: 01 06 00 01 00 38 D9 D8, means write successful.

Step 1: set start position and length in [checking]-[back checking].

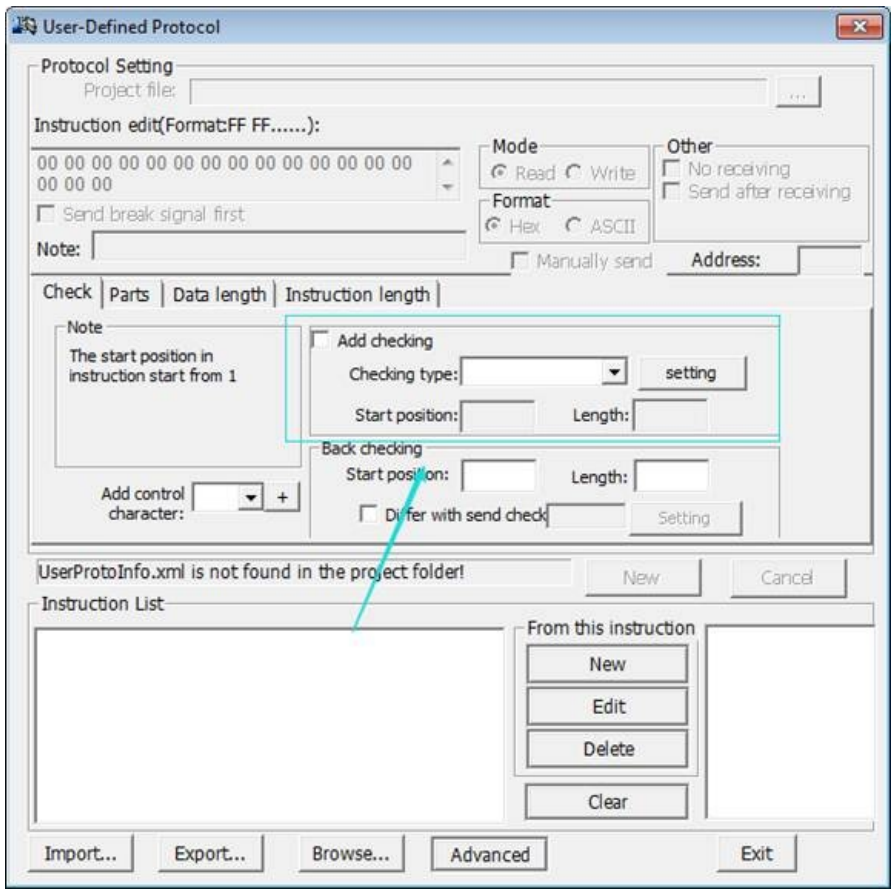
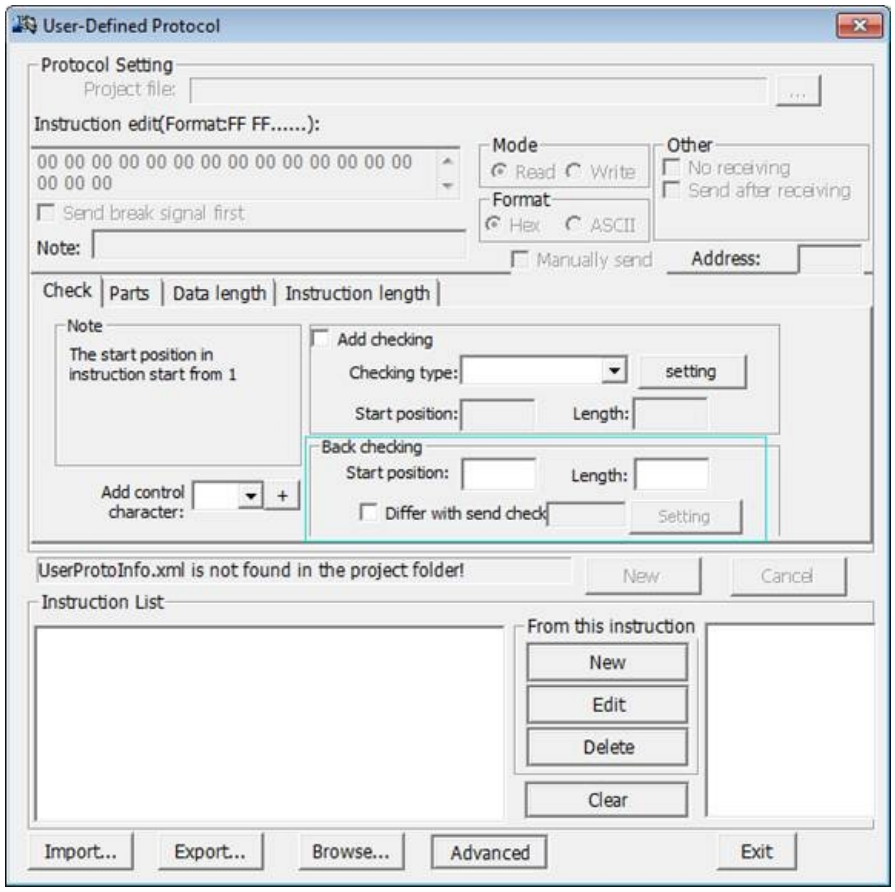


Figure 5

Step 2: set write data in [data length].



Manually send

“manually send”: the command will send while the address is triggered.

Case 1: assign manually send to a command, control bit is 12 (range: 0-9999).

[oncectrlbit] can be found in address setting in [user-defined protocol].

Figure 6

Send after receiving

Send after receiving will applied when the HMI is used as a Client device. The settings shown down below.

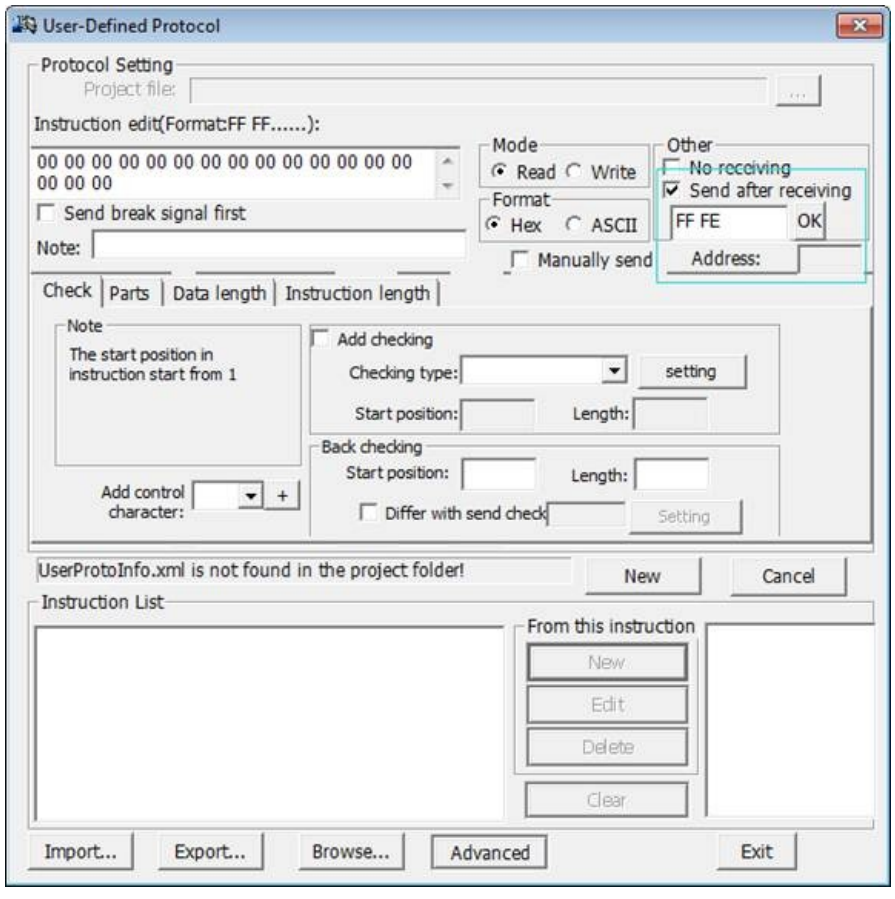

User-Defined Protocol
Protocol Setting
Project file:
Instruction edit(Format:FF FF.....):
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00
Send break signal first
Note:
Mode
Read
Write
Format
Hex
ASCII
Manually send
Other
No receiving
Send after receiving
FF FE
OK
Address:
Check
Parts
Data length
Instruction length
Note
The start position in instruction start from 1
Add checking
Checking type:
setting
Start position:
Length:
Back checking
Start position:
Length:
Add control character:
Differ with send check
Setting
UserProtoInfo.xml is not found in the project folder!
New
Cancel
Instruction List
From this instruction
New
Edit
Delete
Clear
Import...
Export...
Browse...
Advanced
Exit

Figure 7

General

If the device does not support MODBUS standard, and the protocol is not list in HMI compile software, then user can define the protocol by following instruction to realize simply communication functions like send and receive commands.

Settings

1.Choose [other protocol]–[user defined protocol].

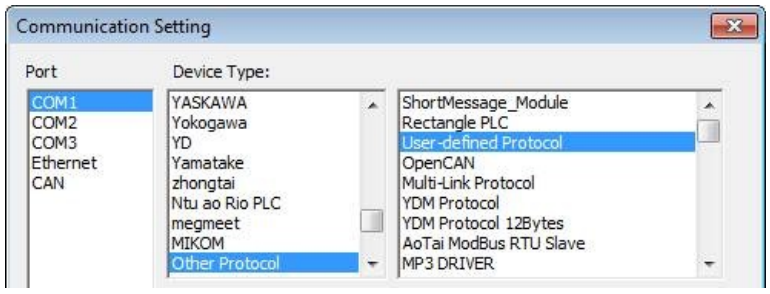


Figure 1

Press [communication]-[user-defined protocol].

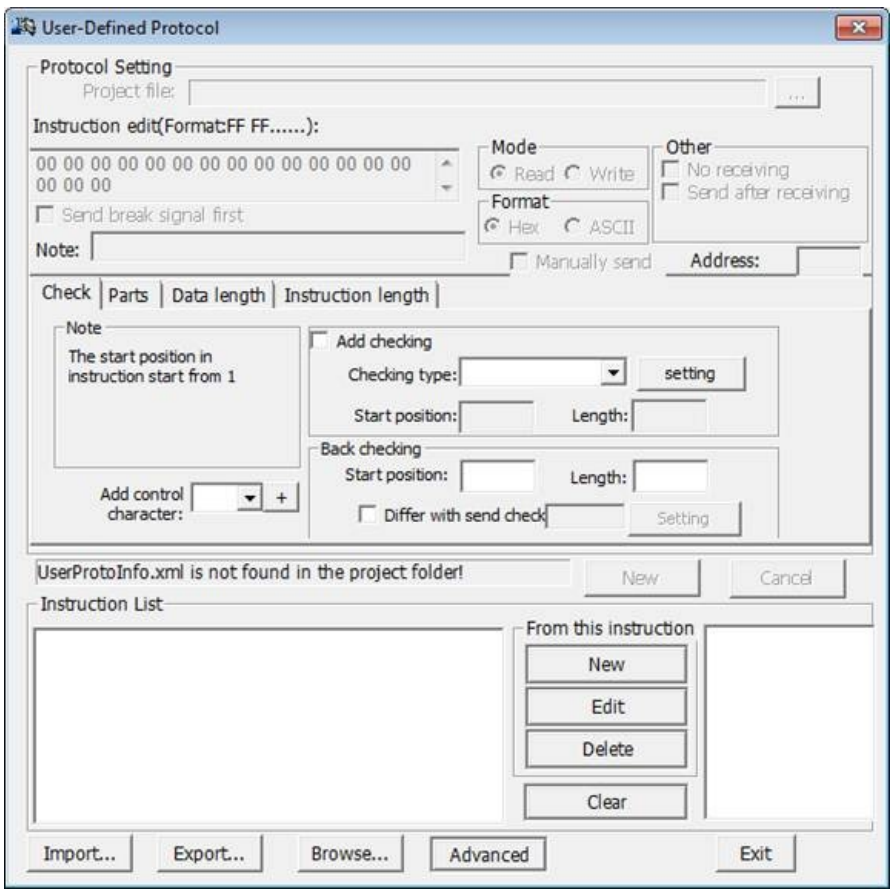


Figure 2

Properties	Description
Instruction edit	The required command.
Mode	Write to address or read from the address.
Format	Encode format: HEX or ASCII.
Other	No receiving: does not respond to the receiving command.
	Send after receiving: respond to the receiving command.
Manually send	Respond once after trigger the address.
	Address: set the trigger address.
Add	Save this setting.
Cancel	Cancel current settings.
Add	Add a new command.
Edit	Edit the designated command.
Delete	Delete the designated command.
Clear	Clear all the commands.
Instruction list	Browse all current commands.
Parts list	Browse all the parts added.
Import	Import the command files to the instruction list.
Export	Export current command settings to local storage.
Browse	Browse local command files.
Advanced	Combine two commands.
Exit	Complete editing and exit setting.

Press [Add] to create a new command.

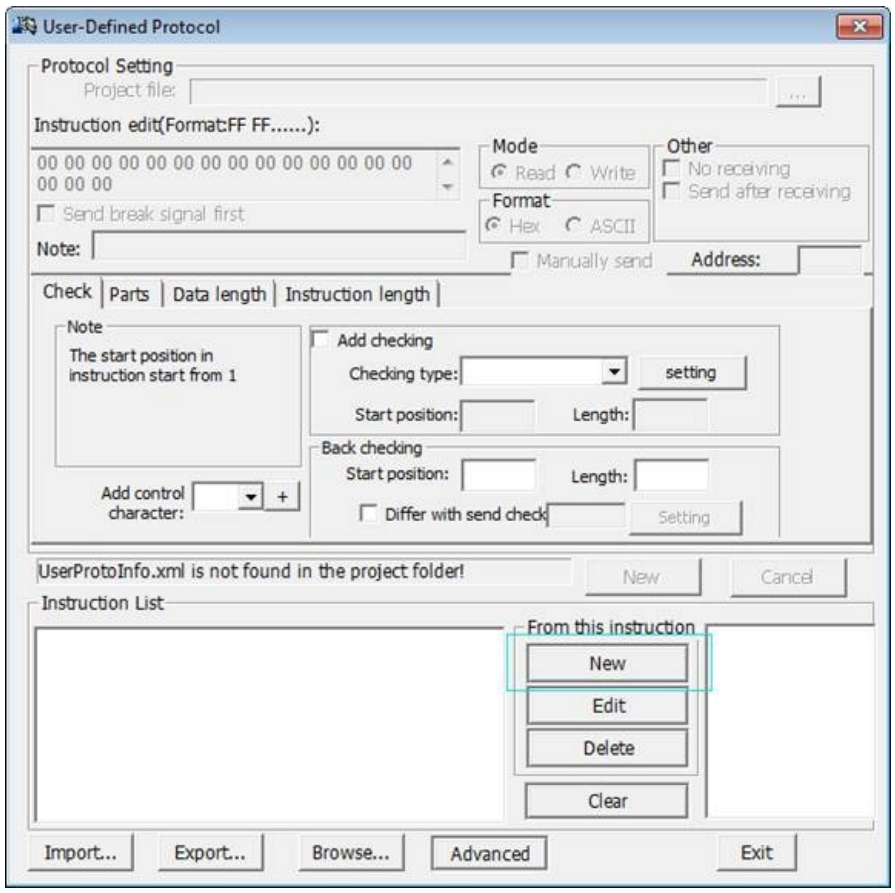


Figure 3

Select the Mode as below.

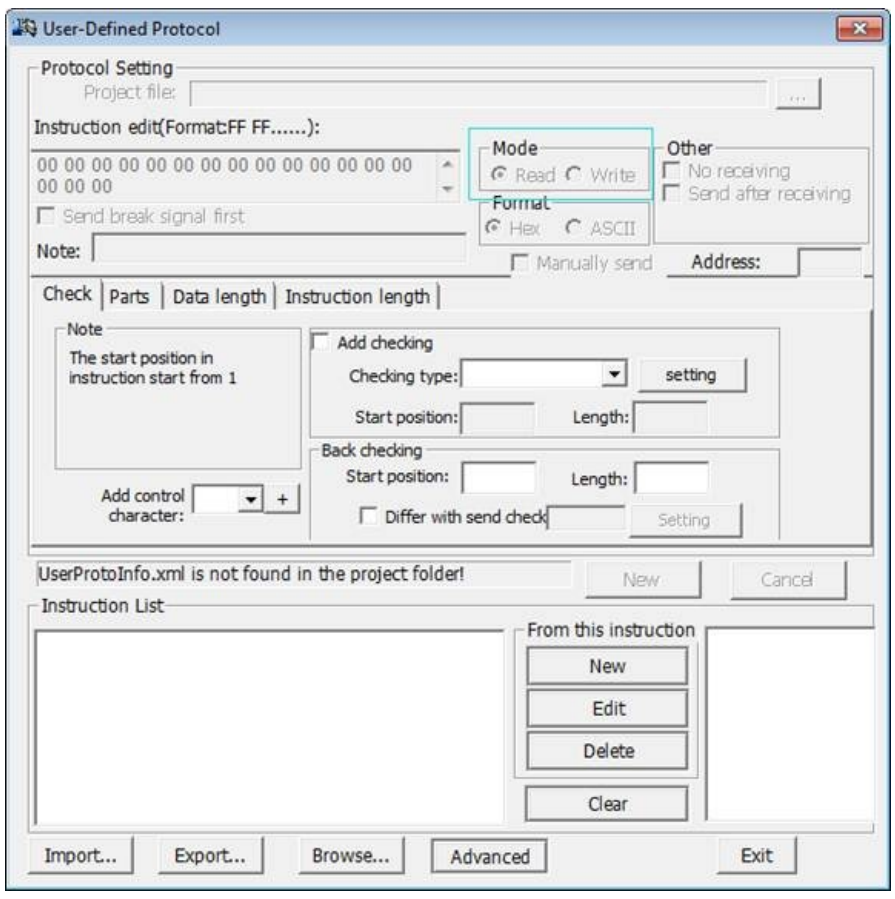


Figure 4

Choose the data format as below.

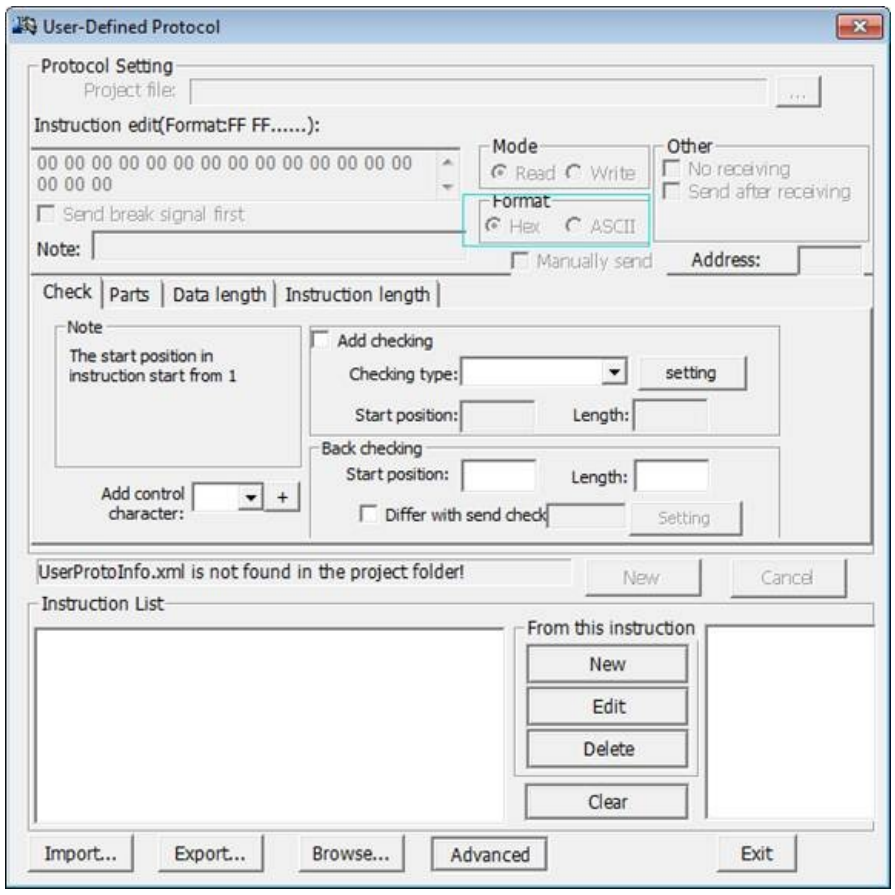


Figure 5

Instruction edit, when the data format is HEX, use two numbers to represent one 16bit number. ASCII format using characters to input, shown as below.

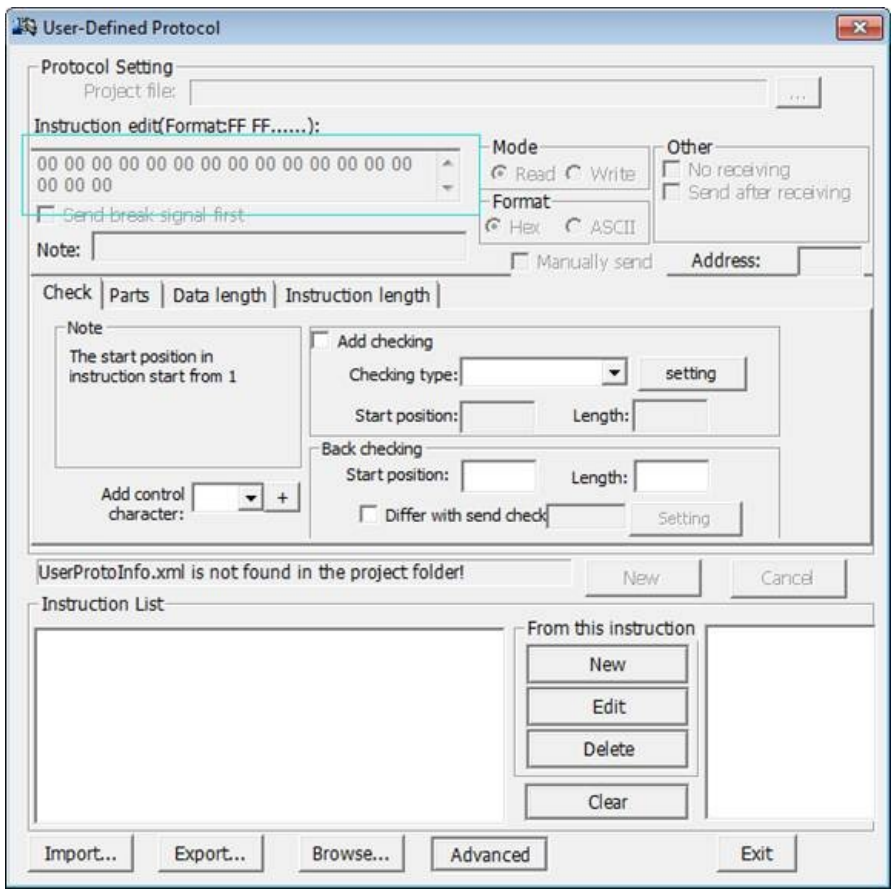


Figure 6

Check settings, shown as below.

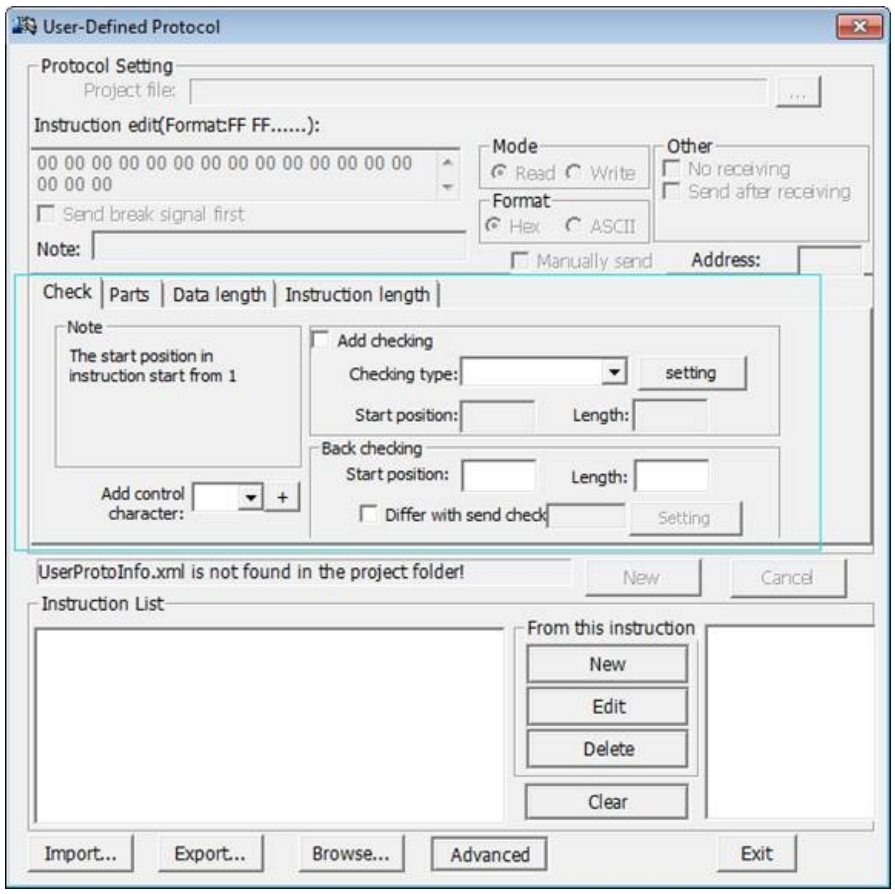


Figure 7

Properties	Description
Add checking	Add checking command when receiving the data.
Back checking	Start position: select the start position of the data which need to check.
	Length: The data length need to be checked.
Add control character	Add control character in ASCII format.
Differ with send check	Set return checking.

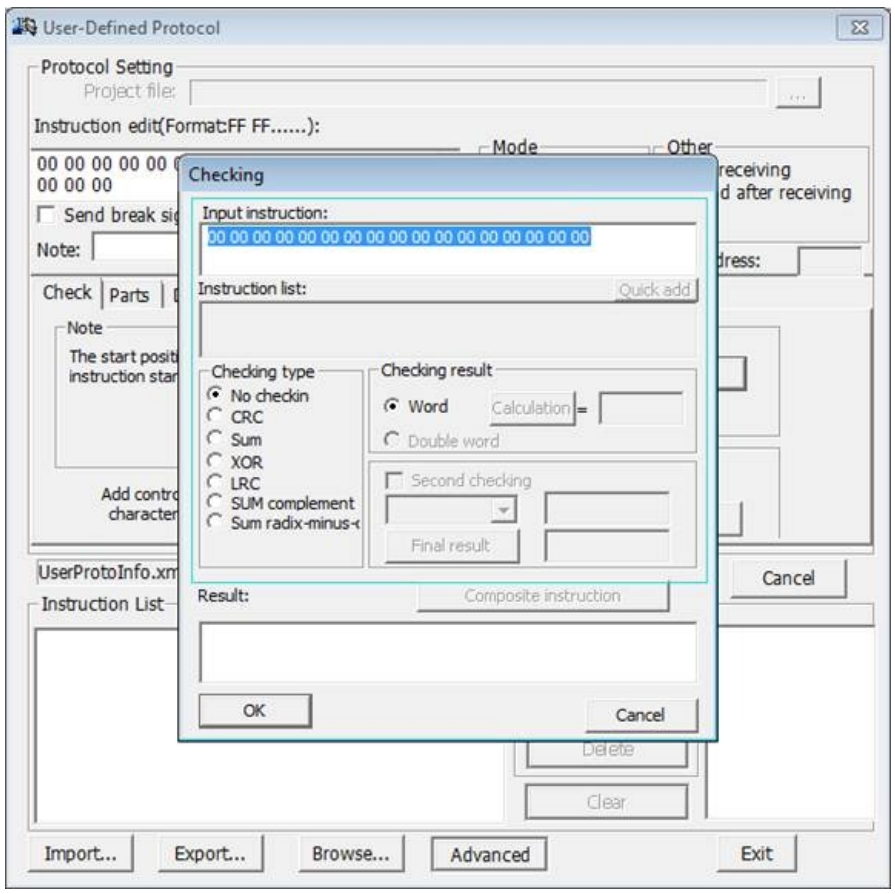


Figure 8

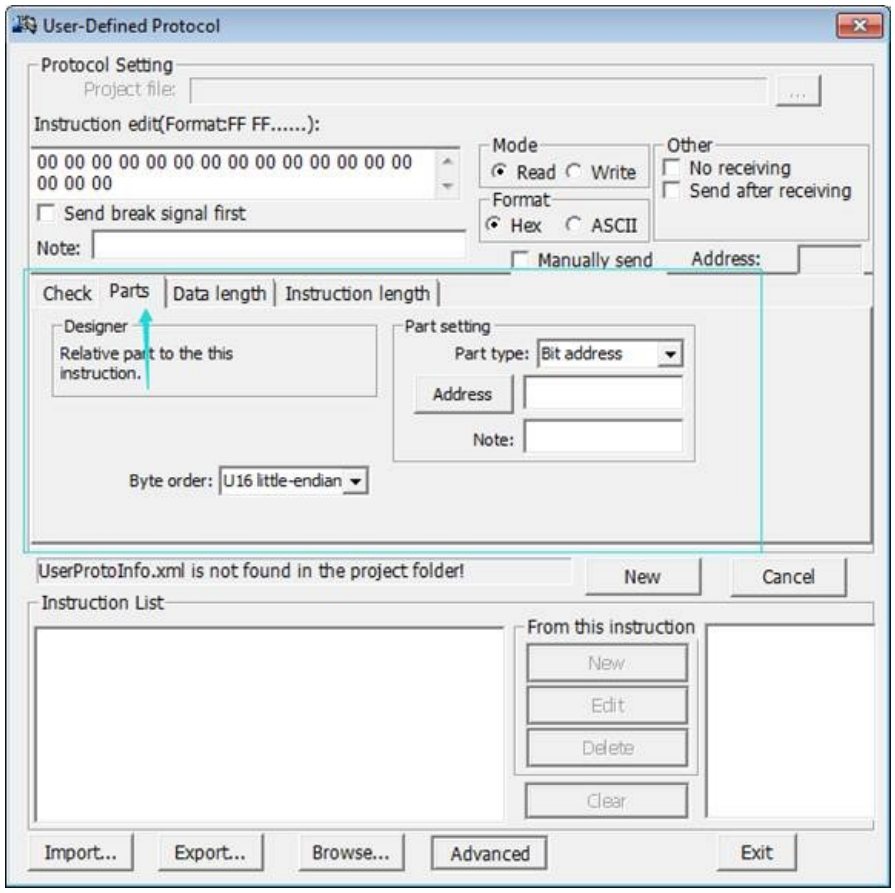


Figure 9

Properties	Description
Part Setting	Part type: bit or word address.
	Address: the triggering address.
	Note: description to the part.
Byte order	The numerical display order.

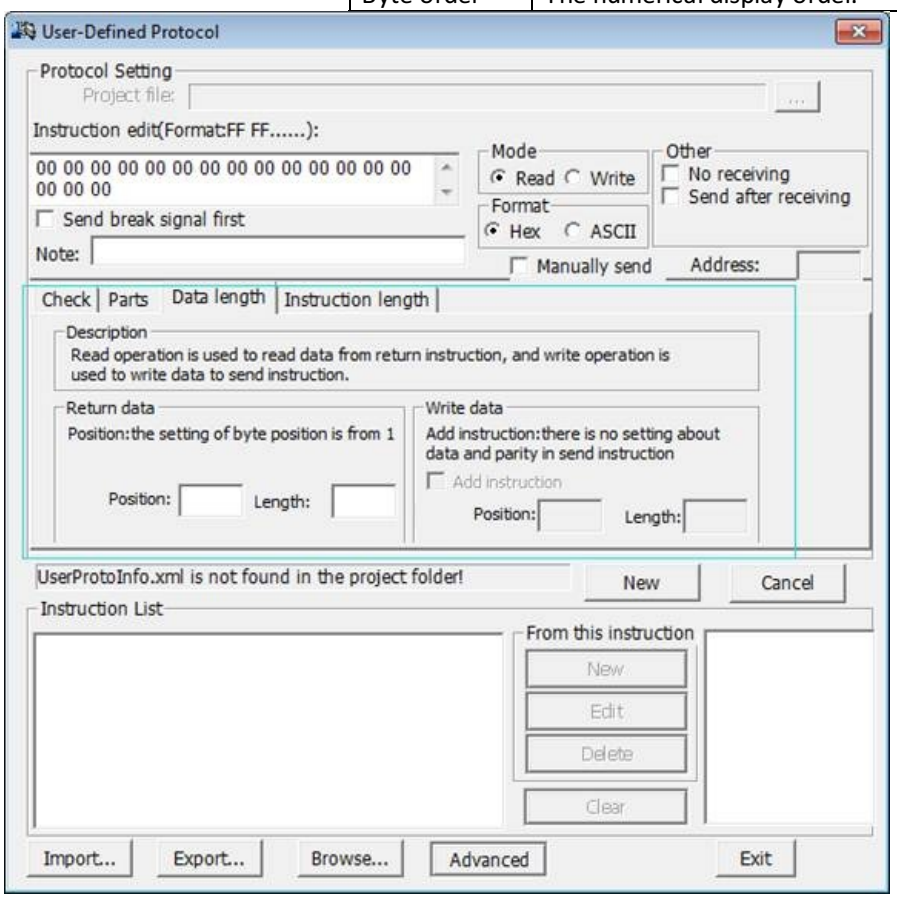


Figure 10

Properties	Description
Return data	Position: the start position of the return data.
	Length: the return data length.
Write data	Position: the start position of the write data.
	Length: the write data length.

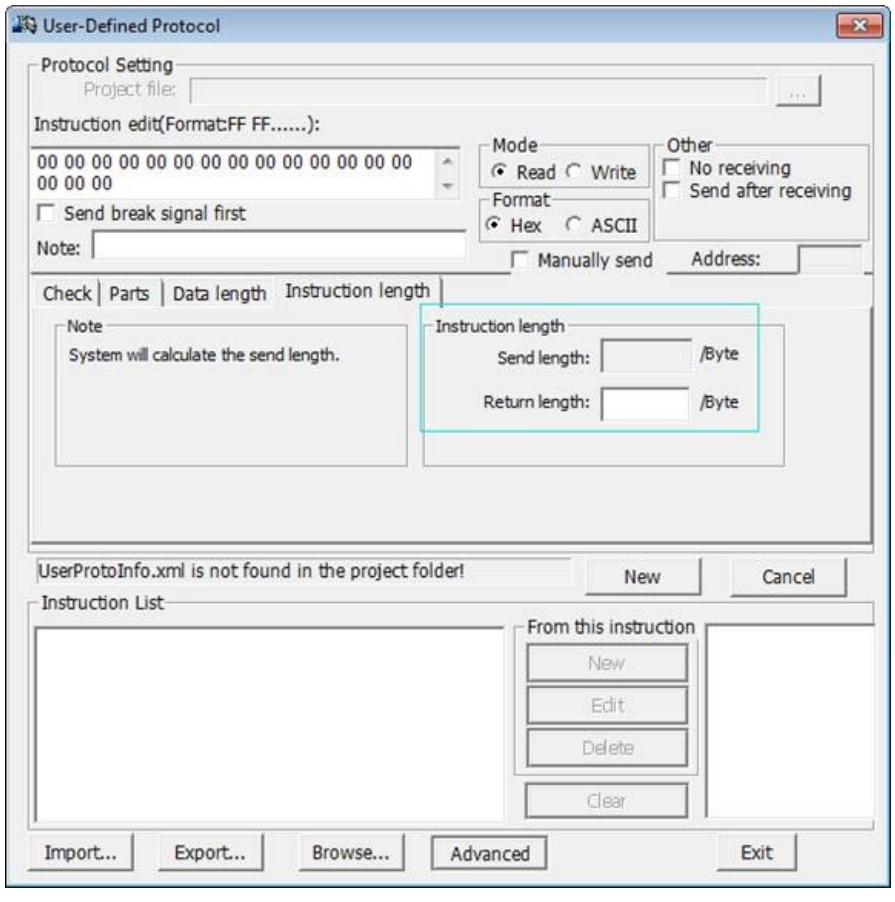


Figure 11

Properties	Description
Send length	The instruction data will only send designated data length.
Return length	The responds data length.

OpenCAN settings

General

Opencan is based on CAN2.0 standard; user can customize the protocol settings.

Instruction

Choose [setting] on menu bar- [communication]-[setting], select OpenCAN, shown as following.

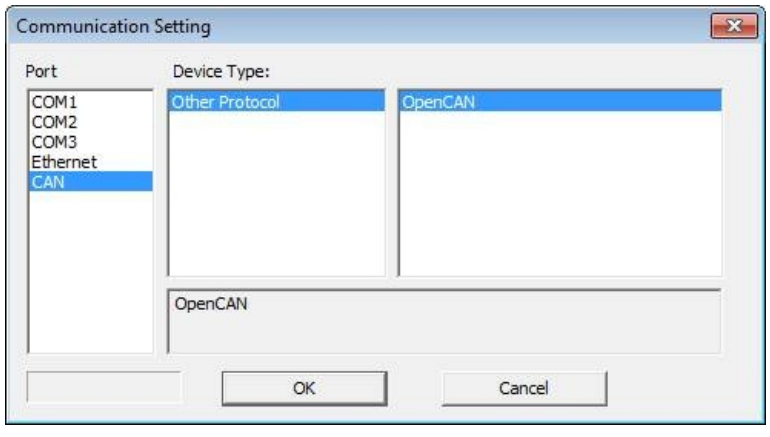


Figure 1

Settings

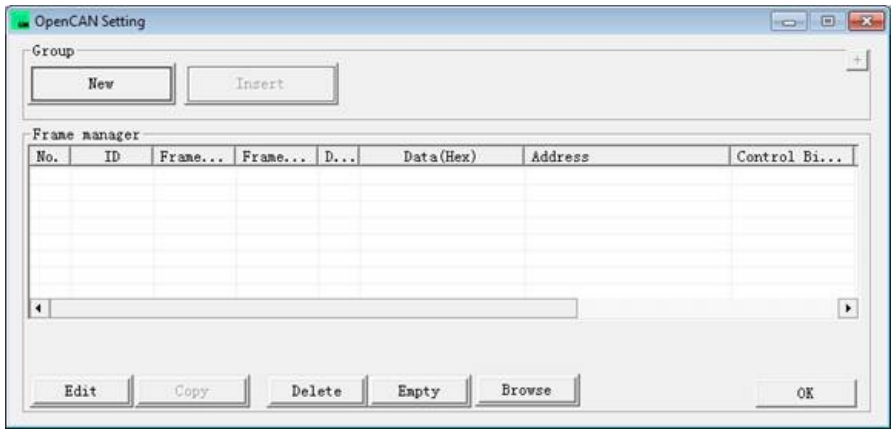


Figure 2

Properties	Description
Add	Add a frame for designated address
Insert	Select the position to insert a frame.
Frame manager	Browse current frame settings.
9 Browse	Show the frame setting files in xml format.

Select [Add] and the following settings will show up, as figure 3.

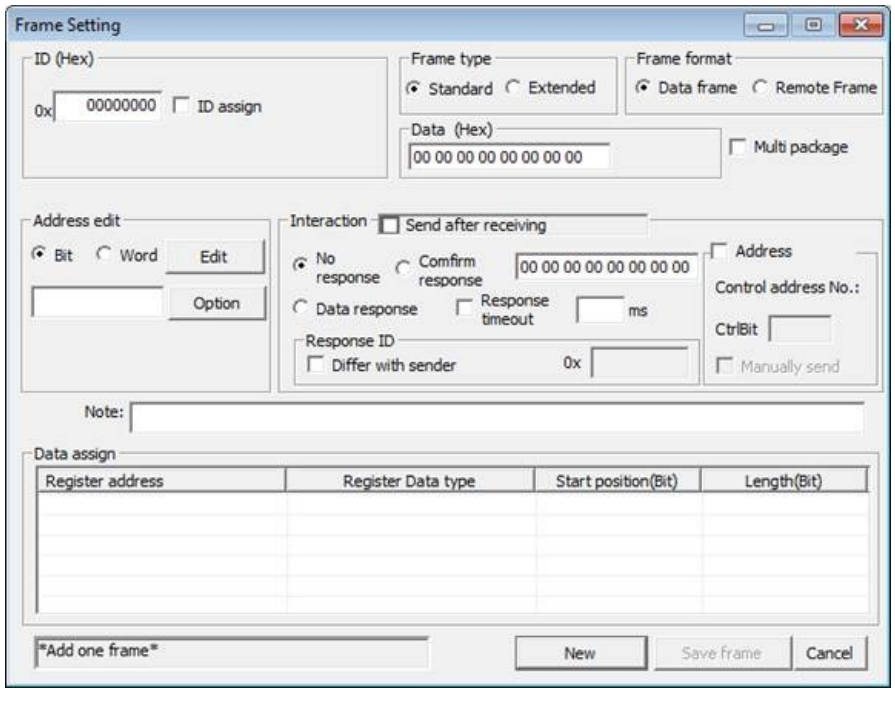


Figure 3

Properties	Description
Frame type	Select between Standard frame and Extended Frame.

Frame format	Select between Data frame and Remote Frame.
Data (Hex)	Use two numbers to represent one 16bit number, separate with blank.
use address	Select address type between bit and word.
Option	Relate the frame and the designated address.
Interaction	HMI send frame and the devices process and respond.
send after receiving	HMI will process and respond after receiving the command.
No response	HMI or devices will not receive any responds
Confirm response	The HMI or devices will check the data received then respond to it.
Data response	HMI or devices will respond the designated data when receiving the frame.
Response ID	The response ID can differ with the sending ID by this setting.

Control address setting shown as below

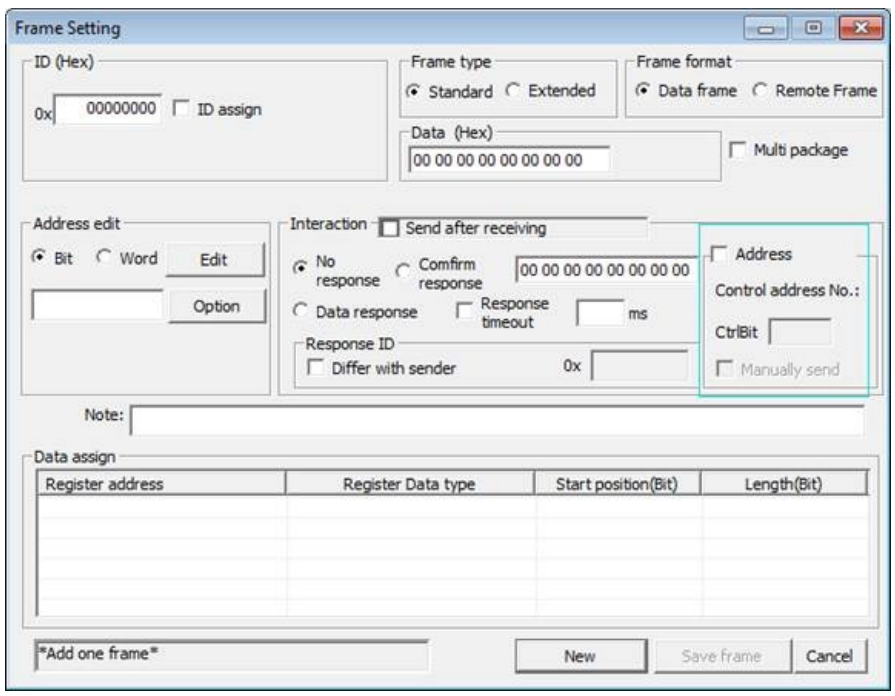


Figure 4

Properties	Description
Address	Send when the designated address value is not 0.
Ctrlbit	CtrlBit address range is 0~255, the CAN command will only work when the address value is 1.
Manually send	Send once for each time the address is triggered.

Modbus All Faction

Modbus Function Code

Notice: In Modbus protocol, allow to access bit address like 100.1, which using the same function code with 16bits read-write.

Word Address	Function Code(HEX)	Data Length	Read or Write
3	04 (Read input address)	16-bits	Read only
	06 (Write single holding address)		
	10 (Write multiple coils)		
4	03 (Read multiple holding addresses)	16-bits	v
	06 (Write single holding address)		
	10 (Write multiple coils)		
W6	03 (Read multiple holding addresses)	16-bits	v
	06 (Write single holding address)		
	10 (Write multiple coils)		
W16	03 (Read multiple holding address)	16-bits	v
	0F (Write multiple addresses)		

Bit Address	Function Code(HEX)	Data Length	Read or Write
0	01(Read coils)	16-bits	v
	05(Write single coil)		
	0F(Write multiple coils)		
1	02(Read discrete inputs)	16-bits	Read only
	05(Write single coil)		
	0F(Write multiple coils)		
W5	01(Read coils)	16-bits	v
	05(Write single coil)		
	0F(Write multiple coils)		
W15	01(Read coils)	16-bits	v
	0F(Write multiple coils)		

Address Mapping (Modbus RTU Master)

Modbus will read the value from HMI local address:

HMI Local Address	Modbus Address
HDX3000.0~HDX3499.15	Bit address type:0 (Range: 0~7999)
HDW3500~HDW7999	Word address type: 4 (Range: 0~4499)

Bit Address Mapping:

HDX3000.0 = Modbus Address 00(Address Type: 0; Address No.:0)
HDX3000.15 = Modbus Address 015(Address Type: 0; Address No.:15)
HDX3001.1 = Modbus Address 017(Address Type: 0; Address No.:17)

Word Address Mapping:

HDW3500 = Modbus Address 40(Address Type: 4; Address No.:0)
HDW3615 = Modbus Address 4115(Address Type: 4; Address No.:115)
HDW4500 = Modbus Address 4100000(Address Type: 4; Address No.:1000)

Secure Startup

If user got the problem with HMI stuck when startup, there are may some problem with HMI project, so user need to download the HMI project again.

Step1:

turn off HMI (disconnect with power supply of HMI).

Figure 1

step 2:

keep pressing on the left bottom corner of HMI (Hold on).

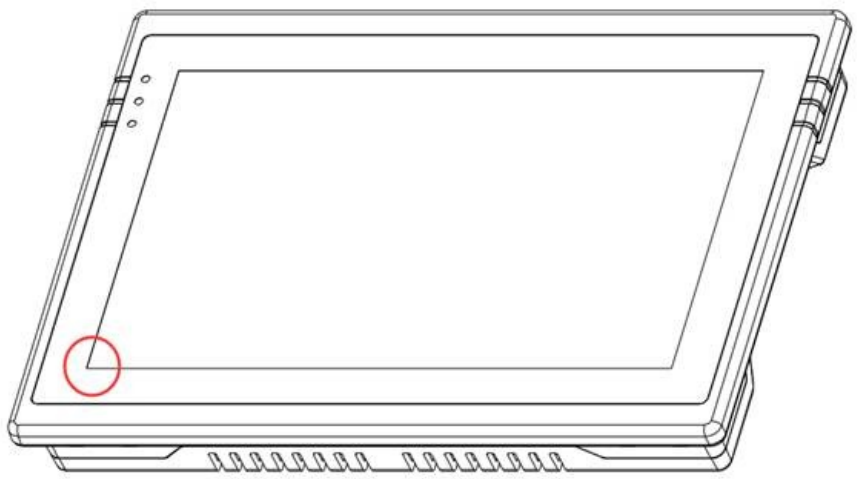


Figure 2

Figure 3

Step 3

Turn on HMI (connect with power supply of HMI) with keeping pressing on left bottom corner of HMI (Step2).

Figure 4

Step 4

Secure startup screen will appear, in this screen, user could download a new project for HMI from PC.

Figure 5

Setup Screen

Keep pressing on the right top corner of Screen for 5 seconds, HMI setup screen will appear.

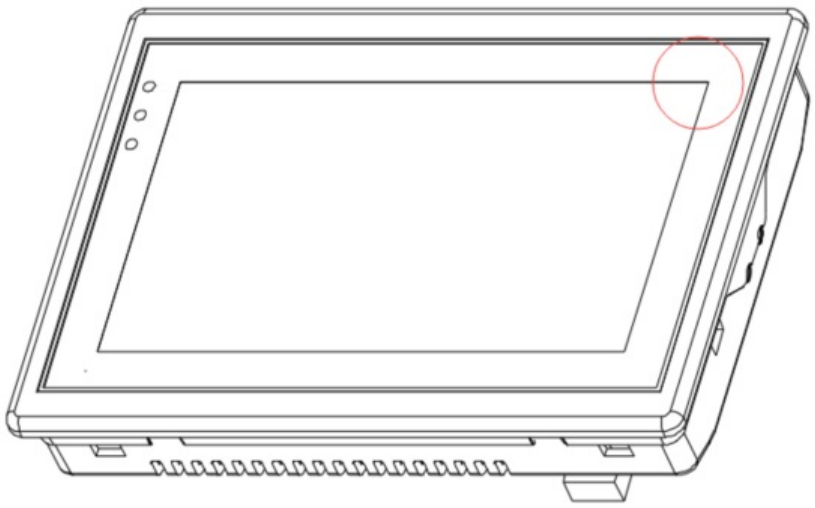


Figure 1

HMI Version:

Click on “Version” to see the version info of HMI.

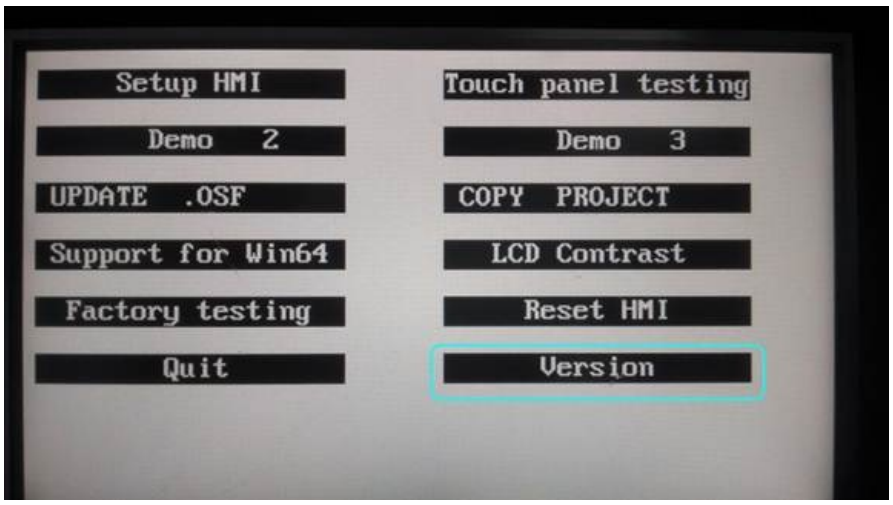


Figure 2

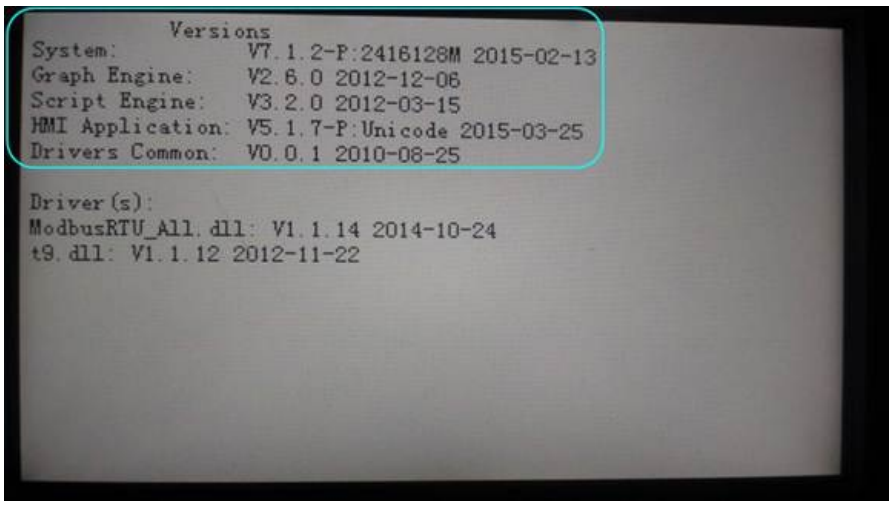


Figure 3

Communication Protocol

Check the current communication protocol in this HMI by clicking on the version; you can see the driver info of HMI.

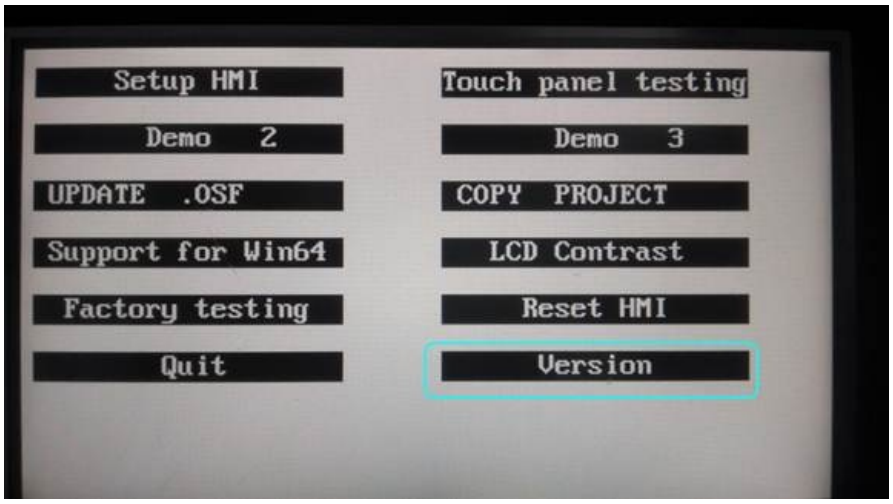


Figure 4

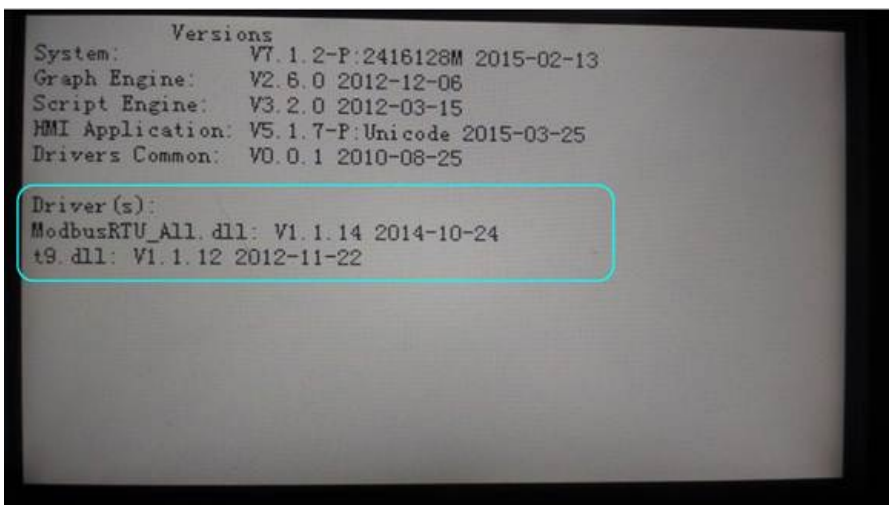


Figure 5

COM port parameters:
Click on “Setup HMI”

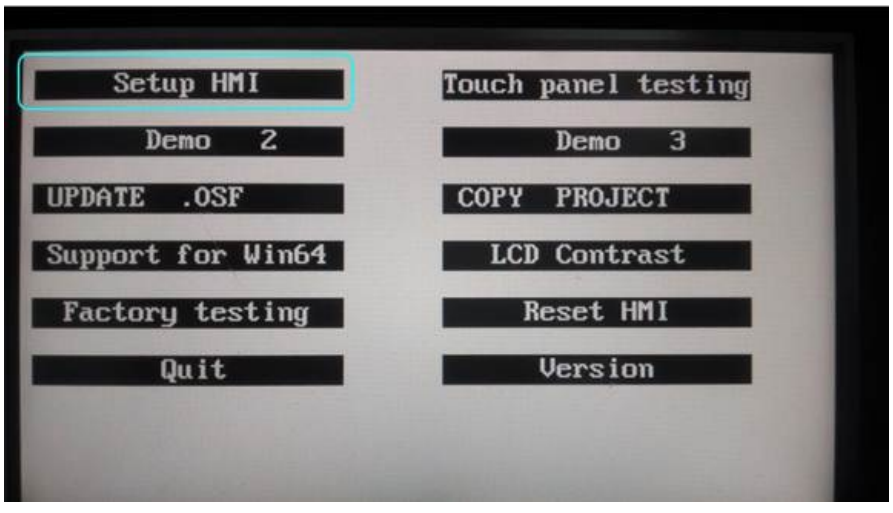


Figure 6

Figure 7

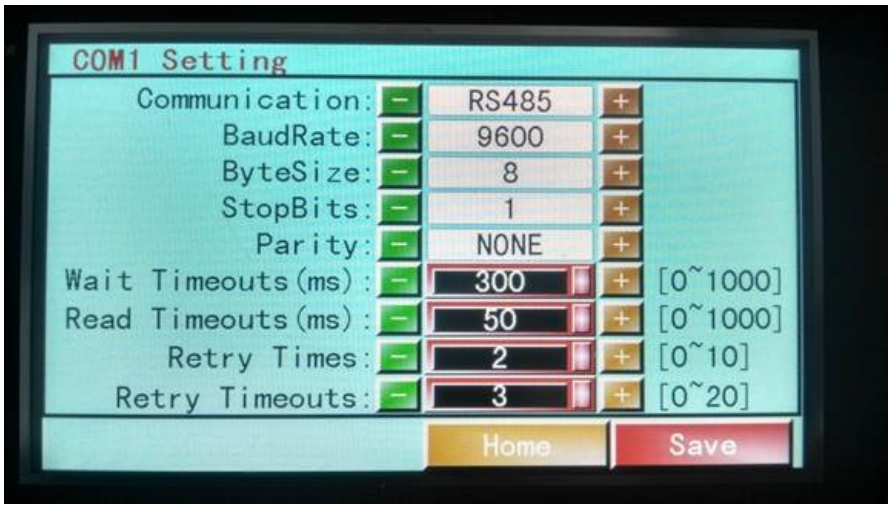


Figure 8

Figure 9

IP Setting

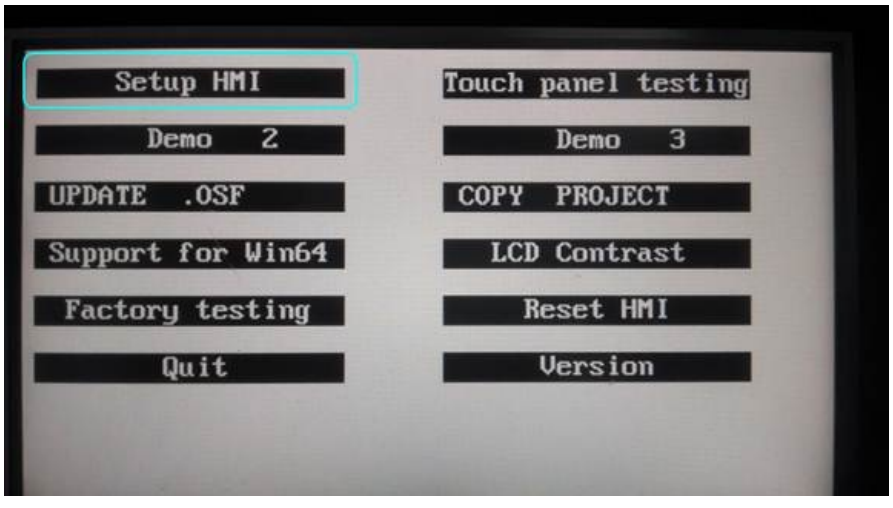


Figure 10

Figure 11

Figure 12

Touch and display setting

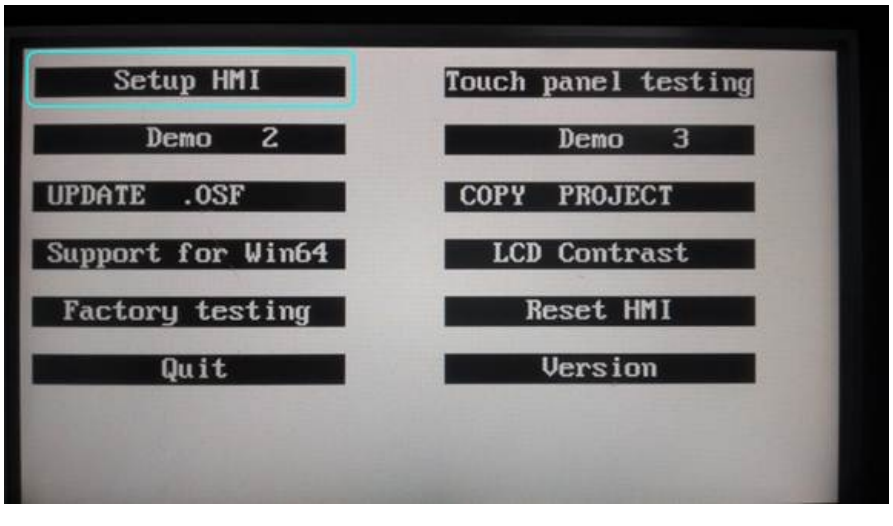


Figure 13

Figure 14

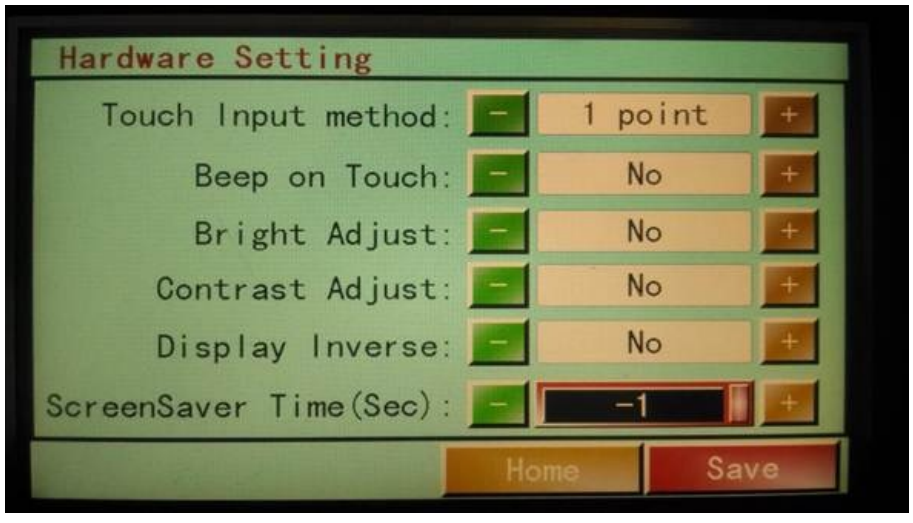


Figure 15

Time setting

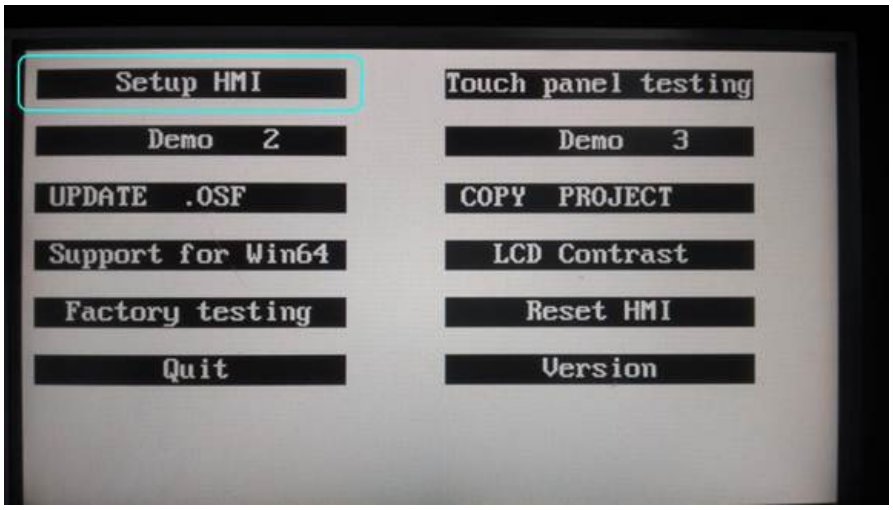


Figure 16

Figure 17

Figure 18

HMI Installation

Description

Buckle is designed to fix HMI to device.

Figure 1

Usage

On each side of the HMI is equipped with a mounting hole, tighten the screws.

Figure 2

Figure 3